ЛОГИЧЕСКИЕ ВОПРОСЫ ТЕОРИИ ТИПОВ ДАННЫХ
(Вычислительные системы)
1986 год Выпуск II4

УДК 510.25:519.68

ПРОЕКТИРОВАНИЕ Σ-ПРОГРАММ. Σ-ОЦЕНИВАЕМОСТЬ

Д.И.Свириденко

Введение

Настоящая работа посвящена обсуждению роли и места конструктивизма и, в частности, уточнению понятия "конструктивная логика" в концепции Σ-программирования [I] применительно к проблеме проектирования Σ-программ [2].

Статья состоит из двух частей. Первая, методологическая (§I), содержит краткий исторический анализ основных подходов к изучению конструктивных аспектов математики. Здесь же указывается на необходимость изучения связей между этими подходами и описывается точка зрения, в рамках которой становится возможным эти связи устанавливать и исследовать. Вторая (§2-§4) представляет собой формальное изложение начального фрагмента данной точки зрения. Желание сделать материал более доступным, ясным и последовательным, а также ограничение на объем статьи заставили автора отказаться от подробных доказательств приводимых результатов. Главная цель - ясно изложить логико-математические аспекты проблемы проектирования Σ-программ.

Таким образом, статья продолжает цикл работ, объединенных общим название "Проектирование Σ-программ", что и отражено в ее заголовке.

§I. Понятие "конструктивное" в математике

Математика со времен ее оформления как самостоятельной науки всегда носила двойственный характер. С одной стороны, она предоставляет исследователю язык и средства для описания точно определенных конструкций, с помощью которых человеку удобно упорядочи -

вать информацию о внешнем мире и получать строго обоснованные следствия из тех предположений, которые он сделал, и тех знаний, которые он уже получил. С другой стороны, при решении математической задачи часто необходимо не просто доказать существование объекта с нужными свойствами,а указать построение такого объекта. Следует заметить, что в самом начале развития математики дескриптивные (описательные) и конструктивные ее аспекты находились в неразрывном единстве. Так, например, доказательство теоремы существования в евклидовой геометрии обязательно дает и построение искомого объекта. Однако появление действительных чисел разрушило эту гармонию, и математикам пришлось сделать выбор: что предпочесть - описание или построение? Победила дескриптивная, или платонистская точка зрения на природу математики, основу которой составила аристотелевская логика.

Возникший разрыв между двумя аспектами математики оставался вне интереса самих математиков вплоть до начала XX века. Первый сигнал, хотя и не воспринятый вначале многими математиками, как тревожный, прозвучал в тот момент, когда математика переживала кризис своих оснований, связанный с парадоксами наивной теории множеств. В связи с этим кризисом, Л.Э.Я.Брауэр в 1908 году написал работу "О недостоверности логических принципов", в которой он подверг критике существующую точку зрения на математику. По его мнению, парадоксы являются лишь побочными, хотя и наиболее заметными проявлениями общей ошибки, допущенной при развитии математики, когда был сделан выбор в сторону дескриптивного аспекта. Одну из основных причин неконструктивности математических выводов в платонистской математике Брауэр видел в необоснованном и повсеместном использовании закона исключенного третьего. Брауэром также был отвергнут как универсальный метод рассуждения от противного, тесно связанный с предыдущим законом. Кроме того, он предлагал отказаться от основной парадигмы платонистской математики - актуальной бесконечности, т.е. от рассмотрения бесконечных множеств как завершенных конструкций. К сожалению, сам Брауэр оказался не в состоянии ясно и точно сформулировать основания новой точки зрения на природу математики и поэтому просто утверждал,что она лучше отвечает интуиции человека (отсюда и название его доктрины - "интуиционизм").

Пожалуй, первыми серьезными попытками уточнить представления Брауэра были исследования А.Н.Колмогорова. Он предложил общую схему интерпретации логической системы Брауэра как исчисления задач. В интерпретации Колмогорова логические связки служат способом композиции задач, построения более сложных задач из простых, где аксиомы выступают как элементарные задачи. Правила вывода при этом играют роль методов композиции решений задач (или методов декомпозиции задач на подзадачи). Однако определение Колмогорова оставалось нестрогим в том пункте, где говорилось о способах решения задач.

Нужно сказать, что человечество с древнейших времен пыталось найти универсальные способы решения задач и, в частности, математических проблем. И хотя все подобные попытки заканчивались неудачей, они имели и положительные результаты - постепенно уточнялось понятие математической задачи и метода ее решения. Непреходящее значение в этом смысле имеют работы Г.Лейбница и Д.Гильберта. Так, согласно Лейбницу путь к созданию "универсальной математики", которую он назвал "логикой творческой силы", и "универсальной науки" вообще лежит через создание универсального языка и универсальных средств оперирования с ним. Лейбниц считал, что формальные правила оперирования с умозаключениями сделают процесс рассуждения легко проверяемым, и если между людьми возникнут споры, то потребуется лишь сказать "подсчитаем!, дабы без дальнейших околичностей выяснилось, кто прав".

К необходимости осуществить такой "подсчет" вплотную подошел Д.Гильберт в связи с выдвинутой им программой обоснования математики. Нужно заметить, что Гильберт был решительно против попыток Брауэра перестроить всю математику. Программа Гильберта основывалась на аксиоматизации математических концепций, построении формальных моделей содержательной математики и последующем исследовании вопросов непротиворечивости таких моделей. При этом выдвигалось требование, чтобы подобные исследования осуществлялись такими средствами, надежность и корректность которых не вызывала бы ни у кого сомнений. В качестве таких средств Гильберт предлагал рассматривать финитные конструкции, в силу чего его подход получил название финитизма Гильберта. Им была создана дошедшая до наших дней точная модель математического языка - языка исчисления предикатов - и введено математическое понятие доказательства. Позднее в работах Чёрча, Клини, Тьюринга, Поста, Гёделя, Маркова и других математических логиков было дано математическое уточнение понятия "метод (способ) решения" - алгоритм. Появилась возможность точно поставить проблему отыскания универсального метода решения всех математических проблем. К счастью,для математики эта проблема была решена отрицательно: Чёрч установил, что классическое исчисление предикатов I-го порядка неразрешимо. Оказалось, что, даже ограничившись задачами элементарной арифметики, мы не в состоянии найти универсального способа их решения. В связи с этим встал вопрос об изучении алгоритмической природы различных математических проблем. Здесь имелись, по крайней мере, две возможности.

Во-первых, располагая точными понятиями "алгоритм","теория", "модель теории", можно было решать эти проблемы, оставаясь в рамках платонистской математики и пользуясь классической логикой.Результатами осуществления этой возможности являются теория нумераций и теория конструктивных моделей (см.,например, [3,4]),изучающие алгоритмическую природу множеств, элементарных теорий и их моделей. У истоков этих теорий стояли выдающиеся математики А.И.Мальцев и А.Тарский. Отметим,что А.И.Мальцев [5] впервые с единых позиций рассмотрел алгоритмические свойства нумерованных алгебр и множеств произвольной природы, допускающих нумерации. В настоящее время теория конструктивных моделей и теория нумераций активно развиваются как у нас в стране, так и за рубежом.

Дальнейшее изучение математических свойств алгоритмических объектов в платонистской математике можно осуществлять в двух направлениях. Во-первых, можно рассматривать все более узкие классы алгоритмов и строить их теории (примером может служить теория алгоритмов с оценками), а, во-вторых, можно, абстрагируясь от некоторых конкретных моментов,строить и исследовать более общие модели вычислений. Так, например,изучение категории нумерованных множеств позволило Ю.Л.Ершову построить теорию f-пространств [6] и их ретрактов - А-пространств [7], обладающую многими полезными для приложений свойствами. Заметим, что аналогичную конструкцию, соответствующую полному A_0-пространству - непрерывную решетку, ввел независимо Д.Скотт, изучая проблему построения денотационной семантики языков программирования [8,9].Желание формализовать интенсиональные аспекты математической теории вычислений, возникшей на базе исследований Ю.Л.Ершова и Д.Скотта, привело последнего к понятию системы окрестностей [10], а впоследствии к понятию информационной системы [11], которое можно рассматривать как один из вариантов реализации тезиса "выводимость=вычислимость". Заметим, что и в случае системы окрестностей,и в случае информационных систем их экстенсионалами являются полные f_0-пространства Ершова. В [12] введено понятие логической системы, имеющее своими экстенсионалами практически весь спектр конструкций, используемых для описания денотационной семантики языков программирования, включая и полные f_0-пространства. Известны и другие модели обобщенной вычислимости. Одна из них положена в основу Σ-программирования и базируется на идее формульного описания алгоритмических свойств объектов. В наших дальнейших построениях будут использованы обе модели обобщенной вычислимости.

