Глава 8

Запросы к базам знаний

База знаний представляет собой совокупность утверждений о некоторой предметной области, собранную экспертами в этой области. Чтобы этими знаниями можно было пользоваться, необходим механизм извлечения информации из базы знаний. Для этого предназначены *запросы*. С простейшими видами запросов мы уже встречались. Так, мы можем спросить, верно ли, что в терминологии $\mathcal{T}$ концепт C всегда содержится в концепте D (то есть: $\mathcal{T} \models C \sqsubseteq D$); верно ли, что в RBox-е $\mathcal{R}$ роль R содержится в роли S (то есть: $\mathcal{R} \models R \sqsubseteq S$); верно ли, что в базе знаний $\mathcal{B}$ индивид a принадлежит концепту C (то есть: $\mathcal{B} \models a\colon C$) или что в $\mathcal{B}$ пара индивидов a, b связана ролью R (то есть: $\mathcal{B} \models aRb$). Вводимые ниже запросы являются обобщением вышеперечисленных.

Пусть задана сигнатура $\Sigma = (\mathsf{CN}, \mathsf{RN}, \mathsf{IN})$, то есть множества атомарных концептов, ролей и индивидов. Для формулировки запросов введем новый сорт символов — счетное множество индивидных переменных $Var = \{x_0, x_1, \ldots\}$. *Атомы* (или атомарные запросы) — это выражения вида $u\colon C$ и uRv, где C — концепт, R — роль, u, v — индивиды (из IN) или переменные (из Var).

Определение 8.1. *Конъюнктивный запрос* — это выражение вида $\exists \vec{y}\,(t_1 \wedge \ldots \wedge t_k)$, где t_i — атомы, $\vec{y} = (y_1, \ldots, y_m)$ — список некоторых переменных, входящих в t_i. Переменные y_j называются *связными*; остальные переменные, входящие в t_i, называются *свободными*. Если $\vec{x} = (x_1, \ldots, x_n)$ — список свободных переменных запроса q, то мы записываем этот запрос как $q(\vec{x})$.

Число свободных переменных запроса q называется его *размерностью* и обозначается $ar(q)$. Запрос размерности n называется также *n-местным* запросом; 0-местные запросы называются *булевыми*.

В дальнейшем конъюнктивные запросы мы будем для краткости называть просто запросами. Поскольку концепты можно считать формулами первого порядка (при стандартном переводе в логику предикатов, см. раздел 1.1), то запросы тоже можно считать формулами первого порядка специального вида. При таком рассмотрении n-местный запрос является формулой с n свободными переменными. Поэтому в конкретной интерпретации $\mathcal{I} = (\Delta, \cdot^{\mathcal{I}})$ для всякого булева запроса q определено, *истинен* ли он в данной интерпретации: $\mathcal{I} \models q$, а всякий n-местный запрос $q(\vec{x})$ при $n \geqslant 1$ задает n-местный предикат:

$$q^{\mathcal{I}} \;=\; \{\, \vec{e} \in \Delta^n \;\mid\; \mathcal{I} \models q(\vec{e}) \,\}.$$

Пусть дан n-местный запрос $q(\vec{x})$, набор из n индивидов $\vec{a}$ и база знаний $\mathcal{B}$.

Определение 8.2. Для $n = 0$: булев запрос q *истинен* в базе знаний $\mathcal{B}$, если $\mathcal{B} \models q$, то есть для любой интерпретации $\mathcal{I}$ из $\mathcal{I} \models \mathcal{B}$ следует $\mathcal{I} \models q$.

Для $n \geqslant 1$: набор индивидов $\vec{a}$ называется *ответом* на запрос $q(\vec{x})$ к базе знаний $\mathcal{B}$, если $\mathcal{B} \models q(\vec{a})$. Другими словами, если для любой интерпретации $\mathcal{I}$ из $\mathcal{I} \models \mathcal{B}$ следует $\mathcal{I} \models q(\vec{a})$.

