
Содержание

Предисловие.....	iii
Глава 1 Обзор возможностей решателя UniCalc	1-1
Вычислительные возможности	1-2
Пользовательский интерфейс.....	1-3
Глава 2 Первые шаги в использовании решателя.....	2-1
Как начать работу с решателем UniCalc.....	2-2
Как ввести математическую модель	2-3
Как получить результаты вычислений	2-3
Упрощенная модель	2-4
Демонстрация некоторых возможностей UniCalc.....	2-4
Глава 3 Язык решателя UniCalc	3-1
Модель	3-2
Алфавит языка	3-2
Имена	3-3
Ключевые слова	3-3
Переменные.....	3-3
Константы	3-4
Массивы.....	3-4
Операции	3-5
Операции отношения.....	3-5
Арифметические операции	3-8
Логические операции.....	3-9
Оператор присваивания	3-11
Циклы, суммы и произведения	3-11
Функции.....	3-13
Стандартные математические функции.....	3-13
Пользовательские функции.....	3-14
Функция дифференцирования	3-14
Комментарии.....	3-15
Формальное описание синтаксиса языка	3-15
Глава 4 Взаимодействие пользователя с решателем	
UniCalc	4-1
Главное меню	4-2
Меню Вычисления.....	4-2
Команда Запустить.....	4-2
Команда Продолжить	4-3
Команда Поиск корней	4-3

Команда Искать дальше	4-5
Команда Предыдущее решение	4-5
Команда Следующее решение	4-5
Команда График	4-5
Меню Параметры.....	4-8
Команда Точность.....	4-8
Команда Диапазоны.....	4-8
Команда Степенная функция	4-9
Команда Аналитика	4-10
Команда Решения.....	4-10
Меню Справка.....	4-11
Команда Содержание.....	4-11
Команда Использование справки	4-11
Команда О программе.....	4-11
Локальные меню	4-12
Пиктограммы	4-13
Функциональные клавиши	4-14

Глава 5 Некоторые рекомендации по решению задач 5-1

О существовании решения.....	5-2
О решении, полученном на UniCalc	5-2
Математические модели с точными значениями параметров	5-2
Математические модели с интервальными параметрами	5-3
Зависимость решения от вида представления системы ..	5-3
Классы задач, для которых могут быть получены слишком широкие интервалы	5-3
Вычисление корней полиномов	5-4
Решение систем линейных уравнений.....	5-5
Решение обратных задач.....	5-6
Решение тригонометрических уравнений.....	5-7
Решение задач оптимизации.....	5-8

Глава 6 Сообщения системы..... 6-1

Сообщения UniCalc	6-2
Синтаксические ошибки.....	6-2
Ошибки в процессе вычислений.....	6-3

Приложение А Справочник по пользовательскому интерфейсу..... 1

Приложение Б Примеры решения задач..... 1

Приложение В Термины..... 1

Предисловие

Это руководство знакомит Вас с программным продуктом UniCalc — интеллектуальным решателем математических задач.

UniCalc предназначен для решения прямых и обратных задач, представленных системами алгебраических уравнений, неравенств и логических выражений. Решаемая система может быть переопределенной или недоопределенной, а параметры уравнений и неравенств могут быть заданы неточно, в виде интервалов допустимых значений. UniCalc позволяет проводить вычисления как с целыми, так и с вещественными переменными, причем они могут входить в систему одновременно. В результате вычислений находится гипербрус, в котором гарантированно лежат все действительные решения системы. Если заданная система не имеет действительных решений, то выдается сообщение о ее несовместности.

UniCalc можно использовать для решения широкого класса прикладных задач в области моделирования процессов (физических, химических, социально-экономических и т.п.), в инженерных расчетах (проектировании), для качественного анализа проблемы в научных исследованиях, а также в учебных целях, т.е. там, где постановка задачи может быть сведена к решению систем уравнений и неравенств, а также отдельных уравнений над полем вещественных чисел или кольцом целых чисел.

UniCalc особенно полезен для инженеров, экономистов, проектировщиков и других специалистов, которым в повседневной практике приходится решать реальные математические задачи среднего размера. Формулировка задачи может содержать десятки и сотни математических выражений. Во многих случаях UniCalc способен решать задачи, которые непосильны для традиционной вычислительной математики, причем, как правило, чем более сложные нелинейные зависимости включает задача, тем лучше с ней справляется решатель.

Уравнения (неравенства), подаваемые на вход решателю, являются либо алгебраическими действительными уравнениями (неравенствами), либо трансцендентными, т.е. уравнениями (неравенствами) с действительными коэффициентами, включающими ряд элементарных функций (\exp , \ln , \sin , \cos и т.д.) от действительных аргументов. Кроме того, к системе могут быть добавлены логические выражения, задающие ограничения на данные или вводящие дополнительные связи между данными. Для более компактной записи математических моделей язык UniCalc включает массивы, операторы цикла, операции суммирования и произведения.

Глава 1

Обзор возможностей решателя UniCalc

Вычислительные возможности

Системы линейных и нелинейных уравнений и неравенств, для решения которых предназначен UniCalc, являются, как правило, математическими моделями некоторых реальных процессов и явлений.

Математические модели, с которыми приходится иметь дело в практических приложениях, характеризуются следующими особенностями:

- Сложность задания точных значений параметров, определяющих модель. Это обусловлено недостаточностью знаний природы изучаемого явления или погрешностью измерительных приборов. Зачастую возможно указать только предельные значения физических величин — интервал их изменения. Отсюда вытекает необходимость решения систем уравнений с неточными параметрами, что практически невозможно сделать с помощью имеющихся решателей.
- Отсутствие хороших начальных приближений решения. Практически все алгоритмы решения нелинейных систем требуют задания начального приближения корней, неудачный выбор которого может привести к тому, что решение не будет найдено.
- Отсутствие стандартных вычислительных алгоритмов для решения многих видов нелинейных систем.

Перечисленные особенности приводят к тому, что при моделировании сложных явлений часто получаются системы уравнений, которые не могут быть решены с применением известных алгоритмов. Поэтому для их решения производится упрощение исходных систем, сведение к другим системам, а также выполняется линеаризация, приводящая к изменению исходной модели.

UniCalc позволяет решать в исходной постановке наряду с обычными системами уравнений и системы, обладающие вышеперечисленными свойствами. Это обусловлено использованием нетрадиционного метода — **метода недоопределенных вычислений (МНВ)**, который дает возможность:

- решать линейные и нелинейные уравнения, неравенства, а также соответствующие системы уравнений и неравенств;
- решать смешанные системы уравнений и неравенств, которые включают разнотипные переменные — целые, вещественные;
- производить вычисления не только с обычными переменными, но и с их областями значений;
- решать системы любой сложности без задания начальных приближений;
- решать системы целочисленных уравнений и неравенств;
- решать некоторые виды оптимизационных задач, в том числе задачи целочисленной оптимизации.

Вычислительный механизм, используемый в UniCalc, позволяет получать для каждой переменной:

- точное значение или некоторый интервал, содержащий решение, если оно единственно;
- интервал значений, содержащий все решения (с возможностью выделения всех точных решений в этом интервале), если решений несколько;
- сообщение о несовместности системы, если решений нет.

Получение интервала в случае системы с точными параметрами и единственным решением означает, что для нахождения точного решения не достаточно имеющейся информации. Возможные действия в этом случае приведены в главе Глава 5.

Каждое из решений в случае системы с несколькими решениями можно найти, добавляя дополнительные ограничения на переменные — в форме уравнений, неравенств и логических формул — или используя команду **Поиск корней**.

Пользовательский интерфейс

Многофункциональный пользовательский интерфейс UniCalc поддерживает все стандартные возможности соответствующей версии Windows для работы с окнами:

- открытие нескольких окон одновременно для работы с различными моделями и их модификациями;
- изменение размеров окон;
- свободное перемещение по экрану окна с изображением различного рода информации (текстов моделей, текстов с результатами) для оптимального их расположения;
- сворачивание и восстановление окон.

UniCalc представляет собой интегрированную среду, которая позволит организовать полный цикл процесса вычислений, начиная с записи математической модели с использованием встроенного в решатель редактора и заканчивая выводом результатов на печать или в виде графиков. Пользовательский интерфейс обеспечивает работу с решателем посредством команд меню, пиктограмм и функциональных клавиш. Для эффективной работы с решателем можно использовать манипулятор «мышь».

Выбрав команду меню или нажав функциональную клавишу (либо комбинацию клавиш), можно вводить тексты моделей и сохранять их в файлах, открывать и редактировать уже существующие в архиве тексты моделей и, задавая необходимую точность, находить решения рассматриваемых задач, а также просматривать различные модели и результаты их расчетов. Решения и получившаяся после аналитических преобразований (см. стр. 4-10) упрощенная модель могут запоминаться в архивах и просматриваться при необходимости.

