IOSR Journal of Engineering (IOSRJEN) **www.iosrjen.org**
ISSN (e): 2250-3021, ISSN (p): 2278-8719
Vol. 08, Issue 10 (October. 2018), ||V (III) || PP 25-31

Task Scheduling in Cloud Computing using Credit Based Cluster Travelling Salesman problem

Animoni Nagaraju[1] Dr.Y. Ramadevi[2]
[1]*(Research Scholar, Department of Computer Science and Engineering, University College of Engineering, Osmania University, Hyderabad, Telangana, Email id: animoni_nagaraju@yahoo.co.in)*
[2]*(Professor ,Department of Computer Science and Engineering , Chaitanya Bharathi Institute of Technology, Hyderabad, Telangana,*
Corresponding Author: Animoni Nagaraju

Abstract: - Task scheduling system problem is a nucleus and demanding issue in Cloud Computing. The Cloud computing resources proficiently and gain the maximum profits with scheduling system is ultimate objective in the one of the Cloud computing service providers. In this paper, we have used credit based cluster travelling sales man problem to evaluate the entire group of task in the task queue and find the minimal completion time of all tasks. Here cost/time matrix has been generated as the fair tendency of a task to be assigned in resources. Her we apply Cloud computing is a latest new computing method of delivering technology to the consumer by using Internet servers for processing and data storage, while the client system uses the data.

Keywords: - TSP, CTSP, Cloud computing, Lexi search Algorithm, Trip schedule ,

---

Date of Submission: 10-10-2018 Date of acceptance: 26-10-2018

I. INTRODUCTION

Many technology experts believe that cloud computing is assured to change the way we access technology and that it may be making changes as the commercialization of the Internet over a decade ago. Cloud computing enables innovation. It makes ease, the need of innovators to find resources to develop, test, and make their innovations available to the user community. Innovators are free to focus on the innovation rather than the logistics of finding and managing resources that enable the innovation. So that efficient task scheduling problems and resource management are relate to the efficiency of the whole cloud computing facilities. These tasks are parallel processed on the nodes of the cluster by the policy which strives to keep the work as close to the data as possible.

The scheduling algorithms in distributed systems usually have the goals of spreading the load on processors and maximizing their utilization while minimizing the total task execution time. Several heuristic algorithms has been introduced in task scheduling. One of the example of NP-hard problem is travelling salesman problem. The travelling sales man problem (TSP): TSP is the one of the oldest and widely studied optimization problem in the field of design of algorithm, Lawler (1985). For the standard TSP we assume that a single sales man based at home city, must visit a number of cities or customers. The problem is to find the salesman minimum cost cycle who leaves the home city, and visit each customer city once and return to the home city. Real life routing or sequencing problems often requires additional decisions or considerations, and also, The Travelling sales man problem is a particular case of assignment problem, which considers a network with '*n*' nodes (or cities) and a cost matrix of order n associated with each node pair (i, j) is given. The problem is to find a least cost tour for a salesman who must visit each of the nodes exactly once except that the salesman should reach starting city. This problem is widely used in many contexts with different algorithms and obtained the optimal cost tour for the Travelling Salesman Problem (TSP), in this problem A set of n nodes (cities), with distance between every ordered node pair, is given. The problem is to find a least distance route for a salesman who must visit each of these cities starting and ending his journey in the same city.

