ISSN 2008-0050 (Print), ISSN 1927-0097 (Online)

Proceedings of the 9th International Conference on Applied Operational Research (ICAOR 2017)
Chung Yuan Christian University
Taoyuan, Taiwan
18-20 December 2017

Benchmark Functions Based Performance Evaluation by Inference Model Pyramid Tree (PT) and Operation Tree (OT)

Li-Chuan Lien [1,4,*], Shou-Bin Chen [2], Jing-Yi Xu [3], Yan-Ni Liu [4] and Qi-Sheng Wang[1]
[1] Chung Yuan Christian University, Taoyuan, Taiwan
[2] Fuzhou CityConstruction Design and Research Institute Co., Ltd., Fuzhou, China
[3] China Construction Fifth Engineering Branch Co., Ltd, Fuzhou, China
[4] Fujian University of Technology, Fuzhou, China

Abstract
Structure equation model (SEM) approaches based on the historical data and perform its formula. Thus, the researcher can observed internal knowledge on problems and ensure sensitivity analysis or decision making via formula. This study used an SEM inference model approach, named pyramid tree (PT), model structure idea was from pyramid and operation trees (OTs). Four triangle layers organize a pyramid object. OT models formula based on the numerical historical data. Particle bee algorithm (PBA) optimize PT model's structure based on the structure behavior that designed by this study. In order to measure PT performance, this study evaluated PT performance based on proposed benchmark functions. This study evaluated performance of PT model's structure with fourteen 2- or 4-dimensional benchmark functions by 100, 400, 700 and 1000 of PBA iterations. Results showed PT performance was satisfied to fit those benchmark functions formula and better than single OT. PT is a self-modeling formula inference model method and more accurate than single OT. It is suitable to solve the problem while human would to figure out the formula based on the historical data such as material composite or other practical problems.

Keywords: pyramid trees; operation trees; particle bee algorithm; benchmark functions

1. Introduction

General inference model such as neural networks (NNs), support vector machines (SVMs) and other artificial intelligence (AI) approaches model problems of black box based on the historical data and widely used in many practical problems that model structure behaviors of natural systems such as human brain and statistic learning etc.,. Structure equation models (SEMs) were proposed in recently, SEM approaches model problems based on the historical data and perform its formula. Thus, the researcher can observed internal knowledge on problems and ensure sensitivity analysis or decision making via formula.

This study used an SEM inference model approach, named pyramid tree (PT) [1], model structure idea was from pyramid and operation trees (OTs) [2-4]. Four triangle layers organize a pyramid object. OT models formula based on the numerical historical data. Particle bee algorithm (PBA) optimize PT model's structure based on the structure behavior that designed by this study. In order to measure PT performance, this study evaluated PT performance based on proposed benchmark functions.

Swarm intelligence (SI) has been of increasing interest to research scientists in recent years. SI was defined by Bonabeau et al. as any attempt to design algorithms or distributed problem-solving devices based on the collective behavior of social insect colonies or other animals [5]. Bonabeau et al. focused primarily on the social behavior of ants [6], fish [7], birds (Particle Swarm Optimization PSO) [8] and bees (Bee Algorithm, BA) [9] etc.

* *Correspondence*: Li-Chuan Lien
Chung Yuan Christian University, 200 Chung Pei Road, Chung Li District, Taoyuan, Taiwan
lclien@cycu.edu.tw

In order to measure PT performance, this study used an optimization hybrid swarm algorithm called the particle bee algorithm (PBA) [10-13] that imitates a particular intelligent behavior of bird and honey bee swarms and integrates their advantages. Thus, PBA used to optimize PT model structure in this study.

2. Methodlogy

2.1 Pyramid tree (PT)

Modeling the Pyramid tree (PT) include two steps as the following (A), (B) and (C) section.

