A Coloured Petri Net Model for Composite Behaviors in Multi-Agent System

Su Jindian Guo Heqing
College of Computer Science and Engineering
South China University of Technology, 510006
Guangzhou, China

Yu Shanshan
School of Information Science and Technology, Computer Science Department
Sun Yat-sen University, 510275
Guangzhou, China

***Abstract*—Traditional informal description methods have obvious shortcomings in describing complex composite behaviors in MAS, not only due to their hierarchical and concurrent properties, but also for their complicated dependencies. In this paper, we propose a formal description model, MAB_CPNs, for composite behaviors among agents based on coloured Petri nets. MAB_CPNs takes data dependent relationship and control dependent relationship into consideration and provide a more comprehensive perspective on composite behaviors. We also present the basic process structure and the formal description of behavior substitution to support the hierarchical structure description of behaviors. Finally, we give a case study.**

***Keywords*—coloured Petri nets, multi agent system,composite behaviors**

I. Introduction

Currently, agent and multi agent system (MAS) become a research hotspot in the field of distributed artificial intelligence and have been widely used in many applications. In MAS, agents are autonomous and act independently. On the other hand, they need to work together because they can't finish those complex tasks individually, which make it necessary to form into many composite behaviors among agents. With the increasing complexity of tasks, the various dependent relationships among composite behaviors become more and more complicated and these make the traditional manual ways of analysis unfeasible. So, how to provide abstract and hierarchical descriptions about composite behaviors and offer the corresponding tools to verify the correctness of abstract process design and flow instances become quite challenging.

The analysis methods on agents' behaviors can be divided into two kinds, informal methods and formal methods. The first one is widely used in the past, but they can't fully depict the concurrency and asynchronism characteristics of agents' behaviors because of lack of exact semantics. While formal methods are based on some appropriate mathematical formalism such as automatic state machine, algebra and graphical theory and can provide good ways for description, construction and analysis of distributed and concurrent system. Coloured Petri nets (CPNs), as a formal method, is a kind of high level of ordinary Petri nets and can express richer semantic meanings by distinguishing various places with different colors, in addition to owning the strengths of quantitative and qualitative analysis capability about concurrent and distributed systems of ordinary Petri nets.

The remainders of this paper are organized as follows. In section 2, we first briefly introduce some current researches about composite behaviors in MAS and point out the existing problems. Then we bring forward a formal model to analyze agents' composite behaviors based on CPNs in section 3. In section 4, we present the basic patterns of process structure of composite behaviors and behavior substitution operations. In section 5, we give a case study. Finally, conclusions and future directions are given.

II. Relative Works

Most of the current researches in MAS focus on the problems of agent communication language [1, 2] or scheduling strategy [3], and seldom pay attention to the composite behaviors. The authors in [4] chose Petri nets as the modeling language to describe the relationships among multi dynamic description logic actions and used reachability graph to analyze them. [5] described the inner structure of agents and analyzed those dynamic relationships among behaviors in details, particularly the concurrent relationships in MAS at the level of system and agent respectively. [6] modeled the aspects of a sociological theory using Petri nets from micro and macro aspects respectively. [7] used hierarchy CPNs to model agents' behaviors and [8] used Petri nets to describe the flow structures and dynamic behavior rules of the behavior coevolution processes. But these researches mainly focused on the descriptions of control dependent relationship among behaviors and neglected the data dependent relationship, which, as a result, couldn't provide a full perspective on composite behaviors.

In this paper, we use ordinary CPNs as the modeling language and can easily extend it to hierarchy CPNs through behavior substitution operations. [9] also points out the hierarchy CPNs is isomorphic to ordinary CPNs.

III. CPNs Model for Compositve Behaviors

Composite behaviors among agents can be regarded as a partial ordered set of behaviors during cooperation processes and can be defined as transitions in CPNs and their inputs and outputs will be mapped to places with different colors. The relationships between behaviors, inputs and outputs will be mapped to the flow relationships between places and transitions. The data flows demonstrate their data dependent relationships and control flows are mainly used to express the logical control relationships of business processes. Both of them are combined to determine the execution of composite behaviors at the runtime.

