Chase-Based Computation of Cores for Existential Rules

Lukas Gerlach

Knowledge-Based Systems Group
Technische Universität Dresden

16.09.2021

Existential Rules

$$\forall \vec{x} \forall \vec{y} . Body(\vec{x}, \vec{y}) \rightarrow \exists \vec{z} . Head(\vec{x}, \vec{z})$$

Existential Rules

$$\forall \vec{x} \forall \vec{y}. Body(\vec{x}, \vec{y}) \rightarrow \exists \vec{z}. Head(\vec{x}, \vec{z})$$

- *Body* and *Head*: conjunctions of atoms
- $\vec{x}, \vec{y}, \vec{z}$: pairwise disjoint lists of variables

Existential Rules

$$\forall \vec{x} \forall \vec{y}. Body(\vec{x}, \vec{y}) \rightarrow \exists \vec{z}. Head(\vec{x}, \vec{z})$$

- *Body* and *Head*: conjunctions of atoms
- $\vec{x}, \vec{y}, \vec{z}$: pairwise disjoint lists of variables

$$Pizza(x) \rightarrow \exists z. SameDeliverer(x, z) \wedge Pizza(z)$$

Existential Rules

$$\forall \vec{x} \forall \vec{y}. Body(\vec{x}, \vec{y}) \rightarrow \exists \vec{z}. Head(\vec{x}, \vec{z})$$

- *Body* and *Head*: conjunctions of atoms
- $\vec{x}, \vec{y}, \vec{z}$: pairwise disjoint lists of variables

$$Pizza(x) \rightarrow \exists z. SameDeliverer(x, z) \wedge Pizza(z)$$

$$WeeklyOrder(y, x) \rightarrow \exists z. WeeklyOrder(x, z)$$

Existential Rules

$$\forall \vec{x} \forall \vec{y}. Body(\vec{x}, \vec{y}) \to \exists \vec{z}. Head(\vec{x}, \vec{z})$$

- *Body* and *Head*: conjunctions of atoms
- $\vec{x}, \vec{y}, \vec{z}$: pairwise disjoint lists of variables

$$Pizza(x) \to \exists z. SameDeliverer(x, z) \land Pizza(z)$$

$$WeeklyOrder(y, x) \to \exists z. WeeklyOrder(x, z)$$

$$Pizza(x) \land WeeklyOrder(x, y) \to Pizza(y) \land SameDeliverer(x, y)$$

Chasing a Universal Core Model

$$Pizza(x) \rightarrow \exists z.SameDeliverer(x, z) \wedge Pizza(z)$$

$$WeeklyOrder(y, x) \rightarrow \exists z.WeeklyOrder(x, z)$$

$$Pizza(x) \wedge WeeklyOrder(x, y) \rightarrow Pizza(y) \wedge SameDeliverer(x, y)$$

Chasing a Universal Core Model

$$Pizza(x) \rightarrow \exists z. SameDeliverer(x, z) \land Pizza(z)$$

$$WeeklyOrder(y, x) \rightarrow \exists z. WeeklyOrder(x, z)$$

$$Pizza(x) \land WeeklyOrder(x, y) \rightarrow Pizza(y) \land SameDeliverer(x, y)$$

Chasing a Universal Core Model

$$Pizza(x) \rightarrow \exists z. SameDeliverer(x, z) \wedge Pizza(z)$$

$$WeeklyOrder(y, x) \rightarrow \exists z. WeeklyOrder(x, z)$$

$$Pizza(x) \wedge WeeklyOrder(x, y) \rightarrow Pizza(y) \wedge SameDeliverer(x, y)$$

Chasing a Universal Core Model

$$Pizza(x) \rightarrow \exists z.SameDeliverer(x, z) \wedge Pizza(z)$$

$$WeeklyOrder(y, x) \rightarrow \exists z.WeeklyOrder(x, z)$$

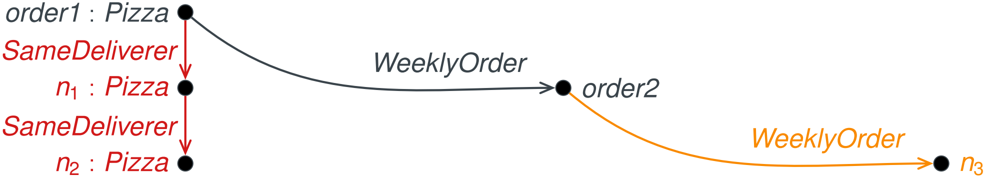
$$Pizza(x) \wedge WeeklyOrder(x, y) \rightarrow Pizza(y) \wedge SameDeliverer(x, y)$$

Chasing a Universal Core Model

$$Pizza(x) \rightarrow \exists z. SameDeliverer(x, z) \wedge Pizza(z)$$

$$WeeklyOrder(y, x) \rightarrow \exists z. WeeklyOrder(x, z)$$

$$Pizza(x) \wedge WeeklyOrder(x, y) \rightarrow Pizza(y) \wedge SameDeliverer(x, y)$$

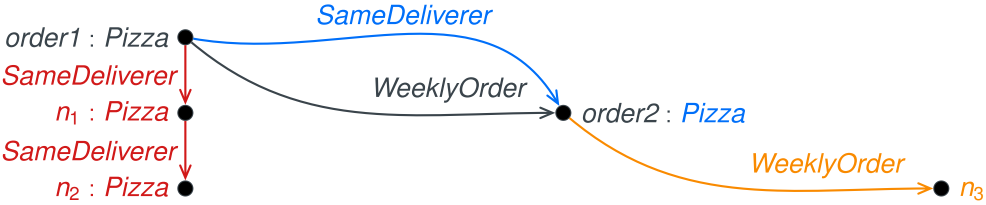


Chasing a Universal Core Model

$$Pizza(x) \to \exists z. SameDeliverer(x, z) \land Pizza(z)$$

$$WeeklyOrder(y, x) \to \exists z. WeeklyOrder(x, z)$$

$$Pizza(x) \land WeeklyOrder(x, y) \to Pizza(y) \land SameDeliverer(x, y)$$

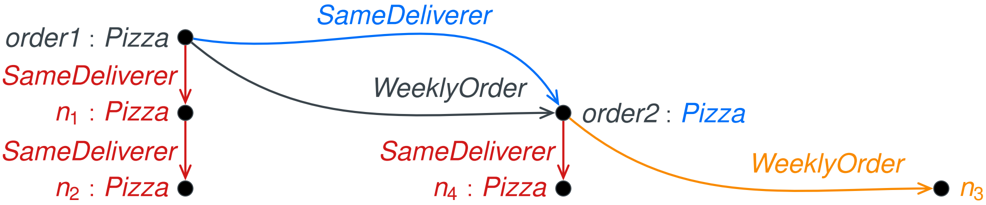


Chasing a Universal Core Model

$$Pizza(x) \rightarrow \exists z. SameDeliverer(x, z) \wedge Pizza(z)$$

$$WeeklyOrder(y, x) \rightarrow \exists z. WeeklyOrder(x, z)$$

$$Pizza(x) \wedge WeeklyOrder(x, y) \rightarrow Pizza(y) \wedge SameDeliverer(x, y)$$

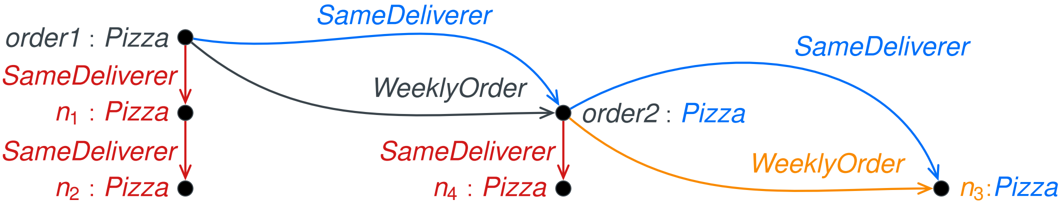


Chasing a Universal Core Model

$$Pizza(x) \rightarrow \exists z. SameDeliverer(x, z) \wedge Pizza(z)$$

$$WeeklyOrder(y, x) \rightarrow \exists z. WeeklyOrder(x, z)$$

$$Pizza(x) \wedge WeeklyOrder(x, y) \rightarrow Pizza(y) \wedge SameDeliverer(x, y)$$

