MAGNETIC OPTIMIZATION ALGORITHM APPROACH FOR TRAVELLING SALESMAN PROBLEM

ENGR. MOHD MUZAFAR BIN ISMAIL
MUHAMMAD IQBAL ZAKARIA
AMAR FAIZ ZAINAL ABIDIN
JUWITA MAD JULIANI
ASRANI LIT
SEYEDALI MIRJALILI
NUR ANIS NORDIN
MUHAMMAD FAIZ MOHAMED SAAID

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Magnetic Optimization Algorithm Approach For Travelling Salesman Problem

Mohd Muzafar Ismail, Muhammad Iqbal Zakaria, Amar Faiz Zainal Abidin, Juwita Mad Juliani, Asrani Lit, Seyedali Mirjalili, Nur Anis Nordin, Muhammad Faiz Mohamed Saaid.

Abstract—**Lately, numerous nature inspired optimization techniques has been applied to combinatorial optimization problems, such as Travelling Salesman Problem. In this paper, we study the implementation of one of the nature inspired optimization techniques called Magnetic Optimization Algorithm in Travelling Salesman Problem. In this implementation, each magnetic agent or particle in Magnetic Optimization Algorithm represents a candidate solution of the Travelling Salesman Problem. The strength of the magnetic force between these particles is inversely proportion to the distance calculated by the Traveling Salesman Problem's solution they represented. Particles with higher magnetic force will attract other particles with relatively lower magnetic force, towards it. The process repeated until satisfying a stopping condition, and the solution with lowest distance is considered as the best-found solution. The performance of the proposed approach is benchmarked with a case study taken from a well-known test bank.**

Index Terms — **Combinatorial Problem; Magnetic Optimization Algorithm; Travelling Salesman Problem.**

I. INTRODUCTION

TRAVELING Salesman Problem (TSP) is a well-known problem which contains a study for finding the shortest probable distance that could be taken by the salesman to travel to several number of given cities just once. Next section will explained in great details the mathematical formulation of TSP. Application of TSP can be seen in many areas such as routing for automated holes drilling process [1-3] and wires routing in very large scale integrated circuit (VLSI) [4]. In the applications mentioned, additional distance covered involved additional costs in term of monetary or time. This is a great motivation for academician, engineers & mathematicians to propose new methods in solving TSP effectively.

In 1985, V. Černý wrote a paper proposing the thermodynamically approach to this problem. In the research, the author presented a Monte Carlo algorithm [5]. S. Lin and B.W. Kernighan used the heuristic algorithm to solve TSP. They used a highly effective heuristic procedure for up to 110 cities [6]. F. H. Khan *et. al* proposed a representation method based on chromosome model using binary matrix and suggested a new fittest criteria to find the optimal solution for TSP. This method is better known as genetic algorithm [7]. In 2011, Weiqi Li introduced a multi-start search approach to dynamic TSP [8]. The algorithm includes the interaction of change and search over time. M. Dorigo and L. M. Gambardella proposed the ant colony system (ACS) algorithm where a set of cooperating agents using an indirect communication mediated by a pheromone they produce on the edges of TSP graph for good solution [9]. S. Yadlapalli *et. al* presented an approximation algorithm for a two-depot, heterogenous TSP with an approximation ratio of 3 with condition of symmetrical cost and satisfaction of triangle inequality [10]. In 2006, Z. Pizlo *et. al* tested the human performance on Euclidean TSP with a pyramid model [11]. C. S. Helvig *et. al* introduced a time-dependent generalization of TSP where a pursuer must intercept in minimum time a set of targets which move with constant velocities [12]. Gilbert Laporte used the fundamental approach of solving TSP which is the exact and approximate algorithms. The author discussed the best of the two algorithms that have been developed to solve TSP [13].

II. FORMULATION OF TRAVELING SALESMAN PROBLEM

Many literatures describes the mathematical representation of TSP. In [15], the authors described TSP as follows: "The TSP can be defined on a complete undirected graph $G = (V, E)$ if it is symmetric or on a directed graph $G = (V, A)$ if it is asymmetric. The set $V = \{1, \dots, n\}$ is the vertex set, $E = \{(i,j): i,j \in V, i < j\}$ is an edge set and $A = \{(i,j): i,j \in V, i \neq j\}$ is an arc set. A cost matrix $C = (c_{ij})$ is defined on E or on A. The cost matrix satisfies the triangle inequality whenever $c_{ij} \leq c_{ik} + c_{kj}$ for all i,j,k. In particular, this case planar problems for which the vertices are

Mohd Muzafar Ismail is with the Faculty of Electronic and Computer Engineering, Universiti Teknikal Malaysia Melaka, 76100 Durian Tunggal, Hang Tuah Jaya, Melaka, Malaysia. (email: muzafar@utem.edu.my).

