



Fast Conformance Analysis based on Activity Log Abstraction

Alok Dixit, [Eric Verbeek](#), Wil van der Aalst

Eindhoven University of Technology

Outline

- Process Mining
- Motivation
- Approach
- Evaluation
- Conclusion

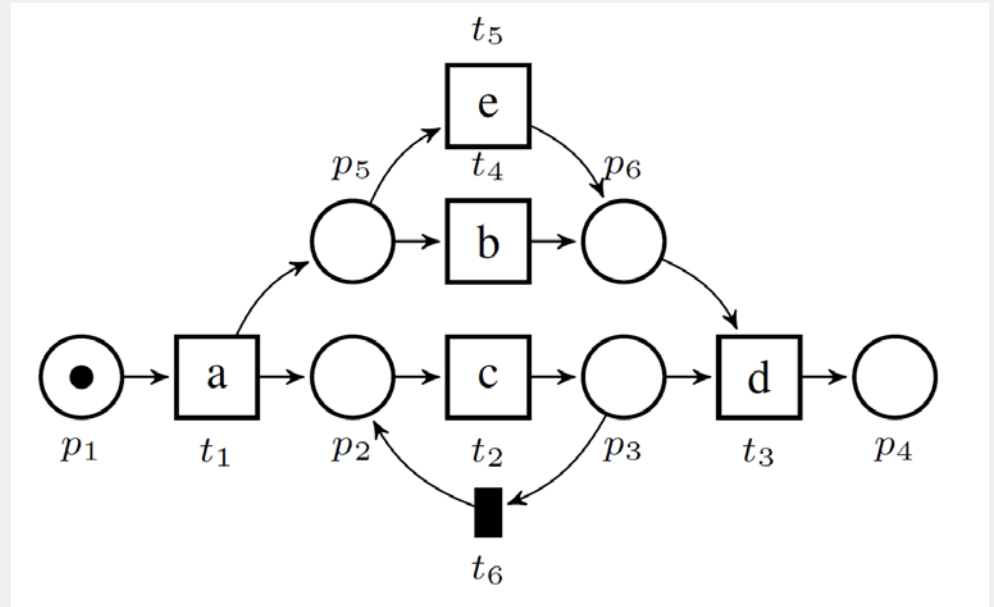
Outline

- Process Mining
- Motivation
- Approach
- Evaluation
- Conclusion

Activity (or Event) Log

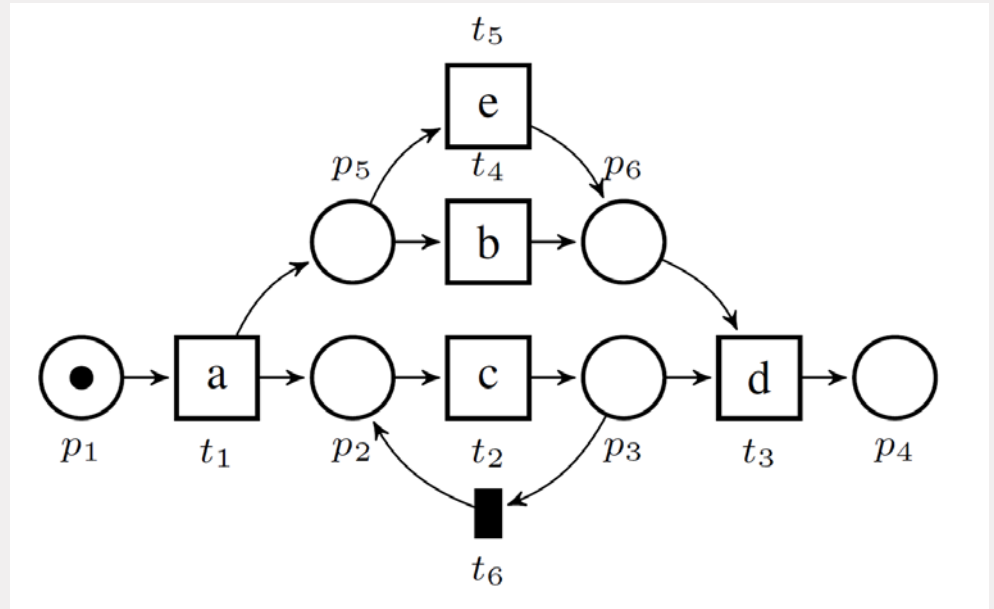
Activity trace	#
<a,b,c,d>	1
<a,b,c,c,d>	5
<a,b,c,c,c,d>	1
<a,e,c,c,d>	1
<a,e,c,d>	6
<a,c,b,d>	3
<a,c,e,c,c,c,d>	1

