Particle Swarm Optimization for Bi-level Pricing Problems in Supply Chains

Ya Gao, Guangquan Zhang, Jie Lu

Decision Systems & e-Service Intelligence Laboratory

Centre for Quantum Computation & Intelligent Systems

Faculty of Engineering & Information Technology, University of Technology, Sydney, PO Box 123, NSW 2007, Australia

{ya.gao, guangquan.zhang, jie.lu}@uts.edu.au

Hui-Ming Wee

Department of Industrial and Systems Engineering

Chung Yuan Christian University, Chungli 32023, Taiwan, ROC

weehm@cycu.edu.tw

With rapid technological innovation and strong competition in hi-tech industries such as computer and communication organizations, the upstream component price and the downstream product cost usually decline significantly with time. As a result, an effective pricing supply chain model is very important. This paper first establishes two bi-level pricing models for pricing problems with the buyer and the vendor in a supply chain designated as the leader and the follower, respectively. A particle swarm optimization (PSO) based algorithm is developed to solve problems defined by these bi-level pricing models. Experiments illustrate that this PSO based algorithm can achieve a profit increase for buyers or vendors if they are treated as the leaders under some situations, compared with the existing methods.

Key words: two-stage supply chain, bi-level programming, hierarchical decision-making, optimization, particle swarm optimization.

1. Introduction

Hi-tech products have the following characters: they have a shorter product life cycle time, a quicker response time, and an increasing need for globalization and massive customization. Moreover, the material purchase cost and product market price are decreasing at a continuous and sustained rate. The lead-time from order to delivery is usually suppressed from 955 (delivery of 95% of order within 5 days) to 1002 (delivery of 100% of order within 2 days) [1]. In some hi-tech industries, eg. computer and communication consumer products, some component costs and product prices are declining at about 1% per week [2]. This implies that purchasing or selling one-week earlier or later will result in about a 1% profit loss.

Many researchers, such as Lev and Weiss [3], Goyal [4], and Gascon [5] have studied the ordering policy in the classic economic order quantity (EOQ) model for finite and infinite horizons. Buzacott [6] and Erel [7] considered a continuous price increase due to inflation. Buzacott [6] assumed compound increasing price and setup costs were due to inflation in a finite horizon. Erel [7] considered a compound-increasing price EOQ model with the inflation rate. Yang and Wee [8] addressed a quick response production strategy with continuous demand and price declining in a finite horizon. Khouja and Park [9] derived an optimal lot size model for a decreasing rate of purchase cost in a finite horizon. All this research with cost/price change was based on a single rank. Most of these traditional EOQ models consider only a buyer’s profit. Recently, Yang el al. [1] developed a collaborative pricing and replenishment policy which took into consideration the perspectives of a vendor and a buyer, simultaneously. However, in reality, the buyer and the vendor in a supply chain would have a competitive relationship by nature. It would be difficult for them to share interests and set prices collaboratively. They need to make decisions based on their own interests, whilst still considering the choice of the other, as the other’s decisions will have an influence on their own interests.

Bi-level programming techniques aim to solve decision problems where each decision entity independently optimizes its own objective, but is affected by the actions from the other entity under a hierarchy [10]. In a bi-level decision problem, a decision entity at the upper level is known as the leader, and at the lower level, the follower [11]. The investigation of bi-level problems is strongly motivated by real world applications, and bi-level programming techniques have been applied with remarkable success in different domains such as mechanics [12], decentralized resource planning [13], electric power markets [14], logistics [15], civil engineering [16], and road network management [17, 18]. Much research has been conducted on the optimality conditions and solution algorithms for bi-level decision problems [19]. A large part of the research on bi-level decision problems has been centered on their linear version, the linear bi-level problems [20], for which nearly two dozen algorithms have been developed [21]. The well known ones include the Kuhn-Tucker approach [22], the Kth-best algorithm [23], the Branch-and-Bound algorithm [22], and genetic algorithm based approaches [24]. Current research on bi-level programming techniques mainly focus on non-linear bi-level decision problems [25], multi-leader bi-level decision problems [25], multi-follower decision bi-level problems [11][26], multi-objective bi-level decision problems [27], and fuzzy bi-level decision problems [15].