(A) OT Rules for operators, Parameters and codes

Figure 1 for a face of the five layers of the operation tree model. The rules for OT operators and parameters are showed in Table 1 to Table 2. The detail rules as the following:

- The first layer (leaf X_1) must be an operator. The leaf search area is code 1 to 9.
- The second, third and fourth layers (leaf X_2 to X_{15}) may be operator or parameters. Each leaf search area is code 1 to 18.
- The fifth layer (leaf X_{16} to X_{31}) must be parameters. Each leaf search area is code 10 to 18.
- In addition, this tree structure complies with the following rules:
    - When a leaf is assigned with code 6 to 9 its right branch would be ignored.
    - When a leaf is assigned with code 11 to 18, it would be an end leaf.
    - When a leaf is assigned with code 10, it would be an end leaf, and produce with a constant between -100 and +100.

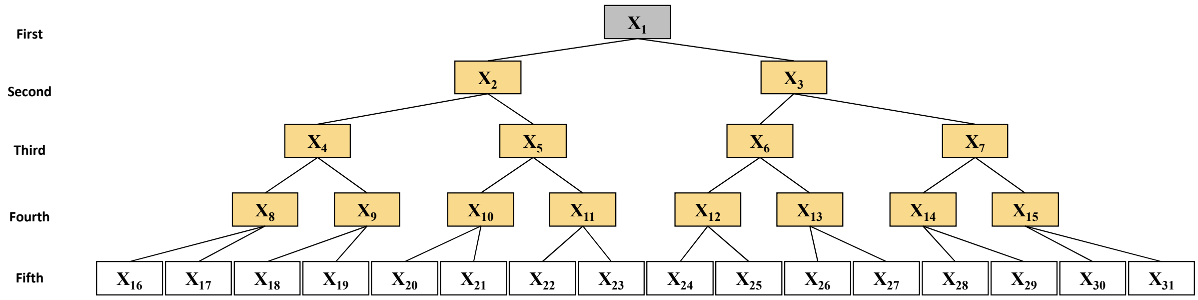

Figure 1 Five layers of operation tree model

Table 1 Rules for OT operators

Code	1	2	3	4	5	6	7	8	9
Operators	+	−	×	÷	Xy	ln	Sin	Cos	Tan

Table 2 Rules for OT parameters and constant

Code	10	11	12	13	14	15	16	17	18
Parameters	Constant	P_1	P_2	P_3	P_4	P_5	P_6	P_7	P_8

(B) PT Rules for operators, parameters and codes

In order to generate self-organizing formula, this study adopts the formula expression of PT which has been developed. PT has a total of four faceted, Figure 1 is a 4-faceted, and each face has four layers of operation tree PT. The PT has four faces (four OTs). Each OT is a formula. Thus, four OTs must follow the rules to combine together:

- Random sort variables to sequence OTs: (1) randomly calculate the sort of the four OT (the combination has 4! cases). (2) The sort of the first and second OTs firstly combine the tree (first OT put in left, second OT put in the right). (3) To followed by the subsequent OTs until the end (combine third and fourth OT).
- Random computing variables to combine OTs: The OTs combine method used Table 1 rule codes. Each combines randomly generated operators (the combination has 9^3 cases) to provide four aspects of OTs in combination.

- When the OTs combine are encoded as ln, Sin, Cos, and Tan, the left OT is retained, and the right OT is deleted.

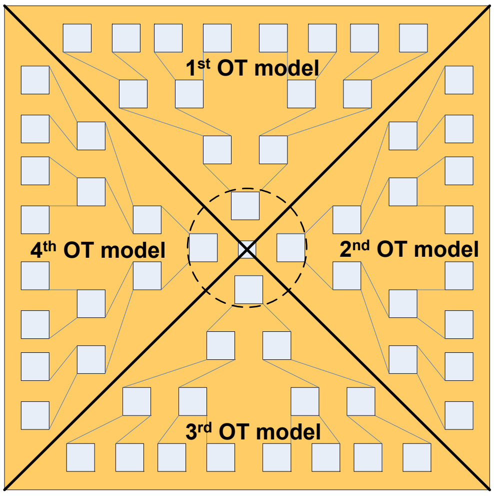

Figure 2 Four layers of pyramid Tree model

(C) Modified output value of PT

The predicted output values of PT usually have oblique phenomenon that there are high linear correlation but high root mean squared error (RMSE) between the predicted values and the actual values in dataset. Therefore, in this study, the single linear regression analysis was employed to modify the oblique phenomenon:

$$y = \alpha + \beta \cdot f \qquad (1)$$

Where

f = predicted output values of PT;

y = actual output values in dataset;

α *and* β = regression coefficients.

