# A Model of Radicalization Growth using Agent-based Social Simulation

Tasio Méndez[1], J. F. Sánchez-Rada[1], Carlos A. Iglesias[1], and Paul Cummings[2]

[1] Intelligent Systems Group, Universidad Politécnica de Madrid, Spain
[2] {tasio.mendez, jf.sanchez, carlosangel.iglesias}@upm.es
http://www.gsi.dit.upm.es
[3] Krasnow Institute, GMU Computational Social Fairfax, VA
{pcumming2}@gmu.edu

**Abstract.** This work presents an agent based model of radicalization growth based on social theories. The model aims at improving the understanding of the influence of social links on radicalism spread. The model consists of two main entities, a Spread Model and an Agent Model. The Spread Model updates the agent relationships based on proximity and homophily, it simulates information diffusion and updates the agents' beliefs. The model has been evaluated implemented in Python with the agent-based social simulator Soil. In addition, it has been evaluated using a sensitivity analysis.

**Keywords:** Radicalization · Terrorism · agent-based social simulation.

## 1 Introduction

Research works on political terrorism began in the early 1970s. These works were focused on collecting empirical data and analyze it for public policy purposes. Terrorist activity was usually attributed to personality disorders or "irrational" thinking [1]. However, this approach has been surpassed and there are many issues that should be considered additionally.

Many scholars, government analysts and politicians point out that since the mid 1990s terrorism has changed. This terrorism is distinguished from the "old" one in the fact that is motivated by religious beliefs and is more fanatical, deadly, and pervasive. It also differ in terms of goals, methods and organization [1, 2].

The fact that some of the drivers of current terrorism involve not only political or religious interests but also include fanaticism, makes it a complex process of radicalization. This radicalization process consists in the progressive adoption of extreme political, social or religious ideals. Nevertheless, this process does not always lead to violence acts such as terrorism [3].

It is of vital importance to understand the properties of radicalization with the purpose of anticipating that violence. The main issue about the fact of understanding how these organizations work is that the information is not always available and if it is available, it is incomplete or inaccurate, what makes it even harder.

One common approach to face terrorism is trying to understand its roots, motivations and practices. In particular, nowadays is of vital importance to understand how terrorist organizations recruit new members and isolate them. Moreover, terrorist organizations have used effectively social media and social networks for interacting with the goal of expanding their networks through real-time information exchange.

As the society and the new forms of communications evolve, terrorists are now able to develop new forms of organization for their purposes. Organizations can thus flatten out their pyramid of authority and control and approach a network form, a group of more or less autonomous, dispersed entities, linked by communications and perhaps nothing more than a common purpose [4]. Thus, terrorist organizations can be modelled as Social Networks (SNs) where vertices represents members of the organization and links represent communication between members.

As terrorist organizations approach a network form and can be modelled as SNs, a research based on Agent-based Social Simulation (ABSS) could be a good starting point for understanding the information flow within the network.

This paper proposes an agent-based model of a terrorist organization growth which has been implemented in Soil [5], an agent-based social simulator designed for modelling social networks.

This remainder of the paper is structured as follows. Sect. 2 introduces the ABSS Soil, paying special attention to its modelling approach as well as specific features developed for modeling problems with a geographical component, as it happens in the radicalization process. Sect. 3 introduces the agent-based model of radicalization. Sect. 4 describes the implementation of the model using Soil, and provides an overview of the simulation results, including a sensitivity analysis of the simulation results to evaluate the developed model. Finally, some conclusions and insights are presented in Sect. 5.

## 2 ABSS Soil

Soil [5] is a modern ABSS for modelling and simulation of SNs. It has been applied to a number of social network simulation models, ranging from rumour propagation to emotion propagation and information diffusion. Each simulation consists of users represented by agents and a network that represents the social links between users.

Agents are characterized by their state and the behaviours they can carry out in every simulation step, usually depending on user state. Each behaviour defines the actions carried out and how the agent state evolves, depending on external factors or social factors. Those external or social factors are controlled by environment agents, which are not assigned to any network node.

The main reason for using this simulator is that most ABSS platforms do not provide support for the analysis of social networks [5]. Two exceptions being Krowdix and Hashkat.

HashKat [6] is a C++ ABSS platform specifically designed for the study and simulation of social networks. It includes facilities for network growth and information diffusion, based on a kinetic Monte Carlo model. It exports information to be processed by machine learning libraries such as NetworkX [7] or R's iGraph [8] and network visualization with Gephi [9].

Krowdix [10] is a Java ABSS for social networks but it is not open source. It uses JUNG [11] for network functions and JFreeChart [12] for visualization. The simulation model considers users, their relationships, user groups and interchanged contents.

As Krowdix, most ABSS platforms are programmed in Java, while Soil uses Python that given its increased popularity, it has very gradual learning curve, readability, clear syntax and availability of libraries for network processing and machine learning. Soil is based in NetworkX, which is the defacto standard library for Social Network Analysis (SNA) of small to medium networks.

