Received: 2 November 2016 | Revised: 19 February 2017 | Accepted: 13 March 2017
DOI: 10.1002/cpe.4153

WILEY

SPECIAL ISSUE PAPER

An IoT-based monitoring approach for cultural heritage sites: The Matera case

Marco Gribaudo[1] | Mauro Iacono[2,3] | Alexander H. Levis[3]

[1]Dipartimento di Informatica, Elettronica e Bioingegneria, Politecnico di Milano, via Ponzio 34/5 Milan, I-20133, Italy
[2]Dipartimento di Matematica e Fisica, Università degli Studi della Campania "Luigi Vanvitelli", viale Lincoln 5 Caserta, I-81100, Italy
[3]System Architectures Laboratory, George Mason University, 4400 University Drive Fairfax, 22030, VA, USA

Correspondence
Mauro Iacono, Dipartimento di Matematica e Fisica, Università degli Studi della Campania "Luigi Vanvitelli", viale Lincoln 5, I-81100, Caserta (CE), Italy.
Email: mauro.iacono@unicampania.it

Summary

Protection and preservation of cultural heritage is an important responsibility for policy makers, public and private institutions, and the citizens themselves. Technologists can make an important contribution by designing monitoring systems for these sites and using the data to prevent incidents. Internet-of-things technology offers, for a sustainable price and with significant flexibility, a wide range of different possibilities, fitting different circumstances: from monitoring the environmental parameters of a room in a museum to sensing structural changes in a historical building and to protecting vulnerable artifacts. In this paper, we consider the case of monitoring an extended cultural heritage area: a UNESCO protected site, the center of Matera, an Italian town that will be a European Capital of Culture in 2019. This city is a unique historical settlement, as buildings are partially carved into the rock that constitutes the geological substrate of the area, a local practice used since the prehistoric age. The extent and density of these structures makes the physical protection of the site a big challenge when the expected large crowds of tourists arrive. The objective of the proposed system is to anticipate the threats in a timely manner so that appropriate actions are taken by the authorities thus avoiding damage to the cultural heritage sites.
We propose a technique for modeling the performances of the Internet-of-things–based monitoring systems that support the planning of incident management in a protected site by exploiting multiple, sparse, heterogeneous, and partially controlled sensors to monitor the behavior of the crowd. The technique is based on the use of Markovian agent models to study the parameters and the dynamics of a scenario, to understand the needs of the monitoring system.

KEYWORDS
cultural heritage site protection, IoT, Markovian agents, performance evaluation

1 | INTRODUCTION

The introduction of the Internet-of-things (IoT) paradigm allows the use of heterogeneous sensors to monitor vast areas at a low cost. The scalability of the approach depends on the possibility of processing data at a high rate: a wide network of heterogeneous devices provides sensing and status information that aligns to the speed, volume, and variety typical of Big Data systems.

One of the fields of application of IoT is the protection of cultural heritage: controlled spaces (eg, museums) can be monitored with a sensing infrastructure that can be easily installed, monitored, and extended, and data can be easily processed in real time to implement alarms or control of environmental parameters such as temperature and humidity or even to perform a continuous analysis of the state of perishable assets. In these cases, if measurements indicate that parameter values are outside the specified range, access to the rooms may be limited or delayed. Of course, IoT is not the only applicable technology for these situations, but it is probably the most flexible state-of-the-art solution.

When the space considered is a wide and inhabited area, such as a town, and its cultural heritage is to be protected, scenarios grow in diversity and complexity and human behavior can hardly be ignored. In protected historical areas such as Pompei, Venice, or the center of Rome, in which the vastness of the area, the nature of the access paths and the presence of inhabitants prevent or obstruct any kind of access control, then, as documented by news reports, the probability of site damage or theft is significantly higher than 0. In such scenarios there is also a more complex problem: emergency management. In case of incidents (terrorist attacks, natural disasters, public demonstrations, or rallies), there is no direct escape path planning that can be applied, and the intervention of police, firemen, national guard, or medical aid personnel is hard because of the complexity of the terrain and the behavior of the crowd (eg, panic). Festivals or events are the ideal scenario to plan a terrorist attack or the defacement of a cultural heritage site. Such situations require additional monitoring that adapts to the size of the crowds, the nature of the event, and the alarm level.

In this paper, we present a quantitative modeling approach for the planning and management of events in vast, populated cultural heritage sites that relies on adaptive IoT-based monitoring and situation assessment systems in which crowd behavior is included. This approach allows both for the modeling the possible scenarios and the design of the main parameters of the needed computing infrastructure. The main advantage of our approach is its capability of catching, by means of an analytical solution, the dynamics of a high number of model elements with a compact, parameterized description and a low computational cost, with an accurateness that is increasing with the number of agents in the system.

The paper is organized as follows: Section 2 details the motivation for this work; Section 3 presents related work; Section 4 describes the modeling approach; Section 5 shows a case study; and Section 6 closes the paper.

2 | MOTIVATION

Wide protected areas of cultural or natural significance are characterized by the presence of constraints of different kinds. Constraints may be related to the preservation of the landscape, (eg, it is not possible to alter the environment with invasive monitoring systems) or to modify existing artifacts or natural obstacles to facilitate the monitoring activities. Similar constraints concern infrastructures: it may not be possible to install artifacts to support services, such as cables, antennas, and energy supply units thus affecting the integrity of infrastructure facilities. In wider areas, there may be geographical constraints that make it physically impossible or too expensive to implement solutions that require physical connections between 2 different locations. Architectural constraints are a special case; they may prevent for historical reasons small modifications of a building such as the installation of wiring. Some wide protected areas are inhabited; there is a resident population that lives its normal life in and around the cultural heritage sites whose life should not be disturbed by monitoring activities nor disrupted by invasive installations. Such areas are dynamic and evolve slowly over time as contrasted to the case of a museum. Consequently, a monitoring system should be dynamic and able to adapt to sudden changes or temporary different conditions, while not disrupting the life of the residents. It is usually not possible to control accesses to such sites, as the freedom of movement of the local population cannot be restricted and the flow of visitors, customers, guests, and tourists on different time scales should not be impeded. Furthermore, the design of such a system should also respect the privacy of the population as protected by relevant laws (eg, Italy's laws for the protection of citizens privacy).

Consequently, a suitable monitoring system should be capable of reconfiguration and be able to scale easily the extent and density of the coverage and the degree of smartness of its supporting applications, while keeping costs sustainable. The infrastructure of the system should guarantee ease of maintainability and manageability, while satisfying the aforementioned scalability and reconfigurability requirements. Internet of things seems to be capable of providing all these properties, while letting the designer free to choose among a wide range of different devices and to adopt different supporting software solutions to administer, collect, and process data with conventional and Big Data approaches. The standards on which IoT relies ensure that the overall architecture of the monitoring system is open and easily extensible and can be modulated to fit special or temporary needs or to experiment with innovative solutions and new approaches. It is quite simple to find off-the-shelf components and to integrate and coordinate different devices that can be remotely managed (when allowed by laws and opportunity), such as high-quality cameras, existing legacy assets, non-invasive sensors (for common areas such as public places or offices), or specially developed sensors for buildings or art objects in open public spaces. Such an enormous flexibility needs solid guidelines to support the design, deployment, and reconfiguration of the infrastructure. Positioning and choosing the right sensors, and in a sufficient number to guarantee coverage, need proper planning that can be supported by computer simulation. The potential of IoT-based solutions also enables the management of risky and critical situations that may be considered as outliers with respect to normality, such as special events that aggregate large and unpredictable crowds in small spaces with a few exits in case of incidents or accidents. In such cases, the density of the crowd influences the risk level, because of impediments to evacuation or because a panicked crowd behaves in an irrational way. The structure of the monitoring system must adapt to different situations such as including additional sensing devices and the incorporation of additional support personnel that is usually not present in normal operating conditions. Specific solutions for crowd monitoring that add to normal cultural heritage site protection goals as well as incident prevention and safety enforcement goals include some proactive initiatives such as providing free WiFi access to visitors to monitor position and movement of mobile devices or even an event-oriented mobile application. An application, in fact, could be used to gather more detailed data about the crowd and to exercise some influence on it. For example, it can direct part of the crowd to different, less critical areas to reduce crowd density, and it can schedule their access to relevant spots opportunistically to maintain sustainability. Or it can modulate the movement of the crowd over time and get some data that are useful for the management of the situation in advance; or, finally, it can exercise partial control of evacuations and reduce panic by issuing directions towards different exits. The presence of the human crowd influences in a direct and in an indirect way the computing needs of a monitoring system. The direct influence comes from the fact that there is a numerically higher number of possible information contributions to be processed, individually provided by mobile devices or by local sensors. The indirect influence arises from a number of situations: an intensification of data traffic and mutual influence in movements; a need for more accurate algorithms to manage data from collective sensing, such as video data processing to count people or to detect anomalous behaviors of single individuals that may reveal the presence of terrorists or misfits, or sudden illness of individuals; a need for human-in-the-loop monitoring that requires a reconfiguration of the policies and a different timing of the procedures; a need for cross-checking data to ensure correct detection of anomalies and reduce false positives; a need for including new complex devices such as drones to control the situation; and, in general, more complex scenarios, characterized by a different local density in different areas that changes over time (eg, a religious procession) with dynamics that have to be explored and identified by means of on line Big Data solutions. Being able to correctly model the possible behavior of crowds is of paramount importance to provide useful data that guide designers to shaping both the sensing infrastructure and its spatial deployment. The actual system workload depends on the number and type of sensors, the complexity of the scenario of the number and location of sensitive and critical points, the density of the population, and the criticality of active threats.

