



# Analytical Study for Cyber Dynamic Continuous Time Vehicle Model with Networked PID Controller

Qasem Abu Al-Haija

Department of Cybersecurity, Princess Sumaya University for Technology (PSUT), Amman 11941, Jordan

Corresponding Author Email: [q.abualhaija@psut.edu.jo](mailto:q.abualhaija@psut.edu.jo)

<https://doi.org/10.18280/mmep.090630>

## ABSTRACT

**Received:** 30 August 2022

**Accepted:** 20 November 2022

**Keywords:**

Cyber-Physical System (CPS), dynamic vehicle model, continuous-time model, PID controller, Transceiver, SimEvents, Simulink

This paper uses different physical scenarios to analyze the cyber dynamic continuous time vehicle model with a networked PID controller. First, we model the problem of stateful continuous-time components for the dynamic physical vehicle motion on a graded and flat road without using a PID controller. Second, we model the problem of stateful continuous-time components for the dynamic physical vehicle motion on a graded road using a PID controller in different specification scenarios. Third, we model the problem of stateful continuous-time components for the cyber-physical dynamic vehicle model motion using a networked PID controller at different specification scenarios. The simulation results indicated the superiority and efficacy of the cyber-physical dynamic vehicle model providing high stability and controllability in the vehicle motion using a PID controller and Transceiver (network node).

## 1. INTRODUCTION

The recent integration of enhanced control systems with innovative information technologies facilitates the rapid advancement of Cyber-Physical Systems [1]. The obligation of getting an effective, dependable, and malleable communication foundation is accomplished by facilitating real-time data conversation between various smart and conventional CPS/IoT elements. Cyber-Physical Systems (CPS) are orchestrated systems that incorporate computation, networking, and physical processes as well as security services against several cyber-attack vectors [2]. CPS combines the dynamics of the physical processes with those of the software and networking components, providing abstractions and modeling, design, and analysis techniques for the integrated real-time CPS [3, 4]. As such, technological evolution has transformed traditional vehicles into autonomous vehicles that play an essential role in developing smart cities [5].

This paper uses different physical scenarios to analyze the cyber dynamic continuous time vehicle model with a networked PID controller. The contributions in this paper can be summarized as follows:

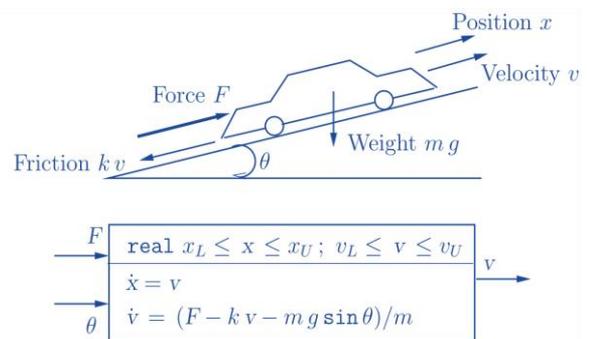
- We model the problem of stateful continuous-time components for the dynamic physical vehicle motion on a graded and flat road using a PID controller in different specification scenarios.
- We model the problem of stateful continuous-time components for the cyber-physical dynamic vehicle model motion using a networked PID controller at different specification scenarios.
- Extensive simulation results are provided to gain insight into the proposed model and the solution approach.

The rest of this paper is organized as follows. Section II describes the problem formulation of vehicle motion. Section III describes the proposed system model and simulation results

for a dynamical vehicle motion system with different design scenarios. Finally, Section IV concludes the paper.

## 2. PROBLEM FORMULATION OF VEHICLE MOTION

The stateful continuous-time component (CTC) model is used to describe how the speed of a vehicle changes because of the force applied to it by the engine [6]. The model is described in Figure 1 below, where the vehicle moves on an uphill single-dimension graded road.



**Figure 1.** Modeling vehicle motion on a Graded Road

The velocity response of this vehicle model can be studied by solving and plotting the ordinary differential equation for the velocity dynamics ( $\dot{v}$ ). This can be solved using several numerical or analytical methods. Here, we will be using the Trapezoidal discrete approximation of derivative with simulation step  $\Delta t=0.10$  sec, as follows:

From the vehicle model:

$$\dot{v} = \frac{dv}{dt} = (F - kv(t) - mg \sin \theta) / m$$

Where,  $v$  stands for the acceleration of the vehicle.

Using Trapezoidal rule:

$$\begin{aligned} \frac{dv}{dt} &\approx \frac{\Delta v}{\Delta t} \Rightarrow \frac{v(k+1) - v(k)}{\Delta t} \\ &= (F - kv(t) - mgsin\theta)/m \\ \therefore v(k+1) &= v(k) + \Delta t(F - kv(t) - mgsin\theta)/m \end{aligned} \quad (1)$$

Therefore, Eq. (1) can be used to describe the velocity response for any force applied to the vehicle. Indeed, the flat

road (i.e., ungraded road) can be modeled by setting the road grad to zero, that is it,  $\theta=0$ , thus:

$$\therefore v(k+1) = v(k) + \Delta t(F - kv(t))/m \quad (2)$$

The continuous-time component model for dynamic vehicle motion on a flat road is illustrated in Figure 2. Note that this model assumes that the rotational inertia of the vehicle wheels is negligible and that the friction resisting the motion is proportional to the vehicle's speed [7, 8].

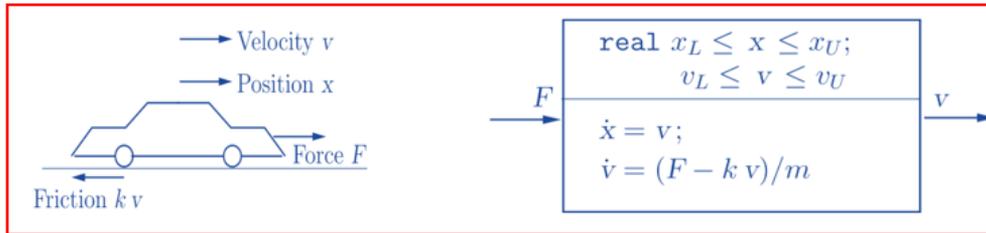


Figure 2. Modeling vehicle motion on a flat road (Ungraded)

### 3. SYSTEM MODELLING AND SIMULATION

In this section, we provide three different methods of modeling the dynamic vehicle model with continuous time components, including the Dynamic vehicle model with no controller, the dynamic vehicle model with the PID Controller, and the Cyber-Physical dynamic vehicle model with the PID Controller.