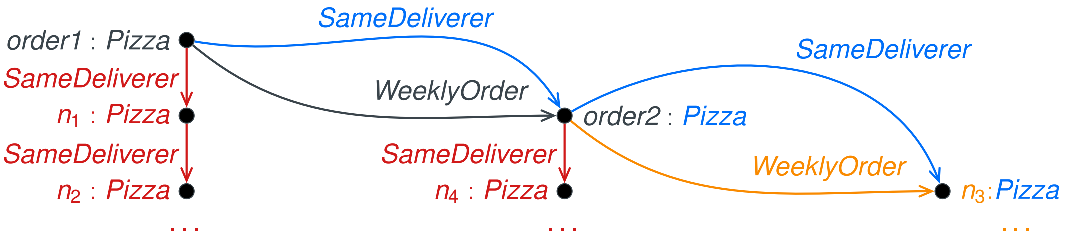


Chasing a Universal Core Model

$$Pizza(x) \rightarrow \exists z. SameDeliverer(x, z) \land Pizza(z)$$

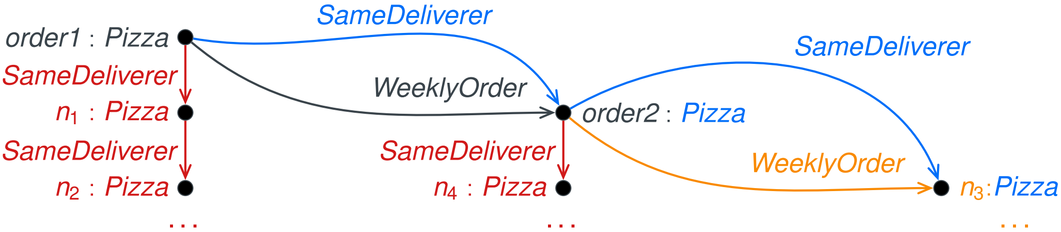
$$WeeklyOrder(y, x) \rightarrow \exists z. WeeklyOrder(x, z)$$

$$Pizza(x) \land WeeklyOrder(x, y) \rightarrow Pizza(y) \land SameDeliverer(x, y)$$



Chasing a Universal Core Model

$$Pizza(x) \to \exists z. SameDeliverer(x, z) \wedge Pizza(z)$$
$$WeeklyOrder(y, x) \to \exists z. WeeklyOrder(x, z)$$
$$Pizza(x) \wedge WeeklyOrder(x, y) \to Pizza(y) \wedge SameDeliverer(x, y)$$



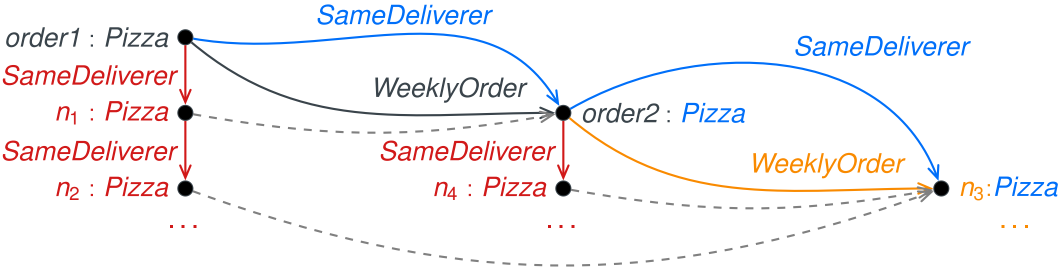
The *restricted chase* and it yields a *universal model*.

Chasing a Universal Core Model

$$Pizza(x) \to \exists z. SameDeliverer(x, z) \land Pizza(z)$$

$$WeeklyOrder(y, x) \to \exists z. WeeklyOrder(x, z)$$

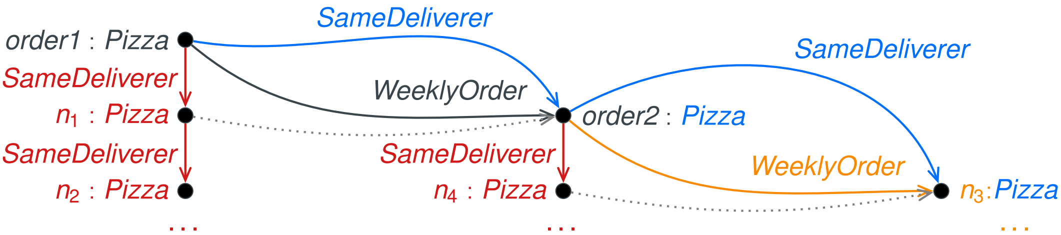
$$Pizza(x) \land WeeklyOrder(x, y) \to Pizza(y) \land SameDeliverer(x, y)$$



The *restricted chase* and it yields a *universal model*.
The *core chase* and it yields a *universal core model*.

Chasing a Universal Core Model

$$Pizza(x) \rightarrow \exists z. SameDeliverer(x, z) \wedge Pizza(z)$$
$$WeeklyOrder(y, x) \rightarrow \exists z. WeeklyOrder(x, z)$$
$$Pizza(x) \wedge WeeklyOrder(x, y) \rightarrow Pizza(y) \wedge SameDeliverer(x, y)$$



The *restricted chase* and it yields a *universal model*.
The *core chase* and it yields a *universal core model*.
Without *alternative matches*, the *restricted chase* also yields a *universal core model*.

Some Rule Set Classifications

$$R \in \mathrm{CT}_{\forall\exists}^{res}$$

For the rule set R and all starting fact sets, some restricted chase sequence terminates.

Some Rule Set Classifications

$$R \in \mathrm{CT}_{\forall\exists}^{res}$$

For the rule set R and all starting fact sets, some restricted chase sequence terminates.

$$\mathrm{CT}_{\forall\forall}^{res} \subset \mathrm{CT}_{\forall\exists}^{res} \subset \mathrm{CT}_{\forall}^{core}$$

Some Rule Set Classifications

$$R \in \mathrm{CT}_{\forall\exists}^{res}$$

For the rule set R and all starting fact sets, some restricted chase sequence terminates.

$$\mathrm{CT}_{\forall\forall}^{res} \subset \mathrm{CT}_{\forall\exists}^{res} \subset \mathrm{CT}_{\forall}^{core}$$

Each of these classes is undecidable.

Some Rule Set Classifications

$$R \in \mathsf{CT}_{\forall\exists}^{res}$$

For the rule set R and all starting fact sets, some restricted chase sequence terminates.

$$\mathsf{CT}_{\forall\forall}^{res} \subset \mathsf{CT}_{\forall\exists}^{res} \subset \mathsf{CT}_{\forall}^{core}$$

Each of these classes is undecidable.

$$R \in \overline{\mathsf{AM}}_{\forall\exists}$$

For the rule set R and all starting fact sets, some restricted chase sequence does not have an alternative match.

Some Rule Set Classifications

$$R \in \mathsf{CT}_{\forall\exists}^{res}$$

For the rule set R and all starting fact sets, some restricted chase sequence terminates.

$$\mathsf{CT}_{\forall\forall}^{res} \subset \mathsf{CT}_{\forall\exists}^{res} \subset \mathsf{CT}_{\forall}^{core}$$

Each of these classes is undecidable.

$$R \in \overline{\mathsf{AM}}_{\forall\exists}$$

For the rule set R and all starting fact sets, some restricted chase sequence does not have an alternative match.

$$\overline{\mathsf{AM}}_{\forall\forall} \subset \overline{\mathsf{AM}}_{\forall\exists}$$

Some Rule Set Classifications

$$R \in \mathsf{CT}_{\forall\exists}^{res}$$

For the rule set R and all starting fact sets, some restricted chase sequence terminates.

$$\mathsf{CT}_{\forall\forall}^{res} \subset \mathsf{CT}_{\forall\exists}^{res} \subset \mathsf{CT}_{\forall}^{core}$$

Each of these classes is undecidable.

$$R \in \overline{\mathsf{AM}}_{\forall\exists}$$

For the rule set R and all starting fact sets, some restricted chase sequence does not have an alternative match.

$$\overline{\mathsf{AM}}_{\forall\forall} \subset \overline{\mathsf{AM}}_{\forall\exists}$$

$\overline{\mathsf{AM}}_{\forall\exists}$ is undecidable whereas $\overline{\mathsf{AM}}_{\forall\forall}$ is decidable.

Relations between Rules [Krötzsch, 2020]

Definition

A rule ρ *restrains* a rule ρ', written $\rho \prec^{\square} \rho'$, if the application of ρ after ρ' may introduce an alternative match for ρ'.

Relations between Rules [Krötzsch, 2020]

Definition

A rule ρ *restrains* a rule ρ', written $\rho \prec^{\square} \rho'$, if the application of ρ after ρ' may introduce an alternative match for ρ'.

