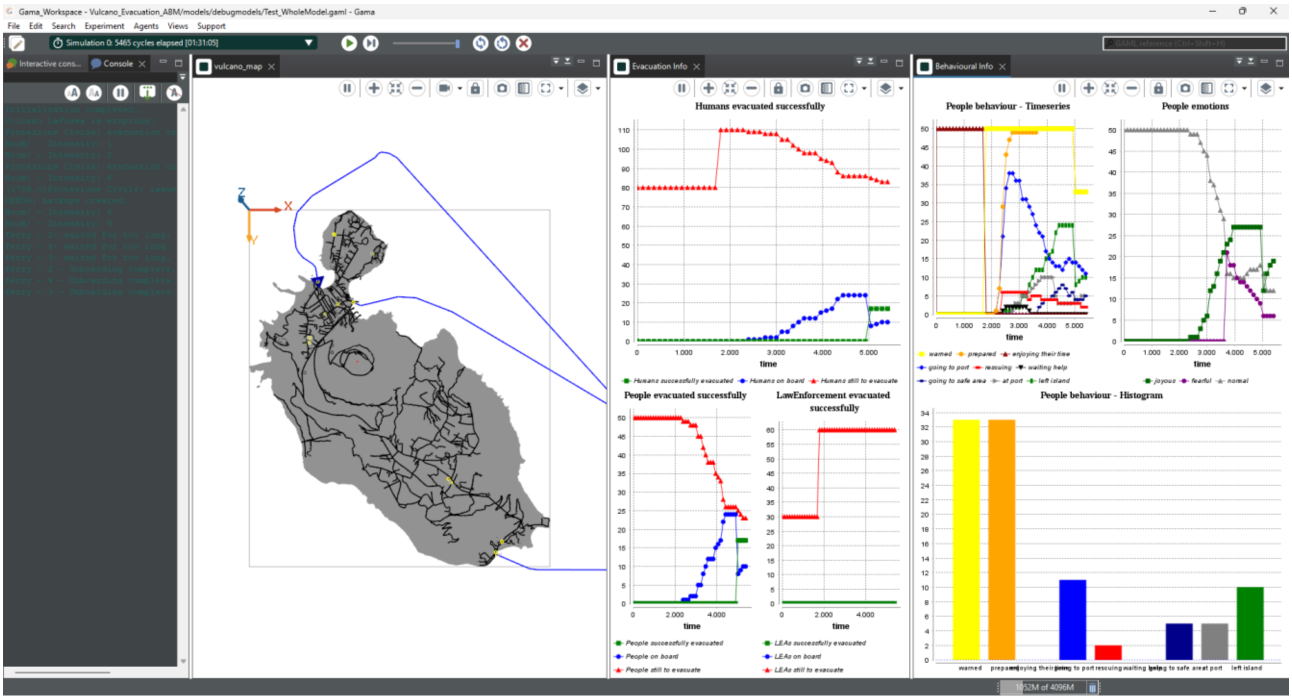
Enrico Sansone, Irene Silvestro, Ilaria Vetrano

Progetto per l'esame di Econofisica e Laboratorio MAS
Settembre 2025 – A.A. 2024/25

Simulating a Volcanic Island Evacuation with Emotions and Social Structure
An Agent-Based Model Applying the BEN Architecture



SIMULATING A VOLCANIC ISLAND EVACUATION WITH EMOTIONS AND SOCIAL TIES 1

PROJECT OVERVIEW 1
LITERATURE REVIEW 1
MODEL OVERVIEW 1
ENVIRONMENT 2
AGENTS ARCHITECTURE 2
CIVIL DEFENSE 2
VOLCANO 3
EVACUATION INFRASTRUCTURES (PORTS, HELIPORTS, WAITING AREAS) 4
EVACUATION VEHICLES (FERRIES, HELICOPTERS) 4
HUMAN (PEOPLE, LEA) 5
People 5
Law Enforcement Agents (LEAs) 9
INPUT DATA 9
EXPERIMENTAL SETTING 9
INITIALIZATION 10
RESULTS 11
FUTURE DEVELOPMENTS 16
CONCLUSIONS 16

REFERENCES 17

Project overview

This project consists in an Agent-Based Model (ABM) built to simulate the evacuation of the population during a volcanic eruption of the island of Vulcano (Italy). Our approach integrates emotions and social structures in the decision-making process of the agents. To further provide a realistic environment, we also used the real-world road network of Vulcano island. The model was developed in GAMA (version 1.9.3).
During a simulation run, the volcano agent acts as a trigger with its “boom”-events that are perceived by people on the island and changes the decisions of agents. The Civil Defence (CD) agent manages and coordinates Law Enforcement Agents (LEAs), the issuing of evacuation order to civilian population and the evacuation resources such as ferries, helicopters and ports as well as monitoring volcanic activity and evacuation status. Only when the civilian population is evacuated, LEAs can leave the island and as soon as they reach Milazzo’s port the simulation is terminated.
Our research aim is to assess the impact that the emotional and social engine have on the evacuation time and to evaluate the effectiveness of a broadcast alert system (IT-Alert) in this crisis scenario by comparing it to a point-to-point one.

Literature Review

This work has been inspired by the BEN (Behavior with Emotions and Norms) architecture described in [1] [1] a formal framework that integrates personality, emotions and social links into agent behavior. While this paradigm has been previously applied to indoor evacuations (e.g. Kiss Nightclub [1]), our project expands the use of BEN architecture to an open-air, real-world emergency scenario.
The small size of the territory and the population make the island of Vulcano an ideal test case for our study and to further ground our simulation in a realistic context, we analyzed the evacuation plan provided by the Italian Civil Protection [2], although slight changes have been adopted to fit the simulation framework.
Another study that was pivotal in our work is [3] by Bonadonna et al., who used Vulcano as a case study for their AnyLogic ABM tool and to assess the economic impact of volcanic hazard and related evacuation. Their model simulates the evacuation of civilians and includes a volcano agent, but it does not incorporate the emotional and social factors that are central to the research presented here nor it supports the onset of any actual volcanic activity.

Model Overview

In our model, the onset of the evacuation process is dictated by the Volcano agent’s behavior for which we introduced a more complex architecture that goes beyond the simple change of internal states (found in [3]) and that gives space to the development of a richer and more complex reproduction of volcanic activity. It is the Civil Defense who is in charge of monitoring the volcanic status and of the initiation of the evacuation process, which was implemented in two different ways: with or without the IT-Alert system. In the former, the Civil Defense immediately orders LEAs to go patrol waiting areas and other key evacuation infrastructure (ports and heliports) and after a fixed time (selected by the modeler) a broadcast message is sent to the whole civilian population. This was implemented by having the Civil Defense insert an *Evacuation Order* belief into people’s knowledge base. In the latter, Civil Defense immediately orders a number of LEAs to alert civilians on a door-to-door basis (point-to-point communication), i.e. by moving around the island and alerting people that are located within a certain distance from the LEA. Finally, evacuation vehicles are alerted after a certain amount of time, chosen by the modeler.
Once people are aware of the evacuation, they have to make a decision among these four plans: *Go to Evacuation Infrastructures*, *Go to Port*, *Go Rescue Someone* or *Wait for Someone*. Here lies our major innovation: the decision-making algorithm (**Figure 2**) involves both the personality traits of the agent and their emotions. Usually, evacuation simulation models assume agents are rational and strictly follow the evacuation plan. In our model we introduce irrationality at two levels. Firstly, we allow agents to make decision depending on their personality traits and emotions which could lead to different actions than the rational one (i.e. go to the evacuation port). Next, we included a parameter to represent the probability of choosing a plan at random, thus allowing for total irrationality. Also note that the decision process can also

be triggered by agents who are not aware of the ongoing evacuation but feel scared in response to volcanic activity.
Once people have reached a port where ferries are available for boarding, they are asked to decide whether they want to board or not, communicating their decision to the Port agent who acts as a mediator. In this decision-making algorithm (**Figure 3**) we introduce, in addition to the emotional dimension, the importance of social links that may impact the decisions taken by the agents. For all cited emotions, an underlying layer of emotional contagion between agents is put in place, as described later.
After the evacuation of the whole population has been completed (i.e. there is no person who hasn't set foot on a ferry yet), LEAs are also allowed to leave the island using both ferries and helicopters.

Environment

The simulation is set within a modeled environment representing the island of Vulcano. The physical and logistical components are based on real-world data to capture the island's geography and infrastructures. Geographical data was downloaded from Open Street Map (OSM) and manipulated using QGIS, an open source GIS software, in order to be used in the model. It includes:

- The La Fossa volcano acting as the primary (and only) source of hazard
- The island's road and walking trail network, which have been merged together
- The maritime routes connecting the island's ports and Milazzo's port
- Buildings and designated waiting areas
- 4 ports, located in Vulcano Porto (Ponente and Levante), Gelso and Milazzo
- 3 heliports[1], located in Piano, Vulcanello and an off-shore heliport to simulate a helicopter carrier stationing northeast of the island

Agents Architecture

In this section we get into the details of each agent species' architecture.

