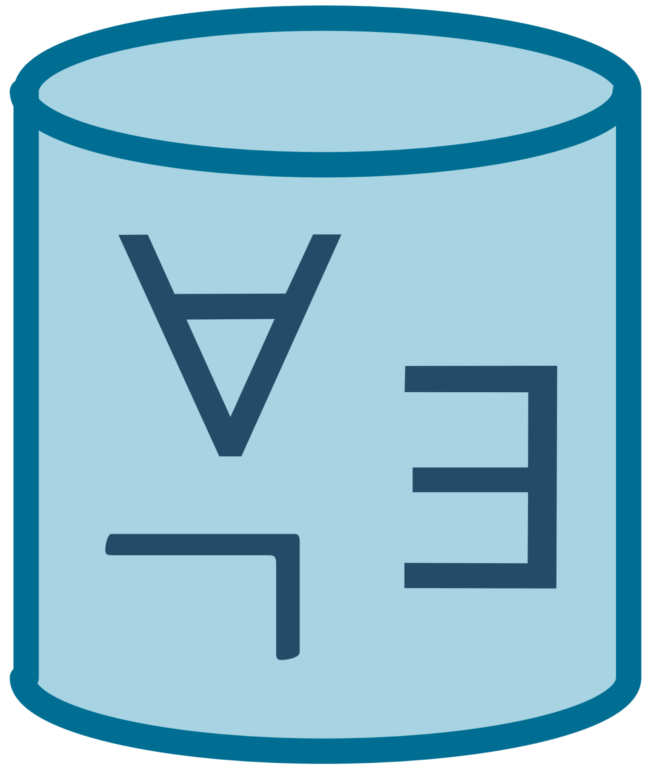
20/7/2015

ECI 2015
Buenos Aires



Fundamentos lógicos de bases de datos
(Logical foundations of databases)

Diego Figueira **Gabriele Puppis**

CNRS LaBRI

0-1 Law

A different perspective: a coarser view on expressiveness...

0-1 Law

A different perspective: a coarser view on expressiveness...

How do FO properties distribute among ALL structures?

Or equally, what percentage of graphs verify a given FO sentence?

0-1 Law

$\mu_n(\mathbf{P})$ = "the probability that a graph with ***n*** nodes satisfies property **P**"

0-1 Law

$\mu_n(\mathbf{P})$ = "the probability that a graph with n nodes satisfies property **P**"

Uniform distribution

(each pair of nodes has an edge with probability ½)

0-1 Law

$\mu_n(\mathbf{P})$ = "the probability that a graph with n nodes satisfies property **P**"

Uniform distribution

(each pair of nodes has an edge with probability ½)

C_n = { graphs with n nodes }

$$\mu_n(\mathbf{P}) = \frac{|\{G \in C_n \mid G \models \mathbf{P}\}|}{|C_n|}$$

$$|C_n| = 2^{n^2}$$

0-1 Law

$\mu_n(\mathbf{P})$ = "the probability that a graph with n nodes satisfies property **P**"

Uniform distribution

(each pair of nodes has an edge with probability ½)

C_n = { graphs with n nodes }

$$\mu_n(\mathbf{P}) = \frac{|\{G \in C_n \mid G \models \mathbf{P}\}|}{|C_n|}$$

$$|C_n| = 2^{n^2}$$

E.g. for **P** = "the graph is complete"

$$\mu_3(\mathbf{P}) = \frac{1}{|C_3|} = \frac{1}{2^{3^2}}$$

0-1 Law

$\mu_n(\mathbf{P})$ = "the probability that a graph with ***n*** nodes satisfies property **P**"

Uniform distribution

(each pair of nodes has an edge with probability ½)

C_n = { graphs with n nodes }

$$\mu_n(\mathbf{P}) = \frac{|\{G \in C_n \mid G \models \mathbf{P}\}|}{|C_n|}$$

$$|C_n| = 2^{n^2}$$

E.g. for **P** = "the graph is complete"

$$\mu_3(\mathbf{P}) = \frac{1}{|C_3|} = \frac{1}{2^{3^2}}$$

$$\mu_\infty(\mathbf{P}) = \lim_{n \to \infty} \mu_n(\mathbf{P})$$

0-1 Law

Theorem. [Glebskii et al. '69, Fagin '76]

For every *FO sentence* ϕ, $\mu_\infty(\phi)$ is either 0 or 1.

0-1 Law

Theorem. [Glebskii et al. '69, Fagin '76]

For every *FO sentence* ϕ, $\mu_\infty(\phi)$ is either 0 or 1 .

Examples:

- ϕ = "there is a triangle" $\mu_3(\phi) = {}^1/_{|C_3|}$ $\mu_{3n}(\phi) \geq 1 - (1 - {}^1/_{|C_3|})^n \to 1$

0-1 Law

Theorem. [Glebskii et al. '69, Fagin '76]

For every *FO sentence* ϕ, $\mu_\infty(\phi)$ is either 0 or 1 .

Examples:

- ϕ = "there is a triangle" $\mu_3(\phi) = {}^1/_{|C_3|}$ $\mu_{3n}(\phi) \geq 1 - (1 - {}^1/_{|C_3|})^n \to 1$
- ϕ = "there no 5-clique" $\mu_\infty(\phi) = 0$

0-1 Law

Theorem. [Glebskii et al. '69, Fagin '76]

For every *FO sentence* ϕ, $\mu_\infty(\phi)$ is either 0 or 1 .

Examples:

- ϕ = "there is a triangle" $\quad \mu_3(\phi) = {}^1/_{|C_3|} \quad \mu_{3n}(\phi) \geq 1 - (1 - {}^1/_{|C_3|})^n \to 1$
- ϕ = "there no 5-clique" $\quad \mu_\infty(\phi) = 0$
- ϕ_H = "there is an occurrence of *H as induced sub-graph*" $\quad \mu_\infty(\phi_H) = 1$

0-1 Law

Theorem. [Glebskii et al. '69, Fagin '76]

For every *FO sentence* ϕ, $\mu_\infty(\phi)$ is either 0 or 1.

Examples:

- ϕ = "there is a triangle" $\mu_3(\phi) = {}^1/_{|C_3|}$ $\mu_{3n}(\phi) \geq 1 - (1 - {}^1/_{|C_3|})^n \to 1$
- ϕ = "there no 5-clique" $\mu_\infty(\phi) = 0$
- ϕ_H = "there is an occurrence of *H as induced sub-graph*" $\mu_\infty(\phi_H) = 1$
- ϕ = "even number of edges"
- ϕ = "even number of nodes"

Your turn!

0-1 Law

Theorem. [Glebskii et al. '69, Fagin '76]

For every *FO sentence* ϕ, $\mu_\infty(\phi)$ is either 0 or 1 .

Examples:

- ϕ = "there is a triangle" $\mu_3(\phi) = {}^1/_{|C_3|}$ $\mu_{3n}(\phi) \geq 1 - (1 - {}^1/_{|C_3|})^n \to 1$
- ϕ = "there no 5-clique" $\mu_\infty(\phi) = 0$
- ϕ_H = "there is an occurrence of *H as induced sub-graph*" $\mu_\infty(\phi_H) = 1$
- ϕ = "even number of edges"
- ϕ = "even number of nodes"

Your turn!

$\mu_\infty(\phi) = {}^1/_2$

$\mu_\infty(\phi)$ not even defined

0-1 Law

Theorem. [Glebskii et al. '69, Fagin '76]

For every *FO sentence* ϕ, $\mu_\infty(\phi)$ is either 0 or 1 .

Examples:

- ϕ = "there is a triangle" $\mu_3(\phi) = {}^1/_{|C_3|}$ $\mu_{3n}(\phi) \geq 1 - (1 - {}^1/_{|C_3|})^n \to 1$
- ϕ = "there no 5-clique" $\mu_\infty(\phi) = 0$
- ϕ_H = "there is an occurrence of *H as induced sub-graph*" $\mu_\infty(\phi_H) = 1$
- ϕ = "even number of edges" $\mu_\infty(\phi) = {}^1/_2$
- ϕ = "even number of nodes" $\mu_\infty(\phi)$ not even defined
- ϕ = "more edges than nodes" $\mu_\infty(\phi) = 1$ (yet not FO-definable!)

Your turn!

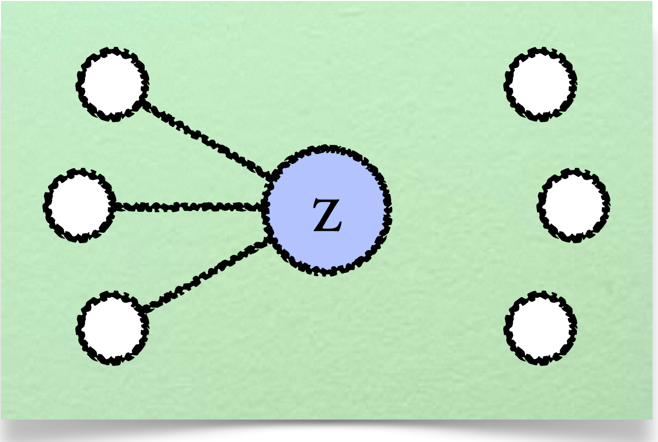
0-1 Law

For every *FO sentence* ϕ, $\mu_\infty(\phi)$ is either 0 or 1.

Let k = quantifier rank of ϕ

$$\delta_k = \forall x_1, \ldots, x_k \, \forall y_1, \ldots, y_k \; \exists z \; \bigwedge_{i,j} x_i \neq y_j \wedge E(x_i, z) \wedge \neg E(y_j, z)$$

(Extension Axiom)



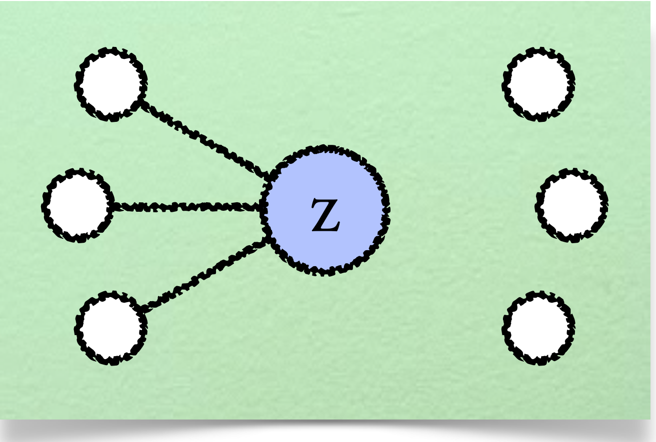
0-1 Law

For every *FO sentence* ϕ, $\mu_\infty(\phi)$ is either 0 or 1.

Let k = quantifier rank of ϕ

$$\delta_k = \forall x_1, \ldots, x_k \, \forall y_1, \ldots, y_k \; \exists z \; \bigwedge_{i,j} x_i \neq y_j \wedge E(x_i, z) \wedge \neg E(y_j, z)$$

(Extension Axiom)



Fact 1: If $G \vDash \delta_k \wedge H \vDash \delta_k$ then
Duplicator survives k rounds on G, H

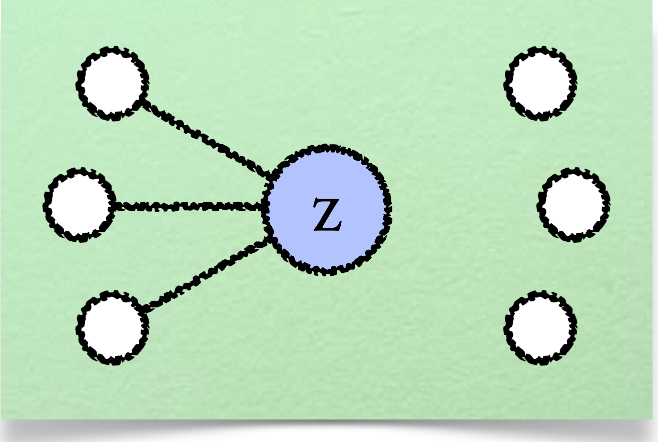
0-1 Law

For every *FO sentence* ϕ, $\mu_\infty(\phi)$ is either 0 or 1.

Let k = quantifier rank of ϕ

$$\delta_k = \forall x_1, \ldots, x_k \, \forall y_1, \ldots, y_k \; \exists z \; \bigwedge_{i,j} x_i \neq y_j \wedge E(x_i, z) \wedge \neg E(y_j, z)$$

(Extension Axiom)



Fact 1: If $G \models \delta_k \wedge H \models \delta_k$ then
Duplicator survives k rounds on G, H

Fact 2: $\mu_\infty(\delta_k) = 1$
(δ_k is almost surely true)

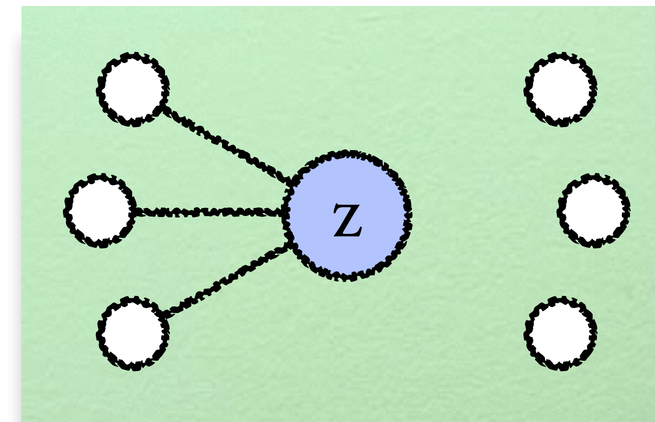
0-1 Law

For every *FO sentence* ϕ, $\mu_\infty(\phi)$ is either 0 or 1.

Let k = quantifier rank of ϕ

$$\delta_k = \forall x_1, \ldots, x_k \, \forall y_1, \ldots, y_k \; \exists z \; \bigwedge_{i,j} x_i \neq y_j \wedge E(x_i, z) \wedge \neg E(y_j, z)$$

(Extension Axiom)



Fact 1: If $G \models \delta_k \;\wedge\; H \models \delta_k$ then
Duplicator survives k rounds on G, H

Fact 2: $\mu_\infty(\delta_k) = 1$
(δ_k is almost surely true)

2 cases

a) There is G $\;G \models \delta_k \wedge \phi \;\Rightarrow$ (by Fact 1) $\forall H$: If $H \models \delta_k$ then $H \models \phi$

Thus, $\mu_\infty(\delta_k) \leq \mu_\infty(\phi)$

$\Rightarrow$ (by Fact 2) $\mu_\infty(\delta_k) = 1$, hence $\mu_\infty(\phi) = 1$

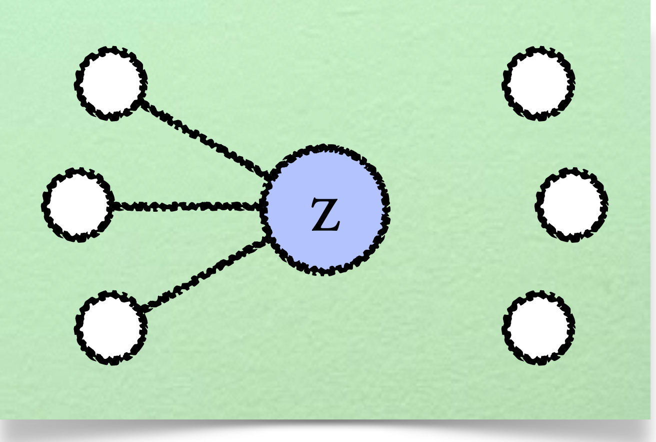
0-1 Law

For every *FO sentence* ϕ, $\mu_\infty(\phi)$ is either 0 or 1.

Let k = quantifier rank of ϕ

$$\delta_k = \forall x_1, \dots, x_k \, \forall y_1, \dots, y_k \; \exists z \; \bigwedge_{i,j} x_i \neq y_j \wedge E(x_i, z) \wedge \neg E(y_j, z)$$

(Extension Axiom)



Fact 1: If $G \vDash \delta_k \wedge H \vDash \delta_k$ then
Duplicator survives k rounds on G, H

Fact 2: $\mu_\infty(\delta_k) = 1$
(δ_k is almost surely true)

2 cases

a) There is G $G \vDash \delta_k \wedge \phi$ $\Rightarrow$ (by Fact 1) $\forall H$: If $H \vDash \delta_k$ then $H \vDash \phi$

Thus, $\mu_\infty(\delta_k) \leq \mu_\infty(\phi)$

$\Rightarrow$ (by Fact 2) $\mu_\infty(\delta_k) = 1$, hence $\mu_\infty(\phi) = 1$

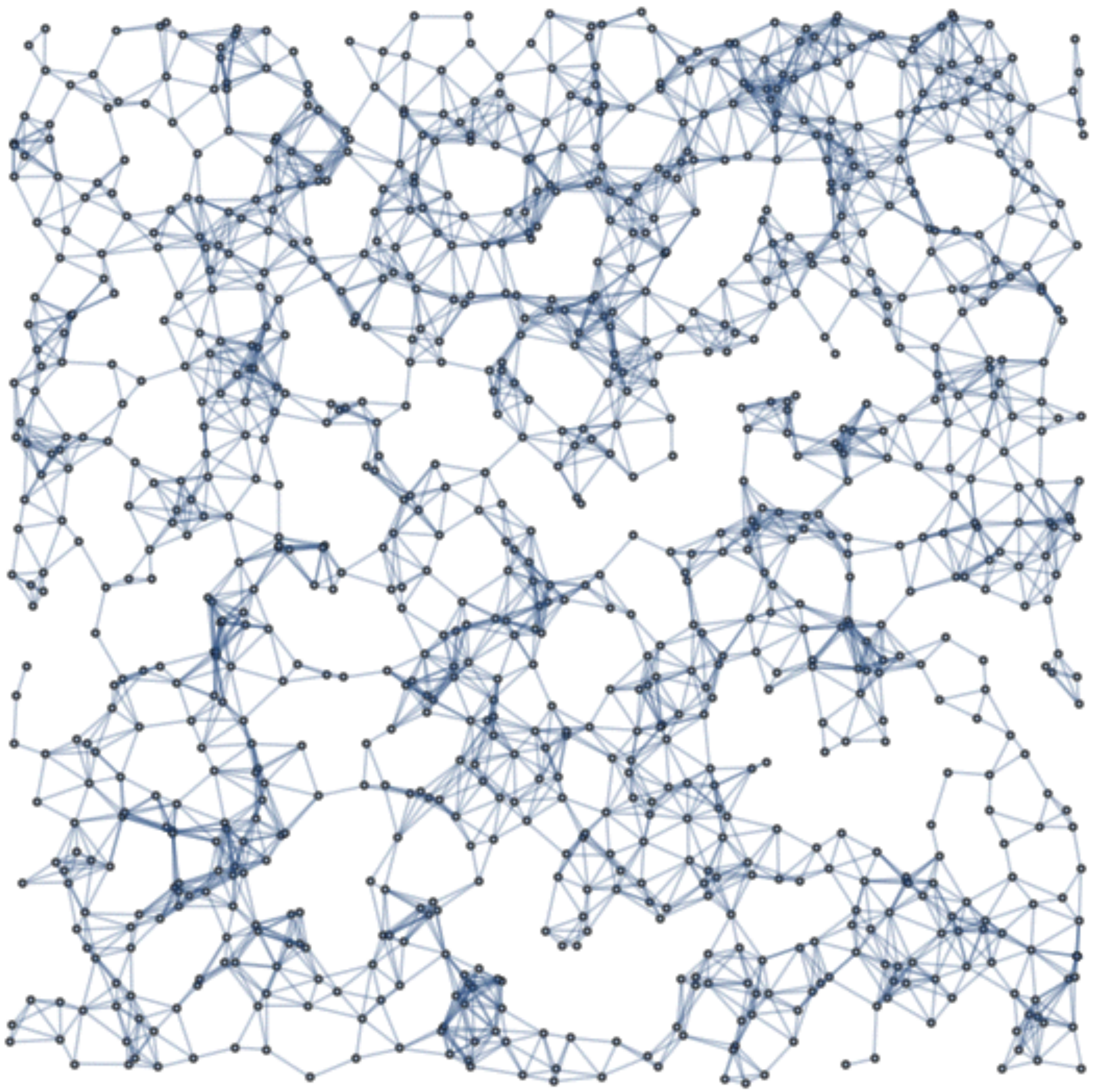
b) There is no $G \vDash \delta_k \wedge \phi$ $\Rightarrow$ (by Fact 2) there is $G \vDash \delta_k$,