Proposition

Consider a chase. If for each rule ρ, all rules ρ' with $\rho' \prec^{\square} \rho$ are applied exhaustively before ρ, then the chase does not have an alternative match.

Relations between Rules [Krötzsch, 2020]

Definition

A rule ρ *restrains* a rule ρ', written $\rho \prec^{\square} \rho'$, if the application of ρ after ρ' may introduce an alternative match for ρ'.

Proposition

Consider a chase. If for each rule ρ, all rules ρ' with $\rho' \prec^{\square} \rho$ are applied exhaustively before ρ, then the chase does not have an alternative match.

Proposition

A rule set R does not have restraining relations iff $R \in \overline{\mathsf{AM}}_{\forall\forall}$.

Relations between Rules [Krötzsch, 2020]

Definition

A rule ρ *restrains* a rule ρ', written $\rho \prec^{\square} \rho'$, if the application of ρ after ρ' may introduce an alternative match for ρ'.

Proposition

Consider a chase. If for each rule ρ, all rules ρ' with $\rho' \prec^{\square} \rho$ are applied exhaustively before ρ, then the chase does not have an alternative match.

Proposition

A rule set R does not have restraining relations iff $R \in \overline{\text{AM}}_{\forall\forall}$.

If we respect restraining relations, we find a restricted chase sequence that yields a core.

Relations between Rules [Krötzsch, 2020]

Definition

A rule ρ' *positively relies* on a rule ρ, written $\rho \prec^{\triangle} \rho'$, if the application of ρ may allow ρ' to be applied.

Relations between Rules [Krötzsch, 2020]

Definition

A rule ρ' *positively relies* on a rule ρ, written $\rho \prec^{\triangle} \rho'$, if the application of ρ may allow ρ' to be applied.

Definition

The *downward closure* $\rho{\downarrow}^{\square}$ of a rule ρ is the set containing each rule ρ' for that we find $\rho'((\prec^{\triangle})^{*} \circ \prec^{\square})^{+}\rho$, i.e. ρ' directly or indirectly restrains ρ.

Relations between Rules [Krötzsch, 2020]

Definition

A rule ρ' *positively relies* on a rule ρ, written $\rho \prec^{\triangle} \rho'$, if the application of ρ may allow ρ' to be applied.

Definition

The *downward closure* $\rho{\downarrow}^{\square}$ of a rule ρ is the set containing each rule ρ' for that we find $\rho'((\prec^{\triangle})^{*} \circ \prec^{\square})^{+}\rho$, i.e. ρ' directly or indirectly restrains ρ.

Definition

A rule set is *core-stratified* if for every rule ρ, we have $\rho \notin \rho{\downarrow}^{\square}$.

The Power of Core Stratification

Core-Stratification is sufficient for $\overline{\mathsf{AM}}_{\forall\exists}$

$$Pizza(x) \rightarrow \exists z. SameDeliverer(x, z) \wedge Pizza(z)$$
$$WeeklyOrder(y, x) \rightarrow \exists z. WeeklyOrder(x, z)$$
$$Pizza(x) \wedge WeeklyOrder(x, y) \rightarrow Pizza(y) \wedge SameDeliverer(x, y)$$

We find $\rho_3 \prec^{\square} \rho_1$, $\rho_2 \prec^{\triangle} \rho_3$, $\rho_3 \prec^{\triangle} \rho_1$ and $\rho_i \prec^{\triangle} \rho_i$ for every $i \in \{1, 2, 3\}$.
Thus, we have $\rho_1{\downarrow}^{\square} = \{\rho_2, \rho_3\}$ and $\rho_2{\downarrow}^{\square} = \rho_3{\downarrow}^{\square} = \emptyset$.

Core-Stratification is sufficient for $\overline{\mathrm{AM}}_{\forall\exists}$

$$Pizza(x) \rightarrow \exists z. SameDeliverer(x, z) \wedge Pizza(z)$$
$$WeeklyOrder(y, x) \rightarrow \exists z. WeeklyOrder(x, z)$$
$$Pizza(x) \wedge WeeklyOrder(x, y) \rightarrow Pizza(y) \wedge SameDeliverer(x, y)$$

We find $\rho_3 \prec^{\square} \rho_1$, $\rho_2 \prec^{\triangle} \rho_3$, $\rho_3 \prec^{\triangle} \rho_1$ and $\rho_i \prec^{\triangle} \rho_i$ for every $i \in \{1, 2, 3\}$.
Thus, we have $\rho_1{\downarrow}^{\square} = \{\rho_2, \rho_3\}$ and $\rho_2{\downarrow}^{\square} = \rho_3{\downarrow}^{\square} = \emptyset$.

Theorem

If a rule set R is core stratified, then $R \in \overline{\mathrm{AM}}_{\forall\exists}$.

(Originally $R \in \mathrm{CT}^{res}_{\forall\forall}$ is also required [Krötzsch, 2020].)

Avoiding Alternative Matches when Chasing

$$\mathit{Pizza}(x) \rightarrow \exists z.\mathit{SameDeliverer}(x, z) \wedge \mathit{Pizza}(z) \qquad (R_2)$$

$$\mathit{WeeklyOrder}(y, x) \rightarrow \exists z.\mathit{WeeklyOrder}(x, z) \qquad (\mathbf{R_1})$$

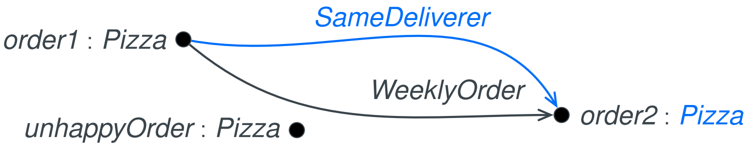
$$\mathit{Pizza}(x) \wedge \mathit{WeeklyOrder}(x, y) \rightarrow \mathit{Pizza}(y) \wedge \mathit{SameDeliverer}(x, y) \qquad (\mathbf{R_1})$$

Avoiding Alternative Matches when Chasing

$$Pizza(x) \to \exists z. SameDeliverer(x, z) \wedge Pizza(z) \qquad (R_2)$$

$$WeeklyOrder(y, x) \to \exists z. WeeklyOrder(x, z) \qquad (\mathbf{R_1})$$

$$Pizza(x) \wedge WeeklyOrder(x, y) \to Pizza(y) \wedge SameDeliverer(x, y) \qquad (\mathbf{R_1})$$



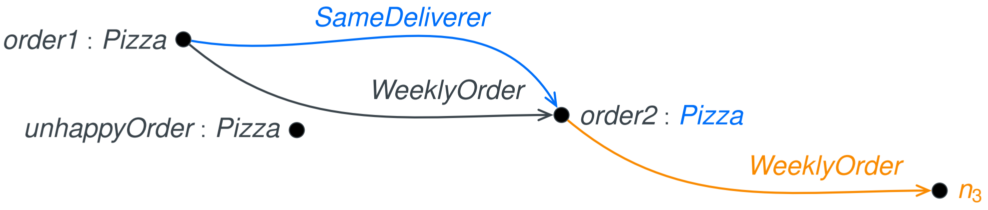
λ_1^1

Avoiding Alternative Matches when Chasing

$$Pizza(x) \rightarrow \exists z.SameDeliverer(x, z) \land Pizza(z) \qquad (R_2)$$

$$WeeklyOrder(y, x) \rightarrow \exists z.WeeklyOrder(x, z) \qquad (\mathbf{R_1})$$

$$Pizza(x) \land WeeklyOrder(x, y) \rightarrow Pizza(y) \land SameDeliverer(x, y) \qquad (\mathbf{R_1})$$



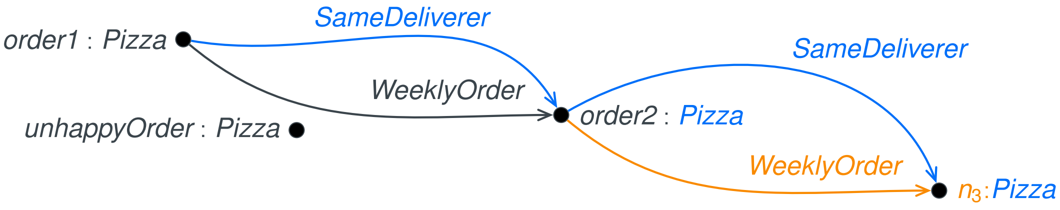
λ_1^1, λ_2^1

Avoiding Alternative Matches when Chasing

$$Pizza(x) \rightarrow \exists z.SameDeliverer(x,z) \wedge Pizza(z) \qquad (R_2)$$

$$WeeklyOrder(y,x) \rightarrow \exists z.WeeklyOrder(x,z) \qquad (\mathbf{R_1})$$

$$Pizza(x) \wedge WeeklyOrder(x,y) \rightarrow Pizza(y) \wedge SameDeliverer(x,y) \qquad (\mathbf{R_1})$$