3 | RELATED WORKS

3.1 | Internet of things and cultural heritage

Internet of things is a technology that enables wireless sensor networks to get integrated into standard based communicating-actuating networks that seamlessly blend into the environment[1] creating the so called ambient intelligence.[2] Smart connectivity capabilities and context-aware computation[3] are the main characteristics of IoT that allow an evolution of the computation paradigm with respect to traditional computation. Ubiquity of sensing and integration with high performance computing facilities (such as cloud-based infrastructures) provide the basis for the implementation of scalable and flexible applications: for a survey on enabling technologies the reader can refer to Al-Fuqaha et al [4] and Li et al[5]: ubiquity may be augmented by interactions with humans by means of mobile devices, in the so called opportunistic IoT paradigm. [6] Internet-of-things systems should cope with a large variety of different problems[7]: integration of heterogeneous devices, scalability, ubiquitous data exchange through proximity wireless technologies, localization and tracking capabilities, self-organization capabilities and resource discovery,[8] embedded security and privacy-preserving mechanisms, semantic interoperability, and data management energy-optimized solutions (that is anyway strictly connected to the intrinsic wireless sensor network nature of the infrastructure[9]). The fitness of IoT devices for solutions for the physical protection of places of interest has been demonstrated in literature (eg, see Chilipirea et al [10]).

At a higher level, IoT is one of the key paradigms that support Smart Cities[11–13]; it allows for a systemic instrumentation of whole cities for smart applications that include monitoring of buildings, traffic, and environmental parameters. On these large scales, security and coordination problems arise for large peer-to-peer networks, such as the ones that support IoT sensing networks.[14,15]

Of course, of paramount importance in the paradigm is data management: a data centric perspective on IoT is presented in Aggarwal et al.[16] When the scale of the system increases (eg, in Smart Cities applications[17]), data mining techniques are needed to exploit collected data,[18] while data management can require Big Data methods.[19]

Internet-of-things technologies provide a flexible solution for cultural heritage–related management and protection applications both at small- (eg, smart museums,[20–23] iotbd16) and large-scale (eg, smart tourism,[24] smart cities, and smart regions[25]), which may be used to understand visitors' behavior and interests.[26]

3.2 | Markovian agents

In this paper, we model the crowd and the other elements of the scenario by means of Markovian agents (MAs),[27,28] a quantitative formalism that allows modeling of behavior and interactions of complex agents on a stochastic basis. Markovian agent–based models can seamlessly scale up to millions of agents, provided that they can be classified into classes with similar elementary behaviors, and provide more accurate results when the number of agents per class increases. Moreover, MA behaviors may be influenced by their relative positions in a geographic area, described as a graph with given topological characteristics, and by mutual interactions that can happen on a local or a global scale, whether based on mutual interactions or broadcast logic. A model consists of a collection of autonomously evolving agents in a given generically defined space (eg, a map) that has a graph structure and in which agents interact (between each other and with the location) by exchanging abstract messages. An agent belongs to a class; each class is described by a finite state automaton-like stochastic model (in this paper, a colored Petri nets (CPN) description is adopted to simplify the presentation and to exploit a high level representation), with an additional message propagation function matrix $\pi(d, l, s, m, t)_{x,y}$ that states how a message of type t generated in state s of an agent of class c_x located at m may be accepted by an agent of class c_y located at l in state d. Messages influence automata: they may evolve from one state to another state by an induced transition (instead of a local transition) that only depends on its internal conditions and its local environment; eventually, a transition may cause the production of a new message.

From the quantitative point of view, the evolution of an agent is evaluated by generating its finite-state continuous-time homogeneous Markov chain, evolving according its specific transition matrix. A single agent class that in a given instant is located in v is fully described by a 5-tuple $MA(v) = (Q(v), \Lambda(v), P(v), A(v), \pi_0(v))$, with $Q(v)$ being the infinitesimal generator matrix of the Markov chain, matrix $\Lambda(v)$ describing the transition rates between the states, $P(v)$ being the probability of generating a message, $A(v)$ the probability of accepting a message, and $\pi_0(v)$ the probability vector in the initial state. The spatial distribution of agents over a space S (continuous or discrete and finite) is described by $\delta : S \rightarrow \Re^+$, and agents are distributed according to a Poisson distribution in each subspace $U \subset S$, with mean $\int \int_U \delta(x)dx$.

Markovian agent have been used for performance evaluation in different application fields, eg, behavior of massively distributed computer architectures,[29,30] the propagation of seismic waves,[31] and the dynamics of cancer cells.[32]

3.3 | Modeling a crowd

For the purposes of this paper, a sound framework for the evaluation of crowd movements and behavior is needed. A sound presentation of related works in the field, together with a very good presentation of the main problems and some important results is in Helbing et al,[33] that we suggest to interested readers and that relies on a description of spatio-temporal patterns in pedestrian crowds as self-organized phenomena that can be modeled as particle in molecular dynamics. The approach considered in this paper is inspired by a multiagent-based model described in Almeida et al. [34] According to the authors, pedestrians, in normal conditions, always try to keep their path the shortest and the easiest possible, avoiding detours even in crowds. In panic or emergency or rush situations, pedestrians change behavior and produce characteristic formations that result from the speed increase and the concentrations in the same paths. In such situations, pedestrians lose their ability for finding an optimal path and tend to reuse known paths nothwithstanding the different conditions (eg, they try and exit from the entrance they previously used), to exhibit herding or flocking behavior, to cause stampedes, to start pushing each other, or to create a dense arch of people in front of doors, blocking in fact the passage, ie, creating a bottleneck effect.

4 | MODELING APPROACH

We describe our scenario using a MAs-based[27] model. In particular, we use 5 different agent classes representing, respectively: visitors, attackers, rescuers, cultural heritage sites, and IoT-based sensing devices. The first 3 agent classes are dynamic in the sense that they move across the territory. The last two are static since they do not change location during the evolution of the scenario.

