Ultimate Pit Limits using Maximum Flow Algorithm Ford and Fulkerson: The Case of Study of North Mutoshi Project, DRC

P. Musenge [a,c], E. K. Chanda [a], Bunda. B [a], J. K Kaunde K [b], F. Bokwala B [c], B. Fyama. M[c] , S. Kanke. M [c], Sebastian. A [d].

[a] Departement of mining Engineering, school of mines , University of Zambia P.O. Box 32379 Lusaka, Zambia [b] Departement of mines and geologie, Mapon University 01 kosakosa,Kindu, D.R Congo

[c] Departement of mines, Polytechnic, University of Lubumbash P.B.1825 Lubumbashi, D.R C

[d] Department of Mining, Predictive Geometallurgy and Geostatistics Lab Queen's University in Kingston, Ontario, K7L 3N6 Canada

***Abstract*:- The determination of an ultimate pit limit of an open-pit mine is an important part of the planning of any mine. This is a phase which is related to the pit optimisation. New tools to solve mathematical programming problems have emerged in recent times prompting researchers to revisit the determination of ultimate open pit limits. Application of these tools such as Python could result in considerable savings in the cost of technical computing for mining companies in this study, a proposed algorithm for Ford and Fulkerson model was developed and coded in Python programming environment to solve the ultimate pit problem and the results are compared to the Lerchs-Grossman algorithm. The research demonstrated that the Ford and Fulkerson model run in Python, can be effectively used to find an ultimate pit limit in 3-dimensions through 2-dimension cross-sections. The ultimate pit limits obtained using the Ford and Fulkerson algorithm programmed in Python found optimal pit design in record time. To validate the proposed Ford and Fulkerson model, a numerical result comparison was done against the best-known LG. The FFA determines the ultimate pit limit by finding the maximum net value from blocks extracted which represent the minimum cuts. This research provides a framework for running the Ford and Fulkerson Algorithm to the mining domain using Python along with a brief description of the method and its application to a real copper project to exemplify its use. The optimum pit values obtained from the FFA and the LG are USD 881 870 000 and USD 880 210 100, respectively. These results gave the adaptation model with the regression of 99,8 % and the net maximum value of USD 891 949 000 compared to the FFA model using python Although the proposed Ford and Fulkerson algorithm demonstrated its efficiency and applicability to deal with the ultimate pit limits.**

Keywords: Ultimate pits, optimisation, maximum flow, Graph, Ford and Fulkerson

INTRODUCTION

The mining activities have high economic risks and the determination of the optimum ultimate pit limit of an ore body greatly affects the economic feasibility of an open-pit mine, many researchers proposed different algorithms and heuristic methods that maximize the economic value of a mine while satisfying operational and extraction sequence constraints. In open-pit mining, the long-term planning problem for the exploitation of the reserve is often divided into two major tasks; the first is the determination of the ultimate pit limit and the production scheduling (Epstein et al., 2012). Optimization techniques applied to the production scheduling problems are based on the use of operations research (OR), which have been introduced in the mining industry since the work of Lerchs and Grossmann in the early 1960s (Lerchs and Grossmann, 1965). Johnson (1968) developed exact algorithms to optimize the long-term production plan in open-pit mining. The optimal solution provided by these exact algorithms may maximize the 2 objective functions but fail to define the mining extraction sequence of a large-scale mine whereby blocks must be mined according to an established time horizon. The mine production schedule optimization must comply with a set of physical, technical, and economic constraints (Khan and Niemann-Delius, 2015). This study focusses on the determination of ultimate contour limits of the pit using the maximum flow algorithm developed by Ford Fulkerson. In the last decades, there have been two major approaches to determining the final pit limit. The first approach is undiscounted profit maximization and the other is maximizing the NPV of the ultimate pit limit. For each approach, some methods and algorithms are presented. In the first approach, initially, the final pit outline to maximize undiscounted profit is determined. Then achievement of the highest net present value (NPV) is planned for the pit production scheduling. Heuristic algorithms such as the floating cone (Pana, 1965) and its improved methods (Wright, 1999) and Korobov (David, Dowd, & Korobov, 1974) were presented for this purpose. The Lerchs-Grossman (1965) algorithm (LG) based on graph theory and the network flow algorithm (Johnson & Barnes, 1988; Yegulalp & Arias, 1992) also determine the final pit through a mathematical approach. Of these, the LG method is the most widely used when designing the final pit limit. Currently, a variety of software programs are being used to solve the problem, each of which has challenges in terms of acquisition, learning and application, all of which require a high degree of mining knowledge, as well as vast experience and high-level skills in computer applications. As a result, existing mathematical methods for handling the mining problem that requires less dependence on software have been introduced and are being enhanced. (Meisam Saleki & Kakaie, 2019).

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 11 Issue 02, February-2022

BACKGROUND AND FORD AND FULKERSON ALGORITHM

ULTIMATE CONTOURS

It is not so long ago that operations research has been used to solve the problems of planning operations in surface mines. One of the first problems which one tried to solve in this field was the determination of the ultimate contours of a pit operation. The ultimate contours of an open-pit mine represent the geometric contours of the mine after its operation or they represent the appearance of the mine at the end of its economic life. Optimal contours are those that maximize total profits regardless of time. This is a long-term planning problem and allows, among other things, to estimate the quantity of ore that it will be possible to extract from the deposit, to estimate the economic duration of the exploitation, to plan the size of the surface installations and the machinery, and to plan the capacity of production. Knowing the final contours also makes it possible to plan and organize short, medium, and long-term operating sequences that will maximize profits.

Several people have tried to design an algorithm that would optimize the final contours of an open-pit mine. The heuristic methods were the first methods developed to determine the final contours of the pit operations. From heuristics developed for the problem of ultimate contours, those which use the moving cone method or, floating cones, are the best known and have been the most widely used in industry because of their simplicity both in terms of understanding and implementation. It was Pana, in 1965, who first proposed this method and it remained for a long time the effective tool of design for open pit mines. Up to everything recently, several authors have been interested in this method. The second type of heuristics is based on the concept of dynamic programming. These methods originated from the 2-D algorithm of Lerchs and Grossmann and were then developed to deal with the three-dimensional problem.

In 1965 Lerchs and Grossmann were the first to present an optimal method for the problem of ultimate pit contours. Based initially on graph theory, Lerchs and Grossmann's algorithm was taken up first by Johnson (1968) then by Picard (1976) who demonstrated that we could model the problem as that of a maximum flow problem in a graph.