$\Rightarrow$ $G \vDash \delta_k \wedge \neg\phi$ $\Rightarrow$ (by case a) $\mu_\infty(\neg\phi) = 1$

0-1 Law

For every *FO sentence* ϕ, $\mu_\infty(\phi)$ is either 0 or 1, *and this depends on whether* $\text{RADO} \vDash \phi$

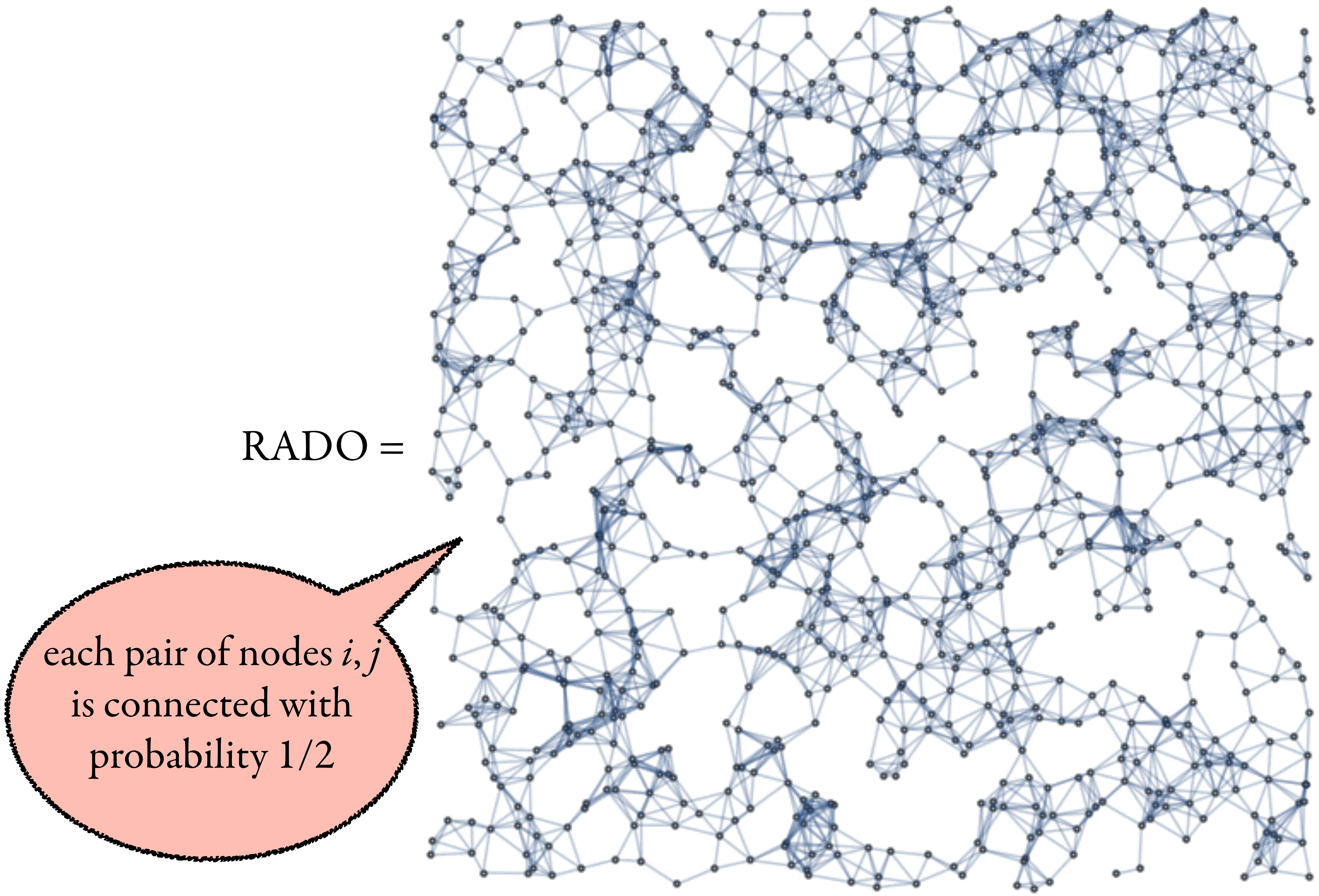
RADO =



0-1 Law

For every *FO sentence* ϕ, $\mu_\infty(\phi)$ is either 0 or 1, *and this depends on whether* RADO $\models \phi$

RADO =

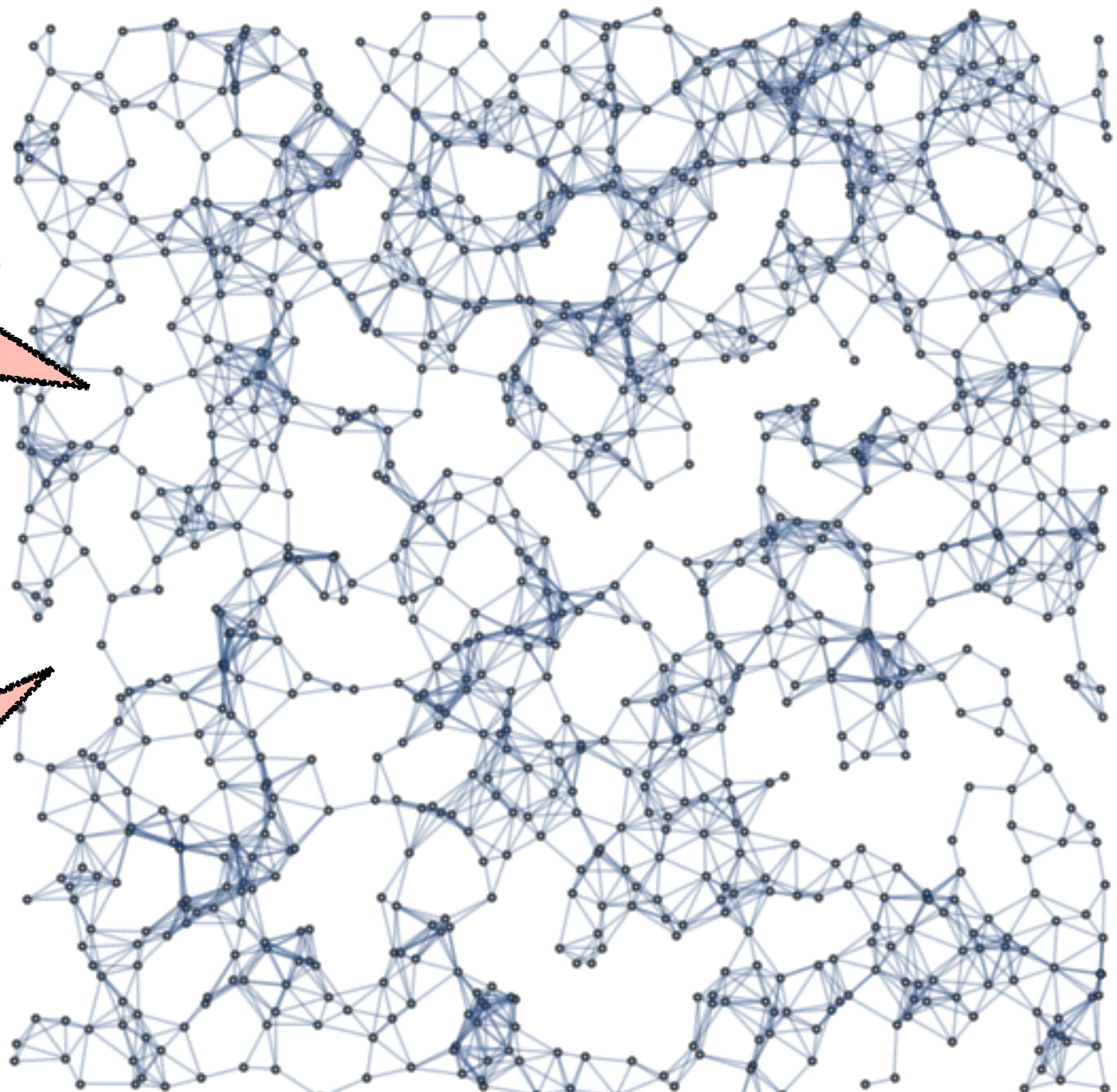


0-1 Law

For every *FO sentence* ϕ, $\mu_\infty(\phi)$ is either 0 or 1, *and this depends on whether* RADO $\models \phi$

each pair of nodes i, j is connected if i-th bit of j is 1

RADO =



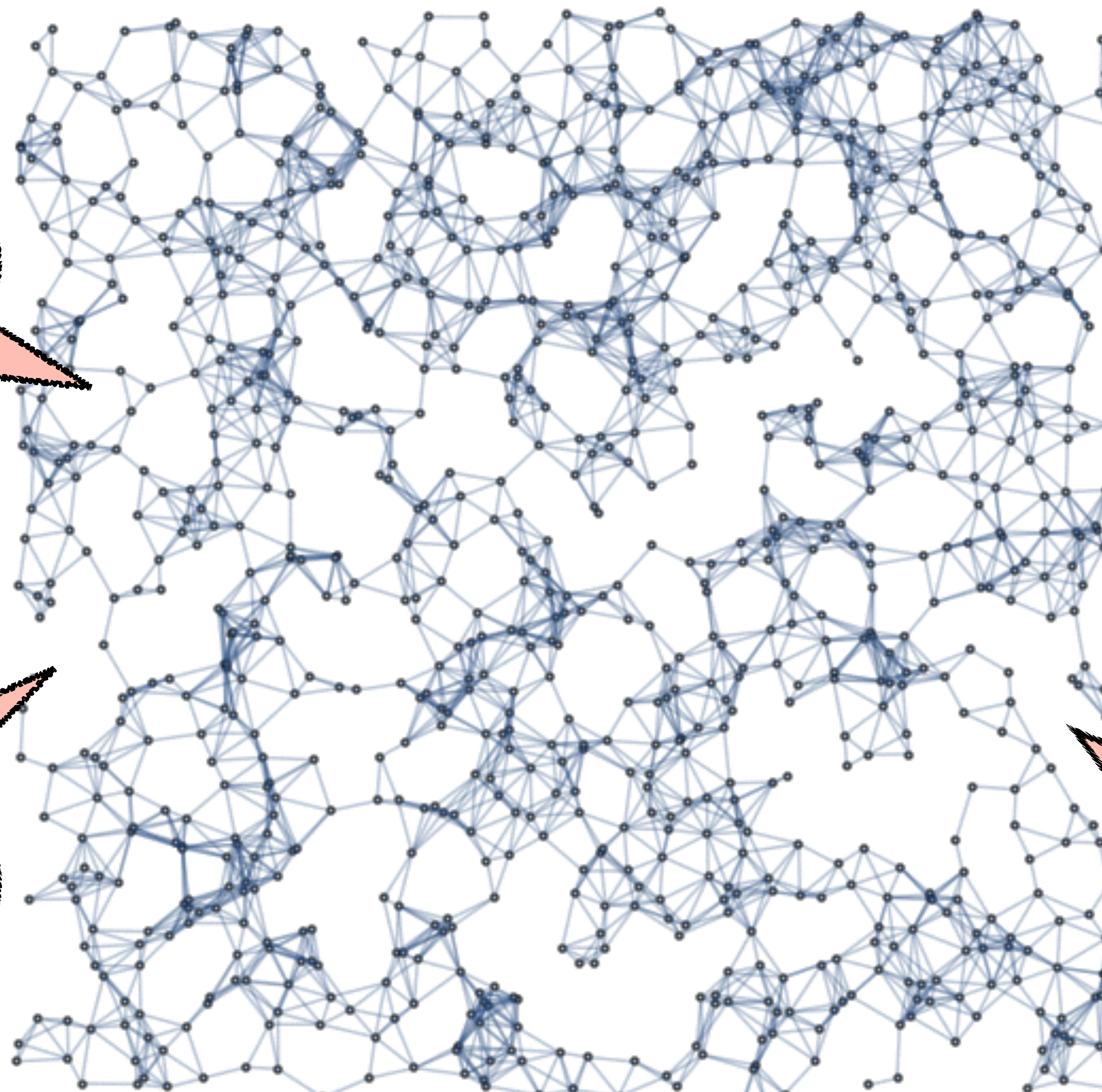
each pair of nodes i, j is connected with probability 1/2

0-1 Law

For every *FO sentence* ϕ, $\mu_\infty(\phi)$ is either 0 or 1, *and this depends on whether* RADO $\vDash \phi$

each pair of nodes i, j
is connected if
i-th bit of j is 1

RADO =



each pair of nodes i, j
is connected with
probability 1/2

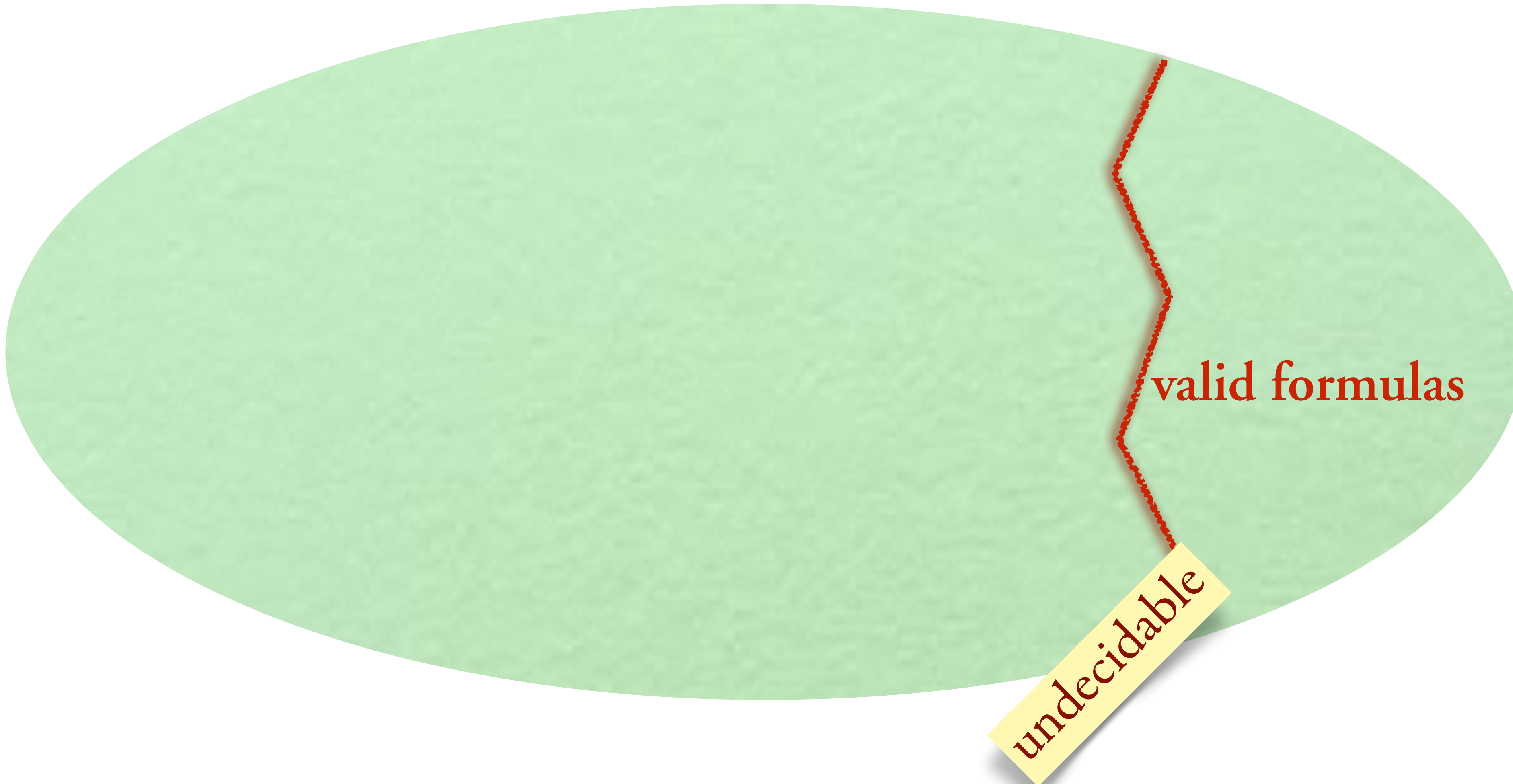
the unique
graph that
satisfies
δ_k for all k

0-1 Law

Theorem. The problem of deciding whether an FO sentence is *almost surely true* ($\mu_\infty = 1$) is PSPACE-complete. [Grandjean '83]

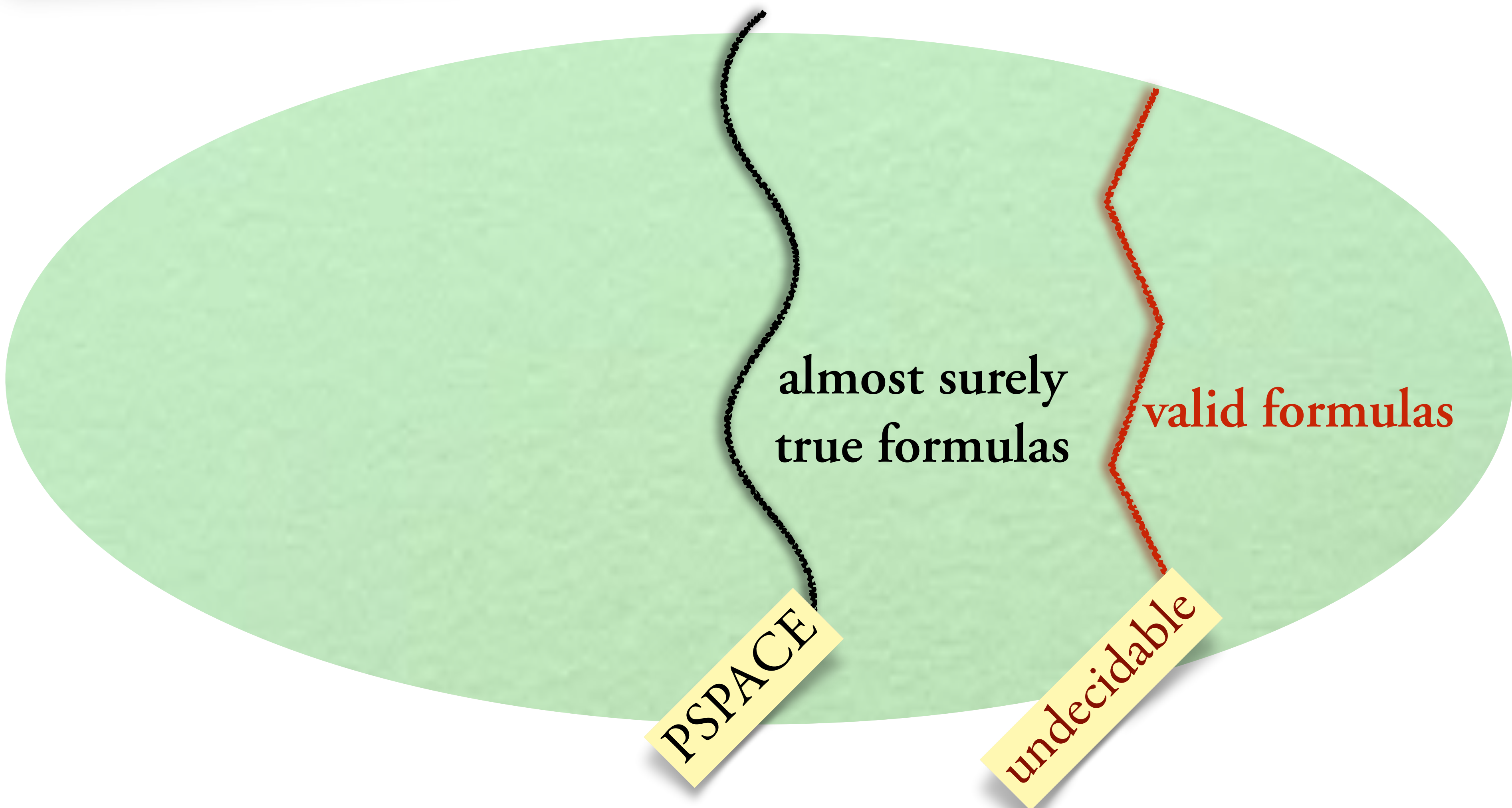
0-1 Law

Theorem. The problem of deciding whether an FO sentence is *almost surely true* ($\mu_\infty = 1$) is PSPACE-complete. [Grandjean '83]



