

A Boyer-Moore Type Algorithm for Timed Pattern Matching

Masaki Waga Takumi Akazaki Ichiro Hasuo

University of Tokyo
25 Aug 2016, FORMATS2016

1

Timed Pattern Matching

2

with

Boyer-Moore Type
Algorithm

Timed Pattern Matching

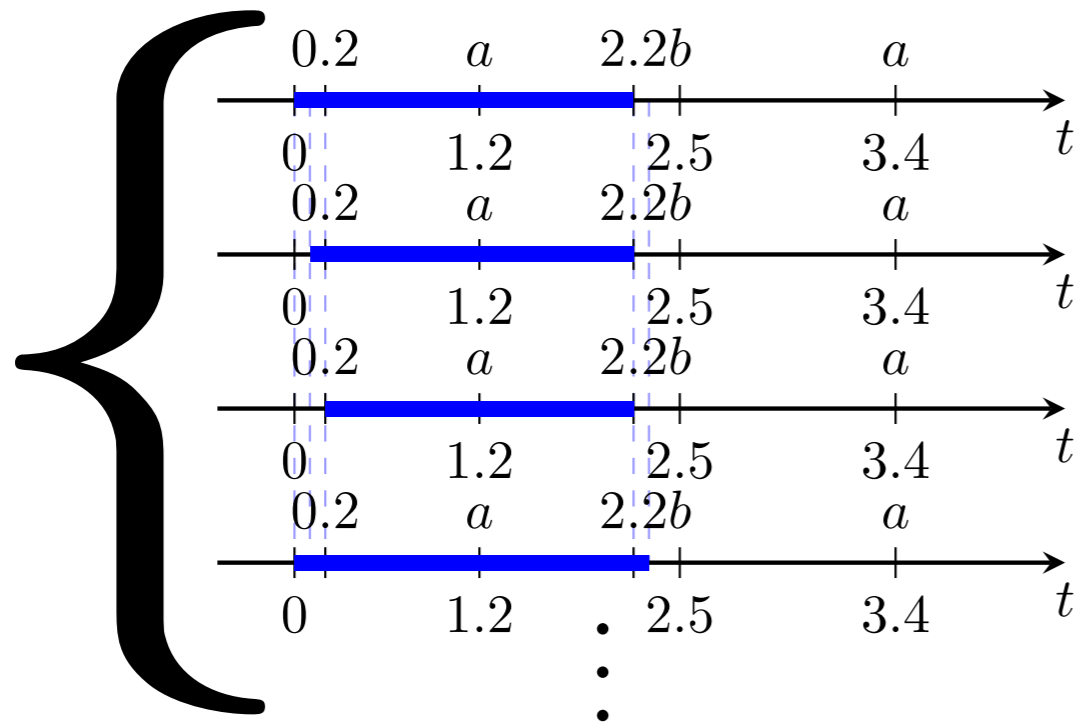
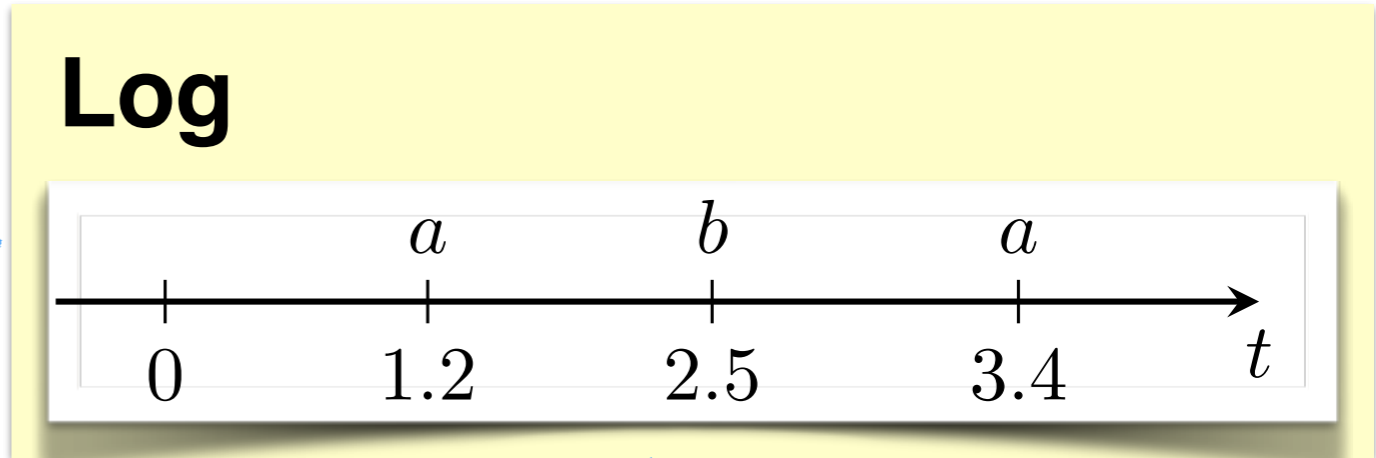
with

Boyer-Moore Type Algorithm

Motivation



running



monitoring
for failing



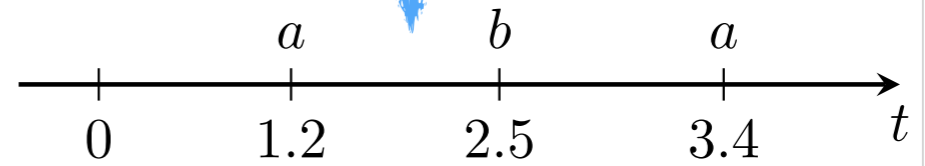
Timed pattern matching
can be used.

Another Motivation



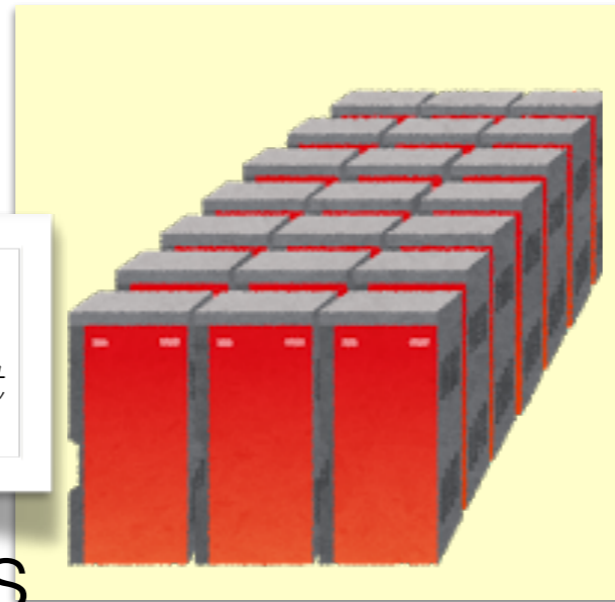
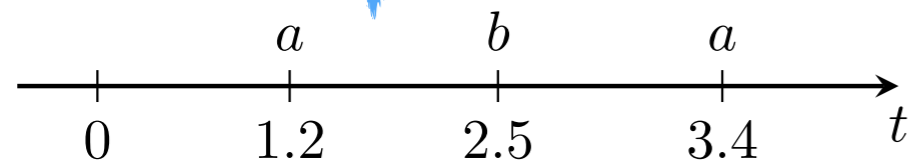
Another Motivation

log with timestamp

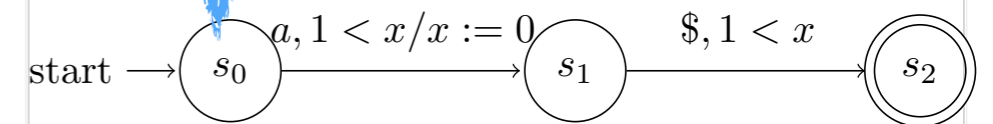


Another Motivation

log with timestamp



pattern



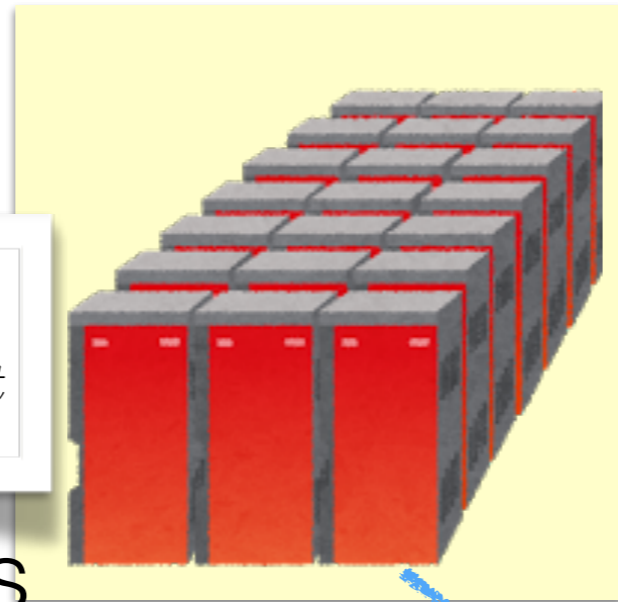
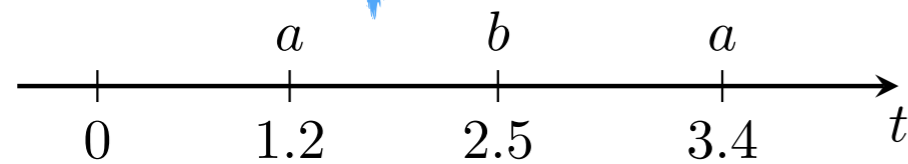
Analyze logs with
timed pattern matching

Send logs
(activities)

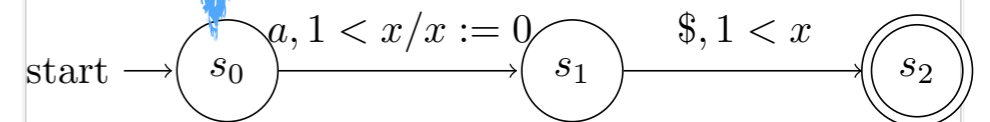


Another Motivation

log with timestamp



pattern



Analyze logs with timed pattern matching

Send logs (activities)



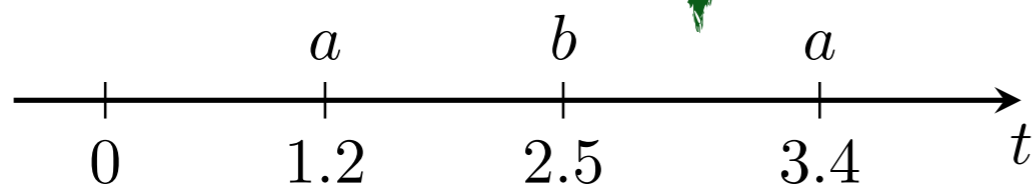
Hit the jackpot!!



Timed Pattern Matching

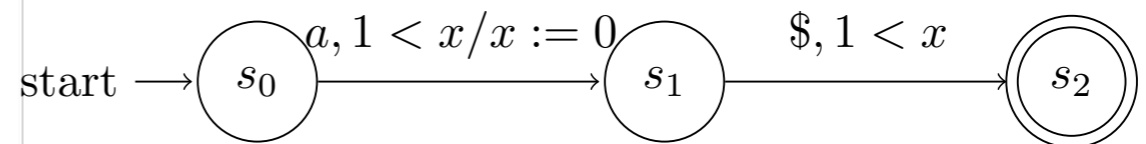
Input

timed word w



Target String

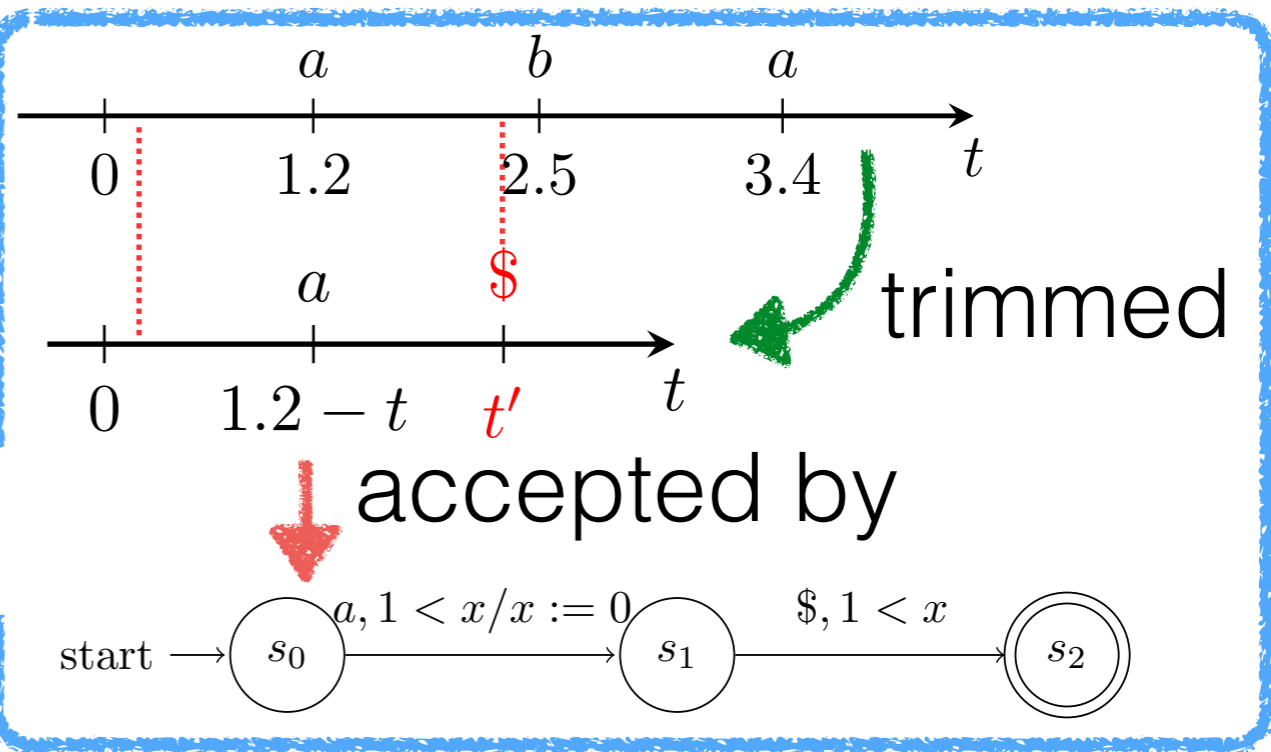
timed automaton \mathcal{A}



Pattern

Output

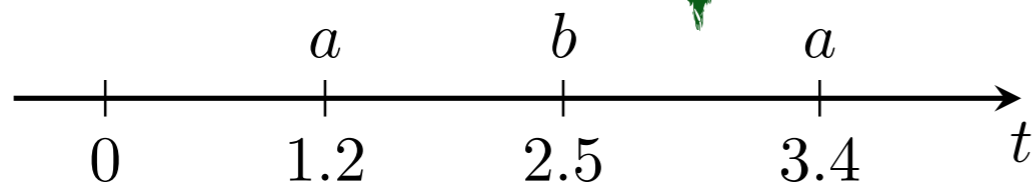
$$\{(t, t') \mid w|_{(t, t')} \in L(\mathcal{A})\}$$



Timed Pattern Matching

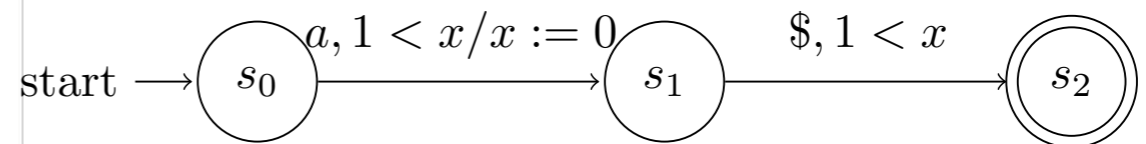
Input

timed word w



Target String

timed automaton \mathcal{A}



Pattern

Output

$$\{(t, t') \mid w|_{(t, t')} \in L(\mathcal{A})\}$$

signal and timed regular expression in [Ulus et al., 2014]

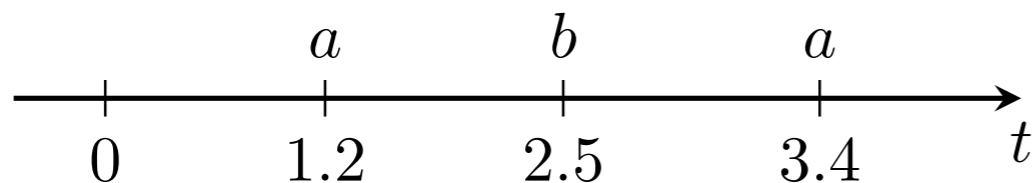
Timed Pattern Matching

Input

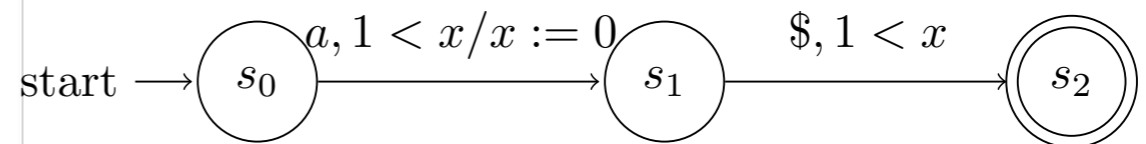
A log of a system

A specification of faults

timed word w



timed automaton \mathcal{A}

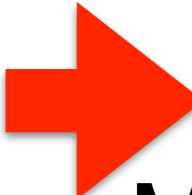


Output

The set of intervals when faults are detected

$$\{(t, t') \mid w|_{(t, t')} \in L(\mathcal{A})\}$$

Timed Pattern Matching with Boyer-Moore type Algorithm

	Target	Pattern	Algorithms
 String Matching	String	String	<ul style="list-style-type: none"> •Naive •Boyer-Moore
Pattern Matching	String	Regular Language	<ul style="list-style-type: none"> •Naive •Boyer-Moore
Timed Pattern Matching	Timed Word	Timed Automaton	<ul style="list-style-type: none"> •Naive •Boyer-Moore

