

Automata based verification over linearly ordered data domains

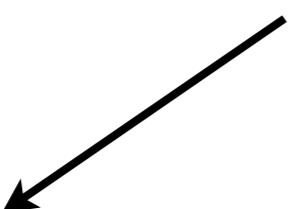
Luc Segoufin
INRIA and ENS Cachan

Szymon Toruńczyk
University of Warsaw

Automata based verification over linearly ordered data domains

Luc Segoufin
INRIA and ENS Cachan

Szymon Toruńczyk
University of Warsaw



Motivation

Motivation



Motivation

Given a system which refers to a database and data values, verify properties of the system

AVIS

Car Hire Home Page Return to Booking Help

Car Hire Offers Avis Locations Products & Services Join Avis Preferred Business Rentals Help and Contacts UK Fleet Home Delivery Newsletter

1 When and Where 2 Your Car Choice 3 Price and Extras 4 Checkout

Avis car hire quote and booking

Find pickup location by
Search for

Return location
Same as pickup location [Change](#)

Rental Start Date/Time 09 00

Rental End Date/Time 09 00

Number of days

Avis Worldwide Discount (AWD) No.

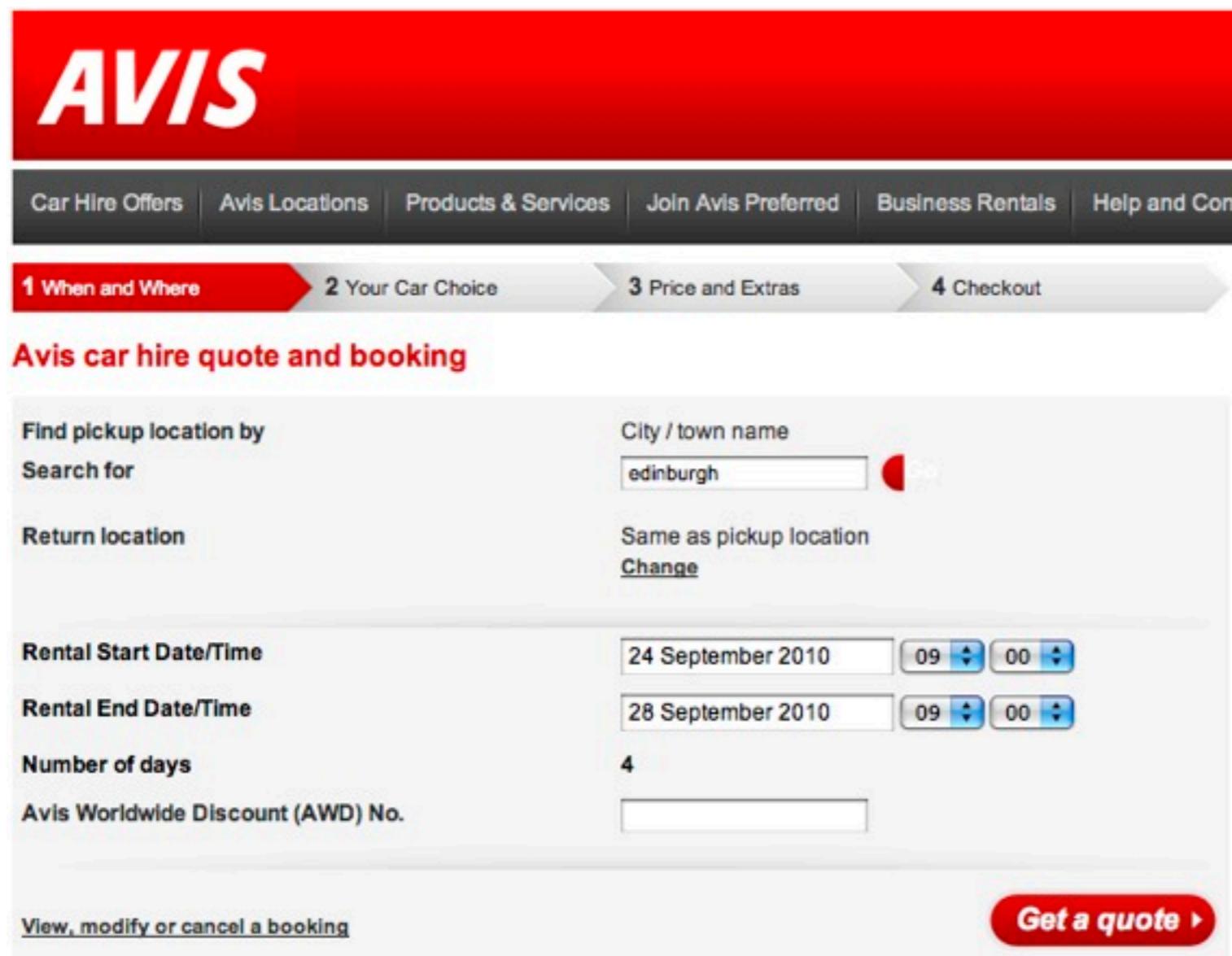
[View, modify or cancel a booking](#) [Get a quote ▶](#)

Sign in [Create an account](#)

[Sign In](#) to your Online account/Avis Preferred

**Car hire delivered
to your front door**

**Click here to book
Home Delivery! ▶**



Car hire from £17 per day
Enjoy great discounts this Autumn. Get on the road and explore the wonderful places the UK has to offer.
Book Now!



Car hire from £15 per day
Explore Europe this Autumn with great discounts which will take you further so you can see more.
Book Now!



Car hire from £17 per day
Explore the world by taking advantage of our amazing Autumn Sale prices.
Book Now!

Motivation

Given a system which refers to a database and data values, verify properties of the system

AVIS

Car Hire Home Page Return to Booking Help

Car Hire Offers Avis Locations Products & Services Join Avis Preferred Business Rentals Help and Contacts UK Fleet Home Delivery Newsletter

1 When and Where 2 Your Car Choice 3 Price and Extras 4 Checkout Sign in Create an account

Avis car hire quote and booking

Find pickup location by
Search for

Return location
Same as pickup location [Change](#)

Rental Start Date/Time 09 00

Rental End Date/Time 09 00

Number of days

Avis Worldwide Discount (AWD) No.

[View, modify or cancel a booking](#)

Get a quote ▶

input values

change state

Car hire delivered to your front door

UK from £17 per day

Car hire from £17 per day

Enjoy great discounts this Autumn. Get on the road and explore the wonderful places the UK has to offer.

Book Now!

Europe £15 per day

Car hire from £15 per day

Explore Europe this Autumn with great discounts which will take you further so you can see more.

Book Now!

Car hire from £17 per day

Explore the world by taking advantage of our amazing Autumn Sale prices.

Book Now!

Motivation

Given a system which refers to a database and data values, verify properties of the system

AVIS

Car Hire Home Page Return to Booking Help

Car Hire Offers Avis Locations Products & Services Join Avis Preferred Business Rentals Help and Contacts UK Fleet Home Delivery Newsletter

1 When and Where 2 Your Car Choice 3 Price and Extras 4 Checkout

You are viewing vehicles available at Edinburgh Airport

Sign in Create an account

Sign in to your Online account/Avis Preferred

Your Booking

1 WHEN AND WHERE Change

Pickup : Edinburgh Airport 23/09/2010 19:00
Return : Edinburgh Airport 28/09/2010 19:00
Rental Days : 5

YOUR CAR CHOICE

3 PRICE AND EXTRAS

4 CHECKOUT

Empty Basket

Small Medium Large Select Series

Price Size Number of Luggage

 Small : Economy (Example of this range : Peugeot 207) Hide Info

Best price £155.65 per rental
Available to book now

Vehicle Features Air Bag - Driver Short Wheel Base Radio/Cassette

Driver age requirements
You must be at least 23 years old to hire this vehicle If you are under 25 years old a Young Driver Surcharge will apply Young Driver surcharge is £11/day + VAT; max £110

Credit card requirements
The number of Credit Cards required when you pick up this vehicle is: 1.

 Medium : Economy (Example of this range : Nissan Note 1.4) More Info

Best price £181.37 per rental
Available to book now

Motivation

Given a system which refers to a database and data values, verify properties of the system

AVIS

Car Hire Home Page Return to Booking Help

Car Hire Offers Avis Locations Products & Services Join Avis Preferred Business Rentals Help and Contacts UK Fleet Home Delivery Newsletter

1 When and Where 2 Your Car Choice 3 Price and Extras 4 Checkout

You are viewing vehicles available at Edinburgh Airport

Small Medium Large Select Series

Price Size Number of Luggage

 Small : Economy (Example of this range : Peugeot 207)
Best price £155.65 per rental
Available to book now

[Hide Info](#)

Vehicle Features
Air Bag - Driver
Short Wheel Base

Radio/Cassette

Driver age requirements
You must be at least 23 years old to hire this vehicle If you are under 25 years old a Young Driver Surcharge will apply Young Driver surcharge is £11/day + VAT; max £110

Credit card requirements
The number of Credit Cards required when you pick up this vehicle is: 1.

Medium : Economy (Example of this range : Nissan Note 1.4)

 Best price £181.37 per rental
Available to book now

[More Info](#)

Sign in [Create an account](#)

[Sign in](#) to your Online account/Avis Preferred

Your Booking

1 WHEN AND WHERE [Change](#)

Pickup : Edinburgh Airport 23/09/2010 19:00

Return : Edinburgh Airport 28/09/2010 19:00

Rental Days : 5

YOUR CAR CHOICE

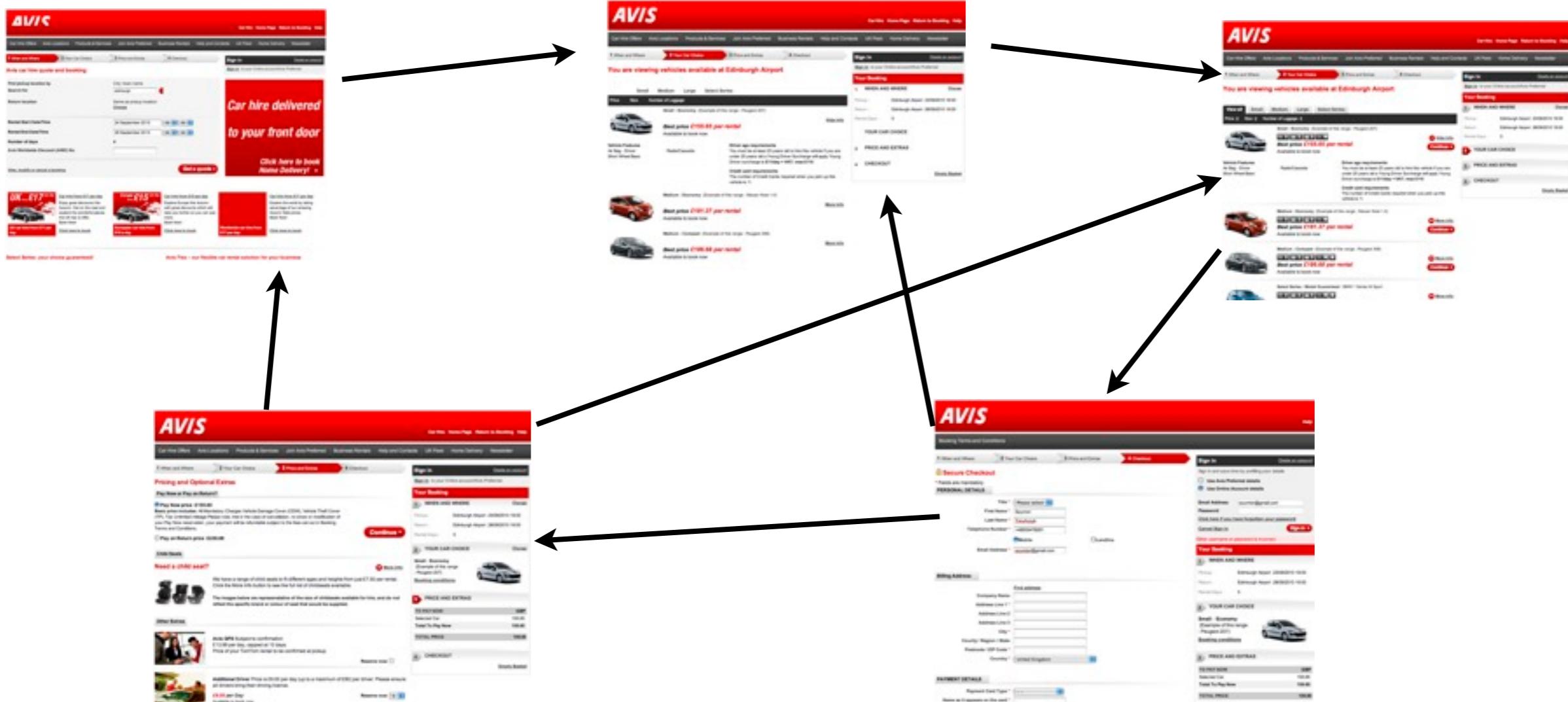
3 PRICE AND EXTRAS

4 CHECKOUT

[Empty Basket](#)

Motivation

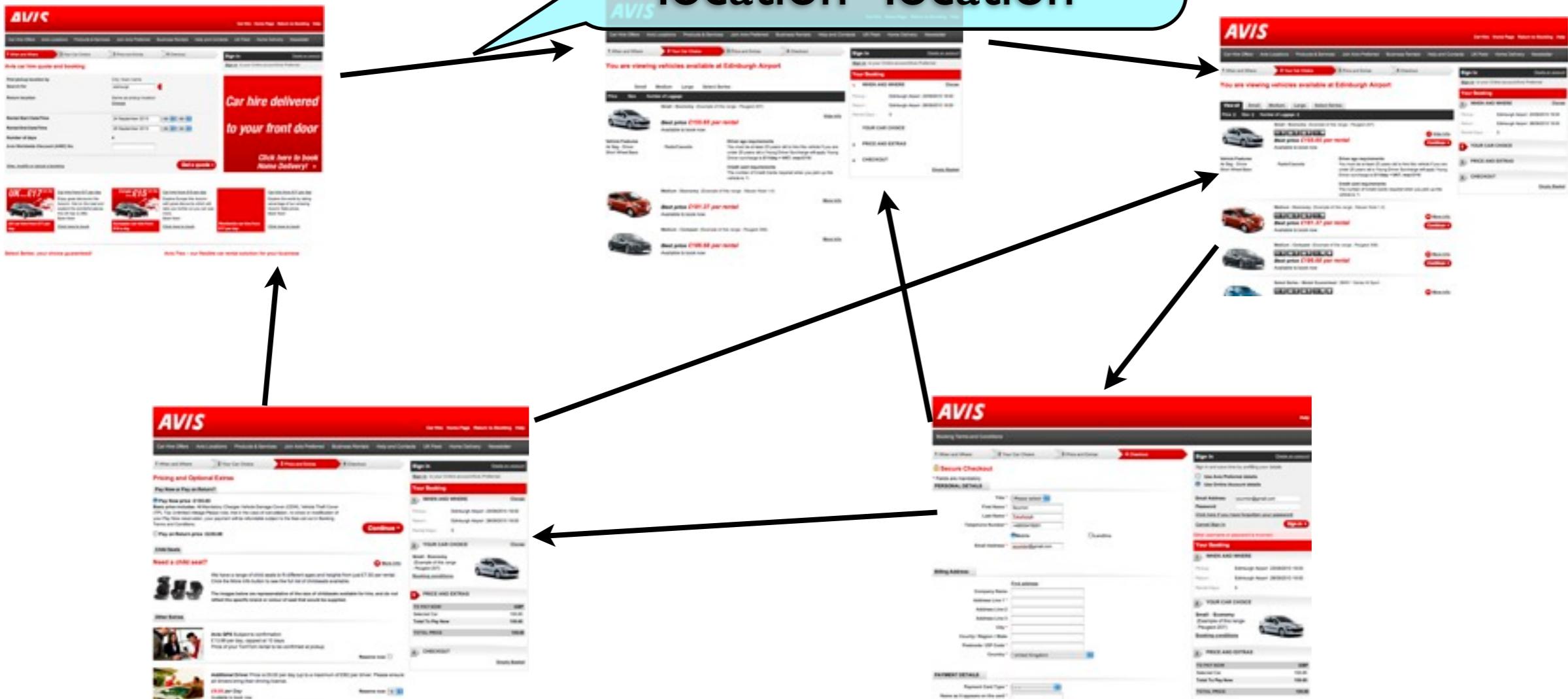
Given a system which refers to a database and data values, verify properties of the system



Motivation

Given a system which refers to a database and data values, verify properties of the system

start_date=start_date'
end_date=end_date'
location=location'



Motivation

Given a system which refers to a database and data values, verify properties of the system

$\text{start_date} = \text{start_date}'$
 $\text{end_date} = \text{end_date}'$
 $\text{location} = \text{location}'$

This screenshot shows the Avis website interface. It displays a search form for car hire, including fields for pickup location (Edinburgh), drop-off location (Edinburgh), and dates (24 September 2014 - 25 September 2014). A prominent red banner states "Car hire delivered to your front door". Below the form, there are three car models listed with their prices: Small (£199.85 per day), Medium (£209.37 per day), and Large (£209.89 per day). At the bottom, there are links for "Secure Checkout" and "Check Out".

This screenshot shows the Avis website displaying vehicles available at Edinburgh Airport. It lists three car models: Small (Small Hatchback, £199.85 per day), Medium (Medium Hatchback, £209.37 per day), and Large (Large Hatchback, £209.89 per day). Each listing includes a "View Details" button. The page also features sections for "Vehicles & Services" and "Business Rates".

This screenshot shows the Avis website with a large blue callout box containing the inequality $\text{pricemin} \leq \text{price} \leq \text{pricemax}$. The page displays a list of vehicles with their respective daily rates: Small (£199.85), Medium (£209.37), and Large (£209.89). The background shows the Avis logo and some vehicle images.

This screenshot shows the Avis website's "Pricing and Optional Extras" section. It includes a "Pay Now or Pay on Return?" checkbox, a "Need a CDW?" section with a "Yes" checkbox, and a "PRICE AND EXTRAS" table. The table shows rates for Small Car (£199.85), Medium Car (£209.37), and Large Car (£209.89). Below the table is a "CheckOut" button. To the right, there is a "Billing Address" form and a "PAYMENT DETAILS" section.

This screenshot shows the Avis website's "Secure Checkout" page. It includes a "PERSONAL DETAILS" section with fields for First Name, Last Name, Middle Name, Home Phone Number, and Email Address. To the right, there is a "Billing Address" form and a "PAYMENT DETAILS" section. A large red "Secure Checkout" button is prominently displayed.

Motivation

Given a system which refers to a database and data values, verify properties of the system

$\text{start_date} = \text{start_date}'$
 $\text{end_date} = \text{end_date}'$
 $\text{location} = \text{location}'$

$\text{pricemin} \leq \text{price}$
 $\leq \text{pricemax}$

CAR(car_id, price)

A screenshot of the Avis website showing a search form for car hire. The form includes fields for pickup location (Edinburgh), drop-off location (Edinburgh), date range (24 September 2012 - 25 September 2012), and vehicle type (Small). A red banner at the bottom right says "Car hire delivered to your front door".

A screenshot of the Avis website showing a list of available vehicles at Edinburgh Airport. It includes small images of the cars, their names, and their prices per day: Small (Basic) £199.85, Medium (Economy) £219.27, and Large (Executive) £239.69.

A screenshot of the Avis website showing a search result page for Edinburgh. It displays a list of vehicles available at Edinburgh Airport, with the same three options as the previous screenshot.

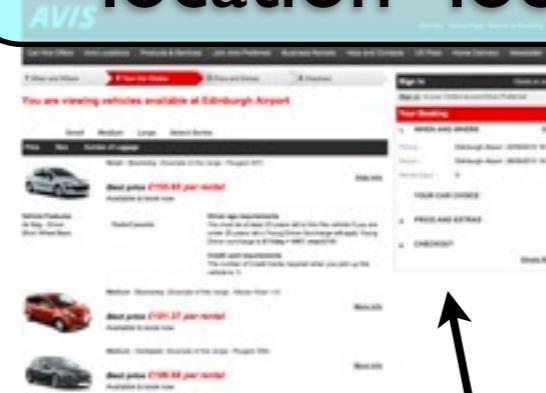
A screenshot of the Avis website showing a booking confirmation page. It shows a summary of the booking details, including the car choice (Small Economy), pickup location (Edinburgh Airport), and price (£199.85). It also includes sections for optional extras like GPS and damage cover.

A screenshot of the Avis website showing a secure checkout page. It includes fields for personal details (First Name, Last Name, Middle Name, Home Number, Email Address), billing address (Company Name, Address Line 1, Address Line 2, Address Line 3, City, Postcode, Country), and payment details (Payment Card Type, Card Number, Expiry Date, CVV2/CVC2). A red banner at the top right says "Secure Checkout".

