

Study on Complexity Metrics of Business Process

Huan Sun, Hong Hou

College of Information Science and Technology, Northwest University,
Xi'an, China

shuan0529@126.com; hhong66@sina.com

Abstract—The understandability of business process is closely related to its complexity, thus, calculating and controlling the complexity of the business process is becoming important. According to the idea of information flow method proposed by shepperd in the software engineering, we present a method named information flow complexity of business process(IF), and evaluate it using Weyuker framework. Finally, comparing IF metric with existing business process complexity measurement metrics and analyzing the difference between IF method and other metrics using two examples.

Keywords-Business Process; Complexity Measurement; Measurement Method; Measurement Framework

I. INTRODUCTION

Business Process Management System (BPMS) has the ability of define and manage the business process, 38 percent of the business process designed by BPMS is needed to redesign^[1]. One of the main reason of this situation is the high complexity of business process executed cross-organization, in general, complexity is closely related to the understandability of the business process, and the higher complexity, the lower understandability. In the process of software development, business process is bridge of the communication among all stakeholders and easier understanding business process can guarantee the quality of communication, and therefore producing high quality software products. Business process metric is the measurement method for quantitative analysis of the characteristics of business processes to improve the execution efficiency, reduce the complexity and enhance the understandability of business process. So we use the business process complexity metrics to measure and control the complexity, and improve the understandability of the business process.

To our knowledge, any one method can't comprehensively measure the complexity of business process. Therefore, we can measure it from four aspects: activity, control-flow, data-flow and resources which are proposed by Cardoso in paper [2]. The following section will present the existing complexity metrics grouped by author^[3-9]. And table 1 is the summary of complexity metrics.

Due to the similarity of software engineering and business process, experts such as Jorge Cardoso, Volker's and Aalst modified and adapted the existing complexity metrics in software to make them suitable for measuring the complexity of the business process. At present most of the business process complexity metrics are the application of

software metrics without considering the relationship between the different task modules in business process. In fact, we can't ignore the links between tasks as it reflects the structure, influences the complexity of business process. To solve the problem above mentioned, we present a metric reflect the complexity from the information flow named information flow complexity of business process (IF), which is proposed according to the idea of information flow method in the software engineering^[10], and evaluate it using Weyuker framework.

II. INFORMATION FLOW COMPLEXITY METRIC

A. Definition of business process complexity

Complexity of business process is the ability to understand, analysis and interpretation of business process^[1]. Generally, it is more difficult to understand with a higher complexity. In the field of software engineering, the study of software complexity metrics is quite mature, proposing a lot of software complexity metrics and applying it to industry^[11-13]. While the studies of the complexity of the business process metrics are in its initial stage, there is no much published work about analyzing the complexity of BPM^[14].

TABLE I. SUMMARY OF COMPLEXITY METRICS OF BUSINESS PROCESS

Author	Metrics in software	Metrics in business process
Cardoso	LOC	NOA, NOAC, NOAJS
	Information flow	IC
	Halstead metric	HPC
	Graph-oriented metrics	CNC, CI, RE
	Cyclomatic number	CFC
Gruhn, Laue	CFS	CW
	Fan-in/Fan-out	Fan-in/Fan-out
		Max/mean nesting depth, Number of handles
		(anti)patterns for BPM
Aalst	Cyclomatic number	ECaM, ECyM
		SM
Vanderfeesten	cognitive complexity	CC
Mendling	Graph theory	$SN(G)$ 、 $\Pi(G)$ 、 $E(G)$ 、 $\Phi(N)$ 、 $CYC(N)$ 、 $TS(G)$

B. Information flow complexity for business process

We can use BPEL, UML activity diagram, YAWL, and EPML modeling language to model business process when business process modeling. In this article we use the YAWL language to descript business process model.

Shepperd has given some definitions used in the information flow metric in paper [10] such as fan in and fan out, in this article the corresponding definitions are defined as follows:

Definition 1: The fan in of task A in the business process refers to the number of the aggregate branch input to the task (NOIT).

Definition 2: The fan out of task A in the business process refers to the number of the split branch output to the task (NOOT).

Put it simply, the fan in of the task in business process which is described by YAWL is the information input into it, and the fan out of the task is the information output it. We have to first calculate the value of fan in and fan out of every task when use IF metric to measure the complexity for business process. The specific steps are as follows:

Step 1: Calculate the complexity of every task module without considering the interaction of task according to (1), and the letter i presents the i th task in business process.

$$IF_i = (NOIT * NOOT)^2. \quad (1)$$

Step 2: Atom tasks and tasks without subtask can directly calculate the complexity using formula (1). While the complexity of the task with subtasks is calculated by the sum of complexity of every subtask contained.

