BirEX JOURNAL
Budapest International Research in Exact Sciences
Medical, Biological, Argiculture, Engineering Science and other related areas
e-ISSN : 2655-7827
p-ISSN : 2655-7835
http://www.bircu-journal.com/index.php/birex

The application of Dynamic Programming in the Ebonyi State Building Materials Industry Ezzamgbo, Nigeria

Patrick Agwu Okpara[1], Sunday Nwokpoku Aloke[2], Nnaemeka Majindu[3], Nelson Nnamdi Ezeike[4], Egwu Ogbonnia Egwu[5], Okike Abel[6]

[1,2,3,4,5,6]*Department of Industrial Mathematics and Health Statistics, David Umahi Federal University of Health Sciences, Uburu, Nigeria*

Email: alokesundayn@gmail.com

Abstract: *This study examined how the dynamic programming technique was applied to the Ebonyi State Building Materials Industry in Ezzamgbo, Nigeria, in order to maximize profits. The study looked at how effective dynamic techniques are in the industry for making decisions. The Administration, Management, Statistical, and Research department of the industry provided the data for the study. Dynamic programming was utilized in this study to model the industry data that was gathered. Concrete blocks, electric poles, and culvert rings are among the three raw materials produced by the industry in a range of sizes. After the model was solved and applied to the industry, it was determined that, in order to maximize profit, the industry should only produce 10.3632-meter high tension electric poles for eight hours. According to the findings, the production of 32 sets of 10.3632-meter high tension electric poles using the entire 8-hour machine time allotted each day yielded the highest profit of N448,000*

Keywords: *Dynamic programming, profit maximum, optimization, high tension poles, Industry.*

I. Introduction

On May 30, 1987, Ebonyi State Building Materials Industry Limited was founded in Ezzamgbo. The industry was located in Nigeria's Ebonyi State. Nigeria's four states enclose the state. Cross River State borders it on the east, Benue State borders it on the north, Enugu State borders it on the west, and Abia State borders it on the south. In this region of the country, the industry was founded to aid in the production of building materials for affordable homes. The following raw materials are produced by the industry: concrete blocks with dimensions of 0.2286 and 0.1524 meters, fancy decorated blocks, electric poles with dimensions of 10.3632, 8.5344, and 7.3152 meters, and convert rings with dimensions of 1.0000, 0.8000, 0.6000, and 0.4000 meters. The state makes money by selling these goods to both domestic and foreign consumers. For the highest level of material production, the industry has a large number of machines. Therefore, dynamic programming which is quicker, less computationally intensive, and more accurate in providing the best solution to the problem was used to solve the profit maximization problem in the products of the Ebonyi State Building Materials In- dustry Limited Ezzamgbo [2], [10]. American mathematician Richard Bellman developed the crucial mathematical programming technique known as dynamic programming in the 1950s [12]. It is a mathematical solution technique used for making a series of decisions [4], [11]. It can convert an n dimension decision problem into multiple one dimension optimization problems and solve each one separately in any given dynamic programming problem [14]. In order to optimize a problem with known input parameters using the dynamic programming technique, we first divide the problem into

DOI: https://doi.org/10.33258/birex.v7i3.8108

Budapest International Research in Exact Sciences (BirEx) Journal
Volume 7, No 3 July 2025, Page: 211-218
e-ISSN: 2655-7827 (Online), p-ISSN: 2655-7835 (Print)
www.bircu-journal.com/index.php/birex
email: birex.journal@gmail.com

