Journal of Advances in Computer Research
Quarterly pISSN: 2345-606x eISSN: 2345-6078
Sari Branch, Islamic Azad University, Sari, I.R.Iran
(Vol. 7, No. 1, February 2016), Pages: 89-100
www.jacr.iausari.ac.ir

Modeling and Performance Analysis of Restaurant's Service Delivery Process Using Colored Petri Nets

Samaneh Ghasemi[✉], **Zeinab Ghorbani Lakterashani**

Department of Computer Engineering, Sari Branch, Islamic Azad University, Sari, Iran

sghasemi915@gmail.com; zn.ghorbani@gmail.com

Received: 2014/05/18; Accepted: 2015/01/13

Abstract

Process mining includes the automated discovery of Petri nets from event logs. These event logs are the starting point for various discovery and analysis techniques of the process. In this paper process model is developed using colored Petri nets (CPNs) and the supporting software package CPN Tools. The restaurant process model is represented for the analysis *using a combination of process mining and simulation. The goal of process mining is deducing knowledge (e.g. a process model, average throughput time) from event logs. The knowledge obtained this way can increase understanding of the processes and can help with redesigning them if needed. The bottlenecks can be detected with Process mining.* We evaluate the *indicators performance, such as throughput time, for different alternative process. Analysis results show that that what type of changes may be beneficial.*

Keywords: *Business process, Colored Petri nets, Performance analysis, Process mining, Bottlenecks*

1. Introduction

Process mining can provide useful insights into running business processes of organizations[6]. By emphasizing business processes rather than hierarchies and that places a special focus on outcomes and customer satisfaction, organizations tend to attain better overall performance, a better esprit de corps and less inter-functional conflicts[1].Business processes usually are modeled as computer-based, dynamic, stochastic, and discrete simulation models. The most common way to represent these models in a computer is using discrete-event simulation[2]. One of the major strengths of simulation modeling as a tool for business process design is that it can help reduce the risks inherent in any type of change. The use of scenario-based what-if analyses enables the design team to test various alternatives and choose the best one [2].

Colored Petri nets are a discrete-event modeling language which is based on Standard ML. CPNs allow for the modeling of complex processes [3][4].

CPN Tools is an industrial strength computer tool for constructing CPNs models. Using CPN Tools, it is possible to investigate the behavior of the modeled system using simulation, to verify properties by means of state space methods and model checking, and to conduct simulation based performance analysis [3]. CPN Tools[1] have been applied in various industrial projects[5].

[1] More information about CPN Tools is available at http://cpntools.org

In this paper, we analyze process model using a combination of simulation and process mining. The objective is to give an approach for analyzing the performance of process and to show how a choice between alternative designs using best practices can be supported.

Process mining techniques can reveal process models and can be combined with other techniques such as simulation. One aspect of process mining is control-flow discovery that is, automatically constructing a process model (e.g., a Petri nets) describing the causal dependencies between activities. Such discovered business processes have proven to be useful for the understanding, redesign, and continuous improvement of business processes. End users can see where business processes deviate from normative business processes, detect bottlenecks, and explore undesired phenomena[6].

The paper is organized as follows. First, we review related work in section 2. Then, section 3 presents a restaurant's service delivery process modeling and simulation that is used throughout this paper. Afterwards, *process mining techniques* are described in section 4.

2. Related work

The work presented in this *paper is related* to *process mining*, i.e., discovering a process model based on some event log. Process mining techniques focus on discovering behavioral aspects from log data. The idea of applying process mining in the context of workflow management was first introduced by Agrawal et al.[7].A similar idea was used in the context of *automating* the discovery of process models using probabilistic and algorithmic approaches by Datta[8].Cook and Wolf in[9]describe a Markov method *that they developed* specifically for process discovery. This model is inherently sequential. In[10] the exploring techniques that can use basic event data captured from an on-going process to generate a formal model of process behavior. Term this kind of data analysis process discovery. They describe using methods: algorithmic grammar inference, Markov models, and neural networks. Note that the results presented in[10] are limited to sequential behavior. Cook and Wolf extend their work to concurrent processes in[11]. They propose specific metrics (such as entropy, event type counts, and causality) and use these metrics to discover models out of event streams. However, they do not provide an approach to generate explicit process models. In [12] the authors show, how to apply region based methods for the synthesis of Petri nets from languages to process mining. Rubin et al. [13] specifically look at process discovery in the context of software processes. The *authors show* that a process mining framework can be used for obtaining process models. In [14] deals with the problem of discovering protocol models by analyzing logs. However, all these techniques do not aim at simulation.

