International Journal of Automation and Computing 14(6), December 2017, 686-695
DOI: 10.1007/s11633-017-1069-8

Optimizing the Double Inverted Pendulum′s Performance via the Uniform Neuro Multiobjective Genetic Algorithm

Dony Hidayat Al-Janan[1,2] Hao-Chin Chang[1] Yeh-Peng Chen[1] Tung-Kuan Liu[1]

[1]Institute of Engineering Science and Technology, National Kaohsiung First University of Science and Technology, Taiwan, China

[2]Mechanical Engineering Departement, Engineeering Faculty, Semarang State University, Semarang, Center of Java, Indonesia

Abstract: An inverted pendulum is a sensitive system of highly coupled parameters, in laboratories, it is popular for modelling nonlinear systems such as mechanisms and control systems, and also for optimizing programmes before those programmes are applied in real situations. This study aims to find the optimum input setting for a double inverted pendulum (DIP), which requires an appropriate input to be able to stand and to achieve robust stability even when the system model is unknown. Such a DIP input could be widely applied in engineering fields for optimizing unknown systems with a limited budget. Previous studies have used various mathematical approaches to optimize settings for DIP, then have designed control algorithms or physical mathematical models. This study did not adopt a mathematical approach for the DIP controller because our DIP has five input parameters within its nondeterministic system model. This paper proposes a novel algorithm, named UniNeuro, that integrates neural networks (NNs) and a uniform design (UD) in a model formed by input and response to the experimental data (metamodel). We employed a hybrid UD multiobjective genetic algorithm (HUDMOGA) for obtaining the optimized setting input parameters. The UD was also embedded in the HUDMOGA for enriching the solution set, whereas each chromosome used for crossover, mutation, and generation of the UD was determined through a selection procedure and derived individually. Subsequently, we combined the Euclidean distance and Pareto front to improve the performance of the algorithm. Finally, DIP equipment was used to confirm the settings. The proposed algorithm can produce 9 alternative configured input parameter values to swing-up then standing in robust stability of the DIP from only 25 training data items and 20 optimized simulation results. In comparison to the full factorial design, this design can save considerable experiment time because the metamodel can be formed by only 25 experiments using the UD. Furthermore, the proposed algorithm can be applied to nonlinear systems with multiple constraints.

Keywords: Double inverted pendulum (DIP), UniNeuro-hybrid UD multiobjective genetic algorithm (HUDMOGA), uniform design (UD), metamodel, euclidean distance.

1 Introduction

In accordance with improvements in the industrial system, Industry-4.0 has recently exhibited rapid growth in industry, and has evolved from using traditional production technologies to integrating automation and data exchange, as well as mechanical, electronic, and manufacturing technologies, thereby bringing together cyber-physical systems and the Internet of services. For this reason, robotics and automation technologies continue to be developed to support these systems.

In addition to its complexity and because of the consequence of multiple connections between several components, a typical automation system has control properties that are nonlinear and cannot be neglected. Furthermore, simple individual systems are highly coupled with others. Sometimes, the output quality can be improved by optimizing the intangible mechanism to control a nonlinear system[1]. Nonlinear systems have been applied in various domains, e.g., almost all robotic mechanisms use nonlinear control. Solving nonlinear systems is a challenge, but sensitivity engendered by high-frequency pulse signals for use in mechatronics systems enables the adoption of a model with conditions similar to those of complex real problems within a utopian system, such as unlimited uncontrolled noise within the system itself. The model formed by the input and output of the experimental data (metamodel) in this study is originated as a popular solver bridge used for unknown models or noise. It developed to become an adaptive optimization controller that can be used a flexible model[2, 3].

Nonlinearity and physical phenomena are strongly connected. One can argue that nonlinearity is ubiquitous, e.g., fluid phenomena, dynamism, elasticity, relativity, combustion, heat transfer, and thermodynamics are all approached by nonlinear equations, except for quantum mechanics. With this attention, research in this field has recently grown dramatically, mostly in the form of analysing all aspects, concepts, and applications of nonlinearity in both the micro and the macro scales. Understanding nonlinearity requires

Research Article
Manuscript received May 22, 2016; accepted September 29, 2016; published online May 24, 2017
This work was supported by Indonesian Government (No. BPPLN DIKTI 3+1).
Recommended by Associate Editor Veljko Potkonjak

an understanding of the basic concept of linearity as the simplest point of access for an ideal system, even in purely theoretical terms. By contrast, when such a system is already known as a linear regime, it faces nonlinearity as a perturbation of itself[4].

The inverted pendulum is a classic model that is typically strongly coupled, contains multiple variables, and has been used in many previous studies regarding nonlinear control systems to validate control theory. A consequence of the nonlinear system is that even if the model is determined, any controlled disturbance may modify the main model. The inverted pendulum system implies a highly nonlinear and unstable device because it is easily influenced by both elements within the system and by outside interference. The two types of inverted pendulum based on cart movement are the rotary and horizontal types. Moreover, single-link and multiple-link types exist. According to various preliminary steps, the inverted pendulum can be controlled in two states: swing-up, which uses the control energy in the pendulum to swing into an upward position, and balancing, which controls the stable position of the pendulum[5].

Some previous studies on double inverted pendulum (DIP) optimization have used proposed control system as the fitness function. Liu et al.[6] optimized the parameters for evaluating preferences for the DIP using a genetic algorithm (EPGA). This approach adopts the traditional genetic algorithm (GA), however, search solutions are not improved by limited searches with Pareto dominance. The EPGA sensory-motor intelligent schema approach is used to control and optimize the DIP. A coded number is used as a chromosome that is optimized to find the most favorable setting for 16 parameters. In another study, Liu et al.[6] used simulation for the design of the DIP controller. A human-simulated intelligent control method was used as an approach controller that was applied in four-phase control positions. An inequalities-based multiobjective GA (MOGA) was used to determine the displacement over the rail boundary, the number of swings, settling time, overshoot of the total energy, and control effort (multiobjective). Finally, the chromosome was applied in the form of real number coding with 16 parameters. Other researchers have used Matlab-Simulink for modelling the inverted pendulum[7] and applied the analysis approach in the dynamic-web for presenting the inverted pendulum to model unstable systems[8].