Чтобы правильно ввести текст модели, Вам следует изучить возможности языка решателя UniCalc и правила записи на нем математических выражений. Описание языка приведено в главе Глава 3.

Познакомившись с командами и языком UniCalc, можно легко записывать новые задачи, модифицировать существующие и находить все их возможные решения.

Глава 2

Первые шаги в использовании решателя

Как начать работу с решателем UniCalc

Если у Вас есть компьютер, совместимый с IBM PC, операционная система MS Windows и решатель UniCalc, то для решения самой сложной задачи Вам потребуется всего несколько минут.

После переписи файлов с дистрибутивной дискеты в произвольный каталог на жестком диске запустите на исполнение файл UNICALC.EXE.

На дистрибутиве в каталоге EXAMPLES записан ряд файлов с описанием математических задач. Чтобы посмотреть, как справляется с ними UniCalc, выберите команду меню **Файл / Открыть** и откройте нужный файл.

Замечание. Если у Вас возникнут вопросы после запуска решателя, обратитесь к системе подсказок, которую можно вызвать в любой момент нажатием клавиши **F1**, или, выбрав команду меню **Справка**, ознакомьтесь с интересующими Вас темами.

Рабочая панель решателя на экране

После вызова UniCalc на экране дисплея появляется рабочая панель с заставкой (Рис. Глава 2.1), на которой имеется информация о решателе. Чтобы начать работу, нажмите клавишу **Esc** или **Enter** или кнопку ОК.

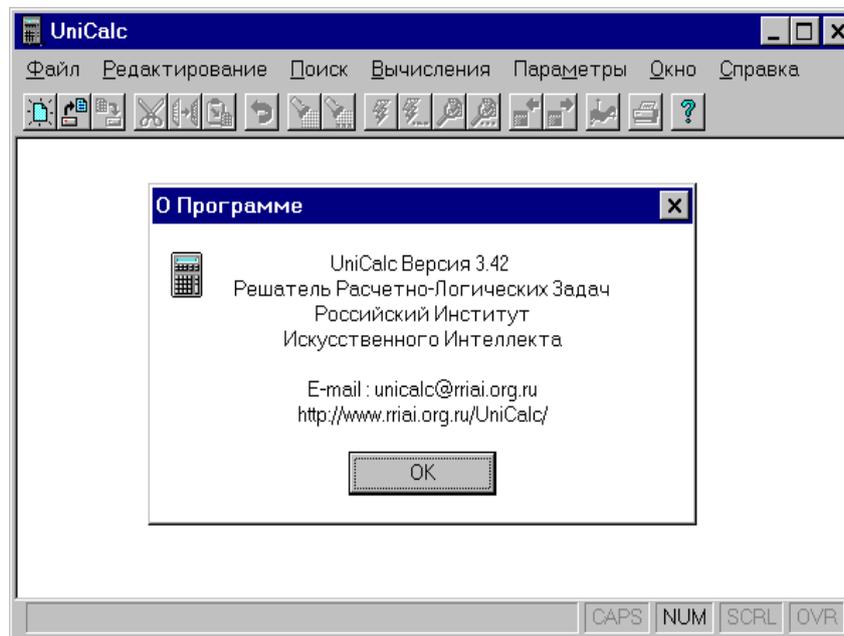


Рис. Глава 2.1. Рабочая панель с заставкой

Как ввести математическую модель

Для ввода текста модели в решатель необходимо вызвать встроенный редактор командой меню **Файл / Создать**. Выражения, описывающие модель на языке UniCalc, вводятся с клавиатуры. После ввода модели ее можно записать в архив командой **Файл / Сохранить как**. Именам файлов, содержащих тексты на языке UniCalc, приписывается расширение *uni*.

Уже существующий файл с описанием модели можно вызвать в окно редактирования командой **Файл / Открыть**.

Как получить результаты вычислений

Для получения решений Вашей задачи после ввода ее условий в окно редактирования надо запустить процесс вычислений. Для этого, если требуется, задайте точность вычисления командой меню **Параметры / Точность** и выберите команду меню **Вычисления / Запустить** (или нажмите клавишу **F9**). Решения выводятся в окно результатов, связанное с окном модели. По умолчанию все расчеты ведутся с точностью 0.0001.

В окне результатов приводятся:

- имена переменных;
- интервал (нижняя и верхняя граница), в котором лежат значения соответствующих переменных.

Если модель содержит много переменных, то для просмотра можно выделить только наиболее важные из них. Для этого надо в окне результатов пометить нужные переменные левой кнопкой мыши или клавишей **Enter**, после чего выполнить команду **Показать только выбранные** меню **Просмотр** (стр. 4-12). В окне останутся только отмеченные переменные, которые будут выводиться и при дальнейших расчетах на этой модели. Для показа всех переменных необходимо выполнить команду **Показать все** меню **Просмотр** (стр. 4-12).

Если значение переменной определено точно, то после знака « \Leftrightarrow » приводится ее точное значение. Если же система несовместна, то в специальном окне будет выдано сообщение об этом, а в окне результатов появятся значения, которые переменные имели перед обнаружением несовместности системы.

Результаты расчетов, так же, как и тексты моделей, можно сохранить на диске. Имя файла с результатами составляется из имени файла, содержащего модель, и расширения *val*.

Чтобы получить отчет о проделанной работе, отредактируйте, если требуется, текст Вашей модели (например, сделайте ее более компактной, вставьте необходимые комментарии) и вызовите команду меню **Файл / Печать**.

Упрощенная модель

В решатель UniCalc встроен препроцессор символьных преобразований, который производит некоторые упрощения в модели. К таким упрощениям относятся символьное дифференцирование, приведение подобных членов и вычисление константных выражений. Модель, получившаяся после символьных преобразований, можно просматривать в окне упрощенной модели, если в пункте меню **Аналитика** выбрать опцию **Показывать упрощенную модель**. Если в меню **Аналитика** задана опция **Решать линейные уравнения**, то преобразованную к треугольному виду систему линейных уравнений можно также просматривать в окне упрощенной модели.

Упрощенная модель может быть записана на диск. Имя файла с упрощенной моделью составляется из имени файла, содержащего модель, и расширения *sim*.

Демонстрация некоторых возможностей UniCalc

Процесс решения задач с помощью UniCalc можно продемонстрировать на примере нахождения корней полинома. Пусть задан полином $x^3 - 3x + 1 = 0$, и требуется найти все его действительные корни.

Воспользуемся одним из способов ввода математической модели: вызовем команду **Файл / Создать**. Далее в рабочем окне с помощью средств встроенного редактора введем полином на языке решателя UniCalc (см. Рис. Глава 2.2) и запустим его на счет (командой меню **Вычисления / Запустить** или клавишей **F9**):

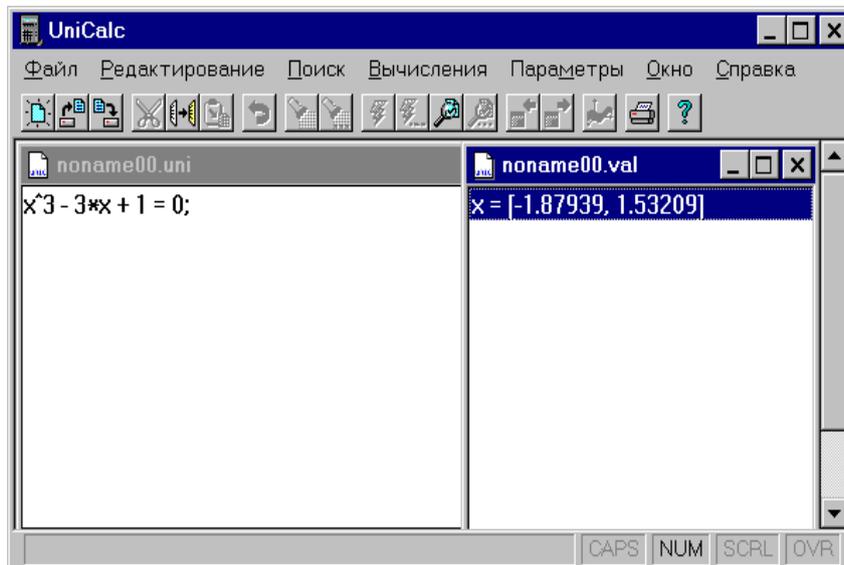


Рис. Глава 2.2. Полином и интервал, содержащий все его корни

В результате вычислений будет найден интервал

$$x = [-1.87939, 1.53209],$$

где лежат все корни полинома. Для нахождения точных значений корней к полиному надо добавить некоторое ограничение. Для простоты попробуем рассмотреть левую полуось: добавим неравенство $x < 0$. После вычислений получится точное значение отрицательного корня:

$$x = [-1.87939, -1.87939].$$

Имеется возможность получить корень с большим числом десятичных знаков, изменив точность командой меню **Параметры / Точность** и повторив вычисления (см. Рис. Глава 2.3).

