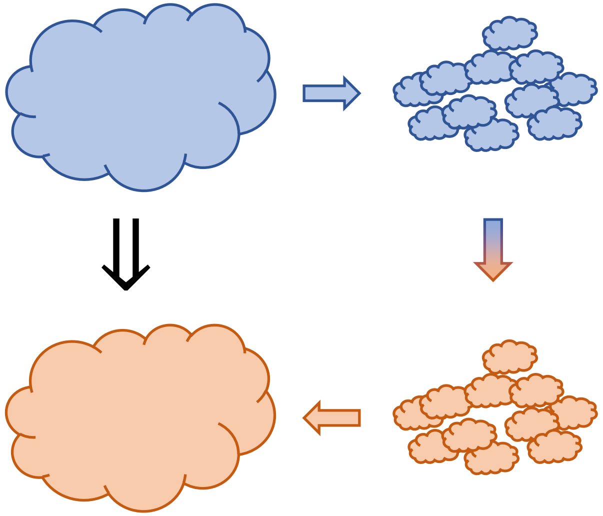
Cellular Automata and Kan Extensions

Alexandre Fernandez, Luidnel Maignan, Antoine Spicher

Automata 2021

Global transformations

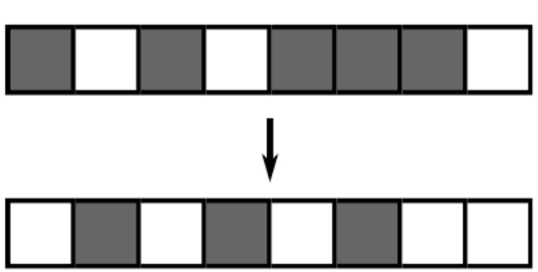
- Synchronous & local & deterministic systems



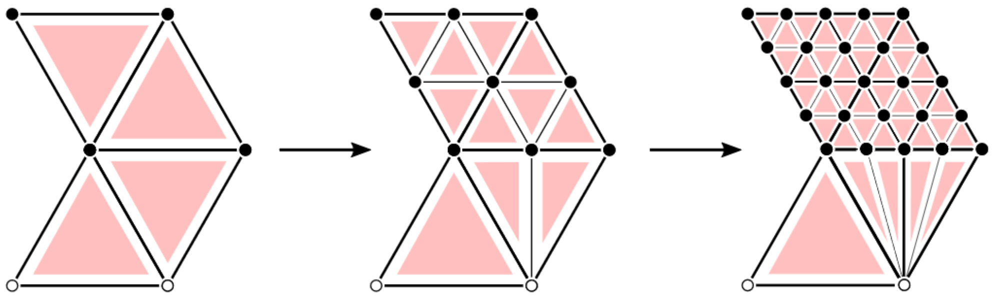
Global transformations

- Synchronous & local & deterministic systems

- Over variety of structures
  - groups, words, trees, graphs, …

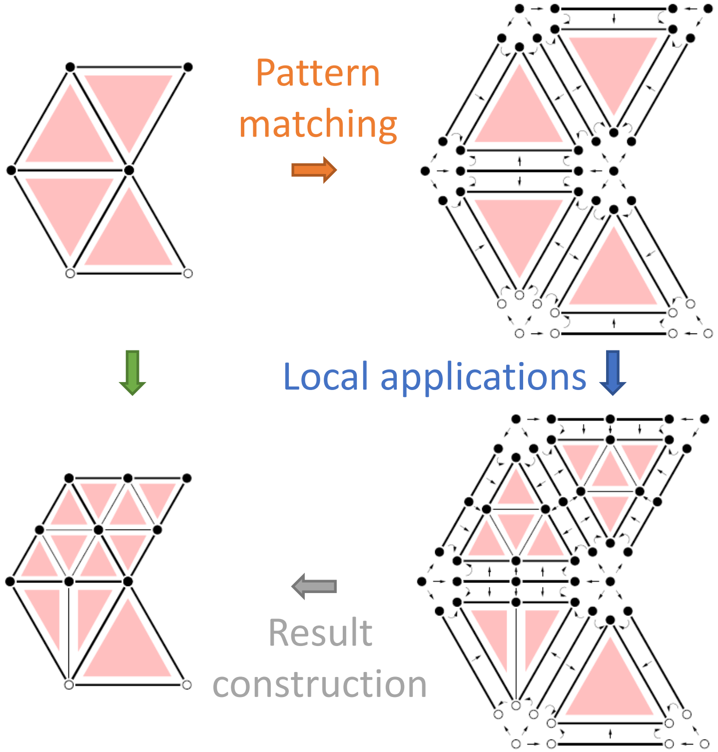


$$abaababa$$
$$\Downarrow$$
$$abaababaabaab$$
$$\Downarrow$$
$$abaababaabaababaababa$$



Global transformations

- Synchronous & local & deterministic systems
- Over variety of structures
  - groups, words, trees, graphs, …
- Category theory



$$T(\mathcal{K}) = \mathrm{Colim}(\mathrm{P} \circ \mathrm{Proj}_{\mathrm{L}/\mathcal{K}})$$

Local vs Global relation

- In cellular automata :
  - Local behavior extended to global
    - Computational description
  - Continuous & uniform & shift equiv global behavior
    - Topological description

Local vs Global relation

- In cellular automata :
  - Local behavior extended to global
    - Computational description
  - Continuous & uniform & shift equiv global behavior
    - Topological description
  - Kan Extensions !
    - Categorical description

Outline

- Two ways to extend with an example
- The poset of sub-configurations
- The two extensions are kan extensions

Outline

- Two ways to extend with an example
- The poset of sub-configurations
- The two extensions are kan extensions

Traffic rule automaton

- Represents traffic, bi-infinite road

Traffic rule automaton

- Represents traffic, bi-infinite road

- Cars going to the right

...

...

Traffic rule automaton

- Represents traffic, bi-infinite road

- Cars going to the right

- A car goes right when no car in front

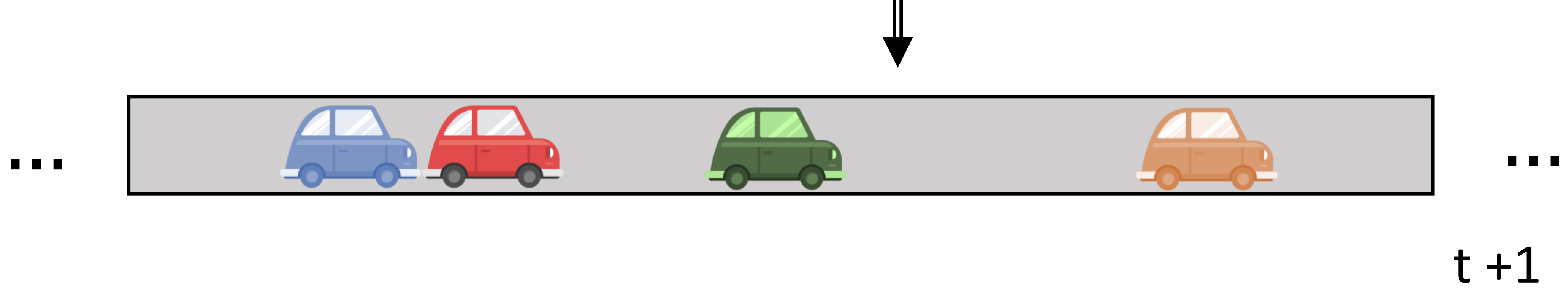
t

Traffic rule automaton

- Represents traffic, bi-infinite road

- Cars going to the right

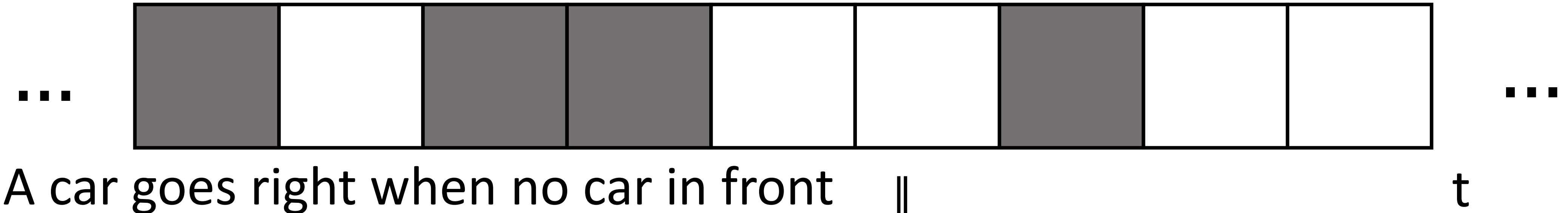
- A car goes right when no car in front



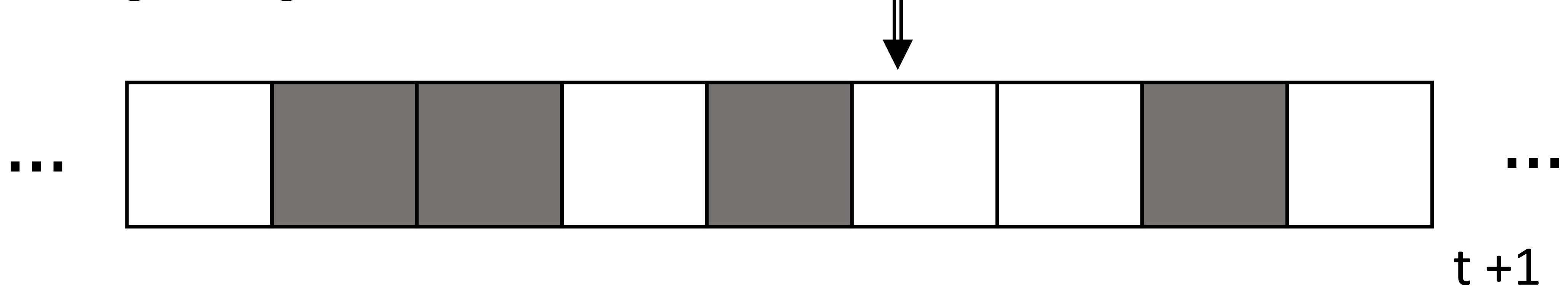
Traffic rule automaton

- Represents traffic, bi-infinite road

- Cars going to the right



- A car goes right when no car in front



Traffic rule automaton

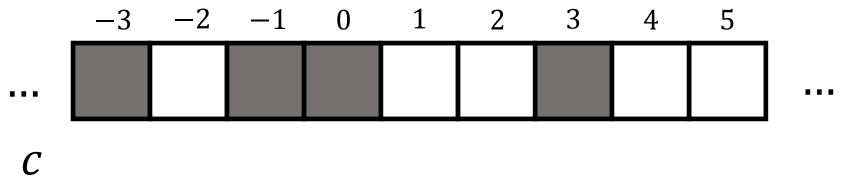
- Q : Finite set of states

$$Q = \{\square, \blacksquare\}$$

- G : Group (space)

$$G = (\mathbb{Z}, +)$$

- $c \in Q^G$: configurations

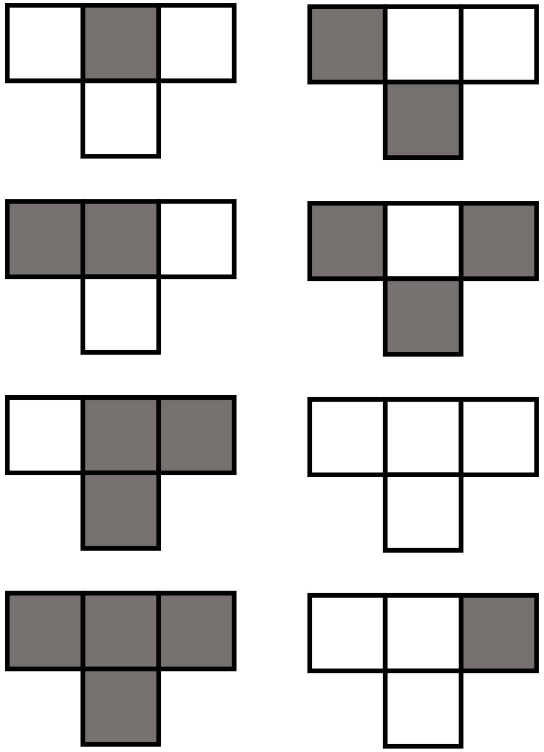


- $N \subseteq G$: finite neighborhood

$$N = \{-1, 0, 1\}$$

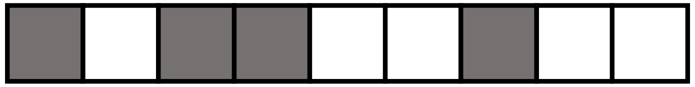
Space time diagram

Evolution Rules



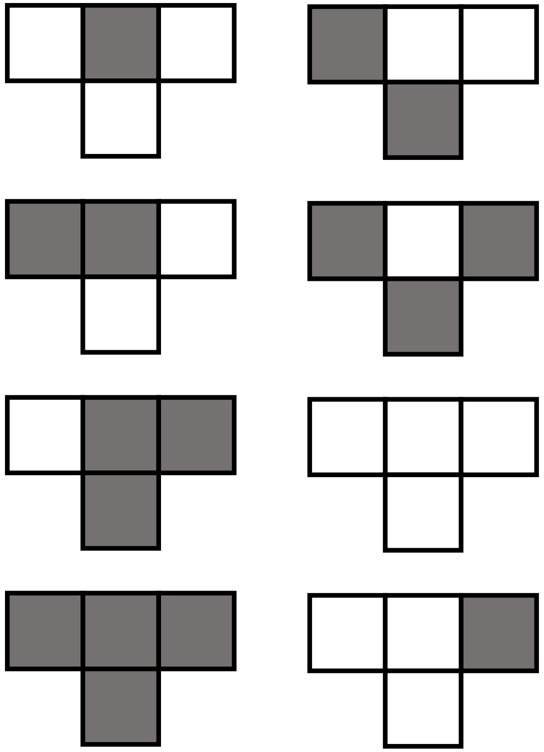
- Evolution for known part ?

Space time diagram



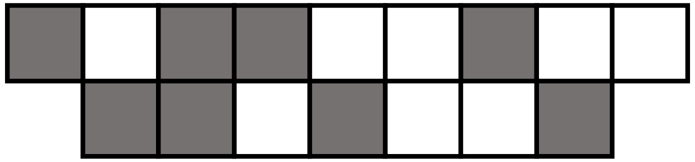
←State

Evolution Rules



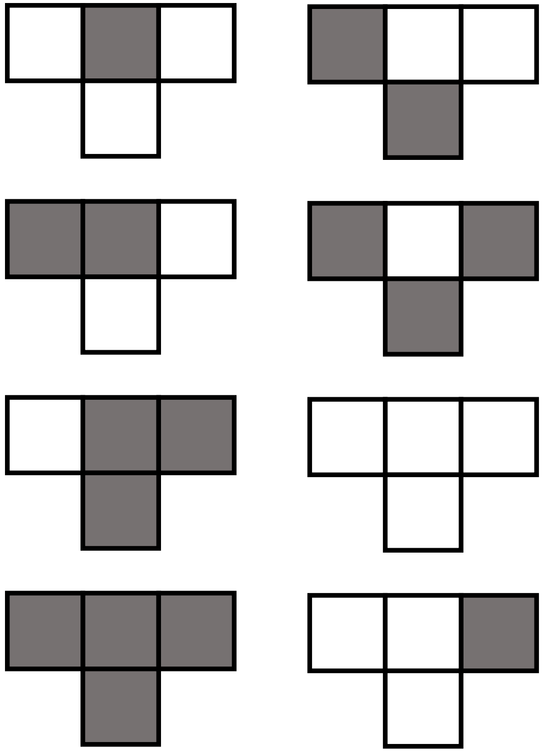
- Evolution for sub-configurations

Space time diagram



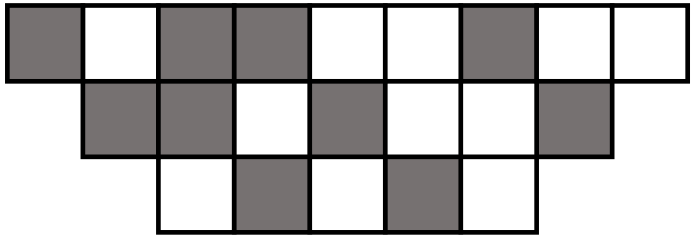
←State

Evolution Rules



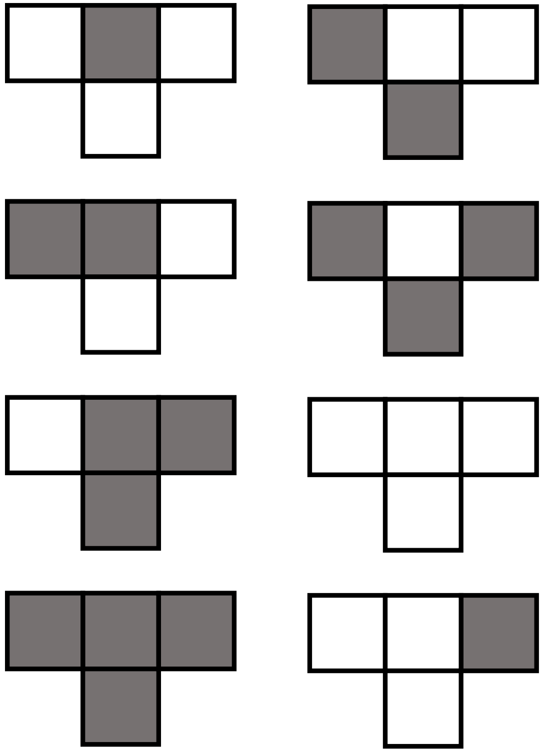
- Evolution for sub-configurations

Space time diagram



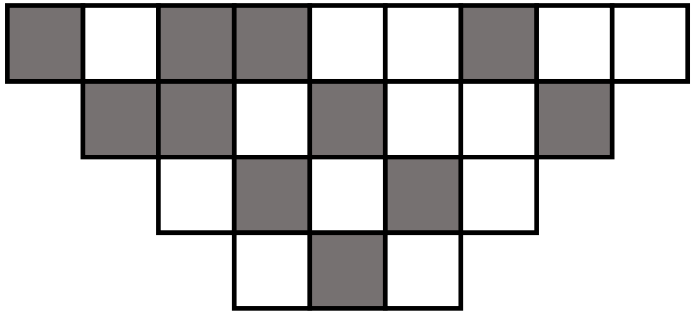
←State

Evolution Rules



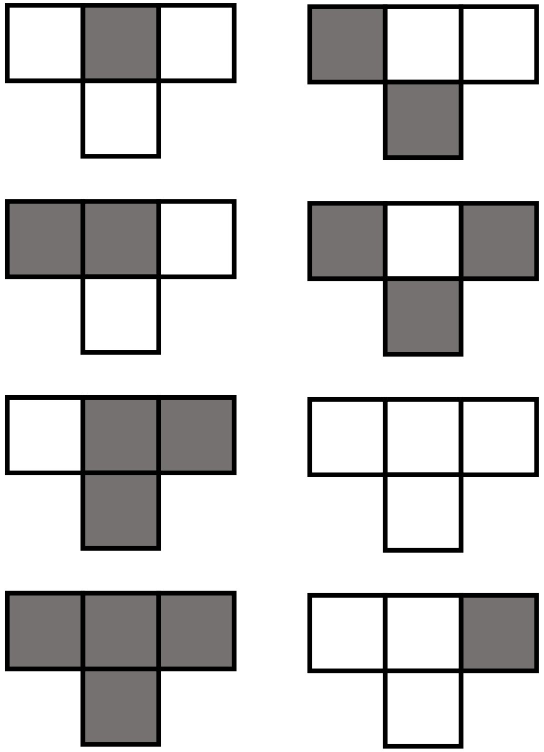
- Evolution for sub-configurations

Space time diagram



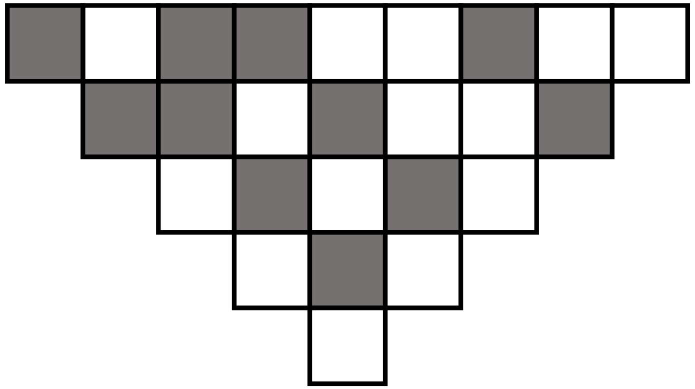
←State

Evolution Rules



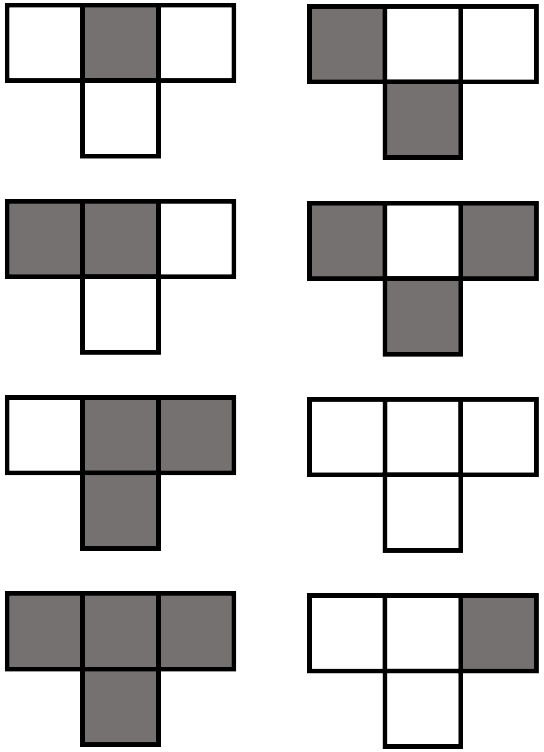
- Evolution for sub-configurations

Space time diagram



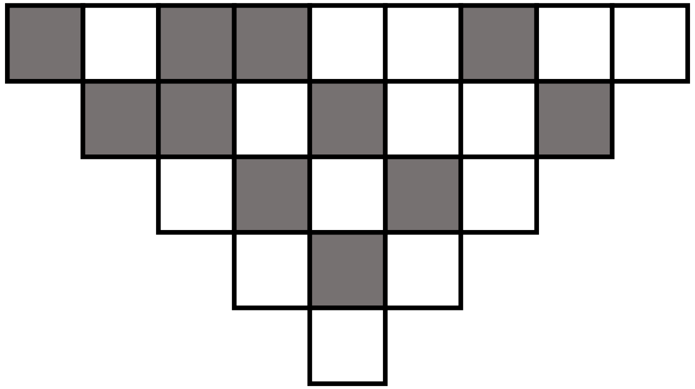
←State

Evolution Rules

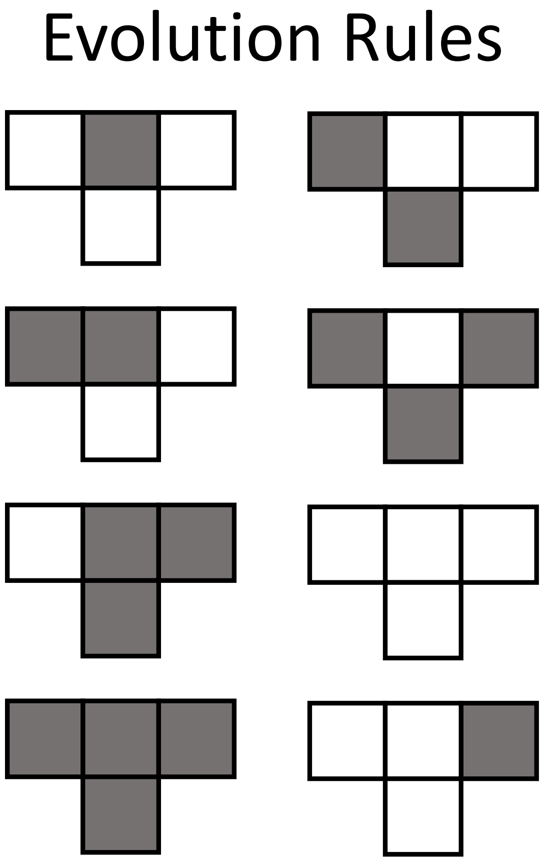


- Evolution for sub-configurations
- Rules extended to sub-configurations !

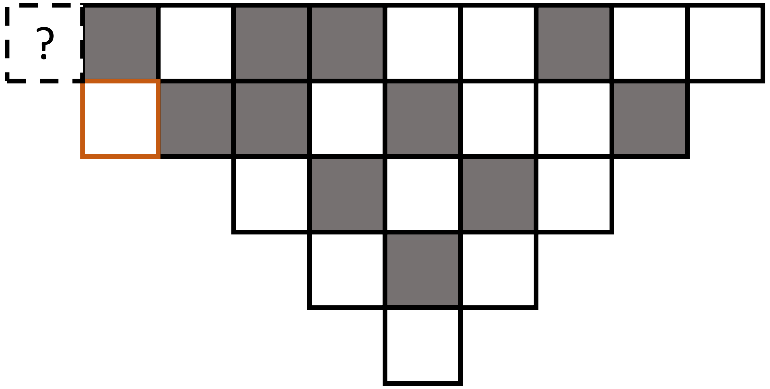
Space time diagram



- We can deduce more !

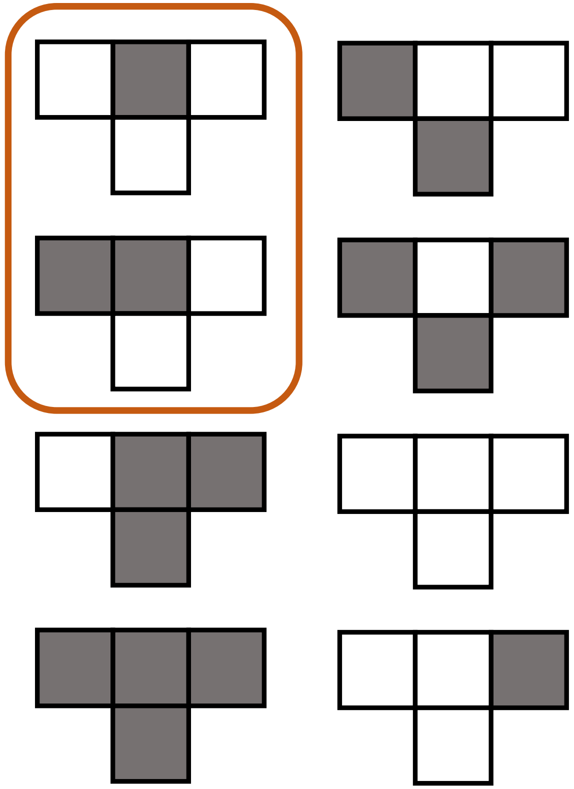


Space time diagram

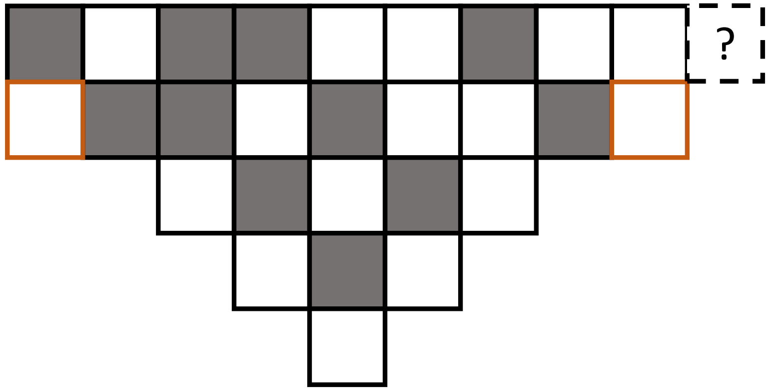


- We can deduce more !

Evolution Rules

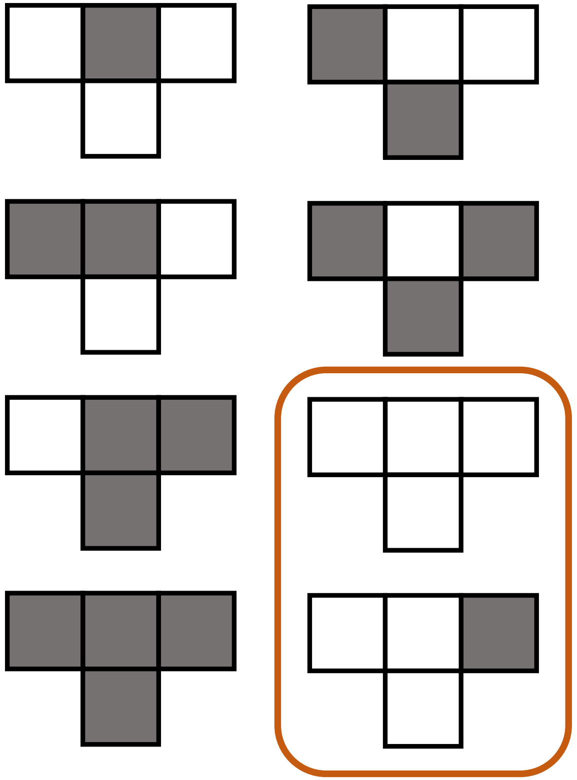


Space time diagram

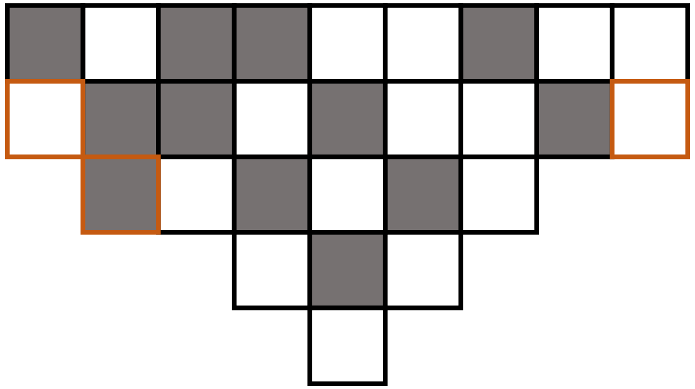


- We can deduce more !

