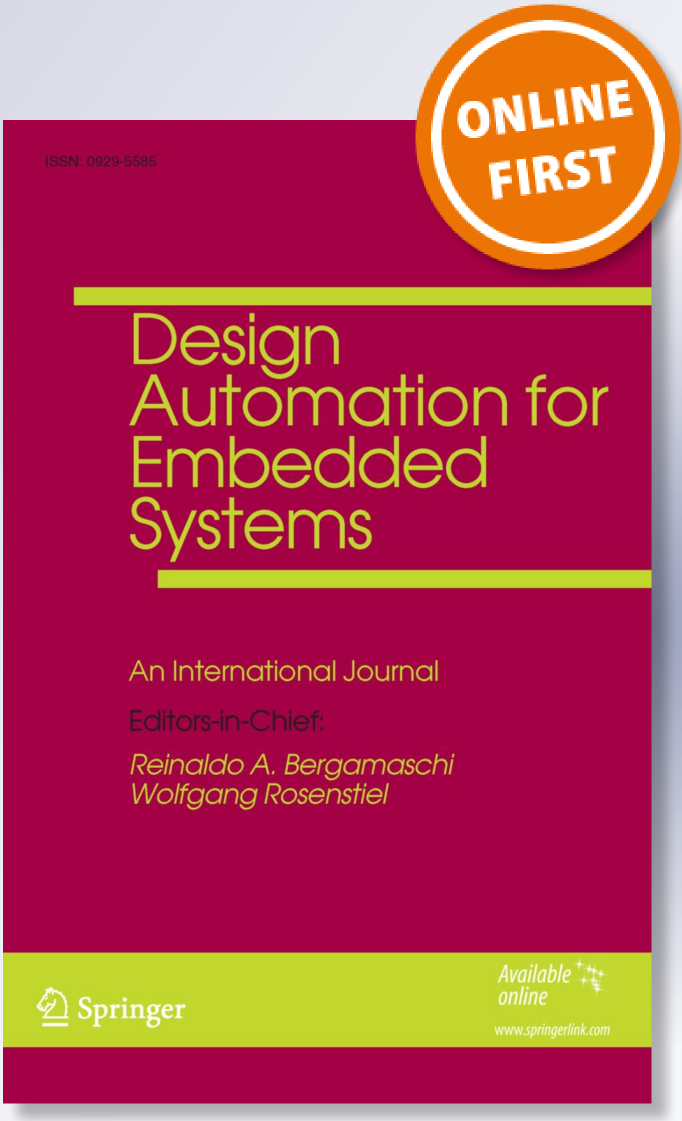
A novel algorithm for real-time framework in multiprocessor environment

Joel Josephson Pottipadu & R. Ramesh

Design Automation for Embedded Systems
An International Journal

ISSN 0929-5585

Des Autom Embed Syst
DOI 10.1007/s10617-017-9195-7



Des Autom Embed Syst
DOI 10.1007/s10617-017-9195-7

CrossMark

A novel algorithm for real-time framework in multiprocessor environment

Joel Josephson Pottipadu[1] · **R. Ramesh**[1]

Received: 16 June 2017 / Accepted: 23 October 2017

Abstract The objective of the paper is to represent the real-time framework in multiprocessor environment task scheduling process by examining the novel algorithm advanced PSO. The advanced PSO algorithm has the metaphor as the basis to facilitate social interaction, which makes a search on space by making adjustments to the trajectories of individual vectors, referred to "particles" as they are considered as the points that move within the multidimensional search space. This algorithm will reduce the turn-around time, burst time and waiting time for multiprocessor task scheduling, when compared to existing algorithm like first come first served algorithm, shortest job first algorithm and round robin scheduling algorithm. This proposed algorithm is developed and making hardware for tank level water control for single input single output tank system and two input two output tank system. LABVIEW software is used to implement the real-time algorithm in hardware and software. The time taken for searching best position for particles will be executed and compare with all algorithms using bar chart will be given in result.

Keywords PSO · Cuckoo search · LABVIEW · Fuzzy logic · Multiprocessor environment

1 Introduction

Scheduling problems in the environment of multiprocessor belong to the combinatorial class of optimization problems, which referred as NP-hard problem. In general, multiprocessor scheduling solves both allocation and scheduling problem with different approaches like PSO, Neural Networks (NN) or Heuristic Algorithms (HA) [1]. In case of existing real-time systems towards m processors, that classified into the global and partitioned scheduling, and each task is allocated to a particular processor statically, and execution takes place on that processor only in partitioned development. Tasks can migrate on m processors dynamically and are

✉ Joel Josephson Pottipadu
joeljosephsonp@gmail.com

1 Department of Electrical and Electronics Engineering, College of Engineering Guindy, Anna University, Chennai, India

Published online: 07 November 2017

allowed to execute on any of the processors in global scheduling [2]. Architects have adopted the multiprocessor architecture for real-time systems because of its superiority with regards to high scalability and lower levels power consumption. The complex issue of utilizing several computational resources arises in the real-time scheduling of applications in multiprocessor as compared to single processor systems. Several algorithms for real-time scheduling have been put forward for multiprocessor architectures, and despite that there is no single dominant answer to the problem. In turn, this is highlighted the necessity to accommodate efficiently these algorithms in the real-time systems. This trend led to some operating systems to have extensibility and functionality,i.e., the structure of modular scheduler where in schedules are treated as extension modules and to host available new scheduling policies [3].

Each task uses the resources shared in an exclusive mode. At any specific time, one task holding a resource cannot give the resource to the requested task immediately but can give the resource only after completion of its execution. Such resources are referred to as critical section resources or Non-Preempted Resources (NPR). The exclusive mode can also involve an NPR, which enables, a high priority task would wait for the low priority task holding an NPR to complete its execution, and such situations are referred to as priority inversion. Deadlock is referred to as situation when high priority task waits for a long time for the resource. Thus, which lead to the development of scheduling algorithm [e.g. Earliest Deadline First (EDF) and Rate Monitoring Algorithm (RMA)] used in sharing in Central Processing Unit (CPU), a resource of serially reusable sets. However, these algorithms are not used for the NPR shared a set of tasks in real time due to two main problems such as priority inversion and deadlock arises while scheduling tasks that share critical resources. To avoid these problems, different protocols have been developed [4–6] using different types of Resource Access Protocol (RAP):TheSingle-processor system follows NPP (Non-Preemptive Protocol), PIP (Priority Inheritance Protocol), HLP (Highest Locker Priority), SRP (Stack Resource Policy), and PCP (Priority Ceiling Protocol) protocols.Themultiprocessor system follows policies and protocols such asMSRP (Multiprocessor Stack Resource Policy), and MPCP (Multiprocessor Priority Ceiling Protocol) [7].