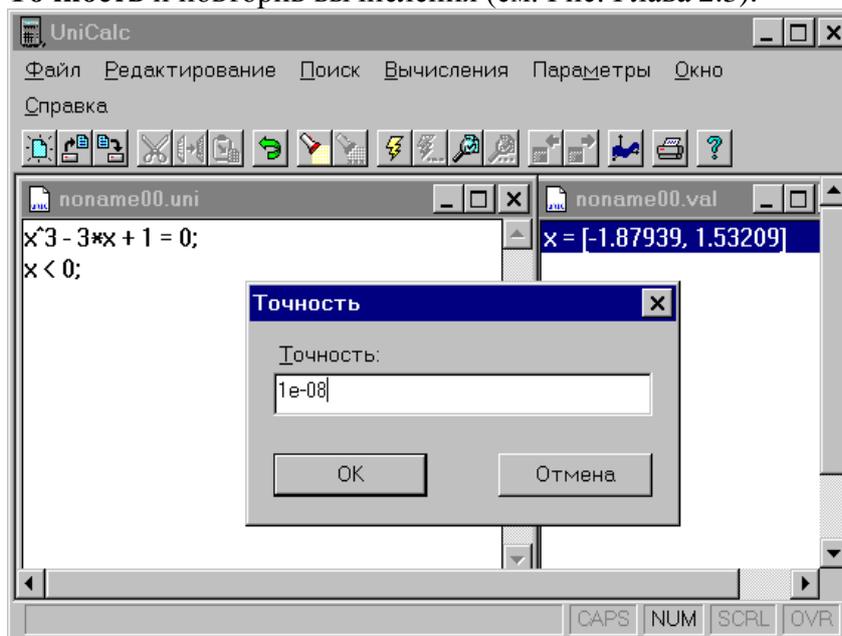


Рис. Глава 2.3. Ввод нового значения точности

При точности 10^{-8} получается следующий корень:

$$x = [-1.87938524, -1.87938524].$$

Наложив ограничения $x > 0$ и $x < 1$, мы найдем второй корень:

$$x = [0.347296, 0.347296].$$

Решение данного уравнения с условием $x > 1$ позволит найти третий корень:

$$x = [1.53209, 1.53209].$$

Если же мы попробуем искать решение при $x > 10$, нам будет выдано сообщение о несовместности системы, т.е. решений данной задачи в поле действительных чисел нет (см. Рис. Глава 2.4).

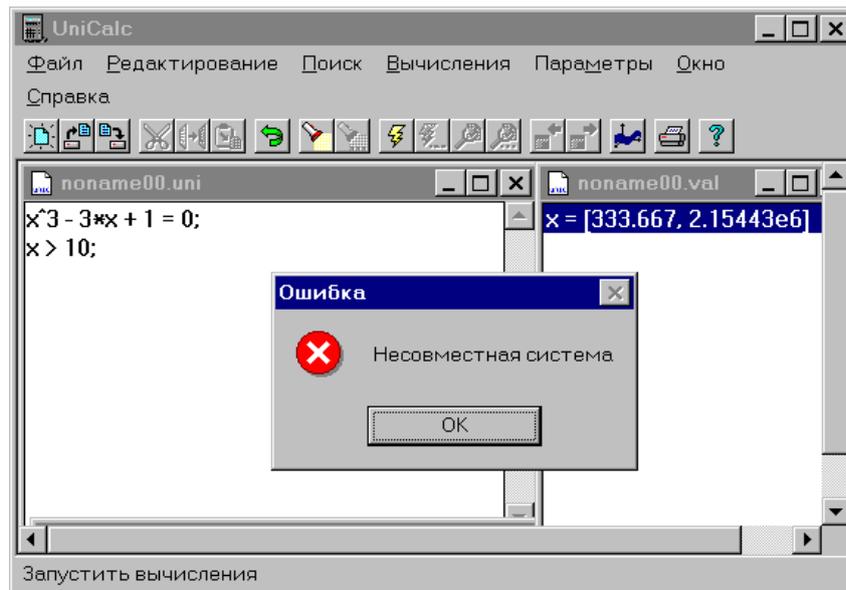


Рис. Глава 2.4. Получено сообщение о несовместности системы

Нахождение корней процедурой Поиск корней

Другой способ нахождения действительных корней состоит в использовании встроенного механизма поиска решений, основанного на методе деления отрезка пополам. Вызов осуществляется командой меню **Вычисления / Поиск корней**.

Перед поиском решения можно выбрать имя переменной, по которой осуществляется поиск корней. Уточнив направление поиска, кнопкой ОК следует запустить процесс нахождения корней (см. Рис. Глава 2.5).

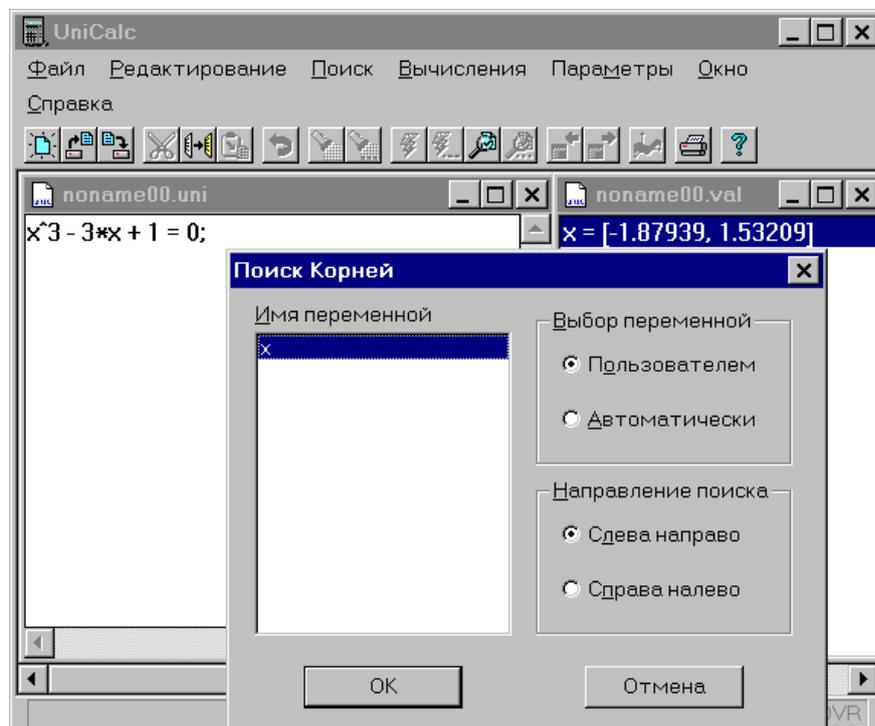


Рис. Глава 2.5. Запуск команды Поиск корней

После нахождения очередного корня делается запрос о необходимости поиска следующего корня (см. Рис. Глава 2.6):

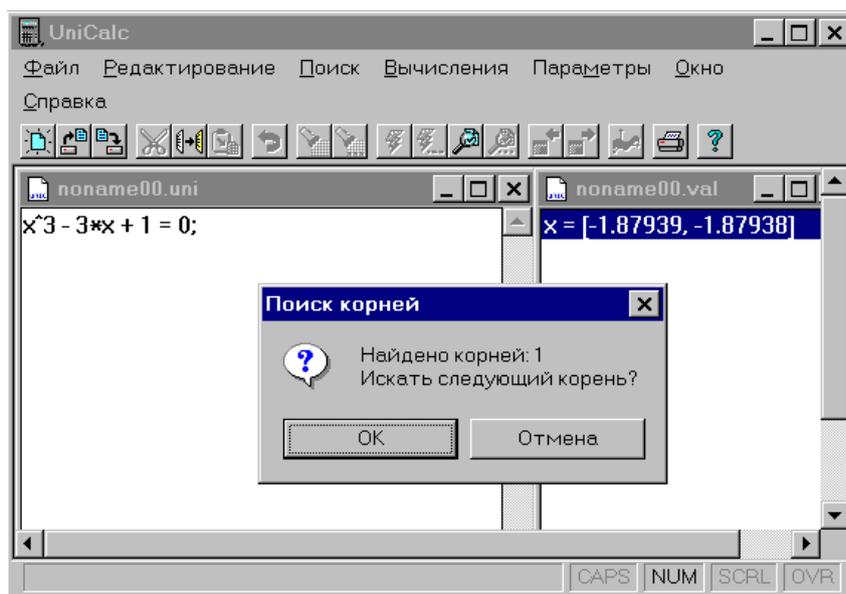


Рис. Глава 2.6. Запрос о следующем корне

Запрос повторяется до тех пор, пока решатель не вычислит все корни. Если найденный корень оказался последним, то выдается соответствующее сообщение (Рис. Глава 2.7).