Simulation has been used for the analysis of business processes since the seventies [15].*When reengineering a process from scratch or when improving an existing process design, simulation can be very valuable* [16].The work reported in [17] is modeling and analyzing of process through simulation of CPNmodel. The authors present a short sample process modeled in CPN but they not use process mining techniques. In[18] the CPN export of ProM is described but not focus on process mining and no case studies are given. There is quite some work on the automatic generation of Petri net models for workflows. In[19] the models created by Protos2CPN transformation which allows the simulation of Protos models in CPN Tools.

The work reported in this paper is the automated discovery of Petri nets from event logs which the starting point for analysis of the different alternative with process improvement goals.

3. Process modeling and simulation

In this section, we describe how the process modeling and simulation is supported by CPN Tools.

3.1 CPN model

When modeling a system by means of a classical Petri net, elements of this system should represented as tokens, places, transitions and connections. The places are used to represent the state of the modeled system. Each place can be marked with one or more tokens .Often it is the case that, in a process model, it is desired to difference between these tokens. In classical Petri net, however, it is not possible to describe the attributes of the token. It is therefore natural to extend classical Petri nets in such a way that each token has a data value attached to it. This data value is called the token color [20].

CPNs extend the classical Petri net with data (colored tokens), time, and hierarchy. Places are typed, i.e., all tokens on a place have a value of some common type. In CPN-terms, this means that all tokens in a given place should belong to the same color set. This implies that each place has a color set (i.e., type). Tokens also timestamps indicating when they can be consumed. Complex models can be structured in a hierarchical manner, i.e., nodes at one level may refer to sub processes at a lower level. CPNs are distributed over the so-called pages. One page describes a network of places and transitions and may refer to other pages.[20][4][3]

For a more detailed introduction to CPNs, we refer to [3][20].

A business process as a collection of a number of inter-related tasks, where a limited number of resources able to perform one or more tasks in the process[21]. The task (also named activity, operation, action, or work item) is some operation on the case. The case (also named process instance) is the “thing” which is being handled, e.g., a customer order, a job application, an insurance claim, a building permit, etc[22].Resources are grouped into roles to facilitate the mapping of resources on task[23].

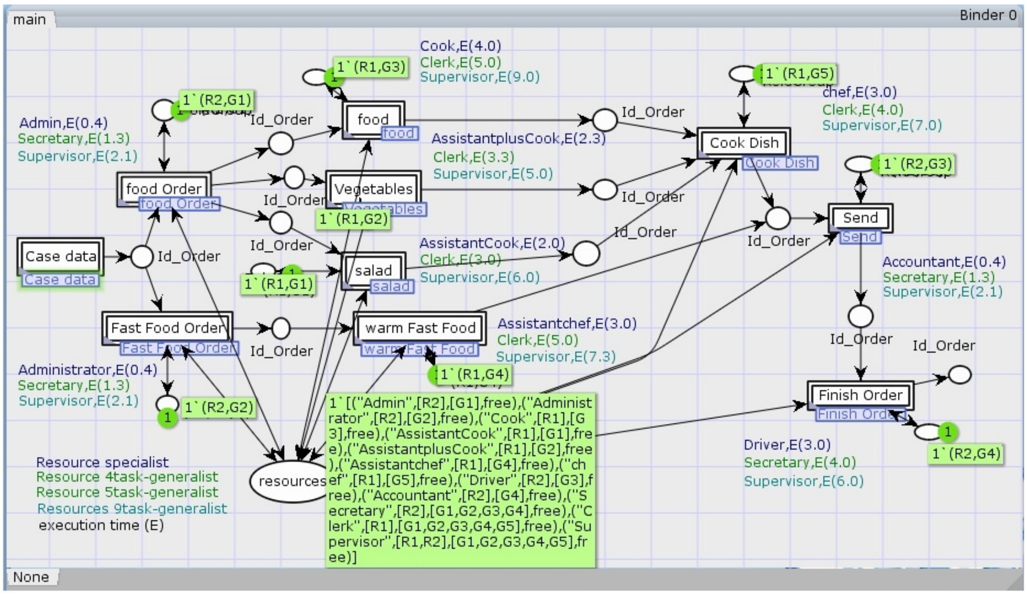

Fig. 1. CPN model of the telephone order process, for each task the execution time (E) is shown

Figure 1 shows the top level or main page (which models the actual execution of tasks) of process model incoming telephone orders and their delivery at a restaurant.