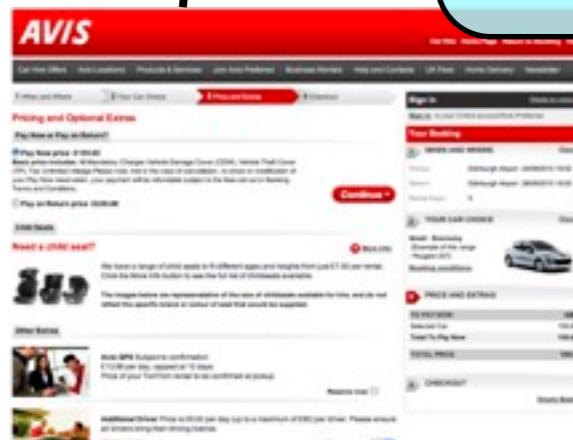
Motivation

Given a system which refers to a database and data values, verify properties of the system

$\text{start_date} = \text{start_date}'$
 $\text{end_date} = \text{end_date}'$
 $\text{location} = \text{location}'$



$\text{CAR}(\text{car_id}, \text{price})$

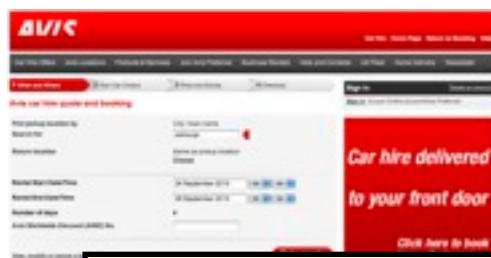


$\text{PAYMENT}(\text{user_id}, \text{car_id},$
 $\text{price}, \text{order_id})$

Motivation

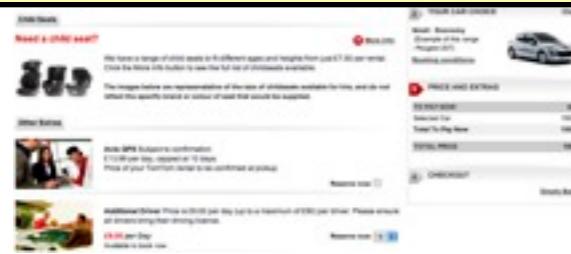
Given a system which refers to a database and data values, verify properties of the system

$\text{start_date} = \text{start_date}'$
 $\text{end_date} = \text{end_date}'$
 $\text{location} = \text{location}'$



$\text{pricemin} < \text{price}$

finally, a car is rented and

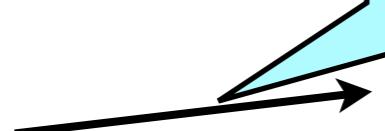


PAYMENT(user_id,car_id,
price,order_id)

Motivation

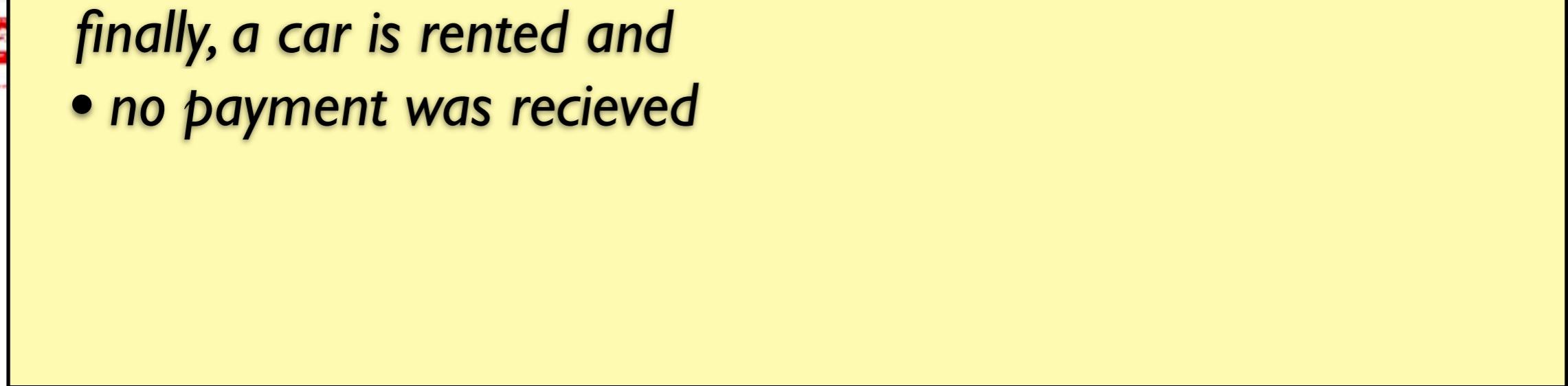
Given a system which refers to a database and data values, verify properties of the system

$\text{start_date} = \text{start_date}'$
 $\text{end_date} = \text{end_date}'$
 $\text{location} = \text{location}'$



$\text{pricemin} < \text{price}$

finally, a car is rented and
• no payment was received

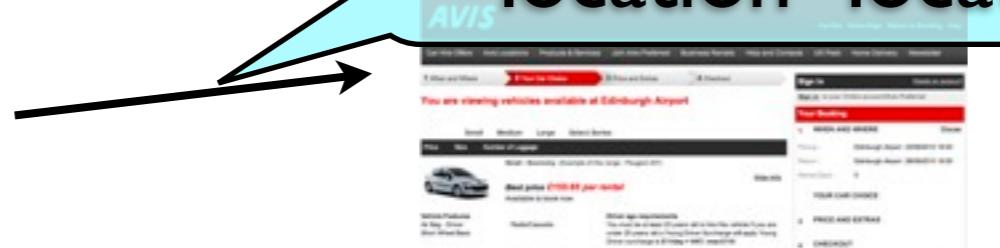


$\text{PAYMENT}(\text{user_id}, \text{car_id}, \text{price}, \text{order_id})$

Motivation

Given a system which refers to a database and data values, verify properties of the system

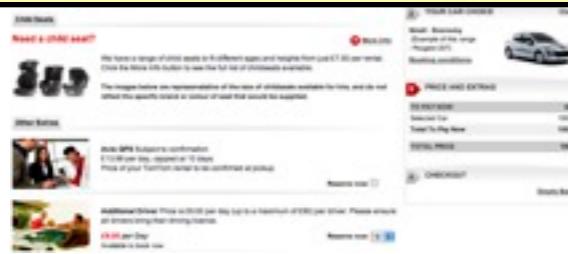
$\text{start_date} = \text{start_date}'$
 $\text{end_date} = \text{end_date}'$
 $\text{location} = \text{location}'$



$\text{pricemin} < \text{price}$

finally, a car is rented and

- no payment was received
- a payment was received but $\text{payed} = 0$



$\text{PAYMENT}(\text{user_id}, \text{car_id}, \text{price}, \text{order_id})$

Motivation

Given a system which refers to a database and data values, verify properties of the system

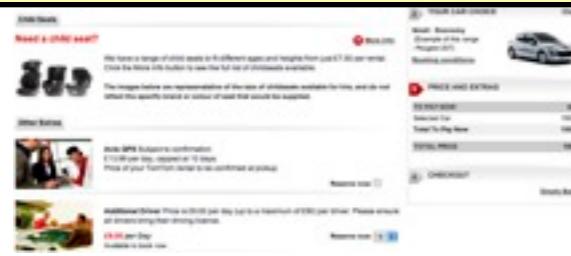
$\text{start_date} = \text{start_date}'$
 $\text{end_date} = \text{end_date}'$
 $\text{location} = \text{location}'$



$\text{pricemin} < \text{price}$

finally, a car is rented and

- no payment was received
- a payment was received but $\text{payed} = 0$
- there was a payment, but $\text{payed} < 100 \text{ & } \text{end_date} > 2050$



$\text{PAYMENT}(\text{user_id}, \text{car_id}, \text{price}, \text{order_id})$

Motivation

Given a system which refers to a database and data values, verify properties of the system

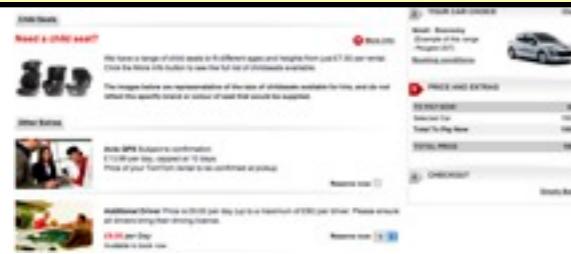
$\text{start_date} = \text{start_date}'$
 $\text{end_date} = \text{end_date}'$
 $\text{location} = \text{location}'$



$\text{pricemin} < \text{price}$

finally, a car is rented and

- no payment was received
- a payment was received but $\text{payed} = 0$
- there was a payment, but $\text{payed} < 100$ & $\text{end_date} > 2050$
- there was a payment, but $\text{payed} < \text{car_price}$



$\text{PAYMENT}(\text{user_id}, \text{car_id}, \text{price}, \text{order_id})$

Extended automaton

a definition that captures timed automata, or vector addition systems

D – fixed domain

Q – finite set of states

X – finite set of variables

$D^X = \{ v: X \rightarrow D \}$ – space of variable valuations

$Q \times D^X$ – space of configurations

restricted
form

$\delta \subseteq (Q \times D^X) \times (Q \times D^X)$ – set of allowed transitions

$I \subseteq (Q \times D^X)$ – set of initial configurations

$F \subseteq (Q \times D^X)$ – set of final configurations

Examples

- *Vector Addition System:* $D = \mathbb{N}$

$$(q, v) \xrightarrow{v' = v + w} (q', v')$$

- *Timed Automata:* $D = \mathbb{R}$

$$(q, v) \xrightarrow{\begin{array}{l} c_1 < 2 \\ c_2 = 0 \end{array}} (q', v')$$

- *(Lossy) channel system:* $D = \{a+b\}^*$

$$(q, v) \xrightarrow{\begin{array}{l} first_a(c_1) \\ c_1 = c_1 \cdot b \end{array}} (q', v')$$

Our setting (without the database)

The domain:

$$\mathcal{D} = \langle D, <, P_1, P_2, P_3, \dots, P_l \rangle$$

\begin{array}{ll} \text{linearly ordered set} & \text{unary predicates} \\[-1ex] \swarrow & \searrow \\[-1ex] & \text{(subsets of } D) \end{array}

Examples

$$\langle \mathbb{N}, <, 0, 100, P_{even}, P_{prime} \rangle$$

$$\langle \mathbb{Q}, <, 0, 100, P_{integer}, P_{<\pi} \rangle$$

$$\langle \{a+b\}^*, <_{lex}, P_{(ab)^*} \rangle$$

Transitions:

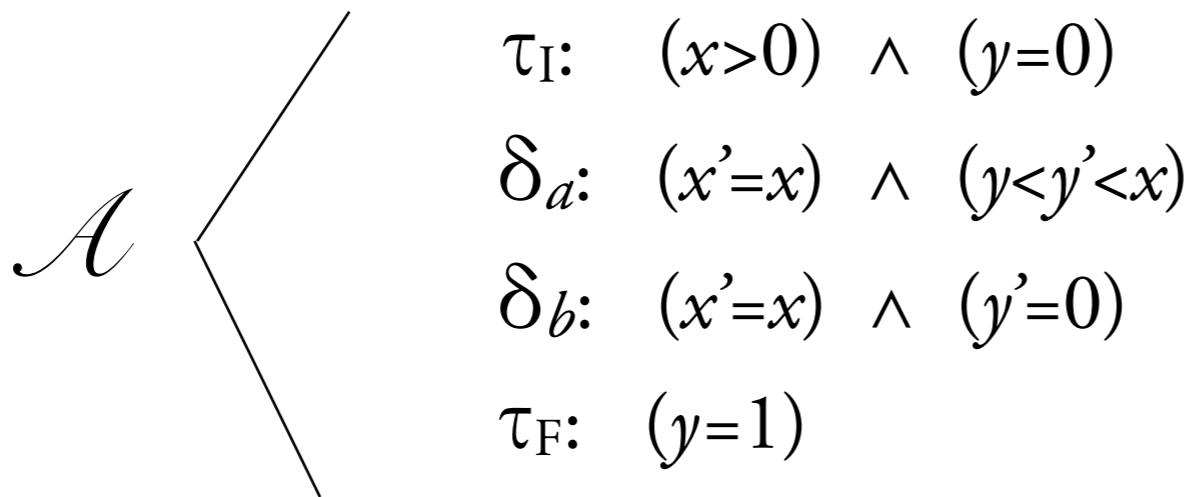
$$I, F \subseteq (Q \times D^X), \quad \delta \subseteq (Q \times D^X) \times (Q \times D^X)$$

are specified by quantifier free formulas over \mathcal{D}

Example D -automaton \mathcal{A}

x,y – variables of \mathcal{A}

no states



Example D -automaton \mathcal{A}

\mathcal{A}

x, y – variables of \mathcal{A}

no states



y

x

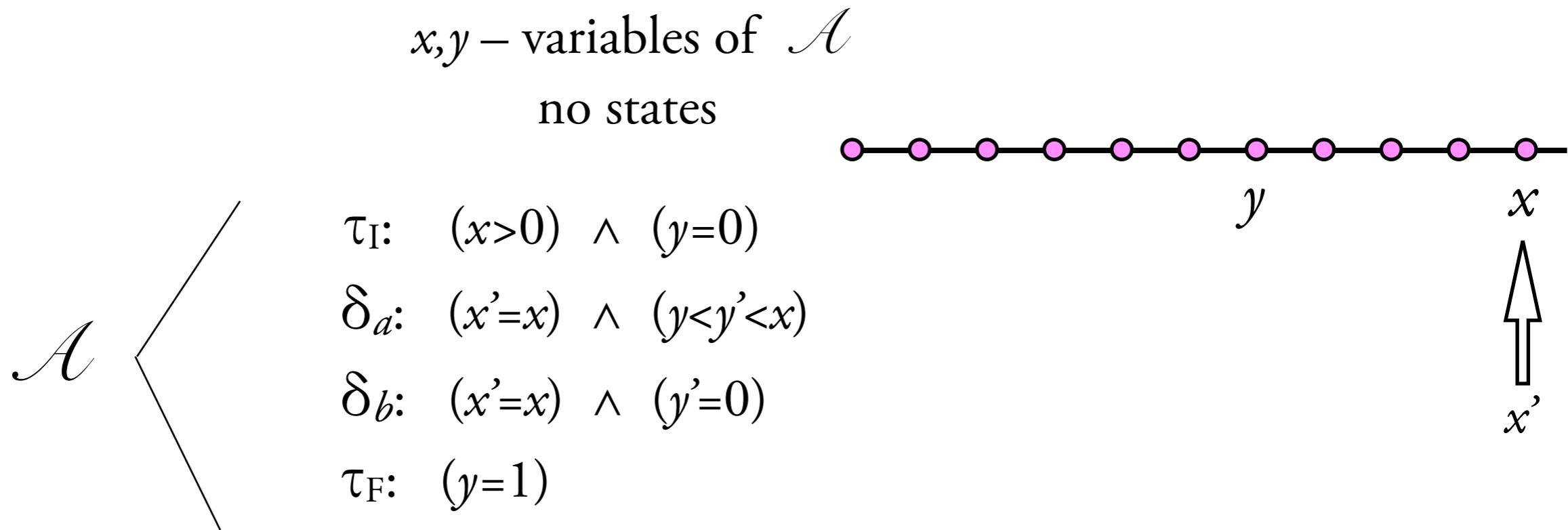
$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

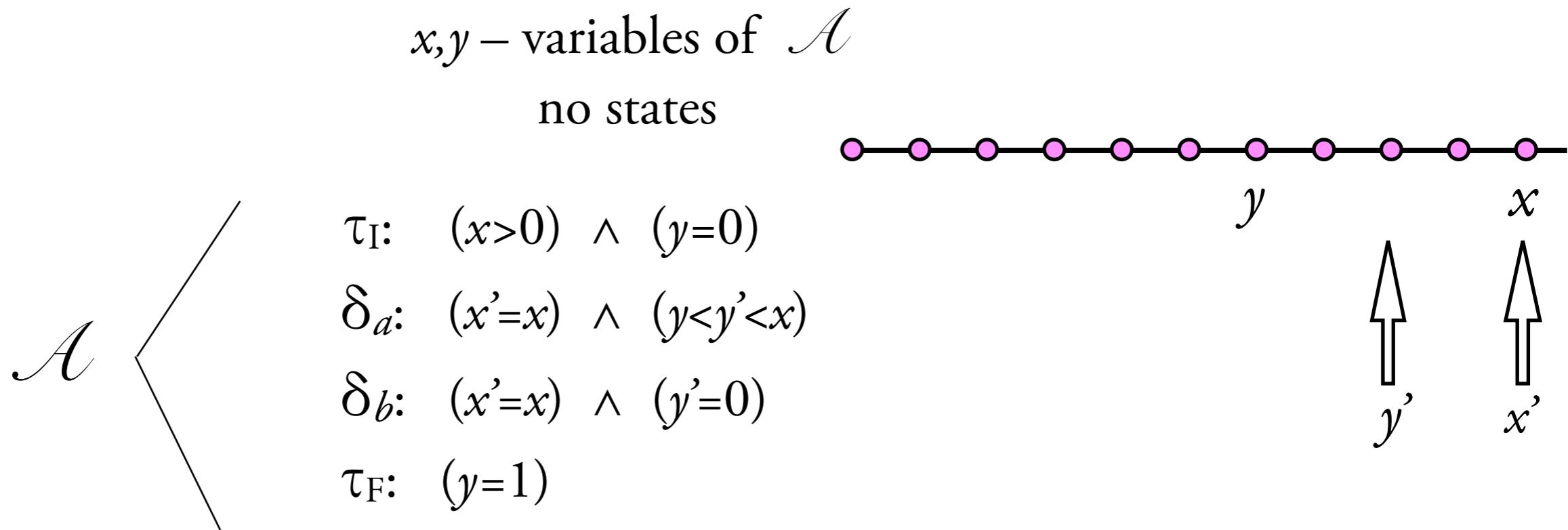
$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$

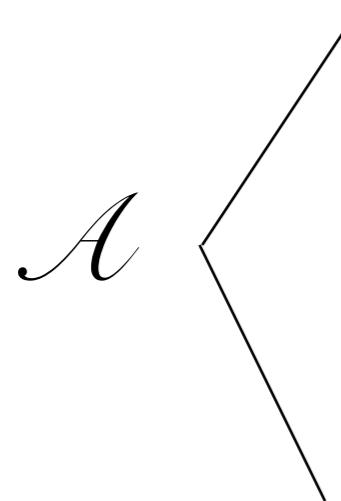
Example D -automaton \mathcal{A}



Example D -automaton \mathcal{A}



Example D -automaton \mathcal{A}



x, y – variables of \mathcal{A}

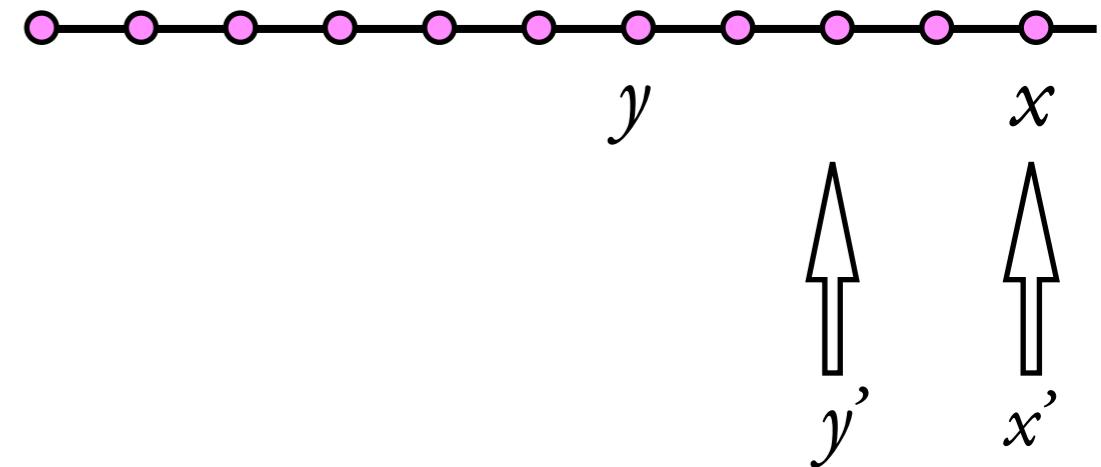
no states

$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

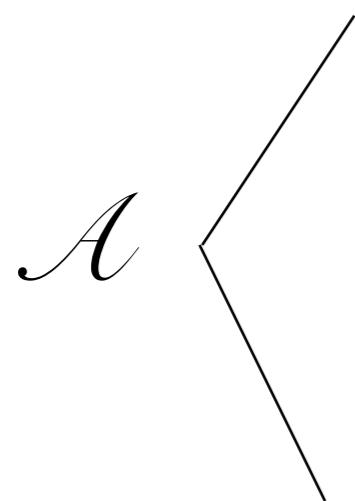
$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$



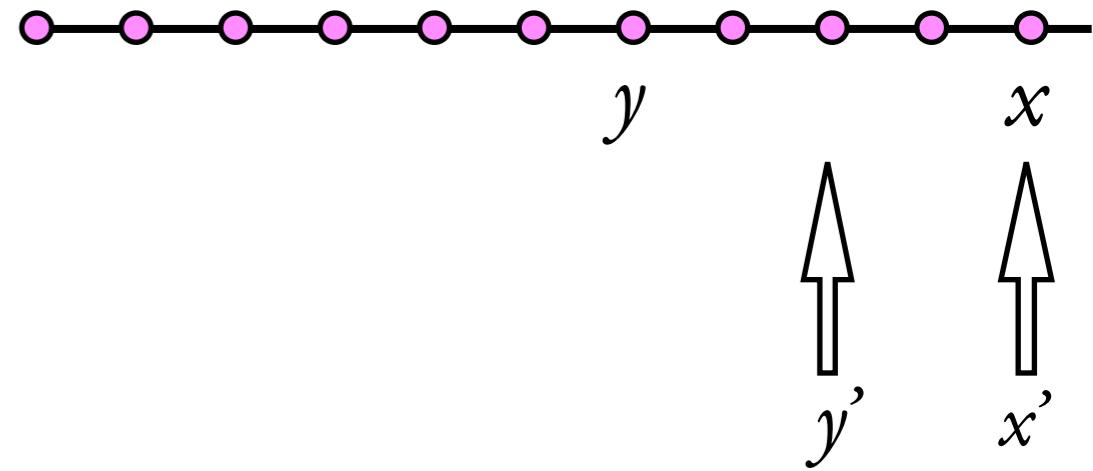
a finite run over \mathbb{Q} :

Example D -automaton \mathcal{A}



x, y – variables of \mathcal{A}

no states



$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

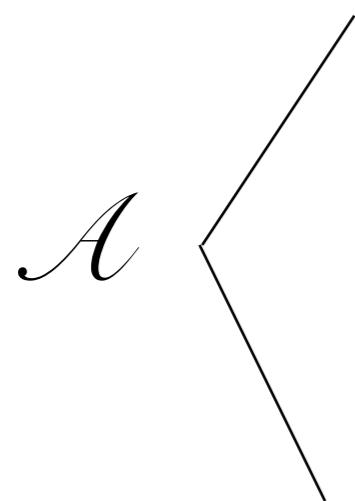
$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$

a finite run over \mathbb{Q} :

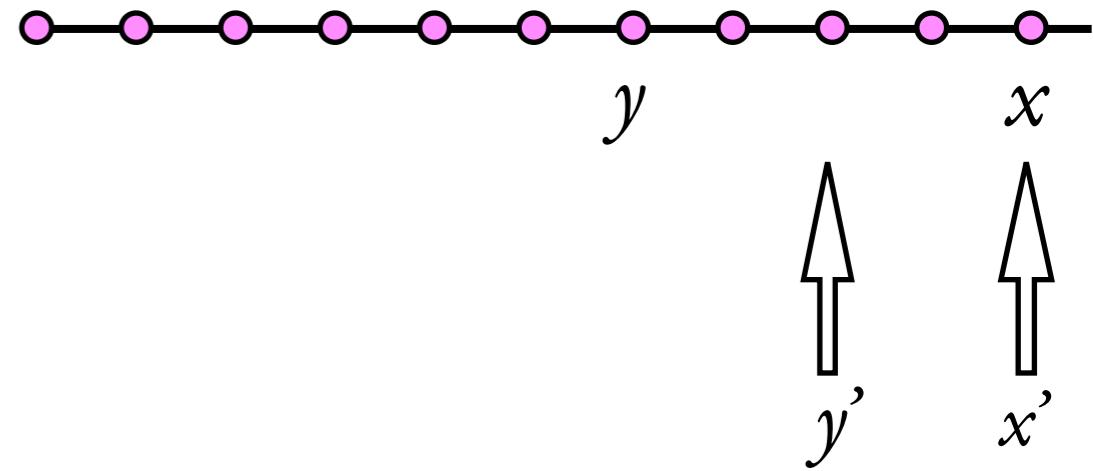
$a a a b a b a a a a a$

Example D -automaton \mathcal{A}



x, y – variables of \mathcal{A}

no states



$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$

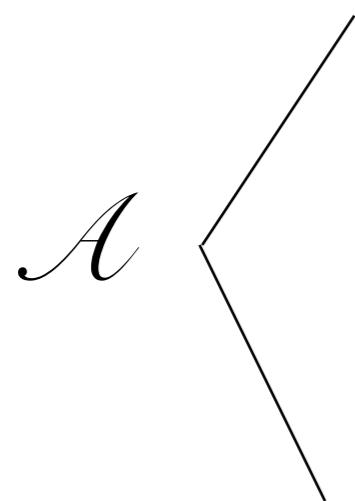
a finite run over \mathbb{Q} :

$a a a b a b a a a a a$

x

y

Example D -automaton \mathcal{A}



x, y – variables of \mathcal{A}

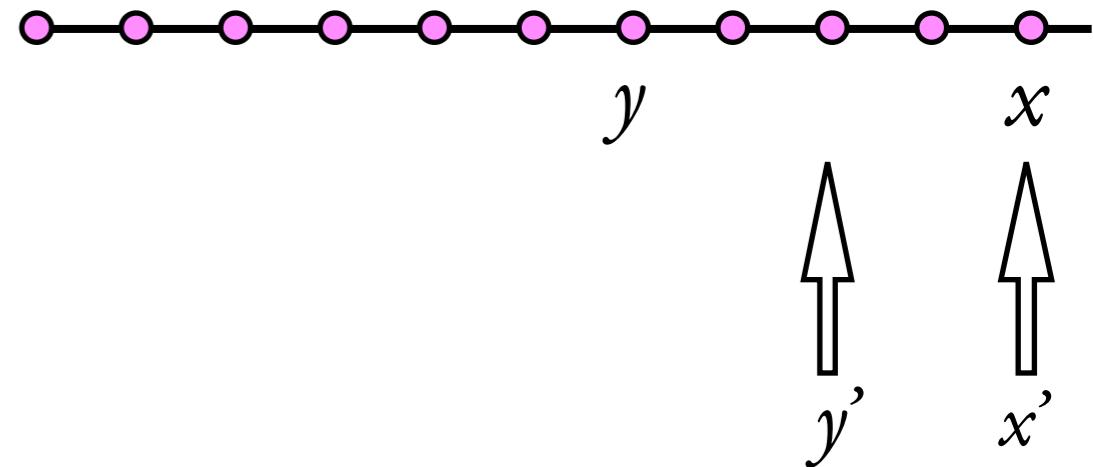
no states

$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$



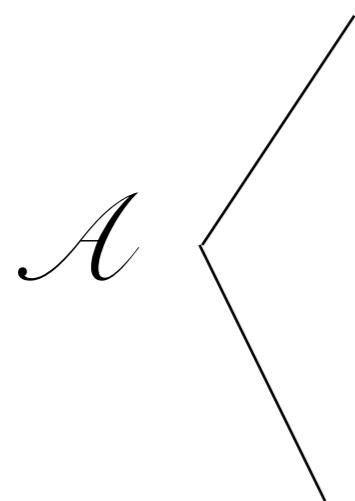
a finite run over \mathbb{Q} :