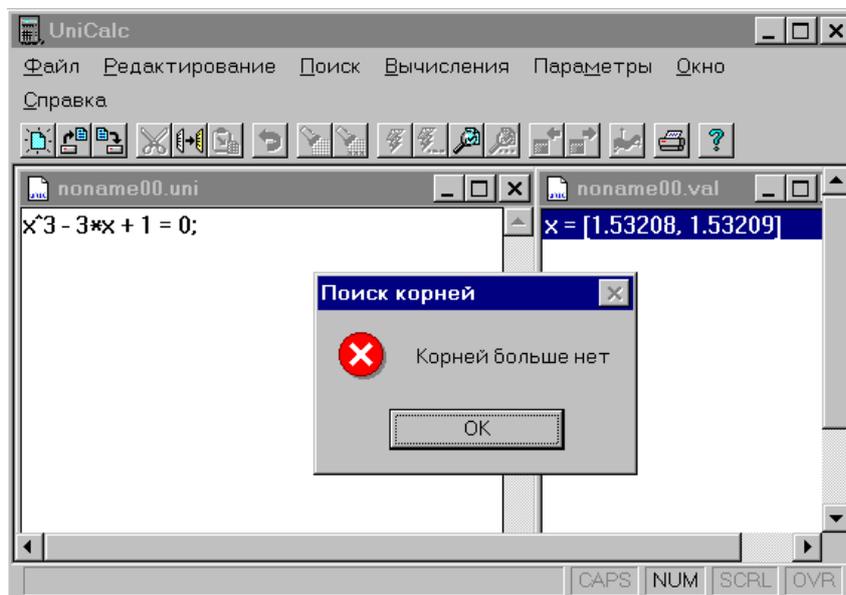


Рис. Глава 2.7. Найден последний корень

В этой главе продемонстрированы только некоторые возможности решателя и предложена самая простая схема работы с UniCalc. Полное представление о решателе и его возможностях Вы получите после прочтения данного руководства.

Глава 3

Язык решателя UniCalc

Язык решателя UniCalc максимально приближен к общепринятому языку записи математических выражений и позволяет легко записывать алгебраические и логические выражения. Для задания выражений используются переменные, константы, специальные символы, ключевые слова, арифметические и логические операции, операции отношения, массивы, циклы, операции суммирования и произведения. В решаемую систему могут быть включены комментарии.

Ниже дается неформальное описание основных языковых конструкций с примерами их использования. Формальное описание синтаксиса языка в форме Бэкуса-Наура дается в последнем разделе этой главы.

Модель

Модель является основной вычислительной единицей языка UniCalc и представляет собой набор *отношений*, описывающих моделируемый объект или явление. Каждое отношение модели является уравнением, неравенством или логическим выражением и заканчивается точкой с запятой.

Уравнения и неравенства записываются в виде двух алгебраических выражений, соединенных операцией отношения. Алгебраические выражения состоят из переменных целого и вещественного типов, точных и интервальных констант и стандартных и пользовательских функций, связанных знаками арифметических операций. Для удобства записи выражений можно использовать переменные с индексами (массивы) и операторы цикла, суммирования и произведения.

Логические выражения представляют собой уравнения и неравенства, связанные логическими операциями. Так как значениями алгебраических выражений в общем случае являются интервалы, то в операциях отношения и логических операциях используется трехзначная логика.

Число переменных и выражений в модели может не совпадать, то есть решаемая система может быть переопределенной или недоопределенной.

Алфавит языка

Алфавит входного языка решателя UniCalc состоит из всех букв латинского и русского алфавитов, всех цифр и набора специальных символов. Специальные символы используются для обозначения операций и построения выражений. В языке определены следующие специальные символы:

+ - * / ^
 () []
 = < >
 . , ; :
 @

Имена

Имена служат для обозначения переменных и функций. Имя должно состоять только из букв и цифр, причем первым символом обязательно должна быть буква. Длина имени не ограничивается, но имя распознается по первым тридцати двум символам. Прописные и строчные буквы различаются.

Примеры допустимых имен:

t, x, ALPHA, EPS, x10, y1.

Примеры недопустимых имен:

1X, EPS-2.

Ключевые слова

В языке UniCalc есть ключевые (зарезервированные) слова, которые не могут использоваться в качестве имен. К ним относятся названия всех стандартных математических функций (см. стр. 3-13) и следующие слова:

and or not for sum prod dif

Переменные

В UniCalc рассматриваются переменные двух типов: целые и вещественные. Тип переменной задается неявно и определяется первым символом ее имени. Имена целых переменных начинаются с букв *i, j, k, l, m, n* для латинского алфавита. Для прописных букв умолчание сохраняется. Остальные переменные считаются вещественными.

Тип переменной определяет способ выполнения арифметических операций над значением, присвоенным соответствующей переменной. Логические переменные в UniCalc в явном виде отсутствуют, однако некоторые операции вырабатывают логический результат (ИСТИНА, ЛОЖЬ или НЕОПРЕДЕЛЕНО), который может использоваться в логических операциях, а логические выражения соответствующим образом интерпретируются решателем.

Любая переменная может быть определена как точно, так и областью допустимых значений. Область допустимых значений задается в виде правильного интервала. На языке UniCalc интервал записывается следующим образом:

[< нижняя граница >, < верхняя граница >].

Нижняя и верхняя границы — это целые или вещественные числа, причем <нижняя граница> \leq <верхняя граница>.

В результате вычислений интервалы могут только сужаться, поэтому одной и той же переменной в модели нельзя присваивать в качестве значения непересекающиеся интервалы, так как последнее приводит к несовместной системе.

Константы

Для задания конкретных числовых значений используются константы. С помощью арифметических операций из них конструируются константные выражения.

В UniCalc определены константы двух типов: целые и вещественные. Целые константы — это целые числа в диапазоне от -2147483647 до $+2147483647$. Вещественные константы — вещественные числа в диапазоне от -10^{19} до $+10^{19}$ (в настоящей версии).

Примеры записи целых констант:

$-10, -5, 0, 15, 100.$

Примеры записи вещественных констант:

$-0.03, 3.14, 0.9999999, 10.44e-5.$

Язык позволяет использовать специальные константы:

$\pi = 3.1415926535897,$
 $e = 2.7182818284590,$
 $g = 9.80665.$

Для записи этих констант используется символ "@", помещаемый перед соответствующим именем (могут использоваться строчные и прописные буквы). Таким образом, запись @pi или @Pi означает константу 3.141..., запись @e или @E означает константу 2.718..., запись @g или @G означает константу 9.80665.

В языке UniCalc можно задавать *интервальные константы*, которые используются для инициализации переменных и в процессе вычислений не могут изменяться. Для этого перед интервалом необходимо поставить специальный символ @.

Примеры записи интервальных констант:

@[-3.5, 2.71], @[-4, 23].

Примеры

$T = [- @\pi/2, @\pi/2];$
 $S = @\pi * r^2;$
 $f = (@e^x - @e^{-x})/2;$
 $F = M * @g^2/2;$
 $a = @[-3.5, 2.71];$

Массивы

Язык UniCalc допускает использование массивов произвольной размерности. Единственным ограничением является то, что длина имени массива вместе с индексным выражением не должна превышать тридцати двух символов. Тип элементов массива определяется по имени массива согласно соглашению об именах переменных в UniCalc.

Индексные выражения должны быть целыми константами или константными выражениями, которые дают в результате целое число. Индексное выражение может быть не целым в единственном случае — когда элемент массива находится в теле определения пользовательской функции. Однако при вызове такой функции все индексные выражения после подстановки параметров должны принимать целочисленное значение. Индексы могут быть не только положительными, но и отрицательными.

Пример

```
SumLine(i) := sum(k = 1, 1, 10; a [i, k]);
s = SumLine(1);
```

Массивы не объявляются. Только те элементы массива, которые входят в модель, будут храниться в памяти и участвовать в вычислениях.

Операции

Все операции, определяемые в UniCalc, делятся на три группы: операции отношения, арифметические и логические. Каждая группа операций имеет соответствующие типы данных и тип результата. Ниже приводится список операций.

Операции отношения

Операции отношения позволяют задавать ограничения на переменные и арифметические выражения. Результатом операций отношения является логическое значение (ЛОЖЬ, ИСТИНА, НЕОПРЕДЕЛЕНО). В языке используются следующие операции отношения:

- = — равно;
- <> — не равно;
- < — меньше;
- > — больше;
- <= — меньше или равно;
- >= — больше или равно.

Операции отношения вычисляются с учетом того, что значения переменных задаются в виде интервала, для которого указываются нижняя и верхняя границы. В таблицах 1 – 8 приводятся формулы, по которым вычисляются результаты операций отношения. В этих формулах использованы следующие обозначения:

- ε — заданная точность вычислений;
- abs — абсолютное значение;
- ВГР(A) — верхняя (правая) граница интервала определения переменной A;
- НГР(A) — нижняя (левая) граница интервала определения переменной A.