Множество ответов на запрос $q(\vec{x})$ к базе знаний $\mathcal{B}$ обозначим через $q[\mathcal{B}] = \{\, \vec{a} \in \mathsf{IN}^n \mid \mathcal{B} \models q(\vec{a}) \,\}$.

Лемма 8.1. *Множество ответов на запрос к базе знаний всегда конечно.*

Доказательство. Если $|\mathsf{IN}| = N$ и $ar(q) = n$, то по определению $|q[\mathcal{B}]| \leqslant N^n$. ◀

Во всякой модели $\mathcal{I} \models \mathcal{B}$ множество $q^{\mathcal{I}}$ есть некоторое подмножество кортежей Δ^n. Из них «имена» имеют лишь кортежи, принадлежащие $(\mathsf{IN}^n)^{\mathcal{I}}$, значит, они составляют некоторое конечное подмножество Δ^n. Множество ответов на n-местный запрос q тоже интерпретируется как конечное подмножество Δ^n. Что можно сказать о связи этих двух множеств?

Лемма 8.2. *Если $\mathcal{I} \models \mathcal{B}$, то $q[\mathcal{B}]^{\mathcal{I}} \subseteq q^{\mathcal{I}} \cap (\mathsf{IN}^n)^{\mathcal{I}}$.*

Доказательство. Включение $q[\mathcal{B}]^{\mathcal{I}} \subseteq (\mathsf{IN}^n)^{\mathcal{I}}$ очевидно. Докажем, что $q[\mathcal{B}]^{\mathcal{I}} \subseteq q^{\mathcal{I}}$. Пусть $\vec{e} \in q[\mathcal{B}]^{\mathcal{I}}$, то есть $\vec{e} = \vec{a}^{\mathcal{I}}$ для некоторого набора индивидов $\vec{a} \in q[\mathcal{B}]$. Последнее означает, что $\mathcal{B} \models q(\vec{a})$. Но поскольку $\mathcal{I} \models \mathcal{B}$, то мы заключаем, что $\mathcal{I} \models q(\vec{a})$, то есть $\mathcal{I} \models q(\vec{e})$ и значит $\vec{e} \in q^{\mathcal{I}}$. ◀

Естественно задать вопрос, будет ли в этой лемме всегда верно равенство? Оказывается, нет.

Пример 8.1. Рассмотрим базу знаний $\mathcal{B} = \langle \varnothing, \mathcal{A} \rangle$, где ABox $\mathcal{A} = \{\, a\, \mathsf{hasChild}\, b,\ b\colon \mathsf{Man},\ a\colon \exists \mathsf{hasChild}.\neg \mathsf{Man} \,\}$.
Сравним следующие два запроса: $q(x) = a\, \mathsf{hasChild}\, x$ и $p(x) = \ x\colon \mathsf{Man}$.
Очевидно, что $q[\mathcal{B}] = p[\mathcal{B}] = \{b\}$. Теперь рассмотрим следующую интерпретацию $\mathcal{I} = (\Delta, \cdot^{\mathcal{I}})$, где

$$\begin{array}{l} \Delta = \{\, \mathsf{Peter},\ \mathsf{John},\ \mathsf{Mary} \,\}, \\ \mathsf{Man}^{\mathcal{I}} = \{\, \mathsf{Peter},\ \mathsf{John} \,\}, \quad a^{\mathcal{I}} = \mathsf{Peter}, \quad b^{\mathcal{I}} = \mathsf{John} \\ \mathsf{hasChild}^{\mathcal{I}} = \{\, \langle \mathsf{Peter},\ \mathsf{John} \rangle,\ \langle \mathsf{Peter},\ \mathsf{Mary} \rangle \,\}. \end{array}$$