Table 1. Four case-study scenarios of the dynamic vehicle

Case	Road	Angle ( $\theta$ )	Force (F)	Initial conditions
I	ungraded (flat) road	$0^\circ$	0N	$(x_0, v_0) = (0 \text{ m}, 10 \text{ m/s})$
II	ungraded (flat) road	$0^\circ$	500N	$(x_0, v_0) = (0 \text{ m}, 0 \text{ m/s})$
III	graded road	$5^\circ$	500N	$(x_0, v_0) = (0 \text{ m}, 0 \text{ m/s})$
IV	graded road	$10^\circ$	500N	$(x_0, v_0) = (0 \text{ m}, 0 \text{ m/s})$

#### 3.1 CTC for dynamic vehicle model with no controller

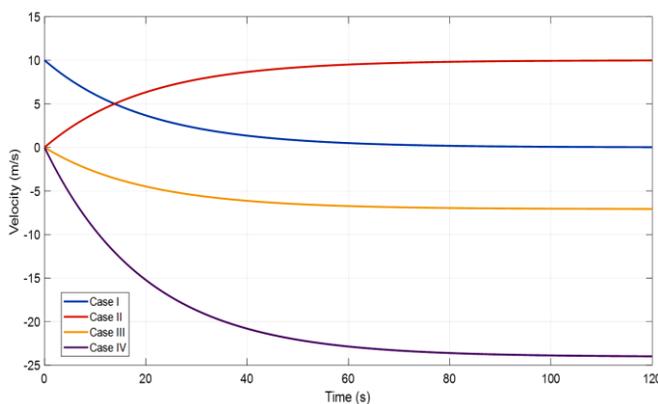


Figure 3. Simulation results for the four case-study scenarios

In this work, we will implement and test the continuous-time component representing a dynamic model of a vehicle using the model above, and dynamics are given that the vehicle's mass ( $m$ ) is 1,000 kg. Gravitational acceleration ( $g$ ) is 9.8 m/s. The friction constant ( $k$ ) is 50 [9]. We consider different modeling cases as given in Table 1. Also, Figure 3 shows the simulation results for the continuous-time

component for the dynamic vehicle model using four different scenarios. The differences between the four cases can be generated by changing the values for  $F$ ,  $\theta$ , and  $v_0$ , as in Table 1. All four scenes tend to be constant in the end as they reach the peak/maximum possible velocity under the given parameters.

#### 3.2 CTC for dynamic vehicle model with PID controller

Indeed, the change of force due to road changes can be modeled using a customized PID controller (i.e., P, PI, or PD controller) [10]. PID controllers (P: proportional, I: integral, D: derivative) are commonly used in industrial control applications to regulate several process variables such as force, temperature, speed, and others] in the control loop feedback mechanism. Figure 4 illustrates the general block diagram of a PID controller [11] in a feedback loop where:  $r(t)$  is the desired process value,  $e(t)$  is the error value calculated continuously by the PID controller as the difference between a desired value ( $r(t)$ ) and a measured process variable ( $y(t)$ ),  $u(t)$  is the control variable determined by a weighted sum of the control terms ( $P$ ,  $I$ ,  $D$ ) and  $y(t)$  is the measured process value.

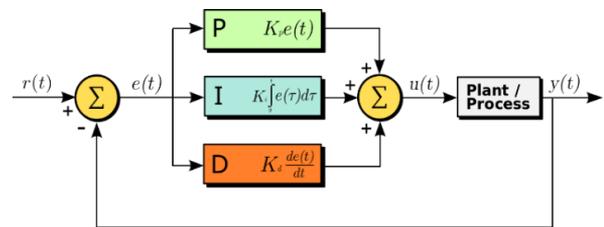


Figure 4. Block diagram of PID controller

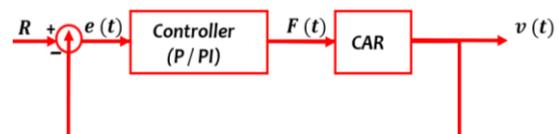


Figure 5. The modified vehicle models

In this work, we consider the implementation and simulation for the modified continuous-time component

representing a dynamic model of a vehicle employing P/PI-Controllers [12, 13] to regulate the required force overtime using the model given in Figure 5 and the modified dynamics of Eq. (2).

To evaluate the provided model, we have used the following general parameters: The mass of the vehicle ( $m$ ) is 1,000 kg, The gravitational acceleration ( $g$ ) is 9.8 m/s<sup>2</sup>, The friction constant ( $k$ ) is 50 (dimensionless coefficient), The sample time ( $\Delta t$ ) is 0.10 sec with total sampling time ( $t_s$ ) of 120 sec, Desired velocity (or called reference point-R) is 10 m/s, Initial conditions ( $x(0), v(0)$ )= $(x_0, v_0)$ =(0m, 0m/s), The velocity error ( $e(t)$ ) is calculated as  $e(t)=R-v(t)$ . Also, the applied force  $F(t)$  to move the vehicle in an uphill direction is no longer assumed to be constant. Instead, we have modeled the variable force as a control variable by applying two different PID controller terms (i.e.,  $P$  and  $PI$ ) as follows:

Using P-Controller: For this case, we have only used the proportional term ( $P$ ) of the PID controller with proportional parameter ( $K_p$ ). Thus,

$F(t) = K_p \cdot e(t) \rightarrow$  substitute this in Eq. (1), then:

$$v(t+1) = v(t) + \frac{\Delta t (K_p \cdot e(t) - kv(t) - mgsin(\theta(t)))}{m} \quad (3)$$

where,

$$\theta(t) = \{Constant (5^\circ \text{ or } 10^\circ), Variable(\frac{1}{3} sin(t/5))\}$$

Using PI-Controller: For this case, we have used two terms of the PID controller: The proportional term ( $P$ ) with proportional parameter ( $K_p$ ) and the integral term ( $I$ ) with integral parameter ( $K_I$ ).

$$F(t) = K_p \cdot e(t) + K_I \cdot \int_{t_i}^{t_{i+1}} e(t) \cdot dt$$

$\rightarrow$  substitute in Eq. (1), then:

$$v(t+1) = v(t) + \frac{\Delta t (K_p \cdot e(t) + K_I \cdot \int_{t_i}^{t_{i+1}} e(t) \cdot dt - kv(t) - mgsin\theta(t))}{m} \quad (4)$$

where,

$$\theta(t) = \{Constant (5^\circ \text{ or } 10^\circ), Variable(\frac{1}{3} sin(t/5))\}$$