Вторая возможность изучения алгоритмических проблем связана с критикой Брауэра и заключается в использовании понятия алгоритма для уточнения понятия "метод (способ) решения проблем (задач)" в интерпретации А.Н.Колмогорова. Предложенный Гильбертом метод формализации математических теорий был использован последователями Брауэра для уточнения представлений о логической системе (логике), лежащей в основе интуиционистской математики. В 1930 г. А.Гейтинг построил формальную систему интуиционистской логики и на ее базе ввел в рассмотрение формальную интуиционистскую арифметику [13]. Таким образом, к концу 30-х годов все уже было готово для реализации упомянутой выше возможности использования понятия алгоритма в интуиционистской математике. Этот шаг в 1945 г. осуществил С.Клини, который построил семантику интуиционистской арифметики на базе теории рекурсивных функций, уточнив колмогоровское понятие задачи следующим образом: метод - это рекурсивная функция, а решение задачи - это нахождение гёделевского номера рекурсивной функции, выполняющей требуемое преобразование (см., например, [14]). Формуле чисто синтаксически сопоставляется условие на рекурсивные операторы (ее конструктивная расшифровка), а оператор, удовлетворяющий этой расшифровке, объявляется реализацией исходной формулы.

Изучению с помощью аппарата реализуемости поддаются такие достаточно сильные системы, как анализ, теория комбинаторов, теория множеств, теория информационных систем Скотта и т.д.

К настоящему времени построен ряд семантик типа реализуемости. Как правило, каждая модификация понятия реализуемости несколько меняет логику. Так, например, при использовании исходной семантики по Клини целесообразно иногда, как показали А.А.Марков и Н.А.Шанин [15], расширить интуиционистскую логику следующим правилом (принцип Маркова):

$$\frac{\forall\bar{x}\,(\Phi \vee \neg\Phi) \qquad \neg\neg\exists\bar{x}\,\Phi}{\exists\bar{x}\,\Phi},$$

содержательно означающим, что доказывать, скажем, конечность числа шагов выполнения алгоритма можно в классической логике. В связи с этим естественно возникает вопрос: какую логическую систему можно считать конструктивной? Заметим, что распространение идеи реализуемости на случай, когда вместо рекурсивных функций используются объекты другой природы, также может изменить логику. Таким образом, конструктивных логик много. Поэтому следующий (и для нас особенно важный) вопрос относится к тому, как строить конструктивные логики, адекватные имеющимся средствам и поставленным целям? И первый, и второй вопросы весьма нетривиальны, и вряд ли можно в настоящий момент дать однозначные и исчерпывающие ответы на них. Цель настоящей работы - показать возможные подходы к решению данных проблем (акцентируя свое внимание, главным образом, на втором вопросе), используя для этого теорию GES [I].Чтобы как-то очертить область возможных поисков, рассмотрим подробнее методологические установки конструктивизма.

В отличие от платонистской конструктивная математика заменяет принцип актуальной бесконечности на постулат потенциальной осуществимости и считает, что объектами математики могут быть только те конструкции, осуществление которых непосредственно очевидно для исследователя, либо самоочевидные комбинации уже существующих конструкций (например, пары объектов, конечные их последовательности (списки) и т.п.). Другими словами, конструктивные объекты - это такие объекты, на которые можно указать. Так,например, натуральные числа ω , представляющие особый интерес для конструктивной математики, мыслятся как объекты, порождаемые итерацией операции: +I - прибавление единицы. И процессы построения (конструирования) математических сущностей и распознавания их свойств с позиций конструктивизма представляют собой умственные акты. Свойства объектов могут быть либо самоочевидными (распознаваемыми на уровне интуиции, озарения), либо обоснованными в рамках некой логико-математической системы, базирующейся на тех логических принципах, которые достаточно полно и точно учитывают специфику проблемной области и деятельности по созданию математических сущностей, относящихся к этой области. В соответствии с этим, например, принцип математической индукции для ω непосредственно следует из вышеописанного способа порождения натуральных чисел. Отсюда конструктивисты делают вывод, что смысл утверждений лучше задавать не в терминах истины и лжи, как это делает, скажем, Тарский, а в терминах обоснованности, где исходным понятием является "конструкция" ("обоснование" рассматривается как частный случай конструкции). При этом зачастую требуется, чтобы условие "конструкции p есть обоснование утверждения Φ" было эффективно распознаваемым. Важным условием, налагаемым конструктивистами на смысл утверждения, является его "денотативность": смысл утверждения должен конструироваться из смыслов составляющих это утверждение компонент. Смысл же простейших (атомарных) утверждений задается непосредственно, прямым его указанием. Одновременно необходимо позаботиться об отсутствии побочного эффекта при соотнесении утверждениям их смысла: соотнесение смысла сложному утверждению не должно повлечь за собой изменения смысла составляющих это утверждение компонент. Таким образом, прежде чем определять конструктивный смысл утверждений, должен быть задан частичный (фундированный) порядок на множестве всех утверждений, уточняющий наше представление о "сложности" утверждений. Далее будем предполагать, что рассматриваемые содержательные математические теории эксплицируются в языке исчисления предикатов I-го порядка. В этом случае наличие логических связок определяет отношение структурной сложности на формулах языка. Одна из возможных "наивных" схем определения конструктивного смысла утверждений, согласованная с этим отношением и удовлетворяющая сформулированным выше требованиям, может быть задана следующим образом:

(i) "обоснование" утверждения $(\Phi \& \Psi)$ задается "конструкцией" (p, q), где p – "обоснование" Φ, q – "обоснование" Ψ;

(ii) "обоснование" $(\Phi \vee \Psi)$ состоит из "обоснования" Φ или "обоснования" Ψ (вместе с указанием, какой из этих фактов имеет место в действительности);

(iii)[ж] "обоснование" $(\Phi \to \Psi)$ представляет собой "конструкцию" θ, относительно которой утверждается (т.е. считается, что мы

ж) Отрицание $\neg\Phi$ вводится как определимая конструкция $(\Phi \to$ **false**$)$, где **false** есть тождественно ложное утверждение. "Обоснование" $\neg\Phi$ состоит в процедуре эффективного распознавания невозможности указать на "обоснование" Φ. Заметим, что возможны и другие определения конструктивного смысла $\neg\Phi$.

можем распознать соответствующее свойство): если есть "обоснование" Φ, то, пользуясь θ, мы можем построить "обоснование" p утверждения Ψ;

(iv) "обоснование" $\exists x\, \Phi(x)$ состоит из "конструкции" (c, p), где p есть "обоснование" $\Phi(c)$ для c из D;

(v) "обоснование" $\forall x\, \Phi(x)$ представляет собой "конструкцию" θ с распознаваемым свойством: если c есть объект из рассматриваемой "области значений" D для переменной x, то, пользуясь информацией о θ, мы можем построить "обоснование" утверждения $\Phi(c)$.

В (iv) и (v) считается, что область D обладает тем свойством, что либо ее объекты сами олицетворяют собой "обоснования" того, что они есть элементы D, либо с каждым таким объектом c из D можно эффективно связать "конструкцию" c', являющуюся "обоснованием" того, что $c \in D$.

Дальнейшая конкретизация понятия "конструктивный смысл" связана с уточнением понятий "конструкция" и "обоснование" в "наивной" схеме. Практически все ныне существующие конструктивные подходы, формулируемые в языке исчисления предикатов, удовлетворяют (при соответствующих уточнениях) этой схеме. В частности, под нее также подпадает финитизм Гильберта. В свою очередь, дальнейшие уточнения понятий "обоснование" и "конструкция" обычно связывают с конкретизацией конструктивных свойств логических связок, в частности, связки $\vee$ и квантора $\exists$. Существующие уточнения носят либо "семантический", либо "синтаксический" характер.

