Infinite graphs with decidable MSO theories

Madhavan Mukund

Chennai Mathematical Institute

92 G N Chetty Rd, Chennai 600 017, India

madhavan@cmi.ac.in

http://www.cmi.ac.in/~madhavan

Relational structures and FOL

- Relational structures $\mathcal{A} = (A, R_1^A, R_2^A, \ldots R_k^A)$

Relational structures and FOL

- Relational structures $\mathcal{A} = (A, R_1^A, R_2^A, \ldots R_k^A)$
  - A is the domain—assume countable

Relational structures and FOL

- Relational structures $\mathcal{A} = (A, R_1^A, R_2^A, \ldots R_k^A)$
  - A is the domain—assume countable
  - Each R_i^A is a relation on A, with arity n_i

Relational structures and FOL

- Relational structures $\mathcal{A} = (A, R_1^A, R_2^A, \ldots R_k^A)$
  - A is the domain—assume countable
  - Each R_i^A is a relation on A, with arity n_i
  - Example: $\mathcal{A} = (\mathbb{N}, 0, \text{succ}, <)$

Relational structures and FOL

- Relational structures $\mathcal{A} = (A, R_1^A, R_2^A, \ldots R_k^A)$
  - A is the domain—assume countable
  - Each R_i^A is a relation on A, with arity n_i
  - Example: $\mathcal{A} = (\mathbb{N}, 0, \text{succ}, <)$
- First order logic over $\mathcal{A}$

Relational structures and FOL

- Relational structures $\mathcal{A} = (A, R_1^A, R_2^A, \ldots R_k^A)$
  - A is the domain—assume countable
  - Each R_i^A is a relation on A, with arity n_i
  - Example: $\mathcal{A} = (\mathbb{N}, 0, \text{succ}, <)$
- First order logic over $\mathcal{A}$
  - Variables x, y that range over A

Relational structures and FOL

- Relational structures $\mathcal{A} = (A, R_1^A, R_2^A, \ldots R_k^A)$
  - A is the domain—assume countable
  - Each R_i^A is a relation on A, with arity n_i
  - Example: $\mathcal{A} = (\mathbb{N}, 0, \text{succ}, <)$
- First order logic over $\mathcal{A}$
  - Variables x, y that range over A
  - Relation symbol R_i for each underlying relation R_i^A

Relational structures and FOL

- Relational structures $\mathcal{A} = (A, R_1^A, R_2^A, \ldots R_k^A)$
  - A is the domain—assume countable
  - Each R_i^A is a relation on A, with arity n_i
  - Example: $\mathcal{A} = (\mathbb{N}, 0, \text{succ}, <)$
- First order logic over $\mathcal{A}$
  - Variables x, y that range over A
  - Relation symbol R_i for each underlying relation R_i^A
  - Propositional connectives $\neg$, $\vee$, $\wedge$, $\Rightarrow$, ...

Relational structures and FOL

- Relational structures $\mathcal{A} = (A, R_1^A, R_2^A, \ldots R_k^A)$
  - A is the domain—assume countable
  - Each R_i^A is a relation on A, with arity n_i
  - Example: $\mathcal{A} = (\mathbb{N}, 0, \text{succ}, <)$
- First order logic over $\mathcal{A}$
  - Variables x, y that range over A
  - Relation symbol R_i for each underlying relation R_i^A
  - Propositional connectives $\neg$, $\vee$, $\wedge$, $\Rightarrow$, ...
  - Quantifiers $\forall$, $\exists$

Relational structures and FOL

- Relational structures $\mathcal{A} = (A, R_1^A, R_2^A, \ldots R_k^A)$
  - A is the domain—assume countable
  - Each R_i^A is a relation on A, with arity n_i
  - Example: $\mathcal{A} = (\mathbb{N}, 0, \text{succ}, <)$
- First order logic over $\mathcal{A}$
  - Variables x, y that range over A
  - Relation symbol R_i for each underlying relation R_i^A
  - Propositional connectives $\neg$, $\vee$, $\wedge$, $\Rightarrow$, ...
  - Quantifiers $\forall$, $\exists$
  - Example: $\forall x \, \exists y \; x < y$, $\forall x \, \exists y \; y < x$

Monadic Second Order Logic

- Add set quantifiers $\forall X$, $\exists X$

Monadic Second Order Logic

- Add set quantifiers $\forall X$, $\exists X$
- Add atomic formulas $x \in Y$ (or $Y(x)$)

Monadic Second Order Logic

- Add set quantifiers $\forall X$, $\exists X$
- Add atomic formulas $x \in Y$ (or $Y(x)$)
  - Subsets are monadic predicates

Monadic Second Order Logic

- Add set quantifiers $\forall X$, $\exists X$
- Add atomic formulas $x \in Y$ (or $Y(x)$)
  - Subsets are monadic predicates
- Example:

Monadic Second Order Logic

- Add set quantifiers $\forall X$, $\exists X$
- Add atomic formulas $x \in Y$ (or $Y(x)$)
  - Subsets are monadic predicates
- Example:
  - $\text{less}(x, y) =$
    $\forall X\ [X(x) \wedge$
    $\qquad \forall u\ \forall v\ (X(u) \wedge \text{succ}(u, v)) \Rightarrow X(v)] \Rightarrow X(y)$

Monadic Second Order Logic

- Add set quantifiers $\forall X$, $\exists X$
- Add atomic formulas $x \in Y$ (or $Y(x)$)
  - Subsets are monadic predicates
- Example:
  - $\text{less}(x, y) =$
    $\forall X\ [X(x) \wedge$
    $\forall u\ \forall v\ (X(u) \wedge \text{succ}(u, v)) \Rightarrow X(v)] \Rightarrow X(y)$

    $<$ is expressible using succ in MSO

Monadic Second Order Logic

- Add set quantifiers $\forall X$, $\exists X$
- Add atomic formulas $x \in Y$ (or $Y(x)$)
  - Subsets are monadic predicates
- Example:
  - $\text{less}(x, y) = \forall X\, [X(x) \wedge \forall u\, \forall v\, (X(u) \wedge \text{succ}(u, v)) \Rightarrow X(v)] \Rightarrow X(y)$

    $<$ is expressible using succ in MSO
  - $\forall X\, [X(0) \wedge (\forall x\, \forall\, y (X(x) \wedge \text{succ}(x, y) \Rightarrow X(y)) \Rightarrow \forall z\, X(z)]$

Monadic Second Order Logic

- Add set quantifiers $\forall X$, $\exists X$
- Add atomic formulas $x \in Y$ (or $Y(x)$)
  - Subsets are monadic predicates
- Example:
  - $\text{less}(x, y) =$
    $\forall X \; [X(x) \wedge$
    $\forall u \, \forall v \; (X(u) \wedge \text{succ}(u, v)) \Rightarrow X(v)] \Rightarrow X(y)$

    $<$ is expressible using succ in MSO
  - $\forall X \; [X(0) \wedge (\forall x \, \forall \, y(X(x) \wedge \text{succ}(x, y) \Rightarrow X(y))$
    $\Rightarrow \forall z \; X(z)]$

    Principle of mathematical induction

When is MSO decidable?

- Given a structure $\mathcal{A} = (A, R_1^A, \ldots, R_k^A)$ and an MSO sentence φ, is φ true in $\mathcal{A}$?

When is MSO decidable?

- Given a structure $\mathcal{A} = (A, R_1^A, \ldots, R_k^A)$ and an MSO sentence φ, is φ true in $\mathcal{A}$?
  - In verification parlance, is the model checking problem for MSO formulas over $\mathcal{A}$ decidable?

When is MSO decidable?

- Given a structure $\mathcal{A} = (A, R_1^A, \ldots, R_k^A)$ and an MSO sentence φ, is φ true in $\mathcal{A}$?
  - In verification parlance, is the model checking problem for MSO formulas over $\mathcal{A}$ decidable?
- If A is finite, MSO is decidable

When is MSO decidable?

