RESEARCH ARTICLE OPEN ACCESS

Challenges and Prospects of using Object Oriented Programming in Solving Linear Programming Problems

F. U. Onu [1], C.L. Nwagbo [2] and R.E. Obiadazie [3]
Department of Computer Science [1]
Ebonyi State University, Abakaliki
Department of Computer Science [2]
Nwafor Orizu College of Education Nsugbe
Department of Computer Education [3]
ChukwuEmekaOdumegwuOjukwu University Anambra State
Nigeria

ABSTARCT

In the world where we live today, resources have continued to dwindle in supply. Partly due to a continuous increase in demand for the resources without a corresponding increase in the supply of such resources. For this reason, efficient distribution of such limited resources for optimum use became a daunting challenge which Linear Programming is the most suitable technique to surmount. This paper presents the challenges and prospects of using Object-Oriented Programming (OOP) in computing the solution to linear programming problems. The study used the three methods of solving transportation problems namely: Northwest corner method; Minimum cost method, and Vogel's approximation method to show that OOP has been the easiest way of computing linear p rogramming problems. The study also showed that most transportation problems solved using linear programming was replicated using C++ programming language which is an OOP. In conclusion it was deduced that notwithstanding the weaknesses of OOP, it is still the easiest and fastest programming technique for solving linear programming problems.

Keywords:- Comparison, Linear programming (LP), Object Oriented Programming (OOP), Prons and Cons, Transportation problem.

I. INTRODUCTION

Object oriented programming systems, is considered a fifth generation programming, with usage in the mid-80's out of necessity to meeting the requirements of applications beyond the data processing applications, which distinguished linear programing systems (fourth generation programming technology).

Linear programming is a mathematical technique for optimum distribution of limited or scarce resources, such as labour, material, machine, money, energy etc. to numerous competing activities such as products, services, jobs and so on, on the basis of a given criteria of optimality. Challenges in the use of linear technology for complex applications such as: Computer Aided Design (CAD), Computer Aided Manufacturing (CAM), Software Engineering, Knowledge Based Systems and Multimedia Systems, rapidly exposed the complexity of linear programming systems [1]. The need to perform complex operations on an existing system and a new generation system (applications) generated a need sthat would be better satisfied by Object Oriented Programming system (OOPs).

The first and major purpose of this work is, solving linear programming (LP) problem using some of transportation methods. The second purpose is solving of transportation problems via Object Oriented Programming. C++ programming language is used to achieve the objective/solution. Results obtained from both LP and Object Oriented Programming solutions are compared.

The methods used to solve transportation problem are:

(1) Northwest Corner method
(2) Minimum Cost method

(3) Vogel's Approximation method

Many definitions of Object Oriented Programming have been developed over the years.[2][3][4]

II. DEFINITION OF OBJECT ORIENTED PROGRAMMING

Object Oriented Programming is a system that allows a more direct representation and modelling of real world problems, which ensure persistence and concurrent sharing of information in applications.

III. COMPLEXITY OF COMPUTING USING LINEAR PROGRAMMING

The term "Linear" is used to describe the proportionate relationship of two or more variables in a model. An amendment/change in one variable will always cause a resulting proportional change in another variable.

In an OOP, the value of an attribute of an object X, whose domain is a new object Y, is the object identifier (OID) of the object Y. Therefore, if an application has already retrieved object X and now would like to retrieve object Y, the object oriented system may recover object Y by looking up its OID. If the OID is a physical address of an object, the object may be retrieved directly; if the OID is logical address, the object might be fetched by looking up a hash table entry (For instance, in a system which maintains a hash table that maps an OID to its physical address) [5]. This would not likely be so easy in LPs, since LPs do not maintain object identifier data system (OIDs).

A. Performance gain in OOPs over LPs

Basis of performance gain in OOPs over LPs is that most OOPs covert the OID stored in an object to memory pointers while the object is loaded into memory. Since LPs do not store OID, they cannot store memory pointers. The capability to navigate through memory resident objects is a basically absent feature in LPs. The performance drawback that results from it cannot be neutralized by simply having a large buffer gap in memory. Therefore, for applications that require constant navigation through linked objects loaded memory, OOPs can significantly outperform LPs [5]. Also, even if OOPs are not indexed, it may be suitable to execute arbitrary queries, which is impossible for linear programming.

