Detecting Temporal Anomaly and Interestingness in Timed Business Process Models

Deng Zhao[1], Zhangbing Zhou[1,3], Yasha Wang[2], Walid Gaaloul[3]
[1] School of Information Engineering, China University of Geosciences (Beijing), Beijing, China
[2] School of Electronics Engineering and Computer Science, Peking University, Beijing, China
[3] Computer Science Department, TELECOM SudParis, Evry, France
Email: dengzhao.cugb@gmail.com, zhangbing.zhou@gmail.com, wangys@sei.pku.edu.cn, walid.gaaloul@mines-telecom.fr

***Abstract*—This paper proposes to derive temporal constraints and granularities corresponding to individual activities, collaborative activities and their connecting edges from event logs. Specifically, a timed hierarchical business process model is constructed. Temporal anomalies are measured with time-constrained and granularity-aware bounds according to user's acceptance of deviant executions. Temporal interestingness, as the complement to anomaly detection, is evaluated as the most probable execution times that are partitioned into user-defined granules and ranked by probability. Experimental evaluations upon public event logs demonstrate the effectiveness and applicability of our proposed model for temporal anomaly and interestingness detection in terms of accuracy and recall, in comparison with the state-of-art's techniques.**

I. Introduction

Business processes are composed of a set of partially ordered activities, where these activities can be Web/REST services or mashup APIs, and they are managed by workflow management systems [1]. Generally, business analysts are naturally interested in detecting *anomalies* in these service-based business processes, because they can be indicators for potential incidents [2]. Automated detection of potential anomalies is essential and crucial in exposing and preventing fraud, misuse, attacks, and errors, and thus to ensure proper business processes executions. Another similar term like anomaly is *interestingness*, which manifests the most interesting and concerned information [3]. Current researches on anomaly and interestingness focus mainly on measurement design and discovery algorithms [4], [5], while few works have investigated from time perspective. In fact, the execution of business processes can have a temporal dimension, such that certain time constraints have to be complied with. Therefore, evaluating anomalies and interestingness with temporal constraints become a challenge.

Techniques have been proposed to investigate temporal constraints for business processes from various aspects, including temporal network representation of event logs [6], the modeling of discrete event systems with temporal constraints [7], the identification of temporal requirements for process modelling languages [8], the verification of business process model as timed automata [9], and the investigation of violation of temporal process constraints [10]. These techniques are mostly exploring certain aspects of temporal constraints involved in business processes, and offering limited support for temporal modeling and analysis. Besides, time stamps in event logs may have various levels of granularities ranging from seconds in precision to coarse granules, which are intrinsically embedded in temporal data but have hardly been considered so far. In this setting, a comprehensive modeling with temporal constraints and temporal granularities is to be explored.

Current researches on temporal anomaly and interestingness detection focus mainly on individual activities and connecting edges [5], [11], whereas collaborative activities have rarely been concerned. Actually, temporal granularites show an unique advantage in collaborating and integrating business activities. On the other hand, due to application complexity, abnormal (or interesting) executions of business activities may not be well defined only by unchanged measurement rules. For instance, users may have different acceptance of deviations of temporal anomalies with respect to different application requirements. While the interesting execution times, generally referred as coarse time intervals, can be partitioned into user-defined granules and those with higher probabilities may be more likely to satisfy users's demands. Therefore, evaluating anomalies and interestingness based on extensive modeling of temporal constraints and granularities is a challenging task.

To address these challenges, we propose to explore temporal constraints and granularities and develop a timed hierarchical business process model, to facilitate the detection of temporal anomalies and interestingness. We summarize the contributions of this paper as follows:

- We propose a timed hierarchical business process model (*THBPM*), which formalizes temporal constraints and granularities upon individual activities, collaborative activities and their connecting edges.
- We measure and evaluate temporal anomalies with time-constrained and granularity-aware bounds according to user's acceptance of deviant executions.
- We assess and identify temporal interestingness as the most probable execution times, which are partitioned into user-defined granules and ranked by probability.

Extensive experiments are conducted leveraging publicly available event logs, and comparison results demonstrate that our work outperforms the state-of-art's techniques.

