Freie Universität Bozen
Libera Università di Bolzano
Università Liedia de Bulsan

Approximating Multi-Perspective Trace Alignment Using Trace Encodings

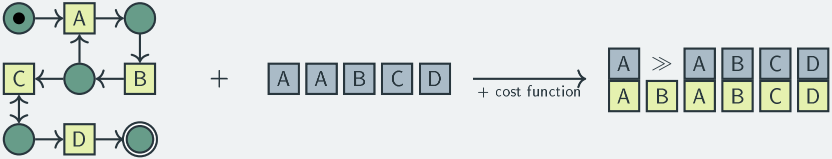
Alessandro Gianola, Jonghyeon Ko, Fabrizio Maria Maggi, Marco Montali, and Sarah Winkler
Free University of Bozen-Bolzano, Italy

21st International Conference on Business Process Management

14 September 2023, Utrecht

The Big Picture

Classical conformance checking



A	≫	A	B	C	D
A	B	A	B	C	D

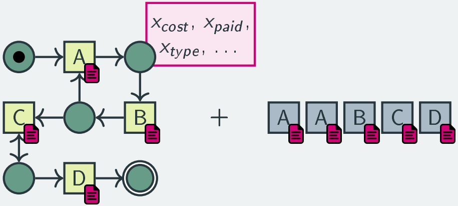
process model

log trace

alignment

The Big Picture

Classical conformance checking

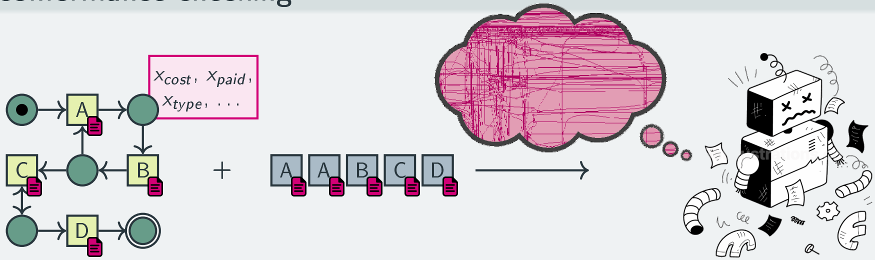


process model

log trace

The Big Picture

Classical conformance checking

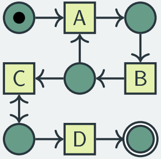


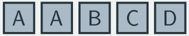
process model

log trace

The Big Picture

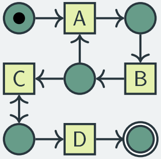
Approximate conformance checking





The Big Picture

Approximate conformance checking



$\xrightarrow[\text{up to length } n]{\text{process runs}}$

A B C D

A B C C D

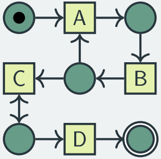
A B A B C D

A B C C C D

A A B C D

The Big Picture

Approximate conformance checking



$\xrightarrow[\text{up to length } n]{\text{process runs}}$

A B C D

A B C C D

A B A B C D

A B C C C D

$\xrightarrow{\text{encode}}$

(1,1,1,1,1)

(1,1,2,1,1)

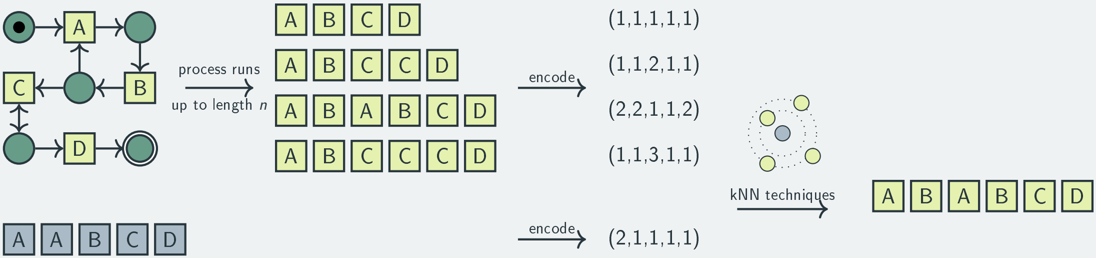
(2,2,1,1,2)

(1,1,3,1,1)

A A B C D $\xrightarrow{\text{encode}}$ (2,1,1,1,1)

The Big Picture

Approximate conformance checking



process runs $\xrightarrow{}$ up to length n

A B C D

A B C C D

A B A B C D

A B C C C D

$\xrightarrow{\text{encode}}$

(1,1,1,1,1)

(1,1,2,1,1)

(2,2,1,1,2)

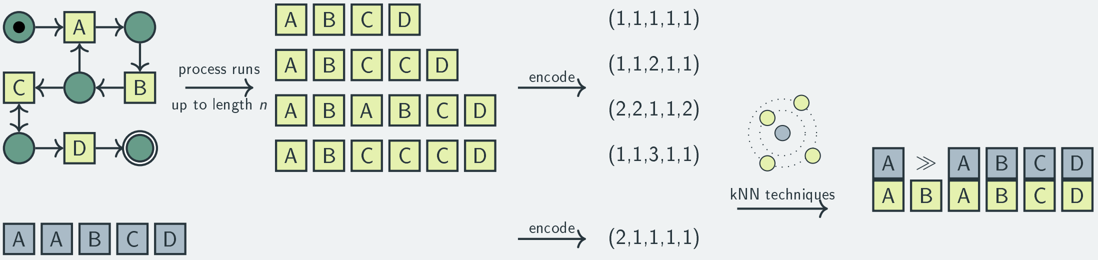
(1,1,3,1,1)

A A B C D $\xrightarrow{\text{encode}}$ (2,1,1,1,1)

kNN techniques $\xrightarrow{}$ A B A B C D

The Big Picture

Approximate conformance checking



process runs
up to length n

A B C D
A B C C D
A B A B C D
A B C C C D

encode

(1,1,1,1,1)
(1,1,2,1,1)
(2,2,1,1,2)
(1,1,3,1,1)