$a a a b a b a a a a a$

x 5

y 0

Example D -automaton \mathcal{A}



x, y – variables of \mathcal{A}

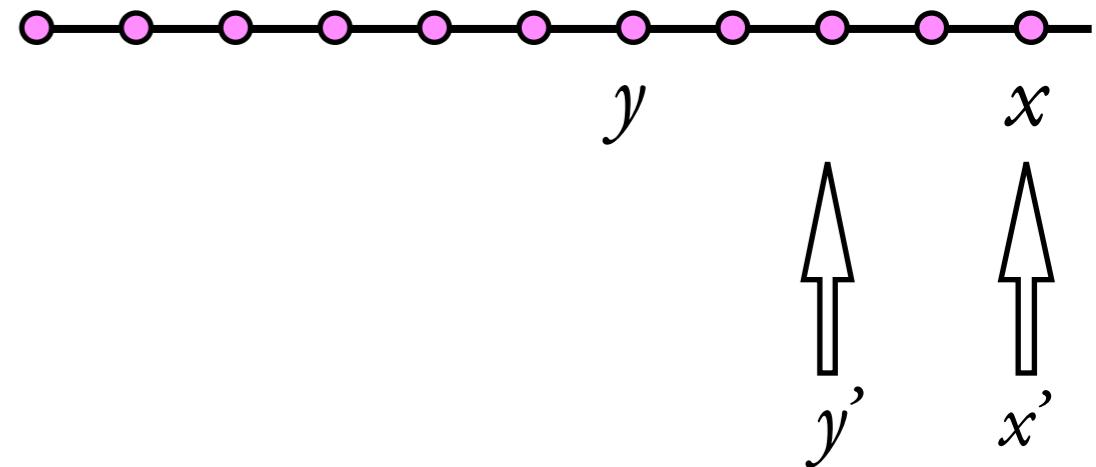
no states

$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$



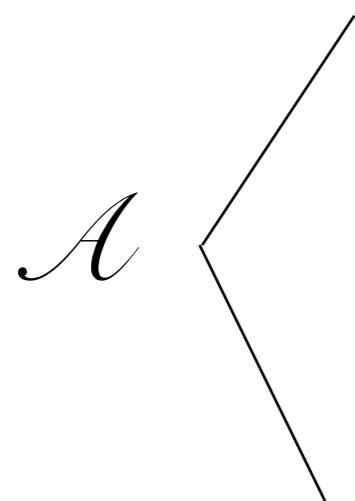
a finite run over \mathbb{Q} :

$a a a b a b a a a a a$

$x \quad 5 \quad 5$

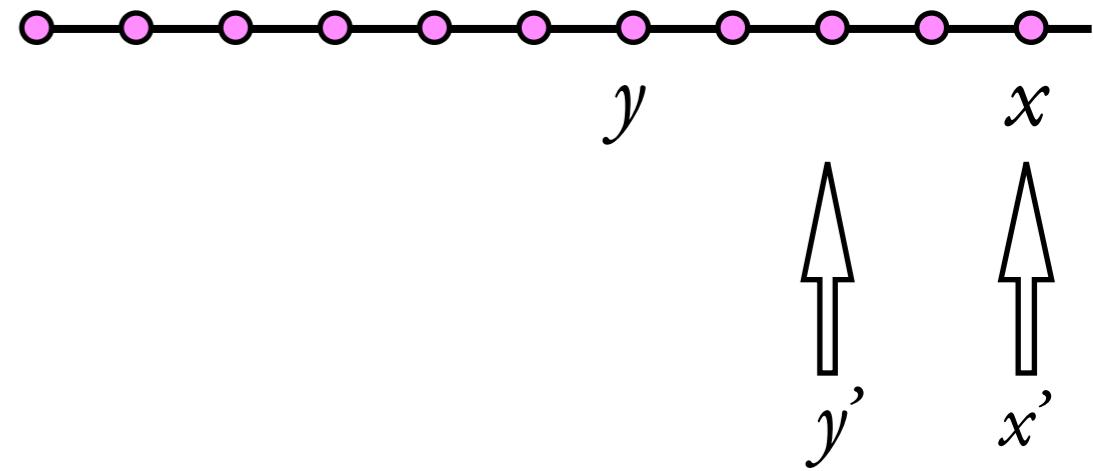
$y \quad 0 \quad 1$

Example D -automaton \mathcal{A}



x, y – variables of \mathcal{A}

no states



$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$

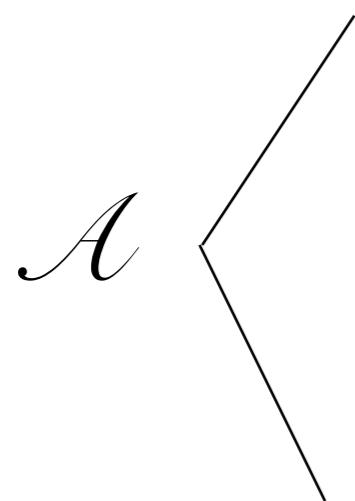
a finite run over \mathbb{Q} :

$a a a b a b a a a a a$

$x \quad 5 \ 5 \ 5$

$y \quad 0 \ 1 \ 2$

Example D -automaton \mathcal{A}



x, y – variables of \mathcal{A}

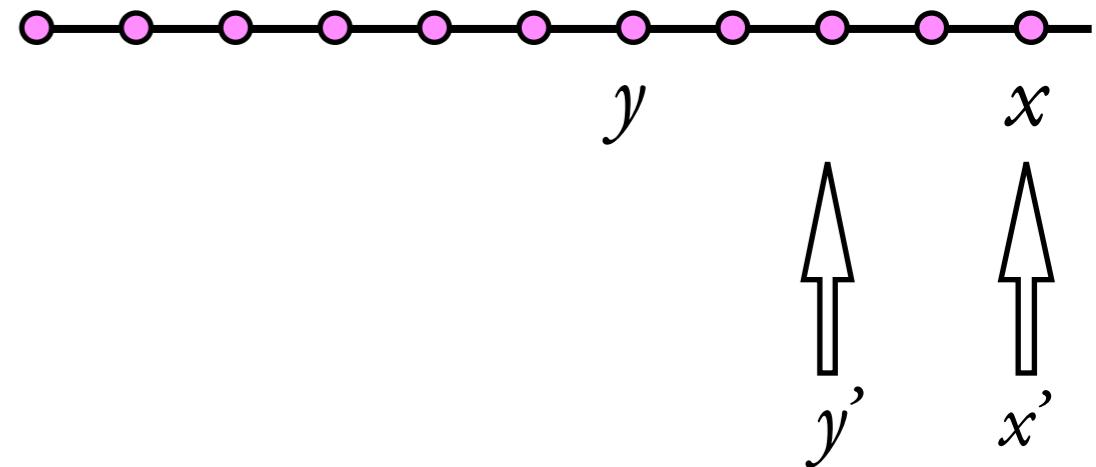
no states

$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$



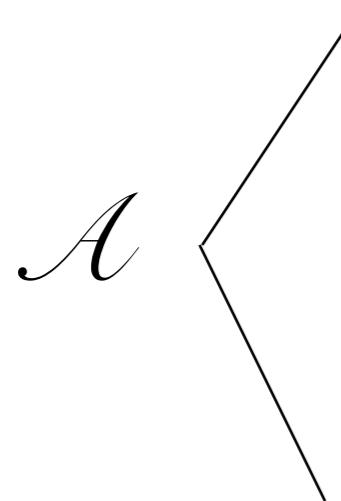
a finite run over \mathbb{Q} :

$a a a b a b a a a a a$

$x \quad 5 \ 5 \ 5 \ 5$

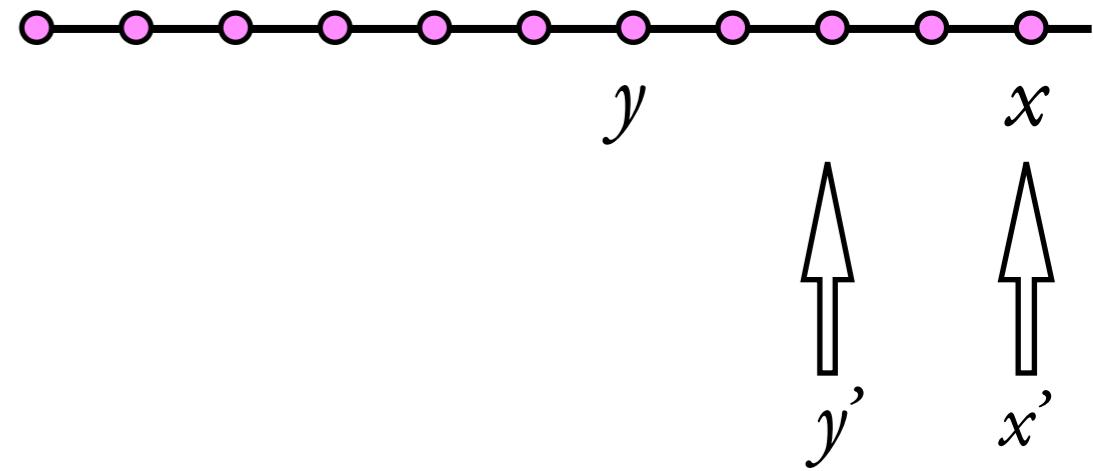
$y \quad 0 \ 1 \ 2 \ 4$

Example D -automaton \mathcal{A}



x, y – variables of \mathcal{A}

no states



$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$

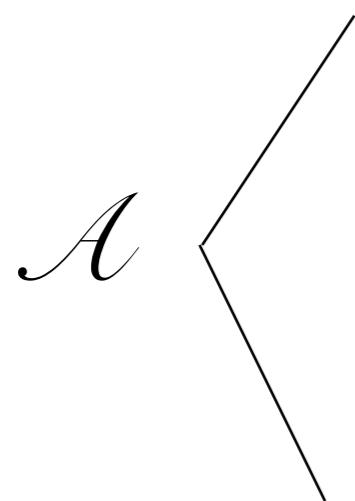
a finite run over \mathbb{Q} :

$a a a b a b a a a a a$

$x \quad 5 \ 5 \ 5 \ 5 \ 5$

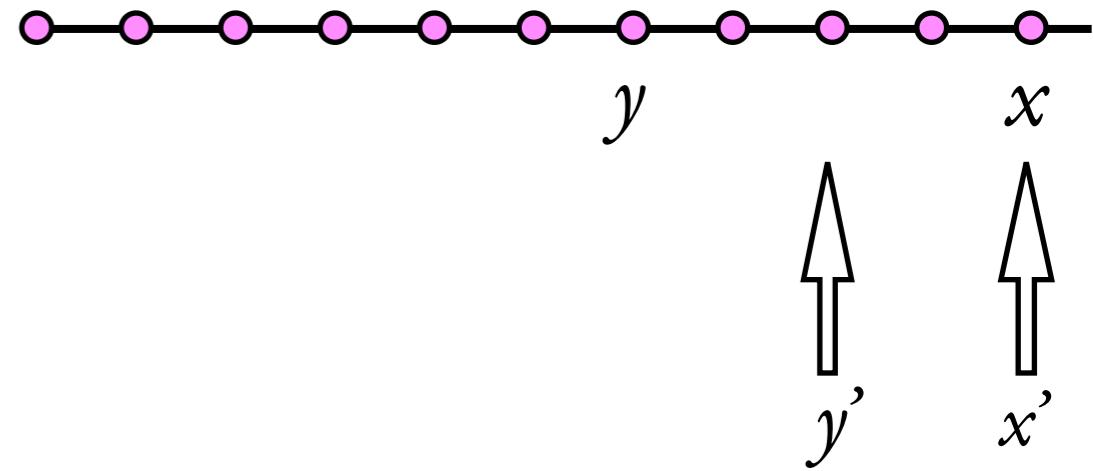
$y \quad 0 \ 1 \ 2 \ 4 \ 0$

Example D -automaton \mathcal{A}



x, y – variables of \mathcal{A}

no states



$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$

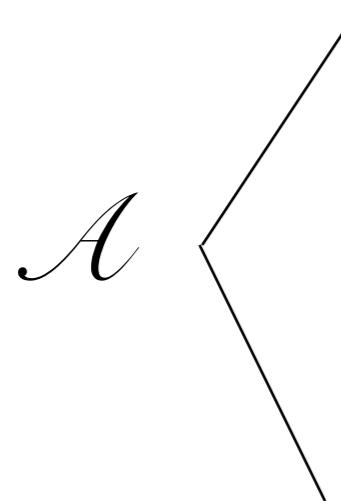
a finite run over \mathbb{Q} :

$a a a b a b a a a a a$

$x \quad 5 \ 5 \ 5 \ 5 \ 5 \ 5 \ 5$

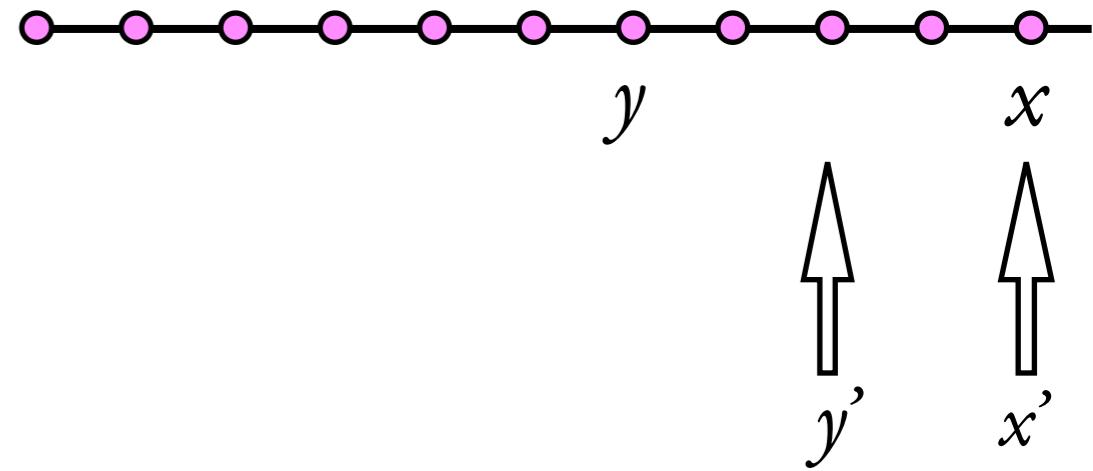
$y \quad 0 \ 1 \ 2 \ 4 \ 0 \ 3$

Example D -automaton \mathcal{A}



x, y – variables of \mathcal{A}

no states



$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$

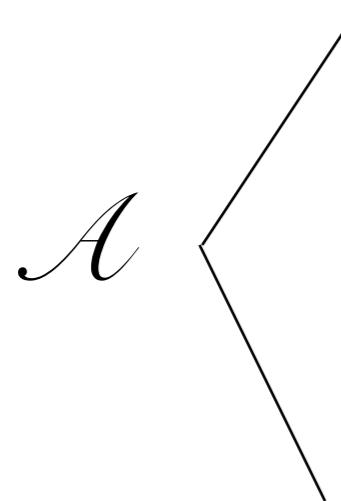
a finite run over \mathbb{Q} :

$a a a b a b a a a a a$

$x \quad 5 \ 5 \ 5 \ 5 \ 5 \ 5 \ 5 \ 5$

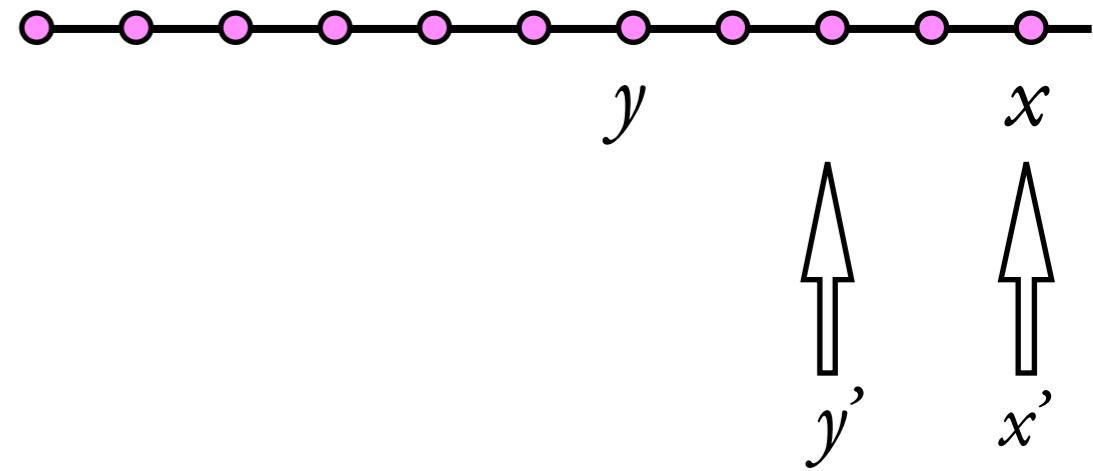
$y \quad 0 \ 1 \ 2 \ 4 \ 0 \ 3 \ 0$

Example D -automaton \mathcal{A}



x, y – variables of \mathcal{A}

no states



$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$

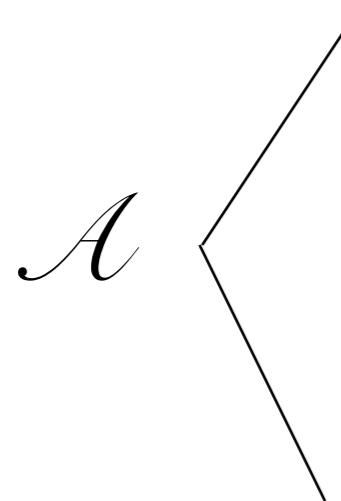
a finite run over \mathbb{Q} :

$a a a b a b a a a a a$

$x \quad 5 \ 5 \ 5 \ 5 \ 5 \ 5 \ 5 \ 5 \ 5$

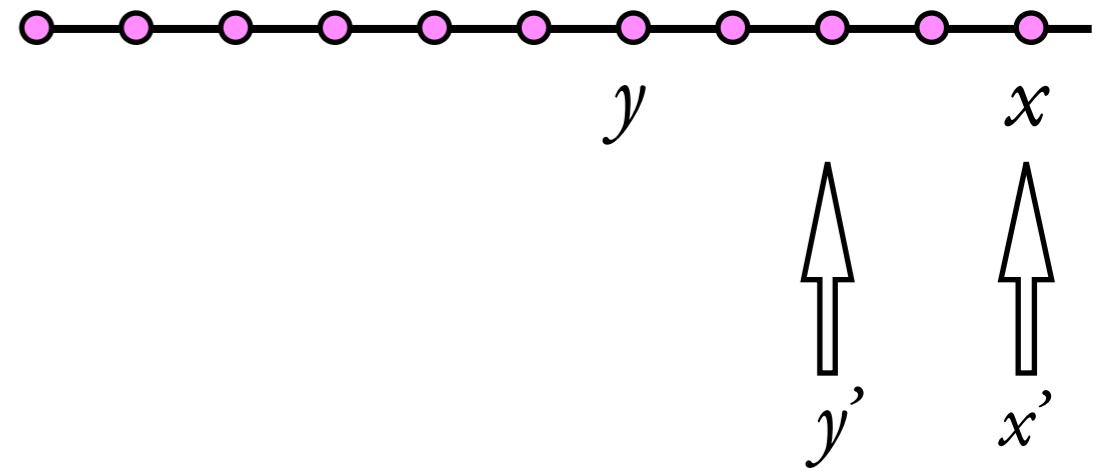
$y \quad 0 \ 1 \ 2 \ 4 \ 0 \ 3 \ 0 \ \frac{1}{4}$

Example D -automaton \mathcal{A}



x, y – variables of \mathcal{A}

no states



$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$

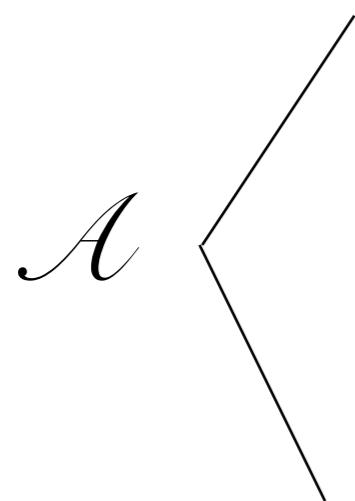
a finite run over \mathbb{Q} :

$a a a b a b a a a a a$

x	5	5	5	5	5	5	5	5
-----	---	---	---	---	---	---	---	---

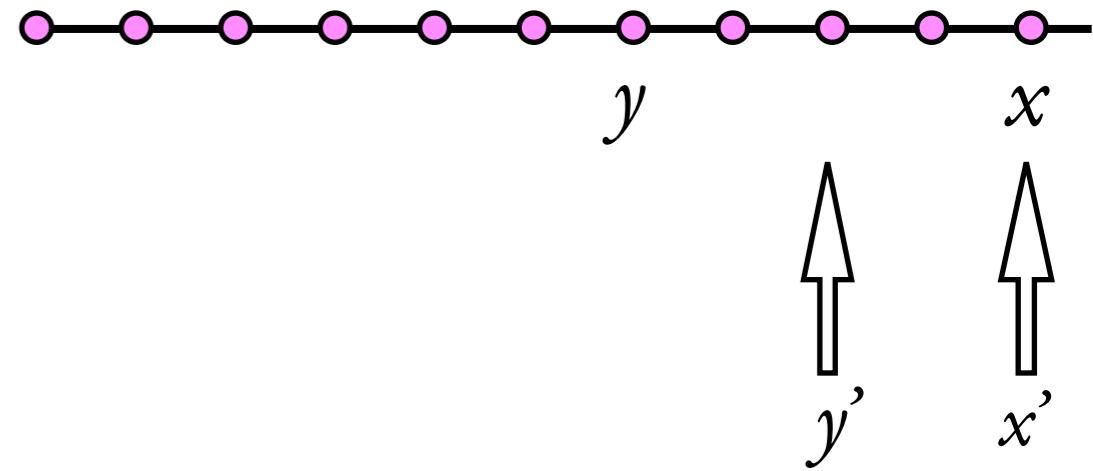
y	0	1	2	4	0	3	0	$\frac{1}{4}$	$\frac{1}{2}$
-----	---	---	---	---	---	---	---	---------------	---------------

Example D -automaton \mathcal{A}



x, y – variables of \mathcal{A}

no states



$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$

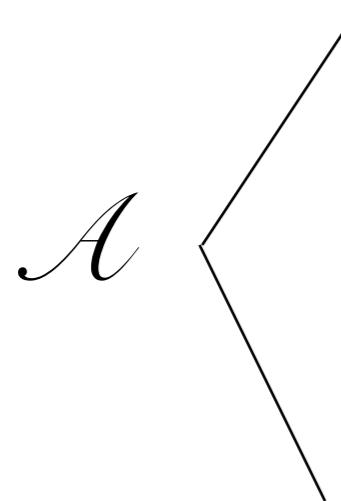
a finite run over \mathbb{Q} :

