Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 125 (2018) 68–76

Procedia Computer Science

www.elsevier.com/locate/procedia

6th International Conference on Smart Computing and Communications, ICSCC 2017, 7-8 December 2017, Kurukshetra, India

A Genetic Algorithm Approach to Autonomous Smart Vehicle Parking system

Diya Thomas*[a], Binsu C. Kovoor[b]

[a]*Department of Computer Science and Engineering, RSET, Kochi, India*
[b]*Department of Information Technology, CUSAT, Kochi, India*

Abstract

The shopping malls are source of entertainment and pleasure for the public during weekends. People struggling to park their vehicle in parking bay of shopping mall is a usual scenario witnessed at those peak times. Customer precious time and fuel is wasted and they get only few time for shopping. Authorities find it difficult to cope up with this situation even after appointing more employees to manage the traffic experienced in the bay. A smart car parking system that could elevate this problem is an urgent requirement for the shopping mall. This paper falls light on this issue by proposing a new prototype for the smart vehicle parking system. A genetic algorithm approach has been taken to address the issue of scheduling the vehicle to the parking bay.

Peer-review under responsibility of the scientific committee of the 6th International Conference on Smart Computing and Communications

Keywords: IOT;Smart parking;Autonomous Trolley;Genetic Algorithm; Queuing theory

1. Introduction

With evolution of IoT paradigm, devices and systems are replaced with ´Smart´ devices and systems that could capture, process, analyze and even make intelligent decision with the help of sensors, actuators, data mining and evolutionary techniques. One of the booming IoT application is development of a ´Smart City´. Other applications include but are not limited to smart waste disposal, smart assisted living, smart transportation. This paper focus on smart city application of IoT. As a step towards the development of smart city, a smart parking system is proposed for the shopping mall.

Shopping mall which is the heart and soul of city is one of the focal point of attraction of the public. The parking bay of the shopping mall congested with vehicles causing a big traffic jam is a usual scenario during the weekends.

* Corresponding author. Tel.: +0-000-000-0000 ; fax: +0-000-000-0000.
E-mail address: diyaabs13@gmail.com, diya.arnold@gmail.com

1877-0509 © 2018 The Authors. Published by Elsevier B.V.
Peer-review under responsibility of the scientific committee of the 6th International Conference on Smart Computing and Communications
10.1016/j.procs.2017.12.011

The shopping mall personnel find it difficult to manage the deadlock situation. Due to the inefficiency of the parking system, vehicles are parked even at the roadside nearby the shopping mall. This hectic problem is addressed in few of the national newspaper yearly [1, 2]. But the situation remains the same even after apply different strategies to alleviate the problem. The root cause of the problem analyzed is the poor parking system employed in the shopping mall. When the problem is closely inspected, the reason for the traffic congestion is not because of lack of parking space but due to the lack of efficient scheduling of vehicles to the parking space. To alleviate the problem, the parking system should implement proper scheduling measures to schedule the vehicle effectively to the appropriate parking region without compromising precious waiting time of the customer. Genetic algorithm (G.A) is an evolutionary algorithm well suited for solving such scheduling problems. Genetic algorithm is a very good optimization technique that iterates through different stages like selection, crossover and mutation to obtain an optimal solution.Genetic Algorithm mimics the process of natural selection using bio-inspired operators like crossover and mutation. It is based on the theory of survival of the fittest. Genetic algorithm consider a population of solutions (individuals).The fitness of each solution is calculated by evaluating a fitness function against each solution. The survival of the individual to the next iteration is purely based on the fitness value of the individual. The individuals with least fitness value will be removed from the population. Good characteristics derived from parent solutions are propagated to the next generation by applying crossover and mutation. This paper propose a prototype and derive the essence of genetic algorithm to solve the scheduling issue experienced in the proposed parking system.

The rest of the paper is organized as follows: The prototype is discussed in detail in section II. Mathematical modelling and genetic algorithm approach employed in the system is also discussed in section II. Section III present the experimental set up and expected results of the system. Section V will concludes the paper.

