Tuning PID using Particle Swarm Optimization For Controlling Temperature of The Infant-Incubator

Jihed El Hadj Ali †, Elyes Feki † and Abdelkader Mami †,

Faculty of Mathematical, Physical and Natural Science of Tunis,
Application Laboratory, Energy Efficiency and Renewable Energy (LAPER),
El Manar University, 2092, Tunisia.

Summary
To guarantee the survival of each child since his birth, especially the preterm babies, we must focus on improving care around the specific newborn, this subject is very important and has big attention for the biomedical company. However, in the developing countries the expensive and high price of those devices (intensive unit care), creates a big challenge to ameliorate Due to suffering from the performance of the typical PID controller in the commercial infant incubator. The Particle Swarm Optimization (PSO) appears as a successful optimization tool. The main research of this paper is to investigate the use of the Particle Swarm Optimization techniques for tuning the gains of the PID of the heater inside the infant-incubator to minimizing the temperature inside the Care Unit. To achieve the performance of this model. Several Computer simulations and experimental results prove that the performance of the optimal PID using the PSO controller gives a superior performance than that of the traditional design methods of the conventional PID controller. For establishing the optimal PID controller the use of the four performances index (IAE, ISE, ITAE, and ITSE) as the objective function.

Key words:
Infant-incubator, Particle Swarm Optimization, PSO, PID-Control, Temperature.

1. Introduction

Until our days, the challenges in engineering have constantly viewed as a major inspiration to take care of the issues of augmenting gains or minimizing the losses. Besides, the obscurity of the optimization problems has been growing with the technology's progress. For the purpose to solve this problem our paper focuses on the use of the PSO approach to control temperature air inside the infant incubator.
In 1995, James Kennedy a social Psychologist and Russell Eberhart an Electrical engineer developed a new evolutionary computational technique known as Particle Swarm Optimization (PSO) for solving continuous and discrete optimization problems [1].

The approach is inspired by the work of Heppner and Grenander which they studied natures flocks of birds, schools of fish and swarms of insects [2, 3]. These ideas were developed into the Particle Swarm Optimizer. Since 1995, this approach witnessed the development of many applications and variants [4, 5].
A basic variant of the PSO algorithm works by having a population (called a swarm) of candidate solution (called particles). These particles are moved around in the search space according to a few simple formulae. The movements of the particles are guided by their own best-known position in the search-space as well as the entire swarm's best-known position. When improved positions are being discovered, these will then come to guide the movements of the swarm. The process is repeated and by doing so it is hoped, but not guaranteed, that a satisfactory solution will eventually be discovered. Here in this technique, a set of particles is put in a d-dimensional search space with a randomly chosen velocity and position. The initial position of the particle is taken as the best position for the start and then the velocity of the particle is updated based on the experience of other particles of the swarming population.
It should be noted that the majority of neonatal incubators is controlled by a PID controller. In what follows, we will describe the different algorithms used in this process.
As indicated by the World Health Organization (WHO), every year, 2.6 million babies pass away on in the initial 28 days of life. For the most part of them was dying in the first week of their life [6].
Numerous scholars have reported the infant incubator, but always they have bordered their research only on the mathematical model starting [7] in 1996 and it continued work on the same topic with [8, 9]. Then, with the study of [10] Gustavo H. C. Oliveira, the appearing of the use of the predictive control and there were several studies since 2010 to 2019 have interested about the control and the

Manuscript received March 5, 2020
Manuscript revised March 20, 2020

building design of the infant incubator [11, 16] with the use of different controller Proportional Integrator Derivative (PID), Model Predictive Control (MPC), Generalized Predictive Control (GPC), Dynamic Matrix Control (DMC), etc ….

There are a few studies investigated the implementation of optimal control. The fuzzy logic method is used [17] to optimize the control of temperature.

The aim of this research is to design a PID Controller using PSO algorithm. The model of a neonatal incubator is used as a plant in this paper. The conventional gain tuning of PID Controller (such as Ziegler-Nichols (ZN) method) usually produces a big overshoot, and therefore modern heuristic approach such as a particle swarm optimization (PSO) algorithm are employed to enhance the capability of traditional techniques.

The outlines of this paper start with a brief introduction; Then, Section 2 presents the methods and materials. The scheme of the PID controller is described in Section 3. Section 4, it highlights the Particle Swarm Optimization. Then, Section 5 is dedicated to simulation results and discussion. Finally, the concluding remarks and future works are presented in Section 6.

2. Methods and materials

The progressive development concerning the servo-control of the infant incubators has growing requirements for a precise evaluation of the potential if those incubators reach the required thermal regulation of the newborn babies. The basis thermo-regulation in the infant-incubator depends on numerous factors regarding the incubator likewise the newborn babies. In order to guarantee the temperature of the body within the normal range, this varies between 36.5° C and 37.5° C.

