Knowledge Systems for Coalition Operations KSCO 2017

November 6-8, 2017
Los Angeles, California, USA

A Semantic-Based Information Integration Framework of Agile Command and Control

Paper ID Number:
72

Authors:
Wen Zhou(National University of Defense Technology)
Weidong Bao(National University of Defense Technology)
Xiaomin Zhu(National University of Defense Technology)
Chao Chen(National University of Defense Technology)
Zhu Wu(Naval Command College)
Changfei Wu(Air Force Early Warning Academy)

Point of Contact:
Wen Zhou
zhouwen@nudt.edu.cn

Relevant Topic Areas:
Intelligent Command and Control (C2)
Coalition and Team Information Sharing
Coordination and Collaboration

A Semantic-Based Information Integration Framework of Agile Command and Control

Wen Zhou[a], Weidong Bao[a], Xiaomin Zhu[a], Chao Chen[a], Zhu Wu[b], Changfei Wu[c]
[a]College of Information Systems and Management, National University of Defense Technology, Changsha, China; [b]Naval Command College, Nanjing, China; [c]Air Force Early Warning Academy, Wuhan, China
Email: zhouwen@nudt.edu.cn

Abstract: Human society is in the period of accelerated development of the information revolution. It not only affects people's life extensively, but also constantly updates our understanding and understanding of the mode of operation in the information age. The ability to carry out effective information integration and provide intelligent information services, has become a joint combat process to obtain the command and control of the agile advantage of the support base. Based on this, this paper proposes a semantic-based information integration framework of agile command and control, which includes two parts: basic module and semantic analysis engine. The basic idea is to change the semantic relation from the pursuit of comprehensive information reasoning to the attention data, the idea of information retrieval into knowledge retrieval, relying on the information integration framework to tap the command information system implied in a variety of rich semantic relations, in order to carry out a higher level of information integration and provide more intelligent knowledge retrieval, and thus agile Command and control.

Key words: agile command and control; information integration; semantic analysis; graph database

1 Introduction

On the battlefield, to carry out agile distributed command and control(C2) for a force composed of a joint force of different countries or different arms, the first problem to be solved is how to realize the interoperability between the command information systems used by different command centers, and the core is the effective integration of information[1]. Besides, agile C2 also requires the command information system to achieve more intelligent knowledge retrieval, rather than the information retrieval has been.

With the development of Internet technology and the wide application of service-oriented technology, many information systems have gradually begun to integrate information according to service needs, and semantic information integration has also made great development. However, in order to achieve agile C2, we also need a higher level of information integration, to achieve more intelligent knowledge retrieval, and these are still some fundamental difficulties.

The first is the limitations of relational databases. Developers have been trying to use relational databases to deal with associated or semi-structured data sets, such as Vysniauskas[2] proposed a hybrid approach for correlating OWL 2 ontologies and relational databases, Chhaya[3] proposed to use D2RQ and Ontop to publish relational databases as associated data, which is a framework for answering SPARQL queries in relational databases[4]. Similar to the Ontop framework, there is GRAPHITE, which is a framework for implementing extensible graph traversal in relational database management systems[5], and Jindal[6] also proposed the use of vertica relational databases for graph analysis. However, as the outliers increase, the macroscopic structure of the data set becomes more complex and irregular, and the relational model will result in a large number of table joins, sparse rows, and nonempty checking logic. Increased connectivity in the world of relationships will translate into an increase in connection operations that will hamper performance and make it difficult for existing databases to respond to changing business requirements. Whether trying to model or correlate associations in a relational database, in addition to increasing the complexity of queries and calculations, it is necessary to deal with the pattern of the double-edged sword, and many times the pattern proved too rigid and fragile. In response to this problem, Cerans[7] proposed semantic refactoring of relational databases, resulting in a modeling framework for semantic databases such as SpiderMass[8], which combined with the social needs of social networks Extensive development, including semantic networks[9] and graph databases[10], such as Graphx[11], Mizan[12] and TAO[13] are very good graph database management systems.

Secondly, the problem of semantic databases defined by ontology modeling language. In order to improve the interoperability between different systems, such as in the standard data exchange format, ontology and consensus-based information model have made great progress. In terms of theory, not only the study of command and control ontology, but also Singapogu discussed the role of ontology in C2SIM[14], and Hansen proposed an information integration scheme based on ontology matching in time constraints[1]. In terms of applications, it includes the use of probabilistic ontology modeling methods to design terrorist decision-making support systems[15], the development of ontology on the hypothesis management of sea development[16], and the mission and means to achieve military assets and mission objectives[17]. However, while ontology is widely accepted, different information systems adopt a variety of different ontology modeling languages, such as Web ontology language, where the interaction between them and the different grammar rules will cause the developer to pay a high cost[18]. And, in order to cope with the adaptability of data requirements, which need to build the semantic library through the ontology modeling language must develop a variety of applications and interfaces[19], where the definition of mapping rules and application re-design and development is a very tedious work[20].

Finally, in order to achieve the perception of the user's intention and enhance the intelligence of the information system, one of its core is the understanding of semantic information system, and there are still a lot of technical problems need to be resolved, such as natural language processing, semantic and artificial Intelligent and so on. In terms of natural language processing, including word sense disambiguation[21, 22], identification of entities and relationships[23], extraction of syntactic and semantic features[24-26]. In semantics, including ontology-based semantic search engine development, common sense and semantic reasoning[27-29] and so on. And the key is to change the concept of information retrieval, that is, by the information retrieval and keyword matching into knowledge retrieval, and the use of semantic and artificial intelligence to achieve self-learning[30], but for now we need to do more in-depth study.

In order to improve the interoperability of information systems and achieve a higher degree of agile C2, we propose a semantic-based information integration framework of agile C2 for the higher level of information integration of command information systems. We construct the middleware by using the Extract Transform Load(ETL) method based on subgraph to realize the transformation from the relational database to the graph database. The core is to realize the deeper and more efficient processing of the associated data by digging the data connection. In order to provide the domain knowledge support for the search query of the database, we also develop the semantic analysis engine to realize the knowledge retrieval based on semantics and further improve the intelligence of the retrieval, and then improve the retrieval of the database, so as to provide users with more intelligent, personalized and professional information services.

