International Journal of Industrial Electronics, Control and Optimization © 2021 IECO
Vol. 4, No. 3, pp. 355-366, July (2021)

Chaos Synchronization in Josephson Junction Using a Nonlinear Model Predictive Controller Based on Particle Filter: Processor in the Loop Implementation

Aylar Khooshehmehri [1†], Saeed Nasrollahi [2], Morteza Aliyari[3]
[1, 2, 3]Faculty of Electrical and Computer Engineering, Malek-Ashtar University of Technology, Iran.

ABSTRACT

In this paper, a model predictive control approach based on a generic particle filter is proposed to synchronize two Josephson junction models with different parameters. For this purpose, an appropriate objective function is defined to assess the particles within the state space. This objective function minimizes simultaneously the tracking error, control effort, and control smoothness. The dynamic optimization problem is solved using a generic particle filter. Here, Josephson junction is described with Resistive Capacitive Inductive Shunted Josephson model, and the synchronization is obtained using the slave–master technique. Moreover, to verify the implementation capability of the proposed algorithm, a processor in loop experiment is performed. The results show that the open-loop system, without the controller, has a chaotic behavior. Numerical simulations are conducted to assess the performance of the proposed algorithm. The results show that the proposed approach can be implemented in a real-time application. Also, the performance of the suggested controller is compared with the proportional integral derivative controller and sliding mode controller.

Article Info

Keywords: Chaos dynamic, Generic particle filter, Josephson junction, Model predictive control, Synchronization.

Article History:
Received 2020-12-24
Accepted 2021-06-29

I. INTRODUCTION

The Josephson junction is a device consisting of two superconducting electrodes connected by a weak junction such as a thin insulation coating. If the intermediate insulation coating is sufficiently thin, the pair of electrons will tunnel from one superconducting electrode to another; This phenomenon is called the Josephson effect [1]. Like many mechanical and electronic systems, the Josephson junction can exhibit chaotic behavior. Researchers have studied chaos in different structures of the Josephson junction [2–4]. Since chaos is an unpredictable phenomenon and may lead to unwanted behavior in the system, it must be suppressed in many cases [5–7]. For example, in the application of Josephson junction such as Josephson parametric amplifiers [8], voltage standard [9], high-frequency oscillators [10], squid magnetometer [11], soliton transistors [12], pulse generators [13], and the biophysical function of neural the circuit [14,15], it is necessary to prevent chaos, and therefore, the chaos will be controlled in Josephson junction. Several studies are conducted to control and synchronize the chaotic behavior of the junction, which will be discussed below.

In [16], an active control method has been used to control and synchronize two coupled Josephson junction models such that the frequency of oscillation of the slave system has been following the master system; here, a linear model has been used to design the controller. In [17], a nonlinear backstepping controller has been considered to control bifurcation as well as chaos in the RCLSJ Josephson junction. A Lyapunov function has been proposed to analyze the stability of the close loop system. In [18], backstepping control theory has been

[†]**Corresponding Author**: khooshehmehri@mut.ac.ir
Tel: +98-02122945141, Fax: +98-02122959233,
Faculty of Electrical and Computer Engineering, Malek-Ashtar University of Technology, Iran.

used to control and synchronization of chaotic dynamics in RCLSJ. The recursive approach has been employed to eliminate the chaotic behavior of the RCLSJ as well as to attain global asymptotic synchronization. In [19], an active backstepping method has been used to control chaos in RCLSJ model. In the proposed approach the number of control functions has been reduced from three to one that was reduced controller complexity. In [20], a nonlinear recursive method has been used to control the bifurcation as well as chaotic behavior of the Josephson junction. Here, a control signal based on the master and slave concept has been designed. In [21], a nonlinear sliding mode controller has been used to control and synchronize the chaos of the Josephson junctions considering the uncertainty of the model. Also, the Lyapunov stability theory has been used to analyze the stability of the system. In [22,23], an iterative learning control has been used for the linearized model of the error dynamics between two Josephson junctions. A real-time feed -forward procedure that uses iterative learning, to modify the trajectory error between systems for tracking two non-identical systems, has been used. In [24], a nonlinear controller based on sliding mode and neural fuzzy has been used for control and synchronization of the error dynamics. The neural fuzzy controller is used to provide a feedback linearization and sliding mode is used to deal with uncertainty in the model. In [25] a backstepping controller is used to achieve a hybrid combination synchronization of three Josephson junctions. One chaotic Josephson junction as the master and two Josephson junctions as the slaves.

Model Predictive Control (MPC) is an interesting subject in control engineering. The benefit of MPC methods is their capability to handle dynamics with nonlinear elements and constraints [26]. MPC approaches have been employed in several practical examples, such as DC-DC Boost Converter [27], a six pulse rectifier [28], a Boost Converter [29], a quadrotor helicopter [30], and large-scale systems [31]. MPCs are classified into nonlinear and linear ones. Nonlinear Model Predictive Control (NMPC) typically involves non-convex optimization problems. An MPC problem can be expressed as an estimation problem. Likewise, an estimation problem can be expressed as a dynamic optimization problem. The combination of MPC and particle-based algorithms provides a tool for the calculation of the control signal in control [32]. In references [32–36], particle-based heuristic methods are used to compute the control signal. Also, the above-mentioned references have not used particle-based heuristic methods to synchronize two Josephson junction models with slightly different parameters.

The contribution of the current study is summarized as follows: a nonlinear model predictive control approach based on a generic particle filter is proposed to synchronize two Josephson junction models with different parameters. For this purpose, an appropriate objective function is defined to assess the particles within the state space. This objective function minimizes simultaneously the tracking error, control effort, and control smoothness. This nonlinear model predictive control problem, which is the dynamic optimization problem is solved using a generic particle filter. To the author's knowledge, the use of a generic particle filter to synchronize two Josephson junction models with different parameters has not been proposed yet. The performance of the suggested approach is achieved in simulations using the Resistive Capacitive Inductive Shunted Josephson model. Numerical simulations are made to confirm the validity of the suggested controller. Moreover, to verify the implementation capability of the proposed algorithm, a processor in the loop experiment is performed.

In this paper, the nonlinear Josephson junction equations are described in section 2. Section 3 explains the controller design. In Section 4, the results of nonlinear simulation and controller performance are presented. Finally, the conclusion is presented.

II. STATEMENT OF THE PROBLEM

The Josephson junction can be described with various electrical models such as Resistive Shunted Josephson (RCJ), Resistive Capacitive Shunted Josephson (RCSJ), and Resistive Capacitive Inductive Shunted Josephson (RCLSJ) [1,2]. Fig. 1 shows the RCSJ model; here, θ is the phase difference between the two superconductors in the junction, R is the junction resistance, and C is the junction capacitor.

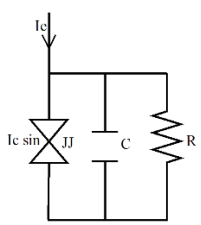


Fig. 1 RCSJ Model

When the external current I_e containing a direct current and alternating current is applied to the junction, the voltage of the two ends of the junction decreases. According to the I-V curve of the junction shown in Fig. 2, the hysteresis behavior is observable in the current-voltage characteristic of the junction.

