International Journal of Innovative Computing, Information and Control
Volume **21**, Number **3**, June **2025**

ICIC International ©2025 ISSN 1349-4198
pp. 701–713

PARTICLE SWARM OPTIMIZED SYNERGETIC TRACKING CONTROL FOR TCP/AQM SYSTEMS

Somchai Poonyanirun* and Adirak Kanchanaharuthai

Department of Electrical Engineering
College of Engineering
Rangsit University
52/347 Muang-Ake, Phaholyothin Road, Lak-Hok, Muang, Patumthai 12000, Thailand
adirak@rsu.ac.th
*Corresponding author: somchai.po@rsu.ac.th

Received October 2024; revised January 2025

Abstract. *This paper tackles the Active Queue Management (AQM) challenge by introducing a synergetic control approach integrated with particle swarm optimization (PSO) for Transmission Control Protocol (TCP) networks. By applying this combined method, the developed control law effectively manages congestion tracking within a nonlinear TCP/AQM framework. Additionally, the nonlinear feedback congestion controller is presented, leveraging the particle swarm optimized synergetic control scheme to maintain target queue lengths and ensure window size stability. The proposed control approach undergoes validation through computer simulations. Simulation results demonstrate the proposed method's effectiveness and allow for comparison with a backstepping-like controller and an integral sliding mode controller. The results confirm that the proposed technique not only achieves effective congestion tracking control but also enhances transient performance.*

Keywords: Congestion tracking control, Adaptive control, Wireless TCP/AQM network

1. **Introduction.** The Internet is undergoing rapid expansion, leading to significant challenges in managing network traffic congestion. This growth could potentially result in network failure, lock-out conditions, and a higher chance of control-loop synchronization issues. As a result, numerous initiatives are underway to address these problems. In recent years, there has been a steady increase in interest in congestion control strategies. Fortunately, a promising solution, known as Active Queue Management (AQM), has been introduced. AQM aims to minimize packet losses, deliver reliable best-effort services with minimal packet drops, and optimize network usage. Among the various AQM techniques, Random Early Detection (RED) was the first to be proposed [1].

The main advantage of this approach is that the average queue length can be used to determine the probability of packet loss. Various techniques [2-11], based on an analytical fluid-flow model, have been explored for TCP/AQM systems to address congestion issues. To stabilize queue length and enhance robustness against external disturbances and model variations, a controller design using Linear Matrix Inequalities (LMI) [2] was introduced, combining robust active queue management with control. Mohammadi et al. [3] proposed a fuzzy-based PID controller for AQM networks in Internet routers to reduce packet loss and improve network utilization under input saturation conditions. A nonlinear model predictive congestion control strategy [4] was developed to mitigate the effects of nonlinear disturbance uncertainties, time-varying delays, and input limitations using the

DOI: 10.24507/ijicic.21.03.701

Lyapunov-Krasovskii functional. Li et al. [5] introduced an adaptive backstepping congestion controller for TCP networks with User Datagram Protocol (UDP) flows, employing the minimax method to address unknown UDP flows, which are treated as external disturbances. Another nonlinear AQM controller [6], combining integral backstepping with the minimax method, was proposed to mitigate the effects of UDP interruptions in TCP networks. Using a blend of integral backstepping and $\mathcal{H}_\infty$ design, a nonlinear congestion tracking control approach [7] was created for the TCP/AQM model. Despite external disturbances and modeling uncertainties, this controller ensured sufficient queue tracking performance and asymptotic stability of all closed-loop system signals. Wang et al. [8] introduced an adaptive fuzzy funnel congestion control strategy for TCP/AQM networks to minimize tracking errors and achieve optimal transient and steady-state performance. A backstepping sliding mode design was developed [9] for nonlinear TCP network congestion systems, incorporating unknown parameters and disturbances, using the minimax method from game theory. Liu et al. [10] proposed an adaptive neural congestion controller for TCP/AQM systems, combining finite-time theory and specified performance control to ensure the queue length converges to the reference in finite time. Additionally, a sliding mode controller [11] was designed to address parameter noise and variation in the TCP/AQM network. A backstepping-like approach [12] was suggested to solve the congestion tracking issue in TCP/AQM nonlinear models, ensuring the queue length can follow the reference queue length. Recently, a dynamic surface asymptotic control method [13] for TCP/AQM networks was introduced to improve congestion management by stabilizing queue length and enhancing tracking accuracy. This new controller outperforms existing methods in transient performance, showing faster response and reduced overshoot, with robust results confirmed through simulations.

Previous research has highlighted the complexity and challenges involved in designing nonlinear controllers. Building on this foundation, the current study introduces an advanced nonlinear controller design to address the congestion tracking control problem in TCP/AQM network models. This paper employs a synergetic control approach [14-16] for the controller design. A key limitation of synergetic control is selecting the control parameters or gains that stabilize the system and ensure good dynamic performance. Existing congestion control methods, such as backstepping and sliding mode control, exhibit inefficiencies in nonlinear systems, including slow response times and sensitivity to parameter tuning. These approaches often lack robustness and adaptability, particularly under dynamic and time-varying network conditions. By integrating Particle Swarm Optimization (PSO) with synergetic control, the proposed method overcomes these challenges, automating parameter tuning and enhancing system stability and performance in nonlinear TCP/AQM systems. The goal of this approach is to develop a stabilizing feedback controller that achieves congestion tracking control in the TCP/AQM network model, ensuring satisfactory dynamic and steady-state performance. Moreover, the proposed control method outperforms backstepping-like control, and integral sliding mode control by achieving asymptotic tracking with zero error.

The main contributions of this work are as follows.

- This study introduces a particle swarm optimized synergetic control method to solve the congestion tracking control problem in TCP/AQM networks, leveraging a nonlinear TCP/AQM dynamic model that has not been previously examined.
- Through Lyapunov theory, it is demonstrated that all signals within the closed-loop system remain bounded, achieving asymptotic tracking.

