ISSN 1512-1232

THE COMPLEXITY OF ALGORITHMS FOR OPTIMIZATION PROBLEMS

Manana Chumburidze[1], Irakli Basheleishvili[2]

Akaki Tsereteli State University, Kutaisi, Georgia ,4600
[1]maminachumb02@gmail.com, [2]basheleishvili.irakli@gmail.com

Abstract

This paper devoted to development the complexity of algorithms for dynamical programming method and greedy approach for solution optimization problems of discrete systems. In particularly graph theoretic approach for solving typical tasks of supply chains management is advanced.

Keywords: *Greedy algorithm, dynamic programming methods, optimization, pseudo code.*

1. Introduction

Supply chain (SC) models play an important role in supply chain management (SCM) for reducing costs and finding better ways to create and deliver value to customers. supply chain innovation has been defined from multiply respective includes several production links (e.g., component construction, assembly, and merging) To be efficient, a supply chain must exploit modern productivity techniques and approaches, for example strategic inventory, reverse logistics, third party logistics, etc. Inventory management is the management of inventory and stock. As an element of supply chain management, inventory management (IM) includes aspects such as controlling and overseeing ordering inventory, storage of inventory, and controlling the amount of product for sale. As an element of supply chain management component construction and merging management (MM) are approaching technical and economic limitations for achieving further efficiency improvements. As these limits are approached, the costs of marginal efficiency improvements at the component level will rise. A systems approach offers creative avenues to further, cost-effective energy savings. Supply management is termed as the detailed process of planning, implementing, and controlling the efficient, cost effective flow and storage of materials and products, and related information within a supply chain to satisfy demand and logistics is recognized as the key enabler that allows a company to increase and maintain its competitive advantage and ensures maximum customer satisfaction. Graph theory is a systematic approach, which has proved to be useful for modeling and analyzing multistage problems of SC.

2. Statement problem of IM

Consider a serial supply chain for a single product in which there are N- facilities. Each facility may carry stocks of the product. In each period, it is possible to order a non-negative amount of product at any facility j from its immediate predecessor facility j-1up to the amount available there if j>1and from a supplier in any amount if j=1. Let Yi be the initial stock at facility in period i. Consider the problem that a production manager faces in scheduling production of a single product to meet a sales forecast over n - periods at minimum total cost. The manager forecasts that the vector of cumulative sales for a single product in the next n period will be a_i (1,.., n).

The manager seeks a vector $x_i \geq 0 -$of cumulative production levels in periods n that minimizes the n-period cost. There is a continuous convex cost $z_i(y_i)$of producing:

$$z_i(y_i) = \min_{x_i}(C_i(x_i) + h_i(x_i + y_{i-1} - a_i) + z_{i-1}), \quad a_i \le x_i + y_i \le a_i + ... + a_N, \quad i = 1,2,..., N$$

$$C_i(x_i) = \delta_i k_i + c_i x_i; \quad \delta_i = \begin{cases} 0, & x_i = 0 \\ 1, & x_i > 0 \end{cases} \quad x_N + y_N = a_N. \tag{1}$$

Solving the problems occurs by method of dividing the operations of ordering products on the multi-stage recursive procedures. Cost during the entire operation is achieved from a sequence of cost at each stage separately. The problem of optimal inventor management have been modeled by using the multistage graph modeling method, because graph is a particular way for visualization of the storing and organizing dates of the inventors and corresponding cost.

Let us consider modeling proceedings of this problem. How to calculate the “right amount” of inventory to stock. For simplest case, we have considered the solution of the problem for the following particular values: The manager forecasts that the vector of cumulative sales for a single product in the next n=3 periods correspondingly: $a_1=3$, $a_2=2$, $a_3=4$ unit products. Initial inventor is $y_1=1$ product. The cost of unit product for each level in periods of ordering is follow: $c_1=c_2=c_3=10$.The cost storing of inventory in each period for unit product are: $h_1=1$, $h_2=3$, $h_3=2$. Orders prices for each stage are: $k_1=3$, $k_2=7$, $k_3=6$. Take in account of subject to the constraints that in the end of planning period of ordering stock inventory should be equal zero.