For the integration part of model dynamics, we have calculated it numerically using the well-known Trapezoidal rule [14]. Finally, to get more insight, we consider several scenarios for vehicle velocity response, including:

•Vehicle Model with P-Controller:

$$K_p=600, \theta=5^\circ=5 \frac{\pi}{180} rad,$$

•Vehicle Model with P-Controller:

$$K_p=600, \theta=10^\circ=10 \frac{\pi}{180} rad,$$

•Vehicle Model with PI-Controller:

$$K_p=600, K_I=40, \theta=5 \frac{\pi}{180} rad,$$

•Vehicle Model with PI-Controller:

$$K_p=600, K_I=40, \theta=10 \frac{\pi}{180} rad,$$

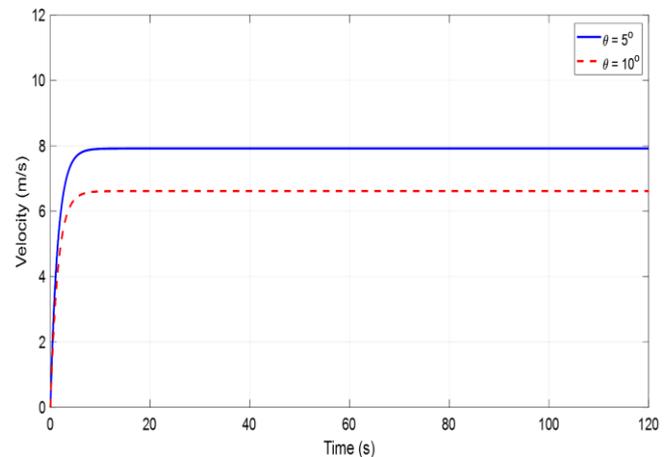
•Vehicle Model with P-Controller:

$$K_p=600, \theta(t) = \frac{1}{3} sin(t/5) rad,$$

•Vehicle Model with PI-Controller:

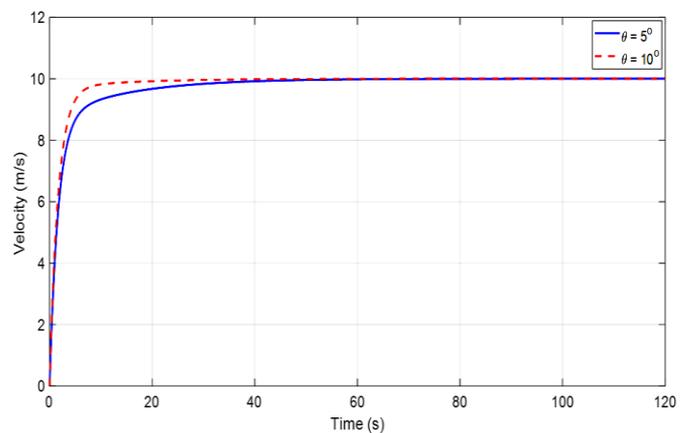
$$K_p=600, K_I=40, \theta(t) = \frac{1}{3} sin(t/5) rad.$$

Besides, we have tried several experiments by tuning the different factors affecting the velocity response, such as the proportional gain  $K_p$ , integral gain  $K_I$ , the  $\theta(t)$ . For instance, Figure 6 shows the velocity response for the dynamic vehicle model using a  $P$ -controller with the following parameters:  $K_p = 600$  and  $\theta = \{5^\circ, 10^\circ\}$ . According to the figure, it can be seen that the  $P$ -controller with the prescribed setup parameters was not able to provide the required force to meet the desired velocity ( $R=10$  m/s) for both cases of road grade ( $5^\circ$  or  $10^\circ$ ). Another gain factor of the PID controller needs to be tuned up to enhance the force controllability, such as the integral gain  $K_I$ .

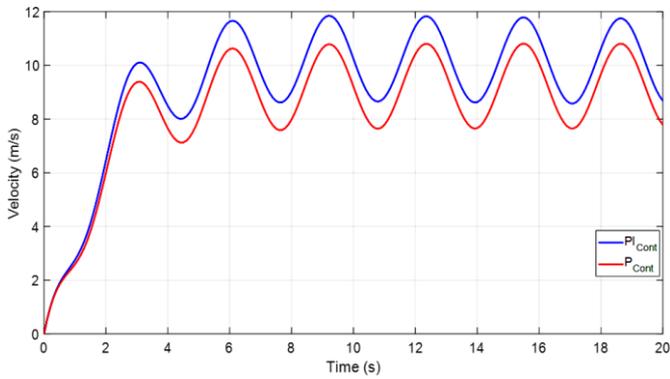


**Figure 6.** The velocity of the vehicle in response to  $R=10$  m/s and  $K_p=600$

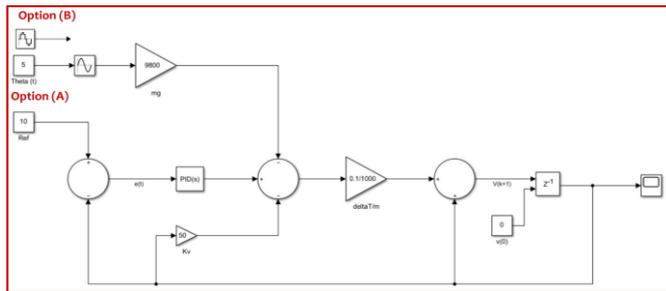
Figure 7 shows the velocity response for the dynamic vehicle model using the PI controller (after adding Integral gain  $I$ ) with the following parameters:  $K_p = 600, K_I = 40$  and  $\theta = \{5^\circ, 10^\circ\}$ . According to the figure, it can be seen that the PI-controller with the prescribed setup parameters has successfully provided the required force to meet the desired velocity ( $R=10$  m/s) for both cases of road grade ( $5^\circ$  or  $10^\circ$ ) with a slightly faster response appeared for the road disturbance input signal of  $10^\circ$ .