Она является моделью базы знаний $\mathcal{B}$. В ней имеем: $\mathsf{IN}^{\mathcal{I}} = \{\, a^{\mathcal{I}},\, b^{\mathcal{I}} \,\} = \{\, \mathsf{Peter},\ \mathsf{John} \,\}$.
Интерпретация запросов такова: $q^{\mathcal{I}} = \{\, \mathsf{John},\ \mathsf{Mary} \,\}$ и $p^{\mathcal{I}} = \{\, \mathsf{Peter},\ \mathsf{John} \,\}$. Тогда:

$$\begin{array}{l} q^{\mathcal{I}} \cap \mathsf{IN}^{\mathcal{I}} = \{\, \mathsf{John},\ \mathsf{Mary} \,\} \cap \{\, \mathsf{Peter},\ \mathsf{John} \,\} = \{\, \mathsf{John} \,\} \\ p^{\mathcal{I}} \cap \mathsf{IN}^{\mathcal{I}} = \{\, \mathsf{Peter},\ \mathsf{John} \,\} \cap \{\, \mathsf{Peter},\ \mathsf{John} \,\} = \{\, \mathsf{Peter},\ \mathsf{John} \,\}. \end{array}$$

Таким образом, имеет место равенство $q[\mathcal{B}]^{\mathcal{I}} = \{\, \mathsf{John} \,\} = q^{\mathcal{I}} \cap (\mathsf{IN}^n)^{\mathcal{I}}$, но в то же время строгое включение $p[\mathcal{B}]^{\mathcal{I}} = \{\, \mathsf{John} \,\} \subset \{\, \mathsf{Peter},\ \mathsf{John} \,\} = p^{\mathcal{I}} \cap (\mathsf{IN}^n)^{\mathcal{I}}$.

8.1 Импликации между запросами

При фиксированной терминологии $\mathcal{T}$ на концептах определено отношение включения (или импликации): $\mathcal{T} \models C \sqsubseteq D$. Аналогично, при фиксированном RBox $\mathcal{R}$ определено отношение включения между ролями: $\mathcal{R} \models R \sqsubseteq S$. Мы распространим отношение включения на произвольные запросы: $\mathcal{T} \models q \sqsubseteq p$, где $\mathcal{T}$ — произвольная терминология (то есть TBox или объединение TBox и RBox, если последние имеются в логике), $q(\vec{x})$ и $p(\vec{x})$ — запросы одинаковой размерности (и более того, имеющие один и тот же набор свободных переменных).

Существует два эквивалентных способа определить включение между запросами (которое мы будем записывать как $\mathcal{T} \models q \sqsubseteq p$). Первый способ обобщает соответствующее определение для концептов.

Определение 8.3 (Включение между запросами, Вариант 1). Запрос $q(\vec{x})$ *содержится* в запросе $p(\vec{x})$ относительно $\mathcal{T}$, если для любой модели $\mathcal{I} \models \mathcal{T}$ имеет место включение $q^{\mathcal{I}} \subseteq p^{\mathcal{I}}$.

Фактически данное определение является стандартным определением логического следования из теории: если $\mathcal{T}', q', p'$ есть переводы $\mathcal{T}, q, p$ в логику первого порядка, то данное выше определение $\mathcal{T} \models q \sqsubseteq p$ равносильно следующему: $\mathcal{T}' \models \forall \vec{x}\, (q(\vec{x}) \to p(\vec{x}))$.

Второе определение включения между запросами основано на том, что запросы предназначены для извлечения информации из баз знаний. Согласно второму определению, запрос q содержится в запросе p относительно $\mathcal{T}$, если ответы на q *всегда* содержатся среди ответов на p. Здесь «всегда» означает: в любой базе знаний, основанной на терминологии $\mathcal{T}$, то есть имеющей вид $\langle \mathcal{T}, \mathcal{A} \rangle$.

Определение 8.4 (Включение между запросами, Вариант 2). Запрос $q(\vec{x})$ *содержится* в запросе $p(\vec{x})$ относительно $\mathcal{T}$, если для любого ABox $\mathcal{A}$, обозначая $\mathcal{B} = \langle \mathcal{T}, \mathcal{A} \rangle$, имеем: $q[\mathcal{B}] \subseteq p[\mathcal{B}]$.

