

Conformance Checking for Timed Process Models

Thomas Chatain

joint work with Neha Rino, Sofia Bellotti and Paolo Ballarini

LMF, ENS Paris-Saclay

Thomas Chatain

2000-2003 : **ingénieur** ENSEIRB + DEA (Bordeaux)

2003 : entrée 3^e année ENS Cachan (Ker Lann)

2003-2006 : **thèse** à l'IRISA (Rennes), directeur : Claude Jard

2006-2007 : **post-doctorat** à Aalborg, Danemark, équipe de Kim G. Larsen

Depuis 2007 : **maître de conférences** à l'ENS Paris-Saclay (ex-ENS Cachan)

2013 : **HDR**

Thèmes de recherche

Méthodes formelles pour
les systèmes distribués et temps-réel

Différents domaines d'application:

diagnostic, vérification, conception, systèmes biologiques, **process mining**

Process Mining

Young and very active research domain

At the interface between

- ▶ Data science
- ▶ Formal methods
- ▶ AI

patient	activity	timestamp	doctor	age	cost
5781	make X-ray	23-1-2014:10.30	Dr. Jones	45	70.00
5541	blood test	23-1-2014:10.18	Dr. Scott	61	40.00
5833	blood test	23-1-2014:10.27	Dr. Scott	24	40.00
5781	blood test	23-1-2014:10.49	Dr. Scott	45	40.00
5781	CT scan	23-1-2014:11.10	Dr. Fox	45	1200.00
5833	surgery	23-1-2014:12.34	Dr. Scott	24	2300.00
5781	handle payment	23-1-2014:12.41	Carol Hope	45	0.00
5541	radiation therapy	23-1-2014:13.57	Dr. Jones	61	140.00
5541	radiation therapy	23-1-2014:13.08	Dr. Jones	61	140.00

Process Mining

Young and very active research domain

At the interface between

- ▶ Data science
- ▶ Formal methods
- ▶ AI

patient	activity	timestamp	doctor	age	cost
5781	make X-ray	23-1-2014:10.30	Dr. Jones	45	70.00
5541	blood test	23-1-2014:10.18	Dr. Scott	61	40.00
5833	blood test	23-1-2014:10.27	Dr. Scott	24	40.00
5781	blood test	23-1-2014:10.49	Dr. Scott	45	40.00
5781	CT scan	23-1-2014:11.10	Dr. Fox	45	1200.00
5833	surgery	23-1-2014:12.34	Dr. Scott	24	2300.00
5781	handle payment	23-1-2014:12.41	Carol Hope	45	0.00
5541	radiation therapy	23-1-2014:13.57	Dr. Jones	61	140.00
5541	radiation therapy	23-1-2014:13.08	Dr. Jones	61	140.00

Discovery and analysis of process models from real process executions

- ▶ analyze medical processes in hospitals
- ▶ improve user interface
- ▶ detect deviant behavior
- ▶ ...

Many (Industrial) Process Mining Tools

- ▶ Celonis, Disco, Minit, ProM, PM4PY...

Event Logs and Data Extraction¹

patient	activity	timestamp	doctor	age	cost
5781	make X-ray	23-1-2014:10.30	Dr. Jones	45	70.00
5541	blood test	23-1-2014:10.18	Dr. Scott	61	40.00
5833	blood test	23-1-2014:10.27	Dr. Scott	24	40.00
5781	blood test	23-1-2014:10.49	Dr. Scott	45	40.00
5781	CT scan	23-1-2014:11.10	Dr. Fox	45	1200.00
5833	surgery	23-1-2014:12.34	Dr. Scott	24	2300.00
5781	handle payment	23-1-2014:12.41	Carol Hope	45	0.00
5541	radiation therapy	23-1-2014:13.57	Dr. Jones	61	140.00
5541	radiation therapy	23-1-2014:13.08	Dr. Jones	61	140.00

¹Acknowledgements to Wil van der Aalst

Event Logs and Data Extraction¹

patient	activity	timestamp
5781	make X-ray	23-1-2014:10.30
5541	blood test	23-1-2014:10.18
5833	blood test	23-1-2014:10.27
5781	blood test	23-1-2014:10.49
5781	CT scan	23-1-2014:11.10
5833	surgery	23-1-2014:12.34
5781	handle payment	23-1-2014:12.41
5541	radiation therapy	23-1-2014:13.57
5541	radiation therapy	23-1-2014:13.08

¹Acknowledgements to Wil van der Aalst

Event Logs and Data Extraction¹

patient	activity	timestamp
5781	make X-ray	23-1-2014:10.30
5541	blood test	23-1-2014:10.18
5833	blood test	23-1-2014:10.27
5781	blood test	23-1-2014:10.49
5781	CT scan	23-1-2014:11.10
5833	surgery	23-1-2014:12.34
5781	handle payment	23-1-2014:12.41
5541	radiation therapy	23-1-2014:13.57
5541	radiation therapy	23-1-2014:13.08

¹Acknowledgements to Wil van der Aalst

Event Logs and Data Extraction¹

patient	activity	timestamp
5781	make X-ray	23-1-2014:10.30
5781	blood test	23-1-2014:10.49
5781	CT scan	23-1-2014:11.10
5781	handle payment	23-1-2014:12.41
5541	blood test	23-1-2014:10.18
5541	radiation therapy	23-1-2014:13.57
5541	radiation therapy	23-1-2014:13.08
5833	blood test	23-1-2014:10.27
5833	surgery	23-1-2014:12.34

¹Acknowledgements to Wil van der Aalst

Event Logs and Data Extraction¹

patient	activity	timestamp
	make X-ray blood test CT scan handle payment	
	blood test radiation therapy radiation therapy	
	blood test surgery	

¹Acknowledgements to Wil van der Aalst

Event Logs and Data Extraction¹

patient	activity	timestamp
	X B C P	
	B R R	
	B S	

¹Acknowledgements to Wil van der Aalst

Event Logs and Data Extraction¹

patient	activity	timestamp
	X B C P	
	B R R	
	B S	

 $\langle X, B, C, P \rangle$ $\langle B, R, R \rangle$ $\langle B, S \rangle$

¹Acknowledgements to Wil van der Aalst

Process Discovery

Automatic construction of a model N from an event log L that represents a partial observation of a system S .

Process Discovery

Automatic construction of a model N from an event log L that represents a partial observation of a system S .

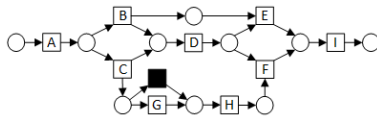
$\langle A, B, D, E, I \rangle$
 $\langle A, C, D, G, H, F, I \rangle$
 $\langle A, C, G, D, H, F, I \rangle$
 $\langle A, C, H, D, F, I \rangle$
 $\langle A, C, D, H, F, I \rangle$

L

Process Discovery

Automatic construction of a model N from an event log L that represents a partial observation of a system S .