2. Proposed System

2.1. System Overview

As shown in Fig. 1., the proposed system includes an android application for customers to book the parking lot in advance. Each user is given a customer id. Customer id can be used by them for future booking. This is to identify frequent customers to shopping mall. At the time of booking, customers are requested to enter the duration for which they will park the vehicle. When the parking time exceed the duration, the system will send a ´Time Exceeded´ alert message to customers to take appropriate action. They are given a grace period of 30 Minutes to take the vehicle from the bay or if the customer wish he can extend the time duration by paying an extra charge at the exit. PayPal is used as the payment gateway to deal with parking payments. Both the payment details and allotment details are stored in a cloud database. The core part of the proposed system is an autonomous trolley. The trolley is used to take the vehicle from the entrance to correct parking slot identified by the system. The customer can leave their vehicle at the entrance and enjoy the shopping.The trolley capture the image of the surrounding after the vehicle is successfully parked in the slot. The image and location map is send to the customer mobile application. The image act as an acknowledgement for the customer that vehicle is correctly parked and location map help them to track their vehicle. Thus the advantage of using an automated trolley is that customers can enjoy their shopping without wasting their precious time in parking and tracking the vehicle. Ultra-sonic sensors are used to detect the parking and unparking of vehicle in a parking slot. These sensors constantly give parking updates to the system and is an important component in the proposed model.

2.2. Problem Definition

As shown Fig. 1., the parking bay is divided into different regions $R_1, R_2, R_3, \ldots, R_m$. Each region is again divided into parking lots $L_1, \ldots, L_2, .. L_n$ in $R_1, R_2, \ldots, R_m$. An automated trolley is used to carry the vehicle from the entrance to the slot in the appropriate region. ´k´ number of automated trolley T_1 to T_k are assigned to the parking bay to do this task. The count ´k´ is determined by employing a mathematical model discussed in next section.These ´k´ number of trolleys are employed to avoid the waiting queue that may built up at the entrance. The problem is to properly schedule ´k´ trolley to appropriate $1, \ldots, m$ regions so that maximum efficiency score, utilization score and reduced waiting time to park the vehicle is achieved.It can also be redefined as the problem to find the best parking region and parking lot to park the vehicle.