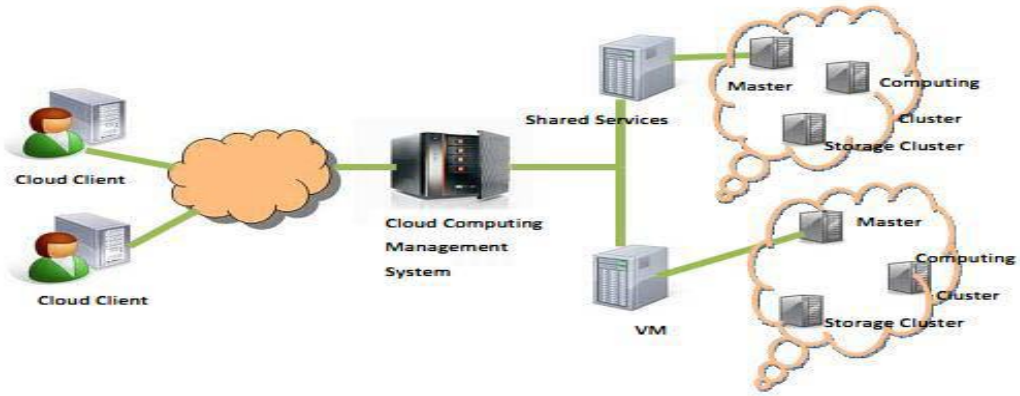
To solve the combinatorial problems different techniques are available. However, no general approach, which is suitable for all discrete programming problems, seems to be available and the methods of finding optimal solutions mainly by search methods (Pandit 1963). For many of the combinatorial programming problems the solution space is finite and hence, it is theoretically to enumerate all the solutions.

Clustered travelling salesman problem (CTSP) deals with a branching scheme, where a cluster is defined as a node set with the characteristic that there exists an optimal solution in which the nodes in the node set are visited by cluster wise. This problem is implemented with lexi-search approach and obtained the exact solution to the CTSP. This CTSP is the application and which utilize Cloud computing resources proficiently and gain the maximum profits with scheduling system is one of the Cloud computing services.

In this Paper, A large-scale cloud computing consider as travelling salesman problem (TSP) can be converted into a cluster distribution service clouds , like cluster travelling sales man problem (CTSP) by clustering so that it can be solved by the lexi search algorithm with optimal criterion, in which we obtain best route for distributing the salesman letters or given task schedule. This CTSP problem is constructed from some randomly generated standard sample sizes. Which gives best tour with minimum time and getting profit cost on the whole cloud computing without risk.

1.1 Cloud Architecture

Cloud computing refers to the use of networked infrastructure software and the capacity to provide resources to users in an on-demand environment. Cloud computing is fully enabled by virtualization technology (hypervisors) and virtual appliances. A virtual appliance is an application that is bundled with all the components that it needs to run, along with a streamlined operating system. In a cloud computing environment, a virtual appliance can be instantly provisioned and decommissioned as needed, without complex configuration of the operating environment. This flexibility is the key advantage to cloud computing, and what distinguishes it from other forms of grid or utility computing and software as a service (SaaS). Cloud computing system scales applications by maximizing and using computing resources more efficiently. One must optimize locking duration, statelessness, sharing pooled resources such as task threads and network connections bus, cache reference data and partition large databases for scaling services to a large number of users, below diagram represents the proposed architecture, present paper concentrates on scheduler manager.



Fig(1): Diagram of proposed architecture

II. PROPOSED METHODOLOGY

Travelling sales man problem is a particular case of assignment problem to find the best Task Scheduling with minimal cost tour, the cluster travelling sales man problem is additional constraint of travelling sales man problem in which we obtain the best tour, implementing Lexi Search approach for the standard CTSP. Which follow initially Generate a particular size of TSP problem(large Cloud), Identify the set of nodes in the usual TSP which is partitioned into k clusters (k-clouds), next each cluster group is considered as a TSP , for which costs are arranged in the increasing order is known alphabet table .

The lexi-search algorithm, that is, systematically we 'generate' incomplete words, from the search table using the new labels, for each cluster with the inter relationship in a clock wise direction , from the first cluster we go to the next cluster without reaching the end node, this is continued, till we reach the home city, and records the original index names as well accumulated costs included in the word so far ,and also bounds for the remaining part of the incomplete word i.e. the bound for all feasible words in the lexical block, for which the current incomplete block is a leader. If this bound is greater than a trial solution value on hand, the leader is discarded and the next incomplete word, of the same length or its next super block leader as the case may be is chosen on the current incomplete word, these steps are recorded in the search table presented below.