Process Model



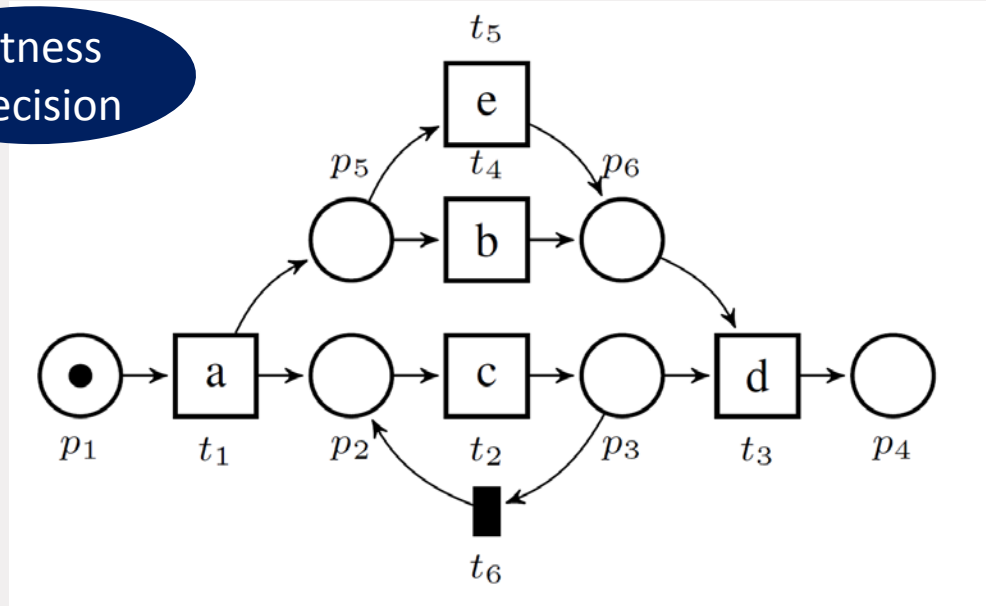
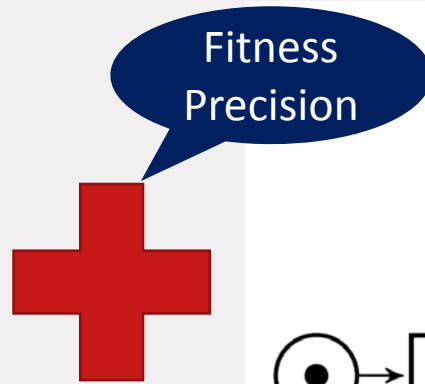
Process Discovery

Activity trace	#
<a,b,c,d>	1
<a,b,c,c,d>	5
<a,b,c,c,c,d>	1
<a,e,c,c,d>	1
<a,e,c,d>	6
<a,c,b,d>	3
<a,c,e,c,c,c,d>	1



Process Conformance

Activity trace	#
$\langle a, b, c, d \rangle$	1
$\langle a, b, c, c, d \rangle$	5
$\langle a, b, c, c, c, d \rangle$	1
$\langle a, e, c, c, d \rangle$	1
$\langle a, e, c, d \rangle$	6
$\langle a, c, b, d \rangle$	3
$\langle a, c, e, c, c, c, d \rangle$	1



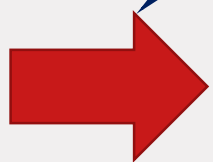
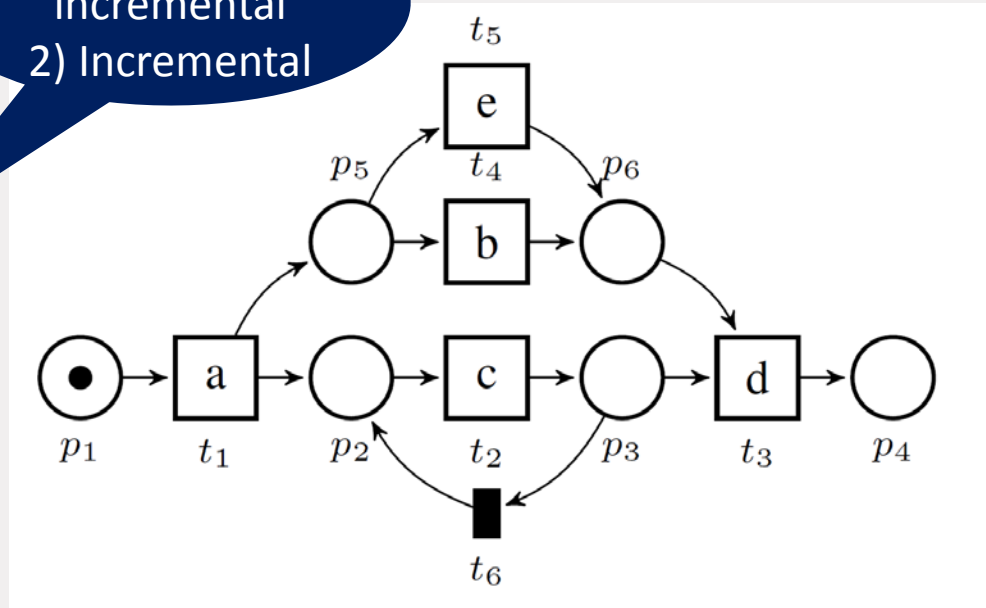
Outline

- Process Mining
 - Process discovery and process conformance
- **Motivation**
- Approach
- Evaluation
- Conclusion