Civil defense

The Civil Defense acts as the *brain* of the evacuation: it is responsible for managing LEAs, coordinating evacuation vehicles and infrastructures, and for monitoring volcanic activity and the overall state of the evacuation by tracking the number of people on the island. For data collection purposes, the Civil Defense agent is allowed to track people's current activity as well as their emotional status.
An essential monitoring task is the one regarding volcanic activity (see the *Volcano* agent section for more details). When the activity reaches Level 2 (i.e. eruption), Civil Defense is set to issue the evacuation order, which is split in two: an order given to LEAs and an order given to alert evacuation vehicles. The issuing of both orders allows for an individual delay, whose duration is at modeler's choice, in order to account for the time needed to actually deliver such orders. In case of an IT-Alert evacuation scenario, civilians will be alerted along with evacuation vehicles, but an additional delay can be set for the issuing of the broadcast message. Note that in the current implementation, only the monitoring of a single volcano agent is supported, though it can be rapidly enhanced to support the monitoring of a greater number of agents.
The Civil Defense's management of LEAs involves assigning at least one LEA to each waiting area (or two, if there are enough LEAs available). In the door-to-door scenario, the remaining LEAs are assigned to perform door-to-door warnings up until a maximum number of LEAs is reached. Any further remaining agent is then assigned to patrol activities. We've allowed Civil Defense to ask for additional support, which will arrive on the island at a time and in a quantity chosen by the modeler.
Furthermore, the Civil Defense agent coordinates all available ferries and helicopters. It monitors the status of ports and heliports to check whether there are people waiting to be evacuated. Once a vehicle is ready, the Civil Defense assigns it a destination, prioritizing the facility with higher number of waiting individuals. If more than a vehicle is available, the one with highest capacity is sent. Additionally, the model allows for an additional delay to be set for the first vehicle dispatch to each facility.

[1] Actually, there are 5 heliports initialized from data though two of them (the one at Milazzo's Hospital and the one on the summit of La Fossa) are not used by the model.

Finally, once the number of civilians on the island reaches zero, the Civil Defense issues the evacuation order for LEAs, allowing them to depart using ferries and helicopters.

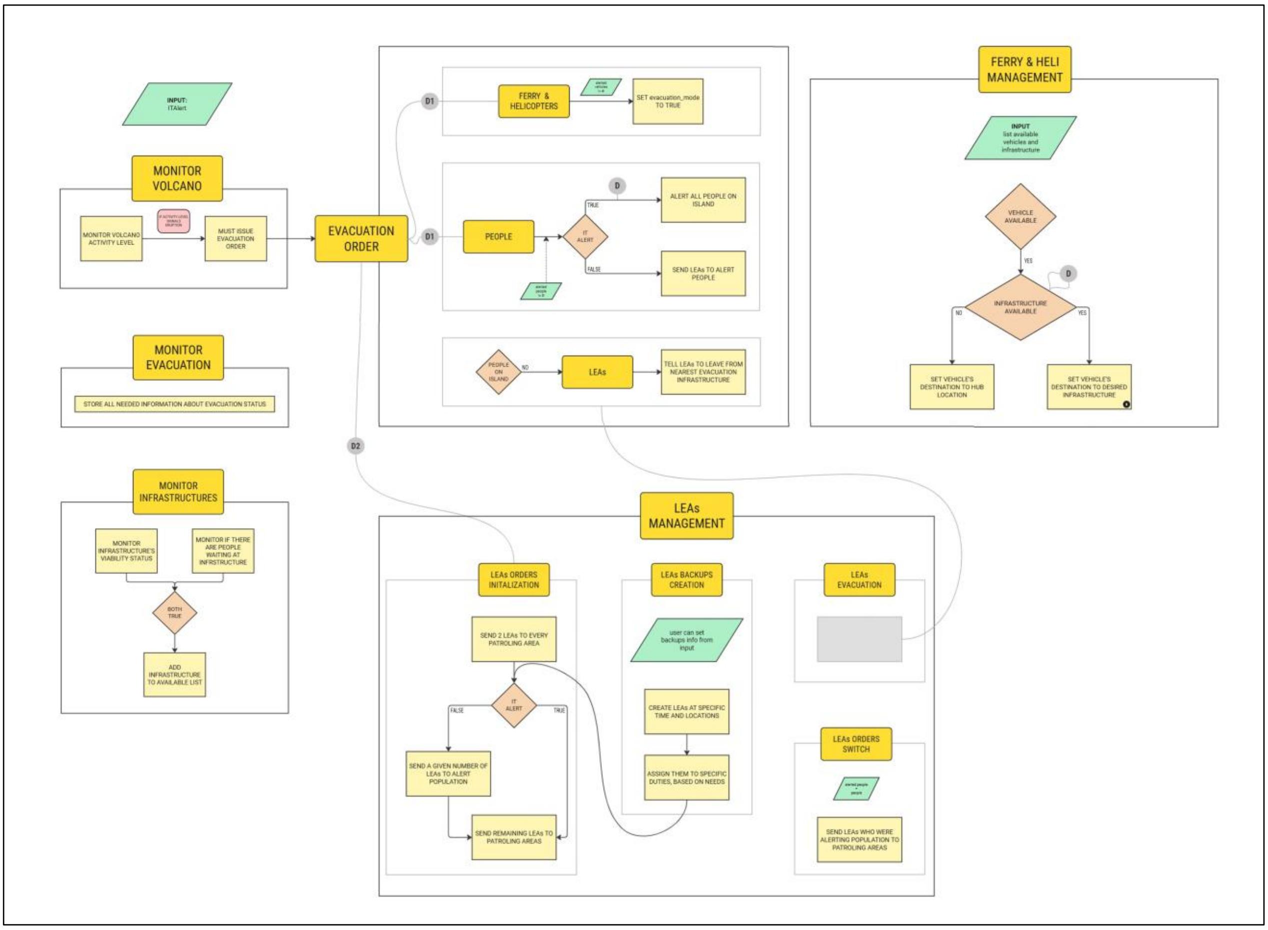


Figure 1. Civil Defense's agent architecture.

VOLCANO

The Volcano agent is meant to be able to recreate the course of a volcanic eruption. In order to do so, it creates a number of *EruptivePhenomenonManager* subagents that are responsible for the modelling of the actual volcanic activity which is carried on with the use of agents of the *EruptivePhenomenon* species. These two agent species serve as a container for the definition of specialized agents, as the former is dedicated to keep track of the volcanic activity and manage eventual delays put in place by the modeler, while the latter is committed to manage the duration of the eruptive phenomenon; therefore, to mimic different eruption events, a more specially designed architecture that share this common structure is needed. The choice for the volcanic activity processes to be enabled in a simulation is given to the modeler, who can then specify custom parameter values for each of them (see **Table *1*** for an example), otherwise they are set to default values. Under the Volcano agent's authority, there is also the updating of its *activity status* which can takes values of 0, 1 or 2, resembling the states of *quiescence, unrest* and *eruption*, respectively. To sum it up, the volcano agent acts as a manager of subagents meant to replicate the actual volcanic activity. Its design also allows to host a number of subagents aimed at reproducing the internal structure of the volcano, but we did not pursue furtherly this feature as it went far beyond the scope of this project.

To better illustrate the architecture of the volcano agent, we proceeded to get into the details of our current model's implementation, which revolves around a single example of volcanic activity: the emission of roaring sounds.

When the activity level reaches a value of 2 (i.e. eruption), it creates (after a given delay) an agent of *EruptivePhenomenonManager* species, which, with a great imagination effort, we called *RoaringSoundEmissionManager*. This agent generate roars of intensity I ranging between 0-9 (following a custom distribution) and with the time interval between two different sound emissions being exponentially

distributed.[2] These roars are designed as specialized agents of the *EruptivePhenomenon* parent species, denoted *RoaringSoundEmission*, whose set of parameters can be found in **Table 1**.
This agent is initialized at Volcano agent's location with a null size and, for every step of its lifetime, it attempts to impact people agents placed within its radius and then updates its size. When a person is exposed to the soundwave, it can be impacted by it with a probability that follows a given model (at modeler's choice). In our implementation, the success of this influential process happens with a probability p given by:

$$p = \frac{I^2}{I_{max}^2}$$

Note that, in order to avoid the repetition of the process for the individual sound emission, the agent keeps track of the people agents that were met in previous timesteps. In case of a successful event (i.e. the person has heard the boom), a belief retaining information on its intensity value is added to the person's belief base with a lifetime dependent on such value (the higher the intensity, the longer the lifetime). Hence, roars are designed to create fear in people, thereby affecting their decision-making process regarding the evacuation.

parameter	*default value*
exponential distribution scale factor	180 s
activation delay	0 s
speed of sound	343.3 m/s
duration	20 s
intensity discrete distribution	$\mathcal{U}(0,9)$

Table 1. Default values for the RoaringSoundEmission agent's parameters.

Evacuation infrastructures (Ports, Heliports, Waiting Areas)

In the model, ports and heliports serve as main hubs for evacuation. They are responsible for managing the evacuation flow by dynamically communicating with individuals to determine who is willing to board a vehicle, then updating their status and location accordingly. Upon arrival of vehicles with people on board, these infrastructure agents handle the unboarding process, where humans are removed from the simulation. Each infrastructure agent monitors both its vehicle and people capacity rates, if it has no available space for vehicles it can refuse the approaching of more incoming vehicles.. The model is designed to also allow for the infrastructure to be unviable following certain conditions (e.g. adverse weather, volcanic activity), with Civil Defense being appointed to monitor the viability status.

