Data-Driven Process Simulation

Peer-reviewed author version

Data-driven Process Simulation

Benoît Depaire, Niels Martin

This is a post-peer-review, pre-copyedit version of an article published in the Encyclopedia of Big Data Technologies. The final authenticated version is available online at: `https://doi.org/10.1007/978-3-319-63962-8_102-1`

Synonyms

-

Definition

Data-driven process simulation is a technique which constructs a computer model that imitates the internal details of a business process and extensively uses data - recorded by information systems supporting the actual process - to do so. The model is used to execute what-if scenarios in order to better understand the actual process behavior and predict the impact of potential changes to the process.

Benoît Depaire

Hasselt University, Research group Business Informatics, Agoralaan Building D, 3590 Diepenbeek, Belgium, e-mail: `benoit.depaire@uhasselt.be`

Niels Martin

Hasselt University, Research group Business Informatics, Agoralaan Building D, 3590 Diepenbeek, Belgium, e-mail: `niels.martin@uhasselt.be`

Overview

Data-driven process simulation

Every organisation executes multiple business processes - e.g. the production, transportation and billing process - which have to be managed properly to generate customer value (Dumas et al 2013). An essential part of business process management is the identification and design of process improvement opportunities - e.g. hire more staff to reduce waiting time at a specific step in the process. Since a business process typically has a complex and dynamic nature, it is often impossible to deduce analytically the full impact of a specific process change. In such setting, process simulation is a valuable instrument.

Process simulation is the imitation of the actual process through the use of a (computer) model. It allows an organisation to verify the consequences of proposed process modifications prior to implementation (Melão and Pidd 2003). The quality of the results of process simulation are directly dependent on the quality of the process simulation model.

Therefore, to construct a high-quality simulation model, the researcher must gather extensive process information as input. Traditionally, such input is gathered by means of business documents, interviews and observations. While valuable, it has been shown that these sources also have limitations:

- Process documentation often describe how a process should be executed which can deviate from the actual process behavior (Mǎruşter and van Beest 2009).
- Interviews with business experts can result in contradictory information (Vincent 1998) and the perception of the interviewees might be heavily influenced by recent experiences within the process (Dickey and Pearson 2005).
- Observational data could suffer from the Hawthorne effect (Robinson 2004) - i.e. the performance of staff members increase simply because they know they are being observed - and often require a significant time investment.

Another type of input are the data and records kept by different information systems about the daily process execution. With the rise of process-aware information systems - supporting the entire process rather than a specific activity (Dumas et al 2005), e.g. Enterprise Resource Planning systems - the amount of these process data are growing rapidly. The benefit of these data is that they provide an objective view on the real process behaviour. The challenge however, is to extract insights from these data which can be used to construct more realistic simulation models.

The goal of data-driven process simulation is to extend the traditional inputs for simulation model construction with such process data and to extract insights from the data with respect to different aspects of the simulation model. As a consequence, it encompasses several techniques which support specific modeling decisions.

Building blocks of a simulation model

When constructing a simulation model, four building blocks can be distinguished: entities, activities, control-flow and resources (Martin et al 2016b). For each of these building blocks, a series of modeling tasks need to be executed during the construction of a simulation model.

An **entity** is the dynamic object that flows through the process model and which instigates action in the process as activities are executed on it by resources (Kelton et al 2015; Tumay 1996) - e.g. a specific order (entity) in an order picking process. During simulation, multiple entities can be simultaneously present within the process and can influence each other's process flow by, e.g., occupying particular resources.

An **activity** is the execution of a specific task by a specific resource on a given entity (Tumay 1996) - e.g. the registration (task) by employee X (resource) of order Y (entity).

The **control-flow** of a process expresses the temporal relationship between the activities. Control-flow consists of sequence flows - modeling 'directly follows' relationships - and gateways - modeling complex process behavior such as choice and parallelism. The control-flow defines the different routes an entity can follow through the process - e.g. an international order might follow a different route in the order picking process than a national order, resulting in different or additional activities that need to be executed on the order.

The actual execution of activities is the responsibility of **resources**. Basic resource types are technical resources, such as software systems, and human resources (Dumas et al 2013; Tumay 1996).