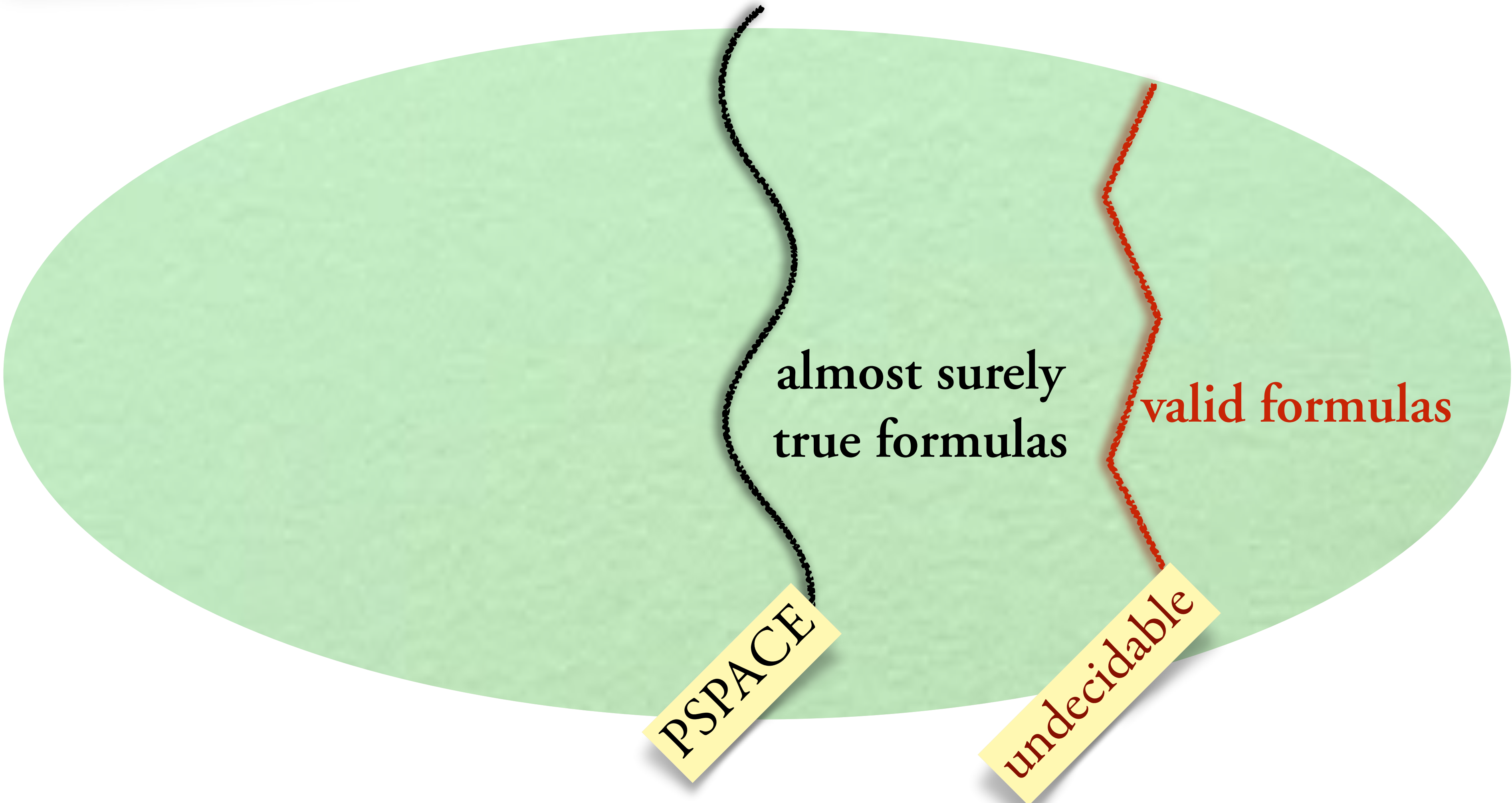
0-1 Law

Theorem. The problem of deciding whether an FO sentence is *almost surely true* $(\mu_\infty = 1)$ is PSPACE-complete. [Grandjean '83]



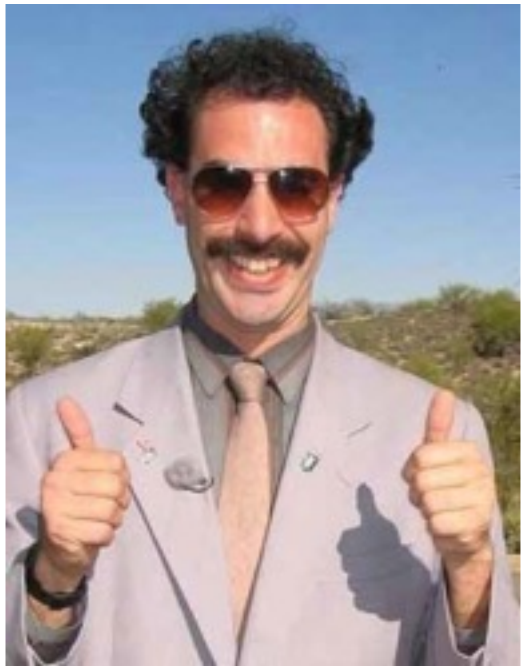
0-1 Law

Theorem. The problem of deciding whether an FO sentence is *almost surely true* ($\mu_\infty = 1$) is PSPACE-complete. [Grandjean '83]



Query evaluation on large databases:

Don't bother evaluating an FO query,
it's either true or false with high probability!



0-1 Law

Does the 0-1 Law apply to real-life databases?

Not quite: database *constraints* easily spoil Extension Axiom.

0-1 Law

Does the 0-1 Law apply to real-life databases?

Not quite: database *constraints* easily spoil Extension Axiom.

Consider:

- functional constraint $\forall x, x', y, y' \; \big(E(x,y) \wedge E(x,y') \Rightarrow y = y' \big) \; \wedge$
  $\big(E(x,y) \wedge E(x',y) \Rightarrow x = x' \big)$ (E is a permutation)
- FO query $\phi = \neg\exists x \; E(x,x)$

0-1 Law

Does the 0-1 Law apply to real-life databases?

Not quite: database *constraints* easily spoil Extension Axiom.

Consider:

- functional constraint $\forall x, x', y, y' \; \big(E(x,y) \wedge E(x,y') \Rightarrow y = y' \big) \wedge$
  $\big(E(x,y) \wedge E(x',y) \Rightarrow x = x' \big)$ (E is a permutation)
- FO query $\phi = \neg\exists x \; E(x,x)$

Probability that a permutation E satisfies $\phi = {}^{!n}/_{n!} \rightarrow e^{-1} = 0.3679\ldots$

0-1 Law

Does the 0-1 Law apply to real-life databases?

Not quite: database *constraints* easily spoil Extension Axiom.

Consider:

- functional constraint $\forall x, x', y, y' \ (E(x,y) \wedge E(x,y') \Rightarrow y = y') \wedge$
  $(E(x,y) \wedge E(x',y) \Rightarrow x = x')$ (E is a permutation)
- FO query $\phi = \neg\exists x \ E(x,x)$

Probability that a permutation E satisfies $\phi = {}^{!n}/_{n!} \rightarrow e^{-1} = 0.3679\ldots$

The 0-1 Law is a tool for proving expressiveness results,
not a statement on the real-life probability of queries being non-empty.

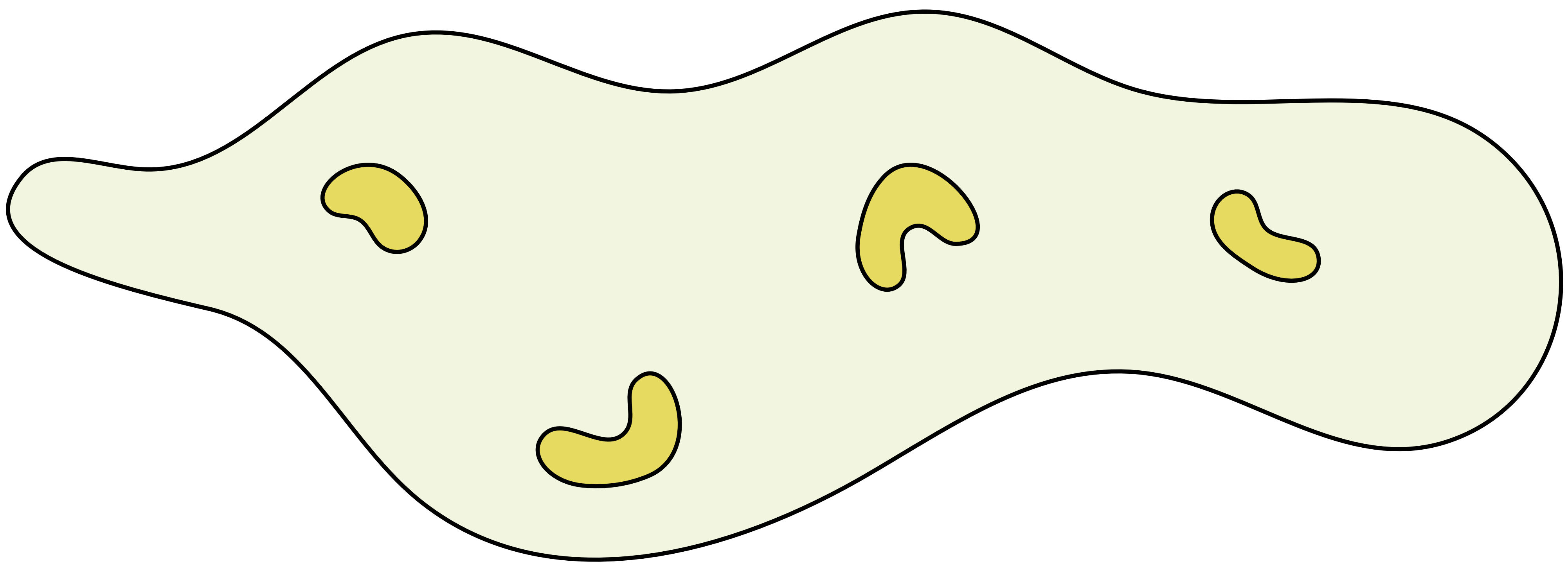
Locality

Idea: First order logic can only express "local" properties

Locality

Idea: First order logic can only express "local" properties

Local = properties of nodes which are close to one another



Hanf locality

Definition. The **Gaifman graph** of a structure $S = (V, R_1, \ldots, R_m)$ is the **undirected** graph

$$G(S) = (V, E) \quad \text{where} \quad E = \{ (u, v) \mid \exists (\ldots, u, \ldots, v, \ldots) \in R_i \text{ for some } i \}$$

Hanf locality

Definition. The **Gaifman graph** of a structure $S = (V, R_1, \ldots, R_m)$ is the **undirected** graph

$$G(S) = (V, E) \quad \text{where} \quad E = \{ (u, v) \mid \exists (\ldots, u, \ldots, v, \ldots) \in R_i \text{ for some } i \}$$

Agent	Name	Drives
007	James Bond	Aston Martin
200	Mr Smith	Cadillac
201	Mrs Smith	Mercedes
3	Jason Bourne	BMW

Car	Country
Aston Martin	UK
Cadillac	USA
Mercedes	Germany
BMW	Germany

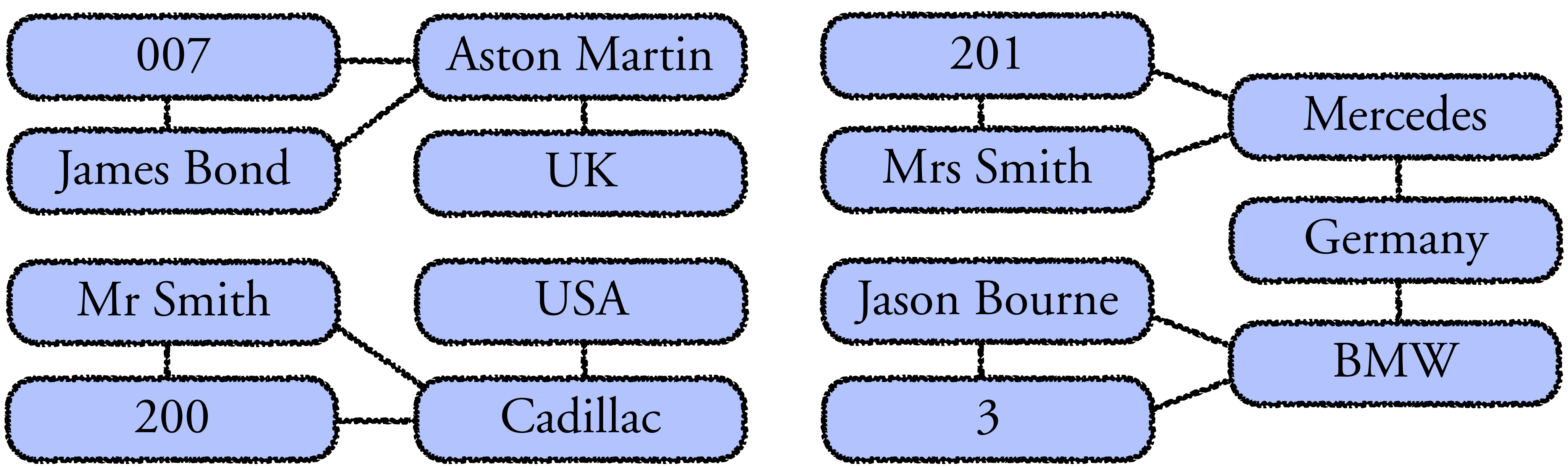
Hanf locality

Definition. The **Gaifman graph** of a structure $S = (V, R_1, \ldots, R_m)$ is the **undirected** graph

$$G(S) = (V, E) \quad \text{where} \quad E = \{ (u,v) \mid \exists\, (\ldots, u, \ldots, v, \ldots) \in R_i \text{ for some } i \}$$

Agent	Name	Drives
007	James Bond	Aston Martin
200	Mr Smith	Cadillac
201	Mrs Smith	Mercedes
3	Jason Bourne	BMW

Car	Country
Aston Martin	UK
Cadillac	USA
Mercedes	Germany
BMW	Germany



Hanf locality

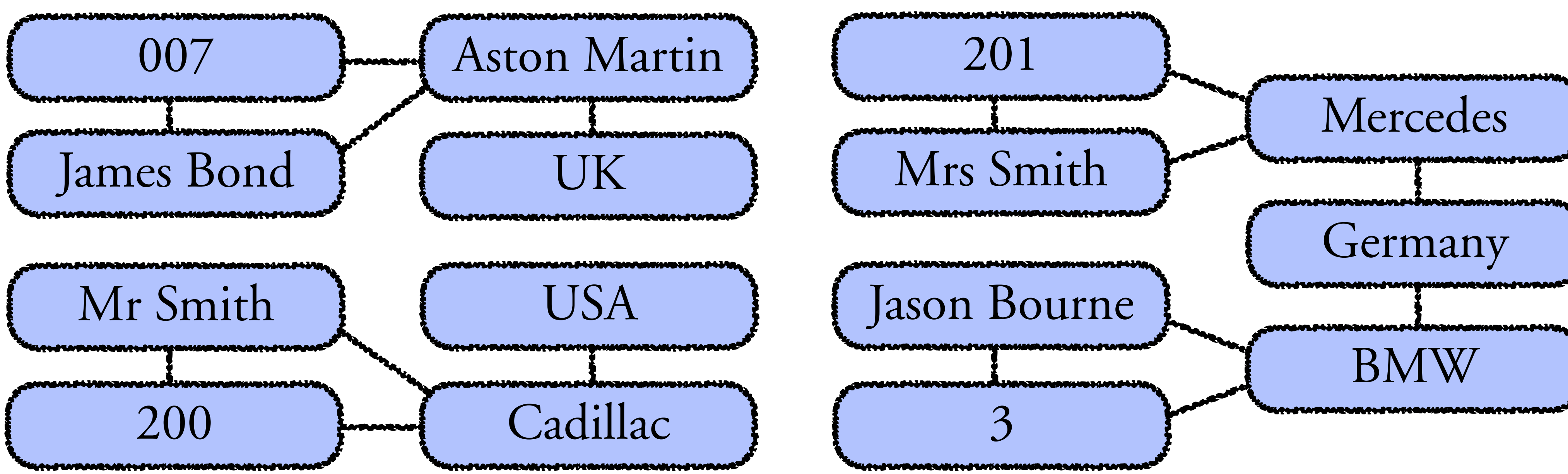
Definition. The **Gaifman graph** of a structure $S = (V, R_1, \dots, R_m)$ is the **undirected** graph

$$G(S) = (V, E) \quad \text{where} \quad E = \{ (u,v) \mid \exists\, (\dots, u, \dots, v, \dots) \in R_i \text{ for some } i \}$$

Agent	Name	Drives
007	James Bond	Aston
200	Mr Smith	Cadi
201	Mrs Smith	Mercedes
3	Jason Bourne	BMW

Car	Country
	UK
	USA
Mercedes	Germany
BMW	Germany

The Gaifman graph of a graph G is the underlying undirected graph.

