

## Article

# Verifying the Smart Contracts of the Port Supply Chain System Based on Probabilistic Model Checking

Yang Liu <sup>1,2</sup>, Ziyu Zhou <sup>1</sup>, Yongsheng Yang <sup>1</sup> and Yan Ma <sup>2,3,\*</sup>

<sup>1</sup> Institute of Logistics Science and Engineering, Shanghai Maritime University, Shanghai 200120, China; lyang@shmtu.edu.cn (Y.L.); 201930510012@stu.shmtu.edu.cn (Z.Z.); 195558@shmtu.edu.cn (Y.Y.)

<sup>2</sup> School of Computing, National University of Singapore, Singapore 117417, Singapore

<sup>3</sup> School of Accounting, Nanjing University of Finance and Economics, Nanjing 210023, China

\* Correspondence: yanma@nufe.edu.cn

**Abstract:** Port supply chains play a very important role in the process of economic globalization. Lack of trust of the mechanism is the main factor in restricting the development of port supply chains. Blockchains have great potential to solve the trust problem among all participants using port supply chains, which can reduce costs and improve efficiency. As the bridge between blockchains and port supply chains, smart contracts reconstruct the business process of blockchain-empowered port supply chains. In this article, we present an architecture of a consortium blockchain-empowered port supply chain system, and propose a system verification framework for the smart contracts of port supply chains with probabilistic behaviors. The smart contracts are modeled as DTMCs (Discrete-Time Markov Chains), which are automatically transformed through the BPMN (Business Process Model and Notation) description of the smart contracts. The requirements are specified by PCTL (Probabilistic Computation Tree Logic). Moreover, we implement the customs clearance process of the Shanghai Yangshan Port based on blockchain Hyperledger Fabric, and reconstruct the clearance process with smart contracts. We use it to demonstrate the effectiveness of this framework, and identify the smart contracts that do not meet the expected needs of users.

**Keywords:** port supply chain; blockchain; smart contract; probabilistic model checking

**Citation:** Liu, Y.; Zhou, Z.; Yang, Y.; Ma, Y. Verifying the Smart Contracts of the Port Supply Chain System Based on Probabilistic Model Checking. *Systems* **2022**, *10*, 19. <https://doi.org/10.3390/systems10010019>

Academic Editors: Ed Pohl

Received: 9 December 2021

Accepted: 10 February 2022

Published: 16 February 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

With the development of economic globalization, the world economy has increasingly become a closely integrated whole system. Port supply chains play a very important role in this process, as they undertake about 80% of the total international trade [1]. A port supply chain refers to when a city center uses its own port to develop the comprehensive service systems that cover all of the functions of supply chains, which exploit advanced ICT (Information and Communication Technologies) to optimize the port's resources and strengthen its radiation ability [2,3]. It takes a port as the nexus that supports interaction between global supply chains and regional production and consumption markets. Port supply chains have become a new engine of national economic development, an important part of national economy, and one of the important symbols to measure the level of modernization and comprehensive national strength of a country [2]. At present, emerging technologies such as the Internet of Things, Cloud Computing and Big Data are promoting the continuous expansion of port supply chains, and are driving a new round of port supply chain industry innovation [3]. However, the participants in port supply chains do not cooperate efficiently and effectively, which is the key to reducing costs and improving efficiency. The direct cause of this situation is a failure to share data and other resources effectively among all participants, and the root cause is the lack of trust of the mechanism among all participants in port supply chains. Various parties in port supply

chains cannot achieve trust consistency, and some core suppliers or distributors cannot provide reliable trust endorsements for upstream and downstream.

Blockchain is a peer-to-peer, distributed ledger that is cryptographically-secure, append-only, immutable (extremely hard to change), and updateable only via consensus or agreement among peers [4]. As it is decentralized, tamper-proof, auditable (provenance), transparent and so on, blockchain has great potential to solve the problem of trust among all participants in port supply chains. In practice, blockchain has just begun to be used in the port supply chain, thus, it is still in the internal experimental stage. The customs of Mexico, Peru, and Costa Rica, with the support of the Inter-American Development Bank, initiated the construction of an operator sharing platform based on blockchain technology in March 2018. Therefore, the Tianjin Port blockchain verification pilot project was launched with the aim of protecting data privacy and maximizing the operation efficiency of the port supply chain. The U.S. Congressional Research Institute discussed the application of blockchain technology in the field of customs supervision in June 2019.

We argue that smart contracts are the bridge for application of blockchain technology to port supply chains. A smart contract is a coded contract written by computer language and automatically verified and executed by a computer. It is a digital form of a paper contract. With the help of smart contracts, any party in the entire transaction process can access the exact same data, which is essential to improve process efficiency. Smart contracts can automatically trigger and ensure the smooth flow of files among participants, so specific access rules can be effectively implemented, and new information can be instantly and securely shared with all related parties. Leena and Sultan [5] summarized the latest research and showed how smart contracts can change the method of fund flow tracking, improve liquidity in the real estate field, reduce risks, and have a positive impact on the safe operation of the national economy. Stefania [6] closely follows the current hot sharing economy, integrates smart contracts to systematically improve smart cities and public sectors, manages services such as house leasing, greatly reduces third-party commissions, reduces the risk of fraud, and prevents potential high-end processes. This has a very similar application in the order process of the port supply chain. Whether the smart contracts are correct determines the success of blockchain empowered port supply chains. The current research on the smart contracts of port supply chains has hardly taken into account the probabilistic aspects of port supply chains. The probabilistic aspects are essential for, among others: (1) port supply chains themselves contain the randomized behaviors, e.g., non-determinism, consensus mechanism; (2) unreliable and unpredictable behaviors incurred by execution environment, e.g., message loss, processor failure; (3) performance evaluation by random variables assigned artificially, e.g., reliability, availability [3].

In this article, we propose a verification framework for the smart contracts of port supply chains with probabilistic behaviors. The smart contracts are modeled as DTMCs (Discrete-Time Markov Chains), which are automatically transformed through the BPMN (Business Process Model and Notation) description of smart contracts. The BPMN is the most widely used and simplest tool to describe the business process of port supply chains. It can clearly display the business relationship between each link and department, and visualize the information flow. It can serve as a communication medium between users of port supply chains and developers of smart contracts. DTMC is the formal model for analyzing the user requirement properties that the smart contracts should satisfy; it can be automatically generated from the BPMN. The requirements, such as safety, reliability and reachability, are specified by PCTL (Probabilistic Computation Tree Logic). We use this framework to model the smart contract for the cargo clearance process of the Shanghai Yangshan Port, to abstract the probabilities involved in each contract during the process based on questionnaires from port practitioners, and we use PCTL to specify user requirement properties.

The remainder of this article is structured as follows. Section 2 describes related works about blockchain-powered port supply chain optimization and smart contract verification, and points out their weaknesses. Section 3 presents the scenario of blockchain smart contract reconstructing of the port supply chain, and proposes the verification framework for the probabilistic model checking of smart contracts for port supply chains. Section 4 takes the port customs clearance process as a case study to demonstrate the effectiveness of this framework. Section 5 provides a conclusion.

## 2. Related Works

Smart contracts using blockchain technology have been mentioned by a large number of risk analysis agencies in the field of port supply chains. The credit rating agency MOODY'S summarized the three major steps to automate the international trade process using smart contracts: Firstly, design smart contracts based on paper letters of credit; Secondly, store the form; Finally, use the blockchain to record the ownership of each link [4]. In order to overcome difficulties such as overcapacity, increasingly strict environmental regulations, and security threats faced by the shipping industry, the Danish block shipping company is committed to building a global shared container platform (GSCP) based on blockchain technology to improve shipping efficiency and business. The digital transformation of processes and infrastructure ensures the long-term profitability of the industry. The project's white paper stated that the platform will save the shipping industry at least 5.7 billion USD in the future and will reduce carbon dioxide emissions by 4.6 million tons per year [7]. In the current supply chain mechanism, people use the traction system to track data, however, this method cannot avoid business friction, let alone update data in real time without manual query. Shuchih Ernest Chang et al. [8] proposed a blockchain-based BPR (Business Process Re-engineering) framework, which can help companies re-engineer cross-border business processes and track some key links in the supply chain process. Blockchain smart contract technology has been proven to have a very positive effect on supplier inventory management, the dynamic realization of the digital supply chain, and efficient transportation [9–12]. With the continuous and widespread application of blockchain smart contracts, coupled with the immutability of blockchain technologies, stability and security have increasingly become the focus of attention among researchers. Antonio proposed ESAF (Ethereum Security Analysis Framework), which can be used as a security monitoring tool for the persistence of a set of target contracts [13]. Haya Hasan et al. [12] used IoT-SC to jointly collect data on the chain, taking the vaccine supply chain as an example to test the relationship between smart contracts and entities, interactions between participants, information flow and so on. They proved that this technology can ensure real-time freight tracking performance which can be applied to multi-level and multi-party settings. It has practical applications in the transportation and information traceability of pharmaceutical products and agricultural products [14–20]. Pietro De Giovanni et al. [21] proved through a game theory model that blockchain technology can reduce business risks and transaction costs, which is more suitable for transactions with large-scale fund transfers such as cross-border bulk trade. Arnab Banerjee et al. [22] comprehensively analyzed the advantages of a blockchain-driven supply chain from the perspectives of ERP (Enterprise Resource Planning) transactions, master data management, order-purchasing, demand and supply management, manufacturing, and logistics management. The traditional concept suggests that smart contracts are designed to achieve traceable and irreversible transactions through the use of distributed databases. However, Gunnar Prause and his team [23] believed that their greater potential is reflected in the promotion of entrepreneurial collaborations across organizational business processes held up by smart supply chains.

