

# Нормализация вычислений

---



Теория формальных языков  
*2021 г.*



## Алг. типы данных как TRS

Отношение одношаговой редукции  $\rightarrow_{\mathcal{T}}$ , определяемое  $\mathcal{T}$  – это наименьшее отношение на термах такое, что  $M[v := \sigma(L)] \rightarrow_{\mathcal{T}} M[v := \sigma(R)]$ , где  $x$  входит в терм  $M$  однажды, и  $L \rightarrow R$  — правило переписывания  $\mathcal{T}$ .  
Многошаговая редукция:  $M \rightarrow\!\!\rightarrow N$ .

В общем случае редукция требует сделать унификацию левой части правила,  $L$ , с некоторым подтермом терма  $M$ . Пример:  
 $A(x, S(y)) \rightarrow S(A(x, y))$  может применяться к терму  $A(A(Z, S(Z)), S(S(Z)))$  двумя разными способами: через подстановку  $x := A(Z, S(Z))$ ,  $y := S(Z)$  и через  $x := Z$ ,  $y := Z$ .

Пусть  $\mathcal{T}$  — конфлюэнтная TRS. Тогда  
 $\mathcal{T} \vdash M = N \Leftrightarrow M \rightarrow\!\!\rightarrow \circ \leftarrow\!\!\leftarrow N$ .



## Виды конфлюэнтности

- (Глобальная) конфлюэнтность:  
$$N_1 \leftarrow M \rightarrow N_2 \Rightarrow N_1 \rightarrow \circ \leftarrow N_2.$$
- Локальная конфлюэнтность:  
$$N_1 \leftarrow M \rightarrow N_2 \Rightarrow N_1 \rightarrow \circ \leftarrow N_2.$$



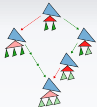
## Виды конфлюэнтности

- (Глобальная) конфлюэнтность:  
$$N_1 \leftarrow M \rightarrow N_2 \Rightarrow N_1 \rightarrow \circ \leftarrow N_2.$$
- Локальная конфлюэнтность:  
$$N_1 \leftarrow M \rightarrow N_2 \Rightarrow N_1 \rightarrow \circ \leftarrow N_2.$$

Рассмотрим следующую TRS:

$\infty \rightarrow S(\infty) \quad \text{Eq?}(x, x) \rightarrow \text{True} \quad \text{Eq?}(x, S(x)) \rightarrow \text{False}$

Она локально конфлюэнтна (проверьте), но не является конфлюэнтной.



## Виды конфлюэнтности

- (Глобальная) конфлюэнтность:  
$$N_1 \leftarrow M \rightarrow N_2 \Rightarrow N_1 \rightarrow \circ \leftarrow N_2.$$
- Локальная конфлюэнтность:  
$$N_1 \leftarrow M \rightarrow N_2 \Rightarrow N_1 \rightarrow \circ \leftarrow N_2.$$

Рассмотрим следующую TRS:

$$\infty \rightarrow S(\infty) \quad \text{Eq?}(x, x) \rightarrow \text{True} \quad \text{Eq?}(x, S(x)) \rightarrow \text{False}$$

Она локально конфлюэнтна (проверьте), но не является конфлюэнтной.

### Лемма Ньюмана

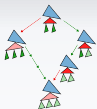
В завершающихся TRS локальная конфлюэнтность эквивалентна конфлюэнтности.



## Критические пары

### Напоминание

Термы  $T_1$  и  $T_2$  унифицируются, если  $\exists \sigma (T_1 \sigma = T_2 \sigma)$ . Если для всех  $\sigma'$  таких, что  $T_1 \sigma' = T_2 \sigma'$  существует подстановка  $\eta$  такая, что  $T_1 \sigma' = (T_1 \sigma) \eta$ , тогда  $\sigma$  — most general unifier для  $T_1$  и  $T_2$  ( $\sigma = \text{mgu}(T_1, T_2)$ ).