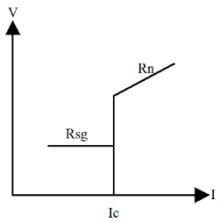


Fig. 2 Current-voltage diagram of Josephson junction

Where R_n is the normal junction resistance, R_{sg} is the sub-gap junction resistance, and I_C is the critical current of the junction. The differential equations governing the model presented in Fig. 1 are as follows [17,20]:

$$c\frac{dv}{dt}+\frac{v}{R}+I_c\sin\theta=I_e=i_o+i_1\sin(\omega t) \quad (1)$$

$$(\frac{h}{2\pi e})\frac{d\theta}{dt}=v \quad (2)$$

Where v is the voltage of the two ends of the junction, h is the Planck constant, e is the electric charge. Dynamic equations show that applying external direct current, ($I_e=i_0$), results in chaotic behavior. The following equation has been achieved by inserting equation 2 in equation 1 and normalizing it [17]:

$$\frac{d^2\theta}{dt^2}+\beta\frac{d\theta}{dt}+\Omega_0^2\sin\theta=A_0+A_1\sin(\omega t) \quad (3)$$

Where $\beta=1/RC$, $\Omega_0=(2\pi eI_c/hC)^{0.5}$, $A_0=(2\pi ei_0/hC)$, and $A_1=(2\pi ei_1/hC)$. When the external alternating current is applied to the Josephson junction, the chaotic behavior is visible at a critical value of the current which is due to the hysteresis behavior of the junction curve [37]. Another model is presented regarding hysteresis behavior, in which the linear resistance in the RCSJ model is replaced with the nonlinear model, $R(V)$, and is defined as follows:

$$R(V)=\begin{cases}R_n & \text{if } |V|>V_g\\ R_{sg} & \text{if } |V|\le V_g\end{cases} \quad (4)$$

Where V_g is the junction gap voltage, $R(V)$ is the step function of the two resistors R_n and R_{sg} which is synchronized well with the practical behavior of the junction. Such parallel resistors are required in high-frequency applications such as Josephson resonators. These parallel resistors and associated wiring result in series resistor inductor effects. Numerical studies conducted on this inductive effect confirm the practical characterization results of the junction at different temperature conditions. Then, the revised RCLSJ model including parallel self-induction is proposed. The RCLSJ model of the Josephson junction is shown in Fig. 3.

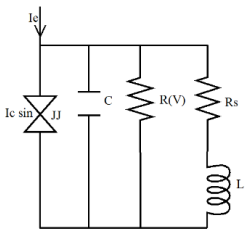


Fig. 3 RCLSJ model

Here, R_s and L are respectively the parallel resistance and parallel self- induction. Also, I_s is the current passing the parallel self- induction. The dynamic equations of this junction are as follows [17,20]:

$$c\frac{dv}{dt}+\frac{v}{R}+I_c\sin\theta+I_s=I_e=i_o+i_1\sin(\omega t) \quad (5)$$

$$(\frac{h}{2\pi e})\frac{d\theta}{dt}=v \quad (6)$$

$$L\frac{dI_s}{dt}+R_sI_s=v \quad (7)$$

The normalized mathematical model of the equations is as follows [17]:

$$\beta_C\ddot{\theta}+g(V)\dot{\theta}+\sin\theta+I_s=i_o+i_1\sin((\omega/\omega_0)\tau) \quad (8)$$

$$\dot{\theta}=V \quad (9)$$

$$\beta_L\dot{I}_s+I_s=V \quad (10)$$

Where $\tau=\omega_0 t$ and $V=(v/I_sR_s)$. Here, the Josephson junction shows chaotic behavior by choosing values of $\beta_C=0.707$, $\beta_L=2.6$, $i_0=1.20$, and initial values of $(0.8,0.58,0.4)$ for states of the system [18]. The purpose of this paper is to synchronize Josephson junction involving chaotic behavior with Josephson junction that does not show chaotic behavior by appropriate initial values and fixed parameters. Assuming $\theta=x_1$, $V=x_2$, and $I_s=x_3$, the state space equations are rewritten as follows:

$$\dot{x}_1=x_2+u_1 \quad (11)$$

$$\dot{x}_2=\frac{1}{\beta_C}(i_0+i_1\sin(\frac{\omega}{\omega_0}\tau)-g(x_2)x_2 \quad (12)$$
$$-\sin(x_1)-x_3)+u_2$$

$$\dot{x}_3=\frac{1}{\beta_L}(x_2-x_3)+u_3 \quad (13)$$

$$g(V)=\begin{cases}0.366 & \text{if } |V|>2.9\\ 0.061 & \text{if } |V|\le 2.9\end{cases} \quad (14)$$

Now, the nonlinear model predictive control based on Particle Filter (PF) will be expressed for chaotic system synchronization. For this purpose, the slave system will be considered with the following equations:

$$\dot{\mathbf{x}}_s = \mathbf{F}_s(\mathbf{x}_s) + \mathbf{u} \quad (15)$$

In equation (15), $\mathbf{x}_s$ is the state variable, $\mathbf{u}$ is the input, and $\mathbf{F}_s$ is the function with appropriate dimensions. Also, the master system equations are considered as follows:

$$\dot{\mathbf{x}}_m = \mathbf{F}_m(\mathbf{x}_m) \quad (16)$$

Here, the purpose is to design the control law in a way that the slave state variables follow the master system state variables. To do this, the synchronization error is defined as: $\mathbf{e(t)} = \mathbf{x}_s - \mathbf{x}_m$. The system error equations can be written as follows using equations (15) and (16):

$$\dot{\mathbf{e}} = \mathbf{F}_s(\mathbf{x}_s) - \mathbf{F}_m(\mathbf{x}_m) + \mathbf{u} \quad (17)$$

Algorithm. 1 Pseudo-Code of the proposed control algorithm

Set the number of particles, prediction horizon, and weighting factors.
Initialize randomly the vector of future control sequence, $\mathbf{u}_j(0)$, for $j \in [1, N]$.
While (until the end of simulation) **do**
Generate the control signals by Eq. (18)
for $j = 1$ to N **do**
Predict the future states by Eq.(19).
Calculate the cost function and weights by Equations (20)-(21).
Normalize the weights by Eq. (22).
Resample the particles
Calculate the control signals by Eq.(24).
end for
end while