Evacuation Vehicles (Ferries, Helicopters)

Two types of evacuation vehicles have been designed in the model: ferries and helicopters. In our implementation they serve two different purposes, as one hand ferries are meant to evacuate both the civilian population and LEAs while on the other hand helicopters take part solely into the final evacuation of LEAs. However, the agents' design can be easily expanded to serve other goals, such as rescuing people in need (e.g. aerial rescue).
Ferries and helicopters are both associated with a hub and a target infrastructure, with the latter representing the infrastructure that has to be currently reached by the vehicle. They also share a common structure as they can be set in normal (e.g. ferries are involved in their ordinary transportation duties) or in evacuation mode, when their goal is to take part in the evacuation process. Once a vehicles arrives at its designated destination, it begins the boarding process up until its maximum capacity is reached (or if the vehicle waited for a too long time interval, whose maximum value is set at modeler's choice), when the vehicle departs to reach its hub. After its arrival at the hub's location, the unboarding process is started and, once the vehicle is emptied, it signals to the Civil Defense that it is available to leave again. A facet that prioritizes the safety of the vehicle over the evacuation duties has been added, though it does not play any role in the current model's implementation as there are no hazard that may impact evacuation vehicles.

[2] This led to the discovery of a bug in GAMA's implementation of the random choice operator connected to the exponential distribution. What is meant to represent the *rate* parameter of the distribution (conventionally denoted as λ), is instead used as the *scale* parameter (β) that also represents the mean. In the model's code we opted to keep the parameter's name as λ, though it serves as the scale parameter.

The main architectural difference between ferries and heliport resides in their movement logic: ferries are restricted to move on the fixed ferry network only, while helicopters are able to move freely in the environment following a simple take-off/landing dynamics.

HUMAN (PEOPLE, LEA)

Human species serves as the parent species for both People and LEAs. This means they share fundamental capabilities, such as movement and the boarding process during the evacuation. A crucial distinction, however, is that only People have emotional and social components that influence their behavior. An example being that even though they share the same boarding process, people can refuse to board a ferry to wait for friends, while LEAs must board when given the opportunity.

People

People are the agents that implement the core innovation of the BEN architecture, as they are modeled on the personality, emotional, and social structures described in [1].

BEN ARCHITECTURE: KEY ELEMENTS

Social relations

Based on [4], BEN describes a social relation using a finite set of 5 variables. Therefore a social relation between a pair of agents i, j is defined as $R_{i,j}(L, D, S, F, T)$ where R is the identifier of the social relation, $L \in [-1,1]$ represents the degree of *Liking*, $D \in [-1,1]$ represents the degree of *Dominance* over the other agent, $S \in [0,1]$ the degree of *Solidarity*, $F \in [0,1]$ the degree of *Familiarity* and $T \in [-1,1]$ the degree of trust.

Personality

BEN makes use of the OCEAN model [5] of personality, where five traits of personality are defined with values ranging in [0,1]:

O	*Openness*	$O = 0$	⟶	narrow-minded
		$O = 1$	⟶	open-minded
C	*Conscientiousness*	$C = 0$	⟶	impulsive
		$C = 1$	⟶	planner
E	*Extroversion*	$E = 0$	⟶	shy
		$E = 1$	⟶	extrovert
A	*Agreeableness*	$A = 0$	⟶	hostile
		$A = 1$	⟶	friendly
N	*Neurotism*	$N = 0$	⟶	neurotic
		$N = 1$	⟶	calm

Emotion

In BEN, each emotion is defined as $Em_i(P, Ag, I, De)$ where Em_i stands for the name of the emotion felt by agent i, P is the predicate representing the fact about which the emotion is expressed, Ag is the agent causing the emotion, I is the intensity of the emotion and De is the decay withdrawn from the emotion's intensity at each timestep.

Among the emotions defined in BEN, we selected fear and joy to be used in our current model, defined as:

$$Fear_i(P, j) \equiv Uncertainty_i(P_j) \,\&\, Desire_i(\neg P)$$
$$Joy_i(P_j, j) \equiv Belief_i(P_j) \,\&\, Desire_i(P)$$

Their intensity is modulated with the use of personality traits (V indicates the value or strength):

$$Fear \rightarrow I[Em_i(P)] = V[Uncertainty_i(P)] \cdot V[Desire_i(P)] \cdot \big(1 + (0{,}5 - N)\big)$$

$$Joy \rightarrow I[Em_i(P)] = V[Belief_i(P)] \cdot V[Desire_i(P)] \cdot \big(1 + (0{,}5 - N)\big)$$

as well as their decay (equal for both):

$$De[Em_i(P)] = N \cdot I[Em_i(P)] \cdot \Delta t$$

where Δt is the time-step duration.

<u>*Emotional Contagion*</u>
BEN allows the inclusion of the social dimension in the emotional one by introducing emotional contagion, thus emotions can arise either from the creation of the relevant predicate or through emotional contagion. The latter is based on the ASCRIBE model [6], and a contagion from agent i to agent j is represented by the tuple:

$$(Em_i, Em_j, Ch_i, R_j, Th)$$

where Em_i is the emotion that triggers the contagion if perceived by j, Em_j is the emotion created by j, Ch_i is the charisma value ($Ch_i \equiv E_i$) of i that indicates its power to express its emotions, R_j is the receptivity value ($R_j \equiv 1 - N_j$) of j that expresses its capacity to be influenced by others and Th is a threshold value (i.e. the contagion is executed if and only if $Ch_i \cdot R_j \geq Th$). The initial intensity of the newly created emotion is given by:

$$I\big[Em_j(P)\big] = \begin{cases} I\big[Em_j(P)\big] + I[Em_i(P)] \cdot Ch_i \cdot R_j, & \text{if } Em_j(P) \text{ exists} \\ I[Em_i(P)] \cdot Ch_i \cdot R_j, & \text{otherwise} \end{cases}$$

In our model we did choose to allow for emotional contagion. For this reason, an agent can acquire an emotion simply by being exposed to another's emotional state, even without experiencing the emotion-triggering event themselves.

In our model, social network is built as follows: the whole civilians population is split into cliques, whose size is uniformly distributed between 5 and 15 people[3]. Social links are then built with asymmetrical features, by picking random values uniformly in the range [0,1]. On the other hand, personality traits have been initialized using a Gaussian distribution with mean 0.5 and variance 0.12.
We also allowed for two emotions: *FearEruption*, which expresses the agent's fear about the possible onset of an Eruption, and *JoyPort,* being relative to the agent's joy of having reached the port. *FearEruption* is based on the predicate *Eruption*, which represents the ongoing of a volcanic eruption. Thus, People agents are initialized with a desire for this eruption not to happen and therefore such emotion is felt whenever the agent is uncertain about the onset of an eruption. In our model, the strength of this uncertainty depends linearly on the intensity of the roaring sounds that have been heard by the agent (i.e. that are currently placed in its belief base). This strength is then modulated by a parameter which is dependent of the agent's personality (see **Table 2**).
People's default behavior is that of wandering around the island up until their decisional process is triggered, which could happen either because they received the evacuation order or because they are considered fearful (i.e. have an intensity value for *FearEruption* greater than a threshold we set at 0.6). In this decision-making process they can choose between four different plans, *Go to Port, Go to Evacuation Infrastructure, Wait for Someone* and *Go Rescue Someone*, after which they set to prepare themselves for pursing it. Note that the preparation phase is only linked to the first time this process is executed, as well as the possibility to choose the *Wait for Someone* plan. We enabled a form of direct communication between each agent and its friend, meant to signal what each agent is doing. Each message contains the sender's location and current activity status and it is represented as an update of a related belief in the receiver's belief base. The list of available statuses is *unknown* (default value), *going to port, rescuing, waiting, at port* and *on board.*
Let's now proceed with a detailed description of each plan:

- ***Go to Port***
  This plan is aimed to make the agent reach the closest port, which we identify as the rational behavior. When chosen, the agent will communicate to its friend that it is going to the nearest port and once the port is reached, a new update will follow.
- ***Go to Evacuation Infrastructure***
  This other plan is very similar to *Go to Port*, but it is extended as its scope is now to reach the closest between a port, a heliport or a waiting area. However, this plan can be chosen if and only if the agent is experiencing fear above a certain threshold (**Figure 2**). If the agent reaches one of the latter two,

[3] The latest generated clique may be as small as a single person.

it is supposed to encounter LEAs who will calm it down and redirect it to the closest port. Note that this is the only plan that does not include the transmission of the status update message.