A fuzzy GA with six weighted parameters and multiple input − single output was employed to optimize DIP parameters. Weighting decreased the number of parameters from six to two, thus reducing the degree of fuzzy logic[9]. A linear quadratic regulator was used to control the stabilisation of the DIP. Jacobian, Eulerian and Lagrangian methods were used in the linearization procedure. The mathematical modelling of the present study used Lagrangian decomposition of kinetic energy and potential energy to determine a feedback value composed of six parameters[10]. In internet-based Java simulations, the DIP was controlled by a decoupled sliding mode controller with multiobjective particle swarm optimization[11]. Furthermore, a spherical inverted pendulum was studied to improve the solutions for standing up to control the output regulation. The spherical inverted pendulum was employed by using an NN (UniNeuro) approach for forming the new model[12, 13]. This study used the DIP of a horizontally-moving cart. The cart was driven by a servo motor that produced rotary movement that was converted to horizontal movement through a double link belt mechanism. Five-parameter input was processed in a signal controlling unit, which was affected by the control pulse input on a servo motor. That motor was influenced by 3 feedback signals from the decoder when the mathematical control system was unknown. This condition occurred when utilization time has elapsed and the system configuration has changed from its initial condition. In accordance with this, the main objective of this study was to determine several new feasible optimal input settings for the DIP. Although this problem can be solved by traditional methods, parameter input value settings were performed by time-consuming trial-and-error repetition, and were tuned on the basis of instinct and intuition. The present paper explains how UniNeuro-hybrid uniform design multiobjective genetic algorithm (HUDMOGA) can optimize the input DIP control settings automatically. In this case, the uniform design (UD) is employed to capture training data from the UD hybrid in NN, and the HUDMOGA is used to generate the optimized parameter settings.

The remainder of this paper is organised as follows. Section 2 discusses the scope of the problem. The proposed method is described in detail in Section 3. The results and a discussion are presented in Section 4. A conclusion is offered in Section 5, and references are listed in the bibliography.

2 Problem definition

An inverted pendulum is an underactuated instrument with an unstable open loop within a highly dynamic nonlinear system. Thus, it is a perfect system for arrangements of both classical and advanced controls. The inverted pendulum′s approach can be applied to a wide range of signal control processes such as robotic processes, or even rocket field processes. The system was initially proposed for modelling such nonlinear phenomena. Because of highly coupled interactions, the nonlinear effect cannot be neglected within the systems response, thus it is useful for studying nonlinear modern control[14].

In an inverted pendulum, the objective equilibrium point is its upright position stability. This condition occurs through the control of such parameters that produce the cart movement for swinging the pendulum from a dead position into a standing position. Meanwhile, the cart movement is controlled in a particular position because of limitations of the rail length. All the mentioned conditions are achieved through horizontal movement of the cart. Hence, the inverted pendulum is classified as an underactuated mechanical system with fewer control inputs than degrees of freedom. This status proves that this classical system has the ability to evaluate all control strategies[14].

This paper explains the optimization of the five DIP input parameters that are changed from the initial condition because of elapsed utilization time. The initial model of the system changed in an undetermined fashion, causing difficulty in controlling the DIP using the standard equipment settings. This problem became more complex because a large amount of information within the product was lost. Thus, conducting this study was similar to conducting blind optimization. In a real setting with a limited budget, engineers must optimize the frequency of an unclear model with similar blind optimization.

The cart of the DIP system is driven horizontally by a servo motor that couples a frictionless link and a belt mechanism used for conversion from a rotary movement to a horizontal movement. The motors amplifier is used to amplify the input signal combined with signals from three encoders for controlling the rotation action of the motor. The motors action is processed by a digital signal processor (DSP). The three encoders serve to capture the reaction of the motor and the pendulum motion, and subsequently deliver the feedback signal to the DSP. Three decoders are placed at three points. The first point is in the motor. This decoder is used for detecting the reaction of the motor to its input. The second decoder is positioned on a joint between the cart and Pendulum 1, and the third is mounted in a joint between Pendulums 1 and 2. These two left decoders serve to identify the movement of the pendulum in each joint.

The movement of the pendulum is a rotary swinging motion caused by inertia horizontal relative movement of the cart coupled with the effect of feedback signals from the aforementioned three decoders. The enumerated reaction to the decoder motion is displayed on three curves, called displacement, velocity, and output signal. Finally, for securing and limiting the motion of the cart, two limit switches are placed on each side. To visualise the principle of the DIP, Figs. 1 and 2 show the DIP controlling system and mechanism, respectively.

Table 1 shows the DIP specifications, including mass, dimensions, moving coefficient, and moment.

Five input parameters are employed to control the DIP system, which has a tiny step (10^{-4}) numerical setting that affects its sensitivity (Table 2). Therefore, a powerful method is required to determine the feasible combination values.

In response to the aforementioned problem, this study aimed to determine suitable input parameter settings for the DIP for driving the pendulum to stand up and maintain stability for a certain period, during which the external pendulum may be subjected to disturbances while the relationship between the five input parameters is unknown. Thus, in this paper, the UniNeuro-HUDMOGA is proposed.