$\langle A, B, D, E, I \rangle$
 $\langle A, C, D, G, H, F, I \rangle$
 $\langle A, C, G, D, H, F, I \rangle$
 $\langle A, C, H, D, F, I \rangle$
 $\langle A, C, D, H, F, I \rangle$

 L  N

Process Discovery: Several Solutions

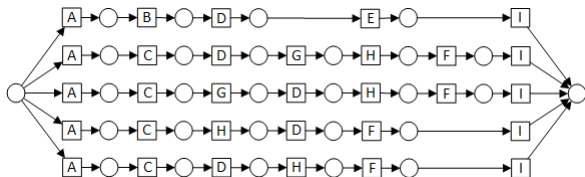
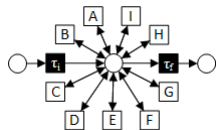
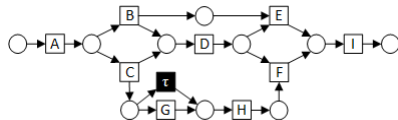
$\langle A, B, D, E, I \rangle$

$\langle A, C, D, G, H, F, I \rangle$

Log: $\langle A, C, G, D, H, F, I \rangle$

$\langle A, C, H, D, F, I \rangle$

$\langle A, C, D, H, F, I \rangle$



Conformance Checking

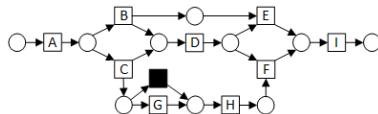
$\langle A, B, D, E, I \rangle$

$\langle A, C, D, G, H, F, I \rangle$

$\langle A, C, G, D, H, F, I \rangle$

$\langle A, C, H, D, F, I \rangle$

$\langle A, C, D, H, F, I \rangle$



Conformance Checking

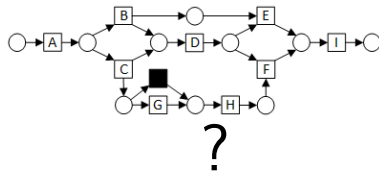
$\langle A, B, D, E, I \rangle$

$\langle A, C, D, G, H, F, I \rangle$

$\langle A, C, G, D, H, F, I \rangle$

$\langle A, C, H, D, F, I \rangle$

$\langle A, C, D, H, F, I \rangle$



Conformance Checking

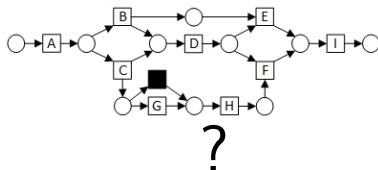
$\langle A, B, D, E, I \rangle$

$\langle A, C, D, G, H, F, I \rangle$

$\langle A, C, G, D, H, F, I \rangle$

$\langle A, C, H, D, F, I \rangle$

$\langle A, C, D, H, F, I \rangle$



Define quality criteria to evaluate models

- ▶ N fits L if $L \subseteq \mathcal{L}(N)$
- ▶ N is **precise** if its runs are *close* to traces of L
- ▶ N **generalizes** L if $\mathcal{L}(N)$ contains some unobserved behavior $\notin L$
- ▶ **simplicity**...

Conformance Checking: Example

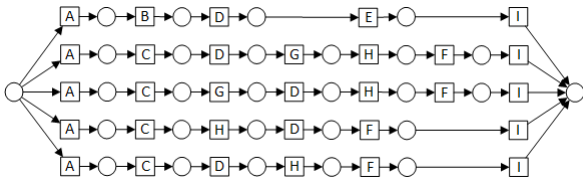
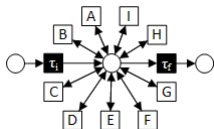
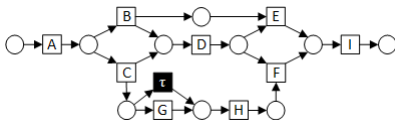
$\langle A, B, D, E, I \rangle$

$\langle A, C, D, G, H, F, I \rangle$

Log: $\langle A, C, G, D, H, F, I \rangle$

$\langle A, C, H, D, F, I \rangle$

$\langle A, C, D, H, F, I \rangle$



Conformance Checking: Example

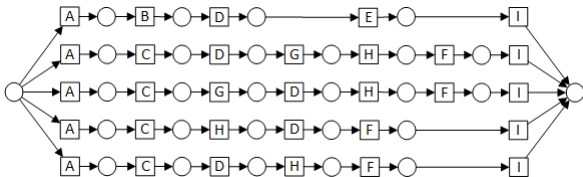
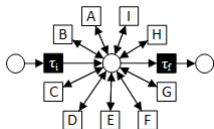
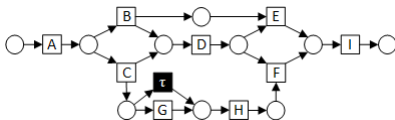
$\langle A, B, D, E, I \rangle$

$\langle A, C, D, G, H, F, I \rangle$

Log: $\langle A, C, G, D, H, F, I \rangle$

$\langle A, C, H, D, F, I \rangle$

$\langle A, C, D, H, F, I \rangle$



Conformance Checking: Example

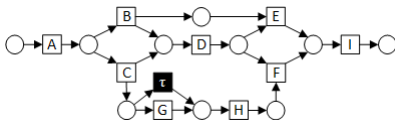
$\langle A, B, D, E, I \rangle$

$\langle A, C, D, G, H, F, I \rangle$

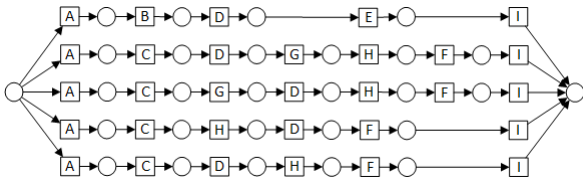
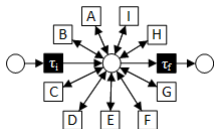
Log: $\langle A, C, G, D, H, F, I \rangle$

$\langle A, C, H, D, F, I \rangle$

$\langle A, C, D, H, F, I \rangle$



fitting
fairly precise
simple
generalizing



Conformance Checking: Example

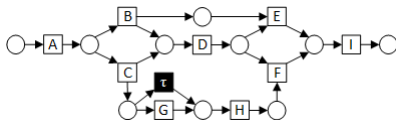
$\langle A, B, D, E, I \rangle$

$\langle A, C, D, G, H, F, I \rangle$

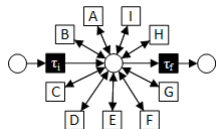
Log: $\langle A, C, G, D, H, F, I \rangle$

$\langle A, C, H, D, F, I \rangle$

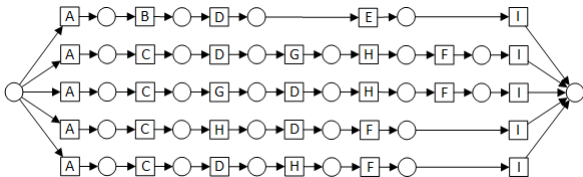
$\langle A, C, D, H, F, I \rangle$



fitting
fairly precise
simple
generalizing



fitting
very imprecise
simple
generalizing



Conformance Checking: Example

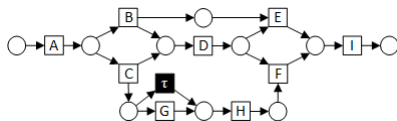
$\langle A, B, D, E, I \rangle$

$\langle A, C, D, G, H, F, I \rangle$

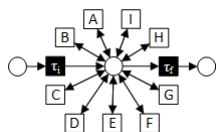
Log: $\langle A, C, G, D, H, F, I \rangle$