A A B C D

encode

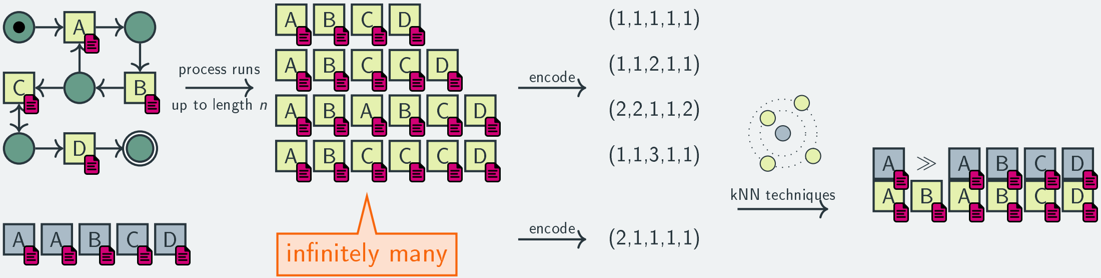
(2,1,1,1,1)

kNN techniques

A	≫	A	B	C	D
A	B	A	B	C	D

The Big Picture

Approximate conformance checking



process runs $\xrightarrow{}$ up to length n

infinitely many

$\xrightarrow{\text{encode}}$

(1,1,1,1,1)

(1,1,2,1,1)

(2,2,1,1,2)

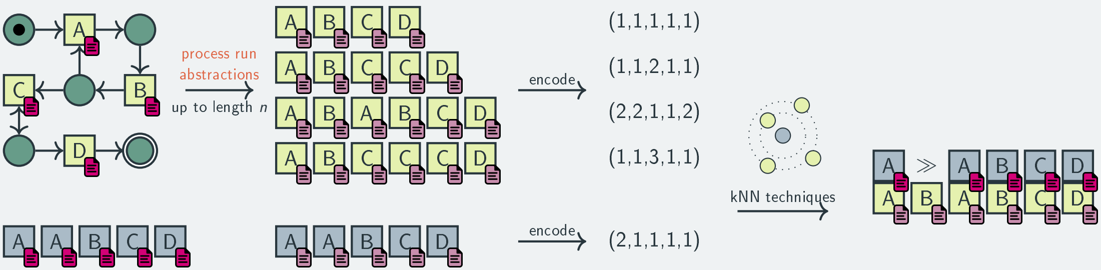
(1,1,3,1,1)

$\xrightarrow{\text{encode}}$ (2,1,1,1,1)

$\xrightarrow{\text{kNN techniques}}$

The Big Picture

Approximate conformance checking



process run abstractions

up to length n

encode

(1,1,1,1,1)

(1,1,2,1,1)

(2,2,1,1,2)

(1,1,3,1,1)

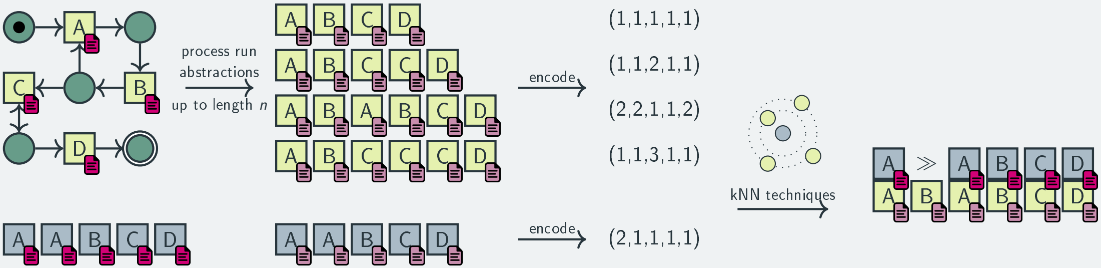
kNN techniques

encode

(2,1,1,1,1)

The Big Picture

Approximate conformance checking



process run abstractions up to length n

encode

(1,1,1,1,1)

(1,1,2,1,1)

(2,2,1,1,2)

(1,1,3,1,1)

encode

(2,1,1,1,1)

kNN techniques

Advantages

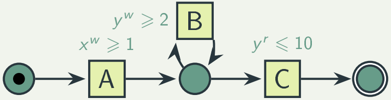
- fast (approximate) conformance checking
- user can choose encoding that prioritizes important trace aspects

Data Petri Nets

Data Petri net (DPN)

- based on Petri net

Example

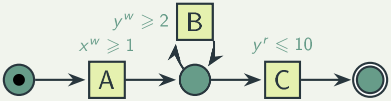


Data Petri Nets

Data Petri net (DPN)

- based on Petri net
- data: set of “global” variables V with numeric domain ($\mathbb{Q}$ or $\mathbb{Z}$)

Example



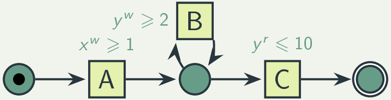
x	$\mathbb{Q}$
y	$\mathbb{Z}$

Data Petri Nets

Data Petri net (DPN)

- based on Petri net
- data: set of "global" variables V with numeric domain ($\mathbb{Q}$ or $\mathbb{Z}$)
- transitions have guards that read and write variables: linear arithmetic expressions over V^r and V^w

Example



x	$\mathbb{Q}$
y	$\mathbb{Z}$

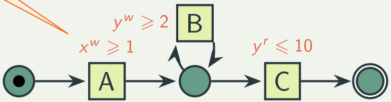
Data Petri Nets

Data Petri net (DPN)

- based on Petri net
- data: set of "global" variables V with numeric domain ($\mathbb{Q}$ or $\mathbb{Z}$)
- transitions have guards that read and write variables: linear arithmetic expressions over V^r and V^w