Our CPN model is a hierarchical model that is divided into ten pages. To ensure straightforward modeling of (large) processes a task sub model is developed. In fig. 2 the detailed sub page for the ‘‘food’’ task, showing how the simulation information related to the ‘‘food’’ task for obtain an event log in the CPN model. The sub page Case data creates cases for which a task needs to be performed.

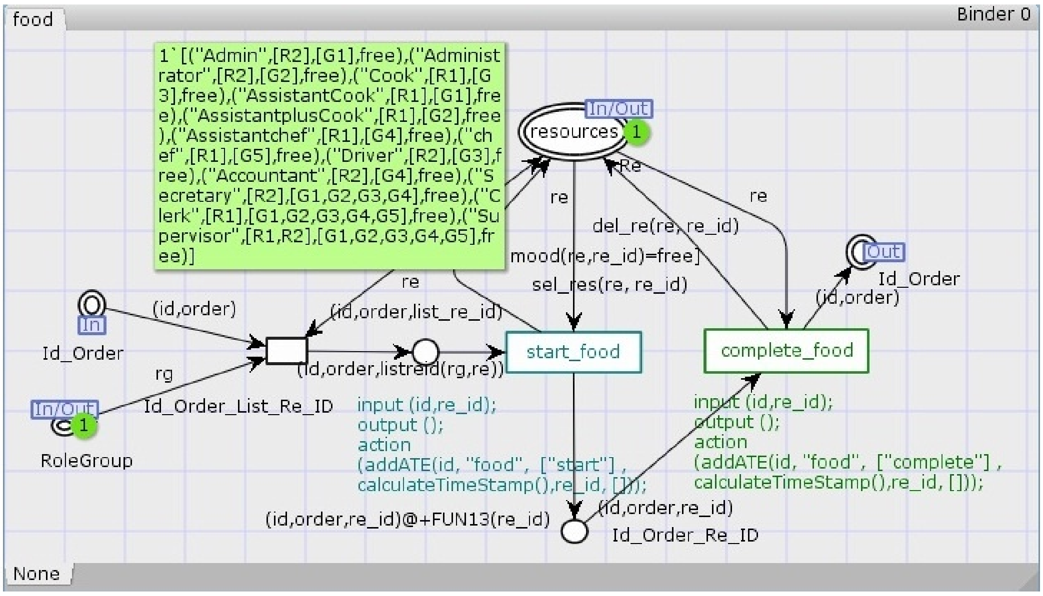

Fig.2.Sub page for the ‘‘food’’ task

Business process improvement is an important means to obtain competitive advantage and improve customer satisfaction. A collection of best practices for business process improvement is given in [24].

In this paper we will see under which conditions two of these best practices (specialist-generalist trade-off and allocation policy) indeed provide performance improvement process.

3.1.1specialist-generalist trade-off

The specialist-generalist trade-off aims at finding the optimal ratio of specialized and generic resources in process. Given a fixed number of resources, the question is how many of the resources should be specialized resources and how many of them should be generic to have an optimal process performance. The specialist as a resource able to perform only one task and using specialists will lower the average execution time spent on a case. A generalist is able to perform more tasks which need more time for execution and using generalists will lower the average waiting time of a case[24].

In fig. 2 we see *three* types of generalists in the restaurant process. There are generalists like the secretary who are able to perform *four* tasks and this type is called 4task-generalist. The initial process model is called process alternative 1 that only specialized resources are available to execute the process.

For the restaurant process we would like to find an alternative process that has a through put time significantly lower than the initial situation with only specialists.

The five process alternatives are stated in table1 with the ratio of specialists, 4task-generalists, 4task-generalists, and 9task-generalists per alternative.

Table 1.Definition of process alternatives

Alt.	Ratio of spec-gen	
1	54	specialists
2	27	specialists
	27	4task-generalists and 5task-generalists
3	18	specialists
	36	4task-generalists and 5task-generalists
4	9	specialists
	15	4task-generalists and 5task-generalists
	30	9task-generalists
5	27	4task-generalists and 5task-generalists
	27	9task-generalists

3.1.2 Allocation methods

Two allocation methods, random allocation and priority based allocation, are modeled. We have chosen to define allocation methods with ML functions. The roles able to perform a task are prioritized and the order in which resources are assigned is based on the priority of their role. For each task the roles able to perform the task are specified. The roles are ordered from high priority to low priority (*see* fig. 1).This random allocation a resource is randomly chosen and allocated. *It means that* available resources with a suitable role all have an equal chance of being selected. Using a flexible assignment policy means that from the available resources the resource with the highest priority role is selected and assigned.. By doing this, more generic resources are 'saved' for other tasks for which this resource might not be suitable. In this way, the waiting time for a suitable resource may be reduced. Another advantage is that the specialized resource is more skilled and will need less time to perform the task [24].