- Given a structure $\mathcal{A} = (A, R_1^A, \ldots, R_k^A)$ and an MSO sentence φ, is φ true in $\mathcal{A}$?
  - In verification parlance, is the model checking problem for MSO formulas over $\mathcal{A}$ decidable?
- If A is finite, MSO is decidable
  - Exhaustively enumerate all possibilities for quantifiers

When is MSO decidable?

- Given a structure $\mathcal{A} = (A, R_1^A, \ldots, R_k^A)$ and an MSO sentence φ, is φ true in A?
  - In verification parlance, is the model checking problem for MSO formulas over $\mathcal{A}$ decidable?
- If A is finite, MSO is decidable
  - Exhaustively enumerate all possibilities for quantifiers
- Theorem [Büchi 1960]
  MSO over $(\mathbb{N}, 0, \mathsf{succ})$ is decidable

When is MSO decidable?

- Given a structure $\mathcal{A} = (A, R_1^A, \ldots, R_k^A)$ and an MSO sentence φ, is φ true in A?
  - In verification parlance, is the model checking problem for MSO formulas over $\mathcal{A}$ decidable?
- If A is finite, MSO is decidable
  - Exhaustively enumerate all possibilities for quantifiers
- Theorem [Büchi 1960]
  MSO over $(\mathbb{N}, 0, \mathrm{succ})$ is decidable
  - S1S — Second order theory of 1 Successor

When is MSO decidable?

- Given a structure $\mathcal{A} = (A, R_1^A, \ldots, R_k^A)$ and an MSO sentence φ, is φ true in A?
  - In verification parlance, is the model checking problem for MSO formulas over $\mathcal{A}$ decidable?
- If A is finite, MSO is decidable
  - Exhaustively enumerate all possibilities for quantifiers
- Theorem [Büchi 1960]
  MSO over $(\mathbb{N}, 0, \text{succ})$ is decidable
  - S1S — Second order theory of 1 Successor
  - S1S formula $\varphi \mapsto$ (Büchi) automaton M_φ

When is MSO decidable?

- Given a structure $\mathcal{A} = (A, R_1^A, \ldots, R_k^A)$ and an MSO sentence φ, is φ true in $\mathcal{A}$?
  - In verification parlance, is the model checking problem for MSO formulas over $\mathcal{A}$ decidable?
- If A is finite, MSO is decidable
  - Exhaustively enumerate all possibilities for quantifiers
- Theorem [Büchi 1960]
  MSO over $(\mathbb{N}, 0, \text{succ})$ is decidable
  - S1S — Second order theory of 1 Successor
  - S1S formula $\varphi \mapsto$ (Büchi) automaton M_φ
  - φ is satisfiable iff $L(M_\varphi)$ is nonempty

When is MSO decidable . . .

- Theorem [Rabin 1969]
  MSO over the infinite binary tree is decidable

When is MSO decidable . . .

- Theorem [Rabin 1969]
  MSO over the infinite binary tree is decidable
  - $T_2 = \{0, 1\}^*$ — nodes of infinite binary tree

When is MSO decidable . . .

- Theorem [Rabin 1969]
  MSO over the infinite binary tree is decidable
  - $T_2 = \{0, 1\}^*$ — nodes of infinite binary tree
  - Relations S_0, S_1 — left and right child

When is MSO decidable . . .

- Theorem [Rabin 1969]
  MSO over the infinite binary tree is decidable
  - $T_2 = \{0, 1\}^*$ — nodes of infinite binary tree
  - Relations S_0, S_1 — left and right child
  - S2S — Second order theory of 2 Successors

When is MSO decidable . . .

- Theorem [Rabin 1969]
  MSO over the infinite binary tree is decidable
  - $T_2 = \{0, 1\}^*$ — nodes of infinite binary tree
  - Relations S_0, S_1 — left and right child
  - S2S — Second order theory of 2 Successors
  - Satisfiability reduces to emptiness for tree automata

When is MSO decidable . . .

- Theorem [Rabin 1969]
  MSO over the infinite binary tree is decidable
  - $T_2 = \{0,1\}^*$ — nodes of infinite binary tree
  - Relations S_0, S_1 — left and right child
  - S2S — Second order theory of 2 Successors
  - Satisfiability reduces to emptiness for tree automata
- Corollary [Rabin 1969]

When is MSO decidable . . .

- Theorem [Rabin 1969]
  MSO over the infinite binary tree is decidable
  - $T_2 = \{0, 1\}^*$ — nodes of infinite binary tree
  - Relations S_0, S_1 — left and right child
  - S2S — Second order theory of 2 Successors
  - Satisfiability reduces to emptiness for tree automata
- Corollary [Rabin 1969]
  - SnS is decidable for all n

When is MSO decidable . . .

- Theorem [Rabin 1969]
  MSO over the infinite binary tree is decidable
  - $T_2 = \{0, 1\}^*$ — nodes of infinite binary tree
  - Relations S_0, S_1 — left and right child
  - S2S — Second order theory of 2 Successors
  - Satisfiability reduces to emptiness for tree automata
- Corollary [Rabin 1969]
  - SnS is decidable for all n
  - SωS is decidable

When is MSO decidable . . .

- Theorem [Rabin 1969]
  MSO over the infinite binary tree is decidable
  - $T_2 = \{0, 1\}^*$ — nodes of infinite binary tree
  - Relations S_0, S_1 — left and right child
  - S2S — Second order theory of 2 Successors
  - Satisfiability reduces to emptiness for tree automata
- Corollary [Rabin 1969]
  - SnS is decidable for all n
  - SωS is decidable
  - MSO over dense linear orders is decidable

When is MSO decidable . . .

- Theorem [Rabin 1969]
  MSO over the infinite binary tree is decidable
  - $T_2 = \{0, 1\}^*$ — nodes of infinite binary tree
  - Relations S_0, S_1 — left and right child
  - S2S — Second order theory of 2 Successors
  - Satisfiability reduces to emptiness for tree automata
- Corollary [Rabin 1969]
  - SnS is decidable for all n
  - SωS is decidable
  - MSO over dense linear orders is decidable
  - . . .

When is MSO decidable . . .

- Theorem [Rabin 1969]
  MSO over the infinite binary tree is decidable
  - $T_2 = \{0, 1\}^*$ — nodes of infinite binary tree
  - Relations S_0, S_1 — left and right child
  - S2S — Second order theory of 2 Successors
  - Satisfiability reduces to emptiness for tree automata
- Corollary [Rabin 1969]
  - SnS is decidable for all n
  - SωS is decidable
  - MSO over dense linear orders is decidable
  - . . .
  - All follow by MSO interpretations in S2S

MSO interpretations: An example

- S3S, MSO over complete ternary tree T_3 is decidable

MSO interpretations: An example

- S3S, MSO over complete ternary tree $\boldsymbol{T_3}$ is decidable
- Consider vertices $\boldsymbol{T = (10 + 110 + 1110)^*}$ in $\boldsymbol{T_2}$

MSO interpretations: An example

- S3S, MSO over complete ternary tree T_3 is decidable
- Consider vertices $T = (10 + 110 + 1110)^*$ in T_2
- Nodes in T: $1^{i_1}0 \ldots 1^{i_m}0$, with $i_1, \ldots, i_m \in \{1, 2, 3\}$

MSO interpretations: An example

- S3S, MSO over complete ternary tree T_3 is decidable
- Consider vertices $T = (10 + 110 + 1110)^*$ in T_2
- Nodes in T: $1^{i_1}0 \ldots 1^{i_m}0$, with $i_1, \ldots, i_m \in \{1, 2, 3\}$
- Represents the node $(i_1 - 1) \ldots (i_m - 1)$ in T_3

MSO interpretations: An example

- S3S, MSO over complete ternary tree T_3 is decidable
- Consider vertices $T = (10 + 110 + 1110)^*$ in T_2
- Nodes in T: $1^{i_1}0 \dots 1^{i_m}0$, with $i_1, \dots, i_m \in \{1, 2, 3\}$
- Represents the node $(i_1 - 1) \dots (i_m - 1)$ in T_3
- In S2S $$T(x) = \forall Y [Y(x) \wedge \forall y((Y(y10) \vee Y(y110) \vee Y(y1110)) \Rightarrow Y(y)) \Rightarrow Y(\epsilon)]$$

MSO interpretations: An example

- S3S, MSO over complete ternary tree T_3 is decidable
- Consider vertices $T = (10 + 110 + 1110)^*$ in T_2
- Nodes in T: $1^{i_1}0 \ldots 1^{i_m}0$, with $i_1, \ldots, i_m \in \{1, 2, 3\}$
- Represents the node $(i_1 - 1) \ldots (i_m - 1)$ in T_3
- In S2S

$$\begin{aligned} T(x) = \forall Y [Y(x) \wedge \\ \forall y ((Y(y10) \vee Y(y110) \vee \\ Y(y1110)) \Rightarrow Y(y)) \\ \Rightarrow Y(\epsilon)] \end{aligned}$$

- Translate S3S formulas over T_3 into S2S formulas over $T \subseteq T_2$

MSO interpretations: An example . . .

