71:00-1/66-2

Д.28 12.99 - 3943/05

Санкт-Петербургский Институт информатики и автоматизации Российской Академии наук

На правах рукописи

Нестеров Вячеслав Михайлович

Твинные арифметики и их применение в методах и алгоритмах двустороннего интервального оценивания

05.13.16. – Применение вычислительной техники, математического моделирования и математических методов в научных исследованиях

Диссертация на соискание ученой степени доктора физико-математических наук

Санкт-Петербург

1999

Содержание

Введение

История развития вычислительной математики уходит корнями в глубокую древность. Вместе с тем особенно бурно развиваться эта область начала вместе с развитием вычислительных машин. Компьютеры позволили реализовать сложные вычислительные алгоритмы. Особая значимость вычислительной математики становится понятной, если принять во внимание те многочисленные практические приложения, в которых она используется.

Вычислительные алгоритмы, где бы и с какой целью они ни применялись, объединяет следующая схема использования. На вход алгоритма подаются входные данные, которые по существу являются целыми или действительными числами. Результатом работы алгоритма является другой (искомый) набор чисел. Ниже нас будут интересовать вычислительные алгоритмы, оперирующие с действительными числами. Практически все такие алгоритмы на выходе выдают не точные решения задачи, а значения, в том или ином смысле близкие к искомым. Отметим несколько причин возникновения погрешностей.

Первая причина – принципиальная неточность алгоритма. В частности, итерационные методы выдают одно за одним значения, приближающие искомое, как правило его не достигая. К примеру, метод Ньютона выдает последовательные приближения корня уравнения.

Вторая причина – невозможность точной реализации алгоритма на цифровой вычислительной машине. Даже если мы имеем точный алгоритм, не всегда можно точно вычислить результат. Пример – вычисление корней квадратного уравнения. Соответствующая формула является абсолютно точной, однако если корни уравнения иррациональны, мы можем их вычислить лишь с некоторой точностью (во всяком случае, используя традиционную арифметику на традиционной ЭВМ). Можно сказать, что источником погрешности является необходимость округления промежуточных и конечных результатов до чисел, представимых на данной машине.

Третья причина погрешностей, возникающих при численных рассчетах, – неточность входных данных. Значительная часть алгоритмов, особенно в приложениях, оперирует не с точными входными данными, а с некоторыми приближениями. Такие приближения могут возникать, например, при снятии измерений физическими приборами, при обработке экспертных знаний, как результат работы других численных алгоритмов и т. д. Получая результат работы алгоритма, мы должны в первую очередь интересоваться тем, насколько этот результат близок к точному решению математической задачи, которую решает наш алгоритм.

Существует множество методов оценки качества полученного решения. Можно вспомнить методы, основанные на оценке константы Липшица для реализуемой функции, методы анализа сходимости последовательных приближений и т. д. Все они, как правило, представляют из себя оценки отличия уже полученного приближения от искомого значения. Такие методы обладают рядом недостатков. Пожалуй, основным является тот недостаток, что не для каждого алгоритма, в принципе, можно оценить погрешность приближения искомого результата. Если же такую оценку можно произвести, то зачастую затраченные вычислительные ресурсы во много раз превышают те, которые были израсходованы на само получение приближения. Вспомним также, что для любой вычислительной машины легко построить алгоритмы, которые только за счет погрешностей округления будут выдавать результат, сколь угодно далеко отстоящий от искомого.

Для того, чтобы можно было строить вычислительные алгоритмы, которые *одновременно* с получением результата выдают и гарантированную оценку погрешности, была высказана идея интервальных вычислений. По свидетельству некоторых исследователей, эта идея зародилась достаточно давно, но в зрелом виде она оформилась к началу 60-х годов и была высказана в монографии Р. Э. Мура [116] (см. также [88]).

Основное содержание идеи интервальных вычислений следующее: производить обработку не чисел, а интервалов, гарантированно содержащих соответствующие точечные значения. В самом простом случае можно взять обычный числовой алгоритм. Он состоит из элементарных операций вроде арифметических операций, вычисления элементарных функций, проверок условий и т. п. Эти операции над числами следует заменить на операции над интервалами таким образом, чтобы все промежуточные результаты числового алгоритма принадлежали соответствующим интервалам – промежуточным результатам интервального алгоритма. В результате работы такого алгоритма получается набор интервалов, каждый из которых (если, конечно, преобразование алгоритма выполнено корректно) гарантированно содержит соответствующий результат исходного числового алгоритма.

Все три источника погрешностей числовых алгоритмов, указанные выше, могут быть учтены с помощью интервального подхода. Вместо приближенных значений, получаемых неинтервальными алгоритмами, их интервальные аналоги на каждом шаге имеют дело с интервалами, гарантированно содержащими точное значение. Погрешности округления также учитываюся – границы интервала округляются в сторону его расширения. Неточные входные данные могут быть заданы соответствующими интервалами.

К сожалению, прямой перевод алгоритмов в интервальную форму редко приводит к удовлетворительным результатам. Можно выделить две группы проблем, возникающих на этом пути. Во-первых, корректный прямой перевод не всегда возможен, например, из-за невсюду определенных интервальных операций (деление на интервал, содержащий ноль). Во-вторых, полученные интервальные алгоритмы часто не обеспечивают необходимую точность, т. е. вычисленные ими интервалы слишком широки. По этим причинам развитие интервальных методов пошло по пути создания специфических алгоритмов, специально ориентированных на работу с интервалами. Часто такие алгоритмы создавались по образу и подобию неинтервальных алгоритмов, но не простой заменой действительных переменных на интервальные переменные и точечных операций на интервальные операции, а более серьезными трансформациями.

Как уже отмечалось, историю развития интервального подхода принято отсчитывать от начала 60-х годов. Сперва наиболее популярным был термин "интервальный анализ". Позже область применения интервальных методов столь расширилась, что вошел в употребление термин "интервальная математика", подчеркивающий универсальность подхода. Пика расцвета рассматриваемая область вычислительной математики достигла, вероятно, к середине 80-х годов. После этого интенсивность исследований стабилизировалась и находится на достаточно высоком уровне.

Развитие интервального подхода наблюдалось и наблюдается в разных направлениях. Очень много сделано для его применения к решению задач линейной алгебры. Большие успехи достигнуты в развитии методов решения нелинейных алгебраических и дифференциальных уравнений. Многое сделано для развития математических основ интервальной математики, для оценки сложностных аспектов получаемых алгоритмов. На русском языке написаны две монографии по основам интервального подхода [9] и [32] и переведена основополагающая книга [1]. Хотя эти книги написаны уже достаточно давно (особенно в сравнении с быстрым развитием описываемой области математики), в них можно найти необходимые начальные сведения по рассматриваемой теме.

В последнее время получил распространение термин "надежные вычисления". Он включает в себя не только интервальные вычисления, но и другие разделы математики, обеспечивающие абсолютную надежность результатов, однако именно интервальные вычисления играют здесь основную роль. Преимуществом интервальных алгоритмов является гарантированность получаемых результатов. Из-за этого всегда особое внимание интервальным методам уделялось в прикладных исследованиях. Интервальные методы получили свое развитие в автоматическом управлении, оптимизации [4], [46], [72], экономике [79], теории экспертных систем [11], других прикладных областях. Хороший обзор приложений дается в сборнике [87] и статьях [55], [96].

Естественно, не могли пройти мимо новой области вычислительной математики и специалисты по созданию программного и даже аппаратного обеспечения [34], [103]. Были созданы специальные языки с интервальными типами данных [47], [48], [50], библиотеки численных методов [69], [70], самое разнообразное математическое и программное обеспечение для реализации интервальных алгоритмов. Были предложены специальные аппаратные средства [150], [151]. Появилось несколько программных систем, в которых интервальные типы данных и интервальные вычисления занимают центральное место [76], [140], [153]. Очень хороший обзор раннего этапа развития интервальных программ дан в работе [34]. Более современный обзор дан в работе [150]. Учебник по одному из языков программирования интервальных алгоритмов был издан недавно в России [10]. Много информации по специализированным библиотекам интервальных вычислений может быть найдено, например, в книгах [69] и [70].

Фундаментальной задачей интервального анализа является задача вычисления (оценки) множества значений функции при том, что известны числовые интервалы, которым принадлежат значения аргументов. Этой задаче будет посвящена значительная часть настоящей работы. Можно сказать, что задача оценки множества значений функции является определяющей при решении широкого спектра задач интервальными методами. Оценка функции входит составной частью в большое количество алгоритмов интервальной математики.

Известные методы интервальных вычислений позволяют решить эту задачу. При этом для достижения необходимой точности таких вычислений были предложены различные усовершенствования базовых методов, такие как использование обобщенных интервальных арифметик и методов глобальной оптимизации, связанных с дроблением исходного интервала. Дальнейшее совершенствование этих методов представляет собой актуальную задачу.

Существующие интервальные методы позволяют локализовать действительное значение или указать внешнюю оценку интервала. В то же время задача поиска других приближений (например, внутреннего, двустороннего) неизвестного интервала изучена слабо и является актуальной задачей, которая будет нами исследована. Решение этой задачи предусматривает построение новых математических объектов и структур, эффективно служащих для локализации действительных значений и интервалов. Одним из обобщений классической интервальной арифметики являются арифметики твинов, которые и составляют основной предмет настоящей работы.

Другой актуальной задачей является построение вычислительных алгоритмов, оперирующих с такими объектами, и позволяющих получать гарантированные оценки решений для различных задач численного анализа.

Методы и алгоритмы, первоначально предназначавшиеся для решения классических задач численного анализа, могут быть перенесены в другие области. Применение интервальных и твинных методов и алгоритмов в других областях знаний, например, в искусственном интеллекте, также является весьма актуальным.

Одновременно с разработкой вычислительных алгоритмов всегда должны рассматриваться методы их компьютерной реализации. Разработка принципов такой реализации а также создание соответствующего программного обеспечения составляют еще одну актуальную задачу.

Таким образом можно сделать вывод, что развитие интервального подхода является актуальным. Исходя из этого сформулируем основную цель диссертационной работы: совершенствование интервального подхода к решению вычислительных задач и разработка теоретических основ твинных методов их решения.

Для достижения поставленной цели в работе решены следующие теоретические и практические задачи:

- введены, исследованы и разработаны теоретические основы обобщенных интервальных и твинных арифметик, для широкого класса задач построены гарантированные алгоритмы, использующие предлагаемые структуры, получены оценки точности и сложности таких алгоритмов;

- разработана методология применения предложенных методов к решению конкретных задач из области вычислительной математики и искусственного интеллекта;

- исследованы вопросы компьютерной реализации твинных алгоритмов и построена программная инструментальная система для этой цели.

Первая глава диссертации носит постановочный характер. Цель первой главы – дать определения, поставить задачи и сформулировать результаты, являющиеся отправной точкой исследований. В отличие от остальных глав, первая глава не содержит новых результатов, полученных автором.

В первых трех параграфах дано определение стандартной интервальной арифметики, приведены ее основные свойства, сформулирована задача внешней оценки множества значений функции. В четвертом параграфе приведены известные результаты, касающиеся точности решения задачи внешней оценки и изложены некоторые идеи, лежащие в основе современных методов глобальной оптимизации.

Далее рассматриваются основные недостатки, присущие стандартной интервальной арифметике. Они могут быть разделены на три категории: незамкнутость интервальной арифметики, неточность получаемых с ее помощью оценок, невозможность работы с бесконечными интервалами. Рассмотрены модификации интервальной арифметики. Описаны обобщения, оперирующие с бесконечными интервалами и обобщения, дающие возможность в некоторых случаях повысить точность оценки множества значений функции путем рассмотрения дополнительной информации, связанной с монотонностью функции. При этом удается принять во внимание взаимозависимость подвыражений выражения, задающего рассматриваемую функцию. В первой главе рассмотрена также машинная арифметика и кратко освещены вопросы компьютерной реализации. Более подробно эти вопросы обсуждены в пятой главе. Кратко излагаются идеи апостериорного интервального анализа Ю. В. Матиясевича.

В последних двух параграфах первой главы рассмотрены алгебраические свойства известных арифметик и сложностные аспекты решения некоторых задач интервальными методами.

Вторая глава содержит основные теоретические результаты, полученные автором. Основная цель главы – построить теорию твинных вычислений. Для ее достижения предложены арифметика твинов и арифметика направленных твинов, исследованы их свойства. Доказана теорема о монотонности по включению естественного твинного расширения рациональных функций. Проведено сравнение с ранее известными твинными структурами. Обоснованы принципы компьютерной реализации твинной арифметики, сохраняющей монотонность по включению.

В первом параграфе получено обобщение результата С. Маркова об оценке множества значений функции средствами интервальной арифметики с нестандартными операциями. Обобщение проведено в двух направлениях: ослаблено условие монотонности и результат доказан для более общего класса функций. Предложен графический формализм, который полезен при работе с конкретными функциями.

Далее дается определение арифметики твинов. Приводится обоснование подхода, состоящее в необходимости механизма двусторонней оценки неизвестных интервалов. Доказывается корректность введенной твинной арифметики и рассматриваются ее свойства. С одной стороны, они обосновывают использование твинов для двусторонней оценки, с другой – создают базу для дальнейшего развития теории и приложений твинной арифметики. После рассмотрения твиной арифметики проводится дальнейшие обобщения твинного подхода. Вводятся направленные твины, которые, в известном смысле, так же соотносятся с обычными твинами, как интервалы Каухера с обычными интервалами. Рассматриваются свойства направленных твинов. Приводятся примеры двусторонней оценки множества значений функции с использованием разных введенных твинных структур. Кратко рассмотрено еще одно обобщение твинной арифметики – твинная арифметика с бесконечными твинами.

Пятый параграф второй главы посвящен сравнению разработанного подхода с известным подходом Гарденеса и Трепата [63], которые также рассматривают твинные структуры. Последние три параграфа второй главы посвящены рассмотрению трех важных моментов, связанных с введенным твинным пространством. Рассмотрено соотношение твинных методов с известными методами глобальной оптимизации, рассмотрены вопросы машинной твинной арифметики, которых мы дополнительно коснемся в пятой главе, доказаны алгебраические свойства введенных твинных пространств.

Цель третьей главы диссертационной работы – исследовать вопросы точности и сложности твинных вычислений. В главе на основе полученного результата о линейной зависимости точности двусторонней оценки от величины интервала, предлагается алгоритм внутренней оценки, основанный на дроблении исходного интервала. Доказывается существование алгоритма с полиномиальным временем выполнения, который получает оценку твинной рациональной функции, причем эта оценка почти всегда точна для достаточно узких входных твинов.

Показано, что вычислительная сложность твинных алгоритмов не превосходит вычислительной сложности их интервальных аналогов.

Глава начинается с рассмотрения некоторых содержательных аспектов твинных вычислений. Далее доказывается теорема о твинных вычислениях функции с однократным вхождением каждой переменной. Доказанная теорема позволяет усовершенствовать алгоритм внутренней оценки, предложенный ранее.

Во втором параграфе третьей главы даются базовые определения точности внутренней и двусторонней оценок. Рассмотрены также точности оценок для базовых твинных операций. Полученные соотношения приводят к выводам относительно областей эффективной применимости твинных вычислений. Получена оценка точности твинных вычислений для функции, удовлетворяющей условию Липшица. Эта оценка позволяет построить эффективный алгоритм внутренней оценки, который описан в виде общей схемы с последующим разъяснением ключевых шагов.

Далее в третьей главе дается еще одно определение твинной арифметики, являющейся обобщением той, которая была рассмотрена ранее. Согласно новому определению, внутренний интервал твина не может быть пустым, но может представлять из себя интервал, у которого правая граница меньше левой. За счет такого обобщения алгоритм внутренней оценки, изложенный ранее, может быть применен к более широкому классу функций из-за снятия одного из входных ограничений алгоритма.

В последнем, шестом, параграфе третьей главы рассматривается вычислительная сложность твинных вычислений. Показано, что оценка вычислительной сложности твинных алгоритмов сводится к аналогичной задаче для интервальных алгоритмов.

Четвертая глава посвящена применениям твинных вычислений. Наряду со второй главой она представляет собой основу диссертационной работы. Цель главы – исследовать общие принципы использования твинов и, используя разработанную теорию, указать пути решения ряда конкретных задач. Основным результатом главы является методология использования твинов в некоторых математических задачах и задачах искусственного интеллекта.

В первом параграфе формулируется основной методологический аспект использования твинов как представителей величин, имеющих интервальную природу. Указывается на аналогию с представлением вещественного значения интервалом.

Далее описывается ряд применений твинов для решения чисто математических задач (задачи двусторонней оценки множеств различной природы, двусторонней оценки решения линейной системы интервальных уравнений, задачи анализа статической системы) и задач искусственного интеллекта. Последние представляются нам наиболее перспективной областью использования полученных результатов. Помимо общих принципов использования твинов для представления интервалов сформулирован ряд конкретных задач и разработаны методы их решения. Рассмотрены, в частности, задача о представлении динамики курса акций, задача обработки экспериментальных данных о ядерных взаимодействиях, задача представления геометрии реального мира. Особое внимание уделено использованию твинов при представлении экспертных знаний в экспертных системах.

Короткий заключительный параграф четвертой главы посвящен основам модальной математики. Показано, что основная задача твинных вычислений допускает переформулировку в терминах задачи установления истинности формулы модальной математики. Такая переформулировка дает возможность установить связь между твинными вычислениями и модальным подходом, что обещает привести к взаимному переносу твинных методов в модальную математику и наоборот.

В пятой главе описана разработанная автором компьютерная реализация твинной арифметики и направленной твинной арифметики. Основная цель главы – описать предложенные автором общие принципы реализации твинных вычислений.

Реализация выполнена в среде языка Pascal-XSC – расширения стандартного Паскаля, специально предназначенного для верифицированных вычислений, в частности включающего в себя все необходимые средства для работы с интервалами.

В начале пятой главы раскрыты преимущества и недостатки высокоуровневой реализации интервальных и твинных арифметик. Сделан вывод о том, в каких случаях предлагаемый путь реализации оказывается эффективным и имеет преимущества перед другими подходами.

Второй параграф главы содержит краткий обзор языка Pascal-XSC, причем основное место уделено тем его средствам, которые представляют для нас особый интерес в рамках настоящей работы. Описаны принципы реализации языка, раскрыты его особенности, которые не имеют места в стандартном Паскале. Далее описываются специальные средства языка Pascal-XSC, обеспечивающие обработку числовых данных с высокой точностью, и, главное, с верификацией результатов. Эта часть особенно важна для главы в целом. Изложенные в ней средства использованы автором при разработке твинной арифметики.

Отдельный параграф пятой главы посвящен архитектуре расширений языка Pascal-XSC. В рассмотрение включены стандартные модули для работы с интервалами, поставляемые вместе с транслятором языка, созданные автором модули для расширенных интервальных арифметик, не рассматриваемые в данной работе, и два модуля для работы с твинами и с направленными твинами, являющиеся основным предметом рассмотрения главы. Описываются типы данных, основные группы операций, процедур и функций, включенные в эти два модуля.

В заключении работы сделан вывод по работе и кратко сформулированы основные ее результаты.

В двух приложениях к основному тексту диссертации собраны описания типов и заголовки описаний операций, процедур и функций, реализованных в модулях, поддерживающих работу с твинами и направленными твинами.

Глава 1. Интервальные арифметики и оценка множества значений функции

1.1 Интервальная арифметика

Введем следующие обозначения. Пусть R – множество всех вещественных чисел. Под интервалом $[a, b]$, $a \in R$, $b \in R$, $a \leq b$ будем понимать множество

$$[a, b] = \{x \mid x \in R \,\&\, a \leq x \leq b\}.$$

Множество всех интервалов обозначим $I(R)$. В обозначениях и определениях, данных в главе 1, мы следуем традициям, принятым в интервальной математике (см., например, [1], [9]).

Интервалы назовем равными, если у них попарно равны левые и правые границы. Унарные арифметические операции над интервалами определяются следующим образом. Пусть $\diamond$ – некоторая унарная операция (например, унарный минус) на множестве вещественных чисел, $A \in I(R)$. Тогда

$$\diamond A = \{\diamond a \mid a \in A\}. \tag{1.1}$$

Бинарные операции над интервалами определяются так. Пусть $\circ$ – бинарная операция на множестве вещественных чисел (например, сложение, вычитание, умножение, деление), $A, B \in I(R)$. Тогда

$$A \circ B = \{a \circ b \mid a \in A,\ b \in B\}. \tag{1.2}$$

Отметим, что в качестве аргументов унарных и бинарных интервальных операций допустимы только такие интервалы, для всех точек которых определены операции $\diamond a$ и $a \circ b$, и множества в правых частях равенств (1.1) и (1.2) являются интервалами.

Аналогичным образом могут быть определены интервальные функции (операции) с n аргументами. Так, для вещественной функции $f(x_1, \ldots, x_n)$ определим интервальную функцию $F(X_1, \ldots, X_n)$ следующим образом

$$F(X_1, \ldots, X_n) = \{y \mid \exists x_1 \in X_1, \ldots, \exists x_n \in X_n : \ f(x_1, \ldots, x_n) = y\}. \quad (1.3)$$

Часто функции (операции), для которых интервальные аналоги определены в соответствии с формулами (1.1)–(1.3), будем называть "базовыми".

Обозначим $A = [a^-, a^+]$, $B = [b^-, b^+]$. Тогда явные формулы для арифметических операций выглядят следующим образом:

$$\begin{aligned}
-A &= [-a^+, -a^-]; && (1.4)\\
A + B &= [a^- + b^-, a^+ + b^+]; && (1.5)\\
A - B &= [a^- - b^+, a^+ - b^-]; && (1.6)\\
A \cdot B &= [\min\{a^-b^-, a^-b^+, a^+b^-, a^+b^+\}, \\
&\qquad \max\{a^-b^-, a^-b^+, a^+b^-, a^+b^+\}]; && (1.7)\\
A/B &= A \cdot [1/b^+, 1/b^-], \ 0 \notin B. && (1.8)
\end{aligned}$$

Согласно формулам (1.1)–(1.3) могут быть введены другие базовые операции. В частности, элементарные функции от интервалов (например, ln, exp, sin) могут быть введены согласно формуле (1.1). Для этого необходимо, чтобы множество в правой части формулы было интервалом, что имеет место, например, для функций, заданных и непрерывных на всем рассматриваемом интервале.

Вырожденный интервал $[a, a]$ отождествим с вещественным числом a. Таким образом, $R \subseteq I(R)$. Нетрудно проверить, что если интервалы A и B вырождены, т. е. их левые границы совпадают с правыми, то интервальные арифметические операции над ними совпадают с обычными операциями над вещественными числами. Таким образом, интервальная арифметика есть обобщение вещественной арифметики; в одном арифметическом выражении можно смешивать интервалы и вещественные числа произвольным образом. Интервальные операции (функции), полученные в соответствии с формулами (1.1) – (1.3), называются интервальными расширениями соответствующих вещественых операций (функций).

Под множеством n-мерных интервалов $I(R^n)$ будем понимать множество

$$\{I_1 \times \cdots \times I_n\} \mid I_i \in I(R)\}.$$

Его элементами являются n-мерные параллелепипеды. Операции над n-мерными интервалами определяются покомпонентно через операции над одномерными интервалами.

Шириной интервала $A = [a^-, a^+]$ назовем величину $a^+ - a^-$. Ширину интервала будем обозначать $|A|$. Ширина многомерного интервала определяется как максимальная ширина его компонент

$$|A_1 \times \cdots \times A_n| = \max_i |A_i|.$$

В качестве синонима термина "ширина интервала" будем использовать термин "длина интервала". Оба эти термина встречаются в литературе.

1.2 Свойства интервальной арифметики

Отметим некоторые легко проверяемые свойства интервальной арифметики (см., например, [9]). Ниже, если не будет оговорено особо, под интервалом понимаем одномерный интервал.

Интервальные умножение и сложение коммутативны и ассоциативны, т. е. для $A, B, C \in I(R)$ имеем

$$\begin{aligned} A+B &= B+A, \\ A+(B+C) &= (A+B)+C, \\ A\cdot B &= B\cdot A, \\ A\cdot(B\cdot C) &= (A\cdot B)\cdot C. \end{aligned}$$

Роли нуля и единицы играют интервалы $[0,0]$ и $[1,1]$, которые отождествляются с вещественными числами 0 и 1.

Для интервала A, если он не вырожден в точку, не существует обратных элементов по сложению и умножению, то есть таких B и C, что $A+B=0$, $A\cdot C=1$.

Закон дистрибутивности в общем случае не выполняется, т. е. соотношение

$$A\cdot(B+C) = A\cdot B + A\cdot C$$

не всегда имеет место. Всегда выполняется более слабое свойство субдистрибутивности

$$A\cdot(B+C) \subseteq A\cdot B + A\cdot C.$$

Фундаментальным свойством интервальных вычислений является монотонность по включению. Для всех операций $\diamond$ и $\circ$, определение которых соответствует (1.1) и (1.2), и для интервалов $A\subseteq C$ и $B\subseteq D$ имеем

$$\begin{aligned} \diamond A &\subseteq \diamond C, \\ A\circ B &\subseteq C\circ D. \end{aligned}$$

Из свойства монотонности по включению легко вытекает следующая теорема, иногда называемая основной теоремой интервальной арифметики [9], [116].

Теорема 1.1. Пусть $F(X_1, \ldots, X_n)$ – выражение от n интервальных переменных X_i, построенное с применением операций, определенных согласно формулам (1.1)–(1.3). Иными словами, $F(X_1, \ldots, X_n)$ – конечная комбинация интервалов $X_1, \ldots, X_n$ и интервальных констант, соединенных интервальными операциями. Тогда из $Y_i \subseteq Z_i$, $i = 1, \ldots, n$, следует

$$F(Y_1, \ldots, Y_n) \subseteq F(Z_1, \ldots, Z_n)$$

при любом наборе интервальных чисел $Z_1, \ldots, Z_n$, для которого интервальные операции в выражении имеют смысл (т. е., например, не встречается деление на интервал, содержащий ноль).

1.3 Задача внешней оценки множества значений функции

Пусть дана вещественная функция $f(x_1, \ldots, x_n)$, $x_i \in R$. Для такой функции можно рассматривать интервальную функцию $\overline{F}(X_1, \ldots, X_n)$, $X_i \in I(R)$ согласно определению

$$\overline{F}(X_1, \ldots, X_n) = \{y \mid \exists x_1 \in X_1, \ldots, \exists x_n \in X_n : \; y = f(x_1, \ldots, x_n)\}.$$

Для любого набора интервалов-аргументов функция $\overline{F}$ выдает множество значений, которые принимает функция f, если каждый из ее аргументов x_i независимо пробегает все значения из соответствующего интервала X_i. Функцию $\overline{F}$ часто называют объединенным интервальным расширением функции f. Она действительно является расширением, потому что для вырожденных (одноточечных) интервалов $X_i = x_i$

$$\overline{F}(X_1, \ldots, X_n) = \overline{F}(x_1, \ldots, x_n) = f(x_1, \ldots, x_n).$$

Ограничимся рассмотрением случая, когда функция f непрерывна и всюду определена в рассматриваемой области, и, следовательно, множества $\overline{F}(X_1, \ldots, X_n)$ являются интервалами. В дальнейшем часто не будем особо оговаривать тот факт, что все значения переменных $x_1, \ldots, x_n$ и соответствующие им интервалы берутся из области $\Omega \subseteq R^n$, в которой рассматриваемая функция всюду определена и непрерывна.

Функцию $F(X_1, \ldots, X_n)$, $X_i \in I(R)$ назовем внешней оценкой для функции f, если

$$\forall i \; x_i \in X_i \;\Rightarrow\; f(x_1, \ldots, x_n) \in F(X_1, \ldots, X_n). \tag{1.9}$$

Если к тому же $F(x_1, \ldots, x_n) = f(x_1, \ldots, x_n)$, то F обычно называют интервальным расширением f.

Для внешней оценки F имеем

$$\overline{F}(X_1, \ldots, X_n) \subseteq F(X_1, \ldots, X_n). \tag{1.10}$$

Исходная функция f, вообще говоря, может быть задана разными способами. Мы будем считать, что f задана в виде символьного выражения, составленного из простейших "базовых" функций, таких как арифметические операции и элементарные функции. Из Теоремы 1.1 следует, что функция F, то есть внешняя оценка для данной функции f, в частности, может быть получена путем замены в аналитическом выражении для f всех вещественных переменных на интервальные и всех базовых функций на их интервальные расширения. Интервальное расширение, полученное таким образом, называется естественным интервальным расширением вещественной функции.

1.4 Точность оценки

Формула (1.10) показывает, что естественное интервальное расширение обеспечивает оценку множества значений функции. Встает вопрос, какова точность оценки. Другими словами в этом разделе нас будет интересовать вопрос о величине

$$|F(X_1, \dots, X_n)| - |\overline{F}(X_1, \dots, X_n)| = w(F, X_1, \dots, X_n),$$

а также ее соотношение с величинами $|X_1|, \dots, |X_n|$. Заметим, что точность интервальной оценки может быть очень низкой. Другими словами, естественное интервальное расширение может существенно отличаться от объединенного интервального расширения. Приведем хрестоматийный пример.

Пример 1.1. Пусть $f(x) = x - x$. Вычислим для $F(X) = X - X$ значение $F([0,1])$. Получаем $F([0,1]) = [0,1] - [0,1] = [-1,1]$. В то же время $\overline{F}([0,1]) = 0$.

Мы видим, что оценка множества значений функции, полученная применением естественного интервального расширения, может по размерам в бесконечное число раз превышать истинное множество значений.

Также нетрудно видеть, что значение оценки, полученной через естественное интервальное расширение, существенно зависит от конкретного аналитического представления данной функции.

Пример 1.2. [18] Пусть $f(x) = x(10 - x)$, $g(x) = 10x - x^2$, $X = [3,5]$.

$$F([3,5]) = [3,5] \cdot (10 - [3,5]) = [3,5] \cdot [5,7] = [15,35];$$

$$G([3,5]) = 10 \cdot [3,5] - [3,5]^2 = [30,50] - [9,25] = [5,41].$$

При этом $\overline{F}([3,5]) = \overline{G}([3,5]) = [21,25]$.

Хорошо известен тот факт, что источником погрешности при интервальном оценивании множества значений функции является взаимозависимость подвыражений рассматриваемого выражения [102], [110], [121]. Для получения более точных оценок такую зависимость необходимо как-то учитывать, чего стандартная интервальная арифметика сделать не может. При применении стандартной интервальной арифметики к функции вида $f(h_1(x), h_2(x))$ оценки, полученные для h_1 и h_2, используются как независимые, и полученный результат будет таким же, как и при оценивании функции $f(h_1(x), h_2(y))$, хотя на самом деле множества значений в двух случаях могут сколь угодно сильно различаться [102].

Один из случаев, когда интервальная оценка дает точное множество значений функции, описывается следующей теоремой.

Теорема 1.2. [116] Пусть $f(x_1, \ldots, x_n)$ – рациональное выражение, в котором каждая переменная встречается не более одного раза. $F(X_1, \ldots, X_n)$ – естественное интервальное расширение $f(x_1, \ldots, x_n)$. Тогда

$$\overline{F}(X_1, \ldots, X_n) = F(X_1, \ldots, X_n)$$

для любого набора $(X_1, \ldots, X_n)$ такого, что $F(X_1, \ldots, X_n)$ имеет смысл.

Теорема легко обобщается с функций, заданных рациональными выражениями, на функции, в записи которых участвуют любые функции, для которых интервальные расширения получены в соответствии с формулами (1.1)–(1.3).

Насколько же отличается интервальная оценка, полученная с использованием естественного расширения, от действительного множества значений функции? Ответ на этот вопрос дает следующая известная теорема (см. [1], [9]).

Теорема 1.3. Пусть $f(x_1, \ldots, x_n)$ – вещественная функция n переменных. Пусть $F(X_1, \ldots, X_n)$, $X_i \in I(R)$, ее естественное интервальное расширение, определенное при всех $X_i \subseteq Y_i$. Пусть функция $\tilde{f}(x_1^{(1)}, \ldots, x_1^{(k_1)}, \ldots, x_n^{(1)}, \ldots, x_n^{(k_n)})$ – выражение для функции f с явным указанием на все вхождения всех переменных. Пусть также для каждой переменной $x_i^{(j)}$ при произвольном выборе фиксированных значений остальных аргументов функция $\tilde{f}$ удовлетворяет условию Липшица на всем множестве Y_i. Тогда существует такая константа $\gamma > 0$, что

$$|F(X_1, \dots, X_n)| - |\overline{F}(X_1, \dots, X_n)| \leq \gamma |X_1 \times \dots \times X_n|,$$

для всех $X_i \subseteq Y_i$, где $|X_1 \times \dots \times X_n| = \max\{|X_1|, \dots, |X_n|\}$.

Теорема 1.3 утверждает, что при принятых условиях точность оценки линейно зависит от размеров интервала, на котором оценивается множество значений функции. Существуют специально построенные расширения, которые обеспечивают квадратичную оценку [1], [9]. Такими расширениями являются центрированная форма [142] и MV-форма [51], [141].

Основываясь на Теореме 1.3, легко сделать вывод, что, в принципе, оценку множества значений функции можно сделать сколь угодно точной. Для этого достаточно разбить исходные интервалы на малые подынтервалы и оценить функцию на каждом подынтервале в отдельности. Объем вычислений при этом сильно возрастает. К счастью, посредством специальных приемов объем вычислений удается сократить [43], [89], [159]. Алгоритмы, полученные на этом пути, получили название алгоритмов глобальной оптимизации. Хороший современный обзор таких алгоритмов содержится в работе [89].

Пусть мы оцениваем функцию f на интервале I. Будем раздельно искать значения глобального минимума и глобального максимума функции. Вторая задача сводится к первой рассмотрением функции $-f$. Пусть, используя интервальные методы, например, вычисляя естественное интервальное расширение $F(J)$, мы можем оценить $\overline{F}(J)$ для любого $J \subseteq I$. Чем меньше размер J, тем меньше $F(J)$ отличается от $\overline{F}(J)$. Разбиваем исходный интервал I на части, например, пополам. Находим $F(I_1)$ и $F(I_2)$. Поместим I_1 и I_2 в список L, причем в голове списка будет находиться интервал I' с наименьшим значением верхней границы $F(I')$. Это значение и будет являться текущей верхней оценкой глобального минимума F на I.

Шаг алгоритма включает в себя извлечение находящегося в голове списка элемента, деление его на две равные половины и вычислении верхних оценок минимумов на этих подынтервалах. Эффективность метода базируется на том, что в первую очередь рассматриваются те подынтервалы исходного интервала I, на которых нахождение глобального минимума наиболее вероятно.

Существуют дополнительные соображения, которые позволяют увеличить эффективность алгоритма за счет удаления из списка подлежащих оценке подынтервалов, на которых гарантированно не достигается глобальный экстремум. На таких соображениях базируются, например, тест средней точки, тест на монотонность, тест на выпуклость и т. д. Теория интервальных алгоритмов глобальной оптимизации к настоящему моменту хорошо развита [59], [72], [78], [143].

1.5 Ограниченность стандартной интервальной арифметики

Стандартная интервальная арифметика имеет ряд недостатков. Первый уже отмечен выше. Он заключается в неточности получаемых оценок множества значений функции.

Второй недостаток – незамкнутость интервальной арифметики. Даже если ограничиться четырьмя арифметическими действиями, мы видим, что не ко всем парам интервалов они могут быть применены: невозможно деление на интервал, содержащий ноль.

В ряде приложений ощущается необходимость работы с бесконечными интервалами. Например, хотелось бы для некоторых функций уметь вычислять оценки множеств вида $\{f(x) \mid x \in [a, +\infty)\}$. Такое желание приводит к введению бесконечных интервалов, на которых мы остановимся в следующем разделе.

1.6 Модификации стандартной интервальной арифметики

1.6.1 Арифметики с бесконечными интервалами

1.6.1.1 Арифметика бесконечных внутренних интервалов

Может быть введено несколько различных арифметик, оперирующих с бесконечными интервалами. Сперва кратко рассмотрим арифметику, которая иногда называется арифметикой бесконечных внутренних интервалов [25], [52], [71], [83], [110]. Мы также будем ее называть арифметикой бесконечных интервалов первого рода. Эта арифметика используется, например, в теории дифференцирования и интегрирования интервальных функций.

Множество R пополняется двумя специальными элементами $-\infty$ и $+\infty$. Будем считать, что $-\infty < a < +\infty$, $a \in R$.

$$R^{\times} = R \cup \{-\infty, +\infty\}.$$

Вместо множества обычных интервалов $I(R)$ рассматривается множество $I(R^{\times})$, которое получается из $I(R)$ путем добавления в него элементов следующего вида

$$\begin{aligned}
[-\infty, b] &= \{x \mid x \in R \;\&\; x \leq b\} \cup \{-\infty\}; \\
[a, +\infty] &= \{x \mid x \in R \;\&\; x \geq a\} \cup \{+\infty\}; \\
[-\infty, -\infty] &= -\infty; \\
[+\infty, +\infty] &= +\infty; \\
[-\infty, +\infty] &= R^{\times}.
\end{aligned}$$

Таким образом

$$I(R^{\times}) = \{[a, b] \mid a, b \in R^{\times} \ \& \ a \leq b\}.$$

В качестве n-мерных интервалов в рамках данной арифметики рассматривается прямое произведение n одномерных интервалов. Множество таких n-мерных интервалов обозначим через $I(R^{\times n})$.

$$I(R^{\times n}) = \{[a_1, b_1] \times \cdots \times [a_n, b_n] \mid a_i, b_i \in R^{\times} \ \& \ a_i \leq b_i\}.$$

Арифметические операции на множестве $I(R^{\times})$ определяются на основе формул (1.1)–(1.2) путем обобщения обычных арифметических операций над интервалами (формулы (1.4)–(1.8)). Достаточно лишь формально принять следующие равенства с символами $-\infty$ и $+\infty$.

Для формулы (1.4):

$$\begin{aligned} -(-\infty) &= +\infty, \\ -(+\infty) &= -\infty; \end{aligned}$$

для формулы (1.5):

$$\begin{aligned} & a + (+\infty) = +\infty + a = +\infty + (+\infty) = +\infty, \\ & a + (-\infty) = -\infty + a = -\infty + (-\infty) = -\infty, \\ & -\infty + (+\infty) = +\infty + (-\infty) = \\ & \qquad = \begin{cases} -\infty, & \text{если это левая граница интервала,} \\ +\infty, & \text{если это правая граница интервала;} \end{cases} \end{aligned}$$

для формулы (1.6):

$$\begin{aligned} & a - (+\infty) = -\infty - a = -\infty - (+\infty) = -\infty, \\ & a - (-\infty) = +\infty - a = +\infty - (-\infty) = +\infty, \\ & -\infty - (-\infty) = +\infty - (+\infty) = \\ & \qquad = \begin{cases} -\infty, & \text{если это левая граница интервала,} \\ +\infty, & \text{если это правая граница интервала;} \end{cases} \end{aligned}$$

для формулы (1.7):

$$a \cdot (+\infty) = +\infty \cdot a = \begin{cases} -\infty, & \text{если } a < 0, \\ 0, & \text{если } a = 0, \\ +\infty, & \text{если } a > 0; \end{cases}$$

$$a \cdot (-\infty) = -\infty \cdot a = \begin{cases} +\infty, & \text{если } a < 0, \\ 0, & \text{если } a = 0, \\ -\infty, & \text{если } a > 0; \end{cases}$$

$$-\infty \cdot (-\infty) = +\infty \cdot (+\infty) = +\infty;$$

$$-\infty \cdot (+\infty) = +\infty \cdot (-\infty) = -\infty;$$

для формулы (1.8):

$$1/(+\infty) = 1/(-\infty) = 0.$$

При применении формулы (1.8) мы, как и прежде, предполагаем, что интервал-делитель не содержит нуля.

Заметим, что нетрудно ввести обобщения интервальных расширений элементарных функций, например,

$$\ln([a, b]) = [\ln(a), \ln(b)], \quad a, b \geq 0,$$

где $\ln(0) = -\infty$, $\ln(+\infty) = +\infty$.

Введенное обобщение позволяет работать с бесконечными интервалами, в частности производить оценку множества значений функции при предположении, что некоторые входные переменные изменяют свои значения в бесконечных интервалах.

К сожалению, введенное множество интервалов неполно относительно четырех арифметических действий. По-прежнему невозможно деление на интервал, содержащий 0.

1.6.1.2 Арифметика Кахана

Другая интервальная арифметика, оперирующая бесконечными интервалами, известна под названием арифметики Кахана. Мы также будем называть ее арифметикой бесконечных интервалов второго рода [106], [110]. Она первоначально была введена исходя из желания сделать пространство интервалов замкнутым относительно четырех арифметических действий. Основная идея этой арифметики состоит в том, что интервалы содержат бесконечность внутри себя.

Дополним множество всех вещественных чисел R единственным элементом ∞ и обозначим $R^+ = R \cup \{\infty\}$. Обобщенными интервалами второго рода будем называть множества $[a, b]$, $a, b \in R^+$. Включим также в множество бесконечных интервалов второго рода пустой интервал, который обозначим Λ. Множество одномерных интервалов второго рода обозначим через $I(R^+)$. Так же как и ранее, множеством n-мерных интервалов является прямое произведение одномерных интервалов.

Введенные обозначения имеют следующий смысл. Если $a, b \in R$, $a \leq b$, то $[a, b]$ – обычный интервал. В качестве $[\infty, b]$ и $[a, \infty]$ обозначаются, соответственно, бесконечные интервалы $\{x \mid x \leq b\}$ и $\{x \mid x \geq a\}$. Множества таких интервалов мы обозначим, соответственно, $I_l(R^+)$ и $I_r(R^+)$. Положим также $[\infty, \infty] = R$. Если $a > b$, тогда $[a, b] = \{x \mid x \geq b \ \vee \ x \leq a\}$. Множество таких интервалов обозначим через $I_d(R^+)$.

Определим арифметические операции над бесконечными интервалами второго рода. Для $x \in R^+$ будем полагать, что $x + \infty = \infty + x = x - \infty = \infty - x = -(\infty) = 1/0 = \infty$, $1/\infty = 0$. Также положим, что присутствие $x \in R^+$ в записи неравенства предполагает, что $x \neq \infty$.

Для $x < 0$ и $y > 0$ или $y = \infty$ будем полагать, что

$$\min(\ldots, x \cdot \infty, \ldots) = \max(\ldots, y \cdot \infty, \ldots) = \infty.$$

Пусть $A = [a^-, a^+]$, $B = [b^-, b^+]$.

$$A + B = \begin{cases} [a^- + b^-, a^+ + b^+], & \text{если } (A, B \in I(R) \cup I_l(R^+) \cup I_r(R^+)) \vee \\ & \quad ((A \in I(R) \vee B \in I(R)) \; \& \; a^- + b^- > a^+ + b^+), \\ [\infty, \infty], & \text{в остальных случаях;} \end{cases}$$

$$-A = [-a^+, -a^-];$$

$$A - B = A + (-B);$$

$$A \cdot B = \begin{cases} [\min(a^-b^-, a^-b^+, a^+b^-, a^+b^+), \max(a^-b^-, a^-b^+, a^+b^-, a^+b^+)], & \\ & \text{если } A, B \in I(R) \cup I_r(R^+), \\ -(-A \cdot B) & \text{если } A \in I_l(R^+), \\ -(A \cdot (-B)) & \text{если } B \in I_l(R^+), \\ A' \cdot B \cup A'' \cdot B & \text{если } A \in I_d(R^+),\ A = A' \cup A'', \\ A \cdot B' \cup A \cdot B'' & \text{если } B \in I_d(R^+),\ B = B' \cup B''; \end{cases}$$

$$1/A = A^{-1} = \begin{cases} \Lambda, & \text{если } a = b = 0, \\ [1/a^+, 1/a^-] & \text{в остальных случаях;} \end{cases}$$

$$A/B = A \cdot B^{-1}.$$

Отметим важное свойство введенной арифметики. Она полна относительно арифметических действий. В частности, возможно деление на интервал, содержащий 0.

1.6.2 Обобщения интервальной арифметики для учета монотонности

1.6.2.1 Арифметика Каухера, метод Гарденеса и Трепата

Одно из наиболее известных обобщений интервальной арифметики известно как арифметика Каухера [61], [62], [63], [65], [83], [110], [112]. Множеством обобщенных интервалов в рамках этой арифметики является множество $I^*(R) = \{[a, b] \mid a, b \in R\}$.

Арифметические операции на множестве $I^*(R)$ могут быть заданы несколькими эквивалентными способами. Мы определим их следующим образом. Пусть $I_1 = [a, b] \in I^*(R)$, $I_2 = [c, d] \in I^*(R)$, $\alpha \in \{-, +\}$. Сложение

$$[a, b] + [c, d] = [a + c, b + d].$$

Умножение. Сперва определим несколько вспомогательных функций.

$$\phi([a, b], \alpha) = \begin{cases} a, & \text{если } \alpha = +; \\ b, & \text{если } \alpha = -; \end{cases}$$

$$\psi([a, b], \alpha) = \begin{cases} b, & \text{если } \alpha = +; \\ a, & \text{если } \alpha = -. \end{cases}$$

$$I_0^*(R) \subseteq I^*(R),\ [a, b] \in I_0^*(R) \Leftrightarrow ab \leq 0.$$

$$\sigma([a, b]) = \begin{cases} +, & \text{если } 0 \leq a \ \&\ 0 \leq b; \\ -, & \text{если } a \leq 0 \ \&\ b \leq 0 \ \&\ [a, b] \neq [0, 0]. \end{cases}$$

$$\theta([a, b]) = \begin{cases} +, & \text{если } a \leq b; \\ -, & \text{если } a > b. \end{cases}$$

Умножение определяется следующей формулой:

$$I_1 \cdot I_2 = \begin{cases} [\phi(I_1, \sigma(I_2)) \cdot \phi(I_2, \sigma(I_1)), \psi(I_1, \sigma(I_2)) \cdot \psi(I_2, \sigma(I_1))], \\ \qquad\qquad\qquad \text{если } I_1, I_2 \in I^*(R) \setminus I_0^*(R); \\ [\psi(I_1, \sigma(I_1)\theta(I_2)) \cdot \phi(I_2, \sigma(I_1)), \psi(I_1, \sigma(I_1)\theta(I_2)) \cdot \psi(I_2, \sigma(I_1))], \\ \qquad\qquad\qquad \text{если } I_1 \in I^*(R) \setminus I_0^*(R),\ I_2 \in I_0^*(R); \\ [\phi(I_1, \sigma(I_2)) \cdot \psi(I_2, \sigma(I_2)\theta(I_1)), \psi(I_1, \sigma(I_2)) \cdot \psi(I_2, \sigma(I_2)\theta(I_1))], \\ \qquad\qquad\qquad \text{если } I_2 \in I^*(R) \setminus I_0^*(R),\ I_1 \in I_0^*(R); \\ [\min(ad, cb), \max(ab, cd)], \ \text{если } I_1, I_2 \in I_0^*(R),\ \theta(I_1) = \theta(I_2) = +; \\ [\max(ab, cd), \min(ad, bc)], \ \text{если } I_1, I_2 \in I_0^*(R),\ \theta(I_1) = \theta(I_2) = -; \\ 0, \qquad \text{если } I_1, I_2 \in I_0^*(R),\ \theta(I_1) \neq \theta(I_2). \end{cases}$$

Вычитание

$$-[a, b] = (-1) \cdot [a, b] = [-b, -a];$$

$$[a, b] - [c, d] = [a, b] + (-1) \cdot [c, d] = [a - d, b - c].$$

Деление

$$1/[a, b] = [1/b, 1/a],\ ab > 0;$$

$$[a, b]/[c, d] = [a, b] \cdot (1/[c, d]) = [a, b] \cdot [1/d, 1/c],\ cd > 0.$$

Многомерным обобщенным интервалом в смысле арифметики Каухера, как обычно, назовем прямое произведение соответствующего количества одномерных интервалов.

Отметим некоторые свойства арифметики Каухера.

$$I_1 + I_2 = I_2 + I_1;$$

$$I_1 \cdot I_2 = I_2 \cdot I_1;$$

$$(I_1 + I_2) + I_3 = I_1 + (I_2 + I_3);$$

$$(I_1 \cdot I_2) \cdot I_3 = I_1 \cdot (I_2 \cdot I_3).$$

Обобщенные интервалы $X = [0,0] = 0$ и $Y = [1,1] = 1$ являются, соответственно, единственным нейтральным элементом для операции сложения и единственным нейтральным элементом для операции умножения.

Крайне важным свойством арифметики Каухера, которое не имеет места в обычной интервальной арифметике, является следующее свойство. Каждый элемент $I \in I(R^*)$ имеет единственный обратный элемент относительно операции сложения. Каждый элемент $I \in I(R^*) \setminus I_0$ имеет единственный обратный элемент относительно операции умножения. Здесь $I_0 = \{[a,b] \mid a \leq 0 \leq b \vee b \leq 0 \leq a\}$.

На арифметике Каухера базируется исторически первый метод оценки множества значений функции, учитывающий дополнительную информацию, связанную с монотонностью. Этот метод был предложен Гарденесом и Трепатом [61], [62]. Его сущность выражена в Теореме 1.4. Для $[a,b] \in I^*(R)$ введем обозначения:

$$\mathrm{dual}([a,b]) = ([b,a]),$$

$$\mathrm{prop}([a,b]) = \begin{cases} [a,b], & \text{если } a \leq b; \\ [b,a], & \text{если } b < a. \end{cases}$$

Будем считать, что $I_1 = [a,b] \subseteq I_2 = [c,d]$ если $c \leq a$ и $b \leq d$.

Теорема 1.4. Пусть $F(X,A)$ – рациональная интервальная функция, имеющая несколько вхождений переменной A. Переменные X и A – обобщенные интервалы в смысле арифметики Каухера. Пусть $F(X,A',A'')$ – выражение для F, непосредственно содержащее вхождения $A' = (A'_1,\dots,A'_p)$, $A'' = (A''_1,\dots,A''_q)$ параметра A. Пусть соответствующая вещественная функция $f(x,a',a'')$, $a' = (a'_1,\dots,a'_p)$, $a'' = (a''_1,\dots,a''_q)$ непрерывна на $\mathrm{prop}(X) \times \mathrm{prop}(A') \times \mathrm{prop}(A'')$. Предположим, что $f(x,a',a'')$ безусловно монотонно возрастает по любой переменной из a' и безусловно монотонно убывает по любой переменной из a'' на $\mathrm{prop}(X) \times \mathrm{prop}(A') \times \mathrm{prop}(A'')$.

