Signature PSO: A novel inertia weight adjustment using fuzzy signature for LQR tuning

Achmad Komarudin[1], Novendra Setyawan[2], Leonardo Kamajaya[3], Mas Nurul Achmadiah[4], Zulfatman[5]
[1,3,4]Department of Electrical Engineering, State Polytechnic of Malang, Indonesia
[2,5]Department of Electrical Engineering, Universitas Muhammadiyah Malang, Indonesia

Article Info

Article history:

Received Mar 24, 2020
Revised May 27, 2020
Accepted Jul 11, 2020

Keywords:

Adaptive adjustment
Fuzzy signature
Inertia weight
Linear quadratic regulator
Particle swarm optimization

ABSTRACT

Particle swarm optimization (PSO) is an optimization algorithm that is simple and reliable to complete optimization. The balance between exploration and exploitation of PSO searching characteristics is maintained by inertia weight. Since this parameter has been introduced, there have been several different strategies to determine the inertia weight during a train of the run. This paper describes the method of adjusting the inertia weights using fuzzy signatures called signature PSO. Some parameters were used as a fuzzy signature variable to represent the particle situation in a run. The implementation to solve the tuning problem of linear quadratic regulator (LQR) control parameters is also presented in this paper. Another weight adjustment strategy is also used as a comparison in performance evaluation using an integral time absolute error (ITAE). Experimental results show that signature PSO was able to give a good approximation to the optimum control parameters of LQR in this case.

Corresponding Author:

Novendra Setyawan,
Department of Electrical Engineering,
Universitas Muhammadiyah Malang,
Jl. Tlogomas No 246, Malang, East Java, Indonesia.
Email: novendra@umm.ac.id

1. INTRODUCTION

Linear state feedback is the simplest way of control scheme to control the multi-output system, including single-input multi-output (SIMO) and multi-input multi-output (MIMO). In a modern optimal control, linear quadratic regulator (LQR) is a state feedback control scheme that finds its applications in engineering due to its inherent stability and robustness. A gain margin at least $(-6, \infty)\ db$ and minimum phase margin $(-60^0, 60^0)$ provided by LQR to enable the multi-output system to reach a satisfactory response even in small perturbations [1, 2]. Generally, the state feedback gains in the LQR method are gained by minimized the quadratic cost function via Algebraic Riccati Equation solution, which consists of weighting metrics, namely Q and R metrics. The performance of LQR is highly dependent on the element Q and R metrics; this is one main issue of LQR in real-time applications. Conventionally, the element of Q and R metrics have been tunned either based on designer experience or via trial and error. However, this approach is not only tedious but time-consuming. Hence, in another method, the LQR design is formulated into an optimization problem and solved ether using the evolutionary or swarm intelligence optimization algorithm such as a particle swarm optimization (PSO) [3-5].

Inspired by natural phenomena, PSO has been widely used to solve optimization problems. PSO was introduced by Kennedy and Eberhart [6] in 1995, which was motivated by the social behavior of individuals such as a group of birds and fish hordes. In solving optimization problems, potential solutions developed in the search space through several iterations. The advantages of the PSO algorithm are its simplicity, and it has several set parameters. The algorithm starts with the random initialization of a population of potential solutions, which are also called particles. Particle populations are also known as swarms, which move through the D-dimensional search space. Several trajectories change the position of a particle with a certain speed based on the experience of the particle and the group. Representation of the position and velocity of each particle is explained in (1).

$$x_i = [x_{i1}, x_{i2}, x_{i3}, \cdots, x_{iD}] \text{ dan } v_i = [v_{i1}, v_{i2}, v_{i3}, \cdots, v_{iD}] \quad (1)$$

where $x_{id} \in [lb, ub]$ is particle position, v_i is particle velocity, $d \in D$, lb is the lower boundary of search with the D-dimension, and ub is the upper limit of the search with the D-dimension. In the PSO, the particle position moves with speed according to the mechanism in (2).

$$\begin{aligned} v_{id}(t+1) &= v_{id}(t) + c_1 \times rand_1 \times \left(p_{id}(t) - x_{id}(t)\right) + c_2 \times rand_2 \times \left(g_d(t) - x_{id}(t)\right) \\ x_{id}(t+1) &= x_{id}(t) + v_{id}(t+1) \end{aligned} \quad (2)$$

where c_1 and c_2 are acceleration constants, $rand_1$ and $rand_2$ are random numbers in $[0,1]$, p_{id} is *personal best* that is the experience of individual particles and g_d is *global best* that is the experience of particle groups.

Balance in local and global search on optimization algorithms through learning in each iteration is essential. Nearly every optimization algorithm has a mechanism to achieve this goal, like mutations and crossing over the genetic algorithm (GA) [7, 8], or the temperature parameters in the simulated annealing (SA) algorithm. As explained in [3], the mechanism of change in particle velocity adds a parameter called inertia (w) so that the particle velocity change is shown in (3).

$$v_{id}(t+1) = w \times v_{id}(t) + c_1 \times rand_1 \times \left(p_{id}(t) - x_{id}(t)\right) + c_2 \times rand_2 \times \left(g_d(t) - x_{id}(t)\right) \quad (3)$$

At first, w is a parameter with a constant value. However, the increase in PSO search performance is less than optimal. As it develops, there are mechanisms for changing the settings of inertia depending on varying conditions that occur in swarm learning. The weight change mechanism aims to regulate the distribution and search capabilities of the PSO algorithm. In [9], they propose the linear decreasing function to changes the parameter. However, better distribution and search are maintained by changing the weights adaptively, as shown in a study carried out by [10].