Muhammad Iqbal Zakaria, Amar Faiz Zainal Abidin, Juwita Mad Juliani and Asrani Lit are/were with Faculty of Electrical Engineering, Universiti Teknologi Malaysia, 81310 UTM Johor Bahru, Johor, Malaysia (email: iqbal.zakaria@fkegraduate.utm.my, anarfaiz@fke.utm.my)

Amar Faiz Zainal Abidin is also with the Laboratoire Electronique, Informatique et Image Université de Bourgogne Le2i,12 rue de la fonderie, 71200 Le Creusot, France.

Seyedali Mirjalili, and Nur Anis Nordin are with the Faculty of Computer Science and Information System, Universiti Teknologi Malaysia, 81310 UTM Johor Bahru, Johor, Malaysia. (email: ali.mirjalili@gmail.com, nuranisnordin@gmail.com)

Muhammad Faiz Mohamed Saaid is with the Department of Biomedical Engineering, Faculty of Engineering, Universiti Malaya, 50603 Kuala Lumpur, Malaysia.(email: faizsaaid17@gmail.com)

$c_{ij} = \sqrt{(X_i - X_j)^2 + (Y_i - Y_j)^2}$ is the Euclidean distance. The triangle inequality is also satisfied if c_{ij} is the length of a shortest path from i to j on G."

In several literatures, c_{ij} is calculated using Manhattan distance as (1).

$$c_{ij} = |X_i - X_j| + |Y_i - Y_j| \quad (1)$$

Other than the definition above, the cost mathematical formulation of a TSP can be said as in (2) [16].

$$c_{total} = \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} \times b_{ij} \quad (2)$$

where

$$\sum_{i=1}^{n} b_{ij} = 1 \; for \; i \in V, i \neq j \quad (3)$$

$$\sum_{j=1}^{n} b_{ij} = 1 \; for \; j \in V, j \neq i \quad (4)$$

$$b_{ij} = 0 \; or \; 1 \; for \; (i,j) \in A \quad (5)$$

Figure 1 and Table 1 shows a simple example on how the mathematical notation works. Given that a salesman from #0 and he wants to visit #1, #3 and #2 before going back to #0. The distance traveled by the salesman can be calculated as shown in Table 1. In this case the cost formulation or distance traveled is calculated using (2). This to help reader to digest the information, easily. In practice, the cost (distance) is calculated using Euclidean distance. For implementation, a case study taken from TSPLIB [17]. This case study consists of 14 cities in Burma. The Cartesian coordinates (x,y) of the cities are given in Table 2.

III. Modeling Traveling Salesman Problem using Magnetic Optimization Algorithm

The magnetic force is one of four basic forces that occurred in our universe [14]. In magnetic field, a particle with higher mass will have a greater magnetic force that attracts other particles with smaller masses to move towards it. This simple concept is adapted to MOA where the mass of a particle at a given time is proportional to the fitness of the problem. According to [14], the cellular lattice model defines how magnetic particle could relate to each other. In other words, this model suggested that each particle could only be influenced by the magnetic force of its neighborhood. This as shown in Figure 2. This model is proposed to ensure that MOA will not prematurely convergence, which leads to local optima solution.

The important part of this paper is how TSP can modeled using MOA. We suggested that each particle in MOA represented a candidate solution of TSP. The candidate solution can be described by its position in the search space.

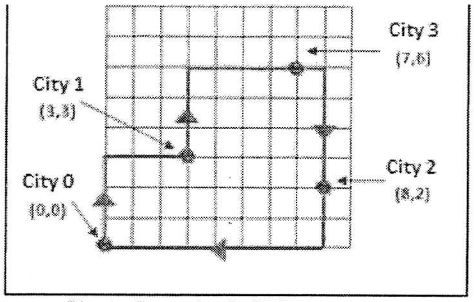


Fig. 1. Example of a TSP with 4 cities.

