

# Application of Intensified Current Search to Multiobjective PID Controller Optimization

**Auttarat Nawikavatan and Satean Tunyasrirut**

Department of Electrical Engineering, Faculty of Engineering, Pathumwan Institute of Technology,  
Bangkok, 10330, Thailand

E-mail: {auttarat@hotmail.com, satean@pit.ac.th}

**Deacha Puangdownreong\***

Department of Electrical Engineering, Graduate School, South-East Asia University, Bangkok, 10160, Thailand

\*Corresponding Author: E-mail: deachap@sau.ac.th

**Abstract**—The intelligent control system design has been changed from the conventional approach to the optimization framework solved by efficient metaheuristics. The intensified current search (ICS) has been recently proposed as one of the most powerful metaheuristics for solving optimization problems. The ICS, the latest modified version of the conventional current search (CS), possesses the memory list (ML) regarded as the exploration strategy and the adaptive radius (AR) and adaptive neighborhood (AN) mechanisms regarded as the exploitation strategy. The ML is used to escape from local entrapment caused by any local solution, while both AR and AN mechanisms are conducted to speed up the search process. In this paper, the application of the ICS to multiobjective PID controller design optimization for the three-phase induction motor ( $3\phi$ -IM) speed control system is proposed. Algorithms of the ICS and its performance evaluation against multiobjective functions are presented. As simulation results, the ICS can provide very satisfactory solutions for all test functions and the  $3\phi$ -IM control application. Moreover, the simulation results of motor control application are confirmed by the experimental results based on dSPACE technology.

**Index Terms**—Intensified Current Search, Multiobjective PID Controller, Metaheuristics, Control System Optimization.

## I. INTRODUCTION

Over two decades, the intelligent control system design has been changed from the conventional paradigm to multiobjective design optimization framework [1]. Such the multiobjective optimization problems can be effectively solved by powerful metaheuristic optimization search techniques. By literatures, many metaheuristics are consecutively developed and launched to perform their effectiveness. Metaheuristics can be classified into population-based and single-solution (trajectory-) based [2,3,4,5]. The most powerful metaheuristics must has at least two major properties, i.e. exploration (or

diversification) to generate diverse solutions to explore the search space on the global scale and exploitation (or intensification) to focus on the search in a local region by exploiting the information to reach the best local solution within this region [2,3,4,5]. Among them, the well-known population-based metaheuristics algorithms are such as genetic algorithm (GA) [6], ant colony optimization (ACO) [7], artificial bee colony (ABC) [8], differential evolution (DE) [9], particle swarm optimization (PSO) [10], harmony search (HS) [11], firefly search (FS) [12], cuckoo search (CuS) [13] and bat-inspired search (BS) [14], whereas the single-solution based metaheuristics algorithms are such as simulated annealing (SA) [15], tabu search (TS) [16] and current search (CS) [17].

In 2012, the current search (CS) was firstly proposed [17,18,19] as one of the single-solution based metaheuristics based on the principle of an electric current behavior in the electric circuits and networks. The CS was successfully applied to control system [18,19,20,21] and analog filter design [22] applications. However, the search process of the CS may be trapped by any local solution. In addition, the search time consumed by the CS is depended on the numbers of search directions. In 2014, the modified version of the CS named the adaptive current search (ACS) was proposed [23]. The ACS possesses the memory list (ML) and the adaptive radius (AR) mechanism to speed up the search process. The ACS was satisfactory applied to assembly line balancing problems [23,24] and transportation problems [25]. Although both CS and ACS performed good performance, their applications are limited by single-objective optimization problems.

Generally, real-world engineering design problems often consist of many objectives which conflict each other [2,3,4,5]. This leads the multiobjective problems much more difficult and complex than single-objective ones. The multiobjective problem possesses multiple optimal solutions forming the so-called Pareto front [2,3,4,5]. The challenge is how to perform the smooth Pareto front containing a set of optimal solutions for all objective functions. In 2014, the intensified current search (ICS) was proposed [26,27,28] as the latest modified version of the CS. The ICS possesses the ML

regarded as the exploration strategy and the AR and the adaptive neighborhood (AN) mechanisms regarded as the exploitation strategy. Performance of the ICS has been evaluated against several single-objective test functions [26]. It was found in [26] that the ICS can provide superior results to CS and ACS. In this paper, the ICS is applied to multiobjective PID design optimization for the three-phase induction motor (3 $\phi$ IM) speed control system. Details of the ICS algorithm are revised for multiobjective purpose. The performance of the ICS will be evaluated against the standard multiobjective test functions. The simulation results of 3 $\phi$ IM control application will be confirmed by the experimental results.

This paper consists of six sections. After an introduction shown in section I, the related works of multiobjective optimization are presented in section II. Details of ICS algorithms are illustrated in section III. The performance study of the ICS via the standard multiobjective test functions is described in section IV. Application of the ICS to multiobjective PID controller optimization for the 3 $\phi$ IM speed control system is provided in section V, while conclusions are given in section VI.

## II. RELATED WORKS OF MULTI-OBJECTIVE OPTIMIZATION

Regarding to the optimization context [2,3,4,5], multiobjective optimization problem can be expressed in (1), where  $f(\mathbf{x})$  is the multiobjective function consisting of  $f_1(\mathbf{x}), \dots, f_n(\mathbf{x})$ ,  $n \geq 2$ ,  $g_j(\mathbf{x})$ ,  $j = 1, 2, \dots, m$ , is the inequality constraints and  $h_k(\mathbf{x})$ ,  $k = 1, 2, \dots, p$ , is the equality constraints. The optimal solutions,  $\mathbf{x}^*$ , are ones can make  $f(\mathbf{x})$  minimum and make both  $g_j(\mathbf{x})$  and  $h_k(\mathbf{x})$  satisfied.

$$\begin{cases} \min f(\mathbf{x}) = \{f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_n(\mathbf{x})\} \\ \text{subject to } g_j(\mathbf{x}) \leq 0, \quad j = 1, \dots, m \\ h_k(\mathbf{x}) = 0, \quad k = 1, \dots, p \end{cases} \quad (1)$$

All solutions  $\mathbf{x}^*$  are called a non-dominated (optimal) solutions if no solutions can be found that dominates them. For a given multiobjective optimization problem, the Pareto optimal set is defined as the set containing all non-dominated solutions. Finally, the Pareto optimal set will be used to perform the Pareto front of a given multiobjective optimization problem of interest.

Engineering design problems can be considered as the optimization problems divided into single-objective or multiobjective problems [2,3,4,5]. For single-objective, an optimization tends to minimize (or maximize) only one objective such as minimize loss or maximize profit. For multiobjective, it tends to minimize (or maximize) several objectives such as minimize loss and minimize cost. In fact, both loss and cost are trade-off. The less the loss, the higher the cost, and vice versa. Many real-world engineering design problems consist of many objectives which are conflict each other [2,3,4,5]. This leads the multiobjective problems much more difficult and complex than single-objective ones such as a problem of

upgrading to the next generation wireless network (NGWN) [29] and structural engineering problems [30].