several decision points. Every point at which a decision needs to be made is referred to as a stage; depending on the type of decision, these stages may be finite or infinite. The information or parameter that connects the stages so that the best choice for the remaining stages can be made without having to reexamine how the decisions for the earlier stages were made is, therefore, the state variable at a particular stage. If not, the state variable serves as an input parameter for each stage. The stage is linked by the variable because, as should be noted, an input parameter to one stage produces an output parameter for that stage, which is the input parameter for the subsequent stage [1], [3]. An appropriately defined state variable ensures that the solution is workable for every state and gives the problem solver the chance to think about each step independently. The beauty of dynamic programming is the principle of optimality, which is also reached by properly defining the state. The type of problem being studied determines how the state variable changes. According to a work force size model, it might be the quantity of workers available each week [9]. In the dynamic programming model, the decision value stands for the range of options from which the best choices can be made. For a production planning model, it could be the variety of goods a business produces or the quantity among other things, the number of units of goods to be included in a cargo loading model or the year in an equipment replacement model [5]. It is common knowledge that dynamic programming (DP) decision-making is a methodical, phased process. A decision benefit equation represents every choice made at every level. Typically, this equation is referred to as the return function [6], [7]. Let's look at the n-stage DP model here, which is as follows:

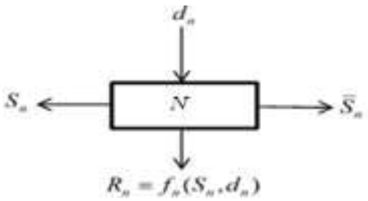

Figure 1: n-stage DP model

Where   is the input state valuable at stage
Sn is the output state (state variable at stage ;
  is the stage number,
  is the decision variables;
  =   is the return function.

Each decision made at each stage in the aforementioned model has a relative value or benefit to the system's overall effectiveness, which is reflected holistically by the so-called return function . Since it returns to the problem's objective function for each set of stage decisions, the equation   is in fact a return function. It should be noted that both the state variable   and the decision variable   are necessary for the return function to function. The choice that produces the highest (or lowest) return for a specific value of the state variable   at stage would be considered optimal. Regardless of the initial stage and decision, the remaining issue with the stage and stage resulting from the first decision is regarded as the initial condition for the optimal set of decisions in a multistage decision process [8].

Table 1. Cost of producing each unit of product

Product Dimension/type		Unit cost(N)	Unit-Quantity
Concrete blocks	0.2286m in width,	348	6
	0.1524m in width	294	6
	Fany decorate	300	6
Culvert Rings	1.000m in diameter	15 ,000	6
	8.000m in diameter	11 ,000	6
	6.000m in diameter	8 ,500	6
	4.000m in diameter	7 ,000	6
Electric poles	10.3632m	23,500	5
	8.5344m	21,000	5
	7.3152m	18,500	5

Table 2. Selling price of each product

Product	Dimension	Selling price(₦)
Concrete blocks	0.2286m in width,	70
	0.1524m in width	58
	Fancy decorated	60
Culvert Rings	1.000m in diameter	3 ,000
	0.8000m in diameter	2 ,500
	0.6000m in diameter	2 ,200
	0.4000m in diameter	2 ,000
Eletric poles	10.3632m	7,500
	8.5344m	6,000
	7.3152m	5,000

II. Research Methods

The data for this study were selected from the Administration, Management, statistical and research department of the industry. We are concerned on the cost of production and the selling price of the products. The industry gave us that six blocks of the same dimension make up a unit that is, it takes the machine a unit time to mould six blocks. Similarly, six culvert rings of the same height are unit products. The data in tables 1 and 2 respectively depict the cost of production of each singular product [13],[15]. It takes 5 minute to mould a unit of concrete blocks of the same dimension, 12 minutes to mould a unit of the same dimension, 12 minutes to mould a unit of culvert rings of the same dimension and 15 minutes to mould a unit of electric poles of the same height. The total machine hour available in a day is 8 hours. It is pertinent to note that dynamic programming determines the best solution to an n-variable problem by developing it into n-variable problem by developing it into n-stage with each stage comprising of a single variable sub-problem. In order to present a clear picture of the recursive nature of the dynamic programming techniques,we consider the optimization problem given below:

$$opt f(x)$$
(2.1) Subject to $d(x): x \epsilon R_n$

In order to solve the above equation 2.1 by dynamic programming, we first decompose it into N-stage sub-program. Thus