Key Research Findings

This section provides the key research results on how process data can be used to improve different modeling aspects of process simulation. It should be noted that, in general, the insights that can be retrieved from process data depend upon the quality and granularity of the available data. To structure the discussion in this section, the four building blocks of a simulation model and their associated modeling decisions are used.

Entities

With respect to the *entity* building block, two modeling decisions will be discussed: the arrival rate and the entity typology.

Entity arrival rate

Modeling the entity arrival rate - i.e. the pace at which new entities arrive at the process - typically consists of a deterministic and a stochastic part. The deterministic part can be a constant or dependent on other context-related factors. The stochastic element is typically modeled as a probability distribution (van der Aalst 2015).

Process data is used to learn the parameters of a predefined probability distribution. The exponential distribution is often used under the assumption that the start of the first activity in a process execution defines the actual arrival of the entity and the entities arrive independent from each other (Rozinat et al 2009). When these assumptions appear too restrictive - e.g. an entity arrived before the first activity was executed on it - the ARPRA algorithm (Martin et al 2016c) accounts for queue formation before the first activity and discovers the proper Gamma distribution - which is a generalization of the exponential distribution.

Entity types

No two entities are exactly the same and it is not uncommon that entity properties have a direct impact on the process execution - e.g. the physical dimensions of an order have a direct impact on the duration of the packaging activity. To simplify complex processes, it is common to define entity types which represent a group of entities similar in terms of specific characteristics (i.e. attributes) (Kelton et al 2015).

When working with entity types, the quality of the defined typology has a direct influence on the quality of the simulation model. To identify relevant entity types, cluster analysis is a useful technique to segment the entities (Xu and Wunsch 2005). Recently, various algorithms have been developed that discover clusters which group entities that have a common execution path and where clusters - entity types - differ in terms of process execution. Various approaches exist: a vector-based approach where the trajectory of an entity is translated into a set of features (Bose and van der Aalst 2010; de Medeiros et al 2008; Delias et al 2015; Greco et al 2006; Song et al 2009), an approach based on the generic edit distance (Bose and van der Aalst 2009), Markov model based clustering (Veiga and Ferreira 2010) and an approach driven by process model accuracy (De Weerdt et al 2013).

Activities

With respect to activities, five important modeling decisions will be discussed: determining which activities need to be included, activity duration, resource requirements, queue discipline, and activity interruptions.

Activity definition

Process data is typically captured at the level of events - i.e. specific moments in time which change the state of the process. Even when data is logged at the level of activities, this data might be at a too granular level for process simulation purposes. In both cases, techniques are needed to identify appropriate activities and to map data to higher-level activities. For this purpose, activity mining (Günther et al 2010) can be used and only requires the event log. Other approaches which incorporate domain knowledge are also available such as in Szimanski et al (2013), Baier et al (2014) and Mannhardt et al (2016).

Activity duration

Modeling the activity duration typically consists of a deterministic and stochastic element. Deterministic differences in activity duration can be dependent on a wide range of entity properties, but also on resource workload (cfr. Pospisil and Hruška (2012) and Nakatumba and van der Aalst (2010); Nakatumba et al (2012)). Process data can be used to discover these dependencies and model them correctly.

The stochastic element is modeled by means of a probability distribution. Activity durations are extracted from the process data - i.e. difference between the start and completion time of an activity - and used to fit the probability distribution.

When either start or completion time of activities are missing, this approach cannot be applied directly. Literature suggest several approaches to deal with such missing data. When only the completion time is recorded, various proxies are suggested for the corresponding start time, such as the completion time of the previous activity for that entity or the time at which the resource executing the activity finished its previous job (Nakatumba 2013; Wombacher and Iacob 2013).When only the start of an activity is recorded, similar proxies have been suggested in Martin et al (2016b). Be aware that several issues such as the interruptibility of activities or the presence of batch processing can render the extraction of duration observations more complex.

Resource requirements

The resource(s) required to execute a particular activity need(s) to be specified. Process data containing resource information can be used to support this modeling task and different techniques exist to reveal the relationship between activities and resources (Huang et al 2011; Liu et al 2008; Ly et al 2006; Senderovich et al 2014).

Queue discipline

In a simulation model, one must define how queues are dealt with at activities. The queue discipline establishes the processing order of queuing entities, such as first-in-first-out (FIFO), last-in-first-out (LIFO) or according to a priority rule.