В этих предположениях верны следующие утверждения:
a) если $f(x, a)$ безусловно монотонно возрастает по a на $(\mathrm{prop}(X) \times \mathrm{prop}(A))$, тогда
если $A = \mathrm{prop}(A)$, то

$$F(X, A) \supseteq F(X, A', \mathrm{dual}\, A'') = \bigvee_{a \in A} F(X, a),$$

и если $\mathrm{dual}(A) = \mathrm{prop}(A)$, то

$$F(X, A) \subseteq F(X, A', \mathrm{dual}(A'')) = \bigwedge_{a \in \mathrm{dual}(A)} F(X, a);$$

b) если $f(x, a)$ безусловно монотонно убывает по a на $(\mathrm{prop}(X) \times \mathrm{prop}(A))$, тогда
если $A = \mathrm{prop}(A)$, то

$$F(X, A) \supseteq F(X, \mathrm{dual}\, A', A'') = \bigvee_{a \in A} F(X, a);$$

и если $\mathrm{dual}(A) = \mathrm{prop}(A)$, то

$$F(X, A) \supseteq F(X, \mathrm{dual}(A'), A'') = \bigwedge_{a \in \mathrm{dual}(A)} F(X, a).$$

Использованы следующие обозначения:

$$\bigvee\{[a_i, b_i] \mid [a_i, b_i] \in I^*(R), i \in \Omega\} = [\inf_{\leq}\{a_i \mid i \in \Omega\}, \sup_{\leq}\{b_i \mid i \in \Omega\}];$$

$$\bigwedge\{[a_i, b_i] \mid [a_i, b_i] \in I^*(R), i \in \Omega\} = [\sup_{\leq}\{a_i \mid i \in \Omega\}, \inf_{\leq}\{b_i \mid i \in \Omega\}].$$

По сравнению с источниками [61], [62], [63] мы несколько изменили обозначения, унифицировав их с нотацией, принятой в настоящей работе.

1.6.2.2 Направленная интервальная арифметика

Направленная интервальная арифметика введена и исследована Марковым [110], [111], [112]. Под направленным интервалом понимается пара (I, α), $I \in I(R)$, $\alpha \in \{-,+\}$, $I(R)$ – множество обычных интервалов. Множество направленных интервалов обозначим за $I^{**}(R)$. Это множество вместе с введенной ниже арифметикой по существу эквивалентно рассмотренной арифметике Каухера. В ряде случаев представление, основанное на направленных интервалах, использовать проще и удобнее, поэтому мы приведем ниже определения и некоторые свойства направленной интервальной арифметики.

Прежде, чем рассмотреть собственно арифметику, дадим несколько предварительных определений. Направленный интервал $([a,b],\alpha)$ будем для простоты обозначать $[a,b;\alpha]$. Пусть $I_1 = [a,b;\alpha]$, $I_2 = [c,d;\beta]$. Для элементов множества $\{-,+\}$ будем формально полагать, что $- \cdot - = + \cdot + = +$ и $- \cdot + = + \cdot - = -$.

Положим

$$\gamma = \begin{cases} \alpha, & \text{если } \alpha = \beta, \\ \alpha, & \text{если } \alpha = -\beta \ \& \ |I_1| > |I_2|, \\ \beta, & \text{если } \alpha = -\beta \ \& \ |I_1| < |I_2|, \\ +, & \text{если } \alpha = -\beta \ \& \ |I_1| = |I_2|. \end{cases}$$

$$I_1 + I_2 = \begin{cases} [a+c, b+d, \gamma], & \text{если } \alpha = \beta, \\ [a+d, b+c, \gamma], & \text{если } \alpha \neq \beta \ \& \ |I_1| \geq |I_2|, \\ [b+c, a+d, \gamma], & \text{если } \alpha \neq \beta \ \& \ |I_1| < |I_2|. \end{cases}$$

$$I_1 - I_2 = \begin{cases} [a-d, b-c, \gamma], & \text{если } \alpha = \beta, \\ [a-c, b-d, \gamma], & \text{если } \alpha \neq \beta \ \& \ |I_1| \geq |I_2|, \\ [b-d, a-c, \gamma], & \text{если } \alpha \neq \beta \ \& \ |I_1| < |I_2|. \end{cases}$$

Заметим, что и в случае сложения, и в случае вычитания направление результирующего интервала γ определяется по одной и той же формуле.

Определим

$$\chi([a,b;\alpha]) = \begin{cases} a/b, & \text{если } \alpha = +, \\ b/a, & \text{если } \alpha = -. \end{cases}$$

$$\tau = \begin{cases} +, & \text{если } \alpha\chi(I_2) + \beta\chi(I_1) \geq 0, \\ -, & \text{если } \alpha\chi(I_2) + \beta\chi(I_1) < 0. \end{cases}$$

Для $I_1, I_2 \in I^{**}(R) \setminus \{0\}$

$$I_1 \cdot I_2 = \begin{cases} ([a,b] \cdot [c,d], \tau), & \text{если } \alpha\beta = +, \\ [bd, ac; \tau], & \text{если } (\alpha > 0 \;\&\; \beta < 0 \;\&\; \chi(I_1) \leq \chi(I_2)) \vee \\ & \qquad (\alpha < 0 \;\&\; \beta > 0 \;\&\; \chi(I_1) > \chi(I_2)), \\ [ac, bd; \tau], & \text{если } (\alpha > 0 \;\&\; \beta < 0 \;\&\; \chi(I_1) > \chi(I_2)) \vee \\ & \qquad (\alpha < 0 \;\&\; \beta > 0 \;\&\; \chi(I_1) \leq \chi(I_2)). \end{cases}$$

Для $I_1 \in I^{**}(R) \setminus \{0\}$, $I_2 \in I^{**}(R)$, $0 \notin I_2$

$$I_1 / I_2 = \begin{cases} ([a,b] / [c,d], \tau), & \text{если } \alpha\beta = +, \\ [b/c, a/d; \tau], & \text{если } (\alpha > 0 \;\&\; \beta < 0 \;\&\; \chi(I_1) \leq \chi(I_2)) \vee \\ & \qquad (\alpha < 0 \;\&\; \beta > 0 \;\&\; \chi(I_1) > \chi(I_2)), \\ [a/d, b/c; \tau], & \text{если } (\alpha > 0 \;\&\; \beta < 0 \;\&\; \chi(I_1) > \chi(I_2)) \vee \\ & \qquad (\alpha < 0 \;\&\; \beta > 0 \;\&\; \chi(I_1) \leq \chi(I_2)). \end{cases}$$

Оценки множества значений функции, использующие арифметику направленных интервалов, базируются на следующей теореме [112].

Теорема 1.5. Пусть функции $f(x)$, $g(x)$ непрерывны и монотонны, $X = ([a,b], \alpha) \in I^{**}(R)$. Определим

$$f(X) = \begin{cases} ([f(a), f(b)], \alpha), & \text{если } f \text{ возрастает;} \\ ([f(b), f(a)], -\alpha), & \text{если } f \text{ убывает.} \end{cases}$$

Тогда

a) если функция $f+g$ – монотонна, то $(f+g)(X)=f(X)+g(X)$;

b) если функция $f-g$ – монотонна, то $(f-g)(X)=f(X)-g(X)_{-}$;

Если, кроме того, функции f и g не равны нулю нигде на рассматриваемом интервале, то

c) если функция fg монотонна, то $(fg)(X)=f(X)_{\sigma(g(X))}\cdot g(X)_{\sigma(f(X))}$;

d) если функция f/g монотонна, то $(f/g)(X)=f(X)_{\sigma(g(X))}/g(X)_{-\sigma(f(X))}$. Приняты следующие обозначения: если $X=([a,b],\alpha)$, то $X_{-}=([a,b],-\alpha)$, $X_{+}=X$; $\sigma(X)=+$, если $a>0$ и $\sigma(X)=-$, если $b<0$.

1.6.2.3 Арифметика с нестандартными операциями

В своих работах С. Марков рассматривает также пополнение стандартной интервальной арифметики некоторыми дополнительными операциями (см. [110], [111], [112]). Эти операции появились из следующих соображений. Пусть $I_1=[a,b;+]\in I^{**}(R)$, $I_2=[c,d;-]\in I^{**}(R)$. Тогда $[a,b]\circ^{-}[c,d]=p(I_1\circ I_2)$, где для $I\in I^{**}(R)$ $p(I)$ обозначает обычный интервал, границы которого совпадают с границами направленного интервала I. Введение операций $+^{-}$, $-^{-}$, $\cdot^{-}$, $/^{-}$ приводит к упрощению ряда формул. Кроме того, ряд утверждений, полученных для направленных интервалов, может быть перенесен в обычную интервальную арифметику.

Пусть $I_1=[a,b]\in I(R)$, $I_2=[c,d]\in I(R)$. Дополнительные операции $+^{-},\cdot^{-},-^{-},/^{-}$ вводятся согласно определениям:

$$I_1+^{-}I_2=[\min(a+d,b+c),\max(a+d,b+c)];$$

$$I_1-^{-}I_2=I_1+^{-}(-I_2);$$

$$I_1 \cdot^- I_2 = \begin{cases} [\min(ad,bc), \max(ad,bc)], & \text{если } 0 \notin [a,b], 0 \notin [c,d], ac > 0; \\ [\min(ac,bd), \max(ac,bd)], & \text{если } 0 \notin [a,b], 0 \notin [c,d], ac < 0; \\ [ac, ad], & \text{если } a > 0, c \le 0 \le d; \\ [bd, bc], & \text{если } b < 0, c \le 0 \le d; \\ [ac, bc], & \text{если } a \le 0 \le b, c > 0; \\ [bd, ad], & \text{если } a \le 0 \le b, d < 0; \\ [\max(ad,bc), \min(ac,bd)], & \text{если } 0 \in [a,b], 0 \in [c,d]. \end{cases}$$

$$I_1 /^- I_2 = I_1 \cdot^- (1/I_2).$$

Введенные операции позволяют доказать следующую теорему, дающую формулы для точной оценки множеств значений монотонных функций.

Теорема 1.6. Пусть функции f и g непрерывны и монотонны на рассматриваемом интервале T, $X \subseteq T$. Обозначим

$$\tau(f,T) = \begin{cases} +, & \text{если } f \text{ возрастает на } T; \\ -, & \text{если } f \text{ убывает на } T. \end{cases}$$

Если функция $h_1 = f + g$ также монотонна на T, то

$$\overline{H}_1(X) = \begin{cases} \overline{F}(X) + \overline{G}(X), & \text{если } \tau(f,T) = \tau(g,T); \\ \overline{F}(X) +^- \overline{G}(X), & \text{если } \tau(f,T) = -\tau(g,T). \end{cases}$$

Если функция $h_2 = f - g$ монотонна на T, то

$$\overline{H}_2(X) = \begin{cases} \overline{F}(X) - \overline{G}(X), & \text{если } \tau(f,T) = -\tau(g,T); \\ \overline{F}(X) -^- \overline{G}(X), & \text{если } \tau(f,T) = \tau(g,T). \end{cases}$$

Если функция $h_3 = f \cdot g$ монотонна на T и функции f и g не меняют знак, то

$$\overline{H}_3(X) = \begin{cases} \overline{F}(X) \cdot \overline{G}(X), & \text{если } \tau(|f|,T) = \tau(|g|,T); \\ \overline{F}(X) \cdot^- \overline{G}(X), & \text{если } \tau(|f|,T) = -\tau(|g|,T). \end{cases}$$

Если функция $h_4 = f/g$ монотонна на T и функции f и g не меняют знак, то

$$\overline{H}_4(X) = \begin{cases} \overline{F}(X)/\overline{G}(X), & \text{если } \tau(|f|,T) = -\tau(|g|,T); \\ \overline{F}(X)/^{-}\overline{G}(X), & \text{если } \tau(|f|,T) = \tau(|g|,T). \end{cases}$$

В работе [56] делается попытка ввести арифметику, которая оперирует с направленными интервалами подобно рассмотренной арифметике на множестве $I^{**}(R)$ и, кроме того, содержит дополнительные операции $+^{-}$, $-^{-}$, $\cdot^{-}$, $/^{-}$. Такая арифметика представляет, как нам кажется, чисто теоретический интерес.

1.6.3 Машинная арифметика

Все, что было сказано ранее о различных интервальных арифметиках, всецело касалось лишь теоретического их рассмотрения. В то же время интервальные арифметики создаются в конечном итоге для практических целей вычислительной математики. При программировании интервальных алгоритмов мы сталкиваемся со спецификой представления чисел в цифровых вычислительных машинах и вынуждены рассматривать не те арифметики, которые введены исходя из теоретических требований, а те, которые годятся для прямой реализации на ЭВМ.

Основная специфика обработки чисел на компьютере состоит в том, что на ЭВМ представимы не все вещественные числа, а лишь конечное их подмножество. Необходимость так или иначе представить любое вещественное число в ЭВМ существует, естественно, не только в интервальных вычислениях, но в контексте наших задач качество такого представления особенно важно из-за необходимости сохранить гарантированность результатов при переходе от теоретических алгоритмов к их конкретной реализации.

Важность переноса алгоритмов на конкретные вычислительные машины была ясна исследователям с момента зарождения интервальных вычислений. Впервые принципы организации интервальной машинной арифметики были изложены в монографии [116]. Введение в проблему также содержится в книге [1], см. также [34]. Основы теории были созданы на ранних этапах развития интервальных вычислений. Они описаны в работах [29], [41], [42], [67]. Из более современных источников отметим работы [104], [118], [161], [162].

Систему обозначений, которая будет употребляться в этом параграфе, мы частично заимствовали из [1].

Вещественные числа в компьютере чаще всего представляются в форме с плавающей точкой

$$x = m \cdot b^e,$$

где m – мантисса, b – основание степени, e – порядок. В традиционных машинах как правило $b = 2$, а мантисса располагется в интервале $[0.5, 1)$. Множество машинных чисел обозначим R_M. Это множество конечно, в нем существуют наибольший и наименьший элементы $r_{\min}$ и $r_{\max}$. Проблема выхода промежуточных и конечных результатов алгоритма за границы интервала $[r_{\min}, r_{\max}]$ не специфична для интервальных вычислений, она настолько же характерна для неинтервальных алгоритмов, насколько и для интервальных; мы не будем ее здесь касаться, считая, что рассматриваемые нами значения достаточно далеки от $r_{\min}$ и $r_{\max}$. Обозначим $R_r = \{x \in R \mid r_{\min} < x < r_{\max}\}$.

Для аппроксимации числа $x \in R_r$ числом $\tilde{x} \in R_M$ будем применять отображение $\mathrm{fl}(x) : R_r \to R_M$. Это отображение будем называть округлением, если выполнено свойство монотонности

$$x \leq y \Rightarrow \mathrm{fl}(x) \leq \mathrm{fl}(y). \tag{1.11}$$

Для функции округления весьма важными как с теоретической, так и с практической точек зрения являются два свойства:

$$x \in R_M \Rightarrow \mathrm{fl}(x) = x,$$

$$x \in R \Rightarrow \forall y \in R_M \ (y \neq \mathrm{fl}(x) \Rightarrow |y - x| \geq |\mathrm{fl}(x) - x|).$$

Если округление удовлетворяет последнему свойству, то говорят, что оно является “округлением к ближайшему”.

В интервальных вычислениях речь обычно идет о направленных округлениях. Рассматривается два вида округлений – округление вниз и округление вверх. Соответствующие функции будем обозначать $\mathrm{fl}_\downarrow$ и $\mathrm{fl}_\uparrow$. Отображение $\mathrm{fl}_\downarrow$, являющееся округлением, будем называть округлением вниз, если выполняется

$$x \in R_r \Rightarrow \mathrm{fl}_\downarrow(x) \leq x;$$

и аналогично отображение $\mathrm{fl}_\uparrow$, являющееся округлением, будем называть округлением вверх, если выполняется

$$x \in R_r \Rightarrow \mathrm{fl}_\uparrow(x) \geq x.$$

Функция округления на множестве вещественных чисел естественным образом индуцирует функцию округления на множестве стандартных интервалов $I(R)$

$$\mathrm{fl}([a, b]) = [\mathrm{fl}(a), \mathrm{fl}(b)].$$

При этом произвольный вещественный интервал округляется до машинного интервала, т. е. интервала, принадлежащего множеству

$$I(R_M) = \{[a, b] \mid a, b \in R_M\}.$$

В интервальных вычислениях, основанных на классической интервальной арифметике, обычно рассматривают округление интервала, результатом которого является объемлющий интервал. Для этого необходимо иметь два направленных округления на множестве действительных чисел, одно из которых ($\mathrm{fl}_\downarrow$) является округлением вниз, другое ($\mathrm{fl}_\uparrow$) – округлением вверх.

На множестве интервалов индуцируется функция округления $\mathrm{fl}_\updownarrow$

$$\mathrm{fl}_\updownarrow(I) \supseteq I = [a, b];$$

$$\mathrm{fl}_\updownarrow(I) = [\mathrm{fl}_\downarrow(a), \mathrm{fl}_\uparrow(b)].$$

Именно такой способ округления интервалов делает возможной реализацию интервальных алгоритмов с гарантированным включением искомых значений в полученные интервалы. В дальнейшем (Глава 2) в контексте твинных арифметик нам потребуются другие способы округления интервалов, например, сужающие округляемый интервал.

Заметим, что если у нас есть функция, округляющая вещественное число вниз, то функцию, округляющую вверх, можно получить по формуле

$$\mathrm{fl}_\uparrow(x) = -(\mathrm{fl}_\downarrow(-x)).$$

Двуместные интервальные операции, например $\circ \in \{+, -, \cdot, /\}$, над интервалами A и B в машинной арифметике выполняются в соответствии с определением

$$C = \mathrm{fl}_\updownarrow(A \circ B), \tag{1.12}$$

одноместные операции $\diamond$ выполняются в соответствии с определением

$$C = \mathrm{fl}_\updownarrow(\diamond B), \tag{1.13}$$

Если машинные интервальные операции заданы в соответствии с формулами (1.12), (1.13), то сохраняется свойство интервальной арифметики, называемое монотонностью по включению. Пусть

$I_1, I_2, J_1, J_2 \in I(R_M)$, $I_1 \subseteq J_1$, $I_2 \subseteq J_2$, $\diamond$ – одноместная, а $\circ$ – двуместная интервальные операции. Если $I_3 = \mathrm{fl}_{\updownarrow}(\diamond I_1)$, $J_3 = \mathrm{fl}_{\updownarrow}(\diamond J_1)$, $I_4 = \mathrm{fl}_{\updownarrow}(I_1 \circ I_2)$ $J_4 = \mathrm{fl}_{\updownarrow}(J_1 \circ J_2)$, то $I_3 \subseteq J_3$ и $I_4 \subseteq J_4$.

Верно также следующее утверждение, из которого следует вывод о корректности выполнения предлагаемых округлений по ходу выполнения интервальных алгоритмов.

Пусть $\diamond$ – одноместная, а $\circ$ – двуместная интервальные операции, $A, B \in I(R_M)$, $a, b \in R_M$, $a \in A$, $b \in B$. Тогда

$$\diamond A \subseteq \mathrm{fl}_{\updownarrow}(\diamond A) \in I(R_M);$$

$$\diamond a \in \mathrm{fl}_{\updownarrow}(\diamond A) \in I(R_M);$$

$$A \circ B \subseteq \mathrm{fl}_{\updownarrow}(A \circ B) \in I(R_M);$$

$$a \circ b \in \mathrm{fl}_{\updownarrow}(A \circ B) \in I(R_M).$$

Во всех проведенных рассуждениях мы молчаливо полагали, что не происходит выхода за границы интервала $[r_{\min}, r_{\max}]$.

Интервальное оценивание функции посредством вычисления ее естественного интервального расширения, если оно выполняется с применением округлений, отвечающих свойствам (1.12), (1.13), дает корректную оценку, хотя и более широкую, чем если бы операции выполнялись с абсолютной точностью. Нетрудно видеть, что чем более точно представляются числа в компьютере, тем меньше разрастаются интервалы в ходе оценки функции и тем точнее оценка.

Зависимость точности получаемых результатов от точности представления чисел в компьютере может быть сформулирована в виде Теоремы 1.7 [1]. Эта теорема характеризует точность выполнения вещественного алгоритма средствами интервальной арифметики на цифровой вычислительной машине. Исходные данные алгоритма при переводе из вещественной в интервальную форму представляются интервалами шириной в единицу младшего представимого разряда. На выходе алгоритма выходные значения представлены интервалами, гарантированно содержащими искомые значения. Абсолютной погрешностью $\Delta(X)$ назовем ширину полученного интервала, включающего искомое значение,

$$\Delta(X) = |X|.$$

Относительной погрешностью $\rho(X)$ назовем абсолютною погрешность, отнесенную к расстоянию интервала от точки 0. Будем считать, что $0 \notin X$.

$$\rho(X) = \frac{\Delta(X)}{\min\{|x| \mid x \in X\}}.$$

Теорема 1.7. Допустим, что машинная интервальная арифметика соответствует свойствам (1.12) и (1.13). Пусть, кроме того, имеется заданный в поле вещественных чисел алгоритм, который выполняется в машинной интервальной арифметике с мантиссой длины t_1. Если затем этот алгоритм выполнить в арифметике с мантиссой длины t_2, где $t_2 = t_1 + l$, $l > 0$, то абсолютная и относительная погрешности уменьшатся в b^l раз. Под алгоритмом здесь понимается однозначно определенная последовательность арифметических операций вместе с конкретными входными данными.

Теоретические основы машинной арифметики нашли свое воплощение в многочисленных программных системах, а также в разработках аппаратных средств.

Если говорить об аппаратных средствах, то в первую очередь необходимо отметить стандарт IEEE 754 [77]. Этот американский стандарт на арифметические процессоры предусматривает четыре вида направленных округлений, которые должны выполняться процессором, а именно: округление к ближайшему, округление к нулю, округление вверх и округление вниз. Ясно, что реализация машинной интервальной арифметики на таком процессоре намного более эффективна, чем на процессоре, не имеющем указанных средств. В последнем случае округления приходится реализовывать программным путем. Заметим, что большинство современных универсальных микропроцессоров и сопроцессоров для работы с числами с плавающей точкой (например, процессоры фирм Intel и Motorola) следуют указанному стандарту. Так что можно сказать, основные сложности в эффективной реализации машинной интервальной арифметики лежат не в отсутствии адекватных аппаратных средств, а в правильном программном их окружении. Вопросы реализации интервальных вычислений на базе стандарта IEEE 754 рассмотрены в работе [80].

К реализации интервальной арифметики имеют непосредственное отношение вопросы организации спецпроцессоров с регулируемой точностью представления чисел [53], [54], [109] и процессоров с поддержкой вычисления точного скалярного произведения [103], [92], [91]. Хорошие обзоры аппаратных средств для интервальных вычислений можно найти в [149], [150], [151], [152].

Программные средства, дающие возможность оперировать с интервальными типами данных, можно разделить на несколько направлений. К первому направлению относятся программные пакеты, позволяющие оперировать числами переменной точности представления [160], [164]. Такие пакеты, разумеется, применяются не только в рамках интервального подхода, но и для нужд интервальных вычислений они также часто используются.

Вторым направлением развития программных средств являются библиотеки для интервальных вычислений. Например, одна из наиболее известных библиотек INTLIB [85], [86] дает пользователю возможность работать с интервалами, выполнять с ними арифметические операции, вычислять элементарные функции, осуществлять ввод/вывод интервальных значений. На базе библиотеки написан специализированный пакет INTBIS [84]. Отметим также библиотеку PROFIL [75], [93].

Практически наиболее важным программным средством для интервальных вычислений являются специализированные языки программирования. Подробно на одном из них мы остановимся в Главе 5. Реализация нескольких таких языков была проведена в восьмидесятых годах. Эти языки, как правило, построены на базе широко известных универсальных языков программирования путем добавления новых типов данных и операций над этими типами данных. Отметим следующие языки: ACRITH-SC [47], PASCAL-SC [163], FORTRAN-SC [48], Modula-SC [58].

Для перечисленных языков характерно наличие средств, предназначенных для организации вычислений с гарантированными границами погрешности: выполнение арифметических операций с гарантированной точностью, направленные округления, вычисление скалярного произведения с максимальной возможной точностью, вычисление элементарных функций с гарантированной точностью.

Языки для гарантированных вычислений следующего поколения помимо перечисленных свойств также дают возможность работать с числами с переменной точностью представления, а также с векторами и матрицами, составленными из таких чисел. Встроены также высокоуровневые средства решения линейных и нелинейных систем уравнений, нахождения нулей полиномов и многих других задач, являющихся типовыми для вычислительной математики. К таким языкам отнесем PASCAL-XSC [10], [69], C-XSC [70].

Нельзя не упомянуть тот факт, что интервальная арифметика часто встраивается в системы компьютерной алгебры. У систем компьютерной алгебры и интервальных вычислений существует одна общая черта. И те и другие обеспечивают надежные, верифицированные вычисления, только совершенно разными способами. Системы компьютерной алгебры производят символьные преобразования объектов, которые по своей сути являются абсолютно корректными. В системах интервальных вычислений производится тщательный автоматический контроль вычислительных погрешностей. Оказывается, существует обширное поле совместного использования систем компьютерной алгебры и интервальных вычислений. Так, например, предварительное символьное преобразование выражений перед их интервальным вычислением может существенно сузить получаемые интервалы. В свою очередь, применение интервальных методов оценки позволяет, сделав предварительные выводы об области значений данного символьного выражения, упростить алгебраические преобразования [18]. Из систем компьютерной алгебры со встроенной интервальной арифметикой отметим систему Mathematica [90].

1.6.4 Апостериорный интервальный анализ

В настоящем параграфе мы рассмотрим алгоритм, автором которого является Ю. В. Матиясевич. Алгоритм служит для оценки множества значений рациональной функции и носит название апостериорного интервального анализа [14]. Алгоритм стоит несколько особняком от других рассмотренных в настоящей работе алгоритмов, предназначенных для этой же цели, но без его рассмотрения наше введение было бы неполным.

Пусть рассматриваемая функция от ν переменных $x_1, \ldots, x_\nu$ задана в виде программы

$$
\begin{aligned}
x_{\nu+1} &:= x_{\sigma_{\nu+1}} \circ_{\nu+1} x_{\tau_{\nu+1}}; \\
&\ldots \\
x_\mu &:= x_{\sigma_\mu} \circ_\mu x_{\tau_\mu};
\end{aligned}
\tag{1.14}
$$

Здесь $x_j \in R$, $\circ_i \in \{+, -, \cdot, /\}$, $\sigma_\lambda < \lambda$, $\tau_\lambda < \lambda$. Итогом работы алгоритма является число $y = x_\mu$. Функцию, описываемую алгоритмом, будем обозначать $Y(x_1, \ldots, x_\nu)$.

Далее будем полагать, что входные величины x_i, $i = 1, \ldots, \nu$ заданы с некоторыми погрешностями, не превосходящими $e_1, \ldots, e_\nu$. Нашей задачей является преобразование программы (1.14) в новую программу с 2ν входными переменными $x_1, e_1, \ldots, x_\nu, e_\nu$ и двумя выходными переменными y и e, которая обладает следующим свойством: для любых $\chi_1, \ldots, \chi_\nu$, таких, что

$$|\chi_i - x_i| \le e_i, \quad i = 1, \ldots, \nu$$

имеет место

$$|Y(\chi_1, \ldots, \chi_\nu) - y| \le e.$$

Одно из решений поставленной задачи дает стандартная интервальная арифметика, а точнее, вычисление естественного интервального расширения функции Y. Ниже будет описано другое, более точное решение.

Обозначим $y^{<i>}(\overline{x}) = \partial Y(\overline{x})/\partial x_i$. Здесь и далее до конца параграфа под $\overline{x}$ понимается $(x_1, \ldots, x_\nu)$ и под $\overline{e}$ понимается $(e_1, \ldots, e_\nu)$.

Обозначим также

$$\Delta(\overline{x}, \overline{e}) = \max_{|\chi_i - x_i| \le e_i} |Y(\chi_1, \ldots, \chi_\nu) - Y(x_1, \ldots, x_\nu)|.$$

Нетрудно видеть, что (для малых e_i)

$$\Delta(\overline{x}, \overline{e}) \approx \sum_{i=1}^{\nu} |y^{<i>}| \cdot e_i. \qquad (1.15)$$

Будем называть точку $\overline{x}$ неособой, если для нее $Y(\overline{x})$ определено и $y^{<i>}(\overline{x}) \neq 0$ при $i = 1, \ldots, \nu$. Будем называть решение нашей задачи (т. е. метода вычисления e и y) асимптотически оптимальным, если следующее условие выполнено для любой неособой точки

$$e/\Delta(\overline{x}, \overline{e}) \xrightarrow{\overline{e} \to \overline{0}} 1.$$

Оценка, построенная на основе вычисления естественного интервального расширения, дает не асимптотически оптимальное решение, а решение, для которого выполняется более слабое условие

$$e/\Delta(\overline{x}, \overline{e}) = O(1).$$

Асимптотически оптимальное решение предложено Хансеном в [71]. В этой работе предложено для вычисления e воспользоваться формулой (1.15) и положить

$$e = \sum_{i=1}^{\nu} |y^{<i>}| \cdot e_i.$$

Полученная оценка является асимптотически оптимальной, но не гарантированной. Для полного решения задачи необходимо модифицировать способ получения оценки так, чтобы он давал гарантированный результат и, кроме того, научиться вычислять частные производные $y^{<i>}$. Такое решение предложено в работе [71]. Определяется специальная структура данных, включающая кроме величин x_j также величины $\partial x_j / \partial x_i$ и особую величину, оценивающую погрешности порядка выше первого. Алгоритм, заданный программой (1.14), построчно преобразовывается таким образом, что оперирует не с переменными x_j, а с переменными, содержащимися в новой структуре данных.

Можно показать, что алгоритм, дающий гарантированную оценку погрешности, по трудоемкости отличается от первоначального алгоритма в $O(\nu)$ раз. Изложенный ниже алгоритм апостериорного интервального анализа отличается от исходного алгоритма по трудоемкости не более, чем в константу раз.

Суть метода апостериорного интервального анализа состоит в использовании для вычисления величины e частных производных, как и в методе Хансена. Однако частные производные вычисляются несколько иным способом, идея которого в первоначальном виде была предложена в [45].

Пусть входные величины алгоритма (1.14) $x_1, \ldots, x_\nu$ являются функциями независимой переменной t, причем $x_i(0) = x_{i,0}$. Очевидно,

$$y'(0) = \sum_{i=1}^{\nu} y^{<i>} x_i'(0). \tag{1.16}$$

Рассмотрим далее задачу: для данного λ ($\nu \leq \lambda \leq \mu$) найти $z_1, \ldots, z_\lambda$ такие, что

$$y'(0) = \sum_{i=1}^{\lambda} z_i x_i'(0). \tag{1.17}$$

Это равенство должно выполняться при произвольных функциях $x_1(t), \ldots, x_\nu(t)$, лишь бы $x_1(0) = x_{1,0}, \ldots, x_\nu(0) = x_{\nu,0}$. Одно очевидное решение прямо вытекает из (1.16). Оно равно

$$z_i = \begin{cases} y^{<i>}, & \text{если } i = 1, \ldots, \nu, \\ 0, & \text{если } i = \nu + 1, \ldots, \lambda. \end{cases}$$

При $\lambda > \nu$ решение не единственно, например, при $\lambda = \mu$ существует еще одно решение:

$$\begin{aligned} z_1 &= 0; \\ &\ldots \\ z_{\mu-1} &= 0; \end{aligned}$$

$$z_\mu \;=\; 1.$$

Добавим программу

$$\begin{aligned} z_1 &:= 0; \\ &\dots \\ z_{\mu-1} &:= 0; \\ z_\mu &:= 1; \end{aligned}$$

в конец программы (1.14). Объединенную программу обозначим P_μ. Построим последовательность программ $P_{\mu-1}, \dots P_\nu$. Каждая программа будет давать решение (1.17).

Программа P_{i-1} получается из программы P_i добавлением двух строк.

Если $\circ_i = \pm$, то добавятся строки

$$\begin{aligned} z_{\sigma_i} &:= z_{\sigma_i} + z_i; \\ z_{\tau_i} &:= z_{\tau_i} \circ_i z_i; \end{aligned}$$

если $\circ_i = \cdot$, то добавятся строки

$$\begin{aligned} z_{\sigma_i} &:= z_{\sigma_i} + z_i x_{\tau_i}; \\ z_{\tau_i} &:= z_{\tau_i} + z_i x_{\sigma_i}; \end{aligned}$$

если $\circ_i = /$, то добавятся строки

$$\begin{aligned} z_{\sigma_i} &:= z_{\sigma_i} + z_i / x_{\tau_i}; \\ z_{\tau_i} &:= z_{\tau_i} - z_i x_i / x_{\tau_i}; \end{aligned}$$

Программа P_ν дает решение для

$$y'(0) = \sum_{i=1}^{\nu} z_i x_i'(0). \tag{1.18}$$

Последнее равенство однозначно определяет набор чисел $z_1, \ldots, z_\nu$. Действительно, это равенство должно выполняться для любого набора функций $x_1(t), \ldots, x_\nu(t)$ (при фиксированных $x_1(0), \ldots, x_\nu(0)$), в частности, и для любого набора

$$x_i(t) = \begin{cases} x_i(0) + t, & \text{если } i = \tau, \\ x_i(0), & \text{если } i \neq \tau; \end{cases}$$

при $1 \leq \tau \leq \nu$.

Из (1.16) и (1.18) следует

$$z_\tau = \sum_{i=1}^{\nu} z_i x_i'(0) = y'(0) = \sum_{i=1}^{\nu} y^{<i>} x_i'(0) = y^{<\tau>}.$$

Из-за единственности решения (1.18) программа P_ν вычисляет частные производные $y^{<1>}, \ldots, y^{<\nu>}$. Программа P_ν содержит не более 6ν арифметических операций.

К сожалению, полученные числа $y^{<1>}, \ldots, y^{<\nu>}$ после подстановки в равенство

$$Y(\overline{\chi}) - Y(\overline{x}) \approx \sum_{i=1}^{\nu} y^{<i>}(\chi_i - x_i)$$

не дают гарантированного решения, так как само это равенство является приближенным.

Можно написать точное равенство

$$Y(\overline{\chi}) - Y(\overline{x}) = \sum_{i=1}^{\nu} y^{<i>}(\overline{\eta})(\chi_i - x_i), \tag{1.19}$$

где $\eta_1, \ldots, \eta_\nu$ – некоторые числа, для которых выполняются неравенства $|\eta_i - x_i| \leq e_i$, $i = 1, \ldots, \nu$. Числа $y^{<i>}(\overline{\eta})$ мы можем вычислить, используя уже полученный алгоритм для вычисления z_i, где в качестве входных данных использованы интервалы с центром в x_i и радиусом e_i. Из-за множителей $\chi_i - x_i$ в формуле (1.19) достаточно иметь довольно грубую оценку для $z_1, \ldots, z_\nu$. Если значения $z_1, \ldots, z_\nu$ вычислены в виде интервалов с центрами в точках $z_{1c}, \ldots, z_{\nu c}$ и с радиусами $z_{1r}, \ldots, z_{\nu r}$, то окончательной уточненной оценкой для величины y будет являться интервал с центром в точке x_μ и с радиусом

$$\sum_{i=1}^{\nu}(|z_{ic}| + z_{ir})e_i.$$

Вычисление согласно алгоритму апостериорного интервального анализа состоит из двух стадий. Первая стадия – вычисления в соответствии с классической интервальной арифметикой. Все промежуточные значения сохраняются в памяти. Вторая стадия – интервальное вычисление z_i, которые обеспечивают высокоточное вычисление погрешности. В работе [14] показано, что полученный алгоритм обладает свойством асимптотической оптимальности.

Выше мы рассматривали апостериорно-интервальный алгоритм для программ простейшей структуры, а именно программ без ветвлений, без циклов, без машинных округлений и т. д. В работе [14] показано, что метод может быть обобщен на программы существенно более общего вида.

1.7 Некоторые алгебраические вопросы

Настоящий параграф посвящен рассмотрению некоторых алгебраических свойств множества обычных интервалов и его обобщений. При этом нас будут в основном интересовать свойства множества стандартных интервалов $I(R)$ и множества интервалов Каухера $I^*(R)$. Факты, приведенные в параграфе, будут использованы позднее для установления алгебраических свойств новых интервальных пространств и пространств твинов. Нами будут также рассмотрены некоторые алгебраические свойства интервальных округлений.

Хорошо известен тот факт (см. [9]), что аддитивная система $< I(R), + >$ образует коммутативную полугруппу, для которой справедлив закон сокращения:

$$\forall A, B, C \in I(R) \quad A + B = A + C \;\Rightarrow\; B = C.$$

Мультипликативная система $< I(R)\backslash\{0\}, \cdot >$ также является коммутативной полугруппой. Если обозначить $J = \{Y \in I(R) \mid 0 \notin Y\}$, то система $< J, \cdot >$ является коммутативной полугруппой, для которой выполняется закон сокращения:

$$\forall A, B, C \in J \quad A \cdot B = A \cdot C \;\Rightarrow\; B = C.$$

Множество $I(R)$, к сожалению, не является группой относительно операции $+$. Действительно, большая часть интервалов, а именно все, ширина которых отлична от нуля, не имеют обратных элементов относительно сложения.

Отметим также, что множество $I(R)$ не является решеткой по отношению к порядку $\subseteq$, т. к. не любое двухэлементное подмножество $I(R)$ имеет точную нижнюю грань (см. [3], [154], [157]). Действительно, для двух непересекающихся интервалов единственным претендентом на роль точной нижней грани, то есть элемента, содержащегося в обоих интервалах, является пустое множество, но оно по определению не является интервалом.

Арифметика Каухера, описанная в разделе 1.6.2.1, во многом обязана своим появлением именно желанию иметь дело с более удобной с алгебраической точки зрения структурой. Описание алгебраических свойств арифметики Каухера можно найти в работах [62], [63], [113], [114], [154], [157].

Если для $A, B \in I^*(R)$ определить $A \subseteq B$ как

$$A \subseteq B \quad \stackrel{\text{def}}{\Longleftrightarrow} \quad A^- \geq B^- \;\&\; A^+ \leq B^+,$$

то нетрудно показать, что множество интервалов Каухера с указанным отношением порядка является условно полной решеткой [3]. Легко проверить, что

$$\sup_{\subseteq}\{A_i \mid i \in \Omega\} = [\inf_{\leq}\{A_i^- \mid i \in \Omega\}, \sup_{\leq}\{A_i^+ \mid i \in \Omega\}] \in I^*(R);$$

$$\inf_{\subseteq}\{A_i \mid i \in \Omega\} = [\sup_{\leq}\{A_i^- \mid i \in \Omega\}, \inf_{\leq}\{A_i^+ \mid i \in \Omega\}] \in I^*(R).$$

Множество интервалов Каухера образует коммутативную группу по операции $+$, что легко может быть установлено непосредственно. Обратным элементом к элементу $A = [a, b] \in I^*(R)$ является элемент $\mathrm{opp}(A) = [-a, -b]$. В арифметике Каухера каждое уравнение вида

$$A + X = B, \quad A, B \in I^*(R)$$

имеет единственное решение. Легко видеть, что коммутативная группа $< I^*(R), + >$ изоморфна стандартной аддитивной группе R^2.

В работах [113], [114] предложен интересный подход, заключающийся в аксиоматическом определении коммутативной полугруппы $< I(R), + >$ с последующим ее вложением в абелеву группу $\mathcal{G}$ в соответствии с канонической схемой, применимой для любой коммутативной полугруппы, удовлетворяющей закону сокращения [12]. $\mathcal{G} = (I(R))^2/E$. Элементами группы $\mathcal{G}$ являются пары (A, B), $A, B \in I(R)$. Множество пар факторизуется по отношению эквивалентности

$$E : (A, B) \sim (C, D) \iff A + D = B + C.$$

Операция сложения задается покомпонентно:

$$(A, B) + (C, D) = (A + C, B + D).$$

Далее устанавливается изоморфизм группы $\mathcal{G}$ и группы $< I^*(R), + >$, что позволяет переносить некоторые результаты, известные для $\mathcal{G}$, на $I^*(R)$ и проектировать их на исходную полугруппу интервалов $I(R)$.

Если говорить о прикладных аспектах интервальных вычислений, то представляет интерес вопрос интервальных округлений. К интервальным округлениям, помимо утилитарного подхода, описанного в разделе 1.6.3, возможен строгий аксиоматический подход [81], [82].

Отображение $\phi: I(R) \to I(R)$ назовем интервальным округлением, если оно удовлетворяет следующим аксиомам:

A1. $\forall I \in I(R) \quad I \subseteq \phi(I)$,

A2. $\forall I, J \in I(R) \quad I \subseteq J \Rightarrow \phi(I) \subseteq \phi(J)$,

A3. $\phi^2 = \phi$.

Под приведенное определение подпадают не только отображения, интуитивно являющиеся округлениями, но и другие, достаточно экзотические примеры. Для того, чтобы иметь дело лишь с содержательными примерами, будем рассматривать только те округления, которые коммутируют между собой. Можно показать, что произведение двух интервальных округлений $\phi\psi$ является интервальным округлением тогда и только тогда, когда ϕ и ψ коммутируют между собой.

Обозначим через $\phi^{(k,l)}([a,b])$ интервальное округление, получаемое округлением левой границы интервала влево до k-го разряда и правой границы интервала вправо до l-го разряда, $k, l \in Z$:

$$\phi^{(k,l)}([a,b]) = ([a_k^{(-)}, b_l^{(+)}]).$$

Такое округление назовем (k,l)-округлением. Основание системы счисления считаем фиксированным. Обозначим множество всех (k,l)-округлений через $\Phi^{(Z,Z)}$. Через Φ обозначим множество всех коммутирующих друг с другом округлений, содержащее $\Phi^{(Z,Z)}$. Множество Φ является коммутативной полугруппой идемпотентов, на которой может быть введено отношение порядка

$$\phi_1 \leq \phi_2, \quad \text{если } \phi_1\phi_2 = \phi_1.$$

Полугруппа Φ с введенным порядком является решеткой.

Алгебраическая теория округлений позволяет рассматривать округления как таковые, без привязки к конкретной машине и даже к конкретной системе счисления.

1.8 Некоторые вопросы сложности интервальных вычислений

Выше (раздел 1.3) было дано определение объединенного интервального расширения. В настоящем разделе мы коснемся вопросов сложности вычисления объединенного интервального расширения, а также сложности задач оценки такого расширения.

Пусть имеется непрерывная вещественная функция $f(x_1, \ldots, x_n)$, $x_i \in R$, и нам необходимо вычислить $\overline{F}(X_1, \ldots, X_n)$, $X_i \in I(R)$. Ниже мы ограничимся только рациональными функциями f с рациональными коэффициентами (т. е. такими функциями, запись которых построена из переменных $x_1, \ldots, x_n$, рациональных констант и знаков четырех арифметических операций $+, -, \cdot, /$). В качестве границ интервалов X_i нами также будут рассматриваться рациональные числа.

Необходимо формализовать понятие сложности вычислений. Какой алгоритм считать сложным, какой простым? Как оценивать сложность конкретного алгоритма? Этот вопрос хорошо изучен в теоретической информатике. Существует несколько разных систем определений. Мы будем считать алгоритм U алгоритмом полиномиальной сложности (выполнимым) тогда и только тогда, когда существует полином P, для которого выполняется $\tau_U(x) \le P(|x|)$, где x – входные данные алгоритма U, $|x|$ – длина входных данных в какой-нибудь кодировке, $\tau_U(x)$ – время выполнения алгоритма, выраженное, например, в количестве выполненных арифметических операций.

Для многих математических задач до сих пор неясно, могут ли они быть решены за полиномиальное время. Существует класс задач, для которых известно, что если одна из задач этого класса разрешима в полиномиальное время, то и остальные задачи класса также могут быть разрешены в полиномиальное время. Такие задачи называются NP-трудными. Большинство исследователей убеждены, что ни для одной из задач класса не существует полиномиального разрешающего алгоритма. Хороший обзор основ теории сложности вычислений можно найти в [66]. Сложности интервальных вычислений посвящена монография [101].

Еще в начале 80-х годов А. Гаганов доказал утверждения, которые дают ответ на вопрос о сложности вычисления множества значений функции [5]. Мы их переформулируем в виде двух следующих теорем.

Теорема 1.8. Пусть $f(x_1, \ldots, x_n)$ – полином многих переменных с рациональными коэффициентами, а $X_1, \ldots, X_n$ – интервалы с рациональными границами. Задача точного вычисления множества значений функции $f(x_1, \ldots, x_n)$ при известных областях изменения ее переменных $X_1, \ldots, X_n$ в общем случае является NP-трудной.

Теорема 1.9. Для любого $\varepsilon > 0$ следующая задача в общем случае является NP-трудной. Для любого полинома многих переменных с рациональными коэффициентами $f(x_1, \ldots, x_n)$ и набора интервалов с рациональными границами $X_1, \ldots, X_n$ найти такой интервал $Y \in IR$, $Y \supseteq \overline{F}(X_1, \ldots, X_n)$, что $|Y| - |\overline{F}(X_1, \ldots, X_n)| \leq \varepsilon$.

В дальнейшем было получено много результатов, устанавливающих NP-трудность некоторых конкретных задач интервальных вычислений. Для ряда задач также была установлена их выполнимость. Задачи получены путем ослабления части условий Теорем 1.8 и 1.9. Эти работы в основном связаны с именами трех авторов – А. Лакеева, В. Крейновича, И. Рона. Хороший краткий обзор этих результатов содержится в [100]. Полный обзор можно найти в [101]. Приведем некоторые, наиболее интересные с нашей точки зрения факты, большая часть которых нам понадобится в дальнейшем.

Теорема 1.10. [101] Для любого $\delta > 0$ задача точного вычисления множества значений полиномиальной функции $f(x_1, \ldots, x_n)$ при $x_i \in X_i = [x_i^-, x_i^+]$ при $|x_i^+ - x_i^-| \leq \delta$ является NP-трудной.

Утверждение этой теоремы интерпретируется следующим образом: даже если ограничиться рассмотрением сколь угодно узких интервалов, вычисление множества значений полиномиальной функции при условии, что значения переменных принадлежат этим узким интервалам, является NP-трудной задачей.

Другая возможность упростить задачу – ограничить количество переменных полинома. На этом пути получен следующий результат [100], [101].

Теорема 1.11. Для каждого n существует алгоритм с полиномиальным временем выполнения, вычисляющий множество значений любой функции $f(x_1, \ldots, x_n)$, являющейся полиномом от n переменных, при условии, что $x_i \in X_i$.

К сожалению, полиномиальный алгоритм, существование которого устанавливается Теоремой 1.11, является примером выполнимого алгоритма, время выполнения которого (в общем случае) измеряется миллионами лет на самом быстром современном компьютере даже для полиномов от четырех переменных [73].

А что произойдет, если ограничить степень рассматриваемых полиномов? Для полиномов первой степени, т. е. линейных функций, задача тривиальна. Действительно, в этом случае

$$f(x_1, \dots, x_n) = a_0 + \sum_{i=1}^{n} a_i x_i,$$

следовательно

$$\overline{F}(X_1, \dots, X_n)$$
$$= [a_0 + \sum_{\substack{i=1 \\ a_i>0}}^{n} a_i X_i^- + \sum_{\substack{i=1 \\ a_i<0}}^{n} a_i X_i^+, \;\; a_0 + \sum_{\substack{i=1 \\ a_i<0}}^{n} a_i X_i^- + \sum_{\substack{i=1 \\ a_i>0}}^{n} a_i X_i^+].$$

Уже для полиномов второй степени задача становится NP-трудной [146].

Если перейти от полиномиальных функций, например, к дробно-линейным, то оказывается, что для них задача решается в полиномиальное время, причем соответствующий алгоритм может быть практически применен и применяется в приложениях [107].

Теорема 1.12. Существует полиномиальный алгоритм, вычисляющий интервал $\overline{F}(X_1, \dots, X_n)$ для любой дробно линейной функции

$$f(x_1, \dots, x_n) = \frac{a_0 + \sum a_i x_i}{b_0 + \sum b_i x_i}.$$

Последняя задача естественным образом приводит к рассмотрению вопроса о сложности решения интервальных линейных систем. Задача вычисления точных границ такого решения в общем случае является NP-трудной [98]. Различные нетривиальные упрощения задачи также не имеют полиномиальных разрешающих алгоритмов [99], [147], [148].

Итак, мы видим, что большинство рассмотренных задач является NP-трудными. Это значит, что таковыми они являются в общем случае, однако для части входных данных они могут решаться в полиномиальное время. С одной стороны, такая ситуация стимулирует разработку разнообразных эвристик, т. е. дополнительных соображений, помогающих в возможных случаях решить проблему. С другой стороны, интересно выяснить соотношение числа случаев, когда задача решается в полиномиальное время, к числу случаев, когда она NP-трудна.

С этой целью в [100] предложен любопытный подход, реализующий хорошо известную идею оценки соотношения мер двух множеств. В указанной работе эта идея переносится с множеств вещественных чисел с ненулевой мерой Лебега на счетное множество рациональных чисел. Вводится аналог понятия "почти всегда" ("почти везде"), которое используется в математике для характеристики свойства, выполняющегося для всех точек множества ненулевой меры за исключением подмножества меры 0.

Пусть $\varepsilon > 0$ – вещественное число, $D \subseteq R^n$ – компактная область с ненулевой мерой Лебега $\mu(D) > 0$, а $P(x)$ – некоторая бинарная функция (свойство), верное для некоторых точек $x \in D$ и неверное для остальных. Будем говорить, что P верно для (D, ε)-почти всех точек $x \in D$, если $\mu(\{x \in D \mid \neg P(x)\}) \leq \varepsilon\mu(D)$.

Пусть $\eta > 0$ – вещественное число. Будем говорить, что интервалы $[X^-, X^+]$ и $[Y^-, Y^+]$ η-близки, если $|X^- - Y^-| \leq \eta$ и $|X^+ - Y^+| \leq \eta$.

Пусть U – алгоритм, решающий задачу оценки множества значений функции, $\eta > 0$, $f(x_1, \ldots, x_n)$ – рациональная функция с рациональными коэффициентами, $X_1, \ldots, X_n$ – набор интервалов. Будем говорить, что U η-точен на f и $X_1, \ldots, X_n$, если для любого набора интервалов $Y_1, \ldots, Y_n$ с рациональными границами, каждый из которых η-близок к соответствующему интервалу X_i, алгоритм U возвращает границы точного множества значений $\overline{F}(Y_1, \ldots, Y_n)$.

Будем говорить, что алгоритм U *почти всегда* точен для узких входных интервалов, если для каждой компактной области D, $\mu(D) > 0$, для каждой рациональной функции с рациональными коэффициентами f, которая конечна на открытом множестве $N \supset D$, и для любого $\varepsilon > 0$ существуют $\delta > 0$ и $\eta > 0$ такие, что для (D, ε)-почти всех $x = (x_1, \ldots, x_n)$ выполняется следующее условие: если все n входных интервалов $X_i = [x_i - \Delta_i, x_i + \Delta_i]$ имеют ширину $2\Delta_i$, $\Delta_i < \delta$, то алгоритм U является η-точным на f и X_i.

