International Journal of Computer and Communication Technology

Volume 4 | Issue 4 Article 6

October 2013

Intelligent Decision Making Using Evolutionary System for Optimizing Product-Mix Model

Suprava Devi
Department of Information Technology, College of Engineering & Technology Bhubaneswar, Orissa , India, suprava_devi@yahoo.in

Sushanta Panigrahi
ctcsushanta@gmail.com

Sunil Kumar Panigrahi
Department of Computer Science, APEX Institute of Technology & Management Bhubaneswar, Orissa , India, ctcsunil@gmail.com

Recommended Citation

Devi, Suprava; Panigrahi, Sushanta; and Panigrahi, Sunil Kumar (2013) "Intelligent Decision Making Using Evolutionary System for Optimizing Product-Mix Model," *International Journal of Computer and Communication Technology*: Vol. 4 : Iss. 4 , Article 6.
DOI: 10.47893/IJCCT.2013.1205
Available at: https://www.interscience.in/ijcct/vol4/iss4/6

Intelligent Decision Making Using Evolutionary System for Optimizing Product-Mix Model

Suprava Devi[1], Sushanta Kumar Panigrahi[2], Sunil Kumar Panigrahi[3]
[1] Department of IT, CET, Bhubaneswar, Orissa, India
[2] Departments of ICT, Fakir Mohan University, Balasore, Orissa, India
[3] Departments of CS, APEX, Bhubaneswar, Orissa, India
E-mail : suprava_devi@yahoo.in[1], ctcsushanta@gmail.com[2], ctcsunil@gmail.com[3]

Abstract:* The development and deployment of managerial decision support system represents an emerging trend in the business and organizational field in which the increased application of Decision Support Systems (DSS) can be compiling by Intelligent Systems (IS). Decision Support Systems (DSS) are a specific class of computerized information system that supports business and organizational decision-making activities. A properly designed DSS is an interactive software-based system intended to help decision makers compile useful information from raw data, documents, personal knowledge, and/or business models to identify and solve problems and make decisions. Competitive business pressures and a desire to leverage existing information technology investments have led many firms to explore the benefits of evolutionary system data management solutions such as Particle Swarm Optimization (PSO) and Genetic Algorithm (GA). This study proposes a new model for product mix problem based on evolutionary system for optimizing constraint values as well as objective function. The formulations of the objective function for the minimization problem. This technology is designed to help businesses to finding multi objective functions.

Keywords: *Linear problem, Evolutionary System, particle swarm optimization, Genetic Algorithm, simplex method*

1. Introduction

Organizations generate and collect large volumes of data, which they use in daily operations. Yet despite this wealth of data, many organizations have been unable to fully capitalize on its value because information implicit in the data is not easy to distinguish. However, to compete effectively today, taking advantage of high-return opportunities in a timely fashion, decision-makers must be able to identify and utilize the information. These requirements imply that an intelligent system must interact with a data warehouse and must interface with decision support systems (DSS), which are used by decision-makers in their daily activities [1].

There is a substantial amount of empirical evidence that human intuitive judgment and decision-making can be far from optimal, and it deteriorates even further with complexity and stress. Because in many situations the quality of decisions is important, aiding the deficiencies of human judgment and decision-making has been a major focus of science throughout history. Disciplines such as statistics, economics, and operations research developed various methods for making rational choices. More recently, these methods, often enhanced by a variety of techniques originating from information science, cognitive psychology, and artificial intelligence, has been implemented in the form of computer programs as integrated computing environments for complex decision making. Such environments are often given the common name of decision support systems (DSS) [1][5][6].

The success of management depends on execution of managerial functions and all managerial functions revolve around decision-making and the manager is a decision maker. Decision making of a company is very complex and risk problem. Due to the constrained nature of the problem, this paper is looking for a new solution that improves the robustness against existing decision with high effectiveness [1]. The present article proposes a novel strategy for the proper choice of the Evolutionary System constants based on the simplex method. In this paper we also presents the comparison and the relative performance of Traditional Method with evolutionary system through which a decision maker can enhance decision making, and asses the benefits of variety of evolutionary system techniques. The objective of this paper is to determine the efficiency and accuracy of complex decision of any company.