Операция <

$r = A < B$, где A и B — целые или вещественные.

Таблица 1

Условие	Результат
$\text{ВГР}(A) < \text{НГР}(B)$	истина
$\text{ВГР}(B) <= \text{НГР}(A)$	ложь
другие случаи	неопределено

Операция <=

$r = A <= B$, где A, B — целые или вещественные

Таблица 2

Условие	Результат
$\text{ВГР}(A) <= \text{НГР}(B)$	истина
$\text{ВГР}(B) < \text{НГР}(A)$	ложь
другие случаи	неопределено

Операция =

$r = A = B$, где A, B — целые

Таблица 3

Условие	Результат
$\text{НГР}(A) = \text{НГР}(B)$ и $\text{ВГР}(A) = \text{ВГР}(B)$	истина
и $\text{НГР}(A) = \text{ВГР}(A)$	
$\text{ВГР}(A) < \text{НГР}(B)$ или $\text{НГР}(A) > \text{ВГР}(B)$	ложь
другие случаи	неопределено

$r = A = B$, где, по крайней мере, один из аргументов A, B — вещественный.

Таблица 4

Условие	Результат
$\text{abs}(\text{НГР}(A) - \text{НГР}(B)) < \varepsilon$ и $\text{abs}(\text{НГР}(A) - \text{ВГР}(B)) < \varepsilon$ и $\text{abs}(\text{НГР}(A) - \text{ВГР}(A)) < \varepsilon$	истина
$\text{НГР}(B) - \text{ВГР}(A) > \varepsilon$ или $\text{НГР}(A) - \text{ВГР}(B) > \varepsilon$	ложь
другие случаи	неопределено

Операция \diamond

$r = A \diamond B$, где A, B — целые

Таблица 5

Условие	Результат
$\text{ВГР}(B) < \text{НГР}(A)$ или $\text{ВГР}(A) < \text{НГР}(B)$	истина
$\text{НГР}(A) = \text{НГР}(B)$ и $\text{ВГР}(A) = \text{ВГР}(B)$ и $\text{НГР}(A) = \text{ВГР}(A)$	ложь
другие случаи	неопределено

$r = A \diamond B$, где, по крайней мере, один из аргументов A, B — вещественный.

Таблица 6

Условие	Результат
$\text{ВГР}(B) < \text{НГР}(A)$ или $\text{ВГР}(A) < \text{НГР}(B)$	истина
$\text{abs}(\text{НГР}(A) - \text{НГР}(B)) < \varepsilon$ и $\text{abs}(\text{ВГР}(A) - \text{ВГР}(B)) < \varepsilon$ и $\text{abs}(\text{ВГР}(A) < \text{НГР}(A)) < \varepsilon$	ложь
другие случаи	неопределено

Операция \geq

$r = A \geq B$, где A, B — целые или вещественные.

Таблица 7

Условие	Результат
$\text{НГР}(A) \geq \text{ВГР}(B)$	истина
$\text{НГР}(B) > \text{ВГР}(A)$	ложь
другие случаи	неопределено

Операция $>$

$r = A > B$, где A, B — целые или вещественные.

Таблица 8

Условие	Результат
$\text{НГР}(A) > \text{ВГР}(B)$	истина
$\text{НГР}(B) \geq \text{ВГР}(A)$	ложь
другие случаи	неопределено

Особенностью операций отношения в UniCalc является их двустороннее действие, т.е. не только значения операндов влияют на результат операции, но и известное значение результата операции может влиять на значения операндов. Таким образом, вычисление операций отношения происходит в обоих направлениях. Из приведенных выше формул видно, что для изменения значений операндов значение операции должно быть равно ИСТИНА или ЛОЖЬ.

Например, если отношение $A > 0$ истинно, то $\text{НГР}(A) > 0$, а если истинно отношение $X = 5$, то $\text{НГР}(X) = \text{ВГР}(X) = 5$.

Таким образом, если операция отношения равенства истинна, то это фактически равносильно операции присваивания. Перед началом вычислений значениям всех операций отношения самого верхнего уровня присваивается ИСТИНА.

Арифметические операции

- $+$ — операция сложения,
- $-$ — операция вычитания и унарный минус,
- $*$ — операция умножения,
- $/$ — операция деления,
- $^$ — операция возведения в степень,

В UniCalc используется общепринятый приоритет операций, т.е. в порядке убывания операции располагаются следующим образом: $^$, $/$, $*$, $-$, $+$. Фактически операции " $+$ " и " $-$ ", " $/$ " и " $*$ " имеют одинаковый приоритет. Для изменения приоритета операций используются

круглые скобки. Тип результата этих операций определяется в зависимости от типов операндов (см. табл. 9).

Таблица 9

Операция	Типы операндов	Тип результата
+	целый, целый	целый
$r = t1 + t2$	целый, веществ. веществ, веществ.	веществ. веществ.
–	целый, целый	целый
$r = t1 - t2$	целый, веществ. веществ., веществ.	веществ. веществ.
*	целый, целый	целый
$r = t1 * t2$	целый, веществ. веществ., веществ.	веществ. веществ.
/	целый, целый	веществ.
$r = t1 / t2$	целый, веществ. веществ., веществ.	веществ. веществ.
^	целый, целый	целый
$r = t1 ^ t2$	целый, веществ. веществ., веществ.	веществ. веществ.

Примеры

Применение операции

$$t = 1/2;$$

$$i = 1/2;$$

$$c = b^3; b = [-3,1]$$

Результат

$$t = 0.5$$

$$i = 0$$

$$c = [0, 1]; b = [0, 1]$$

Логические операции

OR — логическое «или»;

AND — логическое «и»;

NOT — логическое «не»;

\rightarrow — импликация.

Логические операции по приоритету располагаются следующим образом (в порядке убывания): NOT, AND, OR, \rightarrow .

Выражения системы не могут явно включать логические переменные и логические константы, но в систему могут входить логические выражения, включающие операции отношения, связанные логическими операциями.

В UniCalc используется трехзначная логика, т.е. результаты вычисления логических выражений могут принимать одно из трех значений: ЛОЖЬ, ИСТИНА, НЕОПРЕДЕЛЕНО.

Логические операции выполняются в соответствии с правилами, приведенными в таблицах 10 – 13.

Операция AND

$$r = a \text{ AND } b$$

Таблица 10

a \ b	ложь	истина	неопределено
ложь	ложь	ложь	неопределено
истина	ложь	истина	неопределено
неопределено	неопределено	неопределено	неопределено

Операция OR

$$r = a \text{ OR } b$$

Таблица 11

a\b	ложь	истина	неопределено
ложь	ложь	истина	неопределено
истина	истина	истина	истина
неопределено	неопределено	истина	неопределено

Операция NOT

$$r = \text{NOT } a$$

Таблица 12

a	r
ложь	истина
истина	ложь
неопределено	неопределено

Операция \rightarrow (импликация)

$$r = a \rightarrow b$$

Таблица 13

a\b	ложь	истина	неопределено
ложь	истина	истина	истина
истина	ложь	истина	неопределено
неопределено	неопределено	истина	неопределено

Наличие трехзначной логики следует учитывать при формулировании задачи: т.к. в UniCalc переменные определяются на интервалах, то значения операций отношения часто могут быть не определены. Например, если значением переменной X является интервал $[-5, 5]$, то значением неравенства $X > 0$ является НЕОПРЕДЕЛЕНО, т.е.

утверждения об истинности или ложности этого отношения несправедливы. Такие случаи следует учитывать, особенно при использовании операций "OR" и "->".

Логические операции, как и операции отношения, также имеют двусторонний характер, т.е. по значениям операндов вычисляется значение логической операции, а значение операции может использоваться для изменения значения операндов.

Отметим, что не для любой операции по значениям результата можно вычислить значения операндов. Если этого сделать нельзя, то операндам присваивается значение НЕОПРЕДЕЛЕНО. В то же время для некоторых операций по значению результата иногда можно однозначно вычислить значения операндов. Например, если значением операции AND является ИСТИНА, то значениями обоих операндов является ИСТИНА, а если значением операции OR является ЛОЖЬ, то значения операндов тоже ЛОЖЬ. Перед началом вычислений результатам всех логических операций самого верхнего уровня присваивается значение ИСТИНА.

Оператор присваивания

Оператор присваивания используется для инициализации переменных.

Проинициализировать переменную можно либо точным значением, либо интервалом. В первом случае переменная рассматривается как константа, т.е. она уже не может изменяться.

Во втором случае начальный интервал определения переменной задается двумя константными выражениями (нижней и верхней границей). В процессе решения задачи начальный интервал может изменяться, причем он может только стягиваться, вплоть до получения точного значения. Задание начального интервала необходимо в том случае, когда требуется определить область изменения функции или указать интервал определения переменной, относительно которой требуется решить задачу. Задание ограниченных начальных интервалов переменных очень часто ведет к существенному сокращению времени вычислений.