Теорема 1.13. [101], [105] Существует алгоритм U с полиномиальным временем выполнения, который для рациональной функции f с рациональными коэффициентами и интервалов $X_1, \ldots, X_n$ возвращает оценку для $\overline{F}(X_1, \ldots, X_n)$, которая почти всегда точна для узких входных интервалов.

Приведенная теорема показывает, что для узких входных интервалов задача вычисления множества значений рациональной функции почти всегда выполнима.

В первой главе мы рассмотрели существующую теорию интервальных арифметик и ее применение, главным образом к задаче внешней оценки множества значений функции. Показано, что на основе стандартной интервальной арифметики был создан ряд обобщений, преодолевающих те или иные недостатки стандартной интервальной арифметики. Были описаны принципы машинной реализации интервальных вычислений.

Вместе с тем было показано, что весь механизм интервальных оценок предназначен исключительно для локализации неизвестных действительных значений и для внешней оценки неизвестных интервалов. В существующем виде интервальные вычисления не предназначены для получения других типов оценок, например для внутренних и двусторонних оценок неизвестных интервалов. В связи с этим актуальной является задача разработки нового, более общего механизма, который позволит качественно по-иному локализовывать неизвестные интервальные величины. Решению этой задачи будут посвящены последующие главы диссертационной работы. Поставленная задача предполагает не просто построение теории, но и нахождение необходимых алгоритмов, оценок эффективности, принципов машинной реализации. Только рассмотрение всех этих вопросов в совокупности с применением теории к решению конкретных задач позволит сделать вывод о решении поставленной проблемы.

Глава 2. Обобщенные интервальные арифметики и арифметики твинов

2.1 Интервальные арифметики и учет монотонности

В настоящем разделе мы рассмотрим некоторые аспекты учета монотонности функций при оценивании множеств их значений. Такой учет необходим для повышения точности оценки. Постановки задачи мы коснулись в разделе 1.4, а известные результаты, полученные другими исследователями, были изложены в разделе 1.6.2.

Алгоритмы оценки множеств значений функции, основанные на Теоремах 1.5 и 1.6 являются весьма эффективными. Однако их практическое применение затрудняется необходимостью отдельно проверять функции на монотонность, что зачастую является трудной задачей, не решаемой в рамках интервального анализа. Например, оценивая множество значений функции $f = p + q$ при известных оценках p и q мы должны быть убеждены в монотонности f, если, конечно, хотим воспользоваться указанными теоремами.

В работах [121], [122], [123] автором диссертации предлагается способ получения тех же результатов, что и в Теореме 1.6, но при ослабленном условии монотонности. Кроме того, результат получен не только для арифметических операций, но и для более общего класса функций. Функции, принадлежащие этому классу, могут рассматриваться как базовые при построении удобных аналитических представлений оцениваемых функций. Ниже будет рассмотрен также способ графического представления функций, в ряде случаев помогающий проиллюстрировать конкретные рассматриваемые примеры.

Итак, пусть функция $f = g(p(x), q(x))$ задана для всех $x \in X$, $X \in I(R)$, функции g, p и q непрерывны на рассматриваемых интервалах. Пусть также для функций p и q построены интервальные расширения $P(X)$ и $Q(X)$, дающие оценки

$$x \in X \Rightarrow p(x) \in P(X) \ \& \ q(x) \in Q(X).$$

Указанные оценки точны, т. е.

$$\forall y_1 \in P(X) \ \forall y_2 \in Q(X) \ \exists x_1 \in X \ \exists x_2 \in X : \ p(x_1) = y_1, q(x_2) = y_2.$$

Зафиксируем на координатной плоскости R^2 прямоугольник $P(X) \times Q(X)$. На этой же плоскости изобразим кривую, заданную параметрически, т.е. как множество точек с координатами $(p(x), q(x))$. На кривой стрелками отметим направление, соответствующее изменению x в сторону увеличения. Обозначив систему координат на плоскости (t_1, t_2), построим в R^3 график функции $g(t_1, t_2)$. На нашем чертеже мы

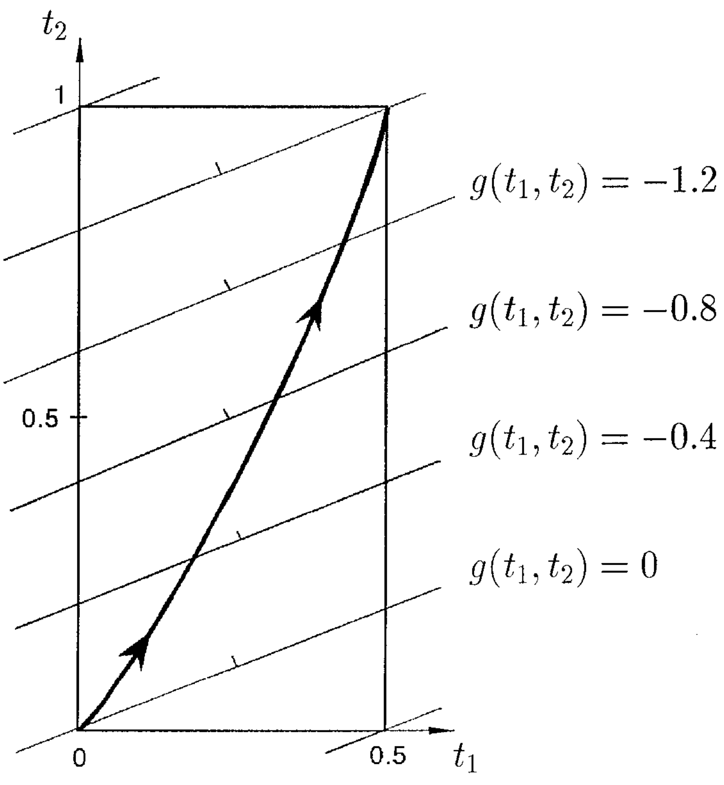

Рис. 2.1.

будем рисовать лишь описанную плоскость и проекции на нее линий одного уровня ($g(t_1, t_2) = \mathrm{const}$) графика функции g. На линиях одного уровня поставим штрихи, направленные в сторону убывания значений функции, как это принято, например, на топографических картах.

Пример 2.1. (Рис. 2.1.) Пусть $p(x) = x$; $q(x) = \mathrm{tg}\,(\pi x/2)$; $g(t_1, t_2) = t_1 - 2t_2$; $X = [0, 1/2]$.

Попробуем на основе этого примера продемонстрировать суть учета зависимости пременных. Если бы в функции $f(x) = g(p(x), q(x))$ два вхождения переменной были бы независимы, т. е. f была бы функцией двух переменных $f(y_1, y_2) = g(p(x_1), q(x_2))$, $y_1 = p(x_1)$, $y_2 = q(x_2)$, каждая из которых независимо пробегает все значения исходных интервалов $Y_1 = [0, 1/2]$ и $Y_2 = [0, 1]$, то множество возможных значений функции $f(y_1, y_2)$ в точности было бы равно множеству значений функции g при аргументах, располагающихся в прямоугольнике $[0, 1/2] \times [0, 1]$. Но, т. к. два аргумента функции g зависимы, не для всех точек (t_1, t_2) прямоугольника можно подобрать такое x, что $t_1 = p(x)$, $t_2 = q(x)$. На самом деле все точки прямоугольника, которые надо рассматривать в качестве аргументов функции g, есть в точности те точки, которые расположены на кривой $(p(x), q(x))$. Как нетрудно вычислить, при оценке множества значений функции f без учета зависимости переменных, мы получаем $\overline{F}(X) \subseteq [-2, 1/2]$. Если же принять во внимание указанную зависимость, то получаем: $\overline{F}(X) \subseteq [-3/2, 0]$.

Пример 2.2. (Рис. 2.2.) Теорему 1.6 можно посредством описанного выше способа построения диаграмм проиллюстрировать следующим образом. Пусть $g(t_1, t_2) = t_1 + t_2$. Возьмем три набора троек (p, q, X), которым соответствуют три изображенные кривые:

1. $X_1 = [0, 1]$, $p_1 = x^2$, $q_1 = 2\sqrt{x}$;

2. $X_2 = [0,1]$, $p_2 = x$, $q_2 = 2x$;

3. $X_3 = [1,2]$, $p_3 = \log_2 x$, $q_3 = 4 - 2x$.

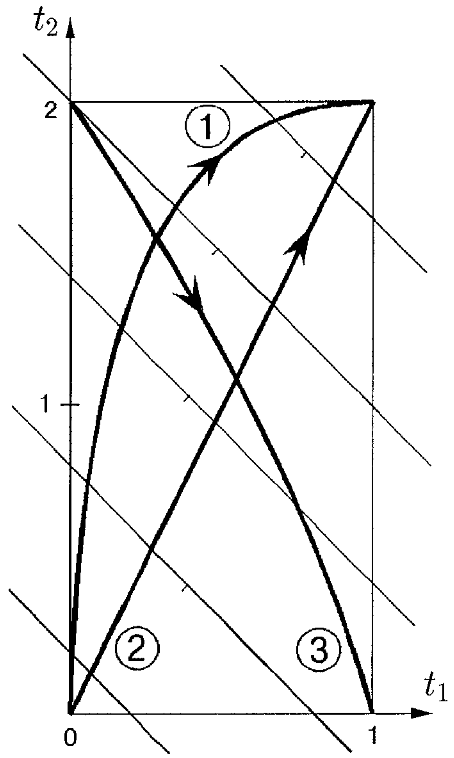

Рис. 2.2.

Во всех трех случаях функции p и q непрерывны и монотонны на соответствующих промежутках. Функция $f = p + q$ во всех случаях также монотонна и непрерывна. В первом и втором случае мы имеем совпадение характера монотонности функций p и q, а в третьем случае функция p возрастает, а функция q убывает.

В соответствии с утверждением Теоремы 1.6 множество значений каждой из функций $f_i = p_i + q_i$ на любом подынтервале $I \subseteq X$ можно вычислить по формулам:

$$\begin{aligned} \overline{F}_i(I) &= \overline{P}_i(I) + \overline{Q}_i(I), \quad i = 1, 2; \\ \overline{F}_3(I) &= \overline{P}_3(I) +^- \overline{Q}_3(I). \end{aligned}$$

Нетрудно видеть, что условия монотонности функций p, q и f являются достаточными, но не необходимыми. Их можно существенно ослабить. Приведем пример диаграммы, которая демонстрирует ситуацию, когда ни одна из указанных функций не является монотонной, однако оценки, проведенные с использованием Теоремы 1.6, остаются верными (рис. 2.3).

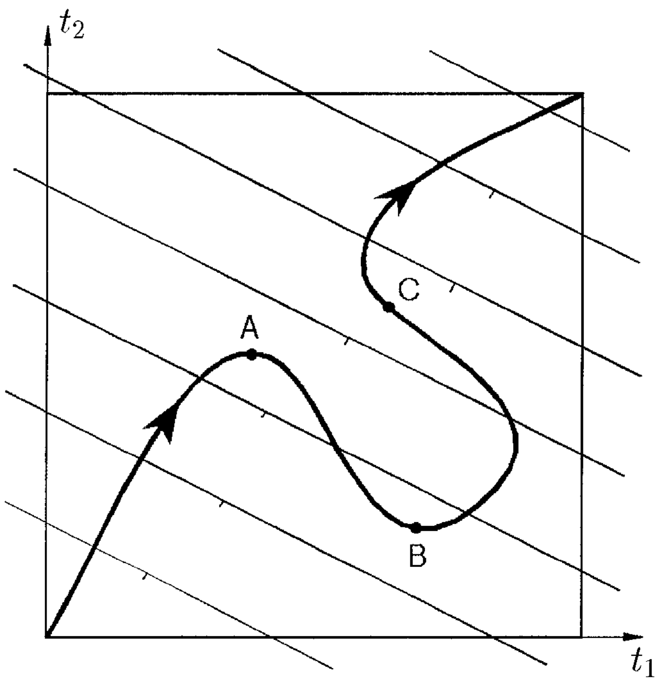

Рис. 2.3.

Действительно, пусть обозначенным на кривой точкам A, B, и C соответствуют значения аргумента x_A, x_B и x_C. Граничные точки интервала X обозначим x_1 и x_2 . Пусть имеем $x_1 < x_A < x_B < x_C < x_2$. Однако, $p(x_1) < p(x_2)$, а $p(x_B) > p(x_C)$ (т. е. p не монотонна), и $q(x_1) < q(x_2)$, а $q(x_A) > q(x_B)$ (т. е. q не монотонна). Кроме того, $f(x_A) > f(x_B)$. Все три функции монотонными не являются. Оказывается, условие монотонности можно заменить другим, более слабым условием, а именно следующим.

Определение 2.1. Будем говорить, что функция $f : R \to R$ удовлетворяет условию преобладания концевых точек на отрезке $I = [x_1, x_2]$ и записывать $f \in Z(I)$, если она непрерывна на I и либо $f(x_1) \leq f(x) \leq$

$f(x_2)$ для всех $x \in I$, либо $f(x_1) \geq f(x) \geq f(x_2)$ для всех $x \in I$. В первом случае будем писать $f \in Z_{\nearrow}(I)$, а во втором случае $f \in Z_{\searrow}(I)$.

Для функции $f(t_1, t_2)$ можно, вообще говоря, построить четыре интервальных расширения F_i, $i = 1, \ldots, 4$:

$$\begin{aligned} F_1([a,b],[c,d]) &= [f(a,c), f(b,d)]; \\ F_2([a,b],[c,d]) &= [f(b,c), f(a,d)]; \\ F_3([a,b],[c,d]) &= [f(a,d), f(b,c)]; \\ F_4([a,b],[c,d]) &= [f(b,d), f(a,c)]. \end{aligned}$$

Для конкретной функции f и для конкретных интервалов-аргументов лишь два из этих расширений, как правило, имеют смысл. Остальные два результатом имеют интервалы, у которых правая граница меньше левой. Докажем следующую теорему.

Теорема 2.1. Пусть задан интервал $X = [x^-, x^+] \in I(R)$ и на нем заданы две функции $p(x)$, $q(x)$. Пусть также для p и для q известны объединенные интервальные расширения $\overline{P}(X) = Y_1$ и $\overline{Q}(X) = Y_2$. Пусть также функция $f(x)$ задана как $f(x) = g(p(x), q(x))$, где g – некоторая функция двух переменных, непрерывная на $Y_1 \times Y_2$. Тогда для объединенного расширения $\overline{F}(X)$ имеем следующие выражения:

$$\begin{aligned} p \in Z_{\nearrow}(X) \,\&\, q \in Z_{\nearrow}(X) \,\&\, f \in Z_{\nearrow}(X) &\Rightarrow \overline{F}(X) = G_1(Y_1, Y_2); \\ p \in Z_{\nearrow}(X) \,\&\, q \in Z_{\nearrow}(X) \,\&\, f \in Z_{\searrow}(X) &\Rightarrow \overline{F}(X) = G_4(Y_1, Y_2); \\ p \in Z_{\nearrow}(X) \,\&\, q \in Z_{\searrow}(X) \,\&\, f \in Z_{\nearrow}(X) &\Rightarrow \overline{F}(X) = G_3(Y_1, Y_2); \\ p \in Z_{\nearrow}(X) \,\&\, q \in Z_{\searrow}(X) \,\&\, f \in Z_{\searrow}(X) &\Rightarrow \overline{F}(X) = G_2(Y_1, Y_2); \\ p \in Z_{\searrow}(X) \,\&\, q \in Z_{\nearrow}(X) \,\&\, f \in Z_{\nearrow}(X) &\Rightarrow \overline{F}(X) = G_2(Y_1, Y_2); \\ p \in Z_{\searrow}(X) \,\&\, q \in Z_{\nearrow}(X) \,\&\, f \in Z_{\searrow}(X) &\Rightarrow \overline{F}(X) = G_3(Y_1, Y_2); \\ p \in Z_{\searrow}(X) \,\&\, q \in Z_{\searrow}(X) \,\&\, f \in Z_{\nearrow}(X) &\Rightarrow \overline{F}(X) = G_4(Y_1, Y_2); \\ p \in Z_{\searrow}(X) \,\&\, q \in Z_{\searrow}(X) \,\&\, f \in Z_{\searrow}(X) &\Rightarrow \overline{F}(X) = G_1(Y_1, Y_2). \end{aligned}$$

Доказательство легко получается, путем рассмотрения значений функций в концевых точках интервала. Рассмотрим, например, первый случай, т. е. когда $p \in Z_{\nearrow}(X)$, $q \in Z_{\nearrow}(X)$, $f \in Z_{\nearrow}(X)$. В этом случае мы имеем:

$$\begin{aligned}
\overline{F}(X) &= [f(x^-), f(x^+)] \\
&= [g(p(x^-), q(x^-)), g(p(x^+), q(x^+))] \\
&= [g(Y_1^-, Y_2^-), g(Y_1^+, Y_2^+)] \\
&= G_1(Y_1, Y_2).
\end{aligned}$$

□

Диаграммы, иллюстрирующие все восемь перечисленных случаев, показаны на рис. 4, а-з.

Пример 2.3. Пусть

$$f(x) = \sqrt{\left|(x+6)^2 - \left(\frac{x^3}{3} - \frac{x}{3} + 2\right)^2\right|}, \quad X = [-2, 2].$$

В множество базовых функций включим следующие: $x + y$, $x - y$, $x \cdot y$, x^2, x^3, $\sqrt{|x^2 - y^2|}$. Обозначим $p(x) = x + 6$, $q(x) = \frac{x^3}{3} - \frac{x}{3} + 2$, $g(t_1, t_2) = \sqrt{|t_1^2 - t_2^2|}$. Используя стандартную интервальную арифметику, мы можем вычислить:

$$\begin{aligned}
\overline{P}(X) &= [-2, 2] + 6 = [4, 8]; \\
\overline{Q}(X) &\subseteq \frac{[-2, 2]^3}{3} - \frac{[-2, 2]}{3} + 2 = \left[-\frac{4}{3}, \frac{16}{3}\right].
\end{aligned}$$

Используя стандартную технику, можно получить оценку

$$\overline{F}(X) \subseteq \sqrt{\left|[4, 8]^2 - \left[-\tfrac{4}{3}, \tfrac{16}{3}\right]^2\right|} = [0, 8].$$

Полученная оценка является явно завышенной. Используя доказанную теорему, можно получить лучшую оценку. Действительно, функция q может быть представлена в виде $q(x) = h(r(x), s(x))$, где $r(x) = \frac{x^3}{3}$, $s(x) = -\frac{x}{3} + 2$ и $h(t_1, t_2) = t_1 + t_2$.

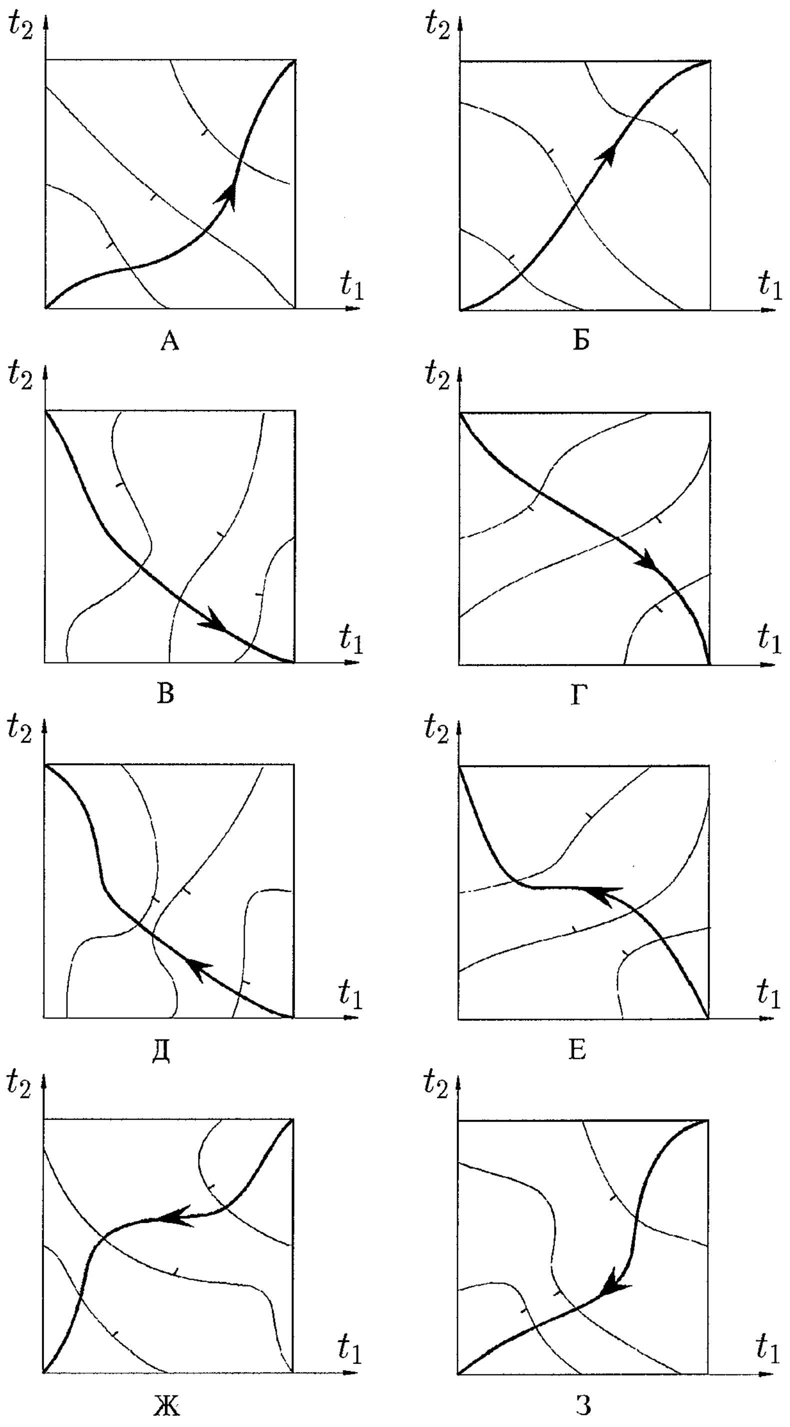

Рис. 2.4.

Легко видеть, что функция q удовлетворяет условиям теоремы, используя которую можно получить точное множество значений q:

$$\overline{Q}(X) = H_3\Big(\Big[-\frac{8}{3}, \frac{8}{3}\Big], \Big[\frac{4}{3}, \frac{8}{3}\Big]\Big) = \Big[h\Big(-\frac{8}{3}, \frac{8}{3}\Big), h\Big(\frac{8}{3}, \frac{4}{3}\Big)\Big] = [0, 4].$$

Функция f также удовлетворяет условиям теоремы, следовательно, можно найти искомое множество значений f.

$$\overline{F}(X) = G_1([4,8],[0,4]) = [g(4,0), g(8,4)] = [4, \sqrt{48}].$$

Особо следует отметить, что функция $q(x)$ не монотонна на интервале $[-2,2]$, поэтому условия Теоремы 1.6 не выполняются и соответствующие формулы не могут быть применены.

Пример носит искусственный характер в том смысле, что если мы знаем аналитическое выражение для функции $f(g(x), h(x))$ и знаем, что $f \in Z(X)$, то множество значений f проще всего вычислить, просто вычислив f в конечных точках интервала. Теорему 2.1 естественно применять в тех случаях, когда мы не знаем аналитического выражения для $g(x)$ и $h(x)$, например, когда $g(x)$ и $h(x)$ представлены алгоритмами с известными свойствами или измерениями физических величин.

Доказанная теорема допускает обобщение в двух направлениях. Во-первых, можно рассматривать базисные функции с количеством переменных, большим двух. Для базисной функции g от n переменных количество возможных расширений будет равно 2^n. Утверждение теоремы будет отличаться от случая двух переменных лишь одним: количество комбинаций характеров монотонности, а, следовательно, и количество строк в заключении теоремы, будет равно 2^{n+1}.

Другая возможность обобщения состоит в рассмотрении исходной функции f от нескольких аргументов [124]. В этом случае нас будет интересовать оценка для f

$$f(x_1, \ldots, x_n) = g(p(x_1, \ldots, x_n), q(x_1, \ldots, x_n))$$

при известных множествах значений $\overline{P}$ и $\overline{Q}$ для p и q. Пусть задан интервал $X = (X_1 \times \cdots \times X_n) \in I(R^n)$. Как и ранее, считаем, что все функции непрерывны на X. Пусть функции f, p и q монотонны по каждой переменной, т. е. если через $x_{i,0}$ обозначить любое фиксированное число из X_i, то

$$\begin{aligned}
\forall i \;\; f(x_{1,0}, \ldots, x_i, \ldots, x_{n,0}) \;&\in\; Z(X_i); \\
\forall i \;\; p(x_{1,0}, \ldots, x_i, \ldots, x_{n,0}) \;&\in\; Z(X_i); \\
\forall i \;\; q(x_{1,0}, \ldots, x_i, \ldots, x_{n,0}) \;&\in\; Z(X_i).
\end{aligned}$$

Причем характер монотонности по некоторой переменной не зависит от конкретных значений фиксированных переменных. Заметим, что не обязательно p и q зависят от всех x_i.

Сигнатурой σ монотонной функции f многих переменных на множестве X назовем набор элементов множества $\{+, -, 0\}$ длиной n. Будем обозначать $\sigma(f, X)$. Если функция возрастает по соответствующей переменной, то в наборе будет присутствовать $+$, если убывает, то $-$, если значение функции реально не зависит от переменной, то будет присутствовать 0. Например, для $s(x, y, z) = x - z$ имеем $\sigma(s, R^3) = (+, 0, -)$. Здесь и далее под монотонностью понимается нестрогая монотонность.

На множестве значений сигнатур введем унарную операцию $-$. Если $\sigma = (\alpha_1, \ldots, \alpha_n)$, то $-\sigma = (-\alpha_1, \ldots, -\alpha_n)$. Считаем, что $-(-) = +$, $-(+) = -$, $-(0) = 0$. Будем также говорить, что сигнатуры $\sigma_1 = (\alpha_1, \ldots, \alpha_n)$ и $\sigma_2 = (\beta_1, \ldots, \beta_n)$ совместны, и записывать $\sigma_1 \simeq \sigma_2$, если для всех i либо $\alpha_i = \beta_i$, либо $\alpha_i = 0$, либо $\beta_i = 0$.

Теорема 2.1 обобщается следующим образом.

Теорема 2.1а. Пусть задан интервал $X \in I(R^n)$ и на нем заданы две функции $p(x_1, \ldots, x_n)$, $q(x_1, \ldots, x_n)$. Пусть также для p и для q известны объединенные интервальные расширения $\overline{P}(X) = Y_1$ и $\overline{Q}(X) = Y_2$.

Пусть также функция $f(x_1,\ldots,x_n)$ задана как

$$f(x_1,\ldots,x_n)=g(p(x_1,\ldots,x_n),q(x_1,\ldots,x_n)),$$

где g – некоторая функция двух переменных, непрерывная на $Y_1\times Y_2$. Если оказывается, что

$$\sigma(f,X)\simeq\sigma(p,X)\ \vee\ \sigma(f,X)\simeq-\sigma(p,X)$$

и

$$\sigma(f,X)\simeq\sigma(q,X)\vee\sigma(f,X)\simeq-\sigma(q,X),$$

то для объединенного расширения $\overline{F}(X)$ имеем следующие выражения:

$$\begin{aligned}
\sigma(f,X)\simeq\sigma(p,X)\ \&\ \sigma(f,X)\simeq\sigma(q,X) &\ \Rightarrow\ \overline{F}(X)=G_1(Y_1,Y_2);\\
\sigma(f,X)\simeq\sigma(p,X)\ \&\ \sigma(f,X)\simeq-\sigma(q,X) &\ \Rightarrow\ \overline{F}(X)=G_3(Y_1,Y_2);\\
\sigma(f,X)\simeq-\sigma(p,X)\ \&\ \sigma(f,X)\simeq\sigma(q,X) &\ \Rightarrow\ \overline{F}(X)=G_2(Y_1,Y_2);\\
\sigma(f,X)\simeq-\sigma(p,X)\ \&\ \sigma(f,X)\simeq-\sigma(q,X) &\ \Rightarrow\ \overline{F}(X)=G_4(Y_1,Y_2).
\end{aligned}$$

Доказательство теоремы легко получить аналогично доказательству Теоремы 2.1, т. е. рассматривая значения функций в концевых точках промежутка. Рассмотрим, например, второй случай, т. е. $\sigma(f,X)\simeq\sigma(p,X)\ \&\ \sigma(f,X)\simeq-\sigma(q,X)$. Пусть $\sigma(f,X)=(\alpha_1,\ldots,\alpha_n)$. Не умаляя общности можно считать, что $\forall i\ \alpha_i\neq 0$. Выберем две точки $\overline{x}^-=(\overline{x}_1^-,\ldots,\overline{x}_n^-)\in X$ и $\overline{x}^+=(\overline{x}_1^+,\ldots,\overline{x}_n^+)\in X$ в соответствии со значением $\sigma(f,X)$. Если $\alpha_i=+$, то в качестве $\overline{x}_i^-$ возьмем x_i^- и в качестве $\overline{x}_i^+$ возьмем x_i^+. Если $\alpha_i=-$, то, наоборот, в качестве $\overline{x}_i^-$ возьмем x_i^+ и в качестве $\overline{x}_i^+$ возьмем x_i^-.

Имеем

$$\begin{aligned}
\overline{F}(X) &= [f(\overline{x}_1^-,\ldots,\overline{x}_n^-),f(\overline{x}_1^+,\ldots,\overline{x}_n^+)]\\
&= [g(p(\overline{x}_1^-,\ldots,\overline{x}_n^-),q(\overline{x}_1^-,\ldots,\overline{x}_n^-)),g(p(\overline{x}_1^+,\ldots,\overline{x}_n^+),q(\overline{x}_1^+,\ldots,\overline{x}_n^+))]
\end{aligned}$$

$$
\begin{aligned}
&= [g(Y_1^-, Y_2^+), g(Y_1^+, Y_2^-)] \\
&= G_3(Y_1, Y_2). \qquad \square
\end{aligned}
$$

2.2 Арифметика твинов

2.2.1 Обоснование подхода

Задача внешнего оценивания множества значений функции – одна из первых, если не первая задача, при решении которой использовались методы интервальной математики. Можно сказать, вся интервальная математика на ранних этапах своего развития сводилась к внешней оценке различных функций. Задачу внешнего оценивания мы сформулировали и рассмотрели выше. Соответствующее определение было дано в разделе 1.3. Кратко задачу можно сформулировать так: оценить множество значений данной функции при условии, что мы знаем оценки для всех ее аргументов. Под "оценить" как правило понимается "указать числовой интервал, который содержит все значения функции".

Задача внутренней оценки множества значений функции не так популярна, как задача внешнего оценивания, что, вероятно, вызвано меньшим количеством приложений, в которых она используется. Оценить множество значений функции изнутри – значит указать такой интервал, все точки которого гарантированно входят в действительное множество ее значений.

Интервальная арифметика в свое время была введена для автоматизации нахождения внешних оценок функции. В простейшем случае внешняя интервальная оценка может быть получена при помощи интервальной функции, которая формируется из оцениваемой функции путем замены всех базисных операций на их интервальные расширения. В случае внутренней оценки такой подход не приносит успеха.

При получении внутренней оценки для функции, являющейся суммой двух других, внутренняя оценка для которых уже известна, необходимо также знать внешние оценки этих двух функций. Таким образом, арифметики интервалов оказывается недостаточно для автоматического получения внутренней оценки. Решает проблему арифметика твинов, позволяющая одновременно получать и внутренние и внешние оценки функций.

Источник погрешностей оценок, с которыми мы сталкиваемся при применении стандартной интервальной арифметики, проявляется и при использовании арифметики твинов. Для борьбы с этим эффектом идея направленных интервалов может быть перенесена на арифметику твинов с получением направленной твинной арифметики. Весь дальнейший материал, изложенный в этой главе, будет посвящен одновременным внешним и внутренним оценкам, повышению их точности и изучению свойств введенных для этой цели математических объектов.

2.2.2 Задача внутренней оценки

Определение внешней оценки дано в разделе 1.3 (формулы (1.9), (1.10)).

Пусть дана функция $f(x_1, \ldots, x_n)$ и известно, что значение каждой переменной x_i пробегает точки интервала X_i, $x_i \in X_i$. Как и раньше, ограничимся рассмотрением случая, когда функция f непрерывна в рассматриваемой области и, следовательно, множества $\overline{F}(X_1, \ldots, X_n)$ являются интервалами.

Функцию $F_l(X_1, \ldots, X_n)$ назовем внутренней оценкой, если

$$\forall y \in F_l(X_1, \ldots, X_n)\ \exists x_1 \in X_1, \ldots, \exists x_n \in X_n :\ f(x_1, \ldots, x_n) = y.$$

Выполняется соотношение

$$F_l(X) \subseteq \overline{F}(X) \subseteq F(X).$$

Как и при рассмотрении внешней оценки, мы будем считать, что f задана в виде символьного выражения, составленного из "базисных" функций, таких как арифметические операции и элементарные функции.

2.2.3 Определение твинной арифметики

Твины в настоящей работе вводятся с целью одновременного получения внешних и внутренних оценок множества значений функции (см. [27], [102], [131]). Твином назовем пару интервалов $T = (X_l, X)$, $X_l \in I(R) \cup \{\emptyset\}$, $X \in I(R)$. $(X, X) = X$ – вырожденный твин, который может рассматриваться как интервал X, подобно тому как интервал $[x, x]$ отождествляется с вещественным числом x. Оценить неизвестный интервал I твином – значит найти такой твин $T = (X_l, X)$ что $X_l \subseteq I \subseteq X$. Обозначим это как $I \sqsubseteq T$.

Введем обозначения для "внешней длины твина" $|T|$ и "внутренней длины твина" $|T|_l$, $|T| = |X|$ и $|T|_l = |X_l|$. Запись $I \sqsubseteq (\emptyset, X)$ обозначает, что существует только внешняя оценка I, равная X. Будем формально полагать, что в этом случае $|T|_l = -1$.

Заметим, что как с содержательной, так и с формальной точек зрения разумно отождествлять твин $(\emptyset, [A, A])$ с твином $([A, A], [A, A])$. Действительно, из оценки интервала I, $I \sqsubseteq (\emptyset, [A, A])$ следует оценка $I \sqsubseteq ([A, A], [A, A])$ и наоборот. Для твина $T = ([A, A], [A, A])$ будем считать $|T|_l = 0$.

Основными свойствами, которым должны удовлетворять все операции вводимой арифметики твинов, являются следующие:

$$X \sqsubseteq T \;\Rightarrow\; \diamond X \sqsubseteq \diamond T; \tag{2.1}$$

$$X \sqsubseteq T_1 \,\&\, Y \sqsubseteq T_2 \;\Rightarrow\; X \circ Y \sqsubseteq T_1 \circ T_2, \tag{2.2}$$

где $\diamond$ и $\circ$, соответственно, любая унарная и любая бинарная операции.

Только выполнение этих свойств позволит проводить твинную оценку интервала путем одновременного выполнения операций над интервалом и оценивающим его твином. Пусть мы оцениваем одновременно снаружи и изнутри некоторую функцию $f_0(x_1, \ldots, x_n)$ при условии $x_i \in X_i$, $i = 1, \ldots, n$ и производим вычисления над твинами шаг за шагом в соответствии с аналитическим представлением f_0. Пусть на некотором шаге мы оцениваем функцию $f = g \circ h$. Для этого мы выполняем арифметическую операцию $T_1 \circ T_2$, где T_1 – твин, оценивающий функцию g, а T_2 – твин, оценивающий функцию h. Пусть $T_1 = ([a^-, a^+], [A^-, A^+])$, $T_2 = ([b^-, b^+], [B^-, B^+])$. Оценки, полученные на предыдущих шагах для функций g и h, содержательно означают, что при любых значениях $x_1, \ldots, x_n$, $x_i \in X_i$, выполняется $g(x_1, \ldots, x_n) \in [A^-, A^+]$ и $h(x_1, \ldots, x_n) \in [B^-, B^+]$, и, кроме того, какие бы точки y_1 из $[a^-, a^+]$ и y_2 из $[b^-, b^+]$ мы ни взяли, существуют наборы $x'_1, \ldots, x'_n$, $x'_i \in X_i$, и $x''_1, \ldots, x''_n$, $x''_i \in X_i$, при которых $g(x'_1, \ldots, x'_n) = y_1$ и $h(x''_1, \ldots, x''_n) = y_2$. Определение твинной операции $T_1 \circ T_2 = ([c^-, c^+], [C^-, C^+])$ должно обеспечивать, чтобы для функции $f = g \circ h$ при любых $x_1, \ldots, x_n$, $x_i \in X_i$, выполнялось $f(x_1, \ldots, x_n) \in [C^-, C^+]$ и для любого $y \in [c^-, c^+]$ существовал набор $x'''_1, \ldots, x'''_n$, $x'''_i \in X_i$, такой, что $f(x'''_1, \ldots, x'''_n) = y$. Особо необходимо отметить, что функции g и h, вообще говоря, будут взаимозависимы, т.е. зависеть от одних и тех же переменных из набора $(x_1, \ldots, x_n)$.

Определение 2.2. Если внутренние интервалы твинов T_1 и T_2 непусты, то обозначим

$$\begin{aligned} T_1 &= ([a^-, a^+], [A^-, A^+]), \\ T_2 &= ([b^-, b^+], [B^-, B^+]). \end{aligned}$$

Для таких твинов определим

$$p = \min(a^- + B^+, b^- + A^+); \tag{2.3}$$

$$q = \max(a^+ + B^-, b^+ + A^-). \tag{2.4}$$

Если мы имеем дело с вырожденным случаем, когда внутренний интервал твина T_1 или внутренний интервал твина T_2 является пустым множеством, то положим, соответственно, $p = b^- + A^+$, $q = b^+ + A^-$ и $p = a^- + B^+$, $q = a^+ + B^-$. Если оба твина одновременно являются вырожденными, то считаем p и q неопределенными.

Определим

$$\phi(I_1, I_2) = \tag{2.5}$$

$$\begin{cases} \min\limits_{\subseteq}\{[c^-, c^+] \mid (c^- \in I_1 \ \& \ c^+ \in I_2) \vee (c^- \in I_2 \ \& \ c^+ \in I_1)\} & \\ & \text{если } I_1 \cap I_2 = Z; \\ \emptyset & \text{в остальных случаях,} \end{cases}$$

где Z – пустое или одноэлементное множество.

$$\psi(I_1, I_2) = \max_{\subseteq}\{[c^-, c^+] \mid c^-, c^+ \in I_1 \cup I_2\}. \tag{2.6}$$

Впоследствии нам также понадобится вариант функции ψ от произвольного числа переменных

$$\psi(I_1, \ldots, I_n) = \max_{\subseteq}\{[c^-, c^+] \mid c^-, c^+ \in I_1 \cup \cdots \cup I_n\}.$$

Твинная арифметика задается следующими формулами.

$$T_1 + T_2 = \begin{cases} ([p, q], [A^- + B^-, A^+ + B^+]), \\ \qquad\qquad \text{если } |T_1| \le |T_2|_l \vee |T_2| \le |T_1|_l; \\ (\emptyset, \quad [A^- + B^-, A^+ + B^+]), \\ \qquad\qquad \text{в остальных случаях.} \end{cases} \tag{2.7}$$

Если два твина имеют непустые внутренние интервалы, то определим

$$\begin{aligned} T_1 \cdot T_2 \ = \ \Big(&\psi(\phi(a^-[B^-, B^+], a^+[B^-, B^+]), \phi(b^-[A^-, A^+], b^+[A^-, A^+])), \\ &[A^-, A^+] \cdot [B^-, B^+]\Big). \end{aligned} \tag{2.8}$$

Если $|T_1|_l = -1$, $|T_2|_l \ne -1$, то

$$T_1 \cdot T_2 = (\phi(b^-[A^-, A^+], b^+[A^-, A^+]), [A^-, A^+] \cdot [B^-, B^+]).$$

Если $|T_2|_l = -1$, $|T_1|_l \ne -1$, то

$$T_1 \cdot T_2 = (\phi(a^-[B^-, B^+], a^+[B^-, B^+]), [A^-, A^+] \cdot [B^-, B^+]).$$

Если $|T_2|_l = |T_1|_l = -1$, то

$$T_1 \cdot T_2 = (\emptyset, [A^-, A^+] \cdot [B^-, B^+]).$$

$$-T_1 = (-[a^-, a^+], -[A^-, A^+]);$$

$$1/T_1 = (1/[a^-, a^+], 1/[A^-, A^+]), \qquad 0 \notin [A^-, A^+].$$

Если внутренний интервал твина, к которому применяется одноместная арифметическая операция, пуст, то внутренний интервал твина-результата также пуст.

Теорема 2.2. Определение твинной арифметики удовлетворяет свойствам (2.1) и (2.2).

Доказательство. Для одноместных операций $-T$ и $1/T$ утверждение очевидно. Действительно, пусть $T = (Y_l, Y)$, и кроме того, формально полагаем, что $-\emptyset = \emptyset$, $1/\emptyset = \emptyset$.

$$\begin{aligned} X \sqsubseteq T \;&\Rightarrow\; Y_l \subseteq X \subseteq Y \\ &\Rightarrow\; -Y_l \subseteq -X \subseteq -Y \\ &\Rightarrow\; -X \sqsubseteq -T; \end{aligned}$$

$$\begin{aligned} X \sqsubseteq T \;&\Rightarrow\; Y_l \subseteq X \subseteq Y \\ &\Rightarrow\; 1/Y_l \subseteq 1/X \subseteq 1/Y \\ &\Rightarrow\; 1/X \sqsubseteq 1/T. \end{aligned}$$

Рассмотрим операцию сложения. Необходимо показать, что (во веденных ранее обозначениях):

1. $X \sqsubseteq T_1 \;\&\; Y \sqsubseteq T_2 \Rightarrow X + Y \subseteq [A^- + B^-, A^+ + B^+]$;
2. $X \sqsubseteq T_1 \;\&\; Y \sqsubseteq T_2 \Rightarrow$ если $|T_1| \leq |T_2|_l \vee |T_2| \leq |T_1|_l$, то $[p, q] \subseteq X + Y$.

Первая часть следует из определения стандартной интервальной арифметики и не нуждается в доказательстве.

Рассмотрим вторую часть. Пусть, например, $|T_2| \leq |T_1|_l$. Случай $|T_1| \leq |T_2|_l$ рассматривается аналогично. Естественно, имеем $|T_1|_l \neq -1$. Из того, что $a^-, a^+ \in X$ следует, что хотя бы по одной точке из интервалов $\delta_1 = a^- + [B^-, B^+]$ и $\delta_2 = a^+ + [B^-, B^+]$ принадлежит $X + Y$. Учитывая, что $\delta_1 \leq \delta_2$, получаем $[a^- + B^+, a^+ + B^-] \subseteq X + Y$. Если учесть, что в сделанном предположении $b^+ + A^- \leq a^+ + B^-$ и $b^- + A^+ \geq a^- + B^+$, то имеем $[p, q] \subseteq X + Y$.

Для доказательства того, что формула умножения твинов обладает свойством (2.2), достаточно показать, что

1. $X \sqsubseteq T_1 \;\&\; Y \sqsubseteq T_2 \Rightarrow X \cdot Y \subseteq [A^-, A^+] \cdot [B^-, B^+]$;

2. Любая точка интервала $\phi(a^-[B^-,B^+],a^+[B^-,B^+])$ и любая точка интервала $\phi(b^-[A^-,A^+],b^+[A^-,A^+])$ принадлежат интервалу $X \cdot Y$.

Первая часть утверждения, как и в случае сложения, вытекает из свойств интервальной арифметики. Для доказательства второй части в силу симметрии достаточно рассмотреть первый интервал. Из того, что $a^-,a^+ \in X$, следует, что хотя бы по одной точке из интервалов $\delta_1 = a^-[B^-,B^+]$ и $\delta_2 = a^+[B^-,B^+]$ принадлежит $X \cdot Y$. Следовательно, если интервалы δ_1 и δ_2 пересекаются не более чем в одной точке, то $\phi(a^-[B^-,B^+],a^+[B^-,B^+]) \subseteq X \cdot Y$.

Заметим, что приведенные рассуждения также доказывают утверждение в ситуации, когда один из внутренних интервалов перемножаемых твинов или оба таких интервала пусты. $\square$

Подобно тому, как мы обобщили унарные операции $-$ и $1/.$ на множество твинов, мы можем обобщить и элементарные функции одной переменной, для которых существуют интервальные расширения. Пусть $f(x)$ – непрерывная элементарная функция, для которой имеется интервальное расширение $F(X)$, которое по определению совпадает с объединенным интервальным расширением, т. е.

$$F(X) = \{y \mid \exists x \in X: \; y = f(x)\}.$$

Тогда расширение элементарной функции на множество твинов естественно определить следующим образом

$$F(T) = F(X_l, X) = (F(X_l), F(X)). \tag{2.9}$$

Естественно при этом полагать, что $F(\emptyset) = \emptyset$. Подразумевается также, что $F(X)$ существует для всех рассматриваемых интервалов X. Нетрудно видеть, что свойство (2.1) выполняется для всех определенных таким образом расширений элементарных функций.

Здесь и далее мы используем один и тот же (в данном случае F) функциональный символ для обозначения двух функций – интервальной функции и ее твинного расширения, надеясь, что это не вызовет недоразумений.

Рассмотрим теперь в общем случае задачу получения твинного расширения для непрерывной функции $f(x,y)$ двух переменных, для которой известно интервальное расширение $F(X,Y)$. Обозначим

$$\begin{aligned} T_1 &= (X_l, X) = ([a^-, a^+], [A^-, A^+]), \\ T_2 &= (Y_l, Y) = ([b^-, b^+], [B^-, B^+]). \end{aligned}$$

Для того, чтобы определить твинное расширение функции двух переменных, обозначим $(Z_l, Z) = F(T_1, T_2)$.

Для семейства интервалов I_j, $j \in J$ через $\Phi_{j\in J}(I_j)$ обозначим

$$\Phi_{j\in J}(I_j) = \begin{cases} [\min\limits_{j\in J} I_j^+, \max\limits_{j\in J} I_j^-], & \text{если границы существуют и} \\ & \text{интервал принадлежит } I(R); \\ \emptyset, & \text{в противном случае.} \end{cases}$$

Определим

$$Z = F(X,Y); \tag{2.10}$$

$$Z_l = \psi\Big(\Phi_{x\in X_l}(F(x,Y)), \Phi_{y\in Y_l}(F(X,y))\Big). \tag{2.11}$$

Заметим, что в силу непрерывности функции f величины $\min_{j\in J} I_j^+$ и $\max_{j\in J} I_j^-$ существуют (непрерывная на отрезке функция достигает в некоторых его точках своих максимума и минимума, см., например [28]).

Формула (2.11) используется в случае, если X_l и Y_l непусты. Если одно из этих множеств пусто, то соответствующий интервал Φ также полагается пустым.

Необходимо сделать следующее замечание. Введенные ранее определения твинных расширений для сложения и умножения вполне соответствуют только что рассмотренному общему случаю произвольных непрерывных функций двух аргументов. Докажем это.

1. Сложение.

$$Z_l = \psi\Big(\Phi_{x\in X_l}(x+Y), \Phi_{y\in Y_l}(X+y)\Big) =$$

$$= \psi\left(\begin{cases}[\min\limits_{x\in X_l}(x+Y)^+, \max\limits_{x\in X_l}(x+Y)^-], & \\ & \text{если интервал принадлежит } I(R) \\ \emptyset, & \text{в противном случае}\end{cases}, \begin{cases}[\min\limits_{y\in Y_l}(X+y)^+, \max\limits_{y\in Y_l}(X+y)^-], & \\ & \text{если интервал принадлежит } I(R) \\ \emptyset, & \text{в противном случае}\end{cases}\right) =$$

$$= \psi\left(\begin{cases}[a^- + B^+, a^+ + B^-] \\ \emptyset\end{cases}, \begin{cases}[b^- + A^+, b^+ + A^-] \\ \emptyset\end{cases}\right) =$$

$$= \begin{cases}[b^- + A^+, b^+ + A^-], & \text{если } |T_1| \le |T_2|_l, \\ [a^- + B^+, a^+ + B^-], & \text{если } |T_2| \le |T_1|_l, \\ \emptyset, & \text{в остальных случаях.}\end{cases}$$

Но так как если $|T_1| \le |T_2|_l$, то $b^- + A^+ \le a^- + B^+$ и $b^+ + A^- \ge a^+ + B^-$, и если $|T_2| \le |T_1|_l$, то $b^- + A^+ \ge a^- + B^+$ и $b^+ + A^- \le a^+ + B^-$, мы можем написать

$$Z_l = \begin{cases}[\min(a^- + B^+, b^- + A^+), \max(a^+ + B^-, b^+ + A^-)], & \\ & \text{если } |T_1| \le |T_2|_l \vee |T_2| \le |T_1|_l; \\ \emptyset, & \text{в остальных случаях.}\end{cases}$$

2. Умножение.

Достаточно заметить, что

$$\Phi_{x\in X_l}(x\cdot Y)=\phi(a^-[B^-,B^+],a^+[B^-,B^+]).$$

В случае непустых внутренних интервалов формула (2.8) оказывается равной формуле (2.11). Если один или оба внутренних интервала оказываются пустыми, то соответствующие формулы также совпадают.

Теорема 2.3. Определение твинного расширения, задаваемое формулами (2.10) и (2.11), удовлетворяет свойству (2.2).

Доказательство. Необходимо доказать, что

$$I\sqsubseteq T_1\ \&\ J\sqsubseteq T_2\ \Rightarrow\ F(I,J)\sqsubseteq F(T_1,T_2).$$

Это утверждение разбивается на два.

1. $F(I,J)\subseteq Z$;

2. $Z_l\subseteq F(I,J)$.

Здесь, как и ранее, Z и Z_l определены формулами (2.10) и (2.11).

Первая часть очевидна по определению интервального расширения функции f. Докажем вторую часть. Надо показать, что

$$\begin{aligned}\Phi_{x\in X_l}(F(x,Y))\ &\subseteq\ F(I,J)\qquad \text{и}\\ \Phi_{y\in Y_l}(F(X,y))\ &\subseteq\ F(I,J).\end{aligned}$$

Достаточно рассмотреть первый случай, второй рассматривается аналогично.

Зафиксируем точку $z\in\Phi_{x\in X_l}(F(x,Y))$. По определению Φ существуют интервалы $F(x_1,Y)\le z$ и $F(x_2,Y)\ge z$, $x_1,x_2\in X_l$. С другой стороны, т. к. $x_1\in X_l$ и $x_2\in X_l$, то хотя бы по одной точке интервалов $F(x_1,Y)$ и $F(x_2,Y)$ принадлежат $F(I,J)$. Назовем эти точки соответственно z_1 и z_2, $z_1\in F(I,J)$, $z_2\in F(I,J)$. Следовательно, $[z_1,z_2]\subseteq F(I,J)$ и $z\in F(I,J)$, что и требовалось доказать. □

2.3 Основные свойства твинной арифметики

Установим свойства введенной арифметики твинов.