Contribution

String Matching

Input

Target String

- Strings str : HERE_IS_A_SIMPLE_EXAMPLE
 pat : EXAMPLE

Output

Pattern

$$\{(i, j) \mid str(i, j) = pat\} = \{(18, 24)\}$$

123456789012345678901234
HERE_IS_A_SIMPLE_EXAMPLE
EXAMPLE

Naive Algorithm for String Matching

HERE_IS_A_SIMPLE_EXAMPLE

EXAMPLE

From <http://www.cs.utexas.edu/users/moore/best-ideas/string-searching/fstrpos-example.html>

M. Waga (UTokyo)

Naive Algorithm for String Matching

HERE_IS_A_SIMPLE_EXAMPLE



EXAMPLE

From <http://www.cs.utexas.edu/users/moore/best-ideas/string-searching/fstrpos-example.html>

Naive Algorithm for String Matching

HERE_IS_A_SIMPLE_EXAMPLE

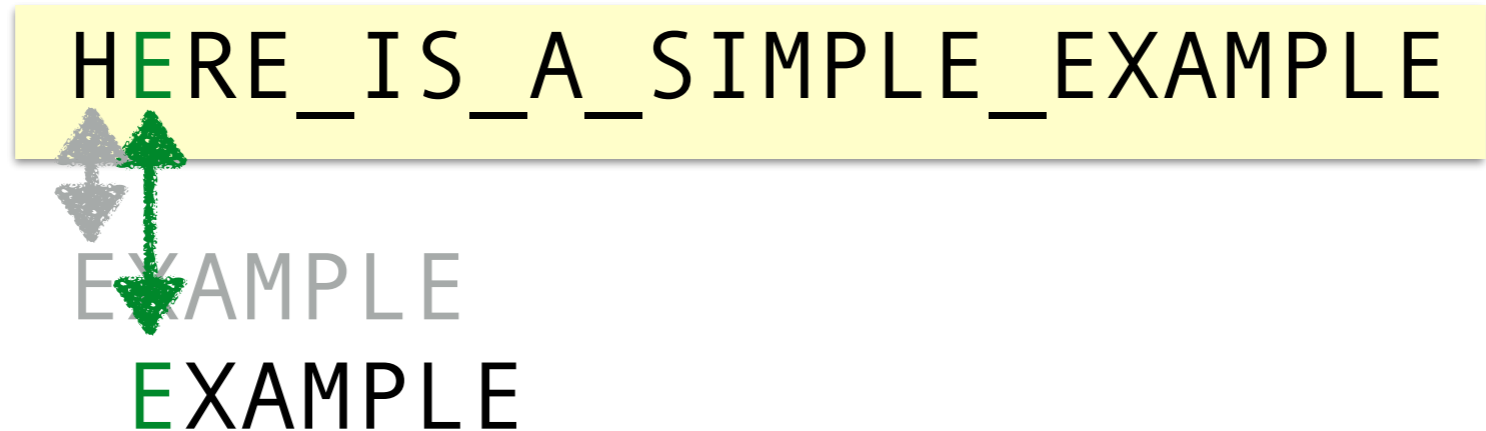


EXAMPLE

EXAMPLE

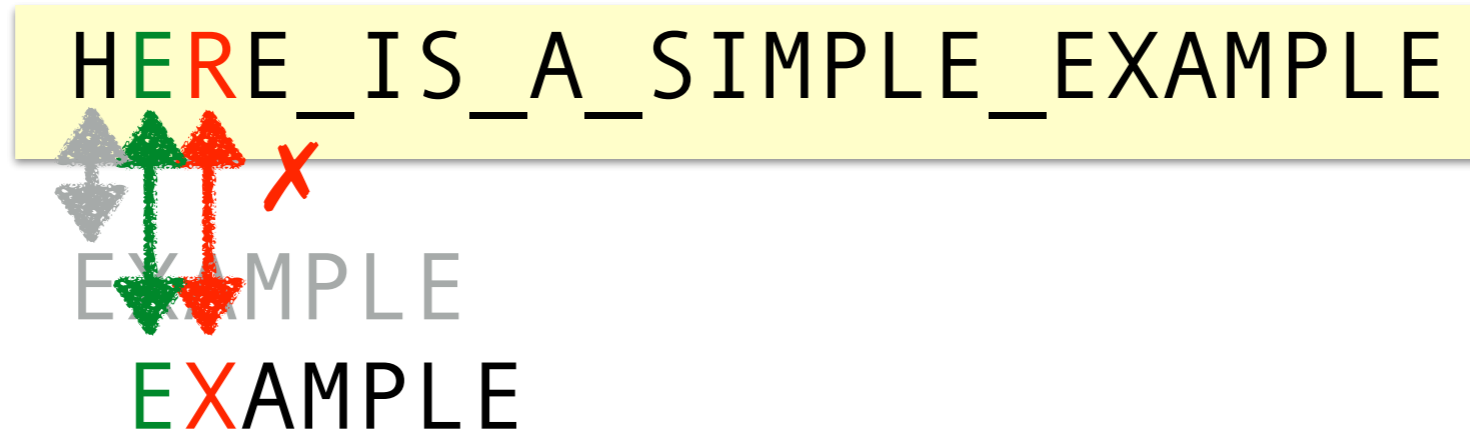
From <http://www.cs.utexas.edu/users/moore/best-ideas/string-searching/fstrpos-example.html>

Naive Algorithm for String Matching



From <http://www.cs.utexas.edu/users/moore/best-ideas/string-searching/fstrpos-example.html>

Naive Algorithm for String Matching



From <http://www.cs.utexas.edu/users/moore/best-ideas/string-searching/fstrpos-example.html>

Naive Algorithm for String Matching

HERE_IS_A_SIMPLE_EXAMPLE



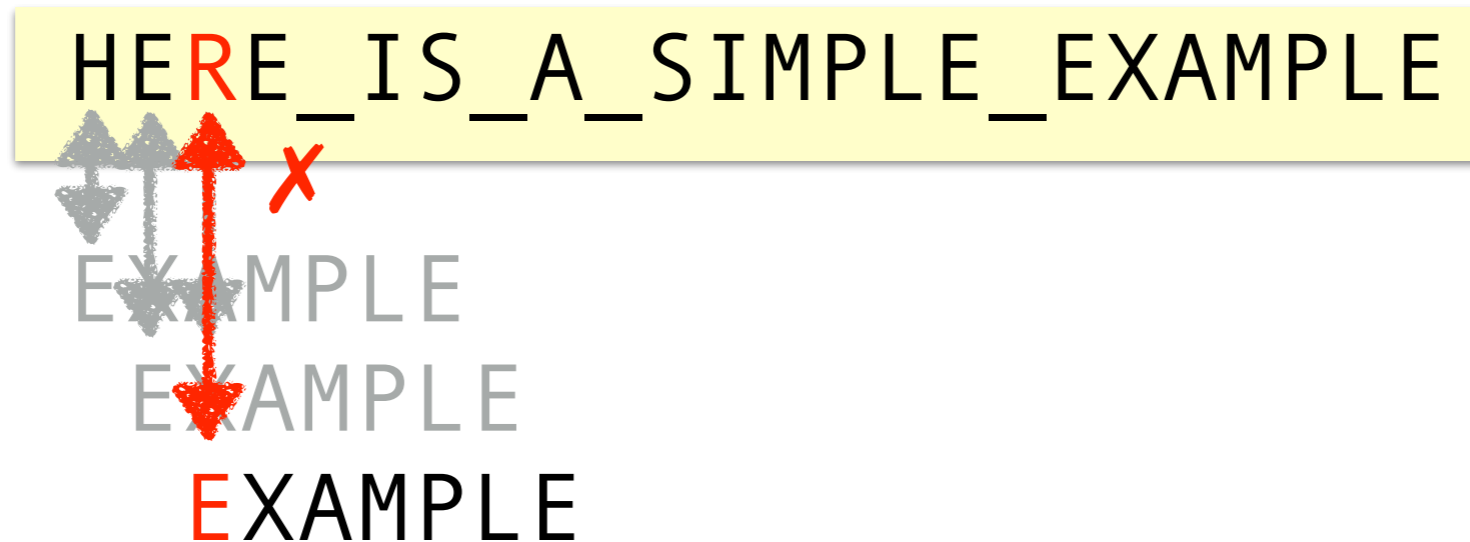
EXAMPLE

EXAMPLE

EXAMPLE

From <http://www.cs.utexas.edu/users/moore/best-ideas/string-searching/fstrpos-example.html>

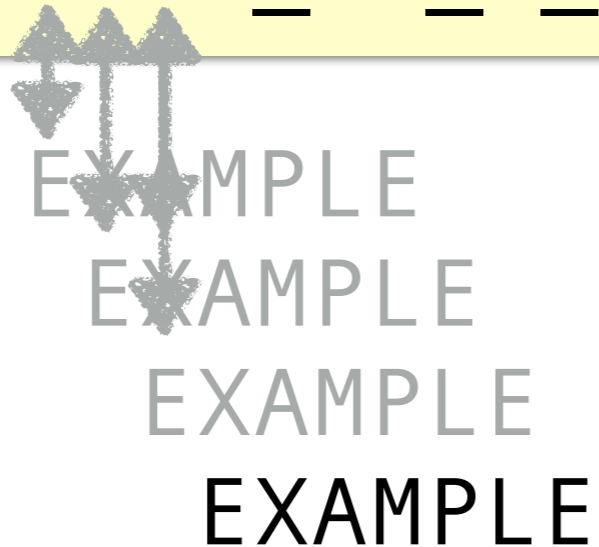
Naive Algorithm for String Matching



From <http://www.cs.utexas.edu/users/moore/best-ideas/string-searching/fstrpos-example.html>

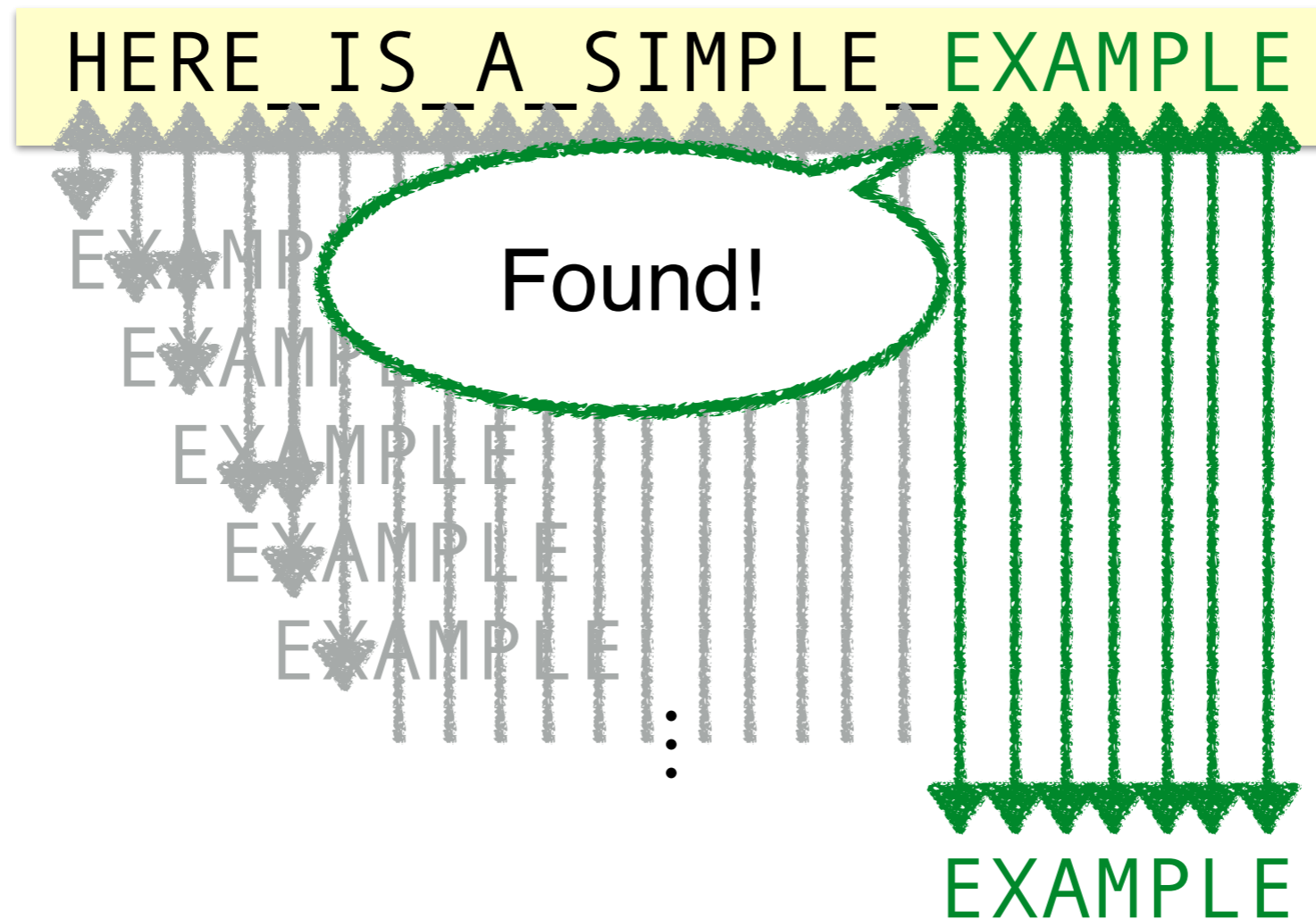
Naive Algorithm for String Matching

HERE_IS_A_SIMPLE_EXAMPLE



From <http://www.cs.utexas.edu/users/moore/best-ideas/string-searching/fstrpos-example.html>

Naive Algorithm for String Matching



From <http://www.cs.utexas.edu/users/moore/best-ideas/string-searching/fstrpos-example.html>

The Boyer-Moore Algorithm

[Boyer & Moore, Commun. ACM '77]

- Optimisation by skipping characters
- SkipValue can be computed beforehand.

Finite

SkipValue : $(\Sigma \times |pat|) \rightarrow \mathbb{Z}_{\geq 0}$

The Boyer-Moore Algorithm

[Boyer & Moore, Commun. ACM '77]

HERE_IS_A_SIMPLE_EXAMPLE

EXAMPLE

From <http://www.cs.utexas.edu/users/moore/best-ideas/string-searching/fstrpos-example.html>

M. Waga (UTokyo)

The Boyer-Moore Algorithm

[Boyer & Moore, Commun. ACM '77]

HERE_I_S_A_SIMPLE_EXAMPLE

EXAMPLE



From <http://www.cs.utexas.edu/users/moore/best-ideas/string-searching/fstrpos-example.html>

M. Waga (UTokyo)

The Boyer-Moore Algorithm

[Boyer & Moore, Commun. ACM '77]

HERE_I_S_A_SIMPLE_EXAMPLE

EXAMPLE

“S” does not appear
in the pattern
(EXAMPLE)

From <http://www.cs.utexas.edu/users/moore/best-ideas/string-searching/fstrpos-example.html>

M. Waga (UTokyo)

The Boyer-Moore Algorithm

[Boyer & Moore, Commun. ACM '77]

HERE_IS_A_SIMPLE_EXAMPLE

EXAMPLE

EXAMPLE

“S” does not appear
in the pattern
(EXAMPLE)

From <http://www.cs.utexas.edu/users/moore/best-ideas/string-searching/fstrpos-example.html>

M. Waga (UTokyo)

The Boyer-Moore Algorithm

[Boyer & Moore, Commun. ACM '77]

HERE_IS_A_SIMPLE_EXAMPLE

EXAMPLE

EXAMPLE

“S” does not appear
in the pattern
(EXAMPLE)

From <http://www.cs.utexas.edu/users/moore/best-ideas/string-searching/fstrpos-example.html>

The Boyer-Moore Algorithm

[Boyer & Moore, Commun. ACM '77]

HERE_IS_A_SIMPLE_EXAMPLE

EXAMPLE

EXAMPLE

“S” does not appear
in the pattern
(EXAMPLE)

“P” appears here

Same!

From <http://www.cs.utexas.edu/users/moore/best-ideas/string-searching/fstrpos-example.html>

The Boyer-Moore Algorithm

[Boyer & Moore, Commun. ACM '77]

HERE_IS_A_SIMPLE_EXAMPLE

EXAMPLE

EXAMPLE

EXAMPLE

“S” does not appear
in the pattern
(EXAMPLE)

“P” appears here

Same!

From <http://www.cs.utexas.edu/users/moore/best-ideas/string-searching/fstrpos-example.html>

The Boyer-Moore Algorithm

[Boyer & Moore, Commun. ACM '77]

HERE_IS_A_SIMPLE_EXAMPLE

EXAMPLE

EXAMPLE

EXAMPLE

“S” does not appear
in the pattern
(EXAMPLE)

“P” appears here

Same!

From <http://www.cs.utexas.edu/users/moore/best-ideas/string-searching/fstrpos-example.html>

The Boyer-Moore Algorithm

[Boyer & Moore, Commun. ACM '77]

HERE_IS_A_SIMPLE_EXAMPLE

EXAMPLE

EXAMPLE

EXAMPLE

“S” does not appear
in the pattern
(EXAMPLE)

“P” appears here

Same!

From <http://www.cs.utexas.edu/users/moore/best-ideas/string-searching/fstrpos-example.html>

The Boyer-Moore Algorithm

[Boyer & Moore, Commun. ACM '77]

HERE_IS_A_SIMPLE_EXAMPLE

EXAMPLE

EXAMPLE

EXAMPLE

“S” does not appear
in the pattern
(EXAMPLE)

“P” appears here

Same!

