IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received April 1, 2019, accepted April 25, 2019, date of publication April 29, 2019, date of current version May 21, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2913784

Formal Analysis of Human-Assisted Smart City Emergency Services

NAZEERUDDIN MOHAMMAD, (Member, IEEE), SHAHABUDDIN MUHAMMAD, ABUL BASHAR, AND MAJID A. KHAN, (Member, IEEE)
College of Computer Engineering and Science, Prince Mohammad Bin Fahd University, Alkhobar 31952, Saudi Arabia

Corresponding author: Nazeeruddin Mohammad (nmohammad@pmu.edu.sa)

ABSTRACT The concept of IoT-based *Smart Cities* has gained momentum in recent years. The research in this domain has focused on modeling key characteristics of future smart cities along with exploring their design and implementation aspects from multiple perspectives. There is, however, a lack of research effort to provide a holistic approach towards modeling smart city services. A comprehensive approach has to view the smart city as a dynamic, uncertain and complex environment where multiple events with varying severity occur in a continuous and non-deterministic manner. These events have to be handled efficiently with the available resources. In this paper, we present a holistic approach to model the probabilistic as well as non-deterministic aspects of the emergency management services of a smart city by using probabilistic model checking. Our proposed model captures the emergency events of varying severity occurring at several locations in a continuous and non-deterministic manner. These events are responded to by a smart emergency management unit (SEMU), which dispatches the required emergency response units (ERUs) to the event location. In addition to modeling a completely automated system, we introduce a human-assisted decision-making process in the model to reduce false alarms. The proposed model can be used to study and evaluate key parameters in designing a smart city emergency management system to meet given service level agreements. We have implemented our proposed model using the *PRISM* model checker and have conducted several case studies to highlight the effectiveness of our proposed approach for different scenarios of varying complexities.

INDEX TERMS Smart city emergency services, PRISM, probabilistic model checking.

I. INTRODUCTION

Growing urbanization is placing huge burden on city infrastructure that makes it a challenging task to optimally utilize the available resources to provide basic services to its citizens. The rapid growth of cities is causing several management issues including traffic congestion, rapid outbreak of contagious disease (epidemics), air pollution, waste management and public safety in case of catastrophic events. *Smart City* is a recent initiative that uses computing technologies to assist manage large cities in an efficient manner. Several definitions of smart city exist in the literature trying to encompass its scope from technology, people and institutional perspectives [1]. The International Telecommunication Union (ITU) has defined smart city as "A smart sustainable city is an innovative city that uses information and communication technologies (ICTs) and other means to improve quality of life, efficiency of urban operation and services, and competitiveness, while ensuring that it meets the needs of present and future generations with respect to economic, social and environmental aspects" [2]. A smart city concept consists of the following main components:

- Technology: The collection of information and communications technologies which enable the sensing and collection of data, transmission of the data through the network and developing applications which utilize the data to extract meaningful information for making smart decisions [3].
- Human: Human capital is an indispensable component of a smart city through which there is an element of learning and attainment of progressive knowledge for the betterment of the society as a whole. Social interactions and mutual collaboration for creativity and novel solutions are proposed and implemented by the people living in the city [1].

The associate editor coordinating the review of this manuscript and approving it for publication was Bora Onat.

- Institution: The institutional governance has a major role to play in order to implement policies, procedures, and laws for the safety, well-being and progress of the people in the smart city, with a focus on optimal utilization of existing infrastructure and services. Institutions also help plan and project future needs and requirements to maintain the smart city objectives [4].

It has been reported that people prefer to live in cities which have low crime rates, and have high security from natural disasters (such as floods, earthquakes and forest fires) and man-made system failures (such as theft/robbery, fire and electricity outages) [5]. It has been found that there is a wide variety of service domains which come under the purview of smart city namely, economy, natural resources, energy, transportation, buildings, living conditions and governance [4]. One of the subdomains under the living conditions domain is to provide public safety in case of emergencies. These emergencies cause a loss of life, economic hardships and mental agony to people. In fact, the IBM global business services executive report states that improving public safety and emergency response time as one of the high-impact areas for smart cities development [6]. The smart city governance therefore needs to predict these emergencies and prevent them as early as possible. However, in the event of such inevitable emergencies, there must be mechanisms and systems which are "smart" enough to deal with them in the best possible manner. One of the promising technologies for public security implementation and monitoring in the context of smart city is the Internet of Things (IoT) [7].

Any initiative based on IoT requires suitable mechanisms to manage its operations. Efficient implementation of public safety systems and procedures requires the accurate and timely information of events, which can be gathered from safety devices (e.g. monitoring cameras and variety of sensors) interconnected through wireless communication technologies (e.g. wireless broadband), as part of the IoT infrastructure. For instance, a fire mishap in a multi-storey building can be sensed through fire/smoke sensors (part of a WSN) and the information about the time, location and event severity can be communicated to the nearest fire-station. This information will be helpful in dispatching appropriate fire-fighting resources to the fire location for immediate rescue operation [5]. However, it must be noted that despite the benefits achieved from an automated monitoring system based on IoT, one must be cautious regarding the possibility of false alarms coming from the sensors. A false alarm is informally defined as a signal that occurs needlessly [8]. For example, a smoke detector can incorrectly identify a human smoking as a fire incident. Sometimes sensors can malfunction due to environmental conditions. False alarms can also be caused with malicious intent. The number of false alarms dealt by emergency response system is generally very high. For example, 56% of the reported fires between 2011-2012 in UK were false alarms [9]. In Salt Lake City, USA, 90-99% of crime related calls received by police department were false alarms [10]. Significant economic losses have been reported in the literature by initiating rescue operations based on false alarms [11]. Several techniques have been reported to handle the false alarms, including the use of multi sensor technologies and manual vetting process conducted by private security company personnel. The human-in-the-loop (security personnel) approach is found to be quite useful in minimizing false alarms and reducing the financial cost of dealing with them [10].

The automated governance mechanism of IoT systems can be assisted by human decision making during its operations. Human operators can use personal expertise and context-specific knowledge to make better judgments in complex situations. It has been noted that humans in the loop can improve the system performance [12], but modeling and incorporating the human behavior in an IoT system is a challenging task. The work presented in [13] identifies several key challenges encountered in incorporating "humans in the loop" for any cyber-physical system. These challenges include:

- Identifying the type and level of human involvement
- Modeling of human behavior in the system
- Incorporating the human model in the system

An IoT based autonomous system can have different degrees of human involvement. In this paper, we focus on systems which are autonomous but can improve the overall performance of the system by human assisted decision making at certain points during their operations. The proper modeling of human physiological, psychological and environmental aspect is also a key challenge. The human decision making is affected by several factors including their work load, fatigue and technical competence.

Figure 1 shows an example scenario where emergency events in a city are managed by a Smart Emergency Management Unit (SEMU). The SEMU authorizes and controls several Emergency Response Units (ERUs) like ambulances, fire engines and police cars. Several types of events can occur in a non-deterministic manner throughout the city (such as accidents, fires, and robberies) and these events can be detected by IoT based mechanisms with possible human assistance and reported to SEMU. The likelihood and severity of these events may vary depending upon several factors such as the location, time of the year and occurrence of large scale events (such as, earthquake, political rallies, etc). The severity of events can be categorized into multiple levels depending upon the potential loss of lives, property and scale of inconvenience. For example, a road accident without any injuries or road blocks can be considered a minor event while a road accident with multiple injuries or traffic blockage can be considered a major event. It is presumed that a human assistance would be required to augment the IoT system to correctly categorize the severity of an event. The accessibility to the location of an event also varies based on factors such as traffic congestion at certain time periods and the type of roads leading to the event location. The likelihood of a certain type of event, its severity and accessibility to the event location can vary from city to city. The smart city

