Journal of Computer Science 4 (4): 349-352, 2008
ISSN 1549-3636

Generalized Shortest Path Problem in Uncertain Environment Based on PSO

[1]Hayder A. Mukhef, [2]Ekhlas M. Farhan and [3]Mohammed R. Jassim
[1]Cultural Electronic Services Centre, University of Technology, Baghdad, Iraq
[2]Department of Electrical and Electronics, University of Technology, Baghdad, Iraq
[3]Center of Training and Work Shops, University of Technology, Baghdad, Iraq

Abstract: Generalized network problems describe situations in which flow can be generated or consumed through arcs. This research considers generalized shortest path problem in stochastic environment in which the changing rate of flow is random. To satisfy the demands of decision makers, three types of models have been formulated to solve the generalized shortest path problem under different decision criteria. The PSO algorithm is build to solve the generalized shortest path problem in stochastic environment. Finally, a numerical example is given to illustrate the effectiveness of the given algorithm.

Key words: Generalized short path, stochastic simulation, PSO, algorithm, environment, network

INTRODUCTION

Generalized shortest path problem can be applied into many practical cases because gaining or losing always exists. In cash flow, an investor who designs his portfolio will consider lots of ways of investment, banks, stocks, futures, options and many others. They bring income or loss which is difficult to predict. At the same time, an investor's type, whether he is a hedger, a speculator or an arbitrageur determines the style of his portfolio.

The shortest path problem, finding the path with minimum distance or cost from a starting node to an ending node, is one of the most fundamental network optimization problems. Generalized shortest path problem (GSP), finding the path with minimum cost in a generalized network, has been studied since the 1960s[4]. In 1966, Charnes and hike[3] designed an algorithm similar to Ford and Fulkerson's labelling method to solve the generalized shortest path problem. Ahuja, Magnanti and Orlin[1] developed a generalized network simplex algorithm to solve the generalized min-cost flow problem. Oldham[8] combined a dynamic programming approach similar to the Bellman-Ford algorithm with Megiddo's parametric search technique to solve the single-source generalized shortest path problem. Some stochastic shortest path models are studied[9,10].

In many cases of generalized shortest path problem, how the arcs might generate or consume flow is unpredictable thus we could not foretell ξ_{ij} to be a deterministic value, that is, multiplier ξ_{ij} does not appear to be deterministic value but a random variable. Therefore it is neither suitable nor effective to solve the problem with deterministic models. To deal with the uncertainty, we assume that each multiplier ξ_{ij} is a random variable and establish stochastic programming models based[5,6,7].

In this study, we consider a generalized network with n vertices with predetermined source vertex 1 and terminal vertex n. The arc multipliers $\xi = \{ \xi_{ij}\}$ are assumed to be random variables and the cost parameter c_{ij} is deterministic. The objective is to find a directed path from the source vertex 1 to the terminal vertex n which takes the minimum cost to transport one unit of flow to vertex n, with three types of criteria to compare the efficiency of the paths in stochastic programming. The PSO algorithm is build to solve the generalized shortest path problem in stochastic environment and a numerical case will be solved to show the validity of the PSO algorithm.

PROBLEM DESCRIPTIONS

A generalized network with stochastic parameters in suitable to model this problem. In the generalized network, vertices represent milestones, in which the capital returns from the previous investment and is available for further investment. Arcs denote investment procedures and ξ_{ij} represents the exchange rate of the investment denoted by arc (i, j). The cost c_{ij}

Corresponding Author: Hayder A. Mukhef, Cultural Electronic Services Centre, University of Technology, Baghdad, Iraq

represents original cost and the handling charge of trading. Similarly, we can also establish a generalized network to formulate a power transportation network in which the loss of power on the cables should be taken into account and an efficient route is wanted.