NetworkX provides functionalities for manipulating and representing graph models, generators of classical and popular graph models, including generators for geometric graphs, and graph algorithms for analyzing graph properties. In addition, NetworkX is interoperable with a great number of graph formats, including GEXF, GML, GraphML and JSON among others.

### 2.1 Architecture

We propose a simulation model of SNs consisting of users represented by agents and a network that represents the social links between users. Agent are characterized by their state (e.g. infected) and the behaviours they can carry out in every simulation step, usually depending on the user state. Each behaviour defines the actions carried out (e.g. tweeting, following a user, etc.) and how the agent state evolves, depending on external factors (e.g. news about a topic) or social factors (e.g. opinion of their friends). Probabilities defined in the configuration control the frequency of actions in every behaviour.

This simulation model has been implemented in the architecture shown in Fig. 1 and consists of four main components.

The *NetworkSimulation* class is in charge of the network simulator engine. It provides forward-time simulation of events in a network based on nxsim [4] and Simpy [13]. Based on configuration parameters, a graph is generated with NetworkX and an agent class is populated to each network node. The main parameters are the network type, number of nodes, maximum simulation time, number of simulations and timeout between each simulation step.

The *BaseAgentBehaviour* class is the basic agent behaviour that should be extended for each social network simulation model. It provides a basic functionality for generation of a JSON file with the status of the agents for its analysis with machine libraries such as Scikit-Learn [14].

The *SoilSimulator* class is in charge of running the simulation pipeline defined in Sect. 2.2, which consists in running the simulation and generating a

[4] https://pypi.python.org/pypi/nxsim

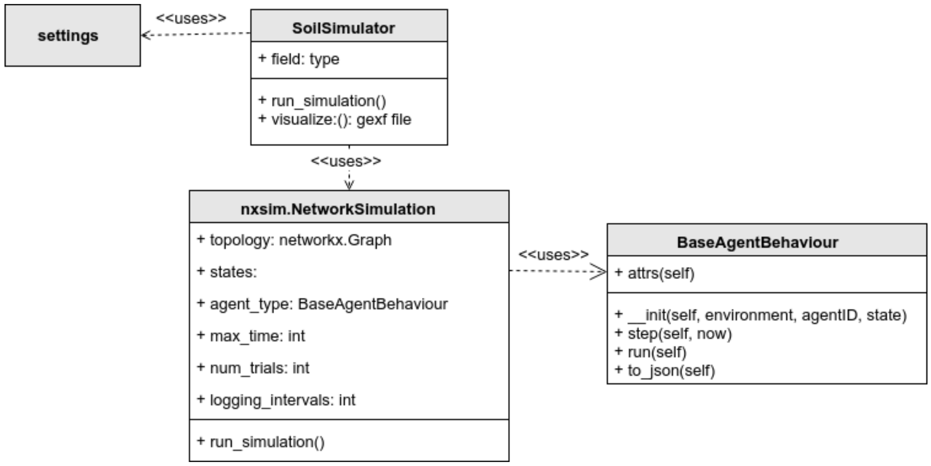



**Fig. 1.** Simulation components

visualisation file in Graph Exchange XML Format (GEXF) which can be visualised with Gephi. In addition, interactive analysis can be done with IPython web interface.

*Settings* groups the general settings for simulations and the settings of the different models available in Soil's simulation model library.

### 2.2 Simulation workflow

An overview of the system's flow is shown in Fig. 2. The simulation workflow consists of three steps: configuration, simulation and visualization.

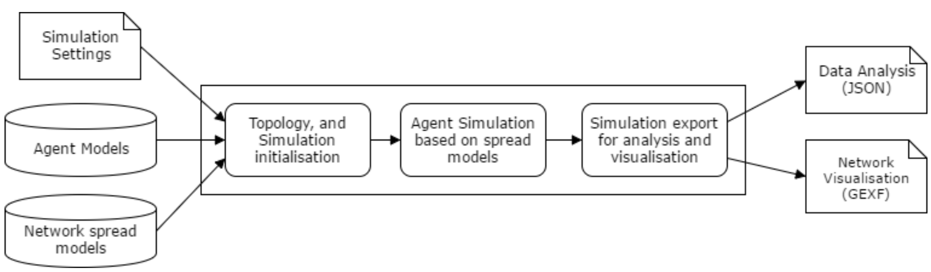



**Fig. 2.** Social simulator's workflow

In the first step, the main parameters of the simulation are configured in the *settings.py* file. The main parameters are: network graph type, number of agents, agent type, maximum time of simulation and time step length. In addition, the parameters of the behaviour model should be configured (e.g. initial states or probability of an agent action). Agent behaviours should be selected from the provided library or developed extending the *BaseAgentBehaviour* class.

Once the simulation is configured, the next step is the simulation, that can be done step by step or a number of steps. The class *BaseAgentBehaviour* stores the status of every agent in every simulation step into a JSON file to be exported once the simulation is finished. This allows us to automatise the process of generating the .gexf file.

Finally, users can carry out further analysis with the JSON file as well as visualize the evolution the simulation with the generated .gexf file with Gephi.