2 Literature review

In the past there are several algorithms have been proposed to solve issues related to the multiprocess environment. Perez-Gonzalez et al. [8] proposed a simple method based on PSO propose a green house model in order to identify several parameters in a proposed mathematical model. The proposed method implementation is carried out in an offline optimization schedule, which uses real data recorded through the LabView. The study findings revealed that the sample-to-sample version of the PSO algorithm increases the good behavior of the computational method, which allows the optimization to be faster and better in the online form than the offline. However, the proposed greenhouse model prototype fails to use optimized model to design control actions to operate the internal greenhouse temperature and relative humidity.

Malarvizhi and Kiruba [9] proposed a control and modeling of theNon-linear system using anintelligent technique using PSO. The process control employs a non-linear system with a Two Tank Interacting Conical Tank System (TTICS) by using the mathematical modeling based onreal-time process data. The study findings revealed that the proposed PID PSO for TTICS has more robust stability and efficiency, and this solves the searching and tuning problems of PID controller parameters more easily and quickly than compared to Ziegler-Nichols method.

Zhe et al. [10] and Schoenwetter et al. [11] determined a framework for rapid experiment implementation in the field of automatic process control. The proposed solution has real-time control experiment that can be easily and rapidly, appreciating and validating advanced control algorithms. The rapid algorithm proposed based on a combination of RTW, OPC, and XPC and implemented in the quadruple-tank process control test rig. The study findings revealed that control prototyping approaches could easily realize real-time modern control algorithm on a real plant.

He et al. [12] aimed to develop advanced control algorithms using a reliable OPC (OLE for Process Control) technology to connect the field process and control equipment and to recognize the data reading in real time. The algorithm developed using PCS-C type process control device, and by using Fuzzy PID control algorithm to achieve double-tank water level control. The study findings revealed that comparative analysis of the effect of PID and fuzzy PID control, Fuzzy-PID controller on-line regulate the three parameters of PID controller in real-time, and can achieve good control effect for the nonlinear object. It found that using the Fuzzy-PID algorithm, the system can quickly into the stable state, and with smaller overshoot and steady high precision,the output curves are rough due to the interference.

Bi et al. [13] introduced a technique of "Beyond the interconnections: Split manufacturing in rf designs" electronics. Split manufacturing gives an innovative resolution in the contradiction of converse engineering and IP piracy along with the globalization of designs of IC flow. Compared to all existing methods, in this method we are explaining a method in order to implement the RF designs. The removal of the top metal layer, the removal of the top two metal layers and the design obfuscation dedicated to RF circuits. These three different types of implementations are presented for security and design. And also in order to measure the protection level of RF designs under split manufacturing, we industrialized a quantitative security evaluation method. For additional controller the application of split manufacturing in RF circuits, three different FEOL and BEOL separation and obfuscation methods were introduced. These systems were established on two RF circuits, (1) a simple Class AB power amplifier and (2) a more sophisticated Class E power amplifier. The investigational consequences established that the unknown passive components, either inductors or capacitors, along with the missing DC biasing conditions, can increase an important amount of uncertainty for the attacker to recover the RF circuits

Yuan and Bi [14] proposed a new technique of "Process and temperature robust voltage multiplier design for rf energy harvesting" MR. In this paper, the process, temperature and development changeability of the voltage multiplier for RF energy gathering has been studied. Here the belongings of threshold voltage drift because of p-channel transistor negative bias temperature unpredictability and temperature difference are observed. Furthermore, a method of threshold voltage compensation is examined in order to develop the output voltage sensitivity in contradiction of development dissimilarities and temperature variation. This threshold voltage compensation method effectually decreases the temperature and process unpredictability on the voltage multiplier presentation. The voltage multiplier using our modified threshold voltage compensation technique, offers robust circuit presentation in contrast to process differences and temperature drift for long term reliability.

Yuan et al. [15] introduced a new technique of "Hot Carrier Injection Stress Effect on a 65 nm LNA at 70 GHz" TDMR. In this paper, the concept of hot carrier injection stress effect on a 65 nm low-noise amplifier at the 70 GHz range of operation has been studied. The low-noise amplifier using 65 nm CMOS expertise has been invented in order to evaluate the effect of voltage stress on the LNA in the millimeter-wave frequency. Examined equations for the sensitivity of threshold voltage shift and mobility degradation on the impact of minimum noise figure presentation are resulting for physical insight. The amplifier of low-noise is stressed under high supply voltage VDD. The investigational information in the frequency of millimeter-wave command illustrate that, the noise figure rises ($\sim$ 2 dB) and small-signal power gain S21 reduces ($\sim$ 3 dB) after substantial HCI stress because of transconductance deprivation as showed by 65 nm separate transistor overstating quantity.

Qutaiba Alasad et al. [16] introduced a new concept of "E2LEMI: Energy-efficient Logic Encryption Using Multiplexer Insertion". In this paper, the insertion of multiplexer on the basis of logic encryption has been studied. Related to existing literature, the analysis of fault impact method will not assurance attaining a 50% Pretense distance for any circuit, and the performance time of its process also very elongated and undesirable in repetition for a large chip. Because of outsourcing of manufacturing chip, countermeasures in contradiction of Integrated Circuit (IC) piracy, reverse engineering, IC overbuilding and hardware Trojans (HTs) developed a scorching investigation topic. So, in order to protect an IC from these occurrences, the technique of logic encryption is measured. This paper proposes to insert the multiplexer (MUX) in two cases: (1) Randomly insert MUXs equal to half of the output bit number and (2) Insert MUXs equal to the number of output bits. Pretense distance is accepted as a security assessment. Furthermore, in place of consuming both RSA cryptography and PUF, we hired the HLU and the LFSR random generator in order to protect the secret key and produce random keys with 0.5 probabilities, individually. Gradually the power, area and delay overheads are decreased for a large circuit, it has suitable production bits, like C6288 and S9234. Finally, this proposed method can outperform the preceding state-of-the-art work in terms of less presentation above whereas attaining a higher security level.

Geetha et al. [17] proposed two-dimension PID fuzzy controller, fuzzy PI+ fuzzy ID to overcome the parameter uncertainties; it improves the fast tracking performance of a processing system. The study findings revealed that the proposed control scheme not only improves the fast-tracking performance, also it increases the robustness of the system. The simulation results revealed that there is considerable improvement in the Self-Tuning Fuzzy PID controller regarding peak overshoot, settling time, peak time, rise time, Integral Square Error (ISE) and Integral Absolute Error (IAE). Hence, the fuzzy PID can be applied to object control processes like Medical Robotics, Industrial Testing, and measurements.