In order to model the generalized shortest path problem, we first give the following indices and parameters.

v = {1,2,. . . , n) the set of nodes, 1 the starting and n the terminal
A = The set of directed arcs available
g = (v, A): the graph
ξ_{ij} = Stochastic arc multipliers, {i,j} ∈ A
ξ_{ij} = Unit costs, {i,j} ∈ A

The objective of the generalized shortest path problem is to find a directed minimum-cost path from 1 to n.

A directed path P is a sequence of arcs (v_o, v_l), (v_l, v_2), (v_2, v3), (v_{m-1},…, v_m) and v_o and $v_{m,}$ are denoted as the source and terminal vertex of the path P. For concise, the path can be equally expressed as (v_o, v_l, v_2,…, v_m).

Different with the classic shortest path problem, in which there is always one unit of flow passing through, flow is gaining or losing in generalized network. Hence we define the concept of feasible flow in the following way.

A flow f is feasible if and only if:

$$\begin{cases} \sum_{(i,j)\in A} f_{ij} & - \quad \sum_{(j,i)\in A} f_{ij} \cdot \xi_{ji} = 0, 1 < i < n \\ \sum_{(i,j)\in A} f_{ij} & - \quad \sum_{(j,i)\in A} f_{ij} \,.\xi_{ji} \rangle 0 \quad i = 1 \\ \sum_{(i,j)\in A} f_{ij} & - \quad \sum_{(j,i)\in A} f_{ij} \,.\xi_{ji} \langle 0 \quad i = n \end{cases}$$

There are many paths from 1 to n and we want to minimize the total cost on the path per unit of flow transported to n. In the flow network with multipliers ξ, let prev(n) denote the vertex just in front of n in a directed path P from 1 to n and the cost function C(P, ξ) is defined as follows:

$$C(P,\xi) = \frac{\sum_{(i,j)\in P} c_{ij}.f_{ij}}{\xi_{prev(n),n}.f_{prev(n),n}} \tag{1}$$

where, f is a feasible flow that only takes positive value on path P, with:

$$\begin{cases} f_{ij} > 0, & (i,j) \in P \\ f_{ij} = 0, & (i,j) \notin P \end{cases}$$

MODELS FOR GSP IN STOCHASTIC ENVIRONMENT

We give the following three types of stochastic programming models in order to meet different optimization demands.

In the models, we use 0-1 decision variables:

$$x = \{x_{ij} | \in A\}$$

to describe the path, where x_{ij}= 1 means that (i, j) ∈ P and vice versa. A solution x is feasible if and only if:

$$\begin{cases} \sum_{(i,j)\in A} x_{1j} & - \quad \sum_{(j,i)\in A} x_{j1} = 1, \\ \sum_{(i,j)\in A} x_{ij} & - \quad \sum_{(j,i)\in A} x_{ji} = 0, \ 2 \le i \le n-1 \\ \sum_{(i,j)\in A} x_{nj} & - \quad \sum_{(j,i)\in A} x_{jn} = -1, \end{cases}$$

If x is feasible, let $P(x) = \{(i,j) \in A \,|\, x_{ij} = 1\}$, denote the path determined by x.

α-cost minimization model: Chance-Constrained Programming (CCP) developed[2,7], serves as a practical approach in uncertain programming. In chance constrained programming, a confidence level a is given and the objective is to minimize a cost °C which satisfies that the chance function Pr{C(P, α)≤°C} is at least α.

Here we establish the stochastic CCP model for the GSP in stochastic environment:

$$\begin{cases} \min {}^{\circ}C : \Pr\{C(P(x),\xi \le {}^{\circ}C\} \ge \alpha, \\ \text{subject}: \\ \sum_{(i,j)\in A} x_{1j} \quad - \quad \sum_{(j,i)\in A} x_{j1} = 1, \\ \sum_{(i,j)\in A} x_{ij} \quad - \quad \sum_{(j,i)\in A} x_{ji} = 0, \ 2 \le i \le n-1 \\ \sum_{(i,j)\in A} x_{nj} \quad - \quad \sum_{(j,i)\in A} x_{jn} = -1, \\ xij \in \{0,1\}, \text{any}(i,j) \in A, \end{cases} \tag{2}$$

where, α is predetermined confidence level.