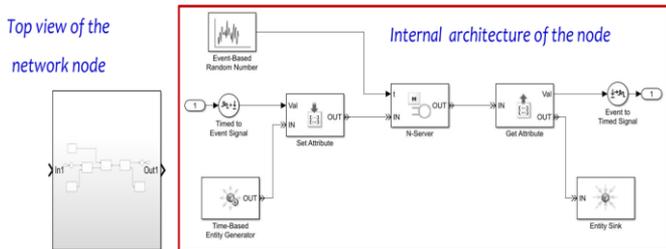
**Figure 7.** The velocity of the vehicle in response to  $R = 10$  m/s,  $K_p = 600, K_I = 40$



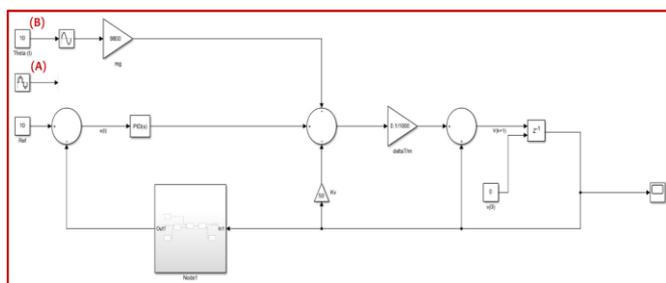
**Figure 8.** Velocity response:  $R = 10 \text{ m/s}$ ,  $K_p = 600$ ,  $K_I = 40$ ,  $\theta(t) = \frac{1}{3} \sin(\frac{t}{5}) \text{ rad}$



**Figure 9.** Simulink Implementation of Vehicle Model with P/PI controllers using two options: (A)  $\theta(t) = 5^\circ$ ,  $\theta(t) = 10^\circ$ , (B)  $\theta(t) = \frac{1}{3} \sin(\frac{t}{5}) \text{ rad}$



**Figure 10.** *SimEvents* model of the network node (transceiver node)



**Figure 11.** Simulink vehicle model with one networked PI-controller

Figure 8 shows the response of the vehicle velocity for the dynamic vehicle model using a continuous-time disturbance input signal following a sinusoidal signal  $\theta(t) = \frac{1}{3} \sin(t/5) \text{ rad}$ . Such a formula models the change in road condition as the vehicle driving on it. Besides, the figure considers the cruise controller using only a P-controller or PI-controller with the following parameters:  $K_p = 600$ ,  $K_I = 40$ . According to the figure, it can be seen that the velocity

response of such a vehicle motion dynamic follows a sinusoidal behavior. The vehicle motion velocity keeps fluctuating as a sinusoidal up and down between 7.5 m/s and 10.5 m/s for the cruise controller implementing the proportional gain ( $P$ ) only. In contrast, it ranges from 8.5 m/s to 11.5 m/s for the cruise controller implementing the proportional and integral gains ( $PI$ ). The results showed that the PI controller recorded more accurate results since the average range of its response approaches 10 m/s while the average response for the P controller approaches 9 m/s.

Moreover, according to the results mentioned above and discussions, we have observed a significant difference between the performance of  $P$  and  $PI$  controllers on the vehicle speed with superiority registered for the  $PI$  controller. Indeed, it would be more interested in acquiring the PI controller as it rapidly converges to the desired velocity [15]. Finally, recall the stateful continuous-time component model of how the speed of a vehicle changes because of the force applied to it by the engine [7] when the vehicle is moving on an uphill single-dimension graded road. In this model, we are controlling the force  $F(t)$  using the PID controller [11], mainly  $P$  &  $PI$  controllers. Finally, we are implementing the complete model using the *Simulink* toolbox [12], as given in Figure 9.

### 3.3 Cyber-physical dynamic vehicle model with PID controller

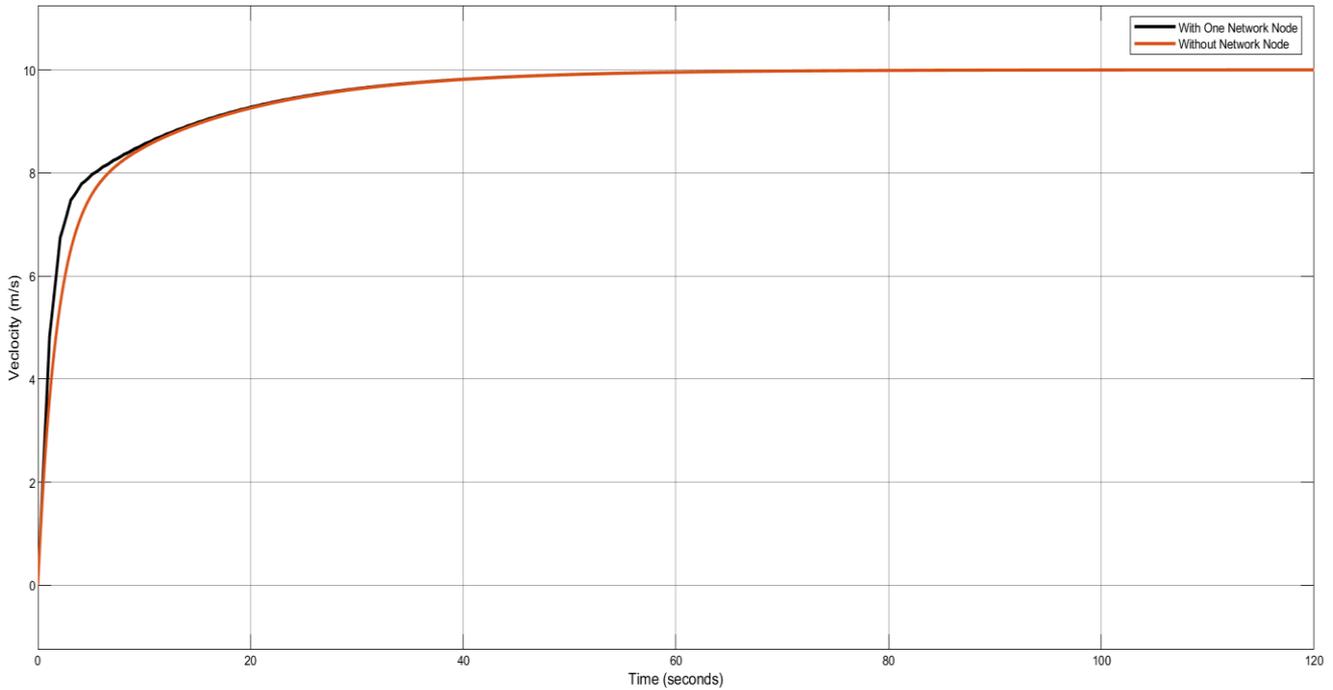
In this case, we are adding a cyber component that can control the physical system remotely through a communication network. Therefore, we have utilized the *SimEvents* toolbox of *Matlab* to adopt the cyber part of the system by developing a network node model (transmit/receive) with the ability to simulate different random network time-delay. Recall the *Simulink* component models of the vehicle and cruise P-controller with vehicle dynamics as one block and the P-controller as another block and using the same parameters:  $K_p = 600$ ,  $K_I = 40$ , constant disturbance  $\theta(t)=10^\circ$ , variable disturbance  $\theta(t) = \frac{1}{3} \sin(\frac{t}{5}) \text{ rad}$ , and finally, the desired vehicle speed  $R=10 \text{ m/s}$ . The developed networking node is illustrated in Figure 10. The component considered the network uniform distribution in the timed-based random number distribution block with a minimum value of 0.01 and a maximum value of 0.1. The uniform distribution is meant to have evenly spaced time units starting at 0.01 to 0.1. the minimum and maximum values can be arbitrarily determined. The selected values (0.01 to 0.1) are more realistic for network nodes [0.01s, 0.1s].