Numerical Illustration
The Lexi search procedure is explained through the solution of a Cluster travelling salesman problem of size 8 x 8. For illustrating the concepts and definitions involved in the problem we have considered N={1,2 ,….,8}, |N|=n=8, |N1|=n1=4, |N2|=n2=4 and |M|=m=8. The cost matrix *C* (*i, j*) of CTSP is given as follows.

Let [Cij] be the given cost matrix where Cij be the cost of city j from city i; where i , j =1,2,….8.

Table-1

∞	6	3	8	5	10	7	8
5	∞	3	7	6	15	8	9
3	8	∞	4	10	7	9	8
5	2	4	∞	7	8	16	5
12	7	8	10	∞	2	3	7
5	9	10	4	3	∞	10	8
10	12	8	10	5	4	∞	3
8	7	10	15	4	2	3	∞

In the above table the value infinity '∞' indicates the non-connectivity of the cities directly. In this numerical example we are given eight cities and the salesman wants to visiting 8 cities only with the condition that the trip schedule starts from headquarter city, in which each of the nodes visit exactly once except that the salesman reach to home city by covering all cluster city's. For the above 8*8 CTSP we partition the cost matrix into 4 sub matrices, and all the sub matrix distances are arranged in the increasing order from i,j =1,2…..8. Now the resulting table is called the Alphabet Table

From the Alphabet Table we select the ordering of cities, if the value of the bound is greater than the trial solution there will be no better solution in the subsequent blocks of the super block. An indicator matrix *X = [X (i, j) / X (i, j)= 0 or 1]* in which *X (i, j)=1* indicates that the j^{th} city is Visited from i^{th} city, otherwise *X (i, j) =0*. *X* is called a solution. The indicator matrix *X* with 0 or 1. In the next section we describe this problem by considering an illustration and is solved by lexi-search Algorithm

Tour represented by the above pattern is (4,2) (2,3)(3,1) (1,5)(5,7) (7,8)(8,6)(6,4) where the sales man start the city 4 is home city and visits first cluster and last cluster and reaches to home city. In this tour optimal solution which exact solution is 4→2→3→1→ 5→7→8→6→4 ,with the value 25 by optimal criterion for defined problem table 1 and procedure explained in the search table .

III. CONCEPTS AND EQUATIONS

3.1 patterns
An indicator two dimensional arrays *X* which is associated with the selection of items from different clusters is called a "**pattern**". A pattern is said to be feasible if *X* has a feasible solution. The pattern represented in X=0 or 1, non selection and selection. Now thevalue of the pattern *X* is defined as follows $V(X)=\sum_{i=1}^{n}\sum_{j=1}^{n}C_{ij}X_{ij}$

The value *V(X)* gives the total cost of the CTSP of the solution represented by *X*. Thus the value of the feasible pattern gives the total cost. In the algorithm, which is developed in the sequel, a search is made for a feasible pattern with the least value. Each pattern of the solution *X* is represented by the set of ordered pairs.

3.2 Alphabet – Table
There is **M= n x n** ordered pairs in the two dimensional array *X*. These are arranged in ascending order of their corresponding costs with indexed from 1 to M individual clusters. Let **SN**= {1, 2… M} be the set of M indices. **D , CD** is the corresponding costs and Cumulative sums of the elements of D. The arrays **SN, D,CD, R, and C** are indicate the serial number, cost of the C*TSP*, Cumulative cost, row and column indices respectively. For the numerical example given in Table – 3, if **r ϵ SN** then [R(r), C(r)] be the cost in their position, D(r)= d[R(r), C(r)] and CD(r)= $\sum D(i), i=1,2,....r.$