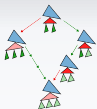


## Критические пары

### Напоминание

Термы  $T_1$  и  $T_2$  унифицируются, если  $\exists \sigma (T_1 \sigma = T_2 \sigma)$ . Если для всех  $\sigma'$  таких, что  $T_1 \sigma' = T_2 \sigma'$  существует подстановка  $\eta$  такая, что  $T_1 \sigma' = (T_1 \sigma) \eta$ , тогда  $\sigma$  — most general unifier для  $T_1$  и  $T_2$  ( $\sigma = \text{mgu}(T_1, T_2)$ ).

Правила  $L_1 \rightarrow R_1$  и  $L_2 \rightarrow R_2$  образуют критическую пару  $\langle R_1 \sigma, (L_1 \sigma)[P \sigma := R_2 \sigma] \rangle$ , если в  $L_1$  существует подтерм  $P$ , не равный переменной, такой что  $P \sigma = L_2 \sigma$ .



## Левострогие TRS

TRS называется левострогим, если в левых частях её правил нет повторных переменных.

Если TRS левострогий и не имеет нетривиальных критических пар, то она конфлюэнтна.

Если HOF редуцируется к нормальной форме, эта нормальная форма единственна. (см. алгебру комбинаторной логики)

Все завершающиеся функции HOF имеют нормальную форму.

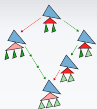




## Теорема о конфлюэнтности

Пара термов  $\langle T_1, T_2 \rangle$  объединяема  $\Leftrightarrow T_1 \rightarrow^* \circ \leftarrow^* T_2$ .  
TRS локально конфлюэнтна  $\Leftrightarrow$  все её критические пары объединяемы.

Рассмотрим TRS  $\mathcal{T} = \{f(g(x)) \rightarrow x, g(f(x)) \rightarrow x\}$ . В  $\mathcal{T}$  единственные критические пары — это  $\langle f(x), f(x) \rangle$  для  $f(g(f(x)))$  и  $\langle g(x), g(x) \rangle$  для  $g(f(g(x)))$  — объединяемые.

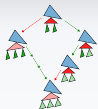


## Теорема о конфлюэнтности

Пара термов  $\langle T_1, T_2 \rangle$  объединяема  $\Leftrightarrow T_1 \rightarrow^* \circ \leftarrow^* T_2$ .  
TRS локально конфлюэнтна  $\Leftrightarrow$  все её критические пары объединяемы.

Рассмотрим TRS  $\mathcal{T} = \{f(g(x)) \rightarrow x, g(f(x)) \rightarrow x\}$ . В  $\mathcal{T}$  единственные критические пары — это  $\langle f(x), f(x) \rangle$  для  $f(g(f(x)))$  и  $\langle g(x), g(x) \rangle$  для  $g(f(g(x)))$  — объединяемые.

Рассмотрим TRS  $\mathcal{T}$  с правилом  $f(f(x)) \rightarrow g(x)$ . Она порождает критическую пару  $\langle f(g(x)), g(f(x)) \rangle$ , не сводимую к общему терму. Если добавить к  $\mathcal{T}$  правило  $f(g(x)) \rightarrow g(f(x))$ , оно не испортит завершаемость (почему?), но появится конфлюэнтность.



## Пример

Аксиомы списка	Аксиомы ошибки
$\text{car}(\text{cons } x \ l) = x$	$\text{car } \text{nil} = \text{error}_a$
$\text{cdr}(\text{cons } x \ l) = l$	$\text{cdr } \text{nil} = \text{error}_l$
$\text{isempty? } \text{nil} = \text{true}$	$\text{cons } \text{error}_a \ l = \text{error}_l$
$\text{isempty? } (\text{cons } x \ l) = \text{false}$	$\text{cons } x \ \text{error}_l = \text{error}_l$
$\text{cond}_a \ \text{true } x \ y = x$	$\text{car } \text{error}_l = \text{error}_a$
$\text{cond}_a \ \text{false } x \ y = y$	$\text{cdr } \text{error}_l = \text{error}_l$
$\text{cond}_b \ \text{true } v1 \ v2 = v1$	$\text{isempty? } \text{error}_l = \text{error}_b$
$\text{cond}_b \ \text{false } v1 \ v2 = v2$	$\text{cond}_a \ \text{error}_b \ x \ y = \text{error}_a$
$\text{cond}_l \ \text{true } l1 \ l2 = l1$	$\text{cond}_b \ \text{error}_b \ x \ y = \text{error}_b$
$\text{cond}_l \ \text{false } l1 \ l2 = l2$	$\text{cond}_l \ \text{error}_b \ x \ y = \text{error}_l$