To make use of the network as mentioned above the node, we have added one instance of the network node model to the vehicle-cruise controller using the *Simulink* model developed in Figure 9. The node has been inserted between the sensor and the controller only to end up with a vehicle model using a *networked PI controller*. The developed model of the vehicle with one networking node is illustrated in Figure 11. Note that  $\theta(t)$  can be used as a constant  $10^\circ$  (option A) or can be used as a variable disturbance using sinusoidal  $\theta(t) = \frac{1}{3} \sin(\frac{t}{5}) \text{ rad}$  (option B).

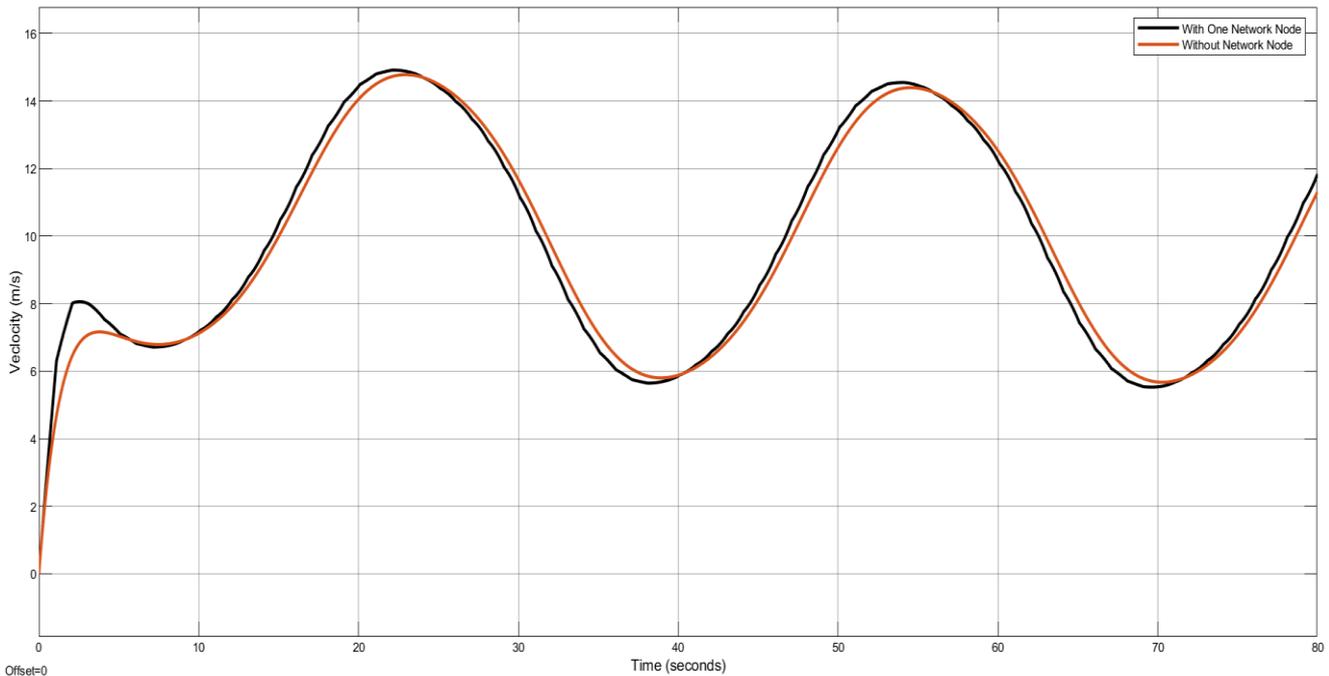
Accordingly, we have tested this model using different min-max values of the random distribution block and by using either constant disturbance  $\theta(t) = 10^\circ$  ( $10 \times (\pi/180) \text{ rad}$ ) or variable disturbance  $\theta(t) = \frac{1}{3} \sin(\frac{t}{5}) \text{ rad}$ . For instance, Figure 12 compares the velocity responses for the vehicle model with *PI-controller* on a graded road with fixed  $\theta(t)=10^\circ$  before and after adding a network node between the sensor and

the controller. The network node employs a random number distribution block in this model to generate time delay values  $\in [0.01s, 0.1s]$  using a uniform random distribution. According to the figure, the vehicle response has been slightly

delayed due to using a network node with the delay mentioned above boundaries.



**Figure 12.** Velocity response of vehicle model with and without incorporating one network node ( $\theta=10^0$ )



**Figure 13.** Velocity response of vehicle model with and without adding one network node ( $\theta(t) = \frac{1}{3} \sin(\frac{t}{5}) \text{ rad}$ )