Lerchs and Grossmann 3-D graph theory algorithm

Lerchs and Grossmann developed the first optimal three-dimensional method for determining the ultimate contours of an open-pit mine, this is the second part of their 1965 article. The Lerchs and Grossmann 3D algorithm is separated into two parts.

The first part is used to build the graph and the second part is used to calculate the maximum weight closure on the graph. During the first part, you build the initial graph by drawing oriented arcs between each of the nodes and the 9 nodes which are above it. Suppose, the two-dimensional deposit presented in Figure 1:

j / i	1	2	3	4	5	6	7
1	-3	-2	-3	-3	-3	-3	-4
2		2	7	3	8	3	

Figure 1: Example of deposit in 2Dimension (E. Chanda, 2021)

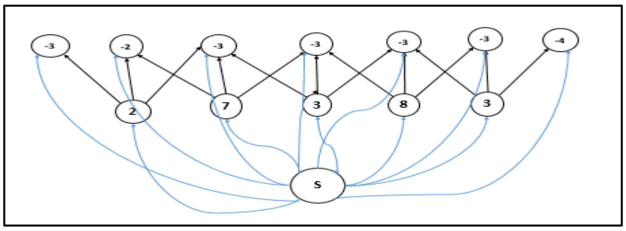

Figure 2: Graph and The initial tree graph representing the constraints of precedence between blocks (E. Chanda, 2018)

Each of the blocks in figure 1 becomes a node of the graph. Each node of the graph is associated with a value represented by the cost or the profit generated by the extraction of the corresponding block. We can associate with this deposit the graph of precedence then must connect all the nodes to an artificial node named S. Thus, a tree is formed where the artificial node s is the root of the tree. Figure 2 shows the initial tree thus created.

The second part of the algorithm consists in finding the maximum closure on this graph. Even though Lerchs and Grossmann's algorithm has been proven for several decades now and is the benchmark algorithm from which all newly developed algorithms are ranked and judged, it was hardly ever used in the industry before the mid-1980s. The main reasons for this phenomenon were practicality, over years, however, it has been remodeled and modified frequently to make it more efficient, Stuart (L992), Zhao and Kim (1992). They have all, at some level, succeeded in overcoming certain limitations mentioned above such as incorporating variable slopes during calculations and reducing resolution times; with the advent of commercial software Whittle, the Lerchs and Grossmann method has become increasingly popular in the industry, from the mid-1980s until now, many years after the release of the original article in 1965. For a long time, the different methods developed over the years to determine the ultimate contours were not favored in the industry because of the implementation problems that they presented, such as their inability to take into account variable slopes for different parts of the pit.

MAXIMUM FLOW ALGORITHM

In 1965, when Lerchs and Grossmann proposed their algorithm, they specified that the problem could be dealt with in several different ways, either by using the dynamic programming, or using graph theory (maximum closure of a directed graph), either as a flow problem or as an analogy hydrostatic (project management problem). In 1968, as part of his doctoral thesis, Johnson was the first to recognize the relation between the problem of the ultimate contours and the maximum flow. In 1976, Picard demonstrated mathematically that the method presented by Lerchs and Grossmann, which consisted of finding the maximal closure on a graph, was equivalent to finding a maximum flow on a graph adapted to the problem of contours ultimate. The new graph is obtained by adding a source node *s* and a node sink *t*. The source node is connected to all the nodes *i* which have a positive value by an arc oriented *(s, i)* of capacity equal to the value of the node. The sink node is connected to all nodes *j* which have a negative or zero value by an oriented arc (*0*, *i*) of capacity equal to the absolute value of the node. In this new graph, the arcs of precedence have an infinite capacity. Figure 3 presents the Picard graph constructed from the deposit presented in Figure 2 in the section on Lerchs and Grossmann 3D.

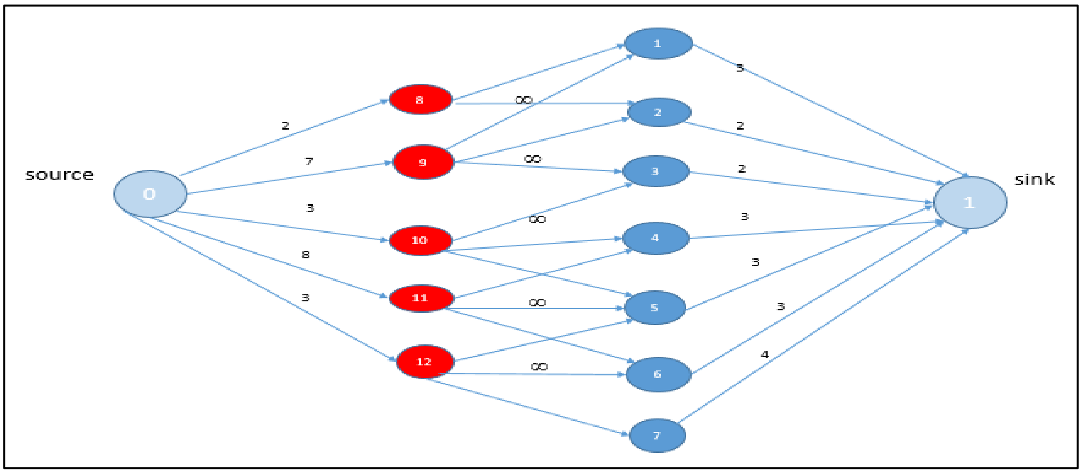

Figure 3: Graph for the calculation of ultimate contours with maximum flow formulation for Ford and Fulkerson Algorithm Chanda, 2019)

THE FORMULATION OF THE MAXFLOW ALGORITHM AND METHODOLOGY

firstly we need to have the economic block value, The equations to compute the value of a block are given as follows (Kubra, 2019): $EBV = R(P - S)gt$ (1)

$R - M_c t - P_c t)$ for the ore blocks (2)

$-M_c t$ for waste bocks

where R is revenue, P is price, S is selling cost, g is grade, t is the tonnage of the block, EBV is economic block value, M_C is mining cost, and P_C is processing cost.