2. Evolutionary System

Evolutionary System is a research area within Computer Science, which draws inspiration from the process of natural evolution. Evolutionary computation, offers practical advantages to the researcher facing difficult optimization problems. These advantages are multifold, including the simplicity of the approach, its robust response to changing circumstance, its flexibility and many other facets. The evolutionary approach can be applied to problems where heuristic solutions are not available or generally lead to unsatisfactory results. Thus evolutionary computing is needed for Developing automated problem solvers, where the most powerful natural problem solvers are human Brain and evolutionary process. In this work evolutionary system such as **PSO** and **GA** are used to optimize the Product-Mix Model [6][7].

2.1 Particle Swarm Optimization (PSO)

A Swarm can be defined as population of interacting elements (particles) that are able to optimize some global objective through collaborative search of space. It is initialized with a group of random particles and then searches for optima by updating generations. At each step, each particle keeps track of the best solution that it has achieved so far and keeps also track of the overall best value that is obtained thus far by all particles in the population. The nature of interactive elements depends on the problem domain. If the search space is an n-dimensional space, the i^{th} particle of the swarm may be represented by an n-dimensional vector $X_i = (x_{i1}, x_{i2},...,x_{in})$. The velocity of this particle can be represented by another n-dimensional vector $Vi = (v_{i1}, v_{i2},\dots,v_{in})$. The fitness of each particle can be evaluated according to the objective function of optimization problem. The best previously visited position of the particle i is noted as its individual best position $pbest_i = (p_{i1}, p_{i2},\dots,p_{in})$. The best position of the swarm is noted as the global best position $gbest_i = (g_1, g_2,\dots,g_n)$. At each step, the velocity of each particle and its new position will be re-estimated according to the following two equations:

$$V_i^{k+1} = \omega V_i^k + c_1 r_1 (pbest_i^k - X_i^k) + c_2 r_2 (gbest^k - X_i^k) \quad (1)$$
$$X_i^{k+1} = X_i^k + V_i^k \quad (2)$$

where, ω is called the inertia weight that controls the impact of previous velocity of particle on its current one. r_1 and r_2 are independently uniformly distributed random variables in the range [0,1]. $C_1$and C_2 are positive constant parameters called acceleration coefficients which control the maximum step size and K denotes evolutionary iterations. In PSO, equation (1) is used to calculate the new velocity according to its previous velocity and to the distance of its current position from both its own best historical position and the best position of the entire population. The particle flies toward a new position according to equation (2). The PSO algorithm is terminated with a maximal number of generations or the best particle position of the entire swarm cannot be improved further after a sufficiently large number of generations. Figure 1 shows the concept of modification of searching points in PSO [12], [13], [14], [15], [16], [19].

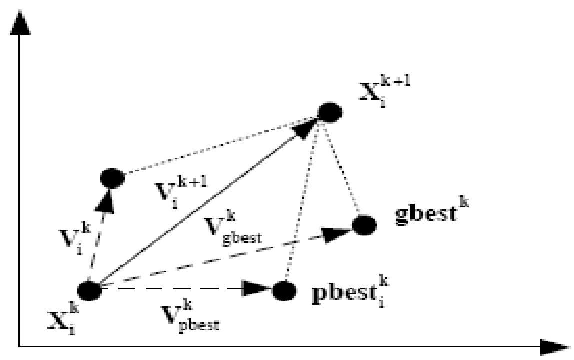

Fig. 1 The concept of modifications of Searching points.

A pseudocode of PSO is given below,

```
// Initialization
For each particle i
Randomly initialize Xi, Vi for particle i
End For
// Optimization
Do
    For each particle i
        Call calculate_fitness_value
        If current_fitness_value is better than
                        previous_best_fitness_value ( pi)
    Then
         Current_fitness_value of particle i becomes pi
      End If
  End For
  Call find_global_best_fitness
  For each particle i
            Call calculate_ Vi   based on eq. (.2)
            Call calculate_ Xi   based on eq.  (.3)
  End For
While MAX_iterations or min_error_criteria is not attained
```

The flow diagram of PSO algorithm is presented in figure 2.

