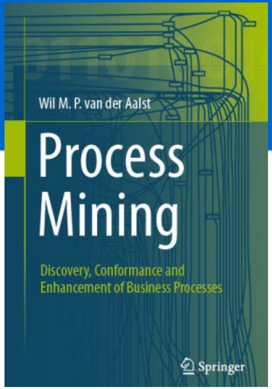
Process Mining: Data Science in Action

Alternative Process Discovery Techniques

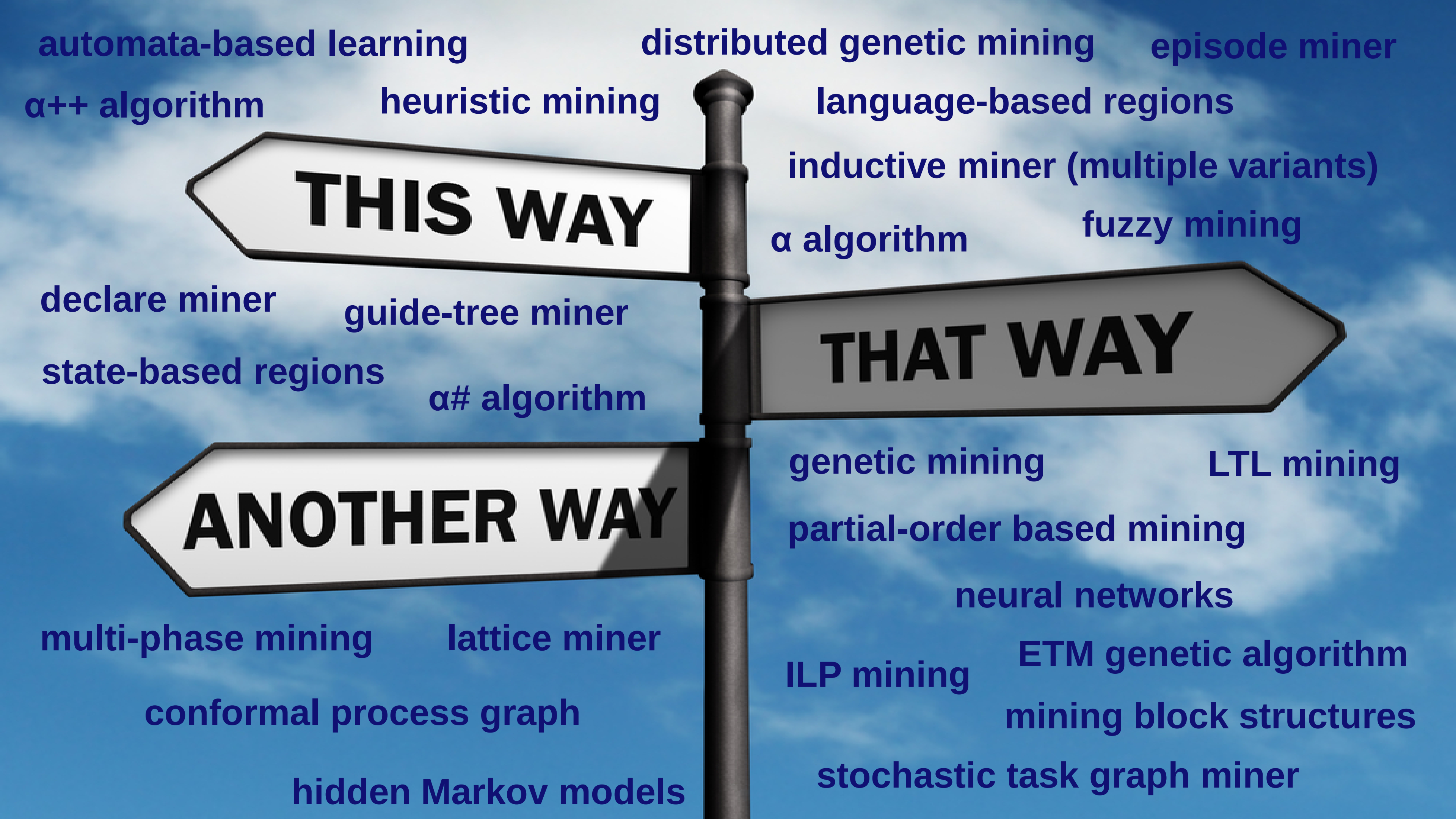
prof.dr.ir. Wil van der Aalst

www.processmining.org

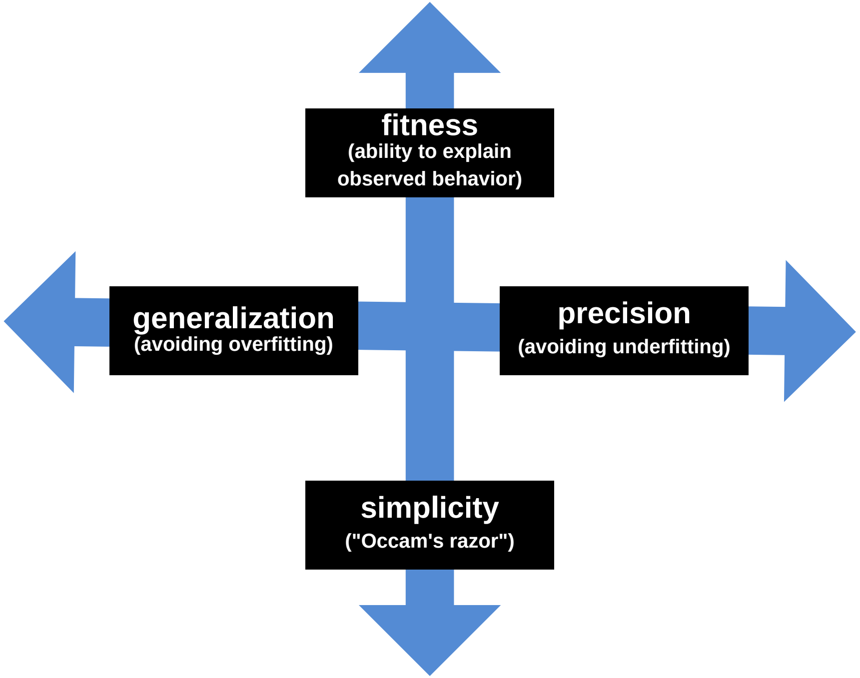


Technische Universiteit Eindhoven University of Technology

Where innovation starts


automata-based learning
distributed genetic mining
episode miner
α++ algorithm
heuristic mining
language-based regions
THIS WAY
inductive miner (multiple variants)
α algorithm
fuzzy mining
declare miner
guide-tree miner
THAT WAY
state-based regions
α# algorithm
ANOTHER WAY
genetic mining
LTL mining
partial-order based mining
neural networks
multi-phase mining
lattice miner
ETM genetic algorithm
ILP mining
conformal process graph
mining block structures
stochastic task graph miner
hidden Markov models

Considerations

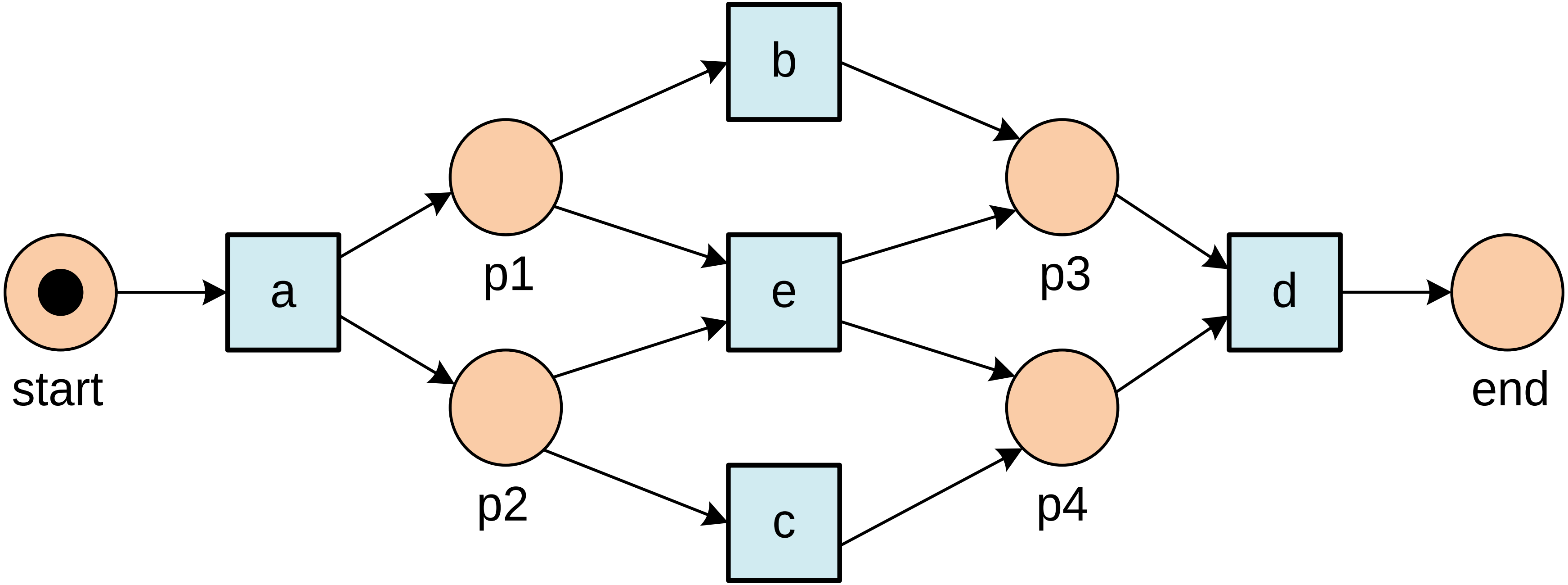


implementation versus approach

process discovery ...

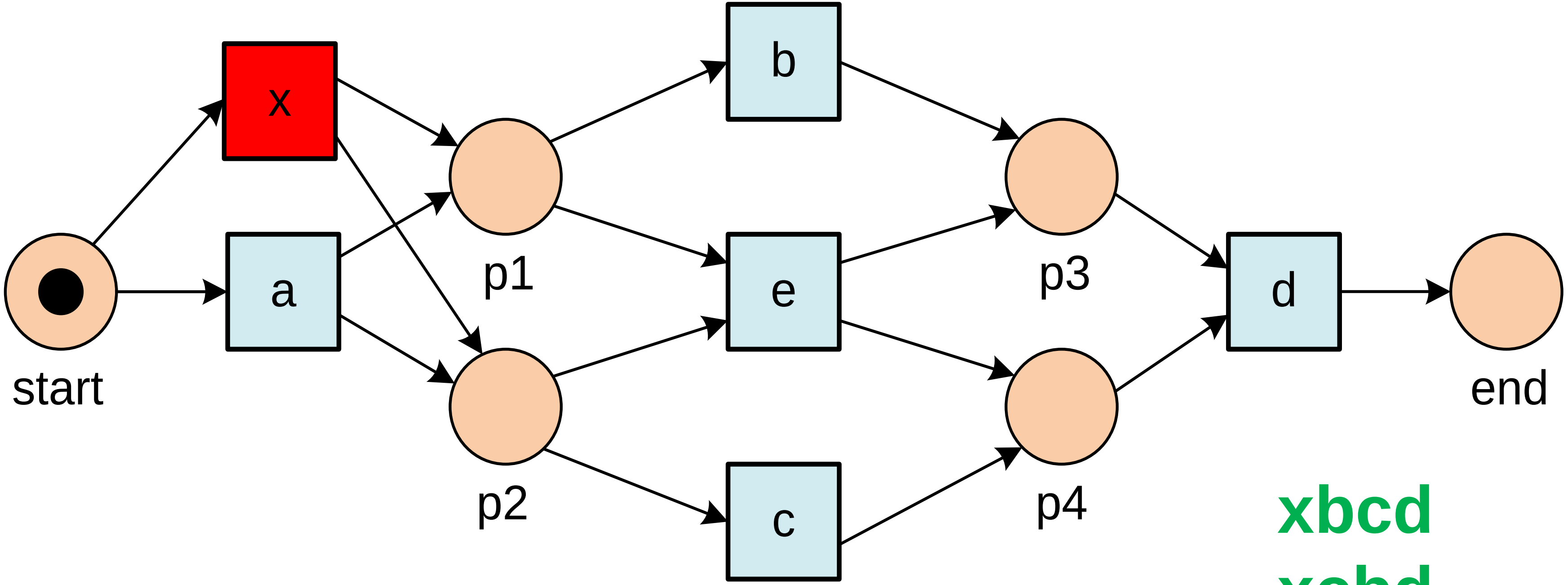
let's take a step back

Question: How to add behavior?



