Computers & Industrial Engineering 157 (2021) 107347

Contents lists available at ScienceDirect

Computers & Industrial Engineering

journal homepage: www.elsevier.com/locate/caie

A new paradigm of modeling for military industrial logistics agent interaction protocol: Command interaction diagram

Xiong Li [a,b,*], Wei Pu [b,*], Wei Zhang [c,*], Xiaodong Zhao [b]

[a] School of Systems Science and Engineering, Sun Yat-sen University, Guangzhou 510006, China
[b] Military Exercise and Training Center, Army Academy of Armored Forces, Beijing 100072, China
[c] School of Management, Xiamen University, Xiamen 361005, China

ARTICLE INFO

Keywords:
Industrial logistics
Industrial logistics system
Agent-based modeling
Industrial logistics simulation
Agent interaction protocol

ABSTRACT

Agent-based modeling is a useful and effective approach to exploring industrial logistics system, while agent interaction protocol is a key to support agent-based industrial logistics modeling. Current modeling methods are mainly developed to face general industrial systems and are in a certain sense difficult to fully meet the need of modeling military industrial logistics system. In this research, we focus on describing and implementing military industrial logistics agent interaction protocol by presenting a new command interaction diagram (CID) oriented paradigm. We not only present the concept and corresponding graphical notation representations of CID for multi-agent interaction process, but also design and develop the CID modeling tool with a function of logic verification, thus realizing the mechanism of automatic transformation from conceptual model of CID to its simulation model. In addition, we provide a case study and perform agent-based military industrial logistics simulation. The simulation demonstration and corresponding validation show that our method can meet actual needs and achieve good results, with more advantages than other traditional modeling methods. This fact proves the feasibility and effectiveness of the proposed method.

1. Introduction

Industrial logistics plays a key role in industrial engineering (Li et al., 2019). It involves the physical inflow and outflow of goods and corresponding services (Barros, 1997). Of course, it can be further divided into product support logistics, production support logistics and industry support logistics (Barros, Riley, & Brown, 2001; Li et al., 2019). With the evolution of technology, the methods for warfare have also changed. In modern wars, as the military application field of industrial logistics, military industrial logistics has an obvious characteristic in logistics entity interaction based on both command orders and information sharing, thus forming an integrated system support capability. In view of this new support mode, it is necessary to conduct more scientific and reasonable analysis and research by establishing simulation models of military industrial logistics system. Modeling military industrial logistics entity interaction has become a key bottleneck problem in exploring the running mechanism of military industrial logistics system.

Although some methods, such as exercise and training, man-in-the-loop simulation experiment, can be used in simulation for military industrial logistics system, there are some disadvantages, including abundant capitals, high cost and inconvenience in operation. As far as traditional analytical modeling methods are concerned, generally, they are difficult to describe the intelligent behavior characteristics of military industrial logistics system. Intelligent agent is a program that maps percepts to actions. It acquires information from its environment ("perceives" the environment) and decides about its actions and performs them (Lesser, 1998; Wooldridge & Jennings, 1999; Shi, 2001). Obviously, the internal member entities of military industrial logistics system, such as logistics command vehicles, repair vehicles and rescue vehicles, can be viewed as intelligent agents. Thus, using agent-based modeling method to describe military industrial logistics entities is a feasible and effective solution.

Agent interaction, i.e., interaction between agents, is the important embodiment of agent swarm intelligence. Modeling agent interaction is the key to modeling social and military systems through multi-agent paradigm. Agent interaction protocol is the abstraction and regulation

* Corresponding authors at: School of Systems Science and Engineering, Sun Yat-sen University, Guangzhou 510006, China (Xiong Li); Military Exercise and Training Center, Army Academy of Armored Forces, Beijing 100072, China (Wei Pu); School of Management, Xiamen University, Xiamen 361005, China (Wei Zhang).

E-mail addresses: lixiong2609@126.com (X. Li), 13810695657@163.com (W. Pu), iwzhang@xmu.edu.cn (W. Zhang).

https://doi.org/10.1016/j.cie.2021.107347
Received 16 December 2019; Received in revised form 11 January 2021; Accepted 18 April 2021
Available online 20 April 2021
0360-8352/

of conversation interaction between agents. It reflects the purpose of interaction between agents. It is closely related to intelligent decision-making and reasoning mechanism of agents. Modeling military industrial logistics agent interaction is one of cores and difficult problems of agent-based modeling for military industrial logistics system.

Thus, we focus our research on modeling of military logistics agent interaction protocol. Based on our previous practice in developing agent-based models and simulation systems, a novel agent interaction modeling paradigm, command interaction diagram (CID), is proposed, and accordingly, military logistics agent interaction protocol model through CID is constructed. CID oriented conceptual modeling and automatic mechanism of transforming conceptual model to its simulation model are proposed, with the designed and implemented modeling tool. As an application, the proposed model and executable simulation program codes generated directly through the transformation mechanism are embedded in a military simulation system. The simulation results show that the interaction protocol meets the needs of simulation for military industrial logistics agents, and verify the feasibility and validity of the proposed approach.

The remainder of this paper is structured as follows. Current related work is introduced in Section 2. Construction of conceptual model for military industrial logistics agent interaction protocol is presented in Section 3. Transformation from conceptual model to simulation model by automatic generation of simulation codes is illustrated in details in Section 4. In Section 5, the proposed approach is applied to a case study to show its usefulness in modeling tactical military industrial logistics system. Section 6 summarizes the conclusion and describes some future work directions.

2. Related work

Industrial logistics plays an important role in our social life. Hofmann and Rüsch (2017) focused on Industry 4.0 and the current status as well as future prospects on logistics, and presented a logistics-oriented Industry 4.0 application model as well as the core components of Industry. Smith and Srinivas (2019) focused on warehouse check-in strategies for improving inbound logistics operations, and proposed the corresponding simulation-based evaluation technology. Industrial logistics system is complex, and activities are highly interactive between the portions of the system. As a result and from a desire to thoroughly understand all portions of the system, a series of simulation models have been developed (Klitz, 1983). Particularly, agent and multi-agent technologies came out in 1980s. Agent-based models developed by do Nascimento and de Lucena (2017), Gao et al. (2018a, 2018b), Bai, Yoo, Deng, Kim, and Gao (2016), and Rachid, Mohamed, and Khouaja (2018) have been proved to be useful and effective to describe social systems behavior. For instance, the effective and innovative research conducted by do Nascimento et al. (2018) showed how flexible points of their agent-based framework were instantiated to generate the Internet of Things application. Moreover, agent-based modeling has been successfully applied to describe organizational routine dynamics and simulate the evolutionary process of organization capability as well as to identify the underlying principles (Gao et al. 2018a, 2018b, Bai et al., 2016). Similarly, Rachid et al. (2018) proposed a bottom-up analysis method focusing on the characteristics of each agent at a micro-level, and presented an agent-based modeling approach in the strategic human resource management.

Agent-based simulation for industrial logistics has attracted the interest of researchers and engineers in this field for decades (Adediran, Al-Bazi, & dos Santos, 2019; Furtado & Eberhardt, 2016; Mishra, Kumar, & Chan, 2012; Reddy, Kumar, Fernandes, & Tiwari, 2017; Serrano-Hernandez, Faulin, Hirsch, & Fikar, 2018). Especially, Serrano-Hernandez et al. (2018) developed an agent-based simulation focusing on the evolution of horizontal cooperation in logistics and transportation over time in both degree of integration and size. The simulation experiments conducted by them highlight that significant savings can be achieved with the degree of cooperation and trust-related issues indicating the highest importance. Moreover, since the concept of agent and agent functions can help to automate a variety of difficult tasks and assist decision-making in flow-shop production, Adediran et al. (2019) proposed an innovative embedded agent-based Production Disruption Inventory-Replenishment (PDIR) framework and implemented a set of agent-based simulation experiments with useful results presenting efficient resource utilization and less order shortage. These novel works reflect the significant progress in the field of agent-based industrial logistics system simulation.

As mentioned earlier, an important prerequisite for implementing system modeling through multi-agent paradigm is appropriate agent interaction protocol. At present, agent interaction protocols mainly include two types: general protocols and special protocols. General protocols, including contract net protocol (Li et al., 2012; Panescu & Pascal, 2016; Gupta & Gallasch, 2016; Lee, Lee, Chen, & Wu, 2012; Hsieh & Chiang, 2009), have been widely and successfully applied in agent-based modeling. With good adaptability, these interaction protocols are highly abstract to human interaction activities and are very suitable for use in open network environment to realize the interaction between different agents. In terms of special protocols, López-Ortega and de la Cruz (2010), Kang and Sim (2012), respectively designed agent interaction collaboration for data exchange between enterprises and raster data mining. Unfortunately, in agent-based military industrial logistics system model, agent interaction is driven by command and control, meaning interaction behaviors have prominent military command characteristic. According to the actual requirement of military modeling, a special protocol model of military industrial logistics agent interaction should be established.

As for the expression form aspect of agent interaction protocol modeling, finite state machine (FSM) (Gupta & Gallasch, 2016; Lee et al., 2012; Winikoff, Yadav, & Padgham, 2018), Agent Unified Modeling Language (AUML) (Belloni & Marcos, 2004; Lee et al., 2012; Panescu & Pascal, 2016; Terzidou, Tsiatsos, & Apostolidis, 2018; Winikoff et al., 2018) and Colored Petri Net (CPN) (Dworzański & Lomazova, 2013; Liu & Heiner, 2013; Panescu & Pascal, 2016), are proved to be appropriate and useful methods. In terms of FSM, the notation is likely to be familiar to software engineers. Also, it is a simple notation (few constructs, and easy to explain) that is graphical in nature, and hence likely to be easier to use. This relates to the need to have a pragmatic notation that is usable by software engineers. Moreover, FSMs are precisely defined, which relates to the need to be able to provide tool support (Winikoff et al., 2018). In terms of AUML, the multi-view notations consist of structural type and behavioral type. The former includes some static diagrams, such as class, object, deployment and component, while the latter contains some dynamic diagrams, such as sequence, state chart, activity and interaction diagrams (Bashir, Lee, Khan, Chang, & Farid, 2016). Due to its good capability to describe agent-based models, AUML can provide built-in mechanisms to introduce new elements for specific domains (Belloni & Marcos, 2004). In terms of CNP, it offers a simple and general mechanism for solving coordination of real-world entities, and thus it can be included in planning, scheduling, reconfiguration and control schemes (Panescu & Pascal, 2016). Similarly, these methods are mainly developed to face general industrial systems and are in a certain sense difficult to fully meet the need of modeling military systems. As for these traditional agent interaction protocol modeling methods, generally speaking, there is a problem that the correspondence relationship between state transition and interaction process is not described clearly and accurately. Specifically, there is a lack of mechanism to transform conceptual model into computer simulation model automatically.