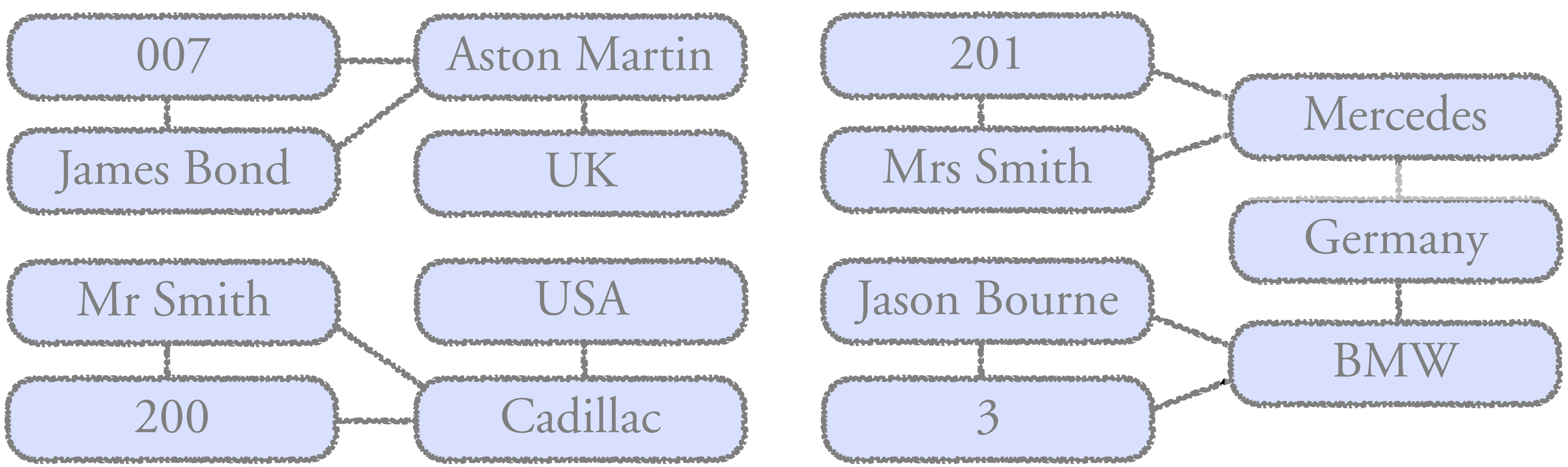


Hanf locality

- dist (u, v) = *distance* between u and v in the Gaifman graph
- $S[u, r]$ = *ball* around u of radius r = sub-structure induced by $\{ v \mid \text{dist}(u,v) \leq r \}$

Agent	Name	Drives
007	James Bond	Aston Martin
200	Mr Smith	Cadillac
201	Mrs Smith	Mercedes
3	Jason Bourne	BMW

Car	Country
Aston Martin	UK
Cadillac	USA
Mercedes	Germany
BMW	Germany

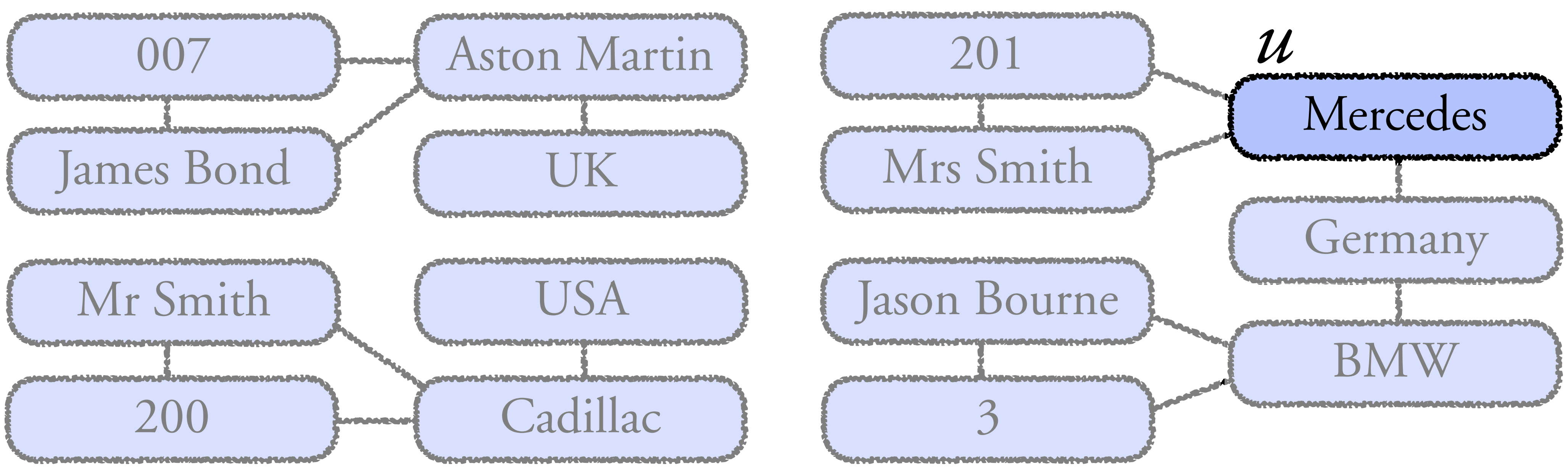


Hanf locality

- $\text{dist}(u, v) = $ *distance* between u and v in the Gaifman graph
- $S[u, r] = $ *ball* around u of radius r = sub-structure induced by $\{ v \mid \text{dist}(u, v) \leq r \}$

Agent	Name	Drives
007	James Bond	Aston Martin
200	Mr Smith	Cadillac
201	Mrs Smith	Mercedes u
3	Jason Bourne	BMW

Car	Country
Aston Martin	UK
Cadillac	USA
u Mercedes	Germany
BMW	Germany

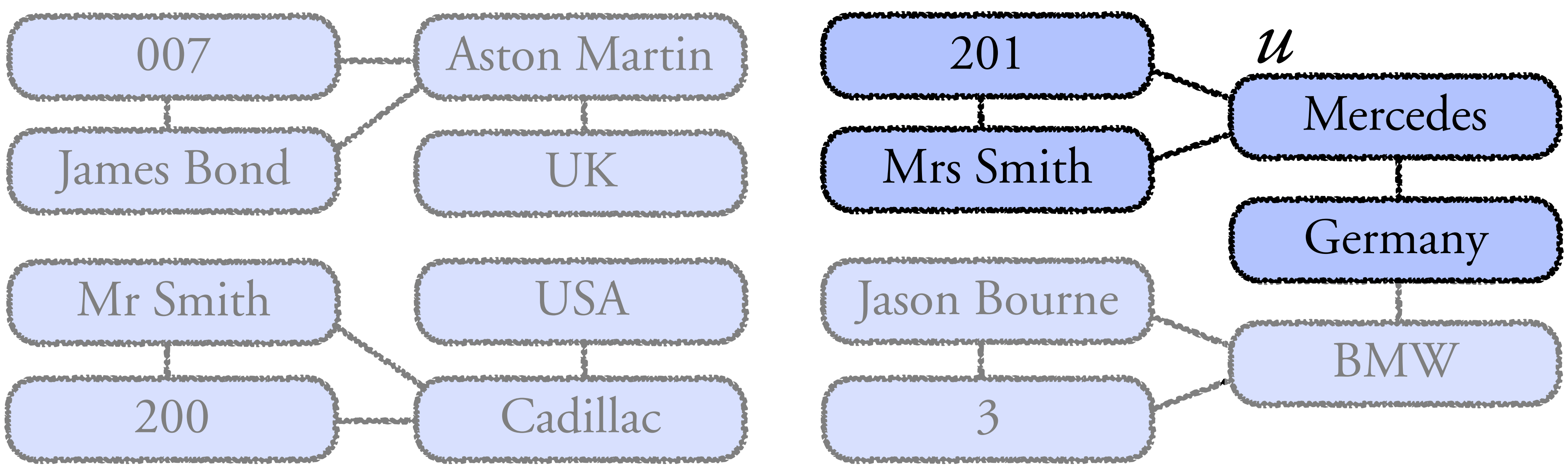


Hanf locality

- $\text{dist}(u, v)$ = *distance* between u and v in the Gaifman graph
- $S[u, r]$ = *ball* around u of radius r = sub-structure induced by $\{ v \mid \text{dist}(u, v) \leq r \}$

Agent	Name	Drives
007	James Bond	Aston Martin
200	Mr Smith	Cadillac
201	Mrs Smith	Mercedes u
3	Jason Bourne	BMW

Car	Country
Aston Martin	UK
Cadillac	USA
u Mercedes	Germany
BMW	Germany

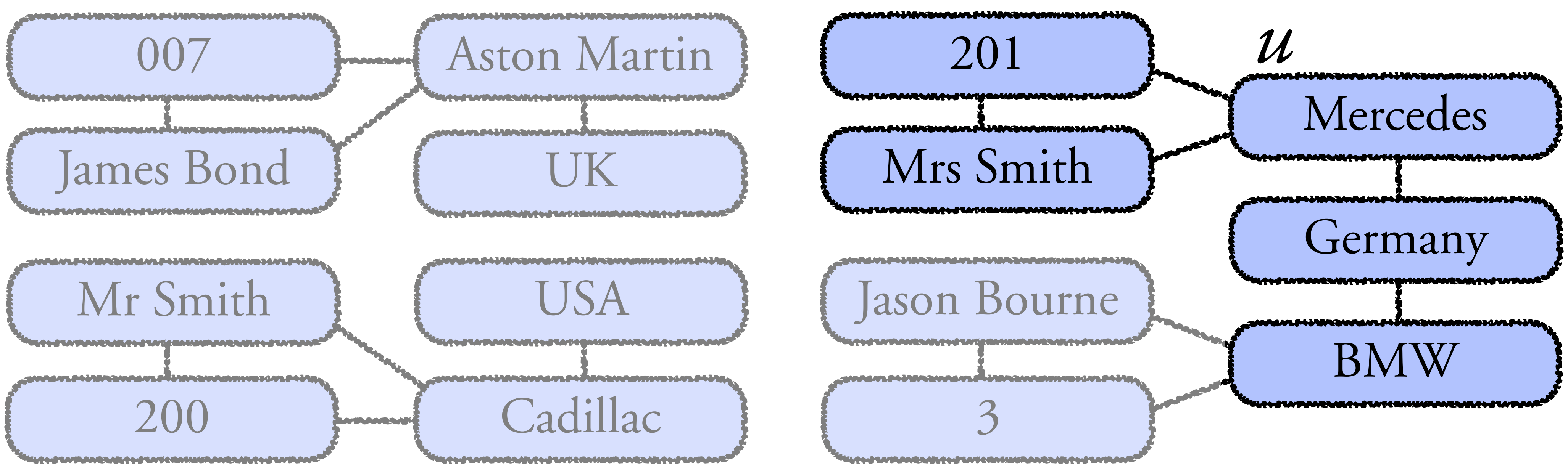


Hanf locality

- $\text{dist}(u,v)$ = *distance* between u and v in the Gaifman graph
- $S[u,r]$ = *ball* around u of radius r = sub-structure induced by $\{ v \mid \text{dist}(u,v) \leq r \}$

Agent	Name	Drives
007	James Bond	Aston Martin
200	Mr Smith	Cadillac
201	Mrs Smith	Mercedes u
3	Jason Bourne	BMW

Car	Country
Aston Martin	UK
Cadillac	USA
u Mercedes	Germany
BMW	Germany

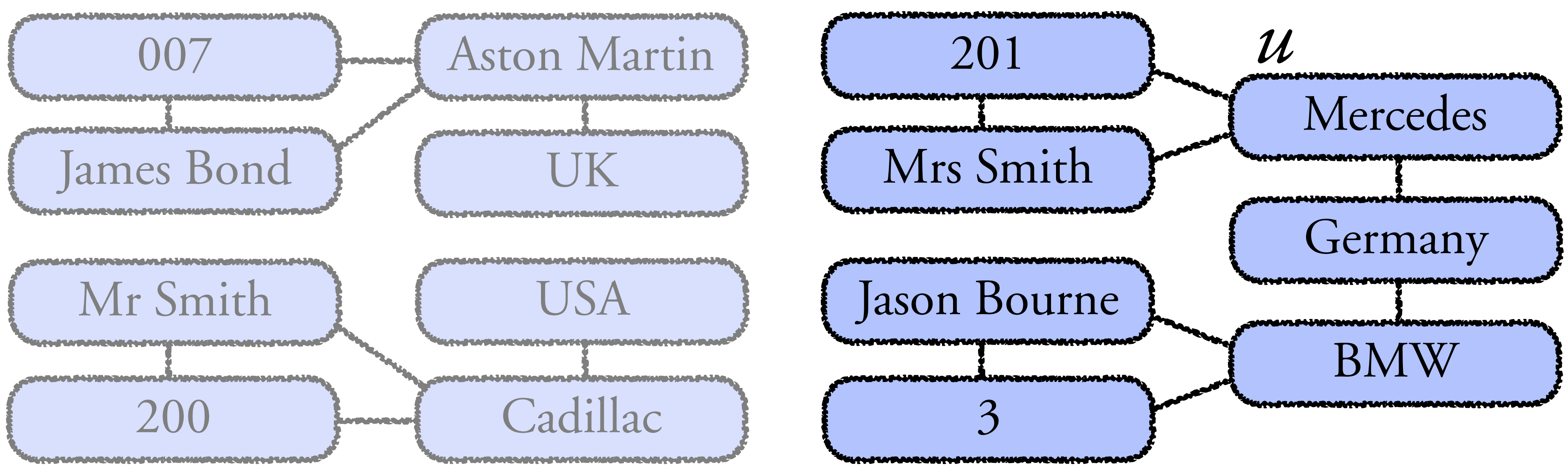


Hanf locality

- dist (u,v) = *distance* between u and v in the Gaifman graph
- $S[u,r]$ = *ball* around u of radius r = sub-structure induced by $\{ v \mid \text{dist}(u,v) \leq r \}$

Agent	Name	Drives
007	James Bond	Aston Martin
200	Mr Smith	Cadillac
201	Mrs Smith	Mercedes u
3	Jason Bourne	BMW

Car	Country
Aston Martin	UK
Cadillac	USA
u Mercedes	Germany
BMW	Germany



Hanf locality

Definition. Two structures S_1 and S_2 are **Hanf(r,t)-equivalent**

iff for each structure B, the two numbers

$$\#u \text{ s.t. } S_1[u,r] \cong B \qquad \#v \text{ s.t. } S_2[v,r] \cong B$$

are *either the same* or *both* $\geq t$.

Hanf locality

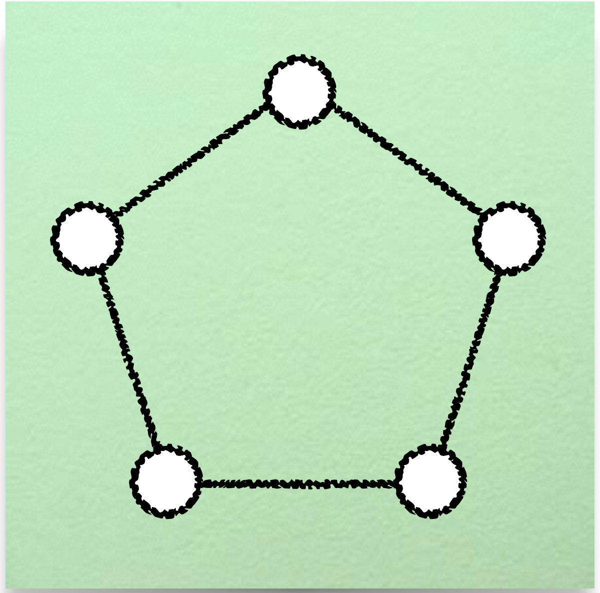
Definition. Two structures S_1 and S_2 are **Hanf(r, t)-equivalent**

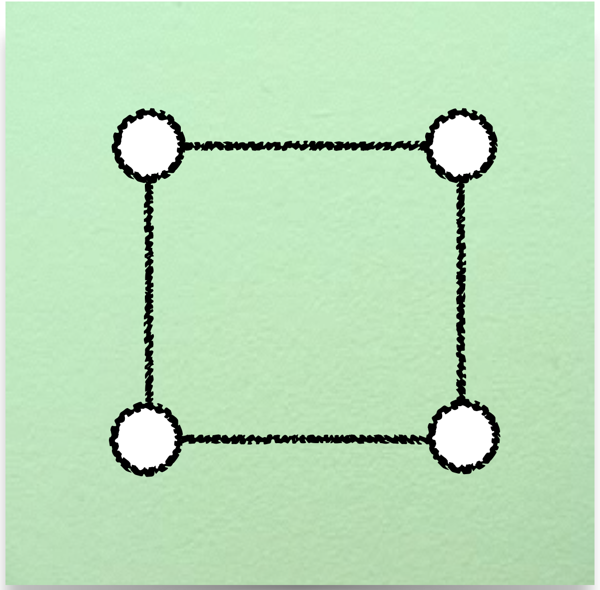
iff for each structure B, the two numbers

$$\#u \text{ s.t. } S_1[u,r] \cong B \qquad \#v \text{ s.t. } S_2[v,r] \cong B$$

are *either the same* or *both* $\geq t$.

Example. S_1, S_2 are Hanf$(1, 1)$-equivalent iff they have the *same balls* of radius 1





Hanf locality

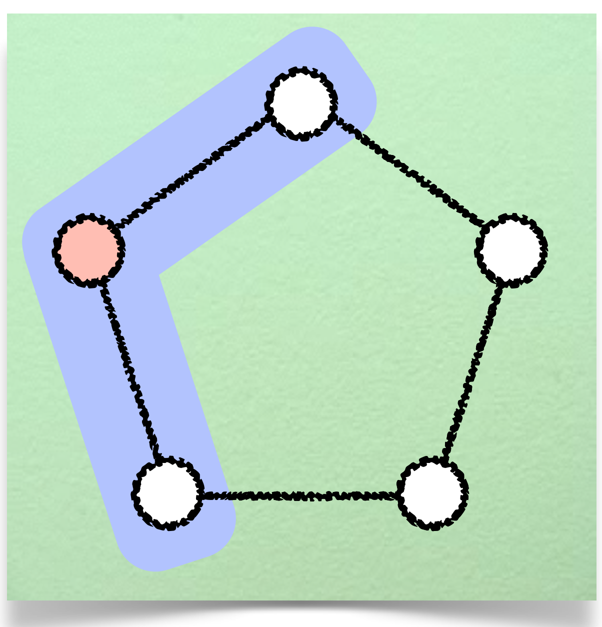
Definition. Two structures S_1 and S_2 are **Hanf(r, t) - equivalent**

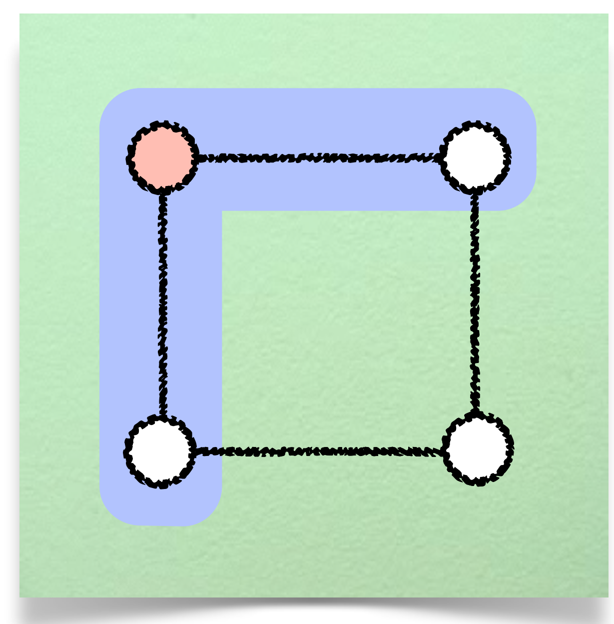
iff for each structure B, the two numbers

$$\#u \text{ s.t. } S_1[u, r] \cong B \qquad \#v \text{ s.t. } S_2[v, r] \cong B$$

are *either the same* or *both* $\geq t$.

Example. S_1, S_2 are Hanf$(1, 1)$ - equivalent iff they have the *same balls* of radius 1





Hanf locality

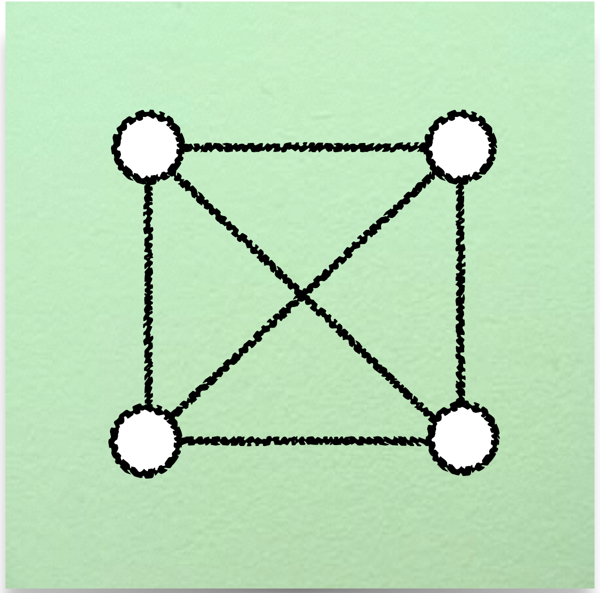
Definition. Two structures S_1 and S_2 are **Hanf(r, t) - equivalent**

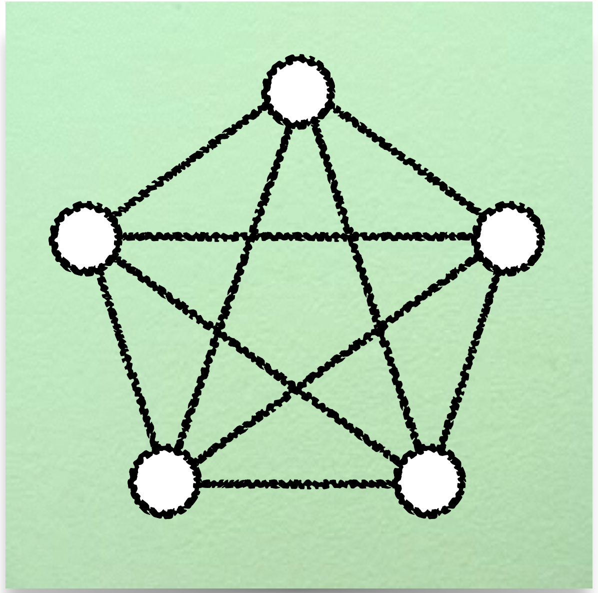
iff for each structure B, the two numbers

$$\#u \text{ s.t. } S_1[u, r] \cong B \qquad \#v \text{ s.t. } S_2[v, r] \cong B$$

are *either the same* or *both* $\geq t$.