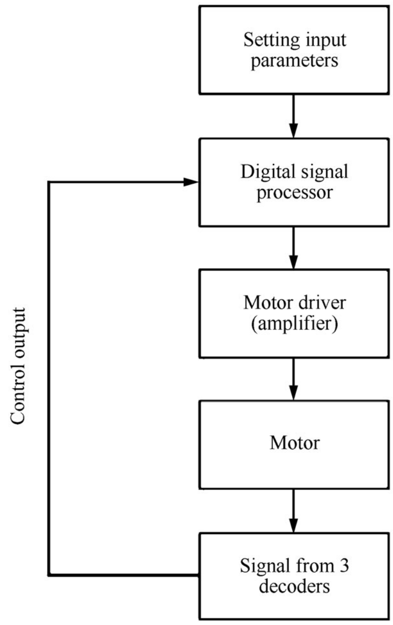

Fig. 1 DIP′s controlling system

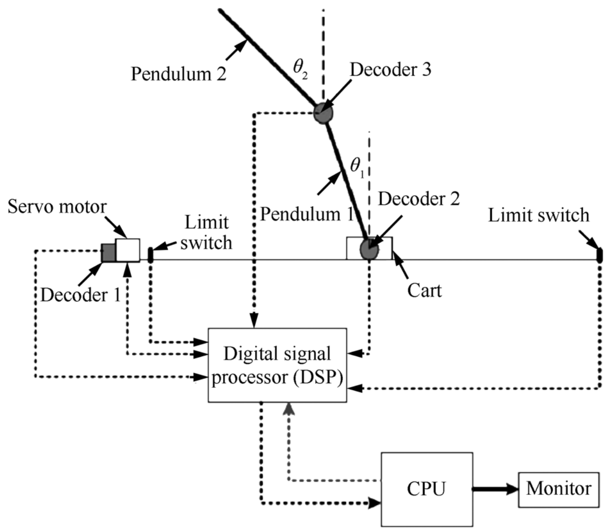

Fig. 2 DIP′s mechanism

Table 1 DIP specification

Parameters	Values
Mass of cart (M)	1.32 kg
Mass of encoders (m_0)	0.208 kg
The center position of each pendulum (l_1, l_2)	0.2 m
Length of each pendulum (L)	0.4 m
Displacement range of cart (x)	$(-0.3, 0.3)$ m
Mass of each pendulum (m_1, m_2)	0.108 kg
Friction coefficient of cart (f_o)	22.915 N·s/m
Friction coefficient of pendulums (f_1, f_2)	0.775 6 N·s·m
Moment of inertia of pendulums (J_l, J_2)	0.001 44 kg·m^2

Table 2 DIP′s parameters input

Input parameters	Lower bound	Upper bound
Initial acceleration	0.4	0.9
Feedback coefficient direction	2.0	2.5
Negative feedback coefficient direction	2.0	2.5
Pull acceleration	5.0	7.0

3 Uniform neuro multi-objective genetic algorithm

The proposed method combines the UD, NNs, and MOGA. Principally, the model for the fitness function of the MOGA programme is created by NNs using the UD to manage the number of experiments for generating model. The complete procedure is explained below.

3.1 Assemble data using UD

Various experimental designs have been developed to minimize the number of experiments for obtaining a result close to the optimal solution within the full factorial design of experiment, or through a selected a number of samples that represent the whole population[15, 16]. The number of experiments had to be reduced in consideration of the costs, e.g., to examine six parameters with six levels, 6^6 experiments traditionally using the full factorial design are required. With the advancement of the experimental design method, the experiments can be reduced to 6^2 or even less. However, the result manages to come close to the most favorable result using the full factorial design. Some researchers have attempted to develop evolutionary experimental designs for reducing the number of samples by fractions of full factorials, such as the central composite design, hypercube, and orthogonal arrays which is developed into Taguchi design and the UD[15, 17].

In this study, the UD is used for determining sample data for developing a model in an NN (UniNeuro) through the training data, the result was embedded in a GA to increase searching performance.

Using the UD, the number of experiments can be decreased considerably from the result of the complete full factorial design because the UD is a space filling design that seeks a point that can be randomly positioned in the domain. Furthermore, the UD can explore the relationships between the underlying model specifications. By using the UD to consider the patterns of the data spread and uniform density, the training sample data represents the true model of the system.

The UD was developed by Fang and Yang, they published their research online, and made the UD open access and easily downloadable (http://www.math.hkbu.edu.hk/UniformDesign/) (UD-web)[17, 18]. The UD-web provides a high number of appraisals for a UD with three classifications for discrepancy of uniformity, as well as 2 to 29 parameters, enabling each design to be applied to an n+1 parameter experiment. Optionally, if the UD-web is not appropriate to the problem, the UD can be developed flexibly using the uniformity of an experimental design with L_2-centered discrepancy (CD), which can be calculated using (1):

$$CD = \left(\frac{9}{8}\right)^s - \frac{2}{n}\sum_{k=1}^{n}\prod_{j=1}^{s} \left(1+\frac{1}{2}|x_{ki}-0.5|+\frac{1}{2}|x_{ji}-0.5|-\frac{1}{2}|x_{ki}-x_{ji}|\right)+ \frac{1}{n^2}\sum_{k=1}^{n}\sum_{j=1}^{n}\sum_{i=1}^{s} \left(1+\frac{1}{2}|x_{ki}-0.5|+\frac{1}{2}|x_{ji}-0.5|-\frac{1}{2}|x_{ki}-x_{ji}|\right) \quad (1)$$

where s is number of parameters, n is number of experiments, k is rows′ number, j is columns′ number, i is parameter′s number. The better UD has the smaller CD value that concludes its uniformity quality. UD has the smaller CD value that concludes its uniformity quality.

Using CD criteria, we can develop a new suitable UD, e.g., we can initially use a Latin square design as the base structure, and then optimize it through an optimization design with a smaller, more favorable CD as the fitness criterion[17, 18].

For a study that only requires five parameters, the UD-web can be sufficient if the following steps are followed:

1) Determine the number of parameters by considering the level of each parameter. The experiment has a minimum number of (number of parameter + 1), and the total level is equal to the number of experiments.

2) Select the appropriate table from UD-web for use in the experiment. In this case, we selected 5^2 experiments.

Table 3 shows the UD table from UD-web for five parameters with six experiments. The number of rows represents the number of experiments performed, and the number of columns denotes the number of parameters (A, B, C, D, E).

