Specification and Verification of Quantitative Properties: Expressions, Logics, and Automata

Paul GASTIN - LSV - ENS Cachan

05/05/2014

Joint work with Benedikt Bollig, Benjamin Monmege and Marc Zeitoun

Model Checking and Evaluation

Model: G

Computation
Document
System

Specification: Φ

First-Order Logic
Temporal Logic
Propositional Dynamic Logic
Regular Expressions

$G \models \Phi$ Boolean

Model Checking and Evaluation

- May an *error* state be reached?
- Is there a book written by X, rented by Y?
- Does this leader election protocol permit to elect the leader?

From **Boolean** to **Quantitative** Verification

- What is the probability for an *error* state to be reached?
- How many books, written by X, have been rented by Y?
- What is the maximal delay ensuring that this leader election protocol permits the election?

$G \models \Phi$ Boolean

$[\![\Phi]\!](G)$ Quantitative

Evaluation via Automata

Model: G

Specification: Φ

Automaton: $\mathcal{A}_\Phi$

Complexity?
Small specifications
Huge models

$G \in L(\mathcal{A}_\Phi)$	$G \models \Phi$	Boolean
$[\![\mathcal{A}_\Phi]\!](G)$	$[\![\Phi]\!](G)$	Quantitative

The Success Story of Automata

An Automata-Theoretic Approach to ...

1956
... *Parsing Regular Expressions*
Kleene's theorem [Kleene 56]

... *Parsing* *Weighted* Expressions
Schützenberger's theorem
[Schützenberger 61]

1960 1961
... *Deciding MSO Over Finite Words*
[Büchi 59, Elgot 61, Trakhtenbrot 61]

1963
... *Modeling* *Probabilistic* Properties
[Rabin 63]

1986
... *Model Checking a System Against LTL*
[Vardi&Wolper 86]

2007
... *Handling* *Weighted MSO Properties*
[Droste&Gastin 07]

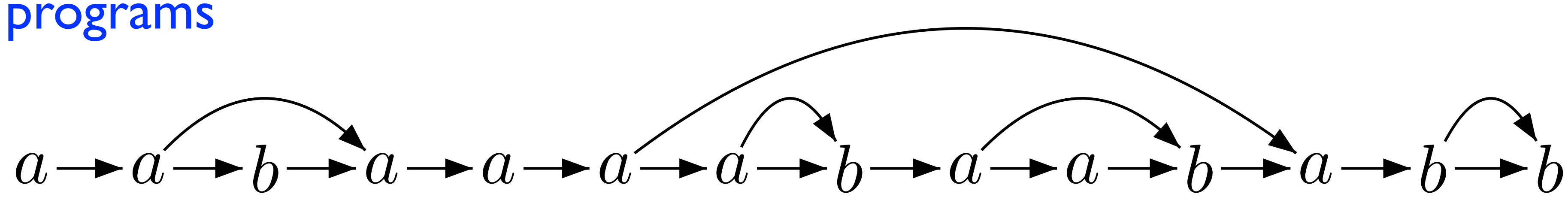
MODELS

Various Models

- Words $a \to a \to b \to a \to a \to a \to a \to b \to a \to a \to b \to a \to b \to b$

  Computations of sequential programs

- Nested Words

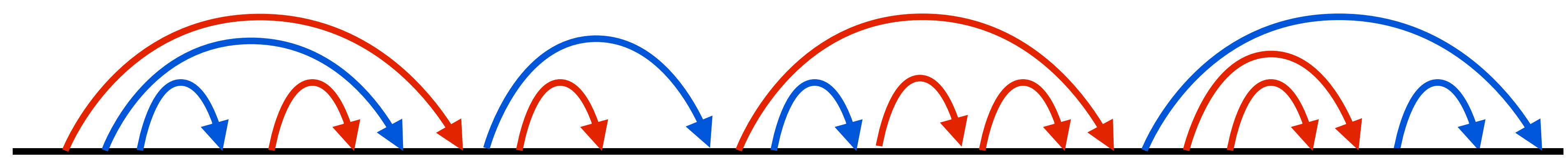


  Computations of recursive programs
  XML documents

```
proc f ()
{ ... }

proc g ()
{ ... }

main (n)
{
  i=0;
  while i<n do
    if i odd then f() else g()
    i++
  done
}
```



Various Models

- Words $a \to a \to b \to a \to a \to a \to a \to b \to a \to a \to b \to a \to b \to b$

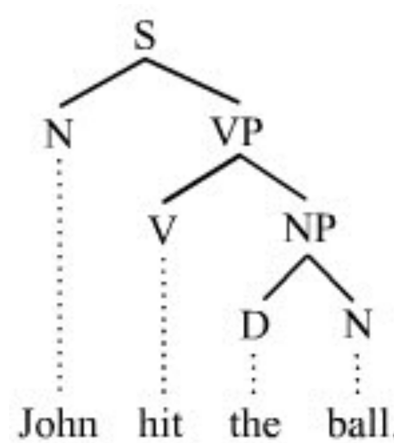
  Computations of sequential programs

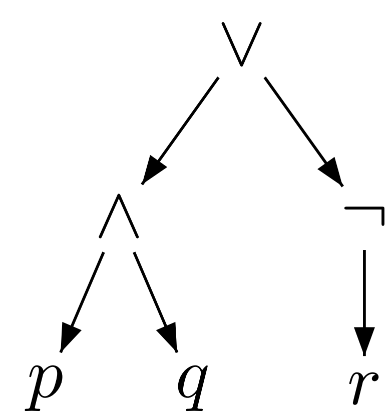
- Nested Words $a \to a \to b \to a \to a \to a \to a \to b \to a \to a \to b \to a \to b \to b$

  Computations of recursive programs
  XML documents

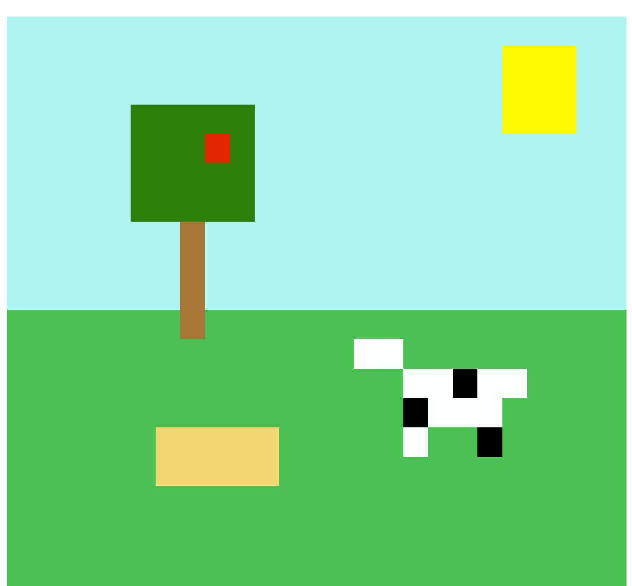
- Ranked Trees

  Expressions, Formulas, Parse trees, ...





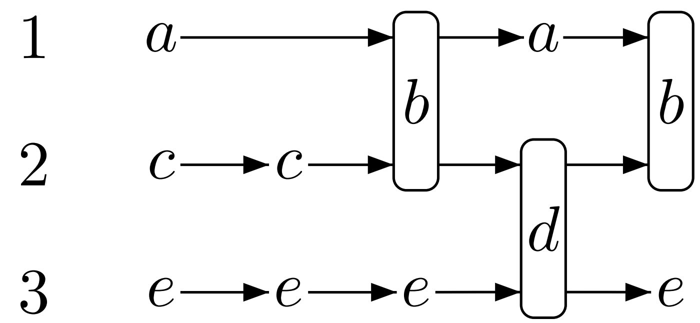
- Pictures

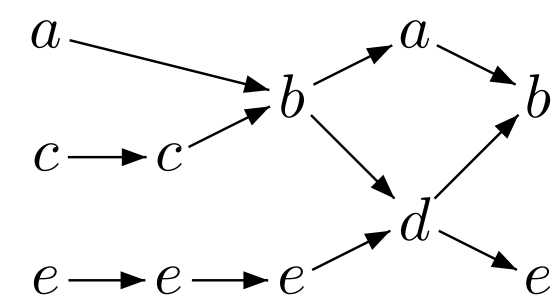


Various Models

- **Mazurkiewicz Traces**

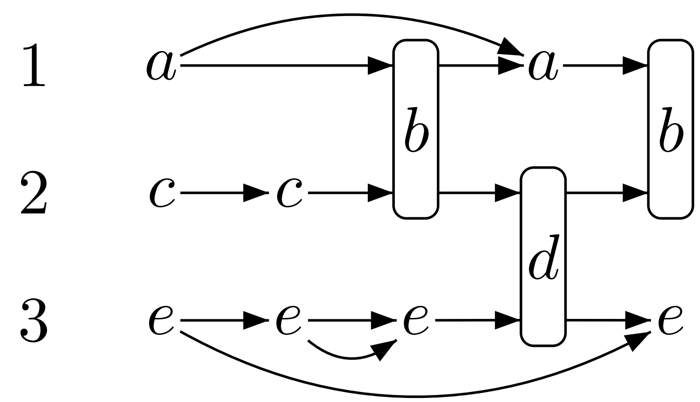
  Computations of concurrent programs
  Communication by Rendez-Vous





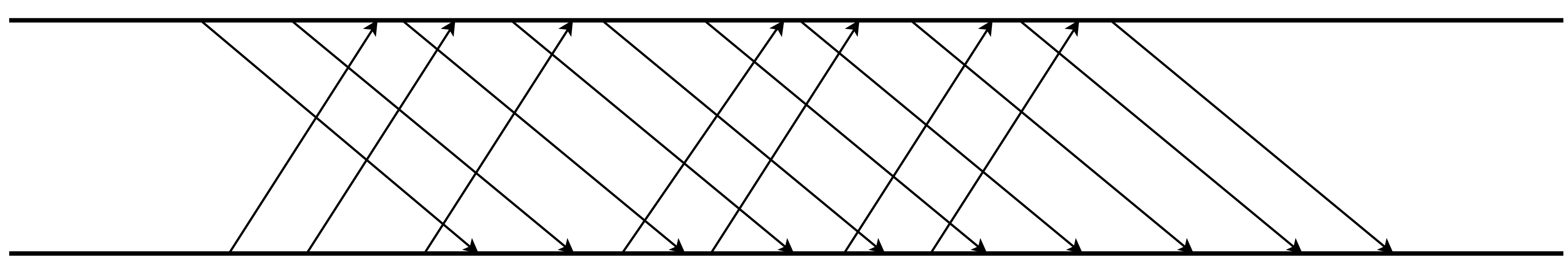
- **Traces with Nestings**

  Concurrent and Recursive Programs
  Communication by Rendez-Vous



- **Message Sequence Charts**

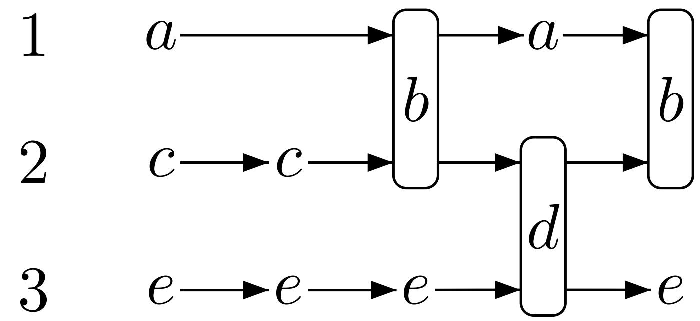
  Communication by FIFO channels

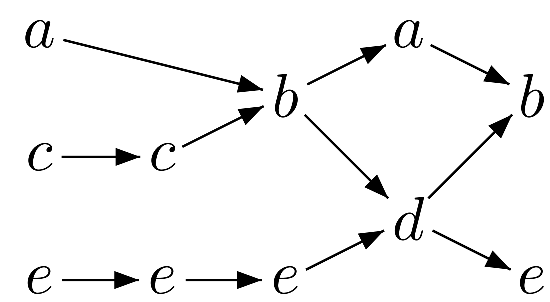


Various Models

- **Mazurkiewicz Traces**

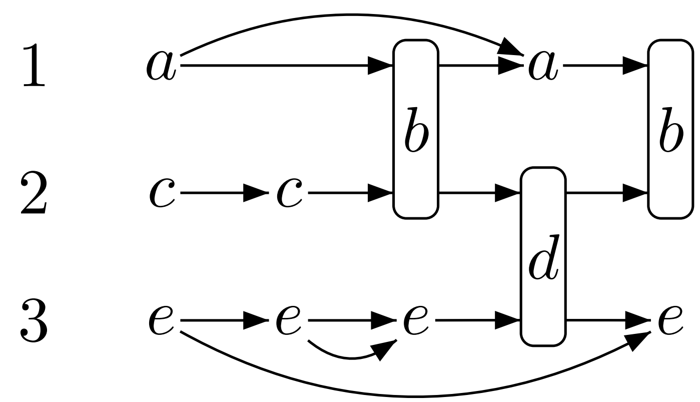
  Computations of concurrent programs
  Communication by Rendez-Vous





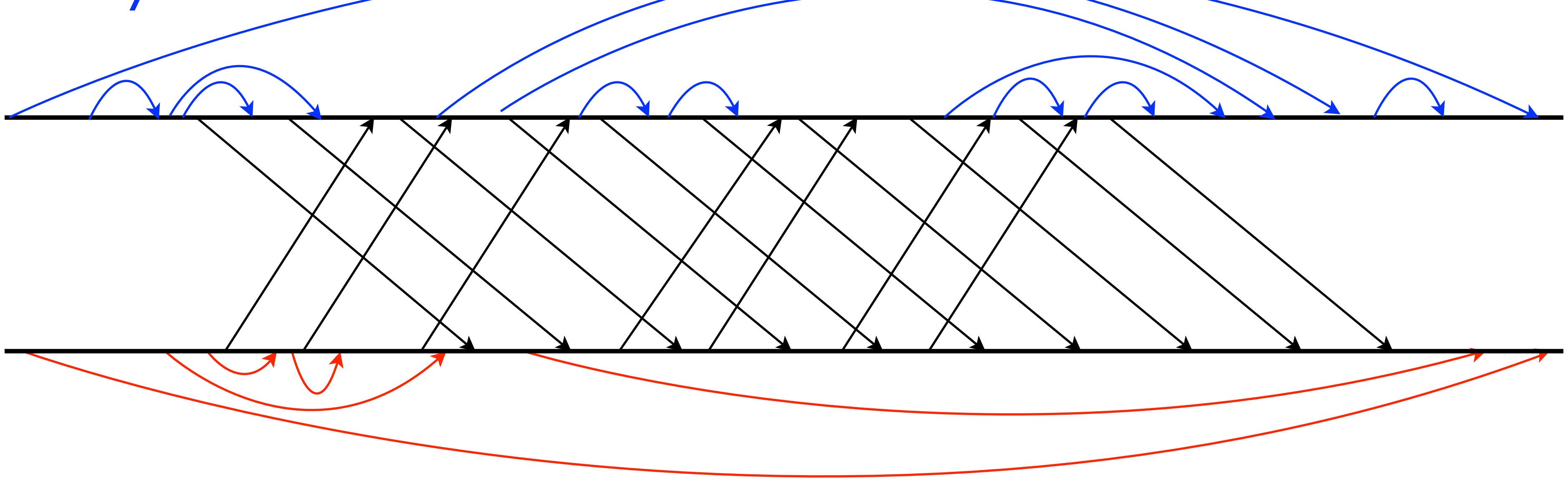
- **Traces with Nestings**

  Concurrent and Recursive Programs
  Communication by Rendez-Vous

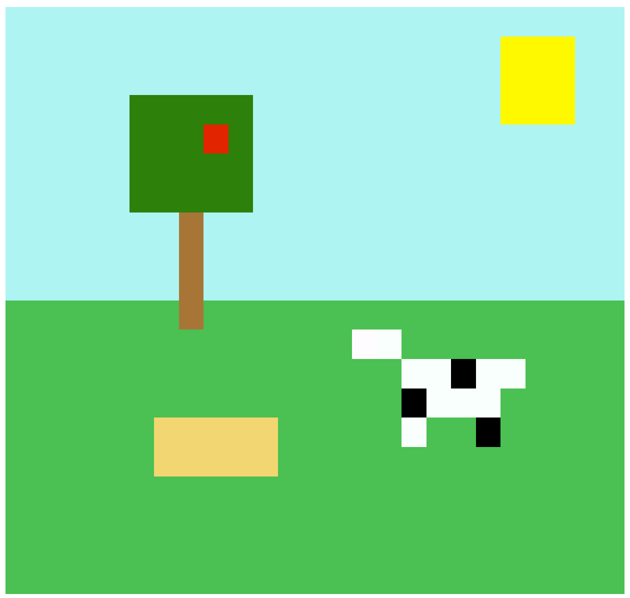


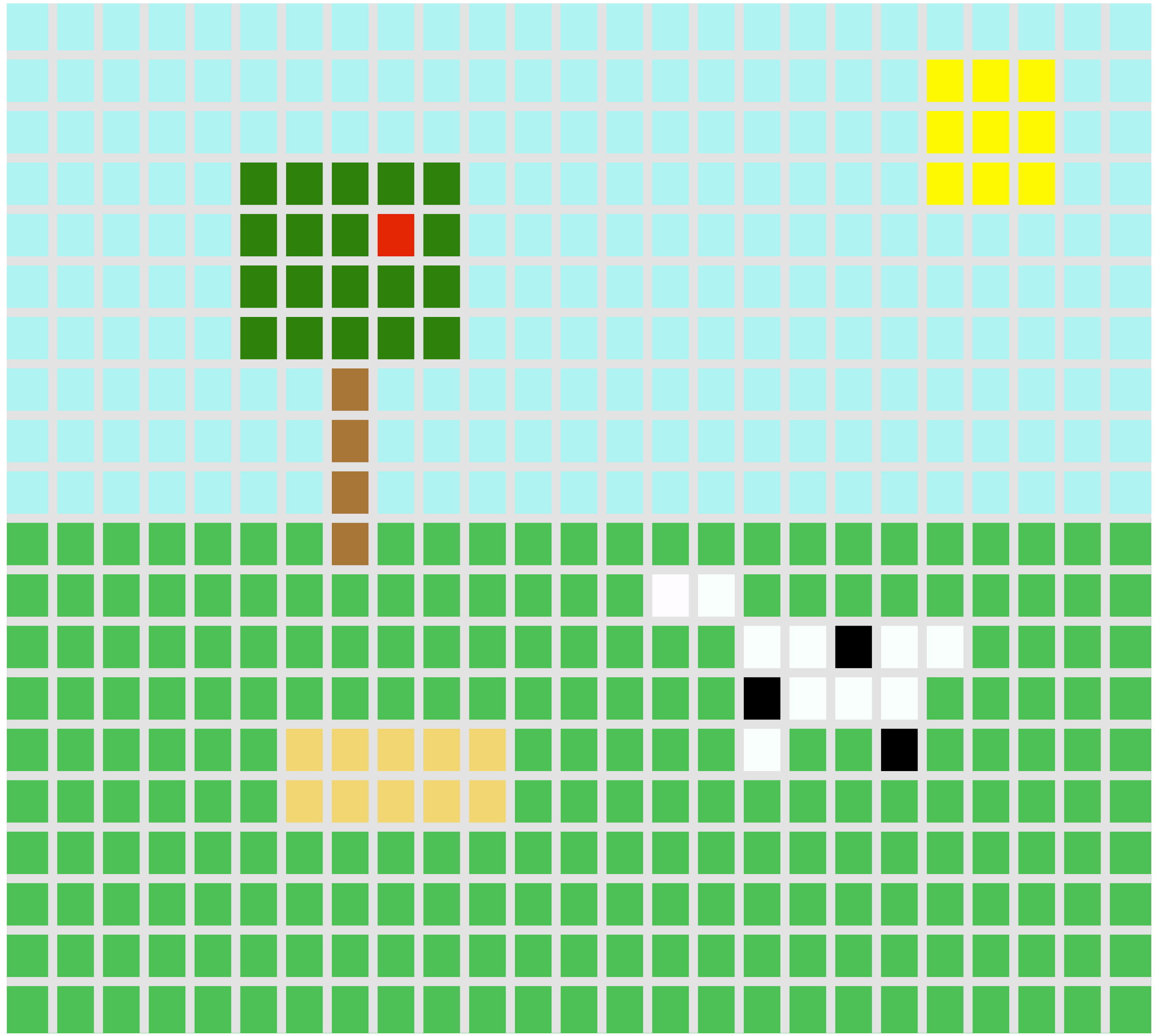
- **Message Sequence Charts with Nestings**