From <http://www.cs.utexas.edu/users/moore/best-ideas/string-searching/fstrpos-example.html>

The Boyer-Moore Algorithm

[Boyer & Moore, Commun. ACM '77]

HERE_IS_A_SIMPLE_EXAMPLE

EXAMPLE

EXAMPLE

EXAMPLE

“S” does not appear
in the pattern
(EXAMPLE)

“P” appears here

Same!

From <http://www.cs.utexas.edu/users/moore/best-ideas/string-searching/fstrpos-example.html>

The Boyer-Moore Algorithm

[Boyer & Moore, Commun. ACM '77]

HERE_IS_A_SIMPLE_EXAMPLE

EXAMPLE

EXAMPLE

EXAM~~P~~LE

“S” does not appear
in the pattern
(EXAMPLE)

“P” appears here

Same!

From <http://www.cs.utexas.edu/users/moore/best-ideas/string-searching/fstrpos-example.html>

The Boyer-Moore Algorithm

[Boyer & Moore, Commun. ACM '77]

HERE_IS_A_SIMPLE_EXAMPLE

EXAMPLE

EXAMPLE

EXAMPLE

EXAMPLE

Same!

“S” does not appear in the pattern (EXAMPLE)

“I” does not appear in the pattern

“P” appears here

From <http://www.cs.utexas.edu/users/moore/best-ideas/string-searching/fstrpos-example.html>

The Boyer-Moore Algorithm

[Boyer & Moore, Commun. ACM '77]

HERE_IS_A_SIMPLE_EXAMPLE

EXAMPLE

EXAMPLE

EXAMPLE

EXAMPLE

EXAMPLE

"S" does not appear

"I" does not appear in the pattern

We can do better!

Same!

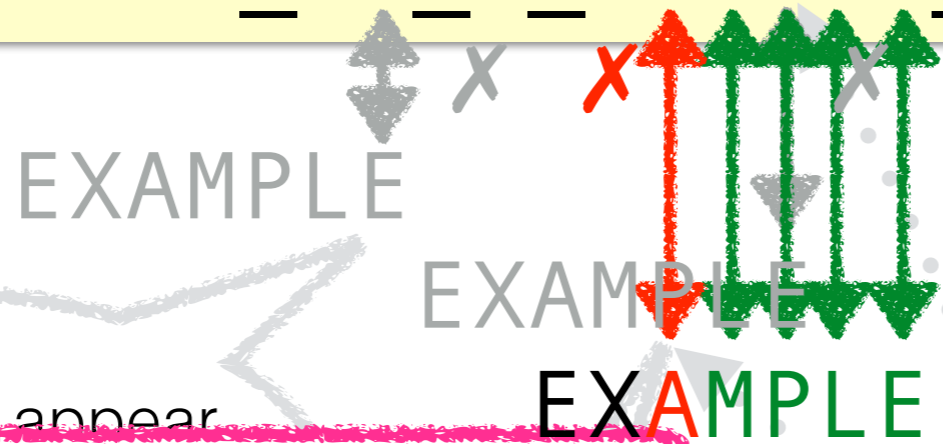
more/best-ideas/string-searching/fstrpos-able.html

M. Waga(UTokyo)

The Boyer-Moore Algorithm

[Boyer & Moore, Commun. ACM '77]

HERE_IS_A_SIMPLE_EXAMPLE



“I” does not appear in the pattern

We can do better!

Given the previous match EXAMPLE,

EXAMPLE

EXAMPLE	×
EXAMPLE	×
EXAMPLE	×
EXAMPLE	×
EXAMPLE	×
EXAMPLE	✓

EXAMPLE

Same!

more/best-ideas/string-searching/fstrpos-able.html

The Boyer-Moore Algorithm

[Boyer & Moore, Commun. ACM '77]

HERE_IS_A_SIMPLE_EXAMPLE

EXAMPLE

EXAMPLE

EXAMPLE

EXAMPLE

EXAMPLE

"S" does not appear

"I" does not appear in the pattern

We can do better!

Given the previous match EXAMPLE,

EXAMPLE

EXAMPLE

EXAMPLE

EXAMPLE

EXAMPLE

EXAMPLE

EXAMPLE

✗

✗

✗

✗

✗

✓

Same!

more/best-ideas/string-searching/fstrpos-ole.html

M. Waga(UTokyo)

The Boyer-Moore Algorithm

[Boyer & Moore, Commun. ACM '77]

HERE_IS_A_SIMPLE_EXAMPLE

EXAMPLE

EXAMPLE

EXAMPLE

EXAMPLE

EXAMPLE

"S" does not appear

"I" does not appear in the pattern

We can do better!

Given the previous match EXAMPLE,

EXAMPLE

EXAMPLE

EXAMPLE

EXAMPLE

EXAMPLE

EXAMPLE

EXAMPLE

✗

✗

✗

✗

✗

✓

more/best-ideas/string-searching/fstrpos-able.html

M. Waga(UTokyo)

The Boyer-Moore Algorithm

[Boyer & Moore, Commun. ACM '77]

HERE_IS_A_SIMPLE_EXAMPLE

EXAMPLE

Same!

"I" does not appear in the pattern

"S" does not appear

We can do better!

Given the previous match EXAMPLE,

EXAMPLE

EXAMPLE ✗

EXAMPLE ✗

EXAMPLE ✗

EXAMPLE ✗

EXAMPLE ✗

EXAMPLE ✓

EXAMPLE

"P" appears here

[more/best-ideas/string-searching/fstrpos-
le.html](http://more/best-ideas/string-searching/fstrpos-
le.html)

M. Waga(UTokyo)

The Boyer-Moore Algorithm

[Boyer & Moore, Commun. ACM '77]

HERE_IS_A_SIMPLE_EXAMPLE

EXAMPLE

EXAMPLE Same!

EXAMPLE

EXAMPLE

EXAMPLE Same!

EXAMPLE

"S" does not appear

"I" does not appear in the pattern

"P" appears here

We can do better!

Given the previous match EXAMPLE,

EXAMPLE

EXAMPLE ✗

EXAMPLE ✗

EXAMPLE ✗

EXAMPLE ✗

EXAMPLE ✗

EXAMPLE ✓

more/best-ideas/string-searching/fstrpos-able.html

M. Waga(UTokyo)

The Boyer-Moore Algorithm

[Boyer & Moore, Commun. ACM '77]

HERE_IS_A_SIMPLE_EXAMPLE

EXAMPLE

EXAMPLE Same!

EXAMPLE

EXAMPLE

EXAMPLE Same!

EXAMPLE

"S" does not appear

"I" does not appear in the pattern

"P" appears here

We can do better!

Given the previous match EXAMPLE,

EXAMPLE

EXAMPLE ✗

EXAMPLE ✗

EXAMPLE ✗

EXAMPLE ✗

EXAMPLE ✗

EXAMPLE ✓

more/best-ideas/string-searching/fstrpos-able.html

M. Waga(UTokyo)

The Boyer-Moore Algorithm

[Boyer & Moore, Commun. ACM '77]

HERE_IS_A_SIMPLE_EXAMPLE

EXAMPLE

EXAMPLE Same!

EXAMPLE

EXAMPLE

EXAMPLE Same!

EXAMPLE

"S" does not appear

" " does not appear in the pattern

"P" appears here

We can do better!

Given the previous match EXAMPLE ,

EXAMPLE

EXAMPLE ✗

EXAMPLE ✗

EXAMPLE ✗

EXAMPLE ✗

EXAMPLE ✗

EXAMPLE ✓

more/best-ideas/string-searching/fstrpos-able.html

M. Waga(UTokyo)

The Boyer-Moore Algorithm

[Boyer & Moore, Commun. ACM '77]

HERE_IS_A_SIMPLE_EXAMPLE

EXAMPLE

EXAMPLE Same!

EXAMPLE

EXAMPLE

EXAMPLE Same!

EXAMPLE

"S" does not appear

" " does not appear in the pattern

"P" appears here

We can do better!

Given the previous match EXAMPLE,

EXAMPLE

EXAMPLE ✗

EXAMPLE ✗

EXAMPLE ✗

EXAMPLE ✗

EXAMPLE ✗

EXAMPLE ✓

more/best-ideas/string-searching/fstrpos-able.html

M. Waga(UTokyo)

The Boyer-Moore Algorithm

[Boyer & Moore, Commun. ACM '77]

HERE_IS_A_SIMPLE_EXAMPLE

EXAMPLE

EXAMPLE Same!

EXAMPLE

EXAMPLE

EXAMPLE Same!

EXAMPLE

"S" does not appear

" " does not appear in the pattern

We can do better!

Given the previous match EXAMPLE,

EXAMPLE

EXAMPLE

✗

EXAMPLE

✗

EXAMPLE

✗

EXAMPLE

✗

EXAMPLE

✗

EXAMPLE

✓

more/best-ideas/string-searching/fstrpos-able.html

"P" appears here

M. Waga(UTokyo)

The Boyer-Moore Algorithm

[Boyer & Moore, Commun. ACM '77]

HERE_IS_A_SIMPLE_EXAMPLE

EXAMPLE

EXAMPLE Same!

EXAMPLE

EXAMPLE

EXAMPLE Same!

EXAMPLE

"S" does not appear

" " does not appear in the pattern

We can do better!

Given the previous match EXAMPLE,

EXAMPLE

EXAMPLE

✗

EXAMPLE

✗

EXAMPLE

✗

EXAMPLE

✗

EXAMPLE

✗

EXAMPLE

✓

more/best-ideas/string-searching/fstrpos-able.html

M. Waga(UTokyo)

"P" appears here

The Boyer-Moore Algorithm

[Boyer & Moore, Commun. ACM '77]

HERE_IS_A_SIMPLE_EXAMPLE

EXAMPLE

EXAMPLE Same!

EXAMPLE

EXAMPLE

EXAMPLE Same!

EXAMPLE

"S" does not appear

" " does not appear in the pattern

We can do better!

Given the previous match EXAMPLE ,

EXAMPLE

EXAMPLE

✗

EXAMPLE

✗

EXAMPLE

✗

EXAMPLE

✗

EXAMPLE

✗

EXAMPLE

✓

more/best-ideas/string-searching/fstrpos-able.html

"P" appears here

M. Waga(UTokyo)

The Boyer-Moore Algorithm

[Boyer & Moore, Commun. ACM '77]

HERE_IS_A_SIMPLE_EXAMPLE

EXAMPLE

EXAMPLE

EXAMPLE

EXAMPLE

EXAMPLE

EXAMPLE

"S" does not appear

" " does not appear in the pattern

We can do better!

Given the previous match EXAMPLE ,

EXAMPLE

EXAMPLE

EXAMPLE

EXAMPLE

EXAMPLE

EXAMPLE

EXAMPLE

✗

✗

✗

✗

✗

✓

Found!

"P" appears here

[more/best-ideas/string-searching/fstrpos-
le.html](http://more/best-ideas/string-searching/fstrpos-
le.html)

M. Waga(UTokyo)

The Boyer-Moore Algorithm

[Boyer & Moore, Commun. ACM '77]

Bottomline:

Move the pattern by possibly more than one.

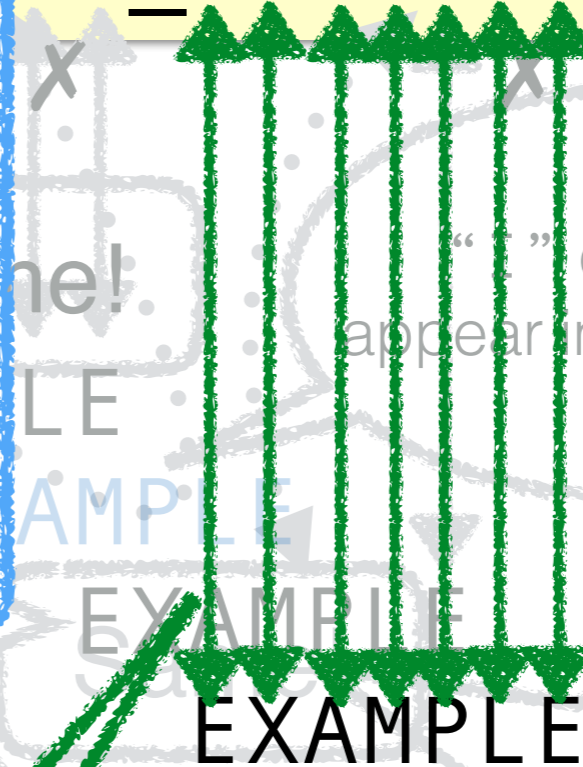
Skip values:

We use two skip values.

EXAMPLE

EXAMPLE	×
EXAMPLE	×
EXAMPLE	×
EXAMPLE	×
EXAMPLE	×
EXAMPLE	✓

LE_EXAMPLE



Found!

[boyer/best-ideas/string-searching/fstrpos-
le.html](https://www.geogebra.org/m/boyer/best-ideas/string-searching/fstrpos-)

M. Waga(UTokyo)

	Target	Pattern	Algorithms
--	--------	---------	------------

String Matching

String

String

- Naive
- Boyer-Moore



Pattern Matching

String

Regular Language

- Naive
- Boyer-Moore

Timed Pattern Matching

Timed Word

Timed Automaton

- Naive
- Boyer-Moore

Contribution

Pattern Matching

Input

Target String

- String str : dbadc dc
- Regular Language L : $dc^*\{ba|dc\}$

Output

Pattern

$$\{(i, j) \mid str(i, j) \in L\} = \{(1, 3), (4, 7)\}$$

$$str(1, 3) = dba \in L, str(4, 7) = dcd \in L$$

Naive Algorithm for Pattern Matching

d b a d c d c

d $\in? L(dc^*\{ba|dc\})$

Naive Algorithm for Pattern Matching

d b a d c d c

d $\in? L(dc^* \{ba | dc\})$

d b $\in? L(dc^* \{ba | dc\})$

Naive Algorithm for Pattern Matching

d b a d c d c

d

d b

d b a ✓

Found!

ε? $L(dc^* \{ba | dc\})$

ε? $L(dc^* \{ba | dc\})$

ε? $L(dc^* \{ba | dc\})$

Naive Algorithm for Pattern Matching

d b a d c d c

d							$\epsilon?$	$L(dc^* \{ba dc\})$
d	b						$\epsilon?$	$L(dc^* \{ba dc\})$
d	b	a					$\epsilon?$	$L(dc^* \{ba dc\})$
							\vdots	
d	b	a	d	c	d	c	$\epsilon?$	$L(dc^* \{ba dc\})$

Found!

Naive Algorithm for Pattern Matching

d b a d c d c

d							$\epsilon?$	$L(dc^* \{ba dc\})$
d	b						$\epsilon?$	$L(dc^* \{ba dc\})$
d	b	a					$\epsilon?$	$L(dc^* \{ba dc\})$
							\vdots	
d	b	a	d	c	d	c	$\epsilon?$	$L(dc^* \{ba dc\})$
	b						$\epsilon?$	$L(dc^* \{ba dc\})$

Found!

Naive Algorithm for Pattern Matching

d b a d c d c

d							$\epsilon?$	$L(dc^* \{ba dc\})$
d	b						$\epsilon?$	$L(dc^* \{ba dc\})$
d	b	a					$\epsilon?$	$L(dc^* \{ba dc\})$
							\vdots	
d	b	a	d	c	d	c	$\epsilon?$	$L(dc^* \{ba dc\})$
	b						$\epsilon?$	$L(dc^* \{ba dc\})$
	b	a					$\epsilon?$	$L(dc^* \{ba dc\})$

Found!

Naive Algorithm for Pattern Matching

d b a d c d c

d							ε?	$L(dc^* \{ba dc\})$
d	b						ε?	$L(dc^* \{ba dc\})$
d	b	a					ε?	$L(dc^* \{ba dc\})$
							⋮	
d	b	a	d	c	d	c	ε?	$L(dc^* \{ba dc\})$
	b						ε?	$L(dc^* \{ba dc\})$
	b	a					ε?	$L(dc^* \{ba dc\})$
							⋮	
			d	c	d	c	ε?	$L(dc^* \{ba dc\})$

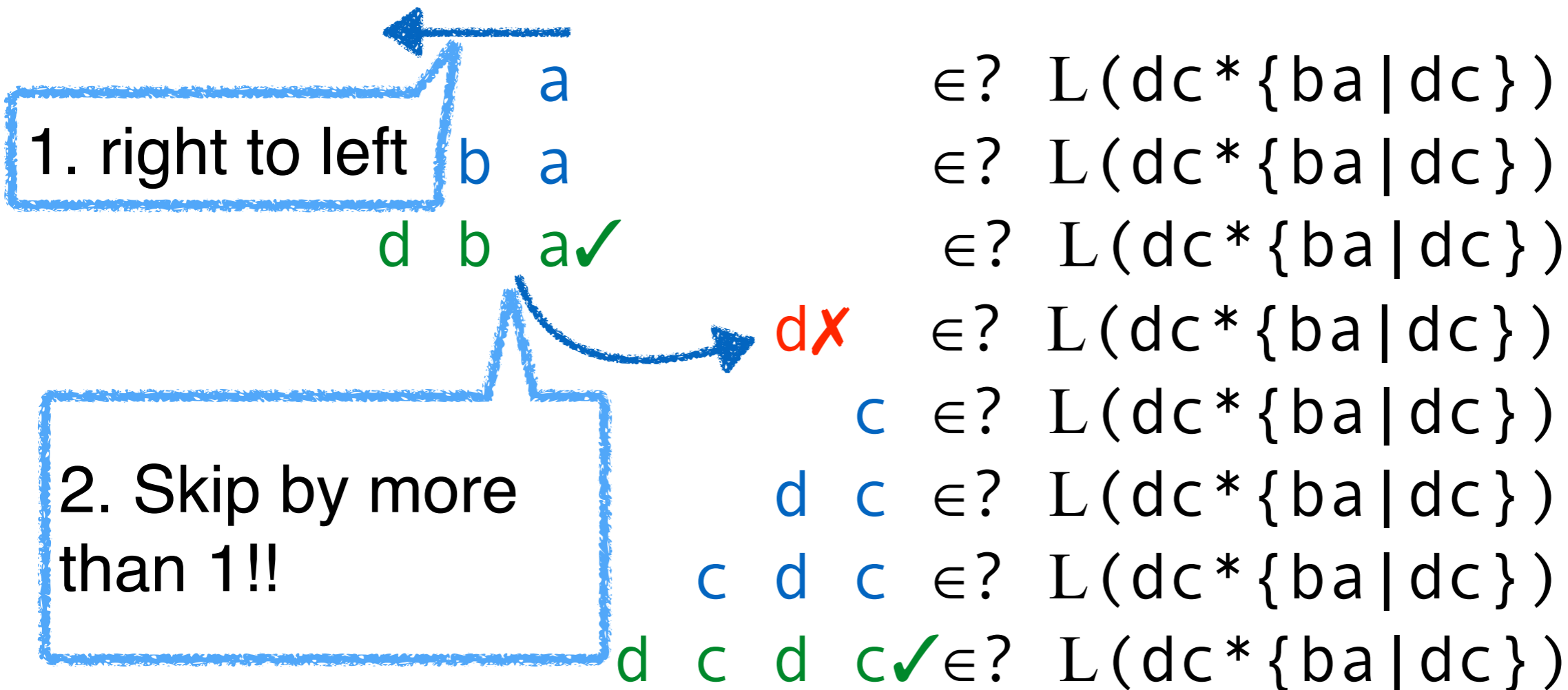
Found!