Example

$\cdot^r$ for read, $\cdot^w$ for written variables



x	$\mathbb{Q}$
y	$\mathbb{Z}$

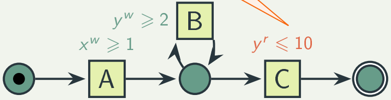
Data Petri Nets

Data Petri net (DPN)

- based on Petri net
- data: set of “global” variables V with numeric domain ($\mathbb{Q}$ or $\mathbb{Z}$)
- transitions have guards that read and write variables: linear arithmetic expressions over V^r and V^w

Example

check if y smaller or equal to 10



x	$\mathbb{Q}$
y	$\mathbb{Z}$

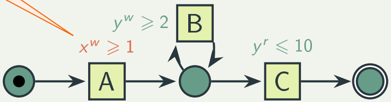
Data Petri Nets

Data Petri net (DPN)

- based on Petri net
- data: set of “global” variables V with numeric domain ($\mathbb{Q}$ or $\mathbb{Z}$)
- transitions have guards that read and write variables: linear arithmetic expressions over V^r and V^w

Example

set x to value greater or equal to 1



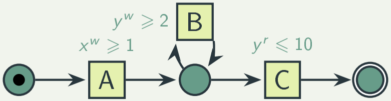
x	$\mathbb{Q}$
y	$\mathbb{Z}$

Data Petri Nets

Data Petri net (DPN)

- based on Petri net
- data: set of “global” variables V with numeric domain ($\mathbb{Q}$ or $\mathbb{Z}$)
- transitions have guards that read and write variables: linear arithmetic expressions over V^r and V^w

Example



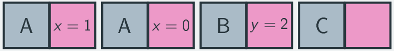
x	$\mathbb{Q}$
y	$\mathbb{Z}$

Assumption

- all guards are conjunctions of variable-to-constant comparisons
- this holds for guards automatically mined from data [Mannhardt et al 2016, de Leoni 2013]

Log trace

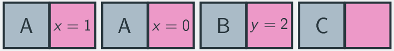
sequence of events containing activity and assignment



A	$x = 1$	A	$x = 0$	B	$y = 2$	C	

Log trace

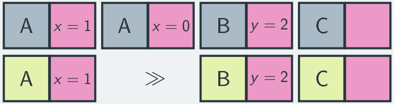
sequence of events containing activity and assignment



A	$x=1$	A	$x=0$	B	$y=2$	C	

Alignment

- combines log trace and model trace in sequence of moves
- every move has cost according to cost function: optimal alignment is the one with lowest cost



A	$x=1$	A	$x=0$	B	$y=2$	C	
A	$x=1$	$\gg$		B	$y=2$	C	

[Balanced multi-perspective conformance checking, Mannhardt et al 2016]

Log trace

sequence of events containing activity and assignment

A	$x=1$	A	$x=0$	B	$y=2$	C	

Alignment

- combines log trace and model trace in sequence of moves
- every move has cost according to cost function: optimal alignment is the one with lowest cost

A	$x=1$	A	$x=0$	B	$y=2$	C	
A	$x=1$	$\gg$		B	$y=2$	C	

[Balanced multi-perspective conformance checking, Mannhardt et al 2016]

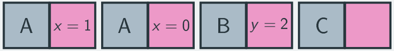
Fact

for DPN $\mathcal{N}$ and trace $\mathbf{e}$, can compute *bound*$(\mathcal{N}, \mathbf{e})$ on length of model trace in optimal alignment

[Data-aware conformance checking with SMT, Felli et al 2023]

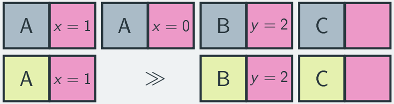
Log trace

sequence of events containing activity and assignment



Alignment

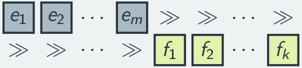
- combines log trace and model trace in sequence of moves
- every move has cost according to cost function:
  optimal alignment is the one with lowest cost



[Balanced multi-perspective conformance checking, Mannhardt et al 2016]

Fact

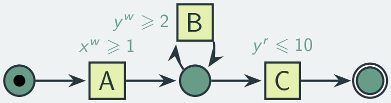
for DPN $\mathcal{N}$ and trace $\mathbf{e}$, can compute *bound*$(\mathcal{N}, \mathbf{e})$ on length of model trace in optimal alignment:
get worst-case alignment from shortest process run $f_1, \dots, f_k$ and $\mathbf{e} = e_1, \dots, e_m$



[Data-aware conformance checking with SMT, Felli et al 2023]

Observation

- infinite number of data values, but behavior only depends on whether values are within bounds

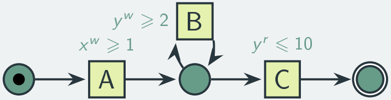


Observation

- infinite number of data values, but behavior only depends on whether values are within bounds
- for every variable, consider set of intervals:

  x: $]-\infty, 1[$ $[1, \infty]$

  y: $]-\infty, 2[$ $[2, 10]$ $]2, \infty]$

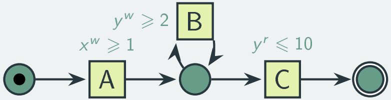


Observation

- infinite number of data values, but behavior only depends on whether values are within bounds
- for every variable, consider set of intervals:

  x: $]{-\infty}, 1[$ $[1, \infty]$

  y: $]{-\infty}, 2[$ $[2, 10]$ $]2, \infty]$

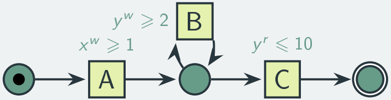


Abstract trace

sequence of abstract events containing activity and assignment to intervals

A	$x \in [1,\infty]$	A	$x \in]{-\infty},1[$	B	$y \in [2,\infty[$	C	

Observation



- infinite number of data values, but behavior only depends on whether values are within bounds
- for every variable, consider set of intervals:

  x: $]-\infty, 1[$ $[1, \infty]$

  y: $]-\infty, 2[$ $[2, 10]$ $]2, \infty]$

- can represent all model traces up to length k by finite set of abstract traces

A	$x \in [1,\infty]$ $y \in]-\infty,2[$	C	$x \in [1,\infty]$ $y \in]-\infty,2[$

. . .