- Successor relations S_0, S_1, S_2 of T_3

MSO interpretations: An example . . .

- Successor relations S_0,S_1,S_2 of T_3

$$\psi_0(x, y) = \exists z(S_1(x, z) \wedge S_0(z, y))$$
$$\psi_1(x, y) = \exists u \exists v(S_1(x, u) \wedge S_1(u, v) \wedge S_0(v, y))$$
$$\psi_2(x, y) = \ldots$$

MSO interpretations: An example . . .

- Successor relations S_0,S_1,S_2 of T_3

$$\psi_0(x, y) = \exists z(S_1(x, z) \wedge S_0(z, y))$$
$$\psi_1(x, y) = \exists u \exists v(S_1(x, u) \wedge S_1(u, v) \wedge S_0(v, y))$$
$$\psi_2(x, y) = \ldots$$

- Relativize quantifiers

MSO interpretations: An example . . .

- Successor relations S_0,S_1,S_2 of T_3

$$\psi_0(x, y) = \exists z(S_1(x, z) \wedge S_0(z, y))$$
$$\psi_1(x, y) = \exists u \exists v(S_1(x, u) \wedge S_1(u, v) \wedge S_0(v, y))$$
$$\psi_2(x, y) = \ldots$$

- Relativize quantifiers

$$\forall x \varphi(x) \text{ in S3S} \mapsto \forall x(T(x) \Rightarrow \tilde{\varphi}(x)) \text{ in S2S}$$

MSO interpretations: An example . . .

- Successor relations S_0,S_1,S_2 of T_3

$$\psi_0(x,y) = \exists z(S_1(x,z) \wedge S_0(z,y))$$
$$\psi_1(x,y) = \exists u \exists v(S_1(x,u) \wedge S_1(u,v) \wedge S_0(v,y))$$
$$\psi_2(x,y) = \ldots$$

- Relativize quantifiers

$\forall x \varphi(x)$ in S3S $\mapsto$ $\forall x(T(x) \Rightarrow \tilde{\varphi}(x))$ in S2S

$\exists X \varphi(X)$ in S3S $\mapsto$ $\exists X(X \subseteq T \wedge \tilde{\varphi}(X))$ in S2S

MSO interpretations

- In general, an MSO interpretation of structure $\mathcal{A}$ in structure $\mathcal{B}$ consists of •

MSO interpretations

- In general, an MSO interpretation of structure $\mathcal{A}$ in structure $\mathcal{B}$ consists of
  - Mapping the domain of $\mathcal{A}$ into a subset of the domain of $\mathcal{B}$ by a domain formula

MSO interpretations

- In general, an MSO interpretation of structure $\mathcal{A}$ in structure $\mathcal{B}$ consists of
  - Mapping the domain of $\mathcal{A}$ into a subset of the domain of $\mathcal{B}$ by a domain formula

    In the example, T_3 was mapped to $T \subseteq T_2$

MSO interpretations

- In general, an MSO interpretation of structure $\mathcal{A}$ in structure $\mathcal{B}$ consists of
  - Mapping the domain of $\mathcal{A}$ into a subset of the domain of $\mathcal{B}$ by a domain formula

    In the example, T_3 was mapped to $T \subseteq T_2$
  - Mapping each relation R_i of $\mathcal{A}$ into an "isomorphic" relation over the subset defined by the domain formula

MSO interpretations

- In general, an MSO interpretation of structure $\mathcal{A}$ in structure $\mathcal{B}$ consists of
  - Mapping the domain of $\mathcal{A}$ into a subset of the domain of $\mathcal{B}$ by a domain formula

    In the example, T_3 was mapped to $T \subseteq T_2$
  - Mapping each relation R_i of $\mathcal{A}$ into an "isomorphic" relation over the subset defined by the domain formula

    In the example, each successor relation S_i over T_3 was mapped to a relation ψ_i over $T \subseteq T_2$

MSO interpretations

- In general, an MSO interpretation of structure $\mathcal{A}$ in structure $\mathcal{B}$ consists of
  - Mapping the domain of $\mathcal{A}$ into a subset of the domain of $\mathcal{B}$ by a domain formula

    In the example, T_3 was mapped to $T \subseteq T_2$
  - Mapping each relation R_i of $\mathcal{A}$ into an "isomorphic" relation over the subset defined by the domain formula

    In the example, each successor relation S_i over T_3 was mapped to a relation ψ_i over $T \subseteq T_2$

Proposition If $\mathcal{A}$ is MSO-interpretable in $\mathcal{B}$ and MSO is decidable over $\mathcal{B}$ then MSO is decidable over $\mathcal{A}$

MSO interpretations: results

- Theorem [Muller-Schupp 1985]
  MSO is decidable over pushdown graphs.

MSO interpretations: results

- Theorem [Muller-Schupp 1985]
  MSO is decidable over pushdown graphs.
- Theorem [Caucal 1996/2003]
  MSO is decidable over prefix-recognizable graphs.

MSO interpretations: results

- Theorem [Muller-Schupp 1985]
  MSO is decidable over pushdown graphs.
- Theorem [Caucal 1996/2003]
  MSO is decidable over prefix-recognizable graphs.
- Both results can be got from MSO interpretations into MSO over the tree T_m, for appropriate m

MSO interpretations: results

- Theorem [Muller-Schupp 1985]
  MSO is decidable over pushdown graphs.
- Theorem [Caucal 1996/2003]
  MSO is decidable over prefix-recognizable graphs.
- Both results can be got from MSO interpretations into MSO over the tree T_m, for appropriate m
- For pushdown graphs, choose m to be number of states plus size of stack alphabet.

MSO interpretations: results

- Theorem [Muller-Schupp 1985]
  MSO is decidable over pushdown graphs.
- Theorem [Caucal 1996/2003]
  MSO is decidable over prefix-recognizable graphs.
- Both results can be got from MSO interpretations into MSO over the tree T_m, for appropriate m
- For pushdown graphs, choose m to be number of states plus size of stack alphabet.
- For prefix-recognizable graphs, choose m to be the size of the alphabet.