Теорема 2.4. Сложение и умножение в арифметике твинов являются коммутативными и ассоциативными операциями, т. е.

$$T_1 + T_2 = T_2 + T_1;$$

$$(T_1 + T_2) + T_3 = T_1 + (T_2 + T_3);$$

$$T_1 \cdot T_2 = T_2 \cdot T_1;$$

$$(T_1 \cdot T_2) \cdot T_3 = T_1 \cdot (T_2 \cdot T_3).$$

Доказательство. Коммутативность по сложению и умножению прямо следует из определений. Действительно, формулы (2.3), (2.4) и (2.7) для сложения и формулы (2.5) и (2.8) для умножения не изменяются от перестановки местами, соответственно, слагаемых и сомножителей.

Докажем ассоциативность сложения. Положим

$$\begin{aligned}(T_1 + T_2) + T_3 &= (Z_l, Z);\\ T_1 + (T_2 + T_3) &= (Z'_l, Z').\end{aligned}$$

Ясно, что ввиду ассоциативности сложения интервалов $Z = Z'$. Необходимо показать, что $Z_l = Z'_l$. Обозначим

$$\begin{aligned}T_1 &= ([a^-, a^+], [A^-, A^+]);\\ T_2 &= ([b^-, b^+], [B^-, B^+]);\\ T_3 &= ([c^-, c^+], [C^-, C^+]).\end{aligned}$$

Естественно, если внутренние интервалы твинов пусты, то границы этих интервалов считаются неопределенными. Обозначим внешний интервал твина T через $\mathrm{ext}(T)$, а внутренний интервал через $\mathrm{int}(T)$.

Рассмотрим сперва случай, когда размер внутреннего интервала одного из твинов T_1, T_2, T_3 больше или равен сумме длин внешних интервалов двух других твинов. Пусть, например, $|T_1|_l \geq |T_2| + |T_3|$. Тогда

$$\begin{aligned}
Z_l &= \operatorname{int}((T_1 + T_2) + T_3) \\
&= \operatorname{int}(([a^- + B^+, a^+ + B^-], [A^- + B^-, A^+ + B^+]) + T_3) \\
&\qquad\qquad (\text{т. к. } |T_1|_l \geq |T_2|) \\
&= [a^- + B^+ + C^+, a^+ + B^- + C^-] \\
&\qquad\qquad (\text{т. к. } |T_1 + T_2|_l = |T_1|_l - |T_2| \geq |T_3|).
\end{aligned}$$

С другой стороны,

$$\begin{aligned}
Z'_l &= \operatorname{int}(T_1 + (T_2 + T_3)) \\
&= [a^-, a^+] + \operatorname{ext}(T_2 + T_3) \qquad (\text{т. к. } |T_1|_l \geq |T_2| + |T_3|) \\
&= [a^- + B^+ + C^+, a^+ + B^- + C^-].
\end{aligned}$$

Следовательно, $Z'_l = Z_l$.

Если размер внутреннего интервала другого твина превышает суммарный размер внешних интервалов двух оставшихся твинов, то проверка производится аналогично.

Рассмотрим теперь случай, когда

$$\begin{aligned}
|T_1|_l &< |T_2| + |T_3|; \\
|T_2|_l &< |T_1| + |T_3|; \\
|T_3|_l &< |T_1| + |T_2|.
\end{aligned}$$

Покажем, что в этом случае $\operatorname{int}((T_1 + T_2) + T_3) = \operatorname{int}(T_1 + (T_2 + T_3)) = \emptyset$. Имеем

$$|T_1 + T_2|_l \leq \max\{|T_1|_l - |T_2|, |T_2|_l - |T_1|, 0\},$$

но $|T_1|_l - |T_2| < |T_3|$ и $|T_2|_l - |T_1| < |T_3|$, откуда $|T_1 + T_2|_l < |T_3|$. С другой стороны, $|T_3|_l < |T_1| + |T_2| = |T_1 + T_2|$, следовательно, $\operatorname{int}((T_1 + T_2) +$

$T_3) = \emptyset$. Эти рассуждения верны и для случая пустых внутренних интервалов. Аналогично показывается, что $\text{int}(T_1 + (T_2 + T_3)) = \emptyset$. Ассоциативность сложения доказана.

Докажем ассоциативность умножения. Положим

$$\begin{aligned}(T_1 \cdot T_2) \cdot T_3 &= (Z_l, Z);\\ T_1 \cdot (T_2 \cdot T_3) &= (Z_l', Z').\end{aligned}$$

Как и в случае сложения, $Z = Z'$ из-за ассоциативности умножения интервалов. Необходимо показать, что $Z_l = Z_l'$. Если внутренние интервалы твинов непусты, то, как и ранее, обозначим

$$\begin{aligned}T_1 &= ([a^-, a^+], [A^-, A^+]);\\ T_2 &= ([b^-, b^+], [B^-, B^+]);\\ T_3 &= ([c^-, c^+], [C^-, C^+]).\end{aligned}$$

Случаи пустых внутренних интервалов будем рассматривать отдельно. Для произвольного интервала $[a, b]$ ненулевой длины обозначим $\text{op}([a, b]) = (a, b)$. Докажем вспомогательные утверждения.

Лемма 2.1. Если $0 \in (A^-, A^+)$ и $0 \in (B^-, B^+)$, то $\text{int}(T_1 \cdot T_2) = \emptyset$.

Доказательство. Действительно,

$$\begin{aligned}\phi(a^-[B^-, B^+], a^+[B^-, B^+]) &= \emptyset,\\ \phi(b^-[A^-, A^+], b^+[A^-, A^+]) &= \emptyset,\end{aligned}$$

следовательно,

$$\psi(\phi(a^-[B^-, B^+], a^+[B^-, B^+]), \phi(b^-[A^-, A^+], b^+[A^-, A^+])) = \emptyset.$$

$\square$

Лемма 2.2. Если $0 \in (A^-, A^+)$ и $\text{int}(T_1) = \emptyset$, то $\text{int}(T_1 \cdot T_2) = \emptyset$.

Доказательство очевидно, т. к. в этом случае

$$\phi(b^-[A^-,A^+],b^+[A^-,A^+]) = \emptyset.$$

$\square$

Продолжим доказательство теоремы.

1. Сперва рассмотрим случай $0 \in (A^-,A^+)$. Этот случай разбивается на три подслучая.

1.1. $0 \in (B^-,B^+)$. Тогда, очевидно, $0 \in \mathrm{op}(\mathrm{ext}(T_1 \cdot T_2))$, и кроме того, по Лемме 2.1 $\mathrm{int}(T_1 \cdot T_2) = \emptyset$. Следовательно, по Лемме 2.2 $\mathrm{int}((T_1 \cdot T_2) \cdot T_3) = \emptyset$. С другой стороны, $0 \in \mathrm{op}(\mathrm{ext}(T_2 \cdot T_3))$, следовательно, по Лемме 2.1 $\mathrm{int}(T_1 \cdot (T_2 \cdot T_3)) = \emptyset$.

1.2. $0 \in (C^-,C^+)$. Тогда $0 \in \mathrm{op}(\mathrm{ext}(T_1 \cdot T_2))$, и по Лемме 2.1 $\mathrm{int}((T_1 \cdot T_2) \cdot T_3) = \emptyset$. С другой стороны, $0 \in \mathrm{op}(\mathrm{ext}(T_2 \cdot T_3))$, следовательно, $\mathrm{int}(T_1 \cdot (T_2 \cdot T_3)) = \emptyset$.

1.3. $0 \le B^-$, $0 \le C^-$. Случаи, когда внешние интервалы твинов T_2 и T_3 не содержат нуля, но располагаются относительно него по-другому, рассматриваются аналогично.

1.3.1. $0 \in (a^-,a^+)$. В этом случае $\mathrm{int}(T_1 \cdot T_2) = [a^-B^+, a^+B^+]$ и, следовательно, $\mathrm{int}((T_1 \cdot T_2) \cdot T_3) = [a^-B^+C^+, a^+B^+C^+]$.

С другой стороны, $\mathrm{int}(T_1 \cdot (T_2 \cdot T_3)) = [a^- \cdot \mathrm{ext}(T_2 \cdot T_3), a^+ \cdot \mathrm{ext}(T_2 \cdot T_3)] = [a^-B^+C^+, a^+B^-C^-]$.

1.3.2. $0 \notin (a^-,a^+)$. Пусть $a^- > 0$, случай $a^- < 0$ рассматривается аналогично. В этом случае

$$\mathrm{int}(T_1 \cdot T_2) = \begin{cases} [a^-B^+, a^+B^-], & \text{если } a^-B^+ \le a^+B^-, \\ \emptyset, & \text{в противном случае,} \end{cases}$$

и, следовательно,

$$\mathrm{int}((T_1 \cdot T_2) \cdot T_3) = \begin{cases} [a^-B^+C^+, a^+B^-C^-], & \text{если } a^-B^+C^+ \le a^+B^-C^-, \\ \emptyset, & \text{в противном случае.} \end{cases}$$

С другой стороны,

$$\begin{aligned}\operatorname{int}(T_1\cdot(T_2\cdot T_3)) &= \phi(a^-\cdot\operatorname{ext}(T_2\cdot T_3), a^+\cdot\operatorname{ext}(T_2\cdot T_3)]\\ &= \begin{cases}[a^-B^+C^+, a^+B^-C^-], & \text{если } a^-B^+C^+\le a^+B^-C^-,\\ \emptyset, & \text{в противном случае.}\end{cases}\end{aligned}$$

Мы завершили рассмотрение первого случая $0\in(A^-,A^+)$. Случаи $0\in(B^-,B^+)$ и $0\in(C^-,C^+)$ рассматриваются абсолютно аналогично.

2. Рассмотрим случай, когда $A^-\ge 0\ \&\ B^-\ge 0\ \&\ C^-\ge 0$. Этот случай завершит доказательство. Ситуация, когда внешние интервалы всех трех твинов не содержат внутри себя ноль, но располагаются по другую/разные от него стороны, рассматривается аналогично.

2.1. Предположим, что размер внутреннего интервала одного из твинов (твина T_1) превышает размеры внешних интервалов двух других твинов в следующем смысле: $a^+B^-C^-\ \ge\ a^-B^+C^+$. Случаи $b^+A^-C^-\ge b^-A^+C^+$ и $c^+B^-A^-\ge c^-B^+A^+$ рассматриваются аналогично. Из предположения, принимая во внимание то, что $C^+\ge C^-$, имеем $a^+B^-\ \ge\ B^+a^-$. Одновременно получаем $b^-A^+\ \ge\ B^-A^+\ \ge\ B^-a^+\ \ge\ B^+a^-\ \ge\ b^+a^-\ \ge\ b^+A^-$, причем $b^-A^+=b^+A^-$ если и выполняется, то лишь тогда, когда $a^-B^+=a^+B^-=b^-A^+=b^+A^-$, следовательно, $\operatorname{int}(T_1\cdot T_2)=[a^-B^+,a^+B^-]$.

Из предположения следует, что $a^-B^+C^+\le a^+B^-C^-$ и в то же время $c^-A^+B^+\ge C^-A^+B^+\ge C^-a^+B^+\ge C^-a^+B^-\ge B^+C^+a^-\ge c^+A^-B^-$. Следовательно, $\operatorname{int}((T_1\cdot T_2)\cdot T_3)=[a^-B^+C^+,a^+B^-C^-]$.

Далее имеем $\operatorname{ext}(T_2\cdot T_3)=[B^-C^-,B^+C^+]$. Кроме того, $a^+B^-C^-\ge a^-B^+C^+$, следовательно, $\operatorname{int}(T_1\cdot(T_2\cdot T_3))=[a^-B^+C^+,a^+B^-C^-]$.

2.2. Пусть

$$a^+B^-C^-\ <\ B^+C^+a^-,$$

$$\begin{aligned} b^+A^-C^- &< A^+C^+b^-, \\ c^+B^-A^- &< B^+A^+c^-. \end{aligned}$$

Тогда можно показать, что

$$\operatorname{int}((T_1 \cdot T_2) \cdot T_3) = \operatorname{int}(T_1 \cdot (T_2 \cdot T_3)) = \emptyset.$$

Действительно, $\operatorname{int}(T_1 \cdot T_2)$ если не пусто, то равно либо $[a^-B^+, a^+B^-]$, либо $[b^-A^+, b^+A^-]$, но множество $\phi(a^-B^+[C^-, C^+], a^+B^-[C^-, C^+])$ является пустым, так как по предположению $a^-B^+C^+ > a^+B^-C^-$. Аналогично, множество $\phi(b^-A^+[C^-, C^+], b^+A^-[C^-, C^+])$ также является пустым.

Кроме того, имеем $c^-A^+B^+ > c^+A^-B^-$, следовательно,

$$\phi(c^-[A^-B^-, A^+B^+], c^+[A^-B^-, A^+B^+]) = \emptyset.$$

Получаем $\operatorname{int}((T_1 \cdot T_2) \cdot T_3) = \emptyset$. Совершенно аналогично, в силу возможности поменять местами T_1, T_2 и T_3 получаем $\operatorname{int}(T_1 \cdot (T_2 \cdot T_3)) = \emptyset$. Ассоциативность умножения доказана. $\square$

Установим еще ряд свойств твинной арифметики.

Единственными нейтральными элементами для, соответственно, сложения и умножения являются твины

$$\begin{aligned} T_0 &= ([0,0],[0,0]) \qquad \text{и} \\ T_1 &= ([1,1],[1,1]). \end{aligned}$$

То, что это действительно нейтральные элементы, проверяется по определению. Пусть $T' = T + T_0$. Предположим сперва, что $\operatorname{int}(T) \neq \emptyset$. Обозначим $T = ([a^-, a^+], [A^-, A^+])$. Имеем $\operatorname{ext}(T') = \operatorname{ext}(T)$ по определению интервальной арифметики. $\operatorname{int}(T') = [\min(a^-, A^+), \max(a^+, A^-)] = [a^-, a^+] = \operatorname{int}(T)$. Если $\operatorname{int}(T) = \emptyset$, то $\operatorname{int}(T') = [A^+, A^-] = \emptyset$.

Пусть теперь $T' = T \cdot T_1$. По определению интервальной арифметики $\mathrm{ext}(T') = \mathrm{ext}(T)$. Предположим сперва, что $\mathrm{int}(T) \neq \emptyset$. Имеем $\phi([a^-, a^-], [a^+, a^+]) = [a^-, a^+]$, $\phi([A^-, A^+], [A^-, A^+]) = \emptyset$, следовательно, $\psi([a^-, a^+], \emptyset) = [a^-, a^+]$, т. е. $\mathrm{int}(T') = [a^-, a^+] = \mathrm{int}(T)$. Если $\mathrm{int}(T) = \emptyset$, то $\mathrm{int}(T') = \phi([A^-, A^+], [A^-, A^+]) = \emptyset$.

Предположим, что существуют другие нейтральные элементы T_0' и T_1'. Тогда $T_0 + T_0' = T_0$ и $T_0' + T_0 = T_0'$. Из коммутативности сложения следует, что $T_0' = T_0$. Абсолютно аналогично доказывается единственность нейтральных элементов для умножения.

Множество твинов не имеет делителей нуля, за исключением твинов вида $(\emptyset, [0, 0])$. Это означает, что из того, что $T_1 \cdot T_2 = 0$ следует, что хотя бы один из твинов T_1 и T_2 равен $([0, 0], [0, 0])$ или $(\emptyset, [0, 0])$. Докажем это. Пусть $T_1 \cdot T_2 = 0$. Внешний интевал одного из твинов T_1 и T_2 обязан быть равным $[0, 0]$, так как множество интервалов не имеет делителей нуля. Пусть, например, $\mathrm{ext}(T_1) = [0, 0]$. Но тогда из того, что $\mathrm{int}(T_1) \subseteq \mathrm{ext}(T_1)$ имеем: либо $\mathrm{int}(T_1) = [0, 0]$, либо $\mathrm{int}(T_1) = \emptyset$ (см. также примечание об отождествлении твинов $(\emptyset, [x, x])$ и $([x, x], [x, x])$ в начале параграфа 2.2.3.).

Закон дистрибутивности для твинов не выполняется хотя бы потому, что он не выполняется для интервалов. Действительно, для выполнения закона дистрибутивности необходимо, чтобы для произвольных твинов T_1, T_2 и T_3 выполнялось

$$T_1 \cdot (T_2 + T_3) = T_1 \cdot T_2 + T_1 \cdot T_3$$

и, в частности,

$$\mathrm{ext}(T_1 \cdot (T_2 + T_3)) = \mathrm{ext}(T_1 \cdot T_2 + T_1 \cdot T_3),$$

что, как известно, не имеет места.

Введем для двух твинов отношение включения. Пусть $T_1 = (I_{1l}, I_1)$, $T_2 = (I_{2l}, I_2)$.

$$T_1 \subseteq T_2 \;\Leftrightarrow\; (I_1 \subseteq I_2) \;\&\; (I_{2l} \subseteq I_{1l}). \tag{2.12}$$

Заметим, что отношение $\subseteq$ на множестве твинов не является теоретико-множественным включением.

Ослаблением свойства дистрибутивности является субдистрибутивность. Обозначим, как и ранее

$$\begin{aligned} T_1 &= ([a^-, a^+], [A^-, A^+]) = (I_{1l}, I_1) \\ T_2 &= ([b^-, b^+], [B^-, B^+]) = (I_{2l}, I_2) \\ T_3 &= ([c^-, c^+], [C^-, C^+]) = (I_{3l}, I_3) \end{aligned}$$

Теорема 2.5. Для трех произвольных твинов имеет место следующее свойство

$$T_1 \cdot (T_2 + T_3) \subseteq T_1 \cdot T_2 + T_1 \cdot T_3.$$

Доказательство. Как известно (см., например, [1]), для множества интервалов выполняется свойство субдистрибутивности, т. е.

$$\mathrm{ext}(T_1 \cdot (T_2 + T_3)) = I_1 \cdot (I_2 + I_3) \subseteq I_1 \cdot I_2 + I_1 \cdot I_3 = \mathrm{ext}(T_1 \cdot T_2 + T_1 \cdot T_3).$$

Остается показать, что

$$\mathrm{int}(T_1 \cdot (T_2 + T_3)) \supseteq \mathrm{int}(T_1 \cdot T_2 + T_1 \cdot T_3).$$

Имеем

$$\begin{aligned} &\mathrm{int}(T_1 \cdot T_2 + T_1 \cdot T_3) \qquad (2.13) \\ &\quad = \psi\Big(\Phi_{x \in \mathrm{int}(T_1 \cdot T_2)}(x + \mathrm{ext}(T_1 \cdot T_3)), \Phi_{x \in \mathrm{int}(T_1 \cdot T_3)}(\mathrm{ext}(T_1 \cdot T_2) + x)\Big) \\ &\quad = \psi\Big(\Phi_{x \in \mathrm{int}(T_1 \cdot T_2)}(x + I_1 \cdot I_3), \Phi_{x \in \mathrm{int}(T_1 \cdot T_3)}(I_1 \cdot I_2 + x)\Big). \end{aligned}$$

Докажем, что

$$\Phi_{x\in\operatorname{int}(T_1\cdot T_2)}(x+I_1\cdot I_3) = \psi(\Phi_{y\in I_{1l}}(y\cdot I_2+I_1\cdot I_3), \Phi_{y\in I_{2l}}(I_1\cdot y+I_1\cdot I_3)). \quad (2.14)$$

Возьмем точку $z \in \Phi_{x\in\operatorname{int}(T_1\cdot T_2)}(x + I_1 \cdot I_3)$. По определению Φ имеем

$$\exists x_1, x_2 \in \operatorname{int}(T_1 \cdot T_2): \quad (x_1 + I_1 \cdot I_3)^+ \le z \le (x_2 + I_1 \cdot I_3)^-.$$

Тогда либо

$$\exists y_1 \in I_{1l}: \quad (y_1 \cdot I_2 + I_1 \cdot I_3)^+ \le z,$$

либо

$$\exists y_2 \in I_{2l}: \quad (I_1 \cdot y + I_1 \cdot I_3)^+ \le z.$$

Следовательно, $z \ge z_1$, где z_1 – левая граница выражения

$$\psi(\Phi_{y\in I_{1l}}(y \cdot I_2 + I_1 \cdot I_3), \Phi_{y\in I_{2l}}(I_1 \cdot y + I_1 \cdot I_3)) = Z_1.$$

Аналогично получаем, что $z \le z_2$, где z_2 – правая граница Z_1. Следовательно, $z \in Z_1$.

Возьмем теперь $z' \in Z_1$. Заметив, что все проведенные рассуждения обратимы, получаем, что $z' \in \Phi_{x\in\operatorname{int}(T_1\cdot T_2)}(x+I_1\cdot I_3)$, и утверждение (2.14) доказано.

Аналогично получаем, что

$$\Phi_{x\in\operatorname{int}(T_1\cdot T_3)}(I_1 \cdot I_2 + x) = \psi(\Phi_{y\in I_{1l}}(I_1 \cdot I_2 + y \cdot I_3), \Phi_{y\in I_{3l}}(I_1 \cdot I_2 + I_1 \cdot y)).$$

Продолжая равенство (2.13), имеем

$$\begin{aligned}
&\operatorname{int}(T_1 \cdot T_2 + T_1 \cdot T_3) && (2.15)\\
&\quad = \psi\Big(\psi(\Phi_{y\in I_{1l}}(y \cdot I_2 + I_1 \cdot I_3), \Phi_{y\in I_{2l}}(I_1 \cdot y + I_1 \cdot I_3)),\\
&\qquad\quad \psi(\Phi_{y\in I_{1l}}(I_1 \cdot I_2 + y \cdot I_3), \Phi_{y\in I_{3l}}(I_1 \cdot I_2 + I_1 \cdot y))\Big)\\
&\quad = \psi\Big(\Phi_{y\in I_{1l}}(y \cdot I_2 + I_1 \cdot I_3), \Phi_{y\in I_{2l}}(I_1 \cdot y + I_1 \cdot I_3),\\
&\qquad\quad \Phi_{y\in I_{1l}}(I_1 \cdot I_2 + y \cdot I_3), \Phi_{y\in I_{3l}}(I_1 \cdot I_2 + I_1 \cdot y)\Big).
\end{aligned}$$

С другой стороны,

$$\begin{aligned}
&\operatorname{int}(T_1 \cdot (T_2 + T_3)) && (2.16)\\
&\quad = \psi\Big(\Phi_{x\in I_{1l}}(x \cdot \operatorname{ext}(T_2 + T_3)), \Phi_{x\in \operatorname{int}(T_2+T_3)}(I_1 \cdot x)\Big)\\
&\quad = \psi\Big(\Phi_{x\in I_{1l}}(x \cdot (I_2 + I_3)), \Phi_{x\in \operatorname{int}(T_2+T_3)}(I_1 \cdot x)).
\end{aligned}$$

Рассуждения, аналогичные тем, что были проведены для доказательства равенства (2.14), позволяют заключить, что

$$\Phi_{x\in \operatorname{int}(T_2 \cdot T_3)}(I_1 \cdot x) = \psi(\Phi_{y\in I_{2l}}(I_1 \cdot (y + I_3)), \Phi_{y\in I_{3l}}(I_1 \cdot (I_2 + y)).$$

Продолжая равенство (2.16), имеем

$$\begin{aligned}
&\operatorname{int}(T_1 \cdot (T_2 + T_3)) && (2.17)\\
&\quad = \psi\Big(\Phi_{x\in I_{1l}}(x \cdot (I_2 + I_3)),\\
&\qquad \psi(\Phi_{y\in I_{2l}}(I_1 \cdot (y + I_3)), \Phi_{y\in I_{3l}}(I_1 \cdot (I_2 + y)))\Big)\\
&\quad = \psi\Big(\Phi_{x\in I_{1l}}(x \cdot (I_2 + I_3)), \Phi_{y\in I_{2l}}(I_1 \cdot (y + I_3)), \Phi_{y\in I_{3l}}(I_1 \cdot (I_2 + y))\Big).
\end{aligned}$$

Нетрудно видеть, что (см., например, [1])

$$\begin{aligned}
\forall x \in I_1 \quad x \cdot I_2 + I_1 \cdot I_3 \ &\supseteq \ x \cdot I_2 + x \cdot I_3 = x \cdot (I_2 + I_3),\\
\forall x \in I_1 \quad I_1 \cdot I_2 + x \cdot I_3 \ &\supseteq \ x \cdot I_2 + x \cdot I_3 = x \cdot (I_2 + I_3),
\end{aligned}$$

следовательно,

$$\Phi_{x\in I_{1l}}(x \cdot I_2 + I_1 \cdot I_3) \ \subseteq \ \Phi_{x\in I_{1l}}(x \cdot (I_2 + I_3)), \qquad (2.18)$$

$$\Phi_{x\in I_{1l}}(I_1 \cdot I_2 + x \cdot I_3) \ \subseteq \ \Phi_{x\in I_{1l}}(x \cdot (I_2 + I_3)). \qquad (2.19)$$

Кроме того, из-за субдистрибутивности, выполняющейся в $I(R)$,

$$\begin{aligned}
\forall y \in I_2 \quad I_1 \cdot y + I_1 \cdot I_3 \ &\supseteq \ I_1 \cdot (y + I_3),\\
\forall y \in I_3 \quad I_1 \cdot I_2 + I_1 \cdot y \ &\supseteq \ I_1 \cdot (I_2 + y),
\end{aligned}$$

следовательно,

$$\begin{aligned}
\Phi_{y\in I_{2l}}(I_1\cdot y + I_1\cdot I_3) &\subseteq \Phi_{y\in I_{2l}}(I_1\cdot(y+I_3)) && (2.20)\\
\Phi_{y\in I_{3l}}(I_1\cdot I_2 + I_1\cdot y) &\subseteq \Phi_{y\in I_{3l}}(I_1\cdot(I_2+y)). && (2.21)
\end{aligned}$$

Таким образом, имея в виду (2.18), (2.19), (2.20), и (2.21), мы получили из (2.15) и (2.17), что

$$\mathrm{int}(T_1\cdot(T_2+T_3)) \supseteq \mathrm{int}(T_1\cdot T_2 + T_1\cdot T_3).$$

□

Докажем две теоремы, установленные автором в работе [132].

Теорема 2.6. Если все твинные операции определены, т. е. не производится деления на твин, внешний интервал которого содержит 0, то для введенных операций $\diamond \in \{-, 1/.\}$ и $\circ \in \{+, \cdot\}$ и произвольных твинов T_i, S_i имеем

$$\begin{aligned}
T_1 \subseteq S_1 \quad &\Rightarrow \quad \diamond T_1 \subseteq \diamond S_1,\\
(T_1 \subseteq S_1)\ \&\ (T_2 \subseteq S_2) \quad &\Rightarrow \quad T_1 \circ T_2 \subseteq S_1 \circ S_2,
\end{aligned}$$

т. е. арифметика твинов обладает свойством монотонности по включению.

Доказательство. Если внутренние интервалы твинов непусты, то обозначим

$$\begin{aligned}
T_1 &= (I_{1l}, I_1) = ([a_1^-, a_1^+], [A_1^-, A_1^+]);\\
T_2 &= (I_{2l}, I_2) = ([a_2^-, a_2^+], [A_2^-, A_2^+]);\\
S_1 &= (J_{1l}, J_1) \doteq ([b_1^-, b_1^+], [B_1^-, B_1^+]);\\
S_2 &= (J_{2l}, J_2) = ([b_2^-, b_2^+], [B_2^-, B_2^+]).
\end{aligned}$$

1. Вычитание.

$$\begin{aligned} T_1 \subseteq S_1 \quad &\Rightarrow \quad (J_{1l} \subseteq I_{1l}) \;\&\; (I_1 \subseteq J_1) \\ &\Rightarrow \quad (-J_{1l} \subseteq -I_{1l}) \;\&\; (-I_1 \subseteq -J_1) \\ &\Rightarrow \quad -T_1 \subseteq -S_1. \end{aligned}$$

2. Взятие обратного элемента.

$$\begin{aligned} T_1 \subseteq S_1 \quad &\Rightarrow \quad (J_{1l} \subseteq I_{1l}) \;\&\; (I_1 \subseteq J_1) \\ &\Rightarrow \quad (1/J_{1l} \subseteq 1/I_{1l}) \;\&\; (1/I_1 \subseteq 1/J_1) \\ &\Rightarrow \quad 1/T_1 \subseteq 1/S_1. \end{aligned}$$

3. Сложение. Если $(T_1 \subseteq S_1) \;\&\; (T_2 \subseteq S_2)$, то по свойствам интервальной арифметики

$$\mathrm{ext}(T_1 + T_2) = I_1 + I_2 \;\subseteq\; J_1 + J_2 = \mathrm{ext}(S_1 + S_2).$$

Остается показать, что

$$\mathrm{int}(T_1 + T_2) \supseteq \mathrm{int}(S_1 + S_2).$$

Пусть внутренние интервалы всех рассматриваемых твинов непусты. Тогда имеем

$$\begin{aligned} \mathrm{int}(T_1 + T_2) &= [\min(a_1^- + A_2^+, a_2^- + A_1^+), \max(a_1^+ + A_2^-, a_2^+ + A_1^-)] \\ &\supseteq [\min(b_1^- + B_2^+, b_2^- + B_1^+), \max(b_1^+ + B_2^-, b_2^+ + B_1^-)] \\ &= \mathrm{int}(S_1 + S_2). \end{aligned}$$

Если $J_{2l} = \emptyset$, $J_{1l} \neq \emptyset$ и $I_{2l} \neq \emptyset$, то имеем

$$\begin{aligned} \mathrm{int}(T_1 + T_2) &= [\min(a_1^- + A_2^+, a_2^- + A_1^+), \max(a_1^+ + A_2^-, a_2^+ + A_1^-)] \\ &\supseteq [b_1^- + B_2^+, b_1^+ + B_2^-] \\ &= \mathrm{int}(S_1 + S_2). \end{aligned}$$

Если $J_{2l} = \emptyset$, $J_{1l} \neq \emptyset$, $I_{2l} = \emptyset$, то имеем

$$\begin{aligned}
\mathrm{int}(T_1 + T_2) &= [a_1^- + A_2^+, a_1^+ + A_2^-] \\
&\supseteq [b_1^- + B_2^+, b_1^+ + B_2^-] \\
&= \mathrm{int}(S_1 + S_2).
\end{aligned}$$

Аналогично рассматриваются случаи, когда $J_{1l} = \emptyset$.

Если $J_{1l} = J_{2l} = \emptyset$, то $\mathrm{int}(S_1 + S_2) = \emptyset$ и утверждение очевидно.

4. Умножение. Если $(T_1 \subseteq S_1)$ & $(T_2 \subseteq S_2)$, то по свойствам интервальной арифметики

$$\mathrm{ext}(T_1 \cdot T_2) = I_1 \cdot I_2 \subseteq J_1 \cdot J_2 = \mathrm{ext}(J_1 \cdot J_2).$$

Остается показать, что

$$\mathrm{int}(T_1 \cdot T_2) \supseteq \mathrm{int}(S_1 \cdot S_2).$$

Докажем, что

$$\phi_1 = \phi(a_1^- \cdot [A_2^-, A_2^+], a_1^+ \cdot [A_2^-, A_2^+]) \supseteq \phi(b_1^- \cdot [B_2^-, B_2^+], b_1^+ \cdot [B_2^-, B_2^+]) = \phi_2.$$

Рассмотрим ряд случаев.

1. $0 \in (A_2^-, A_2^+)$. В этом случае также $0 \in (B_2^-, B_2^+)$ и, следовательно, $\phi_1 = \phi_2 = \emptyset$.

2. $A_2^- \geq 0$. (Случай $A_2^+ \leq 0$ рассматривается аналогично.)

2.1. $a_1^- \geq 0$. (Случай $a_1^+ \leq 0$ рассматривается аналогично.)

$$\phi_1 = [a_1^- A_2^+, a_1^+ A_2^-] \supseteq [b_1^- B_2^+, b_1^+ B_2^-] = \phi_2.$$

Последнее включение верно, если интервал в левой части непуст. Если он пуст, то интервал в правой части также пуст и утверждение очевидно.

2.2. $0 \in (a_1^- a_1^+)$.

$$\phi_1 = [a_1^- A_2^+, a_1^+ A_2^+] \supseteq [b_1^- B_2^+, b_1^+ B_2^+] = \phi_2.$$

Аналогично показывается, что

$$\phi(a_2^- \cdot [A_1^-, A_1^+], a_2^+ \cdot [A_1^-, A_1^+]) \supseteq \phi(b_2^- \cdot [B_1^-, B_1^+], b_2^+ \cdot [B_1^-, B_1^+]),$$

следовательно,

$$\psi(\phi(a_1^- \cdot [A_2^-, A_2^+], a_1^+ \cdot [A_2^-, A_2^+]), \phi(a_2^- \cdot [A_1^-, A_1^+], a_2^+ \cdot [A_1^-, A_1^+]))$$
$$\supseteq \psi(\phi(b_1^- \cdot [B_2^-, B_2^+], b_1^+ \cdot [B_2^-, B_2^+]), \phi(b_2^- \cdot [B_1^-, B_1^+], b_2^+ \cdot [B_1^-, B_1^+])).$$

□

Аналогично Теореме 1.1, которая называется основной теоремой интервальной арифметики, верна следующая теорема, которую можно назвать основной теоремой твинной арифметики.

Теорема 2.7. Пусть $F(T_1, \ldots, T_n)$ – выражение от n твинных переменных T_i, построенное с применением операций сложения, умножения, унарного минуса и взятия обратного элемента. Иными словами, $F(T_1, \ldots, T_n)$ – конечная комбинация твинов $T_1, \ldots, T_n$ и твинных констант, соединенных твинными операциями. Тогда из $S_i \subseteq Q_i$, $i = 1, \ldots, n$, следует

$$F(S_1, \ldots, S_n) \subseteq F(Q_1, \ldots, Q_n)$$

при любом наборе твинов $Q_1, \ldots, Q_n$, для которого твинные операции в выражении имеют смысл (т. е. не встречается взятие обратного элемента от твина, внешний интервал которого содержит ноль).

Доказательство теоремы очевидно. Оно непосредственно вытекает из Теоремы 2.6, которая применяется столько раз, сколько функциональных символов входит в запись функции F.

□

2.4 Дальнейшие обобщения

2.4.1 Направленные твины

2.4.1.1 Направленная твинная арифметика

Направленные интервалы были введены с целью более точной оценки множества значений функции в случае, когда доступна дополнительная информация о характере монотонности оцениваемых функций. Подобно этому можно ввести и направленные твины для более точной одновременной внутренней и внешней оценок.

Вместо монотонности в контексте направленных твинов будем рассматривать т. н. слабую монотонность (см. [123]), или, как мы называли это свойство ранее, – условие преобладания концевых точек (см. раздел 2.1, определение 2.1). Будем говорить, что функция $f(x)$ слабо возрастает на отрезке $[a, b]$, если $f(x) \in Z_{\nearrow}([a, b])$, и что она слабо убывает, если $f(x) \in Z_{\searrow}([a, b])$.

Функцию назовем слабомонотонной, если она либо слабо возрастает, либо слабо убывает. Любая непрерывная монотонная функция является слабо монотонной функцией.

Определение 2.3. Направленный твин есть пара (T, α), T – твин, $\alpha \in \{+, -\}$. Будем обозначать $\overline{F}(X) \sqsubseteq D = (I_l, I, \alpha)$ если известно, что $I_l \subseteq \overline{F}(X) \subseteq I$ и тип слабой монотонности функции f на X совпадает с α ("+" – функция слабо возрастает; "−" – функция слабо убывает).

Для общности мы также рассмотрим твины в форме $(T, 0)$, которая является вырожденной формой направленного твина, эквивалентного обычному твину. Оценка $\overline{F}(X) \sqsubseteq (T, 0)$ означает, что $\overline{F}(X) \sqsubseteq T$ и мы не имеем информации о монотонности f.

Как и в случае обычных твинов, мы не будем отдельно рассматривать твины вида $(\emptyset, [a, a], \alpha)$, отождествив их с $([a, a], [a, a], \alpha)$.

Здесь и далее мы будем обозначать направленные твины и как $(T, \alpha) = ((I_l, I), \alpha)$, и как (I_l, I, α).

Основными свойствами, которым должна удовлетворять арифметика направленных твинов, являются следующие: пусть $D_1 = (T_1, \alpha)$ и $D_2 = (T_2, \beta)$ – направленные твины. Если $g(x)$, $h(x)$ и $f(x) = g(x) \circ h(x)$ непрерывны и, кроме того, оказывается, что $f(x)$ – слабо монотонна, то

$$\overline{G}(X) \sqsubseteq D_1 \;\Rightarrow\; \overline{\diamond G}(X) \sqsubseteq \diamond D_1; \tag{2.22}$$

$$\overline{G}(X) \sqsubseteq D_1 \;\&\; \overline{H}(X) \sqsubseteq D_2 \;\Rightarrow\; \overline{G \circ H}(X) \sqsubseteq D_1 \circ D_2; \tag{2.23}$$

где $\diamond$ и $\circ$ – унарные и бинарные арифметические операции.

Определение 2.4. Арифметику направленных твинов зададим следующими формулами. Пусть D_1 и D_2 – направленные твины.

$$D_1 = (I_{1l}, I_1, \alpha) = (T_1, \alpha) = ([a^-, a^+], [A^-, A^+], \alpha);$$

$$D_2 = (I_{2l}, I_2, \beta) = (T_2, \beta) = ([b^-, b^+], [B^-, B^+], \beta),$$

$\alpha, \beta \in \{-, +, 0\}$. Естественно полагать, что a^-, a^+, b^-, b^+ имеют смысл лишь в том случае, если соответствующие внутренние интервалы непусты. Для символов "+" и "–" будем формально полагать, что $+- = -+ = -$, $++ = -- = +$.

Сложение. Направление твина, являющегося суммой двух других, задается формулой

$$\gamma = \begin{cases} \alpha, & \text{если } \alpha = \beta \neq 0 \;\&\; I_{1l} \neq \emptyset \;\&\; I_{2l} \neq \emptyset; \\ \alpha, & \text{если } \alpha \neq 0 \;\&\; I_{1l} \neq \emptyset \;\&\; \alpha \neq \beta \;\&\; a^+ + B^- \geq B^+ + a^-; \\ \beta, & \text{если } \beta \neq 0 \;\&\; I_{2l} \neq \emptyset \;\&\; \alpha \neq \beta \;\&\; A^+ + b^- \leq b^+ + A^-; \\ 0, & \text{в остальных случаях.} \end{cases}$$

Если $\alpha = \beta \neq 0$, $I_{1l} \neq \emptyset$ и $I_{2l} \neq \emptyset$ то

$$D_1 + D_2 = ([a^- + b^-, a^+ + b^+], [A^- + B^-, A^+ + B^+], \gamma).$$

Если $\alpha \neq \beta$, $\alpha \neq 0$, $\beta \neq 0$, $I_{1l} \neq \emptyset$ и $I_{2l} \neq \emptyset$, то

$$D_1 + D_2 = \begin{cases} (\emptyset, [\min(A^- + b^+, B^- + a^+), \max(a^- + B^+, b^- + A^+)], 0), \\ \hfill \text{если } \gamma = 0; \\ ([a^- + B^+, B^- + a^+], [A^- + b^+, b^- + A^+], \alpha), & \text{если } \gamma = \alpha; \\ ([b^- + A^+, A^- + b^+], [B^- + a^+, a^- + B^+], \beta), & \text{если } \gamma = \beta. \end{cases}$$

Если $\alpha\beta = 0$, или $I_{1l} = \emptyset$, или $I_{2l} = \emptyset$, то

$$D_1 + D_2 = (T_1 + T_2, \gamma).$$

Умножение. Для некоторого твина $T = (I_l, I)$ введем обозначения

$$\begin{aligned} T^{(-)} &= [I^-, I_l^-]; \\ T^{(+)} &= [I_l^+, I^+]. \end{aligned}$$

Для направленных твинов $D_1 = (T_1, \alpha)$ и $D_2 = (T_2, \beta)$ определим

$$S_1 = \begin{cases} T_1^{(-\alpha)} \cdot T_2^{(-\beta)}, & \text{если } I_{1l} \neq \emptyset \ \& \ I_{2l} \neq \emptyset \ \& \ \alpha \neq 0 \ \& \ \beta \neq 0; \\ T_1^{(-\alpha)} \cdot I_2, & \text{если } I_{1l} \neq \emptyset \ \& \ \alpha \neq 0 \ \& \ (I_{2l} = \emptyset \vee \beta = 0); \\ I_1 \cdot T_2^{(-\beta)}, & \text{если } I_{2l} \neq \emptyset \ \& \ \beta \neq 0 \ \& \ (I_{1l} = \emptyset \vee \alpha = 0); \\ I_1 \cdot I_2, & \text{если } (I_{1l} = \emptyset \ \vee \alpha = 0) \ \& \ (I_{2l} = \emptyset \vee \beta = 0); \end{cases} \tag{2.24}$$

$$S_2 = \begin{cases} T_1^{(\alpha)} \cdot T_2^{(\beta)}, & \text{если } I_{1l} \neq \emptyset \ \& \ I_{2l} \neq \emptyset \ \& \ \alpha \neq 0 \ \& \ \beta \neq 0; \\ T_1^{(\alpha)} \cdot I_2, & \text{если } I_{1l} \neq \emptyset \ \& \ \alpha \neq 0 \ \& \ (I_{2l} = \emptyset \ \vee \ \beta = 0); \\ I_1 \cdot T_2^{(\beta)}, & \text{если } I_{2l} \neq \emptyset \ \& \ \beta \neq 0 \ \& \ (I_{1l} = \emptyset \ \vee \ \alpha = 0); \\ I_1 \cdot I_2, & \text{если } (I_{1l} = \emptyset \ \vee \alpha = 0) \ \& \ (I_{2l} = \emptyset \ \vee \beta = 0). \end{cases} \tag{2.25}$$

Произведение двух направленных твинов задается выражением

$$D_1 \cdot D_2 = \begin{cases} (\emptyset, S_1 \cup S_2, 0), & \text{если } S_1 \cap S_2 \neq Z; \\ ([S_1^+, S_2^-], [S_1^-, S_2^+], +), & \text{если } S_1^+ \leq S_2^-; \\ ([S_2^+, S_1^-], [S_2^-, S_1^+], -), & \text{если } S_1^- \geq S_2^+, \end{cases} \tag{2.26}$$

где Z – пустое или одноэлементное множество.

Вычитание и взятие обратного элемента.

$$-D_1 = ([-a^+, -a^-], [-A^+, -A^-], -\alpha);$$

$$1/D_1 = ([1/a^+, 1/a^-], [1/A^+, 1/A^-], -\alpha),$$

в последнем случае считаем, что $A^-A^+ > 0$. Корректность данных определений доказывает следующая теорема.

Теорема 2.8. Определение арифметики направленных твинов удовлетворяет свойствам (2.22) и (2.23).

Доказательство. (Краткий вариант доказательства см. в работе [131].) Для унарных операций теорема очевидна. Действительно, пусть

$$Y = \overline{G}(X) \sqsubseteq D = (T, \alpha).$$

Тогда

$$\begin{aligned} \overline{-G}(X) &= -Y \sqsubseteq -T; \\ \overline{1/G}(X) &= 1/Y \sqsubseteq 1/T. \end{aligned}$$

Если функция g слабо возрастает, то функции $-g$ и $1/g$ слабо убывают, и наоборот, если g слабо убывает, то $-g$ и $1/g$ слабо возрастают. Следовательно,

$$\begin{aligned} \overline{-G}(X) &\sqsubseteq (-T, -\alpha); \\ \overline{1/G}(X) &\sqsubseteq (1/T, -\alpha). \end{aligned}$$

Сложение. Пусть $f(x) = g(x) + h(x)$, причем

$$\begin{aligned} Y = \overline{G}(X) &\sqsubseteq D_1 = (T_1, \alpha) \qquad \text{и} \\ Z = \overline{H}(X) &\sqsubseteq D_2 = (T_2, \beta). \end{aligned}$$

Надо показать, что $\overline{G + H}(X) \sqsubseteq D_1 + D_2$.

1. Пусть $\alpha = \beta \neq 0$, $I_{1l} \neq \emptyset$ и $I_{2l} \neq \emptyset$. Ясно, что характер слабой монотонности $g + h$ совпадает с характером слабой монотонности g и h, т. е. $\gamma = \alpha$. В этом случае из того, что

$$\left(\overline{G}(X)\right)^- \in [A^-, a^-], \quad \left(\overline{H}(X)\right)^- \in [B^-, b^-],$$

следует, что

$$\left(\overline{G+H}(X)\right)^- \in [A^- + B^-, a^- + b^-].$$

Аналогично

$$\left(\overline{G+H}(X)\right)^+ \in [a^+ + b^+, A^+ + B^+].$$

Следовательно,

$$\overline{G+H}(X) \sqsubseteq ([a^- + b^-, a^+ + b^+], [A^- + B^-, A^+ + B^+]).$$

2. Если $\alpha \neq \beta$, $\alpha \neq 0$, $\beta \neq 0$, $I_{1l} \neq \emptyset$ и $I_{2l} \neq \emptyset$, то границы интервала $\overline{G+H}(X)$ заключены в интервалах

$$[A^-, a^-] + [b^+, B^+] = [A^- + b^+, a^- + B^+] \quad \text{и} \tag{2.27}$$

$$[B^-, b^-] + [a^+, A^+] = [B^- + a^+, b^- + A^+]. \tag{2.28}$$

Если $a^- + B^+ \leq B^- + a^+$, то $\gamma = \alpha$ и первый из двух указанных интервалов содержит левую границу $\overline{G+H}(X)$, а второй – правую границу. Если $b^- + A^+ \leq A^- + b^+$, то $\gamma = \beta$ и первый из двух указанных интервалов содержит правую границу $\overline{G+H}(X)$, а второй – левую. Если ни одно из неравенств не выполнено, то мы ничего не можем сказать о знаке γ, равно как не можем указать иного внутреннего интервала для твина $D_1 + D_2$, кроме пустого интервала. Внешний интервал твина $D_1 + D_2$ имеет (как видно из (2.27) и (2.28)) границы $\min(A^- + b^+, B^- + a^+)$ и $\max(a^- + B^+, b^- + A^+)$.

3. Если $\alpha \neq 0$ & $I_{1l} \neq 0$ и либо $\beta = 0$, либо $I_{2l} = \emptyset$, то левую границу интервала для суммы можно оценить интервалом

$$[A^-, a^-] + [B^-, B^+] = [A^- + B^-, a^- + B^+], \tag{2.29}$$

а правую – интервалом

$$[a^+, A^+] + [B^-, B^+] = [a^+ + B^-, A^+ + B^+]. \tag{2.30}$$

Если оказывается, что $a^- + B^+ \leq a^+ + B^-$, то $\gamma = \alpha$. Аналогично, если $\beta \neq 0$ & $I_{2l} \neq 0$ и либо $\alpha = 0$, либо $I_{1l} = \emptyset$, то A^i и a^i в выражениях (2.29) и (2.30) меняются на B^i и b^i и наоборот. Получаем, что $\gamma = \beta$ при условии $b^- + A^+ \leq b^+ + A^-$. В этих случаях нельзя предложить лучшей двусторонней оценки, чем та, которая уже доказана в арифметике обычных твинов.

4. В случае, если одновременно и $\alpha = 0 \ \vee \ I_{1l} = \emptyset$, и $\beta = 0 \ \vee \ I_{2l} = \emptyset$, естественно, имеем, что $\gamma = 0$. В этом случае используется оценка, уже полученная для простых твинов.

Рассмотренные случаи, как легко видеть, доказывают утверждение теоремы для сложения.

Умножение. Пусть мы оцениваем функцию $f(x) = g(x) \cdot h(x)$, причем, как и в случае сложения

$$Y = \overline{G(X)} \ \sqsubseteq \ D_1 = (T_1, \alpha) \qquad \text{и}$$
$$Z = \overline{H(X)} \ \sqsubseteq \ D_2 = (T_2, \beta).$$

Надо показать, что $\overline{G \cdot H}(X) \sqsubseteq D_1 \cdot D_2$.

Пусть $\alpha\beta \neq 0$. Учитывая слабую монотонность g и h, интервалы $T_1^{(-\alpha)}$, $T_1^{(\alpha)}$, $T_2^{(-\beta)}$ и $T_2^{(\beta)}$ содержат, соответственно, значения $g(X^-)$, $g(X^+)$, $h(X^-)$ и $h(X^+)$. Если $\alpha = 0$, то I_1 содержит $g(X^-)$ и $g(X^+)$. Если $\beta = 0$, то I_2 содержит $h(X^-)$ и $h(X^+)$. Получаем, что $f(X^-) \in$

S_1 и $f(X^+) \in S_2$. Если $S_1^+ \leq S_2^-$, то функция слабо возрастает, а если $S_1^- \geq S_2^+$, то слабо убывает. Если ни одно из этих неравенств не выполняется, то о характере монотонности функции ничего заключить нельзя. Формула (2.26) доказана. □

2.4.1.2 Свойства направленной твинной арифметики

Теорема 2.9. Сложение и умножение в арифметике направленных твинов являются коммутативными, но не ассоциативными операциями, т. е.

$$\begin{aligned} D_1 + D_2 &= D_2 + D_1; \\ D_1 \cdot D_2 &= D_2 \cdot D_1; \end{aligned}$$

$$\begin{aligned} \exists D_1, D_2: \quad (D_1 + D_2) + D_3 &\neq D_1 + (D_2 + D_3); \\ \exists D_1, D_2: \quad (D_1 \cdot D_2) \cdot D_3 &\neq D_1 \cdot (D_2 \cdot D_3). \end{aligned}$$

Доказательство. Коммутативность сложения и умножения доказать легко. Достаточно заметить, что формулы для сложения и умножения в определении 2.5 симметричны относительно слагаемых/сомножителей. Если поменять в них местами слагаемые/сомножители, то формулы окажутся тождественными самим себе.

Приведем пример невыполнения закона ассоциативности для сложения.

Пример 2.4. Пусть

$$\begin{aligned} D_1 &= ([1, 10], [0, 11], -); \\ D_2 &= ([1, 5], [0, 6], +); \\ D_3 &= ([1, 5], [0, 7], -). \end{aligned}$$

Тогда

$$\begin{aligned} D_1 + D_2 &= ([7,10],[5,12],-); \\ (D_1 + D_2) + D_3 &= ([8,15],[5,19],-); \\ D_2 + D_3 &= (\emptyset,[5,8],0) \\ D_1 + (D_2 + D_3) &= ([9,15],[5,19],0), \end{aligned}$$

то есть $(D_1 + D_2) + D_3 \neq D_1 + (D_2 + D_3)$.

В следующем примере не выполняется закон ассоциативности для умножения.

