World Academy of Science, Engineering and Technology
International Journal of Computer and Systems Engineering
Vol:18, No:1, 2024

UPPAAL-Based Design and Analysis of Intelligent Parking System

Abobaker M. Q. Farhan, Olof M. A. Saif

Abstract—The demand for parking spaces in urban areas, particularly in developing countries, has led to a significant issue in the absence of sufficient parking spaces in crowded areas, which results in daily traffic congestion as drivers search for parking. This not only affects the appearance of the city but also has indirect impacts on the economy, society, and environment. In response to these challenges, researchers from various countries have sought technical and intelligent solutions to mitigate the problem through the development of smart parking systems. This paper aims to analyze and design three models of parking lots, with a focus on parking time and security. The study used computer software and Uppaal tools to simulate the models and determine the best among them. The results and suggestions provided in the paper aim to reduce the parking problems and improve the overall efficiency and safety of the parking process. The conclusion of the study highlights the importance of utilizing advanced technology to address the pressing issue of insufficient parking spaces in urban areas.

Keywords—Preliminaries, system requirements, timed automata, uppaal.

I. Introduction

THE parking lots play a crucial role in the smooth functioning of traffic and mobility in urban areas. With vehicles spending the majority of their time parked, it is imperative that these parking spaces are designed and managed effectively. These parking lots are established in various areas, including commercial, residential, industrial, and recreational zones.

The exponential growth of the automobile industry, as stated by the International Organization of Automobile Manufacturers [1], with a projected increase of 1% in 2018 to 98 million units, highlights the need for efficient parking solutions.

In this study, we focus on designing and modeling three different parking lot configurations along streets and under buildings. The models aim to accommodate a larger number of vehicles while utilizing limited space effectively. The simulations of the models were carried out using the Uppaal tool, resulting in satisfactory outcomes.

The objective of this paper is to contribute to the development of innovative parking solutions that address the growing demand for parking spaces while ensuring efficient utilization of space and traffic flow.

II. Preliminaries

Timed automata [2], [3] is a mathematical model used to represent and analyze real-time systems. Real-time systems are systems that operate based on the progression of time. The model is derived from finite automata, which are finite state machines, augmented with clocks. The clocks allow for the measurement of the progression of time and are defined in each automaton, with all clocks progressing in synchrony.

In timed automata, clock constraints play a critical role in defining the relationships between the clocks and enforcing timing restrictions in real-time systems. Clock constraints are specified as copulative formulas for atomic restrictions. If a clock constraint is not satisfied during a transition, the transition can only be executed when the constraint is satisfied. There are two types of transitions in timed automata: postponed transitions, executed at the edge of the automaton, and immediate transitions, executed as soon as the constraint is satisfied.

This study focuses on the application of timed automata and clock constraints in the design and analysis of real-time systems. The aim is to provide a comprehensive overview of the theory and practical implications of this mathematical model, with a focus on its use in the verification and validation of real-time systems. The findings of this study can provide valuable insights for researchers, engineers, and practitioners in the field of real-time systems and their design and analysis.

To further understand the concept of clock constraints, we define timed automata as a tuple of symbols N, l_0, E, I that consists of a limited set of control nodes (N), the primary (initial) control node ($\mathrm{L}_0 \in \mathrm{N}$), the edge set (E $\mathrm{N} \times \mathrm{B}(C) \times \Sigma \times 2^C \times \mathrm{N}$), and the invariants for each control node ($\mathrm{I} : \mathrm{N} \rightarrow \mathrm{B}(C)$), where Σ represents real-valued variables for actions and B represents real-valued variables for clocks.

Uppaal [4] is a widely used tool for modeling, simulating, and verifying real-time systems. Developed in collaboration between the University of Uppsala and Aalborg in Denmark and Sweden, Uppaal is free for academic use only and its current version 4.1.19 can be downloaded from ”http://www.uppaal.com”. Uppaal is divided into three main components: the graphical user interface, the verification server, and the stand-alone verifier. The graphical user interface is used for modeling, simulation, and verification, the verification server is used to compute successor states, and the stand-alone verifier is used for verification purposes.

III. System Requirements

The objective of our research is to develop models for secure and efficient vehicle parking lots that minimize the usage of space and time. In this paper, we present three distinct designs for parking lots, each with its own unique features.