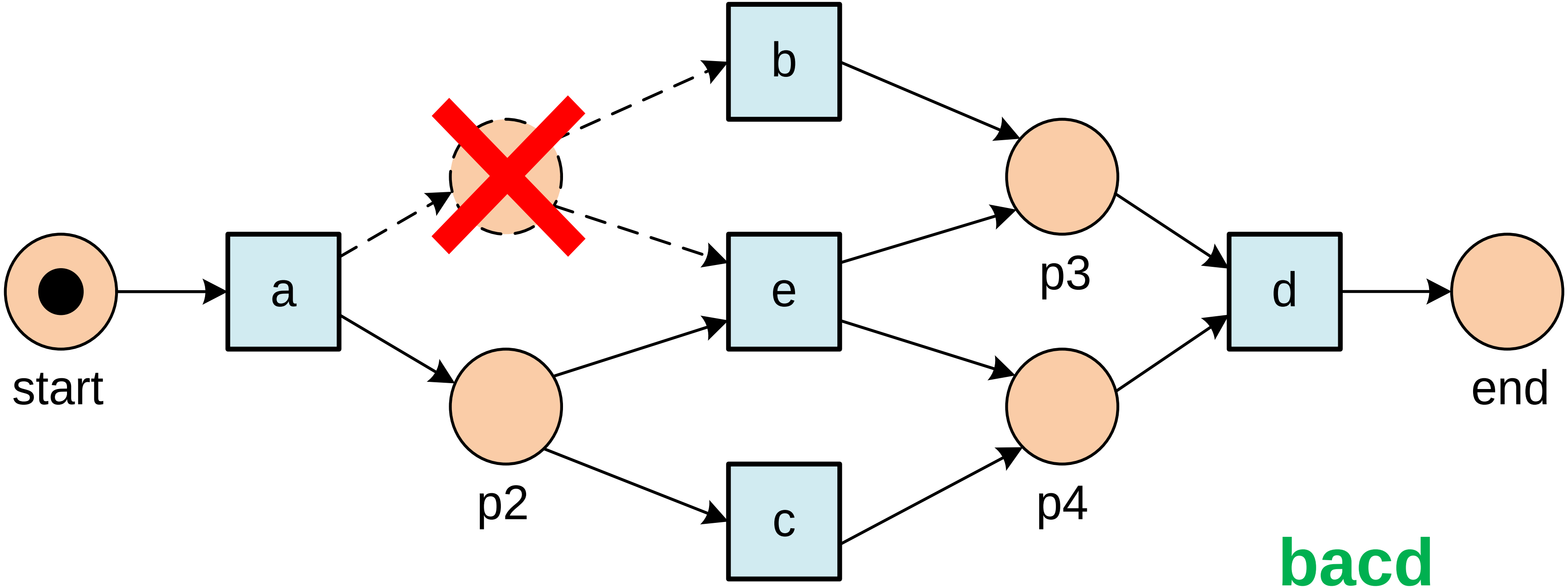
How to modify the Petri net such that it allows for more behavior (more traces, ignore termination)?

Answer (1/3): Adding transitions



xbcd
xcbd
xed

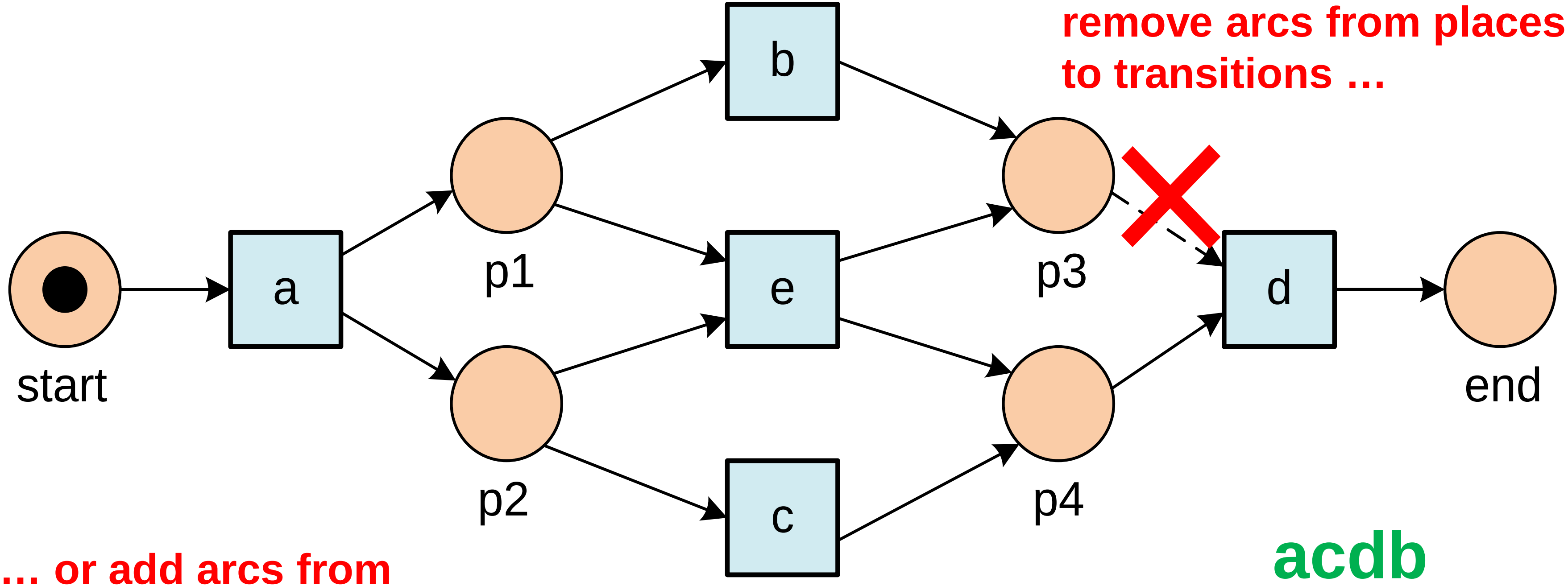
Answer (2/3): Removing places



bacd

...

Answer (3/3): Removing/adding arcs



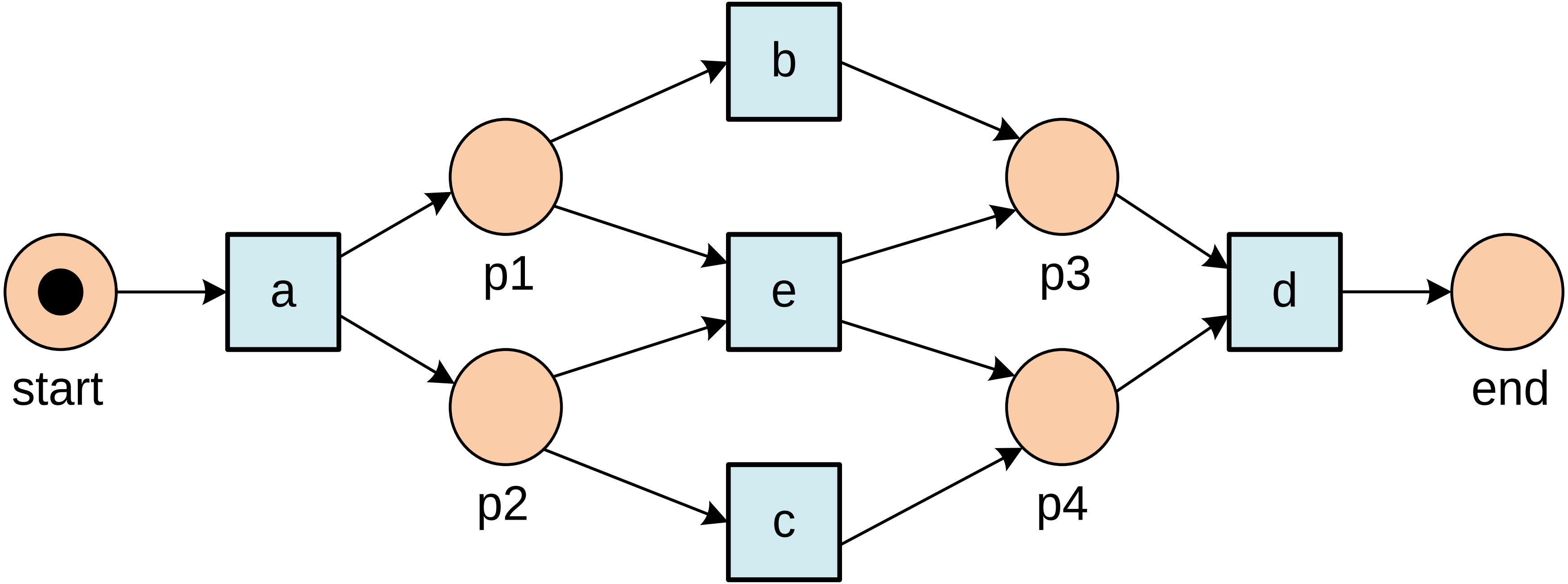
remove arcs from places to transitions …

… or add arcs from transitions to places

acdb

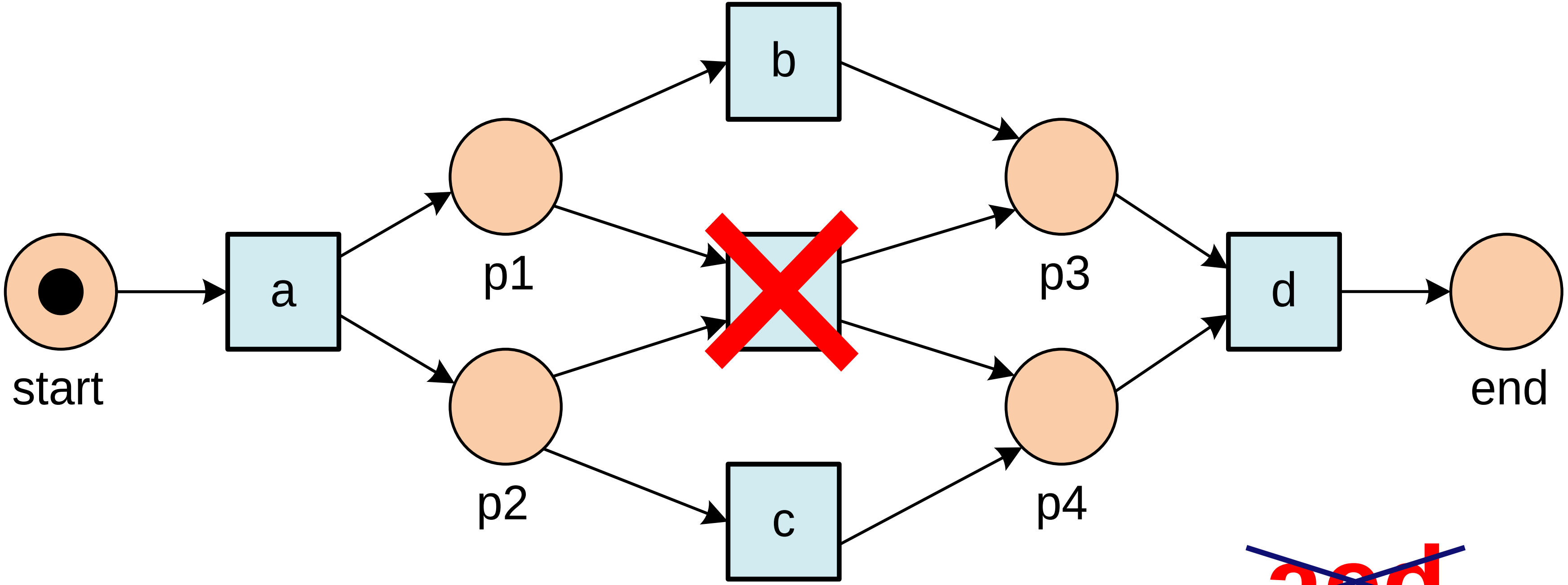
…

Question: How to remove behavior?