According to single linear regression analysis,

$$\alpha = \bar{y} - \beta \cdot \bar{f} \qquad (2)$$

$$\beta = \frac{\sum_{i=1}^{n}(f_i - \bar{f}) \times (y_i - \bar{y})}{\sum_{i=1}^{n}(f_i - \bar{f})^2} \qquad (3)$$

Where

$\bar{y}$ = the mean of the actual output values in the dataset;

$\bar{f}$ = the mean of the output values of the dataset predicted by the PT;

y_i = the actual output value of the i[th] data in the dataset;

f_i = the output value of the i[th] data point predicted by the PT.

2.2 Particle Bee Algorithm (PBA)

Particle bee algorithm (PBA) was proposed by Cheng and Lien [10-13]. It has been successful applied to many case studies. In PBA, the particle bee colony contains four groups, namely (1) number of scout bees (*n*), (2) number of elite sites selected out of n visited sites (*e*), (3) number of best sites out of n visited sites (*b*), and (4) number of bees recruited for the other visited sites (r). The first half of the bee colony consists of elite bees, and the second half includes the best and random bees. The particle bee colony contains two parameters, i.e., number of iteration for elite bees by PSO (*Peitr*) and number of iteration for best bees by PSO (*Pbitr*). PBA flowchart is shown in Figure3.

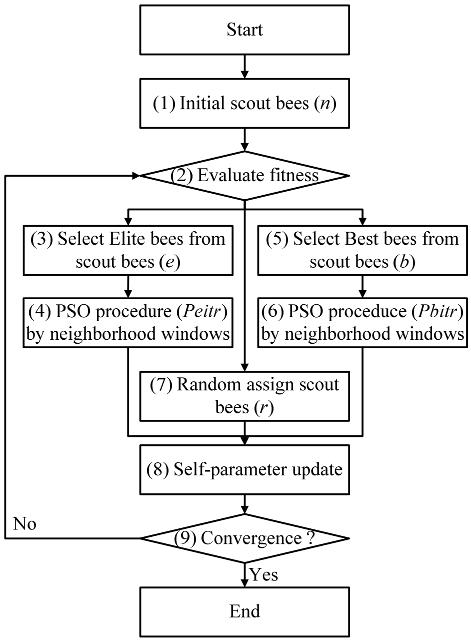

Figure 3 Particle bee algorithm flowchart

3. Experiment

3.1 Benchmark functions

Due to measure PT formula performance, this study evaluated performance of PT with fourteen 2- or 4-dimensional benchmark functions [11] by 100, 400, 700 and 1000 of PBA iterations. Figure 4 showed some of benchmark functions searching space. Table 3 showed an example for Schaffer function dataset. Each benchmark function was random 0 to 1 numbers for 40 times to calculate their formula results (Output). Some of benchmark function are include many parameters (showed in Table 4) and difficult to construct.

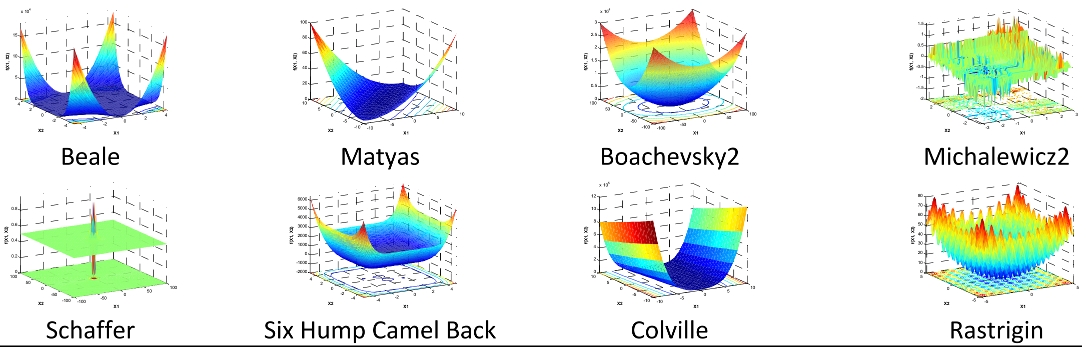

Figure 4 Some of benchmark functions searching space

Table 3 Example for Schaffer function dataset

No.	X_1	X_2	Result (Y)
1	0.07759519	0.60573796	0.12496929
2	0.58826063	0.47562724	0.0148459
…	…	…	…
40	0.48438553	0.40780796	0.0050576