The multiobjective optimization problem possesses multiple optimal solutions forming the Pareto front. The challenge is how to perform the smooth Pareto front containing a set of optimal solutions for all objective functions. By literatures, conventional optimization methods, such as utility function method, global criterion method, bounded objective function method and goal attainment method [30,31,32], often face difficulties for solving multiobjective problems. One of the alternative approaches developed to solve multiobjective problems is the metaheuristics approach [2,3,4,5]. By literatures, the efficient metaheuristics has been consecutively launched for multiobjective optimization. For example, the vector evaluated genetic algorithm (VEGA) [33] is based on the GA with non-dominated solution vector, the non-dominated sorting genetic algorithm II (NSGA-II) [34] is based on the GA with multiple layers of non-dominated solution set, the differential evolution for multiobjective optimization (DEMO) [35] is developed from the conventional DE, the multiobjective cuckoo search (MOCS) [36] is developed from the conventional CS and the multiobjective multipath adaptive tabu search (mMATS) [37] is based on the adaptive tabu search (ATS). Some of multiobjective metaheuristics have been successfully applied to solve real-world engineering problems such as bicycle stem design by NSGA-II [38], welded beam design and disc brake design by MOCS [36] and PID controller design for the automatic voltage regulator (AVR) system by mMATS [37]. In this work, VEGA, NSGA-II, DEMO, MOCS and mMATS will be conducted for performance comparison with the proposed ICS algorithms for multiobjective optimization problems.

## III. INTENSIFIED CURRENT SEARCH ALGORITHMS

The intensified current search (ICS) was firstly proposed in 2014 [26,27,28]. The ICS is latest modified version of the CS based on the principle of current divider in electric circuits and networks. In its algorithm, the ICS possesses the ML, AR and AN mechanisms. The ML regarded as the exploration strategy is used to store the ranked initial solutions at the beginning of search process, record the solution found along each search direction, and contain all local solutions found at the end of each search direction. The ML is also applied to escape the local entrapments caused by local optima. The AR and AN mechanisms regarded as the exploitation strategy are together conducted to speed up the search process.

Algorithms of the ICS can be described by the pseudo code as shown in Fig. 1, while some movements of the ICS over 2D-search space can be visualized by Fig. 2. Once the ICS is applied to solve multiobjective optimization problems, the ICS algorithms need to be modified. The multiobjective function  $f(\mathbf{x})$ , consisting of  $f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_n(\mathbf{x})$ , as stated in (1) will be simultaneously minimized according to its inequality  $g_j(\mathbf{x}) \leq 0$  and equality  $h_k(\mathbf{x}) = 0$  constraints. In each search iteration, the optimal solution will be evaluated via  $f(\mathbf{x})$ . If the optimal

solution found is a non-dominated solution (there is no exist solutions remaining that are better than it), it will be sorted and stored into the Pareto optimal set  $P^*$ . After the search process stopped, all solutions stored in  $P^*$  will be conducted to perform the Pareto front  $PF^*$ . Finally, every solution contained on the  $PF^*$  are the optimal solutions of the multiobjective problem of interest. The modified algorithm of the ICS for multiobjective optimization problem is represented by the pseudo code as shown in Fig. 3.

IV. PERFORMANCE STUDY

To study its performance, the ICS is evaluated against three standard multiobjective test functions [39,40]. The first function is the convex front ZDT1 as stated in (2) where  $d$  is the number of dimensions. The second one is the concave front ZDT2 as expressed in (3). Finally, the third is the discontinuous front ZDT3 as given in (4). The constraints  $g$  and control variables  $x_i$  in (3) and (4) are the same as appeared in (2).

$$\left. \begin{aligned} \text{ZDT1: } f_1(x) &= x_1, \\ f_2(x) &= g(1 - \sqrt{f_1/g}), \\ g &= 1 + \frac{9 \sum_{i=2}^d x_i}{d-1}, \\ x_i &\in [0, 1], \quad i = 1, \dots, 30 \end{aligned} \right\} \quad (2)$$

$$\left. \begin{aligned} \text{ZDT2: } f_1(x) &= x_1, \\ f_2(x) &= g \left( 1 - \frac{f_1}{g} \right)^2 \end{aligned} \right\} \quad (3)$$

**Initialized:**

- Objective function  $f(x)$ ,  $\mathbf{x}=(x_1, \dots, x_d)^T$ ,
- Search space  $\Omega$ ,
- Memory list ML ( $\Psi$ ,  $\Gamma_k$ , and  $\Xi$ ),
- Maximum allowance of solution cycling  $j_{\max}$ ,
- Number of initial solutions  $N$ ,
- Number of neighborhood members  $n$ ,
- Search radius  $R$ ,  $k=j=1$ .
- Random initial solution  $X_i$  within  $\Omega$ .
- Evaluate  $f(X_i)$  then rank  $X_i$  and store in  $\Psi$ .
- Let  $\mathbf{x}_0=X_k$  as selected initial solution.
- $X_{\text{global}}=X_{\text{local}}=\mathbf{x}_0$ .

**while** ( $k \leq N$  or termination criteria: TC);

**while** ( $j \leq j_{\max}$ );

Random  $x_i$  around  $\mathbf{x}_0$  within  $R$ .

Evaluate  $f(x_i)$  and set the best one as  $\mathbf{x}^*$ .

**if**  $f(\mathbf{x}^*) < f(\mathbf{x}_0)$ ;

Keep  $\mathbf{x}_0$  into  $\Gamma_k$ , update  $\mathbf{x}_0=\mathbf{x}^*$  and set  $j=1$ .

**else**

Keep  $\mathbf{x}^*$  into  $\Gamma_k$  and update  $j=j+1$ .

**end**

Activate AR by  $R=\rho R$ ,  $0 < \rho < 1$ .

Invoke AN by  $n=an$ ,  $0 < a < 1$ .

**end**

Update  $X_{\text{local}}=\mathbf{x}_0$ .

Keep  $X_{\text{global}}$  into  $\Xi$ .

**if**  $f(X_{\text{local}}) < f(X_{\text{global}})$ ;

Update  $X_{\text{global}}=X_{\text{local}}$ .

**end**

Update  $k=k+1$  and set  $j=1$ .

Let  $\mathbf{x}_0=X_k$  as selected initial solution.

**end**

Fig.1. Pseudo code of the ICS algorithm.

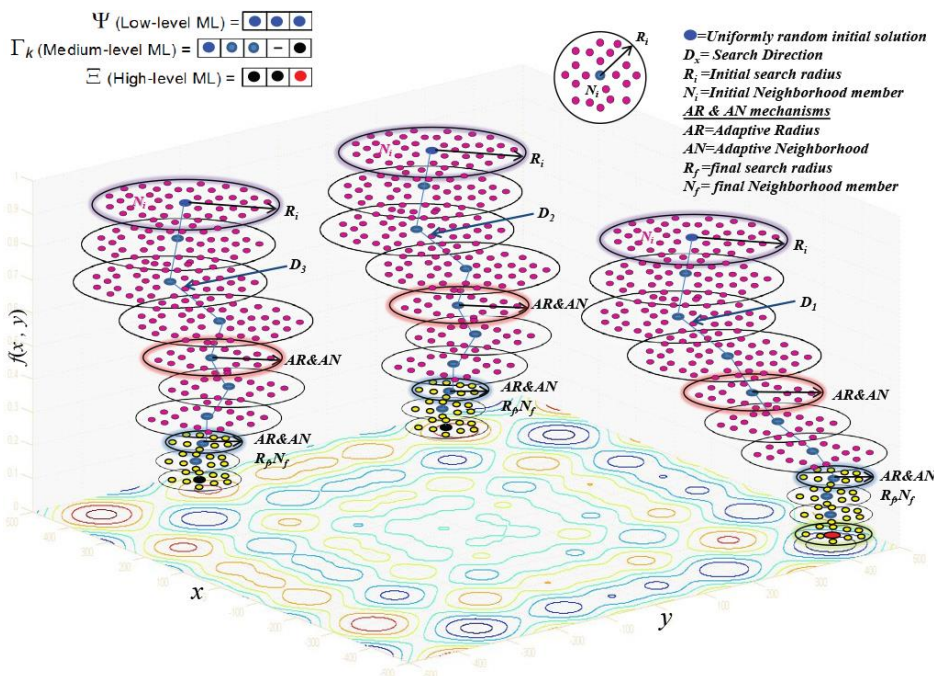


Fig.2. Some movements of the ICS over 2D-search space.