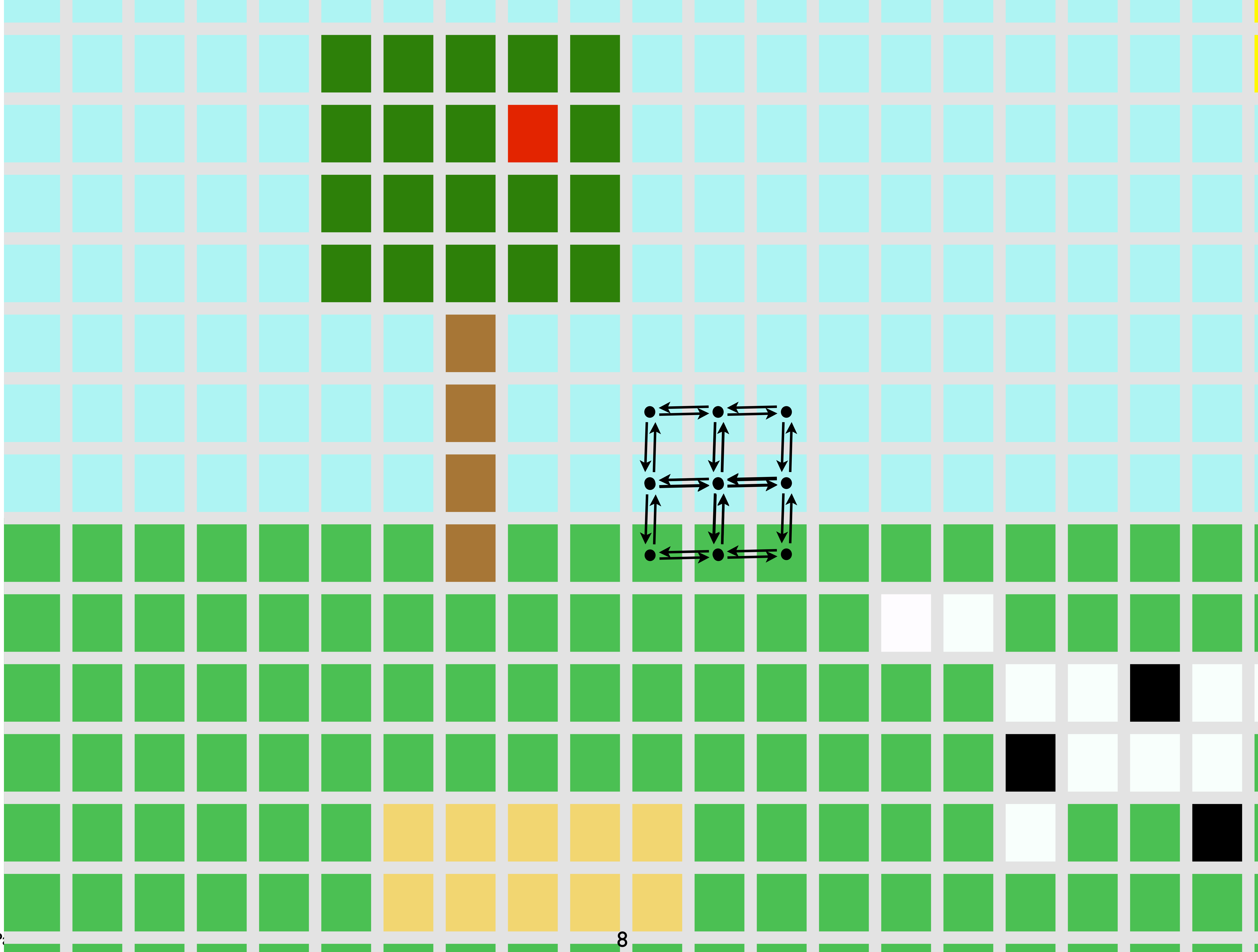
  Communication by FIFO channels



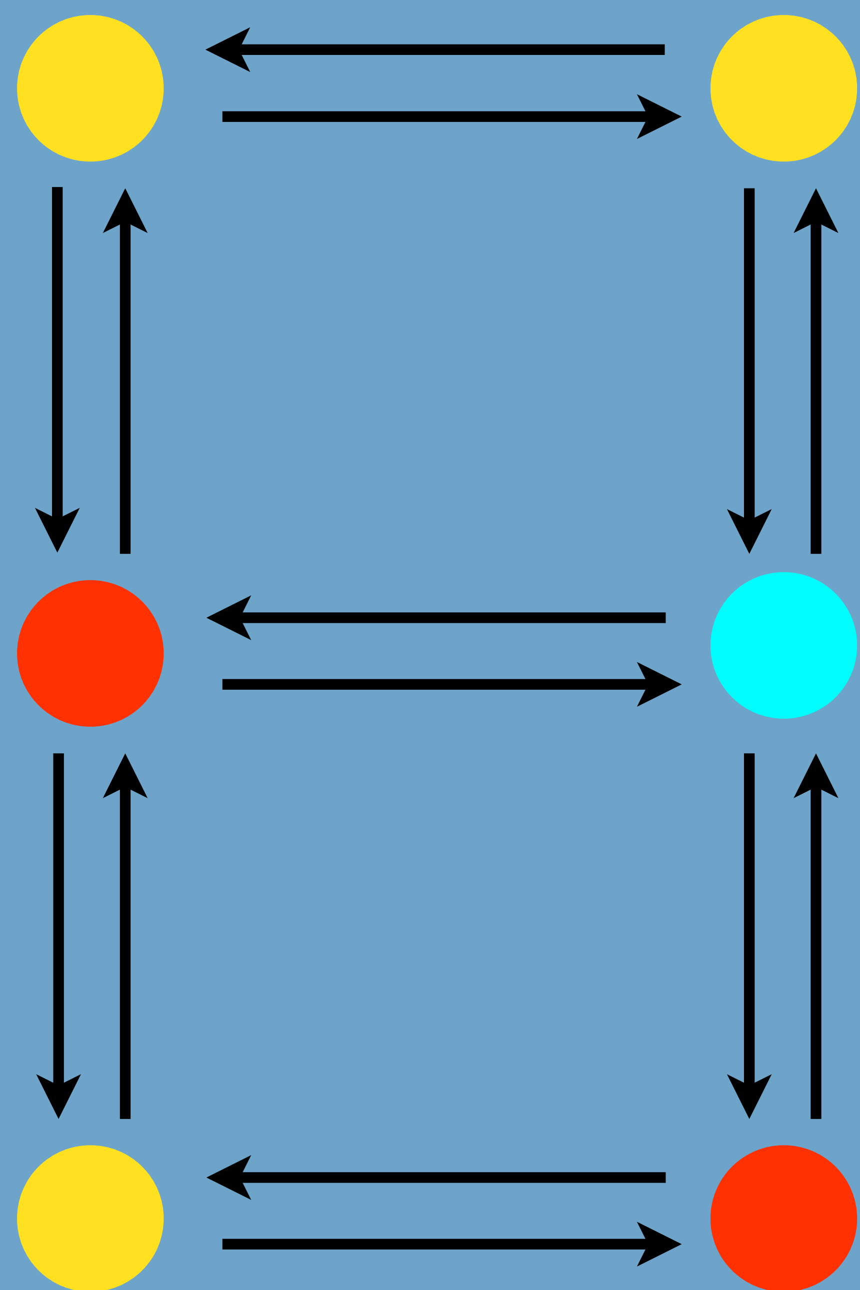
Modeling a picture as a graph







Graphs



$$D = \{\rightarrow, \downarrow\} \cup \{\leftarrow, \uparrow\}$$

$$G = (V, (E_d)_{d \in D}, \lambda)$$

V	set of vertices
λ	labels of vertices
D	set of directions
E_d	set of *d*-edges

deterministic
(hence bounded degree)

Various Models

- Words $a \to a \to b \to a \to a \to a \to a \to b \to a \to a \to b \to a \to b \to b$

  Computations of sequential programs

  $D = \{\to, \leftarrow\}$

- Nested Words $a \to a \to b \to a \to a \to a \to a \to b \to a \to a \to b \to a \to b \to b$

  Computations of recursive programs

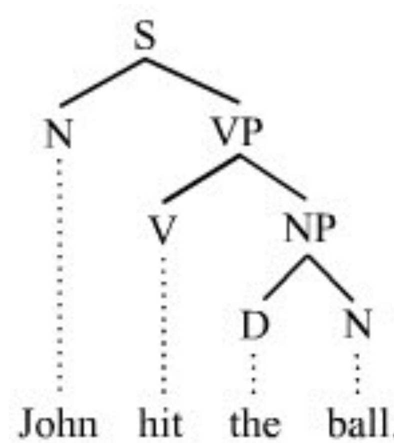
  XML documents

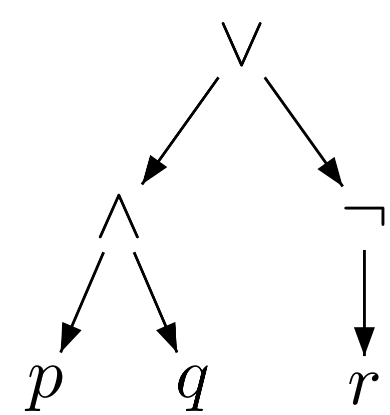
  $D = \{\to, \leftarrow, \curvearrowright, \curvearrowleft\}$

- Ranked Trees

  Expressions, Formulas, Parse trees, ...

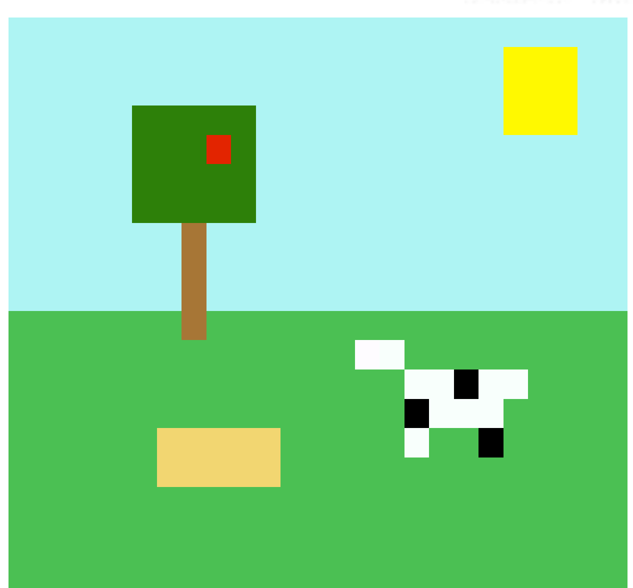
  $D = \{\downarrow_1, \uparrow_1, \downarrow_2, \uparrow_2\}$





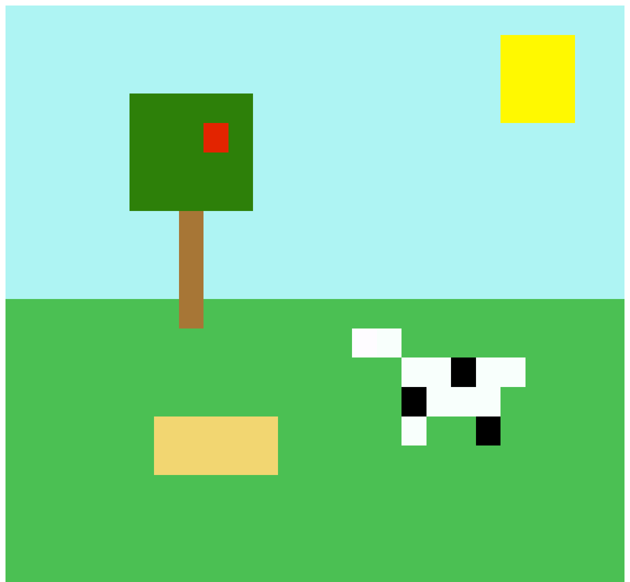
- Pictures

  $D = \{\to, \leftarrow, \downarrow, \uparrow\}$



Weighted
First-Order Logic

Logical Specifications: Query Examples



Is there a line of green pixels?

$$\exists x \forall y \big[(R^\star_\rightarrow(x,y) \vee R^\star_\rightarrow(y,x)) \Rightarrow P_{\blacksquare}(y)\big]$$

How many lines of green pixels are there?

What is the size of the picture?

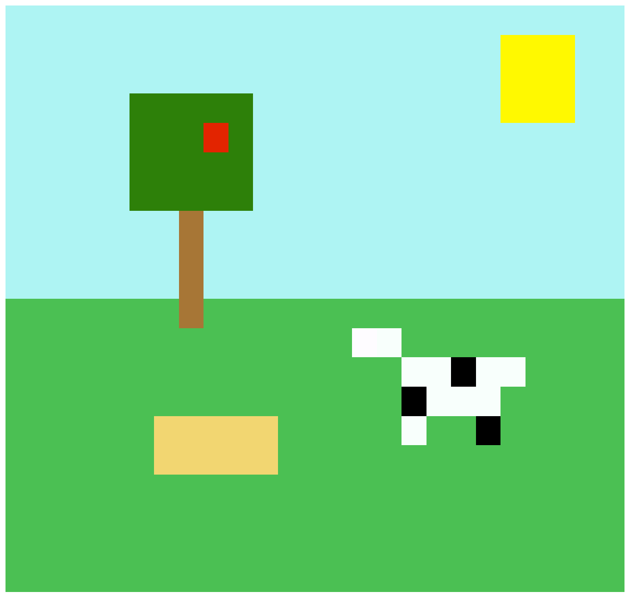
Boolean fragment: first-order logic

$$\varphi ::= \top \mid P_a(x) \mid x = y \mid R_d(x,y) \mid R_d^+(x,y) \mid \neg\varphi \mid \varphi \vee \varphi \mid \exists x\, \varphi$$

with $a \in A \cup \{\triangleright\}$, $d \in D$.

What is the size of the biggest monochromatic rectangle?

Logical Specifications: Query Examples



Is there a line of green pixels?

$$\exists x \forall y \big[(R^\star_\rightarrow(x,y) \vee R^\star_\rightarrow(y,x)) \Rightarrow P_{\blacksquare}(y)\big]$$

How many lines of green pixels are there?

$$\sum_x (\neg \exists y\, R_\rightarrow(y,x)) \wedge (\forall y\, R^*_\rightarrow(x,y) \Rightarrow P_{\blacksquare}(y))\ ?\ 1 : 0$$

What is the size of the picture?

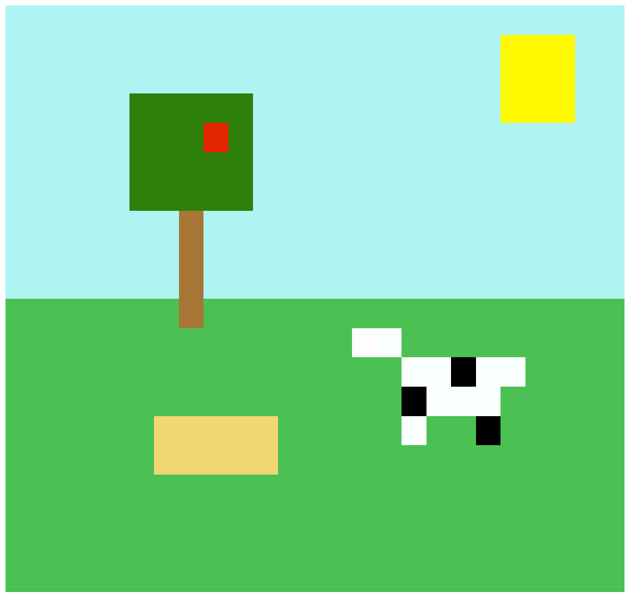
Weighted fragment: first-order logic

$$\Phi ::= s \mid \varphi ?\, \Phi : \Phi \mid \Phi \oplus \Phi \mid \Phi \otimes \Phi \mid \bigoplus_x \Phi \mid \bigotimes_x \Phi$$

with $\varphi \in \mathrm{bFO}$ and $s \in \mathbb{S}$.

What is the size of the biggest monochromatic rectangle?

Logical Specifications: Query Examples



Is there a line of green pixels?

$(\mathbb{N}, +, \times, 0, 1)$

$$\exists x \forall y \big[(R^\star_{\rightarrow}(x,y) \vee R^\star_{\rightarrow}(y,x)) \Rightarrow P_{\blacksquare}(y) \big]$$

How many lines of green pixels are there?

$$\sum_x (\neg \exists y\, R_{\rightarrow}(y,x)) \wedge (\forall y\, R^*_{\rightarrow}(x,y) \Rightarrow P_{\blacksquare}(y))\ ?\ 1 : 0$$

What is the size of the picture?

$$\Big(\sum_x \neg \exists y\, R_{\rightarrow}(y,x) \Big) \times \Big(\sum_x \neg \exists y\, R_{\downarrow}(y,x) \Big)$$

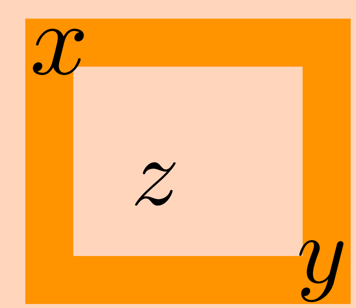
What is the color with maximum number of pixels?

$(\mathbb{N} \cup \{-\infty\}, \max, +, -\infty, 0)$

$$\max \Big(\sum_x P_{\blacksquare}(x), \sum_x P_{\blacksquare}(x), \ldots, \sum_x P_{\blacksquare}(x) \Big)$$

What is the size of the biggest monochromatic rectangle?

$$\max_{x,y} \left[\varphi_{\mathrm{mono}}(x,y)\ ?\ \left(\sum_z \varphi_{\mathrm{rect}}(x,y,z)\ ?\ 1 : 0 \right) : -\infty \right]$$



Full Weighted First-order Logic

Boolean fragment

Path using d-edges

$$\varphi ::= \top \mid P_a(x) \mid x = y \mid R_d(x,y) \mid R_d^+(x,y) \mid \neg\varphi \mid \varphi \vee \varphi \mid \exists x\, \varphi$$

$$\Phi ::= s \mid \varphi ?\, \Phi : \Phi \mid \Phi \oplus \Phi \mid \Phi \otimes \Phi \mid \bigoplus_x \Phi \mid \bigotimes_x \Phi$$

with $a \in A \cup \{\triangleright\}$, $d \in D$, $s \in \mathbb{S}$.

For trees we may use $\downarrow^+$

Quantitative formulas

if .. then .. else ..

Possible Variants:

change Boolean power (AP, MSO, ...)

add more quantitative power (wMSO, wTC...)

