Computers in Industry 65 (2014) 333–344

Contents lists available at ScienceDirect

Computers in Industry

journal homepage: www.elsevier.com/locate/compind

Workflow performance analysis and simulation based on multidimensional workflow net

Liu Sheng[a,c,*], Fan Yushun[b]

[a] State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, 100190 Beijing, China
[b] Department of Automation, Tsinghua University, 100084 Beijing, China
[c] Cloud Computing Center, Chinese Academy of Sciences, 523808, Dongguan, China

ARTICLE INFO

Article history:
Received 20 November 2013
Accepted 5 December 2013
Available online 27 December 2013

Keywords:
MWF-net
Performance analysis
Dwelling time
Probability density

ABSTRACT

Workflow model performance analysis plays an important role in the research of workflow techniques and efficient implementation of workflow management. Instances dwelling times (*IDT*) which consist of waiting times and handle times in a workflow model is a key performance analysis goal. In a workflow model the instances which act as customers and the resources which act as servers form a queuing network. Multidimensional workflow net (*MWF-net*) includes multiple timing workflow nets (TWF-nets) and the organization and resource information. This paper uses queuing theory and *MWF-net* to discuss mean value and probability distribution density function (*PDDF*) of *IDT*. It is assumed that the instances arrive with exponentially distributed inter-arrival times and the resources handle instances within exponentially distributed times or within constant times. First of all, the mean value and *PDDF* of *IDT* in each activity is calculated. Then the mean value and *PDDF* of *IDT* in each control structure of a workflow model is computed. According to the above results a method is proposed for computing the mean value and *PDDF* of *IDT* in a workflow model. Finally an example is used to show that the proposed method can be effectively utilized in practice.

1. Introduction

Workflow technology is becoming increasingly important for achieving a process oriented view of the organization and subsequently process automation. Workflow management systems (WfMS) prove to be an effective means realizing full or partial automation of a business process [1]. Confronted with globalization and ever increasing competition, Quality of Service (QoS) requirements on WfMS, like performance, soundness, and availability, are of crucial importance. Businesses must ensure that the systems they operate not only provide all relevant services, but also meet the performance expectations of their customers. To avoid the pitfalls of inadequate QoS, it is necessary to analyze the expected performance characteristics of WfMS and workflow models. The methods used to do this are part of the discipline called Performance Engineering [3].

A business process is a set of one or more linked procedures or activities that collectively realize a business objective or policy goal, normally within the context of an organizational structure defining functional roles and relationships [1]. Despite the abundance of workflow management systems developed for different types of workflow based on different paradigms [4–7], the lack of rigorous theoretic foundation and then effective model verification and analysis methods has blocked workflow techniques' research and application [15,35,36].

The rationality and correctness analysis should be carried out from four aspects that are relevant for workflow modeling and workflow execution: process control logic, timing constraint logic, resource dependency logic, and information dependency logic [15,34]. The correctness analysis of process control logic aims to avoid the deadlocks or structural conflicts in the execution of a workflow model caused by the errors in its process control. Some verification and conflict detection methods have been discussed in [2,5,8,10,35,41,43,44]. The objective of resource dependency logic verification is to prove correctness of the static or dynamic resource allocation rules and consistency with the process control logic. The information dependency logic cares about the internal consistency of a workflow-related data and the correctness of temporary relation among different workflow application data. The timing constraint verification and analysis deal with the temporal aspects of a workflow model such as deadlines [9,11,36], time scales [12,13,34,37–39,42], schedulability analysis [33], and boundedness verification [14] and time violation handling [16,17]. Quality of Service in Flexible

* Corresponding author. Tel.: +86 15910634676.
E-mail address: liusheng7801@163.com (S. Liu).

0166-3615/$ – see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.compind.2013.12.002

Workflows is discussed in [40]. A workflow net similarity measure method is introduced in [51].

The above analysis can ensure only the functionally working workflow (correctness) but not its operational efficiency. The performance level [15–24,31,53], on the other hand, aims to evaluate the ability of the workflow to meet requirements concerning some key performance indicators such as, maximal parallelism, throughput, service levels, and sensitivity. The analysis of resource availability and utilization, and average turnaround time is performed at this level. Performance analysis of workflow is of great importance in both enterprise applications [54] and scientific computing [52]. Yet it has not got enough attention of researchers commensurate with its importance until now [29]. The performance analysis of a workflow model (business process) is different from that of WfMS architecture [25,26].

The performance analysis can be conducted only after the rationality and correctness analysis has been carried out. So it is assumed that there are no temporal and logical errors in the considered workflow models at the performance analysis stage.

PN (*Perti Net*) are the only formal techniques able to be used for structural modeling and a wide range of qualitative and quantitative analysis [29]. *PN*-based workflow management systems are widely used because of formal semantics, local state-based system description, and abundant analysis techniques [27]. So PNs are a naturally selected mathematical foundation for the formal performance analysis of workflow models. Many researchers use *PN* techniques to study workflow [4,5,7–10,14,18–22] since Zisman used *PN* to model workflow processes [28].

