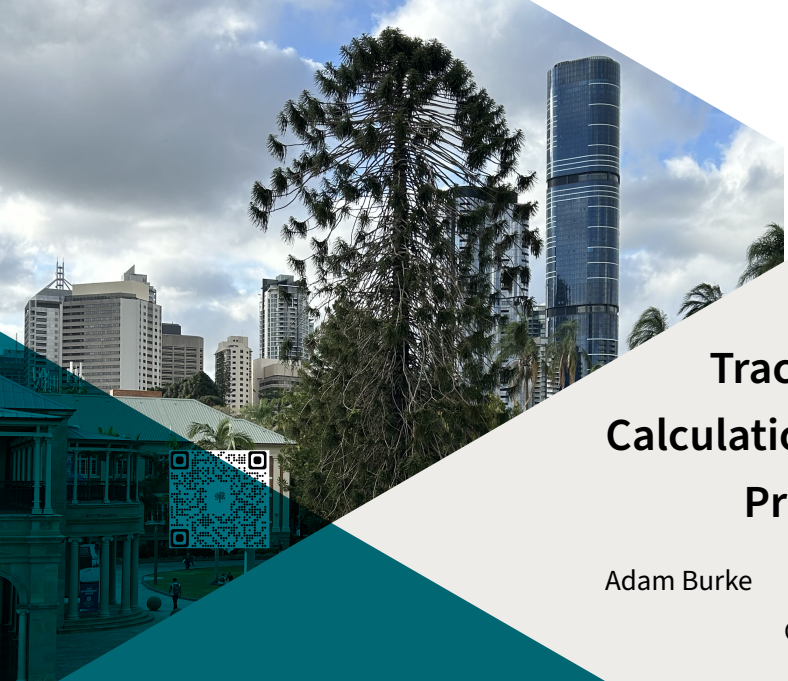


Trace Probability Calculation on Probabilistic Process Trees

Adam Burke Sander Leemans Moe Wynn

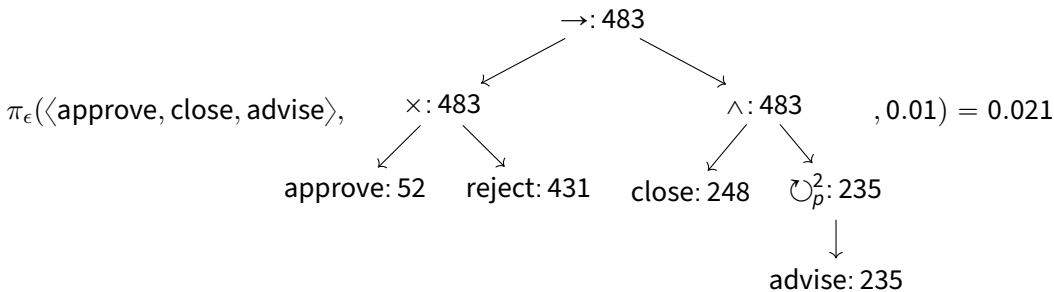
October 20, 2025

Introduction

Trace Probability

- What is the probability of a given trace through a stochastic process model?

Trace Probability - Preview



Trace Probability

- What is the probability of a given trace through a stochastic process model?

Trace Probability

- What is the probability of a given trace through a stochastic process model?
  - ... aka, TRACE-PROB [5]

Trace Probability

- What is the probability of a given trace through a stochastic process model?
  - ... aka, TRACE-PROB [5]
  - A number of solutions exist for Petri nets

Trace Probability

- What is the probability of a given trace through a stochastic process model?
  - ... aka, TRACE-PROB [5]
  - A number of solutions exist for Petri nets
- *This paper* has *two alternative* solutions. Both are

Trace Probability

- What is the probability of a given trace through a stochastic process model?
  - ... aka, TRACE-PROB [5]
  - A number of solutions exist for Petri nets
- *This paper* has *two alternative* solutions. Both are
  - ... closed form

Trace Probability

- What is the probability of a given trace through a stochastic process model?
  - ... aka, TRACE-PROB [5]
  - A number of solutions exist for Petri nets
- *This paper* has *two alternative* solutions. Both are
  - ... closed form
  - ... within an approximation bound

Trace Probability

- What is the probability of a given trace through a stochastic process model?
  - ... aka, TRACE-PROB [5]
  - A number of solutions exist for Petri nets
- *This paper* has *two alternative* solutions. Both are
  - ... closed form
  - ... within an approximation bound
- On (Weighted Stochastic) Finite Automata