$\langle A, C, H, D, F, I \rangle$

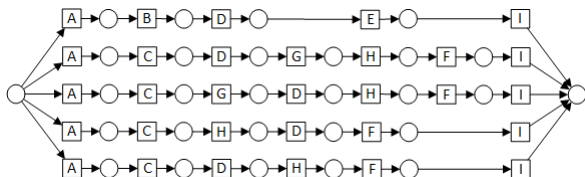
$\langle A, C, D, H, F, I \rangle$



fitting
fairly precise
simple
generalizing



fitting
very imprecise
simple
generalizing



fitting
very precise
not simple
not generalizing

Alignments

Estimating fitness

$\langle A, B, D, E, I \rangle$

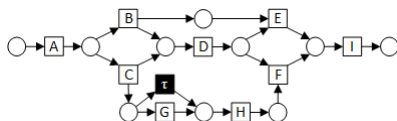
$\langle A, C, D, G, H, F, I \rangle$

$\langle A, C, G, D, H, F, I \rangle$

Log: $\langle A, C, H, D, F, I \rangle$

$\langle A, C, D, H, F, I \rangle$

$\langle A, B, D, G, H, F, I \rangle$



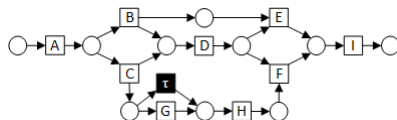
Trace $\sigma = \langle A, B, D, G, H, F, I \rangle$

Alignments

Estimating fitness

Log:

- $\langle A, B, D, E, I \rangle$
- $\langle A, C, D, G, H, F, I \rangle$
- $\langle A, C, G, D, H, F, I \rangle$
- $\langle A, C, H, D, F, I \rangle$
- $\langle A, C, D, H, F, I \rangle$
- $\langle A, B, D, G, H, F, I \rangle$



Trace $\sigma = \langle A, B, D, G, H, F, I \rangle$

Alignment

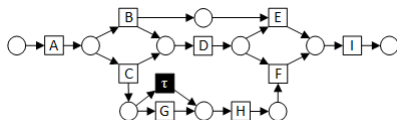
Given a trace σ and a model N ,
an alignment is a full run γ of N which minimizes its (Levenshtein) distance to σ .

Alignments

Estimating fitness

Log:

- $\langle A, B, D, E, I \rangle$
- $\langle A, C, D, G, H, F, I \rangle$
- $\langle A, C, G, D, H, F, I \rangle$
- $\langle A, C, H, D, F, I \rangle$
- $\langle A, C, D, H, F, I \rangle$
- $\langle A, B, D, G, H, F, I \rangle$



Trace $\sigma = \langle A, B, D, G, H, F, I \rangle$

Alignment: $\gamma = \langle A, \textcolor{red}{C}, D, G, H, F, I \rangle$

Alignment

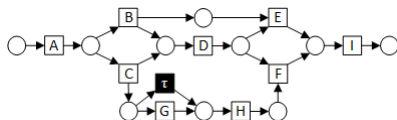
Given a trace σ and a model N ,
an alignment is a full run γ of N which minimizes its (Levenshtein) distance to σ .

Alignments

Estimating fitness

Log:

- $\langle A, B, D, E, I \rangle$
- $\langle A, C, D, G, H, F, I \rangle$
- $\langle A, C, G, D, H, F, I \rangle$
- $\langle A, C, H, D, F, I \rangle$
- $\langle A, C, D, H, F, I \rangle$
- $\langle A, B, D, G, H, F, I \rangle$



Trace $\sigma = \langle A, B, D, G, H, F, I \rangle$

Alignment: $\gamma = \langle A, \textcolor{red}{C}, D, G, H, F, I \rangle$

Alignment

Given a trace σ and a model N ,
an alignment is a full run γ of N which minimizes its (Levenshtein) distance to σ .

Central notion in conformance checking

Timestamps

But sometimes, we want to retain the timing information :

Customer ID	Action	timestamp
70	reg	13:31
70	et	13:32
70	ct	13:33
70	dec	13:37
70	pay	13:38
71	reg	13:32
71	ec	13:34
71	ct	13:34
71	dec	13:36
71	rej	13:36
72	reg	13:34
72	ct	13:36

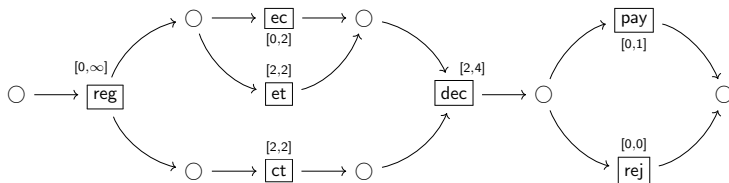
Time-aware Conformance Checking

Example of a process where time matters



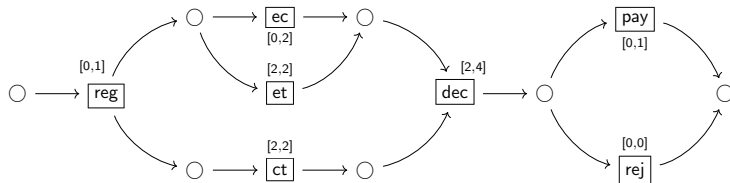
Time Constraints : Time Petri Nets

Now, maybe we could have a process model like :



Timed Alignments [ICPM'22, BPI'23, DKE'24]

Now suppose we keep the timestamps



Say we observe the timed trace

$$(reg, 1) \cdot (ec, 2) \cdot (ec, 2) \cdot (ct, 3) \cdot (dec, 8) \cdot (rej, 8)$$

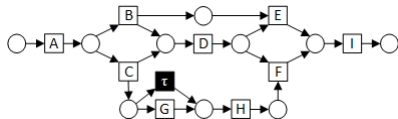
We want a framework timed alignments

The Alignment Problem

The Alignment Problem [Adriansyah14]

Input : A Petri net N and an observed trace σ

Output : A full run γ of N which minimizes its (Levenshtein) distance to σ .



Trace $\sigma = \langle A, B, D, G, H, F, I \rangle$

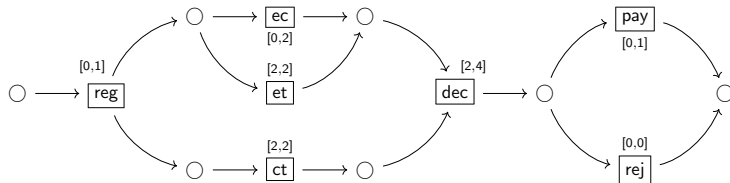
Alignment: $\gamma = \langle A, \textcolor{red}{C}, D, G, H, F, I \rangle$

The General Timed Alignment Problem

GTAP

Input : A **time** Petri net N and an observed **timed** trace σ .

Output : A full **timed** trace $\gamma \in \mathcal{L}(N)$ which minimizes its distance to σ , for some distance function d on timed words.



Trace: $\sigma = (reg, 1) \cdot (ec, 2) \cdot (ec, 2) \cdot (ct, 3) \cdot (dec, 8) \cdot (rej, 8)$

Alignment? $\gamma_1 = (reg, 1) \cdot (ec, 2) \cdot (ct, 3) \cdot (dec, 7) \cdot (rej, 7)$

Alignment? $\gamma_2 = (reg, 1) \cdot (ec, 2) \cdot (ct, 3) \cdot (dec, 7) \cdot (pay, 8)$

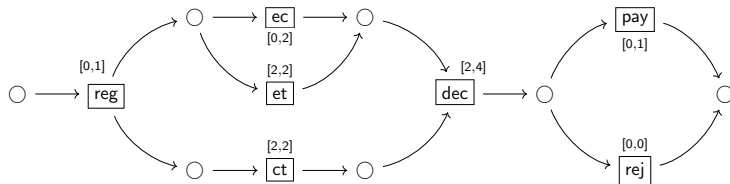
The Purely Timed Alignment Problem



PTAP

Input : A time Petri net N and an observed timed trace $\sigma = (w_1, \theta_1) \dots (w_n, \theta_n)$ such that w is a valid untimed trace of N .