## 3 Radical Simulation Model

### 3.1 Problem

As previously discussed, in the last years, the way people communicate has changed, becoming more relevant social networks, where everyone can exchange messages, images and videos. Terrorist organizations also have moved forward by setting up radio stations, TV channels or Internet websites. These activities allow them to increase their strength, their funds and better recruit new people.

Since terrorist organizations can be modeled as social networks we can study how information is shared and how people become members of groups or even new relationships. Within the proposed model (Sec. 3.2), terrorist groups will be represented as graphs where vertices represent members and edges represent communication between those members.

However, radicalism is not only sustained by flow information. Multiple causes, rather than a single cause should be considered, including social and spacial relations which evolve over time. Estimating their evolution is important for management, command and control structures, as well as for intelligence analysis research purposes. By knowing future social and spacial distributions, analysts can identify emergent leaders, hot spots, and organizational vulnerabilities [15].

In order to approach to the radicalism spread, a spatial distribution is used based on Geometric Graph Generators [16], which provides geographical positions to agents, being able to manage real environments.

The physical space aims to produce more insightful results when considering the spread of terrorism [17]. Properties of space and place are vital components of terrorist training, planning, and activities.

Besides, based on the principle of homophily, as a contact between similar people occurs at a higher rate than among dissimilar people, it is more likely to have contact with those who are closer to us in geographic location than those who are distant [18]. It is theorized that, in general, close proximity in geographic space strongly influences closeness in social space [17].

As it was discussed above, the proposed model will try to approach to the fact of the rise of radicalism within a specified geographic area considering real geographical connections between members.

### 3.2 Model development

Three levels of analysis are widely accepted for the radicalization process [19]: *micro-level* (i.e. the individual level involving feelings of grievance, marginalization, etc.), *meso-level* (i.e. the social environment surrounding radicals and the population and lead to the formation of radical groups), and *macro-level* (i.e. impact of government policies, religion, media, including radicalization of the public opinion and political parties).

The model here proposed is focused on analyzing the macro-level, including limited aspects of the micro level (such as the vulnerability level).

Several aspects have been considered for modeling the radicalism growth at the meso level. First, the model considers the impact of *havens* [20] and *training areas* [21]. Havens, also known as sanctuaries, provide radical groups the possibility to obtain long term funding and serve the purposed of solidifying group cohesion. Terrorist training camps aim at providing indoctrination and teaching for terrorism and are distributed around the world. They foster group identity formation and group cohesion, and require geographical isolation and easy access to weapons.

The modelling of the radicalism spread involves population and places as it was discussed above. People can play two roles: (1) population as the people that can be radicalized and (2) terrorist that spread their message to locals and try to recruit civilians to join the terrorist network.

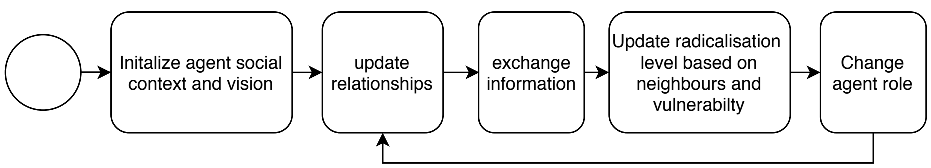



**Fig. 3.** General workflow of the simulation

Based on a previous model proposed by Cummings [17], terrorists have little opportunities for effective training, planning, and other logistic necessities. Those areas are modelled by (1) training environments, which increase the influence to the nodes that are attached to them, and (2) heavens where people is save. The nodes that are joined to havens get less influenced if the heaven is not radicalized, but it could get radicalized and its behaviour will change.

For implementing the environment described, we will use four different agents that interact with each other.

- Spread model in charge of the information flow which determine the state of population. Each node contains a threshold where once reached, the node is marked as informed and it will pass from a civilian state to a radical state.
- Network model in charge of controlling spacial and social relations between population.

- Heavens model which will modify nodes vulnerability depending on heaven state as it is going to be explained below.
- Training areas model which will decrease neighbouring nodes vulnerability.

The network consists on $N$ nodes that have two coordinates, as since Geometric Graph Generators [16] are used, that position each node on a map. The edge between two nodes, indicates direct bidirectional communication between both of them.

All agents are assumed to have similar parameters but are heterogeneous in their representation. Within the spread model, each node develops its own belief about whether the information is valid by calculating weighted mean belief $B_i$ from it neighbors, and combining that with its initial belief $B_0$, which is normalized between 0 and 1 [22]. In addition, in every step two agents will exchange information given a probability of interaction.