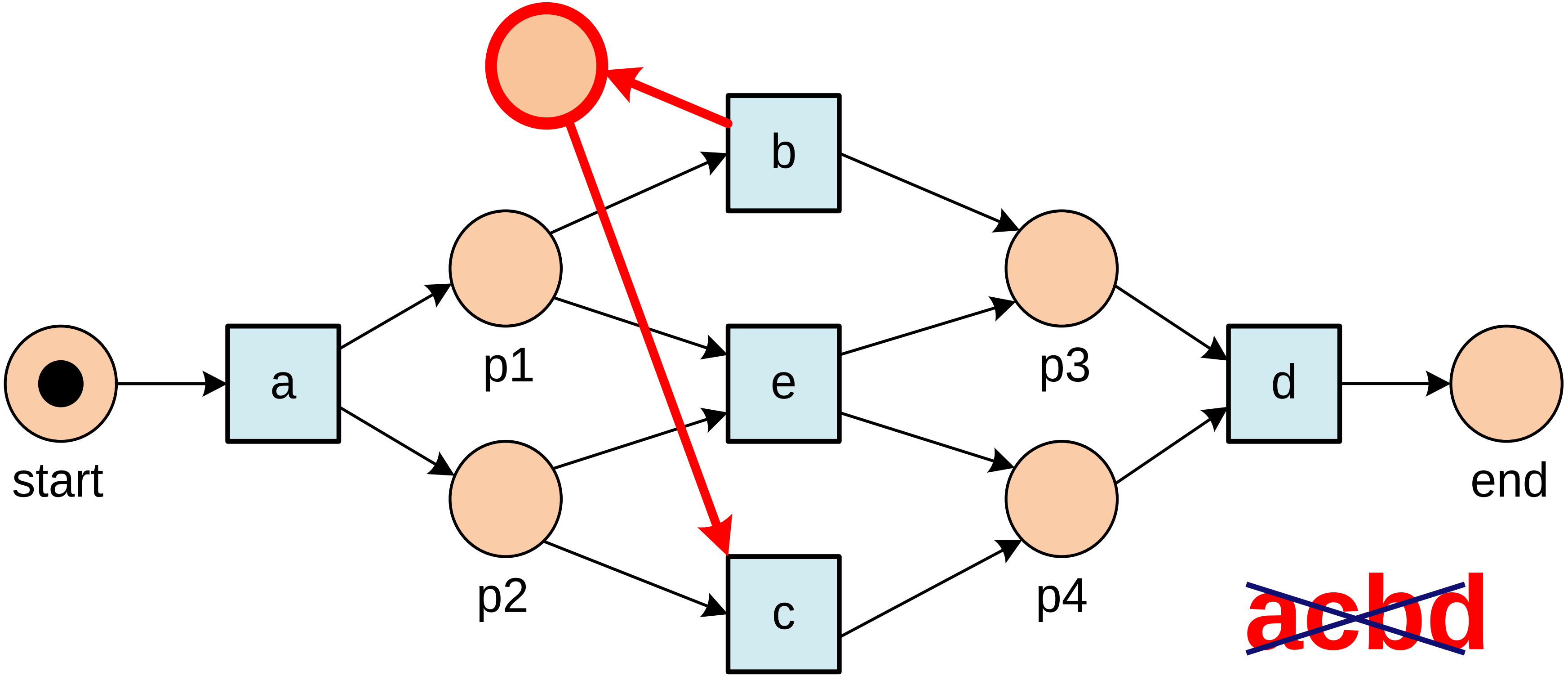
How to modify the Petri net such that it allows for less behavior (ignore termination)?

Answer (1/3): Removing transitions



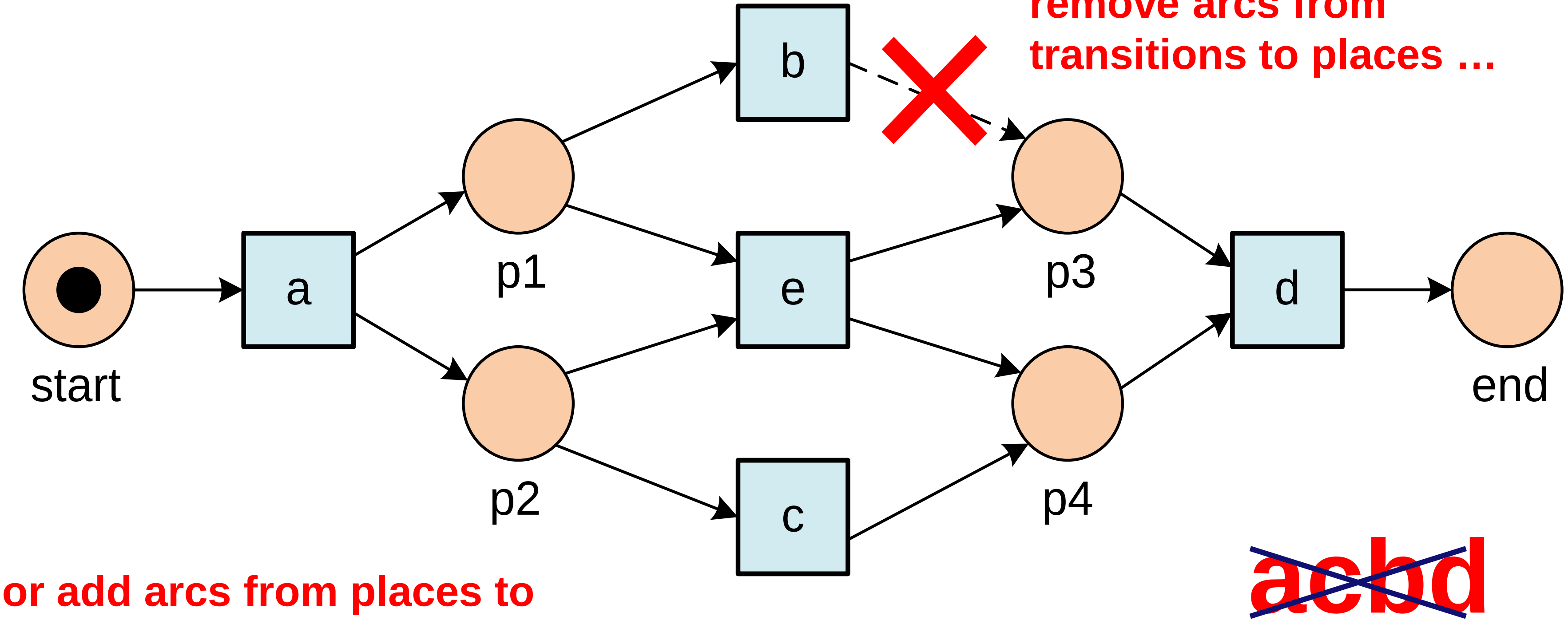
~~aed~~

Answer (2/3): Adding places



TU/e

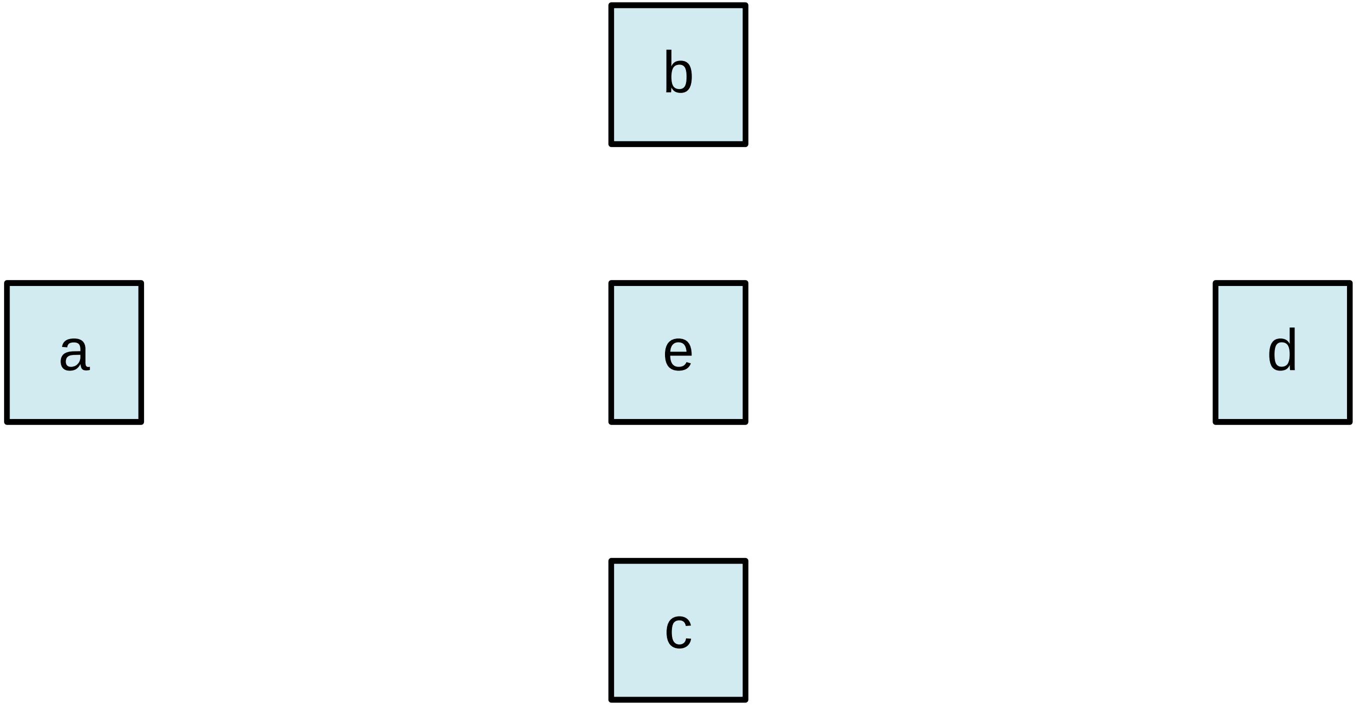
Answer (3/3): Removing/adding arcs



remove arcs from transitions to places …

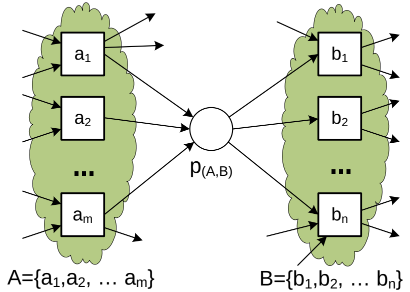
or add arcs from places to transitions …

Process discovery = finding places

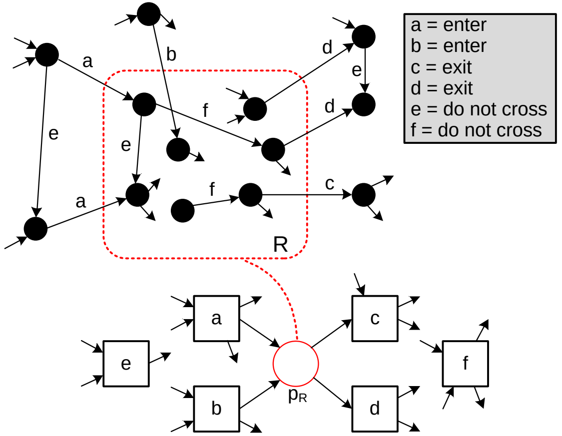


Examples of discovery techniques

- **Alpha algorithm**



- **State-based regions**



- **Next: language-based regions**

$$c \cdot \mathbf{1} + A' \cdot \mathbf{x} - A \cdot \mathbf{y} \geq \mathbf{0}$$

language-based regions

Looks complex, but the idea is simple ...

$$c \cdot \mathbf{1} + A' \cdot \mathbf{x} - A \cdot \mathbf{y} \geq \mathbf{0}$$

just says that places will never "go negative"

any solution *(x,y,c)* is a region

any region *(x,y,c)* is a feasible place

Defining *A*

$$c \cdot \mathbf{1} + A' \cdot \mathbf{x} - A \cdot \mathbf{y} \geq \mathbf{0}$$

⟨a,b⟩^{30}
⟨b,a⟩^{30}

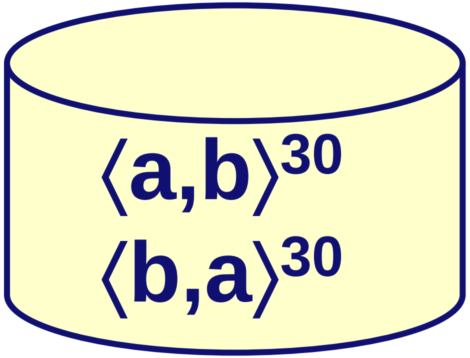
$$\mathcal{L} = \{\epsilon, \langle a \rangle, \langle b \rangle, \langle a, b \rangle, \langle b, a \rangle\}$$

$$A = \begin{array}{c} \\ \langle a \rangle \\ \langle b \rangle \\ \langle a, b \rangle \\ \langle b, a \rangle \end{array} \begin{array}{c} \begin{array}{cc} a & b \end{array} \\ \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 1 \\ 1 & 1 \end{pmatrix} \end{array}$$

TU/e

Defining A'

$$c \cdot \mathbf{1} + A' \cdot \mathbf{x} - A \cdot \mathbf{y} \geq \mathbf{0}$$



$$\mathcal{L} = \{\epsilon, \langle a \rangle, \langle b \rangle, \langle a, b \rangle, \langle b, a \rangle\}$$

$$A' = \begin{array}{r c} & \begin{array}{cc} a & b \end{array} \\ \begin{array}{r} \langle a \rangle \\ \langle b \rangle \\ \langle a, b \rangle \\ \langle b, a \rangle \end{array} & \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{pmatrix} \end{array}$$

Inequation system

$$c \cdot \mathbf{1} + A' \cdot \mathbf{x} - A \cdot \mathbf{y} \geq \mathbf{0}$$

$\langle a,b \rangle^{30}$
$\langle b,a \rangle^{30}$

A' A

$$c \cdot \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x_a \\ x_b \end{pmatrix} - \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 1 \\ 1 & 1 \end{pmatrix} \cdot \begin{pmatrix} y_a \\ y_b \end{pmatrix} \geq \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

c is ... nu... tok... place

x_a is the numb... arcs from transi... to the plac...

x_b ... number of ... nsition ... ace

y_a is the number of arcs from the place to transition a

y_b is the number of arcs from the place to transition b

Inequation system

$$c \cdot \mathbf{1} + A' \cdot \mathbf{x} - A \cdot \mathbf{y} \geq \mathbf{0}$$