In some weight changes, as in [11, 12], they use one variable to measure the conditions in the swarm utilized to change the weight of inertia. In [13], it is using a variable in adjusting the inertia weight of PSO that called success count, to measure the successful particle in every iteration. Another approach is described in [14], diversity or dispersion is defined as the distribution of particles when moving in the search space to update the inertia weight in each iteration. However, the use of one variable to describe the swarm condition as the inertia weight update factor is not enough. Thus, in [10, 15-17] using fuzzy logic to combine more thane one variable as a mechanism to update the PSO parameter. However, using the conventional fuzzy as the algorithm to determine the inertia weight is complicated when the feedback parameter is multi-dimensional. The complexity of the algorithm increased as more variables are used.

Meanwhile, [18] proposed a fuzzy signature to simplify the fuzzy algorithm by making the fuzzy inversion into an aggregation equation. The implementation of the fuzzy signature is also have been tested to simplify the robotic control problem in [19, 20]. Hence, this paper used another simple fuzzy approach, which is the fuzzy signature, as the algorithm to adjust the inertia weight its called signature PSO. The testing was done by implementing the proposed PSO algorithm to LQR parameter tunning problem.

2. LINEAR QUADRATIC REGULATOR

LQR is a control strategy that operates the system with the minimum cost when the dynamic of the system is described in linear form. The control performance or the performance index is measured using a quadratic cost function, which consists of the state and control input of the system. After the performance index is expressed, the optimal state feedback control gain is obtained by solving the state-dependent algebraic riccati equation (ARE). The LQR control strategy has been successfully implemented in a complex system such as tracking control of 2 DoF laboratory helicopter [1], double inverted pendulum [4, 21], and also for tracking control of quadrotor [22]. The main reason the LQR is successfully implemented is the inherent robustness and stability properties, such as a gain margin of at least $(-6, \infty)$ and a phase margin of $(-60^0, 60^0)$ degree. Consider a LTI multivariable to be described in the following (4):

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{4}$$

$$y(t) = Cx(t) + Du(t) \tag{5}$$

where $A \in R^{n\times n}, B \in R^{n\times m}, C \in R^{p\times n}$, and $D \in R^{p\times m}$ are the system matrices. Then X is the state vector, Y is the output state vector, and u is the input vector. The conventional LQR is determined the optimal control signal u^* which minimizes this following performance index:

$$J = \int_0^\infty (x^T(t)Qx(t) + u^T(t)Ru(t)) \tag{6}$$

where $Q = Q^T$ is a semidefinite matrix that penalizes the state and $R = R^T$ is also a definite semidefinite matrix that penalizes the control input. In optimal control theory, the optimal control input is obtained as the following state feedback control law:

$$u(t) = -Kx(t) \tag{7}$$

where the optimal control gain K is described in (8).

$$K = R^{-1}B^T P \tag{8}$$

The optimal control gain depends on P matrix which can be obtained by solving this following Algebraic Riccati Equation

$$A^T P + PA + Q - PBR^{-1}B^T P = 0 \tag{9}$$

where $P \in R^{n\times n}$ is the solution of ARE. The element numbers of weighting matrix Q and R are respectively dependent on the number of state-variable and control input. The composition of the element in this weighting matrix has a significant influence to obtain the optimal performance of LQR [23]. Several works [1, 4, 21, 22] have been selected for the diagonal form of weighting matrices, which make the performance index only as a weighted integral square error of the state and the control input. Conventionally, the weighted matrices of LQR tuned manually [4, 21]; thus, it does not have the optimal performance. Therefore, to overcome the weighting matrices selection of LQR, the proposed PSO algorithm was developed to optimally tuned the weighting matrices Q and R.

3. PROPOSED INERTIA WEIGHT ADAPTATION

3.1. Fuzzy signature weight adaptation

Fuzzy signature is a simple multidimensional fuzzy which is introduced by Koczy [18]. Fuzzy signature has been implemented in the various decision problems, such as controlling mobile robot [19], robot cooperation [20], and data mining [24]. A fuzzy signature is a generalized form of vector fuzzy. The fuzzy signature can be represented as the vector in (10) or a tree structure such as Figure 1.

$$x = \begin{bmatrix} \begin{bmatrix} x_{11} \\ x_{12} \end{bmatrix} \\ \begin{bmatrix} x_{21} \\ \begin{bmatrix} x_{221} \\ x_{222} \\ x_{223} \end{bmatrix} \\ x_{23} \end{bmatrix} \\ \begin{bmatrix} x_{31} \\ x_{32} \end{bmatrix} \end{bmatrix} \tag{10}$$

In (10), $[x_{11} \quad x_{12}]$ is a sub-group from a higher-level structure which is x_1. The $[x_{221} \quad x_{222} \quad x_{223}]$ was combined into x_{22}. The sub-group or branch was connected until the higher level which is $x = [x_1 \quad x_2 \quad x_3]$. The combination of the sub-group was done using some aggregation functions such as max, min, and mean functions. Consider that a_1 is the aggregation function that was resulting x_1; hence, $x_1 = x_{11}a_1x_{12}$. In the sub-group, aggregation function could be identical or different from one another. As described in the implementation of a fuzzy signature in [19], the simple aggregation function that often used is the $min, max,$ and $mean$ function.