Table 3 UD′s table of five parameters ($U_6(6^5)$)

Run	A	B	C	D	E
1	3	1	6	5	4
2	5	2	2	2	5
3	6	5	3	6	3
4	4	4	5	1	1
5	2	6	4	3	6
6	1	3	1	4	2

This study used a UD table for five parameters, with 25 experiments researched. The UD table was used for data collection. Each parameter was evenly divided into a number of levels from the lowest level to the highest level. Subsequently, each parameter was used for data retrieval objectives for standing and stability while standing. Standing success offered a score of 0 to 1 through observation, and seconds were employed as the unit of duration for stability while standing. The standing measurement was performed with some external excitation from Pendulum 2.

3.2 Uniform design embedded in neural network

An NN uses such correlation method of the input–output data layer through a neural mechanism for developing a model of the model (metamodel) as the approach model, and the amount of training data for determining the data distribution is one of the main factors of the NNs success. An NN that processes large quantities of data tends to yield an accurate prediction for gain time consumption, hence, several researchers have focused on the critical issue of determining the efficiency of training data[19]. Therefore, to enrich the quality of the model by conducting an appropriate number of experiments, in this study, the UD is used to determine the number of setting experiments for training data. Finally, the NN introduced the model by calculating the distribution of the weighted learning rate and the deviation for each point neuron using the sigmoid activation function[19]. There is no fixed rule for building the connections in the NN or for determining the approach procedure. Moreover, as a prediction method, the NNs forecasting should be controlled to avoid overfitting or excessive confidence. Therefore, the UD used in this study to distribute the collection of data uniformly with a small number of experiments. UD also generated the potential data that had primary effects for building a metamodel of the NN. This is named UniNeuro.

Before assignment into the NN, the data result from the experiment requires normalization through (2):

$$x_{\rm norm} = \frac{x - x_{\rm min}}{x_{\rm max} - x_{\rm min}} \qquad (2)$$

where $x_{\rm norm}$ is a value that has been normalized on this parameter. x is the value of data in each experimental run. $x_{\rm max}$ and $x_{\rm min}$ are the maximum and minimum values in each parameter[19]. Normalization generates a data value between 0 and 1. This procedure fairly compares all parameters that can be connected to each other without any difference in criteria or unit. The procedure is subsequently used to develop a metamodel in the NN (Fig. 3).

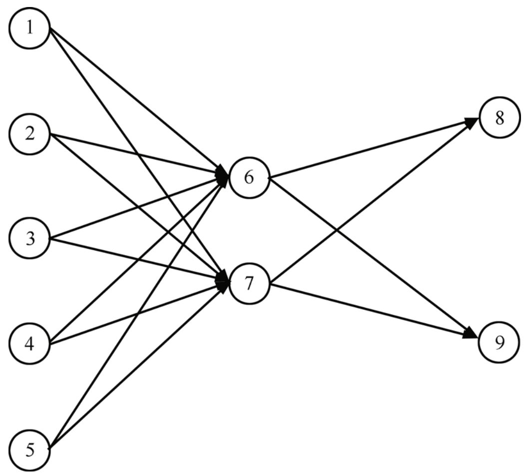

Fig. 3 Neuralnetwork schema

To build the metamodel, this research uses one hidden layer with two neurons. Fig. 3 shows points 1 to 5 denoted five input parameters determined in the UD design of experiments (DOE). Points 6 and 7 are two neurons in a single hidden layer, serving as prediction points derived from neuron information from the input, including weight information. Finally, points 8 and 9 are two objectives (multi-objective) based on the experimental results.

The NN is employed through the following steps:

Step 1. Determine the design of the NN, the number of neurons and hidden layers, the objective, learning rate (α), initial weight at each point neuron, and error criteria using the mean square error (MSE) for objectives 1 and 2. The MSE is used for constraining model overfitting.

Step 2. Arrange the normalized training data that are collected in the UD in the input and output of the NN.

Step 3. Calculate the weight factor, which in this case is calculated using the sigmoid activation function.

Step 4. Build the linear metamodel for the two objectives, the linear metamodel consists of several specific weights implemented through parameters for constructing a value close to that of a real solution.

Step 5. Check the MSE for the two fitness values of the two objectives in the metamodel using part of the training data. If the result is higher than the standard, return to Step 3, otherwise, move on to Step 6.

Step 6. The metamodel is formed, and subsequently used as a fitness function in the HUDMOGA.

3.3 Hybrid uniform design multi-objective genetic algorithm

After the metamodel is formed by the UniNeuro, the optimal parameter settings are achieved through the HUDMOGA, which is applied through the following steps:

Step 1. Determine the parameters of the HUDMOGA, population number (PN), selection number (SN), generation number (GN), and Euclidean number (EN).

Step 2. Initialization is performed to generate a parent chromosome by initially using the UD for five parameters and 25 experiments obtained from the UD-web as real settings that have already been confirmed. Subsequently, the remainder of the population is generated randomly to accelerate the optimization procedure. Considering that the fitness function comes from the metamodel established by the UniNeuro, the value of the parameter is set between 0 and 1. Furthermore, the fitness value is calculated using the metamodel. The fitness value is the sum of the successful standing periods (the larger the better) and the steadiness duration (the larger the better). Finally, the sum of these outputs is evaluated (the smaller the better).

Step 3. The selection process involves a roulette wheel. All the parent chromosomes occupy the roulette wheel based on its probability for the whole result. The roulette wheel is then rotated to select chromosome pairs. Selection of the same chromosome pair more than once must be avoided to maximise the diversity of the solution set[20].

Step 4. The crossover process is used to set the partition crossover that is applied to all the selected chromosome pairs. This method is initiated by determining a number for the genes position, which is subsequently used as the genes fixed position, finally replacing the unfixed gene with another gene from the paired chromosomes. The same procedure is then carried out for the remaining chromosomes[20].

Step 5. All chromosomes from Step 3 are mutated in this step through a neighbor search. This method is applied to chromosomes individually by randomly determining two genes, then generating 4 permutations from these 2 genes, and using the permutation results to develop four new chromosomes in the same gene positions[20].