- ***Wait for Someone***
  When this plan is activated, the agent is hoping for a friend to come to its rescue. It will therefore tell its friends its location and wait for them there. If a friend finally reaches out, both agents will go to the nearest port, otherwise the waiting agent will wait up until a maximum time, whose value is personality-dependent (see **Table 2**).
- ***Go Rescue Someone***
  In this altruistic plan, the goal is to rescue a friend needing help, by going to its location and bring it to a port. The agent firstly communicates its choice to go look for someone else and it then selects the closest of its friends (i.e. whose social link hash the highest values for the sum of *liking* and *familiarity*) between those who it thinks are in need, as it believes their status is "unknown" or "waiting". Next it goes to the friend's location and in case it's there, the rescuer agent will calm its friend (if needed) and will tell it to go to the nearest port. The effectiveness of the calming action is dependent on the *dominance* value of the social link between the two from the rescued agent's perspective. If the friend is not found, the agent will add it a list of friends that don't need help and go to the nearest port. In case the agent has no friends to rescue, it will go to the port.
- ***Prepare***
  This plan is executed only the first time the decision-making process is activated. The agent will just stop moving and wait up until a maximum time, whose value is extracted using a truncated Gaussian with average and standard deviation defined by the modeler, in a similar fashion of [3].

An overview of the decisional process is shown in **Figure 2**, where the influence of personality traits and emotions is illustrated. Note that with a given probability (i.e. 5%) the agent will choose at random between all the available plans. The result of this process consists of an agent's commitment to a plan that will last for a given lifetime, dependent on the agent's personality[4], after which the agent is able to reconsider its previously taken decision.

```
begin
  if random[0,1] <= 0.95 then
    if extroversion >= 0.8 and first_plan_selection
      do WaitForSomeone
    else
      if self.has(fearEruption) and fearEruption >= 0.6
        do GoToEvacuationInfrastructure
      else
        if agreeableness >= 0.6
          do GoRescueSomeone
        else
          do GoToPort
  else
    chooseOnePlanAtRandom
end
execute chosenPlan until execution_time >= decision_lifetime
```

Figure 2. Decision-making algorithm.

Throughout the simulation, emotional contagion is active, which concerns both *FearEruption* and *JoyPort*. In fact, an agent can develop the emotion *FearEruption* even if it didn't notice the boom, but it has encountered someone who is fearful. Thus, the decision process might be triggered for agents who did not actually hear the boom nor did receive the evacuation order, as they experience the *FearEruption* emotion due to emotional contagion. *JoyPort* arises once the agent has arrived at a port where emotional contagion can be triggered: the more people perceive joy, the more joyful the agent feels. This is relevant as when the agent is in the port and a ferry has arrived, it is asked by the Port agent to make a decision whether he wants to board the ferry or not and this decision-making process is again influenced by the agent's emotional status. The algorithm behind this decisional process is depicted in **Figure 3** and it is based on the idea that if the agent is very happy of being in the port, it will act selfishly and decide to board without considering its close

[4] The decision persistence duration is lower-bounded to 5 minutes.

friends' statuses (a close friend is defined as a friend for which the agent has a *liking* grater than 0.5). The weaker its joy, the greater is the number of friends the agent will be waiting to be safe (i.e. with a *on board* or *at port* status) before it agrees on boarding. If the agent refuses to board, it will be asked again in the next timesteps, otherwise it is added to the list of people ready to board the ferry and it will eventually embark it. Once on board, the agent communicates it to its friend and it is taken to Milazzo, where it will be unboarded, then considered successfully evacuated and finally removed from the simulation.

```
compute percentage_of_safe_close_friends
begin
    if self.has(JoyPort) and JoyPort >= 0.7 then
        boarding_decision = true
    else if self.has(JoyPort) and JoyPort ∈ [0.3,0.7) then
        if percentage_of_safe_close_friends >= 0.5 then
            boarding_decision = true
        else
            boarding_decision = false
    else
        if percentage_of_safe_close_friends >= 0.7 then
            boarding_decision = true
        else
            boarding_decision = false
end
```

Figure 3. Boarding decision algorithm.

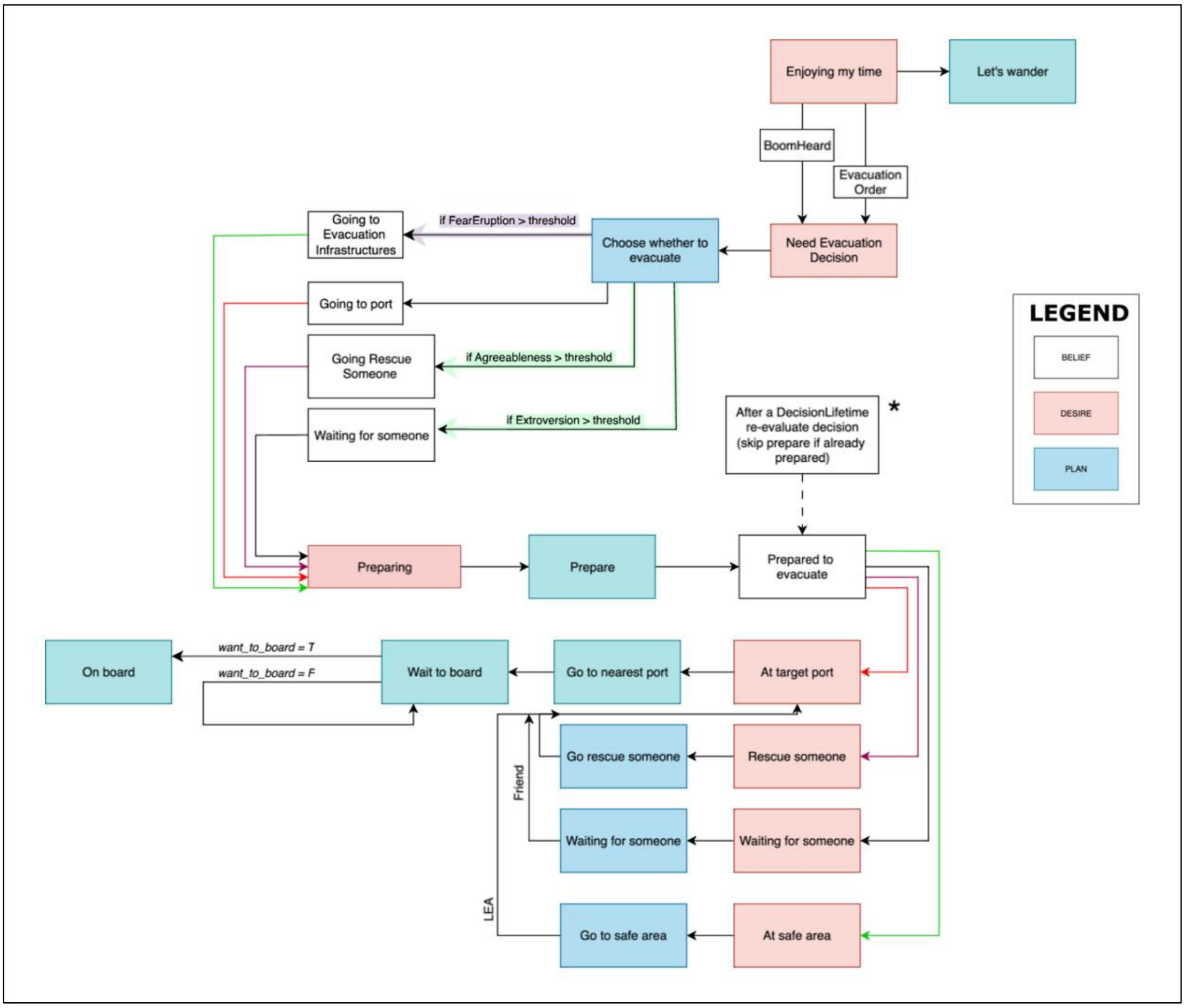


Figure 4. People agents' architecture.

Law Enforcement Agents (LEAs)
The behavior of Law Enforcement Agents (LEAs) is entirely dictated by directives from the Civil Defense. At the beginning of the simulation, the Civil Defense assigns a specific role to each LEA, who can be tasked with two primary duties: patrolling designated areas and, in the door-to-door scenario, warning civilians of evacuation throughout the island.
When patrolling, if a person whose intention is not to get to a port is encountered, the LEA will instruct the individual to reach the port; if that individual is fearful, the LEA will call them down first (the effectiveness of this action is dependent on the *Neurotism* of the person being calmed).
When an evacuation order is issued on a point-by-point basis, a number of the available LEAs is assigned to the door-to-door warning operation of the civilian population. These agents will move all over the road network (by selecting one of its vertices not closer than a given distance) and alert every unalerted person they encounter within a certain radius. Once every person on the island is aware of the evacuation order, all LEAs involved in the warning operation will turn to patrolling duties.
Finally, when told to evacuate by Civil Defense, LEAs will reach the closest evacuation facility (port or heliport) in order to leave the island.
To account for the ability of LEAs to use vehicles to move around, their travel speed is set at a higher value than walking speed, except for when they've reached their final evacuation infrastructure.

