Syntax Rule Construction of BML-based Command and Control Information

Si Li[*,1], Xin-zhong Li[2], Jie Liu[1], Yuan Wang[2] and Shen-yang Liu[1]

[1]*Department of Equipment Command & Management,* Mechanical Engineering College, Shijiazhuang, Hebei, 050003, P.R. China

[2]*Department of Scientific Research*, Shijiazhuang Army Command College, Shijiazhuang 050000,China

***Abstract* — Given the low efficiency in interoperability between the command information system and the simulation system for equipment support, this paper does research on the command and control information in the equipment support command information system and adopts the BNF form to establish the syntax rules for the command information by analyzing the BML systematically from the perspective of syntax and semantics. Finally, the validity of the method is verified through the simulation experiment. Based on this thought, this paper further explores how to solve the low efficiency in interoperability between the information system and the simulation system for equipment support command.**

***Keywords* — battle management language; command and control information; BNF form; syntax rules**

I. INTRODUCTION

The pilot-in-the-loop equipment support simulation system is in essence a process where the interactive information and mutual responses between command entities and executive entities are simulated and realized through the interconnection of the command platform (command information system) and the execution platform (simulation system for virtual military forces). In the process, it remains an essential issue how the traditional interactive information pattern --- "free text" is translated into the normalized description pattern that can be identified and processed by the computer simulation system. However, the current command information system only processes the "free text" as an individual data field instead of directly translating the essential information elements into the data structure required in the simulation model. Therefore, professionals translate large quantities of "free text" into the simulation run language item by item to ensure the successful simulation run and the workload is burdensome. Moreover, such structuralized command simulation information is limited in universality because it is designed for the specific simulation system, thus hindering the development of the command training simulation system.

The use of Battle Management Language (BML) can effectively improve the specification description of "free text" for the equipment support command information, interactive information description in the equipment support simulation process. Based on the BML technique system, in this paper, we not only establish the syntactic analysis pattern of the command and control information, but also build the syntactic rule model in the equipment support command information system. In addition, in this paper, we also provide solutions to improve the interconnection and the interoperability between the command information system and the simulation system for equipment support.

II. BNF FORMAL SYNTACTIC ANALYSIS OF BML

A. *Introduction to BML*

Battle Management Language (BML) is defined as an unambiguous language to command and control forces and armament conducting military operations and to provide situational awareness and a shared, common operational picture, which is shown in Fig. 1.

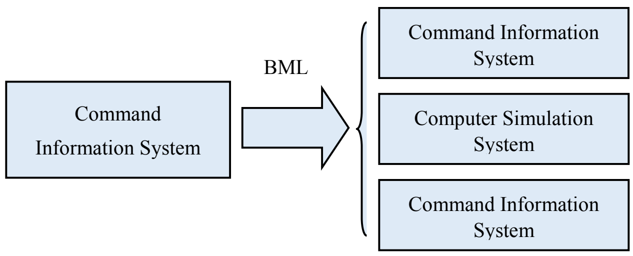

Figure 1. Basic idea of BML

Battle Management Language (BML) is not a new concept and it originates from EAGLE Battle management Language (EAGLE BML) and Command and Control Simulation Interface Language (CCSIL) in the simulated battlefield project in 1990. The US army officially started the BML research from 1998 and established Level 1 Model for Battle Management Language (BML-1) in 1999, which has got continuously updated so far. The concept of BML is not limited to the use of simulated power or real power and it can be applied to the command and control power as well as the equipment involved in all military actions.

BML is intended to translate 5ws into the official concept with the following functions:

(1)Describe the announced order, report, and the situation awareness unambiguously;

(2)Exchange between human, machine, and the simulation system;

(3)Provide the normalized data description using the data model for the command and control information exchange;

(4)Be consistent and compatible with Military Scenario Definition Language;

(5)Support military action plans, execution, and summary in all stages;

(6)Support multi-troops and even multinational ordinance;

(7)Provide the correct and unambiguous glossary for the multinational C2 system;

(8)Provide the operation specification of message distribution for reference.

