March - April 2020
ISSN: 0193-4120 Page No. 10604 - 10616

An Application of C-language in Solving Minimum Spanning Connectivity of Clustered Cities to the Headquarter City

Sangeetham Prasad[1], T.Nageswara Rao[2], G. Balaji Prakash[2], V.B.V.N. Prasad[2], B.Mahaboob[2], M.Sundara Murthy[3]

[1]*S.V.University, Tirupathi.*

[2]*Department of Mathematics, KoneruLakshmiah Education Foundation, Vaddeswaram, Guntur Dist., A.P. INDIA.*

[3]*Retd. Professor, S.V.University, Tirupathi.*

Article Info
Volume 83
Page Number: 10604 - 10616
Publication Issue:
March - April 2020

Article History
Article Received: *24 July 2019*
Revised: *12 September 2019*
Accepted: *15 February 2020*
Publication: *13 April 2020*

Abstract: The primary objective of this article is to present a variant minimum spanning problem called "Minimum Spanning Connectivity of Clustered Cities to the Head Quarter City "which is a generalization of problem of Minimum Spanning Tree (MST)and to develop a Lexi-Search Approach(LSA)built on Pattern Recognition Technique(PRT)in order to solve the problem.Lexi Search Algorithms are shown to be dynamic in a large number of combinatorial situations. Here an innovative algorithm called LSA established on PRT has been proposed in order to acquire the optimal solution and the discussions made here are demonstrated by means of numerical examples. Using C- Language the proposed algorithm is designed and it can be observed that this algothim has the ability to execute enormous problems.

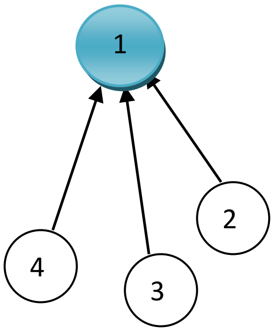
Keywords: *Lexi-Search Approach (LSA), Pattern Recognition Technique (PRT), Distance Matrix, Feasible and Infeasible Solutions, Search Table.*

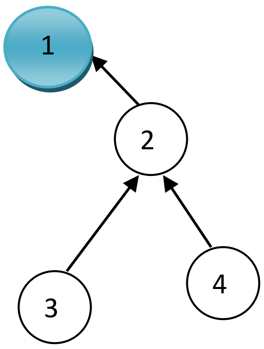
I. INTRODUCTION

One major theme in clustering research over the past two decades has been the automatic classification of quantitatively described objects, without any constraint as to which pairs of such objects might ultimately find themselves in the same class. A second major trend in clustering work has been where there is such an inherent or an imposed representational constraint. This second area of clustering arises naturally in the analysis of point patterns, and lately it has become an increasing interest in the analysis of data in the Geo-sciences. Even though the objectives of contiguity-constrained clustering algorithms may differ in pattern recognition, image processing, urban and regional studies, psychometrics, and so on, underlying principles are often shared, and valuable lessons may be learnt from other disciplines for the design of new algorithms.

Suppose a set N = (1, 2, 3, 4) of cities whose distances (N×N) are specified. The three cities have to connect to the Headquarter city {1}. When each city is individually connected, the total connectivity cost/distance is d (2, 1) + d (3, 1) + d (4, 1) as shown in fig-1. When it is connected as shown in fig-2 the connectivity is sum of d (2, 1) , d (3, 2) and d (4, 2).

March - April 2020
ISSN: 0193-4120 Page No. 10604 - 10616





II. PROBLEM DESCRIPTION

Suppose there is a set of some cities N. The distance between any two cities of them is also known. Among them city '1' consider a head quarter. A few cities of them are considered as a set of sub head quarter cities. The remaining cities of them are converted in to different groups of cities. The number of groups must be less than number of cities in set of sub head quarter cities. The cities in any two groups are distinct and cities in each group with set of sub head quarter cities are also distinct. The aim of the problem is to connect groups of cities to head quarter city through a sub head quarter cities with minimum total distance/cost. Here, the restriction of the problem is cities of any group are connected to any one sub head quarter city either directly or indirectly.