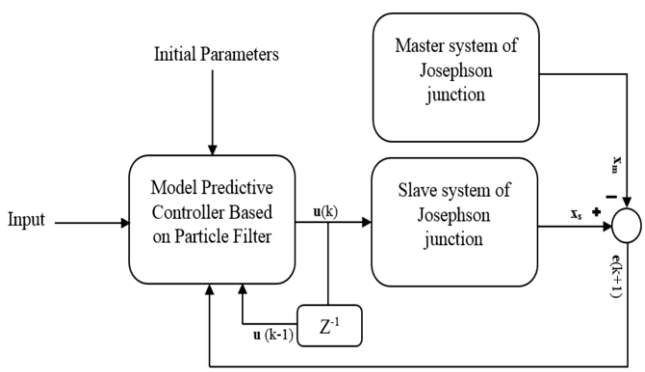


Fig. 4. Block diagram of nonlinear model predictive controller

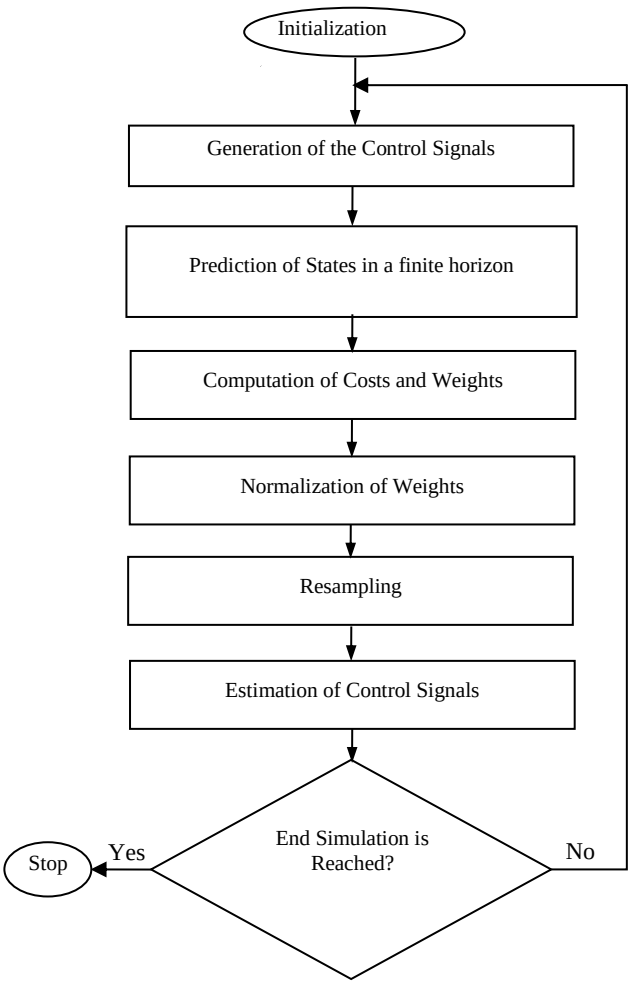


Fig. 5 Flowchart of nonlinear model predictive controller

III. Nonlinear Model Predictive Control Based on Particle Filter

The current study utilizes a single generic particle filter to estimate the finite horizon optimal controls in order to synchronize the chaos in the Josephson junction. The block diagram and flowchart of the proposed control algorithm are illustrated in Fig. 4 and Fig. 5, respectively. The proposed control algorithm has a main loop; it iterates until the end time of the simulation. At first, the initial control signals are generated, randomly. Then, the future control signals are calculated in a finite horizon, using $u(k) = u(k-1) + v(k-1)$, as proposed in [34]. Next, the future states are predicted using the control signals and, the defined cost function is calculated for each particle. A weight is assigned to each particle based on its cost. Weights are normalized. Resampling is performed. Finally, the control signals are estimated based on the current distribution of resampled particles. In addition, the pseudo-code of the PSO-DGG is shown in the algorithm. 1. In the next subsections, these steps are explained in detail.

A. Initialization

The proposed control algorithm has some parameters, set at the beginning. Moreover, for the j-th particle $(j=1,...,\mathrm{N})$, the initial value of future control sequence ($\mathbf{u}_j$) is sampled. A uniform random generator is used to generate $\mathbf{u}_j$ within an adequate range of control.

B. Generation of the Future Control Signals

The future control signals are generated for particle *j* as

$$\mathbf{u}_j(k+i-1)=\mathbf{u}_j(k+i-2)+\boldsymbol{\upsilon}_j(k+i-2), \quad (i=1,\ldots,\mathrm{T_p}) \tag{18}$$

Where, $\boldsymbol{\upsilon}$ is a Gaussian random variable; also, k and $\mathrm{T_p}$ are the time step and the prediction horizon, respectively. It should be noted that $\mathbf{u}_j(k-1)=\hat{\mathbf{u}}(k-1)$.

C. Prediction of the Future States

Using the model presented in equation (17), the future of particles are predicted as:

$$\mathbf{e}_j(k+i)=\mathbf{F}_s(\mathbf{x}_{s_j}(k+i-1))-\mathbf{F}_m(\mathbf{x}_{m_j}(k+i-1))+\mathbf{u}_j(k+i-1) \quad (i=1,\ldots,\mathrm{T_p}) \tag{19}$$

D. Computation and Normalization of Weights

A weight is assigned to each particle based on its cost. The cost of the particle j at time step k, denoted by $J_j(k)$, is computed as

$$J_j(k)=\mathrm{w_{TE}}\sum_{i=k}^{k+\mathrm{T_p}}\left(\mathbf{e}_\mathrm{d}(i)-\mathbf{e}_j(i)\right)^\mathrm{T}\left(\mathbf{e}_\mathrm{d}(i)-\mathbf{e}_j(i)\right)+\mathrm{w_{CE}}\sum_{i=k}^{k+\mathrm{T_p}-1}\mathbf{u}_j(i)^\mathrm{T}\mathbf{u}_j(i)+\mathrm{w_{CS}}\sum_{i=k}^{k+\mathrm{T_p}-1}\Delta\mathbf{u}_j(i)^\mathrm{T}\Delta\mathbf{u}_j(i) \tag{20}$$

where $\mathrm{w_{CS}}$, $\mathrm{w_{CE}}$, and $\mathrm{w_{TE}}$ are weighting factors that control the importance of Control Smoothness (CS), Control Effort (CE), and Tracking Error (TE), respectively. After calculation of the cost, the weight of particle j at time step k, denoted by $w_j(k)$, is evaluated using the following normal function, proposed in [38]:

$$w_j(k)\equiv\exp\left(-0.5J_j(k)\right) \tag{21}$$

Finally, the weights are normalized as follows:

$$\overline{w}_j(k)=\frac{w_j(k)}{\sum_{j=1}^{\mathrm{N}}w_j(k)} \tag{22}$$

Where $\overline{w}_j(k)$ is the normalized weight of particle j at time step k. Therefore, the summation of normalized particles' weight is one.

E. Resampling

After a few iterations, most particles will have negligible weights. Therefore, a large computational effort is spent to update particles whose contribution to the estimation of states is negligible. In other words, the number of effective particles is decreased. This phenomenon is called degeneracy. This phenomenon is an unwanted effect in particle filters. A suitable measure of degeneracy is the effective sample size, defined as [39]:

$$\hat{N}_{\mathrm{eff}}=\frac{1}{\sum_{j=1}^{\mathrm{N}}(\overline{w}_j(k))^2} \tag{23}$$

Considering extreme cases of the problem, it can easily be shown that $1\le\hat{N}_{\mathrm{eff}}\le\mathrm{N}$. Small effective sample size indicates significant degeneracy, i.e., when $\hat{N}_{\mathrm{eff}}$ falls below some threshold N_T [39]. Whenever a significant degeneracy is observed, resampling can be used to reduce the effect of degeneracy [39]. Resampling is performed in different ways. In this paper, systematic resampling [40] is utilized and performed in every iteration. In this method, $\tilde{\mu}\sim U[0,1)$ and $\mu_j=(\tilde{\mu}+(j-1))/\mathrm{N}$ are defined. Then, μ_j is used to select the new *j*-th particle as described in [40]. Resampling is performed based on the normalized particles' weight. The particles' weight depends on their control smoothness, tracking error, and control effort.