IV. THE OBJECTIVES OF THE STUDY

- To identify some significant differences in computing using object oriented over linear programming.
- To examine some Prons and Cons of OOP.
- To find out some areas of complexity in linear programming.
- To compare the two programs (OOP and LP).

V. PRONS AND CONS OF OBJECT ORIENTED (OOP)

Object Oriented Programming offers a lot of benefits to both the program designer and the user. Object Oriented Programming contributes to the solution of several problems associated with the development and the quality of software products. The new technology promises better programmer productivity, better software quality with less maintenance cost.

B. The *major advantages of Object Oriented Programming are :*

- Through inheritance redundant codes where eliminated and use of existing classes can be expanded.
- Programs can be designed from the standard working modules that communicate with one another, rather than having to begin writing the code from scratch.
- It pilots saving of development time and higher productivity.
- The principle of data hiding assists the programmer to build secure programs that cannot be attacked by code in other parts of the program.
- It is likely to have multiple objects to coexist without any interference.

- It is possible to plot objects in the problem domain to those objects in the program.
- It is simple to partition the work in the projects based on objects.
- Software complexity can be easily managed.

C. Cons

In spite of the achievements of OOPs stated above, OOPs have not been able to make major impact because of some weaknesses still existing in OOP model and technology. In OOPs, there is lack of basic features which users of the system has become adapted to. Such features include:

- Lack of interoperability between LPs and OOPs.
- Minimal query optimization.
- Lack of standard query algebra.
- Lack of query facilities.
- No support for views.
- Security concerns.
- No support for dynamic class definition changes.
- Limited support for consistency constraints.
- Limited performance tuning capabilities.
- Limited integration with currently existing Object Oriented Programming systems.

1) ***Interoperability between LPs and OPPs:*** For OOPs to make a major impact on the programming market, the following has to be met:
   - OOPs should be made full-fledged database systems, adequately compactible with LPs- a migration path is needed to permit the coexistence and the gradual migration from the existing products to new products.
   - Architectures of LPs and OOPs have to be unified.
   - The data models of LPs and OOPs have to be unified [5].
   - The critical comparison of OOP to other technologies, Linear programming in particular is complex because of lack of an agreed upon and rigorous definition of OOP.

2) ***Minimal query optimization:*** A major problem in OOPs is the optimization of declarative queries. The additional complexity of the object oriented programming (OOP) complicates the optimization of OOP queries. This additional complexity is due to:

- Additional data types- the user definition of new types and classes through inheritance can both support and thwart optimization of queries.
- Complex objects methods and encapsulation contribute to the complexity of query processing in the OOPs. Complex objects create path expressions that obscure query processing.
- OOPs query languages support the use of nested structures, which may gain highly worse the optimization process, turning it from a local problem to a global one requiring global knowledge of the whole query expression.

3) ***Lack of standard query algebra:*** Lack of query algebra standards is an additional major weakness of OOPs. Which also impairs query optimization. Several diverse formal query languages of algebra and calculi which have been proposed for OOPs [6][7][8]. The majority of these algebras are variable based, i.e. variables used for temporary

results. However, in OOPs there is analogous of execution plan generation, should first define the low level object manipulation primitives [2].

4) ***No support for views in OOPs:*** OOPs do not support views. Although there have been numerous proposals [1][8][9]. There is light

agreement as to how a view mechanism should operate in OOPs. The improvement of an object oriented view capability is complicated by such model features as object identity. What are those identities of the object in a view? Additionally, there has also been the argument that data encapsulation and inheritance make explicit view definitions unnecessary [10].

5) ***Security concerns with OOPs:*** Though LPs support authorization, most OOPs do not support authorization [5]. LPs allow users to donate and revoke privileges to read or change definitions and tuples in relations views [5]. If OOPs are going to develop into more business oriented fields, this features has to be improved. Some OOPs require users to openly set and release locks. LPs automatically put and release locks in user processing query and update statements [5].

6) ***No support for dynamic class definition changes with OOPs:*** In addition to the reality that no single standard data model has yet been developed for OOPs, most OOPs do not permit dynamic changes to the database schema, such as adding a new attribute or method to a class, adding a new superclass to a class, dropping a superclass from class, adding up a new class, and dropping a class. LPs allow the user to dynamically change the database schema by means of ALTER command; a new column may be added to a relation, a relation may be dropped, a column can at times be dropped from a relation [5].