Output : A timing function η such that $\gamma = (w_1, \eta_1) \dots (w_n, \eta_n)$ is a full timed trace of N and minimizes $d(\theta, \eta)$ for some distance function d on timing functions.



Trace: $\sigma = (reg, 1) \cdot (ec, 2) \cdot (ct, 3) \cdot (dec, 8) \cdot (rej, 8)$

Alignment: $\gamma = (reg, 1) \cdot (ec, 2) \cdot (ct, 3) \cdot (dec, 7) \cdot (rej, 7)$

Distances between traces

- ▶ Untimed Alignments : Levenshtein's edit distance
- ▶ What are meaningful edit moves over timestamp series?

Edit Moves

Consider a time series,

$$\sigma = (1, 2, 10, 14, 15)$$

$$\textit{stamp}(+5, 2)\sigma = (1, 7, 10, 14, 15)$$

Cost : $|+5|$.

Edit Moves

Consider a time series,

$$\sigma = (1, 2, 10, 14, 15)$$

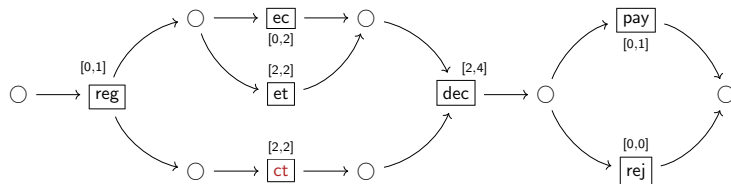
$$\textit{stamp}(+5, 2)\sigma = (1, 7, 10, 14, 15)$$

Cost : $|+5|$.

$$\textit{delay}(+5, 2)\sigma = (1, 7, 15, 19, 20)$$

Cost : $|+5|$.

Delay Edit : Branching example



$$\sigma = (reg, 1) \cdot (ct, 2) \cdot (ec, 2) \cdot (dec, 6) \cdot (rej, 6)$$

$$delay(1, ct)\sigma = (reg, 1) \cdot (ec, 2) \cdot (ct, 3) \cdot (dec, 7) \cdot (rej, 7)$$

Cost : $|1|$

Three distances : Stamp Only Distance (d_t)

Definition

$$d_t(\tau_1 \rightarrow \tau_2) = \min\{cost(m) \mid m \in Stamp^*, m(\tau_1) = \tau_2\}$$

$$d_t((0, 1, 2) \rightarrow (1, 3, 3)) = |1| + |2| + |1| = 4$$

Essentially Manhattan distance.

Three distances : Delay Only Distance (d_θ)

Definition

$$d_\theta(\tau_1 \rightarrow \tau_2) = \min\{\text{cost}(m) \mid m \in \text{Delay}^*, m(\tau_1) = \tau_2\}$$

$$(0, 1, 2) \xrightarrow{\text{delay}(1,1)} (1, 2, 3) \xrightarrow{\text{delay}(1,2)} (1, 3, 4) \xrightarrow{\text{delay}(-1,3)} (1, 3, 3)$$

$$d_\theta((0, 1, 2) \rightarrow (1, 3, 3)) = |1| + |1| + |-1| = 3$$

Essentially Manhattan distance on the gaps between consecutive timestamps.

Three distances : Mixed Moves Distance (d_m)

Definition

$$d_m(\tau_1 \rightarrow \tau_2) = \min\{\text{cost}(m) \mid m \in (\text{Stamp} \cup \text{Delay})^*, m(\tau_1) = \tau_2\}$$

$$(0, 1, 2) \xrightarrow{\text{delay}(1,1)} (1, 2, 3) \xrightarrow{\text{stamp}(1,2)} (1, 3, 3)$$

$$d_m((0, 1, 2) \rightarrow (1, 3, 3)) = |1| + |1| = 2$$

Properties

- ▶ $d_m(\tau_1 \rightarrow \tau_2) \leq d_t(\tau_1 \rightarrow \tau_2)$
- ▶ $d_m(\tau_1 \rightarrow \tau_2) \leq d_\theta(\tau_1 \rightarrow \tau_2)$
- ▶ $d_\theta(\tau_1 \rightarrow \tau_2) \leq 2 \times d_t(\tau_1 \rightarrow \tau_2)$ for sequential models

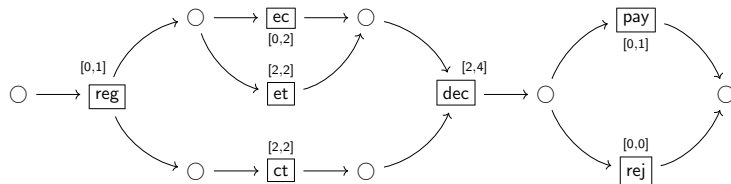
Recall PTAP

With our newly defined distance functions, we set about solving PTAP for each.

PTAP

Input : A time Petri net N and an observed timed trace $\sigma = (w_1, \theta_1) \dots (w_n, \theta_n)$ and a valid causal process of the untimed part w .

Output : A timing function η such that $\gamma = (w_1, \eta_1) \dots (w_n, \eta_n)$ is a full timed trace of N and minimizes $d(\theta, \eta)$ for some distance function d on timing functions.



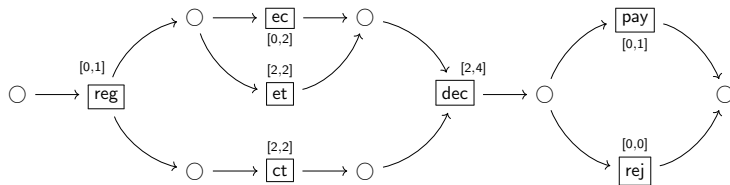
Trace: $\sigma = (reg, 1) \cdot (ec, 2) \cdot (ct, 3) \cdot (dec, 8) \cdot (rej, 8)$

Alignment: $\gamma = (reg, 1) \cdot (ec, 2) \cdot (ct, 3) \cdot (dec, 7) \cdot (rej, 7)$

Delay only, d_θ

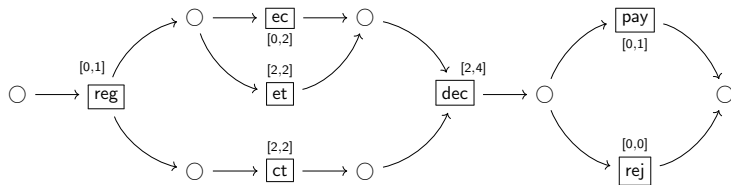
- ▶ For a large class of nets (extended free choice), a linear-time algorithm!
- ▶ This distance allows a naive, greedy algorithm.

Delay only Alignment



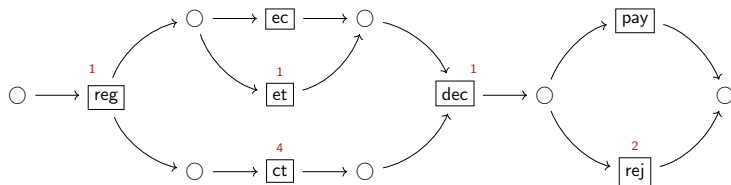
Say we want to align $\sigma = (reg, 1)(et, 2)(ct, 5)(dec, 6)(rej, 8)$ to N .

