OpenSRANE, a Flexible and Extensible Platform for Quantitative Risk Assessment of NaTech Events

Bijan Sayyafzadeh[1] iD **; Abdolreza S. Moghadam[2]** iD **; Mahdi Sharifi[3*]; Eslam Kashi[4]** iD

1. Ph.D. Student, Department of Civil Engineering, Faculty of Technical and Engineering, University of Qom, Qom, Iran
2. Associate Professor, Structural Engineering Research Center, International Institute of Earthquakes Engineering and Seismology (IIEES), Tehran, Iran
3. Assistant Professor, Department of Civil Engineering, Faculty of Technical and Engineering, University of Qom, Qom, Iran
4. Assistant Professor, Department of Chemical Technologies, Iranian Research Organization for Science and Technology (IROST), Tehran, Iran
Corresponding author: *m.sharifi@qom.ac.ir*

https://doi.org/10.22115/SCCE.2023.407882.1685

ARTICLE INFO

ABSTRACT

The effects of natural hazards triggering technological disaster (NaTech) on a society, economy and the environment is a multi-disciplinary research topic. The novelty of the issue and the lack of a standard procedure for risk assessment of this category of incidents show the need for more research in this area. This article introduces OpenSRANE as an open-source, extensible, flexible and object-oriented software for calculating the quantitative risk of NaTech events in process plants. Implementing the software in the Python programming environment provides high flexibility for the modeling and evaluations desired by users. The possibility of implementing the modifications and developments to the existing software as needed by users allows them to add their desired algorithms, elements and models to it, if needed. The software is based on the Monte Carlo method, but it is possible to implement other algorithms and approaches to it. Object-oriented programming and separation of the different parts of the software can increase the readability of the program, allowing researchers in different disciplines to focus easily on studying or developing the desired part with minimal interference from other parts. The applicability of the software has been demonstrated in a case study as well as the ability of the software to calculate results such as the individual risk, scenarios that consider domino effects and physical effects.

1. Introduction

Discussions of natural hazards triggering technological disaster (NaTech) events has become a familiar and concerning topic for governments, researchers and stakeholders in response to disasters that have afflicted societies and countries in recent years. Studies have shown that these are no longer rare events and can no longer be ignored [1]. Examples of NaTech events include hurricane Katrina in the United States with a loss of about USD 330 million, the devastating earthquake and tsunami in 2011 in Japan, the 1999 Kocaeli earthquake in Turkey, the 2001 Gujarat earthquake in India, the April 2015 earthquake in Nepal and the April 2016 earthquake in Ecuador [2–6].

Lack of preparation for these types of incidents has created confusion about providing the appropriate response to existing conditions by decisionmakers, emergency response teams and others. The reason for this includes the lack of a specific approach to evaluate these kinds of event, the lack of statistics about them, the multidisciplinary nature of the topic, the lack of proper interaction between disciplines, the inadequacy of specialized information and the adoption of similar approaches that include process accidents [1,7]. On the other hand, the dependence of modern societies on industrial products and the increasing growth of NaTech events leave no doubt about the necessity of evaluating them [8,9].

Many countries have made no decisions regarding the handling the risk of a NaTech event and have divided the responsibility between ministries or government organizations. In other countries, these responsibilities have been clearly delineated, those in charge of handling them have been selected and these officials have been obliged to formalize the risks in the form of safety documents but without mentioning the method and framework [1,10,11].

Seligson et al. assessed the seismic risk of industrial plants in a large area using many simplifications [12]. Their study can be counted as one of the first studies in the field of NaTech risk assessment. Salzano et al., presented an initial quantitative seismic risk assessment study of the oil industry [13]. In it, they focused on storage tanks as major and dangerous equipment and prepared fragility curves for them in order to estimate the damage probability to the equipment and calculate seismic risk.

The quantitative risk assessment (QRA) method for process plants is presented in a collection known as Colored Books. It has an established place in safety studies of chemical and process factories for assessing the quantitative risk of common process events. It analyzes the sequence of events caused by an initial loss of containment [14,15]. Fabbrocino et al. developed the QRA method, presented an algorithm to quantify the risk of industrial plants under seismic excitation that also focused on oil storage tanks [16]. Antonioni et al. presented a comprehensive algorithm to quantitatively estimate the seismic risk of oil and gas industries under seismic excitation [17]. They considered the domino effect for all critical equipment, but the algorithm was limited to considering domino effects for secondary events and did not cover higher levels [18]. Antonioni et al. showed that the algorithm and framework could be used for all types of external natural excitation, such as floods and earthquakes [19].

The Monte Carlo method is a computerized mathematical technique that allows the use of repeating random samples of parameters to explore the behavior of a complex system. It is widely used for the simulation of physical, chemical and biological systems [20–22]. Abdolhamidzadeh et al. presented a new approach to assess the seismic risk to process plants using the Monte Carlo method to overcome the limitations of analytical methods when handling the uncertainties and complexities of domino effects [23]. Girgin and Krausmann presented a framework for rapid assessment of local and regional risk of NaTech events with minimum available data and presented an online software that can be used by the public [24]. Its user-friendly environment and expandable database are unique features of this online software (Rapid-N). Cozzani et al. evaluated the domino effect as it relates to NaTech events [14]. They reviewed existing methods for QRA of natural events and developed a framework to consider multi-level domino effects.

Antonioni et al. developed a risk assessment algorithm for NaTech events related to flooding and provided a general framework for it [25]. Necci et al. modified an existing algorithm to assess the risk of NaTech floods and earthquakes and presented a modified flowchart to calculate the risk of NaTech events caused by lightning, but ignored the domino effect [26]. Alessandri et al. modified existing algorithms and presented a comprehensive numerical approach based on the Monte Carlo method to assess the seismic risk of NaTech events [15]. This numerical method did not have the limitations of analytical methods for calculating higher orders of the domino effect. Misuri et al. modified and improved a general framework that had been developed to assess the risk of NaTech events in previous studies [27]. They presented an algorithm in which the domino effect was considered for the natural hazard of lightning. Misuri et al. modified the QRA algorithms for NaTech events from past studies and made it possible to consider the effect of multi-level barriers [28]. Their overall approach used the framework proposed by Cozzani et al. and they developed it by adding a new step for safety barrier performance assessment [14,29].

Misuri et al. proposed an innovative framework that extended existing approaches to consider the role of utilities, auxiliary systems and safety barriers [30]. Their framework consists of a direct accident path and an indirect accident path. The direct path is based on existing approaches [14,19,25] and the indirect path is their suggestion for considering possible scenarios caused by the failure of utilities and auxiliary systems. They proved that different categories of hazardous substances may have critical properties that can lead to the development of indirect Natech accidents.

Tugnoli et al. developed the methodology proposed by Cozzani et al. [31] to systematically consider the probability of escalation due to multiple fragment projection in quantitative risk assessment [32]. They showed how consideration of domino scenarios caused by fragment projection can play a significant role in the correct assessment and management of risk.

Sorichetti et al. developed the framework presented by Antonioni et al. [25] for multi-risk assessment of NaTech events [33]. Their method enables the evaluation of NaTech risks from different reference natural events. The presented methodology produces useful indications for which natural hazards should be prioritized in NaTech risk mitigation strategies.

Huang et al. proposed a method to consider multi-hazard scenarios triggered by natural events [34]. In modelling the primary event and the possible domino accidents, this method considers exposure to hazards, the sensitivity of causes and the resilience of assets. Wang et al. suggested a simplified methodology for rapid NaTech risk assessment of multiple hazards under a multi-hazard scenario [35]. They focused on analysis of damage process of equipment, but they did not consider the domino effects.

In a literature review of QRA of domino accidents, Xu et al. emphasized that the framework presented by Cozzani et al. [31] was a complete framework for QRA assessment [36]. Mesa-Gómez et al. reviewed existing methods of QRA for NaTech events and acknowledged that there always have been efforts to improve them [37]. They confirmed the need for more efforts to develop and refine existing methods. Caputo et al. comprehensively reviewed QRA methods of NaTech events and considered the need to develop the existing methods [38].

The previous review shows that the NaTech risk assessment methods have always been under development. The existence of a comprehensive platform can help increase knowledge in this area as quickly as possible and solve many of the problems mentioned in previous studies. Such a platform must have the following characteristics:

- It should be capable of being compatible with different types of studies from other disciplines. As its multidisciplinary nature is one such problem, the platform should be able to create a connection between study results from different disciplines.
- It should have the flexibility to model and combine its components to be used to expand its capacities and implement different algorithms.
- It should easily provide the possibility of accessing validated algorithms used by other researchers for future development.
- It should have the ability to expand, change, modify and upgrade existing algorithms and accept new elements and components.

McKenna (1997) provided finite element method (FEM) object-oriented software called OpenSees with the aim of increasing the software flexibility and extensibility for users to apply custom settings. Although OpenSees was initially prepared for seismic assessment of structures, it was later developed by other researchers to simulate the purposes such as the effect of fire, soil structure interaction and fluid structure interaction. With its development for the Python programming language by Zhu et al. [39], users now can benefit from the numerous libraries available in Python. The flexibility of OpenSees has allowed researchers to provide different FEM models with the desired settings and algorithms for evaluation in their fields. Other researchers have developed a variety of algorithms, behavioral models and elements using OpenSees [40–46]. The objectives outlined in this platform and software are suitable for solving the problems mentioned in previous studies in the field of risk assessment of NaTech events.

The current article introduces the Open System for Risk Assessment of NaTech Events (OpenSRANE) platform in order to perform calculations related to risk assessment of NaTech events. It is based on the Monte Carlo method with the goal of providing high flexibility in simulating NaTech events when performing risk assessments. It also has a simple development

capability to encourage users, researchers and specialists in different fields to act on a single platform with its ease of access to existing codes and algorithms. Python programming language is used in this program. Modeling in the Python programming environment, in addition to providing various modeling capabilities and access to many databases, provides a huge bank of libraries available to users. The object-oriented and open-source nature of the platform allow users to develop it and investigate the various elements and calculation methods and algorithms. The platform algorithm generates different scenarios that automatically considers the domino effect is presented along with a description of the platform structure. Similar approaches have been provided for buildings and infrastructure systems [47].