Table-2: Alphabet table

	First Cluster roup				SecondCluster Group				Third Cluster Group				Fourth Cluster Group			
SN	D	CD	(row,colums)	SN	D	CD	(row,colum)	SN	D	CD	(row,colum)	SN	D	CD	(row,colum)	
1	2	2	(4, 2)	13	5	63	(1, 5)	29	2	198	(5, 6)	41	4	254	(6,4)	
2	3	5	(1, 3)	14	5	68	(4, 8)	30	2	200	(8, 6)	42	5	259	(6, 1)	
3	3	8	(2, 3)	15	6	74	(2, 5)	31	3	203	(5, 7)	43	7	266	(5, 2)	
4	3	11	(3, 1)	16	7	81	(1 ,7)	32	3	206	(6, 5)	44	7	273	(8, 2)	
5	4	15	(3, 4)	17	7	88	(3, 6)	33	3	209	(7, 8)	45	8	281	(5, 3)	
6	4	19	(4, 3)	18	7	95	(4, 5)	34	3	212	(8,7)	46	8	289	(7, 3)	
7	5	24	(2, 1)	19	8	103	(1, 8)	35	4	216	(7, 6)	47	8	297	(8, 1)	
8	5	29	(4, 1)	20	8	111	(2, 7)	36	4	220	(8, 5)	48	9	306	(6, 2)	
9	6	35	(1,2)	21	8	119	(3, 8)	37	5	225	(7, 5)	49	10	316	(5, 4)	
10	7	42	(2, 4)	22	8	127	(4, 6)	38	7	232	(5, 8)	50	10	326	(6, 3)	
11	8	50	(1, 4)	23	9	136	(2, 8)	39	8	240	(6, 8)	51	10	336	(7, 1)	
12	8	58	(3, 2)	24	9	145	(3, 7)	40	10	250	(6, 7)	52	10	346	(7, 4)	
				25	10	155	(1, 6)					53	10	356	(8,3)	
				26	10	165	(3, 5)					54	12	368	(5, 1)	
				27	15	180	(2, 6)					55	12	380	(7, 2)	
				28	16	196	(4, 7)					56	15	395	(6, 4)	

From this Alphabet table we solve the CTSP. The search table give optimal solution using lexi search approach for the 8*8 CTSP.

3.3 Feasible criterion

let $L_k = \{\alpha 1, \alpha 2, ...\alpha k\}, \alpha i \in SN$ be an ordered sequence of k indices from SN. The pattern represented by the ordered pairs whose indices are given by L_k is independent of the order of α_i in the sequence, the indices are arranged in the increasing order such that $\alpha_i \le \alpha_{i+1,} i = 1, 2, ...n-1$. the ordered sequence L_k is called a 'sensible word' if $\alpha_i \le \alpha_{i+1,} i = 1, 2, ...k-1$ other wise we call it as non sensible word . A word L_k said to be partial feasible word if the pattern X represented by L_k is feasible. If k=n the word represents a solution. A leader L_k is said to be feasible if the block of words defined by it contains at least one feasible word. Now the value of the word L_k is defined as follows $V(L_k)=V(L_{k-1})+D(\alpha_k)$ with $V(L_0)=0$

A Lower bound $LB(L_k)$ for the values of the block of words represented by L_k can be defined initial lower bound=22. Feasibility criterion of a partial word its lower bound is 38.

We used to Lexi search algorithm to check the feasibility of a partial word. We start the algorithm with a large value say '∞' as a trial value VT. If the value of a feasible word is known, we can as well start with that value as VT. During the search the value of VT is improved. At the end of the search the current value of VT gives the optimal feasible word. We start the partial word $L_1 = (a_1)$. A partial word L_k is constructed as $L_k = L_{k-1} * (a_k)$, where * indicates concatenation i.e. chain formation. We will calculate the values of V (L_k) and LB (L_k) simultaneously. Then two cases arise one for branching and the other for continuing the search.

1. **LB (L_k) < VT**. Then we check whether L_k is feasible or not. If it is feasible we proceed to consider a partial word of order (k+1), which represents a sub block of the block of words represented by L_k. If L_k is not feasible then consider the next partial word of order by taking another letter which succeeds a_k in the k^{th} position. If all the words of order 'k' are exhausted then we consider the next partial word of order (k–1).

2. **LB (L_k) ≥ VT.** In this case we reject the partial word L_k. We reject the block of word with L_k as leader as not having optimum feasible solution and also reject all partial words of order 'k' that succeeds L_k.