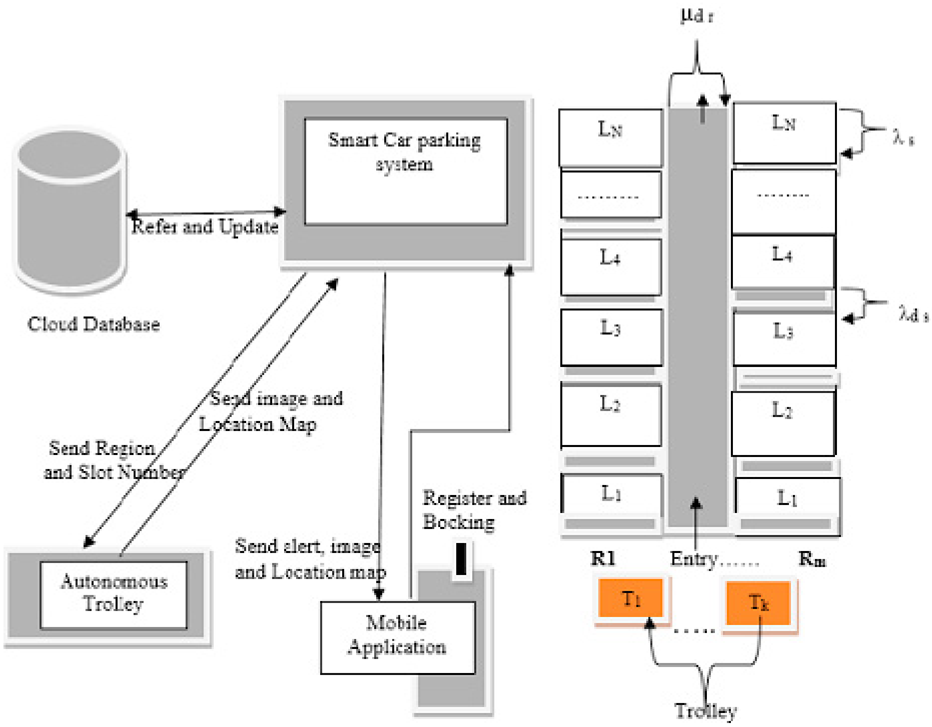

Fig. 1. System Overview and Internal structure

2.3. Mathematical Modelling to calculate ´k´ and Utilization score.

Mathematically modeling is employed to calculate the total number of trolley needed. Average waiting time of vehicle in queue, Efficiency score of the trolley and Utilization score of the parking region are three main parameters which judge the performance of proposed scheme. In the given model, Equation (7), Equation (8) and Equation (11) defines efficiency score, utilization score and average waiting time respectively. Efficiency score is the average number of the vehicles parked successfully by the trolley in a unit time. It is a measure of efficiency of the trolley. Utilization score is average slot allocated in a unit time. It is a measure of how well the region is utilized effectively. Queuing theory is applied for the calculations.

- *Traffic intensity*$[\rho]$

$$\rho = \frac{Rate_{\text{arrv}}}{Rate_{\text{serv}}} \tag{1}$$

where, ρ=traffic intensity,$Rate_{\text{arrv}}$=Arrival rate,$Rate_{\text{serv}}$=Service rate

- *Round trip time of the trolley*$[RTT_t]$

$$RTT_{\text{t}} = \frac{2((N\lambda_{\text{s}} + (N-1)\lambda_{\text{ds}} + \lambda_{\text{dr}})}{v} + delay_{\text{p}} \tag{2}$$

where,delay_{p} =delay in parking vehicle,λ_{ds}=Inter distance between slot s,$(1 \le s \le N)$,λ_{dr} =Inter distance between region r,$(1 \le r \le M)$,λ_{s}=width of slot, N=Maximum Number of parking lot, M=Maximum number of

parking region
Since v is constant, ignoring v in equation 2

$$RTT_{\mathrm{t}} = 2((N\lambda_{\mathrm{s}} + (N-1)\lambda_{\mathrm{ds}} + \lambda_{\mathrm{dr}}) + delay_{\mathrm{p}} \tag{3}$$

- *Average inter arrival time* $[\Delta T]$

$$\Delta T = \sum \frac{(Arrv_{\mathrm{i+1}} - Arrv_{\mathrm{i}})}{x-1} \tag{4}$$

where,$Arrv_{\mathrm{i+1}}$ = Arrival time of i+1$^{\mathrm{th}}$ vehicle ,$(1 \le i \le x-1)$,$Arrv_{\mathrm{i}}$= Arrival time of i$^{\mathrm{th}}$*vehicle*, $(1 \le i \le x-1)$

- *Number of trolley* $[k]$

$$k = \sqrt{\frac{(Prob\,(u)\,u!)}{(e^{-}2)}} \tag{5}$$

where,e=2.718, u=number of arrivals per unit time
$Prob\,(u)$=Probability of number of arrivals u

- *Expected number of vehicles* $[E\,(x)]$

$$E\,(x) = \frac{\left(P\,(0)\left(\frac{\rho^{\mathrm{k+1}}}{kk!}\right)1\right)}{\left(1-\frac{\rho}{k}\right)^{2}} + \rho \tag{6}$$

where,$P\,(0)$=probability that there is no vehicles,ρ=traffic intensity,k=number of trolley