The first design has a length of 50 meters, a width of 18 meters, a parking angle of 70 degrees, and 32 parking spots.

Abobaker M. Q. Farhan is with School of Information and Software Engineering, University of Electronic Science and Technology of, Chengdu, China (corresponding author, e-mail: sam.sldm@yahoo.com).

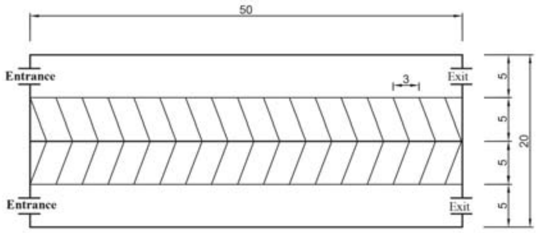

Fig. 1 First design

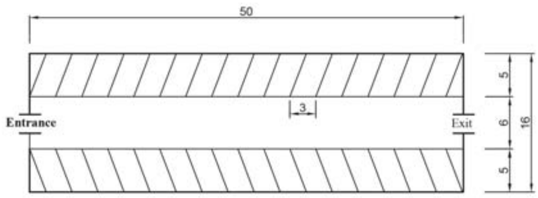

Fig. 2 Second design

It is equipped with two entrance gates and two exit gates, enabling vehicles to enter and exit in a timely manner with minimal time spent in parking, as depicted in Fig. 1.

The second design has a length of 50 meters, a width of 16 meters, a parking angle of 70 degrees, and 32 parking spots. Unlike the first design, it has only one entrance gate and one exit gate, leading to longer parking and departure times, as illustrated in Fig. 2.

The third design, with a length of 51 meters, a width of 16 meters, a parking angle of 90 degrees, and 34 parking spots, offers more parking spots than the first two designs, but requires more time for parking, as depicted in Fig. 3.

IV. Time Automata in Uppaal

A. Timed Automata for Three Designs

In our system, there are three different designs for the parking lot, each with a unique combination of vehicles and parking spaces. The first and second designs comprise 50 vehicles and 32 parking spaces, whereas the third design comprises 34 parking spaces. The system is designed to park 32 vehicles initially and then manage any subsequent vehicles in a queue until a parking space becomes available. This requires a sensor to track the number of vehicles entering the parking lot and manage the queue line efficiently.

The concept of invariant is associated with the expressions that remain unchanged over time. The checker restricts the expressions that are allowed as invariants and requires that these invariants be related to conditions dependent on the clocks. The clocks values must be an integer and these invariants can be modified by changing the 'urgent' channels and 'broadcast' channels, requiring synchronization with the clocks.

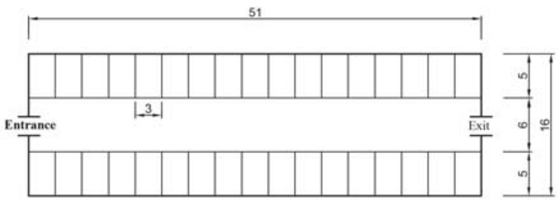

Fig. 3 Third design

The exponential rate is a ratio expression that is either an integer or two integers in the form r:q, where r/q defines the exponential probability distribution. This is used in statistical model checking, for example, the likelihood of leaving a location over time depends on the exponential distribution, shown as: $\Pr(t) = 1 - e^{-\alpha t}$ where t is time and α is a fixed rate, with e equal to 2.718281828.

1) Timed Automata for First Design: The timed automata for the first parking lot is shown in Fig. 4 and the timed automata for the lane parking spots is shown in Fig. 5. The design has 10 exponential rates in the stopping location, with a parking edge of 70 degrees, making it secure and requiring minimal time for parking.
2) Timed Automata for Second Design: The timed automata for the second parking lot is shown in Fig. 6 and the timed automata for the lane parking spots is shown in Fig. 7. The design has 5 exponential rates in the location find_parking_spot, with a parking edge of 70 degrees, requiring lesser time compared to the first design.
3) Timed Automata for Third Design: The timed automata for the third parking lot is shown in Fig. 8 and the timed automata for the lane parking spots is shown in Fig. 9. The design has 1 exponential rate in the location stopping, with a parking edge of 90 degrees, making it the most time-intensive design among the three.