A. Definition of the Basic Color Set

Here, we give the definition of a basic color set .

Definition 1. *D* (Color Set) is defined as follows:

colset *Din* = string;

colset *Dout* = string;

colset *DATA* = with *Din* | *Dout*;

colset *CONTROL* = with st;

Din and *Dout* stand for the data types of input and output respectively, both of which are simplified as string and can be endowed with richer type semantics in practical applications. *Data* stands for all data objects, whose value is *Din* or *Dout*. *CONTROL* is used to distinguish control places from data places and help express the control dependent relationship among behaviors. If not explicitly stated, we suppose that all color sets in the following sections will include *D*.

B. MAB_CPNs for Composite Behaviors

For composite behaviors among agents in MAS, the correspondent CPNs model is a set of transitions and places with various flow relationships.

Definition 2. The composite behavior CPNs Model for MAS, $MA\_CPNs=(\Sigma,P,T,A,K,C,G,E,I)$, is:

(1) Σ is a finite set of color sets, which is composed of all color types in MAS and has $D\subseteq\Sigma$.

(2) $P=P_c\bigcup P_d\bigcup P_o$ ($P_c\bigcap P_d\bigcap P_o=\varnothing$) is a finite set of token places, where $P_d=In\bigcup Out$ is the set of input data token set, In ($In=\{In_1,\cdots,In_n\}$), and output data token set, Out ($Out=\{Out_1,\cdots,Out_m\}$), which has $i\in In\wedge{}^\bullet i=\varnothing$ and $o\in Out\wedge{}^\bullet o=\varnothing$; P_c is a finite set of control places and P_o refers to the remaining place set in MAS, such as knowledge places or rule places.

(3) $T=T_c\bigcup T_s\bigcup T_0$ is a finite set of transitions, where T_c is the set of agents' behaviors; T_0 is the set of zero transitions, which is used to help express the control structure of composite behaviors and don't involve any specific functional operations; T_s refers to the set of other transitions in MAS.

(4) K is a capacity function defined on P_c such that $\forall p\in P_c : K(p)=1$, which means each control place can contain no more than one control token.

(4) $A=A_c\bigcup A_d$ ($A_c\bigcap A_d=\varnothing$) is a finite set of control arc set, A_c, and data arc set, A_d, where $A_c\subseteq(P_c\times T_c)\bigcup(T_c\times P_c)$ and $A_d\subseteq(P_d\times T_d)\bigcup(T_d\times P_d)$.

(5) $C:P\rightarrow\Sigma$ is a color function defined on P, where

$C(P_d)\rightarrow DATA$,

$C(In)\rightarrow Din$

$C(Out)\rightarrow Dout$,

$C(P_c)\rightarrow CONTROL$

$C(P_o)\rightarrow\Sigma\backslash\{CONTROL,DATA\}$.

(6) $G:T\rightarrow G(t)$ is a guard function defined on T representing the preconditions of behaviors; For $\forall t\in T$, there is $Type(G(t))=bool\wedge Type(Var(G(t)))\subseteq\Sigma$.

(7) $E=E_c\bigcup E_d$ is an arc expression function defined on A_c and A_d, and it has $\forall a\in A_c:E_c(a_c)=st$ and $E_d:A_d\rightarrow E_d(a)$, which satisfies:

$$\forall a\in A_d:[Type(E_d(a))=C(p)_{MS}\wedge Type(Var(E_d(a)))\subseteq\Sigma]$$

(8) I is the initialization function defined on P such that $\forall p\in P:[Type(I(p))=C(p)_{MS}]$.

For MAB_CPNs, the entrance and exit of composite behaviors can be defined as $P_i=\{p_i\,|\,p_i\in P_c\wedge{}^\bullet p_i=\varnothing\}$ and $P_o=\{p_o\,|\,p_o\in P_c\wedge p_o^\bullet=\varnothing\}$ respectively.

