Counter Machines & Distributed Automata

A Story about Exchanging Space and Time

Olivier Carton[1]   Bruno Guillon[2]   Fabian Reiter[3]

[1] IRIF, Paris Diderot University

[2] University of Milan

[3] LSV, University of Paris-Saclay

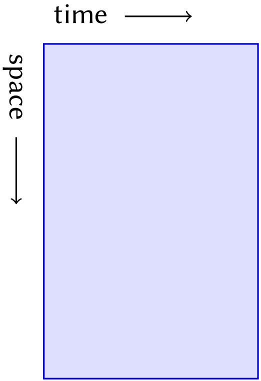
AUTOMATA'18, Ghent, 20 June 2018

A space-time duality

Image credit: NASA/ESA/CXC/SSC/STScI

A space-time duality

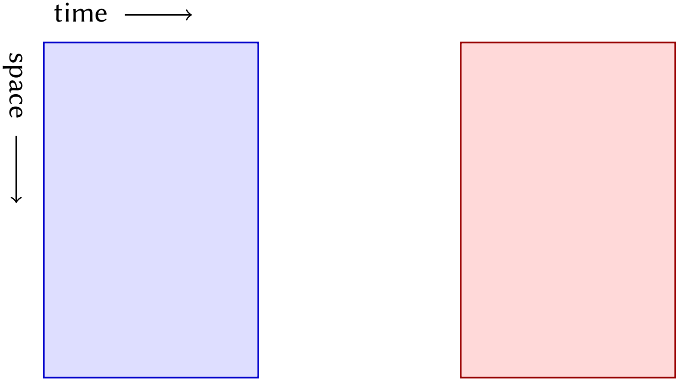
A space-time duality



time ⟶

space ⟶

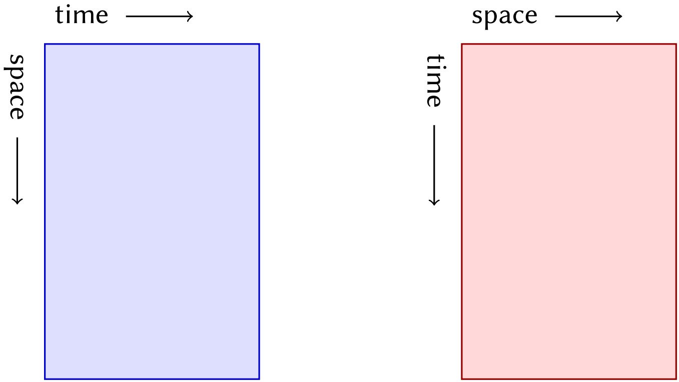
A space-time duality



time ⟶

space

A space-time duality



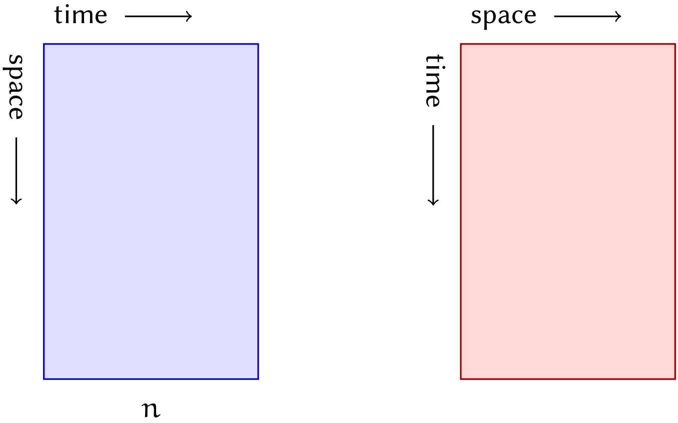
time ⟶

space ⟶

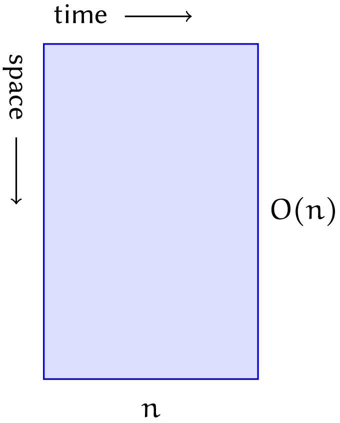
space ⟶

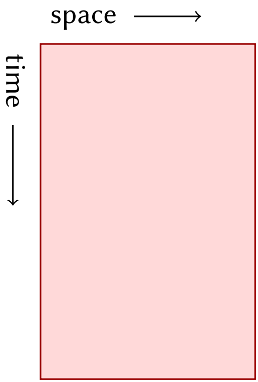
time ⟶

A space-time duality

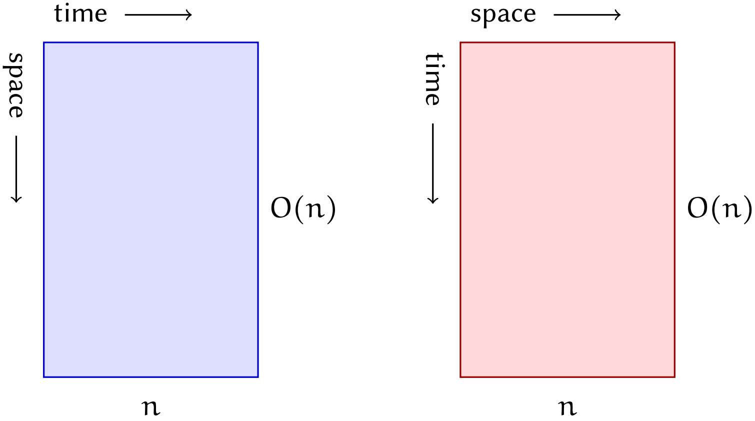


A space-time duality

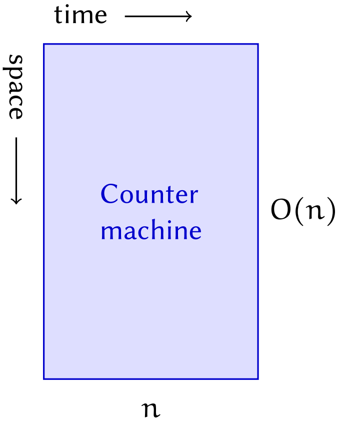


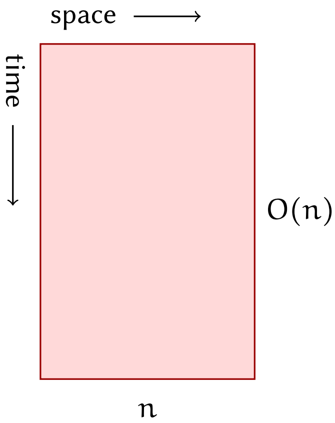


A space-time duality

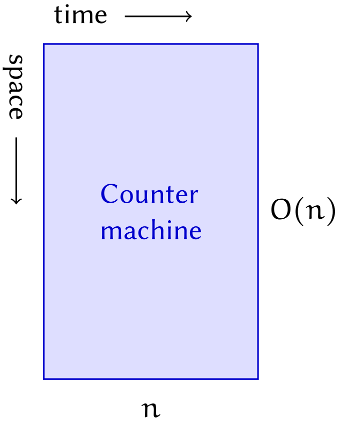


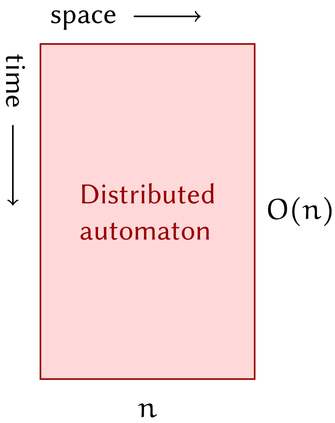
A space-time duality





A space-time duality





Counter machines

Counter machines

a	b	a	c	a	c

Counter machines

a	b	a	c	a	c

$w \in \{a, b, c\}^+$ s.t.

$\forall$ prefix u of w

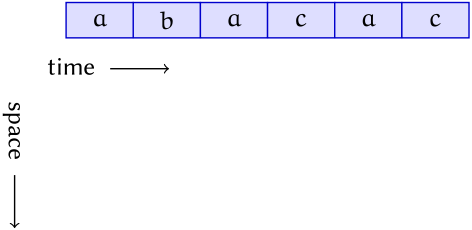
$|u|_a \geq |u|_b \geq |u|_c$

Counter machines

$w \in \{a, b, c\}^+$ s.t.

$\forall$ prefix u of w

$|u|_a \geq |u|_b \geq |u|_c$



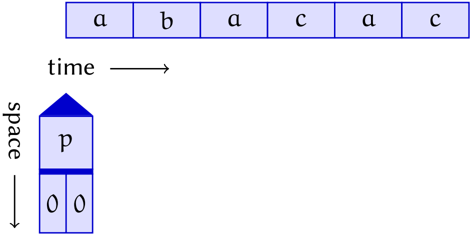
a	b	a	c	a	c

Counter machines

$w \in \{a, b, c\}^{+}$ s.t.

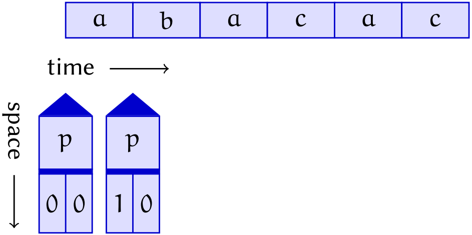
$\forall$ prefix u of w

$|u|_a \geq |u|_b \geq |u|_c$



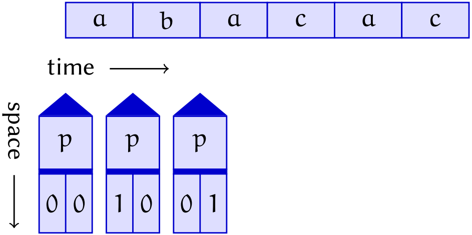
Counter machines

$w \in \{a, b, c\}^+$ s.t.
$\forall$ prefix u of w
$|u|_a \geq |u|_b \geq |u|_c$



Counter machines

$w \in \{a, b, c\}^+$ s.t.
$\forall$ prefix u of w
$|u|_a \geq |u|_b \geq |u|_c$

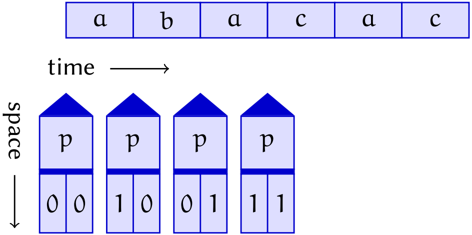


Counter machines

$w \in \{a, b, c\}^+$ s.t.

$\forall$ prefix u of w

$|u|_a \geq |u|_b \geq |u|_c$

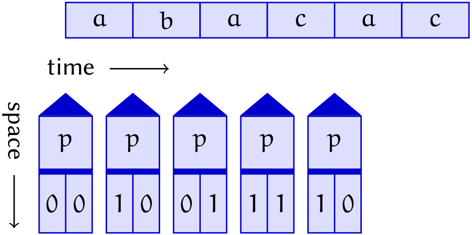


Counter machines

$w \in \{a, b, c\}^+$ s.t.