B. Properties

The interaction of human life with the system and its ability to serve us can lead us to classify it as a synchronous system. These systems serve people in various forms such as computer systems, telecommunications, energy technology, and everything that is related to our daily life, as is the case in our research on a parking lot. We have designed and implemented three different models for the parking lot, each model has its unique features while also incorporating some shared technologies, such as sensors, that bring benefits to the community.

In our study, it is imperative to evaluate certain crucial properties, such as deadlock, reachability, activity, and concurrency, to meet the requirements specified in Section III, as follows:

- Deadlock: To ensure that there will be no deadlock in the system, i.e., the system must be deadlock-free.
- Reachability: To enable easy entry and exit from the parking lot.
- Activity: To ensure that only one vehicle can occupy a parking spot at a time, i.e., two vehicles cannot occupy the same spot concurrently.
- Concurrency: To permit multiple vehicles to occupy the parking lot simultaneously, such as determining the

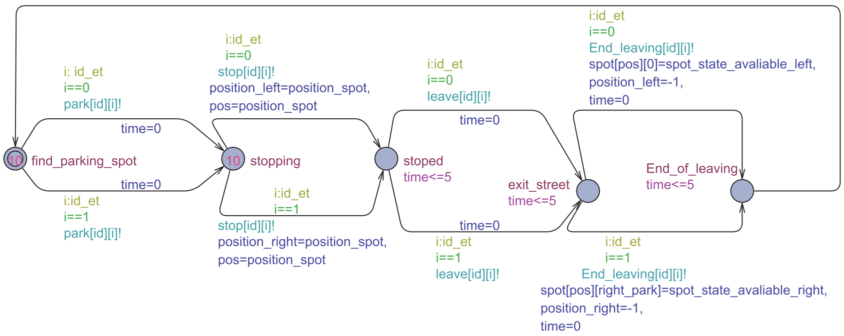

Fig. 4 Timed automaton of the vehicle for the first design

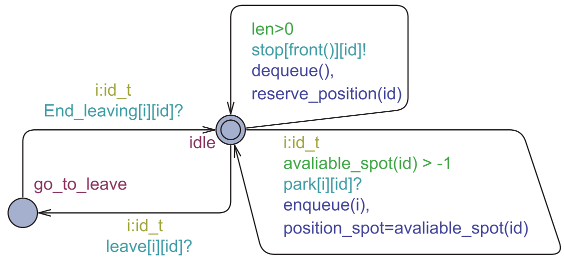

Fig. 5 Timed automaton of the lane for the first design

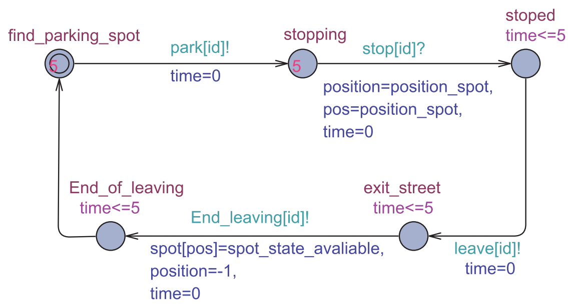

Fig. 6 Timed automaton of the vehicle for the second design

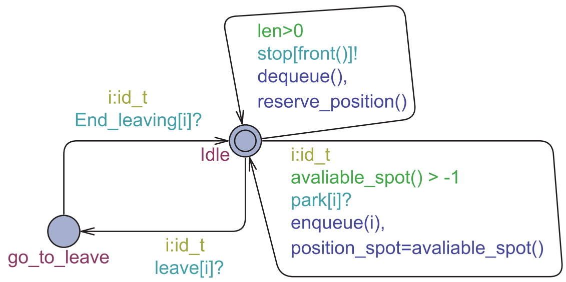

Fig. 7 Timed automaton of the lane for the second design

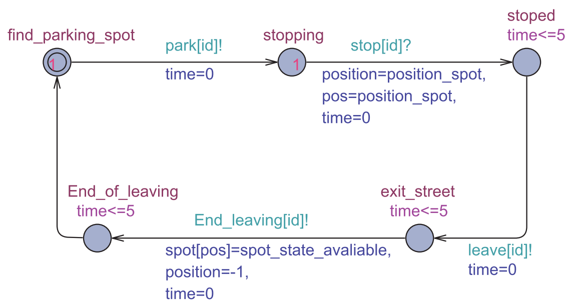

Fig. 8 Timed automaton of the vehicle for the third design

percentage of maximum vehicles stopped in the first 100 runs.

```
a) Deadlock: E<> not deadlock

b) Reachability:
Pr[#<=100] ( <> forall (i:id_t)
car(i).find_parking_spot
and
car(i).exit_street)

c) Activity:
A[] forall (i:id_t) forall (j:id_t)
car(i).stoped
&&
car(j).stoped imply i==j

d) Concurrency:
E[ <= 100 ; 1000]
(max: sum(i:id_t) car(i).stoped)
```