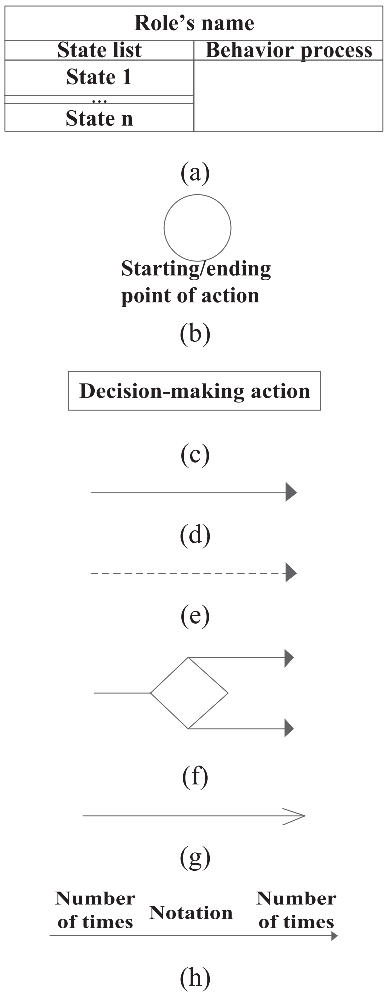


Fig. 1. Elements of a CID and their graphical representations and functions.

3. Construction of conceptual model

3.1. Agent interaction protocol analysis: Proposition of CID

According to the formation of support force in a typical tactical force and the mapping relationships between real entities and agents, we can further describe the constructed structure of agent-based military industrial logistics system, which generally consists of two support units subordinate to the command agent at the troop level. Each support unit includes a command agent and several support element agents, such as repair vehicle agents, rescue vehicle agents, supply vehicle agents, et al. These interaction behaviors include not only storage, supply, service, technical reconnaissance, command and control relationship between superiors and subordinates, but also support and coordination relationships between the same level.

In order to construct the model of military industrial logistics agent interaction protocol, we perform further analysis. Firstly, the types of agents in system organization are abstracted and merged to form force-level logistics command agent, for example, brigade logistics command vehicle agent, logistics support team command agents and general support action element agents. Secondly, according to the actual command content, the interaction list of each agent is established. The interaction of force-level logistics command agent centers on issuing hierarchical or over-hierarchy command orders and support tasks at different levels. Similarly, the interaction actions of logistics support team command agents include issuing hierarchical command orders and support tasks at different levels, as well as stimulating same-level coordination. As for general support action element agents, there are three representative interaction actions, i.e., same-level coordination, task support requesting and autonomous-regulated action.

Thereby, it is necessary to build a protocol system model for military industrial logistics agent interaction and form a complete organization structure to meet the communication needs under various circumstances. Here, we present a novel protocol specification for agent interaction, i.e., CID. The graphical representations and functions of different elements in a CID are shown in Fig. 1. The elements of a CID are follows: (a) interaction role; (b) starting point and ending point of action, representing the actions of transmitting and receiving information from interaction roles; (c) decision-making action, representing the command activities requiring interaction role make decision to influence the interaction process, (d) basic interaction relationship, representing the interaction behavior of different interaction roles, (e) environmental information interaction, representing the messages sent to interaction roles by external environments; (f) branch interactions, representing branch interactions; (g) internal execution process, representing the connection between command interaction and decision-making action; and (h) interaction notation, representing the n-to-m relationship of interactions and corresponding annotations on interactions.

3.2. Conceptual modeling for agent interaction protocol: Formation of CID

CID description of military industrial logistics agent interaction protocol is the core of conceptual modeling for interaction protocol. According to the organization design of military industrial logistics agent interaction protocol, the graphical elements of a CID are defined for each interaction protocol, and corresponding CID description is established, respectively, as shown in Fig. 2. In the CID of each interaction protocol, the interaction role agent takes idle as the basic state, the interaction of some environment information as the trigger condition, and the realization of the current interaction goal as the completion condition to form a finite state transition interaction space, which lays conceptual model foundation for the simulation model generation of the interaction protocol.

These CID models shown in Fig. 2 represent six kinds of agent interaction behaviors, i.e., command by level, over-hierarchy command, command of assigning task, command of requesting support, command of self-autonomous cooperation and command of autonomous-regulated action, respectively. The above command behaviors are the most typical basic interaction behaviors of military entities. In other words, no matter what military mission is carried out, it can be realized through these command behaviors. Therefore, as long as these CID models are constructed, the command interaction in real military industrial logistics system can be described. It should be pointed out that in an actual battle, the command interaction between military entities may adopt different interaction modes in different operational phases according to the real-time battlefield situation. The purpose of our research is to describe the command interaction, simulate the actions of various military entities, and present statistical analysis on their command behaviors.

In this study, we take the conceptual model of CID of self-autonomous cooperative agents as an example. It is the sixth kind of command behavior, i.e., command of autonomous-regulated action. This model describes the command interactions between the same level support agents, which jointly implement the support tasks by self-cooperative way. The model includes two interaction roles: the cooperative initiator and the cooperative participant agent.

Firstly, the cooperative initiator agent finds that it can't accomplish its tasks independently during performing logistics tasks. From the idle state of command interaction, it executes the decision-making action of seeking cooperative objects. After selecting a cooperative object, it sends cooperation requests to the cooperative participant agent. When the cooperative participant agent receives the collaboration request, it

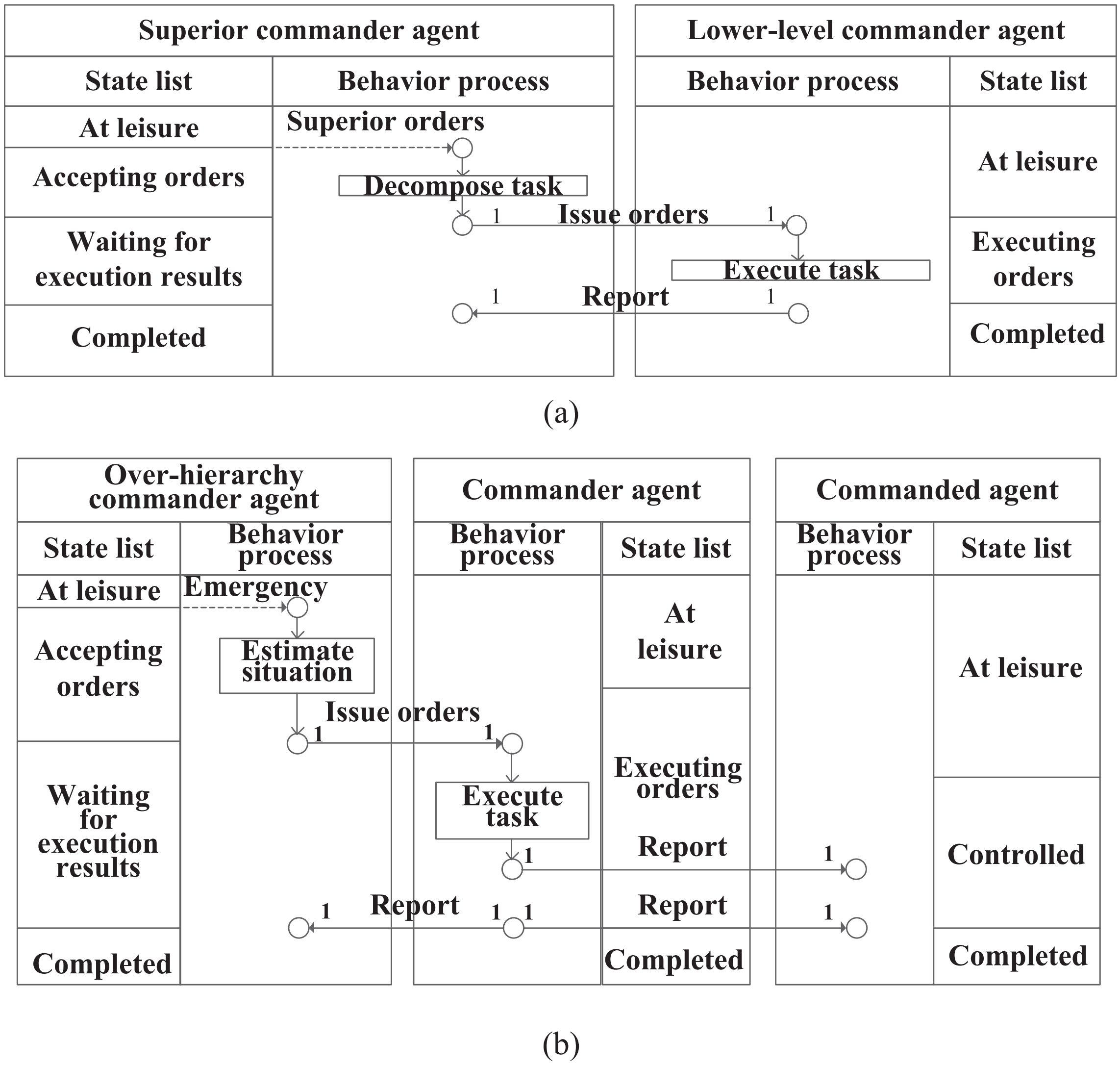


Fig. 2. CID models of military industrial logistics agent interaction protocol.

executes the decision-making action to judge whether it is collaborative or not, and according to the decision-making result, it chooses to send collaborative or uncoordinated interaction information. If the interaction information that does not participate in collaboration is sent, the interactive agent roles of both parties are transferred to the protocol completion state.

Secondly, when the interaction information of participating collaboration is sent, the cooperative initiator agent executes the decision-making action of the cooperative scheme, sends the cooperative control information to the cooperative participant agent, and the cooperative participant agent executes the corresponding cooperative tasks according to the cooperative control information.

Finally, according to the implementation of collaborative tasks, if there is a need to further clarify collaborative matters, collaborative participant agent sends collaborative information notification to the collaborative initiator agent, collaborative initiator agent sends collaborative control information according to the content of the notification, and when the collaborative participant agent completes the collaborative task, it sends collaborative initiator agent to report completion, thus ending the interaction process.