$\forall$ prefix u of w

$|u|_a \geq |u|_b \geq |u|_c$



a b a c a c

time ⟶

space ⟶

p 0 0

p 1 0

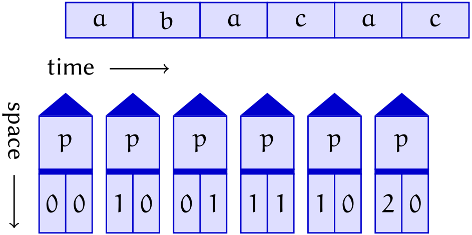
p 0 1

p 1 1

p 1 0

Counter machines

$w \in \{a, b, c\}^+$ s.t.
$\forall$ prefix u of w
$|u|_a \geq |u|_b \geq |u|_c$

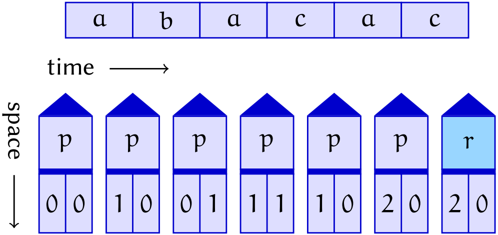


Counter machines

$w \in \{a, b, c\}^+$ s.t.

$\forall$ prefix u of w

$|u|_a \geq |u|_b \geq |u|_c$

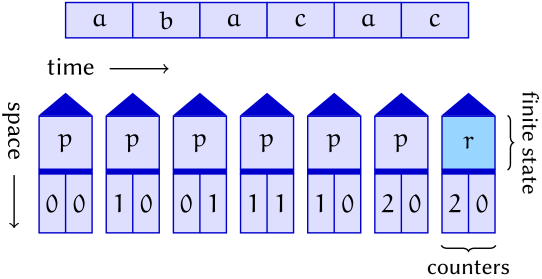


Counter machines

$w \in \{a, b, c\}^+$ s.t.

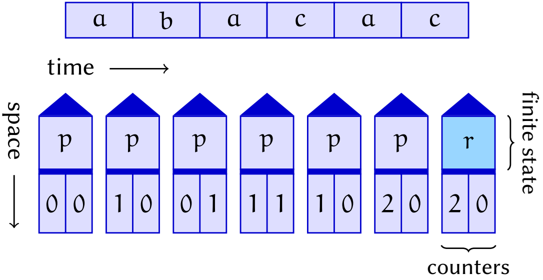
$\forall$ prefix u of w

$|u|_a \geq |u|_b \geq |u|_c$



Counter machines

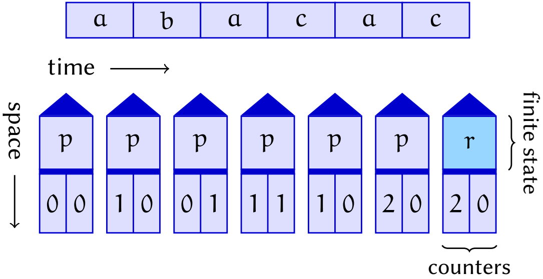
$w \in \{a, b, c\}^+$ s.t.
$\forall$ prefix u of w
$|u|_a \geq |u|_b \geq |u|_c$



- One-way, deterministic, no ε-transitions.

Counter machines

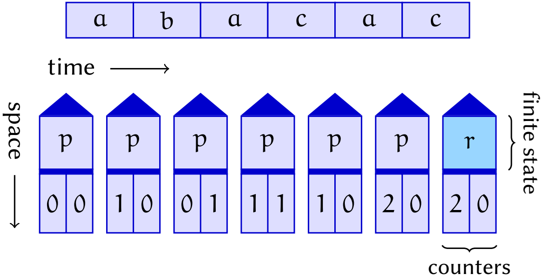
$w \in \{a, b, c\}^+$ s.t.
$\forall$ prefix u of w
$|u|_a \geq |u|_b \geq |u|_c$



- One-way, deterministic, no ε-transitions.
- Multiple counters $x, y, \ldots$ with values in $\mathbb{Z}$.

Counter machines

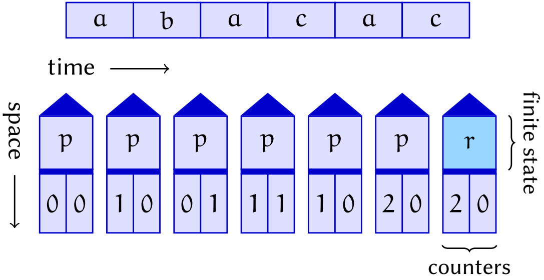
$w \in \{a, b, c\}^+$ s.t.
$\forall$ prefix u of w
$|u|_a \geq |u|_b \geq |u|_c$



- One-way, deterministic, no ε-transitions.
- Multiple counters $x, y, \ldots$ with values in $\mathbb{Z}$.
- Counter values known up to some threshold.

Counter machines

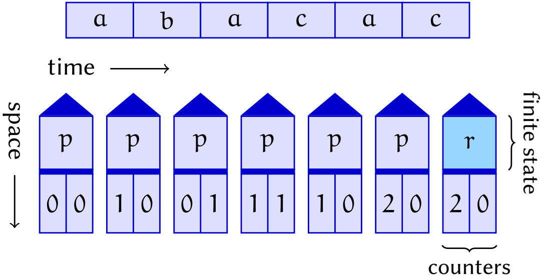
$w \in \{a, b, c\}^+$ s.t.
$\forall$ prefix u of w
$|u|_a \geq |u|_b \geq |u|_c$



- One-way, deterministic, no ε-transitions.
- Multiple counters $x, y, \ldots$ with values in $\mathbb{Z}$.
- Counter values known up to some threshold.
- Counter updates of the form $x \coloneqq y_1 + \cdots + y_n + c$.

Counter machines

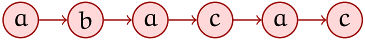
$w \in \{a, b, c\}^+$ s.t.
$\forall$ prefix u of w
$|u|_a \geq |u|_b \geq |u|_c$



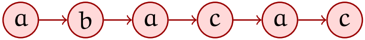
- One-way, deterministic, no ε-transitions.
- Multiple counters $x, y, \ldots$ with values in $\mathbb{Z}$.
- Counter values known up to some threshold.
- Counter updates of the form $\underbrace{x \coloneqq y_1 + \cdots + y_n + c}_{\text{allows copy and sum}}$.

Distributed automata

Distributed automata

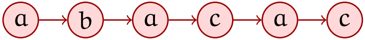


Distributed automata



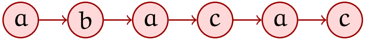
- Finite-state machine on each node.

Distributed automata



- Finite-state machine on each node.
- Synchronous transitions of the form $\delta\colon \underbrace{Q}_{\text{left}} \times \underbrace{Q}_{\text{self}} \to Q$.

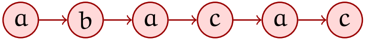
Distributed automata



- Finite-state machine on each node.
- Synchronous transitions of the form $\delta \colon \underbrace{Q}_{\text{left}} \times \underbrace{Q}_{\text{self}} \to Q$.
- Word accepted if last node visits an accepting state.

Distributed automata

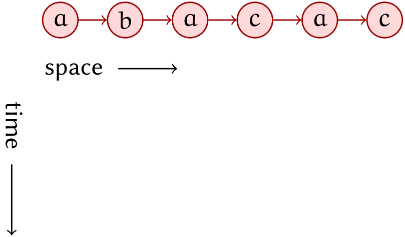
$w \in \{a, b, c\}^+$ s.t.
$\forall$ prefix u of w
$|u|_a \geq |u|_b \geq |u|_c$



- Finite-state machine on each node.
- Synchronous transitions of the form $\delta\colon \underbrace{Q}_{\text{left}} \times \underbrace{Q}_{\text{self}} \to Q$.
- Word accepted if last node visits an accepting state.

Distributed automata

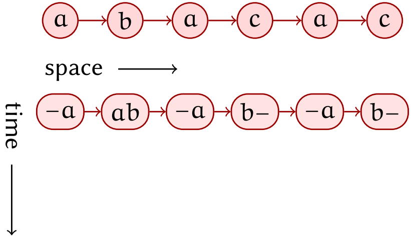
$w \in \{a, b, c\}^+$ s.t.
$\forall$ prefix u of w
$|u|_a \geq |u|_b \geq |u|_c$



- Finite-state machine on each node.
- Synchronous transitions of the form $\delta\colon \underbrace{Q}_{\text{left}} \times \underbrace{Q}_{\text{self}} \to Q$.
- Word accepted if last node visits an accepting state.

Distributed automata

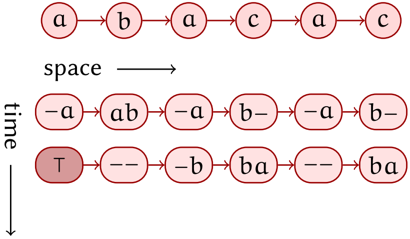
$w \in \{a, b, c\}^+$ s.t.
$\forall$ prefix u of w
$|u|_a \geq |u|_b \geq |u|_c$



- Finite-state machine on each node.
- Synchronous transitions of the form $\delta\colon \underbrace{Q}_{\text{left}} \times \underbrace{Q}_{\text{self}} \to Q$.
- Word accepted if last node visits an accepting state.

Distributed automata

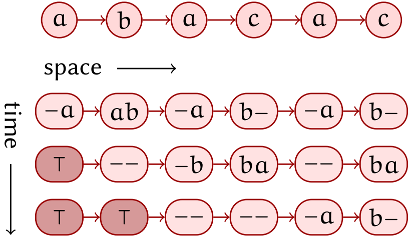
$w \in \{a, b, c\}^+$ s.t.
$\forall$ prefix u of w
$|u|_a \geq |u|_b \geq |u|_c$



- Finite-state machine on each node.
- Synchronous transitions of the form $\delta\colon \underbrace{Q}_{\text{left}} \times \underbrace{Q}_{\text{self}} \to Q$.
- Word accepted if last node visits an accepting state.

Distributed automata

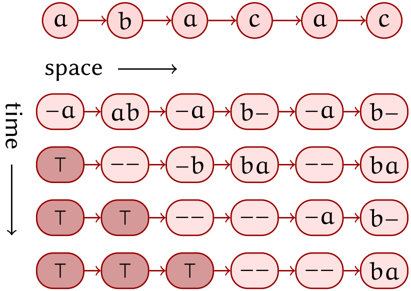
$w \in \{a, b, c\}^+$ s.t.
$\forall$ prefix u of w
$|u|_a \geq |u|_b \geq |u|_c$



- Finite-state machine on each node.
- Synchronous transitions of the form $\delta\colon \underbrace{Q}_{\text{left}} \times \underbrace{Q}_{\text{self}} \to Q$.
- Word accepted if last node visits an accepting state.

Distributed automata

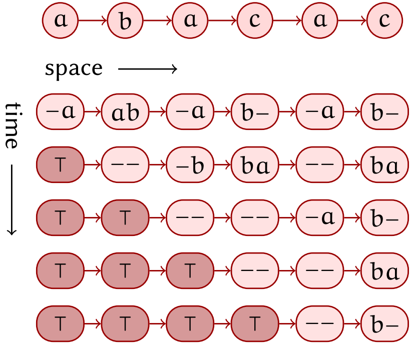
$w \in \{a, b, c\}^+$ s.t.
$\forall$ prefix u of w
$|u|_a \geq |u|_b \geq |u|_c$



- Finite-state machine on each node.
- Synchronous transitions of the form $\delta\colon \underbrace{Q}_{\text{left}} \times \underbrace{Q}_{\text{self}} \to Q$.
- Word accepted if last node visits an accepting state.

Distributed automata

$w \in \{a, b, c\}^+$ s.t.
$\forall$ prefix u of w
$|u|_a \geq |u|_b \geq |u|_c$

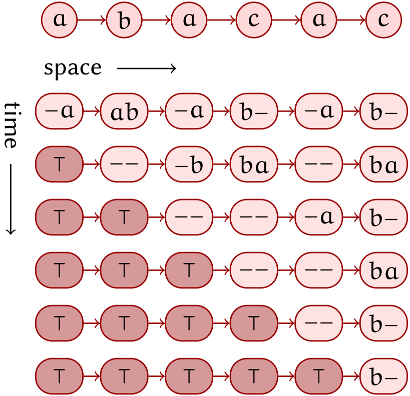


- Finite-state machine on each node.
- Synchronous transitions of the form $\delta\colon \underbrace{Q}_{\text{left}} \times \underbrace{Q}_{\text{self}} \to Q$.
- Word accepted if last node visits an accepting state.