Example. K_n, K_{n+1} are **not** Hanf(1, 1) - equivalent





Hanf locality

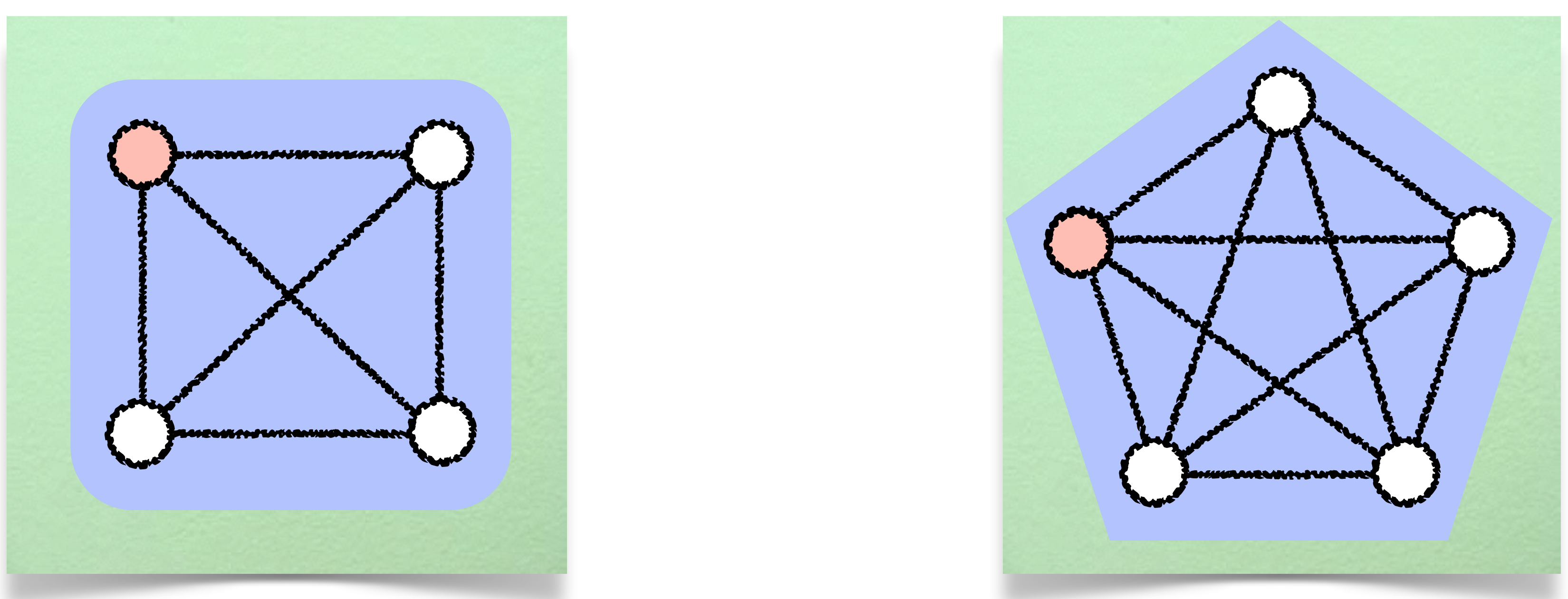
Definition. Two structures S_1 and S_2 are **Hanf(r, t) - equivalent**

iff for each structure B, the two numbers

$$\#u \text{ s.t. } S_1[u,r] \cong B \qquad \#v \text{ s.t. } S_2[v,r] \cong B$$

are *either the same* or *both* $\geq t$.

Example. K_n, K_{n+1} are **not** Hanf(1, 1) - equivalent



Hanf locality

Theorem. S_1, S_2 are n-equivalent (they satisfy the same sentences with quantifier rank n) whenever S_1, S_2 are Hanf(r, t)-equivalent, with $r = 3^n$ and $t = n$.

[Hanf '60]

Hanf locality

Theorem. S_1, S_2 are n-equivalent (they satisfy the same sentences with quantifier rank n)
whenever S_1, S_2 are Hanf(r,t)-equivalent, with $r = 3^n$ and $t = n$.

[Hanf '60]

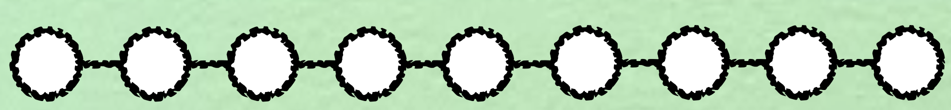
Exercise: prove that *acyclicity* is not FO-definable (even on finite structures)

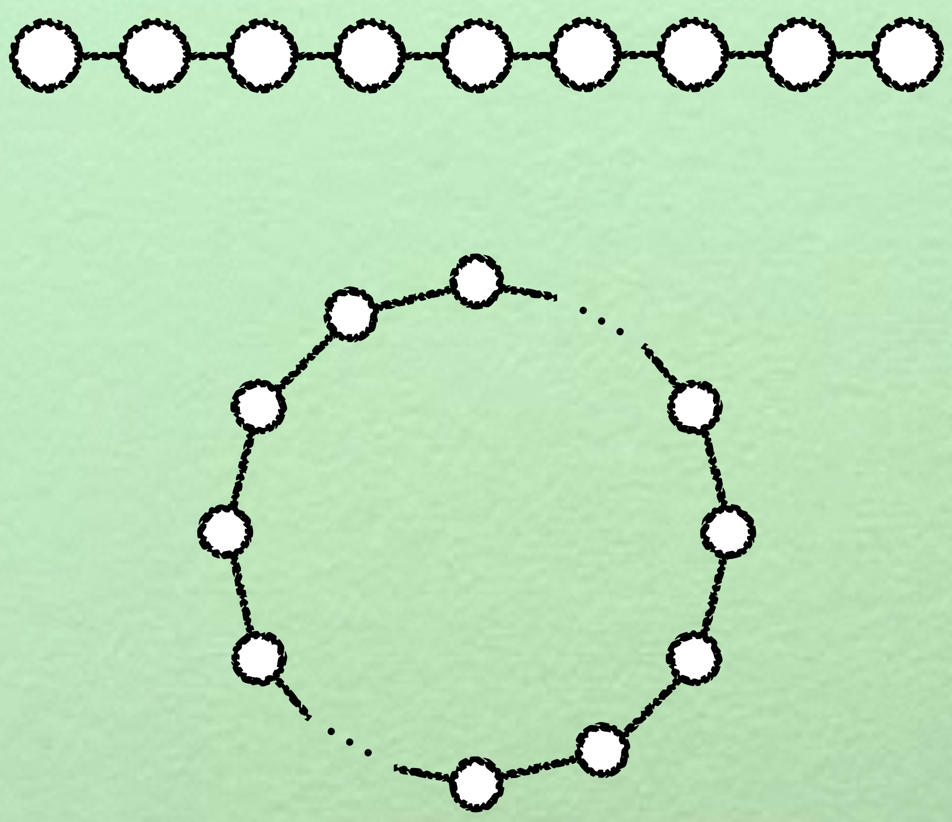
Hanf locality

Theorem. S_1, S_2 are n-equivalent (they satisfy the same sentences with quantifier rank n) whenever S_1, S_2 are Hanf(r,t)-equivalent, with $r = 3^n$ and $t = n$.

[Hanf '60]

Exercise: prove that *acyclicity* is not FO-definable (even on finite structures)





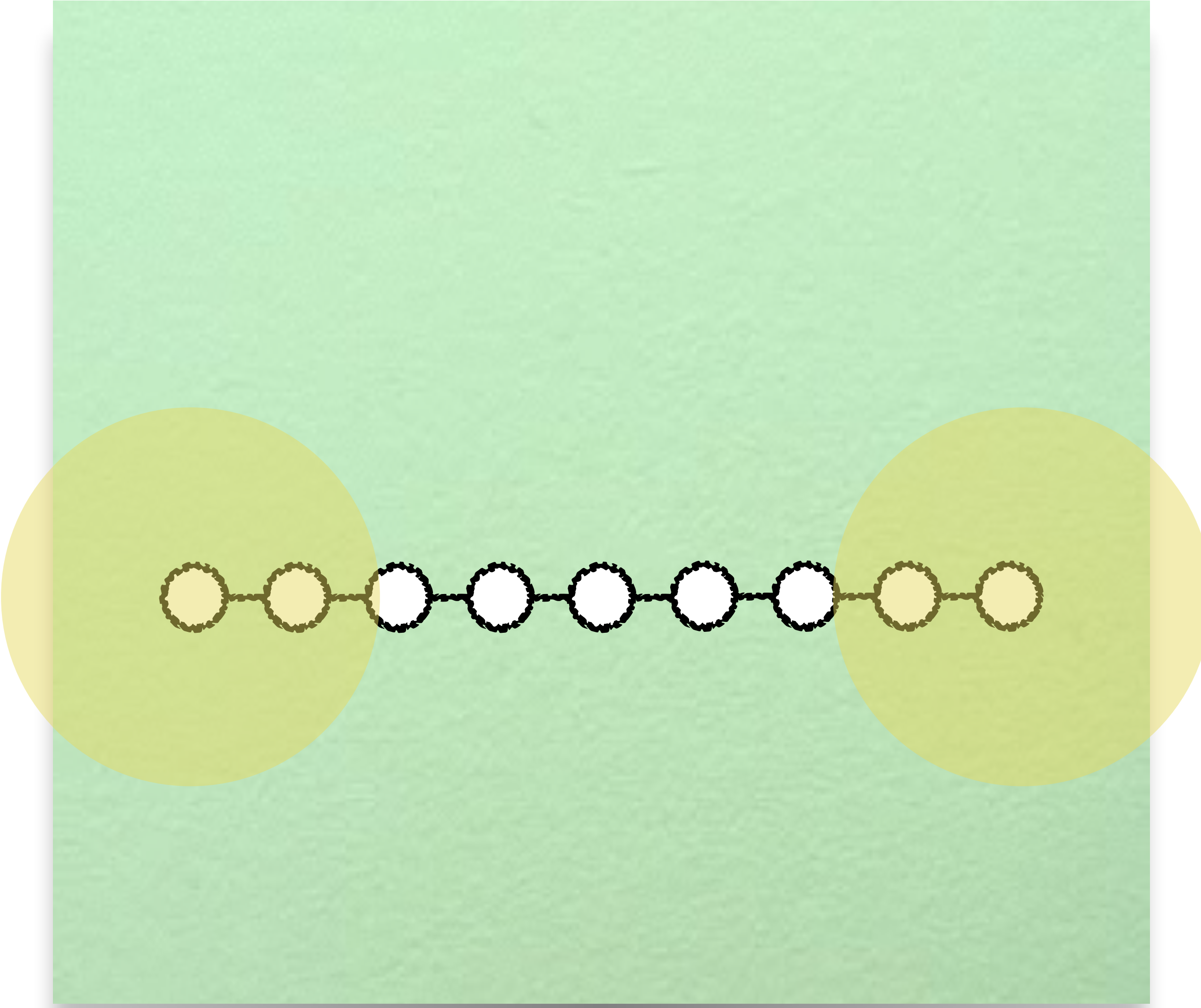
Hanf locality

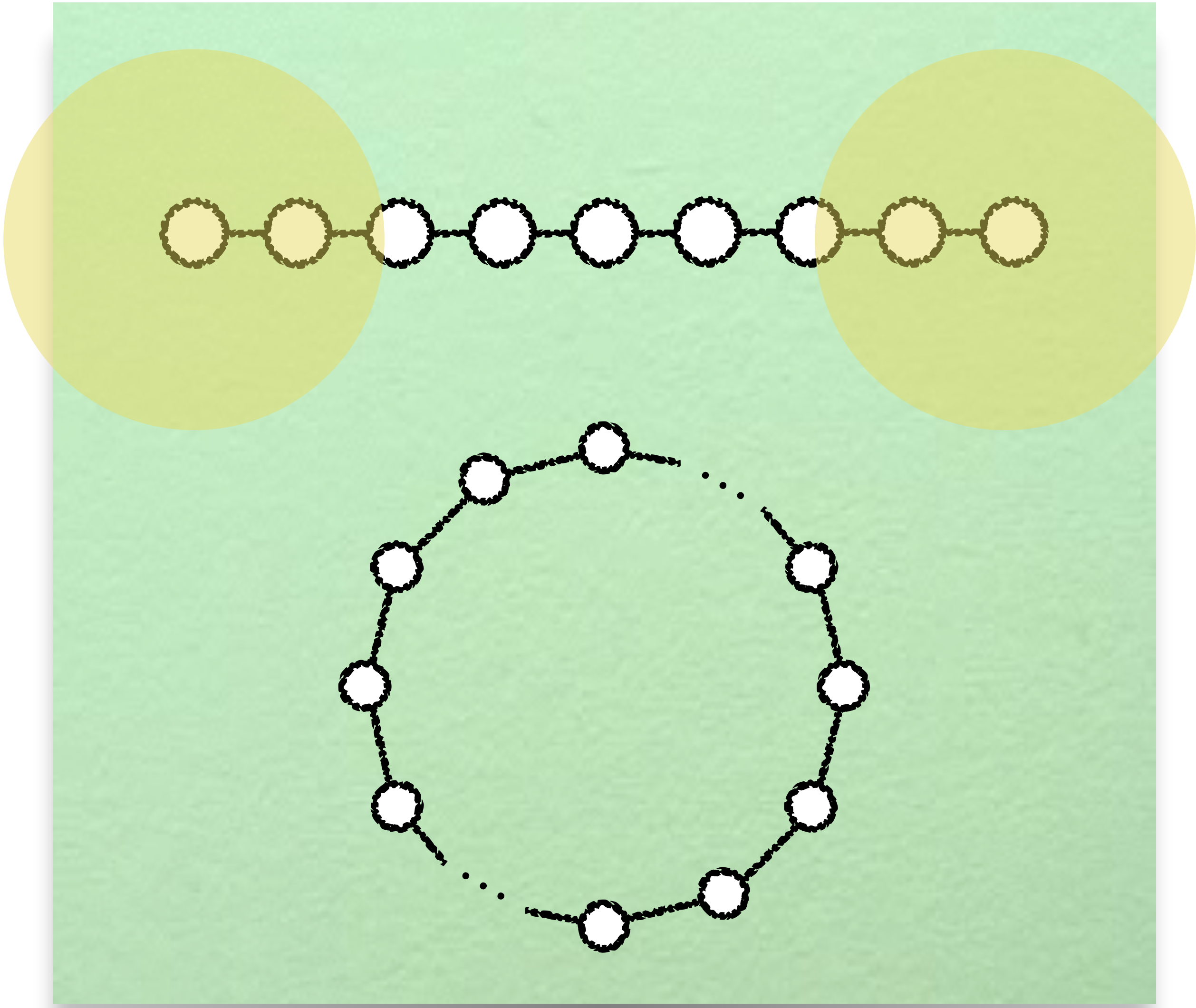
Theorem. S_1, S_2 are n-equivalent (they satisfy the same sentences with quantifier rank n)

whenever S_1, S_2 are Hanf(r,t)-equivalent, with $r = 3^n$ and $t = n$.

[Hanf '60]

Exercise: prove that *acyclicity* is not FO-definable (even on finite structures)