Although studies conducted used PSO algorithm or hybrid of fuzzy but not many studies on the tank and liquid control. Turner [18] analyzed threat alert that originates from the real time application of Deployment Optimization Algorithms. However, this algorithm is easy to coordinate in a DOVE system fails to implement in the tank system. Pandey et al. [19] made a proposal of PSO with the heuristic basis for scheduling the applications related towards cloud resources which consider data transmission cost and computation cost. Then, BRS or Best Resource Selection algorithm was compared. Selvakrishnan and Perumal [20] made a proposal about the scheduler to allocate the application of the host from the number of hosts that are available and the applications by choosing an ideal match. The algorithm that is PSO based which is very effective in scheduling the grid resources to lower the time of execution and time of completion. Sarathchandar et al. [21] make the representation about the particle swarm optimization technique using digital pheromones that offernetter solution traits. The major aim of proposed algorithm is to identify the solution which offers an optimal schedule to lessen the time of flow within a grid environment. Hardoroudi and Chuprat [22] made a proposal to provide a solution to overcome the issues in the task set, and the process time can be raised with PSO algorithm.

In this study, we proposed a novel algorithm based on the features of PSO, cuckoo and fuzzy features, which we named as advanced PSO algorithm. Further, the advanced PSO has been implemented in both software and hardware co-design for the water tank control system.

3 Algorithm

3.1 PSO

Adaptation of particle swarm proved to be successful for optimizing an array of functions that are continuous [23,24]. The algorithm that has the metaphor as the basis to facilitate social interaction, which makes a search on space by making adjustments to the trajectories of individual vectors, referred to "particles" which are considered a point, which moves within the multidimensional space [25–27]. PSO is said to be the evolutionary technique, as it is an algorithm based on population, which forms particles set that represent potential solutions to the issue that is given. Every particle that passes through the space of dimensional search, is associated with the vector's position x, (t) = {Xil(t), Xi2(t),. . ., Xin(t)} and velocity vector v;(t) = {Vil(t),Vi2(t),. . .,Vin(t)} for the evolutionary iteration at present, t. The single particle within the PSO moves in the search space with a velocity which gets adjusted dynamically based on the flying experience it gains and the flying experience of its companions [28]. It is called as the cognition the only model whereas the other called the social only model [29]. With the integration of dual knowledge types, the behavior of a particle inside the PSO is modeled with the help of the equation as shown below [30]:

$$v_i(t+t) = w \times v_i(t) + c_1 \times r_{\text{and}} \times (pbest_i - x_i(t)) + c_2 \times r_{\text{and}} \times (Gbest - x_i(t))$$
$$x_i(t+1) = x_i(t) + v_i(t+1)$$

where c., c2, are known as the acceleration constants; r_{and} stands for the random numbers from 0 to 1; X_{i} (t) denotes the particle's position I at t iteration; V_{i} (t): the particle's velocity I at t iteration; W denotes inertia of weight factor; Gbest stands for the best position of the particles; Pbesti denotes the particle's personal best position i.

3.2 Cuckoo search (CS) algorithm

CS algorithm has the cuckoo breed's parasitic behavior as its basis that teams up with some fruit flies' and birds' follow Levy flight behavior. Few breeds of Cuckoo birds often lay the eggs inside the communal nests. When the host bird finds out that the eggs do not belong to it, the bird either tries to throw away the eggs or leave sits nest immediately to build a new one somewhere. CS is best described with the help of 3 ideal rules, such as:

(a) Every cuckoo bird has the habit of laying an egg at a time and discards the egg in a random fashion in some nest;
(b) The best nest with better-laid eggs would carry it to the coming generations;
(c) The number of host nests available is already fixed, and the cuckoo lays the egg is denoted by the probability of host birth pa € [0, 1] [31].

3.3 Fuzzy logic

Fuzzy technique is used in a successful way in order to control in various fields, Researchers and Engineers considered this fuzzy logic algorithms is used for implementing in Embedded Systems, which performs intelligent functions. Fuzzy logic controller delivering better performance, which is best one than the traditional controller, which is proved by various researchers [32]. The blend of the PID and fuzzy inference system is an ideal approach for controlling the dynamics process and it is non-linear [33]. During the last few decades, the major area of success experienced by fuzzy logic was within the industry. The fuzzy logic application, let us mention the link prevailing among actuator outputs and sensor inputs with "If...Then..." linguistic rules type. The algorithm with fuzzy logic can interpolate and translate such rules as the non-linear mapping that prevails actuator outputs and sensor input signals to get feedback control [34]. Fuzzy logic makes the job of the human designer for fine tuning the control system with the help of trial and success method. Along with many other approaches like the GA, Artificial Neural Networks (ANN) and so on. Fuzzy logic is held as the useful tool for the control system design with non-model basis [35].

3.4 Proposed novel algorithm: advanced PSO

The proposed algorithm is a combination of PSO, FUZZY, and ACO. we have developed a new advanced PSO to control the water level in the tank system. The efficiency of the existing scheduling algorithm and advanced PSO and software implementation of the proposed algorithm is showed in a previous paper.

4 Water tank control

Water tank system is used to illustrate both advanced and traditional multivariable strategies. In Shingare and Joshi [36] quantitative feedback theory has been employed to the robust controller in order to regulate the level of liquid in two glass tanks.A multivariable laboratory process of four interconnected water tanks is considered in Vadigepalli et al. [37] for modeling and robust control of the tank level. In Vadigepalli et al. [37], flow and temperature of a water tank system are controlled employing a backstepping method, whereas a four-tank system laboratory experiment is designed in Rusli et al. [38] to illustrate the effects of time-varying dynamics. Also, apredictive fuzzy modeling technique is employed in Liutkeviius and Dainys [39] for analysis of a closed water tank and an Internal Mode Control–based robust tunable controller design technique is applied to a water tank control system in Duan et al. [40]. This water tank control can be developed through mathematical modeling. The mathematical modeling helps us to explain a system and to differentiate components to make their predictions. Mathematical modeling obtained in many forms such as statistical models, dynamical system, and differential system.

4.1 Single-input single-output (SISO) tank system

The tank level process to be simulated is single-input-single-output (SISO) tank system as shown in Fig. 1. The user can adjust the inlet flow by adjusting the control signal. During the simulation, the level 'h' will be calculated and displayed on the front panel of the tank system at any instant of time. In the SISO tank system, the liquid will flow into the tank through the valve K_1 and the liquid will come out from the tank through valve K_2. Here, we want to maintain the level of the liquid in the tank at desired value; so, the measured output variable is the liquid level h (Fig. 2).