Пример 2.5. Пусть

$$\begin{aligned} D_1 &= ([2,10],[1,20],-); \\ D_2 &= ([2,3],[1,4],+); \\ D_3 &= ([10,11],[3,20],-). \end{aligned}$$

Тогда

$$\begin{aligned} D_1 \cdot D_2 &= ([8,10],[3,40],-); \\ (D_1 \cdot D_2) \cdot D_3 &= ([80,110],[9,800],-); \\ D_2 \cdot D_3 &= (\emptyset,[9,40],0) \\ D_1 \cdot (D_2 \cdot D_3) &= ([80,90],[9,800],-), \end{aligned}$$

то есть $(D_1 \cdot D_2) \cdot D_3 \neq D_1 \cdot (D_2 \cdot D_3)$.

Легко проверить по определению, что при любых арифметических действиях для любого a твин $([a,a],[a,a],+)$ ведет себя так же, как и твин $([a,a],[a,a],-)$. В дальнейшем такие направленные твины будут между собой отождествляться. Нетрудно проверить по определению, что нейтральным элементом относительно сложения является твин

$$([0,0],[0,0],+) = ([0,0],[0,0],-),$$

а нейтральным элементом относительно умножения – твин

$$([1,1],[1,1],+) = ([1,1],[1,1],-).$$

Для двух твинов $D_1 = (I_{1l}, I_1, \alpha)$ и $D_2 = (I_{2l}, I_2, \beta)$ отношение включения естественно определить как

$$D_1 \subseteq D_2 \;\Leftrightarrow\; (I_1 \subseteq I_2) \;\&\; (I_{2l} \subseteq I_{1l}) \;\&\; (\alpha = \beta \vee \beta = 0). \tag{2.31}$$

Теорема 2.10. Для введенных операций $\diamond \in \{-, 1/.\}$ и $\circ \in \{+, \cdot\}$ и направленных твинов D_i, S_i имеем

$$\begin{aligned} D_1 \subseteq E_1 \quad &\Rightarrow \quad \diamond D_1 \subseteq \diamond E_1; \\ (D_1 \subseteq E_1) \;\&\; (D_2 \subseteq E_2) \quad &\Rightarrow \quad D_1 \circ D_2 \subseteq E_1 \circ E_2; \end{aligned}$$

Предполагаем, что в случае взятия обратного значения внешний интервал твина E_1 не содержит нуля.

Доказательство. Если внутренние интервалы твинов непусты, то обозначим

$$\begin{aligned} D_1 = ([a_1^-, a_1^+], [A_1^-, A_1^+], \alpha_1) &= (I_{1l}, I_1, \alpha_1); \\ D_2 = ([a_2^-, a_2^+], [A_2^-, A_2^+], \alpha_1) &= (I_{2l}, I_2, \alpha_2); \\ E_1 = ([b_1^-, b_1^+], [B_1^-, B_1^+], \beta_1) &= (J_{1l}, J_1, \beta_1); \\ E_2 = ([b_2^-, b_2^+], [B_2^-, B_2^+], \beta_2) &= (J_{2l}, J_2, \beta_2). \end{aligned}$$

1. Вычитание. Принимая во внимание рассуждения относительно слабой монотонности функции $-f$ (см. доказательство Теоремы 2.8), получаем

$$\begin{aligned} D_1 \subseteq E_1 \quad &\Rightarrow \quad (J_{1l} \subseteq I_{1l}) \;\&\; (I_1 \subseteq J_1) \;\&\; (\alpha_1 = \beta_1 \vee \beta_1 = 0) \\ &\Rightarrow \quad (-J_{1l} \subseteq -I_{1l}) \;\&\; (-I_1 \subseteq -J_1) \;\&\; (-\alpha_1 = -\beta_1 \vee -\beta_1 = 0) \\ &\Rightarrow \quad -D_1 \subseteq -E_1. \end{aligned}$$

2. Взятие обратного элемента. Ввиду известного характера слабой монотонности функции $1/f$ получаем

$$
\begin{aligned}
D_1 \subseteq E_1 \quad & \Rightarrow \quad (J_{1l} \subseteq I_{1l}) \;\&\; (I_1 \subseteq J_1) \;\&\; (\alpha_1 = \beta_1 \vee \beta_1 = 0) \\
& \Rightarrow \quad (1/J_{1l} \subseteq 1/I_{1l}) \;\&\; (1/I_1 \subseteq 1/J_1) \;\&\; (-\alpha_1 = -\beta_1 \vee -\beta_1 = 0) \\
& \Rightarrow \quad 1/D_1 \subseteq 1/E_1.
\end{aligned}
$$

3. Сложение. Рассмотрим несколько случаев.

A. Пусть $\beta_1 \neq 0$, $\beta_2 \neq 0$, $J_{1l} \neq \emptyset$, $J_{2l} \neq \emptyset$. Имеем $J_{1l} \subseteq I_{1l} \neq \emptyset$, $J_{2l} \subseteq I_{2l} \neq \emptyset$.

A.1. Пусть $\beta_1 = \beta_2$. Тогда $\alpha_1 = \alpha_2 = \beta_1$. Имеем

$$
\begin{aligned}
[a_1^- + a_2^-, a_1^+ + a_2^+] \; & \supseteq \; [b_1^- + b_2^-, b_1^+ + b_2^+], \\
[A_1^- + A_2^-, A_1^+ + A_2^+] \; & \subseteq \; [B_1^- + B_2^-, B_1^+ + B_2^+],
\end{aligned}
$$

следовательно, $D_1 + D_2 \subseteq E_1 + E_2$.

A.2. Пусть $\beta_1 \neq \beta_2$. Тогда $\alpha_1 = \beta_1 = -\alpha_2 = -\beta_2$. Если

$$
b_1^+ + B_2^- \geq B_2^+ + b_1^-, \tag{2.32}
$$

то тем более $a_1^+ + A_2^- \geq A_2^+ + a_1^-$. Следовательно, в этом случае направление твина-суммы $D_1 + D_2$ совпадает с направлением твина-первого слагаемого D_1. Имеем

$$
\begin{aligned}
[a_1^- + A_2^+, A_2^- + a_1^+] \; & \supseteq \; [b_1^- + B_2^+, B_2^- + b_1^+], \\
[A_1^- + a_2^+, a_2^- + A_1^+] \; & \subseteq \; ,[B_1^- + b_2^+, b_2^- + B_1^+],
\end{aligned}
$$

следовательно, $D_1 + D_2 \subseteq E_1 + E_2$. Аналогично рассматривается случай

$$
b_2^+ + B_1^- \geq B_1^+ + b_2^-. \tag{2.33}
$$

Если оба неравенства (2.32) и (2.33) не выполняются, то направление твина $E_1 + E_2$ равно 0, а его внутренний интервал пуст. В этой ситуации для того, чтобы $D_1 + D_2 \subseteq E_1 + E_2$, достаточно того, чтобы внешний интервал $D_1 + D_2$ был подмножеством внешнего интервала $E_1 + E_2$. У твина $D_1 + D_2$ может быть любое направление, но внешний его интервал равен либо $[A_1^- + a_2^+, a_2^- + A_1^+]$, либо $[A_2^- + a_1^+, a_1^- + A_2^+]$, либо

$$[\min(A_1^- + a_2^+, A_2^- + a_1^+), \max(a_2^- + A_1^+, a_1^- + A_2^+)].$$

Во всех трех случаях он является подмножеством

$$[\min(B_1^- + b_2^+, B_2^- + b_1^+), \max(b_2^- + B_1^+, b_1^- + B_2^+)].$$

B. Пусть условия пункта A не выполняются, но $\beta_1 \neq 0$, $J_{1l} \neq \emptyset$. Тогда $\alpha_1 \neq 0$, $I_{1l} \neq \emptyset$. Если $b_1^+ + B_2^- \geq B_2^+ + b_1^-$, то тем более $a_1^+ + A_2^- \geq A_2^+ + a_1^-$. Этот случай рассматривается аналогично первому подслучаю пункта A.2 (см. формулу (2.32)).

C. Случай $\beta_2 \neq 0$, $J_{2l} \neq \emptyset$ получается из предыдущего заменой индекса 1 на 2 и наоборот.

D. Если $\beta_1 = 0 \ \vee \ J_{1l} = \emptyset$ и $\beta_2 = 0 \ \vee \ J_{2l} = \emptyset$, то направление твина $E_1 + E_2$ равно 0. Внешний интервал твина $E_1 + E_2$ равен $[B_1^- + B_2^-, B_1^+ + B_2^+]$ и содержит внешний интервал твина $D_1 + D_2$, каким бы ни было его направление. Внутренний интервал твина $E_1 + E_2$ равен внутреннему интервалу твина $(J_{1l}, J_1) + (J_{2l}, J_2)$, который содержится во внутреннем интервале твина $(I_{1l}, I_1) + (I_{2l}, I_2)$, а потому содержится и во внутреннем интервале направленного твина $D_1 + D_2$. В этом случае также получаем $D_1 + D_2 \subseteq E_1 + E_2$.

4. Умножение. Обозначим через S_{1D} и S_{2D} выражения для S_1 и S_2 (формулы (2.24) и (2.25)), примененные для перемножения D_1 и D_2, а через S_{1E} и S_{2E} – те же выражения, но для E_1 и E_2. Можно показать, что $S_{1D} \subseteq S_{1E}$ и $S_{2D} \subseteq S_{2E}$. Действительно, покажем, например, первое включение, второе показывается аналогично.

Первый сомножитель в определении S_1 (формула (2.24)) обозначим через S'_1, а второй через S''_1. Можно написать $S_{1D} = S'_{1D} \cdot S''_{1D}$ и $S_{1E} = S'_{1E} \cdot S''_{1E}$. Имеем

$$S'_{1D} = \begin{cases} D_1^{(-\alpha_1)}, & \text{если } I_{1l} \neq \emptyset \ \& \ \alpha_1 \neq 0; \\ I_1, & \text{если } I_{1l} = \emptyset \ \vee \ \alpha_1 = 0. \end{cases}$$

$$S'_{1E} = \begin{cases} E_1^{(-\beta_1)}, & \text{если } J_{1l} \neq \emptyset \ \& \ \beta_1 \neq 0; \\ J_1, & \text{если } J_{1l} = \emptyset \ \vee \ \beta_1 = 0. \end{cases}$$

Если $\beta_1 = 0$ или $J_{1l} = \emptyset$, то $S'_{1D} \subseteq S'_{1E}$ потому, что $D_1^{(-\alpha_1)} \subseteq J_1$ и $I_1 \subseteq J_1$. Если $\beta_1 \neq 0$ и $J_{1l} \neq \emptyset$, то $\alpha_1 = \beta_1$. В этом случае $D_1^{(-\alpha_1)} \subseteq E_1^{(-\beta_1)}$ и, следовательно, опять $S'_{1D} \subseteq S'_{1E}$. Аналогично получаем $S''_{1D} \subseteq S''_{1E}$. В результате имеем $S_{1D} \subseteq S_{1E}$.

Обратимся теперь к формуле (2.26). Если $S^+_{1E} \leq S^-_{2E}$, то тем более $S^+_{1D} \leq S^-_{2D}$. В этом случае $[S^+_{1D}, S^-_{2D}] \supseteq [S^+_{1E}, S^-_{2E}]$ и $[S^-_{1D}, S^+_{2D}] \subseteq [S^-_{1E}, S^+_{2E}]$, и, тем самым, $D_1 \cdot D_2 \subseteq E_1 \cdot E_2$. Аналогично рассматривается случай $S^-_{1E} \geq S^+_{2E}$. Наконец, если $S_{1E} \cap S_{2E} \neq Z$, то минимальный интервал, включающий в себя D_{1E} и D_{2E} (т. е. внешний интервал твина $D_1 \cdot D_2$) является подмножеством $S_{1E} \cup S_{2E}$. Учитывая, что внутренний интервал твина $E_1 \cdot E_2$ в этом случае пуст, а направление равно 0, имеем $D_1 \cdot D_2 \subseteq E_1 \cdot E_2$. □

Следствие. Пусть $F(D_1, \dots, D_n)$ – некоторое выражение, в котором D_i – переменные, обозначающие направленные твины. Выражение построено с применением операций сложения, умножения, унарного минуса и взятия обратного элемента. Тогда из $S_i \subseteq Q_i$, $i = 1, \dots, n$, следует

$$F(S_1, \dots, S_n) \subseteq F(Q_1, \dots, Q_n)$$

при любом наборе направленных твинов $Q_1, \dots, Q_n$, для которого твинные операции в выражении имеют смысл.

Следствие доказывается многократным применением Теоремы 2.10. Ее утверждение применяется столько раз, сколько функциональных символов входит в запись функции F.

2.4.1.3 Примеры

Рассмотрим два примера, иллюстрирующие применение арифметики твинов и направленной твинной арифметики.

Пример 2.6. Оценим множество значений функции

$$f(x,y) = (10e^y - x)(\operatorname{tg} y - x^2)$$

при $x \in [1.4, 1.5]$, $y \in [1.4, 1.5]$ средствами введенной твинной арифметики. В приведенных примерах все промежуточные результаты запоминаются с тремя цифрами после десятичной точки. Выполняются направленные округления (см. раздел 2.7). Границы интервала, представляющего внешнюю оценку, округляются во внешнюю сторону, а границы интервала, представляющего внутреннюю оценку – во внутреннюю.

$$\begin{aligned}
x = y &= ([1.4, 1.5], [1.4, 1.5]);\\
-x &= ([-1.5, -1.4], [-1.5, -1.4]);\\
e^y &= ([4.056, 4.481], [4.055, 4.482]);\\
10e^y &= ([40.56, 44.81], [40.55, 44.82]);\\
10e^y - x &= ([39.16, 43.31], [39.05, 43.42]);\\
\operatorname{tg} y &= ([5.798, 14.101], [5.797, 14.102]);\\
x^2 &= ([1.96, 2.25], [1.96, 2.25]);\\
-x^2 &= ([-2.25, -1.96], [-2.25, -1.96]);\\
\operatorname{tg} y - x^2 &= ([3.838, 11.851], [3.547, 12.142]);\\
(10e^y - x)(\operatorname{tg} y - x^2) &= ([166.646, 462.781], [138.51, 527.206]).
\end{aligned}$$

Пример 2.7. Оценим функцию

$$f(x) = (10e^x - x)(\operatorname{tg} x - x^2)$$

при $x \in [1.4, 1.5]$ средствами направленной твинной арифметики. Будем считать, что монотонность рассматриваемых функций уже установлена.

$$\begin{aligned}
x &= ([1.4, 1.5], [1.4, 1.5], +);\\
-x &= ([-1.5, -1.4], [-1.5, -1.4], -);\\
e^x &= ([4.056, 4.481], [4.055, 4.482], +);\\
10e^x &= ([40.56, 44.81], [40.55, 44.82], +);\\
10e^x - x &= ([39.16, 43.31], [39.15, 43.32], +);\\
\operatorname{tg} x &= ([5.798, 14.101], [5.797, 14.102], +);\\
x^2 &= ([1.96, 2.25], [1.96, 2.25], +);\\
-x^2 &= ([-2.25, -1.96], [-2.25, -1.96], -);\\
\operatorname{tg} x - x^2 &= ([3.838, 11.851], [3.837, 11.852], +);\\
(10e^y - x)(\operatorname{tg} y - x^2) &= ([150.297, 513.266], [150.218, 513.429], +).
\end{aligned}$$

Как нетрудно видеть, точность оценки в случае использования арифметики направленных твинов выше. В Примере 2.7 разница между внутренней и внешней оценками вызвана только направленными округлениями промежуточных результатов. Естественно, направленная твинная арифметика может быть применена только в случае, если выполняются условия монотонности.

2.4.2 Твинная арифметика с бесконечными твинами

В настоящем параграфе мы кратко коснемся еще одной возможности обобщения твинной арифметики: в качестве интервалов, входящих в состав твина, будем рассматривать бесконечные интервалы разного рода. Так как развитие теории бесконечных твинов не входит в наши планы, мы не будем доказывать какие-либо утверждения о бесконечных твинах и даже не будем строго формулировать определения соответствующих арифметик. Мы ограничимся лишь несколькими возможными определениями самих бесконечных твинов и кратким обсуждением.

Каковы же основные области применения бесконечных твинов? Если обыкновенные твины позволяют делать двусторонние оценки неизвестных интервалов, то бесконечные твины могут быть использованы для оценки бесконечных интервалов. Представляется, что бесконечные твины могут найти применение в теории интервальных интегралов, реализациях интервального варианта метода Ньютона, в ряде других областей.

Проще всего построить множество бесконечных твинов на базе множеств бесконечных интервалов $I(R^{\times})$ и $I(R^{+})$, введенных в разделах 1.6.1.1 и 1.6.1.2.

Бесконечными твинами первого рода будем называть пары (I_1, I_2), $I_1, I_2 \in I(R^{\times})$, $I_1 \subseteq I_2$, а бесконечными твинами второго рода будем называть пары (I_1, I_2), $I_1, I_2 \in I(R^{+})$, $I_1 \subseteq I_2$. И в том и в другом случае под отношением $\subseteq$ будем понимать обычное теоретико-множественное включение.

Бесконечные твины первого рода мало чем отличаются от обычных, конечных твинов. Единственное отличие – допустимы бесконечные значения для четырех концевых точек твина. Бесконечные твины второго рода позволяют работать с интервалами на проективной прямой. По существу $+\infty$ и $-\infty$ "склеиваются", факторизуются, и твин из двух вложенных один в другой интервалов прямой превращается в два интервала на проективной прямой. Можно также бесконечный твин второго рода рассматривать как твин на окружности, подобрав соответствующее отображение проективной прямой на окружность.

В арифметике бесконечных твинов второго рода операцию $1/T$, а значит, и деление, можно определить для любого твина T, в том числе и для такого, который содержит ноль внутри своего внешнего интервала. Таким образом, такая арифметика будет замкнута относительно четырех арифметических действий.

2.5 Сравнение с другими подходами

Идея твинов, вообще говоря, высказывалась в той или иной форме разными исследователями. По существу, как уже было сказано, она состоит в двусторонней – внутренней и внешней – оценке неизвестного интервала. Другой возможной переформулировкой идеи твинов является рассмотрение их как интервалов над интервалами. Мы остановимся на этом ниже в этом разделе. В работах [15] и [16] твины, хотя и называются по-другому, представляют из себя средство двусторонней оценки неопределенной величины. Упомянем работу [7], в которой твины выступают как элементы банахового пространства.

Безусловно, наиболее глубокое рассмотрение идеи твинов было предложено в серии работ Е. Гарденеса и А. Трепата [61], [62], [63], [64], [65]. К сожалению, эти работы не получили непосредственного дальнейшего развития и ряд имеющихся в них результатов позже был получен заново другими исследователями. Ниже мы кратко изложим подход Гарденеса и Трепата и проведем сравнение с нашим подходом.

Впервые авторы вскользь упоминают понятие твина в работах [61], [62]. (Одна из этих работ во многом является кратким изложением другой.) В этих работах твины упоминаются в контексте машинной реализации интервальной арифметики. В наиболее полном виде теория твинов изложена в работе [63]. Для Гарденеса и Трепата первоочередной задачей является построение достаточно красивой алгебраической структуры, а не та задача двусторонней оценки, о которой мы говорим в первую очередь.

На множестве интервалов Каухера $I^*(R)$ можно определить две пары операций. Первая пара обозначается inf и sup. Она индуцируется отношением $\leq$ на множестве интервалов. (Здесь считается, что $I_1 \leq I_2$, если левая граница первого интервала не больше левой границы втрого интервала, и правая граница первого интервала не больше правой границы второго.) Пусть $A = [A^-, A^+]$ и $B = [B^-, B^+]$. Тогда

$$\begin{aligned} \inf(A, B) &= [\inf(A^-, B^-), \inf(A^+, B^+)]; \\ \sup(A, B) &= [\sup(A^-, B^-), \sup(A^+, B^+)]. \end{aligned}$$

Вторая пара операций обозначается $\vee$ и $\wedge$. Она индуцируются отношением теоретико-множественного включения $\subseteq$.

$$\begin{aligned} A \wedge B &= [\sup(A^-, B^-), \inf(A^+, B^+)]; \\ A \vee B &= [\inf(A^-, B^-), \sup(A^+, B^+)]. \end{aligned}$$

При переходе от чисел к интервалам, интервал образуется как пара $[\inf \omega, \sup \omega]$, то есть минимальное множество соответствующего вида, содержащее все числа из ω. Аналогично, твин может быть определен исходя из интервала, то есть как интервал над интервалами. Твин является минимальным множеством некоторого вида, содержащим группу интервалов. В случае перехода от чисел к интервалам рассматривалось лишь одно отношение порядка – $\leq$. При переходе от интервалов к твинам таких отношений два. Соответственно, дается два определения твинов. Согласно первому – твин есть множество интервалов, заключенных между двумя интервалами I_1 и I_2 в следующем смысле

$$\{I \mid I_1 \le I \le I_2\},$$

а согласно второму – множество интервалов, определяющееся следующим образом

$$\{I \mid I_1 \subseteq I \subseteq I_2\}.$$

Полученные твинные пространства обозначаются, соответственно, $I(I^*(R), \le)$ и $I(I^*(R), \subseteq)$.

Доказывается теорема о том, что указанные твинные пространства эквивалентны. Неформально говоря, и в том и в другом пространстве твин можно задать парой интервалов или, что тоже самое, четверкой точек x_1, x_2, x_3, x_4. В первом пространстве твин задается парой интервалов $[x_2, x_3]$, $[x_1, x_4]$, а во втором пространстве – парой интервалов $[x_1, x_3]$, $[x_2, x_4]$. Между двумя пространствами может быть установлено однозначное соответствие. По существу мы имеем дело с одним и тем же пространством, но представленным разными способами. Обозначим это пространство $I(I^*(R))$.

Авторы работы [63] идут дальше. Они определяют множество $I^*(I^*(R))$. Элементами этого множества являются твины, представленные одной из двух альтернативных форм

$$\begin{aligned} A &= ([a_1, a_2], [a_3, a_4])_{\le}; \\ B &= ([b_1, b_2], [b_3, b_4])_{\subseteq}. \end{aligned}$$

В отличие от множества $I(I^*(R))$, условия $[a_1, a_2] \le [a_3, a_4]$ и $[b_1, b_2] \subseteq [b_3, b_4]$ не обязаны быть выполненными. Пространство твинов $I^*(I^*(R))$, в отличие от введенного нами (см. раздел 2.8), является решеткой и проявляет ряд других "хороших" алгебраических свойств.

Отношения $\le$ и $\subseteq$ на множестве интервалов естественным образом переносятся на множество $I^*(I^*(R))$. Перенос производится поэлементным применением соответствующих отношений к интервалам, участвующим, соответственно, в $\leq$-представлении или $\subseteq$-представлении твина. Помимо этого, на множестве $I^*(I^*(R))$ можно ввести отношение $\subseteq_\tau$ следующим образом

$$A \subseteq_\tau B \iff (\forall X \in I^*(R)\ X \in A \Rightarrow X \in B).$$

Три решетки образуют весьма прозрачную структуру, отраженную на следующей схеме. Схема заимствована нами из работы [63].

Твинная решетка	$(I^*(I^*(R)),$	$\leq,$	$\subseteq,$	$\subseteq_\tau)$
Интервальная решетка	$(I^*(R),$	$\leq,$	$\subseteq)$	
Решетка действительных чисел	$(R,$	$\leq)$		

$\longrightarrow$ Теоретико-множественное обобщение

$\longrightarrow$ Интервальное обобщение

Рис. 2.5.

Тот факт, что твины при рассматриваемом подходе вводятся как интервалы над интервалами, диктует поэлементное определение твинной арифметики. Пусть $A, B, C, D \in I^*(R)$; $(A, B), (C, D) \in I(I^*(R))$; $\circ \in \{+, \cdot\}$. Тогда

$$(A, B) \circ (C, D) = \{X \circ Y \mid X \in (A, B), Y \in (C, D)\} = (A \circ C, B \circ D).$$

Определенные таким образом операции умножения и сложения, во-первых, естественно распространяются на множество $I^*(I^*(R))$, а во-вторых, порождают соответствующие им определения деления и вычитания.

Такое определение арифметических операций не может быть применено для двусторонней оценки множества значений функции. Проиллюстрируем это примером.

Пример 2.8. Пусть $f(x) = 4x + (-x)$ и $x \in X \sqsubseteq ([1,2],[0,3])$. Оценки для функций $4x$ и $-x$ в нашей арифметике и в твинной арифметике из работы [63] совпадают.

$$4x \in Y \sqsubseteq ([4,8],[0,12]) = T_1;$$

$$-x \in Z \sqsubseteq ([-2,-1],[-3,0]) = T_2.$$

Бинарные же операции задаются по-разному. Обозначим твинное сложение, определенное в разделе 2.2.3, как и ранее, знаком $+$, а сложение по Гарденесу и Трепату – знаком $\oplus$.

$$T_1 + T_2 = ([4,5],[-3,12]); \tag{2.34}$$

$$T_1 \oplus T_2 = ([2,7],[-3,12]). \tag{2.35}$$

Однако $F(X) \sqsubseteq ([3,6],[0,9])$, и оценка (2.35), в отличие от (2.34), неверна.

2.6 Твины и глобальная оптимизация

Рассмотрим некоторые вопросы, связывающие теорию твинных арифметик и методы глобальной оптимизации. Последних мы уже кратко коснулись в разделе 1.4, там же можно найти ссылки на литературу.

У твинных методов двусторонней оценки множества значений функции и у методов глобальной оптимизации общая задача: найти гарантированные двусторонние (внешние и внутренние) оценки интервала значений функции. В методах глобальной оптимизации эта задача формулируется следующим образом: найти гарантированную оценку глобальных экстремумов функции. Если для глобального минимума и глобального максимума функции найдены гарантированные включающие интервалы, скажем $[a, b]$ и $[c, d]$, то тем самым найден и оценивающий твин $([b, c], [a, d])$. Для методов глобальной оптимизации характерны следующие черты:

- поиск минимума и максимума производится не одновременно;
- производится поиск всех глобальных минимумов (максимумов), если их несколько;
- необходимо найти минимальные и максимальные значения функции с заранее заданной точностью;
- необходимо найти не только экстремальные значения функции, но и значения аргументов, в которых они достигаются, причем также с заранее заданной точностью; необходимо также найти все такие аргументы, если их несколько;
- зачастую задача поиска глобальных экстремумов решается в присутствии дополнительных ограничений, имеющих, например, вид $g_i(x_1, \ldots, x_n) = 0$, где x_i – аргументы оцениваемой функции.

У твинных методов оценок и у методов глобальной оптимизации различные области приложения. Методы глобальной оптимизации применяются тогда, когда необходимо найти высокоточные оценки экстремумов функции и одновременно удовлетворить ряд других требований, упомянутых выше. При этом сами алгоритмы глобальной оптимизации весьма сложны как с точки зрения их реализации, так и с точки зрения потребляемых вычислительных ресурсов. Основная идея алгоритма поиска глобального экстремума весьма прозрачна, она кратко изложена выше в разделе 1.4. Вместе с тем за 20 лет развития данного направления методы глобальной оптимизации обросли очень большим количеством деталей и усовершенствований, позволяющих эффективно решать поставленную задачу с высокой и, самое главное, гарантированной точностью.

Твинные методы двусторонней оценки сами по себе не могут обеспечить высокой точности локализации экстремумов функции. Их задача в другом – найти двустороннюю оценку множества значений, сделав это без больших затрат вычислительных ресурсов и без индивидуального анализа данной конкретной функции, лишь автоматически применяя твинные алгоритмы. Основная область применения твинных алгоритмов видится в их использовании в составе других известных вычислительных алгоритмов различного назначения. При этом часто успех может быть достигнут заменой интервального оценивания на твинное оценивание в составе известных вычислительных методов. Точность последних при этом возрастает.

Примером одного алгоритма, в котором применение твинного оценивания сулит выгоды, является сам алгоритм поиска глобальных экстремумов. Схема алгоритма, описанная в разделе 1.4, включает в себя три основные части:

- дисциплину дробления основного интервала на части и переупорядочивания интервалов в поддерживаемых списках, а также алгоритм проверки окончания выполнения алгоритма ввиду достижения цели, или ввиду того, что можно сделать вывод о принципиальной ее недостижимости.

- алгоритм оценки функции на любом подынтервале рассматриваемого интервала;
- многочисленные усовершенствования, помогающие сократить потребляемые вычислительные ресурсы путем исключения из рассмотрения интервалов, заведомо не содержащих глобальных экстремумов.

В разнообразных алгоритмах глобальной оптимизации первая и третья части могут существенно изменяться, исходя из области применения конкретного алгоритма и присутствующих дополнительных условий. Вторая часть как правило всегда одна и та же – интервальное вычисление естественного интервального расширения рассматриваемой функции на конкретном подынтервале рассматриваемого интервала. Вторая часть выполняется многократно для разных подынтервалов исходного интервала.

Для определенности предположим, что мы вычисляем глобальный минимум. Обозначим оцениваемую функцию через $f(x)$, ее естественное интервальное расширение через $F(X)$, а твинное расширение через $F_t(T)$. В ходе выполнения второй части алгоритма при классическом подходе вычисляется $F(X)$ для конкретного интервала X. Пусть полученное значение равно $[A, B]$. Далее используется лишь значение B в качестве верхней оценки минимума функции f на X. Чем B ближе к левой границе интервала $\overline{F}(X)$, т. е. чем точнее оценка, тем более эффективно применение всего алгоритма глобальной оптимизации. Полученное значение B используется в других частях алгоритма глобальной оптимизации для переупорядочивания подынтервалов и для получения текущего приближения значения глобального минимума. Если вычисление $F(X)$ заменить на вычисление $F_t(X, X) = ([c, d], [A, B])$, то в качестве верхней оценки минимума функции f на X следует взять число c, которое точнее его приближает, чем B. Оценка повышения эффективности всего алгоритма при такой замене зависит от конкретного вида алгоритма глобальной оптимизации, а также от вида функции f.

2.7 Машинная твинная арифметика

Как и в случае интервальной арифметики, при реализации арифметики твинов на компьютере следует принимать во внимание необходимость округления входных данных, промежуточных и итоговых результатов. Будем считать, что функции $\mathrm{fl}_\downarrow$ и $\mathrm{fl}_\uparrow$, округляющие действительные числа, соответственно, вниз и вверх, нам известны (см. раздел 1.6.3). Обозначим множество твинов через $T(R)$, множество направленных твинов через $T^*(R)$. Множество "машинных твинов", т. е. твинов, границы которых точно представимы на некотором конкретном компьютере, обозначим через $T(R_M)$. Множество направленных машинных твинов обозначим через $T^*(R_M)$.

Введем на множестве интервалов "округление внутрь" следующим образом:

$$\mathrm{fl}_\ddagger([a,b]) = [\mathrm{fl}_\uparrow(a), \mathrm{fl}_\downarrow(b)].$$

Если при округлении между a и b нет ни одного машинного числа, т. е. $\mathrm{fl}_\uparrow(a) > \mathrm{fl}_\downarrow(b)$, то будем считать, что $\mathrm{fl}_\ddagger([a,b]) = \emptyset$.

Так же будем формально полагать, что $\mathrm{fl}_\ddagger(\emptyset) = \emptyset$. На множестве твинов введем функцию округления в соответствии со следующей формулой:

$$\mathrm{fl}_\updownarrow((A,B)) = (\mathrm{fl}_\ddagger(A), \mathrm{fl}_\updownarrow(B)).$$

При таком определении выполняется соотношение $\mathrm{fl}_\updownarrow(T) \supseteq T$, что позволяет использовать машинные твины для двусторонних гарантированных оценок в смысле отношения $\subseteq$ на множестве твинов, введенного в разделе 2.3. Внутренний интервал, обеспечивающий внутреннюю оценку, округляется в сторону сужения, внешний – в сторону расширения.

Аналогично, для направленных твинов функция округления вводится следующим образом:

$$\mathrm{fl}_{\updownarrow}((A, B, \alpha)) = (\mathrm{fl}_{\ddagger}(A), \mathrm{fl}_{\updownarrow}(B), \alpha).$$

Одноместные операции $\diamond$ над машинными твинами задаются в соответствии с формулой

$$\diamond T = \mathrm{fl}_{\updownarrow}(\diamond T),$$

а двуместные $\circ$ – с формулой

$$T_1 \circ T_2 = \mathrm{fl}_{\updownarrow}(T_1 \circ T_2).$$

Если твинные операции задать указанным образом, то арифметика машинных твинов будет монотонна по включению. Пусть $T_i, S_i \in T(R_M)$, $T_1 \subseteq S_1$, $T_2 \subseteq S_2$. Если $T_3 = \mathrm{fl}_{\updownarrow}(\diamond T_1)$, $S_3 = \mathrm{fl}_{\updownarrow}(\diamond S_1)$, $T_4 = \mathrm{fl}_{\updownarrow}(T_1 \circ T_2)$, $S_4 = \mathrm{fl}_{\updownarrow}(S_1 \circ S_2)$, то $T_3 \subseteq S_3$, $T_4 \subseteq S_4$. Выполнение указанного свойства становится очевидным, если вспомнить, что округление вещественных чисел отвечает свойству (1.11) и, следовательно, округление твинов отвечает свойству $T_1 \subseteq T_2 \Rightarrow \mathrm{fl}_{\updownarrow}(T_1) \subseteq \mathrm{fl}_{\updownarrow}(T_2)$.

Пусть $T_1, T_2 \in T(R_M)$, $I_1, I_2 \in I(R_M)$, $I_1 \sqsubseteq T_1$, $I_2 \sqsubseteq T_2$. Тогда из-за основных свойств твинной арифметики (формулы (2.1), (2.2)) и из-за того, что $\mathrm{fl}_{\updownarrow}(T) \supseteq T$ имеем

$$\begin{aligned} \diamond I_1 &\sqsubseteq \mathrm{fl}_{\updownarrow}(\diamond T_1); \\ I_1 \circ I_2 &\sqsubseteq \mathrm{fl}_{\updownarrow}(T_1 \circ T_2). \end{aligned}$$

Из последних утверждений следует вывод о корректности округлений, выполняемых по ходу выполнения твинных оценивающих алгоритмов. Аналогичные результаты верны и для направленных твинов.

2.8 Алгебраические свойства твинных пространств

В разделах 2.3 и 2.4.1.2 мы уже рассматривали некоторые свойства множества обычных твинов и множества направленных твинов. В настоящем параграфе попытаемся систематизировать сказанное. Также мы проведем сравнение свойств твинов со свойствами интервалов, а свойств направленных твинов – со свойствами направленных интервалов.

Как уже отмечалось в разделе 1.7, множество интервалов $I(R)$ образует коммутативную полугруппу относительно операции сложения и коммутативную полугруппу относительно операции умножения. В первой из двух полугрупп выполняется закон сокращения, а для его выполнения в мультипликативной полугруппе из множества $I(R)$ необходимо исключить интервалы, содержащие 0. Множество $I(R)$ не является решеткой относительно порядка $\subseteq$.

Арифметика Каухера $I^*(R)$ является более совершенной алгебраической структурой. Она образует коммутативную группу относительно операции сложения. Относительно операции умножения множество интервалов Каухера не является группой: обратный элемент существует лишь для интервалов, обе границы которых имеют один и тот же знак (см., например, [157]). Множество интервалов Каухера образует решетку относительно порядка $\subseteq$ (см. [3], [157]).

Множество интервалов естественным образом отождествляется с подмножеством множества твинов, а множество интервалов Каухера отождествляется с подмножеством множества направленных твинов, подобно тому как множество действительных чисел отождествляется с подмножеством множества интервалов. Уже отмечалось, что интервал I является вырожденным случаем твина. Такой интервал может быть представлен твином (I, I). Аналогично, направленный интервал (I, α) может быть представлен вырожденным направленным твином (I, I, α).

Если проанализировать определения арифметических операций над твинами и направленными твинами, то легко заметить, что действия над их внешними интервалами не зависят от внутренних интервалов. Указанное обстоятельство приводит к тому, что от твинных пространств не приходится ожидать выполнения более совершенных алгебраических свойств, чем от соответствующих интервальных пространств.

Докажем несколько простых теорем о свойствах рассматриваемых твинных пространств (см. также [133]).

Теорема 2.11. Множество простых твинов является коммутативной полугруппой по операции сложения. В полугруппе закон сокращения не выполняется.

Доказательство. Ассоциативность и коммутативность были показана в разделе 2.3. Закон сокращения не выполняется, что показывает следующий пример.

Пример 2.9. Пусть $T_1 = T_2 = ([2,5],[1,6])$; $T_3 = ([3,4],[1,6])$. Имеем

$$T_1 + T_2 = T_1 + T_3 = (\emptyset, [2, 12]),$$

но $T_2 \neq T_3$.

$\square$

Заметим, что не у каждого твина есть обратный к нему относительно операции сложения. Это ясно, если отметить, что при сложении твинов внешние интервалы подчиняются правилам сложения обычных интервалов, а для обычных интервалов, как хорошо известно [1], [9], в большинстве случаев не существует интервалов, обратных по операции сложения.

Теорема 2.12. Множество простых твинов является коммутативной полугруппой по операции умножения. Закон сокращения для простых твинов не выполняется.

Доказательство. Ассоциативность и коммутативность были показана в разделе 2.3. То, что закон сокращения не выполняется, показывает следующий пример.

Пример 2.10. Пусть, как и в предыдущем примере $T_1 = T_2 = ([2,5],[1,6])$; $T_3 = ([3,4],[1,6])$. Имеем

$$T_1 \cdot T_2 = T_1 \cdot T_3 = (\emptyset, [1,36]),$$

но $T_2 \neq T_3$.

$\square$

Как и в случае сложения, относительно операции умножения не у каждого твина есть обратный элемент. При умножении твинов внешние интервалы подчиняются правилам умножения обычных интервалов, а для обычных интервалов в большинстве случаев не существует интервалов, обратных по операции умножения.

На множестве твинов в качестве отношения порядка рассмотрим отношение включения, заданное формулой (2.12), а на множестве направленных твинов – отношение включения, заданное формулой (2.31). Легко проверяется, что указанные определения задают отношения, удовлетворяющие аксиомам порядка. Действительно, для произвольных простых твинов $T_i = (I_i, J_i)$, $i = 1, 2, 3$ имеем:

- $T_1 \subseteq T_1$, поскольку свойство рефлексивности выполняется для простых интервалов;

- из $T_1 \subseteq T_2$ и $T_2 \subseteq T_3$ следует, что $I_1 \supseteq I_2$, $I_2 \supseteq I_3$, $J_1 \subseteq J_2$ и $J_2 \subseteq J_3$, откуда по транзитивности отношения включения для интервалов получаем $I_1 \supseteq I_3$, $J_1 \subseteq J_3$, следовательно, $T_1 \subseteq T_3$;
- из $T_1 \subseteq T_2$ и $T_2 \subseteq T_1$ следует, что $I_1 \supseteq I_2$, $I_2 \supseteq I_1$, $J_1 \subseteq J_2$ и $J_2 \subseteq J_1$, откуда по антисимметричности отношения включения для интервалов получаем $I_1 = I_2$, $J_1 = J_2$, следовательно, $T_1 = T_2$.

Для произвольных направленных твинов $T_i = (I_i, J_i, \alpha_i)$, $i = 1, 2, 3$ имеем:

- $T_1 \subseteq T_1$, из-за выполнения рефлексивности для интервалов;
- из $T_1 \subseteq T_2$ и $T_2 \subseteq T_3$ следует, что $I_1 \supseteq I_2$, $I_2 \supseteq I_3$, $J_1 \subseteq J_2$ $J_2 \subseteq J_3$, $\alpha_1 = \alpha_2 \vee \alpha_2 = 0$ и $\alpha_2 = \alpha_3 \vee \alpha_3 = 0$, откуда по транзитивности отношения включения для интервалов получаем $I_1 \supseteq I_3$, $J_1 \subseteq J_3$, кроме того, $\alpha_1 = \alpha_3 \vee \alpha_3 = 0$, следовательно, $T_1 \subseteq T_3$;
- из $T_1 \subseteq T_2$ и $T_2 \subseteq T_1$ следует, что $I_1 \supseteq I_2$, $I_2 \supseteq I_1$, $J_1 \subseteq J_2$ и $J_2 \subseteq J_1$, $\alpha_1 = \alpha_2 \vee \alpha_2 = 0$, $\alpha_2 = \alpha_1 \vee \alpha_1 = 0$, откуда по антисимметричности отношения включения для интервалов получаем $I_1 = I_2$, $J_1 = J_2$, кроме того, $\alpha_1 = \alpha_2$, следовательно, $T_1 = T_2$.

Теорема 2.13. Множества простых и направленных твинов не являются решетками.

Доказательство очевидно. Например, в случае простых твинов, согласно определению решетки, каждое множество из двух твинов T_1 и T_2 должно иметь точную нижнюю грань, т. е. должен существовать такой твин T_3, что $T_3 \subseteq T_1$ и $T_3 \subseteq T_2$. Если внешние интервалы твинов T_1 и T_2 не пересекаются, то внешний интервал твина T_3 обязан быть пустым, что недопустимо по определению твинов. Для случая направленных твинов доказательство полностью аналогично. □

Теорема 2.14. Множество направленных твинов не является полугруппой по операциям сложения и умножения. Свойства коммутативности относительно обеих операций выполняются. Не у каждого направленного твина есть обратные элементы относительно сложения и умножения.

Доказательство. Множество направленных твинов не является полугруппой из-за невыполнения свойства ассоциативности. Коммутативность и отсутствие ассоциативности по обеим арифметическим операциям показаны в разделе 2.4.1.2. Необходимо показать отсутствие обратных элементов.

Зафиксируем $D_1 = (I_l, I, 0)$, $|I| > 0$. Какой бы $D_2 = (J_l, J, \beta)$ мы ни взяли, внешний интервал $D_1 + D_2$ будет равен $I + J$, длина которого отлична от нуля. Интервал $D_1 + D_2$ не может быть равен твину $([0,0],[0,0],\alpha)$.

Далее, для такого же D_1 рассмотрим произведение $D_1 \cdot D_2$. Обратимся к формулам (2.24), (2.25), (2.26). Видим, что $|S_1| \geq 0$, $|S_2| \geq 0$, следовательно, длина внешнего интервала твина $D_1 \cdot D_2$ отлична от нуля. Интервал $D_1 \cdot D_2$ не может быть равен твину $([1,1],[1,1],\alpha)$. $\square$

Чем же объясняется отсутствие у множества твинов алгебраических свойств, присущих множеству интервалов? Твины были введены для двусторонней оценки неизвестных интервалов. Твин есть совокупность двух интервалов, один из которых оценивает неизвестный интервал снаружи, другой изнутри. Интервалы, оценивающие снаружи, при операциях с твинами проявляют свойства, отличные от тех, которые проявляются интервалами, оценивающими изнутри. Образовав твины, пары интервалов не проявляют "хороших" алгебраических свойств.

Во второй главе мы построили основу теории твинных вычислений. Были введены и исследованы арифметика твинов и арифметика направленных твинов, использование которых дает возможность решить задачу двустороней интервальной оценки множества значений функции. В главе решена задача создания аппарата для автоматической локализации интервальных значений. Вторая глава создает необходимые предпосылки для построения твинных алгоритмов, для изучения свойств твинных вычислений и для разработки принципов компьютерной реализации. Все эти вопросы будут рассмотрены в последующих главах.

Глава 3. Точность и сложность твинных вычислений

3.1 Семантика твинных вычислений

Расмотрим некоторые вопросы, касающиеся содержательного аспекта интервальных и твинных вычислений. Пусть задана непрерывная вещественная функция $f(x_1, \ldots, x_n)$ и соответствующие ей естественное интервальное и естественное твинное расширения $F(X_1, \ldots, X_n)$ и $F(T_1, \ldots, T_n)$. Функция f задана в виде символьного выражения. Будем считать, что в записи функции f присутствуют лишь те функциональные символы, для которых в разделе 2.2.3 была определена твинная арифметика (т. е. бинарные операции $+$, $\cdot$ и унарные операции $-$ и $1/\cdot$). Это ограничение не является принципиальным – твинные расширения можно определить и для других элементарных функций.

Пусть кроме функции f заданы интервалы I_i, $i = 1, \ldots, n$, и для переменных x_i известно, что их значения принадлежат соответствующим интервалам I_i, $x_i \in I_i$. Как и ранее, действительное множество значений функции F при указанном ограничении на значения переменных будем обозначать $\overline{F}(I_1, \ldots, I_n)$.

Смысл интервального вычисления функции $F(X_1, \ldots, X_n)$ состоит в том, что полученный интервал является внешней оценкой множества $\overline{F}(X_1, \ldots, X_n)$. В этом, как уже отмечалось, заключена первоначальная идея интервальных вычислений. В большинстве случаев полученная оценка является завышенной. Она не учитывает взаимозависимость подвыражений выражения, задающего функцию f. Если каждая переменная x_i входит в выражение, задающее функцию f, однократно, то можно гарантировать, что полученная оценка точна (см. Теорему 1.2).

Смысл твинного вычисления функции $F(T_1, \dots, T_n)$ для значений твинов $T_i = (X_i, X_i)$ состоит в нахождении, во-первых, той же самой внешней оценки, которая является результатом интервальных вычислений, и, во-вторых, внутренней оценки, то есть такого интервала, который гарантированно содержится в интервале $\overline{F}(X_1, \dots, X_n)$. Естественно, как и в случае нахождения внешней оценки, внутренняя оценка в большинстве случаев находится неточно.

Целью настоящего раздела является получение результатов о величине такой неточности, а также разработка методов и алгоритмов ее снижения. Заметим, что в случае внешней оценки мы находимся в более благоприятных обстоятельствах: рост неточности оценки не приводит к ее вырождению. Скажем, внешняя оценка интервала $[0, 1]$ интервалом $[-1000, 1000]$ имеет некоторый смысл, хотя, как правило, и не является допустимой с практической точки зрения. Рост неточности внутренней оценки приводит к ее вырождению. При возрастании неточности внутренней оценки соответствующий интервал "схлопывается". Внутренняя оценка теряет смысл. Эта ситуация нашла свое отражение в формулах твинной арифметики: в ряде случаев в качестве внутреннего интервала твина-результата фигурирует пустое множество.

В отличие от получения внешней оценки, получение внутренней оценки средствами твинной арифметики в случае, если каждая переменная встречается в выражении ровно один раз, не является точным. Введенная твинная арифметика априори учитывает возможную взаимозависимость между аргументами функций многих переменных. В случае, если мы убеждены в том, что аргументы соответствующей функции не зависят друг от друга, мы можем применить другую арифметику. Докажем следующую лемму.

Лемма 3.1. Пусть имеется действительная функция $f(x, y)$, для которой известно интервальное расширение $F(X, Y)$. Известны внешние и внутренние оценки интервалов изменения x и y, то есть известны твины T_1 и T_2 такие, что

$$x \in X \sqsubseteq T_1 = (I_{1l}, I_1);$$
$$y \in Y \sqsubseteq T_2 = (I_{2l}, I_2).$$

Тогда для любого множества пар (x, y) (назовем его Ω), для которого $\Omega \supseteq I_{1l} \times I_{2l}$ имеем

$$f(\Omega) = \{f(x, y) \mid (x, y) \in \Omega\} \supseteq F(I_{1l}, I_{2l}).$$

Доказательство. Возьмем произвольное z, $z \in F(I_{1l}, I_{2l})$. Существуют $x' \in I_{1l}$, $y' \in I_{2l}$ такие, что $f(x', y') = z$. Имеем $(x', y') \in \Omega$, следовательно, $z = f(x', y') \in f(\Omega)$. □

Теорема 3.1. Пусть задана функция $f(x_1, \ldots, x_n)$, причем каждая переменная x_i встречается в записи функции ровно один раз. $F(X_1, \ldots, X_n)$ – естественное интервальное расширение f. Пусть, кроме того, для всех i $X_i \sqsubseteq (I_{il}, I_i)$. Тогда

$$F(I_{1l}, \ldots, I_{nl}) \subseteq \overline{F}(X_1, \ldots, X_n) \subseteq F(I_1, \ldots, I_n). \tag{3.1}$$

Доказательство. В формуле (3.1) необходимо доказать лишь левое включение. Правое следует из свойств интервальной арифметики. Доказательство проводится индукцией по k – количеству функциональных символов, входящих в запись функции f.

Пусть $k = 1$. Если единственный функциональный символ представляет функцию одной переменной, то $F(X) = \overline{F}(X)$. Если единственная функция является функцией от двух переменных, то возьмем в качестве Ω множество $I_{1l} \times I_{2l}$. По Лемме 3.1 непосредственно получим доказательство базы индукции.

Пусть мы доказали теорему для $k \leq m-1$. Докажем для $k = m$. Представим $f(x_1, \ldots, x_n)$ в виде $g(h(x_1, \ldots, x_n))$ или в виде $g(h_1(x_{j_1}, \ldots, x_{j_p}), h_2(x_{s_1}, \ldots, x_{s_q}))$ в зависимости от того, сколько аргументов у базовой функции, находящейся на "самом верхнем уровне" записи выражения для f. Здесь мы не рассматриваем случай базовых функций более чем двух аргументов, хотя это и не принципиально. Если функция g имеет один аргумент, то по индуктивному предположению

$$H(I_{1l}, \ldots, I_{nl}) \subseteq \overline{H}(X_1, \ldots, X_n).$$

Следовательно,

$$F(I_{1l}, \ldots, I_{nl}) = G(H(I_{1l}, \ldots, I_{nl})) \subseteq G(\overline{H}(X_1, \ldots, X_n)) = \overline{F}(X_1 \ldots, X_n).$$

Пусть теперь функция g имеет два аргумента. Заметим, что множества аргументов функций h_1 и h_2 не пересекаются. По индуктивному предположению имеем

$$\begin{aligned} H_1(I_{j_1 l}, \ldots, I_{j_p l}) &\subseteq \overline{H}_1(X_{j_1}, \ldots, X_{j_p}); \\ H_2(I_{s_1 l}, \ldots, I_{s_q l}) &\subseteq \overline{H}_2(X_{s_1}, \ldots, X_{s_q}). \end{aligned}$$

Так как множества $\{X_{j_1}, \ldots, X_{j_p}\}$ и $\{X_{s_1}, \ldots, X_{s_q}\}$ не пересекаются, то имеем

$$\overline{F}(X_1, \ldots, X_n) = G(\overline{H}_1(X_{j_1}, \ldots, X_{j_p}), \overline{H}_2(X_{s_1}, \ldots, X_{s_q})),$$

кроме того, по определению

$$F(I_{1l}, \ldots, I_{nl}) = G(H_1(X_{j_1}, \ldots, X_{j_p}), H_2(X_{s_1}, \ldots, X_{s_q})),$$

следовательно,

$$F(I_{1l}, \ldots, I_{nl}) \subseteq \overline{F}(X_1, \ldots, X_n).$$

□

Пример 3.1. Пусть

$$f(x, y, z) = x - y \cdot z,$$
$$g(x, y, z) = x - y \cdot x,$$
$$T_x = T_y = T_z = ([0, 1], [0, 1]).$$

Функция f удовлетворяет условию Теоремы 3.1, а функция g – не удовлетворяет. Пользуясь теоремой, попытаемся оценить изнутри $\overline{F}(X, Y, Z)$ при $X \sqsubseteq T_x$, $Y \sqsubseteq T_y$, $Z \sqsubseteq T_z$. Имеем:

$$\overline{F}(X, Y, Z) \supseteq [0, 1] - [0, 1] \cdot [0, 1] = [-1, 1].$$

Оценка верна, так как $\overline{F}(X, Y, Z) = [-1, 1]$. Теорема не позволяет применить тот же аппарат для оценки интервала $\overline{G}(X, Y, Z)$, т. к. переменная x имеет два вхождения в запись функции. Действительно, если бы мы попытались оценить $\overline{G}$ путем простых интервальных операций над внутренними оценками аргументов (т. е. внутренними интервалами твинов T_x, T_y и T_z), то получили бы неверную оценку $\overline{G}(X, Y, Z) \supseteq [-1, 1]$. В действительности $\overline{G}(X, Y, Z) = [0, 1]$.