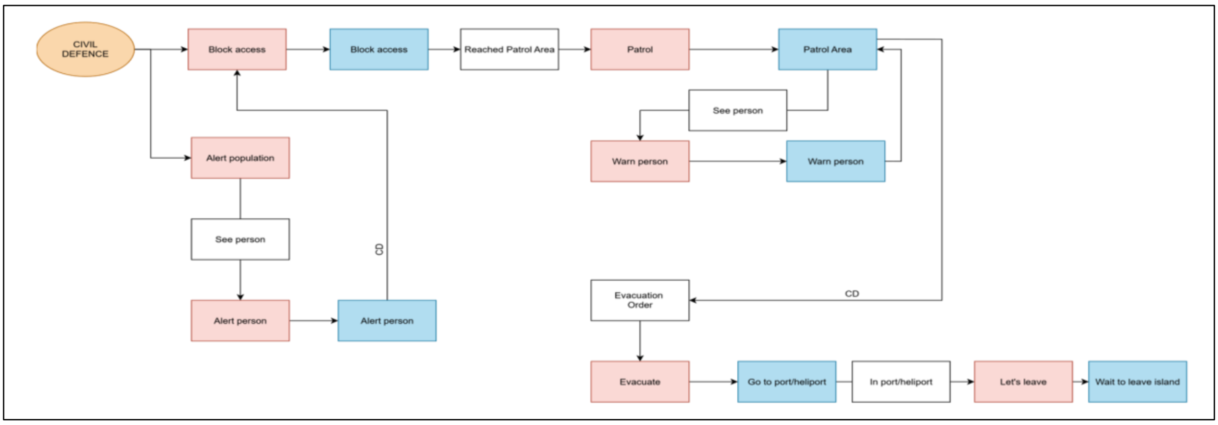


Figure 5. Law Enforcement Agent's (LEA) architecture.

Input data

As mentioned above, geographical data was downloaded from OSM. The vehicle capacity of ports was set as follows: a maximum of one ferry was allowed to dock at the smaller ports of Gelso and Ponente, two at the larger port of Levante, while the port of Milazzo could host a practically unlimited amount of ferries (1000). On a similar fashion, the heliports on the island of Vulcano (in Piano and Vulcanello) could host a helicopter only while the off-shore heliport up to four.
After beginning the collection of data for ferries from the local maritime transportation companies' websites, we opted to follow [3] and have ferries with a capacity of 200, 400, 600 and 800 people and with a cruising speed of 50 km/h (27 kn). We based the helicopter agents on the Augusta Westland AW139 SAR in use by the Italian Coast Guard, setting a capacity of 8 people and a cruising speed of 200 km/h (108 kn).
For some of the model components we looked a work already present in literature, such as [1] for people's personality settings and [3] for people's walking speed. Unfortunately we could not find any reliable data sources for some key features of the model, such as the number of LEAs that are involved in on-shore duties or estimates for the delays related to the issue of orders by the Civil Defense. For these and other aspects of the model, we proceeded on a common-sense basis. A comprehensive list of parameters can be found in **Table** *2*.

Experimental Setting

In this project we aim to determine how an architecture that incorporates the emotional and social behaviors of evacuees affects the overall evacuation time. We also want to investigate the impact that different alert systems, implemented by evacuation authorities, can have on evacuation behavior and time. Thus, the outline of our experiment is as follows.

We ran three sets of ten runs in these scenarios for which we compared the results in the following section:

i. ***IT-Alert with social links, emotions and personality traits enabled***
In this scenario, the model is deployed to its full extent while the evacuation order is issued on a broadcast messaging basis.

ii. ***IT-Alert with social links, emotions and personality traits disabled***
In this other experiment scenario, the use of social, emotional and personality architectures was inhibited thanks to appropriate switches available in GAMA. This way, once people agents were given the evacuation order, they would strictly follow it by preparing and going to the nearest port. Note that due to technical constraints, all social links and personality-related variables had to be initialized to a zero value, apart from *Conscientiousness* (which was set to a value of 0.5). Still, due to the architectures interdiction, they did play no role after the initialization of the agents.

iii. ***Door-to-door with social links, emotions and personality traits enabled***
Same as *i*, however, it is a door-to-door evacuation that is carried on in this scenario.

Each run in a set was initialized with a different seed value, while each set of runs was characterized by the same set of seed values. Each timestep in each run is equivalent to 1 second of real time.

Initialization

We aimed at modelling the evacuation of 1000 civilians (following the *low-season* evacuation scenario found in [3]), but due to hardware limitations we restricted this number to 500 which enabled us to run two parallel simulations. We set the number of LEAs present at the beginning of the simulation to 30, with 30 more to set foot on the island 30 minutes after the first alert of LEAs. In the door-to-door evacuation scenario, a maximum of 30 LEAs were deployed in the warning process. While civilians are initialized at some point of the road network or in a random building with equal probability, LEAs are initialized with a 64% chance of being located in Levante's port, 12% in Gelso's deck 4% in Ponente's deck and with the remaining 20% chance of being in one of the road network nodes.

Evacuation vehicles have been initialized in their standard mode, with ferries being located in any of the ferry network nodes with a 50% probability of being directed to Levante and a 50% probability of being directed to Milazzo (their hub). Helicopters, instead, are located on the off-shore heliport where they reside stationary (with it being their hub). Both vehicles had a boarding and unboarding speed of 4 people/min, while ferries had a maximum waiting time of 3000 s against the helicopters' 1200 s.

Finally, the volcano agent was initialized at activity level 1 (i.e. unrest) though due to update it to level 2 (i.e. eruption) during the very first step of the simulation. Civil Defense then immediately alert LEAs to be on evacuation duties as this alerting process delay is set to zero. After 20 minutes Civil Defense alerts ferries and helicopters and, in the IT-Alert evacuation scenarios, another 10 minutes will go before the civil population is alerted.

An extensive list of all the parameters settings can be found in **Table *2***.

parameter			***value***
civil defense	*time needed to issue the general evacuation order (to ferries and helicopters)*	`time_needed_to_issue_evac_order`	20 min
	time needed by CD to alert LEAs	`time_needed_to_issue_LEA_order`	0 s
	time needed by CD to send a IT-alert message after the general evacuation order is issued	`time_needed_to_issue_italert`	10 min
	time needed before sending a ferry to a given port for the first time	`send_ferry_decision_time`	10 min
	time needed before sending a helicopter to a given port for the first time	`send_helicopter_decision_time`	10 min
	backups arrival time	`backups_arrival_time`	30 min
	backups arrival location	`backups_arrival_location`	any port
	backups number	`backups_number`	30
	maximum number of LEAs deployed to warn the civilian population	`ordering_evac_LEAs_nb`	30
volcano	*scale factor (average) of the exponential distribution in RoaringSoundEmissionManager*	`lambda`	1000 s
ferry	*stop distance at which the vehicle request permission to utilize the intended evacuation infrastructure*	`approach_distance`	1 km
	number of people boarded per second	`boarding_speed`	$1/15\ s^{-1}$
	number of people disembarked per second	`unboarding_speed`	$1/15\ s^{-1}$
	maximum time spent waiting for people to arrive and board the vehicle	`max_waiting_time`	50 min

	maximum number of people that the vehicle can transport	`capacity`	$[2,4,6,8]\times 10^2$
	cruising speed of the vehicle	`cruising_speed`	50 km/h
helicopter	〃	`approach_distance`	100 m
		`boarding_speed`	1/15 s^{-1}
		`unboarding_speed`	1/15 s^{-1}
		`max_waiting_time`	20 min
		`capacity`	8
		`cruising_speed`	200 km/h
people	*maximum distance of agent's perception*	`view_dist`	30 m
	on foot speed	`walking_speed`	$x \sim \mathcal{U}(3,6)$ km/h
	probability for which the decision process is totally random	`total_irrationality`	5%
	time spent preparing	`total_preparing_time`	$x \sim \mathcal{N}^{trunc}(600,300)$ s
	personality traits values	`O,C,E,A,N`	$x \sim \mathcal{N}(0.5,0.12)$
	decision's persistence time	`decision_lifetime`	$\max(300, 1800 \cdot C)$ s
	maximum time spent waiting for a friend in Waiting for Someone	`maximum_waiting_time`	$x \sim \mathcal{U}(4.8 \cdot (0.5 + C),18) \times 10^3$ s
	modulation factor to determine FearEruption's intensity	`max_perceived_boom_intensity`	$18 \cdot (0.5{+}N)$
	size of friends group	`friends_group_size`	$x \sim \mathcal{U}(5,15)$
	social links values	`L,D,S,F,T`	$x \sim \mathcal{U}(0,1)$
	FearEruption reduction coefficient used by a friend when calming one agent	`fear_reduction_factor (friend)`	$0.001 \cdot \max(0.1, D)$
	FearEruption reduction coefficient used by a LEA when calming one agent	`fear_reduction_factor (LEA)`	$0.001 \cdot \max(0.1, N)$
	threshold above which a contagion process for FearEruption is activated	`FearContagionThreshold`	0.5
	threshold above which a contagion process for JoyPort is activated	`JoyContagionThreshold`	0.3
LEAs	*maximum distance of agent's perception*	`view_dist`	30 m
	maximum distance within which People agents are warned about the evacuation	`alert_population_radius`	150 m
	on foot speed	`walking_speed`	$x \sim \mathcal{U}(3,6)$ km/h
	on board of motorized vehicles speed	`vehicle_speed`	25 km/h
global	*threshold above which an agent is considered fearful*	`fear_threshold`	0.6
	threshold above which an agent is considered joyous	`joy_threshold`	0.3

Table 2. Complete listing of parameters used in the model.

RESULTS

In this section, we present the main plots and statistical analyses comparing the results across the three experimental settings. We specifically selected a set of observables that provide an intelligible representation of the ongoing dynamics in the simulation, allowing us to effectively monitor and interpret the system's behavior.