For simulating the SISO tank system, its mathematical model can be developed [41]. The system is designed according to the mathematical model. For developing the mathematical model for SISO tank system, the density of the liquid in the inlet and outlet and in the tank is assumed to be same and also the tank has straight vertical walls.

The notations used in modeling the SISO tank system are

q_{in} Inlet volumetric flow rate (m^3/s)
q_{out} Outlet volumetric flow rate (m^3/s)

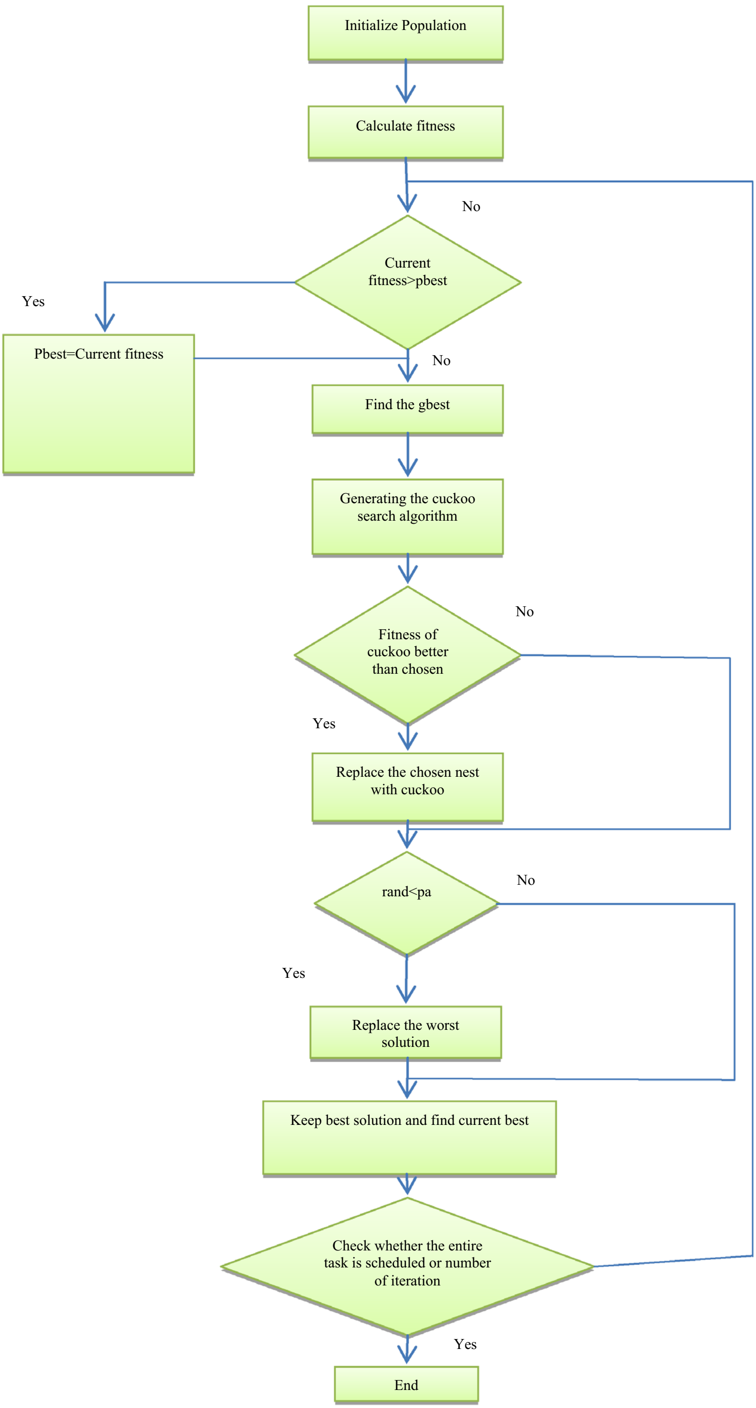


Fig. 1 Flow chart

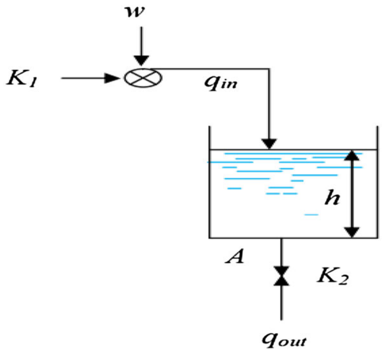


Fig. 2 Single-input single-output (SISO) tank system

V Volume of liquid in the tank (m^3)
h Height of liquid in the tank (m)
ρ Liquid density (Kg/m^3)
A Cross sectional area of the tank (m^2)

The mass of the liquid in the tank can be expressed as

$$m(t) = \rho A h(t) \tag{1}$$

The inlet volumetric flow into the tank is given by

$$q_{in}(t) = K_1 w(t) \tag{2}$$

The outlet volumetric flow through the valve is expressed as the square root of the pressure drop across the valve.

$$q_{out}(t) = K_2 \sqrt{\rho g h(t)} \tag{3}$$

According to the Mass balance equation

$$\frac{d[m(t)]}{dt} = \rho q_{in}(t) - \rho q_{out}(t) \tag{4}$$

$$\frac{d[\rho A h(t)]}{dt} = \rho K_1 w(t) - \rho K_2 \sqrt{\rho g h(t)} \tag{5}$$

$$\frac{dh(t)}{dt} = \left(\frac{1}{A}\right) * [K_1 w(t) - K_2 \sqrt{\rho g h(t)}] \tag{6}$$

In SISO If the process is initially at steady state, the inlet and outlet flow rates are equal. If the inlet volumetric flow rate is suddenly increased while the outlet volumetric flow rate remains constant, the liquid level in the tank will increase until the tank overflows. Similarly, if the outlet volumetric flow rate is increased while the inlet volumetric flow rate remains constant, the tank level will decrease until the tank is empty.

4.2 Two-input two-output (TITO) tank system

The level process to be modeled is the Two-Input-Two-Output (TITO) tank system shown in below Fig. 3. In the figure, q_{i1} and q_{i2} are the two inputs to the tank where as h_1 and h_2 are the output levels for the two tank systems [42]. The user can adjust the input by adjusting the input volumetric flow rates (q_{i1} and q_{i2}) and simultaneously the output levels 'h_1' and 'h_2' are calculated and displayed at any instant of time in the simulation.