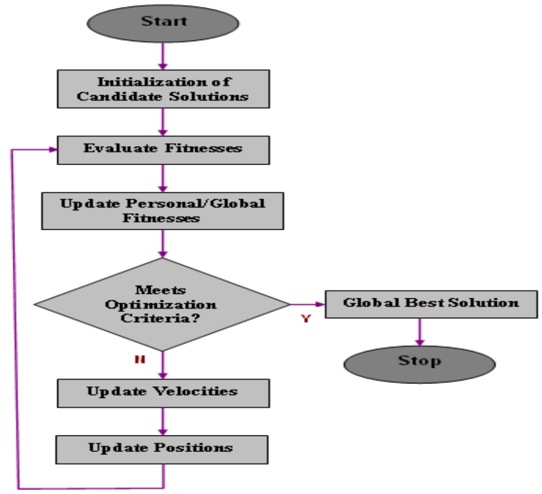

Fig. 2 A simple flow diagram of PSO algorithm.

2.2 Genetic Algorithm (GA)

Optimization is a general tool used in numerous problems of management, engineering and sciences. The life has been shaped by evolution during billions of years. Hence there seems to be things shared or sharable between different sciences that can be beneficial in many ways in the context of optimization and search. In order to implement, analyze, and utilize methods similar to evolution to solve optimization and search problems that are or seem to be computationally hard and/or complex i.e. have many parameters and/or are non-continuous and/or discrete or combinatorial. This has led to the study of genetic algorithms (GA) and other evolutionary methods for global optimization and search to be applied in various branches of management, engineering and sciences. The most attractive features of genetic algorithms are their simplicity, which makes them easy to use in practical problems and on the other hand their ability to solve hard problems [8] [9] [10] [11] [12].

Genetic algorithms (GA) first described by John Holland in 1960s and further developed by Holland and his students and colleagues at the University of Michigan in the 1960s and 1970s. GA used Darwinian Evolution to extract nature optimization strategies that uses them successfully and transforms them for application in mathematical optimization theory to find the global optimum in defined phase space. GA is used to find approximate solutions to difficult problems through a set of methods or techniques inheritance or crossover, mutation, natural selection, and fitness function. Such methods are principles of evolutionary biology applied to computer science. GA is useful for, solving difficult problems and modeling the natural system that inspired design [8] [9] [10] [11] [12].

A pseudo code of GA is given below,

- **Step1** (**Initialization)**: create a (random) starting population and evaluate the fitness of each individual.
- **Step2** (**Recombination)**: recombine the best individuals randomly by crossover and mutation.
- **Step3 (Evaluation)**: evaluate fitness
- **Step4 (Selection)**: select the best ones for the next generation.
- **Step5 (Termination)**: if the solution is found stop, otherwise continue at step 2

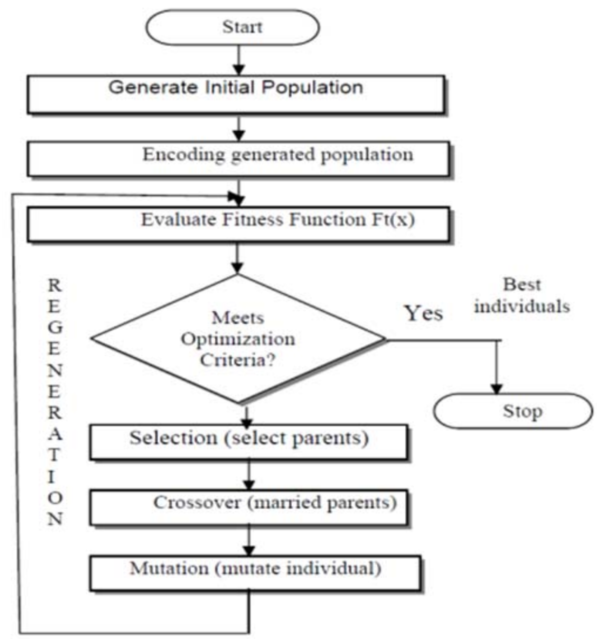

Fig. 3 A simple flow diagram of GA algorithm.