Distributed automata

$w \in \{a, b, c\}^+$ s.t.
$\forall$ prefix u of w
$|u|_a \geq |u|_b \geq |u|_c$

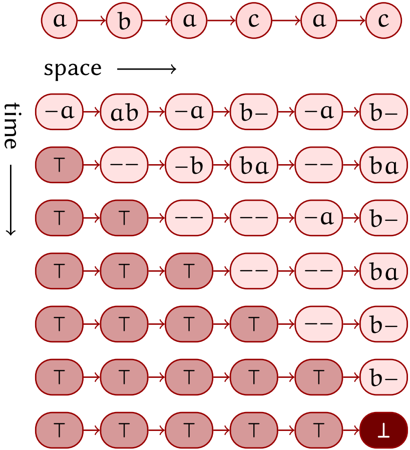
- Finite-state machine on each node.
- Synchronous transitions of the form $\delta\colon \underbrace{Q}_{\text{left}} \times \underbrace{Q}_{\text{self}} \to Q$.
- Word accepted if last node visits an accepting state.



Distributed automata

$w \in \{a, b, c\}^+$ s.t.
$\forall$ prefix u of w
$|u|_a \geq |u|_b \geq |u|_c$

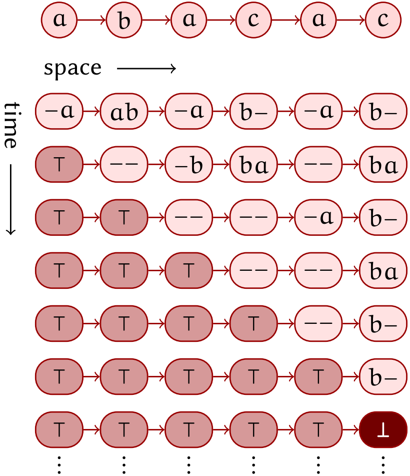
- Finite-state machine on each node.
- Synchronous transitions of the form $\delta\colon \underbrace{Q}_{\text{left}} \times \underbrace{Q}_{\text{self}} \to Q$.
- Word accepted if last node visits an accepting state.



Distributed automata

$w \in \{a, b, c\}^+$ s.t.
$\forall$ prefix u of w
$|u|_a \geq |u|_b \geq |u|_c$

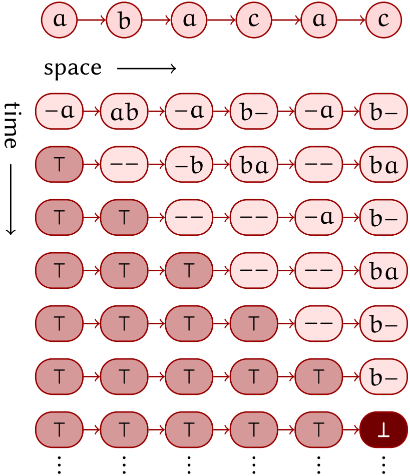
- Finite-state machine on each node.
- Synchronous transitions of the form $\delta\colon \underbrace{Q}_{\text{left}} \times \underbrace{Q}_{\text{self}} \to Q$.
- Word accepted if last node visits an accepting state.



Distributed automata

$w \in \{a, b, c\}^+$ s.t.
$\forall$ prefix u of w
$|u|_a \geq |u|_b \geq |u|_c$

- Finite-state machine on each node.
- Synchronous transitions of the form $\delta\colon \underbrace{Q}_{\text{left}} \times \underbrace{Q}_{\text{self}} \to Q$.
- Word accepted if last node visits an accepting state.



The same as (reversed, one-dimensional) one-way cellular automata.

Our translations

Our translations

Copyless
counter machines

Our translations

Counters can be summed up but not copied.

Copyless counter machines

Our translations

Counters can be summed up but not copied.

Copyless
counter machines

Sumless
counter machines

Our translations

Counters can be summed up but not copied.

Copyless
counter machines

Sumless
counter machines

Counters can be copied but not summed up.

Our translations

Counters can be summed up but not copied.

Copyless
counter machines

Sumless
counter machines

Counters can be copied but not summed up.

LINEAR SPACE

Our translations

Counters can be summed up but not copied.

Copyless
counter machines

Sumless
counter machines

Counters can be copied but not summed up.

LINEAR SPACE

Quasi-acyclic
distributed automata

Our translations

Counters can be summed up but not copied.

Copyless
counter machines

Sumless
counter machines

Counters can be copied but not summed up.

LINEAR SPACE

Quasi-acyclic
distributed automata

The state diagram has no directed cycles except for self-loops.

Our translations

Counters can be summed up but not copied.

Copyless
counter machines

Sumless
counter machines

Counters can be copied but not summed up.

LINEAR SPACE

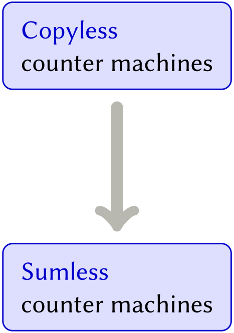
Quasi-acyclic
distributed automata

The state diagram has no directed cycles except for self-loops.

LINEAR TIME

Our translations

Counters can be summed up but not copied.



Copyless counter machines

Sumless counter machines

Counters can be copied but not summed up.

LINEAR SPACE

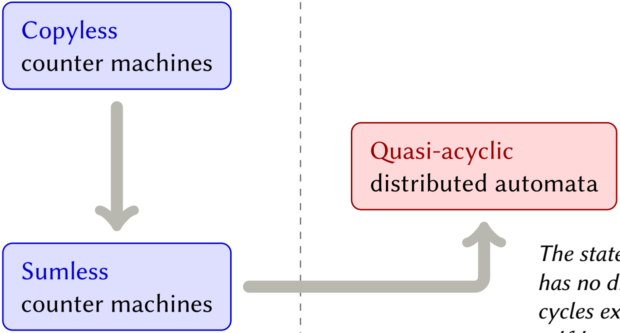
Quasi-acyclic distributed automata

The state diagram has no directed cycles except for self-loops.

LINEAR TIME

Our translations

Counters can be summed up but not copied.



Copyless counter machines

Sumless counter machines

Quasi-acyclic distributed automata

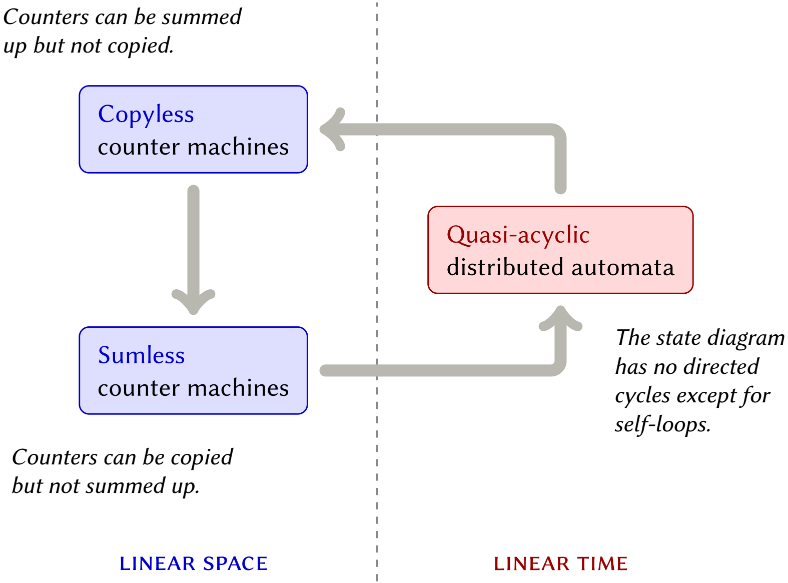
The state diagram has no directed cycles except for self-loops.

Counters can be copied but not summed up.

LINEAR SPACE

LINEAR TIME

Our translations



Counters can be summed up but not copied.

Copyless counter machines

Sumless counter machines

Counters can be copied but not summed up.

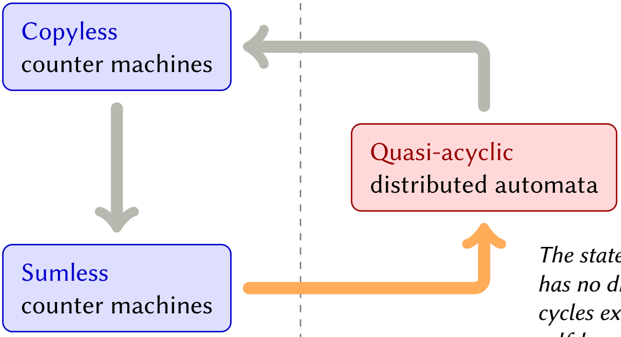
Quasi-acyclic distributed automata

The state diagram has no directed cycles except for self-loops.

LINEAR SPACE

LINEAR TIME

Our translations



Counters can be summed up but not copied.

Copyless counter machines

Sumless counter machines

Counters can be copied but not summed up.

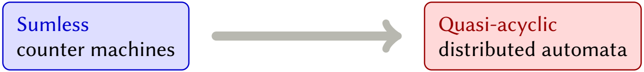
Quasi-acyclic distributed automata

The state diagram has no directed cycles except for self-loops.