Section 2 is a review of existing tools and algorithms and discusses the lack of a comprehensive software for researchers. A comprehensive algorithm is selected to be implemented inside the presented software. Section 3 discusses the components required for NaTech risk analysis and their roles in the algorithm selected for the presented software. Section 4 describes the architecture of the software and the relationships between components. In this section, developers can become familiar with the software structure and how to develop it according their requirements in the future. The steps of modeling in the software are introduced in this section. Section 5 addresses the software documentation and source code and describes the existing materials in the software. The modeling steps introduced in the final part of section 5 are described in more detail in section 6. Section 7 demonstrates the performance of the software using a case study. Finally, section 8 discusses the results, features and abilities of the software.

2. Tools and algorithms used for NaTech quantitative risk assessment

Fabbrocino et al. combined structural seismic risk with the QRA method in the continuation of their studies [16]. He did not mention any special software to implement his numerical calculations and only mentioned PHAST [48] software to calculate the burning rate and fire shape of pool fires. This is not free software nor open-source; thus, it is not possible to access its code and algorithm for making the changes required by researchers.

Antonioni et al. proposed a framework for seismic risk assessment that could be developed for any other natural event [17,19]. They developed the ARIPAR-GIS software in which to implement the proposed algorithm [49,50]. This is a software for individual and social risk assessment under constant resources. The package specifies all possible initial scenarios and performs individual and social risk assessment in ARIPAR-GIS. However, users cannot make changes in the algorithm steps following the initial scenario and do not encounter an integrated package. Because of the closed analytical approach of the package, analysis of the domino effect is not available at all levels.

Girgin and Krausmann developed Rapid-N, a free online free software (WebApp), with a suitable database of hazard and vulnerability data that can interact with geographic maps to draw a plant and its components [24]. Providing approximate relationships to estimate non-existent information, the possible use of relationships by the user and of implementing local and regional risk assessments are among its strong points. However, the program is not open-source and the ability to develop or change the algorithm is not provided for users.

Caputo et al. presented an analytical-numerical systematic structure for seismic risk assessment that presents the dynamics of events as "damage levels" to add accident propagation to the calculations [51]. They implemented their proposed algorithm in MATLAB software, which is relatively expensive. It is thus not available for all researchers and the code is not open-source.

Following the method proposed by Caputo, Alessandri et al. presented a systematic algorithm based on the Monte Carlo method which is able to consider all possible scenarios by considering the domino effect, does not have the limitations of analytical methods and presents a comprehensive framework for seismic risk assessment [15]. They implemented the algorithm in MATLAB under the name of PRAIMUS. The lack of a source code and the use of MATLAB means it is not available for everyone.

Other research has primarily used a combination of software types and hand calculation [25–28]. In one case, the lack of suitable software with which to implement the algorithms forced simplifications to allow hand calculations and the combination of some software platforms [14]. Table 1 lists some of the software used in the reviewed articles with their accessibility and the possibility of development. It is evident that a free and open-source software with good flexibility and extensibility and the ability to access different libraries that offers a simple programming language can help researchers in the rapid development of this research field and cover such limitations.

Table 1
Software used for quantitative NaTech risk assessment and their accessibility.

First author (year)	Approach	Domino-effect	Software	External software	Free	Extensible
Fabbrocino (2005)	analytical	no	-	PHAST	no	-
Antonioni (2007)	analytical	yes	-	ARIPAR-GIS	yes	no
Girgin (2013)	analytical	no	Rapid-N	-	yes	not in main algorithm
Caputo (2017)	analytical	yes	MATLAB Code	-	no	no
Alessandri (2018)	Monte Carlo	yes	PRIAMUS	-	no	no

Mesa-Gómez et al. showed that the basis and structure of algorithms used by researchers over 13 years of research have followed the same steps (collecting primary information, evaluating primary scenarios and calculating risk) [37]. They illustrated this issue by presenting flowcharts for two articles. The addition of the domino effect was the only difference between them. Because of the importance of this effect on the results, it should be considered in other research and studies.

The algorithm presented by Alessandri et al. is comprehensive without having the limitations of analytical methods and can be developed to consider all types of events by considering all types of uncertainty [15]. This allows the possibility of development for other unseen problems, shows the comprehensiveness of the algorithm and its possible future development [52]. The use of the

Monte Carlo method in this algorithm allows all possible domino-effect chains to be clearly seen. All possible damage scenarios and accident propagations can be investigated without concern about the increase in the number of possible scenario chains, which shows its high flexibility. These features meet the goals drawn for preparation of the software. Therefore, because of the generality of this algorithm, it was considered as the base algorithm of the software. It offers comprehensiveness and high flexibility although, in some details, different approaches may have been adopted.

3. Components of risk analysis

It can be said that quantitative risk assessment of a plant experiencing a natural event has four basic steps: collecting primary data, evaluating primary NaTech scenarios, evaluating possible domino effects and risk calculation. When collecting the primary data, information related to the equipment and elements, statistics about natural hazards and environmental information are required. The natural hazard statistics should provide the information required to determine the hazard intensities and their occurrence probability in order to estimate damage to equipment and elements. Also, information about the equipment and elements should provide their locations, structural characteristics, vulnerability under loads such as natural hazards and possible explosive and thermal loads caused by fire to estimate the equipment vulnerability. Environmental information such as wind, temperature and pressure statistics in the region under investigation and day/night duration ratios are also required to allow the QRA algorithms to calculate the outflow rate, material dispersion and physical effects of possible physical events. Up to now, such data has been based on the requirements of previous studies. It is likely that, in the future, different elements and data will be required. Moreover, different types of data may be presented and may not necessarily be in one format or type.

The evaluation of initial NaTech scenarios and the domino effects form the analysis portion of the study. Risk calculation based on the resulting information forms the post-process portion. The assessment of domino effects and calculations related to the outflow rate and dispersion of hazardous materials and the physical effects caused by physical events are based on the models defined for all material states [53,54].

The different data collections and models needed for risk analysis can be categorized as shown in Fig. 1. For each area in Fig. 1, different types of data or models may be available; therefore, the comprehensive software should be flexible and able to receive all types of data and models. In the analysis portion of risk analysis, the evaluation of initial scenarios and domino effects can be carried out by using the required models and data and providing a proper connection between them. In the last step, risk calculations can be done according to the obtained results. Fig. 2 shows the analysis algorithm of one NaTech event scenario considered in the current study.
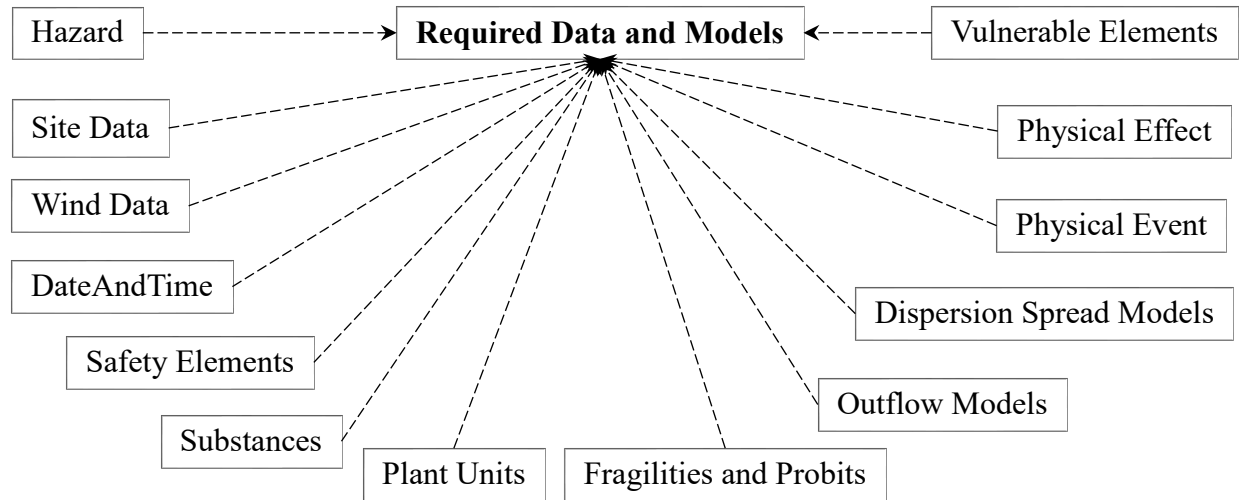
Fig. 1. Required NaTech risk assessment data headers.

As stated, the algorithm has been derived from Alessandri et al. and the damage levels and other parameters are taken from it [15]. In the first step, probabilistic data are sampled according to their defined distributions. Data such as whether it is day or night and the amount, direction and class of wind are included in this section. In the next step (zero-level analysis), the intensity of the natural hazard is sampled, vulnerability of the equipment under the sampled intensity is investigated and damage to equipment is determined. Finally, the initial or primary scenario is formed in this step.

In the next step (fragility outflow analysis), for each one of damaged units that experiences material outflow, the outflow rate is determined by the defined models under structural failure caused by the effect of natural hazards. The material outflow rate can be used to calculate material dispersion in the environment in the dispersion analysis stage. Next, according to the probability defined for the types of possible physical events (using the event tree), the physical events and their effect intensities are determined in the physical event analysis stage. Finally, in the physical effect analysis stage, the defined probit functions for each one of plant units are used to determine damage to the equipment.

In Fig. 2, calculation of the material outflow models that have been selected because of damage determined by the probit functions is denoted as "probit outflow analysis". The calculation of material outflow models that have been selected because of damage determined by fragility functions is denoted as "fragility outflow analysis".