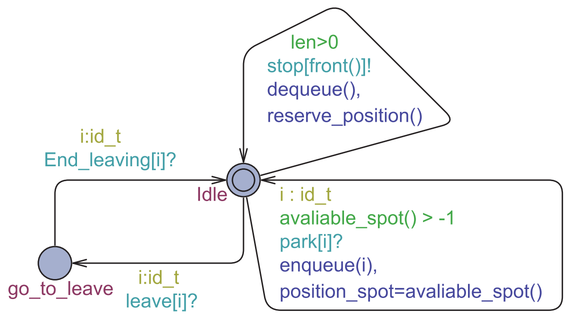

Fig. 9 Timed automaton of the lane for the third design

In our study, we used statistical model checking with Uppaal [5] to verify properties of a system modeled as timed automata [6] communicating via broadcast channels, shared variables, and urgent variables. Three designs were analyzed, and their properties were verified using the Uppaal SMC tool. The results of the verifications are summarized below.

1) First design: The first design was verified for four

properties: (a) deadlock-free system, (b) reachability with easy enter and departure from the park, (c) the ability to park more cars in the same position, and (d) concurrency.
(a) The property of a deadlock-free system was satisfied in the first design.
(b) The property of reachability was also satisfied in the first design, with the rustle of 36 runs during Pr(<>) equal [0,0.0973938], and a confidence level of 0.95.
(c) The property of parking more cars in the same position was not satisfied in the first design.
(d) The property of concurrency was satisfied in the first design, with an average of the highest number of vehicles stopped within the first ten time units being 5.871, based on 1000 runs.

2) Second design: The second design was verified for the same four properties as the first design.
(a) The property of a deadlock-free system was satisfied in the second design.
(b) The property of reachability was also satisfied in the second design, with the rustle of 36 runs during Pr(<>) equal [0,0.0973938], and a confidence level of 0.95.
(c) The property of parking more cars in the same position was not satisfied in the second design.
(d) The property of concurrency was satisfied in the second design, with an average of the highest number of vehicles stopped within the first ten time units being 5.494, based on 1000 runs.

3) Third design: The third design was verified for the same four properties as the first and second designs.
(a) The property of a deadlock-free system was satisfied in the third design.
(b) The property of reachability was also satisfied in the third design, with the rustle of 36 runs during Pr(<>) equal [0,0.0973938], and a confidence level of 0.95.
(c) The property of parking more cars in the same position was not satisfied in the third design.
(d) The property of concurrency was satisfied in the third design, with an average of the highest number of vehicles stopped within the first ten time units being 4.312, based on 1000 runs.

Uppaal Statistical Model Checking (SMC) is a powerful tool for monitoring system processes and evaluating their compliance with specified properties. As described in [5], SMC utilizes a series of techniques to produce statistical results, providing a comprehensive and detailed report on the validity of a design. With its practical applications in industrial and software engineering, and its easy-to-understand and implement nature, SMC has become a widely adopted method for real-time modeling of systems.

In our research, we utilized the concept of timed automata [6], which communicate through various means such as broadcast channels, shared variables, and urgent variables, to construct networks of timed automata with different clock rates and time-invariant features across different locations.

To better analyze the results of our simulations, we employed two graphical representations of the data, namely the frequency histogram and cumulative probability distribution. The frequency histogram, as the name suggests, displays the number of runs at specific moments during the simulation period, while the cumulative probability distribution, a histogram created by adding the data sets of all histograms, is sorted by the total number of runs. Both of these visualizations provide valuable insights into the behavior of the modeled system and aid in the evaluation of its validity.

```
E[ <= 100 ; 1000]
(max: sum(i:id_t) car(i).stoped)
```

1) First design: The first design employed a sample size of 1000 runs, with an expected confidence interval of 5.871 ± 0.0208111. Further details reveal the following parameters: a = 0.05, $\varepsilon = 0.05$, bucket width = 1, bucket count = 2, with 1000 runs (100%) displayed and 0 runs (0%) remaining. The sample span was [5, 6] with a mean of $5.871 \pm 0.0208111350947726$ (95% CI). The cumulative probability distribution and frequency histogram of the sample are depicted in Figs. 10 and 11 respectively.
It is worth mentioning that the use of a sample size of 1000 runs, as in the previous designs, greatly enhances the robustness of the results. The narrow confidence interval and well-defined mean reinforce the validity of the findings. Moreover, the availability of detailed views provides further insight into the parameters and distributions of the sample, adding to the credibility of the results.