Now applying Lexi–Search algorithm by using above procedure obtained optimal tour for give problem to the alphabet table is 4→2→3→1→ 5→7→8→6→4 with optimal solution 25. By applying above process we get best optimal solution for minimizing the sales man costs from visiting neighbourhood cities.

3.4 Search table

Lexi search approach for CTSP detailed step by step procedure explained below, with above feasible criterion implimentation , The column named 1,2,3,4,5,6,7,8 are approach words by using alphabit table (table-2) respectively, the corresponding value V(I) and lower bound LB(I) are indicated Feasible value and Fixation of Bound on the step by step procedure. The column R and C are the row and column indices. the remarks are either acceptability (A) or cycle fail (CF) of the partial words, from the search table on feasible criterion we obtain the feasible solutions are approaches to best optimal solution , from this illustration the best tour is 4→2→3→1→ 5→7→8→6→4 with optimal solution 25 we can see in the search table(table-3).

IV. FIGURES AND TABLES

SEARCH TABLE													
SN	1	2	3	4	5	6	7	8	V	LB(I)	R	C	Remark
1	1								2	22	4	2	A
2		3							5	22	1	3	A
3			5						9	22	3	4	A
4				15					15	26	2	5	A
5					29				17	26	5	6	A
6						32			19	26	6	5	CF
7						33			20	27	7	8	A
8							34		22	26	8	7	CF
9							36		23	27	8	5	CF
10							40		30	34	6	7	A
11								47	38	38	8	1	v.trial =38
12						34			20	28	8	7	A
13							37		25	29	7	5	CF
14							39		28	29	6	8	A
15								51	38	38	7	1	A
16						36			21	24	8	5	A
17							39		23	27	6	8	CF
18							40		31	35	6	7	CF
19						37			22	33	7	5	A
20							40		32	36	6	7	CF
21						39			25	EG	6	8	EG
22					30				17	27	8	6	A
23						31			20	27	5	7	A
24							33		23	27	7	8	A
25								42	28	28<38	6	1	Bound=28
26						32			20	27	6	5	CF
27						33			20	27	7	8	A
28							40		30	28	6	7	BF
29	1	2	5	15	30	31			22	28	5	7	CF
30	2+(22 B)	5+(19B)	9+(16B)	15+(11B)	17+(10B)	22(CF)							
31						38							BF
32					38	33	40		31	28	6	7	BF
33					32				18	28	6	7	CF
34					33				18	29	7	8	BF
35				20	29	33			29	28	7	8	BF
36				23					29	28	2	8	BF
37	1	2	7	17	31				30	28	5	7	BF
38	2+(22 B)	5+(19B)	10+(16B)	17+(11B)	20+(10BF)								
39				21					29	28	3	8	BF
40	1	2	10						12	28	2	4	CF
41		3	4	13	29	33	34		21	28	8	7	CF
42		5+(19B)	8+(16B)	13+(11B)	15+(9B)	18+(7B)	(CF)						
43							40		32	28	6	7	BF
44						34	39		30	28	6	8	BF
45						39			37	28	6	8	BF
46									25	25<28	6	4	FS =25
47					30	31	33	41					
48						33	38		29	25	5	8	BF
49						38			34	25	5	8	BF
50					31				26	25	5	7	BF
51				16					26	25	1	7	BF
52	1	3	5						19	25	3	4	CF
53			11						29	25	1	4	BF
54		4	11						29	25	1	4	BF
55		5							27	25	3	4	BF
56	2								27	25	1	3	BF
END													

Table3

4.1 Computational Experience

In this section, 50 CTSP of sizes 8*8, 10*10, 12*12 & 14*14 are randomly generated and solve by algorithm. The optimum value, sequence and the time for the optimum are listed in the tables 5(a),5(b),5(c)&5(d).