CONCEPTS OF THE MAXIMUM FLOW ALGORITHM FORD AND FULKERSON

The ij blocks are the blocks that make up the economic block value for the Mutoshi deposit, and five primary sections can be explained as follows: the first step concerns the mineral resources modeling and calculation; the next step concerns block aggregation then the software surpac was used for providing the economic block model then translate the problem to optimized with max flow algorithm Ford and Fulkerson using python programming. To carry out the research of this work, two types of methodology, theory and practice. The theory part includes some very advanced theories and The practical part requires knowledge of the several tools and mining software

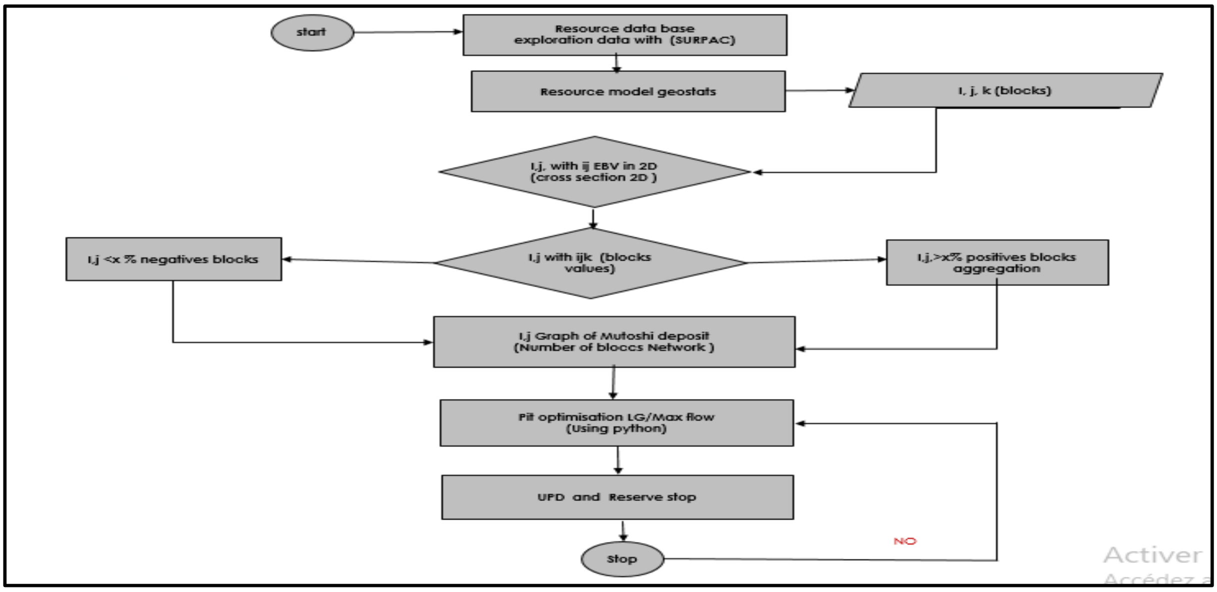

Figure 4: Flowchart of the proposed maxflow algorithm for solving the ultimate pit limit problem

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 11 Issue 02, February-2022

FORD AND FULKERSON ALGORITHM IN THE CONTEXT OF THE UPLO

According to (Goldberg & Tarjan, 2014)The FF algorithm was developed to improve the augmenting paths work and proved to give better results experimentally when compared to other algorithms methods. With the augmenting paths, a new breath-first search is normally started from the source (s) to sink (t) paths once all the pre-examined paths are exhausted. This search in UPLO can be achieved by scanning the blocks (vertices) within the block model. Since it can be very costly to repeat the whole search process every time, the FF algorithm builds two search trees, where one starts from the source and the other from the sink. The algorithm also reuses these trees in their search instead of starting afresh, thus saving on time. The algorithm works by maintaining two search trees T and S to give a min-cut, which are rooted at the sink node (t) and the source node (s), The algorithm iterates three main steps:

Growth step: this is the step where the tree grows by linking the active nodes in the tree to the free nodes whose edges are not saturated until an augmenting path is found. In UPLO, the blocks will be linked depending on the set of blocks that are required to be mined to pave the way for mining a certain ore block. This is achieved by linking the active nodes of the two sets of trees S and T.

Augmentation step: this step enhances the augmenting paths found in the growth step by trying to push maximum flow through the edges such that some edges become saturated.

Adoption step: with the creation of a forest of trees from the augmenting step, the adoption step restores the original setup of S and T trees. This is achieved by looking for new parents with non-saturated edges for the visit from the same tree they have come from. If the visited do not get new valid parents, they become free nodes. This step ends when all the visited cease to exist and the original S and T trees are left. Once the adoption step ends, the algorithm starts again at the growth step until the time when all the active nodes phase-out, thus achieving a maximum flow. Then the min-cut will be created with all saturated nodes and that will be the maximum closure.

APPLICATION OF MAX FLOW ALGORITHM FOR THE CASE STUDY

Formulation of the network max-flow model for Mutoshi deposit

Let $G = (V, A)$ be a direct graph network with $n = | V |$ the number of nodes and with $m = | V |$ the number of arcs. Let with $(u, v) \in A$ be a direct arc from u to v , then C_{uv} and f_{uv} represents the arc capacity (non-negative real number) and arc flow, respectively. By setting a lower bound capacity to zero, a Pseudoflow f assigns to each arc (u, v) a flow f_{uv} so that $0 \leq f_{uv} \leq C_{uv}$. An u, v -graph $G_{st} = (V_{st}, A_{st})$, corresponds to an extension of G with two additional nodes: a source s and a sink t, $V_{st} = V, U\{s, t\}$. The set of arcs A_{st} now includes source-to-node arcs A(s) and node-to-sink arcs A(t), A_{st} = A ∪ A(s) ∪ A(t). A closure graph is an s,t-graph whose arcs with finite capacities are only the arcs adjacent to the sources and sink nodes, residual graph, inflow(u), outflow(u), and excess, are required to provide the step-by-step operation of the generic Pseudoflow algorithm that maximizes the flow. (Sebastian Avalos, 2020), so the set of nodes involved in the maximum flow solution corresponds to the blocks in the resulting ultimate pit limit. (Bai, 2017). Figure 5 illustrates all details about the formulation network of the case study deposit