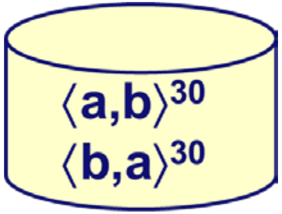
$$c \cdot \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x_a \\ x_b \end{pmatrix} - \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 1 \\ 1 & 1 \end{pmatrix} \cdot \begin{pmatrix} y_a \\ y_b \end{pmatrix} \geq \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$$c - y_a \geq 0$$

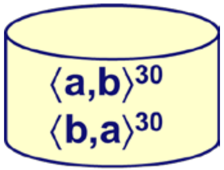
$$c - y_b \geq 0$$

$$c + x_a - y_a - y_b \geq 0$$

$$c + x_b - y_a - y_b \geq 0$$



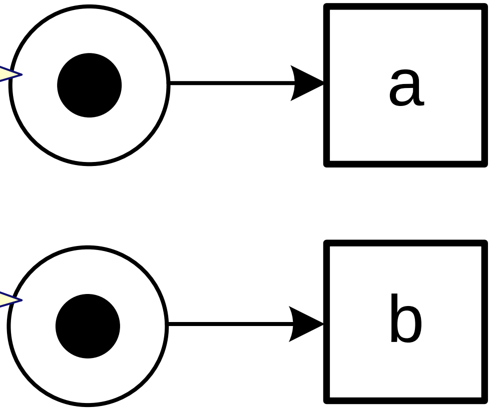
Example solutions



$$c - y_a \geq 0$$
$$c - y_b \geq 0$$
$$c + x_a - y_a - y_b \geq 0$$
$$c + x_b - y_a - y_b \geq 0$$

$c = 1$
$x_a = 0$

$c = 1$
$x_a = 0$
$x_b = 0$
$y_a = 0$
$y_b = 1$

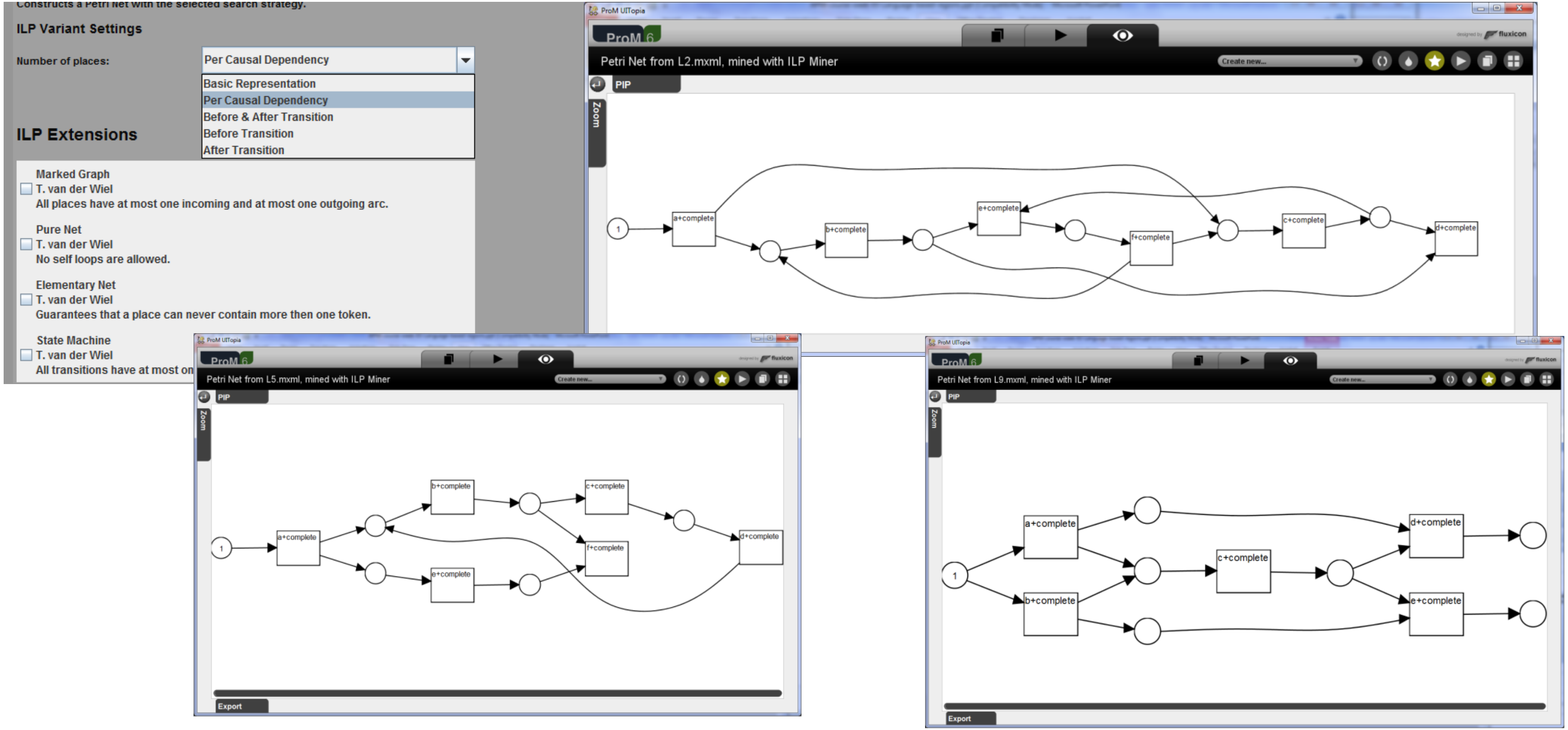


Language-based regions

$$c \cdot \mathbf{1} + A' \cdot \mathbf{x} - A \cdot \mathbf{y} \geq \mathbf{0}$$

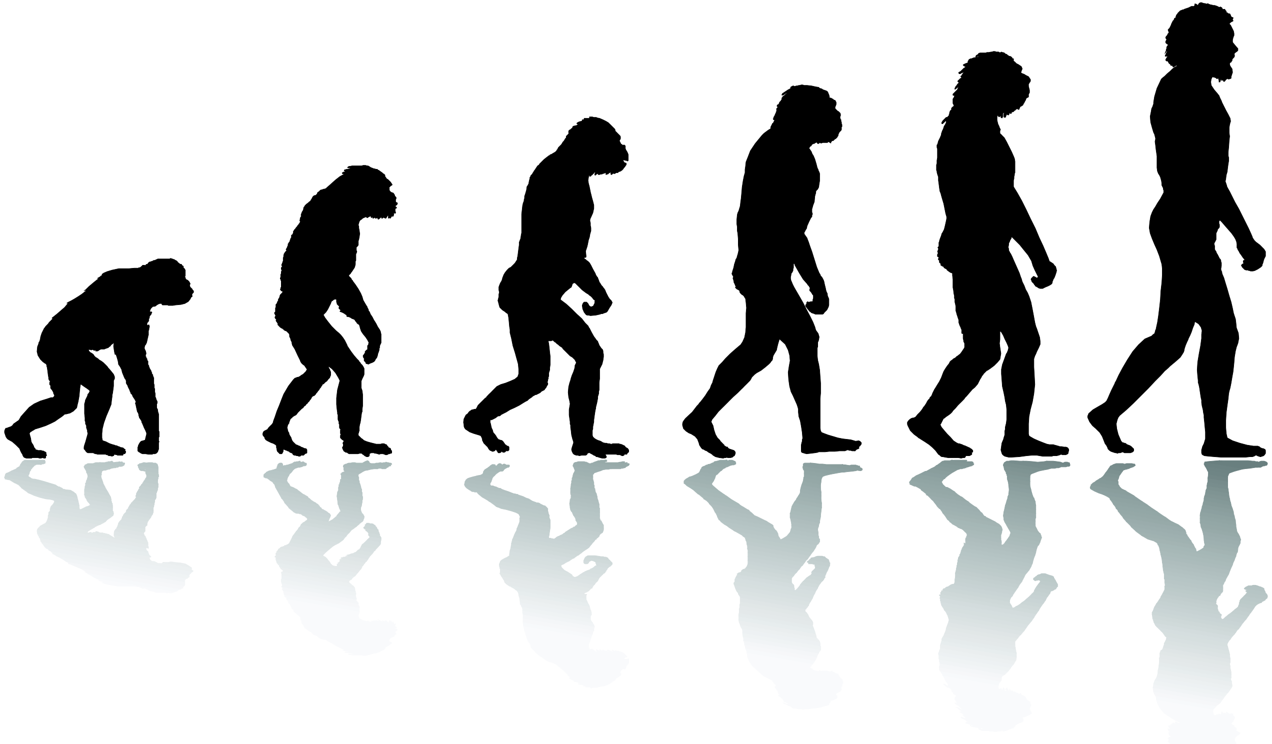
- **Any solution is a feasible place.**
- **Additional constraints can be added easily (empty at end, limited fan-in or fan-out, etc.).**
- **Goal function can be used to select the most interesting places.**
- **Optimization problem (e.g. ILP)!**
  (ILP = Integer Linear Programming)

ProM's ILP miner

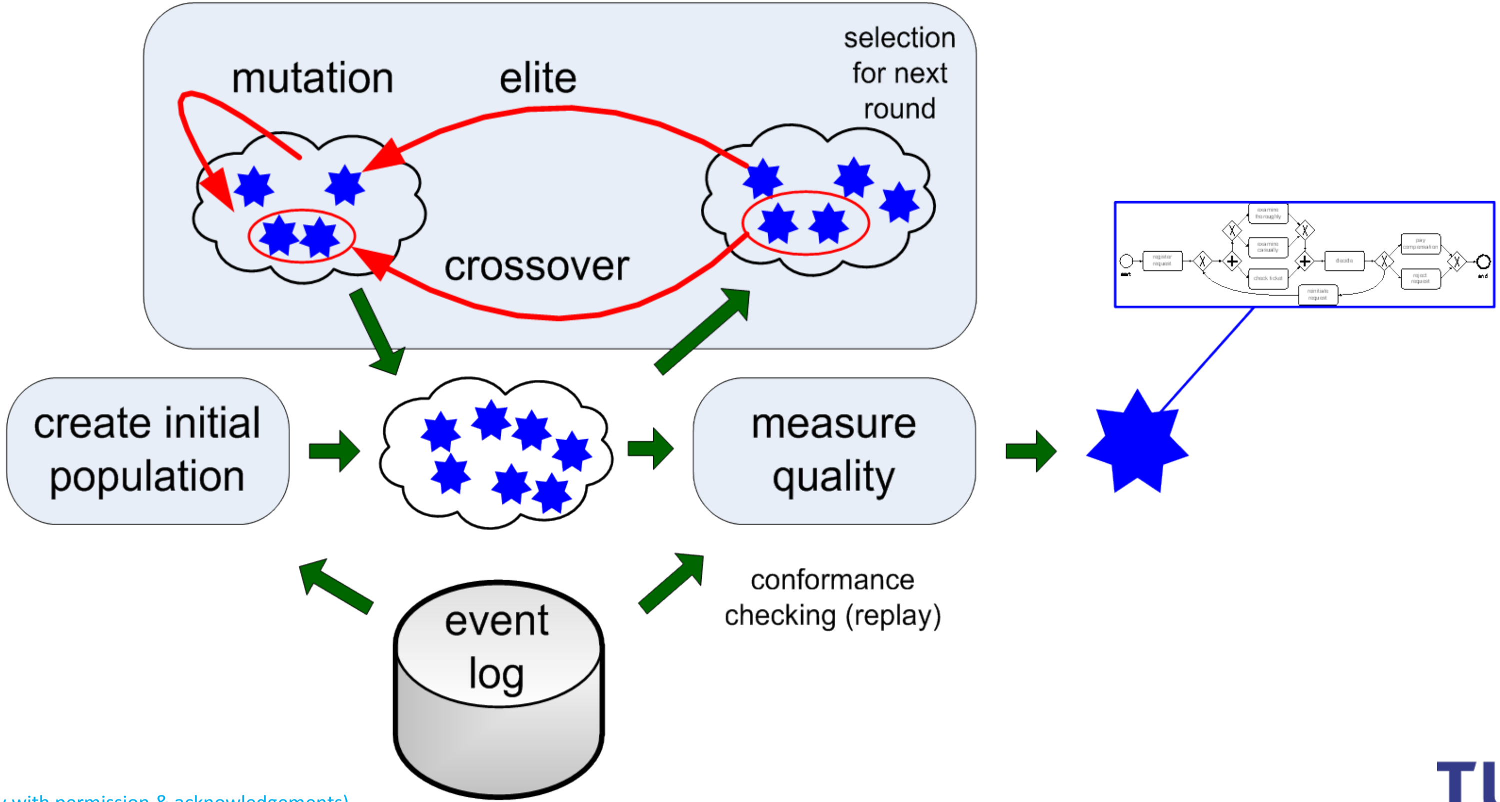


genetic
process
mining

Let evolution do its work ...