Quantitative Semantics

$$[\![s]\!](G,\sigma) = s$$

Weights from a semiring
$(\mathbb{S}, +, \times, 0, 1)$

$$[\![\varphi?\,\Phi_1 : \Phi_2]\!](G,\sigma) = \begin{cases} [\![\Phi_1]\!](G,\sigma) & \text{if } G,\sigma \models \varphi \\ [\![\Phi_2]\!](G,\sigma) & \text{otherwise} \end{cases}$$

$$[\![\Phi_1 \oplus \Phi_2]\!](G,\sigma) = [\![\Phi_1]\!](G,\sigma) + [\![\Phi_2]\!](G,\sigma)$$

Valuation of free variables
$\sigma : \mathcal{V} \to \mathrm{pos}(G)$

$$[\![\Phi_1 \otimes \Phi_2]\!](G,\sigma) = [\![\Phi_1]\!](G$$

$$[\![\textstyle\bigoplus_x \Phi]\!](G,\sigma) = \sum_{k \in \mathrm{pos}(G)} [\![\Phi]\!](G,\sigma[x \mapsto k])$$

$$[\![\textstyle\bigotimes_x \Phi]\!](G,\sigma) = \prod_{k \in \mathrm{pos}(G)} [\![\Phi]\!](G,\sigma[x \mapsto k])$$

Quantitative Semantics

Weights from a semiring $(\mathbb{S}, +, \times, 0, 1)$

$$[\![s]\!](G, \sigma) = s$$

$$[\![\varphi ? \, \Phi_1 : \Phi_2]\!](G, \sigma) = \begin{cases} [\![\Phi_1]\!](G, \sigma) & \text{if } G, \sigma \models \varphi \\ [\![\Phi_2]\!](G, \sigma) & \text{otherwise} \end{cases}$$

$$[\![\Phi_1 \oplus \Phi_2]\!](G, \sigma) = [\![\Phi_1]\!](G, \sigma) + [\![\Phi_2]\!](G, \sigma)$$

$$[\![\Phi_1 \otimes \Phi_2]\!](G, \sigma) = [\![\Phi_1]\!](G, \sigma) \times [\![\Phi_2]\!](G, \sigma)$$

$$[\![\textstyle\bigoplus_x \Phi]\!](G, \sigma) = \sum_{k \in \mathrm{pos}(G)} [\![\Phi]\!](G, \sigma[x \mapsto k])$$

$$[\![\textstyle\bigotimes_x \Phi]\!](G, \sigma) = \prod_{k \in \mathrm{pos}(G)} [\![\Phi]\!](G, \sigma[x \mapsto k])$$

Weight Domains: Semirings

$$(\mathbb{S}, +, \times, 0, 1)$$

associative and commutative, with neutral element 0

associative, with neutral element 1, distributive over addition

zero of the multiplicative operation

$$(\mathbb{R}, +, \times, 0, 1)$$
$$(\mathbb{Q}, +, \times, 0, 1)$$
$$(\mathbb{Z}, +, \times, 0, 1)$$
$$(\mathbb{N}, +, \times, 0, 1)$$

$$(\{0,1\}, \vee, \wedge, 0, 1)$$
$$([0,1], \max, \min, 0, 1)$$
$$(\mathbb{R} \cup \{-\infty, +\infty\}, \max, \min, -\infty, +\infty)$$

$$\mathbb{B}\langle\langle A^* \rangle\rangle = (\mathcal{P}(A^*), \cup, \cdot, \emptyset, \{\varepsilon\})$$
$$\mathbb{N}\langle\langle A^* \rangle\rangle = (\mathbb{N}^{A^*}, +, \cdot, \mathbf{0}, \mathbf{1}_\varepsilon)$$

$$(\mathbb{R} \cup \{+\infty\}, \min, +, +\infty, 0)$$
$$(\mathbb{R} \cup \{-\infty\}, \max, +, -\infty, 0)$$

Restricted Weighted MSO Logic

Boolean fragment

$\varphi ::= \top \mid P_a(x) \mid x = y \mid R_d(x,y) \mid R_d^+(x,y) \mid \neg\varphi \mid \varphi \vee \varphi \mid \exists x\, \varphi \mid x \in X \mid \exists X\, \varphi$

$\Phi ::= s \mid \varphi ?\, \Phi : \Phi \mid \Phi \oplus \Phi \mid \quad \mid \bigoplus_x \Phi \mid \bigotimes_x \Psi \mid \bigoplus_X \Phi$

with $a \in A \cup \{\triangleright\}$, $d \in D$, $s \in \mathbb{S}$.

Quantitative formulas

Almost boolean

$\Psi ::= s \mid \varphi ?\, \Psi : \Psi$

Droste&Gastin 07

Restricted wMSO = weighted automata

Restricted Weighted MSO Logic

Boolean fragment

commutative semiring

$\varphi ::= \top \mid P_a(x) \mid x = y \mid R_d(x,y) \mid R_d^*(x,y) \mid \neg\varphi \mid \varphi \vee \varphi \mid \exists x\, \varphi \mid x \in X \mid \exists X\, \varphi$

$\Phi ::= s \mid \varphi ?\, \Phi : \Phi \mid \Phi \oplus \Phi \mid \Phi \otimes \Phi \mid \bigoplus_x \Phi \mid \bigotimes_x \Psi \mid \bigoplus_X \Phi$

with $a \in A \cup \{\triangleright\}$, $d \in D$, $s \in \mathbb{S}$.

Quantitative formulas

Almost boolean

$\Psi ::= s \mid \varphi ?\, \Psi : \Psi$

Droste&Gastin 07

Restricted wMSO = weighted automata

Semirings

vs

Average

discounted sum

. . .

Uninterpreted Weights

Do not interpret weights from W

Semiring of multi-sets of weight sequences

$$\mathbb{S} = \mathbb{N}\langle\langle W^* \rangle\rangle = (\mathbb{N}^{W^*}, +, \cdot, \mathbf{0}, \mathbf{1}_\varepsilon)$$

Restricted wMSO = weighted automata
Full wFO = pebble weighted automata

Uninterpreted Weights

Do not interpret weights from W

Semiring of multi-sets of weight sequences

$$\mathbb{S} = \mathbb{N}\langle\langle W^* \rangle\rangle = (\mathbb{N}^{W^*}, +, \cdot, \mathbf{0}, \mathbf{1}_\varepsilon)$$

Valuations yields multi-sets of values $\mathbb{N}\langle\langle W^* \rangle\rangle \to \mathbb{N}^S$

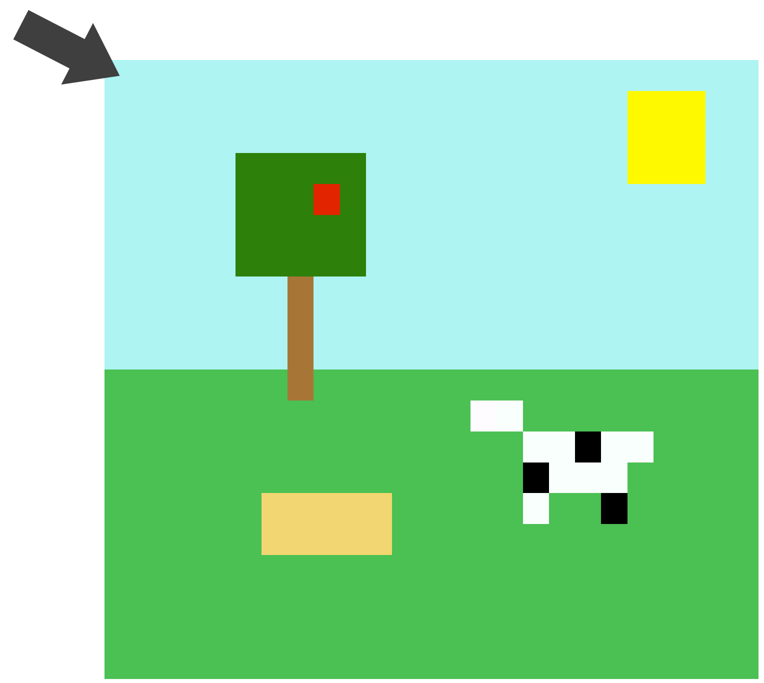
- Average: $\texttt{val}(w_1 w_2 \cdots w_n) = \frac{w_1 + w_2 + \cdots + w_n}{n}$
- Discounted sum: $\texttt{val}_\lambda(w_1 w_2 \cdots w_n) = \lambda^1 w_1 + \lambda^2 w_2 + \cdots + \lambda^n w_n$
- Cost-Reward: $\texttt{val}((r_1, c_1)(r_2, c_2) \cdots (r_n, c_n)) = \frac{r_1 + r_2 + \cdots + r_n}{c_1 + c_2 + \cdots c_n}$
- Probabilities:
  $\texttt{val}((p_1, w_1)(p_2, w_2) \cdots (p_n, w_n)) = (p_1 p_2 \cdots p_n, w_1 + w_2 + \cdots w_n)$

Aggregation of multi-sets of values $\mathbb{N}^S \to T$

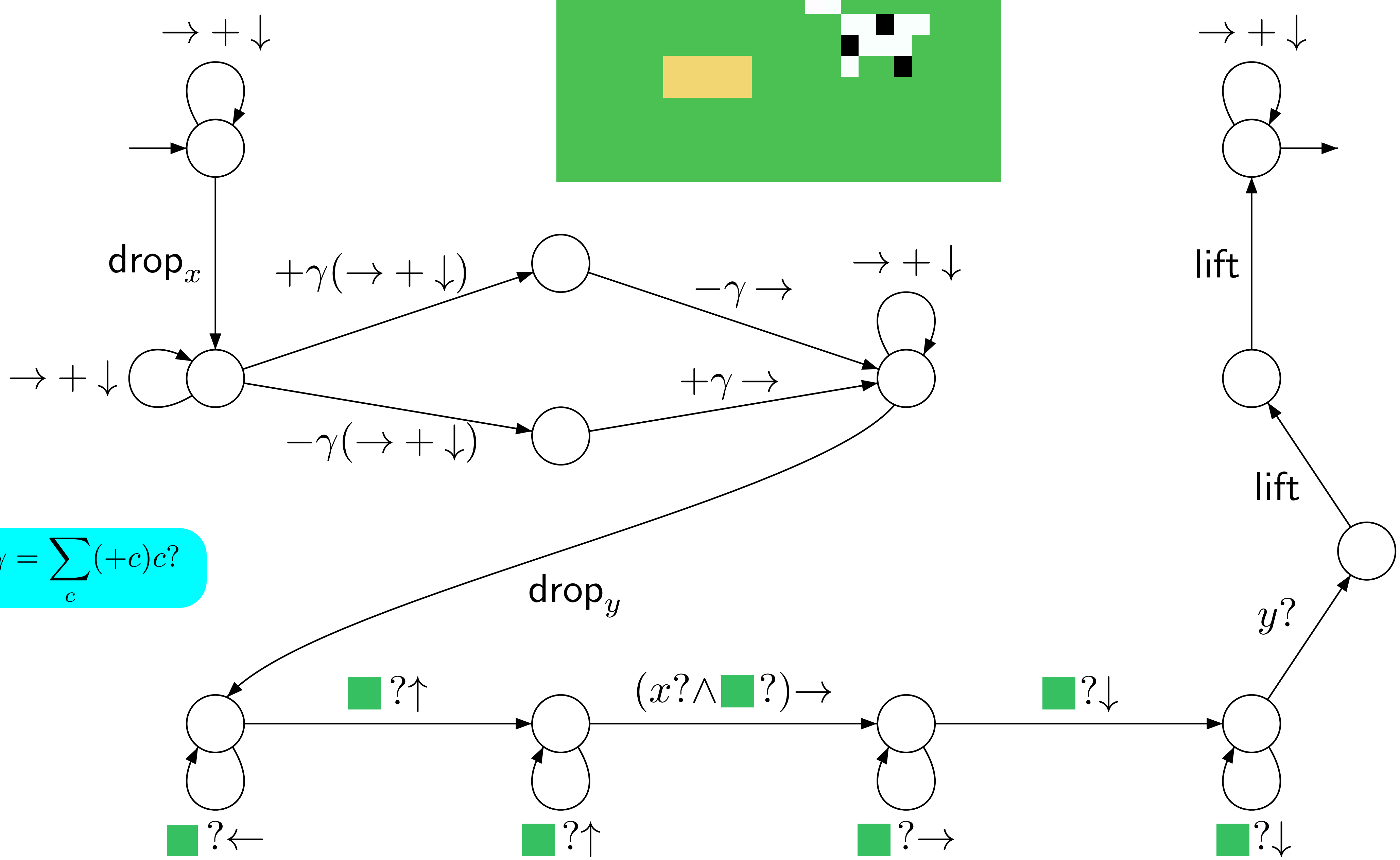
Max, Min, Sum, Average, Expectation, …

Walking Weighted Automata with Pebbles

Pebble Walking Weighted Automata



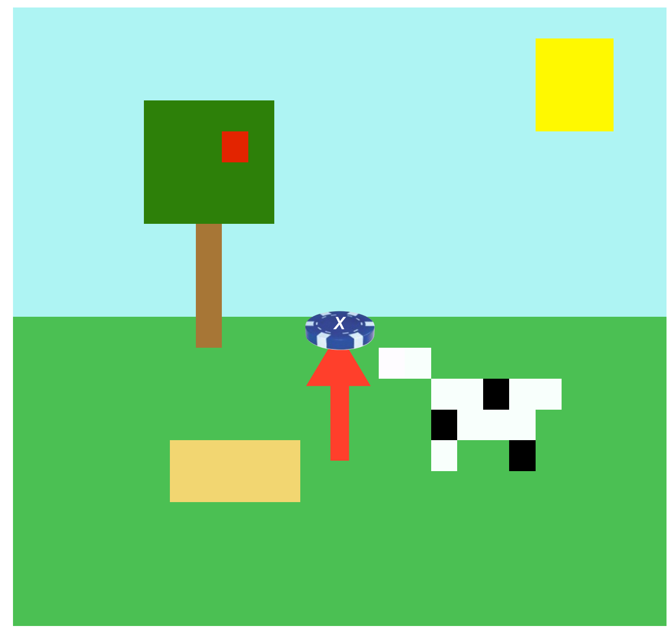
$(\mathbb{Z} \cup \{-\infty\}, \max, +, -\infty, 0)$



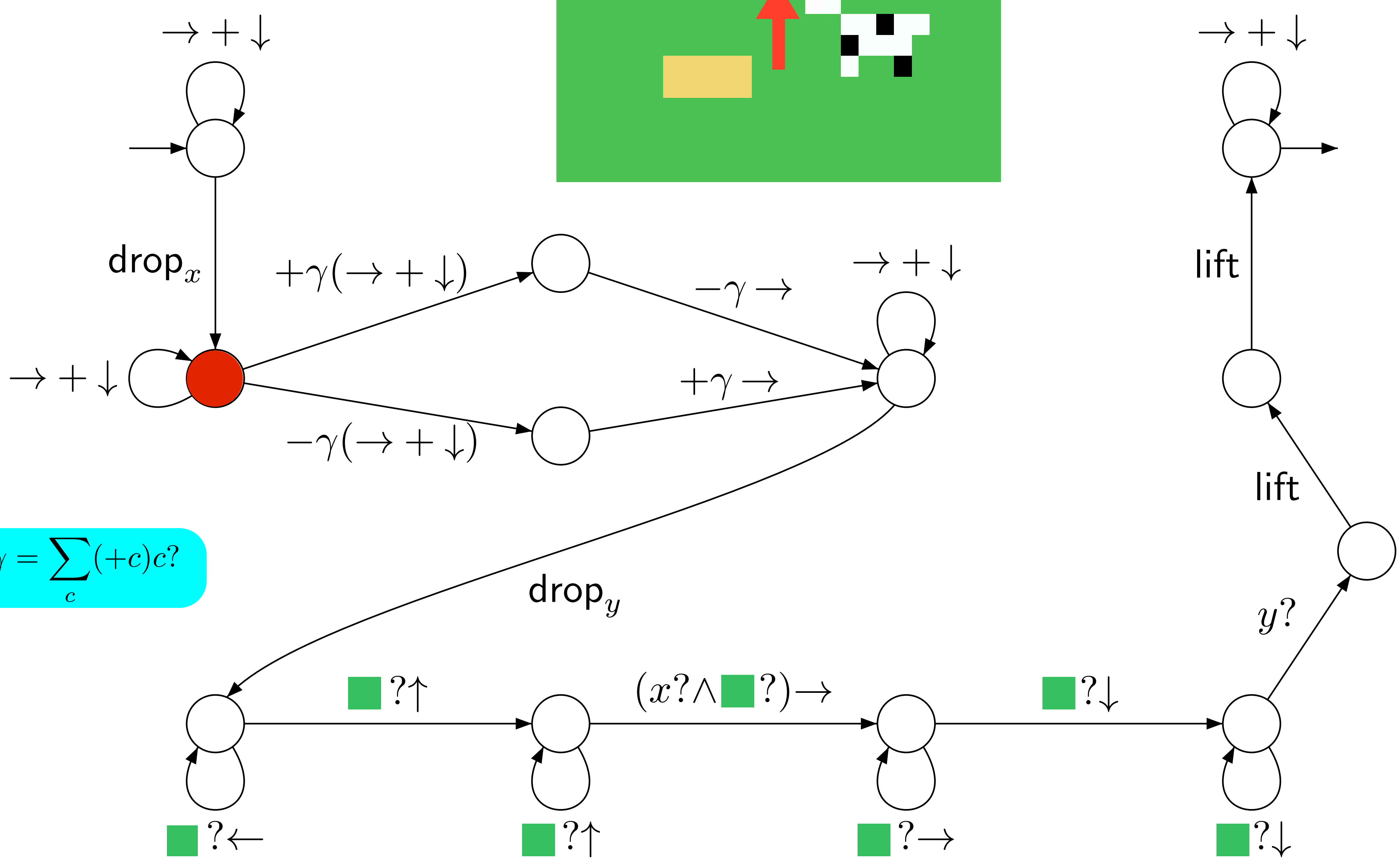
$\gamma = \sum_c (+c)c?$

Pebble automata over words and trees: [Globerman&Harel 96], [Engelfriet&Hoogeboom 99]

Pebble Walking Weighted Automata



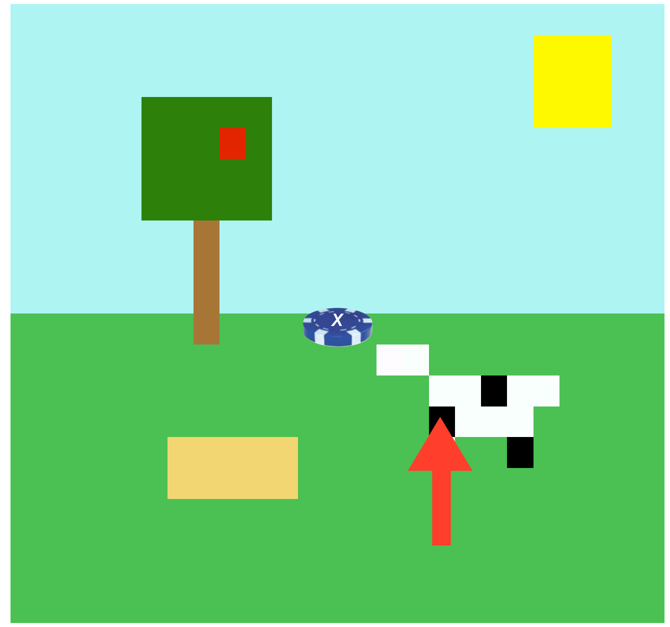
$(\mathbb{Z} \cup \{-\infty\}, \max, +, -\infty, 0)$