The mean belief is calculated given its own vulnerability and the neighbours influence as well as the information spread intensity ($\alpha$) which is also normalized and consider how much information is exchanged in every step of the simulation.

$$B_e = \sum_{i=0}^{n} \frac{B_i\ D_i}{\sum_{j=0}^{n} D_j} \tag{1}$$

The node influence $D_i$ parameter has been included in Eq. 1 – where $n$ is the number of neighbours of the node – as the change in behavior that one person causes in another as a result of an interaction [23] measured as degree centrality that is defined as the number of adjacencies upon a node, which is the sum of each row in the adjacency matrix representing the network. It can be interpreted within social networks as a measure of immediate influence – the ability to infect others directly or in one time period [24]. This SNA function returns values that are normalized by dividing by the maximum possible degree in a simple graph $N - 1$ where $N$ is the number of nodes in G.

$$B_n = B_e\ \alpha + B_0\ (1 - \alpha) \quad ; \quad 0 \leq \alpha \leq 1 \tag{2}$$

As it was explained above, in Eq. 2 the parameter to indicate the information spread intensity is included. When its value is 0%, no information is exchanged and when it increases, the knowledge diffusion grows.

$$B_i = B_n\ N_v + B_0\ (1 - N_v) \quad ; \quad 0 \leq N_v \leq 1 \tag{3}$$

The node vulnerability ($N_v$) parameter is included in Eq. 3 as the extent to which individuals conform or adopt variable attributes such as opinions from their attached nodes. In other words, if $N_v = 1$, the node will be fully influenced by their connected nodes, where a value of $N_v = 0$, would mean it would not be influenced by connected nodes, so no change in the network is expected. Thus, individuals who are merely sympathetic may be influenced to more extreme opinions by their friends after they join the terrorist network.

Once the mean belief developed by the agent reach the threshold, it is marked as informed and it will change its state from civilian to radical. Every agent in radical state will be only influenced by radical agents since the radical experience no restraining influence from non-radicals [25]. Furthermore, once an agent is in the radical state, the information spread intensity will began to value 100%, as once you are radical the most information you get from another radical agents.

With the purpose of determining the most important nodes within the terrorist network, they are marked as *leaders* based on the SNA function: betweenness centrality [22], that is defined of a node $v$ as the sum of the fraction of all-pairs shortest paths that pass through $v$.

As node vulnerability ($N_v$) was explained above, training areas and havens will modify this attribute depending on their status. Training areas will decrease the parameter from its neighbours, where a value of 1 for training area influence will make all its neighbours fully vulnerable. However, a value of 1 for haven influence will make invulnerable all its neighbours when the state of the haven is not radical. Nevertheless, once the haven is marked as radical, its behaviour will be similar to training areas.

Finally, the network model in charge of controlling spacial and social relations takes into account that agents have the opportunity to interact with other agents. They select an agent to interact with according to a probability of interaction – different from the one mentioned above – based on two parameters: (1) social distance and (2) spatial proximity.

On one side, social distance (SD) take into account the fact that if two agents must cross many social links, then the probability should be low and vice versa. It compute it by finding the shortest path between to agents and then dividing one by the number of links in that path.

$$SD_{i,j} = \frac{1}{|A\ A_{i,j}|} \tag{4}$$

where $|A\ A_{i,j}|$ is the shortest path from $i$ to $j$. When computing the social distance, each agent can only reach all those nodes that are withing its sphere of influence parameter. An agent can recognize and distinguish the closeness of other agents withing the sphere of influence, but it can't differentiate the closeness when the interacting agent is outside the perimeter.

On the other side, spatial proximity (SP) takes into account that two agents at the same location are more likely to talk than being in different locations. Some might argue that SP is not significant in the Internet age. However, in the terrorism domain, attending the same training area or the same location is a critical interaction indicator [15].

As Geometric Graph Generators returns coordinates normalized between 0 and 1, the probability of being at the same location will be computed as the inverse of the distance between two agents.

$$SP_{i,j} = (1 - |d_{i,j}|) \tag{5}$$

where $|d_{i,j}|$ is the distance between the nodes. Like in SD the probability is bounded by a sphere of influence parameter, in SP the probability will be bounded by a vision range parameter. All agents outside this perimeter will be unreachable by the current agent.

Once defined both parameters, we can compute the probability of interaction that it will be calculated as following.

$$P_{i,j}^{Interaction} = \omega_1\ SD_{i,j} + \omega_2\ SP_{i,j} \tag{6}$$

where $\omega_1$ and $\omega_2$ are the weights of SD and SP respectively with the purpose of customizing the environment.

None of the parameters will limit the probability of interaction. Thus, the candidate agents will be the sum of all the agents that are inside the perimeter of the sphere of influence or the vision range.

**Table 1.** Simulation input parameters.

| **Model** | **Name** | **Implication** |
|---|---|---|
| Terrorist Spread | information_spread_intensity | The amount of information exchanged in every step of the simulation. |
| | terrorist_additional_influence | Additional influence added to agents whom status is radical. |
| | min_vulnerability | The minimum vulnerability that an agent could have (*default 0*). |
| | max_vulnerability | The maximum vulnerability that an agent could have. The allocation of this parameter follows a continuous uniform distribution. The maximum value that this parameter can take is the unit. |
| | prob_interaction | The probability that two agents exchange information in one step. |
| Training Area | training_influence | The influence that a training area applies to its neighbours. |
| Haven | haven_influence | The influence that a haven applies to its neighbours. |
| Terrorist Network | sphere_influence | The maximum number of social links that an agent can cross for a new interaction. |
| | vision_range | The range on the spatial-route network specifying the maximum distance an agent can move for a new interaction. |
| | weight_social_distance | The weight of social distance (SD) to calculate the interaction probability. |
| | weight_link_distance | The weight of spatial proximity (SP) to calculate the interaction probability. |

Thereby, an agent can establish a new way of communication with its candidate agents, so the probability of interaction is calculated between each agent and its candidate agents.