3. Traditional Method

Applying some well-defined mathematical algorithm known as optimization technique in which the decision theory is based on the assumptions of rational decision makers, whose objective is to optimize the attainment of goals? A well-known Optimization method is linear programming [3] [4].

3.1 Linear Programming

A linear programming is the most commonly applied form of constrained optimization. It may be defined as the problem of maximizing or minimizing a linear function subject to linear constraints. The constraints may be equalities or inequalities. The main components of linear programming problem are decision variable, variable bounds, constraints and objective functions [2] [3] [4].

Problem: Product Mix Linear Programming Model [2].

Goal: Maximize Total Profit / Month
Decision variables: $X_1$and X_2
Uncontrollable variables and parameters:
Market requirements: $X_1>=0$; $X_2>=0$
Profit contribution of each X_1 is 3 and X_2 is 2
Result variable: Profit=$3X_1+2X_2$
Constraints:
$X_1+X_2 <= 4$
$X_1-X_2 <= 2$

4. Experimental Result & Analysis

The goal of the optimization is usually to find the minima or the maxima of particular function. Sometimes, the relationships among the objectives of the optimization problem are so complex that the profit and cost function cannot be defined, or even there is no point in defining a quantitative function (e.g. when the goal is to optimize the quality of a product when the quality is determined by human taste). In this kind of situation, it is very difficult to apply traditional optimization algorithms.

In this section a number of experiments are carried out which outlines the effectiveness of the algorithm described above. The purpose of these experiments is to compare the performance of Simplex Method with Particle Swarm Optimization and Genetic Algorithm for the Product Mix Linear Programming Model. The experiments were conducted on 'Mat lab' and 'c' programming tool. Experimental results obtained from these algorithms were generated with 150 iteration per data point e.g. 40 different populations were created for all the algorithms and each algorithm was run 30 independent runs per data. The best result for each data was produce data point. For each algorithm there are number of different parameters, which need to varied to "fine-tune" the optimization process. Below we have given comparison graphs for objective values and fitness values.

4.1 Simplex Method

It is a scientific approach to automate managerial decision making and it consists of steps i.e. Define the problem, Classify the problem into a standard category, Construct a mathematical model, Find and evaluate potential solutions to model, Choose and recommend a solution to problem [3] [4].

For the above product mix model and the Solution is found as $X_1 = 3$ and $X_2 = 1$, Profit=Rs 11 after 10 to 12 generations.

Table 1: Objective values after 120 generation

Generations	***Traditional LP***	
	X_1	***X_2***
10	3	1
20	3	1
30	3	1
40	3	1
50	3	1
60	3	1
70	3	1
80	3	1
90	3	1
100	3	1
110	3	1
120	3	1

4.2 Product-Mix Model Using PSO

For the above Product model the Particle swarm optimization was set to,

Population size = 40 Maximum iteration = 500

Max Weight = 0.4   Min Weight = 0.9(Decreasing order)

C_1 & C_2 = 1.4   Dimension = 2

Velocity = 0 to 10(increasing order)

Agent initialization between 0 & 1

Fitness Function is, $3X_1 + 2X_2$ in maximization,

$X_1 + X_2 <= 4$

$X_1 - X_2 <= 2$, $X_1, X_2>=0$

Weight = Wmax-((Wmax-Wmin)/max. iter) × iter

Velocity = Vmin + (Vmax-Vmin) × Random (pop, dim) where Vmin=0 & Vmax=10

Table 2: Objective values after 120 generation

Generations	***PSO***

	X_1	X_2
10	2.9593	1.0149
20	2.9975	1.0025
30	2.9999	1
40	3	1
50	3	1
60	3	1
70	3	1
80	3	1
90	3	1
100	3	1
110	3	1
120	3	1

4.2 Product-Mix Model Using GA

For the above Product-Mix model the genetic algorithm population size was set to 40, the probabilities for crossover was set to 0.02 and for mutation was set to 0.45, the initial weight was set to [0.0, 0.0], maximum error rate was set to 0.001, 30 independent runs was sets per data and maximum iteration was considered as 500 for all the test problems because it was the best configuration found. The sequences of program operation are selection, crossover and mutation.