Figure 6 focuses on the evolution of fear in selected runs by showing the trend of the percentage of people (over the total) that can be considered fearful (i.e. with a fear intensity greater than 0.6). We can clearly observe the onset of fear as a consequence of the emission of roaring sounds by the volcano agent as well as the contribution due to the contagion dynamics. In the carried out runs, the maximum share of fearful people were in the range going from 8% to 60% for the IT-Alert scenario, while we found them to be between 20% and 70% for the door-to-door scenario.

In **Figure 7**, the detailed behavior of people agents during the whole evacuation process is depicted. The two figures on the top of **Figure 7** represents the same run pictured in **Figure 6**. We can clearly see how the behavior of agents can be impacted by the emotional status of the agents. For example, the early onset of strong sonic emission by the volcano in **Figure 7** (top-right) prompts a great number of agents to make a decision before receiving the evacuation order. For comparison purposes, we've put on the bottom-left of **Figure 7** an evacuation in the door-to-door scenario and on the bottom-right an evacuation with the emotional and social components of the model disabled.

As shown in **Figure 8** the warning process carried out on a door-to-door basis is slower than if it were done with the help of broadcast messaging. However, thanks to the absence of delays in the door-to-door evacuation scenario, we have that almost 80% of people is warned by the time IT-Alert would be issued. This could also be due to the high speed with which LEAs are able to move through the island, hence it lets them reach the majority of the population in a short period of time. Indeed, the warning of the remaining fraction of the population seems to be more challenging, as it is probably located in more remote areas.

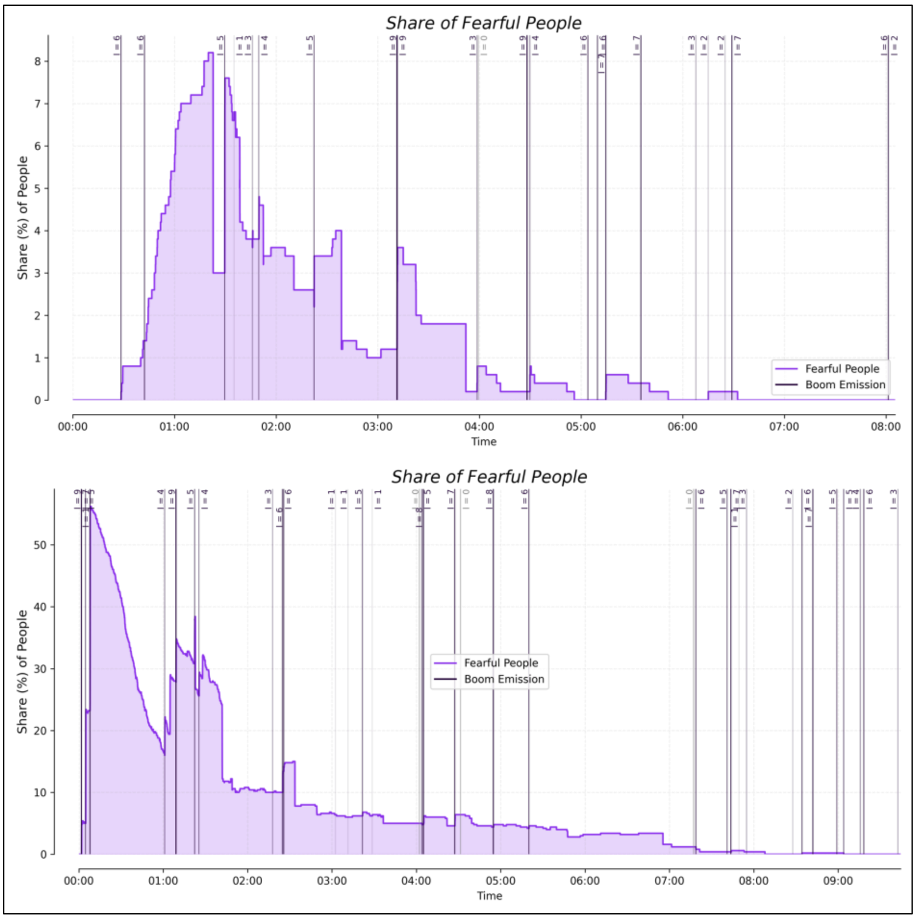


Figure 6. Percentage of fearful people over-time in the IT-Alert scenario.

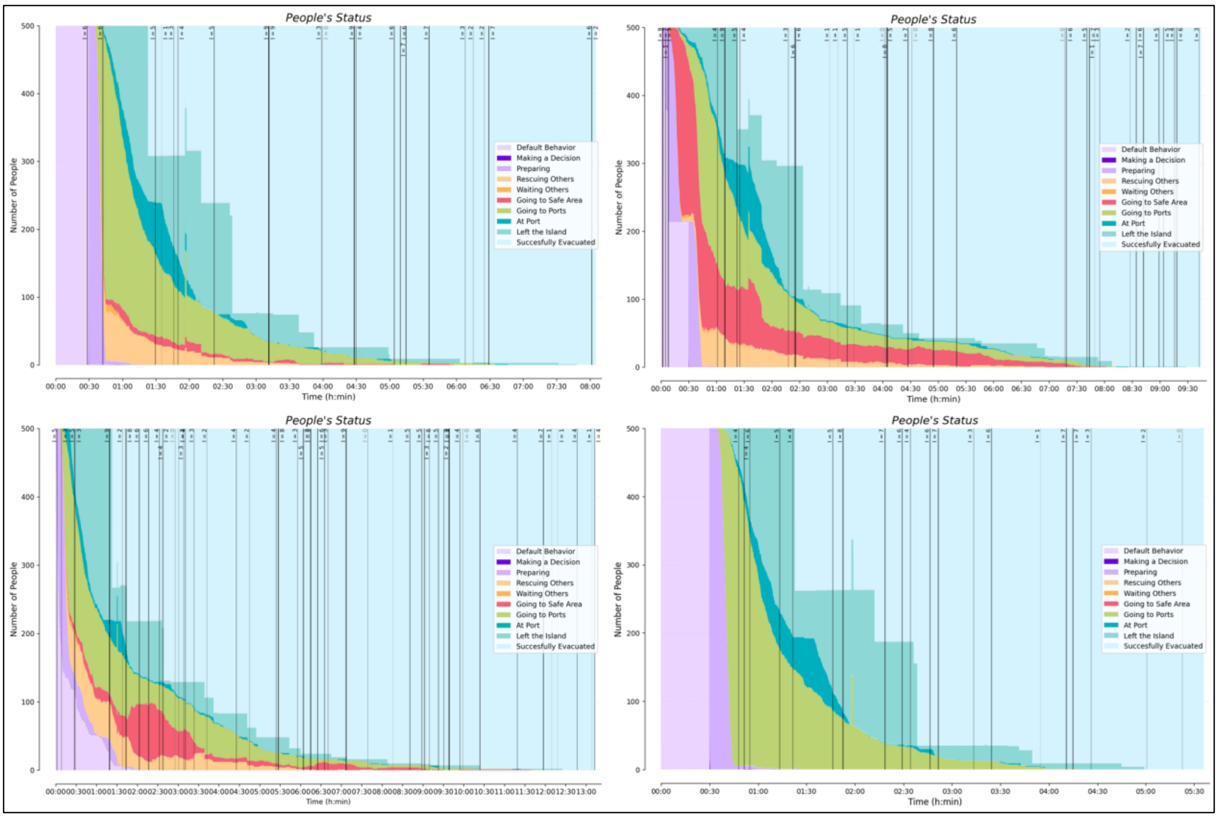


Figure 7. The evolution of people's behavior over a simulation, for the IT-Alert scenario with emotions enabled (top), for the door-to-door scenario (bottom-left) and the It-Alert scenario with emotions disabled (bottom-right).

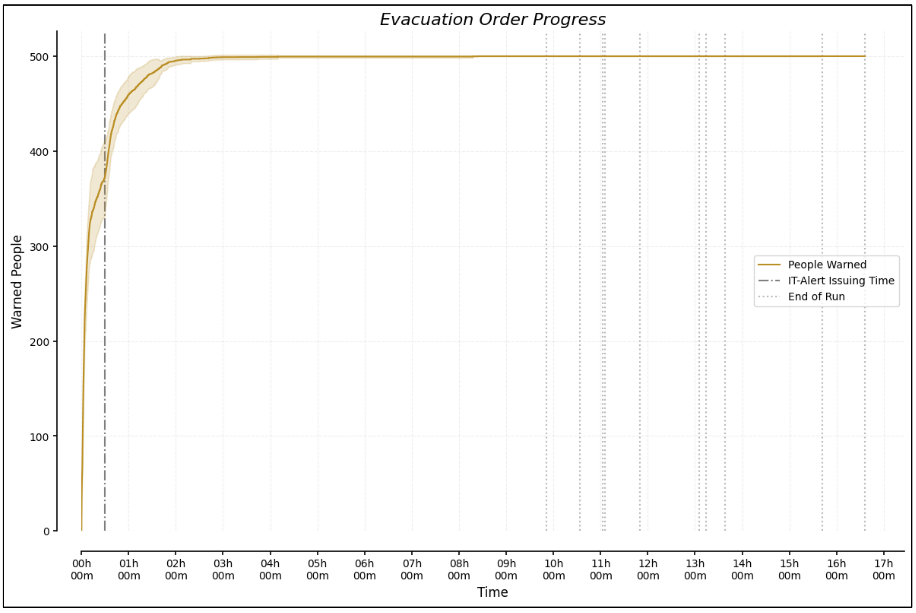


Figure 8. Average evacuation order propagation in the door-to-door scenario.