Step 6. To increase variety among the children, the chromosomes from the crossover and mutation are combined and then selected for 2 chromosomes randomly. Subsequently, the selected chromosomes are generated by experiments using a unique combination of a two − level UD for five parameters and 25 combinations[17, 18]. A total of 21 unique combinations are obtained. Because of its flexibility, the UD can be set according to the combination of the two levels and the number of available experiments[17, 18]. Principally, this procedure is similar to the hybrid Taguchi GA[20], but with the two levels of the Taguchi DOE replaced with the two levels of the unique UD.

Step 7. The chromosomes from Steps 4−6 are combined in each child′s chromosome, and the parents′ and children′s chromosomes are combined. The chromosome with the shortest fitness value is stored as the most favourable chromosome and updated every generation. The remaining NP−1 chromosomes are subjected to Darwinian selection (survival of the fittest).

Step 8. For maintaining diversity among the solution sets in each generation, the remaining chromosomes for the iteration described in Step 7 are filtered using Euclidean distance and selected for NP−1 chromosomes. The Euclidean distance method is used as a vector distance between each pair of chromosomes in sequence. Through this method, the various solution sets are proofed for their diversity based on the solution combination positions. In this case, because we have five solution sets, five solution dimensions are formed. The higher the Euclidean distance value, the higher the diversity among the solutions produced. An excessively high Euclidean distance only produces one single solution because of its distinct nature.

Step 9. Selected results from Step 8 are ranked using the Pareto method. As a multiobjective problem, the two objectives trade off. In this case, we first chose the most feasible solution for each running programme, and then confirmed the solution with the automatic system. Alternative solutions were selected by Pareto front as the most favorable alternatives based on their abilities to update each generation[21, 22].

Step 10. This study uses the number of iterations as a stopping criterion. If the iterations performed are lower than the GN, the process is repeated from Step 3. If the iteration is met, continue to Step 11.

Step 11. The most favorable result is displayed. This result is checked with the DIP equipment to verify its reliability.

The global procedure of the UniNeuro-HUDMOGA is shown in Fig. 4.

4 Result and discussion

To examine the proposed algorithm, the UniNeuro-HUDMOGA was programmed in Matlab that was run on an Intel ® Core ™ i7-2600 CPU at 3.40 GHz and 8 GB RAM. The UniNeuro used the following parameters: learning rate (α)=0.1, mean square error (MSE) = 0.01, initial weight for $w_{n=1-5,m=6,7} = -0.5$ to 0.5 (selected randomly), initial weight for $w_{n=6-7,m=8,9} = -1$ to 1 (selected randomly); $\theta_{n=6-9} = -0.5$ to 0.5 (selected randomly). The HUDMOGA parameters were set as follows: PN = 100, SN (selection number) = 0.8×PN, GN = 1 000, and EN = 0.2. In addition, to confirm that this method could stably obtain the right solution, the algorithm was run 20 times.

In this study, data was retrieved by dividing the range of each parameter setting into 25 level sections, and then applying the settings on the UD table obtained from the UD-web. The data obtained served as training data for the UniNeuro to form the metamodel. Subsequently, the metamodel was used as a fitness function of the HUDMOGA. Finally, the settings results were confirmed by the DIP equipment (Table 4).

Table 4 Optimized results by using UniNeuro-HUDMOGA

A	B	C	D	E	Swing time (s)	Steady time (s)
0.590 4	2.115 2	2.115 2	5.451 2	5.964 8	8.0	900
0.778 3	2.021 7	2.035 7	5.250 0	5.272 7	6.5	900
0.886 3	2.031 4	2.037 5	5.145 0	5.123 6	7.0	900
0.900 0	2.500 0	2.481 1	5.125 0	5.866 0	5.5	900
0.900 0	2.499 9	2.478 2	5.125 1	5.920 9	5.0	900
0.900 0	2.500 0	2.481 1	5.125 0	5.866 9	5.5	900
0.438 5	2.023 8	2.020 0	5.117 6	5.166 7	11.0	900
0.900 0	2.026 4	2.055 6	5.125 0	6.321 5	7.0	900
0.439 5	2.001 5	2.020 2	5.124 5	5.182 0	11.0	120

Notes: A: Initial acceleration, B: Feedback coefficient direction, C: Negative feedback coefficient direction, D: Pull acceleration, E: Reverse acceleration.

Table 4 shows the DIP optimized result set obtained from running the UniNeuro-HUDMOGA 20 times. After running the algorithm, nine setting parameters were selected according to their success in confirming the DIP. Swing time denotes the time taken to swing followed by standing up, steady time denoted the time period for standing with interruptions from any excitations on the outside of Pendulum 2.