Inspired with the multidimensional and straightforward fuzzy signature, the parameter feedback success count in [13], diversity [14] and current best performance evaluation (CBPE) [10] were combined using a fuzzy signature as a measurement of the condition of swarm and performance evaluation of the most recent PSO algorithm. The structure of the fuzzy signature inertia adaptation function in the PSO algorithm or signature PSO is described in Figure 2 and was obtained using (11).

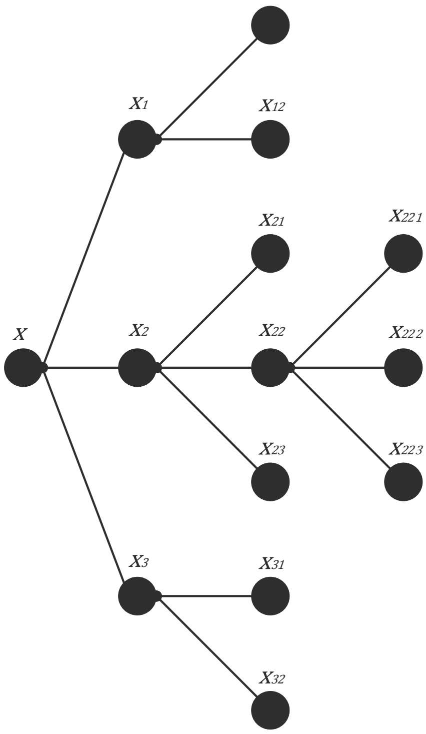

Figure 1. The tree structure of the fuzzy signature

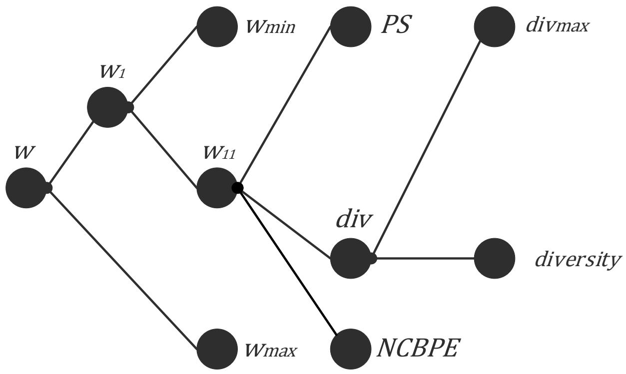

Figure 2. The structure of the fuzzy signature as inertia weight adaptation function

In this structure, three aggregation functions were used, which are max, min, and mean. The inertia weight w is the higher level of the structure, which was adapted according to the value of the feedback parameter. The adaptation of w was constrained with div_{max}, w_{min}, and w_{max} where the value of div_{max} is 1, w_{min} is 0.4, and w_{max} is 0.9 according to [11, 14, 25].

$$w = \begin{bmatrix} w_{max} \\ \begin{bmatrix} w_{min} \\ \begin{bmatrix} PS \\ \begin{bmatrix} diversity \\ div_{max} \end{bmatrix} \\ NCBPE \end{bmatrix} \end{bmatrix} \end{bmatrix} \quad (11)$$

In (11), the successful count of particles is obtained in the following function [13]:

$$SC_i(t) = \begin{cases} 1, & f(x_{id}(t)) < f(P_{id}(t)) \\ 0, & else \end{cases} \quad (12)$$

where SC_i is the count of particle which has the best position to minimize the objective function. Hence, the percentage of success ($\boldsymbol{PS}$) is calculated using (13).

$$PS(t) = \frac{1}{N}\sum_{i=1}^{N} SC_i\,(t) \quad (13)$$

The diversity variable is described as the distribution of particles when moving in the search space, as shown in (14) [14]:

$$diversity = \frac{1}{N}\sum_{i=1}^{N}\sqrt{\sum_{d=1}^{D}(x_{id} - g_d)^2} \quad (14)$$

The last variable used in signature PSO to update the inertia weight is CBPE that measures the best fitness value by the most recent best candidate solution. As parameter feedback, CBPE is normalized to the following (16).

$$NCBPE = \frac{CBPE - CBPE_{min}}{CBPE_{max} - CBPE_{min}} \quad (15)$$

The example of the aggregating process in the structure of Figure 2 or (11) is described as the following example with initial variable value and using min aggregation function, as shown in (16).

$$w = \begin{bmatrix} 1 \\ \begin{bmatrix} 0,4 \\ \begin{bmatrix} 0,4 \\ \begin{bmatrix} 0,5 \\ 1 \end{bmatrix} \end{bmatrix} \\ 0.3 \end{bmatrix} \end{bmatrix} \quad (16)$$