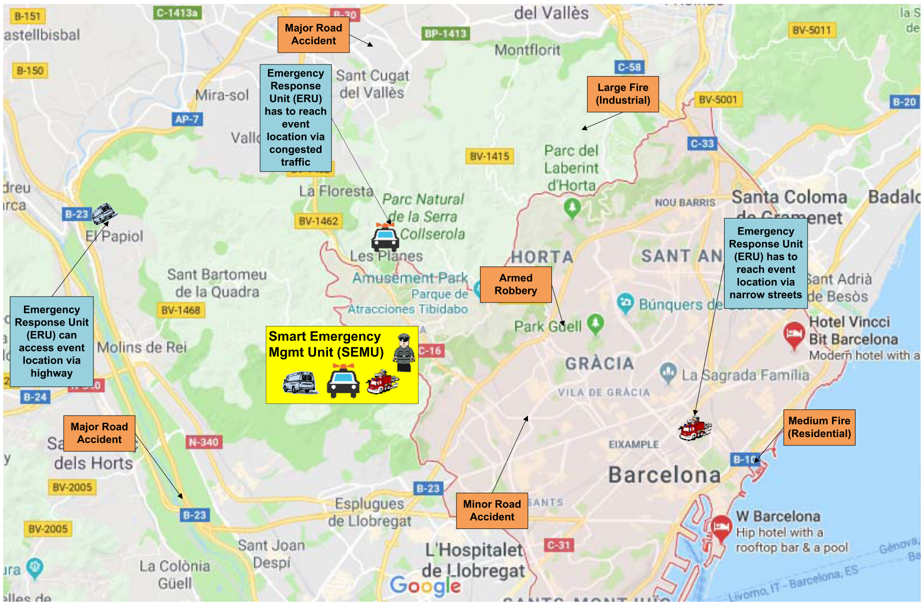

FIGURE 1. An example scenario illustrating several types of events handled by Smart Emergency Management Unit (SEMU).

designers can obtain such data from city's historical records (previous occurrences) or human domain expertise. Given the nature of IoT based system for event detection, the reported event and its related information (like severity) might not be accurate and thus has to be modeled as a probabilistic or a non-deterministic measure.

The SEMU responds to events occurring throughout the city by dispatching appropriate ERUs. The ERU dispatch policy has to take into account the above mentioned factors and the availability of resources (ERUs, ERU operational staff, materials etc.). Given the non-deterministic and probabilistic nature of events, the event location accessibility and the availability of resources, a human operator might be needed in the SEMU to assist in the efficient assignment of appropriate resources for event handling.

In this paper we model the concept of smart city that is managed by the following key components:

- "Event detector (ED)": This component consists of a variety of sensors that detect various kinds of incidents such as fire, road accidents, etc. ED detects the occurrence of an event and propagates this information via available networking channels to the smart emergency management unit.
- "Smart emergency management unit (SEMU)": It is the main unit responsible for reacting to the emergency and taking appropriate action. This unit is a human-in-the-loop based system, which may resort to human expertise if needed. Based on a particular type of event, it responds to the event by sending relevant emergency response unit to the location of the event.
- "Emergency response unit (ERU)": This unit is sent by SEMU to take care of the emergency situation.

Our model captures the interaction among these components and helps us analyze various key parameters that one must take into consideration in the design of a smart city. Designing a smart city is a challenging task because of the following issues:

- Events such as fire, burglary, accident, may occur without any pattern. In other words, their occurrence may not be fully predictable, thereby making it difficult to model them as a simple probabilistic measure. Their non-deterministic nature needs to be preserved in order to realistically capture the occurrence of such events.
- Sensors deployed to detect events are prone to errors and may initiate false alarms. Detecting these false alarms is essential in order to utilize key resources, such as ambulance or fire engine, of a smart city in efficient manner.
- The automated system deployed to detect false alarm may also suffer from its inherent constraints and may not perform at 100% accuracy. A human-in-the-loop may help compensate for the errors inherent in a fully automated system.
- Although the involvement of human in the partially automated system (hybrid system) helps alleviate the problems encountered in the fully automated system, this does not come without a price. A human operator may also suffer from various constraints, such as fatigue and work overload conditions, thereby resulting in degradation of performance even in the hybrid system.
- The degree of human involvement in the hybrid system within SEMU is another design parameter that needs to be selected wisely. Various degrees of human involvement may suffice for different types of regions within the same city.
- A variety of external factors, such as traffic condition at certain time of a day, as well as internal factors, such as the availability of resources present at the time of an

event, influence the emergency response time of ERU in a smart city. In the presence of these uncertainties, guaranteeing a certain SLA (service level agreement) needs careful analysis.

Considering the above mentioned parameters and constraints inherent in the system, designing a smart city involves analyzing various situations before the actual deployment and implementation of various hardware and software resources. Our work focuses on these parameters and addresses key issues in the design and management of a smart city. In particular, with the help of model checking, we are able to answer key design questions in the pre-design phase in a smart city. Some of these questions include:

- What is the probability to serve an emergency request generated from a remote place in a given time in a city of particular size?
- How do the traffic conditions effect the incident response time in a city of particular area?
- What level of human involvement is needed to reduce the number of false alarms to meet a particular service level agreement (SLA)?

The proposed smart city model allows us to answer such questions before the actual deployment of the resources. To the best of our knowledge, this is the first time the aforementioned parameters are used to answer such questions in the design of a smart city.

II. BACKGROUND

In this section we present the relevant background about different ways of modeling IoT based smart city governance aspects. We also elaborate how probabilistic model checker PRISM can be used to model a system that exhibits probabilistic as well as non-deterministic behavior.

A. RELATED WORK

A smart city initiative heavily depends on the IoT technology. Thus, in order to implement an IoT-based smart city, it is essential to focus on the different modeling approaches to capture the system dynamics [12]. Several approaches have been proposed in the literature whose brief summary is presented below.

The deployment of the IoT-based systems in the context of smart cities significantly increases the complexity, not only in the number of nodes and structure, but also in the increased heterogeneity of protocols and mechanisms. In order to capture and model such a complex system behavior the concept of ''islands of resilience''is proposed which continue to function even when disconnected from main system during emergency situations [14].

A wireless sensor network (WSN) consisting of multiple smoke and fire sensors was designed and evaluated for early detection of fire and GSM system was employed to avoid false alarms [11]. Simulated environment using Fire Dynamics Simulator was used to verify the results achieved from the real-time system. Their focus is on the dynamics of how a fire spreads and the prediction of such incidents.

For the city planners to effectively deal with emergency events, an eMergency-CREAtivity Machine (M-CREAM) tool has been proposed which is a model-based elicitation of emergency management scenarios. An emergency case study involved modelling an environmental hazard related to drinking water contamination with the aid of ''mini-stories'' concept [15].

In the context of smart city management, a general system design and review is proposed which consists of a central agent which monitors the functioning of three constituent layers, namely, UAV layer, Robot layer and WSN layer to achieve the efficient management of emergency situations [16]. Our proposed work is targeting the same domain but provides a comprehensive model checking methodology to evaluate the desired resource availability and incident response time for managing large scale emergency units without conducting a wholesome simulation for all possible scenarios.

The behavior of an IoT-based smart application management system is modelled using Finite State Machines and the Continuous Density Input/Output Hidden Markov Model (CD-IOHMM). The uncertainties of the system behavior are captured using probabilities and the models are verified using real datasets [17]. A layered modelling architecture (using Hybrid Particle Swarm Optimization (HPSO)) has been proposed to capture the uncertainties and heterogeneity inherent in a smart city which efficiently allocates required resources at the location of an emergency event [18].

Colored Petrinets (CPN) have been used to model a sustainable security framework for IoT systems catering to the eHealth service under the smart city domain [19]. The CPN uses a bipartite graph to model the places and transitions which are then connected through directed edges to represent the dynamics of the system under consideration. However, our proposed approach (PMC) is a formal verification tool which can be used to examine whether the model of a system, which probably was implemented using CPN, meets the requirements of the system.

An IOT-based smart home security system was modeled to study the security risks involved in such systems. They used PRISM, a probabilistic model checking tool, to represent the threat models through Markov Decision Processes (MDP) and evaluated risks for various system configurations [20]. However their focus was on security aspects of the IoT-based systems and our focus is on the optimal resource utilization for public safety services.

Renewable and pollution-free energy generation is one of the prime concerns for a smart city infrastructure which has been studied as a microgrid [21]. The authors have used IoT technology for monitoring and control of microgrid and have verified their reliability through probabilistic model checking tool (PRISM). Their work is specific in nature which studies the failure rate of individual system components and its effect on the overall system reliability. However they do not consider the effect of human-in-the-loop which is a prominent component of a smart-city setup.

Vehicle-to-Vehicle (V2V) networks are used in a smart city to alert drivers in case of safety hazards such traffic congestion, road diversions and approaching of emergency service vehicles [22]. To study the survivability of such networks, the paper modeled the V2V network as a CTMC and used the PRISM tool to perform impact analysis under node and link failure conditions. This work models the V2V network as a CTMC assuming only probabilistic nature of the system. However, we consider both probabilistic and non-determinism in our modeling framework.