In systematic resampling, a particle may be selected more than one time (replication) and a particle may never be selected. In other words, high-performance particles have more chance to be selected or replicated and low performance particles have less chance to be survived. A selected particle has an estimation of the future control sequence, $u(k+i-1)$, $(i=1,...,\mathrm{T_p})$. The estimation of control is performed using the estimated control of resampled particles.

F. Estimation of Control Signal

After resampling, the control signal vector of particle j at time step k is estimated as:

$$\hat{\mathbf{u}}(k)\approx\sum_{j=1}^{N}\overline{w}_j(k)\mathbf{u}_j(k) \tag{24}$$

Where the normalized weights $\overline{w}_j(k)$ are calculated using equation (22). The estimated vector consists of the estimated control sequences. Therefore, the estimated control signals for time step k (elements $1, \mathrm{T_p}, 2\mathrm{T_p}$ of $\hat{\mathbf{u}}$) are calculated using the estimated control signals of N resampled particles (element $1, \mathrm{T_p}, 2\mathrm{T_p}$ of $\hat{\mathbf{u}}_j, j=1,...,\mathrm{N}$).

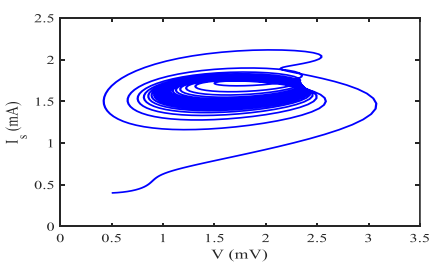
IV. Simulation Results

In this section, the results of the simulations, in Matlab software, are presented. The purpose of the simulation is to synchronize the slave with the master system. So that, the error between master and slave variables converges to zero. The constant parameter values of the dynamic equations of the Josephson junction are given in TABLE I. To solve the differential equations, the Matlab Ordinary Differential Equation function solver (ODE45) is used. Also, the values of the control parameters are as follows: $w_{TE} = 20$, $w_{CE} = 1$, $w_{CS} = 0.5$, $T_P = 10$, $N = 300$, and $\sigma_v = 0.1$. The relative importance of the corresponding performance indices determines the value of weighting factors. The value of the prediction horizon affects the overshoot of the system [41]. Also, The lower and upper bounds of the control signal determine the maximum value of σ_v [34]. The performance and the computational outlay of the proposed algorithm depend on the number of particles.

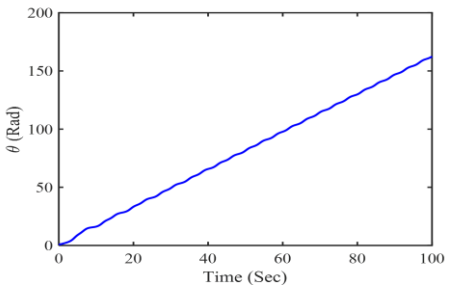
TABLE I
Parameters of RCLSJ Problem

Parameter	Master Model	Slave Model
β_C	-0.05	0.707
β_L	2.6	2.6
i_0	0.9	1.20

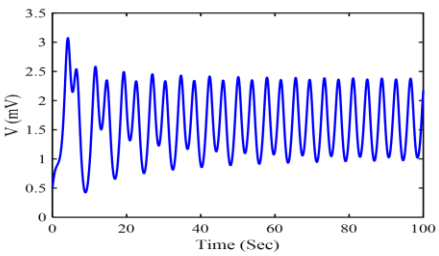
The slave system phase portrait has been shown in Fig. 6 (a) for the initial conditions (0.8,0.5,0.4). As can be seen, the slave system has chaotic behavior for these parameters and initial conditions. Fig. 6 (b)-(d) show the phase difference between the two superconductors at the junction, voltage, and current, respectively. The results of these initial conditions show that the value of phase difference has an unstable and incremental behavior over time, while the values of voltage and current show an oscillatory behavior. This oscillatory behavior in the Josephson junction characteristic curve will be unfavorable in the form of a high-frequency system, and controlling this chaotic behavior is required in order to use this segment in the middle class of a system.



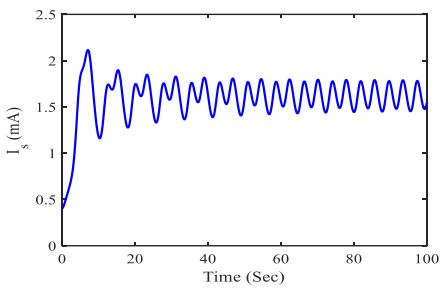
(a) the phase portrait



(b) the phase difference



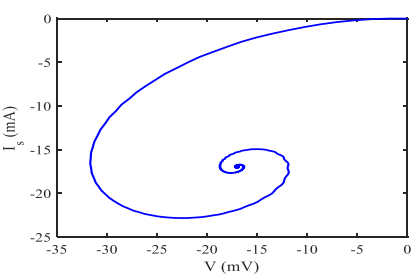
(c) the voltage



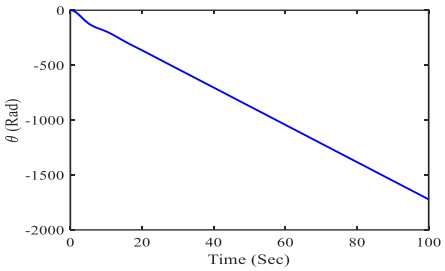
(d) the current versus time

Fig. 6 The slave system: (a) the phase portrait; (b) the phase difference; (c) the voltage; (d) the current versus time.

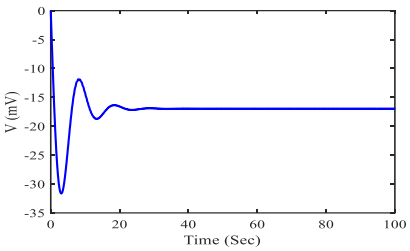
Moreover, the master system phase portrait has been shown in Fig. 7 (a) for the initial conditions (0,0,0). The master system has stable behavior for these parameters and initial conditions. Fig. 6 (b)-(d) show the phase difference, voltage, and current, respectively. It is clear that the two systems are definitely unsynchronized for different initial conditions. As can be seen in Fig. 7, if the current through the junction is varying with time then the phase difference across it must also be changing with time, and it can be shown, Eq. (9), that a voltage V is developed across it, related to the rate of the phase difference.