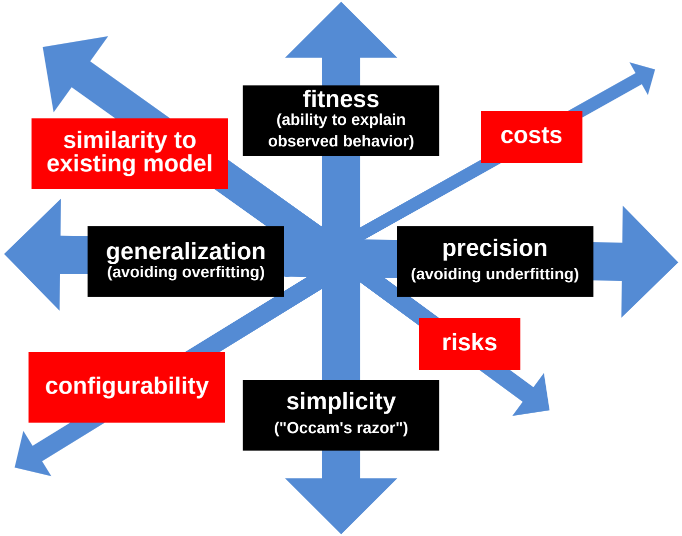


Approach


mutation
elite
selection
for next
round
crossover
create initial
population
measure
quality
event
log
conformance
checking (replay)

Properties of genetic mining

- For larger processes or event logs: very, very slow.
- Very flexible: easy to add new forces (add quality measures).
- Often used when confronted with a new question, followed by more efficient approaches.




inductive
mining

Split event logs based on activity labels

abdef
acdef
adbef
adcef
abdeg
acdeg
adbeg
adceg

Split {a,b,c,d,e,f,g,h} into {a,b,c,d} and {e,f,g} using sequence decomposition

abdef
acdef
adbef
adcef
abdeg
acdeg
adbeg
adceg

Result

abdef
acdef
adbef
adcef
abdeg
acdeg
adbeg
adceg

→

abd
acd
adb
adc
abd
acd
adb
adc

seq

ef
ef
ef
ef
eg
eg
eg
eg

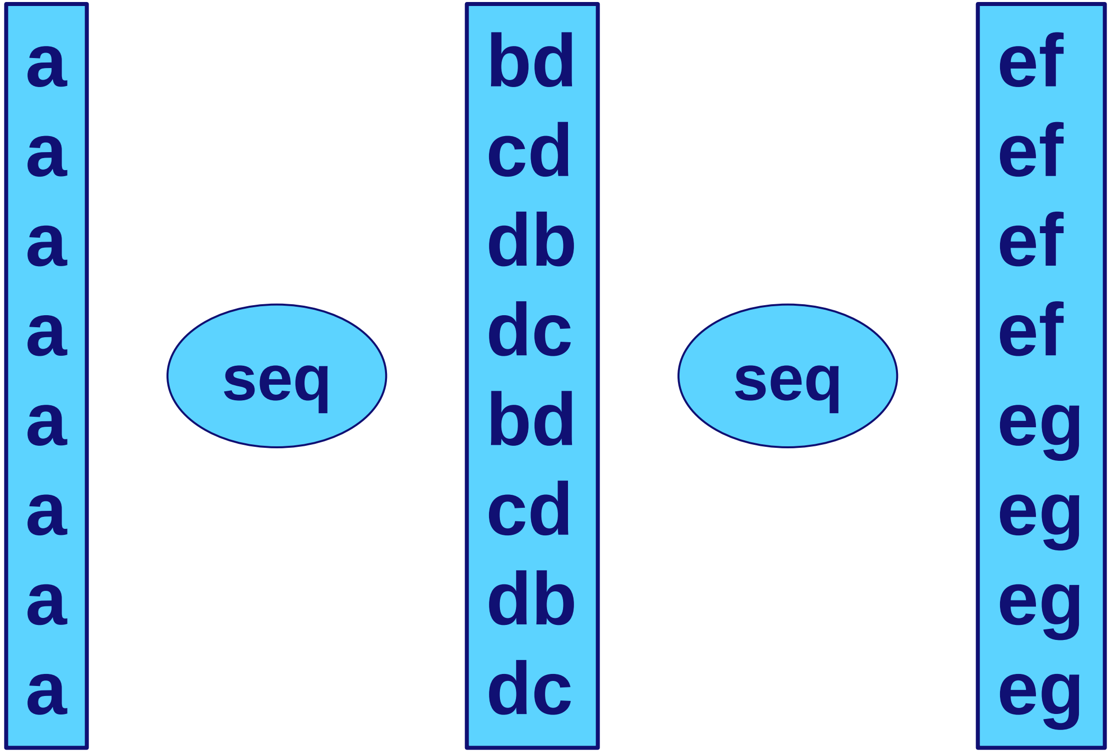
Split {a,b,c,d} into {a} and {b,c,d} using sequence decomposition