Let there be a set of 'n' cities defined as N = {1, 2, 3,………,n}. Here city '1' can be considered as head quarter. The distance between any two cities is denoted as D (i, j) in matrix form, where i, j Є N. Let SH be a set of 'q' cities as sub head quarter cities denoted asSH = $\{\alpha_1, \alpha_2, \alpha_3, \dots \dots, \alpha_q\}$, where SHCN. The remaining n-(q+1) cities formed as G_1, G_2, …….,G_p of 'p' groups of distinct cities, where n-(q+1) = $|G_1|+|G_2|+\dots+|G_P|$. Here $G_i \cap G_j = ø$, where i is not equal to j and i, j=1, 2, 3, …., p. The aim of the problem is to connect groups of cities to head quarter city through some sub head quarter cities with minimum total distance/cost. Here the restriction of the problem is cities of any group are connected to any one sub head quarter city only either directly or indirectly.

III. MATHEMATICAL FORMULATION

$|N| = n, |SH| = q, \qquad 1 \in N$

$|G_1| = g_1, |G_2| = g_2, \quad |G_p| = g_p, G_i \subset N, \ \sum_{i=1}^{p} g_i = g, SH \subset N$

(SH) ∩ (UG_i) = ø and $G_i \cap G_j = \emptyset, i \neq j.$

$i, j = 1 \text{ to } p$

Min Z= $\sum_{i \in N} \sum_{j \in N} D(i,j)\, X(i,j)$………..(1)

$\sum_{j \in SH} X(j,1) = p$ ……….. (2)

$X(i,j) = 0 \quad if\ i \in G_s, j \in G_t \quad where\ s \neq t\ ;$

s, t=1, 2,…, p ……(3)

$\sum_{i \in N} \sum_{j \in N} X(i,j) = (g+p) - 1$ ………….(4)

$\sum_{i \in G_s, j \in G_s \cup \{\alpha\}} X(i,j) = g_s, where\ \alpha \in SH$

& $s = 1, 2, 3, \dots \dots \dots \dots, p$……………..……..(5)

$X(i,j) =$ 1or 0 ………...……....……….(6)

Equation (1) represents the objective of the problem, i.e., minimize the distance/cost.

Equation (2) denotes number of groups formed from the cities of set SH (q>p).

Equation (3) indicates there is no connectivity between any two cities of two different groups.

Equation (4) states total number of connectivity must be equal to sum of total cities in all groups and number of groups.

Equation (5) shows number of connectivity formed from a group to a sub head quarter city.

March - April 2020
ISSN: 0193-4120 Page No. 10604 - 10616

Equation (6) represents if a city 'i' connected to city 'j' is 1, otherwise 0.

IV.NUMERICAL FORMULATION

The improvised concepts and algorithm are exemplified by a algebraic example in which cities total no. is N = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}. In the example, given below first the optimal solution in Lexi-search approach using the "Pattern Recognition Technique" from cities to the head quarter is derived. City "1"is considered the head quarter. The distance (i, i) is considered ∞ and also disconnectivity to the matching cities. D (i, j)'s considered natural numbers.The distance matrix D (i, j) is appearing in **Tab-1**

Tab – 1

		1	2	3	4	5	6	7	8	9	10
D (i,j)	1	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
	2	∞	∞	7	13	∞	7	2	∞	8	∞
	3	4	∞	∞	∞	∞	∞	∞	∞	∞	∞
	4	∞	6	15	∞	∞	4	8	∞	1	∞
	5	∞	∞	2	∞	∞	∞	3	14	4	9
	6	∞	14	10	6	∞	∞	6	∞	11	∞
	7	3	∞	∞	∞	∞	∞	∞	∞	∞	∞
	8	∞	∞	13	∞	5	∞	12	∞	5	14
	9	9	∞	∞	∞	∞	∞	∞	∞	∞	∞
	10	∞	∞	7	∞	12	∞	3	10	15	∞

Here we considered city 1 as head quarter, city 3, city 7, city 9 as sub head quarters defined as SH={3, 7, 9 }, first group consists of cities 2, 4, 6 mentioned as G_1={2, 4, 6 } and second group consists of cities 5, 8, 10 denoted by G_2={5,8,10}. The objective of the problem is that the cities of two groups should connect to head quarter city 1 through few cities in SH (sub head quarter) with minimum distance.

Hence D (4, 2) = 6

V. FEASIBLE SOLUTION

Suppose {(4,9),(5,3),(3,1),(8,5),(6,4),(2,6),(10,3),(9,1)} represents feasible solution. The following **fig-1** signifies the above feasible solution. The diamond shape represents headquarter, hexagon shape represents sub headquarter cities and rectangle shape represents groups which consists of cities that are represented by the circles. Value at arc epitomizes distance/cost between the respective two cities.