TABLE I
CALCULATION OF TOTAL DISTANCE IN FIGURE 1

$i \to j$	c_{ij}	p_{ij}	$c_{ij} \times p_{ij}$
$0 \to 0$	$\|0-0\| + \|0-0\| = 0$	0	0
$0 \to 1$	$\|0-3\| + \|0-3\| = 6$	1	6
$0 \to 2$	$\|0-8\| + \|0-2\| = 10$	0	0
$0 \to 3$	$\|0-7\| + \|0-6\| = 13$	0	0
$1 \to 0$	$\|3-0\| + \|3-0\| = 6$	0	0
$1 \to 1$	$\|3-3\| + \|3-3\| = 0$	0	0
$1 \to 2$	$\|3-8\| + \|3-2\| = 6$	0	0
$1 \to 3$	$\|3-7\| + \|3-6\| = 7$	1	7
$2 \to 0$	$\|8-0\| + \|2-0\| = 10$	1	10
$2 \to 1$	$\|8-3\| + \|2-3\| = 6$	0	0
$2 \to 3$	$\|8-7\| + \|2-6\| = 5$	0	0
$2 \to 2$	$\|8-8\| + \|2-2\| = 0$	0	0
$3 \to 0$	$\|7-0\| + \|6-0\| = 13$	0	0
$3 \to 1$	$\|7-3\| + \|6-3\| = 7$	0	0
$3 \to 2$	$\|7-8\| + \|6-2\| = 5$	1	5
$3 \to 3$	$\|7-7\| + \|6-6\| = 0$	0	0
$d_{travel} = \sum_{i=1}^{n}\sum_{j=1}^{n} c_{ij} \times p_{ij}$			28 units

TABLE II
COORDINATES OF THE CASE STUDY, BURMA14

Cities	Coordinates	Cities	Coordinates
1	(16.47, 96.10)	2	(16.47, 94.44)
3	(20.09, 92.54)	4	(22.39, 93.37)
5	(25.23, 97.24)	6	(22.00, 96.05)
7	(20.47, 97.02)	8	(17.20, 96.29)
9	(16.30, 97.38)	10	(14.05, 98.12)
11	(16.53, 97.38)	12	(21.52, 95.59)
13	(19.41, 97.13)	14	(20.09, 94.55)

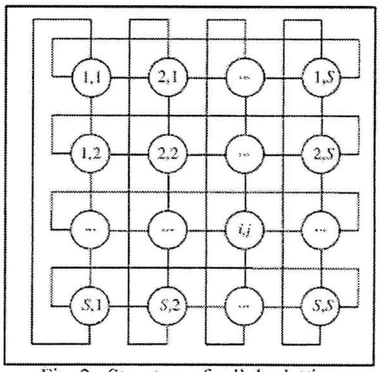


Fig. 2. Structure of cellular lattice

TSP can be generalized as in (6).

$$x_{ij,k}^{z,t} = [vote\ for\ 1st\ city, \ldots, vote\ for\ nth\ city] \quad (6)$$

where x^z is the z[th] particle position in search space. i and j indicate the coordinate of the particle in cellular lattice formation mentioned earlier. k is the dimension of the particle position. Maximum dimension of a particle position in TSP is equal to the number of city need to be visited. The city with highest vote will be the 1[st] city to be visited by the salesman while the city with the least vote will be the last city to be visited before the salesman returned to the original city (#0). For example given that $x_{11}^{z=1} = [1, 10, 3]$ and $x_{31}^{z=7} = [12, -7, 2]$. The 1[st] particle suggests a solution of $0 \rightarrow 2 \rightarrow 3 \rightarrow 1 \rightarrow 0$ and the 7[th] particle suggests the solution of $0 \rightarrow 1 \rightarrow 3 \rightarrow 2 \rightarrow 0$.

Similar to other nature-inspired optimization techniques, MOA consists of 3 main parts: initialization, fitness evaluation, and improvement of agents. Algorithm 1 shows the pseudo code of MOA for TSP.