- The proposed design technique is both simple and highly effective, outperforming backstepping-like and integral sliding mode controllers, with enhanced dynamic performance, including reduced overshoot and quicker oscillation damping.

The structure of this paper is as follows. Section 2 offers an overview of the TCP/AQM network system's dynamic model and the problem statement. Section 3 details the nonlinear synergetic control design. Section 4 provides a brief introduction to particle swarm optimization. Section 5 presents simulation results validating the effectiveness of the proposed design. Finally, Section 6 concludes the paper.

2. **TCP/AQM System Model.** The dynamic model used in this paper is derived from the fluid model of the TCP congestion-avoidance method, as outlined in [17, 18]. Based on the results presented in [17, 18], the nonlinear TCP/AQM model can be expressed as

$$\begin{cases} \dot{W}(t) = \dfrac{1}{R(t)}(1 - p(t)) - \dfrac{W(t)}{2}\dfrac{W(t)}{R(t)}p(t) \\ \dot{q}(t) = \dfrac{NW(t)}{R(t)} - C(t) \\ R(t) = T_p + \dfrac{q(t)}{C(t)} \end{cases} \tag{1}$$

where $W(t) \in [W_{\min}, W_{\max}]$ denotes the TCP window size, $R(t)$ refers to the round-trip delay, and $p(t)$ represents the fraction of packets that are marked or dropped in the queue, with the constraint $0 \le p(t) \le 1$. N denotes the number of TCP connections. The queue length is indicated by $q(t) \in [q_{\min}, q_{\max}]$, $C(t)$ signifies the link capacity, and T_p indicates the propagation delay. In this work, $C(t)$ is assumed to be constant and is referred to simply as C.

The dynamic behavior of the window size in Equation (1) follows the "additive increase, multiplicative decrease" principle [17, 18]. The first term of $\dot{W}(t)$, $1/R(t)$, indicates that the window size increases by one for every round-trip time $R(t)$. In contrast, the second term, $\frac{W(t)}{2}\frac{W(t)}{R(t)}$, represents a halving of the window size when congestion occurs, resulting in packet loss. Similarly, the first term of $q(t)$, $N/R(t)W(t)$, describes a newly arriving queuing packet.

To streamline the state-space equation of the system (1), we introduce the following state variables:

$$x_1 = q - q_r, \quad x_2 = -C + \frac{NW}{R} \tag{2}$$

To ensure that the queue length q accurately tracks the target queue length q_r, we introduce an additional state variable aimed to eliminate the error between them, defined as $x_0 = \int_0^t (q(\tau) - q_r)d\tau$. As a result, the vector of state variables used in this design procedure is expressed as $x = [x_0, x_1, x_2]^T = \left[\int_0^t (q(\tau) - q_r)d\tau, q - q_r, -C + \frac{NW}{R}\right]^T$. By differentiating these state variables, we can reformulate the dynamic model of the nonlinear TCP/AQM system into an affine nonlinear system, as follows:

$$\dot{x} = f(x) + g(x)u(x) \tag{3}$$

with

$$f(x) = \begin{bmatrix} f_0(x_0) \\ f_1(x_0, x_1) \\ f_2(x_0, x_1, x_2) \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ \dfrac{N}{R^2} \end{bmatrix}, \quad x = \begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix}$$

$$g(x) = \begin{bmatrix} 0 \\ 0 \\ g_2(x) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -\left(\dfrac{N}{R^2} + \dfrac{(x_2+C)^2}{2N}\right) \end{bmatrix}, \quad u(x) = p(t) \tag{4}$$

Additionally, the operating region is defined as the set $\mathcal{D} = \{x \in \mathbb{R} \times \mathbb{R} \times \mathbb{R}\}$. The open-loop operating equilibrium is represented by $x_e = [x_{0e}, x_{1e}, x_{2e}]^T = [0, 0, 0]^T$.

Problem statement: The goal of this paper is to design a nonlinear controller p to address the queue tracking problem in a congestion control scheme using the synergetic control method combined with a particle swarm optimization. The proposed controller meets the following desired criteria: (i) the queue length q can accurately track the desired queue length q_r; (ii) the window size W remains stable and bounded; (iii) the fraction of packets marked for dropping in the queue is minimized; and (iv) the entire closed-loop system is asymptotically stable at the equilibrium point x_e, with all resulting closed-loop signals remaining bounded.

The following assumption is established to achieve the aforementioned objectives.

Assumption 2.1. *All state variables $x_0, x_1, x_2 \in \mathbb{R}$, are assumed to be measurable.*

In the following section, the particle swarm-optimized synergetic congestion tracking control design is developed to develop a feedback stabilizing nonlinear control. This control is designed in a step-by-step manner to achieve the desired performance outcomes.

3. Design of Synergetic Controller.

3.1. Synergetic control method.

According to the findings in [14], the synergetic control scheme is an invariant-manifold-based control method suitable for controlling nonlinear, dynamic, and high-dimensional systems. As previously mentioned in Section 1, this method has been successfully applied in various practical applications [7-19]. The synergetic control approach, based on the Analytical Design of Aggregated Regulator (ADAR) method [14-16], is briefly introduced.

Let us consider an n-dimensional nonlinear dynamic equation[1] in the state space form as

$$\dot{x}(t) = f(x, u, t) \tag{5}$$

where $x \in \mathbb{R}^n$ represents the system state variable vector, $u \in \mathbb{R}^m$ is the control input vector, to be designed, t denotes time, and $x_e \in \mathbb{R}^n$ is a selectable equilibrium point to be stabilized, respectively. The synergetic controller design is summarized in the following steps.