$a a a b a b a a a a a$

x	5	5	5	5	5	5	5	5	5
-----	---	---	---	---	---	---	---	---	---

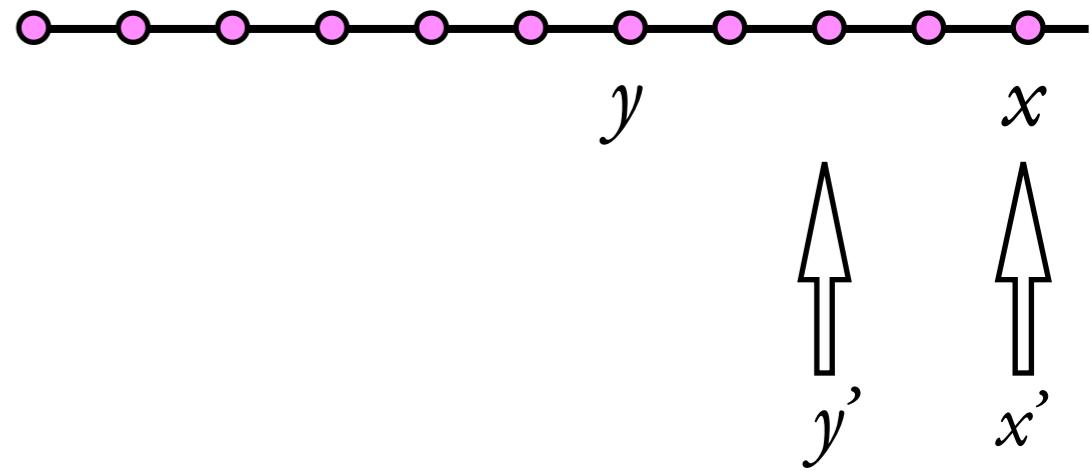
y	0	1	2	4	0	3	0	$\frac{1}{4}$	$\frac{1}{2}$	$\frac{2}{3}$
-----	---	---	---	---	---	---	---	---------------	---------------	---------------

Example D -automaton \mathcal{A}



x, y – variables of \mathcal{A}

no states

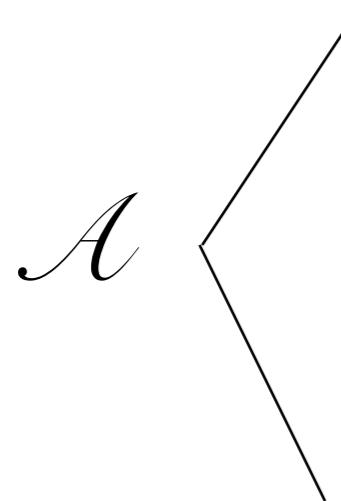


a finite run over \mathbb{Q} :

$a a a b a b a a a a a$

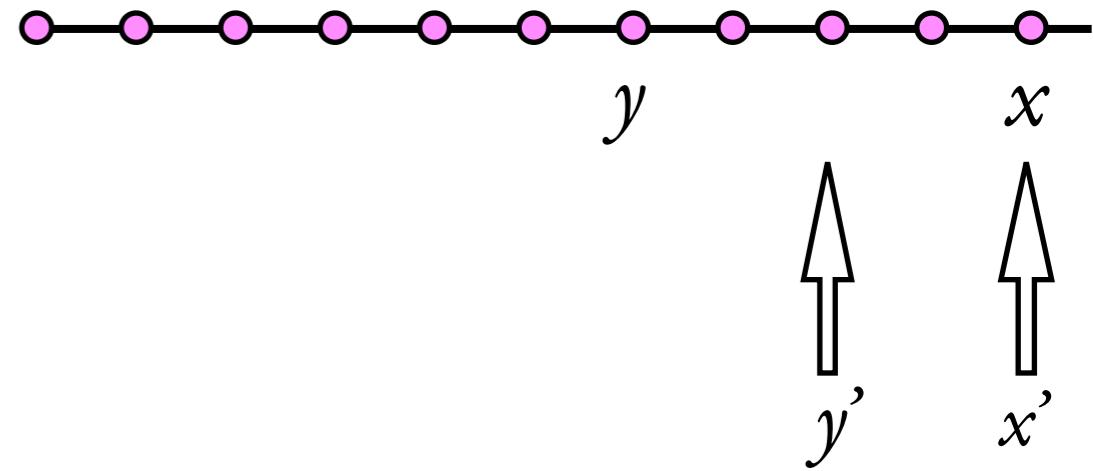
x	5	5	5	5	5	5	5	5	5	5	
y	0	1	2	4	0	3	0	$\frac{1}{4}$	$\frac{1}{2}$	$\frac{2}{3}$	$\frac{4}{5}$

Example D -automaton \mathcal{A}



x, y – variables of \mathcal{A}

no states



$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

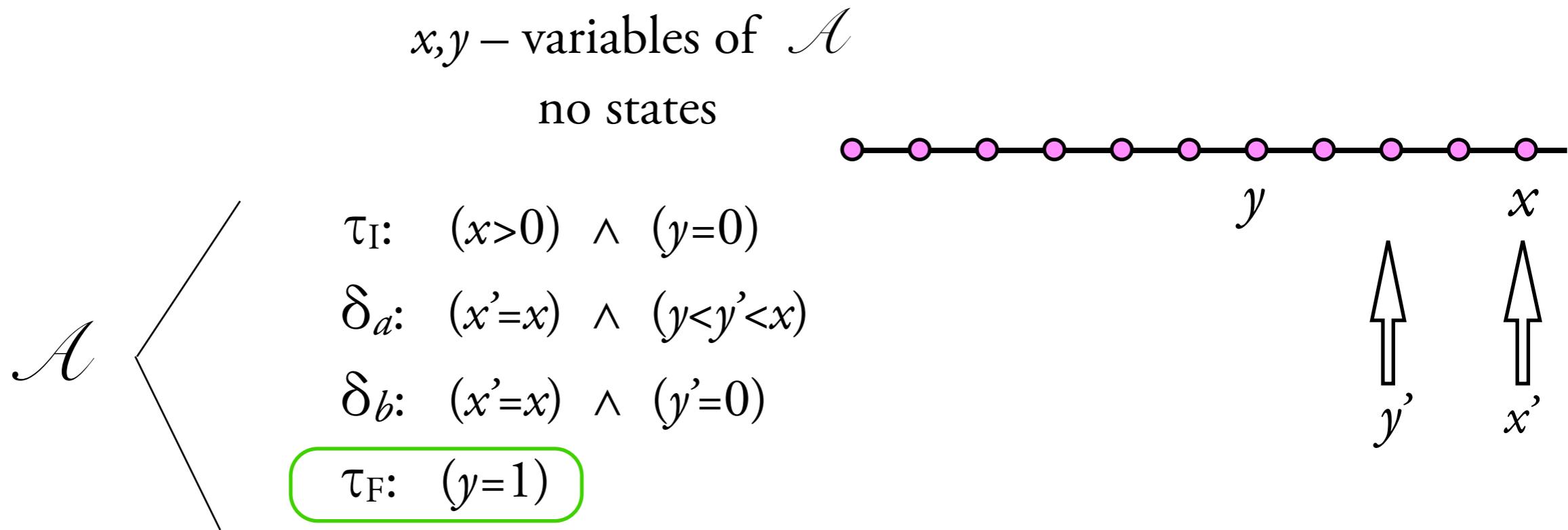
$$\tau_F: (y = 1)$$

a finite run over \mathbb{Q} :

$a a a b a b a a a a a$

x	5	5	5	5	5	5	5	5	5	5
y	0	1	2	4	0	3	0	$\frac{1}{4}$	$\frac{1}{2}$	$\frac{2}{3}$

Example D -automaton \mathcal{A}



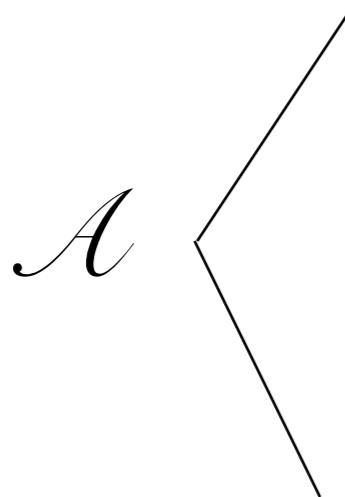
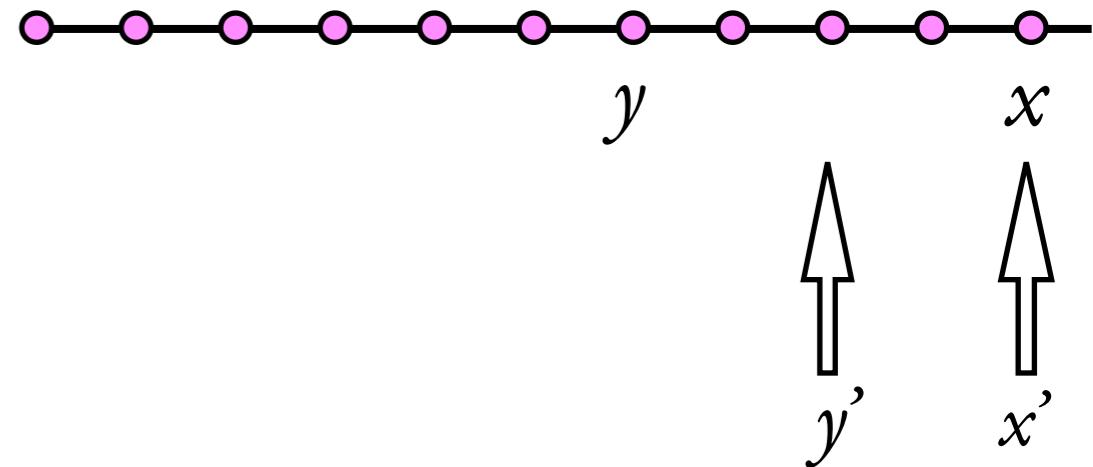
a finite run over \mathbb{Q} :

a	a	a	b	a	b	a	a	a	a
x	5	5	5	5	5	5	5	5	5
y	0	1	2	4	0	3	0	$\frac{1}{4}$	$\frac{1}{2}$

Example D -automaton \mathcal{A}

x, y – variables of \mathcal{A}

no states



$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$

a finite run over \mathbb{Q} :

$a a a b a b a a a a a$

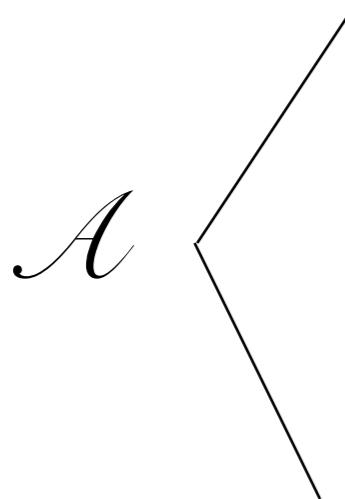
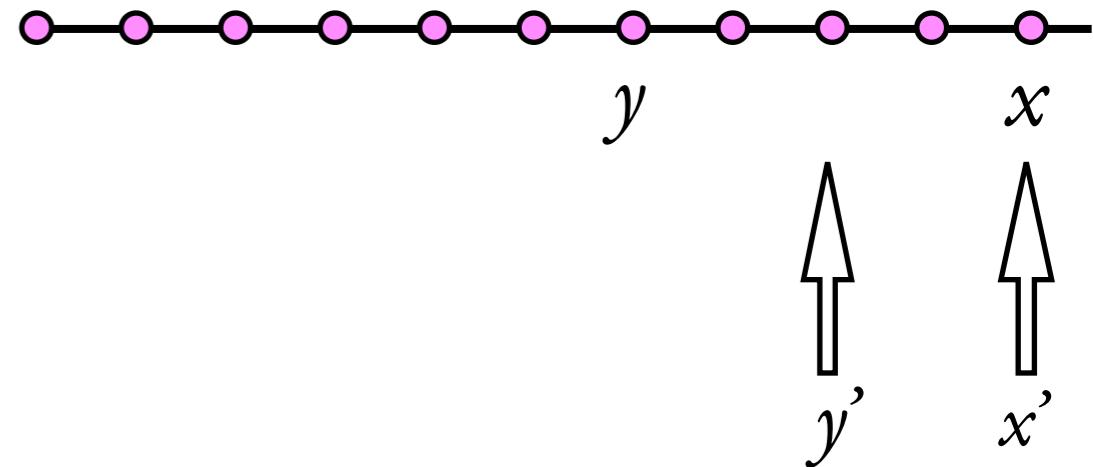
x	5	5	5	5	5	5	5	5	5	5	5
y	0	1	2	4	0	3	0	$\frac{1}{4}$	$\frac{1}{2}$	$\frac{2}{3}$	$\frac{4}{5}$

accepted language: $(a+b)^*a$

Example D -automaton \mathcal{A}

x, y – variables of \mathcal{A}

no states



$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$

a finite run over \mathbb{Q} :

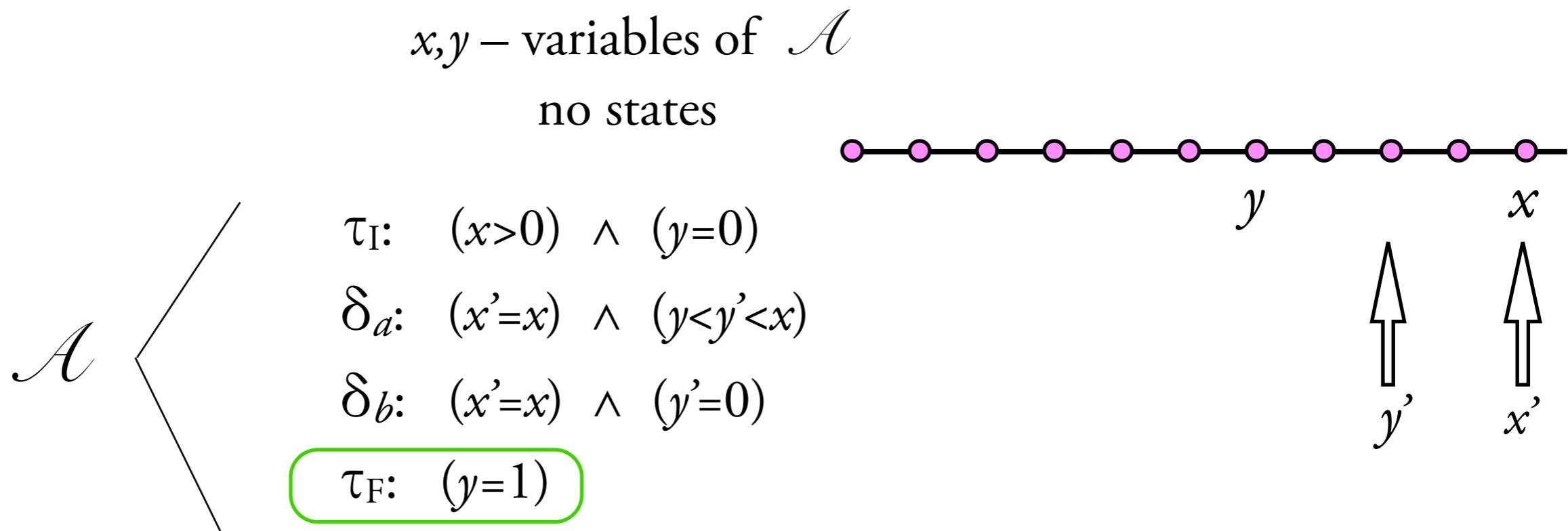
$a a a b a b a a a a a$

x	5	5	5	5	5	5	5	5	5	5	5
y	0	1	2	4	0	3	0	$\frac{1}{4}$	$\frac{1}{2}$	$\frac{2}{3}$	$\frac{4}{5}$

a finite run over \mathbb{N} :

accepted language: $(a+b)^*a$

Example D -automaton \mathcal{A}



a finite run over \mathbb{Q} :

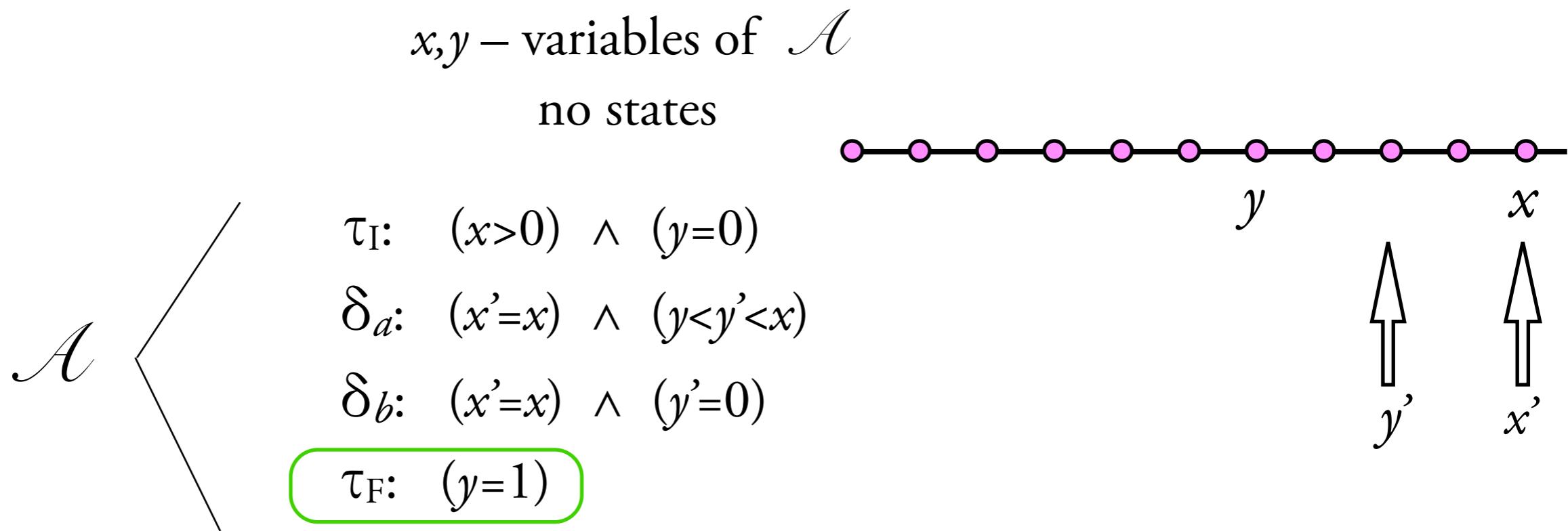
a	a	a	b	a	b	a	a	a	a
x	5	5	5	5	5	5	5	5	5
y	0	1	2	4	0	3	0	$\frac{1}{4}$	$\frac{1}{2}$

accepted language: $(a+b)^*a$

a finite run over \mathbb{N} :

a a a b a b a a a a

Example D -automaton \mathcal{A}



a finite run over \mathbb{Q} :

a	a	a	b	a	b	a	a	a	a
x	5	5	5	5	5	5	5	5	5
y	0	1	2	4	0	3	0	$\frac{1}{4}$	$\frac{1}{2}$

accepted language: $(a+b)^*a$

a finite run over \mathbb{N} :

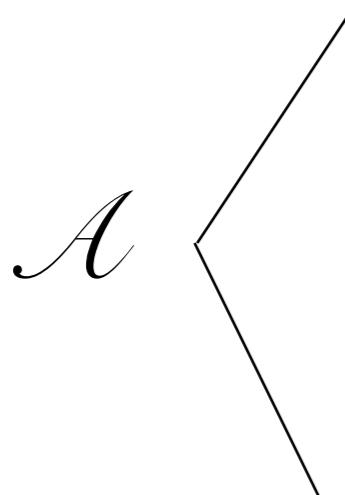
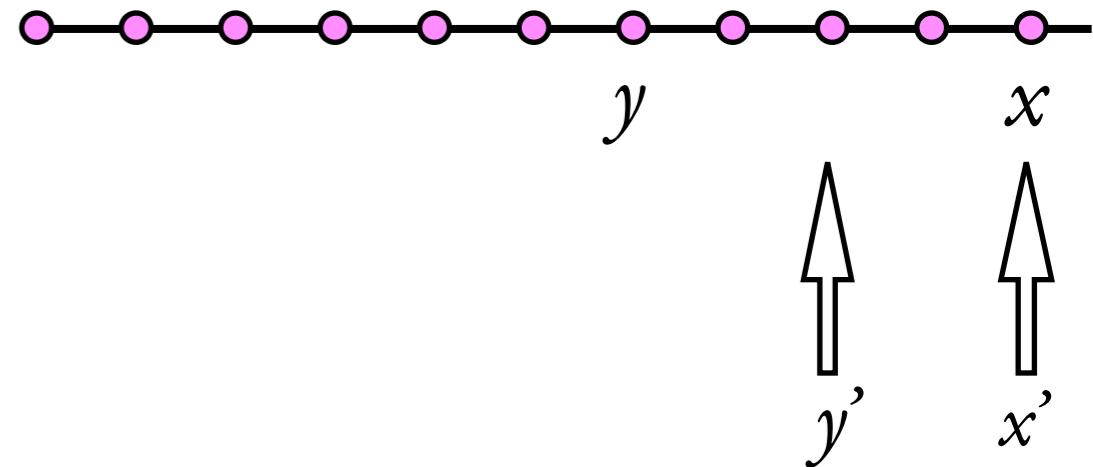
a a a b a b a a a a

x
 y

Example D -automaton \mathcal{A}

x, y – variables of \mathcal{A}

no states



$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$

a finite run over \mathbb{Q} :

$a a a b a b a a a a a$

x	5	5	5	5	5	5	5	5	5	5	5	5
y	0	1	2	4	0	3	0	$\frac{1}{4}$	$\frac{1}{2}$	$\frac{2}{3}$	$\frac{4}{5}$	1

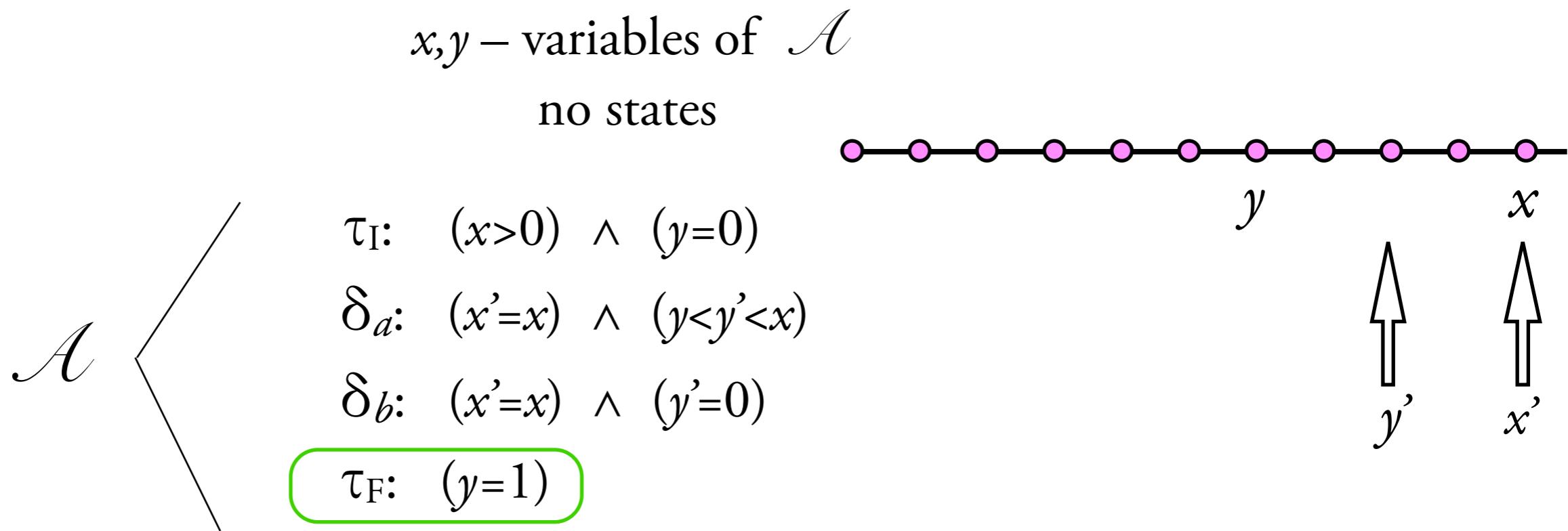
a finite run over \mathbb{N} :

$a a a b a b a a a a a$

x	5	5	5	5	5	5	5	5	5	5	5	5
y	0	1	2	4	0	3	0	1	2	3	4	

accepted language: $(a+b)^*a$

Example D -automaton \mathcal{A}



a finite run over \mathbb{Q} :

a	a	a	b	a	b	a	a	a	a
x	5	5	5	5	5	5	5	5	5
y	0	1	2	4	0	3	0	$\frac{1}{4}$	$\frac{1}{2}$

