Linear acceleration for 2D cellular automata

Anaël Grandjean, Victor Poupet, Gaétan Richard, Véronique Terrier

Journées SDA2 - 5 Juillet 2017

Table of Contents

1 Introduction

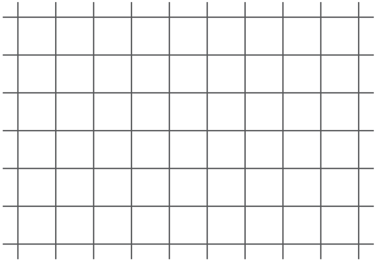
2 Linear speed-up for all neighborhoods

3 Linear speed-up with arbitrary input

4 Conclusion

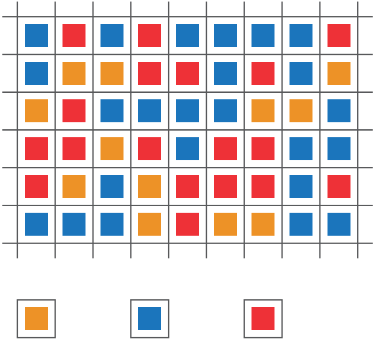
I. Introduction

Cellular Automata



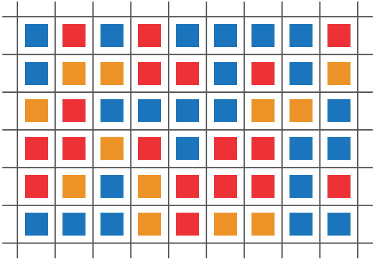
- 2-dimensional ($\mathbb{Z}^2$)

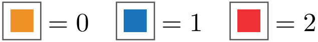
Cellular Automata

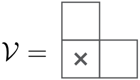


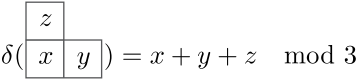
- 2-dimensional ($\mathbb{Z}^2$)
- finite set of states Q

Cellular Automata



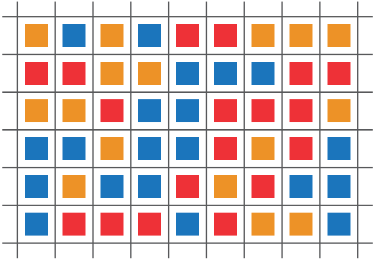


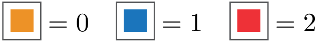


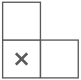


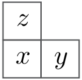
- 2-dimensional ($\mathbb{Z}^2$)
- finite set of states $\mathcal{Q}$
- neighborhood $\mathcal{V} \subseteq_F \mathbb{Z}^2$
- local transition function $\delta : \mathcal{Q}^{\mathcal{V}} \to \mathcal{Q}$

Cellular Automata





$\mathcal{V} =$ 

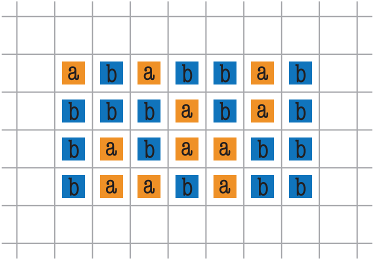


$\delta(\ \cdot\ ) = x + y + z \mod 3$

- 2-dimensional ($\mathbb{Z}^2$)
- finite set of states $\mathcal{Q}$
- neighborhood $\mathcal{V} \subseteq_F \mathbb{Z}^2$
- local transition function $\delta : \mathcal{Q}^{\mathcal{V}} \to \mathcal{Q}$

$\to$ Global transition function

Language recognition



Consider two-dimensional languages over a finite alphabet Σ.

- Finite rectangular words.

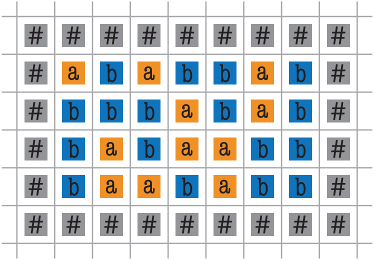
Language recognition

#	#	#	#	#	#	#	#	#
#	a	b	a	b	b	a	b	#
#	b	b	b	a	b	a	b	#
#	b	a	b	a	a	b	b	#
#	b	a	a	b	a	b	b	#
#	#	#	#	#	#	#	#	#

Consider two-dimensional languages over a finite alphabet Σ.

- Finite rectangular words.
- Quiescent state $\# \in Q \setminus \Sigma$ as filler.