Unfolding graph structures

- Graphs with edge labels I and vertex labels J

Unfolding graph structures

- Graphs with edge labels I and vertex labels J
- $G = (V, (E_i)_{i \in I}, (P_j)_{j \in J})$

Unfolding graph structures

- Graphs with edge labels I and vertex labels J
- $G = (V, (E_i)_{i \in I}, (P_j)_{j \in J})$
- Unfold G from $v_0 \in V$ into
  $G' = (V', (E'_i)_{i \in I}, (P'_j)_{j \in J})$

Unfolding graph structures

- Graphs with edge labels I and vertex labels J
- $G = (V, (E_i)_{i \in I}, (P_j)_{j \in J})$
- Unfold G from $v_0 \in V$ into
  $G' = (V', (E'_i)_{i \in I}, (P'_j)_{j \in J})$
  - V' : all paths $v_0 i_1 v_1 \ldots i_k v_k$

Unfolding graph structures

- Graphs with edge labels I and vertex labels J
- $G = (V, (E_i)_{i \in I}, (P_j)_{j \in J})$
- Unfold G from $v_0 \in V$ into
  $G' = (V', (E'_i)_{i \in I}, (P'_j)_{j \in J})$
  - V' : all paths $v_0 i_1 v_1 \ldots i_k v_k$
  - $(p, q) \in E'_i$ iff q extends p by edge from E_i

Unfolding graph structures

- Graphs with edge labels I and vertex labels J
- $G = (V, (E_i)_{i \in I}, (P_j)_{j \in J})$
- Unfold G from $v_0 \in V$ into
  $G' = (V', (E'_i)_{i \in I}, (P'_j)_{j \in J})$
  - V' : all paths $v_0 i_1 v_1 \ldots i_k v_k$
  - $(p, q) \in E'_i$ iff q extends p by edge from E_i
  - $p \in P'_j$ iff last vertex in p is in P_j

Unfolding graph structures

- Graphs with edge labels I and vertex labels J
- $G = (V, (E_i)_{i \in I}, (P_j)_{j \in J})$
- Unfold G from $v_0 \in V$ into $G' = (V', (E'_i)_{i \in I}, (P'_j)_{j \in J})$
  - V' : all paths $v_0 i_1 v_1 \ldots i_k v_k$
  - $(p, q) \in E'_i$ iff q extends p by edge from E_i
  - $p \in P'_j$ iff last vertex in p is in P_j
- Example: $G_0 = (\{v_0\}, E_0 = E_1 = \{(v_0, v_0)\})$

Unfolding graph structures

- Graphs with edge labels I and vertex labels J
- $G = (V, (E_i)_{i \in I}, (P_j)_{j \in J})$
- Unfold G from $v_0 \in V$ into
  $G' = (V', (E'_i)_{i \in I}, (P'_j)_{j \in J})$
  - V' : all paths $v_0 i_1 v_1 \ldots i_k v_k$
  - $(p, q) \in E'_i$ iff q extends p by edge from E_i
  - $p \in P'_j$ iff last vertex in p is in P_j
- Example: $G_0 = (\{v_0\}, E_0 = E_1 = \{(v_0, v_0)\})$

  Unfolding of G_0 is the binary tree T_2

Unfolding graphs

- Theorem [Courcelle and Walukiewicz, 1998]

  If MSO is decidable for a graph, then MSO is also decidable for its unfolding from any MSO-definable vertex.

Unfolding graphs

- Theorem [Courcelle and Walukiewicz, 1998]

  If MSO is decidable for a graph, then MSO is also decidable for its unfolding from any MSO-definable vertex.

- Decidability of S2S follows from trivial decidability of MSO over G_0!

Unfolding graphs

- Theorem [Courcelle and Walukiewicz, 1998]

  If MSO is decidable for a graph, then MSO is also decidable for its unfolding from any MSO-definable vertex.

- Decidability of S2S follows from trivial decidability of MSO over G_0!

- Theorem also holds for a different type of unfolding called tree iteration

  Due to [Muchnik (reported by Semenov 1985)] and [Walukiewicz 2002]

The Caucal hierarchy [Caucal, 2002]

- $\mathcal{T}_0$ = the class of finite trees

The Caucal hierarchy [Caucal, 2002]

- $\mathcal{T}_0$ = the class of finite trees
- $\mathcal{G}_n$ = the class of graphs which are MSO-interpretable in a tree of $\mathcal{T}_n$

The Caucal hierarchy [Caucal, 2002]

- $\mathcal{T}_0$ = the class of finite trees
- $\mathcal{G}_n$ = the class of graphs which are MSO-interpretable in a tree of $\mathcal{T}_n$
- $\mathcal{T}_{n+1}$ = the class of unfoldings of graphs in $\mathcal{G}_n$

The Caucal hierarchy [Caucal, 2002]

- $\mathcal{T}_0$ = the class of finite trees
- $\mathcal{G}_n$ = the class of graphs which are MSO-interpretable in a tree of $\mathcal{T}_n$
- $\mathcal{T}_{n+1}$ = the class of unfoldings of graphs in $\mathcal{G}_n$
- MSO is decidable for each structure in the Caucal hierarchy

The Caucal hierarchy [Caucal, 2002]

- $\mathcal{T}_0$ = the class of finite trees
- $\mathcal{G}_n$ = the class of graphs which are MSO-interpretable in a tree of $\mathcal{T}_n$
- $\mathcal{T}_{n+1}$ = the class of unfoldings of graphs in $\mathcal{G}_n$
- MSO is decidable for each structure in the Caucal hierarchy
  - Trivially for finite trees in $\mathcal{T}_0$

The Caucal hierarchy [Caucal, 2002]

- $\mathcal{T}_0$ = the class of finite trees
- $\mathcal{G}_n$ = the class of graphs which are MSO-interpretable in a tree of $\mathcal{T}_n$
- $\mathcal{T}_{n+1}$ = the class of unfoldings of graphs in $\mathcal{G}_n$
- MSO is decidable for each structure in the Caucal hierarchy
  - Trivially for finite trees in $\mathcal{T}_0$
  - For higher levels, follows from what we have seen so far

The Caucal hierarchy [Caucal, 2002]

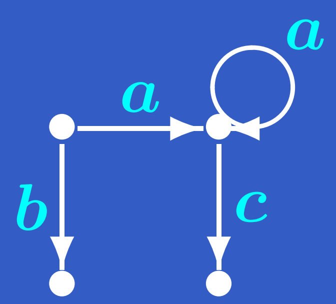
- $\mathcal{T}_0$ = the class of finite trees
- $\mathcal{G}_n$ = the class of graphs which are MSO-interpretable in a tree of $\mathcal{T}_n$
- $\mathcal{T}_{n+1}$ = the class of unfoldings of graphs in $\mathcal{G}_n$
- MSO is decidable for each structure in the Caucal hierarchy
  - Trivially for finite trees in $\mathcal{T}_0$
  - For higher levels, follows from what we have seen so far
- $\mathcal{G}_0$ is the class of finite graphs

The Caucal hierarchy [Caucal, 2002]

- $\mathcal{T}_0$ = the class of finite trees
- $\mathcal{G}_n$ = the class of graphs which are MSO-interpretable in a tree of $\mathcal{T}_n$
- $\mathcal{T}_{n+1}$ = the class of unfoldings of graphs in $\mathcal{G}_n$
- MSO is decidable for each structure in the Caucal hierarchy
  - Trivially for finite trees in $\mathcal{T}_0$
  - For higher levels, follows from what we have seen so far
- $\mathcal{G}_0$ is the class of finite graphs
- $\mathcal{T}_1$ is the class of regular trees

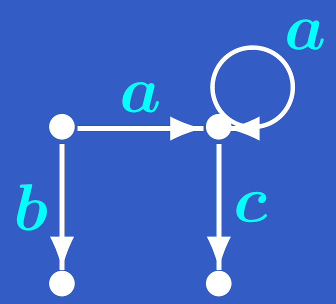
The Caucal hierarchy, by example

A finite graph in $\mathcal{G}_0$ . . .

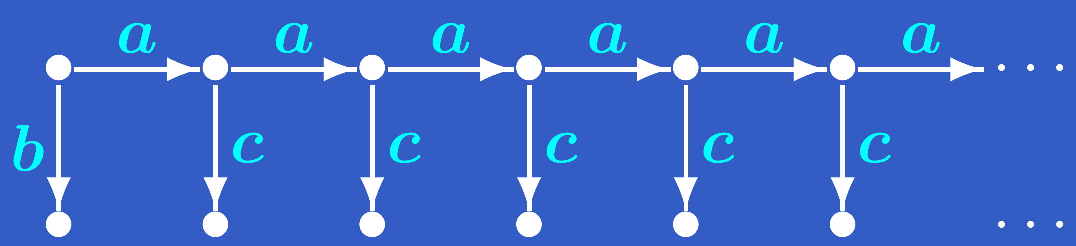


The Caucal hierarchy, by example

A finite graph in $\mathcal{G}_0$ . . .