4.1 | Agent description formalism

Agents behavior is described using a variation of the CPN formalism,[35] whose primitives are summarized in Figure 1. Tokens might be either standard (ie, indistinguishable markers) or colored: the latter are characterized by attributes selected from a color set $< S >$. Places that contain colored tokens are labeled with the corresponding color set. Transitions can be of 3 different types: timed, event, and synchronizing. All transitions are enabled when their input places (ie, places to which they are connected with input arcs) have enough tokens (of the specific color, if required, as specified by the arc inscriptions). When a transition fires, it it removes the token from the input places that enabled the firing and generates tokens (of a given color, if required, as specified by the arc inscriptions) in the output places (ie, places to which they are connected with output arcs). When specific colors are involved, arcs are inscribed with the required attributes. Note that the color associated with an output arc might be computed with a function.

The exact moment in which an enabled transition fires depends on its type. Timed transitions fire after a specified (random) delay. Event transitions fire as soon as the event specified by the associated guard function becomes true. Synchronizing transitions fire together with the event to which they are synchronized. In particular, each transition of this type has associated a label that corresponds to the name given to a transition of the 2 other types. As soon as a timed or event transition with the matching name fires, the corresponding synchronizing transitions are enabled.

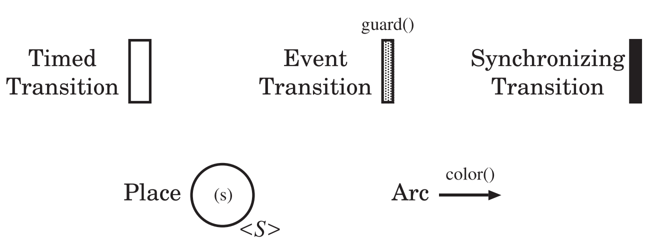

FIGURE 1 The colored Petri nets formalism used to describe the agents' behavior

Note that the main contribution of this work is not the definition of a new formalism to describe the behavior of MAs: the proposed CPN dialect is used just to give a graphical description of the behavior of the considered agent classes.

4.2 | Model description

In the proposed model, we use 2 color sets, as summarized in Table 1. In particular, color set $S = \{s_1, \ldots, s_N\}$ is the discrete set containing N cultural heritage points considered in the model. We also use the integer color set $L = \{1, \ldots L_{max}\}$ to denote the danger levels of a particular attacked site.

Visitors are modeled with the agent shown in Figure 2. They start visiting a site s_i, that corresponds to the initial marking of place `Visiting`. As soon as they end their visit, timed transition `End` fires. The firing time of this transition can depend on both the choice of site s_i, and on the crowd currently visiting that site. Function *chV*() determines the next site to be visited, and the transfer in modeled by the marking of place `Moving`. The target is reached when the guard function *at*(*s*) associated with the event transition `Reaching` evaluates as true.

During the visit to 1 site, an attack can occur: this is modeled by the firing of the synchronizing transition `Event`. To better show the effect of synchronizing events, small places, represented with dashed lines, are used to emphasize the possibility of this transition of firing because of the occurrence of external events that insert a token in the dashed places. This convention will be used throughout the paper. If the event occurs, visitors start to leave the current site s_i by moving the token to place `Escaping`. Two events can occur to a visitor in this circum-

TABLE 1 Color sets

Color Set	Description
$< S >$	Cultural heritage sites
$< L >$	Danger level (1 … L_{max})

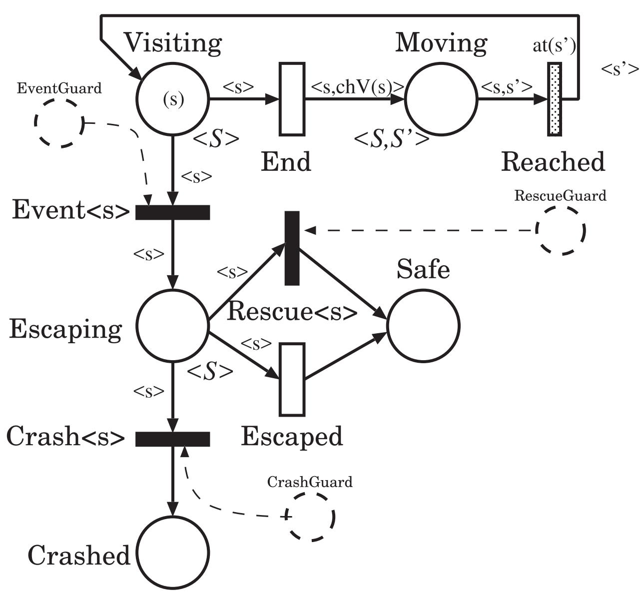

FIGURE 2 The visitor agent

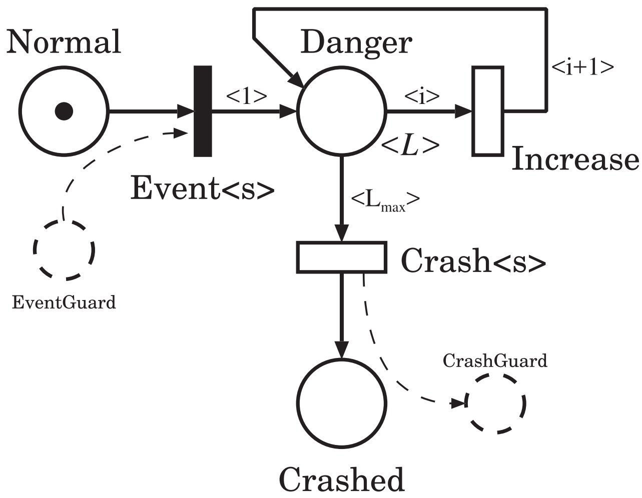

FIGURE 3 The site agent

stance: (1) the visitor can be rescued by a rescuer (event denoted by the firing of the synchronizing transition `Rescue`), or (2) the visitor can independently manage to escape (firing of timed transition `Escaped`, whose average firing time is again conditioned by both the considered site s_i and the current crowd visiting the same location). Both events will lead the visitor to marking the place `Safe`, denoting a successful escape. However, should the site s_i collapse before the visitor is rescued or escapes, the synchronizing transition `Crash` will fire and a token will be generated in place `Crashed`.

Sites are modeled by the agent shown in Figure 3. They are in normal state (denoted by the marking of place `Normal`), unless an attacker succeeds in his attack (modeled by the firing of synchronizing transition `Event`). In this case, place `Danger` becomes marked with a colored token from the color set L that denotes the danger level of the situation. The danger level can increase by the firing of the timed transition Increase until the maximum level L_{max} is reached. At this point the site collapses by the firing of the timed transition `Crash` and the marking of place `Crashed`.

The behavior of the attacker is modeled by the CPN shown in Figure 4. Specifically, the attacker starts waiting for the best moment to attack, as denoted by the marking of place `Waiting`. The decision of starting the attack is denoted by the firing of time transition `Act`, and the selection of the target site is performed by function *chA*() that specifies the token s_i of the color set $< S >$. Motion is described, as for the visitor case, by place `Moving` and event transition `Reached`. When the target site s_i has been reached, place `Preparing` becomes marked and the threat is realized with the firing of timed transition `Event`. While the attacker is moving and preparing to realize the threat, the attacker might be stopped by a rescuer, by firing of the synchronizing transition `Catch` and the consequent marking of place `Caught`.

Sensor agents behaves as shown in Figure 5. They continuously monitor the relevant area (when place `Monitoring` is marked), and they can detect an attacker preparing his attack. In particular, detection occurs when event transition `Detect` becomes enabled because of the guard function *spot*() evaluating as true. In this case, place `Calling` becomes marked and, after a signaling delay, a rescuer becomes notified by the firing of timed transition `Call`.