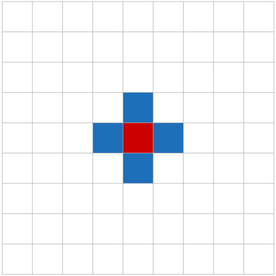
Language recognition

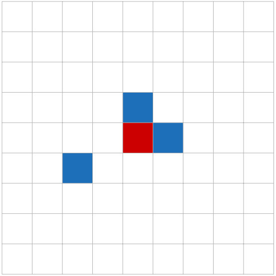


Consider two-dimensional languages over a finite alphabet Σ.

- Finite rectangular words.
- Quiescent state $\# \in Q \setminus \Sigma$ as filler.
- Recognition at the origin cell.

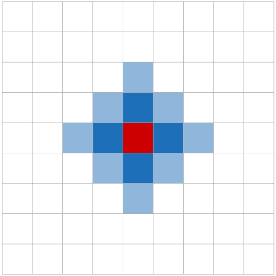
About neighborhoods

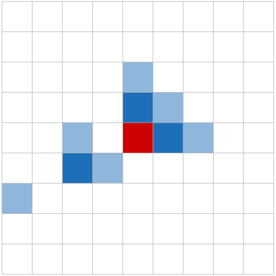




- $\mathcal{V}^{n+1} = \mathcal{V}^n \oplus \mathcal{V}$

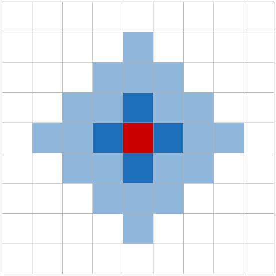
About neighborhoods

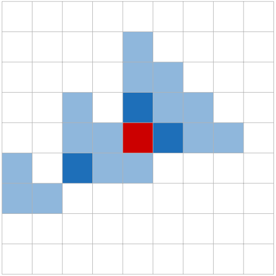




- $\mathcal{V}^{n+1} = \mathcal{V}^{n} \oplus \mathcal{V}$

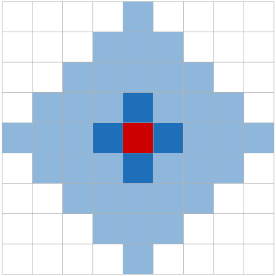
About neighborhoods

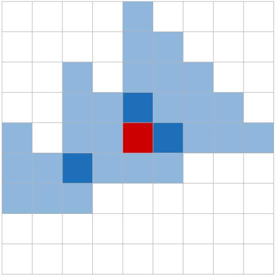




- $\mathcal{V}^{n+1} = \mathcal{V}^n \oplus \mathcal{V}$

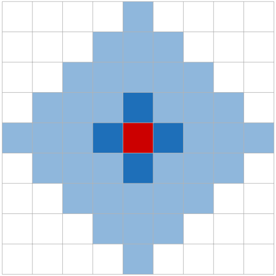
About neighborhoods

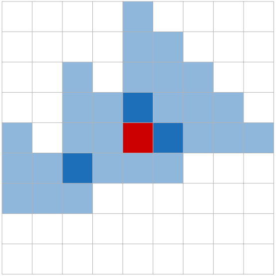




- $\mathcal{V}^{n+1} = \mathcal{V}^n \oplus \mathcal{V}$

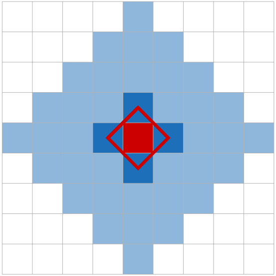
About neighborhoods

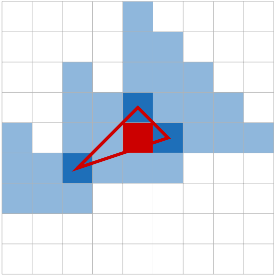




- $\mathcal{V}^{n+1} = \mathcal{V}^n \oplus \mathcal{V}$
- $\mathcal{V}$ is complete iff $\bigcup_{k \in \mathbb{N}} \mathcal{V}^k = \mathbb{Z}^2$