Итак, оба варианта определения можно записать следующим образом:
Вариант 1: $\mathcal{T} \models q \sqsubseteq p$, если из $\mathcal{I} \models q(\vec{e})$ следует $\mathcal{I} \models p(\vec{e})$ для любой модели $\mathcal{I} \models \mathcal{T}$ и любого $\vec{e} \in \Delta^n$.
Вариант 2: $\mathcal{T} \models q \sqsubseteq p$, если из $\mathcal{B} \models q(\vec{a})$ следует $\mathcal{B} \models p(\vec{a})$ для любого ABox $\mathcal{A}$ и набора $\vec{a}$, где $\mathcal{B} = \langle \mathcal{T}, \mathcal{A} \rangle$.

Для булевых запросов эти определения нужно слегка изменить:
Вариант 1: $\mathcal{T} \models q \sqsubseteq p$, если из $\mathcal{I} \models q$ следует $\mathcal{I} \models p$ для любой модели $\mathcal{I} \models \mathcal{T}$.
Вариант 2: $\mathcal{T} \models q \sqsubseteq p$, если из $\mathcal{B} \models q$ следует $\mathcal{B} \models p$ для любого ABox $\mathcal{A}$, где $\mathcal{B} = \langle \mathcal{T}, \mathcal{A} \rangle$.

Теорема 8.3. *Два определения включения запросов эквивалентны друг другу.*

Прежде чем доказывать теорему, введем вспомогательную конструкцию. Пусть запрос q имеет вид: $q(\vec{x}) = \exists \vec{y} \bigwedge_{i=1}^{k} t_i(\vec{x}, \vec{y})$, где каждый t_i есть атом. Введем свежие (не входящие в $\mathcal{T}, q, p$) наборы индивидов $\vec{a}, \vec{c}$ в количестве $|\vec{a}| = |\vec{x}|$ и $|\vec{c}| = |\vec{y}|$. Построим так называемый *канонический ABox* для запроса q:

$$\mathcal{A}_q \ := \ \{\, t_i(\vec{a}, \vec{c}) \ |\ 1 \leqslant i \leqslant k \,\}.$$

Обозначим также $\mathcal{B}_q = \langle \mathcal{T}, \mathcal{A}_q \rangle$. Тогда теорема 8.3 вытекает из следующей леммы.

Лемма 8.4. *Следующие три условия эквивалентны:*

(a) $\mathcal{T} \models q \sqsubseteq p$ *согласно определению 1;*
(b) $\mathcal{T} \models q \sqsubseteq p$ *согласно определению 2;*
(c) $\mathcal{B}_q \models p(\vec{a})$.

Доказательство. Пусть дана терминология $\mathcal{T}$ и два запроса $q(\vec{x})$ и $p(\vec{x})$ размерности $ar(q) = ar(p) = n \geqslant 1$ (для $n = 0$ доказательство аналогично; проведите его самостоятельно).

(a) $\Rightarrow$ (b)

Допустим, что $\mathcal{T} \models q \sqsubseteq p$ согласно определению 1. Чтобы доказать, что $\mathcal{T} \models q \sqsubseteq p$ согласно определению 2, рассмотрим произвольный ABox $\mathcal{A}$, обозначим $\mathcal{B} = \langle \mathcal{T}, \mathcal{A} \rangle$, и возьмем любой набор из n индивидов $\vec{a}$, такой что $\mathcal{B} \models q(\vec{a})$. Требуется доказать, что $\mathcal{B} \models p(\vec{a})$. Для этого возьмем любую модель $\mathcal{I} \models \mathcal{B}$ и покажем, что $\mathcal{I} \models p(\vec{a})$. Обозначим $\vec{e} = \vec{a}^{\mathcal{I}} \in \Delta^n$.

Так как $\mathcal{I} \models \mathcal{B}$ и $\mathcal{B} \models q(\vec{a})$, то $\mathcal{I} \models q(\vec{a})$, то есть $\mathcal{I} \models q(\vec{e})$. Согласно определению 1, из этого следует $\mathcal{I} \models p(\vec{e})$. Следовательно, $\mathcal{I} \models p(\vec{a})$, что и требовалось.