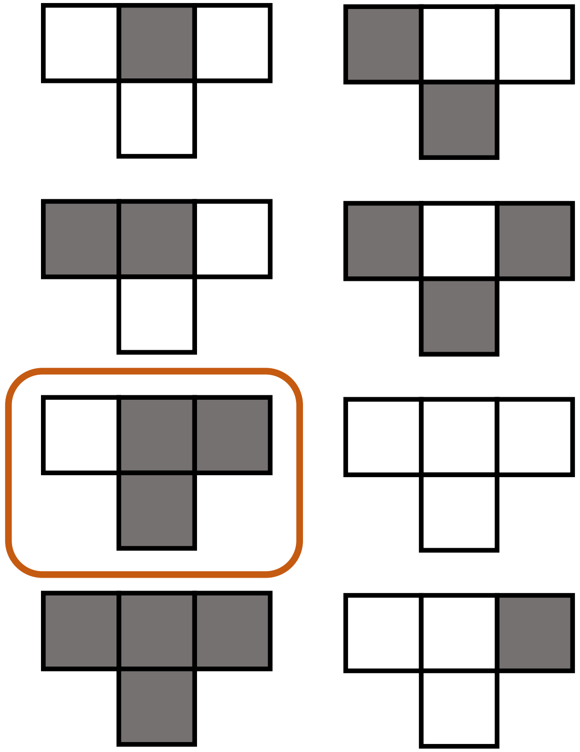
Evolution Rules



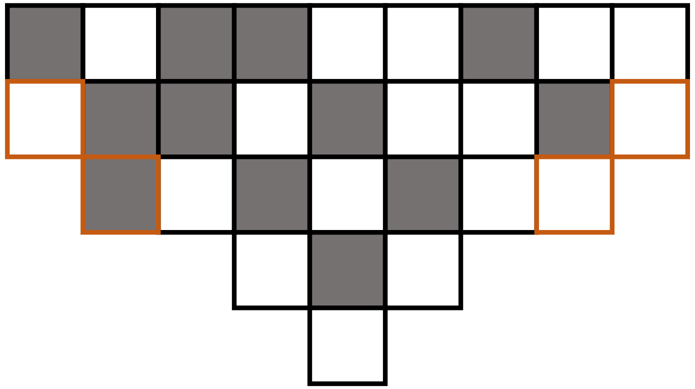
Space time diagram



- We can deduce more !

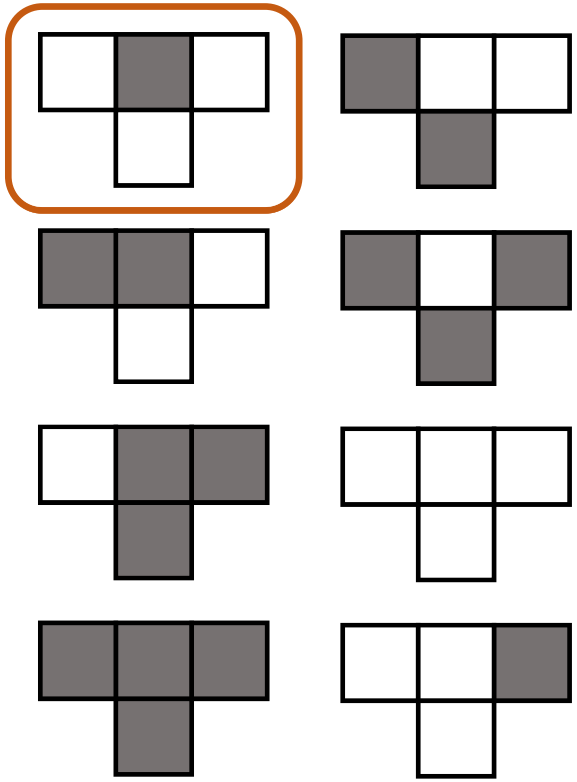


Space time diagram

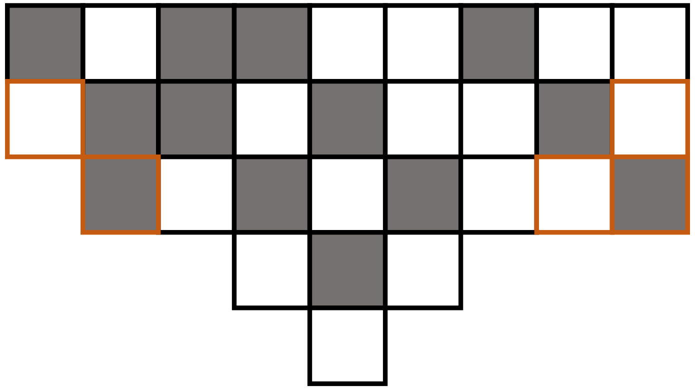


- We can deduce more !

Evolution Rules

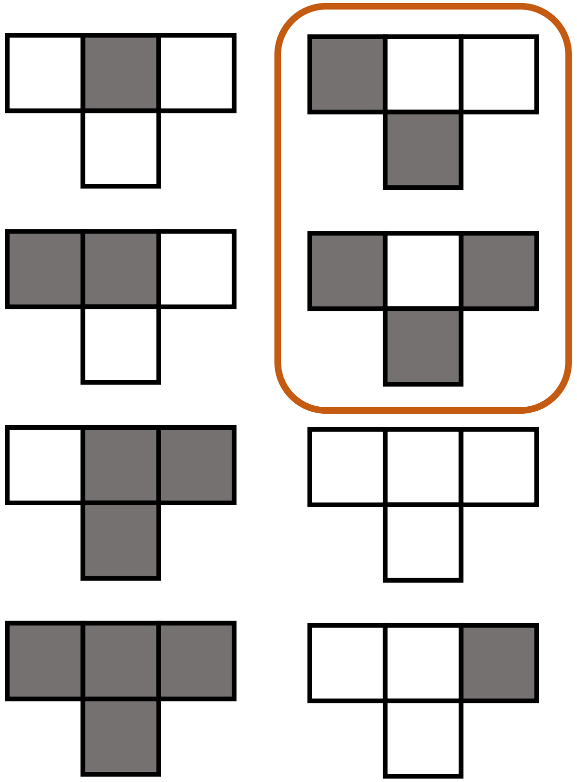


Space time diagram

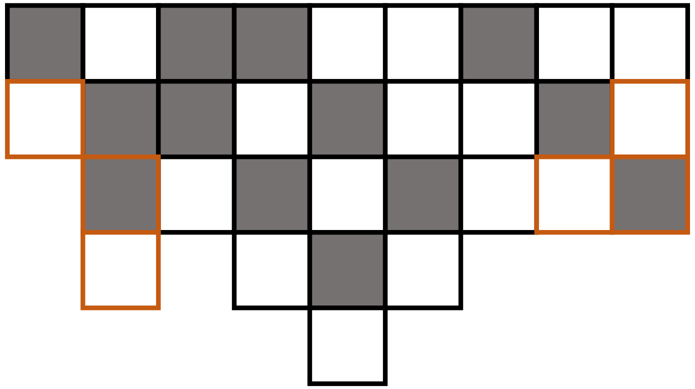


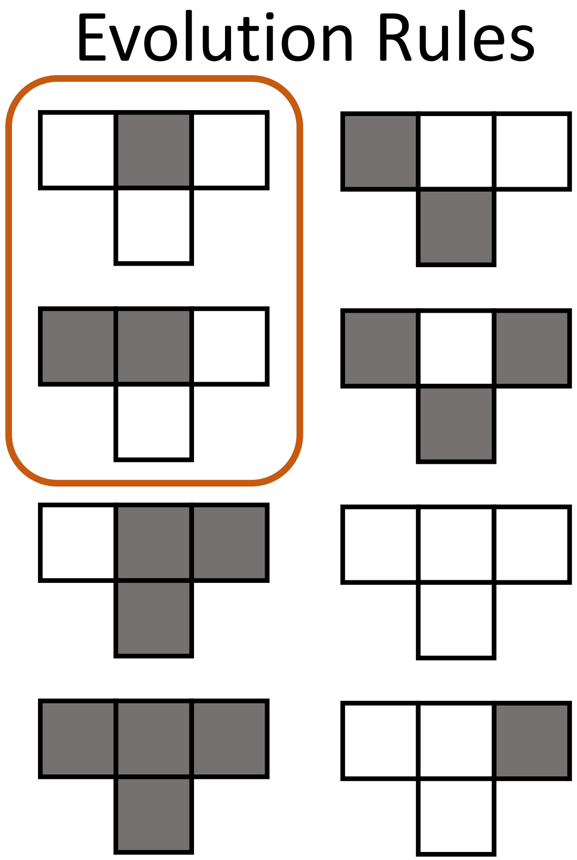
- We can deduce more !

Evolution Rules



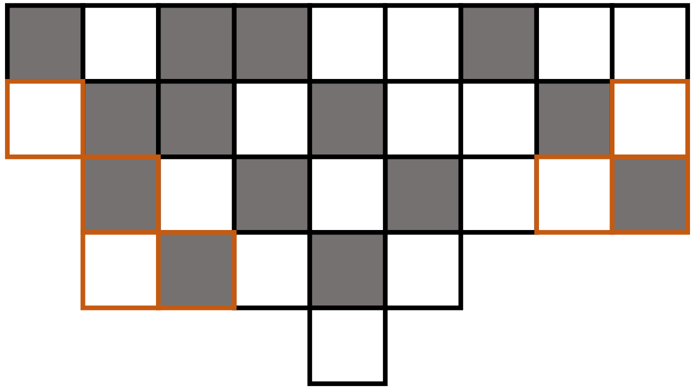
Space time diagram





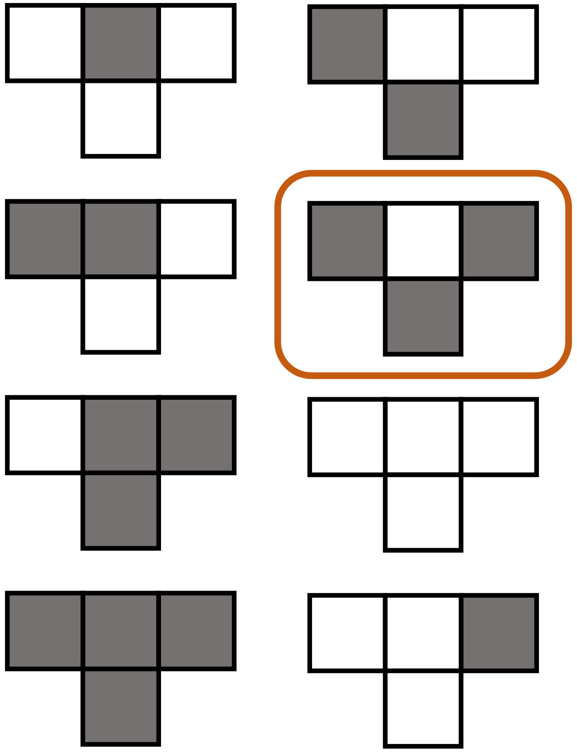
- We can deduce more !

Space time diagram

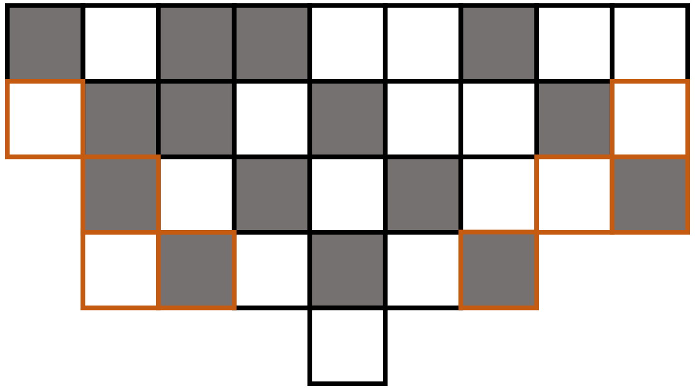


- We can deduce more !

Evolution Rules

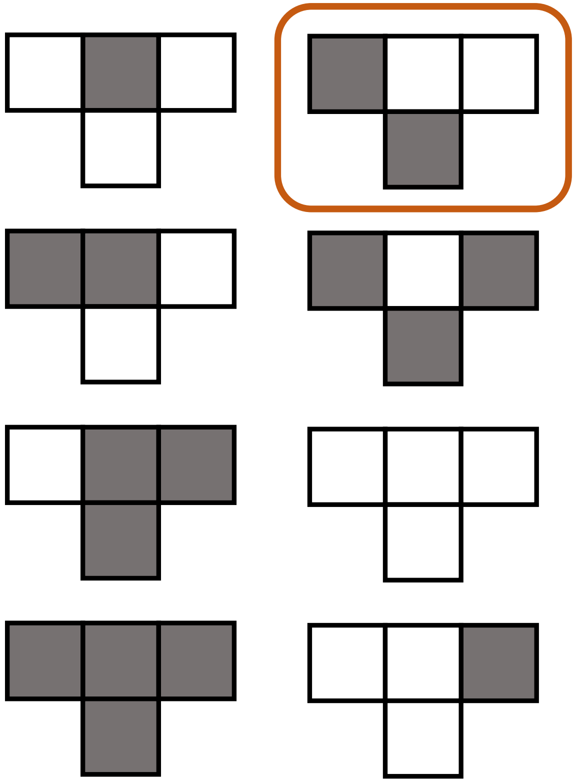


Space time diagram

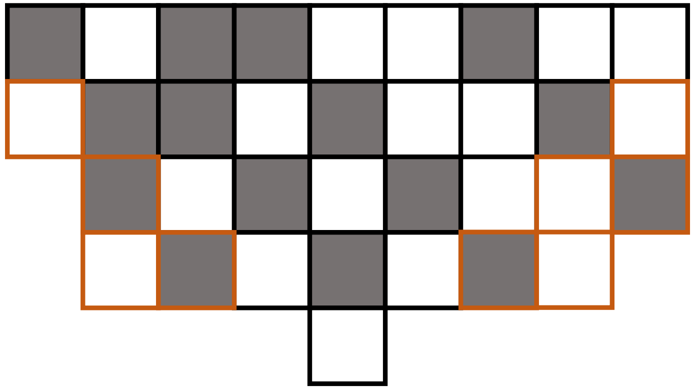


- We can deduce more !

Evolution Rules

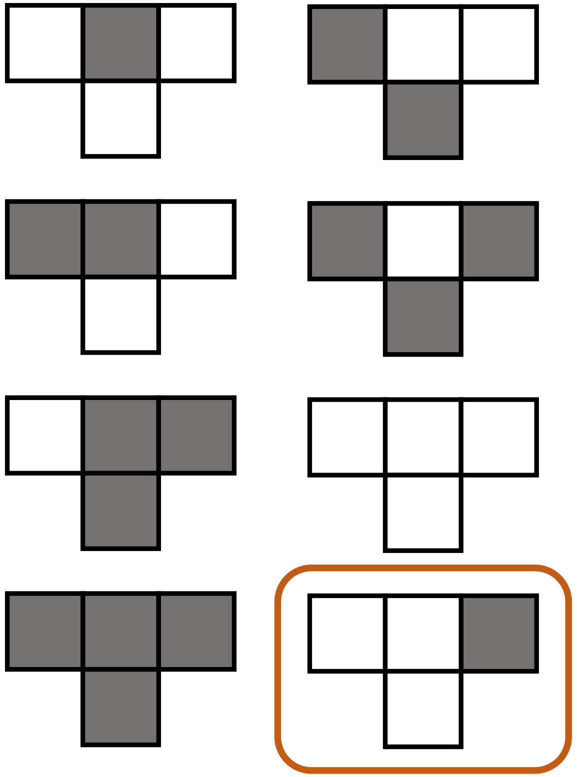


Space time diagram

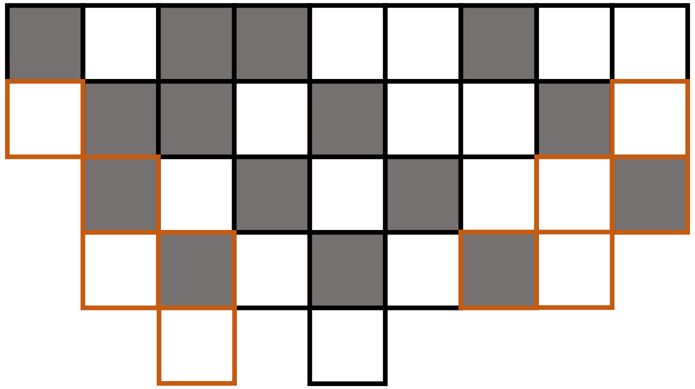


- We can deduce more !

Evolution Rules

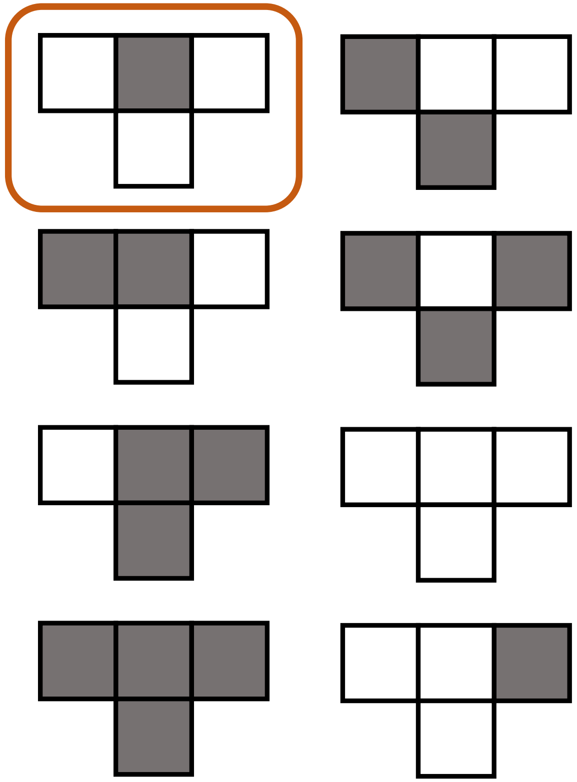


Space time diagram

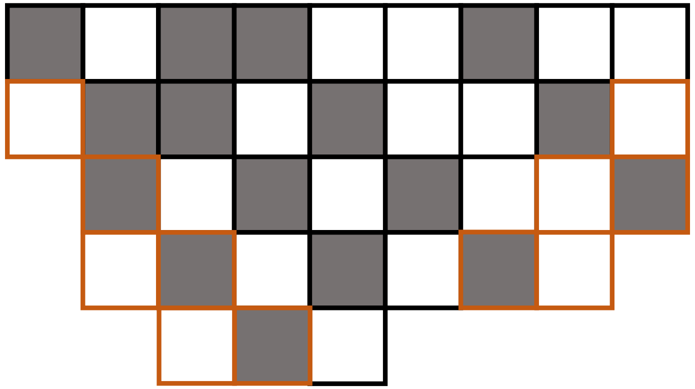


- We can deduce more !

Evolution Rules

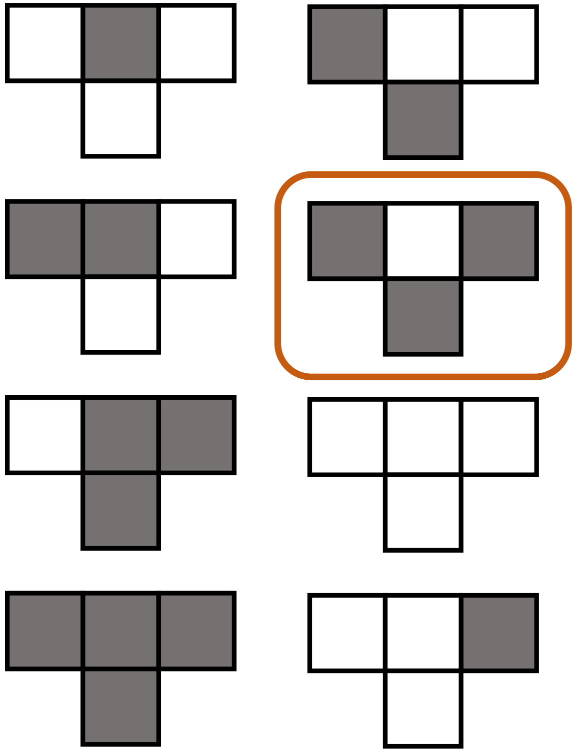


Space time diagram

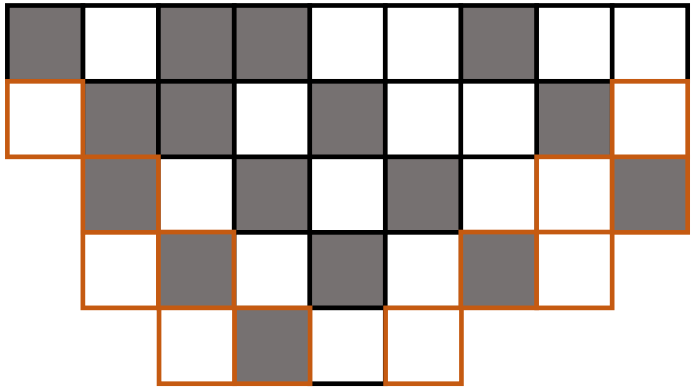


- We can deduce more !

Evolution Rules

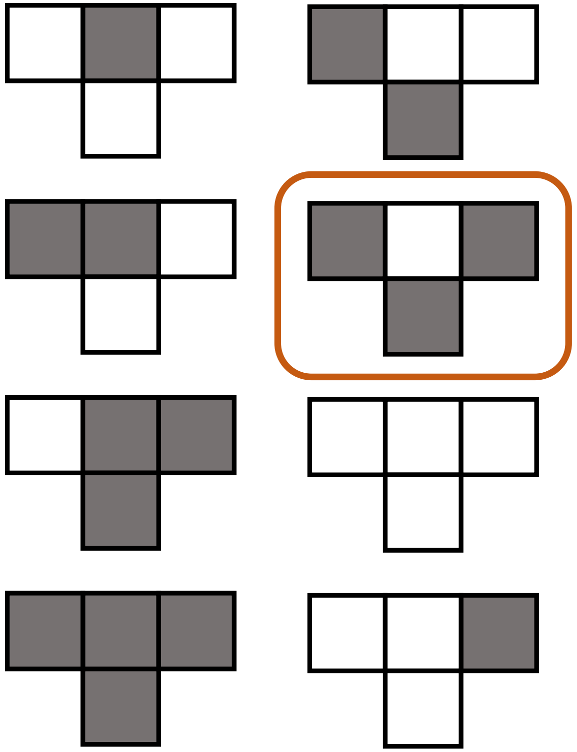


Space time diagram

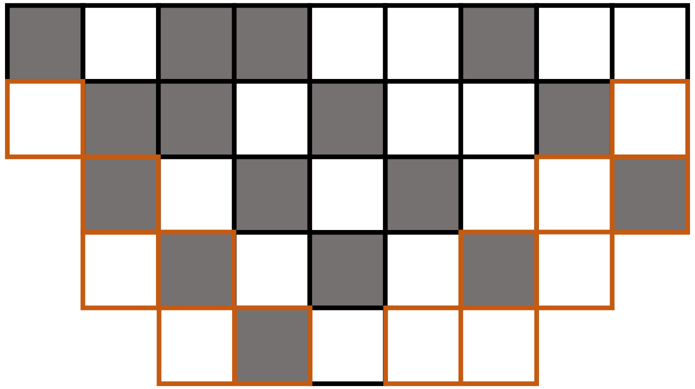


- We can deduce more !

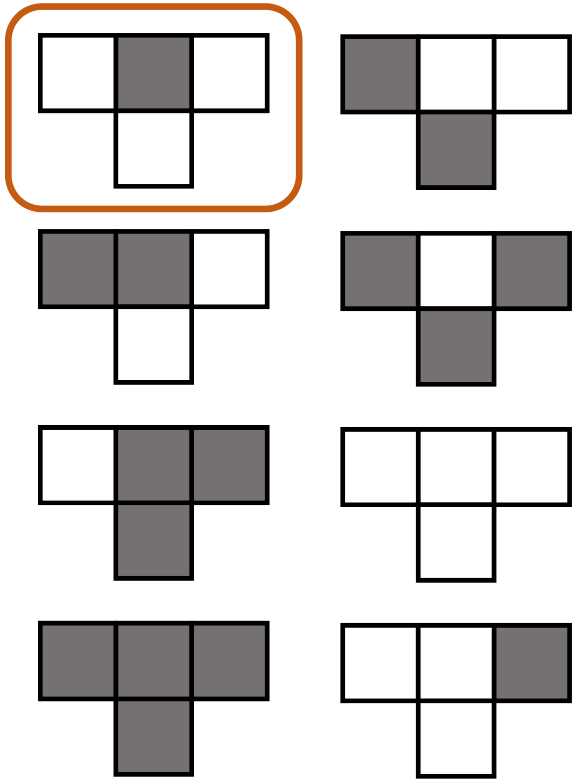
Evolution Rules



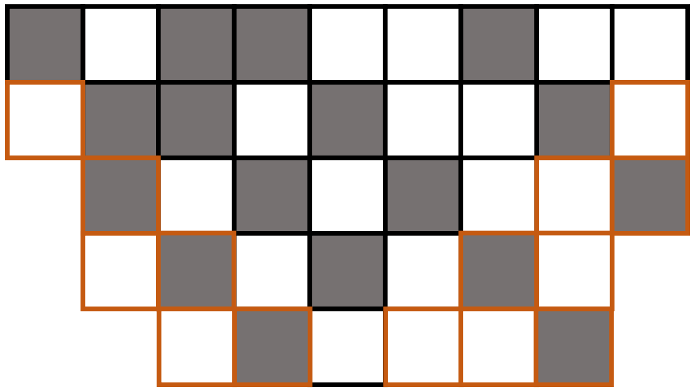
Space time diagram



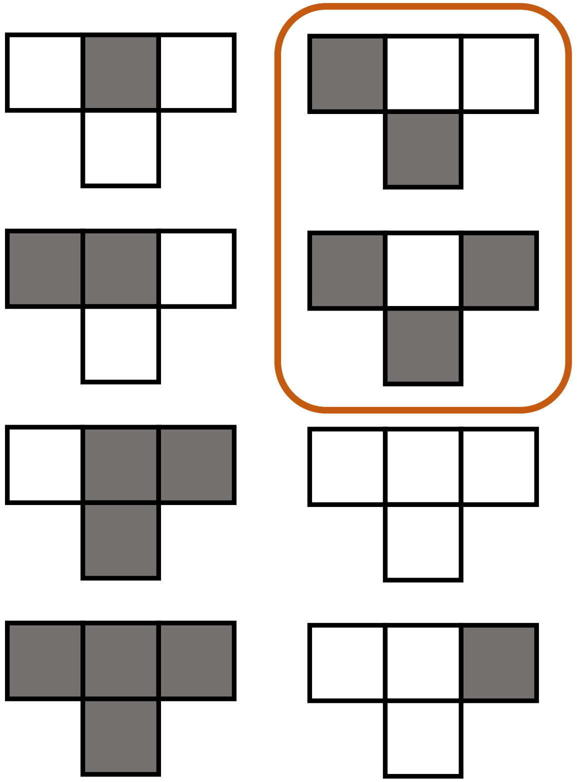
- We can deduce more !



Space time diagram

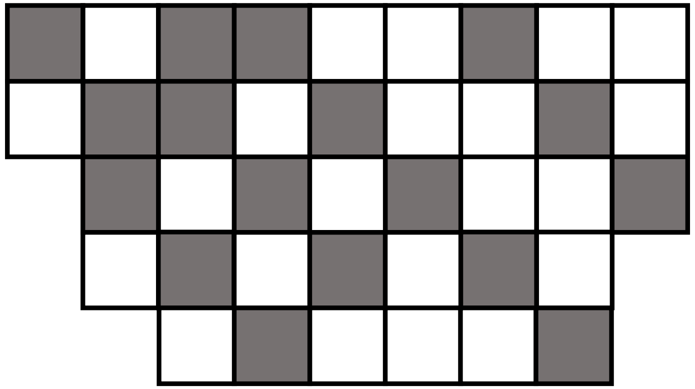


Evolution Rules

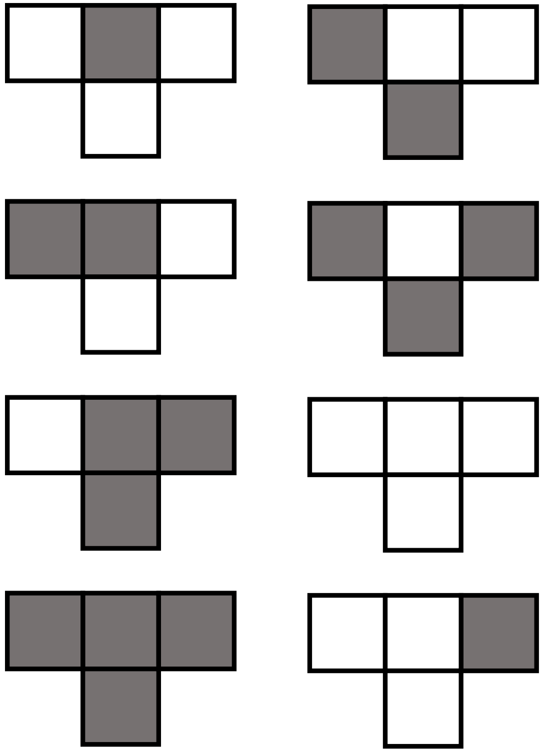


- We can deduce more !