При "семантических" уточнениях "наивная" схема превращается в формальное определение конструктивной семантики языка исчисления предикатов. Таковы, например, реализуемость по Клини (и ее различные модификации) или интерпретация Гёделя для интуиционистской арифметики[ж]. Для этих семантик семантическую сложность формул увеличивают не только наличие кванторов, но и вложенность импликаций, в отличие от классической интерпретации по Тарскому, где существенное значение имеет лишь число перемен кванторов.

При "синтаксических" уточнениях понятие "обоснование" обычно трактуется как "доказательство" или "класс доказательств" в некоторой логической системе (подход Правитца, подход "формулы как ти-

---

ж) Заметим, что возможны семантики "истинностного" типа – псевдобулевы алгебры, топологические модели Бета и Крипке и др.

пы" и др.). При этом, в соответствии с "наивной" схемой, такая логическая система должна удовлетворять так называемым D - и E-свойствам, являющимся "синтаксической" переформулировкой конструктивных свойств связки $\vee$ и квантора $\exists$. Более того, многими конструктивистами эти свойства рассматриваются как классификационный признак конструктивности логических систем. Такими свойствами обладает, например, интуиционистская логика. Более точно, пусть $\mathcal{L}$ - некоторая логическая система и $\vdash_{\mathcal{L}}$ обозначает выводимость в этой системе. Тогда

а) D-свойство: если $\vdash_{\mathcal{L}} (\Phi \vee \Psi)$, то $\vdash_{\mathcal{L}} \Phi$, или $\vdash_{\mathcal{L}} \Psi$;

б) E-свойство: если $\vdash_{\mathcal{L}} \exists x\ \Phi(x)$, то $\vdash_{\mathcal{L}} \Phi(t)$ для некоторого терма t.

Для приложений (в частности, для синтеза программ) наиболее существенным признается E-свойство. Более того, в отдельных работах по синтезу программ (см., например, [16, 17]) предлагается считать логическую систему конструктивной, если можно указать алгоритм, строящий (извлекающий) по доказательству формулы $\exists x\ \Phi(x)$ терм t, удовлетворяющий свойству Φ. Поскольку в приложениях существенным является простота алгоритма извлечения, то дополнительно иногда требуют от этого алгоритма его "естественности" [17], понимая под этим то, что каждая компонента извлекаемой программы должна быть образом некоторой конструкции доказательства. Во многих случаях этот и подобные ему классификационные признаки (назовем их <u>синтаксическими</u>) хотя и позволяют в определенной мере удовлетворительно отвечать на первый интересующий нас вопрос: какие логические системы считать конструктивными?, - но они совершенно бесполезны при поисках ответа на более важный для нас второй вопрос: как строить конструктивные логики, откуда они берутся? Другими словами, нужны признаки - "руководства к действию". Один из таких признаков предлагается в настоящей работе. По причинам, которые станут понятными позже, этот признак (или лучше метод) отнесен к разряду <u>семантических</u>.

Главные методологические положения предлагаемого метода[*)] следующие:

*) Предлагаемый метод использует опыт работ [18, 19]. Идея определения понятия Σ-оценки (§3) с помощью информационных систем Д.Скотта принадлежат А.А.Воронкову.

а) постулируется первичность семантических аспектов при уточнении "наивной" схемы;

б) при семантическом уточнении "наивной" схемы используется формульная определимость алгоритмических сущностей;

в) конструктивный смысл утверждения определяется как "Σ-определимая истинность" этого утверждения в модели.

Таким образом, предлагаемый метод представляет собой попытку свести в единую схему представления о конструктивном, полученные в рамках платонистской и конструктивной математик. Ибо при этом конструктивный смысл утверждения определяется как "эффективно" проверяемое на модели высказывание об истинности рассматриваемого утверждения в этой же модели. В предлагаемом подходе будут описаны два способа соотнесения конструктивного смысла утверждениям языка. В первом случае (соответствующая семантика называется Σ-оцениваемостью) в качестве носителей "частичной" информации об истинности того или иного утверждения выступают элементы списочной надстройки рассматриваемой модели и, следовательно, для записи всей информации об истинности утверждения используются уже подмножества списков. "Эффективность" подобной кодировки знаний истинности утверждений в соответствии с идеями Σ-программирования заключается в формульной определимости (Σ-формулами) этих подмножеств. Подобная семантика языка позволяет определить адекватную ей логическую систему: выводимость утверждения в логической системе должна повлечь его Σ-оцениваемость. Поскольку Σ-оцениваемость является частным случаем классической истинности, то для построения и изучения таких логик может быть успешно использован опыт, накопленный в теории моделей. Построенную в соответствии с вышеупомянутым критерием адекватности проблемно-ориентированную логическую систему естественно назвать семантически конструктивной. Таким образом, вместо синтаксического критерия предлагается семантический критерий конструктивности логических систем. Естественно встает вопрос о сравнении этих двух критериев. Эта проблема тем более интересна, поскольку интуиционистская логика и упоминавшийся выше принцип Маркова являются одновременно синтаксически и семантически конструктивными. Более того, для семантически конструктивных логических систем оказываются верными семантические варианты D- и E-свойств. В связи с этим возникает следующий вопрос: будет ли синтаксически конструктивная система (например, у которой существует алгоритм извлечения программ из доказательств или для которой справедливы D- и E-свойства) семантически конструктивной, и обратно? Ответ, вообще говоря, отрицательный. Оказывается, можно привести пример синтаксически конструктивной логической системы, обогащение которой принципом исключенного третьего делает ее противоречивой. Очевидно, что такая система не может быть семантически конструктивной. Вместе с тем существует семантически конструктивная логическая система, для которой не имеют места синтаксические D- и E-свойства, хотя верны, как уже говорилось, семантические их аналоги. Подобная ортогональность классификационных признаков придает проблеме их соотнесения друг другу (в частности, поиску условий, при которых логическая система, удовлетворяющая одному из критериев, удовлетворяет и другому) особый интерес.

Второй способ соотнесения конструктивного смысла утверждениям языка носит название Σ-реализуемости и следует в общих чертах идее реализуемости по Клини. Только в качестве реализаций формул здесь используются не рекурсивные функции или функционалы конечных типов, а предикаты, "вычисляемые" Σ-выражениями - конструкциями языка, построенного Ю.Л.Ершовым. Подобная семантика оказалась во многих отношениях удобнее и нагляднее, чем предыдущая, хотя, как будет показано впоследствии, обе семантики в определенном смысле эквивалентны. Если первая семантика позволяет извлекать из конструктивных доказательств Σ-программы, то вторая - так называемые n-программы, т.е. конструкции более высокого порядка, поскольку аргументами n-программ могут быть вновь n-программы в том числе и Σ-программы. Заметим, что n-программы могут быть эффективно транслируемы в язык Σ-программ (что и позволяет ставить вопрос об эквивалентности двух семантик).

В данной статье будет описана только первая семантика - Σ-оцениваемость. Перейдем к точным формулировкам.

§2. L-системы

Как было отмечено в §I, в теоретическом программировании широкое применение нашли f_0-пространства Ершова: в их терминах удовлетворительно описывается денотационная семантика языков программирования. Поскольку в концепции Σ-программирования алгоритмические свойства задаются формульными описаниями, что позволяет считать Σ-фрагмент языка теории GES языком логического (предикатного) программирования, то естественно попытаться построить семантику этого фрагмента и даже всего языка (воспринимая его как язык спецификаций) также в терминах f_0-пространств. Однако нас данная семантика будет интересовать, главным образом, в связи с созданием метода извлечения Σ-программ из доказательств. Поэтому строиться она будет в более интенсиональных терминах, а именно в терминах информационных систем Скотта [II]. Учитывая то обстоятельство, что теория GES имеет дело со списками, будет использоваться списочная модификация информационной системы Скотта – L-система. Настоящий параграф посвящен описанию понятия L-системы и операций над этими конструкциями.