In terms of the actual system development, the command and control information, whether they are orders or reports, can be briefly classified into 5Ws, but the simple classification can not be identified by the simulation system or activate the running of the simulation model. For example, WHO contains the order issuer and the order executor in terms of order and WHEN covers the starting time and the finishing time of the task. Such orders require the further analysis of the given information. Therefore, the formal syntax is essential to the BML design which can be automatically processed by the computer simulation system.

B. *Formal Syntactic Choice of BML*

Syntax is an indispensable part of any complete language and BML is not excluded.

American linguist Noam Chomsky is once mentioned in Syntactic Structures, and the classification of grammars and languages is divided into four classes: Type-0 syntax (unrestricted grammars), Type-1 grammars (context-sensitive grammars), Type-2 grammars (context-free grammars), and Type-3 grammars (regular grammars).

Type-0 grammars (unrestricted grammars) are unlimited in grammars and generate all grammatical structures.

Type-1 grammars have rules of the form $\alpha A \beta \rightarrow \alpha \gamma \beta$ with A (a nonterminal) and α, β and γ (strings of terminals and nonterminals). The strings α and β may be empty, but γ must be nonempty and contains at least one symbol. The rules can be interpreted: A is extended into γ in the context of α and β. This type of grammars is termed the context-sensitive grammars.

Type-2 grammars are defined by rules of the form $A \rightarrow \gamma$ with A (a nonterminal) and γ (a string of terminals and nonterminals). The rules can be interpreted that A is extended into γ. Contrary to Type-1 grammars, the context is not involved. Thus, this type of grammars is termed context-free grammars.

Type-3 grammars have more restrictions for rules and are termed regular grammars.

Considering Type-0 grammars and Type-3 grammars are rarely used, in this paper, we save lengthy discussion. Noam Chomsky thinks that the natural language is context sensitive and it means that the natural language is generated by Type-1 grammars. However, BML requires the automatic processing. Generally speaking, the syntax developed in the computational linguistics is limited to the context-free language processing, i.e. the language generated by Type-2 grammars. Therefore, the context-free grammars are the most appropriate to BML formal descriptive grammars.

BNF Form (Backus-Naur Form) is a metalanguage to express the context-free grammars and is a mathematic method used for the formal definition of linguistic grammars. BNF Form is a scientific and advanced pattern put forward by American philosopher of science Thomas Kuhn, meaning that scientists working together should have a common code and objective.

C. *BNF Analysis*

BNF can be basically expressed as:

$$A := \{\langle B \rangle\}[C]$$
$$B := "1" | "a"$$

Where: "=" is construction symbol, meaning "be defined as", "< >" consists of compulsory fields, "[ ]" contains optional fields, "{ }" consists of items repeating from 0 to infinity, "|" means selecting one item on the left or the right, equals to "or " in meaning, and the symbol in " " represents the terminal and means no further analysis.

Literature [7] seeks to analyze the BML syntaxusing BNF form. On the basis of literature [7], this paper makes a further exploration and analysis of the BML syntaxby means of BNF.

D. *Analysis of Equipment Support Command and Control Information*

Mainly ten types of command and control information are available in the equipment support command information system, which is shown in Fig. 2. The classification is made in terms of the intended use and the timing.

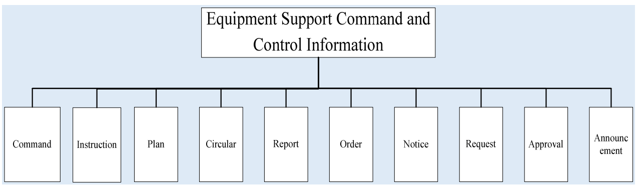

Figure 2. Classification of equipment support command and control information

Although different syntax rules are used in the above information, the underlying methods and principles to establish the syntax rules are consistent. Therefore, this paper mainly focuses on the establishment of syntax rules for the most common orders used in the wartime equipment support command and control information when describing the basic methods in constructing syntax rules.