Space time diagram



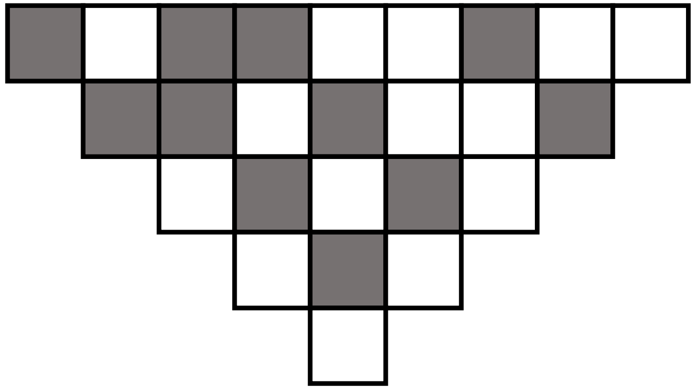
Evolution Rules



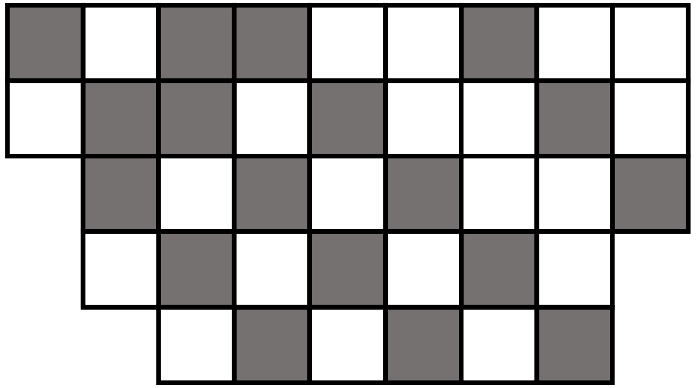
- An other extension to sub-configurations

Two ways of extending

- Coarse transition function $\underline{\Delta}$

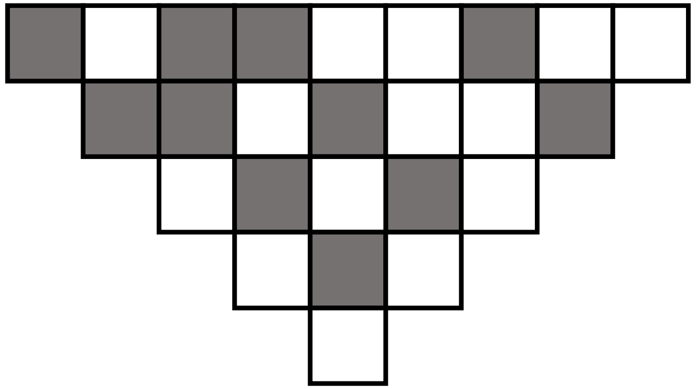


- Fine transition function $\overline{\Delta}$

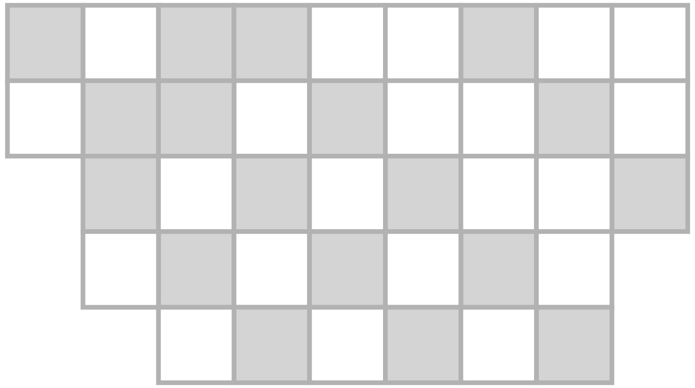


Two ways of extending

- Coarse transition function $\underline{\Delta}$

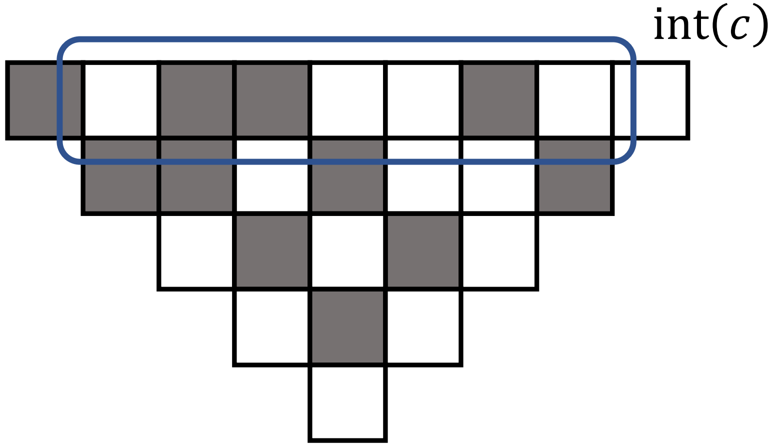


- Fine transition function $\overline{\Delta}$



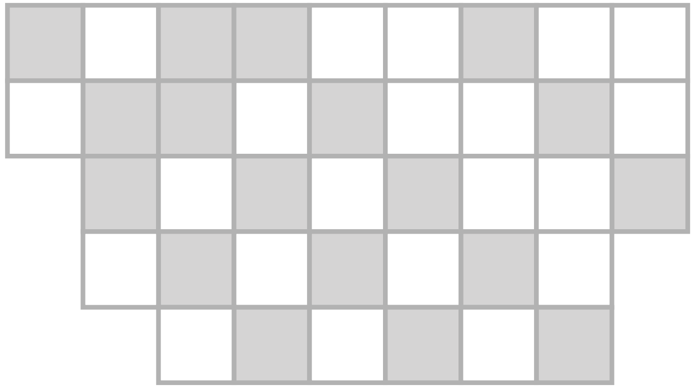
Two ways of extending

- Coarse transition function $\underline{\Delta}$
  - Deduce with full neighborhood



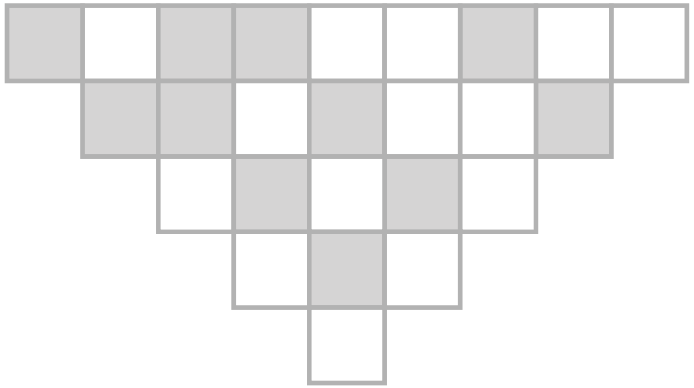
  - Minimal extension

- Fine transition function $\overline{\Delta}$



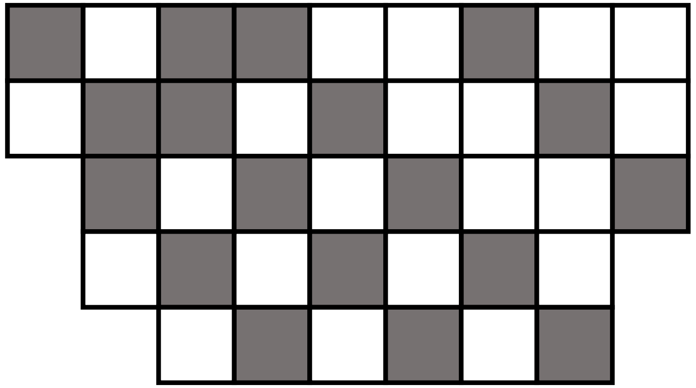
Two ways of extending

- Coarse transition function $\underline{\Delta}$
  - Deduce with full neighborhood



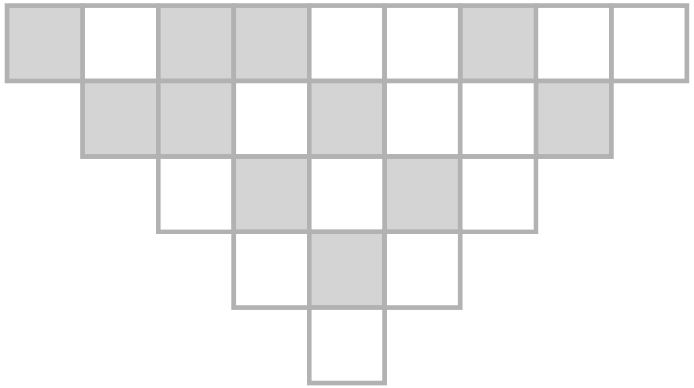
  - Minimal extension

- Fine transition function $\overline{\Delta}$



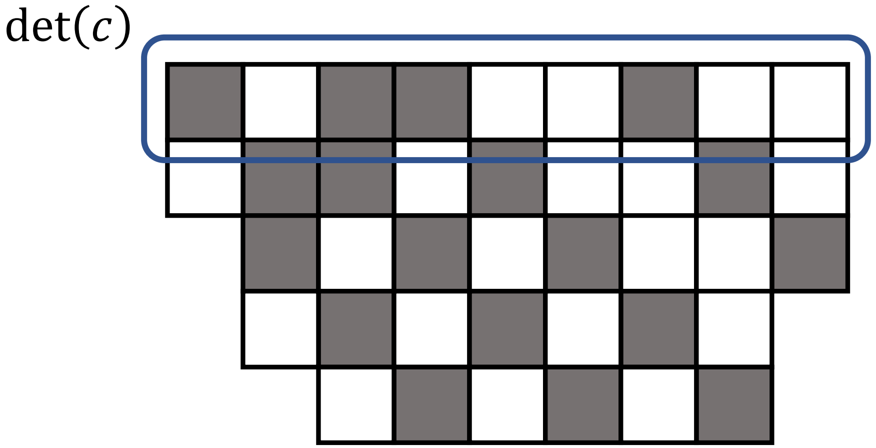
Two ways of extending

- Coarse transition function $\underline{\Delta}$
  - Deduce with full neighborhood



  - Minimal extension

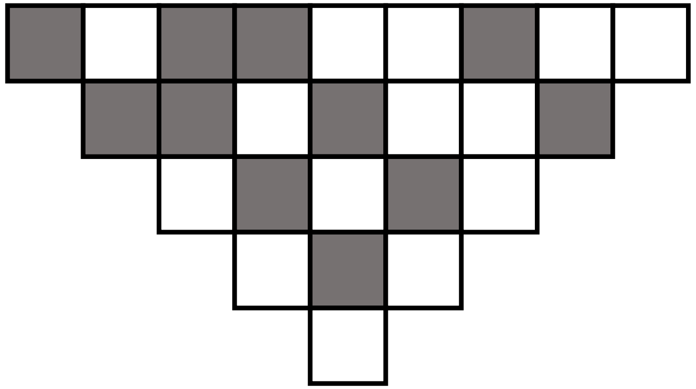
- Fine transition function $\overline{\Delta}$
  - Deduce without neighborhood



  - Maximal extension

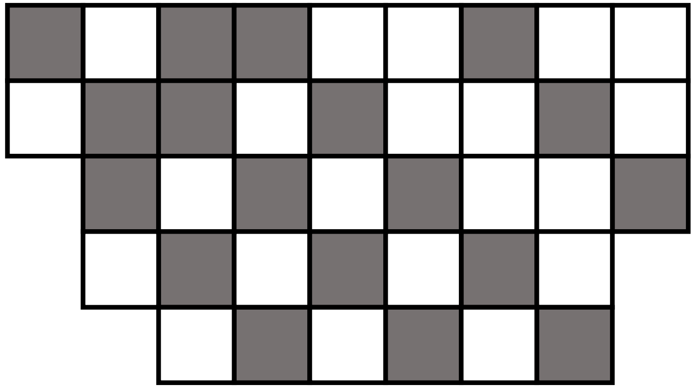
Two ways of extending

- Coarse transition function $\underline{\Delta}$
  - Deduce with full neighborhood



- Is a left Kan extension !

- Fine transition function $\overline{\Delta}$
  - Deduce without neighborhood



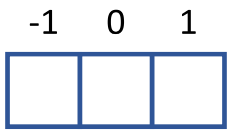
- Is a right Kan extension !

Outline

- Two ways to extend with an example
- The poset of sub-configurations
- The two extensions are kan extensions

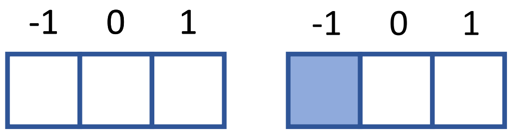
Local configurations

- Local configuration maps $l\colon N \to Q$



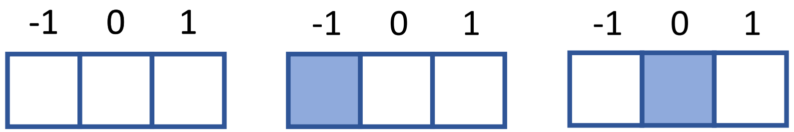
Local configurations

- Local configuration maps $l: N \to Q$



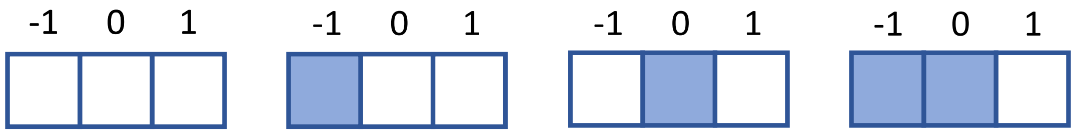
Local configurations

- Local configuration maps $l\colon N \to Q$



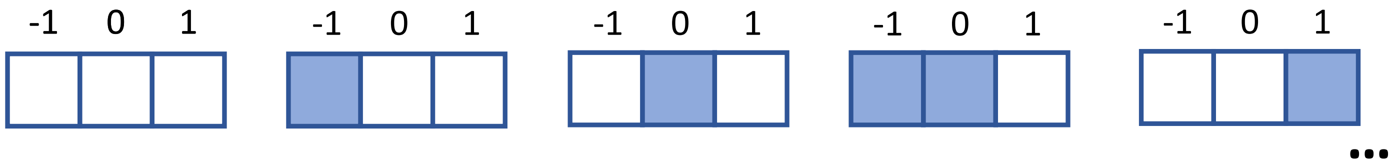
Local configurations

- Local configuration maps $l: N \to Q$



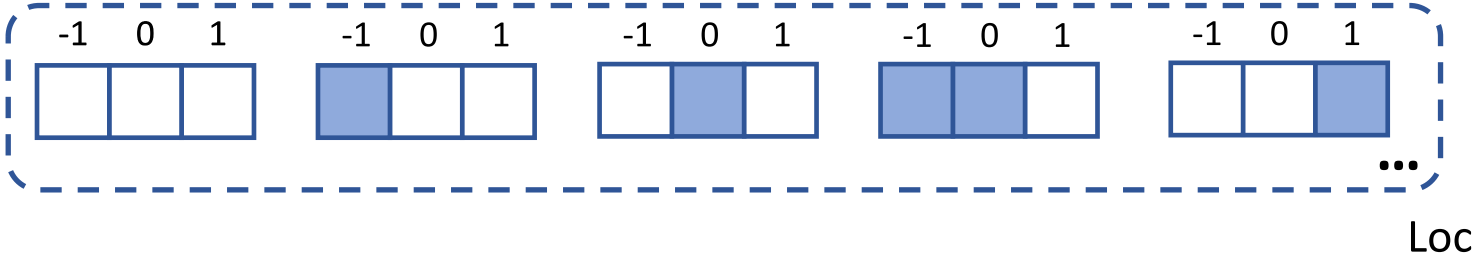
Local configurations

- Local configuration maps $l: N \rightarrow Q$



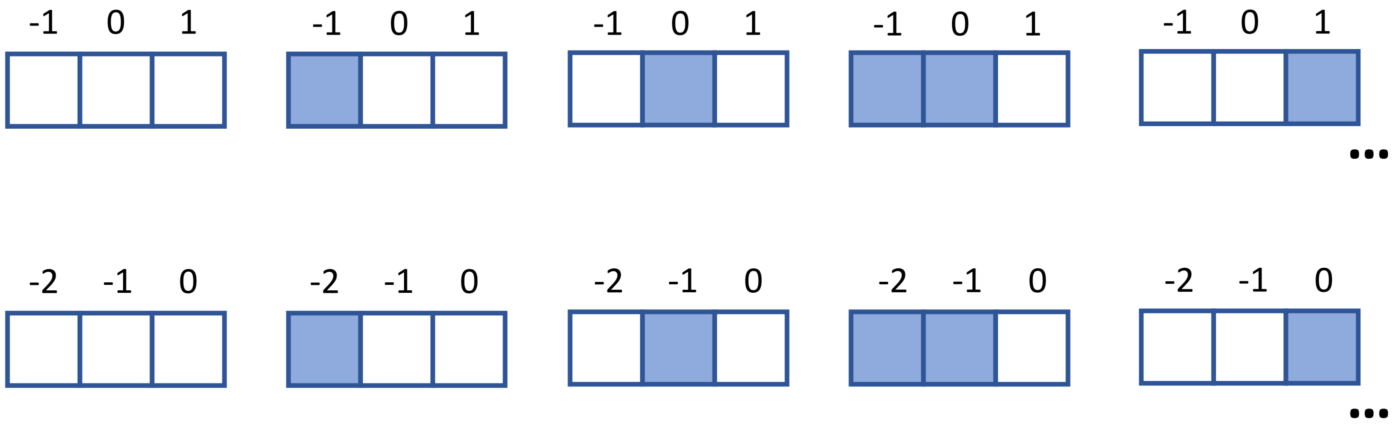
Set of local configurations

- Set of local configuration: $\mathrm{Loc} \coloneqq Q^N$



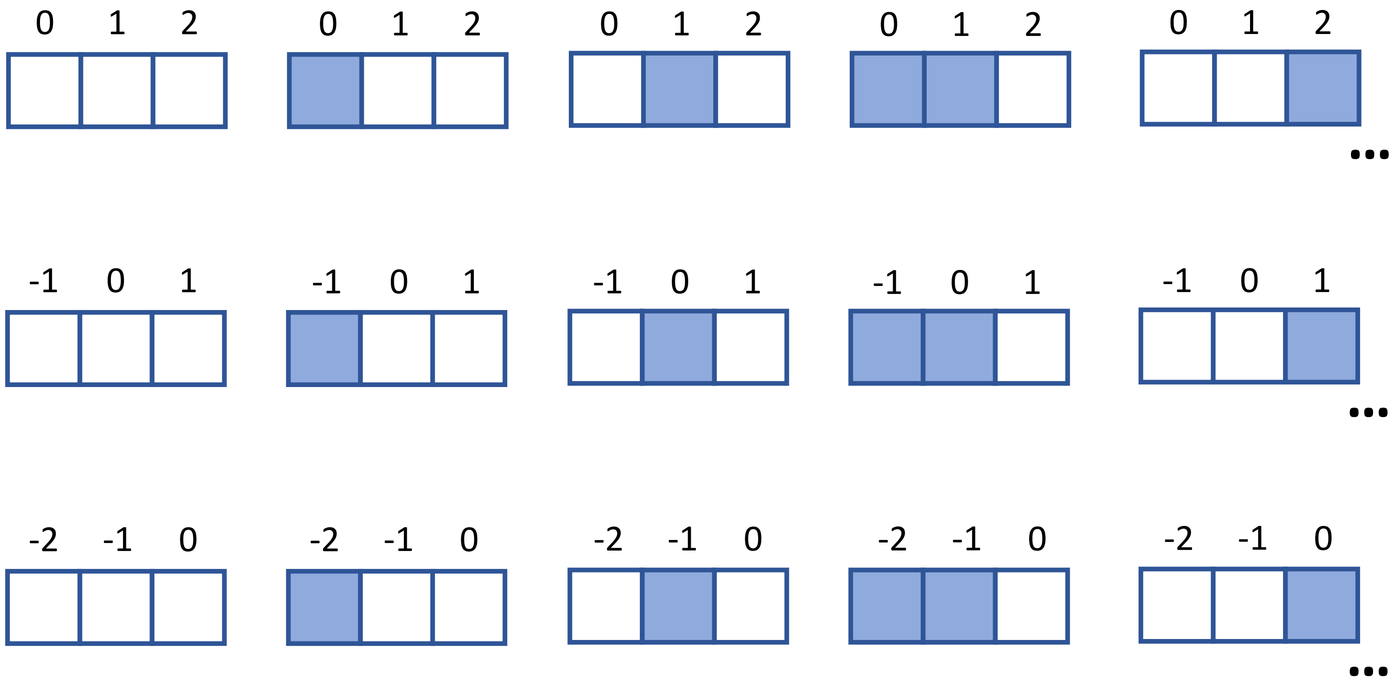
Shifted local configurations

- Shifted local configuration, maps $l: p + N \to Q$

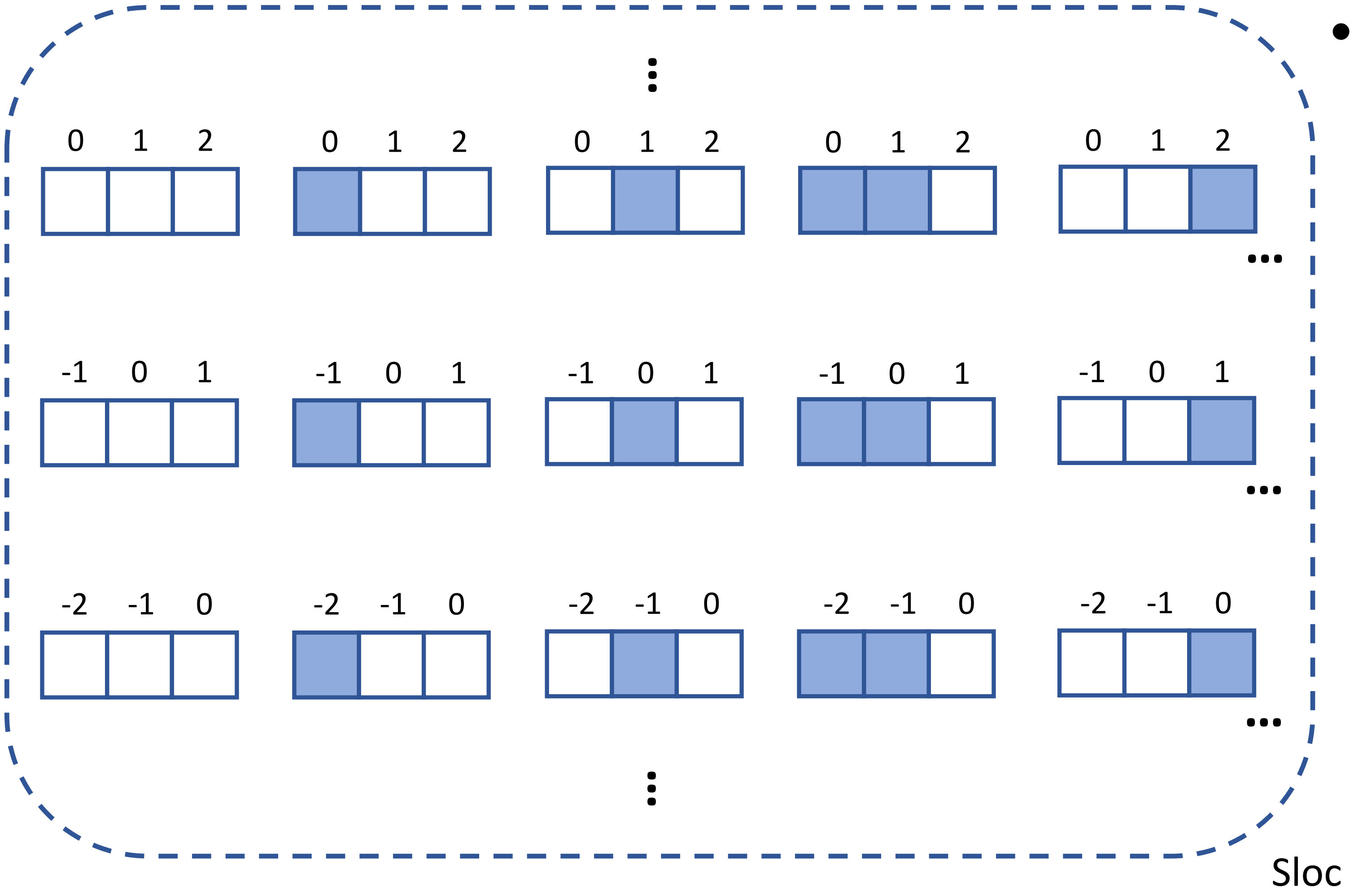


Shifted local configurations

- Shifted local configuration, maps $l\colon p + N \to Q$



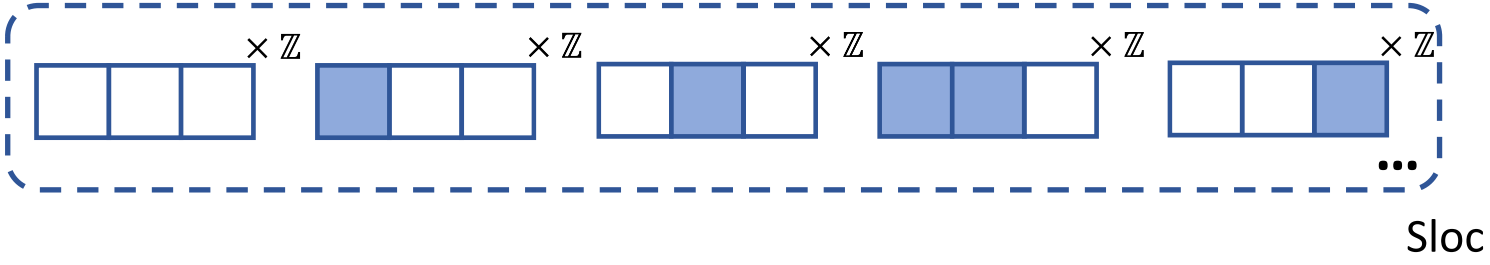
Set of shifted local configurations



- Shifted local configurations :

$$Sloc \coloneqq \bigcup_{p \in \mathbb{Z}} p + N \rightarrow Q$$

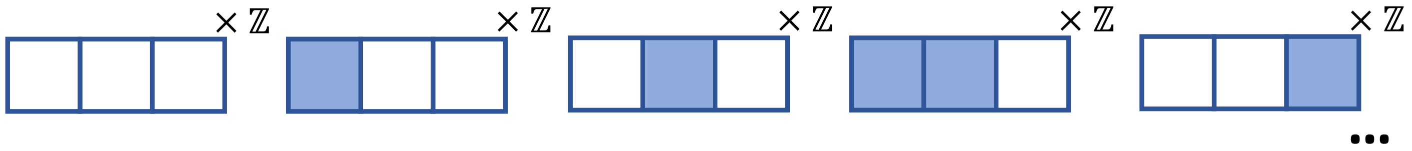
Set of shifted local configurations



- Shifted local configurations :

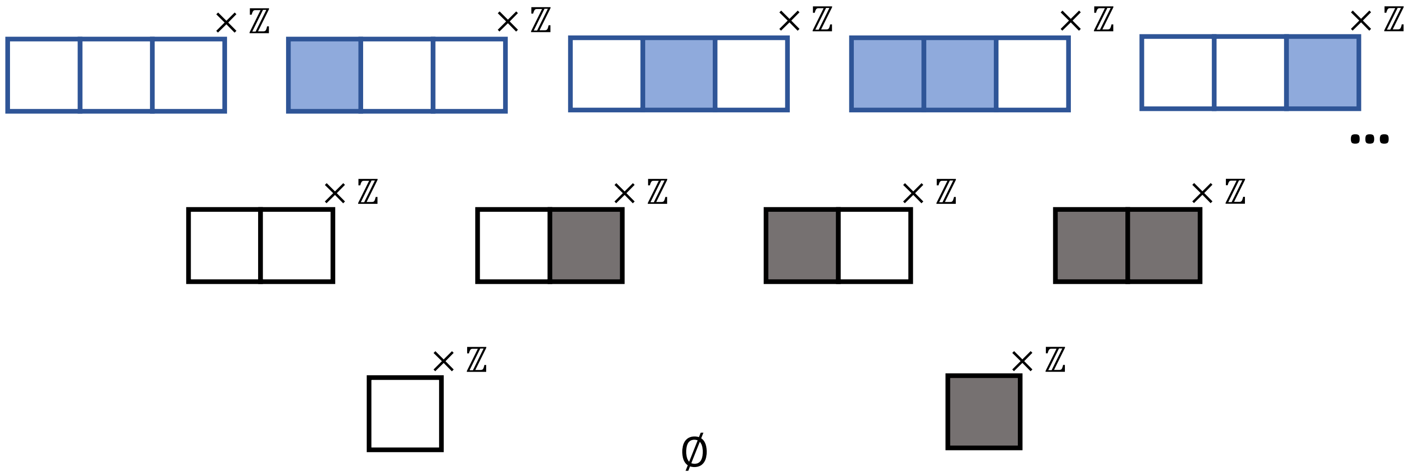
$$Sloc \coloneqq \bigcup_{p \in \mathbb{Z}} p + N \to Q$$