Delay only Alignment

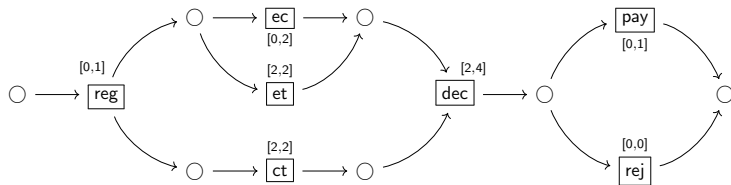


Say we want to align $\sigma = (reg, 1)(et, 2)(ct, 5)(dec, 6)(rej, 8)$ to N .

Consider the delays taken for a faulty “execution” guided by σ .

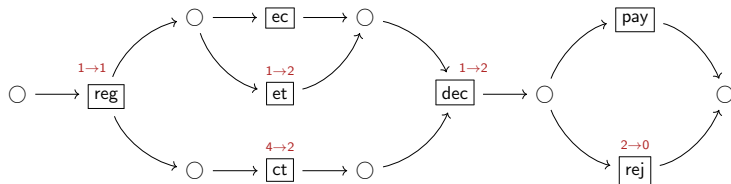


Delay only Alignment

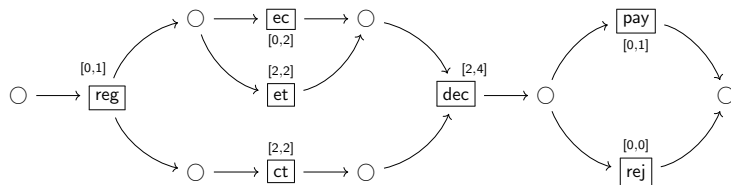


Say we want to align $\sigma = (reg, 1)(et, 2)(ct, 5)(dec, 6)(rej, 8)$ to N .

Consider the delays taken for a faulty “execution” guided by σ .
Pick the best values possible for them, naively.

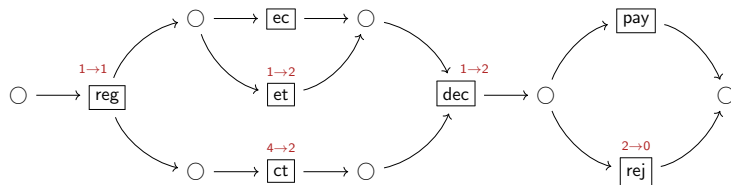


Delay only Alignment



Say we want to align $\sigma = (reg, 1)(et, 2)(ct, 5)(dec, 6)(rej, 8)$ to N .

Consider the delays taken for a faulty “execution” guided by σ .
Pick the best values possible for them, naively.

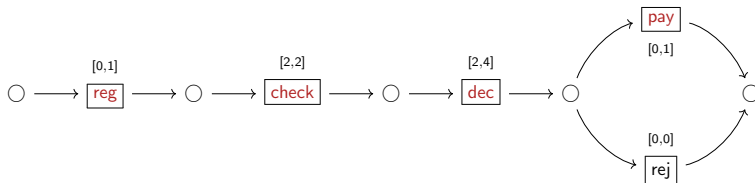


From which we reconstruct the closest $\gamma = (reg, 1)(et, 3)(ct, 3)(dec, 5)(rej, 5)$.

Stamp only, d_t

- ▶ In general for bounded nets, we have an encoding as an LPP.
- ▶ For sequential causal processes, a direct quadratic time algorithm!

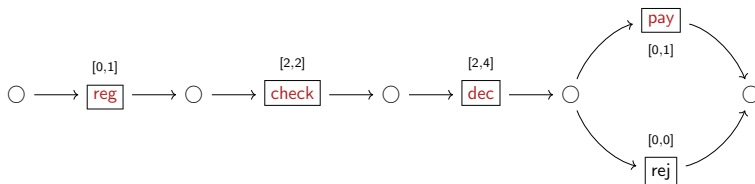
Stamp only Alignment



Trace: $\sigma = (\text{reg}, 1) \cdot (\text{check}, 2) \cdot (\text{dec}, 4) \cdot (\text{pay}, 6)$

Optimal alignments

Stamp only Alignment

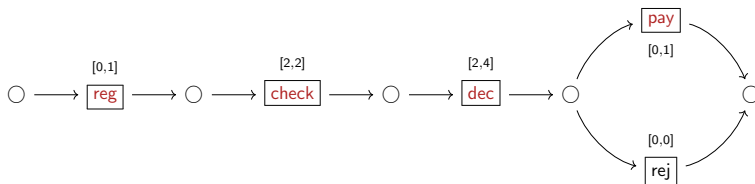


Trace: $\sigma = (reg, 1) \cdot (check, 2) \cdot (dec, 4) \cdot (pay, 6)$

Optimal alignments
(examples): $\gamma = (reg, 0) \cdot (check, 2) \cdot (dec, 4) \cdot (pay, 5)$
 $\gamma = (reg, 1) \cdot (check, 3) \cdot (dec, 5) \cdot (pay, 6)$

Alignment cost: $d_t(\sigma, \gamma) = 2$

Stamp only Alignment



Trace: $\sigma = (reg, 1) \cdot (check, 2) \cdot (dec, 4) \cdot (pay, 6)$

Optimal alignments (examples): $\gamma = (reg, 0) \cdot (check, 2) \cdot (dec, 4) \cdot (pay, 5)$
 $\gamma = (reg, 1) \cdot (check, 3) \cdot (dec, 5) \cdot (pay, 6)$

Alignment cost: $d_t(\sigma, \gamma) = 2$

How to compute them?

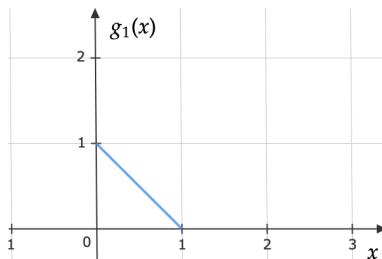
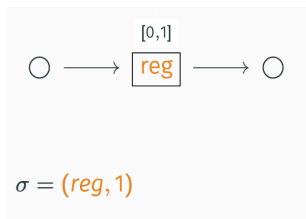
Stamps Only



$$\sigma = (\text{reg}, 1)(\text{check}, 2)(\text{dec}, 4)$$

Define $g_i(d)$ the minimum cost of aligning the i -length prefix $\sigma_1 \dots \sigma_i$ to any $\gamma_1 \dots \gamma_i \in \mathcal{L}^i(N)$, with the constraint that the timestamp of γ_i should be d .