As it was explained, the aim of the model is trying to approach to the fact of the radicalism spread withing a specified geographic area. For that reason, in Table 1 all parameters of the simulation are detailed for representing a scenario as real as possible. Aside from all the parameters explained, the network can be modelled using one of the random network generation methods from NetworkX. It is also possible to control the ratio of each type of agent.

## 4 Experimental results

The model has been implemented using the Soil Simulator as it was discussed above. The scenario represents a specified geographic area that can be customized with the purpose of approaching a real scenario.

Every agent exchange information several times during the simulation and every portion of time is known as *step*. One one hand, in every step an agent belonging to the Network Model will update its relationships based on the input parameters. After this action, the control is passed to the Spread Model that will be in charge of how the information will flow in that step. The current agent will be influenced by its neighbours depending on their internal parameters values.

On the other hand, if the current agent is a heaven or a training area, the step will consist on modifying the internal parameters of their neighbours as it was explained in the previous section.

With the purpose of making the simulations more interactive, a web application has been developed using D3.js [26] for visualizing the results. As we can notice in Fig. 4 the simulation returns a graph that is presented in the main area of the web application. The graph can be positioned in a map, and it could be represented depending on the step, being able to see it evolve over time. Furthermore, the interface allows users filtering the results or changing the simulation parameters.

The application not only allows the user to visualize the results, it also provides statistics and the option of running more simulations changing the input parameters as it is displayed in Fig. 5. The web application also allows users to export the results of the simulation in different formats such as GEXF [27] or JSONGraph[5] to be analyzed with any other tool.

The model has been evaluated using two different sensitivity analysis methods. The first one is a local approach known as One-at-Time (OAT) approach, that studies small input perturbations on the model output. To bring about this method, 1.000 simulations have been launched with different input values and have been analyzed using the Seaborn [28] library available for Python for exploring and understanding the results.

[5] http://netflix.github.io/falcor/documentation/jsongraph.html

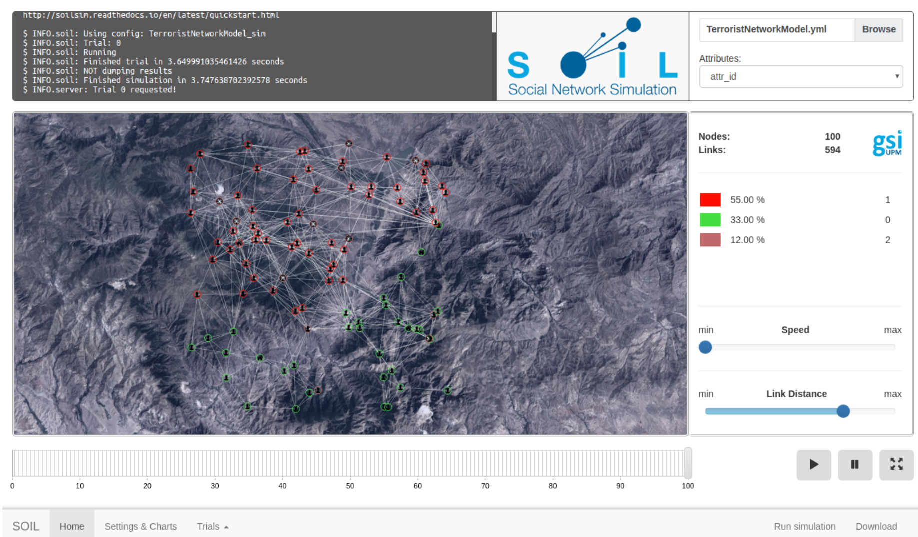



**Fig. 4.** Visualization of the simulation

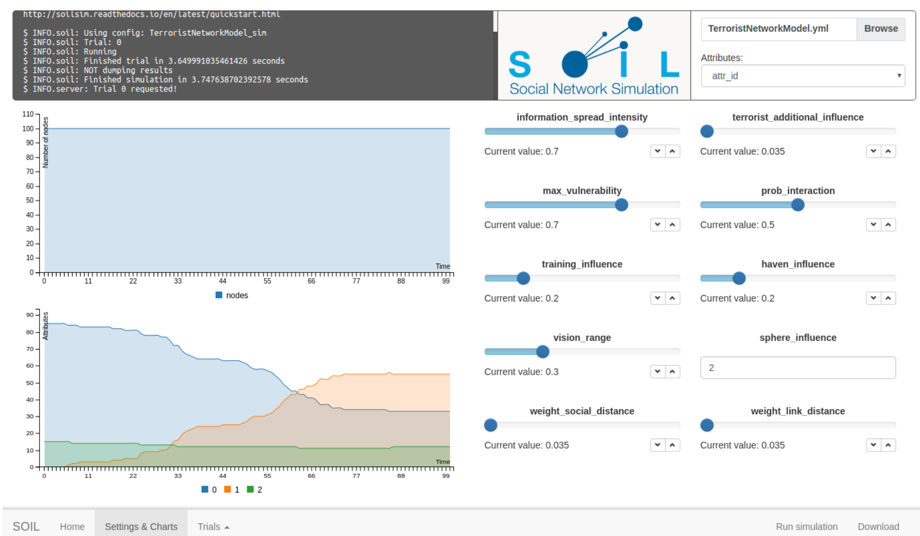



**Fig. 5.** Visualization of the simulation

The other method applied is the Morris method [29] that is referred to as "global sensitivity analysis" that in contrast to local sensitivity analysis, it considers the whole variation range of the inputs [30]. This method is computed using the SALib [31] library for Python.

