A Method for Verifying the Behavioral Consistency between Public Business Process and Private Business Process Based on Environment

Yitong Chen
Software College of Yunnan University, Kunming 650504, Yunnan, China
cyt133@hotmail.com

Abstract. The cross-organizational business process is characterized by privacy, collaboration, autonomy, dynamic interaction structure, etc. It is necessary to model and analyze the cross-organizational business process in order to ensure the accuracy and consistency of process interaction among different organizations. The open workflow net proposed on the basis of Petri net is used as a tool to model the cross-organizational business process. The cross-organizational business process is divided into two parts, namely private process and public process. The private process is regarded as the internal view of the organization, and the public process is regarded as the external view of the interaction between an organization and other organizations. The interaction behavior of model verification is regarded as the starting point with regard to the behavioral consistency problem of the model. A method for verifying the behavioral consistency based on environment is proposed, which is more relaxed than the existing method for verifying the behavioral consistency based on mutual simulation and more consistent with the asynchronous communication among organizations.

1. Research status
It is proposed in literature [1] that Workflow Net is regarded as a tool for modeling the business process, and it is proposed in literature [2] that oWF-net is regarded as a tool to model the cross-organizational business process on the basis of the work-flow network according to the modeling and analysis of the business process. Some scholars put forward the idea of dividing the business process into a private process and a public process [3, 4] in order to protect the internal privacy of organizations aiming at the modeling and analysis of the cross-organizational business process. The public process is obtained by abstracting the behavior of the private process. Therefore, it is necessary to ensure that the interaction behavior of the public process is consistent with that of the private process when the public process is extracted from the private process. The behavioral consistency between business processes is mostly analyzed from the PI - algorithm perspective in existing studies. For example, Bi-Simulation is used as the criteria to verify the behavioral consistency between public and private processes in literature [5, 6]. However, the interaction among most organizations is based on asynchronous message communication in reality. Messages can not be received according to the transmission order generally. It is required that the order of messages should be ignored when the consistency between the public process and the private
process is determined. The public process of the Figure 1 (a) can be simulated by the private process in the Figure 1 (b) according to the definition of mutual simulation [7], and the private process of the Figure 1 (b) can not be simulated by the public process in the Figure 1 (a). It is not correct in actual interactions. The message order should not be regarded as a factor in determining the behavioral consistency.

Figure 1. Online shopping process.

2. Environment construction and behavioral consistency standard

A method for verifying the behavioral consistency between public process and private process based on environment verification is proposed in the paper according to the research status. The order of messages is ignored in the method. The following conclusion can be obtained through the method in the paper regardless of the public process in the Figure 1 (a), namely order determination or notification of the delivery address to the merchant: the public process of users in Figure 1 (a) and the private process of users in Figure 1 (b) have behavioral consistency.

2.1. Environment

The interaction behaviors of possible interaction objects in an organization can be determined according to the interaction behaviors of the organization. We call the interaction objects of the interaction behaviors meeting the requirements of the organization as interaction environment of the organization. We only care about the environment message sent to the organization and organization message sent to the environment. Therefore, the environment can be described in terms of message sequences and changes in message sequences. Since an automaton is a formal tool to describe the transformational relation among states represented by letter sequences, messages are expressed as letters, the automat can be used for expressing the environment, and the formal definition of the environment is given as follows.

Definition 2.1 (environment): \( \Sigma \) is set as the private process described by an oWF-net, then the environment \( E \) corresponding to \( \Sigma \) is a quintet meeting the following conditions \( E = (Q, A, f, \lambda, \Omega) \):

- \( A \) refers to a message set, in addition, \( A \subseteq P_I \cup P_O \) \((1)\)
- \( \lambda \) refers to a finite state set, in addition, \( \lambda \subseteq Q \) \((2)\)
- \( Q \) refers to a finite state set, in addition, \( Q \subseteq \{w \mid w \in A^*\} \), each state is expressed as \( q_i, q_i \in Q \), wherein \( A^* \) is the closure of \( A \) \((2)\)
- \( A \) refers to a only initial state, in addition \( \lambda \subseteq Q \) \((3)\)
- \( \Omega \) refers to a final state set, in addition \( \Omega \subseteq Q \) \((4)\)
f refers to a transfer function, f refers to the mapping on \( Q \times A \rightarrow Q \). In addition, \( f(w, x) = \{ \text{wa} \} \), if \( wa \in Q, a \in A \), otherwise \( f(w, x) = \emptyset \) \( (5) \)

Wherein, it is indicated in the condition (1) that the message is expressed by the letter representing the interface library; it is indicated in the condition (2) that the state \( q_i \) in the state set \( Q \) represents the message sequence; the one-time state transition described by transfer function \( f \) represents one-time message sending or receiving of environment. The interactions between an organization and its corresponding environment are recorded in the state. The state transition of an organization’s environment is determined by the interaction behavior of the organization. Therefore, the environment corresponding to the organization is determined as long as the state set is determined.