Step 3: The complexity of the business process is the sum of complexity of all the tasks in it, which is calculated by (2). When calculate it using (2), the complexity of the subtasks can't calculate repeatedly. In equation (2) n presents the number of tasks, and i presents the i th task

$$IF = \sum_{i=1}^n IF_i. \quad (2)$$

As it is described above, we can calculate the complexity of the business process and the number of tasks through step1, step 2 and step 3.

C. The application of IF

In section 3.1 we describe the complexity metric IF and the specific steps of how to use IF to calculate the complexity for business process. Here we will give a travel booking business process described in YAWL as depicted in Fig.1, and the concrete realization of calculating the

complexity of every task in business process is shown in table 2.

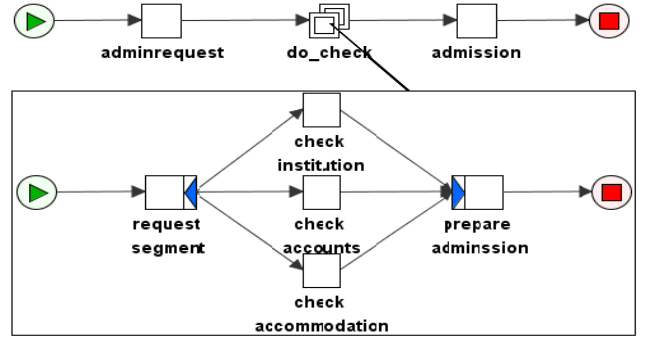


Figure 1. Business process of new student admission.

As is shown in table 2, there are 7 tasks in the business process, and the second and third lines in table 2 are respectively corresponding to the fan in and fan out of every task, and the last line represents the complexity of each task. In Fig.1, the task do_check is a composite task, so its complexity is the sum of all the tasks in the sub process. The complexity of the business process is calculated as $IF=1+21+1=23$, the sum of the complexity of every task.

III. VALIDATION OF IF

In software metrics, the function of measurement and method in software engineering is increasingly weakened, and the reason for the result is lacking of validation for business process [1]. Thus, We validate IF metric using Weyuker framework to avoid this problem. Weyuker framework is an analytic method which is widely used in the software metrics, and it based on formal theory with nine theorems to evaluate metrics [15]. Due to the similarity of software engineering and business process, we validate IF metric using Weyuker framework and decide whether it can measure the complexity for business process. Now we describe the Weyuker framework and validate the IF metric.

Theorem 1: There are two business processes P , Q with different complexity, so the complexity of P is not equal to the complexity of Q .

$(\exists P)(\exists Q)(|P| \neq |Q|) (|P| \text{ and } |Q| \text{ represent the value of complexity, the same as follows})$

This theorem is the basic of other theorems. It asserts that a good metric should distinguish different business processes, and return different values for different business processes. IF satisfies theorem 1 as it can return different value for different business processes.

TABLE II. THE COMPLEXITY OF EVERY TASK IN BUSINESS PROCESS OF STUDENT ADMISSION

Name	Admin Request	Do_check					Admin
		Request segment	Check institution	Check accounts	Check accommodation	Prepare admission	
NOIT	1	1	1	1	1	3	1
NOOT	1	3	1	1	1	1	1
$(NOIT * NOOT)^2$	1	9	1	1	1	9	1
21							

Theorem 2: There only a finite number of business processes P makes its complexity be C based on a hypothesis that C is nonnegative.

(let C be an nonegative number, then there are only finitely processes for which $|P| = C$)

This theorem asserts that a good metric must detect the changing of a business process. IF satisfies this theorem as it can detect the changing when the number of tasks changes in a process, however, it not satisfies this theorem as the task type changes but with the same number of tasks.

Theorem 3: There exist two different business processes with the same complexity.

$$(\exists P)(\exists Q)(P \neq Q \text{ and } |P| = |Q|)$$

This theorem asserts that two different business processes with the identical influence, for such processes, a good metric should return the same value. Two processes P, Q with the same structure, same task number and interface have the same complexity value, and IF satisfies theorem 3.