- *Efficiency score of trolley t* $[\eta t_i]$

$$\eta_{\mathrm{ti}} = \frac{\left(E\,(x) - T_{\mathrm{q}}\right)}{k} \tag{7}$$

where,T_{q}=Queue time

- *Utilization score of region r*$[\mu r_i]$

$$\mu r_{\mathrm{i}} = \frac{\left(N - \left(E\,(x) - T_{\mathrm{q}}\right)\right)}{N} \tag{8}$$

where,N = maximum number of regions

- *Expected average number of vehicles in the systems* $\left[E_{avg}\,(n)\right]$

$$E_{\mathrm{avg}}\,(n) = E\,(x) + \rho \tag{9}$$

- *Average total time*$\left[T_{avg}\right]$

$$T_{\mathrm{avg}} = \frac{E_{\mathrm{avg}}\,(n)}{Rate_{\mathrm{arrv}}} \tag{10}$$

- *Expected average waiting time* $\left[WT_{avg}\right]$

$$WT_{\text{avg}} = T_{\text{avg}} - \frac{1}{Rate_{\text{serv}}} \tag{11}$$

2.4. Smart Car Parking Algorithm using Genetic algorithm

Genetic algorithm (G.A) is heuristic search and optimization technique [3, 4, 5, 6, 7, 8] that are well suited for solving the search problems like the one mentioned above. An optimal solution for parking problem can be obtained by applying G.A. Optimal solution is the best parking region and slot to park the vehicle. The scenario considered to apply genetic algorithm is a parking bay with parking regions and the parking slots in the region, which is filled in first come first served order. The problem is to find a best parking region so that efficiency in-terms of utilization score and efficiency score is improved without compromising average waiting time. The mechanism used in each stages of G.A is shown in the table 1 below:

Table 1. Methods Adopted

Stages	Methods
Encoding	Binary
Selection	Roulette wheel selection
Crossover	Three Parent crossover
Mutation	Bit String Mutation
Stopping condition	Maximum iterations

Chromosome structure

The problem is to find out the best parking region.Each parking region is represented as a chromosome. In the chromosome structure as shown in Fig. 2.,each bit in the chromosome represent slot in the parking region. If the slot is occupied the value of the slot is 0 and if it not occupied the value is 1. Each bit is represented by C_i where $1 \leq i \leq N$ gives information about slot occupancy.

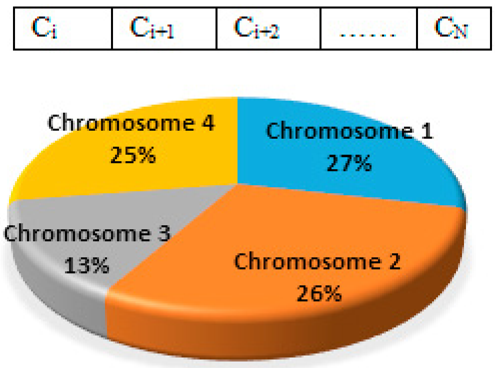

Fig. 2. Chromosome Structure and Roulette wheel selection

Initial Population Generation

The initial population size is set to 4 which is equal to number of parking region. If a parking region consist of 8 slots then the number of bits needed to represent the chromosome is 8. In general of there are ´M´ parking region with ´N´ number of parking slot in each region then our population consist of ´M´ chromosome with chromosome length equal to ´N´.The encoded chromosome structure is shown in Table 2.

Selection

Table 2. Encoding-Initial Population

Chromosome Number	chromosome
1	00011111
2	01111111
3	00000011
4	00001111

The selection operation select the best solution from the population of solution based on the fitness function $F(s)$. The $F(s)$ of solution ´s´ is defined in Equation (12).Roulette wheel selection strategy is employed.

$$F(s) = \frac{\sum \alpha_i C_i}{N} \tag{12}$$

where $\alpha_i = 2^{i+1}$ and $1 \leq i \leq N$

α_i is the weight assigned to each bit in the solution or chromosome As an example, consider the chromosome 3 (00000001) listed in table 3. The fitness F(chromosome 3), probability of selection P(chromosome 3), Percentage probability PP(chromosome 3) and expected count E(chromosome 3) are calculated as follows:

F(00000001)=$2^0 0 + 2^1 0 + 2^2 0 + 2^3 0 + 2^4 0 + 2^5 0 + 2^6 0 + 2^7 1 = \frac{128}{8} = 16$