---

**Initialized:**

- Multiobjective function  $f(\mathbf{x}) = \{f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_n(\mathbf{x})\}$ ,  
 $\mathbf{x} = (x_1, \dots, x_d)^T$ ,
- Pareto optimal set  $P^*$ ,
- Pareto front  $PF^*$ ,
- Search space  $\Omega$ ,
- Memory list ML ( $\Psi$ ,  $\Gamma_k$ , and  $\Xi$ ),
- Maximum allowance of solution cycling  $j_{\max}$ ,
- Number of initial solutions  $N$ ,
- Number of neighborhood members  $n$ ,
- Search radius  $R$ ,  $k=j=1$ .
- Random initial solution  $X_i$  within  $\Omega$ .
- Evaluate  $f(X_i)$  then rank  $X_i$  and store in  $\Psi$ .
- Let  $x_0 = X_k$  as selected initial solution.
- $X_{\text{global}} = X_{\text{local}} = x_0$ .

**while** ( $k \leq N$  or termination criteria: TC);

**while** ( $j \leq j_{\max}$ );

Random  $x_i$  around  $x_0$  within  $R$ .

Evaluate  $f(x_i)$  and set the best one as  $x^*$ .

**if** ( $x^* = \text{non-dominated solution}$ );

Sort and store  $x^*$  into  $P^*$ .

**end**

**if**  $f(x^*) < f(x_0)$ ;

Keep  $x_0$  into  $\Gamma_k$ , update  $x_0 = x^*$  and set  $j=1$ .

**else**

Keep  $x^*$  into  $\Gamma_k$  and update  $j=j+1$ .

**end**

Activate AR by  $R = \rho R$ ,  $0 < \rho < 1$ .

Invoke AN by  $n = \alpha n$ ,  $0 < \alpha < 1$ .

**end**

Update  $X_{\text{local}} = x_0$ .

Keep  $X_{\text{global}}$  into  $\Xi$ .

**if**  $f(X_{\text{local}}) < f(X_{\text{global}})$ ;

Update  $X_{\text{global}} = X_{\text{local}}$ .

**end**

Update  $k=k+1$  and set  $j=1$ .

Let  $x_0 = X_k$  as selected initial solution.

Sort the current Pareto optima.

**End**

Perform the Pareto front  $PF^*$ .

Report Pareto optimal solutions.

---

Fig.3. Pseudo code of the modified ICS algorithm for multiobjective optimization problem.

$$\left. \begin{aligned} \text{ZDT3: } f_1(x) &= x_1, \\ f_2(x) &= g \left( 1 - \sqrt{\frac{f_1}{g}} - \frac{f_1}{g} \sin(10\pi f_1) \right) \end{aligned} \right\} \quad (4)$$

$$E_f = \|PF_e - PF_t\| = \sum_{j=1}^N (PF_e^j - PF_t^j)^2 \quad (5)$$

The results obtained by the ICS will be compared with those obtained by VEGA, NSGA-II, DEMO, MOCS and mMATS. In comparison, the error  $E_f$  between the estimated Pareto front  $PF_e$  and its correspondingly true front  $PF_t$  is evaluated via the formulation stated in (5), where  $N$  is the number of sorted solutions.

To gain the best performance, the appropriate ICS's search parameters need to be set. The ICS algorithms

were coded by MATLAB. The search parameters of the ICS consist of number of initial solutions  $N$ , number of neighborhood member  $n$ , number of solution cycling  $j_{\max}$ , initial search radius  $R$ , AR and AN mechanisms. Different ranges of these parameters are investigated. A fixed number of iterations of 2,000 is set as the termination criteria (TC). By varying  $N = 10, 20, 30, 40, 50, \dots, 100$ ,  $n = 10, 20, 30, 40, 50, \dots, 100$ ,  $j_{\max} = 5, 10, 20, \dots, 50$ ,  $R = 5\%, 10\%, 15\%, \dots, 50\%$  of search space, AR and AN = 2, 3, 4, 5 states, it was found that the best parameters for most functions are  $N = 50$ ,  $n = 60$ ,  $j_{\max} = 10$ ,  $R = 20\%$  of search space, AR = 2 states {(i) at 750<sup>th</sup> iteration, adjusting  $R = 0.1R$  and (ii) at 1,500<sup>th</sup> iteration, adjusting  $R = 0.01R$ } and AN = 2 states {(i) at 750<sup>th</sup> iteration, adjusting  $n = 40$  and (ii) at 1,500<sup>th</sup> iteration, adjusting  $n = 20$ }. These best parameters will be used for all problems in this paper, while the search parameters of VEGA, NSGA-II, DEMO, MOCS and mMATS and will be set as the recommendations according to their corresponding references in [33], [34], [35], [36] and [37], respectively. With the identical TC, results obtained by the ICS and other selected algorithms are summarized in Table 1. The Pareto fronts obtained by the ICS and the true fronts of functions ZDT1 – ZDT3 are depicted in Fig. 4 – Fig. 6, respectively. Referring to Table 1, the ICS shows superior results to other algorithms with less error  $E_f$ . Moreover, Fig. 4 – Fig. 6 reveal that the ICS can provide the smooth Pareto fronts very coincide with the true fronts of each standard multiobjective test function.

Table 1. Results of three standard multiobjective test functions

Methods	Error $E_f$		
	ZDT1	ZDT2	ZDT3
VEGA	2.79e-02	2.37e-03	3.29e-01
NSGA-II	3.33e-02	7.24e-02	1.14e-01
DEMO	2.08e-03	7.55e-04	2.18e-03
MOCS	1.27e-04	2.23e-04	2.88e-04
mMATS	1.14e-04	2.15e-04	2.47e-04
<b>ICS</b>	<b>1.02e-04</b>	<b>2.01e-04</b>	<b>2.18e-04</b>

## V. MULTI-OBJECTIVE PID OPTIMIZATION

The ICS will be applied to design the optimal PID controller for the  $3\phi$ IM speed control system based on the multiobjective optimization. Referring to control context, the PID feedback control loop is represented by the block diagram in Fig. 7, where  $R(s)$  is the reference input,  $C(s)$  is the controlled output,  $E(s)$  is the error signal between  $R(s)$  and  $C(s)$ ,  $U(s)$  is the control signal,  $D(s)$  is disturbance signal,  $G_p(s)$  and  $G_c(s)$  are the plant and the controller transfer functions, respectively. The PID controller located at the forward path receives  $E(s)$  and generates  $U(s)$  to control  $C(s)$  and regulate  $D(s)$  referring to  $R(s)$ . The time-domain control signal (output of PID controller) is stated in (6), while the  $s$ -domain transfer function of the PID controller is expressed in (7), where  $K_p$ ,  $K_i$  and  $K_d$  are the proportional, integral and derivative gains, respectively [41,42]. Therefore, the closed loop

transfer function with PID controller is given in (8). Once consider the 3ϕ-IM as the plant in Fig. 7,  $R(s)$  implies the reference speed and  $C(s)$  stands for the actual speed.