## Пример

Аксиомы списка	Аксиомы ошибки
$\text{car}(\text{cons } x \text{ l}) = x$	$\text{car nil} = \text{error}_a$
$\text{cdr}(\text{cons } x \text{ l}) = \text{l}$	$\text{cdr nil} = \text{error}_l$
$\text{isempty? nil} = \text{true}$	$\text{cons error}_a \text{ l} = \text{error}_l$
$\text{isempty? (cons } x \text{ l)} = \text{false}$	$\text{cons } x \text{ error}_l = \text{error}_l$
$\text{cond}_a \text{ true } x \text{ y} = x$	$\text{car error}_l = \text{error}_a$
$\text{cond}_a \text{ false } x \text{ y} = y$	$\text{cdr error}_l = \text{error}_l$
$\text{cond}_b \text{ true } v1 \text{ v2} = v1$	$\text{isempty? error}_l = \text{error}_b$
$\text{cond}_b \text{ false } v1 \text{ v2} = v2$	$\text{cond}_a \text{ error}_b \text{ x y} = \text{error}_a$
$\text{cond}_l \text{ true } l1 \text{ l2} = l1$	$\text{cond}_b \text{ error}_b \text{ x y} = \text{error}_b$
$\text{cond}_l \text{ false } l1 \text{ l2} = l2$	$\text{cond}_l \text{ error}_b \text{ x y} = \text{error}_l$
Безопасная кр. пара: $\langle \text{error}_a, \text{car error}_l \rangle$ (правила $\text{car}(\text{cons } x \text{ l}) = x$ и $\text{cons error}_a \text{ l} = \text{error}_l$ ).	



## Пример

Аксиомы списка	Аксиомы ошибки
$\text{car}(\text{cons } x \text{ l}) = x$	$\text{car nil} = \text{error}_a$
$\text{cdr}(\text{cons } x \text{ l}) = \text{l}$	$\text{cdr nil} = \text{error}_l$
$\text{isempty? nil} = \text{true}$	$\text{cons error}_a \text{ l} = \text{error}_l$
$\text{isempty? (cons } x \text{ l)} = \text{false}$	$\text{cons } x \text{ error}_l = \text{error}_l$
$\text{cond}_a \text{ true } x \text{ y} = x$	$\text{car error}_l = \text{error}_a$
$\text{cond}_a \text{ false } x \text{ y} = y$	$\text{cdr error}_l = \text{error}_l$
$\text{cond}_b \text{ true } v1 \text{ v2} = v1$	$\text{isempty? error}_l = \text{error}_b$
$\text{cond}_b \text{ false } v1 \text{ v2} = v2$	$\text{cond}_a \text{ error}_b \text{ x y} = \text{error}_a$
$\text{cond}_l \text{ true } l1 \text{ l2} = l1$	$\text{cond}_b \text{ error}_b \text{ x y} = \text{error}_b$
$\text{cond}_l \text{ false } l1 \text{ l2} = l2$	$\text{cond}_l \text{ error}_b \text{ x y} = \text{error}_l$

Безопасная кр. пара:  $\langle \text{error}_a, \text{car error}_l \rangle$  (правила  $\text{car}(\text{cons } x \text{ l}) = x$  и  $\text{cons error}_a \text{ l} = \text{error}_l$ ).

Проблемная кр. пара:  $\langle \text{l}, \text{cdr error}_l \rangle$  (правила  $\text{cdr}(\text{cons } x \text{ l}) = \text{l}$  и  $\text{cons error}_a \text{ l} = \text{error}_l$ ). Спецификация противоречива.