Table 4 Fourteen benchmark functions formula

No	Function	D	Formulation
1	Beale	2	$f(x)=(1.5-x_1+x_1x_2)^2+(2.25-x_1+x_1x_2^2)^2+(2.625-x_1+x_1x_2^3)^2$
2	Booth	2	$f(x)=(x_1+2x_2-7)^2+(2x_1+x_2-5)^2$
3	Matyas	2	$f(x)=0.26(x_1^2+x_2^2)-0.48x_1x_2$
4	McCormick	2	$f(x)=\mathrm{SIN}(X_1+X_2)+(X_1-X_2)^2-1.5\times X_1+2.5\times X_2+1$
5	Three-hump camel	2	$f(x)=2\times X_1{}^2-1.05\times X_1{}^4+\frac{X_1{}^6}{6}+X_1\times X_2+X_2{}^2$
6	Boachevsky2	2	$f(x)=x_1^2+2x_2^2-0.3\cos(3\pi x_1)(4\pi x_2)+0.3$
7	Michalewicz2	2	$f(x)=-\sum_{i=1}^{D}\sin(x_i)(\sin(ix_i^2/\pi))^{20}$
8	Schaffer	2	$f(x)=0.5+\frac{\sin^2(\sqrt{x_1^2+x_2^2})-0.5}{(1+0.001(x_1^2+x_2^2))^2}$
9	Six Hump Camel Back	2	$f(x)=4x_1^2-2.1x_1^4+\frac{1}{3}x_1^6+x_1x_2-4x_2^2+4x_2^4$
10	Colville	4	$f(x)=100(x_1^2-x_2)^2+(x_1-1)^2+(x_3-1)^2+90(x_3^2-x_4)^2$ $+10.1(x_2-1)^2+(x_4-1)^2+19.8(x_2-1)(x_4-1)$
11	Dixon-Price	4	$f(x)=(x_1-1)^2+\sum_{i=2}^{D}i(2x_i^2-x_i-1)^2$
12	Rastrigin	4	$f(x)=\sum_{i=1}^{D}(x_i^2-10\cos(2\pi x_i)+10)$
13	Sphere	4	$f(x)=\sum_{i=1}^{D}x_i^2$
14	Step	4	$f(x)=\sum_{i=1}^{D}(x_i+0.5)^2$

3.2 Fitness function

As the objective of this study is to produce an accurate model to predict benchmark function formula, the Root Mean Squared Error (RMSE) was adopted as the evaluation function (fitness function) of solutions:

$$RMSE=\sqrt{\frac{\sum_{i=1}^{n}(\hat{y}_i-y_i)^2}{n}} \quad \text{......(4)}$$

Where $\hat{y}_i$ represents the modified predicted output value of the ith data point; y_i represents the actual output value of the ith data; and *n* represents the number of data.

3.3 PBA parameter setting

In this study, PBA interior parameters is setting as shown in Table 5

Table 5 Parameter values used in the experiments

PBA parameters setting	
n	50
e	n/2
b	n/4
r	n/4
w	0.9~0.7
v	*X*min/10~*X*max/10
Peitr	15
Pbitr	9

where *n* is population size (colony size); *w* is inertia weight; *v* is limit of velocity; *e* is elite bee number; *b* is best bee number; *r* is random bee number; *Peitr* is PSO iteration of elite bees; *Pbitr* is PSO iteration of best bees.