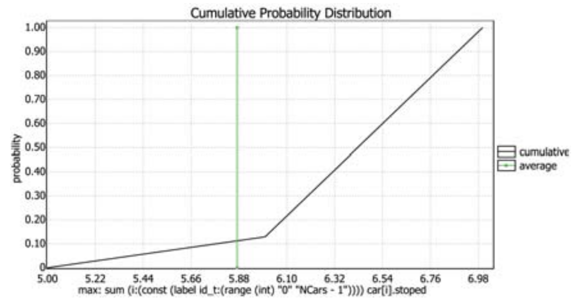

Fig. 10 Cumulative Probability Distribution of the maximum number of the vehicles stayed within 100-time units in the first design

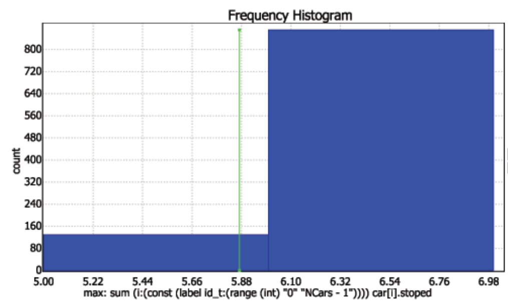

Fig. 11 Frequency histogram of the maximum number of the vehicles stayed within 100-time units in the first design

2) Second design: The second design also utilized a sample size of 1000 runs, with an expected confidence interval

of 5.481 ± 0.0604384. Similar to the previous design, the results display the following parameters: a = 0.05, ε = 0.05, bucket width = 1, bucket count = 6, with 1000 runs (100%) displayed and 0 runs (0%) remaining. The sample span was [1, 6] with a mean of 5.481 ± 0.0604384473199455 (95% CI). The cumulative probability distribution and frequency histogram of the sample are illustrated in Figs. 12 and 13 respectively.

It is noteworthy that the use of a sample size of 1000 runs again highlights the robustness of the results. The narrow confidence interval and the well-defined mean further demonstrate the validity of the results. Additionally, the availability of detailed views allows for further analysis of the parameters and distributions of the sample, adding to the credibility of the findings.

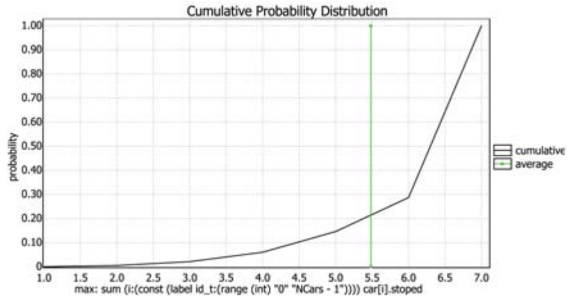

Fig. 12 Cumulative Probability Distribution of the maximum number of the vehicles stayed within 100-time units in the second design

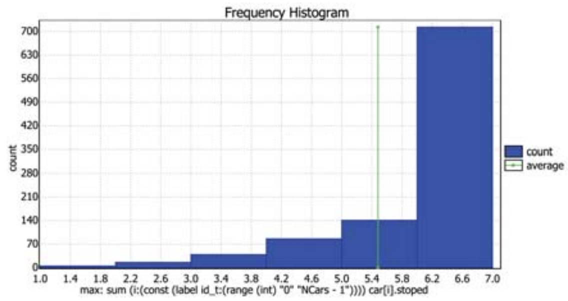

Fig. 13 Frequency histogram of the maximum number of the vehicles stayed within 100-time units in the second design

3) Third design: The third design utilized a sample size of 1000 runs, with an expected confidence interval of 4.332 ± 0.0796968. The results display the following parameters: a = 0.05, ε = 0.05, bucket width = 1, bucket count = 6, and a total of 1000 runs (100%) were displayed, with 0 runs (0%) remaining. The sample span was [1, 6] with a mean of 4.332 ± 0.0796968070505537 (95% CI). The cumulative probability distribution and frequency histogram of the sample are illustrated in Fig. 14 and Fig. 15 respectively.