## Исправление ошибок

$\text{car}(\text{cons } x \text{ } l) = x$

$\text{cdr}(\text{cons } x \text{ } l) = l$

$\text{isempty? nil} = \text{true}$

$\text{isempty? (cons } x \text{ } l) = \text{false}$

$\text{cond}_a \text{ true } x \text{ } y = x$

$\text{cond}_a \text{ false } x \text{ } y = y$

$\text{cond}_b \text{ true } v1 \text{ } v2 = v1$

$\text{cond}_b \text{ false } v1 \text{ } v2 = v2$

$\text{cond}_l \text{ true } l1 \text{ } l2 = l1$

$\text{cond}_l \text{ false } l1 \text{ } l2 = l2$

$\text{error?}_l \text{ error}_l = \text{true}$

$\text{car nil} = \text{error}_a$

$\text{cdr nil} = \text{error}_l$

$\text{cons error}_a \text{ } l = \text{error}_l$

$\text{cons } x \text{ error}_l = \text{error}_l$

$\text{car error}_l = \text{error}_a$

$\text{cdr error}_l = \text{error}_l$

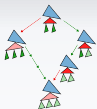
$\text{isempty? error}_l = \text{error}_b$

$\text{cond}_a \text{ error}_b \text{ } x \text{ } y = \text{error}_a$

$\text{cond}_b \text{ error}_b \text{ } x \text{ } y = \text{error}_b$

$\text{cond}_l \text{ error}_b \text{ } x \text{ } y = \text{error}_l$

$\text{error?}_l \text{ nil} = \text{false}$



## Исправление ошибок

~~car(cons x l) = x~~  
~~cdr(cons x l) = l~~  
isempty? nil = true  
~~isempty? (cons x l) = false~~  
 $\text{cond}_a \text{ true } x \ y = x$   
 $\text{cond}_a \text{ false } x \ y = y$   
 $\text{cond}_b \text{ true } v1 \ v2 = v1$   
 $\text{cond}_b \text{ false } v1 \ v2 = v2$   
 $\text{cond}_l \text{ true } l1 \ l2 = l1$   
 $\text{cond}_l \text{ false } l1 \ l2 = l2$   
 $\text{error?}_l \text{ error}_l = \text{true}$

$\text{car nil} = \text{error}_a$   
 $\text{cdr nil} = \text{error}_l$   
 $\text{cons error}_a \ l = \text{error}_l$   
 $\text{cons } x \ \text{error}_l = \text{error}_l$   
 $\text{car error}_l = \text{error}_a$   
 $\text{cdr error}_l = \text{error}_l$   
 $\text{isempty? error}_l = \text{error}_b$   
 $\text{cond}_a \text{ error}_b \ x \ y = \text{error}_a$   
 $\text{cond}_b \text{ error}_b \ x \ y = \text{error}_b$   
 $\text{cond}_l \text{ error}_b \ x \ y = \text{error}_l$   
 $\text{error?}_l \text{ nil} = \text{false}$

$\text{error?}_l (\text{cons } x \ l) = \text{cond}_b (\text{error?}_a \ x) \ \text{true}$   
 $\qquad (\text{cond}_b (\text{error?}_l \ l) \ \text{true} \ \text{false})$

$\text{car}(\text{cons } x \ l) = \text{cond}_a (\text{error?}_a \ x) \ \text{error}_a$   
 $\qquad (\text{cond}_a (\text{error?}_l \ l) \ \text{error}_a \ x)$

*...и т.д. для  $\text{error?}_a$  и  $\text{error?}_b$  и для двух других проблемных правил.*

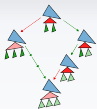


## Алгоритм Кнута–Бендикса

Пополнение правилами переписывания по критическим парам не меняет отношение доказуемого равенства в TRS, но может плохо повлиять на завершаемость  $\Rightarrow$  пополнение должно контролироваться нетеровым (wfo) порядком  $\preceq$ .