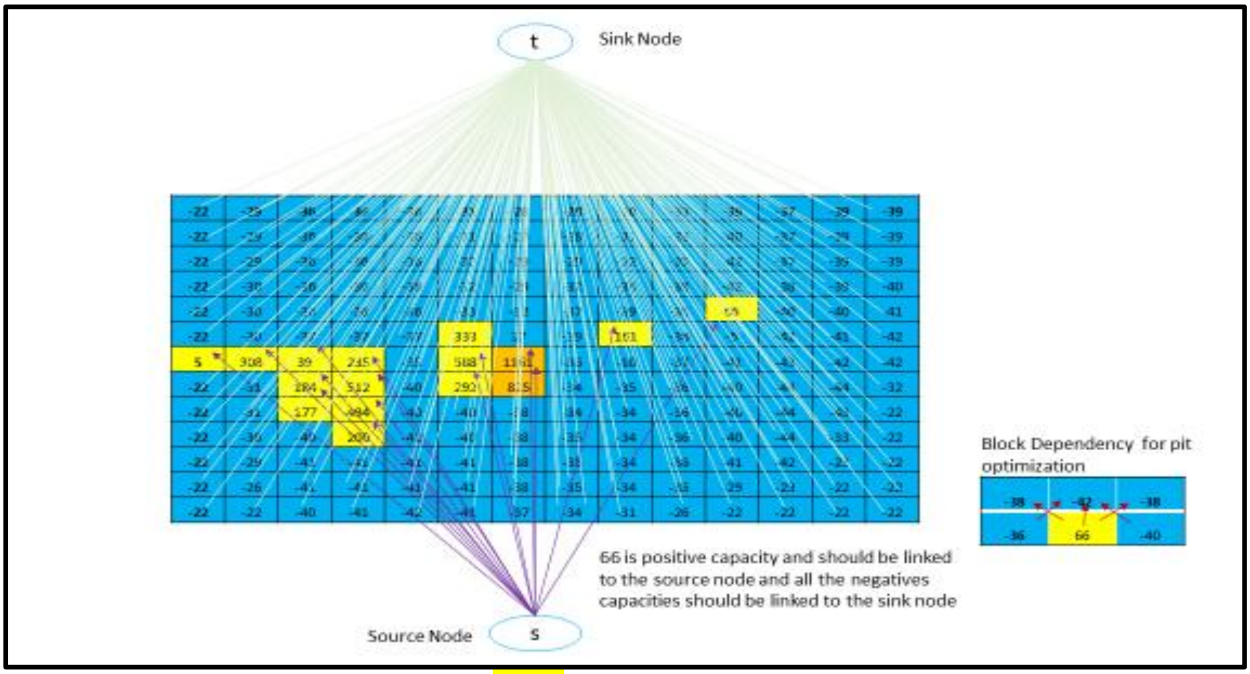

Figure 5.: Graph representing the Mutoshi block model and block dependencies (cross section 3)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 11 Issue 02, February-2022

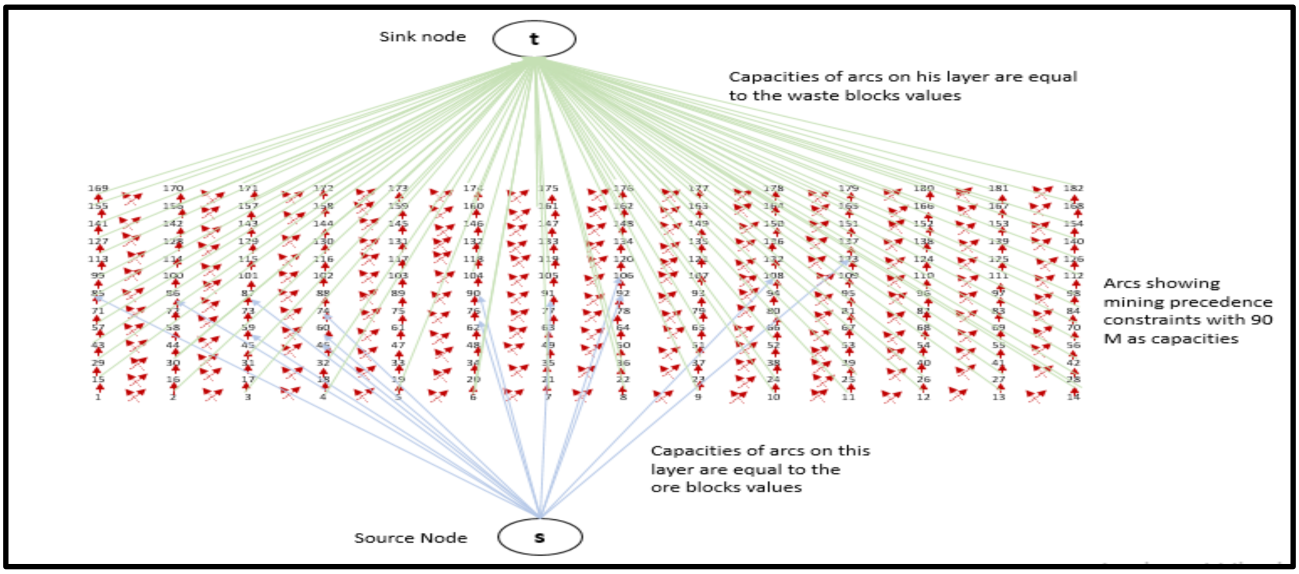

Figure 6: Graph showing the augmenting paths from source (s) to sink (t).

Assuming a maximum slope angle of 45°, the active nodes link the free neighboring nodes, which in this case are the waste blocks in blue colored that have to be mined for the ore block to be mined. The growth continues until the two trees join, as shown in Figure 6. Since the Ford Fulkerson maximum flow algorithm is an augmenting path method, it makes sure that the flow-conservation constraints and the capacity constraints are adhered to, such that: (Akisa David Mwangi, 2020), Chanda, 2018)

Adjacency Matrix for Mutoshi deposit in 2 Dimensions

The adjacency matrix of a graph $G = (V, E)$ is an n × n matrix $A_G = (a_{ij})$, where $V = \{v_1, v_2,,, v_n\}$ is the vertex set, E is the edge set of G and a_{ij} is the number of edges between the vertices v_i and v_j. In the adjacency matrix of a directed graph, a_{ij} equals the number of arcs from the vertex v_i to v_j in the current case n = 14 and 13

Table 1: Description of the Mutoshi Adjacency matrix for one section

A_G	Mutoshi Adjacency matrix on one cross-section
N	182 blocks from with source node 0 and sink node 183
$V = \{v_1, v_2,,, v_n\}$	184 vertices represent the blocks
a_{ij}	662 arcs, 480 arcs (internals) and 182 arcs (externals)
E	662 edges

n (rows)= 13 vertices n (columns)= 14 vertices

$\sum_n n(v) * 14$(columns) = 168 arcs $\sum_n n(v) * 13$(rows) = 312 arcs $\sum_n (Ev) * 1$ = 480 arcs (externals)