4. Transformation from conceptual model to simulation model

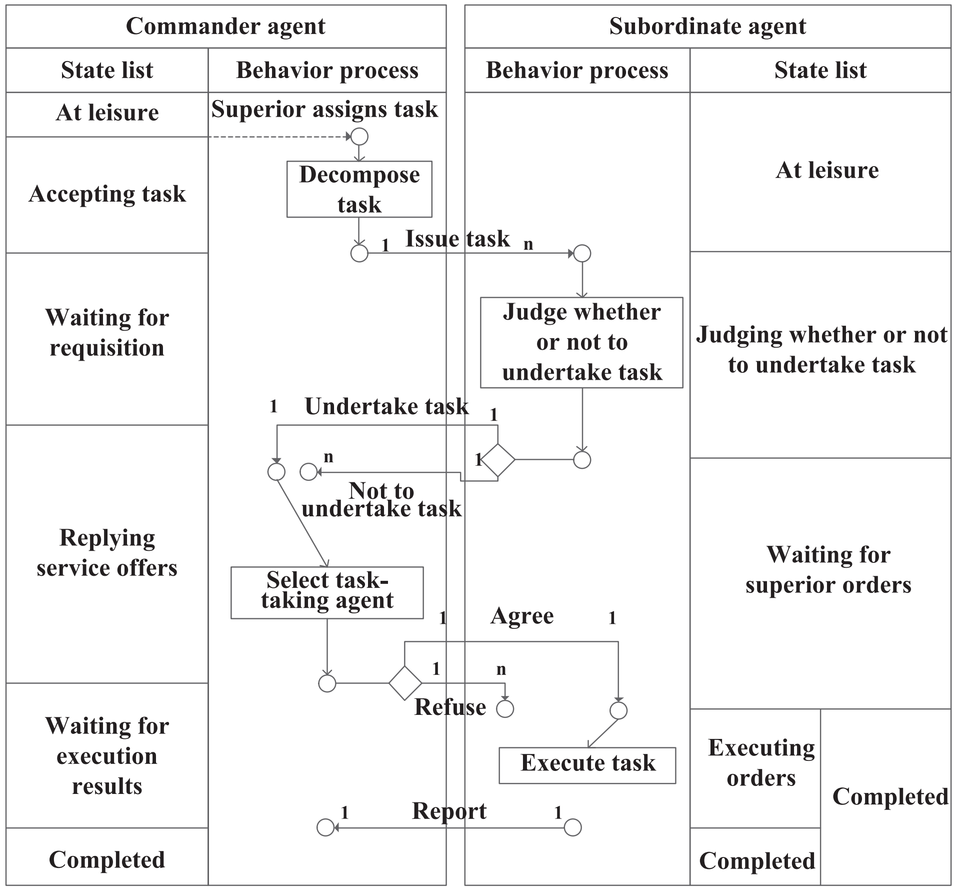
4.1. Method: Transformation from CID to simulation model

The transformation from a CID to its simulation model is the key of agent interaction protocol modeling based on CID. It is the process of converting a visualized CID model into the corresponding simulation model. Its essence is the mapping from the graphical elements of a CID to the corresponding simulation codes. According to the principles of object-oriented programming, the transformation relationship between the graphical elements of a CID and the code class, function and statement in its simulation model is shown in Table 1.

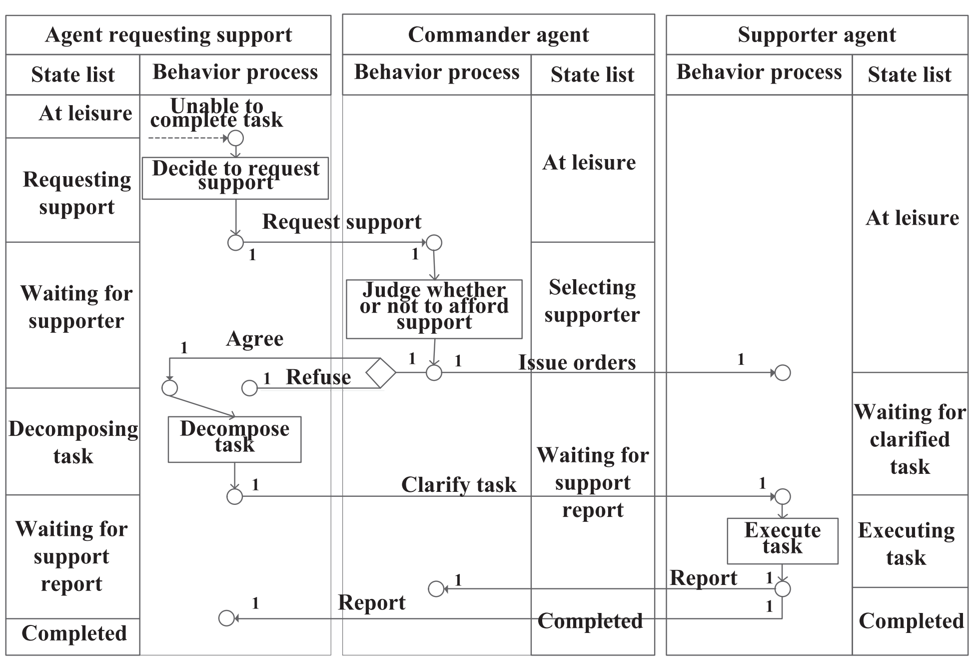
Specifically, the method of transformation from CID to simulation model can be described as follows.

(1) Each interaction protocol is transformed into an interaction protocol subclass inherited from InteractionProtocols, and the interaction role agent in each protocol is transformed into an agent role subclass inherited from AgentActor. An interaction protocol subclass includes several agent role subclasses that interact with each other.

(2) Each agent role subclass includes an Interact(int) function, which is the core function of the agent role subclass and the key to implement the interaction protocol model. Its main function is to determine the next interaction according to the current state (received messages,



(c)



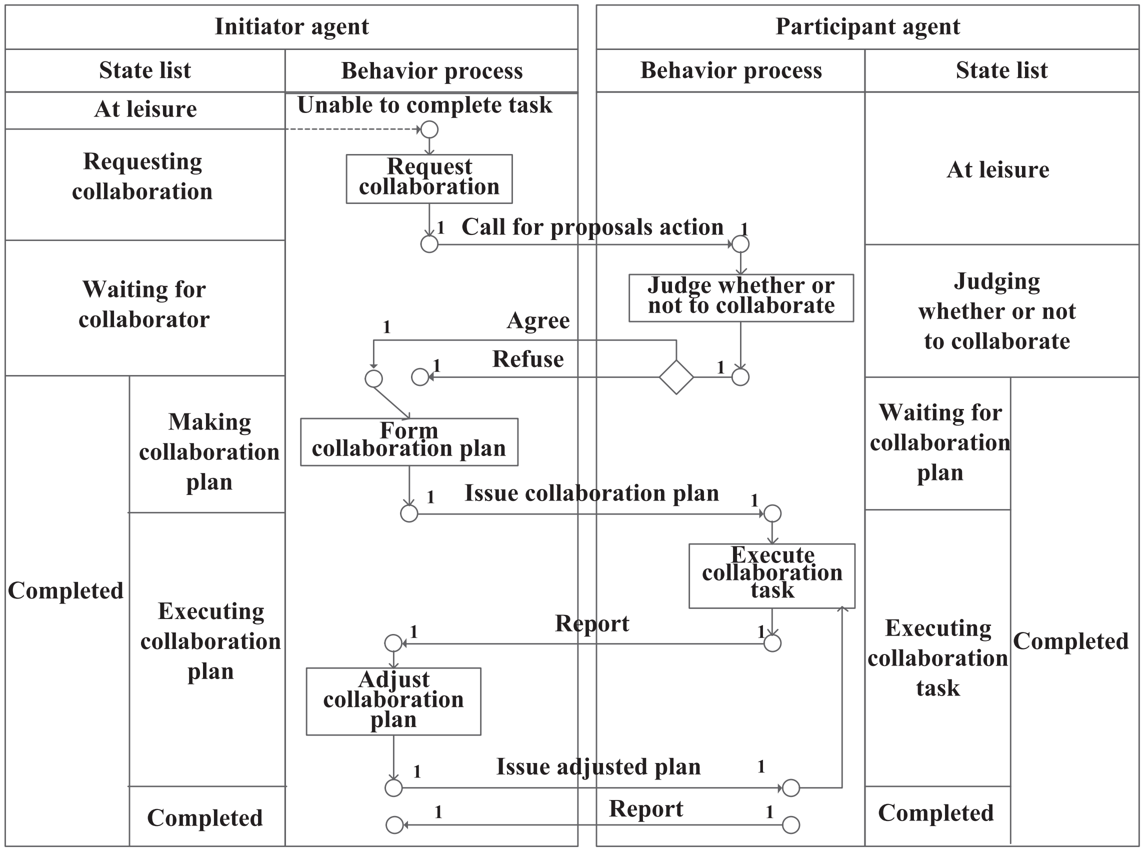
(d)

Fig. 2. (*continued*).

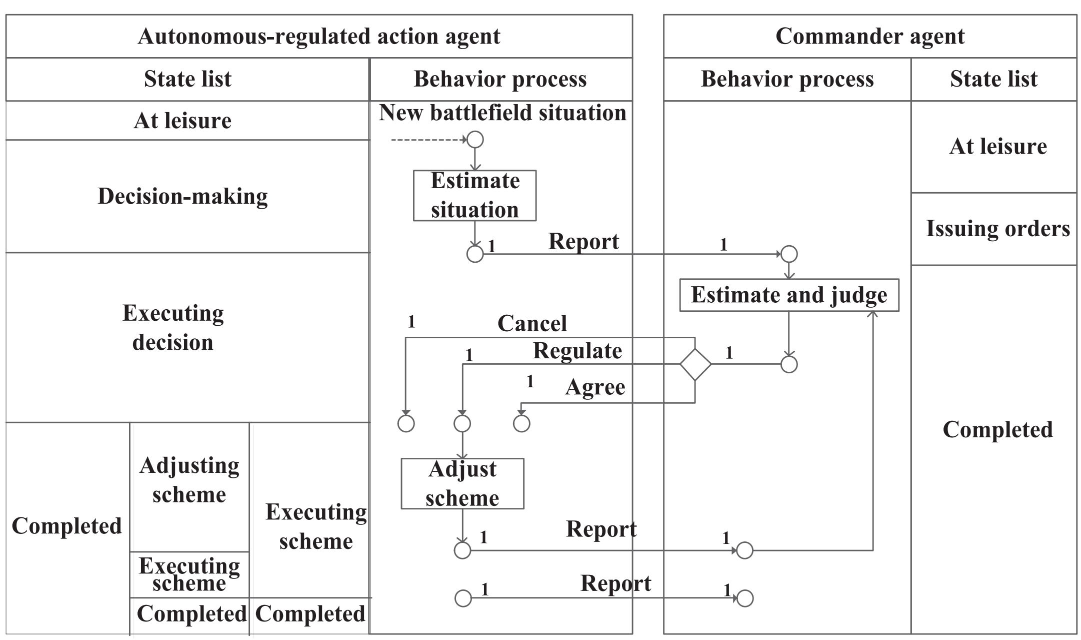
interaction state of the agent role).