7) ***Limited support for consistency constraints in OOPs:*** There are no mechanisms to assert key properties of attributes (for example, an attribute of a class cannot be declared the primary key of the class) or uniqueness constraints, precisely consistency constraints would be more user friendly, less error prone, and more easily available for inspection and modification.

8) ***limited integration with existing OOP systems:*** It is difficult to re-write object oriented programs for constant data management. Several problems do arise:

- Naming conflict.
- Class hierarchies have to rewritten.
- OOPs trend to overwrite system operations [10].

D. Other features that OOPs do not yet support

Examples of extra features that OOPs do not yet support are triggers, meta data, management features [10], constraints such as UNIQUE and NULL [5][11].

VI. METHODS

Three methods were used to find out a solution for balanced transportation problem: Northwest Corner method, Minimum Cost method and Vogel's approximation method.

The three methods differ in the "quality" of the starting basic solution they produce. Better starting solution yields a smaller objective value. Then Illustrative example is solved by these three methods.

The Northwest-corner method see fig.3: the solution start with the northwest- corner cell route(1,1) in fig3, that is row 1 colum1 of fig3.

allocate as much as possible to the selected cell then adjust the associated amounts of supply and demand by subtracting the allocated amount.

cross out the row or Column with zero supply or demand, to signify that, no further assignments can be made in that row or column.

If both a row and a column net to zero concurrently, cross out one only and leave a zero supply and demand in the uncrossed-out row (column).

If is exactly only one row or column is left uncrossed out, stop .otherwise, move to the cell to the right if a column has just been crossed out or below if a row has been crossed out .Go to step (a). [2]

The minimum-cost method see fig4 this has a better starting solution by focusing on the cheapest routes. The method starts by a) assigning as much as possible demand to the cell with the smallest unit cost. b) Then the satisfied row or column is crossed out and the amounts of supply and demand are adjusted accordingly. If a row and a column are satisfied concurrently, only one is crossed out, the same as in the northwest –corner method c) look for the uncrossed-out cell with the smallest unit cost then repeat the process until it remains only a row or column uncrossed out . [2]

The Vogel's Approximation Method is an enhanced version of the minimum-cost method that generally produces better starting solutions.

For each row/column determine a penalty measure by subtracting the smallest unit cost element in the row /column from the next smallest unit cost element in the same row /column.

Identify the row or column with the largest penalty. Break ties at random. Allocate as much as possible to the variable with the least unit cost in the selected row or column .Adjust the supply and demand and cross out the row /column.

If a row and column are satisfied at the same time, then one of the two is crossed out, and the remaining row /column is assigned zero supply and demand.

If exactly one row or column with zero supply or demand remains uncrossed out, stop.

If one row /column with positive supply and demand remains uncrossed out, determine the basic variables in the row /column by the least cost method then stop.

If all the uncrossed out rows and columns have (remaining) zero supply and demand, determine the zero basic variables using the least-cost method stop.[2]. Else, go to step (a)

We describe the three methods (mentioned above) of transportation model in LP using the three algorithms and a flow chart for each algorithm. After designing algorithms for the three methods we develop C++ program for each one.

VII. OOP THE EASIER SOLUTION TO LINEAR PROGRAMMING PROBLEMS

Solving transportation model problem via LP: transportation model is a particular type of network problems for conveying a commodity from source (e.g. factories) to destinations (e,g. warehouse). Transportation problem involves getting the minimum cos plan to transport a commodity from a number of sources (m) to number of destination (n). using linear programming method in solving transportation problem, we determine value of objective function which minimize the cost used for transporting and also determine the number of unit that can be transported from source (i) to destination (j). If xij is the number of units moved from source (i) to destination (j). the equivalent linear programming model will be thus: [5].