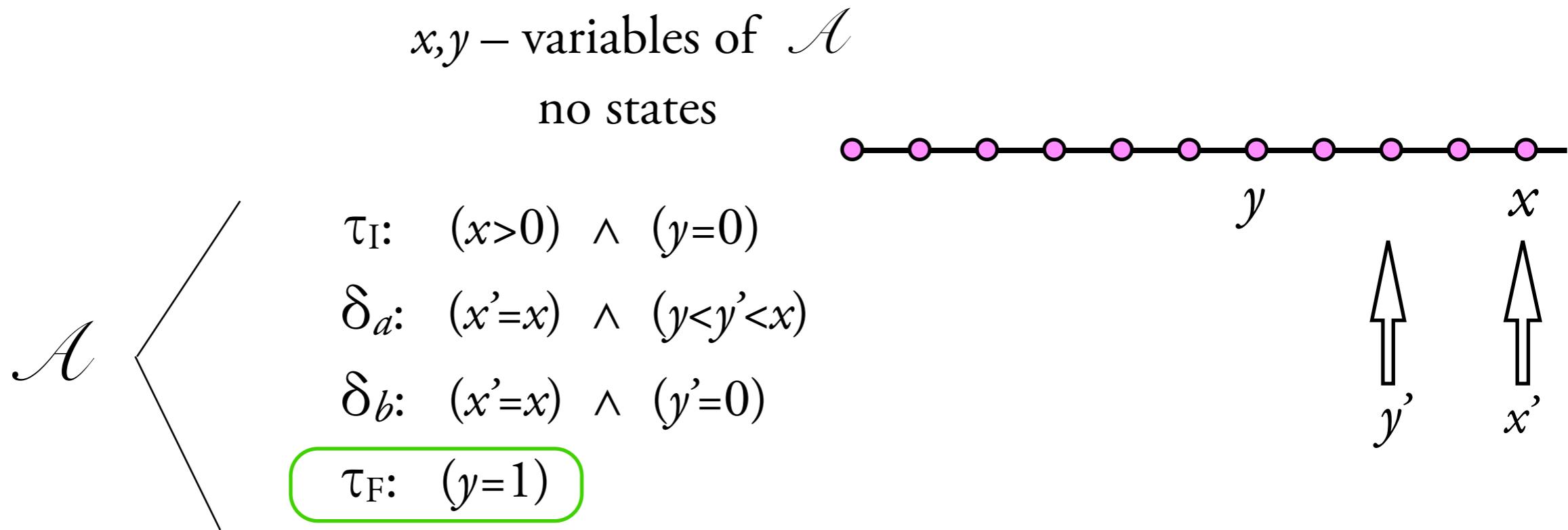
accepted language: $(a+b)^*a$

a finite run over \mathbb{N} :

a	a	a	b	a	b	a	a	a
x	5	5	5	5	5	5	5	5
y	0	1	2	4	0	3	0	1

accepted language: $(a^*b)^*a$

Example D -automaton \mathcal{A}



a finite run over \mathbb{Q} :

a	a	a	b	a	b	a	a	a	a
x	5	5	5	5	5	5	5	5	5
y	0	1	2	4	0	3	0	$\frac{1}{4}$	$\frac{1}{2}$

accepted language: $(a+b)^*a$

a finite run over \mathbb{N} :

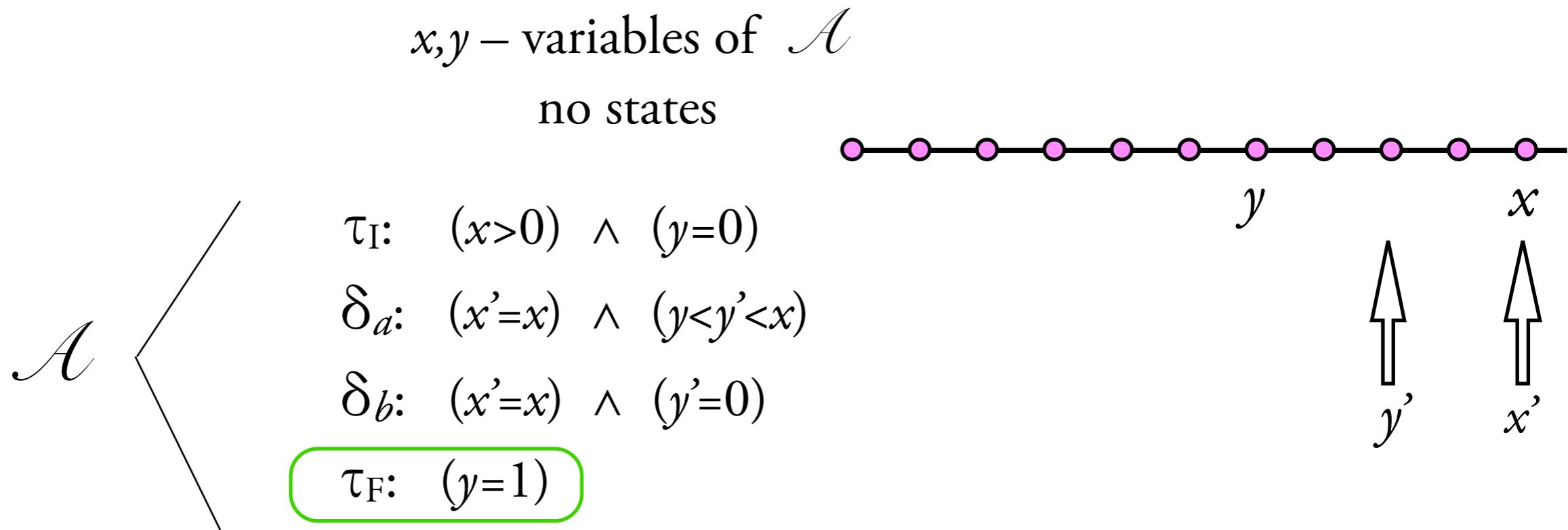
a	a	a	b	a	b	a	a	a
x	5	5	5	5	5	5	5	5
y	0	1	2	4	0	3	0	1

accepted language: $(a^*b)^*a$

an infinite run over \mathbb{N} :

a	a	a	b	a	b	a	a	a
x	5	5	5	5	5	5	5	5
y	0	1	2	4	0	3	0	1

Example D -automaton \mathcal{A}



a finite run over \mathbb{Q} :

a	a	a	b	a	b	a	a	a	a
x	5	5	5	5	5	5	5	5	5
y	0	1	2	4	0	3	0	$\frac{1}{4}$	$\frac{1}{2}$

accepted language: $(a+b)^*a$

a finite run over \mathbb{N} :

a	a	a	b	a	b	a	a	a
x	5	5	5	5	5	5	5	5
y	0	1	2	4	0	3	0	1

accepted language: $(a^*b)^*a$

an infinite run over \mathbb{N} :

a	a	a	b	a	b	a	a	a
x	5	5	5	5	5	5	5	5
y	0	1	2	4	0	3	0	1

accepted language: $(a^B b)^\omega$

Adding database constraints

x, y, z, \dots – a finite set of variables

R, S, \dots – a finite set of relational symbols

$\tau_I: (x=0)$

$\delta_a: (x' > x) \wedge R(x) \wedge \neg S(x, y')$

$\delta_b: (x' = 0)$

$\tau_F: (x=1)$

Adding database constraints

x, y, z, \dots – a finite set of variables

R, S, \dots – a finite set of relational symbols

$\tau_I: (x=0)$

$\delta_a: (x' > x) \wedge R(x) \wedge \neg S(x, y')$

$\delta_b: (x' = 0)$

$\tau_F: (x=1)$

Adding database constraints

x, y, z, \dots – a finite set of variables

R, S, \dots – a finite set of relational symbols

$\tau_I: (x=0)$

$\delta_a: (x' > x) \wedge R(x) \wedge \neg S(x, y')$

$\delta_b: (x' = 0)$

$\tau_F: (x=1)$

EMPTINESS: is there a *finite* database M , a word w ,
and an accepting run over w consistent with M ?

Adding database constraints

x, y, z, \dots – a finite set of variables

R, S, \dots – a finite set of relational symbols

$\tau_I: (x=0)$

$\delta_a: (x' > x) \wedge R(x) \wedge \neg S(x, y')$

$\delta_b: (x' = 0)$

$\tau_F: (x=1)$

EMPTINESS: is there a *finite* database M , a word w , and an accepting run over w consistent with M ?

a finite run over \mathbb{N} :

Adding database constraints

x, y, z, \dots – a finite set of variables

R, S, \dots – a finite set of relational symbols

$\tau_I: (x=0)$

$\delta_a: (x' > x) \wedge R(x) \wedge \neg S(x, y')$

$\delta_b: (x' = 0)$

$\tau_F: (x=1)$

EMPTINESS: is there a *finite* database M , a word w , and an accepting run over w consistent with M ?

a finite run over \mathbb{N} :

$a a a b a b a$

Adding database constraints

x, y, z, \dots – a finite set of variables

R, S, \dots – a finite set of relational symbols

$\tau_I: (x=0)$

$\delta_a: (x' > x) \wedge R(x) \wedge \neg S(x, y')$

$\delta_b: (x' = 0)$

$\tau_F: (x=1)$

EMPTINESS: is there a *finite* database M , a word w , and an accepting run over w consistent with M ?

a finite run over \mathbb{N} :

$a a a b a b a$

$R = \{(0), (1), (2), (3), (4)\}$ $S = \{\}$

Adding database constraints

x, y, z, \dots – a finite set of variables

R, S, \dots – a finite set of relational symbols

$\tau_I: (x=0)$

$\delta_a: (x' > x) \wedge R(x) \wedge \neg S(x, y')$

$\delta_b: (x' = 0)$

$\tau_F: (x=1)$

EMPTINESS: is there a *finite* database M , a word w , and an accepting run over w consistent with M ?

a finite run over \mathbb{N} :

$a a a b a b a$

$x \quad 0 \ 2 \ 3 \ 4 \ 0 \ 3 \ 0 \ 1$

$R = \{(0), (1), (2), (3), (4)\} \quad S = \{\}$

Adding database constraints

x, y, z, \dots – a finite set of variables

R, S, \dots – a finite set of relational symbols

$\tau_I: (x=0)$

$\delta_a: (x' > x) \wedge R(x) \wedge \neg S(x, y')$

$\delta_b: (x' = 0)$

$\tau_F: (x=1)$

EMPTINESS: is there a *finite* database M , a word w , and an accepting run over w consistent with M ?

a finite run over \mathbb{N} :

$a a a b a b a$

$x \quad 0 \ 2 \ 3 \ 4 \ 0 \ 3 \ 0 \ 1$

$R=\{(0), (1), (2), (3), (4)\} \quad S=\{\}$

accepted language: $(a^*b)^*a$

Adding database constraints

x, y, z, \dots – a finite set of variables

R, S, \dots – a finite set of relational symbols

$\tau_I: (x=0)$

$\delta_a: (x' > x) \wedge R(x) \wedge \neg S(x, y')$

$\delta_b: (x' = 0)$

$\tau_F: (x=1)$

EMPTINESS: is there a *finite* database M , a word w , and an accepting run over w consistent with M ?

a finite run over \mathbb{N} :

$a a a b a b a$

$x \quad 0 \ 2 \ 3 \ 4 \ 0 \ 3 \ 0 \ 1$

$R=\{(0), (1), (2), (3), (4)\} \quad S=\{\}$

an infinite run over \mathbb{N} :

accepted language: $(a^*b)^*a$

Adding database constraints

x, y, z, \dots – a finite set of variables

R, S, \dots – a finite set of relational symbols

$\tau_I: (x=0)$

$\delta_a: (x' > x) \wedge R(x) \wedge \neg S(x, y')$

$\delta_b: (x' = 0)$

$\tau_F: (x=1)$

EMPTINESS: is there a *finite* database M , a word w , and an accepting run over w consistent with M ?

a finite run over \mathbb{N} :

$a a a b a b a$

$x \quad 0 \ 2 \ 3 \ 4 \ 0 \ 3 \ 0 \ 1$

$R=\{(0), (1), (2), (3), (4)\} \quad S=\{\}$

accepted language: $(a^*b)^*a$

an infinite run over \mathbb{N} :

$a a a b a b a a b a a \dots$

Adding database constraints

x, y, z, \dots – a finite set of variables

R, S, \dots – a finite set of relational symbols

$\tau_I: (x=0)$

$\delta_a: (x' > x) \wedge R(x) \wedge \neg S(x, y')$

$\delta_b: (x' = 0)$

$\tau_F: (x=1)$

EMPTINESS: is there a *finite* database M , a word w , and an accepting run over w consistent with M ?

a finite run over \mathbb{N} :

$a a a b a b a$
 $x \quad 0 \ 2 \ 3 \ 4 \ 0 \ 3 \ 0 \ 1$

$R=\{(0),(1),(2),(3),(4)\} \quad S=\{\}$

accepted language: $(a^*b)^*a$

an infinite run over \mathbb{N} :

$a a a b a b a a b a a \dots$
 $x \quad 0 \ 1 \ 3 \ 4 \ 0 \ 3 \ 0 \ 1 \ 4 \ 0 \ 1 \ 3$

Adding database constraints

x, y, z, \dots – a finite set of variables

R, S, \dots – a finite set of relational symbols

$\tau_I: (x=0)$

$\delta_a: (x' > x) \wedge R(x) \wedge \neg S(x, y')$

$\delta_b: (x' = 0)$

$\tau_F: (x=1)$

EMPTINESS: is there a *finite* database M , a word w , and an accepting run over w consistent with M ?

a finite run over \mathbb{N} :

$a a a b a b a$
 $x \quad 0 \ 2 \ 3 \ 4 \ 0 \ 3 \ 0 \ 1$

$R=\{(0),(1),(2),(3),(4)\} \quad S=\{\}$

accepted language: $(a^*b)^*a$

an infinite run over \mathbb{N} :

$a a a b a b a a b a a \dots$
 $x \quad 0 \ 1 \ 3 \ 4 \ 0 \ 3 \ 0 \ 1 \ 4 \ 0 \ 1 \ 3$

$R=\{(0),(1),(2),(3),(4)\} \quad S=\{\}$

Adding database constraints

x, y, z, \dots – a finite set of variables

R, S, \dots – a finite set of relational symbols

$\tau_I: (x=0)$

$\delta_a: (x' > x) \wedge R(x) \wedge \neg S(x, y')$

$\delta_b: (x' = 0)$

$\tau_F: (x=1)$

EMPTINESS: is there a *finite* database M , a word w , and an accepting run over w consistent with M ?

a finite run over \mathbb{N} :

$a a a b a b a$
 $x \quad 0 \ 2 \ 3 \ 4 \ 0 \ 3 \ 0 \ 1$

$R=\{(0),(1),(2),(3),(4)\} \quad S=\{\}$

accepted language: $(a^*b)^*a$

an infinite run over \mathbb{N} :

$a a a b a b a a b a a \dots$
 $x \quad 0 \ 1 \ 3 \ 4 \ 0 \ 3 \ 0 \ 1 \ 4 \ 0 \ 1 \ 3$

$R=\{(0),(1),(2),(3),(4)\} \quad S=\{\}$

accepted language: $(a^B b)^\omega$

Adding database constraints

x, y, z, \dots – a finite set of variables
 R, S, \dots – a finite set of relational symbols

$\tau_I: (x=0)$

$\delta_a: (x > x) \wedge R(x) \wedge \neg S(x, y)$
 $\delta_b: (x = 0)$

Decide emptiness

$\tau_F: (x = 1)$

of D -automata

EMPTINESS: is there a finite database M , a word w ,
over ω -words, with databases
and an accepting run over w consistent with M ?

a finite run over \mathbb{N} :

$a a a b a b a$
 $x \quad 0 \ 2 \ 3 \ 4 \ 0 \ 3 \ 0 \ 1$

$R = \{(0), (1), (2), (3), (4)\}$ $S = \{\}$

accepted language: $(a^*b)^*a$

an infinite run over \mathbb{N} :

$a a a b a b a a b a a \dots$
 $x \quad 0 \ 1 \ 3 \ 4 \ 0 \ 3 \ 0 \ 1 \ 4 \ 0 \ 1 \ 3$

$R = \{(0), (1), (2), (3), (4)\}$ $S = \{\}$

accepted language: $(a^B b)^\omega$

Known results deciding emptiness

	no database	with database
$\mathbb{N}, =$	Kaminski, Francez (94)	Deutsch, Sui, Vianu, Zhou (06) PSPACE
$\mathbb{Q}, <$	Čerans (94) PSPACE	Deutsch, Hull, Patrizi, Vianu (09) PSPACE
$\mathbb{N}, <$	Čerans (94) NONPRIMITIVE	?

Known results deciding emptiness

	no database	with database
$\mathbb{N}, =$	Kaminski, Francez (94)	Deutsch, Sui, Vianu, Zhou (06) PSPACE
$\mathbb{Q}, <$	Čerans (94) PSPACE	Deutsch, Hull, Patrizi, Vianu (09) PSPACE
$\mathbb{N}, <$	Čerans (94) NONPRIMITIVE	?
$\mathcal{D}, <$	PSPACE	PSPACE

Known results deciding emptiness

	no database	with database
$\mathbb{N}, =$	Kaminski, Francez (94)	Deutsch, Sui, Vianu, Zhou (06) PSPACE
$\mathbb{Q}, <$	Čerans (94) PSPACE	Deutsch, Hull, Patrizi, Vianu (09) PSPACE
$\mathbb{N}, <$	Čerans (94) NONPRIMITIVE	PSPACE
$\mathcal{D}, <$	PSPACE	PSPACE

Known results deciding emptiness

	no database	with database
$\mathbb{N}, =$	Kaminski, Francez (94)	Deutsch, Sui, Vianu, Zhou (06) PSPACE
$\mathbb{Q}, <$	Čerans (94) PSPACE	Deutsch, Hull, Patrizi, Vianu (09) PSPACE
$\mathbb{N}, <$	PSPACE	PSPACE
$\mathcal{D}, <$	PSPACE	PSPACE

Dense case

- $D = \mathbb{Q}$
- finite words
- no database

Dense case

- $D = \mathbb{Q}$
- finite words
- no database

$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$

Dense case

- $D = \mathbb{Q}$
- finite words
- no database

$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$

$0 < y < 1 < x, \quad 0 < x < 1 = y, \quad 0 \leq 1 < y < x, \quad 0 < 1 \leq y < x, \quad 0 = y < x, \quad 0 < y < 1 = x, \dots$

Dense case

- $D = \mathbb{Q}$
- finite words
- no database

$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$

$0 < y < 1 < x, \quad 0 < x < 1 = y, \quad 0 \leq 1 < y \leq x, \quad 0 < 1 \leq y < x, \quad 0 \leq y < x, \quad 0 = y < x, \quad \dots$



Dense case

- $D = \mathbb{Q}$
- finite words
- no database

$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$

$0 < y < 1 < x, \quad 0 < x < 1 = y, \quad 0 \leq 1 < y < x, \quad 0 < 1 \leq y < x, \quad 0 = y < 1 < x, \quad 0 < y < 1 = x, \dots$



$$\Psi_{\exists(x,y)}$$

Dense case

- $D = \mathbb{Q}$
- finite words
- no database

$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$

$0 < y < 1 < x, \quad 0 < x < 1 = y, \quad 0 \leq 1 < y < x, \quad 0 < 1 \leq y < x, \quad 0 = y < 1 < x, \quad 0 < y < 1 = x, \dots$

$$\Psi(x', y') \quad \Psi(x, y)$$

\curvearrowleft_a

Dense case

- $D = \mathbb{Q}$
- finite words
- no database

$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$

$0 < y < 1 < x, \quad 0 < x < 1 = y, \quad 0 \leq 1 < y < x, \quad 0 < 1 = y < x, \quad 0 = y < 1 < x, \quad 0 = y < 1 = x, \dots$

The region on construction

$$\exists(x', y'): \quad \exists(x, y): \quad \delta_a(x, y, x', y')$$

Dense case

- $D = \mathbb{Q}$
- finite words
- no database

$$\tau_I: (x > 0) \wedge (y = 0)$$

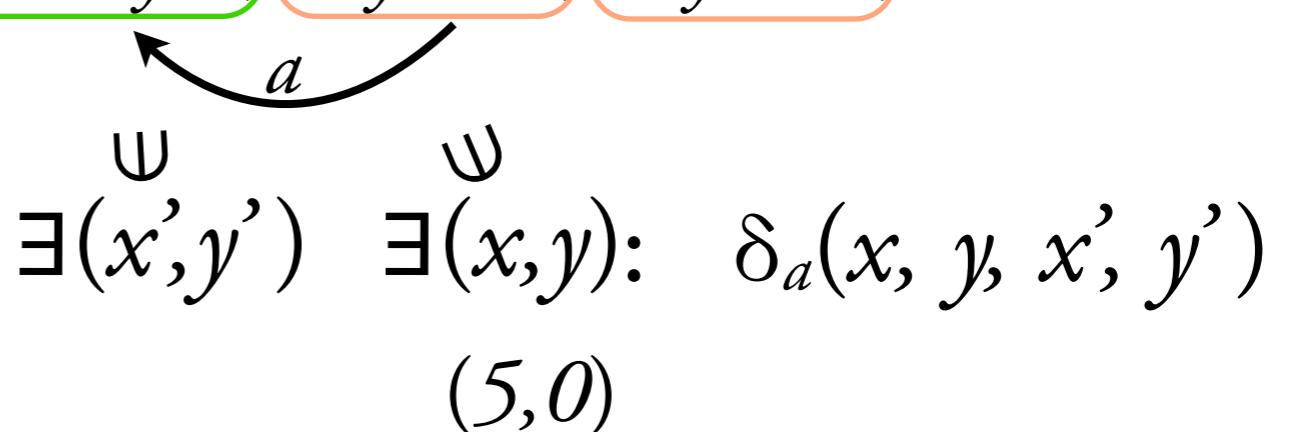
$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$

$0 < y < 1 < x, \quad 0 < x < 1 = y, \quad 0 \leq 1 < y \leq x, \quad 0 < 1 \leq y < x, \quad 0 \leq y < x, \quad 0 = y < 1 = x, \dots$

The region on construction


$$\Psi(x', y') \quad \Psi(x, y): \quad \delta_a(x, y, x', y')$$
$$(5, 0)$$

Dense case

- $D = \mathbb{Q}$
- finite words
- no database

$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$

$0 < y < 1 < x, \quad 0 < x < 1 = y, \quad 0 \leq 1 < y < x, \quad 0 < 1 = y < x, \quad 0 = y < 1 < x, \quad \dots$

The region on construction

$$\exists(x', y'): \quad \exists(x, y): \quad \delta_a(x, y, x', y')$$

$(5, 1) \quad (5, 0)$

Dense case

- $D = \mathbb{Q}$
- finite words
- no database

$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$

$0 < y < 1 < x, \quad 0 < x < 1 = y, \quad 0 \leq 1 < y \leq x, \quad 0 < 1 \leq y < x, \quad 0 \leq y < x, \quad 0 = y < x, \quad \dots$



Dense case

- $D = \mathbb{Q}$
- finite words
- no database

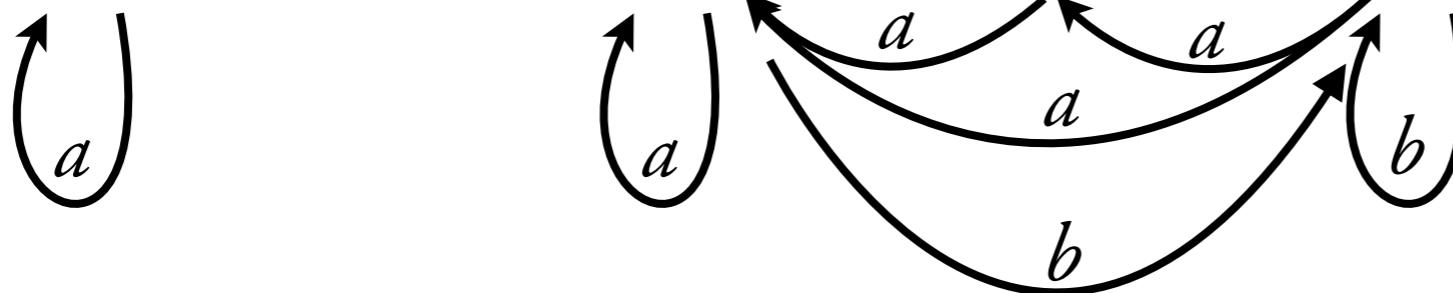
$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$