A System of Systems (SoS) concept is used to model a mass casualty incident (e.g. massive fire, building collapse) response system in a smart city, using probabilistic modeling tool (PRISM) [23]. Such a model captures uncertain behaviors of a SoS in a quantitative manner and performs statistical model checking in order to verify a user-defined SoS-level goal (e.g. minimizing average response time to a casualty incident). Their work emphasizes the modeling of various flavors of SoS through probabilistic means and does not elaborate on the exact model details.

A probabilistic approach is proposed which formally describes and analyzes the reliability and cost-related properties of the service composition in IoT [24]. A service composition in IoT is modeled as a Markov Decision Process (MDP), which specifies the reliability of service operations. PRISM, a probabilistic model checker, is then used to verify and analyze the specified properties. This work is the closest to our approach, however they focus only on single type of event (fire mishap) and also they do not take into account the human involvement in decision-making.

Based on the literature survey performed above, it is proposed to model a system (a smart city in our case) using PMC due to the following benefits. Firstly, it gives the system modeler a flexibility to visualize and test various scenarios which can happen in a city. Without actually a particular event happening he can know the effects of such an event on the city dynamics. Secondly, a "real" city infrastructure need not be experimented with an actual emergency event, which may adversely affect the lives and property. Lastly, a parametric analysis can be easily performed by observing the effect of a particular parameter of interest (e.g. traffic congestion) on the system behavior. This study can help to predict the resource requirements which can be arranged in advance so as to be prepared to deal with the emergency event, if it actually happens.

B. PROBABILISTIC MODEL CHECKING

Probabilistic model checking is a formal technique that can capture systems with probabilistic as well as nondeterministic behavior. The system is usually represented in the form of a state-transition diagram where each state represents a set of values that the system assumes at a particular instant of time. Actions are defined on each state that trigger transitions where the system moves from one state to the other. An action with a known probability distribution may take the system to one of the several possible states with some probability. If there are more than one action available at a state, an action may be chosen nondeterministically. Once the system is modeled with the help of state-transition diagram, it is often encoded in some meta language so that the system properties can be verified against this model. System properties are captured using formulas in some form of temporal logic, such as linear temporal logic (LTL). The syntax and semantics of the formulas in temporal logic are well defined. Temporal logic is equipped with rich temporal operators that can be used to reason about various time epochs in future. With the help of these operators, we can capture key properties that we want to verify in the system.

PRISM is a well known probabilistic model checker that can be used to model systems with probabilistic as well as nondeterministic behavior [25]. PRISM has gained its popularity because various systems from diverse application domains has been modeled and analyzed with the help of this tool. Some of the application domains where PRISM has been successfully used include communication protocols, distributed algorithms, security protocols, and biological systems [26]–[30]. PRISM typically captures a system with the help of the following stochastic models:

1) Discrete-time Markov chains (DTMCs)
2) Continuous-time Markov chains (CTMCs)
3) Markov decision processes (MDPs)
4) Probabilistic automata (PAs)
5) Probabilistic timed automata (PTAs)

We use PRISM for capturing the behavior of each component of a smart city. In particular, we design the abstract model of the components of the system in the form of MDP and DTMC. As there are more than one components involved, each part of the system is modeled individually. The model of the entire system is obtained through parallel composition of individual elements. This model is encoded in the language of PRISM and then it is fed to the tool for the purpose of analysis. The required specification can be given in the form of *linear temporal logic (LTL)*. PRISM is used to either check if the system satisfies the given specifications under a strategy, or to synthesize a strategy that meets some specifications.

III. COMPUTATIONAL MODEL

The system consists of three main components namely "Event Detector (ED)", "Smart Emergency Management Unit (SEMU)", and "Emergency Response Unit (ERU)". Due to its probabilistic nature, the SEMU is modeled as a Discrete Time Markov Chain (DTMC), whereas the ED and ERU are modeled as Markov Decision Process (MDP) because events are generated non-deterministically. The models for ED, SEMU and ERU are depicted in Figures 2, 3 and 4 respectively.

Since any type of event can happen at any location, the ED module models both the occurrence and the location of an event nondeterministically. The model assumes input coming from a variety of sensors installed on the key locations within a city. These sensors produce large volumes of data that has to be collected and fused to suggest occurrence of an event [31].

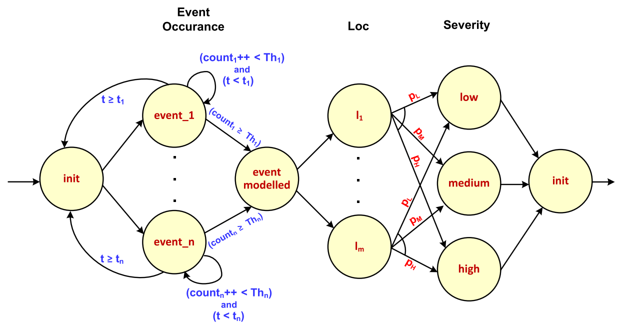

FIGURE 2. Markov Decision Process (MDP) modeling of Event Detector (ED).

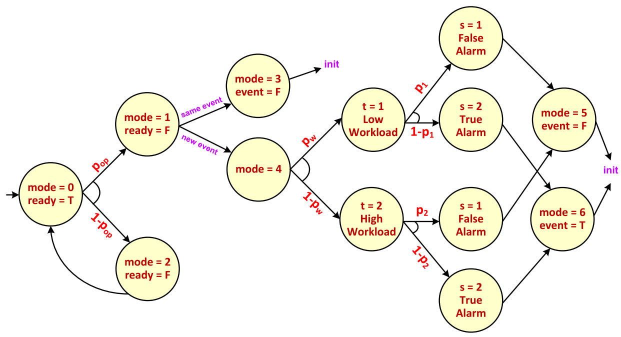

FIGURE 3. Discrete Time Markov Chain (DTMC) modeling of human-in-the-loop based Smart Emergency Management Unit (SEMU).

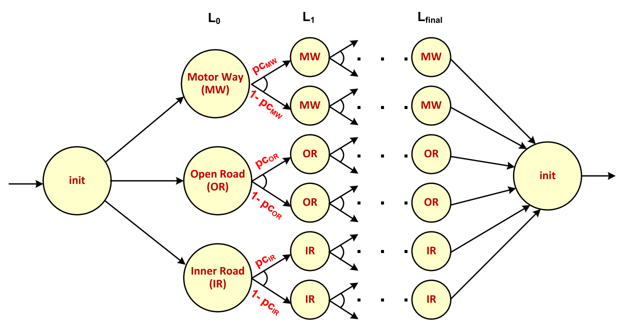

FIGURE 4. Markov Decision Process (MDP) modeling of Emergency Response Unit (ERU).

Several important resources of Smart City (SC) are monitored by a variety of sensors. For example, streets can be monitored by acoustic sensors and cameras and buildings can be monitored with fire, heat, or door sensors. SC authorities choose appropriate sensors based on requirements, such as environmental conditions, accuracy, range, cost, resolution, etc. The sensors report anomaly in the sensed data by forwarding it to event detector. For example, in the case of heat sensor, it reports to ED when it detects anomaly in temperature. In a smart city, CCTV output can be fed to Computer Vision based algorithms for detecting possible anomalies (such as accidents). Similarly, human can also provide input to ED. For example, in the case of an emergency, such as bank robbery, someone in the bank can initiate an emergency signal.

ED module examines sensors' data based on count-based and time-based thresholds. These thresholds may differ depending on sensor and event types. Once ED module reports an emergency event, the sensed data from these sensors is examined at SEMU to possibly respond to an event. Human operators can help in reducing the number of false events. ERUs (ambulances, fire brigades, security units) are sent in response to these emergency events. Consider the following sets:

$$E = \{e_1, e_2, \dots e_n\},$$
$$Th = \{th_1, th_2, \dots th_n \mid th_i \in \mathbb{N}\}, \quad \text{and}$$
$$t = \{t_1, t_2, \dots t_n \mid t_i \in \mathbb{N}\}$$

and the functions *Sensor-Threshold (ST)* and *Timer-Threshold (TT)* such that $\forall e_i \in E, th_i \in Th, t_i \in t$:

$$th_i = ST(e_i) \quad \text{and}$$
$$t_i = TT(e_i)$$

That is, for every event e_i, the model selects appropriate thresholds, sensor threshold th_i, and timer threshold t_i, that dictate when to move to the next state. For an event e_i, a value $count_i \in \mathbb{N}$ is incremented as a sensor input is received. As soon as $count_i \geq th_i$ within certain timeout period t_i, an event is recorded by the ED model as shown in Fig. 2. Otherwise, if the time passes beyond t_i, the model resets itself to the initial state. The threshold values for an event depends upon the type of event. For instance, in the case of fire in a forest, the threshold values (th_i and t_i) for detecting the fire may be relatively higher since many sensors need to detect fire and propagate this information within t_i time units.