Found!

The Boyer-Moore type Algorithm for Pattern Matching

[Watson & Watson, Sci. Comput. Program. '03]

d b a d c d c



The Boyer-Moore type Algorithm for Pattern Matching

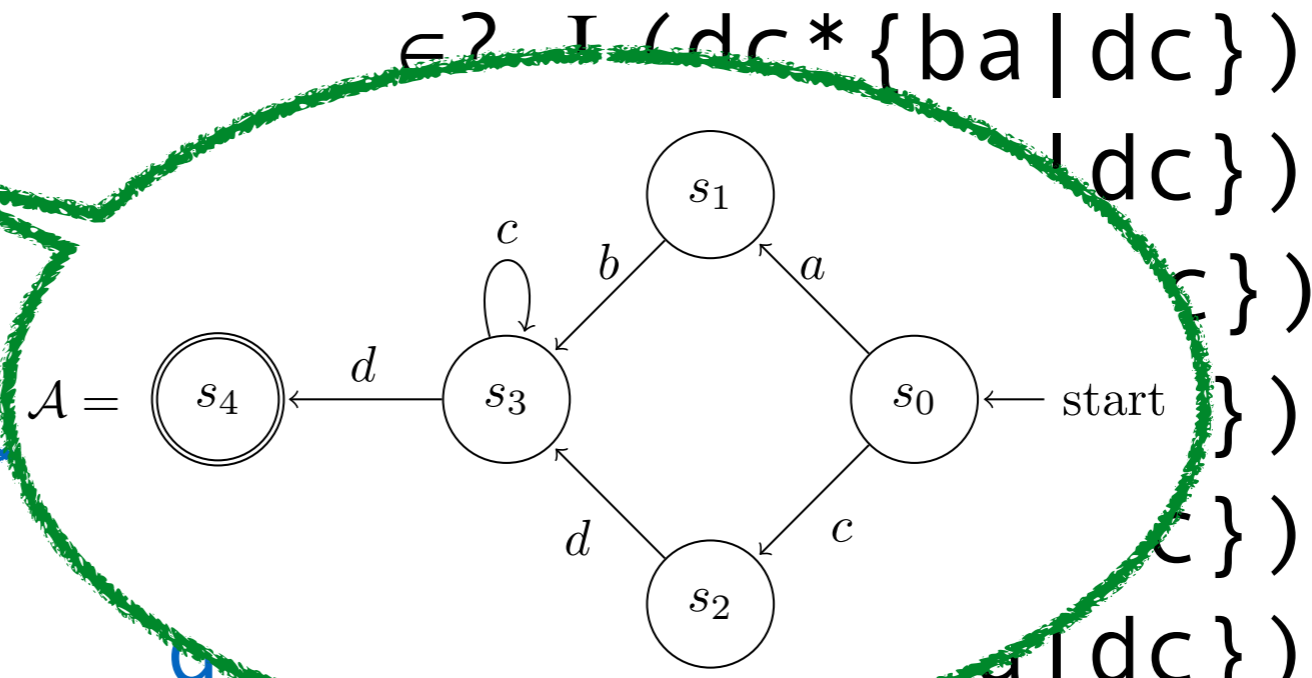
[Watson & Watson, Sci. Comput. Program. '03]

d b a d c d c

1. right to left

a
b a
d b a ✓

2. Skip by more than 1!!



a | dc })
c d c $\epsilon? L(dc^* \{ba|dc\})$
d c d c ✓ $\epsilon? L(dc^* \{ba|dc\})$

The Boyer-Moore type Algorithm for Pattern Matching

[Watson & Watson, Sci. Comput. Program. '03]

d b a d c d c

1. right to left

a
b a
d b a ✓

$\epsilon? L(dc^* \{ba|dc\})$
 $\epsilon? L(dc^* \{ba|dc\})$
 $\epsilon? L(dc^* \{ba|dc\})$

2. Skip by more than 1!!

Similar to the SkipValue in the Boyer-Moore algorithm for string matching.

c d
d c d c ✓

The Boyer-Moore type Algorithm for Pattern Matching

[Watson & Watson, Sci. Comput. Program. '03]

For string matching:

Finite

$$\text{SkipValue} : (\Sigma \times |pat|) \rightarrow \mathbb{Z}_{\geq 0}$$

We cannot
precompute
SkipValue!

For pattern matching:

Infinite!

$$\text{SkipValue} : (\Sigma \times \sup \{ |w| \mid w \in L \}) \rightarrow \mathbb{Z}_{\geq 0}$$

The Boyer-Moore type Algorithm for Pattern Matching

[Watson & Watson, Sci. Comput. Program. '03]

For **string** matching:

Finite

$$\text{SkipValue} : (\Sigma \times |pat|) \rightarrow \mathbb{Z}_{\geq 0}$$

We cannot
precompute
SkipValue!

For **pattern** matching:

Infinite!

~~$$\text{SkipValue} : (\Sigma \times \sup\{|w| \mid w \in L\}) \rightarrow \mathbb{Z}_{> 0}$$~~

Finite

$$\text{SkipValue} : (\Sigma \times \{\text{states of the pattern autom.}\}) \rightarrow \mathbb{Z}_{\geq 0}$$

Timed Pattern Matching with Boyer-Moore type Algorithm

Contribution

Timed Word

[Alur & Dill, Theor. Comput. Sci. '94]

Target String

Def.

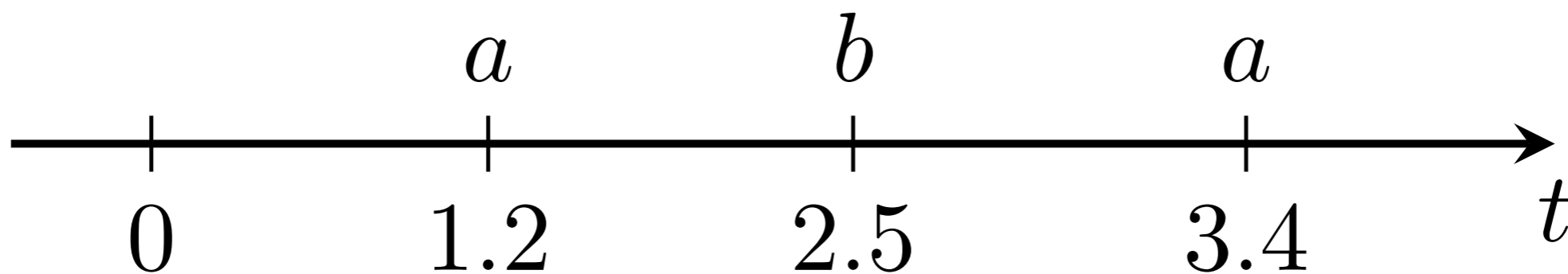
A *timed word* is

$$((\sigma_1, \tau_1), (\sigma_2, \tau_2), \dots, (\sigma_n, \tau_n))$$

where $\sigma_1, \sigma_2, \dots, \sigma_n \in \Sigma$, and

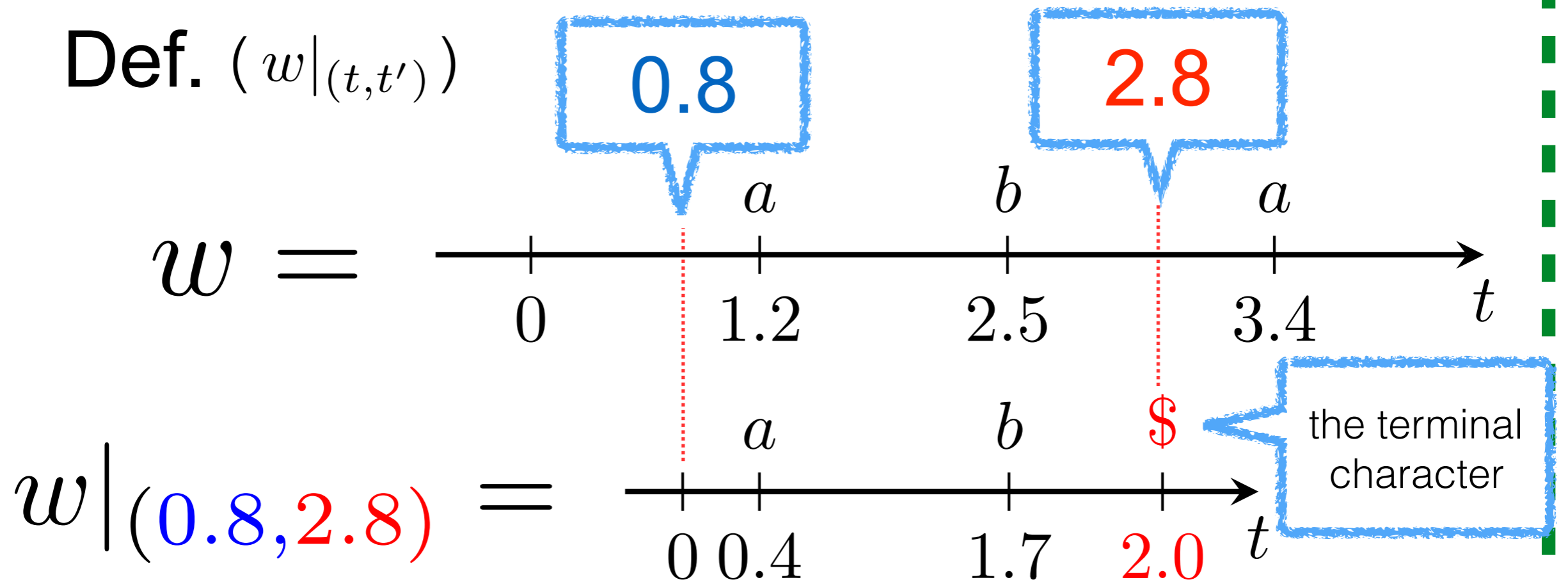
$\tau_1, \tau_2, \dots, \tau_n \in \mathbb{R}_{>0}$ such that

$$\tau_1 < \tau_2 < \dots < \tau_n$$



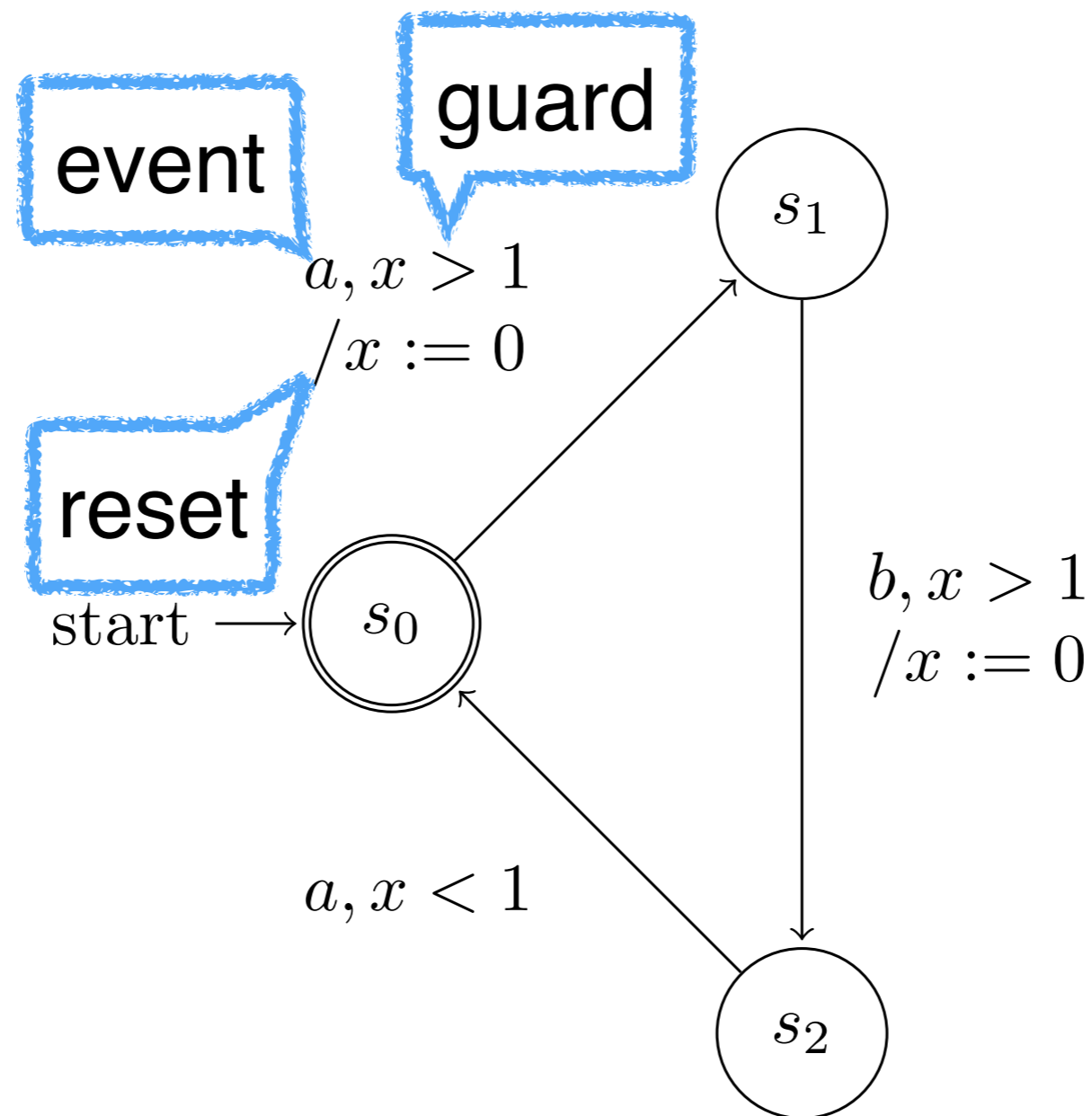
Timed Word Segment

Def. $(w|_{(t,t')})$



Timed Automata Pattern

[Alur & Dill, Theor. Comput. Sci. '94]

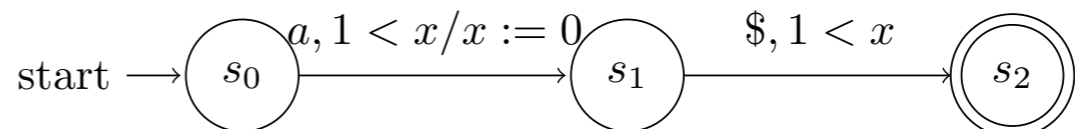
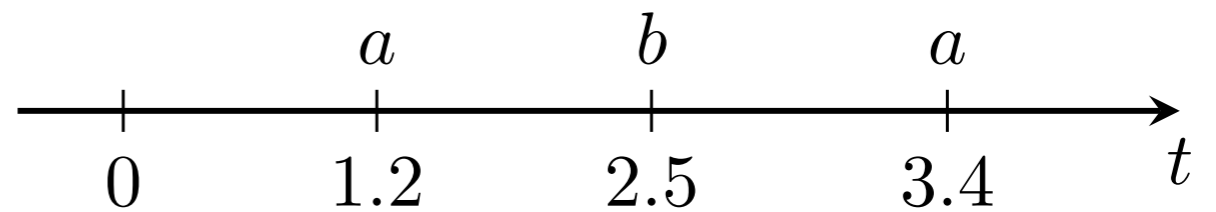