(3) Each subclass of agent roles includes a DecAction(int) function, which realizes all decision-making actions in the CID and distinguishes actions according to the parameters.

(4) The state and state transition of each agent role are implemented by switch…{case:…;break;} statement, and each state corresponds to a



(e)



(f)

Fig. 2. (*continued*).

macro definition.

(5) The graphical elements of branch interaction relationship in each agent role interaction are implemented by being transformed into if … then … statement.

According to this method, the related functional base classes for our agent interaction protocol are designed. The class diagram is shown in Fig. 3. The attributes and main methods of functional base classes are defined in the diagram. It mainly includes three basic classes: Interaction, InteractionProtocols and AgentActor. InteractionProtocols and AgentActor inherit from Interaction classes, and constitute a one-to-

Table 1
Transformation relationship between the graphical elements of a CID and its simulation model.

No.	Graphical elements of a CID	Classes, functions and statements of simulation codes
1	Interaction protocol	A subclass of InteractionProtocols
2	Interaction role agent	A subclass of AgentActor
3	Command interaction	void AgentActor:: Interact(int) function
4	Decision-making action	int AgentActor::DecAction(int) function
5	State transition	switch…{case:…;break;} statement
6	Logic relation of interactions	if … then … statement

many relationship.

The query process of an interaction protocol based on functional base classes is shown in Fig. 4. Firstly, agent 2 receives the interaction information sent by agent 1. According to the interaction protocol, interaction content and interaction object adopted by interaction information, agent 2 calls Interact() function of the corresponding interaction protocol class InteractionProtocols object. Then, according to the receiving object of interaction information, this function calls Interact() function of the corresponding role class AgentActor object. Subsequently, according to the interaction content, this function calls StateTranslate() function of the object to realize state transition, as well as creates and returns the Interaction class object. Finally, agent 2 sends the corresponding interaction information to agent 1 according to the attribute data of the returned Interaction class object.

4.2. Transformation implementation: CID modeling tool

According to the graphical symbol definition of CID and the method of transformation from CID to simulation model, a CID modeling tool is designed and implemented. By this tool, we can construct the visual conceptual model of agent interaction protocol. According to this conceptual model, the simulation model can generate automatically the simulation model codes. Of course, we also design and implement a logic verification module. The conceptual modeling interface, corresponding simulation codes generation interface and logic verification interface are illustrated in Fig. 5 (a), Fig. 5 (b) and Fig. 5 (c), respectively.

In Fig. 5 (a), we use this modeling tool to construct conceptual models as described earlier. These models are visualized through the modeling tool, which can show the interaction process between different role agents more clearly. Note that the process of establishing the conceptual model of agent interaction protocol is actually based on the graphical representations method given above. The use of this tool not only makes the modeling process visualized, but also has great convenience.

In Fig. 5 (b), we use this modeling tool to realize the automatic transformation from conceptual model to computer simulation executable codes. Note that this transformation process is based on the method of transformation from CID to simulation model, as mentioned earlier.

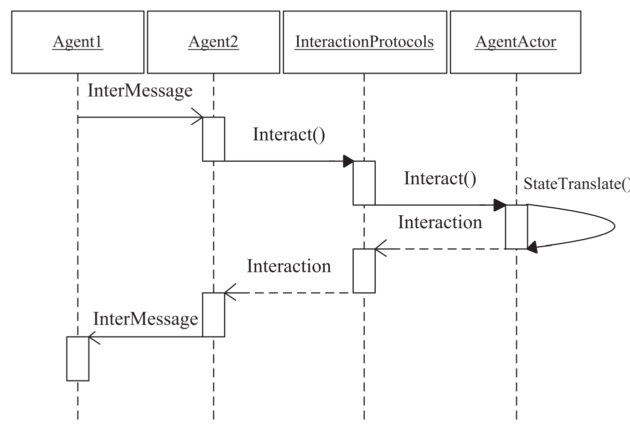


Fig. 4. Query process of an interaction protocol.

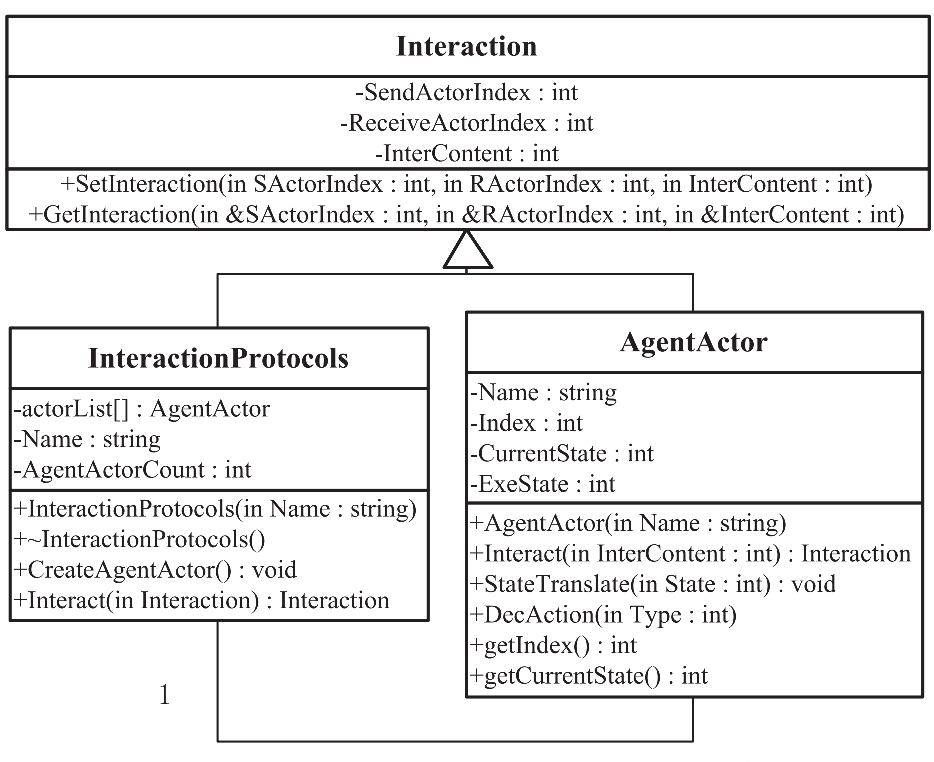
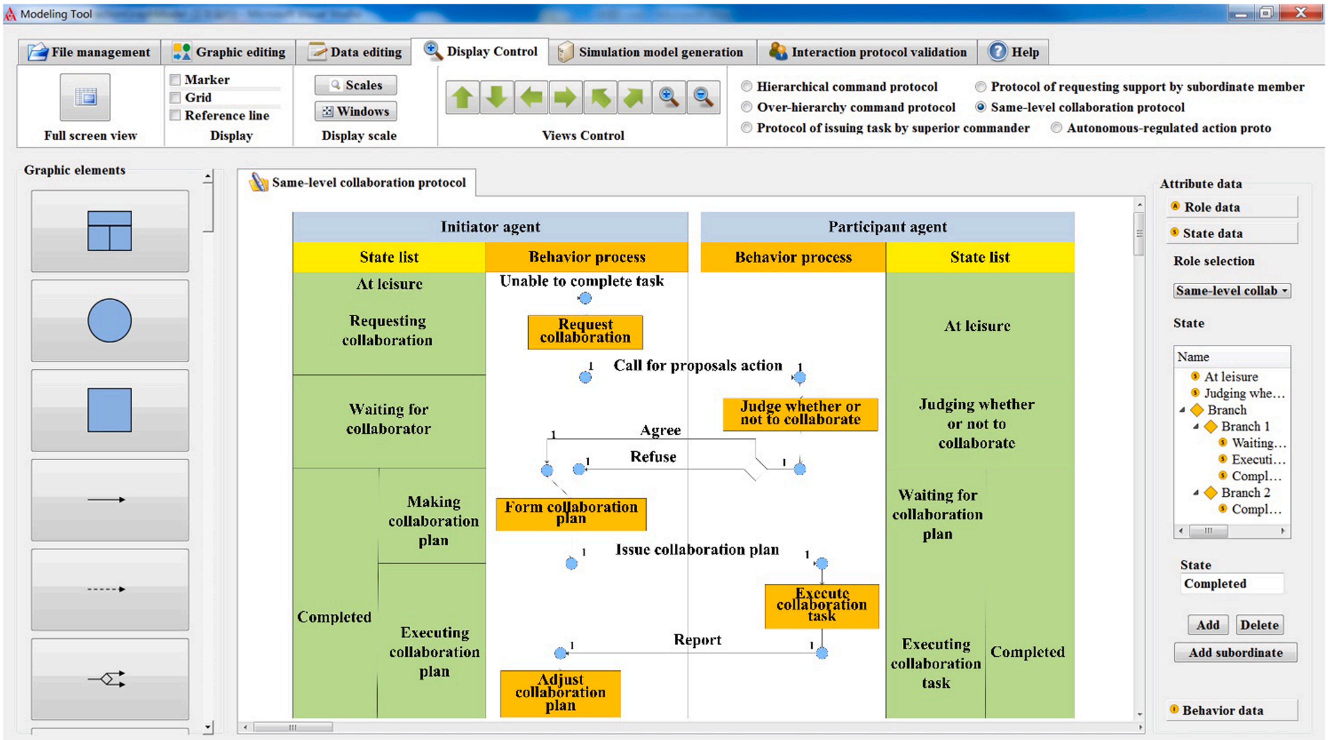
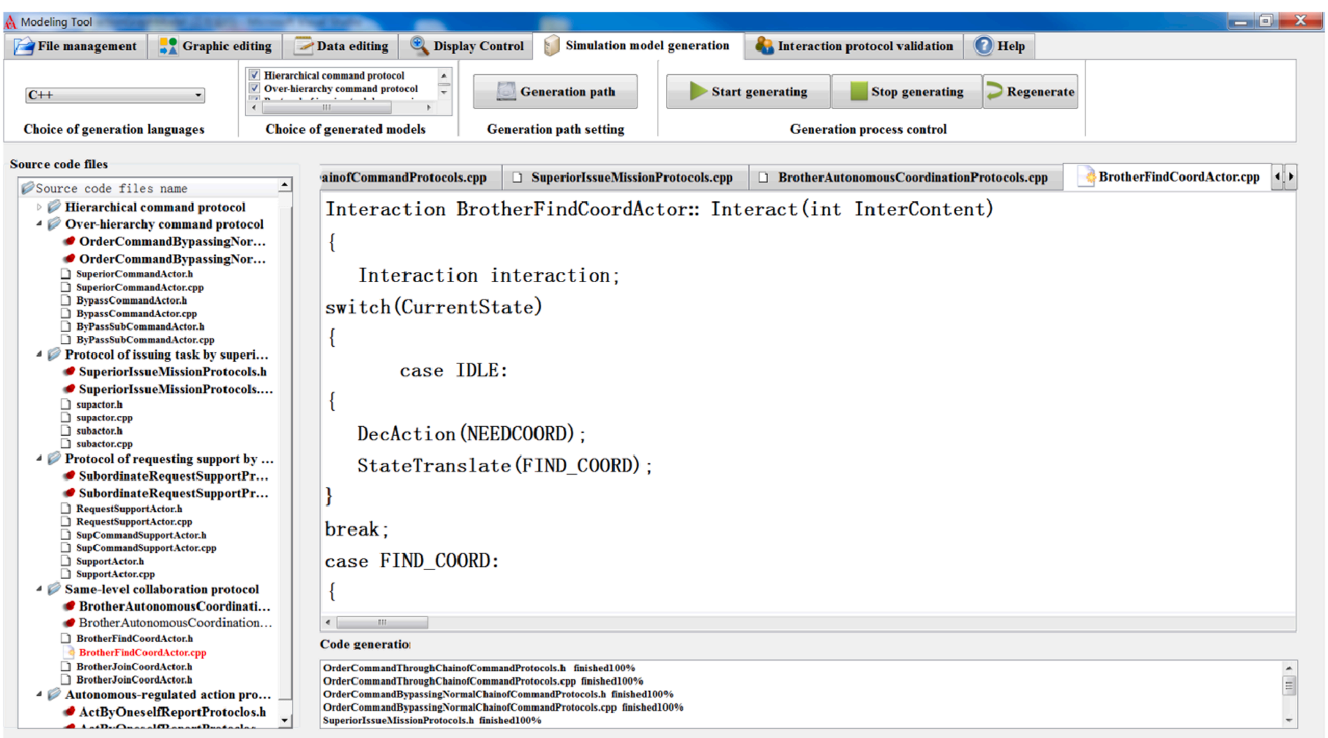