$\gamma = \sum_c (+c)c?$

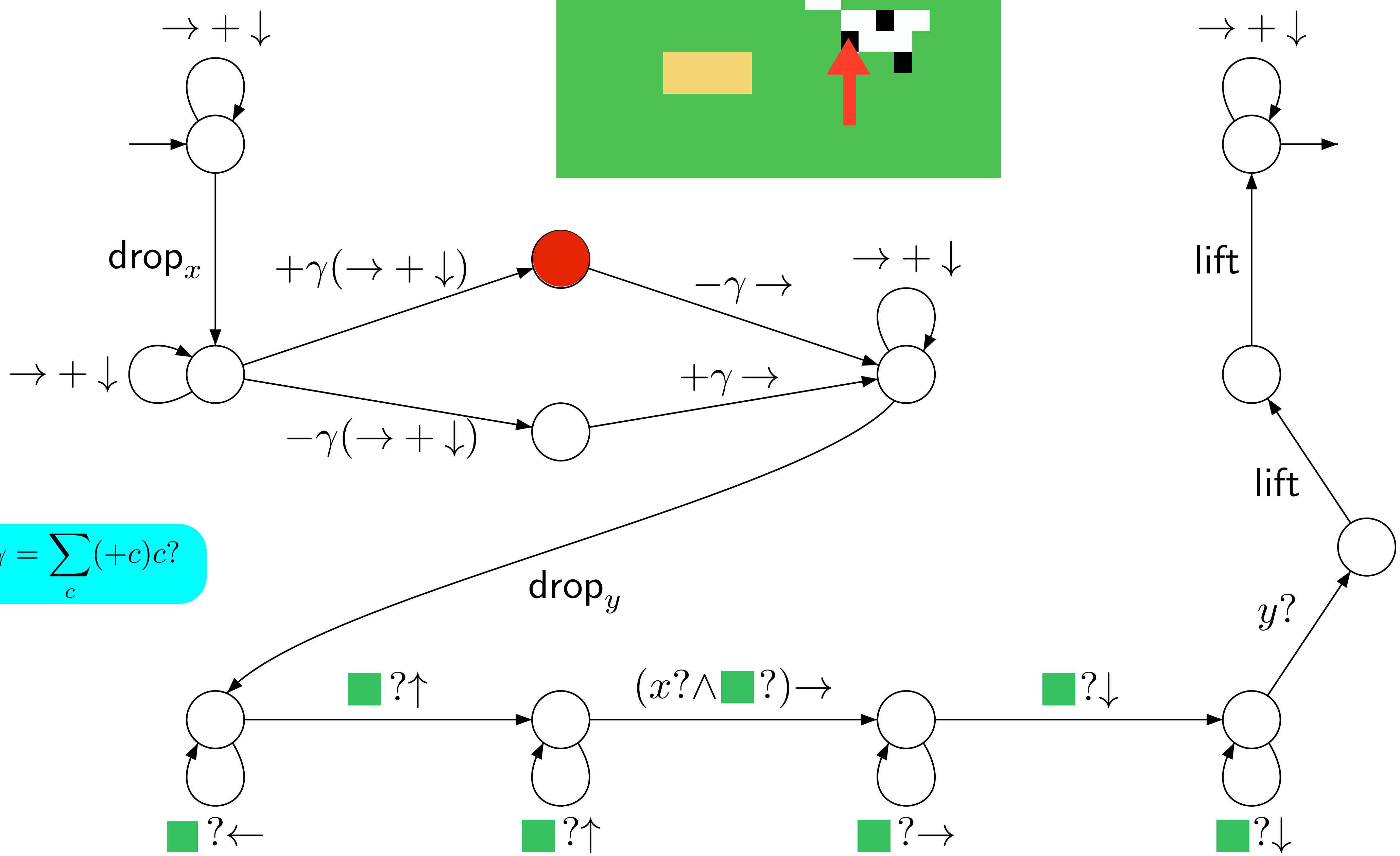
Pebble automata over words and trees: [Globerman&Harel 96], [Engelfriet&Hoogeboom 99]

Pebble Walking Weighted Automata



$(\mathbb{Z} \cup \{-\infty\}, \max, +, -\infty, 0)$

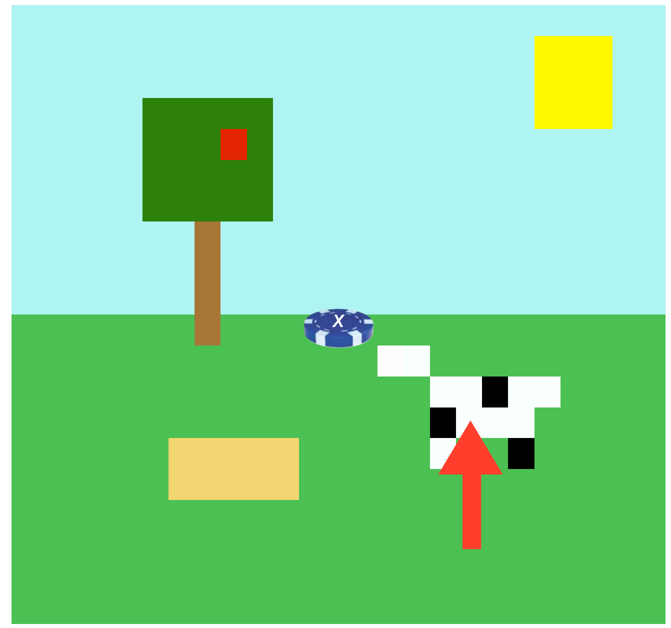
+ 255



$\gamma = \sum_c (+c)c?$

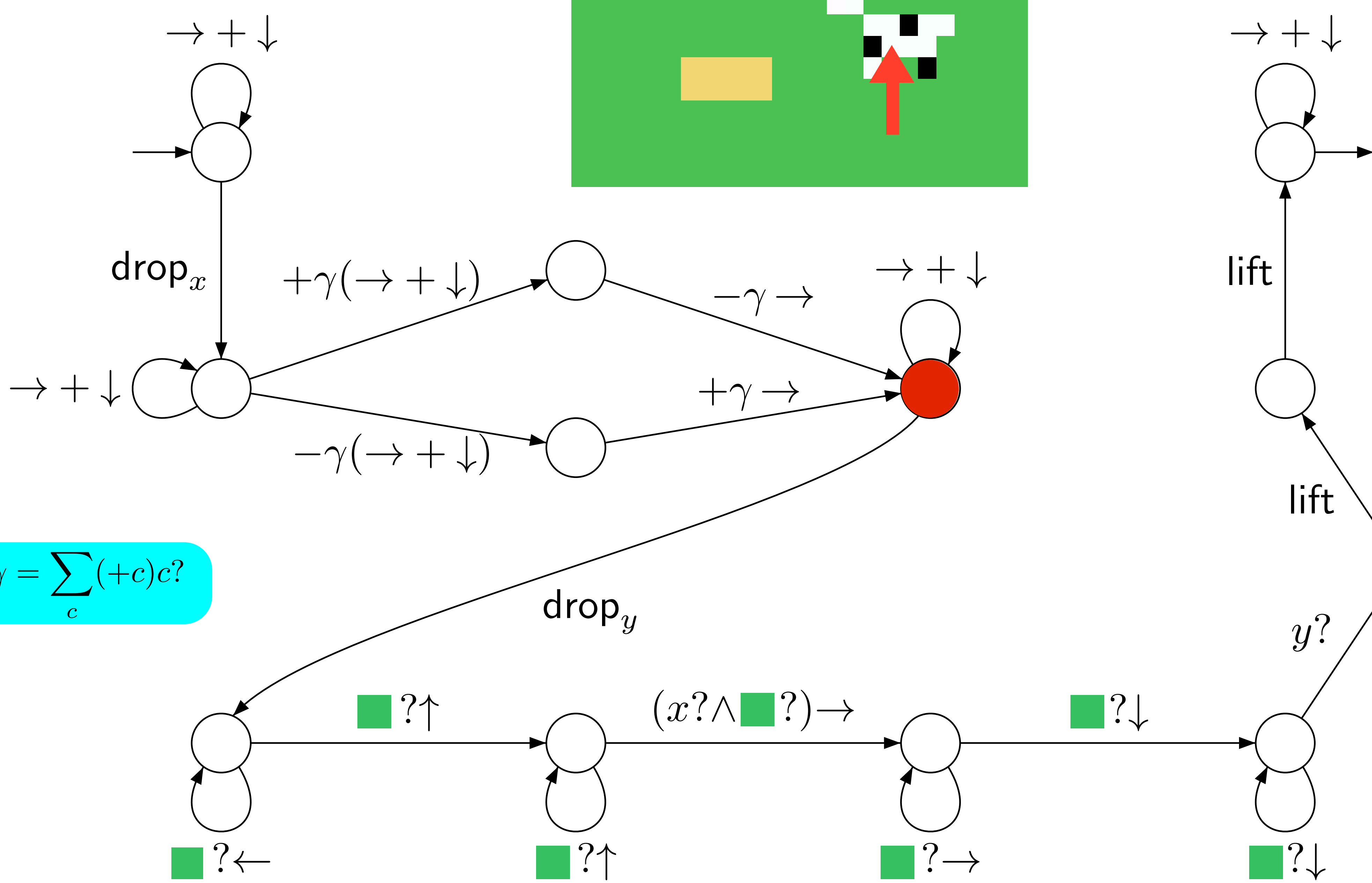
Pebble automata over words and trees: [Globerman&Harel 96], [Engelfriet&Hoogeboom 99]

Pebble Walking Weighted Automata



$(\mathbb{Z} \cup \{-\infty\}, \max, +, -\infty, 0)$

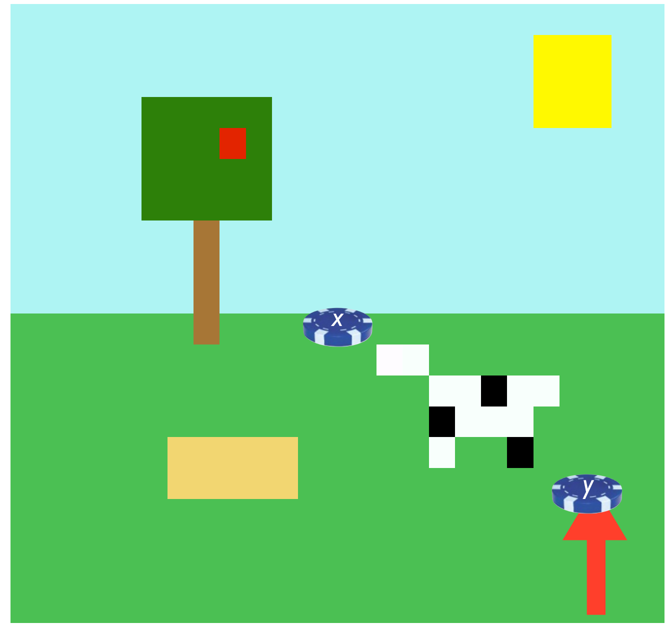
+ 255 - 0



$\gamma = \sum_c (+c)c?$

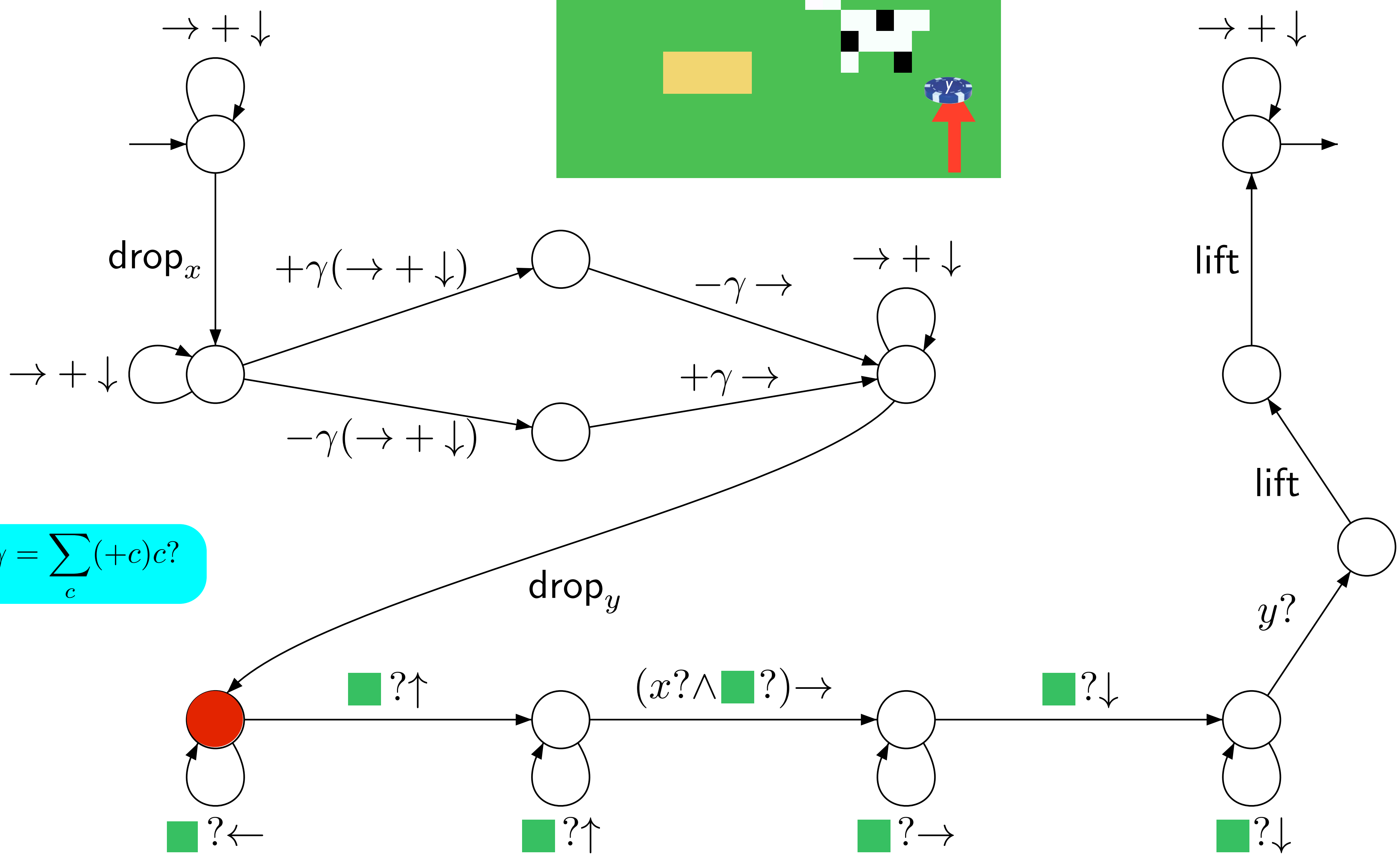
Pebble automata over words and trees: [Globerman&Harel 96], [Engelfriet&Hoogeboom 99]

Pebble Walking Weighted Automata



$(\mathbb{Z} \cup \{-\infty\}, \max, +, -\infty, 0)$

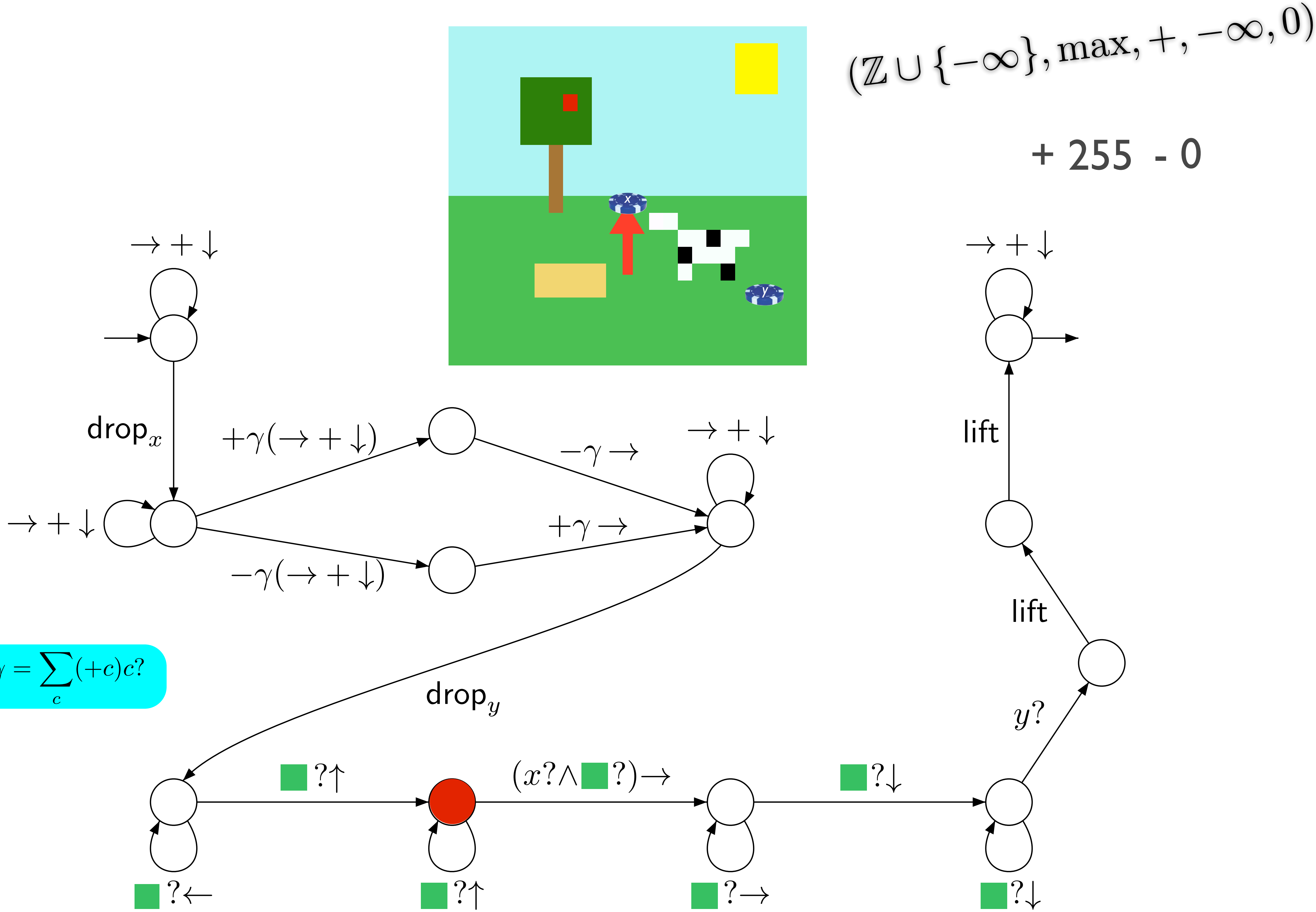
+ 255 - 0



$\gamma = \sum_c (+c)c?$

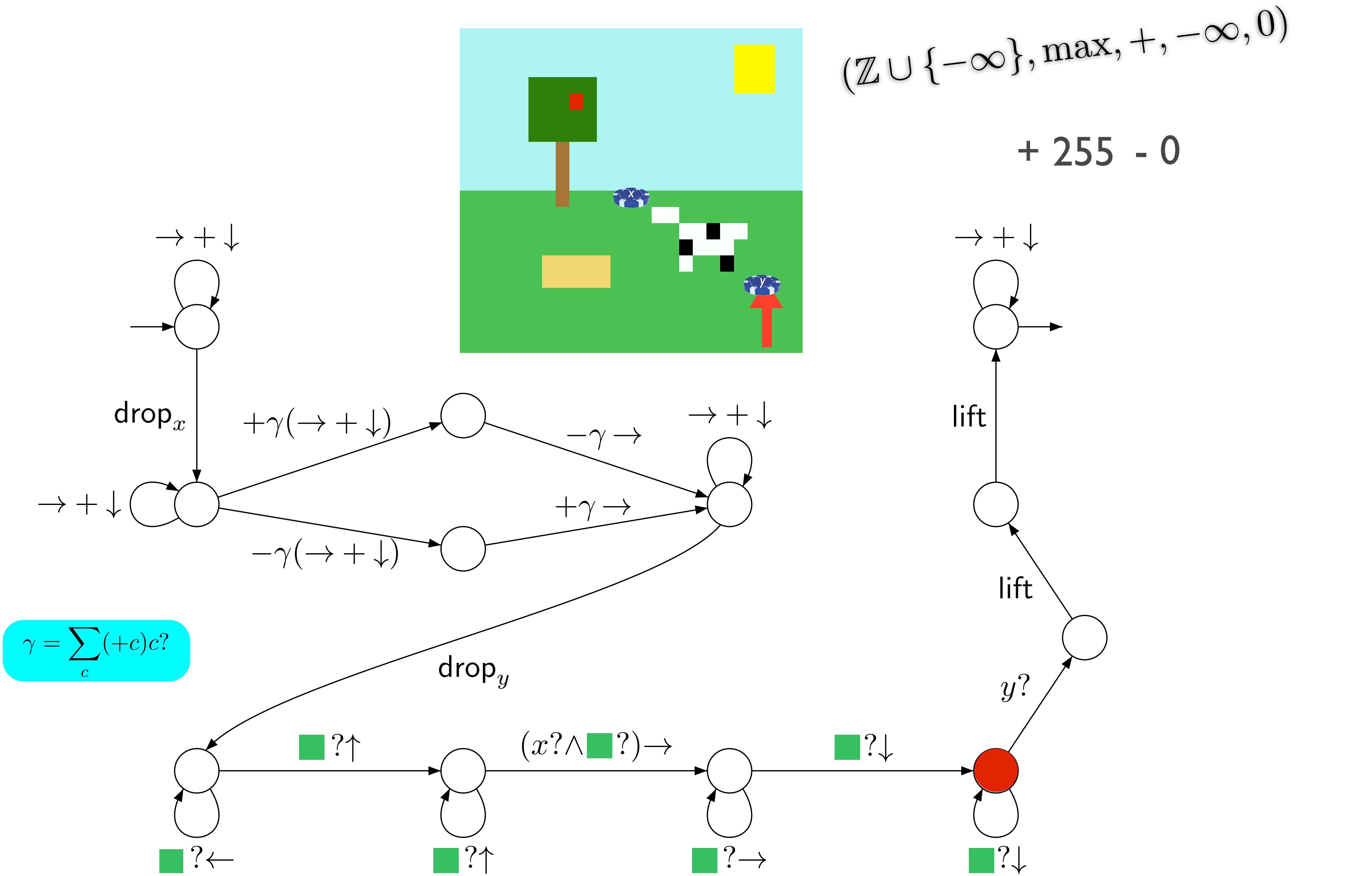
Pebble automata over words and trees: [Globerman&Harel 96], [Engelfriet&Hoogeboom 99]