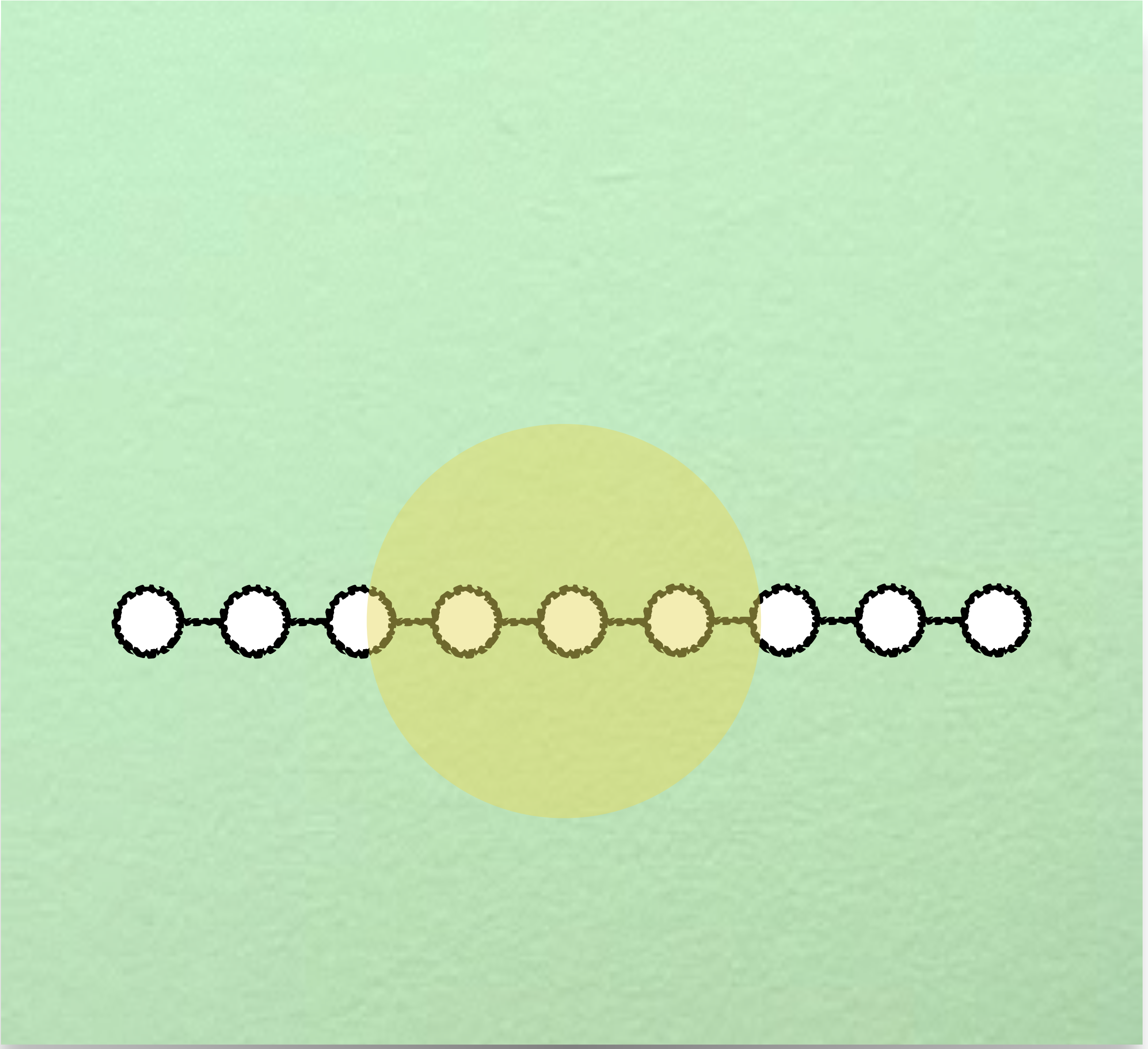


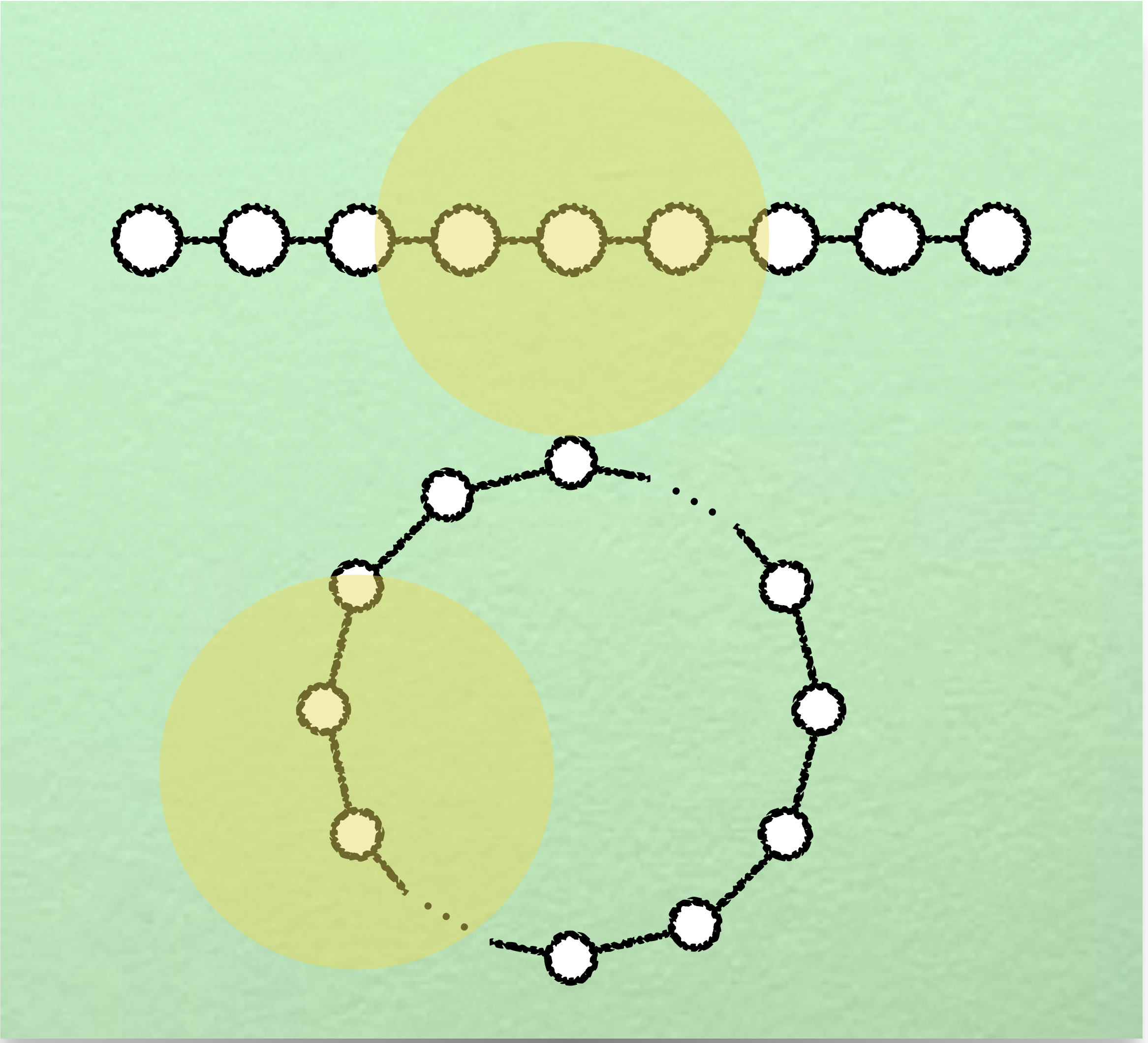


Hanf locality

Theorem. S_1, S_2 are n-equivalent (they satisfy the same sentences with quantifier rank n)
whenever S_1, S_2 are Hanf(r,t)-equivalent, with $r = 3^n$ and $t = n$.
[Hanf '60]

Exercise: prove that *acyclicity* is not FO-definable (even on finite structures)



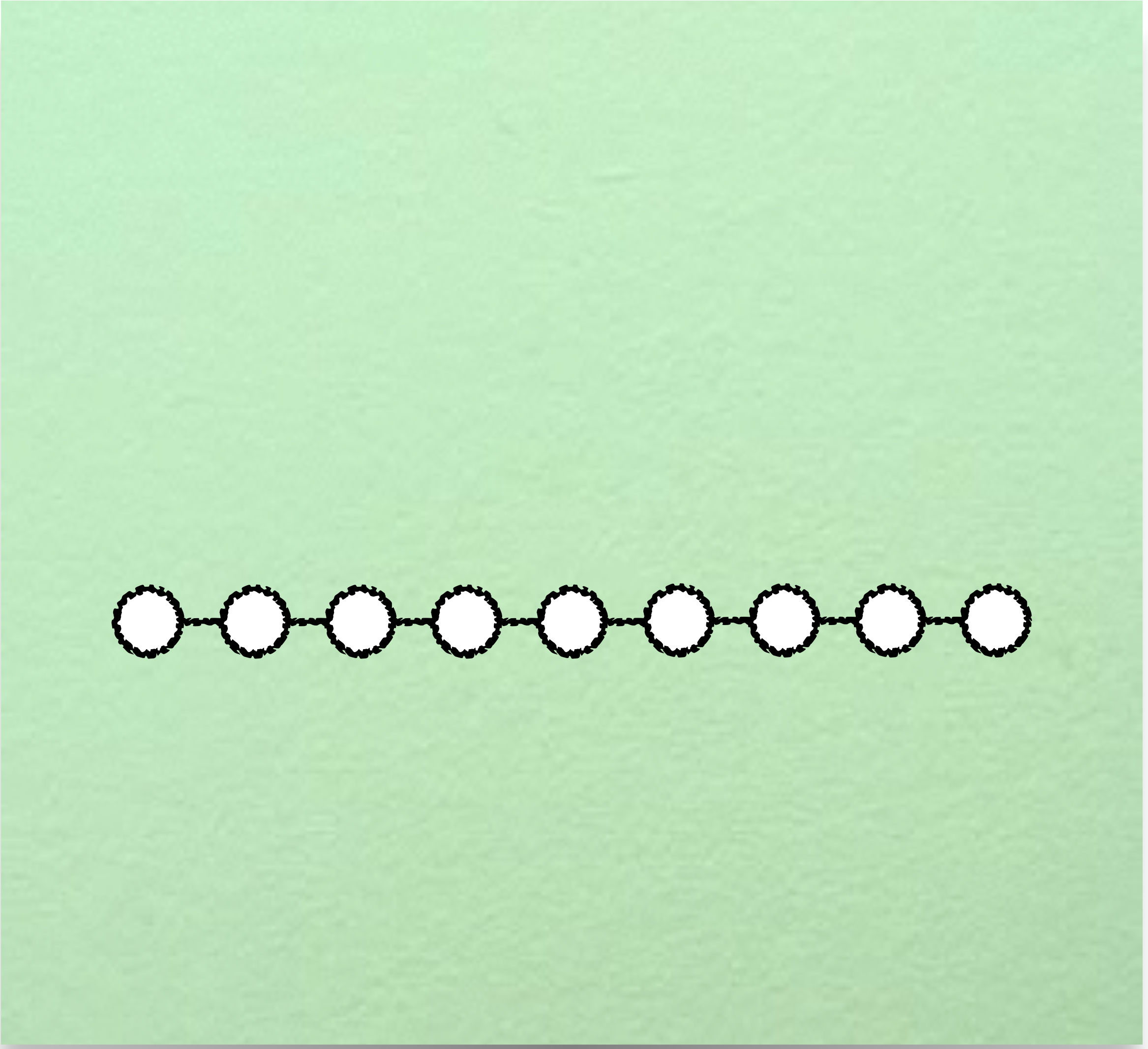


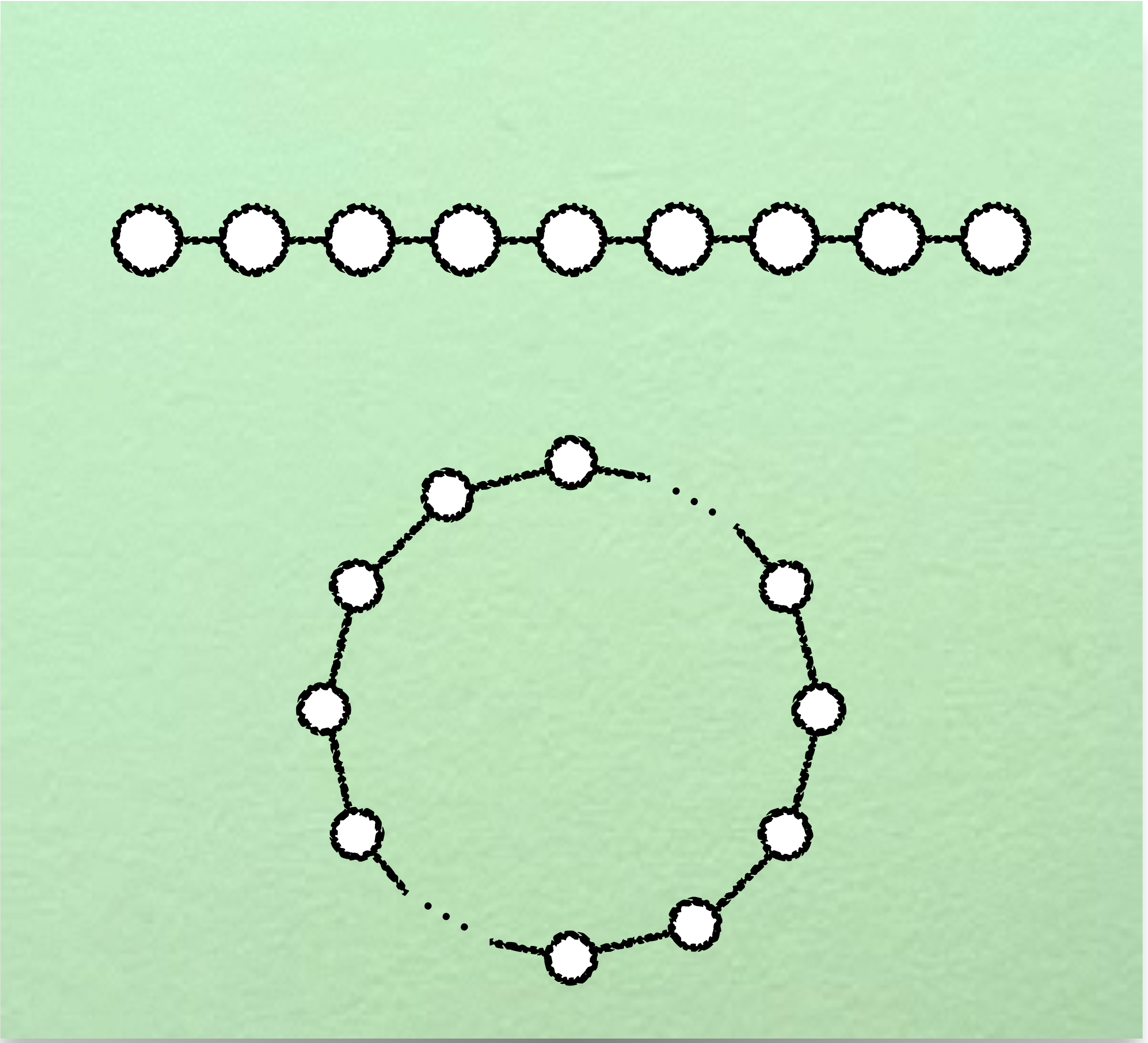
Hanf locality

Theorem. S_1, S_2 are n-equivalent (they satisfy the same sentences with quantifier rank n) whenever S_1, S_2 are Hanf(r,t)-equivalent, with $r = 3^n$ and $t = n$.

[Hanf '60]

Exercise: prove that *acyclicity* is not FO-definable (even on finite structures)





Hanf locality

Theorem. S_1, S_2 are n-equivalent (they satisfy the same sentences with quantifier rank n) whenever S_1, S_2 are Hanf(r, t)-equivalent, with $r = 3^n$ and $t = n$.

[Hanf '60]

Exercise: prove that testing whether a binary tree is *complete* is not FO-definable

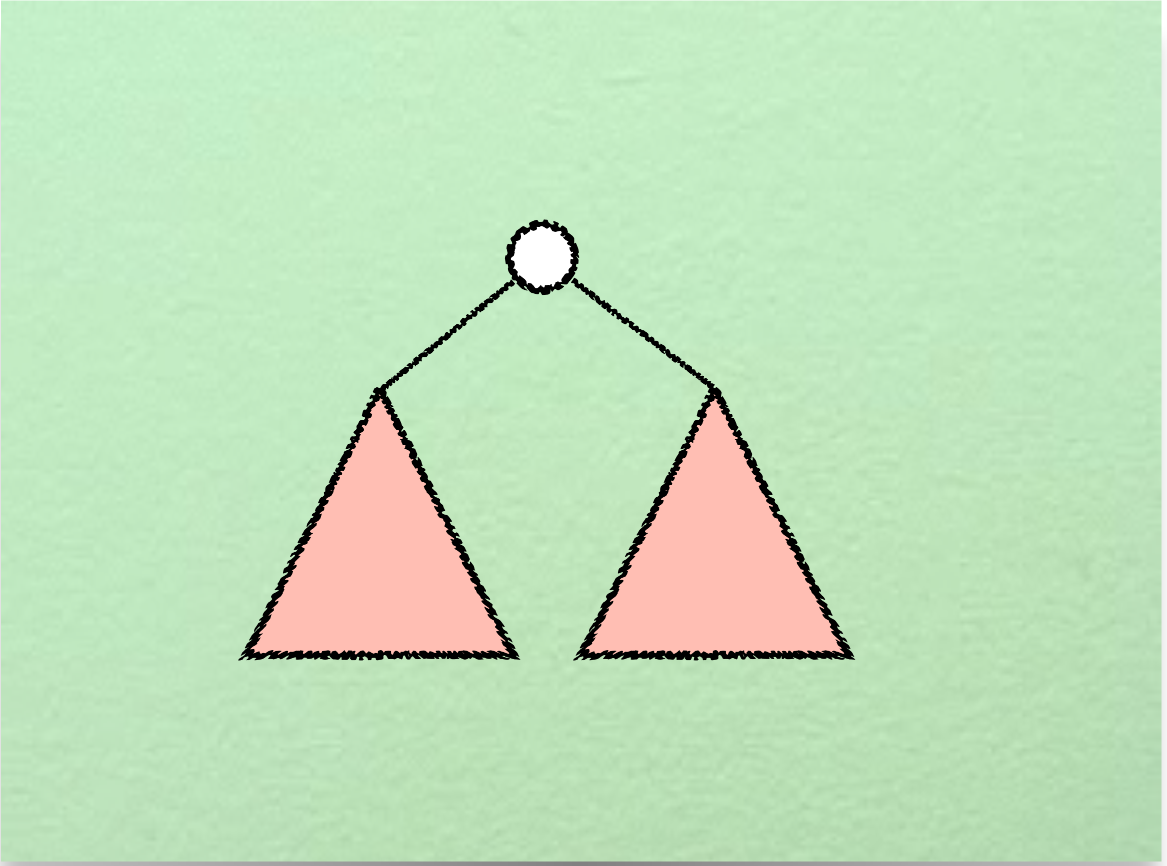
Hanf locality

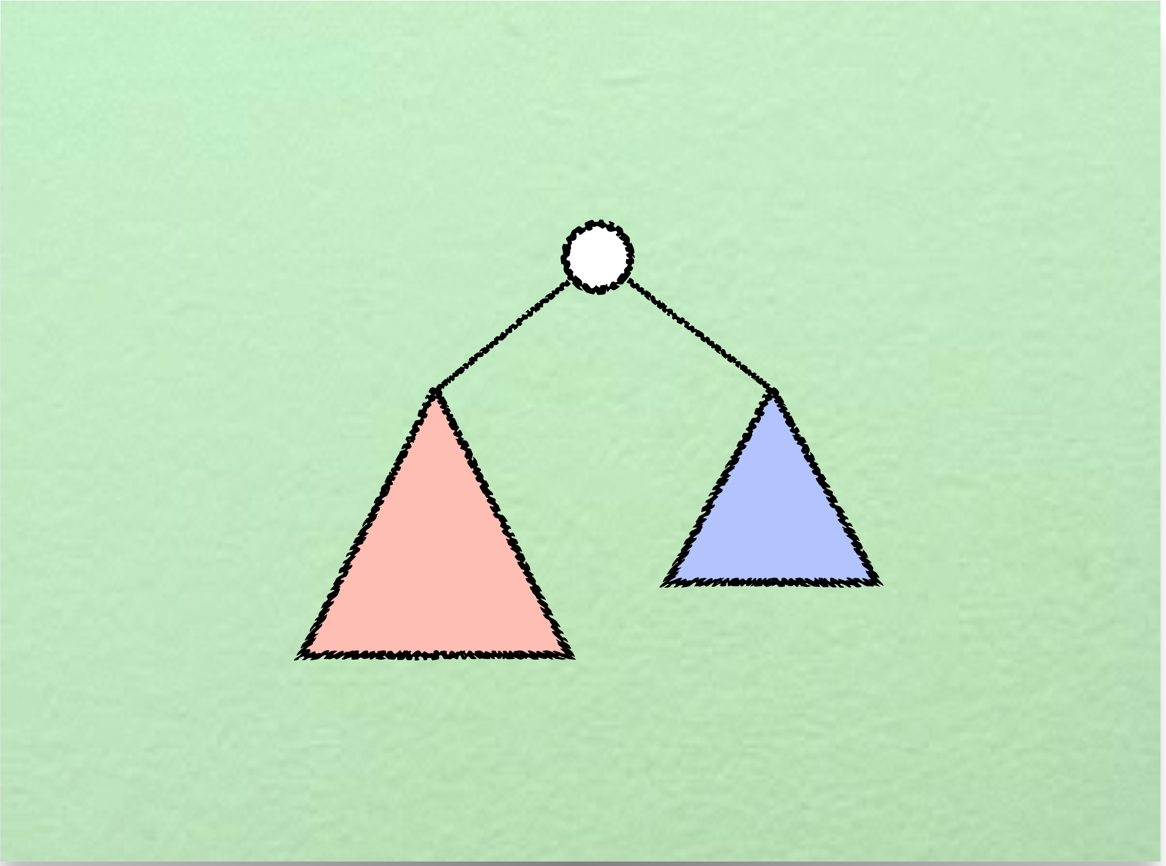
Theorem. S_1, S_2 are n-equivalent (they satisfy the same sentences with quantifier rank n)

whenever S_1, S_2 are Hanf(r,t)-equivalent, with $r = 3^n$ and $t = n$.

[Hanf '60]

Exercise: prove that testing whether a binary tree is *complete* is not FO-definable





Hanf locality

Theorem. S_1, S_2 are n-equivalent (they satisfy the same sentences with quantifier rank n) whenever S_1, S_2 are Hanf(r, t)-equivalent, with $r = 3^n$ and $t = n$.

[Hanf '60]

Hanf locality

Theorem. S_1, S_2 are n-equivalent (they satisfy the same sentences with quantifier rank n) whenever S_1, S_2 are Hanf(r,t)-equivalent, with $r = 3^n$ and $t = n$.

[Hanf '60]

Why so **BIG**?

Hanf locality

Theorem. S_1, S_2 are n-equivalent (they satisfy the same sentences with quantifier rank n) whenever S_1, S_2 are Hanf(r,t)-equivalent, with $r = 3^n$ and $t = n$. [Hanf '60]

Why so **BIG**?

Remember $\mathbf{d}_k(x,y)$ = "there is a path of length 2^k from x to y"

Hanf locality

Theorem. S_1, S_2 are n-equivalent (they satisfy the same sentences with quantifier rank n)
whenever S_1, S_2 are Hanf(r, t)-equivalent, with $r = 3^n$ and $t = n$.
[Hanf '60]

Why so **BIG**?

Remember $\mathbf{d}_k(x,y)$ = "there is a path of length 2^k from x to y"

$$d_0(x, y) = E(x, y), \text{ and}$$
$$d_k(x,y) = \exists z\, (\, d_{k-1}(x, z) \wedge d_{k-1}(z, y)\,)$$

$\mathrm{qr}(\mathbf{d}_k) = k$

Hanf locality

Theorem. S_1, S_2 are n-equivalent (they satisfy the same sentences with quantifier rank n)

whenever S_1, S_2 are Hanf(r,t)-equivalent, with $r = 3^n$ and $t = n$.

[Hanf '60]

Why so **BIG**?