The rest of this article is arranged as follows. In Section 2, we introduce a semantic-based information integration framework of agile C2. Section 3 describes the basic modules in the information integration framework, including the relational database to map database conversion, domain model building and syntax analysis. Section 4 focuses on the core of the information integration framework -- the composition of the semantic analysis engine, and an example is described. Section 5 is the conclusion.

2. Semantic-based information integration framework of agile C2

In the process of joint operations, in order to achieve a higher degree of agile C2, we not only need to achieve interoperability between different information systems, and we also need information systems to understand the user's intent to achieve semantic-based knowledge retrieval, and further enhance the retrieval of intelligent to provide users with more intelligent, personalized and professional information services. At present, to achieve this goal also has many difficulties and challenges, especially information integration is the foundation. To solve this problem, we propose a semantic-based information integration framework of agile C2, as shown in Fig. 1.

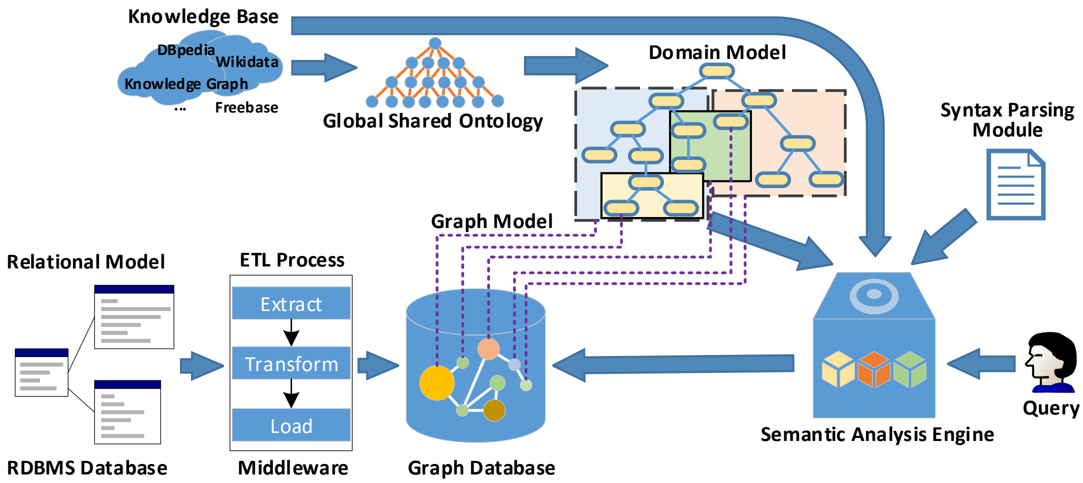

Fig.1 Overall system architecture

In terms of composition, we include two basic parts of the basic module and the semantic analysis engine in our proposed framework. Among them, the basic module includes middleware, a graph database, a domain model and a parsing module for converting the relational database into a graph database, and the semantic analysis engine includes a universal package module, an extension semantic module, a matching definition module and a search module constitute.

In terms of process, our framework mainly includes four processes. First, we develop the relational database into the graph database through the middleware. Secondly, we develop the domain model and establish the mapping rules of the domain model and the graph database. Then we introduce the syntax parsing module of natural language processing, and implement the structured processing of the query. The Finally, we develop a semantic analysis engine to achieve semantic-based knowledge retrieval.

3. Basic module

3.1 Middleware and graph database

With the advent of large data age, NoSQL database has seen rapid development. Graph Database is one of the NoSQL family and one of the most developed database technologies since 2013. Common database systems include AllegroGraph, DEX, HyperGraphDB, InfiniteGraph, Neo4j, and so on[10]. Some relational database management systems also began to support graph data, such as Oracle to increase the space of large data and data support. It can be said that the database database technology is one of the most popular research topics.

In the aspect of multi-level relationship, shortest path, PageRank and so on, the graph database can adopt the matching algorithm of graphs, and the iterative level is less in the node relation query process. The query efficiency is obviously better than the relational database, especially when the data is large, the advantage is more obvious.

Although the semantics can be described to a certain extent by the graph, and the description and understanding of semantics occupy an important position in the C2 process of joint operations. However, the existing data is stored in the form of relational data, In the form of organizational data, between the table through the primary key - foreign key for the association, the way is simple. In contrast to this, the graph data is stored in the form of graphs, nodes and edges are the basic representation elements of the graph, and the data representation is more complex[31]. Therefore, migrating from a relational database to a graph database is a practical solution. How to achieve the relationship between the data to the map data conversion is to achieve the existing application to the data application transformation key. The problem to be solved in this paper is how to improve the quality of the converted graph data in the process of ETL relational data to graph data, and to convert the relational data into graph data efficiently and efficiently.

ETL is actually a process of describing data extraction, conversion, and loading of data from source data. Common tools include Informatica, Datastage, OWB, MS DTS, Beeload, Kettle and so on[31]. ETL quality is mainly measured from the aspects of correctness, completeness and consistency. Because traditional ETL tools such as Informatica and Kettle are mainly used for the research and practical application of relational data to relational data extraction, transformation and conversion from relational data to relational data using ETL. However, in the use of ETL to achieve from the relational data into graph data, due to the relatively late development of the database, and the standard is not uniform, so the research and application is relatively small. Now the main problems include the quality of the converted graph data is not high, when faced with complex relational database conversion efficiency is not high, the conversion results are not conducive to distributed storage.

Therefore, by transplanting some successful algorithms in traditional ETL, such as drawing some successful methods of ETL under the framework of Spark GraphX, we adopt a method of efficient batch data extraction, parallel conversion and batch loading. GraphX is the development of Apache distributed graph calculation framework, which is mainly to solve the problem of distributed computing. It provides ETL, which provides data extraction, cleaning, conversion, loading tools and exploratory analysis, and through a single system for iterative calculations. In addition, it can view the same data, graphics and collections, through RDDS efficient transformation, connecting graphics, and can be customized through the Pregel API iterative algorithm. It is comparable to the fastest professional-grade graphics processing system while preserving flexibility, fault tolerance and ease of use.