If at least one unit is damaged by physical effects which result in outflow of hazardous material, after calculating the outflow rate, the steps identified after the dispersion analysis stage should be repeated so that no further damage will lead to outflow of material. This leads to consideration of the domino effect in the model up to any level. Completion of the domino effect calculations will help determine the vulnerability of the specific areas based on the probit functions defined in the vulnerable-elements analysis stage. Vulnerable-elements include elements such as the

population, greenspace or any other element that can be damaged by physical effects. The probability of the occurrence of any type of event E is equal to:

$$P[E] = \frac{\sum_{i=1}^{N} \delta_i(E)}{N} \tag{1}$$

where $\delta_i(E)$ is equal to 1 if event E has occurred in the ith simulation (otherwise is equal to 0) and N is the total number of simulations. Any event can be considered as event E, including a specific scenario, cost amount, number of damage levels, etc.
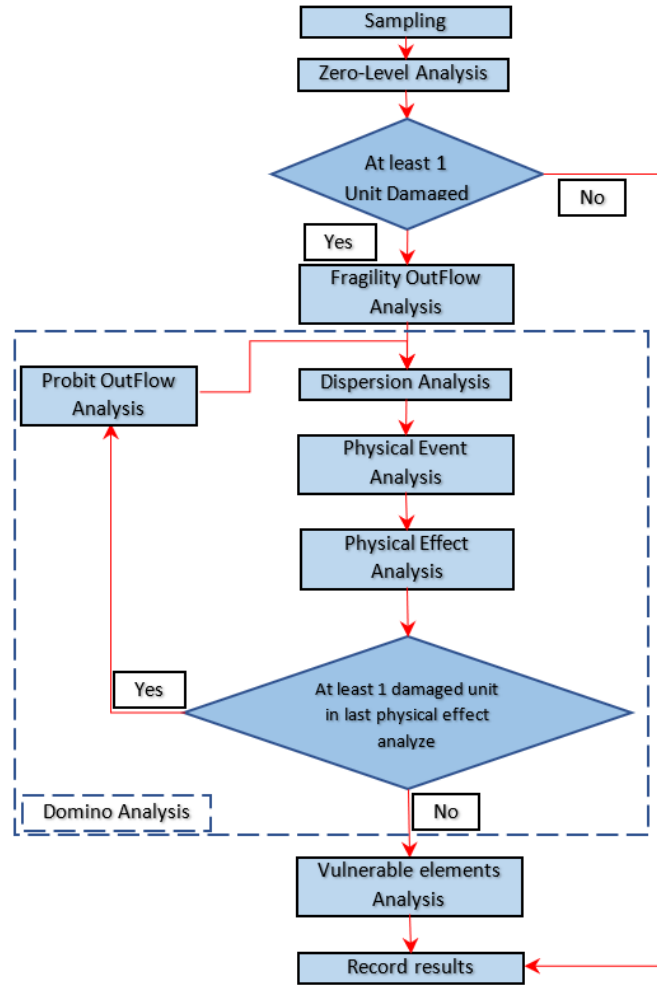

Fig. 2. Analysis algorithm for a NaTech scenario.

If the conditional probability of event E needed to be considered, it can be calculated by using the simulations made with the desired conditional value. For example, the probability of event E under the condition of a natural hazard occurrence having intensity $IM = IM_0$ is:

$$P[E|IM_0] = \frac{\sum_{i=1}^{N} \delta_i(E, IM_0)}{N(IM_0)} \tag{2}$$

where $N(IM_0)$ is the number of simulations with an intensity equal to IM_0 and $\delta_i(E, IM_0)$ is equal to 1 if the event E occurs at an intensity equal to IM_0 (otherwise, it will be equal to 0).

It has been observed that quantitative risk analysis consists of many components and models, each of which can have many variations. The models required to simulate each of the stages can be of various types with different assumptions. The possible variations in the results can be envisaged. Qualified software should be able to cover all of these changes and variations so that users can aim for different goals and undertake evaluations and changes.

4. Architecture of the software

OpenSRANE software was designed in Python programming language to achieve the goals mentioned in this article. It has an object-oriented architecture inspired by OpenSees, which was produced to simulate and solve equations using the finite element method and is based on the production and combination of objects. For quantitative risk analysis using OpenSRANE, Python scripting language has been extended by adding additional commands; thus, users and developers of the software should be familiar with this programming language. The software consists of a set of classes that can be combined with each other and with new classes developed by other users. The framework of software has been designed such that the classes for each part have been placed in separate subpackages to make them easier to understand and more readable. Fig. 3 shows the basic subpackages of the program along with their relationships to each other.
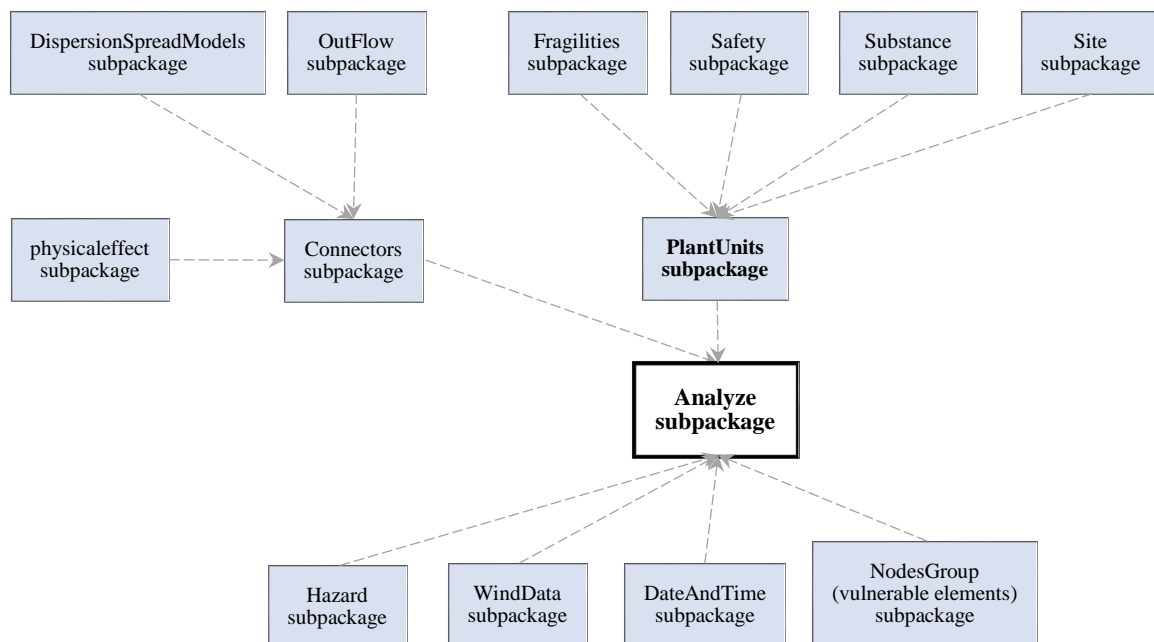
Fig. 3. OpenSRANE subpackages and their relation.

Regarding to the extensibility of the program, the subpackages can be changed in the future. The classes in each subpackage produce similar objects with different behaviors, characteristics or input parameters. Some subpackages have purposes other than producing objects, such as the Analyze subpackage, which uses defined objects to implement the algorithm shown in Fig. 2 to follow the steps of NaTech risk analysis. Other subpackages, such as Misc, Plot and Recorders are used to store the desired outputs, draw the model and provide the commands required for

modeling purposes. These subpackages have not been shown in Fig. 3 because they do not play a key role in the main calculation process of the program and to avoid clutter.

The classes in the Fragilities, Safety, Substance and Site subpackages define the primary data required for allocation to the objects made from the PlantUnits subpackage classes. The classes available in the OutFlow, DispersionSpreadModels and PhysicalEffect subpackages are used to define models of the material outflow and dispersion and occurrence of physical events. The classes in the Connectors subpackage connect the models to the relevant equipment according to the structural fragility model, equipment content and Outflow model.

By using the classes in Hazard, WindData, DateAndTime and NodesGroup subpackages, other required environmental information can be obtained from the user. All the information received from the user is used to produce various objects from the relevant classes. Finally, the Analyze subpackage can be used to perform the analysis steps (Fig. 2) by employing and combining the defined objects. The analysis section has access to all defined objects and, according to need, each object can either be elicited or set.

The design of the classes which produce the objects in the subpackages is such that new objects with different characteristics and behaviors can be easily added to the software. They then can be used in the body of the software. This architecture of the software plays a fundamental role in its extensibility. In this architecture, all objects are produced with a single identifier (Tag) from a mother class (NewClass) which is inherited by other classes in the subpackages and are managed and referred by the ObjManager class. It is possible to refer to an object in any existing or new class that is compatible with the software architecture by assigning it a separate and unique identifier Tag. The software will automatically recognize these objects and add them to the set of objects to be used in the software.

The combination of objects leads to high flexibility and extensibility compared to the inheritance mode. From the software engineering perspective, this allows researchers and developers to focus aspects specific to their interests with minimal attention to other aspects of the software [40]. All classes in a subpackage are required to provide a minimal set of results within a specific set of predefined parameters. For this purpose, a mother class (GlobalParameters) is used for each subpackage, in which the output parameters, structure and methods included in the classes of a subpackage are determined. By inheriting the classes from the mother class, the subpackage classes also will include these parameters and the algorithms written in that class should set values for the parameters. Therefore, all the existing classes in a subpackage and the objects made from them will ultimately have specific outputs within specific parameters that will be the same as the response of the subpackage.