Remember $\mathbf{d}_k(x,y)$ = "there is a path of length 2^k from x to y"

$$d_0(x, y) = E(x, y), \text{ and}$$
$$d_k(x,y) = \exists z \, (d_{k-1}(x, z) \wedge d_{k-1}(z, y))$$

$qr(\mathbf{d}_k) = k$

$2{\cdot}2^n{+}1$

$2{\cdot}2^n$

Hanf locality

Theorem. S_1, S_2 are n-equivalent (they satisfy the same sentences with quantifier rank n)

whenever S_1, S_2 are Hanf(r,t)-equivalent, with $r = 3^n$ and $t = n$.

[Hanf '60]

Why so **BIG**?

Remember $\mathbf{d}_k(x,y)$ = "there is a path of length 2^k from x to y"

$d_0(x, y) = E(x, y)$, and
$d_k(x,y) = \exists z\ (\ d_{k-1}(x, z) \wedge d_{k-1}(z, y)\ )$

$qr(\mathbf{d}_k) = k$

$2{\cdot}2^n{+}1$ $\qquad\qquad$ $2{\cdot}2^n$

Not (n+2)-equivalent yet they have the same $2^n{-}1$ balls.

Gaifman locality

What about queries?

Eg: Is reachability expressible in FO?

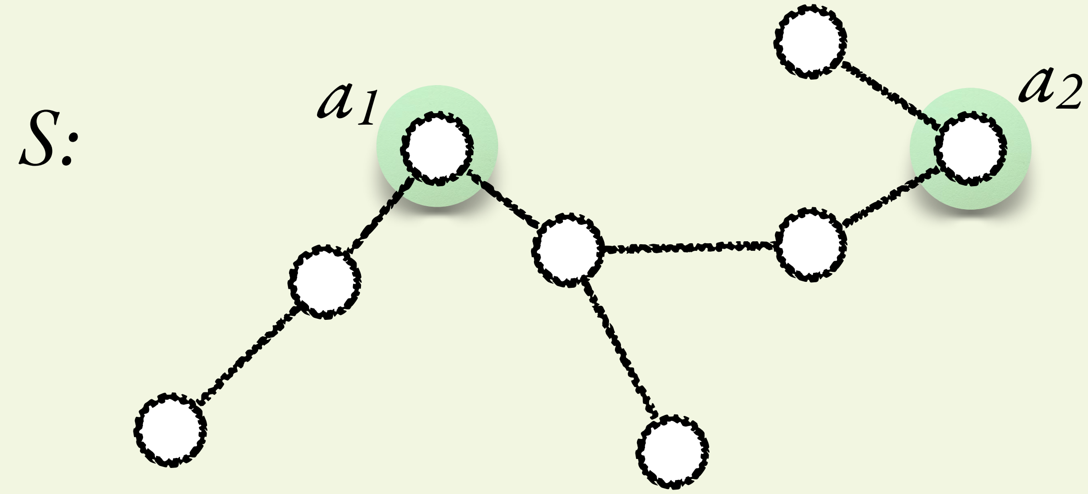
What about equivalence on the same structure?

When are two points indistinguishable?

Gaifman locality

$S[(a_1, \ldots, a_n), r]$ = induced substructure of S

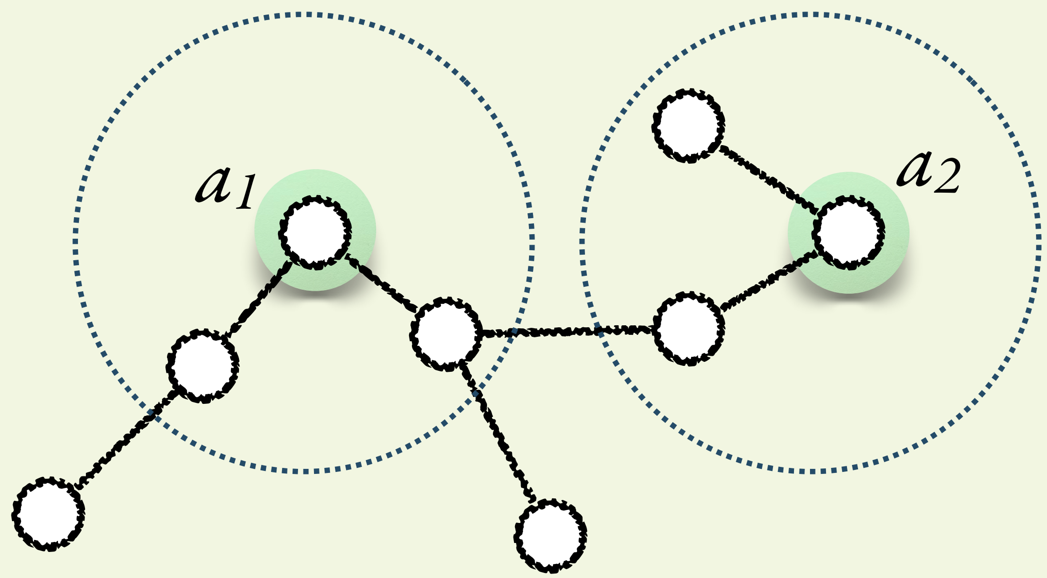
of elements at distance $\leq r$ of some a_i in the Gaifman graph.



Gaifman locality

$S[(a_1, ..., a_n), r]$ = induced substructure of S

of elements at distance $\leq r$ of some a_i in the Gaifman graph.

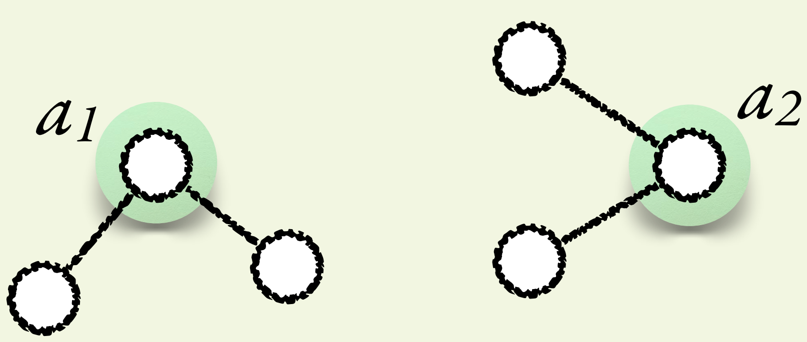


$S[(a_1, a_2), 1]$

Gaifman locality

$S[(a_1, \ldots, a_n), r]$ = induced substructure of S

of elements at distance $\leq r$ of some a_i in the Gaifman graph.



$S[(a_1, a_2), 1]$

Gaifman locality

$S[(a_1, \ldots, a_n), r]$ = induced substructure of S

of elements at distance $\leq r$ of some a_i in the Gaifman graph.

Gaifman locality

For any $\phi \in$ FO of quantifier rank k and structure S,

$$S[(a_1, \ldots, a_n), r] \cong S[(b_1, \ldots, b_n), r] \quad \text{for } r = 3^{k+1}$$

implies

$$(a_1, \ldots, a_n) \in \phi(S) \quad \text{iff} \quad (b_1, \ldots, b_n) \in \phi(S)$$

Idea: If the neighbourhoods of two tuples are the same,
the formula cannot distinguish them.

Gaifman locality vs Hanf locality

Difference between Hanf- and Gaifman-locality:

Hanf-locality relates **two different structures,**

Gaifman-locality talks about definability in **one structure**

Gaifman locality vs Hanf locality

Difference between Hanf- and Gaifman-locality:

Hanf-locality relates **two different structures**,

Gaifman-locality talks about definability in **one structure**

Inside S,

3^{k+1}-balls of $(a_1,\ldots,a_n) = 3^{k+1}$-balls of $(b_1,\ldots,b_n)$

$\Downarrow$

$(a_1,\ldots,a_n)$ indistinguishable from $(b_1,\ldots,b_n)$ through **formulas** of qr $\leq k$

Gaifman locality vs Hanf locality

Difference between Hanf- and Gaifman-locality:

Hanf-locality relates **two different structures**,

S_1 and S_2 have the same # of balls of radius 3^k, **up to threshold *k***

$\Downarrow$

They verify the same **sentences** of qr $\leq k$

Gaifman-locality talks about definability in **one structure**

Inside S,

3^{k+1}-balls of $(a_1,\ldots,a_n) = 3^{k+1}$-balls of $(b_1,\ldots,b_n)$

$\Downarrow$

$(a_1,\ldots,a_n)$ indistinguishable from $(b_1,\ldots,b_n)$ through **formulas** of qr $\leq k$

Gaifman locality

Schema to show non-expressibility results is, as usual:

A query $Q(x_1,\ldots,x_n)$ is not FO-definable if:

for every $\boldsymbol{k}$ there is a structure S_k and $(a_1, \ldots, a_n)$, $(b_1, \ldots, b_n)$ such that

- $S_k[(a_1, \ldots, a_n), 3^{k+1}] \cong S_k[(b_1, \ldots, b_n), 3^{k+1}]$
- $(a_1, \ldots, a_n) \in Q(S_k)$, $(b_1, \ldots, b_n) \notin Q(S_k)$

Proof: If Q were expressible with a formula of quantifier rank k,

then $(a_1, \ldots, a_n) \in Q(S_k)$ iff $(b_1, \ldots, b_n) \in Q(S_k)$. Absurd!

Gaifman locality

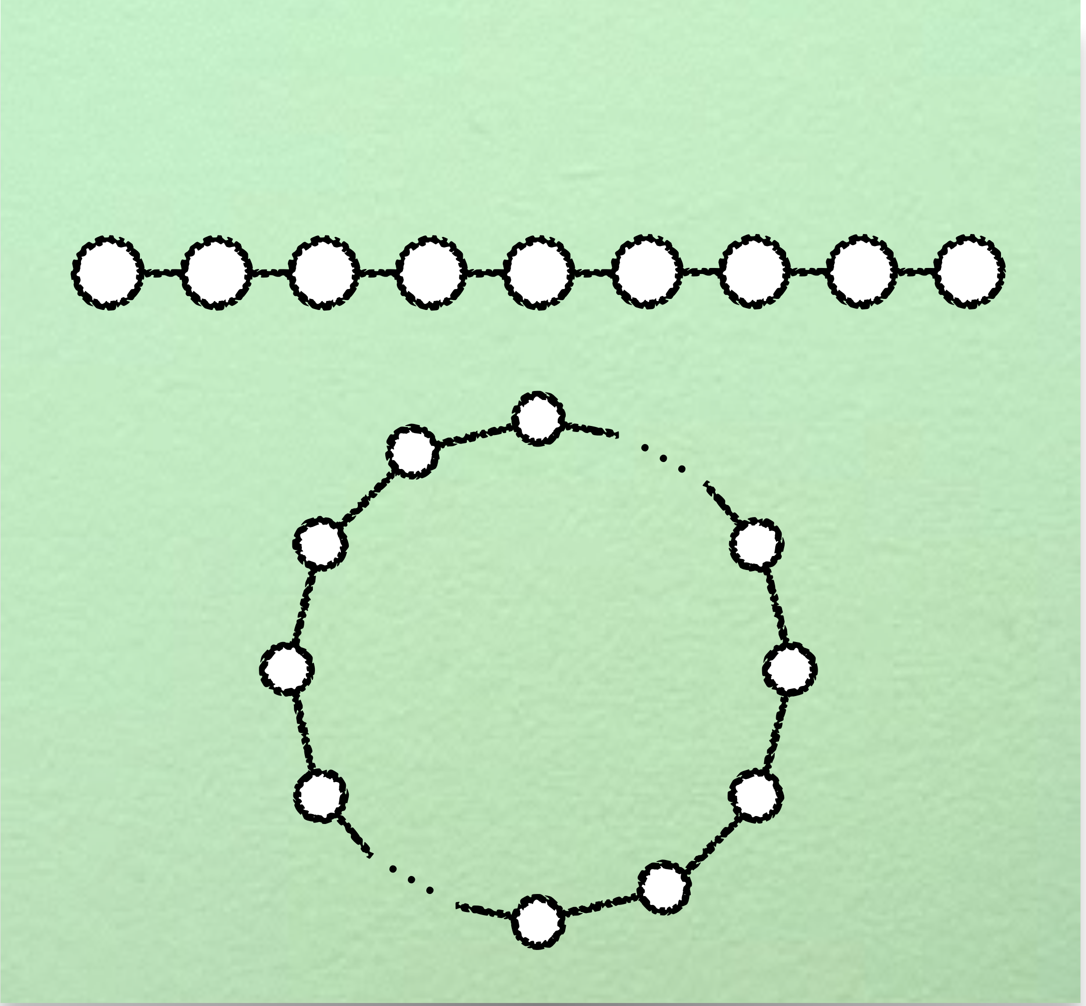
Reachability is not FO definable.

For every k, we build S_k :

Gaifman locality

Reachability is not FO definable.

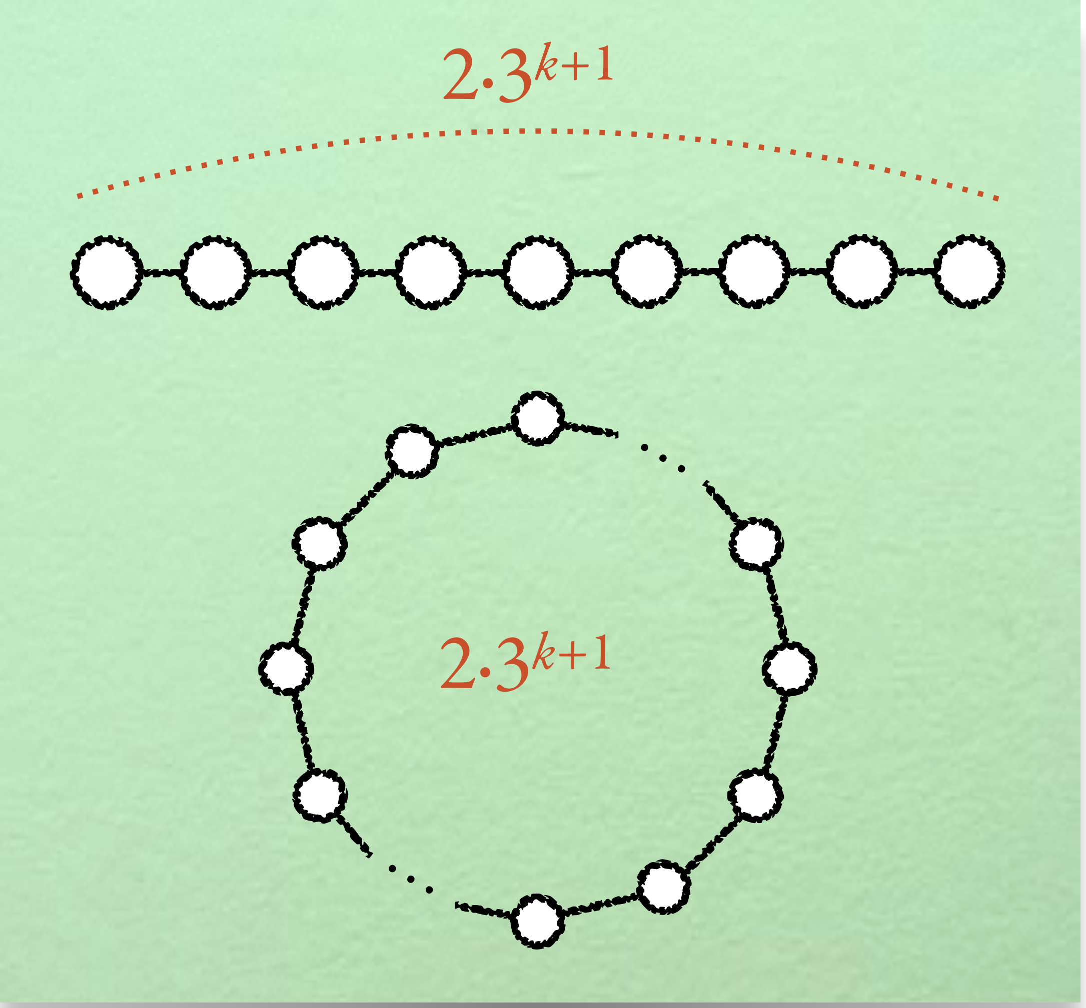
For every k, we build S_k :



Gaifman locality

Reachability is not FO definable.

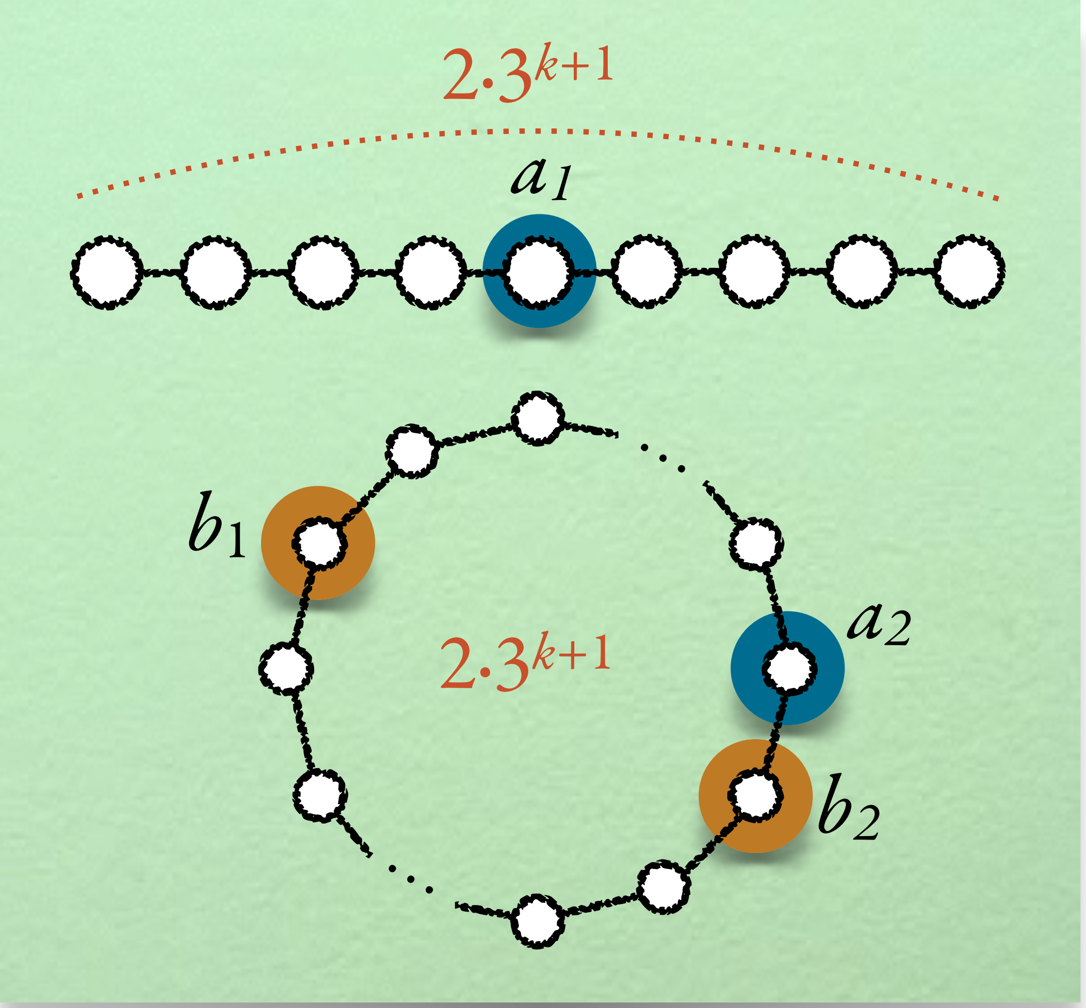
For every k, we build S_k :



Gaifman locality

Reachability is not FO definable.