In this paper, we use the ETL method based on subgraph proposed in [31] to construct the middleware, including six steps.

(1) Load relational database schema: read the relational database schema definition information, its table definition, the relationship between the definition were loaded into the corresponding list.

(2) Establish the relational database model corresponding to the model diagram: that is based on the relational database table definition, relationship definition information to build the relationship model.

(3) Split the relationship between the database model for a number of sub-mode: traversal mode diagram, get the corresponding set of points C, according to the relationship between the model will be split into several sub-mode.

(4) Loop processing sub-mode: parallel pairs of sub-mode ETL, according to the sub-mode of the main table definition information, bulk loading of the main table records, the relationship between these records to map data conversion. And then according to the sub-mode of the relationship between the table, followed by loading the other table data, and the relationship between the data to the map data conversion. Finally, we get the subpattern pattern and subgraph data corresponding to the subpattern.

(5) Optimize the conversion results: the use of indexing, merging nodes, mergers and relationships, etc., to reduce the complexity of graph data and improve the efficiency of data query.

(6) Store the conversion results, the conversion results will be stored in bulk to the database.

In addition, distributed storage can be used as a choice for large scale data storage management. In this paper, the distributed graph data analysis method is based on the distributed storage of graph data. Through the design and management of control procedures, optimizing and integrating the database storage technology and graph data distributed computing framework, we finally unified the distributed data storage and distributed computing.

3.2 Domain model

In order to construct the domain model, we refer to the idea of the ontology architecture, as shown in Fig. 2, in the reference [18, 32]. We use not only domain experts to build domain models, but more importantly, we use the common Joint C3 Information Exchange Data Model(JC3IEDM)[33], and combined with different tasks to guide the domain model to create different domain ontology. Moreover, in order to make the domain model of each component can be compared with each other, we allow each component has its own semantic description on the basis of each component is required from the shared ontology The This approach allows us to overcome the comparative problems in multiple ontology methods and also alleviate the shortcomings of a single ontology approach that is susceptible to changes in source information.

In addition, our global shared ontology comes from the open knowledge base, such as DBpedia, Knowledge Graph, Wikidata, Freebase and so on. They are an important complementary to JC3IEDM and other data models, whose main role is to improve the command control information integration process of data interoperability. Take DBpedia as an example. DBpedia's concepts and relationships to facilitate data interoperability in command information integration. DBpedia is Wikipedia, which is a knowledge base of human knowledge resources, maintained by a large number of people. The project balances this huge knowledge resource by extracting structured data from Wikipedia's entries in a more efficient way. Based on Wikipedia datasets, DBpedia allows users to query complex issues and link other data sets to other Wikipedia datasets. At present, DBpedia describes more than 340 million events, of which 1.5 million have been classified as ontology.

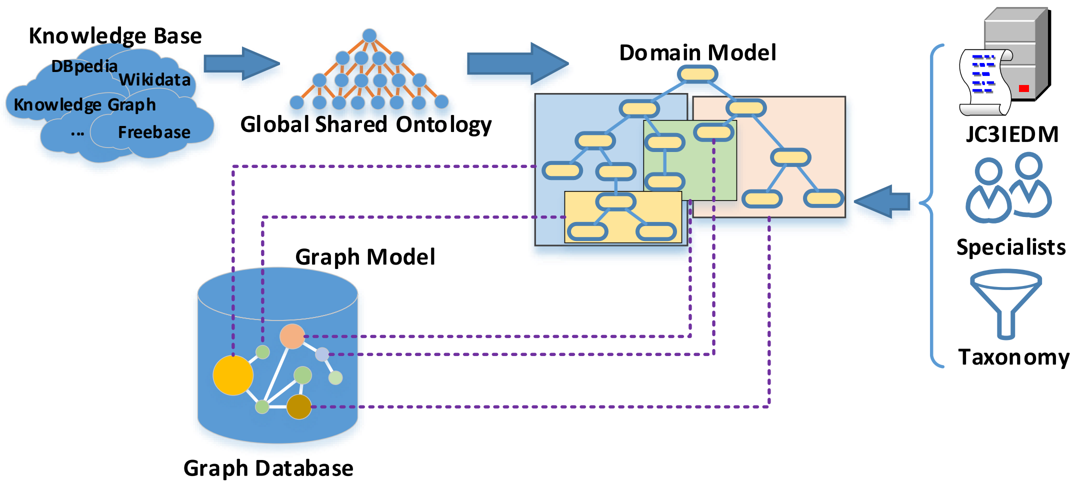

Fig.2 Domain model architecture

It should be noted that DBpedia storage format is RDF triple, and RDF is proposed by the W3C to describe the network resources standards. It uses a simple description of the way, that is, the subject (subject), predicate (predicate), object (Object) composed of triples to represent the resources. The RDF data general representation is a (S, P, O) triplet, and a set of RDF data can form a RDF directed graph. The RDF graph can be represented by a tagged node and a tagged edge, where each triple corresponds to a "node-edge-node" subgraph on the graph and it states the relationship between the subject and the object of the object as expressed by the predicate. The node of a RDF graph is the subject and object of all the triples it contains, and the direction of the edge always points to the object. RDF graphs can usually be viewed as a directed marker.

In order to facilitate the construction of the domain model, and to achieve docking with the map database, we use the map database Neo4j on the collection of DBpedia RDF triples stored. Neo4j is called a property graph, including vertices, relationships, and properties[34]. The storage of data is mainly divided into three types of data storage, such as nodes, relationships, nodes or relational properties. Whether it is a vertex or relationship, can have any number of properties, the property is similar to a hashMap storage. Neo4j focuses on the performance of a large number of connection queries when the performance of the problem. In addition, Neo4j also provides a very fast graphical algorithm, recommended system and OLAP style analysis, it can be said, Neo4j is a high performance, high reliability, scalable, fully compatible ACID map database[34]. Since each Node, Relationship, and Property in Neo4j are stored independently and follow the natural order, if you want to find the corresponding node in the diagram, you must rely on the index. Neo4j mainly provides Lucene-based Full-text index mechanism to achieve the search for nodes and relationships. The RDF data storage model diagram based on the graphics database Neo4j is shown in Fig. 3.