Trace Probability

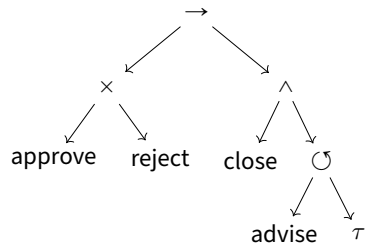
- What is the probability of a given trace through a stochastic process model?
  - ... aka, TRACE-PROB [5]
  - A number of solutions exist for Petri nets
- *This paper* has *two alternative* solutions. Both are
  - ... closed form
  - ... within an approximation bound
- On (Weighted Stochastic) Finite Automata
- On (Probabilistic) Process Trees

Trace Probability

- What is the probability of a given trace through a stochastic process model?
  - ... aka, TRACE-PROB [5]
  - A number of solutions exist for Petri nets
- *This paper* has *two alternative* solutions. Both are
  - ... closed form
  - ... within an approximation bound
- On (Weighted Stochastic) Finite Automata
- On (Probabilistic) Process Trees
  - ... that have an established translation to labelled stochastic nets

Process Trees

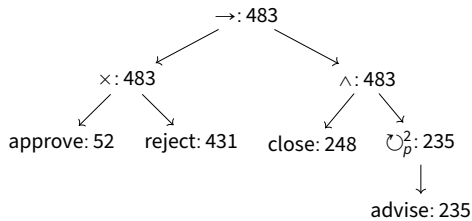
- Describe processes as composed, recursive, sub-processes
- Internal structure and external output of Inductive Miner [4] (and a few other things)
- Well-known translation[a] to block-structured Petri nets
-
  - Leaf nodes - activities and silents (τ
  - $\rightarrow$ Sequence
  - $\times$ Choice
  - $\wedge$ Concurrency
  - $\circlearrowleft$ Redo Loop



[a] ... depending on what conference one is presenting at ...

Probabilistic Process Trees

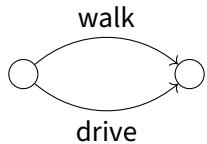
- Add weights to process trees [1]
  - Leaf nodes - activities and silents (τ)
  - $\rightarrow$ Sequence
  - $\times$ Choice, a weighted choice
  - $\wedge$ Concurrency, a weighted race block
  - $\circlearrowright_n^m$ Fixed Loop of length m
  - $\circlearrowright_n^\rho$ Probabilistic Loop with exit chance $\frac{1}{\rho}$
- Weight invariants between parents and children
- Translation to labelled stochastic Petri nets



Weighted Stochastic Finite Automata (WSFAs)

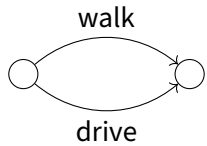
Weighted Stochastic Finite Automata by Example

- Finite Automata

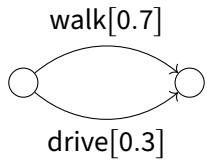


Weighted Stochastic Finite Automata by Example

- Finite Automata

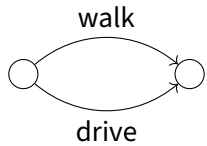


- Stochastic Finite Automata [6]

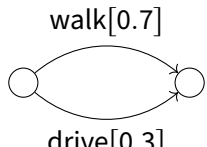


Weighted Stochastic Finite Automata by Example

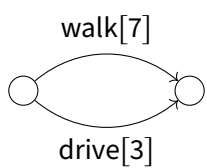
- Finite Automata



- Stochastic Finite Automata [6]



- Weighted Stochastic Finite Automata



Weighted Stochastic Finite Automata Definition

A *Weighted Stochastic Finite Automaton* (WSFA) is a five-tuple $(S, Act, \hookrightarrow, s_0, s_\omega)$, where

- S is a non-empty, finite set of states,

An assembly language for weighted stochastic process representations.

Weighted Stochastic Finite Automata Definition

A *Weighted Stochastic Finite Automaton* (WSFA) is a five-tuple $(S, Act, \hookrightarrow, s_0, s_\omega)$, where

- S is a non-empty, finite set of states,
- Act is the activities performed by the automata, $Act \subseteq \mathcal{A} \cup \{\tau\}$

An assembly language for weighted stochastic process representations.