III. CONCLUSION ESTABLISH BNF SYNTAX RULES FOR BML

According to the basic requirements of the context-free grammar, the context-free syntax G is defined as:

$$G := \langle N \rangle \langle T \rangle \langle P \rangle \langle S \rangle$$

Where G is the syntax itself and the remaining parts are elements contained in the grammar. N is a series of nonterminal symbols, T is a series of terminal symbols and satisfies the formula $N \cap T = \phi$, P is a series of generation rules, and S is the start symbol of N, a symbolic pointer to generate the correct grammar.

Based on the requirements mentioned above in the context-free grammar, the command and control order can be described as following:

$$S := \langle CI \rangle \{\langle B \rangle\} \{[Sp]\} \{[T]\}$$

The rule means that the order contains four parts, which is shown in TABLE I.

TABLE I. ELEMENTS AND MEANINGS OF S

Elements	Meanings
CI	Represent the commander's intention
B	Represent the task mandated to the security unit
SP	Represent the collaborative space, i.e. the space scope and the relevant information of the whole task
T	Represent the collaborative time, i.e. the preceding and subsequent relationship between all tasks in the whole combat process

Sp and T are optional elements, which are required B contains subtasks. Because CI, B, Sp, and T are nonterminal symbols, it requires further grammatical analysis.

A. Establish CI Syntax Rule Model

The commander's intention CI collectively reflects the purpose of accomplishing tasks:

$$CI := \{\langle Why \rangle\}$$

$$Why := \langle Qualifier1 \rangle \langle Label \rangle$$

$$Qualifier1 := "to" \mid "inorderto"$$

$$Label := "10001" \mid "10002"$$

The first formula expresses that the commander's intention is made up of multi-purposes. Every purpose can be expressed in the second formula, which is made up of the purpose qualifiers and the task labels, as shown in the third formula. The task labels are made up of a series of terminal symbols for the sole identification task B1, as shown in the fourth formula. Every terminal symbol can be obtained from the BML data model base.

B. Establish B Syntax Rule Model

B represents the task mandated to the combat unit: $B := \{\langle B1 \rangle\}$

$$B1 := \langle Verb \rangle \langle Tasker \rangle \{\langle Taskee \rangle\} \langle Where \rangle \langle Start\_when \rangle [End\_when] \langle Label \rangle \{[Mod]\}$$

Every task B can be made up of one or several subtasks B1. Elements and meaning contained in B1 are represented in TABLE II:

TABLE II. ELEMENTS AND MEANINGS CONTAINED IN B1

Elements	Meanings
Verb	Represent the executed action of the security unit
Tasker	Represent the one who commands
Taskee	Represent one or several command executors
Where	Represent the location or route of the task execution
State_When	Represent the starting time of the task execution
End_when	Represent the time of the task completion
Label	Represent the sole zone bit and this command can be cited by other commands through the zone bit
Mod	Represent the explanation supplementary to the whole task

The corresponding syntax rules of B1 are represented as following:

$$Verb := "defend" \mid "advance" \mid "assist" \mid "block" \mid "march"$$

Tasker: ="commander" | "equipment commander"

Taskee: ="battalion I" | "repair battalion"

$$Where := \langle At\_Where \rangle \mid \langle Route\_Where \rangle$$

$$State\_when := \langle Qualifier3\_Point\_in\_Time \rangle \mid \langle Qualifier4\_B \rangle$$

$$end\_when := \langle Qualifier3\_Point\_in\_Time \rangle \mid \langle Qualifier4\_B \rangle$$

Label:="mechanic operating" | "drive in" | "attack"

$$Mod := \langle Appendix \rangle$$

In terms of the elements contained in B1, some elements have been analyzed into the corresponding metalanguage of the underlying rules base and other elements require further analysis. Regarding these elements, the corresponding syntax rules are established as following:

1) Where

"Where" can be described as a specific location "At_Where" or a route "Route_Where". Location_name, E, W, Y, and S are all terminal symbols, as exemplified in the formula. Similarly, the information of these terminal symbols also can be obtained from the corresponding metalanguage word groups of BML underlying rule base.