abd
acd
adb
adc
abd
acd
adb
adc

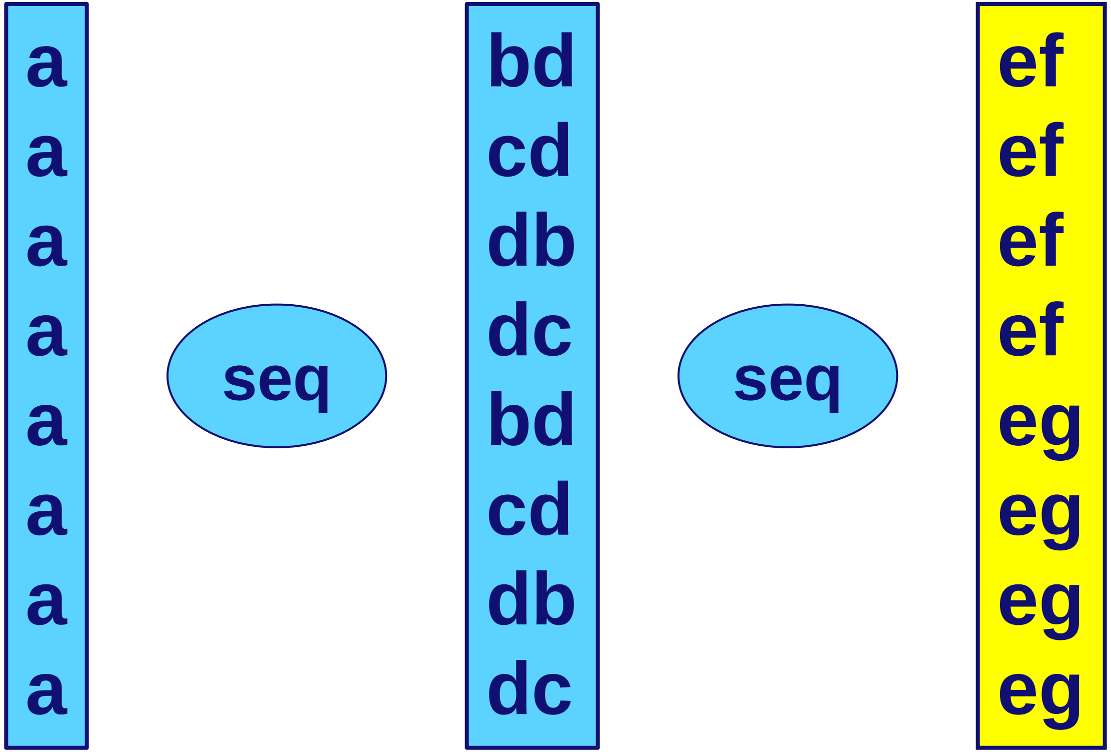
seq

ef
ef
ef
ef
eg
eg
eg
eg

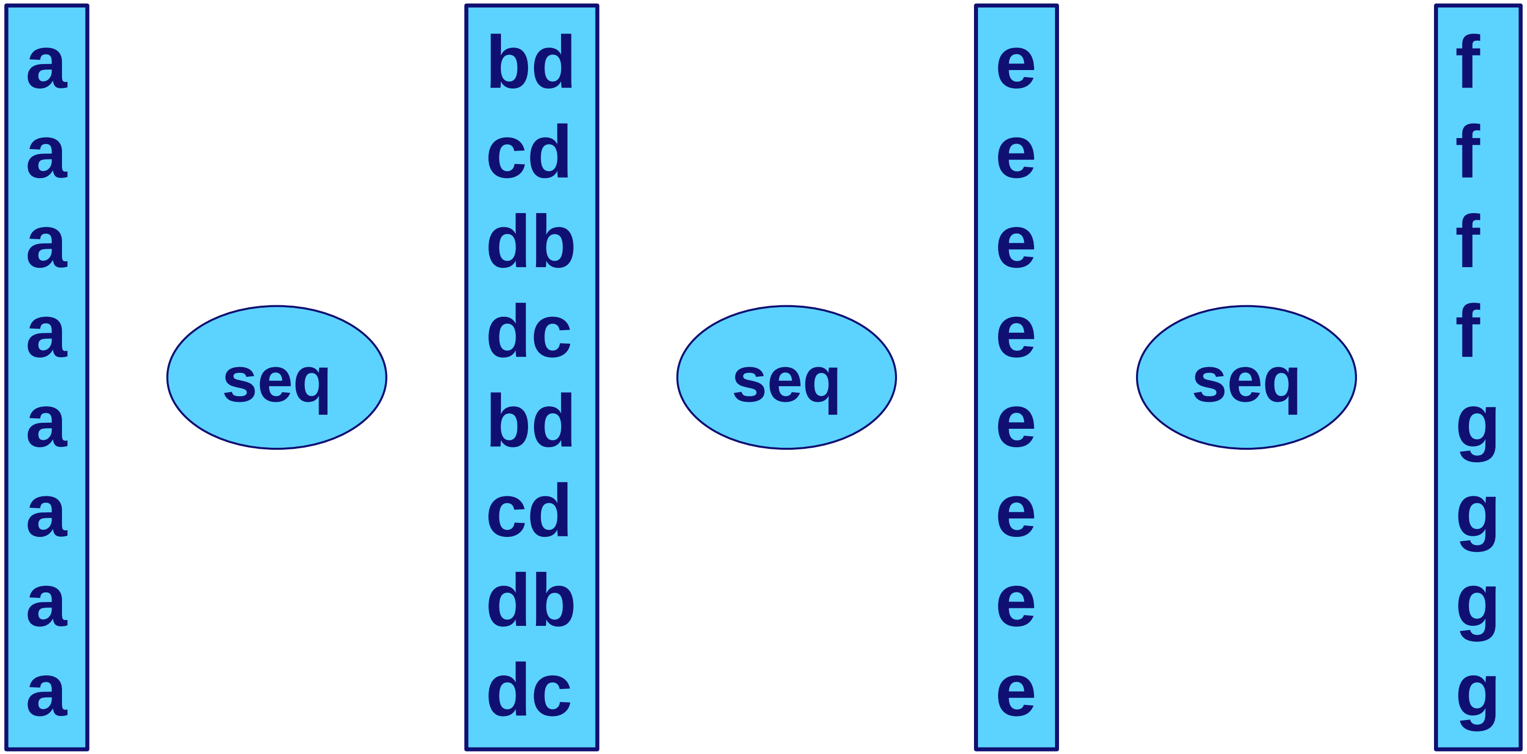
Result



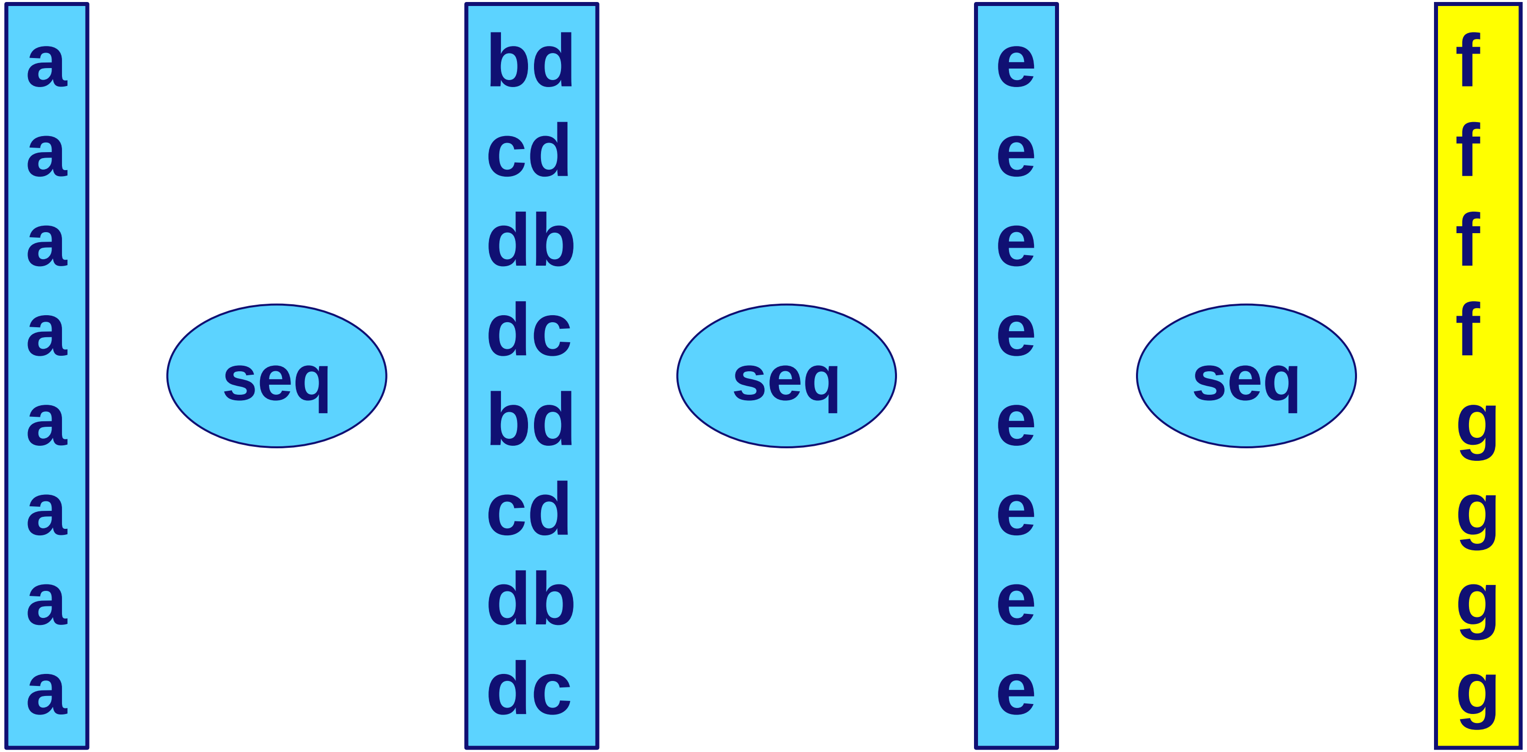
Split {e,f,g} into {e} and {f,g} using sequence decomposition



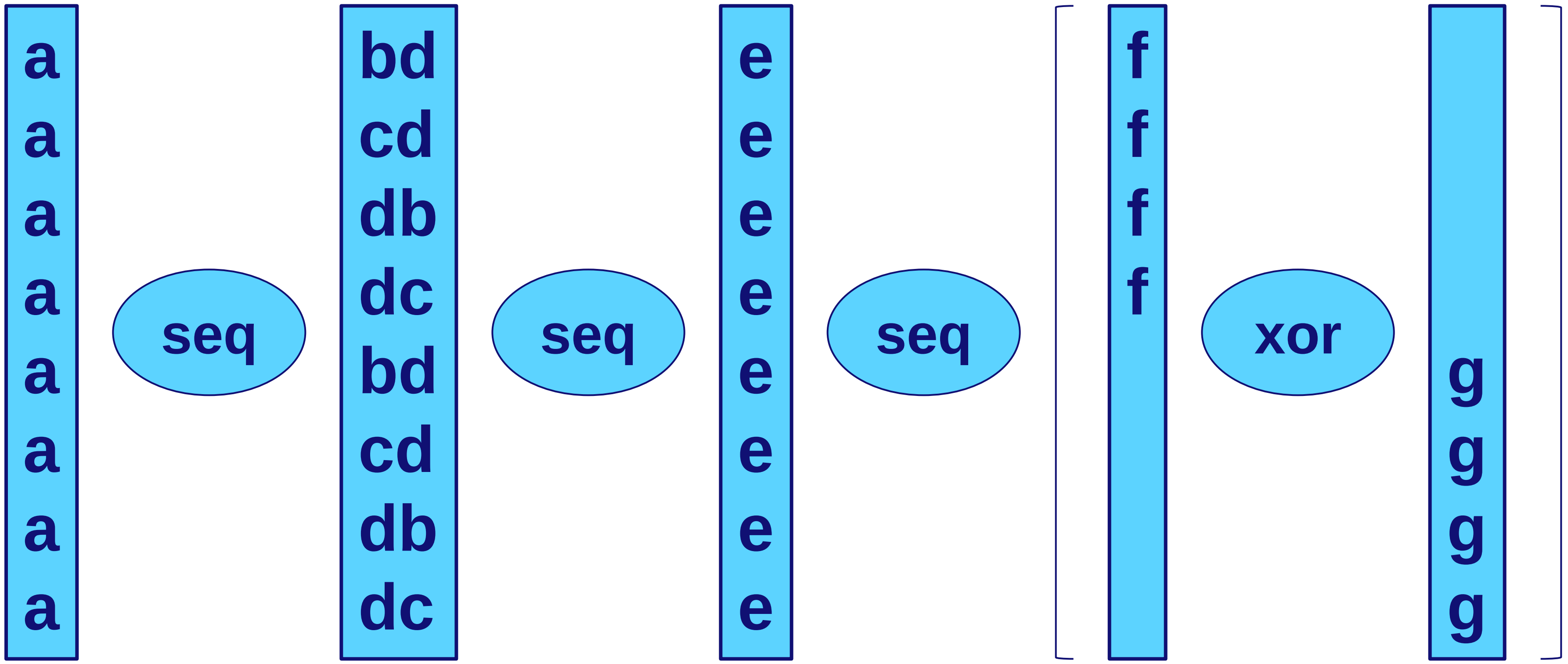
Result



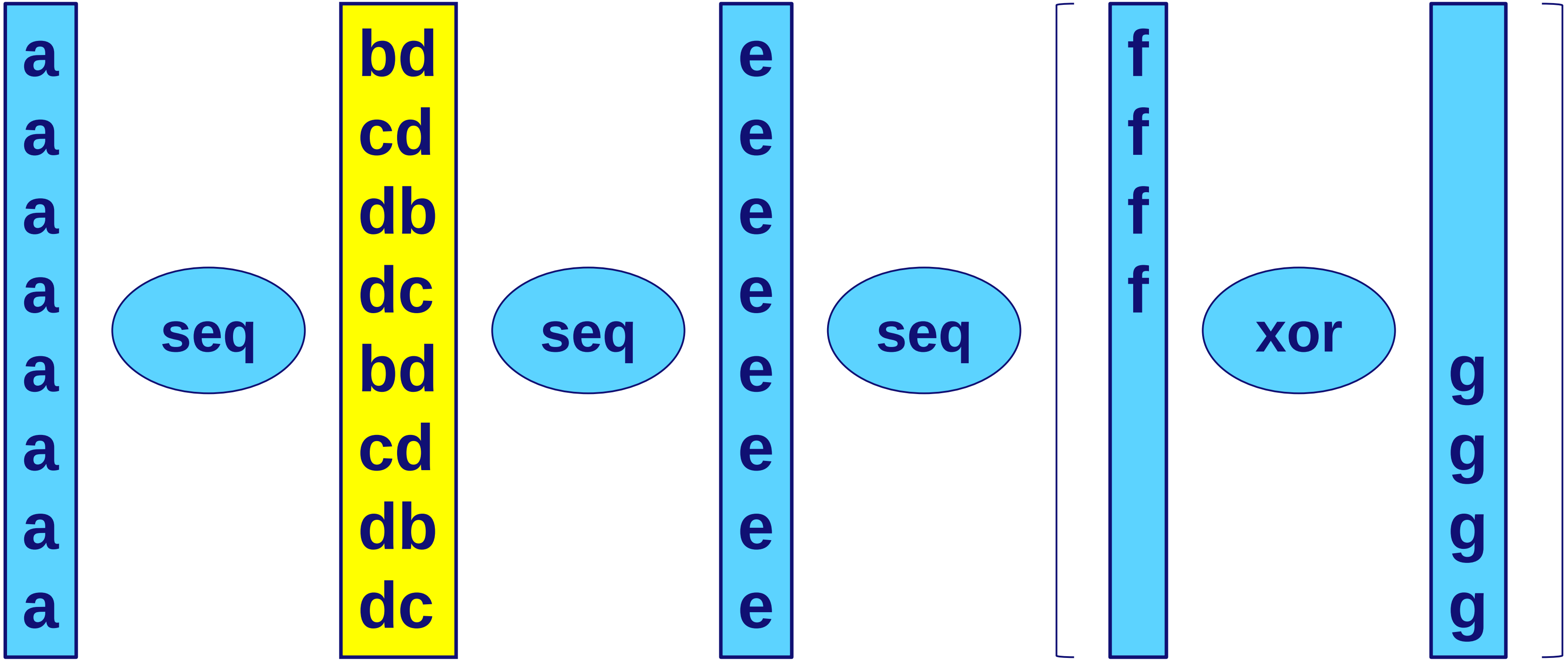
Split {f,g} into {f} and {g} using XOR decomposition



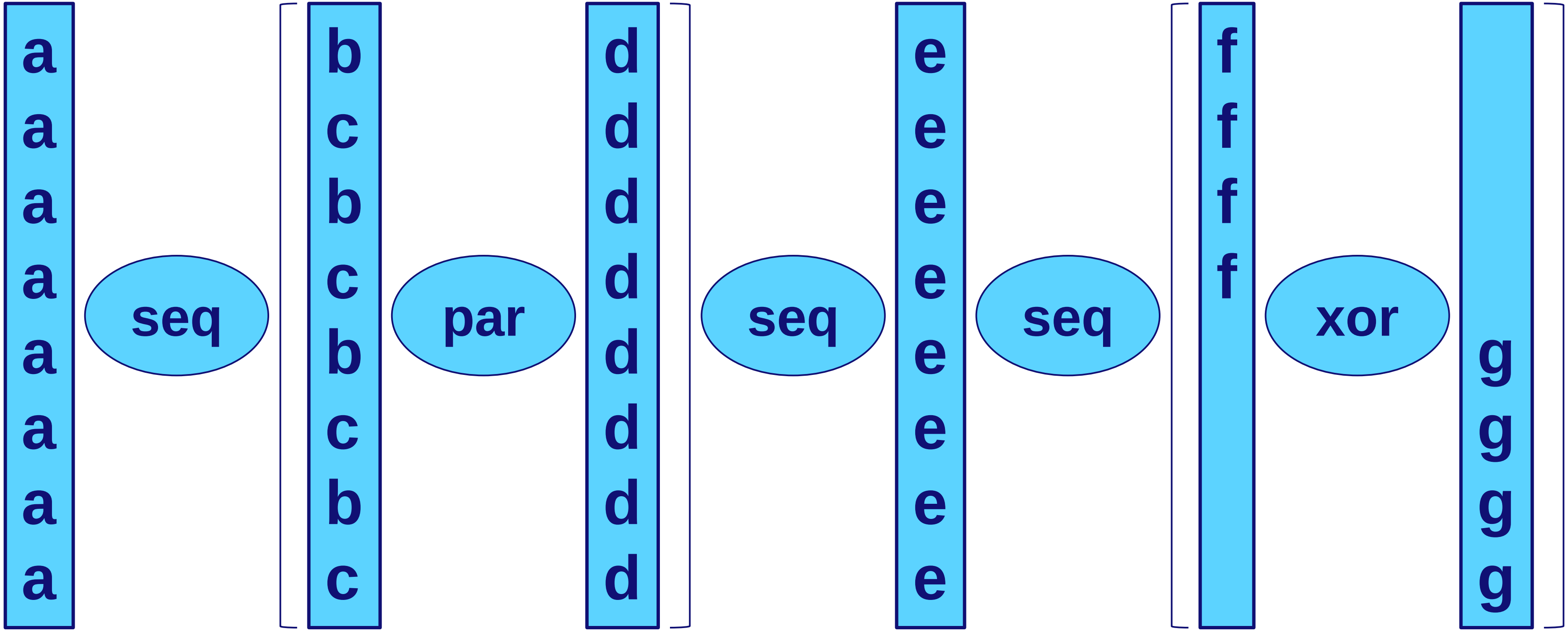
Result


a
a
a
a
a
a
a
a
seq
bd
cd
db
dc
bd
cd
db
dc
seq
e
e
e
e
e
e
e
e
seq
f
f
f
f
xor
g
g
g
g