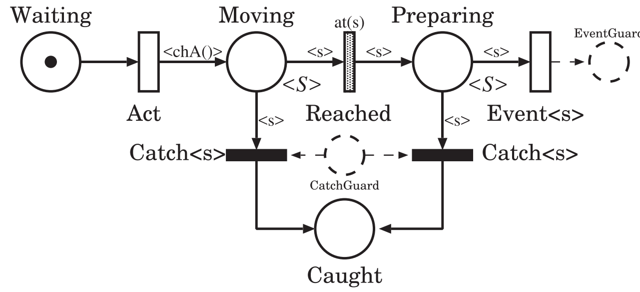

FIGURE 4 The attacker agent

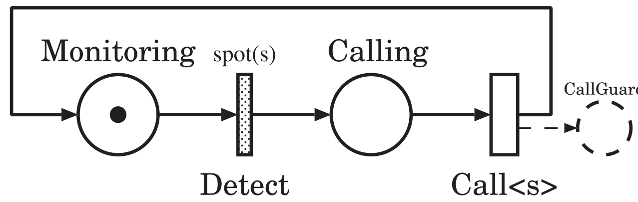

FIGURE 5 The sensor agent

Figure 6 shows the model of the rescuer agents. When place `Patrolling` is marked, rescuers are defending the site denoted by the attribute *s* associated with the corresponding token. During patrolling, they can discover an attacker, thanks to the firing of the timed transition `Catch`. Agents can move to another site s_j for the occurrence of any of 3 possible events. They can autonomously decide to move to another location: this is denoted by the firing of time transition *Next*, and the next site is decided by function *chR*(). The rescuer might be called by a signal produced by a sensor: this is denoted by the firing of the synchronizing transition `Call`. They might also be attracted to a site where an attacker has succeeded realizing the threat, as represented by the firing of synchronizing transition `Event`. Motion is modeled using the `Moving` place and the `Reached` transition as was done for visitors and attacker agents. A response to a call and identification of an attacker can also happen when a rescuer is moving from 1 site to another due to the `Catch` and `Call` transitions, both having as output place the `Moving` place. During rescue operations, a rescuer can successfully save some visitors (firing of timed transition `Rescue`); the intervention ends if the site collapses. This is denoted by the firing of synchronizing transition `Crash` and the marking of place `Crashed`.

Tables 2 and 3 summarize, respectively, the events that lead to synchronizations together with the agents that cause them, and the functions used to both define guards for event transitions and color selection in arc inscriptions.

Agents are distributed over a graph ρ that represents the paths (roads, streets, and steps) where visitors, rescuers, and attackers can move. Each segment of the graph is characterized by a velocity function $v_a(\rho, x)$ that defines the speed at which an agent of class $a \in \{$Visitor,Attacker,Rescuer$\}$ moves in a point x of the road, depending on the density of agents on the graph ρ. In the scenario, there is a finite set of cultural heritage sites positioned in specific points on the graph, and a finite set of sensing devices, strategically placed over the territory. A simple topology with 12 interconnection roads, 4 cultural heritage sites, and 10 sensors is depicted in Figure 7.

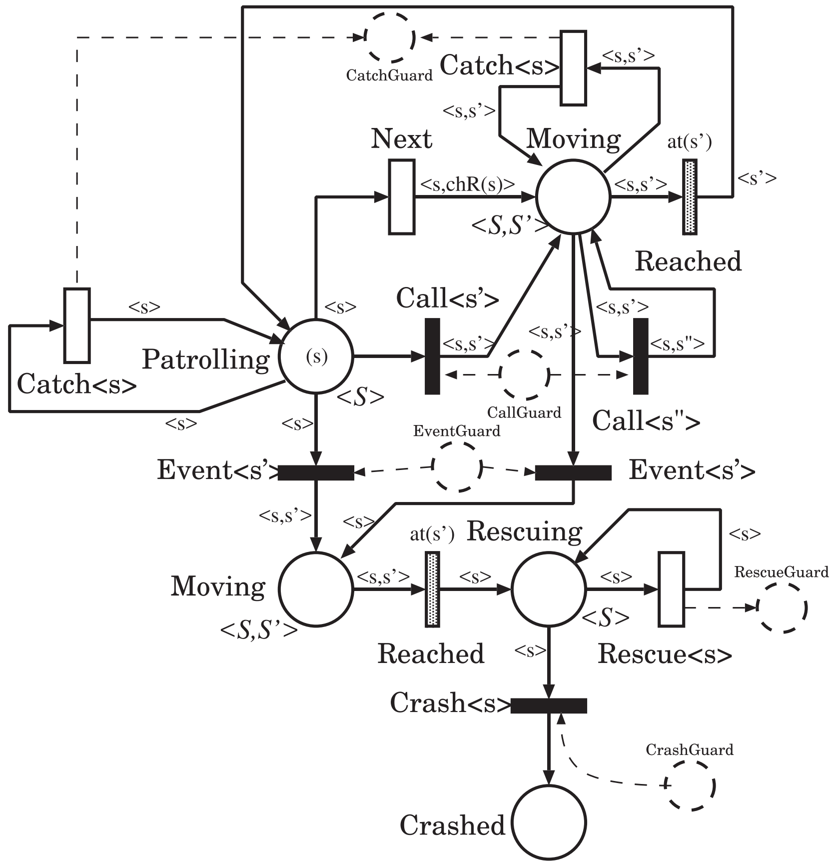

FIGURE 6 The rescuer agent

TABLE 2 Synchronizing events

Event	Generated by	Description
Event< *s* >	Attacker	The attack has success at location *s*
Crash< *s* >	Site	The site at location *s* collapses
Rescue< *s* >	Rescuer	Rescuer at location *s* performs a successful rescue
Catch< *s* >	Rescuer	Rescuer catches attacker at location *s*
Call< *s* >	Sensor	Sensor detects an attacker at location *s*

TABLE 3 Functions and guards

Function	Description
`chV()`	New destination for a visitor
`chA()`	Target location of an attack
`chR()`	Next area to be patrolled by a rescuer
`at(s)`	Agent reaches location *s*
`spot(s)`	Sensor spots an attacker

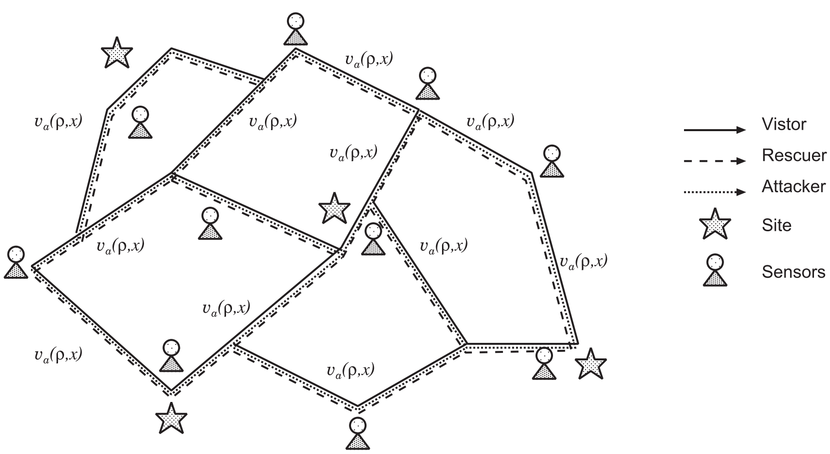

FIGURE 7 The environment

TABLE 4 Functions and guards

ID	Site	ID	Site
1	Cattedrale di Matera	A	Chiese rupestri di San Nicola dei Greci
2	Museo della fotografia	B	Museo Laboratorio della Civilt Contadina
3	Fondazione SoutHeritage	C	Casa Noha
4	Sassi in Miniatura	D	Fondazione Zetema
5	Le Grandi Mostre nei Sassi	E	MUSMA - Museo della scultura contemporanea
6	La Scaletta Matera	F	Conservatorio statale di Musica
7	S. Giorgio al Paradiso - Sassi	G	Palazzo Bernardini
8	Piazza del Sedile	H	Mastropi-Teatro delle Ombre
9	Conservatorio Statale di Musica ER Duni	I	Polizia di Stato - Questura
10	Chiesa di S.Francesco D'Assisi	II	Polizia di Stato - Compartimento Polizia
11	Casa Ortega - Museo delle Arti applicate	III	Carabinieri - Comando provinciale
12	Casa D'imperio	IV	Polizia Stradale
		V	Vigili del Fuoco - Comando Provinciale
		VI	Vigili del Fuoco