Due to the immutability of the blockchain, the smart contract must be strictly verified before being put on the chain. Li et al. [24] proposed a formal method of BPMN (Business Process Modeling Notation) based on the extended Petri net model. They used model-driven development technology to design BPMN model elements to the extended Petri

net model elements. Transformation rules and performing the mapping through the ATL model transformation language achieves formalized automatic execution. Tala Najam and Alexander Perucci [25] converted the BPMN2 choreography diagram into a color choreography network mapping for the lack of formal semantics of the BPMN2 specification, but they did not solve the problem that the Petri net model cannot reflect the content of time and does not support the construction of large-scale models. Due to the immutability of the blockchain, any error will become a permanent error once it occurs. In this regard, researchers have designed a large number of methods and tools to verify the correctness of smart contracts, the most representative of which is formal verification. Researchers have designed many formal verification frameworks and analyzed mainstream platforms [26–30]. Mouhamad Almakhour and his team [31] classified the existing smart contract verification tools and introduced a series of analysis tools such as Oyente and MAIAN, and deeply analyzed the cutting-edge technology of smart contract verification from the perspectives of static verification and runtime verification. Devrim Unal et al. [32] put forward the FPM-RBAC model from the perspective of policy regulation. He analyzed smart contracts from the three perspectives of identity management, access control, and compliance checking, and fully considered transaction risks and security requirements in 5G networks. Zhang and Mackey [33,34] proposed several methods for the verification of smart contracts for the online social network and antifraud framework. Nguyen and Li [35,36] used a certificate authentication system and puncturable signatures to detect smart contracts and improve overall fault tolerance. Lennart Ante [37] proposed six mainstream smart contract analysis streams, including smart contract standardization, verification and security, and blockchain and smart contracts for the disruption of existing processes and industries. Prashar et al. [38] used OMNeT++ to verify contract stability. Amritraj Singh et al. [28] proposed DSL (Domain specific languages) to formalize smart contracts on the basis of formal testing and automated verification. Flora has done significant research in the field of BPMN modeling smart contracts. Flora Amato et al. [39] believe that smart contracts must be based on laws to restrict the behavior of participants, thus, they proposed a formal model for verifying the compliance of smart contracts in the IoT environment. Compared with the BPMN, this model pays more attention to the interaction between participants. At the same time, they use the TCTL (timed computational tree logics) formula to describe the attributes. At the same time, Wei Wan [40] considered the relationship between PCTL and DTMC and made a relatively complete theoretical summary and expansion. Flavio Corradini [41] introduced the relationship between the blockchain and the BPMN and designed a simple model-driven automatic blockchain code generator.

In short, there have been attempts to apply smart contracts to the port supply chain. Their successful application will achieve a decentralized, traceable, efficient, and trusted global supply chain system. However, correctness research on the smart contracts of port supply chains has not been involved. Some work has been accumulated on the verification of smart contracts themselves, but this has not considered the random factor in the business process execution of smart contracts. Based on this, we propose a verification framework for the smart contracts of port supply chains with probabilistic behaviors.

### 3. Verification Framework

#### 3.1. Blockchain Empowered Port Supply Chain

The emergence of blockchain technology has brought new possibilities for solving the complicated trust problems of the port supply chain. As shown in Table 1, there are three types of blockchains: public blockchains, private blockchains and consortium blockchains [42]. The public blockchain is completely decentralized and suitable for multi-user scenarios, the private blockchain is completely centralized and suitable for independent organizations, and the consortium blockchain is partially decentralized and suitable for the common scenarios of multiple organizations. The three types of blockchain have

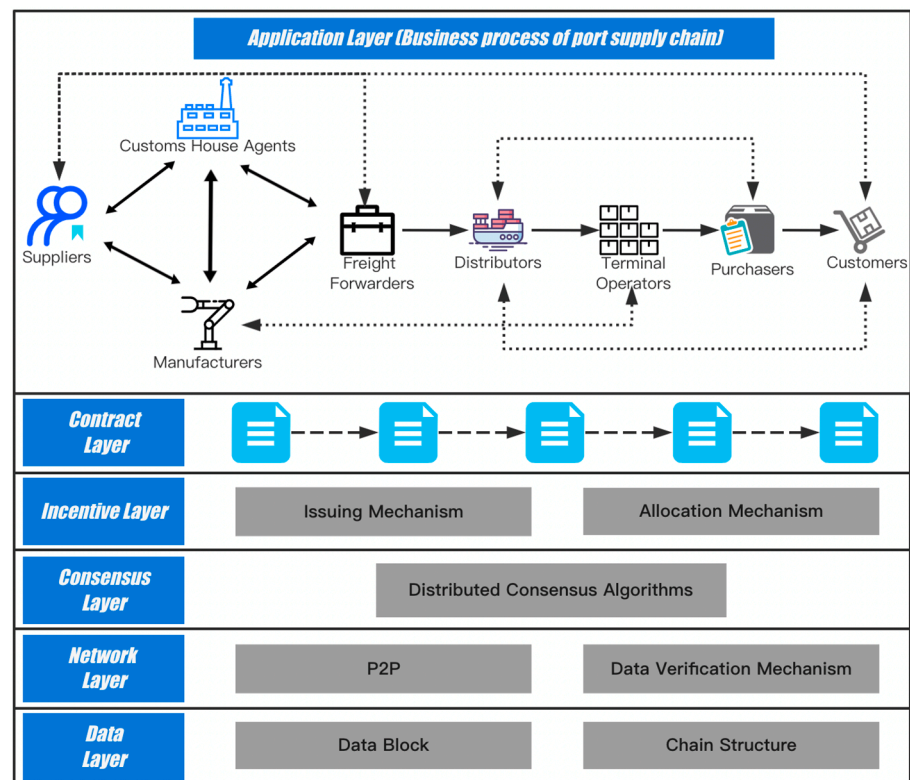
different features in supply chain management. The public blockchain mainly reflects network security, while the private blockchain and the consortium blockchain have the advantage of real-time information transmission. The existing port supply chain involves a large number of departments. There is no unified information verification standard throughout the entire process, and a large number of manual audits are required, while the blockchain smart contract technology has an independent and general consensus mechanism. Users who agree to a certain encryption algorithm will use the smart contract by default. At the same time, the encrypted information verification is extremely fast, which can save a lot of time in the intermediate process.