We may also compare the timeseries regarding state of preparation of the civilian population with the help of **Figure 9**, which depicts the progress of evacuation order reception and the degree of preparedness for civilians for a single run of both, the door-to-door and IT-Alert evacuation scenarios.

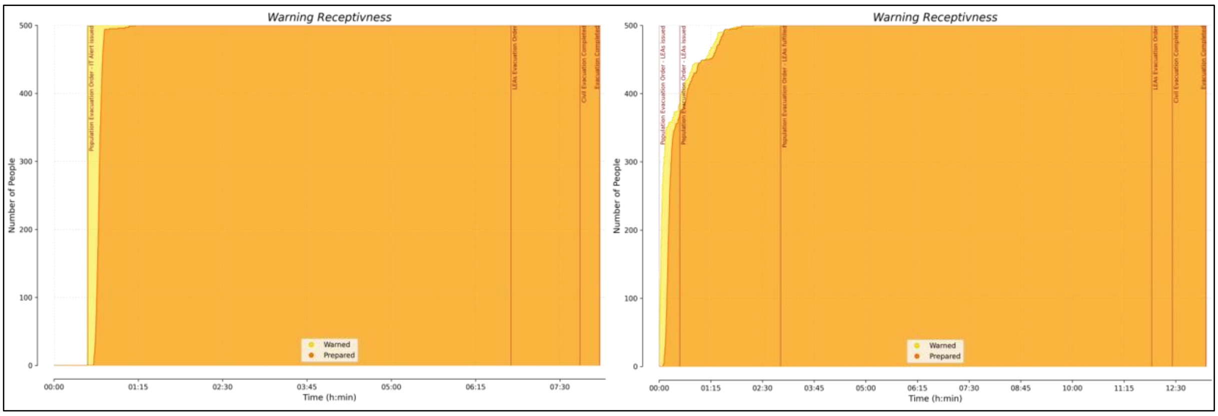


Figure 9. Evacuation order propagation and evacuees' degree of preparedness in a run from the IT-Alert (left) and the door-to-door (right), both with the emotional and social structure enabled.

The evacuation trend (i.e. the number of civilians who are still on the island) for the three scenarios is illustrated in **Figure 10**. Keeping in mind that an agent is considered to have left the island as soon as it sat foot on a ferry, it is easy to interpret the initial advantage of the door-to-door evacuation scenario over the IT-Alert ones as it just mirrors the absence of delays for the warning phase in such scenario. By comparing the two different IT-Alert scenarios it emerges that the presence of an emotional component can lead to an anticipation of the warning-issuance, although it is rapidly mitigated as people are involved in irrational behaviors.

A more comprehensive evaluation of the evacuation effectiveness can be obtained by examining **Figure 11** (top). Here, the average evacuation time over all runs is depicted, where the error bars represents the standard deviation computed over each set of runs. By evacuation time we refer to the length of the time interval incurring between the time of first issuing of a civil evacuation order by Civil Defense and the time of the last civilian evacuee to set foot on the port of Milazzo. If this time interval is extended to also include the complete evacuation of LEAs, results don't show much difference as depicted in **Figure 11** (bottom). We can observe how the presence of the emotional and social components in evacuees has two effects regarding the evacuation time: one hand it increases the time needed to fulfil the evacuation process while on the other hand it increases its variability. Also, the point-to-point systems to alert the population is found to be less efficient than a broadcast system, such as IT-Alert, as the resulting average evacuation time is higher.

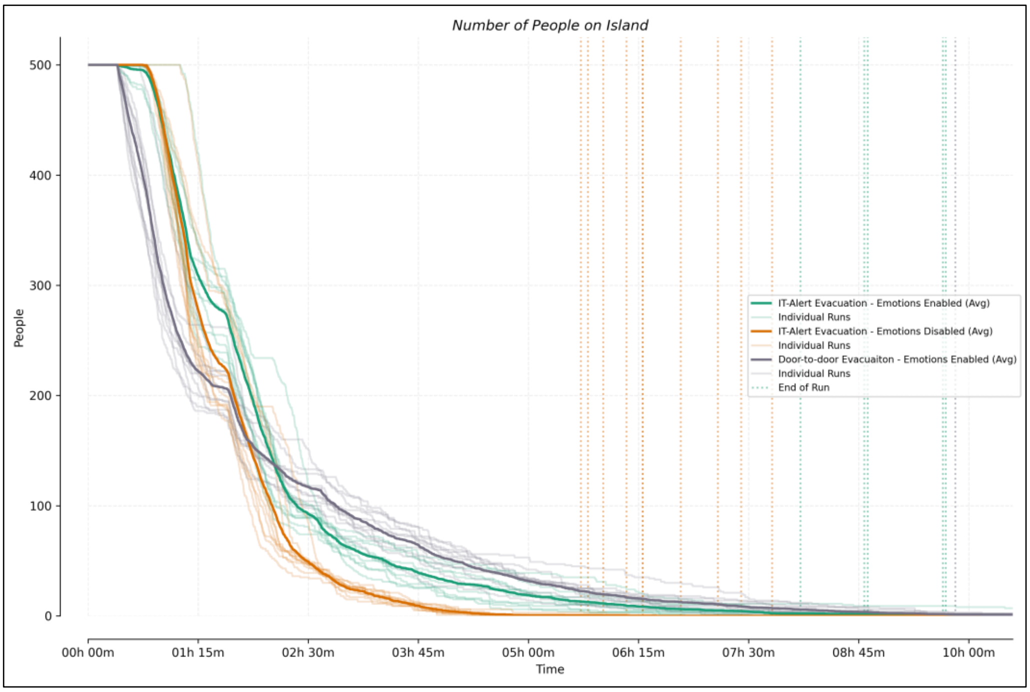


Figure 10. Trends for the number of people still on the island over the whole experiment. (for visualization purposes the horizontal axis has been cropped)

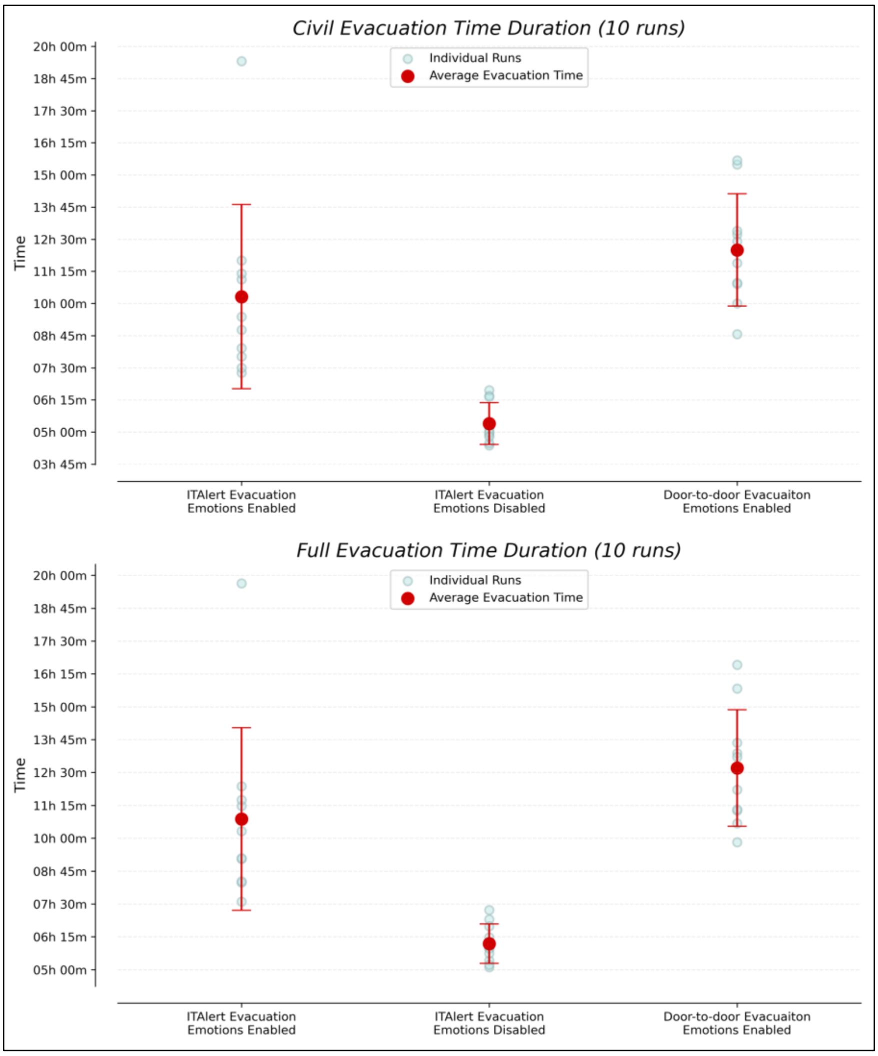


Figure 11. Average and individual evacuation times considering the civilian population only (top) and with the inclusion of Law Enforcement Agents (bottom).