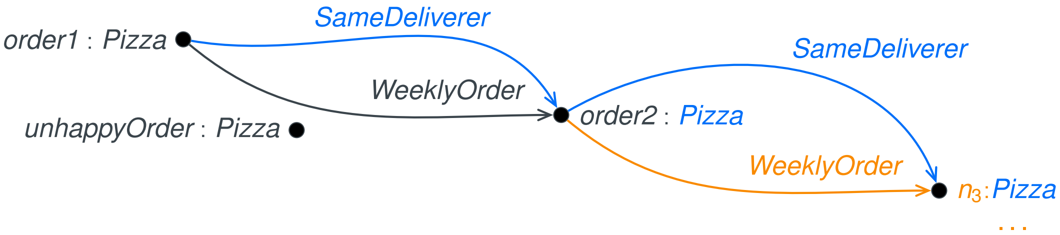
$\lambda_1^1, \lambda_2^1, \lambda_3^1$

Avoiding Alternative Matches when Chasing

$$Pizza(x) \rightarrow \exists z. SameDeliverer(x, z) \land Pizza(z) \qquad (R_2)$$

$$WeeklyOrder(y, x) \rightarrow \exists z. WeeklyOrder(x, z) \qquad (\mathbf{R_1})$$

$$Pizza(x) \land WeeklyOrder(x, y) \rightarrow Pizza(y) \land SameDeliverer(x, y) \qquad (\mathbf{R_1})$$



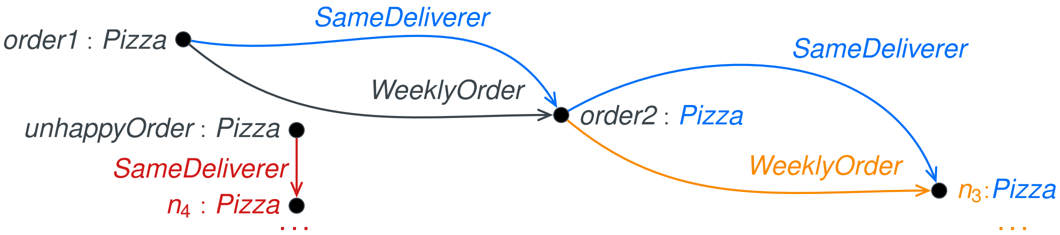
$\lambda_1^1, \lambda_2^1, \lambda_3^1, \lambda_4^1, \ldots$

Avoiding Alternative Matches when Chasing

$$Pizza(x) \to \exists z. SameDeliverer(x, z) \land Pizza(z) \qquad (\mathbf{R_2})$$

$$WeeklyOrder(y, x) \to \exists z. WeeklyOrder(x, z) \qquad (\mathbf{R_1})$$

$$Pizza(x) \land WeeklyOrder(x, y) \to Pizza(y) \land SameDeliverer(x, y) \qquad (\mathbf{R_1})$$



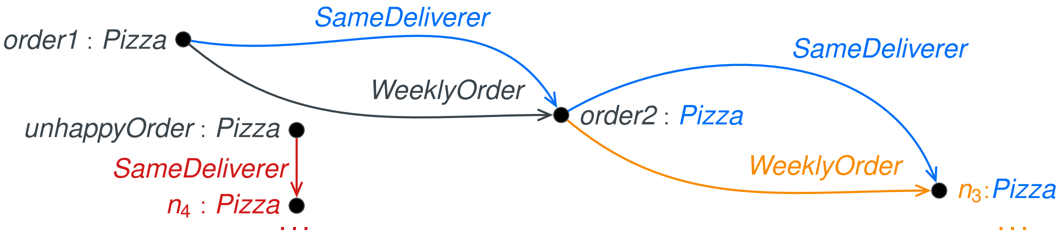
$$\lambda_1^1, \lambda_2^1, \lambda_3^1, \lambda_4^1, \ldots [, \lambda_1^2, \lambda_2^2, \ldots]$$

Avoiding Alternative Matches when Chasing

$$Pizza(x) \to \exists z. SameDeliverer(x, z) \wedge Pizza(z) \qquad (\mathbf{R_2})$$

$$WeeklyOrder(y, x) \to \exists z. WeeklyOrder(x, z) \qquad (\mathbf{R_1})$$

$$Pizza(x) \wedge WeeklyOrder(x, y) \to Pizza(y) \wedge SameDeliverer(x, y) \qquad (\mathbf{R_1})$$



$$\lambda_1^1, \lambda_2^1, \lambda_3^1, \lambda_4^1, \ldots [, \lambda_1^2, \lambda_2^2, \ldots]$$

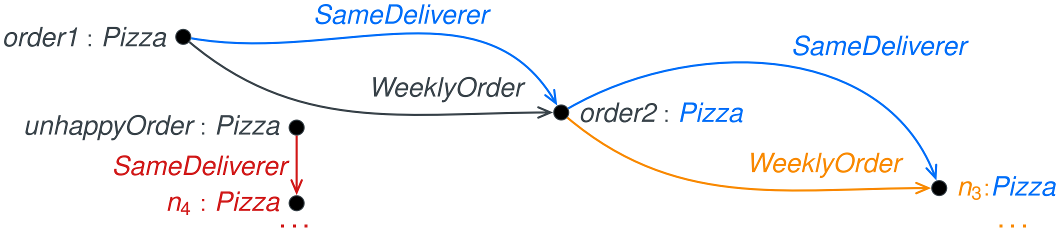
This is a *transfinite chase sequence* on a *restrained partitioning*.

Avoiding Alternative Matches when Chasing

$$Pizza(x) \to \exists z. SameDeliverer(x,z) \wedge Pizza(z) \qquad (\mathbf{R_2})$$

$$WeeklyOrder(y,x) \to \exists z. WeeklyOrder(x,z) \qquad (\mathbf{R_1})$$

$$Pizza(x) \wedge WeeklyOrder(x,y) \to Pizza(y) \wedge SameDeliverer(x,y) \qquad (\mathbf{R_1})$$



$\lambda_1^1, \lambda_2^1, \lambda_3^1, \lambda_4^1, \ldots [, \lambda_1^2, \lambda_2^2, \ldots]$

This is a *transfinite chase sequence* on a *restrained partitioning*.

Observation: ρ_1 applications could be done earlier: $\lambda_1^1, \lambda_2^1, \lambda_1^2, \lambda_3^1, \lambda_4^1, \lambda_2^2, \ldots$

Restricted and Core Chase coincide

Theorem

For a rule set $R \in \overline{\mathsf{AM}}_{\forall\exists}$*, we have* $R \in \mathsf{CT}^{res}_{\forall\exists}$ *iff* $R \in \mathsf{CT}^{core}_{\forall}$*.*

Restricted and Core Chase coincide

Theorem

For a rule set $R \in \overline{\mathsf{AM}}_{\forall\exists}$, we have $R \in \mathsf{CT}^{res}_{\forall\exists}$ iff $R \in \mathsf{CT}^{core}_{\forall}$.

Corollary

A transfinite chase sequence on a restrained partitioning terminates (yielding a finite universal core model) iff a finite universal (core) model exists.

Restricted and Core Chase coincide

Theorem

For a rule set $R \in \overline{\mathsf{AM}}_{\forall\exists}$, we have $R \in \mathsf{CT}^{res}_{\forall\exists}$ iff $R \in \mathsf{CT}^{core}_{\forall}$.

Corollary

A transfinite chase sequence on a restrained partitioning terminates (yielding a finite universal core model) iff a finite universal (core) model exists.

Theorem

For a rule set $R \in \overline{\mathsf{AM}}_{\forall\forall}$, we have $R \in \mathsf{CT}^{res}_{\forall\forall}$ iff $R \in \mathsf{CT}^{res}_{\forall\exists}$ iff $R \in \mathsf{CT}^{core}_{\forall}$.

($\mathsf{CT}^{res}_{\forall\forall}$ is decidable for single-head guarded existential rules.)

Decidability of $CT^{res}_{\forall\forall}$

Proposition (Fairness Theorem [Gogacz et al., 2020])

For a rule set of only single-head rules, if there exists an unfair non-terminating restricted chase sequence, then there exists a fair non-terminating restricted chase sequence.

Decidability of $\mathrm{CT}_{\forall\forall}^{res}$

Proposition (Fairness Theorem [Gogacz et al., 2020])

For a rule set of only single-head rules, if there exists an unfair non-terminating restricted chase sequence, then there exists a fair non-terminating restricted chase sequence.