4. Result finding

This study has completed fourteen benchmark functions of the OT and PT formula. This study used OT and PT to build each benchmark function formula 15 times, with the best solution as the final result. Artificially to judgment the OT or PT formula similarity for the original benchmark function formula. The mechanism of judgment is mainly whether the number of parameters is similar to the original benchmark function formula, and then determine the degree of similarity between 1 and 4 of the benchmark function formula, and 4 is the most similar. The results are shown in Table 6. Figure 5 is the similarity of the OT model for the benchmark function formula in different PBA iterations. Figure 6 is the similarity of the PT model for the benchmark function formula in different PBA iterations. Figure 7 is the total similarity of the reference functions on OT and PT in different PBA iterations. The results showed that OT total similarity score was 26.5 lower than PT at 29.5. As a result, the PT has performed a better formula modeling function. In addition, Table 6 and Figure 5 to Figure 7 show that OT and PT similarity scores are not good. The main reason should be that the number of iterations is not enough, whether OT or PT, cannot effectively produce similar to the model of the fourteen benchmark functions.

Table 6 Benchmark functions formula similarity results by OT and PT

Functions \ Similarity)	OT					PT				
	100	400	700	1000	AVG/Total	100	400	700	1000	AVG/Total
Balae	2	1	2	2	1.75	1	1	2	2	1.5
Booth	3	2	3	2	2.5	1	2	2	1	1.5
Matyas	1	2	1	2	1.5	1	2	2	3	2
McCormick	1	1	1	2	1.25	3	2	3	2	2.5
Three-hump camel	1	2	2	3	2	2	1	3	3	2.25
Boachersky2	1	2	2	1	1.5	1	1	2	1	1.25
Michalewicz2	1	1	2	2	1.5	2	2	2	2	2
Schaffer	1	1	1	1	1	3	2	2	2	2.25
Six hump camel back	2	2	1	1	1.5	2	2	4	2	2.5
Colville	1	2	3	3	2.25	2	2	3	3	2.5
Dixon price	3	2	2	3	2.5	2	2	2	4	2.5
Rastrigin	3	2	2	1	2	3	2	2	3	2.5
Sphere	2	3	3	4	3	2	3	2	2	2.25
Step	1	3	1	3	2	2	2	2	2	2
Total	23	26	26	30	26.25	27	26	33	32	29.5
	105					118				

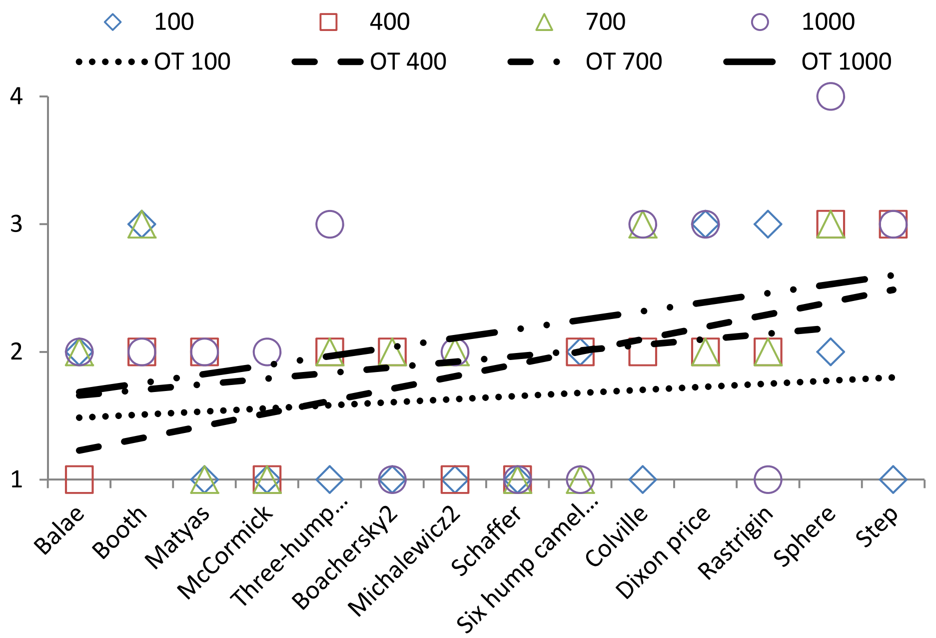

Figure 5 Similarity of the OT model for the benchmark function formula in different PBA iterations