Table 3: Objective values after 120 generation

Generations	*GA*	
	X1	*X2*
10	2.0893	0.1037
20	2.2204	0.2276
30	2.3346	0.3425
40	2.4634	0.4652
50	2.5778	0.578
60	2.687	0.6994
70	2.8322	0.8422
80	2.9382	0.9545
90	2.9989	1.001
100	2.9989	1.001
110	2.9999	1.0001
120	2.9999	1.0001

Table 4 and Table 5 summarize the empirical results of the LP Model and Proposed PSO and GA Model on optimization of the Product Mix Problem for fitness value and maximization of profit respectively. The result by the test dataset show that the accuracy and multi-objective resultant of PSO and GA model is much better than obtained from the LP Simplex model and figure 4 and 5 are the graphically representation of fitness value and optimization value respectively.

Table 4: Fitness values after 120 generation

Generation	*Traditional LP*		*PSO*		*GA*	
	X_1	X_2	X_1	X_2	X_1	X_2
10	3	1	2.959	1.014	2.089	0.103
20	3	1	2.997	1.002	2.220	0.227
30	3	1	2.999	1	2.334	0.342
40	3	1	3	1	2.463	0.465
50	3	1	3	1	2.577	0.578
60	3	1	3	1	2.687	0.699
70	3	1	3	1	2.832	0.842
80	3	1	3	1	2.938	0.954
90	3	1	3	1	2.998	1.001
100	3	1	3	1	2.998	1.001
110	3	1	3	1	2.999	1.000
120	3	1	3	1	2.999	1.000

Table 5: Objective values after 120 generation

Generation	*Traditional LP*	*GA*	*PSO*
10	11	6.4763	10.1077
20	11	7.1164	10.9965
30	11	7.689	10.9997
40	11	8.3207	11
50	11	8.8894	11
60	11	9.4597	11
70	11	10.1811	11
80	11	10.7237	11
90	11	10.9987	11
100	11	10.9987	11
110	11	10.9998	11
120	11	10.9998	11

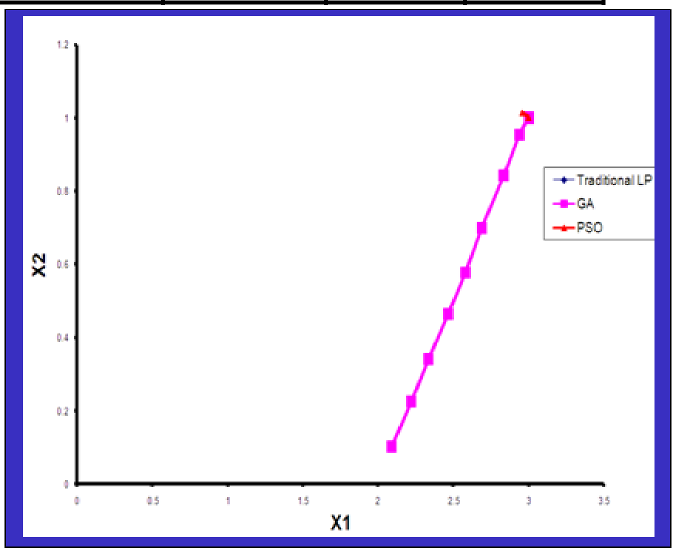

Fig. 4 Fitness comparison graph.

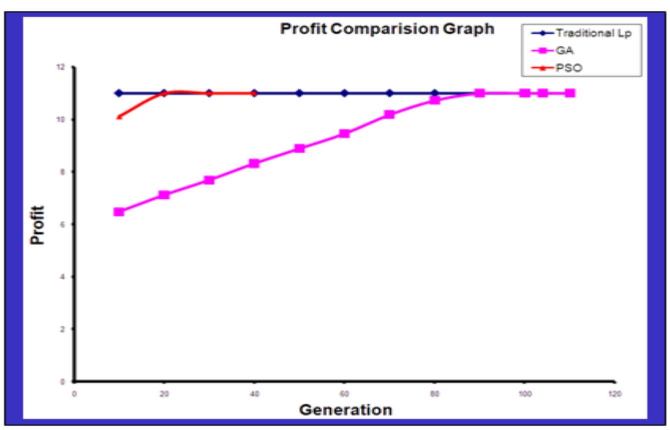

Fig. 5 Objective comparison graph.