$$At\_Where := \langle Qualifier2 \rangle \langle Location \rangle$$

$$Qualifer2 := "at" \mid "around"$$

$$Location := \{\langle Location\_name \rangle\} \mid \{\langle Position \rangle\} \mid \{\langle Location\_name\_Position \rangle\}$$

$$Location\_name := "beijing" \mid "XXtown"$$

$$Position := "(" \langle X \rangle "," \langle Y \rangle ")"$$

$$X := \langle E \rangle \mid \langle W \rangle$$

$$E := "0^\circ" "\le" \langle Value \rangle "\le" "180^\circ" "E"$$

$$W := "0^\circ" "\le" \langle Value \rangle "\le" "180^\circ" "W"$$

$$value := "1^\circ" \mid "2.234567^\circ"$$

$$Y := \langle N \rangle \mid \langle S \rangle$$

$$N := "0^\circ" "\le" \langle Value \rangle "\le" "90^\circ" "N"$$

$$S := "0^\circ" "\le" \langle Value \rangle "\le" "90^\circ" "S"$$

$$Route\_Where := \langle Source\_Destination\_Path \rangle \mid \langle Source\_Path \rangle \mid \langle Destination\_Path \rangle \mid \langle Path \rangle$$

$$Source\_Destination\_Path := \langle Source \rangle \langle Destination \rangle \langle Path \rangle$$

$$Source := "from" < Location >$$

$$Destination := "to" < Location >$$

$$Source\_Path := \langle Source \rangle \langle Path \rangle$$

$$Destination\_Path := \langle Destination \rangle \langle Path \rangle$$

$$Path := "along" \{ < Location > \}$$

2) State_When

State_When can be represented as the certain timing in the beginning or the start time relevant to another task.

Where Month, Day, and Year are optional items, and Hour, Minute, and Second are compulsory items. All the six items are terminal symbols.

End_When is an optional element with its descriptive syntaxsimilar to that of State_When. Thus, the corresponding discussion is redundant.

3) Mod

Mod is also an optional element with multiple replications. Generally, it involves the information relevant to the running of the simulation model, such as the marching speed, and morale etc, which is shown in the formula below.

Appendix:="at XX speed" | "staff"

C. Establish Sp Syntax Rule Model

Sp represents the area situation of the whole task:

$$Sp := \langle Control\_Feature \rangle \langle Tasker \rangle [Taskee] \langle Start\_When \rangle [End\_When] \langle Label \rangle$$

$$Control\_Feature := \langle area \rangle$$

$$area := "around" \langle Location \rangle$$

Control_Feature collectively describes the area coverage and displays the area situation. "Sp" differs from "Where" in that Sp focuses more on the overall situation. For example, "Where" represents the execution area of custom tasks in the process of descriptive syntax because one task can be broken down into different subtasks, while "Sp" collectively reflects the area coverage of the whole task, thus constructing the space relationship between tasks.

D. Establish T Syntax Rule Model

T represents the time relationship between the whole task and the subsequent task:

$$T := \langle Temporal\_term \rangle \langle Qualifier4 \rangle \langle Label \rangle \langle Label \rangle$$

$$Temporal\_term := "start" \mid "finish"$$

Temporal_term represents the task of the first Label and the preceding and subsequent relationship between tasks of the second Label. The example below represents that the label_1 task starts at the end of the label_2 task.

Start at_the_end_of label_1 Label_2

Similarly, "T" differs from "Start_When" in that T focuses more on the overall situation. Start_When reflects the time relationship between subtasks and T reflects the sequential relationship between complicated tasks in the combat process.

IV. APPLICATION

According to the established rules and principles, the pilot-in-the-loop equipment support interactive information system is formed through the system development. The input interface of the command is represented in Fig. 3 and the command identification interface is displayed in Fig. 4.