Environment is an abstraction of interactive objects by describing interactive behaviors. The environment for describing the interactive objects with the same interactive behaviors is not unique since there are many interactive behaviors. It is concluded that the environment is the maximum environment if its interaction behaviors of an environment contains all other possible environment interactions.

All open workflows \( S \) are restricted in the paper excluding loops. The message sequence is finite, and the state set representing the message sequence is also finite under the premise. There is a state set containing all states, and the environment determined by the state set contains all states. Namely, the maximum environment must be available and unique. The definitions of the maximum environment are given as follows.

Definition 2.2 (maximum environment): \( E = (Q, A, f, \lambda, \Omega) \) is set the environment of an organization. We call the environment \( E \) as the maximum environment. When and only when \( Q_0 \subseteq Q \) aiming at all possible environments of the organization \( E_i = (Q_i, A, f, \lambda, \Omega) \), in addition \( Q_0 \cup Q_1 \cup \ldots \cup \ldots \cup Q_n = Q \), wherein \( n \) is the number of possible environments for the organization.

2.2. Construction environment
There may be false deadlocks in the execution of the process due to the features of the cross-organizational business process. Namely, the process cannot be continued because of waiting for messages sent by interactive objects, the following conditions can be executed by the process after the message is received. However, it is impossible to execute the following processes after to-be-waited messages are received under all circumstances. The definition of deadlocks is expanded in the paper in order to distinguish between fake deadlocks and deadlocks that cannot be eliminated due to own process problems of the organization. Deadlocks that can be eliminated after the appropriate interaction behavior of the interacting objects is executed are called external deadlocks, and nonerasable deadlocks are called internal deadlocks.

The algorithm of the construction maximum environment is obtained aiming at include all states. The algorithm steps of the construction maximum environment are introduced as follows, which consists of generating and eliminating redundancy.

Algorithm 2.1 (Generating environment)
Input: private process \( \Sigma = (P, T, F, M) \), to-be-processed message set \( S = P_0 \cup P_I \) of \( \Sigma \) (the message is named after the identity of the interface library);
Output: environment \( E \):
(a) Stop construction, and return to the current environment \( E \) that has been constructed if \( S \) is empty.
(b) Stop construction, and return to the current environment \( E \) that has been constructed if the current sign of \( \Sigma \) is the end sign \( o \).
(c) Stop construction and return to the current environment \( E \) that has been constructed if there is an internal deadlock under the current sign \( q_i \) or an external deadlock that cannot be continued through message interaction;
(d) Otherwise, environment \( E \) is empty, and the initial state \( q_i = \{ \emptyset \ (m_i) \} \) is constructed;
(e) Otherwise, the states \( q_i = \{ P_1, \ldots, P_n (m_i) \} \), \( P_i \in P_0 \cup P_I \) are generated, the message \( P \) is followed by?
message sent and received by E should be contrary to the to-be-sent and to-be-received message. $m_i \in \mathcal{M}$ represents the current state of $\Sigma$.

(f) The subsequent state $q_{i+1} \ldots q_{i+n}$ of the state $q_i$ is generated. A $q_i$ subsequent state is generated if $S$ contains a to-be-processed message $s_i$. The transfer function $f(q_i, s_i) = \{ q_{i+1}, \ldots, q_{i+n} \}$ is generated. The message $s_i$ is deleted from $S$ if the subsequent state of one corresponding message $s_i$ is generated once. $\Sigma$ signs are updated to the sign that the corresponding message of the subsequent state is processed by $\Sigma$.

(g) The algorithm 3.1 is called recursively until environment $E$ is generated.

Algorithm 2.2 (redundancy elimination)
Input: environment $E$ without redundancy elimination;
Output: environment $E$ after redundancy elimination;
(h) Each end state is checked;
(i) The end state should be deleted if an internal deadlock exists under the identity corresponding to the end state;
(j) The state for producing the first external deadlock is recalled upwards if an external deadlock exists under the identifier corresponding to the end state, the state and subsequent states are deleted;
(k) Steps a to care repeated until there is no state to delete;
(l) The identity of the private process $\Sigma$ in each state is deleted finally.