From the below **figure-1**, first group of cities are connected to headquarter city through the sub headquarter city 9 in such a way that cities 2 and 6 are mapped and cities 6 and 4 are mapped then city 4 is mapped to head quarter through sub headquart city 9. The second group of cities are connected to headquarter city through sub headquarter city 3 in such a way that city 8 is mapped to city 5 and city 5 is mapped to head quarter through sub headquarter city 3. City 10 is connected directly to the headquarter city through sub headquarter city 3. The distance/cost is

Z= D(4,9)+D(5,3)+D(3,1)+D(8,5)+D(6,4)+D(2,6)+D(10,3)+D(9,1)

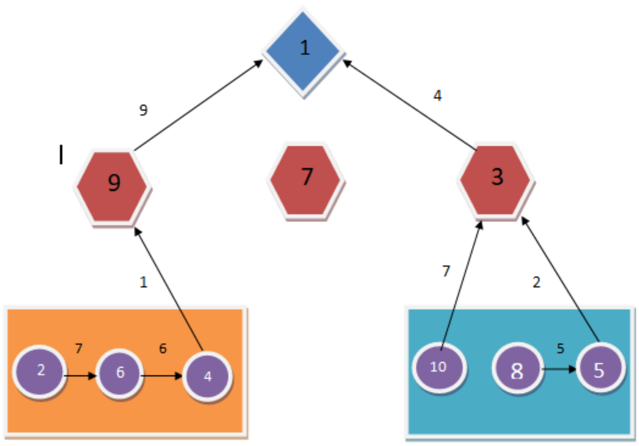
= 1+2+4+5+6+7+7+9

= 41

The total distances = 41 unit

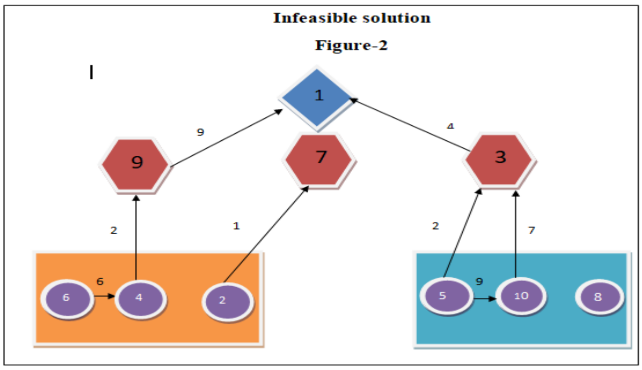
Feasible solution

Figure-1



VI. INFEASIBLE SOLUTION

The collection{(4,9),(2,7),(5,3),(3,1),(6,4),(10,3),(5,10),(9,1)} represents an infeasible solution for minimum spanning network connectivity from cities to the head quarter. **Fig-2**is an infeasible solution. The diamond shape gives headquarter, hexagon shape means sub headquarter cities and rectangle shape represents groups which consists of cities represented by circles. Entries at arc denotes distance/cost between the respective two cities. Consider a set ordered pairs {(4,9),(2,7),(5,3),(3,1),(6,4),(10,3),(5, 10),(9,1)} represents an infeasible solution.From the below**Fig-2**,



in first group, city 4 is associated to sub headquarter city 9 and city 2 is connected to sub headquarter city 7, according to the hypothesis, the cities in a group should be connected to the headquarter city through one sub headquarter city only.Initially, city 4 is in first group, connected to sub headquarter city 9. In the same way, city 2 should be connedceted to sub headquarter city 9.But here, the city 2 is connected to sub headquarter city 7 has indicated an infeasible solution. Further in second group, city 5 is mapped to the sub headquarter city 3 and mapped to city 10 which is also an infeasible solution. According to hypothesis, cities in a group shouldbe connected to the headquarter through the subheadquarter city only once either directly orindirectly. Here, city 5 is connected to sub headquarter 3 and connected from city 10 that indicated infeasible solution to this problem. The distance/cost is

$$Z = D(4,9)+D(2,7)+D(5,3)+D(3,1)+D(6,4)+D(10,3)+D(5,10)+D(9,1)$$
$$= 1+2+2+4+6+7+9+9$$
$$= 40.$$

The total distances = 40 units.