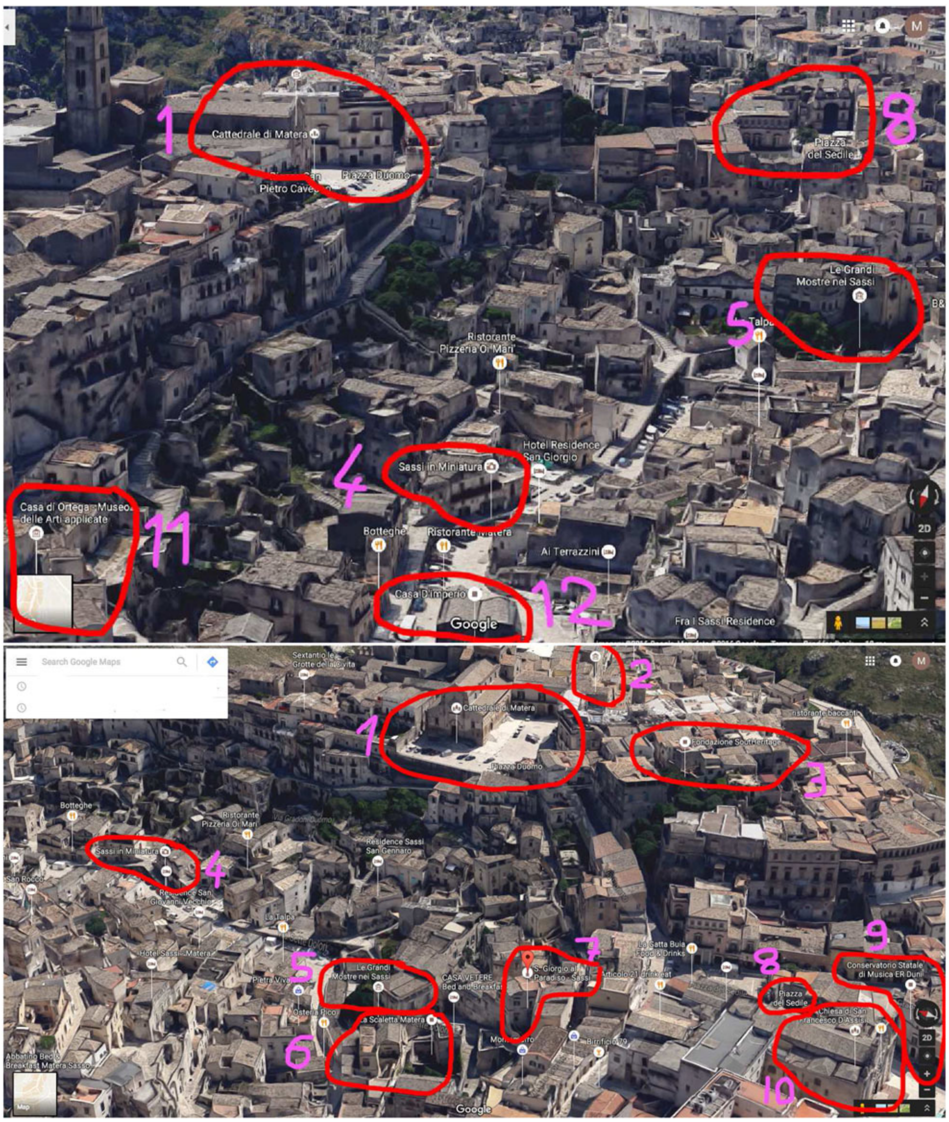

FIGURE 8 Two views of the city of Matera, with the position of some sites emphasized

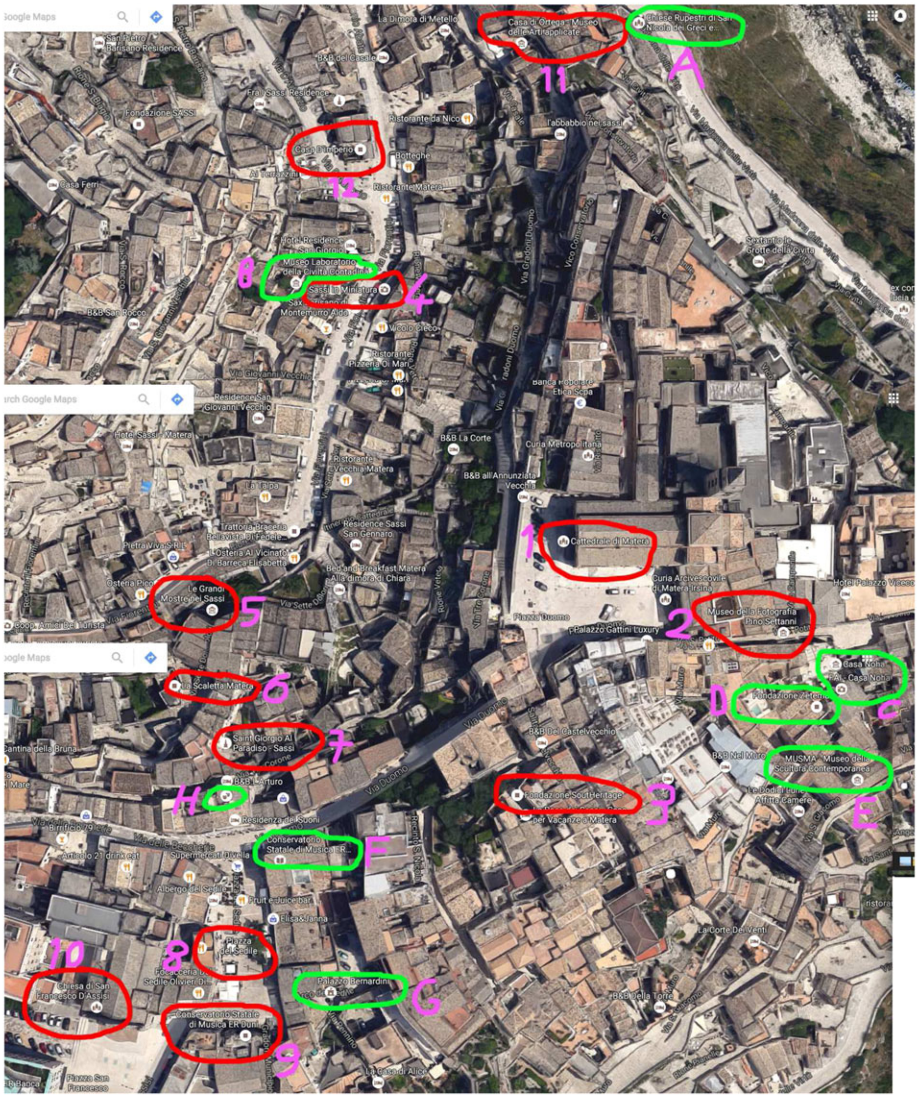

FIGURE 9 The graph of the scenario superposed over the map

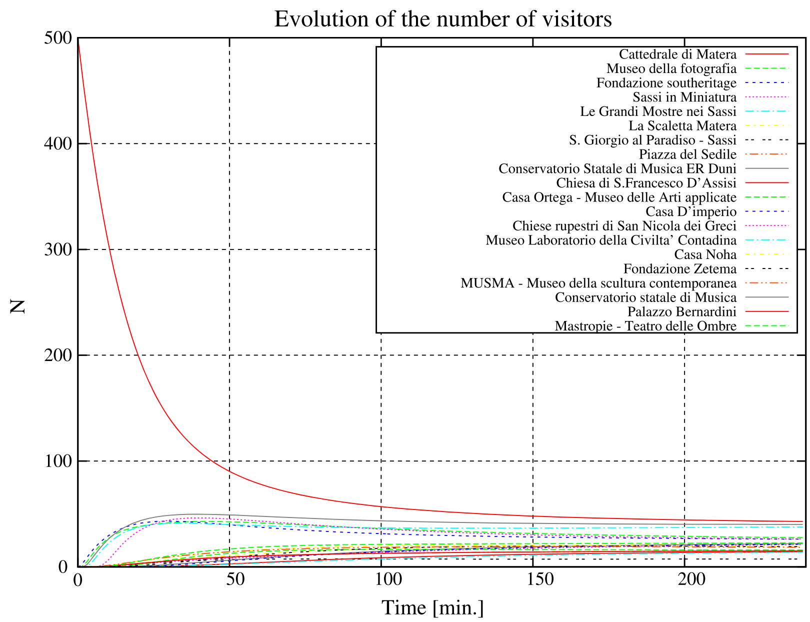

FIGURE 10 Evolution of the number of visitors in the first scenario

4.3 | Modeling IoT

The focus of this paper is on the use of IoT for the protection of cultural heritage area. Internet of things enters in the model in the following ways:

- Sensor agents represents typical IoT devices which can be surveillance cameras, ambient monitoring device, microphones, etc. The associated *spot*() function defines the way in which they can identify a possible threat.
- The level of danger of a site (modeled by the attribute of marking of place `Danger` of the corresponding agent) can evolve at different speeds depending on the characteristic of IoT devices equipped with sensors and actuators that can automatically perform safety actions.
- The ability of rescuers to perform `Catch` and `Rescue` actions (expressed by the corresponding firing time distribution) can be influenced by the IoT devices that they can exploit during their mission.
- The selection of the next site to be reached by visitors and rescuers, modeled respectively by color assignment functions *chV*() and *chR*() can reflect the use of smart applications that help in selecting the best choice depending on the current state of the whole scenario.
- The speed at which agents can move $v_a(\rho, x)$ can be influenced by IoT devices equipped in their means of transportation (that could include bikes, push scooters, segways, etc).

4.4 | Solution technique

The models are analyzed by a mixed technique that includes both standard discrete event simulation and MA mean-field approximation. In particular, static agents (cultural heritage sites and IoT sensing devices) and attackers are modeled with discrete event simulation since they compose a small population of the model. Visitors and rescuers are instead modeled using the mean-field solution approach typical of MA models. This allows to avoid the problems that mean-field models present when considering limited population, while allowing to consider large populations of visitors and rescuers that would be impossible to manage with discrete event simulation alone.

5 | A CASE STUDY: MATERA

The city of Matera, Italy, is a very peculiar place, characterized by limited accessibility. Popular traveling blog sites, such as "Never Ending Voyage" defines the place as "the most spectacular city in Italy."*

Most of the city of Matera is carved into the stone, and several public places, like hotels and restaurants are located in caves. Buildings climb up and down the hillside and houses are piled on top of each other, with the roofs of some acting as streets for other. In particular, we focus on the historical center of the city, where we identify a set of important locations that we model with the agent sites presented in Figure 3. In particular, Table 4 summarizes the main sites defined in the scenario: they include both cultural heritage sites (1-12) and (A-H), among which visitor agents move, and rescuers bases (I-VI). Figure 8 shows a couple of views of the city with the 3D function of the satellite vision from Google Maps. We have also identified a set of possible interconnecting roads that are travelled on foot inside the historical center of the town, and where rescuers can move by faster means in the area outside to reach the cultural heritage sites more quickly. Figure 9 shows the graph superposed on the satellite view of the area.

5.1 | Scenarios

We present 2 scenarios to show the potential of the approach. The first scenario describes an ordinary situation, in which, after a religious service in the main church on the center, that is the Cathedral of Matera (Cattedrale di Matera, #1 in Figure 8), people leave the place and go back to their occupations. The goal of this scenario is to show how people behave spontaneously, with no menace, to provide a baseline reference for the next scenario. In this scenario, we hypothesize that 500 people leave the church at the same time, in different directions. The traveling speed has been set to match the one given by Google Map when considering the best route that connects each of the 2 places at the opposite ends of an edge of the topology graph.

Figure 10 shows the evolution of the number of persons in the different relevant locations that are instrumented by means of sensors, as time passes. In this case, agents move along the map, and visit the considered sites. Each visit has an average duration of 20 minutes; after this, each visitor agent moves randomly to one of the closest neighboring sites. The figure shows that some locations have a peak in the number of visitors, as they are along the path towards other locations, that then decreases to a lower and almost constant value. The evolution of every single curve on the graph depends on the walking distance, in minutes, of each location from the main church. The final different number of visitors in the various locations depends on its position with respect to the others; it shows that some location might become more popular than others because they may be closer to some other very visited site. Future extensions of the work will consider a variable visiting time to match the expected number of visitors of each site. Figure 11 shows the detail of the dynamics of the visitors' movements, to let the reader appreciate better the evolution.

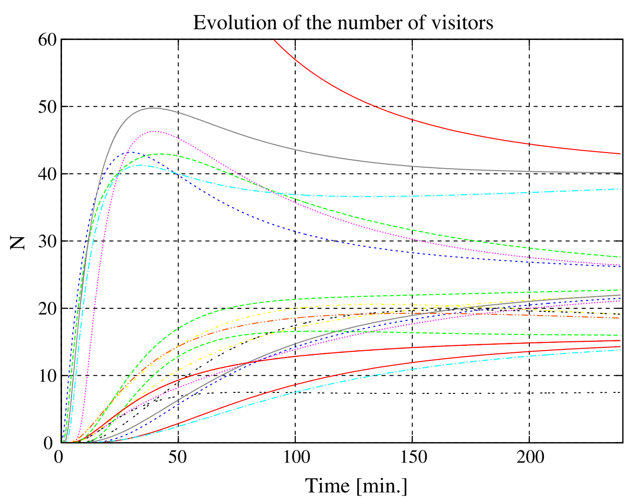

FIGURE 11 Detail of the evolution of the number of visitors in the first scenario. The key of the plot is the same as in Figure 10

*http://www.neverendingvoyage.com/sassi-matera-italy/

FIGURE 12 Evolution of visitor agents on the map. The area of the circles is proportional to the number of visitors in 1 site. The width of the lines of the arrows is proportional to the number of visitors on the roads

The movement of the visitors is shown on the map in Figure 12. In particular, the number of visitors in a site is represented by a circle, whose area is proportional to the number of people at that site. The arrows that interconnect the sites represent the moving visitors from 1 place to another. In particular, the width of the line that connects 2 sites is proportional to the number of visitors in that particular section of the path. As it can be seen in Figure 12A, at time $T = 30$ seconds. Most of the visitors are at the cathedral, but roads immediately near the church start to be populated. After 5 minutes (Figure 12B), most of the people is still at the cathedral, even if some of the neighbor sites starts to become visited. After 15 minutes (Figure 12C), even some of the sites that are a little bit further away from the cathedral start to become visited. It is interesting to see how the number of people on the road reduces from $T = 5$ minutes to $T = 15$ minutes; this is because many visitors, after crowding the streets outside the church at the end of the celebration, start to enjoy different sites, reducing the quantity of people on the road. Figure 12D shows instead the steady state of the system, 2 hours after the celebration, when visitors have decided autonomously to continue their visit to the city in different ways.