The explanations have been provided here for developers and more details have been provided in the developer documentation. Individuals who use the software do not need to focus on the concepts and contents of this section (sec. 4). It is sufficient for them to read the user guide, although this knowledge will lead to a deeper understanding of how the software works. The main steps of implementing a model are shown in Fig. 4 and changes in these steps are likely in the future, considering the software extensibility.
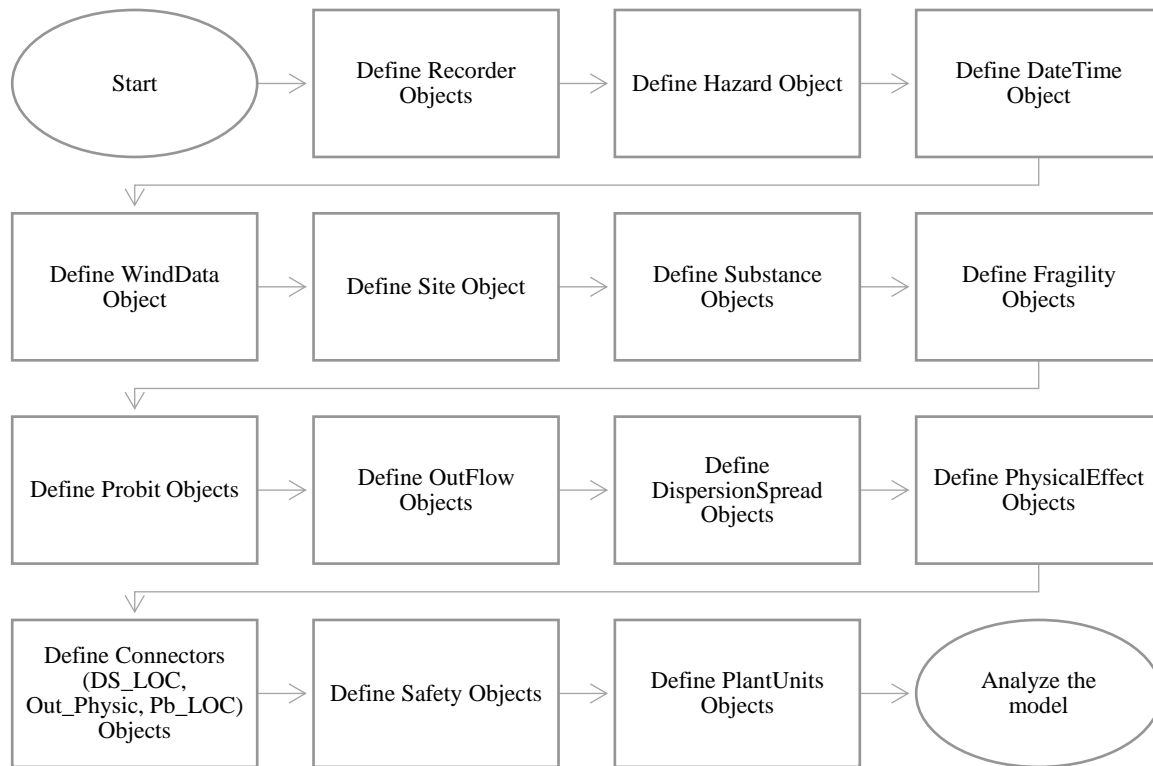
Fig. 4. OpenSRANE analysis steps.

Considering that the coding of these steps must be done in the Python environment, users can use the features of the Python programming language in addition to program modeling commands to meet their purposes. The advantages of Python include the use of the Python libraries, including 2D and 3D drawing libraries, use of iteration loops for modeling and working with data, use of numerical libraries such as Numpy, Scipy and the statistics library Pandas, access and managing databases such as the weather and seismic databases, information on watercourses and geodata. Section 5 provides more details about the subpackages in Fig. 3 and the path shown in Fig. 4.

5. Access to OpenSRANE

It has been a challenge for researchers to access the results of articles and research without high expenditures and obstacles to speeding up the development of research fields. If the software and codes that are developed during scientific research are accessible to the public as open-source software, such free access to a software source will increase their chance of more effective and faster development. Since 2018, the number of articles connected to open-source resources has increased from 50% to 75%, which can increase the effectiveness of open-source articles [55].

GitHub website (https://github.com/) is a platform for sharing open-source software where more than 20,000 types of software related to academic studies have been published between 2014 and 2018. Management of changes requested to codes in order to improve them, free public access to the program codes, observation and management of program changes made in different versions over time and development of parts of the program by different researchers are among the

advantages and capabilities of GitHub and similar platforms. Studies indicate that this platform currently is more popular than other platforms and the increase in the use of these platforms by academic communities is evidence of their success [56].

GitHub was chosen as the platform for presenting the prepared software source. The software can be directly installed by users as a library called OpenSRANE for the Python programming environment. By visiting https://opensrane.github.io/OpenSRANE_Documentation, users and developers can find the software documentation and source code address to use or participate in the development of the program.

Documents on and examples of how to use the software and verifications for all types of existing modules in the software are located at the address provided. In the documentation part, the command guides for users are placed under UsersGuideline, where users can learn about the program commands and how to use them as well as the modeling steps. The Developers Guideline (under DevelopersGuideline) describes the public and private details of subpackages aside from those found under UsersGuideline as well as how to develop them. The verifications for the modules are available under Verifications at this location. The documents are all editable files to which developers can add content or edit.

6. Functionality of OpenSRANE details and components

Correct understanding of the purposes of the subpackages will lead to more accurate, appropriate and effective modeling and application of the software. The following subsections explain the details of the modeling path (Fig. 4) and the available subpackages and their general input values, purposes and applications.

6.1. Defining recorders

In the first step, considering the need to save the results and in order to manage the system memory, it is necessary to define the storage object or objects from the Recorder subpackage. This way, the results or data specified by user will be recorded for use in post-processing operations.

6.2. Defining environmental data

In the next step, information about the day/night duration ratio, probability distribution of wind speed, direction and class in the region and about the intensity and probability of natural hazards and site data should be defined for the software by the Hazard, WindData, DateAndTime and site subpackages. As wind information is usually different in the day and at night, the software uses the day/night ratio and random sampling to determine the time of the ith simulation. Then, the relevant wind data are called for sampling for use in other parts of the software, such as material dispersion models in the environment. According to the defined intensity and probability data for natural hazards in the Hazard subpackage, each simulation uses random sampling of the defined data to select a hazard intensity for the ith simulation. These subpackages should be defined only once; defining them more than once is not currently applicable. The software will obtain the

required information about the environment pressure and temperature to use in calculations on the release and dispersion of materials into the environment from the data defined in the generated object of the site subpackage.

6.3. Defining substances

In this step, the types and characteristics of materials used in the equipment should be defined using the Substance subpackages. Because a wide range of materials could be used, any number of materials can be defined for the software using the Substance subpackage. During analysis, the required material information for all models and calculations are extracted from the objects defined in this stage.

6.4. Defining fragilities

Using Fragilities subpackages, types of fragility and vulnerability curves (probit) should be defined for the software. Data of the equipment fragility and vulnerability and vulnerable elements vulnerability can be defined as many times as needed using the Fragilities subpackage. The software can calculate the probability of equipment or elements vulnerability under any amount of external excitation intensity (natural hazard, overpressure caused by explosion, thermal radiation) using the defined data. Users also can define multiple fragilities for one unit such that each represents a possible failure mode.

6.5. Defining outflow, dispersion and physical effect models

Before defining the available equipment data at the site, it is necessary that users using OutFlow, DispersionSpreadModels and PhysicalEffect subpackages define models for the material outflows from the equipment, models for material dispersion into the environment and models for various physical events. The input parameters will vary depending on the model used.

6.6. Defining connectors

Connecting or assigning the defined models to the equipment can be done indirectly by using the classes in the Connectors subpackage. These classes make it unnecessary to directly assign a defined model to each one of plant units (which makes modeling easier, especially for huge models). It also allows the user to assign more than one model having unequal occurrence probabilities to a damage state or a material outflow state and provides more flexibility for the user when applying different types of models. The DS_LOC (damage state-loss of containment) class from the Connectors subpackage assigns outflow model types with different occurrence probabilities to a damage state. By governing the damage state, one of the outflow models connected to that damage state (by DS_LOC class) can be selected as the material outflow model (according to the defined probability distribution). In this way, it will be possible to consider various types of possible material outflow models for each damage state in the calculations.

Similarly, the Out_Physic (outflow-physical event) class assigns physical event models with defined occurrence probabilities to a defined material outflow model. Thus, for a specific

material outflow model, one of the assigned physical event models will be selected according to their occurrence probability.

The Pb_LOC (probit-loss of containment) class works similarly to the DS_LOC class by assigning the material outflow models to the probit functions. When governing equipment damage caused by the relevant probit function, the material outflow models assigned to the governed probit function will be considered and selected as the equipment material outflow model. As for the DS_LOC class, several material outflow models can be assigned to probit functions with arbitrary occurrence probabilities.

6.7. Defining safety elements and equipment

Ultimately, before the onset of the analysis operation, after defining the safety elements, such as dikes (using the Safety subpackage), all equipment that is available onsite must be defined using the PlantUnits subpackage. To define each one of plant units, its location and site, content, volume of content, fragility functions, equipment internal pressure and temperature, probit functions for vulnerability to overpressure and thermal loads, among others, should be defined. Private characteristics such as the geometric characteristics of the equipment, should be defined using the specific class parameters depending on the type of equipment. In order to be able to refer to an object defined by users, the software will request a unique tag number from the user during modeling. The Tag number defined for each object of each sub-package should be unique. The user can use these tag numbers to refer to previously defined objects in different stages of modeling.

6.8. Analysis

After defining the equipment available onsite, the modeling process will be complete and analysis of the model using the Analyze subpackage can begin and be repeated with the simulation numbers requested by the user. In each simulation, the steps shown in Fig. 2 are implemented and results are saved according to the type of Recorder object/s defined. Completion of the analysis process will allow users to perform statistical operations on the results. This is called the post-processing stage.

Each of the mentioned subpackages, in addition to receiving input data, may have internal functions for various purposes. These include performing the required calculations based on the input data and saving the results having descriptions that are not necessary to readers and would make the article voluminous and which are available on the website.

7. Tank farm analysis using the software

In order to check performance of the developed software, a tank farm comprising eight tanks containing gasoline was investigated as a case study. The plot plan of the site and the software output are shown in Fig. 5. The characteristics of the tanks are shown in Table 2. The content level is considered to be the highest possible. All tanks are unanchored and surrounded by dikes.