**Table 1.** Feature comparison among blockchains.

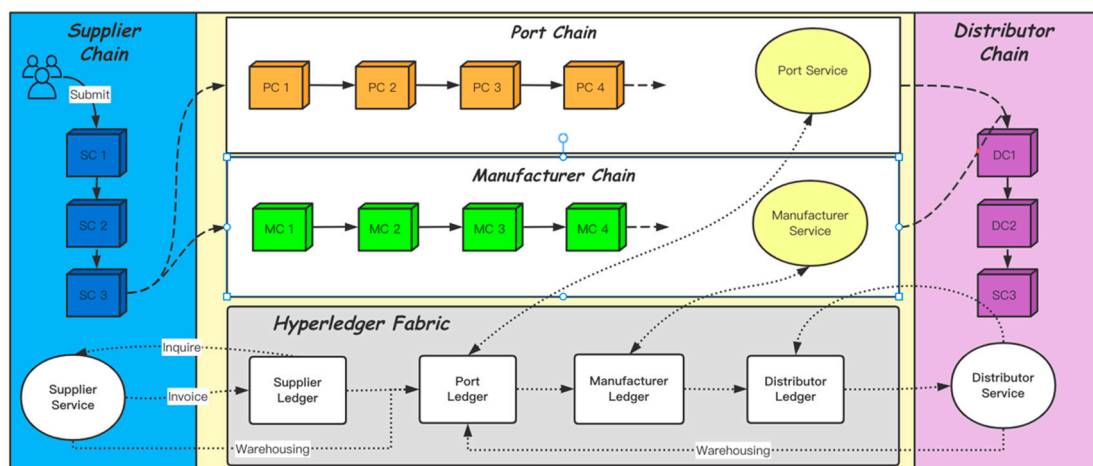
Features	Public Blockchain	Private Blockchain	Consortium Blockchain
Accessibility	Anyone	Central Incharge	Multi-central Incharge
Ledger Keeper	Anyone	Central Incharge	Permissioned identities
Consensus Mechanism	PoW/PoS	Solo/PBFT	Distributed consensus algorithm
Incentive Mechanism	Need	Optional	No
Transaction Speed	Slow	Lighter & Faster	Lighter & Faster
Centralization Degree	Decentralization	Weak Centralization	Strong Centralization
Representative	Bitcoin/Ethereum	Ark Blockchain	R3/Hyperledger Fabric
Programming Language	C++/Solidity	C++/Java	Java/Go
Main features in supply chain management [43]	Transparency/Traceability/Cyber-security	Traceability/Real-time information sharing/Visibility	Traceability/Real-time information sharing/Flexibility

The port supply chain has the following three main characteristics and requirements: (1) medium transaction speed and medium number of transactions, (2) multiple participants, and (3) high transaction information privacy and time-sensitiveness of transportation information. The public chain cannot solve the problem of transaction speed and cannot guarantee the confidentiality of transaction information, while the strong centralization mechanism of the private chain makes it impossible to have multiple participants, which is contrary to the demands of port supply chains. At present, almost all supply chain blockchain platforms use private chains to ensure that information is not leaked; these are currently in the internal test stage. In this article, we chose the consortium blockchain Hyperledger Fabric to reconstruct the port supply chain, as shown in Figure 1. This will be beneficial for large-scale port supply chains. The participants of the port supply chains are authorized to join and participate in data maintenance together. At the same time, the processing speed of 10,000 transactions per second is sufficient to meet the transaction volume requirements of port supply chains.

It can be seen intuitively, from the above figure, that smart contracts are used to implement the business process of port supply chains. Based on the consensus mechanism, once the supply chain information is uploaded to the blockchain, it cannot be changed. Any authorized user can check the chain information at any time. Port administrators can easily check the authenticity of the information by hash verification. We designed the on-chain and off-chain information transfer process of port supply chains as shown in Figure 2. The supplier uploads the purchasing information and transportation information to the supplier chain (SC), and the Hyperledger Fabric automatically integrates it with the port chain (PC) and manufacturer chain (MC), which can also automatically generate a distributor chain (DC) at any time for purchasers to track the progress of the transportation. Due to the characteristics of the Hyperledger Fabric, all authorized participants can read and upload the information on the chain, so multiple sub-chains can be deployed at the same time.



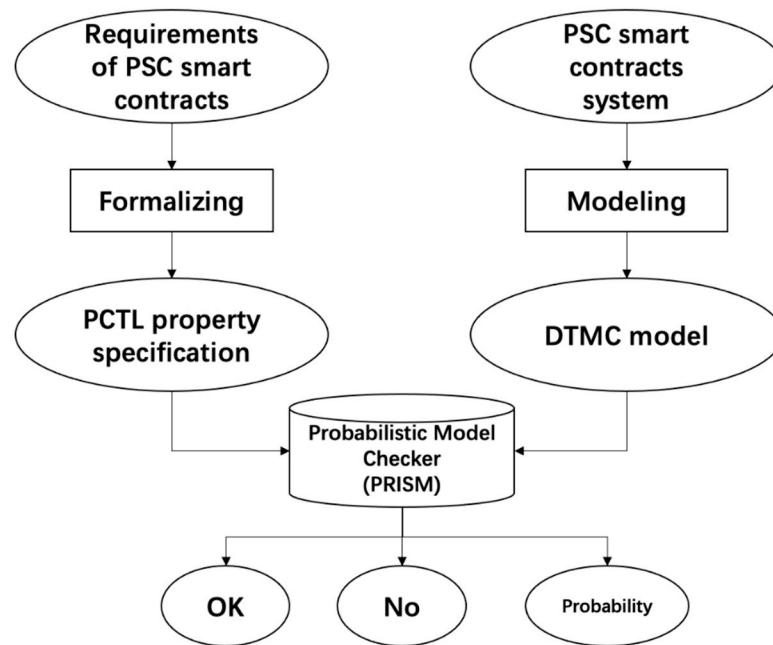
**Figure 1.** The hierarchical structure of a blockchain empowered port supply chain system.



**Figure 2.** On-chain and off-chain information transfer process.

### 3.2. Verification Framework for the Smart Contracts of Port Supply Chains

The overall verification framework of this paper is shown in Figure 3. In the formalizing branch, the requirements of PSC (port supply chain) smart contracts are specified by the PCTL (Probabilistic Computation Tree Logic) formula. In the modeling branch, the PSC smart contracts described in BPMN (Business Process Model and Notation) are modeled as DTMC (Discrete-Time Markov Chains) through a conversion algorithm. The verification process is implemented by the probabilistic model checker PRISM.



**Figure 3.** Verification framework diagram.

### 3.3. Modeling PSC Smart Contracts


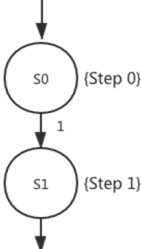
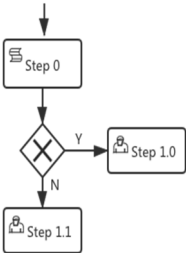
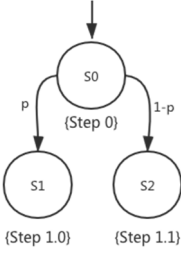
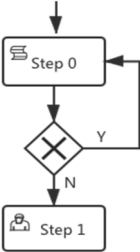
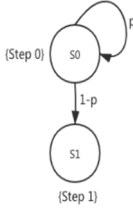
Smart contracts are the programming codes run on blockchain platforms, which are very difficult to understand for the non-professional participants of port supply chains. As a quasi-natural language, BPMN can act as the intermediate language for all users in port supply chains, including business analysts, software developers, and business managers and supervisors. It provides a graphical model for creating a business process that can be used to construct a visual business process diagram of the port supply chain. There are dedicated tools for translating the BPMN into smart contract programming code, such as Caterpillar [44]. We model PSC smart contracts from the BPMN description of smart contracts. Moreover, we model the probabilistic aspects of smart contracts, which are an inherent characteristic of port supply chains. The reasons for this are as follows: (1) port supply chains themselves demonstrate randomized behaviors, e.g., non-determinism, consensus mechanism, (2) unreliable and unpredictable behaviors incurred by execution environment, e.g., message loss, processor failure, and (3) performance evaluation by random variables assigned artificially, e.g., reliability, availability [3]. We use the DMTC modes to model PSC smart contracts.

The DTMC model of a PSC smart contract is defined as a six-tuple,  $SC = (S_{normal}, S_{abnormal}, S_{init}, P, AP, L)$ .  $S_{normal}$ : A set of limited non-empty states, which means that the PSC smart contract is in a normal state space. This set contains the normal status, for example, the supplier sub-contract is normal or the transfer sub-contract is normal. In addition, it includes the initial status and the end status of the port supply chain;  $S_{abnormal}$ : A set of limited non-empty states, which indicates that the PSC smart contract is in an abnormal state space. This set contains all abnormal states, such as the quality inspection contract failed or the goods were lost;  $s_{init} \in S_{normal}$  represents the initial state;  $L: S \rightarrow 2^{AP}$  is a label function used to describe the set of propositions on the state, and identify the sub-smart contracts being executed or abnormal situations corresponding to each state.  $AP$  is the set of atomic propositions.  $S = S_{normal} \vee S_{abnormal}$  is a collection of all state spaces;  $P: S \times S \rightarrow [0, 1]$  is the state transition matrix, indicating whether it will transition from one state to another and the probability of transition.