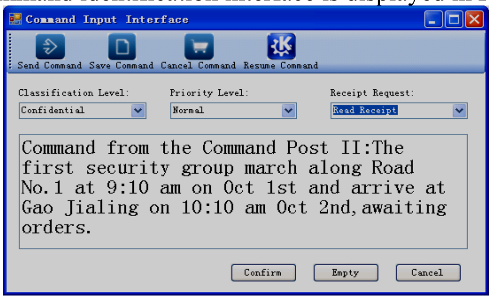

Figure 3. Command input interface

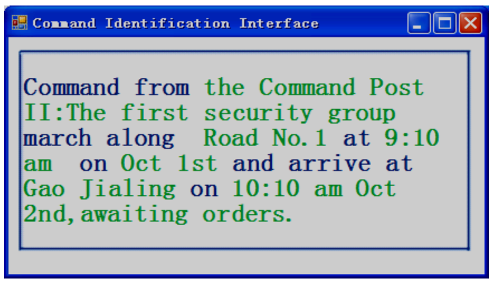

Figure 4. Command identification interface

The order in Fig. 4 has been cited as the example: Command from the Command Post II: The first security group march along Road Nr. 1 at 9:10 am on Oct. 1st and arrive at GaoJialing on 10:10 am Oct. 2nd, awaiting orders.

The order is decomposed as following using the established rule:

$$CI := "to" Label\_1$$

$$Label\_1 := "awaiting\ orders"$$

$$B := "march\ along","command\ post\ II",$$
$$"first\ security\ group"$$
$$\langle Destination\_Path\rangle\langle Start\_when\rangle[End\_when]"Label\_1"$$
$$Start\_when := "at""10""1""2015""9""10""00"$$
$$End\_when := "early\_than""10""2""2015""10""10""00"$$
$$Route\_Where := "to""Gaojialing","Road\ No.1"$$

The decomposed BML semantic metalanguage word groups such as “Gaojialing” are automatically identified and the virtual forces is driven for the simulated model running, realizing the interconnection between the commander and the simulation system. The strategic metalanguage word group bank is constructed in the system and the input free text instruction is automatically identified as grammatical word groups marked in green color in order to reflect the identification effect. In the regular simulation training, the information is input according to the established syntax rules, improving the interaction efficiency between human and machine.

V. CONCLUSIONS

This paper adopts the BML descriptive syntax to analyze various grammatical characteristics and the context-free syntax is selected for BML. BML grammatical rules related to the command and control information for the equipment support command information system are constructed and the corresponding semantic model base is established by using BNF form. Based on this, a set of prototype system is developed for the interactive processing of the equipment support information. This paper establishes the grammatical rules for the equipment support command on the basis of the grammatical rule methods. Other types of grammatical rules for the command and control information will be further constructed in the future. This paper contributes to the study of improving the interoperability mechanism between the equipment support command information system and the simulation system.

REFERENCES

[1] Sun Xiao, Liu Zhongpeng, Ding Lijun, “Study on Extended Battle Management Language (XBML)”, *Computer Measurement & Control*, vol. 19, pp. 67-71, Aug. 2013.

[2] Ma Kaicheng, “Simulation System for Computer Contracted Tactical Operations”, *Maritime Press*, 1989.

[3] Zhu Li, Bao Guangyu, “Study on BML Action Semantics Labeling Methods Based on Atomic Action”, *Warship Electronic Engineering*, vol. 31, pp. 142-145, Dec. 2011.

[4] Hopcroft J E, Ullman JD, “Introduction to Automata Theory Languages and Computation”, *Addison-Wesleg*, 1979.

[5] Dr. Ulrich Schade, Dr. Michael R. Hieb, “Formalizing Battle Management Language A Syntaxfor Specifying Orders”, *06S-SIW-068*.

[6] Crocker D, Overell P, “Augmented BNF for Syntax Specification: ABNF [EB/OL]”, *http://www.IETF.org/RFC/RFC224.txt.*

[7] Yuan Limiao, Bao Guangyu, Zhu Li, “BNF-based Battle Management Language Syntax Notation”, *Journal of Military Communications Technology*, vol. 31, pp. 35-39, Apr. 2010.