The problem of the Fairness Theorem [Gogacz et al., 2020]:

$$S(a, b, b)$$

$$\rho_1 := S(x, y, y) \rightarrow \exists z. S(x, z, y) \wedge S(z, y, y)$$

$$\rho_2 := S(x, y, z) \rightarrow S(z, z, z)$$

Decidability of $\mathrm{CT}_{\forall\forall}^{res}$

Proposition (Fairness Theorem [Gogacz et al., 2020])

For a rule set of only single-head rules, if there exists an unfair non-terminating restricted chase sequence, then there exists a fair non-terminating restricted chase sequence.

The problem of the Fairness Theorem [Gogacz et al., 2020]:

$$S(a, b, b)$$
$$\rho_1 := S(x, y, y) \to \exists z. S(x, z, y) \wedge S(z, y, y)$$
$$\rho_2 := S(x, y, z) \to S(z, z, z)$$

By ρ_1, we obtain:

Decidability of $CT_{\forall\forall}^{res}$

Proposition (Fairness Theorem [Gogacz et al., 2020])

For a rule set of only single-head rules, if there exists an unfair non-terminating restricted chase sequence, then there exists a fair non-terminating restricted chase sequence.

The problem of the Fairness Theorem [Gogacz et al., 2020]:

$$S(a, b, b)$$

$$\rho_1 := S(x, y, y) \rightarrow \exists z. S(x, z, y) \wedge S(z, y, y)$$

$$\rho_2 := S(x, y, z) \rightarrow S(z, z, z)$$

By ρ_1, we obtain:

$S(a, n_1, b), S(n_1, b, b)$

Decidability of $CT_{\forall\forall}^{res}$

Proposition (Fairness Theorem [Gogacz et al., 2020])

For a rule set of only single-head rules, if there exists an unfair non-terminating restricted chase sequence, then there exists a fair non-terminating restricted chase sequence.

The problem of the Fairness Theorem [Gogacz et al., 2020]:

$$S(a, b, b)$$
$$\rho_1 := S(x, y, y) \to \exists z. S(x, z, y) \land S(z, y, y)$$
$$\rho_2 := S(x, y, z) \to S(z, z, z)$$

By ρ_1, we obtain:

$$S(a, n_1, b), S(n_1, b, b), S(n_1, n_2, b), S(n_2, b, b)$$

Decidability of $CT^{res}_{\forall\forall}$

Proposition (Fairness Theorem [Gogacz et al., 2020])

For a rule set of only single-head rules, if there exists an unfair non-terminating restricted chase sequence, then there exists a fair non-terminating restricted chase sequence.

The problem of the Fairness Theorem [Gogacz et al., 2020]:

$$
\begin{aligned}
& S(a, b, b) \\
\rho_1 &:= S(x, y, y) \to \exists z. S(x, z, y) \wedge S(z, y, y) \\
\rho_2 &:= S(x, y, z) \to S(z, z, z)
\end{aligned}
$$

By ρ_1, we obtain:

$S(a, n_1, b), S(n_1, b, b), S(n_1, n_2, b), S(n_2, b, b), \ldots$

Decidability of $CT^{res}_{\forall\forall}$

Proposition (Fairness Theorem [Gogacz et al., 2020])

For a rule set of only single-head rules, if there exists an unfair non-terminating restricted chase sequence, then there exists a fair non-terminating restricted chase sequence.

The problem of the Fairness Theorem [Gogacz et al., 2020]:

$$S(a, b, b)$$

$$\rho_1 \coloneqq S(x, y, y) \to \exists z. S(x, z, y) \wedge S(z, y, y)$$

$$\rho_2 \coloneqq S(x, y, z) \to S(z, z, z)$$

By ρ_1, we obtain:

$S(a, n_1, b), S(n_1, b, b), S(n_1, n_2, b), S(n_2, b, b), \ldots$

Any application of ρ_2 yields $S(b, b, b)$ and blocks all (further) applications of ρ_1.

Decidability of $CT_{\forall\forall}^{res}$

Proposition (Fairness Theorem [Gogacz et al., 2020])

For a rule set of only single-head rules, if there exists an unfair non-terminating restricted chase sequence, then there exists a fair non-terminating restricted chase sequence.

The problem of the Fairness Theorem [Gogacz et al., 2020]:

$$S(a, b, b)$$
$$\rho_1 := S(x, y, y) \to \exists z. S(x, z, y) \land S(z, y, y)$$
$$\rho_2 := S(x, y, z) \to S(z, z, z)$$

By ρ_1, we obtain:
$S(a, n_1, b), S(n_1, b, b), S(n_1, n_2, b), S(n_2, b, b), \ldots$
Any application of ρ_2 yields $S(b, b, b)$ and blocks all (further) applications of ρ_1.
ρ_2 restrains ρ_1 and no infinite fair sequence exists.

Decidability of $CT_{\forall\forall}^{res}$

Proposition (Fairness Theorem [Gogacz et al., 2020])

For a rule set of only single-head rules, if there exists an unfair non-terminating restricted chase sequence, then there exists a fair non-terminating restricted chase sequence.

The problem of the Fairness Theorem [Gogacz et al., 2020]:

$$S(a, b, b)$$
$$\rho_1 := S(x, y, y) \rightarrow \exists z.S(x, z, y) \wedge S(z, y, y)$$
$$\rho_2 := S(x, y, z) \rightarrow S(z, z, z)$$

By ρ_1, we obtain:

$S(a, n_1, b), S(n_1, b, b), S(n_1, n_2, b), S(n_2, b, b), \ldots$

Any application of ρ_2 yields $S(b, b, b)$ and blocks all (further) applications of ρ_1.

ρ_2 restrains ρ_1 and no infinite fair sequence exists.

However: There are also single-head rules that restrain each other.

Decidability of $\mathrm{CT}^{res}_{\forall\forall}$

Proposition (Fairness Theorem [Gogacz et al., 2020])

For a rule set of only single-head rules, if there exists an unfair non-terminating restricted chase sequence, then there exists a fair non-terminating restricted chase sequence.

The problem of the Fairness Theorem [Gogacz et al., 2020]:

$$S(a, b, b)$$
$$\rho_1 := S(x, y, y) \to \exists z.S(x, z, y) \land S(z, y, y)$$
$$\rho_2 := S(x, y, z) \to S(z, z, z)$$

By ρ_1, we obtain:
$S(a, n_1, b), S(n_1, b, b), S(n_1, n_2, b), S(n_2, b, b), \ldots$
Any application of ρ_2 yields $S(b, b, b)$ and blocks all (further) applications of ρ_1.
ρ_2 **strongly restrains** ρ_1 and no infinite fair sequence exists.

Decidability of $CT_{\forall\forall}^{res}$

Proposition (Fairness Theorem [Gogacz et al., 2020])

For a rule set of only single-head rules, if there exists an unfair non-terminating restricted chase sequence, then there exists a fair non-terminating restricted chase sequence.

Decidability of $CT_{\forall\forall}^{res}$

Proposition (Fairness Theorem [Gogacz et al., 2020])

For a rule set of only single-head rules, if there exists an unfair non-terminating restricted chase sequence, then there exists a fair non-terminating restricted chase sequence.

Theorem

For a rule set without strong restraining relations, if there exists an unfair non-terminating restricted chase sequence, then there exists a fair non-terminating restricted chase sequence.

Decidability of $\mathsf{CT}^{res}_{\forall\forall}$

Proposition (Fairness Theorem [Gogacz et al., 2020])

For a rule set of only single-head rules, if there exists an unfair non-terminating restricted chase sequence, then there exists a fair non-terminating restricted chase sequence.

Theorem

For a rule set without strong restraining relations, if there exists an unfair non-terminating restricted chase sequence, then there exists a fair non-terminating restricted chase sequence.

Conjecture

Consider a guarded rule set R without strong restraining relations. It is decidable if $R \in \mathsf{CT}^{res}_{\forall\forall}$.

(We obtain decidability for $\mathsf{CT}^{core}_{\forall}$ for guarded rule sets without restraining relations.)