Process data can be used to discover the queuing discipline that is applied. Suriadi et al (2017) propose a method to extract FIFO, LIFO and priority-based processing patterns from process-related data and considers three perspectives: the resource, activity and case perspective, each representing potential levels of aggregation.

Activity interruptibility and unexpected interruptions

With respect to interruptions, the activity interruptibility and unexpected interruptions need to be modeled. The activity's interruptibility expresses whether it can be interrupted while an entity is being processed. If activities are interruptible, one must still model the interruption frequency, duration and nature. The nature of an interruption refers to whether the activity execution stops immediately (preemptive) or whether the running entity can still be finished (non-preemptive) (Hopp and Spearman 2011).

Process data containing specific information such as interruption events or resource break information (Senderovich et al 2014) can be used directly to inform the aforementioned modeling tasks. However, often such specific information is missing and the interruption time is simply part of the activity duration. Under these circumstances, duration outlier analysis can be used (Pika et al 2013; Rogge-Solti and Kasneci 2014) to inform the researcher on activity interruptions.

Control-flow

With respect to the control-flow, two aspects need to be modeled: the control-flow definition between the activities and the gateway routing logic.

Control-flow definition

The control-flow determines the path(s) an entity can follow throughout the process and defines sequentiality, choice and concurrency between activities. A multitude of algorithms exist to retrieve control-flow models from process data (De Weerdt et al 2012; van der Aalst 2016). Several of these algorithms have been applied in a simulation context such as the alpha-algorithm (Aguirre et al 2013; Rozinat et al 2008, 2009), fuzzy mining (van Beest and Măruşter 2007) and heuristic mining (Leyer and Moormann 2015; Măruşter and van Beest 2009).

Gateway routing logic

For gateways representing choice the routing logic needs to be defined, which can be stochastic - each outgoing path has a specific probability-, deterministic - business rules govern the choice- or a combination of both.

In literature, Rozinat and van der Aalst (2006a,b) use decision trees to discover the decision logic from process data, which is also applied in a simulation context (Rozinat et al 2008, 2009). Schonenberg et al (2010) use regression analysis to mine decision logic from process data. The work of de Leoni et al (2013) allows one to mine rules from process data which are linear combinations of multiple entity attributes. All these approaches result in deterministic rules, but can be easily extended with a stochastic element as suggested by Pospisil and Hruška (2012). A purely stochastic approach is presented in Leyer and Moormann (2015) which simply learns the routing probabilities for each choice.

Resources

With respect to modeling the resources, three modeling tasks are discussed: the definition of resource roles, resource schedules, and batch processing.

Resource roles

Resource roles group resources performing similar activities and allows one to simplify the simulation model - e.g. activities are assigned to roles instead of specific resources.

Various work on how to retrieve insights on resource roles from process data exist. One approach is to perform a cluster analysis on resource-activity frequency matrix retrieved from the process data (Song and van der Aalst 2008; Rozinat et al 2009). Another approach is to focus on handover relationships in the data (Burattin et al 2013) or to cluster resources on the number of entities they collaborated on (Ferreira and Alves 2012).

Resource schedules

A resource schedule expresses its availability for the simulated process, which can differ from the official working schedule as a resource is often involved in multiple processes. Process data can help to gather insights in the resource's availability for the process.

A first approach to use process data is to estimate the start and end time of a working day from the process data (Wombacher et al 2011). Other approaches estimate resource availability as a percentage of the total time a resource is executing activi-

ties (Liu et al 2012; van der Aalst et al 2010), which however do not provide exact availability timeframes. If more fine-grained insights are needed, the work of Martin et al (2016a) provides a method to mine daily availability records for resources.

Batch processing

Batch processing refers to a resource's tendency to accumulate entities in order to process them simultaneously, concurrently or sequentially. Again, process data can be used to study the actual batching behaviour.

Wen et al (2013) developed a control-flow discovery algorithm when simultaneous batch processing occurs in a process. Similar to Liu and Hu (2007), they assume that events are only recorded for one of the cases in a batch. While Wen et al (2013) focus on the control-flow, Nakatumba (2013) states that batching behaviour is present when two sets of activity instances on a particular working day are separated by a time period of more than one hour. A more systematic approach is proposed in Martin et al (2017). Based on a distinction between simultaneous, concurrent and sequential batch processing, an algorithm is proposed which detects batching behaviour. Moreover, metrics such as the batch size are calculated which can be used to specify batching parameters in a simulation model (Martin et al 2017).