$0 < y < 1 < x, 0 < x < 1 = y, 0 \leq 1 < y < x, 0 < 1 = y < x, 0 = y < 1 < x, 0 = y < 1 = x, \dots$



Dense case

- $D = \mathbb{Q}$
- finite words
- no database

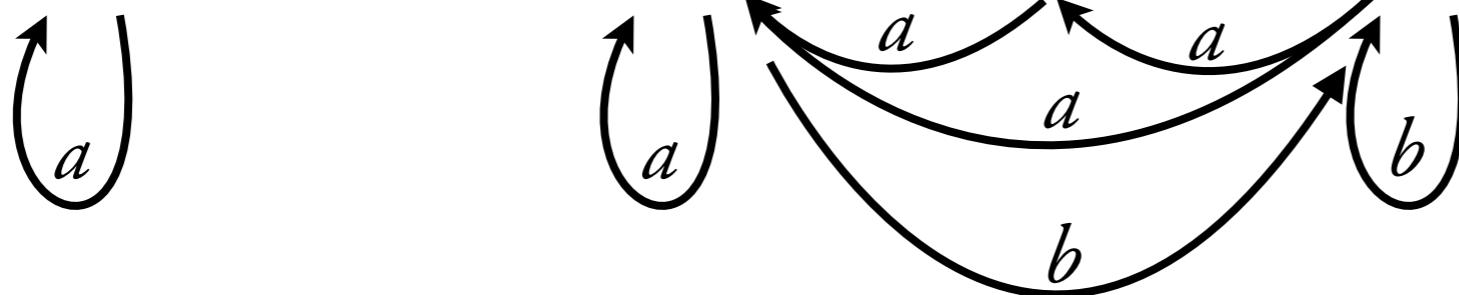
$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$

$0 < y < 1 < x, 0 < x < 1 = y, 0 \leq 1 < y < x, 0 < 1 = y < x, 0 \leq y < 1 < x, 0 = y < 1 = x, \dots$



bisimulation \rightarrow runs of the region automaton correspond to runs of \mathcal{A}

Dense case

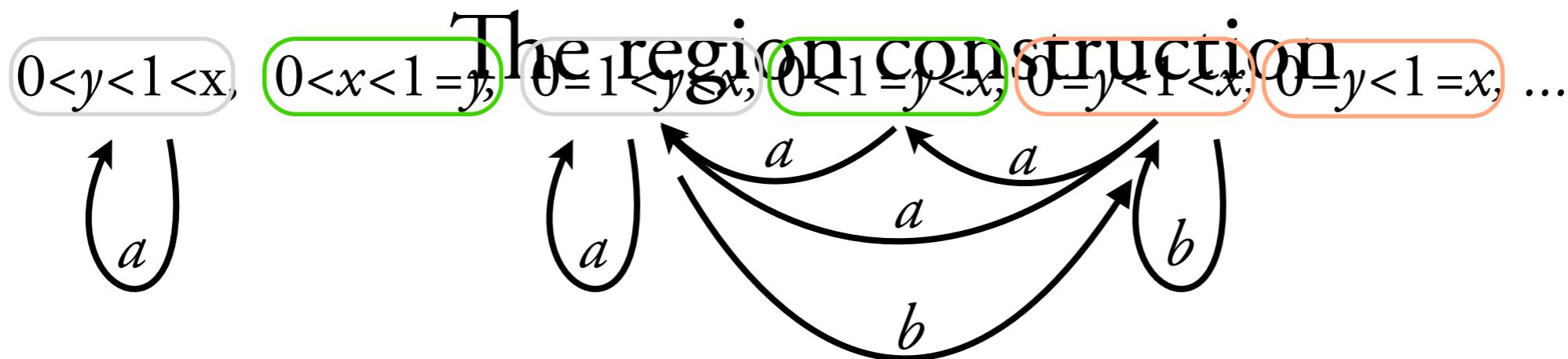
- $D = \mathbb{Q}$
- finite words
- no database

$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$



a a a b a b a a a a a

bisimulation \rightarrow runs of the region
automaton correspond to runs of \mathcal{A}

Dense case

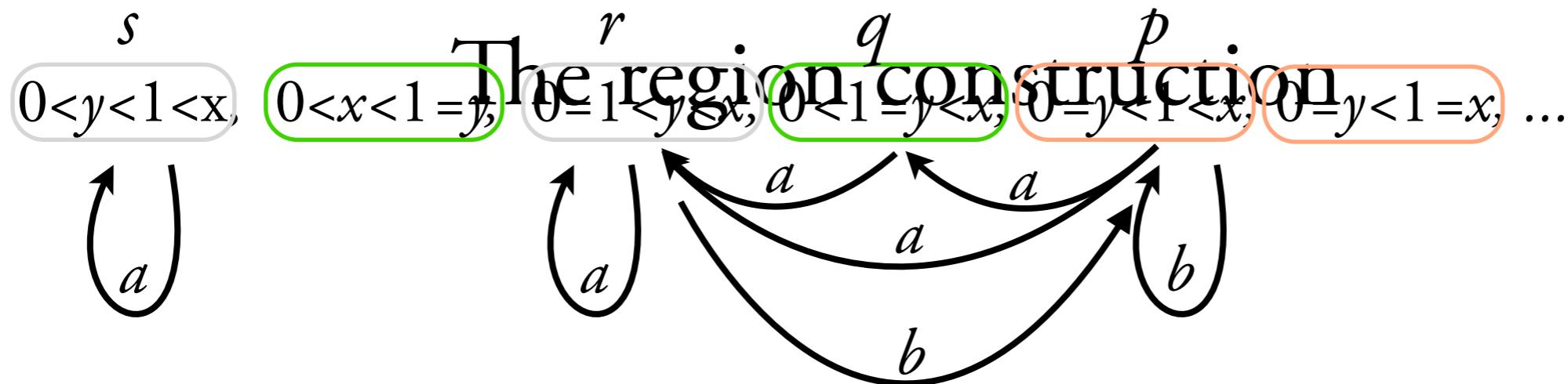
- $D = \mathbb{Q}$
- finite words
- no database

$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$



a a a b a b a a a a a

bisimulation \rightarrow runs of the region automaton correspond to runs of \mathcal{A}

Dense case

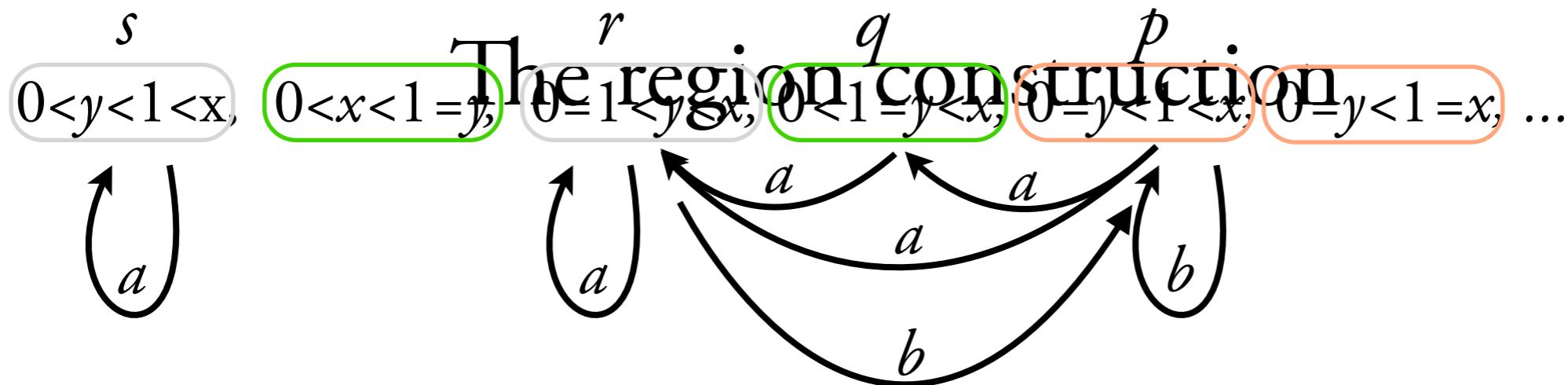
- $D = \mathbb{Q}$
- finite words
- no database

$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$



a	a	a	b	a	b	a	a	a	a
p	q	r	r	q	r	q	s	s	s

bisimulation \rightarrow runs of the region automaton correspond to runs of \mathcal{A}

Dense case

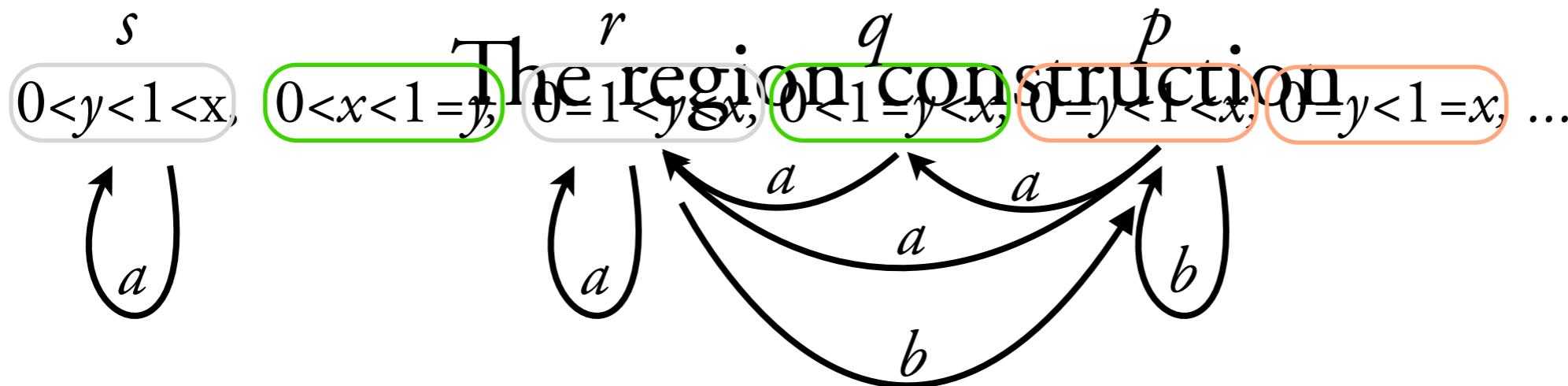
- $D = \mathbb{Q}$
- finite words
- no database

$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$



$a a a b a b a a a a a$
 $p q r r q r q s s s s q$

x
 y

bisimulation \rightarrow runs of the region automaton correspond to runs of \mathcal{A}

Dense case

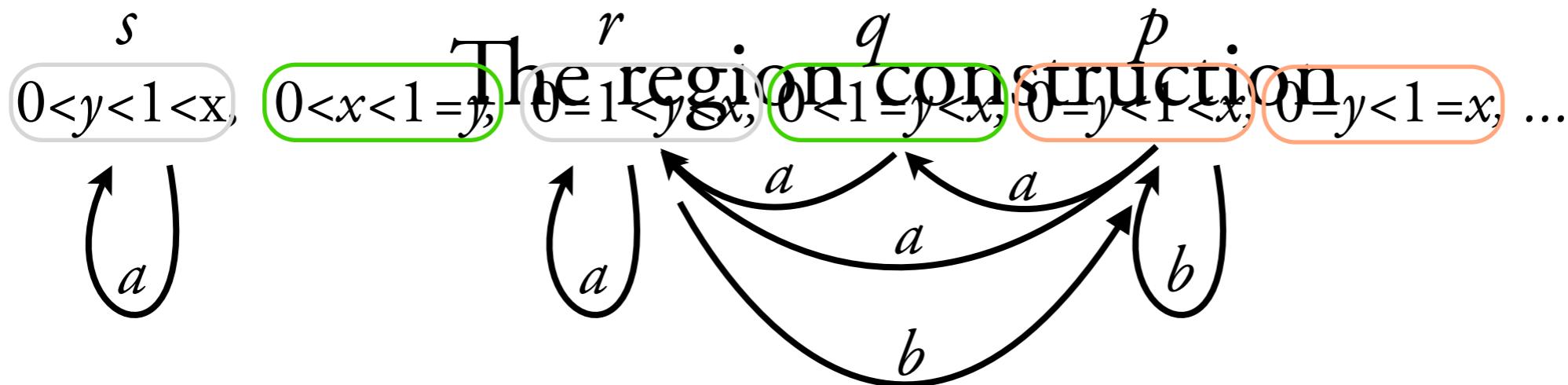
- $D = \mathbb{Q}$
- finite words
- no database

$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$



a	a	a	b	a	b	a	a	a	a
p	q	r	r	q	r	q	s	s	s

x	5
y	0

bisimulation \rightarrow runs of the region automaton correspond to runs of \mathcal{A}

Dense case

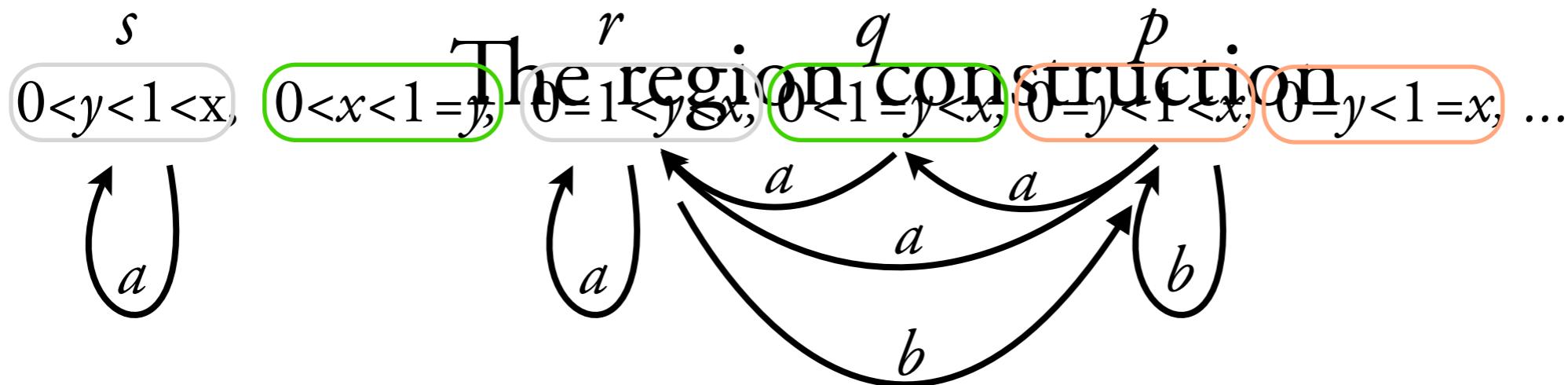
- $D = \mathbb{Q}$
- finite words
- no database

$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$



a	a	a	b	a	b	a	a	a	a
p	q	r	r	q	r	q	s	s	s

x	5	5
y	0	1

bisimulation \rightarrow runs of the region automaton correspond to runs of \mathcal{A}

Dense case

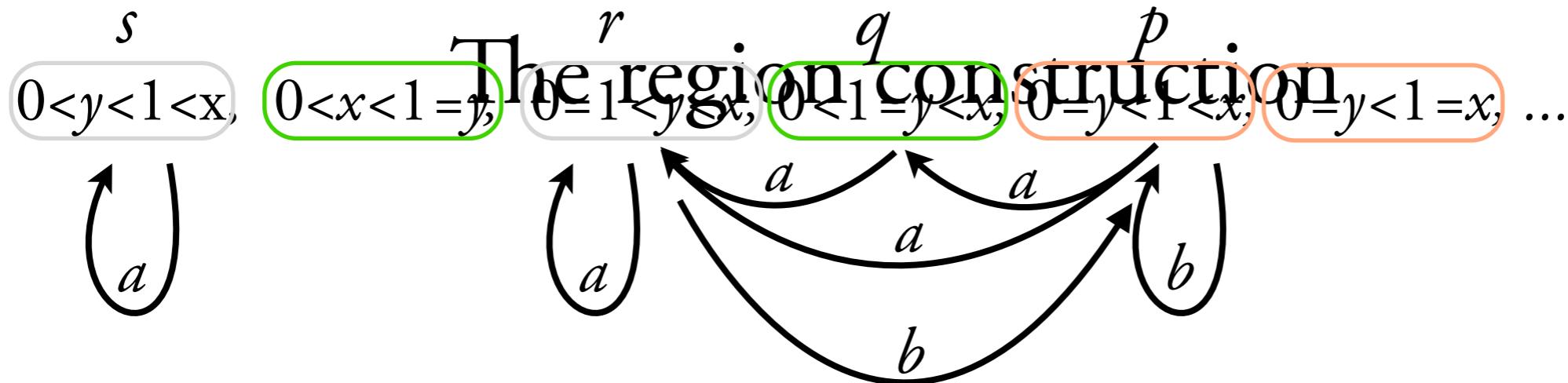
- $D = \mathbb{Q}$
- finite words
- no database

$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$



a	a	a	b	a	b	a	a	a	a
p	q	r	r	q	r	q	s	s	s

x	5	5	5
y	0	1	2

bisimulation \rightarrow runs of the region automaton correspond to runs of \mathcal{A}

Dense case

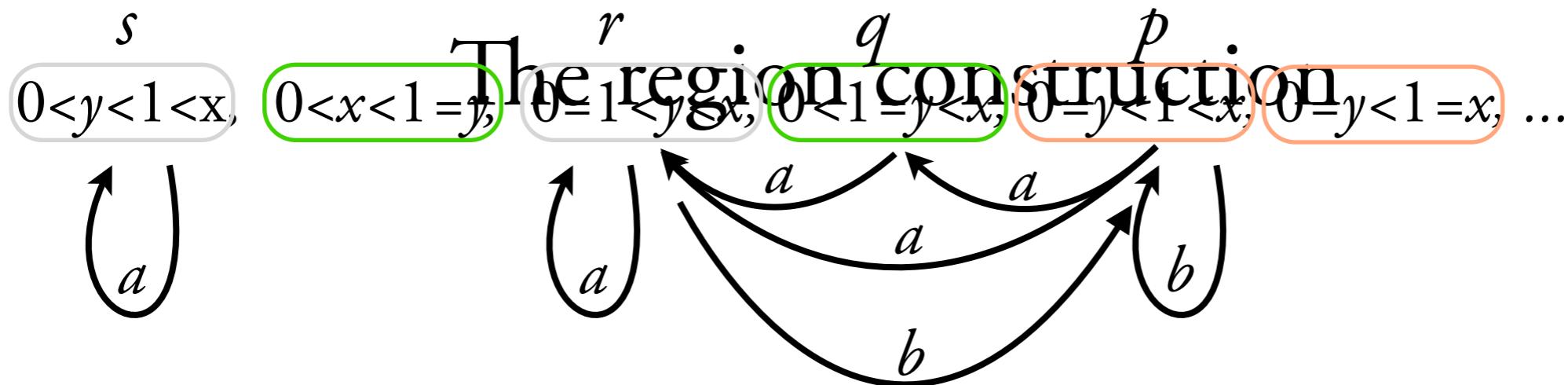
- $D = \mathbb{Q}$
- finite words
- no database

$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$



a	a	a	b	a	b	a	a	a	a
p	q	r	r	q	r	q	s	s	s

x	5	5	5	5
y	0	1	2	4

bisimulation \rightarrow runs of the region automaton correspond to runs of \mathcal{A}

Dense case

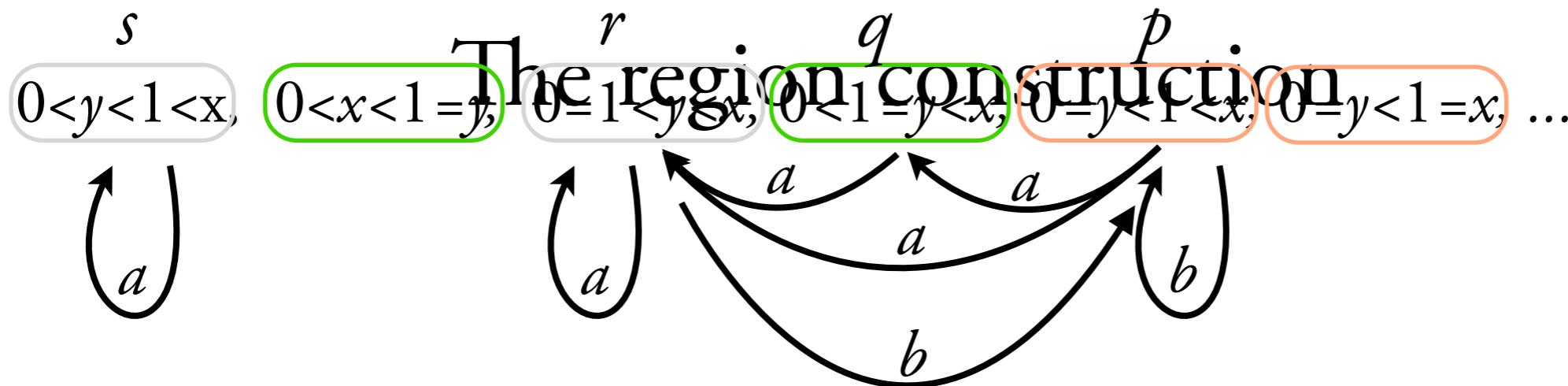
- $D = \mathbb{Q}$
- finite words
- no database

$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$



a	a	a	b	a	b	a	a	a	a
p	q	r	r	q	r	q	s	s	s

x	5	5	5	5	5
y	0	1	2	4	0

bisimulation \rightarrow runs of the region automaton correspond to runs of \mathcal{A}

Dense case

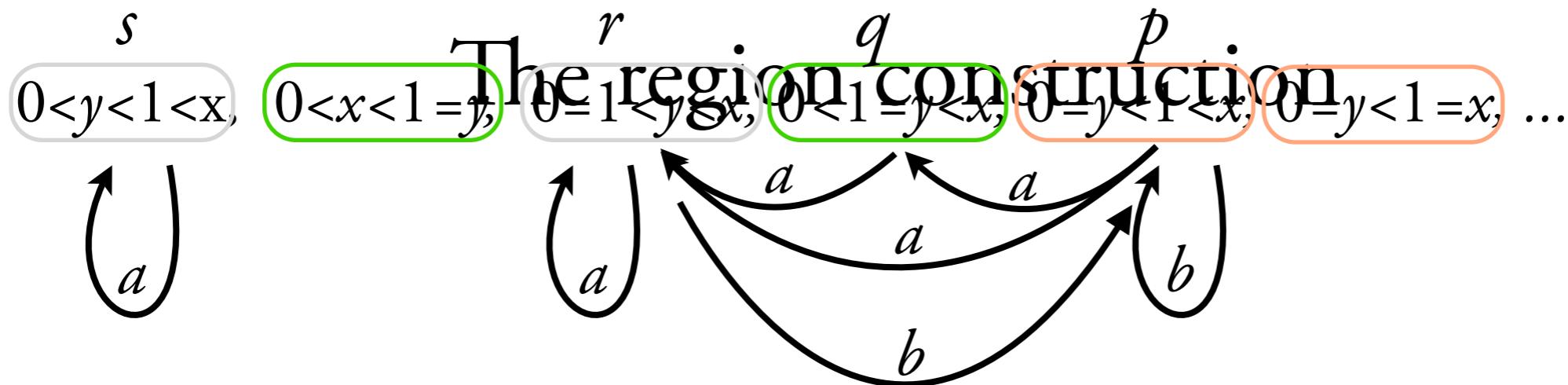
- $D = \mathbb{Q}$
- finite words
- no database

$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$



s	$0 < y < 1 < x,$
	$0 < x < 1 = y,$
	$0 \leq 1 < y < x,$
	$0 < 1 = y < x,$

$a a a b a b a a a a a$
 $p q r r q r q s s s s q$
 $x \quad 5 \ 5 \ 5 \ 5 \ 5 \ 5$
 $y \quad 0 \ 1 \ 2 \ 4 \ 0 \ 3$

bisimulation \rightarrow runs of the region automaton correspond to runs of \mathcal{A}