Pebble Walking Weighted Automata



Pebble automata over words and trees: [Globerman&Harel 96], [Engelfriet&Hoogeboom 99]

Pebble Walking Weighted Automata

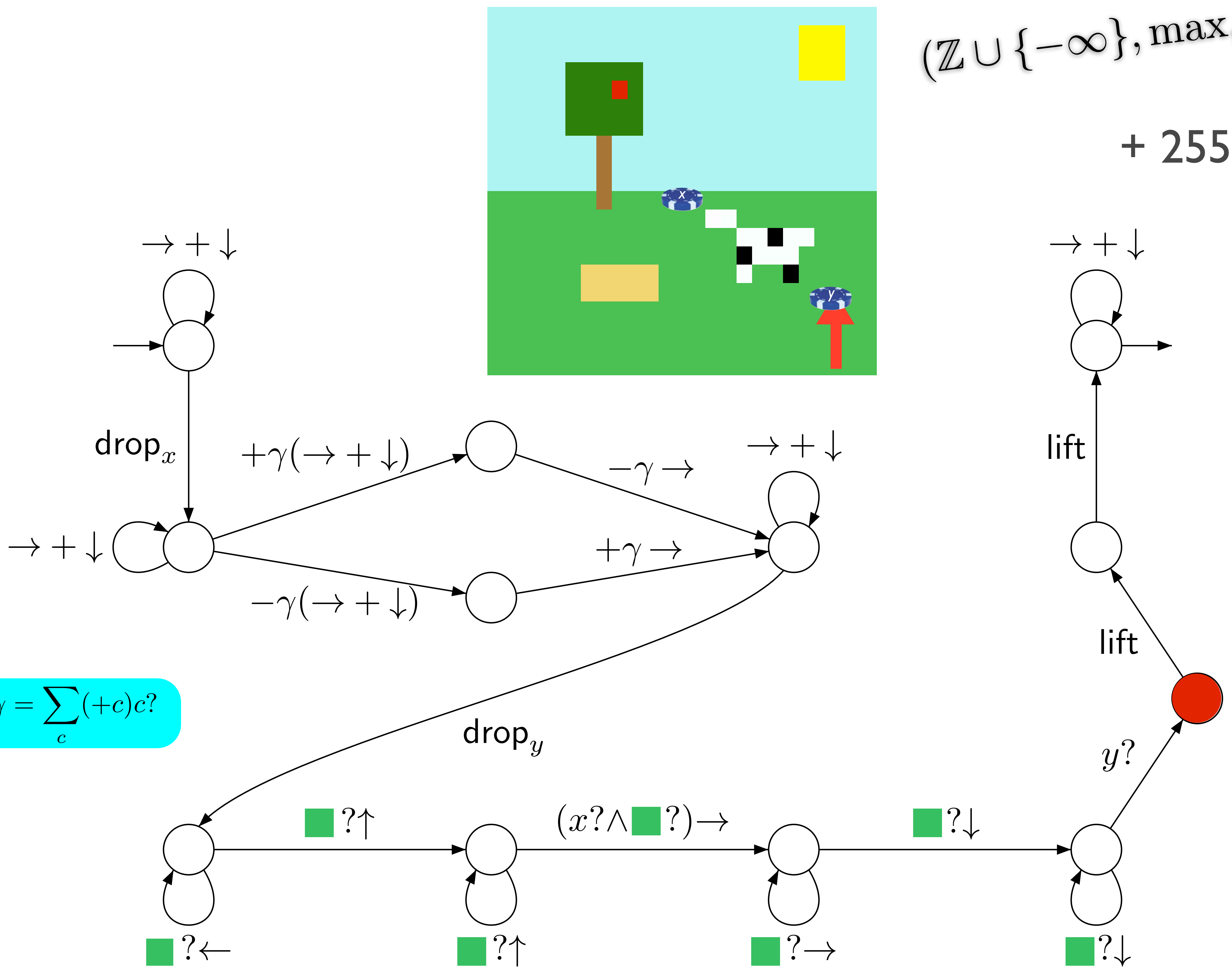


Pebble automata over words and trees: [Globerman&Harel 96], [Engelfriet&Hoogeboom 99]

Pebble Walking Weighted Automata

$(\mathbb{Z} \cup \{-\infty\}, \max, +, -\infty, 0)$

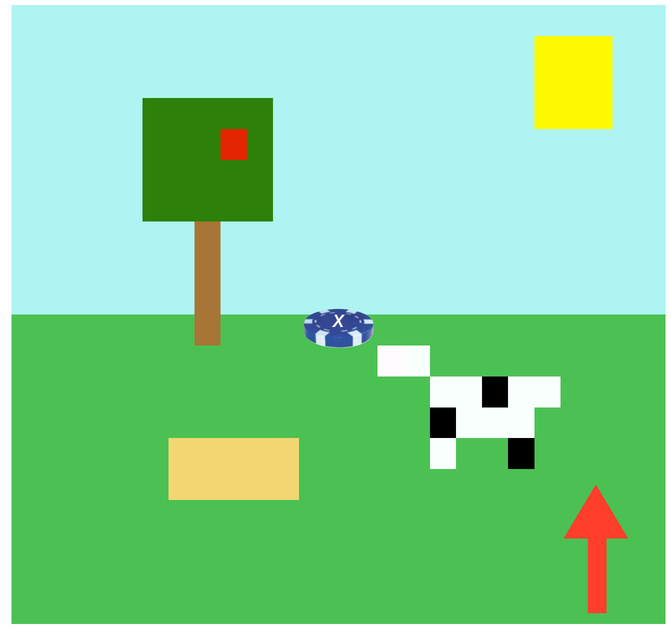
+ 255 - 0



$\gamma = \sum_c (+c)c?$

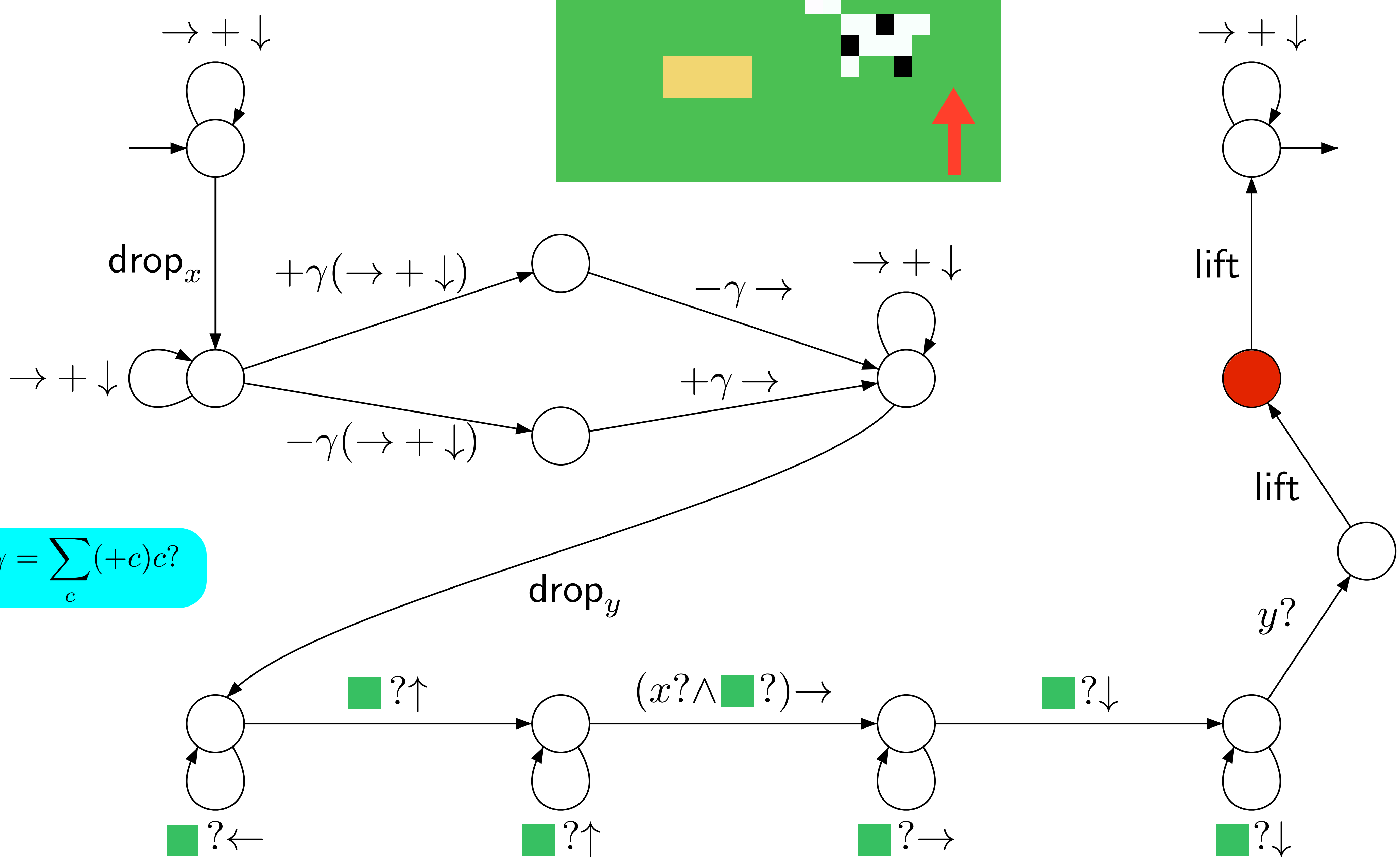
Pebble automata over words and trees: [Globerman&Harel 96], [Engelfriet&Hoogeboom 99]

Pebble Walking Weighted Automata



$(\mathbb{Z} \cup \{-\infty\}, \max, +, -\infty, 0)$

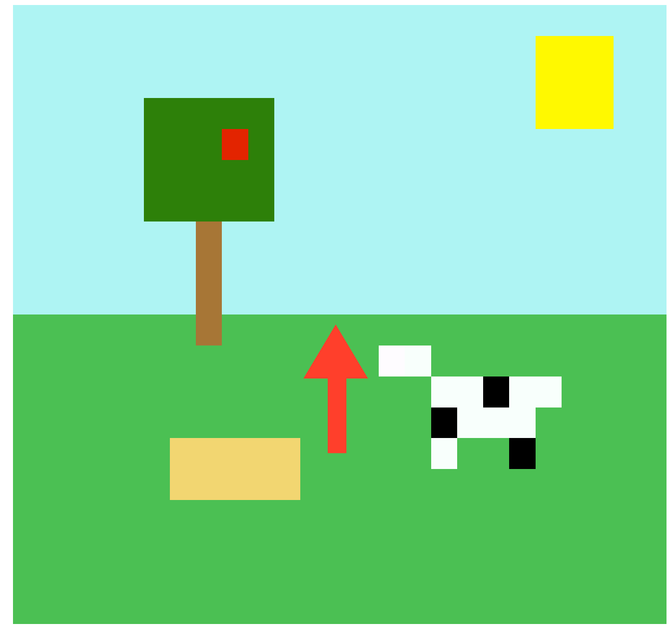
+ 255 - 0



$\gamma = \sum_c (+c)c?$

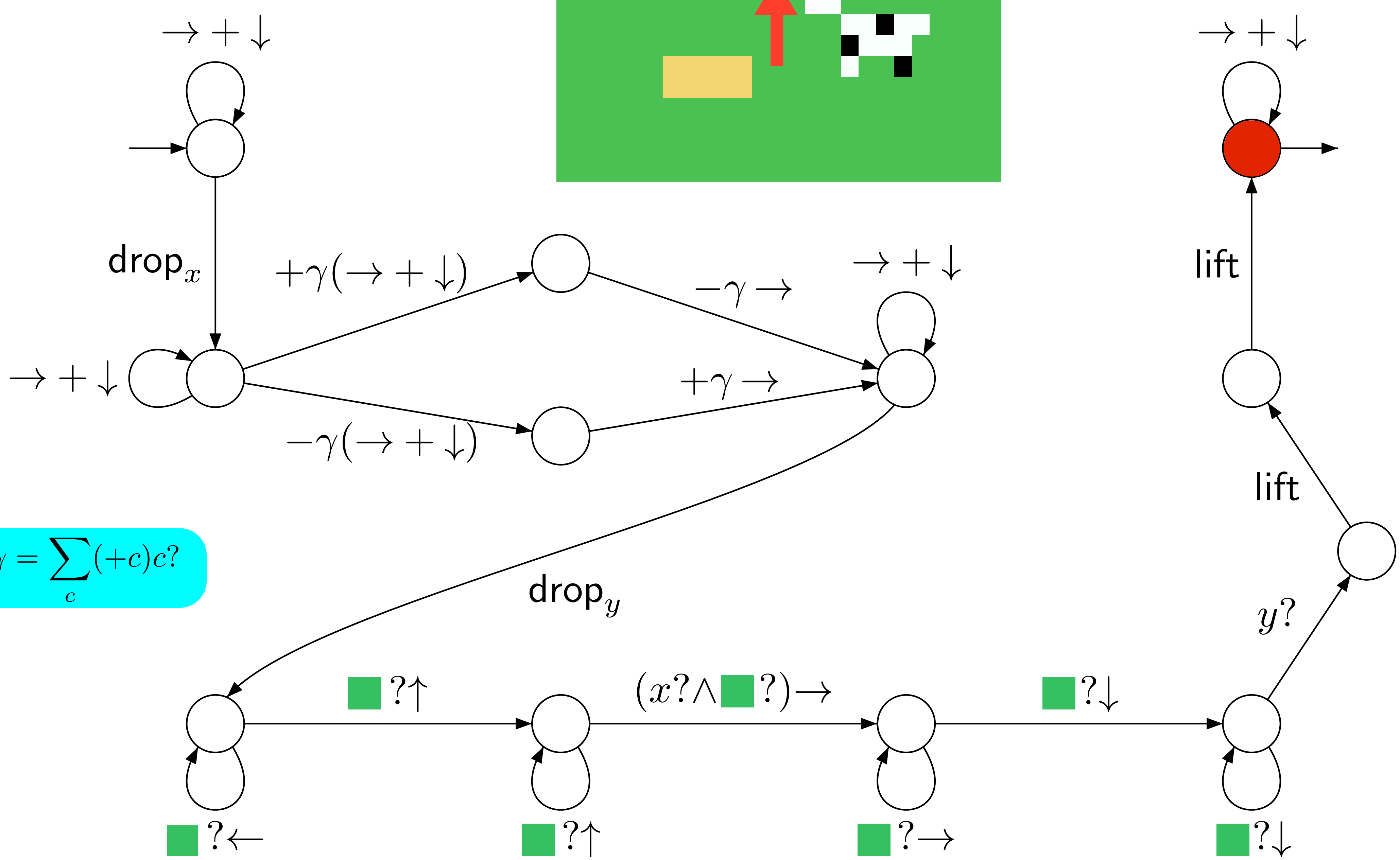
Pebble automata over words and trees: [Globerman&Harel 96], [Engelfriet&Hoogeboom 99]

Pebble Walking Weighted Automata



$$(\mathbb{Z} \cup \{-\infty\}, \max, +, -\infty, 0)$$

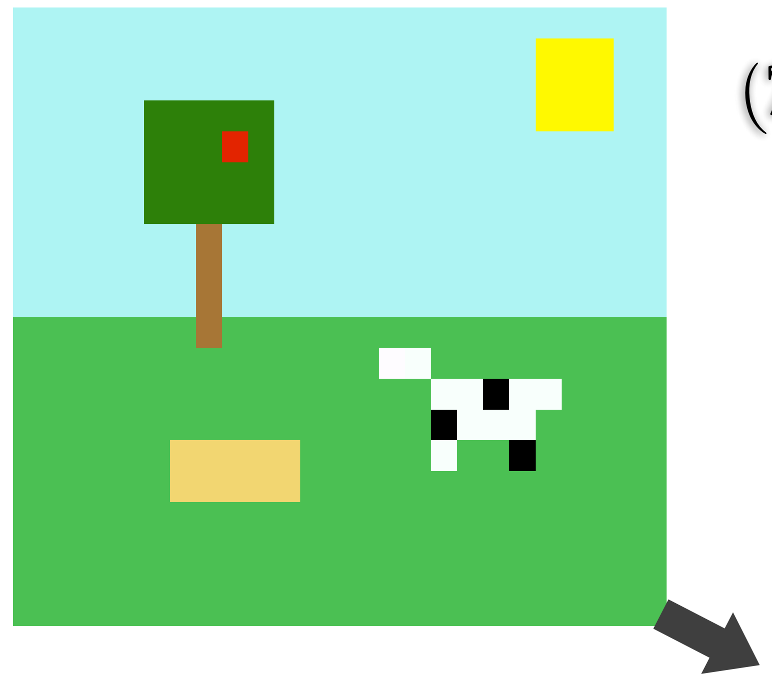
+ 255 - 0



$$\gamma = \sum_c (+c)c?$$

Pebble automata over words and trees: [Globerman&Harel 96], [Engelfriet&Hoogeboom 99]