Examples of Application

Literature provides examples in which real-life process data is extensively used to support the construction of simulation models. For example, Rozinat et al (2009) consider the complaint handling process and the invoice handling process in municipalities and use process data to specificy the entity arrival rate, activity duration, control-flow, gateway routing logic, and resource roles. In another example, Leyer and Moormann (2015) simulate the loan application process at a financial institution and use real-life process data to model the activity duration, control-flow, and gateway routing logic.

While the aforementioned references use process data to support multiple modeling decisions, dedicated papers focusing on a single decision also apply their proposed techniques to real-life data. For instance, Mărușter and van Beest (2009) mainly focus on the use of process data to model the simulation model's control-flow and consider three real-life processes: a process of a gas company, the fine collection process of a governmental institution and the navigation process in an agricultural decision support system. In another example, Martin et al (2017) retrieve insights in resources' batching behavior from a call center's and a production department's process data.

Future Directions for Research

Despite the valuable research efforts outlined above, four key directions of additional work to exploit process data for the development of a simulation model can be distinguished.

Firstly, there is room for new, improved and dedicated algorithms which support specific modeling decisions using process data. This article only discusses a subset of modeling tasks required to build a simulation model. For other modeling tasks, e.g. the additional ones included in Martin et al (2016b), no clear starting points for future research could be identified in literature. For some modeling decisions included in this article, the state of the art refers to approaches which were not directly developed with simulation in mind. For instance, the use of clustering to identify entity types results in distinct process models for each entity type, requiring an additional aggregation step to produce a single model. Dedicated algorithms can exploit the specific characteristics of process data and specific needs of process simulation. Additionally, existing (dedicated) algorithms can also be extended to provide even more insights. For example, the ARPRA algorithm (Martin et al 2016c) can be extended to discover non-stationary arrival patterns.

Secondly, a common assumption amongst process discovery techniques is that both start and completion of an activity are logged. Often the data does not meet these assumptions and e.g. only the completion times are recorded. In such cases, the researcher must make additional assumptions to apply existing algorithms. Therefore, future research on start time estimation and imputation is worth pursuing. Alternatively, existing algorithms can also be extended such that they do not longer require e.g. explicit start timestamps.

Thirdly, most existing algorithms focus on a single modeling aspect, whereas they are in fact related to other modeling decisions. Future algorithms for supporting modeling tasks could therefore benefit from an integrated approach. For example, the resource schedules can be taken into account when mining the queue discipline as a resource might start prioritizing urgent tasks at the end of the working day.

Finally, a single analysis package should be created which takes process data as an input and provides comprehensive support for a multitude of simulation modeling decisions. This involves an integration of the current algorithms with yet to be developed techniques for other modeling decisions. Even though the development of such an integrated package does not require the development of new scientific knowledge, it is required to facilitate data-driven simulation in practice.

References

van der Aalst WMP (2015) Business process simulation survival guide. In: vom Brocke J, Rosemann M (eds) Handbook on Business Process Management 1, Springer, Heidelberg, pp 337–370

van der Aalst WMP (2016) Process mining: data science in action. Springer, Heidelberg

van der Aalst WMP, Nakatumba J, Rozinat A, Russell N (2010) Business process simulation. In: vom Brocke J, Rosemann M (eds) Handbook on Business Process Management, Springer, Heidelberg, pp 313–338

Aguirre S, Parra C, Alvarado J (2013) Combination of process mining and simulation techniques for business process redesign: a methodological approach. Lecture Notes in Business Information Processing 162:24–43

Baier T, Mendling J, Weske M (2014) Bridging abstraction layers in process mining. Information Systems 46:123–139

van Beest NRTP, Măruşter L (2007) A process mining approach to redesign business processes - a case study in gas industry. In: Proceedings of the 2007 International Symposium on Symbolic and Numeric Algorithms for Scientific Computing, pp 541–548

Bose RPJC, van der Aalst WMP (2009) Context aware trace clustering: towards improving process mining results. In: Proceedings of the Ninth SIAM International Conference on Data Mining, pp 401–412