The primary model outputs of interest for the sensitivity analysis are the radical population understood as the number of agents that have become radical from those who were not radical at the beginning and the mean radicalism within the network.

Both outputs will be measured taking into account different types of simulations. On one side, the network model will be studied assuming that the spread model inherit the another. On the other side, three different topologies (small world, scale free and random clustered) will be analyzed.

In Table 2 the Morris indices are detailed for the network model and mean radicalism output order by $\mu^*$. A total of 200 trajectories were built for the model which results in 1.800 samples. Fig. 6 plots results on the graph $(\mu^*, \sigma)$ and identifies the probability of interaction, the maximum vulnerability and the information spread intensity as the strongest influence on the mean radicalism within the network.

The analysis have been made using a random clustered topology that is created based on proximity between nodes for 100 nodes, and with same number of radical agents at the beginning.

However, taking into account the population radicalized in a simulation as we can notice in Table 3 and Fig. 7 are similar, but the maximum vulnerability and the information spread intensity is in this case more influential than the probability of interaction.

Morris indices for the three different topologies have similarities as the weight of the radical agents for the distribution through the network is the most influential parameter for both outputs as it can be notices in Fig. 8 for Scale Free and Small World topologies. In addition, the model output linearly depends on the weight of the agents because $\sigma_j \ll \mu_j^* \forall_j$. Nevertheless, the size of the network have no influence on the two model outputs.

The furthest point is the weight of the radical agents and it has a linear influence on the input while the closest points to zero are the size of the network and the input parameters for modeling the topology.

## 5 Conclusions

Understanding radicalization roots is a first step for being able to define and apply suitable counter-terrorism measures. There are many challenges for analyzing terrorism networks, given the lack of public datasets and the sensibility of this information. Nonetheless, the application of agent based social simulation is an effective technique for modeling non linear adaptive systems, and they enable analyzing and validating social theories of the radicalization process.

In this work we present a model and a tool for agent-based modeling of radical terrorist networks. We have propose building the agent-based model around two main concepts, the Network Model and the Agent Model. While the first is in charge of managing agent relationships, the second defines the specific behaviour of every agent. This approach has been applied for modeling terrorist growth. The proposed model is focused on analyzing the impact of the information exchange and environmental radicalization in the radicalization process.The evaluation and analysis of the simulation results provides insight regarding the importance of the simulation parameters, including the network characteristics.

## Acknowledgements

This work is supported by the Spanish Ministry of Economy and Competitiveness under the R&D projects SEMOLA (TEC2015-68284-R), by the Regional Government of Madrid through the project MOSI-AGIL-CM (grant P2013/ICE-3019, co-funded by EU Structural Funds FSE and FEDER); by the European Union through the project Trivalent (Grant Agreement no: 740934) and by the Ministry of Education, Culture and Sport through the mobility research stay grant PRX17/00417.

**Table 2.** Morris indices for network model and mean radicalism output.

| Parameter | $\mu$ | $\mu^*$ | $\sigma$ |
|---|---|---|---|
| prob_interaction | 0.320 631 | 0.367 384 | 0.517 95 |
| max_vulnerability | 0.243 827 | 0.349 831 | 0.413 981 |
| information_spread_intensity | 0.252 602 | 0.324 202 | 0.379 572 |
| terrorist_additional_influence | 0.036 039 | 0.128 335 | 0.206 991 |
| weight_social_distance | −0.004 388 | 0.110 129 | 0.186 007 |
| vision_range | 0.019 502 | 0.109 09 | 0.180 97 |
| sphere_influence | 0.006 756 | 0.107 522 | 0.173 183 |
| weight_link_distance | 0.007 996 | 0.101 815 | 0.179 93 |