Split {b,c,d} into {b,c} and {d} using AND decomposition

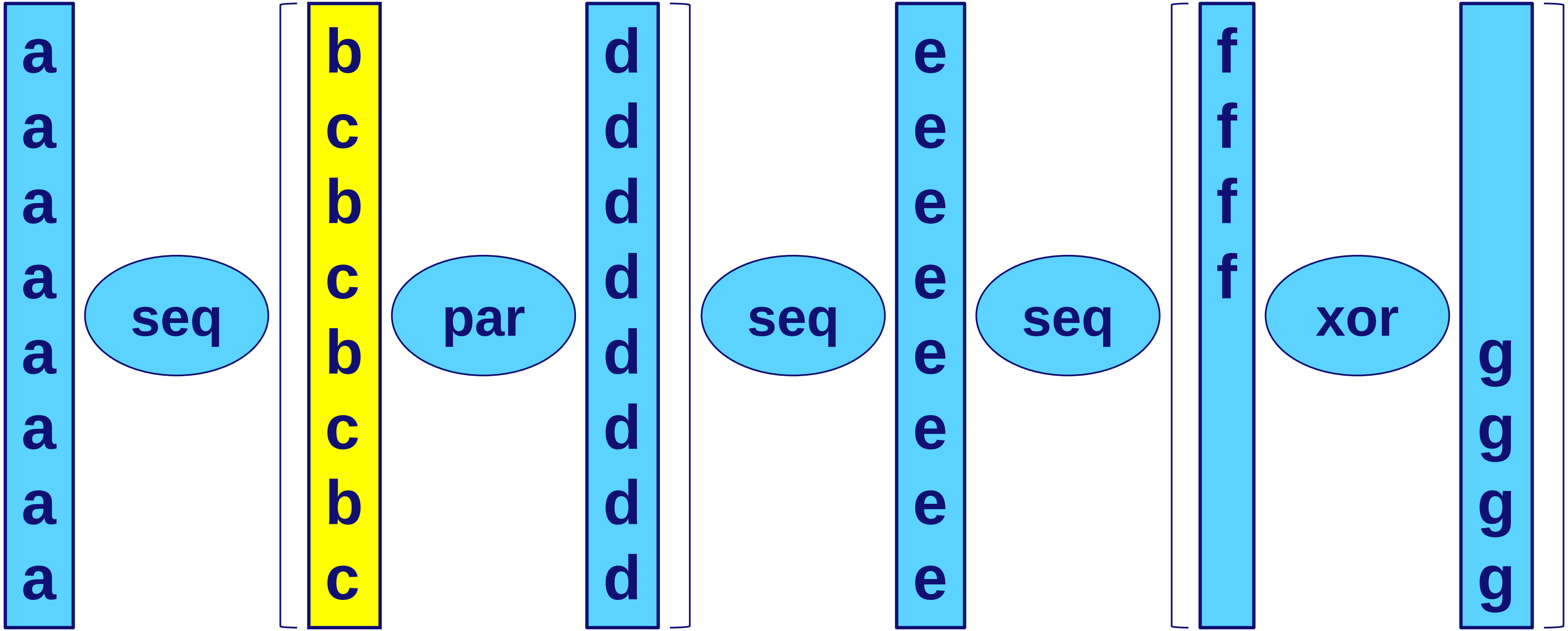


Result

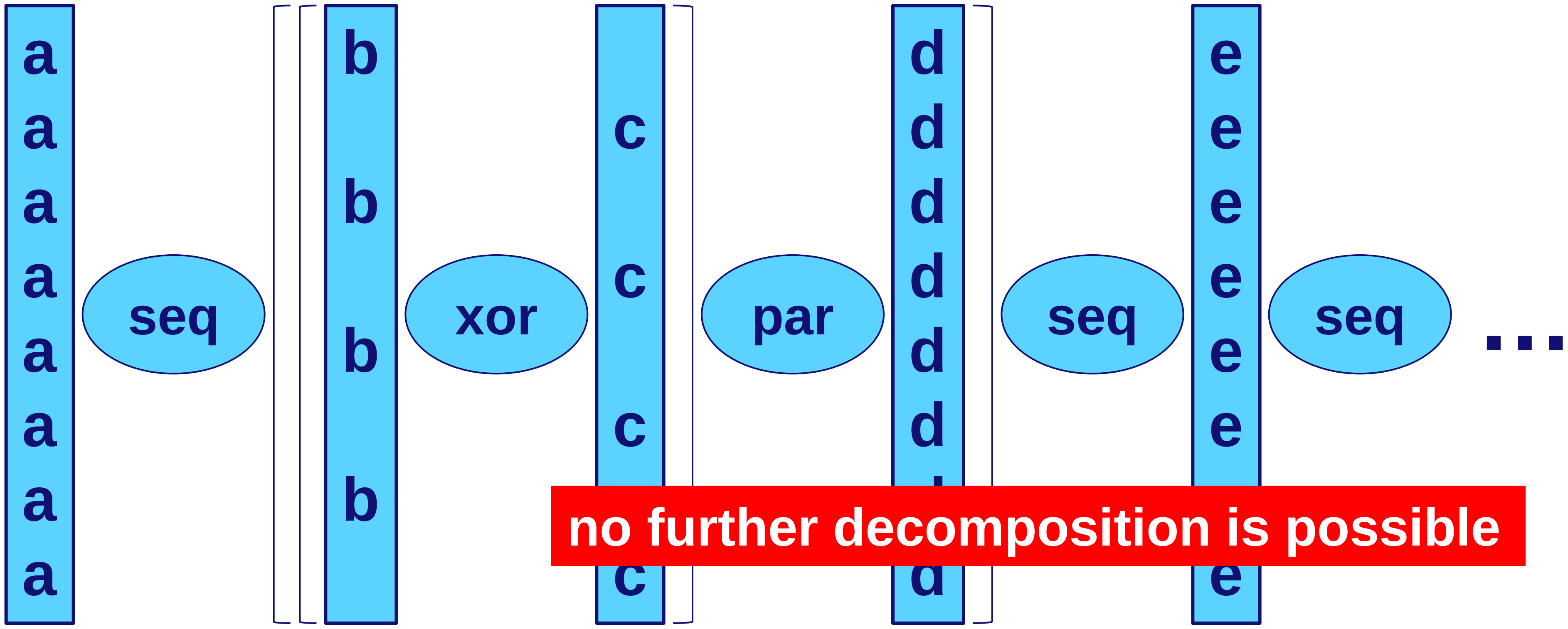


a a a a a a a a
seq
b c b c b c b c
par
d d d d d d d d
seq
e e e e e e e e
seq
f f f f
xor
g g g g

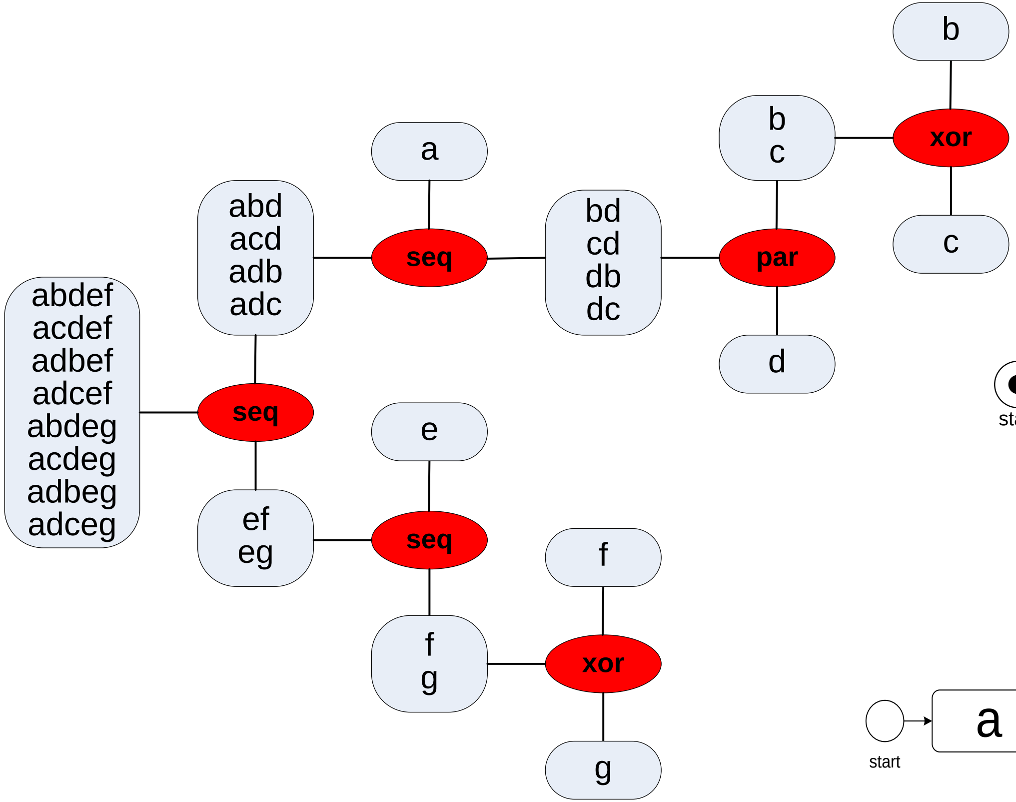
Split {b,c} into {b} and {c} using XOR decomposition

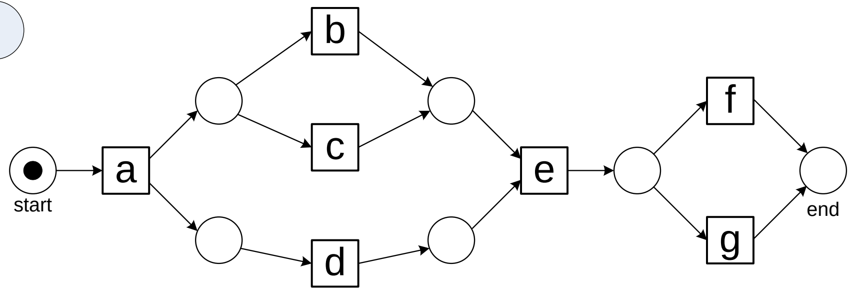


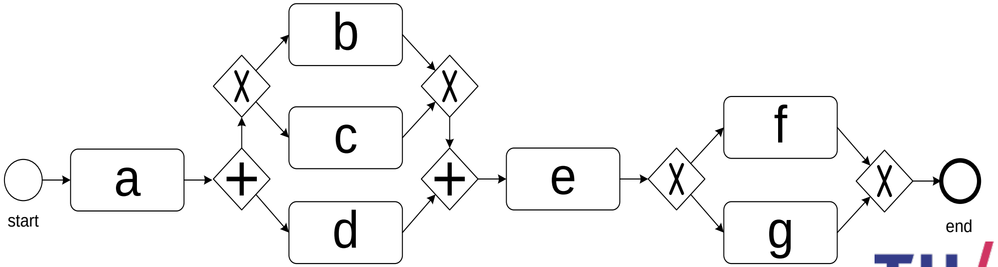
Result


a a a a a a a a
seq
b b b b
xor
c c c c c
par
d d d d d d d
seq
e e e e e e e
seq
...
no further decomposition is possible