LINEAR SPACE

LINEAR TIME

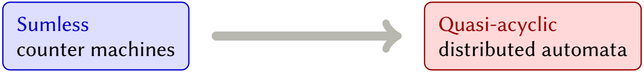
Using time as space



Sumless
counter machines

Quasi-acyclic
distributed automata

Using time as space

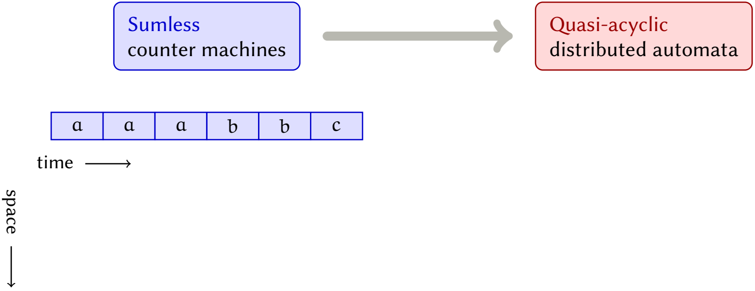


Sumless
counter machines

Quasi-acyclic
distributed automata

a	a	a	b	b	c

Using time as space

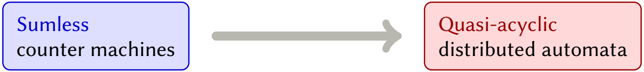


Sumless counter machines

Quasi-acyclic distributed automata

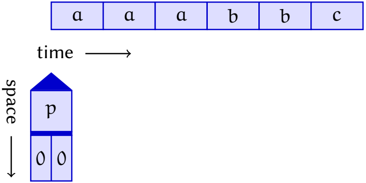
a	a	a	b	b	c

Using time as space



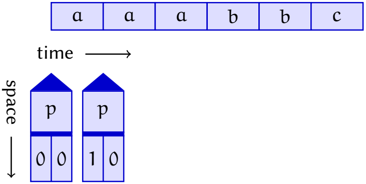
Sumless counter machines

Quasi-acyclic distributed automata

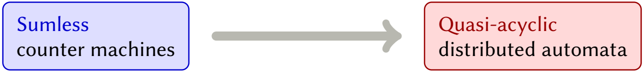


Using time as space

Sumless counter machines → Quasi-acyclic distributed automata

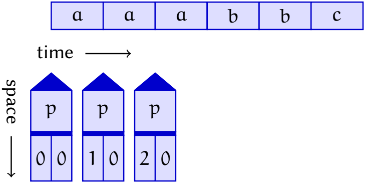


Using time as space

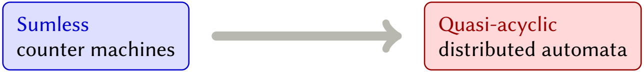


Sumless counter machines

Quasi-acyclic distributed automata

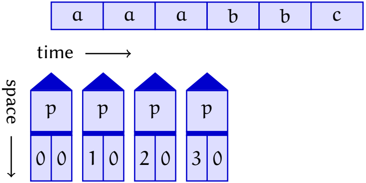


Using time as space



Sumless
counter machines

Quasi-acyclic
distributed automata

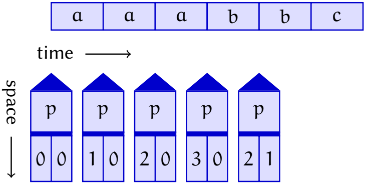


Using time as space

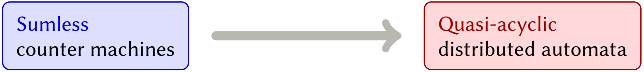
Sumless
counter machines

→

Quasi-acyclic
distributed automata

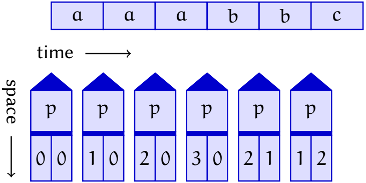


Using time as space

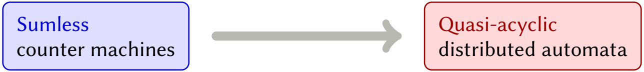


Sumless counter machines

Quasi-acyclic distributed automata

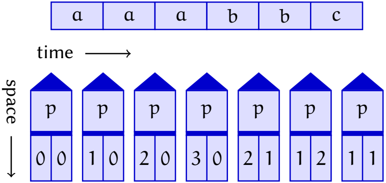


Using time as space



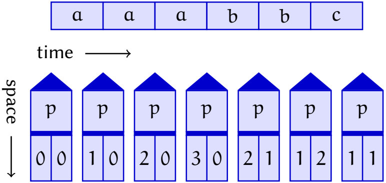
Sumless counter machines

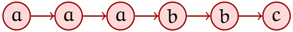
Quasi-acyclic distributed automata



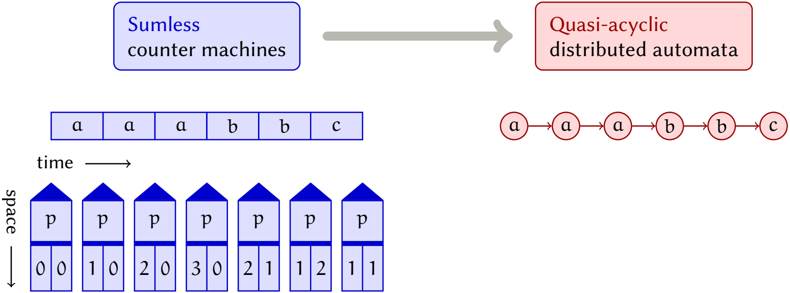
Using time as space

Sumless counter machines → Quasi-acyclic distributed automata





Using time as space



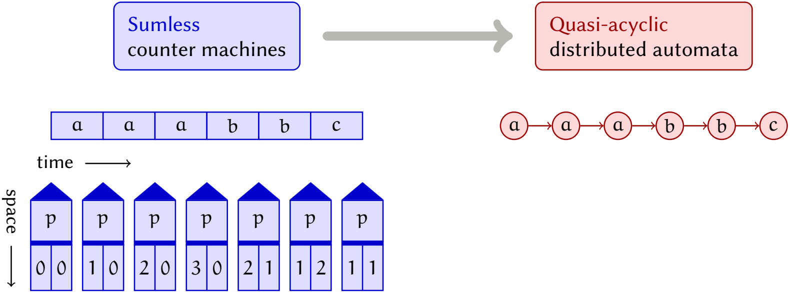
Sumless counter machines

Quasi-acyclic distributed automata

Without loss of generality:

- Nonnegative counter values.

Using time as space



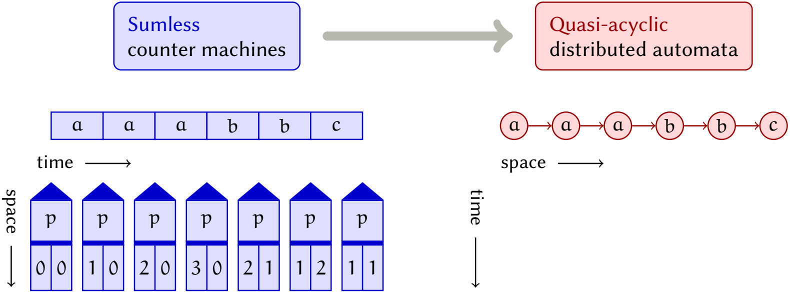
Sumless counter machines

Quasi-acyclic distributed automata

Without loss of generality:

- Nonnegative counter values.
- Counters can only be tested for 0, incremented by $-1, 0, 1$, and copied.

Using time as space



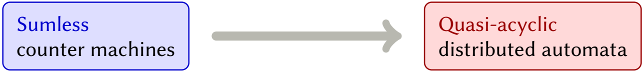
Sumless counter machines

Quasi-acyclic distributed automata

Without loss of generality:

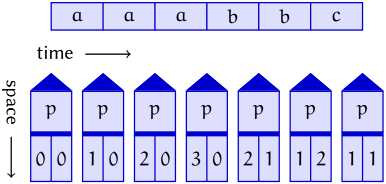
- Nonnegative counter values.
- Counters can only be tested for 0, incremented by $-1, 0, 1$, and copied.

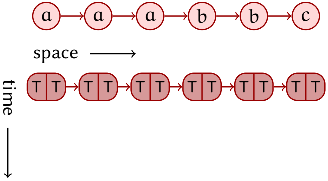
Using time as space



Sumless
counter machines

Quasi-acyclic
distributed automata



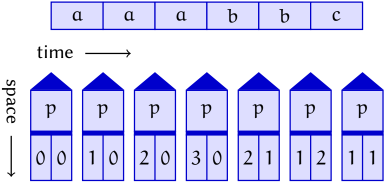


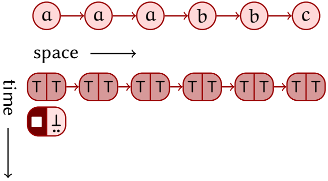
Without loss of generality:

- Nonnegative counter values.
- Counters can only be tested for 0, incremented by $-1, 0, 1,$ and copied.

Using time as space

Sumless counter machines → Quasi-acyclic distributed automata

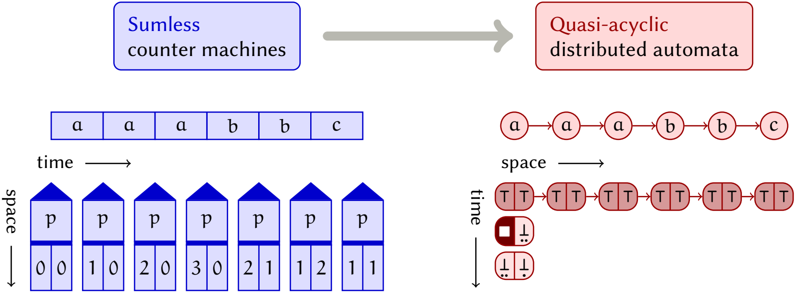




Without loss of generality:

- Nonnegative counter values.
- Counters can only be tested for 0, incremented by $-1, 0, 1,$ and copied.

Using time as space

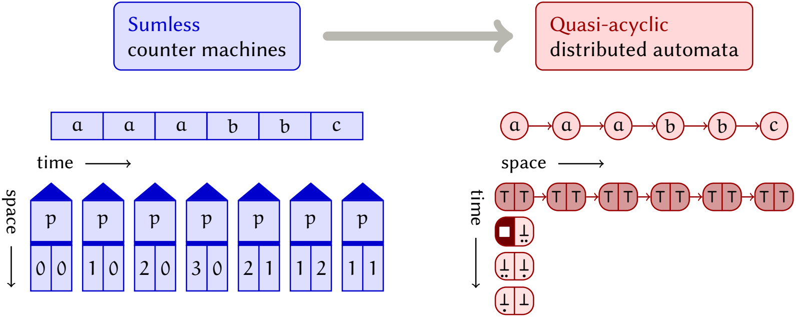


Sumless counter machines → Quasi-acyclic distributed automata

Without loss of generality:

- Nonnegative counter values.
- Counters can only be tested for 0, incremented by $-1, 0, 1$, and copied.

Using time as space

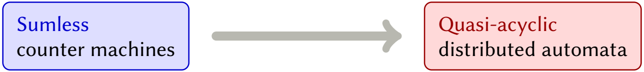


Sumless counter machines → Quasi-acyclic distributed automata

Without loss of generality:

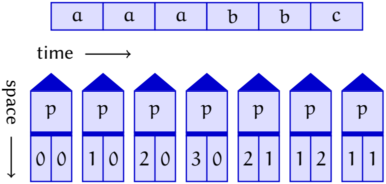
- Nonnegative counter values.
- Counters can only be tested for 0, incremented by $-1, 0, 1$, and copied.

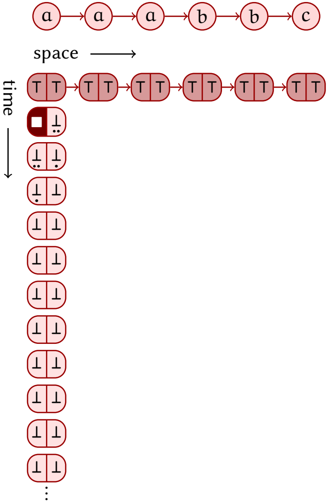
Using time as space



Sumless counter machines

Quasi-acyclic distributed automata

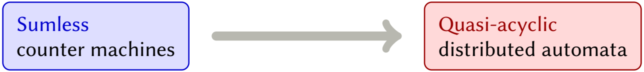




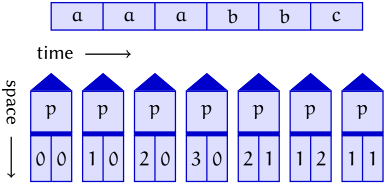
Without loss of generality:

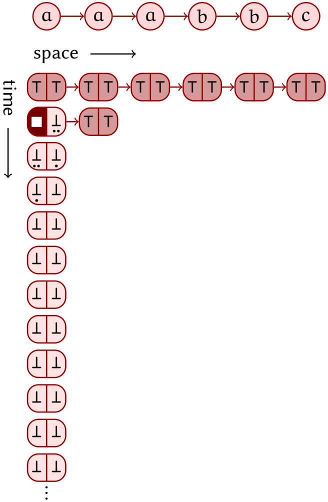
- Nonnegative counter values.
- Counters can only be tested for 0, incremented by $-1, 0, 1,$ and copied.

Using time as space


Sumless counter machines

Quasi-acyclic distributed automata

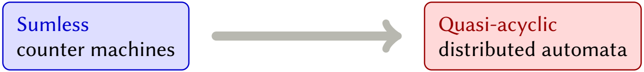




Without loss of generality:

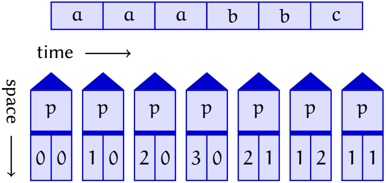
- Nonnegative counter values.
- Counters can only be tested for 0, incremented by $-1, 0, 1,$ and copied.

