



HAL
open science

A Simulation Framework Dedicated to Characterizing Risks and Cascading Effects in Collaborative Networks

Tianyuan Zhang, Jiayao Li, Frederick Benaben

► **To cite this version:**

Tianyuan Zhang, Jiayao Li, Frederick Benaben. A Simulation Framework Dedicated to Characterizing Risks and Cascading Effects in Collaborative Networks. PRO-VE 2022 - 23rd IFIP WG 5.5 Working Conference on Virtual Enterprises, Sep 2022, Lisbonne, Portugal. 463-474 (chap. 37), 10.1007/978-3-031-14844-6_37. hal-03775883

HAL Id: hal-03775883

<https://imt-mines-albi.hal.science/hal-03775883v1>

Submitted on 3 Oct 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

A Simulation Framework Dedicated to Characterizing Risks and Cascading Effects in Collaborative Networks

Tianyuan Zhang^(✉), Jiayao Li, and Frédérick Bénaben

Centre Génie Industriel, IMT Mines Albi-Carmaux, 81000 Albi, France
{tianyuan.zhang, jiayao.li, frederick.benaben}@mines-albi.fr

Abstract. Cascading effects describe risk interdependencies, whereby the occurrence of one risk may trigger one or more risks with potential propagation chains in complex systems. In this study, on the basis of a formalized model namely *danger-risk-consequence* chain, a generic simulation framework is proposed to characterize risk causal processes and cascading effects within collaborative networks. Risk-related components and the causal relationships between them are visualized by abstractly representing the instantaneous state of the considered collaborative network as a directed graph. Furthermore, the simulation of trajectories of the state evolution over time is realized by knowledge-driven automatic inference of causal chains and propagation chains, thus enabling the tracing of cascading effects within complex systems. The presented simulation framework provides a solid foundation for a systemic understanding of risks, which implies an innovative tool that helps decision-makers to identify, prevent and mitigate cascading effects within collaborative networks (e.g., supply chains).

1 Introduction

Over the past decades, participation in collaborative networks has become a vital avenue for any organization to improve profitability through sharing competencies and resources [1]. Organizations in collaborative networks are more exposed to risks since they are interdependent [2, 3]. In the context of enterprise collaboration, the risk of an individual company is no longer confined to its own risk; it is in fact the risk of all partners [4]. *Cascading effects* are implied by such risk interrelations. Specifically, as described by Buldyrev et al. [5], “*the occurrence of one risk can trigger further risks, thus creating cascading effect*”. It signifies a high vulnerability to perturbations with amplifying impacts. Therefore, it’s important to identify, prevent and mitigate cascading effects within collaborative networks for enterprises.

To date, risk interdependencies and cascading effects have attracted extensive attention in academic research. The vast majority of studies fall within the context of the

supply chain, which can be regarded as a typical collaborative network. The available studies can be broadly classified into three categories according to their purposes: 1) vulnerability analysis in the context of cascading effects; 2) evaluation of the impact of cascading effects; 3) conceptual modeling of cascading effects [6–9]. Among the above research, simulation approaches were adopted in most of the studies for the following reasons:

First, cascading effects are caused by low-frequency high-impact disruptions in most cases [10, 11]. There are fewer real cases and data available for study. Simulation can be used to address this rareness.

Second, the ultimate goal of studying cascading effects is to find prevention and mitigation strategies. Simulation can provide a risk-free environment for testing different prevention and mitigation measures.

Third, cascading effects are dynamic processes. Simulation is dedicated to tracking the evolution and characterizing the dynamic nature, and thus helping to understand the triggers and mechanisms of cascading effects [12].

Although simulation methods are widely used in cascading effects research, most of the studies use simulation only as a tool to simulate the behavior of complex systems experiencing cascading effects in a specific context, or for a specific case. There is a lack of a generic simulation framework dedicated to characterizing risks and cascading effects in collaborative networks. This study aims to address this absence by constructing a novel simulation framework on the basis of a formalized model namely *danger-risk-consequence* chain.

The article is organized as follows. Section 2 presents a literature review of cascading effect simulation-based modeling and highlights the limitations of existing research. Section 3 provides a brief introduction of the preliminary work, mainly the *danger-risk-consequence* chain model. Section 4 introduces the proposed simulation framework and Sect. 5 presents an illustrative use case based on the proposed framework. The final section concludes the contributions and limitations of this work, as well as the future directions.