Пусть дана TRS  $\mathcal{T}$ , доказуемо завершающаяся в ФуМА  $\mathcal{A}$  с wfo  $\preceq$ . Будем расширять  $\mathcal{T}$  правилами  $L_i \rightarrow R_i$ , где  $\langle L_i, R_i \rangle$  — критические пары и  $R_i^{\mathcal{A}} \preceq L_i^{\mathcal{A}}$  до неподвижной точки расширения. Результат — конфлюэнтная  $\mathcal{T}$ .





## Проверка алг. типа данных с помощью TRS

- Анализ всех критических пар конечен.
- Если удалось получить конфлюэнтную систему пополнением, тогда легко проверить, корректны ли выполняющиеся в ней эквивалентности, построив нормальные формы всех термов.
- Если нашлась критическая пара вида  $\langle x, t \rangle$ , тогда спецификация противоречива.



## Бесконечная пополняемость

Рассмотрим  $\text{trs}$  с двумя правилами:

$$u(x) + u(y) \rightarrow u(x + y), \quad (x + y) + z \rightarrow x + (y + z).$$

Для доказательства завершаемости в ней положим

$$x +^A y = (y)^x, \quad u^A(x) = x^2.$$



## Бесконечная пополняемость

Рассмотрим  $\text{trs}$  с двумя правилами:

$$u(x) + u(y) \rightarrow u(x + y), \quad (x + y) + z \rightarrow x + (y + z).$$

Для доказательства завершаемости в ней положим

$$x +^A y = (y)^x, \quad u^A(x) = x^2.$$

Первая критическая пара:  $\langle u(x) + (u(y) + z), u(x + y) + z \rangle$ .

Пополним  $\text{trs}$  правилом  $u(x + y) + z \rightarrow u(x) + (u(y) + z)$  и проверим, сохранится ли монотонность в ФуМА.



## Бесконечная пополняемость

Рассмотрим  $\text{trs}$  с двумя правилами:

$$u(x) + u(y) \rightarrow u(x + y), \quad (x + y) + z \rightarrow x + (y + z).$$

Для доказательства завершаемости в ней положим  $x +^A y = (y)^x$ ,  $u^A(x) = x^2$ .

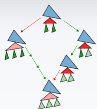
Первая критическая пара:  $\langle u(x) + (u(y) + z), u(x + y) + z \rangle$ .

Пополним  $\text{trs}$  правилом  $u(x + y) + z \rightarrow u(x) + (u(y) + z)$  и проверим, сохранится ли монотонность в ФуМА.

Каждая новая критическая пара

$\langle u^n(x) + (u^n(y) + z), u^n(x + y) + z \rangle$  будет порождать очередное правило переписывания

$u^n(x + y) + z \rightarrow u^n(x) + (u^n(y) + z)$ , убывающее в ФуМА.



## Вычисление базисов Грёбнера

Зададим упорядочение  $\triangleleft$  на  $n$ -ках из  $\mathbb{N}$  (степенях переменных в мономе) такое, что:

- $\forall s([0, \dots, 0] \triangleleft s)$ ;
- $\forall s, t_1, t_2(t_1 \triangleleft t_2 \Rightarrow s + t_1 \triangleleft s + t_2)$ .

Инициальный моном  $\text{in}_{\triangleleft}(g)$  полинома  $g$  — это моном, больший всех других мономов  $g$  в смысле  $\triangleleft$ .

Выберем мономиальный порядок  $\triangleleft$ . Найти базис  $G$  идеала  $I$  полиномов над полем такой, что инициальный идеал  $\text{in}_{\triangleleft}(I) = \langle \text{in}_{\triangleleft}(g) \mid g \in G \rangle$  (порождается инициальными мономами полиномов из  $G$ ).

Критерий Бюхбергера:  $G$  — базис Грёбнера, если все  $r = \frac{\max(\text{in}_{\triangleleft}(g_i), \text{in}_{\triangleleft}(g_j))}{\text{in}_{\triangleleft}(g_i)} * g_i - \frac{\max(\text{in}_{\triangleleft}(g_i), \text{in}_{\triangleleft}(g_j))}{\text{in}_{\triangleleft}(g_j)} * g_j$  имеют разложение в  $G$ . Операция максимума  $\max$  — поэлементная.