Weighted Stochastic Finite Automata Definition

A *Weighted Stochastic Finite Automaton* (WSFA) is a five-tuple $(S, Act, \hookrightarrow, s_0, s_\omega)$, where

- S is a non-empty, finite set of states,
- Act is the activities performed by the automata, $Act \subseteq \mathcal{A} \cup \{\tau\}$
- $\hookrightarrow \subset S \times Act \times S \rightarrow \mathbb{R}^+$ is a weighted transition function

An assembly language for weighted stochastic process representations.

Weighted Stochastic Finite Automata Definition

A *Weighted Stochastic Finite Automaton* (WSFA) is a five-tuple $(S, Act, \hookrightarrow, s_0, s_\omega)$, where

- S is a non-empty, finite set of states,
- Act is the activities performed by the automata, $Act \subseteq \mathcal{A} \cup \{\tau\}$
- $\hookrightarrow \subset S \times Act \times S \rightarrow \mathbb{R}^+$ is a weighted transition function
- probability of transition is a weighted choice

An assembly language for weighted stochastic process representations.

Weighted Stochastic Finite Automata Definition

A *Weighted Stochastic Finite Automaton* (WSFA) is a five-tuple $(S, Act, \hookrightarrow, s_0, s_\omega)$, where

- S is a non-empty, finite set of states,
- Act is the activities performed by the automata, $Act \subseteq \mathcal{A} \cup \{\tau\}$
- $\hookrightarrow \subset S \times Act \times S \rightarrow \mathbb{R}^+$ is a weighted transition function
- probability of transition is a weighted choice
- graph is fully connected

An assembly language for weighted stochastic process representations.

Weighted Stochastic Finite Automata Definition

A *Weighted Stochastic Finite Automaton* (WSFA) is a five-tuple $(S, Act, \hookrightarrow, s_0, s_\omega)$, where

- S is a non-empty, finite set of states,
- Act is the activities performed by the automata, $Act \subseteq \mathcal{A} \cup \{\tau\}$
- $\hookrightarrow \subset S \times Act \times S \to \mathbb{R}^+$ is a weighted transition function
- probability of transition is a weighted choice
- graph is fully connected
- $s_0 \in S$ is the initial state, and has no incoming arcs

An assembly language for weighted stochastic process representations.

Weighted Stochastic Finite Automata Definition

A *Weighted Stochastic Finite Automaton* (WSFA) is a five-tuple $(S, Act, \hookrightarrow, s_0, s_\omega)$, where

- S is a non-empty, finite set of states,
- Act is the activities performed by the automata, $Act \subseteq \mathcal{A} \cup \{\tau\}$
- $\hookrightarrow \subset S \times Act \times S \to \mathbb{R}^+$ is a weighted transition function
- probability of transition is a weighted choice
- graph is fully connected
- $s_0 \in S$ is the initial state, and has no incoming arcs
- $s_\omega \in S$ is the final state, and a deadlock state, with no outgoing arcs.

An assembly language for weighted stochastic process representations.

Trace Probability on WSFAs

- Product of probabilities for each event in trace

Trace Probability on WSFAs

- Product of probabilities for each event in trace
- Loops and silence cause simple solution to create an infinite number of paths

Trace Probability on WSFAs

- Product of probabilities for each event in trace
- Loops and silence cause simple solution to create an infinite number of paths
  - ... as for other TRACE-PROB settings

Trace Probability on WSFAs

- Product of probabilities for each event in trace
- Loops and silence cause simple solution to create an infinite number of paths
  - ... as for other TRACE-PROB settings
- Approximate within bound

Trace Probability on WSFAs

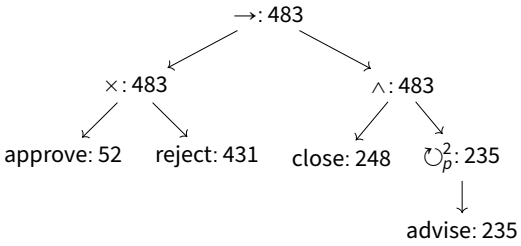
- Product of probabilities for each event in trace
- Loops and silence cause simple solution to create an infinite number of paths
  - ... as for other TRACE-PROB settings
- Approximate within bound
  - Take approximation epsilon parameter and allocate among paths