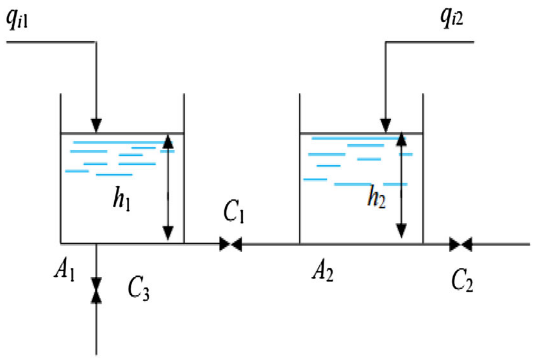


Fig. 3 Two-input two-output (TITO) tank system

An unsteady-state mass balance around the first tank can be written as

$$\rho A_1 \frac{dh_1(t)}{dt} = \rho q_{i1}(t) - \rho q_{12}(t) - \rho q_1(t) \tag{7}$$

$$A_1 \frac{dh_1(t)}{dt} = q_{i1}(t) - q_{i2}(t) - q_1(t) \tag{8}$$

An unsteady-state mass balance around the second tank can be written as

$$\rho A_2 \frac{dh_2(t)}{dt} = \rho q_{i2}(t) - \rho q_{12}(t) - \rho q_2(t) \tag{9}$$

$$A_2 \frac{dh_2(t)}{dt} = q_{i2}(t) - q_{12}(t) - q_2(t) \tag{10}$$

The flow of liquid through a valve is given by the valve equation as

$$q(t) = \frac{C_v}{7.48}\sqrt{\frac{\Delta P(t)}{G}} = \frac{C_v}{7.48}\sqrt{\frac{\rho g h(t)}{144 g_c G}} \tag{11}$$

$$q(t) = C'_v \sqrt{h(t)} \tag{12}$$

where, C_v = Valve coefficient, $h(t)$ = Level in tank, g_c = Conversion tank, $\Delta P(t)$ = Pressure drop the tank, G = Specific gravity of liquid flowing through the valve.

$$C'_v = \frac{C_v}{7.48}\sqrt{\frac{\rho g}{144 g_c G}} = C_v Z \tag{13}$$

The integration between the tanks from the valve Eq. (13) for the flow q_{12} is

$$q_{12}(t) = C'_{v12}\sqrt{h_1(t) - h_2(t)} \tag{14}$$

where,

$$C'_{v12} = C_{v12} Z$$

C_{v12} = Coefficient of the valve connected between the two tanks.

Equation (14) shows that the flow between the two tanks depends on the levels in both the tanks, each affecting the other. Hence, the system is called as an interacting system. The flow through the valve connected to the first tank is given as:

$$q_1(t) = C'_{v1}\sqrt{h_1(t)} \tag{15}$$

where,

$$C'_{v1} = C_{v1}Z$$

C_{v1} = Coefficient of the valve connected to the first tank

The flow through the valve connected to the second tank is given as

$$q_2(t) = C'_{v2}\sqrt{h_2(t)} \tag{16}$$

where,

$$C'_{v2} = C_{v2}Z$$

C_{v2} = Coefficient of the valve connected to the second tank

Substituting Eqs. (14), (15) and (16) into Eqs. (7) and (9)

$$A_1\frac{dh_1(t)}{dt} = q_{i1}(t) - C'_{v12}\sqrt{h_1(t) - h_2(t)} - C'_{v1}\sqrt{h_1(t)} \tag{17}$$

$$A_2\frac{dh_2(t)}{dt} = q_{i2}(t) - C'_{v12}\sqrt{h_1(t) - h_2(t)} - C'_{v2}\sqrt{h_2(t)} \tag{18}$$

Equations (17) and (18) are non-linear equations due to the square root terms present in the relations i.e. the outputs will vary nonlinearly with the inputs.

5 Results

In this study hardware software co-design and hardware implementation have been made in LabView (ver 15), and the findings of the proposed algorithm are shown below.

5.1 Hardware implementation of novel algorithms

The population size is considered as 30 while the input variables are named as indicators. There are two input indicators is used in the study (x & y). The output variables are named as capacitors (c1 & c2). Indicators are presented in the L.H.S and capacitors are located in the R.H.S as shown in the above Fig. 4. The maximum time taken to execute a task is mentioned in the small comment box. The output values are presented in the following Fig. 5.

The abbreviation of output parameters obtain from this method are,

Pop size—population size
Part size—particle size
P best—personal or local best
W max—maximum weight
W min—minimum weight
C1—social parameter
C2—cognitive parameter
x—current velocity
y—modified velocity
z—updated velocity
a—current search
b—modified search
n—population size