Probability of selection P(s) is as follows:

$$P(s) = \frac{F(s)_i}{\sum F(s)_i} \tag{13}$$

where $1 \leq i \leq N$

Example:P(00000001) = $\frac{16}{117.75} = 0.13$

Percentage probability PP(s) is as follows:

$$PP(s) = P(s)(100) \tag{14}$$

Example:PP(00000001) = (0.13)100=13%

Expected Count E(s) is defined below:

$$E(s) = \frac{F(s)_i}{Average(F(s)_i)} \tag{15}$$

$$Average(F(s)) = \frac{\sum F(s)_i}{N} \tag{16}$$

where $1 \leq i \leq N$

Example:E(00000001)= $\frac{16}{14.71}$=1.08

Similarly fitness, probability of selection, percentage probability and expected count of the remaining chromosomes are calculated. The obtained results for each chromosome are listed in table 3. The roulette wheel constructed after above PP(s) calculation of all the chromosome is shown in Fig.2. After the roulette wheel selection, the actual count of the chromosomes is also obtained and is listed as the last column in table 3. Actual count of the chromosome will determine the chromosome from the population that will be moved to the mating pool. The actual count of the chromosome ´s´ A(s) equal to ´i´ means that the chromosome ´s´ will appear in the mating pool ´i´ times.
For example A(chromosome 2) is equal to 2 as shown in table 3. So chromosome 2 (01111111) will appear in the mating pool twice. The mating pool constructed based on the actual count of the chromosome is shown in the table 4.

Based on the actual count A(s), chromosome 2 is chosen twice and chromosome 1 and 4 is chosen once to be in the mating pool as shown in table 4. The crossover and mutation operation is applied to these chromosomes in the mating pool to generate offspring. Since actual count of the chromosome 3 (least fit) is zero, it is not taken to the mating pool and will be completely removed from the current population. The crossover and mutation operations are described

Table 3. Selection

Chromosome Number	chromosome	Fitness $[F(s)_i]$	P(s)	PP(s)	E(s)	A(s)
1	00011111	31	0.26	26%	2.10	1
2	01111111	31.75	0.27	27%	2.22	2
3	00000011	16	0.13	13%	1.08	0
4	00001111	30	0.25	25%	2.03	1
Sum $[\sum F(s)_i]$		117.75	0.91	100%	7.43	4
Average $[\frac{(\sum F(s)_i)}{N}]$		14.71	0.22	25%	1.85	1
Maximum		32.75	0.27	27%	2.22	2

Table 4. Crossover

Chromosome number	Mating pool	Chromosome number of the chosen parent	Offspring	F(s)
1	01111111	1 ,2 ,3	01111111	31.7
2	01111111	2 ,3 ,4	00011111	31
3	00011111	1 ,2 ,4	01111111	31.7
4	00001111	1 ,4 ,3	00011111	31
Sum $[\sum F(s)_i]$				125.4

Table 5. Mutation

Chromosome Number	Offspring after crossover	Random Position	Offspring after mutation	F(s)
1	01111111	1	11111111	31.7
2	00011111	3	00111111	31.5
3	01111111	2	00111111	31.5
4	00011111	3	00111111	31.5
Sum $[\sum F(s)_i]$				126.3

Table 6. Next Generation

Chromosome Number	chromosome
1	11111111
2	00111111
3	00111111
4	00111111

below.

Crossover

The cross over operation used to solve the problem is three parent crossover. Bit-wise comparison of the first two parents are done. The bit in each location of offspring is chosen based on the following condition. If the bit in the same location of the first two parent chromosomes are same then that bit will be chosen for offspring. But if the bit differ then the bit specified at the same location in third parent chromosome will be chosen. The crossover probability is 100%.