Bose RPJC, van der Aalst WMP (2010) Trace clustering based on conserved patterns: towards achieving better process models. Lecture Notes in Business Information Processing 43:170–181

Burattin A, Sperduti A, Veluscek M (2013) Business models enhancement through discovery of roles. In: Proceedings of the 2013 IEEE Symposium on Computational Intelligence and Data Mining, pp 103–110

De Weerdt J, De Backer M, Vanthienen J, Baesens B (2012) A multi-dimensional quality assessment of state-of-the-art process discovery algorithms using real-life event logs. Information Systems 37(7):654–676

De Weerdt J, Vanthienen J, Baesens B, vanden Broucke SKLM (2013) Active trace clustering for improved process discovery. IEEE Transactions on Knowledge and Data Engineering 25(12):2708–2720

Delias P, Doumpos M, Grigoroudis E, Manolitzas P, Matsatsinis N (2015) Supporting healthcare management decisions via robust clustering of event logs. Knowledge-Based Systems 84:203–213

Dickey D, Pearson C (2005) Recency effect in college student course evaluations. Practical Assessment, Research and Evaluation 10(6):1–10

Dumas M, van der Aalst WMP, Ter Hofstede AH (2005) Process-aware information systems: bridging people and software through process technology. John Wiley & Sons, Hoboken

Dumas M, La Rosa M, Mendling J, Reijers HA (2013) Fundamentals of business process management. Springer, Heidelberg

Ferreira DR, Alves C (2012) Discovering user communities in large event logs. Lecture Notes in Business Information Processing 99:123–134

Greco G, Guzzo A, Ponieri L, Sacca D (2006) Discovering expressive process models by clustering log traces. IEEE Transactions on Knowledge and Data Engineering 18(8):1010–1027

Günther CW, Rozinat A, van der Aalst WMP (2010) Activity mining by global trace segmentation. Lecture Notes in Business Information Processing 43:128–139

Hopp WJ, Spearman ML (2011) Factory physics. Waveland Press, Long Grove

Huang Z, Lu X, Duan H (2011) Mining association rules to support resource allocation in business process management. Expert Systems with Applications 38(8):9483–9490

Kelton W, Sadowski R, Zupick N (2015) Simulation with Arena. McGraw-Hill, New York

de Leoni M, Dumas M, García-Bañuelos L (2013) Discovering branching conditions from business process execution logs. Lecture Notes in Computer Science 7793:114–129

Leyer M, Moormann J (2015) Comparing concepts for shop floor control of information-processing services in a job shop setting: a case from the financial services sector. International Journal of Production Research 53(4):1168–1179

Liu J, Hu J (2007) Dynamic batch processing in workflows: model and implementation. Future Generation Computer Systems 23(3):338–347

Liu Y, Wang J, Yang Y, Sun J (2008) A semi-automatic approach for workflow staff assignment. Computers in Industry 59(5):463–476

Liu Y, Zhang H, Li C, Jiao RJ (2012) Workflow simulation for operational decision support using event graph through process mining. Decision Support Systems 52(3):685–697

Ly LT, Rinderle S, Dadam P, Reichert M (2006) Mining staff assignment rules from event-based data. Lecture Notes in Computer Science 3812:177–190

Mannhardt F, de Leoni M, Reijers HA, van der Aalst WMP, Toussaint J (2016) From low-level events to activities - a pattern-based approach. Lecture Notes in Computer Science 9850:125–141

Martin N, Bax F, Depaire B, Caris A (2016a) Retrieving resource availability insights from event logs. In: Proceedings of the 2016 IEEE International Conference on Enterprise Distributed Object Computing, pp 69–78

Martin N, Depaire B, Caris A (2016b) The use of process mining in business process simulation model construction: structuring the field. Business & Information Systems Engineering 58(1):73–87

Martin N, Depaire B, Caris A (2016c) Using event logs to model interarrival times in business process simulation. Lecture Notes in Business Information Processing 256:255–267

Martin N, Swennen M, Depaire B, Jans M, Caris A, Vanhoof K (2017) Retrieving batch organisation of work insights from event logs. Decision Support Systems 100:119–128

de Medeiros AKA, Guzzo A, Greco G, van der Aalst WMP, Saccà D (2008) Process mining based on clustering: a quest for precision. Lecture Notes in Computer Science 4928:17–29