The lower level aggregation function was obtained between $diversity$ and div_{max} using $mean$ function as shown in (17)

$$w = \begin{bmatrix} 1 \\ \begin{bmatrix} 0,4 \\ \begin{bmatrix} 0,4 \\ 0,5 \\ 0.45 \end{bmatrix} \end{bmatrix} \end{bmatrix} \rightarrow \begin{bmatrix} 1 \\ \begin{bmatrix} 0,4 \\ 0,45 \end{bmatrix} \end{bmatrix} \quad (17)$$

In the higher level, aggregation was obtained using the max and min function as the following equation

$$w = \begin{bmatrix} 1 \\ \begin{bmatrix} 0,4 \\ 0,45 \end{bmatrix} \end{bmatrix} \rightarrow \begin{bmatrix} 1 \\ 0,45 \end{bmatrix} = 0,45 \quad (18)$$

Thus, the inertia weight w from the initial condition of (16) was $0,45$.

3.2. The implementation of signature PSO for LQR tunning

The objective of signature PSO implementation is to determine the element of state weighting Q on LQR, which is a $R^{n\times n}$ positive semidefinite matrix and the input weighting matrix R, which the size of it depends on the number of input variables. To simplify the implementation of optimization problem the weighting matrices of Q and R are chosen as diagonal matrices that can be described in equation bellow

$$Q = I_{n\times n}[q_1 \quad q_2 \quad \cdots \quad q_n]^T \text{ and } R = I_{s\times s}[r_1 \quad r_2 \quad \cdots \quad r_s]^T \quad (19)$$

where n and s is the number of the state variable and input variable based on the LTI system in (4). Therefore, to determine of element $[q_1 \quad q_2 \quad \cdots \quad q_n]^T$ and $[r_1 \quad r_2 \quad \cdots \quad r_s]^T$ the particle form in (1) become the following (20).

$$x_i = [x_{i1}, x_{i2}, x_{i3}, \cdots, x_{iD}] = [q_{i1}, q_{i2}, \cdots, q_{in}, r_{i1}, r_{i2}, \cdots, r_{is}] \quad (20)$$

With those initial values of the particles, signature PSO computes the corresponding global best of the particles, which is obtained by minimizing the fitness function. The fitness function characterizes the convergence of the optimization algorithms towards the comprehensive solution. Some of the commonly used fitness functions in the state feedback controller design are integral of the absolute error (IAE), integral of the square error (ISE), and integral of the time-weighted absolute error (ITAE). In the present study, the following ITAE is chosen as the fitness evaluation function:

$$f(x_{id}) = \int_0^{\infty} t|e(t)| \quad (21)$$

where $e(t) = x_r - x$, with x_r is the state reference of the state variable. Thus, the completely LQR tunning using the signature PSO algorithm is described with pseudo-code in the following Figure 3.

```
Start
Initialize c1,c2, and lower and upper bound of search space;
Initialize w_max, w_min, div_max, and w = 1;
For i=1 to number of particles
Initialize particle as equation (20) ;
P_i = x_i ;
End
While (termination condition is false)
   SC=0; diversity = 0; NCBPE = 0;
    For i=1 to number of particle
      For d=1 to D
       Update velocity and position of particle using equation.(2);
       Evaluate particle with objective function using equation (21);
       End
    End
 Update P_id and G_d;
 Compute PS, diversity, and NCBPE using (13)(14) and (15);
 Aggregate diversity and div_max using min function resulting a1;
 Aggregate PS, NCBPE, and a1 using mean function resulting a2;
 Aggregate a2 and w_min using max function;
 Update Inertia parameter using min function aggregation;
End
Optimal solution = G_d ;
End
```

Figure 3. Pseudocode of signature PSO for LQR tunning

4. RESULTS AND DISCUSSION

In this section, the proposed algorithm was tested by comparing the PSO signature with Adaptive Inertia Weight or AIW PSO [13], which is the optimal PSO development in solving optimization problems. Moreover, it was also compared to PSO with constant inertia parameters or PSO standards [11]. The LTI system parameter in (4) that used to test the algorithm is the double inverted pendulum system [4], which has six variable states and one input variable that described bellow.

$$A = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 7.11 & 0.77 & 0 & 0 & 0 \\ 0 & 36.20 & -15.70 & 0 & 0 & 0 \\ 0 & -29.08 & 24.92 & 0 & 0 & 0 \end{bmatrix}; \; B = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0.45 \\ -0.56 \\ 0.11 \end{bmatrix}; \qquad (22)$$

$$C = I_{6\times 6}; D = 0 \qquad (23)$$

As the number of the optimized variable is seven parameters, which six state and one input variable in LQR design, so seven dimensions of the particle are chosen. The test was carried out with the same parameters in all algorithm which the number particle is 20 and the c_1 and c_2 is set into 2. In the conventional or standard PSO, the inertia weight is set as 0.8 constant value, whereas in AIW and signature PSO, the inertia weight is modified adaptively with different strategies. The inertia weight adaptation strategies of signature PSO is adjusted according to (11) with some aggregation function that described in pseudocode in Figure 3.