The Objective function:

$$\text{Minimize}\quad Z = \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}\, x_{ij}$$

Subject to

$$\sum_{j=1}^{n} x_{ij} = s_{ii} \quad \text{fori=1,2,\dots\dots m}$$

$$\sum_{i=1}^{m} x_{ij} = d_j \quad \text{for j=1,2,\dots\dots n}$$

Then

Xij<=0 for all I to j

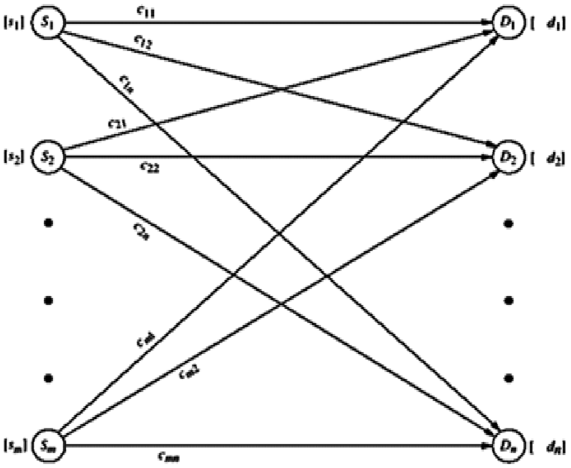
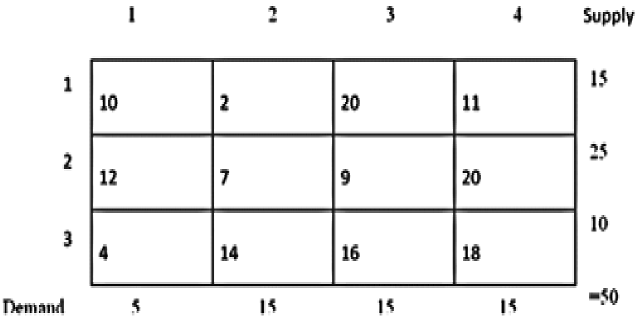


Fig 1. Transportation problem network

A transportation problem is said to be balanced if the total supply from every sources equals to the total demand within all destinations. If not it is called unbalanced.

E. Instructive Example (Sunstar Motor/Truck Transportation)

Sunstar Motor/Truck transportation company conveys truckloads of maize from three harvesters to four mills [2]. The supply (in truckloads) with the demand (also in truckloads) together with the unit transportation costs per truckload on the diverse routes are summarized in the transportation model in Fig 2.



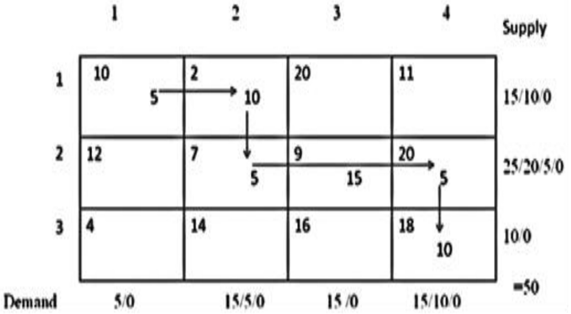
	1	2	3	4	Supply
1	10	2	20	11	15
2	12	7	9	20	25
3	4	14	16	18	10
Demand	5	15	15	15	=50

Fig 2 Transportation model of example (Sunstat Motor/Truck transportation)

The model try to find the minimum-cost of shipping schedule between the harvester and the mills. This is equivalent to determining the number xij truckload from harvester i to mill j (i=1, 2, 3; j=1, 2, 3, 4).

F. *Northwest-Corner method*

Example of the procedure to the model shows, the starting basic solution in Fig. 3



	1	2	3	4	Supply
1	10 5	2 10	20	11	15/10/0
2	12	7 5	9 15	20 5	25/20/5/0
3	4	14	16	18 10	10/0
Demand	5/0	15/5/0	15 /0	15/10/0	=50

Fig. 3 The starting solution using Northwest-corner method

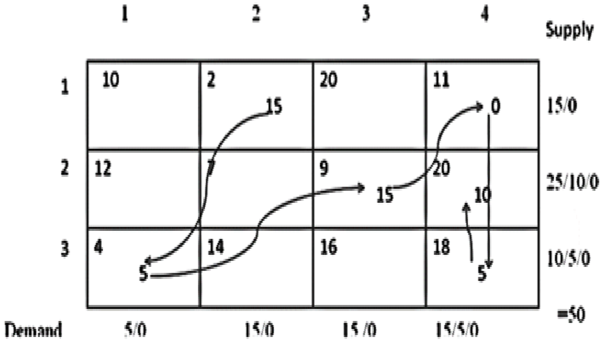
The Starting basic Solution is given as
X11=5, x12=10, x22=5, x23=15, x24=5, x34=10.
The objective function value is :
Z=5×10+10×2+5×7+15×9+5×20+10×18=$520.