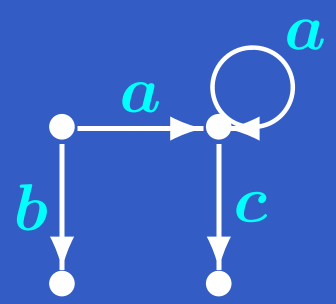


. . . its unfolding in $\mathcal{T}_1$ . . .

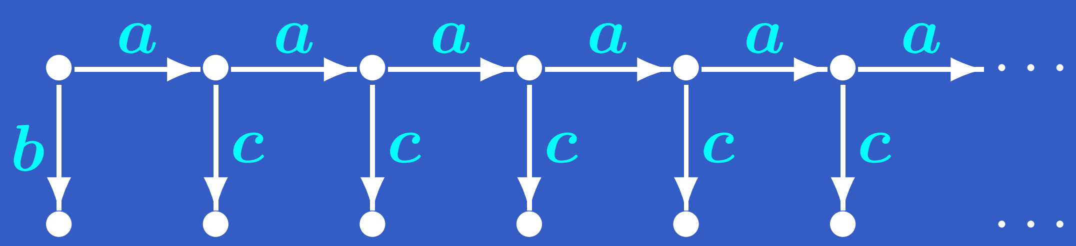


The Caucal hierarchy, by example

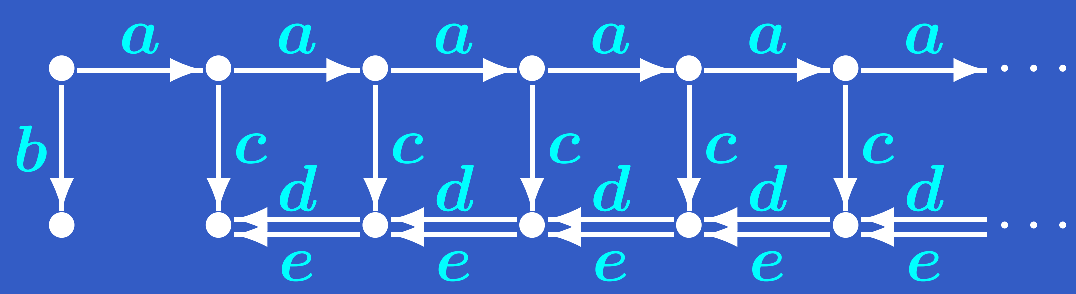
A finite graph in $\mathcal{G}_0$ . . .



. . . its unfolding in $\mathcal{T}_1$ . . .

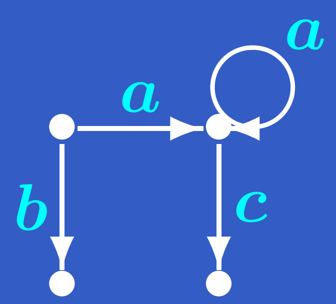


. . . and a pushdown graph in $\mathcal{G}_1$ by MSO-interpretation in the unfolding . . .

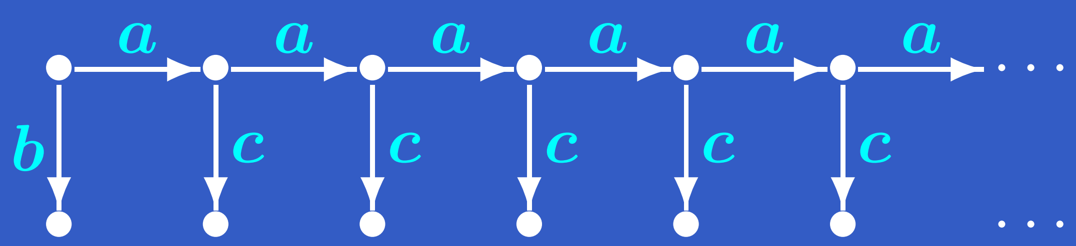


The Caucal hierarchy, by example

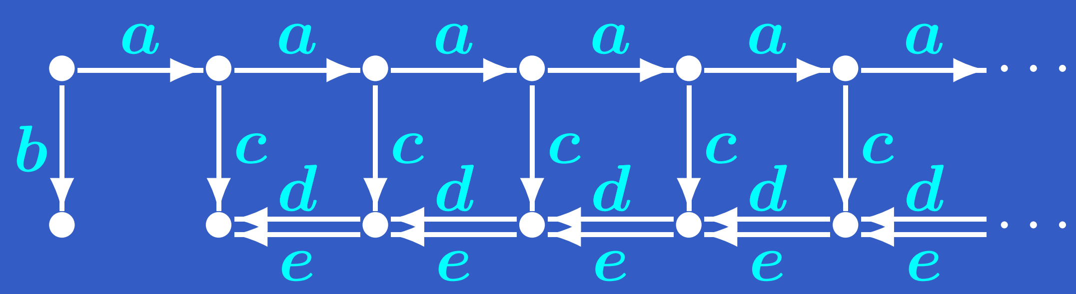
A finite graph in $\mathcal{G}_0$ ...



...its unfolding in $\mathcal{T}_1$ ...



...and a pushdown graph in $\mathcal{G}_1$ by MSO-interpretation in the unfolding ...

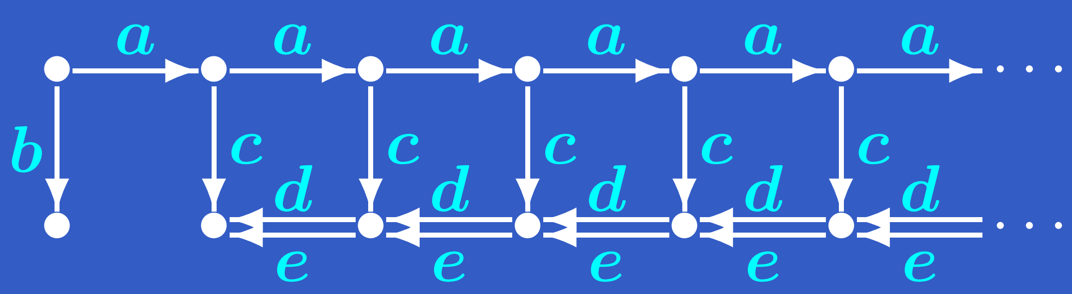


$\psi_d(x, y)$=

$$\psi_e(x, y) = \exists z \exists z' (E_a(z, z') \wedge E_c(z, y) \wedge E_c(z', x))$$

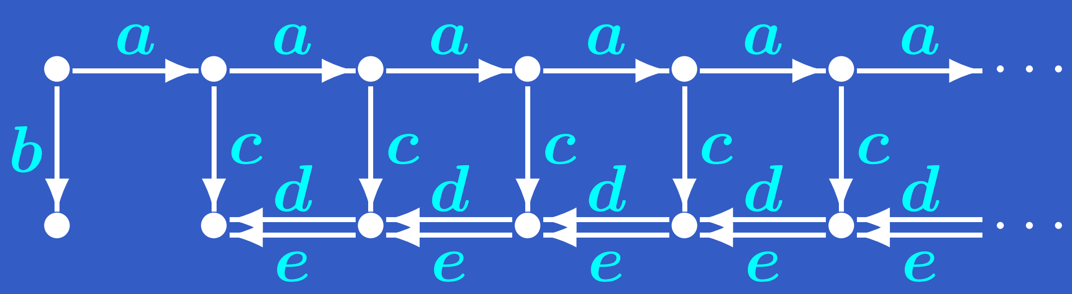
The Caucal hierarchy, by example . . .