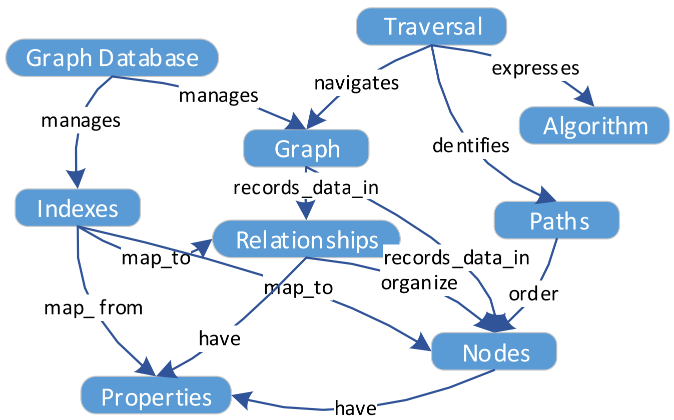

Fig.3 RDF data storage model based on graphic database Neo4j[34]

3.3 Syntax parsing module

The syntax parsing module is the basis of semantic retrieval. After receiving the user to retrieve the task, the first need for the user's search task for syntactic analysis, which belongs to the natural language processing field of information extraction problem. The joint extraction of entities and relations as a key task of information extraction, the implementation method can be simply divided into two categories: one is the series extraction method. The other is a joint extraction method. The concatenation method divides the problem into two sub-tasks in series, that is, the entity extraction model is used to extract the entity, and then the relational extraction model is used to get the relationship between the entity pairs. The advantage is that it is easy to optimize the entity recognition task and the relationship extraction task, but the disadvantage is that they are aimed at obtaining the intermediate product (entity or relationship type) of the triplet, and the result of the entity recognition will further affect the result of the relationship extraction, resulting in error accumulation[35]. Unlike the concatenation method, the joint extraction method uses a model to extract entities and their relationships simultaneously, enabling better integration of information between entities and their relationships. However, there are many problems with the existing joint extraction methods, such as: most of the joint extraction model requires manual participation in the construction of features; based on the end to end of the joint extraction model, because in the process of modeling the extraction of entities and their relationship between the information Redundancy and other issues[35].

In order to realize the automatic labeling of the sequence of text words, we use the end-to-end model proposed in [35]. The model structure is shown in Fig. 4. In this model structure, it contains a bi-directional long memory (Bi-LSTM) layer for encoding the input statement and the LSTM-based decoding layer with offset loss. Deviations can increase the relevance of physical labels. The word embedding layer transforms the one-hot representation vector of each word into a low-dimensional dense word embedding vector (dimension 300), and the Bi-LSTM encoding layer (the number of layers is 300) is used to obtain the coding information of the word, the LSTM decoding layer (the number of layers is 600) is used to generate the tag sequence. We add offset loss to enhance the relevance of the entity tag.

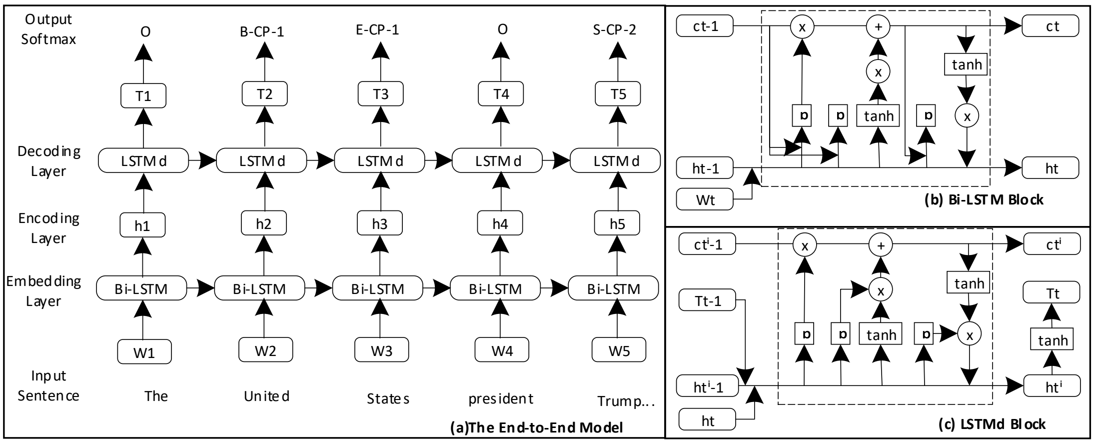

Fig.4 An End-to-End model to produce tags sequence[35]

4. Semantic analysis engine

4.1 Semantic Analysis Engine Architecture

The ultimate goal of developing a semantic analysis engine is to achieve semantic-based knowledge retrieval. Semantic analysis engine is the core component of our proposed semantic-based information integration framework of agile C2, which includes five groups, including crowdsourcing module, extension semantic module, external reference library, matching definition module and search module, as shown in Fig. 5.