First, the incubator related factors contain temperature, airflow rates, wall thickness, material type, and humidity level. Secondly, the parameters for the infants which affect the thermo-regulation encompass the physical composition, the volume, the skin thickness, the ratio of surface area, metabolic rate, the maturity level, and the size. A mathematical model for infant incubator is used [15]. It's based on the law of mass and conservation of the heat for the purpose to achieve the physical model.

Figure 1 represents the complete infant incubator system that will be subdivided into six homogeneous compartments; the neonate core, skin, incubator air space, heater, wall, and mattress.

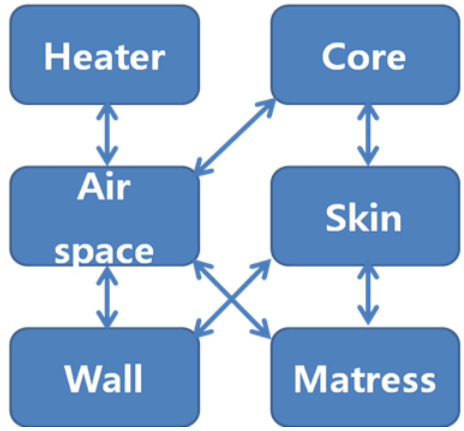

Fig. 1 The six different compartments of the neonate system.

According to the literature, it is obvious that there are abundant studies and projects that evaluated the parameters related to the incubator and the influence on the thermo-regulation inside the incubator. For any extra information is detailed in previous papers about the model or the identification [12, 13, and 14].

2.1 The temperature control system

In our study, we pay interest only in the optimization of the heating system of the neonate incubator. This system is Single-Input Single-Output (SISO) which has as input the heating power and as output the temperature. The neonatal incubator DRAGER 8000 C from Maternal and Neonatal Unit of Rabta-Tunisia is used [15, 16].

Generally, the common point of the infant incubator is highlighted in their similar performing. The presence of a fan has a purpose to turn up a filtered ambient air above the heating device and a water holder for the humidity [14, 15, 16, 17], but in our study, we dried this container to just controlled the temperature [15] it is illustrated in Figure 2.

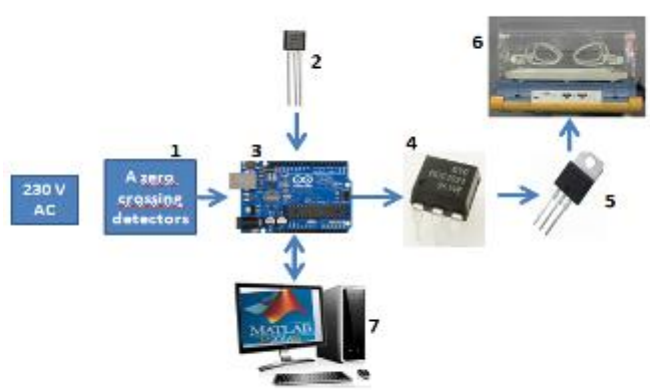

Fig. 2 The schematic of the Arduino connected to the infant incubator.

In Table 1 shows the various components and their function were delineated in Figure 2.

Table 1: The description of various part of the system.

Number	*Components*	Description
1	A zero crossing detector	Used to generate a synchronous pulse related to the AC voltage phase angle.
2	A sensor LM35	To measure the interior temperature of the incubator.
3	Arduino board:	- An acquisition card: to convert conditioned sensor signals to digital values - A control card: sends the command each time zero Is detected.
4	An optocoupler MOC3021	Placed in between the microcontroller and TRIAC to isolate the high voltage side of loads and the low voltage side of the Arduino microcontroller.
5	A Triac	To control the power delivered to AC loads. To isolate the high voltage side of loads and low voltage side of the Arduino microcontroller.
6	An incubator DRAGER 8000C	This is the system to identify and control.
7	A computer (with MATLAB software)	For the generation of control signals and recording of measurements.

The study of the acquisition system of the temperature has divided into two different sections which are the software and hardware. The first section is the hardware which is designed by the incubator process, the Arduino platform and the LM35 sensor. Secondly, concerning the software is depicted by the Arduino firmware and connected with Matlab 2016b environment. The Figure 2 illustrates the distinct part of the system. This device is set up in appropriate conditions to achieve good performance. So it is assigned in a room with constant temperature represented by 27 °C, besides this room is protected from the solar radiation to be certain and avoiding any disagreeable effect.