Table 2
Characteristics of storage tanks.

| Tanks	Tag	Diameter (m)	Height (m)	Liquid level (m)
TK-100 (D~A)	5,6,7,8	22.8	13.5	12
TK-200 (C~A)	2,3,4	24.4	13.5	12
TK-300	1	20.0	13.5	12
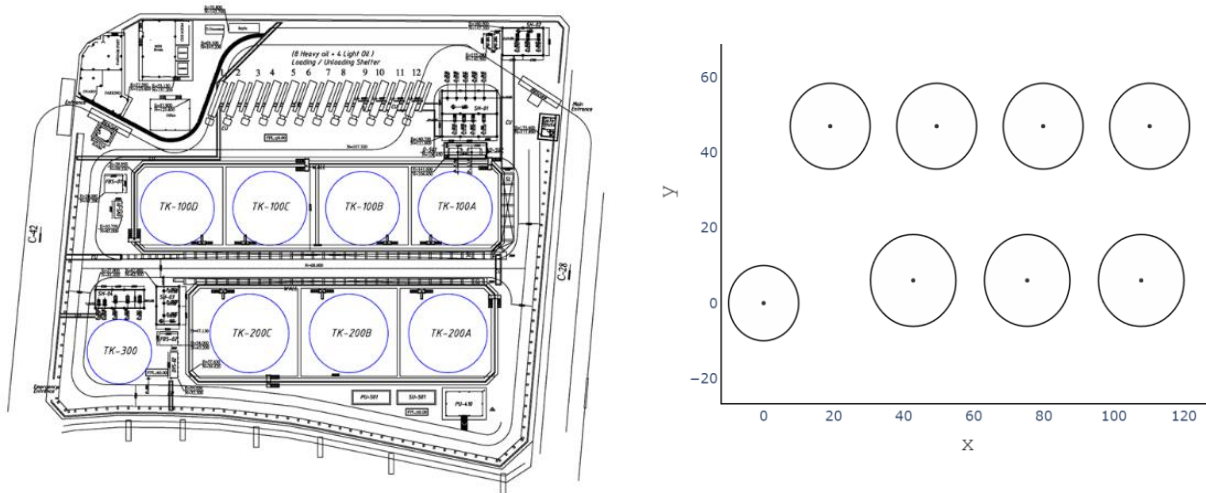
Fig. 5. Site plan (left) and OpenSRANE plot (right).

The seismic hazard curve of the site was extracted from a study conducted by Boostan et al. [57] It was fitted to the present curve to obtain the annual regional hazard curve. Fig. 6 shows the seismic hazard and fitting curves.

$$y = 0.0004x^{-1.82}$$
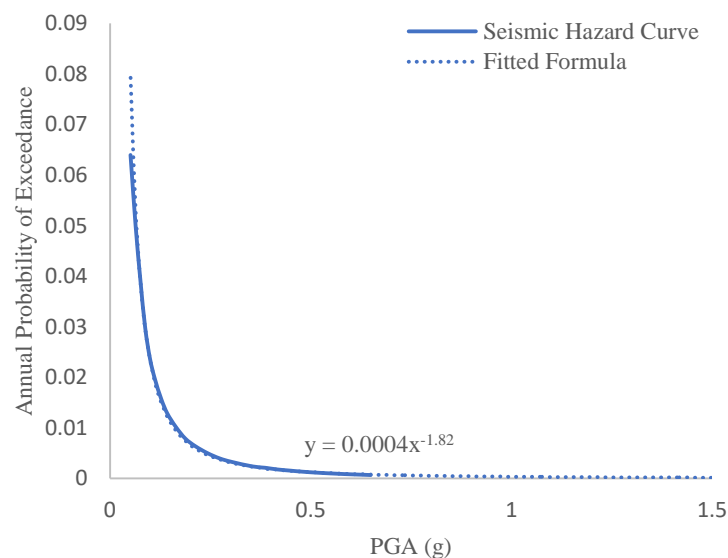
Fig. 6. Seismic hazard and fitted curve [57].

In order to determine the tank fragility curves, results from a study conducted by Razzaghi et. al [58] were used for the two damage modes which lead to outflow of the tank contents (severe damage (D4) and collapse (D5)). Considering H/D = 0.6, the relevant fragility curves were selected for all tanks. Fig. 7 shows the output of the software in the form of fragility curves. Users can define several fragility curves having different material outflow states with the desired occurrence probabilities. The flexibility of the program makes it possible for users to define different fragility states based on different content levels.
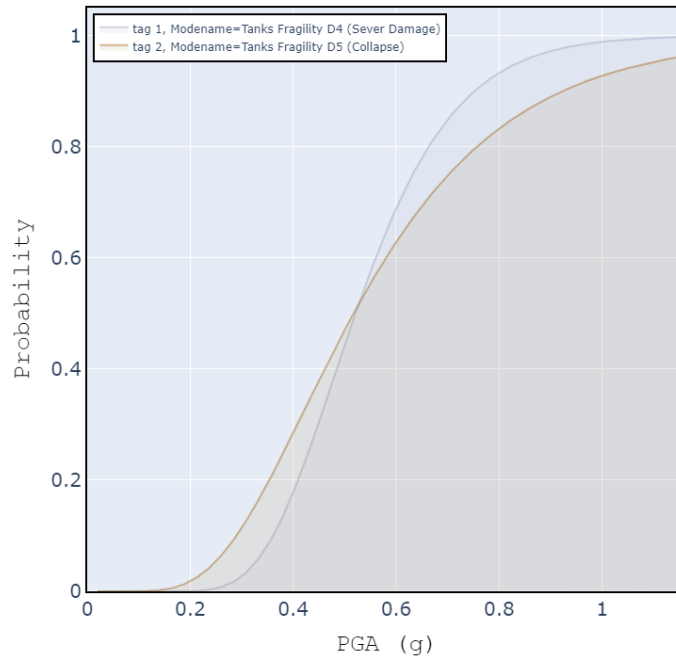
Fig. 7. Tank fragility curve (OpenSRANE output).

To calculate the vulnerability of tanks for domino-effect analysis and to calculate the mortality risk, it is necessary to define the parameters required for the probit functions. For this purpose, upper and lower limits defined for the vulnerability of equipment and the mortality of people that are available in different sources [15,54] were extracted as described in Table 3. In the upper limit, the vulnerability was considered as equal to 100%. In the lower limit, the vulnerability was considered as equal to 0% and the probit curve was calculated assuming a normal distribution in the mentioned range.

Table 3. Upper and lower limits of vulnerability under physical effects.

	Heat radiation lower limit $\left(\frac{kw}{m^2}\right)$	Heat radiation upper limit $\left(\frac{kw}{m^2}\right)$	Over pressure lower limit (bar)	Over pressure upper limit (bar)
plant units	12.5	37.5	0.3	0.6
human	4.0	12.5	0.1	0.3

The region wind information was defined in the form of the wind rose shown in Fig. 8. A wind rose is a graphic tool used by meteorologists to demonstrate how the wind speed and direction

are typically distributed at a particular location. Using a polar coordinate system of gridding, the frequency of winds over a time period is plotted according to the wind direction, with the colored bands denoting the wind speed ranges. The direction of the longest spoke shows the wind direction with the greatest frequency. The data were considered to be the same for day and night. The wind classes considered for each speed range shown in the image are equal to F, D, B, E, D, D in ascending order.

Because of the high number of analyses required, the analytical models described by Casal were used to calculate the material outflows and dispersion into the environment, although users can add another desired model to the software because of its extensibility [54]. Similarly, fire point source and TNT models were used to calculate the thermal radiation and overpressure caused by vapor-cloud explosions (VCE). Again, users can add another model to the software for this purpose, if desired.
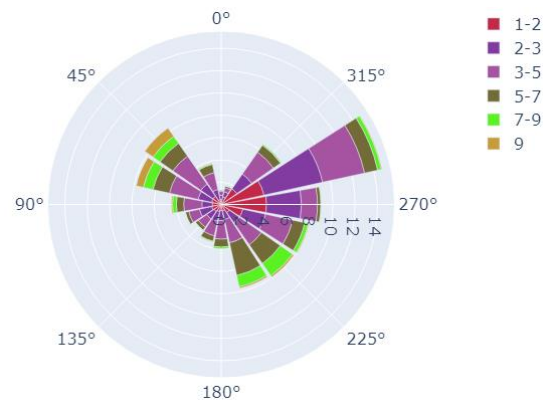
Fig. 8. Wind-rose data (OpenSRANE output) in m/s.

An event tree is used to determine the probability of the occurrence of physical events caused by dispersion of substances in the environment. There are many studies that present various types of event trees for different substances [59,60]. The event tree shown in Fig. 9 was used to determine the probability of physical events occurring as result of dispersion of the damaged tank contents. Here, for the tank contents being dispersed into the environment, the probability of VCE, pool fire and safe dispersion occurrence are equal to 35.7%, 55.48% and 8.82%, respectively.
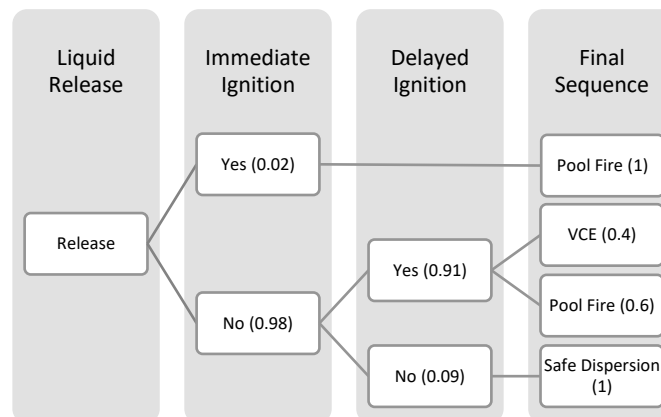

Fig. 9. Event tree considered for release of hazardous materials [15,60].