If we unfold

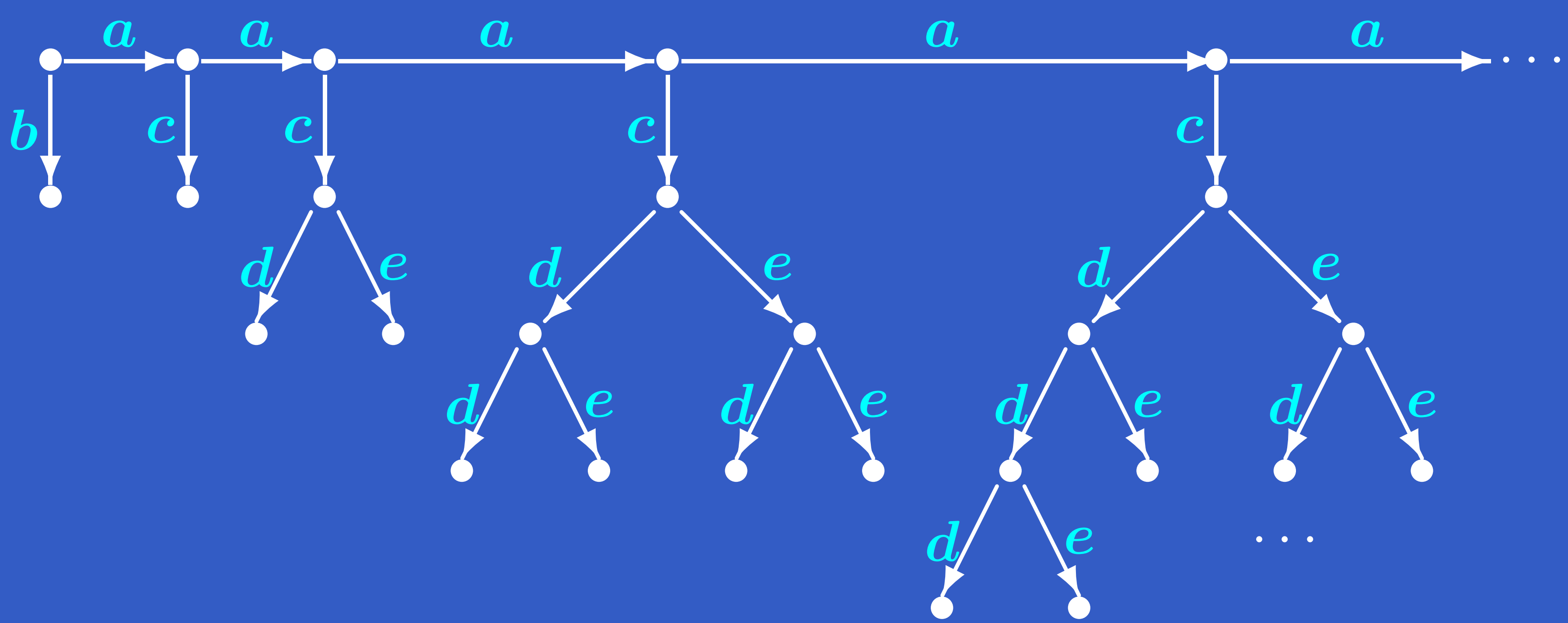


The Caucal hierarchy, by example . . .