Fig. 3. Functional base classes.



(a)

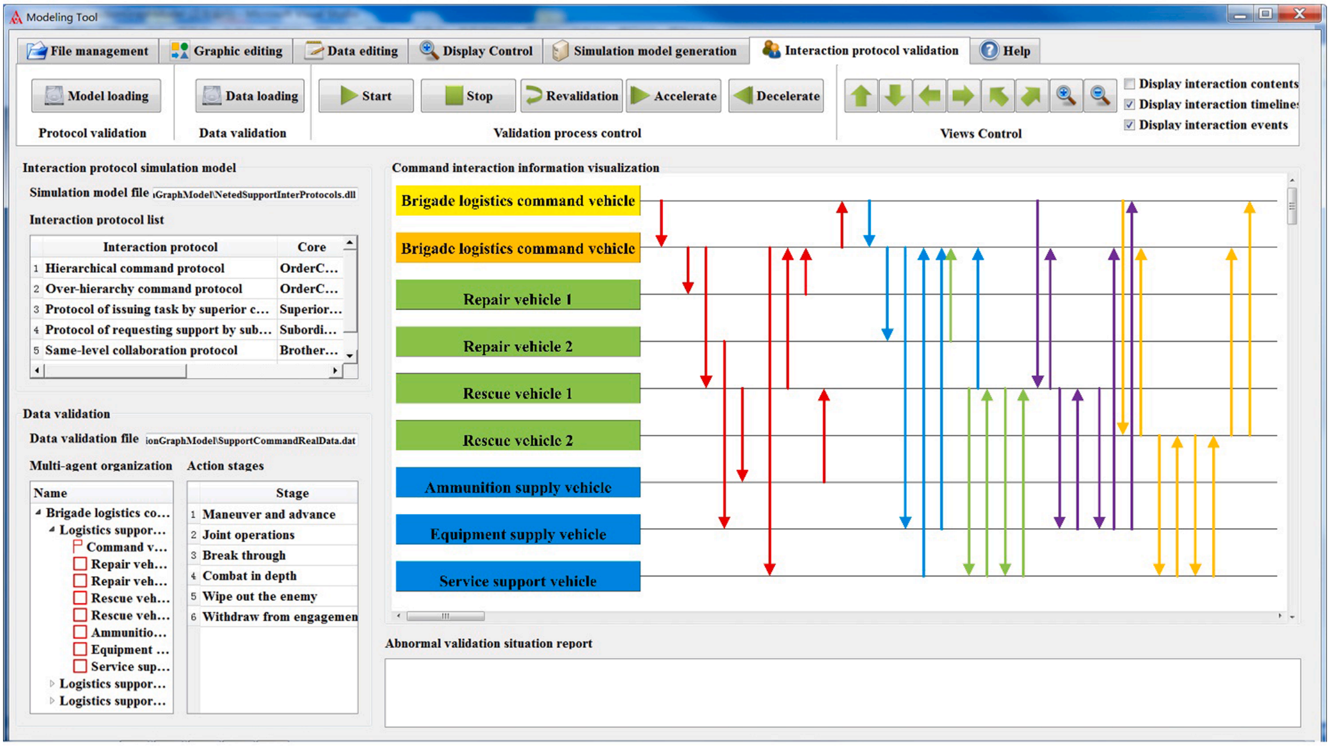


(b)

Fig. 5. CID modeling tool.

This makes the development of computer simulation program more professional, and greatly reduces the workload of software development. This advantage makes it possible for conceptual modelers to dominate agent-based modeling and simulation, which makes commanders more willing to use agent-based military logistics simulation experiments.

As for the logic verification module shown in Fig. 5 (c), it is used to provide the function of logic verification for simulation model, which



(c)

Fig. 5. (*continued*).

centers on: consistency between simulation model and conceptual model in concurrent state, deadlock in logic process of simulation model and accessibility of all states of simulation model. Based on the tool developed, the checking data of military industrial logistics entity object and concurrent interaction generation are constructed, and logic verification of military industrial logistics agent interaction protocol simulation model is carried out. As shown in Fig. 5 (c), the arrows with different colors indicate different interaction protocols. After logic verification, the simulation model of the interaction protocol is consistent with the conceptual model, and there are no deadlock, unreachable state and other interaction anomalies, which proves to be qualified.

5. Application: A case study

5.1. Conditions settings

In order to simulate and verify the military industrial logistics agent interaction protocol model, a military simulation scenario is constructed, as following. (1) The battle event happens in the topography of a plain with some low hills. (2) The opposed forces are combined army forces with conventional weapons. On the battlefield, a combat platform, e.g., a tank, needs to be supported by military industrial logistics system consisting of command vehicles, repair vehicles, rescue vehicles and supply vehicles. (3) Red Force attacks Blue Force and Blue Force launches defensive operations. No matter attack or defense, combat platforms and equipment support platforms take respective command behaviors according to the operational needs in order to perform military tasks. (4) Neither Red Force nor Blue Force have made big tactical mistakes, and Red Force wins finally. In fact, this scenario relates to simulation conditions settings, including battlefield environment, combat forces and target tasks, support resources and support force.

In peacetime, war experiment using real forces and weapons can be regarded as an effective way to test the combat effectiveness of troops. The conditions of the above military simulation scenario are derived from a real war experiment in RHZ region of northern China in August 2011. 55 military industrial logistics entities took part in the battle. Through participating in this war experiment, we collected data about combat and military industrial logistics. Especially, data centering on command behaviors in a complete battle were recorded.

This study is the continuation and deepening of our previous researches on agent-based military modeling. Here, we further specify the setting of support forces to demonstrate military industrial logistics agent interaction protocol from the perspective of CID. According to the quantity and type of support force and its command relationship, we have set up a force-level equipment support command agent with three subordinated support units. Each support unit includes a command agent and six support elements agents (maintenance, rescue, ammunition storage, equipment storage, service and technical reconnaissance). In this simulation, 55 military industrial logistics agents are used to demonstrate the real-world battle event. Command and be-commanded relationship is formed between superiors and subordinates, and collaborative support relationship is formed between role agents in at the same command level.

Through the above modeling tool developed by us, the simulation model codes of military industrial logistics agent interaction protocol are compiled in Microsoft Visual Studio 2010 compiler environment with Qt 4.8.4 framework. The hardware environment for system operation: CPU Xeon 3.2 GHz, memory 32G, NVIDIA Quadro5000, and Windows 764-bit flagship version. According to the structure of the interaction protocol, six dynamic libraries including OrderCommand ThroughChainofCommandProtocols.dll, representing different interaction protocols, are generated. These dynamic libraries are embedded in the "agent-based military industrial logistics simulation system" developed earlier by us, and the protocol is selected and used in the simulation process by using the autonomous decision-making function of the existing agents in the simulation system.