978-1-7281-8786-0/20/$31.00 ©2020 IEEE
DOI 10.1109/ICWS49710.2020.00061

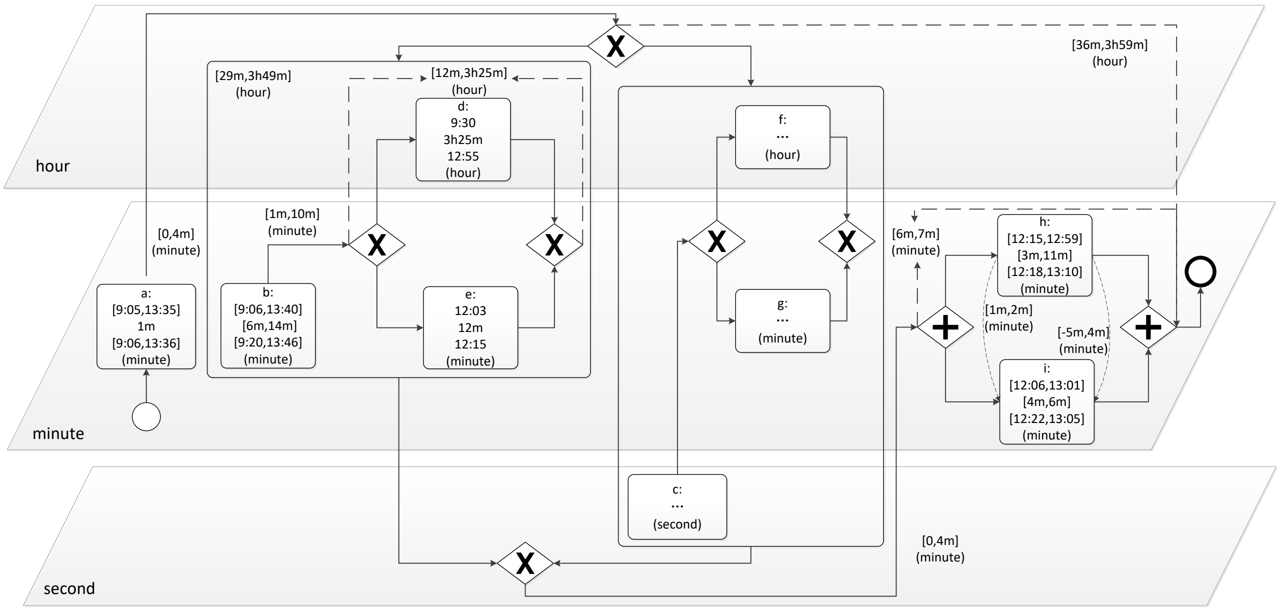

Figure 1. A sample timed hierarchical business process model of three layers, where temporal constraints and granularities are specified upon activities, gateways and their connecting edges.

II. Temporal Constraint and Granularity

Temporal constraints can be absolute and relative ones [12], and their definitions are given as follows:

Definition 1: (**Relative temporal constraints**) are used to specify requirements for activity duration, temporal relation and temporal dependency as follows:

- *Activity duration*: An activity atv is registered with its starting time and finishing time, and expressed with a time duration $T(atv)$.
- *Temporal relation*: Considering temporal relations between two activities (denoted atv_i, atv_j), atv_j can be executed *after, met by, equals, overlapped by, contains, started by,* or *finished by* atv_i [6].
- *Temporal dependency*: Four kinds of temporal dependencies, including *Start-to-Start* (*SS*), *Start-to-Finish* (*SF*), *Finish-to-Start* (*FS*) and *Finish-To-Finish* (*FF*), are specified according to temporal relations.

Definition 2: (**Absolute temporal constraints**) are specified to define starting times and finishing times of activities:

- *Starting time*: Start No Earlier Than (*SNET*), Start No Later Than (*SNLT*), or Must Start On (*MSO*).
- *Finishing time*: Finish No Earlier Than (*FNET*), Finish No Later Than (*FNLT*), or Must Finish On (*MFO*).

A granularity is a calendar-dependent partitioning function which maps time line clock chronons into certain granules [13]. In this paper, we adopt the widely-used Gregorian calendar, which is a collection of mapping functions (denoted *MappingFunc*), for describing various granularities. It supports granularities including *second*, *minute*, *hour*, *day*, *week*, *fortnight*, *month* and *year*, and is capable of covering the commonly appearing granularities.