3. PID control scheme

Since 1910, the PID controller has been widely used in many fields [18, 19]. There are many diversities of the PID controller, but the most common PID controller is presented in Figure 3, is the feedback-loop with a single input and a single output:

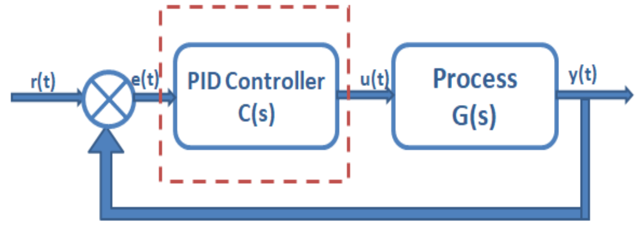

Fig. 3 The feedback control system.

A PID regulator is acquired by the combination of these three actions (proportional, integral, derivative) and it basically satisfies the accompanying three capacities:

- It gives a control signal considering the advancement of the yield signal contrasted with the set point.
- It wipes static error due to the term integrator.
- It foresees the variations of the exit thanks to the term derivative.

The ordinary implementation of the PID controller (Parallel Structure) is demonstrated in Figure 4. Starting with e(t) which is the signal attack the block of PID control and the obtained excitation signal is the total of the error signal damaged by the proportional, integral and derivative actions.

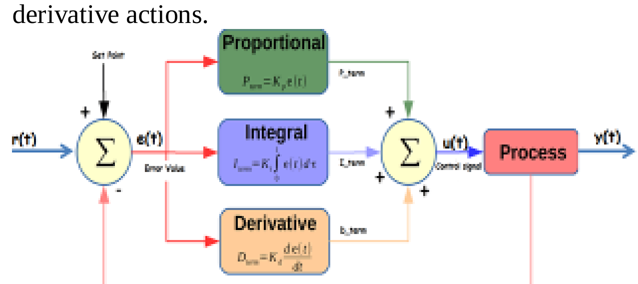

Fig. 4 The block diagram of PID control structure.

The PID controller target is to minimize the error between a measured process variable of the controlled system and a reference (set-point), by computing the error and engender an adjustment signal to the system from the error. The block diagram of a typical PID controller is exhibited in the Figure 4, with r(t) is the reference value, y(t) is the output of the controlled system, e(t) is the error between r(t) and y(t), whereas u(t) is the output control signal of the PID controller.

A regular PID controller includes three different pieces: the proportional item, the integral item and the derivative item as shown in the previous figure. Initially, the proportional term provides an output variable which is

proportional to the actual value of the error. The beneficence of an integral part is proportional to the magnitude of the error and its duration. Lastly, the derivative control is applied to restrict the magnitude of the overshoot provided by an integral part and enhance the mixed controller-process stability.
The output control signal of a conventional PID controller is expounded as follows [19, 20, 21]:

$$u(t) = k_p e(t) + k_i \int_0^t e(t)dt + k_d \frac{de(t)}{dt} \quad (1)$$

Where K_p, K_i and K_d being referred to the proportional gain, the integral gain and the derivative gain accordingly. In addition, we can reformulate the equation (1) in Laplace form in the following term:

$$u(s) = e(t)\left(k_p + \frac{k_i}{s} + k_d s\right) \quad (2)$$

The transfer function of the PID controller or the control law is considering as follow:

$$c(s) = \frac{u(s)}{e(s)} = \left(k_p + \frac{k_i}{s} + k_d s\right) \quad (3)$$

Furthermore, the performance criteria are determined as a measurement of the system error to illustrate the system performance of the developed PID controller. The use of this technique an ‘optimum system’ can generally be designed and a set of PID parameters in the system can be adjusted to meet the required specification. In general, the PID control system is composed of four indices to represent the performance of the system.
Starting with the most popular one is the Integral of the Absolute Error (IAE) which is written as follows [19]:

$$IAE = \int_0^\infty |e(t)| dt \quad (4)$$

The second method is the integral of the square error (ISE) which is expressed as follow;

$$ISE = \int_0^\infty e^2(t).dt \quad (5)$$

The third method is Integral time absolute error (ITAE) which is despite as follow:

$$ITAE = \int_0^\infty t.|e(t)| dt \quad (6)$$

Finally, the method of integral time of the square error (ITSE) as follow:

$$ITSE = \int_0^\infty t.e^2(t).dt \quad (7)$$

Where e(t) is the difference or deviation (error) between the response and the desired set point.
Furthermore, for the PSO-based tuning PID controller, the obvious index performance will be used as the objective function. In other terms, the objective in the PSO-based optimization is to find a set of PID parameters such that the feedback control system has a minimum performance index [20].