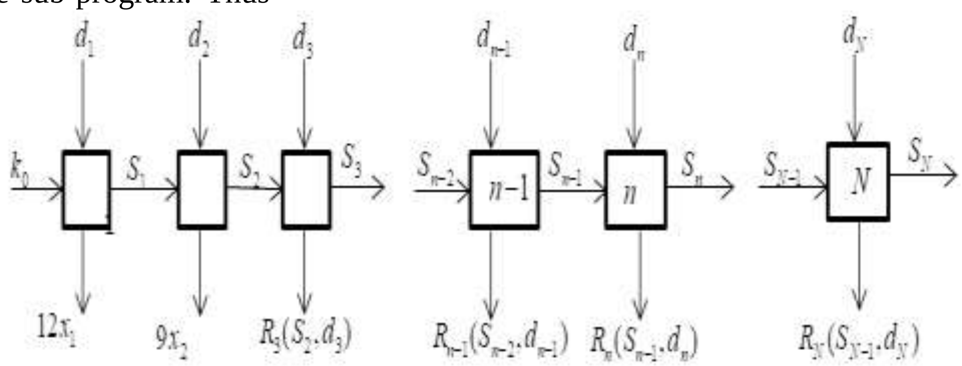

Figure 2: N-stage sub-program: Input-Output chain

Where S_i = the state variable; $i = 0,1,\dots,N$.
S_0 is the initial state while SN is the terminal state.
R_j = the stage return functions, $j = 1,2,\dots,N$
N_j = the stages, j = 1,2,...,N
d_j = the decision variables, $j = 1,2,\dots,N$

The figure (2) above is called an input-output chain. An input to a stage is the state for that stage while an output from that stage becomes a state for the next stage [6], [9].

III. Results and Discussion

In this research, we have to make a production-planning schedule that will maximize the profit of the industry. Let $x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9, x_{10}$ be the respective number of 0.2286m, 0.1524m, for concrete blocks and fancy decorated blocks

$1.0000m, 0.8000m, 0.6000m, 0.4000m$ culvert rings and $10.3632m, 8.5344m, 7.3152m$ electric poles produced by the industry respectively. The cost of producing each product is as follows:

$x1$	Blocks will cost $\frac{348}{6}x_1 = 58x_1$
x_2	Blocks will cost $\frac{294}{6}x_2 = 49x_2$
x_3	Blocks will cost $\frac{300}{6}x_3 = 50x_3$
x_4	Culvert rings will cost $\frac{1500}{6}x_4 = 2502x_4$
x_5	Culvert rings will cost $\frac{11000}{6}x_5 = 1833.33x_5$
x_6	Culvert rings will cost $\frac{8500}{6}x_5 = 1416.67x_6$
x_7	Culvert rings will cost $\frac{7000}{6}x_7 = 1166.67x_7$
x_8	Electric pole will cost $\frac{23500}{6}x_8 = 4700x_8$
x_9	Electric pole will cost $\frac{21000}{6}x_9 = 4200x_9$
x_{10}	Electric pole will cost $\frac{18500}{6}x_{10} = 3700x_{10}$

Similarly, from table 1, the selling prices of these products are as follows:

$x1$	Blocks sell at	₦ $70x_1$
x_2	Blocks sell at	₦ $58x_2$
x_3	Blocks sell at	₦ $60x_3$
x_4	Culvert rings will cost	₦ $3000x_4$
x_5	Culvert rings will cost	₦ $2500x_5$
x_6	Culvert rings will	₦ $2200x_6$
x_7	Culvert rings will cost	₦ $2000x_7$
x_8	Electric pole will cost	₦ $7500x_8$
x_9	Electric pole will cost	₦ $6000x_9$
x_{10}	Electric pole will cost	₦ $5000x_{10}$

The profit function P is gotten by subtracting the sum of the cost prices from the sum of the selling prices to be given as:

Hence the objective function is given as:

MaxP = 12x1 + 9x2 + 10x3 + 500x4 + 666.67x5 + 783.33x6 + 833.33x7 + 2800x8 + 180x9 + 1300x10. (3.1)

We consider the time constraints it takes 5 minutes to mould a set of blocks of the same dimensions, 12 minutes for culvert rings of the same dimension and 15 minutes for electric poles of the same dimension and there are a total of 8 machines hours (480 minutes) available in each day. Therefore, we must have