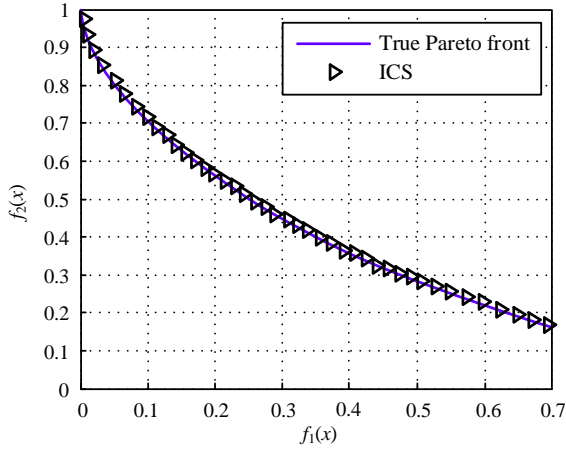


Fig.4. Pareto front of ZDT1 (convex front).

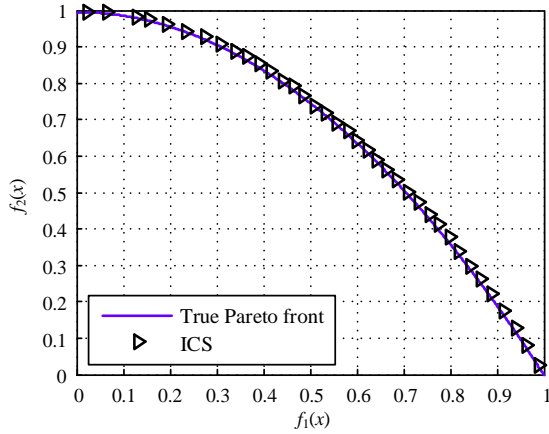


Fig.5. Pareto front of ZDT2 (concave front).

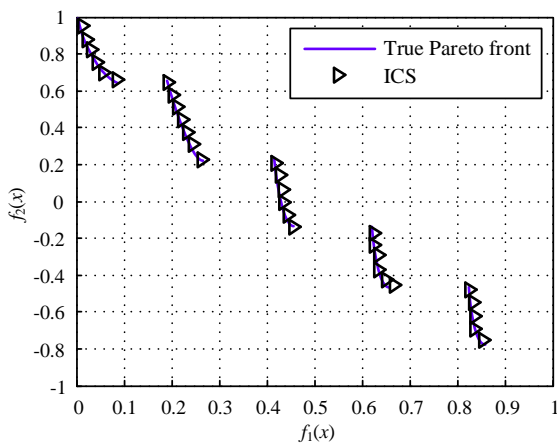


Fig.6. Pareto front of ZDT3 (discontinuous front).

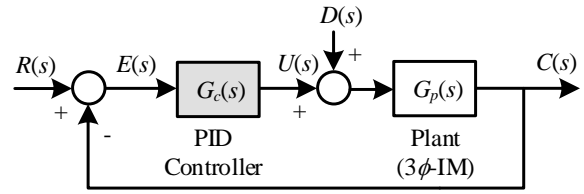


Fig.7. PID control loop.

$$u(t) = K_p e(t) + K_i \int_0^t e(t) dt + K_d \frac{de(t)}{dt} \quad (6)$$

$$G_c(s) = K_p + \frac{K_i}{s} + K_d s \quad (7)$$

$$\frac{C(s)}{R(s)} = \frac{\left( K_p + \frac{K_i}{s} + K_d s \right) G_p(s)}{1 + \left( K_p + \frac{K_i}{s} + K_d s \right) G_p(s)} \quad (8)$$

The plant model  $G_p(s)$  of the 3ϕ-IM needs to be developed. From our previous work [27,28], the plant model  $G_p(s)$  of the 3ϕ-IM has been identified by using a 0.37 kW, 1400 rpm, 50 Hz, 4-pole, delta-connected, squirrel-cage 3ϕ-IM. Such the motor has been tested to record its speed dynamics at 800, 1,000 and 1,200 rpm. The third-order transfer function has been identified as stated in (9). Good agreement between the model plot and the experimental speed can be observed in Fig. 8. The plant model  $G_p(s)$  in (9) will be used as the plant,  $G_p(s)$ , in Fig. 7.

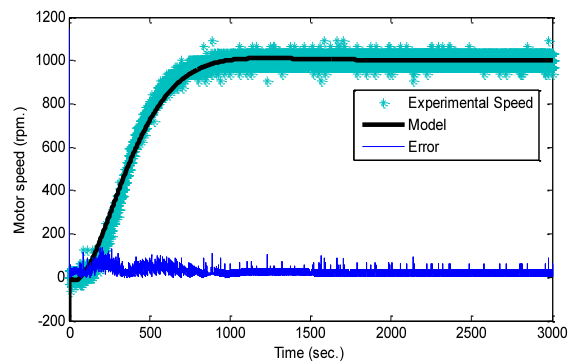


Fig.8. Model plot against experimental speed.

$$G_p(s) = \frac{0.02875}{1.028s^3 + 1240s^2 + 9.773s + 0.2878} \quad (9)$$

The multiobjective PID design optimization problem is classified to optimize the parameters  $K_p$ ,  $K_i$  and  $K_d$  in order to obtain the satisfactory responses. Applying the ICS to design the PID controller of the  $3\phi$ -IM speed control system can be represented by the block diagram in Fig. 9. Based on the practical requirements, the rise time ( $t_r$ ) and the maximum overshoot ( $M_p$ ) are selected to be the multiobjective functions because they conflict each other significantly. The  $t_r$  and  $M_p$  are set as  $f_1(x)$  and  $f_2(x)$ , respectively. The multiobjective PID design optimization problem can be formulated as expressed (10). The multiobjective  $f(x)$  in Fig. 9 will be fed back to the ICS tuning block to be minimized in order to find the optimal PID controller's parameters which satisfy to its corresponding constraints stated in (10).

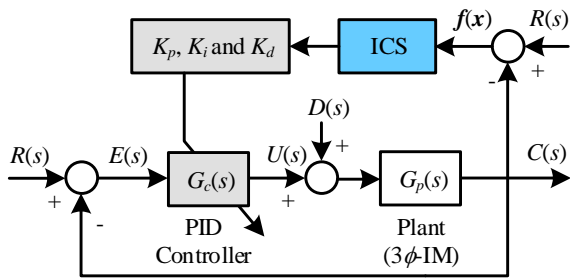


Fig.9. ICS-based multiobjective PID optimization.