Each state of the DTMC model corresponds to an event in the BPMN diagram. The transitions between states correspond to the migration of events. The sequence flows and

message flows in the BPMN description represent the completion of the port supply chain sub-process of the previous department and the procedures of the next department. The corresponding mapping rules are shown in Table 2. The left side of the table is the sub-process of the BPMN components, and the right side is the corresponding sub-process of the DTMC model. We automate this process by Algorithm 1.

**Table 2.** Mapping PSC smart contracts in BPMN to DTMC.

Sequential event conversion	
	
Conditional branch event conversion	
	
Cyclic event conversion	
	

Algorithm 1 presents a conversion algorithm which can convert the PSC smart contracts in BPMN description into a DTMC model. The premise of the conversion algorithm is to accurately divide the complete BPMN description into sub-models and then map them. It is more of a flowchart level mapping than a semantic mapping.



**Algorithm 1:** DTMC Model Mapping**Input:** BPMN =  $(E, A^M, A^S, A^B, G^D, G^P)$ **Output:** DTMC[s]

```

1   $s \leftarrow 0$ 
2  STATES.PUSH( $E_0$ )
3  while STATES  $\neq \emptyset$  do
4     $m \leftarrow$  STATES.POP()
5    Array[]  $\leftarrow m$ 
6    if ( $m \in S \cup A^M \cup A^S \cup A^B$ ) then
7      for all the  $n \in \text{LATER}(m)$  do
8        Array[]  $\leftarrow$  Array[]  $\cup (m, n)$ 
9        Emit Array[]
10     DTMC[s]  $\leftarrow$  Array[]
11     Prob[]  $\leftarrow p$ 
12      $s \leftarrow s + 1$ 
13     if ( $n \notin \text{BPMN}[]$ ) then
14       Array[].PUSH( $n$ )
15  SUBPROCESS  $\leftarrow \emptyset$ 
16  if ( $x \in G^D$ ) then
17    STATES.PUSH( $G^D$ )
18    Search for  $G^P$ 
19    STATES.PUSH( $G^P$ )
20     $x \leftarrow$  STATES.POP()
21     $y \leftarrow$  STATES.POP()
22    Array[]  $\leftarrow (x, y)$ 
23    Emit Array[]
24    DTMC[s]  $\leftarrow$  Array[]
25     $x \leftarrow x + 1$ 
26     $y \leftarrow y + 1$ 
27     $s \leftarrow s + 1$ 
28  END SUBPROCESS

```

The algorithm takes the events, activities and gateways of the BPMN as an input, and performs the following operations: first, the counter  $s$  that records the number of nesting times is set to zero, the initial event  $E_0$  is pushed into the stack STATES, and then the following loop is started. The purpose is to add each node of the BPMN to Array[]. Pop the event  $m$  from the stack to Array, and make the following judgment: if  $m$  belongs to  $\{S$  (Original Event),  $A^M$  (Manual Activity),  $A^S$  (Script Activity),  $A^B$  (Business Activity)}, push all the subsequent nodes  $n \in \text{LATER}(m)$  that meet the conditions into Array[], and emit them into the DTMC state variable. If the node  $n$  does not belong to events, activities, and two restricted gateways (decision gateway and Parallel gateway), the node will pop out. However, for Decision Gateway  $G^D$  and Parallel Gateway  $G^P$ , it needs to be converted in the order of  $G^P$  first and  $G^D$  later, which is placed in SUBPROCESS here.

**Theorem 1.** For a BPMN process, if it can be transformed into a DTMC, the process must be physically reachable.

**Proof.** Let the highest layer of the DTMC be  $k$ , the next higher layers are  $k-1, k-2, \dots$ , and the lowest layer is 0. There is only a sequential structure at the  $k$  layer. For a BPMN with only a sequential structure, all states are physically reachable, so the subnet is physically reachable. At the  $k-1$  layer, all  $k$  layer subnets are represented by nodes such as events and gateways, and the  $k-1$  layer has only a sequential structure. Therefore, the  $k-1$  layer

including the  $k$  layer is also physically reachable. By analogy, all states of layer 0 are physically reachable, so that the DTMC model is physically reachable.  $\square$

Algorithm 1 has an upper complexity bound of  $O(n^2)$ , where  $n$  is the number of nodes in the BPMN process. The upper limit corresponds to the cumbersome degree of the supply chain process. In the most common case, only the decision gateway appears in the BPMN process, and there are no different types of gateway nesting.

### 3.4. Properties Specification

We use PCTL to specify a set of user requirements. PCTL is defined by the following syntax:

$$\Phi ::= \text{true} | a | \Phi \wedge \Phi | \neg \Phi | P_{\bowtie p}(\Psi) \quad (1)$$

$$\Psi ::= X\Phi | \Phi U^{\leq t} \Phi \quad (2)$$

where  $p \in [0,1]$ ,  $\bowtie \in [<, \leq, >, \geq]$ ,  $t \in R_{\geq 0}$ ,  $a$  is the atomic proposition. The tense operators  $X$  and  $U$  are called Next and Until, which is the same as the temporal logic. The formulas produced by  $\Phi$  are called state formulas, and their true value can be judged by the true value of each state included. The formulas generated by  $\Psi$  are called path formulas, and their true value needs to be evaluated by calculating each execution path.

In most cases,  $\Phi$  only describes the atomic proposition corresponding to the absorption state. When it describes the failure state, the probability constraint is expressed as  $\leq x$ , where  $x$  is the upper bound of the failure probability; when it describes the success state, the probability constraint is expressed as  $\geq x$ , where  $x$  is the lower bound of the probability of success.

As shown in Table 3, we set up the following DTMC states to model the status of PSC smart contracts, respectively, to make the property specification more universal.  $Current_{sup}$  represents the extent of the supplier link among them. Since there is more than one supplier, goods circulation and fund exchange are between suppliers, and they also bring out the document information, thus, it is necessary to split this link in order to present the smart contract of each exchange process. There is a multimodal transportation situation in the transportation part.  $Current_{trans}$  is used to present the sub-contracts generated by different transportation means or different transportation stages of the same transportation means, and it is used to track the steps of the transportation link. There will be multiple buyers or distributors in the procurement process, and  $Current_{pur}$  is used to track the progress of the goods flowing among the buyers.  $End_{sup}$ ,  $End_{trans}$ ,  $End_{pur}$  are Boolean variables used to monitor whether the supplier link, transportation link, and purchase link are actually completed or not. The completion is 1, and the error is 0. Finally, we use  $Current_{PSC}$  and  $End_{PSC}$  to monitor the progress of the entire port supply chain ( $m, n, u \leq v$ ).

**Table 3.** PSC smart contracts states and ranges.

State	Ranges
$Current_{sup}$	$\{0,1,2,\dots,m\}$
$End_{sup}$	$\{0,1\}$
$Current_{trans}$	$\{0,1,2,\dots,n\}$
$End_{trans}$	$\{0,1\}$
$Current_{pur}$	$\{0,1,2,\dots,u\}$
$End_{pur}$	$\{0,1\}$
$Current_{PSC}$	$\{0,1,2,\dots,v\}$
$End_{PSC}$	$\{0,1\}$

On the basis of the above states' definition, we specify requirement and regulation properties for the PSC smart contracts.

**Property 1:** What is the probability that a certain batch of goods will complete the entire port supply chain smart contract?

$$P = ? [F(Current_{PSC} = v) \& (End_{PSC} = 1)] \quad (3)$$

Since it is impossible to guarantee the smooth completion of each branch chain in the actual process, the probability is usually less than 1. Under the premise that all links are not required to be perfectly realized, we can make the probability interval in [0.95,1] fuzzy to 1, so that people can compare the actual value with the expected value.