## Алгоритмы Бюхбергера и Кнута–Бендикса

- ❶ Если в  $F$  выполнен критерий Бюхбергера  $\Rightarrow$  базис построен.
- ❷ Иначе существует критическая пара  $\langle g_i, g_j \rangle$ , порождающая неразложимый  $r$ . Добавляем  $r$  в базис и продолжаем.



## Алгоритмы Бюхбергера и Кнута–Бендикса

- ❶ Если в  $F$  выполнен критерий Бюхбергера  $\Rightarrow$  базис построен.
- ❷ Иначе существует критическая пара  $\langle g_i, g_j \rangle$ , порождающая неразложимый  $r$ . Добавляем  $r$  в базис и продолжаем.

Конечность вычисления  $\text{lfp}$  (наименьшей неподвижной точки) гарантируется теоремой Гильберта о конечнопорожденности идеалов.



## Алгоритмы Бюхбергера и Кнута–Бендикса

- 1 Если в  $F$  выполнен критерий Бюхбергера  $\Rightarrow$  базис построен.
- 2 Иначе существует критическая пара  $\langle g_i, g_j \rangle$ , порождающая неразложимый  $r$ . Добавляем  $r$  в базис и продолжаем.

Конечность вычисления  $\text{lfp}$  (наименьшей неподвижной точки) гарантируется теоремой Гильберта о конечнопорожденности идеалов.

Алгоритм Бюхбергера (почти) моделируется алгоритмом пополнения Кнута–Бендикса над алгеброй с правилами переписывания — аксиомами кольца. Для каждого полинома  $p \in F$ , где  $p = \text{in}_{\triangleleft} p + p'$ , добавляем правила  $\text{in}_{\triangleleft} p \rightarrow p'$  и  $\text{in}_{\triangleleft} px \rightarrow p' * x$ .





## Алгоритмы Бюхбергера и Кнута–Бендикса

- 1 Если в  $F$  выполнен критерий Бюхбергера  $\Rightarrow$  базис построен.
- 2 Иначе существует критическая пара  $\langle g_i, g_j \rangle$ , порождающая неразложимый  $r$ . Добавляем  $r$  в базис и продолжаем.

Конечность вычисления  $\text{lfp}$  (наименьшей неподвижной точки) гарантируется теоремой Гильберта о конечнопорожденности идеалов.

Алгоритм Бюхбергера (почти) моделируется алгоритмом пополнения Кнута–Бендикса над алгеброй с правилами переписывания — аксиомами кольца. Для каждого полинома  $p \in F$ , где  $p = \text{in}_{\triangleleft} p + p'$ , добавляем правила  $\text{in}_{\triangleleft} p \rightarrow p'$  и  $\text{in}_{\triangleleft} px \rightarrow p' * x$ .

Проблема: коммутативность и ассоциативность убивают нётеровость  $\text{trs}$  и завершаемость пополнения.



## Свободные модели

Рассмотрим сигнатуру  $\Sigma$  и множество формул логики первого порядка  $P$  над  $\Sigma$ . Алгебра  $\mathcal{A}$  называется свободной моделью (loose model) над  $\langle \Sigma, P \rangle$ , если  $\mathcal{A} \models P$ .



## Свободные модели

Рассмотрим сигнатуру  $\Sigma$  и множество формул логики первого порядка  $P$  над  $\Sigma$ . Алгебра  $\mathcal{A}$  называется свободной моделью (loose model) над  $\langle \Sigma, P \rangle$ , если  $\mathcal{A} \models P$ .

### Теорема об изоморфизме

Пусть  $\mathcal{A}, \mathcal{A}'$  — алгебры,  $\phi$  — изоморфизм  $\mathcal{A} \rightarrow \mathcal{A}'$ . Тогда для всякой FO-формулы  $P(X)$  и окружения  $\alpha(X)$  выполняется эквивалентность  $\mathcal{A}, \alpha \models P \Leftrightarrow \mathcal{A}', \phi \circ \alpha \models P$ .