If we unfold

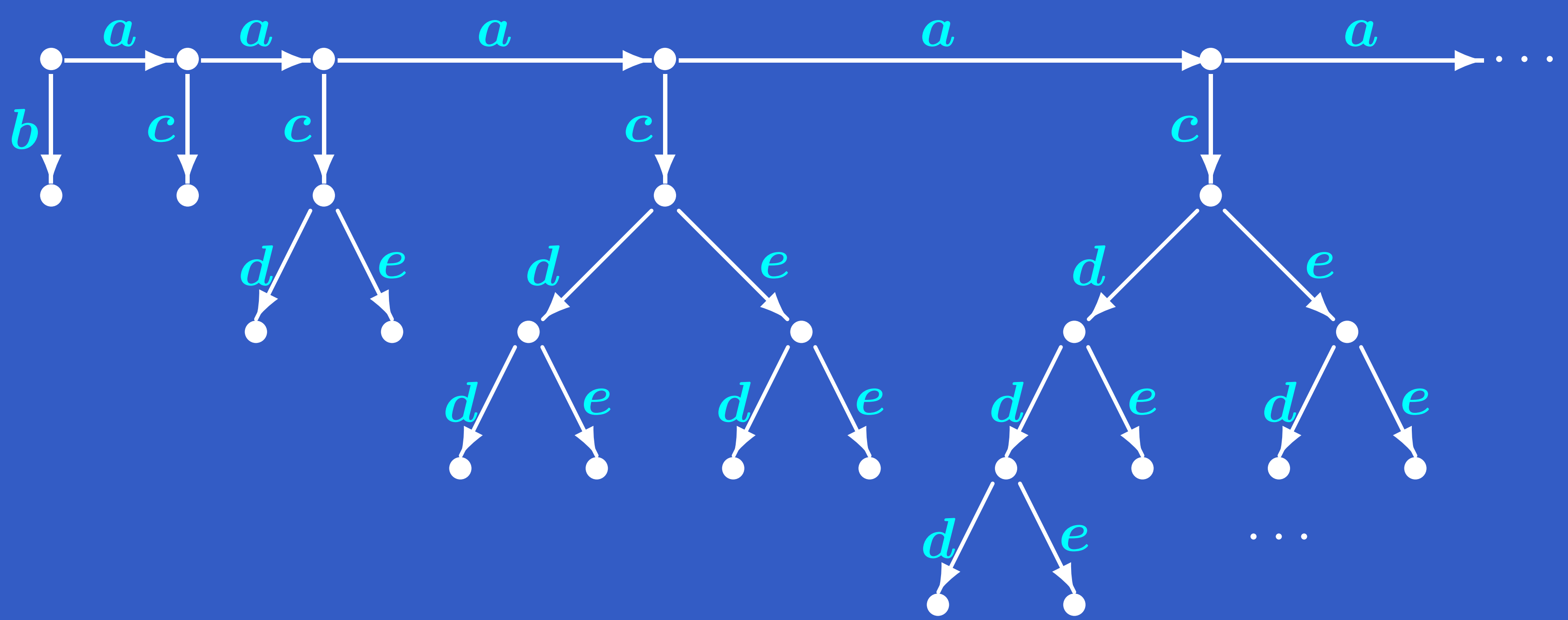


we get a tree in $\mathcal{T}_2$



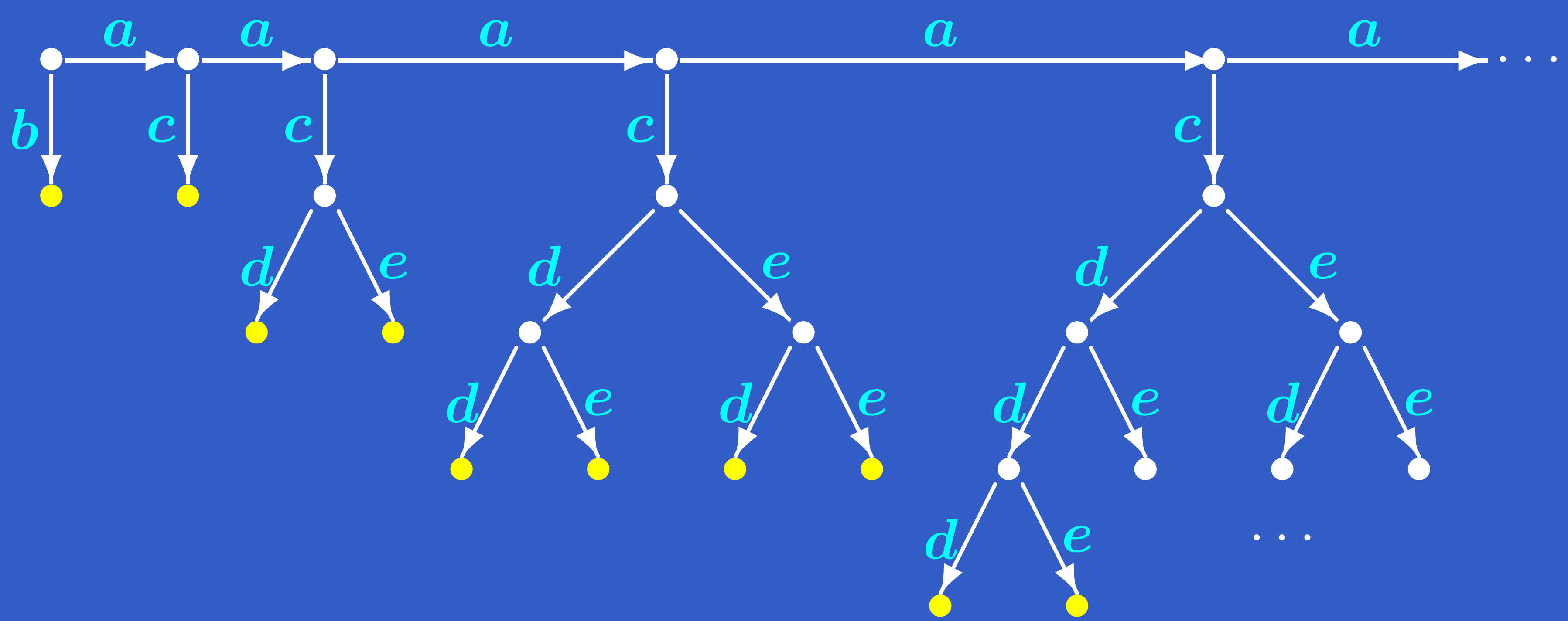
The Caucal hierarchy, by example . . .

- By an MSO-interpretation, we can identify a graph in $\mathcal{G}_2$ at the leaves of this tree



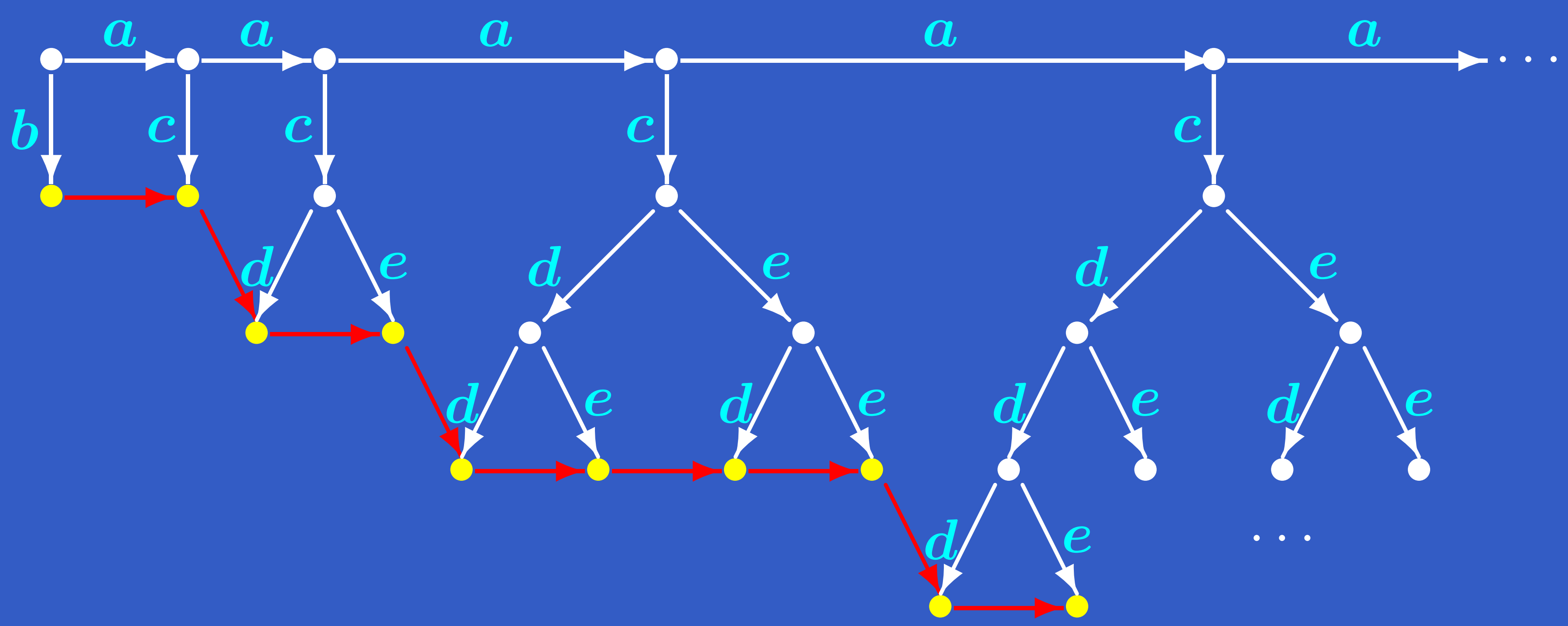
The Caucal hierarchy, by example . . .

- By an MSO-interpretation, we can identify a graph in $\mathcal{G}_2$ at the leaves of this tree



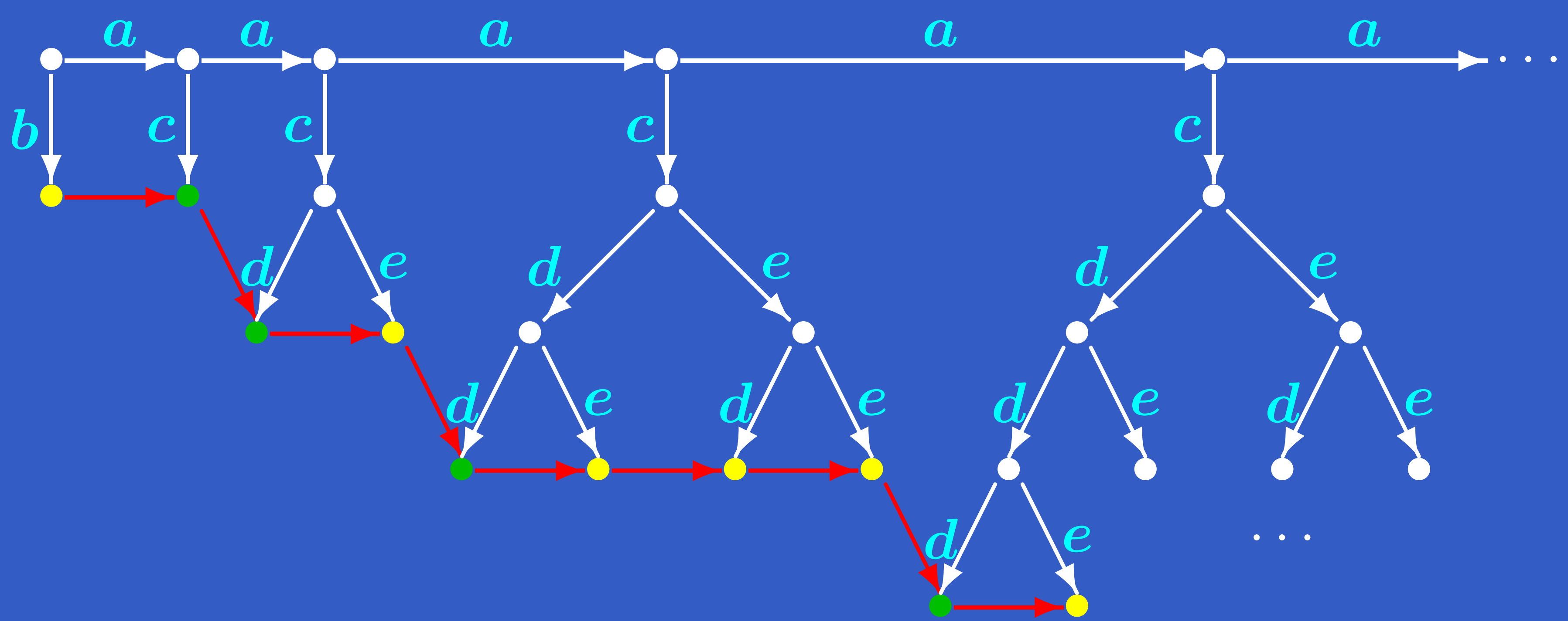
The Caucal hierarchy, by example . . .