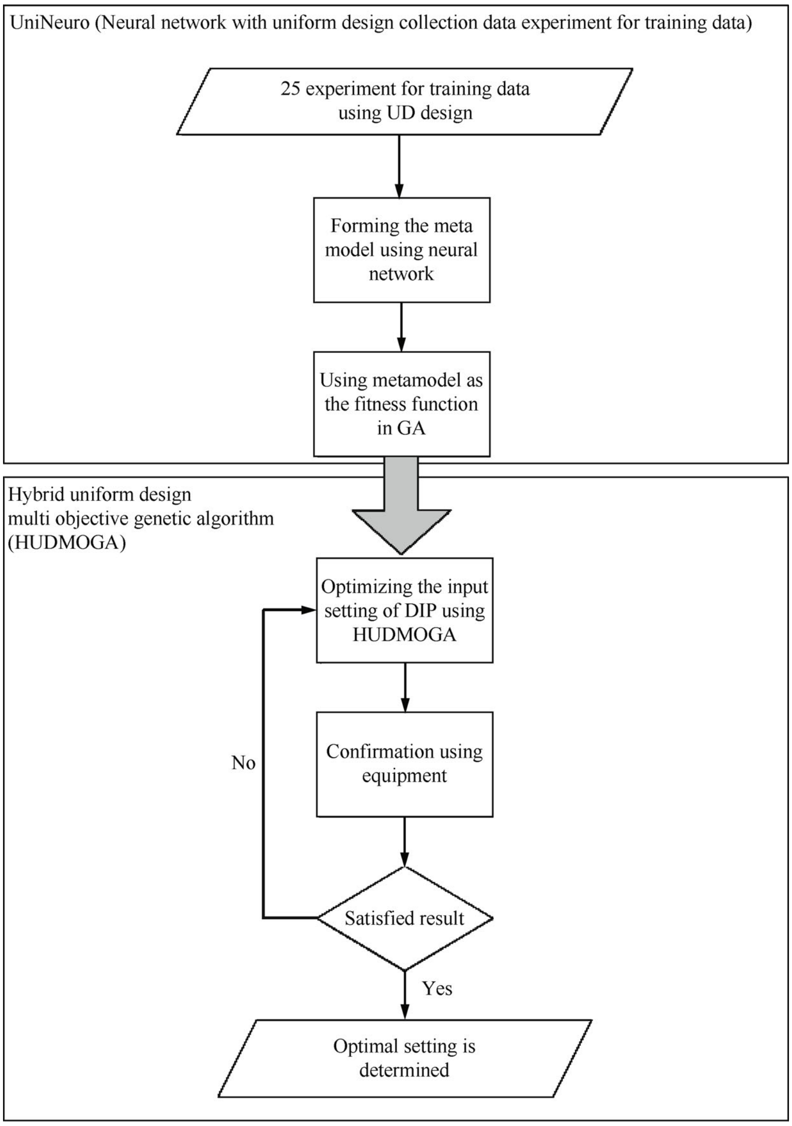

Fig. 4 UniNeuro-HUDMOGA design

Steady time was judged for 900 s to conclude the optimized settings.

The proposed algorithm has nine setting values that can be chosen, previous studies have used only one setting, which can be determined as the optimization target caused by the rigid approach model[5, 6, 9−11, 21]. Moreover, the feasible solution may still only be available by running this programme, and subsequently updating the training data by using 20 confirmed settings. Hence, because of the artificial learning method, this development training data may increase the probability of obtaining the optimized result. The result of the optimization procedure can be rendered more precise if the model is closer to its real condition. Because of space limitations in this paper, only one parameter setting result is shown for each of displacement, velocity, and the output signal curve (Figs. 5–7).

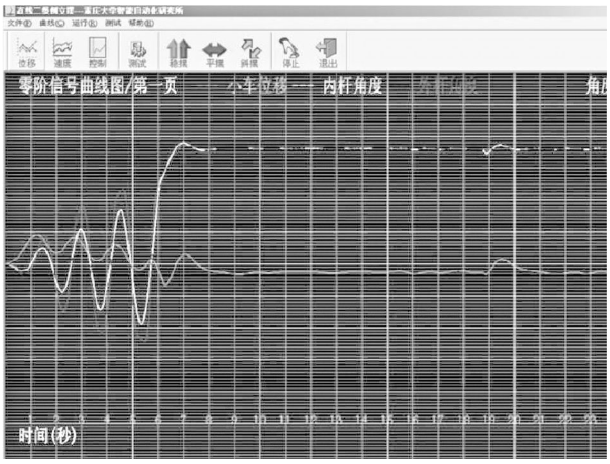

Fig. 5 DIP′s displacement curve

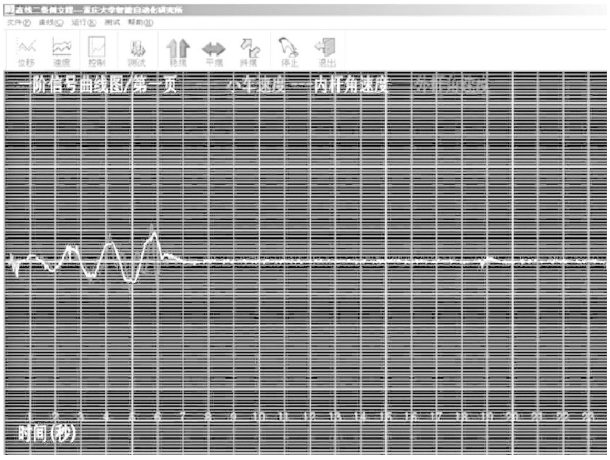

Fig. 6 DIP′s velocity curve

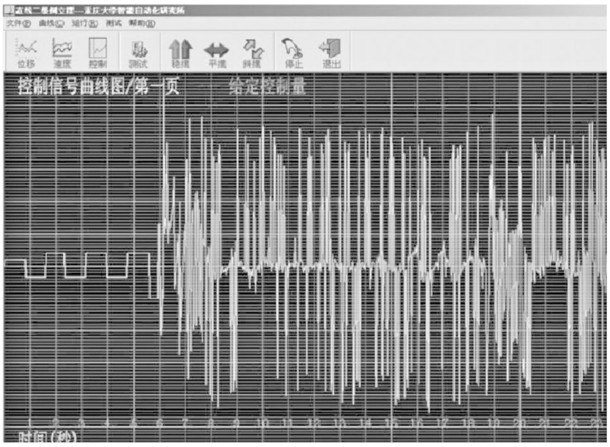

Fig. 7 DIP′s signal curve

The result proved that the UniNeuro is able to modify a metamodel that is used as a fitness function for the HUDMOGA. In other words, using the UD for determining the experiment training data set in the UniNeuro is extremely efficient for capturing or predicting the model of the DIP. Nine optimal settings can be produced from only 25 experiments. For future research, some modifications would be required to verify the robustness of the UniNeuro and improve its probability of success, similar to the adaptive fuzzy model used to obtain vital information for improving the auto-error correction model[3, 23, 24], and to a modified NN as a wavelet[25].

The optimization procedure was finalised using the metamodel from the UniNeuro as the fitness function of the HUDMOGA. In the optimization procedure, the HUDMOGA proved its performance by generating the true result for the optimal setting of the DIP, demonstrating this step as being able to obtain several true result settings, and also as being able to avoid being trapped in local optima. While the 11 failure settings from 20 simulations may have been caused by an incorrect metamodel produced by the UniNeuro as the prediction method, common sense suggests that it belonged to a limited amount of training data.