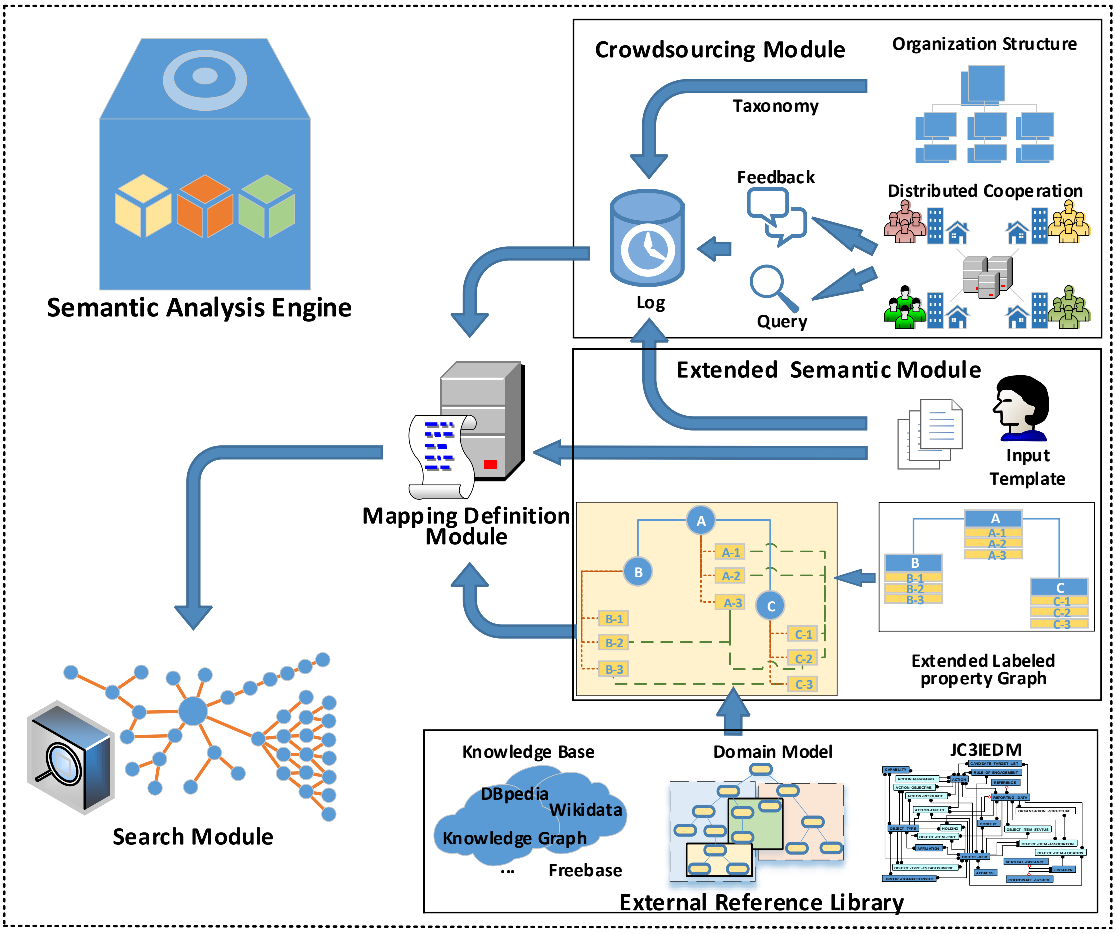

Fig.5 Semantic analysis engine architecture

The semantic analysis engine's workflow is as follows:

The first is to expand the semantic module. As the syntax analysis module has extracted the entity and the relationship from the user's query task, in the semantic analysis engine, we need to further extract the extracted entities and relationships as the extended labeled property graph. Its main function is to enter the subgraph as a matching definition. In the process of modeling the extended labeled property graph, it requires the user to enter or define a template, which is mainly used to define the query question and focus of the user. At the same time, we can also enter or identify property information related to a specific field in the template, and this information is recorded in the log.

The second is an external reference library. It consists of open knowledge base, domain model and JC3IEDM, in which open knowledge base provides a wide range of factual knowledge, and domain model and JC3IEDM provide C2 domain knowledge. Because the extended labeled property graph needs to be guided with task-related knowledge in the process of modeling, and it also needs to be provide constraints in the search process, the external reference library can provide extended knowledge and constraints as effective guidance.

The third is the crowdsourcing module. It is mainly based on the log for the next step in the definition of matching to provide more information available, which is to further realize the semantic retrieval of an important basis for intelligence. In the crowdsourcing module, we will record and organize the system in the user's query records and feedback, on the other hand we will record the user's organizational structure and other related information and label information. This information can provide assistance for more accurate matching of user search intentions. In addition, the template information of the user query and the feedback to the final search result are automatically recorded in the log.

The fourth is the matching definition module. Its main function is to define the subgraph and ranges that need to be matched for the next search, with the purpose of providing a context for the search. It aggregates the information in the crowdsourcing module and the extended semantic modules and converts them into subgraph with matching breadth and depth constraints that need to be matched.

The last part is the search module. The main function of this module is to connect the semantic analysis engine and the graph database, its goal is to convert the subgraph with search breadth and depth constraints into the query language of the graph database and search the database in the graph database.

In the semantic analysis engine, how to model the expansion of the labeled property graph is the foundation. Here we focus on the process of the proposed labeled property graph.

The extended labeled property graph is based on the labeled property graph, and it is the concept of the graph model in Neo4j. The labeled property graph is made up of nodes, links, properties, and labels. Among them, the node contains properties that can exist in any form of key-value pairs. In Neo4j, the key is usually a string, the value can be Java string and primitive data, or an array of these data types. The nodes can be tagged with one or more tags, which label the nodes together and represent their roles in those data[34]. Links connect nodes, thus forming the graph. Each link has a direction, a name, a start node, and an end node. The direction and name of the link make the structure of the node rich in semantics. And the link can also have properties, by adding properties on the link, you can provide metadata to graph algorithms, also can add additional semantics (including features and weights). In addition, it can also be used for runtime constraint query. Because the labeled property graph can provide these advantages of semantics, our modeling process with it as the core foundation. However, since the properties of the nodes and links in the labeled property graph exist only as semantic metadata, the semantic metadata itself has a very rich intrinsic relationship, especially when combined with user intent and contextual information. Therefore, in order to make full use of the content of these semantic links, we expand the labeled property graph, relax the constraints on the properties of nodes and links, and establish links between nodes and associated properties based on user intent and contextual information.

An example of modeling the expanded labeled property graph is shown in Fig. 6. The user's input template provides user intent information, the external reference library provides the basis for property association and label selection, and the crowdsourcing module provides contextual information for the extended labeled property graph modeling. It should be noted that the expanded labeled property graph contains a mechanism of learning mechanism, that is, through structure learning, property learning, path learning, rule learning and multi-modal learning to establish an embedding learning mechanism in the process of modeling. And then we can use this self-learning process to continuously strengthen the function of the expanded labeled property graph.