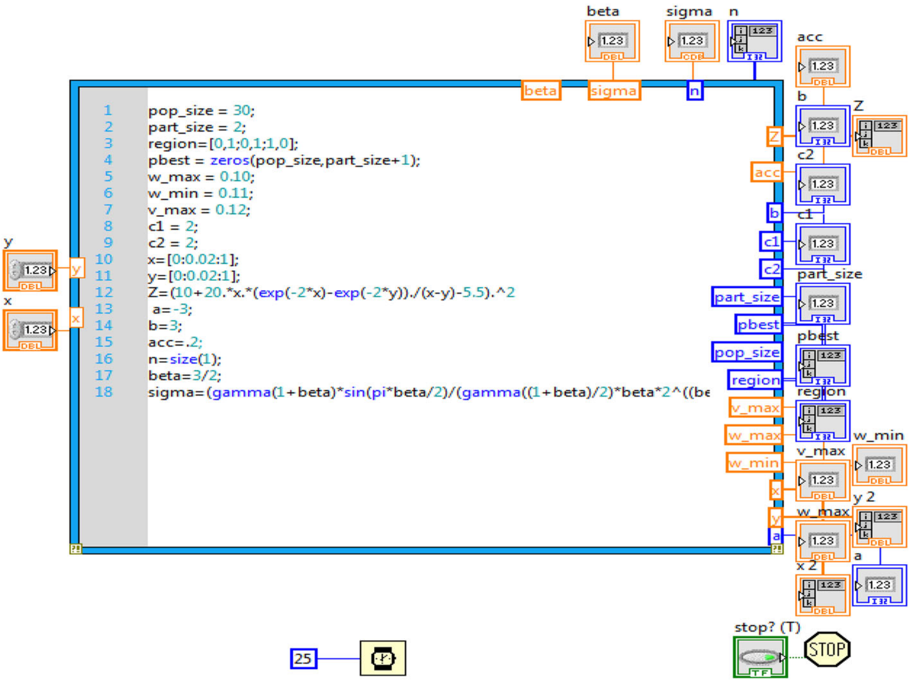


Fig. 4 Back panel

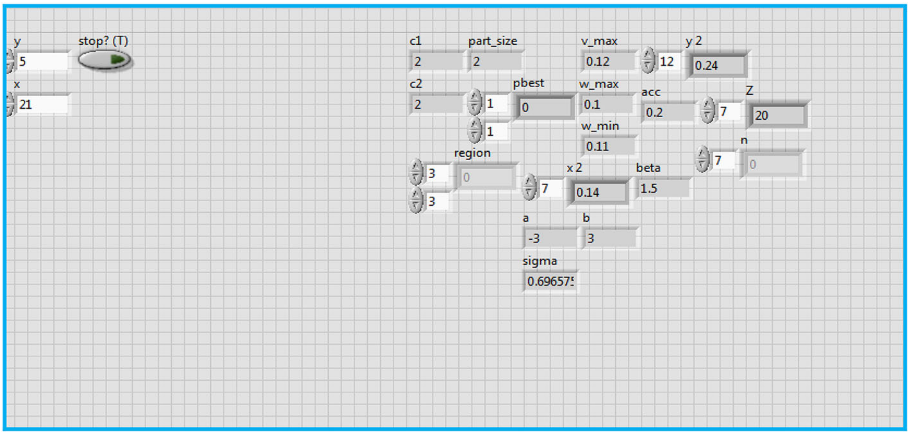


Fig. 5 Front panel

Beta—initial particle position
Sigma—updated particle position
Y2—fitness function
z—population increase
v max—maximum velocity taken by the particle to shift from one position to another

The simulation results obtained from this proposed work is shown in Fig. 5.

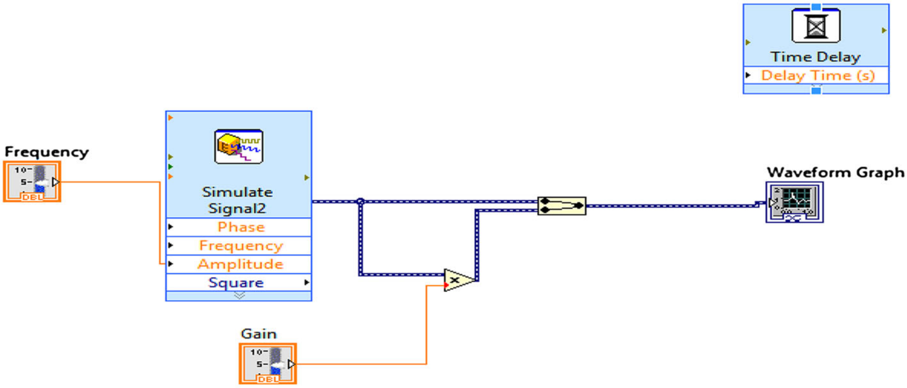


Fig. 6 Front panel

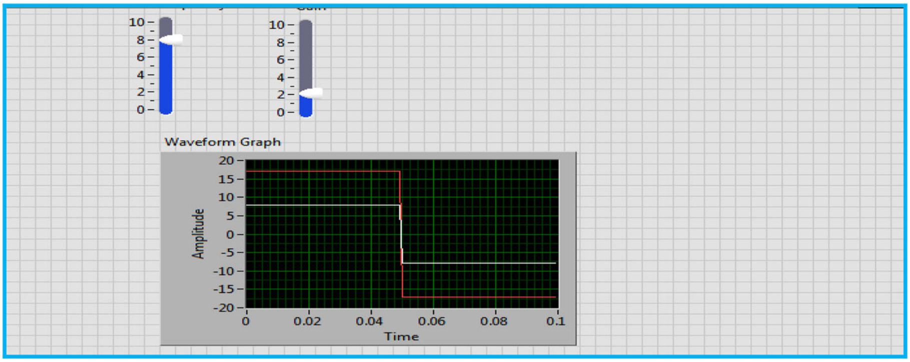


Fig. 7 Particle position graph

5.2 Hardware software co-design

Figure 6 shows the front panel diagram for proposed work. Here frequency and gain are considered as velocity and position of the particle. Signal 2 is considered as weight of the particle.

The waveform is obtained from the simulation is shown in Fig. 7. This waveform displays change in two parameters velocity (time, x-axis) and position (Amplitude, y-axis) of the particle.

The above diagram represents the hardware and software co-design implemented in LabView software. Here the frequency is given as 8 while gain as 2 when the waveform graph has been plotted; the gain gets multiplied with the frequency due to the multiplier in the design.

Figure 9 described the comparison of burst time with existing algorithm and proposed advanced PSO algorithm. The proposed PSO algorithm takes less burst time as shown in Table 1.

The bar chart of Figs. 8, 9 and 10 shows the comparison of turn-around time, burst time and waiting time with existing and proposed algorithm. Here process1, process2, process3 and process 4 are mentioned as p1, p2, p3 and p4.

In First Come First Served (FCFS) algorithm

Table 1 Algorithm details with the input parameters

Algorithms/Time (min)		FCFS		SJF		RR		Proposed advanced PSO	
	Process	Time	No. of particle	Time	No. of particle	Time	No. of particle	Time	No. of particle
Turn around time	Process 1	2	2	3	3	2	2	1	1
	Process 2	5	5	16	15	11.5	11	3	3
	Process 3	6	6	16	16	12	12	5	6
	Process 4	12	11	10	10	7	7	9	9
Burst time	Process 1	2	2	3	3	2	2	0.75	1
	Process 2	3	3	6	6	4	4	1.75	2
	Process 3	1	1	4	4	3	3	3	3
	Process 4	5	5	2	2	1	1	2.75	3
Waiting time	Process 1	0	0	2	2	0	0	0	0
	Process 2	2	2	10	10	7	7	0.75	1
	Process 3	5	5	12	12	9	9	3.5	3
	Process 4	6	6	7.5	8	5.5	6	5.75	6

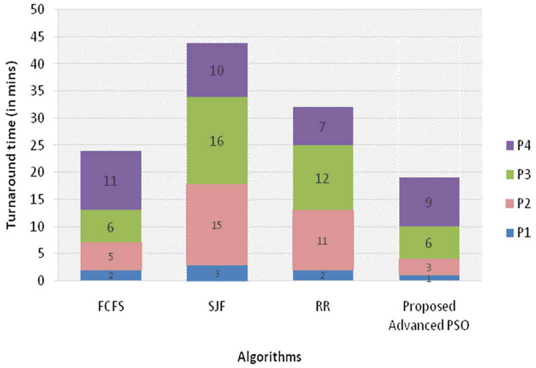


Fig. 8 Comparison of turnaround time

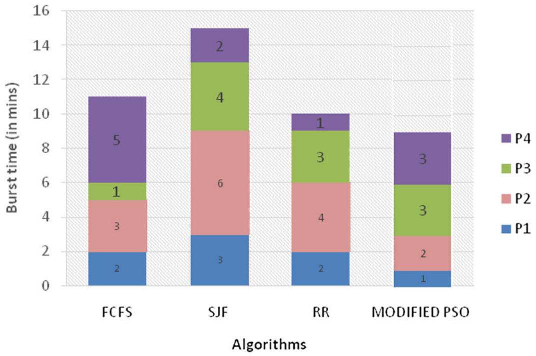


Fig. 9 Comparison of burst time

Process 1 has 2 particles for searching best position
Process 2 has 5 particles for searching best position
Process 3 has 6 particles for searching best position
Process 4 has 11 particles for searching best position