For every k, we build S_k :

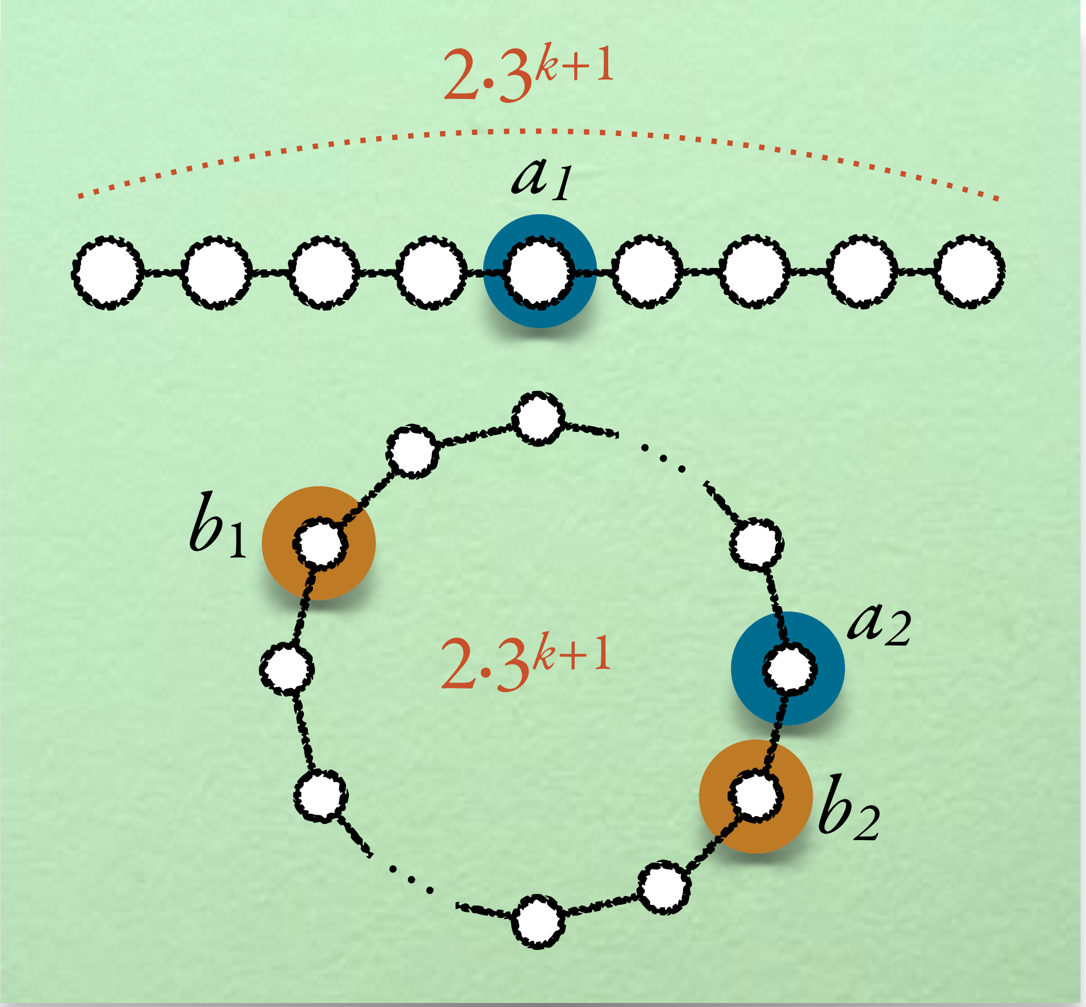


Gaifman locality

Reachability is not FO definable.

For every k, we build S_k :

And $S_k\,[(a_1, a_2), 3^{k+1}] \;\cong\; S_k\,[(b_1, b_2), 3^{k+1}]$



Gaifman locality

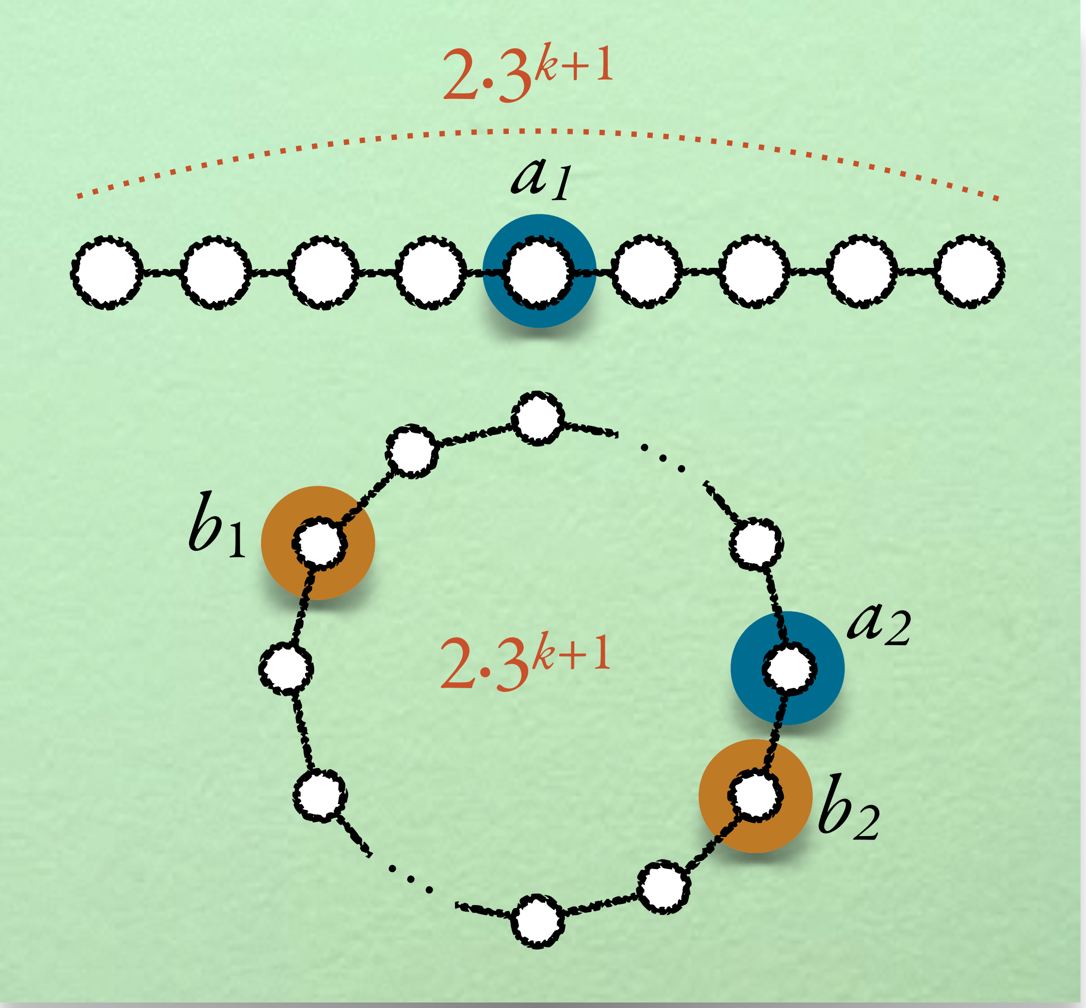
Reachability is not FO definable.

For every k, we build S_k :

And $S_k[(a_1, a_2), 3^{k+1}] \cong S_k[(b_1, b_2), 3^{k+1}]$

However,

- b_2 is reachable from b_1,
- a_2 is **not** reachable from a_1.



Gaifman locality

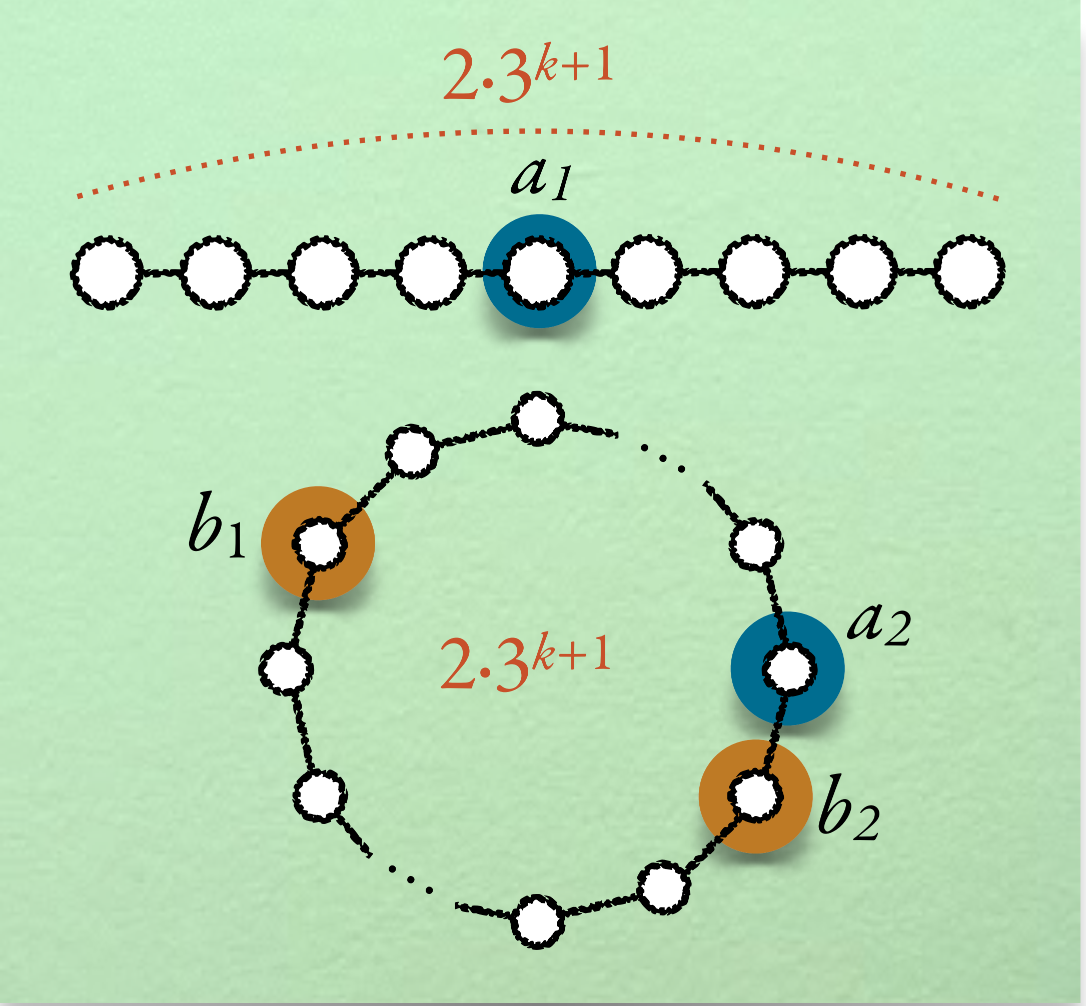
Reachability is not FO definable.

For every k, we build S_k :

And $S_k[(a_1, a_2), 3^{k+1}] \cong S_k[(b_1, b_2), 3^{k+1}]$

However,

- b_2 is reachable from b_1,
- a_2 is **not** reachable from a_1.

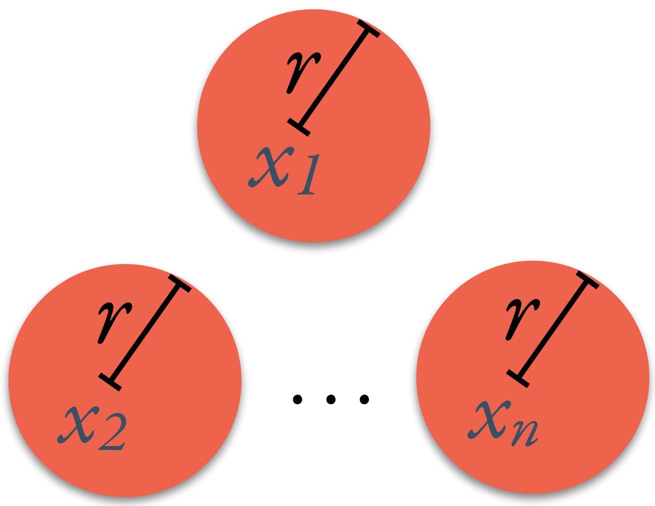


Your turn! Q(x) = "x is a vertex separator"

Gaifman Theorem

Basic local sentence:

$\exists\, x_1, \ldots, x_n$

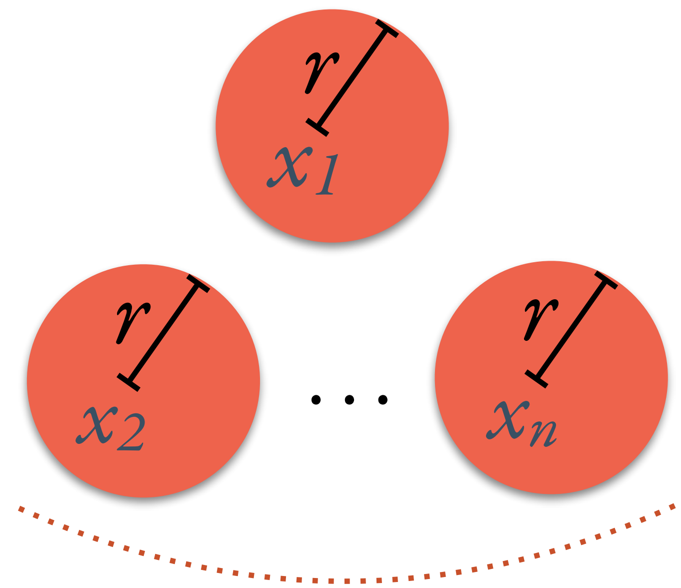


$\wedge\ \psi_1(x_1) \wedge \cdots \wedge \psi_n(x_n)$

Gaifman Theorem

Basic local sentence:

$\exists x_1, \ldots, x_n$



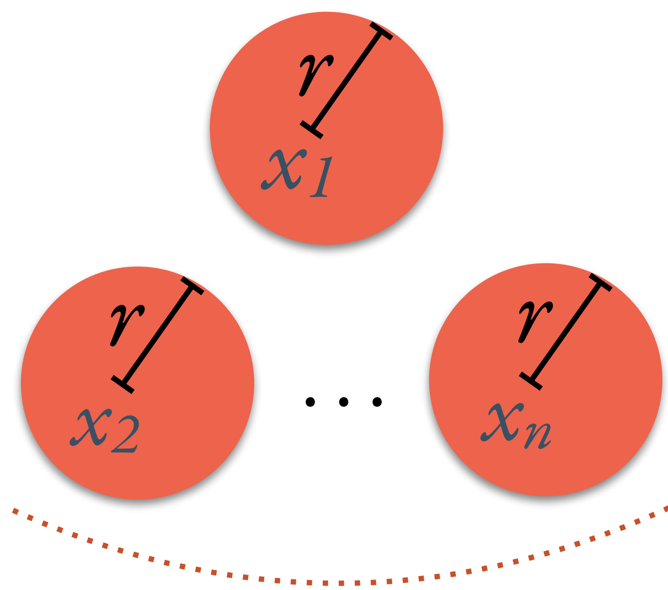
$\wedge\ \psi_1(x_1) \wedge \cdots \wedge \psi_n(x_n)$

disjoint r-balls around $x_1, \ldots, x_n$

Gaifman Theorem

Basic local sentence:

$\exists\, x_1, \ldots, x_n$ $\wedge\, \psi_1(x_1) \wedge \cdots \wedge \psi_n(x_n)$



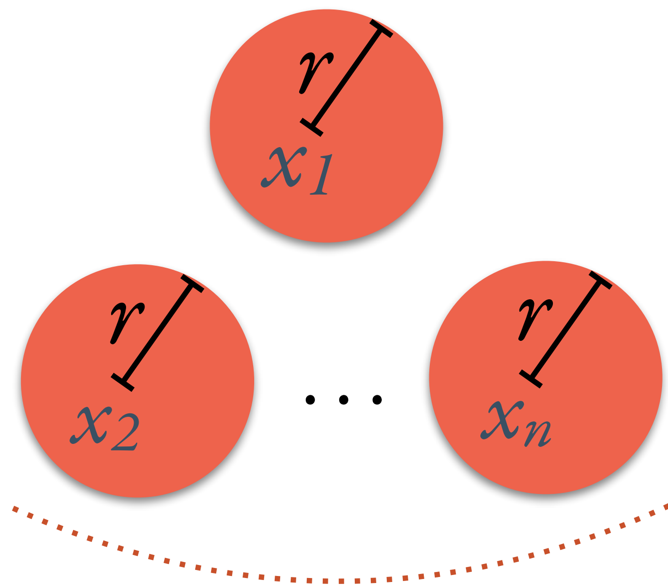
disjoint r-balls around $x_1, \ldots, x_n$

r-local formulas

Gaifman Theorem

Basic local sentence:

$\exists\, x_1, \ldots, x_n$



disjoint r-balls around $x_1, \ldots, x_n$

$\wedge\, \psi_1(x_1) \wedge \cdots \wedge \psi_n(x_n)$

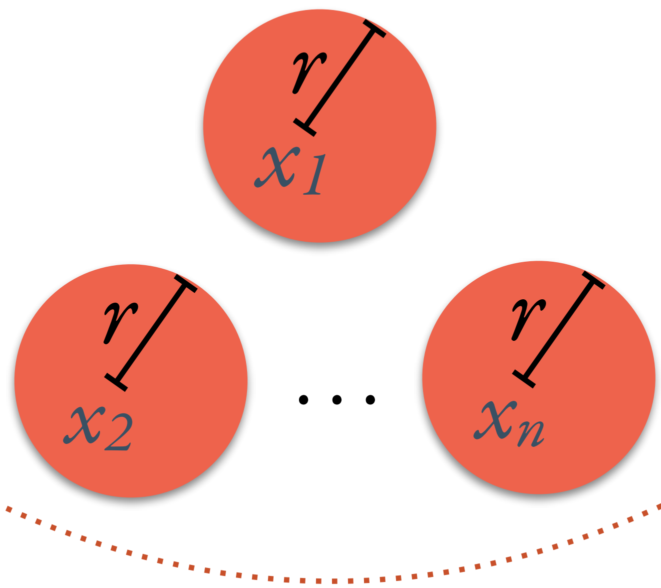
r-local formulas

Inside $\psi_i(x_i)$ we interpret $\exists y \,.\, \phi$ as $\exists y \,.\, \mathrm{d}(x_i, y) \leq r \wedge \phi$

Gaifman Theorem

Basic local sentence:

$\exists\, x_1, \ldots, x_n$



disjoint r-balls around $x_1, \ldots, x_n$

$\wedge\ \psi_1(x_1) \wedge \cdots \wedge \psi_n(x_n)$

r-local formulas

Inside $\psi_i(x_i)$ we interpret
$\exists y \,.\, \phi$ as $\exists y \,.\, \mathrm{d}(x_i, y) \leq r \wedge \phi$

Gaifman Theorem: Every FO sentence is equivalent to a boolean combination of **basic local sentences**.

Recap

Recap

EF games

FO sentences with quantifier rank n

=

winning strategies for Spoiler in the n-round EF game

Recap

EF games

FO sentences with quantifier rank n

=

winning strategies for Spoiler in the n-round EF game

0-1 Law

FO sentences are almost always true or almost always false

Recap

EF games

FO sentences with quantifier rank n

=

winning strategies for Spoiler in the n-round EF game

0-1 Law

FO sentences are almost always true or almost always false

Hanf locality

FO sentences with quantifier rank n

=

counting 3^n sized balls up to n

Recap

EF games	FO sentences with quantifier rank n = winning strategies for Spoiler in the n-round EF game
0-1 Law	FO sentences are almost always true or almost always false
Hanf locality	FO sentences with quantifier rank n = counting 3^n sized balls up to n
Gaifman locality	Queries of quantifier rank n output tuples closed under 3^{n+1} balls.

Recap

EF games

FO sentences with quantifier rank n

=

winning strategies for Spoiler in the n-round EF game

0-1 Law

FO sentences are almost always true or almost always false

Hanf locality

FO sentences with quantifier rank n

=

counting 3^n sized balls up to n

Gaifman locality

Queries of quantifier rank n output tuples closed under 3^{n+1} balls.

Gaifman Theorem

An FO sentence can only say

"there are some points at distance ≥2r

whose r-balls are isomorphic to certain structures"

or a boolean combination of that.

Some Bibliography

- Libkin, "Elements of Finite Model Theory", Springer, 2004.

- Otto, "Finite Model Theory", Springer, 2005

  (freely available at www.mathematik.tu-darmstadt.de/~otto/LEHRE/FMT0809.ps)

- Väänänen, "A Short course on Finite Model Theory", 1994.

  (available at www.math.helsinki.fi/logic/people/jouko.vaananen/shortcourse.pdf)