Only one entrance can exactly describe the start of the process, while more than one exits means the same process can lead to different results according to the semantic conditions.

TABLE I. MAPPING RELATIONSHIPS BETWEEN MAS AND MAB_CPNS ELEMENTS

MAS	MAB_CPNs
Sets of Data Types	Σ
Agents' Behaviors	T_c
Input Data of Behaviors	In
Preconditions of Behaviors	$G(t)$ ($t\in T$)
Output Data of Behaviors	Out
Initialization Conditions	I
Process Entrance	P_i
Process Exit	P_o

IV. PROCESS STRUCTURES AND BEHAVIOR SUBSTITUTION

A. Process Structure

In MAS, agents often need to cooperate with each other in order to finish tasks, which as a result form into complex composite behaviors. The basic process structures of composite behaviors include sequence, parallel, choice and circular and their correspondent algebra operators are $a=a_1\bullet a_2|a_1\oplus a_2|a_1||a_2|na_1$.

(1) Sequence Pattern (see Figure 1)

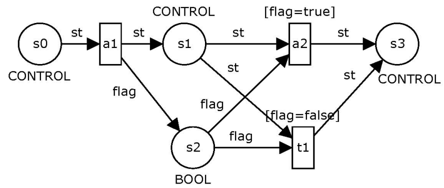

Figure 1. Sequence Pattern $a_1 \bullet a_2$

$a = a_1 \bullet a_2$, where $\bullet$ is the sequence operator, means a_1 will execute after a_2. t_1 is a zero transition representing that the process will directly return when a_1 fails, in stead of executing a_2 continuously.

(2)Choice Pattern (see Figure 2)

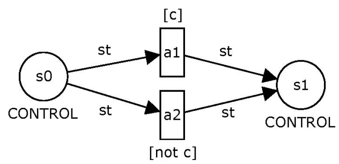

Figure 2. Choice Pattern $a_1 \oplus a_2$

$a = a_1 \oplus a_2$, where $\oplus$ is the mutually exclusive operator, means to choose a certain one among several mutual exclusive behavior and block the remaining.

(3)Parallel Pattern (see Figure 3)

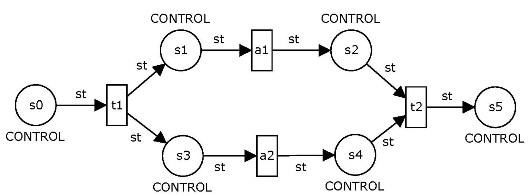

Figure 3. Parallel Pattern $a = a_1 \| a_2$

$a = a_1 \| a_2$, where $\|$ is the parallel operator, means the control flow is separated into several behavior branches which can be executed in parallel. t_1 and t_2 are two zero transitions used to help express the parallel structure.

(4) Circular Pattern (see Figure 4)

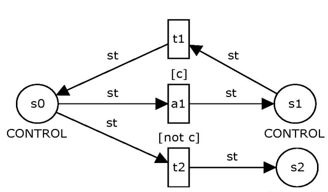

Figure 4. Circular Pattern na_1

$a = na_1$ means that the composite behaviors are to execute a_1 in n time. t_1 is a zero transition used to express the circular structure, while zero transition t_2 means to exit the loop when the preconditions of a_1 can't be satisfied. a_1 and t_2 are two mutual exclusive transitions and they are restricted by a mutual exclusive guard function c.

B. *Behavior Substitution*

During the processes of modeling agents' complex composite behaviors, we often prefer to build a sketch structure first and refine it gradually to get a hierarchical model at different abstract level. These can be regarded as substitution operations from the viewpoint of Petri nets theory. Substitution operations in CPNs are divided into two kinds, place substitution or transition substitution. Because we focus on modeling composite behaviors, so we mainly discuss behavior substitution operations and neglect the other one. The formal description is shown as followings.