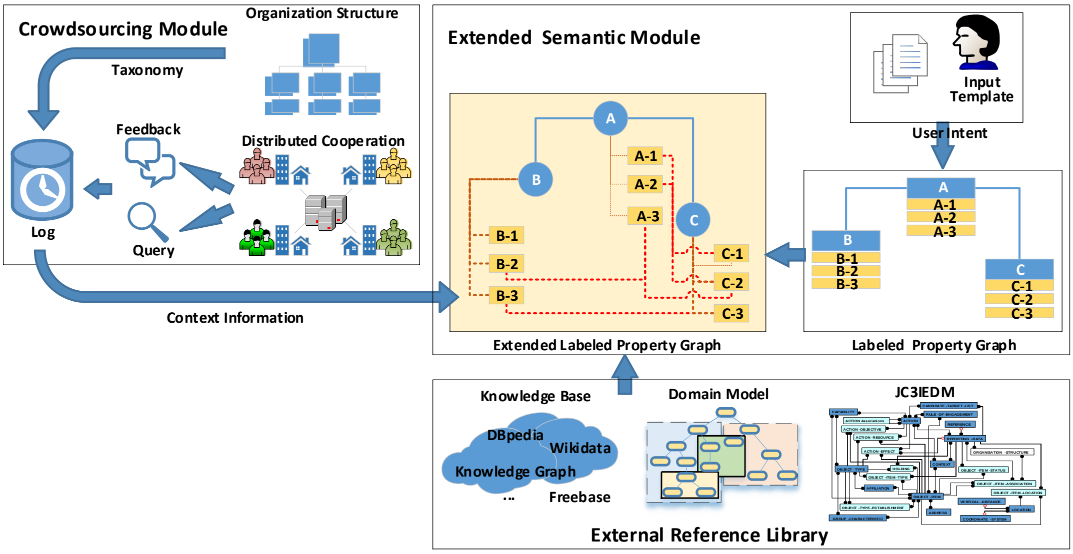

Fig.6 An example for the modeling of extended labeled property graph

In Fig. 6, we firstly need to model the user intent information according to the modeling process of the labeled property graph in Neo4j. There are three nodes A, B, and C, and each node has three properties information. Then we use the external reference library and crowdsourcing module with related knowledge and contextual information to provide more links between the properties of the nodes and links in the labeled property graph. For example, we can establish an association between the B-2 property information of the Node B and the A-3 property information of the A node.

4.2 Example

As an example, we assume that there is a user's input template which actually represents a semantic search scenario for resource in a joint combat C2 process: At T + K, the troop A arrives at the C zone via the B zone and, in conjunction with the troop E, strikes against the combatant D, and the troop E locates in the F zone.

How do you semantically model the C2 intent reflected in the semantic search scenario for resources? According to the information integration framework proposed in this paper, the core is to design the C2 intention of the user input template using the extended labeled property graph, and then provide a richer implicit semantic information service. For example, it can answer the question in the user's input template that what impact factors affect the troop A with the troop E to implement combat operations? That is, in the troop A and E operations, what factors that affect collaboration they need to consider(such as regional location, level of equipment, combat effectiveness, environmental factors, etc.)? And what is the relationship between these factors?

We first express the C2 intent as:

- Troop A passes through zone B.
- Troop A arrive at the C zone.
- Troop A is with Troop E.
- Troop A strike at the combatant D.
- Troop E strike at the combatant D.
- Troop E locates in the F zone.

Fig. 7 shows the basic modeling results for the above C2 intent.

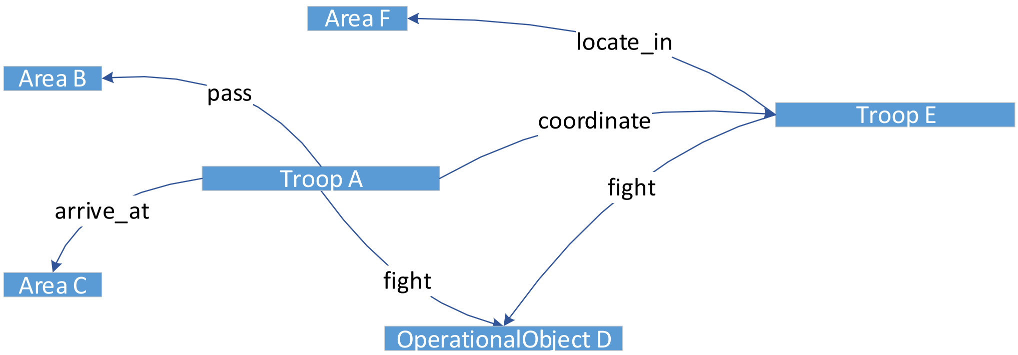

Fig.7 An example for semantic search (the basic modeling result)

On the basis of Fig. 7, we need to use the external reference library and the crowdsourcing module to exploit richer C2 semantic information. The external reference library provides the basis for property association and label selection, the crowdsourcing module provides contextual information for the extended labeled property graph modeling, and the user's input template also provides questions and focus about the C2 intent and other relevant information. We use the expanded labeled property graph to model. That is, we link the properties and select labels according to the C2 domain knowledge in the domain model and JC3IEDM in the external reference library. And use the contextual information in the crowdsourcing module and the open knowledge in the external reference library to automatically add more property information and potential links between different properties for the entities.

It should be noted that our model relies on the user input template, domain model and JC3IEDM in the external reference library to determine the focus of the model and the breadth and depth of the relationship extension, thus forming a final available subgraph for searching and performing a search in the graph database. Fig. 8 is the final result of modeling using the extended labeled property graph proposed in this paper.

5 Conclusion

The joint operation of the information age requires C2 to be agile, and effective information integration is the basis for supporting the C2 agility[32], and it further requires the command information system to provide intelligent information services such as knowledge retrieval. Although so far there have been many different solutions, including the use of semantic web ideas in the process of information integration to achieve reasoning function. But we found in practice that there are still many difficulties in implementing the reasoning in the process of information integration. Therefore, we try to set our goal to carry out a higher level of information integration, to provide more intelligent knowledge retrieval, and then to achieve agile command.