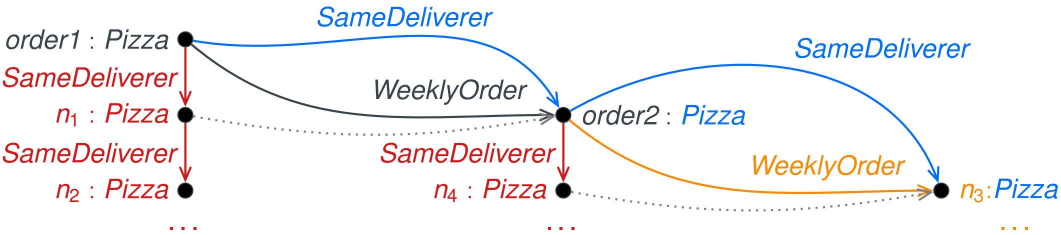
Computing Cores for Non-Core-Stratified Rule Sets

Computing Cores directly with Alternative Matches

$$Pizza(x) \to \exists z. SameDeliverer(x, z) \land Pizza(z)$$

$$WeeklyOrder(y, x) \to \exists z. WeeklyOrder(x, z)$$

$$Pizza(x) \land WeeklyOrder(x, y) \to Pizza(y) \land SameDeliverer(x, y)$$

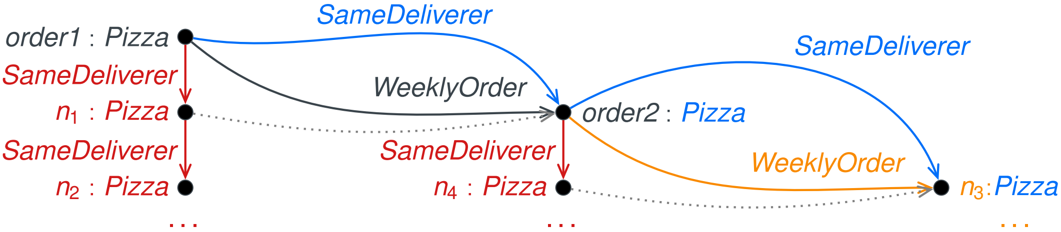


Computing Cores directly with Alternative Matches

$$Pizza(x) \rightarrow \exists z. SameDeliverer(x, z) \wedge Pizza(z)$$

$$WeeklyOrder(y, x) \rightarrow \exists z. WeeklyOrder(x, z)$$

$$Pizza(x) \wedge WeeklyOrder(x, y) \rightarrow Pizza(y) \wedge SameDeliverer(x, y)$$



Problem: Alternative Matches do not always yield an endomorphism over the fact set.

Compute Cores directly with Alternative Matches

$$\rightarrow \exists z . P(z)$$

$$P(x) \rightarrow \exists z . Q(x, z)$$

$$Q(x, y) \rightarrow \exists z . Q(z, y) \wedge Q(z, c) \wedge P(z) \wedge S(z, y)$$

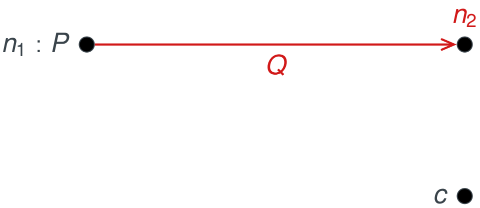
$$Q(x, y) \wedge S(x, z) \rightarrow S(x, y)$$

Compute Cores directly with Alternative Matches

$$\rightarrow \exists z.P(z)$$
$$P(x) \rightarrow \exists z.Q(x,z)$$
$$Q(x,y) \rightarrow \exists z.Q(z,y) \wedge Q(z,c) \wedge P(z) \wedge S(z,y)$$
$$Q(x,y) \wedge S(x,z) \rightarrow S(x,y)$$

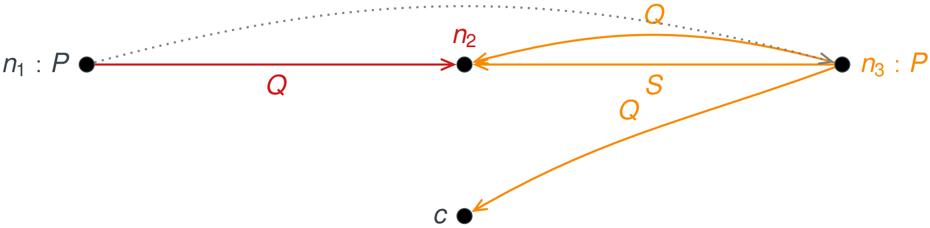
Compute Cores directly with Alternative Matches

$$\to \exists z.P(z)$$
$$P(x) \to \exists z.Q(x, z)$$
$$Q(x, y) \to \exists z.Q(z, y) \wedge Q(z, c) \wedge P(z) \wedge S(z, y)$$
$$Q(x, y) \wedge S(x, z) \to S(x, y)$$



Compute Cores directly with Alternative Matches

$$\rightarrow \exists z.P(z)$$
$$P(x) \rightarrow \exists z.Q(x,z)$$
$$Q(x,y) \rightarrow \exists z.Q(z,y) \wedge Q(z,c) \wedge P(z) \wedge S(z,y)$$
$$Q(x,y) \wedge S(x,z) \rightarrow S(x,y)$$



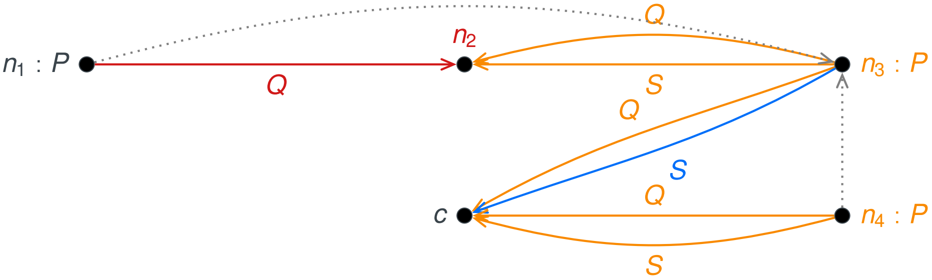
Compute Cores directly with Alternative Matches

$$\rightarrow \exists z.P(z)$$

$$P(x) \rightarrow \exists z.Q(x, z)$$

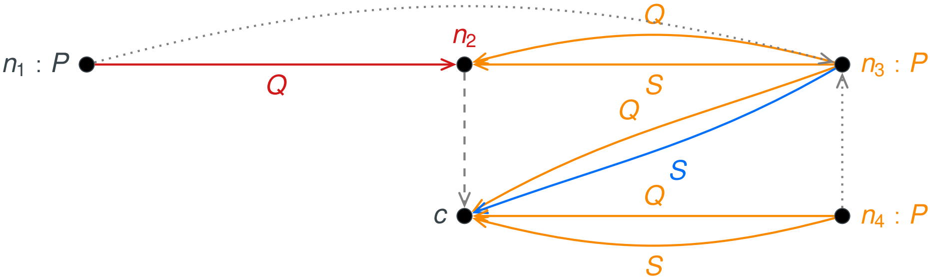
$$Q(x, y) \rightarrow \exists z.Q(z, y) \wedge Q(z, c) \wedge P(z) \wedge S(z, y)$$

$$Q(x, y) \wedge S(x, z) \rightarrow S(x, y)$$



Compute Cores directly with Alternative Matches

$$\rightarrow \exists z.P(z)$$
$$P(x) \rightarrow \exists z.Q(x,z)$$
$$Q(x,y) \rightarrow \exists z.Q(z,y) \wedge Q(z,c) \wedge P(z) \wedge S(z,y)$$
$$Q(x,y) \wedge S(x,z) \rightarrow S(x,y)$$



Problem: After remappings of nulls, other remappings may be necessary that are not captured by alternative matches.

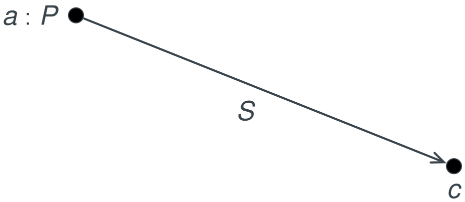
The Hybrid Chase

$$P(x) \rightarrow \exists z.Q(x,z) \qquad (R_1)$$

$$Q(x,y) \rightarrow \exists z.Q(z,y) \wedge Q(z,c) \wedge P(z) \wedge S(z,y) \qquad (R_2)$$

$$Q(x,y) \wedge S(x,z) \rightarrow S(x,y) \qquad (R_1)$$

The *hybrid chase* on a *relaxed restrained partitioning* is defined like the transfinite chase but uses the core chase in the last sequence.