(b) $\Rightarrow$ (c)

Очевидно, что $\mathcal{A}_q \models t_i(\vec{a}, \vec{c})$, поскольку каждый факт $t_i(\vec{a}, \vec{c})$ содержится в $\mathcal{A}_q$. Значит, $\mathcal{A}_q \models \bigwedge_{i=1}^{k} t_i(\vec{a}, \vec{c})$ и по правилам логики $\mathcal{A}_q \models \exists \vec{y} \bigwedge_{i=1}^{k} t_i(\vec{a}, \vec{y})$. Таким образом, $\mathcal{B}_q \models q(\vec{a})$.

Теперь допустим, что $\mathcal{T} \models q \sqsubseteq p$ согласно определению 2. Это значит, что для любого ABox (в частности, для $\mathcal{A}_q$) и любого набора из n индивидов (в частности, для $\vec{a}$) если $\mathcal{B}_q \models q(\vec{a})$ (а это верно по доказанному выше), то $\mathcal{B}_q \models p(\vec{a})$, что и требовалось.

(c) $\Rightarrow$ (a)

Пусть $\mathcal{B}_q \models p(\vec{a})$. Чтобы доказать, что $\mathcal{T} \models q \sqsubseteq p$ согласно определению 1, возьмем любую модель $\mathcal{I} \models \mathcal{T}$ и любой набор элементов $\vec{e} \in \Delta^n$, такой что $\mathcal{I} \models q(\vec{e})$, и покажем, что $\mathcal{I} \models p(\vec{e})$.

Так как $\mathcal{I} \models q(\vec{e})$, то существует набор элементов $\vec{o} \in \Delta^m$, такой что $\mathcal{I} \models t_i(\vec{e}, \vec{o})$ для всех i. Расширим интерпретацию $\mathcal{I}$ на новые индивиды $\vec{a}$ и $\vec{c}$, положив $\vec{a}^{\mathcal{I}} := \vec{e}$ и $\vec{c}^{\mathcal{I}} := \vec{o}$. Тогда $\mathcal{I} \models t_i(\vec{a}, \vec{c})$ для всех i, и тем самым $\mathcal{I} \models \mathcal{B}_q$. Но поскольку $\mathcal{B}_q \models p(\vec{a})$, то $\mathcal{I} \models p(\vec{a})$, то есть $\mathcal{I} \models p(\vec{e})$, что и требовалось. ◀

Обратим внимание, что в доказательстве импликации **(a) $\Rightarrow$ (b)** нигде не использовалось, что $q(\vec{x})$ и $p(\vec{x})$ являются конъюнктивными запросами; они могли быть любыми формулами первого (и не только) порядка. В доказательстве обратной импликации существенно использовалось, что $q(\vec{x})$ является конъюнктивным запросом (при построении канонического ABox), в то время как $p(\vec{x})$ здесь снова могла быть любой формулой.

Упражнение 8.1. Докажите, что следующее определение вложения запросов равносильно двум рассмотренным выше:

$\mathcal{T} \models q \sqsubseteq p$, если для любой базы знаний $\mathcal{B} = \langle \mathcal{T}', \mathcal{A} \rangle$, где $\mathcal{T}' \supseteq \mathcal{T}$, имеет место $q[\mathcal{B}] \subseteq p[\mathcal{B}]$.

Потребность проверять вложение запросов возникает, в частности, при оптимизации запросов. По исходному (сложному) запросу q, принимая во внимание заданную терминологию $\mathcal{T}$, строят (более простой) запрос q' и затем проверяют их эквивалентность: $\mathcal{T} \models q \equiv q'$ путем проверки двух импликаций.

8.2 Алгоритмические проблемы для запросов

Основными алгоритмическими проблемами, которые требуется решать на практике и для которых изучается вопрос о разрешимости и сложности, являются следующие:

(QA) **Query answering** — проблема построения множества ответов на запрос:

Дано: база знаний $\mathcal{B}$ и запрос $q(\vec{x})$ размерности $|\vec{x}| = n \geqslant 1$.
Найти: множество ответов $q[\mathcal{B}]$ на запрос q к базе знаний $\mathcal{B}$.