The test was carried out ten times, with 100 iterations at one run to find the solution parameters. Experiments were carried out with the same plant conditions using a computer with i5 processor and 4 Gb ram, where the results are displayed in Table 1 with I is the identity matrix with a size of 6 × 6. The test results show that although the maximum ITAE value of AIW PSO was smaller than the signature PSO, in the ten tests, the average run of ITAE of the signature PSO was 7% lower than other PSOs. An exciting factor which can be noted from Table 1, which gives the statistical parameters of both signature and AIW PSO algorithms, is that both the standard deviation and minimum value of fitness function optimized using signature PSO are less than that of the AIW and conventional PSO, which proves that the signature PSO can result in better consistent performance. In addition, the average ITAE minimum value obtained by the PSO signature was also smaller even though the significant level of change was only 0.01%. However, in terms of robustness in the signature, the PSO algorithm was better than others as indicated by a 30% smaller standard deviation.

Table 1. Results of ten algorithm tests

		PSO signature	PSO AIW	PSO Standard (Constant Inertia)
Max	Solution	$Q = I \times \begin{bmatrix} 1000 \\ 1000 \\ 934.071 \\ 0.0001 \\ 132.776 \\ 159.530 \end{bmatrix}$ $R = 0.0001$	$Q = I \times \begin{bmatrix} 1000 \\ 0.0001 \\ 1000 \\ 53.949 \\ 0.0001 \\ 0.0001 \end{bmatrix}$ $R = 0.0001$	$Q = I \times \begin{bmatrix} 1000 \\ 0.0001 \\ 0.0001 \\ 62.500 \\ 0.0001 \\ 0.0001 \end{bmatrix}$ $R = 0.0001$
	ITAE	115.2376671288983	**114.4096521684920**	114.4795653046360
Mean	Solution	$Q = I \times \begin{bmatrix} 473.333 \\ 993.209 \\ 122.818 \\ 14.2667 \\ 13.2777 \\ 168.139 \end{bmatrix}$ $R = 0.0001$	$Q = I \times \begin{bmatrix} 737.192 \\ 800 \\ 329.367 \\ 35.005 \\ 0.03312 \\ 249.437 \end{bmatrix}$ $R = 0.0001$	$Q = I \times \begin{bmatrix} 815.329 \\ 425.889 \\ 300 \\ 43.438 \\ 0.0001 \\ 521.733 \end{bmatrix}$ $R = 0.0001$
	ITAE	**112.4207734642174**	113.2260563066121	113.6314747140301
Min	Solution	$Q = I \times \begin{bmatrix} 122.425 \\ 1000 \\ 0.0001 \\ 0.499 \\ 0.0001 \\ 0.0001 \end{bmatrix}$ $R = 0.0001$	$Q = I \times \begin{bmatrix} 125.0103 \\ 1000 \\ 0.0001 \\ 0.5129 \\ 0.0001 \\ 0.0001 \end{bmatrix}$ $R = 0.0001$	$Q = I \times \begin{bmatrix} 121.931 \\ 1000 \\ 0.0001 \\ 0.497 \\ 0.0001 \\ 0.0001 \end{bmatrix}$ $R = 0.0001$
	ITAE	**111.1641977917066**	111.1642385127179	114.4795653046360
Std		**164.194098719274**	249.8872660913316	258.1420283312436

Based on the ITAE fitness function, the optimization algorithms are executed for the specified number of iterations, and the global best of the particles, which are the weights of LQR, are obtained. The Fitness evaluation value in every iteration of signature, AIW, and standard PSO is depicted in Figure 4. It can be noted that the fitness function of signature and AIW PSO converges faster than that of the conventional PSO. However, signature PSO slightly faster ten iterations than AIW PSO, which can found a solution in just 28 iterations. These data substantiate that the introduction of a fuzzy signature as inertia weight update strategies in the velocity update equation of PSO significantly improves the convergence speed and accuracy of the algorithm.

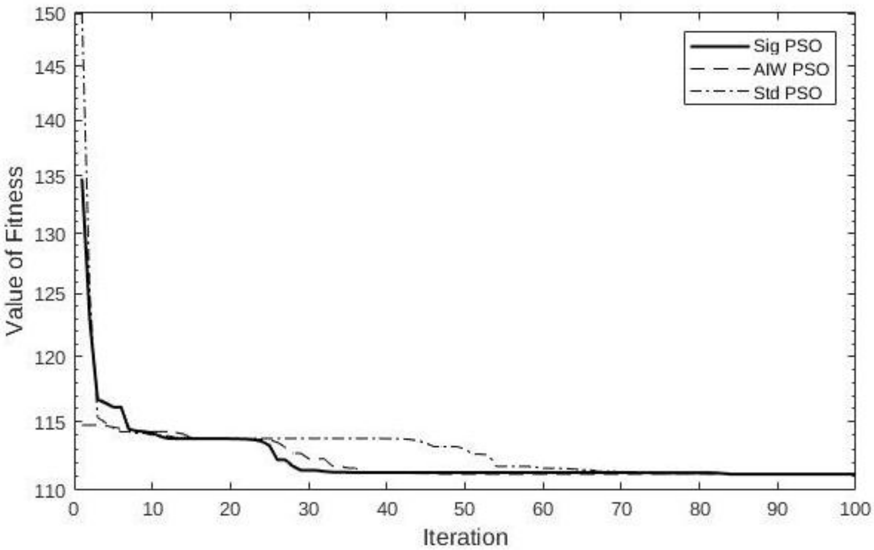

Figure 4. Fitness evaluation value in every iteration of signature, AIW, and standard PSO

In these tests, the initial condition of the third state of the system which is the upper pendulum angle was set into 0.08 radians with another state 0, so the state was $x = [0 \quad 0 \quad 0.08 \quad 0 \quad 0 \quad 0]$. In the optimum results, each algorithm is shown in Figure 5, Figure 6, and Figure 7. Figure 5 illustrated the train position, which is the first state variable of the system. Despite having a higher peak first seconds of simulation, the results obtained by Signature PSO can return to settle conditions with a faster time at 2.2 seconds or 0.5 seconds faster than the results of AIW PSO solution other optimizations algorithms.