ОПРЕДЕЛЕНИЕ I. <u>L-система</u> – это тройка

$$\mathbf{A} = \langle D_A, Con_A, \vdash_A \rangle,$$

где D_A – непустое множество, Con_A и $\vdash_A$ – подмножества множества $S^1(D_A)$ линейных списков над D_A, удовлетворяющие следующим условиям: для каждых $\alpha, \beta \in S^1(D_A)$ и $d, d_1 \in D_A$

(i) $(d) \in Con_A$;

(ii) $\alpha \in Con_A, \beta \neq nil, |\beta| \subseteq |\alpha| \to \beta \in Con_A$ (здесь $|\beta| \subseteq |\alpha| \leftrightharpoons$ $\leftrightharpoons (\forall \gamma \in \beta)(\gamma \in \alpha)$);

(iii) $\alpha \in Con_A, cons(\alpha, d) \in \vdash_A \to cons(\alpha, d) \in Con_A$;

(iv) $d \in \alpha \to cons(\alpha, d) \in \vdash_A$;

(v) $cons(\alpha, d) \in \vdash_A$, $|\beta| = |\alpha| \to cons(\beta, d) \in \vdash_A$;

(vi) $\forall d \in \alpha (cons(\beta, d) \in \vdash_A)$, $cons(\alpha, d_1) \in \vdash_A \to cons(\beta, d_1) \in \vdash_A$.

В дальнейшем тот факт, что $cons(\alpha, d) \in \vdash_A$ будем обозначать через $\alpha \vdash_A d$.

Определим для каждой L-системы $\mathbf{A}$ множество $|\mathbf{A}|$, называемое <u>$\mathbf{A}$-областью</u>, следующим образом: для каждого $a \subseteq D_A$

$$a \in |\mathbf{A}| \leftrightharpoons \forall d \in D_A \forall \beta \in S^1(a)[\beta \in Con_A \ (\beta \vdash_A d \to d \in a)],$$

где $S^1(a)$ – множество линейных списков над a, Заметим, что для каждой L-системы A наименьшим элементом в $[A]$ будет $\emptyset$.

На L-систему можно смотреть как на некий "вычислитель", где D_A - это множество значений, которые потенциально могут поступать на вход "вычислителя" и появляться на его выходе. Элементы Con_A - это те упорядоченные совокупности значений, на которых "вычислитель" определен, т.е. вырабатывает некоторое значение. И, наконец, $\vdash_A$ можно рассматривать как механизм порождения "вычислений". Таким образом, элемент из $[A]$ - это совокупность непротиворечивых наборов значений, "замкнутая" относительно порождаемых данным "вычислителем" значений. Подобная "вычислительная" интерпретация подсказывает естественную возможность формульного определения "вычислителя" в теории GES. Об этом будет идти речь в следующем параграфе. А сейчас мы укажем на связь L-систем с f_0-пространствами [6,II].

ПРЕДЛОЖЕНИЕ I. Если A - L-система, то $[A]$ - полное f_0-пространство. Обратно, для каждого полного f_0-пространства F с наибольшим элементом можно построить L-систему A такую, что $F = [A]$ (с точностью до изоморфизма). □

Определим операции над L-системами. Пусть A и B - две L-системы. В дальнейшем нам понадобятся следующие функции:

1) $pr_0 : S^1(D_{A+B}) \to S^1(D_A)$,

$$pr_0(\alpha) \leftrightharpoons \begin{cases} nil, \text{ если } \alpha = nil \text{ или } \alpha = ((d,(nil))); \\ (d), \text{ если } \alpha = (d,nil); \\ conc\,(pr_0(tail(\alpha)), pr_0(cons(nil, head(\alpha)))) - \\ \qquad\qquad \text{в остальных случаях.} \end{cases}$$

2) $pr_1 : S^1(D_{A+B}) \to S^1(D_B)$,

$$pr_1(\alpha) \leftrightharpoons \begin{cases} nil, \text{ если } \alpha = nil \text{ или } \alpha = ((d,nil)); \\ (d), \text{ если } \alpha = ((d,(nil))); \\ conc(pr_1(tail(\alpha)), pr_1(cons(nil, head(\alpha)))) - \\ \qquad\qquad \text{в остальных случаях.} \end{cases}$$

Полагаем

а) $A \oplus B = \langle D_{A+B}, Con_{A+B}, \underset{A+B}{\vdash} \rangle$,

где а$_1$) $D_{A+B} \leftrightharpoons \{(d, nil) \mid d \in D_A\} \cup \{(d, (nil)) \mid d \in D_B\}$;

а$_2$) $\alpha \in Con_{A+B} \leftrightharpoons [(pr_0(\alpha) \in Con_A \,\&\, pr_1(\alpha) = nil) \vee (pr_1(\alpha) \in$
$\in Con_B \,\&\, pr_0(\alpha) = nil)]$;

а$_3$) $\alpha \underset{A+B}{\vdash} d \leftrightharpoons [(d = (d', nil) \,\&\, pr_0(\alpha) \neq nil \,\&$
$\& \, pr_0(\alpha) \underset{A}{\vdash} d') \vee (d = (d'', (nil)) \,\&\, pr_1(\alpha) \neq nil \quad pr_1(\alpha) \underset{B}{\vdash} d'')]$;

б) $A \otimes B = \langle D_{A\times B}, Con_{A\times B}, \underset{A\times B}{\vdash} \rangle$,

где

б$_1$) $D_{A\times B} = D_{A+B}$;

б$_2$) $\alpha \in Con_{A\times B} \leftrightharpoons [pr_0(\alpha) \in Con_A \,\&\, pr_1(\alpha) \in Con_B]$;

б$_3$) $\alpha \underset{A\times B}{\vdash} d \leftrightharpoons [(d = (d'', nil) \,\&\, pr_0(\alpha) \underset{A}{\vdash} d') \vee$
$\vee (d = (d'', (nil) \,\&\, pr_1(\alpha) \underset{B}{\vdash} d'')]$;

в) $A \ominus B \leftrightharpoons \langle D_{A\to B}, Con_{A\to B}, \underset{A\to B}{\vdash} \rangle$,

где

в$_1$) $D_{A\to B} \leftrightharpoons \{(\alpha, \beta) \mid \alpha \in Con_A, \beta \in Con_\beta\}$;

в$_2$) Пусть $\gamma \in S^1(D_{A\to B})$ и $\gamma = ((\alpha_1, \beta_1), \ldots, (\alpha_n, \beta_n))$;

где $\alpha_i \in Con_A$, $\beta_i \in Con_B$, $i = 1, \ldots, n$;

$$\gamma \in Con_{A\to B} \leftrightharpoons \forall \{i_1, \ldots, i_k\} \subseteq \{1, \ldots, n\}(\{i_1, \ldots, i_k\} \neq \emptyset \,\&$$
$$\& \, (\alpha_{i_1}, \ldots, \alpha_{i_k}) \in Con_A \Rightarrow (\beta_{i_1}, \ldots, \beta_{i_k}) \in Con_B).$$

Через $\alpha \vdash \beta$ будем обозначать тот факт, что $(\forall d \in \beta)(\alpha \vdash d)$.

в$_3$) $\gamma \underset{A\to B}{\vdash} (\alpha, \beta) = (\beta_{i_1}, \ldots, \beta_{i_k}) \underset{B}{\vdash} \beta$,

где $\{i_1, \ldots, i_k\} = \{i \mid \alpha \underset{A}{\vdash} \alpha_i\}$.