To solve the pricing problem in a supply chain more practically, this paper uses bi-level programming techniques to develop two bi-level pricing models. One bi-level pricing model considers a buyer as the leader who has the privilege of deciding first, and the vendor as the follower who makes decisions after the buyer, whilst the other pricing model takes a vendor's profit as priority and makes the vendor the leader and the buyer the follower. These two pricing models allow a buyer and a vendor to make decisions sequentially, fully considering the mutual influences of each other. Both the buyer and the vendor aim to maximize their profits in a supply chain system, but their decisions are related to each other in a hierarchical way.

This paper is organized as follows. Following the Section 1 introduction, Section 2 briefly introduces bi-level programming and provides a mathematic bi-level decision model. Section 3 establishes two bi-level pricing models for the buyer and the vendor in a supply chain by designating the buyer and the vendor as the leader, respectively. To solve problems defined by these two bi-level pricing models, a PSO based algorithm is developed in Section 4. Section 5 employs an example to carry out the experiments. Finally, conclusions and further studies are outlined in Section 6.

2. Preliminaries

Bi-level programming typically models bi-level decision problems, in which the objectives and the constraints of both the upper and the lower level decision entities (leader and follower) are expressed by linear or nonlinear functions, as follows [18]:

For $x \in X \subset R^n$, $y \in Y \subset R^m$, $F : X \times Y \to R^1$, and $f : X \times Y \to R^1$,

$$\min_{x \in X} F(x, y)$$

subject to $G(x, y) \le 0$

$$\min_{y \in Y} f(x, y)$$

subject to $g(x, y) \le 0$

where the variables *x, y* are called the leader's and the follower's decision variables respectively, *F*(*x, y*) and *f*(*x, y*) are the leader's and the follower's objective functions.

This model aims to find a solution to the upper level problem $\min_{x \in X} F(x, y)$ subject to its constraint $G(x, y) \le 0$. For each value of the leader's variable *x, y* is the solution of the lower level problem $\min_{y \in Y} f(x, y)$ under its constraint $g(x, y) \le 0$.

3. Two Bi-level Pricing Models in a Supply Chain

In this section, by switching the leader and follower roles, respectively, between a buyer and a vendor, we develop two bi-level pricing models in a supply chain.

The formulation for the pricing problem in a supply chain in this paper is developed from the assumptions of Yang et al. [1]:

(1) A vendor and a buyer's replenishment rates are instantaneous.

(2) The component purchase cost and the product price to an end consumer decline at a continuous rate per unit time.

(3) The finite planning horizon and the constant demand rate are considered.

(4) Each replenishment time interval is the same.

(5) No shortage is allowed.

(6) The purchase lead-time is constant.

Based on the above assumptions, the buyer's net profit in a buyer-vendor system can be calculated by [1]:

$$NP_b = \frac{P_{m0}D}{\ln(1-r_m)}[e^{H\ln(1-r_m)} - 1] - P_{b0}Q\frac{1-(1-r_b)^H}{1-(1-r_b)^{\frac{H}{mn}}} - \frac{F_b HP_{b0}Q}{2mn}\frac{1-(1-r_b)^H}{1-(1-r_b)^{\frac{H}{mn}}} - mnC_b \quad (1)$$

A vendor's net profit can be calculated by [1]:

$$NP_v = P_{b0}Q\frac{1-(1-r_b)^H}{1-(1-r_b)^{\frac{H}{mn}}} - P_{v0}mQ\frac{1-(1-r_v)^H}{1-(1-r_v)^{\frac{H}{n}}} - \frac{F_v HP_{v0}(m-1)Q}{2n}\frac{1-(1-r_v)^H}{1-(1-r_v)^{\frac{H}{n}}} - nC_v \quad (2)$$

In (1), a buyer controls m, the number of the buyer's lot size deliveries per vendor's lot size; and r_m, the weekly decline-rate of market price to an end-consumer. In (2), a vendor controls n, the number of orders that the vendor places for the item from a supplier in the planning horizon; r_b, the weekly decline-rate of the buyer's purchase cost; and r_v, the weekly decline-rate of the vendor's purchase cost. All other parameters defined in the problem are constants, which may change if other specific problems are introduced. The explanations of symbols used in the above two formulas are listed in Table 1.