For the three links, there can be more detailed specifications as follows:

**Property 2:** What is the probability that the supplier link smart contracts are completed but the entire process of the port supply chain is not completed?

$$P = ? [F(End_{Sup} = 1) \& (End_{PSC} = 0)] \quad (4)$$

**Property 3:** What is the probability that the transportation link smart contracts are completed but the entire process is not completed?

$$P = ? [F(End_{Trans} = 1) \& (End_{PSC} = 0)] \quad (5)$$

**Property 4:** What is the probability that the procurement process of the smart contracts are completed but the entire process is not completed?

$$P = ? [F(End_{Pur} = 1) \& (End_{PSC} = 0)] \quad (6)$$

The above three properties allow the port supply chain supervision agency to track the completion of a certain batch of goods in a timely manner, but it is impossible to know where the problem lies when the probability drops. Therefore, we propose the following three more detailed specifications:

**Property 5:** What is the probability that the supplier link progresses to the step  $x(x < m)$  but the total status of the supplier link smart contracts are completed and the whole process status is also completed?

$$P = ? [F(Current_{Sup} = x) \& (End_{Sup} = 1) \& (End_{PSC} = 0)] \quad (7)$$

**Property 6:** What is the probability that the transportation link progresses to the step  $y(y < n)$  but the total status of the transportation link smart contracts are completed and the whole process status is also completed?

$$P = ? [F(End_{Sup} = 1) \& (Current_{Trans} = y) \& (End_{Trans} = 1) \& (End_{PSC} = 0)] \quad (8)$$

**Property 7:** What is the probability that the procurement process progresses to the step  $z(z < u)$  but the overall status of the procurement process smart contracts are completed and the whole process status is also completed?

$$P = ? [F(End_{Trans} = 1) \& (Current_{Pur} = z) \& (End_{Pur} = 1) \& (End_{PSC} = 0)] \quad (9)$$

#### 3.4.1. Accuracy

The accuracy of the migration rate must be ensured before model checking. Due to the different levels of manual participation in each link of the port supply chain, we define the manual participation time between all levels of the suppliers as  $\lambda_1 \sim \lambda_M$ , and the manual participation time between all levels of the purchasers as  $\mu_1 \sim \mu_N$ , where necessary participation time is defined as  $\lambda_m$  and  $\mu_n$ , respectively, and the accuracy is analyzed through the rewards structure in PRISM.

Randomly generate  $\frac{1}{\lambda}$  and  $\frac{1}{\mu}$  under different node numbers, and the accuracy of the entire model is shown in Table 4. Obviously, when the number of sub-contracts is too

large, it will greatly affect the completion of the entire process and reduce the operating efficiency of the main contract.

**Table 4.** Accuracy under different N.

N	Property		
	Nodes	Time Per Iter: (s)	Probability (False)
2	14	<0.00001	0.072
4	30	<0.00001	0.081
8	57	0.00001	0.113
16	105	0.00001	0.141
32	197	0.00002	0.179
64	377	0.00001	0.239
128	729	0.00002	0.285
256	1437	0.00012	0.373
512	2849	0.00004	0.490

### 3.4.2. Reachability

The essence of reachability analysis is to check whether each smart contract is effectively invoked. Each sub-contract of the port supply chain running on the blockchain platform must be accurately invoked and effectively executed to ensure the smooth progress of the entire process. However, in reality, it is difficult to ensure that the hash values of each sub-contract are consistent. We created a module for each contract model. If the contract verification fails, it will be marked as affected, otherwise it will be marked as correct. We designed an algorithm to verify reachability which is shown in Algorithm 2. We can initially obtain whether the status of each sub-process is reachable and whether the status result is true. If the total process is true, then further calculations can be made.

---

#### Algorithm 2: Reachability algorithm

---

**Input:** the property formula to be verified  $F$

**Output:** True, False

1. Preprocess  $F$ , if the formula is false, return **False**
  2. Select the next state variable that has no value
  3. Deduction
  4. The derived formula = = true, return **True**
  5. Conflict, then
  6. Analyze the conflict and go back
  7. Cannot go back, return **False**
  8. No conflict is deduced, return to Step 2
- 

Taking three supply terminal processes as an example, the PRISM module code is as follows in Algorithm 3:

---

#### Algorithm 3: PRISM module code of three supply terminal processes

---

**module** *Sup\_1*

Sup1\_affected: **bool** init false;

[Sup1\_affected] = **false** &  $M \geq m \rightarrow (\text{Sup1\_affected}' = \text{true});$

[Sup2\_correct] Sup2\_correct\_SupplierSC = **true** & Sup1\_affected = **true**  $\rightarrow 1:\text{true};$

[Sup3\_correct] Sup3\_correct\_SupplierSC = **true** & Sup1\_affected = **true**  $\rightarrow 1:\text{true};$

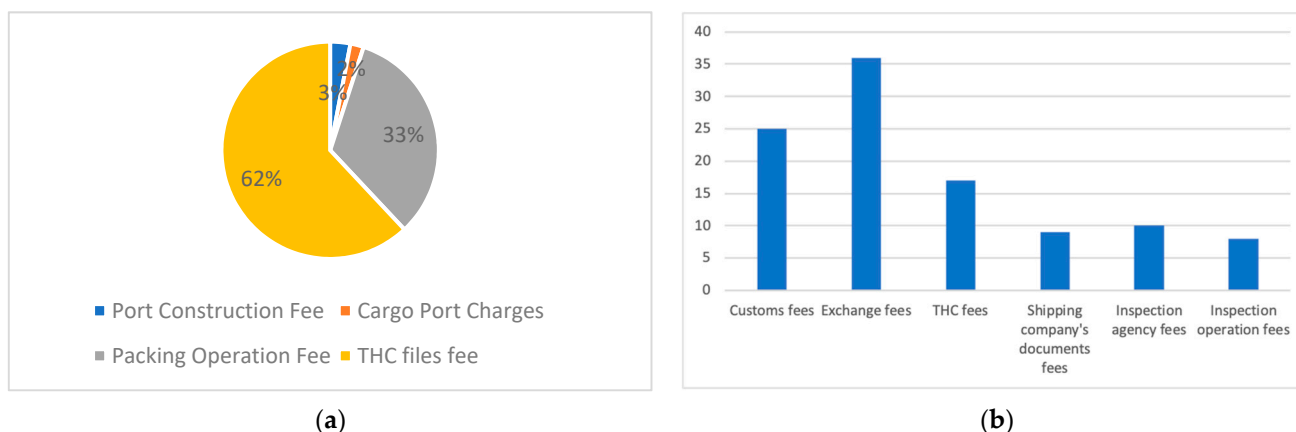
**endmodule**

---

#### 4. Case Study

We have selected the port customs clearance process of port supply chains as a case study. Import and export customs procedures refer to the activities of going through customs procedures for inbound and outbound goods that are subject to different customs systems in accordance with the law.

The way that customs clearance has changed from the traditional ‘territorial declaration-port inspection and release’ to the current ‘integrated customs clearance’ is that various obstacles between departments and regions have basically been cleared. However, there are still many cumbersome processes that have not been eliminated. Figure 4a shows that according to the 2020 Shanghai Water Transport Port Container Freight Cross-border Trade Expenses and Compliance Cost Assessment Report, the shipping company’s document fee reached JPY 1071, which is an amazing proportion. Figure 4b shows that in the questionnaires for import and export companies (55 import responses and 52 export responses), the replies with subjective experience higher than the average mainly focused on customs declaration fees, order replacement fees, and terminal THC (Terminal Handling Charge) fees. It can be seen that there is still a lot of room for optimization in the port supply chain customs clearance process. For the convenience of research, we model the macroscopic customs clearance process smart contracts. The macro-level customs clearance process includes multiple suppliers, purchasers and transportation links. In order to promote the linkage between ports and shipping enterprises, speed up the application and information sharing of digital platforms, and promote the electronic release of imported containers based on blockchain, we have tentatively conducted pilot projects in the Shanghai Yangshan port of China. We cooperated with the Shanghai Yangshan Port to upload and model its customs clearance process, reducing the average processing time of major imported e-commerce cargo documents from 2 days to less than 4 h.



**Figure 4.** Disadvantages in the customs clearance process (a) The proportion of various expenses in customs clearance expenses; (b) Customs clearance fees and exchange fees account for a high proportion.

Appendix A is a simplified version of the on-site customs declaration manual for a certain customs’ rapid customs clearance. We can see that for import and export companies, they have to fill in a large number of forms and provide a large number of letters each time they import customs clearance and export customs clearance, which is the root cause affecting the efficiency of customs clearance.