Лемма 3.2. Для функции $f(x_1, \ldots, x_n)$ с однократным вхождением всех переменных способ получения внутренней оценки, предлагаемый Теоремой 3.1, дает точную оценку в том смысле, что существуют такие $X_i' \sqsubseteq (I_{il}, I_i)$, $i = 1, \ldots, n$, что

$$F(I_{1l}, \ldots, I_{nl}) = \overline{F}(X_1', \ldots, X_n').$$

Доказательство. В качестве таких X_i' возьмем I_{il}. По Теореме 1.2

$$\overline{F}(I_{1l}, \ldots, I_{nl}) = F(I_{1l}, \ldots, I_{nl}),$$

что доказывает лемму. $\square$

Как результат Теоремы 3.1 получаем алгоритм двусторонней оценки множества значений функции $f(x_1, \ldots, x_n)$ при условии $x_i \in X_i \sqsubseteq T_i = (I_{il}, I_i)$, который мы опишем неформально.

Шагом алгоритма является вычисление $G(T_1)$ или $G(T_1, T_2)$ для некоторой функции g, входящей в состав выражения для функции f. T_1 и T_2 – уже полученные внешние и внутренние оценки для соответствующих подвыражений. Всего таких шагов надо сделать k, где k – количество функциональных символов, входящих в запись f.

Пусть g входит в запись f в виде

$$g(h(x_{j_1}, \ldots, x_{j_q})),$$

где h – некоторое подвыражение выражения для записи f. Внутренняя и внешняя оценки для g получаются, соответственно, из внутренней и внешней оценок для h применением к ним интервальной функции G.

Пусть g входит в запись f в виде

$$g(h_1(x_{j_1}, \ldots, x_{j_p}), h_2(x_{s_1}, \ldots, x_{s_q})),$$

где h_1 и h_2 – некоторые подвыражения выражения для функции f. Если наборы переменных, являющихся параметрами h_1 и h_2, не пересекаются, то внутренняя оценка для g получается из внутренних оценок для h_1 и h_2 применением простой интервальной арифметики. Если наборы переменных пересекаются, то должна применяться арифметика твинов. Внешняя оценка для g получается применением простой интервальной арифметики.

3.2 Постановка задачи и некоторые наблюдения

Вопросы точности внешнего интервального оценивания рассмотрены нами в разделе 1.4. Здесь мы сформулируем задачу оценки точности твинных вычислений [130], [132].

Определение 3.1. Пусть интервал X оценивается твином $T = (I_l, I)$, $X \sqsubseteq T$. Величину $|X| - |I_l|$ будем называть точностью внутренней оценки интервала X твином T.

Определение 3.2. В обозначениях предыдущего определения величину $\delta(T) = |I| - |I_l|$ будем называть абсолютной точностью двусторонней оценки, а величину $\varepsilon(T) = \delta(T)/|T|$ – относительной точностью двусторонней оценки неизвестного интервала твином T.

Ниже нас будут интересовать зависимости величин, которые только что определены, от размеров интервалов или твинов, являющихся аргументами оцениваемых функций и от вида функции. Рассмотрим точность внутренней оценки в случае, если оцениваемая функция является базовой, например, сложением или умножением [130].

Пусть мы имеем $I = I_1 + I_2$ и $I_1 \sqsubseteq T_1$, $I_2 \sqsubseteq T_2$, где I, I_1, I_2 – интервалы, а T_1 и T_2 – твины. Тогда согласно определению для $T = T_1 + T_2$ имеем $I \sqsubseteq T$. Какова же точность оценки, обеспечиваемой твином T? Обозначим

$$\begin{aligned} T_1 &= (a, A) = ([a^-, a^+], [A^-, A^+]), \\ T_2 &= (b, B) = ([b^-, b^+], [B^-, B^+]). \end{aligned}$$

Можно показать, что в случае $(|A| > |b|)$ & $(|B| > |a|)$ будем иметь $\varepsilon(T) = 1$. В случаях $|B| < |a|$ и $|A| < |b|$ имеем, соответственно,

$$\begin{aligned} \varepsilon(T) &= 1 - \frac{|a| - |B|}{|A| + |B|}, \\ \varepsilon(T) &= 1 - \frac{|b| - |A|}{|A| + |B|}. \end{aligned}$$

Действительно, если $|A| > |b|$ и $|B| > |a|$, то, обратившись к формулам (2.3), (2.4) и (2.7), видим, что $a^- + B^+ > a^+ + B^-$ и $b^- + A^+ > b^+ + A^-$, кроме того, всегда $a^- + B^+ > b^+ + A^-$ и $b^- + A^+ > a^+ + B^-$. Следовательно, $p > q$ и $\delta(T) = |T|$, $\varepsilon(T) = 1$.

Если, например, $|B| < |a|$, то $[p, q] = [a^- + B^+, a^+ + B^-]$ и $|[p, q]| = |a| - |B|$. Следовательно, $\delta(T) = |A| + |B| - (|a| - |B|) = |A| - |a| + 2|B|$, откуда непосредственно следует указанное выражение для $\varepsilon(T)$.

Из указанных соотношений видно: для успешной оценки по указанной схеме необходимо, чтобы внутренний интервал одного твина был больше внешнего интервала другого твина. Пусть больший твин обозначен через T_1, а меньший через T_2. Получаемая оценка тем точнее, чем меньше длина T_2 и чем длина внутреннего интервала твина T_1 ближе к длине его внешнего интервала.

Рассмотрим умножение твинов. Пусть $I = I_1 \cdot I_2$ и $I_1 \sqsubseteq T_1$, $I_2 \sqsubseteq T_2$, где I, I_1, I_2 – интервалы, а T_1 и T_2 – твины. Тогда, согласно определению для $T = T_1 \cdot T_2$, имеем $I \sqsubseteq T$.

Существует несколько различных случаев, в которых выражения для оценки точности различаются. Существенных из них три. Остальные либо сводятся к трем перестановкой сомножителей или изменением знака, либо являются тривиальными. Полученные ниже формулы непосредственно выводятся из формул (2.5) и (2.8), определяющих умножение твинов.

1. $0 \notin A, 0 \notin B$. Для простоты можно считать, что $A > 0$ и $B > 0$, остальные подслучаи легко сводятся к этому. Если $a^+/a^- < B^+/B^-$ и $b^+/b^- < A^+/A^-$, то $\varepsilon(T) = 1$. Если $a^+/a^- \geq B^+/B^-$, то

$$\varepsilon(T) = \frac{(l(A) + 1)(l(B) - 1) + \sigma_1 l(B) + \sigma_2}{l(A)l(B) - 1},$$

где $\sigma_1 = (a^- - A^-)/A^-$, $\sigma_2 = (A^+ - a^+)/A^-$, $l(A) = A^+/A^-$, $l(B) = B^+/B^-$. Подслучай $b^+/b^- \geq A^+/A^-$ сводится к предыдущему взаимной заменой T_1 и T_2.

2. $0 \in a$, $0 \notin B$. В этом случае оценка точности следующая:

$$\varepsilon(T) = 1 - \frac{a^+ - a^-}{l(B)(A^+ - A^-)}.$$

3. $0 \in A$, $0 \notin a$, $0 \notin B$. Этот случай в известном смысле носит вырожденный характер. Твин T_1 расположен таким образом, что 0 располагается вне внутреннего интервала и внутри внешнего. Как и в первом случае, будем считать, что $a^+/a^- \geq B^+/B^-$. Оценка имеет вид

$$\varepsilon(T) = 1 - \frac{a^+ - a^-}{A^+ - A^-} + \frac{a^+(l(B) - 1)}{(A^+ - A^-)l(B)}.$$

Анализ полученных выражений приводит к следующим заключениям. Твинная оценка приводит к нетривиальному результату, если хотя бы у одного твина внешний интервал не содержит нуля. Твинная оценка произведения двух интервалов тем лучше, чем лучше выполняются следующие условия: одна из величин $l(A)$ и $l(B)$ должна быть существенно больше другой; у твина, у которого эта величина больше, внутренний интервал должен быть близок по длине к внешнему.

3.3 Точность твинных вычислений

Рассмотрим задачу двусторонней оценки множества значений непрерывной функции одной переменной. Пусть дана функция $f(x)$, $x \in R$. Функция f, как и ранее в нашем изложении, задана в виде суперпозиции фиксированных базовых функций одной или двух переменных. Пусть также задан интервал $I \in I(R)$, на котором функцию необходимо оценить. При оценке f мы пользуемся твинными расширениями базовых функций, задаваемыми формулами (2.9), (2.10) и (2.11).

Определение 3.3. Будем говорить, что двусторонняя твинная оценка f на I невырождена (и, соответственно, что функция f невырождена), если при применении твинных расширений базовых функций двух переменных никогда оба значения Φ в формуле (2.11) одновременно не обращаются в пустые множества. Определение одинаково применимо как для функции одной переменной, так и для функции многих переменных.

Определение 3.4. Будем говорить, что двусторонняя твинная оценка функции f одной переменной абсолютно невырождена на I (и, соответственно, что f абсолютно невырождена), если для любого интервала $I' \subseteq I$ она невырождена на I'.

Пример 3.2. Невырожденная функция не обязательно является абсолютно невырожденной. Пусть $f(x) = 2x^2 + (x-1)^2$. $I = [0,1]$. Оценка на I невырождена. Действительно, положим $T = ([0,1],[0,1])$.

$$F(T) = ([0,2],[0,2]) + ([0,1],[0,1]) = ([1,2],[0,3]).$$

С другой стороны, возьмем $I' = [0, 0.5]$, $T' = (I', I')$. Имеем

$$F(T') = ([0,0.5],[0,0.5]) + ([0.25,1],[0.25,1]) = (\emptyset,[0.25,1.5]).$$

В случае оценивания функции нескольких переменных в общем случае нельзя подходящим образом определить понятие, аналогичное абсолютной невырожденности: выбирая $X' \subseteq X$ и $Y' \subseteq Y$, как правило, можно добиться вырожденности оценки $f(x,y)$ на $X' \times Y'$ просто изменением соотношения длин X' и Y'. Целесообразно, например, для функций многих переменных ограничиться аргументами с фиксированными соотношениями сторон. Например, можно в качестве аргументов рассматривать n-мерные интервалы вида

$$([x_1, x_1 + \alpha_1 \Delta], \ldots, [x_n, x_n + \alpha_n \Delta]),$$

где $\alpha_1, \ldots, \alpha_n$ – фиксированные числа, а $x_1, \ldots, x_n, \Delta$ могут быть выбраны любыми.

Определение 3.5. Будем говорить, что двусторонняя твинная оценка функции f многих переменных $(\alpha_1, \ldots, \alpha_n)$-абсолютно невырождена на $I \in I(R^n)$ (и, соответственно, что f $(\alpha_1, \ldots, \alpha_n)$-абсолютно невырождена), если для любого интервала

$$I' \subseteq I, \quad I' = ([x_1, x_1 + \alpha_1 \Delta], \ldots, [x_n, x_n + \alpha_n \Delta])$$

она невырождена на I'.

Отметим два свойства внутренней и двусторонней оценок, которые оформим в виде леммы.

Лемма 3.3. Пусть непрерывная функция $f(x)$, $x \in R^n$ удовлетворяет условию Липшица на интервале $X \in I(R^n)$. Тогда существуют такие α и β, что если посредством твинных вычислений получена двусторонняя оценка для $X' \subseteq X$:

$$\overline{F}(X') \sqsubseteq T = (I'_l, I'),$$

то:

а) Точность внутренней оценки удовлетворяет свойству

$$|\overline{F}(X')| - |I'_l| \leq \alpha |X'|.$$

б) Точность двусторонней оценки удовлетворяет свойству

$$|I'| - |I'_l| \leq \beta |X'|.$$

Доказательство следует из того, что для функций, удовлетворяющих условию Липшица, имеем

1. $\exists\gamma:\ |\overline{F}(X')| \leq \gamma|X'|$ (по определению условия Липшица);
2. $\exists\delta:\ |I'| \leq \delta|X'|$ (по Теореме 1.3).

$\square$

3.4 Алгоритм поиска внутренней оценки, использующий разбиение исходного интервала

Как нами было установлено в предыдущем параграфе, точность внутренней оценки зависит от размеров интервала, на котором оценивается функция. Чем у́же интервал, тем более точной является оценка. В этой ситуации можно разбить исходный интервал на части, оценить функцию на каждом подынтервале в отдельности, объединить полученные оценки и тем самым получить оценку функции на исходном интервале. Эта идея не нова. Она была с успехом применена при работе с внешними оценками (см. разделы 1.4 и 2.6). В настоящем параграфе мы применим эту идею для внутренней оценки множества значений функции. Результатом будет являться схема соответствующего алгоритма.

Пусть задан интервал I, который разбит на два подынтервала, пересекающихся в одной точке s, $I = I' \cup I''$, $I' \cap I'' = s$. Известны оценки

$$\begin{aligned} \overline{F}(I') &\sqsubseteq T' = (X_l', X'); \\ \overline{F}(I'') &\sqsubseteq T'' = (X_l'', X''). \end{aligned}$$

Нетрудно видеть, что твин $T = (X_l, X)$, оценивающий $\overline{F}(I)$, может быть получен следующим образом:

$$X_l = \psi(X_l', X_l''); \quad X = \psi(X', X''),$$

где функция ψ определяется формулой (2.6).

Для внутренней оценки верно также следующее утверждение: если $I' \subseteq I$ и известно, что внутренняя оценка для $\overline{F}(I')$ равна X'_l, то $\overline{F}(I) \supseteq X'_l$, следовательно, X'_l может быть взята также в качестве внутренней оценки для $\overline{F}(I)$.

Приведенные рассуждения служат обоснованием для следующей схемы алгоритма получения внутренней оценки множества значений функции посредством твинной арифметики. Пусть задана функция f, которую необходимо оценить изнутри на множестве $I \in I(R^n)$. Если $n = 1$, то будем считать, что оценка f на I абсолютно невырождена, а если $n > 1$, что существует набор $(\alpha_1, \ldots, \alpha_n)$, для которого оценка $(\alpha_1, \ldots, \alpha_n)$-абсолютно невырождена.

Сперва мы приведем общую схему алгоритма, а потом поясним некоторые детали. Наш алгоритм будет искать наилучшее приближение левой границы внутренней оценки. Поиск правой границы производится аналогично. Для поиска правой границы можно также применить алгоритм поиска левой границы к функции $-f$.

Алгоритм 1. Общий алгоритм поиска приближенной левой границы интервала, являющегося внутренней оценкой множества значений известной функции на известном интервале.

1. Поместим в список L единственный элемент – исходный интервал I. Вычислим твинную оценку $F((I, I))$. Пусть она равна $([a^-, a^+], [A^-, A^+])$. В качестве текущего приближения левой границы внутренней оценки (назовем его z) примем a^-.

2. Извлечем из списка L первый элемент. Назовем его I'.

3. Разобьем I' на части в соответствии с каким-нибудь правилом, которое конкретизируем ниже. Полученные интервалы назовем $I'_1, \ldots, I'_k$.

4. Для каждого $j = 1, \ldots, k$ вычислим $F((I'_j, I'_j)) = ([a_j^-, a_j^+], [A_j^-, A_j^+])$.

5. Положим новое значение z равным минимуму из его старого значения и всех значений $a_1^-, \ldots, a_k^-$.
6. Проведем сортировку интервалов $I_1', \ldots, I_k'$ и часть из них исключим из рассмотрения.
7. Оставшиеся интервалы I_j' включим в список L, переупорядочив список L в соответствии с каким-нибудь критерием.
8. Пересмотрим список L, исключив, если это возможно, часть содержащихся в нем интервалов, про которые можно сделать вывод, что оценки функции на них не способны повлиять на итоговый результат.
9. Проверим, не следует ли завершить алгоритм. Если да, то закончим его выполнение. Результатом будет текущее значение z. Если нет – перейдем к пункту 2.

Заметим, что целый ряд шагов алгоритма (а именно, шаги 3, 6, 7, 8, 9) недоопределен. Эти шаги могу быть определены по-разному. Соответственно, мы получим семейство алгоритмов, отличающихся друг от друга областью эффективного применения, вычислительной сложностью и т. д. Мы рассмотрим эти шаги несколько ниже, а пока обсудим сам алгоритм.

Список L содержит подынтервалы исходного интервала. Он организован таким образом, что, во-первых, подынтервалы I', на которых левая граница интервала $\overline{F}(I')$ гарантированно удалена от левой границы $\overline{F}(I)$, в нем не содержатся и, во-вторых, порядок элементов списка L таков, что подынтервалы, находящиеся в голове списка, при их рассмотрении с наибольшей вероятностью дадут хорошее приближение к левой границе $\overline{F}(I)$. Следует также отметить, что из-за абсолютной невырожденности все внутренние оценки существуют в виде непустых интервалов.

На шаге 3 алгоритма необходимо разбить рассматриваемый интервал на части. В одномерном случае самым простым выходом будет разбить интервал пополам. В многомерном случае необходимо так разбить интервал I, чтобы полученные подынтервалы имели соотношение длин сторон $\alpha_1 : \alpha_2 : \cdots : \alpha_n$. Если интервал I' разбить вдоль каждой координаты пополам, то получим 2^n подынтервалов, удовлетворяющих нужному условию. Мы молчаливо предполагаем, что исходный интервал I также имеет нужное соотношение длин сторон. Если это не так, то до начала выполнения алгоритма необходимо преобразовать задачу. Исходный интервал I необходимо разбить на подынтервалы с нужным соотношением длин сторон, выполнить алгоритм с каждым подынтервалом в отдельности и объединить полученные разультаты.

Нахождение исходного разбиения интервала I на подынтервалы с фиксированным соотношением длин сторон – отдельная задача, выходящая за рамки нашего рассмотрения. Отметим, что в большинстве случаев, если оценка функции $(\alpha_1, \dots, \alpha_n)$-абсолютно невырождена, то она и $(\beta_1, \dots, \beta_n)$-абсолютно невырождена для $\beta_1, \dots, \beta_n$, близких к соответствующим $\alpha_1, \dots, \alpha_n$. Варьируя α_i в некоторой ε-окрестности, легко предложить искомое исходное разбиение.

На шаге 6 необходмо отсортировать подынтервалы исходного интервала и часть из них исключить из расмотрения. Наиболее простым критерием сортировки может быть следующий. Если для некоторого интервала I'_j на шаге 4 была получена внешняя оценка $[A_j^-, A_j^+]$ и $A_j^- \geq z$, то интервал I'_j можно исключить из рассмотрения. Ясно, что никакой его подынтервал не может улучшить текущее приближение z. Представляется, что на шестом шаге алгоритма возможно введение дополнительных критериев, позволяющих отбросить лишние подынтервалы и тем повысить эффективность алгоритма.

На седьмом шаге новые интервалы должны быть включены в список L таким образом, чтобы в голове списка располагался интервал, на котором достигается наименьшее значение левой границы внутренней оценки, и последующие интервалы располагались в порядке возрастания данного показателя. Заметим, что на данном шаге алгоритма будет полезен и более тонкий критерий упорядочивания интервалов в списке L. Например, представляется разумным ограничить разброс размеров подынтервалов, содержащихся в L. Например, возможен следующий критерий: помещать в голову списка интервал с наименьшим значением левой границы внутренней оценки лишь тогда, когда отношение размеров наибольшего и наименьшего интервалов, содержащихся в L, не превышает 2^m при фиксированном m. Если соотношение превышает 2^m, то необходимо помещать в голову списка интервал с наибольшим размером.

На шаге 8 следует исключить из списка L все ранее включенные в него интервалы, которые не удовлетворяют критерию $A_j^- < z$ для нового значения z.

Вопрос об условии прекращения работы алгоритма является достаточно сложным. Из практических соображений хотелось бы прекращать работу алгоритма тогда, когда мы убеждены, что z отстоит от нижней границы интервала $\overline{F}(I)$ не более, чем на заданное значение ε. Если рассмотреть все интервалы I_j, находящиеся в списке на некотором шаге алгоритма, и найти для них $Z = \min\limits_j(A_j^-)$, то нижняя граница $\overline{F}(I)$ находится в интервале $[Z, z]$. Длина такого интервала и может являться критерием окончания работы алгоритма. Если критерий упорядочивания списка L выбран таким образом, что размер наибольшего интервала в нем неограниченно убывает (как, например, в предложенном выше упорядочивании), то в случае, если функция f

удовлетворяет условию Липшица, по Теореме 1.3 и Лемме 3.3 размеры интервала $[Z, z]$ будут неограниченно уменьшаться.

Действительно, для любого j имеем $|a_j^- - A_j^-| \leq \frac{\alpha}{2}|I_j|$, $z \in [a_j^-, A_j^-]$, следовательно, $|Z - z| = |\min A_j^- - z| \leq \frac{\alpha}{2} \max |I_j|$. При неограниченном уменьшении $|I_j|$ величина $|Z - z|$ также будет неограниченно убывать.

3.5 Новое определение твинной арифметики

Как нетрудно видеть, существенным условием применения алгоритма, описанного в предыдущем параграфе, является абсолютная невырожденность твинной оценки рассматриваемой функции. Условие приводит к ограничениям в применении алгоритма. Оно тем более неприятно, что оно, как правило, не может быть легко проверено до начала работы алгоритма. Невыполнение этого условия может выявиться лишь в ходе работы алгоритма, когда для некоторого подынтервала исходного интервала окажется, что оценка функции на нем является вырожденной.

С другой сторны, условие абсолютной невырожденности можно ослабить путем некоторой модификации самого алгоритма. Действительно, невырожденность оценки существенна лишь для подынтервалов, на которых граница внутренней оценки множества значений близка к границе действительного множества значений. Для других подынтервалов невырожденность несущественна, потому что в ходе работы алгоритма оценки функции на них вообще выпадают из рассмотрения.

Ниже мы предложим еще одно определение твинной арифметики [132]. Эта арифметика свободна от вырождения внутреннего интервала: внутренний интервал может иметь левую границу больше правой. Применение такой арифметики может оказаться полезным как для алгоритма внутренней оценки из предыдущего параграфа, так и для других твинных алгоритмов.

Множество обобщенных твинов обозначим через $T^*(R)$.

Определение 3.6. Элементами множества $T^*(R)$ являются пары интервалов вида

$$D = (Y_1, Y_2) = ([Y_1^-, Y_1^+], [Y_2^-, Y_2^+]), \quad Y_1^- \leq Y_2^-, \ Y_1^+ \leq Y_2^+.$$

Определение 3.7. Оценить неизвестный интервал $I = [x, y]$ обобщенным твином – значит найти такой $T = ([a_1, a_2], [a_3, a_4])$, что

$$x \in [a_1, a_2], \quad y \in [a_3, a_4].$$

Обозначим факт такой оценки через $I \sqsubseteq^* T$.

Заметим, что если при рассмотрении обобщенных твинов ограничиться лишь такими твинами, у которых $Y_1^+ \leq Y_2^-$, то между обычными и обобщенными твинами будет иметь место взаимнооднозначное соответствие. Если обычный твин обозначить $(X_l, X) = ([X_l^-, X_l^+], [X^-, X^+])$, то соответствие задается соотношениями

$$Y_1^- = X^-; \ Y_1^+ = X_l^-; \ Y_2^- = X_l^+; \ Y_2^+ = X^+.$$

Определение 3.8. Пусть $T_1, T_2 \in T^*(R)$, $T_1 = ([a_1, a_2], [a_3, a_4])$, $T_2 = ([b_1, b_2], [b_3, b_4])$. Определим арифметические действия над обобщенными твинами следующим образом:

$$\begin{aligned}
-T_1 &= ([-a_4, -a_3], [-a_2, -a_1]);\\
1/T_1 &= ([1/a_4, 1/a_3], [1/a_2, 1/a_1]), \quad 0 \notin [a_1, a_4];\\
T_1 + T_2 &= ([a_1 + b_1, \min(a_2 + b_4, b_2 + a_4)], [\max(a_3 + b_1, b_3 + a_1), a_4 + b_4]).
\end{aligned}$$

Для определения умножения введем дополнительные обозначения:

$$S_1 = a_2 \cdot [b_1, b_4]; \quad S_2 = a_3 \cdot [b_1, b_4];$$

$$S_3 = b_2 \cdot [a_1, a_4]; \quad S_4 = b_3 \cdot [a_1, a_4].$$

Тогда

$$\begin{aligned} T_1 \cdot T_2 \;=\; & ([\min(a_1b_1, a_1b_4, a_4b_1, a_4b_4), \min_{i=1,\dots,4} S_i^+], \\ & [\max_{i=1,\dots,4} S_i^-, \; \max(a_1b_1, a_1b_4, a_4b_1, a_4b_4)]). \end{aligned}$$

Утверждение 3.1. Пусть X и Y – интервалы, а T_1 и T_2 – обобщенные твины. Тогда

$$X \sqsubseteq^* T_1 \;\Rightarrow\; \diamond X \sqsubseteq^* \diamond T_1;$$

$$X \sqsubseteq^* T_1 \,\&\, Y \sqsubseteq T_2 \;\Rightarrow\; X \circ Y \sqsubseteq^* T_1 \circ T_2,$$

где $\diamond$ и $\circ$, соответственно, унарная и бинарная операции из числа только что определенных.

Утверждение мы приводим без доказательства. Доказательство аналогично доказательству Теоремы 2.2.

3.6 Сложность твинных вычислений

В разделе 1.8 нами был приведен ряд ранее доказанных утверждений о сложности интервальных вычислений. Что же можно сказать о сложности твинных вычислений? В настоящем параграфе мы покажем, что результаты, касающиеся сложности интервальных вычислений, могут быть перенесены на твинный случай.

В теории интервальных вычислений в качестве основной рассматривается задача оценки вычислительной сложности алгоритмов поиска точных (или ε-приближенных) границ объединенного интервального расширения функции некоторого вида, чаще всего полинома. При этом такие алгоритмы должны работать на любых функциях рассматриваемого вида при любых (или, во всяком случае, принадлежащих достаточно широкому классу) входных интервалах.

Пусть задана функция $f(x)$, $x \in R^n$. Обозначим $T = (I_l, I)$. Объединенным твинным расширением $\overline{F}(T)$ непрерывной функции f естественно назвать функцию, которая вычисляет твин, отвечающий следующим условиям:

- внутренний интервал твина содержит в точности те значения функции, которые она принимает, при условии, что аргумент принадлежит I_l;

- внешний интервал твина содержит в точности те значения функции, которые она принимает, при условии, что аргумент принадлежит I.

Можно записать

$$\overline{F}(T) = (\{y \mid \exists x' \in I_l : \ y = f(x')\}, \{y \mid \exists x' \in I : \ y = f(x')\}).$$

Такое понимание естественного твинного расширения соответствует нашему пониманию твина как двух интервалов, между которыми заключен неизвестный интервал.

Имеем $\overline{F}(T) = (\overline{F}(I_l), \overline{F}(I))$. Сложность вычисления объединенного твинного расширения равна сложности вычисления соответствующего объединенного интервального расширения (помноженной на 2). Теоремы 1.8 – 1.12 переносятся на твинный случай.

Более интересным является обобщение Теоремы 1.13 на твинный случай. В дополнение к определениям раздела 1.8 введем несколько новых определений. Большинство обозначений заимствовано из раздела 1.8.

Определение 3.9. Пусть $\eta > 0$ – вещественное число. Будем говорить, что твины $([x^-, x^+], [X^-, X^+])$ и $([y^-, y^+], [Y^-, Y^+])$ η-близки, если $|x^- - y^-| \leq \eta$, $|x^+ - y^+| \leq \eta$, $|X^- - Y^-| \leq \eta$ и $|X^+ - Y^+| \leq \eta$.

Определение 3.10. Пусть U – алгоритм, вычисляющий объединенное твинное расширение функции, $\eta > 0$, $f(x_1, \ldots, x_n)$ – рациональная функция с рациональными коэффициентами, $T_1, \ldots, T_n$ – набор твинов. Будем говорить, что U η-точен на f и $T_1, \ldots, T_n$, если для любого набора твинов $S_1, \ldots, S_n$ с рациональными границами, каждый из которых η-близок к соответствующему твину T_i, алгоритм U возвращает четыре точные границы $\overline{F}(S_1, \ldots, S_n)$.

Определение 3.11. Будем говорить, что алгоритм U *почти всегда* точен для узких входных твинов, если для каждой компактной области D, $\mu(D) > 0$, для каждой рациональной функции с рациональными коэффициентами f, которая конечна на открытом множестве $N \supset D$, и для любого $\varepsilon > 0$ существуют $\delta > 0$ и $\eta > 0$ такие, что для (D, ε)-почти всех $x = (x_1, \ldots, x_n)$ выполняется следующее условие: если все n входных твинов $T_i = ([x_i - \Delta_i, x_i + \Delta_i], [x_i - \Delta_i', x_i + \Delta_i''])$ отвечают условию $\Delta_i < \delta$, $\Delta_i' < \delta$, $\Delta_i'' < \delta$, то алгоритм U является η-точным на f и T_i.

Теорема 3.2. Существует алгоритм U с полиномиальным временем выполнения, который для рациональной функции f с рациональными коэффициентами и твинов $T_1, \ldots, T_n$ возвращает оценку для $\overline{F}(T_1, \ldots, T_n)$, которая почти всегда точна для узких входных твинов.

Доказательство. По Теореме 1.13 существует полиномиальный алгоритм U', который вычисляет оценку для $\overline{F}(I_{l1}, \ldots, I_{ln})$, где I_{li} – внутренние интервалы твинов T_i, которая почти всегда точна для узких входных интервалов. Этот же самый алгоритм U' вычисляет оценку для $\overline{F}(I_1, \ldots, I_n)$, где I_i – внешние интервалы твинов T_i, обладающую аналогичными свойствами.

В качестве алгоритма, существование которого доказывается Теоремой 3.2, возьмем алгоритм U, состоящий из двукратного выполнения алгоритма U' для наборов внутренних и внешних интервалов. Ясно, что он будет иметь полиномиальное время выполнения. Остается показать, что он будет почти всегда точен для узких твинов.

Зафиксируем D и f из определения почти всегда точного твинного алгоритма (Определение 3.11). Зафиксируем также ε. По Теореме 1.13 для $\varepsilon/2$ существуют δ и η такие, что:

- для $(D, \varepsilon/2)$-почти всех $x = (x_1, \ldots, x_n)$ и узких $[x_i^-, x_i^+] = [x_i - \Delta_i, x_i + \Delta_i]$, $\Delta_i < \delta$ алгоритм U' является η-точным на f и $[x_i^-, x_i^+]$.
- для $(D, \varepsilon/2)$-почти всех $x' = (x'_1, \ldots, x'_n)$ и узких $[X_i^-, X_i^+] = [x'_i - \Delta'_i, x'_i + \Delta'_i]$, $\Delta'_i < \delta$ алгоритм U' является η-точным на f и $[X_i^-, X_i^+]$.

Рассмотрим меру множества, для которого выполняется следующее условие: если все n входных твинов

$$T_i = ([x_i - \Delta_i, x_i + \Delta_i], [x_i - \Delta'_i, x_i + \Delta''_i])$$

отвечают условию $\Delta_i < \delta$, $\Delta'_i < \delta$, $\Delta''_i < \delta$, то алгоритм U является η-точным на f и T_i. Обозначим это множество D^*.

Для того, чтобы алгоритм являлся η-точным, необходимо, чтобы η-точными являлись одновременно алгоритмы, вычисляющие внутренний и внешний интервал твина. Обозначим через D_1 и D_2 подмножества множества D, на которых алгоритм U' неточно вычисляет, соответственно, внутренний и внешний интервалы. Имеем: $\mu(D_1) < \frac{\varepsilon}{2}\mu(D)$ и $\mu(D_2) < \frac{\varepsilon}{2}\mu(D)$. Следовательно, $\mu(D_1 \cup D_2) < \varepsilon\mu(D)$, откуда $\mu(D) - \mu(D^*) < \varepsilon\mu(D)$. Следовательно, весь алгоритм U почти всегда точен.

□

В третьей главе было проведено исследование свойств твинных вычислений, в первую очередь тех свойств, которые связаны с точностью твинных алгоритмов. Основным практическим результатом является схема алгоритма поиска внутренней оценки. Результаты, полученные в главе, служат переходом от теоретических построений второй главы к практическим задачам, часть из которых будет рассмотрена ниже.

Глава 4. Применение твинных вычислений

4.1 Основные области использования твинов

В предыдущих главах было продемонстрировано, что аппарат твинных вычислений может быть применен для решения различных задач. В настоящей главе мы рассмотрим несколько областей приложения твинных вычислений. Наиболее эффективное применение твинные вычисления найдут при решении чисто математических задач, а также разнообразных задач, объединяемых научным направлением "искусственный интеллект". Что же общего у таких применений? Какие фундаментальные свойства окружающего мира можно хорошо описать твинами?

За аналогией опять обратимся к интервальным вычислениям. Пусть у нас имеется некоторая действительная величина, точное значение которой неизвестно. Она может иметь как физическую природу, так и чисто математическое (абстрактное) происхождение. Для работы с этой величиной, например, на компьютере, мы должны представить ее числом, как правило числом, принадлежащим некоторому множеству (целых чисел, рациональных чисел, машинных чисел и т. д.). Такое представление не может нас полностью удовлетворить – мы не знаем отклонения действительного значения величины от значения-представителя. На помощь приходит интервал, который представляет неизвестную величину, заключая ее внутри себя.

Кроме вещественных величин часто приходится иметь дело с изначально интервальными величинами. В физическом мире такие величины появляются, как правило, при рассмотрении всех возможных значений действительной величины при изменении параметра, от которого эта величина зависит. Параметром может быть, например, время или координата. Конкретные примеры будут приведены ниже в этой главе. Точное значение рассматриваемой интервальной величины неизвестно. Подобно тому, как действительная величина представлялась интервалом, интервальная величина должна быть представлена твином, который оценивает неизвестный интервал снаружи и изнутри. В рамках математических задач роль оцениваемого интервала также играют параметризованные вещественные величины при определенных границах изменения параметра (параметров). Как рассмотренные ниже, так и примеры использования твинов, оставленные за рамками настоящей работы, в значительной степени соответствуют разработанному здесь подходу.

4.2 Применение твинов при решении математических задач

Как уже неоднократно отмечалось, "основная задача" интервального анализа может быть сформулирована следующим образом: для функции $f(x)$, $x \in R^n$ и для данных интервалов $X_1, \ldots, X_n$ вычислить или оценить множество значений фунции при условии принадлежности аргументов x_i интервалам X_i. Под оценкой, как правило, понимается внешняя оценка некоторым интервалом.

Указанная задача может быть обобщена в нескольких направлениях. Во-первых, вместо множеств значений функций мы можем рассматривать неизвестные множества, полученные иными путями, например, множества значений, являющиеся решениями систем уравнений и/или неравенств или множества, на которых явно или неявно заданный предикат получает истинное значение.

Во-вторых, вместо оценки интервалом неизвестное множество можно оценивать множествами другого типа, например обобщенными интервалами (в частности теми, что были рассмотрены в предыдущих главах).

В-третьих, можно отказаться от условия внешней оценки. Оценка может быть внутренней, может быть какой-нибудь иной, близкой в некотором смысле к оцениваемому множеству.

В работах [33], [158] предлагается более широкое понимание рассматриваемого круга задач. Под интервальной (локализационной) задачей понимается четверка $(\mathcal{S}, \mathcal{E}, \mathcal{M}, \varrho)$. Здесь $\mathcal{S}$ – семейство оцениваемых множеств, $\mathcal{E}$ – семейство оценочных множеств. Символом $\mathcal{M}$ обозначается модальность, т. е. отношение между элементами множеств $\mathcal{S}$ и $\mathcal{E}$. "Основная задача" интервального анализа требует, чтобы оцениваемое множество было подмножеством оценивающего. Символом ϱ обозначена метрика, измеряющая каким-либо образом близость оценивающего и оцениваемого множеств.

Для одного и того же оцениваемого множества можно по-разному зафиксировать оценивающие множества и модальность. Таким образом, мы можем оценить одно и то же неизвестное множество двумя или большим числом принципиально различных способов. Каждый из таких способов добавляет новую информацию о неизвестном оцениваемом множестве.

В работах [35], [36], [108] предлагается идея многоаспектности интервальных вычислений. Она предусматривает одновременное вычисление различных оценивающих множеств. При реализации различных алгоритмов поддержание многоаспектности способно, во-первых, увеличить наши знания об оцениваемом объекте за счет разного типа оценивающих множеств и разных типов оценки (т. е. модальностей) и, во-вторых, повысить точность оценки за счет взаимного уточнения разных оценивающих множеств на промежуточных этапах выполнения алгоритма.

Мы только что показали два полюса в формулировании задачи интервальных вычислений. Певый полюс – исходная задача, с которой берет свое начало интервальная математика. Второй – максимально обобщенная задача локализации неизвестного множества. Твинная арифметика занимает на этой шкале промежуточное место. Посредством твинных вычислений мы одновременно вычисляем два аспекта неизвестного множества, являющегося интервалом. Первый аспект – внешняя оценка, та самая, что обеспечивается традиционными интервальными вычислениями, второй – внутренняя оценка.

Если посмотреть, что представляют из себя элементы четверки $(\mathcal{S}, \mathcal{E}, \mathcal{M}, \varrho)$ в случае твинных вычислений, то можно увидеть, что в качестве множества $\mathcal{S}$ выступают интервалы, в качестве $\mathcal{E}$ – твины, в качестве модальности $\mathcal{M}$ выступает отношение $I \sqsubseteq T$, заданное при определении твинной арифметики, а в качестве метрики ϱ – абсолютная точность двусторонней оценки, определенная в разделе 3.2.

Рассмотрим несколько примеров твинной оценки в конкретных математических задачах.

Пример 4.1. Первый пример фактически уже был рассмотрен. Он состоит в двусторонней оценке множества значений функции, явно заданной в виде символьного выражения. На самом деле множество конкретных практически важных задач сводится именно к этой постановке. Задача двусторонней оценки может быть решена многочисленными способами, различающимися по трудоемкости, точности и т. д. Нами ранее были рассмотрены алгоритмы прямой двусторонней оценки, модифицированный интервальный алгоритм глобальной оптимизации (параграф 2.6), алгоритм внутренней оценки с разбиением исходного интервала (параграф 3.4).

Пример 4.2. Второй пример связан с решением интервальной линейной системы уравнений. Постановка задачи заимствована из работ [154], [157], [31].

Рассматривается интервальная система линейных алгебраических уравнений

$$\mathbf{A}x = \mathbf{b}, \tag{4.1}$$

где $\mathbf{A}$ – интервальная матрица размерности $n \times n$, $\mathbf{b}$ – вектор размерности n. Задача поиска решения системы (4.1) может быть сформулирована несколькими способами, два из которых мы сейчас приведем.

Объединенным множеством решений системы (4.1) называется множество решений всех точечных систем $Ax = b$ при $A \in \mathbf{A}$ и $b \in \mathbf{b}$, то есть множество

$$\Sigma_{\exists\exists}(\mathbf{A}, \mathbf{b}) = \{x \in R^n \mid \exists A \in \mathbf{A}\ \exists b \in \mathbf{b} : Ax = b\}.$$

Как известно, множество $\Sigma_{\exists\exists}(\mathbf{A}, \mathbf{b})$ является многогранным множеством специального вида [31].

Другим определением множества решений (4.1) является так называемое допустимое множество решений. Допустимым множеством решений системы (4.1) является объединение всех точечных векторов $x \in R^n$ таких, что для любого $a \in \mathbf{A}$ выполняется $Ax \in B$, то есть множество

$$\Sigma_{\forall\exists}(\mathbf{A}, \mathbf{b}) = \{x \in R^n \mid \forall A \in \mathbf{A}\ \exists b \in \mathbf{b} : Ax = b\}.$$

Указанное множество, как и объединенное множество решений, является многогранным множеством.

В качестве численного примера рассмотрим систему, ставшую хрестоматийной [44], [31].

$$\begin{pmatrix} [2,4] & [-2,1] \\ [-1,2] & [2,4] \end{pmatrix} x = \begin{pmatrix} [-2,2] \\ [-2,2] \end{pmatrix}.$$

Объединенное и допустимое множества решений показаны на рисунке 4.1.

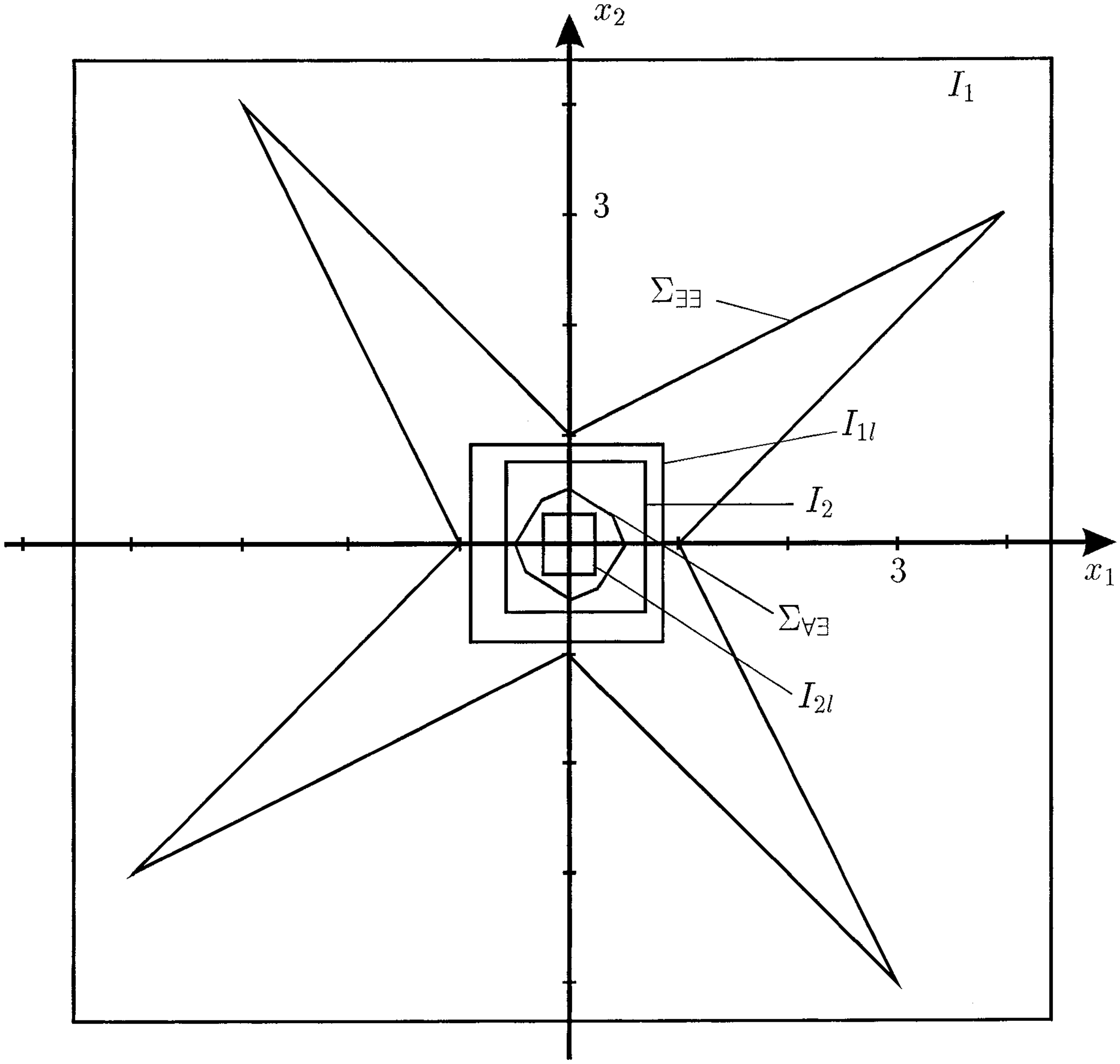

Рис. 4.1.

В работе [13] показано, что даже задача установления, являются ли множества $\Sigma_{\exists\exists}(\mathbf{A}, \mathbf{b})$ и $\Sigma_{\forall\exists}(\mathbf{A}, \mathbf{b})$ пустыми, является NP-полной. Для большого числа практически важных задач, таких, как задачи анализа и моделирования статических систем с интервальной неопределенностью, анализа чувствительности управляемых систем, исследовании операций и теории принятия решений [30], [136], [137], [155], [156], необходимо нахождение оценивающих твинов, для которых

$$\Sigma_{\exists\exists}(\mathbf{A},\mathbf{b}) \ \sqsubseteq \ T_1 = (I_{1l}, I_1);$$
$$\Sigma_{\forall\exists}(\mathbf{A},\mathbf{b}) \ \sqsubseteq \ T_2 = (I_{2l}, I_2).$$

Здесь мы позволили себе употребить символ $\sqsubseteq$ для обозначения двусторонней оценки многогранного множества, не дав ему строгого определения и надеясь, что оно очевидно, в частности из рис. 4.1.

Для нахождения внешних и внутренних оценок решений линейных систем интервальных уравнений предложены разнообразные методы (см., например, [2], [6], [136], [137], [154], [157] и др.). Полученные двусторонние оценки приводят к твинному приближению множества решений. Такие твинные приближения могут быть использованы при решении упомянутых выше задач моделирования статических систем с неопределенностью, анализа чувствительности, теории принятия решений и др., при представлении самого решения, вычисление которого NP-полно. При этом для операций с такими приближениями может быть использован весь аппарат твинной арифметики.

Пример 4.3. Рассмотрим задачу анализа статической системы в следующей постановке. Пусть внутреннее состояние системы описывается одной переменной $x \in R$. Единственный выходной параметр системы обозначим через b, $b \in R$. Входные параметры системы обозначим через $a_1, \ldots, a_n$. Мы намеренно ограничиваемся такой одномерной постановкой задачи, хотя, в общем случае, она имеет многомерную формулировку.

Пусть нам также известен закон функционирования системы, то есть зависимость между входом и выходом, в виде следующего уравнения

$$f(a_1, \ldots, a_n, x) = b. \tag{4.2}$$

Будем полагать, что f – непрерывная функция. Про входные параметры системы известно, что они изменяются в пределах $A_1, \ldots, A_n$, $a_i \in A_i$, $A_i \in I(R)$, $i = 1, \ldots, n$, а выходной параметр b принадлежит интервалу достижимости B, $b \in B$, $B \in I(R)$. Нашей задачей будет, исходя из известной информации о входных и выходных параметрах системы, оценить ее внутреннее состояние.

Мы будем использовать два соображения. Во-первых, зная интервалы изменения входных и выходных параметров системы, мы можем выяснить максимально широкий интервал допустимых значений внутреннего состояния, который мы обозначим через X,

$$X = \{x \in R \mid \exists a_1 \in A_1, \ldots, \exists a_n \in A_n, \exists b \in B : f(a_1, \ldots, a_n, x) = b\}. \tag{4.3}$$

Указанное множество X является внешней оценкой параметра состояния системы, совсем не обязательно все точки интервала X реально должны реализовываться при конкретном воплощении системы. Второй оценкой множества значений состояния системы является его внутренняя оценка, обозначим ее через X_l. Фиксация каждого значения внутреннего состояния $x \in X_l$ обеспечивает корректное состояние системы, то есть принадлежность выходного параметра соответствующему интервалу при любых значениях входных параметров. Имеем

$$X_l = \{x \in R \mid \forall a_1 \in A_1, \ldots, \forall a_n \in A_n : f(a_1, \ldots, a_n, x) \in B\}. \tag{4.4}$$

Нетрудно видеть, что $X_l \subseteq X$.

Если мы хотим, с одной стороны, рассматривать лишь реально возможные значения внутреннего состояния системы, а с другой стороны, включить в рассмотрение все те значения параметра, которые обеспечивают корректность при всех входных данных, то следует иметь дело с интервалом значений I, который мы оценили как $I \sqsubseteq (X_l, X)$.

Две различные трактовки внутреннего состояния системы как управляющего параметра и как неконтролируемого параметра, являющегося функцией внешних воздействий, хорошо известны. Различные примеры, обсуждение и ссылки можно найти в работах [30], [155]. В них автор рассматривает многомерный вариант задачи, который существенно сложнее одномерного хотя бы ввиду того, что множества, соответствующие множествам X_l и X в одномерном случае, в многомерном имеют непростую структуру и могут лишь быть оценены снаружи или изнутри (см. предыдущий пример). В нашем примере мы хотим лишь подчеркнуть, что неизвестный интервал параметров системы естественно представлять твином. Заметим, что изучаемый параметр x часто полагают входным параметром системы, также вкладывая одну из двух (управляющий/управляемый) семантик. В нашем примере мы полагаем параметр внутренним состоянием системы, что, как нам кажется, упрощает изложение.

Покажем, как в нашем случае вычислить интервалы X_l и X. Предположим дополнительно, что уравнение (4.2) разрешимо относительно x,

$$x = g(a_1, \dots, a_n, b). \tag{4.5}$$

Как нетрудно видеть из (4.3), множество X может быть получено по следующей формуле

$$X = \overline{G}(A_1, \dots, A_n, B)$$

или оценено снаружи с использованием естественного интервального расширения G функции g:

$$X \subseteq G(A_1, \ldots, A_n, B).$$

Рассмотрим один из возможных путей вычисления/оценки интервала X_l в одном из конкретных случаев.

Пусть x выражается через $a_1, \ldots, a_n, b$ формулой (4.5) и в запись функции g входят базисные функции $+$, $\cdot$ и унарные $-$ и $1/\cdot$, т.е. функция g – рациональная. Будем также считать, что b входит в выражение для функции g однократно. Докажем следующее утверждение.

Утверждение 4.1. Обозначим через $A'_1, \ldots, A'_n$ обобщенные интервалы Каухера, равные, соответственно, $\mathrm{dual}(A_1), \ldots, \mathrm{dual}(A_n)$ (см. раздел 1.6.2.1). Вычислим в арифметике Каухера функцию $G(A'_1, \ldots, A'_n, B)$, где G – естественное интервальное расширение g. Предположим, что в ходе вычислений не производится умножение и деление на интервал, границы которого имеют разный знак или хотя бы одна из границ равна нулю. Результат обозначим через X'_l. Если $X'_l \in I(R)$, то $X'_l \subseteq X_l$, то есть X'_l является внутренней оценкой X_l. Если, кроме того, в выражении для функции g переменные $a_1, \ldots, a_n$ встречаются по одному разу, то $X'_l = X_l$.