It is important to note that the explicit requirement of 1000 runs greatly contributes to the robustness of the results. This can be seen in the confidence interval, which is narrow and clearly defined, indicating a high degree of certainty in the results. Furthermore, the detailed views provide additional insight into the parameters and distributions of the sample, further strengthening the validity of the results.

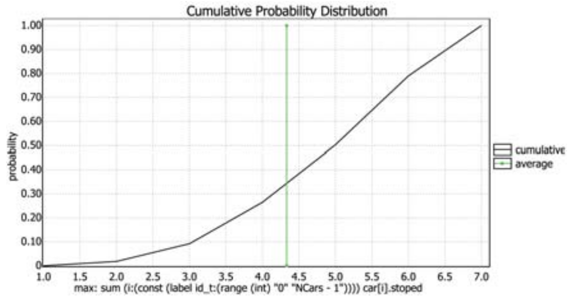

Fig. 14 Cumulative Probability Distribution of the maximum number of the vehicles stayed within 100-time units in the third design

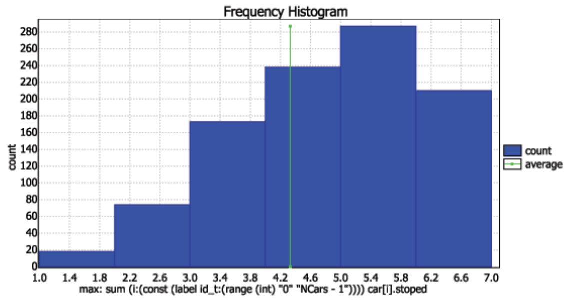

Fig. 15 Frequency histogram of the maximum number of the vehicles stayed within 100-time units in the third design

V. Related Work

In this section, a consensus among researchers has been established that the arrangement of vehicular traffic on the roads, along with the provision of parking facilities for vehicles, is crucial for the improvement of cities. Kamble et al. [7] proposed the design of a smart parking system that operates through an Android application. This application is capable of monitoring the parking status of vehicles in a designated area by determining whether they are parked or not. The system operates through online bookings, whereby spaces are reserved for a specified period, after which the reservation is terminated either through manual intervention or automatically.

According to Kamble et al. [7], the system offers several advantages, such as the ability of users to cancel their reservations at any time and the availability of two types of bookings: parking time bookings and pre-bookings for one day or a week at a specific time on a particular day. However, Kamble et al. [7] also identified a limitation in the system. When the parking period exceeds the specified time, and the vehicle does not leave the parking lot, the system may incorrectly record the parking space as empty. This problem occurs when a user had made a reservation before the specified period, but failed to retrieve their vehicle before the reservation period ended. This highlights the need for further improvements to the system to address this issue.

Chaudhari et al. [8] present a smart parking system that utilizes infrared LM358N and ESP8266 NodeMCU sensors.

The sensors are installed in parking lots to provide real-time information on the availability of parking spaces. However, the implementation of the system is hindered by its high cost, which is attributed to the large number of sensors required, as well as the need for electricity and internet connectivity. Additionally, some countries may face challenges in implementing the system due to factors such as congested roads and unregulated cities, as was the case in India as noted by Chaudhari et al. [8].

In a similar study, Mishra et al. [9] propose a smart parking system that uses infrared devices and a display screen to monitor the availability of parking spaces. The system allows drivers to reserve a parking space either in-person or through a pre-booking system on the website. Unlike Chaudhari et al.'s system, Mishra et al.'s system requires fewer sensors, making it more cost-effective and time-efficient. The system is also similar to those proposed by Kamble et al. [7] and Chaudhari et al. [8].

Kharde et al. [10] present a smart parking system that uses wireless technology, such as GPS and RFID devices, to guide drivers to available parking spaces. The system sends the location of available parking spaces to drivers via SMS messages. However, the implementation of this system may incur a higher financial cost due to the need for RFID devices at each parking space. Despite this, the system offers the advantage of providing drivers with quick access to available parking spaces without the need to search for an open spot.