VII. ALGORITHMS

STEP0 : IX = 0 GOTO1
STEP1 : IS (IR (RA) = 1) IF YES GOTO 17
IF NO GOTO 2
STEP2 : GR = G (RA); GC = G (CA) GOTO 3

STEP3 : IS (GR = GC) IF YES GOTO 4
IF NO GOTO 5
STEP4 : IS (GR = 1) IF YES GOTO 17
IF NO GOTO 12
STEP5 : IS (GC = 0) IF YES GOTO 6
IF NO GOTO 7
STEP6 : IS (GR = 1) IF YES GOTO 10
IF NO GOTO 17
STEP7 : IS (GC =1) IF YES GOTO 8
IF NO GOTO 17
STEP8 : IS (CLR (GR) = 0) IF YES GOTO 15
IF NO GO TO 9
STEP9 : IS (CLR (GR) = CA) IF YES GOTO 16
IF NO GOTO 17
STEP10: IS (KLR = P-2) IF YES GOTO 11
IF NO GOTO 16
STEP11: IS (IC (RA) = 1) IF YES GOTO 16
IF NO GOTO 17
STEP12: W=CA GOTO 13

STEP13: IS [SW (W) =0] IF YES GOTO 16
IF NO GO TO 14
STEP14: W=SW (W)
IS (W=RA) IF YES GOTO17
IF NO GOTO 13
STEP15:CLR (GR) = CA
KLR=KLR+1GOTO 16
STEP16 : IX=1 GOTO 17
STEP17: STOP

ALGORITHM 2
(LEXI-SEARCH CALCULATION)

STEP0 : Initialization

The arrays SN, R, C, D, DC, KA, KB, KC, P, Q, R, M, V, LB and values of N are made available. SA, DR and IK are initialized. The values I=1, S=0, VT=9999 and Max= $(n^2 - 2n - 1) - (p - 2)$

STEP1 : S = S + 1
IS(S > Max) IF YES GOTO 7
IF NO GOTO 2

March - April 2020
ISSN: 0193-4120 Page No. 10604 - 10616

STEP2 : RA=R(S), CA=C(S)
V (I) = V (I-1) + D (S)
LB (I) = V (I) + DC (S+P-2 - I) – DC (S)
IS (LB (I) ≥ VT)IF YES GOTO 7
IF NO GOTO 3

STEP3 :Check Feasibility using Algorithm 1
STEP4 : IS (I =P-2) IF YES GOTO 6
IF NO GOTO 5
STEP5 : IR (RA) =1
IC (CA) = IC (CA) +1
L (I) = S
SW (CA) = RA
Max = Max + 1, I = I+1GO TO 1
STEP6 : VT = V (I) = LB (I)
(Print the full length word)
Record VT GOTO 8
STEP7 :IS (I = 1) IF YES GOTO 9
IF NO GOTO 8
STEP8 :L (I) = 0, I=I-1
IR (RA) =0
IC (CA) =IC (CA)-1
SW (CA) =0 GO TO 1
STEP9: STOP

VIII. SEARCH TABLE

The rules of receiving an optimal word using the above algorithm for the illustrative numerical example are given in the following Tab-2.

Tab-2

SN	1	2	3	4	5	6	7	8	V	LB	R	C	RE
1	1								1	22	4	9	A
2		2							3	22	2	7	R
3		3							3	24	5	3	A
4			4						6	24	5	7	R
5			5						6	26	7	1	R
6			6						6	28	10	7	R
7			7						7	31	3	1	A
8				8					11	31	4	6	R
9				9					11	33	5	9	R
10				10					12	35	8	5	A
11					11				17	35	8	9	R
12					12				18	37	4	2	R
13					13				18	38	6	4	A
14						14			24	38	6	7	R
15						15			25	39	2	3	R
16						16			25	40	2	6	A
17							17		32	40	10	3	A
18								18	40	40	2	9	R
19								19	40	40	4	7	R
20								20	41	41	5	10	R
21								21	41	41	9	1	A, =VT
22							18		33	41	2	9	R, =VT

23						17			25	41	10	3	R,=VT
24					14				18	39	6	7	R
25					15				19	41	2	3	R,=VT
26				11					12	37	8	9	R
27				12					13	39	4	2	R
28				13					13	40	6	4	A
29					14				19	40	6	7	R
30					15				20	42	2	3	R,>VT
31				14					13	42	6	7	R,>VT
32			8						7	33	4	6	R
33			9						7	35	5	9	R
34			10						8	38	8	5	A
35				11					13	38	8	9	R
36				12					14	40	4	2	R
37				13					14	41	6	4	R,=VT
38			11						8	40	8	9	R
39			12						9	42	4	2	R,>VT
40		4							4	27	5	7	A
41			5						7	27	7	1	A
42				6					10	27	10	7	A
43					7				14	27	3	1	R
44					8				14	28	4	6	R
45					9				14	30	5	9	R
46					10				15	32	8	5	A
47						11			20	32	8	9	R
48						12			21	33	4	2	R
49						13			21	34	6	4	A
50							14		27	34	6	7	R
51							15		28	35	2	3	R
52							16		28	35	2	6	A
53								17	35	35	10	3	R
54								18	36	36	2	9	R
55								19	36	36	4	7	R

