Computers & Industrial Engineering 101 (2016) 372–390

Contents lists available at ScienceDirect

Computers & Industrial Engineering

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

TIMSPAT – Reachability graph search-based optimization tool for colored Petri net-based scheduling

Olatunde T. Baruwa [a,*], Miquel A. Piera [a], Antoni Guasch [b]

[a] Logistics and Aeronautics Unit, Department of Telecommunications and Systems Engineering, Universitat Autònoma de Barcelona, 08202, Campus Sabadell, Barcelona, Spain
[b] Departament d'Enginyeria de Sistemes i Automàtica, FIB (Campus Nord), Universitat Politècnica de Catalunya, 08034 Barcelona, Spain

ARTICLE INFO

Article history:
Received 2 July 2015
Received in revised form 14 May 2016
Accepted 29 July 2016
Available online 2 August 2016

Keywords:
Colored Petri net
Scheduling
Reachability graph
Heuristic search
Discrete event system
Flexible manufacturing system
Simulation

ABSTRACT

The combination of Petri net (PN) modeling with AI-based heuristic search (HS) algorithms (PNHS) has been successfully applied as an integrated approach to deal with scheduling problems that can be transformed into a search problem in the reachability graph. While several efficient HS algorithms have been proposed albeit using timed PN, the practical application of these algorithms requires an appropriate tool to facilitate its development and analysis. However, there is a lack of tool support for the optimization of timed colored PN (TCPN) models based on the PNHS approach for schedule generation. Because of its complex data structure, TCPN-based scheduling has often been limited to simulation-based performance analysis only. Also, it is quite difficult to evaluate the strength and tractability of algorithms for different scheduling scenarios due to the different computing platforms, programming languages and data structures employed. In this light, this paper presents a new tool called TIMSPAT, developed to overcome the shortcomings of existing tools. Some features that distinguish this tool are the collection of several HS algorithms, XML-based model integration, the event-driven exploration of the timed state space including its condensed variant, localized enabling of transitions, the introduction of static place, and the easy-to-use syntax statements. The tool is easily extensible and can be integrated as a component into existing PN simulators and software environments. A comparative study is performed on a real-world eyeglass production system to demonstrate the application of the tool for scheduling purposes.

1. Introduction

In the manufacturing industry, companies are consistently faced with the challenge on how to maximize the utilization of limited resources to perform a collection of tasks (jobs) while optimizing a certain performance measure in order to meet customer due dates given the changing customers demands and tight production requirements. Efficient planning and scheduling policies are critical to the survival of companies in today's globally competitive market. Due to the complexity and level of detail required by real-world systems, optimization of scheduling problems are known to be NP-hard since the computation time to obtain an optimal schedule grows exponentially with the problem size (Xie & Allen, 2015).

While most solution approaches are largely dominated by optimization models based on mathematical programming, there has been a consistent rise in the use of formal modeling techniques such as timed Petri nets (TPNs) (Baruwa & Piera, 2016; Tuncel & Bayhan, 2007) and timed Automata (TA) (Harjunkoski et al., 2014; Nishi & Wakatake, 2014) for planning and scheduling. Besides their capability to validate and verify the behavior of systems, the simulation capabilities of these formal methods make it more flexible in combining them with solution approaches from Operations Research, Artificial Intelligence (AI), and the Computer Science domains. Specifically, Petri nets (PNs) are a powerful graphical and mathematical modeling tool, which have been extensively used to model, simulate, and analyze discrete-event systems characterized by concurrency, parallelism, causal dependency, resource sharing and synchronization (Basak & Albayrak, 2015; Murata, 1989). Since its inception in Carl Adam Petri's PhD dissertation on "Communication with Automata" in 1962, it has gained recognition in the research community in addressing manufacturing and logistic systems including its application in a number of different disciplines like communications, transportation, robotics, engineering, business and air traffic management (Zurawski & Zhou, 1994).

* Corresponding author.
E-mail addresses: olatundetemitope.baruwa@e-campus.uab.cat, tbaruwa@gmail.com (O.T. Baruwa), miquelangel.piera@uab.es (M.A. Piera), toni.guasch@upc.edu (A. Guasch).

http://dx.doi.org/10.1016/j.cie.2016.07.031

This paper presents a TIMed State space Performance Analysis Tool (TIMSPAT) for the modeling and analysis of scheduling problems described by the timed colored PN (TCPN) formalism based on the integrated PN and heuristic search (PNHS) scheduling methodology. TIMSPAT is a tool developed as part of the PhD thesis in (Baruwa, 2015). To the best of our knowledge, this is the first attempt at providing a scheduling tool for the optimization of TCPN models with reachability graph search-based HS methods. Thanks to the common data structure of AI HS methods, the tool is capable of incorporating several search algorithms in a single executable. So far, nine algorithms have been implemented, ranging from the classical A* (Russell & Norvig, 2009), hybrid heuristic search (Huang, Sun, Sun, & Zhao, 2010; Mejia & Odrey, 2005; Moro, Yu, & Kelleher, 2000; Reyes, Yu, Kelleher, & Lloyd, 2002) to anytime algorithms (Baruwa & Piera, 2014; Baruwa, Piera, & Guasch, 2015; Malone & Yuan, 2014; Zhang, 1999; Zhang & Korf, 1995).