2 Literature Review

The term of cascading effect has been introduced into academic literatures over the last two decades and has received significant attention in a wide range of enquiries. According to [13], cascading effect is consistent with the metaphor of topping dominoes, that is, the first domino is pushed down, it crashes into the next domino and keeps pushing down to the end of the sequence. FEMA describes cascading effect as a universal dynamic form that may multiply the impact of combinations of hazards from a perspective of emergency management [14]. Cascading effect is also related to ripple effect that concerns disruption propagation in supply chains [15]. Generally speaking, this effect is “*the dynamics present, in which an initial impact can trigger other phenomena that lead to consequences with significant magnitudes*” [16].

As suggested by Helbing et al. [3], cascading effects can be considered as a direct output of the evolution of complex systems, which means that they change constantly over time. Hence, simulation-based modeling naturally performs an important role on

cascading effect studies, given its ability to cope with time-dependent risk analysis, vulnerability analysis and performance impact [15].

Llaguno et al. [10] proposes a conceptual framework to present the characteristics of ripple effect in supply chains, which is validated by a system dynamics simulation model. Ghadge et al. [12] utilizes system dynamics approach to simulate different types of risks and cascading effects. Besides, agent-based simulation and discrete-event simulation have been widely used to model cascading effects and their resilience performance. For example, Lohmer et al. [17] quantitatively presents an agent-based simulation model to analyze ripple effects and the impact of introducing blockchain technology on supply chain resilience. Ivanov [18] simulates a coronavirus outbreak case by using discrete-event modeling, which demonstrates how simulation-based methodology can be used to predict the impacts of disruptions and cascading effects. Furthermore, in addition to the simulation methods introduced above, graph-based studies also need to be named for cascading effect analysis [15]. For instance, Khakzad and Reniers [6] proposes a set of graph metrics and measurements for vulnerability analysis of process plant, given that cascading effects are represented as a directed graph.

To summarize, significant strides can be recently witnessed from academic literatures to cascading effect simulation-based modeling. However, on one hand, most of studies focus on utilizing the simulation methods to analyze resilience and robustness performance for a considered complex system in the light of cascading effects [10, 17, 19], instead of simulating cascading effects. On the other hand, there are plenty of simulation models proposed either in the context of a specific domain (*e.g.*, supply chains [12, 15]) or on a specific case [18]. In such cases, we argue that such simulation tools are lack of university and flexibility, which may be hard to reuse or reproduce under other contexts. In a nutshell, there is a lack of a more generic simulation framework dedicated to characterizing the causes and propagation processes of cascading effects. Our study aims to bridge this research gap.

3 Preliminary Work

To simulate risks and cascading effects within collaborative networks, a formalized model that can characterize the risk causal processes and propagation processes is needed. A conceptual approach, namely *danger-risk-consequence* chain [9, 20, 21], dedicated to modeling risks and cascading effects within complex systems is adopted as the basis of the proposed simulation framework.

As shown in Fig. 1, the *danger-risk-consequence* chain was firstly introduced in [20] as a formal description of risk dependencies. To conceptualize the risk causal processes, five risk-related components (*danger*, *stake*, *risk*, *event*, and *consequence*) are identified to represent the internal causal mechanism of risks [9, 21]. The definitions of the five risk-related components used in the *danger-risk-consequence* chain are stated as follows.

Danger is defined as the hazardous characteristic or situation of the considered collaborative network that may lead to undesirable negative impacts. *Stake* is defined following the instruction of ISO 7010:2019, as “*an item, thing or entity that has potential or actual value to a considered system or its environment*” [22]. Compared with the conceptual description, *risk* is further clarified as the latent effect of a *danger* acting on