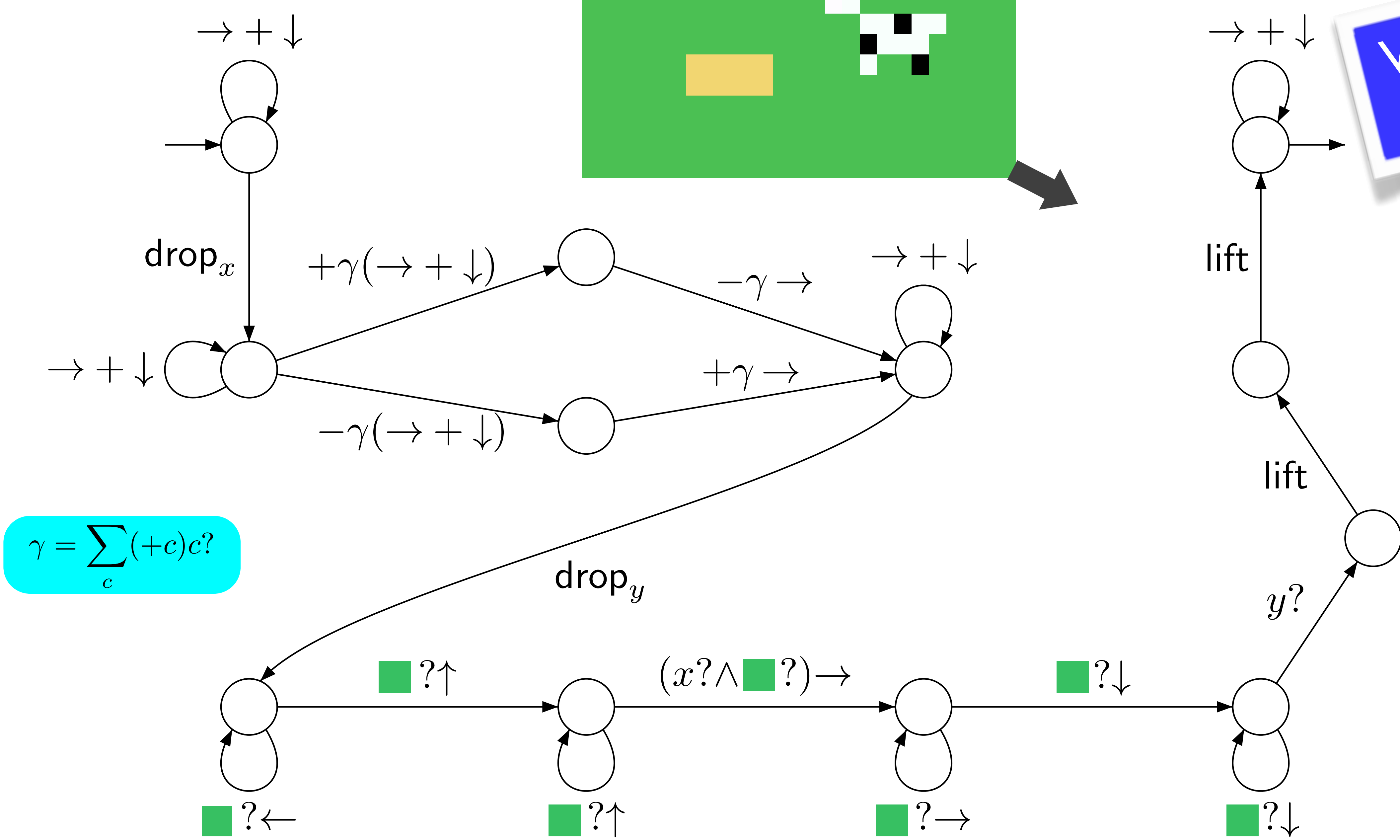
Pebble Walking Weighted Automata



$(\mathbb{Z} \cup \{-\infty\}, \max, +, -\infty, 0)$

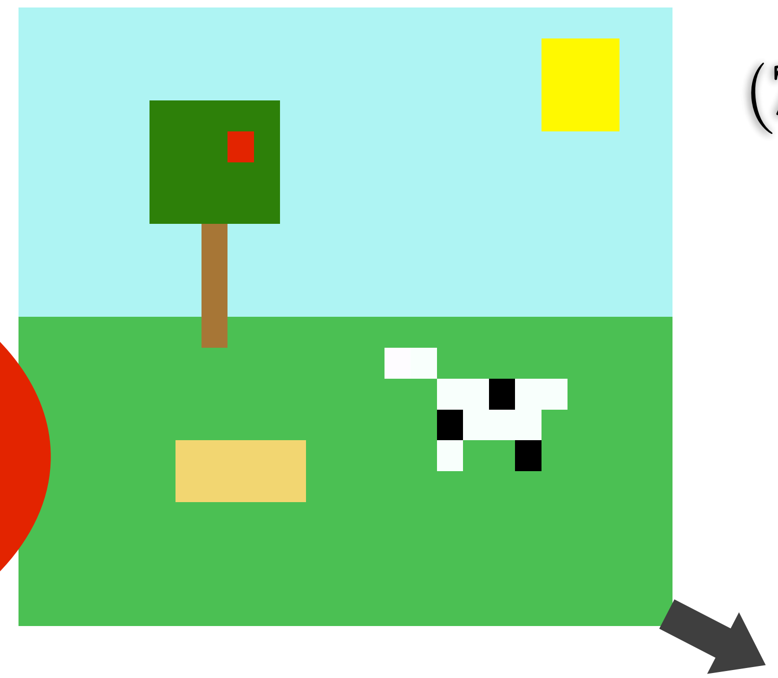
+ 255 - 0

Weight of the run: 255



Pebble automata over words and trees: [Globerman&Harel 96], [Engelfriet&Hoogeboom 99]

Pebble Walking Weighted Automata

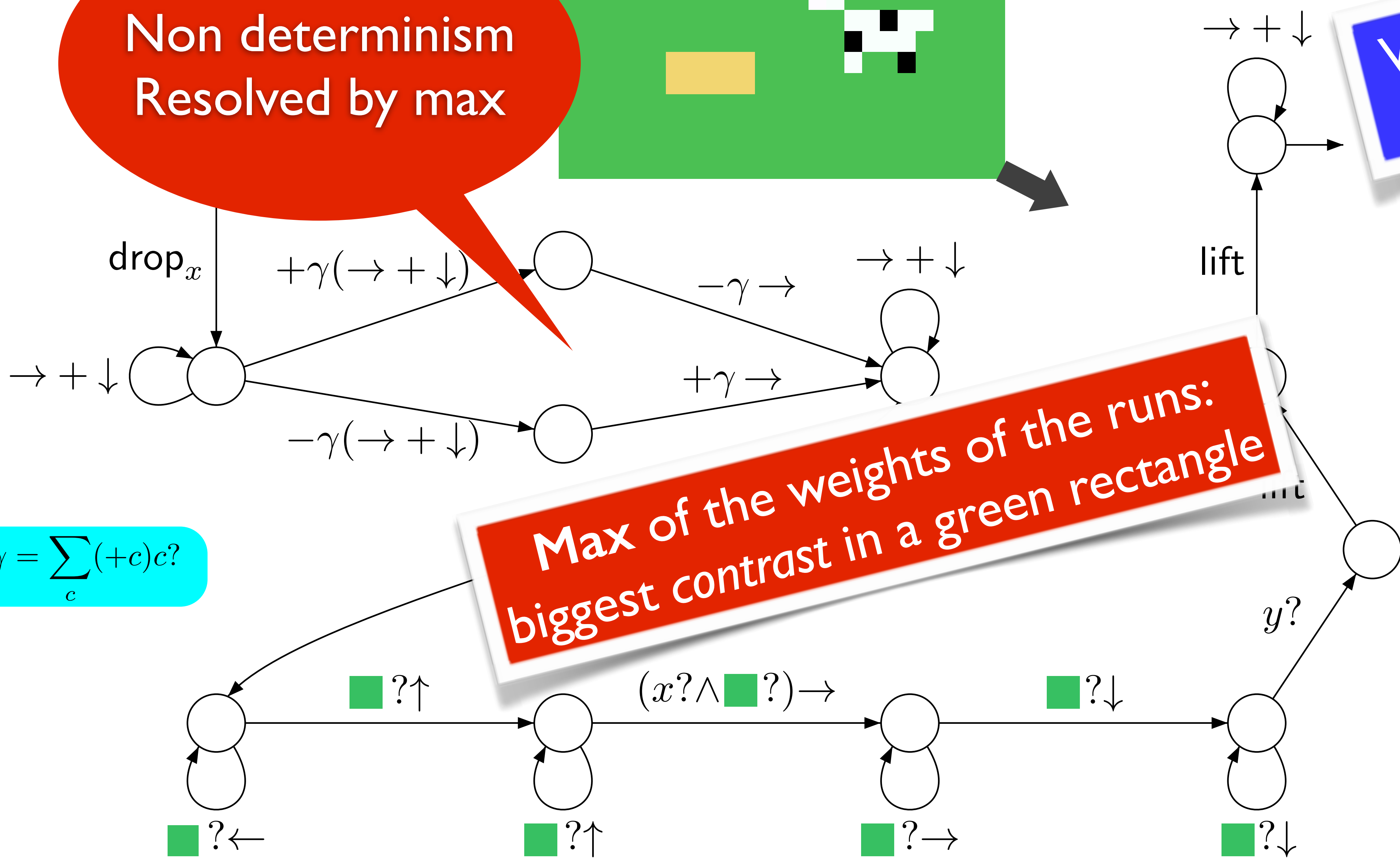


$(\mathbb{Z} \cup \{-\infty\}, \max, +, -\infty, 0)$

+ 255 - 0

Non determinism
Resolved by max

Weight of the run: 255



Max of the weights of the runs: biggest contrast in a green rectangle

$\gamma = \sum_c (+c)c?$

Pebble automata over words and trees: [Globerman&Harel 96], [Engelfriet&Hoogeboom 99]

Pebble Weighted Automata

$$\mathcal{A} = (Q, A, D, I, \Delta, F)$$

finite set of states

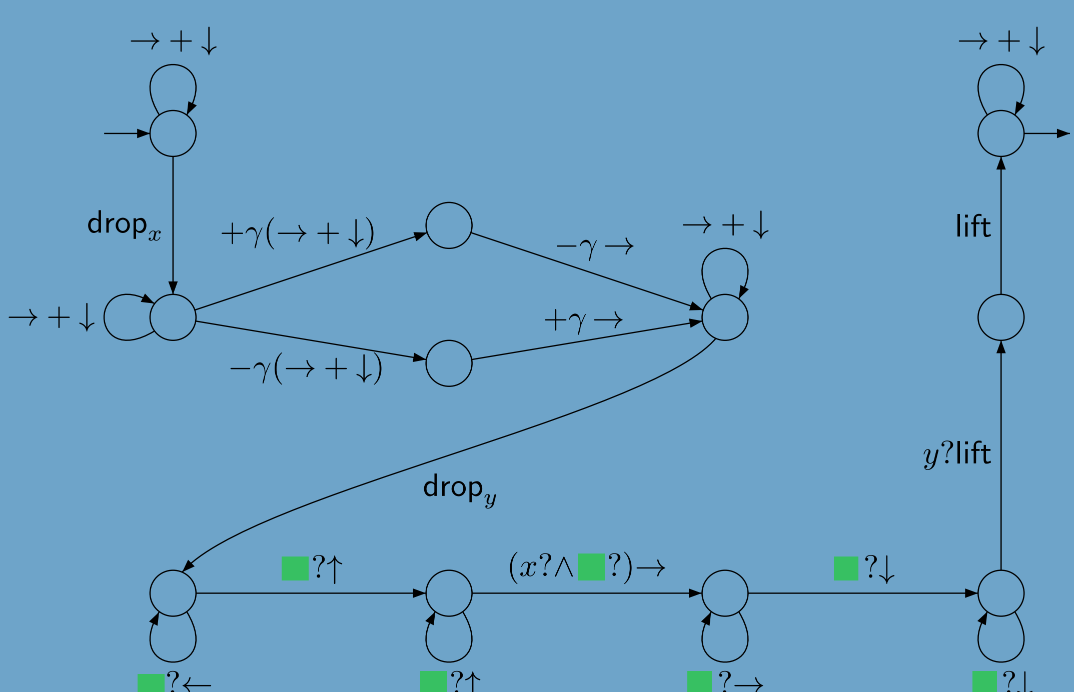
alphabet

set of directions

vector of initial weights

matrix of transition

vector of final weights



transitions: linear combination of (test, action)

boolean combination of

T, *a*?, *x*?, *d*?, init?, final?

move *d*,

drop a pebble,

lift the last dropped pebble

Semantics of Pebble Weighted Automata

Configuration of $\mathcal{A}$: (G, q, π, v)

state

stack of dropped pebbles

current vertex

Run over a graph G : finite sequence of configurations

Weight of a run: multiplication of the weights of the transitions and the initial and final weights

Semantics $[\![\mathcal{A}]\!](G)$: sum of the weights of the runs

Navigation possibly leads to infinitely many runs!

Restrict to *continuous* semirings

Weight Domains: Continuous Semirings

$$(\mathbb{S}, +, \times, 0, 1)$$

every infinite sum exists and is the limit of finite *approximate* sums,
keeping good properties of usual semiring

$$\cancel{(\mathbb{R}, +, \times, 0, 1)}$$
$$\cancel{(\mathbb{Q}, +, \times, 0, 1)}$$
$$\cancel{(\mathbb{Z}, +, \times, 0, 1)}$$
$$\cancel{(\mathbb{N}, +, \times, 0, 1)}$$

$$(\{0,1\}, \vee, \wedge, 0, 1)$$
$$([0,1], \max, \min, 0, 1)$$
$$(\mathbb{R} \cup \{-\infty, +\infty\}, \max, \min, -\infty, +\infty)$$

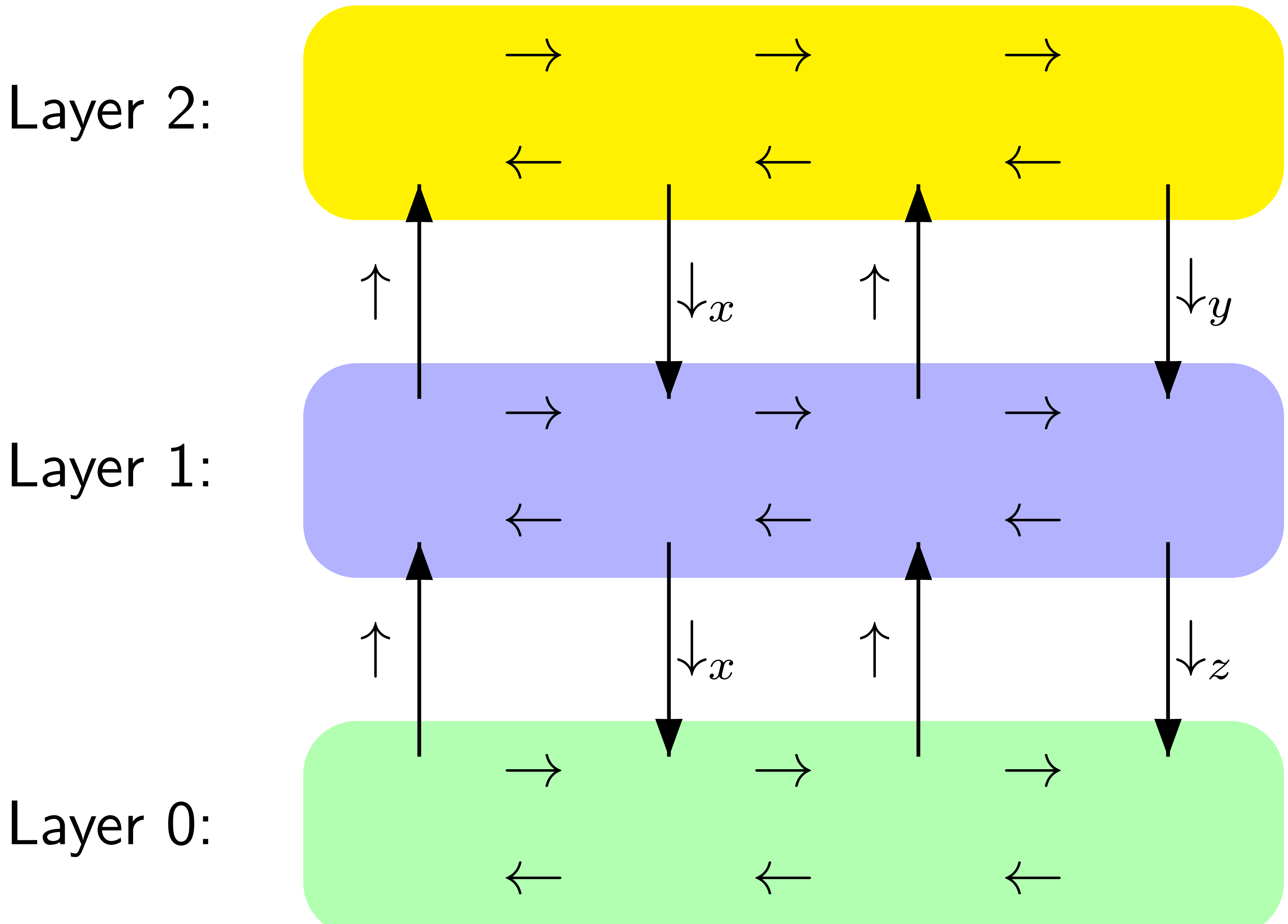
$$(\mathbb{R}^+ \cup \{+\infty\}, +, \times, 0, 1)$$
$$(\mathbb{N} \cup \{+\infty\}, +, \times, 0, 1)$$

$$(\mathbb{R} \cup \{+\infty\}, \min, +, +\infty, 0)$$
$$(\mathbb{R} \cup \{-\infty\}, \max, +, -\infty, 0)$$

$$\mathbb{R} \cup \{-\infty, +\infty\}$$

$$(\mathfrak{P}(A^\star), \cup, \cdot, \emptyset, \{\varepsilon\})$$

Layered Automata



Weighted FO and pebWA

wFO vs pebWA

Weighted First-Order logic

- Denotational semantics
- Boolean fragment
- Quantitative formulas

Weighted Automata with pebbles

- Operational semantics
- Only quantitative computations

wFO vs pebWA

Searchable Graphs

$$G = (V, (R_d)_{d \in D}, \lambda, \triangleright)$$

$\triangleright$ initial vertex

$\leq$ total order over vertices, computable with navigating automata

Weighted First-Orde

- Denotational sema
- Boolean fragment
- Quantitative formulas

We

- C
- C

Thm: Consider a searchable class of graphs.