Table -5(a) CLUSTERED TRAVELLING SALES MAN PROBLEM OF SIZE : 8X8

S.NO	OPTIMAL	OPTIMAL SEQUENCE	TIME(MILL.SEC)
1	22	1→3-->4-->2-->7-->6-->5-->8-->1	0
2	25	4-->2-->7-->5-->6-->8-->1-->3-->4	0
3	13	2-->1-->6-->8-->7-->5-->4-->3-->2	0

Table-5(b) CLUSTERED TRAVELLING SALES MAN PROBLEM OF SIZE :10X10

S.NO	OPTIMAL	OPTIMAL SEQUENCE	TIME(MILL SEC)
1	25	1→3→6→8→9→7→10→2→5→4→1	0
2	16	4→3→7→9→6→10→8→5→2→1→4	0
3	27	2→3→4→1→5→6→7→10→9→8→2	0

Table -5(c) CLUSTERED TRAVELLING SALES MAN PROBLEM OF SIZE : 12X12

S.NO	OPTIMAL	OPTIMAL SEQUENCE	TIME(SEC)
1	24	1→5→9→11→10→12→7→8→2→6→4→3→1	2
2	28	2→1→7→11→8→9→10→12→3→6→4→5→2	2
3	27	1→2→4→3→8→9→12→10→7→11→6→5→1	2

Table -5(d) CLUSTERED TRAVELLING SALES MAN PROBLEM OF SIZE : 14X14

S.NO	OPTIMAL	OPTIMAL SEQUENCE	TIME(MINUTES)
1	25	1→5→14→10→12→9→11→8→13→2→3→6→4→7→1	3
2	28	7→5→4→9→10→8→12→14→13→11→6→1→3→2→7	3
3	30	2→4→6→5→1→3→10→12→9→8→14→11→13→7→2	3

V. CONCLUSION

With the advancement of Cloud technologies rapidly, there is a new need for tools to study and analyze the benefits of the technology and how best to apply the technology to large-scaled applications. The proposed method considers the scheduling problem as the cluster travelling salesman problem like branch and bound where the cost matrix gives the cost of a task to be assigned into a resource.

From the present motivations of CTSP by lexi-search approach, feasible criterion which gives an exact solution to the problem evaluate the entire group of task in the task queue and find the minimal completion time of all task with profit, is tried for quite a number of problems with various sizes 8,10,12,14&18. The computational observation is that size of cloud increases, the time for optimality increases as verified on computer 512 MB ram. Therefore, The CTSP gives exact optimum tour and optimal solution.

REFERENCES

[1]. Shalini Arora and Puri , M.C. “A variant of time minimizing assignment problem” European Journal of Operational Research 110, 314-325 (1998).

[2]. Dong, J. Luo, L. Gao, and L. Ge, "A Grid Task Scheduling Algorithm Based on QoS Priority Grouping,"In the Proceedings of the Fifth International Conference on Grid and Cooperative Computing (GCC’06), IEEE, (2006).

[3]. Yair Amir, Baruch Awerbuch, Amnon Barak, R. Sean Borgstrom, Arie Keren, "An Opportunity Cost Approach for Job Assignment in a Scalable Computing Cluster", IEEE Transactions on Parallel and Distributed Systems , Vol. 11, No.7, July (2000).