Algorithm 1: Pseudo Code of MOA for Travelling Salesman Problem (TSP)

```
Initialize TSP parameters: n and c_ij
Initialize MOA parameters: S, ρ, α, w and h
Randomly positioned the particle in search space with a
 cellular lattice-like structure
while not termination condition do
  t = t + 1
  Find each particle's fitness using (2) and store it as
   magnetic field, b^z
  if the particle fitness greater than global best do
    Store solution offered by the particle and the distance
     value
  end
  Normalize b^z using (8)
  Evaluate the particle's mass, m_ij^{z,t} using (9)
  for all particles x_ij^{z,t} in X^t do
    Let f_ij^z = 0
    Find n_ij^z
    for all particles x_ij^{z,t} in N_ij do
      update f_ij^z using (10)
    end
    for all particles x_ij^{z,t} in X^t do
      update v_ij,k^{z,t+1} using (11)
      update x_ij,k^{z,t+1} using (12)
      perform correction if necessary
    end
  end
end
Display global best solution
```

The algorithm starts by initializing TSP and MOA parameters. The parameters and the suggested values are listed in table 2 using the proposed model mentioned earlier. This can be mathematically written as (7).

$$x_{ij,k}^{z,t} = R(u_k, l_k) \text{ for } i,j = 1,2,..,S, k = 1,2,..,n \text{ and } t = 0 \quad (7)$$

By taking the example earlier with additional particle, these particles position after randomly assigned are $x_{11}^{1,0} = [1, 10, 3]$, $x_{31}^{7,0} = [12, -7, 2]$ and $x_{22}^{5,0} = [7, 10, 21]$,. These steps are the first phase of any nature-inspired optimization techniques: the initialization phase.

In this paper, the proposed approach uses maximum iteration as the stopping condition. Then, we enter the 2[nd] phase of any nature-inspired optimization techniques: the fitness evaluation. The fitness of the particles are calculated using equation X and stored as magnetic field, b^z. In this example the fitness values of particles 1, 5 and 7 are 0.0357, 0.0333 and 0.0357, respectively. Thus, $b^{z=1} = 0.0357$, $b^{z=5} = 0.0333$ and $b^{z=7} = 0.0357$. Then, we normalized the magnetic value using (8).

$$b_{ij}^z = \frac{b_{ij}^z - \min_{i,j=1\rightarrow S}(B_{ij}^t)}{\max_{i,j=1\rightarrow S}(B_{ij}^t) - \min_{i,j=1\rightarrow S}(B_{ij}^t)} \quad (8)$$

In this case the new magnetic field value for particle 1, 5 and 7 are 1, 0 and 1. After the magnetic fields of the particles are calculated, the mass of the particles, m_{ij}^z are calculated using (9). α and ρ are constant parameters of MOA. Following are the 3[rd] phase of any nature-inspired optimization algorithm, the improvement of the agents or the learning phase.

$$m_{ij}^z = \alpha + \rho \times b_{ij}^z \quad (9)$$

If $\alpha = 1$ and $\rho = 1$, the mass of the particles are 2, 1 and 2. Then, identify the lattice neighbors of the particles. Next, the magnetic force, f_{ij}^z is calculated using (10).

$$f_{ij}^z = \frac{\left(x_{uv,k}^{z,t} - x_{ij,k}^{z,t}\right) \times b_{uv}^{z,t}}{\sqrt{\frac{1}{n} \times \sum_{k=1}^{n} \left(\frac{x_{uv,k}^{z,t} - x_{ij,k}^{z,t}}{u_k - l_k}\right)^2}} \quad (10)$$

Based on the force, the particle velocity and the particle position is updated using (11) and (12).

$$v_{ij,k}^{z,t+1} = \frac{f_{ij}^{z,t}}{m_{ij}^z} \times R(u_k, l_k) \quad (11)$$

$$x_{ij,k}^{z,t+1} = x_{ij,k}^{z,t} + v_{ij,k}^{z,t+1} \quad (12)$$

After updating the particle position, it is necessary to check for any redundancy in the value of the voting. For example, $x_{11}^{1,1} = [9, 9, 6]$ leads to an invalid solution. Here, it can be simply corrected by initializing that particle using (7). The process continues to repeat until maximum iteration is met. Then the global best solution is taken as final solution.