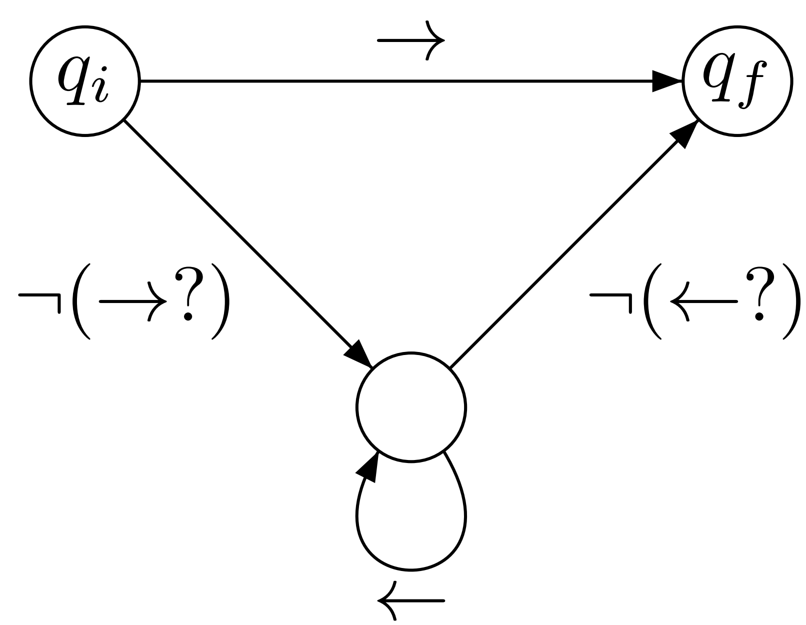
For each wFO formula, we can construct an equivalent layered pebWA,
states linear in the size of the formula
layers = quantifier depth of the formula
pebble names = # variable names in the formula

Searchable Graphs

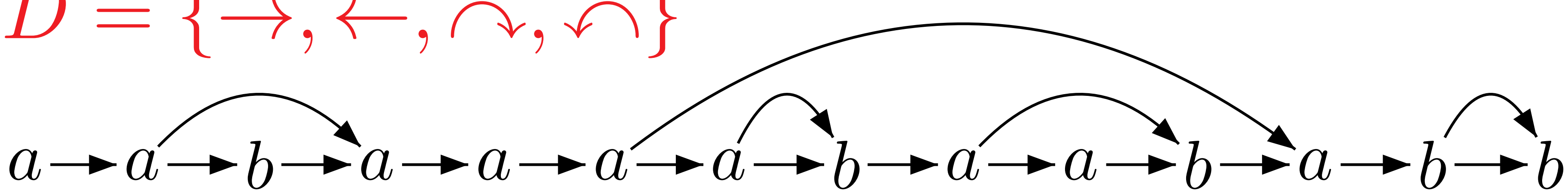
A class $\mathfrak{G}$ of graphs is searchable if there exists a walking automaton $\mathcal{A}_{\mathfrak{G}}$ with two states q_i and q_f such that for each graph $G = (V, (R_d)_{d \in D}, \lambda, \triangleright)$ in $\mathfrak{G}$

- There exists a total order $\leq$ on E with minimal element $\triangleright$,
- for every $v \in V$, there is a unique run of $\mathcal{A}_{\mathfrak{G}}$ starting from (G, q_i, v) and ending in state q_f,
- this run ends in configuration (G, q_0, v') with $v \lessdot v'$ if v is not maximal in $(V, \leq)$ and $v' = \triangleright$ otherwise.

The walking automaton $\mathcal{A}_{\mathfrak{G}}$ is the guide for the class $\mathfrak{G}$.



- **Words**
  Computations of sequential programs
  $D = \{\rightarrow, \leftarrow\}$
- **Nested Words**
  Computations of recursive programs
  XML documents
  $D = \{\rightarrow, \leftarrow, \curvearrowright, \curvearrowleft\}$



Searchable Graphs

A class $\mathfrak{G}$ of graphs is searchable if there exists a walking automaton $\mathcal{A}_{\mathfrak{G}}$ with two states q_i and q_f such that for each graph $G = (V, (R_d)_{d \in D}, \lambda, \triangleright)$ in $\mathfrak{G}$

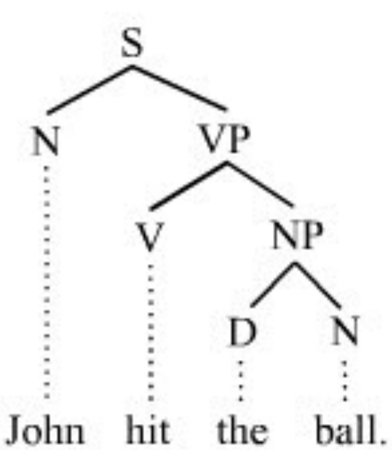
- There exists a total order $\leq$ on E with minimal element $\triangleright$,
- for every $v \in V$, there is a unique run of $\mathcal{A}_{\mathfrak{G}}$ starting from (G, q_i, v) and ending in state q_f,
- this run ends in configuration (G, q_0, v') with $v \lessdot v'$ if v is not maximal in $(V, \leq)$ and $v' = \triangleright$ otherwise.

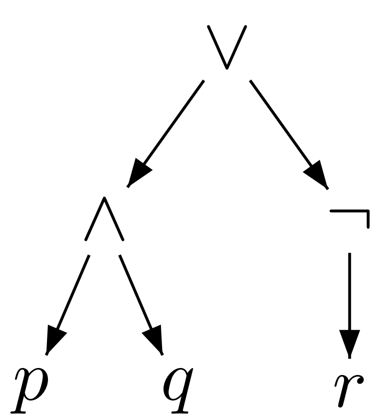
The walking automaton $\mathcal{A}_{\mathfrak{G}}$ is the guide for the class $\mathfrak{G}$.

- **Ranked Trees**

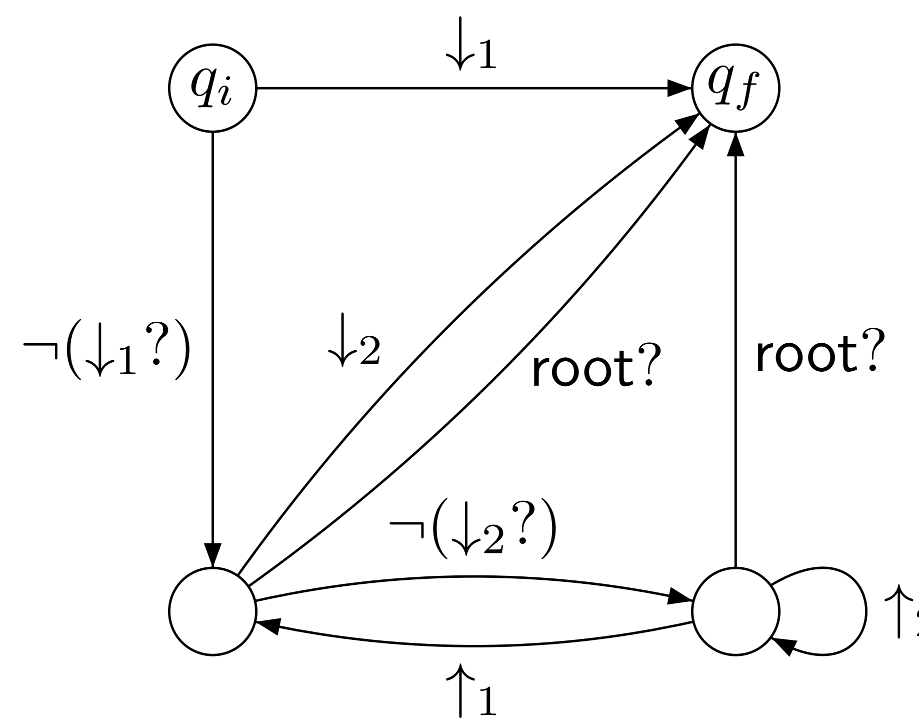
Expressions, Formulas, Parse trees, ...

$D = \{\downarrow_1, \uparrow_1, \downarrow_2, \uparrow_2\}$





DFS



wFO to pebWA

$$\varphi ::= \top \mid P_a(x) \mid x = y \mid R_d(x,y) \mid R_d^+(x,y) \mid \neg\varphi \mid \varphi \vee \varphi \mid \exists x\, \varphi$$

$$\Phi ::= s \mid \varphi ?\, \Phi : \Phi \mid \Phi \oplus \Phi \mid \Phi \otimes \Phi \mid \bigoplus \ \Phi \mid \ldots$$

with $a \in A \cup \{\triangleright\}$, $d \in D$, $s \in \mathbb{S}$.

Runs end on the marker

Searchable graphs

$\Phi \bigoplus \Psi$ disjoint union of automata

$$\Phi \bigotimes \Psi \qquad \longrightarrow \boxed{\mathcal{A}^{\Phi}} \xrightarrow{\triangleright ?} \boxed{\mathcal{A}^{\Psi}} \xrightarrow{\triangleright ?}$$

wFO to pebWA

$$\varphi ::= \top \mid P_a(x) \mid x = y \mid R_d(x,y) \mid R_d^+(x,y) \mid \neg\varphi \mid \varphi \vee \varphi \mid \exists x\, \varphi$$

$$\Phi ::= s \mid \varphi ? \Phi : \Phi \mid \Phi \oplus \Phi \mid \Phi \otimes \Phi \mid \bigoplus \Phi \mid \ldots$$

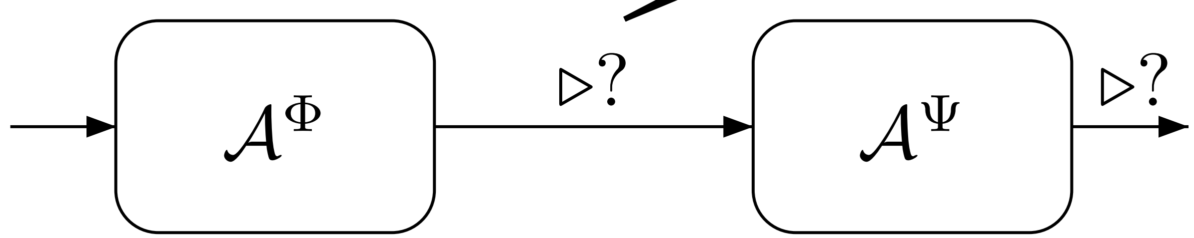
with $a \in A \cup \{\triangleright\}$, $d \in D$, $s \in \mathbb{S}$.

Runs end on the marker

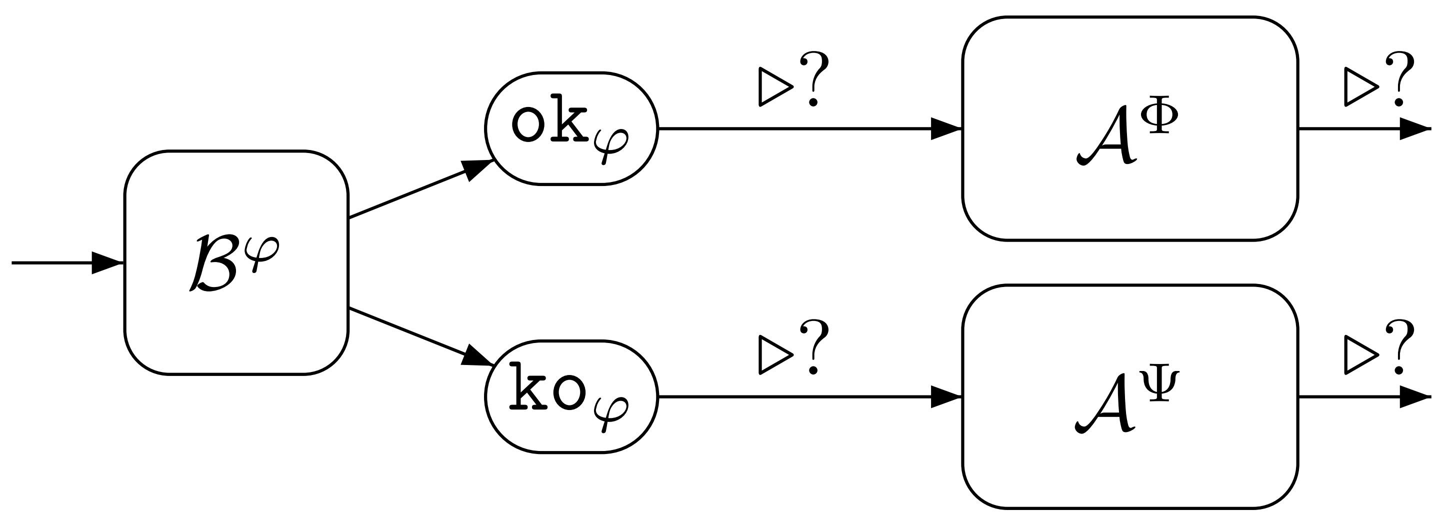
Searchable graphs

$\Phi \bigoplus \Psi$ disjoint union of automata

$\Phi \bigotimes \Psi$



$\varphi ? \Phi : \Psi$



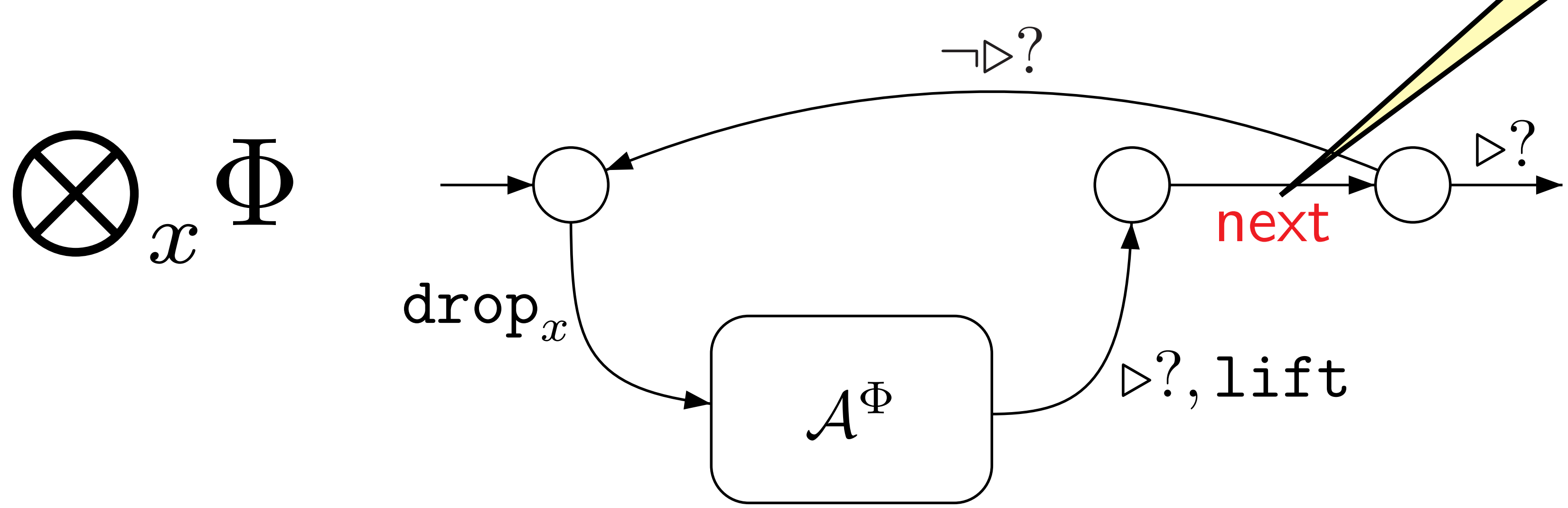
wFO to pebWA

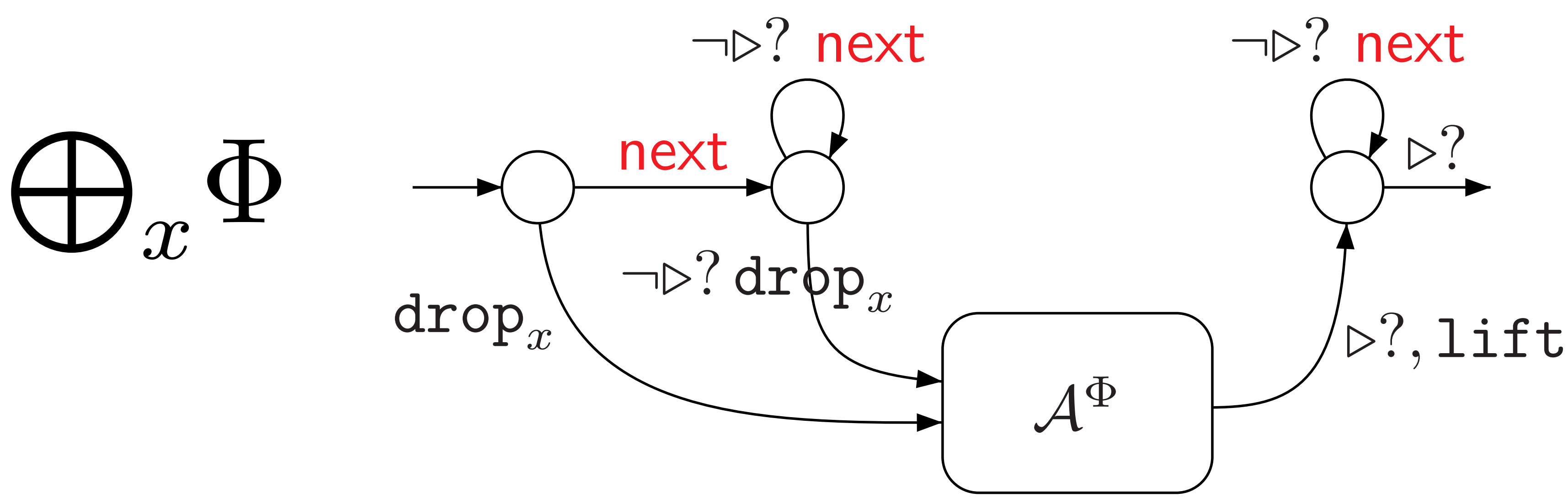
$$\varphi ::= \top \mid P_a(x) \mid x = y \mid x \xrightarrow{d} y \mid x \xrightarrow{d^+} y \mid \neg\varphi \mid \varphi \vee \varphi \mid \varphi \wedge \varphi \mid \exists x\, \varphi \mid \forall x\, \varphi$$

$$\Phi ::= s \mid \varphi ?\, \Phi : \Phi \mid \Phi \oplus \Phi \mid \Phi \otimes \Phi \mid \bigoplus_x \Phi \mid \bigotimes_x \Phi$$

with $a \in A \cup \{\triangleright\}$, $d \in D$, $s \in \mathbb{S}$.

Searchable





wFO to pebWA

Challenging for the *Boolean part*:
we need unambiguous automata

Use deterministic automata of size non-elementary...

Take advantage of the pebbles to build linear sized automata

$$\varphi ::= \top \mid P_a(x) \mid x = y \mid R_d(x, y) \mid R_d^+(x, y) \mid \neg\varphi \mid \varphi \vee \varphi \mid \exists x\, \varphi$$

$$\Phi ::= s \mid \varphi ?\, \Phi : \Phi \mid \Phi \oplus \Phi \mid \Phi \otimes \Phi \mid \bigoplus\nolimits_x \Phi \mid \bigotimes\nolimits_x \Phi$$

with $a \in A \cup \{\triangleright\}$, $d \in D$, $s \in \mathbb{S}$.