In Shortest Job First (SJF) algorithm

Process 1 has 3 particles for searching best position
Process 2 has 16 particles for searching best position
Process 3 has 15 particles for searching best position
Process 4 has 10 particles for searching best position

In Round Robin Scheduling (RR) algorithm

Process 1 has 2 particles for searching best position
Process 2 has 11 particles for searching best position
Process 3 has 12 particles for searching best position
Process 4 has 7 particles for searching best position

In Proposed Advanced PSO algorithm

Process 1 has 1 particle for searching best position
Process 2 has 3 particles for searching best position
Process 3 has 6 particles for searching best position
Process 4 has 9 particles for searching best position

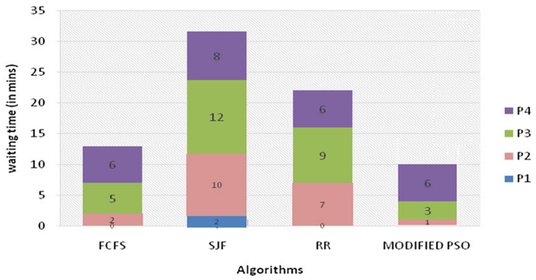


Fig. 10 Comparison of waiting time

Figure 8 described the comparison of turn-around time with existing algorithm and proposed advanced PSO algorithm. In each algorithm, different numbers of particles are chosen for searching best position. From the bar chart, proposed PSO algorithm for multiprocessor takes less time for finding best position, when compare all other algorithm. The comparison of this algorithms is shown in Table 1.

Figure 10 described the comparison of turn-around time for existing algorithm and proposed advanced PSO algorithm. The proposed PSO algorithm takes less waiting time as shown in Table 1.

6 Conclusion and future recommendation

The study has presented a novel algorithm utilizing the features of PSO, Fuzzy and Cuckoo search for the control of water level in the tank. Our previous study had implemented the proposed algorithm based on software implementation while the proposed technique showed the hardware software co-design in LabVIEW. The paper found that when comparisons with existing algorithm like FCFS, SJF and RR the modified PSO seems to be more effective regarding process time, waiting time and burst time. In future, this proposed technique can be enhanced by combining it with GA to enhance the effectiveness of the performance.

References

1. Kamieth J, Steinbach T, Korf F, Schmidt TC (2014) Design of TDMA-based in-car networks: applying multiprocessor scheduling strategies on time-triggered switched ethernet communication. In: Proceedings of 2014 IEEE emerging technologies factory automation, pp 1–9. https://doi.org/10.1109/ETFA.2014.7005119
2. Shin I, Easwaran A, Lee I (2008) Framework hierarchical scheduling, for virtual clustering of multiprocessors. In: Euromicro conference real-time systems IEEE 2008, pp 181–190. https://doi.org/10.1109/ECRTS.2008.28
3. Kwon J, Cho H, Ravindran B (2012) framework accommodating categorized multiprocessor real-time scheduling in the RTSJ. In: Proceedings of 10th international workshop on Java technologies real-time embedded system–JTRES '12. ACM Press, New York, pp 18–25. https://doi.org/10.1145/2388936.2388941
4. Nemati F, Behnam M, Nolte T (2011) Systems independently-developed real-time, on multi-cores with shared resources. In IEEE 23rd Euromicro conference real-time systems 2011, pp 251–261. https://doi.org/10.1109/ECRTS.2011.31
5. Shah A, Kotecha K (2009) Algorithm adaptive scheduling, for real-time multiprocessor systems. In: IEEE International advance computing conference IEEE 2009, pp 35–39. https://doi.org/10.1109/IADCC.2009.4808976