For Example, consider the chromosome 2 (01111111) in the mating pool as shown in second column of the table 4. As in the third column of the table 4, the parents randomly chosen for crossover of chromosome 2 is chromosome 2, chromosome 3 and chromosome 4. Cross over operation applied on chromosome 2 is shown below. Similarly crossover operation is applied to remaining chromosome in the mating pool. The offspring's obtained after crossover

is listed in the fourth column of the table 4. Fitness value of each offspring is calculated and is listed as fifth column in table 4. The fitness function is calculated using the same procedure described under selection operation.
Example:
Before Crossover
Parent 1(Chromosome2)- 01111111
Parent 2(chromosome 3)- 00011111
Parent 3(chromosome 4)- 00001111
After crossover
Offspring-00011111

Mutation

Mutation is done on the offspring listed in column four of table 4. The bits at the random position are flipped. For example, consider the offspring 1 (01111111) listed in second column of table 5. The random position chosen to flip the bit in offspring 1 is the first position. Mutation probability is 0.12.The bit in first position of offspring 1 was originally 0 and is flipped to 1 after mutation as shown below.
Offspring 1 after crossover: 01111111
Offspring 1 after mutation: 11111111
Remaining offspring's generated after crossover are all mutated like the offspring 1. The mutated offspring's are shown in column 4 of table 5. After crossover and mutation, there is an increase in the total fitness value of the offspring compared to their parents.
The fitness value of the mutated offspring are calculated and is shown as the last column in table 5.These fitness value of the individual chromosome will decide their existence to the next generation of population.The next generation of population obtained is shown in table 6. To obtain an optimal solution, all the stages discussed above is repeated until a stopping condition is met.

Algorithm

Step 1: Encode the solution using binary encoding scheme.
Step 2: Generate population of solution where each individual in the population is a binary encoded solution.
Step 3: Apply Roulette wheel selection to select the best individual to do mating and generate offspring.
  Step 3.1: Calculate the fitness value of each solution using the defined fitness function.
  Step 3.2: Calculate the probability of selection, Probability Percentage and Expected count of the individuals.
  Step 3.3: Find out the actual count after applying Roulette wheel selection.
Step 4: Create the mating pool by choosing the individual based on the actual count.
Step 5: Generate offspring by Applying three parent crossover to the individuals in the mating pool.
Step 6: Apply single point mutation with mutation probability 0.12 to the generated offspring after crossover.
Step 7: Calculate the total and average fitness value of the offspring.
Step 8: Create the next generation of offspring based on the fitness value.
Step 9: Repeat step 3 to Step 7 until maximum iteration is reached.
Step 10: Return the optimal solution.

3. Experimental setup and Expected result

The best parking region is successfully calculated using Genetic Algorithm. Genetic algorithm is implemented in C-programming language. The crossover probability and mutation probability is 100% and 25% respectively. The output of G.A is feed to the proposed system for decision making. Performance analysis is done and is shown graphically. Using the equation (7),equation (8) and equation (11) efficiency score, utilization score and the average waiting time are calculated and are used as performance analysis metrics in the proposed system. The Region utilization graph constructed using utilization score given in equation (8) is shown as in Fig. 3(a).The graph clearly depict that the parking region can be efficiently utilized without any space wastage when optimization algorithm like G.A is employed in smart parking system.Similarly in Fig. 3(b) shows that in smart parking system with G.A ,the trolley can efficiently park the vehicle in the accurate slot even if there is an increase in number of vehicles. But in system without G.A where random allocation of slot is employed, the efficiency of trolley to park the vehicle decline with increase in number of vehicles.This is due to lack of optimization done in choosing an optimal slot.The efficiency graph is plotted using the efficiency score calculated as per the equation (7). The waiting time to park the vehicle is comparatively less in the proposed system by employing multiple autonomous trolley and Genetic algorithm for optimization and decision making. The waiting time graph is calculated based on average waiting time given in equation (11) and is shown in Fig. 3(c).