Let us construct dynamical programming graph of inventory management in SC (Fig.1).

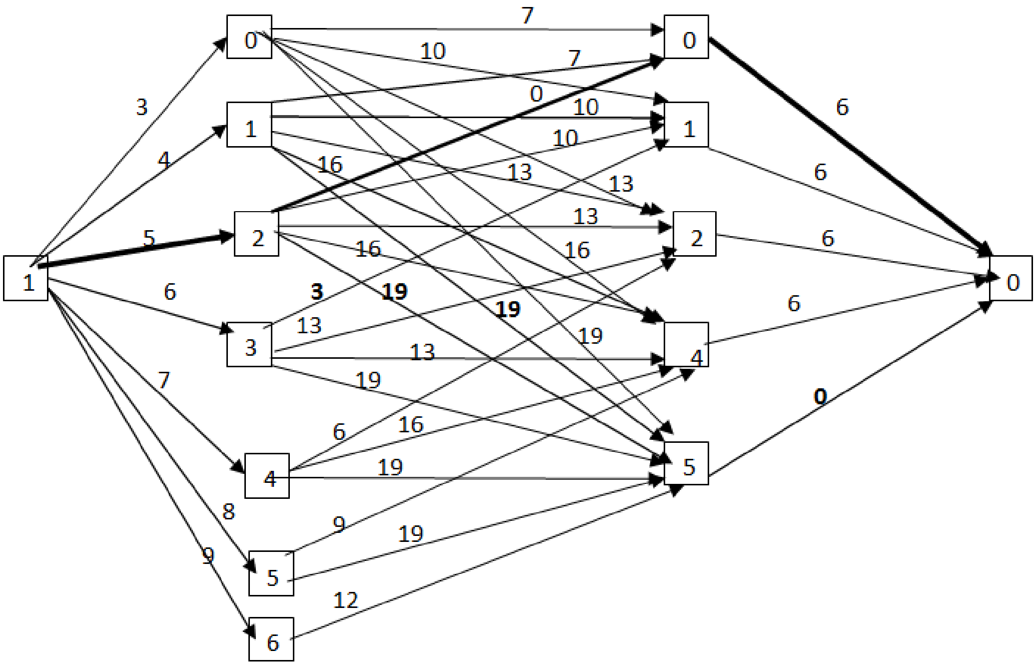

Fig. 1 Dynamical Programming Graph of IM

Allow us j=0, 1, 2, 3 are designates the stages ordering process of planning period. The vertex -V_{jk} (j is number of stage, k is number of current vertex) of graph are used for storing the current stock value (from the beginning to end).The edges of graphs are labeled by the cost of cumulative sales of product the corresponding stage of transition. Terminal vertexes store initial and finally value of inventers in the beginning and end of planning period correspondingly. Exactly in this particular case accordingly to subject of task they equals 1 and other value of other vertexes have been calculated by the recurrent formula:

$$V_{30}=V_{2k}+x_{3k}-a_3. \tag{2}$$

ISSN 1512-1232

Last formula return number of vertex previous stage: $V_{21}=0;V_{22}=1;V_{23}=2;V_{24}=3;V_{25}=4$; Similarly we have: $V_{2k}=V_{1kj}+x_{2kj}-a_{2j}$, k=0..4;j=0..6, Therefor we will get the following values of previous stage:$V_{11}=0;V_{12}=1;V_{13}=2;V_{14}=3;V_{15}=4;V_{16}=5;V_{17}=6$.