- **Step 1:** Define a macro-variable as $\varphi(x)$ where $\varphi(x)$ is a function of the system states. In line with synergetic control theory, this macro-variable is used to determine a stabilizing control law $u(x) = u(x, \varphi(x))$ that guides the system trajectories toward the desired manifold $\mathcal{M}$, defined by $\varphi(x) = 0$, thus fulfilling the desired control specifications.
- **Step 2:** Develop a control law that can guide the system states onto the designated manifold $\mathcal{M}$ and maintain them on this manifold thereafter, subject to an evolution constraint that can be represented by the following equation.

$$T\dot{\varphi}(x) + \varphi(x) = 0, \quad T > 0 \tag{6}$$

where T is a controller parameter that influences the rate at which the system trajectories converge to the manifold $\mathcal{M}$.

[1] It is assumed that throughout this paper all functions and mappings are $\mathbb{C}^\infty$.

- **Step 3:** Compute the time derivative of the chosen macro-variable $\varphi(x)$ concerning the system variable x, applying the chain rule of differentiation. Then, substitute either (3) or (5) into (6), yielding the following result:

$$T\frac{\partial\varphi(x)}{\partial x}f(x,u,t)+\varphi(x)=0 \tag{7}$$

By appropriately defining a suitable macro-variable and choosing the control parameter T, the expression above (7) can be directly solved to obtain the desired controller $u(x)$ which can be expressed as follows:

$$u(x)=\phi(x,\varphi(x),T,t) \tag{8}$$

From (8), it is evident that once the evolution constraint in (7) is resolved, we obtain an analytical control law that ensures the desired control specifications are met. This resulting control law depends on the system state variable, the chosen macro-variable, and the control parameter T. Depending on the control parameters selected by the designer, various desirable characteristics for the overall closed-loop dynamics, such as global stability, insensitivity to parameters, and specific dynamic properties, can be achieved.

3.2. **Synergetic control design.** In this subsection, the design procedure based on synergetic control scheme above is developed step by step. From (4), an error coordinate variable is defined as

$$e=\begin{bmatrix} e_0 \\ e_1 \\ e_1 \end{bmatrix}=\begin{bmatrix} x_0-x_{0e} \\ x_1-x_{1e} \\ x_2-x_{2e} \end{bmatrix}=\begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix} \tag{9}$$

Step 1: In order to apply the synergetic control to the modeled system, based on the error variables (9), let us introduce the following macro-variable:

$$\varphi(x)=c_0e_0+c_1e_1+c_2e_2 \tag{10}$$

where c_0, c_1 and c_2 are positive design parameters.

Step 2: The objectives of the proposed controller design are to direct the system trajectories and ensure they stay on the desired manifold $\varphi(x)=0$ hereafter. In the dynamics of the evolution, the macro-variable is defined as follows:

$$T\dot{\varphi}+\varphi=0, \quad T>0. \tag{11}$$

where T represents the predefined controller parameters that indicate the convergence speed of the closed-loop dynamics to the desired manifold $\varphi(x)=0$.

Therefore, the time derivative of the error variables in (10) along the system trajectory in (4) is expressed as follows:

$$\dot{\varphi}(x)=-\frac{1}{T}\varphi(x)=c_0\dot{e}_0+c_1\dot{e}_1+c_2\dot{e}_2 \tag{12}$$

Step 3: To design a stabilizing controller, the Lyapunov stability strategy will be employed to assess the stability of the overall closed-loop dynamics. Accordingly, we define a Lyapunov function as follows:

$$V(\varphi,t)=\frac{1}{2}\varphi^2 \tag{13}$$

The time derivative of the Lyapunov function can be stated as

$$\dot{V}(t)=\varphi\dot{\varphi} \tag{14}$$

After differentiating (13) and substituting (12) into (14), we have

$$\dot{V}(t)=\varphi(c_0\dot{e}_0+c_1\dot{e}_1+c_2\dot{e}_2) \tag{15}$$

According to Lyapunov stability theory, $\dot{V}(t)$ must be a negative definite function. Therefore, a suitable control law is selected as

$$u(x) = -\frac{1}{g_2}\left[f_2 + \frac{1}{c_2}\left(\frac{\varphi}{T} + c_0x_1 + c_1x_2\right)\right] \quad (16)$$

This controller $u(x)$ is a nonlinear function that is affected by the choice of the control parameters:

$$C = [c_0, c_1, c_2, T]$$

The setting of these control parameters or gains is the topic of the next section.

4. **Particle Swarm Optimization (PSO) Algorithm.** Particle Swarm Optimization (PSO) [25] is an optimization technique based on the collective behavior of birds or fish. It utilizes a swarm of particles (representing potential solutions) that navigate through the search space to discover the optimal solution. This algorithm can be successfully integrated into control system designs, including nonlinear predictive control [27] and nonlinear PID control [28-30]. The control gains are represented as $C = [c_0, c_1, c_2, T]$. In the TCP/AQM system, four control parameters must be simultaneously selected to ensure the closed-loop system achieves asymptotic stability. The goal is to minimize a performance index, or objective function J, such as the Integral of Time-weighted Absolute Error (ITAE), by adjusting the control gains. The problem is formulated as follows:

$$J = \int_0^T t^2(|x_0| + |x_1| + |x_2|)dt$$

where T denotes the final time for simulation. The ITAE is commonly used to evaluate control systems, with a lower value indicating better performance (i.e., faster error reduction and better system stability).

The steps of the PSO algorithm are as follows:

1) **Initiation:**
- **Define the Problem:** Identify the objective function J that needs to be minimized. This function will evaluate the performance of the controller with the selected parameters.
- **Initialize Parameters:**
  - Number of particles N: Choose the number of particles in the swarm.
  - Dimension D: Set $D = 4$ for the four controller parameters.
  - Initialize each particle's position $\mathbf{x}_i$ (controller parameters) randomly within defined bounds:

$$\mathbf{x}_i = [c_0, c_1, c_2, T]$$

  - Initialize each particle's velocity $\mathbf{v}_i$ randomly:

$$\mathbf{v}_i = [v_0, v_1, v_2]$$

  - Initialize the admissible range for the controller gains $c_i^{\min} < c_i < c_i^{\max}$ for $i = 0, 1, 2$, and $T^{\min} < T < T^{\max}$.
- **Set PSO Parameters:**
  - Inertia weight w: Control the impact of the previous velocity on the current velocity.
  - Acceleration coefficients k_1 and k_2: Control the influence of the particle's best position and the swarm's best position.
  - Maximum number of iterations $\max_{iter}$: Set a stopping criterion for the algorithm.