[4]. Agarwal, A., Kumar, P.: Economical Duplication Based Task Scheduling for Heterogeneous and Homogeneous Computing Systems. In: Proceedings of the Advance Computing Conference, 2009 (IACC' 09), pp. 87-93, IEEE Computer Society (2009).
[5]. Remzi H. Arpaci, Andrea C. Dusseau, Amin M. Vahdat, et al , "The Interaction of Parallel and Sequential Workloads on Network of Machines", Proc. of ACM SIGMETRICS/Performance Joint Internationa Conference on Measurement and Modeling of Computer Systems , pp. 267-278, May,(1995).
[6]. Liu, H., Abraham, A., Hassanien, A.E., "Scheduling jobs on computational grids using a fuzzy particle swarm optimization Algorithm", Future Generation Computer Systems (2009), doi:10,1016/j.future.2009.05.022.
[7]. Braun, T. D. Siegel, H. J, Beck, N. , L. Boloni, M. Maheswar an, Reuther, A. I., Robertson, J.P, Theys, M. D , Yao, B. Hensgen, D., and Freund, R. F. "A comparison of eleven static heuristics for mapping a class of independent tasks onto heterogeneous distributed computing systems," Journal of Parallel and Distributed Computing, vol. 61, issue 6, pp. 810- 837, Jun.(2001).
[8]. Sandeep Tayal / (IJAEST) International Journal of Advanced Engineering Sciences And Technologies Vol No. 5, IssueNo. 2, 111 – 115.
[9]. Linguo Gong, Xian-He Sun, and Edward F. Waston, "Performance Modeling and Prediction of Non-Dedicated Network Computing", IEEE Trans. on Computer , Vol. 51, No 9, September,(2002).
[10]. Downey, A.B., "Using pathchar to estimate Internet link characteristics", Proc. SIGCOMM 1999, pp. 241-250, Cambridge, MA, Sept. (1999).
[11]. T.D.Braun, H.J.Siegel, N.Beck, D.A.Hensgen, R.F.Freund, A comparison of eleven static heuristics for mapping a class of independent tasks on heterogeneous distributed systems,Journal of Parrallel and Distributed Computing,pp.810- 837(2001)
[12]. E.Ullah Munir, J. Li, and Sh. Shi, QoS Sufferage Heuristic for Independent Task Scheduling in Grid.Information Technology Journal, 6 (8): 1166-1170 (2007)
[13]. Balas, E., and.Christofides, N "A restricted Lagrangian Approach to The Travelling Salesman Problem". Mathematical Programming, 21, 19-46 (1981)
[14]. Bellman, R.E.: "Dynamic programming treatment of the Travelling Salesman Problem". Journal of ACM, 9, 961-63, (1962).
[15]. Bellmore, M. And Mellone, J.C. : "Pathology of Traveling Salesman Sub tour Elimination Algorithms". Operations Research, 19, 278-307,(1971).
[16]. Bently, J.L.: "Fast Algorithms for Geometric Travelling Salesman Problems". ORSA Journal of Computer, 4, 387-411.(1992)
[17]. Bhatia, S.K., and A.N. Rocha "Traveling Salesman ". Computer Society of India Communications, 3-10, (1987).
[18]. Bianco, Mingozzi, S, Ricciardelli, S., and Spadoni, M.: "Exact and Heuristic Procedures for the Travelling Salesman Problem with Precedence Contraints, Based on Dynamic Programming". Information Systems & Operational Research, 32, 19-32,(1994)
[19]. Chisman, J.A.: "The Clustered Traveling Salesman Problem". Computers Op Res, 2, 115-119. (1975)
[20]. Lokin, F.C.J.: "Procedures for Travelling Salesman Problems with Additional Constraints". Eur Opl Res, 3, 135-141. (1978).
[21]. S.N.N. Pandit, Subrahmanyam, Y.V. "Enumeration of all optimal job sequence", Opsearch (l-2) ,35-39, (1975)
[22]. Arabi E. keshk, Ashraf B. El-Sisi, Medhat A. Tawfeek, "Cloud Task Scheduling for Load Balancing based on Intelligent Strategy",I.J. Intelligent Systems and Applications, 2014, 05, 25-36,. DOI: 10.5815/ijisa.2014.05.02.
[23]. Longhui Wang ,Yong Wang and Yudong Xie , "Implementation of a Parallel Algorithm Based on a Spark Cloud Computing Platform", Algorithms 2015, 8, 407-414; doi:10.3390/a8030407

IOSR Journal of Engineering (IOSRJEN) is UGC approved Journal with Sl. No. 3240, Journal no. 48995.

Animoni Nagaraju. " Task Scheduling in Cloud Computing using Credit Based Cluster Travelling Salesman problem." IOSR Journal of Engineering (IOSRJEN), vol. 08, no. 10, 2018, pp. 25-31.