A	$x \in [1,\infty]$ $y \in]-\infty,2[$	B	$x \in [1,\infty]$ $y \in [2,10]$	B	$x \in [1,\infty]$ $y \in [2,10]$	C	$x \in [1,\infty]$ $y \in [2,10]$

Abstract trace

sequence of abstract events containing activity and assignment to intervals

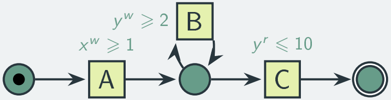
A	$x \in [1,\infty]$	A	$x \in]-\infty,1[$	B	$y \in [2,\infty[$	C	

Observation

- infinite number of data values, but behavior only depends on whether values are within bounds
- for every variable, consider set of intervals:

  $x:\quad]-\infty,1[\quad [1,\infty]$

  $y:\quad]-\infty,2[\quad [2,10]\quad]2,\infty]$
- can represent all model traces up to length k by finite set of abstract traces



A	$x\in[1,\infty]$ $y\in]-\infty,2[$	C	$x\in[1,\infty]$ $y\in]-\infty,2[$	$\cdots$	A	$x\in[1,\infty]$ $y\in]-\infty,2[$	B	$x\in[1,\infty]$ $y\in[2,10]$	B	$x\in[1,\infty]$ $y\in[2,10]$	C	$x\in[1,\infty]$ $y\in[2,10]$

Abstract trace

sequence of abstract events containing activity and assignment to intervals

A	$x\in[1,\infty]$	A	$x\in]-\infty,1[$	B	$y\in[2,\infty[$	C	

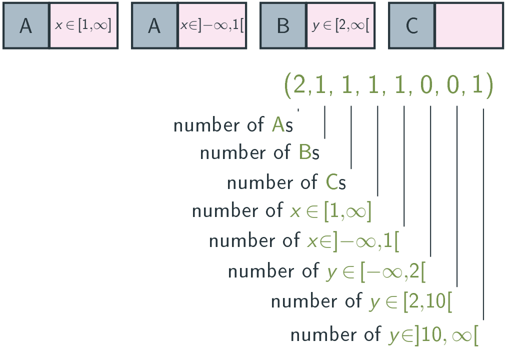
Key idea

- compute set T of all abstract model traces up to length $\mathit{bound}(\mathcal{N},\mathbf{e})$
- given log trace $\mathbf{e}$, search for closest model trace in T

Trace Encodings

1. Aggregate

count frequency of activity labels and intervals



A $x \in [1,\infty]$ A $x\in]-\infty,1[$ B $y \in [2,\infty[$ C

$(2,1, 1, 1, 1, 0, 0, 1)$

number of As

number of Bs

number of Cs

number of $x \in [1,\infty]$

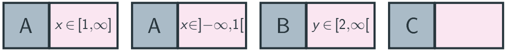
number of $x\in]-\infty,1[$

number of $y \in [-\infty,2[$

number of $y \in [2,10[$

number of $y\in]10, \infty[$

Trace Encodings



A $x \in [1,\infty]$ A $x \in]-\infty,1[$ B $y \in [2,\infty[$ C

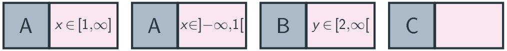
1. **Aggregate**
count frequency of activity labels and intervals

$(2, 1, 1, 1, 1, 0, 0, 1)$

2. **Boolean**
presence of activity labels and intervals

$(1, 1, 1, 1, 1, 0, 0, 1)$

Trace Encodings



1. Aggregate
count frequency of activity labels and intervals

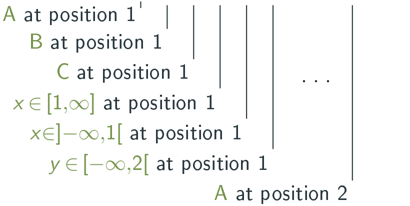
$(2,1,\ 1,\ 1,\ 1,\ 0,\ 0,\ 1)$

2. Boolean
presence of activity labels and intervals

$(1,\ 1,\ 1,\ 1,\ 1,\ 0,\ 0,\ 1)$

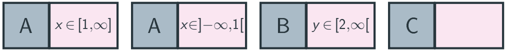
3. Complex index
for each trace position:
presence of each activity and each interval

$(1,0,\ 0,\ 1,\ 0,\ 0,\ 0,\ 0,\ 0,\ldots)$



A at position 1
B at position 1
C at position 1
$x \in [1,\infty]$ at position 1
$x \in]-\infty,1[$ at position 1
$y \in [-\infty,2[$ at position 1
A at position 2
...

Trace Encodings


A $x \in [1,\infty]$ A $x \in]-\infty,1[$ B $y \in [2,\infty[$ C

1. Aggregate

count frequency of activity labels and intervals

$(2,1,1,1,1,0,0,1)$

2. Boolean

presence of activity labels and intervals

$(1,1,1,1,1,0,0,1)$

3. Complex index

for each trace position:
presence of each activity and each interval

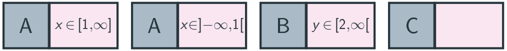
$(1,0,0,1,0,0,0,0,0,\ldots)$

4. Last state

like aggregate for control flow, but only last interval for data values

$(2,1,1,1,3)$

Trace Encodings



A $x \in [1,\infty]$ A $x \in]-\infty,1[$ B $y \in [2,\infty[$ C

1. Aggregate
count frequency of activity labels and intervals

$(2,1,1,1,1,0,0,1)$

2. Boolean
presence of activity labels and intervals

$(1,1,1,1,1,0,0,1)$

3. Complex index
for each trace position:
presence of each activity and each interval

$(1,0,0,1,0,0,0,0,0,\ldots)$

4. Last state
like aggregate for control flow, but only last interval for data values

$(2,1,1,1,3)$

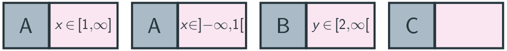
5. Aggregate + *p*-grams
enhance aggregate encoding with *p*-grams for activity labels