As mentioned in section 2, the case study is taken from a website hosted by University of Heidelberg. The website provided ample number of problems related to TSP. A TSP consists of 14 cities in Burma is taken as a case study due to its popularity. The coordinates of the cities have been listed in table 2. For the case study, MOA parameters used are as listed in table 3. The computation is done using a laptop equipped with Intel Centrino Duo, 1.8GHz with 2GB RAM. The result obtained from the case study listed in table 4. Table 5 listed all the images of the best-found route for all computations. The best solution found is on the 10th run, which has a distance of 31.376. This solution is better than the suggested solution by TSPLIB, which are 33.23.

V. CONCLUSION

This paper presents the application of MOA with voting modeling in TSP. The objective is to find the shortest distance for the salesman to visit all the cities once. Result obtained from the case study shows that the proposed approach managed to find a better solution compared to the solution suggested by TSPLIB. This study can be extended by tuning the parameters of MOA.

COORDINATES OF THE CASE STUDY, BURMA14

Parameters	Value
Number of agents, h	250
Number of iterations, ω	1000
Number of computations	10
ρ	1
α	1

TABLE IV
RESULT OBTAINED FROM SIMULATION

Criteria	Values
The least iteration while global convergence (iteration)	111.00
The average iteration while global convergence (iteration)	513.40
Length of optimal solution found	31.376
Average fitness after computing 10 computations	32.432

REFERENCES

[1] A. Adam, A. F. Z. Abidin, Z. Ibrahim, A. R. Husain, Z. M. Yusof, I. Ibrahim, "A Particle Swarm Optimization Approach to Robotic Drill Route Optimization," *Fourth Asia International Conference on Mathematical/Analytical Modelling and Computer Simulation*, pp 60-64, 2010.

[2] M. H. Othman, A. F. Z. Abidin, A. Adam, Z. M. Yusof, Z. Ibrahim, S. M. Mustaza, L. Y. Yang, "A Binary Particle Swarm Optimization Approach for Routing in PCB Holes Drilling Process," *First International Conference on Robotic Automation System*, pp. 201-206, 2011.

[3] M. S. Saelal, A. F. Z. Abidin, A. Adam, N. Nordin, K. Khalil, J. A. A. Mukred, Z. M. Yusof, Z. Ibrahim, "An Ant Colony System for Routing in PCB Holes Drilling Process," *The International Symposium on Innovative Management, Information & Production*, pp. 292-297, 2011.

[4] M. N. Ayob, Z. M. Yusof, A. Adam, A. F. Z. Abidin, I. Ibrahim, Z. Ibrahim, S. Sudin, N. S.-Husin, M. K. Hani, "A Particle Swarm Optimization Approach for Routing in VLSI," *Second International Conference on Computational Intelligence, Communication Systems and Networks*, pp. 49-53, 2010.

[5] V. Černý, 'Thermodynamical Approach to the Traveling Salesman Problem: An Efficient Simulation Algorithm', *Journal of Optimization Theory and Applications*, Vol. 45, No. 1, pp.41–51, 1985.

[6] S. Lin, B.W. Kernighan, 'An Effective Heuristic Algorithm for the Traveling-Salesman Problem', Operations Research, vol. 21, no.2 , pp 498-516, 1973.

[7] F. H. Khan, N. Khan, S. Inayatullah, S. T. Nizami, "Solving TSP Problem by Using Genetic Algorithm," *International Journal of Basic & Applied Sciences*, vol. 9. no. 10, pp 79-88.

[8] W. Li, "A Parallel Multi-Start Search Algorithm for Dynamic Traveling Salesman Problem," *10th International Conference on Experimental Algorithm*, pp 65-75, 2011.

[9] M. Dorigo, L. M. Gambardella, "Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem," *Evolutionary Computation, IEEE Transactions*, vol. 1, pp 53-66, 1997.

[10] S. Yadlapalli, S. Rathiman, S. Darbha, "3-Approximation Algorithm for a Two Depot, Heterogenous Traveling Salesman Problem," *Optimization Letter*, 2010.

[11] Z. Pizlo, E. Stefanov, J. Saalweachter, Z. Li,Y. Haxhimusa, W. G. Kropatsch, "Traveling Salesman Problem: A Foveating Pyramid Model," *The Journal of Problem Solving*, vol. 1, no. 1, 2006.

[12] C. S. Helvig, G. Robins, A. Zelikovsky, "The moving-target Traveling Salesman Problem," *Journal of Algorithms*, no. 49, pp 153-174, 2003.