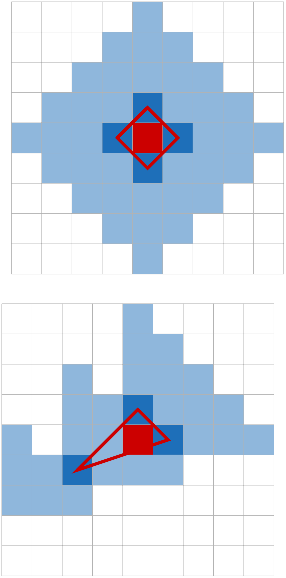
About neighborhoods





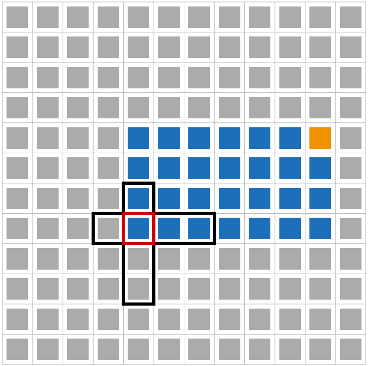
- $\mathcal{V}^{n+1} = \mathcal{V}^n \oplus \mathcal{V}$
- $\mathcal{V}$ is complete iff $\bigcup_{k \in \mathbb{N}} \mathcal{V}^k = \mathbb{Z}^2$
- $\mathrm{CH}(\mathcal{V})$: the smallest polygon of $\mathbb{R}^2$ containing $\mathcal{V}$

About neighborhoods



- $\mathcal{V}^{n+1} = \mathcal{V}^n \oplus \mathcal{V}$
- $\mathcal{V}$ is complete iff $\bigcup_{k\in\mathbb{N}} \mathcal{V}^k = \mathbb{Z}^2$
- $\mathrm{CH}(\mathcal{V})$: the smallest polygon of $\mathbb{R}^2$ containing $\mathcal{V}$
- convex : $\mathcal{V} = \mathrm{CH}(\mathcal{V}) \bigcap \mathbb{Z}^2$

Real time

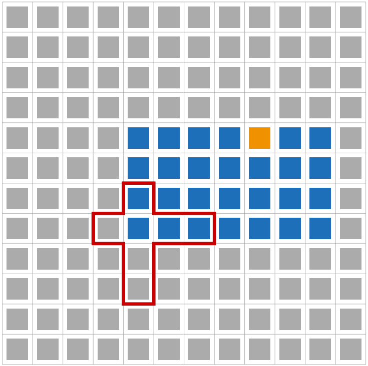


The real time $(\mathrm{RT}_{\mathcal{V}})$ is the lowest t such that $[\![0, n]\!] \times [\![0, m]\!] \subset \mathcal{V}^t(0)$.

$\mathrm{CA}_{\mathcal{V}}(\mathrm{RT})$ is the set of all languages recognizable in real time with $\mathcal{V}$.

$\mathrm{CA}_{\mathcal{V}}(\mathrm{LT}) = \bigcup_{n \in \mathbb{N}} \mathrm{CA}_{\mathcal{V}}(n\,\mathrm{RT})$ is the set of all languages recognizable in linear time with $\mathcal{V}$.

Real time

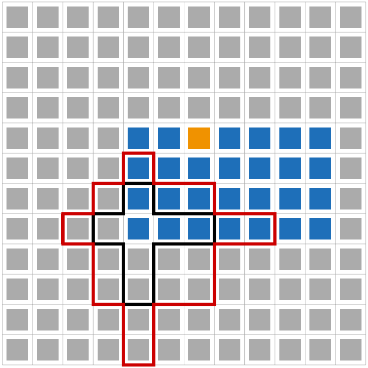


The real time $(\mathrm{RT}_{\mathcal{V}})$ is the lowest t such that $[\![0,n]\!] \times [\![0,m]\!] \subset \mathcal{V}^t(0)$.

$\mathrm{CA}_{\mathcal{V}}(\mathrm{RT})$ is the set of all languages recognizable in real time with $\mathcal{V}$.

$\mathrm{CA}_{\mathcal{V}}(\mathrm{LT}) = \bigcup_{n\in\mathbb{N}} \mathrm{CA}_{\mathcal{V}}(n\,\mathrm{RT})$ is the set of all languages recognizable in linear time with $\mathcal{V}$.

Real time

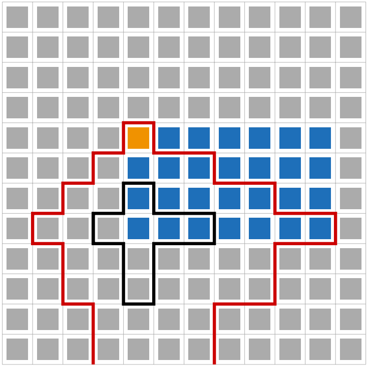


The real time $(\mathrm{RT}_{\mathcal{V}})$ is the lowest t such that $[\![0,n]\!] \times [\![0,m]\!] \subset \mathcal{V}^t(0)$.