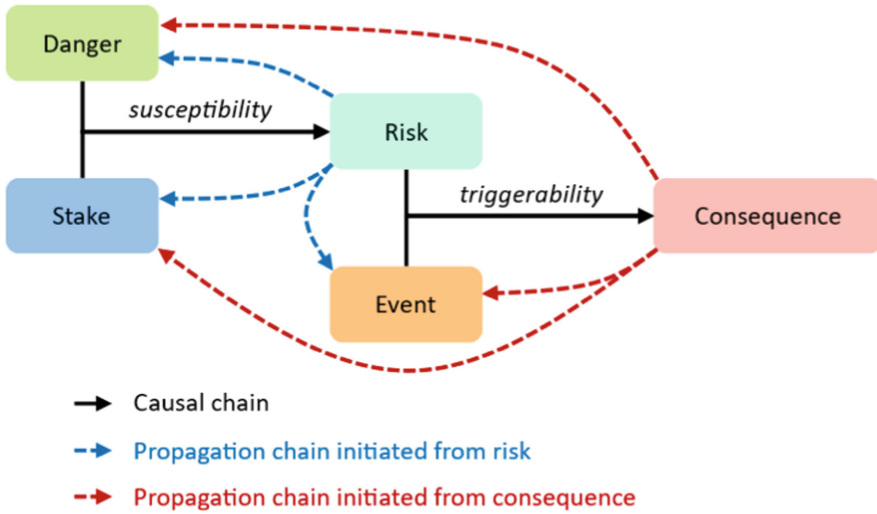


Fig. 1. The formalized *danger-risk-consequence* chain model [9, 21]. (Color figure online)

a *stake*. *Event* can be seen as a fact that has occurred or a condition that has been fulfilled, marking the transition of a *risk* from a potential state to a realized state. *Consequence* can be defined as the undesirable negative impacts caused by the realization of the *risk* [9, 21]. Compatible with the broad understanding of risks, *danger* can be regarded as the source of *risk*, *stake* can be considered as the *risk*-bearing entity, *event* can be seen as the trigger of *risk*, while *consequence* can be described as the impacts of *risk*.

By breaking down the complex concept of risks into five risk-related components, the *danger-risk-consequence* chain summarized the causal mechanism of risks into two ternary relations: *susceptibility* and *triggerability*. *Susceptibility* describes the causal relationship between *danger*, *stake*, and *risk*, that is, a *stake* is susceptible to a *danger* and thus generates a *risk*. Similarly, *triggerability* defines the causal relationship between *risk*, *event*, and *consequence*, that is, an *event* triggers a *risk* and thus realizes a *consequence* [9, 21]. In summary, the causal processes of risks are formally described through the two causal chains (i.e., *susceptibility* and *triggerability*) in the *danger-risk-consequence* chain model.

In addition to risk causal processes, the *danger-risk-consequence* chain model characterizes the cascading effect by introducing the propagation chain to represent the risk propagation processes. *Risk* and *consequence* can be regarded as the results in causal chains, on the contrary, these two components are treated as the causes that generate impacts on *danger*, *stake*, and *event* in propagation chains. Therefore, risk propagation processes are formalized as the potential propagation connections initiated from *risk* (blue connections in Fig. 1) and *consequence* (red connections in Fig. 1), respectively. It is worth noting that the source and target of the propagation chain may belong to the same causal chain or be separated into different causal chains. A series of potential propagation chains being activated means that the cascading effect takes place.

Compared with the descriptive definitions of risks and cascading effects, the *danger-risk-consequence* chain model provides a highly formalized tool that can be used to simulate the risk causal processes and propagation processes. Furthermore, instead of modeling the collaborative networks, the *danger-risk-consequence* chain model directly interprets risk causal mechanism and propagation phenomena, thus enables the direct simulation of risks and cascading effects.

4 The Proposed Simulation Framework

Based on the preliminary work, especially the *danger-risk-consequence* chain model, a generic simulation framework is proposed to simulate the risk causal processes and cascading effects in collaborative networks. Inspired by the philosophies of agent-based modeling and graph theory, the proposed simulation framework is designed as a hybrid approach through object-oriented design (OOD) and is implemented with the python (version 3.10.0) programming language.

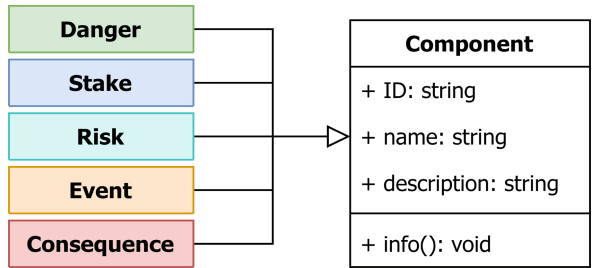


Fig. 2. The UML class diagram that illustrates the OOD of the proposed simulation framework regarding the five risk-related components.