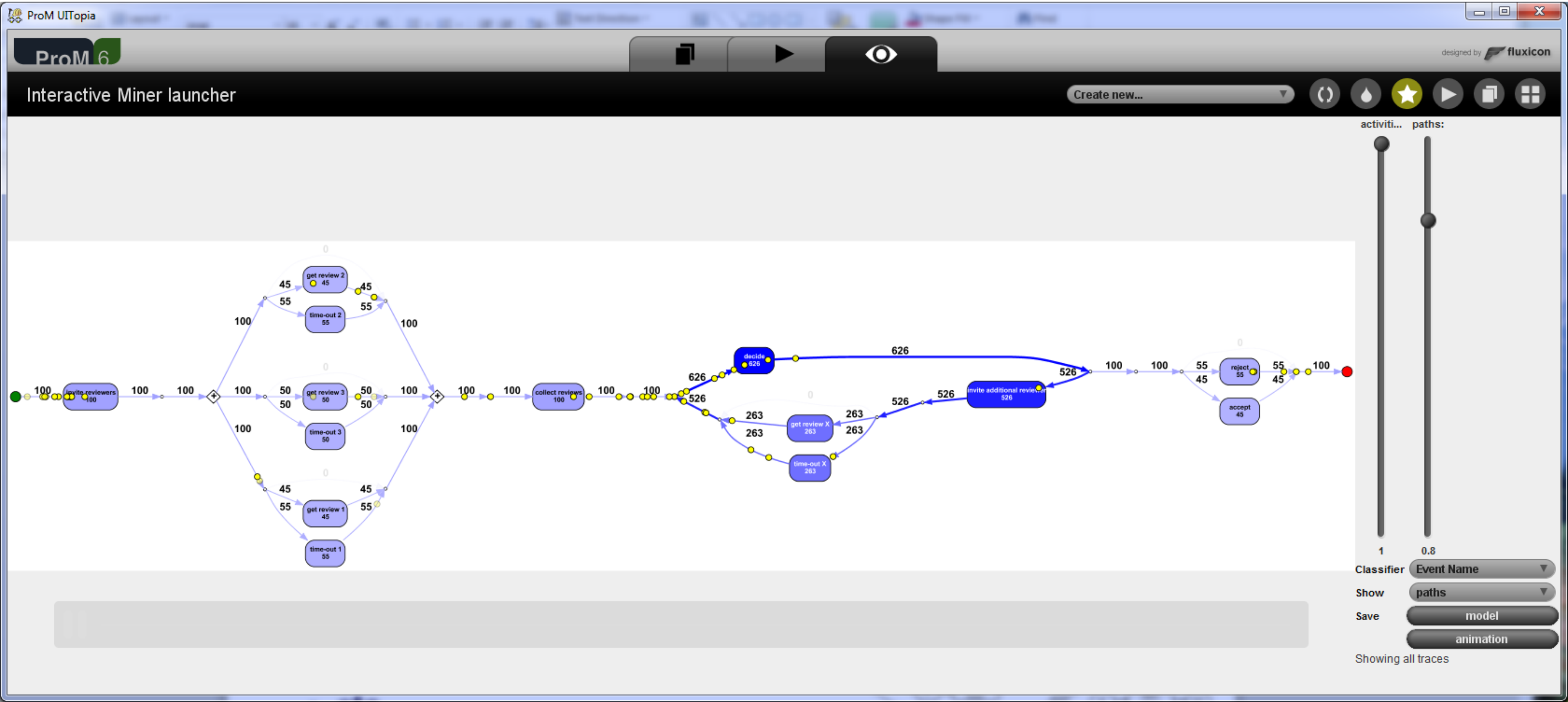
Process tree



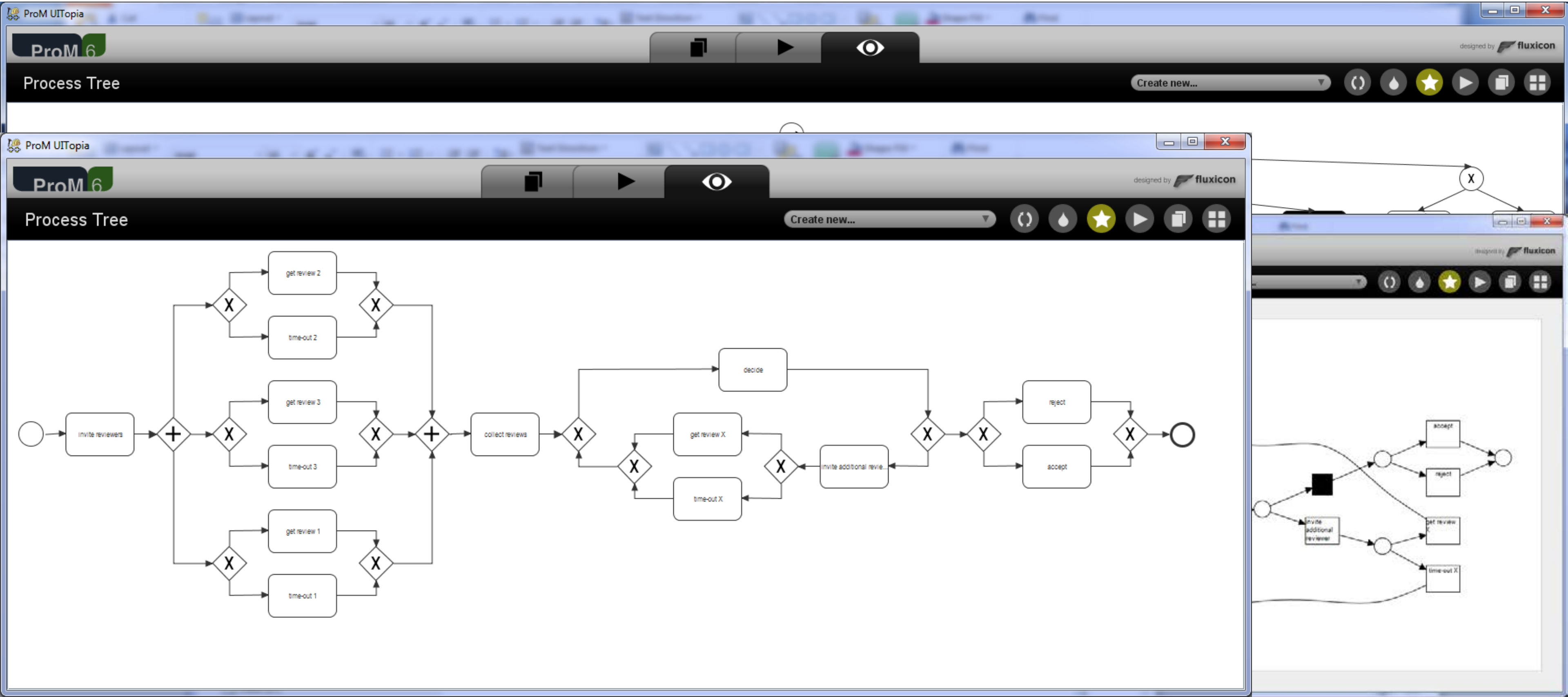




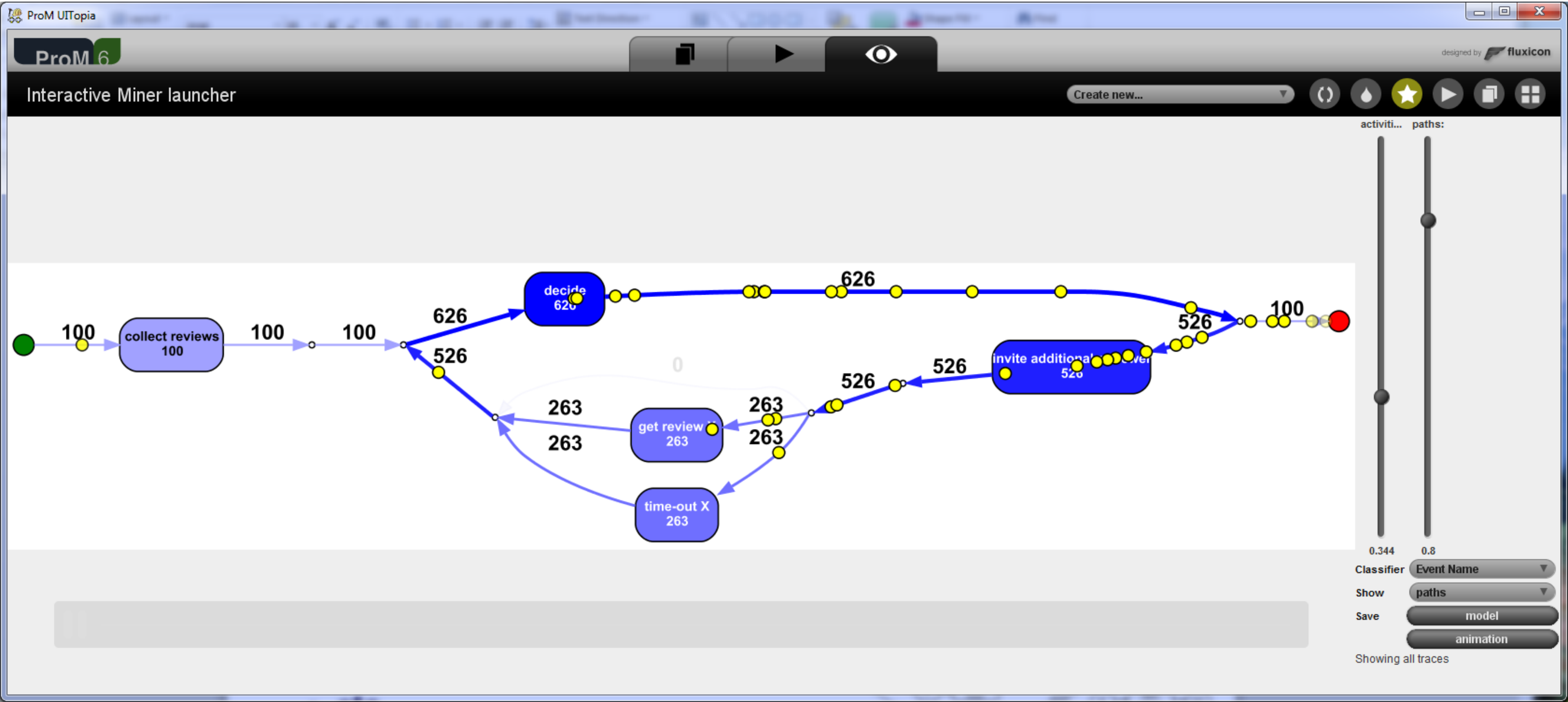
ProM's inductive miner

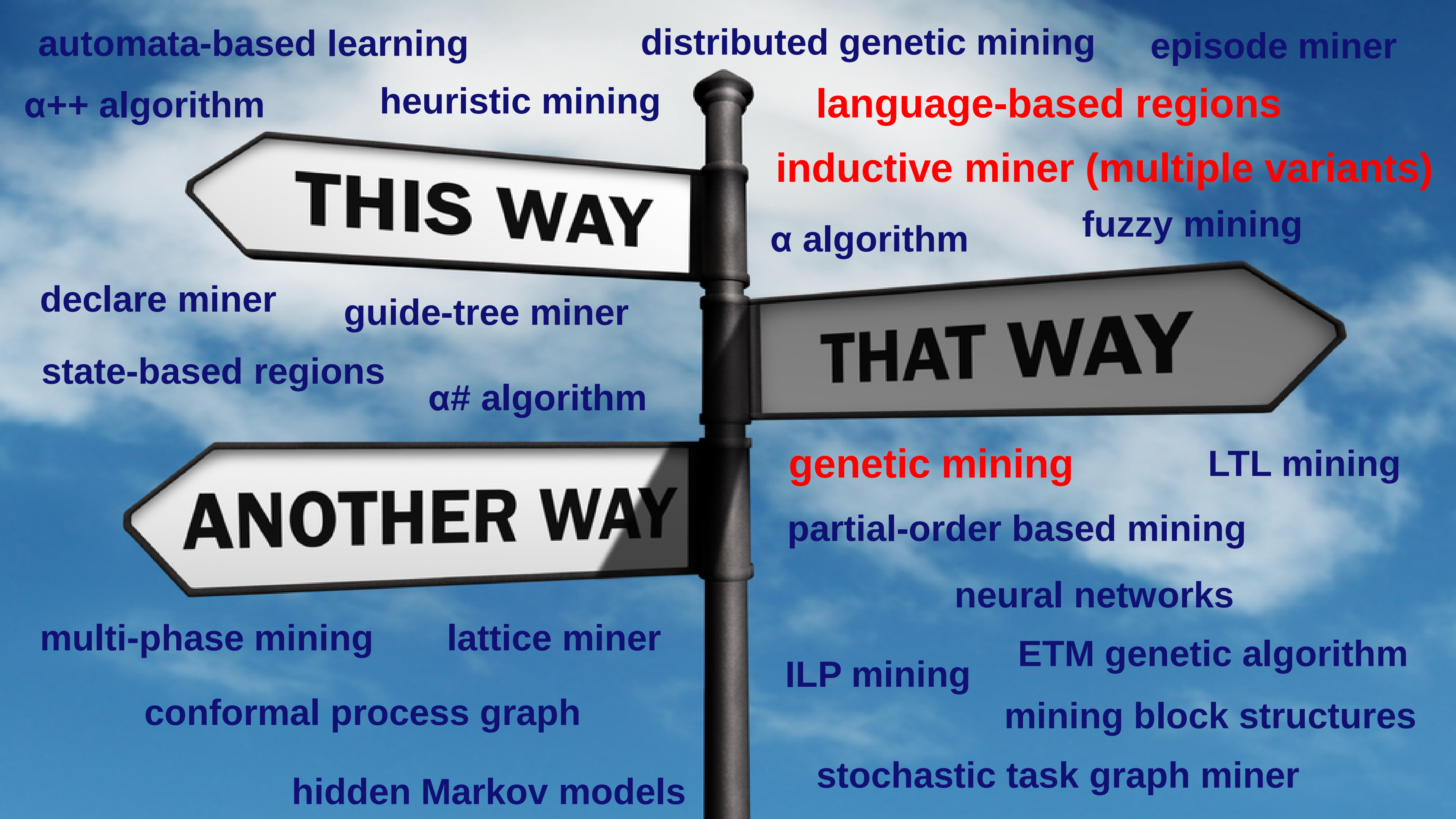


Result can be visualized using different notations



Seamless simplification of models




automata-based learning
distributed genetic mining
episode miner
α++ algorithm
heuristic mining
language-based regions
inductive miner (multiple variants)
THIS WAY
α algorithm
fuzzy mining
declare miner
guide-tree miner
THAT WAY
state-based regions
α# algorithm
genetic mining
LTL mining
ANOTHER WAY
partial-order based mining
neural networks
multi-phase mining
lattice miner
ETM genetic algorithm
ILP mining
conformal process graph
mining block structures
stochastic task graph miner
hidden Markov models

Part I: Preliminaries

- **Chapter 1** Introduction
- **Chapter 2** Process Modeling and Analysis
- **Chapter 3** Data Mining

Part II: From Event Logs to Process Models

- **Chapter 4** Getting the Data
- **Chapter 5** Process Discovery: An Introduction
- **Chapter 6** Advanced Process Discovery T[illegible]ues

Part III: Beyond Process Discovery

- **Chapter 7** Conformance Checking
- **Chapter 8** Mining Additional Perspectives
- **Chapter 9** Operational Support

Part IV: Putting Process Mining to Work

- **Chapter 10** Tool Support
- **Chapter 11** Analyzing “Lasagna Processes”
- **Chapter 12** Analyzing “Spaghetti Processes”

Part V: Reflection

- **Chapter 13** Cartography and Navigation
- **Chapter 14** Epilogue