3.2 Generate Event Logs

The event logs are the starting point for various discovery and analysis techniques that help to gain insight into certain characteristics of the process.The simulation was run for 180 and 120 cases(*see fig. 3*).During the simulation in CPN Tools, separate event logs were generated for each case.

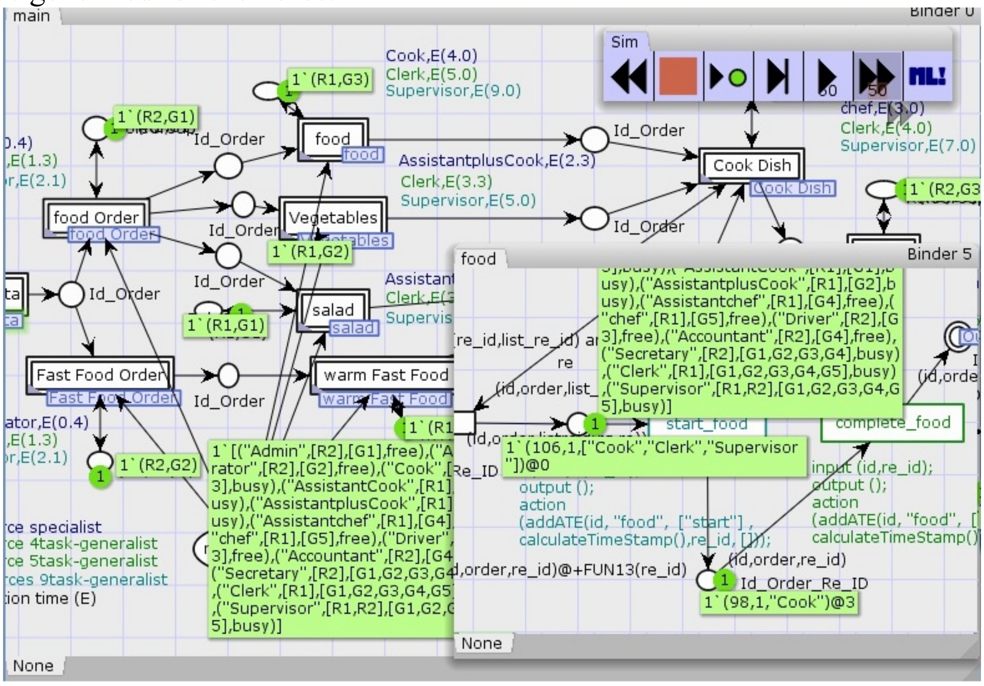

Fig. 3 .Running simulation model in CPN Tools

4. Process Mining

Process mining is one research area to find useful information from *process execution logs*. The process mining techniques are classified in three basic types, namely discovery, conformance and extension [25]. The ProM framework[1] is an extensible tool suite that supports a wide variety of process mining techniques in the form of plug-ins. The ProM framework focuses on business process analysis in the broadest sense-that is, both design time analysis and runtime analysis[6] .

Process mining refers to the monitoring of running process instances (e.g., their progress, bottlenecks and times spent in each task) and their analysis results (e.g., percentage of instances not completing successfully) to inform users about unusual or undesired situations (i.e., alerts) [26] .It targets the automatic discovery of information from an event log. This discovered information can be used to deploy new systems that support the execution of business processes or as a feedback tool that helps in analyzing and improving enacted business processes [27].

4.1. Event Log Transformation

An event log can be converted to a MXML[2] using the CPN Tools plug-in of the ProM import framework, a de facto standard format for process mining that can be used in the ProM framework[18][27][28].As depicted in Fig. 4, the target format of any

[1] More information about this open source framework is available at http://www.processmining.org
[2] a shorthand for “Mining XML”

conversions performed by the ProM import framework.