Suppose $N = (\Sigma, P, T, A, K, C, G, E, I)$ and $N' = (\Sigma', P', T', A', K', C', G', E', I')$ are two different CPNs models and $t \in T$.

Definition 3: N'' is a new CPNs model after using N' to replace the transition t of N, which makes N become a transition subnet of N''. The formal description of N'' is shown as follows:

$\Sigma'' = \Sigma' \cup \Sigma$, $P'' = P' \cup P$, $T'' = (T' - \{t\}) \cup T$, $K'' = K$, $C'' = C' \cup C$, $G'' = G' \cup \{G - G(t)\}$, $E'' = E' \cup E$, $I'' = I$, $A'' = A' \cup (A - A(t)) \cup A''(N, N')$, where

(1) $A(t) = A \cup (P \times \{t\} \cup \{t\} \times P)$;

(2) $A''(N, N') \subseteq \{(p,t),(t,p) \mid p \in P, t \in T'\}$.

$A''(N, N')$ should meet:

a) $|A''(N, N')| = |A(t)|$;

b) $\forall p \in P: (p \in {}^{\bullet}t \Leftrightarrow \exists t \in T': p \in {}^{\bullet}t) \wedge (p \in t^{\bullet} \Leftrightarrow \exists t \in T': p \in t^{\bullet})$

The sketch of a behavior substitution example is shown in Figure 5.

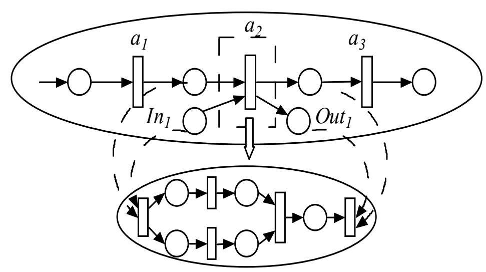

Figure 5. The Sketch of Behavior Substitution Example

V. CASE STUDY

Here, we use an online itinerary as an example to demonstrate the applications of MAB_CPNs. Suppose a user wants to attend a meeting from city A to B and he needs to develop an itinerary before leaving, which includes transport types and hotel booking services. There are some restrictions on choosing vehicles. If the traveling time by train from A to B is less than or equal to six hours, he will travel by train; Otherwise, he will choose airplane.

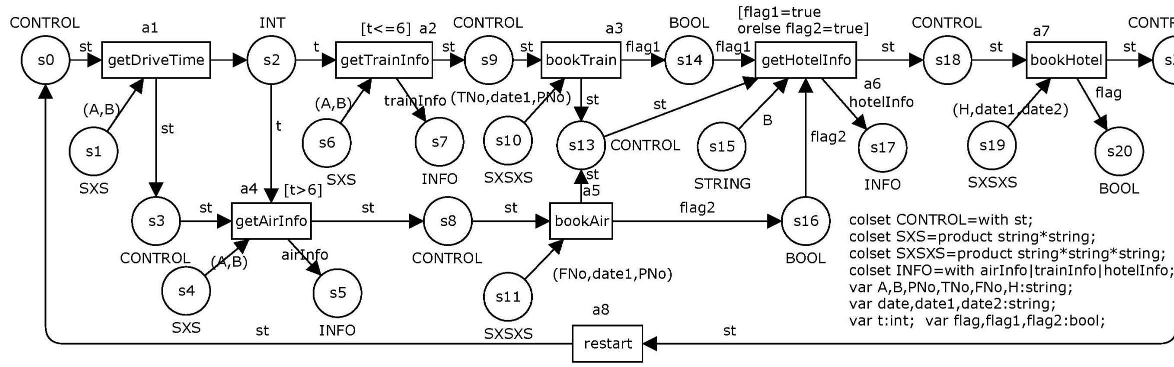

Figure 6. MAB_CPNs Model of Online Itinerary

The MAB_CPNs model of online itinerary using CPN Tools [10] is shown in Figure 6.

Suppose that the behavior a_4 is also made up of other behaviors, then its' behavior substitution operation of a_4 is shown in Figure 7.