Probability maximization model: In some cases with tight cost bond, decision-makers need a plan with maximal probability that the cost can be restricted. We can deal with this kind of problems by using Dependent-Chance Programming (DCP) initialized[7] to maximize the probability that we can meet the demand. The DCP model of GSP in stochastic a hybrid intelligent algorithm to solve the above environment is established as follows:

$$\begin{cases} \max \Pr\{C(P(x),\xi \le C^{o}\} \\ \text{subject:} \\ \sum_{(i,j)\in A} x_{1j} \quad - \quad \sum_{(j,i)\in A} x_{j1} = 1, \\ \sum_{(i,j)\in A} x_{ij} \quad - \quad \sum_{(j,i)\in A} x_{ji} = 0,\ 2 \le i \le n-1 \\ \sum_{(i,j)\in A} x_{nj} \quad - \quad \sum_{(j,i)\in A} x_{jn} = -1, \\ xij \in \{0,1\},\ \text{any}(i,j) \in A \end{cases} \tag{3}$$

where, °C is a predetermined cost bound.

PSO algorithm: The PSO Algorithm can be described as follows[11]:

Consider a swarm of p particles, with each particle's position representing a possible solution point in the design problem space D. For each particle i, Kennedy and Eberhart proposed that its[12]:

x^{i}_{k}	=	Particle position
v^{i}_{k}	=	Particle velocity
p^{i}_{k}	=	Best remembered individual particle position
p^{g}_{k}	=	Best remembered swarm position
c_1, c_2	=	Cognitive and social parameters
r_1, r_2	=	Random numbers between 0 and 1

Position of individual particles updated as follows:

$$x^{i}_{k+1} = x^{i}_{k} + x^{i}_{k+1} \tag{4}$$

with the velocity calculated as follows:

$$v^{i}_{k+1} = v^{i}_{k} + c_1 r_1 (p^{i}_{k} - x^{i}_{k}) + c_2 r_2 (p^{g}_{k} - x^{i}_{k}) \tag{5}$$

The PSO algorithm flow can be described as follows:

1. Initialize

- Set constants k_{max}, c1, c2, v^{max}_{o}
- Set counters k = 0, i = 1
- Randomly initialize particle positions $x^{i}_{0} \in D \text{ in } \mathfrak{R}^{n}$
- Randomly initialize particle velocities $0 \le v^{i}_{0} \le v^{max}_{o}$ for i = 1,…,p
- Evaluate fitness values $f^{\,i}_{0}$ using design space co-ordinates $x^{\,i}_{0}$ for i = 1,…p using Eq. 3
- Set f f^{i}_{best} = $f^{\,i}_{0}$, P^{i} $x^{\,i}_{0}$ for i = 1,…p
- Set f^{g}_{best} to best fi_{best} and go to corresponding $x^{\,i}_{0}$

2. Optimize

- Update particle velocity vector v^{i}_{k+1} using Eq. 5
- Update particle position vector x^{i}_{k+1} using Eq. 4
- Evaluate fitness value f^{i}_{k} using design space co-ordinates x^{i}_{k} using Eq. 3
- If $f^{i}_{k} \le f^{i}_{best}$ then $f^{i}_{best} = f^{i}_{k}, P^{i} = x^{i}_{k}$
- If $f^{i}_{k} \le f^{g}_{best}$ then $f^{i}_{best} = f^{g}_{k}, P^{g} = x^{i}_{k}$
- If k > k_{max} go to 3
- If stopping condition is satisfied then go to 3
- Increment i. If i > p then increment k and set i = 1
- Go to 2(a)

3. Report results
4. Terminate

NUMERICAL EXPERIMENTS

Presented here is an instance of generalized shortest path problem in stochastic environment on Fig. 1. The costs and multipliers on each arc are listed in Table 1.