Because messages are communicated through the interface library in the private process $\Sigma$, the to-be-processed message set $S$=POPI also contains all the to-be-processed messages according to algorithm 2.1. A corresponding message sequence is generated for every possible message communication in the current state when the algorithm 2.1 generates a subsequent state of a certain state. Therefore, the environment generated by the algorithm 2.1 contains all states. However, some state sets of the environment obtained according to algorithm 2.1 are not correct, and some states will cause deadlock in private process, and such states should not be included in the state set. All states that will cause deadlock will be deleted from the state set of the environment according to the algorithm 2.2. The state set of the finally obtained environment contains all possible environment states. The quantity is neither more nor less. The environment obtained according to algorithm 2.1 and algorithm 2.2 is the maximum environment finally.

2.3. Verification of behavioral consistency based on environment

We still need to know whether the organization and the environment of the organization can interact normally or not if there are organization and environment of the organization, and we can describe the interaction thereof by combining the state of the environment with the identifier of the organization. Such a system combining the state of environment with the identifier of the organization is called a composite system. We only use the library with token to replace the identifier in order to describe the interaction between organization and environment more intuitively. The definition of the composite system is given as follows.

Definition 2.3 (composite system): $\Sigma$ is set as a private process, and $E$ is the environment corresponding to the private process $\Sigma$, then the composite system formed of $\Sigma$ and $E$ is a triplet $\mathcal{N}=(S, Q, F)$ that satisfies the following conditions.

$S$ is the library sequence with token in $\Sigma$, $s=<x_1, x_2>$, where $x_1 \in P^*_M$, $x_2 \in P^*_E$ (1)

$Q$ is E state set, $q \in Q$ (2)

$F$ is a flow relation on $\mathcal{N}$. If there is a transition $t$ in $\Sigma$, $s$ becomes $s'$ after $t$ is executed, then $F$ represents the conversion from $(s, q)$ to $(s', q)$. If there is a message $p$ in $E$, $s$ becomes $s'$ and $q$ becomes $q'$ after the message $p$ is processed by $\Sigma$, then $F$ represents the conversion from $(s, q)$ to $(s', q')$ (3)

$(s, \emptyset, q_1)$ is the initial state, $s_1=\emptyset$, $q_1$ is the initial state of environment $E$ (4)

If there is a composite system for describing the interaction between an organization and its corresponding environment, it is possible to judge whether the interaction between the organization
and its corresponding environment is normal or not through the composite system. Normal interaction is defined in the paper with reference to the idea of weak termination defined by Professor Aalst [8].

Definition 2.4 (normal interaction): N is set as the composite system composed of private process and environment E. If each state accessible from the initial state can reach the final state of a library sequence s=<$\delta$>, wherein $\delta$=\emptyset, we conclude that the private process and its corresponding environment can interact normally.

We are concerned about whether the public process can interact with the process of the same organization normally or not after the private process is replaced with its corresponding public process if the private process can interact normally with the process of other organizations when the public process is extracted from the private process.

Therefore, the criteria for examining whether the public process and private process have consistent behaviors in the paper are shown as follows: if the private process can interact with the processes of other organizations normally, then the public process can interact with the processes of the same organizations normally after the private process is replaced with its corresponding public process. We consider that the behavior of the public process has consistent with that of private process. Otherwise, it is considered that the behavior of the public process is inconsistent with that of private process. On the contrary, if the public process can normally interact with the processes of other organizations, the private process can normally interact with the processes of the same organizations after the public process is replaced with its corresponding private process. We assume that the behavior of private process is consistent with that of public process.

The environment proposed in the paper can be described according to the interaction objects which are given by the interaction behavior of the process itself, can interact with the process normally, and have the same interaction behavior. The maximum environment is described by the interaction object which is given according to the interaction behavior of the process itself, can interact with the process normally, and have the same normal interaction behavior. Interaction objects that are not described in the maximum environment cannot interact with the corresponding process normally. Such interaction objects are not considered during verification of the behavioral consistency between the public process and private process. After all interaction objects capable of interacting with the public process normally and all interaction objects capable of interacting with the private process normally are obtained, the behavioral consistency of public process and private process is verified, which can be translated into the verification of the maximum environment consistency of the public process and private process. If the maximum environment of public process and private process is consistent, then the public process and private process can interact with the same environment normally. The public process can also interact with the environment of the private process after the private process is replaced with the public process. It can be concluded that the behaviors of public process and private process are consistent according to the criteria for verifying the behavioral consistency between the public process and the private process. The definition of the behavioral consistency in the paper is shown as follows.