The results indicate that the proposed algorithm can solve this problem effectively and efficiently, if it is run 20 times using the 25 real data from the UD. It is evident that the proposed algorithm represents an outstanding trial-and-error method that makes the setting process simple, and saves experimentation time. However, the reliability of the proposed algorithm must be improved to increase the performance of the optimization procedure. Furthermore, some particular aims such as standing time could be constrained.

Using the UD in assigning the training data experiment for NN is proven to be effective. In this case, information regarding the model could have been suggested even without the trend of the graphic result. Instead, the UD can promote a small sample that represents the entire feasible setting of the DIP. This is the intention of the UD: to be assigned to serve as the CD as the criteria of uniformity qualities. A lower CD denotes higher uniformity, which also implies that the UD is easy to use for solving any problem anywhere that requires capturing the whole system with limited data because of constraints such as limited time, manpower, and budget. The UD can be developed personally by the user, especially if the number of parameters is not available from the UD-web (maximum: 29 parameters). Through employing CD considerations as criteria (fitness function) and then using this method for optimization designs such as cost functions or smaller-the-better targets, this solution can be highly popular in terms of DOE effectiveness and the corresponding optimization problems.

In comparison to a Taguchi design and an optimization design, the UD is more efficient in many parameters. Whereas a Taguchi design focuses on the level to determine the experiment, the UD does not. The UD can be easily developed by the user, whereas a Taguchi design cannot be, because the aim of a Taguchi design is to propose robustness settings from the appropriate combinations shown by the signal-to-noise ratios, and to analyse the mean effect of each parameter[26].

5 Conclusions

This study intended to determine the most favorable settings for DIP input for swing-standing up, and then stable robust in standing position against any outside interference. The model control system function was unknown, thus optimizing the parameters required an NN metamodel, which in turn was formed by training data retrieved by the UD, named UniNeuro. Furthermore, the HUDMOGA embedded in the UD was used to optimize the DIP′s parameter settings based on the metamodel as the fitness function. The HUDMOGA begins with initialization by the UD, then does roulette wheel selection, then partition crossover, and finally neighbor search for mutation. The Euclidean distance used for maintaining diversity in the solution is filtered by the Pareto front.

The results of this study show the nine most favorable parameter settings that influencing the DIP′s to swing and then stand up robustly in a time period determined by outside interference. In comparison to using a trial-and-error method, this method proved to be considerably faster because it only requires 25 data as the training data for the UniNeuro to develop the metamodel. This method is a novel method that can be performed in more than one setting, as concluded in previous studies[6, 9−11, 21]. Furthermore, it is more closely based on real situations than the simulation approach[7, 8, 14]. The UD is highly efficient for generating UniNeuro training data in DOE, because this search can be spread uniformly on the solution search space, and can also obtain results close to those of a full factorial design with an extremely low number of experiments. The accuracy of the model requires improvement by adaptive arrangement of the structures of the metamodel[3]. Objective criteria could also be affected by multiple constraints and various disturbances.

This research is appropriately useful for optimizing a system undetermined by such model information. This method can achieve the advanced prediction of a metamodel that deals with input and response for forming the linear approach function. The DIP model is generally nonlinear, but this approach uses linear regression. Therefore, it makes sense that the proposed method is worthwhile for application in real situations. Sometimes, the information of the system model is unknown, and the optimal solution is only close to itself, thus trial-and-error searches would be prohibitively expensive, but the proposed method can find a solution after a small number of experiments. For example, in this study, the method succeeded in optimizing and obtaining five DIP inputs of the undetermined model for the targets of standing and remaining stable in the upright position. Only 45 experiments were required for generating the advanced settings. If either the trial-and-error or full factorial design were used, the experiment time would increase dramatically because of the tiny level range in each parameter.

References

[1] A. Shukla, H. Karki. Application of robotics in onshore oil and gas industry-a review Part I. *Robotics and Autonomous Systems*, vol. 75, pp. 490–507, 2016.

[2] J. A. Anderson. *An Introduction to Neural Networks*, Cambridge, Mass, UK: MIT Press, 1995.

[3] P. P. Palmes, S. Usui. Robustness, evolvability, and optimality of evolutionary neural networks. *Biosystems*, vol. 82, no. 2, pp. 168–188, 2005.

[4] A. Medio, M. Lines. *Nonlinear Dynamics: A Primer*, Cambridge, UK: Cambridge University Press, 2001.

[5] T. K. Liu, C. H. Chen, Z. S. Li, J. H. Chou. Method of inequalities-based multiobjective genetic algorithm for optimizing a cart-double-pendulum system. *International Journal of Automation and Computing*, vol. 6, no. 1, pp. 29–37, 2009.

[6] T. K. Liu, Y. C. Shen, Z. S. Li. An application of multiobjective optimization genetic algorithm for cart-double-pendulum-system control. In *Proceedings of IEEE International Conference on Systems, Man and Cybernetics*, IEEE, Taipei, China, pp. 3292–3297, 2006.

[7] L. B. Prasad, B. Tyagi, H. O. Gupta. Optimal control of nonlinear inverted pendulum system using PID controller and LQR: Performance analysis without and with disturbance input. *International Journal of Automation and Computing*, vol. 11, no. 6, pp. 661–670, 2014.

[8] F. Gazdoš. Introducing a new tool for studying unstable systems. *International Journal of Automation and Computing*, vol. 11, no. 6, pp. 580–587, 2014.

[9] C. J. Ding, P. Duan, M. L. Zhang, Y. F. Zhang. Double inverted pendulum system control strategy based on fuzzy genetic algorithm. In *Proceedings of IEEE International Conference on Automation and Logistics*, IEEE, Shenyang, China, pp. 1318–1323, 2009.