Process Discovery

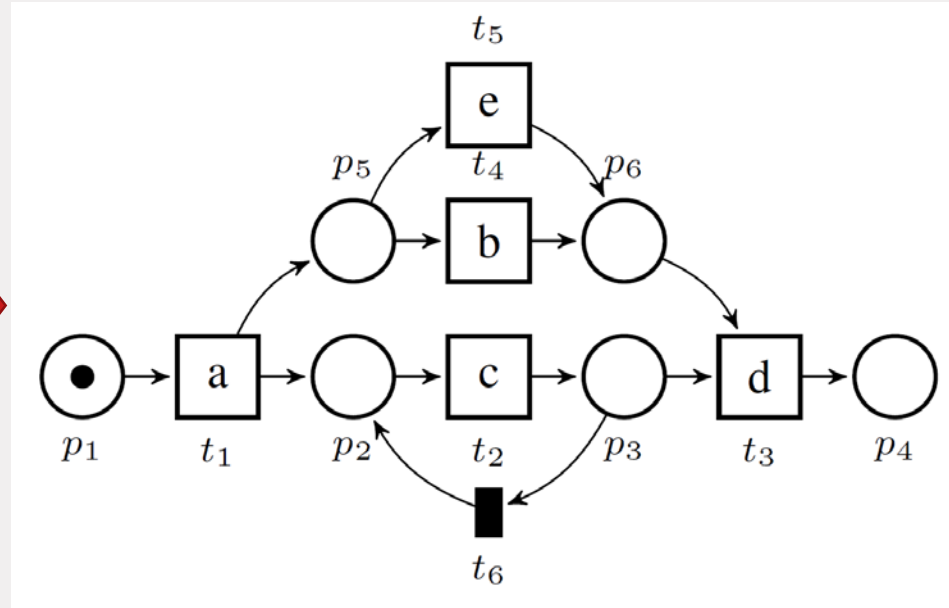
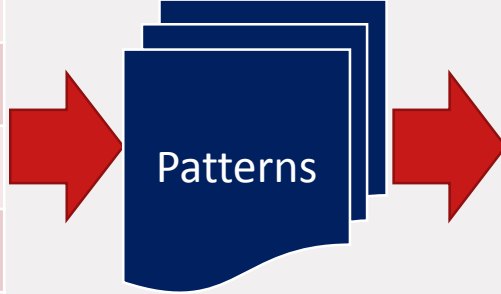
Activity trace	#
<a,b,c,d>	1
<a,b,c,c,d>	5
<a,b,c,c,c,d>	1
<a,e,c,c,d>	1
<a,e,c,d>	6
<a,c,b,d>	3
<a,c,e,c,c,c,d>	1

1) Non-incremental
2) Incremental

Non-incremental Process Discovery

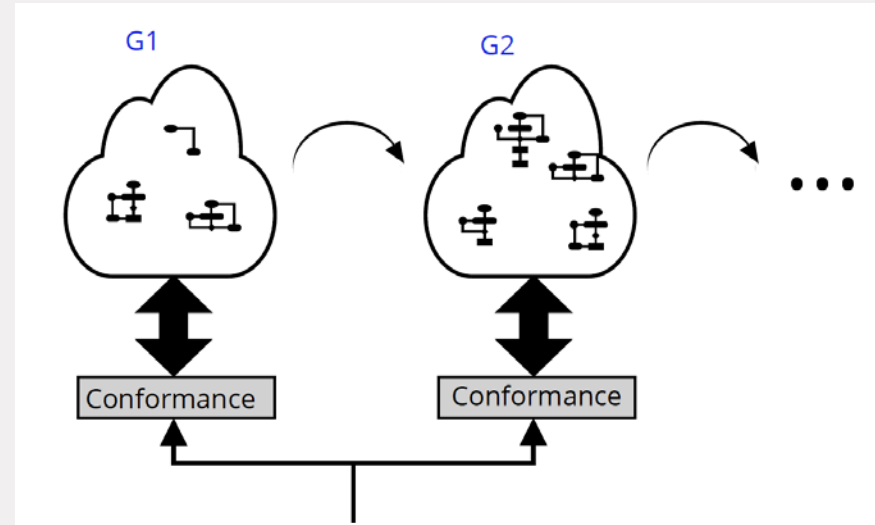
Activity trace	#
<a,b,c,d>	1
<a,b,c,c,d>	5
<a,b,c,c,c,d>	1
<a,e,c,c,d>	1
<a,e,c,d>	6
<a,c,b,d>	3
<a,c,e,c,c,c,d>	1



Incremental Process Discovery I

Examples: Genetic Discovery.

- Multiple process models created in each generation
- Every process model in each generation is compared to the activity log
- *Best* process models make it to the next generation

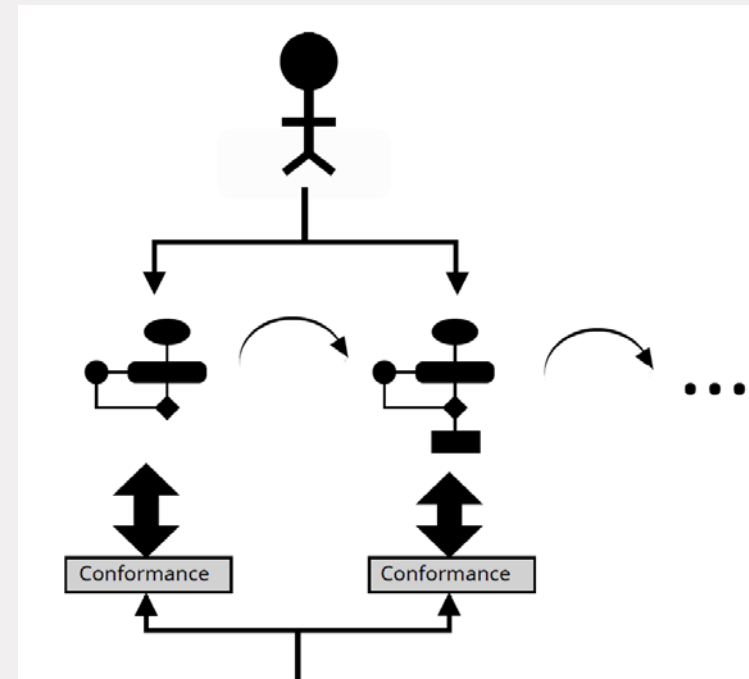


Activity trace	#
<a,b,c,d>	1
<a,b,c,c,d>	5

Incremental Process Discovery II

Example: Human-in-the-loop process discovery

- User incrementally creates a process model
- Every change in the process model is compared to the event log – using conformance analysis



Activity trace	#
<a,b,c,d>	1
<a,b,c,c,d>	5

For incremental process discovery ...