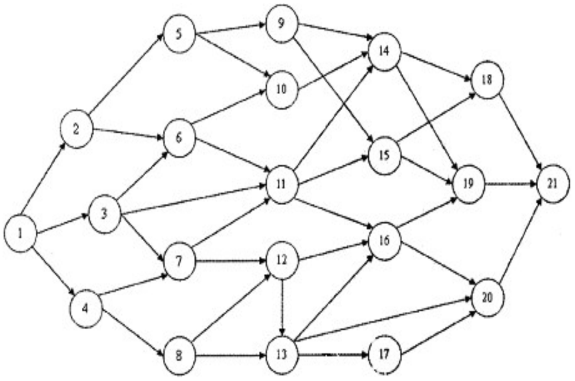

Fig. 1: A network graph

Table 1: Randam multiplier and costs

Arc	Multiplier	Cost	Arc	Multiplier	Cost
(1, 2)	N (1.15, 0.25)	6.3	(1, 3)	N (1.1, 0.35)	7.0
(1, 4)	N (1.2, 0.35)	7.0	(2, 5)	N (1.0, 0.3)	7.5
(2, 6)	N (1.25, 0.35)	2.0	(3, 6)	N (1.6, 0.95)	8.0
(3, 7)	N (0.9, 0.5)	5.0	(3, 10)	N (1.25, 0.35)	4.5
(4, 7)	N (0.95, 0.3)	6.3	(4, 8)	N (0.9, 0.45)	4.5
(5, 9)	N (0.85, 0.25)	6.2	(5, 10)	N (1.15, 0.55)	5.6
(6, 10)	N (1.0, 0.45)	6.6	(6, 11)	N (1.05, 0.5)	8.0
(7, 11)	N (1.3, 0.5)	8.5	(7, 12)	N (0.55, 0.35)	5.2
(8, 12)	N (1.0, 0.3)	7.1	(8, 13)	N (1.45, 0.35)	9.0
(9, 14)	N (1.1, 0.5)	3.5	(9, 15)	N (1.0, 0.5)	2.0
(10, 14)	N (1.05, 0.45)	7.5	(11, 14)	N (1.3, 0.4)	8.2
(11, 15)	N (1.15, 0.45)	7.5	(11, 16)	N (0.9, 0.35)	6.0
(12, 13)	N (1.1, 0.35)	7.5	(12, 16)	N (1.35, 0.5)	9.5
(13, 17)	N (0.75, 0.25)	6.2	(13, 18)	N (0.95, 0.55)	6.4
(13, 20)	N (1.25, 0.5)	8.7	(14, 18)	N (1.15, 0.35)	6.7
(14, 19)	N (1.25, 0.45)	8.6	(15, 18)	N (1.15, 0.5)	9.5
(15, 19)	N (1.45, 0.5)	9.8	(16, 19)	N (1.95, 0.35)	5.8
(16, 20)	N (1.05, 0.35)	5.0	(17, 20)	N (0.95, 0.3)	5.6
(18, 21)	N (1.2, 0.4)	3.5	(19, 21)	N (0.8, 0.3)	5.2
(20, 21)	N (0.9, 0.2)	6.7			

Table 2: α-cost minimization model

p	D	$c_1 = c_2$	Probability	Error (%)
40	4	0.3	0.883	1.06
40	4	0.2	0.883	1.02
40	3	0.3	0.880	0.69
30	4	0.3	0.877	0.34
30	4	0.2	0.874	0.00
30	3	0.3	0.881	0.84

Table 3: Probability maximization model

p	D	$c_1 = c_2$	Probability	Error (%)
40	4	0.3	0.883	1.06
40	4	0.2	0.883	1.02
40	3	0.3	0.880	0.69
30	4	0.3	0.877	0.34
30	4	0.2	0.874	0.00
30	3	0.3	0.881	0.84

The multipliers are assumed to be Normal random variables denoted by N (μ, σ^2) in which μ denotes the expected value and σ^2 denotes the variance of the Normal random variable.