Идея использования арифметики Каухера для подобных вычислений принадлежит Гарденесу и Трепату (см., например, [61]).

Доказательство утверждения. Возьмем произвольный $x_0 \in X'_l$. Докажем, что

$$\forall a_1 \in A_1, \ldots, \forall a_n \in A_n \ \exists b \in B : \ g(a_1, \ldots, a_n, b) = x_0.$$

Уравнение

$$G(A'_1, \ldots, A'_n, B) = X'_l \qquad (4.6)$$

будем разрешать относительно B, шаг за шагом "раскручивая" выражение для G, т. е. перенося в правую часть подвыражения, не содержащие переменную B. Такой перенос всегда возможен в арифметике Каухера, т. к. интервальные уравнения $X + A = B$ и $X \cdot C = D$ в арифметике Каухера всегда разрешимы, если только интервал C имеет границы одного знака и не равные нулю, что мы дополнительно предположили в условиях утверждения. При разрешении уравнения (4.6) будут производиться четыре вида преобразований:

$$
\begin{aligned}
B + A = X &\quad\longrightarrow\quad B = X + (-\mathrm{dual}(A));\\
B \cdot A = X &\quad\longrightarrow\quad B = X \cdot (1/\mathrm{dual}(A));\\
-B = X &\quad\longrightarrow\quad B = -X;\\
1/B = X &\quad\longrightarrow\quad B = 1/X.
\end{aligned}
$$

Корректность таких преобразований обеспечивается элементарными свойствами арифметики Каухера (см., например, [62]). В последнем преобразовании $0 \notin X$, т. к. X получен как результат операции $1/B$.

По ходу решения в правой части уравнения будут появляться либо параметры A_i, либо конструкции вида $\mathrm{dual}(G_1(A'_k, \ldots, A'_m))$. Принимая во внимание известное свойство арифметики Каухера $\mathrm{dual}(I_1) \circ \mathrm{dual}(I_2) = \mathrm{dual}(I_1 \circ I_2)$, имеем

$$\mathrm{dual}(G_1(A'_k, \ldots, A'_m)) = G_1(A_k, \ldots, A_m).$$

Получаем уравнение

$$B = H(A_1, \ldots, A_n, X'_l), \tag{4.7}$$

которое полностью равносильно уравнению (4.6). Последнее уравнение содержит только обычные интервалы. Из него следует, что для любых $a_i \in A_i$, $i = 1, \ldots, n$ и $x_0 \in X'_l$ существует $b \in B$ такое, что выполняется (4.7) и, следовательно, (4.6). Итак, мы показали, что $x_0 \in X_l$.

Остается показать, что если все a_i встречаются в выражении для g однократно, то $X_l' = X_l$. По определению X_l имеем:

$$\forall x_1 \in X_l\ \forall a_i \in A_i, i = 1, \ldots, n\ \exists b \in B :\ x_1 = g(a_1, \ldots, a_n, b).$$

Производя преобразования, аналогичные тем, что были сделаны при переходе от (4.6) к (4.7), но не с интервальными, а с аналогичными действительными выражениями, получаем

$$\forall x_1 \in X_l\ \forall a_i \in A_i, i = 1, \ldots, n\ \exists b \in B :\ b = h(a_1, \ldots, a_n, x_1),$$

откуда, учитывая то, что $a_1, \ldots, a_n$ входят в h однократно и, следовательно, $H(A_1, \ldots, A_n, X_l) = \overline{H}(A_1, \ldots, A_n, X_l)$, имеем

$$B \supseteq H(A_1, \ldots, A_n, X_l).$$

Принимая во внимание (4.7) по свойству монотонности по включению получаем $X_l \subseteq X_l'$, следовательно $X_l = X_l'$. $\square$

Приведем один классичский пример, используемый для иллюстрации статических систем с интервальной неопределенностью еще со времен работы [61] (рис. 4.2).

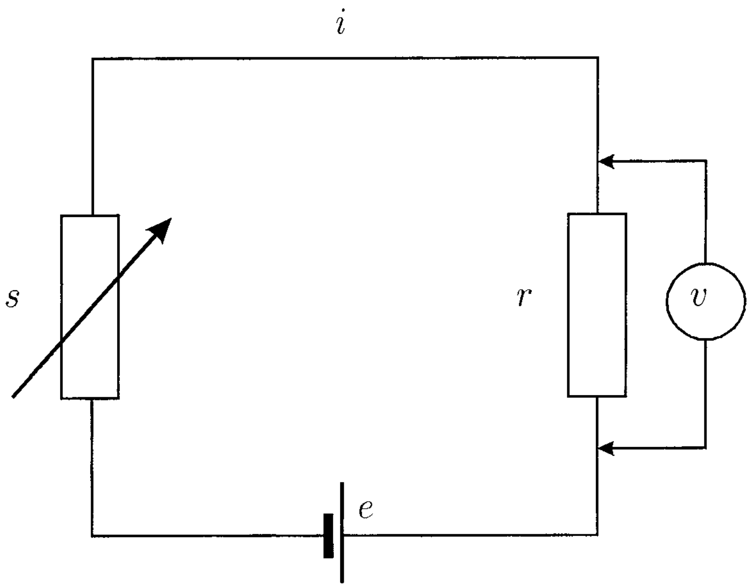

Рис. 4.2.

Дана электрическая цепь, состоящая из последовательно соединенных двух сопротивлений s и r, а также источника тока e. Входными данными системы являются: значение сопротивления r, изменяющееся в интервале R, и значение напряжения e, изменяющееся в интервале E. Выходным параметром является падение напряжения на сопротивлении r, обозначенное через v, которое необходимо поддерживать в интервале V. Внутреннее состояние системы характеризуется сопротивлением s, интервал изменения которого и является предметом рассматриваемого примера. Положим, что заданы следующие числовые значения интервалов: $V = [10, 20]$, $E = [40, 50]$, $R = [10, 11]$.

Как было указано выше, можно рассматривать две задачи. Первая – найти максимально возможный интервал изменения внутреннего состояния системы. Вторая – найти интервал допустимых значений внутреннего состояния системы, каждое из которых обеспечивает поддержание выходного параметра в нужном интервале при любых значениях входных параметров. Исходя из приведенных выше теоретических рассмотрений, можно вычислить соответствующие интервалы для s, которые мы обозначим через S и S_l.

Основной закон поведения системы, исходя из закона Ома, записывается в следующем виде:

$$s = r\Big(\frac{e}{v} - 1\Big).$$

Исходя из приведенных соображений о вычислении S и S_l, а также из того, что в выражении для s все параметры встречаются по одному разу, имеем

$$\begin{aligned} S &= [10, 11] \cdot \Big(\frac{[40, 50]}{[10, 20]} - 1\Big) = [10, 11] \cdot ([2, 5] - 1) \\ &= [10, 11] \cdot [1, 4] = [10, 44]; \end{aligned}$$

$$\begin{aligned} S_l &= [11,10]\cdot\left(\frac{[50,40]}{[10,20]}-1\right) = [11,10]\cdot([2.5,4]-1) \\ &= [11,10]\cdot[1.5,3] = [16.5,30]. \end{aligned}$$

4.3 Твины и задачи искусственного интеллекта

Аппарат интервальной математики достаточно широко используется при решении задач искусственного интеллекта. В задачах искусственного интеллекта традиционно приходится иметь дело с нечетко определенными значениями, полученными, например, путем обработки экспертных знаний. Другой источник неточных данных – физические инструменты, обеспечивающие лишь ограниченную точность. Такие данные также могут быть внесены в базу знаний наряду с экспертными знаниями.

Для математической обработки нечетких данных в задачах искусственного интеллекта используются несколько подходов, например, статистический подход, теория нечетких множеств. Методы интервальной математики также уже достаточно давно используются как в теоретических работах по искусственному интеллекту, так и в практических разработках. Хорошие обзоры приложений интервальной математики к задачам искусственного интеллекта можно найти в следующих источниках: [37], [40], [97], [145]. Из программных систем упомянем CLINAID [94], [95].

Приведем несколько примеров, или, лучше сказать, упомянем несколько идей использования интервалов при решении задач искусственного интеллекта.

В работах [68] и [134] рассматривается задача обработки знаний, полученных от экспертов. При применении традиционных вероятностных методов представления и обработки неопределенностей мера вероятности того или иного утверждения обычно представляется одним действительным числом из промежутка $[0, 1]$. В большинстве реальных ситуаций эксперт не может точно описать вероятность того или иного события, но может задать ее в виде интервала. Интервал при этом несет больше информации, чем просто число. Если эксперт оценивает вероятность интервалом, то он не только сообщает свое мнение о наиболее вероятном значении величины, но и характеризует меру своей уверенности. Чем у́же сообщенный интервал, тем более эксперт уверен в своей оценке.

Другим источником возникновения интервальных данных в экспертных системах является следующий. Пусть мы знаем вероятности некоторых атомарных утверждений и некоторых их логических комбинаций, но вероятность любой логической комбинации не может быть определена однозначно. В этом случае не остается ничего лучшего, чем в ответ на запрос о вероятности того или иного утверждения выдавать интервал, которому гарантированно принадлежит запрашиваемое значение (см. [57]).

При работе с базами знаний, использующими интервальные оценки вероятности событий, могут быть использованы интервальные методы, разработанные ранее для совсем других целей. Например, весьма важной является задача установления непротиворечивости алгебраической Байесовской сети [68], [134]. Это задача, как показано во второй из процитированных работ, может быть решена чисто интервальным алгоритмом, полученным модификацией известных алгоритмов глобальной оптимизации.

Другой областью искусственного интеллекта, где нашли применение интервальные методы, является описание мира, в котором функционирует управляемый (наблюдаемый) объект. Таким миром может быть как обычный дву- или трехмерный геометрический мир, так и абстрактное пространство с абстрактно описанными принадлежащими ему областями. В реальных задачах мир, окружающий интересующий нас объект, не может быть описан абсолютно точно. Размеры и взаимное расположение объектов задаются с некоторой неопределенностью, при обработке которой может быть использована интервальная техника. Положение управляемого объекта, его скорость, другие характеристики также описываются интервалами.

Очень часто интервальные методы используются совместно с методами нечеткой логики, что совершенно естественно: часто "степень уверенности" естественно описывать не действительным числом, а некоторым интервалом. Кроме того, широко распространены обобщения теории нечетких множеств, когда "степень уверенности" приписывается не каждому рассматриваемому значению, а интервалу рассматриваемых значений [95].

В качестве еще одного примера использования интервалов в типичной задаче искусственного интеллекта приведем распознавание образов, в данном случае – распознавание голоса человека [119]. Задано несколько входных тестовых сигналов, каждый из которых после первичного анализа представляется n действительными значениями $(x_1^{(k)}, \ldots, x_n^{(k)})$. Для каждого такого набора распознаватель обязан установить, что, собственно, было сказано, то есть вычислить некоторое значение $y_k = f(x_1^{(k)}, \ldots, x_n^{(k)})$. Для разных сигналов значения y_k должны быть разными. Вид функции f известен с точностью до параметров $c_1, \ldots, c_s$. Например, функция может быть линейной, то есть иметь вид $f = c_1 \cdot x_1 + \cdots + c_n \cdot x_n + c_{n+1}$. Для того, чтобы распознаватель смог начать работу, его необходимо обучить. В процессе обучения распознавателю подаются на вход тестовые примеры и путем подбора коэффициентов c_i находится функция f, которая в дальнейшем будет использоваться для распознавания. Ситуация осложняется тем, что тестовые примеры никогда не могут быть воспроизведены с абсолютной точностью, одна и та же команда, сказанная два раза одним и тем же человеком, звучит несколько по-разному. Если мы не будем принимать во внимание это обстоятельство, мы всегда будем считать два экземпляра одной и той же команды разными командами.

Реальный распознаватель должен иметь дело не с действительными входными данными, а с интервальными, соответственно в результате первоначального обучения должна быть найдена интервальная функция f. Ее значения также будут интервалами Y_k. Анализ взаимного расположения интервалов Y_p и Y_q позволяет сделать вывод, одинаковые или различные команды были произнесены. Мы, разумеется, предельно упростили постановку задачи в последнем примере, однако в реальных задачах распознавания используется высказанная идея.

Твины, как математические объекты, являются обобщением интервалов. Их естественно использовать в тех случаях, когда необходимо работать с двусторонними оценками интервальных величин. Обсудим два конкретных подхода, при которых может с успехом использоваться твинная техника.

Первый подход связан с применением твинных экспертных оценок, которое лежит на поверхности. Пусть мы имеем дело с некоторой величиной, которая имеет *принципиально интервальную* природу. Например, колебание курса валюты или определенных акций в течение определенного промежутка времени, скажем, календарного месяца. Ясно, что курс будет колебаться, но не ясны ни центральная точка такого колебания, ни его амплитуда. Экспертные знания о таком колебании естественно представить в виде твина. Внешний интервал оценивает колебания снаружи, внутренний – изнутри. Внутренний интервал имеет не меньшее значение, чем внешний. При биржевой игре важно представлять себе не только интервал значений, который гарантированно включает в себя колебание курса, но интервал значений, который курс гарантированно пройдет в течение месяца. Только если внутренний интервал имеет достаточную длину, игра на бирже в течение данного промежутка времени имеет смысл.

Другим примером твинной оценки, которая может быть использована в специализированной экспертной системе, является обработка наблюдений отдельных видов взаимодействий элементарных частиц, наблюдаемых в циклотроне. Предположим, в ходе исследований решается следующая, достаточно характерная, задача – определение физических параметров частиц, при которых возможно интересующее нас взаимодействие. Параметры частиц, как и все физические величины, могут быть определены лишь с некоторой точностью. Пусть интересующее нас взаимодействие наблюдалось k раз и каждый раз измеренные параметры частиц были оценены интервалами. Предположим, что мы обрабатываем для каждого взаимодействия n параметров. Каждое наблюдение характеризуется n интервалами $X_1^{(i)}, \ldots, X_n^{(i)}$, $i = 1, \ldots, k$. Наша задача – описать, при каких значениях параметров возможна интересующая нас реакция. Будем полагать, что если реакция возможна при значениях параметра x' и x'', $x' \leq x''$, то она возможна и при x_0, $x' \leq x_0 \leq x''$. Если количество зафиксированных реакций достаточно велико (мы сейчас отвлекаемся от вопроса статистики распределений параметров частиц), то можно утверждать, что i-й параметр необходимо должен находиться в интервале $X_i = \bigcup\limits_{j=1}^{k} X_i^{(j)} + \varepsilon$. Здесь ε – малый интервал, оцениваемый из статистических соображений. Определим интервал $X_{il} = [\min\limits_{j=1}^{k} X_i^{(j)+}, \max\limits_{j=1}^{k} X_i^{(j)-}]$. Пусть $X_{il} \neq \emptyset$. Можно утверждать, что если i-й параметр находится в интервале X_{il}, то существуют значения других параметров, при которых рассматриваемая реакция возможна. Таким образом мы оценили неизвестную нам физическую величину твином (X_{il}, X_i).

Рассмотрим еще один пример. Известна задача описания геометрических свойств дву- и трехмерного миров. В задачах искусственного интеллекта она используется, в частности, при планировании движений робота. Одной из модификаций задачи является работа с моделями динамического мира, в которых положение всех или части объектов изменяется во времени. Рассмотрим упрощенную двумерную модель, в которой объектами являются квадраты со сторонами, параллельными координатным осям. Квадраты могут перемещаться по плоскости. Рассмотрим один такой квадрат (рис. 4.3).

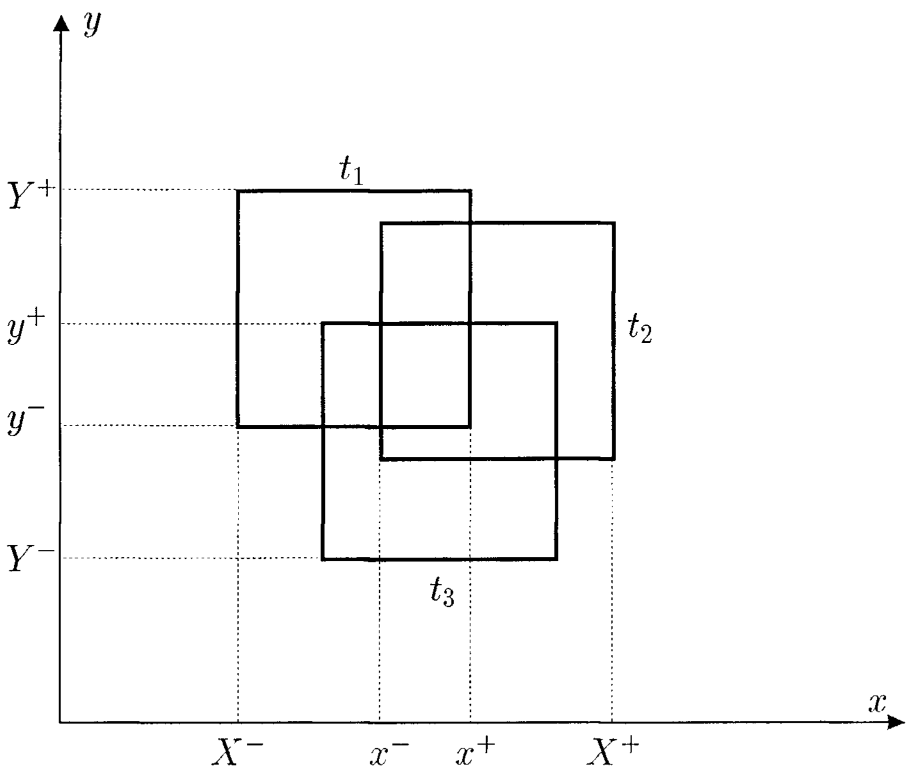

Рис. 4.3.

Предположим, что перемещения квадрата ограничены. Такая система часто встречается на практике, например, если речь идет о перемещении деталей некоторого механизма.

Проекция квадрата на каждую из осей в любой момент времени является отрезком. Для оценки перемещения квадрата за какой-то фиксированный промежуток времени необходимо, во-первых, найти интервал значений координаты, который вмещает в себя все такие отрезки и, во-вторых, найти другой интервал, каждая точка которого в любой момент времени "накрыта" проекцией интервала. Если, скажем, по оси x первый такой интервал обозначить через $X = [X^-, X^+]$, а второй интервал через $X_l = [x^-, x^+]$, то мы получим твинную оценку перемещений объекта. Оценивающим твином будет твин (X_l, X). Имея такие оценки для всех присутствующих объектов, можно, например, планировать движение управляемого объекта. Точки пространства, принадлежащие двумерным интервалам $[x^-, x^+] \times [y^-, y^+]$, являются недостижимыми, а достижимость точек пространства из множества $[X^-, X^+] \times [Y^-, Y^+] \setminus [x^-, x^+] \times [y^-, y^+]$ зависит от времени.

Второй подход менее очевиден. Он связан не с твинным оцениванием интервальных величин, а с разными методами обработки экспертных оценок. Предположим, k экспертов оценивают неизвестную действительную величину. Каждая оценка представляет собой интервал. Естественно, оценки, данные разными экспертами, не совпадают. Обозначим такие оценки через X_i, а саму неизвестную величину через x.

Далее возможно два варианта. Если $\bigcap\limits_{i=1}^{k} X_i = \emptyset$, то очевидно, что по крайней мере некоторые эксперты ошибаются. Мы можем в этом случае говорить лишь о той или иной степени доверия разным оценкам. Если положить $X = [\min\limits_{i=1}^{k} X_i^-, \max\limits_{i=1}^{k} X_i^+]$ и $X_l = [\min\limits_{i=1}^{k} X_i^+, \max\limits_{i=1}^{k} X_i^-]$, то твинная оценка (X_l, X) в большинстве случаев достаточно адекватно представляет неизвестную величину x. Внешний интервал твина при этом является грубой оценкой, которая, однако, выполняется с достаточно высокой вероятностью. Внутренний интервал твина дает более точную оценку, в которой мы уверены в меньшей степени, но которая подходит для использования во многих практических ситуациях.

Второй вариант имеет место, если $\bigcap\limits_{i=1}^{k} X_i \neq \emptyset$. В этой ситуации мы может принять другую семантику и, соответственно, другой путь обработки оценок. Мы можем предположить, что с достаточно высокой вероятностью никто из экспертов не ошибается, и что, кроме того, с абсолютной уверенностью можно утверждать, что хотя бы один эксперт указал корректный интервал оценки. Тогда, обозначив $Y = \bigcup\limits_{i=1}^{k} X_i$ и $Y_l = \bigcap\limits_{i=1}^{k} X_i$, получаем твин (Y_l, Y), оценивающий величину x в соответствии с указанной семантикой: интервал Y дает гарантированную оценку, интервал Y_l дает вероятную оценку. Оценка числового значения вероятности последней – отдельная задача, которую, как нам кажется, уместно решать неформальными методами (путем "оценки вероятностей оценок" разных экспертов) или статистическими методами (путем анализа предыдущих результатов работы с этими же экспертами и аналогичными задачами).

Только что рассмотренный второй вариант допускает ряд модификаций. Первая модификация предусматривает наличие двух экспертов. Первый эксперт дает весьма вероятные оценки, которые, однако, очень широки и не всегда могут быть с успехом использованы, скажем, в ходе логического вывода в рамках экспертной системы. Второй эксперт дает достаточно узкие оценки, вероятность выполнения которых существенно ниже, чем у первого эксперта. Такие две оценки представляют в чистом виде твин, который может послужить для представления знаний в ходе обработки в экспертной системе. Заметим, что аналогичная задача, но для большего числа экспертов и решаемая без применения твинов, рассматривается в работе [138].

Еще одна модификация предусматривает хранение и обработку в качестве единицы знаний о неизвестной величине твина, внешний интервал которого получен строгими методами оценки, не связанными с привлечением экспертов и другими субъективными подходами. Внутренний интервал твина получается в результате обработки экспертных оценок и, в отличие от внешнего, является негарантированным. Выводы, полученные на основе знаний, представленных таким смешанным образом, несут в себе, во-первых, гарантированную оценку, которая, как известно, часто бывает слишком неточной, и, во-вторых, экспертную оценку, которая с некоторой вероятностью может оказаться некорректной, но зато более точно представляет интересующую нас величину.

Итак, в данном параграфе мы убедились, что ряд конкретных задач естественным образом формулируется в терминах твинов. Однако в каждом случае необходимо выбрать твинную арифметику, которая адекватно отражает законы того реального мира, который рассматривается в конкретной задаче. Введенные и рассмотренные в предыдущих главах арифметики предназначены для тех приложений, в которых мы имеем дело с двусторонней оценкой неизвестного интервала. К таким приложениям относятся примеры с атомными реакциями, колебаниями курсов валют, геометрией реального мира. При работе с твинами, возникающими иным путем, каждый раз необходимо модифицировать арифметику с тем, чтобы она удовлетворяла конкретным условиям задачи.

В заключении параграфа отметим, что твины могут быть использованы при решении конкретных задач, например, при реализации реальных экспертных систем, в том случае, если существует адекватная машинная реализация твинной арифметики, служащая базисом для последующих приложений. Одна из возможных реализаций предложена в пятой главе.

4.4 Твины и модальная математика

В последнее время активизировались исследования в области модальной математики. В настоящем параграфе мы кратко напомним основные определения, принимаемые в рамках модального подхода, и укажем на его связь с твинными вычислениями. Установление такой связи позволит в дальнейшем использовать твинные методы, в первую очередь – возможности по их машинной реализации, при решении задач планирования, управления, оптимизации и т. д., то есть задач, для которых модальный подход и создается. С другой стороны, результаты, полученные в рамках модального подхода, например, результаты о сложности модальных вычислений, могут быть использованы в твинных вычислениях.

Модальный подход (иногда его еще называют модальной математикой, хотя нам такой термин представляется слишком общим) берет свое начало в работах испанских исследователей и восходит к работе [60]. Введение в него также можно найти в Добавлении С книги [101], которое называется "От интервальных вычислений к модальной математике". В этой же работе доказываются интересные результаты, касающиеся сложности модальных вычислений.

Во многих вычислительных задачах мы сталкиваемся с необходимостью либо найти значение, которое *должно* отвечать некоторому свойству либо *может* отвечать некоторому свойству. Например, имея дело с системой управления, мы ищем значения управляемых параметров, при которых система *должна* сохранять стабильность. С другой стороны, имея дело с наблюдением сложной физической системы, мы задаемся целью найти множество значений, которые *может* принимать интересующий нас параметр.

Переходя на язык твинов и возвращаясь к задаче двустороннего оценивания неизвестного интервала твином, мы замечаем, что любая точка внутреннего интервала твина *должна* принадлежать неизвестному интервалу, а любая точка внешнего интервала твина *может* принадлежать оцениваемому интервалу.

Модальный подход, который мы излагаем, в частности, базируется на понятиях модальной логики (см., например, [115], [144]). Оттуда же заимствованы обозначения: высказываение "A может иметь место" обозначается через $\diamond A$, а высказывание "A должно иметь место" – через $\Box A$.

В рамках модального подхода рассматриваются формулы, являющиеся обобщением обычных формул логики первого порядка. Опишем такие формулы. Формулы языка модальной логики строятся из вещественных констант, переменных, обозначающих вещественные числа, специальных модальных переменных, арифметических и логических операций, знаков отношений $=, \leq, <, \geq, >$, логических кванторов. Обычные переменные мы будем обозначать строчными латинскими буквами $(x, y, \ldots)$, а модальные переменные – прописными рукописными латинскими буквами $(\mathcal{X}, \mathcal{Y}, \ldots)$.

Определение 4.1. Термом назовем переменную (в том числе модальную), константу или конструкцию вида (t), $t \circ t'$ или $\star t$. Здесь $\circ$ и $\star$ –

соответственно любая бинарная и унарная арифметические операции, а t и t' – термы.

Элементарной формулой называется конструкция $t \circ t'$, где t и t' – термы, а $\circ$ – знак арифметического отношения.

Определение 4.2. Формулой назовем элементарную формулу, конструкции вида (F), $F \circ F'$, $\star F$ (здесь $\circ$ и $\star$ – соответственно любая бинарная и унарная логические операции), конструкции вида $\forall x_i F$, $\exists x_i F$, $\diamond F$ и $\Box F$. Через F и F' обозначены формулы, через x_i – простая переменная.

Формула называется бескванторной, если она не содержит кванторов, и безмодальной, если она не содержит знаков $\diamond$ и $\Box$. Формула называется замкнутой, если каждая простая переменная x_i находится в области действия соответствующего квантора, и каждая модальная переменная $\mathcal{X}_i$ находится в области действия некоторого символа модальности. Нас в дальнейшем будет интересовать работа с замкнутыми формулами.

Для определения значения формулы F необходимо зафиксировать множество $\Omega \subseteq R^n$, где n – общее количество модальных переменных в формуле. Значение истинности замкнутой формулы определяется следующим образом: вещественные значения термов и логические значения формул, не содержащих модальные переменные, определяются в соответствии с очевидными правилами отношений, арифметических и логических операций и правил действий с кванторами. Формула $\diamond F$ верна тогда и только тогда, когда существует набор $(\mathcal{X}_1, \ldots, \mathcal{X}_n) \in \Omega$, для которого F истинна. Формула $\Box F$ верна тогда и только тогда, когда для любого набора $(\mathcal{X}_1, \ldots, \mathcal{X}_n) \in \Omega$ F истинна.

Определение 4.3. В модальной математике замкнутая формула является верной, если ее значение равно истине для любого множества $\Omega \subseteq R^n$.

Основной задачей, рассматриваемой в рамках модального подхода, является задача проверки, являтся ли некоторая формула верной.

Многие известные задачи допускают переформулировку в терминах модальной математики. Например, основная задача интервального анализа – задача оценки множества значений функции может быть записана следующим образом [101]:

$$(\Box(x_1^- \le \mathcal{X}_1 \le x_1^+) \& \cdots \& \Box(x_n^- \le \mathcal{X}_n \le x_n^+)) \Rightarrow \\ \Box(y^- \le f(\mathcal{X}_1, \ldots, \mathcal{X}_n) \le y^+).$$

Здесь $x_1^-, \ldots, x_n^-, x_1^+, \ldots, x_n^+, y^-, y^+$ – некоторые константы. Если мы имеем возможность проверить, верна ли эта формула для разных значений y^- и y^+, то можно оценить множество значений функции с любой необходимой точностью. Для этого следует применить алгоритм поиска подходящих y^- и y^+, основанный, например, на половинном делении.

Задача внутренней оценки множества значений функции также может быть записана в виде формулы модальной математики:

$$(\Box(z^- \le \mathcal{Z} \le z^+) \ \& \ \forall t_1(x_1^- \le t_1 \le x_1^+ \Rightarrow \Diamond\mathcal{X}_1 = t_1) \cdots \\ \& \ \forall t_n(x_n^- \le t_n \le x_n^+ \Rightarrow \Diamond\mathcal{X}_n = t_n)) \Rightarrow \\ \Diamond f(\mathcal{X}_1, \ldots, \mathcal{X}_n) = \mathcal{Z}.$$

В работе [101] доказано, что установление истинности бескванторной формулы с модальностями является NP-трудной задачей, что соответствует рассмотренной ранее сложности интервальных вычислений.

В четвертой главе нами были рассмотрены принципы применения твинных вычислений для решения конкретных задач. Был не только решен ряд задач, но и сформулированы методологические основы применения твинных вычислений. Все внимание было уделено теоретическим аспектам приложений. Вопросы компьютерной реализации будут рассмотрены нами ниже.

Глава 5. Реализация твинных арифметик

5.1 Язык высокого уровня для интервальных и твинных вычислений

Во введении, разделах 1.6.3 и 2.7 мы уже кратко коснулись вопросов реализации интервальных и твинных вычислений на ЭВМ. В настоящей главе мы будем говорить о конкретной реализации твинных арифметик в среде языка Pascal-XSC.

Уже отмечалось, что существуют разнообразные программные средства для работы с интервалами. Сюда относятся и программные пакеты, и библиотеки для интервальных вычислений. Наиболее удобными для пользователя являются языки высокого уровня, позволяющие непосредственно программировать интервальные и твинные алгоритмы. Отметим преимущества и недостатки таких языков. Видимо, будет уместно вспомнить преимущества и недостатки обычных, неинтервальных языков высокого уровня по сравнению с низкоуровневыми языками типа ассемблера. Языки высокого уровня более удобны для пользователя, на них быстрее и удобнее писать и отлаживать программмы. Языки низкого уровня не дают пользователю развитых средств, однако написанные на них программы более эффективны.

Если говорить о специализированных интервальных программных средствах, то прослеживается полная аналогия: пакеты и библиотеки программ позволяют создать весьма эффективные программные модули, однако работа с ними требует от программиста большого опыта и затрат труда. Интервальные языки высокого уровня просты в обращении, программы, написанные на них, легко читаются, однако не всегда эффективно выполняются. Отметим также, что библиотеки программ, как правило, ориентированы на какой-то определенный класс приложений, в то время как языки высокого уровня универсальны.

Сказанное позволяет сделать вывод о возможных путях использования интервальных и твинных языков: они полезны при программировании алгоритмов с относительно невысокой вычислительной сложностью, для алгоритмов, в которых объем содержательного анализа данных и логических преобразований превалирует над чисто счетной работой. Они также незаменимы при необходимости стыковки с уже написанными программами, возможно, неинтервальной природы.

Интервальные языки высокого уровня незаменимы при обучении. Они, как правило, содержат в чистом виде средства, критические в отношении правильного кодирования численных алгоритмов. Обучаемый не должен тратить усилия на техническое оформление таких базовых действий, как направленное округление или получение скалярного произведения с максимально возможной точностью. Все эти средства уже присутствуют в языке, ими можно пользоваться как базовыми примитивами. Начинающий пользователь, таким образом, освобождается от необходимости создания своей собственной среды примитивов и может уделять все свое внимание самим программируемым алгоритмам. Если говорить конкретно о языке Pascal-XSC, то уместно вспомнить, что язык Паскаль, лежащий в его основе, в свое время был создан Виртом именно как язык, специально предназначенный для целей обучения [8].

Даже если интервальный язык высокого уровня оказывается недостаточно эффективным при программировании численных алгоритмов, он с успехом может быть использован и используется для прототипирования. Отладка больших программных систем на нем весьма удобна. В ходе последующей реализации окончательного варианта программной системы критические с точки зрения эффективности части системы могут быть переписаны, например, на языке ассемблера или на Си, который, как мы увидим, лежит в основе Pascal-XSC.

5.2 Язык Pascal-XSC и его окружение

5.2.1 Общая информация

Язык Pascal-XSC [10] является расширением стандартного Паскаля [8]. Он был разработан в Институте прикладной информатики г. Карлсруэ под руководством проф. У. Кулиша. Помимо языка Pascal-XSC были созданы и другие аналогичные языки на базе Фортрана, Си, Модулы. Они получили названия Fortran-XSC, C-XSC, Modula-SC. Наиболее известным и используемым языком является Pascal-XSC. Для него сделано наибольшее число реализаций, он чаще других применяется для решения практических задач.

Созданию языка и его реализации предшествовали интенсивные исследования в области компьютерной арифметики. Результаты исследований были опубликованы во многих работах, основной из которых является работа [103]. В ходе исследований стало ясно, что язык, предназначенный для программирования численных расчетов с гарантированной точностью, должен обладать рядом свойств. Например, он должен поддерживать векторную обработку, направленные округления, точное векторное произведение и т. д. Все эти и некоторые другие средства в той или иной степени были реализованы в языке, но не все они в равной степени нам интересны. Нас в основном будут интересовать те средства языка и те особенности его реализации, которые соотносятся с рассматриваемыми в настоящей работе задачами. Сюда относится, главным образом, реализованная арифметика и все, что с ней связано. Соответственно, более подробно мы рассмотрим именно эти особенности языка Pascal-XSC. Ниже они будут применены в реализации твинной арифметики. Параграф 5.1.2 будет посвящен особенностям языка, имеющим общий характер, параграф 5.1.3 – тем конкретным средствам языка, которые наиболее интересны в контексте твинной арифметики.

Язык Pascal-XSC был реализован на многих платформах, начиная от персональных компьютеров и кончая суперкомпьютерами. Для реализации была выбрана следующая схема. Исходный текст программы на языке Pascal-XSC подается на вход прекомпилятору, переводящему его в программу на языке Си. Сам прекомпилятор также написан на Си. Полученная программа далее транслируется каким-либо из многочисленных имеющихся трансляторов с Си в объектный код конкретной машины.

Благодаря такой технологии компиляции исходных текстов, программы, написанные на Pascal-XSC, могут легко переноситься практически на любую платформу, лишь бы на ней можно было компилировать и выполнять программы, написанные на Си. Конечно, такая двухступенчатая технология работы с программами на Pascal-XSC менее эффективна, чем прямая компиляция исходного текста в машинный код, но зато разработка транслятора не требует чрезмерных ресурсов и программы легко переносимы.

Популярности языка среди специалистов по вычислительной математике немало способствовал тот факт, что на Pascal-XSC были разработаны прикладные программные модули для решения характерных численных задач [69]. Имеются библиотеки, ориентированные на решение систем линейных уравнений, обращение матриц, решение систем нелинейных уравнений, нахождение собственных чисел и собственных векторов, нахождение корней полинома, численное интегрирование, автоматическое дифференцирование, решение задачи Коши и краевых задач для обыкновенных дифференциальных уравнений.

5.2.2 Языковые особенности

В языке Pascal-XSC допустимо определять функции с результатом произвольного типа, а не только с результатом простого типа, как это сделано в стандартном Паскале. Например, результатом функции может быть массив. Если у нас есть описание

```
type polynomial = array [0..max_deg] of real;
```

то можно определить функцию сложения полиномов следующим образом:

```
function add (a, b: polynomial): polynomial;
  var i: integer;
  begin
    for i:=0 to max_deg do
      add[i]:=a[i]+b[i];
  end;
```

Здесь и далее часть используемых примеров получена модификацией фрагментов программ из работы [10].

Создатели языка пошли дальше. Они дали пользователю возможность определять свои операции и даже переопределять имеющиеся операции. Например, для переменных типа `polynomial` сложение можно определить и так:

```
operator + (a, b: polynomial) res_poly: polynomial;
  var i: integer;
  begin
    for i:=0 to max_deg do
```

```
        res_poly[i]:=a[i]+b[i];
    end;
```

Pascal-XSC дает пользователю возможность использовать одно и то же имя процедуры или функции для обозначения разных процедур/функций. Например, можно определить функцию `arcsin` для аргумента типа `real` и для аргумента типа `interval`. Для этого необходимо дважды описать функцию для разных значений аргумента. Какое определение применить в каждом случае – решает транслятор, исходя из типов и количества аргументов.

Знаки операций также могут совмещать несколько определений операций. Даже операция присваивания $:=$ может быть доопределена пользователем для присваивания значений различных типов.

Пользователю предоставляется возможность делить программу на модули. Каждый модуль компилируется отдельно. Некоторые константы, типы, переменные, процедуры и т. д., описанные в модуле, объявляются глобальными и могут быть использованы в других модулях. В виде модуля обычно оформляется библиотека процедур, функций и операций, связанных между собой по назначению и по типам обрабатываемых данных. Библиотеки, работающие с твинами, будут оформлены нами в виде модулей.

Из других средств, присутствующих в Pascal-XSC и отсутствующих в стандартном Паскале, отметим динамические массивы и строки переменной длины.

5.2.3 Специальные средства

Остановимся на средствах, играющих ключевую роль при построении алгоритмов с гарантированной точностью результатов. Именно эти средства более других будут нас интересовать при построении твинных библиотек.

Язык Pascal-XSC обеспечивает направленные округления. Для этого каждая из четырех арифметических операций `+`, `-`, `*`, `/` имеет три варианта употребления. Если написать просто `a/b`, то при необходимости результат операции будет округлен к ближайшему представимому на данном компьютере числу. Если мы хотим округлить результат операции к ближайшему непревосходящему сам результат машинному числу, то в качестве знака операции необходимо употребить одно из следующих зарезервированных сочетаний: `+<`, `-<`, `*<`, `/<`. Аналогично округления вправо можно добиться применением операций `+>`, `->`, `*>`, `/>`.

Округления, естественно, производятся не только при выполнении арифметических операций. Если мы используем в программе числовой литерал, то должны понимать, что при его переводе в форму, представимую в машине, он будет преобразован, и результат такого преобразования не всегда может быть сделан равным самому числу. Если записать константу обычным образом, то она будет округлена до ближайшего машинного числа. Если есть необходимость округлить константу в сторону уменьшения или увеличения, то следует использовать, соответственно, форму `(< 'число')` или `(> 'число')`. В качестве `'число'` обозначено произвольное число, записанное по правилам стандартного Паскаля. Например, возможны следующие литералы для чисел типа `real`:

```
(< 5.856)
(> -165.5E-11)
```

Еще одна конструкция языка, в которой производится округление чисел, – ввод/вывод. Для того, чтобы указать машине на необходимость выполнения направленного округления при чтении и записи вещественных чисел, существует специальный формат для операторов ввода/вывода. В операторах `read`, `write`, `readln`, `writeln` вслед за именем переменной необходимо после двоеточия указать отрицательное, положительное число или 0. Если указанное число отрицательное, то производится округление влево, если положительное – то вправо, если 0, или число вовсе не указано – то округление производится к ближайшему представимому числу. Например,

```
read(x: -1, y: 0);
writeln(a, b: 1, c: -1);
```

В языке существует мощный механизм так называемых высокоточных выражений (#-выражений). Для их реализации в язык введен специальный тип `dotprecision`, представляющий собой формат с фиксированной точкой, позволяющий представлять весь диапазон произведений чисел с плавающей точкой. Он в основном предназначается для хранения скалярных произведений, то есть сумм произведений чисел с плавающей точкой. При помещении результата вычисления скалярного произведения в переменную типа `dotprecision` никакого округления не производится.

В простейшей своей форме высокоточные выражения используются для абсолютно точного вычисления выражений, математически эквивалентных скалярным произведениям. Синтаксически они выглядят следующим образом:

```
# (‘выражение’);
#* (‘выражение’);
#< (‘выражение’);
#> (‘выражение’);
```

где `'выражение'` – выражение, математически эквивалентное скалярному произведению. В первом случае результат вычисления выражения имеет тип `dotprecision` и равен абсолютно точному его значению безо всяких округлений. В трех других случаях результат имеет тип `real` и равен тому же значению, но округленному, соответственно, до ближайшего, ближайшего с недостатком и ближайшего с избытком числа типа `real`. Например, при описаниях

```
var a,b: array [1..10] of real;
    x,y: real;
    d:   dotprecision;
    i:   integer;
```

возможен следующий фрагмент программы:

```
d:=#(0);
for i:=1 to 10 do d:=#(d+a[i]*b[i]);
x:=#*(d);
y:=#>(a*b);
```

После его выполнения в переменной `x` окажется скалярное произведение векторов `a` и `b`, округленное к ближайшему машинному числу, а в переменной `y` то же значение, округленное до ближайшего машинного числа, не меньшего точного результата. Заметим, что в обоих случаях округление осуществляется лишь один раз и применяется к конечному результату.

Аппарат высокоточных выражений в языке Pascal-XSC представляет собой весьма мощное средство, особенно эффективное при действиях с векторами и матрицами.

Язык, как это делается и в стандартном Паскале, предусматривает вычисление стандартных функций, таких как `sin`, `tan`, `sqrt`, `log10` и т. п. Разработчики языка гарантируют, что округление результата всегда происходит к ближайшему машинному числу. К сожалению, не предусмотрено направленных округлений для элементарных функций. При реализации интервальных алгоритмов часто встречается ситуация, когда необходимо гарантировать, что округление произведено с избытком (с недостатком), пусть даже не к ближайшему машинному числу. В этом случае необходимо к вычисленному результату элементарной функции применить одну из операций `pred` или `suc`, которые вычисляют предыдущее и последующее машинные числа для своего аргумента. Заметим, что в стандартном Паскале операции `pred` и `suc` не применяются к значениям типа `real`.

Язык предоставляет пользователю все необходимые средства для работы с интервалами. В нем предусмотрен предопределенный тип `interval`, объявленный следующим образом:

```
type interval = record
  inf, sup: real;
end;
```

Строго говоря, в состав языкового ядра примитивы для работы с типом `interval` не входят. Они собраны в отдельном модуле `I_ARI`, который необходимо подключать, если есть необходимость использовать входящие в его состав операции, функции и процедуры.

Для интервальных значений предусмотрены арифметические операции, округляющие результат до наименьшего представимого объемлющего интервала. Операции отношения для двух значений типа `interval` трактуются в теоретико-множественном смысле. Реализованы следующие операции: `=` (равно), `<>` (не равно), `<` (строгое подмножество), `<=` (подмножество), `>` (строгое надмножество), `>=` (надмножество). Операция `in` реализована в двух вариантах. Если оба ее операнда – интервалы, то первый должен быть строгим подмножеством второго (эквивалент <), а если первый операнд – вещественное число, а второй – интервал, то число должно принадлежать интервалу. Операция >< проверяет интервалы на непересечение, а операции `+*` и `**` вычисляют, соответственно, интервальную оболочку и пересечение двух интервалов.

Для преобразования между типами `real` и `interval` используются следующие функции:

`intval(r1, r2)`	Интервал с нижней границей `r1` и верхней границей `r2`.
`intval(r)`	Интервал с обеими границами, равными `r`.
`inf(i)`	Нижняя граница интервала `i`.
`sup(i)`	Верхняя граница интервала `i`.

Присваивание интервальной переменной `i` может быть осуществлено как оператором `i:=r`, так и `i:=j`, где `r` – переменная типа `real`, а `j` – переменная типа `interval`.

Все вещественные функции, существующие для вещественных аргументов, реализованы и для интервальных аргументов. При этом результирующий интервал включает в себя точное значение интервальной функции, которое, возможно, не может быть непосредственно представлено на машине.

Для ввода-вывода интервальных значений в модуле `I_ARI` реализованы процедуры

```
procedure read (var f: text; var a: interval);
procedure write (var f: text; a: interval);
```

Первый параметр необязательный, количество вводимых или выводимых значений может быть произвольным. Существуют варианты процедур `readln` и `writeln`.

При вводе значения интервального типа оно должно присутствовать во входном потоке в виде [x,y] или в виде x. В первом случае вводится наиболее узкий интервал, объемлющий [x,y], представимый на машине. Во втором случае производятся те же действия, что и при вводе интервала [x,x]. При выводе интервала он печатается в форме [x,y], при этом он округляется в сторону расширения.

В состав языка входят модули CI_ARI для комплексной интервальной арифметики, MVI_ARI для интервальных векторно-матричных преобразований, MVCI_ARI для комплексных интервальных векторно-матричных преобразований. Эти модули расширяют средства интервальной арифметики, предоставляемые в модуле I_ARI.

5.3 Расширение языка Pascal-XSC

Над ядром языка Pascal-XSC надстроены модули, некоторые из которых мы перечислили выше. Автором настоящей работы реализованы следующие модули:

- II1 – для работы с бесконечными внутренними интервалами (см. раздел 1.6.1.1);
- II2 – для работы с интервалами Кахана (см. раздел 1.6.1.2);
- IID – для работы с направленными интервалами (см. раздел 1.6.2.2);
- T_ARI – для работы с твинами;
- TD_ARI – для работы с направленными твинами.

Первые три модуля не являются предметом рассмотрения настоящей работы. Информацию о них можно найти в работах [135], [127].

Четвертый и пятый модули непосредственно являются реализацией твинных арифметик. Их описанию посвящены следующие два раздела.

Общая структура модулей изображена на рис. 5.1. Модули, входящие в стандартный комплект языка Pascal-XSC, обведены пунктирной

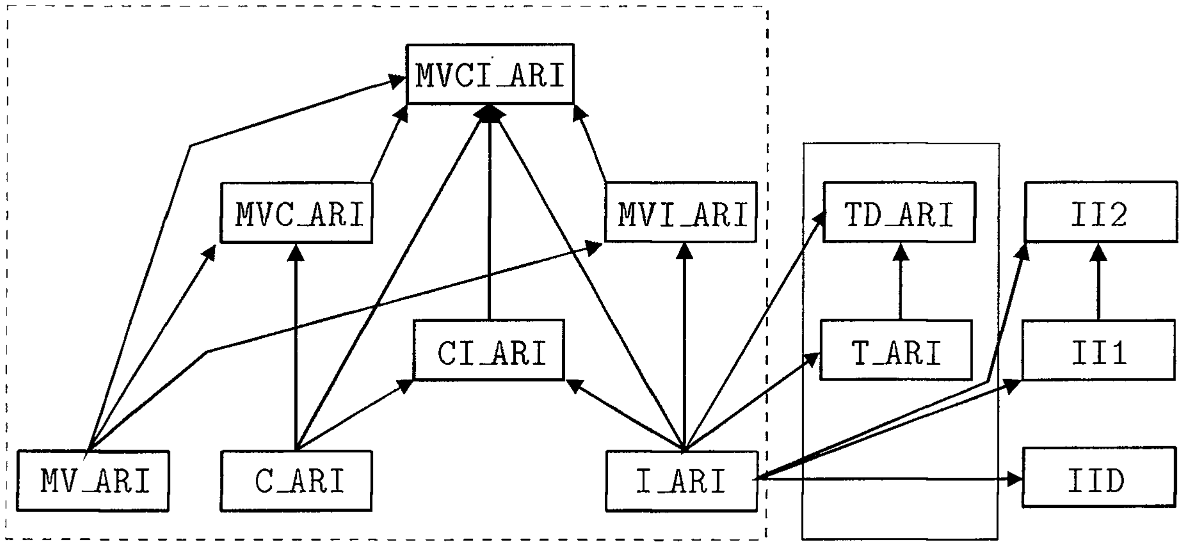

Рис. 5.1.

чертой, а модули, которые реализованы автором настоящей работы, – сплошной чертой. Если два модуля соединены стрелкой, то это означает, что модуль, находящийся в основании стрелки, используется при определении модуля, на который стрелка указывает.

5.4 Модуль для работы с твинами

Модуль, определяющий типы, операции, процедуры и функции для работы с твинами, называется T_ARI. Арифметика, которую он реализует, полностью соответствует той, которая описана в разделах 2.2.3 и 2.7. Модуль T_ARI использует стандартный модуль I_ARI. Типы, операции, функции и процедуры, реализованные в модуле T_ARI, приведены в Приложении 1.

В качестве основного типа данных, переменные и константы которого предствляют твины, описан тип `twin`.

```
type twin = record
  tp:  boolean;
  int: interval;
  out: interval;
end;
```

Логическая переменная `tp` несет в себе информацию, является ли твин вырожденным. Если `tp=true`, то твин невырожденный и интервалы `int` и `out` задают, соответственно, внутренний и внешний интервалы твина. Если `tp=false`, то твин является вырожденным, значение `out` задает его внешний интервал, а значение `int` игнорируется.

Для формирования переменных типа `twin` используется функция `twinwal`, доступная в трех вариантах. Пусть действуют описания

```
var r1, r2, r3, r4: real;
    i1, i2: interval;
    t: twin;
```

Тогда после выполнения

```
t:=twinval(i1,i2);
```

твин `t` получает значение с внутренним интервалом, равным `i1`, и внешним интервалом, равным `i2`. Если интервал `i1` не является подмножеством интервала `i2`, то `t` получает значения вырожденного твина с пустым внутренним интервалом и внешним интервалом, равным `i2`.

Другой вариант формирования твинного значения при помощи функции `twinwal` выглядит следующим образом:

```
t:=twinval(r1,r2,r3,r4);
```

Предполагается, что r1≤r2 и r3≤r4, иначе фиксируется ошибка. Действия функции такие же, как и в предыдущем варианте, если i1=[r1,r2], i2=[r3,r4].

Функция twinval допускает следующие две формы вызова:

```
t:=twinval(i1);
t:=twinval(r1);
```

В первом случае переменной t присваивается твин, у которого внутренний интервал равен внешнему и равен i1, а во втором случае, кроме того, оба интервала имеют верхние границы, совпадающие с нижними и равные r1.

Аналогичного действия можно добиться, воспользовавшись совмещенной операцией присваивания, записав, соответственно,

```
t:=i1;
t:=r1;
```

Логическая функция degtwin с единственным аргументом типа twin возвращает значение true только для вырожденных твинов с пустым внутренним интервалом.

Для доступа к полям записи типа twin можно воспользоваться обычными средствами Паскаля и написать, например, t.int, t.out, t.int.sup и т. д. Для сохранения единообразия с модулем I_ARI определены шесть функций int(t), ext(t), intinf(t), intsup(t), extinf(t), extsup(t). Первые две своими значениями имеют, соответственно, внутренний и внешний интервалы твина, а остальные четыре, соответственно, нижнюю границу внутреннего интервала, верхнюю границу внутреннего интервала, нижнюю границу внешнего интервала и верхнюю границу внешнего интервала.