In the second scenario, there is not a crisis, but an unexpected event that does not include the presence of a crowd. Because Matera will be the cultural capital of Europe in 2019, its museums have asked

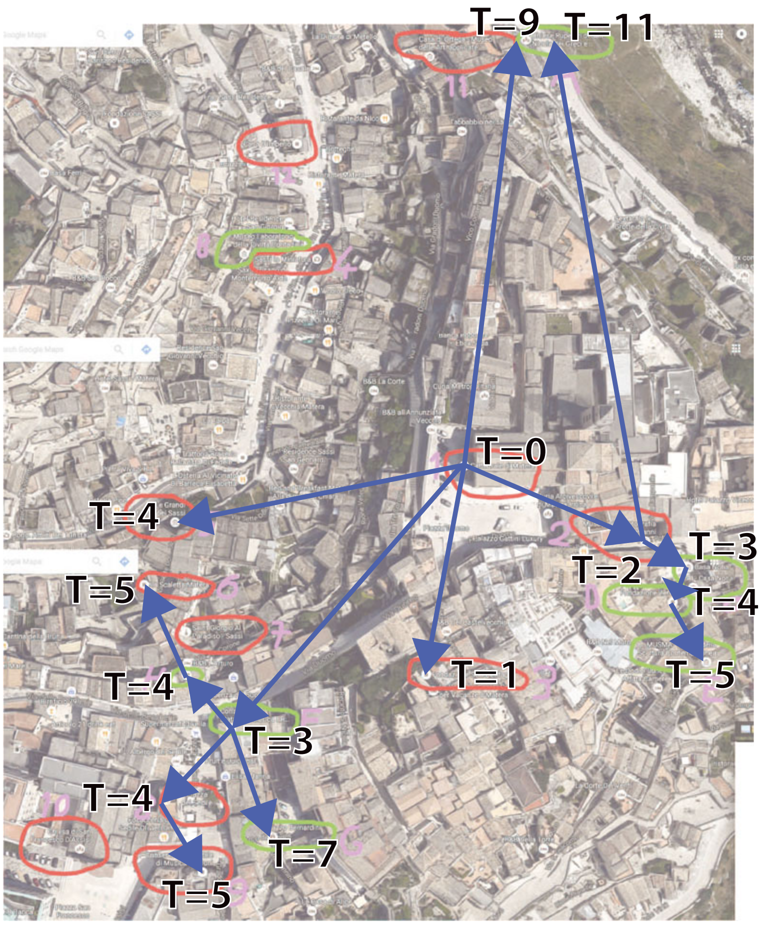

FIGURE 13 Possible escape routes of the attackers

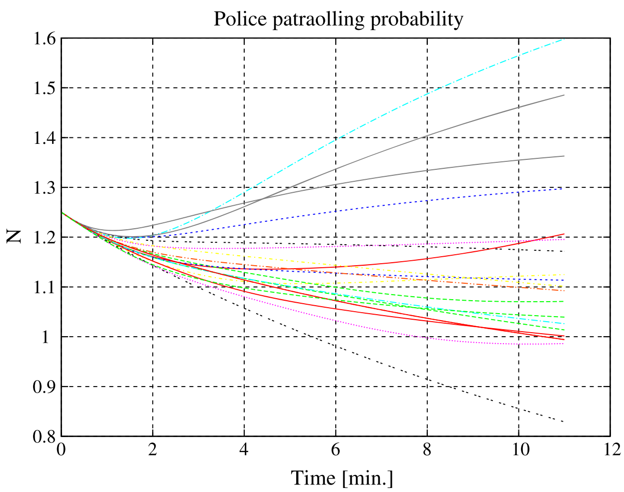

FIGURE 14 Patrolling route probability of the rescuer. The key of the plot is the same as in Figure 10

for help from other museums in Italy and Europe to bring in for the year Matera-related art that they have. Indeed, some museums have responded and a substantial number of ancient sculpture and artifacts has come to Matera. Included is a famous painting of the Madonna dell'Antani, that is to be exhibited in the cathedral. This has motivated a gang of art thieves (or attachers) to plan a heist under the assumption that Materas cathedral museums will not have all the protection that major museums have. The thieves are targeting the cathedral. They plan to do the heist at night on the Saturday before the Sunday celebrations on January 6. What they have not accounted for is that the sensor fusion center personnel is doing a last minute check of all the systems before the grand opening the next day. One operator notices a car on Piazza Duomo even though it has been decreed that no cars should be there in anticipation of the Sunday morning crowd. The operator alerts the center leadership. They focus the sensors on the car and the activity at the cathedral while they notify police and Carabinieri, who send emergency response personnel to the area. However, because of the inaccessibility of the area, the thieves are able to remove the painting and start the car. The sensors track the car and the police apprehend them shortly thereafter.

Figure 13 shows the possible escape routes that might be tried by the thieves. In particular, arrows indicate the patterns they can use to leave from the cathedral. The timing information defines when the attacker will reach a specific point during his escape. The IoT sensors have to check the routes followed by the attacker and direct the police and other forces toward the area.

Without being guided by the IoT, the police will follow a predefined patrolling route. Figure 14 shows the average number of agents in each location, starting from an uniform distribution over all the places and following a pseudo random route intended to reach each possible

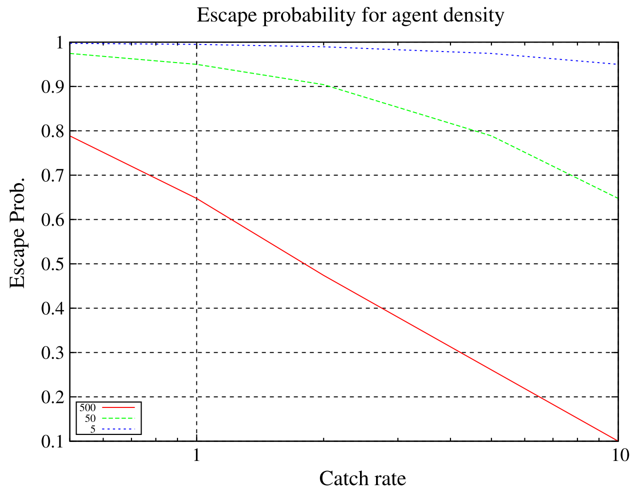

FIGURE 15 Escaping probability as function of the agent density and amount of focusing over a suspect

spot in a even but unpredictable way. The plot focuses on a period of 11 minutes, the maximum time the attacker might require to leave the area along the slowest route.

In our model, an attacker is caught by the firing of transition `Catch` in both the rescuer (Figure 6) and attacker (Figure 4) agents. The IoT can increase the probability of firing of this transition in 2 possible ways. Firstly it can increase the focus of the policemen over the suspect making easier for them to spot the escaping target. In our model, this corresponds to the firing rate for transition `Catch` for the rescuer. Secondly, it can increase the number of rescuer agent in the area where the attacker is located by firing the `Call` transition and force the agent to move in the proper location. Figure 15 shows the effects on the probability of the attacker to successfully escape. In particular, the horizontal axis shows the rate of transition `Catch`: its value, measured in events per minutes, characterizes a Poisson process that models the occurrence of a successful apprehension of an attacker by a rescuer. The 3 different lines, named, respectively, as 5, 50, and 500, describe the equivalent agent density that can be reached by the IoT alerting the rescuers. For example, the value 50 describes the case in which the IoT can bring to the places where the attacker has been spotted a number of rescuer agents that would be otherwise obtained with 50 agents randomly patrolling the area, without IoT network in place. As expected, the probability of catching the attacker increases (thus, the escape probability shown in Figure 15 decreases) by either increasing the catch rate or the equivalent number of agents. It also shows that it seems to be more effective to increase the capability of the IoT network to direct the rescuers in the correct place where the attacker is moving, rather than increase the ability of focusing on the right suspects.

6 | CONCLUSIONS

In this paper, we proposed a study for a system in which IoT devices are used to monitor and protect a cultural heritage site such as the city of Matera, in Italy. The main feature of this work has been the use of a single model that is able to capture, at an high level, a large number of interacting entities of different types: IoT devices, visitors, sites, attackers, protectors, spatial topology, and interconnecting networks. Although many components are greatly simplified, they could still allow us to study some realistic scenarios and use them to estimate the performance and properly size and guide the deployment/upgrade of a IoT infrastructure.

The presented results show the impact of the application of the use and of a proper deployment of IoT devices to support the design and the implementation of protection strategies and actions in defense of cultural heritage and relevant sites. The approach is very flexible and provides the specialists with qualitative and quantitative evaluation means of the effectiveness of the designed protection policies.

Future works will concentrate on testing different, more advanced, attack patterns in different locations. The approach will also be improved by integrating it into multiformalism and multisolution modeling methodologies, to simplify the description of the considered scenario and improve the usability of the modeling technique for specialists that are not experts of modeling and performance evaluation.