March - April 2020
ISSN: 0193-4120 Page No. 10604 - 10616

56								20	37	37	5	10	R
57								21	37	37	9	1	A, =VT
58							17		28	36	10	3	R
59							18		29	37	2	9	R, =VT
60						14			21	35	6	7	R
61						15			22	36	2	3	R
62						16			22	37	2	6	R, =VT
63					11				15	33	8	9	R
64					12				16	35	4	2	R
65					13				16	36	6	4	A
66						14			22	36	6	7	R
67						15			23	37	2	3	R, =VT
68					14				16	37	6	7	R, =VT
69				7					11	29	3	1	R
70				8					11	31	4	6	R
71				9					11	33	5	9	R
72				10					12	35	8	5	A
73					11				17	35	8	9	R
74					12				18	37	4	2	R, =VT
75				11					12	37	8	9	R, =VT
76			6						7	29	10	7	A
77				7					11	29	3	1	R
78				8					11	31	4	6	R
79				9					11	33	5	9	R
80				10					12	35	8	5	A
81					11				17	35	8	9	R
82					12				18	37	4	2	R, =VT
83				11					12	37	8	9	R, =VT
84			7						8	32	3	1	R
85			8						8	34	4	6	R
86			9						8	36	5	9	R

March - April 2020
ISSN: 0193-4120 Page No. 10604 - 10616

87			10						9	39	8	5	R, > VT
88		5							4	29	7	1	A
89			6						7	29	10	7	A
90				7					11	29	3	1	R
91				8					11	31	4	6	R
92				9					11	33	5	9	R
93				10					12	35	8	5	A
94					11				17	35	8	9	R
95					12				18	37	4	2	R, = VT
96				11					12	37	8	9	R, = VT
97			7						8	32	3	1	R
98			8						8	34	4	6	R
99			9						8	36	5	9	R
100			10						9	39	8	5	R, > VT
101		6							4	32	10	7	A
102			7						8	32	3	1	R
103			8						8	34	4	6	R
104			9						8	36	5	9	R
105			10						9	39	8	5	R, > VT
106		7							5	35	3	1	A
107			8						9	35	4	6	R
108			9						9	37	5	9	R, = VT
109		8							5	37	4	6	R, = VT
110	2								2	25	2	7	A
111		3							4	25	5	3	A
112			4						7	25	5	7	R
113			5						7	27	7	1	A
114				6					10	27	10	7	R
115				7					11	29	3	1	A
116					8				15	29	4	6	A
117						9			19	29	5	9	R
118						10			20	31	8	5	A
119							1		2	3	8	9	R

March - April 2020
ISSN: 0193-4120 Page No. 10604 - 10616

							1		5	1			
120							12		26	32	4	2	R
121							13		26	32	6	4	R
122							14		26	33	6	7	A
123								15	33	33	2	3	R
124								16	33	33	2	6	R
125								17	33	33	10	3	A,=VT
126							15		27	34	2	3	R,>VT
127						11			20	32	8	9	R
128						12			21	33	4	2	R,=VT
129					9				15	31	5	9	R
130					10				16	33	8	5	R,=VT
131				8					11	31	4	6	A
132					9				15	31	5	9	R
133					10				16	33	8	5	R,=VT
134				9					1	3	5	9	R,
									1	3			=VT
135			6						7	29	10	7	R
136			7						8	32	3	1	A
137				8					12	32	4	6	A
138					9				16	32	5	9	R
139					10				17	34	8	5	R,>VT
140				9					12	34	5	9	R,>VT
141			8						8	34	4	6	R,>VT
142		4							5	28	5	7	R
143		5							5	30	7	1	A
144			6						8	30	10	7	R
145			7						9	33	3	1	R,=VT
146		6							5	33	10	7	R,=VT
147	3								2	28	5	3	A
148		4							5	2	5	7	R