4. Particle Swarm Optimization PSO-algorithm

At the beginning of the 1990s, numerous studies and works concerning the social comportment of animal groups were developed. These researches and studies demonstrated that certain animals under their groups (or swarm) taken, for example, birds or fishes are might be able to exchange information amid their group, and that kind of skill provides for them a significant advantage to survive [1]. These studies inspired Kennedy and Eberhart [2] suggested the PSO algorithm in 1995. The PSO algorithm, was simulated animal social behavior, involving birds, insects, herds, bees, and fishes. In the PSO algorithm, the system is initialized with a population of random solutions which are called (particles) or (intelligent agents) moving via space, and every single potential solution is also allocated a randomized velocity. This algorithm draws on the movement of information among the particles of the population named (swarm). Each particle regulates its trajectory to accomplish its optimum solution called (fitness) which is reached so far. This value is known as (pbest). Moreover, each candidate is able to adjust its trajectory regards to the best previous position obtain through any member of its neighborhood. This value is named (gbest). Each candidate flies in the search space with an adaptive velocity [22].
The fitness function assesses the efficiency of particles to establish the best appropriate solution is obtained. The fitness of the best individual enhances over time and usually declines to stagnate towards the end of the run. In an ideal way, the sluggishness of the process concurs with the successful discovery of the global optimum [23].
Let D be the dimension of the search space taken into consideration and D-dimensional vector represented as X1= (Xi1, …, Xid, …, XiD) designate the current position of ith particle of the swarm. Then the best position ever visited by the particle denotes by Xipbest = (Xi1pbest, Xi2pbest, …, XiDpbest).
Xgbest= (Xi1gbest, Xi2gest, …, XiDgest) depicts the (gbest), i.e., the best position obtained thus far by any particle in the population. The velocity of the particle Vi can be represented by Vi= (Vi1, …, Vid, …, ViD) of ith

particle. Vimax= (V1max, Vi2max, ViDmax) denotes the upper bound on the absolute value of the velocity with which the particle can move at each step. The velocity of the particle and its own position will be defined according to the bellowing equations [24, 25]:

$$V_{id} = w*V_{id} + C_1R_1\left(X_{idpbest} - X_{id}\right) + C_2T_2\left(X_{gbest} - X_{id}\right) \quad (8)$$

$$V_{id} = V_{d\max},\ V_{id} > V_{d\max} - V_{d\max}, V_{id} < -V_{d\max} \quad (9)$$

$$X_{id} = X_{id} + V_{id} \quad (10)$$

Mention that C1 and C2 are positive constants, represents the cognitive and social parameter, respectively, R1 and R2 are random numbers uniformly distributed in the range [0,1]; and ω is inertia weight to balance the global and local search ability. In general, the PSO algorithm can be given the PSO process [25, 26] is as follows:

Step 1:
Specify the lower and upper bounds of the three controller parameters and initialize randomly the individuals of the population including searching points, velocities, (gbest) and (pbest).

Step 2:
Compare the fitness value of the actual particle with (pbest) the entire particle swarm. When the actual value is better it will act as the actual particle (pbest).

Step 3:
Compare the fitness value of the current particle with (gbest) of experiencing the entire particle swarm. If the current value is better, it will act as current (gbest).

Step 4:
Update particle velocity and position according to the velocity and position.

Step 5:
If not assembly the termination condition (frequently the default number of iterations and the fitness value of the lower limit), go back to the step (2). Otherwise, exit the algorithm and obtain the optimal solution.

5. Simulation results and discussion

The element of the heating model of the neonate is assessed as an input-output box. In this model, the temperature of the heater and the electrical power of the radiator has considered as the input-output variables. In this system, the set point is defined as 37 °C and the input energy varies between 0 and 100%, i.e. equivalent to 0 and 400 Watt. Besides, the system can be modeled as a time-invariant system, linear and causal.

To carry out a temperature control, we took real measurements of inputs/outputs of the incubator in the laboratory; the Figure 5 and Figure 6 represent respectively the signal of the interior temperature and the control signal.

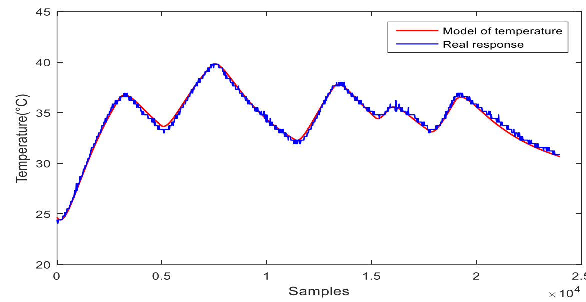

Fig. 5 The estimated and real temperature.

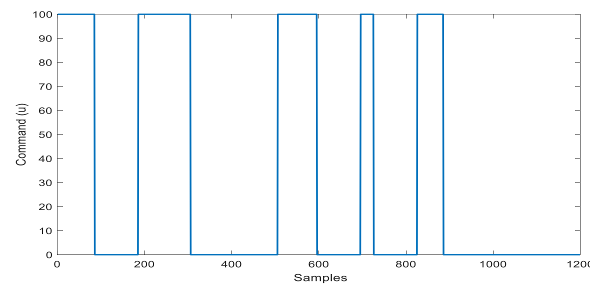

Fig. 6 Input signal.