Definition 2.5 (behavioral consistency): the maximum environment corresponding to the process X is set as $E_{1a}(Q_1, A_1, f_1, \lambda_1, \Omega_1)$, the maximum environment corresponding to the process Y is set as $E_{2a}(Q_2, A_2, f_2, \lambda_2, \Omega_2)$, and we conclude that the behavior of the process X is consistent with that of the process Y. When and only when $E_1$ and $E_2$ meet the following conditions.

$X$ and $E_1$ can interact normally, $Y$ and $E_2$ can interact normally (1)

$A_1 = A_2$ (2)

$Q_1 \supseteq Q_2$ (3)

$f_1 \supseteq f_2$ (4)

$\Omega_1 \supseteq \Omega_2$ (5)

$\lambda_1 = \lambda_2$ (6)

It is required in the condition (2) that the message received and sent by the process X and the process Y is consistent. It is required in condition (3) and condition (4) that the maximum environment corresponding to the process X includes the maximum environment corresponding to the process Y.
namely $E_1 \sqsupset E_2$. When $E_1 = E_2$ only, we think that the behavior of the process $X$ is consistent with that of the process $Y$. In addition, the behavior of the process $Y$ is consistent with that of the process $X$.

Now, we adopt the online shopping process in Figure 1 as an example. We use the method of the paper to verify whether the user's public process in Figure 1 (a) is consistent with the user's private process in Figure 1 (b) or not.

![Figure 2. Maximum environment of user process in online shopping process.](image)

Algorithm 2.1 and algorithm 2.2 are respectively applied to generate the maximum environment aiming at the user's public process in Figure 1 (a) and the user's private process in Figure 1 (b), and the composite system is applied to check whether the generated maximum environment can interact with the process normally or not. The maximum environment shown in Figure 2 is finally obtained (wherein, user order determination, receiving address, payment request, payment confirmation and receiving confirmation are respectively represented by a, b, c, d and e in the Figure). Namely, the maximum environment corresponding to the public process of the user in figure 1 (a) and the private process of the user in figure 1 (b) is the maximum environment shown in Figure 2. It can be seen from the figure that the public process of the user and the private process of the user receive and send consistent messages: a, b, c, d and e. Moreover, the maximum environment corresponding to the user's public process and the user's private process is the same maximum environment.

3. Summary and prospect

The paper starts with the perspective of the actual interaction across organizations aiming at the interaction consistency for private and public processes. Since the interactions between the organizations are mostly implemented through the asynchronous communication as the precondition, a method for verifying the behavior consistency is proposed, which is more consistent with actual situation than mutual simulation, namely the method for verifying the behavior consistency based on environment.
Since the current work in the paper is based on the premise that business processes do not contain loop structures, although most business processes do not contain loop structures in reality, some business processes still contain loop structures. Therefore, the method proposed in the paper still keeps the behavior consistency of public and private processes at the expense of their modeling capabilities. Therefore, relaxation of the restriction will be considered in the future, thereby the method in the paper can be universally applicable.

References
[1] W. M. P. VAN DER AALST. THE APPLICATION OF PETRI NETS TO WORKFLOW MANAGEMENT [J]. Journal of Circuits Systems & Computers, 2011, 8(01):21-66.
[2] Massuthe P, Reisig W, Schmidt K. An Operating Guideline Approach to the SOA [C] South-East European Workshop on Formal Methods. 2005:35--43.
[3] Wang Jing, Hu Hao, Yu Ping, etc. Modeling of cross-organizational process by combining public view and object Petri net [J]. Computer Science and Exploration, 2014, 8(1):18-27.
[4] Chebbi I, Dustdar S, Tata S. The view-based approach to dynamic inter-organizational workflow cooperation [M]. Elsevier Science Publishers B. V. 2006.
[5] Aalst W M P V D, Basten T. Inheritance of workflows: an approach to tackling problems related to change [J]. Theoretical Computer Science, 2002, 270(1-2):125-203.
[6] Basten T, Aalst W M P V D. Inheritance of behavior [J]. Journal of Logic & Algebraic Programming, 2001, 47(2):47-145.
[7] Robin Milner, Milner, Lin Huimin. Communication and mobile system: π calculus [M]. Tsinghua University Press, 2009.
[8] Aalst W M P V D, Lohmann N, Massuthe P, et al. From Public Views to Private Views - Correctness-by-Design for Services [M]. Web Services and Formal Methods. Springer Berlin Heidelberg, 2008:139-153.