G. *Minimum Cost Method*

The minimum-cost method is applied to Example (SunStar Transportation) as follows:

- Cell (1,2) has the least unit cost in the fig4 (=$2). The much that can be transported by trucks from (1,2) is (15,15)=15 which satisfy both row 1 and column 2 simultaneously, we arbitrarily cross out column 2 and amend the supply in row 1 to 0.
- Cell (3,1) is the smallest uncrossed-out unit cost (=$4).Assign x31=5, cross out column 1 because it is saturated and adjust the demand of row 3 to 10-5=5 truckloads.
- Abiding by these rules, in the same manner ,we successively assign 15 truckloads to cell (2,3),0 truckloads tocell (1,4),0 truckloads to cell(3,4),5 and 10 truckloads to cell (2,4).

The ensuing starting solution is summarized in this Fig. 4. The arrows show the order in which the allocations were made.



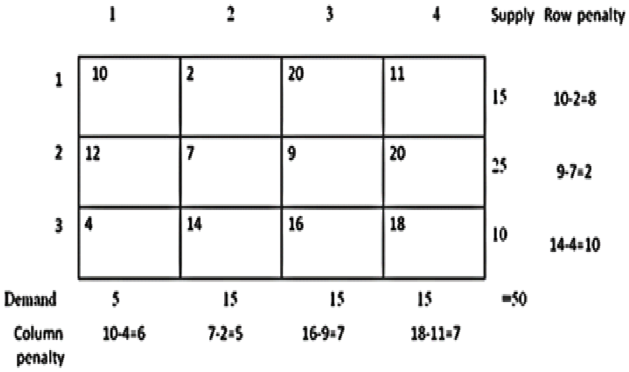
	1	2	3	4	Supply
1	10	2 15	20	11 0	15/0
2	12	7	9 15	20 10	25/10/0
3	4 5	14	16	18 5	10/5/0
Demand	5/0	15/0	15 /0	15/5/0	=50

Fig. 4 The starting solution using minimum-cost method

The starting solution (consist of 6 basic variables) which is: x12=15, x14=0, x23=15, x24=10, x31=5, x34=5.

The associated objective function value is: Z= 15x2+0x11+15x9+10x20+5x4+ 5x18=$475 .

The value of the least cost starting solution is better than of the northwest –corner method since it yields a smaller value of Z ($475 versus $520 in the northwest cornermethod



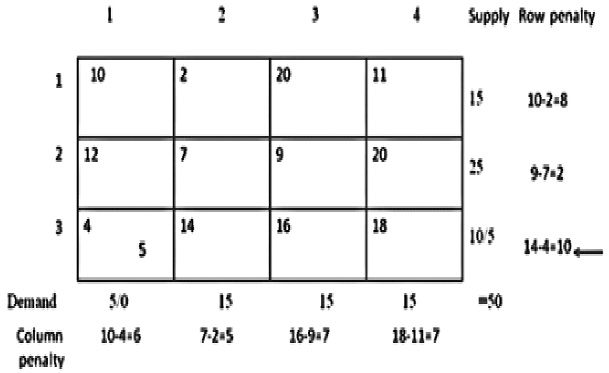
	1	2	3	4	Supply	Row penalty
1	10	2	20	11	15	10-2=8
2	12	7	9	20	25	9-7=2
3	4	14	16	18	10	14-4=10
Demand	5	15	15	15	=50	
Column penalty	10-4=6	7-2=5	16-9=7	18-11=7		

Fig. 5 Ist Step to determine the starting solution using (VAM)

H. Vogel's Approximation Method (VAM).

VAM is applied to (Sunstar motor/truckTransportation) for examples in the following manner:

- Starting from row 3 since row 3 has the largest penalty =10 and cell (3,1) has the least unit cost in the row, the amount 5 is assigned to x31.Column 1 is now satisfied and should be crossed out as in fig 6. Next, new penalties were recomputed in the Fig. 7.
- It shows that row one has the highest penalty (=9) hence ,we assign the maximum amount likely to least unit cost cell of row 1 (1,2),which yields x12=15 and simultaneously satisfies both row 1 and column 2 .we randomly cross out column 2 and adjust thesupply in row 1 to zero as in Fig. 7