... we need fast process conformance ...

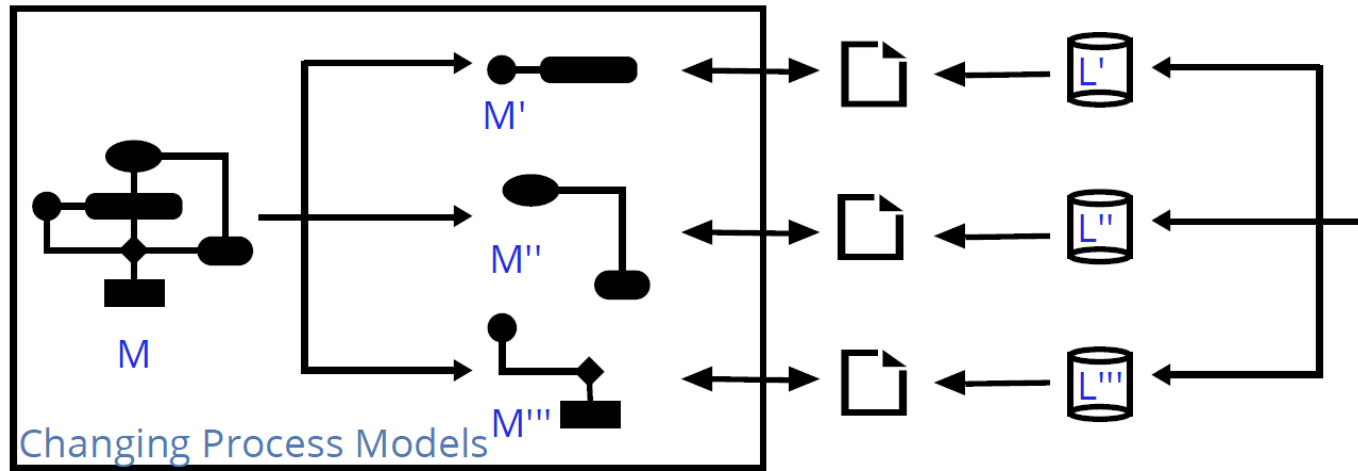
... as in every iteration we need to check conformance

Outline

- Process Mining
 - Process discovery and process conformance
- Motivation:
 - For incremental process discovery, we need fast process conformance
- **Approach**
- Evaluation
- Conclusion

Approach

1. Extract patterns from activity log *initially (only once)*
2. Extract similar patterns from the (changing) process models
3. Compare the patterns to gain conformance insights



Activity trace	#
$\langle a, b, c, d \rangle$	1
$\langle a, b, c, c, d \rangle$	5
$\langle a, b, c, c, c, d \rangle$	1
$\langle a, e, c, c, d \rangle$	1
$\langle a, e, c, d \rangle$	6
$\langle a, c, b, d \rangle$	3
$\langle a, c, e, c, c, c, d \rangle$	1

Patterns

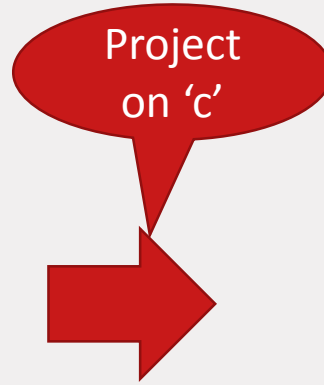
- Unary patterns
- Binary patterns
- Fitness and precision

Patterns

- Unary patterns
- Binary patterns
- Fitness and precision

Extracting Unary Patterns from Activity Log

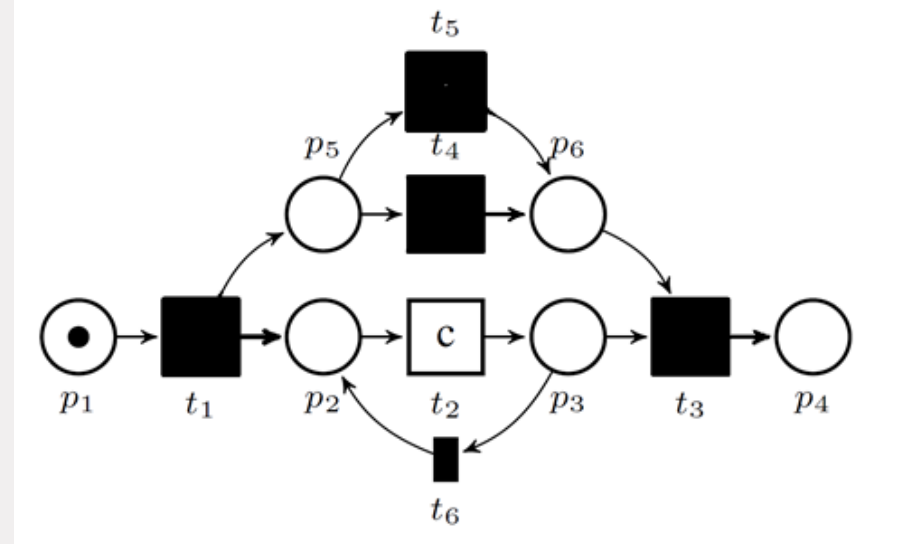
Activity trace	#
<a,b,c,d>	1
<a,b,c,c,d>	5
<a,b,c,c,c,d>	1
<a,e,c,c,d>	1
<a,e,c,d>	6
<a,c,b,d>	3
<a,c,e,c,c,c,d>	1