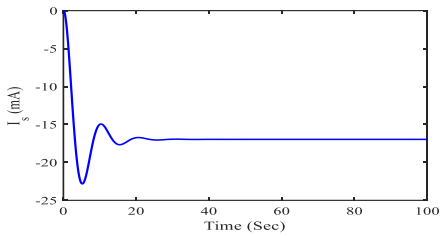
(a) the phase portrait



(b) the phase difference



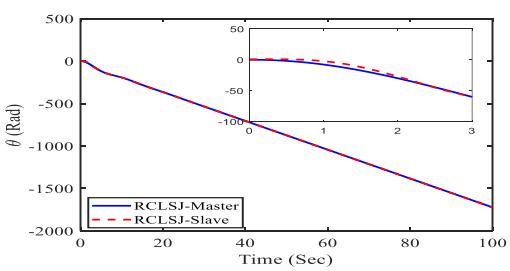
(c) the voltage



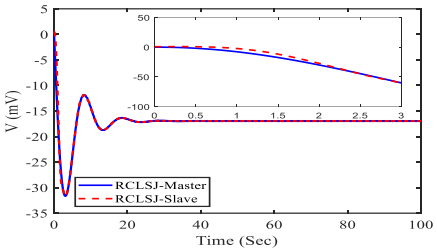
(d) the current versus time

Fig. 7. The master system: (a) the phase portrait; (b) the phase difference; (c) the voltage; (d) the current versus time

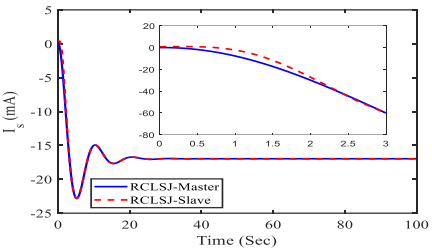
Fig. 8 shows the state variables of the master and slave systems. As shown in Fig. 8 (a)-(c), the objective is to synchronize the slave system with the master system. In this simulation scenario, the desired value of the phase difference is a ramp with slope -17 Rad, the desired values of the voltage is -17 V, and the desired values of the current is -17 mA. By comparing the simulation results, it is clear that the system has improved from its chaotic behavior and shows a stable manner. The results show that the phase difference, the voltage, and the current of the slave system will reach the desired value after about 3, 3, and 4 seconds, respectively.



(a) phase difference



(b) voltage



(c) current versus time.

Fig. 8. The state variable of the slave and master systems when the controller is applied: (a) phase difference; (b) voltage; (c) current versus time.

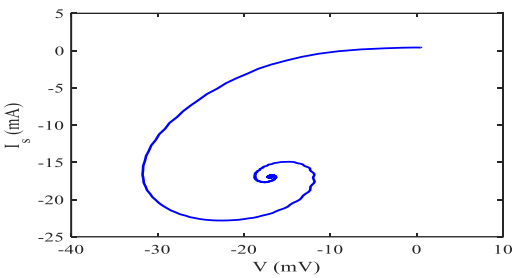
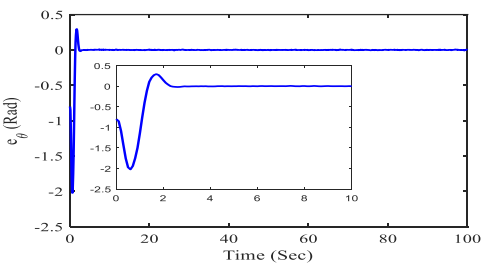
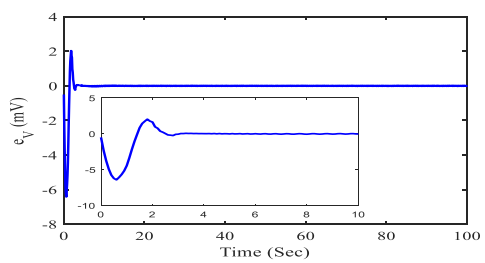


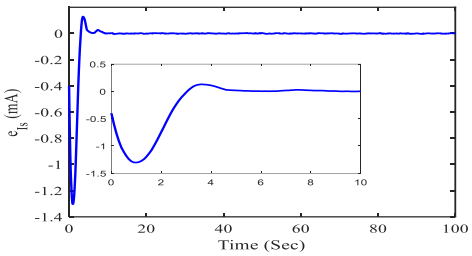
Fig. 9. Slave system phase plane when the controller is applied



(a) Phase difference tracking error

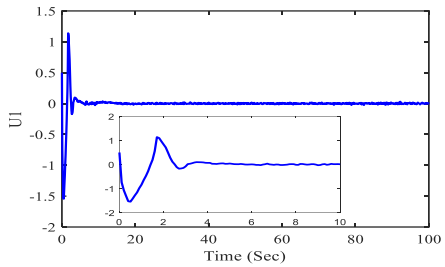


(b) voltage tracking error

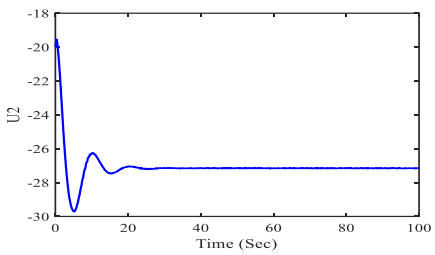


(c) current tracking error when the controller is applied.

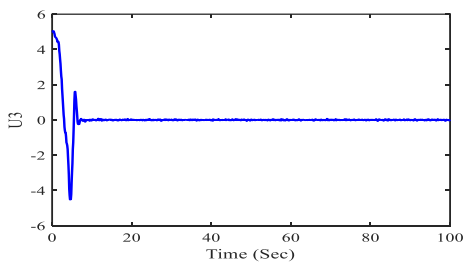
Fig. 10. The tracking error of state variables: (a) Phase difference tracking error; (b) voltage tracking error; (c) current tracking error when the controller is applied.



(a) u_1 versus time



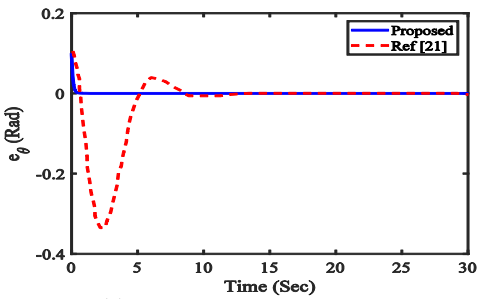
(b) u_2 versus time



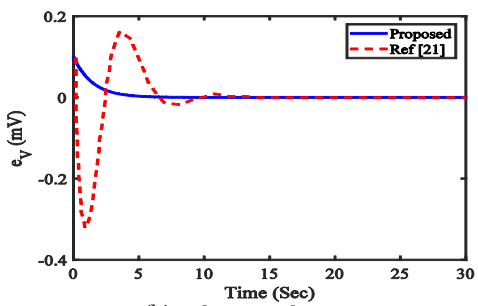
(c) u_3 versus time

Fig. 11. Control signals versus time: (a) u_1 versus time; (b) u_2 versus time; (c) u_3 versus time