III. Timed Hierarchical Business Process

Business processes can be mined from event logs leveraging the widely-adopted inductive miner [1], and expressed in terms of the Business Process Modeling Notation (*BPMN*) format. Activities, gateways and their connecting edges are formalized from the structural perspective, while no temporal information is specified in a business process model. To bridge this gap, we propose to bring temporal constraints into business processes and derive timed activities, gateways and their connecting edges, respectively [12], as follows:

1) Temporal constraints of activities are extracted, where (i) starting times (and finishing times) are constrained by *SNET*, *SNLT*, *MSO* (and *FNET*, *FNLT*, *MFO*), and (ii) activity durations are formalized as time durations.
2) Activities executed sequently, selectively or parallelly can be considered as the whole (i.e., in terms of a collaborative activity), and its starting time, finishing time and activity duration are specified accordingly.
3) Finally, we investigate temporal relations and dependencies by comparing their starting times and finishing times as introduced in section II.

Based on temporal constraints generated above, a timed hierarchical business process model is constructed through bringing temporal granularities to these timed activities, gateways and their connecting edges, and defined as follows:

Definition 3: (**Timed Hierarchical Business Process Model (THBPM)**) A timed hierarchical business process model is a tuple $thbpm$ = $(TA,\ TG,\ TE,\ TGR)$ where:

- *TA is a set of timed activities.*
- *TG is a set of timed gateways.*
- *TE is a set of timed edges where* $TE \subseteq (TA \times TA) \cup (TA \times TG) \cup (TG \times TG)$.
- *TGR is a set of temporal granularities assigned to timed activities, gateways and edges* $TA \cup TG \cup TE$.

Generally, a timed hierarchical business process model is discovered by composing activities from fine granularities to coarse ones based on their control dependencies, as presented by Algorithm 1. Firstly, each activity, gateway and edge is accompanied with its most probable temporal granularity (lines 1-7). Activities executed sequently, selectively or parallelly with the finest granularity should be composed as a collaborative activity. This composition procedure specified by lines 8-17 iterates until no activity is available for composition. Thereafter, a layer hierarchy of our proposed model, with respect to the finest granularity, is constructed (lines 18-19). This composition process repeats until all activities are composed as a whole, and a timed hierarchical business process model is established. An sample timed hierarchical business process model is given in Figure 1.

IV. Temporal Anomaly and Interestingness Detection

A. Temporal Anomaly Detection

Instead of predefining fixed acceptable bounds for anomalous behaviour detection, we measure and detect temporal anomalies through specifying time-constrained and granularity-aware bounds in a more customized manner on the basis of our proposed timed hierarchical business process model. Formally, a granularity-based temporal anomaly is defined as follows:

Definition 4: (**Granularity-based Temporal Anomaly**) A granularity-based temporal anomaly is determined based on a tuple ta = $(gra,\ lb,\ ub,\ tagra,\ pn)$ where

- *gra is a specified granularity as presented in Sect. III.*
- *lb is an observed lower bound.*
- *ub is an observed upper bound.*
- *tagra (≤ gra) is a predefined granularity for specifying customized lower and upper bounds.*
- *pn is a partitioning number corresponding to tagra.*

To be specific, more customized lower and upper bounds, which vary with respect to user's acceptance of deviations of anomalies, are given by

$$lb(ta) = Math.Floor(lb,\ i \times tagra/pn) \tag{1}$$

$$ub(ta) = Math.Ceil(ub,\ j \times tagra/pn) \tag{2}$$

Algorithm 1 Timed Hierarchical BP Discovery

Require:
- $elog$: an event log.
- bpm: the discovered business process model.

Ensure:
- $thbpm$: a timed hierarchical business process model.

1: $TA \cup TG \cup TE \leftarrow TimedBPDiscovery(elog,\ bpm)$
2: **for** each item $itm_i \in TA \cup TG \cup TE$ **do**
3:   $itm_i.gra \leftarrow GraProb(itm_i)$
4:   **if** $itm_i \in TA$ **then**
5:     $mg \leftarrow Math.Min(mg,\ itm_i.gra)$
6:   **end if**
7: **end for**
8: **while** $\exists$ a pair atv_i $(\rightarrow$ / $\times$ / $+)$ $atv_j \in TA$ **do**
9:   $layer_{mg} \leftarrow null$
10:   **for** each pair atv_i $(\rightarrow$ / $\times$ / $+)$ $atv_j \in TA \wedge$ $(atv_i.gra = mg \vee atv_j.gra = mg)$ **do**
11:     $atv_n \leftarrow atv_i$ $(\rightarrow$ / $\times$ / $+)$ atv_j
12:     $TA \leftarrow TA \cup atv_n$
13:     $TA \leftarrow TA - \{atv_i,\ atv_j\}$
14:     $layer_{mg} \leftarrow layer_{mg} \cup atv_i\ (\wedge\ atv_i.gra = mg)$
15:     $layer_{mg} \leftarrow layer_{mg} \cup atv_j\ (\wedge\ atv_j.gra = mg)$
16:     $layer_{mg} \leftarrow layer_{mg} \cup te(\in TE)$ between atv_i and atv_j
17:   **end for**
18:   $thbpm \leftarrow thbpm \cup layer_{mg}$
19:   $mg \leftarrow MappingFunc(mg,\ coarser)$
20: **end while**