We use the Identity Toolbox (graphical interface). There are several methods available for the estimation of models: ARMAX, ARX, OE, We estimate with the different structures, we set the sampling time equal to 20 seconds. The transfer function obtained in discrete is as follows [16]:

$$H(z) = z^{-10} * \frac{5.94746e-004z^{-1} + 5.9412e-004z^{-2}}{1-0.8467z^{-1}+0.06462z^{-2}} \quad (11)$$

But for this study, we used the continuous form determined as follows:

$$H(s) = e^{-200} * \frac{3.458e^{-05}s^2\ -3.472e^{-05}*s + 9.103e^{-07}}{s^{-3} + 0.04344*s^2 + 0.02515*s + 5.347e^{-06}} \quad (12)$$

The model established in the Simulink environment is shown in Figure 7.

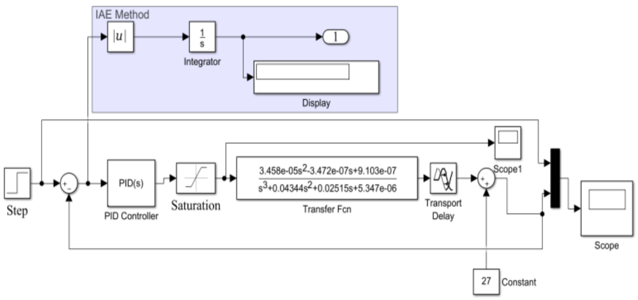

Fig. 7 Simulink model for the incubator.

In the Figure 7 the output port will be replaced each time by one of the four index performance.

5.1. PID-Particle Swarm Optimization tuning

The Particle Swarm Optimization method is applied as a meta-heuristic approach to tune and adjusts the parameters of the PID controller with locale particle coefficient C1, and global particle coefficient C2 both equal 2, the number of particles is 50 and the total number of the iteration is 100 times. In addition, for the test of the PSO method, we used the four cost functions to reduce the error between the closed-loop temperature inside the infant-incubator and the set point of the temperature.

The fitness error functions represented by IAE, ISE, ITAE, ITSE, are used in the comparative study with particle swarm optimization which was mentioned in the previous section with more details. The temperature target inside the unit care is assigned at 37 ℃ as a set point for the closed-loop PID controlled model [27, 28 and 29].

The parameters of the PID optimized based on the Particle Swarm Optimization, the process represented as follows in Figure 8 [30]:

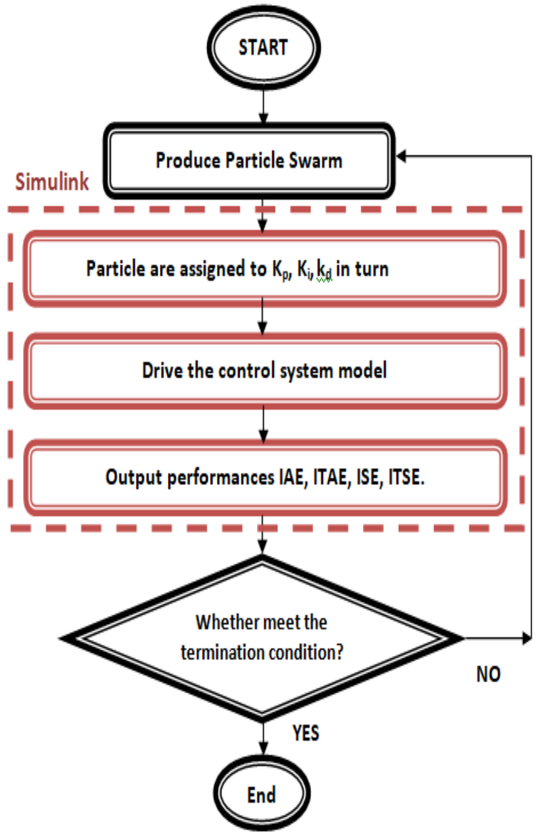

Fig. 8 The optimization process.

In Figure 8 present, the bridge of connection between the PSO algorithms to the Simulink model (in the red box in the Figure 8) is particles the PID parameters and the fitness values related particles.

In model of the infant-incubator, we concentrate on the significant evaluation of the temperature control inside the incubator system. To ensure the improved performance of the proposed approach of PID control optimized with the PSO method.

The controller parameters for the four different performances and also the criteria performance are summarized in Table 2 and Table 3, respectively.

Table 2: Tuning PSO-method comparative analysis.

PSO Method	Kp	Ki	Kd	Figure
PID-IAE	67.2853	0.03	293.4653	Fig.9
PID-ISE	72.7676	0.0155	14.6395	Fig.10
PID-ITAE	59.9788	0.0125	155.1518	Fig.11
PID-ITSE	75.5318	0.03	300	Fig.12

Table 3: The criteria performance of the PSO-algorithm.