$\mathrm{CA}_{\mathcal{V}}(\mathrm{RT})$ is the set of all languages recognizable in real time with $\mathcal{V}$.

$\mathrm{CA}_{\mathcal{V}}(\mathrm{LT}) = \bigcup_{n\in\mathbb{N}} \mathrm{CA}_{\mathcal{V}}(n\,\mathrm{RT})$ is the set of all languages recognizable in linear time with $\mathcal{V}$.

Real time

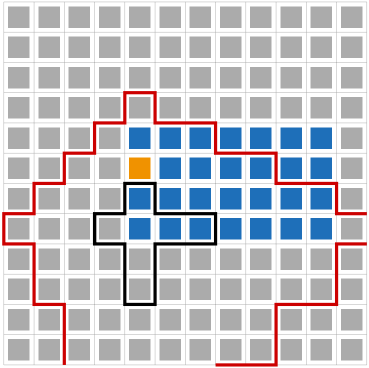


The real time $(\mathrm{RT}_{\mathcal{V}})$ is the lowest t such that $[\![0,n]\!] \times [\![0,m]\!] \subset \mathcal{V}^t(0)$.

$\mathrm{CA}_{\mathcal{V}}(\mathrm{RT})$ is the set of all languages recognizable in real time with $\mathcal{V}$.

$\mathrm{CA}_{\mathcal{V}}(\mathrm{LT}) = \bigcup_{n\in\mathbb{N}} \mathrm{CA}_{\mathcal{V}}(n\,\mathrm{RT})$ is the set of all languages recognizable in linear time with $\mathcal{V}$.

Real time

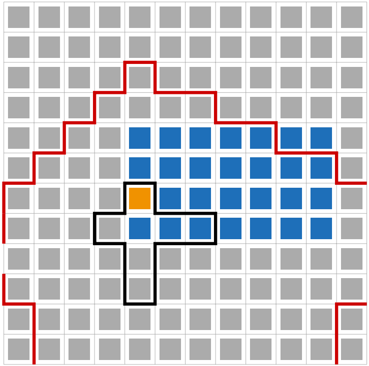


The real time $(\mathrm{RT}_{\mathcal{V}})$ is the lowest t such that $[\![0, n]\!] \times [\![0, m]\!] \subset \mathcal{V}^t(0)$.

$\mathrm{CA}_{\mathcal{V}}(\mathrm{RT})$ is the set of all languages recognizable in real time with $\mathcal{V}$.

$\mathrm{CA}_{\mathcal{V}}(\mathrm{LT}) = \bigcup_{n \in \mathbb{N}} \mathrm{CA}_{\mathcal{V}}(n\,\mathrm{RT})$ is the set of all languages recognizable in linear time with $\mathcal{V}$.

Real time

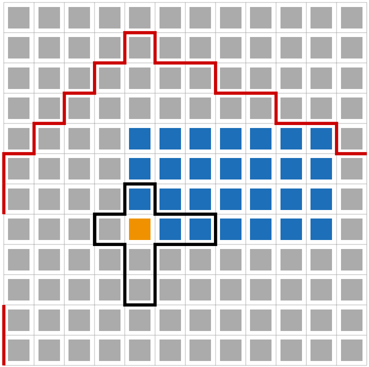


The real time $(\mathrm{RT}_{\mathcal{V}})$ is the lowest t such that $[\![0,n]\!] \times [\![0,m]\!] \subset \mathcal{V}^t(0)$.

$\mathrm{CA}_{\mathcal{V}}(\mathrm{RT})$ is the set of all languages recognizable in real time with $\mathcal{V}$.

$\mathrm{CA}_{\mathcal{V}}(\mathrm{LT}) = \bigcup_{n\in\mathbb{N}} \mathrm{CA}_{\mathcal{V}}(n\,\mathrm{RT})$ is the set of all languages recognizable in linear time with $\mathcal{V}$.

Real time



The real time $(\mathrm{RT}_{\mathcal{V}})$ is the lowest t such that $[\![0,n]\!] \times [\![0,m]\!] \subset \mathcal{V}^t(0)$.

$\mathrm{CA}_{\mathcal{V}}(\mathrm{RT})$ is the set of all languages recognizable in real time with $\mathcal{V}$.