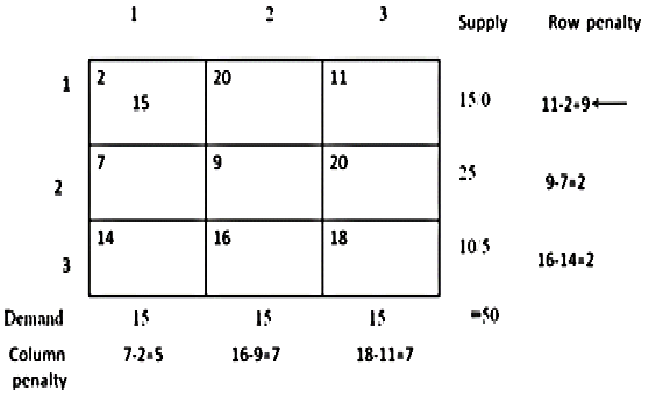
	1	2	3	4	Supply	Row penalty
1	10	2	20	11	15	10-2=8
2	12	7	9	20	25	9-7=2
3	4 5	14	16	18	10/5	14-4=10 ←
Demand	5/0	15	15	15	=50	
Column penalty	10-4=6	7-2=5	16-9=7	18-11=7		

Fig. 6 2nd Step to determine the starting solution using (VAM)

- Continuing in the same manner, row 2 having the highest penalty (=11) then we assign x23=15 ,which crosses out column 3 and leaves 10 units in row 2 . then just column 4 is left , and it has a positive supply of 15 units .

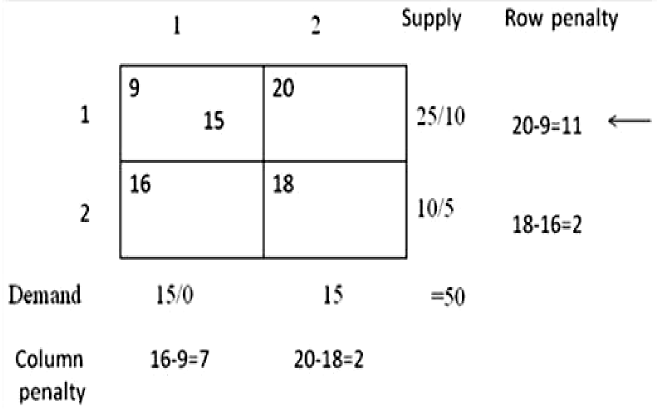
Using the least-cost method to that column, we successively assign x14=0, x34=5and x24=10 (verify) as in Fig 8.

The associated objective function value is: Z=15×2 + 0×11 + 15×9 + 10×20 + 5×4 + 5×18=$475. VAM produces a well gain starting Solution.



	1	2	3	Supply	Row penalty
1	2 15	20	11	15/0	11-2=9 ←
2	7	9	20	25	9-7=2
3	14	16	18	10/5	16-14=2
Demand	15	15	15	=50	
Column penalty	7-2=5	16-9=7	18-11=7		

Fig. 7 3rd Step to determine the starting solution using (VAM)



	1	2	Supply	Row penalty
1	9 15	20	25/10	20-9=11 ←
2	16	18	10/5	18-16=2
Demand	15/0	15	=50	
Column penalty	16-9=7	20-18=2		

Fig. 8 *Step4 to determine the starting solution using (VAM)*

VIII. OBJECT-ORIENTED PROGRAMMING

Object-Oriented Programming (OOP) is a method of completion in which programs are organized as cooperative collections of objects and every object is an instance of some classes, classes are related to one another via inheritance relationship [8]. Within Object-Oriented programming, the data and functions are integrated. An object is like a box holding it data and its functions which can operate on the data [9]. Object-Oriented programming languages Provides large flexibility, clarity and reusability through inheritance. It pilots to faster software development, increased quality, easier maintenance, also flexible modifiability.

Objects are the essential elements for executing object oriented programs while classes are the vital elements for defining object-oriented programs. If any of these elements is lost, it is not an object-oriented program [9]. object oriented programming languages includes java, C# and C++ etc.

IX. ANSWERING TRANSPORTATION PROBLEM USING C++ LANGUAGE

We have to describe the three methods (mentioned above) of transportation model in LP. We used C++ language to facilitate obtaining the result and the complex problems which take longer time using LP solution. comparison between each solution using C++ program and LP solution confirmed that the same result then compare among different solutions for choosing less value of the objective function.