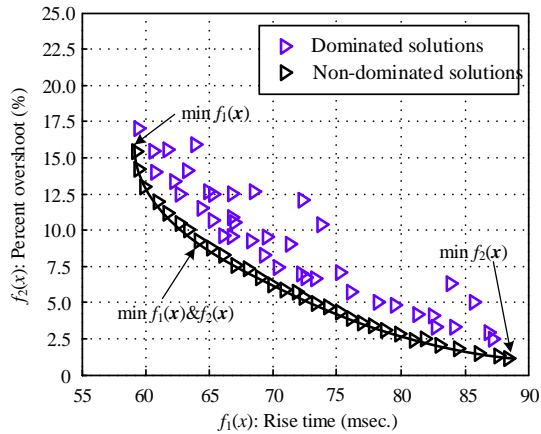
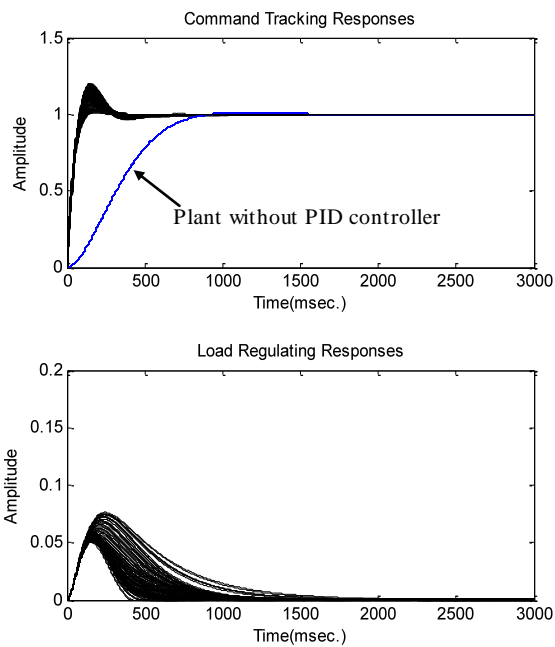
$$\begin{aligned}
 &\text{Minimize } f(x) = \{f_1(x), f_2(x)\}, \\
 &\quad f_1(x) = t_r, \\
 &\quad f_2(x) = M_p, \\
 &\text{subject to } \left. \begin{aligned}
 &0 \leq M_p \leq 20\%, \quad 5 \leq K_p \leq 15, \\
 &0 \leq K_i \leq 0.1, \quad 500 \leq K_d \leq 1,000
 \end{aligned} \right\} \quad (10)
 \end{aligned}$$

A. Simulation Results

In this sub-section, the ICS coded by MATLAB with the same values of the search parameters and TC set in the previous section are conducted to the multiobjective PID design optimization problem. Once applying the ICS with 40 trials of optimization process to find the best solution, 23 sets of optimal PID controllers are successfully obtained and summarized in Table 2, where  $t_s$  is settling time,  $E_{ss}$  is steady-state error and  $t_{reg}$  is regulating time, respectively. The Pareto front plotted between  $f_1(x)$  and  $f_2(x)$  belonging to those non-dominated solutions is depicted in Fig. 10. Simulation results consisting of the command tracking and load regulating responses of the  $3\phi$ -IM speed control system with PID controller designed the ICS are depicted in Fig. 11. Referring to Table 2 and Fig. 11, it was found that the optimal PID controller's parameters obtained by the ICS and their corresponding responses are very satisfactory according to the design constraints defined in (10). This can be noticed that the ICS can be successfully applied to multiobjective PID controller design optimization problem for  $3\phi$ -IM speed control system. Referring to Fig. 10 and Table 2, the non-dominated solutions shown in Fig. 10 are the optimal solution. The Pareto front in Fig. 10 performs tread-off phenomenal between the  $f_1(x)$  and  $f_2(x)$ . This leads the user can select these solutions freely. For example, once the user prefer the minimum rise time, the solution  $\min f_1(x)$  in Fig. 10 (PID entry-1 in Table 2) should be selected. When the user prefer the minimum overshoot, the solution  $\min f_2(x)$  in Fig. 10 (PID entry-23 in Table 2) should be collected. Finally, if the user prefer both minimum rise time and minimum overshoot, the compromised solution ( $\min f_1(x) \& f_2(x)$ ) in Fig. 10 (PID entry-8 in Table 2) should be considered.

Table 2. Optimal PID controllers obtained by the ICS and their corresponding responses

Entry	PID controllers			System responses					Remark
	$K_p$	$K_i$	$K_d$	$t_r$ (ms.)	$t_s$ (ms.)	$M_p$ (%)	$E_{ss}$ (%)	$t_{reg}$ (s.)	
0.	---Plant without PID controller---			782.25	924.42	0.5402	0.0000		
1.	<b>14.9993</b>	<b>0.0899</b>	<b>989.8136</b>	<b>59.0215</b>	<b>460.5414</b>	<b>15.5009</b>	<b>0.0000</b>	<b>2.0612</b>	$\min f_1(x)$
2.	14.9992	0.0787	966.9804	60.1024	311.1247	14.5065	0.0000	2.0212	
3.	14.9992	0.0597	981.4458	61.0114	300.3626	12.5008	0.0000	1.7492	
4.	14.9390	0.0415	999.9787	62.8457	267.3547	10.5012	0.0000	1.5448	
5.	14.9739	0.0392	956.4697	63.1247	262.0210	9.4072	0.0000	1.3959	
6.	14.4945	0.0392	991.3053	64.3654	267.0017	8.8130	0.0000	1.1803	
7.	13.8479	0.0399	999.9712	65.2147	278.7441	7.5013	0.0000	1.0045	
8.	<b>13.6428</b>	<b>0.0352</b>	<b>974.7195</b>	<b>66.2211</b>	<b>266.1658</b>	<b>7.0416</b>	<b>0.0000</b>	<b>0.9148</b>	$\min f_1(x) \& f_2(x)$
9.	13.3999	0.0357	999.9725	67.2047	269.9512	6.5049	0.0000	0.9076	
10.	12.9967	0.0352	978.1313	68.3710	275.2351	6.4012	0.0000	0.9005	
11.	12.6486	0.0361	999.9906	69.9854	282.3517	6.0102	0.0000	0.8803	
12.	12.3733	0.0344	999.9089	70.0924	278.2145	5.7019	0.0000	0.8678	
13.	12.1649	0.0337	982.3622	71.1014	280.6985	5.5161	0.0000	0.8357	
14.	11.8031	0.0340	999.9453	72.3033	285.2145	5.2034	0.0000	0.8016	
15.	11.6738	0.0327	976.9918	73.2014	283.2278	5.0391	0.0000	0.7852	
16.	11.1702	0.0313	999.9455	74.4748	275.6214	4.4626	0.0000	0.7545	
17.	10.9003	0.0305	976.4722	76.8426	279.2154	4.0216	0.0000	0.7204	
18.	10.7115	0.0296	986.3490	77.6140	269.6987	3.5068	0.0000	0.6971	
19.	10.3069	0.0291	989.2462	78.6874	266.6536	3.0316	0.0000	0.6705	
20.	10.0011	0.0255	999.7894	80.3251	119.6610	2.7933	0.0000	0.6527	
21.	10.0252	0.0234	990.7593	81.0012	120.1024	2.2349	0.0000	0.6126	
22.	10.0327	0.0229	941.0724	82.7140	199.9578	2.0089	0.0000	0.5979	
23.	<b>10.0015</b>	<b>0.0202</b>	<b>930.1657</b>	<b>83.0124</b>	<b>121.0054</b>	<b>1.4993</b>	<b>0.0000</b>	<b>0.5901</b>	$\min f_2(x)$

Fig.10. Pareto front between  $f_1(x)$  and  $f_2(x)$ .Fig.11. Simulation results: command tracking and load regulating responses of  $3\phi$ IM with PID controller designed by ICS.

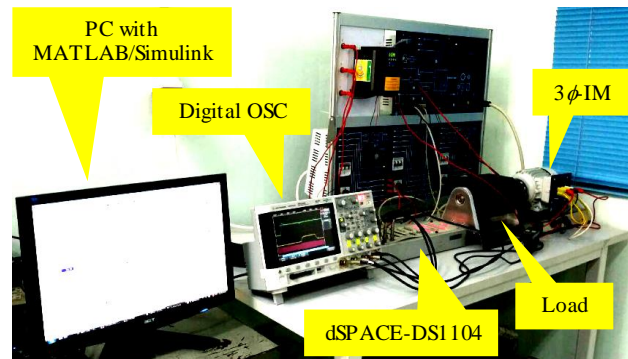
The simulation results of  $3\phi$ IM speed controlled system with multiobjective PID controller designed by the ICS shown in Fig. 11 will be confirmed by the experimental results as described in next sub-section.