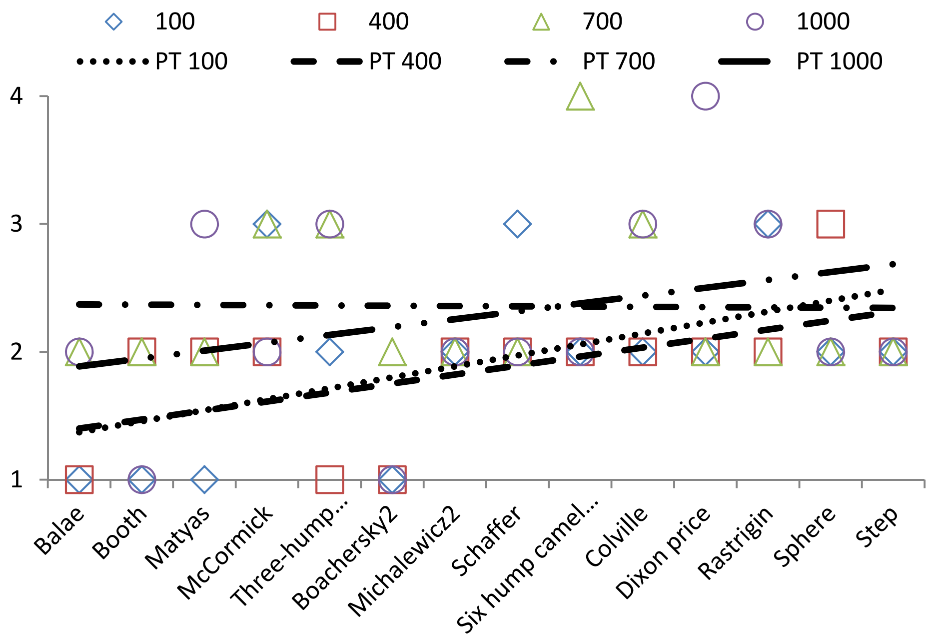

Figure 6 Similarity of the PT model for the benchmark function formula in different PBA iterations

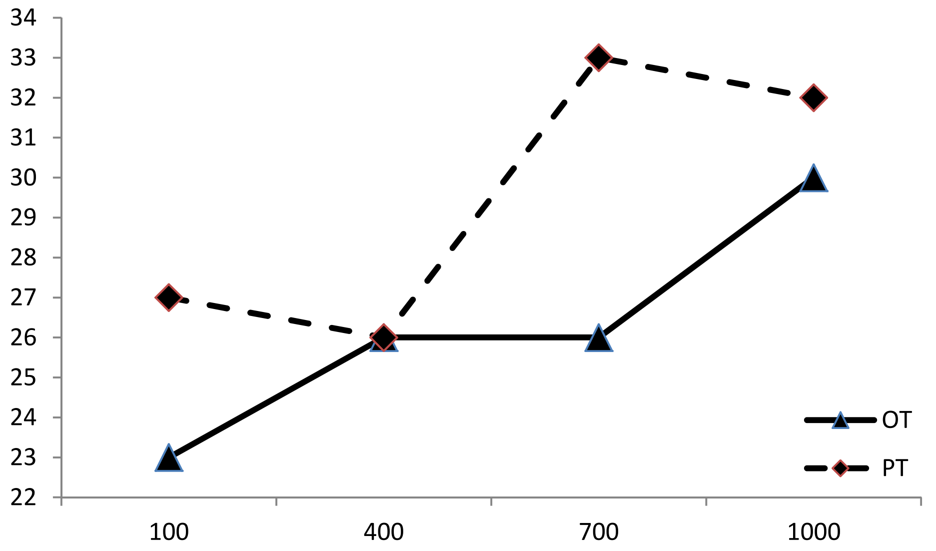

Figure 7 Total similarities of the benchmark functions on OT and PT in different PBA iterations

Conclusion

This study evaluated performance of PT model's structure with fifteen 2- or 4-dimensional benchmark functions by 100, 400, 700 and 1000 of PBA iterations. Those benchmark functions both combined with many complex mathematic operators. General SEMs would not satisfy to fit those benchmark functions formula. Results showed PT performance was satisfied to fit those benchmark functions formula and better than single OT. Some of PT self-modeling formula also discussed in this study and compared with those benchmark functions. PT is a self-modeling formula inference model method and more accurate than single OT. It is suitable to solve the problem while human would to figure out the formula based on the historical data such as material composite or other practical problems.