The result of northwest-corner method program by C++ language is the cost of transportation =$520. The number of units transported starting from source i todestination j .

```
we transport
supply [0] to demand[0] =5
supply [0] to demand[1] =10
supply [1] to demand[1] =5
supply [1] to demand[2] =15
supply [1] to demand[3] =5
supply [2] to demand[3] =10
Press any key to continue
```

The result of minimum-cost method program by C++ language is the cost of transportation =$475. The number of units transported from source i to destination j

```
we transport
supply [0] to demand[1] =15
supply [1] to demand[2] =15
supply [1] to demand[3] =10
supply [2] to demand[0] =5
supply [2] to demand[3] =5
Press any key to continue
```

The result of vogel approximation method program by C++ language is the cost of transportation =$475

The number of units transported from source i to destination j

```
we transport
supply [0] to demand[1] =15
supply [1] to demand[2] =15
supply [1] to demand[3] =10
supply [2] to demand[0] =5
supply [2] to demand[3] =5
Press any key to continue
```

The results of the programs using C++ language are equal to LP solution only that the solution using C++ language is faster and easier than LP solution.

X. DISCUSSION AND EVALUATION

Computation with northwest corner method is quick method due to it takes short time but yields a bad solution because it is very far from optimal solution. Vogel’s approximation method and Minimum-cost method is used to obtain the shortest road. Advantage of Vogel’s approximation method and Minimum-cost method yields the best starting basic solution because it gives initial solution very close to optimal solution but the solution of Vogel’s approximation methods is slow because computations take long time. The cost of transportation with Vogel's approximation method and Minimum-cost method is less than north-west corner method.

We used C++ language to facilitate getting the result and the complex problems which take long time using LP solution. After running these programs we compared between each solution using C++ program and LP solution which show that the same result are compare between different solutions for choosing less value of the objective function.

The main ideas from designing three C++ programs is to show the good techniques of oop language in handling complexity in leaner programming. It also saves time, money, and effort.

In the example (*Sunstar Transportation)* the three C++ programs is to minimize the cost of transportation and determine the number of units transport from source i to destination j.

The results are shown as follows.

The result of northwest-corner method program by C++ language is the cost of transportation =$520 then number of units transported from source i to destination j .

XI. CONCLUSION

The results/output from Object Oriented programming has led to the conclusion that Object oriented is one of the most important concepts in programming which assists the programmer to develop programs more effectively and efficiently. The major feature is the flexibility of classes and objects which saves programmer’s time. It provides several features such as data abstraction, encapsulation, polymorphism, inheritance which has made it possible for the object oriented programming model to be used in the real world situations. Furthermore, it is much easier, and thereforea shorter way of computing LP problems. In other to become a successful programmer, one must go through the concept of Object Oriented programming thoroughly.

REFERENCES

[1] Khoshafian, S, *Object-Oriented Databases* (New York: John Wiley & Sons, 1993).

[2] Straube, D.D and Ozsu, M.T(1991). “Execution Plan Generation for an Object-Oriented Data Model,” *Proceedings of the 2nd International Conference onDeductive and Object-Oriented Databases*, Munich, Germany, 1991, pp. 43

[3] Kim, W., *Introduction to Object-Oriented Databases* (Cambridge, MA: The MIT Press, 1995).

[4] John S, McGregor, David A. S, *A Practical Guide to Testing Object-Oriented Software* (USA, Addison- Wesley Professional,2001).

[5] Kim, W., Object-Oriented Database Systems: Promises, Reality, and Future in *Modern Database Systems: The Object Model, Interoperability and Beyond*, (Addison Wesley, ACM Press, 2000) 255-280.

[6] Prem K. G and Hira D.S., Operations Research”, An Introduction (Delhi, S.Chand and Co., Ltd, 2002).

[7] Banerjee, Queries in object oriented databases, in *Proc. IEEE Data Engineering Conf.*, Feb.2002.

[8] Grady B., “*Object-Oriented Analysis And Design*, (USA, NJ: Prentice-Hall, 2002)

[9] Peretz S., *functional and object-oriented analysis and design (an integrated methodology)* (USA, Idea Group Inc, 2006).

[10] Dittrich, K.A., and Dittrich, K.R, *Achievements and weakness of object-oriented databases,journal of object technology vol. 2, No.4*, 2000 pp. 238-252.