Set of sub-configurations



- Shifted local configurations

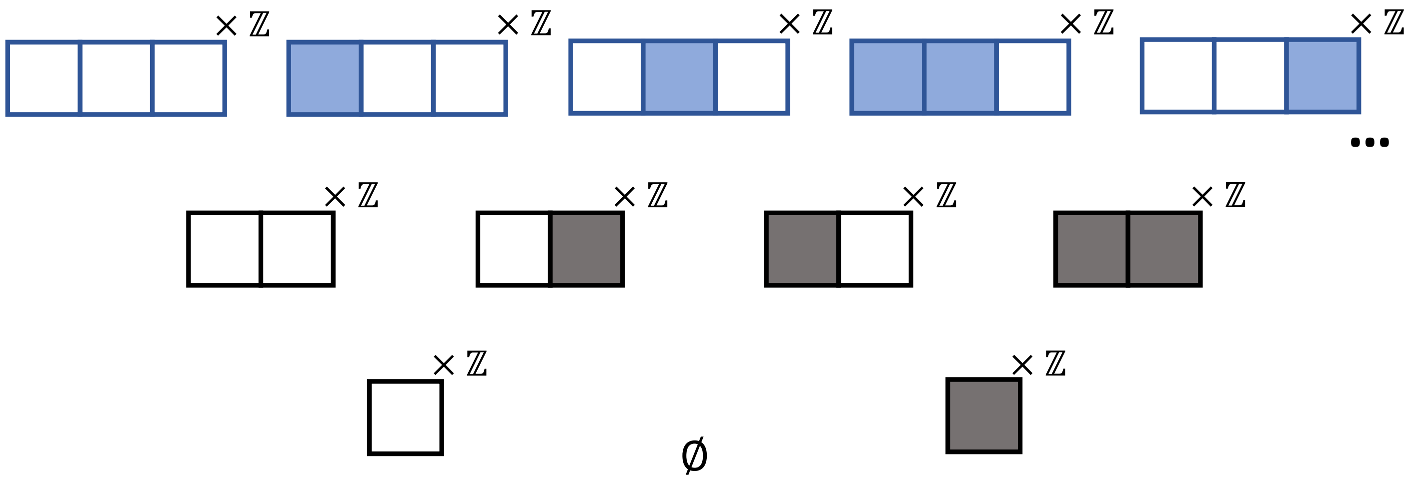
Set of sub-configurations



- Shifted local configurations
- Shifted sub-local configurations

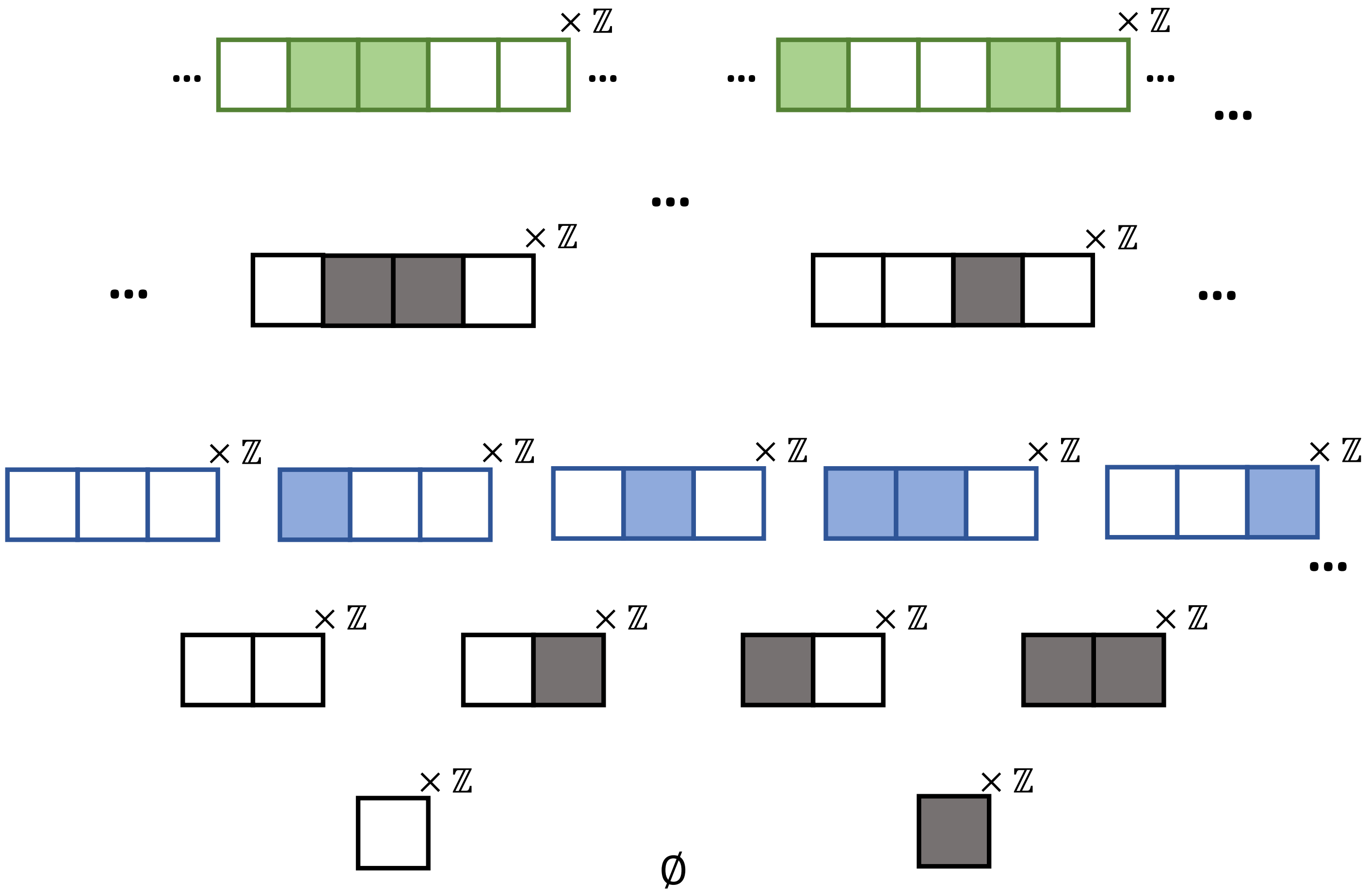
Set of sub-configurations

- Configurations



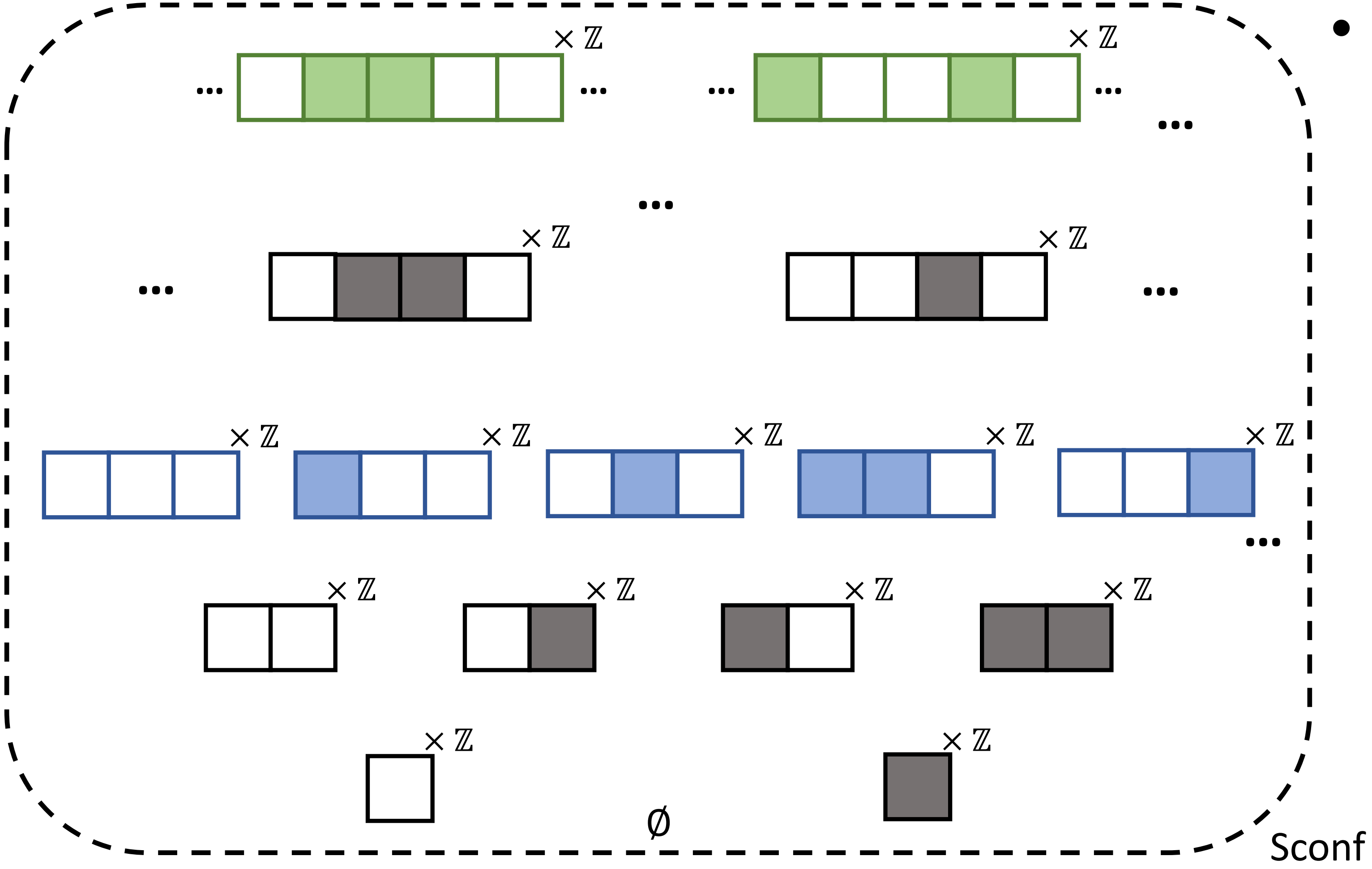
- Shifted local configurations
- Shifted sub-local configurations

Set of sub-configurations



- Configurations
- Shifted local configurations
- Shifted sub-local configurations

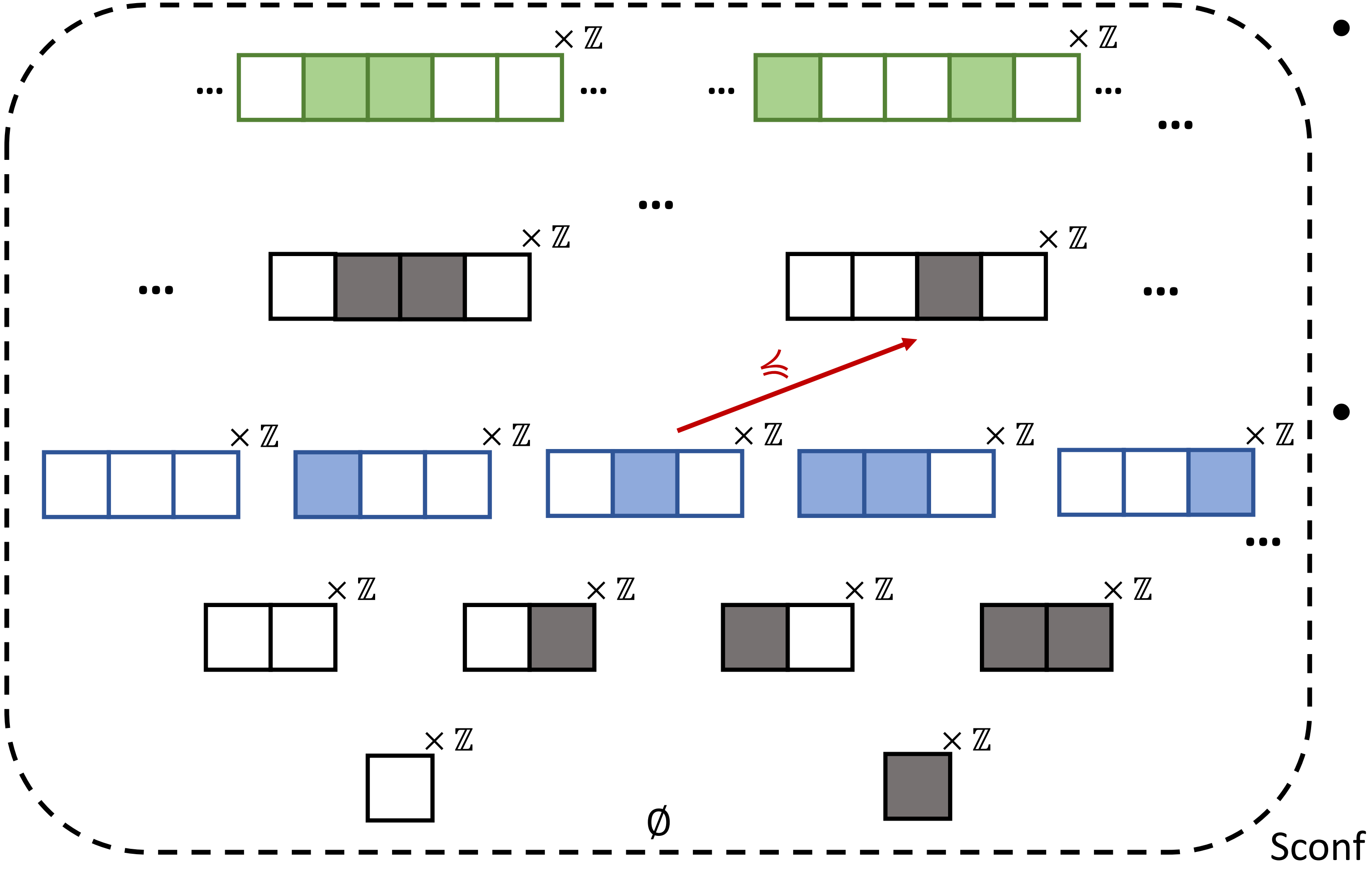
Set of sub-configurations



- Sub configurations :

$$Pconf \coloneqq \bigcup_{S \subseteq \mathbb{Z}} Q^S$$

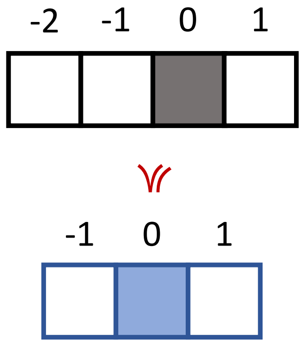
Poset of sub-configurations



- Partial configurations :

$$Pconf \coloneqq \bigcup_{S \subseteq \mathbb{Z}} Q^S$$

- Partial order relation :

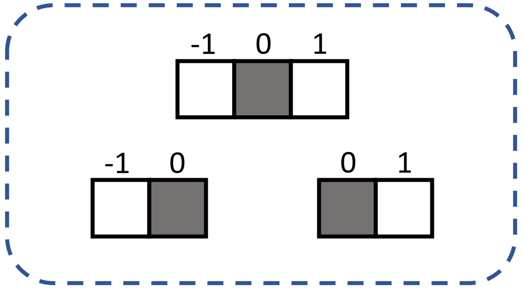


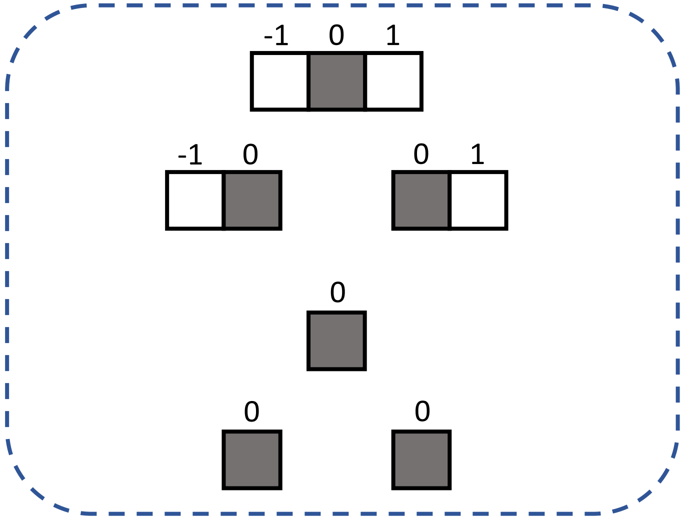
Simpler category theory

- Simple frame of work…
- 2-category of posets

Simpler category theory

- Simple frame of work…

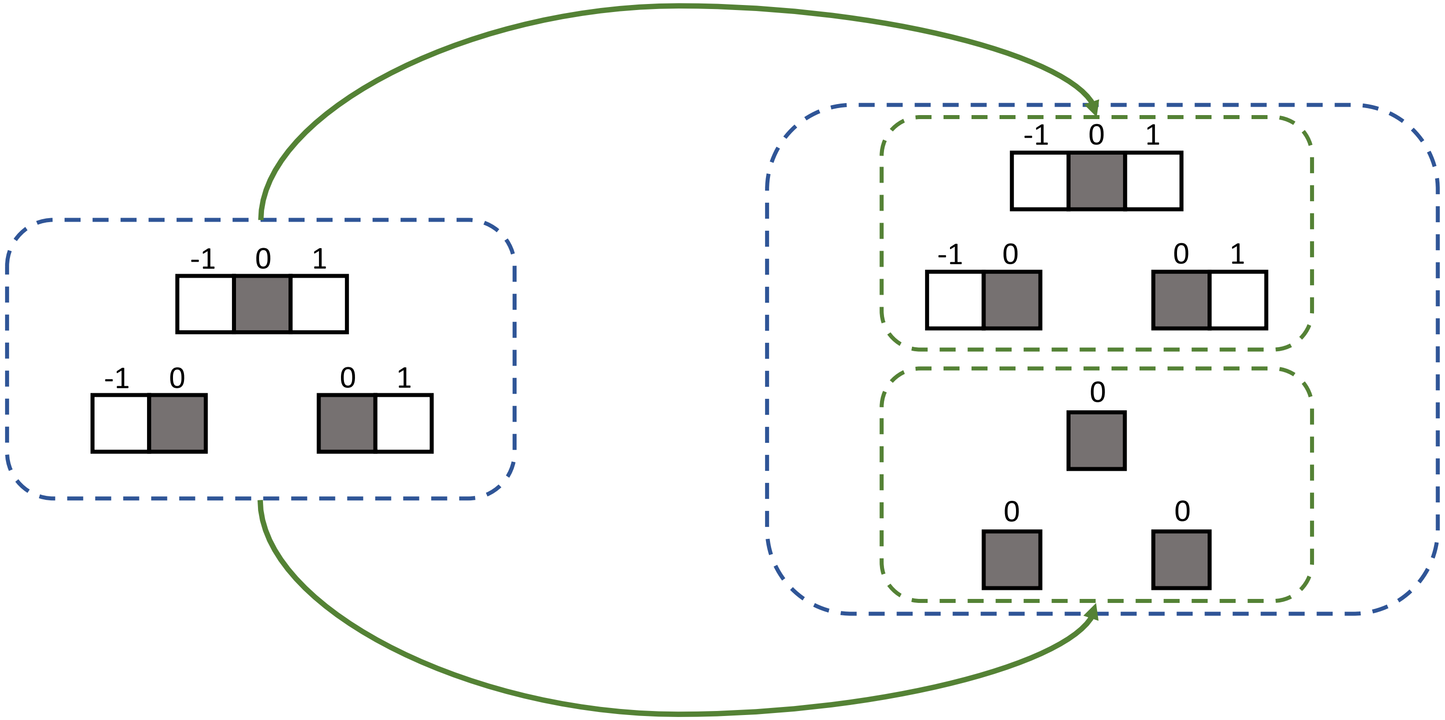




- 2-category of posets :
  - Categories: Posets

Simpler category theory

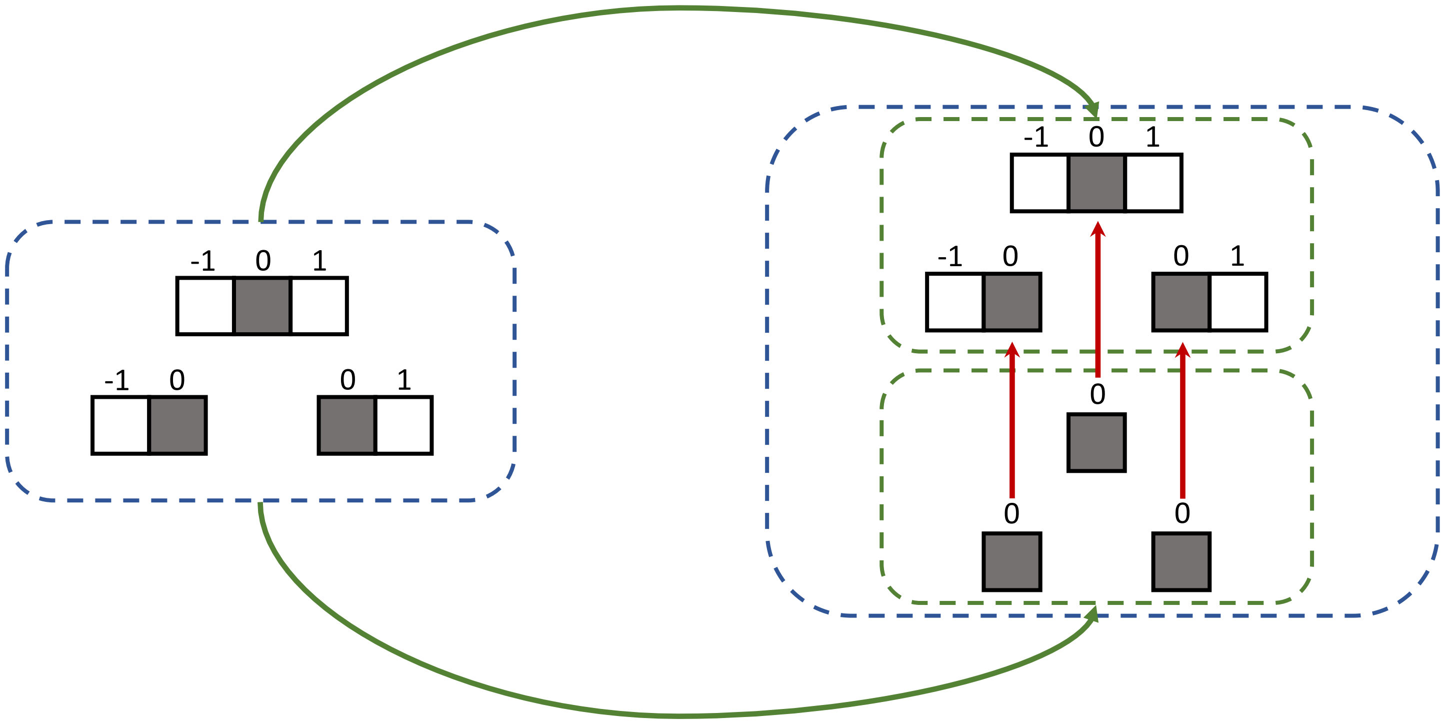
- Simple frame of work…



- 2-category of posets :
  - Categories: Posets
  - Functors: monotonous functions

Simpler category theory

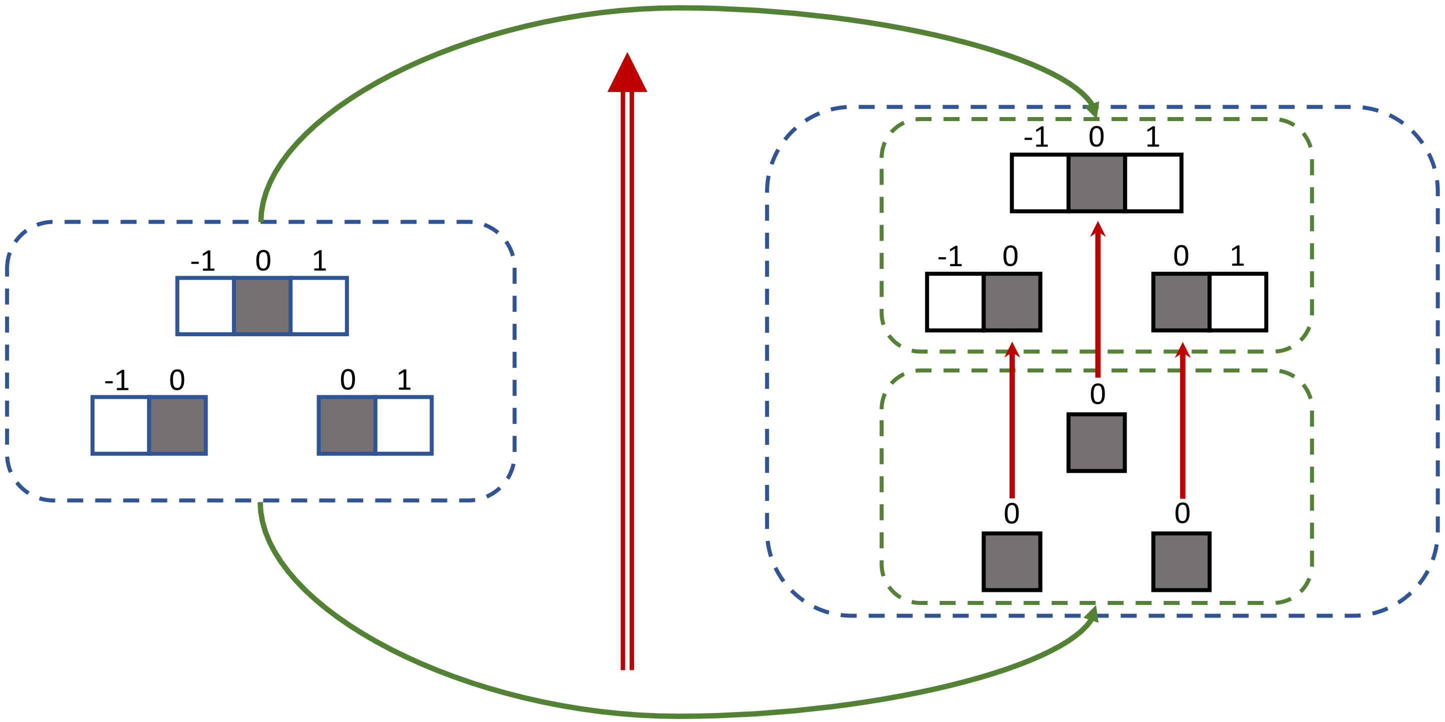
- Simple frame of work…



- 2-category of posets :
  - Categories: Posets
  - Functors: monotonous functions
  - Natural transformations: partial order over functors

Simpler category theory

- Simple frame of work…



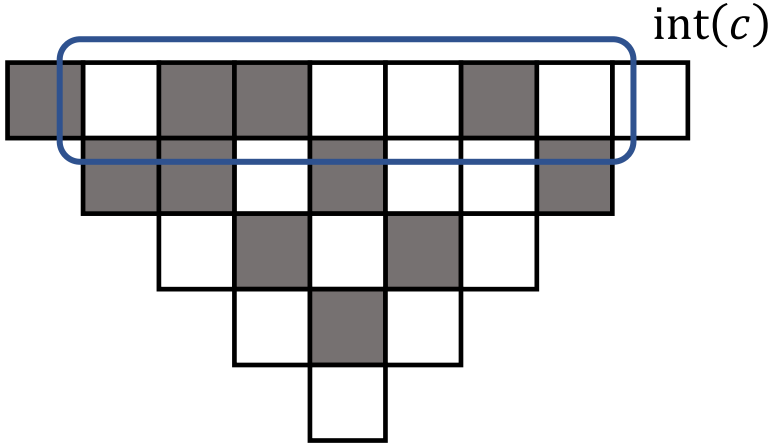
- 2-category of posets :
  - Categories: Posets
  - Functors: monotonous functions
  - Natural transformations: partial order over functors

Outline

- Two ways to extend with an example
- The poset of sub-configurations
- The two extensions are kan extensions

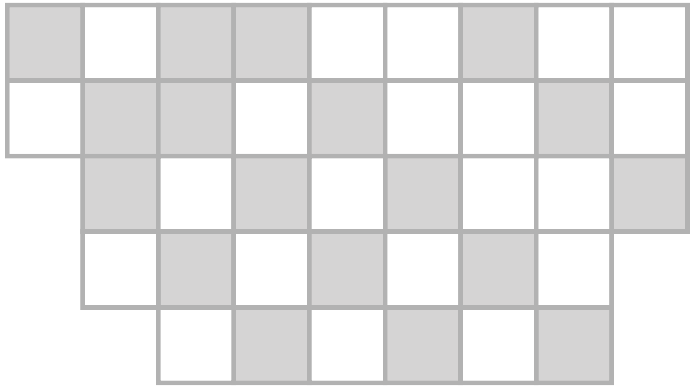
Two way of extending

- Coarse transition function $\underline{\Delta}$
  - Deduce with full neighborhood



- Is a left Kan extension !

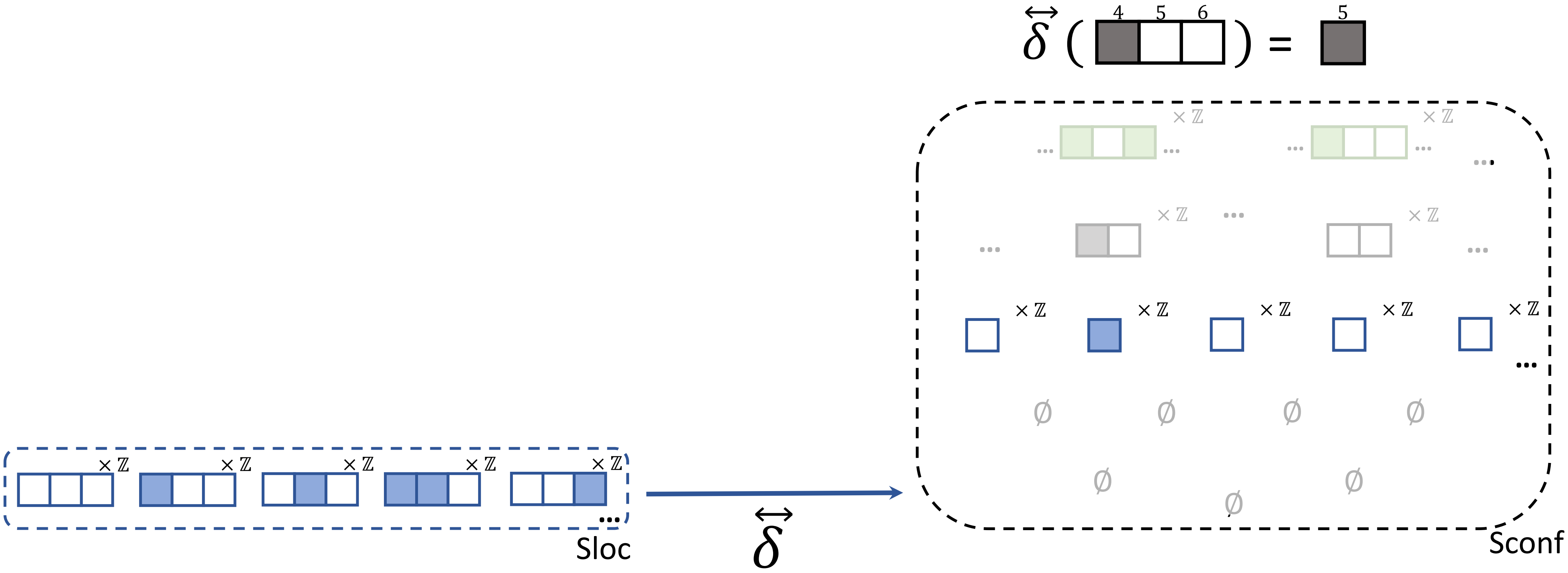
- Fine transition function $\overline{\Delta}$
  - Deduce without neighborhood



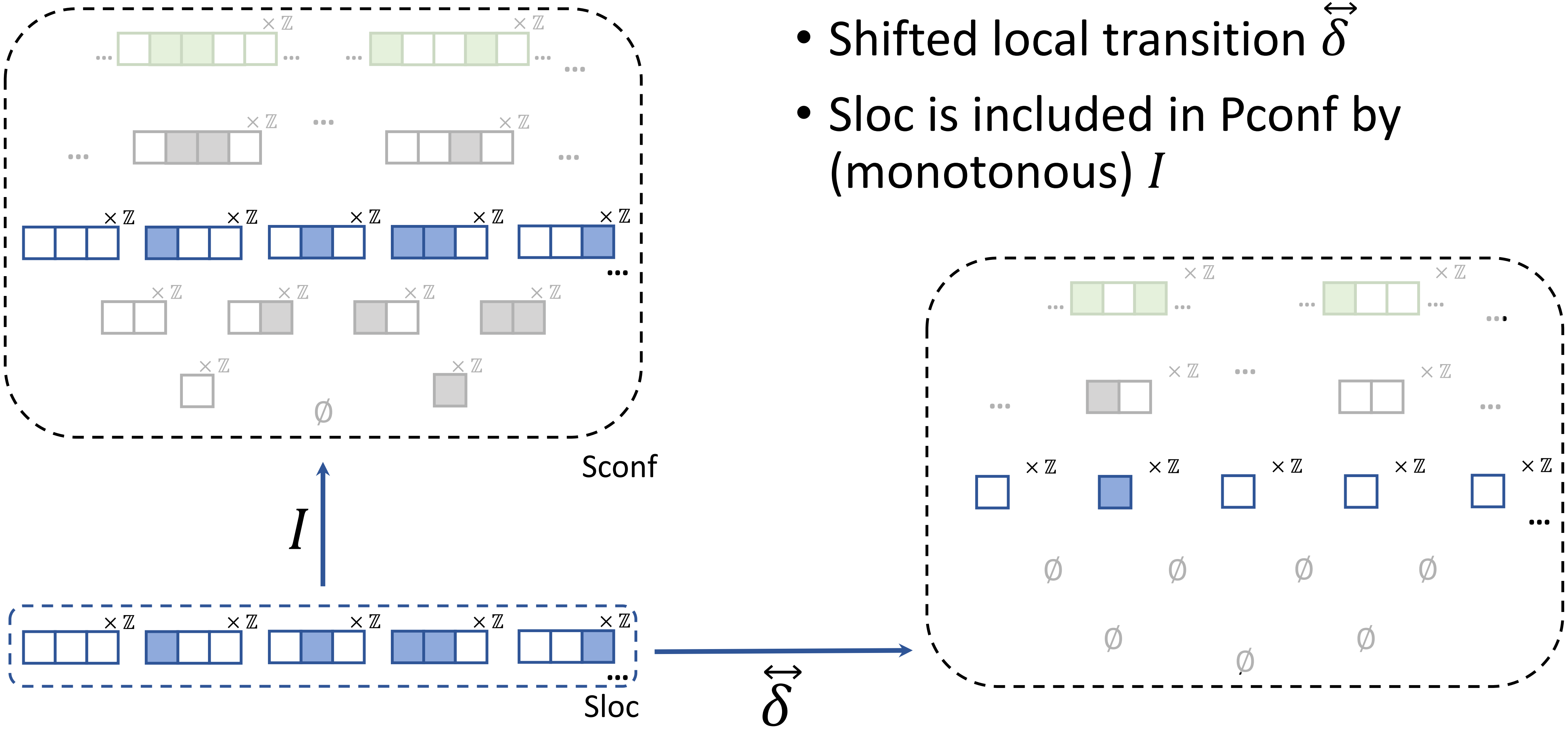
- Is a right Kan extension !

