Теория атрибутных типов (черновик)

Михайлов Анатолий Андреевич (601-41)

28 января 2025 г.

Аннотация

В работе описывается формализм, основывающийся на теории типов, имеющий свойства, полезные в области формальных доказательств. В работе приведены примеры использования формализма в доказательстве теорем в арифметике Пеано. Работа написана студентом первого курса, не разбирающегося в области, о которой он пишет. Тем не менее, его находка показалась ему достаточно интересной, чтобы ей поделиться, даже не имея глубины понимания темы для достаточного исследования.

О черновике У меня не получилось в полной мере формально описать систему, т.к. не достаточно знаний. В частности, у меня не получилось указать в определениях переменное количество типов (вместо этого я указывал всего два типа или троеточие, надеясь на интуицию читателя). Прошу помочь с этим: подсказать что и где можно прочитать по теме, проконсультироваться с знающими о теории типов. Я «понахватался верхов», у меня «каша в голове», хочу привести все это в порядок.

Основная идея Выражение может иметь несколько типов одновременно. В таком случае, типы можно воспринимать как атрибуты этого выражения. Пример: утверждение «$a : A, B$» значит «выражение a имеет атрибуты A и B». Но, это еще не значит, что выражение a не имеет атрибута C. Это значит лишь то, что для выражения a не доказан атрибут C. Его можно получить, применив, например a к $f : A \to C$.

Проверка типов Пусть дана функция $f : (A, B) \to C$. Тогда применение этой функции к аргументу a считается возможным, если $a : A, B$, при этом a может иметь любое количество дополнительных типов, главное чтобы были типы A и B.

Несколько типов функции Эта особенность позволяет элегантно (по мнению автора) реализовать варианты функции. Отличный пример – определение функции S в примерах с арифметикой Пеано. Механизм выбора варианта прост – выбирается самый левый тип функции, подходящий по атрибутам аргументу. Соответственно, слева нужно размещать самые частные случаи, а справа самые общие.

Типы как утверждения Эта интерпретация также применима к описываемому формализму. Пример: $x : Nat, Even$ означает, что x – натуральное четное число. Переводя на язык логики предикатов – $Nat(x) \wedge Even(x)$. Каждое утверждение о типизации можно тривиально перевести на язык логики предикатов.

Описание в системе натурального вывода Этот параграф требует значительной доработки/переработки. Если a,b – термы, A,B,C,D – типы, то:

Абстракция:

$$\frac{a :\ A, ... \quad b = \lambda x :\ (B, ...).\, a}{b :\ (B, ...) \to (A, ...)}$$

Применение:

$$\frac{a :\ A, ... \quad b :\ (A \to (C, ...)), (B \to (C, D, ...))}{(ba) :\ C, ...}$$

$$\frac{a :\ B, ... \quad b :\ (A \to (C, ...)), (B \to (C, D, ...))}{(ba) :\ C, D, ...}$$

Доказательство четности числа 4 в арифметике Пеано Оно вдохновлялось доказательством в системе Twelf, но получилось значительно короче и дополнительно включает в себя определение нечетных чисел.
Аксиомы:

$$0 :\ Nat,\ Even$$

$$S :\ (Nat, Even) \to (Nat, Odd), (Nat, Odd) \to (Nat, Even)$$

Построения:

$$1 = (S0) \qquad 2 = (S1) \qquad 3 = (S2) \qquad 4 = (S3)$$

Термы 1, 3 будут иметь атрибуты (Nat, Odd)
Термы 0, 2, 4 будут иметь атрибуты $(Nat, Even)$

Доказательство четности/нечетности числа n+2 в арифметике Пеано Аксиомы:

$$0 : \, Nat, \, Even$$

$$S : \, (Nat, Even) \to (Nat, Odd), (Nat, Odd) \to (Nat, Even)$$

Построения:

$$S' = \lambda n : \, Nat. \, (S(Sn))$$

S' имеет атрибуты $(Nat, Even) \to (Nat, Even), (Nat, Odd) \to (Nat, Odd)$

О применении в языках программирования Это изначальная цель разработки. Есть идеи о реализации языка программирования или системы автоматизированных доказательств (или того и другого, как в Coq), основывающегося на этом формализме. Есть опыт по реализации простых формализмов (типа λ-исчисления), но я понимаю, что мне не хватает знаний для чего-либо серьезного.