Consider the following sets representing all possible event locations and the parameters associated with each event:

$$Loc = \{l_1, l_2, \dots l_m\} \quad \text{and}$$
$$Param = \{P_1, P_2, \dots P_n\}$$

such that $\forall e_i \in E, P_i \in Param$ represents a list of parameters that is related to event e_i. The ED detects the severity of an event as a function f that depends on the event type $e_i \in E$, its location $l_j \in Loc$, and the parameters $P_i = \langle p_1^i, \dots, p_k^i \rangle$ relevant to that event. That is:

$$sev_i = f(e_i, l_j, p_1^i, \dots, p_k^i).$$

For every parameter p_j^i related to an event e_i in $\langle p_1^i \dots p_k^i \rangle$, there is a corresponding threshold value T_j^i in $\langle T_1^i \dots T_k^i \rangle$ where $\forall j \in \{1 \dots k\}$, $T_j^i \in \{T_j^{low}, T_j^{med}, T_j^{high}\}$. Thus, the severity $sev_i \in \{low, medium, high\}$ of an event e_i can be defined such that the following formula φ evaluates to *true*.

$$\varphi = p_1^i \geq T_1^i \circ p_2^i \geq T_2^i \circ \dots \circ p_k^i \geq T_k^i$$

Here the operator $\circ \in \{\wedge, \vee\}$ joins each propositional statement $p_j^i \geq T_j^i$ with others via conjunction or disjunction. Although in the present model, these thresholds are selected probabilistically, but in any realistic scenario, each sensor will have its own threshold value that depends on the event type and its severity. For instance, in the case of environmental monitoring, the severity of the environment pollution can be calculated from the sensor readings for air quality, noise levels, and luminosity levels at a particular location [32]. The abnormal levels of activity observed by the relevant sensors may indicate an event to be severe, and may be calculated by fusing different types of available spatial data [33]. For a particular event e_i at a given location l_j, we model its severity probabilistically and the event detector moves to one of the states based on severity level *high, medium,* or *low* with the probability p_h, p_m, or p_l such that the following holds true.

$$p_h + p_m + p_l = 1$$

After an event is detected by event detector, the SEMU chooses the operator's assistance with a probability p_{op} (see Fig. 3). The operator's involvement depends on the severity of the event. If the severity level of the event is *high*, the highest value of p_{op} is chosen. On the other hand, if the event is of *low* severity, p_{op} is reduced by a constant factor c. That is:

$$p_{op} = \begin{cases} p_{op}^{max} & \text{if } sev_i = high \\ c \cdot p_{op}^{max} & \text{otherwise } (c < 1) \end{cases}$$

Based on the recent historical event pattern, the operator may identify the possibility of a false alarm. The operator may also resort to a detailed analysis before identifying false alarms. The operator's performance varies in different workload conditions. That is, an operator with low workload examines the event with greater accuracy as compared to an operator with a high workload. Similarly, the operator gets fatigued and his performance deteriorates as time progresses. We leverage the operator's model proposed in [34]. The operator with low workload examines the event with probabilities $p_{low}(f_p)$ and $1 - p_{low}(f_p)$ where f_p measures the operator's fatigue level and $p_{low}(f_p)$ is a function over variable f_p. For a fixed fatigue threshold F_{th},

$$p_{low}(f_p) = \begin{cases} p_{low}(0) & f_p < F_{th} \\ f_d \cdot p_{low}(0) & f_p \geq F_{th} \end{cases}$$

That is, after a certain threshold, operator's accuracy is discounted by a factor f_d (which is less than 1). The fatigue discount factor of the operator also depends on the severity of the event as a severe event tends to increase the focus resulting in improved operator's performance. We model it be adjusting f_d by a constant factor. Operator's accuracy at high workload $p_{high}(f_p)$ and $1 - p_{high}(f_p)$ is defined in the similar way.

Emergency response unit responds after detecting an event by setting its destination to be the location of the event. Based on location information provided by the event detector, ERU chooses the fastest route available. In the model the route choice is made nondeterministically (see Fig. 4). We assume that city routes comprise of three different types of roads, *motorway (MW), open road (OR), or inner road (IR)* as per EU norms [35]. The traffic conditions on roads is determined by the probability of congestion p_{c_i} where $i \in \{MW, OR, IR\}$. For any particular road, these probabilities can be obtained as the average number of vehicles crossing a street per hour. For instance, [36] plots the average cars per hour at different times of the day for a street with two lanes in Duesseldorf over two months obtained from real-world detectors.

TABLE 1. The notations used in the model.

Notations	Description
e_i	An event from the set $E = \{e_1, e_2, \ldots e_n\}$
th_i	The threshold ($\in Th$) of sensors' input for event e_i
t_i	The threshold ($\in t$) of timeout for event e_i
$count_i$	The sensors' input counter for event e_i
l_i	The location of an event ($\in Loc$)
p_j^i	The j^{th} parameter in $P_i \in Param$ related to event e_i
T_j^i	The threshold value corresponding to p_j^i
sev_i	The severity of an event e_i
p_m	The probability that the event's severity is m where $m \in \{l, m, h\}$
p_{op}	The probability of operator's involvement
F_{th}	Fatigue threshold of the operator
f_p	Operator's fatigue level ($\in \mathbb{N}$)
p_{low}	The probability of operator's accuracy at low workload
p_{high}	The probability of operator's accuracy at high workload
p_{c_i}	The probability of congestion at road i
L_i	Current location of the ERU in its path $Path_{eru}$

We model the city as a virtual grid of size $n \times n$ where each block of the grid (represented as $[i, j]$) represents an area/zone/district within the city as represented by location L_k in Fig. 4 where $k \in \{1, 2, \ldots final\}$. Here $L_{final} \in Loc$ represents the location of the event. Even though the ERU selects the best path to its destination, this never guarantees that the congestion on the selected road stays uniform along the way. Therefore, the congestion on the road is recomputed every time when ERU moves from one location/block to the other. The path that ERU follows to its destination is represented as the following tuple:

$$Path_{eru} = \langle L_1, L_2, \ldots L_{final} \rangle$$

such that L_1 represents the staring position of the ERU, L_{final} represents the location of the event, and for every L_k in the $Path_{eru}$:

$$L_k = [i, j] \Rightarrow L_{k+1} = [i, j \pm 1] \vee [i \pm 1, j]$$

Each side of the cell in the virtual grid represents unit distance, which takes one unit time to travel. However, the actual travel time is determined by road type and traffic conditions. The service provider is located at the city center, and depending upon the availability an ERU is immediately dispatched to the requested location at the occurrence of an event. The notations used in our model are summarized in Table 1.

The interaction among ED, ERU and SEMU components is captured by parallel composition of these models. That is, the alphabetized parallel composition of modules M_{ed}, M_{eru}, and M_{semu} that synchronize on only actions appearing in all the modules. This parallel composition is associative and can be applied to more than two modules at once. Further details about other parallel composition operators are available at [37]. In general, interaction between two models M_1 and M_2 is captured as $M_1||M_2 = (Q_1 \times Q_2, (\bar{q_1}, \bar{q_2}), \mathcal{A}_1 \times \mathcal{A}_2, \delta)$ where the transition relation δ of the product MDP captures the run of the system. δ is defined such that two MDPs synchronize on common actions and interleave otherwise. That is, $((q_1, q_2), a, \mu_1 \times \mu_2) \in \delta$ if and only if one of the following holds:

- Both models synchronize on common action. That is, $(q_1, a, \mu_1) \in \delta_1, (q_2, a, \mu_2) \in \delta_2$, and $a \in \mathcal{A}_1 \cap \mathcal{A}_2$
- M_1 takes the transition on its action. That is, $(q_1, a, \mu_1) \in \delta_1, \mu_2 = \eta_{q_2}$ and $a \in \mathcal{A}_1 \backslash \mathcal{A}_2$
- M_2 the transition on its action. That is, $(q_2, a, \mu_2) \in \delta_2$, $\mu_1 = \eta_{q_1}$ and $a \in \mathcal{A}_2 \backslash \mathcal{A}_1$