Dense case

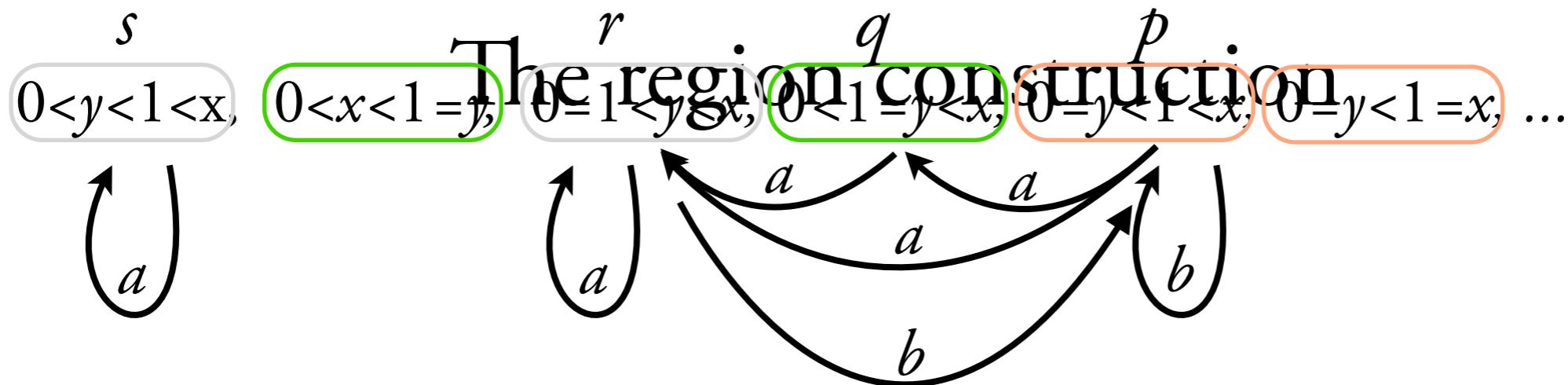
- $D = \mathbb{Q}$
- finite words
- no database

$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$



s	$0 < y < 1 < x,$
	$0 < x < 1 = y,$
	$0 \leq 1 < y < x,$
	$0 < 1 = y < x,$
	$0 \leq y < 1 < x,$
	$0 \leq y < 1 = x,$
	\dots

$a a a b a b a a a a a$
 $p q r r q r q s s s s q$

x	5 5 5 5 5 5 5
y	0 1 2 4 0 3 0

bisimulation \rightarrow runs of the region automaton correspond to runs of \mathcal{A}

Dense case

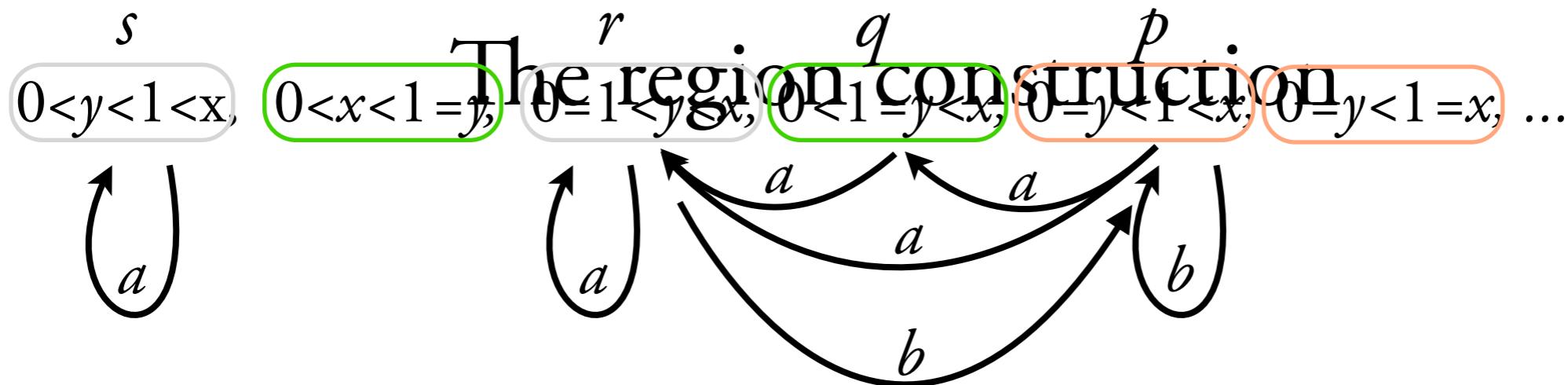
- $D = \mathbb{Q}$
- finite words
- no database

$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$



s	$0 < y < 1 < x,$
	$0 < x < 1 = y,$
	$0 \leq 1 < y < x,$
	$0 < 1 = y < x,$
	$0 \leq y < 1 < x,$
	$0 \leq y < 1 = x,$...

$a a a b a b a a a a a$
 $p q r r q r q s s s s q$

x	5 5 5 5 5 5 5 5
y	0 1 2 4 0 3 0 .1

bisimulation \rightarrow runs of the region automaton correspond to runs of \mathcal{A}

Dense case

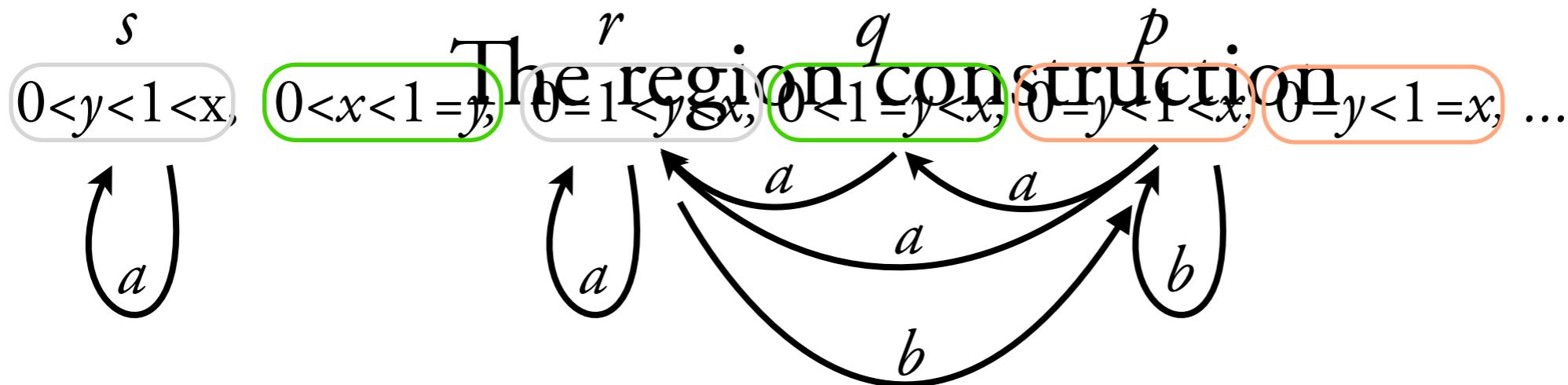
- $D = \mathbb{Q}$
- finite words
- no database

$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$



s	$0 < y < 1 < x,$
	$0 < x < 1 = y,$
	$0 \leq 1 < y < x,$
	$0 < 1 = y < x,$

a	$a a a b a b a a a a a$
p	$p q r r q r q s s s s q$
x	$5 5 5 5 5 5 5 5 5 5$
y	$0 1 2 4 0 3 0 .1 .2$

bisimulation \rightarrow runs of the region automaton correspond to runs of \mathcal{A}

Dense case

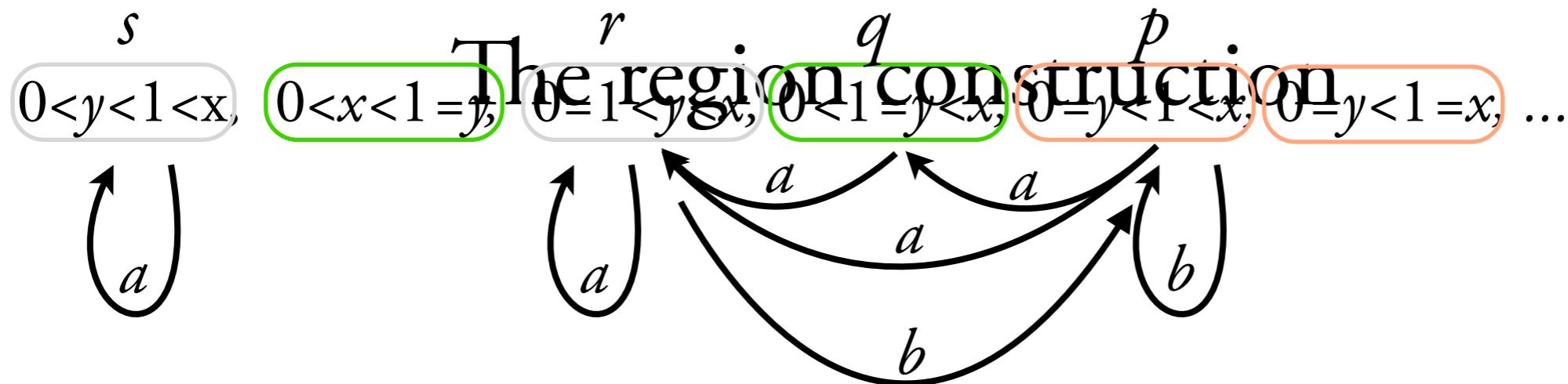
- $D = \mathbb{Q}$
- finite words
- no database

$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$



s	$0 < y < 1 < x,$
	$0 < x < 1 = y,$
	$0 \leq 1 < y < x,$
	$0 < 1 = y < x,$

a	$a a a b a b a a a a a$
p	$p q r r q r q s s s s q$
x	$5 5 5 5 5 5 5 5 5 5$
y	$0 1 2 4 0 3 0 .1 .2 .4$

bisimulation \rightarrow runs of the region automaton correspond to runs of \mathcal{A}

Dense case

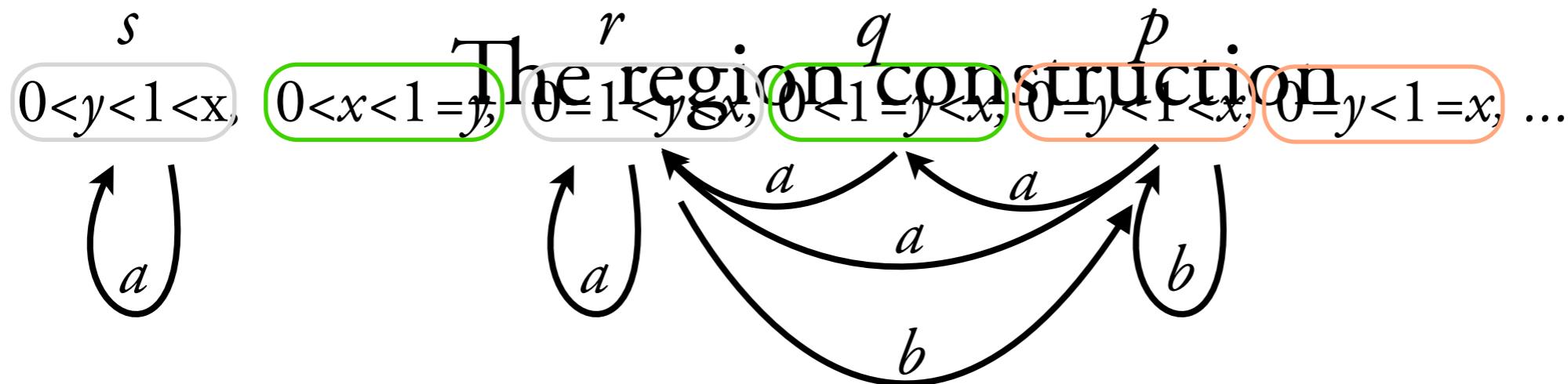
- $D = \mathbb{Q}$
- finite words
- no database

$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$



a	a	a	b	a	b	a	a	a	a
p	q	r	r	q	r	q	s	s	s
x	5	5	5	5	5	5	5	5	5
y	0	1	2	4	0	3	0	.1	.2

bisimulation \rightarrow runs of the region automaton correspond to runs of \mathcal{A}

Dense case

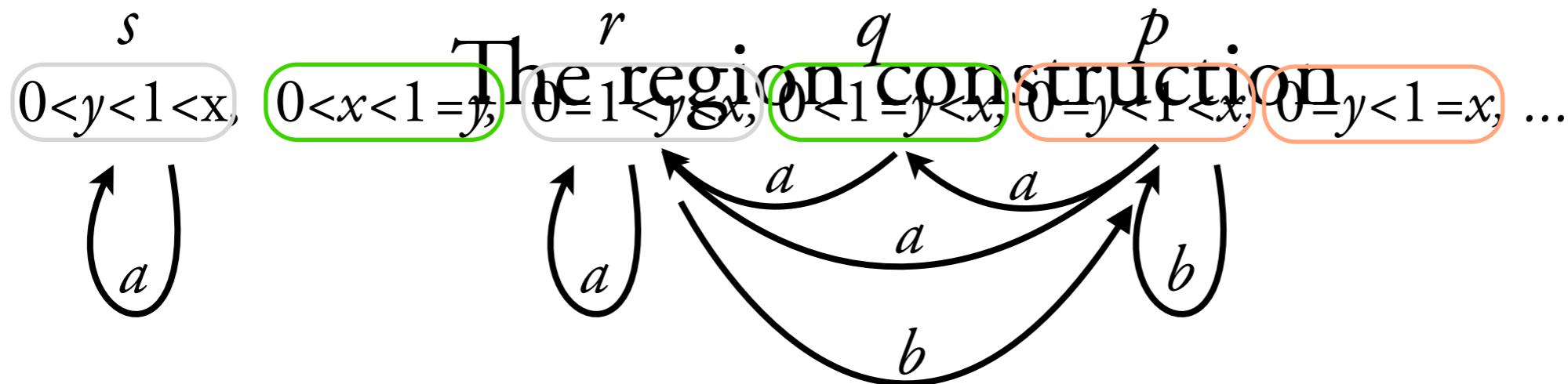
- $D = \mathbb{Q}$
- finite words
- no database

$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$



s	$0 < y < 1 < x,$
	$0 < x < 1 = y,$
	$0 \leq 1 < y < x,$
	$0 < 1 = y < x,$
	$0 \leq y < 1 < x,$
	$0 \leq y < 1 = x,$

$a a a b a b a a a a a$
 $p q r r q r q s s s s q$

x	5 5 5 5 5 5 5 5 5 5 5
y	0 1 2 4 0 3 0 .1 .2 .4 .9 1

bisimulation \rightarrow runs of the region automaton correspond to runs of \mathcal{A}

Dense case

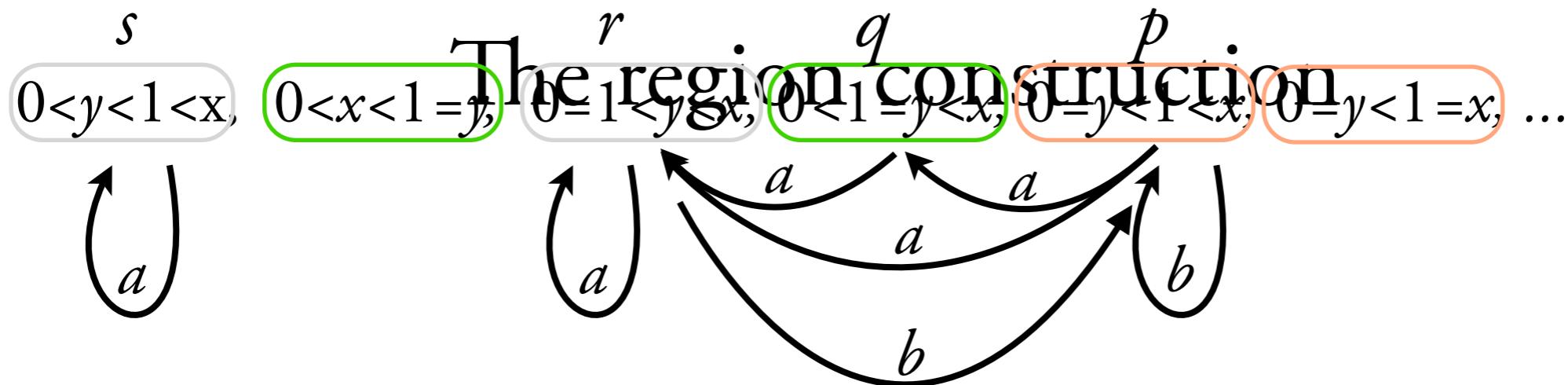
- $D = \mathbb{Q}$
- finite words
- no database

$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$



a	a	a	b	a	b	a	a	a	a
p	q	r	r	q	r	q	s	s	s
x	5	5	5	5	5	5	5	5	5
y	0	1	2	4	0	3	0	.1	.2

bisimulation \rightarrow runs of the region automaton correspond to runs of \mathcal{A}

Dense case

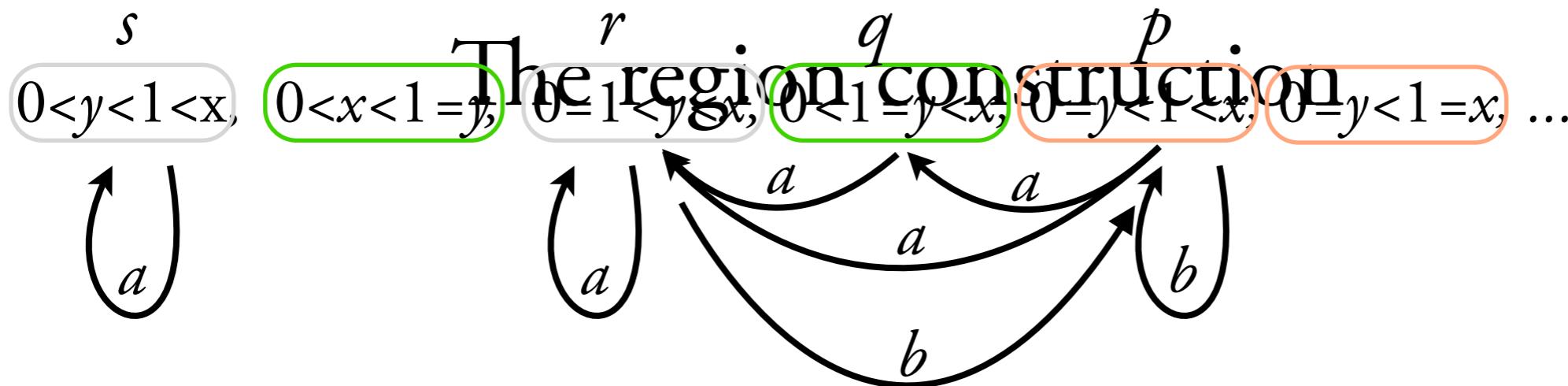
- $D = \mathbb{Q}$
- finite words
- no database

$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$



a	a	a	b	a	b	a	a	a	a
p	q	r	r	q	r	q	s	s	s
x	5	5	5	5	5	5	5	5	5
y	0	1	2	4	0	3	0	.1	.2

bisimulation \rightarrow runs of the region automaton correspond to runs of \mathcal{A}
accepted language: $(a+b)^*a$

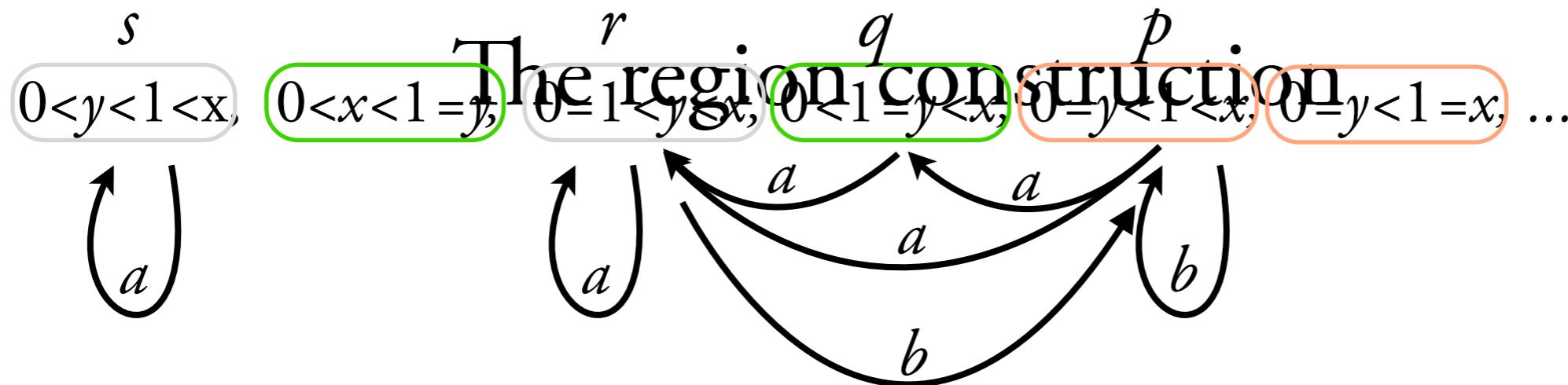
- $D = \mathbb{Q}$
- finite words
- no database

$$\tau_I: (x>0) \wedge (y=0)$$

$$\delta_a: (x'=x) \wedge (y < y' < x)$$

$$\delta_b: (x'=x) \wedge (y'=0)$$

$$\tau_F: (y=1)$$



a	a	a	b	a	b	a	a	a	a
p	q	r	r	q	r	q	s	s	s
x	5	5	5	5	5	5	5	5	5
y	0	1	2	4	0	3	0	.1	.2

bisimulation \rightarrow runs of the region automaton correspond to runs of \mathcal{A}

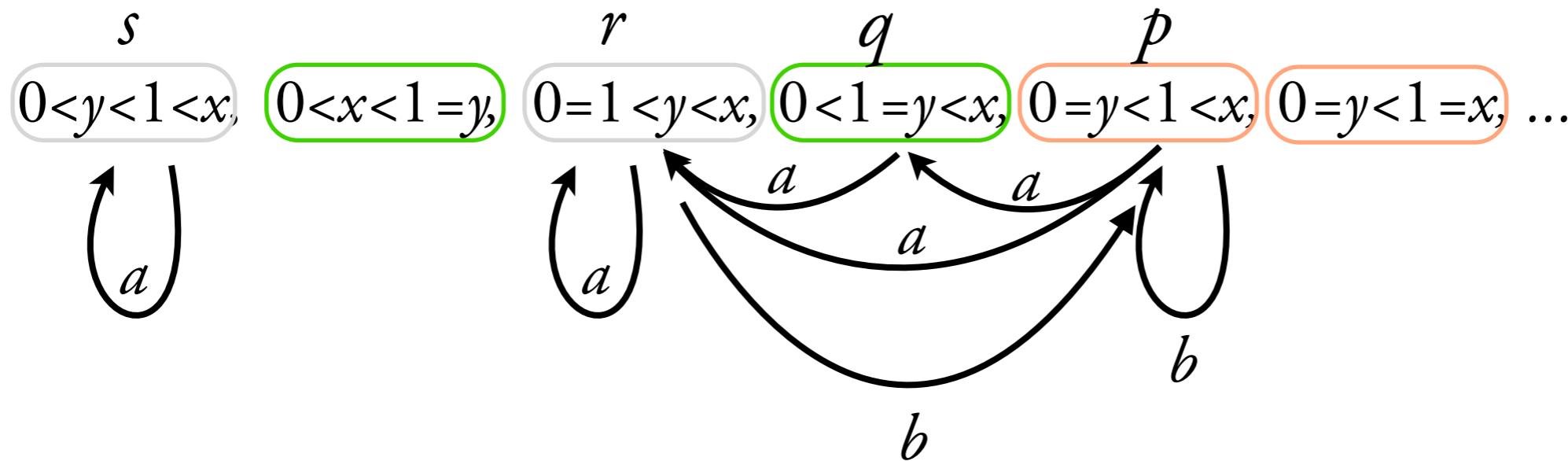
- $D = \mathbb{N}$
- finite words
- no database

$$\tau_I: (x>0) \wedge (y=0)$$

$$\delta_a: (x'=x) \wedge (y < y' < x)$$

$$\delta_b: (x'=x) \wedge (y'=0)$$

$$\tau_F: (y=1)$$



bisimulation \rightarrow runs of the region
automaton correspond to runs of \mathcal{A}