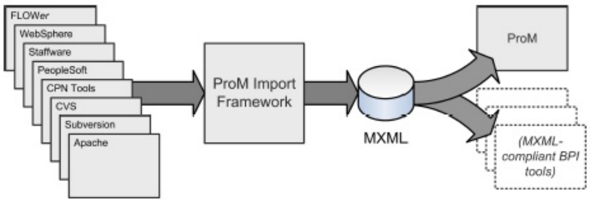

Fig. 4.ProMimport framework for event log conversion [28]

4.2. MXML format

MXML format is based on XML[1].A process log in MXML contains several process instances (i.e., cases), whereas each process instance contains a number of audit trail entries (i.e., events). Each audit trail entry records task name, event type, originator and time stamp[18].

Figure 5 depicts a screenshot of the event log in MXML format. The depicted screenshot shows the details about one of the process instances, which contains fourteen audit trail entries.

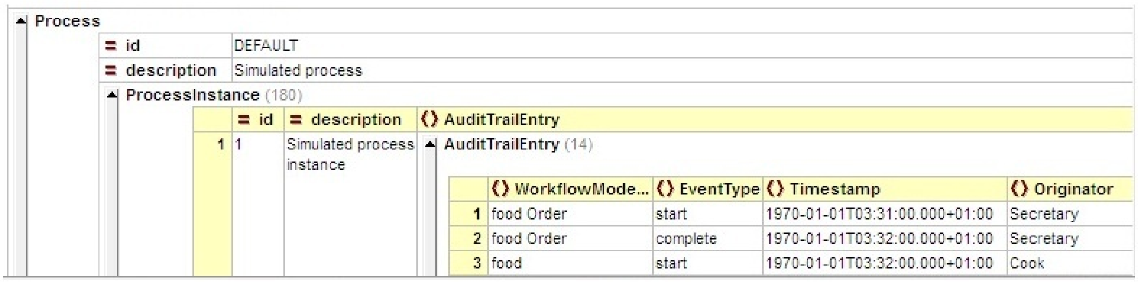

***Fig. 5.Fragment of the example log in MXML format viewed using*XML Spy.**

4.3 Alpha algorithm

Alpha algorithm is one of earliest developed plug-ins in Prom framework. The alpha algorithm scans the event log for particular patterns. For example, if task A is followed by task B, but task B is never followed by task A, then it is assumed that there is a causal dependency between activities A and B. To reflect this dependency, the corresponding Petri net should have a place connecting transitions A and B [6]. We choose the alpha algorithm plug-in, which constructs a process model in terms of a Petri net based on this process model (*see fig. 6*).

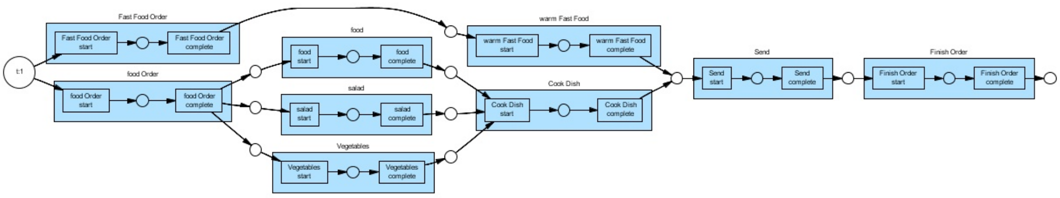

Fig. 6.Petri net discovered by ProM frameworkusing the Alpha algorithm

---

[1] *Extensible Markup Language*

4.4 Performance Analysis of Alternatives

In this section, we describe how the presented discovery approach is supported by ProM framework. We can discover the performance perspective of the process by applying the Performance Analysis with Petri net plug-in. The plug-in evaluates the time stamps in the log and projects the extracted performance information on places and transitions. It graphically shows the bottlenecks by coloring places according to the time that is spent in this part of the process (red indicates a high waiting time, while blue means less waiting time) . A bottleneck in a process is a place with a high average waiting time. A screenshot of the bottlenecks for the restaurant process is shown in Fig.7.