Activity trace	#
<c>	10
<c,c>	6
<c,c,c>	1
<c,c,c,c>	1

Extracting Unary Patterns from Process Model

- For each activity in the process model, a unary footprint pattern represents pair of values (min, max).
- Project model on a single activity, e.g. Activity 'c'
 - Compute the (min) minimum number of times 'c' can be executed.
min = 1
 - Compute the (max) maximum number of times 'c' can be executed.
max = ∞
 - Unary pattern of c in model = (1, ∞)

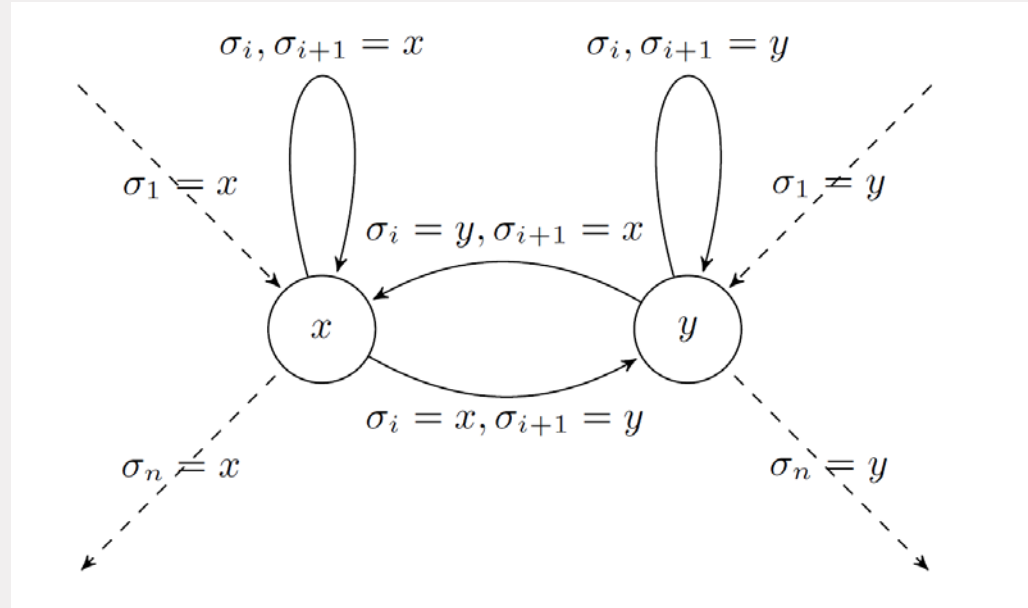


Patterns

- Unary patterns
- Binary patterns
- Fitness and precision

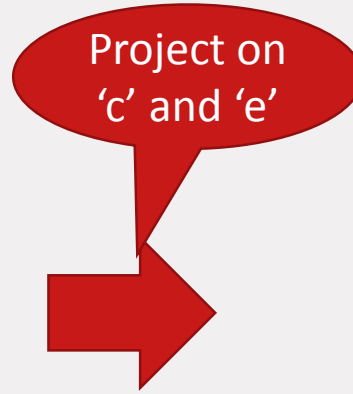
Extracting Binary Patterns from Activity Log

- Pairwise relation between activities in a trace – aggregated over the entire log
- Project a trace on two activities (x,y)
- Consider the following projected trace
 $\sigma = \langle \sigma_1, \sigma_2, \dots, \sigma_n \rangle$;
Such that for any '1 <= i <= n':
 $\sigma_i = 'x'$ or $\sigma_i = 'y'$
- Remove all the *unsatisfied* arcs



Extracting Binary Patterns from Activity Log

Activity trace	#
<a,b,c,d>	1
<a,b,c,c,d>	5
<a,b,c,c,c,d>	1
<a,e,c,c,d>	1
<a,e,c,d>	6
<a,c,b,d>	3
<a,c,e,c,c,c,d>	1