March - April 2020
ISSN: 0193-4120 Page No. 10604 - 10616

										8			
149		5							5	30	7	1	A
150			6						8	30	10	7	R
151			7						9	33	3	1	R, = VT
152		6							5	33	10	7	R, = VT
153	4								3	31	5	7	A
154		5							6	31	7	1	A
155			6						9	31	10	7	A
156				7					13	31	3	1	A
157					8				17	31	4	6	A
158						9			21	31	5	9	R
159						10			22	33	8	5	R, = VT
160					9				17	33	5	9	R, = VT
161				8					13	33	4	6	R, = VT
162			7						10	34	3	1	R, > V
													T
163		6							6	34	10	7	R, > VT
164	5								3	34	7	1	R, > VT

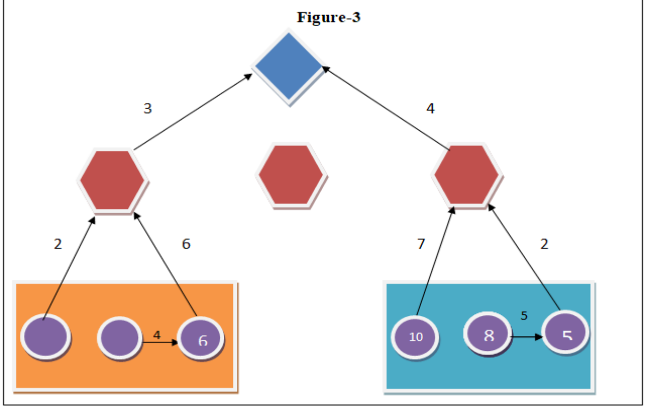
IX. COMMENTS

Tab – 6, provides optimal solution with the optimal word is $\mathbf{L_8}$ = **(2, 3, 5, 7, 8, 10, 14, 17)**. Optimal solution at final state **VT is 33** at final state which is appearing in the **125th** rowand the corresponding ordered pairs are **(2,7),(5,3),(7,1),(3,1),(4,6),(8,5),(6,7),(10,3)**.

Tab– 3

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
L	2	3	5	7	8	10	14	17	-	-
IR	-	1	1	1	1	1	1	1	-	1
IC	1+1	-	1+1	-	1	1	1+1	-	-	-
SW	-	7	1	6	3	7	1	5	-	3

The ordered pairs set **{(2,7), (5,3), (7,1), (3,1), (4,6), (8,5), (6,7), (10,3)}** represents an optimal solution.



Optimal solution

Figure-3

TEST Engineering & Management

March - April 2020
ISSN: 0193-4120 Page No. 10604 - 10616

From the above **Fig-3**, the first group of cities are connected to the headquarter city through a sub headquarter city 7, in such a way, that city 2 is connected to the headquarter through a sub head quarter 7. City 4 is mapped to city 6 and city 6 is mapped to the head quarter through a sub headquart city 7. Similarly, second group of cities are connected to the headquarter city through a sub headquarter city 3, in such a way, that city 8 is mapped to city 5 and city 5 is mapped to the head quarter through a sub headquarter city 3. City 10 is connected to the headquarter city through a sub headquarter city 3. The distance/cost

$$Z = D(2,7) + D(5,3) + D(7,1) + D(3,1) + D(4,6) + D(8,5) + D(6,7) + D(10,3)$$

$$= 2 + 2 + 3 + 4 + 4 + 5 + 6 + 7$$

$$= 33$$

The total distances = 33 units.

Consider an ordered pair set**{(2,7),(5,3),(7,1),(3,1),(4,6),(8,5),(6,7),(10,3)}**, which is a feasible solution

tab-4

$$X(i,j) = \begin{bmatrix} 0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&1&0&0&0 \\ 1&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&1&0&0&0&0 \\ 0&0&1&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&1&0&0&0 \\ 1&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&1&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0 \\ 0&0&1&0&0&0&0&0&0&0 \end{bmatrix}$$

X. Experimental Resultsand Comparison Details

The table given below displays the computational results for Lexi-Search algorithm using PRT. H= Head quarter, SH = number of sub headquarters, P = the number of groups/clusters, G_1, G_2 and G_3 are groups/clusters some cities.