Melão N, Pidd M (2003) Use of business process simulation: a survey of practitioners. Journal of the Operational Research Society 54(1):2–10

Mǎruşter L, van Beest NRTP (2009) Redesigning business processes: a methodology based on simulation and process mining techniques. Knowledge and Information Systems 21(3):267–297

Nakatumba J (2013) Resource-aware business process management: analysis and support. PhD thesis, Eindhoven University of Technology

Nakatumba J, van der Aalst WMP (2010) Analyzing resource behavior using process mining. Lecture Notes in Business Information Processing 43:69–80

Nakatumba J, Westergaard M, van der Aalst WMP (2012) Generating event logs with workload-dependent speeds from simulation models. Lecture Notes in Business Information Processing 112:383–397

Pika A, van der Aalst WMP, Fidge CJ, ter Hofstede AHM, Wynn MT (2013) Predicting deadline transgressions using event logs. Lecture Notes in Business Information Processing 132:211–216

Pospisil M, Hruška T (2012) Business process simulation for predictions. In: Proceedings of the Second International Conference on Business Intelligence and Technology, pp 14–18

Robinson S (2004) Simulation: the practice of model development and use. Wiley, Chichester

Rogge-Solti A, Kasneci G (2014) Temporal anomaly detection in business processes. Lecture Notes in Computer Science 8659:234–249

Rozinat A, van der Aalst WMP (2006a) Decision mining in business processes. Tech. Rep. BPM Center Report BPM-06-10

Rozinat A, van der Aalst WMP (2006b) Decision mining in ProM. Lecture Notes in Computer Science 4102:420–425

Rozinat A, Mans RS, Song M, van der Aalst WMP (2008) Discovering colored Petri nets from event logs. International Journal on Software Tools for Technology Transfer 10(1):57–74

Rozinat A, Mans RS, Song M, van der Aalst WMP (2009) Discovering simulation models. Information Systems 34(3):305–327

Schonenberg H, Jian J, Sidorova N, Van Der Aalst W (2010) Business trend analysis by simulation. Lecture Notes in Computer Science 6051:515–529

Senderovich A, Weidlich M, Gal A, Mandelbaum A (2014) Mining resource scheduling protocols. Lecture Notes in Computer Science 8659:200–216

Song M, van der Aalst WMP (2008) Towards comprehensive support for organizational mining. Decision Support Systems 46(1):300–317

Song M, Günther CW, Van der Aalst WMP (2009) Trace clustering in process mining. Lecture Notes in Business Information Processing 17:109–120

Suriadi S, Wynn MT, Xu J, van der Aalst WM, ter Hofstede AH (2017) Discovering work prioritisation patterns from event logs. Decision Support Systems 100:77–92

Szimanski F, Ralha CG, Wagner G, Ferreira DR (2013) Improving business process models with agent-based simulation and process mining. Lecture Notes in Business Information Processing 147:124–138

Tumay K (1996) Business process simulation. In: Proceedings of the 1996 Winter Simulation Conference, pp 55–60

Veiga GM, Ferreira DR (2010) Understanding spaghetti models with sequence clustering in ProM. Lecture Notes in Business Information Processing 43:92–103

Vincent S (1998) Input data analysis. In: Banks J (ed) Handbook of simulation: principles, advances, applications, and practice, John Wiley & Sons, Hoboken, pp 3–30

Wen Y, Chen Z, Liu J, Chen J (2013) Mining batch processing workflow models from event logs. Concurrency and Computation: Practice and Experience 25(13):1928–1942, DOI 10.1002/cpe. 2991

Wombacher A, Iacob ME (2013) Start time and duration distribution estimation in semi-structured processes. In: Proceedings of the 28th Annual ACM Symposium on Applied Computing, pp 1403–1409

Wombacher A, Iacob M, Haitsma M (2011) Towards a performance estimate in semi-structured processes. In: Proceedings of the 2011 IEEE International Conference on Service-Oriented Computing and Applications, pp 1–5

Xu R, Wunsch D (2005) Survey of clustering algorithms. Neural Networks, IEEE Transactions on 16(3):645–678

Cross-References

- Automated process discovery
- Decision discovery in business processes
- Queue Mining