3. Solution problem of IM

We will consider the dynamical process of ordering. The solution of complex problems is based on breaking them down into simpler substructure in order to obtain products:

- In the beginning date of initial and last inventory in the terminal nodes is stored. According to the methods of dynamic programming there is the only variant of stock (beginning and end of processing look at vertex of the column j=0and j=3 of graph).
- In the first stage, some volume of *required products* is ordered. There are the 7 possible ways of ordering (look at vertex of the column j=1of graph), in the nodes the column j=1(first stage) the date of inventory on the end first stages is stored.
- In the second stage, some volume of *required products* is ordered. There are the 35 possible ways of ordering (look at vertex of the column j=2of graph), in the nodes the column j=2(second stage) the date of inventory on the end second stages is stored.
- In the last stage, some volume of *required products* is ordered. There are the 7 possible ways of ordering (look at vertex of the column j=3of graph) to obtain the desired objective or result, in the nodes the column j=2(second stage) the date of inventory on the end second stages is stored.
- Solve model according to the method of the dynamic programming is considered from the terminal node of column j=3 before initial node the column j=0 of graph.
- According to (2) formula there are 7 possible ways of conditional optimal profit between P2 and P3(look at graph-diagram), for example, one of them: the way 1 - 0 is defined: in P2 stored 1 unit production , the require product calculated by the formula (2):
- 0=1+x-4 and get the value 3, relevant cost calculated by the formula (1) and gets the value 6. Analogically may be considered the conditional cost between j=0 and j=1, (look graph-diagram), for example: the way 1-3defined: ordered products has a value 5, required products get the value 3 and reserved product gat the value 3, korresponding cost get the value 3+3=6.

  Let us consider j=1 to j=2: 3-2: ordered product is: 1; requared product: 2; and rezerv get the value 2. Cost :7+2*3=13; j=2 to j=3: 2-0:ordered product is: 2; requared product: 4; and rezerv get the value:0.Cost :6.

4. Solution problem of MM

In the work we have considered practical problems of optimal management of resources of company to find the tour of component construction to obtain the best available benefit given in the defined financial domain.

Let $(C_i(k_j))$ - vector of parameters the management merging. $(R_i(k_j))$ - Vector of corresponding probability of quality of construction (i=1, n;j=1,m), S- total sum construction, n-number of stages. k_j -number of components in the j-chain.It is required to construct -matrix of parameters the optimal management construction of components, satisfy the following constraint condition $\sum_{i=1}^{m} C_i(k_j) \le S$ and with regard to the following criteria of optimization: $R = \prod_{i=1}^{m} R_i(k_j) \Rightarrow \max$ For simplest case, we have considered the

ISSN 1512-1232

solution of the problem for the following particular values: s=10, (table 1).

Table 1. Table of date

k_j	J= I		J= II		J= III	
№	price-C_1	Probability-R_1	price - C_2	Probability - R_2	price - C_2	Probability – R_3
1	1	0,6	3	0,7	2	0,5
2	2	0,8	5	0,8	4	0,7
3	3	0,9	6	0,9	5	0,9

We will gate to the following corresponding multistage graph of dynamical programming (Fig. 2)