Table-5

S. No	No. of cities	No. of clusters	Published model	Proposed model
1	18	3	0.879121	0.8432
2	20	3	1.703386	0.8655
3	22	3	0.76923	0.7593
4	25	3	1.923077	0.9124

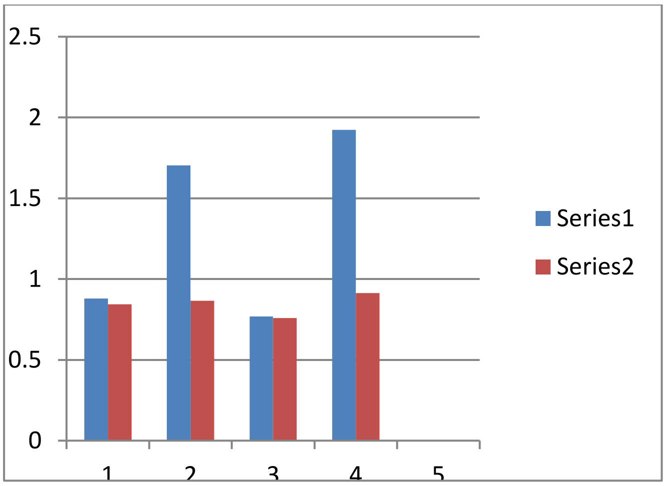
Table – 6

SN	N	H	SH	P			VT	CPU Run Time in seconds
				G1	G2	G3		Avg. AT + ST
1	5	1	2	2	-	-	35	0.0000
2	6	1	2	2	2	-	47	0.0000
3	8	1	3	2	2	-	54	0.0000
4	9	1	3	3	2	-	62	0.1234
5	9	1	3	2	3	-	65	0.1586
6	10	1	3	4	2	-	179	0.2357
7	10	1	3	3	3	-	142	0.2531
8	10	1	3	2	4	-	156	0.3258
9	11	1	3	2	3	2	194	0.5493
10	11	1	3	3	2	2	172	0.6123

11	12	1	3	3	2	3	185	0.6349
12	15	1	4	3	4	3	189	0.7352
13	18	1	5	4	5	3	196	0.8432
14	20	1	5	4	5	5	201	0.8655
15	22	1	6	5	6	4	224	0.7593
16	25	1	7	6	6	5	247	0.9124

In the graph given below second series characterize that CPU run time to achieve optimal solution by proposed model and first series denote that CPU run time for examining the optimal solution by the published model. Moreover the proposed model receives less time than published model in presenting the solution.

GRAPH-1



Using this series2 gives CPU run time to obtain optimal solution by proposed model and series 1 gives CPU run time for examining the optimal solution by the published model. Also the proposed model assumes less time than published model for giving the solution.

XI. CONCLUSION AND FUTURE RESEARCH

In the above discussion an exact algorithm called LSA (Lexi-Search Algorithm) built on PRT (Pattern Recognition Technique)has been presented in order to solve the **MINIMUM SPANNING CONNECTIVITY OF CLUSTERED CITIES TO THE HEADQUARTER CITY**. LSAs have been shown to be more effective in a large number of combinatorial situations. At initial stage a model is framed into a 0-1 programming phenomena and then an LSA based on PRT is established for obtaining an optimal solution. The problem is exemplified with appropriate numerical example. Using C-language the algorithm is proposed and the calculations are reported and compared with the existing results. One can observe that to obtain an optimal solution the CPU run time is reasonably less for higher values of the problem than the published model. Using these results one can strongly contemplate that the proposed algorithm can perform larger size problems. In the context of future research two problems namely Vehicle Routing Problem with Inter-Loading Facilities and a Variant Constraint Bulk Transshipment Problem can be proposed and analyzed by means of C-Language

References

1. Gendreau,M., Laporte, G., and Potvin, J.-Y.(2002): *" Metaheuristics for the capacitated vehicle routing problem,"* in: P. Toth, D. Vigo (Eds.), The Vehicle Routing Problem,Monographs on Discrete Mathematics and Applications, SIAM, Philadelphia, pp. 129–154.
2. Hollis, B.L., Forbes,M.A., and Douglas,B.E. (2006): *"Vehicle routing and crew scheduling for metropolitan mail distribution at Australia Post,"* European Journal of Operational Research, vol. 173, no. 1, pp. 133–150.
3. Baldacci, R., Hadjiconstantinou, E., and Mingozzi,A.(2003): "An exact algorithm for the traveling salesman problem with deliveries and collections," Networks, vol. 42, no.1, 26–41.
4. Anju Gupta, VanitaVerma, and Puri, M. C. (1996): "Time-cost Trade-off- Relations in Bulk Transportation Problem," Journal of Information and Optimization Sciences, 16(2), 317-325.
5 .Borovska P., Lazarova, M. and Bahudejla, S. (2007): *"Strategies for Parallel Genetic Computation of Optimization Problems,"* Journal Biotechnologies & Biotechnological Equipment, Vol.21, No.2, pp.241-246.