The presence of an outlier in the IT-Alert evacuation scenario with the emotional and social components enabled stands out in **Figure 11**. Hence in this section we aim at investigating it further with the help of **Figure 12**.

From **Figure 12** (top-right), we can assess that it is just a small number of individuals that causes such a longer evacuation time and by looking at **Figure 12** (top-left), we can see how all the remaining civilians on the island are scared and are attempting to reach any safe area, not necessarily a port. This is interesting as it is a good example of how the emotional architecture can impact the evacuation time substantially. The consequent presence of irrational behavior can lead to hardly predictable inefficiencies and delays and is therefore worthy of investigation, as they must be taken into account when designing evacuation policies.

However, it must be observed that in a real-world scenario, if few people were still to be evacuated and they were directed towards an area patrolled by the evacuation authorities, it's highly likely they would be promptly escorted to the nearest port, speeding up the conclusion of the evacuation.

In ***Table 3***, all evacuation time results have been summarized.

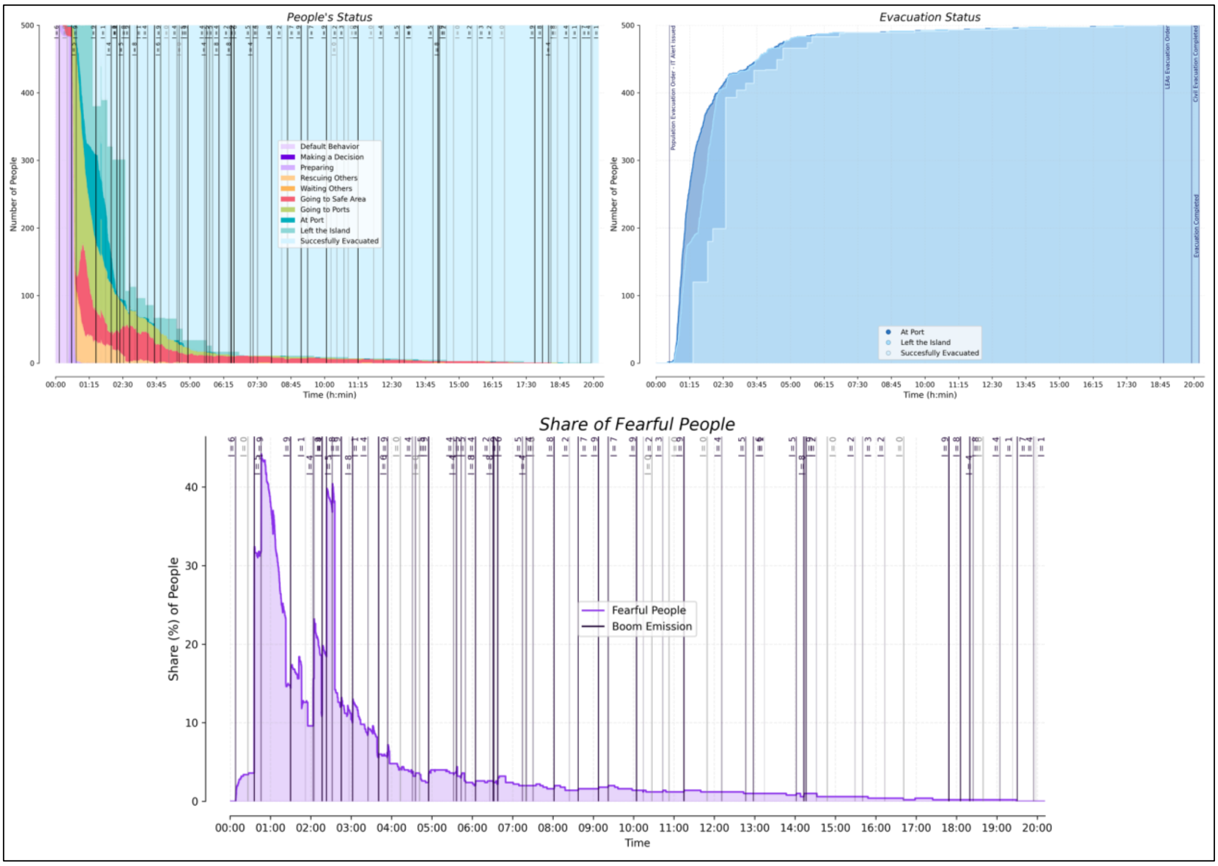


Figure 12. In-depth view of the outlier coming from the IT-Alert emotions enabled scenario, containing the civilians' behavior status over-time (top-left), their evacuation status (top-right) and their emotional status limited to fear (bottom).

	Civil Evacuation Time		**Full Evacuation Time**	
	Mean	*Standard Deviation*	*Mean*	*Standard Deviation*
IT-Alert – Emotions (outlier removed)	10h 16m (9h 15m)	3h 35m (1h 40m)	10h 44m 9h 44m	3h 29m (1h 35m)
IT-Alert – No Emotions	5h 20m	49m	6h	45m
No IT-Alert - Emotions	12h 4m	2h 11m	12h 40m	2h 13m

Table 3. Evacuation time-related results summary.

Future Developments

To further enhance the model's realism and complexity, several key areas for future development have been identified. Firstly, the emotional range of civilian agents could be expanded beyond simple fear and joy to allow for more nuanced and realistic behaviors. For example, emotions related to the social structure, such as 'happy for' someone or 'sorry for' someone, could be incorporated. On a similar note, it would be interesting to introduce a variety of new irrational behaviors. A further development related to the emotional section of the model lies in the integration of the social and emotional engine into LEAs as well, although with personality traits that differs from the civilian population's. Concurrently, incorporating a system of social norms and obligations is important, as the integration of a system of sanctions to prevent civilians from engaging in dangerous or unwanted behaviors.

Another critical area of focus is the development of a more detailed volcano phenomenology, which would introduce a dynamic threat level and allow for more realistic evacuation scenarios. In this context, the model must be capable of handling the possibility of an unusable evacuation infrastructure following the occurrence of severe events. Finally, in this regard, the model might be also extended to agents (e.g. humans, evacuation vehicles) who can be physically harmed, and to infrastructures, such as roads, can be interrupted, in order to introduce a new level of risk.

In addition, the model could be further expanded to include more complex behaviors for LEAs, such as a non-stationary patrol of certain areas, and evacuation, such as the rescuing of people in need with the use helicopters. Not only, as another area of development is the environment. For instance, the inclusion of different and changing maritime and weather conditions, or the different treating of roads and walking trails, could represent two suitable and relevant improvements.

Conclusions

This work presented an agent-based model designed to reproduce the evacuation of Vulcano Island during a volcanic emergency, with the explicit goal of integrating emotions and social relationships into the decision-making process of the agents. By going beyond the rational behavior of agents, the model captures a richer spectrum of possible individual responses, including hesitations, irrational choices and the influence of social contagion.

The results demonstrate that emotions and social structure shape the evacuation dynamics, not only by increasing variability in evacuation times, but also by producing situations in which small groups of individuals may delay the overall process. This underlines the necessity of acknowledging psychological and social complexity when planning for real-world crisis scenarios.

Equally important is the comparison between the IT-Alert and the door-to-door scenarios. The simulations reveal that the IT-Alert system manages to rapidly disseminating evacuation orders across the entire populations. However, door-to-door communication, though slower in reaching isolated individuals, is still able to mobilize a large amount of civilians during the early stages of the emergency. This suggests that an hybrid strategy, which combines the immediacy of broadcast alerts with the targeted effectiveness of a point-to-point interaction, may offer the most robust solutions for evacuation management as it mitigates the single point of failure risk inherent in broadcast messaging.

The findings highlight the relevance of integrating behavioral realism into evacuation modeling. Beyond the specific case of Vulcano, this approach may lead to a better understanding of how people respond to policies enacted by authorities during an evacuation. By explicitly integrating emotions, social influence and communication strategies in the analysis, the model provides a framework that can support evacuation authorities in designing evacuation plans that are more complete and, hopefully, effective as more attention is given to the complexity of human behavior.

References

[1] M. Bourgais, M. Taillandier and L. Vercouter, "BEN: An Architecture for the Behavior of Social Agents," *Journal of Artificial Societies and Social Simulation,* vol. 23, no. 4, 2020.

[2] Dipartimento Protezione Civile, "Piano nazionale di protezione civile per il rischio vulcanico sull'isola di Vulcano," 2021.

[3] C. Bonadonna et al., "Assessing the effectiveness and the economic impact of evacuation: the case of the island of Vulcano, Italy," *Nat. Hazards Earth Syst. Sci,* vol. 22, pp. 1083-1108, 2022.

[4] J. Svennevig, Getting Acquainted in Conversation, Amsterdam: John Benjamins Publishing, 2000.

[5] R. R. McCrae and O. P. John, "An introduction to the five-factor model and its applications," *Journal of Personality,* vol. 60, no. 2, pp. 175-215, 1992.

[6] T. Bosse et al., "Multi-agent model for mutual absorption of emotions.," in *European Conference of Modelling and Simulation 2009*, Madrid, 2009.