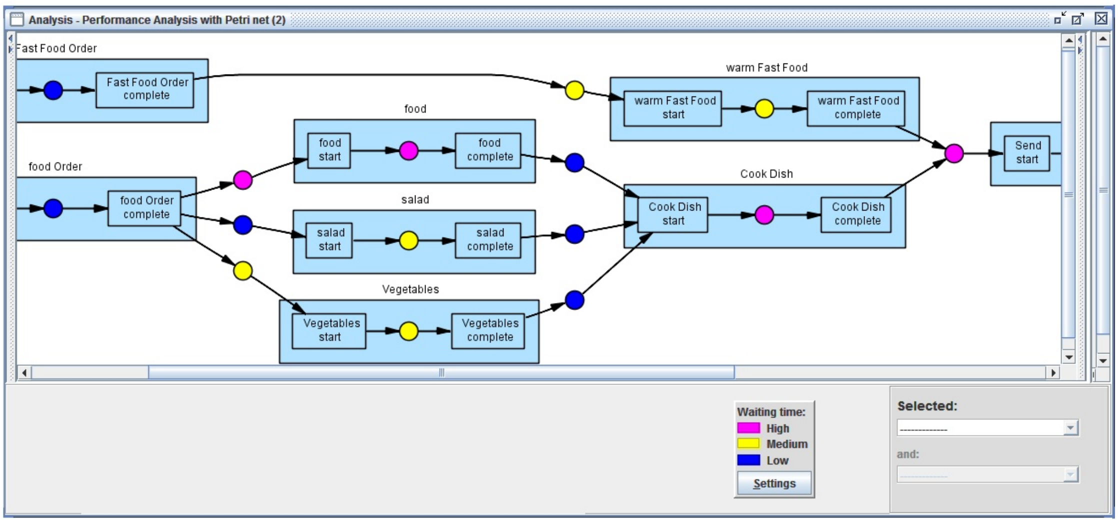

Fig. 7.Performance analysis plug-in

Using this technique, we assessed some important performance metrics of the process. For instance, we were able to extract the average time of execution of each task. In fig. 13(a), we can see several statistics for the waiting, execution time task, which we have selected. The information about execution times and waiting times *some of the activities* for 180 *cases* are shown in fig. 8.

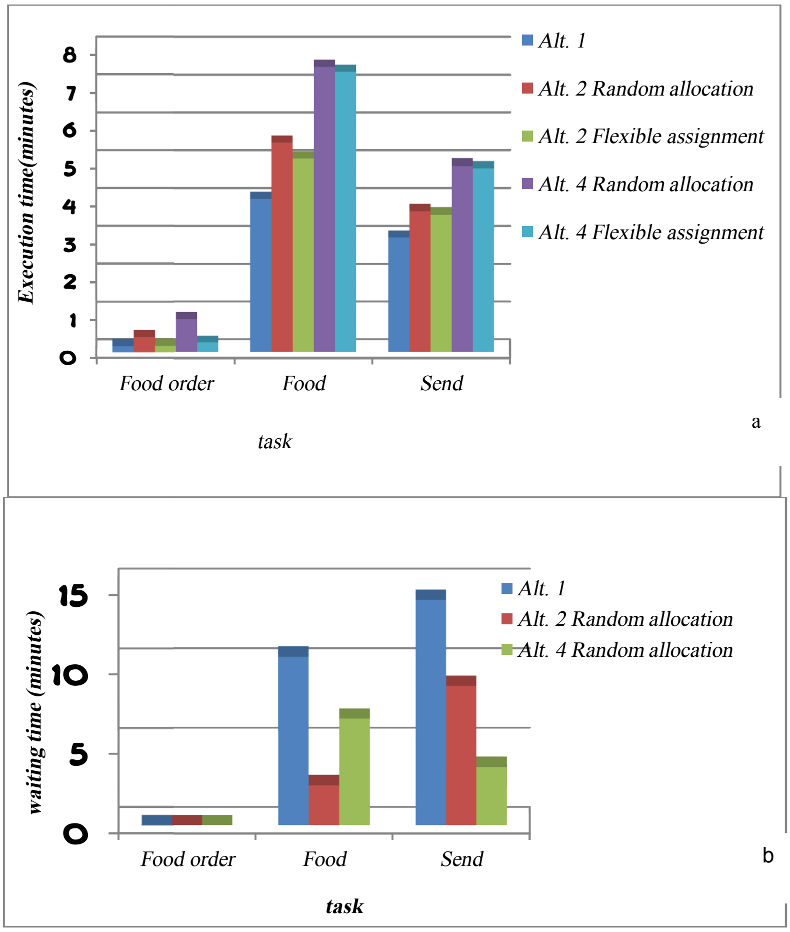

Fig. 8.Performance analysis results on* some of the activities *for 180* cases. *(a) Execution time. (b) Waiting time