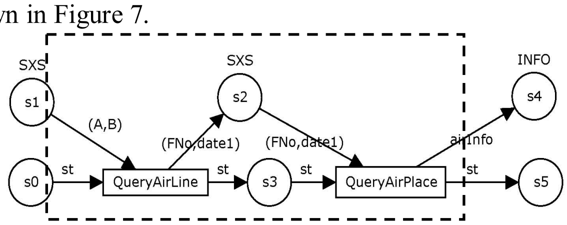

Figure 7. Behavior Substitution of a_4

From Figure 6 and 7, we can see that MAB_CPNs can depict agents' composite behaviors more comprehensively by incorporating data dependent relationship and control dependent relationship. In addition, behavior substitution operations enable MAB_CPNs to provide hierarchical descriptions about composite behaviors at different abstract levels.

VI. Conclusions and Future Work

Dynamic agent cooperation is one of the challenge problems of MAS and how to describe and verify the composite behaviors among agents before real execution is quite critical. In this paper, we try to use CPNs technology as the modeling tool to build a formal description model about the composite behaviors in MAS and incorporate data dependent relationship and control dependent relationship to express richer behavior semantics. We also present the basic pattern of process structure and the formal description of behavior substitution operations.

But our discussions about composite behaviors in MAS are still at the initial state and we will continue to analyze them in more details and their dynamic characteristics such as conflict and concurrency in the future.

Acknowledgment

This research was supported by the Scientific Research Foundation for Ph.d of South China University of Technology.

References

[1] H. Bing-di, Z. Li-ying, "Interaction Protocols Modeling for Multi-Agent Systems Based on Colored Petri Net," Computer Technology and Development. Shanxi, China, vol.16, pp.42-43,46, November 2006.

[2] C. Jin-feng, L. Hui-yi, "A Colored Petri Net Based Approach for Multi-Agent Interactions," Shanghai, China, Computer Applications and Software, vol. 22, pp.142-144, October 2005.

[3] B. Quan, Z. Min-jie, Z. Hai-jun, "A Coloured Petri Net Based Strategy for Multi-Agent Scheduling," Proceedings of the Rational, Robust, and Secure Negotiation Mechanisms in Multi-Agent Systems (RRS'05), IEEE Society, 2005, pp.3-10

[4] M. Bing-xian, X. Ying-lei, "on Dynamic description Logic Actions Relationship with Petri Nets," Acta Automatica Sinica, Beijing, vol. 22, pp. 1144-1149, November 2007.

[5] Y. Ji-guo, M. Bing-xian, C. Bao-xiang, et al, "Modeling of Multi-agent Systems Behaviors by Using Hierarchical Petri Nets," Computer Engineering, Shanghai, China, vol. 32, pp. 26-28, August 2006.

[6] M. Köhler, M. Martens, H. Rölke, "Modelling Social Behaviour with Petri net based Multi-Agent Systems," In Proceedings of the Workshop MASHO'03, Hamburg, Germany., 2003.

[7] M. Bing-xian, X. Ying-lei, W. Zhe-hui, "Modeling Agents Behaviors with Hierarchical Colored Petri Nets," Acta Simulata Systematica Sinica, Beijng, China, vol.15, pp.114-118, August 2003.

[8] L. Jie, D. Jian-min, "The Model Study of Multi-Agent Behaviors Coevolution Based on Petri Net," Microelectronics & Computer, Xian, China, vol.24, pp.152-156, 2007.

[9] K. Jensen, "Coloured Petri Nets. Basic Concepts, Analysis Methods and Practical Use, Volume 1, Basic Concept," Monographs in Theoretical Computer Science. Berlin, Heidelberg, New York: Springer-Verlag, 1997.

[10] K. Jensen, L. Michael Kristensen, L. Wells, "Coloured Petri Nets and CPN Tools for Modelling and Validation of Concurrent Systems," *International Journal on Software Tools for Technology Transfer*, vol.9, pp.213-254, 2007.