Coarse as left Kan extension

- Shifted local transition $\overleftrightarrow{\delta}$:
  - Applies local rules over Sloc



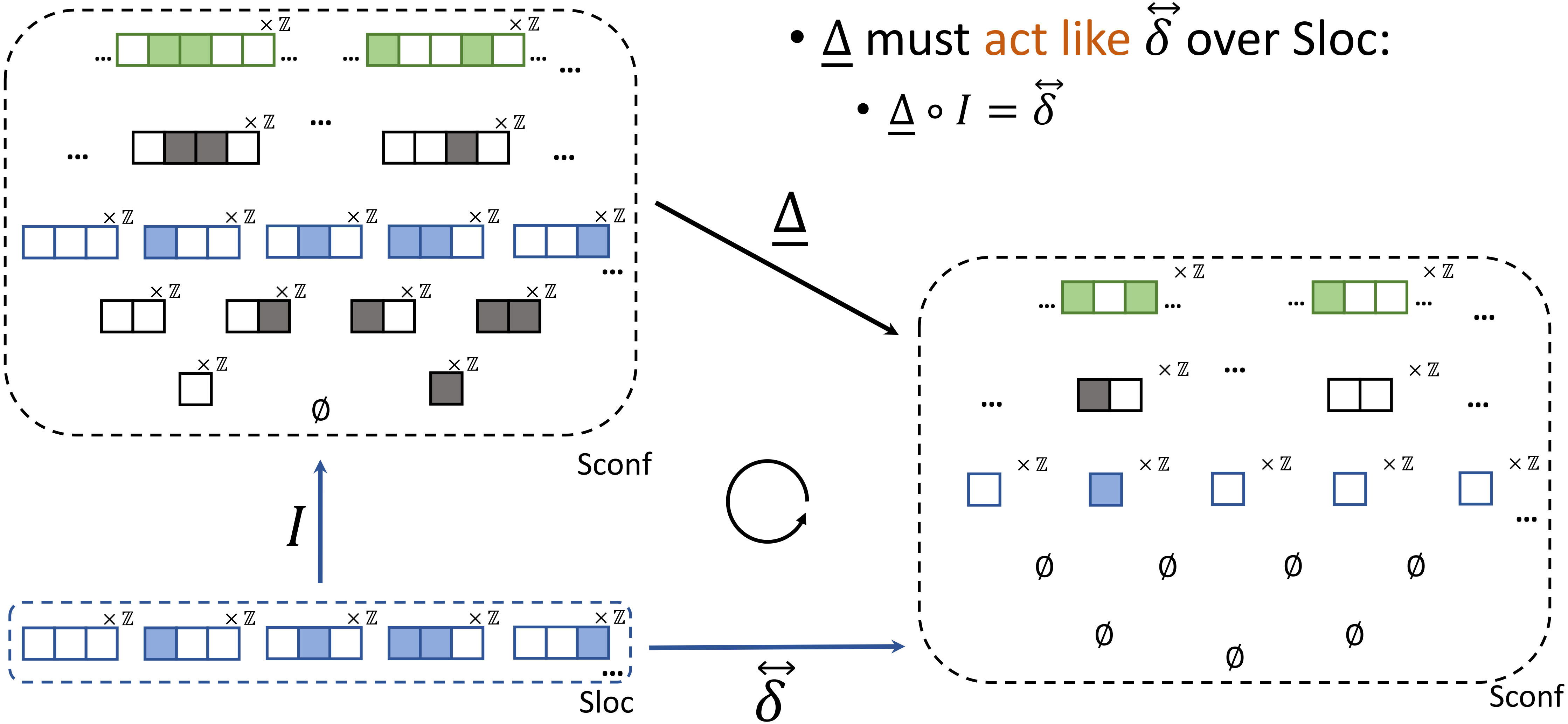
Coarse as left Kan extension



- Shifted local transition $\overleftrightarrow{\delta}$
- Sloc is included in Pconf by (monotonous) I

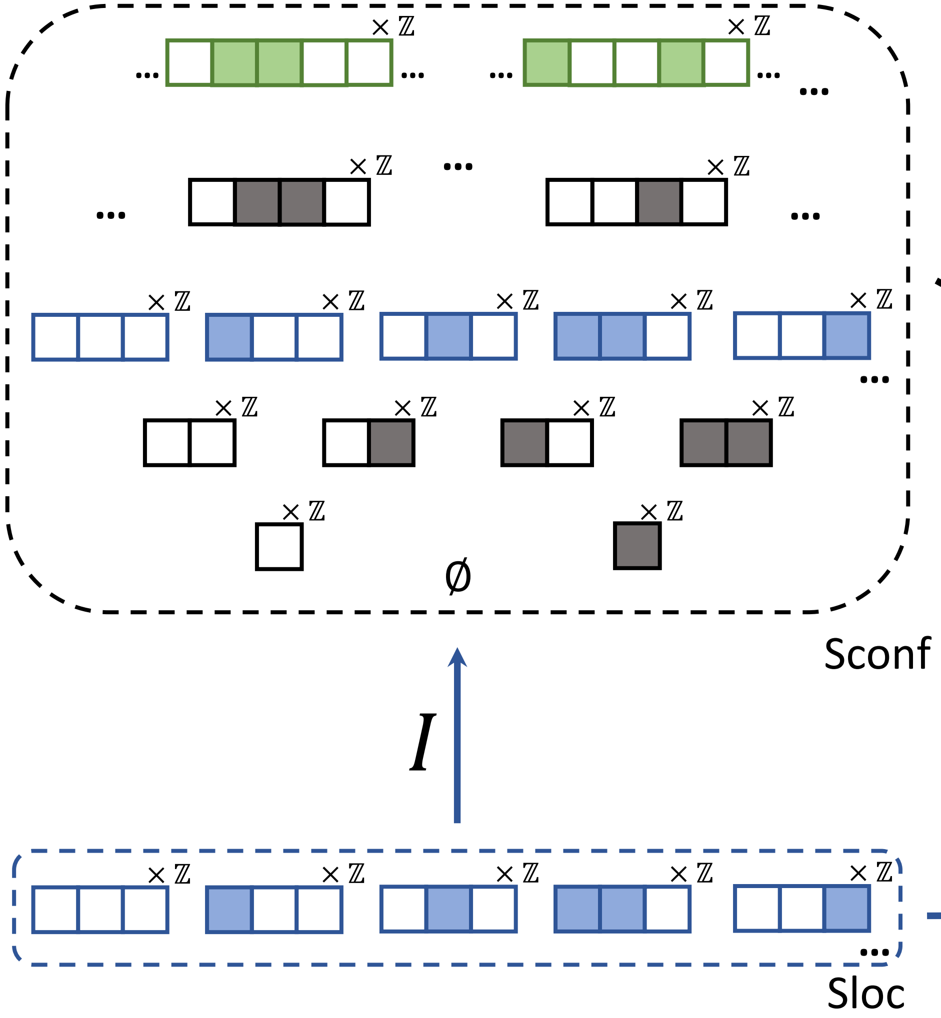
Coarse as left Kan extension

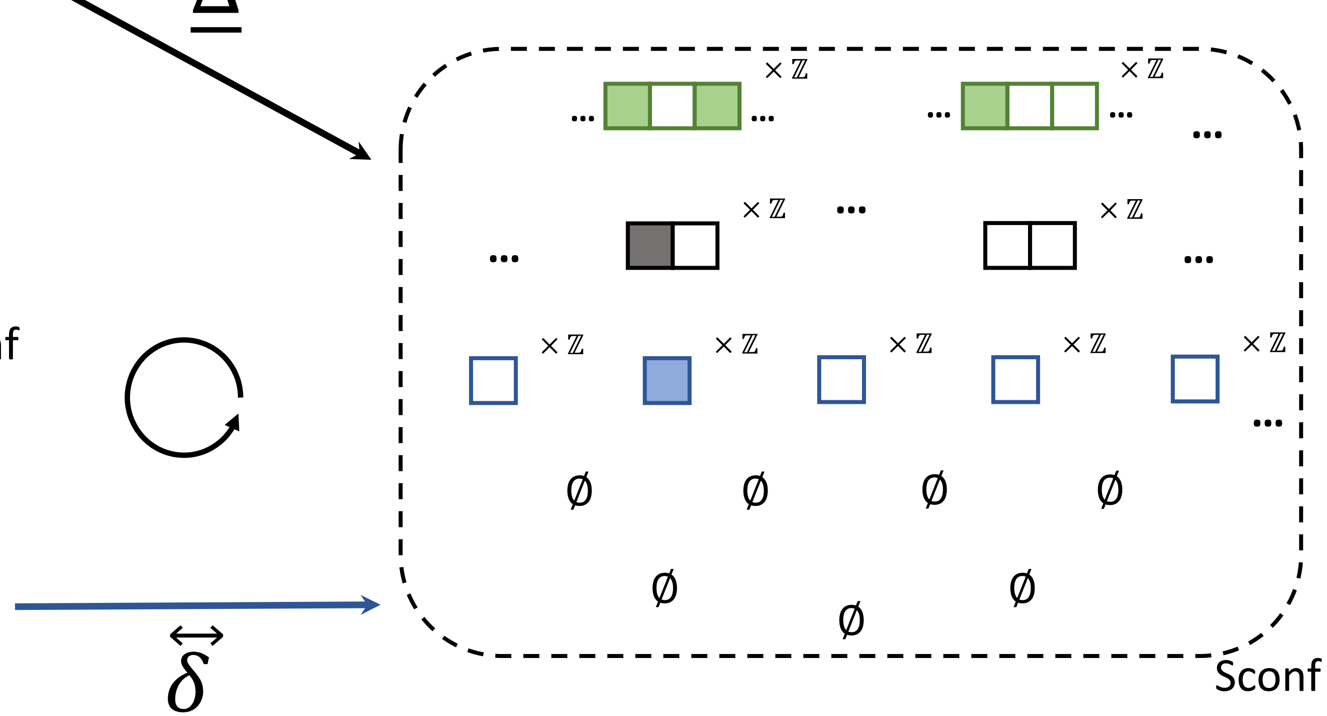
- $\underline{\Delta}$ must act like $\overleftrightarrow{\delta}$ over Sloc:
  - $\underline{\Delta} \circ I = \overleftrightarrow{\delta}$



Coarse as left Kan extension

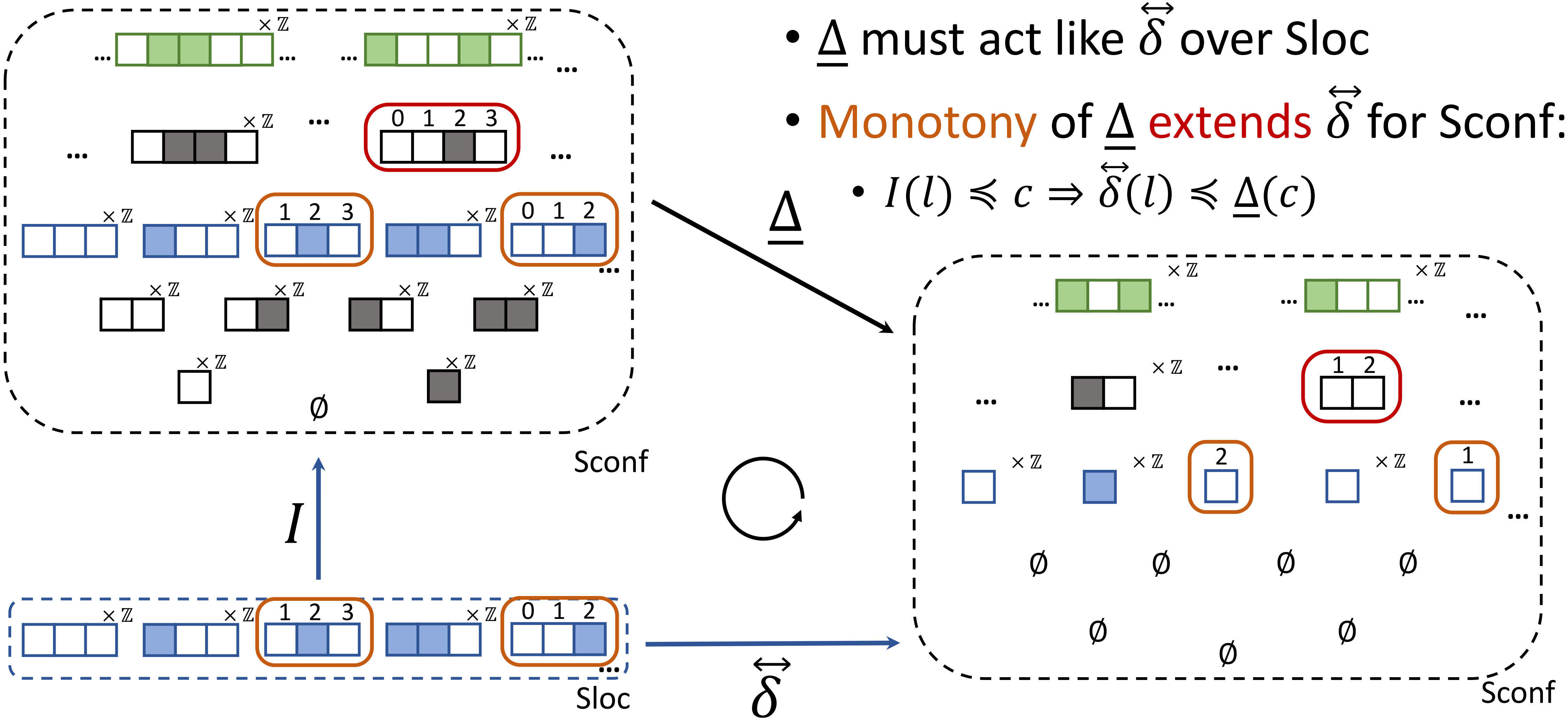
- $\underline{\Delta}$ must act like $\overleftrightarrow{\delta}$ over Sloc
- Monotony of $\underline{\Delta}$ extends $\overleftrightarrow{\delta}$ for Sconf:
  - $I(l) \preccurlyeq c \Rightarrow \overleftrightarrow{\delta}(l) \preccurlyeq \underline{\Delta}(c)$





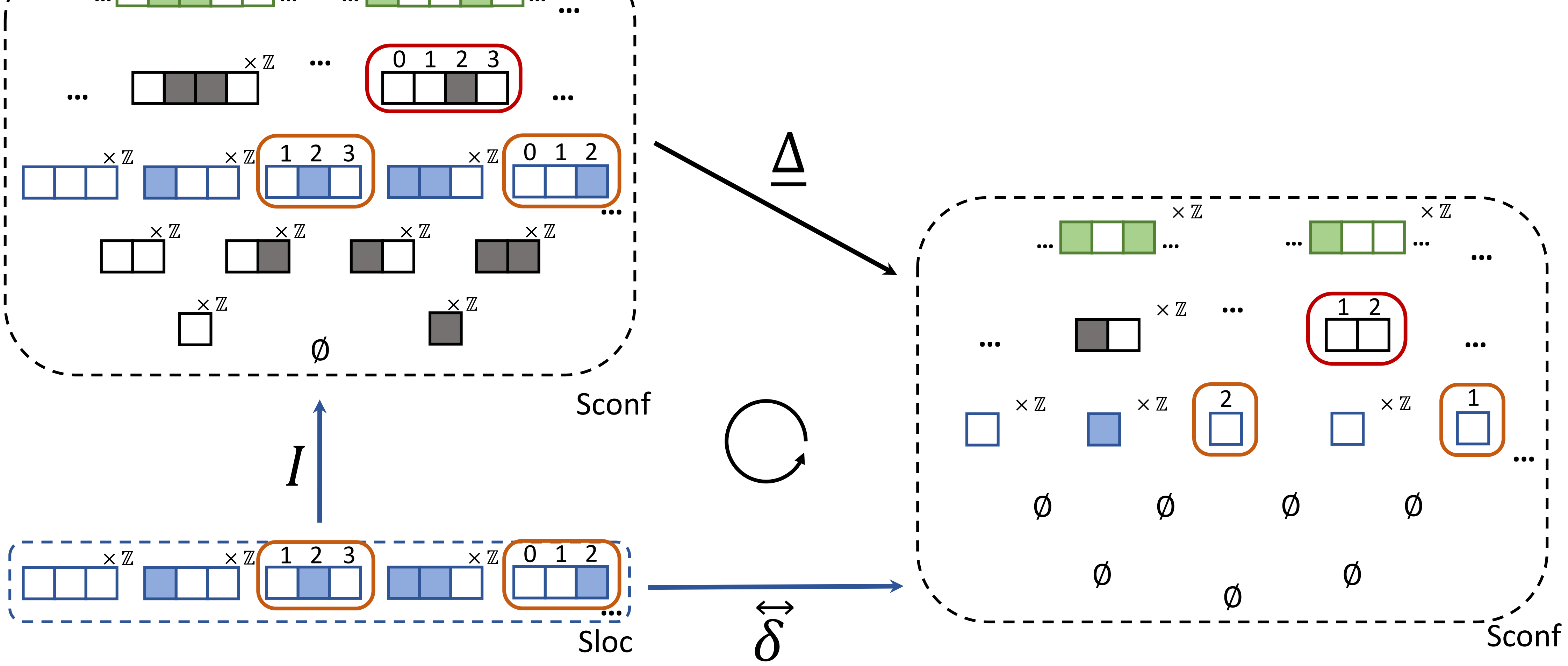
Coarse as left Kan extension

- $\underline{\Delta}$ must act like $\overleftrightarrow{\delta}$ over Sloc
- Monotony of $\underline{\Delta}$ extends $\overleftrightarrow{\delta}$ for Sconf:
  - $I(l) \preccurlyeq c \Rightarrow \overleftrightarrow{\delta}(l) \preccurlyeq \underline{\Delta}(c)$



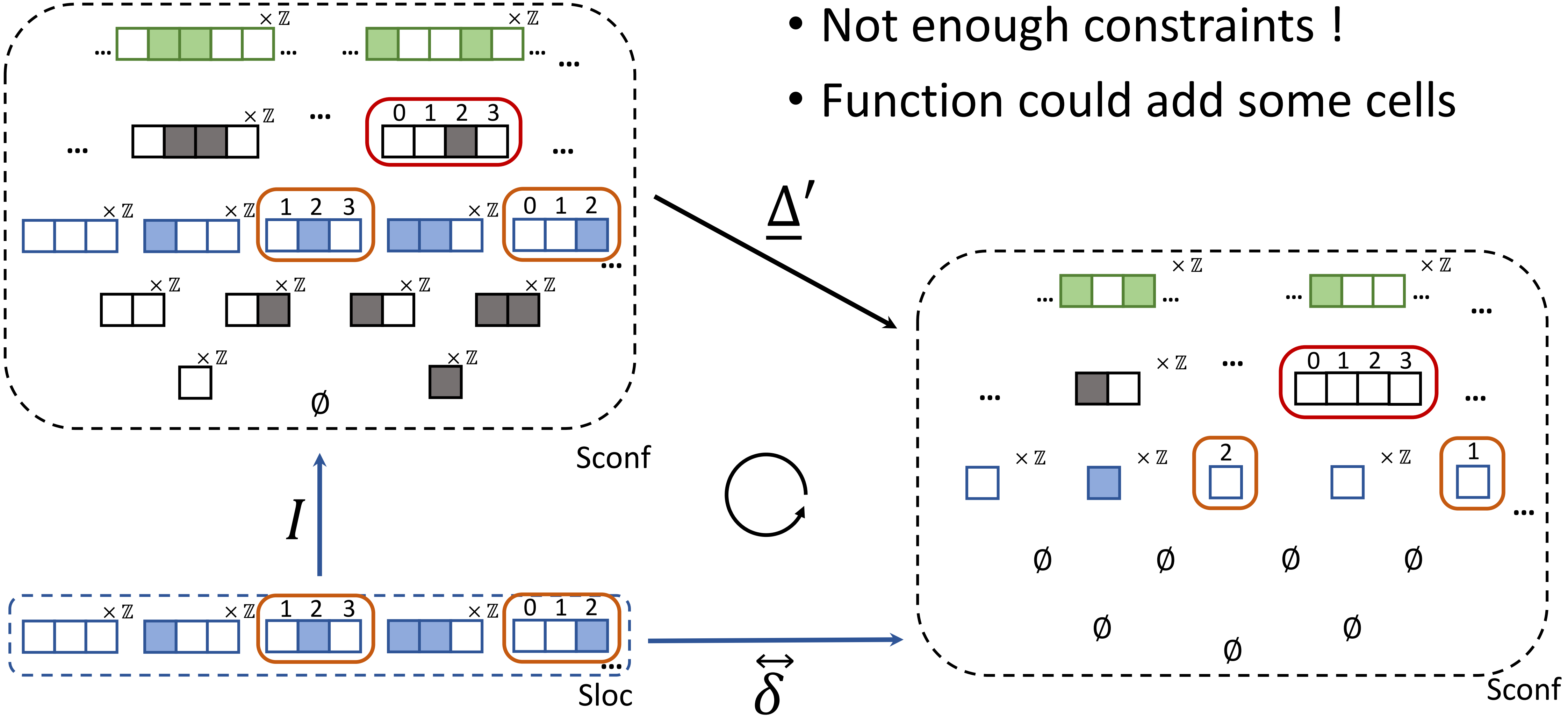
Coarse as left Kan extension

- Not enough constraints !



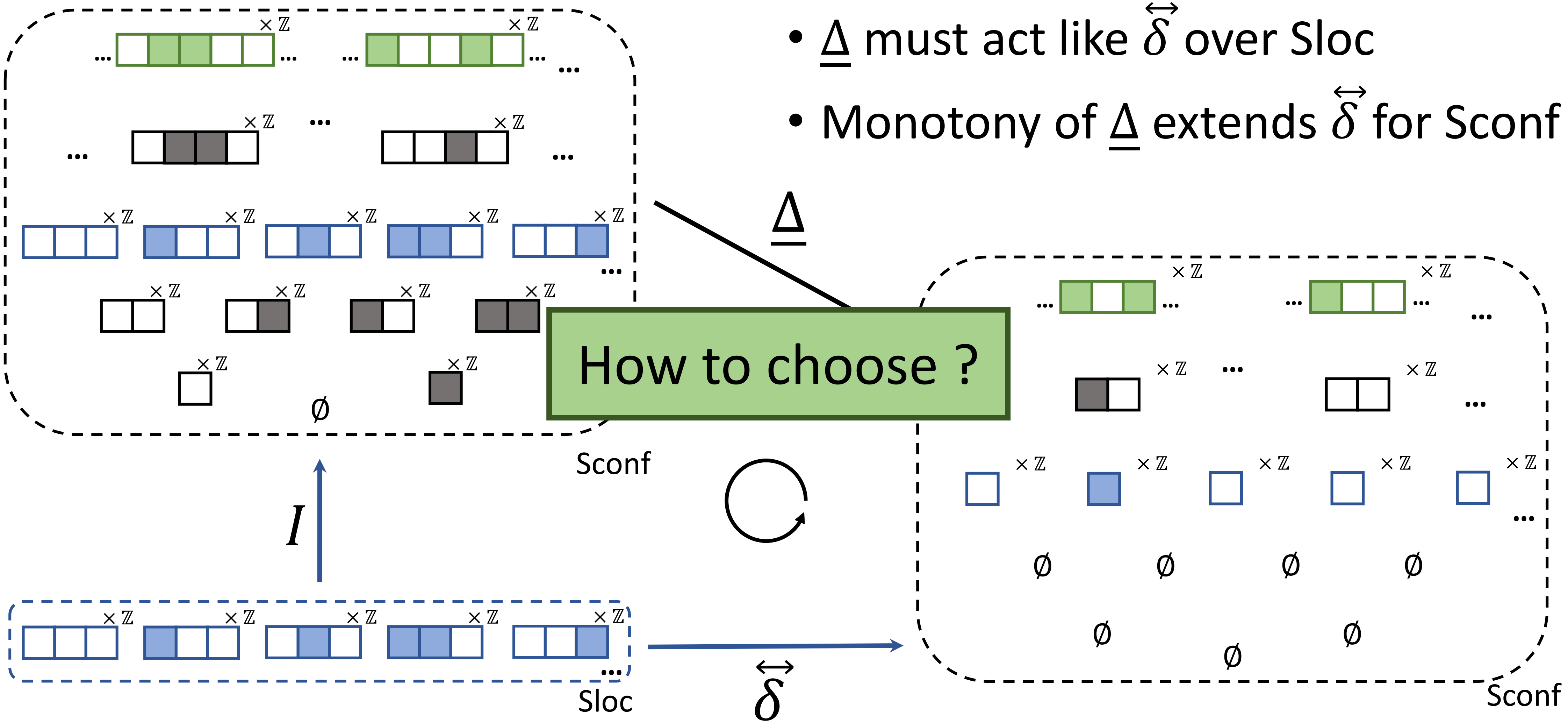
Coarse as left Kan extension

- Not enough constraints !
- Function could add some cells

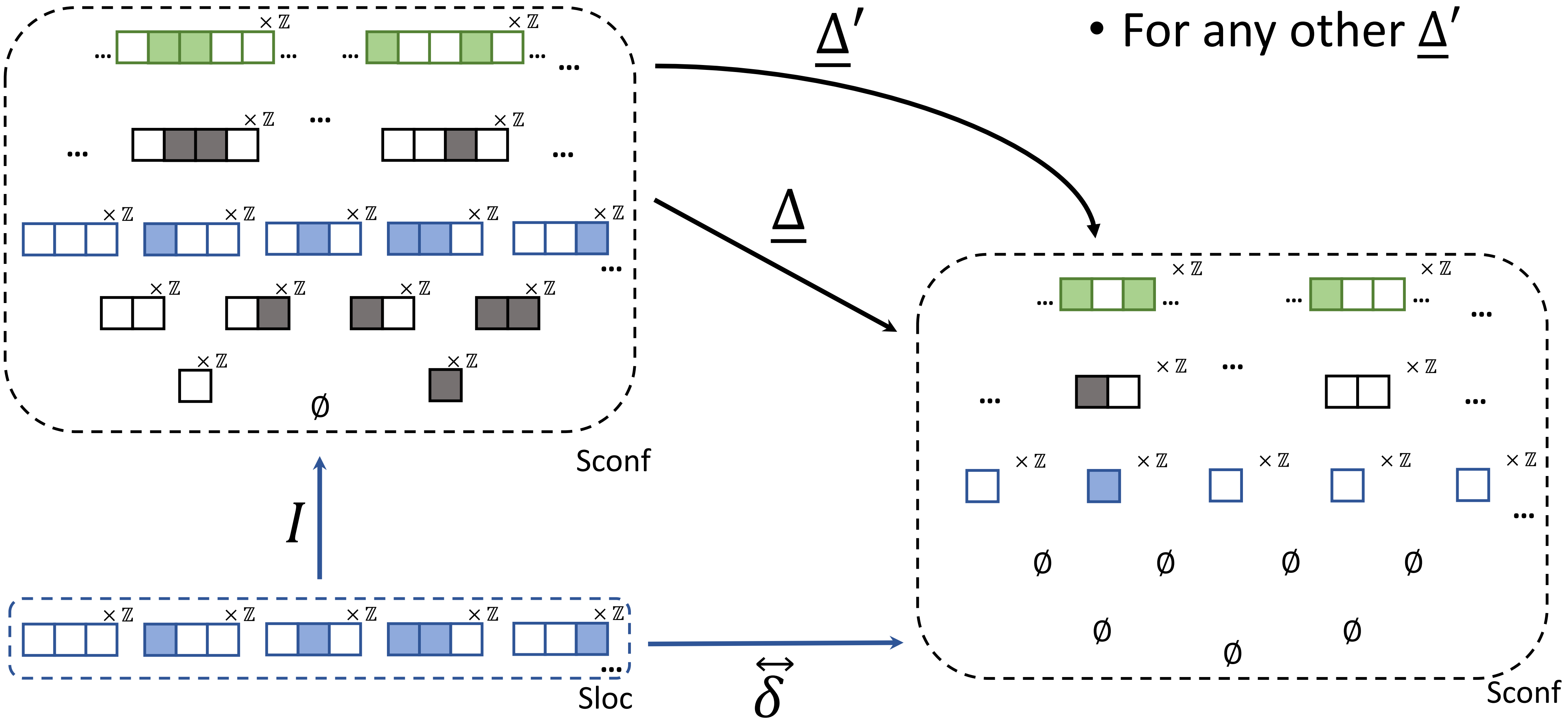


Coarse as left Kan extension

- $\underline{\Delta}$ must act like $\overleftrightarrow{\delta}$ over Sloc
- Monotony of $\underline{\Delta}$ extends $\overleftrightarrow{\delta}$ for Sconf

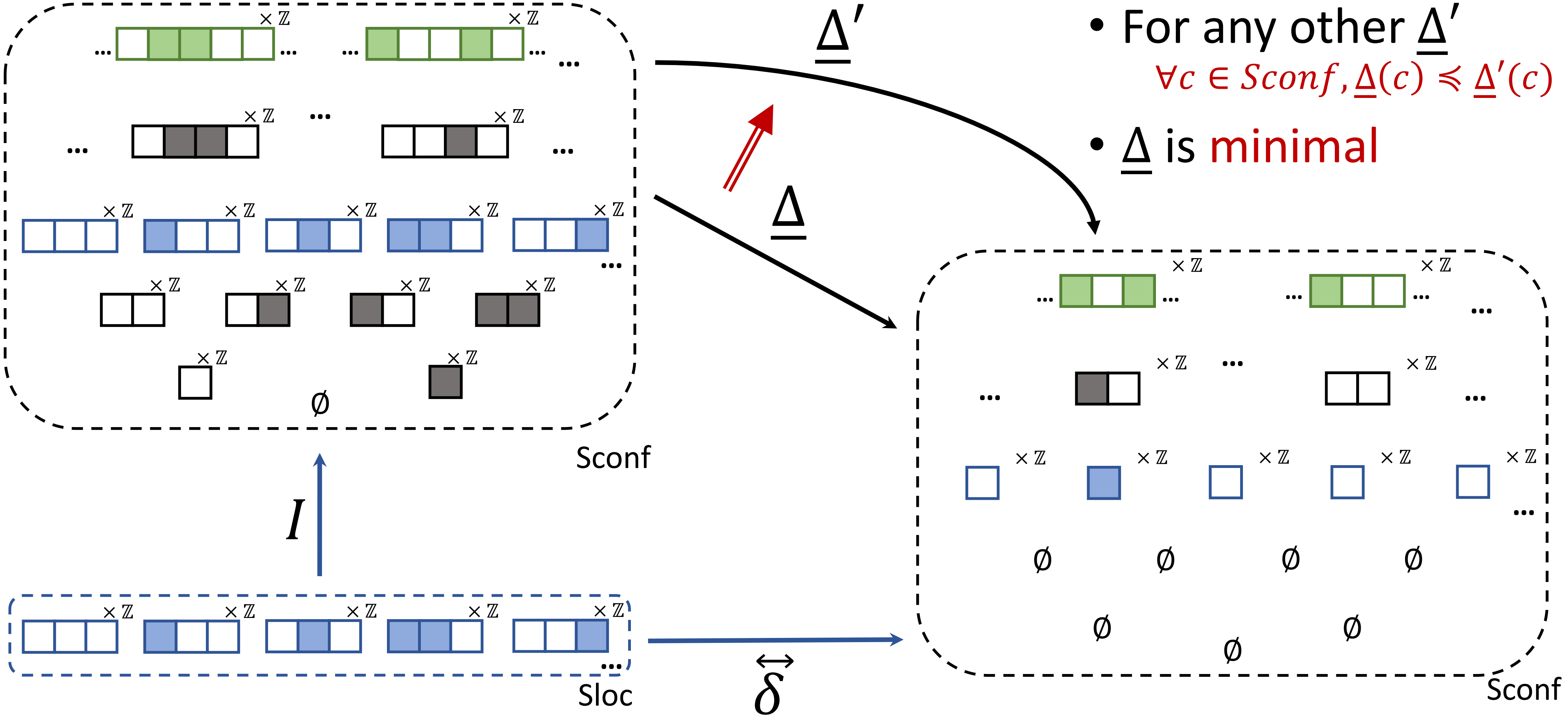


Coarse as left Kan extension



- For any other $\underline{\Delta}'$

Coarse as left Kan extension



- For any other $\underline{\Delta}'$

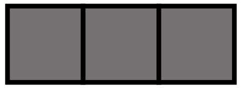
  $\forall c \in Sconf, \underline{\Delta}(c) \preccurlyeq \underline{\Delta}'(c)$

- $\underline{\Delta}$ is minimal

Coarse as left Kan extension

- If torsion of space, more complicated…

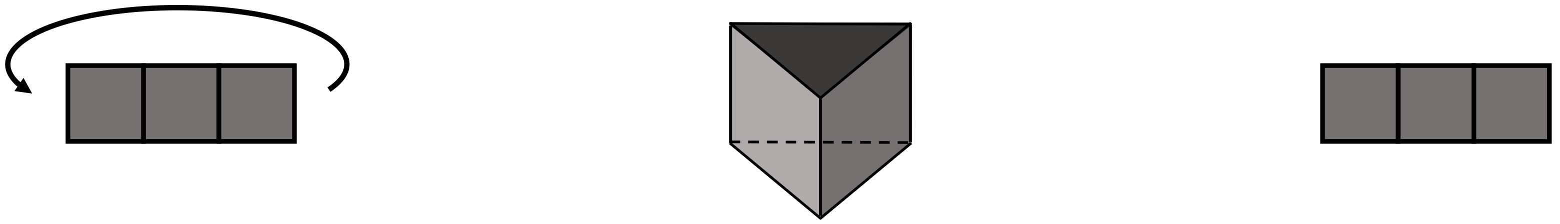
Space : closed ribbon



Sub-configuration

Coarse as left Kan extension

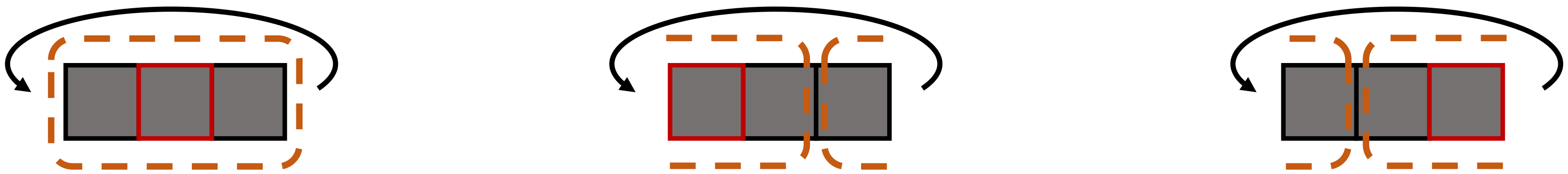
- If torsion of space, more complicated…



Space : closed ribbon

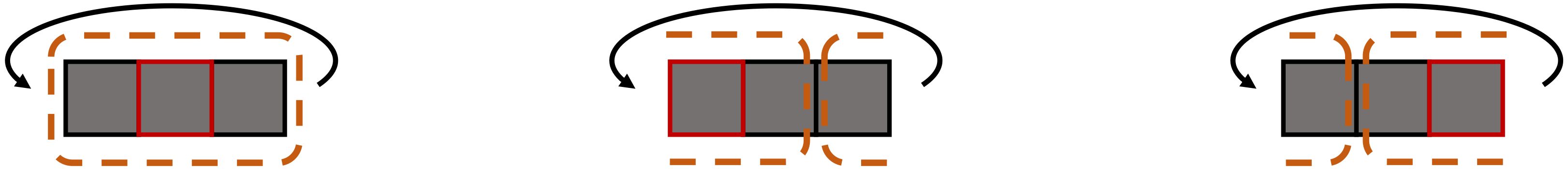
Sub-configuration

- Sub-configuration = multiple neighborhoods !