Discrete case

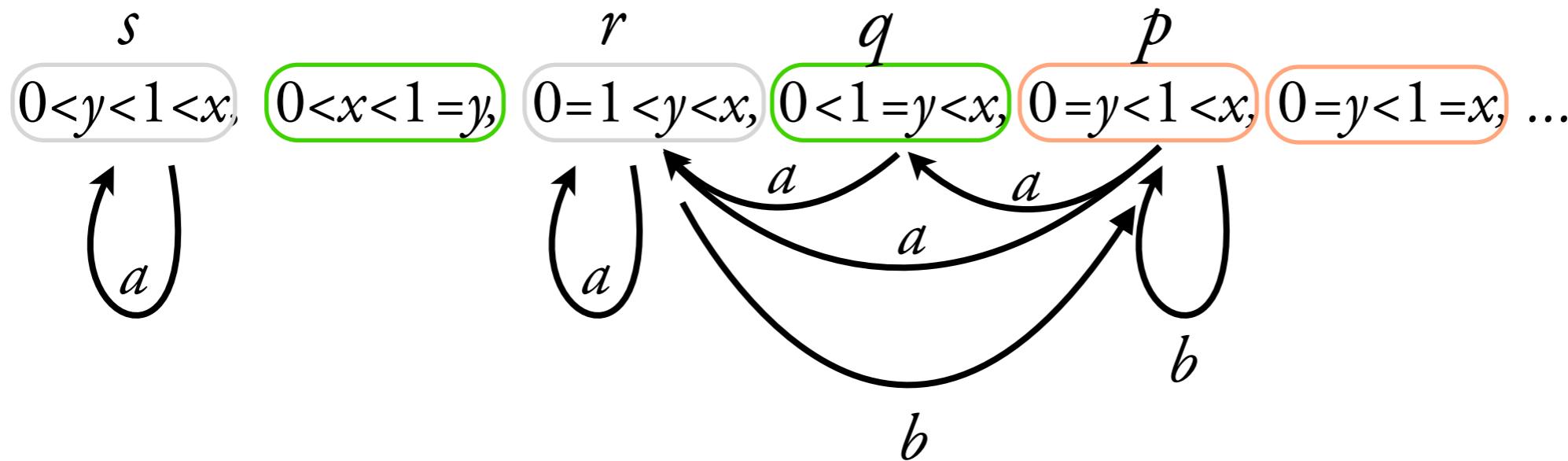
- $D = \mathbb{N}$
- finite words
- no database

$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$



bisimulation \rightarrow runs of the region
automaton correspond to runs of \mathcal{A}

Discrete case

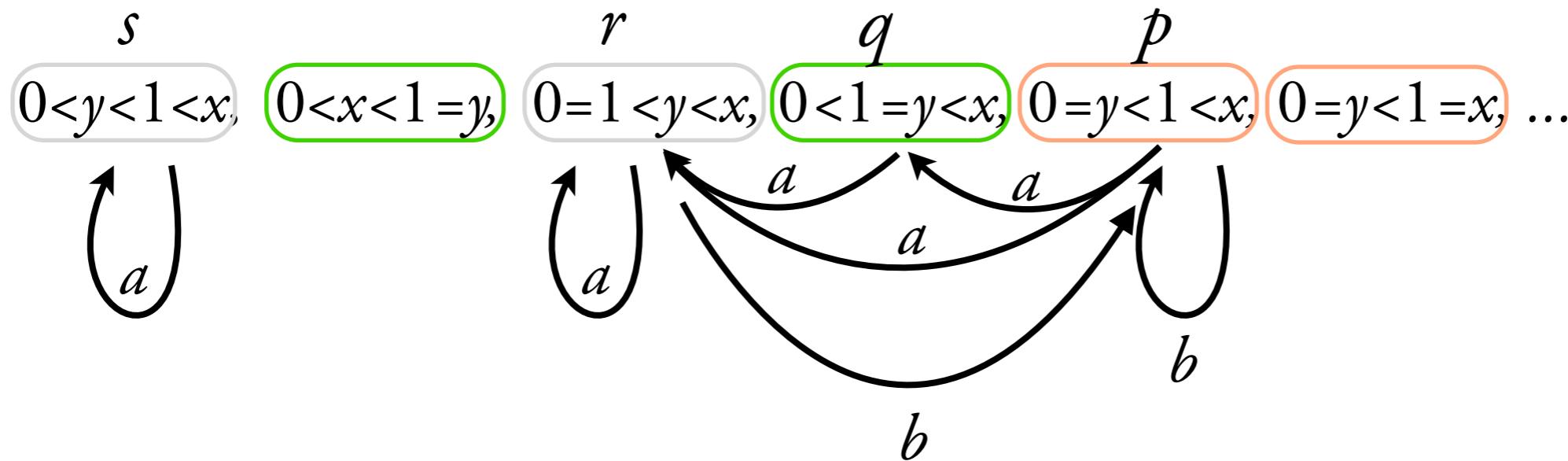
- $D = \mathbb{N}$
- finite words
- no database

$$\tau_I: (x>0) \wedge (y=0)$$

$$\delta_a: (x'=x) \wedge (y < y' < x)$$

$$\delta_b: (x'=x) \wedge (y'=0)$$

$$\tau_F: (y=1)$$



bisimulation \rightarrow runs of the region
 automaton correspond to runs of \mathcal{A}

Discrete case

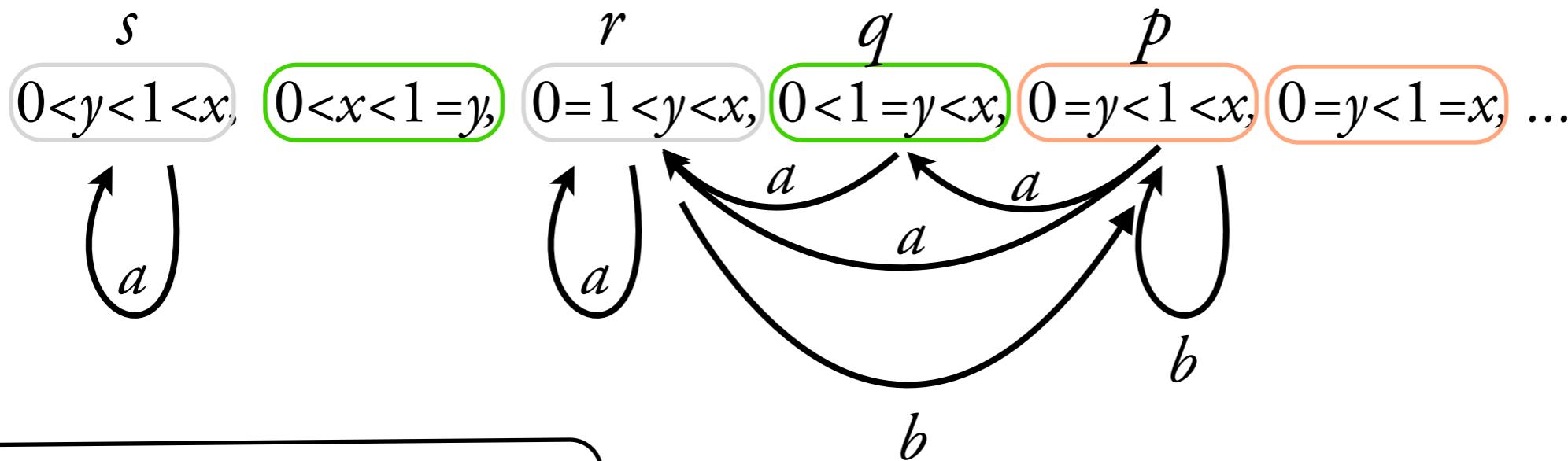
- $D = \mathbb{N}$
- finite words
- no database

$$\tau_I: (x>0) \wedge (y=0)$$

$$\delta_a: (x'=x) \wedge (y < y' < x)$$

$$\delta_b: (x'=x) \wedge (y'=0)$$

$$\tau_F: (y=1)$$



$a a a b a b a a a a a$
 $p s r r q r p s r r r q$

~~bisimulation \rightarrow runs of the region
automaton correspond to runs of \mathcal{A}~~

Discrete case

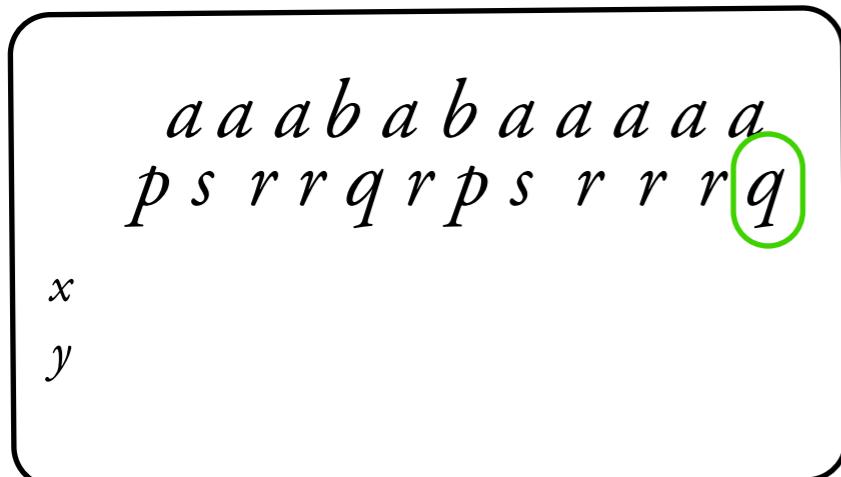
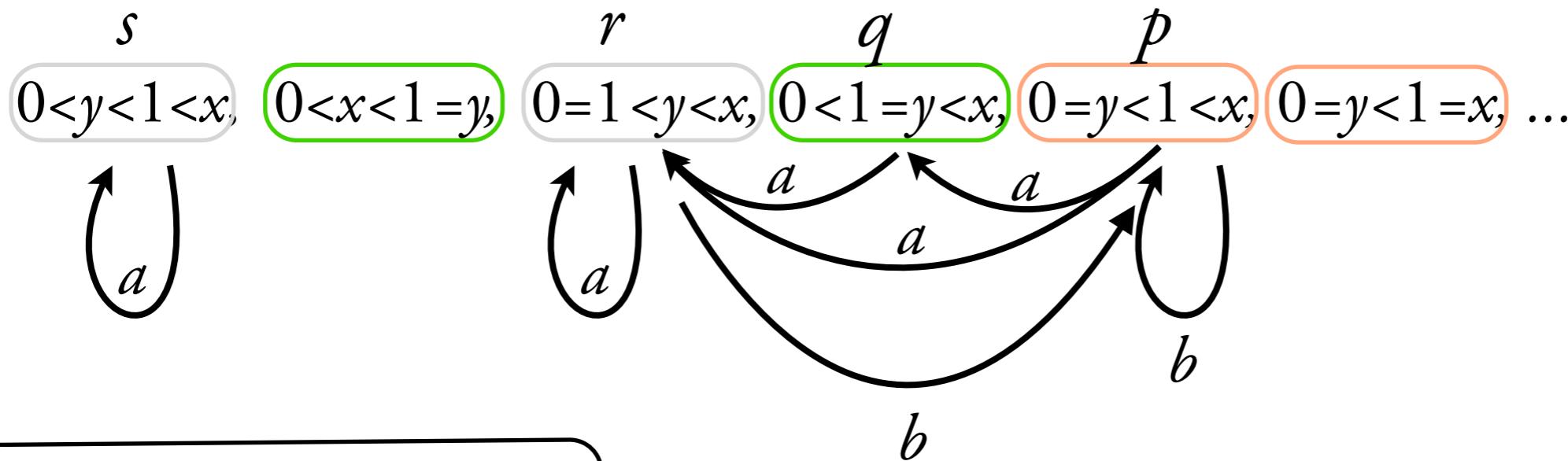
- $D = \mathbb{N}$
- finite words
- no database

$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$



~~bisimulation \rightarrow runs of the region automaton correspond to runs of \mathcal{A}~~

Discrete case

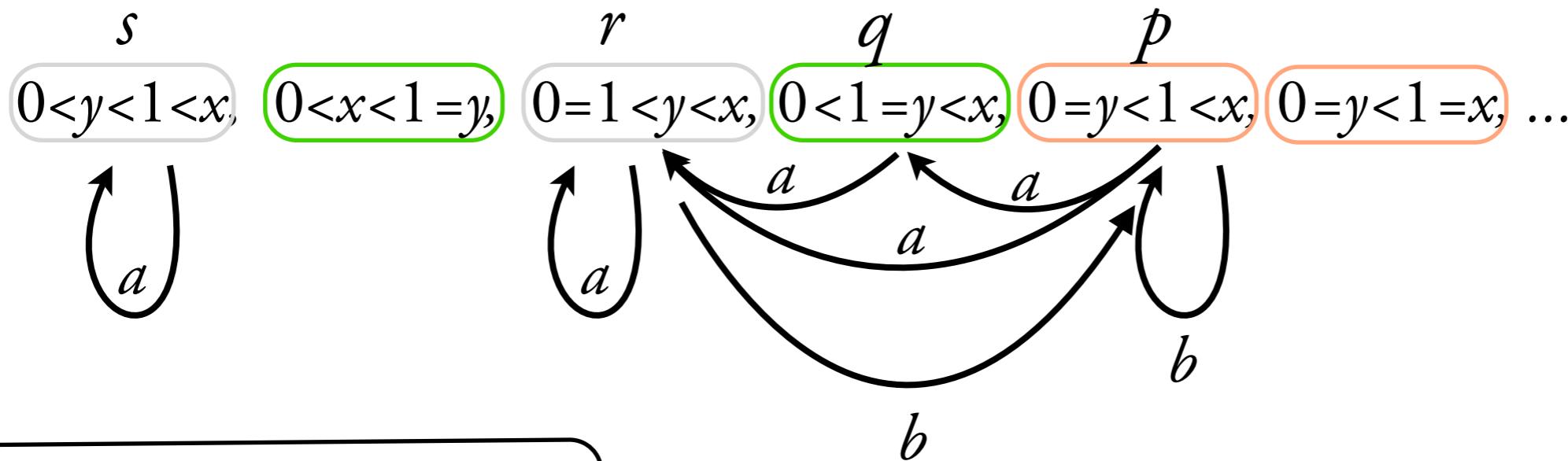
- $D = \mathbb{N}$
- finite words
- no database

$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$



s	$0 < y < 1 < x$
r	$0 < x < 1 = y$
q	$0 = 1 < y < x$
p	$0 < 1 = y < x$

x 5 5 5 5 5 5 5 5
 y 0 1 2 4 0 4 0 1

~~bisimulation \rightarrow runs of the region automaton correspond to runs of \mathcal{A}~~

Discrete case

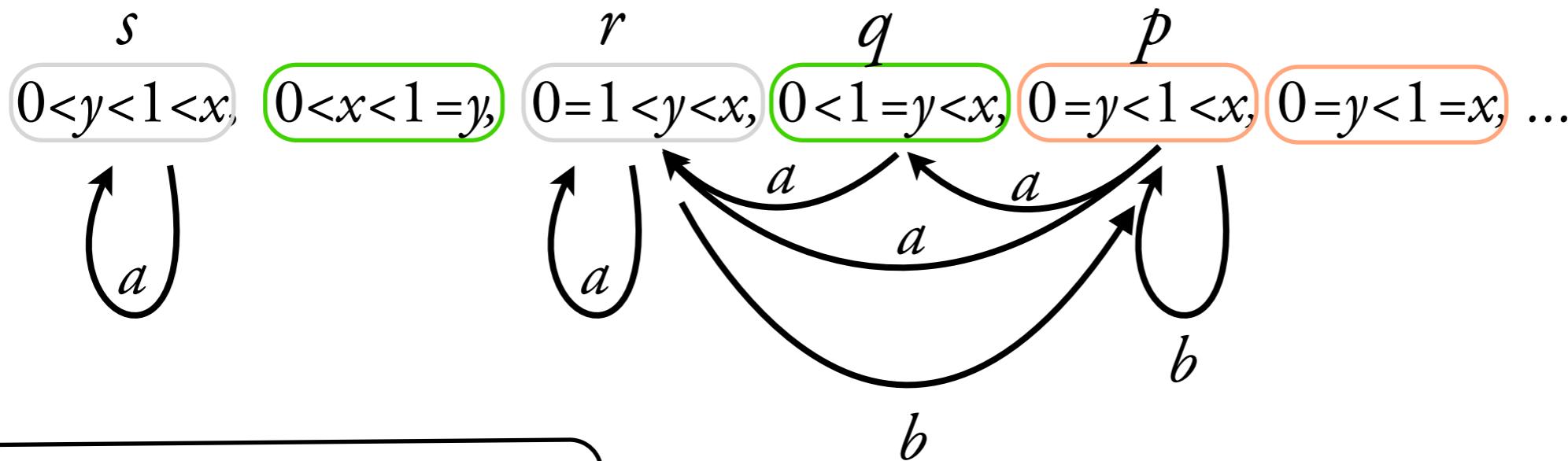
- $D = \mathbb{N}$
- finite words
- no database

$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$



s	$0 < y < 1 < x$
r	$0 < x < 1 = y$
q	$0 = 1 < y < x$
p	$0 < 1 = y < x$

x 5 5 5 5 5 5 5 5 5
 y 0 1 2 4 0 4 0 1 4

~~bisimulation \rightarrow runs of the region automaton correspond to runs of \mathcal{A}~~

Discrete case

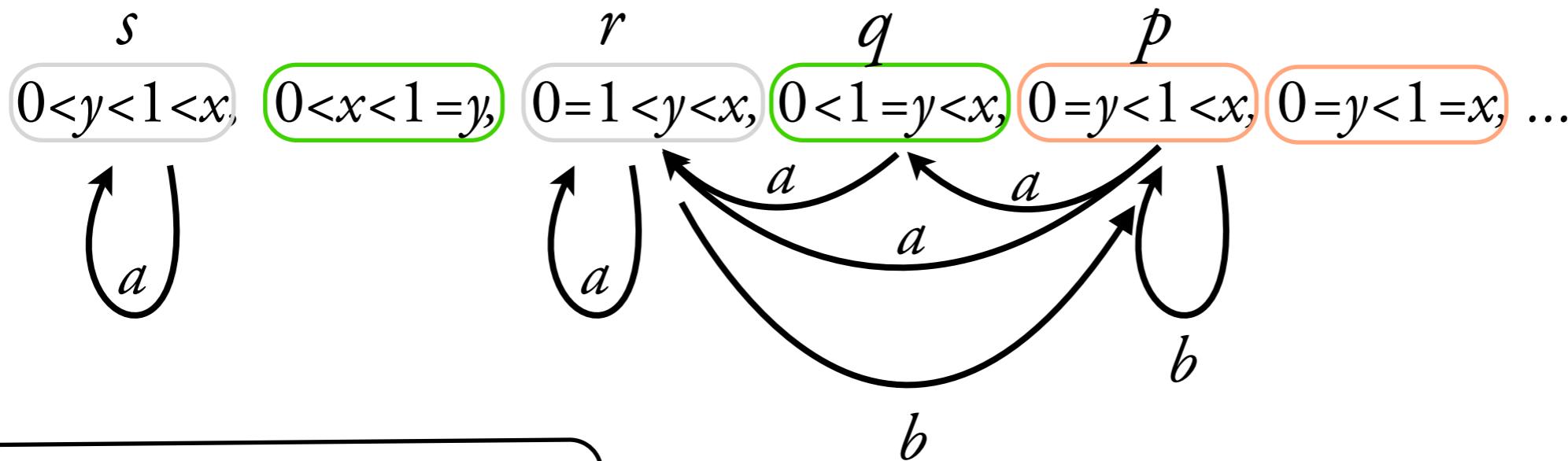
- $D = \mathbb{N}$
- finite words
- no database

$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$



a	a	a	b	a	b	a	a	a	a
p	s	r	r	q	r	r	r	r	q
x	5	5	5	5	5	5	5	5	?
y	0	1	2	4	0	4	0	1	4

bisimulation \rightarrow runs of the region
automaton correspond to runs of \mathcal{A}

Discrete case

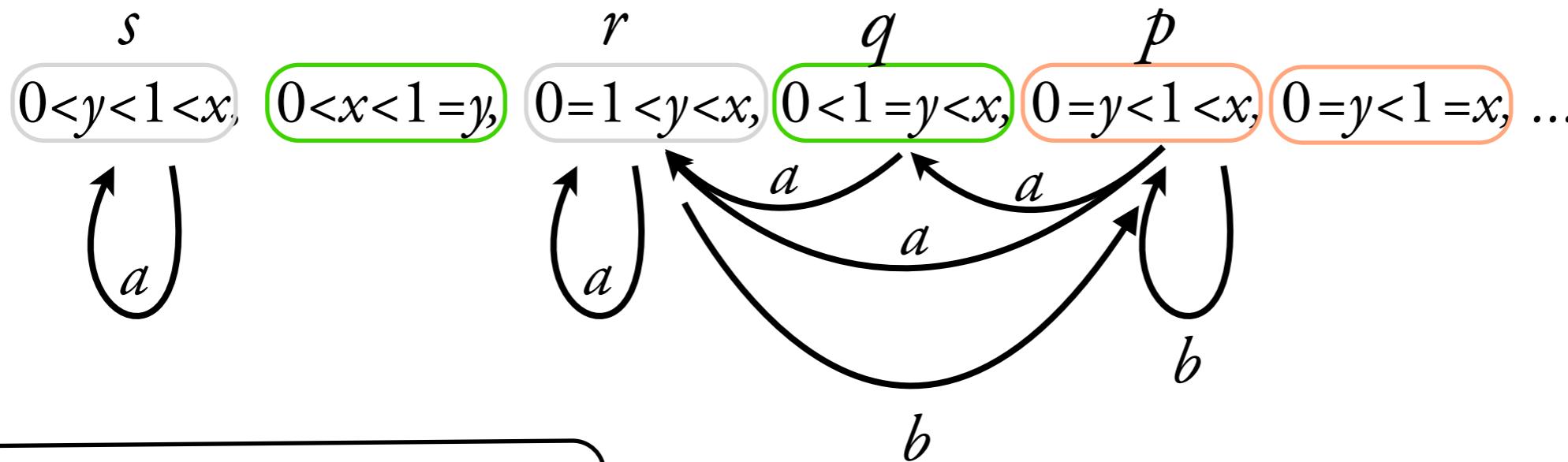
- $D = \mathbb{N}$
- finite words
- no database

$$\tau_I: (x > 0) \wedge (y = 0)$$

$$\delta_a: (x' = x) \wedge (y < y' < x)$$

$$\delta_b: (x' = x) \wedge (y' = 0)$$

$$\tau_F: (y = 1)$$



a	a	a	b	a	b	a	a	a	a
p	s	r	r	q	r	p	s	r	r
x	5	5	5	5	5	5	5	5	?
y	0	1	2	4	0	4	0	1	4

but: in any cell and any n we can find configurations which are *n-bisimilar*
 \rightarrow *finite* runs of the region automaton correspond to runs of \mathcal{A}

An elaboration of these ideas yields the results for infinite runs and database constraints and unary predicates in \mathcal{D} .

An elaboration of these ideas yields the results for infinite runs and database constraints and unary predicates in \mathcal{D} .

Theorem. For all \mathcal{D} , \mathcal{D} -automata over finite words accept regular languages.

An elaboration of these ideas yields the results for infinite runs and database constraints and unary predicates in \mathcal{D} .

Theorem. For all \mathcal{D} , \mathcal{D} -automata over finite words accept regular languages.

Theorem. \mathbb{N} -automata over infinite words accept B-regular languages.

An elaboration of these ideas yields the results for infinite runs and database constraints and unary predicates in \mathcal{D} .

Theorem. For all \mathcal{D} , \mathcal{D} -automata over finite words accept regular languages.

Theorem. \mathbb{N} -automata over infinite words accept B-regular languages.

Theorem. For any reasonable structure \mathcal{D} , emptiness of \mathcal{D} -automata is decidable and in PSPACE.

An elaboration of these ideas yields the results for infinite runs and database constraints and unary predicates in \mathcal{D} .

Theorem. For all \mathcal{D} , \mathcal{D} -automata over finite words accept regular languages.

Theorem. \mathbb{N} -automata over infinite words accept B-regular languages.

Theorem. For any reasonable structure \mathcal{D} , emptiness of \mathcal{D} -automata is decidable and in PSPACE.

Corollary. Deciding LTL+data tests properties of \mathcal{D} -automata is PSPACE-complete.

Thank you!

Thank you!



Thank you!