PSO Method	**Rise time(s)**	**Settling time(s)**	**Overshoot %**
PID-IAE	360	4.02e+03	11.3%
PID-ISE	349	1.21e+03	6.25%
PID-ITAE	490	922	0.564%
PID-ITSE	314	3.56e+03	13.2%

The step response of the different objective functions are showed in Figure 9 to Figure 12 according to the above parameters:

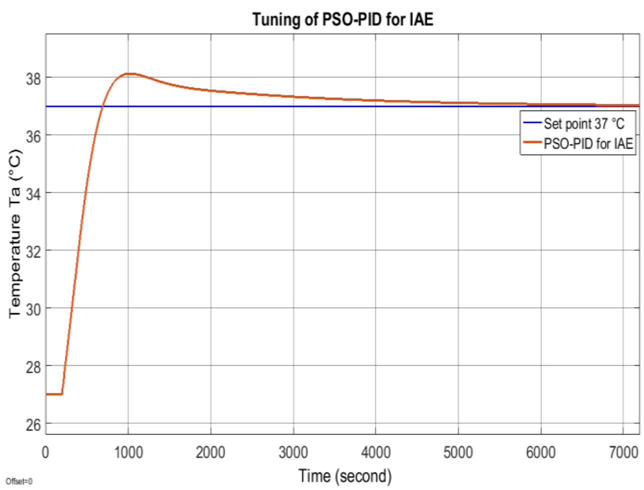

Fig. 9 Response of the IAE objective function.

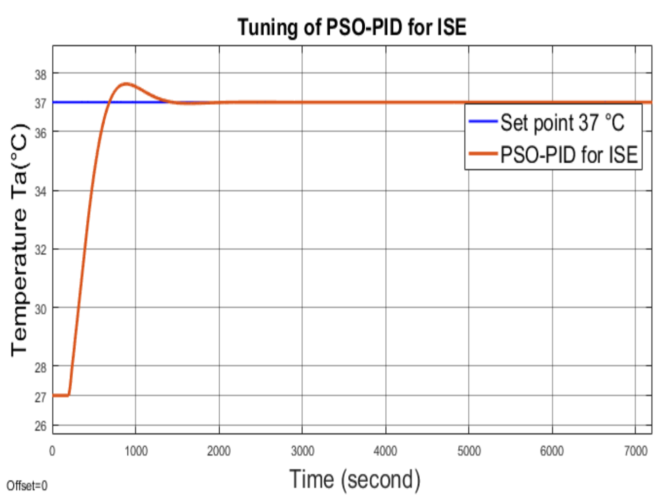

Fig. 10 Response of the ISE objective function.

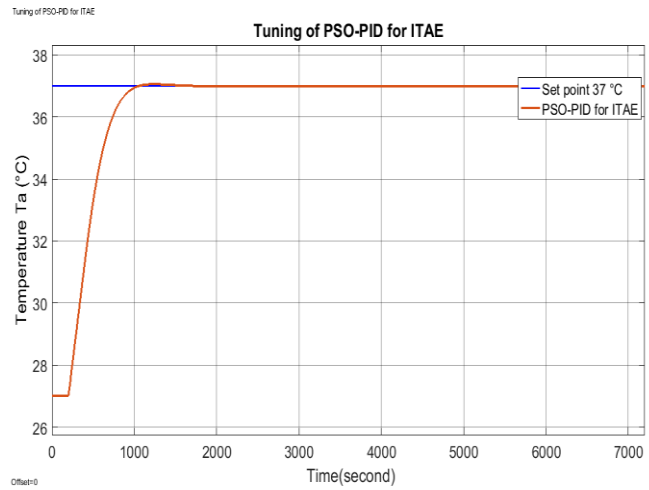

Fig. 11 Response of the ITAE objective function.

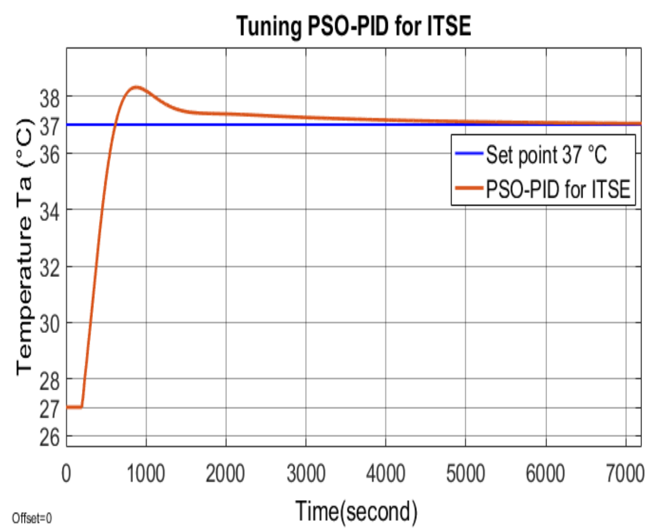

Fig. 12 Response of the ITSE objective function.