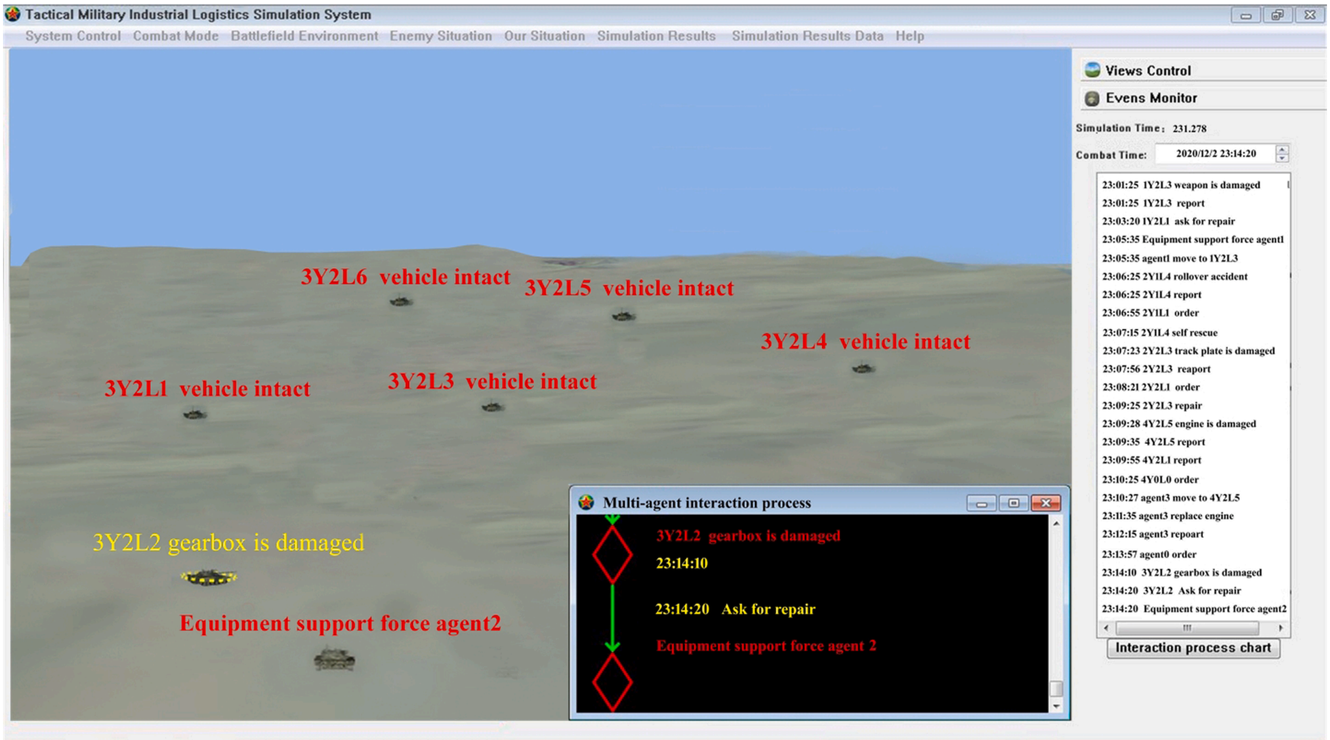


Fig. 6. Simulation demonstration interface.

5.2. Simulation demonstration

Based on the proposed scenario, agent-based simulation demonstration is carried out using military industrial logistics agent interaction protocol. The simulation demonstration interface is shown in Fig. 6, which presents two-dimensional view on real-time simulation situation. Of course, the process of battle damage generation and corresponding operations management of battle damage assessment and repair by using support resources and support force are also illustrated in Fig. 6, from which we can see the running process of different role agents performing support tasks by military industrial logistics agent interaction protocol.

This research focuses on command behaviors in military industrial logistics system. Since entity members from this real-world system in the battle event are mapped into respective agents in the simulation system, agent interaction behaviors are essentially used to reflect command behaviors executed by these entities. In other words, according to the set initial conditions, each agent runs independently on the specified virtual battlefield, and the final simulation demonstration process represents a real combat process. Therefore, once the simulation results are consistent with the actual correct system behaviors, we can consider the simulation model is reasonable. This fact means that the consistency between a certain command behavior in actual combat and the simulated command behavior in military simulation can be tested.

In addition, we introduce a metric, the correctness of command behavior, referring to the credibility and accuracy of the simulation results of a certain command behavior, which also indicates the stability of the simulation system. If the simulation system runs k times, the number of times a certain command behavior occurs in the i-th simulation experiment is recorded as y_i and the mean times is defined as $\overline{y}$, then the mean correctness of command behavior can be calculated as $C = \frac{1}{k}\sum_{i=1}^{k}(1 - \frac{|y_i - \overline{y}|}{\overline{y}})$. Its value range is [0, 1]. The closer C approaches 1, the more credible and accurate the simulation results are, and the more stable the simulation system is.

Due to the randomness of simulation processes and results, a certain number of simulation experiments are needed to provide the statistics. Specifically, if the simulation demonstration replications are too few, that is to say, the sample of simulation results is too small, it is difficult to evaluate the credibility and accuracy of the simulation results. Obviously, simulation results with high credibility and high accuracy can help users to make decisions. However, there is no unified standard for the credibility and accuracy of simulation results, so it is necessary to determine the number of simulation experiments according to the actual situation. The credibility and accuracy of simulation results are determined by confidence level and the half length of confidence interval, respectively. In other words, the half length of confidence interval should be minimized at a higher confidence level. Generally, at a given confidence level 95%, when the number of simulation experiments exceeds 20, the half length of confidence interval can be calculated as $L = Z(\frac{\alpha}{2}) \times SD/\sqrt{R}$, where $\alpha = 1 - CI$, thus $R = Z(\frac{\alpha}{2})^2 \times SD^2/L^2$ (Wu & Liu, 2008; Yan & Pu, 2008). Here, SD represents standard deviation, CI represents confidence interval, and R represents the number of simulation experiments, i.e., the replications of simulation demonstrations in a round of tests under the same scenario.

Through 30 simulation experiments, the correctness and the corresponding standard deviation of each command behavior in the simulation are calculated, and then the confidence interval and the half length of confidence interval are determined. After multiple rounds of 30 replications of simulation demonstrations, it can be seen that the lower bounds of the confidence interval of over-hierarchy command and command of autonomous-regulated action are less than 0.8. In fact, 0.8 is usually regarded as a standard to verify the mean correctness of a behavior model in the field of military simulation (Hu, Yang, Si, & Zhang, 2008; Li, 2013). Through checking the simulation model, we can determine that this is due to the insufficient number of simulation experiments. In order to improve the credibility and accuracy of the simulation results, that is, to reduce the half length of confidence interval, it is necessary to increase the number of simulation experiments. Given that the half length of confidence interval is unknown, we add 10 simulation experiments for each round of replications of simulation demonstrations to determine the simulation times until the credibility

Table 2
Comparison of statistics on the correctness of command behavior under 30 and 100 replications of simulation demonstrations.

Replications	Command behavior	*Mean*	*SD*	*CI*: 95%
30	Command by level	0.9559	0.0285	(0.9457, 0.9661)
	Over-hierarchy command	0.8048	0.0896	(0.7728, 0.8369)
	Command of assigning task	0.8614	0.0606	(0.8397, 0.8831)
	Command of requesting support	0.8897	0.0710	(0.8644, 0.9152)
	Command of self-autonomous cooperation	0.8494	0.0801	(0.8207, 0.8780)
	Command of autonomous-regulated action	0.8182	0.0904	(0.7859, 0.8506)
100	Command by level	0.9489	0.0312	(0.9428, 0.9550)
	Over-hierarchy command	0.8199	0.0808	(0.8040, 0.8357)
	Command of assigning task	0.8416	0.0808	(0.8258, 0.8575)
	Command of requesting support	0.8725	0.0782	(0.8572, 0.8879)
	Command of self-autonomous cooperation	0.8835	0.0775	(0.8683, 0.8987)
	Command of autonomous-regulated action	0.8411	0.0849	(0.8245, 0.8578)

and accuracy requirement is reached, that is, the lower bounds of the confidence interval of the correctness of all command behaviors are not less than 0.8. When the number of simulation experiments is increased to 100, the confidence interval can meet the credibility and accuracy requirement based on the statistical analysis results of multiple rounds of 100 replications of simulation demonstrations. Comparison of the statistical analysis results of the correctness of six kinds of command behaviors by running 30 and 100 simulation experiments is listed in Table 2. Correspondingly, correctness of command behavior under 100 replications of simulation demonstrations is shown in Fig. 7.

In order to verify the consistency between the simulation experiment results and the real-world system operation, we further present statistics on the six most typical kinds of command behaviors and comparison of the consistency, as shown in Fig. 8 and Table 3, respectively. Fig. 8 presents six histograms, which illustrate the distribution of the numbers of times the six kinds of command behaviors occurs in 100 simulation experiments, respectively. These statistical results show that they are basically characterized by normal distributions. In Table 3, N is the number of times a certain command behavior actually occurs in the battle event mentioned above. n_i represents the number of times a command behavior occurs in the i-th simulation experiment, and AV_1 is the average value of n_i. AV_2 and AV_3 denote the average value of absolute deviations and relative deviations between the number of times a command behavior occurs in the simulation experiments and the actual number of times occurring in the battle event, respectively. SR is the square root of the mean square of deviations between the number of times occurring in the simulation experiments and the actual number of times occurring in the battle event. These parameters can be calculated as $AV_1 = \frac{1}{100}\sum_{i=1}^{100} n_i$, $AV_2 = \frac{1}{100}\sum_{i=1}^{100}|n_i - N|$, $AV_3 = \frac{1}{100}\sum_{i=1}^{100}\frac{|n_i-N|}{N}$, and $SR = \sqrt{\frac{1}{100}\sum_{i=1}^{n}(n_i - N)^2}$, respectively.

We carry through verification, validation and accreditation (VV&A) for the agent-based military industrial logistics simulation model. As professionals engaged in the disciplines of military command, military industrial logistics and command information system applications, four military experts are invited to participate in this work. This work includes five aspects: conceptual model verification and validation, simulation model generation process verification and validation, simulation model function verification, simulation model execution validation, and model accreditation.

(1) Conceptual model verification and validation. In terms of the conceptual model, on the one hand, we check whether the conceptual model is constructed using right methods; on the other hand, we check whether the agent interaction model is consistent with the interaction process of the entity members in real military industrial logistics system. Now that the concept model is verified and validated, we start the follow-up work.

(2) Simulation model generation process verification and validation. After performing conceptual model verification and validation, we focus on simulation model generation process verification and validation. This task is mainly to monitor and verify simulation execution. Specifically, monitoring and verifying the simulation process of command behavior under a certain military simulation scenario can be used to judge whether simulation execution is correct.

(3) Simulation model function verification. When monitoring the command interaction information data of a certain agent based on these simulation demonstrations, we use the logic verification module and find that the interaction process is strictly carried out according to the interaction protocol, and there is no problem that the demonstration process does not conform to the actual support process due to the design defects of the interaction protocol. Therefore, military industrial logistics agent interaction protocol satisfies the requirement of command interaction between agents in this agent-based simulation system. In other words, this agent interaction protocol model has been proved to be feasible and effective.