where $Math.Floor(lb, i \times tagra/pn)$ returns the maximum value of $i \times tagra/pn$ that is no bigger than lb, while $Math.Ceil(ub, j \times tagra/pn)$ returns the minimum value of $j \times tagra/pn$ no smaller than ub ($i, j \in \{0, pn\}$). Therefore, anomalies are identified when temporal data are not within customized allowed bounds. For instance, given a tuple $ta = (hour, 12m, 3h25m, hour, 24)$ corresponding to $\{d, e\}$ in Figure 1, $\{d, e\}$ is not identified as a temporal anomaly if it's within the range $[0, 4h]$ calculated by above equations, e.g., $3h40m$. In this paper, $tagra$ is usually smaller than gra, and partitioning numbers for granularity *second*, *minute*, *hour* and *day* are 6, 6, 24 and 7. We calculate accuracy and recall of temporal anomaly detection as follows:

$$Accuracy(ta) = \frac{|TP_{ta} + TN_{ta}|}{|TP_{ta} + TN_{ta} + FN_{ta}|} \tag{3}$$

$$Recall(ta) = \frac{|TP_{ta}|}{|TP_{ta} + FN_{ta}|} \tag{4}$$

where TN_{ta} denotes temporal samples within $[lb, ub]$, TP_{ta} denotes that not within $[lb(ta), ub(ta)]$, while FN_{ta} denotes remaining samples within $(lb(ta), lb)$ or $(ub, ub(ta))$.

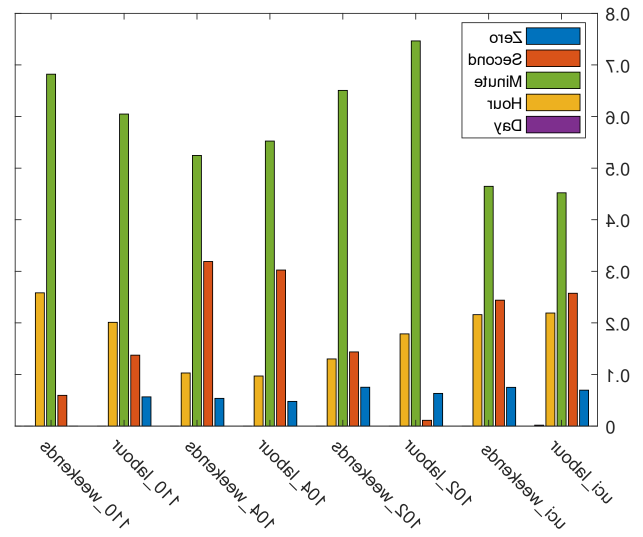

Figure 2. Average percentages of granularities of activities' durations which ranging from *seconds* to *days*.

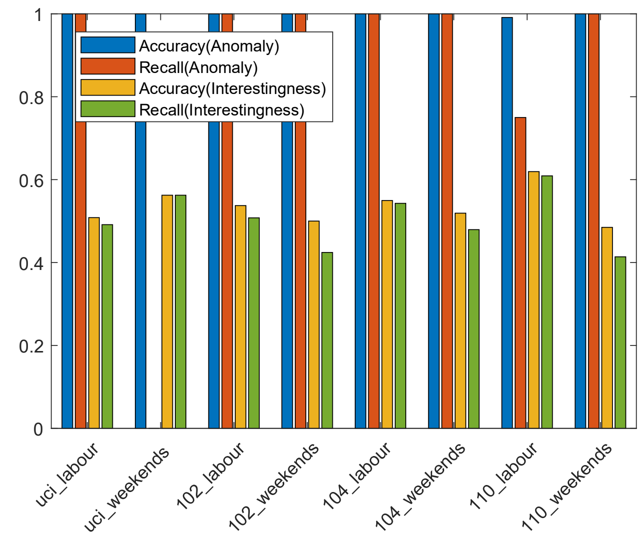

Figure 3. Average accuracy and recall of temporal anomalies and interestingness of activities' starting times.