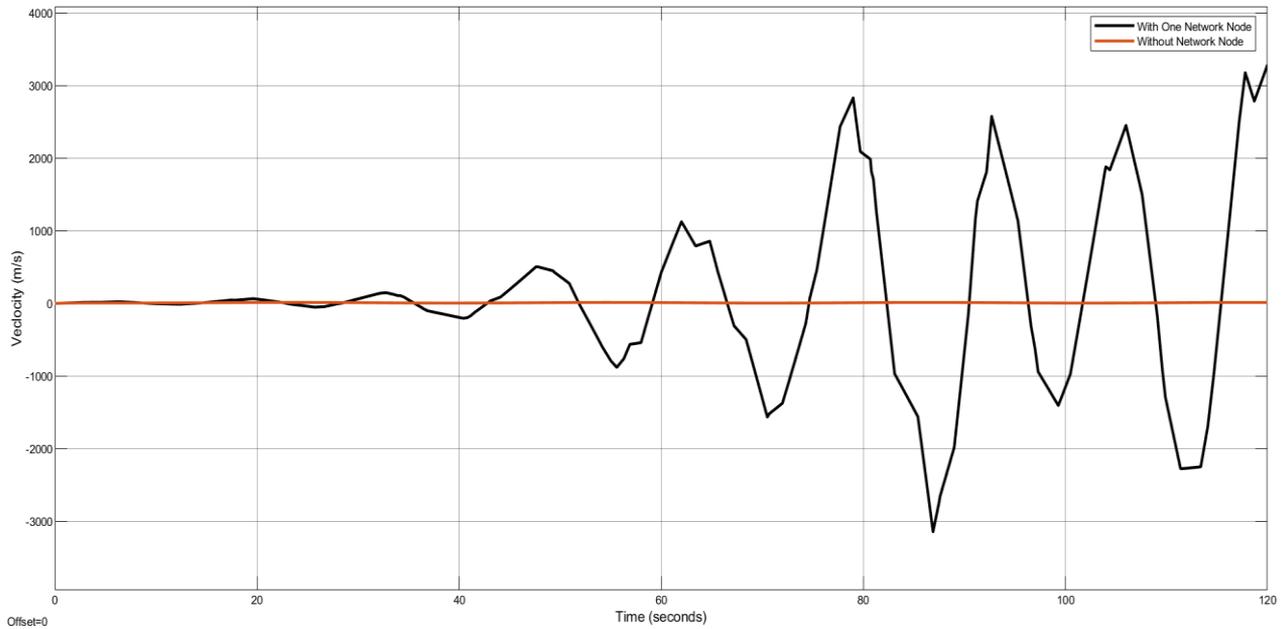
Figure 13 illustrates the velocity responses for the vehicle model with *PI controller* on a graded road with variable sinusoidal disturbance  $\theta(t) = \frac{1}{3} \sin(\frac{t}{5}) \text{ rad}$ . The model considers two configurations: with and without adding a network node between the sensor and the controller. The main reason for adding the network node is to turn the system into a Cyber-Physical System. Without the network nodes (cyber

part), the system is assumed to be a physical system node with no communication ability. Again, the network node uses a *uniform* random number distribution for time delay values  $\in [0.01s, 0.1s]$ , and thus, the vehicle response has been slightly delayed due to the use of network node.

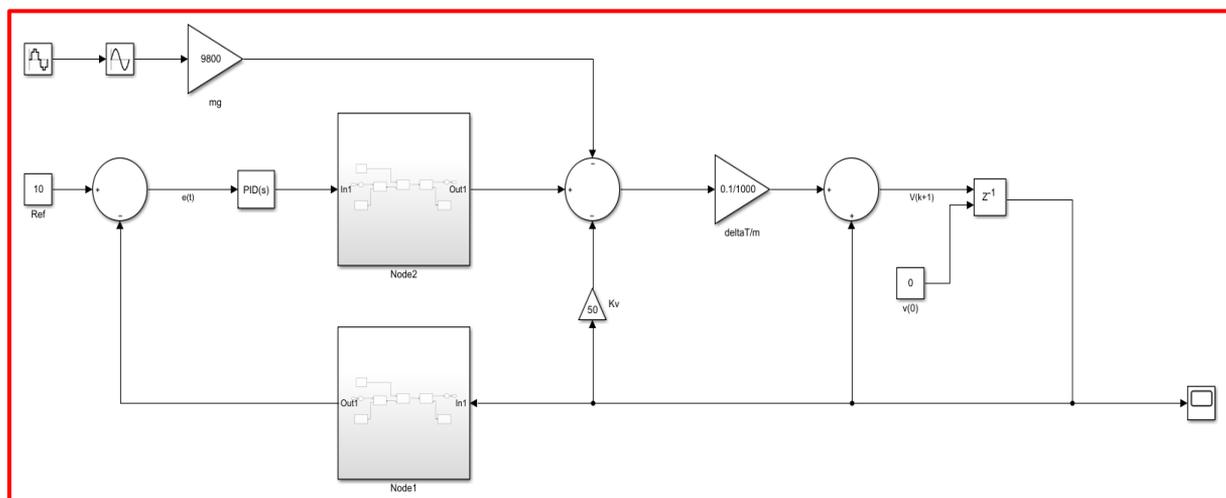
Moreover, to analyze the effect of network delay on the system, we have tested the vehicle velocity response with five different *min-max* values of the random distribution block.

This includes: [0.01s, 0.1s], [0.05s, 0.1s], [0.1s, 1.0s], [0.5s, 2.0s] and [1.0s, 5.0s]. Almost the system was stable in all tests except for the last testing interval [1.0s, 5.0s] (largest delay range). The system showed unstable behavior under the same

parameters and simulation time, as illustrated in Figure 14. Note that the system showed an unstable response in this random interval regardless of the disturbance of the road.



**Figure 14.** Velocity response of vehicle model with one network node using random delay time  $\in [1.0s, 5.0s]$



**Figure 15.** Simulink vehicle model with two networked PI-controller

To enhance the system stability in response to the cyber control, we have added two instances of the network node model to the vehicle-cruise controller using the *Simulink* model developed in Figure 9. One has been inserted between the sensor and the controller and another node between the controller and the vehicle model to end up with a cyber-physical vehicle model using *two networked PI-controller*. The developed model of the vehicle model with two networking nodes is illustrated in Figure 15. the figure shows the variable disturbance block  $\theta(t)$ . Again,  $\theta(t)$  can be used as a constant  $10^0$  block.