$(2,1,1,1,1,0,0,1,1,1,\ldots)$

weighted number of occurrences of AA

weighted number of occurrences of AB

Trace Encodings



A $x \in [1,\infty]$ A $x \in]-\infty,1[$ B $y \in [2,\infty[$ C

1. Aggregate
count frequency of activity labels and intervals

$(2, 1, 1, 1, 1, 0, 0, 1)$

2. Boolean
presence of activity labels and intervals

$(1, 1, 1, 1, 1, 0, 0, 1)$

3. Complex index
for each trace position:
presence of each activity and each interval

$(1, 0, 0, 1, 0, 0, 0, 0, 0, \ldots)$

4. Last state
like aggregate for control flow, but only last interval for data values

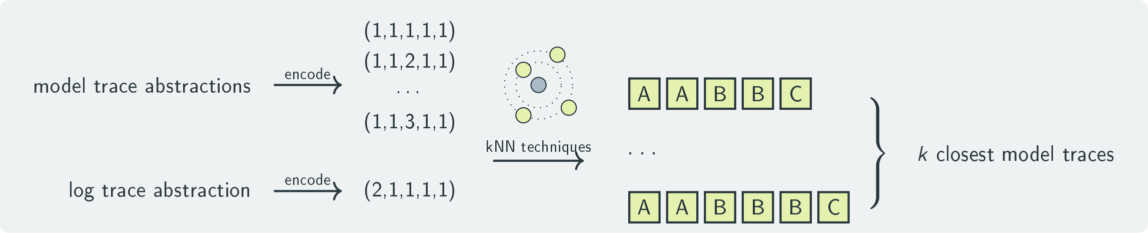
$(2, 1, 1, 1, 3)$

5. Aggregate + *p*-grams
enhance aggregate encoding with *p*-grams for activity labels

$(2, 1, 1, 1, 1, 0, 0, 1, 1, 1, \ldots)$

[Complex symbolic sequence encodings for predictive monitoring of business processes, Leontjeva et al 2015]

Solving the Alignment Problem using *k*NN



model trace abstractions $\xrightarrow{\text{encode}}$ (1,1,1,1,1)

(1,1,2,1,1)

. . .

(1,1,3,1,1)

log trace abstraction $\xrightarrow{\text{encode}}$ (2,1,1,1,1)

$\xrightarrow{\text{kNN techniques}}$

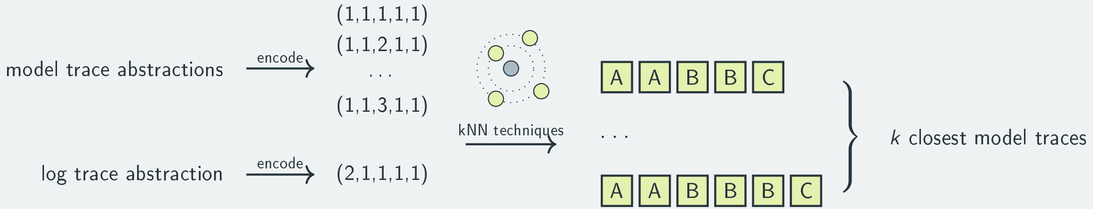
A A B B C

. . .

A A B B B C

k closest model traces

Solving the Alignment Problem using *k*NN



model trace abstractions $\xrightarrow{\text{encode}}$ (1,1,1,1,1) (1,1,2,1,1) . . . (1,1,3,1,1)

log trace abstraction $\xrightarrow{\text{encode}}$ (2,1,1,1,1)

kNN techniques

A A B B C

. . .

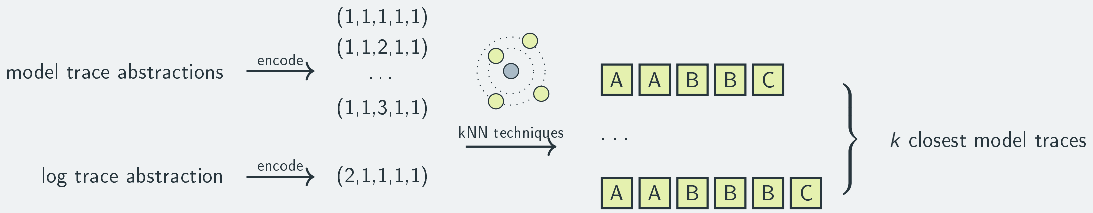
A A B B B C

k closest model traces

Technical details

- experiment with different distance measures: euclidean, manhattan, cosine

Solving the Alignment Problem using *k*NN



model trace abstractions $\xrightarrow{\text{encode}}$ (1,1,1,1,1)
(1,1,2,1,1)
. . .
(1,1,3,1,1)

log trace abstraction $\xrightarrow{\text{encode}}$ (2,1,1,1,1)

kNN techniques

A A B B C
. . .
A A B B B C

k closest model traces

Technical details

- experiment with different distance measures: euclidean, manhattan, cosine
- useful: weigh vector components to give same weight to control flow and data (improve precision by 25%)

Experiments

Setup

- compute model trace abstractions up to certain length for sepsis and road fines data sets
- for every log trace
  - compute optimal alignment using CoCoMoT, and respective (abstract) model trace $\mathbf{f}$
  - compute k best-matching (abstract) model traces P using approximate approach

Experiments

Setup

- compute model trace abstractions up to certain length for sepsis and road fines data sets
- for every log trace
  - compute optimal alignment using CoCoMoT, and respective (abstract) model trace $\mathbf{f}$
  - compute k best-matching (abstract) model traces P using approximate approach
- measure on entire log
  - precision: how often $\mathbf{f}$ is in P
  - similarity: average distance of $\mathbf{f}$ to model traces in P