Нетрудно проверить, что операции $\oplus$, $\otimes$ и $\ominus$ корректны. Другими словами, верно следующее

ПРЕДЛОЖЕНИЕ 2. Если $\mathbb{A}$ и $\mathbb{B}$ – L-системы, то $\mathbb{A}\oplus\mathbb{B}$, $\mathbb{A}\otimes\mathbb{B}$ и $\mathbb{A}\ominus\mathbb{B}$ также L-системы.□

Пусть $\mathbb{A}$ и $\mathbb{B}$ – L-системы, $a \subseteq D_{\mathbb{A}}$, $b \subseteq D_{\mathbb{B}}$ и $f \subseteq D_{\mathbb{A}\to\mathbb{B}}$. Полагаем

$$in_{\mathbb{A}}(a) \leftrightharpoons \{(d,nil)\,|\,d \in a\},$$

$$in_{\mathbb{B}}(b) \leftrightharpoons \{(d,(nil))\,|\,d \in b\},$$

$$[a,b] \leftrightharpoons in_{\mathbb{A}}(a) \cup in_{\mathbb{B}}(b),$$

$$f[a] = \{d\,|\,\exists\alpha,\beta \in Con_{\mathbb{A}},\ \exists\gamma \in Con_{\mathbb{B}}(\alpha \in S^1(a)\ \&\ (\beta,\gamma) \in f\ \&\ \&\ \beta \vdash_{\mathbb{A}} \alpha\ \&\ \gamma \vdash_{\mathbb{B}} d\}.$$

ПРЕДЛОЖЕНИЕ 3. Если $\mathbb{A}$, $\mathbb{B}$ – L-системы, $a\subseteq D_{\mathbb{A}}$, $b \subseteq D_{\mathbb{B}}$ и $f \subseteq D_{\mathbb{A}\to\mathbb{B}}$, то $in_{\mathbb{A}}(a), in_{\mathbb{B}}(b) \in [\mathbb{A}\oplus\mathbb{B}]$ $[a,b] \in [\mathbb{A}\otimes\mathbb{B}]$ и $f[a] \in [\![\mathbb{B}]\!]$. Более того, справедливы следующие равенства (с точностью до изоморфизма): $[\![\mathbb{A}\oplus\mathbb{B}]\!] = [\![\mathbb{A}]\!]+[\![\mathbb{B}]\!]$, $[\![\mathbb{A}\otimes\mathbb{B}]\!] = [\![\mathbb{A}]\!] \times [\![\mathbb{B}]\!]$, $[\mathbb{A}\ominus\mathbb{B}] = [[\![\mathbb{A}]\!]\to[\![\mathbb{B}]\!]]$, где "+" – прямая сумма, "x" – декартово произведение и "→" – порождение пространства всех непрерывных отображений.□

Пусть $\mathbb{A} = \langle D_{\mathbb{A}}, Con_{\mathbb{A}}, \vdash_{\mathbb{A}} \rangle$ – L-система. Рассмотрим множество ${}^{\perp}D \leftrightharpoons D_{\mathbb{A}} \cup \{\perp\}$, $\perp \notin D_{\mathbb{A}}$. Определим рекурсивно функцию $pr(\perp)$: $S^1({}^{\perp}D) \to S^1(D_{\mathbb{A}})$

$$pr(\perp)(\alpha) \leftrightharpoons \begin{cases} nil, \alpha = nil \text{ или } \alpha = (\perp); \\ \alpha, \ \perp \notin \alpha; \\ conc(pr(\perp)(tail(\alpha)),\ pr(\perp)(cons(nil,head(\alpha)))) - \\ \qquad \text{в остальных случаях.} \end{cases}$$

Используя эту функцию, определим множество ${}^{\perp}Con$ и ${}^{\perp}\vdash$ следующим образом:

$$\alpha \in {}^{\perp}Con \leftrightharpoons [\alpha = (\perp) \vee (\perp \in \alpha\ \&\ pr(\perp)(\alpha) \in Con_{\mathbb{A}})],$$

$$\alpha\, {}^{\perp}\!\vdash d = [(\alpha \neq nil\ \&\ d = \perp) \vee (d \neq \perp\ \&\ pr(\perp)(\alpha) \vdash_{\mathbb{A}} d)].$$

Полагаем ${}^{\perp}\mathbb{A} \leftrightharpoons \langle {}^{\perp}D, {}^{\perp}Con, {}^{\perp}\vdash \rangle$.

ЛЕММА I. ${}^{\perp}\mathbf{A}$ – L-система и $[\mathbf{A}] \cong [{}^{\perp}\mathbf{A}] \setminus \{\emptyset\}$.□

Пусть теперь $\mathbf{M}$ – произвольное непустое множество и $\mathcal{A} = (\mathbf{A}_m)_{m\in M}$ – семейство L-систем. Рассмотрим множество

$$D^M \leftrightharpoons \langle \{\langle m,d\rangle \mid m \in M,\ d \in \mathbf{A}_m\}.$$

Для каждого $m\in M$ определим функцию $pr(m)\colon S^1(D^M) \to S^1(D_m)$ следующим образом:

$$pr(m)(\alpha) = \begin{cases} nil,\ \alpha = nil \text{ или } \alpha = (\langle n,d\rangle), \text{ где } n \neq m; \\ (d),\ \alpha = (\langle m,d\rangle); \\ conc(pr(m)(tail(\alpha)), pr(m)(cons(nil, head(\alpha)))) - \\ \qquad \text{в остальных случаях.} \end{cases}$$

Используя функции $pr(m)$, определим следующие множества:

$$\alpha \in Con^M \leftrightharpoons (\forall \langle m,d\rangle \in \alpha)(pr(m)(\alpha) \in Con_{A_m})$$

и

$$\alpha \overset{M}{\vdash} \langle m,d\rangle = pr(m)(\alpha) \underset{A_m}{\vdash} d.$$

Окончательно полагаем $\mathcal{A}^M = \langle D^M, Con^M, \overset{M}{\vdash}\rangle$.

ЛЕММА 2. $\mathcal{A}^M$ – L-система и $[\![\mathcal{A}^M]\!] = \prod_{m\in M} [\mathbf{A}_m]$.□

Пусть $\langle A,\sqsubseteq\rangle$ – некоторое полное f_0-пространство и M – произвольное множество. Рассмотрим множество $[M \xrightarrow{\sim} A]$ всех частичных непрерывных функций из M в A с частичным порядком $\leq$, где (запись $f(x){\downarrow}$ будет означать, что функция f в точке x определена):

$$f \leq g \leftrightharpoons \forall x\,(f(x){\downarrow} \to g(x){\downarrow}\ \&\ f(x) \sqsubseteq g(x)).$$

Нетрудно показать, что $\langle [M \xrightarrow{\sim} A], \leq\rangle$ – полное f_0-пространство. Из двух предыдущих лемм вытекает справедливость следующего утверждения.

ПРЕДЛОЖЕНИЕ 4. Если $\mathbf{A}$ – L-система и M – некоторое непустое множество, то

$$[M \xrightarrow{\sim} [\![\mathbf{A}]\!]] \cong [\![({}^{\perp}\mathcal{A})^M]\!],$$

где $\mathcal{A} = (\mathbf{A}_m)_{m\in M}$ и для каждого $m\in M$ $\mathbf{A}_m = \mathbf{A}$.□

В дальнейшем, если в семействе $\mathcal{A}$ каждая L-система есть $\mathbf{A}$, будем вместо $\mathcal{A}^M$ писать $\mathbf{A}^M$. В силу предложения 4 мы элемент $a\in [\![({}^{\perp}\mathbf{A})^M]\!]$ можем воспринимать как частичную функцию из M в $[\mathbf{A}]$

и через $dom(a)$ будем обозначать область определения соответствующей этому элементу частичной функции, а через $a(m)$ – значение соответствующей функции на аргументе m.

Заканчивая данный параграф, мы укажем еще на две L-системы, которые нам понадобятся в последующем:

$$\mathbb{1} \rightleftharpoons \langle \{ nil\}, \{(nil)\}, \{((nil), nil)\} \rangle,$$

и

$$M' = \langle M, S^1(M)\backslash\{nil\}, \underset{M}{\vdash} \rangle,$$

где M – непустое множество и $\alpha \vdash m \leftrightharpoons m \in \alpha$. Заметим, что $M \cong \cong [\![M']\!]\backslash\{\emptyset\}$.

§3. Σ-оцениваемость

Рассмотрим некоторую модель $\mathfrak{M}$ сигнатуры σ_0 и модель $s(\mathfrak{M}) = = \langle \mathfrak{M}, S(\mathfrak{M}) \rangle$ сигнатуры $\sigma = \sigma_0 \cup \sigma(GES)$. Пусть $\Phi(x)$ – формула языка L^σ и $\bar{x}=x_1,\dots,x_n$ – набор различных переменных, куда входят все свободные формулы Φ. Соотнесем теперь формуле Φ пару $\langle A_\Phi, \underline{cl}_\Phi(m_1,\dots,m_n) \rangle$, где A_Φ – L-система, а $\underline{cl}_\Phi(m_1,\dots,m_n) \subseteq \subseteq [\![A_\Phi]\!]$ – семейство элементов, "подтверждающих" истинность формулы Φ на модели $s(\mathfrak{M})$, где $m_1,\dots,m_n \in |s(\mathfrak{M})|$. В дальнейшем тот факт, что $a \in \underline{cl}_\Phi(m_1,\dots,m_n)$, будем обозначать как $a\ \underline{cl}\ (m_1,\dots \dots,m_n)\Phi$ и говорить, что "a есть $(m_1,\dots,m_n)$-оценка формулы Φ". Соотнесение $\Phi \mapsto \langle A_\Phi, \underline{cl}_\Phi(m_1,\dots,m_n)\rangle$ задается следующей индуктивной схемой.