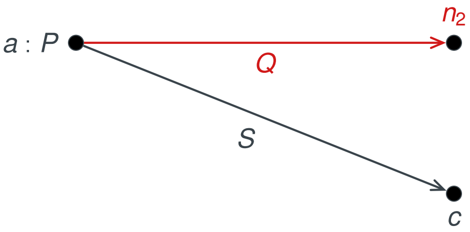
The Hybrid Chase

$$P(x) \rightarrow \exists z.Q(x,z) \quad (\mathbf{R_1})$$

$$Q(x,y) \rightarrow \exists z.Q(z,y) \wedge Q(z,c) \wedge P(z) \wedge S(z,y) \quad (R_2)$$

$$Q(x,y) \wedge S(x,z) \rightarrow S(x,y) \quad (\mathbf{R_1})$$

The *hybrid chase* on a *relaxed restrained partitioning* is defined like the transfinite chase but uses the core chase in the last sequence.



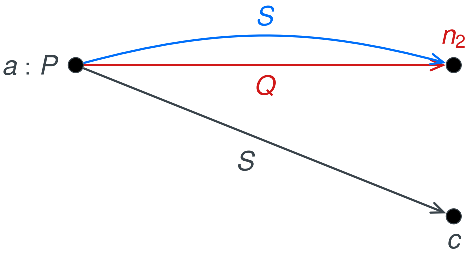
The Hybrid Chase

$$P(x) \rightarrow \exists z.Q(x,z) \quad (\mathbf{R_1})$$

$$Q(x,y) \rightarrow \exists z.Q(z,y) \wedge Q(z,c) \wedge P(z) \wedge S(z,y) \quad (R_2)$$

$$Q(x,y) \wedge S(x,z) \rightarrow S(x,y) \quad (\mathbf{R_1})$$

The *hybrid chase* on a *relaxed restrained partitioning* is defined like the transfinite chase but uses the core chase in the last sequence.



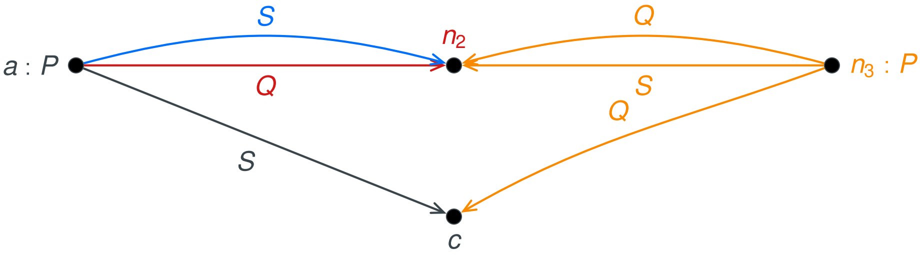
The Hybrid Chase

$$P(x) \rightarrow \exists z.Q(x,z) \qquad (\mathbf{R_1})$$

$$Q(x,y) \rightarrow \exists z.Q(z,y) \wedge Q(z,c) \wedge P(z) \wedge S(z,y) \qquad (\mathbf{R_2})$$

$$Q(x,y) \wedge S(x,z) \rightarrow S(x,y) \qquad (\mathbf{R_1})$$

The *hybrid chase* on a *relaxed restrained partitioning* is defined like the transfinite chase but uses the core chase in the last sequence.



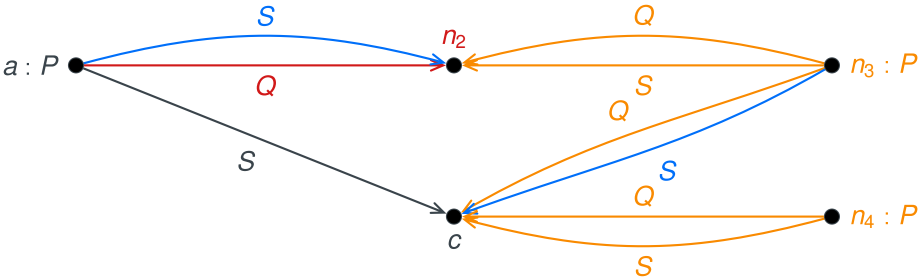
The Hybrid Chase

$$P(x) \rightarrow \exists z. Q(x, z) \quad (\mathbf{R_1})$$

$$Q(x, y) \rightarrow \exists z. Q(z, y) \land Q(z, c) \land P(z) \land S(z, y) \quad (\mathbf{R_2})$$

$$Q(x, y) \land S(x, z) \rightarrow S(x, y) \quad (\mathbf{R_1})$$

The *hybrid chase* on a *relaxed restrained partitioning* is defined like the transfinite chase but uses the core chase in the last sequence.



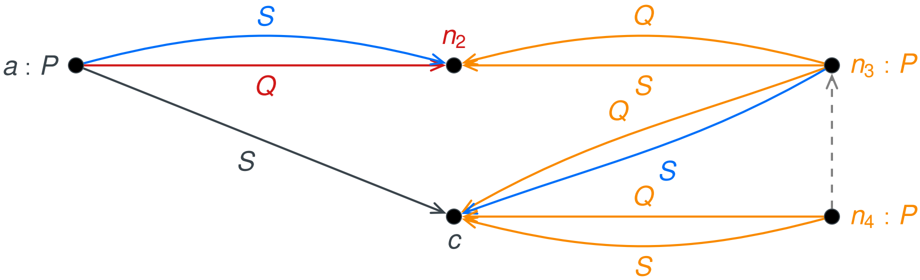
The Hybrid Chase

$$P(x) \to \exists z. Q(x, z) \qquad (\mathbf{R_1})$$

$$Q(x, y) \to \exists z. Q(z, y) \wedge Q(z, c) \wedge P(z) \wedge S(z, y) \qquad (\mathbf{R_2})$$

$$Q(x, y) \wedge S(x, z) \to S(x, y) \qquad (\mathbf{R_1})$$

The *hybrid chase* on a *relaxed restrained partitioning* is defined like the transfinite chase but uses the core chase in the last sequence.



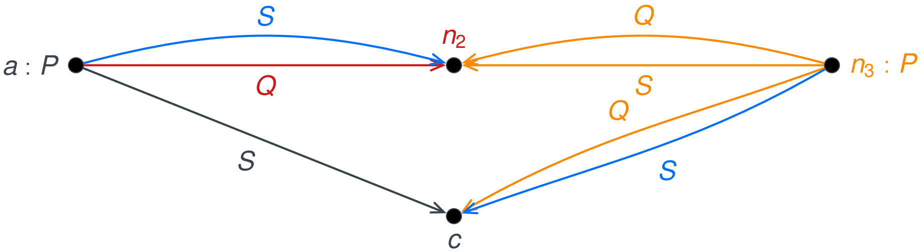
The Hybrid Chase

$$P(x) \to \exists z.Q(x,z) \quad (\mathbf{R_1})$$

$$Q(x,y) \to \exists z.Q(z,y) \wedge Q(z,c) \wedge P(z) \wedge S(z,y) \quad (\mathbf{R_2})$$

$$Q(x,y) \wedge S(x,z) \to S(x,y) \quad (\mathbf{R_1})$$

The *hybrid chase* on a *relaxed restrained partitioning* is defined like the transfinite chase but uses the core chase in the last sequence.



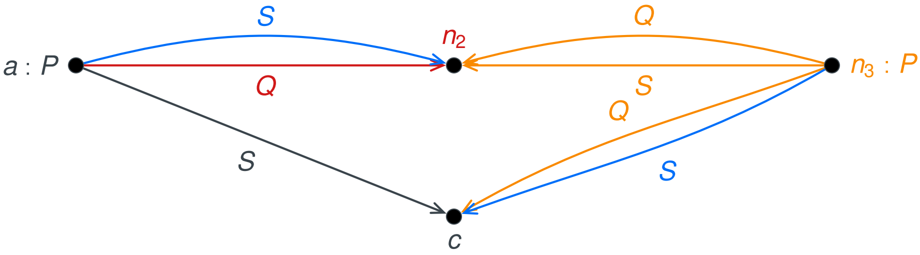
The Hybrid Chase

$$P(x) \to \exists z.Q(x,z) \qquad (\mathbf{R_1})$$

$$Q(x,y) \to \exists z.Q(z,y) \wedge Q(z,c) \wedge P(z) \wedge S(z,y) \qquad (\mathbf{R_2})$$

$$Q(x,y) \wedge S(x,z) \to S(x,y) \qquad (\mathbf{R_1})$$

The *hybrid chase* on a *relaxed restrained partitioning* is defined like the transfinite chase but uses the core chase in the last sequence.



Nulls that are introduced before the last sequence can be treated as constants.

Summary

Results:

- Restricted and core chase coincide for core-stratified rule sets.
- Conjecture: Slightly larger fragment of guarded rules for which $\mathrm{CT}_{\forall\forall}^{res}$ is decidable.
- Ideas for more efficient computation of universal core models for arbitrary rule sets.