Similarity (road fines, sepsis)

Encoding method	Distance metric	k	Similarity (ref = CoCoMot)	Time (sec)
aggregate	Cosine	(1/3/5/10)	(0.971 / 0.971 / 0.971 / 0.973)	(0.07 / 0.07 / 0.07 / 0.07)
aggregate	Euclidean	(1/3/5/10)	(0.963 / 0.963 / 0.963 / 0.960)	(0.11 / 0.12 / 0.12 / 0.12)
aggregate	Manhattan	(1/3/5/10)	(0.933 / 0.938 / 0.942 / 0.948)	(0.18 / 0.13 / 0.16 / 0.14)
boolean	Cosine	(1/3/5/10)	(0.978 / 0.978 / 0.978 / 0.977)	(0.06 / 0.07 / 0.08 / 0.08)
boolean	Euclidean	(1/3/5/10)	(0.978 / 0.978 / 0.978 / 0.977)	(0.16 / 0.12 / 0.12 / 0.17)
boolean	Manhattan	(1/3/5/10)	(0.978 / 0.978 / 0.978 / 0.977)	(0.13 / 0.13 / 0.14 / 0.14)
complexindex	Cosine	(1[illegible]	[illegible]	[illegible]25 / 0.25 / 0.25)
complexindex	Euclidean	(1[illegible]	[illegible]	[illegible]31 / 0.35 / 0.35)
complexindex	Manhattan	(1[illegible]	[illegible]	[illegible]58 / 0.70 / 0.64)
laststate	Cosine	(1[illegible]	[illegible]	[illegible]07 / 0.07 / 0.07)
laststate	Euclidean	(1[illegible]	[illegible]	[illegible]12 / 0.12 / 0.17)
laststate	Manhattan	(1/3/5/10)	(0.985 / 0.985 / 0.984 / 0.984)	(0.13 / 0.18 / 0.13 / 0.14)
p-gram+aggregate	Cosine	(1/3/5/10)	(**0.9**[illegible] / **0.996** / **0.996** / **0.995**)	(0.06 / 0.07 / 0.06 / 0.07)
p-gram+aggregate	Euclidean	(1/3/5/10)	(**0.996** / **0.996** / **0.996** / **0.995**)	(0.10 / 0.10 / 0.15 / 0.11)
p-gram+aggregate	Manhattan	(1/3/5/10)	(**0.996** / **0.996** / **0.996** / **0.995**)	(0.15 / 0.12 / 0.13 / 0.12)

p-gram+aggregate:
0.99 for best single trace

Encoding method	Distance metric	k	Simi[illegible]	[illegible]
aggregate	Cosine	(1/3/5/10)	(0.60[illegible]	[illegible]
aggregate	Euclidean	(1/3/5/10)	(0.60[illegible]	[illegible]
aggregate	Manhattan	(1/3/5/10)	(0.60[illegible]	[illegible]
boolean	Cosine	(1/3/5/10)	(0.606 / 0.606 / 0.6[illegible] / 0.606)	(0.01 / 0.01 / 0.01 / 0.01)
boolean	Euclidean	(1/3/5/10)	(0.606 / 0.606 / 0.606 / 0.607)	(0.02 / 0.02 / 0.01 / 0.02)
boolean	Manhattan	(1/3/5/10)	(0.606 / 0.606 / 0.606 / 0.606)	(0.02 / 0.02 / 0.02 / 0.02)
complexindex	Cosine	(1/3/5/10)	(**0.620** / **0.620** / **0.620** / 0.620)	(0.04 / 0.04 / 0.04 / 0.04)
complexindex	Euclidean	(1/3/5/10)	(0.619 / **0.620** / **0.620** / 0.620)	(0.04 / 0.04 / 0.04 / 0.04)
complexindex	Manhattan	(1/3/5/10)	(0.619 / **0.620** / **0.620** / **0.621**)	(0.05 / 0.05 / 0.05 / 0.05)
laststate	Cosine	(1/3/5/10)	(0.603 / 0.602 / 0.601 / 0.602)	(0.01 / 0.01 / 0.01 / 0.01)
laststate	Euclidean	(1/3/5/10)	(0.603 / 0.601 / 0.600 / 0.600)	(0.02 / 0.02 / 0.02 / 0.02)
laststate	Manhattan	(1/3/5/10)	(0.603 / 0.601 / 0.600 / 0.601)	(0.01 / 0.02 / 0.02 / 0.02)
p-gram+aggregate	Cosine	(1/3/5/10)	(0.604 / 0.607 / 0.607 / 0.607)	(0.01 / 0.01 / 0.01 / 0.01)
p-gram+aggregate	Euclidean	(1/3/5/10)	(0.605 / 0.607 / 0.607 / 0.607)	(0.03 / 0.02 / 0.02 / 0.02)
p-gram+aggregate	Manhattan	(1/3/5/10)	(0.605 / 0.607 / 0.606 / 0.607)	(0.02 / 0.02 / 0.02 / 0.02)

complexindex:
0.62 for best single trace

Similarity (road fines, sepsis)