- By an MSO-interpretation, we can identify a graph in $\mathcal{G}_2$ at the leaves of this tree



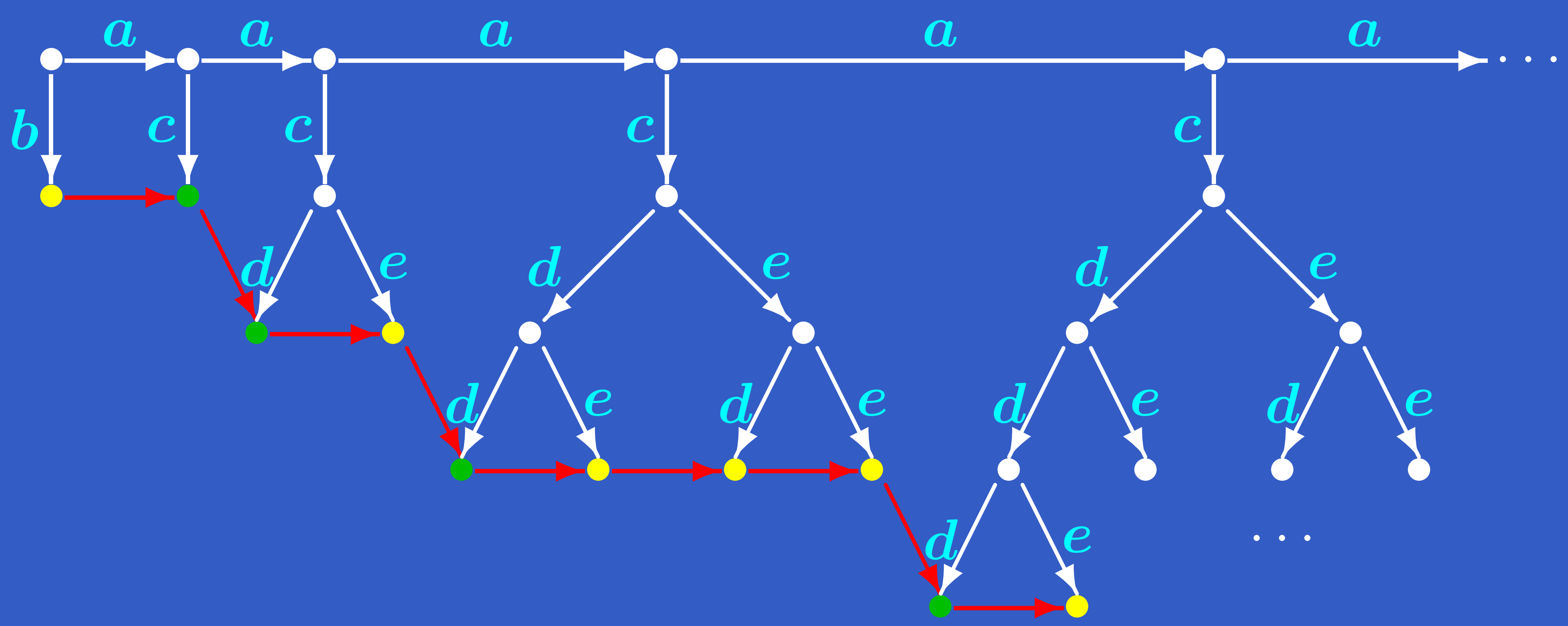
The Caucal hierarchy, by example . . .

- By an MSO-interpretation, we can identify a graph in $\mathcal{G}_2$ at the leaves of this tree



The Caucal hierarchy, by example . . .

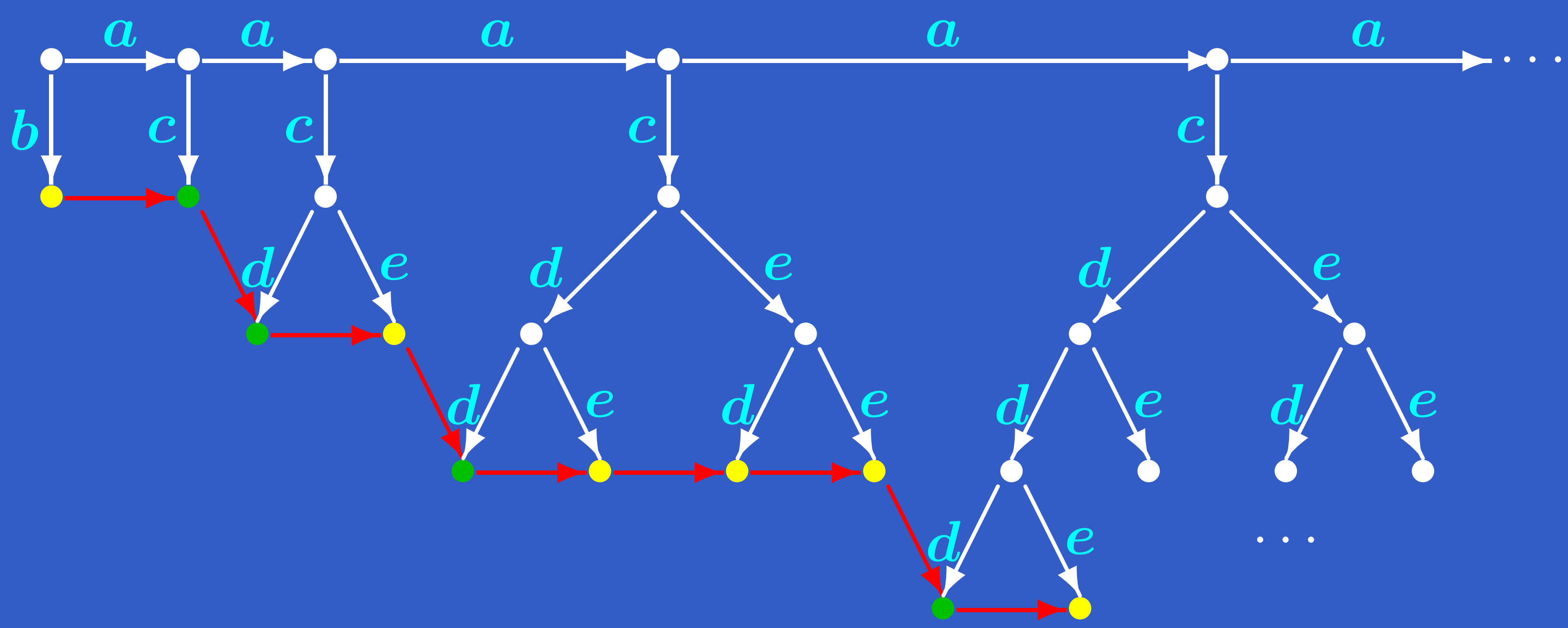
- By an MSO-interpretation, we can identify a graph in $\mathcal{G}_2$ at the leaves of this tree



- This is isomorphic to the structure $(\mathbb{N}, \text{succ}, P_2)$, where P_2 is the predicate powers of two

The Caucal hierarchy, by example . . .

- By an MSO-interpretation, we can identify a graph in $\mathcal{G}_2$ at the leaves of this tree



- This is isomorphic to the structure $(\mathbb{N}, \text{succ}, P_2)$, where P_2 is the predicate powers of two
- Original proof of decidability of MSO for $(\mathbb{N}, \text{succ}, P_2)$ by [Elgot and Rabin, 1966] was "non uniform"!

Constructing Infinite Graphs with a Decidable MSO-Theory

Wolfgang Thomas

Invited talk, MFCS 2003

The paper is available from Wolfgang Thomas's webpage.