The product model represents the execution of both running in parallel and synchronizing at common actions. In the absence of non-determinism in a model M_1, synthesizing a strategy π for the product MDP $(M_1||M_2)$ gives the strategy π' for M_2. That is, $(M_1||M_2)^{\pi} = M_1||M_2^{\pi'}$. The resulting MDP can be solved for finding the optimal value V^* for each state $s \in S$ using the following Bellman's equation:

$$V^*(s) = \max_a \{R(s, a) + \gamma \sum_{s' \in S} P(s' \mid s, a)V^*(s')\}$$

Here R is the reward function and γ is the discount on future rewards. Value (or policy) iteration can be used to solve this equation where V^* is computed iteratively until it converges. Now the optimal policy (or strategy) can be calculated as:

$$\pi^*(s) = \arg\max_a \{R(s, a) + \gamma \sum_{s' \in S} P(s' \mid s, a)V^*(s')\}$$

Figure 5 depicts the schematic flow of our work. It shows how PRISM model checker takes input from non-deterministic model interaction, system specifications that meet application objectives and component-level constraints, and verifies if the system meets the desired specifications under given constraints. In particular,

- Based on the requirements of each application, system specifications are written in the form of LTL (linear temporal logic) formulae.
- Constraints related to event generator, event handler, or service provider can be encoded in PRISM language as module variables or predicates in PRISM commands.
- Interaction among each module, with its own relevant parameters, represents non-deterministic behavior of the system and is captured as the product MDP.

The output of the PRISM model checker is of two types. It verifies or calculates the probability to meet given property specification based on system parameters. It can also calculate the value of the reward based on input parameters of the system and given LTL specifications. For example, it can calculate the probability to service an emergency request within given constraints (such as in T time units for varying city sizes). In the property specification $P_{max} = ?[\Phi]$, Φ could be any LTL expression that needs to be evaluated, such as an expression indicating that ERU reaches the event location. Here P_{max} is the maximum probability to meet the given property specification. PRISM can also be used to calculate the reward (such as *expected incident response time*) based on system parameters (such as *city size* and *probability of congestion*). To calculate reward, the property specification $R\{reward\} = ?[\Phi]$ can be used where R is the PRISM construct for calculating the reward.

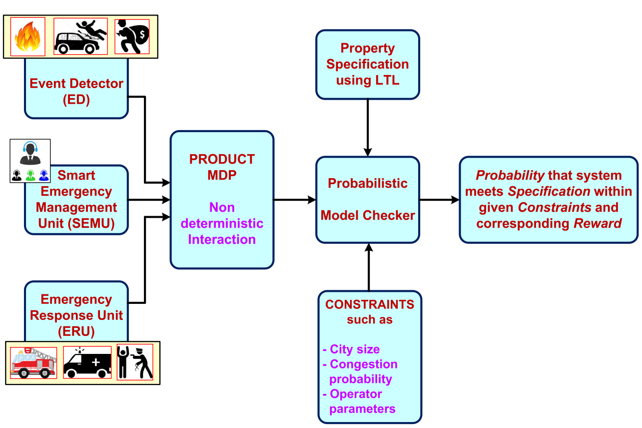

FIGURE 5. Schematic diagram of the proposed system.

IV. MODEL IMPLEMENTATION AND EVALUATION

The system model described in Section III is implemented in PRISM modeling language and the system properties are verified by using PRISM model checker. This section describes several Smart City (SC) scenarios, and discusses how Probabilistic Model Checking (PMC) can be leveraged to estimate SC performance metrics. The time and distance units are kept generic in all scenarios for better applicability. For example, we assumed the time it takes to traverse one distance unit on motorways is one time unit. One distance unit could be one meter or mile or kilometer, etc; similarly one time unit could be one second or minute or hour, etc. We have divided the evaluation results into two main cases: (i) smart cities with fully autonomous emergency response services, and (ii) smart cities with operator assisted emergency response services. The list of parameters values are given in Table 2.

A. AUTONOMOUS EMERGENCY RESPONSE SYSTEM

This case study considers the smart cities (SCs) that have centrally located emergency response units (ERUs) in one SEMU, which are dispatched autonomously by an automated system without any involvement of skilled human operators. We are considering different SCs with areas varying from 25 to 900 sq. units.

1) ESTIMATION OF EMERGENCY RESPONSE LATENCY

The quality of emergency response depends on how quickly a service team (ERU) has reached the incident location. We are measuring the latency using Incident Response Time (IRT),

TABLE 2. Parameters used in PRISM models.

Parameters	Values
City size (n×n regions)	5×5 to 30×30
City area	25-900 sq. units
Road types	MW, OR, IR
Probability of volatility (p_e)	0.1-0.9
Probability of congestion p_c	0.1-0.9
Probability of workload (low/high) (p_w)	0.5
Probability of accuracy in low load (p_1)	1
Probability of accuracy in high load (p_2)	0.7
Probability of operator's involvement (p_{op})	0-0.9
Fatigue threshold (F_{th})	1-10
Fatigue discount factor (f_d)	0.7

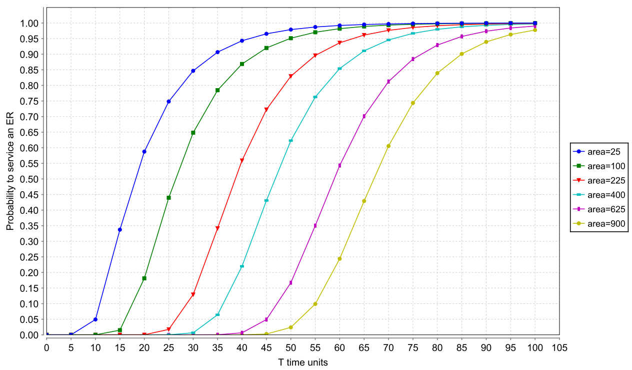

FIGURE 6. The probability to reach the incident location within *T* time units for varying smart city sizes.

which refers to the time difference between incident generation and an ERU reaching the incident location assuming that ERU is readily available in SEMU. IRT is an important parameter in the design of SC's emergency services. High IRT may significantly increase human and financial losses [38]. Figure 6 shows the probability to service an emergency request (ER) generated from a distant corner of the city within T time units for varying city areas. We assumed the roads are moderately congested ($p_c = 0.5$) and road type is MW. The objective of this plot is to estimate the best/worst IRT to reach remote areas of the city. From this figure, we can see that the minimum and maximum latency to service the request is 6 and 60 time units respectively for a city of area 25 square units. The latency quickly increases with the increase in city area. For a large SC with area 900 sq. units, the minimum IRT is 46 time units. Under these circumstances, if a SC requires IRT to be less than 55 time units for 90% of the time, then the city area cannot exceed 225 sq. units. The IRT can be reduced by deploying more Smart Emergency Management Units (SEMUs) across the city. Ideally each suburb of the city should have one SEMU, which may not be feasible due to cost and space constraints.

2) EMERGENCY RESPONSE LATENCY UNDER VARIOUS TRAFFIC CONDITIONS

Smart city traffic conditions are influenced by time of the day, season, activities, etc. Bad traffic conditions, as point out in [38], significantly influence ERU journey time, fuel consumption, and air pollution. To study the impact of traffic conditions we varied the traffic congestion probabilities from 0 (no roads are congested) to 0.8 (highly congested roads). When the probability of congestion is 1.0 then the ERU will not be able to move. High and low values of p_c may represent various realistic situations. For example, low values of p_c can model the situation when ERU has nondeterministically chosen a highway to reach its destination. On the other hand, if inner city roads are selected by ERU, p_c will assume a higher value. Our model helps in determining the influence of traffic congestion on SC emergency services.

Fig. 7 shows the IRT with respect to city size for varying probabilities of congestion. It can be noticed from the figure that low to moderate traffic congestion ($p_c < 0.5$) doesn't drastically change the IRT especially for small cities (i.e. SCs < 100 sq units). However, heavy traffic congestion ($p_c = 0.9$), which are more likely on IRs and less likely on MWs, has severe impact on IRT. In other words, large cities with high probability of congestion should deploy more SEMUs in order to meet the required response time requirements.