Development of the model

We begin by importing four python packages (or libraries): NumPy; NetworkX; pseudoflow2 and time. the full code is given in the figure below

```
# Python program for finding min-cut which represents the ultimate pit limit for the open pit in 2D in the given graph which is a cross-section from 3D
from collections import defaultdict
class Graph:
   def __init__(self, graph):
       self.graph = graph  # residual graph
       self.org_graph = [i[:] for i in graph]
       self.ROW = len(graph)
       self.COL = len(graph[0])

   def BFS(self, s, t, parent):
       visited = [False] * (self.ROW)
       queue = []
       queue.append(s)
       visited[s] = True

       while queue:
           u = queue.pop(0)
```

```
        for ind, val in enumerate(self.graph[u]):
            if visited[ind] == False and val > 0:
                queue.append(ind)
                visited[ind] = True
                parent[ind] = u
    return True if visited[t] else False

  def minCut(self, source, sink):
    parent = [-1] * (self.ROW)
    max_flow = 0

        while self.BFS(source, sink, parent):
        path_flow = float("Inf")
        s = sink

        while (s != source):
            path_flow = min(path_flow, self.graph[parent[s]][s])
            s = parent[s]
        max_flow += path_flow
        v = sink
        while (v != source):
            u = parent[v]
            self.graph[u][v] -= path_flow
            self.graph[v][u] += path_flow
            v = parent[v]

    for i in range(self.ROW):
        for j in range(self.COL):
            if self.graph[i][j] == 0 and self.org_graph[i][j] > 0:
                print(str(i) + " - " + str(j))
graph = [[adjacency matrix]]
g = Graph(graph)
source = 0; sink = N
g.minCut(source, sink)

# This code is contributed by pathie Musenge for the mining domain
```

Figure 7: Proposed maximum flow algorithm of the Ford and Fulkerson algorithm

Following are steps to print all edges of the minimum cut: Run the Ford-Fulkerson algorithm and consider the final residual graph; Find the set of vertices that are reachable from the source in the residual graph; All edges which are from a reachable vertex to a non-reachable vertex are minimum cut edges. Print all such edges. Solution of the max-flow model for Mutoshi deposit using Python Program The maximum pit value can be realized by separating the trees S and T using the invalid parts created by the saturated edges linking the two trees. The saturated edges from the source node to the ore blocks are normally broken to avoid the support of the ore blocks to the overlying waste blocks, the other saturated edges from the waste blocks to the sink node are also broken to avoid the support of these waste blocks from the underlying ore blocks. This also makes sure that there are no outgoing arcs from the maximum closure. In this case, the ultimate pit contains the blocks in table 2 with a maximum close value of USD 3203 and his ultimate pit value as shown in Figure 8

0 – 85	72 – 183	117 – 183	135 – 183	151 – 183	167 – 183
0 – 86	75 – 183	118 – 183	136 – 183	152 – 183	169 – 183
0 – 87	89 – 183	119 – 183	137 – 183	155 – 183	170 – 183
0 – 88	92 – 183	120 – 183	138 – 183	156 – 183	171 – 183
0 – 90	99 – 183	121 – 183	141 – 183	157 – 183	172 – 183
0 – 86	100 – 183	122 – 183	142 – 183	158 – 183	173 – 183
0 – 87	101 – 183	127 – 183	143 – 183	159 – 183	174 – 183
0 – 88	102 – 183	128 – 183	144 – 183	160 – 183	175 – 183
0 – 90	103 – 183	129 – 183	145 – 183	161 – 183	176 – 183
0 – 104	106 – 183	130 – 183	146 – 183	162 – 183	177 – 183
0 – 105	113 – 183	131 – 183	147 – 183	163 – 183	178 – 183
0 – 107	114 – 183	132 – 183	148 – 183	164 – 183	179 – 183
0 – 123	115 – 183	133 – 183	149 – 183	165 – 183	180 – 183
61 – 183	116 – 183	134 – 183	150 – 183	166 – 183	181 – 183
					182 – 183

Figure 5. 1: Ultimate pit limit using Maxflow algorithm Ford and Fulkerson on section 3

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 11 Issue 02, February-2022

Ultimate pit limits for each 2-dimensional section (13 cross-sections)

To demonstrate the computation speed of the maximum flow algorithm, a series testing of 13 cross-sections through block models was used. The 45-degree slope angle is adopted for the block model. The number of arcs created for this slope setting and the economic value is listed in Table 2 Note that, the actual number of blocks and arcs used in the optimization is called active blocks and active arcs, in f, the active blocks are colored in red, blue, orange and yellow. (Active blocks represent the blocks that contain the ultimate pit limits in each section, and all the precedent blocks linked to the active blocks need to be removed to access the ore.

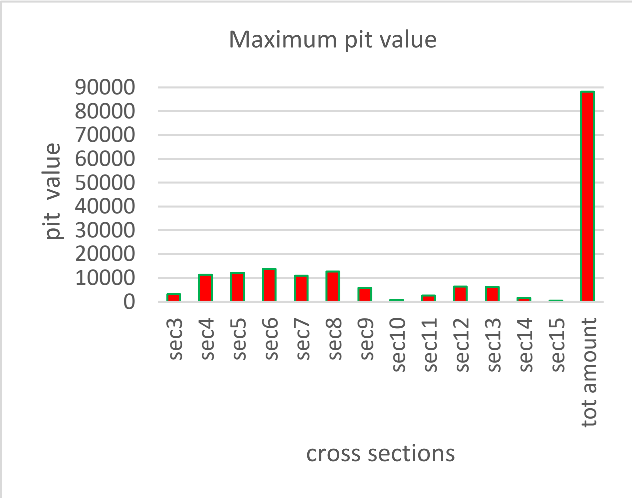

Figure 9: Maximum pit value for all 13 cross sections

FINAL ULTIMATE PIT LIMITS FOR THE DEPOSIT

The limit of the final pit is made up of 13 cross-sections which were mineralized, and of these sections, we have a set of blocks which represents the min-cut found during programming with the python application with their respective limits and the figure 10 is the Mutoshi ultimate pit limits, the pit representation of all the sections in 3 faces is and the construction of the pit shell was carried out in windows excel, In figure 10:

COMPARISON OF MAX-FLOW RESULTS WITH LERCHS AND GROSSMANN

Study case Ultimate pit limits using the Lerchs and Grossmann algorithm concept

The essence of the LG algorithm is to split the economic block model into parallel vertical sections or 2D space, then determine the contour of the pit on each section that yields the maximum profit. The general configuration of the pit contour on a cross-section (2D) consists of three sides: two walls inclined at a certain slope angle and the bottom level of the pit. Analytically, the LG seeks to maximize the objective function Z shown in Equation 6.1 to design an optimum pit: (musema, 2020)