In the PNHS approach, the main idea is to formulate the optimization of a scheduling problem as an AI automated planning problem (Bensalem, Havelund, & Orlandini, 2014; Ghosh, Dasgupta, & Ramesh, 2015) that takes as input the T(C) PN model, the initial state, and a desired goal state, and then produces a sequence of actions to achieve the goal (complete schedule) using heuristic search. Here, the scheduling problem is transformed into a search-based optimization problem in the reachability graph (RG or state space) of finding the optimal or near-optimal sequence of transition firings from some initial state to the goal state, which minimizes some performance measure. RG analysis is a reliable and efficient method to obtain optimal schedules since it can be used as an automated decision support tool to generate all the possible alternatives of the system configuration. A basic intuition underlying the use of RG is that all the reachable states in the T (C) PN are represented as nodes, and the transformation of these states that triggers a change in the system state, as edges. However, the state explosion problem has limited its practical applicability.

AI-based HS methods (Huang, Jiang, & Zhang, 2014; Lei, Xing, Han, Xiong, & Ge, 2014; Mejia & Odrey, 2005; Reyes et al., 2002; Xiong & Zhou, 1998; Yu, Reyes, Cang, & Lloyd, 2003) have been proposed to simulate only the best scenarios by generating partial RGs with heuristic functions (to guide the search) that rely on the knowledge of the production plans. From Tuncel and Bayhan (2007)'s review, the PNHS approach has proved to be an efficient method for solving production scheduling problems. However, majority of the works on PNHS have practically used TPN only. Although the focus is on manufacturing systems, the methodology can be applied to a broad class of scheduling problems since PN is not driven by a problem domain (Denaro & Pezzè, 2004). Recent studies have demonstrated the capability of the PNHS approach to deal with realistic industrial engineering applications such as the train rescheduling problem for a Dutch railway network (Wang, Ma, Goverde, & Wang, 2016) and resource-constrained project scheduling in the animation and videogame industry (Mejia et al., 2016).

Most model checking tools (Kordon et al., 2015, 2012) can be extended to solve the optimal reachability problem for scheduling, where the goal state is the specification of the property to be verified in the model checking problem (Cimatti, Edelkamp, Fox, Magazzeni, & Plaku, 2015; Li, Dong, Sun, Liu, & Sun, 2014). However, these tools have not been enabled with optimization capabilities because most verification algorithms are designed to perform an exhaustive exploration of the RG with untimed nets. Although HSF-SPIN (Edelkamp & Jabbar, 2006) is targeted at directed model checking with HS methods. On the other hand, PNHS scheduling requires only a partial state space exploration of timed nets to find an optimal or near-optimal solution guided by heuristic functions. The use of HS-RG algorithms appears to be a well-developed method (in tools) for other modeling formalisms like TA and Promela. State-of-the art tools like SPIN (Ruys, 2003), UPAAL-CORA (Behrmann et al., 2001; Behrmann, Larsen, & Rasmussen, 2005), and TAOpt (Panek, Stursberg, & Engell, 2006; Schoppmeyer, Subbiah, Valdes, & Engell, 2014) have been used for scheduling in job shops, process systems engineering and chemical production. However, no attempt has been made for TCPN-based scheduling.

The optimization of stochastic nets is not considered in this paper due to the continuous time domain and memoryless property of exponential time distributions (Zimmermann, 2012) that does not make them amenable to PNHS. An alternative approach that is well suited for stochastic nets is the optimization by means of simulation (Latorre-Biel et al., 2015) where tools like TIMENET (Bodenstein & Zimmermann, 2014; Zimmermann, 2012), GreatSPN (Amparore, 2014), PIPE2 (Dingle, Knottenbelt, & Suto, 2009), and CPN Tools (Jensen, Kristensen, & Wells, 2007) can be used. Notwithstanding, deterministic PNs have been proved to be successful for applications in uncertain and dynamic environments (Mejia et al., 2016; Wang et al., 2016) both as simulation and optimization models using the reactive scheduling strategy where rescheduling can be used to handle unexpected events or disruptions.

This paper is structured as follows. The remaining parts of Section 1 discusses the motivation and the state-of-the-art review on PN-based tools. Section 2 describes the TIMSPAT architecture and its main components in detail. Section 3 presents the case study of the flexible manufacturing cell and its corresponding TCPN model. Section 4 reports the computational and benchmarking results of the HS algorithms implemented considering several production scenarios, while Section 5 concludes the paper and presents the future plans in place to improve the tool's robustness.

1.1. Motivation

In a dynamic manufacturing environment, production managers are usually confronted with constantly changing scheduling scenarios in their day-to-day activities on the shop floor. Different scenarios may arise as a result of changes in the production mix, product types, due dates, part shortages and unanticipated events like machine failure. The availability of a number of solution algorithms can allow production managers make better decisions considered acceptable for each scenario. However, not all existing algorithms can be suited to all kinds of scheduling problems that arise on the shop floor. While it is possible to adapt an algorithm to different production scheduling scenarios, it may turn out to be inefficient for some. Putting different algorithms at the disposal of the production managers given the situations they are best suited for, may go a long way to aid their decision making. Therefore, it is important to provide the decision makers with a platform that supports a neutral representation in which the different solution algorithms could be automatically tested to select the best solution reached or the best algorithm suitable to solve the given problem at hand. However, there is a lack of decision support tools based on PNHS that can afford the aforementioned concept.

One of the advantages presented by the PNHS approach is that different search algorithms can be implemented to evaluate the best schedule of a particular manufacturing scenario. Several heuristic search methods have been developed for PNHS (Huang et al., 2014; Huang, Sun, & Sun, 2008; Huang et al., 2010; Lei et al., 2014; Mejia & Odrey, 2005; Reyes et al., 2002; Xiong & Zhou, 1998; Yu et al., 2003). However, it is quite difficult to evaluate and benchmark the efficiency of these algorithms in terms of time and solution quality due to the different computing platforms, programming languages and data structures used.