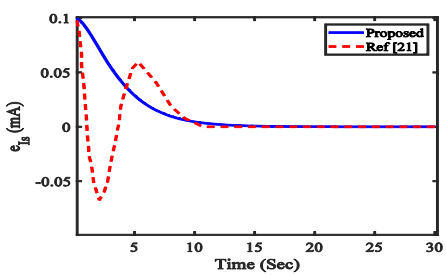
Fig. 9 shows the slave system phase plane when the controller is applied. It is clear that the slave systems are now well synchronized and have a stable response. Fig. 10 (a)-(c) illustrates the tracking error of the state variables. As can be seen in Fig. 10, the error dynamics of state variables have been reached zero. Furthermore, the simulation results show how these control signals are able to make the two systems synchronized. For additional investigation, Fig. 11 shows the time history of the control signals. As can be seen, the control signals become stable after a while. Also, the performance of the suggested controller is compared with the Proportional Integral Derivative (PID) controller and sliding mode controller [21]. For assessment, certain criteria, including Integral of Time multiplied Absolute Error (ITAE), Integral of Absolute Error (IAE), Integral of Time multiplied Square Error (ITSE), and Integral of Square Error (ISE) are calculated for the three system state variables. Control effort is also calculated to show energy consumption and their values are shown in TABLE II. For this purpose, the mean square integral of the control signals is calculated. Parameters of the PID controller are as: $P = 6.7$, $I = 0.6$, and $D = 2.1$. These parameters are achieved using the Ant Colony optimization algorithm to minimize the IAE. As shown in Table 2, results demonstrate the benefit of the suggested method in comparison with the PID and [21]. Fig. 12 compares the performance of the proposed algorithm with [21]. As presented in Fig. 12, the overshoot and the settling time of the proposed controller are less than those of [21]. Also, a Monte Carlo [42] simulation is achieved to compute the statistical performance and convergence of the suggested control algorithm. For this purpose, the ISE is calculated for the tracking error of the voltage. The average of the ISE versus the number of runs is shown in Fig. 13 for 50, 100, 200, and 500 runs. According to Fig. 13, performing 100 runs seems to be enough for convergence of the statistical properties.



(a) Phase difference tracking error



(b) voltage tracking error



(c) current tracking error

Fig. 12. Comparison with reference [21]: (a) Phase difference tracking error; (b) voltage tracking error; (c) current tracking error

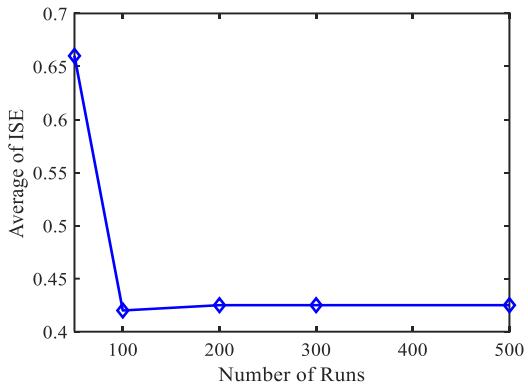
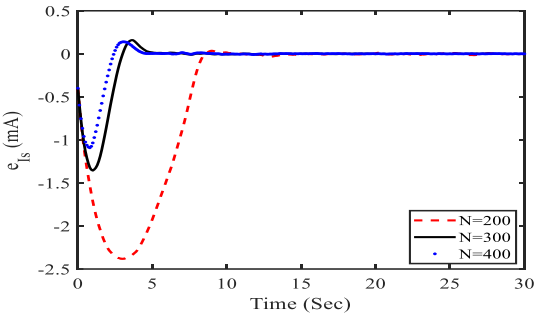


Fig. 13 Average of the ISE versus the number of run

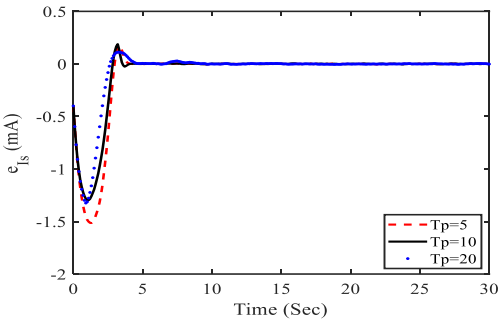
TABLE II
Comparison of the Proposed Controller with PID and Ref [21]

Controller	Control effort	State variable	Time	ISE	IAE	ITSE	ITAE
Proposed control algorithm	2.45×10^5	θ	20	2.54	4.75	0.46	4.79
		V		0.66	6.56	1.25	40.75
		I_s		10.46	13.94	7.36	14.51
PID	2.84×10^5	θ	20	2.83	4.98	0.52	5.01
		V		0.74	6.85	1.51	42.34
		I_s		11.01	14.21	7.79	14.86
Proposed control algorithm	3.34×10^5	θ	30	0.09	1.33	0.02	0.83
		V		0.12	1.59	0.02	0.76
		I_s		0.06	1.19	0.11	2.58
Ref [21]	3.79×10^5	θ	30	1.29	5.82	2.97	14.18
		V		1.02	5.12	1.23	8.33
		I_s		0.19	2.44	0.15	4.66

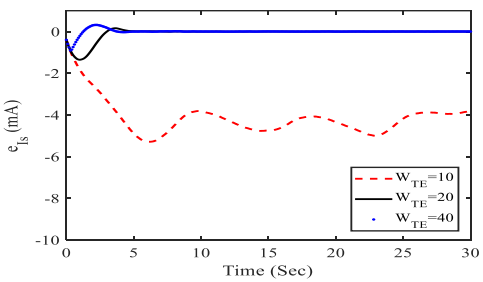
To achieve the effect of some parameters in the proposed control algorithm based on PF, a sensitivity evaluation is done on the error signal. The effect of the number of particles has been illustrated in Fig.14 (a). As can be seen, reducing the value of particles increases the error. Fig.14 (b) illustrate the effect of the prediction horizon. Increasing the value of the prediction horizon reduces overshoot. Moreover, reducing the weight of the tracking error to 10 causes the divergence of the error. The authors have implemented proposed controller in MATLAB, and a computer with a Core 5 Duo 2.9 GHz CPU and 4 GByte RAM is used for simulation. The computing time for one step of the problem, is approximately 0.0089 sec, while the sampling time of this problem is 0.01 sec.



(a) effect of particle numbers



(b) prediction horizon



(c) the weight of tracking error

Fig. 14 Sensitivity evaluation of the offered control algorithm: (a) effect of particle numbers; (b) prediction horizon; (c) the weight of tracking error

A. Processor in the Loop Experiment

In this section, the proposed control algorithm is evaluated through a PIL experiment. PIL is used to show the implementation capability of the algorithm [43]. Schematic of the experiment is illustrated in Fig. 15. To perform the PIL experiment, "Simulink Real-Time" is utilized. Simulink Real-Time [44] enables real-time execution of the pursuer-target simulation model. Moreover, the "run on hardware" tool compiles C code generating from the proposed algorithm and programs the hardware device (Arduino Due) using the Simulink Coder. A serial link is used to send the outputs of the pursuer-target simulation to the hardware. The hardware device calculates the pursuer acceleration command and sends it back to the computer to simulate the pursuer-target engagement in the next time step. Results of the PIL experiment are shown in Fig. 16. Comparison of the implementation results and the numerical simulation are provided in Fig. 16 (a) and (b) for the Phase difference tracking

error and the control signal, respectively. These results demonstrate that the proposed algorithm has been successfully implemented on the processor. Ideally, the PIL test and simulation results should be the same. There are small differences between the experimental results obtained using the PIL test and the simulation results. These differences are usually caused by practical problems such as the delay of received packets and the stochastic nature of the noise from one experiment to another. By using the User Datagram Protocol (UDP) protocol, instead of serial connection, send and receive data is done without any connection delay and reliable packets.