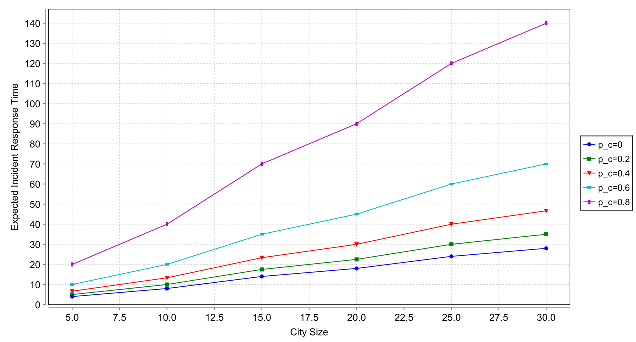

FIGURE 7. The time to reach the incident location for varying city sizes at different congestion levels.

3) EMERGENCY RESPONSE LATENCY WITH ADDITIONAL SEMUS

As discussed in previous subsection, installing more SEMUs will reduce the IRT in smart cities (SCs). Careful pre-positioning of SEMUs can reduce IRT and the total number of required SEMUs as suggested in [39]. We have studied the influence of a number of SEMUs (N_{semu}) on IRT for various SCs under different traffic conditions (p_c). It is assumed that emergency requests in a SC are uniformly distributed, and hence additional SEMUs are placed evenly across the city. Fig. 8 shows the N_{semu} needed for different city sizes under distinct traffic conditions. The chosen city sizes are: small (100 sq. units), medium (200 sq. units), and large (400 sq. units). These city sizes can be mapped to any realistic units in sq. km or sq. miles. The traffic conditions on roads could be: *NoCongestion, LowCongestion, MidCongestion, HighCongestion*. The cities with no congestion are represented by ($p_c = 0$), whereas high congestion is represented by ($p_c = 0.9$). The results show

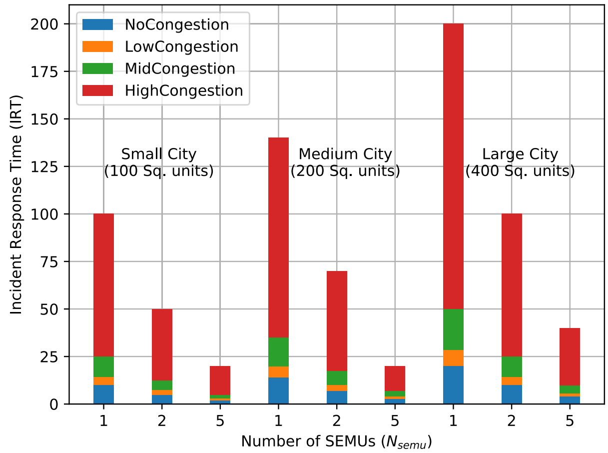

FIGURE 8. **IRT with different number of SEMU locations for three smart city sizes under various traffic conditions.**

that the IRT decreases proportionately with the increase in N_{semu}. Although its influence appears to be higher under high congestion conditions, the percentage of change is the same. IRT decreases by 80% when N_{semu} is increased to 5, regardless of traffic conditions and city areas. In other words, to maintain the same level of response time, N_{semu} should be increased with the traffic congestion. For example, to achieve the same IRT during the high traffic congestion ($p_c = 0.9$) as that of the IRT when no traffic congestion ($p_c = 0$), we need at least 10 uniformly distributed SEMUs across the city.

B. OPERATOR ASSISTED EMERGENCY RESPONSE SYSTEM

In the previous scenarios, the decisions to dispatch ERUs were made by an autonomous system. Its task was to take decisions based on the received sensor inputs. There are several situations where available sensor information is not sufficient to take correct decisions, and may result in false positive or negative alarms. Involvement of a skilled human operator in the decision process will improve the quality of decisions there by reducing the number of false alarms. However, human operators have their own drawbacks. Operator decision making capabilities (quality) are influenced by their skill levels, current workload, and fatigue thresholds. These factors need to be considered in the design of smart city emergency response systems that are assisted by human operators. In this subsection, we are examining the effect of different skilled human operators on the service availability and number of false alarms.

1) FALSE ALARMS

In Section IV-A, we assumed that all service requests are genuine and there are no false or missed alarms. In reality, due to various reasons, such as sensor failures, there is a chance of error in event detection. If these errors are not identified properly, they may result in either unnecessary ERU dispatches or emergency events may not get timely service. A skilled operator can help in reducing/eliminating the FAs. However, the operator's performance is influenced by his current workload and fatigue factors.

Fig. 9 shows the percentage of false alarms for different levels of operator involvement (p_{op}), fatigue thresholds (F_{th}) and workloads (W). The percentage of operator involvement in a session is modeled using p_{op}. We chose four levels of p_{op} - 0, 0.3, 0.6 and 0.9. The value $p_{op} = 0$ refers to the case when there is no operator. From this figure, it is evident that operator involvement reduces the false alarms considerably. For example, when $p_{op} = 0.9$ the false alarm percentage dropped by 81% as compared to the case with no operator involvement. However, this result didn't account for operator workload or his fatigue tolerance. In reality, the human operator will not produce ideal results under all workloads. His decision making skills are affected by his current workload and fatigue thresholds.

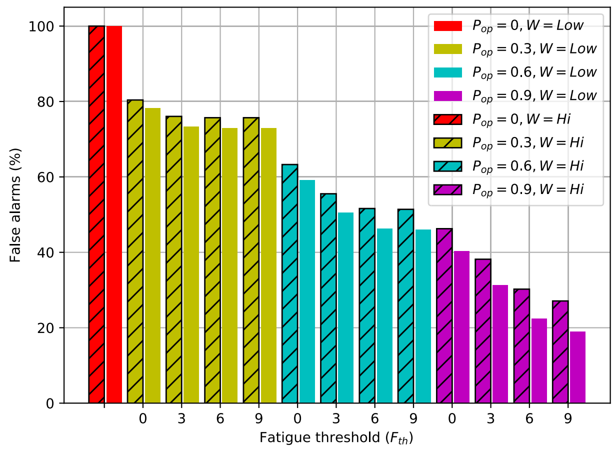

FIGURE 9. **False alarm percentage for various levels of skilled operator involvement.**

We considered four different operator fatigue thresholds (F_{th}) representing four diverse operators; higher values of F_{th} mean more robust operator, and workload will have any little influence on his performance. The results show that robust operator improves the quality of emergency services noticeably, especially when p_{op} is high. When $p_{op} = 0.9$, a robust operator (i.e. $F_{th} = 9$) reduces the FA percentage by 21% than that of the operator with low fatigue threshold ($F_{th} = 0$). Please note that F_{th} has no influence on the performance when there is no operator ($p_{op} = 0$), and hence the variance with F_{th} are omitted for this case from the figure.

We further examined different workload levels of operator. Fig. 9 shows the results for two workloads (low and high). Higher workload degrades the operator performance in all cases. Its impact is more when the operator is heavily involved in the decision process, regardless of operator's fatigue threshold. For example, under high workloads with $p_{op} = 0.9$ and $F_{th} = 9$, the FA percentage increases by 6% as compared to less loaded operator.

2) SERVICE AVAILABILITY

In this section, we study the probability of ERU availability for cities with diverse incident rates. We denote average incident rate as probability of volatility (P_{vty}). We consider three cities with low ($P_{vty} = 0.1$), medium ($P_{vty} = 0.5$) and high ($P_{vty} = 0.9$) incident rates. We are measuring the probability of ERU availability (P_{SA}) during the times of unrest (volatile times). We also changed the volatility periods (T_{vty}) from small duration (20) to long duration (80) time units. We study these parameters for different levels of operator involvement.

Fig. 10 shows how ERU availability (P_{SA}) is affected for cities with various incident rates, volatility periods, and levels of operator involvement. We can see that P_{SA} quickly drops to low values when the volatility period (T_{vty}) is high. In the cities with low probability of volatility (P_{vty}), the ERU availability (P_{SA}) is high. A fully autonomous SEMU can service a city with incident rate less than 10% and volatility duration less than 20 time units. However, in the cities with high incident rates an operator involvement up to 60% can still guarantee a high service availability.

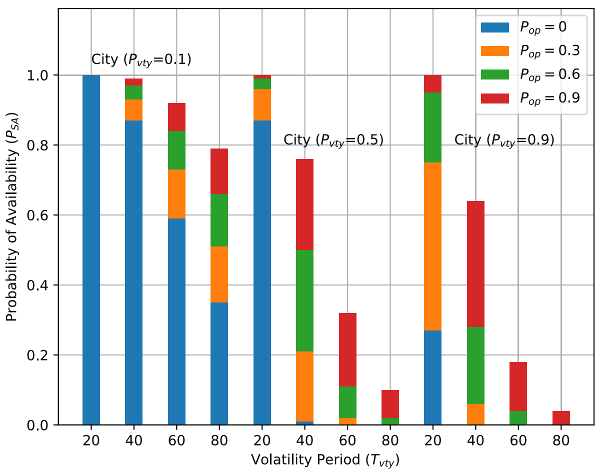

FIGURE 10. Probability of service availability for cities with different incident rates during periods of distress.