2) **Evaluate the Initial Particles:** For each particle, evaluate the objective function $f(\mathbf{x}_i)$ to determine the fitness of each particle:

$$\mathit{fitness}_i = f(\mathbf{x}_i)$$

3) **Update Personal and Global Best:**
   - Initialize personal best $\mathbf{p}_i$ for each particle as its initial position: $\mathbf{p}_i = \mathbf{x}_i$
   - Initialize global best $\mathbf{g}$ as the position of the best particle found so far:

$$\mathbf{g} = \arg\min(\mathit{fitness}_i) \text{ for all } i$$

4) **Main PSO Loop:** Repeat the following steps until the maximum number of iterations is reached or convergence criteria are met:
   - **Update Velocity and Position:** For each particle i:
     – Update the particle's velocity:

$$\mathbf{v}_i = w\mathbf{v}_i + k_1 r_1(\mathbf{p}_i - \mathbf{x}_i) + k_2 r_2(\mathbf{g} - \mathbf{x}_i)$$

     where r_1 and r_2 are random numbers between 0 and 1.
     – Update the particle's position: $\mathbf{x}_i = \mathbf{x}_i + \mathbf{v}_i$.
   - **Boundary Conditions:** Ensure that the updated position $\mathbf{x}_i$ remains within the defined bounds. If it exceeds the bounds, clamp it back to the nearest bound.
   - **Evaluate Fitness:** Calculate the fitness of the updated position: $\mathit{fitness}_i = f(\mathbf{x}_i)$.
   - **Update Personal Best:** If the new fitness is better than the personal best:

$$\text{if } \mathit{fitness}_i < f(\mathbf{p}_i) \text{ then } \mathbf{p}_i = \mathbf{x}_i$$

   - **Update Global Best:** If the new fitness is better than the global best:

$$\text{if } \mathit{fitness}_i < f(\mathbf{g}) \text{ then } \mathbf{g} = \mathbf{x}_i$$

5) **Output:** After completing the iterations, the optimal controller parameters will be the values in the global best position $\mathbf{g}$.
6) **Conclusion:** The PSO algorithm iteratively searches for the optimal values of the four controller parameters, enhancing the performance of the TCP/AQM control system based on the objective function defined at the beginning.

Remark 4.1. *The proposed particle swarm optimized synergetic control method has several key limitations, including computational complexity, sensitivity to parameter tuning, and reduced performance in high-delay or noisy networks. These challenges highlight areas for future improvement. Furthermore, future work has been expanded to include specific directions, such as developing adaptive control mechanisms for unknown parameters, validating the method in real-world network environments, exploring hybrid optimization strategies, addressing multi-objective optimization, and studying scalability in complex networks. These efforts aim to enhance the method's robustness, adaptability, and practical applicability.*

5. **Simulation Results.** In this section, the effectiveness of the proposed controller is evaluated and verified in computer simulations under the following parameters of the networks.

$$C = 1750 \text{ packets/s}, \quad T_p = 0.1 \text{ s}, \quad q_r = 100 \text{ packets}$$

In the simulations, the initial parameters are set to $q_0 = 99$, $W_0 = 4$, $x_0 = 1$. Following [25, 26] for PSO parameters, the search space has a dimension of 4, with a population size of 20. The acceleration factors are $k_1 = k_2 = 2.05$, and the inertial weight factor w ranges from $[0.4, 0.9]$. The maximum count is 50. The admissible range for each controller gain is as follows: $0 \leq c_1 \leq 100$, $0 \leq c_2 \leq 100$, $0.01 \leq c_3 \leq 100$, and $0.0001 \leq T \leq 0.1$.

Parameters are initialized at $[0.5, 1.5, 2, 0.001]$, and, after running multiple independent trials to reduce random variation, the optimized controller values are finalized as $c_1 = 99.1440$, $c_2 = 3.2024$, $c_3 = 0.0100$, $T = 0.0004$ and $J = 6.1098$.

Time-domain simulations are conducted using computer simulations to evaluate the dynamic performance of the designed controller, as described in (16), in the system being examined. The effectiveness of the proposed controller is compared with that of the following controllers:

- Backstepping-Like Controller (BSLC) [12]

$$u = p = -\frac{1}{g_2(x_0, x_1, x_2)} \left[\frac{c_2 Q + P + x_1 + c_1 x_2 + c_1 \dot{P} + \dot{f}_1(x_0, x_1) + \dot{g}_1(x_0, x_1) x_2}{g_1(x_0, x_1)} + f_2(x_0, x_1, x_2) \right] \tag{17}$$

where

$$P = c_0 x_0 + x_1, \quad \dot{P} = c_0 x_1 + x_2, \quad Q = x_0 + c_0 x_1 + c_1 P + f_1(x_0, x_1) + g_1(x_0, x_1) x_2.$$

The controller parameters of the backstepping-like control law are chosen as follows: $c_i = 0.5$, $i = 0, 1, 2$.

- Integral Sliding Mode Controller (ISDM)

$$u = p = -\frac{(K\text{sign}(S) + c_2(f_1(x_0, x_1) + g_1(x_0, x_1)x_2) + c_2 f_2(x_0, x_1, x_2))}{c_2 g_2(x_0, x_1, x_2)} \tag{18}$$

where $S(x) = c_1 x_1 + c_2 x_2$. The controller parameters of the integral sliding mode control law are chosen as follows: $c_1 = 2$, $c_2 = 1$.