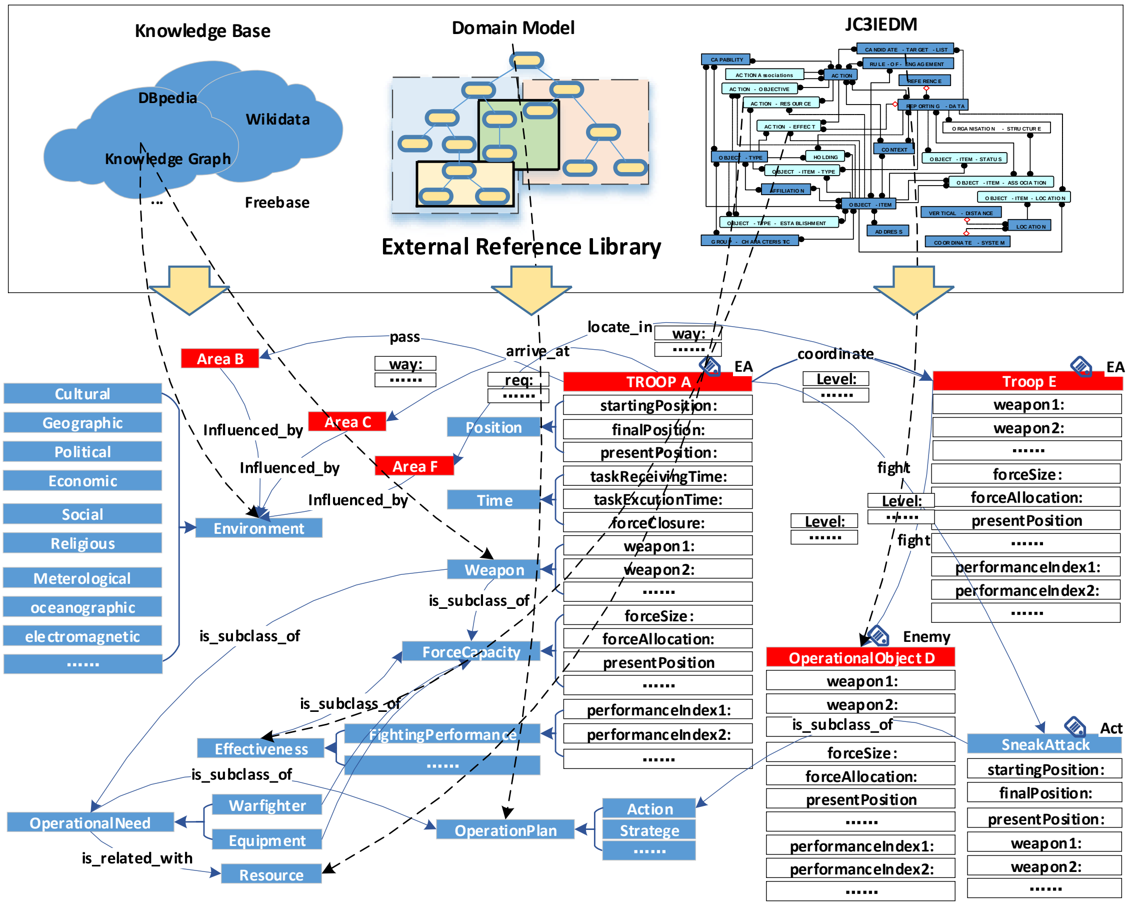

Fig.8 An example for semantic search (the final modeling result)

To this end, this paper presents a semantic-based information integration framework of agile C2. The core includes two aspects. First, we convert the relational database into a graph database to facilitate semantic-based knowledge retrieval. The second is that we construct the crowdsourcing module and the extended the semantic module in the semantic analysis engine. Moreover, we introduce an external reference library and use the open knowledge and C2 domain knowledge in the external reference library, together with the contextual information provided by the crowdsourcing module, to explore the richer semantic links from the C2 intent, and to provide users with more intelligent, personalized and professional information service.

It should be noted that, according to the classification method proposed in the reference [36], our goal is to establish a framework that can integrate data model, expression rule, knowledge reasoning and semantic retrieval from the perspective of vocabulary, grammar and ontology. This framework has adaptability, extensibility and versatility. We hope that this framework will facilitate communication, address the ambiguity of needs and understanding, and provide help for the proper realization of semantic-based agile C2 information integration. However, there is no in-depth consideration of agreements and services in our proposed framework, and implementation and testing tools are not yet fully implemented, and these are an important part of our future work.

In addition, in our framework, the external reference library plays an important role, especially JC3IEDM, which provides not only the support of the establishment of the C2 domain model, but also the basis of liking properties and selecting labels for the extended labeled property graph. However, JC3IEDM is stored on a traditional relational database which doesn't support semantic retrieval well, and it doesn't contain rich semantics. Moreover, it can't represent all the terms in the field of operations and cover the whole relationship among them[37]. Therefore, we support good semantic retrieval through graph database, and further enrich the semantic information in our model through the contextual information provided by open knowledge base and crowdsourcing module. One of our future work is to improve the semantic analysis engine in the C2 information integration framework by establishing a more effective context-aware mechanism, improving the self-learning mechanism of the extended labeled property graph, and regulating the mechanism of user input template.

Acknowledgements

This work is supported by National Natural Science Foundation of China (NSFC No.71702186), the Scientific Research Project of National University of Defense Technology under grant ZK16-03-57. We thank all anonymous reviewers for their constructive comments.

References

[1] B. J. Hansen. Towards Ontology Matching Suitable for Information Integration in Time-Critical Situations. DTIC Document, 2011.

[2] E. Vysniauskas, L. Nemuraiteand A. Sukys. A hybrid approach for relating OWL 2 ontologies and relational databases. *International Conference on Business Informatics Research*. Springer, 2010: 86-101.

[3] P. Chhaya, K. Leeand K. Shin et al. Using D2RQ and Ontop to publish relational database as Linked Data. *Ubiquitous and Future Networks (ICUFN), 2016 Eighth International Conference on*. IEEE, 2016: 694-698.

[4] D. Calvanese, B. Cogreland S. Komla-Ebri et al., "Ontop: answering SPARQL queries over relational databases," *Semantic Web*,vol. 8,no. 3, pp.471-487,2017.

[5] M. Paradies, W. Lehnerand C. Bornhövd. GRAPHITE: an extensible graph traversal framework for relational database management systems. *Proceedings of the 27th International Conference on Scientific and Statistical Database Management*. ACM, 2015: 29.

[6] A. Jindal, S. Madden, M. Castellanosand M. Hsu. Graph analytics using vertica relational database. *Big Data (Big Data), 2015 IEEE International Conference on*. IEEE, 2015: 1191-1200.