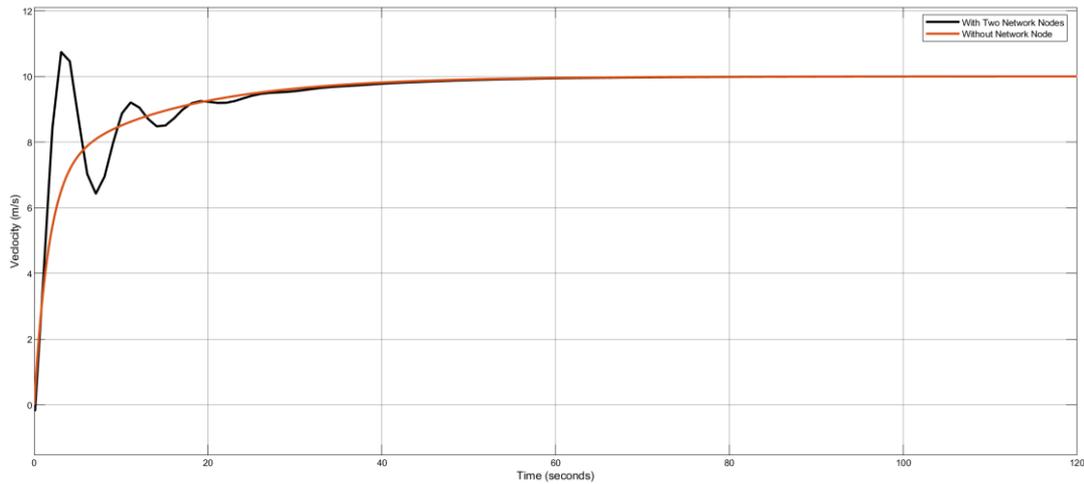
Similarly, we tested the velocity response for this model using different uniformly distributed random min-max values of the network nodes using  $\theta(t) = \{either 10^0 or (1/3) \sin(t/5) rad\}$ . Figure 16 compares the velocity responses for the vehicle model with *PI-controller* on a graded road with

fixed  $\theta(t) = 10^0$  before and after adding two network nodes: one node was added between the sensor and the controller and another node between the controller and the vehicle model. In this test, we consider both network nodes to have a uniformly distributed time delay value  $\in [0.01s, 0.1s]$ . According to the figure, adding more network nodes resulted in an additional delay in the response signal, which spikes for the first couple of times.

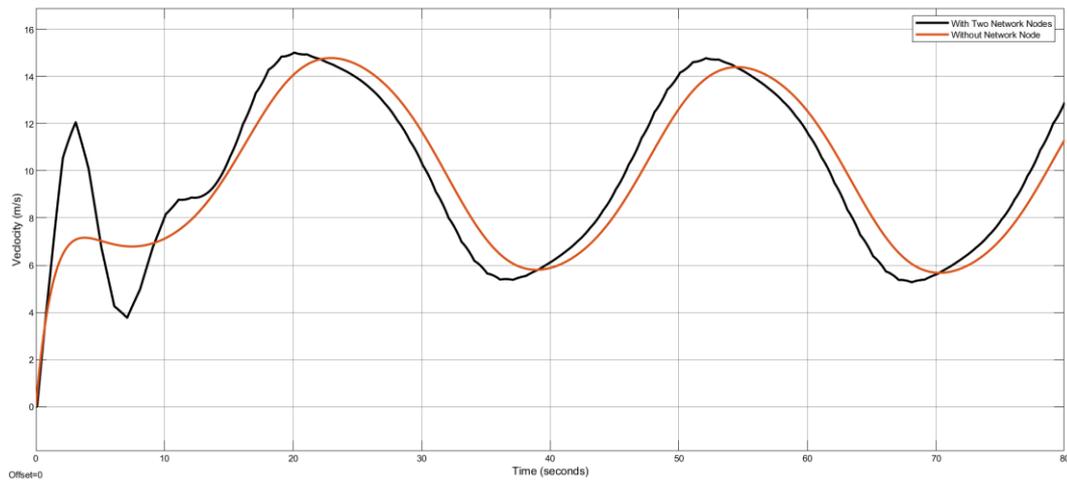
Figure 17 presents the velocity responses for the vehicle model with *PI-controller* of variable graded road  $\theta(t) = (1/3) \sin(t/5)$  rad considering two configurations: with and without adding two network nodes to the vehicle model. Besides, both network nodes employ a *uniformly* distributed time delay  $\in [0.01s, 0.1s]$ . As a result, the plot shows that adding more network nodes resulted in delaying the vehicle response following the delay values, with a signal spike

appearing at the beginning of the response. Comparing the experimental results shown in Figure 13 and Figure 17, you can see that the cyber physical system incorporating one network node has a lower delay than one incorporating two

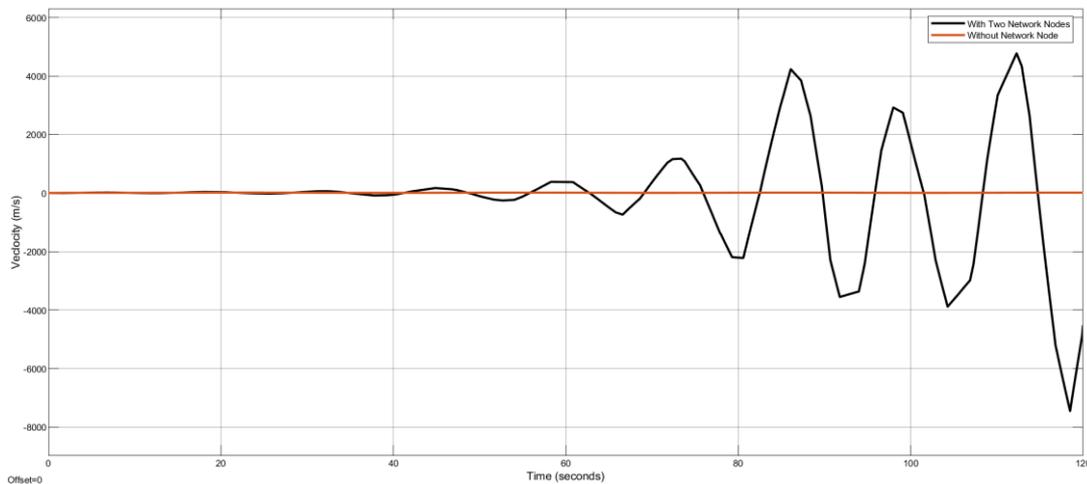
nodes. Please focus on how the black curve in the case of one node (Figure 13) is more tightly coupled to the red curve (no network at all). At the same time, it has a larger delay in the case of two nodes (Figure 17).



**Figure 16.** Velocity response of vehicle model with and without incorporating two network nodes ( $\theta=10^\circ$ )



**Figure 17.** Velocity response of vehicle model with and without adding two network nodes ( $\theta(t) = \frac{1}{3} \sin(\frac{t}{5}) \text{ rad}$ )



**Figure 18.** Velocity response of vehicle model with two network nodes using random delay time  $\in [0.5s, 2.0s]$

Finally, we have also performed five tests for the vehicle velocity in response to varying the *min-max* values of the

random distribution block, including the intervals: [0.01s, 0.1s], [0.05s, 0.1s], [0.1s, 1.0s], [0.5s, 2.0s] and [1.0s, 5.0s].