A *PN* is a graphical and mathematical modeling tool. It consists of places, transitions, and arcs that connect them. Input arcs connect places with transitions, while output arcs start at a transition and end at a place. There are other types of arcs, e.g. inhibitor arcs. Places can contain tokens; the current state of the modeled system (the marking) is given by the number (and type if the tokens are distinguishable) of tokens in each place. Transitions are active components. They model activities which can occur (the transition fires), thus changing the state of the system (the marking of the Petri net). Transitions are only allowed to fire if they are enabled, which means that all the preconditions for the activity must be fulfilled (there are enough tokens available in the input places). When the transition fires, it removes tokens from its input places and adds some at all of its output places. We usually use a bar to represent a transition, a circle to represent a place, and a dot to represent a token.

PNs which model workflow process definition are called *WF-nets* (*Workflow nets*) [4,32]. A *PN* is called a *WF-net* if and only if:

(1) *PN* has two special places: a source place and a sink place. The source place has no input transitions while the sink place has no output transitions; and
(2) If we add a new transition to *PN* which connects source place with the sink place, then the resulting *PN* is strongly connected.

A *WF-net* presents only process control specification of a workflow model. In order to perform its time dimension verification and analysis, its specification should be extended to express its temporal behavior. Various works [12,14,46–50] introduce time into *PN*-based workflow models. Based on the semantics of *Time Petri Net* (*TPN*), *Time Workflow net* (*TWF-net*) [12,46,47] is proposed by regarding a timing constraint as a delay pair consisting of its lower and upper bounds. The definitions and notations of *TWF-net* coming from [12,46,47,50] is briefly introduced here.

TWF-net is a three tuple (*WF-net*, *FI*, *M*), where *WF-net* is a *Workflow net*. *WF-net* is also a three tuple (*P*, *T*, *F*). $P = \{p_1, p_2, \ldots, p_m\}$ is a set of places representing the state of a instance or the condition of its output transitions; $T = \{t_1, t_2, \ldots, t_n\}$ is a set of transitions representing activities of the workflow model; *F* is a set of directed arcs linking places and transitions, and employed to describe precedence relations among activities; *FI* is a set of nonnegative real number pairs $[l, u]$ related to each transition, which is used to represent the minimum firing time and the maximum firing time respectively; *M* is a vector of *m*-dimensional markings where $M(p)$ denotes the number of tokens representing the number of instances in *p*.

There are usually two types of transitions in *TWF-net*, i.e., activity transitions and routing transitions. The former ones represent the activity nodes in a workflow model. The latter ones determine the control structures among former ones, e.g., and-split, and-join, or-split and or-join. Routing transitions are associated with a time interval [0,0] because they fire once they are enabled. For simplicity the time interval tags of routing transitions are omitted. Assume transition *t* is associated with a time interval $[l, u], (0 \le l \le u)$. And let *s* and $\tau(t)$ denote the enabled time and the actual firing time of *t*, respectively. We have $s + l \le \tau(t) \le s + u$.

The definition of *MWF-net* (*Multidimensional Workflow net*) is proposed by [15]. *MWF-net* describes the relations between multiple workflow processes, and the resource and organization structure they share. It is a five tuple (W, O, R, F_P, F_R) where *W* is a set of *TWF-nets*. *O* is a set of roles defined in the organization perspective while *R* is a set of resource pools defined in the resource perspective; F_P describes mapping relation between process perspective and organization perspective while F_R represents binary relation between organization perspective and resource perspective.

Methods are discussed to compute the workload that arrival instances generate for the various resource pools and the lower bound of average turnaround time of instances [15]. This paper adopts *MWF-nets* [15] as a base mechanism to represent a performance analysis oriented workflow model.

2. Related works

A high-level stochastic *PN* (*SPN*) is used to model the routing constructs of a workflow, and then a method to compute throughput time of the process is presented [20]. Based on four performance equivalent formulae, the performance of a workflow is approximately analyzed in [21]. These two techniques both aim at calculating instances' execution time and ignoring waiting time. The probability density of execution time is not taken into account, and cannot be applied to a workflow process of which the resources have stochastic service time. All the control structures are mapped into *Generalized stochastic PN* (*GSPN*) [19,22], and then a method based on a *CTMC* (*continuous time discrete state Markov chain*) to obtain lower bounds of the execution performance is discussed. A so-called load equivalence aggregation model derived from GSPN has been developed in [18], and then some performance related measures of human resources in a workflow by obtained by simulating the model. By defining change time, a performance evaluation model for the dynamic workflow changes is brought forward in [23]. However, the technique can be used for only acyclic time WF-net in which the arrival intervals of instances are constant. A queuing network is used to model the workflow [24,31]. A method is yielded to identify the critical path of a workflow model and determine the minimum number of servers for the critical activity [24]. Some approximate approaches are employed in [31] for workforce configuration, and then the corresponding network is analyzed. But these