The simulation results illustrate the effectiveness of the designed method based on the following criteria: the tracking error between the actual queue length and the target queue length approaches zero; the window size stays within bounds; and the queue's packet loss ratio is low, remaining within the interval $[0, 1]$.

The simulation findings are presented and discussed below. Figures 1 and 2 show the time histories of the following state variables using the proposed technique: the integral of the error between the queue length and the desired queue length x_0, the queue length q, the window size W, the state variable x_2, the packet loss ratio p, and the macro-variable $\varphi(x)$. Figure 3 illustrates the time histories of the same state variables – x_0, q, W, x_2 and p – under the BSLC design. In Figure 4, the time responses of x_0, q, W, x_2, p, and the sliding surface $S(x)$ are shown under the ISDM design. The figures clearly illustrate that the proposed control law provides substantially improved dynamic performance compared to the other methods. Specifically, the time responses of the proposed controller reach the desired equilibrium point in just 0.5 seconds, whereas the other methods require over 3 seconds. Nevertheless, the packet loss ratio $u = p(t)$ for the proposed method is greater than that of the other methods. Notably, the tracking error quickly converges to zero ($q \to q_r$) with both the proposed and BSLC methods, while this is not the case for the ISDM method. This indicates that the queue length closely and smoothly follows the target queue length. Additionally, the proposed method improves other important aspects of dynamic performance, such as decreasing oscillatory overshoot and reducing rise and settling times. As shown in Figure 2, the macro-variable $\varphi(x)$ also stabilizes at zero rapidly.

It is crucial to highlight that although the other controllers achieve the objectives presented in the paper, their transient performance falls short compared to the proposed controller. Figure 3 demonstrates that while the BSLC scheme responds faster than the

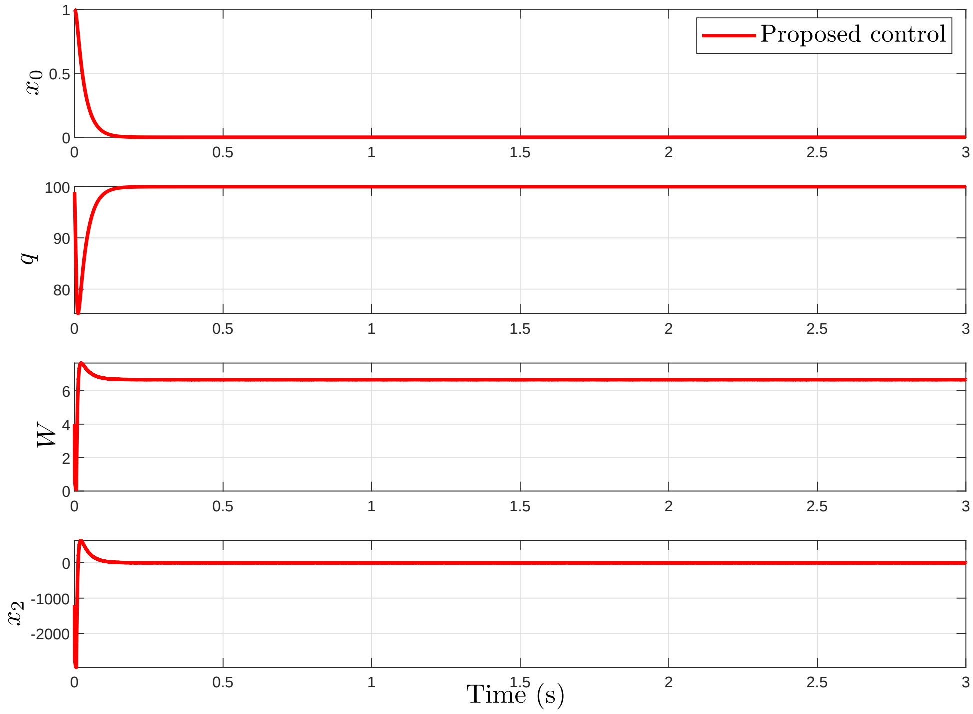


FIGURE 1. Controller performance – $x_0 = \int_0^t (q(\tau) - q_r) d\tau$, the queue length q, window size W, and x_2 under the proposed controller

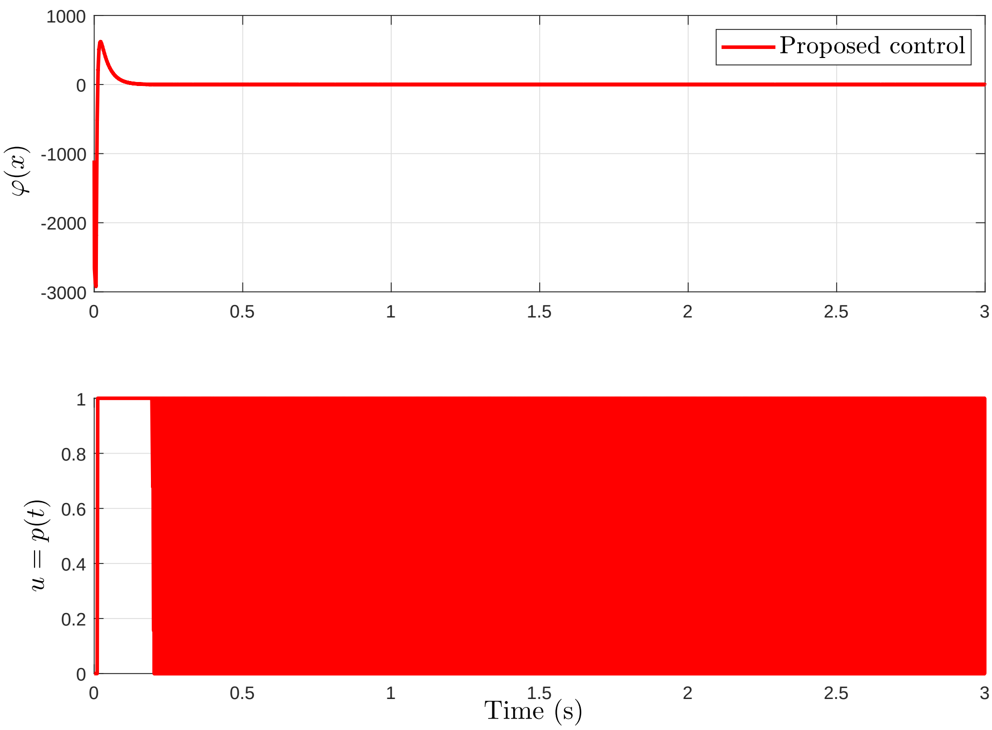


FIGURE 2. Controller performance – The macro-variable $\varphi(x)$ and packet loss ratio $u = p(t)$ under the proposed controller