Other details related to the modeling can be seen from the examples on the GitHub page. After completing the model, analysis of the investigated plant was done by performing one million (1M) simulations. The number of simulations could be greater or less depending on the type and level of information and accuracy required. As the software is equipped with the ability for parallel analysis, the time required for analysis could be very short depending on the number and properties of system processing cores. The analysis of the present model was done on a personal computer with 8×3.89 GHz cores (AMD Ryzen 7 3800X) and 16.0 GB of RAM in less than 60 minutes.

Various results can be obtained in the post-processing stage and users can extract them according to need by performing statistical analysis on the resulting data. The probability of the different generated scenarios, occurrence probability of different damage chains and results at different damage levels as well as individual and social risk curves are some of the results that can be obtained in the post-process stage using the software analysis results. Extraction of the probability of vulnerability occurrences at the zero level according to the equipment fragility curves and also the regional hazard curve can provide information about equipment vulnerability at this level in the study region. Fig. 10 shows the occurrence probability of all generated scenarios at all damage levels based on the ID number of the generated scenarios. Fig. 11 shows the same diagram for the zero-damage level.

Users can process and extract the desired results from the data in the post-processing stage. Table 4 shows the ten highest occurrence-probability scenarios that have been extracted from results. The naming rule that has been considered in second column of Table 4 makes the understanding of scenarios easier for users. According to this rule, each damage level number in parentheses is followed by a colon and the corresponding damaged equipment tags are denoted in brackets. Data specific to each level are separated by a dashed line. For example (0):[8]-(1):[7] shows that, in damage level 0, the tank having tag 8 has been damaged and, in the next damage level (damage level 1), the tank having tag 7 has been damaged.
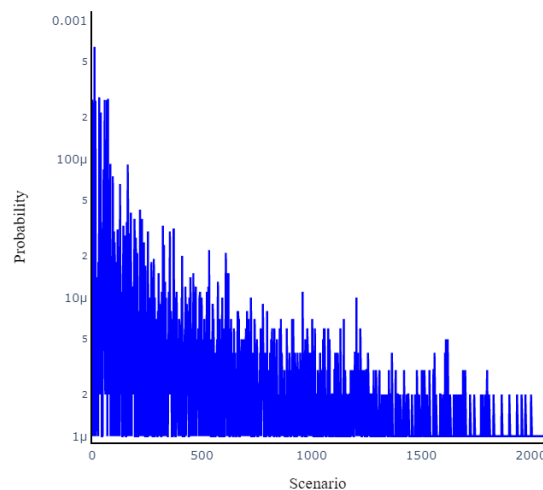
**Fig. 10.** Probability of scenario (by ID).
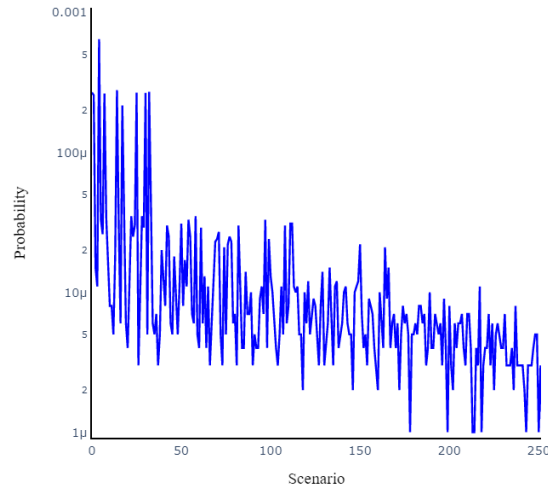
Fig. 11. Probability of scenario at zero-damage level.

Table 4. Top ten most probable damage scenarios.

Row	Scenario Name (Damage level): [plant unit tag]	Probability
1	None	0.995
2	(0):[1,2,3,4,5,6,7,8]	0.000644
3	(0):[3]	0.000278
4	(0):[7]	0.000271
5	(0):[5]	0.000268
6	(0):[1]	0.000267
7	(0):[4]	0.000266
8	(0):[6]	0.000263
9	(0):[8]	0.000256
10	(0):[2]	0.000216
11	(0):[8]-(1):[7]	0.000128

Because all plant units have the same fragility, the damage scenario probability of each unit at zero level should be the same. This expectation has been estimated according to the results obtained in rows 3 to 10 of Table . The high vulnerability of the equipment in the plant is evident and the scenario of all equipment damage is highly probable according to their fragility curves and hazard defined for the region. The results have been calculated independently (manually) for the zero level (vulnerability under the hazard curve) and the accuracy of the results was controlled. Obviously, plants with a variety of equipment and different fragilities will present different results.

Many chains of possible scenarios have been produced because of considering domino effects. For each scenario, the effect of the physical events on other equipment and vulnerable elements is assessable. Fig. 12 shows three rarely produced scenarios along with the sequence of equipment damage (damage chain).
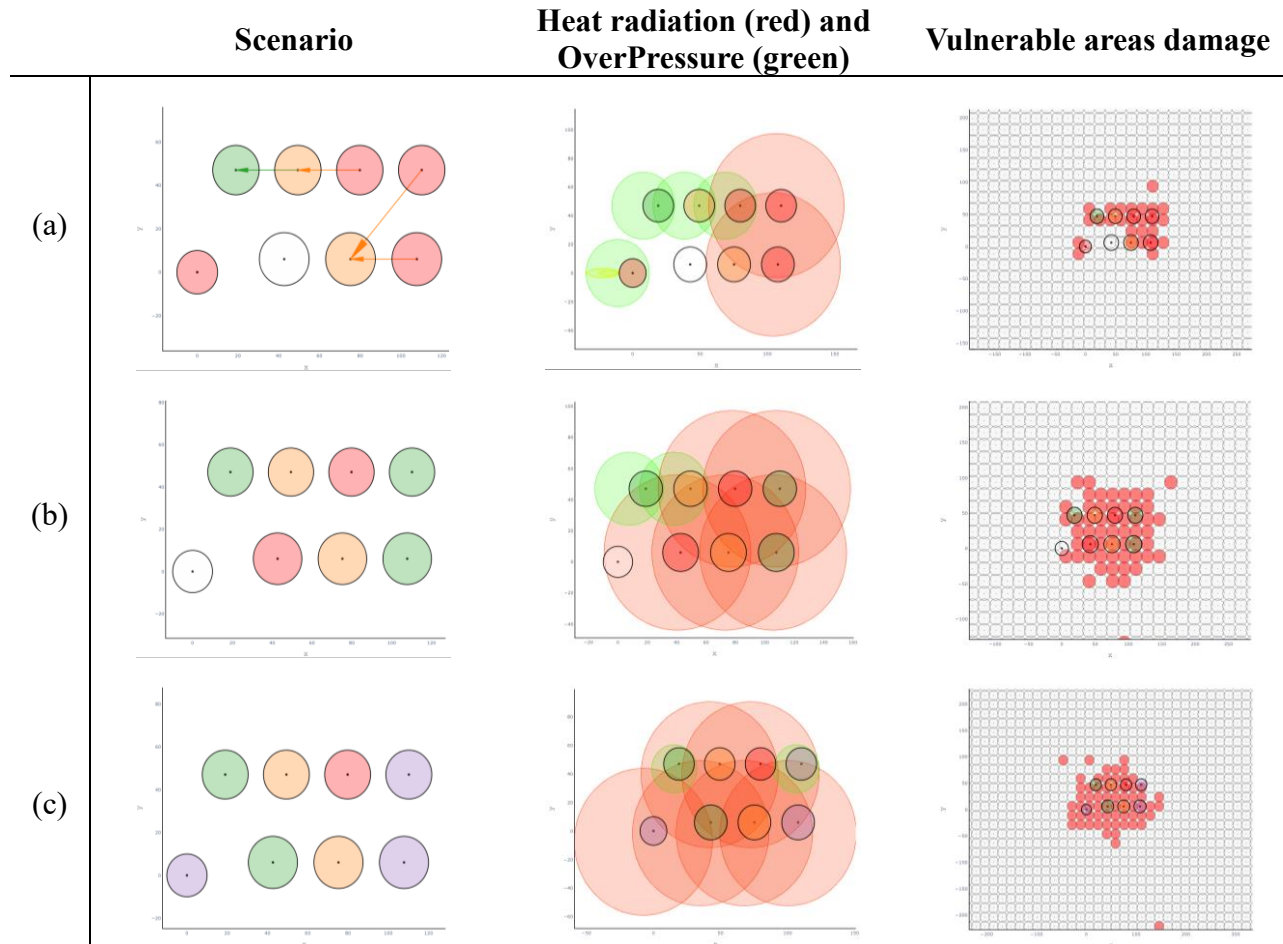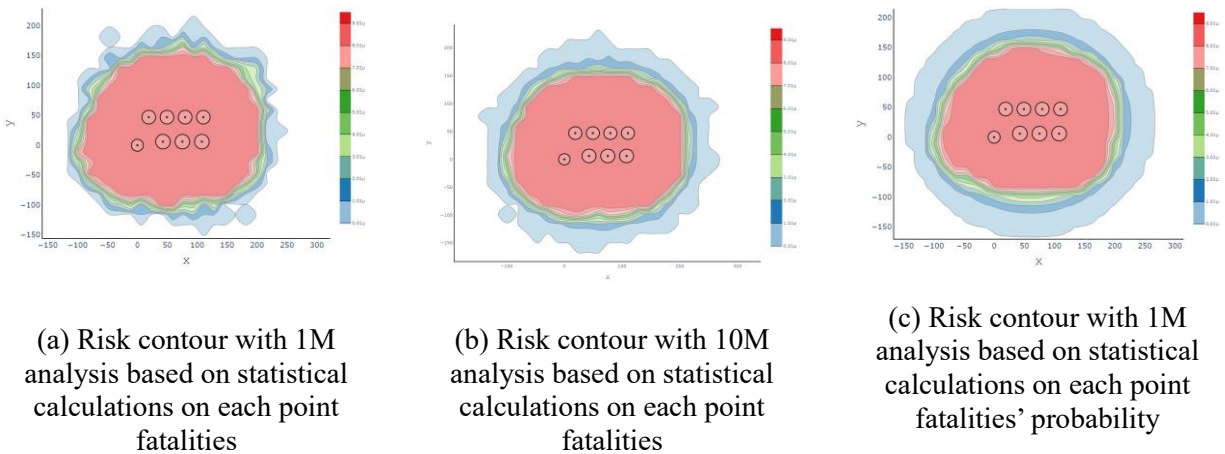
Fig. 12. Sample scenario created by OpenSRANE considering a yearly probability of occurrence of: (a) 2e-6; (b) 2e-6; (c) 1e-6. Plant unit colors in scenario column show damage levels: red: level 0; orange: level 1; green: level 2; purple: level 3. Red areas in vulnerable zones show damaged areas.