4.1 The OOD of the *Danger-Risk-Consequence* Chain Model

As the basis of the proposed simulation framework, the *danger-risk-consequence* chain model is implemented first, which defines the core elements that the simulation framework can operate on. As shown in Fig. 2, the five risk-related components of the *danger-risk-consequence* chain model are defined as five classes that inherit from the same class: “Component”. Thus, the five classes (“Danger”, “Stake”, “Risk”, “Event”, and “Consequence”) share the same attributes: “ID”, “name”, and “description”. It is important to note that each instance has a unique “ID”, but different instances can have the same “name” and “description”. Sharing the same “name” and “description” means that two different instances belong to the same identified risk-related component, but the different “IDs” indicate that they may exist in different time and space and are different instances. In addition to the attributes, all five classes provide the method “info()” to obtain corresponding information.

Apart from the five risk-related components, the proposed simulation framework implements the risk causal chain and propagation chain as two packages respectively.

The “Causal Chain” package contains two classes: “Susceptibility” and “Triggerability”. Each class has three attributes to record all known components, and an additional attribute to record the existence of causal relationships corresponding to all possible combinations of these components. Similarly, the “Propagation Chain” package also contains two classes: “Creation” and “Deletion”. The “Creation” class describes the propagation relationship that a *risk/consequence* might create a *danger/stake/event*, the “Deletion” class describes the propagation relationship that a *risk/consequence* might delete a *danger/stake*. An *event* cannot be deleted because it represents a fact that has occurred or a condition that has been fulfilled. All these four classes provide the method “check()” to check whether the corresponding causal or propagation chain exists (Fig. 3).

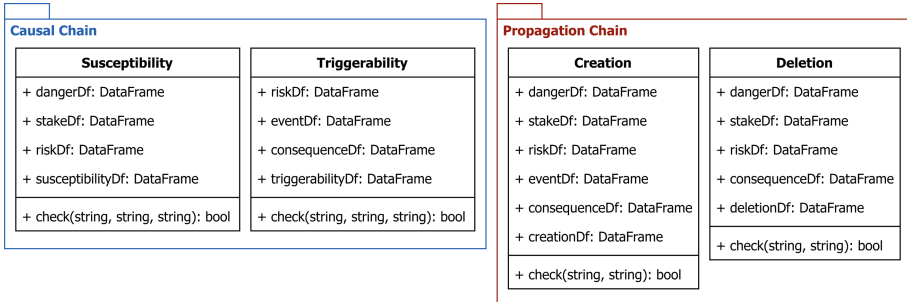


Fig. 3. The UML class diagram that illustrates the object-oriented design of the proposed simulation framework regarding the risk causal chains and propagation chains.

4.2 The OOD of the Proposed Simulation Framework

After defining the core elements that can be manipulated, the proposed simulation framework is designed to consist of four classes: “KnowledgeBase”, “State”, “Action”, and “Experiment”. Figure 4 illustrates the overall picture of the whole framework, which describes the relationships between these four classes. The “KnowledgeBase” class is designed to store all the knowledge of the simulated collaborative network, which involves all the identified risk-related components as well as causal and propagation chains between them. Therefore, the “KnowledgeBase” can be used to characterize the simulated system and drive the simulation process.

With the support of knowledge, the “State” class is defined to describe an instantaneous state of the simulated collaborative network. Each instantaneous state is formalized as a directed graph derived from the *danger-risk-consequence* chain model. For causal chains, the binary relations are adopted to replace the ternary ones in the original model for better compatibility with the form of directed graphs (see Fig. 5). More specifically, *susceptibility* is split into 3 binary relations: a *stake* is susceptible to a *danger*, a *danger* generates a *risk*, and a *risk* concerns a *stake*. Similarly, *triggerability* is split as follows: an *event* triggers a *risk*, a *risk* defines a *consequence*, and a *consequence* is realized by an *event*. The directed graph describes all risk-related components present in the collaborative network at the moment, and the causal chains between them. In addition, for