In [11], Shaikh et al. propose a smart parking system that uses LED sensors and robots to guide drivers to available parking spaces in a multi-level parking lot. The system uses infrared technology and power-saving sensors to regulate the parking lot. The system allows drivers to reserve a parking space through a computer or mobile device. This system is similar in design to Kharde et al.'s system in [10] and the technology used is similar to that of Kamble et al. [7], Chaudhari et al. [8], and Mishra et al. [9].

In [12], Gandhi presents a method for guiding drivers to available parking spaces in a parking lot using sensors and LED devices. The system is designed to choose the shortest path to the available parking space and save time and fuel. However, the system has some drawbacks, including the inability to distinguish between reserved and vacant parking spaces and the need for manual registration of vehicle numbers at the entrance and exit gates. Gandhi suggests the use of various algorithms, such as Bellmans Ford, Dijkstras, Floyd-Warshalls, and A * Algorithms, to optimize the shortest path to available parking spaces.

In our study, we propose a smart parking system that utilizes sensors at the entrance of a parking lot to monitor the flow of vehicles entering and leaving the parking lot. Additionally, we use queue techniques to manage the parking lot when it is full. Our study provides a detailed analysis of the design and performance of three different models of smart parking systems, examining factors.

VI. Conclusion and Future Work

In conclusion, our paper has presented a comprehensive analysis and verification of three real-time parking lot models using the timed automaton and Uppaal model checker. The results indicate that the first design is more efficient than the second and third designs, in terms of service, cost, and security. Our research has met its objectives in providing efficient and secure parking solutions for a larger number of vehicles.

Moving forward, we aim to expand our research by incorporating additional types of parking facilities, including disabled parking, electronic parking, and automatic parking. This will further enhance the functionality and efficiency of our parking systems, and provide an even more comprehensive solution for the needs of society. Our ultimate goal is to contribute to the development of intelligent and sustainable parking systems that can benefit both individuals and society as a whole.

References

[1] http://www.oica.net/.
[2] R. Alur and D. L. Dill, "A theory of timed automata," *Theoretical computer science*, vol. 126, no. 2, pp. 183–235, 1994.
[3] J. Bengtsson and W. Yi, "Timed automata: Semantics, algorithms and tools," in *Advanced Course on Petri Nets*. Springer, 2003, pp. 87–124.
[4] R. Alur and D. L. Dill, "A theory of timed automata," *Theoretical computer science*, vol. 126, no. 2, pp. 183–235, 1994.
[5] S. K. Jha, E. M. Clarke, C. J. Langmead, A. Legay, A. Platzer, and P. Zuliani, "A bayesian approach to model checking biological systems," in *International Conference on Computational Methods in Systems Biology*. Springer, 2009, pp. 218–234.
[6] P. Bulychev, A. David, K. G. Larsen, M. Mikučionis, D. B. Poulsen, A. Legay, and Z. Wang, "Uppaal-smc: Statistical model checking for priced timed automata," *arXiv preprint arXiv:1207.1272*, pp. 1–16, 2012.
[7] P. Kamble, S. Chandgude, K. Deshpande, C. Kumari, and K. Gaikwad, "Smart parking system," *International Journal of Advance Research and Development*, vol. 3, no. 4, pp. 183–186, 2018.
[8] P. Chaudhari, R. Kumar, R. Mistra, and P. Jorvekar, "Smart parking system," *International Research Journal of Engineering and Technology (IRJET)*, vol. 05, no. 01, pp. 637–639, 2018.
[9] B. Mishra, A. Verma, A. Gupta, and M. S. Singh, "Smart parking system," *International Research Journal of Engineering and Technology (IRJET)*, vol. 05, no. 04, pp. 09–13, 2018.
[10] P. Kharde, S. Pal, and S. Kawle, "Smart parking system," *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, vol. 3, no. 1, pp. 639–641, 2018.
[11] F. I. Shaikh, P. N. Jadhav, S. P. Bandarkar, O. P. Kulkarni, and N. B. Shardoor, "Smart parking system based on embedded system and sensor network," *International Journal of Computer Applications*, vol. 140, no. 12, pp. 45–51, 2016.
[12] S. A. Gandhi, "Smart parking system," *ASIAN JOURNAL FOR CONVERGENCE IN TECHNOLOGY (AJCT)-UGC LISTED*, vol. 4, no. I, 2018.