Table 1. Explanations on symbols used in (1) and (2)

n	number of orders that a vendor places for the item from a supplier in the planning horizon
m	number of buyer's lot size deliveries per vendor's lot size
Q	buyer's lot size
r_b	weekly decline-rate of the buyer's purchase cost
D	weekly demand rate
r_v	weekly decline-rate of the vendor's purchase cost
r_m	weekly decline-rate of market price to the end-consumer
H	weekly length of the planning horizon
Fv	vendor's holding cost per dollar per week
Fb	buyer's holding cost per dollar per week
Cv	vendor's ordering cost per order
Cb	buyer's ordering cost per order
Pv0	vendor's unit purchase cost at the initial time
Pb0	buyer's unit purchase cost at the initial time

Pm0	market price to the end consumer at the initial time
Pv(*t*)	vendor's unit purchase cost in week *t*
Pb(*t*)	buyer's unit purchase cost in week *t*
Pm(*t*)	market price to the end consumer in week *t*
NPv	vendor's net profit in the planning horizon
NPb	buyer's net profit in the planning horizon
NP	joint net profit of both the vendor and the buyer in the planning horizon

When making the pricing strategy, if we take the buyer's point of view to make his or her profit a priority over a vendor, we can designate a buyer as the leader and a vendor as the follower. By combining Formulas (1) and (2), we establish a bi-level pricing model in a supply chain as follows:

$$\max_{m,r_m} NP_b(m,r_m,n,r_b,r_v) = \frac{P_{m0}D}{\ln(1-r_m)}[e^{H\ln(1-r_m)}-1] - P_{b0}Q\frac{1-(1-r_b)^H}{1-(1-r_b)^{\frac{H}{mn}}} - \frac{F_b H P_{b0} Q}{2mn}\frac{1-(1-r_b)^H}{1-(1-r_b)^{\frac{H}{mn}}} - mnC_b$$

subject to $m > 0$ (3)

$$0.0001 \le r_m \le 0.5$$

$$\max_{n,r_b,r_v} NP_v(m,r_m,n,r_b,r_v) = P_{b0}Q\frac{1-(1-r_b)^H}{1-(1-r_b)^{\frac{H}{mn}}} - P_{v0}mQ\frac{1-(1-r_v)^H}{1-(1-r_v)^{\frac{H}{n}}} - \frac{F_v H P_{v0}(m-1)Q}{2n}\frac{1-(1-r_v)^H}{1-(1-r_v)^{\frac{H}{n}}} - nC_v$$

subject to $n > 0$

$$0.0001 \le r_b \le 0.5$$

$$0.0001 \le r_v \le 0.5$$

In this model, both the buyer and the vendor adjust their own controlling variables respectively, wishing to maximize their own profits, under specific constraints. The buyer is the leader, who makes a decision first; and the vendor is the follower, who makes a decision after the buyer.

If we take the point of view of a vendor to make his or her profit a priority over a buyer, we can designate the vendor as the leader and the buyer as the follower. By combining Formulas (1) and (2), we establish another bi-level pricing model in a supply chain as follows:

$$\max_{n,r_b,r_v} NP_v(m,r_m,n,r_b,r_v) = P_{b0}Q\frac{1-(1-r_b)^H}{1-(1-r_b)^{\frac{H}{mn}}} - P_{v0}mQ\frac{1-(1-r_v)^H}{1-(1-r_v)^{\frac{H}{n}}} - \frac{F_v HP_{v0}(m-1)Q}{2n}\frac{1-(1-r_v)^H}{1-(1-r_v)^{\frac{H}{n}}} - nC_v$$