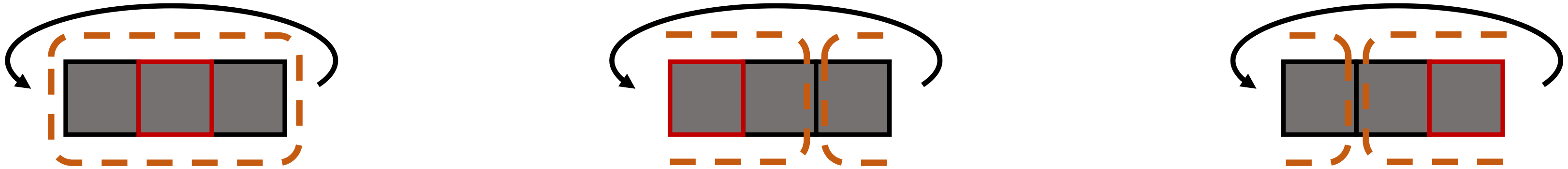
Coarse as left Kan extension

- If torsion of space, more complicated…
- Sub-configuration = multiple neighborhoods !



Coarse as left Kan extension

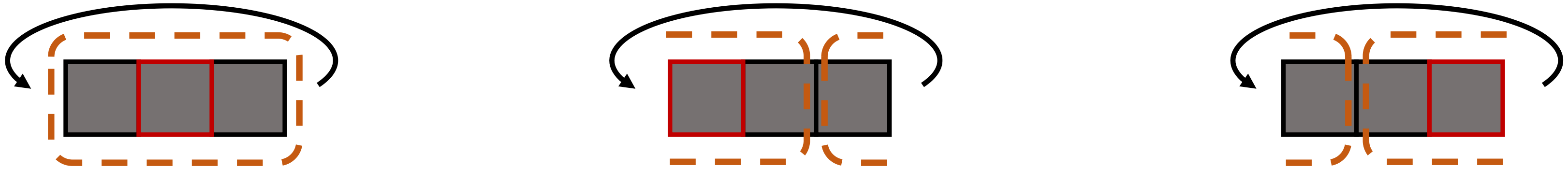
- If torsion of space, more complicated…
- Sub-configuration = multiple neighborhoods !



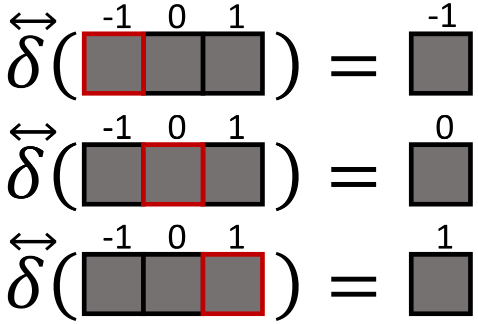
- Rules must know the center, extension does not

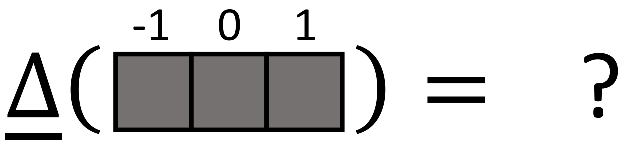
Coarse as left Kan extension

- If torsion of space, more complicated…
- Sub-configuration = multiple neighborhoods !



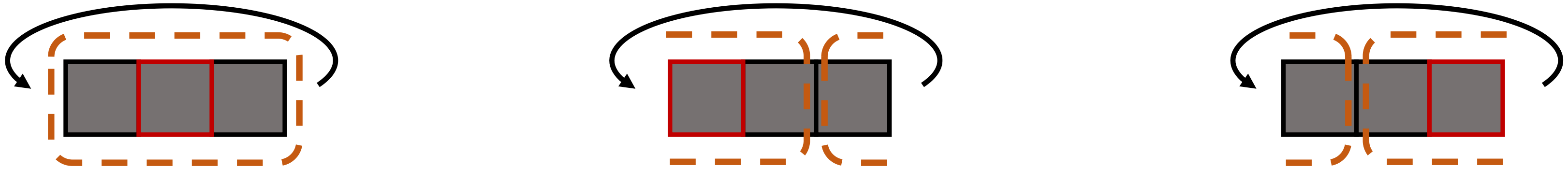
- Rules must know the center, extension does not
  - Extension = ?



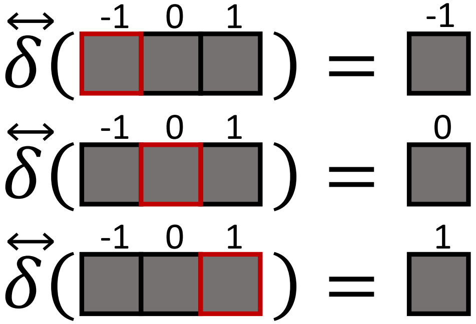


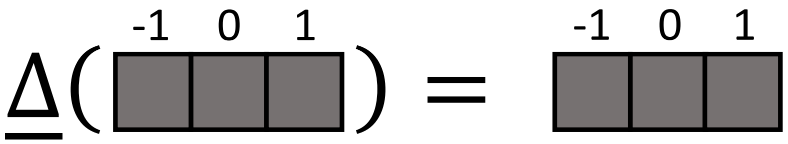
Coarse as left Kan extension

- If torsion of space, more complicated…
- Sub-configuration = multiple neighborhoods !



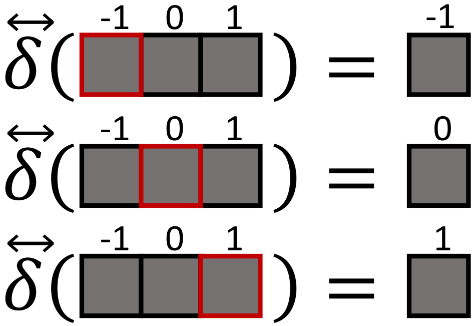
- Rules must know the center, extension does not
  - Extension = multiple rules at one time !

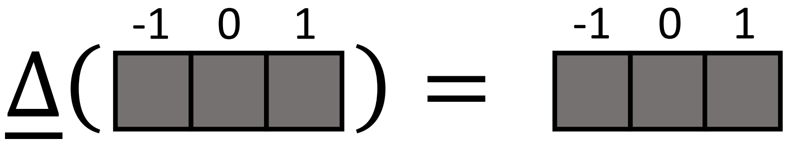




Coarse as left Kan extension

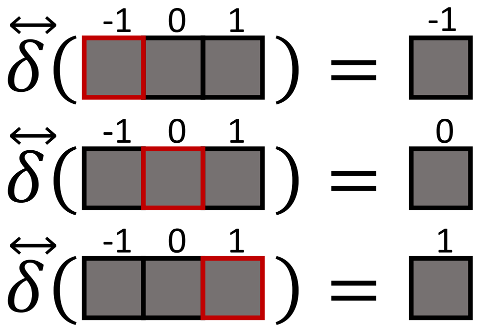
- If torsion of space, more complicated…
- Sub-configuration = multiple neighborhoods !
- Rules must know the center, extension does not
  - Extension = multiple rules at one time !

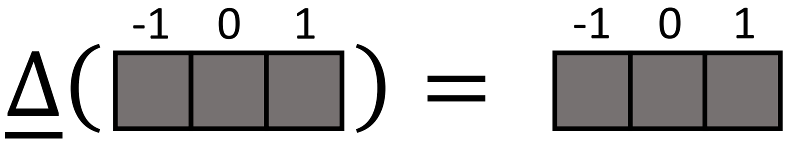




Coarse as left Kan extension

- If torsion of space, more complicated…
- Sub-configuration = multiple neighborhoods !
- Rules must know the center, extension does not
  - Extension = multiple rules at one time !





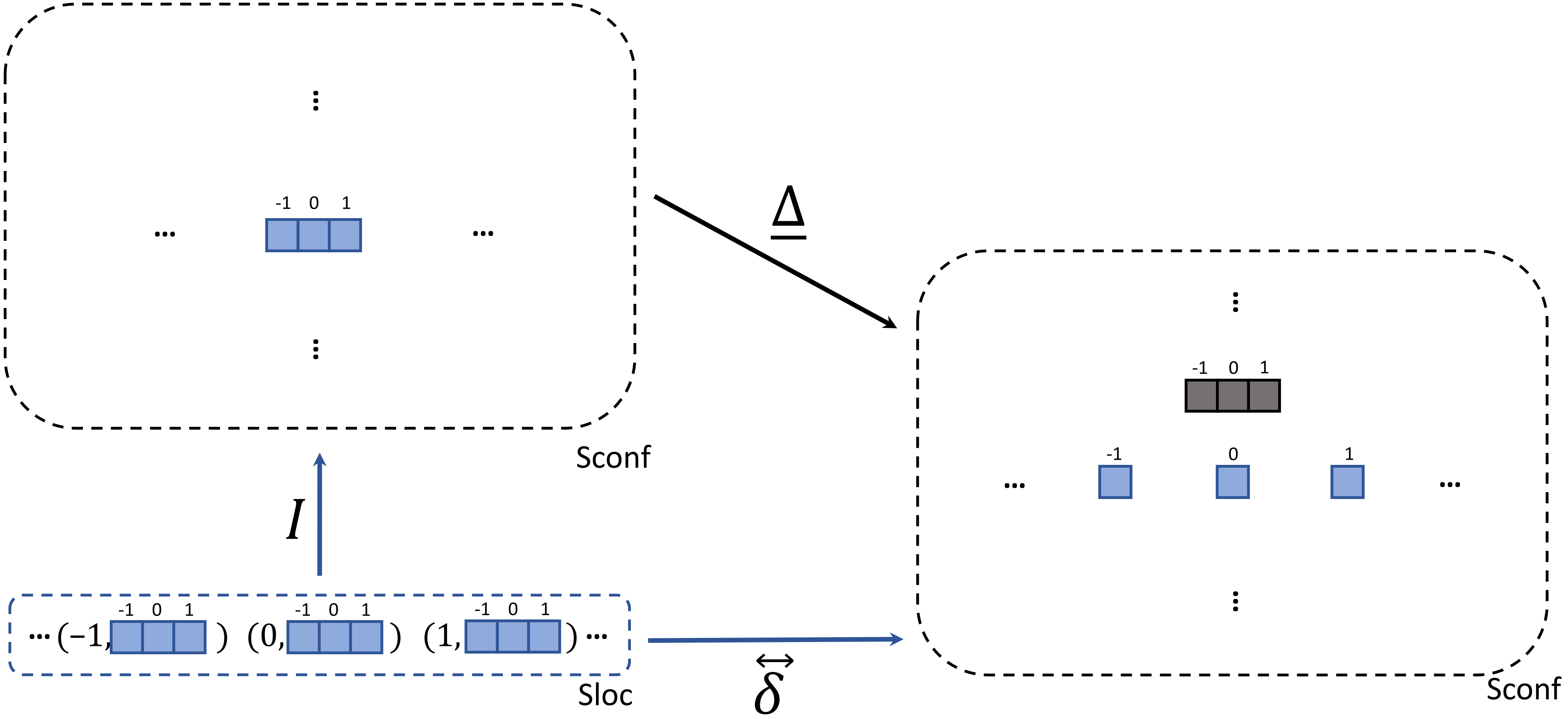
- Redefine $Sloc \coloneqq \bigcup_{p \in G} \{p\} \times Q^{(p+N)}$

Coarse as left Kan extension

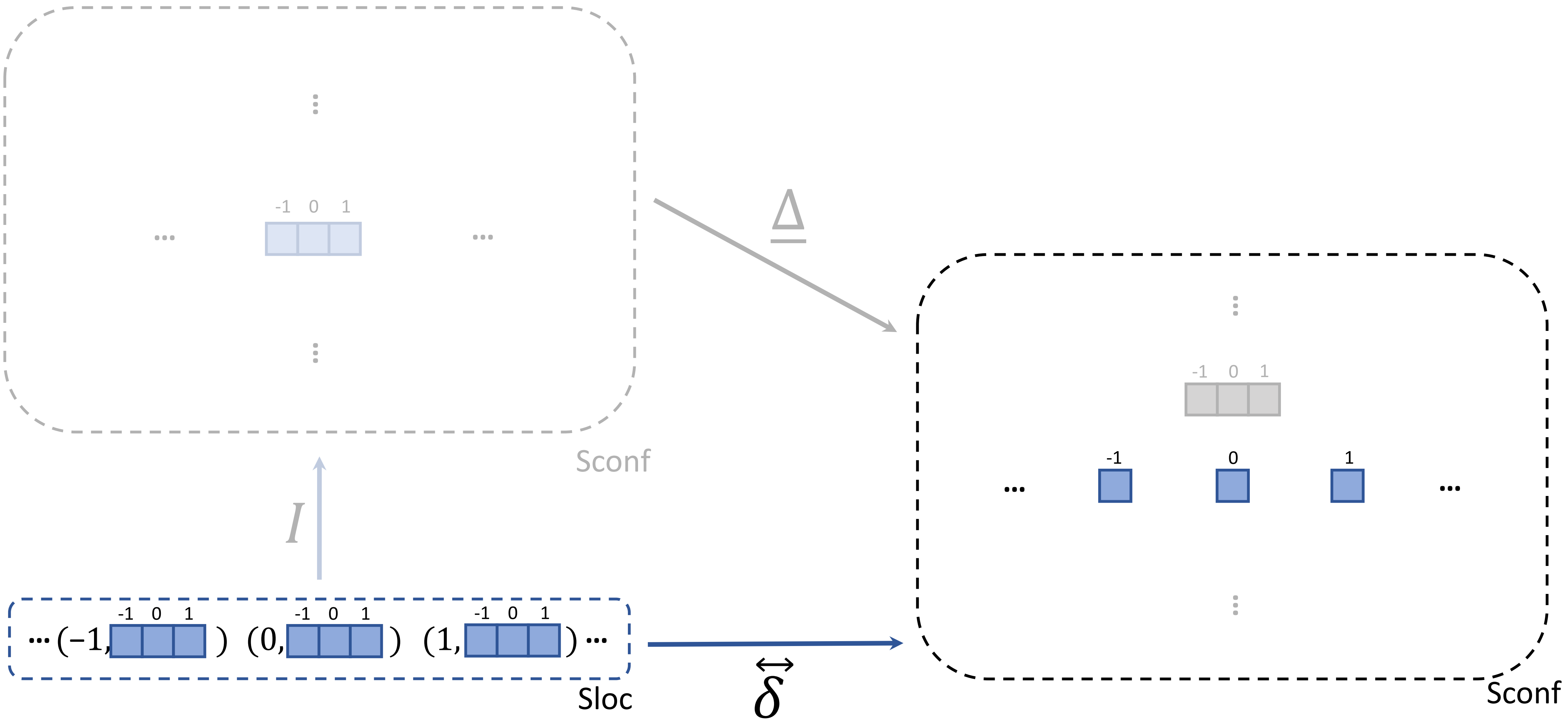
- If torsion of space, more complicated…
- Sub-configuration = multiple neighborhoods !
- Rules must know the center, extension does not
  - Extension = multiple rules at one time !

- Redefine $Sloc \coloneqq \bigcup_{p \in G} \{p\} \times Q^{(p+N)}$
- And $I: Sloc \to SConf \coloneqq \pi_2$ not injective !

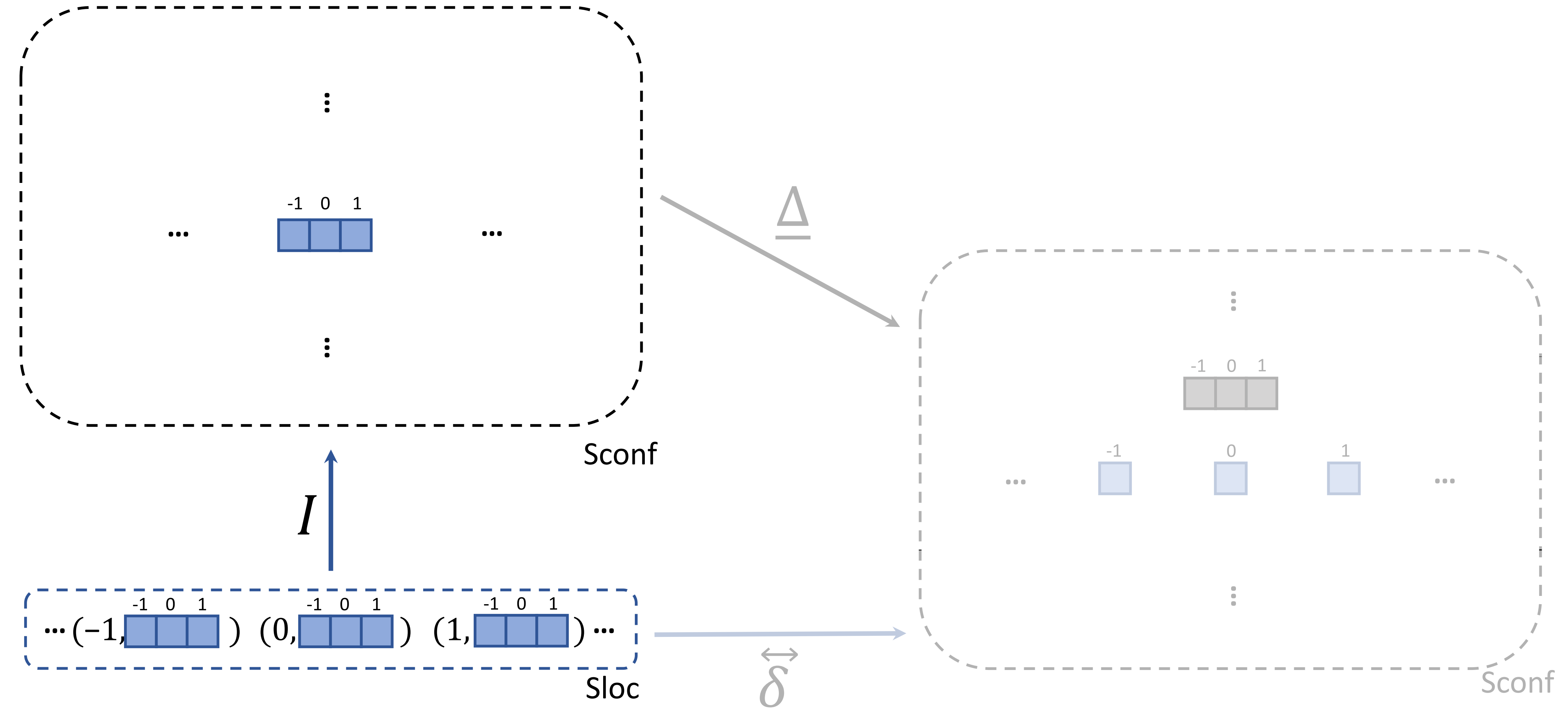
Coarse as left Kan extension



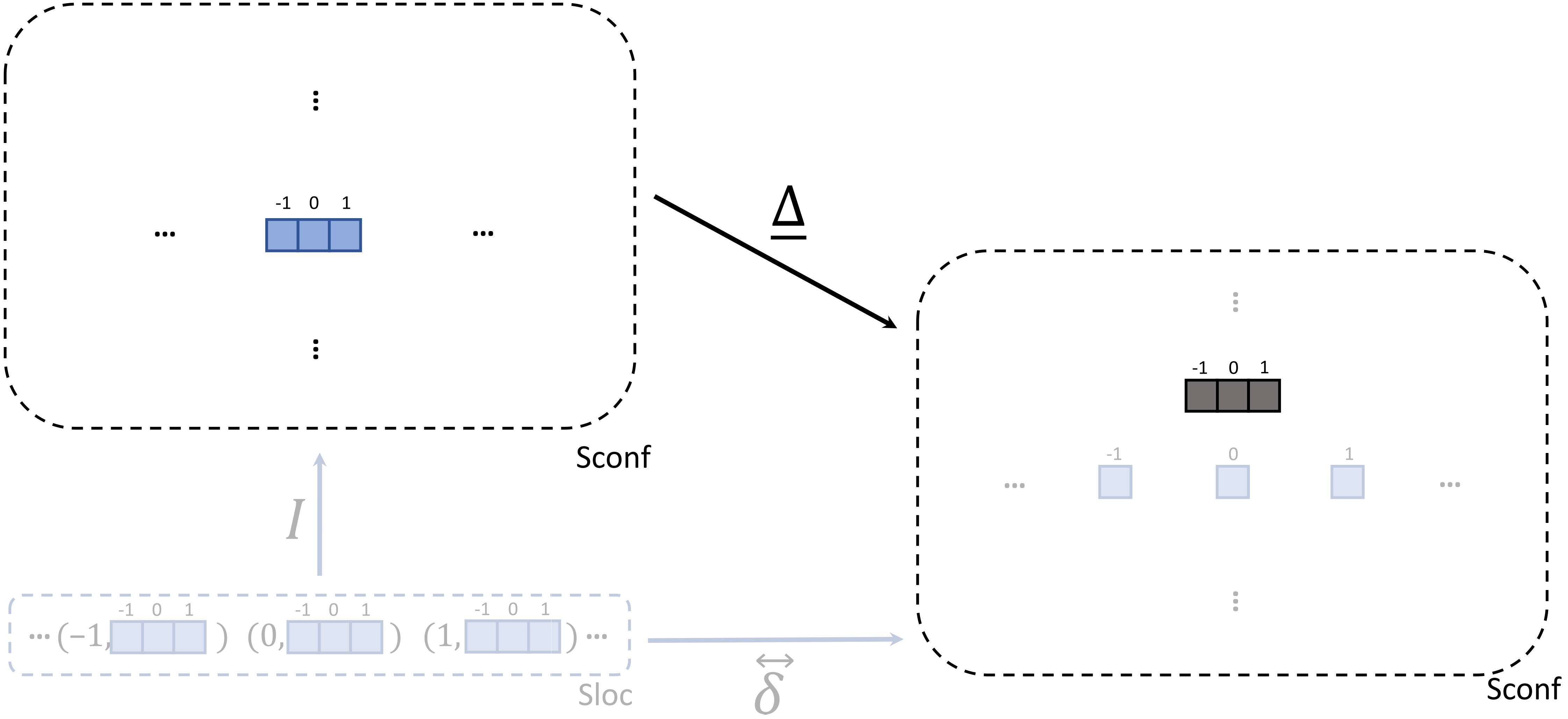
Coarse as left Kan extension



Coarse as left Kan extension

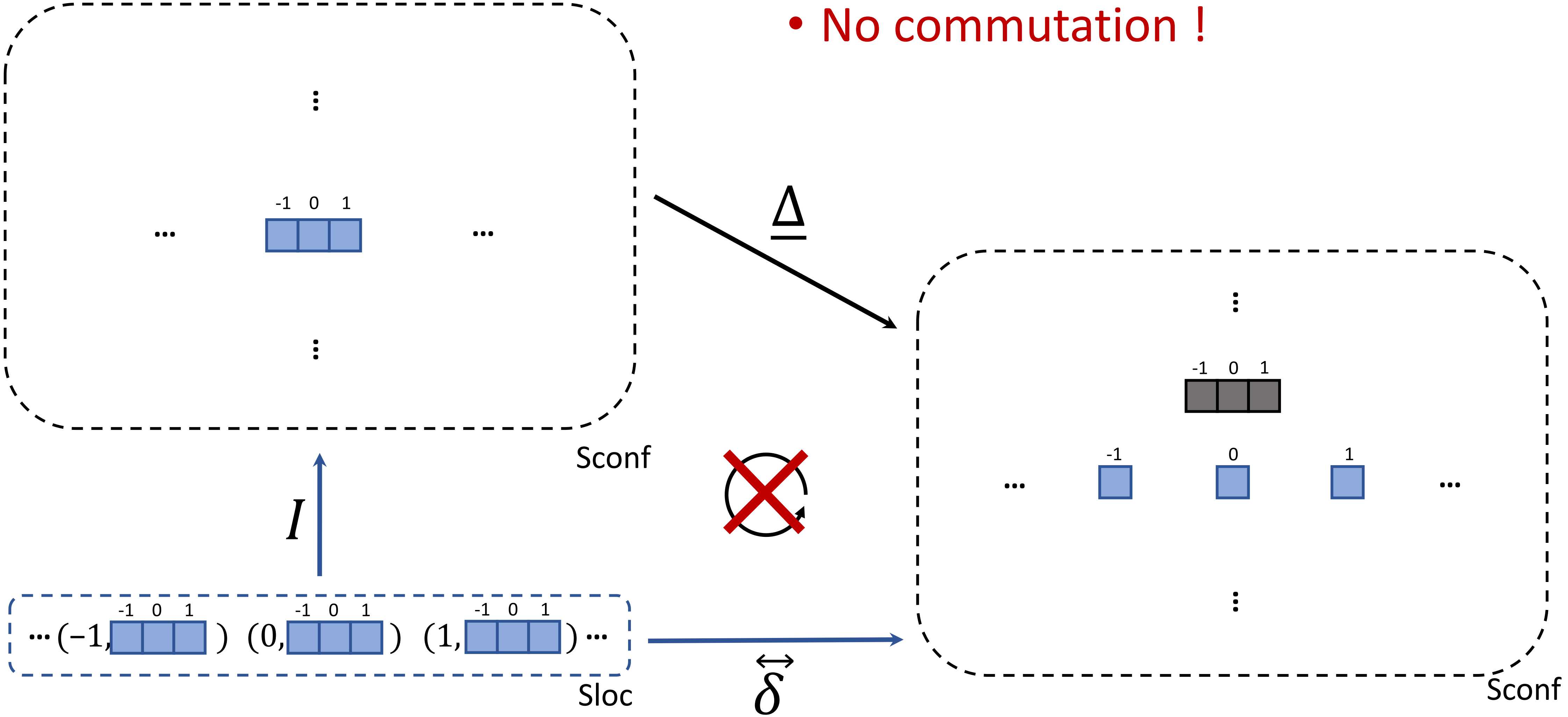


Coarse as left Kan extension



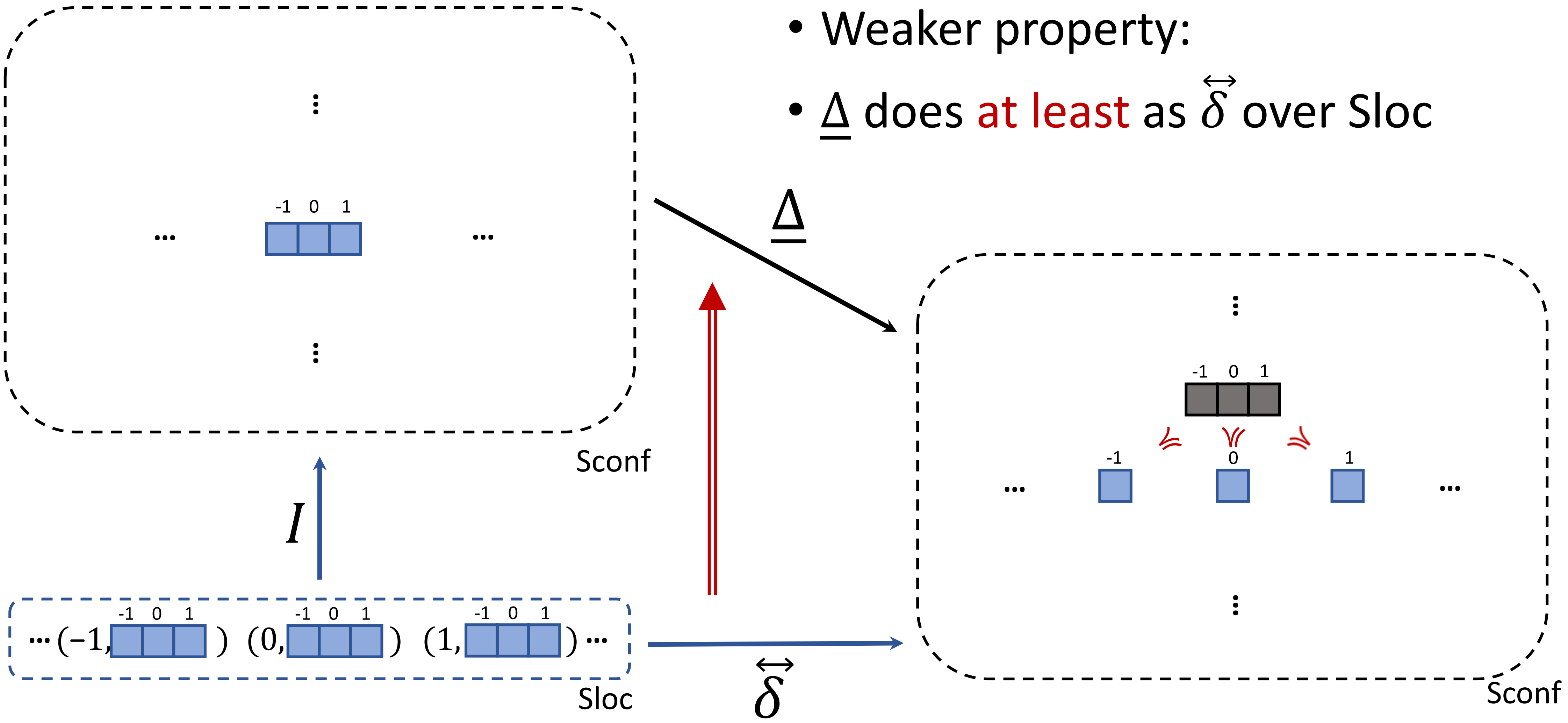
Coarse as left Kan extension

- No commutation !