Примеры оператора присваивания:

```
x = 999;
y = [5 + 6 * 10, 999].
```

Циклы, суммы и произведения

Для лаконичной записи повторяющихся операций, а также для операций суммирования и произведения в язык введены оператор цикла **for** и операции суммирования **sum** и произведения **prod**. Они имеют следующую структуру:

```
for (<имя_счетчика> =
      <нач_знач>, <шаг>, <конеч_знач>; <отношение>)
```

sum (<имя_счетчика> =
<нач_знач>, <шаг>, <конеч_знач>; <выражение>)

prod (<имя_счетчика> =
<нач_знач>, <шаг>, <конеч_знач>; <выражение>),

где <нач_знач>, <шаг>, <конеч_знач> и <выражение> — алгебраические выражения, а <отношение> — произвольное отношение языка UniCalc или оператор цикла **for**. Каждый из элементов <отношение> и <выражение> называется *телом* соответствующей конструкции. Операции **sum** и **prod** сами являются арифметическими выражениями, допуская таким образом вложенные конструкции. Степень вложенности оператора **for**, а также операций **sum** и **prod** может быть произвольной.

Для всех этих конструкций выполняются следующие правила:

1. Выражения <нач_знач>, <шаг> и <конеч_знач> должны быть константами или константными выражениями (не обязательно целыми и положительными). Каждое из них может быть произвольным арифметическим выражением в единственном случае — когда конструкция находится в теле пользовательской функции. Однако при вызове такой функции все выражения после подстановки параметров должны принимать константные значения.

2. Каждый шаг итерационного процесса начинается с проверки выполнения условия:

<конеч_знач> < <нач_знач>, если <шаг> > 0,

<конеч_знач> > <нач_знач>, если <шаг> < 0.

Если условие выполняется, итерационный процесс заканчивается, иначе вычисляется значение тела, которое обрабатывается в соответствии со смыслом конструкции для значения счетчика, равного <нач_знач>. Затем <нач_знач> изменяется на значение <шаг> и происходит переход на следующий шаг итерации.

Если параметры <нач_знач>, <шаг> и <конеч_знач> такие, что итерационный процесс никогда не начнется, то итерационный процесс считается пустым. В этом случае для цикла **for** тело не будет выполнено ни разу, результатом операции **sum** будет 0, а результатом операции **prod** будет 1.

Примеры

а) На языке UniCalc выражение $sum1000 = \sum_{n=1}^{1000} \frac{1}{n^2}$ можно определить

через пользовательскую функцию f:

```
f(n) := sum(x = n, -1, 1; 1 / (x^2));
```

```
sum1000 = f(1000);
```

б) $p = \prod_{i=1}^{10} \sum_{j=1}^i a_j$ на языке UniCalc можно записать следующим образом:

```
p = prod(i = 1, 1, 10; sum(j = 1, 1, i; a[j]));
```

в) Инициализация единичной матрицы:

```
for(i = 1, 1, 10;
  for(j = 1, 1, 10;
    (i=j -> a[i,j]=1) and (i<>j -> a[i,j]=0)));
```

г) Примеры невыполнимых итерационных процессов:

Результат выполнения отношения

```
d = prod(i = 1, 1, -1; r*f[i]);
```

будет равен единице.

Результат выполнения следующей операции будет нулевым:

```
sum(j = 1, 1, -3; a[j]);
```

Функции

Для задания выражений, зависящих от параметров, и вычисления ряда общеупотребимых математических функций язык UniCalc включает средства для определения и использования функций. При этом все функции делятся на два класса: *стандартные* и *пользовательские*. Основное их различие состоит в том, что стандартные функции перед использованием описывать не требуется, а пользовательские функции должны быть описаны (т.е. требуется указать способ их вычисления).

Стандартные математические функции

В список стандартных функций включены наиболее часто используемые математические функции. Все эти функции, за исключением функций **max** и **min**, имеют один аргумент, и их результат — вещественное значение (в общем случае интервальное). Функции **max** и **min** имеют переменное число аргументов, а их результатом является целое значение, если все аргументы целые, и вещественное в противном случае.

Результатом функции **sign** является +1 для положительного аргумента, -1 для отрицательного и 0 для нулевого. Если аргументом является интервал, содержащий 0, то результатом этой функции будет интервал [-1, 1].

Результатом функции **lower(x)** является значение нижней границы интервала x , а результатом функции **upper(x)** — соответственно значение верхней границы интервала x . Эти функции не могут входить в выражения, а служат для присваивания некоторым переменным значения нижних и верхних границ интервалов.

Пример использования функций **lower** и **upper**.

```
d := lower(x);
```

$$s := upper(x);$$

$$f = d + sin(s);$$

Ниже приводится список стандартных функций языка решателя UniCalc. При записи имен стандартных функций прописные и строчные буквы не различаются.

$\sin(x)$ – синус;
 $\cos(x)$ – косинус;
 $\text{tg}(x)$ – тангенс;
 $\text{ctg}(x)$ – котангенс;
 $\text{asin}(x)$ – арксинус;
 $\text{acos}(x)$ – арккосинус;
 $\text{atan}(x)$ – арктангенс;
 $\text{sqrt}(x)$ – квадратный корень;
 $\ln(x)$ – натуральный логарифм;
 $\exp(x)$ – экспонента;
 $\text{abs}(x)$ – абсолютное значение;
 $\text{sign}(x)$ – знак числа;
 $\max(a, b, \dots, c)$ – максимальное значение;
 $\min(a, b, \dots, c)$ – минимальное значение;
 $\text{lower}(x)$ – нижнее значение
 $\text{upper}(x)$ – верхнее значение

Пользовательские функции

Все функции, которые Вы предполагаете использовать в своей задаче, должны быть описаны в начале Вашей системы выражений.

При описании пользовательской функции указываются имя функции, ее параметры и выражение, определяющее тело функции:

$$\langle \text{имя} \rangle (\langle \text{параметры} \rangle) := \langle \text{арифметическое выражение} \rangle ;$$

Число параметров должно быть больше нуля и не ограничивается сверху. Функции с переменным числом параметров не допускаются.

Примеры описания пользовательских функций:

$$f(x, y) := x^2 + y^2 - 1;$$

$$g(x, y) := \exp(x) * \sin(y) + \exp(y) * \sin(x);$$

Вызов функций:

$$f(23.1, z + 5) = 0;$$

$$g(x, 1) = 3;$$

Функция дифференцирования

UniCalc позволяет использовать операцию дифференцирования при формулировании задачи. С этой целью в решатель включена

операция символьного (формального) дифференцирования. Вызов этой операции осуществляется с помощью встроенной функции **dif**.

Функция **dif** последовательно дифференцирует заданное выражение по переменным, указанным в списке параметров. Каждой переменной из списка можно сопоставить натуральное число, означающее кратность производной по данной переменной. Если для переменной кратность дифференцирования не указана, то по этой переменной вычисляется первая производная. Результатом функции **dif** является продифференцированное выражение.

Список параметров функции разделяется запятыми, а каждый элемент списка записывается в виде:

<переменная дифференцирования> : <кратность>,

где *< кратность >* может быть опущена.

Примеры использования функции дифференцирования:

$x^2 * \text{dif}(3 * x^4 - 5 * x^3 + x^2 + 5 * x - 2, x: 2) + x * y < 0;$

$g \times 1 y 3(x, y) := \text{dif}(g(), x, y: 3);$

$\text{dif}(\exp(x) * \sin(y) + \exp(y) * \sin(x), x, y: 3) = 0;$

Комментарии

Любая последовательность символов, заключенная между комбинациями символов (* и *), является комментарием и может вставляться в любом месте программы. При трансляции все эти символы игнорируются.

Часто при отладке программы требуется удалить из модели некоторые выражения, а затем вставить их вновь. Наиболее легкий способ реализации состоит в объявлении соответствующих выражений комментариями. Для автоматизации процесса «закомментирования» и «раскомментирования» одного выражения в редакторе UniCalc предусмотрена специальная команда, вызываемая нажатием клавиши **F7**.

Формальное описание синтаксиса языка

В этом разделе приведено формальное описание синтаксиса языка решателя UniCalc на основе нотации Бэкуса-Наура. При этом используются следующие метасимволы, не являющиеся символами входного языка:

- $::=$ – для определения синтаксической конструкции;
- $|$ – для разделения альтернатив;
- $\{ \}$ – для возможного повторения заключенных в эти скобки объектов нуль или более раз.

Все синтаксические конструкции заключаются в угловые скобки.