In the hybrid intelligent algorithm for the numerical examples, three parameters are given in advance:

The swarm of p particles, with each particle's position representing a possible solution point in the design problem space D and c_1, c_2: Cognitive and social parameters

We do some experiments in different parameters and show the fluctuation rate of the result influenced by the three parameters.

Firstly, we calculate the $\alpha = 0.9$-cost minimization model from Eq. 2.

After running the PSO algorithm, the best path P* = (1,3,11,14,18,21).

The various data are presented and compared in Table 2. We can see that the objective values vary a little with different experiment parameters. In order to compare the difference among these objective values, we introduce the relative error parameter. It corresponds with the last column named by error in Table 2. The relative error is calculated by the formula:

$$|\text{actual value-optimal value}|/\text{optimal value}\times 100\%$$

In Table 2 we can see that the results differ with different parameters applied. The maximal error is 3.14%, which shows the effectiveness of the hybrid intelligent algorithm.

Secondly, we calculate the maximum credibility that the °C cost is less than or equal to 80.00 in Eq. 3.

After experiments, the best path P* = (1, 2, 6, 10, 14, 18, 21) is found.

Table 3 shows that the objective value fluctuates with different genetic parameters. However, the maximal error is 1.53%, which shows that the algorithm is effective.

CONCLUSION

In this study, we consider a generalized shortest path problem in stochastic environment. We established and α cost minimization model and a probability maximization model, the α-cost minimization model in order to satisfy different optimization demands. Then we integrated PSO to design a intelligent algorithm to solve the problem. Finally, the effectiveness of the PSO algorithm was shown by numerical examples.

REFERENCES

1. Ahuja, R.K., T.L. Magnanti and J.B. Orlin, 1993. Network Flows: Theory, Algorithms and Applications. Englewood Cliffs: Prentice-Hall, ch.5.
2. Charnes, A. and W.W. Cooper, 1959. Chance constrained programming. Manage. Sci., 6: 73-79.
3. Charnes, A. and W.M. Raike, 1966. One-pass algorithms for some generalized network problems. Perations Res., 14: 914-924.
4. Dantzig, G.B., 1963. Linear programming and extensions. Princeton University, Press Princeton, NJ.ch.22.
5. Cox L. A. Davis, L.,and Qiu Y 1991. Dynamic snticipatory routing in circuit-switched telecommunication networks *Handbook of Genetic Algorithms*, ed L Davis (New York: Van Nostrand) ch.11, pp.124-43.
6. Gen, M. and R. Cheng, 2000. Genetic Algorithms and Engineering Optimization. (Wiley Series in Engineering Design and Automation) .John Wiley & Sons, pps. 512. ISBN 0471315311.
7. Liu, B., 1997. Dependent-chance programming: A class of stochastic programming. Comput. Math. Appl., 34: 89-104.
8. Oldham, J.D., 2001. Combinatorial approximation algorithms for generalized flow problems. J. Algorith., 38: 135-169.
9. Takau, A., 2000. Recent Developments in maximum flow algorithms. J. Operations Res. Soc. Jap., 43: 476-499.
10. Ji, X., 2005. Models and algorithm for stochastic shortest path problem. Appl. Math. Computat., 170: 503-514.
11. Kennedy, J. and R.C. Eberhart, 1995. Particle swarm optimization. Proc. IEEE Int. Conf. Neural Networks, 5: 1942-1948.
12. Schutte, J.F., J.A. Reinbolt, B.J. Fregly, R.T. Haftka and A.D. George, 2004. Parallel global optimization with the particle swarm algorithm. Int. J. Humerical Methods Eng., 6: 2296-2305.