In the lower pendulum angle, despite has the same settling time and peak amplitude, which is 2.5 seconds and 0.2 radians with other PSO algorithms, the excellent tunned solution from signature PSO can reduce the oscillation in 0.5 seconds, which depicted in Figure 6. Figure 7 illustrated the third state variable of the pendulum system, which is the upper pendulum angle. It has the same settling time with other algorithms; the signature PSO tunned the LQR weight that can reduce the overshoot of the upper pendulum angle.

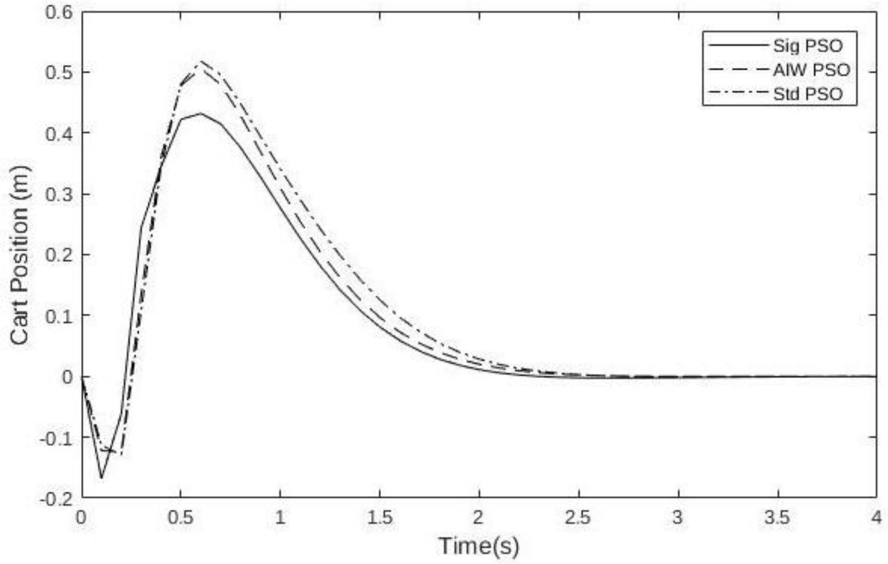

Figure 5. Comparison of train position responses results

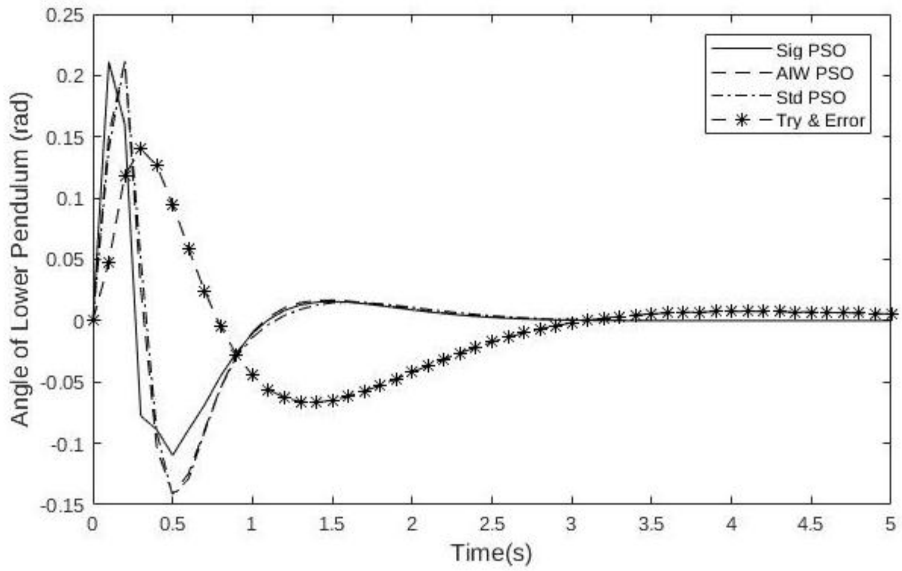

Figure 6. Comparison of the lower pendulum angle response results

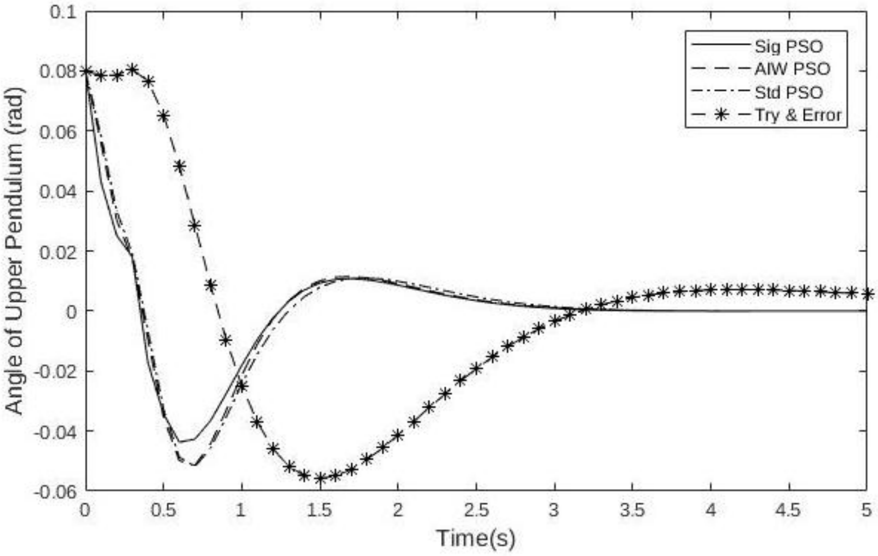

Figure 7. Comparison of the upper pendulum angle response results

5. CONCLUSION

In this paper, we proposed the algorithm that changes the inertia weights on PSO using fuzzy signatures. Using a fuzzy signature, it can accommodate many parameters that describe the swarm conditions in PSO. In this paper, the parameters proposed were PS, diversity, div_{max} and w_{min}. By using a fuzzy signature, although it had many parameters for weight changes, it had a simpler algorithm compared to fuzzy in general. Based on the proposed algorithm and testing that had been done on the optimization problem of LQR control, the signature PSO algorithm has a faster, optimal, and robust performance.

REFERENCES

[1] E. Vinodh Kumar, G. S. Raaja, and J. Jerome, “Adaptive PSO for optimal LQR tracking control of 2 DoF laboratory helicopter,” *Applied Soft Computing*, vol. 41, pp. 77–90, 2016.

[2] S. Karthick, J. Jerome, E. V. Kumar, and G. Raaja, “APSO based weighting matrices selection of LQR applied to tracking control of SIMO system,” in *Proceedings of 3rd International Conference on Advanced Computing, Networking and Informatics*, Springer, New Delhi, pp. 11–20, 2016.

[3] A. Aula, S. Ahmad and R. Akmeliawati, "PSO-based state feedback regulator for stabilizing a two-wheeled wheelchair in balancing mode," *2015 10th Asian Control Conference (ASCC)*, Kota Kinabalu, pp. 1-5, 2015.