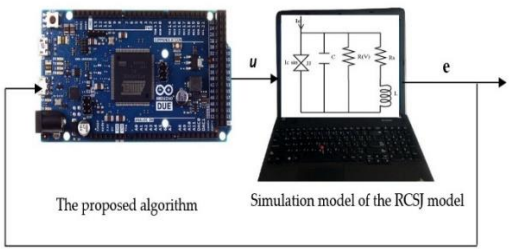
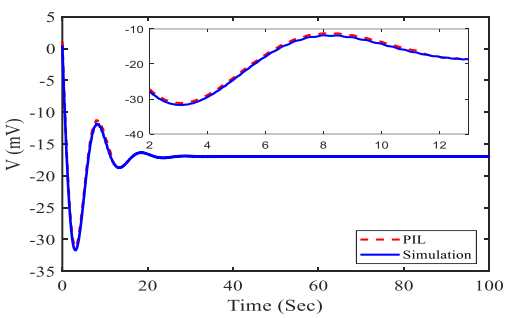
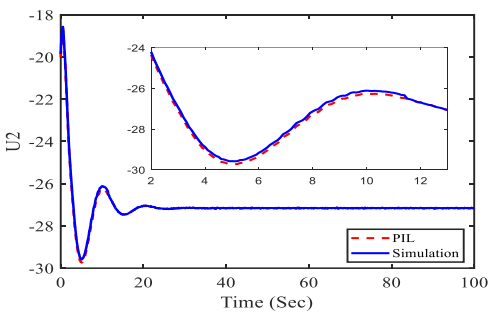


Fig. 15. Schematic of the PIL experiment used to verify the proposed algorithm



(a) the voltage versus time



(b) the control signal versus time.

Fig. 16. Comparison of the PIL experiment with numerical simulation: (a) the voltage versus time; (b) the control signal versus time.

V. Conclusion

In this paper, a control approach based on the generic particle filter was proposed to synchronize two Josephson junction models with different parameters. For this purpose, an appropriate cost function was defined to calculate the particles within the state space. The synchronization was obtained using the slave–master technique. Here, Josephson junction is described with Resistive Capacitive Inductive Shunted Josephson model. Moreover, the implementation of the proposed algorithm on a target platform was conducted successfully through the PIL experiment. A Monte Carlo simulation was done to calculate the statistical performance of the offered algorithm. The performance of the suggested method was compared with the PID and sliding mode controller. Also, a sensitivity evaluation was done on the error signal to achieve the effect of the controller parameters. The results show that the offered approach can effectively synchronize two Josephson junction models with different parameters. Clearly, there are some weaknesses. First, calculating the analytical stability of the proposed algorithm is hard. Secondly, in some cases, one cannot say anything about the number of particles needed for calculating the control signals. For future work, it is recommended to work on these two weaknesses. That is, to prove the stability analytically and to present a systematic method for the optimal determination of the number of particles.

References

[1] KK. Likharev, *Dynamics of Josephson Junctions and Circuits*, 1th ed, Gordon and Breach science publishers, 1986.

[2] SK. Dana, DC. Sengupta, KD.Edoh, “Chaotic Dynamics in Josephson Junction”. *IEEE Trans Circuits Syst I Fundam Theory*, Vol. 48, No. 8, pp. 990-996, Apr 2001.

[3] F. Salam, S.Sastry, “Dynamics of the Forced Josephson Junction Circuit: the Regions of Chaos”. *IEEE Trans Circuits Syst*, Vol. 32, No. 8, pp. 784-796, Aug 1985.

[4] H. Mehrara, F. Raissi, A. Erfanian, “Vortex-Antivortex Pair Interaction With Microwave Standing Waves: A Chaos Analysis of Josephson Fluxonic Diode for Microwave Applications”, *IEEE Trans Appl Supercond*, Vol. 29, No. 7, pp. 150-158, Oct 2019.

[5] B. Rezaie, M.R.J. Motlagh, M. Analoui, S. Khorsandi, “Stabilizing fixed points of time-delay systems close to the Hopf bifurcation using a dynamic delayed feedback control method”, *J. Phys. A Math. Theor*, Vol. 42, No. 39, pp. 1-24, Sep 2009.

[6] S.H. Shahalami, F. Rajab Nejad, “Design of Adaptive Back-Stepping Controller for Chaos Control in Boost Converter and Controller Coefficients Optimization Using CHPSO Algorithm”, *Int. J. Ind. Electron. Control Optim*, Vol. 3, No. 3, pp. 249–257, Sum 2020.

[7] K. Akbari, B. Rezaie, S. Khari, “Designing full-order sliding mode controller based on ANFIS approximator for uncertain nonlinear chaotic systems”, *Int. J. Ind. Electron. Control Optim*, Vol. 2, No. 1, pp. 39–46, Win 2019.

[8] C. Eichler, A. Wallraff, “Controlling the Dynamic Range of a Josephson Parametric Amplifier”, *EPJ Quantum Technol*, Vol. 1, No. 2, pp. 1–19, Dec 2014.

[9] S.P. Benz, C.A. Hamilton, “Application of the Josephson effect to voltage metrology”. *Proc. IEEE, pp.* 1617–1629, 2004.

[10] DS. Goldobin, LS. Klimenko, “Resonances and Multistability in a Josephson Junction Connected to a Resonator”, *Phys Rev E*, Vol. 97, No. 2, pp. 022203, Feb 2018.

[11] J. Diggins, JF. Ralph, TP. Spiller, TD. Clark, H. Prance, RJ. Prance, et al, “Chaos Generated Noise in Radio Frequency

SQUID Magnetometers". *AIP Conf. Proc.*, Vol. 371, Apr 1996

[12] F. Raissi, A. Khooshemehri, A.Erfanian, "Three-Terminal Superconducting Digital Transistor". *IEEE Trans Appl Supercond*, Vol. 29, No. 4, pp. 1-6, Dec 2018.

[13] CA. Donnelly, JA. Brevik, NE. Flowers-Jacobs, AE. Fox, PD. Dresselhaus, PF. Hopkins, et al, "Quantized Pulse Propagation in Josephson Junction Arrays", *IEEE Trans Appl Supercond*, Vol. 30, No. 3, pp. 1-8, Jul 2019.

[14] Y. Zhang, P. Zhou, J. Tang, J. Ma, "Mode selection in a neuron driven by Josephson junction current in presence of magnetic field", *Chinese J. Phys*, Vol. 71, No. 1, pp. 72–84, Jun 2021.

[15] Y. Zhang, Y. Xu, Z. Yao, J. Ma, "A feasible neuron for estimating the magnetic field effect", *Nonlinear Dyn*. Vol. 102, No. 3, pp. 1849–1867, Nov 2020.

[16] A. Uçar, KE. Lonngren, E-W, Bai, "Chaos Synchronization in RCL-Shunted Josephson Junction Via Active Control". *Chaos, Solitons & Fractals*, Vol. 31, No. 1, pp. 105–111, Jan 2007.

[17] AM. Harb, BA. Harb, "Controlling Chaos in Josephson-Junction Using Nonlinear Backstepping Controller". *IEEE Trans Appl Supercond*, Vol. 16, No. 4, pp. 1988-1998, Dec 2006.

[18] UE. Vincent, A. Ucar, JA. Laoye, SO. Kareem, "Control and Synchronization of Chaos in RCL-Shunted Josephson Junction Using Backstepping Design". *Phys C Supercond*, Vol. 485, No. 5, pp. 374-382, Mar 2008.