<модель> ::= < выражение >; { < выражение >; }
 <выражение> ::= < логическое > | < описание функции > |
 < инициализация > | < простое выражение > |
 < оператор цикла > | < операция суммирования > |
 < операция произведения >
 < логическое > ::= < отношение > |
 < отношение > < логическая операция > < отношение > |
 NOT < логическое > | (< логическое >)
 < отношение > ::= < простое выражение > < операция отношения >
 < простое выражение >
 < простое выражение > ::= < слагаемое > | < знак > < слагаемое > |
 < простое выражение > < знак > < слагаемое >
 < слагаемое > ::= < множитель > |
 < слагаемое > < мультипликативная операция > < множитель >
 < мультипликативная операция > ::= * | /
 < множитель > ::= < переменная > | < константа без знака > |
 (< простое выражение >) | < вызов функции > | < степень >
 < переменная > ::= < имя переменной > | < переменная с индексом >
 < имя переменной > ::= < имя >
 < переменная с индексом > ::= < имя переменной > [< индексное выражение >]
 < индексное выражение > ::= < простое выражение > { , < простое выражение > }
 < имя > ::= < буква > { < буква > | < цифра > }
 < константа без знака > ::= < целое без знака > | < вещественное без знака >
 < целое без знака > ::= < цифра > { < цифра > }
 < вещественное без знака > ::= < целое без знака > . < целое без знака > |
 < целое без знака > . < целое без знака > E < порядок > |
 < целое без знака > E < порядок >
 < порядок > ::= < целое без знака > | < знак > < целое без знака >
 < знак > ::= + | -
 < вызов функции > ::= < имя функции > (< фактический параметр >
 { , < фактический параметр > }) | < дифференцирование >
 < имя функции > ::= < имя > | < имя стандартной функции >
 < фактический параметр > ::= < простое выражение > |
 < имя переменной > | < вызов функции >
 < имя стандартной функции > ::= cos | sin | tg | ctg | exp | ln |
 min | max | abs | sign | sqrt
 < дифференцирование > ::= dif (< функция дифференцирования > ,
 < параметр дифференцирования > { , < параметр дифференцирования > })
 < функция дифференцирования > ::= < простое выражение > | < имя функции > ()
 < параметр дифференцирования > ::= < имя переменной > { : < кратность > })
 < кратность > ::= < целое без знака >

< степень > ::= < множитель > < операция степени > < множитель >
 < операция степени > ::= ^
 < операция отношения > ::= < | <= | > | >= | <> | =
 < логическая операция > ::= and | or | ->
 < описание функции > ::= < заголовок функции > := < тело функции >
 < заголовок функции > ::= < имя функции >
 (< формальный параметр > {, < формальный параметр >})
 < формальный параметр > ::= < имя переменной >
 < тело функции > ::= < простое выражение >
 < инициализация > ::= < инициализация константой > |
 < инициализация интервалом > |
 < инициализация интервальной константой >
 < инициализация константой > ::= < переменная > = < константное выражение >
 < инициализация интервалом > ::= < переменная > = < интервал >
 < инициализация интервальной константой > ::=
 < переменная > = @< интервал >
 < интервал > ::= [< константное выражение >, < константное выражение >]
 < константное выражение > ::= < простое константное выражение > |
 <знак> < простое константное выражение > |
 <константное выражение>
 <знак операции> < простое константное выражение >
 < простое константное выражение > ::= < константа без знака > |
 (<константное выражение>)
 < оператор цикла > ::= for (<заголовок цикла> ; <отношение>)
 <заголовок цикла> ::= <инициализация константой>, <шаг>,
 <конечное значение>
 < операция суммирования > ::= sum (<заголовок цикла>; <выражение>)
 < операция произведения > ::= prod (<заголовок цикла>; <выражение>)
 < шаг > ::= < константное выражение >
 < конечное значение > ::= < константное выражение >
 < знак операции > ::= <знак> | < мультипликативная операция > |
 < операция степени >

Глава 4

Взаимодействие пользователя с решателем UniCalc

Главное меню

Главное меню содержит входы в соответствующие подменю:

- Файл
- Редактирование
- Поиск
- Вычисления
- Параметры
- Окно
- Справка

Подменю **Файл**, **Редактирование**, **Поиск**, **Окно** содержат стандартные команды интерфейса Windows. Поэтому в дальнейшем мы не будем описывать работу с ними.

Меню Вычисления

Запуск процесса вычислений осуществляется с помощью меню **Вычисления**. Оно содержит следующие команды:

Запустить	F9
Поиск корней	F8
Продолжить	Ctrl+F9
Искать дальше	Ctrl+F8

Команда Запустить

Команда **Запустить** используется для запуска на счет модели, текст которой находится в активном окне редактирования.

После запуска вычислений появляется окно *Вычисления*. В строке *Вычисления* этого окна указывается полное имя модели, а в строке *Время счета* — время счета в часах, минутах и секундах. В нижней части окна расположена кнопка **Отмена**, предназначенная для прерывания процесса вычислений. После нажатия этой кнопки на экране появится окно результатов с текущими значениями переменных. Прервав вычисления, можно переходить в другие окна и выполнять действия по редактированию. Для продолжения вычислений надо выполнить команду **Продолжить** (см. стр. 4-3). По успешном прохождении трансляции задачи и окончании процесса вычислений в окне *Вычисления* появится кнопка **ОК**. После нажатия этой кнопки полученные решения будут выведены в окно результатов.

Если система не имеет решений в области действительных чисел, то появится окно с сообщением «Несовместная система» и в окне решений будут показаны значения переменных перед обнаружением несовместности.

В случае если модель имеет синтаксические ошибки, появится окно *Сообщение* с описанием найденных ошибок. Каждое сообщение содержит номер ошибочной строки в тексте с описанием модели. Для перехода к ошибочному месту в тексте модели достаточно установить курсор на строку с сообщением об ошибке и нажать клавишу **Enter** или левую кнопку мыши. После исполнения команды курсор будет установлен на то место в тексте, где была обнаружена ошибка в момент трансляции.

Команда Продолжить

Команда **Продолжить** используется для продолжения счета после того, как процесс вычислений был прерван с помощью кнопки **Отмена**. Команда дублируется комбинацией клавиш **Ctrl + F9**. Если же будет нажата клавиша **F9**, то процесс вычислений будет запущен с самого начала и продолжение прерванных вычислений станет невозможным.

Если после прерывания вычислений произошла смена активного окна, то выполнение команды **Продолжить** приводит к тому, что окно, содержащее модель, для которой проводятся вычисления, становится активным.

Команда Поиск корней

Команда **Поиск корней** используется для запуска процедуры выделения точных корней из интервала значений, полученного в результате вычислений (Рис. Глава 4.1). Точным корнем по данной переменной считается интервал, у которого разность между правой и левой границами меньше ε , где ε — установленная точность вычислений (см. стр. 4-8).

При исполнении команды на экране появляется окно диалога, в котором имеются три поля выбора: поле с именем переменной, поле выбора переменной поиска и поле с направлением поиска.

В поле *Имя переменной* приводится весь список имен переменных модели, относительно которых можно проводить расчеты по уточнению корней. Переменная поиска может выбираться пользователем или автоматически.

Поиск корней осуществляется с использованием метода деления отрезка пополам. Он может проходить в двух направлениях: слева направо и справа налево. Поиск корней слева направо означает, что первый найденный корень будет минимальным. По умолчанию поиск ведется слева направо.

Рис. Глава 4.1. Окно диалога команды **Поиск корней**

Во время поиска корня на экране появляется окно *Поиск корней*, аналогичное окну *Вычисления* команды **Запустить**. Кроме уже описанных строк и кнопок оно имеет строку «Найдено корней: », в которой сообщается о количестве найденных корней.

После нахождения корня по выбранной переменной показываются текущие значения всех переменных системы и ищутся переменные, у которых правая и левая граница отличаются больше, чем на ε . Если таких переменных нет, то делается запрос о поиске следующего корня: «Искать следующий корень?». Если Вы хотите получить следующий корень, нажмите кнопку **ОК**, в противном случае нажмите кнопку **Отмена**. Если же переменные с неточными значениями еще остались и Вы находитесь в пользовательском режиме выбора переменной, то появляется запрос «Искать по другой переменной?». Если Вас интересуют точные значения других переменных модели, то нажмите кнопку **ОК**. После этого появится окно диалога команды **Поиск корня**, и Вы можете сделать очередной выбор имени переменной и направления поиска по выбранной переменной. При поиске по следующей переменной фиксируются значения предыдущих переменных поиска, и процесс деления интервала значений проводится для вновь выбранной переменной. Если в данном отрезке корней не существует, то происходит автоматический возврат к предыдущей переменной и процесс деления продолжается по ней. В дальнейшем переход к поиску по выбранным переменным осуществляется без дополнительного запроса.

В том случае, если Вам не нужны точные значения других переменных модели, то в запросе «Искать по другой переменной?» нажмите кнопку **Отмена**, после чего Вам будет задан вопрос о необходимости поиска следующего корня, описанный выше.