(4) Simulation model execution validation. In terms of simulation model execution, we verify whether the simulation system works well and whether the simulation data is reasonable. As mentioned above, the correctness of command behavior is focused on as the statistical parameter for the simulation design. The results in Table 2 show that the simulation purpose can be achieved when the simulation system runs 100 times. Referring to the numerical results and statistical analysis from Fig. 7, Fig. 8 and Table 3, after consulting with military experts we think that the agent interaction behaviors are basically consistent with

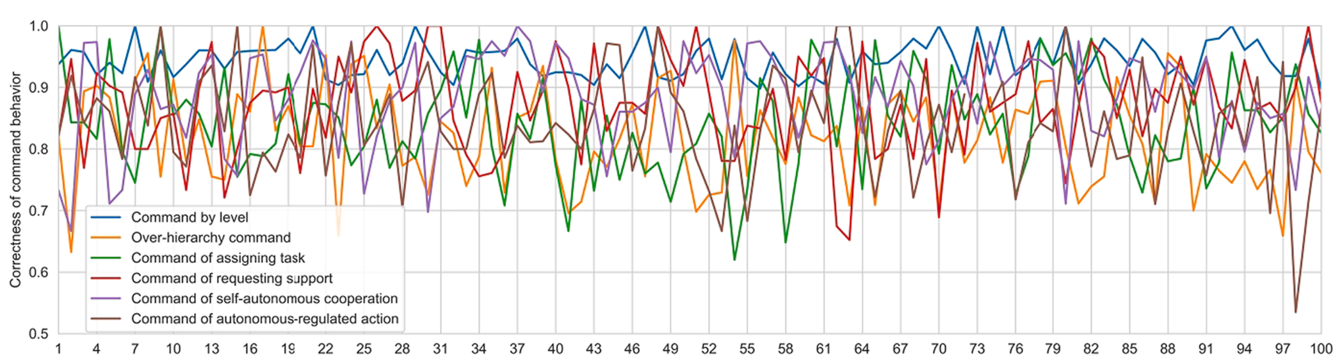


Fig. 7. Correctness of command behavior.

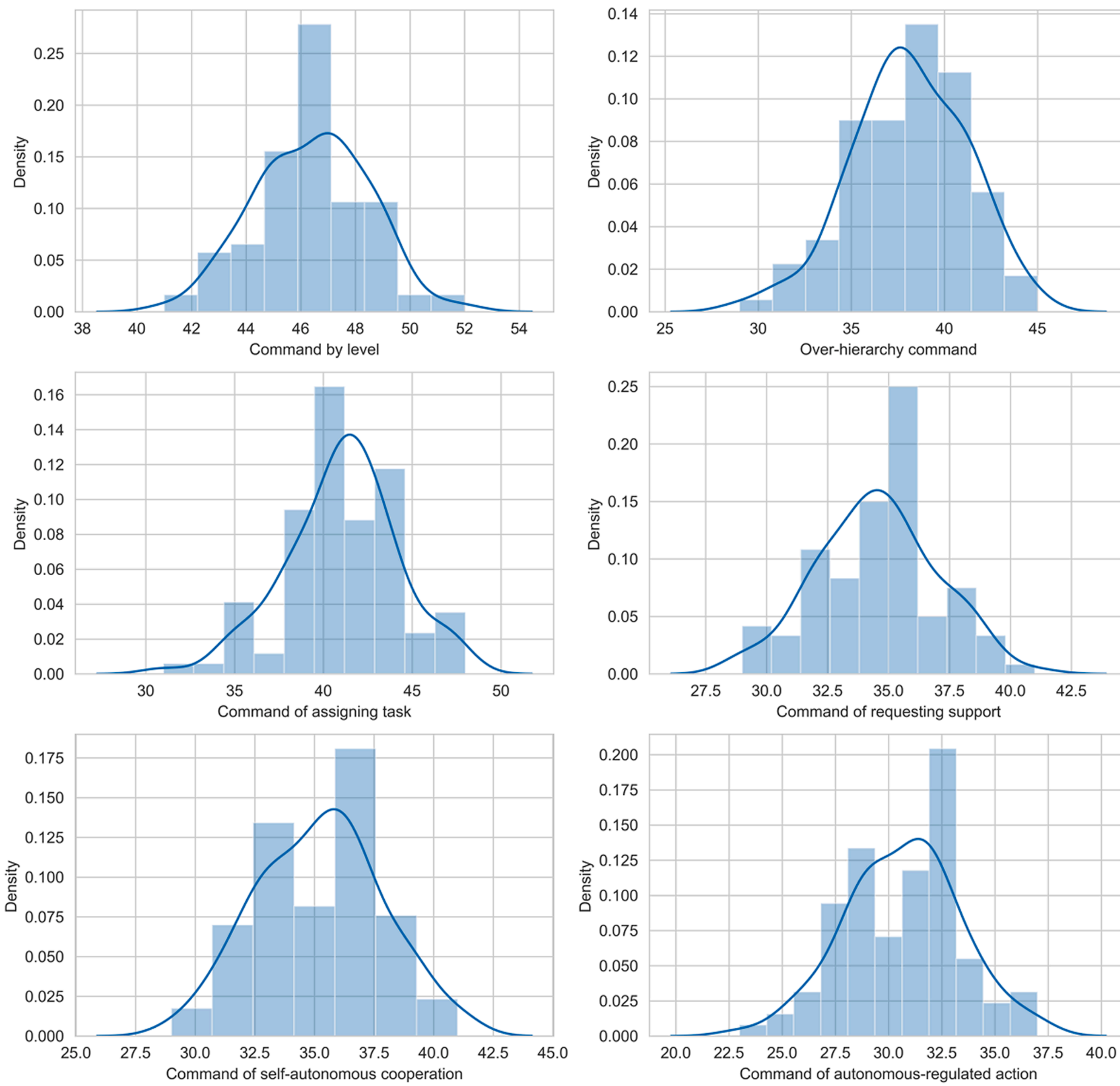


Fig. 8. Statistics on the six kinds of command behaviors.

Table 3
Comparison between simulation experiment results and real war experiment results.

Command behavior	Real war experiment results	Simulation experiment results			
	N	AV_1	AV_2	AV_3	SR
Command by level	47	46.41	1.67	0.0355	2.14
Over-hierarchy command	40	38.09	2.89	0.0723	3.61
Command of assigning task	42	41.03	2.45	0.0583	3.27
Command of requesting support	37	34.52	2.9	0.0784	3.47
Command of self-autonomous cooperation	35	35.22	2.1	0.0600	2.61
Command of autonomous-regulated action	32	30.66	2.32	0.0725	3.00

the actual military industrial logistics experience. This fact proves the validity and reasonability of the simulation system and corresponding simulation data.

(5) Model accreditation. Table 2 and Fig. 7 show that command by level has the highest mean correctness and over-hierarchy command has the lowest mean correctness under 100 replications of simulation demonstrations. According to the analysis of military experts, these results are consistent with the actual combat experience, that is to say, command by level is relatively easy to implement, while over-hierarchy command is the most difficult. Thus, our model can be accredited. Considering the complexity of command behavior and the uncertainty of simulation process, military experts also evaluate the statistical results in Fig. 8 and Table 3 as justified. This justification means that the simulation model can be approved.

5.3. Comparison with other methods

For some time, FSM, AUML and CPN have attracted the interest of researchers in modeling of agent interaction protocol, as mentioned earlier. FSM is a formal representation of state and state transition, and is more suitable for describing the state transition of a single entity, since its description of interaction sequence between multiple entities is weak. AUML protocol diagram can be used to standardize the description of interaction protocol between agents. Compared with the FSM method, the description of interaction sequence between multiple entities is stronger. However, its description of internal state and internal process of agents is weaker. CPN is based on a strictly defined formal language and has stronger description ability. Through CPN, model designers can accurately express the interaction protocol, and can verify the protocol based on CPN verification tools. Unfortunately, the CPN method is more abstract and complex to master and use, which is not convenient for military personnel to directly use for model description.

The results of case study have proven the convenience of our model. Compared to the above traditional modeling methods, our method, with more advantages, is more efficient, more accurate and easier to use in agent interaction protocol modeling. The comparison with other methods is listed in Table 4.

As far as representation forms are concerned, FSM, AUML and CPN are useful modeling methods with graphical notation representation for multi-agent interaction process. By interpreting the identified elements of multi-agent interaction process into graphical units, they are convenient to describe the conceptual model of real system activities. In this research, we present CID oriented paradigm by absorbing the graphical notation representation principles of these methods. In fact, the proposed functional base classes for our agent interaction protocol and the query process of an interaction protocol are described by UML class and UML sequence diagram, respectively. However, our method provides graphical description for agent-based military logistics system model by establishing its rule symbol system and forming visualization representations of CID. This representation form is completely different from FSM, AUML and CPN. Especially, our method provides interaction protocol codes generation capability. This is an obvious advantage over other methods.

As far as research procedures are concerned, it is difficult to provide a systematic procedure to describe multi-agent interaction process for military industrial logistics system, since it is a typical intelligent complex social system. FSM, AUML and CPN all have state, interaction and internal process description capability, by respective graphical notations. Of course, these description procedures themselves are complex. In particular, it is difficult for FSM to describe correspondence of state-interaction process. Further, the efficiency of model development and VV&A reflects the performance of research procedures. In terms of conceptual modeling, the development steps of the FSM and AUML modeling methods are twice or more than our method. In terms of simulation modeling, additional computer simulation codes need to be written when using FSM, AUML and CPN. Although conceptual model verification and validation of the above three methods are simpler than our method, but the iteration times of subsequent simulation model verification and validations are more, and the total number of iteration times are also more than our method. In our research, formation of CID involves conceptual modeling for military industrial logistics agent interaction protocol. In other words, by building CID, the conceptual model of military industrial logistics system can be constructed. Note that this procedure is followed by simulation modeling and simulation experiment by automatic transformation from conceptual model to simulation model. Thus, by CID modeling tool, the above model can be automatically transformed to executable simulation program codes to support corresponding agent-based simulation. Because of the use of executable conceptual model construction and subsequent automatic generation of simulation model, our method focuses the main modeling work on the conceptual modeling stage to avoid the problem of more development iterations when implementing the agent-based simulation model. Thus, compared with the traditional methods, our method improves the efficiency of modeling for agent interaction protocol.

6. Conclusion and future work

At present, agent-based modeling has become a hotspot in the field of military modeling. The purpose of agent-based military industrial logistics modeling is to demonstrate the individual and organizational behaviors of the actual logistics system by mapping the entities to corresponding agents and analyzing the multi-agent interaction behaviors. Thus, modeling military industrial logistics agent interaction protocol is one of core tasks.

To overcome the challenging problem on the mechanism of transforming conceptual model into computer simulation model automatically, a CID oriented modeling method for automatic generation of simulation codes is proposed. Based on the definition of graphical symbols of CID, the conceptual model of military logistics agent interaction protocol is constructed. By presenting the method of CID simulation model generation, the corresponding modeling tool is developed. With a case study on military logistics operation simulation, the interaction protocol is constructed and implemented. The proposed method and its feasibility and validity are verified.

Of course, the CID oriented modeling method has some details to be improved and perfected, including graphical representations and modeling tool. In the future, we will continue to improve the method centering on the graphical notation system and logic checking function to form a more general standard specification for agent interaction protocol modeling, so as to improve the efficiency and credibility of agent-based military modeling and simulation.