B. Temporal Interestingness Detection

Temporal interestingness, as the complement to temporal anomaly detection, is evaluated as interesting time intervals specified by temporal granularities, and is formally defined with awareness of granularities as follows:

Definition 5: (**Granularity-based Temporal Interestingness**) Granularity-based temporal interestingness is determined based on a tuple ti = (gra, pn, $thrd$) where

- *gra is a specified granularity as in Section III.*
- *pn is a partitioning number corresponding to gra.*
- *thrd is a probability threshold for indicating temporal interestingness.*

Specifically, temporal interestingness is stated as time intervals (denoted $lub(ti)$) with a total probability bigger than a pre-specified threshold $thrd$ as follows:

$$lub(ti) = \sum_{k=0}^{pn-1} [unit \times k, unit \times (k+1)) \tag{5}$$

where

$$\sum_{k=0}^{pn-1} prob[unit \times k, unit \times (k+1)) \geq thrd \tag{6}$$

Note that $prob[unit \times k, unit \times (k+1))$ returns the probability of the input interval, where $unit$ = gra/pn, ($k \in \{0,\ pn-1\}$). Given a sample tuple ti = ($hour$, 12, 70%) corresponding to $\{d, e\}$ as shown in Figure 1, it indicates that a duration with granularity *hour* (i.e., 24 hours) is partitioned into 12 time units. Temporal interestingness is identified when the total probability of time interval(s) $[unit \times k, unit \times (k+1))$ (e.g., $[2h, 4h)$, k = 1) is higher than 70%. In this paper, we set pn as the same values for anomaly detection, and $thrd$ is set to 70%. We calculate accuracy and recall of temporal interestingness as follows:

$$Accuracy(ti) = \frac{|TP_{ti} + TN_{ti}|}{|TP_{ti} + TN_{ti} + FN_{ti}|} \tag{7}$$

$$Recall(ti) = \frac{|TP_{ti}|}{|TP_{ti} + FN_{ti}|} \tag{8}$$

where the symbol TP_{ti} refers to the numer of temporal samples within $lub(ti)$, while the symbol FN_{ti} specifies the number of temporal samples not within $lub(ti)$, but within $[lb(ta), ub(ta)]$. The symbol TN_{ti} returns the number of temporal samples not within $[lb(ta), ub(ta)]$.

V. Implementation and Evaluation

A prototype has been implemented in JAVA, and experiments have been conducted based on publicly-available real-life event logs at the *4TU Centre for Research Data* [1]. The logs size ranging from 6 traces (368 events) to 43 traces (4200 events), while we take 80% of event logs for model discovery, and remaining 20% for performance evaluation.

Experimental evaluations are conducted based on different event logs as shown in Figures 2 - 3 to investigate the applicability of our proposed model upon various data sets. We can observe that the majority of activities in these event logs are usually executed for minutes as shown in Figure 2, which indicate that different temporal constraints can be derived from event logs and expressed with various temporal granularities. While temporal anomaly and interestingness detection are implemented as shown in Figure 3, which demonstrate their complementary and effectiveness.

[1] http://data.4tu.nl/repository/collection:event_logs

Table I
A COMPARATIVE RESULT REGARDING TO TEMPORAL CONSTRAINTS AND GRANULARITIES IN RELATED TECHNIQUES AS INTRODUCED IN SECTION **??**.

Approaches [Ref.]	Intra-activity		Inter-activity			Collaborative Activities	Temporal Granularity
	Duration	Starting/Finishing Time	Interval	Temporal Relation	Temporal Dependency		
TNR [6]	✓		✓	✓			
PNDES [7]	✓		✓		✓		
TCMBP [8]	✓	✓	✓		✓	✓	
TAN [9]	✓						
TPC [10]	✓				✓		
EBPMN [14]	✓						
TCBPM [15]	✓	✓	✓		✓		
STP [16]	✓	✓	✓		✓		
Our *THBPM*	✓	✓	✓	✓	✓	✓	✓

In conclusion, these results demonstrate the applicability of temporal granularities mined from the event logs, and the effectiveness of temporal anomaly and interestingness detection in terms of accuracy and recall.