$\mathrm{CA}_{\mathcal{V}}(\mathrm{LT}) = \bigcup_{n\in\mathbb{N}} \mathrm{CA}_{\mathcal{V}}(n\,\mathrm{RT})$ is the set of all languages recognizable in linear time with $\mathcal{V}$.

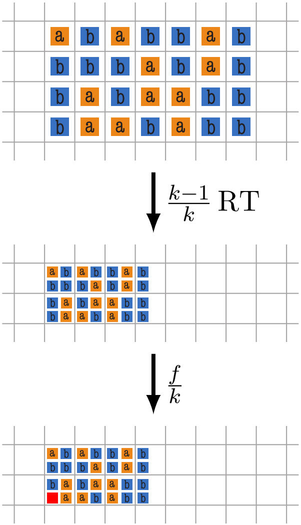
II. Linear speed-up for all neighborhoods

Linear speed-up theorem

Theorem

$$\mathrm{CA}_{\mathcal{V}}(\mathrm{RT} + f) = \mathrm{CA}_{\mathcal{V}}((1+\epsilon)\,\mathrm{RT} + \epsilon f)$$

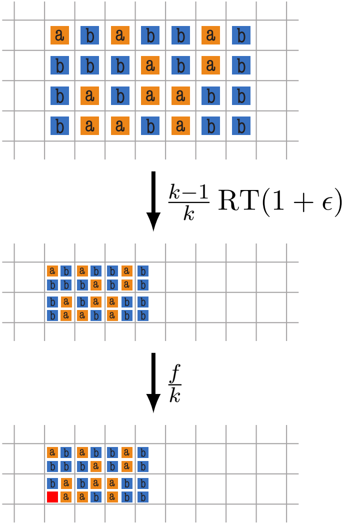
How to accelerate



Two main steps :

- Compress the input
- Compute faster

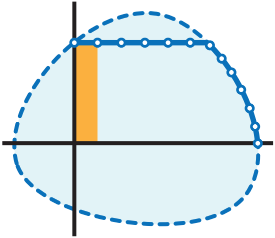
How to accelerate



Two main steps :

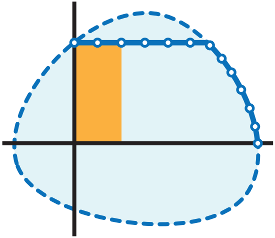
- Compress the input
- Compute faster

Approximation with rectangles



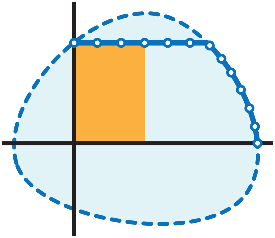
- each $\mathcal{R}_i \subset \mathcal{V}$
- $\mathcal{V}$ can simulate each $\mathcal{R}_i$
- for each input at least one $\mathcal{R}_i$ has a real time very close to $\mathcal{V}$

Approximation with rectangles



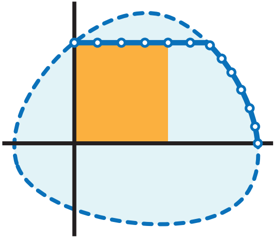
- each $\mathcal{R}_i \subset \mathcal{V}$
- $\mathcal{V}$ can simulate each $\mathcal{R}_i$
- for each input at least one $\mathcal{R}_i$ has a real time very close to $\mathcal{V}$

Approximation with rectangles



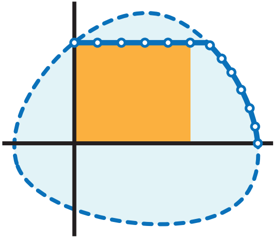
- each $\mathcal{R}_i \subset \mathcal{V}$
- $\mathcal{V}$ can simulate each $\mathcal{R}_i$
- for each input at least one $\mathcal{R}_i$ has a real time very close to $\mathcal{V}$

Approximation with rectangles



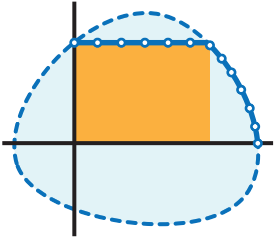
- each $\mathcal{R}_i \subset \mathcal{V}$
- $\mathcal{V}$ can simulate each $\mathcal{R}_i$
- for each input at least one $\mathcal{R}_i$ has a real time very close to $\mathcal{V}$