[19] A.N. Njah, K.S. Ojo, G.A. Adebayo, A.O. Obawole, "Generalized control and synchronization of chaos in RCL-shunted Josephson junction using backstepping design", *Phys. C Supercond*, Vol. 470, No. 13, pp. 558–564, Jul 2010.

[20] AM. Harb, BA. Harb, "Chaos Synchronization in Josephson Junctions". *J Supercond Nov Magn*, Vol. 25, No. 6, pp. 1647-1653, Aug 2012.

[21] D-Y. Chen, W-L. Zhao, X-Y. Ma, R-F. Zhang, "Control and Synchronization of Chaos in RCL-Shunted Josephson Junction with Noise Disturbance Using Only One Controller Term". *Abstr. Appl. Anal.*, Vol 2012, No. 1, pp. 1-15, Jul 2012.

[22] C.-K. Cheng, P.C.-P. Chao, "Trajectory tracking between Josephson junction and classical chaotic system via iterative learning control", *Appl. Sci*. Vol. 8, No. 8, pp. 1285, Aug 2018.

[23] C-K .Cheng, PC-P. Chao, "Chaos Synchronization Between Josephson Junction and Classical Chaotic System via Iterative Learning Control". IEEE Int. Conf. Appl. Syst. Invent, pp. 1232–5, Jun 2018.

[24] TBT. Nguyen, "Adaptive MIMO Controller Design for Chaos Synchronization in Coupled Josephson Junctions via Fuzzy Neural Networks". *J Adv Eng Comput*, Vol. 1, No. 1, pp. 80-86, Dec 2017.

[25] K.S. Ojo, A.N. Njah, O.I. Olusola, M.O. Omeike, "Generalized reduced-order hybrid combination synchronization of three Josephson junctions via backstepping technique", *Nonlinear Dyn*, Vol. 77, No. 3, pp. 583–595, Aug 2014.

[26] E. Camacho, Alba C. *Model Predictive Control*, Springer science & business media, 2013.

[27] M. Ehsani, M. Saeidi, H. Radmanesh, A. Abrishamifar, "Comparisons between Generalized Predictive Control and Linear Controllers in Multi-Input DC-DC Boost Converter", *Int. J. Ind. Electron. Control Optim*. Vol. 3, No. 1, pp. 27–34 , Win 2020.

[28] H. Radmanesh, M. Saeidi, "Linear Modelling of Six Pulse Rectifier and Designee of Model Predictive Controller with Stability Analysis". *Int. J. Ind. Electron. Control Optim*, Vol. 3, No. 4, pp. 491–501, Sum 2020.

[29] M. Heidari, "Maximum Wind Energy Extraction by Using Neural Network Estimation and Predictive Control of Boost Converter", *Int. J. Ind. Electron. Control Optim*. Vol. 1, No. 2 pp. 115–120, Sum 2018.

[30] S. Jalili, B. Rezaie, Z. Rahmani, "A novel hybrid model predictive control design with application to a quadrotor helicopter". *Optim. Control Appl. Methods*, Vol. 39, No. 4, pp. 1301–1322, Jul 2018.

[31] A. Mirzaei, A. Ramezani, "Cooperative distributed constrained model predictive control for uncertain nonlinear large scale systems". *Int. J. Ind. Electron. Control Optim*. Vol. 4, No. 1, pp. 87-98, Jan 2020.

[32] H. Nobahari, S. Nasrollahi, "A Non-Linear Estimation and Model Predictive Control Algorithm Based on Ant Colony Optimization". *Trans Inst Meas Control*, Vol. 41, No. 4, pp. 1123-1138, Feb 2019.

[33] S. Botchu, S. Ungarala, "Nonlinear Model Predictive Control Based on Sequential Monte Carlo State Estimation". *IFAC Proc*, Vol. 40, Jan 2007.

[34] D. Stahl, J. Hauth , "PF-MPC: Particle Filter-Model Predictive Control". *Syst & Control Lett*, Vol. 60, No. 8, pp. 632-643, Aug 2011.

[35] M. Sarailoo, Z. Rahmani, B. Rezaie, "A Novel Model Predictive Control Scheme Based on Bees Algorithm in a Class of Nonlinear Systems: Application to a Three Tank System". *Neurocomputing*, Vol. 152, No. 1, pp. 294-304, Mar 2015.

[36] F. Rajabi, B. Rezaie, Z. Rahmani, "A Novel Nonlinear Model Predictive Control Design Based on a Hybrid Particle Swarm Optimization-Sequential Quadratic Programming Algorithm: Application to an Evaporator System". *Trans Inst Meas Control*, Vol. 38, No. 1, pp. 23-32, Dec 2016.

[37] T. Van Duzer, CW. Turner, *Principles of Superconductive Devices and Circuits*, Edward Arnold, USA, 1981.

[38] D. Simon, *Optimal state estimation: Kalman, H infinity, and nonlinear approaches*; John Wiley & Sons, 2006.

[39] M.S. Arulampalam, S. Maskell, N. Gordon, T. A Clapp, "Tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking " *Signal Process IEEE Trans*, Vol. 50, No. 2, pp. 174–188, Aug 2002.

[40] B. Ristic, S. Arulampalam, N. Gordon, *Beyond the Kalman filter: Particle filters for tracking applications*, Artech house Boston Press, 2004.

[41] J.A. Rossiter, *Model-based predictive control: a practical approach*, CRC press, 2003.

[42] R.Y. Rubinstein, D.P. Kroese, *Simulation and the Monte Carlo method*, John Wiley & Sons press, 2016.

[43] H. Nobahari, Hadi, S. Nasrollahi, "A nonlinear robust model predictive differential game guidance algorithm based on the particle swarm optimization", *J Franklin Inst*, Vol. 357, No. 15, pp. 11042-11071, Aug 2020.

[44] A. Kurniawan, *Getting Started with Matlab Simulink and Arduino*, PE Press, 2013.

Aylar Khooshehmehri received the B.S degree in Electrical Engineering from K. N. Toosi University of Technology, Tehran, Iran, in 2007. Also, she received the M.S. and PhD degree in Electrical Engineering from Malek-Ashtar University of Technology in 2011 and 2018,

respectively. Her research interests include superconducting amplifying and detector devices, MEMS, HRG and semiconductor field effect devices.

Saeed Nasrollahi received the B.S. and M.S. degrees in Electrical Engineering from Malek-Ashtar University of Technology, Tehran, Iran, in 2007 and 2010, respectively. Also he received Ph.D. degree in aerospace engineering from Sharif University of Technology in 2019. He is currently working as an assistant professor at the electrical engineering department at the Malek-Ashtar University of Technology. His research interests are nonlinear control, model predictive control, nonlinear estimation, and their application in guidance and navigation.

Morteza Aliyari was born in Zanjan province, Iran in 1996. He received his bachelor in Electrical Engineering, Control Systems field in 2015. He has hands-on experience in robotic, artificial intelligence and path planning. He was a member of the robotic group which attended at FIRA cup competition and accomplished the third rank of competition. In 2017, he received his master degree in the same field and work as a research assistant in the same university concurrently. His current research interests include robotics, UAVs, Path planning, Navigation, FOG and RLG sensors.