Activity trace	#
<c>	4
<c,c>	5
<c,c,c>	1
<e,c,c,c,c>	1
<e,c>	6
<c,e,c,c,c>	1

Extracting Binary Patterns from Activity Log

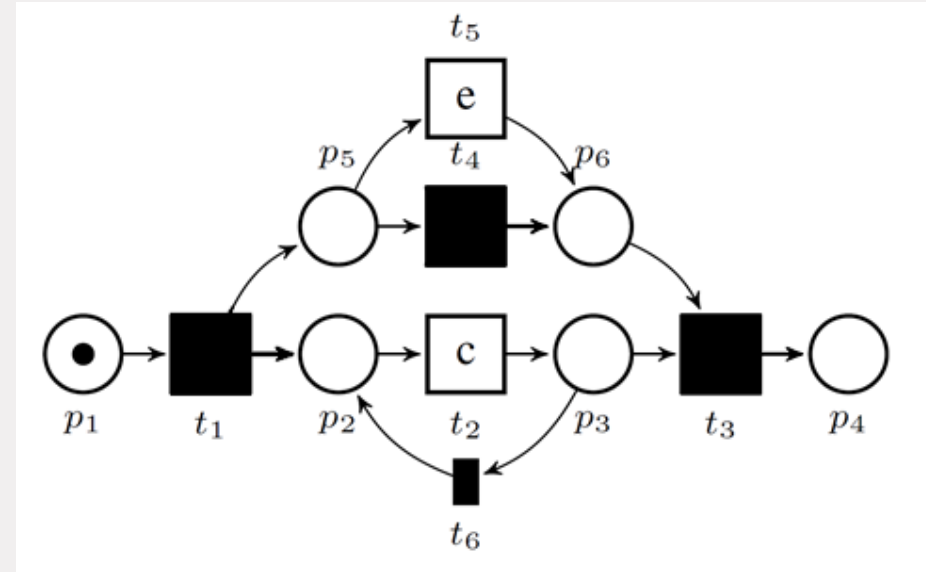
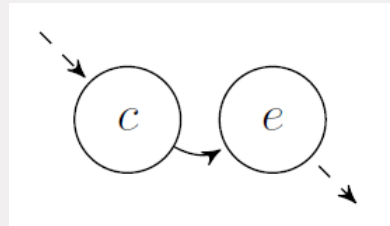
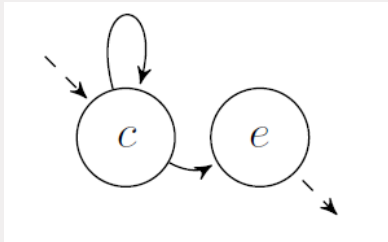
Activity trace	#
<C>	4
<C,C>	5
<C,C,C>	1
<e,c,c,c,c>	1
<e,c>	6
<c,e,c,c,c>	1



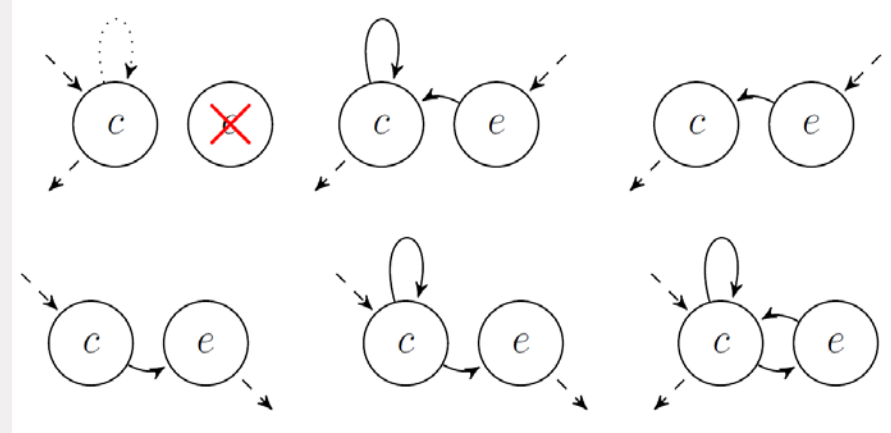
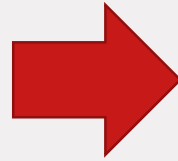
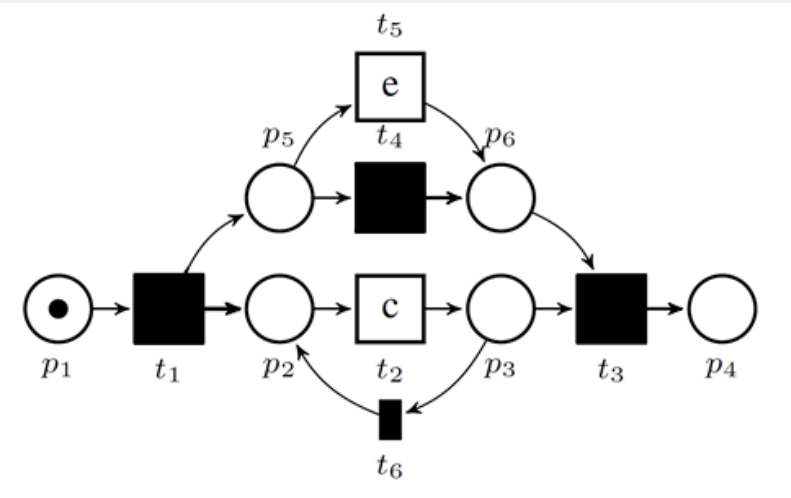
Binary footprint pattern	Frequency
	10
	1
	6
	1

Extracting Binary Patterns from Process Model

1. Project model on a pair of activities, e.g. Activity 'c' and 'e'
2. Verify individually if each of the patterns is executable, by using minimal traces
E.g., $\langle c,c,e \rangle$ relates to the left-most pattern whereas, $\langle c,e \rangle$ relates to the right-most pattern:



Extracting Binary Patterns from Process Model



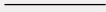
- From T to B, L to R: $\langle c \rangle, \langle e, c, c \rangle, \langle e, c \rangle, \langle c, e \rangle, \langle c, c, e \rangle, \langle c, c, e, c \rangle$