Approximation with rectangles



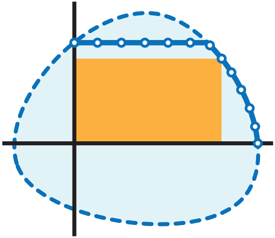
- each $\mathcal{R}_i \subset \mathcal{V}$
- $\mathcal{V}$ can simulate each $\mathcal{R}_i$
- for each input at least one $\mathcal{R}_i$ has a real time very close to $\mathcal{V}$

Approximation with rectangles



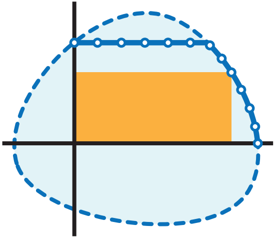
- each $\mathcal{R}_i \subset \mathcal{V}$
- $\mathcal{V}$ can simulate each $\mathcal{R}_i$
- for each input at least one $\mathcal{R}_i$ has a real time very close to $\mathcal{V}$

Approximation with rectangles



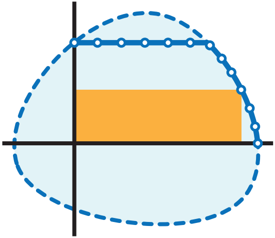
- each $\mathcal{R}_i \subset \mathcal{V}$
- $\mathcal{V}$ can simulate each $\mathcal{R}_i$
- for each input at least one $\mathcal{R}_i$ has a real time very close to $\mathcal{V}$

Approximation with rectangles



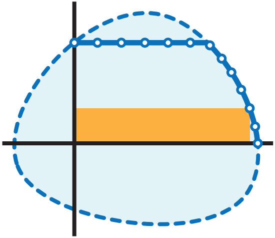
- each $\mathcal{R}_i \subset \mathcal{V}$
- $\mathcal{V}$ can simulate each $\mathcal{R}_i$
- for each input at least one $\mathcal{R}_i$ has a real time very close to $\mathcal{V}$

Approximation with rectangles



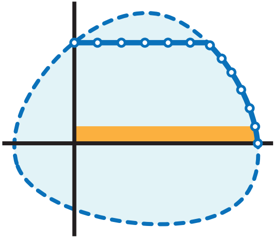
- each $\mathcal{R}_i \subset \mathcal{V}$
- $\mathcal{V}$ can simulate each $\mathcal{R}_i$
- for each input at least one $\mathcal{R}_i$ has a real time very close to $\mathcal{V}$

Approximation with rectangles



- each $\mathcal{R}_i \subset \mathcal{V}$
- $\mathcal{V}$ can simulate each $\mathcal{R}_i$
- for each input at least one $\mathcal{R}_i$ has a real time very close to $\mathcal{V}$

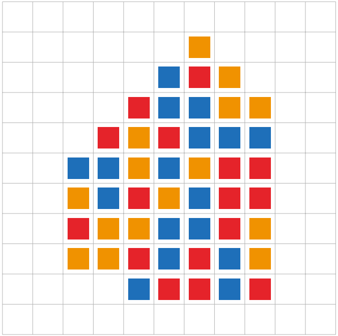
Approximation with rectangles



- each $\mathcal{R}_i \subset \mathcal{V}$
- $\mathcal{V}$ can simulate each $\mathcal{R}_i$
- for each input at least one $\mathcal{R}_i$ has a real time very close to $\mathcal{V}$

III. Linear speed-up with arbitrary input

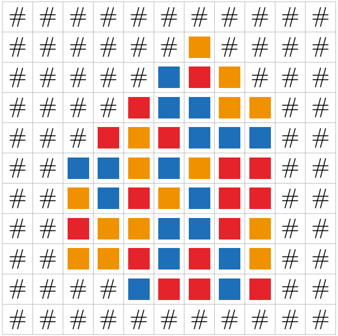
Language recognition



Consider two-dimensional languages over a finite alphabet Σ.

- Finite connex words.

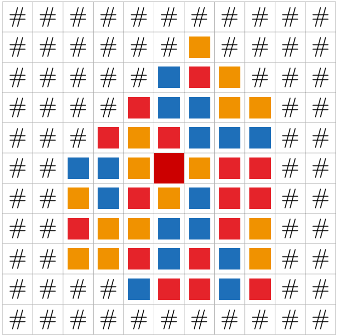
Language recognition



Consider two-dimensional languages over a finite alphabet Σ.

- Finite connex words.
- Quiescent state $\# \in Q \setminus \Sigma$ as filler.