Timed Pattern Matching

Input Target String

- timed word w

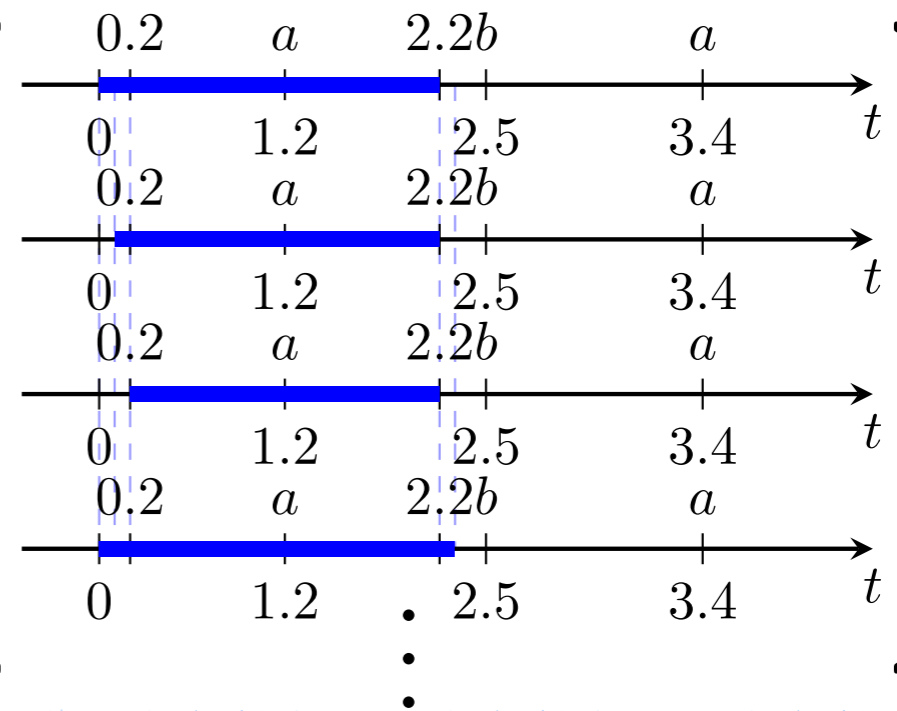
- timed automaton \mathcal{A}



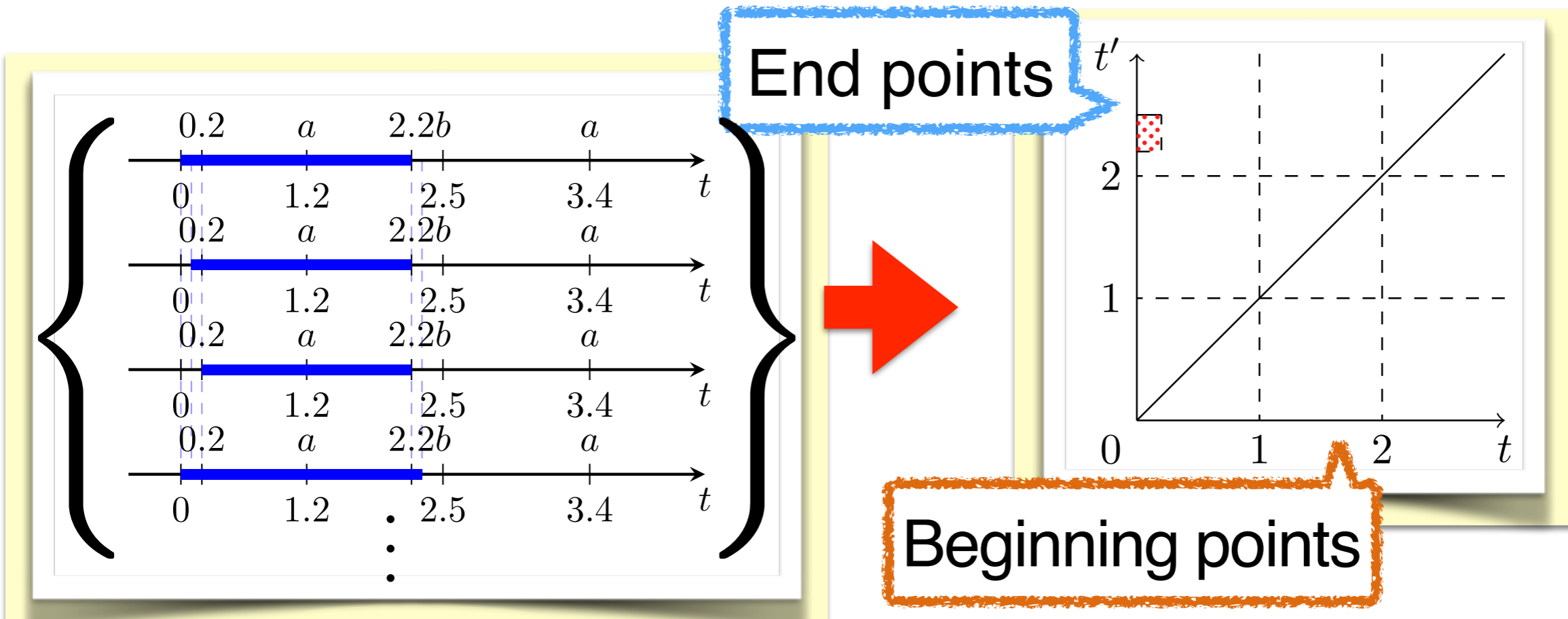
Output

Pattern

$$\{(t, t') \mid t \mid (t, t') \in L(\mathcal{A})\}$$

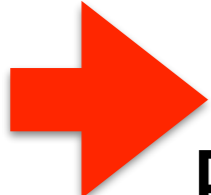


Timed Pattern Matching



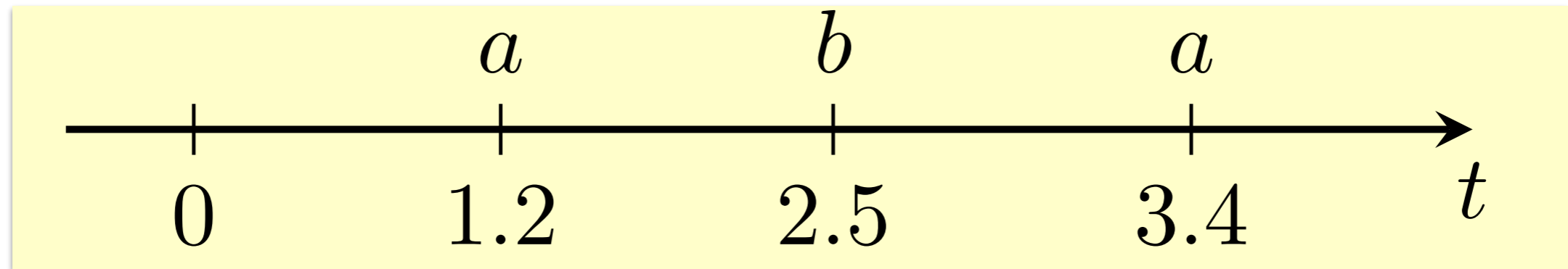
Thm.

- The answer of timed pattern matching for the spec.
- written in timed automaton can be represented by finite
- union of intervals of $t, t', t' - t$

	Target	Pattern	Algorithms
String Matching	String	String	<ul style="list-style-type: none"> •Naive •Boyer-Moore
Pattern Matching	String	Regular Language	<ul style="list-style-type: none"> •Naive •Boyer-Moore
 Timed Pattern Matching	Timed Word	Timed Automaton	<ul style="list-style-type: none"> •Naive •Boyer-Moore

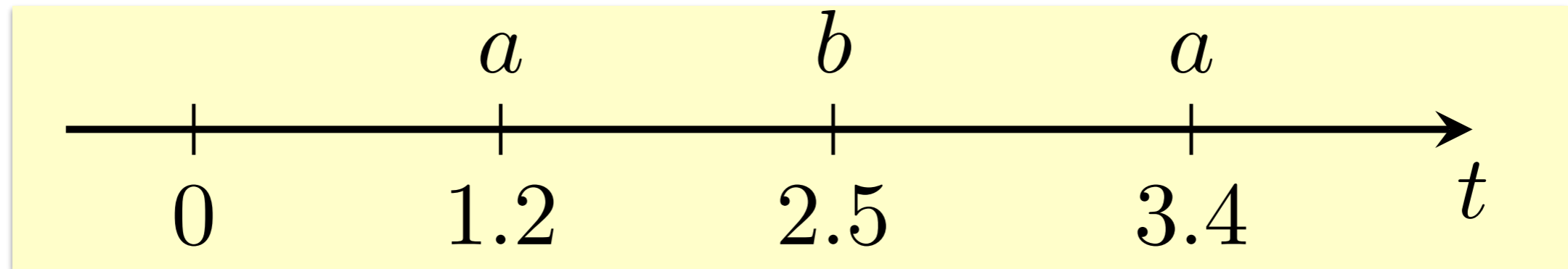
Contribution

Naive Algorithm for Timed Pattern Matching



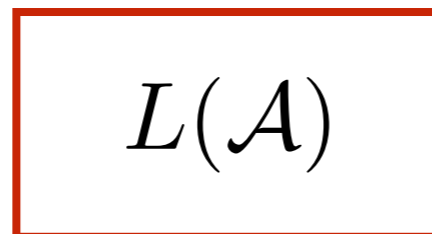
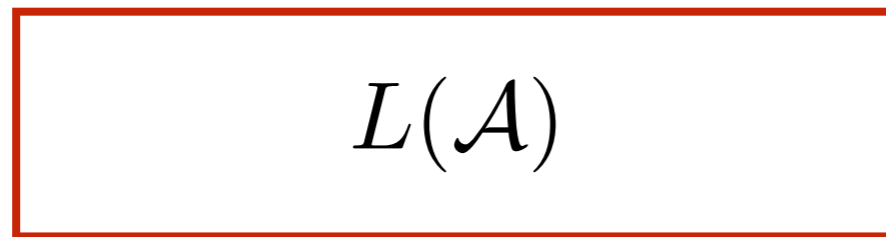
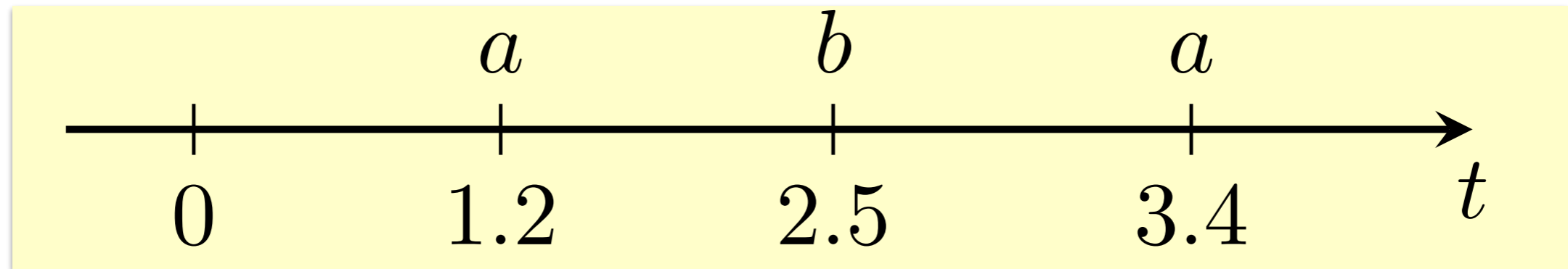
$L(\mathcal{A})$

Naive Algorithm for Timed Pattern Matching

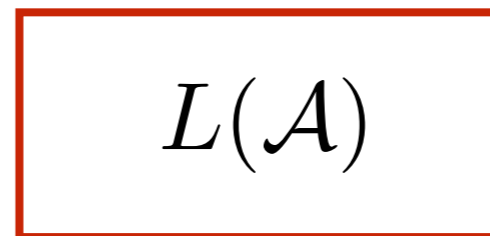
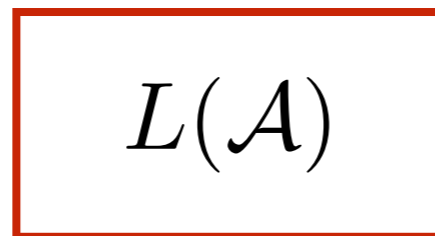
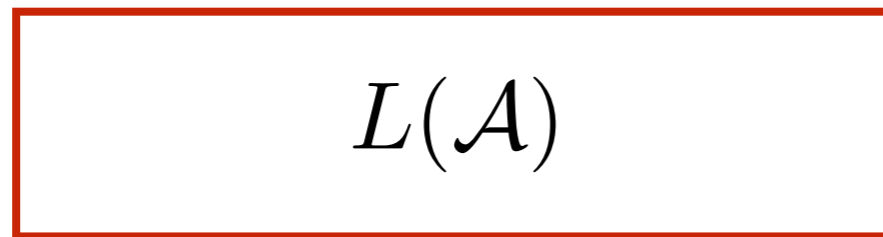
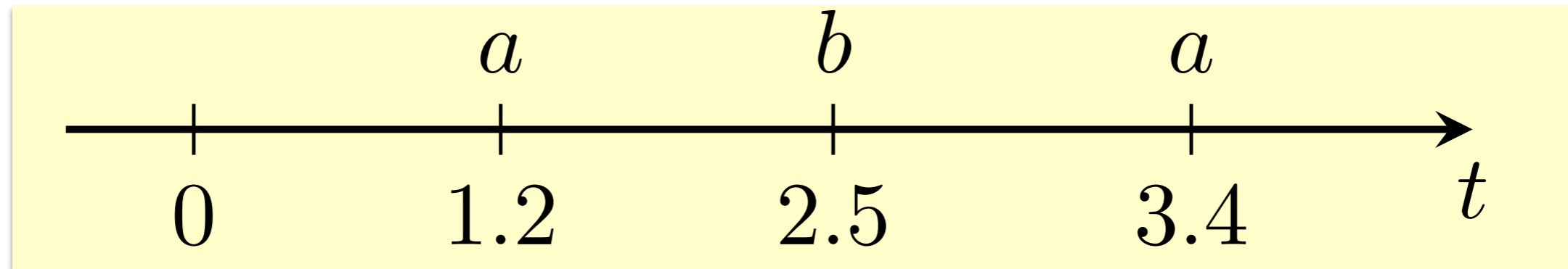


$L(\mathcal{A})$

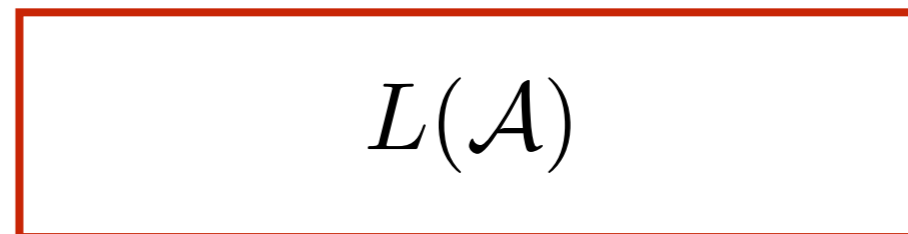
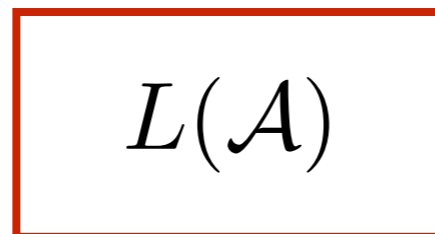
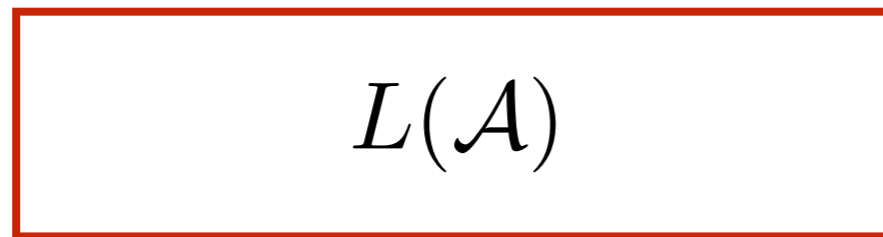
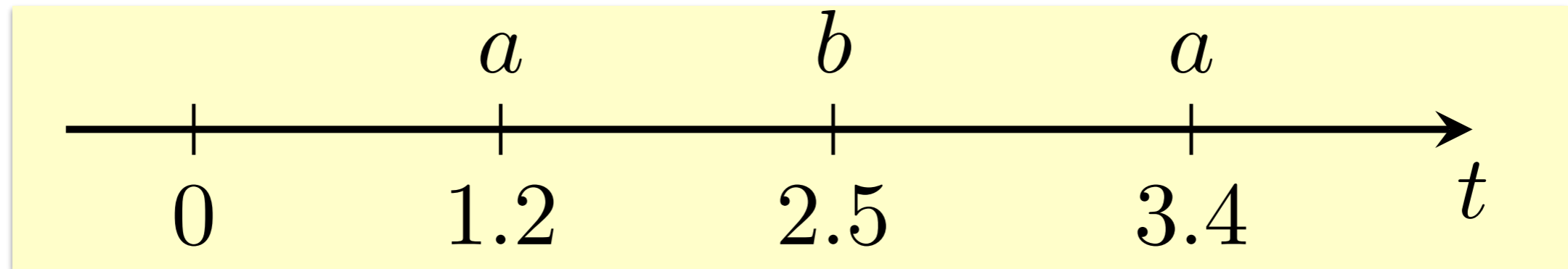
Naive Algorithm for Timed Pattern Matching