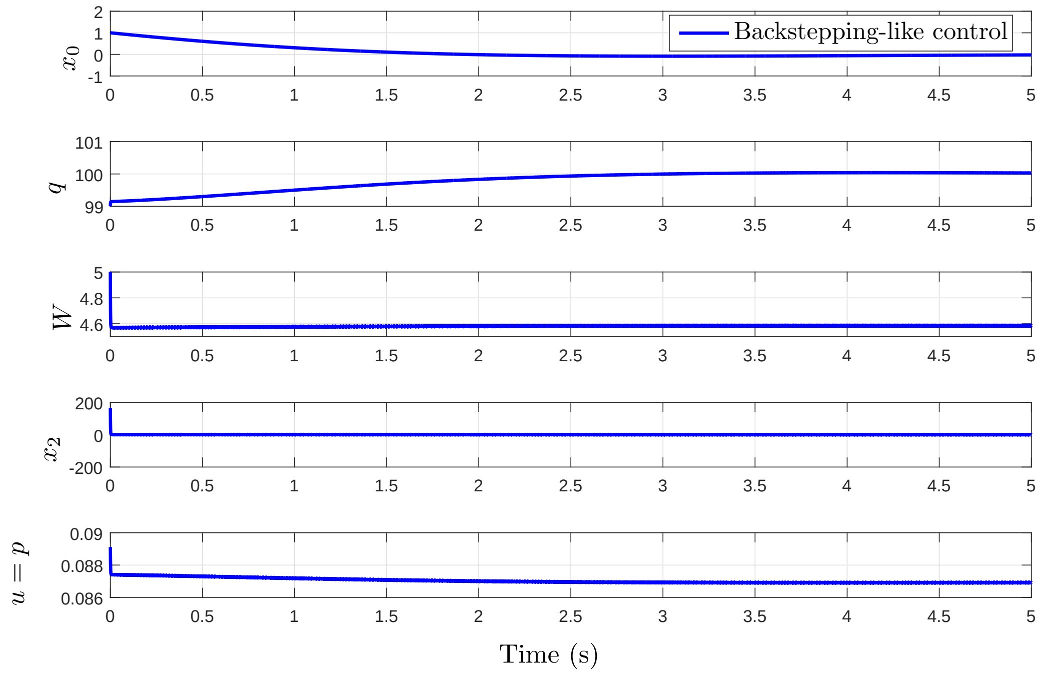


FIGURE 3. Controller performance – $x_0 = \int_0^t (q(\tau) - q_r) d\tau$, the queue length q, window size W, x_2, and packet loss ratio ($u = p$) under the backstepping-like controller

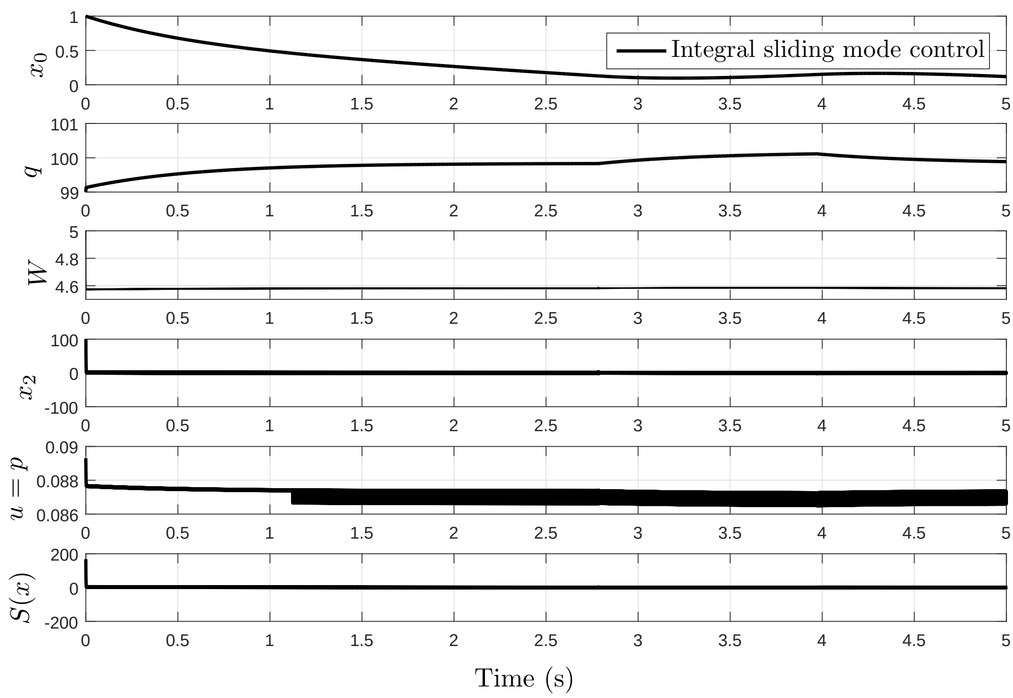


FIGURE 4. Controller performance – $x_0 = \int_0^t (q(\tau) - q_r) d\tau$, the queue length q, window size W, x_2, packet loss ratio ($u = p$) and the sliding surface $S(x)$ under the integral sliding mode controller

ISDM method, it is still considerably slower than the proposed controller's response time. Figure 4 shows that chattering effects prevent the tracking error between the queue length and desired queue length from reaching zero. Additionally, the ISDM method's sliding surface oscillates around zero rather than settling at zero, due to unwanted chattering effects in the control input (packet loss ratio). Furthermore, Figures 2-4 confirm that the packet loss ratio $u = p(t)$ remains within the required interval $[0, 1]$ when using the proposed controller, the BSLC controller, and the ISDM controller.