[4] S. Khatoon, D. K. Chaturvedi, N. Hasan and M. Istiyaque, "Optimal control of a double inverted pendulum by linearization technique," *2017 International Conference on Multimedia, Signal Processing and Communication Technologies (IMPACT)*, Aligarh, pp. 123-127, 2017.

[5] M. Assahubulkahfi, Y. Md. Sam, A. Maseleno, and M. Huda, “LQR Tuning by Particle Swarm Optimization of Full Car Suspension System,” *International Journal of Engineering & Technology (UAE)*, vol. 7, no. 2.13, pp. 328-331, 2018.

[6] R. Eberhart and J. Kennedy, “A new optimizer using particle swarm theory,” *MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, pp. 39–43, 1995.

[7] F. Yu and X. Xu, “A short-term load forecasting model of natural gas based on optimized genetic algorithm and improved BP neural network,” *Applied Energy*, vol. 134, pp. 102–113, 2014.

[8] N. H. Trưởng and D.-N. Dao, “New hybrid between NSGA-III with multi-objective particle swarm optimization to multi-objective robust optimization design for Powertrain mount system of electric vehicles,” *Advances in Mechanical Engineering*, vol. 12, no. 2, pp. 1-14, 2020.

[9] A. Ratnaweera, S. K. Halgamuge and H. C. Watson, "Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients," in *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 3, pp. 240-255, June 2004.

[10] Yuhui Shi and R. C. Eberhart, "Fuzzy adaptive particle swarm optimization," *Proceedings of the 2001 Congress on Evolutionary Computation (IEEE Cat. No.01TH8546)*, Seoul, South Korea, vol. 1, pp. 101-106, 2001.

[11] Y. Shi and R. C. Eberhart, "Empirical study of particle swarm optimization," *Proceedings of the 1999 Congress on Evolutionary Computation-CEC99 (Cat. No. 99TH8406)*, Washington, DC, USA, vol. 3, pp. 1945-1950, 1999.

[12] J. C. Bansal, P. K. Singh, M. Saraswat, A. Verma, S. S. Jadon and A. Abraham, "Inertia Weight strategies in Particle Swarm Optimization," *2011 Third World Congress on Nature and Biologically Inspired Computing*, Salamanca, pp. 633-640, 2011.