1. Если $\Phi(x_1,\dots,x_n)$ – элементарная формула, то полагаем $A_\Phi = \mathbb{1}$ и $a\ \underline{cl}(m_1,\dots,m_n)\Phi \leftrightharpoons [a = \{nil\}\ \&\ s(\mathfrak{M}) \models \Phi(m_1,\dots,m_n)]$.

2. Если $\Phi = \Phi_1\ \&\ \Phi_2$, то $A_\Phi = A_{\Phi_1} \otimes A_{\Phi_2}$ и $[a,b]\ \underline{cl}(m_1,\dots \dots,m_n)\Phi \leftrightharpoons [a\ \underline{cl}(m_1,\dots,m_n)\ \Phi_1$ и $b\ \underline{cl}(m_1,\dots,m_n)\Phi_2]$.

3. Если $\Phi = \Phi_1 \vee \Phi_2$, то $A_\Phi = {}^{\perp}(\mathcal{A}^{\{\Phi_1,\Phi_2\}})$ (здесь $\mathcal{A} = (A_{\Phi_i})_{i=1}^2$)

и

$$a\ \underline{cl}(m_1,\dots,m_n)\Phi \leftrightharpoons [\exists i \in \{\Phi_1,\Phi_2\}(a(i)\downarrow)\ \&\ \forall i \in \{\Phi_1,\Phi_2\}$$
$$(a(i)\downarrow \rightarrow a(i)\ \underline{cl}(m_1,\dots,m_n)i)].$$

4. Если $\Phi = \Phi_1 \to \Phi_2$, то $A_\Phi = A_{\Phi_1} \ominus A_{\Phi_2}$ и

$$f \ \underline{cl}(m_1,\dots,m_n)\Phi \rightleftharpoons (\forall a \in [\![A_{\Phi_1}]\!])(a \ \underline{cl}(m_1,\dots,m_n)\Phi_1 \Rightarrow$$
$$\Rightarrow f[a] \ \underline{cl}(m_1,\dots,m_n)\Phi_2).$$

5. Если $\Phi = \neg\Phi_1$, то $A_\Phi = A_{\Phi_1} \ominus \mathbb{1}$ и

$$a \ \underline{cl}(m_1,\dots,m_n)\Phi \rightleftharpoons a \ \underline{cl}(m_1,\dots,m_n)(\Phi_1 \to (\mathrm{nil} = (\mathrm{nil}))).$$

Для случаев 6 и 7 предполагается, что x не входит в набор $\bar{\bar{x}} = x_1,\dots,x_n$ и $m \in |s(\mathfrak{M})|$ выступает в роли значения для x.

6. Если $\Phi = \exists x \Phi_1$, то $A_\Phi \rightleftharpoons {}^{\perp}(A_{\Phi_1}^{|s(\mathfrak{M})|})$ и

$$a \ \underline{cl}(m_1,\dots,m_n)\Phi \rightleftharpoons [\mathrm{dom}(a) \neq \emptyset \ \& \ (\forall m \in \mathrm{dom}(a))$$
$$(a(m)(m_1,\dots,m_n,m)\Phi_1].$$

7. Если $\Phi = \forall x \Phi_1$, то $A_\Phi \rightleftharpoons |s(\mathfrak{M})| \to A_{\Phi_1}$ и

$$f \ \underline{cl}(m_1,\dots,m_n) \rightleftharpoons (\forall m \in |s(\mathfrak{M})|)(f[\{m\}]\mathrm{cl}(m_1,\dots,m_n,m)\Phi_1).$$

На этом определение семантики языка L^σ заканчивается. Приступая к изучению ее свойств, отметим прежде всего следующее

ПРЕДЛОЖЕНИЕ 5. Для каждой формулы Φ область $[\![A_\Phi]\!]$ является полной решеткой. □

Из предложения следует, что для каждого множества $X \subseteq [\![A_\Phi]\!]$ существует его точная верхняя грань $\bigsqcup X$. В связи с этим интерес представляет следующее

ПРЕДЛОЖЕНИЕ 6. Пусть Φ - замкнутая формула и $X \subseteq [\![A_\Phi]\!]$. Если для каждого $a \in X$ имеет место $a \ \underline{cl} \ \Phi$, то и $\bigsqcup X \ \underline{cl} \ \Phi$. □

Будем говорить, что формула Φ языка L^σ <u>харропова</u>, если все вхождения в нее квантора $\exists$ и связок $\vee$ находятся под знаком отрицания "$\neg$". Для харроповых формул имеет место "монотонность" построенной семантики.

ПРЕДЛОЖЕНИЕ 7. Если Φ - замкнутая харропова формула, $a,b \in [\![A_\Phi]\!]$, $a \subseteq b$ и $a \ \underline{cl} \ \Phi$, то $b \ \underline{cl} \ \Phi$.

Из предложений 5 и 7 вытекает

СЛЕДСТВИЕ I. Если Φ - харропова формула $a \ \underline{cl} \ \Phi$, то $D_{A_\Phi} \ \underline{cl} \ \Phi$. □

В дальнейшем будем говорить, что замкнутая формула оцениваема в $\mathbf{s}(\mathfrak{M})$, если для некоторого $a \in [\![A_\Phi]\!]$ имеет место a cl Φ. В этом случае будем говорить также, что a является оценкой Φ или что a оценивает Φ. Открытая формула считается оцениваемой, если оцениваемо ее универсальное замыкание.

Следующая теорема говорит о наличии тесной связи между классической истинностью и оцениваемостью, что, кстати, объясняет запись "a cl Φ"(classical).

ТЕОРЕМА I. Для замкнутой формулы Φ эквивалентны следующие утверждения:

(i) $\mathbf{s}(\mathfrak{M}) \models \Phi$;

(ii) Φ оцениваема в $\mathbf{s}(\mathfrak{M})$. □

Пусть $\mathbf{s}(\mathfrak{M}) \models$ GES . В этом случае можно считать, что каждая L-система A_Φ строится внутри списочной надстройки $\mathbf{s}(\mathfrak{M})$. Следовательно, говоря о том, что a оценивает Φ, можно a рассматривать как подмножество $\mathbf{s}(\mathfrak{M})$. Все это позволяет сформулировать следующее центральное определение. Напомним, что подмножество $A \subseteq |\mathbf{s}(\mathfrak{M})|^n, n \leq 1$, называется Σ-определимым, если найдется такая Σ-формула $\Phi(x_1,\ldots,x_n)$ сигнатуры σ, что $A = \{\langle a_1,\ldots,a_n\rangle \mid \mathbf{s}(\mathfrak{M}) \models \Phi(a_1,\ldots,a_n)\}$.

ОПРЕДЕЛЕНИЕ 2. Замкнутая формула Φ сигнатуры σ называется Σ-оцениваемой в $\mathbf{s}(\mathfrak{M})$, если найдется такое Σ-определимое в $\mathbf{s}(\mathfrak{M})$ множество a, что a cl Φ. В этом случае будем также говорить, что a есть Σ-оценка Φ в $\mathbf{s}(\mathfrak{M})$ (в символах $a \, Ⓢ \, \Phi$), и если Ψ - Σ-формула, определяющая a, то называть Ψ также Σ-оценкой Φ и писать $\Psi \, Ⓢ \, \Phi$.