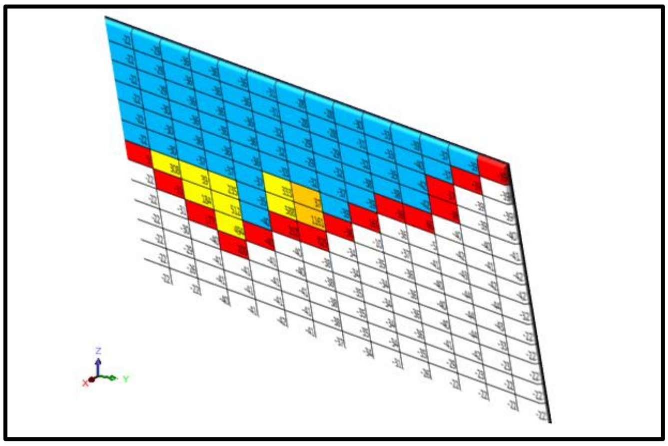

Figure 8: Ultimate pit limit using Maxflow algorithm Ford and Fulkerson on section 3

- the blocks in blue, yellow, and orange blocks represent the active blocks which are joined by the active
- the blocks in black represent the set of blocks that constitute the blocks that must not be mined and will remain in our mines
- the blocks in blue represent the set of all the wastes blocks that continue within the limit to be exploited and those in yellow color are the ore blocks also contained within the limit of the blocks to be mined
- all the blocks in red represent the ultimate limit and its blocks will also be extracted

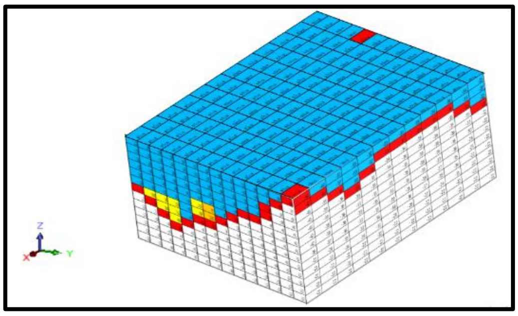

Figure 10: a representation of Mutoshi ultimate pit limit using maximum flow algorithm

$$Max\ Z = \sum_{i,j}^{n} BE\,V_{ij}$$

The first step is to compute the cumulative profits M_{ij} realized after the extraction of a single column of blocks having at its base the block x_{ij} . The calculation of M_{ij} within a column is independent of other columns as shown in Equation 3 where the index l identifies all blocks involved.

$$M_{ij} = \sum_{i\,i}^{n} BE\,V_{ij}$$

$$P_{ij} \stackrel{i\,i}{=} M_{ij} + \max\{\ c_{j+1,j-1}\}$$

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 11 Issue 02, February-2022

Table 1: Introduction of a dummy row on a cross-section block model of M_{ij} values

						Cross-section of Mij									
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	-22	-29	-36	-36	-36	-31	-28	-28	-30	-33	-39	-37	-39	-39	0
0	-44	-58	-72	-72	-72	-62	-56	-56	-61	-66	-79	-74	-78	-78	0
0	-66	-87	-108	-108	-108	-94	-85	-85	-93	-101	-121	-111	-117	-117	0
0	-88	-117	-144	-144	-144	-126	-114	-117	-131	-139	-163	-149	-156	-157	0
0	-110	-147	-180	-180	-180	-159	-146	-154	-170	-175	-97	-189	-196	-198	0
0	-132	-177	-217	-217	-217	174	-109	-193	-9	-211	-102	-231	-237	-240	0
0	-127	131	-178	18	-252	762	1052	-229	-19	-248	-143	-274	-279	-282	0
0	-149	100	6	530	-292	1054	1877	-263	-54	-284	-183	-318	-323	-314	0
0	-171	69	183	1024	-332	1014	1839	-297	-88	-320	-223	-362	-366	-336	0
0	-193	39	143	1224	-373	974	1801	-332	-122	-356	-263	-406	-399	-358	0
0	-215	10	102	1183	-414	933	1763	-367	-156	-392	-304	-448	-422	-380	0
0	-237	-16	61	1142	-455	892	1725	-402	-190	-427	-333	-471	-444	-402	0
0	-259	-38	21	1101	-497	851	1688	-436	-221	-453	-355	-493	-466	-424	0

The process continues until the last column generating overall cumulative values P_{ij} on each block. The optimum pit contour on the section is that which yields the maximum cumulative value and in the current case, the net value for the cross-section is USD 3175.

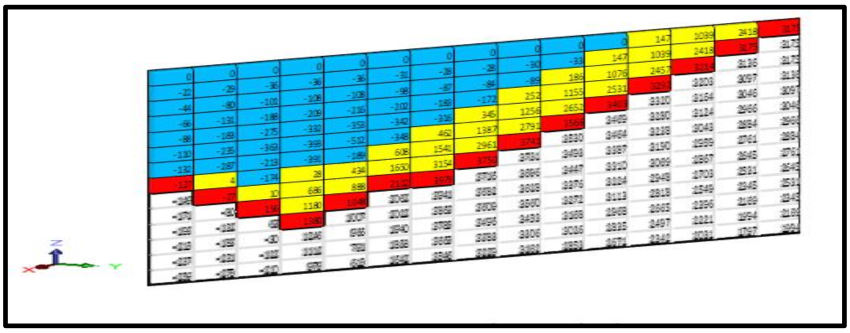

Figure 11: ultimate pit limit using LG for Mutoshi block model cross-section 3

COMPARISON OF OPTIMAL PIT VALUE

The ultimate pit limit for this deposit was calculated using the FFA and LG. The FFA maximum flow algorithm gave a maximum pit value of USD 881 780 000 in 25s with 1261 as some active blocks. The LG gave the maximum pit value of USD 880 211 but still with a long time of 45s with 1443 number of active blocks, table 6.3 gave the percentage square around 99,9 This, therefore, shows that the FFA maximum flow can be applied in the mining industry ultimate pit optimization since the results shown the on others sections the same values with LG model, which is already being applied. Table 5.13 gives a summary of the comparison of maximum flow algorithm Results for Ford and Fulkerson and Lechrs and Grossmann. The two final ultimate pits obtained from FFA and LG for Mutoshi deposit for this deposit are shown in Fgure 12.