### B. Experimental Results

To confirm the simulation results, the experimentation of  $3\phi$ IM speed control system need to be done. The PID controllers designed by the ICS are implemented by the dSPACE-DS1104 R&D controller board [43,44] shown in Fig. 12 as the real-time embedded control system. The dSPACE-DS1104 controller board consists of the central processing unit PPC603e, 250 MHz as the main CPU, the digital signal processor TMS320F240 as the slave DSP, 12- and 16-bit A/D and D/A convertors and PWM-invertor drive. This dSPACE-DS1104 R&D controller board is designed to interface with MATLAB/Simulink running on PC.



Fig.12. The dSPACE-DS1104 R&amp;D controller board [44].

Fig.13. Testing rig of  $3\phi$ IM speed control system.

The developed  $3\phi$ IM system is performed as the V/f control manner, while the tachogenerator is employed as the speed sensor. With this implementation, the developed  $3\phi$ IM speed control system can be communicated and handled via MATLAB/Simulink on PC. The  $3\phi$ IM speed control system is developed as the testing rig as displayed in Fig. 13. In this work, the five selected cases, i.e. PID controller entry-1 ( $\min f_1(x)$ ), entry-4, entry-8 ( $\min f_1(x) \& f_2(x)$ ), entry-15 and entry-23 ( $\min f_2(x)$ ) in Table 2, are performed. The five selected cases are tested and depicted in Fig. 14 – Fig. 18, respectively.

For the first case of experimental results with the PID controller entry-1 ( $\min f_1(x)$ ), the  $K_p = 14.9993$ ,  $K_i = 0.0899$  and  $K_d = 989.8136$  are tuned by MATLAB/Simulink interfacing the dSPACE-DS1104 controller board. The reference speed, the actual speed, the control signal of PID controller and the motor current of the developed  $3\phi$ IM controlled system are measured as visualized in Fig. 14. It was found that the actual speed provides  $t_r = 75.50$  ms.,  $t_s = 545.15$  ms.,  $M_p = 16.50\%$  and  $E_{ss} = 0.00\%$ . Once the load disturbance is applied and released, the actual speed can be efficiently regulated with  $t_{reg} = 2.15$  s. This case provides the minimum rise time according to the simulation results.

For the second case by using the PID controller entry-4, the  $K_p = 14.9390$ ,  $K_i = 0.0415$  and  $K_d = 999.9787$  are adjusted by MATLAB/Simulink. The reference speed, the actual speed, the control signal and the motor current of the  $3\phi$ IM controlled system are recorded as displayed in Fig. 15. It was found that the actual speed provides  $t_r =$

80.22 ms.,  $t_s = 505.14$  ms.,  $M_p = 12.25\%$  and  $E_{ss} = 0.00\%$ . When the load disturbance is applied and released, the actual speed can be efficiently regulated with  $t_{reg} = 1.75$  s.

For the third case by using the PID controller entry-8 ( $\min f_1(x) \& f_2(x)$ ), the  $K_p = 13.6428$ ,  $K_i = 0.0352$  and  $K_d = 974.7195$  are tuned by MATLAB/Simulink. The reference speed, the actual speed, the control signal and the motor current of the system are recorded as shown in Fig. 16. The actual speed provides  $t_r = 85.65$  ms.,  $t_s = 385.45$  ms.,  $M_p = 8.50\%$  and  $E_{ss} = 0.00\%$ . Once the load disturbance is applied and released, the actual speed can be efficiently regulated with  $t_{reg} = 1.25$  s. This case shows both minimum rise time and minimum overshoot according to the simulation results.

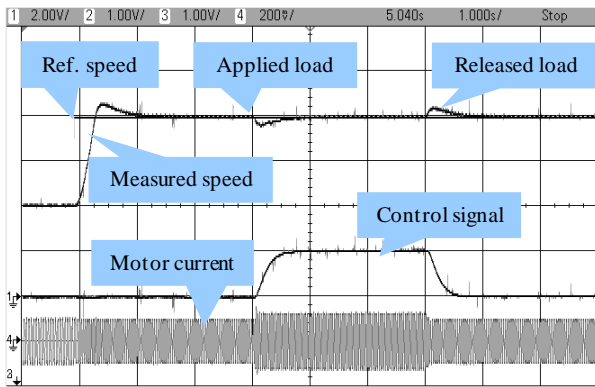


Fig.14. Experimental results: command tracking and load regulating responses of 3φ-IM with PID controller entry-1 ( $\min f_1(x)$ ).

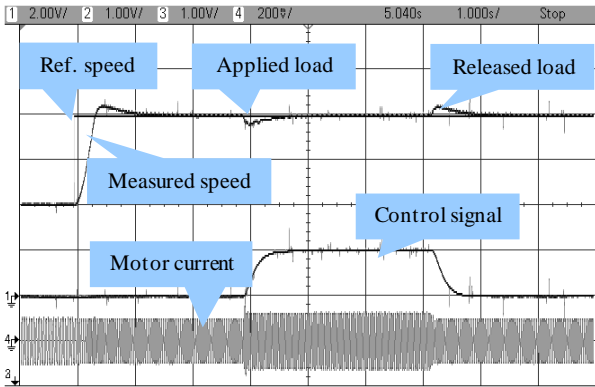


Fig.15. Experimental results: command tracking and load regulating responses of 3φ-IM with PID controller entry-4.

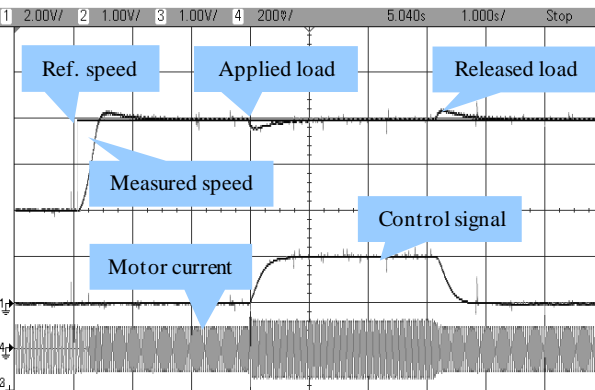


Fig.16. Experimental results: command tracking and load regulating responses of 3φ-IM with PID controller entry-8 ( $\min f_1(x) \& f_2(x)$ ).

For the fourth case by using the PID controller entry-15, the  $K_p = 11.6738$ ,  $K_i = 0.0327$  and  $K_d = 976.9918$  are adjusted. From Fig. 17, the actual speed provides  $t_r = 91.25$  ms.,  $t_s = 345.25$  ms.,  $M_p = 5.75\%$  and  $E_{ss} = 0.00\%$ . The actual speed is also regulated with  $t_{reg} = 1.05$  s.

For the final case by using the PID controller entry-23 ( $\min f_2(x)$ ), the  $K_p = 10.0015$ ,  $K_i = 0.0202$  and  $K_d = 930.1657$  are tuned. From Fig. 18, the actual speed gives  $t_r = 102.55$  ms.,  $t_s = 245.25$  ms.,  $M_p = 1.45\%$ ,  $E_{ss} = 0.00\%$  and  $t_{reg} = 0.75$  s. once load occurred. This case shows the minimum overshoot according to the simulation results.