The tuned PID controllers based on the PSO approach should be compared over their time-domain responses, in addition to it with the performance index from the four major error criterion techniques of Integral Time of Absolute Error (ITAE), Integral of Absolute Error (IAE), Integral Square of Error (ISE), and Integral Time Square of Error (ITAE) were illustrated in paragraph III in details.

The controller robustness means the ability to tolerate a degree of transformation in the process parameters safely, to not damage the feedback system and make it unstable.

The range of the PID gains was set based on the transfer function of the system and the suitable gains were bounded by the number of iterations. For the linear movement, Table 2 shows the PID gains for each method while Table II shows their dynamical results. The controller, which has superior performance, is considered as the best controller.

The PID gains range was determined based on the transfer function of the neonatal system and the adequate gains were restricted by the number of iterations. Table 2 present the PID gains for each method while Table 3 shows their dynamic results.

In the simulation section, the input energy of the heat resistance varies from [0-400 watt] equal to [0-100%]. Also, the reference is equal to 37 °C.

The control of the air temperature provides the results in a set of figures starting from Figure 9 to Figure 12. On the other hand, Table II shows that the settling time varies from 922 seconds to 4020 seconds with a different objective function.

Tuning by PSO results shows the best behavior which is stable and steady. This behavior is similar to each other when the system was tuned by ISE and ITAE because the rise time it is very close it swings between 490 seconds and 349 seconds, but the result in dynamics with high oscillations and overshoot make the difference to choose the superior controller.

In our case, after all those experimental tests, and from all the results it is obvious that the controller based on PSO-PID with the ITAE objective function is giving the best performance to control the temperature air inside the infant-incubator.

In general, the Powerful of PSO is manifested in the simplicity of the implementation, easily parallelized for concurrent processing, derivative-free, very few parameters in the algorithm, very efficient global search algorithm and it is insensitive to scale of the design. On the other hand, the weakness of this algorithm is a very slow convergence in the refined search stage.

6. Conclusion and future works

The objective of this paper is to develop an optimized PID controller for tuning the temperature inside the newborn incubator using a meta-heuristic method Particle Swarm Optimization.

Right now, we noted that the PSO algorithm has the ability to find the optimum solution dependent on the utilization of different criteria errors as an objective function. The simulation and experimental results showed that the PSO-ITAE algorithm has proven to be a better performance in controlling comparing with other methods. All these PSO-algorithms have exceptional highlights in finding the best solutions. Regardless of these points of interest, they again suffer from different issues.

So researchers propose that, the idea of using the hybrid algorithms to win the deficiencies which happen in the individual algorithms.

References

[1] Kennedy, J., & Eberhart, R. (1995, November). Particle swarm optimization. In Proceedings of ICNN'95-International Conference on Neural Networks (Vol. 4, pp. 1942-1948). IEEE.

[2] Eberhart, R., & Kennedy, J. (1995, October). A new optimizer using particle swarm theory. In MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science (pp. 39-43). IEEE.

[3] Heppner, F., & Grenander, U. (1990). A stochastic nonlinear model for coordinated bird flocks. The ubiquity of chaos, 233, 238.

[4] Sengupta, S., Basak, S., & Peters, R. A. (2019). Particle Swarm Optimization: A survey of historical and recent developments with hybridization perspectives. Machine Learning and Knowledge Extraction, 1(1), 157-191.

[5] Vis, J. K. (2009) Particle Swarm Optimizer for Finding Robust Optima. Leiden Institute of Advanced Computer Science: Holland.

[6] Ending Preventable Newborn Deaths and stillbirths. https://www.who.int/maternal_child_adolescent/newborns/every-newborn/newborns-stillbirths-75percent-prev.

[7] Rojas, R. D., Bell, E. F., & Dove, E. L. (1970). A mathematical model of premature baby thermoregulation and infant incubator dynamics. WIT Transactions on Biomedicine and Health, 3.

[8] Fischer, B., Ludwig, M., & Meister, A. (2001). The thermoregulation of infants: Modeling and numerical simulation. BIT Numerical Mathematics, 41(5), 950-966.

[9] Oliveira, G. H., Amorim, M. F., & Pacholok, C. (2006). A real-time predictive scheme for controlling hygrothermal conditions of neonate incubators. IFAC Proceedings Volumes, 39(18), 387-392.

[10] Oliveira, G. H., & Ushijima, L. H. (2008). A hybrid predictive control scheme for hygrothermal process. IFAC Proceedings Volumes, 41(2), 3641-3646.