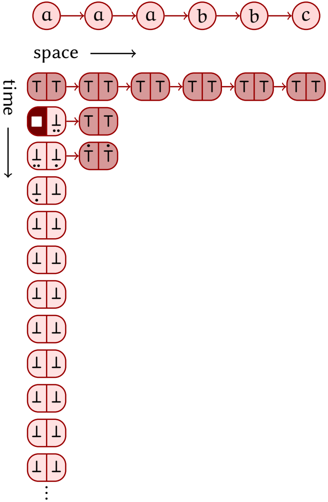
Using time as space



Sumless counter machines

Quasi-acyclic distributed automata

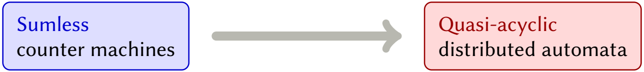




Without loss of generality:

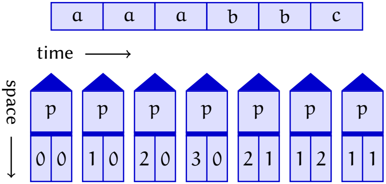
- Nonnegative counter values.
- Counters can only be tested for 0, incremented by $-1, 0, 1$, and copied.

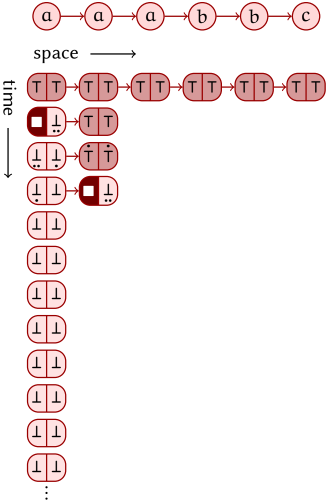
Using time as space



Sumless counter machines

Quasi-acyclic distributed automata





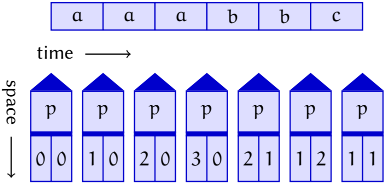
Without loss of generality:

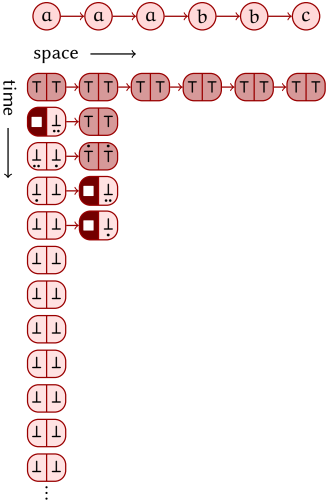
- Nonnegative counter values.
- Counters can only be tested for 0, incremented by $-1, 0, 1$, and copied.

Using time as space

Sumless counter machines

Quasi-acyclic distributed automata

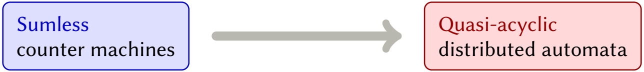


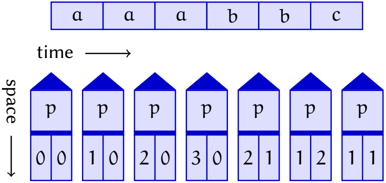


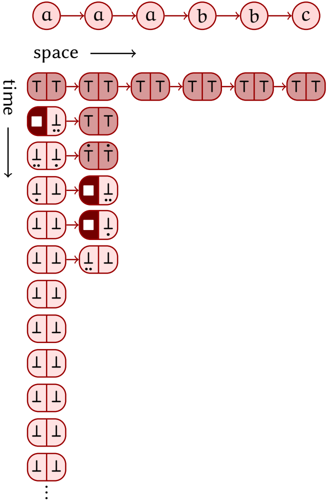
Without loss of generality:

- Nonnegative counter values.
- Counters can only be tested for 0, incremented by $-1, 0, 1$, and copied.

Using time as space



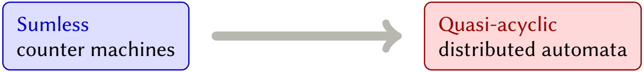




Without loss of generality:

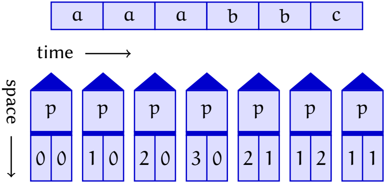
- Nonnegative counter values.
- Counters can only be tested for 0, incremented by $-1, 0, 1$, and copied.

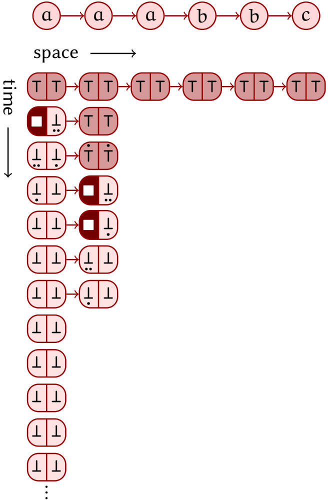
Using time as space



Sumless counter machines

Quasi-acyclic distributed automata

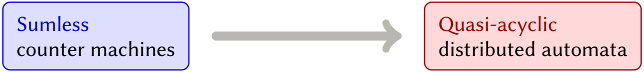




Without loss of generality:

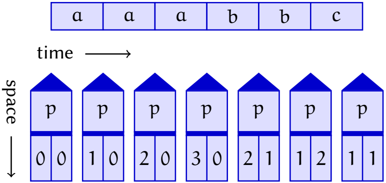
- Nonnegative counter values.
- Counters can only be tested for 0, incremented by $-1, 0, 1$, and copied.

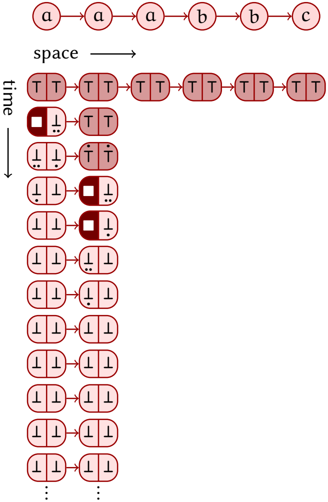
Using time as space



Sumless counter machines

Quasi-acyclic distributed automata



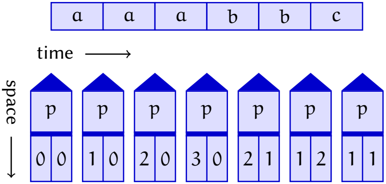


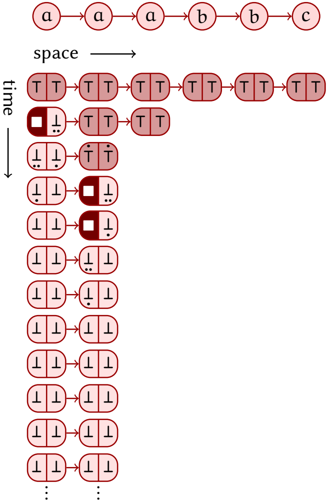
Without loss of generality:

- Nonnegative counter values.
- Counters can only be tested for 0, incremented by $-1, 0, 1,$ and copied.

Using time as space

Sumless counter machines → Quasi-acyclic distributed automata

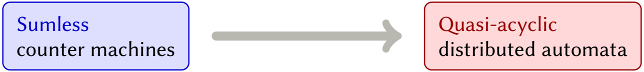


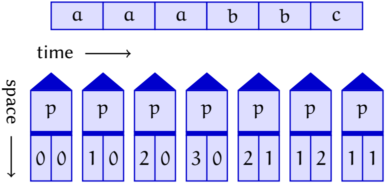


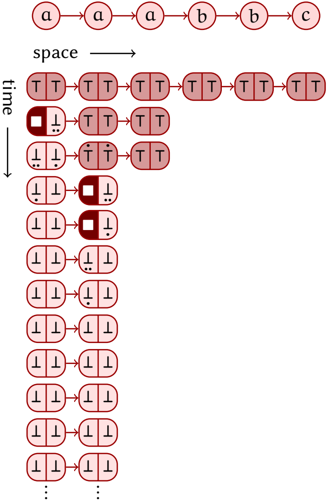
Without loss of generality:

- Nonnegative counter values.
- Counters can only be tested for 0, incremented by $-1, 0, 1,$ and copied.

Using time as space



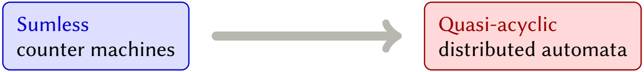


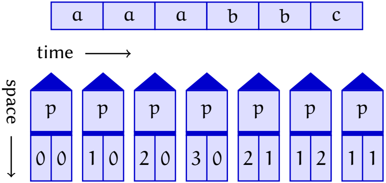


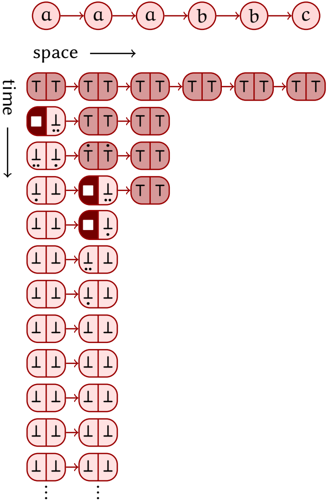
Without loss of generality:

- Nonnegative counter values.
- Counters can only be tested for 0, incremented by $-1, 0, 1,$ and copied.

Using time as space



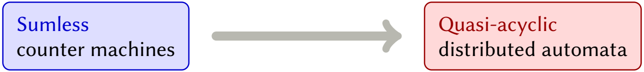


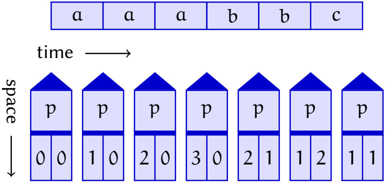


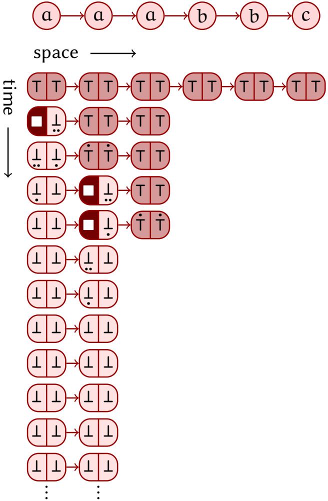
Without loss of generality:

- Nonnegative counter values.
- Counters can only be tested for 0, incremented by $-1, 0, 1,$ and copied.

Using time as space







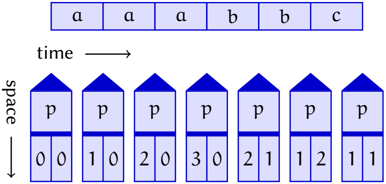
Without loss of generality:

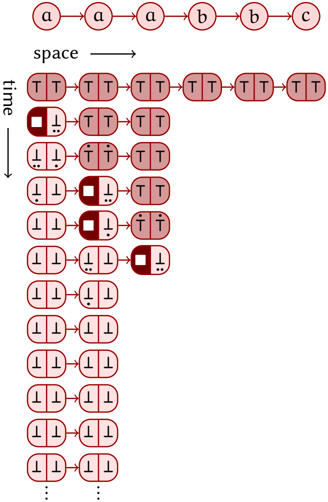
- Nonnegative counter values.
- Counters can only be tested for 0, incremented by $-1, 0, 1$, and copied.