Table 5: Introduction of a dummy row on a cross-section block model of M_{ij} values

Comparison		
Sections	Maxflow FF maximum pit value	LG maximum pit value
sec3	3203	3175
sec4	11288	11301
sec5	12218	12242.4
sec6	13828	13852
sec7	10938	10920.5
sec8	12681	12177.2
sec9	5873	5931.4
sec10	739	814.08
sec11	2607	2606.9
sec12	6361	6334.01
sec13	6270	6256.367
sec14	1696	1933.4
sec15	476	476.8
Total value	88178	88021.057
Active blocks	1261	1443
Total time /seconds	25	45
Difference between the total amount	156.947	

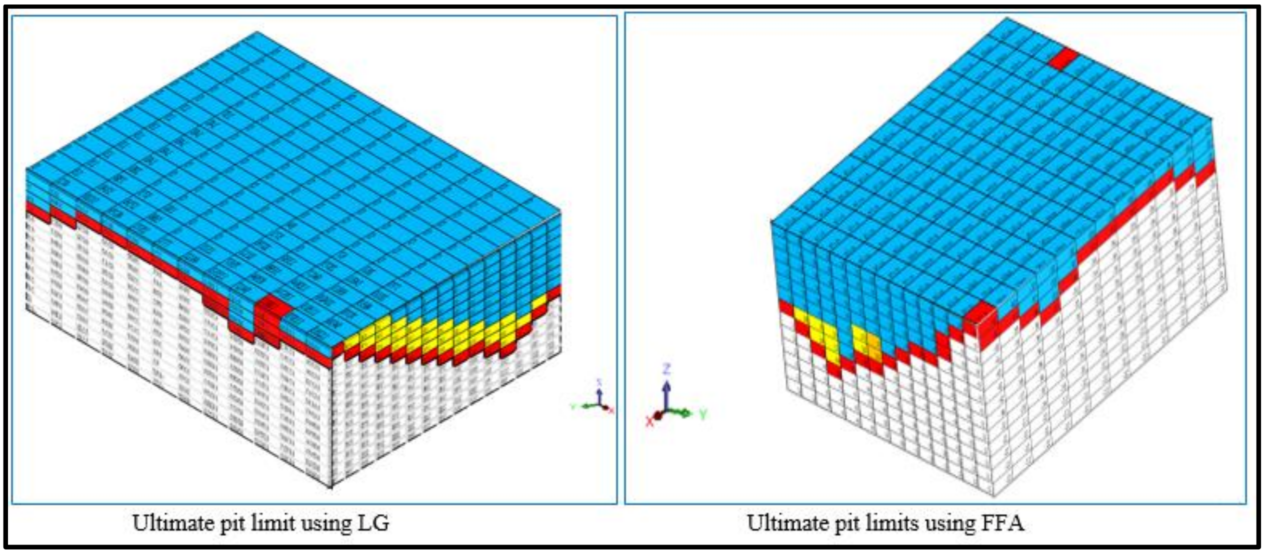

Figure 11: The finals ultimate pits obtained from FFA and LG for Mutoshi

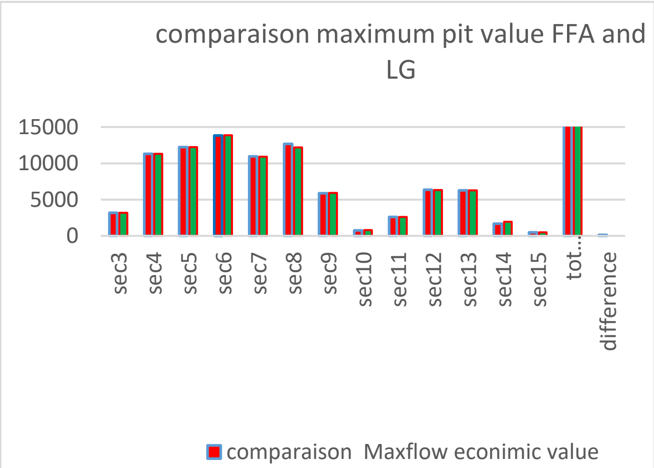

Figure13: comparison maximum pit value for FFA and LG

Table 6: Table regression statistics

SUMMARY OUTPUT	
Regression Statistics	
Multiple R	0.999537
R Square	0.999075
Adjusted R Square	0.99899
Standard Error	154.9037
Observations	13

COMPARISON OF MODEL PERFORMANCE

The coefficient of determination (R-squared) was used. R-squared shows how well the regression model fits the observed data, the correlation coefficient interprets the R values as 0 < R < 30% implies weak correlation, 30% < R < 70% implies moderate correlation and R > 70% implies strong correlation. (Amankwah, 2011) figure 13 compare results between FFA and LG using regression analysis

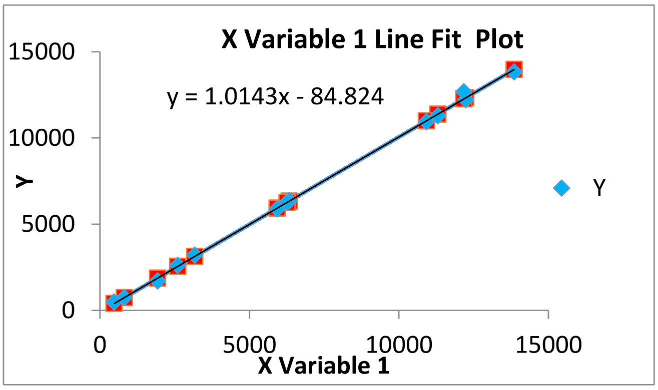

Figure 14: correlation between FFA model and LG model

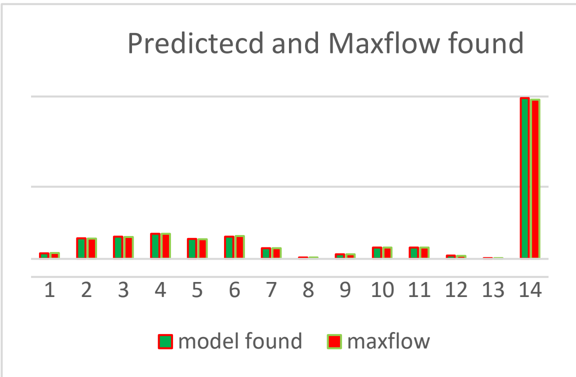

Figure 15: predicted model and FFA model

Application of the model to Maxflow algorithm Ford and Fulkerson, the model found y = 1.0143x - 84.824