$$\text{subject to} \quad n > 0 \qquad (4)$$

$$0.0001 \le r_b \le 0.5$$

$$0.0001 \le r_v \le 0.5$$

$$\max_{m,r_m} NP_b(m,r_m,n,r_b,r_v) = \frac{P_{m0}D}{\ln(1-r_m)}[e^{H\ln(1-r_m)} - 1] - P_{b0}Q\frac{1-(1-r_b)^H}{1-(1-r_b)^{\frac{H}{mn}}} - \frac{F_b HP_{b0}Q}{2mn}\frac{1-(1-r_b)^H}{1-(1-r_b)^{\frac{H}{mn}}} - mnC_b$$

$$\text{subject to} \quad m > 0$$

$$0.0001 \le r_m \le 0.5$$

In this model, both a buyer and a vendor adjust their own controlling variables respectively, wishing to maximize their own profits, under specific constraints. The vendor is the leader, who makes the first decision; and the buyer is the follower, who makes a decision after the buyer.

The above two bi-level pricing models describe non-linear bi-level decision problems, for which there is no solution in the classical method. To reach solutions for problems defined by these bi-level decision models, we will develop a PSO-based algorithm in next section.

4. A Particle Swarm Optimization Based Algorithm

In this section, we use the strategy adopted in the PSO method [28] to develop a PSO-based algorithm to reach solutions for problems defined by Formulas (3) and (4).

Figure 1 outlines the structure and process of this algorithm. We first sample the leader-controlled variables to find some candidates for a leader. We then use the PSO method, together with the Stretching technology [29], to obtain the follower's response for every leader's choice. Thus, a pool of candidate solutions for both the leader and the follower is formed. By pushing every solution pair towards the current best ones, the solution pool is updated. Once a solution is reached for the leader, the Stretching technology [29] is used to avoid local optimization. We repeat this procedure by a pre-defined count and reach a final solution.

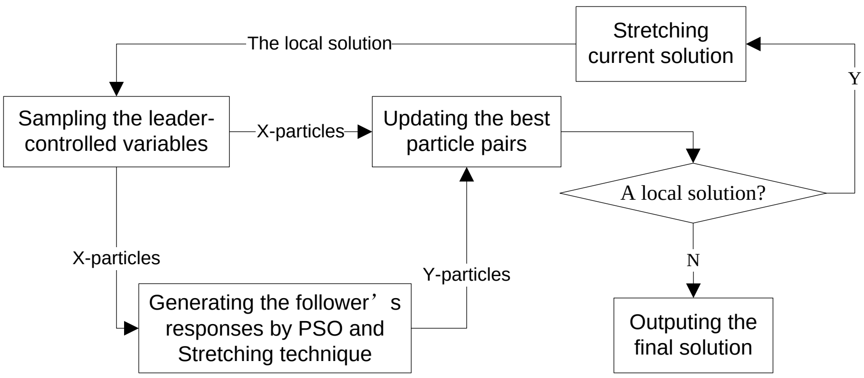

Figure 1 The outline of the PSO based algorithm

The details of this algorithm are specified below:

Algorithm 1: A PSO based algorithm for bi-level pricing problems

Step 1: Sample N_l particles of x_i, and the corresponding velocities v_{x_i};

Step 2: Initiate the leader's loop counter $k_l = 0$;

Step 3: For the k-th particle, $k = 1, \ldots, N_l$, generate the response from the follower;

Step 3.1 : Sample N_f candidates y_i and the corresponding velocities v_{y_i}, $i = 1, \ldots, N_f$;

Step 3.2 : Initiate the follower's loop counter $k_f = 0$;

Step 3.3 : Record the best particles p_{y_i} and y^* from p_{y_i}, $i = 1, \ldots, N_f$;

Step 3.4 : Update the follower's velocities and positions using

$$v_{y_i}^{k+1} = w_f v_{y_i}^{K} + c_f r_{1l}^K (p_{y_i} - y_i^K) + c_f r_{2l}^K (y^{*K} - y_i^K)$$
$$y_i^{K+1} = y_i^K + v_{y_i}^{k+1}$$

Step 3.5 : $k_f = k_f + 1$;

Step 3.6 : If $k_f \geqslant MaxK_f$ or the solution changes for several consecutive generations are small enough, then we use Stretching technology to obtain the global solution and go to Step 3.7. Otherwise go to Step 3.4;

Step 3.7 : Output y^* as the response from the follower.