The quantitative results for the queue length q are presented in Table 1. This table summarizes key performance metrics – such as steady-state error, rise time, and overshoot – for the proposed controller in comparison to the baseline controllers: the Backstepping-Like Controller (BSLC) and the Integral Sliding Mode Controller (ISDM).

Table 1. Quantitative results in terms of steady-state error, rise time, and percentage overshoot for the actual queue length with baseline controllers

	Steady-state error	Rise time [s]	Percentage overshoot [%]
Proposed control	0	0.18	0
BSLC	0.0519	2.48	0.0052
ISDM	0.1113	3.00	0.1158

Table 1 highlights the superior performance of the proposed controller compared to the baseline methods across various metrics. For steady-state error, the proposed controller achieves perfect accuracy with a value of zero, ensuring the actual queue length matches the desired value, while the baseline controllers exhibit small but noticeable steady-state errors. Regarding rise time, the proposed controller reaches the desired equilibrium in just 0.18 seconds, significantly faster than the baseline controllers, which take over 2.48 seconds and 3 seconds, respectively. Furthermore, the proposed controller eliminates overshoot entirely, offering a marked improvement over the baseline methods and delivering smoother and more stable transient responses.

In conclusion, the simulation results demonstrate that the proposed method outperforms the BSLC and ISDM methods. It is evident that, in accordance with the desired criteria, the queue length q quickly tracks the target queue length q_r, while the window size W remains stable under the suggested control strategy. Furthermore, this approach showcases improved dynamic properties, as reflected by the rapid dampening of oscillations throughout all time trajectories.

6. **Conclusion.** This study presents a nonlinear controller designed to tackle the queue tracking problem for congestion control, utilizing the particle swarm optimized synergetic congestion tracking technique. The simulation results reveal that the proposed control method is effective; it ensures stability and boundedness in both the window size and all closed-loop system trajectories, while also enabling tracking errors between the actual queue and the reference queue lengths to converge to zero rapidly. Additionally, the design process shows superior performance compared to the backstepping-like and integral sliding mode methods in terms of transient control. Moreover, the results demonstrate that the proposed controller effectively addresses the congestion tracking issue and enhances transient performance in the closed-loop system dynamics, meeting the desired objectives. Future work will focus on developing adaptive synergetic control to handle unknown or time-varying parameters and testing the proposed method on real-world network topologies to evaluate its practical applicability. Additionally, exploring hybrid optimization

strategies and multi-objective frameworks could further enhance the robustness and scalability of the approach.

REFERENCES

[1] S. Floyd and V. Jacobson, Random early detection gateways for congestion avoidance, *IEEE/ACM Transactions on Networking*, vol.1, no.4, pp.397-413, 1993.

[2] C. Zhou, J. He and Q. Chen, A robust active queue management scheme for network congestion, *Computer & Electrical Engineering*, vol.39, no.2, pp.285-294, 2013.

[3] S. Mohammadi, H. M. Pour, M. Jafari and A. Javadi, Fuzzy-based PID active queue manager for TCP/IP networks, *Proc. of the 10th International Conference on Information Science, Signal Processing and Their Applications (ISSPA 2010)*, Kuala Lumpur, Malaysia, pp.434-439, 2010.

[4] C. Han, M. Li, Y. Jing, L. Lei, Z. Pang and D. Sun, Nonlinear model predictive congestion control for networks, *IFAC-PapersOnLine*, vol.50, no.1, pp.552-557, 2017.

[5] Z. H. Li, Y. Liu and Y.-W. Jing, Design of adaptive backstepping congestion controller for TCP networks with UDP flows based on minimax, *ISA Transactions*, vol.95, pp.27-34, 2019.

[6] Z. H. Li, Y. Liu and Y.-W. Jing, Acitve queue management algorithm for TCP networks with integral backstepping and minimax, *International Journal of Control, Automation and Systems*, vol.17, no.4, pp.1059-1066, 2019.

[7] Y. Liu, X. Liu, Y. Jing, Z. Zhang and X. Chen, Congestion tracking control for uncertain TCP/AQM network based on integral backstepping, *ISA Transactions*, vol.89, pp.131-138, 2019.

[8] K. Wang, Y. Liu, X. Liu, Y. Jing and S. Zhang, Adaptive fuzzy funnel congestion control for TCP/AQM network, *ISA Transactions*, vol.95, pp.11-17, 2019.

[9] K. Wang, Y. Jing and S. Zhang, Minimax controller design based on backstepping sliding mode for TCP network systems, *Proc. of 2017 29th Chinese Control and Decision Conference (CCDC)*, Chongqing, China, pp.764-768, 2017.

[10] Y. Liu, Y. Jing and X. Chen, Adaptive neural practically finite-time congestion control for TCP/AQM network, *Neurocomputing*, vol.351, pp.26-32, 2019.

[11] M. Jalili-Kharaajoo, Sliding mode queue management in TCP/AQM networks, *Proc. of International Conference on Telecommunication (ICT 2004)*, pp.638-643, 2004.

[12] K. Wongsurith, A. Kanchanaharuthai, S. Poonyanirun and P. Ngamsom, A backstepping-like congestion tracking control for TCP/AQM systems, *Kasem Bundit Engineering Journal*, vol.12, pp.18-33, 2022.