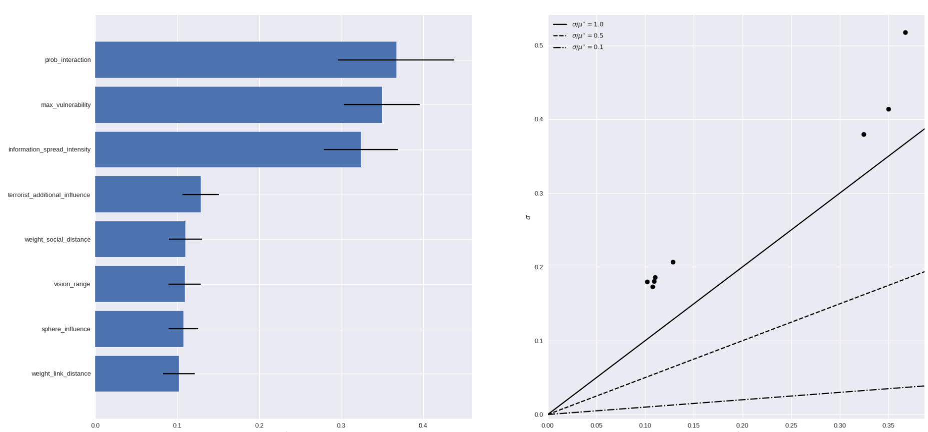



**Fig. 6.** Morris method results representation for network model and mean radicalism output for 200 trajectories

**Table 3.** Morris indices for network model and radical population output.

| Parameter | $\mu$ | $\mu^*$ | $\sigma$ |
|---|---|---|---|
| max_vulnerability | 0.466 355 | 0.484 857 | 0.596 371 |
| information_spread_intensity | 0.392 325 | 0.402 566 | 0.541 922 |
| prob_interaction | 0.268 707 | 0.331 403 | 0.568 499 |
| terrorist_additional_influence | 0.092 038 | 0.186 473 | 0.415 794 |
| weight_link_distance | −0.012 333 | 0.181 102 | 0.401 011 |
| vision_range | −0.001 680 | 0.176 981 | 0.380 602 |
| sphere_influence | 0.005 437 | 0.169 812 | 0.358 775 |
| weight_social_distance | 0.003 899 | 0.165 475 | 0.375 792 |

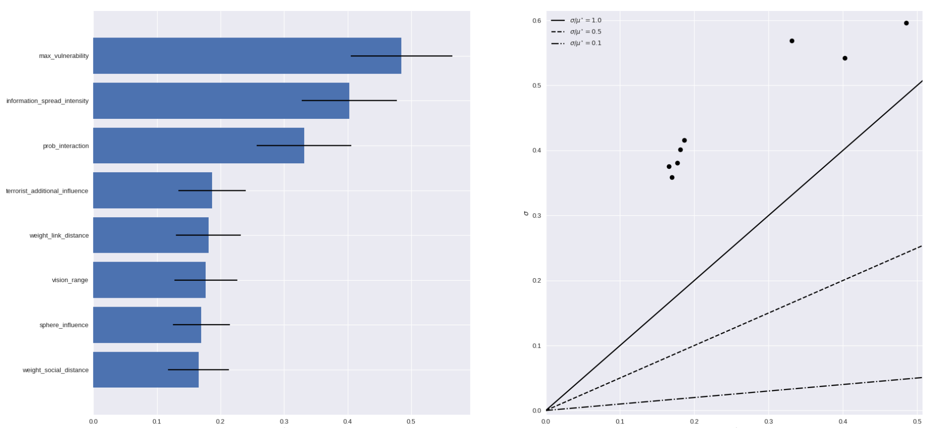
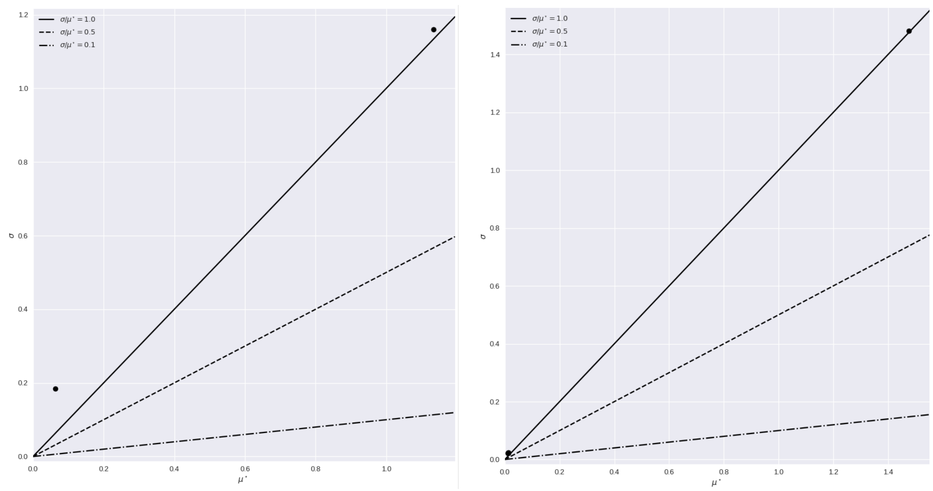