Summary

Results:

- Restricted and core chase coincide for core-stratified rule sets.
- Conjecture: Slightly larger fragment of guarded rules for which $\mathrm{CT}_{\forall\forall}^{res}$ is decidable.
- Ideas for more efficient computation of universal core models for arbitrary rule sets.

Open Questions / Future Work:

- Is $\overline{\mathrm{AM}}_{\forall\exists}$ decidable for (single-head) guarded existential rules?
- Is $\mathrm{CT}_{\forall\exists}^{res}$ decidable for (single-head) guarded existential rules?

Summary

Results:

- Restricted and core chase coincide for core-stratified rule sets.
- Conjecture: Slightly larger fragment of guarded rules for which $\mathrm{CT}_{\forall\forall}^{res}$ is decidable.
- Ideas for more efficient computation of universal core models for arbitrary rule sets.

Open Questions / Future Work:

- Is $\overline{\mathrm{AM}}_{\forall\exists}$ decidable for (single-head) guarded existential rules?
- Is $\mathrm{CT}_{\forall\exists}^{res}$ decidable for (single-head) guarded existential rules?
- Verify decidability of $\mathrm{CT}_{\forall\forall}^{res}$ for guarded rules without strong restraining relations.
- Implement/Evaluate/Improve core computation heuristic and hybrid chase.

References I

(2010).
Proceedings of the 25th Annual IEEE Symposium on Logic in Computer Science, LICS 2010, 11-14 July 2010, Edinburgh, United Kingdom. IEEE Computer Society.

Abiteboul, S., Hull, R., and Vianu, V. (1995).
Foundations of Databases.
Addison-Wesley.

Baget, J., Leclère, M., Mugnier, M., and Salvat, E. (2011a).
On rules with existential variables: Walking the decidability line.
Artif. Intell., 175(9-10):1620–1654.

Baget, J., Mugnier, M., Rudolph, S., and Thomazo, M. (2011b).
Walking the complexity lines for generalized guarded existential rules.
In [Walsh, 2011], pages 712–717.

References II

Bárány, V., Gottlob, G., and Otto, M. (2010).
Querying the guarded fragment.
In [DBL, 2010], pages 1–10.

Barceló, P. and Calautti, M., editors (2019).
22nd International Conference on Database Theory, ICDT 2019, March 26-28, 2019, Lisbon, Portugal, volume 127 of *LIPIcs*. Schloss Dagstuhl - Leibniz-Zentrum für Informatik.

Bauslaugh, B. L. (1995).
Core-like properties of infinite graphs and structures.
Discret. Math., 138(1-3):101–111.

Beeri, C. and Vardi, M. Y. (1981).
The implication problem for data dependencies.
In [Even and Kariv, 1981], pages 73–85.

References III

Beeri, C. and Vardi, M. Y. (1984).
A proof procedure for data dependencies.
J. ACM, 31(4):718–741.

Benedikt, M., Konstantinidis, G., Mecca, G., Motik, B., Papotti, P., Santoro, D., and Tsamoura, E. (2017).
Benchmarking the chase.
In [Sallinger et al., 2017], pages 37–52.

Boutilier, C., editor (2009).
IJCAI 2009, Proceedings of the 21st International Joint Conference on Artificial Intelligence, Pasadena, California, USA, July 11-17, 2009.

Calautti, M., Gottlob, G., and Pieris, A. (2015).
Chase termination for guarded existential rules.
In [Milo and Calvanese, 2015], pages 91–103.

References IV

- Calvanese, D., Erdem, E., and Thielscher, M., editors (2020).
  Proceedings of the 17th International Conference on Principles of Knowledge Representation and Reasoning, KR 2020, Rhodes, Greece, September 12-18, 2020.
- Carral, D., Dragoste, I., and Krötzsch, M. (2017).
  Restricted chase (non)termination for existential rules with disjunctions.
  In [Sierra, 2017], pages 922–928.
- Carral, D., Krötzsch, M., Marx, M., Ozaki, A., and Rudolph, S. (2018).
  Preserving constraints with the stable chase.
  In [Kimelfeld and Amsterdamer, 2018], pages 12:1–12:19.
- Deutsch, A., Nash, A., and Remmel, J. B. (2008).
  The chase revisited.
  In [Lenzerini and Lembo, 2008], pages 149–158.

References V

Esparza, J., Fraigniaud, P., Husfeldt, T., and Koutsoupias, E., editors (2014).
Automata, Languages, and Programming - 41st International Colloquium, ICALP 2014, Copenhagen, Denmark, July 8-11, 2014, Proceedings, Part II, volume 8573 of *Lecture Notes in Computer Science*. Springer.

Even, S. and Kariv, O., editors (1981).
Automata, Languages and Programming, 8th Colloquium, Acre (Akko), Israel, July 13-17, 1981, Proceedings, volume 115 of *Lecture Notes in Computer Science*. Springer.

Fagin, R., Kolaitis, P. G., Miller, R. J., and Popa, L. (2005).
Data exchange: semantics and query answering.
Theor. Comput. Sci., 336(1):89–124.

References VI

Fagin, R., Kolaitis, P. G., and Popa, L. (2003).
Data exchange: getting to the core.
In [Neven et al., 2003], pages 90–101.

Gogacz, T. and Marcinkowski, J. (2014).
All-instances termination of chase is undecidable.
In [Esparza et al., 2014], pages 293–304.

Gogacz, T., Marcinkowski, J., and Pieris, A. (2020).
All-instances restricted chase termination.
In [Suciu et al., 2020], pages 245–258.

Gottlob, G. and Nash, A. (2008).
Efficient core computation in data exchange.
J. ACM, 55(2):9:1–9:49.

References VII

Grahne, G. and Onet, A. (2018).
Anatomy of the chase.
Fundam. Informaticae, 157(3):221–270.

Grau, B. C., Horrocks, I., Krötzsch, M., Kupke, C., Magka, D., Motik, B., and Wang, Z. (2013).
Acyclicity notions for existential rules and their application to query answering in ontologies.
J. Artif. Intell. Res., 47:741–808.

Hell, P. and Nesetril, J. (1992).
The core of a graph.
Discret. Math., 109(1-3):117–126.

References VIII

Kimelfeld, B. and Amsterdamer, Y., editors (2018).
21st International Conference on Database Theory, ICDT 2018, March 26-29, 2018, Vienna, Austria, volume 98 of *LIPIcs*. Schloss Dagstuhl - Leibniz-Zentrum für Informatik.

Krötzsch, M. (2020).
Computing cores for existential rules with the standard chase and ASP.
In [Calvanese et al., 2020], pages 603–613.

Krötzsch, M., Marx, M., and Rudolph, S. (2019).
The power of the terminating chase (invited talk).
In [Barceló and Calautti, 2019], pages 3:1–3:17.

Leclère, M., Mugnier, M., Thomazo, M., and Ulliana, F. (2019).
A single approach to decide chase termination on linear existential rules.
In [Barceló and Calautti, 2019], pages 18:1–18:19.

References IX

Lenzerini, M. and Lembo, D., editors (2008).
Proceedings of the Twenty-Seventh ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems, PODS 2008, June 9-11, 2008, Vancouver, BC, Canada. ACM.

Maier, D., Mendelzon, A. O., and Sagiv, Y. (1979).
Testing implications of data dependencies.
ACM Trans. Database Syst., 4(4):455–469.

Milo, T. and Calvanese, D., editors (2015).
Proceedings of the 34th ACM Symposium on Principles of Database Systems, PODS 2015, Melbourne, Victoria, Australia, May 31 - June 4, 2015. ACM.

Neven, F., Beeri, C., and Milo, T., editors (2003).
Proceedings of the Twenty-Second ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems, June 9-12, 2003, San Diego, CA, USA. ACM.

References X

Pichler, R. and Savenkov, V. (2009).
Demo: Data exchange modeling tool.
Proc. VLDB Endow., 2(2):1606–1609.

Sallinger, E., den Bussche, J. V., and Geerts, F., editors (2017).
Proceedings of the 36th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems, PODS 2017, Chicago, IL, USA, May 14-19, 2017. ACM.

Sierra, C., editor (2017).
Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017. ijcai.org.

Suciu, D., Tao, Y., and Wei, Z., editors (2020).
Proceedings of the 39th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems, PODS 2020, Portland, OR, USA, June 14-19, 2020. ACM.

References XI

Walsh, T., editor (2011).
IJCAI 2011, Proceedings of the 22nd International Joint Conference on Artificial Intelligence, Barcelona, Catalonia, Spain, July 16-22, 2011. IJCAI/AAAI.