Acknowledgements

This study was sponsored by the Ministry of Science and Technology of Taiwan under Grant MOST 106-2218-E-033-005-, the Fuzhou CityConstruction Design and Research Institute Co., Ltd. under Grant GY-H-16023, the Education Department of Fujian Province under Grant GY-Z-15120 and JAT160317. The author also sincerely appreciates the constructive comments provided by the anonymous reviewers.

Reference

[1] Li-Chuan Lien*, Yan-Ni Liu, Peng-Cheng Zhang, Liao Zhi-Cheng, "On the Performance Evaluation of a Novel Inference Model Pyramid Tree (PT) by Particle Bee Algorithm," *2016 International Research Symposium on Engineering and Technology*. (Singapore)

[2] I-Cheng Yeh, Li-Chuan Lien* and Chien-Hua Peng, "Building Strength Models for High-Performance Concrete at Different Ages Using Genetic Operation Trees, Nonlinear Regression, and Neural Networks," *Engineering with Computers*, Vol.26, No.1, pp.61-73 (2010). (SCI/EI)

[3] I-Cheng Yeh, Li-Chuan Lien* and Chien-Hua Peng, "Modeling strength of high-performance concrete using genetic operation trees with pruning techniques," *Computers and Concrete*, Vol.6, No.3, pp. 203-223 (2009). (SCI/EI)

[4] I-Cheng Yeh* and Li-Chuan Lien, "Knowledge Discovery of Concrete Material Using Genetic Operation Trees," *Expert Systems with Applications*, Vol.36, No.3, pp.5807-5812 (2009). (SCI/EI)

[5] Bonabeau E., Dorigo M., and Theraulaz G., Swarm Intelligence: From Natural to Artificial Intelligence, *Oxford University Press*, New York (1999).

[6] Dorigo, M., "Optimization, Learning and Natural Algorithms," *Ph.D. Thesis, Politecnico di Milano*, Italy (1992).

[7] Li, X. L., "A new intelligent optimization-artificial fish swarm algorithm," *Ph.D. Thesis, Zhejiang University of Zhejiang*, China (2003).

[8] Kennedy J. and Eberhart R.C., "Particle swarm optimization," *In Proceedings of the 1995 IEEE International Conference on Neural Networks*, Vol.4, pp.1942-1948 (1995).

[9] Pham D.T., Koc E., Ghanbarzadeh A., Otri S., Rahim S. and Zaidi M., "The bees algorithm - a novel tool for complex optimization problems," *In Proceedings of the Second International Virtual Conference on Intelligent Production Machines and Systems*, pp.454-461 (2006).

[10] Lien Li-Chuan, Cheng Min-Yuan, "A hybrid swarm intelligence based particle-bee algorithm for construction site layout optimization," *Expert Systems with Applications*, Vol.39, No.10, pp.9642-9650 (2012).

[11] Cheng Min-Yuan, Lien Li-Chuan, "A Hybrid AI-based Particle Bee Algorithm (PBA) for Benchmark Functions and Facility Layout Optimization," ASCE, Journal of Computing in Civil Engineering, Vol.26, No.5, pp.612-624 (2012).

[12] Lien Li-Chuan, Cheng Min-Yuan, "Particle bee algorithm for tower crane layout with material quantity supply and demand optimization," *Automation in Construction*, Vol.45, No.9, pp.25-32 (2014).

[13] Cheng Min-Yuan, Lien Li-Chuan, "A Hybrid AI-based Particle Bee Algorithm (PBA) for Facility Layout Optimization," *Engineering with Computers*, Vol.28, No.1, pp.57-69 (2011).