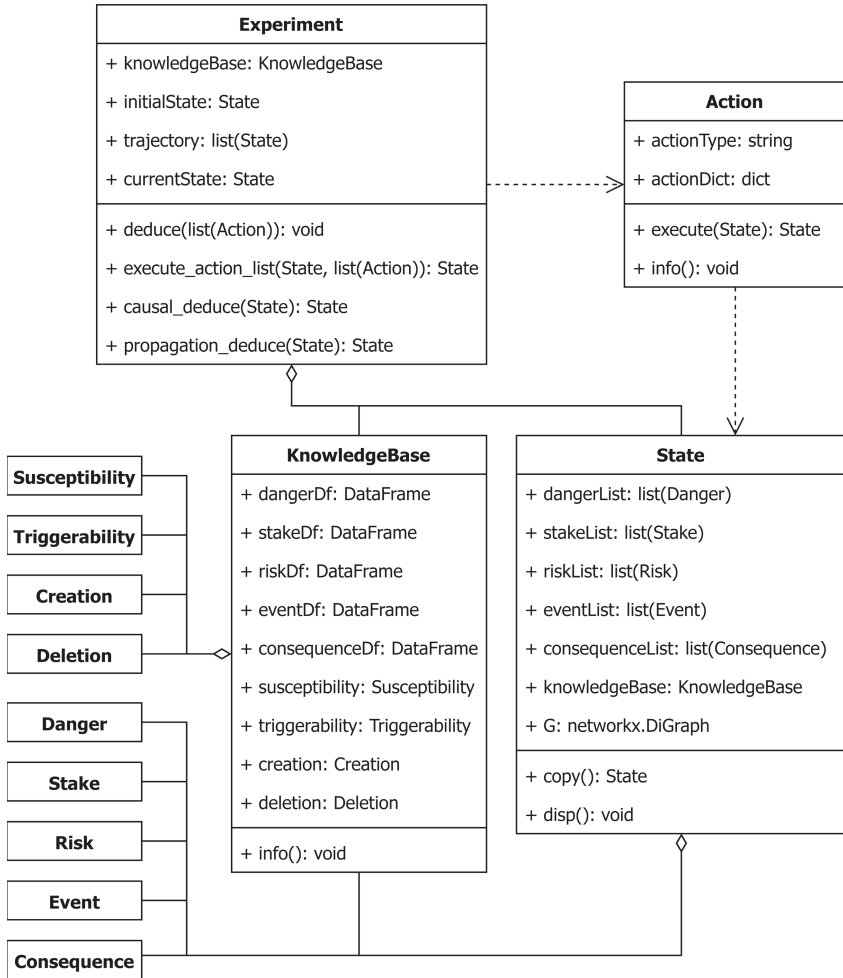


Fig. 4. The UML class diagram that illustrates the OOD of the whole proposed simulation framework.

propagation chains, the binary relation “Creation” is also recorded in the “State” class and presented in the directed graph to better visualize and track cascading effects. The python package NetworkX (version 2.6.3) is used for implementing the directed graph [23].

Apart from the “KnowledgeBase” and “State” class, the proposed simulation framework also provides an “Action” class to capture the actual situation that a collaborative network might face disruptions and interventions. The “Action” class has a method “execute()” that takes a “State” as input and returns a “State” as the result after the disruption or intervention has been executed. This class can be used to break the stable state of the system during a simulation or to test the impact of risk management measures on the system.

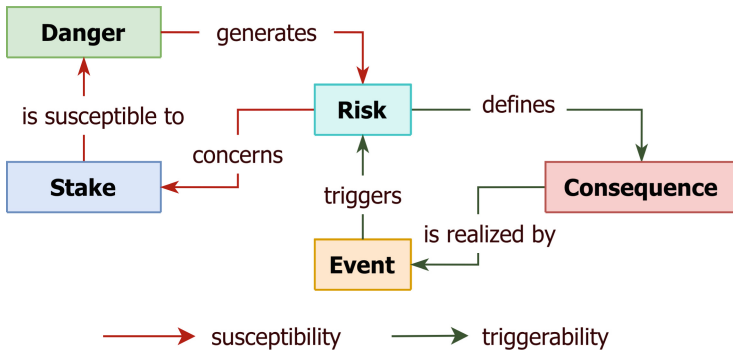


Fig. 5. Illustration of the binary causal relationships between five risk-related components in the directed graph.