Naive Algorithm for Timed Pattern Matching

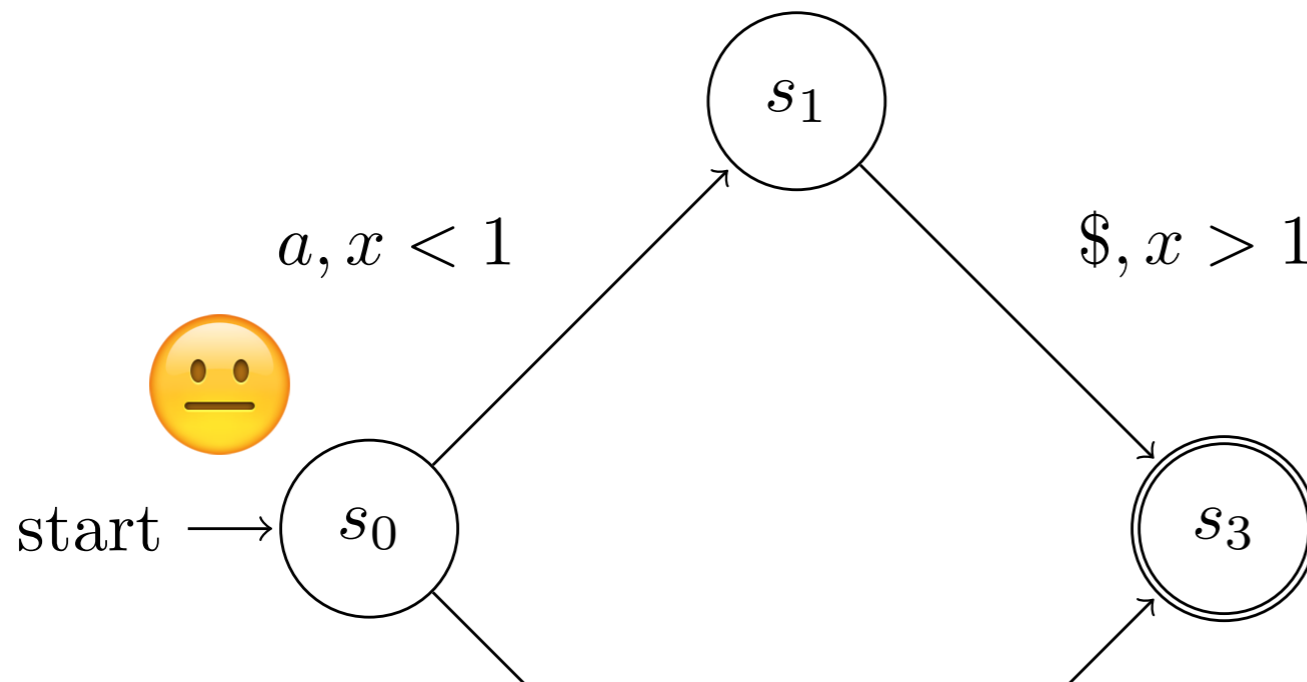
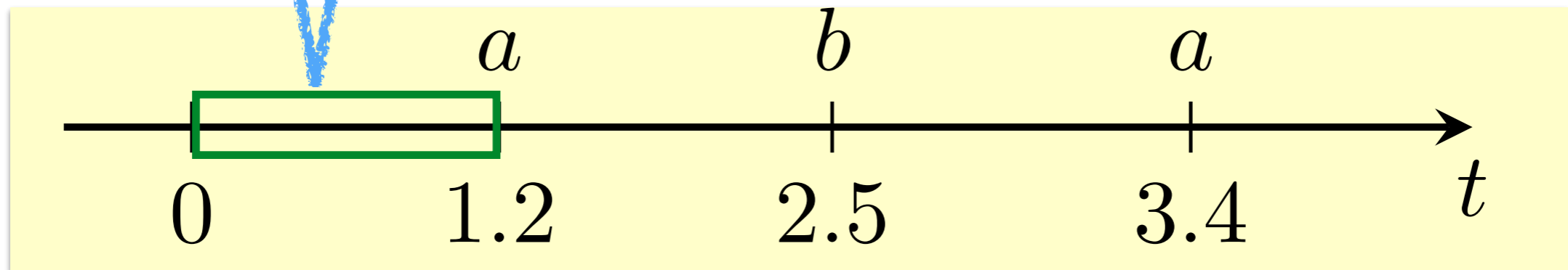


Naive Algorithm for Timed Pattern Matching



Naive Algorithm for Timed Pattern Matching

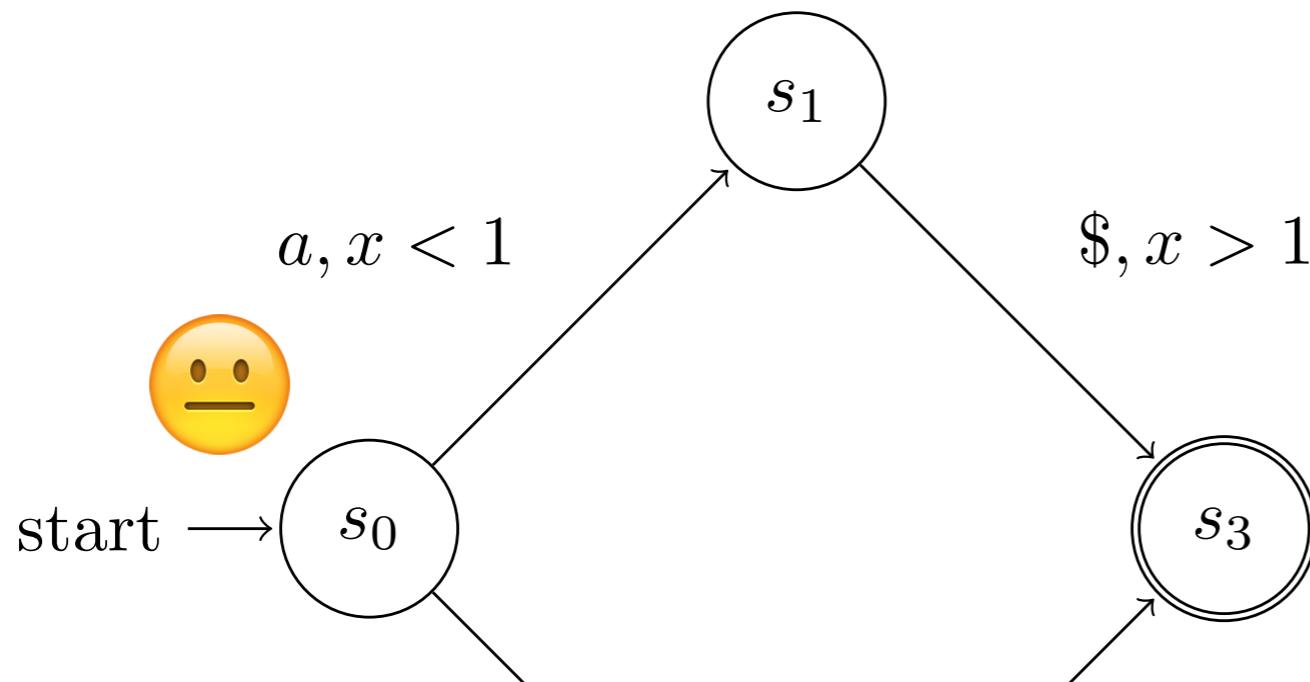
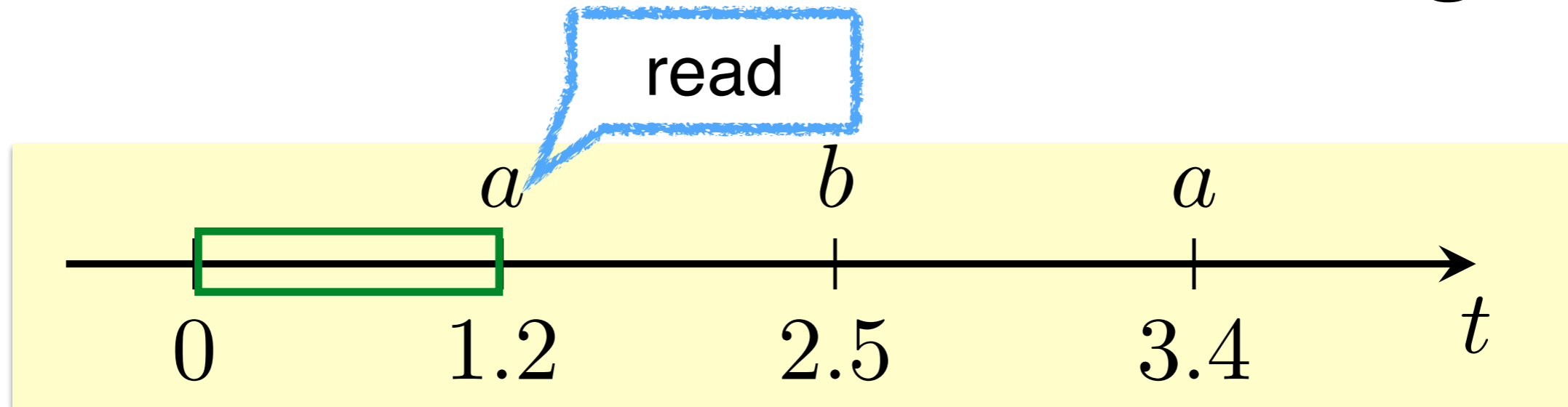
Assume any point is OK



Constraint

$$0 \leq t < 1.2$$

Naive Algorithm for Timed Pattern Matching

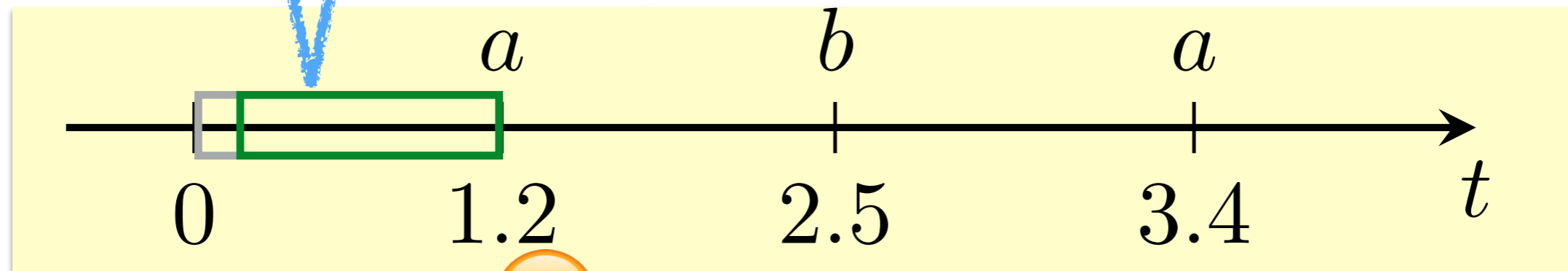


Constraint

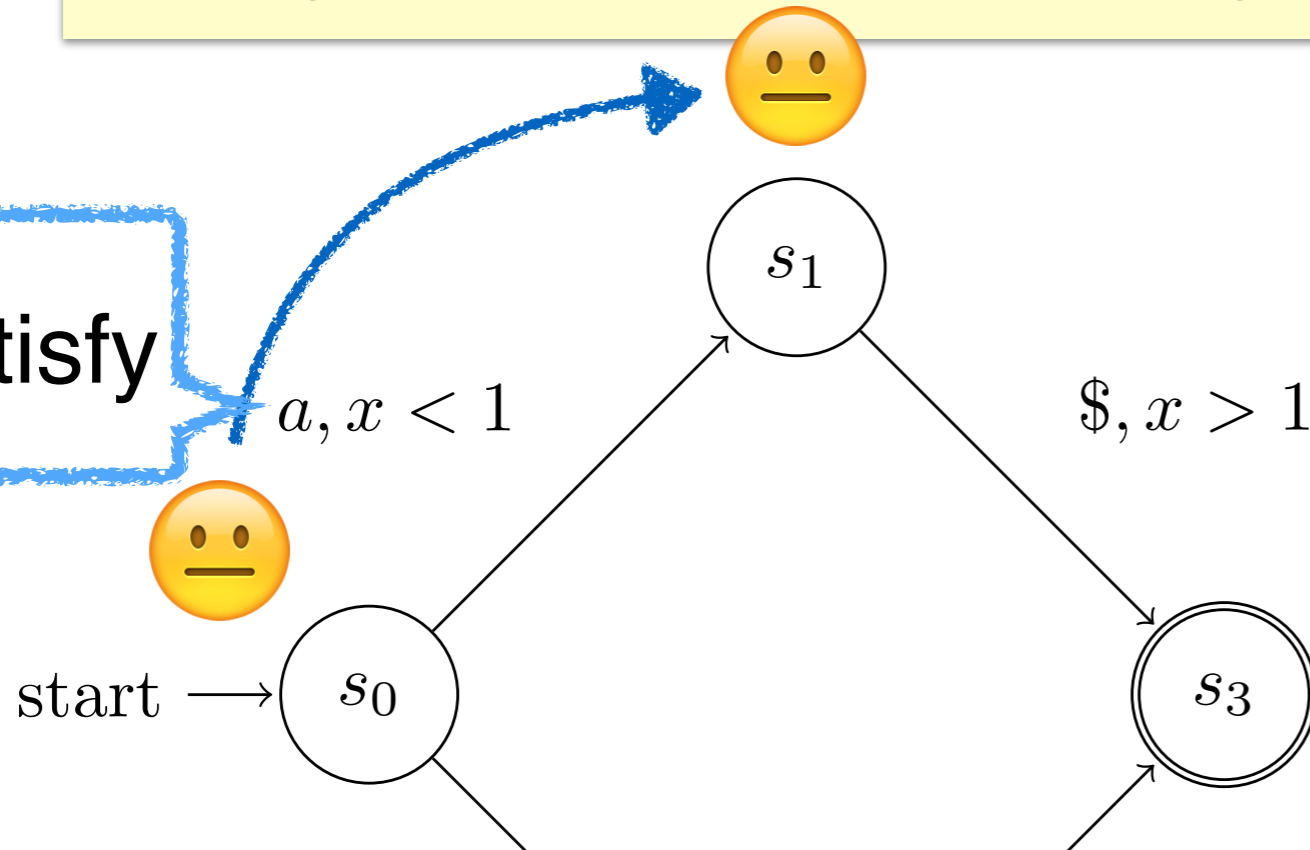
$$0 \leq t < 1.2$$

Naive Algorithm for Timed Pattern Matching

narrow the constraint



satisfy

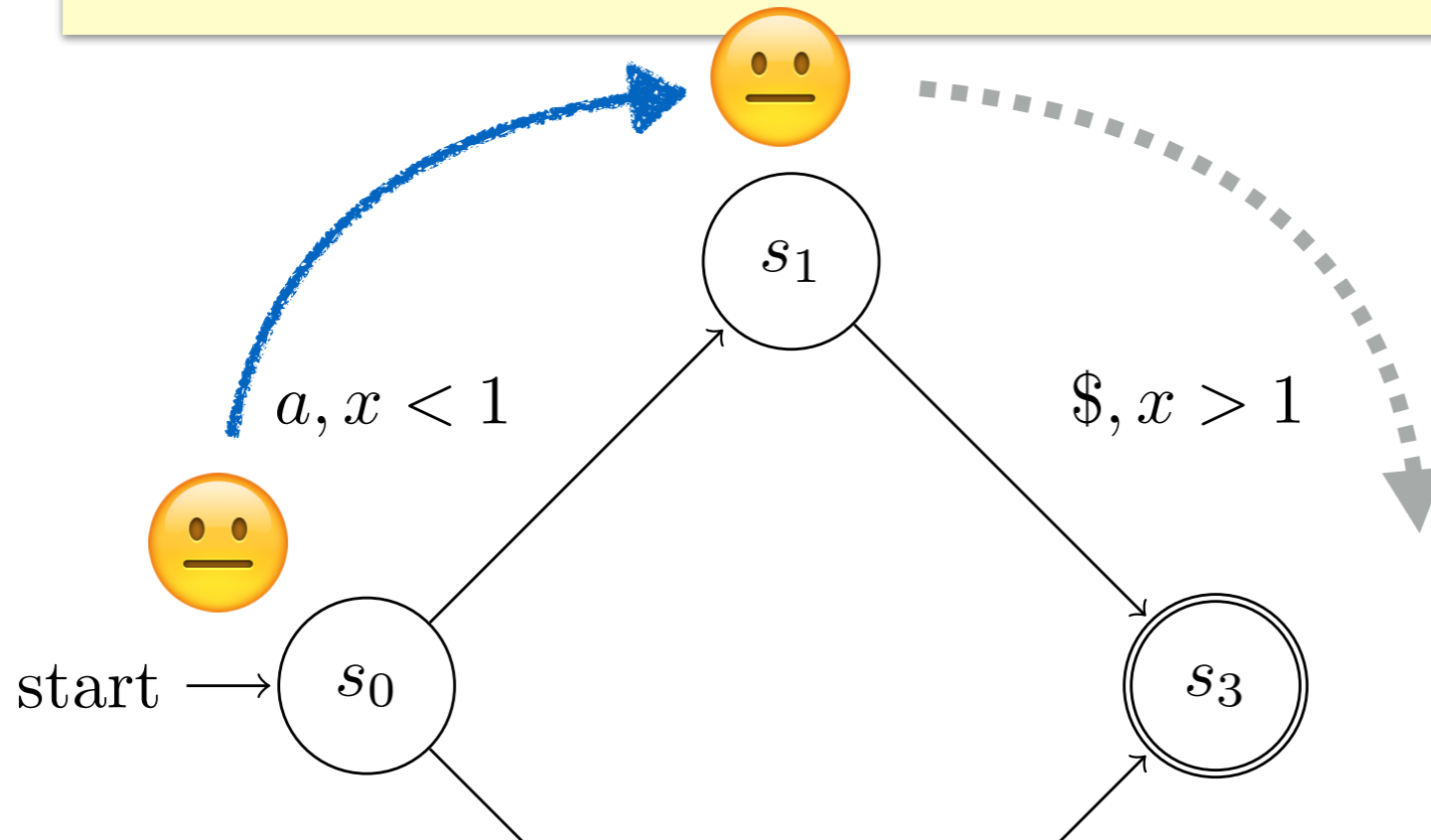
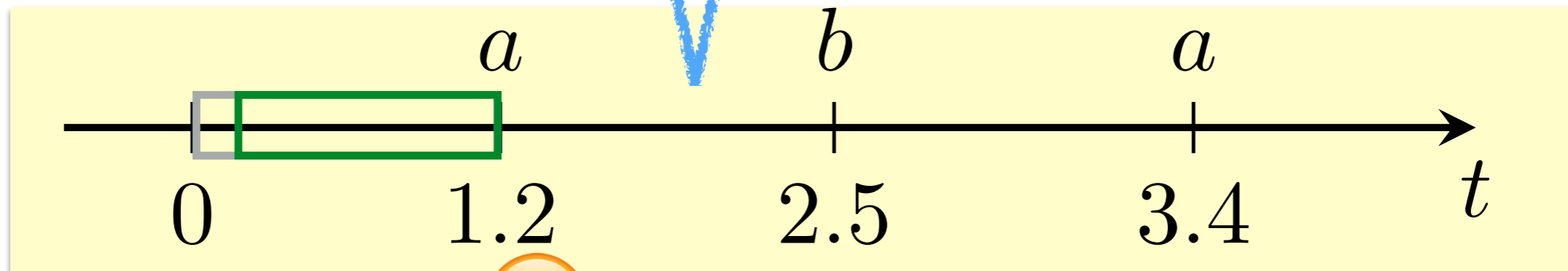