Using time as space

Sumless counter machines

Quasi-acyclic distributed automata



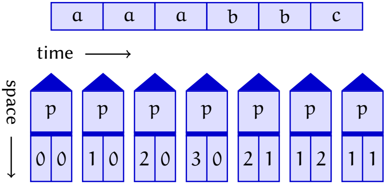


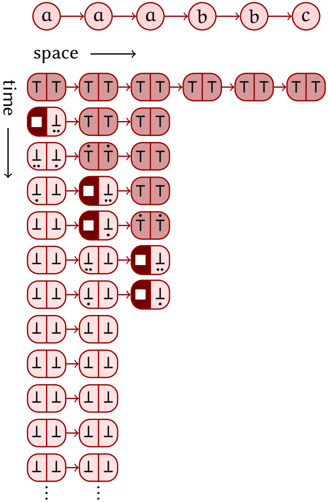
Without loss of generality:

- Nonnegative counter values.
- Counters can only be tested for 0, incremented by $-1, 0, 1$, and copied.

Using time as space

Sumless counter machines → Quasi-acyclic distributed automata

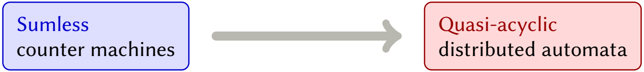




Without loss of generality:

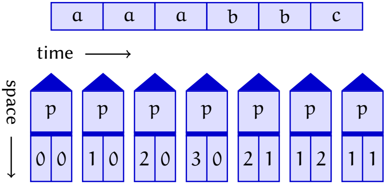
- Nonnegative counter values.
- Counters can only be tested for 0, incremented by $-1, 0, 1$, and copied.

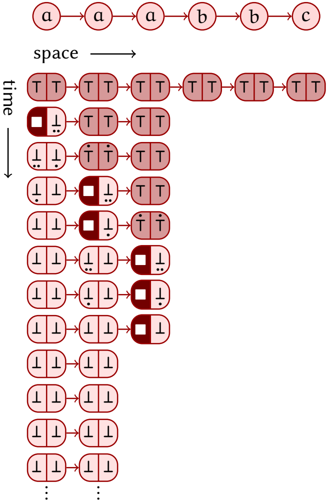
Using time as space



Sumless counter machines

Quasi-acyclic distributed automata





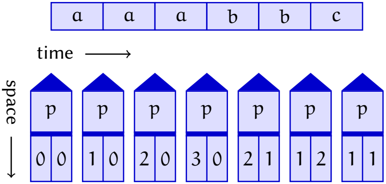
Without loss of generality:

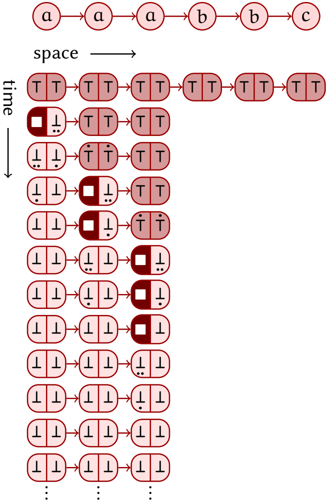
- Nonnegative counter values.
- Counters can only be tested for 0, incremented by $-1, 0, 1$, and copied.

Using time as space

Sumless counter machines

Quasi-acyclic distributed automata

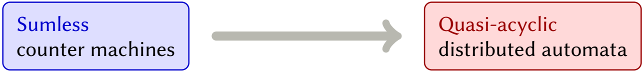




Without loss of generality:

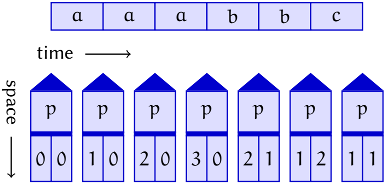
- Nonnegative counter values.
- Counters can only be tested for 0, incremented by $-1, 0, 1,$ and copied.

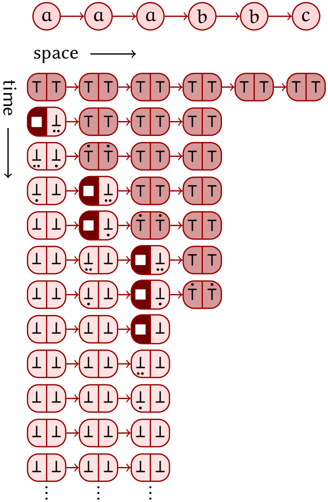
Using time as space



Sumless counter machines

Quasi-acyclic distributed automata



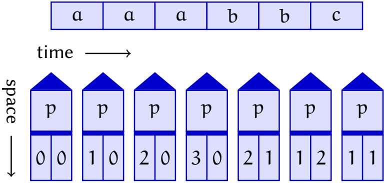


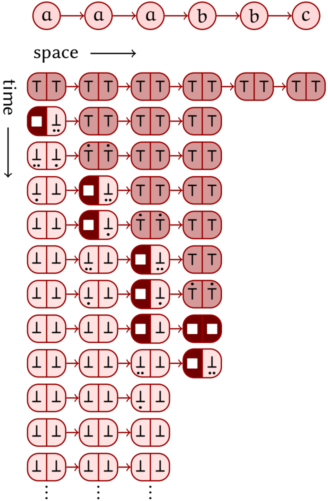
Without loss of generality:

- Nonnegative counter values.
- Counters can only be tested for 0, incremented by $-1, 0, 1$, and copied.

Using time as space

Sumless counter machines → Quasi-acyclic distributed automata

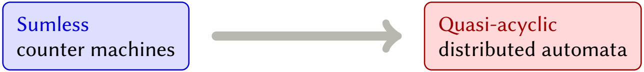


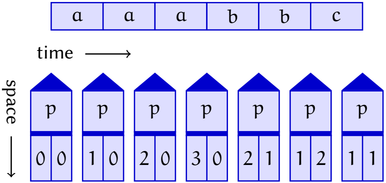


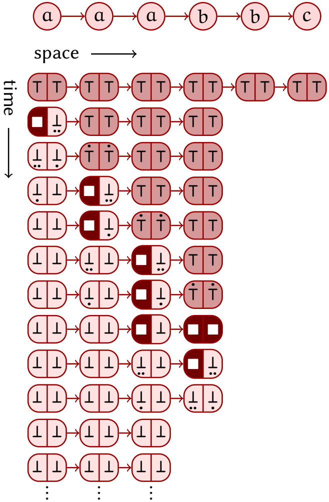
Without loss of generality:

- Nonnegative counter values.
- Counters can only be tested for 0, incremented by $-1, 0, 1,$ and copied.

Using time as space



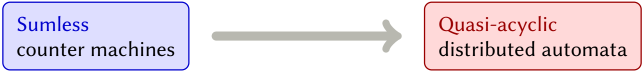




Without loss of generality:

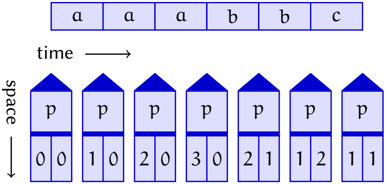
- Nonnegative counter values.
- Counters can only be tested for 0, incremented by $-1, 0, 1$, and copied.

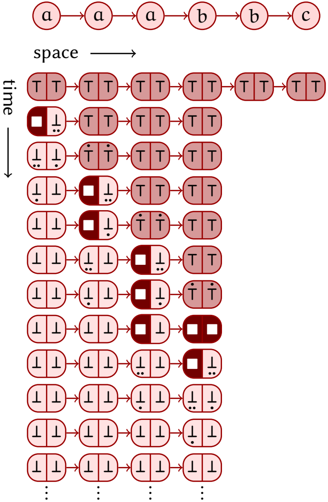
Using time as space



Sumless counter machines

Quasi-acyclic distributed automata

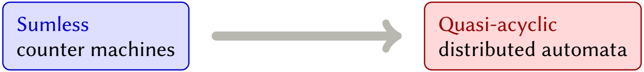




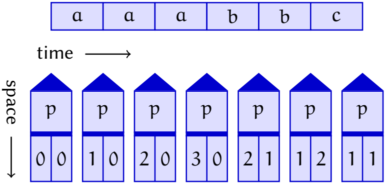
Without loss of generality:

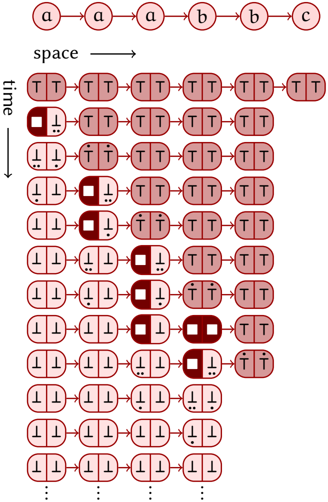
- Nonnegative counter values.
- Counters can only be tested for 0, incremented by $-1, 0, 1$, and copied.

Using time as space


Sumless counter machines

Quasi-acyclic distributed automata

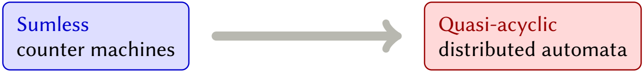




Without loss of generality:

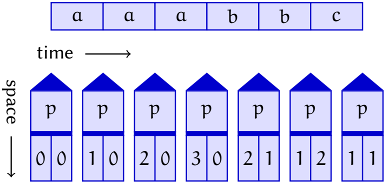
- Nonnegative counter values.
- Counters can only be tested for 0, incremented by $-1, 0, 1$, and copied.

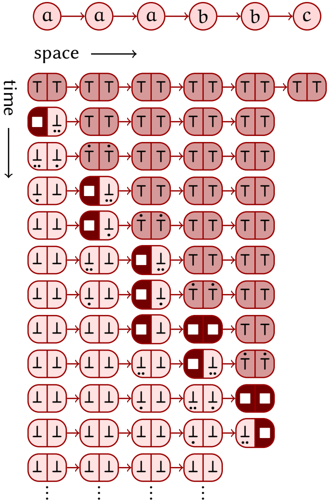
Using time as space



Sumless counter machines

Quasi-acyclic distributed automata

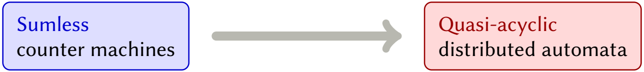




Without loss of generality:

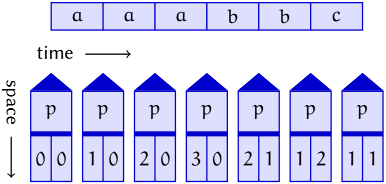
- Nonnegative counter values.
- Counters can only be tested for 0, incremented by $-1, 0, 1,$ and copied.

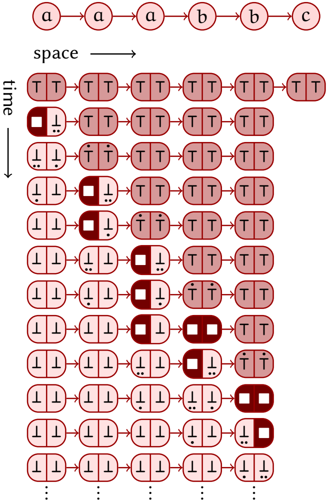
Using time as space



Sumless counter machines

Quasi-acyclic distributed automata





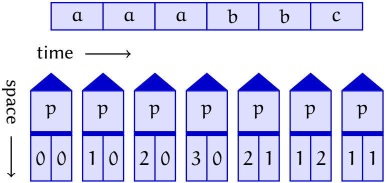
Without loss of generality:

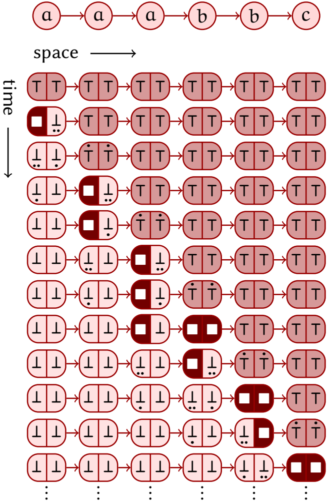
- Nonnegative counter values.
- Counters can only be tested for 0, incremented by $-1, 0, 1$, and copied.