(QE) **Query entailment** — проблема нахождения ответа на булев запрос:

Дано: база знаний $\mathcal{B}$ и *булев* запрос q.
Узнать: верно ли, что запрос q следует из $\mathcal{B}$, то есть $\mathcal{B} \models q$?

(QC) **Query containment** или **Query subsumption** — проблема включения одного запроса в другой:

Дано: терминология $\mathcal{T}$ и два запроса q и p одной размерности.
Узнать: содержится ли запрос q в запросе p относительно $\mathcal{T}$, то есть $\mathcal{T} \models q \sqsubseteq p$?

Последние две проблемы являются так называемыми *проблемами разрешения*, то есть проблемами с ответом типа 'да/нет'; первая же проблема таковой не является. Однако первая проблема сводится к многократному решению второй. Действительно, чтобы найти множество ответов $q[\mathcal{B}]$, достаточно перебрать все наборы индивидов $\vec{a} \in \mathbb{N}^n$ и для каждого проверить, верно ли $\mathcal{B} \models q(\vec{a})$, где $q(\vec{a})$ является булевым запросом. Поэтому обычно для различных логик исследуют лишь вторую и третью проблемы. Более того, как показывает следующая лемма, они сводятся друг к другу, так что достаточно устанавливать разрешимость (и сложность) лишь одной из них.

Лемма 8.5. *Проблемы* (QE) *и* (QC) *сводятся друг к другу (за полиномиальное время).*

Доказательство.
• (QE) $\leqslant$ (QC): Пусть дана база знаний $\mathcal{B} = \langle \mathcal{T}, \mathcal{A} \rangle$ и булев запрос q. Введем конъюнкцию всех фактов из ABox $\mathcal{A}$: $p := \bigwedge \mathcal{A} = \bigwedge_{\alpha \in \mathcal{A}} \alpha$. Очевидно, что p — булев конъюнктивный запрос. Тогда проблема (QE) сводится к (QC) посредством следующей эквивалентности:

$$\mathcal{B} \models q \iff \mathcal{T} \models p \sqsubseteq q.$$

Она следует из теоремы дедукции для логики предикатов: $\Gamma, A \models B \iff \Gamma \models A \to B$.

• (QE) $\geqslant$ (QC): Пусть дана терминология $\mathcal{T}$ и два запроса $q(\vec{x})$ и $p(\vec{x})$ одинаковой размерности. Построим для q канонический ABox $\mathcal{A}_q$ и обозначим $\mathcal{B}_q = \langle \mathcal{T}, \mathcal{A}_q \rangle$. Тогда проблема (QC) сводится к (QE) посредством следующей эквивалентности:

$$\mathcal{T} \models q \sqsubseteq p \iff \mathcal{B}_q \models p(\vec{a}).$$

Она была доказана в лемме 8.4 ◄

Теорема 8.6. *Из конъюнктивных запросов устранимы константы и конъюнкты вида $y\colon C$.*

Тем самым проблема (QE) *для произвольных булевых запросов сводится к той же проблеме для запросов вида $\exists \vec{y}\, Q(\vec{y})$, где $Q(\vec{y})$ состоит лишь из конъюнктов вида $y_i R y_j$.*

Доказательство. Пусть дана база знаний $\mathcal{B}$ и булев запрос q.

Устранение констант. Пусть $\vec{a} = (a_1, \ldots, a_n)$ — константы, входящие в q, тем самым $q = \exists \vec{y}\, Q(\vec{y}, \vec{a})$. Тогда заменим в q каждую константу a_i на свежую переменную z_i, добавим конъюнкты $z_i\colon A_i$, где $A_1, \ldots, A_n$ — свежие атомарные концепты, и навесим кванторы по переменным $\vec{z}$. В результате получим запрос, не содержащий констант:

$$q' = \exists \vec{y} \exists \vec{z} \big(Q(\vec{y}, \vec{z}) \wedge z_1\colon A_1 \wedge \ldots \wedge z_n\colon A_n \big).$$