Для целей проектирования Σ-программ большой интерес представляет следующая

ТЕОРЕМА 2. Если $\Psi \, Ⓢ \, \forall \bar{x} \, \exists \bar{y} \, \Phi(\bar{x}, \bar{y})$, то по Ψ можно эффективно построить такую Σ-формулу Ψ', что

$$\mathbf{s}(\mathfrak{M}) \models (\Psi' \to \Phi) \,\&\, \forall \bar{x}(\exists \bar{y}\, \Phi \to \exists \bar{y}\, \Psi').\ \square$$

Из этой теоремы следует, что если мы умеем находить для некоторой спецификации задачи, заданной формулой $\Phi(\bar{x},\bar{y})$ языка L^σ, ее Σ-оценку, то тем самым мы можем указать и полную аппроксимацию формулы Φ, т.е. такую Σ-формулу Ψ', для которой выполняется заключение теоремы 2. Заметим, что предложение 6 дает нам возможность строить все более хорошие полные аппроксимации (Ψ_1 лучше

Ψ_2, если $\mathbf{s}(\mathfrak{M}) \models (\Psi_2 \to \Psi_1)$). Интерпретация этих результатов применительно к задаче проектирования Σ-программ будет дана в последующих публикациях. А сейчас вернемся к обсуждению свойств введенной семантики. Заметим, что Σ-оцениваемость в некоторых случаях совпадает с конструктивной истинностью [18]. Точнее, имеет место следующее

ПРЕДЛОЖЕНИЕ 8. Пусть $(\mathfrak{M},\nu)$ - нумерованная модель сигнатуры σ_0, которая вместе с нумерацией ν являются Σ-определимыми множествами в $\mathbf{s}(\mathfrak{M})=\langle \mathfrak{M}, \mathbf{s}^{fin}(\mathfrak{M})\rangle$. Тогда замкнутая формула Φ Σ-оцениваемая в $\mathbf{s}(\mathfrak{M})$ в том и только в том случае, когда существует ν^*-вполне перечислимое множество a такое, что $a \underline{\mathrm{cl}} \Phi$, где ν^*-нумерация $\mathbf{s}^{fin}(\mathfrak{M})$, построенная по ν. $\square$

СЛЕДСТВИЕ 2 [18]. Если $\mathfrak{M}$ - сильно конструктивная модель, Φ - замкнутая формула сигнатуры σ_0, то $\mathfrak{M} \models \Phi \Leftrightarrow \Phi$ Σ-оценивaема в $\mathbf{s}(\mathfrak{M})$. $\square$

Заканчивая параграф, отметим следующий факт, вытекающий из теоремы I и следствия I.

ПРЕДЛОЖЕНИЕ 9. Пусть Φ - замкнутая харропова формула. Тогда $\mathbf{s}(\mathfrak{M}) \models \Phi \Leftrightarrow \Phi$ Σ-оценивaема. $\square$

§4. Σ-системы

Настоящий параграф посвящен формализации понятия "конструктивная логика" в соответствии с установками §I. Сразу же отметим, что под логической системой будет пониматься тройка $\langle$ язык, класс допустимых формул, правила вывода $\rangle$. При этом требуется, чтобы указанный класс формул и правила вывода были согласованы между собой в определенном смысле. Такое понимание логической системы, в отличие от традиционного, допускает создание более широкого класса "конструктивных логик", поскольку появляется возможность манипулирования уже не одним параметром - правилами вывода (аксиомы понимаются как беспосылочные правила вывода), а двумя - классом допустимых описаний и дедуктивным механизмом. Обратим также внима -

ние читателя на семантический характер предлагаемой формализации понятия "конструктивная логика", что соответствует духу исходных установок концепции Σ-программирования, провозглашающего приоритетность семантических аспектов [I].

Пусть $s(\mathfrak{M}) = \langle \mathfrak{M}^{\sigma_0}, S(\mathfrak{M})\rangle$ - модель теории GES сигнатуры $\sigma = \sigma_0 \cup \sigma(\text{GES})$, а L^σ - язык этой же сигнатуры. Будем предполагать, что L^σ позволяет не только высказываться о свойствах объектов модели $s(\mathfrak{M})$, но и рассматривать сами формулы языка L^σ как элементы модели. Этот факт может достигаться различными способами. Например, можно считать, что в модели $\mathfrak{M}^{\sigma_0}$ имеются сорта ТЕРМ и ФОРМУЛА или, скажем, сорт ЛИТЕРА. В последнем случае формулы могут рассматриваться как элементы некого предиката $\text{FORMULA} \subseteq S(\mathfrak{M})$. Можно просто считать, что имеется некая такая кодировка формул языка L^σ списками надстройки $S(\mathfrak{M})$, что множество FORMULA соответствующих кодов является Δ-определимым. В дальнейшем, чтобы отличать формулу как синтаксический объект от ее кода (элемента $S(\mathfrak{M})$), будем использовать обозначения Φ и $\hat{\Phi}$ соответственно.

Имея в своем распоряжении кодировку $\Phi \mapsto \hat{\Phi}$, можно и правило вывода рассматривать как Δ-предикат, элементы которого являются кодами конкретных применений этого правила - конфигураций. Действительно, если $\frac{\Phi_1 \ldots \Phi_n}{\Phi}$ есть некоторая конфигурация, то список $((\hat{\Phi}_1, \ldots, \hat{\Phi}_n), \hat{\Phi})$ можно считать кодом этой конфигурации. Заметим, что если Φ - аксиома, то ее кодом в этом случае будет $(\text{nil}, \hat{\Phi})$. Похожим образом кодируются правила вывода в секвенциальном виде или правила естественного вывода (в последнем случае посылками правил вывода могут выступать сами выводы). В дальнейшем будет использован в основном секвенциальный вид исчислений.

Пусть L^σ - язык исчисления предикатов сигнатуры σ и $\mathfrak{F}$ - множество <u>всех</u> формул этого языка. Тройку $\mathcal{F} = \langle L^\sigma, F, (\text{Rule}_i)_{i=1}^n\rangle$, где $F \subseteq \mathfrak{F}$ - непустое подмножество формул и $(\text{Rule}_i)_{i=1}^n$ - конечное семейство правил вывода в виде $\frac{\Gamma_1 \mapsto \Delta_1 \ \ldots \ \Gamma_m \mapsto \Delta_m}{\Gamma \mapsto \Delta}$, где $\mapsto$ - знак секвенции, Γ_i, Δ_i, Γ, Δ - наборы формул из F, $i=1,\ldots,m$, будем называть <u>логической системой</u>. Если $\mathcal{F}$ - логическая система, то через $\vdash_{\mathcal{F}}$ будем обозначать отношение выводимости в $\mathcal{F}$. Для секвенции $\Gamma \mapsto \Delta$ будем говорить, что она оцениваема или Σ-оцениваема, если оцениваема или Σ-оцениваема формула $(\& \Gamma \mapsto \vee \Delta)$, где $\& \Gamma$ - конъюнкция формул из Γ, $\vee \Delta$ - дизъюнкция формул из Δ.

ОПРЕДЕЛЕНИЕ 3. Логическая система $\mathcal{F} = \langle L^{\sigma}, \mathbb{F}, (Rule_i)_{i=1}^{n} \rangle$ называется Σ-системой для модели $s(\mathfrak{M})$, если существует такое семейство Σ-предикатов $(P_i)_{i=1}^{n}$ в $s(\mathfrak{M})$, что для любой конфигурации $((\Gamma_1 \hat{\mapsto} \Delta_1, \ldots, \Gamma_{n_i} \hat{\mapsto} \Delta_{n_i}), \Gamma \hat{\mapsto} \Delta) \in \widehat{Rule}_i$ и любых $(a_j)_{j=1}^{n_i}$ таких, что $a_j \ \underline{cl}\ \Gamma_j \mapsto \Delta_j$, $j=1, \ldots, n_i$, имеет место

$$\{d \mid \exists d_1 \in a_1 \ldots \exists d_{n_i} \in a_{n_i} P_i(\Gamma_1 \hat{\mapsto} \Delta_1, \ldots, \Gamma_{n_i} \hat{\mapsto} \Delta_{n_i}, \Gamma \hat{\mapsto} \Delta,$$

$$d_1, \ldots, d_{n_i}, d\} \ \underline{cl}\ \Gamma \mapsto \Delta.$$

Следующая теорема связывает понятия Σ-оценивания и Σ-системы, демонстрируя их адекватность друг другу.

ТЕОРЕМА 3. Если $\mathcal{F}$ - Σ-система для $s(\mathfrak{M})$, $\Gamma \models_{\mathcal{F}} \Phi$ и каждая Ψ из Γ Σ-оцениваема, то и Φ является Σ-оцениваемой формулой в $s(\mathfrak{M})$. □

На "конструктивность" Σ-систем указывает и наличие у них семантического аналога D- и E-свойств.