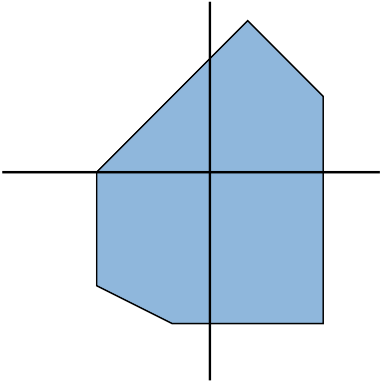
Language recognition



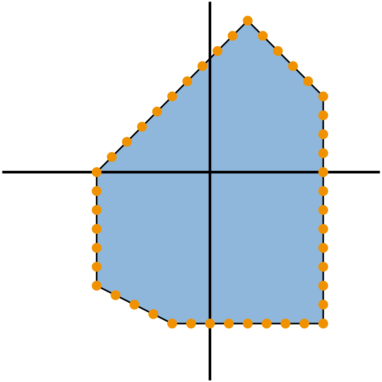
Consider two-dimensional languages over a finite alphabet Σ.

- Finite connex words.
- Quiescent state $\# \in \mathcal{Q} \setminus \Sigma$ as filler.
- Recognition at a given origin cell.

Compression

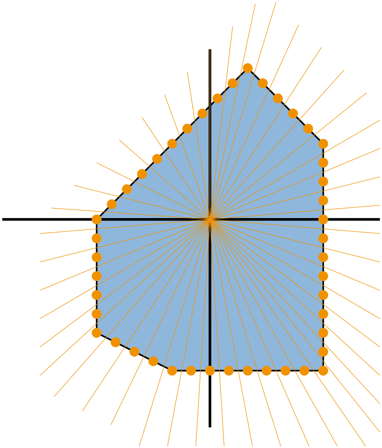


Compression



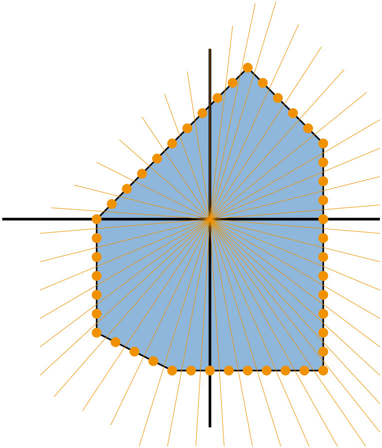
- Choose a finite number of directions.

Compression



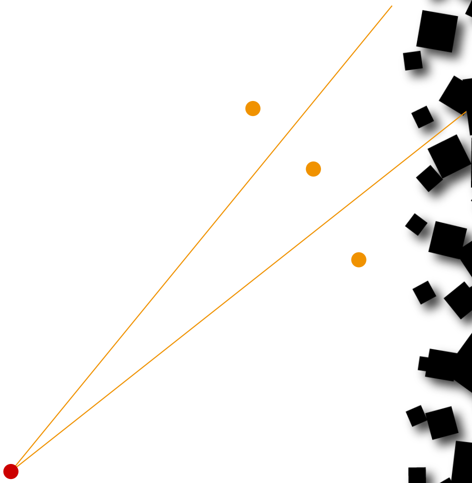
- Choose a finite number of directions.
- Partition the plane into cones.

Compression



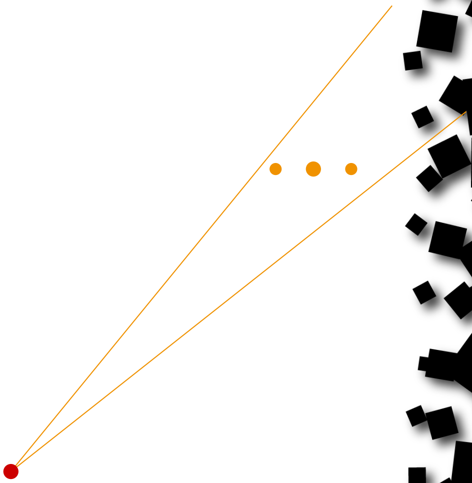
- Choose a finite number of directions.
- Partition the plane into cones.
- Compress each cone on a different layer

Compression of a cone



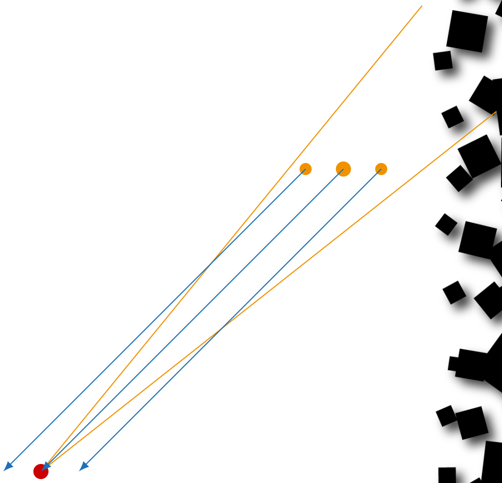
- Compress only the cone.