Рассмотрим базу знаний $\mathcal{B}' = \mathcal{B} \cup \{a_1\colon A_1, \ldots, a_n\colon A_n\}$. Легко доказать следующую эквивалентность:

$$\mathcal{B} \models q \iff \mathcal{B}' \models q'.$$

Устранение конъюнктов вида $y\colon C$. Пусть запрос имеет вид $q = \exists \vec{y} \big(Q(\vec{y}) \wedge y_1\colon C_1 \wedge \ldots \wedge y_n\colon C_n \big)$, где $Q(\vec{y})$ состоит из конъюнктов вида $y_i R y_j$. Построим сначала запрос q', заменив в q каждый конъюнкт $y_i\colon C_i$ на $y_i\colon \exists R_i.\top$, где $R_1, \ldots, R_n$ — свежие роли. Рассмотрим базу знаний $\mathcal{B}' = \mathcal{B} \cup \{C_i \equiv \exists R_i.\top \mid 1 \leqslant i \leqslant n\}$. Легко доказать следующую эквивалентность:

$$\mathcal{B} \models q \iff \mathcal{B}' \models q'.$$

Поскольку $y_i\colon \exists R_i.\top$ эквивалентно $\exists z_i\, (y_i R z_i)$, то q' эквивалентен запросу

$$q'' = \exists \vec{y} \exists \vec{z} \big(Q(\vec{y}) \wedge y_1 R_1 z_1 \wedge \ldots \wedge y_n R_n z_n \big),$$

имеющему требуемый вид. □

8.3 Сложность проблемы отвечания на запросы

Обзор...

8.4 Запросы с неравенством неразрешимы

Вспомним, что запросы строятся из атомов вида $u{:}C$ и uRv. Что еще можно было бы добавить в синтаксис запросов? Очевидно, что равенства $x = y$ и $x = a$ (где x, y — переменные, a — индивид) не дают ничего нового, поскольку $\exists x\,(\varphi(x) \wedge x = a)$ эквивалентно $\varphi(a)$, а также $\exists x, y\,(\varphi(x, y) \wedge x = y)$ эквивалентно $\exists x\,\varphi(x, x)$. Можно попробовать добавить неравенство: $x \neq y$. Однако в этом случае проблема (QE), то есть $\mathcal{B} \models q$, оказывается неразрешимой даже для логики $\mathcal{ALC}$, что мы и покажем в этом разделе.

Будем рассматривать *запросы с неравенством*, имеющие вид $q(\vec{x}) = \exists \vec{y} \bigwedge_{i=1}^{k} t_i(\vec{x}, \vec{y})$, в которых каждый атом t_i имеет вид $u{:}C$, uRv, $u \neq v$, где u, v — переменные либо индивиды.

Теорема 8.7. *Проблема нахождения ответа на запросы с неравенством неразрешима для логики $\mathcal{ALC}$.*

Доказательство. Сведём проблему домино (см. раздел 14.2) к нашей проблеме (точнее, к ее дополнению).

Пусть дана некоторая система домино $\mathcal{D} = (D, H, V)$, где $D = \{d_1, \ldots, d_n\}$, $H, V \subseteq D \times D$. Введем атомарные концепты $D_1, \ldots, D_n$ и роли R, S. Построим базу знаний $\mathcal{B}_\mathcal{D} = \langle \mathcal{T}_\mathcal{D}, \varnothing \rangle$, где терминология $\mathcal{T}_\mathcal{D}$ состоит из следующих аксиом (для всех $1 \leqslant i, j \leqslant n$):

(1) $\top \sqsubseteq \exists R$	**(3)** $\top \sqsubseteq D_1 \sqcup \ldots \sqcup D_n$	**(5)** $D_i \sqsubseteq \forall R. \bigsqcup\limits_{j:\langle d_i, d_j \rangle \in H} D_j$
(2) $\top \sqsubseteq \exists S$	**(4)** $D_i \sqcap D_j \sqsubseteq \bot$	**(6)** $D_i \sqsubseteq \forall S. \bigsqcup\limits_{j:\langle d_i, d_j \rangle \in V} D_j$

Возьмем следующий булев запрос: $\boxed{q = \exists x, y, y', z, z' \bigl(xRy \wedge ySz \wedge xSy' \wedge y'Rz' \wedge z \neq z'\bigr)}$

Для завершения доказательства теоремы нам осталось доказать следующую эквивалентность.