ТЕОРЕМА 4. Пусть $\mathcal{F}$ - Σ-система для $s(\mathfrak{M})$. Тогда существуют две Σ-формулы $D(x,y)$ и $E(x,y)$ такие, что

(i) если Π - доказательство в $\mathcal{F}$ замкнутой формулы $(\Phi \vee \Psi)$, то $s(\mathfrak{M}) \models D(\hat{\Pi}, nil) \vee \vee D(\hat{\Pi}, (nil))$, и если $s(\mathfrak{M}) \models D(\hat{\Pi}, nil)$, то $s(\mathfrak{M}) \models \Phi$, а если $s(\mathfrak{M}) \models D(\hat{\Pi}, (nil))$, то $s(\mathfrak{M}) \models \Psi$;

(ii) если Π - доказательство в $\mathcal{F}$ замкнутой формулы $\exists x\, \Phi(x)$, то существует $m \in |s(\mathfrak{M})|$ такой, что $s(\mathfrak{M}) \models E(\hat{\Pi}, m)$ и $s(\mathfrak{M}) \models \Phi(m)$. □

В связи с этими теоремами, естественно возникает вопрос о соотнесении понятия "Σ-система" и традиционного понимания конструктивности логических систем (см. §I). Прежде всего отметим следующий факт.

ТЕОРЕМА 5. Интуиционистская логика $\underline{Int} = \langle L^{\sigma}, \mathfrak{F}, \underline{In} \rangle$ является Σ-системой для любой модели $s(\mathfrak{M})$ теории GES. □

В [I] отмечалось, что аксиомы теории GES носят конструктивный характер. Следующая теорема уточняет это высказывание. Пусть $\mathbf{F}$ – некоторая схема аксиом теории GES (если Φ – аксиома, то полагаем $\mathbf{F} = \{\Phi\}$).

ТЕОРЕМА 6. Логическая система $\langle L^{\sigma}, \mathbf{F}, \{(\text{nil}, \widehat{\mapsto\Phi}) \mid \Phi \in \mathbf{F}\}\rangle$ является Σ-системой для каждой модели теории GES.□

Из теорем 5 и 6 вытекает, что Σ-системой для каждой модели теории GES будет интуиционистская теория списочных расширений IGES. В силу предложения 9 к IGES, не нарушая ее свойств быть Σ-системой для некоторой выделенной модели $s(\mathfrak{M})$ теории GES, можно добавлять в качестве аксиом харроповы формулы (точнее, соответствующие им секвенции), истинные в $s(\mathfrak{M})$. Тем самым мы получаем возможность создавать для $s(\mathfrak{M})$ широкий класс Σ-систем. Отметим, что не нарушает свойства "быть Σ-системой для любой модели теории GES" и принцип Маркова.

Важное методологическое значение для концепции Σ-программирования имеет нижеприводимая теорема, фактически служащая еще одним обоснованием в пользу выбора класса Σ-формул в качестве основы для языка логического программирования.

Пусть $\mathcal{L} = \langle L^{\sigma}, \mathbf{\Sigma}, \underline{\text{Cl}} \rangle$, где $\mathbf{\Sigma}$ – класс Σ-формул сигнатуры σ, $\underline{\text{Cl}}$ – семейство правил вывода классической логики (с соответствующими ограничениями).

ТЕОРЕМА 7. Логическая система $\mathcal{L} = \langle L^{\sigma}, \mathbf{\Sigma}, \underline{\text{Cl}} \rangle$ является Σ-системой для любой модели теории GES.□

В связи с этой теоремой интерес представляет следующий результат. Обозначим через $\mathbf{D} \subseteq \mathfrak{F}$ множество всех формул языка L^{σ}, находящихся в дизъюнктивной нормальной форме, а через $\underline{\text{Res}}$ – дедуктивный механизм, известный под названием "метод резолюций" [20]. В $\mathbf{D}$ выделим подкласс $\mathbf{H}$ – совокупность всех хорновых дизъюнктов.

ПРЕДЛОЖЕНИЕ 10. Логическая система $\mathcal{H} = \langle L^{\sigma}, \mathbf{H}, \text{Res} \rangle$ является Σ-системой для любой модели теории GES. □

Все предыдущие результаты указывают на наличие широкого класса Σ-систем для моделей теории GES, позволяя конструировать содержательные проблемно-ориентировочные Σ-системы, в рамках которых можно достаточно удовлетворительно решать задачу синтеза Σ-программ, о чем будет идти речь в последующих публикациях.

В заключение автор считает своим приятным долгом поблагодарить С.С.Гончарова, К.Ф.Самохвалова, В.Ю.Сазонова и Ю.Г.Косарева за их критические замечания, способствовавшие существенному улучшению изложения материала данной работы.

Литература

I. ГОНЧАРОВ С.С., СВИРИДЕНКО Д.И. Σ-программирование.-В кн.: Логико-математические основы проблемы МОЗ (Вычислительные системы, вып. I07). Новосибирск, I985, с.3-29.

2. СВИРИДЕНКО Д.И. Проектирование Σ-программ. Постановка проблемы. -В кн.: Методы анализа данных (Вычислительные системы, вып. III), Новосибирск, I985, с. I08-I27.

3. ЕРШОВ Ю.Л. Теория нумераций. -М.: Наука, I977. - 4I6 с.

4. ЕРШОВ Ю.Л. Проблемы разрешимости и конструктивные модели. -М.: Наука, I980. - 4I5 с.

5. МАЛЬЦЕВ А.И. Конструктивные алгебры. I. -УМН, I96I,т.I6, № 3, с.3-60.

6. ЕРШОВ Ю.Л. Вычислимые функционалы конечных типов. - Алгебра и логика, I972, т.II, № 4, с.367-437.

7. ЕРШОВ Ю.Л. Теория А-пространств. -Алгебра и логика, I973, т.I2, №4, с.369-4I6.

8. SCOTT D.S. Continuous lattices.- In: Toposes, Algebraic Geometry and Logic, Lecture Notes in Math.- Berlin: Springer-Verlag,1972,v.274,p.97-136.

9. SCOTT D.S. Data types as lattices. - SIAM Journal of Computing,1977,v.5,p.522-587.

I0. SCOTT D.S. Lecture on a Mathematical Theory of Computation. - Oxford, Oxford University Computing Laboratory, Programming Reseach Group, PRG-19, 1981. - 148 p.

II. SCOTT D.S. Domains for denotational semantics. - In: Automata, Languages and Programming, Lecture Notes in Computer Science. Berlin, Springer-Verlag,1982,v.140,p.577-612.

I2. САЗОНОВ В.Ю., СВИРИДЕНКО Д.И. Абстрактная выводимость и теория областей. -Тезисы УП Всесоюз. конф. по мат.логике, Новосибирск, I984, с.I58.

I3. ГЕЙТИНГ А. Интуиционизм.- М.: Мир, I965. - II0 с.

I4. КЛИНИ С.К., ВЕСЛИ Р.Ю.Основания интуиционистской математики. -М.: Наука, I978. - 272 с.

I5. МАРКОВ А.А. О логике конструктивной математики. - Вестник МГУ, сер.мат., I970, №2, с.7-29.

I6. СВИРИДЕНКО Д.И. Предпосылки логического подхода к программированию. -В кн.: Философские основания научной теории, Новосибирск, Наука, I985, с.9I-I07.

I7. НЕПЕЙВОДА Н.Н. Математическая теория синтеза программ. (Обзор.) -В кн.: Синтез программ, г.Устинов, I985, с.4-8.

18. ВОРОНКОВ А.А. Конструктивная семантика для теории моделей. -Тезисы Всесоюз. конфер. по прикладной логике, Новосибирск, 1985, с. 42-44.

19. ВОРОНКОВ А.А., СВИРИДЕНКО Д.И. О конструктивных логиках. -Тезисы Всесоюз. конфер. по прикладной логике, Новосибирск,1985, с. 44-46.

20. ЧЕНЬ Ч., ЛИ Р. Математическая логика и автоматическое доказательство теорем. -М.: Наука, 1983.

Поступила в ред.-изд.отд.
22 января 1986 года