Aggregating and depending on the above three class, the “Experiment” class is designed for implementing the simulation experiment given a specific collaborative network. The “knowledgeBase” attribute provides knowledge of the considered system for driving the inference, the “initialState” and “currentState” attribute represents the starting point and the status of the simulated system respectively, while the “trajectory” attribute is composed of a series of states and records the state evolution over time. The “Experiment” class provides a method “deduce()” to perform the automatic knowledge-driven deduction. Each deduction step consists of three sub-steps: first, a list of actions is executed to inject disruptions or intervention into the simulated system; second, the causal relationships are inferred based on the identified causal chains in the knowledge base, new *risk/consequence* are generated; third, new *danger/stake/event* are created, or existing *danger/stake* are deleted according to the known propagation chains in the knowledge base. After each deduction step is completed, the resulting instance of “State” will be added to the “trajectory” attribute, and the “currentState” attribute will be updated.

It is worth noting that risks as well as cascading effects in a collaborative network are largely determined by properties of the network itself, such as the structure of the network. Therefore, the simulation framework proposed in this study defines the “knowledgeBase” attribute to describe the related characteristics of the considered system as knowledge to support the simulation of risks and cascading effects. Furthermore, the “trajectory” attribute and the “deduce()” method are implemented to capture the dynamic nature of interdependent risks and the cascading effects. Instead of simulating how a collaborative network behaves, the proposed simulation framework directly simulates risk causal and propagation processes, thus provides a powerful tool for better understanding the trigger and mechanism of risks and cascading effects from both the macro- and micro-level perspectives.

5 An Illustrative Use Case

For illustrative purposes, a use case is simulated using the proposed framework in a scenario of construction supply chain, which can be considered as a classical collaborative network. The simulated supply chain consists of three stakeholders: the material

supplier, subcontractor, and contractor. The subcontractor needs to pay the material supplier in advance and to receive the payment from the contractor only after the delivery is completed. Therefore, the subcontractor's cash flow is an important indicator of the status of this supply chain. However, with only one material supplier in the initial supply chain, the network faces the intrinsic risk of supply delays or disruptions. Once this risk is triggered, potential cascading effects will be activated, ultimately resulting in the tightness of subcontractor's circulating assets and the break of the stable status of the collaborative network.

Table 1. Identified risk-related components of the simulated construction supply chain.

Type	Name	Description
<i>Danger</i>	D1	Single source of supply on cements
	D2	Temporary procurement from a new supplier
	D3	The subcontractor pays before receiving required materials
<i>Stake</i>	S1	Subcontractor
	S2	Material Supplier
	S3	Contractor
<i>Risk</i>	R1	Supply delay or disruption
	R2	The quality of cements can't be guaranteed
	R3	The after-sale service can't be guaranteed
	R4	Unable to recover the advance payment
<i>Event</i>	E1	The production of the cement supplier is interrupted
	E2	The rate of defective products is too high
	E3	The new supplier refuses of return
	E4	The material supplier refuses to refund advance payment
<i>Consequence</i>	C1	The subcontractor fails to deliver on time
	C2	Return is needed
	C3	The new supplier fails to deliver all the required cements
	C4	Bad debts, tightness of circulating assets

To characterize the construction supply chain and drive the simulation of risks and cascading effects considering the above context, a series of risk-related components are extracted based on expertise and historical data (see Table 1). These components are organized using the *danger-risk-consequence* chain model, and the causal chains and propagation chains between them are also identified as knowledge.

The initial state of the simulated system consists only two components: D1 and S1. Driven by the knowledge base, a simulation experiment is performed. There are two disruptions (E1 after State 1 and E2 after State 3) injected into the system during the simulation. The resulting trajectory of the state evolution over time is shown in

Fig. 6. After each automatic deduction, the directed graph representing the instantaneous state becomes more and more complex. New components are added according to the simulated risk causal processes, and multiple risk propagation processes are activated during the state evolution. In contrast to other commonly used simulation methods, the risk causal relationships are initiatively visualized, and the underlying cascading effect are successfully simulated and tracked through the proposed simulation framework.