Step 4: Record p_{x_i}, x_i^*, $i = 1,\ldots, N_l$ for each x_i, $i = 1,\ldots, N_l$;

Step 5: Update velocities and positions using

$$v_{x_i}^{k+1} = w_l v_{x_i}^K + c_l r_{1l}^K (p_{x_i} - x_i^K) + c_l r_{2l}^K (x_i^{*K} - x_i^K)$$
$$x_i^{K+1} = x_i^K + v_{x_i}^{k+1}$$

Step 6: $k_l = k_l + 1$;

Step 7: If $k_l \geq MaxK_l$, we use Stretching technology for the current leaders' solutions to obtain the global solution.

[end]

Notations used in Algorithm 1 are detailed in Table 2.

Table 2. Explanation of some notations used in the PSO-based algorithm

N_l	the number of candidate solutions (particles) for a leader
N_f	the number of candidate solutions (particles) for the follower
x_i	the *i*-th candidate solution for the leader
p_{x_i}	the best previously visited position of x_i
x_i^*	current best one for particle x_i
v_{x_i}	the velocity of x_i
k_l	current iteration number for the upper-level problem
y_i	the *i*-th candidate solution for the controlling variables from the follower
p_{y_i}	the best previously visited position of y_i
y^*	current best one for particle y
v_{y_i}	the velocity of y_i
k_f	current iteration number for the lower-level problem
$MaxK_l$	the predefined max iteration number for k_l
$MaxK_f$	the predefined max iteration number for k_f
w_l, w_f	inertia weights for the leader and the follower respectively (co-efficients for PSO)

c_l, c_f	acceleration constants for the leader and the follower respectively (co-efficients for PSO)
r_{1l}, r_{2l}	random numbers uniformly distributed in [0, 1] for the leader and the follower (co-efficients for PSO)

5. An Example and Experiments

In this section, we illustrate the bi-level pricing model and the PSO based algorithm developed in this study by the following numerical example where the parameters are given as follows:

(1) Demand rate per week, D =400 units
(2) Vendor's unit purchase cost at the initial time, $Pv0$ = \$4
(3) Buyer's unit purchase cost at the initial time, $Pb0$ = \$5
(4) Market price to the end consumer from the buyer at the initial time, $Pm0$ = \$6
(5) Buyer's ordering cost per order, Cb = \$30
(6) Vendor's ordering cost per order, Cv = \$1,000
(7) Buyer's holding cost per dollar per week, Fb = 0.004
(8) Vendor's holding cost per dollar per week, Fv = 0.004
(9) Time horizon considered, H = 52 weeks

Yang et al. [1] deals with this problem by solving a single level optimization problem: $NP = NP_b + NP_v$, where only the net profit of a buyer and a vendor must be the same and only m and n are adjustable decision variables. We relax the constraint of equal profit, and add r_m, r_b, and r_v as decision variables. By using the PSO based algorithm developed in this study to solve problems defined by Formulas (3) and (4), we obtain solutions for both the buyer and vendor. To evaluate the results of this research, we compare these results with the results from the original model by Yang et al [1] under a different negotiation factor α, which is defined as $\alpha = NPv / NPb$. To make the comparison fair and reasonable, besides m and n, we add r_m, r_b, and r_v as decision variables to be changeable to maximise the profit in Yang et al's model [1]. Table 3 lists solutions from this research and solutions from the model by Yang et al [1].