Using time as space

Sumless counter machines

Quasi-acyclic distributed automata

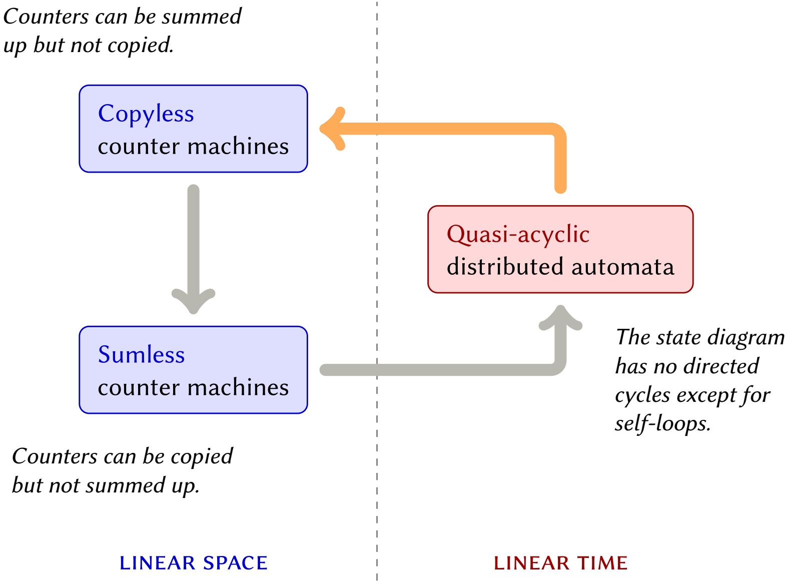




Without loss of generality:

- Nonnegative counter values.
- Counters can only be tested for 0, incremented by $-1, 0, 1$, and copied.

Our translations



Counters can be summed up but not copied.

Copyless counter machines

Sumless counter machines

Counters can be copied but not summed up.

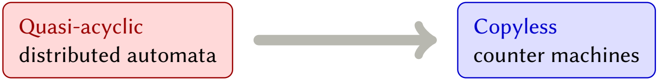
Quasi-acyclic distributed automata

The state diagram has no directed cycles except for self-loops.