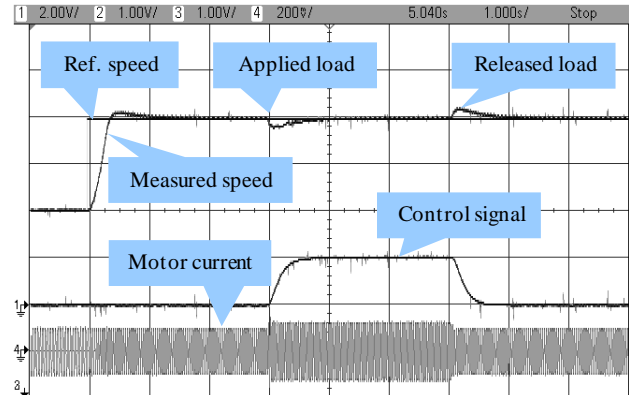


Fig.17. Experimental results: command tracking and load regulating responses of 3φ-IM with PID controller entry-15.

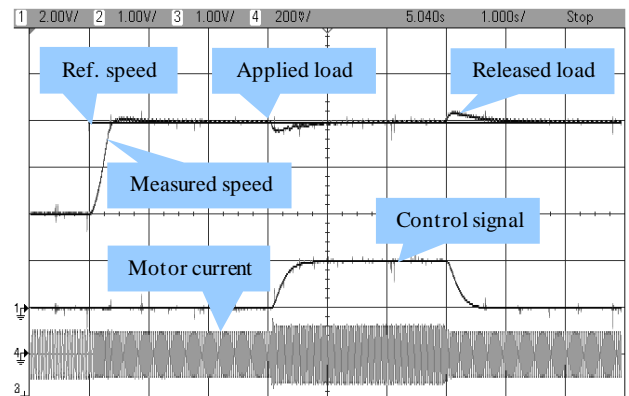


Fig.18. Experimental results: command tracking and load regulating responses of 3φ-IM with PID controller entry-23 ( $\min f_2(x)$ ).

From Fig. 14 – Fig. 18, the magnitude of the motor current are varied according to load occurrence. Once the load is applied, the magnitude of the motor current will be increased for compensation. Conversely, the magnitude of the motor current will be decreased, when the load is released. As overall system performance, it can be concluded from simulation and experimental results that the 3φIM speed control system with the PID controller designed by the ICS can provide very satisfactory results both tracking and regulating speed responses.

## VI. CONCLUSIONS

An application of the intensified current search (ICS) to multiobjective PID controller design for the three-



phase induction motor ( $3\phi$ -IM) speed control system has been proposed in this paper. The ICS algorithm possessing the memory list (ML), the adaptive radius (AR) and the adaptive neighborhood (AN) mechanisms, has been revised. To study its effectiveness and robustness, the ICS has been evaluated against three standard multiobjective test functions. Results obtained have been compared with those obtained by VEGA, NSGA-II, DEMO, MOCS and mMATS. As results, it was found that the ICS could provide superior results to other algorithms with less error. The ICS has been applied to multiobjective PID controller design optimization for  $3\phi$ -IM speed control system. From results of this application, it can be concluded that the ICS can successfully provide the optimal PID controllers for  $3\phi$ -IM speed control system giving very satisfactory responses. Good agreement between simulation and experimental results has been confirmed.

#### ACKNOWLEDGMENT

The authors wish to thank the anonymous reviewers for their constructive comments that greatly contributed to improve the final version of the paper. Also, they wish to thank the Editors for their generous comments and support during the review process.

#### REFERENCES

- [1] V. Zakian, *Control Systems Design: A New Framework*, Springer-Verlag, 2005.
- [2] F. Glover and G. A. Kochenberger, *Handbook of Metaheuristics*, Kluwer Academic Publishers, Dordrecht, 2003.
- [3] E. G. Talbi, *Metaheuristics form Design to Implementation*, John Wiley & Sons, Hoboken, 2009.
- [4] D. T. Pham and D. Karaboga, *Intelligent Optimisation Techniques*, Springer, London, 2000.
- [5] X. S. Yang, *Engineering Optimization: An Introduction with Metaheuristic Applications*, John Wiley & Sons, 2010.
- [6] J. H. Holland, *Adaptation in Natural and Artificial Systems*, University of Michigan Press, Ann Arbor, Michigan, USA, 1975, re-issued by MIT Press, 1992.
- [7] M. Dorigo, "Optimization, learning and natural algorithms," *Ph.D. thesis*, Politecnico di Milano, Italie, 1992.
- [8] D. Karaboga, "An idea based on honeybee swarm for numerical optimization," *Technical Report TR06*, Erciyes University, Engineering Faculty, Computer Engineering Department, 2005.
- [9] K. V. Price and R. Storn, "Differential evolution: a simple evolution strategy for fast optimization," *Dr. Dobb's J.*, vol. 22(4), 1997, pp. 18–24.
- [10] J. Kennedy and R. Eberhart, "Particle swarm optimization," *Proceedings of IEEE International Conference on Neural Networks*, vol. 4, 1995, pp. 1942–1948.
- [11] Z. W. Geem, *Music-Inspired Harmony Search Algorithm: Theory and Applications*, Springer, 2009.
- [12] X. S. Yang, "Firefly algorithms for multimodal optimization", *Stochastic Algorithms: Foundations and Applications (SAGA 2009)*, Lecture Notes in Computer Sciences, vol. 5792, 2009, pp. 169 – 178.
- [13] X. S. Yang and S. Deb, "Engineering optimisation by cuckoo search," *International Journal of Mathematical Modelling and Numerical Optimisation*, vol. 1(4), 2010, pp. 330–343.
- [14] X. S. Yang, "A new metaheuristic bat-inspired algorithm," *Nature Inspired Cooperative Strategies for Optimization (NISCO 2010)* (Eds. J. R. Gonzalez et al.), Studies in Computational Intelligence, Springer Berlin, 284, Springer, 2010, pp. 65–74.
- [15] S. Kirkpatrick, C. D. Gelatt and M. P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220(4598), 1983, pp.671–680.
- [16] F. Glover and M. Laguna, *Tabu Search*, Kluwer Academic Publishers, 1997.
- [17] A. Sukulin and D. Puangdownreong, "A novel metaheuristic optimization algorithm: current search," *The 11<sup>th</sup> WSEAS International Conference on Artificial Intelligence, Knowledge Engineering and Data Bases (AIKED '12)*, Cambridge, UK, 2012, pp.125–130.
- [18] A. Sukulin and D. Puangdownreong, "Control synthesis for unstable systems via current search," *The 11<sup>th</sup> WSEAS International Conference on Artificial Intelligence, Knowledge Engineering and Data Bases (AIKED '12)*, Cambridge, UK, 2012, pp.131–136.
- [19] A. Sukulin, D. Puangdownreong and S. Suwannarongsri, "Design of PID controllers for unstable systems using current search," *The 4<sup>th</sup> KKU International Engineering Conference 2012 (KKU-IENC 2012)*, 2012, pp.141–146.
- [20] D. Puangdownreong, "Application of current search to optimum PID controller design," *Intelligent Control and Automation*, vol. 3(4), 2012, pp. 303–312.
- [21] D. Puangdownreong, "Current search: performance evaluation and application to DC motor speed control system design," *Intelligent Control and Automation*, vol. 4(1), 2013, pp. 42–54.
- [22] D. Puangdownreong and A. Sukulin, "Current search and applications in analog filter design problems," *Journal of Communication and Computer*, vol. 9(9), 2012, pp. 1083–1096.
- [23] S. Suwannarongsri, T. Bunnag and W. Klinbun, "Energy resource management of assembly line balancing problem using modified current search method," *International Journal of Intelligent Systems and Applications (IJISA)*, vol. 6(3), 2014, pp. 1 – 11.
- [24] S. Suwannarongsri, T. Bunnag and W. Klinbun, "Optimization of energy resource management for assembly line balancing using adaptive current search," *American Journal of Operations Research*, vol. 4(1), 2014, pp. 8–21.
- [25] S. Suwannarongsri, T. Bunnag and W. Klinbun, "Traveling transportation problem optimization by adaptive current search method," *International Journal of Modern Education and Computer Science*, vol. 6(5), 2014, pp. 33–45.
- [26] A. Nawikavatan, S. Tunyasrirut and D. Puangdownreong, "Application of intensified current search to optimum PID controller design in AVR system," *Lecture Notes in Computer Science*, 2014, pp. 255–266.
- [27] A. Nawikavatan, S. Tunyasrirut and D. Puangdownreong, "Optimal PID controller design for three-phase induction motor speed control by intensified current search," *The 19<sup>th</sup> International Annual Symposium on Computational Science and Engineering (ANSCSE19)*, 2015, pp.104–109.
- [28] C. Thamarat, D. Puangdownreong, A. Nawikavatan and S. Tunyasrirut, "Multiobjective optimization of PID controller of three-phase induction motor speed control using intensified current search," *Global Engineering & Applied Science Conference 2015*, 2015, pp. 82–90.