Encoding method	Distance metric	k	Similarity (ref = CoCoMot)	Time (sec)
aggregate	Cosine	(1/3/5/10)	(0.971 / 0.971 / 0.971 / 0.973)	(0.07 / 0.07 / 0.07 / 0.07)
aggregate	Euclidean	(1/3/5/10)	(0.963 / 0.963 / 0.963 / 0.960)	(0.11 / 0.12 / 0.12 / 0.12)
aggregate	Manhattan	(1/3/5/10)	(0.933 / 0.938 / 0.942 / 0.948)	(0.18 / 0.13 / 0.16 / 0.14)
boolean	Cosine	(1/3/5/10)	(0.978 / 0.978 / 0.978 / 0.977)	(0.06 / 0.07 / 0.08 / 0.08)
boolean	Euclidean	(1/3/5/10)	(0.978 / 0.978 / 0.978 / 0.977)	(0.16 / 0.12 / 0.12 / 0.17)
boolean	Manhattan	(1/3/5/10)	(0.978 / 0.978 / 0.978 / 0.977)	(0.13 / 0.13 / 0.14 / 0.14)
complexindex	Cosine	(1		25 / 0.25 / 0.25)
complexindex	Euclidean	(1		31 / 0.35 / 0.35)
complexindex	Manhattan	(1		68 / 0.70 / 0.64)
laststate	Cosine	(1		07 / 0.07 / 0.07)
laststate	Euclidean	(1		12 / 0.12 / 0.17)
laststate	Manhattan	(1/3/5/10)	(0.985 / 0.985 / 0.984 / 0.984)	(0.13 / 0.18 / 0.13 / 0.14)
p-gram+aggregate	Cosine	(1/3/5/10)	(**0.996** / **0.996** / **0.996** / **0.995**)	(0.06 / 0.07 / 0.06 / 0.07)
p-gram+aggregate	Euclidean	(1/3/5/10)	(**0.996** / **0.996** / **0.996** / **0.995**)	(0.10 / 0.10 / 0.15 / 0.11)
p-gram+aggregate	Manhattan	(1/3/5/10)	(**0.996** / **0.996** / **0.996** / **0.995**)	(0.15 / 0.12 / 0.13 / 0.12)

p-gram+aggregate:
0.99 for best single trace

Encoding method	Distance metric	k	Simi	
aggregate	Cosine	(1/3/5/10)	(0.60	
aggregate	Euclidean	(1/3/5/10)	(0.60	
aggregate	Manhattan	(1/3/5/10)	(0.606 / 0.606 / 0.606 / 0.606)	(0.02 / 0.02 / 0.02 / 0.02)
boolean	Cosine	(1/3/5/10)	(0.606 / 0.606 / 0.606 / 0.606)	(0.01 / 0.01 / 0.01 / 0.01)
boolean	Euclidean	(1/3/5/10)	(0.606 / 0.606 / 0.606 / 0.607)	(0.02 / 0.02 / 0.01 / 0.02)
boolean	Manhattan	(1/3/5/10)	(0.606 / 0.606 / 0.606 / 0.606)	(0.02 / 0.02 / 0.02 / 0.02)
complexindex	Cosine	(1/3/5/10)	(**0.620** / **0.620** / **0.620** / 0.620)	(0.04 / 0.04 / 0.04 / 0.04)
complexindex	Euclidean	(1/3/5/10)	(0.619 / **0.620** / **0.620** / 0.620)	(0.04 / 0.04 / 0.04 / 0.04)
complexindex	Manhattan	(1/3/5/10)	(0.619 / **0.620** / **0.620** / **0.621**)	(0.05 / 0.05 / 0.05 / 0.05)
laststate	Cosine	(1/3/5/10)	(0.603 / 0.602 / 0.601 / 0.602)	(0.01 / 0.01 / 0.01 / 0.01)
laststate	Euclidean	(1/3/5/10)	(0.603 / 0.601 / 0.600 / 0.600)	(0.02 / 0.02 / 0.02 / 0.02)
laststate	Manhattan	(1/3/5/10)	(0.603 / 0.601 / 0.600 / 0.601)	(0.01 / 0.02 / 0.02 / 0.02)
p-gram+aggregate	Cosine	(1/3/5/10)	(0.604 / 0.607 / 0.607 / 0.607)	(0.01 / 0.01 / 0.01 / 0.01)
p-gram+aggregate	Euclidean	(1/3/5/10)	(0.605 / 0.607 / 0.607 / 0.607)	(0.03 / 0.02 / 0.02 / 0.02)
p-gram+aggregate	Manhattan	(1/3/5/10)	(0.605 / 0.607 / 0.606 / 0.607)	(0.02 / 0.02 / 0.02 / 0.02)

complexindex:
0.62 for best single trace

Precision (road fines, sepsis)

Encoding method	Distance metric	k (%)	Precision (ref = CoCoMot)	Time (sec)
aggregate	Cosine	(10% / 20% / 30%)	(0.613 / 0.740 / 0.803)	(0.08 / 0.09 / 0.10)
aggregate	Euclidean	(10% / 20% / 30%)	(0.195 / 0.501 / 0.551)	(0.19 / 0.13 / 0.17)
aggregate	Manhattan	(10% / 20% / 30%)	(0.195 / 0.518 / 0.555)	(0.14 / 0.21 / 0.16)
boolean	Cosine	(10% / 20% / 30%)	(0.580 / 0.712 / 0.756)	(0.08 / 0.17 / 0.10)
boolean	Euclidean	(10% / 20% / 30%)	(0.597 / 0.725 / 0.808)	(0.13 / 0.19 / 0.13)
boolean	Manhattan	(10% / 20% / 30%)	(0.596 / 0.729 / 0.828)	(0.16 / 0.15 / 0.16)
complexindex	Cosine	(10% / 20		28 / 0.24 / 0.27)
complexindex	Euclidean	(10% / 20		36 / 0.33 / 0.37)
complexindex	Manhattan	(10% / 20		70 / 0.73 / 0.67)
laststate	Cosine	(10% / 20		08 / 0.08 / 0.17)
laststate	Euclidean	(10% / 20		12 / 0.13 / 0.14)
laststate	Manhattan	(10% / 20% / 30%)	910 / 0.754 / 0.882)	(0.16 / 0.18 / 0.16)
p-gram+aggregate	Cosine	(10% / 20% / 30%)	(0.705 / 0.776 / 0.817)	(0.07 / 0.08 / 0.08)
p-gram+aggregate	Euclidean	(10% / 20% / 30%)	(0.715 / 0.776 / 0.853)	(0.11 / 0.12 / 0.12)
p-gram+aggregate	Manhattan	(10% / 20% / 30%)	(**0.719** / **0.798** / **0.898**)	(0.12 / 0.14 / 0.18)