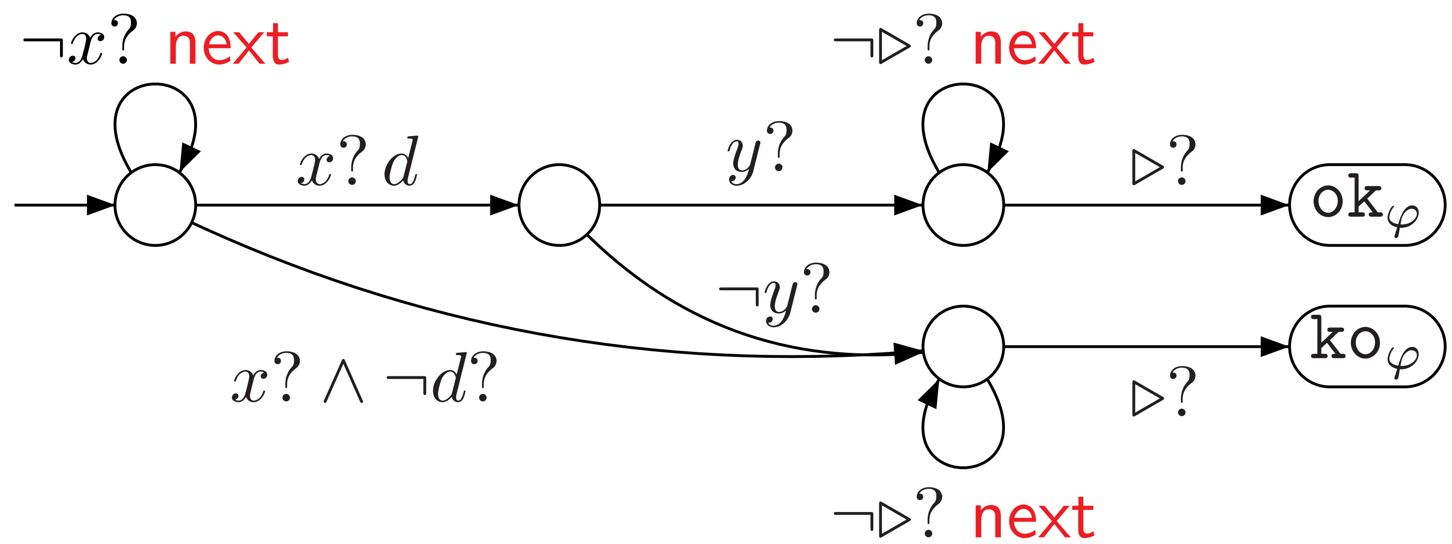
wFO to pebWA

$$\varphi ::= \top \mid P_a(x) \mid x = y \mid R_d(x,y) \mid R_d^+(x,y) \mid \neg\varphi \mid \varphi \vee \varphi \mid \exists x\, \varphi$$

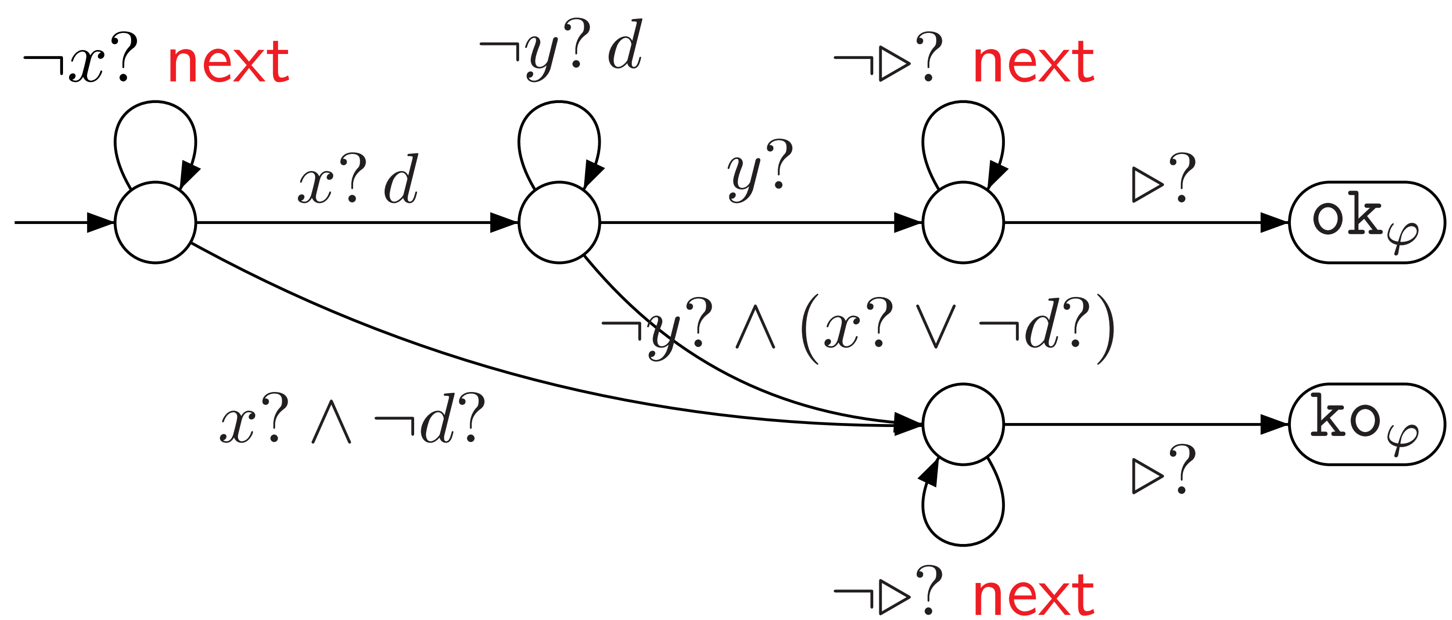
$$\Phi ::= s \mid \varphi ?\, \Phi : \Phi \mid \Phi \oplus \Phi \mid \Phi \otimes \Phi \mid \bigoplus_x \Phi \mid \bigotimes_x \Phi$$

with $a \in A \cup \{\triangleright\}$, $d \in D$, $s \in \mathbb{S}$.

$$\varphi = R_d(x,y)$$



$$\varphi = R_d^+(x,y)$$



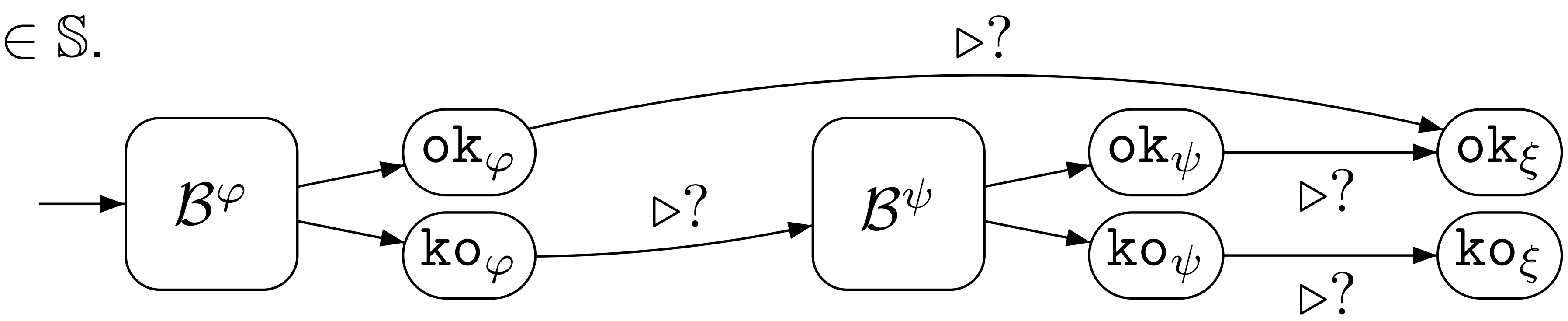
wFO to pebWA

$$\varphi ::= \top \mid P_a(x) \mid x = y \mid R_d(x,y) \mid R_d^+(x,y) \mid \neg\varphi \mid \varphi \vee \varphi \mid \exists x\, \varphi$$

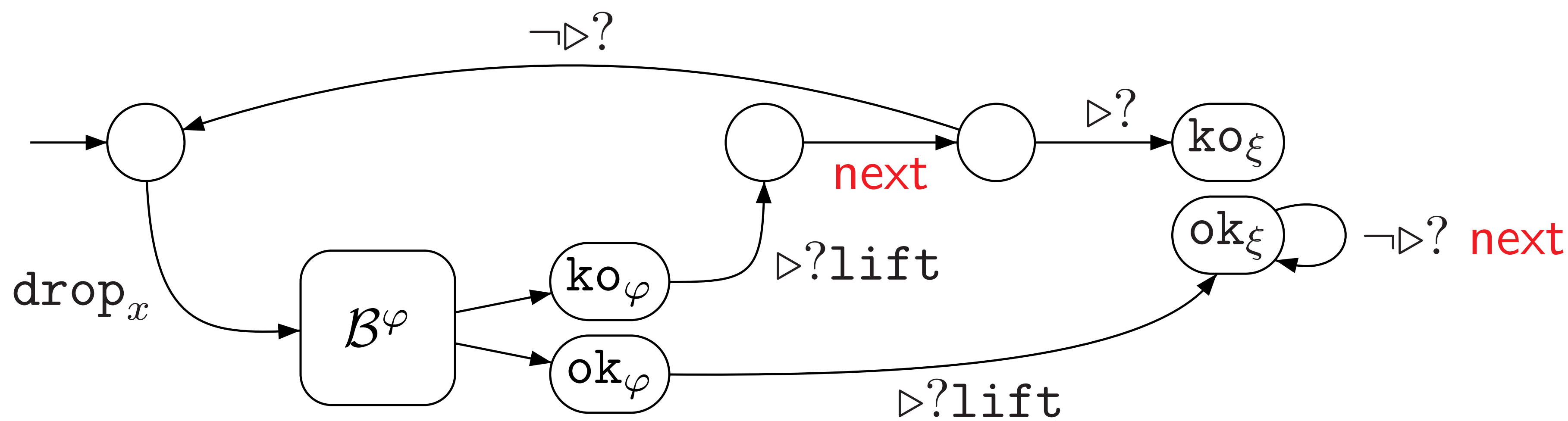
$$\Phi ::= s \mid \varphi ?\, \Phi : \Phi \mid \Phi \oplus \Phi \mid \Phi \otimes \Phi \mid \bigoplus_x \Phi \mid \bigotimes_x \Phi$$

with $a \in A \cup \{\triangleright\}$, $d \in D$, $s \in \mathbb{S}$.

Disjunction/conjunction
$\xi = \varphi \vee \psi$



Existential/Universal quantifications
$\xi = \exists x\, \varphi$



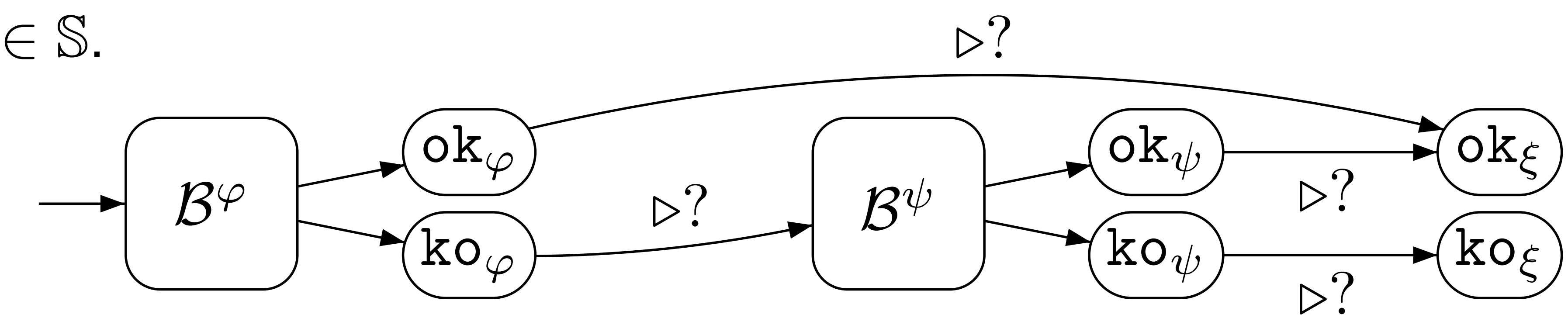
wFO to pebWA

$$\varphi ::= \top \mid P_a(x) \mid x = y \mid R_d(x,y) \mid R_d^+(x,y) \mid \neg\varphi \mid \varphi \vee \varphi \mid \exists x\, \varphi$$

$$\Phi ::= s \mid \varphi ?\, \Phi : \Phi \mid \Phi \oplus \Phi \mid \Phi \otimes \Phi \mid \bigoplus_x \Phi \mid \bigotimes_x \Phi$$

with $a \in A \cup \{\triangleright\}$, $d \in D$, $s \in \mathbb{S}$.

Disjunction/conjunction
$\xi = \varphi \vee \psi$



Exist

qu

next

Thm: Consider a searchable class of graphs.

For each wFO formula, we can construct an equivalent layered pebWA,
\# states linear in the size of the formula
\# layers = quantifier depth of the formula
\# pebble names = # variable names in the formula

Model Checking
Query Evaluation

Complexity of evaluating queries

Given a layered pebWA with p pebble names and a graph, we can compute $[\![\mathcal{A}]\!](G)$ with $O((p+1)|Q|^3|V|^{p+3})$ scalar operations (sum, product, star).

Reusability: *the number of layers (quantifier depth in formulas) may be much bigger than the number of pebble names (x and y, e.g.)*

Given a layered pebWA with p pebble names and a (nested) word, we can compute $[\![\mathcal{A}]\!](W)$ with $O((p+1)|Q|^3|W|^{p+1})$ scalar operations (sum, product, star).

strongly layered: *each layer is associated with a fixed pebble name:* $O(|Q|^3|W|^p)$

Getting rid of variables

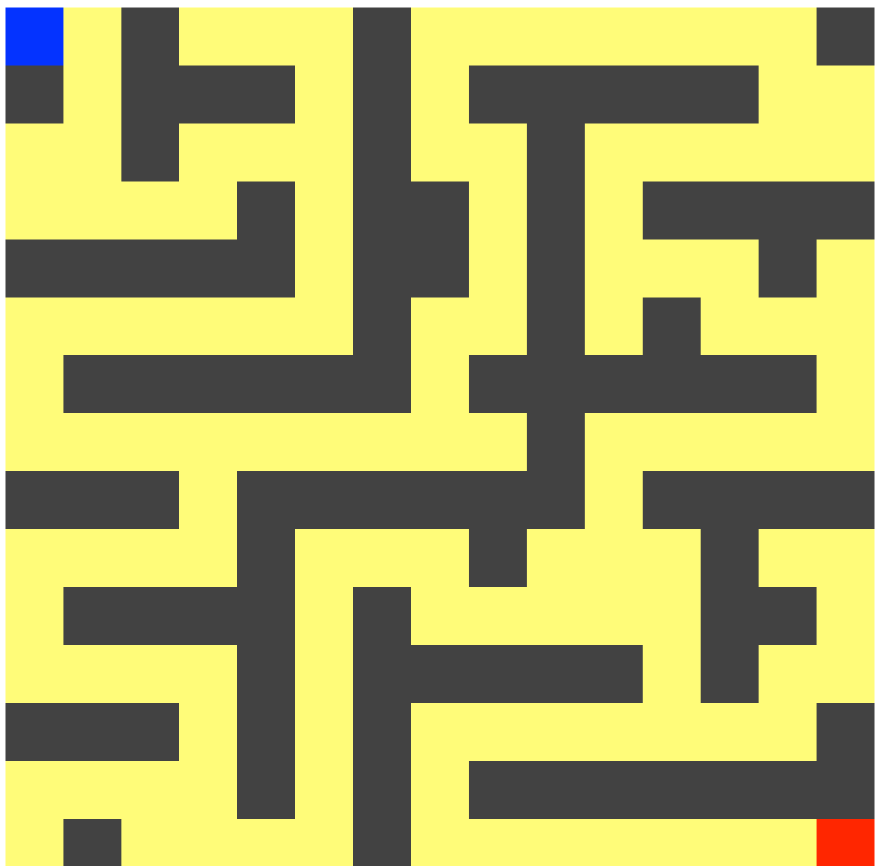
Weighted Expressions

Dynamic Logics (PDL)

Temporal Logics

Expressing more properties: path

In First-Order Logic, you cannot follow unbounded *complex* paths



A path: $\rightarrow\downarrow\downarrow\downarrow\rightarrow\rightarrow\uparrow$

A pattern: $(\downarrow + \rightarrow)^\star \cdot \uparrow \cdot (\downarrow + \leftarrow)^\star$

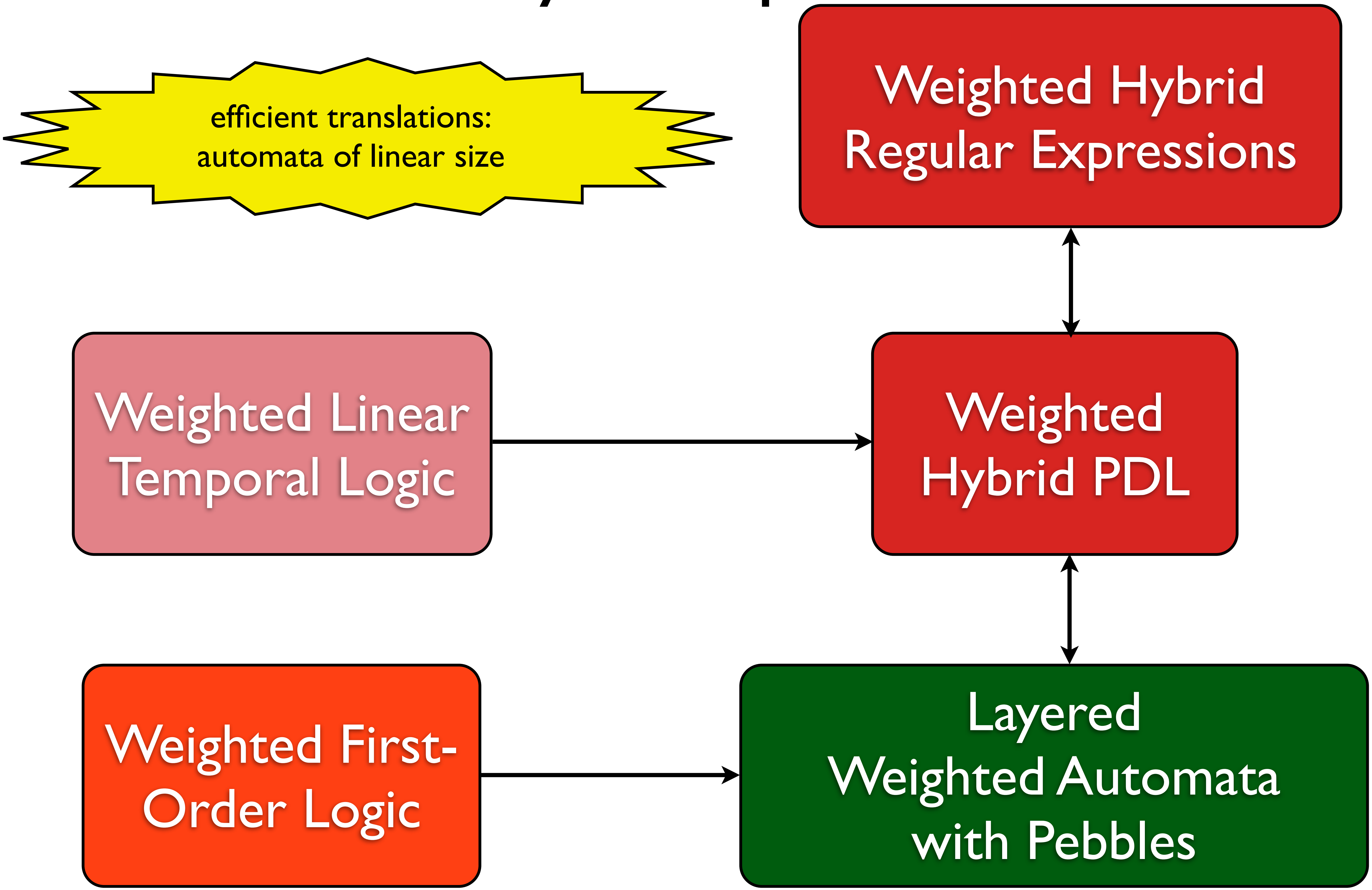
What is the length of the shortest path from $\textcolor{blue}{\blacksquare}$ to $\textcolor{red}{\blacksquare}$?

$(\mathbb{N} \cup \{+\infty\}, \min, +, +\infty, 0)$

$$\left(\left(\textcolor{blue}{\blacksquare}\,? \vee \textcolor{yellow}{\blacksquare}\,?\right) \cdot \textcolor{red}{1} \cdot (\rightarrow + \leftarrow + \downarrow + \uparrow)\right)^\star \cdot \textcolor{red}{\blacksquare}\,?$$

Weighted Expressions and ***Hybrid Weighted Expressions***
navigating over graphs

Summary of Expressiveness



Summary

Several ways of specifying quantitative properties of graphs:
regular expressions with pebbles,
weighted hybrid PDL, weighted temporal logic,
weighted first-order logic.

Efficient translation of the queries into pebWA

Efficient evaluation (model-checking of database community)

Perspectives

Extend wFO with a bounded transitive closure operator in order to have the same expressive power as pebWA.
(See next talk of Benjamin)

Add data or weights to the models (graphs)

Lift Model evaluation to System evaluation

Add comparisons of weights to the specification formalisms