A comparison of recent research approaches, which are developed for providing formalized temporal constraints and ensuring business processes adhering to these derived temporal constraints, with respect to our proposed *THBPM* in terms of supported temporal constraints and granularities are presented at Table I. From the comparison results, we may safely draw the conclusion that our technique is more comprehensive and effective in specifying and formalizing temporal constraints and granularities.

VI. CONCLUSION

In this paper we develop a timed hierarchical business process model with temporal constraints and granularities in order to support temporal modeling and analysis of service-based business processes. Based on which temporal anomalies and interestingness are estimated and identified. Temporal anomaly and interestingness detection are proved to be applicable in terms of accuracy and recall, and experimental results based on real-life event logs show the effectiveness and applicability of our proposed model in comparison with the state-of-art's techniques.

ACKNOWLEDGMENT

This work was supported by the National Key Research and Development Program of China.

REFERENCES

[1] W. M. van der Aalst, "Process discovery from event data: Relating models and logs through abstractions," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 8, no. 3, p. e1244, 2018.

[2] S. Pauwels and T. Calders, "Detecting anomalies in hybrid business process logs," *ACM SIGAPP Applied Computing Review*, vol. 19, pp. 18–30, 2019.

[3] A. Masood and S. Soong, "Measuring interestingness-perspectives on anomaly detection," *Thrombosis Research*, vol. 117, no. 3, pp. 307–14, 2013.

[4] N. Schwenzfeier and V. Gruhn, "Toward a practical process model for anomaly detection systems," in *IEEE/ACM 1st International Workshop on Software Engineering for Cognitive Services*, 2018, pp. 41–44.

[5] A. Cecconi, C. Di Ciccio, and J. De Giacomo, Giuseppe nd Mendling, "Interestingness of traces in declarative process mining: The janus ltlpf approach," in *Business Process Management*, 2018, pp. 121–138.

[6] A. Senderovich, M. Weidlich, and A. Gal, "Context-aware temporal network representation of event logs: Model and methods for process performance analysis," *Information Systems*, vol. 84, pp. 240 – 254, 2019.

[7] S. Rachidi, E. Leclercq, Y. Pign, and D. Lefebvre, "Pn modeling of discrete event systems with temporal constraints," in *International Conference on System Theory, Control and Computing*, 2017, pp. 70–75.

[8] J. L. Pereira and J. Varajao, "The temporal dimension of business processes: Requirements and challenges," *International Journal of Computer Applications in Technology*, vol. 59, no. 1, pp. 74 – 81, 2019.

[9] L. E. M. Morales, C. Monsalve, and M. Villavicencio, "Formal verification of business processes as timed automata," in *Iberian Conference on Information Systems and Technologies*, 2017, pp. 1–6.

[10] A. Kumar and R. R. Barton, "Controlled violation of temporal process constraints - models, algorithms and results," *Information Systems*, vol. 64, pp. 410–424, 2017.

[11] T. Nolle, S. Luettgen, A. Seeliger, and M. Mehlhauser, "Binet: Multi-perspective business process anomaly classification," *Information Systems*, p. 101458, 2019.

[12] Z. Zhou, D. Zhao, and L. Liu, "Formal verification of temporal constraints for mobile service-based business process models," *IEEE Access*, vol. 6, pp. 59 843–59 852, 2018.

[13] C. E. Dyreson, W. S. Evans, H. Lin, and R. T. Snodgrass, "Efficiently supporting temporal granularities," *IEEE Transactions on Knowledge and Data Engineering*, vol. 12, no. 4, pp. 568–587, 2000.

[14] K. Watahiki, F. Ishikawa, and K. Hiraishi, "Formal verification of business processes with temporal and resource constraints," in *IEEE International Conference on Systems, Man, and Cybernetics*, 2011, pp. 1173–1180.

[15] S. Cheikhrouhou, S. Kallel, N. Guermouche, and M. Jmaiel, "Enhancing formal specification and verification of temporal constraints in business processes," in *IEEE International Conference on Services Computing*, 2014, pp. 701–708.

[16] A. Lanz, M. Reichert, and B. Weber, "Process time patterns: A formal foundation," *Information Systems*, vol. 57, pp. 38–68, 2016.