As a result, the system showed stable behavior in the first three interval tests. However, under the same parameters and simulation time, the system becomes unstable at the distribution range of [0.5s, 2.0s] and [1.0s, 5.0s]. Figure 18 shows the velocity response for the vehicle model employing the distribution range of [0.5s, 2.0s] regardless of the disturbance of the road.

#### 4. CONCLUSION

Cyber-Physical Systems (CPSs), which integrate computation, networking, and physical processes, play a progressively more valuable role in essential transportation and everyday life. Due to physical restrictions, embedded computers and networks may offer spread to several further control challenges that may result in shortfalls of immense cost-cutting advantages or confusion of public living. As a result, it is considered to appropriately investigate cyber-physical vehicle systems' controllability and stability issues to ensure such systems operate safely. This paper analyzes the system dynamics of a continuous-time cyber-physical vehicle system using different physical scenarios. Especially, the study considers stateful continuous-time components for the dynamic physical vehicle motion on a graded road and a flat road (1) without using a PID controller, (2) using a PID controller in different specification scenarios, and (3) cyber-physical dynamic vehicle model motion using a networked PID controller at different specification scenarios. Extensive simulation results were provided to gain insight into the proposed model and the solution approach. The system evaluation showed the advantage of the cyber-physical dynamic vehicle system delivering high stability and controllability in the vehicle motion using a PID controller and Transceiver. In future, we will seek to adopt intelligent controllers making use of optimizable neural networks systems [16], fuzzy Nero computing, and machine/deep learning models [17, 18] in addition to the ability to acquire the data form Heterogeneous Sources.

#### REFERENCES

- [1] Smadi, A.A., Ajao, B.T., Johnson, B.K., Lei, H., Chakhchoukh, Y., Abu Al-Haija, Q. (2021). A comprehensive survey on cyber-physical smart grid testbed architectures: Requirements and Challenges. *Electronics*, 10(9): 1043. <https://doi.org/10.3390/electronics10091043>
- [2] Al-Haija, Q.A. (2011). On the security of cyber-physical systems against stochastic cyber-attacks models. *IEEE International IOT, Electronics and Mechatronics Conference (IEMTROICS)*, Toronto, Canada, pp. 1-6. <https://doi.org/10.1109/IEMTRONICS52119.2021.9422623>
- [3] Al-Fareed, H., Alghamdi, O., Alshuraya, A., Alqahtani, M., Alwasfer, S., Aljomea, A., Rahman, A., Aljameel, S., Krishnasamy, G. (2022). Simulator for scheduling real-time systems with reduced power consumption. *Mathematical Modelling of Engineering Problems*, 9(5): 1225-1232. <https://doi.org/10.18280/mmep.090509>
- [4] Al-Haija, Q.A., McCurry, C.D., Zein-Sabatto, S. (2020). A real-time node connectivity algorithm for synchronous cyber-physical and IoT network systems. In *2020 Southeast Con*, pp. 1-8. <https://doi.org/10.1109/SoutheastCon44009.2020.9249730>
- [5] Alsulami, A.A., Abu Al-Haija, Q., Alqahtani, A., Alsini, R. (2022). Symmetrical simulation scheme for anomaly detection in autonomous vehicles based on LSTM Model. *Symmetry*, 14(7): 1450. <https://doi.org/10.3390/sym14071450>
- [6] Alur, R. (2015). *Principles of Cyber-Physical Systems*. Massachusetts Institute of Technology (MIT) Press.
- [7] Hu, X., Wang, H., Tang, X. (2017). Cyber-physical control for energy-saving vehicles following with connectivity. *IEEE Transactions on Industrial Electronics*, 64(11): 8578-8587. <https://doi.org/10.1109/TIE.2017.2703673>
- [8] Liu, H., Sun, D., Zhao, M. (2016). Analysis of traffic flow based on car-following theory: A cyber-physical perspective. *Nonlinear Dynamics*, 84(2): 881-893. <https://doi.org/10.1007/s11071-015-2534-y>
- [9] Tehrani, K.A., Mpanda, A. (2012). *Introduction to PID Controllers - Theory, Tuning, and Application to Frontier Areas*. InTech-Open Press.
- [10] Thakur, P., Sehgal, V.K. (2021). Emerging architecture for heterogeneous smart Cyber-Physical Systems for industry 5.0. *Computers & Industrial Engineering*, 162: 107750. <https://doi.org/10.1016/j.cie.2021.107750>
- [11] Kim, Y., Chen, A., Jadhav, S., Gloster, C.S., Le, T., Hsu, P. (2016). Project-based courses in control Cyber-Physical System co-design. In *Proceedings of the 2016 Workshop on Embedded and Cyber-Physical Systems Education*, pp. 1-4. <https://doi.org/10.1145/3005329.3005333>
- [12] Škraba, A., Stanovov, V., Semenkina, E. (2020). Development of a control systems kit for studying PID controllers in the framework of Cyber-Physical Systems. In *IOP Conference Series: Materials Science and Engineering*, 734(1): 012105. <https://doi.org/10.1088/1757-899X/734/1/012105>
- [13] Simulink. [https://www.tutorialspoint.com/matlab/matlab\\_simulink.htm](https://www.tutorialspoint.com/matlab/matlab_simulink.htm), accessed on June 1st, 2022.
- [14] Chapra, S., Canale, R. (2010). *Numerical Methods for Engineers*, Sixth Edition. McGraw-Hill, Higher Education.
- [15] Abood, L.H., Haitham, R. (2022). Design an optimal fractional order PI controller for congestion avoidance in internet routers. *Mathematical Modelling of Engineering Problems*, 9(5): 1321-1326. <https://doi.org/10.18280/mmep.090521>
- [16] Abu Al-Haija, Q., Krichen, M. (2022). A lightweight in-vehicle alcohol detection using smart sensing and supervised learning. *Computers*, 11: 121. <https://doi.org/10.3390/computers11080121>
- [17] Abu Al-Haija, Q., Alsulami, A.A. (2022). Detection of fake replay attack signals on remote keyless controlled vehicles using pre-trained deep neural network. *Electronics*, 11: 3376. <https://doi.org/10.3390/electronics11203376>
- [18] Al-Haija, Q.A., Gharaibeh, M., Odeh, A. (2022). Detection in adverse weather conditions for autonomous vehicles via deep learning. *AI*, 3: 303-317. <https://doi.org/10.3390/ai3020019>