Theorem 4: Two business processes P, Q with the same function have different complexity.

$$(\exists P)(\exists Q)(P \equiv Q \text{ and } |P| \neq |Q|)$$

This theorem asserts that two business processes of the same function may not has the same internal structure, that is to say, one function has different ways to realize it. In fact, the way to realize a function decides the complexity of the business process. As is shown in above, we know that a good metric should measure a business process from the internal structure of a business process. IF metric can detect the change as one of the fan in, fan out and number of tasks changes. We can conclude that IF satisfies theorem 4.

Theorem 5: For all processes $P, Q, P:Q$ represents the combining of the P and Q , and the complexity of $|P:Q|$ is more than or equal to the complexity of $|P| + |Q|$.

$$(\forall P)(\forall Q)(|P| + |Q| \leq |P:Q|)$$

This theorem asserts that the interacting of two business processes may produce extra or zero complexity. IF satisfies theorem 5 as the complexity changes in the interacting of two business processes when the fan in and fan out change.

Theorem 6: There exist business processes P, Q , and R, P has the same complexity with Q , the complexity of the new business process $|P:R|$ is not equal to $|Q:R|$.

$$(\exists P)(\exists Q)(\exists R)(|P| = |Q| \text{ and } |P:R| \neq |Q:R|)$$

This theorem asserts that there has two business processes with the same complexity, but when combining to the third business process, the resulting business processes have different complexity which is decided by the internal structure of the new business processes. IF can detect the change in the interacting of business processes, so it satisfies it.

Theorem 7: Business processes P, Q is composed of the same elements with the same number and the same type, $|P|$ is not equal to $|Q|$ for which the order of these elements is different.

$(\exists P)(\exists Q)$ (if Q is formed by permuting the order of the tasks of P , then $|P| \neq |Q|$)

This theorem asserts that the order of the business processes elements affect the complexity. Two identical processes have different complexity when we use IF

measuring them, because the fan in and fan out of one task are different. Thus IF satisfies it.

Theorem 8: There exist two business processes P, Q , they have the same complexity if P is the rename of Q .

$$(\exists P)(\exists Q)(\text{if } P \text{ is a renaming of } Q, \text{ then } |P| = |Q|)$$

This theorem asserts that the rename of processes can't change its complexity as complexity of business process is not related to the name. IF satisfies theorem 8.

Theorem 9: There exist two business processes P and Q for which $|P:Q|$ is more than $|P| + |Q|$.

$$(\exists P)(\exists Q)(|P| + |Q| < |P:Q|)$$

This theorem asserts that interaction between parts of the business process can produce extra complexity. The complexity grows when the interaction happens or new node adds, and all these factors can change the complexity when we measure it using IF metric, so IF satisfies theorem 9.

Through evaluation and validation for IF metric use Weyuker framework, we know that IF is an effective metric for complexity measurement as IF satisfies the nine theorem in majority of the situation. Using Weyuker framework validates IF metric just from the theory aspect, we also need to validate it in practical, it's also the work we should do next.

IV. ANALYSIS OF IF

In this section we give two business processes as shown in Fig.2 and Fig.3. Then we calculate the complexity of business processes from different aspects using IF metric and other existing complexity metrics, analyzing it according to the results and finding the advantages.

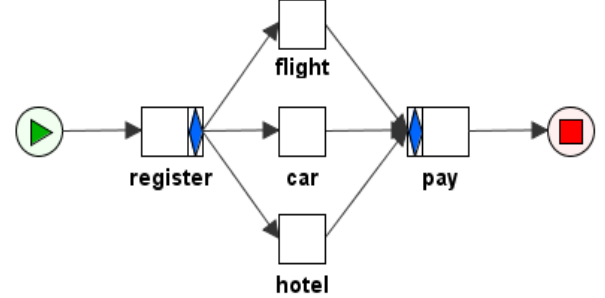


Figure 2. Business process of travel booking.

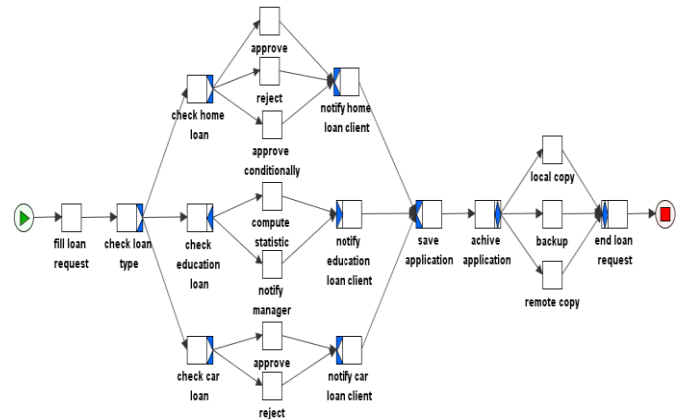


Figure 3. Business process of loan application.