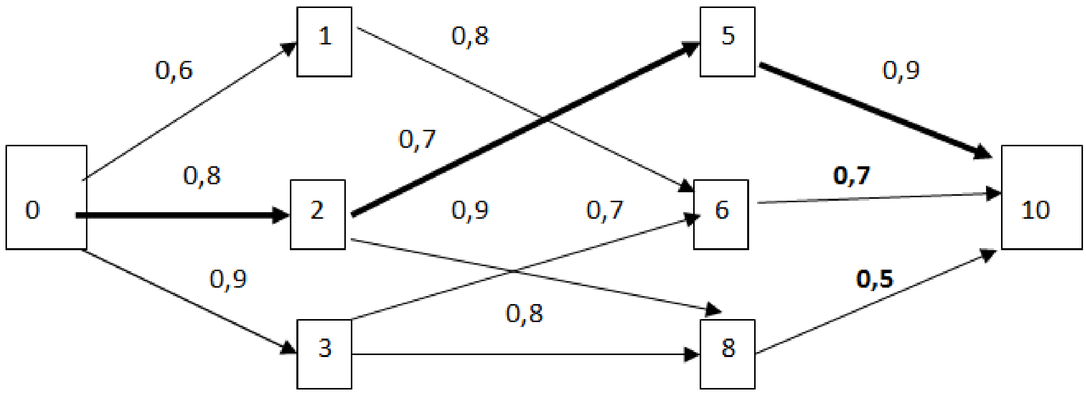

Fig. 2 Dynamical Programing graph of optimal construction components

The vertices ($V_{jk,}$ where j is number of stage) of graph are used for storing the sum of cost of select component on the given stage (from the beginning to the end). The edges of graphs are labeled by the probability of corresponding stage of merging components. By the dynamical programming algorithms the following components are selected : k_{11}=2, k_{22}=1, k_{33}=3. Result of optimal cost of probability of component construction get the value: $R=\prod_{i=1}^{m} R_i(k_i)^* = 0.8*0.7*0.9 = 0.504$. Let us consider greedy algorithm approaches. Accordingly of table 1. We will get the following table 2.

Table 2. Initial data of greedy algorithm

k_j	Benefit-Y_1	Benefit-Y_2	Benefit-Y_3
1	0,6	0,2	0,25
2	0,4	0,16	0,17
3	0,3	0,15	0,18

where $Y_i(k_j)=R_i(k_j)/C_i(k_j)$.

Result of greedy strategy we can illustrate by coloring graph:

ISSN 1512-1232

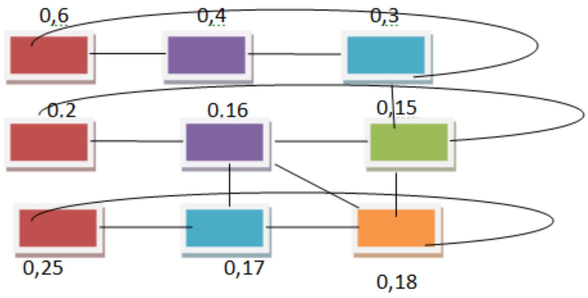

Fig. 3. Coloring Graph of Greedy Algorithm

The following components are selected by greedy algorithms: $k_{11}=1$, $k_{21}=1$, $k_{31}=1$. Result of optimal cost of probability accordingly Table 1. Get the value:

$$R = R_1 k_1 * R_2 k_1 * R_3 k_1 = 0.6 * 0.7 * 0.5 = 0.21$$

Pseudo code for *greedy* algorithm has the form:

```
Pseudocode- greedy()
{
G[m], list< vertex > Clr;
for (Vertex =1; Vertex<m; Vertex ++)
     {
      for ∀v ∈G[Vertex]
      found=false;
     }
      for ∀ w ∈ Clr (Vertex)
     {
      if  v∈Adj(w )
             {
      found= true
             }
     else if found=false
     {
     Clr.poush-back.(v)
     }
     }
}
```

5. Conclusion

In this work dynamic programming and greedy algorithm strategy for solving optimization problems of (SC) models is investigated. The particular examples inventor management and optimization problems for component construction are considered. Approximation algorithms solved multistage problems are constructed.

References

1. Jeremy C.Stein, " Internal Capital Markets and the Competition for Corporate Resources". DOI: 10.1111/j.1540-6261.1997.tb03810.x
2. Basheleishvili, I. (2018). The Algorithm of Selection and Functions Distribution of Multifunctional Personnel-Case when the Number of Functions is Greater than the Number of Personnel. Journal of Technical Science and Technologies, 6(2).
3. Chumburidze, M., Bahseleishvili, I.,(2018). Dynamic Programming and Greedy Algorithm Strategy for Solving Several classes of Graph Optimization Problems. Broad Research in Artificial Intelligence and Neuroscience- BRAIN. 9(3).
4. Berge C.The Theory of Graphs and its Applications, Wiley, N.Y.1962.
5. Sniedovich, M. (2010), Dynamic Programming: Foundations and Principles, Taylor & Francis, ISBN 978-0-8247-4099-3
6. G.Bang-Jensen, G. Gutin and A. Yeo, When the greedy algorithm fails. Discrete Optimization 1 (2004), 121–127
7. G. Bendall and F. Margot, Greedy Type Resistance of Combinatorial Problems, Discrete Optimization 3 (2006), 288–298

Article received: 2018-06-21