Patterns

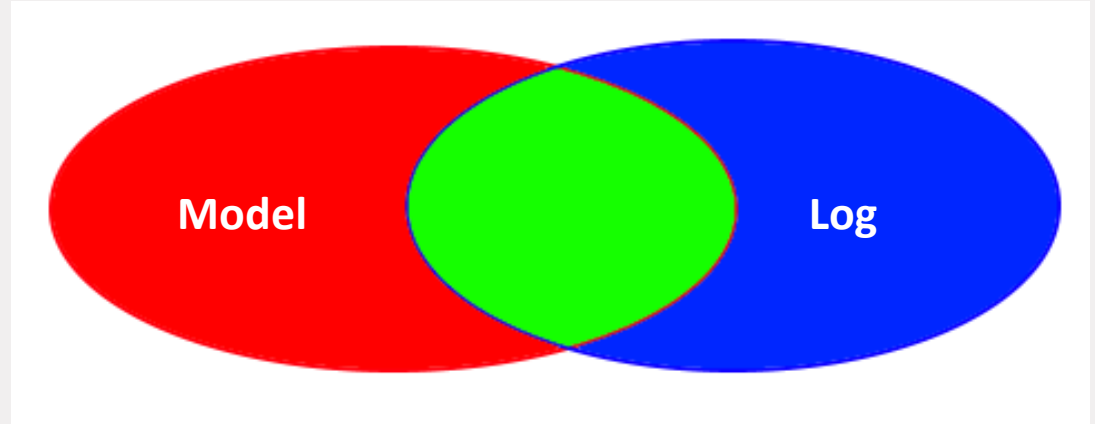
- Unary patterns
- Binary patterns
- Fitness and precision

Compare Patterns to Compute Fitness and Precision

Fitness : Amount of behavior from the Log, also displayed by the model



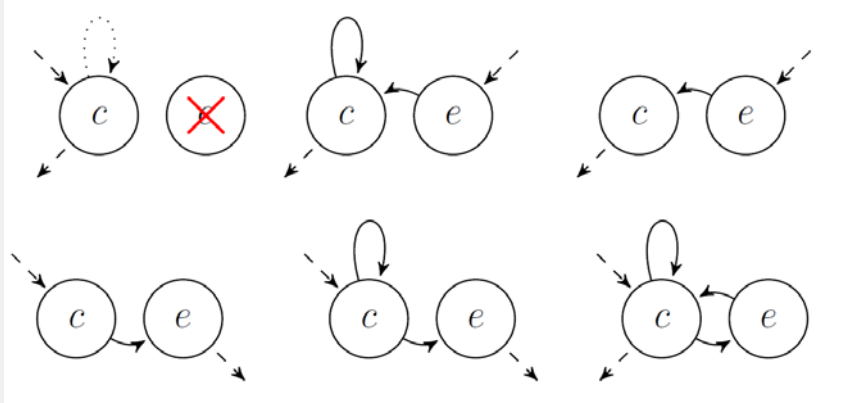
Precision : Amount of extra behavior allowed by the process model



Computed by comparing unary and binary patterns between log and model (0 to 1)

Example fitness and precision

Model



Fitness = 1

Precision = 0.67

Log

Binary footprint pattern	Frequency
	10
	1
	6
	1

Outline

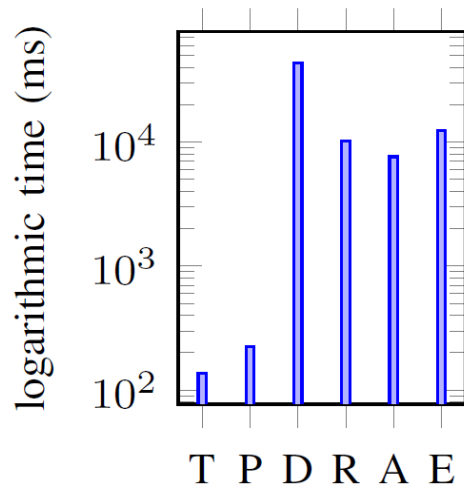
- Process Mining
 - Process discovery and process conformance
- Motivation
 - For interactive process discovery, we need fast process conformance
- Approach
 - Use unary and binary patterns for fast process conformance
- **Evaluation**
- Conclusion

Evaluation

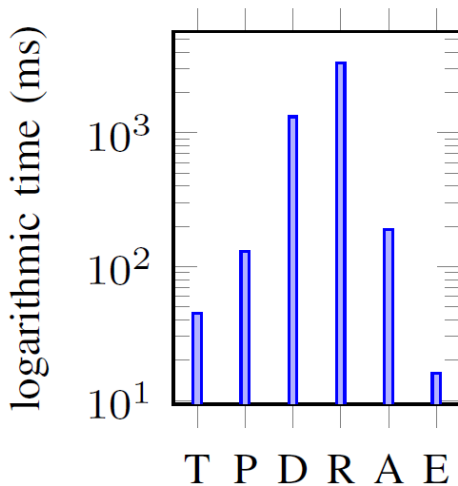
Comparison of our approach with state-of-the-art approaches, using following steps:

1. Use five real-life activity logs
2. For each activity log, discover three types of process models using inductive miner infrequent at the following settings
 - a) Noise threshold = 0
 - b) Noise threshold = 50
 - c) Noise threshold = 100
3. Calculate conformance of each process model w.r.t. the corresponding activity logs using state-of-the-art approaches and our approach

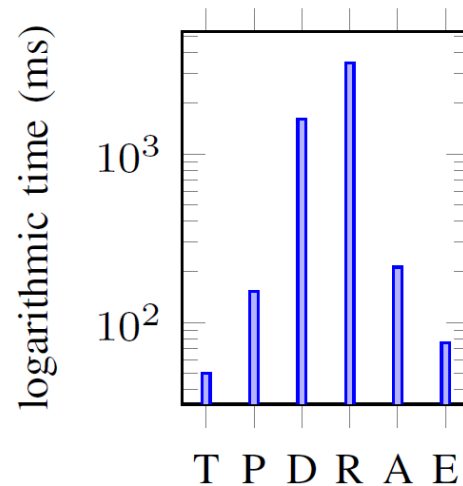
Time



(a) Sepsis.