REFERENCES

1. Gubbi J, Buyya R, Marusic S, Palaniswami M. Internet of things (IoT): a vision, architectural elements, and future directions. *Future Gener Comput Syst*. 2013;29(7):1645–1660. Including Special sections: Cyber-enabled Distributed Computing for Ubiquitous Cloud and Network Services and Cloud Computing and Scientific Applications - Big Data, Scalable Analytics, and Beyond.
2. Whitmore A, Agarwal A, Da Xu L. The internet of things—a survey of topics and trends. *Inf Syst Front*. 2015;17(2):261–274.
3. Perera C, Zaslavsky A, Christen P, Georgakopoulos D. Context aware computing for the internet of things: A survey. *IEEE Commun Surv Tutorials*. 2014;16(1):414–454.
4. Al-Fuqaha A, Guizani M, Mohammadi M, Aledhari M, Ayyash M. Internet of things: a survey on enabling technologies, protocols, and applications. *IEEE Commun Surv Tutorials*. 2015;17(4):2347–2376.
5. Li S, Xu LD, Zhao S. The internet of things: a survey. *Inf Syst Frontiers*. 2015;17(2):243–259.
6. Guo B, Zhang D, Wang Z, Yu Z, Zhou X. Opportunistic IoT: exploring the harmonious interaction between human and the internet of things. *J Network Comput Appl*. 2013;36(6):1531–1539.
7. Miorandi D, Sicari S, Pellegrini FD, Chlamtac I. Internet of things: Vision, applications and research challenges. *Ad Hoc Networks*. 2012;10(7):1497–1516.
8. Palmieri F. Scalable service discovery in ubiquitous and pervasive computing architectures: a percolation-driven approach. *Future Gener Comput Syst*. 2013;29(3):693–703.
9. D'Arienzo M, Iacono M, Marrone S, Nardone R. Estimation of the energy consumption of mobile sensors in WSN environmental monitoring applications. *Advanced Information Networking and Applications Workshops (WAINA), 2013 27th International Conference on*, Barcelona, Spain; 2013:1588–1593.
10. Chilipirea C, Ursache A, Popa DO, Pop F. Energy efficiency and robustness for IoT: building a smart home security system. *2016 IEEE 12th International Conference on Intelligent Computer Communication and Processing (ICCP)*, Cluj-Napoca, Romania; 2016:43–48.
11. Perera C, Zaslavsky A, Christen P, Georgakopoulos D. Sensing as a service model for smart cities supported by internet of things. *Trans Emerg Telecommun Technol*. 2014;25(1):81–93.
12. Zanella A, Bui N, Castellani A, Vangelista L, Zorzi M. Internet of things for smart cities. *IEEE Internet Things J*. 2014;1(1):22–32.
13. Hancke GP, Silva BdC, Hancke GP, Jr. The role of advanced sensing in smart cities. *Sensors*. 2013;13(1):393.

14. Visan A, Pop F, Cristea V. Decentralized trust management in peer-to-peer systems. *2011 10th Int Symp parallel distributed comput*, Cluj-Napoca, Romania; 2011:232–239.
15. Ghit B, Pop F, Cristea V. Epidemic-style global load monitoring in large-scale overlay networks. *2010 International Conference on P2P, Parallel, Grid, Cloud and Internet Computing*, Fukuoka, Japan; 2010:393–398.
16. Aggarwal CC, Ashish N, Sheth A. *The Internet of things: A Survey from the Data-Centric Perspective*. US: Boston, MA: Springer; 2013;383–428.
17. Jin J, Gubbi J, Marusic S, Palaniswami M. An information framework for creating a smart city through internet of things. *IEEE Internet of Things J*. 2014;1(2):112–121.
18. Tsai CW, Lai CF, Chiang MC, Yang LT. Data mining for internet of things: a survey. *IEEE Commun Surv Tutorials*. 2014;16(1):77–97.
19. Chen M, Mao S, Liu Y. Big data: a survey. *Mobile Networks Appl*. 2014;19(2):171–209.
20. Chianese A, Piccialli F. Designing a smart museum: when cultural heritage joins IoT. *2014 Eighth International Conference on Next Generation Mobile Apps, Services and Technologies*, University of Oxford; 2014:300–306.
21. Alletto S, Cucchiara R, Fiore GD, Mainetti L, Mighali V, Patrono L, Serra G. An indoor location-aware system for an IoT-based smart museum. *IEEE Internet of Things J*. 2016;3(2):244–253.
22. Chianese A, Piccialli F. *Improving user Experience of Cultural Environment Through IoT: The Beauty or the Truth Case Study*. Cham: Springer International Publishing; 2015;11–20. https://doi.org/10.1007/978-3-319-19830-9_2
23. Marulli F, Pareschi R, Baldacci D. The internet of speaking things and its applications to cultural heritage. *Proceedings of the International Conference on Internet of Things and Big Data*, Rome, Italy; 2016:107–117. https://doi.org/10.5220/0005877701070117
24. Sun Y, Song H, Jara AJ, Bie R. Internet of things and big data analytics for smart and connected communities. *IEEEAccess*. 2016;4:766–773.
25. Jara AJ, Sun Y, Song H, Bie R, Genoud D, Bocchi Y. Internet of things for cultural heritage of smart cities and smart regions. *Advanced Information Networking and Applications Workshops (WAINA), 2015 IEEE 29 th International Conferenceon*, Gwangju, Korea; 2015:668–675.
26. Chianese A, Marulli F, Piccialli F, Benedusi P, Jung JE. An associative engines based approach supporting collaborative analytics in the internet of cultural things. *Future Gener Comput Syst*. 2017;66:187–198.
27. Bobbio A, Cerotti D, Gribaudo M, Iacono M, Manini D. *Markovian Agent Models: A Dynamic Population of Interdependent Markovian Agents*. Cham: Springer International Publishing; 2016;185–203.
28. Gribaudo M, Iacono M. A different perspective of agent-based techniques: Markovian agents. In: Kolodziej J, Correia L, Manuel Molina J, eds. *Intelligent Agents in Data-Intensive Computing*, Studies in Big Data,vol. 14: Springer International Publishing: Cham (ZG), Switzerland; 2016:51–70.
29. Barbierato E, Gribaudo M, Iacono M. Modeling and evaluating the effects of Big Data storage resource all ocation in global scale cloud architectures. *Int J Data Warehouse Min*. 2016;12(2):1–20.
30. Castiglione A, Gribaudo M, Iacono M, Palmieri F. Modeling performances of concurrent big data applications. *Softw Pract Experience*. 2015;45(8):1127–1144.
31. Cerotti D, Gribaudo M, Bobbio A. Disaster propagation in heterogeneous media via Markovian agents. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. 2009;5508LNCS:328–335.citedBy0.
32. Cordero F, Fornari C, Gribaudo M, Manini D. Markovian agents population models to study cancer evolution. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. 2014;8499LNCS:16–32. citedBy0.
33. Helbing D, Farkas I, Molnàr P, Vicsek T. Simulation of pedestrian crowds in normal and evacuation situations. In:Schreckenberg M.,Sharma SD,eds. *Pedestrian and Evacuation Dynamics*. Berlin: Springer; 2002:21–58.
34. Almeida JE, Rossetti RJF, Coelho AL. Crowd simulation modeling applied to emergency and evacuation simulations using multi-agent systems. *CoRR*. 2013;abs/1303.4692.http://dblp.uni-trier.de/db/journals/corr/corr1303.html#abs-1303-4692
35. Jensen K, Kristensen LM, Wells L. Coloured Petrinets and CPN tools for modelling and validation of concurrent systems. *Int J Softw Tools Technol Transfer*. 2007;9(3):213–254.

How to cite this article: Gribaudo M, Iacono M, Levis AH. An IoT-based monitoring approach for cultural heritage sites: The Matera case. *Concurrency Computat: Pract Exper*. 2017;29:e4153. https://doi.org/10.1002/cpe.4153