6. Abuali, F. N., Wainwright, R. L. and Schoenefeld, D.A.(1995): “Determinant factorization: A new encoding scheme for spanning trees applied to the probabilistic minimum spanning tree problem,” in: L.J. Eshelman (Ed.), Proceedings of the Sixth International Conference on Generic Algorithms, Morgan Kaufmann Publishers, San Francisco, California, pp. 470 - 475.

7. Herer, Y.T., Tzur, M., and Yücesan, E. (2006)): “The multi-location transshipment problem,” IIE Transactions, 38, pp. 185–200.
8. Gendreau,M., Laporte, G., and Potvin, J.-Y.(2002): “ Metaheuristics for the capacitated vehicle routing problem,” in: P. Toth, D. Vigo (Eds.), The Vehicle Routing Problem, Monographs on Discrete Mathematics and Applications, SIAM, Philadelphia, pp. 129–154
9. G.Vijayalakshmi (2013), Lexi-Search Approach to Travelling Salesman Problem, IOSR Journal of Mathematics (IOSR-JM), Volume6, Issue4, May-June2013, Pp1-08
10. K.ChendraSekhar et.al (2012),A Problem Recognition Lexisearch Approach to traveling salesman problem with additional constraints,International Journal on Computer Science and Engineering(IJSCE),vol4.No 02,Feb2012 pgs307-320
11. Zakir Hussain Ahmed (2011): A data-guided lexisearch algorithm for the bottleneck travelling salesman problem, International Journal of Operations Research 12(1):20-33
12. Abuali, F. N., Wainwright, R. L. and Schoenefeld, D.A.(1995): “Determinant factorization: A new encoding scheme for spanning trees applied to the probabilistic minimum spanning tree problem,” in: L.J. Eshelman (Ed.), Proceedings of the Sixth International Conference on Generic Algorithms, Morgan Kaufmann Publishers, San Francisco, California, pp. 470 – 475
13. ShaliniArora et.al (2017):”A Lexisearch Algorithm for a time minimizing assignment problem”, OPSEARCH 35,193-213(1998), Springer Link
14. Amit Kumar, AmarpreetKaur, and Anila Gupta. (2011): “New methods for solving fuzzy transportation problems with some additional transshipments,” ASOR Bulletin, 30(1), 42-61.
15. Kek, A.G.H., Cheu,R.L., and Meng,Q.(2008): “Distance-constrained capacitated vehicle routing problems with flexible assignment of start and end depots,” Mathematical and Computer Modelling, vol. 47, no. 1-2, pp. 140–152
16. Archetti, C., Hertz, A., and Speranza, M.G. (2006): “A tabu search algorithm for the split delivery vehicle routing problem,” Transportation Science 40, 64 - 73.
17. Balakrishna, U. (2009): [a] “Generalized Time Dependent Travelling Salesman Problem with Cyclic constraint [GTSP]”, [b] “Generalized Time Dependent Travelling Salesman Problem (cluster constraint) [GTDTSP] models,”
18. Bhavani, V. &Sundara Murthy M. (2006): “Truncated M-Travelling Salesmen Problem,” OPSEARCH, 43(2).
19. Borovska P., Lazarova, M. and Bahudejla, S. (2007): “Strategies for Parallel Genetic Computation of Optimization Problems,” Journal Biotechnologies & Biotechnological Equipment, Vol.21, No.2, pp.241-246.
20. Lee, Y.H., Jung, J.W., and Jeon, Y.S. (2007): “An effective lateral transshipment policy to improve service level in the supply chain,” International Journal of Production Economics, Vol. 106, 115–126.
21. Khurana, A. and Arora, S. (2011): “Solving transshipment problems with mixed constraints,” International Journal of Management Science and Engineering Management, 6(4):292–297.
22. Kek, A.G.H., Cheu,R.L., and Meng,Q.(2008): “Distance-constrained capacitated vehicle routing problems with flexible assignment of start and end depots,” Mathematical and Computer Modelling, vol. 47, no. 1-2, pp. 140–152.
23. Karuno, Y., Tachibana, T., and Yamashita, K. (2009): “A transsipment problem with a permutable transit vector,” Proceedings of JSME/SSJ International Symposium on Scheduling2009, 163–169.