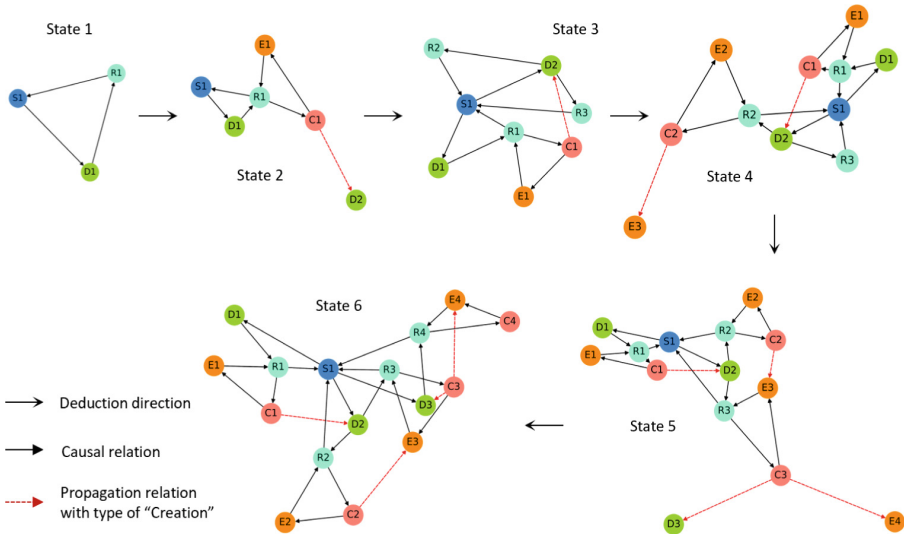


Fig. 6. The simulated trajectory of the state evolution over time.

6 Conclusion and Future Work

This study proposes a generic simulation framework dedicated to characterizing risks and cascading effects in collaborative networks. Based on a formalized *danger-risk-consequence* chain model, the presented framework is able to simulate risk casual processes and propagation processes and thus can directly simulate cascading effect itself, rather than reflecting it indirectly by simulating the behavior of a considered system. Furthermore, the static knowledge base in the framework can be used to organize the risk-related characteristics of the considered systems, while the dynamic trajectories obtained from simulations can be used to capture the dynamic nature of cascading effect. The simulation framework is implemented with the python (version 3.10.0) programming language, thus is easy to be coupled with other powerful tools in domains, such as scikit-learn for machine learning and pandas for data sciences.

The future work will aim to make the proposed simulation framework easier to apply in practice, with the following directions to be explored:

- 1) Automatic mapping the considered collaborative network as a static knowledge base following the form defined by the *danger-risk-consequence* chain model;

- 2) Identifying and mining of the causal relationships and propagation relationships among risk-related components;
- 3) Refining the current simulation framework, introducing stochasticity regarding the occurrence of disruptions, providing tools for testing human intervention measures and evaluating the impacts of cascading effects.