Table 3. Summary and comparison of running results

	m	r_m	N	r_b	r_v	NPb	NPv
Yang et al. [1] ($\alpha \geq 2$)	2	0.0001	9	0.0068	0.5	35,008	69,946
Yang et al. [1] ($1.5 \leq \alpha \leq 2$)	2	0.0001	9	0.01	0.5	41,280	63,710
Yang et al. [1] ($1 \leq \alpha \leq 1.5$)	2	0.0001	9	0.017	0.5	52,990	52,068
Yang et al. [1] ($0.5 \leq \alpha \leq 1$)	1	0.0001	9	0.032	0.5	68,548	36,605
Yang et al. [1] ($\alpha < 0.5$)	Not applicable						
This study (buyer as leader)	5	0.0071	6	0.0372	0.0753	52,399	16,866
This study (vendor as leader)	3	0.0015	7	0.0026	0.0767	21,359	64,165

From Table 3, we can see that, using the bi-level pricing model (buyer as leader) developed in this paper, the buyer's profit will increase compared with Yang's model when $\alpha \geq 1.5$. If the vendor is taken as the leader, he or she can achieve a profit increase when $\alpha \leq 2$, which is true for most pricing problems in a supply chain. As the follower, the vendor or the buyer is bound to lose, despite the range of the negotiation factor α. This is understandable, because in a bi-level decision situation, we always take the leader's interest as a priority.

These results reveal that when applying bi-level programming technologies on pricing problems in supply chains, some improvements can be achieved for a play (a buyer or a vendor) if he or she is the leader.

6. Conclusions and Further Studies

Based on the pricing and replenishment decision problems proposed by Yang et al [1], this paper develops two bi-level pricing models in a supply chain. To solve problems defined by these two models, a PSO based algorithm is used. Experiment results show that the bi-level pricing models and the PSO based algorithm can achieve profit improvements for both buyers and vendors under some situations, as compared with the model by Yang et al [1]. In the two-stage vendor-buyer inventory system, our experimental data show that the vendor, as leader, outperforms the buyer as leader. This is because a vendor, as the leader, improves the actual consumption rates; the vendor making the first decision ensures that production matches demand more closely, reduces inventory and improves business performance. This is why the VMI (vendor managed inventory) has become very popular in recent years.

In the future, our research will focus on the following studies:

(1) Further research and experiments will be undertaken to explore more complex applications of bi-level programming techniques in supply chain management, such as fuzzy bi-level pricing problems and multi-leader multi-follower bi-level pricing problems.

(2) Arising from real world bi-level decision problems in supply chain management, fuzzy bi-level programming techniques and multi-leader multi-follower bi-level programming techniques will be developed to deal with more practical bi-level decision situations.

Acknowledgments

The work presented in this paper was supported by Australian Research Council (ARC) under discovery grant DP0557154.

References

1. Yang, P. C., Wee, H. M., Yu, J.: Collaborative pricing and replenishment policy for hi-tech industry. Journal of the Operational Research Society. 58, 894–909 (2007)
2. Sern, L. C.: Present and future of supply chain in information and electronic industry. Supply Chain Management Conference for Electronic Industry. 6–27 (2003)
3. Lev, B., Weiss, H: Inventory models with cost changes. Operations Research 38. 53–63 (1990)
4. Goyal, S. K.: A note on inventory models with cost changes. Operations Research. 40, 414–415 (1992)
5. Gascon, A.: On the finite horizon eoq model with cost changes. Operations Research 43, 716–717(1995)
6. Buzacott, J. A.: Economic order quantities with inflation. Operations Research. 26, 553-558 (1975)
7. Erel, E.: The effect of continuous price change in the eoq. Omega. 20, 523–527 (1992)
8. Yang, P. C., Wee, H. M.: A quick response production strategy to market demand. Production Planning & Control. 12, 326–334 (2001)