5. Conclusion

In some cases, achievement of optimization problems can not be defined in quantitative way. In this kind of situation, it is very difficult to apply traditional and common optimization methods. But PSO and GA may be a good approach. This paper presented a new approach for the product mix linear programming model with simplified & standard algorithm to optimize combinatorial problem. All the algorithms are based on search technique to further improve individual's fitness that may keep high population, diversity and reduce the likelihood premature convergence. Our objective is to determine the performance of PSO & GA algorithm in comparison with simplex method for the complex decisions.

6. References

[1] "Decision Support Systems and Intelligent Systems", E. Turban and J. Aronson, Prentice Hall.

[2] "Operation Research", S. D. Sharma, Kedarnath, Ramnath & Co., 2000.

[3] "Linear Programming", Thomas S. Ferguson.

[4] James K. Strayer, Linear Programming and Applications, (1989) Springer-Verlag.

[5] "Decision Support Systems", Marek J. Druzdzel and Roger R. Flynn, University of Pittsburgh, Pittsburgh, PA 15260

[6] Marek J. Druzdzel. Probabilistic Reasoning in Decision Support Systems: From Computation to Common Sense. PhD thesis, Department of Engineering and Public Policy, Carnegie Mellon University, Pittsburgh, PA, December 1992.

[7] L.A. Zadeh, Some reflections on soft computing, granular computing and their roles in the conception, design and utilization of information/intelligent systems Soft Computing 2 (1998) (Springer-Verlag 1998).

[8] Kathryn, A.D., "Genetic Algorithms a Tool for OR?", Journal of the operational Research Society 47, pp. 550-561, (1996).

[9] A Directed Genetic Algorithm for Treating Linear Programming Problems, Dr. Bayoumi M.A.H. & El-Feky E.Z., Cairo University, Egypt.

[10] "Genetic algorithms and financial applications", Davis L., Deoeck, 1994.

[11] "Evolutionary Module Acquisition", Angeline, P. J. and Pollack, J. B., In Proceedings of the Second Annual

Conference on Evolutionary Programming, D.B. Fogel and W. Atmar (eds.), La Jolla, CA: Evolutionary.

[12] " Swarm Intelligence", J. Kennedy and R. Eberhart, Morgan Kaufmann Publishers, San Francisco, CA, 2001.

[13]. "An Introduction to Particle Swarm Optimization", Matthew Settles Programming Society, 1993.

[14] J. Kennedy and R. Eberhart (1995). "Particle swarm optimization," in *Proc. IEEE Int. Conf. Neural Networks*, vol. 4, pp. 1942–1947.

[15] Y. Shi and R. C. Eberhart, "Parameter selection in particle swarm optimization," in *Proc. Evolutionary Programming VII*, vol. 1447, 1998, pp. 591–600.

[16] M. Clerc, J. Kennedy (2002). "The particle swarm: explosion, stability and convergence in a multi-dimensional complex space", In IEEE Transaction on Evolutionary Computation Vol 6, pp. 58–73.

[17] Kou, G., Liu, X., Peng, Y., Shi, Y., Wise, M., & Xu, W. (2003). Multiple criteria linear programming approach to data mining: Models, algorithm designs and software development. *Optimization Methods and Software, 18*, 453-473.

[18] "Research and Trends in Data Mining Technologies and Applications", David Taniar, Monash University, Australia, Idea Group Publishing, Melbourne.

[19] Saeed Rategh, Farbod Razzazi, Amir Masoud Rahmani, Shayan Oveis Gharan "A Time Warping Speech Recognition System Based on Particle Swarm Optimization" in Second Asia International Conference on Modelling & Simulation-2008.

[20] Begambre O. and Laier J.E. "A hybrid Particle Swarm Optimization – Simplex algorithm (PSOS) for structural damage identification" in ELSEVIER journal in Advances in Engineering Software 40 (2009) P 883-891.