Для ввода и вывода твинных данных предусмотрены варианты процедур `read` и `write` с заголовками

```
procedure read (var f: text; var a: twin);
procedure write (var f: text; a: twin);
```

Первый параметр-файл может быть опущен, количество параметров ввода или вывода может быть произвольным. В файле или входном потоке элемент данных должен находиться в одной из следующих форм:

Элемент данных	Вводимое значение
`([a,b],[c,d])`	$([a,b],[c,d])$
`(empty,[c,d])`	$(\emptyset,[c,d])$
`(empty,c)`	$(\emptyset,[c,c])$
`(a,[c,d])`	$([a,a],[c,d])$
`(a,a)`	$([a,a],[a,a])$
`[a,b]`	$([a,b],[a,b])$
`a`	$([a,a],[a,a])$

Таб. 5.1.

В Таблице 5.1 `a`, `b`, `c`, `d` обозначают вещественные числа, записанные в соответствии со стандартным Паскалем, а `empty` – специальное слово, кодирующее во входном потоке пустое множество.

Если вводимые числа не могут быть точно представлены машинными числами, то они подвергаются направленному округлению, причем внутренний интервал твина округляется в сторону сужения, а внешний интервал – в сторону расширения.

Вывод интервального значения осуществляется в одной из двух форм: `([a,b],[c,d])` или `(empty,[c,d])`. При переводе чисел из двоичного машинного представления в десятичное внешнее представление при необходимости производятся округления. Естественно, внешний интервал округляется в сторону расширения, а внутренний – в сторону сужения.

В модуле T_ARI предусмотрены унарные операции + и - и бинарные операции +, -, *, /, заданные в соответствии с определениями из раздела 2.2.3. При выполнении бинарных операций производится округление согласно общему правилу: внешний интервал твина расширяется, внутренний интервал твина сужается. В бинарных арифметических операциях с твинами допускается в качестве одного из операндов использовать интервал или вещественное число. При этом автоматически будет произведено необходимое преобразование: интервал i будет преобразован к твину (i,i), а значение r типа real – к твину ([r,r],[r,r]). Как видно из Приложения 1, для реализации всего разнообразия сочетаний твин-интервал, твин-число, интервал-твин и т.п. определено соответствующее количество операций с разными типами аргументов.

Операции отношения между твинами задаются следующим образом. Отношения = (равно) и <> (не равно) определяются очевидно: для равенства двух твинов необходимо попарное равенство всех их компонент. Отношения <, <=, >, >= имеют теоретико-множественный смысл и реализуют определение, заданное формулой 2.12. Для того, чтобы твин t1 был подмножеством твина t2, необходимо, чтобы внешний интервал t1 был подмножеством внешнего интервала t2 и внутренний интервал t2 был подмножеством внутреннего интервала t1. Как и в случае арифметических операций, в операциях отношения допускается использовать интервалы и вещественные числа в позициях твинов. Преобразование интервалов и чисел к типу twin производится так же, как и для арифметических операций.

Отношение между интервалом и твином, обозначаемое знаком $\sqsubseteq$ и заданное в разделе 2.2.3, реализуется операцией `in` с заголовком описания

```
operator in (i: interval; t: twin) r: boolean;
```

Существует вариант операции с первым аргументом типа `real`, который будет преобразован к типу `interval` в соответствии с общим правилом.

В модуле T_ARI определено большое количество математических функций, хотя их количество несколько меньше, чем в модуле I_ARI. Полный список функций приведен в Приложении 1. Все эти функции одного аргумента. Они возвращают в качестве значения твин, оценивающий множество значений функции снаружи и изнутри при известной оценке аргумента функции. При вычислении производятся направленные округления согласно общим правилам.

5.5 Модуль для работы с направленными твинами

В модуле TD_ARI определены типы, операции, процедуры и функции, позволяющие работать с направленными твинами. Арифметика, реализованная модулем, теоретически описана разделе 2.4.1. Модуль TD_ARI использует стандартный модуль I_ARI и модуль T_ARI, описанный в предыдущем параграфе. Описания типов, заголовки описаний операций, функций и процедур, реализованных в модуле TD_ARI, содержатся в Приложении 2.

В качестве основного типа данных, переменные и константы которого представляют направленные твины, описан тип `dtwin`. Кроме того, описан вспомогательный тип `mon`, значения которого представляют вид монотонности функций, т. е. третий элемент направленных твинов.

```
type mon=(-1,0,1);
type dtwin = record
  tp:  boolean;
  dir: mon;
  int: interval;
  out: interval;
end;
```

Тип dtwin повторяет тип twin с той лишь разницей, что в нем введено дополнительное поле dir.

Для задания переменных типа dtwin используется функция dtwinval, для которой определено несколько вариантов. Пусть действуют описания

```
var r1, r2, r3, r4: real;
    i1, i2: interval;
    m: mon;
    t: twin;
    d: twin;
```

Тогда после выполнения

```
d:=dtwinval(i1,i2,m);
```

направленный твин d получает значение с внутренним интервалом, равным i1, внешним интервалом, равным i2, и направлением, задаваемым m. Если интервал i1 не является подмножеством интервала i2, то d получает значение вырожденного твина с пустым внутренним интервалом и внешним интервалом, равным i2.

Другой вариант формирования твинного значения при помощи функции dtwinwal выглядит следующим образом:

```
d:=dtwinval(r1,r2,r3,r4,m);
```

Предполагается, что r1≤r2 и r3≤r4, иначе фиксируется ошибка. Действия функции такие же, как и в предыдущем варианте, если i1=[r1,r2], i2=[r3,r4].

Функция `dtwinval` допускает следующие формы вызова:

```
d:=dtwinval(t1,m);
d:=dtwinval(t1);
d:=dtwinval(i1,i2);
d:=dtwinval(r1,r2,r3,r4);
d:=dtwinval(i1,m);
d:=dtwinval(i1);
d:=dtwinval(r1,m);
d:=dtwinval(r1);
```

Отсутствие направления у твина эквивалентно заданию нулевого направления. В варианте 1 происходит формирование направленного твина из твина и отдельно заданного направления. В вариантах 5-8 прежде всего происходит преобразование значения типа `real` или типа `interval` к типу `twin`. Это преобразование происходит по правилам, аналогичным правилам, применяемым в модуле, оперирующем обычными твинами.

Допустимы операции присваивания следующего вида:

```
d:=t;
d:=i1;
d:=r1;
```

Логическая функция `degtwin` с единственным аргументом типа `dtwin`, как и в случае твинного аргумента, возвращает значение `true` только для вырожденных направленных твинов с пустым внутренним интервалом.

Для доступа к полям записи типа `dtwin` можно воспользоваться обычными средствами Паскаля и написать, например, `d.m`, `t.int.sup` и т. д. Также определены семь функций `int(d)`, `ext(d)`, `intinf(d)`, `intsup(d)`, `extinf(d)`, `extsup(d)`, `direction(d)`, имеющие своими значениями интервалы, входящие в состав направленного твина, границы этих интервалов и направление твина.

Ввод и вывод направленных твинов производятся процедурами `read` и `write` с заголовками

```
procedure read (var f: text; var a: dtwin);
procedure write (var f: text; a: dtwin);
```

Первый параметр-файл может быть опущен, количество параметров ввода или вывода может быть произвольным. В файле или входном потоке элемент данных должен находиться в одной из следующих форм:

Элемент данных	Вводимое значение
`([a,b],[c,d],m)`	$([a,b],[c,d],m)$
`([a,b],[c,d])`	$([a,b],[c,d],0)$
`(empty,[c,d],m)`	$(\emptyset,[c,d],m)$
`(empty,[c,d])`	$(\emptyset,[c,d],0)$
`(empty,c,m)`	$(\emptyset,[c,c],m)$
`(empty,c)`	$(\emptyset,[c,c],0)$
`(a,[c,d],m)`	$([a,a],[c,d],m)$
`(a,[c,d])`	$([a,a],[c,d],0)$
`(a,a,m)`	$([a,a],[a,a],m)$
`(a,a)`	$([a,a],[a,a],0)$
`[a,b]`	$([a,b],[a,b],0)$
`a`	$([a,a],[a,a],0)$

Таб. 5.2.

В Таблице 5.2 a, b, c, d, empty имеют тот же смысл, что и в Таблице 5.1, а m обозначает одно из чисел -1, 0, 1.

При вводе и выводе при необходимости производятся округления, аналогичные тем, что описаны в предыдущем разделе для обычных твинов.

В модуле TD_ARI предусмотрены унарные операции + и - и бинарные операции +, -, *, /, заданные в соответствии с определениями из раздела 2.4.1. При выполнении бинарных операций производится округление согласно обычному правилу: внешний интервал твина расширяется, внутренний интервал твина сужается. В бинарных арифметических операциях с направленными твинами в качестве одного из операндов может быть использовано вещественное число, твин или интервал. При этом число, твин или интервал автоматически будут приведены к типу dtwin.

Между направленными твинами заданы операции отношения точно также, как и в случае простых твинов. Для выполнения всех отношений, кроме <>, направления твинов должны совпадать, а для элементов int и out должны выполняться такие же отношения, как и в случае обычных твинов. Отношение <> выполняется, если хотя бы один элемент у двух направленных твинов не совпадает. Преобразование интервалов и чисел к типу dtwin при необходимости производится так же, как и для арифметических операций. Для направленных твинов введено дополнительное отношение <<=, соответствующее определению, даваемому формулой 2.31.

В модуле TD_ARI определены математические функции одного аргумента, список которых можно найти в Приложении 2. В качестве значения они возвращают направленный твин, оценивающий множество значений функции снаружи и изнутри при известной твинной оценке аргумента функции. Направление результирующего твина определяется характером монотонности функции на внешнем интервале твина-аргумента. При вычислении производятся направленные округления согласно общим правилам.

В качестве простейшего примера приведем запись программы на предлагаемом расширении языка Pascal-XSC. Программа проводит вычисления, ранее описанные в Примерах 2.6 и 2.7. Назначение программы – проиллюстрировать простоту записи алгоритмов на языке Pascal-XSC, ее близость к естественной математической нотации.

```
program test;
use I_ARI, T_ARI, TD_ARI;
var x,y,z: twin;
    p,s: dtwin;
begin
  x:=twinval(1.4,1.5);
  y:=x;
  p:=dtwinval(x,1);
  z:=(10*exp(y)-x)*(tan(y)-sqr(x));
  s:=(10*exp(p)-p)*(tan(p)-sqr(p));
  writeln(z);
  writeln(s);
end.
```

В пятой главе были разработаны принципы организации инструментальной программной системы для программирования твинных вычислений и описан созданный автором пакет, расширяющий язык Pascal-XSC. Предлагаемый пакет может быть использован для программирования рассмотренных в предыдущих главах твинных алгоритмов и для решения любых практических задач посредством твинных вычислений.

Заключение

В диссертации разработаны теоретические основы твинных вычислений, которые в целом представляют собой дальнейшее развитие интервальных методов в вычислительной математике. Теория твинных вычислений служит для создания новых алгоритмов, решающих актуальные вычислительные задачи, в первую очередь, задачи двустороннего интервального оценивания неизвестных числовых множеств.

Разработанная теория, в частности, включает в себя определение твинной и направленной твинной арифметик, доказательство их свойств, разработку базовых алгоритмов интервального оценивания, определение их точности и сложности, изучение возможных приложений и компьютерной реализации.

Положения теории и пути ее применения формируются следующими основными результатами.

1. Предложена твинная арифметика, доказаны ее основные свойства.
2. Исследованы задачи внутренней и двусторонней оценки множества значений функции. Разработаны методы применения твинной арифметики для решения этих задач. Доказана их корректность.
3. Предложена направленная твинная арифметика как обобщение простой твинной арифметики. Доказаны ее свойства. Показано, что применение направленной твинной арифметики позволяет повысить точность полученных оценок путем учета дополнительной информации о монотонности оцениваемых функций.
4. Предложен алгоритм поиска глобальных экстремумов на основе использования твинной арифметики и деления множества определения функции на части.

5. Показана линейная зависимость точности двусторонней оценки от интервала, на котором производится оценка, в случае, если функция удовлетворяет условию Липшица.

6. Доказано, что вычислительная сложность конкретных алгоритмов вычисления двусторонней оценки, реализованных средствами твинных вычислений, эквивалентна вычислительной сложности соответствующих алгоритмов вычисления внешней оценки, реализованных средствами интервальных вычислений. Доказано существование алгоритма с полиномиальным временем выполнения, который для рациональной функции дает почти всегда точную оценку для достаточно узких интервалов.

7. Предложен алгоритм внутренней оценки, использующий разбиение исходного интервала.

8. Исследовано применение аппарата твинных вычислений к задачам вычислительной математики (к задаче оценки решений линейных систем, задаче анализа статической системы).

9. Исследовано применение предложенной теории к решению задач искусственного интеллекта (к задаче обработки экспертных знаний, задаче представления геометрии реального мира, задаче анализа экспериментальных данных и др.).

10. Сформулированы и обоснованы основные принципы машинной твинной арифметики. На их основе построено расширение языка Pascal-XSC, предоставляющее все необходимые средства для программирования твинных вычислений.

Литература

[1] Алефельд Г., Херцбергер Ю. *Введение в интервальные вычисления.* Мир, М., 1987, 360 с.

[2] Ащепков Л. Т. *К проблеме увеличения живучести управляемых систем.* В кн: Бельтюков Б. А., Булатов В. П. (ред.). Модели и методы исследования операций. Наука, Новосибирск, 1988, с. 69–85.

[3] Биркгоф Г. *Теория решеток.* Наука, М., 1984.

[4] Вощинин А. П., Сотиров Г. Р. *Оптимизация в условиях неопределенности.* Моск. энерг. ин-тут, М., Техника, София, 1990, 224 с.

[5] Гаганов А. А. *О сложности вычисления интервала значений полинома многих переменных.* Кибернетика **4** (1995), с. 6–8.

[6] Захаров А. В., Шокин Ю. И. *Синтез систем управления при интервальной неопределенности параметров их математических моделей.* Доклады Академии наук СССР **299** (2) (1988), с. 292–295.

[7] Зюзин В. С. *Твины и методы решения системы твинных уравнений.* В кн. Информационно-оперативный материал (интервальный анализ), Препринт N. 6, ВЦ СО АН СССР, Красноярск, 1988, с. 19-21.

[8] Йенсен К, Вирт Н. *Паскаль: руководство для пользователя.* М., Финансы и статистика, 1989.

[9] Калмыков С. А., Шокин Ю. И., Юлдашев З. Х. *Методы интервального анализа.* Наука, Новосибирск, 1986, 222 с.

[10] Клатте Р., Кулиш У., Неага М., Рац Д., Улльрих Х. *Pascal-XSC. Руководство по языку и учебный курс.* Теревинф, М., 1997, 336 с.

[11] Крейнович В. Я., Нгуен Х. Т., Городецкий В. И., Нестеров В. М., Тулупьев А. К. *Применение интервальных степеней доверия: аналитический обзор.* В кн. Р. М. Юсупов (ред.) Информационные технологии и интеллектуальные методы. Выпуск 3. Санкт-Петербург, 1999, с. 6–61.

[12] Курош А. Г. *Лекции по общей алгебре.* Наука, М., 1973.

[13] Лакеев А. В., Носков С. И. *Описание множества решений линейного уравнения с интервально заданным оператором и правой частью.* Доклады Академии наук **330** (4) (1993), с. 430–433.

[14] Матиясевич Ю. В. *Вещественные числа и ЭВМ.* В кн. Кибернетика и вычислительная техника, вып. 2. Наука, М., 1986, с. 104–133.

[15] Нариньяни А. С. *Недоопределенные множества – новый тип данных для представления знаний.* Препринт N. 232, ВЦ СО АН СССР, Новосибирск, 1980.

[16] Нариньяни А. С. *Средства моделирования неполноты данных в аппарате представления знаний.* В кн. Нариньяни А. С. (ред.) Представление знаний и моделирование процессов понимания. ВЦ СО АН СССР, Новосибирск, 1980.

[17] Нестеров В. М. *Автоматическое доказательство символьных неравенств в системе синтеза программ.* В кн.: Проблемы со-

вершенствования синтеза, тестирования, верификации и отладки программ. Т. II. Рига, 1986, с. 48–49.

[18] Нестеров В. М. *Автоматическое символьное решение уравнений.* Дисс. канд. ф.-м. н., ЛИИАН, 1988, 144 с.

[19] Нестеров В. М. *Об одном обобщении интервальной арифметики.* В кн.: Информационно-оперативный материал (интервальный анализ), Препринт N. 6. Красноярск, 1988, с. 31–33.

[20] Нестеров В. М. *Установление истинности логических выражений, составленных из неравенств.* В кн.: Материалы первого Сов.–Болг. семинара по числовой обработке, 19-23 окт. 1988. Переславль-Залесский, 1988, с. 10–15.

[21] Нестеров В. М. *Об интервальных расширениях функций* $R \to \{\mathrm{true}, \mathrm{false}\}$. В кн.: Конференция “Интервальная математика”. Саратов, 1989, с. 40–42.

[22] Нестеров В. М. *Совместное использование методов компьютерной алгебры и интервального анализа при создании программной системы с элементами искусственного интеллекта.* В кн.: Теория и применение искусственного интеллекта. Сборник трудов. Т. II. Созопол, 1989, с. 249–253.

[23] Нестеров В. М. *Интервальное вычисление логических выражений, составленных из неравенств.* В кн.: Информационно-оперативный материал (интервальный анализ), Препринт N. 9. Красноярск, 1989, с. 24–26.

[24] Нестеров В. М. *Оценка вычислительной сложности алгоритма Мура.* В кн.: Информационно-оперативный материал (интервальный анализ), Препринт N. 9. Красноярск, 1989, с. 26–28.

[25] Нестеров В. М. *Об одном обобщении интервального анализа и его применении для оценки множества значений функции.* В кн. Математические методы построения и анализа алгоритмов. Ленинград, 1990, с. 109–124.

[26] Нестеров В. М. *Вычисление интервальных расширений функций с использованием обобщенных интервальных арифметик.* В кн. Международная конференция по интервальным и стохастическим методам в науке и технике. Москва, 1992, с. 119–120.

[27] Нестеров В. М. *Интервальные арифметики и оценка множества значений функции.* В кн. Р. М. Юсупов (ред.) Теоретические основы и прикладные задачи интеллектуальных информационных технологий. Санкт-Петербург, 1998, с. 280–289.

[28] Никольский С. М. *Курс математического анализа, Т. 1.* Наука, М., 1983, 464 с.

[29] Фомин Ю. Т., Шокин Ю. И. *Введение в машинную интервальную арифметику.* Сиб. отд. ИТМП, Новосибирск, 1983, 34 с.

[30] Шарый С. П. *Новый подход к анализу статических систем с интервальной неопределенностью в данных.* Вычислительные технологии **2** (1) (1997), с. 71–83.

[31] Шарый С. П. *Алгебраический подход к анализу линейных статических систем с интервальной неопределенностью.* Известия Академии наук. Теория и системы управления (3) (1997), с. 51–61.

[32] Шокин Ю. И. *Интервальный анализ.* Наука, Новосибирск, 1981, 112 с.

[33] Шокин Ю. И. *Об интервальных задачах, интервальных алгоритмах и их трудоемкости.* Вычислительные технологии **1** (1) (1996), с. 98–115.

[34] Яковлев А. Г. *Интервальные вычисления на ЭВМ. Обзор.* В кн.: [1], с. 336–352.

[35] Яковлев А. Г. *Локусы и локализационные вычисления.* В кн.: Конференция "Интервальная математика". Саратов, 1989, с. 54–56.

[36] Яковлев А. Г. *Мультиаспектность в программировании локализационных (интервальных) вычислений.* В кн.: Конференция "Интервальная математика". Саратов, 1990, с. 113–120.

[37] *Abstracts for a workshop on interval methods in artificial intelligence.* In: Abstracts for an International Conference on Numerical Analysis with Automatic Result Verification. Lafayette, LA, 1993, part 2, p. 1–36.

[38] Adams E., Kulisch U. (eds) *Scientific computing with automatic result verification.* Academic Press, N. Y., 1993.

[39] Alefeld G., Frommer A., Lang B. (eds) *Scientific computing and validated numerics.* Academie Verlag, Berlin, 1996.

[40] Alefeld G., Trejo R. A. (eds) *Interval computations and its applications to reasoning under uncertainty, knowledge representation and control theory.* Mexico City, 1998.

[41] Apostolatos N., Kulisch U. *Grundlagen einer Maschinenintervallarithmetik.* Computing **2** (2) (1967), p. 89–104.

[42] Apostolatos N., Kulisch U. *Approximation der erweiterten Intervallarithmetik durch die einfache Maschinenintervallarithmetik.* Computing **2** (3) (1967), p. 181–194.

[43] Asaithambi N. S., Shen Zuhe, Moore R. E. *On computing the range of values.* Computing **28** (3) (1982), p. 225–237.

[44] Barth W., Nuding E. *Optimale Lösung von Intervallgleichungssystemen.* Computing **12** (1974), p. 117–125.

[45] Baur W., Strassen V. *The complexity of partial derivatives.* Theoretical Computer Science **22** (3) (1983), p. 317–330.

[46] Berner S. *A parallel method for verified global optimization.* In: [39], p. 200–206.

[47] Bleher J. H., Röder A. E., Rump S. *ACRITH: High accuracy arithmetic – an advanced tool for numerical computation.* In: Proceedings of 7th Symposium on Computer Arithmetic, 1985, p. 318–321.

[48] Bleher J. H., Rump S. M., Kulisch U., Metzger M., Ullrich Ch., Walter W. *FORTRAN-SC: A study of a FORTRAN extension for engineering/scientific computation with access to ACRITH.* Computing **39** (1987), p. 93–110.

[49] Bohlender G. *What do we need beyond IEEE arithmetic?* In: [161], p. 1–32.

[50] Bohlender G., Rall L., Ullrich Ch., Wolff von Gudenberg J. *PASCAL–SC: A computer language for scientific computation.* Academic Press, New York, 1987.

[51] Caprani O., Madsen K. *Mean value forms in interval analysis* Computing **25** (2) (1980), p. 147–154.

[52] Caprani O., Madsen K., Rall L.B. *Integration of interval functions.* SIAM J. of Math. Anal. **12** (3) (1981), p. 321–341.

[53] Carter T. M. *Cascade: Hardware for high/variable precision arithmetic.* In: Proceedings of the 9th Symposium on Computer Arithmetic, 1989, p. 184–191.

[54] Cohen M. S., Hull T. E., Hamacher V. C. *CADAC: A controlled precision decimal arithmetic unit.* IEEE Transactions on Computers **C-32** (1983), p. 370–377.

[55] Corliss G. F. *Industrial applications of interval techniques.* In: [161], p. 91–113.

[56] Dimitrova N., Markov S. M., Popova E. *Extended interval arithmetics: New results and applications.* In: Atanassova L., Herzberger J. (eds) Computer Arithmetic and Enclosure Methods, North-Holland, Amsterdam, 1992, p. 225–232.

[57] Fagin R., Halpern J. Y., Megiddo N. A. *Logic for reasoning about probabilities.* In: Proceedings of 3th IEEE Symposium on Logic and Computer Science, 1988.

[58] Falcó Korn C., König S., Gutzwiller S. *MODULA-SC: A precompiler to Modula-2.* In: [162], p. 371–384.

[59] Floudas A., Pardalos P. (eds) *Recent advances in global optimization.* Princeton Univ. Press, Princeton, 1992.

[60] Gardeñes E., Mielgo H., Trepat A. *Modal intervals: reason and ground semantics.* In: Nickel K. (ed.) Interval Mathematics 1985, (Lecture Notes in Computer Science **212**), Springer Verlag, Berlin etc., 1986, p. 27–35.

[61] Gardeñes E., Trepat A. *The interval computing system SIGLA-PL/1(0).* Freiburger Intervall-Berichte **8** (1979), 55 p.

[62] Gardeñes E., Trepat A. *Fundamentals of SIGLA, an interval computing system over the completed set of intervals.* Computing **24** (1980), p. 161–179.

[63] Gardeñes E., Trepat A., Janer J.M. *SIGLA-PL/1: Development and applications.* In: Nickel K. (ed.) Interval Mathematics, 1980, Academic Press, N.Y., 1980, p. 301–315.

[64] Gardeñes E., Trepat A., Janer J. M. *Approaches to simulation and to the linear problem in the Sigla system.* Freiburger Intervall-Berichte **8** (1981), p. 1–28.

[65] Gardeñes E., Trepat A., Mielgo H. *Present perspective of the SIGLA interval system.* Freiburger Intervall-Berichte **9** (1982), 57 p.

[66] Garey M. R., Johnson D. S. *Computers and intractability: A guide to the theory of NP-completeness.* Freeman, San Francisco, 1979.

[67] Good D. I., London R. L. *Computer interval arithmetic: Definition and proof of correct implementation.* J. of ACM **17** (1970), p. 603–612.

[68] Gorodetski V. I., Nesterov V. M. *Interval probabilities and knowledge engineering.* In: [40], p. 15–20.

[69] Hammer R., Hocks M., Kulisch U., Ratz D. *Numerical toolbox for verified computing I. Basic numerical problems.* Springer, Berlin etc, 1991.

[70] Hammer R., Hocks M., Kulisch U., Ratz D. *C++ toolbox for verified computing. Basic numerical problems.* Springer, Berlin etc, 1996.

[71] Hansen E. *A generalized interval arithmetic.* In: Nickel K. (ed.) Interval Mathematics (Lecture Notes in Computer Science **29**), Springer, Berlin, 1975, p. 7–18.

[72] Hansen E. *Global optimization using interval analysis.* Marcel Dekker, New York, 1992.

[73] Heintz J., Roy M.-F., Solerno P. *On the theoretical and practical complexity of the existential theory of reals.* The Computer Journal **36** (5) (1993).

[74] Hung T. Nguyen, Kreinovich V., Nesterov V., Nakamura M. *On hardware support for interval computations and for soft computing: theorems.* IEEE Transactions on Fuzzy Systems **5** (1) (1997), p. 108–127.

[75] Hyvönen E., De Pascale S. *C++ library family for interval computations.* In: [97], p. 85–90.

[76] Hyvönen E., De Pascale S. *Interval computations on the spreadsheet.* In: [87], p. 169–209.

[77] *IEEE standard 754 for binary floating point arithmetic.* American National Standards Institute, 1985.

[78] Jansson C. *On self-validating methods for optimization problems.* In: Herzberger J. (ed.) Topics in Validated Computations, North-Holland, Amsterdam, 1994, p. 381–439.

[79] Jerrell M. E. *Applications of interval computations to regional economic input-output models.* In: [87], p. 133–143.

[80] Kahan W. P. *Interval arithmetic options in the proposed IEEE floating point arithmetic standard.* In: [139], p. 99–128.

[81] Kaminsky T. E. *Interval rounding off latice.* In: International Congress on Computer Systems and Applied Mathematics. St. Petersburg, 1993, p. 90–91.

[82] Kaminsky T. E. *On semigroup of interval roundings.* In: International Conference on Interval and Computer-Algebraic Methods in Science and Engineering. St. Petersburg, 1994, p. 129–130.

[83] Kaucher E. *Interval analysis in the extended interval space IR.* Computing, Suppl. **2** (1980), p. 33–49.

[84] Kearfott R. B., Novoa M. *INTBIS, a portable interval Newton bisection package.* ACM Transactions on Mathematical Software **16** (1990), p. 152–157.

[85] Kearfott R. B., Dawande M., Du K., Hu C. *A portable FORTRAN 77 elementary function library.* Interval Computations 3 (6) (1992), p. 96–105.

[86] Kearfott R. B., Dawande M., Du K., Hu C. *Algorithm 737: INTLIB: A portable FORTRAN 77 interval standard function library.* ACM Transactions on Mathematical Software **20** (1994), p. 447–459.

[87] Kearfott R. B., Kreinovich V. (eds) *Applications of interval computations.* Kluwer Academic Publishers, Dordrecht etc., 1996, XIII+428 p.

[88] Kearfott R. B., Kreinovich V. *Applications of interval computations: An introduction.* In: [87], p. 1–22.

[89] Kearfott R. B. *A review of techniques in the verified solution of constrained global optimization problem.* In: [87], p. 23–59.

[90] Keiper J. B. *Interval arithmetic in Mathematica.* Interval Computations 3 (1993), p. 76–87.

[91] Knöfel A. *Hardware kernel for scientific/engineering computations.* In: [38], p. 549–570.

[92] Knöfel A. *Fast hardware units for the computation for accurate dot products.* In: Proceedings of the 10th Symposium on Computer Arithmetic, 1991, p. 70–75.

[93] Knüppel O. *PROFIL/BIAS – A fast interval library.* Computing **53** (1994), p. 277–288.

[94] Kohout L. J., Anderson J., Bandler W. *Knowledge-based systems for multiple environments.* Ashgate Publ. (Gover), Aldershot, 1992.

[95] Kohout L. J., Stabile I. *Interval-valued inference in medical knowledge-based system CLINAID.* Interval Computations (3) (1993), p. 88–115.

[96] Kreinovich V. *Data processing beyond traditional statistics: Applications of interval computations. A brief introduction.* In: [97], p. 13-21.

[97] Kreinovich V. (ed.) *APIC'95 international workshop on applications of interval computations.* El Paso, TX, 1995.

[98] Kreinovich V., Lakeyev A., Noskov S. *Optimal solution of interval linear systems is intractable (NP-hard).* Interval Computations 1 (1993), p. 6–14.

[99] Kreinovich V., Lakeyev A., Noskov S. *Approximate linear algebra is intractable.* Linear Algebra and Its Applications **232** (1) (1996), p. 45–54.

[100] Kreinovich V., Lakeyev A., Rohn J. *Computational complexity of interval algebraic problems: some are feasible and some are computationally intractable – a survey.* In: [39], p. 293–306.

[101] Kreinovich V., Lakeyev A., Rohn J., Kahl P., *Computational complexity and feasibility of data processing and interval computations.* Kluwer, Dordrecht, 1998.

[102] Kreinovich V., Nesterov V. M., Zheludeva N. A. *Interval methods that are guaranteed to underestimate (and the resulting new justification of Kaucher arithmetic).* Reliable Computing **2** (2) (1996), p. 119–124.

[103] Kulisch U., Miranker W. L. *Computer arithmetic in theory and practice.* Academic Press, New York, 1981, XIII+252 p.

[104] Kulisch U., Stetter H. J. (eds) *Scientific computing with automatic result verification,* Computing Supplementum **6**, Springer-Verlag, Vienna, 1989.

[105] Lakeyev A., Kreinovich V. *If input intervals are small enough, then interval computations are almost always easy.* In: [97], p. 134–139.

[106] Laveuve S. E. *Definition einer Kahan-Arithmetik und ihre Implementierung.* In: Nickel K. (ed.) Interval mathematics. (Lecture Notes in Computer Science **29**), Springer Verlag, Berlin etc., 1975, p. 236–245.

[107] Lea R. N., Kreinovich V., Trejo R. *Optimal interval enclosures for fractionally-linear functions, and their application to intelligent control.* Reliable Computing **2** (3) (1996), p. 265–285.

[108] Lerch M., Wolff von Gudenberg J. *Multiaspect interval types.* In: SCAN'98, IMACS/GAMM International Symposium on Scientific Computing, Computer Arithmetic and Validated Numerics, September 22-25, 1998, Budapest, Hungary. Budapest, 1998, p. 97.

[109] Lohner R. J. *Interval arithmetic in staggered correction format.* In: [38], p. 301–321.

[110] Markov S. M. *Extended interval arithmetic involving infinite intervals.* Mathematica Balkanica **6** (1992), p. 269–304.

[111] Markov, S. *On the presentation of ranges of monotone functions using interval arithmetic.* Interval Computations 4 (6) (1992), p. 19–31.

[112] Markov, S. *On directed interval arithmetic and its applications.* J. UCS **2** (7) (1995), p. 510–521.

[113] Markov, S. *On the foundations of interval arithmetic.* In: [39], p. 307–313.

[114] Markov, S. *Isomorphic embeddings of abstract interval systems.* Reliable Computing **3** (3) (1997), p. 199–207.

[115] Mints G. *A short introduction to modal logic.* CSLI, Stanford University, Stanford, 1992.

[116] Moore R. E. *Interval analysis.* Prentice Hall, Inglewood Cliffs, 1966, 145 p.

[117] Moore R. E. *Methods and applications of interval analysis.* SIAM, Philadelphia, 1979, XI+190 p.

[118] Moore R. E. *Reliability in computing: The role of interval methods in scientific computations.* Academic Press, N. Y., 1988.

[119] Nava P., Traylor J. M. *Speaker independent voice recognition with a fuzzy neural network.* In: Proceedings of the Fifth IEEE International Conference on Fuzzy Systems FUZZ-IEEE'96. New Orleans, September 8-11, 1996, v. 3, p. 2049–2052.

[120] Nesterov V. M. *Estimating a range of values of functions using extended interval arithmetics.* Interval Computations 4 (6) (1992), p. 48–53.

[121] Nesterov V. M. *How to use monotonicity-type information to get better estimates of the range of real-valued functions.* Interval Computations 4 (1993), p. 3–12.

[122] Nesterov V. M. *On some generalizations of interval arithmetic.* In: Abstracts for an International Conference on Numerical Analysis with Automatic Result Verification. Lafayette, 1993, p. 70.

[123] Nesterov V. M. *Interval arithmetic of the weakly monotone functions.* In: International Congress on Computer Systems and Applied Mathematics. St.-Petersburg, 1993, p. 98.

[124] Nesterov V. M. *On estimation of the range of values of function using a generalization of interval arithmetic. The multidimensional case.*

In: IMACS/GAMM International Symposium on Scientific Computing, Computer Arithmetic and Validated Numerics. Vienna, 1993, p 88.

[125] Nesterov V. M. *Interval analogues of Hilbert's 13th problem.* In: International Conference on Interval and Computer-Algebraic Methods. St.-Petersburg, 1994, p. 185–186.

[126] Nesterov V. M. *Implementation of generalized interval arithmetics.* In: International Symposium on Scientific Computing, Computer Arithmetic and Validated Numerics. Wuppertal, 1995, p. 97.

[127] Nesterov V. M. *Directed twin arithmetic.* In: II Workshop on Computer arithmetic, Interval and Symbolic Computation. Recife, 1996, p. 61–63.

[128] Nesterov V. M., Kreinovich V. *The worse, the better: A survey of paradoxical computational complexity of interval computations.* In: II Workshop on Computer arithmetic, Interval and Symbolic Computation. Recife, 1996, p. 61a–63a.

[129] Nesterov V. M. *Interval and twin arithmetics.* In: International Conference on Interval Methods and Computer Aided Proofs in Science and Engineering. Wuerzburg, 1996, p. 89–90.

[130] Nesterov V. M. *On accuracy of estimation in twin arithmetic.* In: SCAN'97, GAMM/IMACS International Symposium on Scientific Computing, Computer Arithmetic and Validated Numerics. Lyon, 1997, p. XII-5 – XII-7.

[131] Nesterov V. M. *Interval and twin arithmetics.* Reliable Computing **3** (4) (1997), p. 369–380.

[132] Nesterov V. M. *Two definitions of twin arithmetic. Improvement of the estimation accuracy.* In: Interval'98, International Conference on Interval Methods and their Applications on Global Optimization, Nanjing, April 20–23, 1998. Nanjing, 1998, p. 102–104.

[133] Nesterov V. *Algebraic properties of twin spaces.* In: SCAN'98, IMACS/GAMM International Symposium on Scientific Computing, Computer Arithmetic and Validated Numerics, September 22-25, 1998, Budapest, Hungary. Budapest, 1998, p. 121–122.

[134] Nesterov V. M., Gorodetski V. I. *Interval algorithm for checking the consistency of the algebraic Bayes net.* In: [40], p. 28–29.

[135] Nesterov V., Verchinine K. *Implementation of generalized Interval Arithmetics.* Technical Report. Université Paris XII, IUT, Départment Informatique de Fontainebleau, 1995.

[136] Neumaier A. *Tolerance analysis with interval arithmetic.* Freiburger Intervall-Berichte **86** (9) (1979), p. 5–19.

[137] Neumaier A. *Interval methods for systems of equations.* Cambridge University Press, Cambridge, 1990.

[138] Nguyen H. T., Kreinovich V. *Nested intervals and sets: Concepts, relations to fuzzy sets, and applications.* In: [87], p. 227–244.

[139] Nickel K. (ed.) *Interval mathematics.* Academic Press, New York etc., 1980, 227 p.

[140] Petunin D., Semenov A. *The use of multiintervals in the Unicalc solver.* In: [39], p. 91–97.

[141] Rall L. B. *Mean value and Taylor forms in interval analysis.* SIAM J. of Math. Anal. **14** (2) (1983), p. 223–238.

[142] Ratschek H., Schröder G. *Centered forms for functions in several variables.* J. of Math. Anal. and Applic. **82** (2) (1981), p. 543–552.

[143] Ratschek H., Rokne J. *New computer methods for global optimization.* Wiley, New York, 1988.

[144] Reyes S., Clarke M. *Logic for computer science.* Addison-Wesley, Wokingham, 1990.

[145] Rocha L. M., Kreinovich V., Kearfott R. B. *Computing uncertainty in interval based sets.* In: [87], p. 337–380.

[146] Rohn J. *NP-hardness results for some linear and quadratic problems.* Technical Report No. 619, Institute of Computer science, Academy of Sciencies of the Czech Republic, Prague, 1995, 11 p.

[147] Rohn J. *Linear interval equations: Computing enclosures with bounded relative overestimation is NP-Hard.* In: [87], p. 81–89.

[148] Rohn J., Kreinovich V. *Computing exact componentwise bounds on solutions of linear systems with interval data is NP-hard.* SIAM Journal on Matrix Analysis and Applications **16** (1995), p. 415–420.

[149] Schulte M. J., Swartzlander E. E. Jr. *A software interface and hardware design for variable precision interval arithmetic.* Reliable Computing **1** (4) (1995), p. 324–342.

[150] Schulte M. J., Swartzlander E. E. Jr. *Software and hardware techniques for accurate self-validating arithmetic.* In: [87], p. 381–404.

[151] Schulte M. J., Swartzlander E. E. Jr. *Variable-precision, interval arithmetic coprocessors.* Reliable Computing **2** (1) (1996), p. 47–62.

[152] Schulte M. J., Swartzlander E. E. Jr. *A processor for accurate, self-validating computing.* In: [39], p. 25–31.

[153] Semenov A. L. *Solving optimization problems with help of the Unicalc solver.* In: [87], p. 211–225.

[154] Shary S. P. *Algebraic approach to the interval linear static identification, tolerance, and control problems, or One more application of Kaucher arithmetic.* Reliable Computing **2** (1) (1996), p. 3–33.

[155] Shary S. P. *A new approach to the analysis of static systems under interval uncertainty.* In: [39], p. 118–132.

[156] Shary S. P. *Algebraic solutions to interval linear equations and their applications.* In: Alefeld G., Herzberger J. (eds) Numerical Methods and Error Bounds. Akademie Verlag, Berlin, 1996, p. 224–233.

[157] Shary S. P. *Algebraic approach in the outer problem for interval linear equations.* Reliable Computing **3** (2) (1997), p. 103–135.

[158] Shokin Yu. I. *On interval problems, interval algorithms and their computational complexity.* In: [39], p. 314–328.

[159] Skelboe S. *Computation of rational interval functions.* BIT **14** (1974), p. 87–95.

[160] Smith D. M. *Algorithm 693: A FORTRAN package for floating-point multiple-precision arithmetic.* ACM Transactions on Mathematical Software **17** (1991), p. 273–283.

[161] Ullrich Ch. (ed.) *Computer arithmetic and self-validating numerical methods.* Academic Press, New York etc, 1990.

[162] Ullrich Ch. (ed.) *Contributions to computer arithmetic and self-validating numerical methods.* J. C. Baltzer, 1990.

[163] Wolff von Gudenberg J. *PASCAL-SC: A Pascal extension for scientific computation.* In: Proceedings of the 10th IMACS World Congress on System Simulation and Scientific Computation. 1982, p. 402–482.

[164] Wyatt W. T., Lozier D. W., Orser D. J. *A portable extended precision arithmetic package and library with FORTRAN precompiler.* ACM Transactions on Mathematical Software **2** (1976), p. 209–231.

Приложение 1. Содержание модуля T_ARI

```
module T_ARI;

use I_ARI;

global type twin = record
         tp: boolean;
         int: interval;
         out: interval;
       end;

    {Функции для формирования твинных значений}
global function twinval (i1, i2: interval): twin;
global function twinval (r1, r2, r3, r4: real): twin;
global function twinval (i1: interval): twin;
global function twinval (r1: real): twin;

    {Совмещенные операции присваивания}
global operator := (var x: twin; y: interval);
global operator := (var x: twin; r: real);

    {Анализ твина на вырожденность}
global function degtwin (x: twin): boolean;

    {Доступ к элементам твина}
global function int (x: twin): interval;
global function ext (x: twin): interval;
global function intinf (x: twin): real;
global function intsup (x: twin): real;
global function extinf (x: twin): real;
global function extsup (x: twin): real;

    {Ввод-вывод твинов}
global procedure read (var f: text; var a: twin);
global procedure write (var f: text; a: twin);

    {Арифметика}
global operator + (x: twin) y: twin;
global operator - (x: twin) y: twin;
```

```
global operator + (x, y: twin) z: twin;
global operator - (x, y: twin) z: twin;
global operator + (x: twin; y: interval) z: twin;
global operator - (x: twin; y: interval) z: twin;
global operator + (x: twin; y: real) z: twin;
global operator - (x: twin; y: real) z: twin;
global operator + (x: interval; y: twin) z: twin;
global operator - (x: interval; y: twin) z: twin;
global operator + (x: real; y: twin) z: twin;
global operator - (x: real; y: twin) z: twin;

global operator * (x, y: twin) z: twin;
global operator / (x, y: twin) z: twin;
global operator * (x: twin; y: interval) z: twin;
global operator / (x: twin; y: interval) z: twin;
global operator * (x: twin; y: real) z: twin;
global operator / (x: twin; y: real) z: twin;
global operator * (x: interval; y: twin) z: twin;
global operator / (x: interval; y: twin) z: twin;
global operator * (x: real; y: twin) z: twin;
global operator / (x: real; y: twin) z: twin;

    {Теоретико-множественные отношения}
global operator = (x, y: twin) b: boolean;
global operator <> (x, y: twin) b: boolean;
global operator < (x, y: twin) b: boolean;
global operator <= (x, y: twin) b: boolean;
global operator > (x, y: twin) b: boolean;
global operator >= (x, y: twin) b: boolean;

global operator = (x: twin; y: interval) b: boolean;
global operator <> (x: twin; y: interval) b: boolean;
global operator < (x: twin; y: interval) b: boolean;
global operator <= (x: twin; y: interval) b: boolean;
global operator > (x: twin; y: interval) b: boolean;
global operator >= (x: twin; y: interval) b: boolean;

global operator = (x: twin; y: real) b: boolean;
global operator <> (x: twin; y: real) b: boolean;
global operator < (x: twin; y: real) b: boolean;
```

```
global operator <= (x: twin; y: real) b: boolean;
global operator > (x: twin; y: real) b: boolean;
global operator >= (x: twin; y: real) b: boolean;

global operator = (x: interval; y: twin) b: boolean;
global operator <> (x: interval; y: twin) b: boolean;
global operator < (x: interval; y: twin) b: boolean;
global operator <= (x: interval; y: twin) b: boolean;
global operator > (x: interval; y: twin) b: boolean;
global operator >= (x: interval; y: twin) b: boolean;

global operator = (x: real; y: twin) b: boolean;
global operator <> (x: real; y: twin) b: boolean;
global operator < (x: real; y: twin) b: boolean;
global operator <= (x: real; y: twin) b: boolean;
global operator > (x: real; y: twin) b: boolean;
global operator >= (x: real; y: twin) b: boolean;

global operator in (i: interval; t: twin) b: boolean;
global operator in (r: real; t: twin) b: boolean;

    {Математические функции}
global function sqr (t: twin): twin;
global function sqt (t: twin): twin;
global function exp (t: twin): twin;
global function ln (t: twin): twin;
global function sin (t: twin): twin;
global function cos (t: twin): twin;
global function tan (t: twin): twin;
global function arcsin (t: twin): twin;
global function arccos (t: twin): twin;
global function arctan (t: twin): twin;
```

Приложение 2. Содержание модуля TD_ARI

```
module TD_ARI;

use I_ARI, T_ARI;

global type mon=(-1,0,1);

global type dtwin = record
         tp:  boolean;
         dir:  mon;
         int:  interval;
         out:  interval;
       end;

    {Функции для формирования твинных значений}
global function dtwinval (t1: twin; m: mon): dtwin;
global function dtwinval (t1: twin): dtwin;
global function dtwinval (i1, i2: interval; m: mon): dtwin;
global function dtwinval (i1, i2: interval): dtwin;
global function dtwinval (r1, r2, r3, r4: real; m: mon): dtwin;
global function dtwinval (r1, r2, r3, r4: real): dtwin;
global function dtwinval (i1: interval; m: mon): dtwin;
global function dtwinval (i1: interval): dtwin;
global function dtwinval (r1: real; m: mon): dtwin;
global function dtwinval (r1: real): dtwin;

    {Совмещенные операции присваивания}
global operator := (var x: dtwin; y: twin);
global operator := (var x: dtwin; y: interval);
global operator := (var x: dtwin; r: real);

    {Анализ твина на вырожденность}
global function degtwin (x: dtwin): boolean;

    {Доступ к элементам твина}
global function direction (x: dtwin): mon;
global function int (x: dtwin): interval;
global function ext (x: dtwin): interval;
global function intinf (x: dtwin): real;
```

```
global function intsup (x: dtwin): real;
global function extinf (x: dtwin): real;
global function extsup (x: dtwin): real;

    {Ввод-вывод твинов}
global procedure read (var f: text; var a: dtwin);
global procedure write (var f: text; a: dtwin);

    {Арифметика}
global operator + (x: dtwin) y: dtwin;
global operator - (x: dtwin) y: dtwin;

global operator + (x, y: dtwin) z: dtwin;
global operator - (x, y: dtwin) z: dtwin;
global operator + (x: dtwin; y: twin) z: dtwin;
global operator - (x: dtwin; y: twin) z: dtwin;
global operator + (x: dtwin; y: interval) z: dtwin;
global operator - (x: dtwin; y: interval) z: dtwin;
global operator + (x: dtwin; y: real) z: dtwin;
global operator - (x: dtwin; y: real) z: dtwin;
global operator + (x: twin; y: dtwin) z: dtwin;
global operator - (x: twin; y: dtwin) z: dtwin;
global operator + (x: interval; y: dtwin) z: dtwin;
global operator - (x: interval; y: dtwin) z: dtwin;
global operator + (x: real; y: dtwin) z: dtwin;
global operator - (x: real; y: dtwin) z: dtwin;

global operator * (x, y: dtwin) z: dtwin;
global operator / (x, y: dtwin) z: dtwin;
global operator * (x: dtwin; y: twin) z: dtwin;
global operator / (x: dtwin; y: twin) z: dtwin;
global operator * (x: dtwin; y: interval) z: dtwin;
global operator / (x: dtwin; y: interval) z: dtwin;
global operator * (x: dtwin; y: real) z: dtwin;
global operator / (x: dtwin; y: real) z: dtwin;
global operator * (x: twin; y: dtwin) z: dtwin;
global operator / (x: twin; y: dtwin) z: dtwin;
global operator * (x: interval; y: dtwin) z: dtwin;
global operator / (x: interval; y: dtwin) z: dtwin;
global operator * (x: real; y: dtwin) z: dtwin;
```

```
global operator / (x: real; y: dtwin) z: dtwin;

    {Теоретико-множественные отношения}
global operator = (x, y: dtwin) b: boolean;
global operator <> (x, y: dtwin) b: boolean;
global operator < (x, y: dtwin) b: boolean;
global operator <= (x, y: dtwin) b: boolean;
global operator > (x, y: dtwin) b: boolean;
global operator >= (x, y: dtwin) b: boolean;

global operator = (x: dtwin; y: twin) b: boolean;
global operator <> (x: dtwin; y: twin) b: boolean;
global operator < (x: dtwin; y: twin) b: boolean;
global operator <= (x: dtwin; y: twin) b: boolean;
global operator > (x: dtwin; y: twin) b: boolean;
global operator >= (x: dtwin; y: twin) b: boolean;

global operator = (x: dtwin; y: interval) b: boolean;
global operator <> (x: dtwin; y: interval) b: boolean;
global operator < (x: dtwin; y: interval) b: boolean;
global operator <= (x: dtwin; y: interval) b: boolean;
global operator > (x: dtwin; y: interval) b: boolean;
global operator >= (x: dtwin; y: interval) b: boolean;

global operator = (x: dtwin; y: real) b: boolean;
global operator <> (x: dtwin; y: real) b: boolean;
global operator < (x: dtwin; y: real) b: boolean;
global operator <= (x: dtwin; y: real) b: boolean;
global operator > (x: dtwin; y: real) b: boolean;
global operator >= (x: dtwin; y: real) b: boolean;

global operator = (x: twin; y: dtwin) b: boolean;
global operator <> (x: twin; y: dtwin) b: boolean;
global operator < (x: twin; y: dtwin) b: boolean;
global operator <= (x: twin; y: dtwin) b: boolean;
global operator > (x: twin; y: dtwin) b: boolean;
global operator >= (x: twin; y: dtwin) b: boolean;

global operator = (x: interval; y: dtwin) b: boolean;
global operator <> (x: interval; y: dtwin) b: boolean;
```

```
global operator < (x: interval; y: dtwin) b: boolean;
global operator <= (x: interval; y: dtwin) b: boolean;
global operator > (x: interval; y: dtwin) b: boolean;
global operator >= (x: interval; y: dtwin) b: boolean;

global operator = (x: real; y: dtwin) b: boolean;
global operator <> (x: real; y: dtwin) b: boolean;
global operator < (x: real; y: dtwin) b: boolean;
global operator <= (x: real; y: dtwin) b: boolean;
global operator > (x: real; y: dtwin) b: boolean;
global operator >= (x: real; y: dtwin) b: boolean;

global operator <<= (t1, t2: dtwin) b: boolean;

    {Математические функции}
global function sqr (t: dtwin): dtwin;
global function sqt (t: dtwin): dtwin;
global function exp (t: dtwin): dtwin;
global function ln (t: dtwin): dtwin;
global function sin (t: dtwin): dtwin;
global function cos (t: dtwin): dtwin;
global function tan (t: dtwin): dtwin;
global function arcsin (t: dtwin): dtwin;
global function arccos (t: dtwin): dtwin;
global function arctan (t: dtwin): dtwin;
```