Furthermore, we look for improvements of the average CT[1] of the process and this measure is used as performance indicator. CT(also known as throughput time) is one of the most important measures of performance of a business process. This value is frequently the main focus when comparing the performance of alternative process designs. The CT is the time that it takes to complete an individual job from start to finish. In other words, it is the time that it takes for a job to go from the entry point to the exit point of a process. This is the time that a customer experiences. For example, in an Internet ordering process, CT may be the time elapsed from the time a customer places an order until the order is received at home. CT analysis is a valuable tool for identifying opportunities to improve process performance. For example, if an insurance company finds out that it takes 100 days (on average) to process a new commercial account and that the actual processing time is only about 2 days, then there might be an opportunity for a significant process improvement with respect to CT [2].

Table 2 shows the results for each of the alternatives we perform a simulation contains in total 180 cases and 120 cases and measure the average throughput time.

[1]Cycle Time

Table 2 .Results for random allocation and flexible assignment

		Average CT (minutes)	
Alt.	***Allocation method***	***180 cases***	***120 cases***
1	Random allocation / Flexible assignment	*29.18*	23.62
2	Random allocation	25.35	18.15
	Flexible assignment	19.43	17.07
3	Random allocation	22.04	16.59
	Flexible assignment	18.08	16.27
4	Random allocation	31.11	26.52
	Flexible assignment	29.53	20.97
5	Random allocation	33.59	28.49
	Flexible assignment	30.07	23.46

For the alternative 1 (initial process) both allocations will perform the results same, because there are only specialists to allocate. The results show us that improvement of a process for alternative 2 and 3has the lowest average throughput time by using ratio the best trade-off is made between the shorter execution times of the specialists and the flexibility *offered* by a generalist, when compared to the initial process. Table 6shows the results for random allocation and flexible assignment .With random allocation there is a high chance that a generalist would be selected, while flexible assignment would favor the selection of a specialist. The results of the application of flexible assignment to the restaurant process led to an improvement of process alternatives.

5. Conclusions and Future Work

In this paper we developed a restaurant process model for the analysis of process alternatives. The model represents an abstraction of a business process. The process alternatives with application the two of these best practices (specialist-generalist trade-off and flexible assignment policy) for improvement of the process were modeled and evaluated. A combination of automatic discovery process models using ProM Framework and the simulation has been used for measure performance indicator. This simulation model is represented as a CPN. We have found alternatives with the lowest average throughput time, which led to an improvement of the initial process.

CPN uses an integer global variable to represent time. As the simulation evolves, the number of rounding executed increases. The resulting loss of accuracy may cause undesirable effects in complex models. It should be mentioned that CPN has some flexibility to express time behavior, which is controlled by model designers. It also allows for the creation of arbitrary discrete time distributions, being only necessary to implement a random number generator for that distribution.

As a future work, we will use stochastic Petri nets model to give a more precise for analyzing the performance of process model. Furthermore, we see opportunities for the extension of the developed model. Other process characteristics could be added to the model. Next to this more performance indicators could be taken into model. The current model focuses on the timing aspect, but also the performance on cost, quality or flexibility could be interesting.

References

[1] K. McCormack, "Business process orientation: do you have it?," Qual. Prog., vol. 34, no. 1, pp. 51–60, 2001.

[2] M. Laguna and J. Marklund, Business process modeling, simulation and design. CRC Press, 2013.

[3] K. Jensen, L. M. Kristensen, and L. Wells, "Coloured Petri Nets and CPN Tools for modelling and validation of concurrent systems," Int. J. Softw. Tools Technol. Transf., vol. 9, no. 3–4, pp. 213–254, 2007.

[4] J. Kurt, "Coloured Petri nets: Basic concepts, analysis methods and practical use," EATCS Monogr. Theor. Comput. Sci. 2nd Ed. Berlin Springer-Verlag, 1997.

[5] "University of Arhus, Department of Computer Science, CPN Tools, main page," 13-Nov-2014. [Online]. Available: http://cs.au.dk/cpnets/industrial-use/. [Accessed: 22-Dec-2014].

[6] W. van der Aalst and C. Stahl, Modeling business processes: a petri net-oriented approach. MIT press, 2011.

[7] R. Agrawal, D. Gunopulos, and F. Leymann, Mining process models from workflow logs. Springer, 1998.

[8] A. Datta, "Automating the discovery of as-is business process models: Probabilistic and algorithmic approaches," Inf. Syst. Res., vol. 9, no. 3, pp. 275–301, 1998.