Example

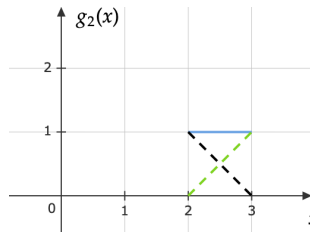
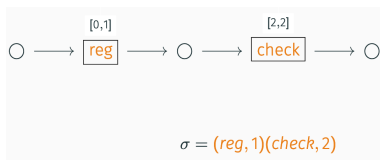


$$g_1(d) = d_t((reg, d) \rightarrow (reg, 1)) = |d - 1| = |d - \sigma_1|$$

Stamp only algorithm for sequential models, continued

Split cost at the last letter to get the following recursion :

$$g_{i+1}(d) = |d - \sigma_{i+1}| + \min_{d-d' \in [a,b]} g_i(d')$$

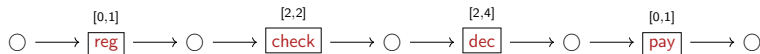
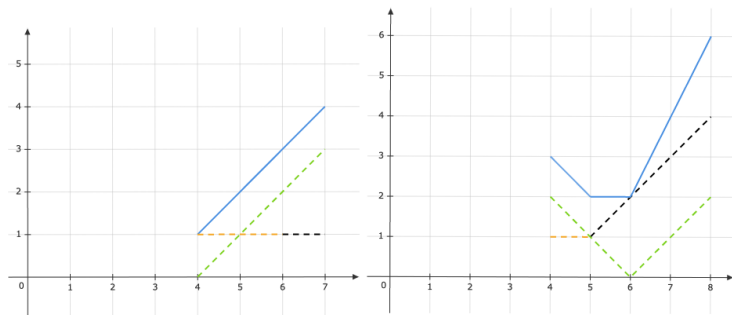


$$d_t((\text{reg}, d-2)(\text{check}, d) \rightarrow (\text{reg}, 1)(\text{check}, 2)) =$$

$$g_2(d) = |d - 2| + (g_1(d - 2))$$

Diagrams: g_3, g_4

In the i th graph, the colour key is $g_i(d)$, $\min_{d-d' \in [a,b]} g_{i-1}(d')$, and $|d - \sigma_i|$



Trace: $\sigma = (reg, 1) \cdot (check, 2) \cdot (dec, 4) \cdot (pay, 6)$

Optimal alignments $\gamma = (reg, 0) \cdot (check, 2) \cdot (dec, 4) \cdot (pay, 5)$

(examples): $\gamma = (reg, 1) \cdot (check, 3) \cdot (dec, 5) \cdot (pay, 6)$

Summary: Algorithm for Stamp only Alignment

- ▶ Compute the g_i 's incrementally
 - ▶ Store them as lists of segments
 - ▶ no more than i segments for g_i
 - ▶ slopes of g_i segments are integers in $[-i, i]$
- $O(n^2)$ time
- ▶ The minimum cost of alignment is $c = \min_d g_n(d)$.
 - ▶ Reconstruct an optimal alignment γ :
 - ▶ Pick some d which minimizes the cost $g_n(d)$, d is the timestamp of γ_n
 - ▶ Then backtrack to get the timestamp of γ_{n-1}
 - ▶ ...

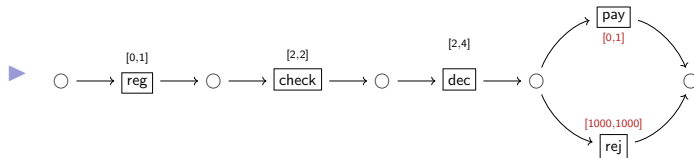
Implementation in Python

Available at <https://github.com/NehaRino/TimedAlignments>

Summary of Contributions

- ▶ We set the problem of General and Purely Timed Alignment for process models.
- ▶ We defined three metrics for timed words, and studied the purely timed alignment problem over them.
- ▶ For certain classes of process models, we provide original and efficient algorithms to solve purely timed alignment
 - ▶ Quadratic time for Stamp only Alignment
 - ▶ Linear time for Mixed moves Alignment

Future Directions : Back to GTAP



- Approach 1 : First align actions, then timestamps
- Consider : $\sigma = (reg, 0)(check, 2)(dec, 4)(pay, 1004)$
- Approach 1 is quite biased towards actions, but it is a start.
- Approach 2 : set a cost for action edits.

Aligning Observed Timed Traces with Timed Stochastic Models joint work with Sofia Bellotti, Paolo Ballarini

An example

Recorded traces (log):

$\langle a, b, c, d \rangle$

$\langle a, b, c, d \rangle$

$\langle a, c, b, d \rangle$

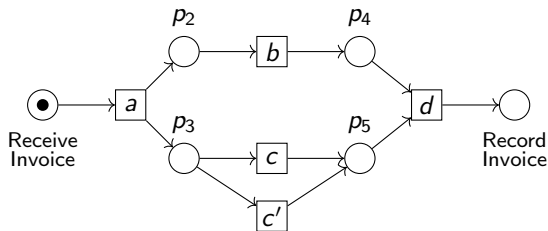
$\langle a, b, c, d \rangle$

$\langle a, b, c, d \rangle$

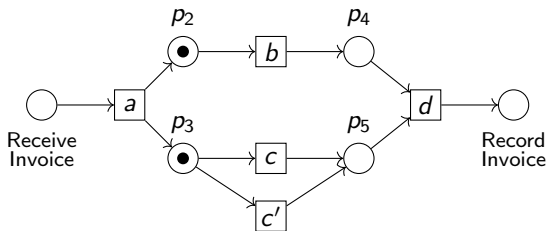
$\langle a, b, c', d \rangle$

- ▶ a : capture invoice data
- ▶ b : verify purchase order
- ▶ c : check fraud (already known client)
- ▶ c' : check fraud (new client)
- ▶ d : validate, approve, pay invoice

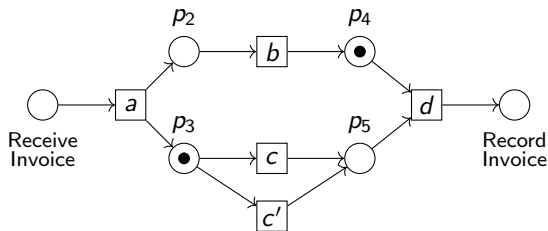
An example

 $\langle a, b, c, d \rangle$ $\langle a, b, c, d \rangle$ $\langle a, c, b, d \rangle$ $\langle a, b, c, d \rangle$ $\langle a, b, c, d \rangle$ $\langle a, b, c', d \rangle$ 

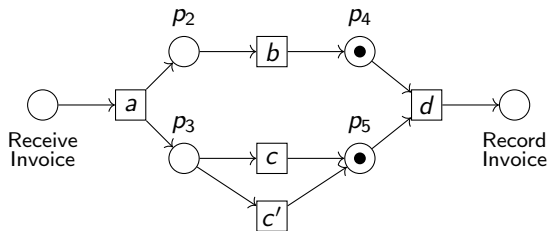
An example

 $\langle a, b, c, d \rangle$ $\langle a, b, c, d \rangle$ $\langle a, c, b, d \rangle$ $\langle a, b, c, d \rangle$ $\langle a, b, c, d \rangle$ $\langle a, b, c', d \rangle$ 

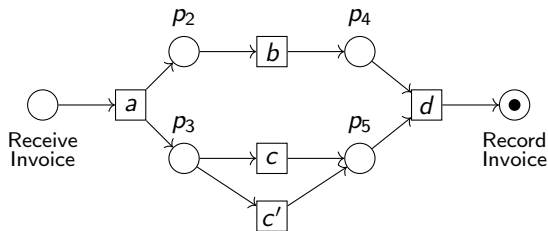
An example

 $\langle a, b, c, d \rangle$ $\langle a, b, c, d \rangle$ $\langle a, c, b, d \rangle$ $\langle a, b, c, d \rangle$ $\langle a, b, c, d \rangle$ $\langle a, b, c', d \rangle$ 