References

1. Camarinha-Matos, L.M., Afsarmanesh, H., Galeano, N., Molina, A.: Collaborative networked organizations - concepts and practice in manufacturing enterprises. *Comput. Ind. Eng.* **57**, 46–60 (2009). <https://doi.org/10.1016/j.cie.2008.11.024>
2. Wulan, M., Petrovic, D.: A fuzzy logic based system for risk analysis and evaluation within enterprise collaborations. *Comput. Ind.* **63**, 739–748 (2012). <https://doi.org/10.1016/j.com.pind.2012.08.012>
3. Helbing, D.: Globally networked risks and how to respond. *Nature* **497**, 51–59 (2013). <https://doi.org/10.1038/nature12047>
4. Hallikas, J., Varis, J.: Risk management in value networks. In: Zsidisin, G.A., Ritchie, B. (eds.) *Supply Chain Risk: A Handbook of Assessment, Management, and Performance*, pp. 35–52. Springer, Cham (2009). https://doi.org/10.1007/978-0-387-79934-6_3
5. Buldyrev, S.V., Parshani, R., Paul, G., Stanley, H.E., Havlin, S.: Catastrophic cascade of failures in interdependent networks. *Nature* **464**, 1025–1028 (2010)
6. Khakzad, N., Reniers, G.: Using graph theory to analyze the vulnerability of process plants in the context of cascading effects. *Reliab. Eng. Syst. Saf.* **143**, 63–73 (2015). <https://doi.org/10.1016/j.ress.2015.04.015>
7. Li, Y., Chen, K., Collignon, S., Ivanov, D.: Ripple effect in the supply chain network: forward and backward disruption propagation, network health and firm vulnerability. *Eur. J. Oper. Res.* **291**, 1117–1131 (2021)
8. Ma, G., Wu, Z., Jia, J., Shang, S.: Safety risk factors comprehensive analysis for construction project: combined cascading effect and machine learning approach. *Saf. Sci.* **143**, 105410 (2021)
9. Li, J., Bénaben, F., Gou, J., Mu, W.: A proposal for risk identification approach in collaborative networks considering susceptibility to danger. In: Camarinha-Matos, L.M., Afsarmanesh, H., Rezgui, Y. (eds.) *PRO-VE 2018. IAICT*, vol. 534, pp. 74–84. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-99127-6_7
10. Llaguno, A., Mula, J., Campuzano-Bolarin, F.: State of the art, conceptual framework and simulation analysis of the ripple effect on supply chains. *Int. J. Prod. Res.* (2021). <https://doi.org/10.1080/00207543.2021.1877842>
11. Kinra, A., Ivanov, D., Das, A., Dolgui, A.: Ripple effect quantification by supplier risk exposure assessment. *Int. J. Prod. Res.* **58**, 5559–5578 (2020). <https://doi.org/10.1080/00207543.2019.1675919>
12. Ghadge, A., Er, M., Ivanov, D., Chaudhuri, A.: Visualisation of ripple effect in supply chains under long-term, simultaneous disruptions: a system dynamics approach. *Int. J. Prod. Res.* (2021). <https://doi.org/10.1080/00207543.2021.1987547>
13. May, F.: Cascading disaster models in postburn flash flood. In: Butler, B.W., Cook, W.C. (eds.) *The Fire Environment—Innovations, Management, and Policy; Conference Proceedings*, 26–30 March 2007; Destin, FL. Proceedings RMRS-P-46CD. Fort Collins, CO: U.S. Department of Agriculture, Forest Service, Rocky Mountain Research Station, CD-ROM, pp. 443–464, 046 (2007)

14. FEMA: Principles of Emergency Management. Washington, DC (2006)
15. Ivanov, D.: Simulation-based ripple effect modelling in the supply chain. *Int. J. Prod. Res.* **55**, 2083–2101 (2017). <https://doi.org/10.1080/00207543.2016.1275873>
16. Pescaroli, G., Alexander, D.: A definition of cascading disasters and cascading effects: Going beyond the “toppling dominos” metaphor. *Planet@Risk*, vol. 3 (2015)
17. Lohmer, J., Bugert, N., Lasch, R.: Analysis of resilience strategies and ripple effect in blockchain-coordinated supply chains: an agent-based simulation study. *Int. J. Prod. Econ.* **228**, 107882 (2020). <https://doi.org/10.1016/j.ijpe.2020.107882>
18. Ivanov, D.: Predicting the impacts of epidemic outbreaks on global supply chains: a simulation-based analysis on the coronavirus outbreak (COVID-19/SARS-CoV-2) case. *Transp. Res. Part E Logistics Transp. Rev.* **136**, 101922 (2020). <https://doi.org/10.1016/j.tre.2020.101922>
19. Giannoccaro, I., Iftikhar, A.: Mitigating ripple effect in supply networks: the effect of trust and topology on resilience. *Int. J. Prod. Res.* **60**, 1178–1195 (2022). <https://doi.org/10.1080/00207543.2020.1853844>
20. Bénaben, F., Barthe-Delanoë, A.-M., Lauras, M., Truptil, S.: Collaborative systems in crisis management: a proposal for a conceptual framework. In: Camarinha-Matos, L.M., Afsarmanesh, H. (eds.) *PRO-VE 2014. IAICT*, vol. 434, pp. 396–405. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-662-44745-1_39
21. Li, J.: Cascading effects of risk dedicated to collaborative networked organizations. Doctoral dissertation, Beijing Jiaotong University, Beijing (2020)
22. ISO/TC 145/SC 2 - Safety identification, signs, shapes, symbols and colours. <https://www.iso.org/cms/render/live/en/sites/isoorg/contents/data/committee/05/26/52692.html>. Accessed 5 May 2022
23. NetworkX—NetworkX documentation. <https://networkx.org/>. Accessed 5 May 2022