[9] J. E. Cook and A. L. Wolf, "Discovering models of software processes from event-based data," ACM Trans. Softw. Eng. Methodol., vol. 7, no. 3, pp. 215–249, 1998.

[10] J. E. Cook and A. L. Wolf, "Automating process discovery through event-data analysis," in Software Engineering, 1995. ICSE 1995. 17th International Conference on, 1995, p. 73.

[11] J. E. Cook and A. L. Wolf, Event-based detection of concurrency, vol. 23, no. 6. ACM, 1998, pp. 35–45.

[12] R. Bergenthum, J. Desel, R. Lorenz, and S. Mauser, "Process mining based on regions of languages," in Business Process Management, Springer, 2007, pp. 375–383.

[13] V. Rubin, C. W. Günther, W. M. P. Van Der Aalst, E. Kindler, B. F. Van Dongen, and W. Schäfer, "Process mining framework for software processes," in Software Process Dynamics and Agility, Springer, 2007, pp. 169–181.

[14] H. R. Motahari-Nezhad, R. Saint-Paul, B. Benatallah, and F. Casati, "Protocol Discovery from Imperfect Service Interaction Logs," in 2007 IEEE 23rd International Conference on Data Engineering, 2007, pp. 1405–1409.

[15] R. E. Shannon, "Systems simulation," 1975.

[16] R. Ardhaldjian and M. Fahner, "Using simulation in the business process reengineering effort-since simulation models are tools used to describe systems and their dynamic behavior in iterative ways, they should be more widely," Ind. Eng., vol. 26, no. 7, pp. 60–61, 1994.

[17] M. Netjes, W. M. P. van der Aalst, and H. A. Reijers, "Analysis of resource-constrained processes with Colored Petri Nets," in Sixth Workshop and Tutorial on Practical Use of Coloured Petri Nets and the CPN Tools, 2005, vol. 576, pp. 251–266.

[18] A. Rozinat, R. S. Mans, M. Song, and W. M. P. van der Aalst, "Discovering colored Petri nets from event logs," Int. J. Softw. Tools Technol. Transf., vol. 10, no. 1, pp. 57–74, 2008.

[19] F. Gottschalk, W. M. P. van der Aalst, M. H. Jansen-Vullers, and H. M. W. Verbeek, "Protos2CPN: using colored Petri nets for configuring and testing business processes," Int. J. Softw. Tools Technol. Transf., vol. 10, no. 1, pp. 95–110, Sep. 2007.

[20] K. Jensen and L. M. Kristensen, Coloured Petri nets: modelling and validation of concurrent systems. Springer, 2009.

[21] W. Van Der Aalst and K. M. Van Hee, Workflow management: models, methods, and systems. MIT press, 2004.

[22] W. M. P. van der Aalst, H. A. Reijers, A. J. M. M. Weijters, B. F. van Dongen, A. K. Alves de Medeiros, M. Song, and H. M. W. Verbeek, "Business process mining: An industrial application," Inf. Syst., vol. 32, no. 5, pp. 713–732, 2007.

[23] A. Kumar, W. M. P. Van Der Aalst, and E. M. W. Verbeek, "Dynamic work distribution in workflow management systems: How to balance quality and performance," J. Manag. Inf. Syst., vol. 18, no. 3, pp. 157–194, 2002.

[24] H. Reijers and S. Limanmansar, "Best practices in business process redesign: an overview and qualitative evaluation of successful redesign heuristics," Omega, vol. 33, no. 4, pp. 283–306, Aug. 2005.

[25] W. M. P. van der Aalst, B. F. van Dongen, J. Herbst, L. Maruster, G. Schimm, and A. J. M. M. Weijters, "Workflow mining: A survey of issues and approaches," Data Knowl. Eng., vol. 47, no. 2, pp. 237–267, 2003.

[26] D. Grigori, F. Casati, M. Castellanos, U. Dayal, M. Sayal, and M.-C. Shan, "Business process intelligence," Comput. Ind., vol. 53, no. 3, pp. 321–343, 2004.

[27] A. K. A. De Medeiros and C. W. Günther, "Process mining: Using CPN tools to create test logs for mining algorithms," in Proceedings of the sixth workshop on the practical use of coloured Petri nets and CPN tools (CPN 2005), 2005, vol. 576.

[28] C. W. Günther and W. M. P. van der Aalst, "A generic import framework for process event logs," in Business Process Management Workshops, 2006, pp. 81–92.