[10] M. Nalavade, M. J. Bhagat, V. V. Patil. Balancing double inverted pendulum on cart by linearization technique. In *Proceedings of International Journal of Recent Technology and Engineering*, ISSN, pp. 2277–3878, 2014.

[11] M. J. Mahmoodabadi, S. A. Mostaghim, A. Bagheri, N. Nariman-Zadeh. Pareto optimal design of the decoupled sliding mode controller for an inverted pendulum system and its stability simulation via Java programming. *Mathematical and Computer Modelling*, vol. 57, no. 5–6, pp. 1070–1082, 2013.

[12] Z. W. Ping, J. Huang. Approximate output regulation of spherical inverted pendulum by neural network control. *Neurocomputing*, vol. 85, pp. 38–44, 2012.

[13] Z. W. Ping. Tracking problems of a spherical inverted pendulum via neural network enhanced design. *Neurocomputing*, vol. 106, pp. 137–147, 2013.

[14] K. U. P. Lakshmi. Design of robust energy control for cart-inverted pendulum. *International Journal of Engineering and Technology*, vol. 4, no. 1, pp. 66–76, 2007.

[15] D. C. Montgomery. *Design and Analysis of Experiments*, 7th ed., Hoboken, USA: John Wiley and Sons, 2009.

[16] K. Yang, B. S. El-Haik. *Design for Six Sigma*, New York, USA: McGraw Hill, 2003.

[17] K. T. Fang, R. Z. Li, A. Sudjianto. *Design and Modeling for Computer Experiments*, Boca Raton, USA: CRC Press, 2005.

[18] K. T. Fang, D. K. J. Lin. Ch. 4. Uniform experimental designs and their applications in industry. *Handbook of Statistics*, vol. 22, pp. 131–170, 2003.

[19] G. Dreyfus. *Neural Networks: Methodology and Applications*, Heidelberg, Germany: Springer, 2005.

[20] D. H. Al-Janan, T. K. Liu. Path optimization of CNC PCB drilling using hybrid Taguchi genetic algorithm. *Kybernetes*, vol. 45, no. 1, pp. 107–125, 2016.

[21] T. K. Liu, T. Ishihara. Multi-objective genetic algorithms for the method of inequalities. *Control Systems Design*, V. Zakian, Ed., London, UK: Springer, pp. 231–248, 2005.

[22] A. Konak, D. W. Coit, A. E. Smith. Multi-objective optimization using genetic algorithms: A tutorial. *Reliability Engineering & System Safety*, vol. 91, no. 9, pp. 992–1007, 2006.

[23] X. Q. Li, D. Wang, Z. M. Fu. Adaptive NN dynamic surface control for a class of uncertain non-affine pure-feedback systems with unknown time-delay. *International Journal of Automation and Computing*, vol. 13, no. 3, pp. 268–276, 2016.

[24] X. J. Wu, X. L. Wu, X. Y. Luo. Adaptive neural network dynamic surface control for a class of nonlinear systems with uncertain time delays. *International Journal of Automation and Computing*, vol. 13, no. 4, pp. 409–416, 2016.

[25] K. Owa, S. Sharma, R. Sutton. A wavelet neural network based non-linear model predictive controller for a multi-variable coupled tank system. *International Journal of Automation and Computing*, vol. 12, no. 2, pp. 156–170, 2015.

[26] W. Y. Fowlkes, C. M. Creveling. *Engineering Methods for Robust Product Design*, Reading, Mass, USA: Addison-Wesley, 1995.

Dony Hidayat Al-Janan received the B. Sc. degree in mechanical engineering from the Muhammadiyah University of Surakarta, Indonesia in 2001, received the M. Sc. degree in mechanical engineering from Gadjahmada University, Indonesia in 2004. From 2006 to 2013, he has been also a lecturer of Computer Programming of Engineering Faculty, Semarang State University. Since 2013, he is the Ph. D. degree candidate at Institute of Engineering Science and Technology, National Kaohsiung First University of Science and Technology, Taiwan, China, supported by Indonesian Directorate General of Higher Education (DIKTI) scholarship in scheme BPPLN DIKTI 3+1.

His research interests include developing a metamodel and system optimization design.

E-mail: u0215917@nkfust.edu.tw
ORCID iD: 0000-0002-8485-8420

Hao-Chin Chang received the B. Sc. and M. Sc. degrees in marine engineering from National Kaohsiung Marine University, Taiwan, China in 2009 and 2011, respectively. He received Ph. D. degree from the Institute of Engineering Science and Technology, National Kaohsiung First University of Science and Technology, Taiwan, China in 2015.

His research interests include artificial intelligence and applications of multiobjective optimization genetic algorithms.

E-mail: h5201141@gmail.com

Yeh-Peng Chen received the B. Sc. degree in information engineering from I-Shou University, Taiwan, China in 1996, and the M. Sc. and Ph. D. degrees in information management science and engineering science and technology from the National Kaohsiung First University of Science and Technology, Taiwan, China in 2003 and 2015, respectively.

His research interests include artificial intelligence, applications of multiobjective optimization genetic algorithms, power systems, and data mining.

E-mail: tom.b0519@msa.hinet.net

Tung-Kuan Liu received the B. Sc. degree in mechanical engineering from the National Akita University, Japan in 1992, and the M. Sc. and Ph. D. degrees in mechanical engineering and information science from the National Tohoku University, Japan in 1994 and 1997, respectively. He is a professor in Department of Mechanical and Automation Engineering and Graduate Program of Industrial Design, National Kaohsiung First University of Science and Technology, Taiwan, China since February 2011. During October 1997 to July 1999, he was a senior manager with the Institute of Information Industry, Taiwan, China. From August 1999 to July 2002, he was also an assistant professor with the Department of Marketing and Distribution Management, National Kaohsiung First University of Science and Technology, Taiwan, China.

His research and teaching interests include artificial intelligence, applications of multi objective optimization genetic algorithms, and integrated manufacturing and business systems.

E-mail: tkliu@nkfust.edu.tw (Corresponding author)
ORCID iD: 0000-0002-3581-574X