[13] S. Poonyanirun and A. Kanchanaharuthai, Dynamic surface asymptotic congestion tracking control for TCP/AQM systems, *International Journal of Innovative Computing, Information and Control*, vol.19, no.1, pp.73-86, 2023.

[14] A. Kolesnikov, G. Veselov et al., *Modern Applied Control Theory: Synergetic Approach in Control Theory*, vol.2, TSURE Press, Moscow, Taganrog, 2000 (in Russian).

[15] A. Kolesnikov, Introduction of synergetic control, *2014 American Control Conference*, pp.3013-3016, 2014.

[16] A. Kolesnikov, G. Veselov, A. Kolesnikov, A. Monti, F. Ponci, E. Santi and R. Dougal, Synergetic synthesis of DC-DC boost converter controllers: Theory and experimental analysis, *APEC. 17th Annual IEEE Applied Power Electronics Conference and Exposition (Cat. No. 02CH37335)*, Dallas, TX, USA, pp.409-415, 2002.

[17] V. Misra, W. B. Gong and D. Towsely, Fluid-based analysis of a network of AQM routers supporting TCP flows with an application to RED, *Proc. of ACM/SIGCOMM*, pp.151-160, 2000.

[18] C. V. Hollot, V. Misra, D. Towsley and W. Gong, Analysis and design of controllers for AQM routers supporting TCP flows, *IEEE Transactions on Automation Control*, vol.47, pp.945-959, 2002.

[19] E. Santi, A. Monti, D. Li, K. Proddutur and R. A. Dougal, Synergetic control for DC-DC boost converter: Implementation options, *IEEE Transactions on Industry Applications*, vol.39, no.6, pp.1803-1813, 2003.

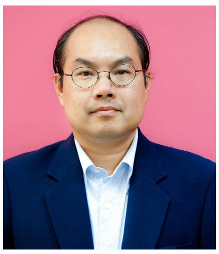
[20] P. Zhao, W. Yao, S. Wang, J. Wen and S. Cheng, Decentralized nonlinear synergetic power system stabilizers design for power system stability enhancement, *International Transactions on Electrical Energy Systems*, vol.24, no.9, pp.1356-1368, 2014.

[21] A. Kanchanahanathai and A. Boonyaprapasorn, A synergetic control design of generator excitation and STATCOM of power systems, *Rangsit Journal of Arts and Sciences*, vol.4, no.1, pp.59-70, 2012.

[22] P. Zhao, W. Yao, J. Wen, L. Jiang, S. Wang and S. Cheng, Improved synergetic excitation control for transient stability enhancement and voltage regulation of power systems, *International Journal of Electrical Power & Energy Systems*, vol.68, 2015.
[23] G. Veselov, A. Sklyarov, S. Sklyarov and V. Semenov, Synergetic approach to the quadrotor helicopter control in an environment with external disturbances, *2016 International Siberian Conference on Control and Communications (SIBCON)*, pp.1-6, 2016.
[24] J. Qian, K. Li, H. Wu, J. Yang and X. Li, Synergetic control of grid-connected photovoltaic systems, *International Journal of Photoenergy*, vol.2017, Article ID 5051489, 2017.
[25] T. M. Shami, A. A. El-Saleh, M. Alswaitti, Q. Al-Tashi, M. A. Summakieh and S. Mirjalili, Particle swarm optimization: A comprehensive survey, *IEEE Access*, vol.10, pp.10031-10061, 2022.
[26] W. Zhang, D. Ma, J. J. Wei and H. F. Liang, A parameter selection strategy for particle swarm optmization based on particle positions, *Expert Systems with Applications*, vol.41, pp.3576-3584, 2014.
[27] J. Zietkiewicz, P. Kozierski and W. Giernacki, Particle swarm optimisation in nonlinear model predictive control; comprehensive simulation study for two selected problems, *International Journal of Control*, vol.94, pp.2623-2639, 2020.
[28] X. Sun, N. Liu, R. Shen, K. Wang, Z. Zhao and X. Sheng, Nonlinear PID controller parameters optimization using improved particle swarm optimization algorithm for the CNC system, *Applied Sciences*, vol.12, no.20, 10269, 2022.
[29] S. Charkoutsis and M. Kara-Mohamed, A Particle Swarm Optimization tuned nonlinear PID controller with improved performance and robustness for First Order Plus Time Delay systems, *Results in Control and Optimization*, vol.12, 100289, 2023.
[30] Z. Maeva, C. Zoleko, A. S. T. Kammogne, M. S. Siewe, A. T. Azar, S. Ahmed and I. A. Hameed, Optimized nonlinear PID control for maximum power point tracking in PV systems using particle swarm optimization, *Mathematical and Computational Applications*, vol.29, no.5, 88, 2024.

Author Biography

Somchai Poonyanirun received the B.Eng. degree in Telecommunication Engineering from King Mongkut's Institute of Technology Ladkrabang, Thailand, M.Eng. degree in Electrical Engineering from Chulalongkorn University in 1995 and 1998, respectively. Currently, he is a Lecturer at the Department of Electrical Engineering at Rangsit University, Thailand. His research interests include communication system and control together with congestion tracking control for TCP/AQM network systems.



Adirak Kanchanaharuthai received the B.Eng. degree in Control Engineering from King Mongkut's Institute of Technology Ladkrabang, Thailand, the M.Eng. degree in Electrical Engineering from Chulalongkorn University, Thailand, and the Ph.D. degree in Systems and Control Engineering from Case Western Reserve University, Cleveland, OH, USA, in 1995, 1997, and 2012, respectively. Currently, he is an Associate Professor at the Department of Electrical Engineering, Rangsit University, Thailand. His main research interests include power system dynamics, stability and control, renewable energy systems, energy storage systems, and applications of nonlinear control theory to power systems, Flexible Alternating Current Transmission Systems (FACTS) devices, and congestion tracking control for TCP/AQM network systems.