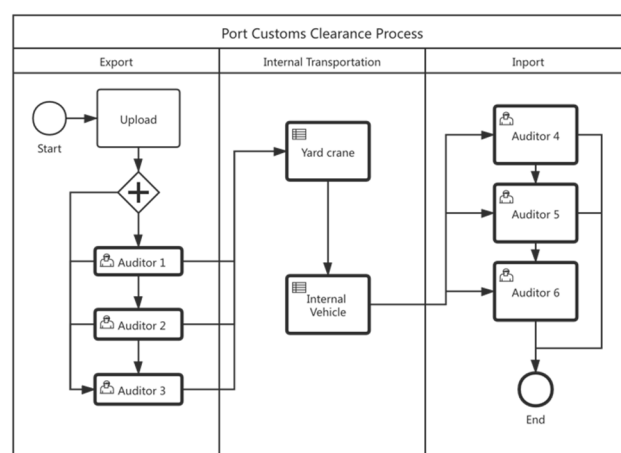
##### 4.1. Modeling Smart Contracts of the Port Customs Clearance Process

Suppliers need to fill in a large amount of contract information during the export stage of goods, such as manifest contracts, inspection and quarantine contracts, and tax payment contracts mentioned in Appendix A. The customs clearance process is an important section of the port supply chain, in which all the import and export goods must

go through various procedures of entry inspection when they arrive in another country. It plays a critical role in order to ensure the safety of the goods and enable the owner to pick them up smoothly. However, each port department has different requirements for verification of contract internal information. For example, the customs department needs to check the goods required to pay customs duties and tax deductions according to law, while reviewing the value and exchange rate of the goods, the customs declaration department needs to proofread the delivery, loading and transportation information in the manifest, and the quarantine department needs to check whether the quarantine information of the cargo is effective. All departments must upload the results to the smart contract after completing the information verification. The validation of each document is time-consuming. The whole process takes 2–3 days, however, it only takes a few hours to validate with a smart contract, instead of manual validation.

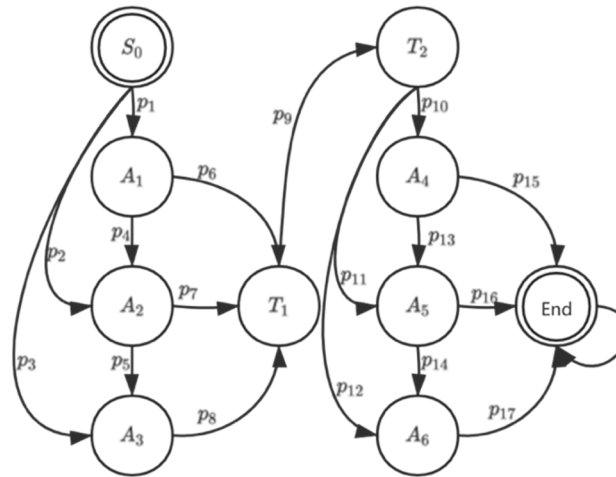
There are many reasons for difficulties and these can be divided into two categories: external and internal. External reasons may include delays caused by weather conditions, lower-than-expected quality of raw materials, delays caused by the dispatch of containers in and out of the port, and supplier or purchaser's breach of contract. Internal reasons may be caused by improper filling of contract information. In the smart contracts of various links in the port supply chain, there is information, such as Port Loading, Port Discharging, Cargo Description, Consignee's name and address, which is greatly affected by the filling specifications, and there is a considerable probability that the hash verification will not pass. This article only discusses external reasons.

Since a certain container or a certain ship of goods corresponds to a large number of suppliers, the failure of any supplier's sub-contract will have an impact on the entire supplier contract, so the relationship between each supplier's sub-contract is 'And'. There is a calling relationship between contracts. When different types of suppliers upload information and call contracts, there is a certain probability that they will default. For each supplier, when completing the various processes in Appendix A, such as filling in the manifest smart contract, completing the export customs declaration smart contract, or applying for the customs clearance inspection and quarantine smart contract, there will be a certain probability that they will not pass. At the same time, purchasers also face the same problems when applying for an import customs clearance smart contract and completing tariff contracts. In this article, we consider such a situation: for a supplier's cross-border transportation smart contract, three different departments are required to review at the export terminal which are independent of each other. During the cargo transportation stage, only the mode of transportation within the port is considered. When goods arrive at the port for the import process, three different departments are also required. Every department needs an auditor to proofread the information. The BPMN description of smart contracts from the customs clearance process is shown in Figure 5.



**Figure 5.** The BPMN description of smart contracts from the port customs clearance process.

Smart contracts in BPMN description can be converted into a DTMC model through Algorithm 1 and shown in Figure 6. Three auditors correspond to the customs declaration department, the taxation department, and the inspection and quarantine department, respectively. Some goods, such as bulk goods, do not need to be inspected by the quarantine department, while duty-free goods do not need to be reviewed by the tax department. The symbol  $p$  represents the transition probability between two departments.



**Figure 6.** DTMC model of smart contracts from the customs clearance process.

#### 4.2. Verification Results and Analysis

Let  $m = n = u = 3$ ,  $v = 8$ , we can redefine the property specification in Section 3.4 as follows:

**Property 1:** What is the probability that all contracts are failed?

$$P = ? [F(Current_{PSC} = 8) \& (End_{PSC} = 0)] \quad (10)$$

**Property 2–Property 4:** use a similar description.

**Property 5:** What is the probability that the export contract 2 is successfully passed and the entire customs clearance process contract is failed?

$$P = ? [F(Current_{ex} = 2) \& (End_{ex} = 1) \& (End_{PSC} = 0)] \quad (11)$$

**Property 6:** What is the probability that the transition contract 2 is successfully passed and the entire customs clearance process contract is failed?

$$P = ? [F(End_{ex} = 1) \& (Current_{Trans} = 2) \& (End_{Trans} = 1) \& (End_{PSC} = 0)] \quad (12)$$

**Property 7:** What is the probability that the import contract 2 is successfully passed and the entire customs clearance process contract is failed?

$$P = ? [F(End_{Trans} = 1) \& (Current_{ex} = 2) \& (End_{ex} = 1) \& (End_{PSC} = 0)] \quad (13)$$

In order to make the code more universal, we present part of the PRISM code in a modularized style in Figure 7. The user can freely define the number of states and the probability of state transitions according to the actual situation. We count them by N and Max. The characteristic  $x12$  represents the migration from state one to state two,  $p1$  represents the migration probability, and meanwhile the success or failure is represented by a Boolean variable.

```

1 dtmc
2 const double p1;
3 const double p2;
4 const double p3;
5 const Max;
6 //further probabilities if needed
7 module N
8 N: [0..Max];
9 [x12]N=0->(N'=N+1);
10 [x13]N=0->(N'=N+1);
11 [x14]N=0->(N'=N+1);
12 [x23]N=0->(N'=N+1);
13 [x24]N=0->(N'=N+1);
14 //further transitions if needed
15 endmodule
16 module M12
17 x12:bool;
18 [x12]true->p1:(x12'=true)+(1-p1):(x12'=false);
19 endmodule
20 module M13
21 x13:bool;
22 [x13]true->p2:(x13'=true)+(1-p2):(x13'=false);
23 endmodule
24 //further module if needed

```

Figure 7. PRISM Code of the DTMC model of the customs clearance process.

According to the questionnaire survey of port practitioners in the Shanghai Yangshan Port, we selected the actual passing probability of three main links and preset the success probability of the smart contract for the export part and the import part as 80%, 90% and 95%. By default, there are no accidents in the smart contract for internal port transportation. We took property one in the first set of preset values as an example for verification, and set the probability index in reverse to get the calculation result shown in Figure 8. The rest of the verification results of the three sets of preset values are shown in Table 5.

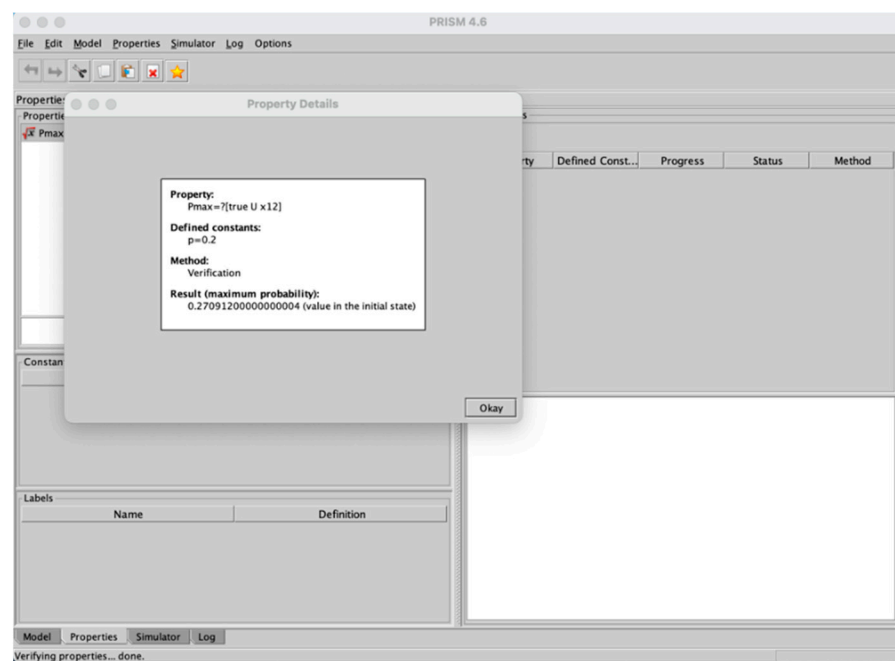


Figure 8. PRISM calculation result of Property 1 in Set 1.