9. Khouja, M., Park, S.: Optimal lot sizing under continuous price decrease. Omega. 31, 539–545 (2003)
10. Tsoukalas, A., Rustem, B., Pistikopoulos, E. N.: A global optimization algorithm for generalized semi-infinite, continuous minimax with coupled constraints and bi-level problems. Journal of Global Optimization. 44, 235-250 (2009)
11. Lu, J., Shi, C., Zhang, G.: On bilevel multi-follower decision making: General framework and solutions. Information Sciences. 176, 1607–1627 (2006)
12. Panagiotopoulos, P. D., Mistakidis, E. S., Stavroulakis, G. E., Panagouli, O. K.: Multilevel optimization methods in mechanics. In Migdalas, A., Pardalos, P. M., Varbrand, P. (eds.) Multilevel Optimization: Algorithms and Applications, pp. 51-90. Kluwer Academic Publishers (1997)
13. Yu, H., Dang, C., Wang, S.: Game theoretical analysis of buy-it-now price Auctions. International Journal of Information Technology and Decision Making. 5, 557–581 (2006)
14. Hobbs, B. F., Metzler, B., Pang, J. S.: Strategic gaming analysis for electric power system: an mpec approach. IEEE Transactions on Power System. 15, 637–645 (2000)
15. Zhang, G., Lu, J.: Model and approach of fuzzy bilevel decision making for logistics planning problem. Journal of Enterprise Information Management. 20, 178–197 (2007)
16. Fortuny-Amat, J., McCarl, B.: A Representation and Economic Interpretation of a Two-Level Programming Problem. Journal of the Operational Research Society. 32, 783-792 (1981)
17. Feng, C., Wen, C.: Bi-level and multi-objective model to control traffic flow into the disaster area post earthquake. Journal of the Eastern Asia Society for Transportation Studies. 6, 4253–4268 (2005)
18. Gao, Y., Zhang, G., Lu, J., Gao, S.: A bilevel model for railway train set organizing optimization. 2007 International Conference on Intelligent Systems and Knowledge Engineering (ISKE2007). 777–782 (2007)
19. Chinchuluun A., Pardalos P.M., Huang H-X.: Multilevel (Hierarchical) Optimization: Complexity Issues, Optimality Conditions, Algorithms. In: Gao, D, Sherali, H. (eds.) Advances in Applied Mathematics and Global Optimization, pp. 197-221. Springer (2009)
20. White D. J., Anandalinga, G.: A penalty function approach for solving bi-level linear programs. Journal of Global Optimization. 3, 397-419 (1993)
21. Wen, U. P., Hsu, S. T.: Linear bilevel programming problems-a review. Journal of Operational Research Society. 42, 123–133 (1991)
22. Bard, J.: Practical bilevel optimization. Kluwer Academic Publishers. Dordrecht. The Netherlands. (1998)
23. Lu, J, Zhang, G. and Dillon, T.: Fuzzy multi-objective bilevel decision making by an approximation kth-best approach. Multi-Valued Logic and Soft Computing. 14, 205-232 (2007)
24. Calvete, H. I., Gale, C., Mateo, P. M.: A new approach for solving linear bilevel problems using genetic algorithms. European Journal of Operational Research. 188, 14-28 (2008)
25. Zhang, G., Zhang G., Gao, Y., Lu, J. Competitive strategic bidding optimization in electricity markets using bi-level programming and swarm technique. accepted by IEEE Transactions on Industrial Electronics. (2009)
26. Zhang, G., Lu, J.: Fuzzy bilevel programming with multiple objectives and cooperative multiple followers. Journal of Global Optimization. 47, 403-419 (2010)
27. Gao, Y., Zhang, G., Lu, J.: A λ-cut and goal programming based algorithm for fuzzy linear multiple objective bi-level optimization. accepted by IEEE Transactions on Fuzzy Systems (2009)
28. Lu, H., Chen, W.: Self-adaptive velocity particle swarm optimization for solving constrained optimization problems. Journal of Global Optimization. 41, 427-445 (2008)
29. Jiang, Y., Hu, T., Huang, C., Wu, X.: An improved particle swarm optimization algorithm. Applied Mathematics and Computation. 193, 231-239 (2007)