[7] K. Cerans, G. Barzdinsand G. Bumans et al., "Relational Database Semantic Re-Engineering Technology and Tools," *Baltic Journal of Modern Computing*,vol. 2,no. 3, pp.183,2014.

[8] R. Winkler, "SpiderMass: semantic database creation and tripartite metabolite identification strategy," *J MASS SPECTROM*,vol. 50,no. 3, pp.538-541,2015.

[9] J. F. Sowa. *Principles of semantic networks: Explorations in the representation of knowledge*. Morgan Kaufmann, 2014.
[10] C. Vicknair, M. Maciasand Z. Zhao et al. A comparison of a graph database and a relational database: a data provenance perspective. *Proceedings of the 48th annual Southeast regional conference*. ACM, 2010: 42.
[11] R. S. Xin, J. E. Gonzalez, M. J. Franklinand I. Stoica. Graphx: A resilient distributed graph system on spark. *First International Workshop on Graph Data Management Experiences and Systems*. ACM, 2013: 2.
[12] Z. Khayyat, K. Awaraand A. Alonazi et al. Mizan: a system for dynamic load balancing in large-scale graph processing. *Proceedings of the 8th ACM European Conference on Computer Systems*. ACM, 2013: 169-182.
[13] N. Bronson, Z. Amsdenand G. Cabrera et al. TAO: Facebook's Distributed Data Store for the Social Graph. *USENIX Annual Technical Conference*. 2013: 49-60.
[14] S. S. Singapogu, K. Guptonand U. Schade, "The Role of Ontology in C2SIM," *21st ICCRTS*,vol.,2016.
[15] R. Haberlin, P. C. Da Costaand K. B. Laskey. Probabilistic ontology architecture for a terrorist identification decision support system. DTIC Document, 2014.
[16] R. Haberlin, P. C. Da Costaand K. B. Laskey. An Ontology for Hypothesis Management in the Maritime Domain. DTIC Document, 2011.
[17] P. H. Deitz, J. R. Michaelis, B. E. Brayand M. A. Kolodny. The Missions & Means Framework (MMF) Ontology: Matching Military Assets to Mission Objectives. *ICCRTS Conference*. 2016.
[18] J. Park, E. K. Leeand Q. Wang et al. Health-connect: An ontology-based model-driven information integration framework and its application to integrating clinical databases. *Information Reuse and Integration (IRI), 2012 IEEE 13th International Conference on*. IEEE, 2012: 393-400.
[19] P. Mitra, N. F. Noyand A. R. Jaiswal. Omen: A probabilistic ontology mapping tool. *International Semantic Web Conference*. Springer, 2005: 537-547.
[20] N. Choi, I. Songand H. Han, "A survey on ontology mapping," *ACM Sigmod Record*,vol. 35,no. 3, pp.34-41,2006.
[21] A. Moro, A. Raganatoand R. Navigli, "Entity linking meets word sense disambiguation: a unified approach," *Transactions of the Association for Computational Linguistics*,vol. 2, pp.231-244,2014.
[22] S. Cucerzan, "Large-scale named entity disambiguation based on Wikipedia data,",2007.
[23] J. G Enríquez, V. Lee, M. Goto, F. J Domínguez-Mayoand M. J Escalona. Entity Identification Problem in Big and Open Data. *Proceedings of the 17th International Conference on Enterprise Information Systems-Volume 1*. SCITEPRESS-Science and Technology Publications, Lda, 2015: 404-408.
[24] E. Gabrilovichand S. Markovitch, "Wikipedia-Based Semantic Interpretation for Natural Language Processing," *arXiv preprint arXiv:1401.5697*,vol.,2014.
[25] S. Li, Z. Yang, J. Xiuand C. Liu. Research on an intelligent semantic-based information extraction framework. *Communications and Information Technologies (ISCIT), 2016 16th International Symposium on*. IEEE, 2016: 676-680.

[26] N. Kambhatla. Combining lexical, syntactic, and semantic features with maximum entropy models for extracting relations. *Proceedings of the ACL 2004 on Interactive poster and demonstration sessions*. Association for Computational Linguistics, 2004: 22.
[27] H. Jaeger, "Artificial intelligence: Deep neural reasoning," *NATURE*,vol. 538,no. 7626, pp.467-468,2016.
[28] A. Hunter, "A probabilistic approach to modelling uncertain logical arguments," *INT J APPROX REASON*,vol. 54,no. 1, pp.47-81,2013.
[29] E. Davisand G. Marcus, "Commonsense reasoning and commonsense knowledge in artificial intelligence," *COMMUN ACM*,vol. 58,no. 9, pp.92-103,2015.
[30] R. S. Michalski, J. G. Carbonelland T. M. Mitchell. *Machine learning: An artificial intelligence approach*. Springer Science & Business Media, 2013.
[31] Q. Ding. Research on Data Extraction and Distributed Graph Data Management. Yunnan University, 2016.
[32] A. DeFrancescoand B. McQueary. A Semantics-Based Approach to Schema Matching and Transformation in Network Centric Environments. *15st ICCRTS*. 2010.
[33] Multilateral Interoperability Programme (MIP)[EB/OL][2017-09-15]. https://public.mip-interop.org/Pages/Default.aspx.
[34] J. Webber. A programmatic introduction to neo4j. *Proceedings of the 3rd annual conference on Systems, programming, and applications: software for humanity*. ACM, 2012: 217-218.
[34] J. Tangand Z. Luo, "Research on RDF data storage based on graph database Neo4j," *Information Technology*,vol.,no. 6, pp.115-117,2015.
[35] S. Zheng, F. Wangand H. Bao et al., "Joint Extraction of Entities and Relations Based on a Novel Tagging Scheme,",2017.
[36] Huang H, Li X, Wang X. "Key Issues in C-BML: A Survey," *Journal of Command and Control*, vol.1, no.3, pp.241-253,2015.
[37] Schade U, Biermann J, Frey M, et al. "From battle management language (BML) to automatic information fusion," *Information Fusion and Geographic Information Systems*, vol., no.10, pp.84-95,2007.