An example

 $\langle a, b, c, d \rangle$ $\langle a, b, c, d \rangle$ $\langle a, c, b, d \rangle$ $\langle a, b, c, d \rangle$ $\langle a, b, c, d \rangle$ $\langle a, b, c', d \rangle$ 

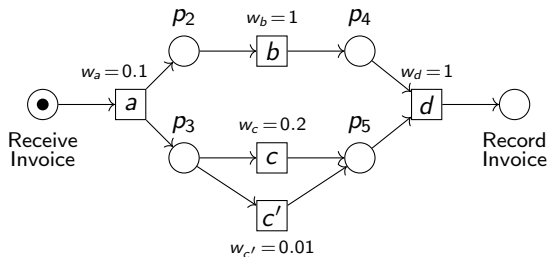
An example

 $\langle a, b, c, d \rangle$ $\langle a, b, c, d \rangle$ $\langle a, c, b, d \rangle$ $\langle a, b, c, d \rangle$ $\langle a, b, c, d \rangle$ $\langle a, b, c', d \rangle$ 

Stochastic Process Model

 $\langle a, b, c, d \rangle^4$
 $\langle a, c, b, d \rangle$
 $\langle a, b, c', d \rangle$

Weighted transitions

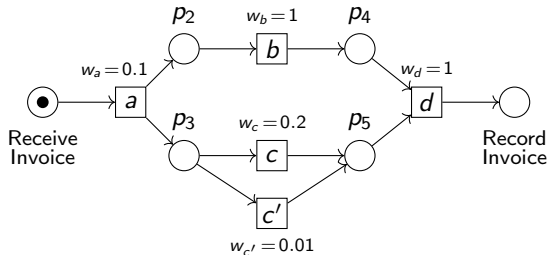


Often represent the fact that some runs are observed more frequently than others.
Give a probability distribution over the runs (probabilistic language).

Timed Stochastic Process Model

Now weights are interpreted as rates of exponential distribution laws for the waited time before a transition fires.

- $w_a = 0.1$ means: probability $0.1 \times dt$ that a fires in the next dt time units.
 → exponential distribution with expected waiting time = $\frac{1}{0.1} = 10$ units.



Semantics defined using Continuous-Time Markov Chains

→ (dense) probability distribution over **timed runs**

Aligning Timed Traces with Timed Stochastic Models

$\langle (a, 10.2), (b, 11.3), (c, 14.1), (d, 14.9) \rangle$

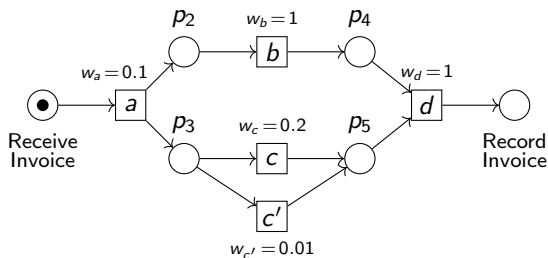
$\langle (a, 9.0), (b, 10.2), (c, 15.6), (d, 16.2) \rangle$

$\langle (a, 1.1), (b, 10.2), (c, 14.6), (d, 15.5) \rangle$

$\langle (a, 8.3), (b, 9.1), (c, 15.1), (d, 16.2) \rangle$

$\langle (a, 12.3), (c, 12.5), (b, 13.2), (d, 19.1) \rangle$

$\langle (a, 9.2), (b, 10.2), (c', 45.2), (d, 45.9) \rangle$



Aligning Timed Traces with Timed Stochastic Models

$\langle (a, 10.2), (b, 11.3), (c, 14.1), (d, 14.9) \rangle$

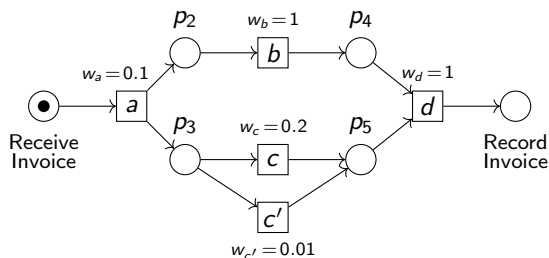
$\langle (a, 9.0), (b, 10.2), (c, 15.6), (d, 16.2) \rangle$

$\langle (a, 1.1), (b, 10.2), (c, 14.6), (d, 15.5) \rangle$

$\langle (a, 8.3), (b, 9.1), (c, 15.1), (d, 16.2) \rangle$

$\langle (a, 12.3), (c, 12.5), (b, 13.2), (d, 19.1) \rangle$

$\langle (a, 9.2), (b, 10.2), (c', 45.2), (d, 45.9) \rangle$



Assumption: perfectly fitting model!

... but some recorded traces have very low likelihood.

Aligning Timed Traces with Timed Stochastic Models

$\langle (a, 10.2), (b, 11.3), (c, 14.1), (d, 14.9) \rangle \rightarrow$ very likely

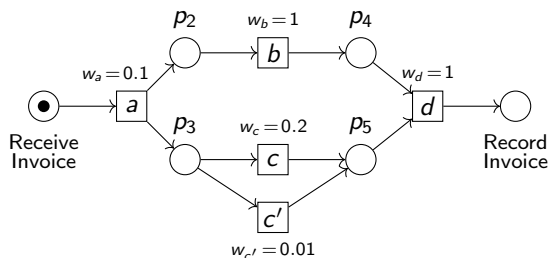
$\langle (a, 9.0), (b, 10.2), (c, 15.6), (d, 16.2) \rangle \rightarrow$ very likely

$\langle (a, 1.1), (b, 10.2), (c, 14.6), (d, 15.5) \rangle \rightarrow$ surprising

$\langle (a, 8.3), (b, 9.1), (c, 15.1), (d, 16.2) \rangle \rightarrow$ very likely

$\langle (a, 12.3), (c, 12.5), (b, 13.2), (d, 19.1) \rangle \rightarrow$ surprising

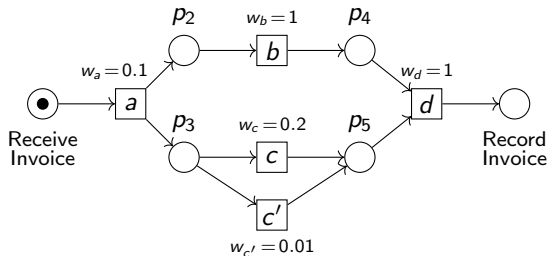
$\langle (a, 9.2), (b, 10.2), (c', 45.2), (d, 45.9) \rangle$



Assumption: perfectly fitting model!

...but some recorded traces have very low likelihood.

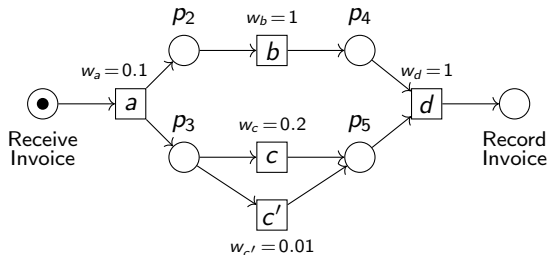
Aligning Timed Traces with Timed Stochastic Models



$\langle (a, 1.1), (b, 10.2), (c, 14.6), (d, 15.5) \rangle \rightarrow \text{surprising}$

Very unlikely as it is

Aligning Timed Traces with Timed Stochastic Models



$\langle (a, 1.1), (b, 10.2), (c, 14.6), (d, 15.5) \rangle \rightarrow \text{surprising}$

Very unlikely as it is

... what if a actually occurred later (around 9) and its timestamp was misrecorded?