[13] A. Nickabadi, M. M. Ebadzadeh, and R. Safabakhsh, "A novel particle swarm optimization algorithm with adaptive inertia weight," *Applied soft computing*, vol. 11, no. 4, pp. 3658–3670, 2011.
[14] F. Olivas, F. Valdez, O. Castillo, and P. Melin, "Dynamic parameter adaptation in particle swarm optimization using interval type-2 fuzzy logic," *Soft Computing*, vol. 20, no. 3, pp. 1057–1070, 2016.
[15] A. M. Abdelbar, S. Abdelshahid and D. C. Wunsch, "Fuzzy PSO: a generalization of particle swarm optimization," *Proceedings. 2005 IEEE International Joint Conference on Neural Networks*, Montreal, Que., vol. 2, pp. 1086-1091, 2005.
[16] Hongbo Liu and A. Abraham, "Fuzzy adaptive turbulent particle swarm optimization," *Fifth International Conference on Hybrid Intelligent Systems (HIS'05)*, Rio de Janeiro, Brazil, p. 6, 2005.
[17] M. S. Nobile, P. Cazzaniga, D. Besozzi, R. Colombo, G. Mauri, and G. Pasi, "Fuzzy Self-Tuning PSO: A settings-free algorithm for global optimization," *Swarm and evolutionary computation*, vol. 39, pp. 70–85, 2018.
[18] L. T. Kóczy, T. Vámos, and G. Biró, "Fuzzy signatures," *Proc. Eurofuse-SIC*, vol. 99, pp. 25–28, 1999.
[19] A. Ballagi and L. T. Koczy, "Fuzzy Signature Based Mobil Robot Motion Control System," *2008 6th International Symposium on Applied Machine Intelligence and Informatics*, Herlany, pp. 29-33, 2008.
[20] Á. Ballagi, L. T. Kóczy, and T. Gedeon, "Robot Cooperation without Explicit Communication by Fuzzy Signatures and Decision Trees.," in *IFSA/EUSFLAT Conference*, pp. 1468–1473, 2009.
[21] R. Banerjee, N. Dey, U. Mondal and B. Hazra, "Stabilization of Double Link Inverted Pendulum Using LQR," *2018 International Conference on Current Trends towards Converging Technologies (ICCTCT)*, Coimbatore, pp. 1-6, 2018.
[22] T. Agustinah, F. Isdaryani, and M. Nuh, "Tracking Control of Quadrotor Using Static Output Feedback with Modified Command-Generator Tracker," *International Review of Automatic Control (I. RE. A. CO.)*, vol. 9, no. 4, 2016.
[23] Desineni Subbaram Naidu, "Optimal control," vol. 400, CRC Press, 2010.
[24] . Vamos, L. T. Koczy and G. Biro, "Fuzzy signatures in data mining," *Proceedings Joint 9th IFSA World Congress and 20th NAFIPS International Conference (Cat. No. 01TH8569)*, Vancouver, BC, Canada, vol. 5, pp. 2842-2846, 2001.
[25] Y. Shi and R. Eberhart, "A modified particle swarm optimizer," *1998 IEEE International Conference on Evolutionary Computation Proceedings. IEEE World Congress on Computational Intelligence (Cat. No.98TH8360)*, Anchorage, AK, USA, pp. 69-73, 1998.

BIOGRAPHIES OF AUTHORS

Achmad Komarudin received Bachelor Engineer in Electrical Engineering at University of Brawijaya Malang, Indonesia on 1985. He has obtained his Master of Management Of Electrical Technology from Faculty of Science and Technology of University of Technology, Surabaya, Indonesia on 2009. Currently he is working as Senior Lecturer in Department of Electrical Enginering, State Polytechnic of Malang. His interested research is about Control System, Microcontroler, Electronics, and Electronic System Design.

Novendra Setyawan completed Bachelor Engineer in Electrical Engineering at University of Muhammadiyah Malang, Indonesia on 2015. He has obtained his master degree in Control System from Departement of Electrical Engineering of Institute Teknologi Sepuluh Nopember, Surabaya, Indonesia on 2017. He is currently a junior lecturer in Department of Electrical Enginering of UMM. His area of research is about optimization method, Artificial Intelligent, Robotics and Control System.

Leonardo Kamajaya did Bachelor of Applied Science in Electronic Engineering, Electronic Engineering Polytechnic Institute of Surabaya, EEPIS, Indonesia on 2009. He received his Master of Science in Graduate Institute of Electronic Engineering from Chang Gung University, Taiwan on 2016. Currently he is working as junior lecturer in Department of Electrical Enginering, State Polytechnic of Malang. His interested research is about Biosensor, Renewable Energy, Control System and Embedded System.

Mas Nurul Achmadiah completed Bachelor of Applied Science in Electronic Engineering, Electronic Engineering Polytechnic Institute of Surabaya, EEPIS, Indonesia on 2009. She received his Master degree in Control System from Departement of Electrical Engineering of Institute Teknologi Sepuluh Nopember, Surabaya, Indonesia on 2017. Currently he is working as junior lecturer in Department of Electrical Enginering, State Polytechnic of Malang. His interested research is about Robotics, Control System and Embedded System.

Zulfatman completed B. Eng. (Electrical) from University of Muhammadiyah Malang in 2003. He received the Master degree and a PhD degree in Control System Engineering from Universiti Teknologi in 2009 and 2015. He is currently a Senior Lecturer in the Department of Electrical Engineering, Faculty of Engineering, University of Muhammadiyah Malang, Indonesia. His area of research includes control systems for renewable energy and mechatronics.