In the left column of Fig. 12(a), the scenario portrays four red tanks that have been damaged during the earthquake (damage level 0). The physical effects of the damaged red tanks have caused damage to the orange tanks (damage level 1). Finally, the physical effects of the damaged orange tanks have caused damage to the green tank (damage level 2). In the middle column of Fig. 12, heat radiation from fires (light red circles) and overpressure of explosions (light green circles) are shown according to the dose defined by the user. In the right column of Fig. 12, the damaged vulnerable areas or elements are shown. The vulnerable areas/elements are defined by the user and their vulnerability should be calculated using defined probit functions for these areas/elements.

For damage to vulnerable areas such as the one shown in the right column of Fig. 12, the individual risk can be obtained using statistical analysis at each point the vulnerable area and calculating the expected fatalities. The social risk also can be calculated by defining the population location and density, and the number of analyses or simulations required will vary depending on the required accuracy.

Fig. 13(a) shows the individual risk for range 1e-5 to 1e-8 based on the results of 1M simulations. Because of the numerical nature of the method, the results for areas with low probability values did not converge due to the lack of sufficient data. Suitable convergence can be obtained for this range by increasing the number of simulations or analyses to 10M, as shown in Fig. 13(b). Instead of increasing the number of analyses and performing statistical analysis on the number of fatalities in each vulnerable area point, it is possible to perform statistical calculations on the probability of fatality which calculated from the intensity of the resulting physical events at any point. The convergence this latter approach is much faster and the results after 1M analysis are shown in Fig. 13(c). Other results, such as those shown in [15,51,61], can be extracted from the software outputs.

| (a) Risk contour with 1M analysis based on statistical calculations on each point fatalities | (b) Risk contour with 10M analysis based on statistical calculations on each point fatalities | (c) Risk contour with 1M analysis based on statistical calculations on each point fatalities' probability |

Fig. 13. Individual risk (OpenSRANE output).

8. Conclusions

The current article introduces a flexible and extensible software for quantitative risk analysis of NaTech events. The prepared software is based on the Monte Carlo simulation method and was prepared for seismic hazards, although it can be developed for other natural hazards and other methods. The structure of the software has been explained along with an example and path to accessing the source code as well as software guidance and other relevant information has been explained. The framework of the software is based on the object combinations to provide the possibility of separating different parts of the software in order to study and develop each part separately. Because of the multi-disciplinary nature of the subject, this feature allows specialists in different fields to easily focus on their desired sections to modify or develop them independently from other sections.

The software in the form of separate sections have been implemented in Python which is a simple syntax programming language with high readability that is very similar to natural language. These features make the code more understandable and readable for developers. Users can also investigate different models and parameters using the coding capabilities. They can

benefit from the advantages of the Python libraries such as Matplotlib, Plotly, Numpy, Scipy, Pandas and Flask.

Providing a model in the programming environment is a feature of the software that may be considered as a limitation, especially for users that are not familiar with a programming language. On the other hand, the feature does allow users to use various techniques in modeling and easily interact with other types of software and libraries. From this perspective, it is an advantage of the software and not a limitation.

The ease of access to different data banks (seismic events, weather conditions, etc.) and the possibility of using them in the models is an important advantage of Python programming language. Access to the source code of the software in the GitHub environment allows researchers to access the benefits and capabilities of GitHub and share their opinions, algorithms, and codes on a common platform so that researchers can develop them.

The application of the software was demonstrated in a case study. A model was provided by the software and a quantitative seismic assessment was carried out for it. The damage scenarios with their domino effects, individual risk, heat radiation and overpressure radius for each physical effect are some the results that were exported from the software. All the results were calculated systematically because of implementation of the comprehensive algorithm. Also, the implemented algorithm proves the ability of the platform to implement various algorithms and export the desired results.

The initial algorithm for software analysis is written numerically, but the program has the ability to be modified or improved for more optimal methods and easily be developed for other analytical and numerical methods. It also can be developed for different natural hazards and can accommodate unseen elements as required by users. Equipping the software to use parallel processing has significantly increased the speed of analysis and reduce the time required for analyses with many simulation (depending on the number of computer cores used). Use of the software in the coding environment allows users to implement a variety of parametric evaluations, uncertainty and sensitivity analyses in addition to modifying and upgrading existing algorithms.

Acknowledgements

Funding

Conflicts of interest

The authors declare no conflict of interest.

References

[1]   Krausmann E, Necci A. Thinking the unthinkable: A perspective on Natech risks and Black Swans. Saf Sci 2021;139:105255. https://doi.org/10.1016/j.ssci.2021.105255.

[2]   Santella N, Steinberg LJ, Aguirra GA. Empirical Estimation of the Conditional Probability of Natech Events Within the United States. Risk Anal 2011;31:951–68. https://doi.org/10.1111/J.1539-6924.2010.01561.X.

[3]   Krausmann E, Cruz AM. Impact of the 11 March 2011, Great East Japan earthquake and tsunami on the chemical industry. Nat Hazards 2013;67:811–28. https://doi.org/10.1007/s11069-013-0607-0.

[4]   Cruz AM, Steinberg LJ. Industry preparedness for earthquakes and earthquake-triggered hazmat accidents in the 1999 Kocaeli earthquake. Earthq Spectra 2005;21:285–303. https://doi.org/10.1193/1.1889442.

[5]   Madabhushi SPG, Haigh SK, Earthquake Engineering Field Investigation Team (Great Britain), Institution of Structural Engineers (Great Britain). The Bhuj, India, earthquake of 26th January 2001 : a field report by EEFIT 2005.

[6]   Nascimento KRDS, Alencar MH. Management of risks in natural disasters: A systematic review of the literature on NATECH events. J Loss Prev Process Ind 2016;44:347–59. https://doi.org/10.1016/j.jlp.2016.10.003.

[7]   Krausmann E, Cruz AM, Salzano E. Natech Risk Assessment and Management: Reducing the Risk of Natural-Hazard Impact on Hazardous Installations. 2016.

[8]   Damle S, Mani SK, Balamurugan G. Natech guide words: A new approach to assess and manage natech risk to ensure business continuity. J Loss Prev Process Ind 2021;72. https://doi.org/10.1016/j.jlp.2021.104564.

[9]   Cruz AM. Natech disasters: a review of practices, lessons learned and future research needs. 5th Annu IIASA-DPRI Forum 2005:14–8.

[10]   Cruz AM, Suarez-Paba MC. Advances in Natech research: An overview. Prog Disaster Sci 2019;1. https://doi.org/10.1016/j.pdisas.2019.100013.

[11]   European Union. DIRECTIVE 2012/18/EU OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL of 4 July 2012 on the control of major-accident hazards involving dangerous substances, amending and subsequently repealing Council Directive 96/82/EC. Off.J. Eur. Union, L197, 1–37. 2012.

[12]   H.A.Seligson, R.T.Eguchi, K.J.Tierney, K.Richmond. Chemical Hazards, Mitigation and Preparedness in Areas of High Seismic Risk: A Methodology for Estimating the Risk of Post-EQ Hazardous Materials Rel. - MCEER: Earthquake Engineering to Extreme Events - University at Buffalo. NCEER-96-0013 1996.

[13]   Salzano E, Iervolino I, Fabbrocino G. Seismic risk of atmospheric storage tanks in the framework of quantitative risk analysis. J Loss Prev Process Ind 2003;16:403–9. https://doi.org/10.1016/S0950-4230(03)00052-4.

[14]   Cozzani V, Antonioni G, Landucci G, Tugnoli A, Bonvicini S, Spadoni G. Quantitative assessment of domino and NaTech scenarios in complex industrial areas. J Loss Prev Process Ind 2014;28:10–22. https://doi.org/10.1016/j.jlp.2013.07.009.

[15]   Alessandri S, Caputo AC, Corritore D, Giannini R, Paolacci F, Phan HN. Probabilistic risk analysis of process plants under seismic loading based on Monte Carlo simulations. J Loss Prev Process Ind 2018;53:136–48. https://doi.org/10.1016/j.jlp.2017.12.013.

[16]   Fabbrocino G, Iervolino I, Orlando F, Salzano E. Quantitative risk analysis of oil storage facilities in seismic areas. J Hazard Mater 2005;123:61–9. https://doi.org/10.1016/j.jhazmat.2005.04.015.

[17] Antonioni G, Spadoni G, Cozzani V. A methodology for the quantitative risk assessment of major accidents triggered by seismic events. J Hazard Mater 2007;147:48–59. https://doi.org/10.1016/j.jhazmat.2006.12.043.

[18] Caputo AC, Giannini R, Paolacci F. Quantitative Seismic Risk Assessment of Process Plants: State of the Art Review and Directions for Future Research. Vol. 8 Seism. Eng., vol. 8, American Society of Mechanical Engineers; 2015. https://doi.org/10.1115/PVP2015-45374.

[19] Antonioni G, Bonvicini S, Spadoni G, Cozzani V. Development of a framework for the risk assessment of Na-Tech accidental events. Reliab Eng Syst Saf 2009;94:1442–50. https://doi.org/10.1016/j.ress.2009.02.026.