5(x1 + x2 + x3) + 12(x4 + x5 + x6 + x7) + 15(x8 + x9 + x10) ≤ 480 as the only contraint of equation 3.1. We can now form an optimization as

MaxP = 12x1+9x2+10x3+500x4+666.67x5+783.33x6+833.33x7+2800x8+180x9+1300x10, subject to

5x1 + 5x2 + 5x3 + 12x4 + 12x5 + 12x6 + 12x7 + 15x8 + 15x9 + 15x10 (3.2)

xj ≥ 0,j = 1,2,..., 10.

The dynamic programming problem is solved by using the recursive equation starting from the first through the last stage that is obtaining the sequence. f1 f2 f3 ... fn. The dynamic programming decomposes the optimization problem in the equation above into ten stages, where the stages are the ten different products produced by the industry. The decision variables are the time taken to produce each set of the products. Finally, the state of the problem becomes total amount of time available at each stage for the production of the item of the stage.

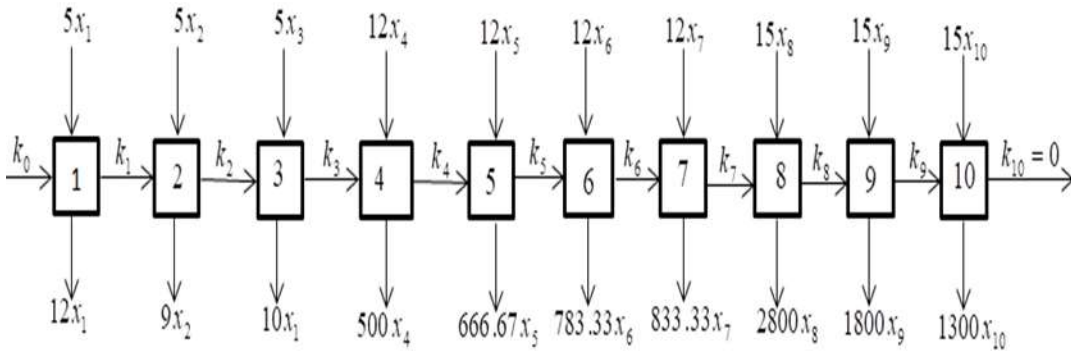

Figure 3: input-output diagram

From figure 3 above, we observe that
$k_0 = 480$, $k_1 = 480 - 5x_1$, $k_2 = k_1 - 5x_2$
So, in general
$k_i = k_{i-1} - \alpha x_j$, (3.3)
where $i = 0,1,..., 9; j = 1,2,..., 10$. We calculate the maximum profit by forward recursion. Equation 3.3 is the transformation equation. From equation 3.3, we have $k_i = 0$, whenever $k_{1-i} = 480$ and $k_i = 480$ whenever $k_{1-i} = 0$. However at points where $k_i = 0$, we have that the number of products to be produced is zero which is trivial. Hence, we turn attention to case where $k_i = 480$, working by forward recursion, since the terminal state is known to be zero in whatever case, we have the following sequence of calculations: For stage $n = 1$, we have
Max $p_1 = 12x_1$ subject to $k_0 \leq 5x_1$.
At maximum, $k_0 = 5x_1$, It implies that $480 = 5x_1 \Longrightarrow x_1 = 96$. Hence, p_1 maximum = • 12×96 = • 1152.00.
Stage $n = 2$, we have
Max $p_2 = 9x_2$ S.t $k_1 \leq 5x_2$.
S.t $k_1 \leq 5x_2$.
At maximum, $480 = 5\, x_2 \Longrightarrow x_2 = 96$ hence, p_2 maximum = • 9×96 = • 864.00
The nontrivial solution of the model equation above using the dynamic programming gives the following results as presented in the table below:

Table 3: Cost of producing each unit of product

S/N	Product	xj	unit profit(₦)	Total profit(₦)
1	0.2286m blocks	96	1,152.00	6,912.00
2	0.1524m blocks	96	864.00	5,184.00
3	Fany block	96	960.00	5,760.00
4	1.0000m Culvert Rings	40	20 ,000.00	120,000.00
5	0.80000m culvert rings	40	26 ,666.00	160,000.80
6	0.60000m culvert rings	40	31 ,333.20	187,999.20
7	0.4000m culvert rings	40	33 ,333.20	199,999.20
8	10.3632m electric poles	32	89,600.00	448,000.00
9	8.5344m electric poles	32	57,600.00	288,000.00
10	7.3152m electric poles	32	41,600.00	208,000.00

It is pertinent to note that the total profit obtained in table 3 above by multiplying the unit profit by the number of product that make the set.

IV. Conclusion

When 32 sets of 10.3632m high tension electric poles were produced using the entire 8-hour machine time allotted each day, the maximum profit realized was N 448,000.00. It was demonstrated that, in comparison to concrete blocks and culvert rings, the profit contributions of electric poles measuring 8.5344m and 7.3152m are on the higher end of the spectrum. Since it is the product that yielded the highest profit, the industry should focus on producing all types of electric poles.

References

D. Sven. Nonlinear and Dynamic programming: An Introduction. Springer-Verlag INien New York, 1975, 4th ed, pp. 60-117.

D.P. Bertsekas. Dynamic programming and optimal control. 2000, 2nd. 37-52.

E. N. Large. Dynamic programming for the analysis of serial behaviours. Behavior Research methods, Instruments and computer vol. 25, 1993, PP. 238-241.

E. O. Amien and S. A. Adedoyin. Application of Dynamic programming in decision making in Benin city, Nigeria. Sahel Analyst: Journal of Mgt Sciences University of Maiduguri, vol. 18, no. 1, 2020.

G. Robert, M. Carten and S. Peter. A discipline of dynamic programming over sequence data. Elsevier. Science of computer programming, vol. 51, 2004, PP. 215-263

K. Brajesh. Role of Dynamic programming for Data Analysis and Mathematical Application. International journal of Innovative Research in Technology (IJIRT), vol. 6, no. 11, 2020.

N. S. Nina and A.K. Evenity. Dynamic programming to Identification problems. World journal of Engineering and Technologies. vol. 4, no. 30, 2016.

N. Stokey, R. E. Lucas, and E. Prescolt Recursive methods in Economic Dynamic Harvard University. 1989, press 4th ed. 49-73.

R. Bellman. Dynamic programming. Princeton University Press 5th Ed., 1957, Pp.266- 293.

R. N. Awan, C. N. Andrew, G. P. Mario, W. Scott Wayne, A. D. Jared and E.D. Dawson. Powertrain fuel consumption modelling and bench mark Analysis of a parallel p4 Hybrid Electric vehicle using Dynamic programming. Journal of Transportation Technologies. vol. 12, no. 4, 2022.

S. Anika and P. Jitendra. Dynamic programming: A Comprehensive review of Algo- rithms, Application and Advances. Journal of Engineering Technologies and Innovative Research (JETIR), Vol. 5, no. 12, 2018.

S. M. Mathew, S. G. Henderson and T. Huseyin. Turning Approximate Dynamic programming policies for Amblance Redeployment via Direct search Stochastic system. Vol. 3, no. 2, 2013, pp. 322-361. doi:10.1214/10-ssy020

S. M. Mirmohseni, S. H. Nasseri and M. H. Khaviari. A new fuzzy hybrid dynamic programming for scheduling weighted jobs on single machine. Journal of applied research on industrial engineering, vol. 4, no. 2, 2017, pp. 97-115.

S. Nirmala and M. Jeeva. A dynamic programming approach to optimal manpower recruitment and promotion policies for the two grade system. African Journal of Mathematics and Computer Science Research, Vol. 3, no. 12, 2010, pp. 297-301

Z. Jingling, L. Baoyou, and W. Fuxiang. The application of dynamic programming in the system optimization of environmental problem. Proceeding of the 2nd International conference on system Engineering and Modeling (ICSEM-13), 2013