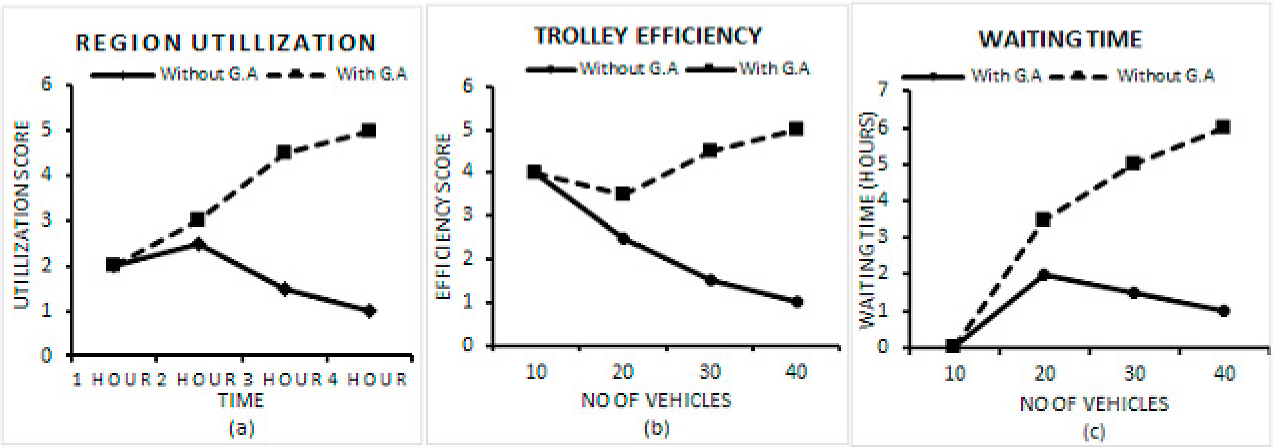

Fig. 3. Performance Analysis

4. Conclusion

The autonomous trolley system and the genetic algorithm implementation to find the best parking slot, make the proposed system efficient in terms of space utilization and trolley efficiency. Genetic algorithm help the proposed system to optimize the searching process for the best region without much time wastage. The Autonomous trolley system reduces the waiting time of the customer in queue. The customer can leave their vehicle at the entrance and the parking will be taken care by the smart trolley. The queuing theory is applied to derive the utilization,efficiency and waiting time factor of the system. The performance analysis graph shows that the utilization and efficiency factor is increased with G.A without compromising the reduced waiting time as compared to a system without G.A. There is no doubt that the proposed system will outperform the existing system in terms of efficiency of parking the vehicle and utilization of parking space . The genetic algorithm approach and application of queuing theory model, helped in alleviating the parking space scarcity and vehicle scheduling problem. Thus the proposed model is a complete smart car parking solution for the parking problem experienced in the shopping mall.

References

[1] Official website of Kochi city police. Available from: http://www.kochicitypolice.org/traffic-police.html ,2003 (accessed 23.02.2016).
[2] Hiran Unnikrishnan,editor. The Hindu Newspaper Article. Available from:http://www.thehindu.com/news/national/kerala/no-space-to-park-in-kochi/article7034337.ece
[3] Soh Chun Khang , Teoh Jie Hong . Wireless Mobile-based Shopping Mall Car Parking System. In: IEEE Asia-Pacific Services Computing;2010.
[4] G. Zhang,L. Gao, Y. Shi.An effective genetic algorithm for theflexible job-shop scheduling problem.In:Expert Systems with Applications;2011;38:3563-3573.
[5] J. C. Chen. Flexible job shop scheduling with parallel machines using Genetic Algorithm and Grouping Genetic Algorithm.In: Expert Systems with Applications; 2012.
[6] Y. Li,Y. Chen. A Genetic Algorithm for Job-Shop Scheduling.In:Journal of Software;2010;5:269274.
[7] Charles C. Peck, Atam P. Dhawan.Genetic algorithms as global random search methods: An alternative perspective.In:Evolutionary Computation;1995.
[8] Pratibha Bajpai, Manoj Kumar. Genetic Algorithm an Approach to Solve Global Optimization Problems.In: Indian Journal of Computer Science and Engineering;2010;1:3.