Aligning Timed Traces with Timed Stochastic Models

$$\hat{\sigma} = \langle (a, 1.1), (b, 10.2), (c, 14.6), (d, 15.5) \rangle \rightarrow \text{surprising}$$

We define the alignment of an observed trace $\hat{\sigma} = \langle (a_1, \hat{\theta}_1), \dots, (a_n, \hat{\theta}_n) \rangle$ with a timed stochastic model as a model run $\sigma = \langle (t_1, \theta_1), \dots, (t_n, \theta_n) \rangle$ such that

- ▶ $\lambda(t_i) = a_i$ for all i
- ▶ the timestamps of σ optimize an objective function which is a **compromise** between
 - ▶ likelihood of σ and
 - ▶ closeness to the observed timestamps: $\|\theta - \hat{\theta}\|_{L_1} = \sum_i |\theta_i - \hat{\theta}_i|$

Alignment of Timed Traces with Timed Stochastic Models

becomes an optimization problem

Minimize the objective function

$$\alpha(-\ell(\sigma)) + \beta \|\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}\|_{L_1} \quad (1)$$

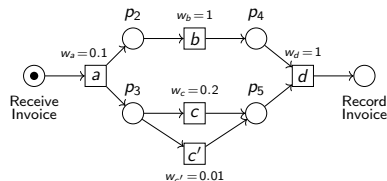
where

- ▶ $\ell(\sigma) = \prod_{i=1}^n w_{t_i} e^{-W_{t_i}(\theta_i - \theta_{i-1})}$ denotes the likelihood of $\sigma = \langle (t_1, \theta_1), \dots, (t_n, \theta_n) \rangle$ according to the model, and
- ▶ $\alpha, \beta \geq 0$ are tunable parameters such that $\alpha + \beta = 1$. The choice of α and β balances the emphasis between the likelihood and the distance to observed times.

Experiments

We used log-likelihood instead of likelihood

- makes the objective function linear



Results

for the observed trace $\hat{\sigma} = \langle (a, 1.1), (b, 10.2), (c, 14.6), (d, 15.5) \rangle$

α	β	θ
0.0000	1.0000	[1.1, 10.2, 14.6, 15.5]
0.2500	0.7500	[1.1, 10.2, 14.6, 15.5]
0.4760	0.5240	[10.2, 10.2, 14.6, 15.5]
0.5000	0.5000	[10.2, 10.2, 14.6, 15.5]
0.5560	0.4440	[10.2, 10.2, 15.5, 15.5]
0.7500	0.2500	[10.2, 10.2, 15.5, 15.5]
0.9520	0.0480	[15.5, 15.5, 15.5, 15.5]
1.0000	0.0000	[15.5, 15.5, 15.5, 15.5]

Conclusion

- ▶ Defined the first alignment notion between observed timed traces and timed stochastic process models
- ▶ Experimental results confirm our intuition of how a trace aligns to a model
- ▶ Because of the likelihood of traces, even alignment to a perfectly fitting model is the result of a compromise that we solve using optimization techniques

Perspectives

- ▶ For non-fitting models, combine with the techniques known for untimed models, usually based on model moves and log moves

Anti-alignments

Anti-alignments: Measuring Precision

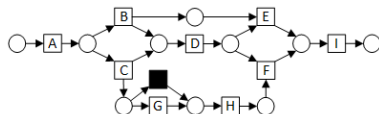
$\langle A, B, D, E, I \rangle$

$\langle A, C, D, G, H, F, I \rangle$

$\langle A, C, G, D, H, F, I \rangle$

$\langle A, C, H, D, F, I \rangle$

$\langle A, C, D, H, F, I \rangle$



Quality criteria to evaluate models

- ▶ N fits L if $L \subseteq \mathcal{L}(N)$
- ▶ N is **precise** if its runs are *close* to traces of L
- ▶ N **generalizes** L if $\mathcal{L}(N)$ contains some unobserved behavior $\notin L$
- ▶ **simplicity**...

Alignments are central for measuring **fitness**

Anti-alignments: Measuring Precision

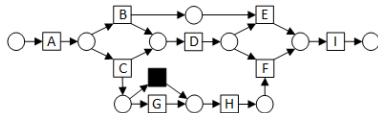
$\langle A, B, D, E, I \rangle$

$\langle A, C, D, G, H, F, I \rangle$

$\langle A, C, G, D, H, F, I \rangle$

$\langle A, C, H, D, F, I \rangle$

$\langle A, C, D, H, F, I \rangle$



Quality criteria to evaluate models

- ▶ N fits L if $L \subseteq \mathcal{L}(N)$
- ▶ N is **precise** if its runs are *close* to traces of L
- ▶ N **generalizes** L if $\mathcal{L}(N)$ contains some unobserved behavior $\notin L$
- ▶ **simplicity**...

Alignments are central for measuring **fitness**

Anti-alignments are used to measure **precision**.

- ▶ PhD thesis of Mathilde Boltenhagen

Anti-alignments – Motivation

Estimating precision

A model is **precise** if it produces only traces which are *close* to log traces.

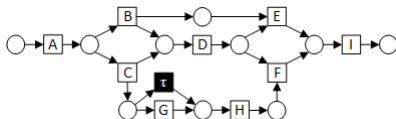
Log L :

$\langle A, C, D, G, H, F, I \rangle$

$\langle A, C, G, D, H, F, I \rangle$

$\langle A, C, D, H, F, I \rangle$

$\langle A, C, H, D, F, I \rangle$



Motivation

In order to measure precision, find the run of N which is most **misaligned** with the log L .

Anti-alignments – Motivation

Estimating precision

A model is **precise** if it produces only traces which are *close* to log traces.

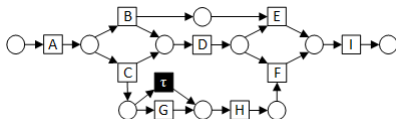
Log L :

$\langle A, C, D, G, H, F, I \rangle$

$\langle A, C, G, D, H, F, I \rangle$

$\langle A, C, D, H, F, I \rangle$

$\langle A, C, H, D, F, I \rangle$



Motivation

In order to measure precision, find the run of N which is most **misaligned** with the log L .

Here: $\langle A, B, D, E, I \rangle$

Anti-alignments – Results

Precision metric based on anti-alignments [ICATPN'2016, BPM'2016, ICPM'2021, Inf. Systems'2021]

- ▶ Anti-alignment serves as witness for *imprecision*
- ▶ Precision metric verifies properties of [Tax et al., IPL 2018]
- ▶ Several implementations: SAT encoding, then very efficient approximations (discounted edit distance)

Anti-alignments for timed models [ATAED'25]

(joint work with Stefano Bavaro and Boudewijn van Dongen)

Find the timing allowed by the model which is the furthest from what has been observed.

- ▶ Encoding as a Mixed Integer Programming Problem

Future Directions

Conformance checking

- ▶ Take into account
 - ▶ time information
 - ▶ other quantitative information (cost, mark)

Model repair

- ▶ How to modify/edit a model incrementally in order to improve its quality?

Comparison of models

- ▶ Spot differences between several models of the same process

Expressiveness of formalisms/classes of models, e.g. BPMN vs. Petri nets

Thank you!