Compression of a cone



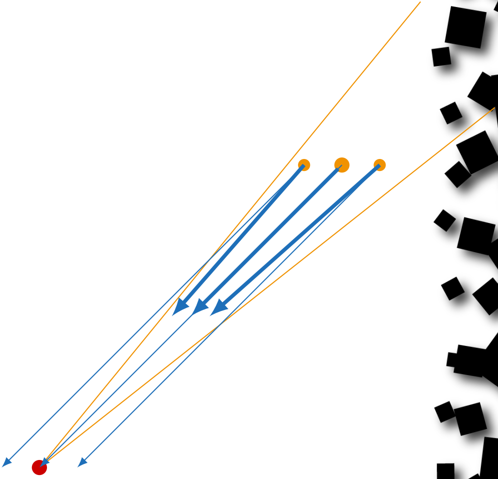
- Compress only the cone.

Compression of a cone



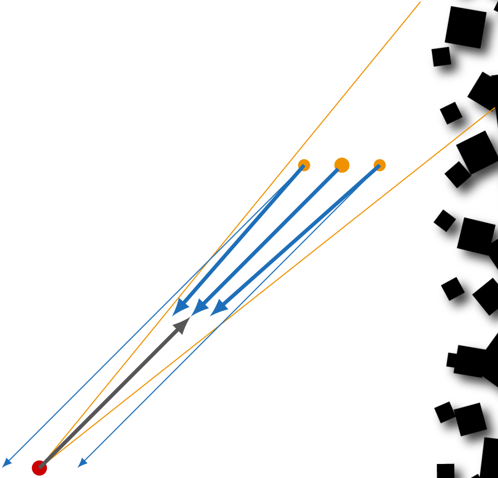
- Compress only the cone.
- All information goes into the same direction.

Compression of a cone



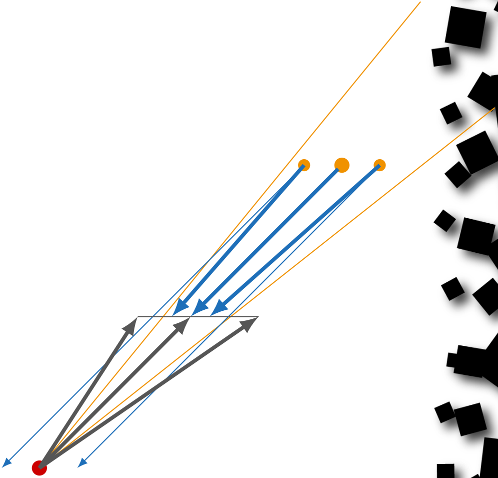
- Compress only the cone.
- All information goes into the same direction.
- Origin sends signals to stop information.

Compression of a cone



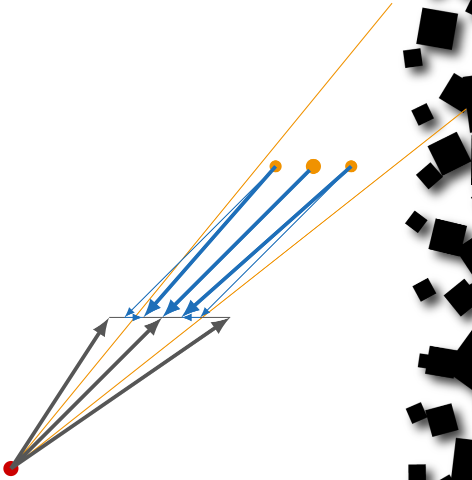
- Compress only the cone.
- All information goes into the same direction.
- Origin sends signals to stop information.

Compression of a cone



- Compress only the cone.
- All information goes into the same direction.
- Origin sends signals to stop information.

Compression of a cone



- Compress only the cone.
- All information goes into the same direction.
- Origin sends signals to stop information.
- Correction is small.

IV. Conclusion

Conclusion

- This construction works in any dimensin.

- In this model, $\mathrm{CA}_{\mathcal{V}}(\mathrm{RT}) \neq \mathrm{CA}_{\mathcal{V}}((1+\epsilon)\,\mathrm{RT})$.

- Much work to do to define the model properly