**Fig. 7.** Morris method results representation for network model and radical population output for 200 trajectories



(a) Scale Free output (b) Small World output

**Fig. 8.** Morris method results representation for radical population output for 200 trajectories

## References


1. Martha Crenshaw. The psychology of terrorism: An agenda for the 21st century. *Political psychology*, 21(2):405–420, 2000.
2. Alexander Spencer. Questioning the concept of 'new terrorism'. *Peace, Conflict and Development*, pages 1–33, 2006.
3. Oliver Gruebner, Martin Sykora, Sarah R Lowe, Ketan Shankardass, Ludovic Trinquart, Tom Jackson, SV Subramanian, and Sandro Galea. Mental health surveillance after the terrorist attacks in paris. *The Lancet*, 387(10034):2195–2196, 2016.
4. David Tucker. What is new about the new terrorism and how dangerous is it? *Terrorism and Political Violence*, 13(3):1–14, 2001.
5. Jesús M. Sánchez, Carlos A. Iglesias, and J. Fernando Sánchez-Rada. Soil: An Agent-Based Social Simulator in Python for Modelling and Simulation of Social Networks. In Bajo J. Vale Z. Demazeau Y., Davidsson P., editor, *Advances in Practical Applications of Cyber-Physical Multi-Agent Systems: The PAAMS Collection*, volume 10349 of *LNAI*, pages 234–245. PAAMS 2017, Springer Verlag, June 2017.
6. Kevin Ryczko, Adam Domurad, Nicholas Buhagiar, and Isaac Tamblyn. Hashkat: large-scale simulations of online social networks. *Social Network Analysis and Mining*, 7(1):4, 2017.
7. Aric Hagberg, Pieter Swart, and Daniel S Chult. Exploring network structure, dynamics, and function using networkx. Technical report, Los Alamos National Lab.(LANL), Los Alamos, NM (United States), 2008.
8. Gabor Csardi and Tamas Nepusz. The igraph software package for complex network research. *InterJournal, Complex Systems*, 1695(5):1–9, 2006.
9. Mathieu Bastian, Sebastien Heymann, Mathieu Jacomy, et al. Gephi: an open source software for exploring and manipulating networks. *Icwsm*, 8:361–362, 2009.

10. Diego Blanco-Moreno, Rubén Fuentes-Fernández, and Juan Pavón. Simulation of online social networks with krowdix. In *Computational Aspects of Social Networks (CASoN), 2011 International Conference on*, pages 13–18. IEEE, 2011.
11. Joshua O'Madadhain, Danyel Fisher, Padhraic Smyth, Scott White, and Yan-Biao Boey. Analysis and visualization of network data using jung. *Journal of Statistical Software*, 10(2):1–35, 2005.
12. David Gilbert. The jfreechart class library. *Developer Guide. Object Refinery*, 7, 2002.
13. Norm Matloff. Introduction to discrete-event simulation and the simpy language. *Davis, CA. Dept of Computer Science. University of California at Davis. Retrieved on August*, 2:2009, 2008.
14. Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *Journal of machine learning research*, 12(Oct):2825–2830, 2011.
15. Il-Chul Moon and Kathleen M Carley. Modeling and simulating terrorist networks in social and geospatial dimensions. *IEEE Intelligent Systems*, 22(5), 2007.
16. Mathew Penrose. *Random geometric graphs*. Oxford university press, 2003.
17. Paul Cummings and Chalinda Weerasinghe. Modeling the characteristics of radical ideological growth using an agent based model methodology. In *MODSIM World*, 2017.
18. Miller McPherson, Lynn Smith-Lovin, and James M Cook. Birds of a feather: Homophily in social networks. *Annual review of sociology*, 27(1):415–444, 2001.
19. Rositsa Dzhekova, N Stoynova, A Kojouharov, M Mancheva, D Anagnostou, and E Tsenkov. Understanding radicalisation. review of literature. *Center for the Study of Democracy, Sofia*, 2016.
20. Ari Jean-Baptiste. *Terrorist Safe Havens: Towards an Understanding of What They Accomplish for Terrorist Organizations*. PhD thesis, University of Kansas, 2010.
21. James JF Forest. Terrorist training centers around the world: A brief review. *The Making of a Terrorist: Recruitment, Training and Root Causes*, 2, 2005.
22. Paul Cummings. Modeling the characteristics of radical ideological growth using an agent basedmodel methodology. Master Thesis, George Mason University, 2017.
23. Lisa Rashotte. Social influence. *The Blackwell encyclopedia of sociology*, 2007.
24. Stephen P Borgatti. Centrality and network flow. *Social networks*, 27(1):55–71, 2005.
25. Michael Genkin and Alexander Gutfraind. How do terrorist cells self-assemble: Insights from an agent-based model of radicalization. Technical report, SSRN, July 2011.
26. Nick Qi Zhu. *Data visualization with D3. js cookbook*. Packt Publishing Ltd, 2013.
27. GEXF Working Group et al. Gexf file format, 2015.
28. Kevin Sheppard. Introduction to python for econometrics, statistics and data analysis. *Self-published, University of Oxford, version*, 2, 2012.
29. Max D Morris. Factorial sampling plans for preliminary computational experiments. *Technometrics*, 33(2):161–174, 1991.
30. Bertrand Iooss and Paul Lemaître. A review on global sensitivity analysis methods. In *Uncertainty management in simulation-optimization of complex systems*, pages 101–122. Springer, 2015.
31. Jon Herman and Will Usher. Salib: an open-source python library for sensitivity analysis. *The Journal of Open Source Software*, 2(9), 2017.