Лемма 8.7.1. *Системой $\mathcal{D}$ можно замостить сеть* $\mathbb{N}\times\mathbb{N} \quad\Longleftrightarrow\quad \mathcal{T}_\mathcal{D} \not\models q$.

Прежде чем доказывать лемму, переформулируем правую часть эквивалентности. Утверждение $\mathcal{T}_\mathcal{D} \not\models q$ означает, что существует модель $\mathcal{I} \models \mathcal{T}_\mathcal{D}$, такая что $\mathcal{I} \not\models q$, то есть $\mathcal{I} \models \neg q$. Формула $\neg q$ эквивалентна следующей:

$$\forall x, y, y', z, z' \bigl(xRy \wedge ySz \wedge xSy' \wedge y'Rz' \to z = z'\bigr).$$

Таким образом, $\neg q$ означает, что пойдя из некоторой точки x сначала «вправо» (по R), а затем «вверх» (по S), мы придем в ту же точку, как если бы мы пошли сначала «вверх», а затем «вправо». Эта формула поможет «построить» сетку, подобную $\mathbb{N}\times\mathbb{N}$. Теперь докажем лемму.

($\Rightarrow$) Пусть существует замощение $t{:}\, \mathbb{N}\times\mathbb{N} \to D$ системой $\mathcal{D}$ сети $\mathbb{N}\times\mathbb{N}$. Строим модель $\mathcal{I} = (\Delta, \cdot^\mathcal{I})$ следующим образом: $\Delta = \mathbb{N}\times\mathbb{N}$; полагаем $\langle x, y\rangle R^\mathcal{I} \langle x{+}1, y\rangle$ и $\langle x, y\rangle S^\mathcal{I} \langle x, y{+}1\rangle$ для всех $x, y \in \mathbb{N}$; наконец, $\langle x, y\rangle \in D_i^\mathcal{I} \Leftrightarrow t(x, y) = d_i$, для всех $1 \leqslant i \leqslant n$. Легко проверить, что $\mathcal{I} \models \mathcal{T}_\mathcal{D}$ и $\mathcal{I} \models \neg q$.

($\Leftarrow$) Пусть $\mathcal{I} \models \mathcal{T}_\mathcal{D}$ и $\mathcal{I} \models \neg q$. Покажем, как извлечь из модели $\mathcal{I}$ замощение системой $\mathcal{D}$ сети $\mathbb{N}\times\mathbb{N}$. Имеем $\exists e_{00} \in \Delta$. По аксиомам **(1)** и **(2)**, истинным в $\mathcal{I}$, существуют точки $e_{10}, e_{11}, e_{01}, e'_{11} \in \Delta$, соединенные «ребрами» следующим образом: $e_{00} R^\mathcal{I} e_{10} S^\mathcal{I} e_{11}$ и $e_{00} S^\mathcal{I} e_{01} R^\mathcal{I} e'_{11}$. По формуле $\neg q$, верной в $\mathcal{I}$, мы заключаем: $e_{11} = e'_{11}$. Продолжая эти рассуждения по индукции, мы находим в Δ семейство (не обязательно различных) точек $\{e_{xy} \mid x, y \in \mathbb{N}\}$, соединённых «ребрами» $R^\mathcal{I}$ и $S^\mathcal{I}$ подобно $\mathbb{N}\times\mathbb{N}$.

Согласно аксиомам **(3)** и **(4)**, каждая точка e_{xy} принадлежит ровно одному из концептов $D_i^\mathcal{I}$. Поэтому, положив $t(x, y) = d_i \Leftrightarrow e_{xy} \in D_i^\mathcal{I}$, мы получим корректно определенную функцию $t{:}\, \mathbb{N}\times\mathbb{N} \to D$. Ввиду истинности в $\mathcal{I}$ аксиом **(5)** и **(6)**, функция t является замощением, что и требовалось. ◀