For higher volatility duration an operator must be involved to maintain service availability. This means, in rare situations, when safer cities have high periods of unrest, P_{SA} can be improved by increasing the operator involvement (p_{op}). However, for the cities with high incident rates ($P_{vty} \geq 0.6$), a good level of P_{SA} cannot be maintained even with full operator involvement. In such circumstances, we might need to bring extra ERUs from other cities.

C. SUMMARY OF RESULTS

As demonstrated in the previous subsections, with the help of model checking a smart city designer will be able to decide key parameters before the actual deployment of the infrastructure. Our approach allows a designer to answer a broad range of questions. Here are the few examples:

- In cities of varying sizes (small, medium, large) and at a given traffic congestion, how many SEMUs are required to maintain certain incident response time? For example, in a city of 100 sq. units with no congestion five SEMUs are needed to achieve IRT of two time units.
- In the presence of human operators of varying skills set and workload conditions, what is the degree of operator's involvement needed to maintain the false alarm rate under certain threshold? For example, a skilled operator working at light workload conditions is required to be involved at least 90% of the time to keep false alarm rate less than 20%.
- What is the maximum area of a city that could be serviced, if the response time needs to be less than certain value for majority of cases? For example, if a city requires response time to be less than 55 time units for 90% of the cases, then the city area cannot exceed 225 sq. units.
- How will the response time get affected if a city experiences sudden traffic congestion? For example, the response time in a city of 400 sq. units will be increased by 58% if the probability of congestion is increased by 40%.
- In a city of varying volatility, what is the degree of operator's involvement needed to keep the availability of resources (such as ERUs) within required threshold? For example, to maintain service availability around 90% in a moderately volatile city, a minimum of 60% operator's involvement is needed.

In general, our work allows a designer to analyze a complex sets of parameters to meet a variety of service level quality requirements in a smart city without actual deployment or time consuming simulations. However, modeling big systems has to be done with extreme care in order to avoid state explosion problem. For example, the complex interaction among multiple SEMUs and ERUs in a smart city may result in huge number of states and hence the system needs to be carefully modeled following the general guidelines given in [37]. Another limitation to our work is related to those type of events that are catastrophic in nature. These events may result in total collapse of smart city infrastructure and are not handled in our present work.

V. CONCLUSION AND FUTURE WORK

This paper modeled emergency management services in smart cities using probabilistic model checking, and demonstrated its benefits in planning and analyzing the performance of these services. We have presented a holistic approach to model the smart cities as a dynamic, uncertain and complex environment where multiple events with varying severity take place in a continuous and non-deterministic manner at several locations in the city. Our model takes into consideration completely autonomous emergency management systems as well as semi-autonomous systems that leverage the proficiency of a human operator. We have taken several parameters into account including city size and types, event severity, event location, incident rate, traffic congestion, and number of ERUs. Additionally we have also considered key operator's characteristics, such as fatigue, workload, operator involvement and studied their impact on emergency management. By using PRISM, we have modeled extensive set of scenarios to capture the effect of these parameters and have provided useful insights.

There are many possible extensions to the proposed model. It can be extended to find the optimal placement of SEMUs in smart cities depending on city parameters such as event distribution, population density, business locations, road congestion/conditions, etc. Another interesting extension to this model is malicious human behavior to induce false alarms in the system, and study robustness of the system in such scenarios. The model can be extended to a game-based strategy where PRISM-games can be used to assess various parameters against each other, thus creating a Pareto curve between two or more optimization variables. Prism-games allow minimizing or maximizing the optimization variables resulting into a Pareto curve that can be used to make an optimal choice among various variables. This can also lead to the design of an efficiency evaluation system for comparing models with different, possibly conflicting, parameters in emergency management system. These extensions will be studied as part of future work.

REFERENCES

[1] T. Nam and T. A. Pardo, "Conceptualizing smart city with dimensions of technology, people, and institutions," in *Proc. 12th Annu. Int. Digit. Government Res. Conf.*, New York, NY, USA: Jun. 2011, pp. 282–291.

[2] (2018). *ITU-T, Focus Group on Smart Sustainable Cities*. Accessed: Apr. 2, 2018. [Online]. Available: https://www.itu.int/en/ITU-T/focusgroups/ssc/Pages/default.aspx

[3] R. Lea, "Smart cities: An overview of the technology trends driving smart cities," *IEEE Advancing Technol. Humanity*, Mar. 2017.

[4] P. Neirotti, A. De Marco, A. C. Cagliano, G. Mangano, and F. Scorrano, "Current trends in Smart City initiatives: Some stylised facts," *Cities*, vol. 38, pp. 25–36, Jun. 2014.

[5] E. Borgia, "The Internet of Things vision: Key features, applications and open issues," *Comput. Commun.*, vol. 54, pp. 1–31, Dec. 2014.

[6] S. Dirks, C. Gurdgiev, and M. Keeling, *Smarter Cities for Smarter Growth: How Cities Can Optimize Their Systems for the Talent-Based Economy*. IBM Institute for Business Value, May 2010. [Online]. Available: https://ssrn.com/abstract=2001907

[7] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, "Internet of Things for smart cities," *IEEE Internet Things J.*, vol. 1, no. 1, pp. 22–32, Feb. 2014.

[8] (2018). *False Alarm*. Accessed: Nov. 20, 2018. [Online]. Available: https://www.merriam-webster.com/dictionary/false20alarm

[9] BRE Group. *Causes of False Fire Alarms in Buildings (Bre Trust Briefing Paper)*. Accessed: Nov. 20, 2018. [Online]. Available: https://www.bre.co.uk/page.jsp?id=3391

[10] S. Philip Schaenman, A. Horvath, and P. Harry Hatry. *Opportunities for Police Cost Savings Without Sacrificing Service Quality: Reducing False Alarms*. Accessed: Nov. 20, 2018. [Online]. Available: https://www.urban.org/research/publication/opportunities-police-cost-sa

[11] F. Saeed, A. Paul, A. Rehman, W. H. Hong, and H. Seo, "Iot-based intelligent modeling of smart home environment for fire prevention and safety," *J. Sens. Actuator Netw.*, vol. 7, no. 1, p. 11, Mar. 2018.

[12] J. A. Stankovic, "Research directions for the Internet of Things," *IEEE Internet Things J.*, vol. 1, no. 1, pp. 3–9, Feb. 2014.

[13] S. Munir, J. A. Stankovic, C.-J. M. Liang, and S. Lin, "Cyber physical system challenges for human-in-the-loop control," in *Proc. 8th Int. Workshop Feedback Comput.*, San Jose, CA, May 2013, pp. 1–4.

[14] J. P. G. Sterbenz, "Smart city and IoT resilience, survivability, and disruption tolerance: Challenges, modelling, and a survey of research opportunities," in *Proc. 9th Int. Workshop Resilient Netw. Design Modeling (RNDM)*, Sep. 2017, pp. 1–6.

[15] V. M. A. De Nicola, M. Melchiori, and M. L. Villani, "M-CREAM: A tool for creative modelling of emergency scenarios in smart cities," in *Proc. Eur. Semantic Web Conf.*, Aug. 2018, pp. 83–88.

[16] S. Wan, J. Lu, P. Fan, and K. B. Letaief, "To smart city: Public safety network design for emergency," *IEEE Access*, vol. 6, pp. 1451–1460, 2018.

[17] G. Rocher, J.-Y. Tigli, and S. Lavirotte, "Probabilistic models toward controlling smart-* environments," *IEEE Access*, vol. 5, pp. 12338–12352, 2017.

[18] H. Ye, "Research on emergency resource scheduling in smart city based on HPSO algorithm," *Int. J. Smart Home*, vol. 9, no. 3, pp. 1–12, 2015.

[19] A. B. Torjusen, H. Abie, E. Paintsil, D. Trcek, and A. Skomedal, "Towards run-time verification of adaptive security for IoT in eHealth," in *Proc. sEur. Conf. Softw. Archit. Workshops*, New York, NY, USA, Aug. 2014, p. 4. [Online]. Available: http://doi.acm.org/10.1145/2642803.2642807

[20] M. Mohsin, M. U. Sardar, O. Hasan, and Z. Anwar, "IoTRiskAnalyzer: A probabilistic model checking based framework for formal risk analytics of the Internet of Things," *IEEE Access*, vol. 5, pp. 5494–5505, 2017.