LINEAR SPACE

LINEAR TIME

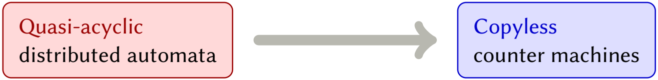
Using space as time



Quasi-acyclic
distributed automata

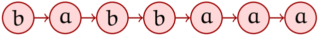
Copyless
counter machines

Using space as time

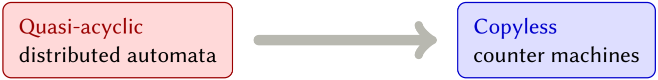


Quasi-acyclic distributed automata

Copyless counter machines

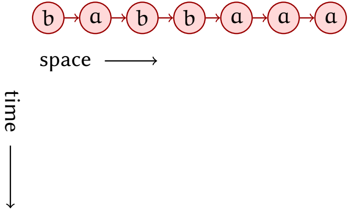


Using space as time



Quasi-acyclic distributed automata

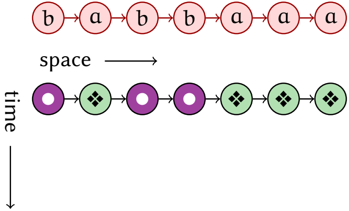
Copyless counter machines



Using space as time

Quasi-acyclic distributed automata

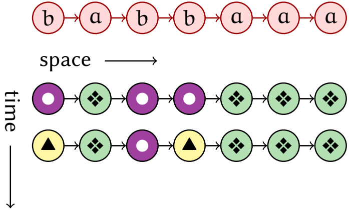
Copyless counter machines



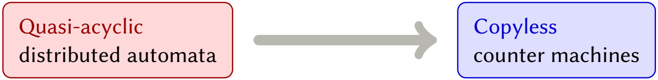
Using space as time

Quasi-acyclic distributed automata

Copyless counter machines

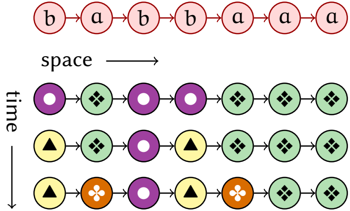


Using space as time

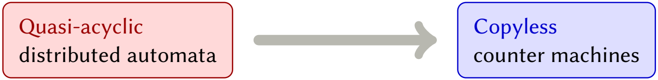


Quasi-acyclic distributed automata

Copyless counter machines

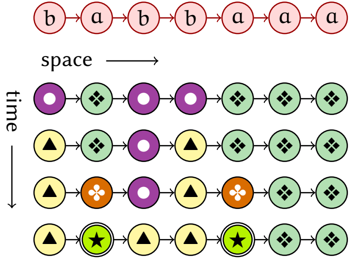


Using space as time



Quasi-acyclic distributed automata

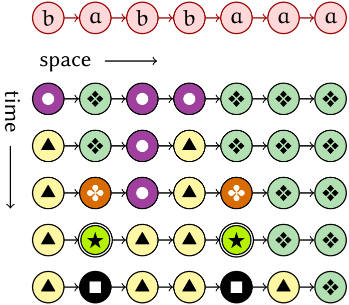
Copyless counter machines



Using space as time

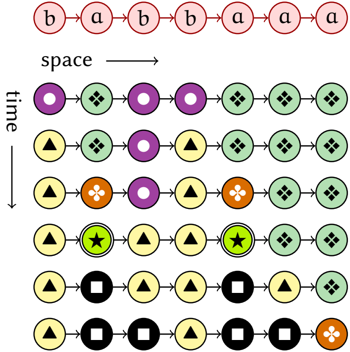
Quasi-acyclic distributed automata

Copyless counter machines

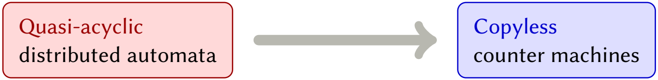


Using space as time

Quasi-acyclic distributed automata → Copyless counter machines

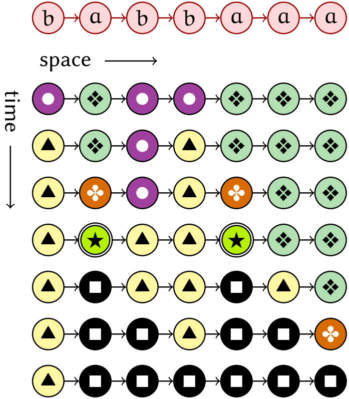


Using space as time



Quasi-acyclic distributed automata

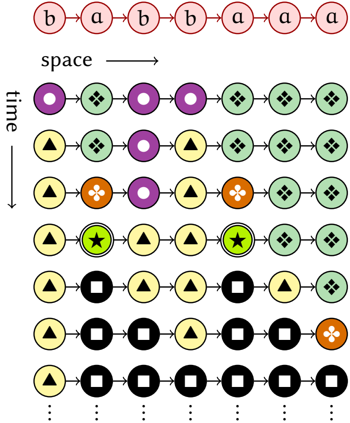
Copyless counter machines



Using space as time

Quasi-acyclic distributed automata

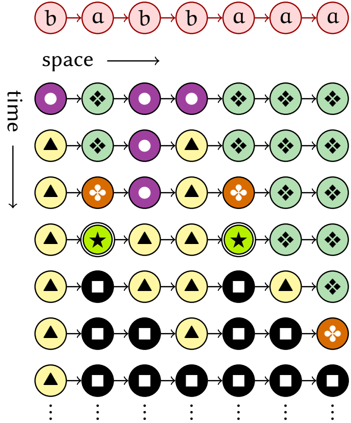
Copyless counter machines



Using space as time

Quasi-acyclic distributed automata

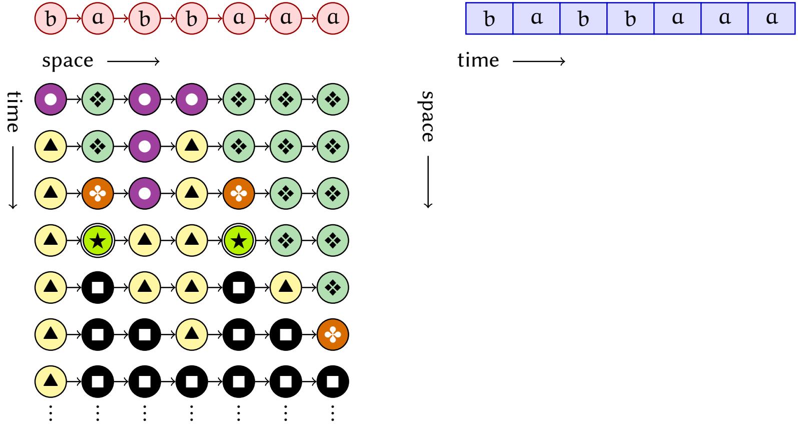
Copyless counter machines



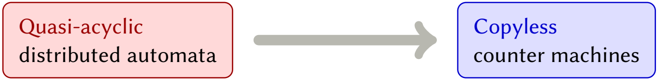
b	a	b	b	a	a	a

Using space as time

Quasi-acyclic distributed automata → Copyless counter machines

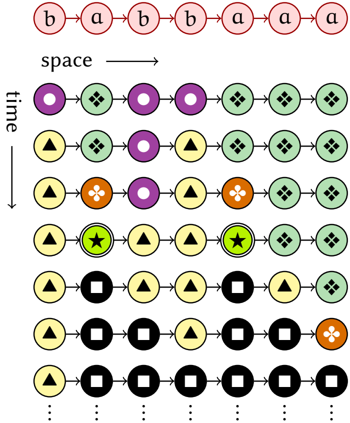


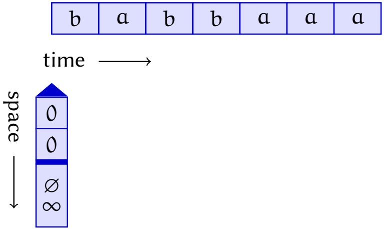
Using space as time



Quasi-acyclic distributed automata

Copyless counter machines



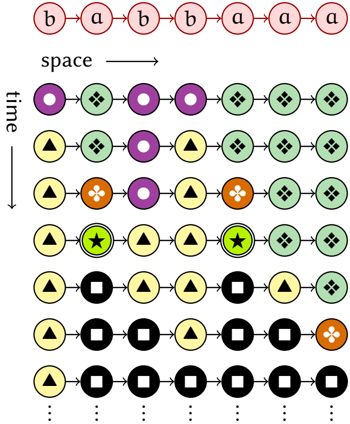


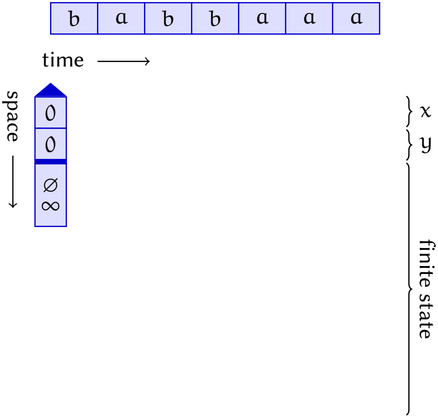
Using space as time

Quasi-acyclic
distributed automata

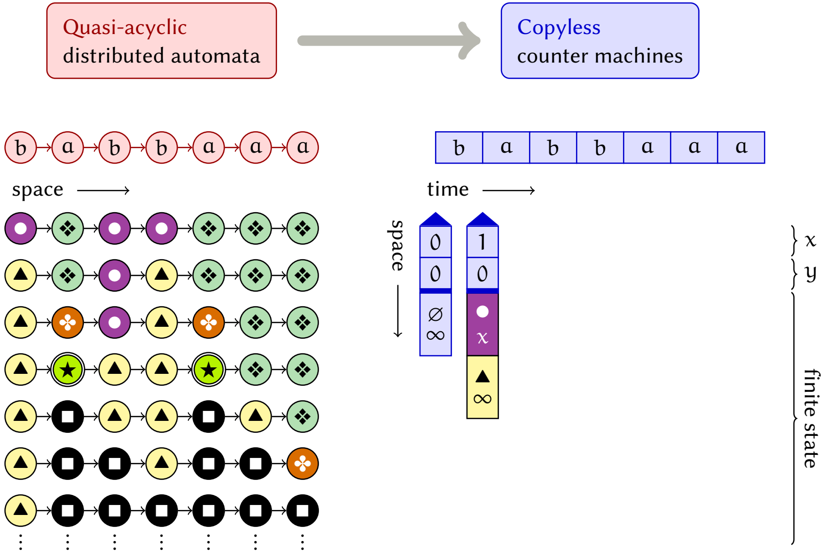
→

Copyless
counter machines



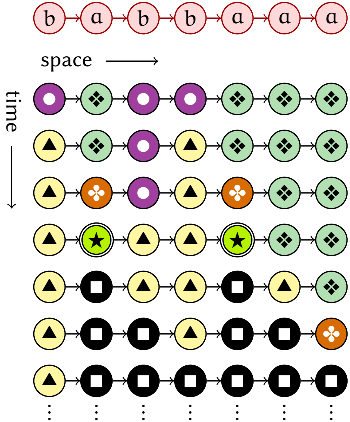


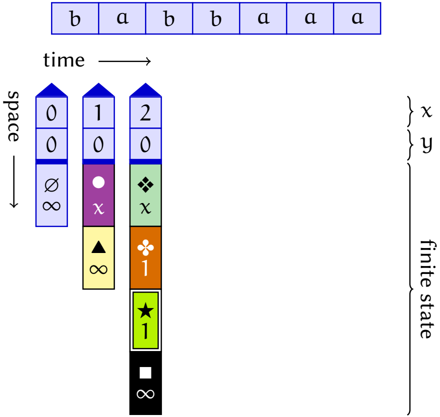
Using space as time



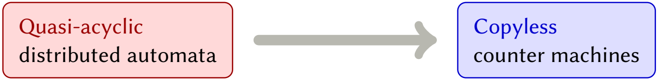
Using space as time

Quasi-acyclic distributed automata → Copyless counter machines

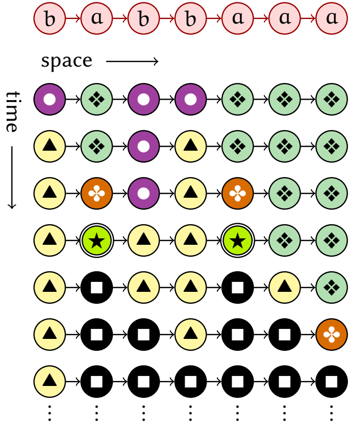


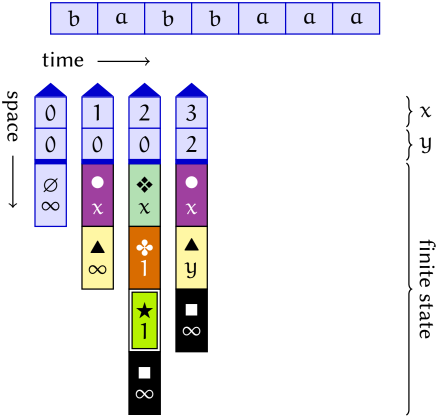


Using space as time


Quasi-acyclic distributed automata

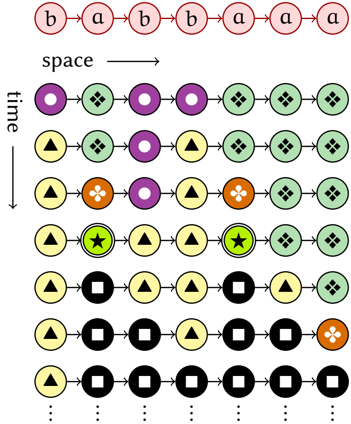
Copyless counter machines

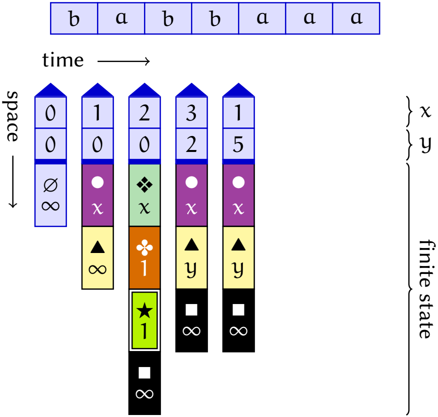




Using space as time

Quasi-acyclic distributed automata → Copyless counter machines

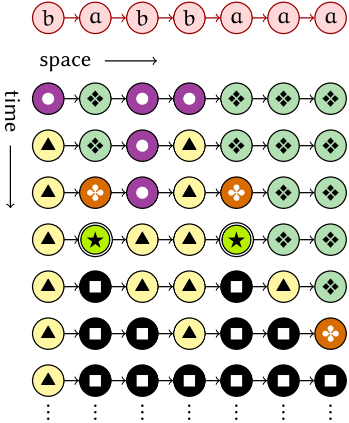


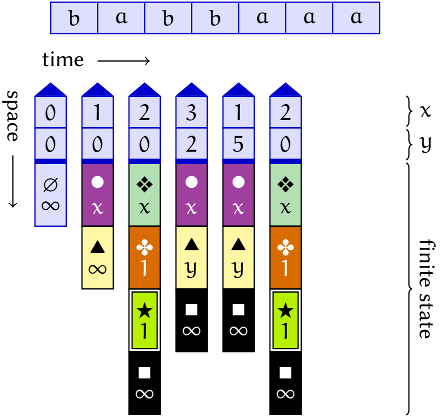


Using space as time

Quasi-acyclic distributed automata

Copyless counter machines

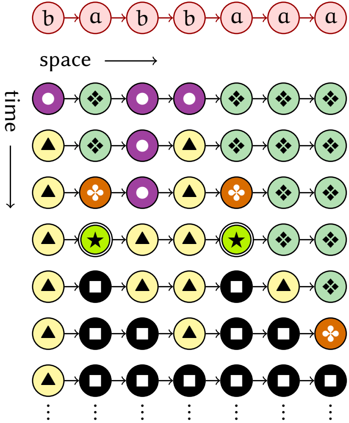


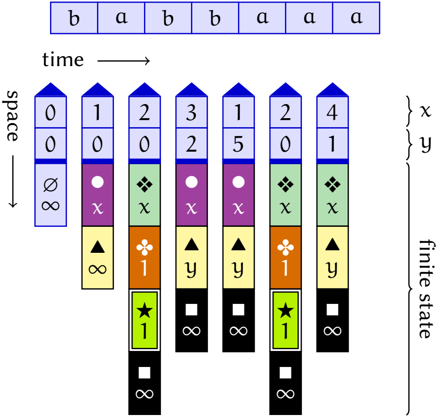


Using space as time

Quasi-acyclic distributed automata

Copyless counter machines

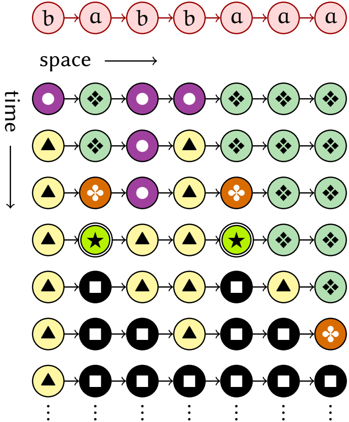


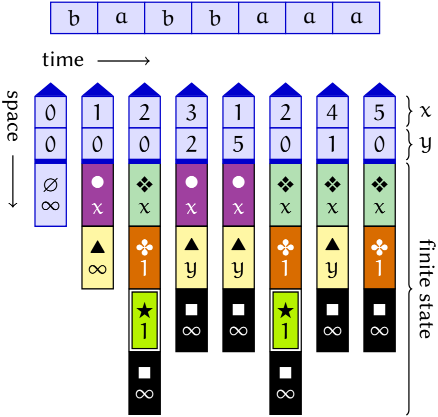


Using space as time

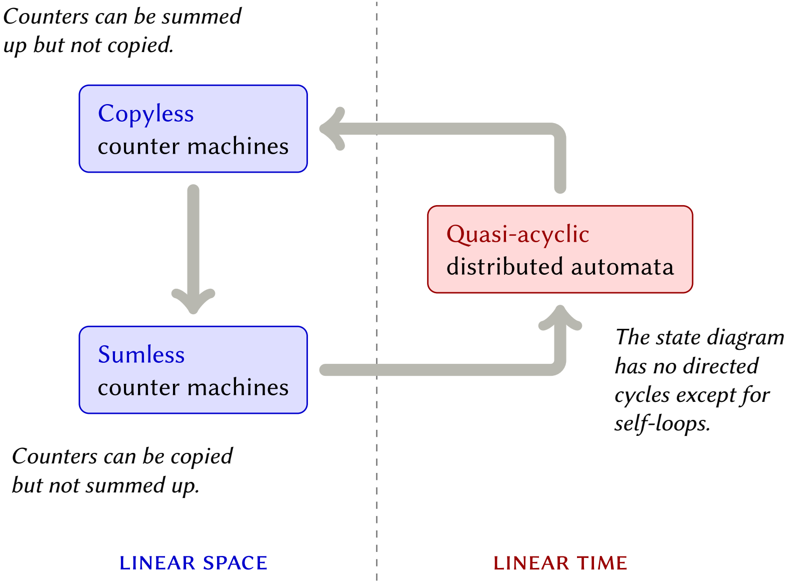
Quasi-acyclic distributed automata

Copyless counter machines





Our translations



Counters can be summed up but not copied.

Copyless counter machines

Sumless counter machines

Counters can be copied but not summed up.

Quasi-acyclic distributed automata

The state diagram has no directed cycles except for self-loops.

LINEAR SPACE

LINEAR TIME

Possible future work

Possible future work

RESTRICTION:

- Counter machines that are both copyless and sumless.

Possible future work

RESTRICTION:

- Counter machines that are both copyless and sumless.

EXTENSIONS:

- Relax requirements (copyless, sumless, quasi-acyclic).

Possible future work

RESTRICTION:

- Counter machines that are both copyless and sumless.

EXTENSIONS:

- Relax requirements (copyless, sumless, quasi-acyclic).
- From one-way to two-way.

Possible future work

RESTRICTION:

- Counter machines that are both copyless and sumless.

EXTENSIONS:

- Relax requirements (copyless, sumless, quasi-acyclic).
- From one-way to two-way.
- From deterministic to nondeterministic (or alternating).

Possible future work

RESTRICTION:

- Counter machines that are both copyless and sumless.

EXTENSIONS:

- Relax requirements (copyless, sumless, quasi-acyclic).
- From one-way to two-way.
- From deterministic to nondeterministic (or alternating).

Thanks!