- [29] D-N. Le, "Improving genetic algorithm to solve multi-objectives optimal of upgrading infrastructure in NGWN," *International Journal of Intelligent Systems and Applications (IJISA)*, vol. 5(12), 2013, pp.53-63.
- [30] S. S. Rao, *Engineering Optimization: Theory and Practice*, John Wiley & Sons, 2009.
- [31] C. L. Hwang and A.S.M. Masud, *Multiple Objective Decision Making: Methods and Applications*, Springer-Verlag, Berlin, 1979.
- [32] S. S. Rao and H. R. Eslampour, "Multistage multiobjective optimization of gearboxes," *ASME Journal of Mechanisms, Transmissions, and Automation in Design*, vol. 108, 1986, pp. 461-468.
- [33] J. D. Schaffer, "Multiple objective optimization with vector evaluated genetic algorithms," *The 1<sup>st</sup> International Conference on Genetic Algorithms*, 1985, pp. 93-100.
- [34] K. Deb, A. Pratap, S. Agarwal and T. Mayarivan, "A fast and elitist multiobjective algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, 2002, pp. 182-197.
- [35] T. Robic and B. Filipic, "DEMO: differential evolution for multiobjective optimization," *Lecture Notes in Computer Sciences*, vol. 3410(2005), 2005, pp. 520-533.
- [36] X. S. Yang and S. Deb, "Multiobjective cuckoo search for design optimization," *Computers & Operations Research*, vol. 40, 2013, pp. 1616-1624.
- [37] D. Puangdownreong, "Multiobjective multipath adaptive tabu search for optimal PID controller design," *International Journal of Intelligent Systems and Applications (IJISA)*, vol. 7(8), 2015, pp. 51-58, 2015.
- [38] H. Ghiasi, D. Pasini and L. Lessard, "A non-dominated sorting hybrid algorithm for multi-objective optimization of engineering problems," *Engineering Optimization*, vol. 43(1), 2011, pp. 39-59.
- [39] E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach," *IEEE Transactions on Evolutionary Computation*, vol. 3, 1999, pp.257-271.
- [40] E. Zitzler, K. Deb, and L. Thiele, "Comparison of multiobjective evolutionary algorithms: empirical results," *Evolution Computing*, vol. 8, 2000, pp.173-195.
- [41] K. Ogata, *Modern Control Engineering*, Prentice Hall, New Jersey, 2010.
- [42] R. C. Dorf and R. H. Bishop, *Modern Control Systems*, Prentice Hall, Upper Saddle River, New Jersey, 2005.
- [43] N. Quijano, K. Passino and S. Jogi, *A Tutorial Introduction to Control Systems Development and Implementation with dSPACE*, Dept. of Electrical Engineering, The Ohio State University, 2002.
- [44] [http://www-personal.umich.edu/~aghaffar/dSPACE\\_tutorial .pdf](http://www-personal.umich.edu/~aghaffar/dSPACE_tutorial.pdf), access on 31 March, 2016.

## Authors' Profiles



**Auttarat Nawikavatan** received his B.Eng. degree in electrical engineering from South-East Asia University (SAU), Bangkok, Thailand, in 2001, B.A. degree in political science from Ramkhamhaeng University, Bangkok, Thailand, in 2002 and M.Eng. degree in electrical engineering from King Mongkut's

Institute of Technology Ladkrabang (KMITL), Bangkok, Thailand, in 2005, respectively. Now, he is a Ph.D. candidate in electrical engineering, faculty of engineering, Pathumwan

Institute of Technology (PIT), Bangkok, Thailand.

Since 2006, he has been an the lecturer at Department of Electrical Engineering, Faculty of Engineering, South-East Asia University, Bangkok, Thailand. His research interests include system identification, intelligent control, power electronics and motor drives.



**Satean Tunyasrirut** received his B.S.I.Ed. in electrical engineering and M.S. Tech.Ed. in electrical technology from King Mongkut's Institute of Technology North Bangkok (KMITNB), Bangkok, Thailand in 1986 and 1994, respectively. He received the B.Eng in electrical engineering from Rajamangala University of Technology Thanyaburi (RMUTT), Thailand, in 2003 and D.Eng in electrical engineering from King Mongkut's Institute of Technology Ladkrabang (KMITL), Bangkok, Thailand, in 2007.

In 1995, he was awarded with the Japan International Cooperation Agency (JICA) scholarship for training the Industrial Robotics at Kumamoto National College of Technology, Japan. Since 2005, he has been an associated professor at Department of Instrumentation Engineering, Pathumwan Institute of Technology (PIT), Bangkok, Thailand. His research interests include modern control, intelligent control, power electronics, electrical machine and motor drives.



**Deacha Puangdownreong** received his the B.Eng. degree in electrical engineering from South-East Asia University (SAU), Bangkok, Thailand, in 1993, M.Eng. degree in control engineering from King Mongkut's Institute of Technology Ladkrabang (KMITL), Bangkok, Thailand, in 1996, and Ph.D. degree in electrical engineering from Suranaree University of Technology (SUT), Nakhon Ratchasima, Thailand, in 2005, respectively.

Since 1994, he has been with the Department of Electrical Engineering, Faculty of Engineering, South-East Asia University, where he is currently an associated professor of electrical engineering. He has authored 4 books and published as authors and coauthors of more than 120 research and technical articles in peer-reviewed journals and conference proceedings nationally and internationally. He has been listed in Marquis Who's Who in the World, Marquis Who's Who in Science and Engineering, and Top 100 Engineers-2011 in International Biographical Center, Cambridge, UK. His research interests include control synthesis and identification, metaheuristics and search algorithms as well as their engineering applications.

**How to cite this paper:** Auttarat Nawikavatan, Satean Tunyasrirut, Deacha Puangdownreong, "Application of Intensified Current Search to Multiobjective PID Controller Optimization", *International Journal of Intelligent Systems and Applications (IJISA)*, Vol.8, No.11, pp.51-60, 2016. DOI: 10.5815/ijisa.2016.11.06