6. Ras J, Cheng AMK (2009) An evaluation of the dynamic and static multiprocessor priority ceiling protocol and the multiprocessor stack resource policy in an SMP system. In: 15th IEEE real-time embedded technology and application symposium, pp 13–22. https://doi.org/10.1109/RTAS.2009.10
7. Das L, Saha S (2014) A correlation framework for different resource access protocol in real time system. Int J Mod Educ Comput Sci 6:31–37. https://doi.org/10.5815/ijmecs.2014.04.05
8. Perez-Gonzalez A, Begovich O, Ruiz-Leon J (2014) Modeling of a greenhouse prototype using PSO algorithm based on a LabViewTM application. In: 11th International conference electrical engineering computer science and automation control, IEEE 2014, pp 1–6. https://doi.org/10.1109/ICEEE.2014.6978281
9. Malarvizhi K, Kiruba R (2015) Design and implementation of two tank conical interacting system using intelligent technique (PSO). In: International conference advanced computing communication systems IEEE 2015, pp 1–5. https://doi.org/10.1109/ICACCS.2015.7324101
10. Zhe D, Yuntao S, Zhijun L, Dehui S (2012) Design and implementation of an integrated experimental control platform using rapid control prototyping methods. In: IEEE 31st Chinese conference control (CCC), 2012, Hefei, pp 5896–5901
11. Schoenwetter D, Veldema R, Fey D (2015) FREACSIM–a framework for creating and simulating real-time capable network on chip systems and applications. http://www3.informatik.uni-erlangen.de/DE/Persons/schoenwe/freacsim.pdf
12. He S, Liu C, Song Z, Wang Z (2014) Real-time intelligent control of liquid level system based on MCGS and MATLAB. In: IEEE international conference machine learning and cybernetics, pp 131–136. https://doi.org/10.1109/ICMLC.2014.7009105
13. Bi Y, Yuan JS, Jin Y (2015) Beyond the interconnections: split manufacturing in RF designs. J Electron 4(3):541–564
14. Yuan JS, Bi Y (2015) Process and temperature robust voltage multiplier design for rf energy harvesting MR. J Microelectron Reliab 55(1):107–113
15. Yuan JS, Xu Y, Yen SD, Bi Y, Hwag GW (2014) Hot carrier injection stress effect on a 65 nm LNA at 70 GHz TDMR. IEEE Trans Dev Mater Reliab 14(3):931–934
16. Alasad Q, Bi Y, Yuan JS (2017) E2LEMI: energy-efficient logic encryption using multiplexer insertion. J Electron 6(1):16
17. Geetha M, Manikandan P, Shanmugapriya P, Silambarasan V, Naveen R (2013) Real-time implementation and performance analysis of two dimension PID fuzzy controller for continuous stirred tank reactor. In: Fourth international conference computational communications and networking technologies, pp 1–5. https://doi.org/10.1109/ICCCNT.2013.6726807
18. Turner H (2013) Validation framework for multiprocessor and distributed scheduling algorithms. In: Student research conference Virginia space grant, Virginia, pp 1–4. http://www.vsgc.odu.edu/awardees/20122013/abstracts/Papers-Grad/TurnerHamilton-Paper.pdf
19. Pandey S, Wu L, Guru SM, Buyya R (2010) A particle swarm optimization-based heuristic, for scheduling workflow applications in cloud computing environments. In: 24th IEEE international conference advanced information networking applications IEEE, Perth, WA, pp 400–407. https://doi.org/10.1109/AINA.2010.31
20. Selvakrishnan S, Perumal V (2010) Optimization of grid resource scheduling using particle swarm optimization algorithm, ICTACT J Soft Comput, pp 98–101. http://ictactjournals.in/paper/ijsc6_page_98_101.pdf
21. Sarathchandar AP, Priyesh V, Miriam DDH (2012) Grid scheduling using improved particle swarm optimization with digital pheromones. Int J Sci Eng Res 3:1–6. http://www.sarathchandar.in/paper/grid_scheduling_sarath.pdf
22. Hardoroudi AH, Chuprat SB (2015) Task scheduling algorithm base on particle swarm optimization (PSO) and ant colony optimization (ACO) to execute tasks in overloaded situation for preemptive system. ARPN J Eng Appl Sci 10:499–505. http://www.arpnjournals.com/jeas/research_papers/rp_2015/jeas_0215_1482.pdf
23. Kennedy J, Eberhart R (1995) Particle swarm optimization. In: Proceedings of ICNN'95 international conference neural networks, pp 1942–1948. https://doi.org/10.1109/ICNN.1995.488968
24. Parsopoulos KE, Vrahatis MN (2002) Recent approaches to global optimization problems through particle swarm optimization. Nat Comput 1:235–306. https://doi.org/10.1023/A:1016568309421
25. Van Den Bergh F, Engelbrecht AP (2006) A study of particle swarm optimization particle trajectories. Inf Sci 176:937–971. https://doi.org/10.1016/j.ins.2005.02.003
26. Clerc M, Kennedy J (2002) The particle swarm-explosion, stability, and convergence in a multidimensional complex space. IEEE Trans Evol Comput 6:58–73. https://doi.org/10.1109/4235.985692

27. Tripathi PK, Bandyopadhyay S, Pal SK (2007) Multi-objective particle swarm optimization with time variant inertia and acceleration coefficients. Inf Sci 177:5033–5049. https://doi.org/10.1016/j.ins.2007.06.018
28. Gaing Z-L (2004) A particle swarm optimization approach for optimum design of PID controller in AVR system. IEEE Trans Energy Convers 19:384–391. https://doi.org/10.1109/TEC.2003.821821
29. Kennedy J, Eberhart R (1997) A discrete binary version of the particle swarm algorithm. In: IEEE international conference systems man, cybernetics, computional cybernetics simulation, pp 4104–4108. https://doi.org/10.1109/ICSMC.1997.637339
30. Li SA, Wong CC, Yu CJ, Hsu CC (2010) Hardware, software co-design for particle swarm optimization algorithm. In: IEEE international conference system, man, cybernetics, pp 3762–3767. https://doi.org/10.1109/ICSMC.2010.5641826
31. Kamat S, Karegowda AG (2014) A brief survey on cuckoo search applications. In: International conference advanced computional communication engineering (ACCE-2014), Department of CSE & ISE, Vemana Institute of Technology, Bengaluru, India, pp 7–14
32. Nour M, Ooi J, Chan KY (2007) Fuzzy logic control vs. conventional PID control of an inverted pendulum robot. In: International conference on intellignet and advanced systems, pp 209–214. https://doi.org/10.1109/ICIAS.2007.4658376
33. Neokleous K, Neocleous C, Schizas C (2006) A comparison of classical, neural and fuzzy control for an underwater vehicle. In: Proceedings of 7th WSEAS international conference on neural networks, pp 61–66
34. Cox E (1998) The fuzzy systems handbook, 2nd edn. Academic Press Limited, London
35. Nhivekar GS, Nirmale SS, Mudholker RR (2011) Implementation of fuzzy logic control algorithm in embedded microcomputers for dedicated application. Int J Eng Sci Technol 3:276–283. http://www.ijest-ng.com/vol3_no4/ijest-ng-vol3-no4-pp276-283.pdf
36. Shingare P, Joshi MA (2007) Modeling and robust control of level in hybrid tanks. In: Proceedings of world academy of science engineering and technology, The Pennsylvania State University, pp 279–283. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.193.756
37. Vadigepalli R, Gatzke E, Doyle F (2001) Robust control of a multivariable experimental four-tank system. Ind Eng 40:1916–1927. https://doi.org/10.1021/ie000381p
38. Rusli E, Ang S, Braatz RD (2004) Quadruple tank process control experiment. J Chem Eng Educ 38:1–25. http://home.hit.no/~hansha/documents/theses/2010/4-Tank/Background/Misc/quadtank.pdf
39. Liutkeviius R, Dainys S (2005) Hybrid fuzzy model of a nonlinear plant. Inf Technol Control 34:51–56. http://www.itc.ktu.lt/index.php/ITC/article/view/11972/6645
40. Duan Y, Boulet B, Michalska H (2007) Application of IMC-based robust tunable controller design to water tank level regulation. In: Proceeding MIC '07, 26th IASTED international conference model identification, control. ACTA Press, Anaheim, pp 285–290. http://dl.acm.org/citation.cfm?id=1710649
41. She J-H, Odajima H, Hashimoto H, Higashiguchi M (1998) Flow and temperature control of a tank system by backstepping method. In: Fourth international conference on motion vibration control, ETH Zurich, Switzerland, pp 449–454
42. Pradeepkannan D, Sathiyamoorthy S (2014) Control of a non-linear coupled spherical tank process using GA tuned PID controller. In: IEEE international conference on advancecd communication control and computing technologies, pp 130–135. https://doi.org/10.1109/ICACCCT.2014.7019324