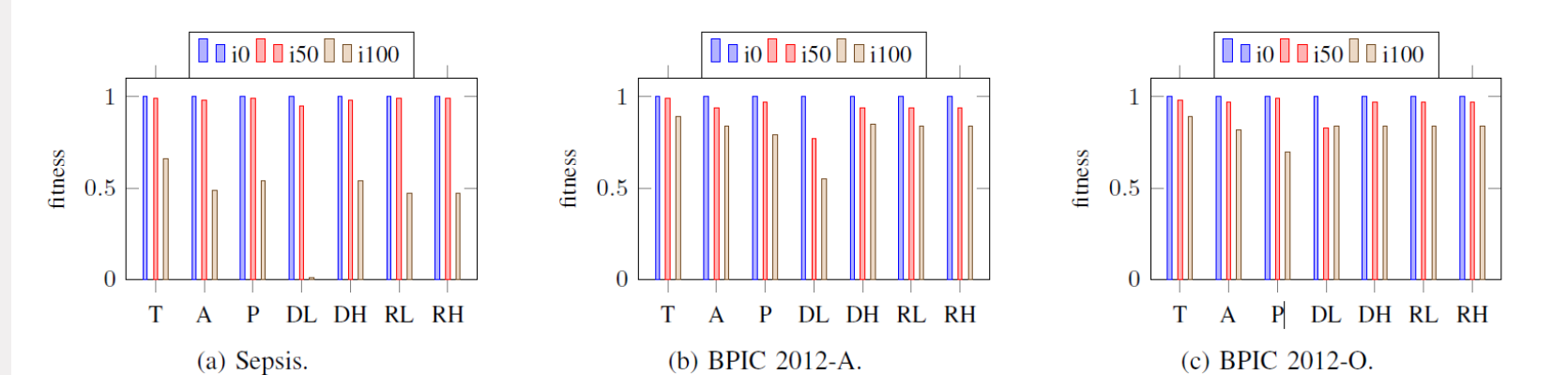


(b) BPIC 2012-A.

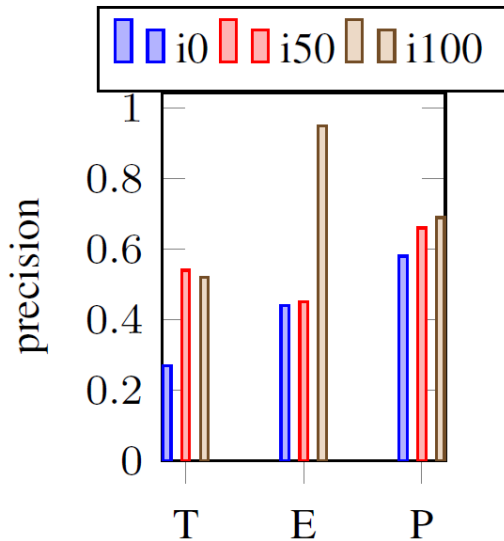


(c) BPIC 2012-O.

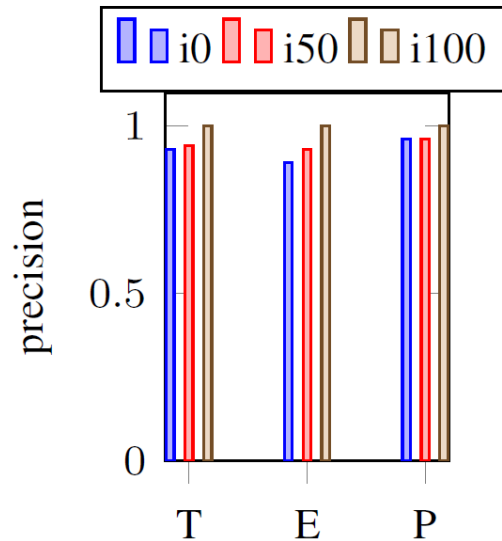
Fitness



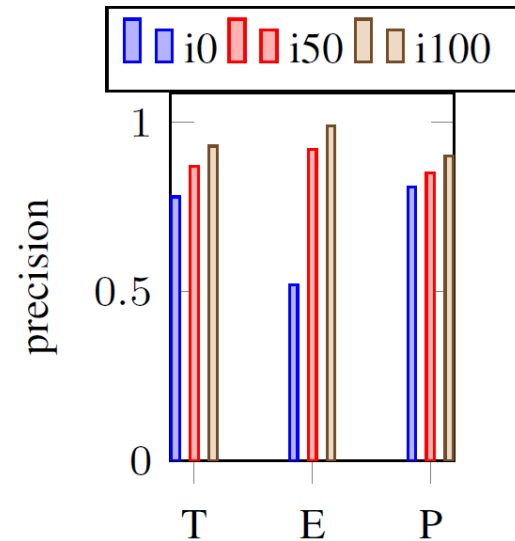
Precision



(a) Sepsis.



(b) BPIC 2012-A.



(c) BPIC 2012-O.

Outline

- Process Mining
 - Process discovery and process conformance
- Motivation
 - For interactive process discovery, we need fast process conformance
- Approach
 - Use unary and binary patterns for fast process conformance
- Evaluation
 - Fast enough with reasonable accuracy on fitness and precision
- Conclusion

Conclusion

- By re-using pre-calculated patterns of the activity log, we improve the performance times of calculating the conformance between a process model and the activity log
- The approximated conformance results calculated using our technique are comparable to many state-of-the-art techniques, and hence provide a reliable estimate of actual conformance scores
- Therefore, the proposed technique is suited in the context of incremental process discovery settings

Thank you!

Questions?