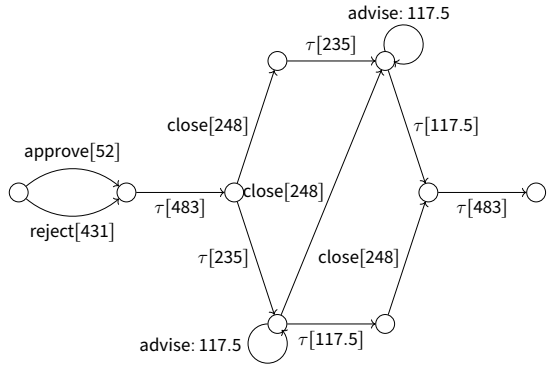
Trace Probability on WSFAs

- Product of probabilities for each event in trace
- Loops and silence cause simple solution to create an infinite number of paths
  - ... as for other TRACE-PROB settings
- Approximate within bound
  - Take approximation epsilon parameter and allocate among paths
  - When states repeat, stop reducing, terminate when below threshold

Probabilistic Process Trees (PPTs)

PPTs as WSFAs





Trace Probability on PPTs

- Exploit structure of PPTs

Trace Probability on PPTs

- Exploit structure of PPTs
- Approximate within probability bound

Trace Probability on PPTs

- Exploit structure of PPTs
- Approximate within probability bound
- Solve recursively by operator

Trace Probability on PPTs

- Exploit structure of PPTs
- Approximate within probability bound
- Solve recursively by operator
- Distinguish trees with finite silent activities (τ-bounded)

Trace Probability on PPTs

- Exploit structure of PPTs
- Approximate within probability bound
- Solve recursively by operator
- Distinguish trees with finite silent activities (τ-bounded)
- Concurrency uses WSFA translation

Approximation Within Bound

- For bound ϵ, unroll $\circlearrowright_p^\rho$ loop k times where $\frac{(\rho-1)^{k+1}}{(\rho)^k} \leqslant \epsilon$

Approximation Within Bound

- For bound ϵ, unroll $\circlearrowright_p^\rho$ loop k times where $\frac{(\rho-1)^{k+1}}{(\rho)^k} \leqslant \epsilon$
- Can then show sum of of fixed loop probabilities within $\pm\epsilon$

Approximation Within Bound

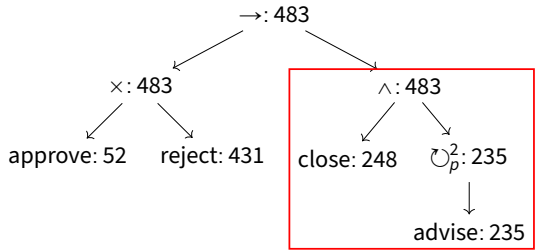
- For bound ϵ, unroll $\circlearrowright_p^\rho$ loop k times where $\frac{(\rho-1)^{k+1}}{(\rho)^k} \leqslant \epsilon$
- Can then show sum of of fixed loop probabilities within $\pm\epsilon$
- Fixed loops just behave as sequences

Approximation Within Bound

- For bound ϵ, unroll $\circlearrowright_p^\rho$ loop k times where $\frac{(\rho-1)^{k+1}}{(\rho)^k} \leqslant \epsilon$
- Can then show sum of of fixed loop probabilities within $\pm\epsilon$
- Fixed loops just behave as sequences
- Exact solutions for empty traces on loops, loops of a single leaf activity, also exist (from sum of a geometric progression)

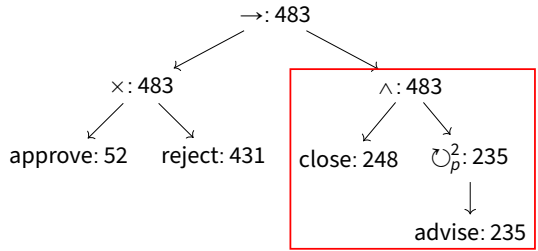
Concurrency

- In PPTs, concurrency is non-local



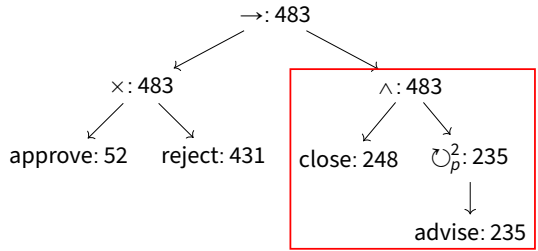
Concurrency

- In PPTs, concurrency is non-local
- ... but scoped to the top ancestral subtree parented by a concurrency operator, $\wedge$



Concurrency

- In PPTs, concurrency is non-local
- ... but scoped to the top ancestral subtree parented by a concurrency operator, $\wedge$
- Make use of the WSFA calculation for the entire subtree