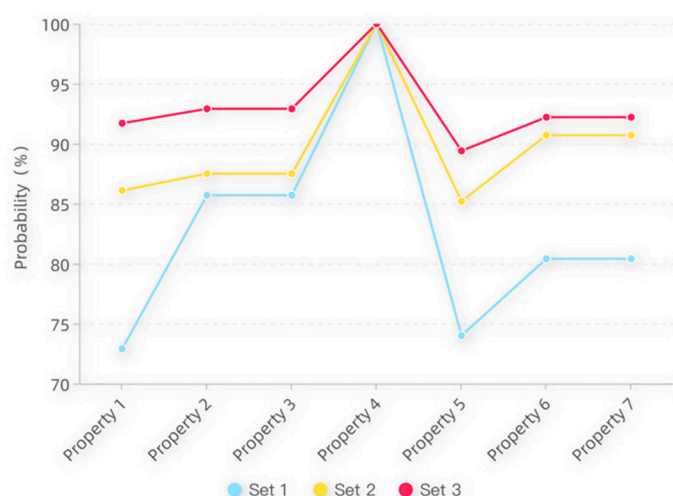
Table 5. PRISM calculation results of each property.

	Set 1	Set 2	Set 3
<b>Property 1</b>	27.1%	13.9%	8.3%
<b>Property 2</b>	14.3%	12.5%	7.1%
<b>Property 3</b>	14.3%	12.5%	7.1%
<b>Property 4</b>	100%	100%	100%



<b>Property 5</b>	26%	14.8%	10.6%
<b>Property 6</b>	19.6%	9.3%	7.8%
<b>Property 7</b>	19.6%	9.3%	7.8%

The status of smart contracts in the entire port supply chain can be analyzed with different probabilities, and fault contracts can be located under different preset values to achieve the purpose of supervision. Through the longitudinal analysis of different preset value groups, the influence of different parts in the port supply chain on the success rate can be judged separately in Figure 9. International logistics and port practitioners informed that the pass rate of each link must reach 90–95% to ensure the continuous and efficient operation of the entire port customs clearance business. Thus, we set the expected probability to 90%.



**Figure 9.** Failure analysis under expectation probability.

Since we assume that the transportation contract is not damaged, the results of **Property 2** and **Property 3** are the same. There is no obvious difference between the impact of export contracts and import contracts on the overall success rate, which shows that what really affects the port supply chain smart contract is the failure probability of the local contract. If the value of **Property 5~7** is greater than the complement set of **Property 2~4**, the damage contract can be located as export-department contract 3 and import-department contract 3. If the probability of damage to the internal transport contract is considered, we can use the same method to compare **Property 3** and **Property 6**. In this case, for the export section, only the probability of the third set of preset values is higher than the expected probability. At the import section, the pass rate of contract 5 in the three sets of preset values is lower than 90%, and only contracts 6 and 7 in the second and third set of preset values are higher than expected. Therefore, this method can quickly locate unqualified smart contracts.

Through comparing the values of the same property under different preset probabilities horizontally, it can be seen that the smaller the number of state transitions, which means the smaller the number of sub-contracts, the higher the success rate of the entire port supply chain contract. At the same time, the export contracts have a more important impact on the overall success rate than the import contracts, which requires contract programmers and suppliers to pay more attention to accuracy in the process of information uploading and contract construction.

## 5. Conclusions

In this article, we present an architecture of consortium blockchain empowered port supply chains, and propose a framework for the smart contracts of port supply chains with probabilistic behaviors. The consortium blockchain is a compromise between decentralization and efficiency. The public blockchain takes too long time to validate transactions, while the private blockchain has a high degree of centralization which is contrary to the original intention of a blockchain. The consortium blockchain combines the advantages of both. The smart contracts are modeled as DTMCs, which are automatically transformed through the BPMN description of smart contracts. The requirements, are specified by PCTL. A case study of the port customs clearance process of port supply chains is used to demonstrate the effectiveness of this framework. The limitation of this work is that this framework only considers the probability aspect of the port supply chain, and the time variable, reward mechanism, etc., are not included, which also affects the integrity of the smart contract for port supply chains. In the future, we will exploit MDP (Markov decision process) to the model smart contracts of port supply chains, which can include non-deterministic and probabilistic behaviors simultaneously. Alternatively, we will use CTMDP (continuous-time Markov decision process) to model smart contracts which involve continuous-time behaviors, and we will consider the more complex user requirement properties, such as security, privacy and liveness, which can be specified by PCTL\* (super set of PCTL).

**Author Contributions:** Conceptualization, Y.L. and Z.Z.; methodology, Y.L. and Y.M.; software, Z.Z.; validation, Y.L., Z.Z., and Y.M.; formal analysis, Y.L.; investigation, Y.L.; resources, Z.Z.; data curation, Z.Z.; writing—original draft preparation, Y.L. and Z.Z.; writing—review and editing, Y.L.; visualization, Z.Z.; supervision, Y.Y.; project administration, Y.Y.; funding acquisition, Y.L. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by MOE Humanities and Social Sciences Foundation of China under Grant Nos. 20YJCZH102.

**Institutional Review Board Statement:** Not applicable

**Informed Consent Statement:** Not applicable

**Data Availability Statement:** Not applicable

**Acknowledgments:** Thanks to Marta Kwiatkowska at The University of Oxford, UK, she has inspired us through her books and papers, and particularly through having a direct discussion with her.

**Conflicts of Interest:** The authors declare no conflict of interest.

## Appendix A. Port Customs Clearance Declaration Procedures

	Import		Export
Manifest transmission and customs declaration entry declaration	Transmission of manifest electronic data to customs through the platform	Customs declaration entry and declaration link	Entry and declaration
	Entry of customs declaration		
	Customs declaration		
	Customs declaration chargeback		Customs declaration chargeback
Customs review and on-site presentation	H2000 manual review	Customs review	H2000 manual review
	EDI, POP query		EDI, POP query
	On-site delivery, release delivery, post-delivery (paperless customs clearance)		On-site delivery, release delivery, post-delivery (paperless customs clearance)
	Pay taxes		Pay taxes

Inspection and release	The logistics monitoring department handles the second confirmation of the manifest	Inspection and release	The terminal supervision department handles inspection and release procedures
	Application for clearance procedures at the clearance post of the Customs Clearance Section		The enterprise ships the ship with the “Export Goods Shipment List”
	The terminal supervision department handles inspection and release procedures		Logistics Monitoring Section handles ship export customs clearance procedures
Customs clearance and issuance certificate		Customs clearance and issuance certificate	