Table 7: results FFA and regression results

data			
sections		Maxflow Ford Fulkerson Algorithm	Model found
1	Sec3	3203	3135.5785
2	Sec4	11288	11377.7803
3	Sec5	12218	12332.64232
4	Sec6	13828	13965.2596
5	Sec7	10938	10991.83915
6	Sec8	12681	12266.50996
7	Sec9	5873	5931.39502
8	Sec10	739	740.897344
9	Sec11	2607	2559.35467
10	Sec12	6361	6339.762343
11	Sec13	6270	6261.009048
12	Sec14	1696	1876.22362
13	Sec15	476	398.79424
	Total value	88178	89194.93412

CONCLUSIONS AND RECOMMENDATIONS

Ultimate pit limit optimization (UPLO), playing a major role in the mining industry and algorithms for solving UPLO, have been developed and improved by various researchers since the 1960s. This research project aimed at finding the ultimate pit limit for Mutoshi deposit, a Ford and Fulkerson algorithm model was developed and coded in Python programming language to be run on a block model through 13 cross-sections in two- dimensions. The application of the proposed Ford and Fulkerson algorithm in this research determined the best results and applicability in the mining domain compared to results obtained using Lerchs and Grossmann with the same data set. However,

The implementation of the Ford and Fulkerson algorithm to solve the ultimate pit limits problem in an open-pit mine has shown encouraging results and its applicability compared to Lerchs and Grossmann when applied to the case study. It was demonstrated that the Ford and Fulkerson Algorithm fits well in solving the ultimate pit limits problem as those two methods consider Lerchs and

Grossman as the base model already found and proved through several authors as the first mathematical method to solve the ultimate pit limits problem. From two methods we found a model for validation and compare the results from this one the applicability of the Ford and Fulkerson algorithm has been proved to be used in the mining domain.

To test and validate the proposed FFA model, this study used also LG which is a known method for solving the ultimate pit limits problem. The deposit has 3,304 blocks, with 60 in X, 60 in Y, and 20 in the Z direction as user blocks, the FFA used 2366 as blocks and LG used 2366 Blocks. When applied the FF model, the FFA provided an ultimate pit limit with 4424 arcs and 1261 active blocks, and the results in excel provide 1443 active blocks for LG.

ACKNOWLEDGEMENTS

This paper presents part of work conducted by the first author in partial fulfilment of the requirements for an master research suty degree at the school of mines, University of Zambia, Lusaka, Republic of Zambia

REFERENCES

[1] Abzalov, 2006. Localised uniform conditioning (LUC): A new approach for direct modeling of small blocks.Mathematical Geology. Volume II, pp. 27-45.
[2] Akisa David Mwangi, Z. J., 2020. Ultimate Pit Limit Optimization using Boykov-Kolmogorov Maximum. *Journal of Mining and Environment (JME)*, pp. 4-7.
[3] Amankwah, H., 2011. Mathematical Optimization Models and Methods for Open-Pit Mining. In: Linkoping: LiU-Tryck, Linkoping, pp. 23-30.
[4] AUGER, G., 2000. STUDY AND PROGRAMMING OF THE MAXIMUM FLOW ALGORITHM. In: Montreal: POLYTECHNIC SCHOOL OF MONTREAL, pp. 23-45.
[5] Ayisi, M. O., 2015. 3D Block Modeling and Reserve Estimation of a Garnet Deposit. pp. 13-22.
[6] Bai, Y., 2017. pseudoflow method for pit optimization. *technical report.*
[7] Chanda, E., 2018. advances in applied strategic mine planning. In: r. Dimitrakopoulos, ed. Montreal: s.n.
[8] Goldberg, R. E. & Tarjan, 2014. Efficient Maximum Flow Algorithms. *Magazine Archive*, Volume vol. 57.
[9] HEUREUX, 2011. MODEL OF OPTIMIZATION FOR MEDIUM PLANNING. pp. 23-30.
[10] HOCHBAUM, C., 1998. Performance analysis and best implementations of old and new algorithms for the open-pit mining problem. *to appear in naval research quartly*, p. 40.
[11] Hustrulid & Kuchta, M., 2016. Open pit mine planning and design.. Volume Volume 1-fundamentals.
[12] KABEZYA, 2017. Apllication of goetechnical model in the short term planning of open pit case of the North Mutoshi project.. pp. 4-41.
[13] KIM, Y., 1978. Ultimate pit limit design methodologies using conputer models - The state of the art. Mining Engineering. pp. 1454-1459.
[14] KÜBRA, A. F., 2019. A NEW APPROACH TO 3-DIMENSIONAL OPTIMIZATION OF ULTIMATE PIT GEOMETRY FOR OPEN PIT MINES WITH VARIABLE OVERALL SLOPE ANGLES. pp. 8-25.

[15] Kubra, F., 2019. *A NEW APPROACH TO 3-DIMENSIONAL OPTIMIZATION OF ULTIMATE.* le caire: MIDDLE EAST TECHNICAL UNIVERSITY.
[16] kwiri J, G., 2017. Mine planning and optimisation techniques used in surface mining. pp. 200-220.
[17] Meisam Saleki, M. A. & Kakaie, R., 2019. Mathematical relationship between ultimate pit limits generated by discounted and undiscounted block value maximization in open pit mining. *Journal of Sustainable Mining Elsevier,* pp. 94-96.
[18] Muke, P. M., 2019. Open-pit mine production scheduling using the genetic algorithm. *research gates,* pp. 15-24.
[19] musema, m., 2020. o p e n – p i t m i n e p r o d u c t i o n s c h e d u l i n g u s i n g t h e g e n e t i c a l g o r i t h m. pp. 20-26.
[20] Osanloo, M. G. J. K., 2008. Long-term open-pit mine production planning : a review of models and algorithms. *International Journal of Mining, Reclamation and Environment,* pp. 3-25.
[21] Sebastian Avalos, J. M. O., 2020. A guide for pit optimization with Pseudoflow in python 1. *Annual Report,* pp. 186-187.
[22] Shiv Prakash, H. A., 2015. Simulation and Optimization in Open Pit Mining. *Applications of Cumputers and Operations Research in Mineral Industry, Alsaka,* pp. 534-540.
[23] ww.developers.google.com/optimization/flow/maxflow, 2020. *Netwok flows.* [Online] Available at: ww.developers.google.com/optimization/flow/maxflow