The result of measurement for business process in Fig.2 and Fig.3 using IF, NOA, CFC and Nesting depth is shown in table 3.

TABLE III. THE RESULT

Method	Travel Booking	Loan Application
IF	21	81
NOA	5	21
CFC	7	16
Nesting depth	1	2

Business process in Fig.3 is complex than that in Fig.2 as it looks, and from the table 3 we can see that the result is right as the value in third column is larger than that in the second column. NOA calculates complexity through the number of task, but it can't rightly express the complexity for sequential tasks. From comparing other three metrics, we know that IF which is more easily detecting the changes is more sensitively than others. In addition, IF measures complexity from aspects of fan in and fan out, however, other metrics consider none or only one aspect of it. As is described above we conclude that IF is a good metric for measuring complexity for business process.

V. CONCLUSION

The understandability of business process is closely related to its complexity, so we should control the complexity of business process for better understanding. According to the idea of information flow method proposed by shepperd in the software engineering, we present a new, adaptive method named information flow complexity of business process, and evaluate it using Weyuker framework, and the experimental validation is what we will do next step. We also will study the application of measurement framework integrated with different complexity metrics, measuring business process comprehensively to control it and for better understandability.

ACKNOWLEDGMENT

I would like to show my deepest gratitude to my supervisor, Prof. Hou Hong, and she has provided me with valuable guidance in every stage of writing of this paper. I will extend my thank s to all people in our Lab. for their help and concern.

REFERENCES

- [1] J. Cardoso, "Evaluating the process control-flow complexity measure," Web Services, 2005. Conference on web Services. Orlando, USA:IEEE press,2005, pp.803-804.
- [2] J. Cardoso, "Control-flow complexity measurement of processes and Weyuker's properties," 6th International Enformatika Conference. Vol. 8, Oct, 2005, pp.213-218.
- [3] G. M. Muketha, A. A. A. Ghani, M. H. Selamat, and R.Atan, A survey of business process complexity metrics, Information Technology Journal, vol. 7, Sept. 2010, pp.1336-1344.

- [4] J. Cardoso, "Process control-flow complexity metric: An empirical validation," Services Computing, 2006. SCC'06. IEEE International Conference on. IEEE press, sept. 2006, pp.167-173.
- [5] J. Cardoso, J. Mendling, G. Neumann, and H. A. Reijers, "A discourse on complexity of process models," Business process management workshops. Springer Berlin Heidelberg, Jan. 2006, pp.117-178.
- [6] Gruhn, Volker, and Ralf Laue, "Adopting the cognitive complexity measure for business process models," Cognitive Informatics, 2006. ICCI 2006. 5th IEEE International Conference on, IEEE press, Vol. 1.July.2006, pp.236-241.
- [7] K. B. Lassen, and Wil M. P. van der Aalst," Complexity metrics for workflow nets," Information and Software Technology, IEEE press, vol. 3, 2009, pp. 610-626.
- [8] I. Vanderfeesten, J. Cardoso, J.Mendling, Hajo A. Reijers, and Wil van der Aalst, "Quality metrics for business process models," BPM and Workflow handbook, 2007, pp.179-190.
- [9] J. Mendling, and G. Neumann, "Error Metrics for Business Process Models," In CAiSE Forum, 2007, pp. 53-56.
- [10] D. C. Ince, and M. J. Shepperd, "An empirical and theoretical analysis of an information flow-based system design metric," ESEC'89. Springer Berlin Heidelberg, 1989, pp. 86-99.
- [11] FENG Da-cheng, CHEN Li-rong, "Complexity-based software defect prediction," Computer Engineering and Design, vol. 32, 2011, pp.213-217.
- [12] CHEN Yu, "Research based on the complexity metrics to improve the quality of software design," Northwestern Polytechnical University, 2007.
- [13] ZHANG Kun, "Software Complexity Measurement and Control in Space Transportation System," Measurement & Control Technology in Aerospace, 2008, pp.011-012.
- [14] Gruhn, Volker, and Ralf Laue, "Complexity metrics for business process models," Proc. ICBIS(06), IEEE Press, Sept. 2006, pp.1-12.
- [15] Weyuker, and J. Elaine, "Evaluating software complexity measures," Software Engineering, IEEE press, vol. 9,1988, pp.1357-1365.