Complexity

- Time complexity: $O(n \cdot k \cdot 2^m)$

Complexity

- Time complexity: $O(n \cdot k \cdot 2^m)$
  - n nodes in PPT

Complexity

- Time complexity: $O(n \cdot k \cdot 2^m)$
  - n nodes in PPT
  - k approximation bound

Complexity

- Time complexity: $O(n \cdot k \cdot 2^m)$
  - n nodes in PPT
  - k approximation bound
  - m largest concurrent tree

Complexity

- Time complexity: $O(n \cdot k \cdot 2^m)$
  - n nodes in PPT
  - k approximation bound
  - m largest concurrent tree
- Dominated by large concurrent trees, when present

Complexity

- Time complexity: $O(n \cdot k \cdot 2^m)$
  - n nodes in PPT
  - k approximation bound
  - m largest concurrent tree
- Dominated by large concurrent trees, when present
- More of a problem in low quality models

Complexity

- Time complexity: $O(n \cdot k \cdot 2^m)$
  - n nodes in PPT
  - k approximation bound
  - m largest concurrent tree
- Dominated by large concurrent trees, when present
- More of a problem in low quality models
- ... speculatively, stochastic perspective suggests avoiding broad-scope concurrency as a *goal* for discovery algorithms

Other Solutions

Petri Nets

- Linear Optimisation [5]

Petri Nets

- Linear Optimisation [5]
- Probabilistic Grammars with EM sampling [7]

Petri Nets

- Linear Optimisation [5]
- Probabilistic Grammars with EM sampling [7]
- Token game playout [2]

Process Trees

- Publication race condition ...

Process Trees

- Publication race condition ...
- András Horváth, Paolo Ballarini, and Pierre Cry - *Probabilistic Process Discovery with Stochastic Process Trees* [3]

Process Trees

- Publication race condition ...
- András Horváth, Paolo Ballarini, and Pierre Cry - *Probabilistic Process Discovery with Stochastic Process Trees* [3]
- Fixed probabilities at nodes to extend Process Trees

Process Trees

- Publication race condition ...
- András Horváth, Paolo Ballarini, and Pierre Cry - *Probabilistic Process Discovery with Stochastic Process Trees* [3]
- Fixed probabilities at nodes to extend Process Trees
- Trades-off direct weight interpretation for stochastic Petri nets for more straightforward calculations

Process Trees

- Publication race condition ...
- András Horváth, Paolo Ballarini, and Pierre Cry - *Probabilistic Process Discovery with Stochastic Process Trees* [3]
- Fixed probabilities at nodes to extend Process Trees
- Trades-off direct weight interpretation for stochastic Petri nets for more straightforward calculations
  - ... though other translations may well be possible

Process Trees

- Publication race condition ...
- András Horváth, Paolo Ballarini, and Pierre Cry - *Probabilistic Process Discovery with Stochastic Process Trees* [3]
- Fixed probabilities at nodes to extend Process Trees
- Trades-off direct weight interpretation for stochastic Petri nets for more straightforward calculations
  - ... though other translations may well be possible
- Uses maximum trace length (like playout solution [2] and no doubt others)

Process Trees

- Publication race condition ...
- András Horváth, Paolo Ballarini, and Pierre Cry - *Probabilistic Process Discovery with Stochastic Process Trees* [3]
- Fixed probabilities at nodes to extend Process Trees
- Trades-off direct weight interpretation for stochastic Petri nets for more straightforward calculations
  - ... though other translations may well be possible
- Uses maximum trace length (like playout solution [2] and no doubt others)
- Notes potential approximation bound solution

Conclusion

- Closed form solution shown, with approximation bound

Conclusion

- Closed form solution shown, with approximation bound
- Process trees productive data structure for stochastic perspective

Conclusion

- Closed form solution shown, with approximation bound
- Process trees productive data structure for stochastic perspective
- Structure around concurrency has large impact on model costs