Table 4
Comparison of related methods and our method.

Methods	Representative work(s)	Overall description	Usability	Convenience	Efficiency	
					Development steps	VV&A steps
FSM	Gupta and Gallasch (2016); Lee et al. (2012); Winikoff et al. (2018)	A behavioral view form for system state description with graphical notations	More suitable for agent behaviors in agent-based military industrial logistics system model	Convenient to describe the conceptual model of agent state and state transitions	In the conceptual modeling stage, for each command behavior, an initiator and a participant with respective event lists need to be built. In this way, the FSM development step is twice as much as our method. In the simulation modeling stage, on the basis of building conceptual FSM model, it is necessary to write simulation model codes to support simulation experiments.	Iteration times of conceptual model verification: 1 Iteration times of conceptual model validation: 1 Iteration times of simulation model generation process verification: 4 ~ 5 Iteration times of simulation model generation process validation: 2 ~ 3 Iteration times of simulation model function verification: 6 ~ 7 Iteration times of simulation model execution validation: 5 ~ 6 Iteration times of model accreditation: 1 Total number of iteration times: 20 ~ 24
AUML	Belloni and Marcos (2004); Lee et al. (2012); Panescu and Pascal (2016); Terzidou et al. (2018); Winikoff et al. (2018)	An agent interaction view form in UML with graphical notations	More expressive for modeling agent activities and data flows inside agent-based military industrial logistics system model	Convenient to describe the conceptual model of agent state and state transitions, agent internal process, and interaction	In the conceptual modeling stage, at least one static diagram and one dynamic diagram should be constructed for each command behavior to describe the structural type and behavioral type, respectively. However, in our method, only one CID model is needed for each basic behavior. Thus, the AUML development step is more than twice that of our method. Similarly, in the simulation modeling stage, program codes need to be written after building the conceptual AUML model.	Same as the FSM modeling method
CPN	Dworzański and Lomazova (2013); Liu and Heiner (2013); Panescu and Pascal (2016)	A process diagram view form with graphical notations	Able to represent complex agent and interaction process semantics in agent-based military industrial logistics system model	Convenient to describe the conceptual model of agent state and state transitions, agent internal process, and interaction	In the conceptual modeling stage, initiators and participants can be designed into a CPN model for each command behavior. Its development steps are similar to our method. However, when constructing the corresponding simulation model, it is also necessary to write program codes.	Same as the FSM modeling method
CID	Our work	A CID oriented view form with graphical notations	Applicable to study agent state and behavior process, including agent behaviors, activities,	Convenient to describe the conceptual model of agent state and state transitions,	In the conceptual modeling stage, only one CID model is needed for each command behavior. In	Iteration times of conceptual model

(*continued on next page*)

Table 4 (*continued*)

Methods	Representative work(s)	Overall description	Usability	Convenience	Efficiency	
					Development steps	VV&A steps
		and corresponding modeling tool	data flows and interaction process semantics, in agent-based military industrial logistics system model with a function of logic verification by automatic generation of simulation codes	agent internal process, interaction, and correspondence relationship between state transition and interaction process	the simulation modeling stage, the program codes can be automatically transformed to execute simulation experiment based on the conceptual CID model.	verification: 6 Iteration times of conceptual model validation: 5 Iteration times of simulation model generation process verification: 1 Iteration times of simulation model generation process validation: 1 Iteration times of simulation model function verification: 1 Iteration times of simulation model execution validation: 1 Iteration times of model accreditation: 1 Total number of iteration times: 16

CRediT authorship contribution statement

Xiong Li: Conceptualization, Methodology, Formal analysis, Investigation, Writing - original draft, Writing - review & editing, Supervision, Project administration, Funding acquisition. **Wei Pu:** Software, Visualization. **Wei Zhang:** Data curation. **Xiaodong Zhao:** Validation.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

This research is supported by National Natural Science Foundation of China (grant number 61473311, 70901075), Natural Science Foundation of Beijing Municipality (grant number 9142017), and military projects funded by the Chinese Army (grant number 2016JJ22, 2014CX08).

References

Adediran, T. V., Al-Bazi, A., & dos Santos, L. E. (2019). Agent-based modelling and heuristic approach for solving complex OEM flow-shop productions under customer disruptions. *Computers & Industrial Engineering, 133*, 29–41.

Bai, B., Yoo, B., Deng, X., Kim, I., & Gao, D. (2016). *Linking routines to the evolution of IT capability on agent-based modeling and simulation: A dynamic perspective* (p. 184). Article ID: Computational and Mathematical Organization Theory.

Barros, L. (1997). A global view of industrial logistics. *Production Management, 4*, 150–158.

Barros, L. L., Riley, M., & Brown, D. (2001). Special millennium issue of the EJOR: A global view of industrial logistics. *European Journal of Operational Research, 129*, 231–234.

Bashir, R. S., Lee, S. P., Khan, S. U. R., Chang, V., & Farid, S. (2016). UML models consistency management: Guidelines for software quality manager. *International Journal of Information Management, 36*, 883–899.

Belloni, E., & Marcos, C. (2004). MAM-UML: An UML profile for the modeling of mobile-agent applications, in. In *the XXIV International Conference of the Chilean Computer Science Society* (pp. 3–13).

do Nascimento, N. M., de Lucena, C. J. P. (2017). FIoT: An agent-based framework for self-adaptive and self-organizing applications based on the Internet of Things. Information Sciences, 378, 161–176.

Dworzański, L. W., & Lomazova, I. A. (2013). CPN tools_assisted simulation and verification of nested petri nets. *Automatic Control & Computer Sciences, 47*, 393–402.

Furtado, B. A., & Eberhardt, I. D. R. (2016). A simple agent-based spatial model of the economy: Tools for policy. *Journal of Artificial Societies and Social Simulation, 19*, 12.

Gao, D., Squazzoni, F., & Deng, X. (2018a). The role of cognitive artifacts in organizational routine dynamics: An agent-based model. *Computational and Mathematical Organization Theory, 24*, 473–499.

Gao, D., Squazzoni, F., Deng, X. (2018b). The intertwining impact of intraorganizational and routine networks on routine replication dynamics: An agent-based model. Complexity, Article ID 8496235.

Gupta, A. K., & Gallasch, G. E. (2016). Equivalence class verification of the contract net protocol-extension. *International Journal on Software Tools for Technology Transfer, 18*, 685–706.

Hofmann, E., & Rüsch, M. (2017). Industry 4.0 and the current status as well as future prospects on logistics. *Computers in Industry, 89*, 23–34.

Hsieh, F.-S., Chiang, C. Y. (2009). Workflow planning in holonic manufacturing systems with extended contract net protocol, B.-C. Chien et al. (Eds.): IEA/AIE 2009, LNAI 5579, pp. 701–710.

Hu, X., Yang, J., Si, G., & Zhang, M. (2008). *Complex War System Simulation Analysis & Experiment*. Beijing: National Defense University Press.

Kang, J., & Sim, K. M. (2012). A multiagent brokering protocol for supporting grid resource discovery. *Applied Intelligence, 37*, 527–542.

Klitz, J. K. (1983). Simulation of an automated logistics and manufacturing system. *European Journal of Operational Research, 14*(1), 36–39.

Lee, J., Lee, S.-J., Chen, H.-M., & Wu, C.-L. (2012). Composing web services enacted by autonomous agents through agent-centric contract net protocol. *Information & Software Technology, 54*, 951–967.

Lesser, V. (1998). *Autonomous agents and multi-agent systems*. Boston: Kluwer.

Li, X. (2013). *Agent-based warfare modeling*. Beijing: National Defense Industry Press.

Li, X., Dong, Z., Wang, H., Meng, Q., Wang, Z., & Zhang, Y. (2012). A novel approach to selecting contractor in agent-based multi-sensor battlefield reconnaissance simulation. *International Journal of Computational Intelligence Systems, 5*(6), 985–995.

Li, X., Zhao, X., Pu, W., Chen, P., Liu, F., & He, Z. (2019). Optimal decisions for operations management of BDAR: A military industrial logistics data analytics perspective. *Computers & Industrial Engineering, 137*(106100), 1–12.

Liu, F., & Heiner, M. (2013). Modeling membrane systems using colored stochastic Petri nets. *Natural Computing, 12*, 617–629.

López-Ortega, O., & de la Cruz, K. L. (2010). A communication protocol among agents for exchanging data in the extended enterprise. *International Journal of Advanced Manufacturing Technology, 51*, 383–396.

Mishra, N., Kumar, V., & Chan, F. T. S. (2012). A multi-agent architecture for reverse logistics in a green supply chain. *International Journal of Production Research, 50*, 2396–2406.

Panescu, D., & Pascal, C. (2016). Holonic coordination obtained by joining the contract net protocol with constraint satisfaction. *Computers in Industry, 81*, 36–46.

Rachid, B., Mohamed, T., & Khouaja, M. A. (2018). An agent based modeling approach in the strategic human resource management, including endogenous and exogenous factors. *Simulation Modelling Practice and Theory, 88*, 32–47.

Reddy, R. H., Kumar, S. K., Fernandes, K. J., & Tiwari, M. K. (2017). A multi-agent system based simulation approach for planning procurement operations and scheduling with multiple cross-docks. *Computers & Industrial Engineering, 107*, 289–300.

Serrano-Hernandez, A., Faulin, J., Hirsch, P., & Fikar, C. (2018). Agent-based simulation for horizontal cooperation in logistics and transportation: From the individual to the grand coalition. *Simulation Modelling Practice and Theory, 85*, 47–59.

Shi, Z. (2001). *Intelligent agents and their applications*. Beijing: Science Press.

Smith, D., & Srinivas, S. (2019). A simulation-based evaluation of warehouse check-in strategies for improving inbound logistics operations. *Simulation Modelling Practice and Theory, 94*, 303–320.

Terzidou, T., Tsiatsos, T., & Apostolidis, H. (2018). Architecture and interaction protocol for pedagogical-empathic agents in 3D virtual learning environments. *Multimedia Tools & Applications, 77*, 27661–27684.

Winikoff, M., Yadav, N., & Padgham, L. (2018). A new hierarchical agent protocol notation. *Autonomous Agents and Multi-Agent Systems, 32*, 59–133.

Wooldridge, M., & Jennings, N. R. (1999). The cooperative problem-solving process. *Journal of Logic and Computation, 9*, 563–592.

Wu, X., & Liu, Q. (2008). *Equipment test and evaluation*. Beijing: National Defense Industry Press.

Yan, Z., & Pu, X. (2008). *Modern military sampling test method and its application*. Beijing: National Defense Industry Press.