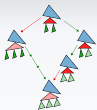
## Свободные модели

Рассмотрим сигнатуру  $\Sigma$  и множество формул логики первого порядка  $P$  над  $\Sigma$ . Алгебра  $\mathcal{A}$  называется свободной моделью (loose model) над  $\langle \Sigma, P \rangle$ , если  $\mathcal{A} \models P$ .

### Теорема об изоморфизме

Пусть  $\mathcal{A}, \mathcal{A}'$  — алгебры,  $\phi$  — изоморфизм  $\mathcal{A} \rightarrow \mathcal{A}'$ . Тогда для всякой FO-формулы  $P(X)$  и окружения  $\alpha(X)$  выполняется эквивалентность  $\mathcal{A}, \alpha \models P \Leftrightarrow \mathcal{A}', \phi \circ \alpha \models P$ .

Невыразимость:  $\mathcal{A}, \alpha(x) \models P(x) \Rightarrow \mathcal{A} \models \forall x P(x)$



## Свободные модели

Рассмотрим сигнатуру  $\Sigma$  и множество формул логики первого порядка  $P$  над  $\Sigma$ . Алгебра  $\mathcal{A}$  называется свободной моделью (loose model) над  $\langle \Sigma, P \rangle$ , если  $\mathcal{A} \models P$ .

### Теорема об изоморфизме

Пусть  $\mathcal{A}, \mathcal{A}'$  — алгебры,  $\phi$  — изоморфизм  $\mathcal{A} \rightarrow \mathcal{A}'$ . Тогда для всякой FO-формулы  $P(X)$  и окружения  $\alpha(X)$  выполняется эквивалентность  $\mathcal{A}, \alpha \models P \Leftrightarrow \mathcal{A}', \phi \circ \alpha \models P$ .

Невыразимость:  $\mathcal{A}, \alpha(x) \models P(x) \Rightarrow \mathcal{A} \models \forall x P(x)$

Рассмотрим алгебру над  $\mathbb{R}$  с  $0$  и  $+$  (со стандартными аксиомами). Единственные предикаты от одной переменной, выражимые в этой алгебре и не являющиеся общезначимыми — это  $x = 0$  и  $x \neq 0$ .



## Выразимость в модели $\mathbb{R}$

Рассмотрим алгебру  $\mathcal{R}$  над  $\mathbb{R}$  с константами 0 и 1 и операциями  $+$ ,  $*$  (в стандартной аксиоматике).

- Можно ли в  $\mathcal{R}$  выразить  $>$ ?
- Пусть  $\phi$  — автоморфизм  $\mathcal{R} \rightarrow \mathcal{R}$ . Пусть  $q$  — рациональное число. Охарактеризовать предикат  $P(\phi(q), q)$ .
- Охарактеризовать все автоморфизмы  $\mathcal{R}$ .



## Виды программных семантик

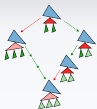
- ❶ Аксиоматическая — логические правила вывода, описывающие пути вычислений. Определяет равенство между программами.



## Виды программных семантик

- ❶ Аксиоматическая — логические правила вывода, описывающие пути вычислений. Определяет равенство между программами.
- ❷ Денотационная — алгебраическая модель, определяющая функцию означивания (денотат):  $\llbracket P \rrbracket$ .
  - Инициальные модели — + рекурсия;
  - Свободные модели — выразимость.





## Виды программных семантик

- 1 Аксиоматическая — логические правила вывода, описывающие пути вычислений. Определяет равенство между программами.
- 2 Денотационная — алгебраическая модель, определяющая функцию означивания (денотат):  $\llbracket P \rrbracket$ .
  - Инициальные модели — + рекурсия;
  - Свободные модели — выразимость.
- 3 Операционная — описание исполнения программ в конкретной модели вычислений.