## References

- Han, C. Assessing the impacts of port supply chain integration on port performance. *Asian J. Shipp. Logist.* **2018**, *34*, 129–135.
- Botti, A.; Monda, A.; Pellicano, M.; Torre, C. The Re-Conceptualization of the Port Supply Chain as a Smart Port Service System: The Case of the Port of Salerno. *Systems* **2017**, *5*, 35.
- Dotolia, M.; Fantia, M.P.; Manginia, A.M.; Steccob, G.; Ukovich, W. The impact of ICT on intermodal transportation systems: A modelling approach by Petri nets. *Control. Eng. Pract.* **2010**, *18*, 893–903.
- Jafari-Sadeghi, V.; Garcia-Perez, A.; Candel, E.; Couturier, J. Exploring the impact of digital transformation on technology entrepreneurship and technological market expansion: The role of technology readiness, exploration and exploitation. *J. Bus. Res.* **2021**, *12*, 100–111.
- Alotaibi, L.S.; Alshamrani, S.S. Smart Contract: Security and Privacy. *Comput. Syst. Sci. Eng.* **2021**, *38*, 93–101.
- Fiorentino, S.; Bartolucci, S. Blockchain-based smart contracts as new governance tools for the sharing economy. *Cities* **2021**, *117*, 103325.
- Tsiulin, S.; Reinau, K.H.; Hilmola, O.-P.; Goryaev, N.; Karam, A. Blockchain-based applications in shipping and port management: A literature review towards defining key conceptual frameworks. *Rev. Int. Bus. Strategy* **2020**, *2*, 201–224.
- Chang, S.E.; Chen, Y.C.; Lu, M.F. Supply chain re-engineering using blockchain technology: A case of smart contract based tracking process. *Technol. Forecast. Soc. Change* **2019**, *144*, 1–11.
- Rogerson, M.; Parry, G.C. Blockchain: Case studies in food supply chain visibility. *Supply Chain. Manag. Int. J.* **2020**, *25*, 601–614.
- De Giovanni, P. Digital supply chain through dynamic inventory and smart contracts. *Mathematics* **2019**, *7*, 1235. <https://doi.org/10.3390/math7121235>.
- Dolgui, A.; Ivanov, D.; Potryasaev, S.; Sokolov, B.; Ivanova, M.; Werner, F. Blockchain-oriented dynamic modelling of smart contract design and execution in the supply chain. *Int. J. Prod. Res.* **2020**, *58*, 2184–2199.
- Hasan, H.; AlHadhrani, E.; AlDhaheer, A.; Salaha, K.; Jayaraman, R. Smart contract-based approach for efficient shipment management. *Comput. Ind. Eng.* **2019**, *136*, 149–159.
- Vivar, A.L.; Orozco, A.L.S.; Villalba, L.J.G. A security framework for Ethereum smart contracts. *Comput. Commun.* **2021**, *172*, 119–129.
- Pranto, T.H.; Noman, A.A.; Mahmud, A.; Bahalul Haque, A.K.M. Blockchain and smart contract for IoT enabled smart agriculture. *PeerJ Comput. Sci.* **2021**, *7*, e407.
- Omar, I.A.; Jayaraman, R.; Debe, M.S.; Salah, K.; Yaqoob, I.; Omar, M. Automating procurement contracts in the healthcare supply chain using blockchain smart contracts. *IEEE Access* **2021**, *9*, 37397–37409.
- Omar, I.A.; Hasan, H.R.; Jayaraman, R.; Salah, K.; Omar, M. Implementing decentralized auctions using blockchain smart contracts. *Technol. Forecast. Soc. Change* **2021**, *168*, 120786.
- Ahmed, M.; Taconet, C.; Ould, M.; Chabridon, S.; Bouzeghoub, A. IoT Data Qualification for a Logistic Chain Traceability Smart Contract. *Sensors* **2021**, *21*, 2239.
- Yoo, M.; Won, Y. A study on the transparent price tracing system in supply chain management based on blockchain. *Sustainability* **2018**, *10*, 4037.
- Xu, Z.; Zhang, J.; Song, Z.; Liu, Y.; Li, J.; Zhou, J. A scheme for intelligent blockchain-based manufacturing industry supply chain management. *Computing* **2021**, *103*, 1771–1790.
- Liu, X.; Muhammad, K.; Lloret, J.; Chen, Y.-W.; Yuan, S.-M. Elastic and cost-effective data carrier architecture for smart contract in blockchain. *Future Gener. Comput. Syst.* **2019**, *100*, 590–599.
- De Giovanni, P. Blockchain and smart contracts in supply chain management: A game theoretic model. *Int. J. Prod. Econ.* **2020**, *228*, 107855.
- Banerjee, A. Blockchain technology: Supply chain insights from ERP. *Adv. Comput.* **2018**, *111*, 69–98.
- Prause, G. Smart contracts for smart supply chains. *IFAC-PapersOnLine* **2019**, *52*, 2501–2506.
- Li, Z.; Zhou, X.; Wu, K.; Chen, F.-B. Formalization of BPMN based on extended Petri net model. *Comput. Sci.* **2016**, *43*, 40–48.
- Najem, T.; Perucci, A. Mapping BPMN2 Service Choreographies to Colored Petri Nets. In *Proceedings of the International Conference on Software Engineering and Formal Methods, Oslo, Norway, 16–20 September 2019*; Springer: Cham, Switzerland, 2019; pp. 85–100.
- Sun, T.; Yu, W. A formal verification framework for security issues of blockchain smart contracts. *Electronics* **2020**, *9*, 255.

27. Hang, L.; Kim, D.H. Reliable task management based on a smart contract for runtime verification of sensing and actuating tasks in IoT environments. *Sensors* **2020**, *20*, 1207.
28. Singh, A.; Parizi, R.M.; Zhang, Q.; Raymond, C.K.K.; Dehghantanh, A. Blockchain smart contracts formalization: Approaches and challenges to address vulnerabilities. *Comput. Secur.* **2020**, *88*, 101654.
29. Huh, J.H.; Kim, S.K. Verification plan using neural algorithm blockchain smart contract for secure P2P real estate transactions. *Electronics* **2020**, *9*, 1052.
30. Osterland, T.; Rose, T. Model checking smart contracts for ethereum. *Pervasive Mob. Comput.* **2020**, *63*, 101129.
31. Almakhour, M.; Sliman, L.; Samhat, A.E.; Mellouk, A. Verification of smart contracts: A survey. *Pervasive Mob. Comput.* **2020**, *67*, 101227.
32. Unal, D.; Hammoudeh, M.; Kiraz, M.S. Policy specification and verification for blockchain and smart contracts in 5G networks. *ICT Express* **2020**, *6*, 43–47.
33. Zhang, S.; Yao, T.; Arthur Sandor, V.K.; Weng, T.H.; Liang, W.; Su, J. A novel blockchain-based privacy-preserving framework for online social networks. *Connect. Sci.* **2020**, *33*, 555–575.
34. Mackey, T.K.; Miyachi, K.; Fung, D.; Qian, S.; Short, J. Combating health care fraud and abuse: Conceptualization and prototyping study of a blockchain antifraud framework. *J. Med. Internet Res.* **2020**, *22*, e18623.
35. Nguyen, B.M.; Dao, T.C.; Do, B.L. Towards a blockchain-based certificate authentication system in Vietnam. *PeerJ Comput. Sci.* **2020**, *6*, e266.
36. Li, X.; Xu, J.; Fan, X.; Wang, Y.; Zhang, Z. Puncturable Signatures and Applications in Proof-of-Stake Blockchain Protocols. *IEEE Trans. Inf. Forensics Secur.* **2020**, *15*, 3872–3885.
37. Ante, L. Smart Contracts on the Blockchain—A Bibliometric Analysis and Review. *Telemat. Inform.* **2020**, *57*, 101519.
38. Prashar, D.; Jha, N.; Jha, S.; Joshi, G.P.; Seo, C. Integrating IOT and blockchain for ensuring road safety: An unconventional approach. *Sensors* **2020**, *20*, 3296.
39. Amato, F.; Cozzolino, G.; Moscato, F.; Moscato, V.; Xhafa, F. A Model for Verification and Validation of Law Compliance of Smart-Contracts in IoT Environment. *IEEE Trans. Ind. Inform.* **2021**, *17*, 7752–7759.
40. Wan, W.; Bentahar, J.; Hamza, A.B. Model checking epistemic–probabilistic logic using probabilistic interpreted systems. *Knowl.-Based Syst.* **2013**, *50*, 279–295.
41. Corradini, F.; Marcelletti, A.; Morichetta, A.; Polini, A.; Re, B.; Scala, E.; Tiezzi, F. Model-driven engineering for multi-party business processes on multiple blockchains. *Blockchain Res. Appl.* **2021**, *2*, 100018.
42. Liang, C.; Feng, L.; Zhengyi, X.; Hao, D. Blockchain: Research on the Application Progress of the Internet of Things. *Internet Things Technol.* **2018**, *8*, 100–103.
43. Aslam, J.; Saleem, A.; Khan, N.T.; Kim, B.Y. Factors influencing blockchain adoption in supply chain management practices: A study based on the oil industry. *J. Innov. Knowl.* **2021**, *6*, 124–134.
44. López-Pintado, O.; García-Bañuelos, L.; Dumas, M.; Weber, I.; Ponomarev, A. Caterpillar: A business process execution engine on the Ethereum blockchain. *Softw. Pract. Exp.* **2019**, *49*, 1162–1193.