[13] G. Laporte, "The traveling Salesman Problem: An Overview of Exact and Approximate Algorithms," *European Journal of Operational Research*, no. 59, pp 231-247, 1992.

[14] M. H. Tayarani-N, M.R. Akbarzadeh-T, "Magnetic Optimization Algorithms a new synthesis," *IEEE Congress on Evolutionary Computation*, pp. 2659 – 2664, 2008.

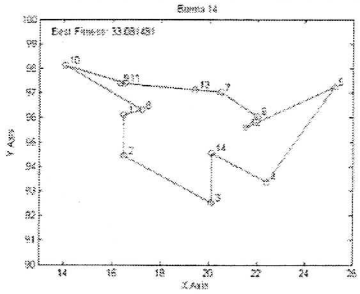
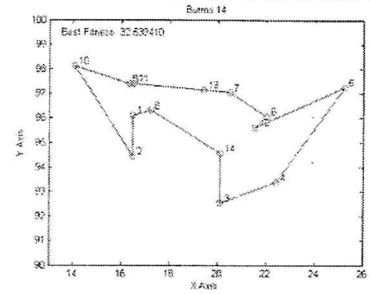
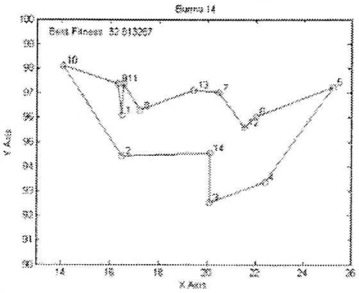
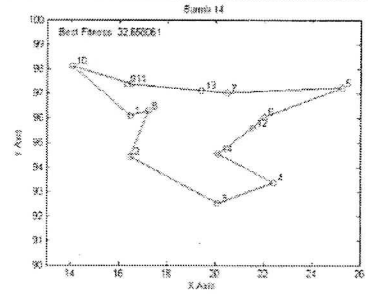
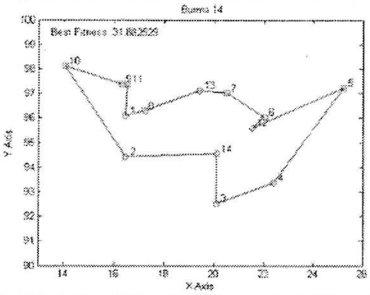
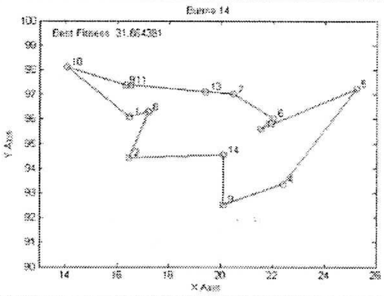
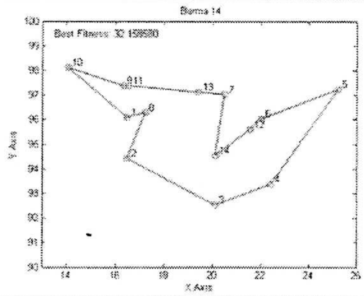
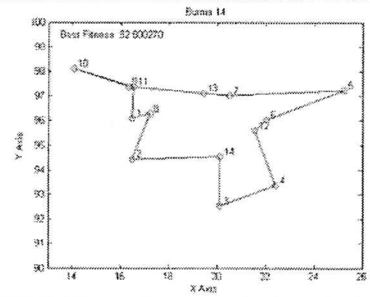
[15] R. Matai, S. Singh, M. L. Mittal. "Traveling Salesman Problem: an Overview of Applications, Formulations, and Solution Approaches," *Traveling Salesman Problem, Theory and Applications*. InTech, 2010.

[16] G. B. Dantzig, D. R. Fulkerson, S. M. Johnson "Solution of a large-scale travelling salesman problem," 1954.

[17] TSLIB, Universität Heidelberg. URL: http://comopt.ifi.uniheidelberg.de/software/TSPLIB95/

TABLE V

IMAGES OF BEST SOLUTION ROUTE FOR ALL SIMULATIONS.

Run 01	Run 02
	
Run 03	Run 04
	
Run 05	Run 06
	
Run 07	Run 08
	
Run 09	Run 10