[21] L. Liang, K. Zheng, Z. Wei, Y. Wang, S. Wu, and X. Huang, "Model checking of IoT system in microgrid," in *Proc. 8th Int. Conf. Inf. Technol. Med. Educ. (ITME)*, Dec. 2016, pp. 601–605.

[22] L. Jin, G. Zhang, and J. Wang, "Probabilistic model checking-based survivability analysis in vehicle-to-vehicle networks," *China Commun.*, vol. 15, no. 1, pp. 118–127, Jan. 2018.

[23] D. Seo *et al.*, "Modeling and verification for different types of system of systems using PRISM," in *Proc. IEEE/ACM 4th Int. Workshop Softw. Eng. Syst.-of-Syst. (SESoS)*, New York, NY, USA, May 2016, pp. 12–18. [Online]. Available: http://doi.acm.org/10.1145/2897829.2897833

[24] L. Li, Z. Jin, G. Li, L. Zheng, and Q. Wei, "Modeling and analyzing the reliability and cost of service composition in the IoT: A probabilistic approach," in *Proc. IEEE 19th Int. Conf. Web Services*, Jun. 2012, pp. 584–591.

[25] M. Kwiatkowska, G. Norman, and D. Parker, "PRISM 4.0: Verification of probabilistic real-time systems," in *Proc. Int. Conf. Comput. Aided Verification*, Jul. 2011, pp. 585–591.

[26] B. Barbot and M. Kwiatkowska, "On quantitative modelling and verification of DNA walker circuits using stochastic Petri nets," in *Proc. Int. Conf. Appl. Theory Petri Nets Concurrency*, Jun. 2015, pp. 1–32.

[27] A. Soeanu, *et al.*, "Transportation risk analysis using probabilistic model checking," *Expert Syst. Appl.*, vol. 42, no. 9, pp. 4410–4421, Jun. 2015.

[28] C. von Essen, B. Jobstmann, D. Parker, and R. Varshneya, "Synthesizing efficient systems in probabilistic environments," *Acta Inform.*, vol. 53, no. 4, pp. 425–457, 2016.

[29] C. Mu, P. Dittrich, D. Parker, and J. E. Rowe, "Formal quantitative analysis of reaction networks using chemical organisation theory," in *Proc. Int. Conf. Comput. Methods Syst. Biol.*. Sep. 2016, pp. 232–251.

[30] Z. Aslanyan, F. Nielson, and D. Parker, "Quantitative verification and synthesis of attack-defence scenarios," in *Proc. IEEE 29th Comput. Secur. Found. Symp. (CSF)*, Jun. 2016, pp. 109–119.

[31] H. Hu, Y. Wen, T.-S. Chua, and X. Li, "Toward scalable systems for big data analytics: A technology tutorial," *IEEE Access*, vol. 2, pp. 652–687, 2014.

[32] L. Sanchez *et al.*, "SmartSantander: IoT experimentation over a smart city testbed," *Comput. Netw.*, vol. 61, pp. 217–238, Mar. 2014.

[33] C. Hultquist, E. Sava, G. Cervone, and N. Waters, "Damage assessment of the urban environment during disasters using volunteered geographic information," in *Big Data for Regional Science*, pp. 443–457, 2017.

[34] L. Feng, C. Wiltsche, L. Humphrey, and U. Topcu, "Synthesis of human-in-the-loop control protocols for autonomous systems," *IEEE Trans. Autom. Sci. Eng.*, vol. 13, no. 2, pp. 450–462, Apr. 2016.

[35] *European Commission Speed Limits*. Accessed: Nov. 20, 2018. [Online]. Available: https://ec.europa.eu/transport/road_safety/specialist/knowledge/speed/speed_limits_en

[36] T. Vranken, B. Sliwa, C. Wietfeld, and M. Schreckenberg. (2018). "Performance comparison of dynamic vehicle routing methods for minimizing the global dwell time in upcoming smart cities." [Online]. Available: https://arxiv.org/abs/1805.09414

[37] (2019). *PRISM Manual*. Accessed: Nov. 20, 2018. [Online]. Available: https://www.prismmodelchecker.org/manual/

[38] H. M. Amer, H. A. Al-Kashoash, A. Kemp, L. Mihaylova, and M. Mayfield, "Coalition game for emergency vehicles re-routing in smart cities," in *Proc. IEEE 10th Sensor Array Multichannel Signal Process. Workshop (SAM)*, Jul. 2018, pp. 306–310.

[39] M. F. N. Maghfiroh, M. Hossain, and S. Hanaoka, "Minimising emergency response time of ambulances through pre-positioning in Dhaka city, Bangladesh," *Int. J. Logistics Res. Appl.*, vol. 21, no. 1, pp. 53–71, 2018.

NAZEERUDDIN MOHAMMAD received the B.E. degree in electronics and communications engineering from Osmania University, India, in 1996, the M.S. degree in systems engineering from the King Fahd University of Petroleum & Minerals (KFUPM), Dhahran, Saudi Arabia, in 1999, and the Ph.D. degree from the School of Computing and Information Engineering, University of Ulster, Coleraine, U.K., in 2007.

He was an IT Infrastructure Manager/Administrator and a Lecturer with KFUPM (KSA), as a Postgraduate Researcher with the University of Ulster, U.K., as a Research Intern with Cisco Systems, USA, and as an IT Infrastructure Manager/Administrator and a Researcher with the Portland House Research Group, Australia. He is currently an Assistant Professor with the College of Computer Engineering and Science, Prince Mohammad Bin Fahd University, Saudi Arabia.

Dr. Mohammad was a recipient of the M.S. Research Scholarship from KFUPM, in 1997, the Vice Chancellors Research Scholarship from the University of Ulster, in 2004, and the first prize in the Faculty of Engineering Business Plan Competition at the University of Ulster, in 2006.

SHAHABUDDIN MUHAMMAD received the B.E. degree from the NED University of Engineering and Technology, Pakistan, in 1999, and the Ph.D. degree from the University of Central Florida, Orlando, USA, in 2007. He has extensive software development experience at various companies in USA, including Adaptec, Orlando, FL, USA, Riverbed Technologies, San Francisco, CA, USA, and Microsoft, Redmond, WA, USA.

He is currently an Assistant Professor with the College of Computer Engineering and Science, Prince Mohammad Bin Fahd University, Saudi Arabia. His research interests include formal verification, sensor networks, security protocols, and the Internet of Things.

ABUL BASHAR received the B.E. degree in electronics & communications engineering from Osmania University, India, in 1995, the M.S. degree in electrical engineering from the King Fahd University of Petroleum & Minerals (KFUPM), Dhahran, Saudi Arabia, in 1999, and the Ph.D. from the School of Computing and Information Engineering, University of Ulster, Coleraine, U.K., in 2011.

He was a Lecturer with the Electrical Engineering Department, KFUPM, Saudi Arabia, as a Postgraduate Researcher with the University of Ulster, U.K., as a Research Intern with British Telecom, U.K., and as an IT Administrator with the Information Technology Center, KFUPM. He is currently an Assistant Professor with the College of Computer Engineering and Science, Prince Mohammad Bin Fahd University, Saudi Arabia.

Dr. Bashar has more than 30 journal and conference publications and actively involved as reviewer for renowned journals (the IEEE TNSM, ACCESS, TRANSACTIONS ON CYBERNETICS). He was a member in TPCs of prestigious international conferences. He was a recipient of the M.S. Research Scholarship from KFUPM, in 1996, the Vice Chancellors Research Scholarship from the University of Ulster, in 2008, and a Gold Medallist from Osmania University, in 1995.

MAJID A. KHAN received the B.S. degree in computer system engineering from the Ghulam Ishaq Khan Institute (GIKI), Pakistan, in 1997, and he Ph.D. degree from the University of Central Florida (UCF), Orlando, USA, in 2007. He has extensive software development experience at various multinational companies, including Net-Sol Technologies, Inc., Lahore, Pakistan, and at Hewlett-Packard (HP Software), Roseville, CA, USA. He has also worked as the main contributor at several start-up companies.

He is currently an Assistant Professor with the College of Computer Engineering and Science, Prince Mohammad Bin Fahd University (PMU). He has research interests include machine learning and distributed/parallel computing with applications in several domains including software engineering, algebraic structures, and modeling of large scale systems.