Breaking Innovations in Stochastic Process Forestry

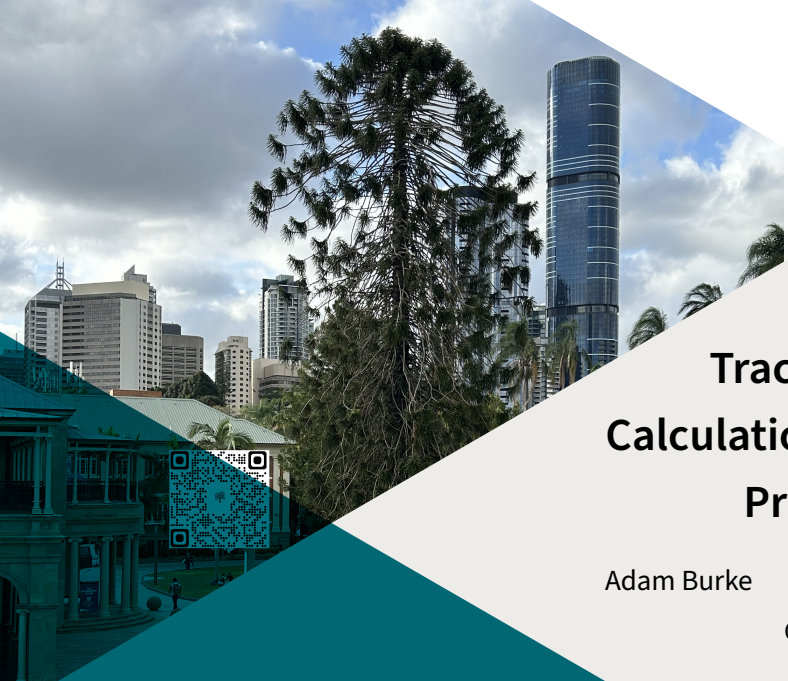
- Also here at ICPM 2025 ...

Breaking Innovations in Stochastic Process Forestry

- Also here at ICPM 2025 ...
- Pierre Cry, András Horváth, and Paolo Ballarini - *Stochastic Process Trees: A Formal Framework for Stochastic Process Discovery*

Breaking Innovations in Stochastic Process Forestry

- Also here at ICPM 2025 ...
- Pierre Cry, András Horváth, and Paolo Ballarini - *Stochastic Process Trees: A Formal Framework for Stochastic Process Discovery*
- Philipp Bär, Adam T. Burke, Moe T. Wynn, and Sander J. J. Leemans - *Skip Probabilities for Subprocesses*



Trace Probability Calculation on Probabilistic Process Trees

Adam Burke Sander Leemans Moe Wynn

October 20, 2025

References

Bibliography I

[1] Adam Burke, Sander J. J. Leemans, and Moe T. Wynn. “Discovering Stochastic Process Models By Reduction and Abstraction”. In: *Application and Theory of Petri Nets and Concurrency*. Lecture Notes in Computer Science. Springer, 2021, pp. 312–336.

[2] Adam T. Burke, Sander J. J. Leemans, Moe T. Wynn, Wil M. P. van der Aalst, and Arthur H. M. ter Hofstede. “A chance for models to show their quality: Stochastic process model-log dimensions”. In: *Information Systems* 124 (Sept. 2024), p. 102382. ISSN: 0306-4379.

[3] András Horváth, Paolo Ballarini, and Pierre Cry. “Probabilistic Process Discovery with Stochastic Process Trees”. In: *EAI VALUETOOLS 2024*. arXiv:2504.05765. Milan, 2024.

[4] Sander J. J. Leemans, Dirk Fahland, and Wil M. P. van der Aalst. “Scalable Process Discovery with Guarantees”. en. In: *Enterprise, Business-Process and Information Systems Modeling*. Ed. by Khaled Gaaloul, Rainer Schmidt, Selmin Nurcan, Sérgio Guerreiro, and Qin Ma. Lecture Notes in Business Information Processing. Cham: Springer International Publishing, 2015, pp. 85–101.

Bibliography II

[5] Sander J. J. Leemans, Fabrizio Maria Maggi, and Marco Montali. “Reasoning on Labelled Petri Nets and Their Dynamics in a Stochastic Setting”. en. In: *Business Process Management*. Lecture Notes in Computer Science. 2022, pp. 324–342.

[6] E. Vidal, F. Thollard, C. de la Higuera, F. Casacuberta, and R. C. Carrasco. “Probabilistic finite-state machines - part I”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 27.7 (July 2005), pp. 1013–1025. ISSN: 1939-3539.

[7] Akio Watanabe, Yousuke Takahashi, Hiroki Ikeuchi, and Kotaro Matsuda. “Grammar-Based Process Model Representation for Probabilistic Conformance Checking”. In: *2022 4th International Conference on Process Mining (ICPM)*. IEEE, 2022, pp. 88–95.