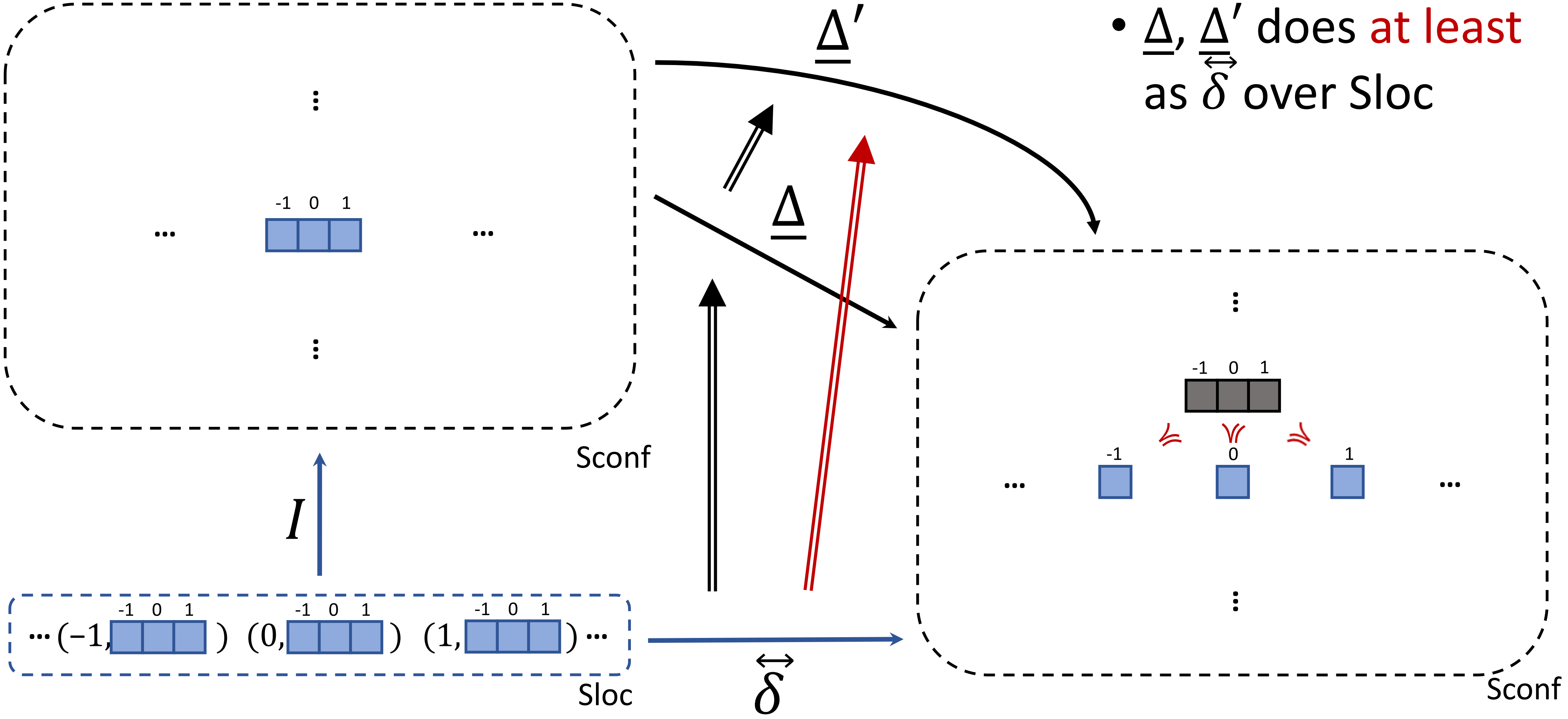


Coarse as left Kan extension

- Weaker property:
- $\underline{\Delta}$ does at least as $\overleftrightarrow{\delta}$ over Sloc

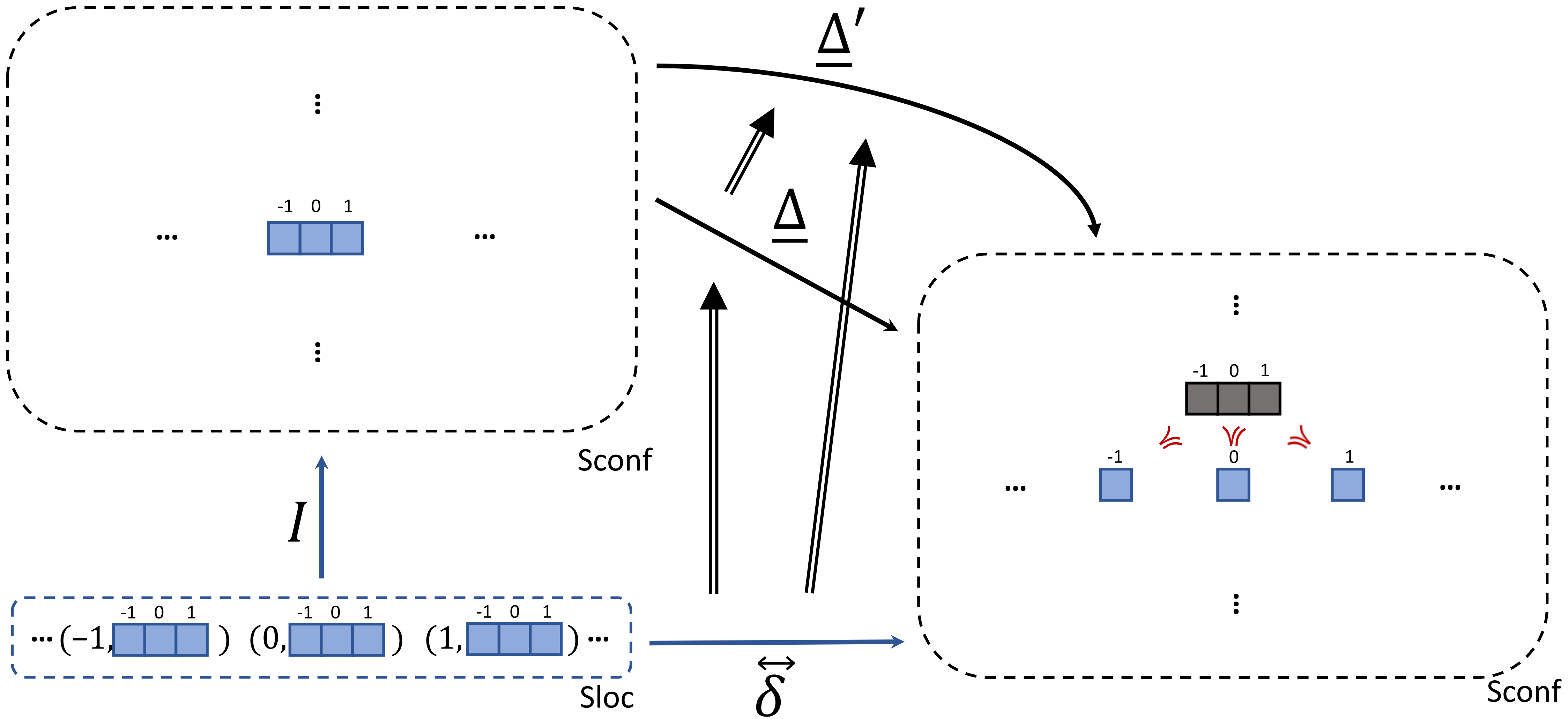


Coarse as left Kan extension



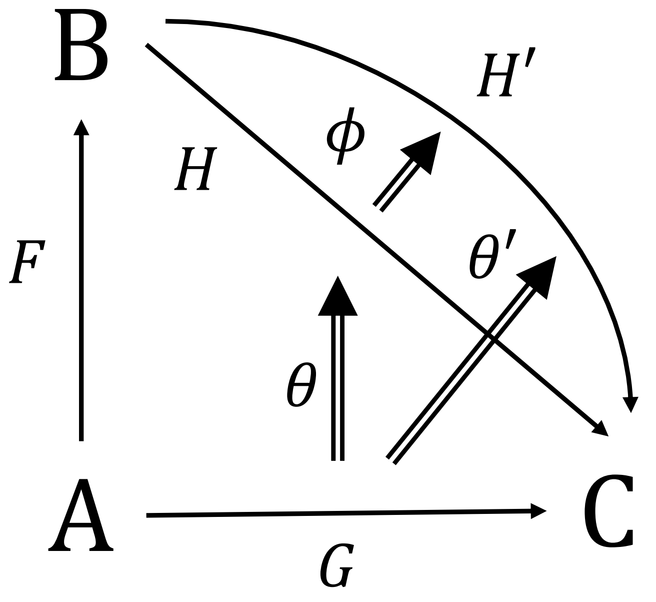
- $\underline{\Delta}$, $\underline{\Delta}'$ does at least as $\overleftrightarrow{\delta}$ over Sloc

Coarse as left Kan extension



Left Kan Extensions

- Left Kan Extension of G along F is some $\langle H, \theta \rangle$



- $\theta: G \Rightarrow H \circ F$

- Universal :
  - Given $\langle H', \theta' \rangle$, exists unique ϕ s.t.

  - $\theta' = (\phi \circ F) \cdot \theta$

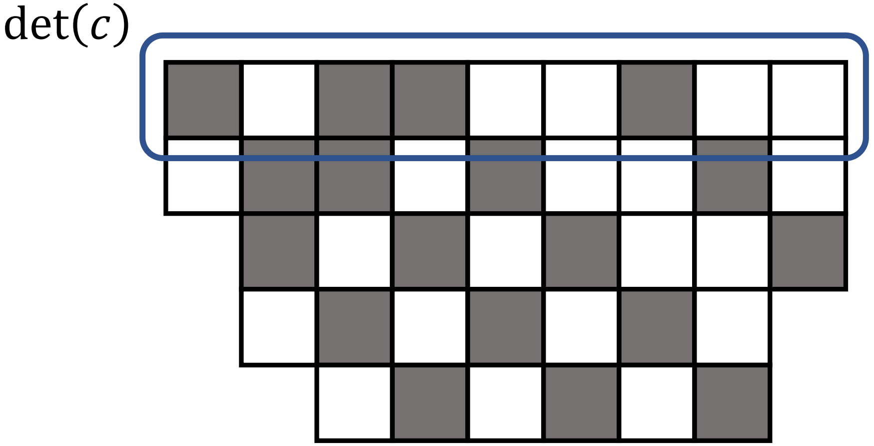
- H « **minimal** » and unique

Two way of extending

- Coarse transition function $\underline{\Delta}$
  - Deduce with full neighborhood

- Is a left Kan extension !

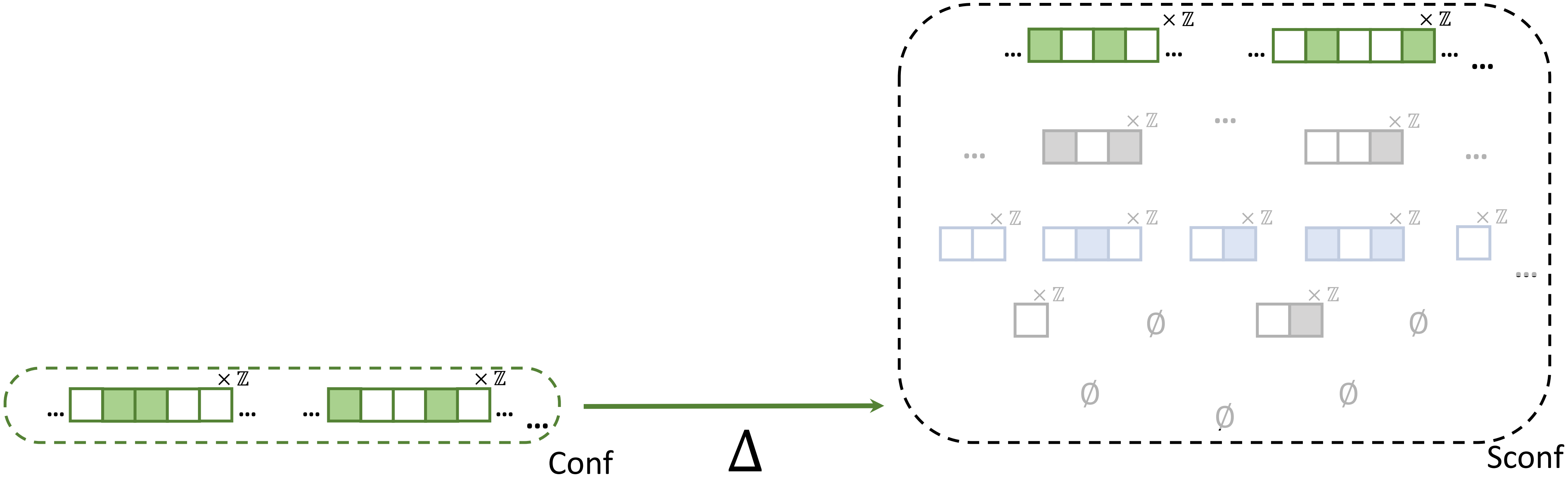
- Fine transition function $\overline{\Delta}$
  - Deduce without neighborhood



- Is a right Kan extension !

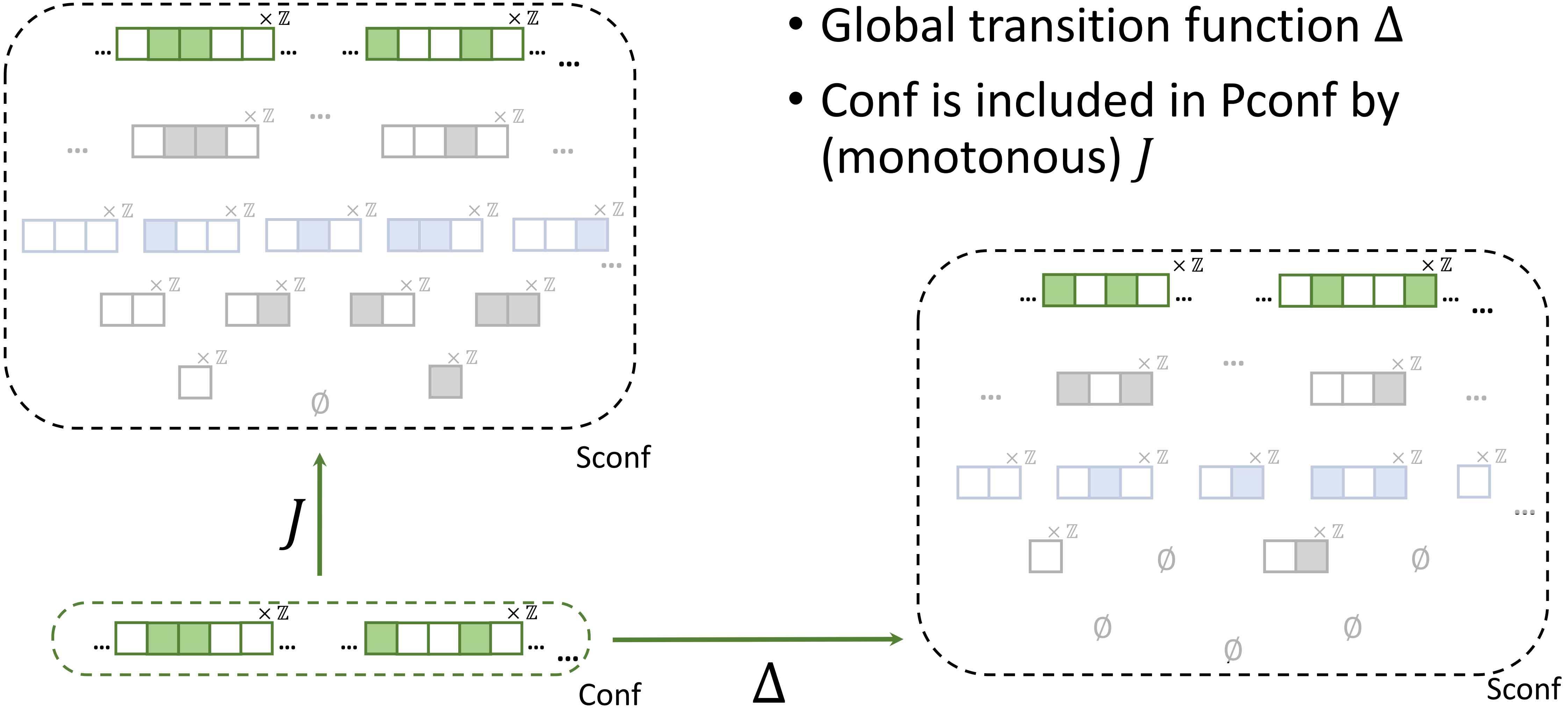
Fine as right Kan extension

- Global transition function Δ



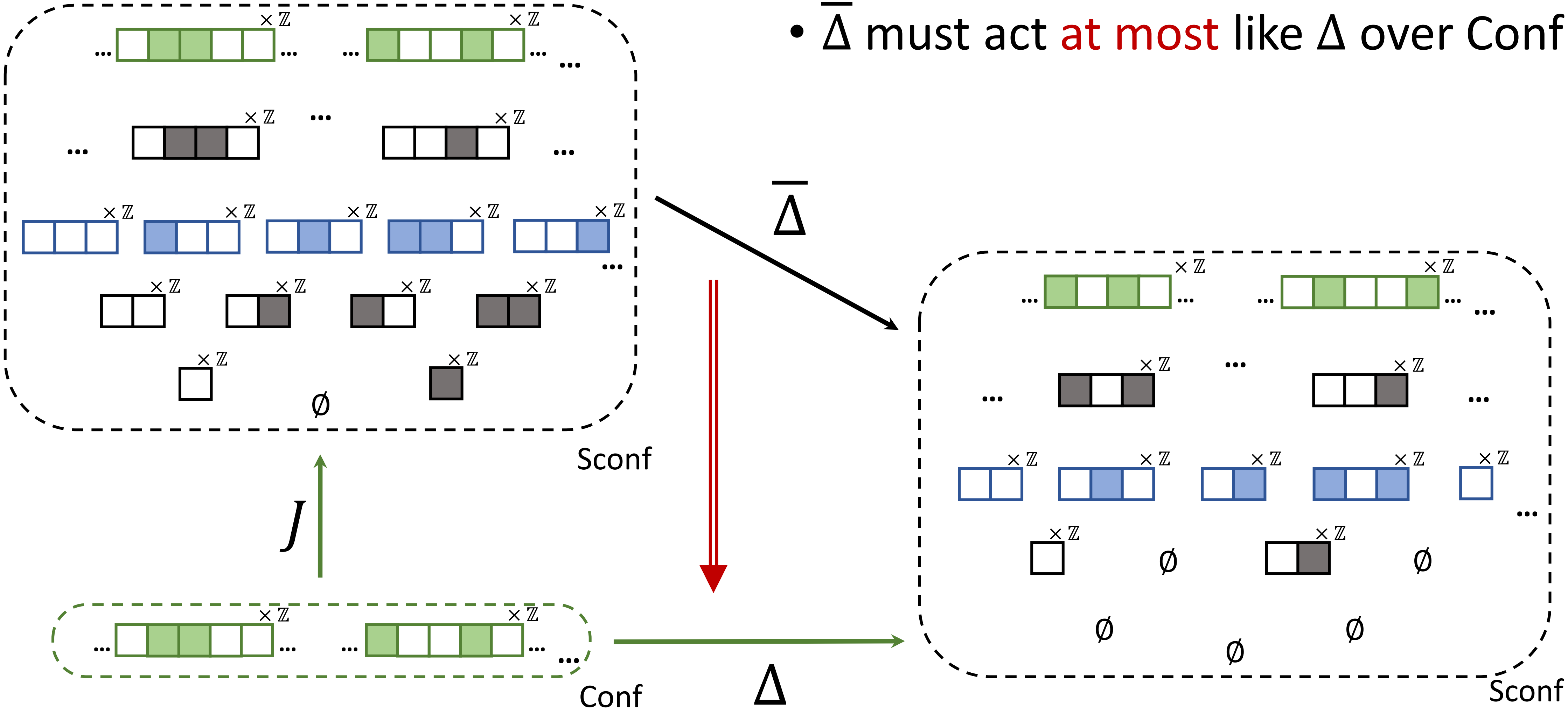
Fine as right Kan extension

- Global transition function Δ
- Conf is included in Pconf by (monotonous) J



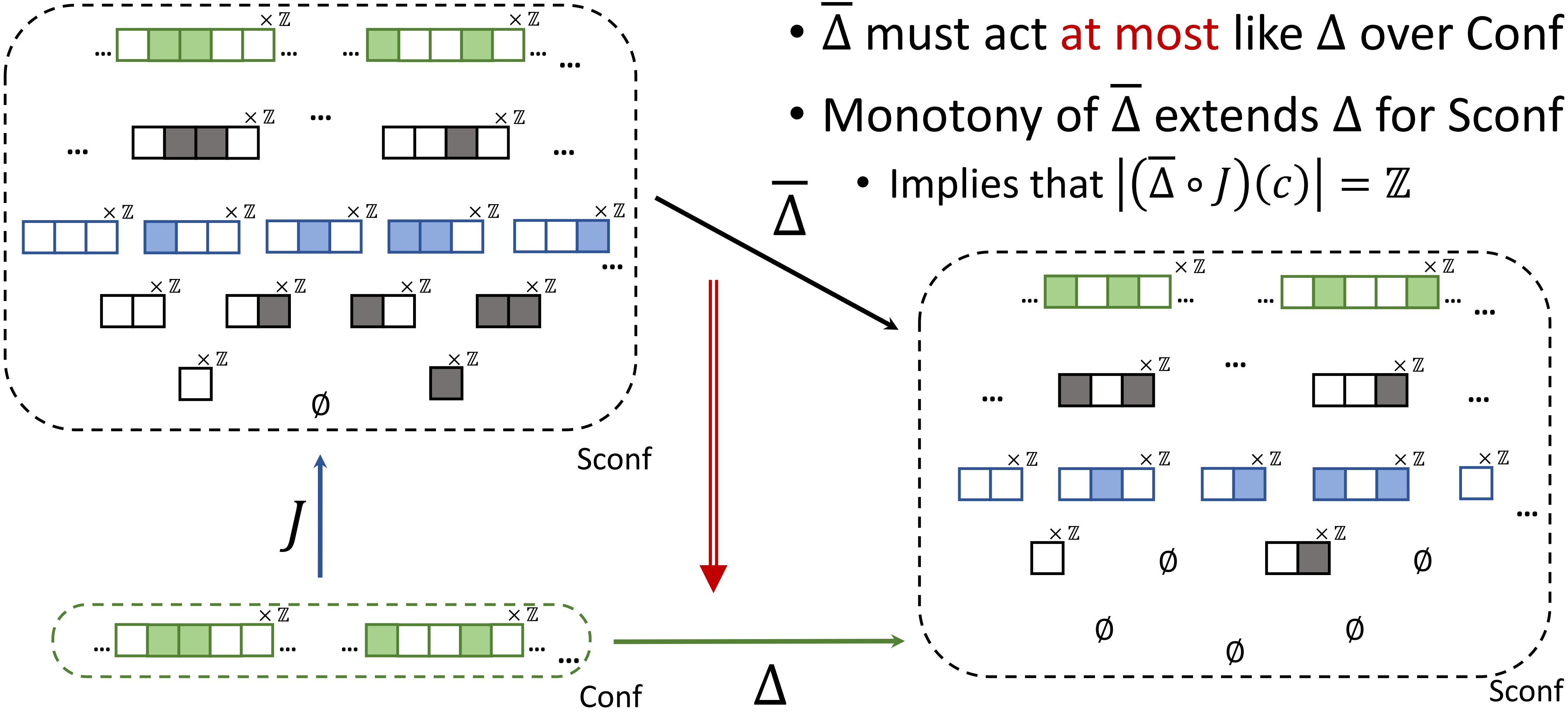
Fine as right Kan extension

- $\bar{\Delta}$ must act at most like Δ over Conf



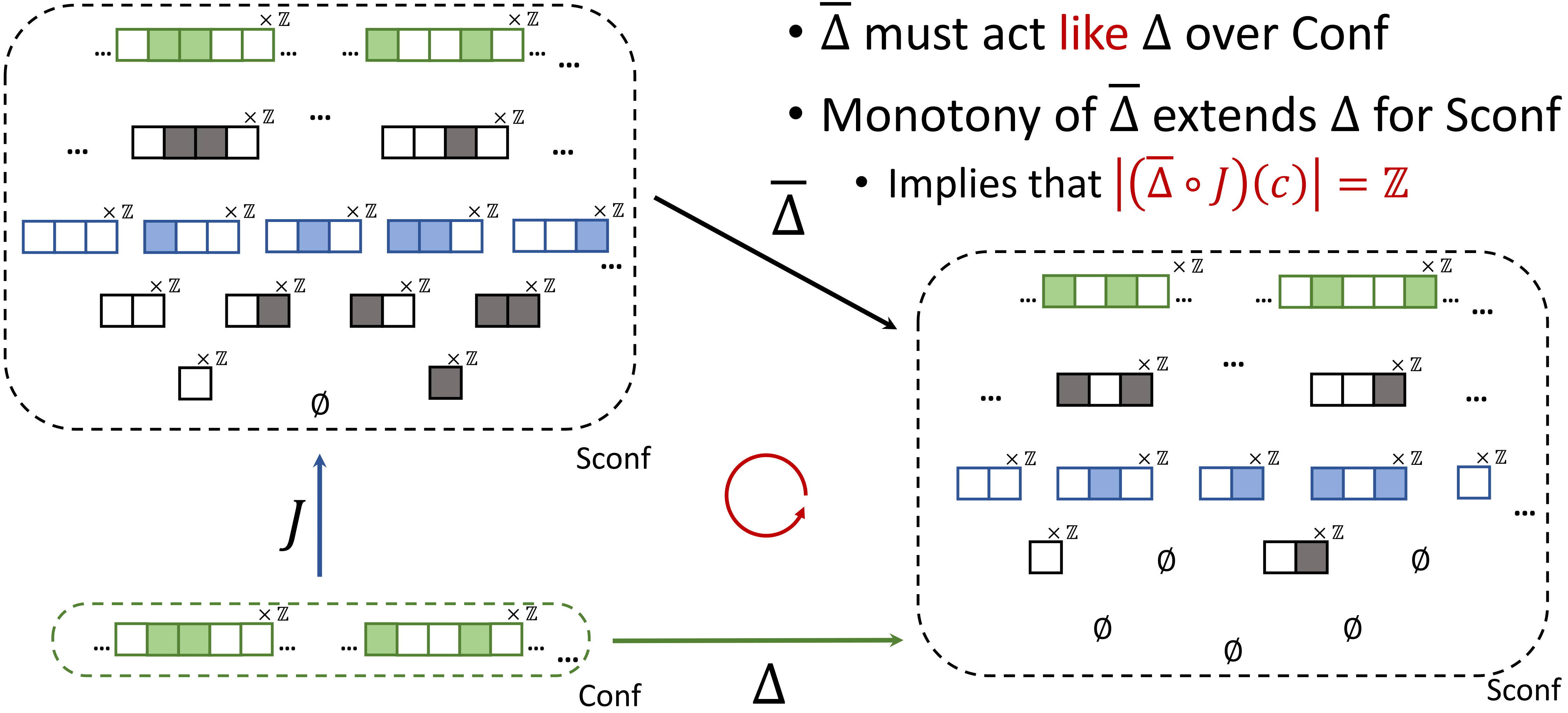
Fine as right Kan extension

- $\overline{\Delta}$ must act at most like Δ over Conf
- Monotony of $\overline{\Delta}$ extends Δ for Sconf
  - Implies that $\left|\left(\overline{\Delta} \circ J\right)(c)\right| = \mathbb{Z}$



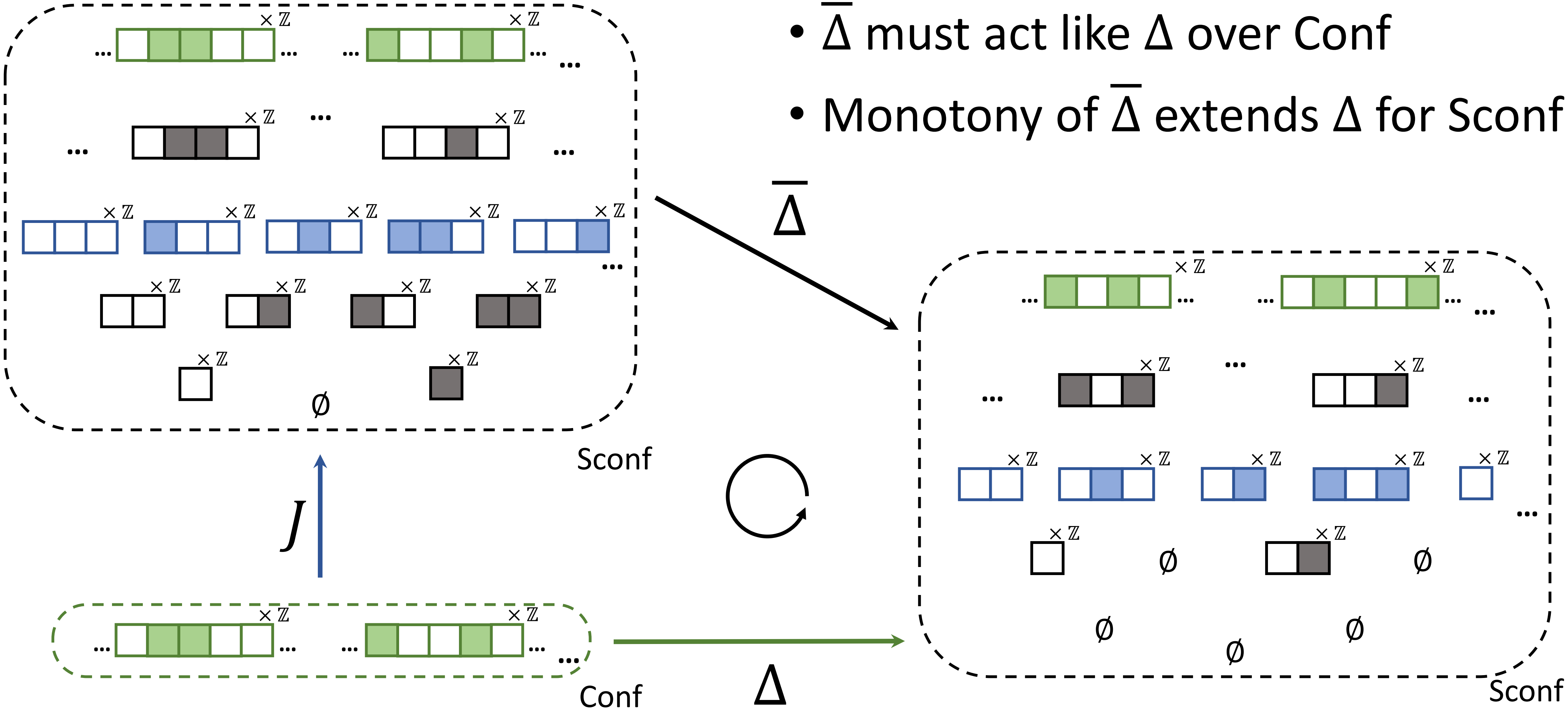
Fine as right Kan extension

- $\overline{\Delta}$ must act like Δ over Conf
- Monotony of $\overline{\Delta}$ extends Δ for Sconf
  - Implies that $\left|\left(\overline{\Delta} \circ J\right)(c)\right| = \mathbb{Z}$