[20] Sayyafzadeh B, Omidi A, Rasoolan I. Mesoscopic Generation of Random Concrete Structure Using Equivalent Space Method. J Soft Comput Civ Eng 2019;3:82–94. https://doi.org/10.22115/scce.2020.171191.1096.

[21] Rezvani N, Bolduc DL. Monte Carlo Analysis. Encycl Toxicol Third Ed 2014:393–6. https://doi.org/10.1016/B978-0-12-386454-3.00631-X.

[22] Shahsavani H, Vafaei S, Mikaeil R. Site Selection for Limestone Paper Plant Using AHP-Monte Carlo Approach. J Soft Comput Civ Eng 2019;3:36–46. https://doi.org/10.22115/scce.2018.130134.1063.

[23] Abdolhamidzadeh B, Abbasi T, Rashtchian D, Abbasi SA. A new method for assessing domino effect in chemical process industry. J Hazard Mater 2010;182:416–26. https://doi.org/10.1016/j.jhazmat.2010.06.049.

[24] Girgin S, Krausmann E. RAPID-N: Rapid natech risk assessment and mapping framework. J Loss Prev Process Ind 2013;26:949–60. https://doi.org/10.1016/j.jlp.2013.10.004.

[25] Antonioni G, Landucci G, Necci A, Gheorghiu D, Cozzani V. Quantitative assessment of risk due to NaTech scenarios caused by floods. Reliab Eng Syst Saf 2015;142:334–45. https://doi.org/10.1016/j.ress.2015.05.020.

[26] Necci A, Antonioni G, Bonvicini S, Cozzani V. Quantitative assessment of risk due to major accidents triggered by lightning. Reliab Eng Syst Saf 2016;154:60–72. https://doi.org/10.1016/j.ress.2016.05.009.

[27] Misuri A, Antonioni G, Cozzani V. Quantitative risk assessment of domino effect in Natech scenarios triggered by lightning. J Loss Prev Process Ind 2020;64. https://doi.org/10.1016/j.jlp.2020.104095.

[28] Misuri A, Landucci G, Cozzani V. Assessment of risk modification due to safety barrier performance degradation in Natech events. Reliab Eng Syst Saf 2021;212. https://doi.org/10.1016/j.ress.2021.107634.

[29] Misuri A, Ricci F, Sorichetti R, Cozzani V. The Effect of Safety Barrier Degradation on the Severity of Primary Natech Scenarios. Reliab Eng Syst Saf 2023;235:109272. https://doi.org/10.1016/J.RESS.2023.109272.

[30] Misuri A, Cozzani V. A paradigm shift in the assessment of Natech scenarios in chemical and process facilities. Process Saf Environ Prot 2021;152:338–51. https://doi.org/10.1016/J.PSEP.2021.06.018.

[31] Cozzani V, Gubinelli G, Antonioni G, Spadoni G, Zanelli S. The assessment of risk caused by domino effect in quantitative area risk analysis. J Hazard Mater 2005;127:14–30. https://doi.org/10.1016/j.jhazmat.2005.07.003.

[32] Tugnoli A, Scarponi GE, Antonioni G, Cozzani V. Quantitative assessment of domino effect and escalation scenarios caused by fragment projection. Reliab Eng Syst Saf 2022;217:108059. https://doi.org/10.1016/J.RESS.2021.108059.

[33] Sorichetti R, Misuri A, Di Padova A, Miglionico R, Tallone F, Cozzani V. Multi-risk Approach to the Qra of Natech Scenarios in the Chemical and Process Industry. Chem Eng Trans 2022;91:493–8. https://doi.org/10.3303/CET2291083.

[34] Huang K, Chen G, Khan F. Vulnerability assessment method for domino effects analysis in chemical clusters. Process Saf Environ Prot 2022;164:539–54. https://doi.org/10.1016/J.PSEP.2022.06.021.

[35] Wang J, Weng W. A simplified methodology for rapid Natech risk assessment of flood-wind-hail multi-hazard scenario. Nat Hazards 2023:1–23. https://doi.org/10.1007/S11069-022-05770-5/METRICS.

[36] Xu Y, Reniers G, Yang M, Yuan S, Chen C. Uncertainties and their treatment in the quantitative risk assessment of domino effects: Classification and review. Process Saf Environ Prot 2023;172:971–85. https://doi.org/10.1016/J.PSEP.2023.02.082.

[37] Mesa-Gómez A, Casal J, Sánchez-Silva M, Muñoz F. Advances and gaps in natech quantitative risk analysis. Processes 2021;9:1–14. https://doi.org/10.3390/pr9010040.

[38] Caputo AC, Paolacci F, Bursi OS, Giannini R. Problems and Perspectives in Seismic Quantitative Risk Analysis of Chemical Process Plants. J Press Vessel Technol Trans ASME 2019;141. https://doi.org/10.1115/1.4040804.

[39] Zhu M, McKenna F, Scott MH. OpenSeesPy: Python library for the OpenSees finite element framework. SoftwareX 2018;7:6–11. https://doi.org/10.1016/j.softx.2017.10.009.

[40] Mckenna F, Scott MH, Fenves GL. Nonlinear Finite-Element Analysis Software Architecture Using Object Composition 2010. https://doi.org/10.1061/ASCECP.1943-5487.0000002.

[41] Jiang J, Usmani A. Modeling of steel frame structures in fire using OpenSees. Comput Struct 2013;118:90–9. https://doi.org/10.1016/j.compstruc.2012.07.013.

[42] Zhu M, Scott MH. Modeling fluid-structure interaction by the particle finite element method in OpenSees. Comput Struct 2014;132:12–21. https://doi.org/10.1016/j.compstruc.2013.11.002.

[43] McKenna F. Object-oriented finite-element programming:Frameworks for analysis, algorithms, and parallel computing. Univ. of California, Berkeley, 1997.

[44] Scott MH, Fenves GL, Mckenna ; Frank, Filippou FC. Software Patterns for Nonlinear Beam-Column Models 2008. https://doi.org/10.1061/ASCE0733-94452008134:4562.

[45] Scott MH, Asce AM, Haukaas T. Software Framework for Parameter Updating and Finite-Element Response Sensitivity Analysis 2008. https://doi.org/10.1061/ASCE0887-3801200822:5281.

[46] Hosseini Gelekolai SM, Kenarangi H. Application of Opensees Software in Structural Modeling and Analysis. Tehran, Iran: Azadeh Publications.(In Persian); 2013.

[47] Nasrazadani H, Mahsuli M. Probabilistic Framework for Evaluating Community Resilience: Integration of Risk Models and Agent-Based Simulation. J Struct Eng 2020;146:04020250. https://doi.org/10.1061/(ASCE)ST.1943-541X.0002810.

[48] http://www.dnv.com. PHAST n.d.

[49] Egidi D, Foraboschi FP, Spadoni G, Amendola A. The ARIPAR project: analysis of the major accident risks connected with industrial and transportation activities in the Ravenna area. Reliab Eng Syst Saf 1995;49:75–89. https://doi.org/10.1016/0951-8320(95)00026-X.

[50] Spadoni G, Egidi D, Contini S. Through ARIPAR-GIS the quantified area risk analysis supports land-use planning activities. J Hazard Mater 2000;71:423–37. https://doi.org/10.1016/S0304-3894(99)00091-6.

[51] Caputo AC, Vigna A. Numerical Simulation of Seismic Risk and Loss Propagation Effects in Process Plants: An Oil Refinery Case Study. Am Soc Mech Eng Press Vessel Pip Div PVP 2017;8. https://doi.org/10.1115/PVP2017-65465.

[52] Caputo AC, Kalemi B, Paolacci F, Corritore D. Computing resilience of process plants under Na-Tech events: Methodology and application to sesmic loading scenarios. Reliab Eng Syst Saf 2020;195. https://doi.org/10.1016/j.ress.2019.106685.

[53] C.J.H. van den Bosch RAPMWNJDEABWPMMAC van den BWFJMEJCAM van DRMM van W. Methods for the calculation of Physical Effects Due to releases of hazardous materials (liquids and gases). 2005.

[54] Joaquim Casal. Evaluation of the Effects and Consequences of Major Accidents in Industrial Plants. Elsevier; 2018. https://doi.org/10.1016/C2016-0-00740-4.

[55] Wattanakriengkrai S, Chinthanet B, Hata H, Kula RG, Treude C, Guo J, et al. GitHub repositories with links to academic papers: Public access, traceability, and evolution. J Syst Softw 2022;183. https://doi.org/10.1016/j.jss.2021.111117.

[56] Escamilla E, Klein M, Cooper T, Rampin V, Weigle MC, Nelson ML. The Rise of GitHub in Scholarly Publications 2022.

[57] Boostan E, Tahernia N, Shafiee A. Fuzzy—probabilistic seismic hazard assessment, case study: Tehran region, Iran. Nat Hazards 2015;77:525–41. https://doi.org/10.1007/s11069-014-1537-1.

[58] Razzaghi MS, Eshghi S. Probabilistic Seismic Safety Evaluation of Precode Cylindrical Oil Tanks. J Perform Constr Facil 2015;29. https://doi.org/10.1061/(asce)cf.1943-5509.0000669.

[59] Necci A, Antonioni G, Krausmann E, Argenti F, Landucci G, Cozzani V. Accident Scenarios Caused by Lightning Impact on Atmospheric Storage Tanks. vol. 32, 2013.

[60] Ibrahim HA, Syed HS. Hazard Analysis of Crude Oil Storage Tank Farm. Int J ChemTech Res 2018;11:300–8. https://doi.org/10.20902/ijctr.2018.111132.

[61] Alessandri S, Caputo AC, Corritore D, Renato G, Paolacci F, Phan HN. On the Use of Proper Fragility Models for Quantitative Seismic Risk Assessment of Process Plants in Seismic Prone Areas. Vol. 8 Seism. Eng., vol. 8, American Society of Mechanical Engineers; 2017. https://doi.org/10.1115/PVP2017-65137.