0.71 for best 10%,
0.90 for best 30%

Encoding method	Distance metric	k (%)	Precision (ref	
aggregate	Cosine	(10% / 20% / 30%)	(0.813 / 0.833	
aggregate	Euclidean	(10% / 20% / 30%)	(0.888 / 0.898	
aggregate	Manhattan	(10% / 20% / 30%)	(0.888 / 0.898	
boolean	Cosine	(10% / 20% / 30%)	(0.776 / 0.800	
boolean	Euclidean	(10% / 20% / 30%)	(0.854 / 0.864 / 0.873)	(0.02 / 0.02 / 0.02)
boolean	Manhattan	(10% / 20% / 30%)	(0.852 / 0.864 / 0.873)	(0.02 / 0.02 / 0.03)
complexindex	Cosine	(10% / 20% / 30%)	(0.816 / 0.816 / 0.888)	(0.04 / 0.04 / 0.04)
complexindex	Euclidean	(10% / 20% / 30%)	(0.891 / 0.931 / 0.949)	(0.05 / 0.04 / 0.05)
complexindex	Manhattan	(10% / 20% / 30%)	(**0.924** / **0.938** / **0.951**)	(0.08 / 0.07 / 0.07)
laststate	Cosine	(10% / 20% / 30%)	(0.822 / 0.822 / 0.822)	(0.01 / 0.01 / 0.01)
laststate	Euclidean	(10% / 20% / 30%)	(0.857 / 0.923 / 0.924)	(0.01 / 0.01 / 0.02)
laststate	Manhattan	(10% / 20% / 30%)	(0.855 / 0.923 / 0.924)	(0.02 / 0.02 / 0.02)
p-gram+aggregate	Cosine	(10% / 20% / 30%)	(0.864 / 0.881 / 0.891)	(0.01 / 0.01 / 0.01)
p-gram+aggregate	Euclidean	(10% / 20% / 30%)	(0.914 / 0.926 / 0.939)	(0.02 / 0.02 / 0.02)
p-gram+aggregate	Manhattan	(10% / 20% / 30%)	(0.914 / 0.926 / 0.939)	(0.04 / 0.03 / 0.03)

0.92 for best 10%,
0.95 for best 30%

Experiments

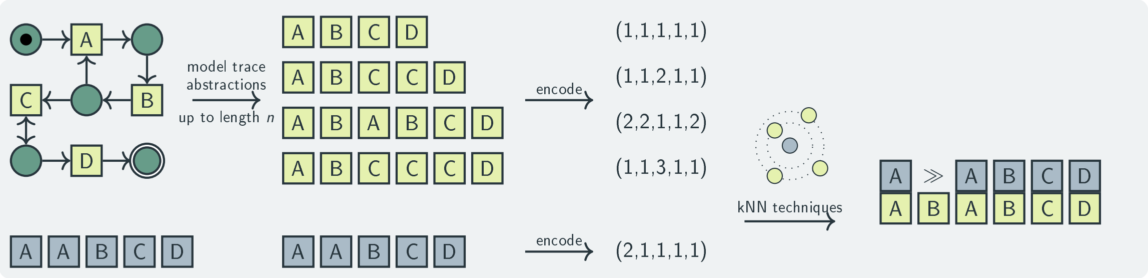
Setup

- compute model trace abstractions up to certain length for sepsis and road fines data sets
- for every log trace
  - compute optimal alignment using CoCoMoT, and respective (abstract) model trace $\mathbf{f}$
  - compute k best-matching (abstract) model traces P using approximate approach
- measure on entire log
  - precision: how often $\mathbf{f}$ is in P
  - similarity: average distance of $\mathbf{f}$ to model traces in P

Performance

approximate approach is ~ 100 times faster than CoCoMoT, ~ 10 times faster than ProM

Conclusion



model trace abstractions $\xrightarrow{\hspace{2em}}$ up to length n

A B C D

A B C C D

A B A B C D

A B C C C D

$\xrightarrow{\text{encode}}$

(1,1,1,1,1)

(1,1,2,1,1)

(2,2,1,1,2)

(1,1,3,1,1)

$\xrightarrow{\text{kNN techniques}}$

A ≫ A B C D

A B A B C D

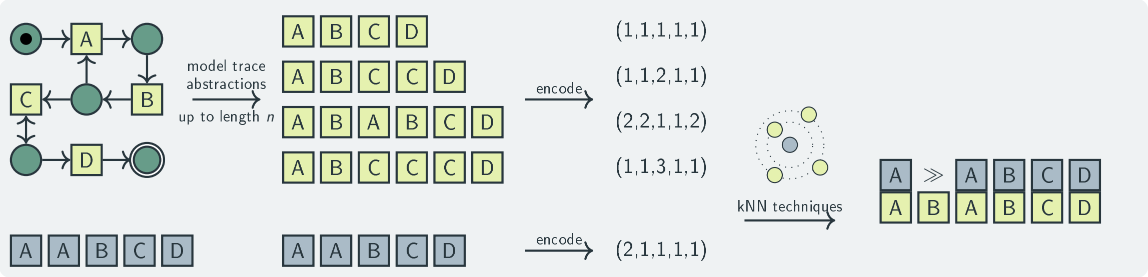
A A B C D

A A B C D $\xrightarrow{\text{encode}}$ (2,1,1,1,1)

Advantages

- fast (approximate) conformance checking
- approximation as preprocessing:
  precise conformance checking can be applied to returned model traces

Conclusion



Advantages

- fast (approximate) conformance checking
- approximation as preprocessing:
  precise conformance checking can be applied to returned model traces

Future work

extend to probabilistic trace alignment and more general DPNs