1	
2	
3	Supplemental Online Material
4 5 6	The pseudo code of the Travel Path Distance calculation between $TF_1$ and $TF_2$
7 8	Start Travel_Path_Distance
9 10 11	Input a set of ARCs
12 13	Input TF <sub>1</sub> and TF <sub>2</sub>
14 15	***Step 1***
17 18	For each coordinate c in ARCs
19 20	Find $c$ , which gives the minimum Euclidean distance between $c$ and TF <sub>1</sub>
21 22 23	Set $c$ as SP; and the minimum distance as $L_1$
24 25	Find <i>c</i> , which gives the minimum Euclidean distance between <i>c</i> and $TF_2$
26 27 28	Set <i>c</i> as FP; and the minimum distance as $L_3$
29 30	Next coordinate c
31 32 33	***Step 2*** CP - SP
34	Cr – Sr
35 36 37	Do While CP <> FP
38 39	**2.1** Set Array NP(1 to 4)
40 41 42	For $n = 1$ to 4
43 44	For each coordinate c in ARCs If the Euclidean distance between c and $CP = 1$ Then
45 46 47	NP( $n$ ) = $c$
48 49	n = n + 1
50 51 52	End If
53 54	Next coordinate c
55 56 57	Next <i>n</i>
58 59	** $2.2$ ** For $n = 1$ to 4
60	

2	
3	If $NP(n) \leq PP$ Then
4	
5	$\Gamma_{in} = 1 \text{ ND}(x)$ and $\Gamma_{in} = 1$
6	Find $NP(n)$ which gives the minimum Euclidean distance between $NP(n)$ and FP
7	
8	Set <i>n</i> as <i>min1n</i>
9	
10	Find NP( $n$ ) which gives the second minimum Euclidean distance between NP( $n$ ) and FP
11	That $(n)$ which gives the second minimum Excitation distance between $(n)$ and $(1)$
12	
13	Set <i>n</i> as <i>min2n</i>
14	
15	End If
16	
17	If this CP is a Turning Point Then
18	
19	TD - CD
20	Ir - Cr
21	
22	PTP = PP
23	
24	End If
25	
26	Next n
27	
28	
29	**2.3** If outward $\geq$ maxoutward Then this move is not allowed so select a fix
30	
31	method
32	
33	Method 1 Switch the Start and Finish Points
34	
35	CD = ED: $ED = SD$ : $SD = CD$ : and recolvulate <i>L</i> from the haging ing
36	$Cr - rr$ , $rr - Sr$ , $Sr - Cr$ , and recalculate $L_2$ from the beginning
37	
38	Method 2 Return to the Turning Point and take a new turn
39	
40	CP = TP; $PP = PTP$ ; and take the second min next point
41	
42	Method 3 Increase margutward
43	Wethod 5 meredse maxbalwara
44	
45	**2.4** Else outwara < maxoutwara 1 nen this move is allowed so update the points
46	
47	PP = CP
48	
49	If move to the min next point Then $CP = NP(minn)$
50	
51	Else move to the second min part point Then $CP = NP(min 2n)$
52	Ense move to the second minimext point Then $CF = NF(mm2n)$
53	
54	End If
55	
56	$L_2 = L_2 + 1$
57	
58	If the Euclidean distance between CP and FP is shorter or moving toward FP Then
59	If the Euclidean distance between of and fit is shorter of moving toward ff filen
60	

	outward = 0
	Else the Euclidean distance between CP and FP is longer or moving outward FP Then
	outward = outward + 1
	End If
	End If
	Loop
	***Step 3***
	If $L_1 \leq L_2$ or $L_3 \leq L_2$ Then
	Travel_Path_Distance = $L_1 + L_2 + L_3$ (Equation (5))
	Else
	Travel_Path_Distance = Euclidean distance between $TF_1$ and $TF_2$
	End If
	End Travel_Path_Distance
The pse	eudo code of the Particle Swarm Optimisation (PSO) algorithm
Ş	Start PSO algorithm
]	Initialise input variables
;	***Step 1. Encode and populate the first generation of Particles in the swarm***
]	For each Particle <i>i</i> in <i>swarmsize</i>
	Populate Particle <i>i</i>
	If Particle <i>i</i> is invalid Then repopulate it
]	Next Particle
]	Do While the ending criteria are not met
;	***Step 2. Evaluate the fitness value of particles and remember the best particles***
]	For each Particle <i>i</i> in <i>swarmsize</i>

2 3	Calculate the Fitness value using Equation (1)
4	
5	If the Fitness value is better than the individual $pbest_i$ value Then
6	
7 8	Set the current Particle value as the new <i>phest</i> .
9	
10	End If
11	
12	If the Fitness value is better then the global chest value Then
13	If the Fitness value is better than the global goest value Then
14	
15	Set the current Particle value as the new gbest
16	
17	End If
18	
19	Next Particle
20	
22	***Step 3. Update velocities and positions of Particles in the swarm***
23	
24	For each Particle <i>i</i> in <i>swarmsize</i>
25	
26	For each Facility f in a Particle
27	
28	Generate the random variable of Facility f
29	
3U 21	Calculate the velocity of Facility f of the Particle using Equation (6)
37	Calculate the velocity of Facility for the Facilitie using Equation (0)
33	Undate the position of Equility $f$ of the Particle using Equation (7)
34	Optime the position of Facility for the Fattlete using Equation (7)
35	If Facility fof Domiala i is within the site hour damy (Fountion (2))
36	If Facility $j$ of Facilicie $i$ is within the site boundary (Equation (2))
37	If Facility (af Dentials i down at some law with the other facilities (Fountion (2))
38	If Facility <i>f</i> of Particle <i>t</i> does not overlap with the other facilities (Equation (3))
39	
40	If Facility f of Particle i is satisfied with inter-facility distance constraint (Equation (4))
41	
42	Then
44	
45	Exit For; this Facility f of Particle i is valid
46	
47	Else
48	
49	Reset previous facilities before Facility f
50	
51	Loop and regenerate new position of Facility f
52	
55 54	End If
55	
56	Next Facility
57	TYORE I WORTLY
58	Next Particle

1	
2	
3	***Step 4. Evolve the swarm until the ending criteria are met***
4	
5	I can until the ording aritaria are mot
6	Loop until the ending criteria are met
7	
8	End PSO algorithm
9	
10	
11	
12	
13	
14	
15	
16	
17	
18	
19	
20	
21	
22	
23	
24	
25	
26	
27	
28	
29	
30	
31	
32	
33	
34	
35	
30	
رد در	
20	
<u>40</u>	
-τυ Δ1	
Δ <sup>2</sup>	
43	
44	
45	
46	
47	
48	
49	
50	
51	
52	
53	
54	
55	
56	
57	

URL: http://mc.manuscriptcentral.com/rcme Email: RCME-peerreview@journals.tandf.co.uk