Constraint

$$0.2 < t < 1.2$$

Naive Algorithm for Timed Pattern Matching

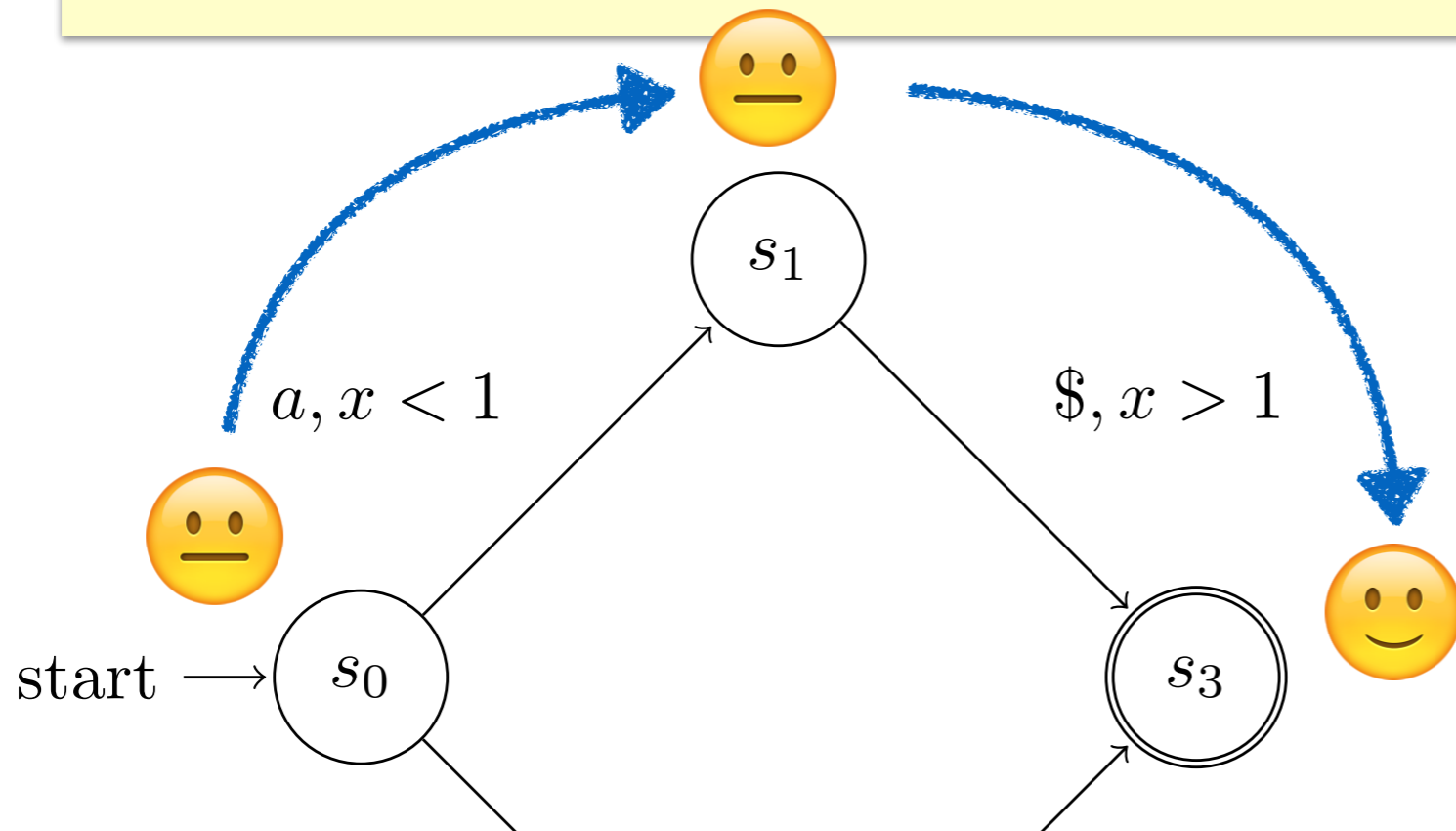
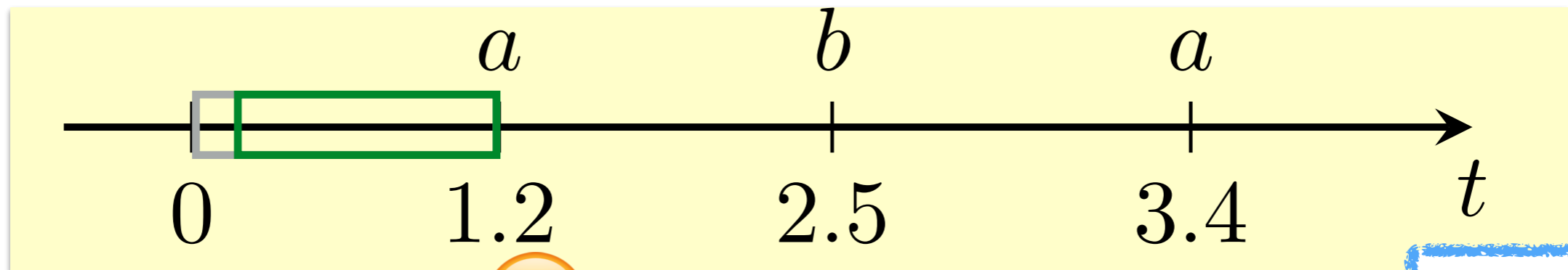
try to insert \$ (terminate character)



Constraints

$$\begin{cases} 0.2 < t < 1.2 \\ 1.2 < t' \leq 2.5 \\ 1 < x = t' - t \end{cases}$$

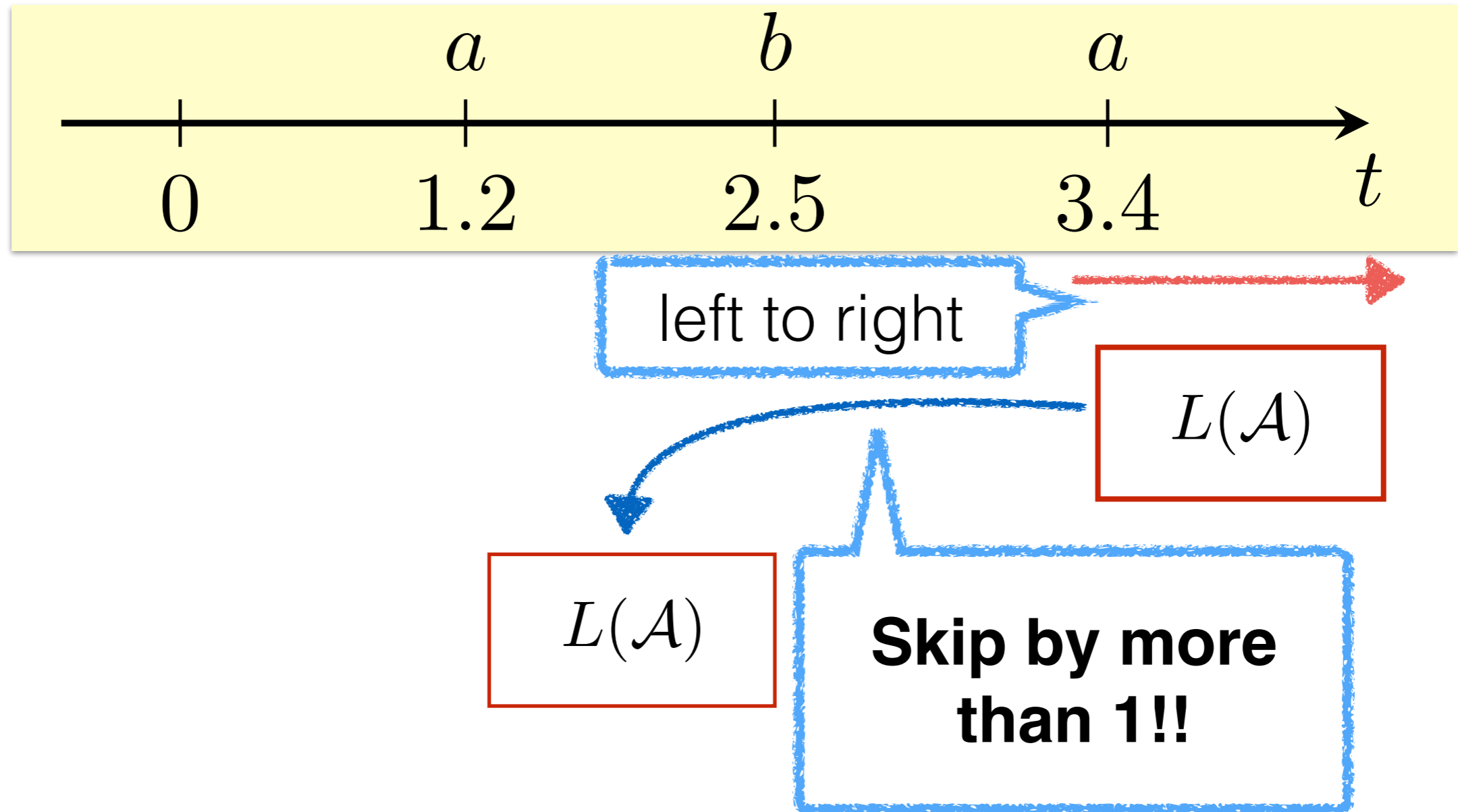
Naive Algorithm for Timed Pattern Matching



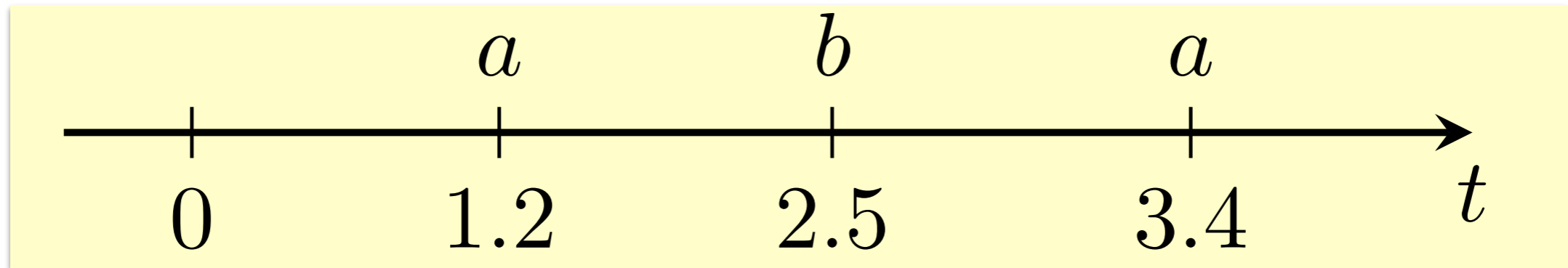
Answer

$$\begin{cases} 0.2 < t < 1.2 \\ 1.2 < t' \leq 2.5 \\ 1 < t' - t \end{cases}$$

Boyer-Moore Type Algorithm for Timed Pattern Matching



Boyer-Moore Type Algorithm for Timed Pattern Matching



left to right 

$L(\mathcal{A})$

Similar to the SkipValue in the Boyer-Moore type algorithm for pattern matching.

$L(\mathcal{A})$

Skip by more than 1!!

Boyer-Moore Type Algorithm for Timed Pattern Matching

Finite

SkipValue : $(\Sigma \times \{\text{states of the pattern autom.}\}) \rightarrow \mathbb{Z}_{\geq 0}$

Difficulty: Uncountably many
comparison of
timed words.

Idea: Representation by a run of a region automaton

String Matching

EXAMPLE



EXAMPLE



EXAMPLE



EXAMPLE



EXAMPLE



EXAMPLE



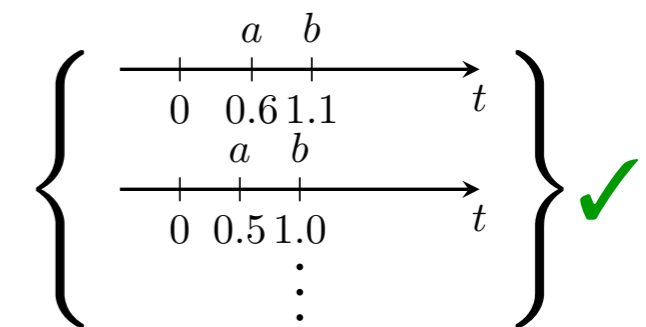
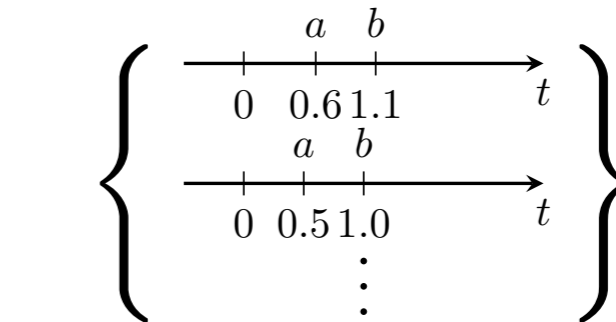
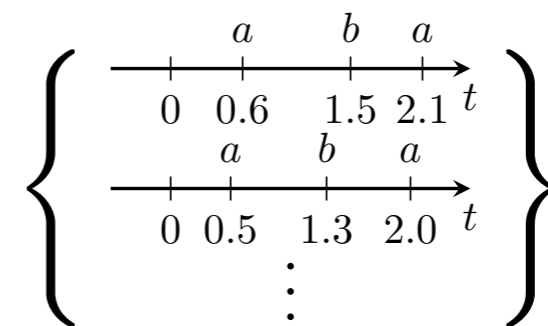
EXAMPLE



Boyer-Moore Type Algorithm for Timed Pattern Matching

Skip Value : $(\Sigma \times \{\text{states of the p}\})$

Timed Pattern Matching



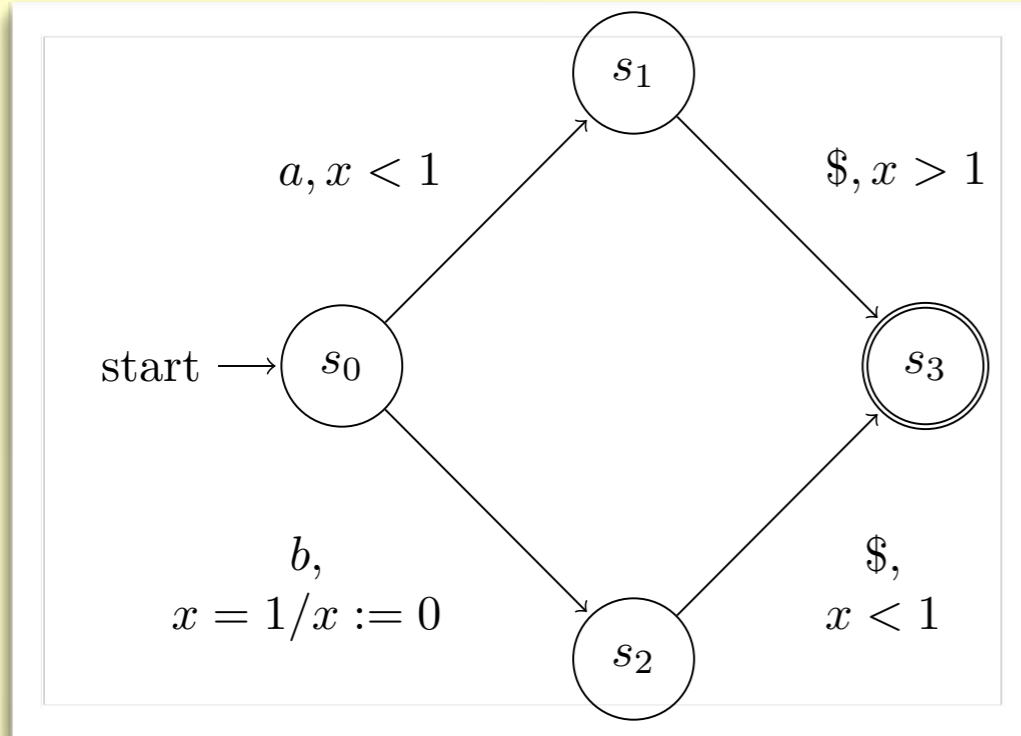
Difficulty: Uncountably many comparison of timed words.