[11] Cavalcante, M. U., Torrico, B. C., da Mota Almeida, O., de Souza Braga, A. P., & da Costa Filho, F. L. M. (2010, November). Filtered model-based predictive control applied to the temperature and humidity control of a neonatal incubator. In 2010 9th IEEE/IAS International Conference on Industry Applications-INDUSCON 2010 (pp. 1-6). IEEE.

[12] Zermani, M. A., Feki, E., & Mami, A. (2011). Application of adaptive predictive control to a newborn incubator. Am. J. Eng. Appl. Sci, 4, 235-243.

[13] Zermani, M. A., Feki, E., & Mami, A. (2014). Building simulation model of infant-incubator system with decoupling predictive controller. IRBM, 35(4), 189-201.

[14] Feki, E., Zermani, M. A., & Mami, A. (2017). GPC temperature control of simulation model infant-incubator and practice with arduino board. IJACSA, 8 (6), 46-59.

[15] Ali, J. E. H., Feki, E., & Mami, A. (2019). Dynamic Matrix Control DMC using the Tuning Procedure based on

First Order Plus Dead Time for Infant-Incubator.10 (6), 358-367.
[16] Dräger. Incubator User Manual;Drager_Incubator_8000SC, Germany,http://www.frankshospitalworkshop.com/equipment/documents/infant_incubators/user_manuals/Drager_Incubator_8000SC_-_User_manual.pdf.
[17] Shabaan, A. R., El-Metwally, S. M., Farghaly, M. M., & Sharawi, A. A. (2013, September). PID and fuzzy logic optimized control for temperature in infant incubators. In 2013 5th International Conference on Modelling, Identification and Control (ICMIC) (pp. 53-59). IEEE.
[18] Jaen-Cuellar, A. Y., de J. Romero-Troncoso, R., Morales-Velazquez, L., & Osornio-Rios, R. A. (2013). PID-controller tuning optimization with genetic algorithms in servo systems. International Journal of Advanced Robotic Systems, 10(9), 324.
[19] Aydoğdu, Ö., & Korkmaz, M. (2019). Optimal design of a variable coefficient fractional order PID controller by using heuristic optimization algorithms, 10 (3), 314-320.
[20] Solihin, M. I., Tack, L. F., & Kean, M. L. (2011, January). Tuning of PID controller using particle swarm optimization (PSO). In Proceeding of the International Conference on Advanced Science, Engineering and Information Technology (Vol. 1, pp. 458-461).
[21] Bonyadi, M. R., & Michalewicz, Z. (2017). Particle swarm optimization for single objective continuous space problems: a review. Evolutionary Computation Journal. 25 (1), 1-54.
[22] Giriraj Kumar, S. M., Sivasankar, R., Radhakrishnan, T. K., Dharmalingam, V., & Anantharaman, N. (2008). Particle swarm optimization technique based design of Pi controller for a real-time non-linear process. Instrumentation Science and Technology, 36(5), 525-542.
[23] Le, D. N., Nguyen, N. G., & Le, V. T. (2012). A novel PSO-based algorithm for the optimal location of controllers in wireless networks. International Journal of Computer Science and Network Security (IJCSNS), 12(08), 23-27.
[24] Yazd, H. G. H., Arabshahi, S. J., Tavousi, M., & Alvani, A. (2015). Optimal designing of concrete gravity Dam using particle swarm optimization algorithm (PSO). Indian Journal of Science and Technology, 8(12), 1-10.
[25] Babaie, O., & Esfahany, M. N. (2020). Optimization and heat integration of hybrid R-HIDiC and pervaporation by combining GA and PSO algorithm in TAME synthesis. Separation and Purification Technology, 236, 116288.
[26] Azar, A. T., Ammar, H. H., Ibrahim, Z. F., Ibrahim, H. A., Mohamed, N. A., & Taha, M. A. (2019, October). Implementation of PID Controller with PSO Tuning for Autonomous Vehicle. In International Conference on Advanced Intelligent Systems and Informatics (pp. 288-299). Springer, Cham..
[27] Solihin, M. I., Tack, L. F., & Kean, M. L. (2011, January). Tuning of PID controller using particle swarm optimization (PSO). In Proceeding of the International Conference on Advanced Science, Engineering and Information Technology (Vol. 1, pp. 458-461).
[28] Azar, A. T., Ali, N., Makarem, S., Diab, M. K., & Ammar, H. H. (2019, October). Design and Implementation of a Ball and Beam PID Control System Based on Metaheuristic Techniques. In International Conference on Advanced Intelligent Systems and Informatics (pp. 313-325). Springer, Cham..
[29] Nagaraj, B., Subha, S., & Rampriya, B. (2008). Tuning algorithms for PID controller using soft computing techniques. International Journal of Computer Science and Network Security, 8(4), 278-281.