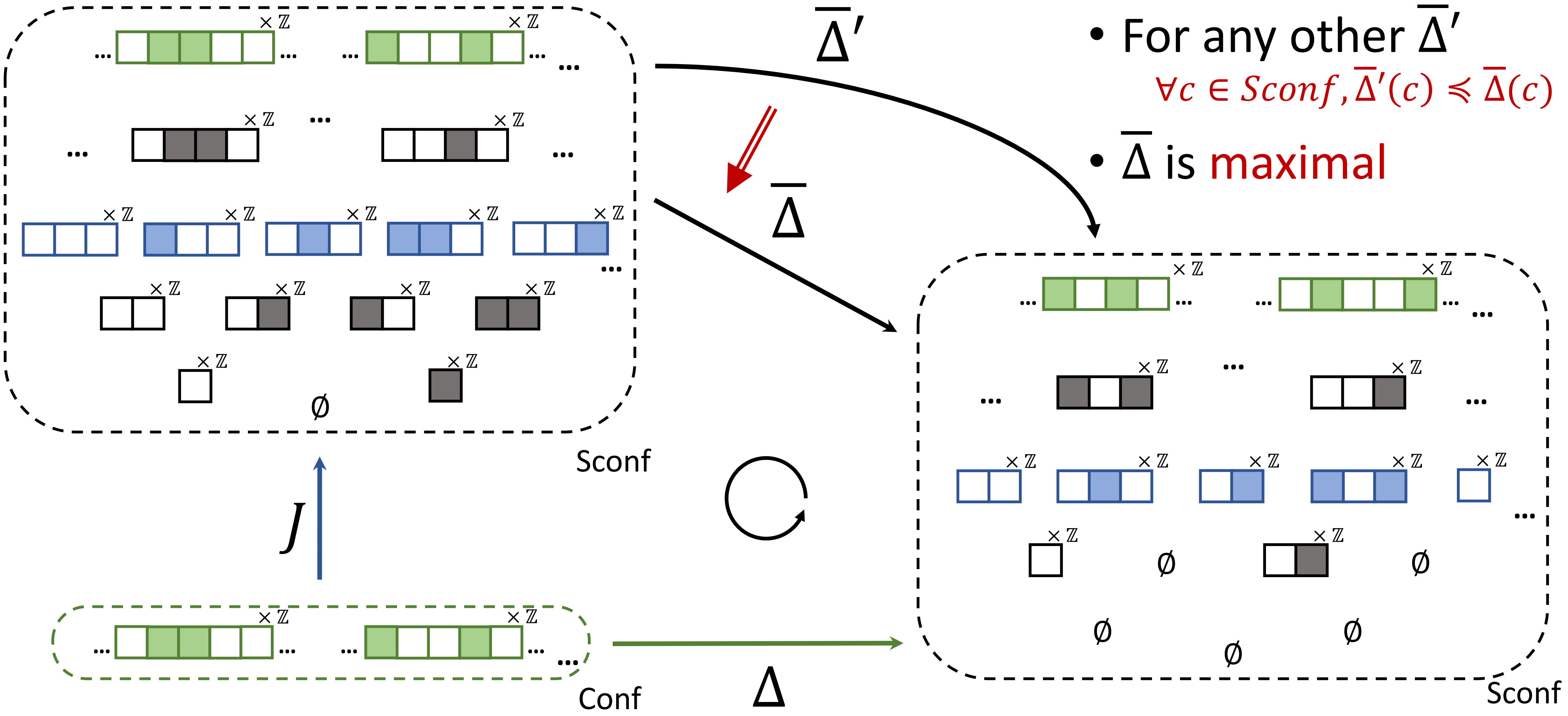


Fine as right Kan extension

- $\overline{\Delta}$ must act like Δ over Conf
- Monotony of $\overline{\Delta}$ extends Δ for Sconf



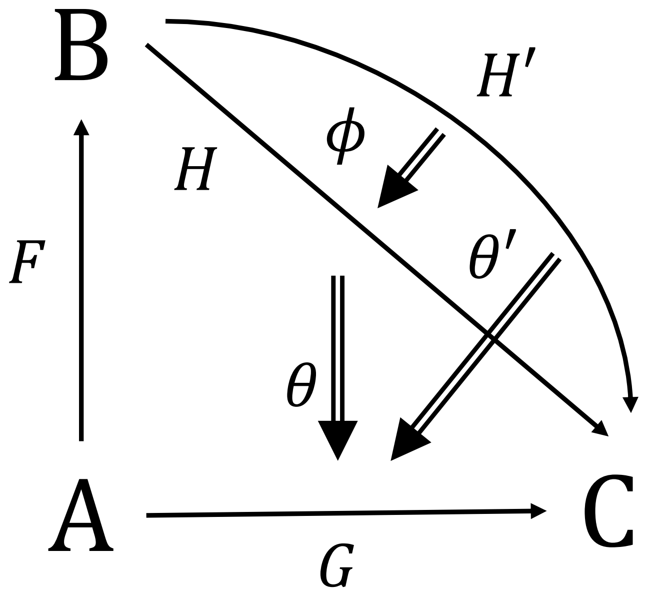
Fine as right Kan extension



- For any other $\overline{\Delta}'$
  $\forall c \in Sconf, \overline{\Delta}'(c) \preccurlyeq \overline{\Delta}(c)$
- $\overline{\Delta}$ is maximal

(Right) Kan Extensions

- (Right) Kan Extension of G along F is : $\langle H, \theta \rangle$



- $\theta: G \Rightarrow H \circ F$
- Universal :
  - Given $\langle H', \theta' \rangle$, exists unique ϕ s.t.
  - $\theta' = \theta \cdot (\phi \circ F)$

- H « **maximal** » and unique

Fine as Lan of Ran

- Fine obtained from global transition function

- Global transition is an infinite object

- Get fine extension from local rules ?
  - Computable with « bottom up » procedure

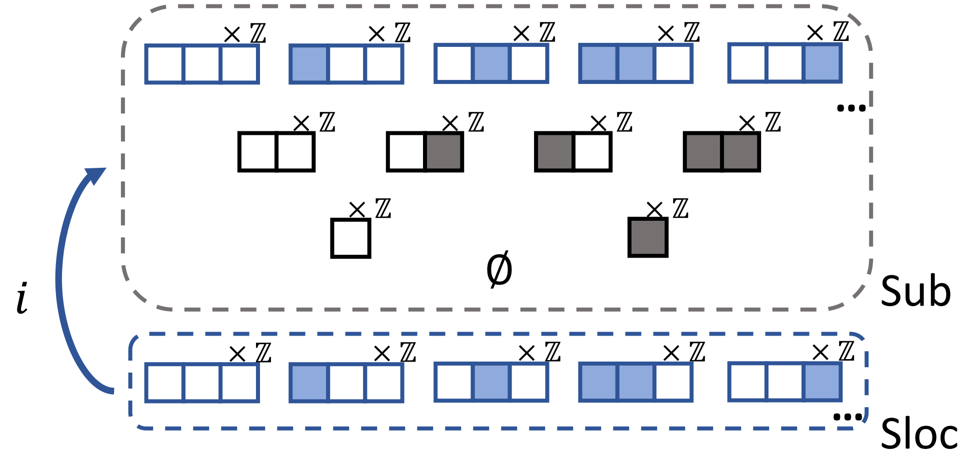
Fine as Lan of Ran

- Fine obtained from global transition function
- Global transition is an infinite object
- Get fine extension from local rules ?
  - Computable with « bottom up » procedure
  - Yes we Kan !

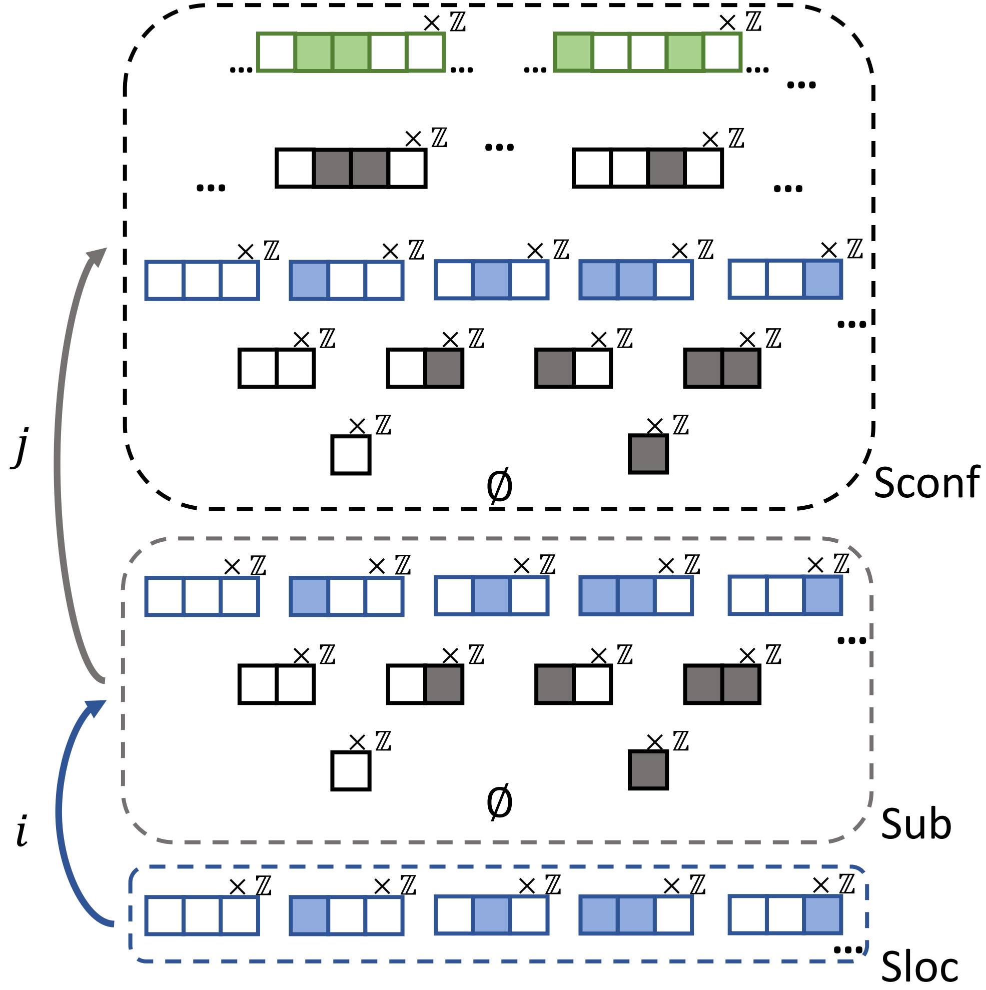
Fine as Lan of Ran

- Sub-local configurations :

$$Sub = \bigcup_{p \in \mathbb{Z},\, S \subseteq N} \{p\} \times Q^{p+S}$$



Fine as Lan of Ran



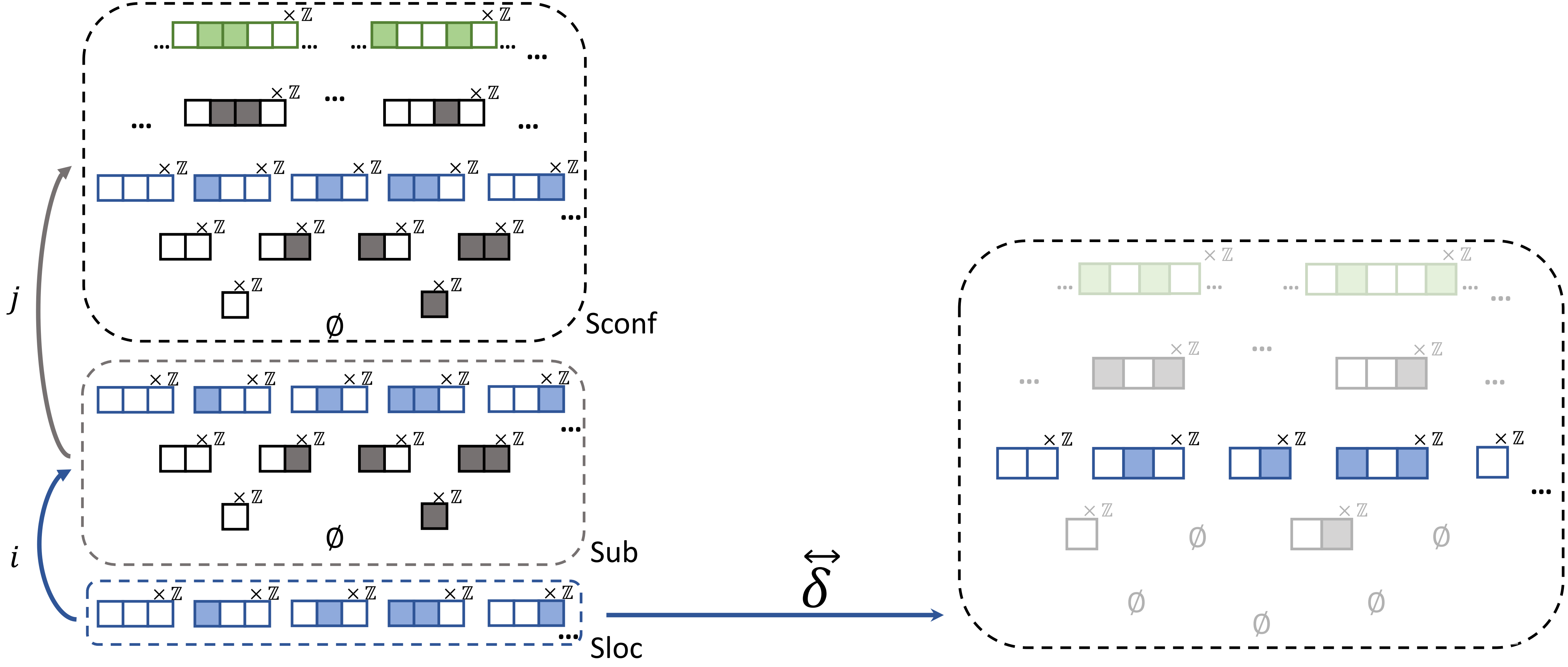
- Sub-local configurations :

$$Sub = \bigcup_{p\in\mathbb{Z},\, S\subseteq N} \{p\} \times Q^{p+S}$$

- i: $Sloc \to Sub$ inclusion

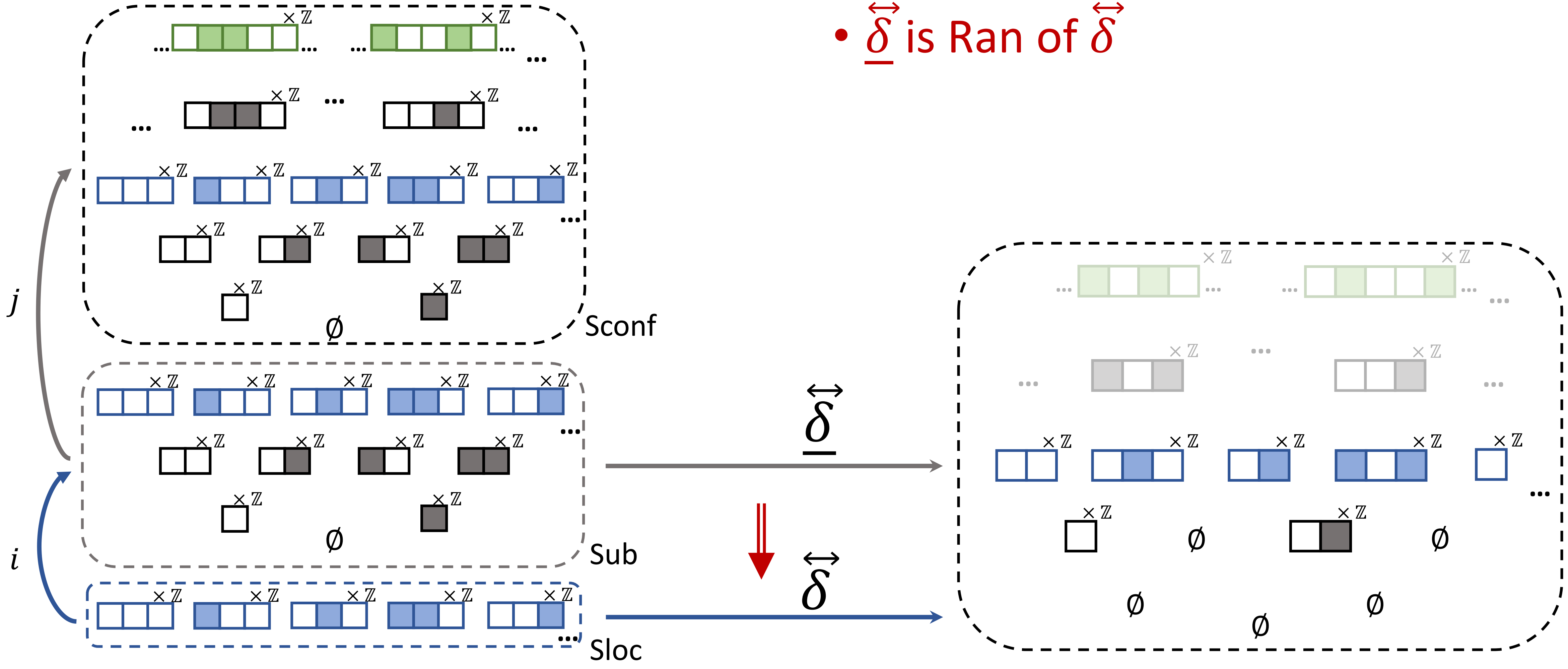
- j: $Sub \to Sconf \; := \pi_2$

Fine as Lan of Ran



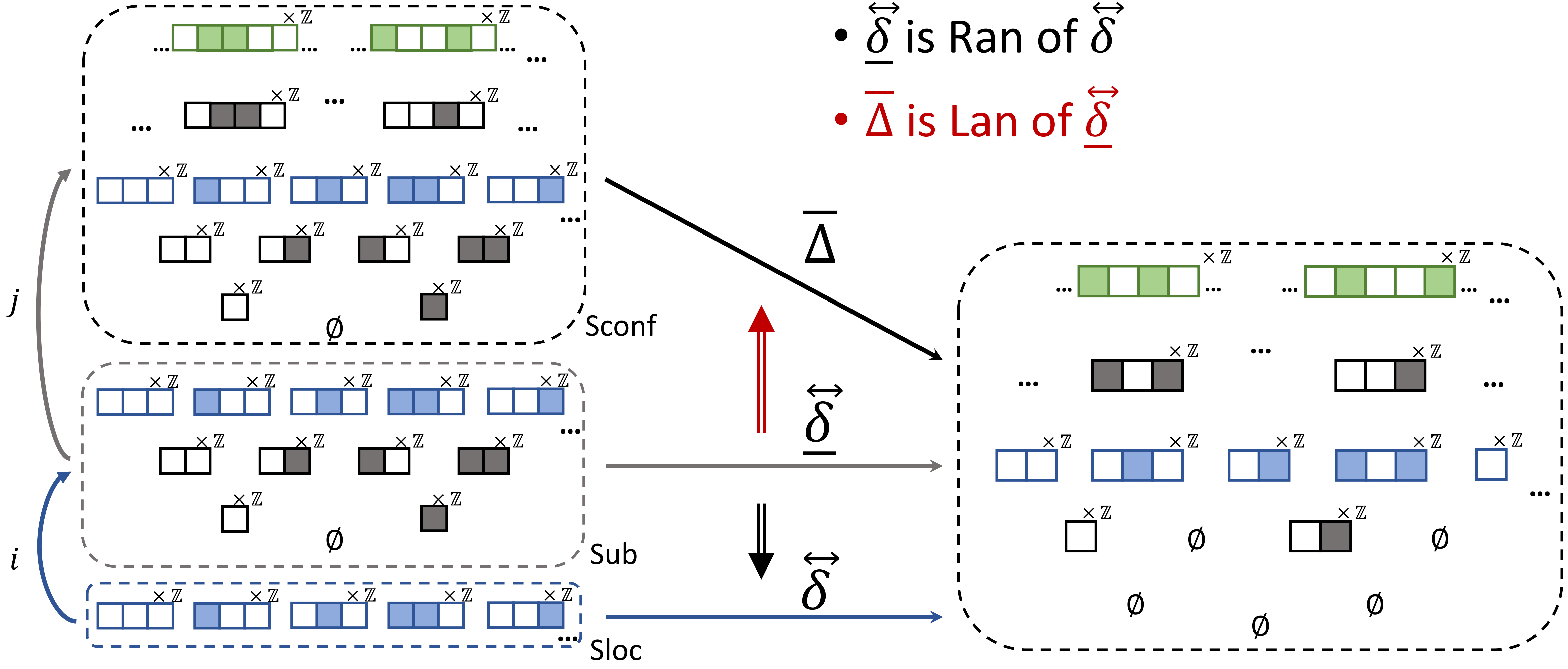
Fine as Lan of Ran

- $\underline{\overleftrightarrow{\delta}}$ is Ran of $\overleftrightarrow{\delta}$



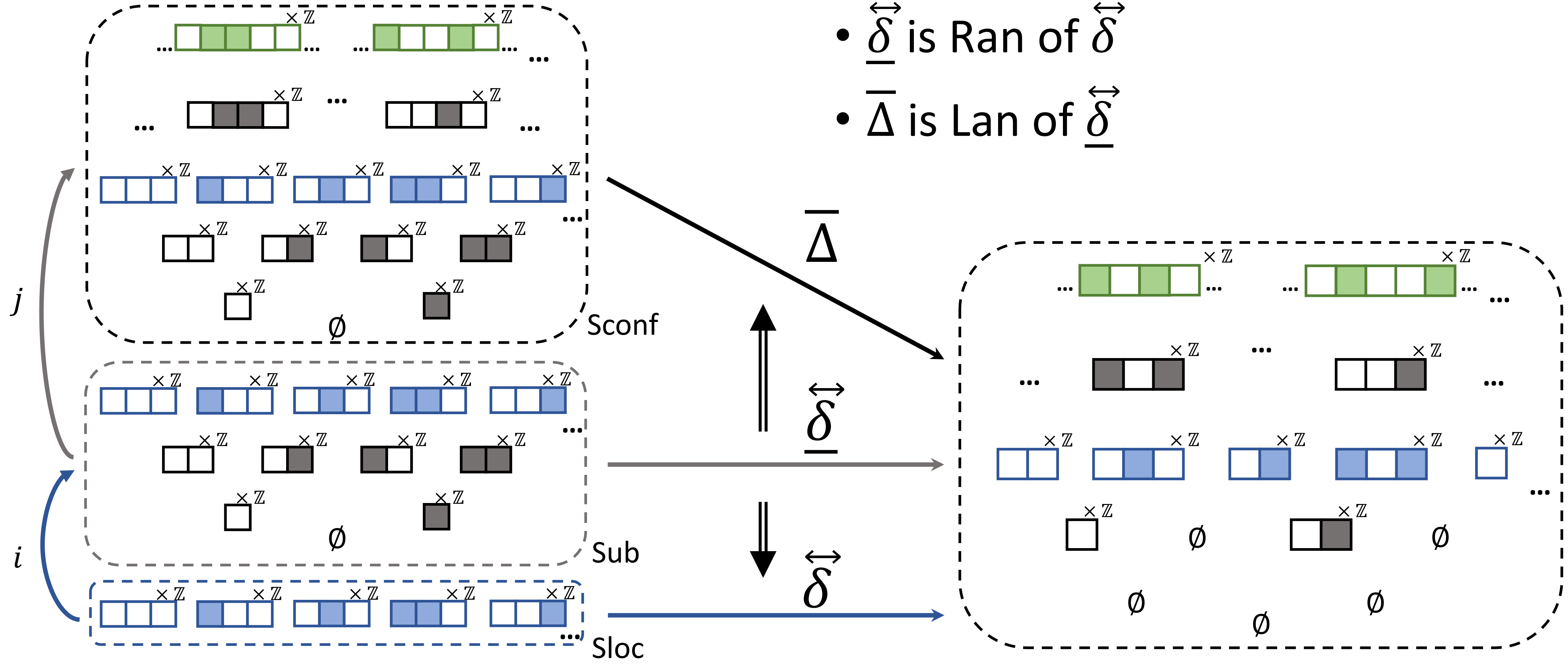
Fine as Lan of Ran

- $\underline{\overleftrightarrow{\delta}}$ is Ran of $\overleftrightarrow{\delta}$
- $\overline{\Delta}$ is Lan of $\underline{\overleftrightarrow{\delta}}$



Fine as Lan of Ran

- $\underline{\overleftrightarrow{\delta}}$ is Ran of $\overleftrightarrow{\delta}$
- $\overline{\Delta}$ is Lan of $\underline{\overleftrightarrow{\delta}}$



Summary

- Studying local vs global relationship
  - Kan extension good tool
  - Multiple ways to extend
  - Describes extensions using intermediary steps

- Cellular automata to introduce categorical concepts
  - Categories, functors, natural transformations
  - Kan extensions

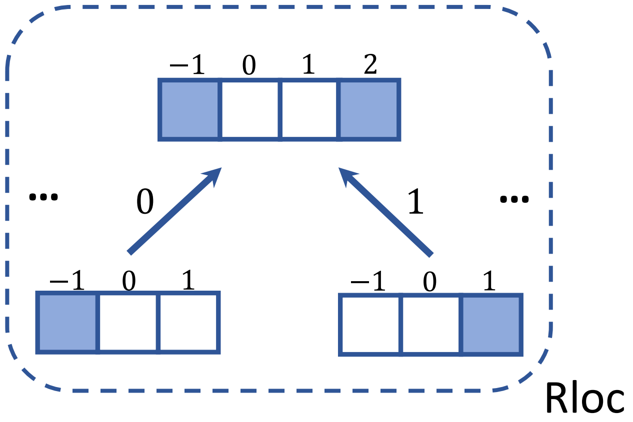
Perspectives

- Non-regular CA :
  - Different behavior at different position
  - Shifted local transition function → any function

Perspectives

- Non-regular CA

- Consider only local configuration ?
  - Sconf category, shifts are inclusions
  - Computation up to iso → shift-equivalent automata
  - Local configurations : Rloc for relative positionning



Perspectives

- Non-regular CA
- Consider only local configuration ?
  - Shift-equivalent automata

Link with Curtis–Hedlund–Lyndon ?

Perspectives

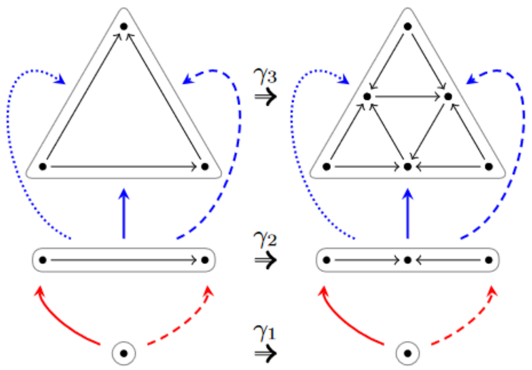
- Non-regular CA

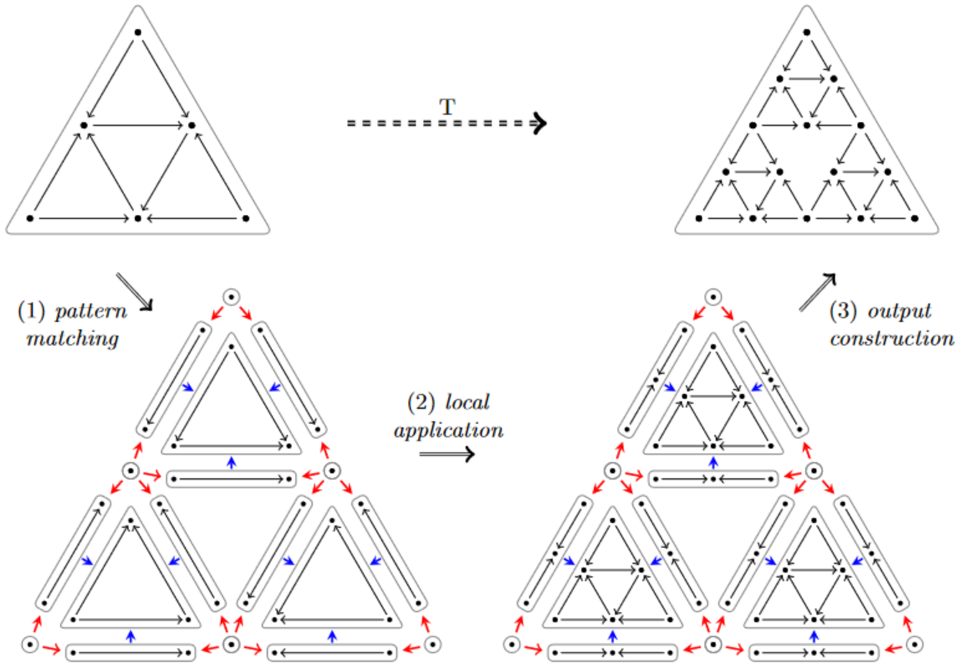
- Consider only local configuration
  - Shift-equivalent automata

- Other data structures

Perspectives

- Non-regular CA
- Consider only local configuration
  - Shift-equivalent automata
- Other data structures
  - Global Transformations
  - Words, Graphs, Trees …
  - Possibility to change the space





Perspectives

- Non-regular CA
- Consider only local configuration
  - Shift-equivalent automata
- Other data structures
- Kan extensions here are pointwise !
  - Computable