Idea: Representation by a run of a region automaton

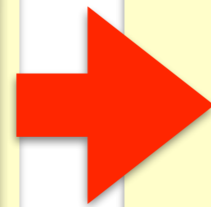
Region Automata

[Alur & Dill, Theor. Comput. Sci. '94]

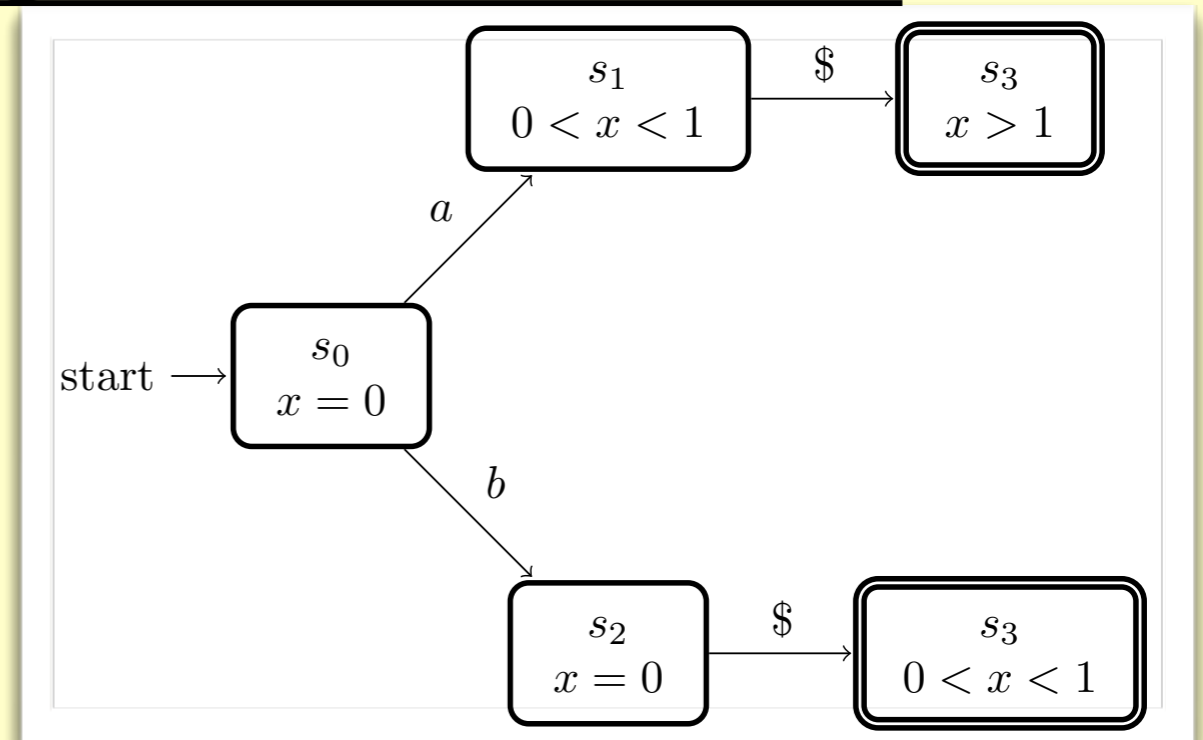
Timed Automaton



States \times Clock Values
is **infinite**



Region Automaton

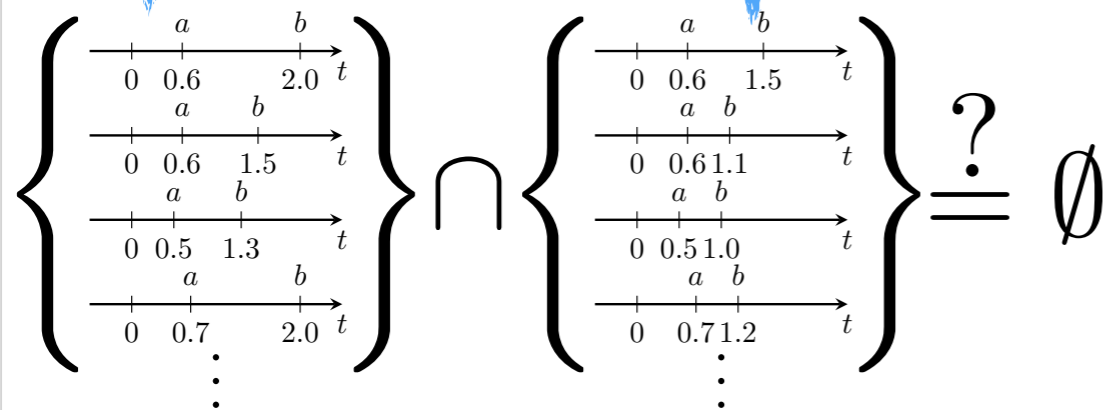


Regions = Clock Values / \sim
RegionStates = States \times Regions
is **finite!**

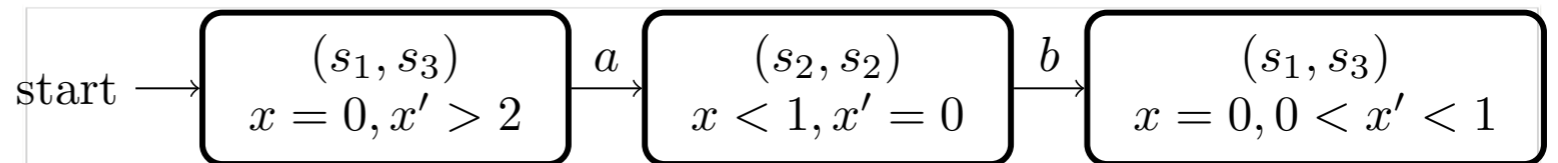
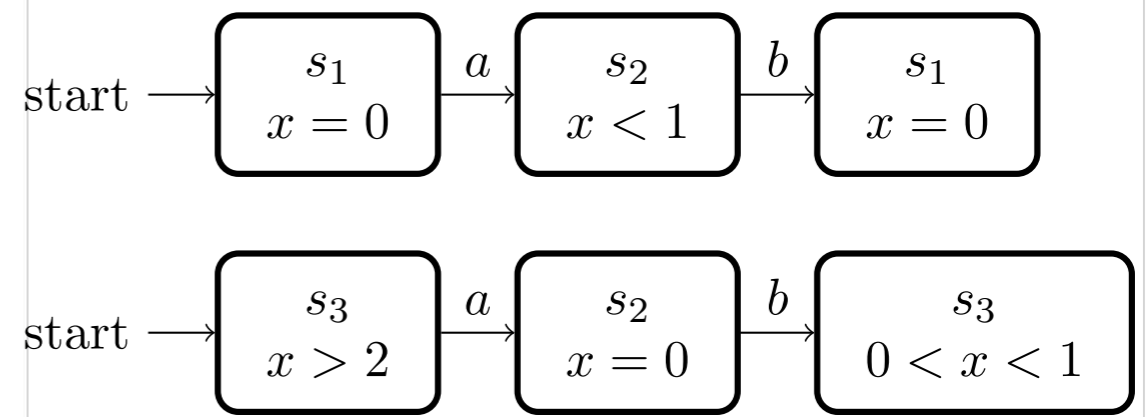
Skip Value

over approx. of
accepting
words (shifted)

over approx.
of read words



compare



is a run of $R(\mathcal{A} \times \mathcal{A})$?

Experiments

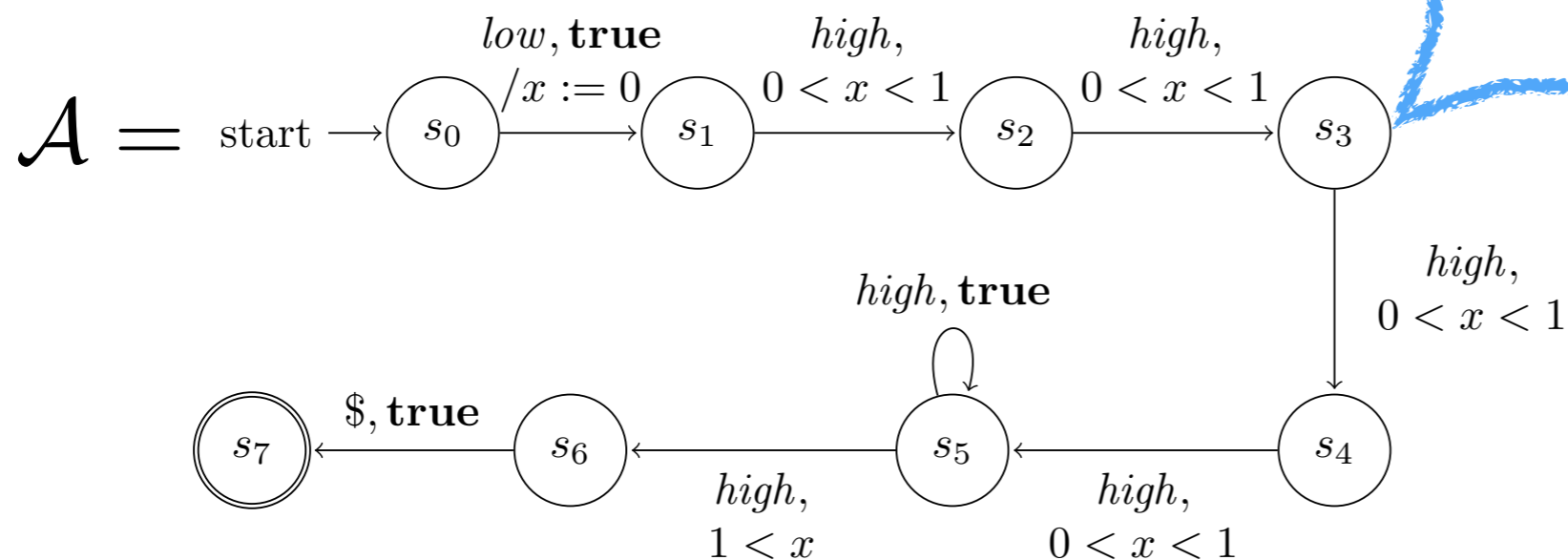
- Environment: MacBook Air Core i7-2677M
RAM 3.7GB
- Implemented in C++
- Show two cases out of five cases in the paper
- Used engine model of Simulink in Case1
- Case2 is taken from [Ulus et al., FORMATS '14]

Case 1

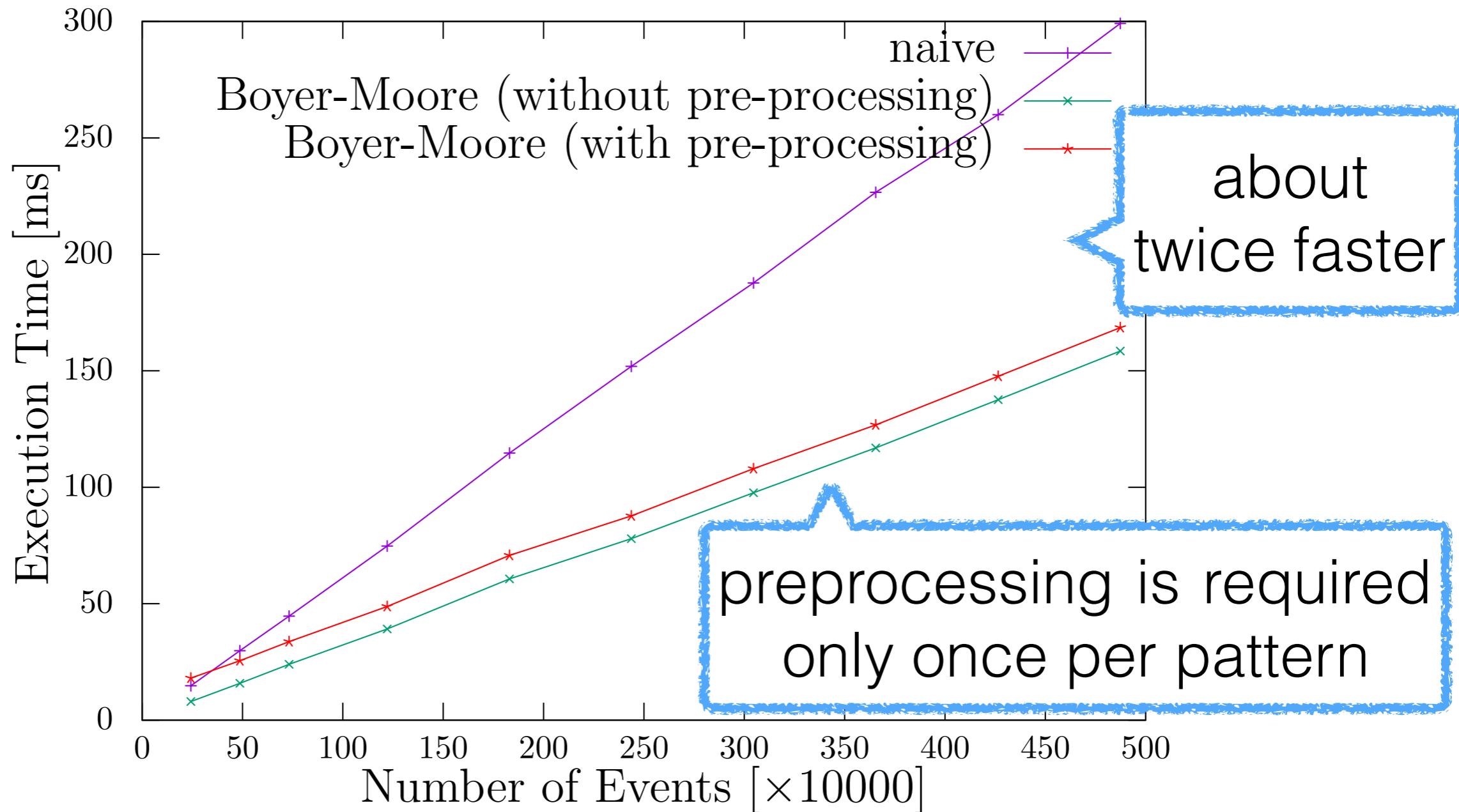
- Close to our target application

w : sequential binarized (high/low) logs of the torque of an engine simulation in Simulink

Match when the torque is too high for a long time.



Case 1



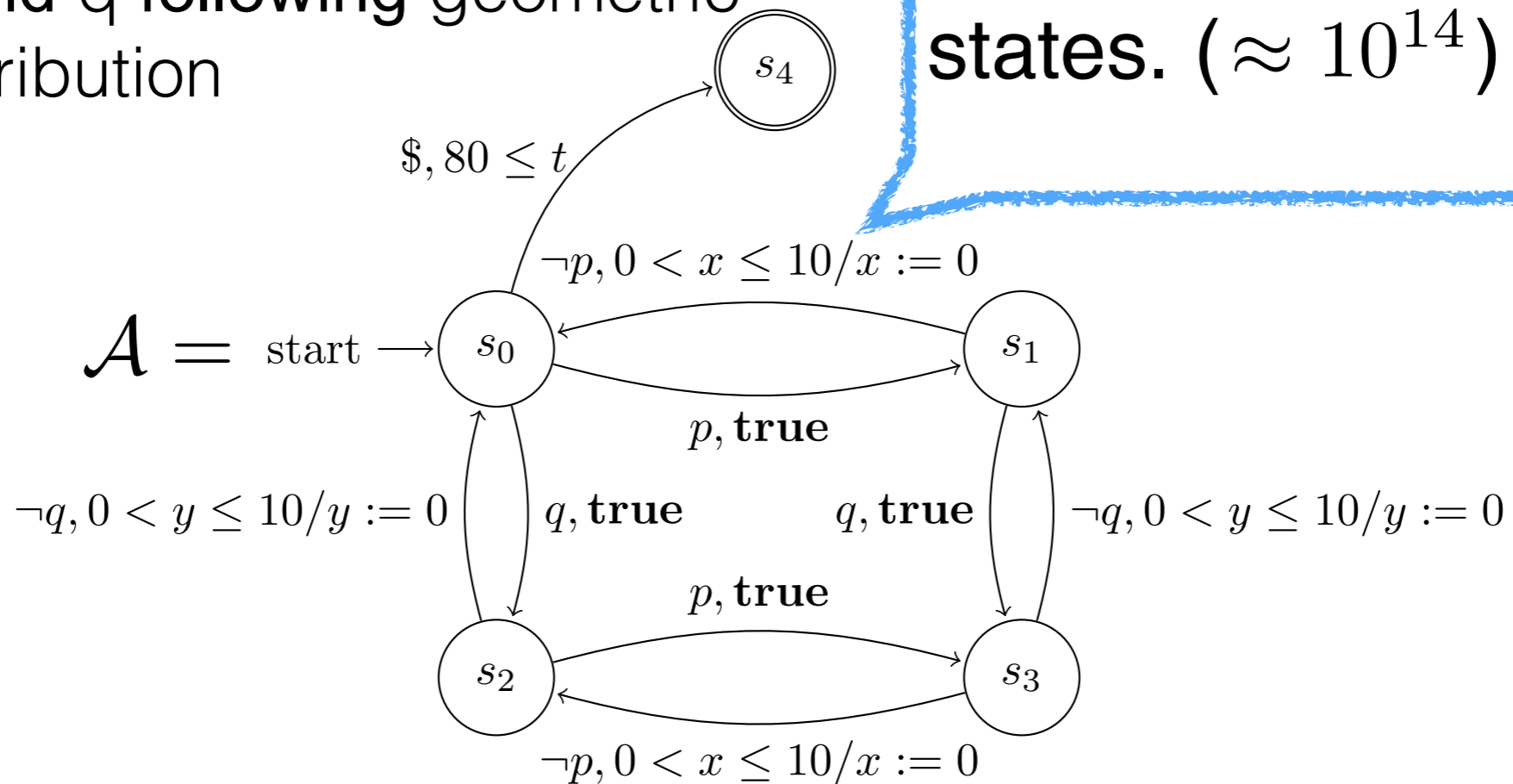
- Used the average of the 10 times run.

Case 2

- The cost of the precomputation is high.
- Taken from [Ulus et al., FORMATS '14]

w : on/off switching of two states
 p and q following geometric
distribution

$R(\mathcal{A} \times \mathcal{A})$ has many
states. ($\approx 10^{14}$)



Case 2

- The precomputation in our Boyer-Moore type algorithm **did not finish** due to the lack of RAM.
- The number of regions increases **exponentially**.
(bounded by $|C|! \cdot 2^{|C|} \cdot \prod_{x \in C} (2c_x + 2)$)
 - Using a **zone automaton** instead of a region automaton may be a solution.
- The naive algorithm worked well because it does not use region automata.

Conclusions and Future Work

- Solved the timed pattern matching problem with our Boyer-Moore type algorithm.
 - About **twice faster** than the naive one at most
 - Its space complexity is **too large** for some instances
- Reduce the space complexity of its precomputation
 - Using a **zone automaton** in place of a region automaton?