После того как все корни из данного интервала выделены, решатель выдает сообщение: «Корней больше нет!».

Если получено сообщение: «Корней нет!», то это означает, что данная система вообще не имеет действительных решений.

Корни, полученные с помощью команды **Поиск корней**, можно просматривать нажатием клавиш **Alt + F7** (листание назад по списку корней) и **Alt + F8** (листание вперед по списку корней).

Команда Искать дальше

Команда **Искать дальше** предназначена для продолжения процесса поиска корней после его прерывания.

Команда Предыдущее решение

Команда **Предыдущее решение** предназначена для просмотра предыдущего решения в окне результатов.

Команда Следующее решение

Команда **Следующее решение** предназначена для просмотра следующего решения в окне результатов.

Команда График

Команда График предназначена для графического вывода пользовательских функций, заданных явно и описанных в активном окне редактирования. Пользовательская функция, график которой необходимо вывести, должна иметь только один формальный параметр вещественного типа. В одно графическое окно можно выводить графики нескольких функций, при этом имя формальной переменной должно быть одним и тем же для всех пользовательских функций.

Перед построением графиков модель из активного окна редактирования запускается на счет и результаты вычислений выводятся в окно результатов. Таким образом, интервальные параметры, входящие в аналитическое представление пользовательских функций, при построении графиков будут принимать значения, которые им будут присвоены после вычислений. Графики интервальных функций изображаются двумя линиями – нижней и верхней границами, близкими по цвету. По желанию пользователя внутренняя область интервальной функции может быть залитой. После построения графика из него можно вырезать отдельные фрагменты с изменением масштаба, перемещать график по экрану и помещать график в буфер для последующего его копирования в другие документы.

После выбора команды **График** появится окно диалога **настройка графики** (см. Рис. 4.2).

Диалог настроек графики

Этот диалог содержит настройки для **графики**:

- **Список функций** из текущей модели для графического вывода (допускаются лишь явные функции от одного аргумента). В **Списке функций** можно использовать в качестве разделителей пробелы, запятые или точки с запятой;
- **Начальная точка**. Начальное значение аргумента функции или функций;
- **Конечная точка**. Конечное значение аргумента функции или функций;

- **Заполнять недоопределенные значения.** При включенной опции **Заполнять недоопределенные значения** недоопределенные значения отображаются областью с темными границами, заполненной светлым фоном. В противоположном случае недоопределенные значения отображаются верхней более светлой и нижней более темной границами;
- **Показывать предупреждения.** При включенной опции **Показывать предупреждения** в **Окно сообщений** выводятся сообщения о попытке изобразить функцию вне пределов ее области определения и др.

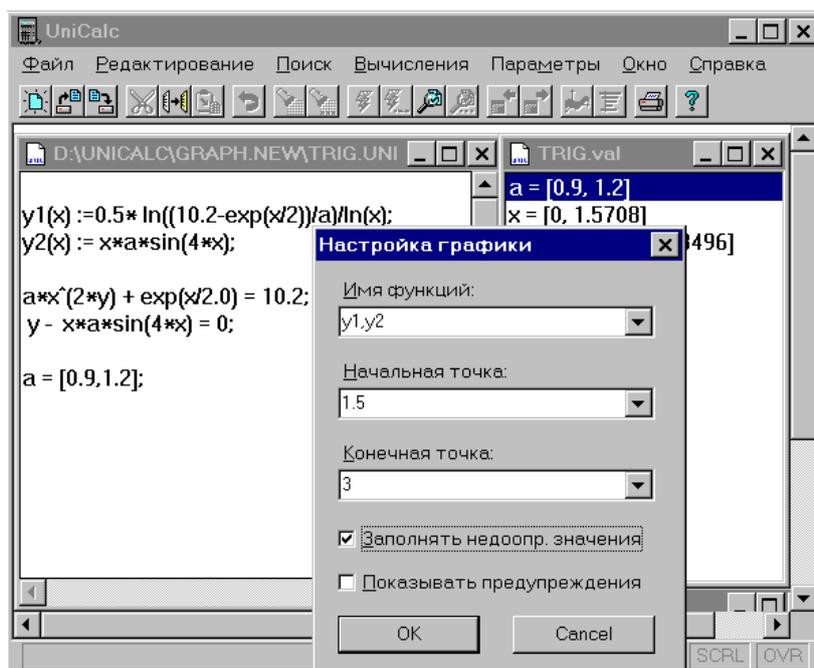


Рис. 4.2. Окно настройки графики

В начальный момент масштаб по вертикали выбирается автоматически, чтобы вместились графики всех функций.

Окно графика

После нажатия клавиши ОК в окне графических настроек вычисляются все значения функций для построения графиков, после чего открывается графическое окно, в котором рисуются все перечисленные пользователем функции. В правой части окна находится легенда графика – список функций с их цветами на графике (см. Рис. 4.3.).

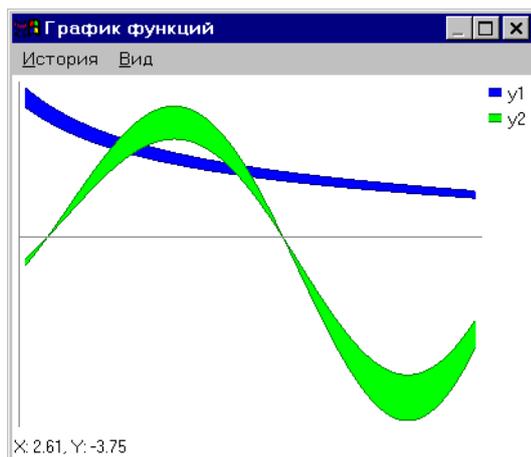


Рис. 4.3. Окно графики

Большую часть окна занимает график функций.

В верхней части окна находится МЕНЮ.

В нижней части находится строка, в которой показываются текущие координаты курсора, или, если курсор находится за пределами графика функций, координаты показываемой части графика. При масштабировании и сдвиге показываются координаты части графика, которая будет показываться после выполнения действия.

Функции рисуются одна поверх другой, в порядке, заданном в диалоге настроек. При включенной опции Заполнять недоопределенные значения, одни функции могут скрывать другие. Двойное нажатие на названии функции в правой части окна перерисует эту функцию поверх всех остальных.

Для постоянного изменения порядка рисования нужно изменить порядок в списке функций. Изменение масштаба графика осуществляется выделением мышкой области на графике. При нажатии на левую кнопку и выделении области произойдет увеличение – выделенный участок расширится на все окно. При нажатии на правую кнопку произойдет уменьшение – все, что было в окне, сожмется в выделенный участок, при этом будут пересчитаны масштаб и значения функций. Если удерживать клавишу Shift, то можно сдвинуть график не меня масштаб.

Изменения масштаба записываются в стек, по которому можно перемещаться либо с помощью меню, либо одиночными нажатиями на одну из кнопок мыши – нажатие на левую кнопку мыши означает перемещение вперед по стеку, на правую – назад.

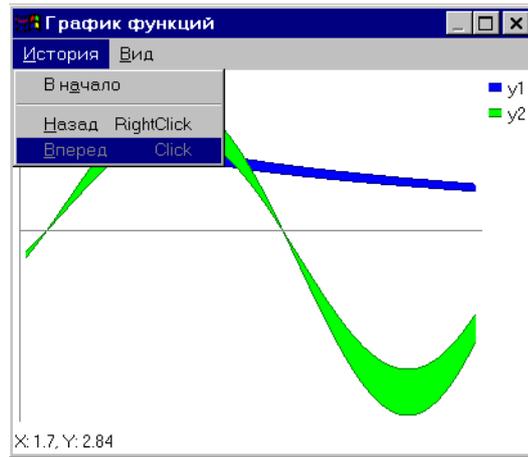
Меню графики

Меню окна графики содержит следующие команды:

История (см. Рис. 4.4):

Назад – взять предыдущий масштаб из стека

Вперед – взять следующий масштаб из стека

Рис. 4.4. Меню **История** графического окна**Вид**

Заполнять недоопределенные значения – переключение способа отрисовки недоопределенных данных.

Копировать в буфер – помещает в буфер обмена график с легендой и координатами окна.

Меню Параметры

Меню **Параметры** содержит следующие команды :

- Точность
- Диапазоны
- Степенная функция
- Аналитика
- Решения

Команда Точность

Команда **Точность** предназначена для задания точности вычислений. По этой команде на экране появляется окно с полем ввода, куда необходимо ввести значение точности. По умолчанию при вычислении границ интервалов используется точность 0.0001.

Изменяя значение точности, надо помнить, что увеличение точности влияет на скорость получения результатов. Чем выше заданная точность, тем больше итераций приходится совершать вычислителю для достижения конечного результата. Время счета может колебаться от нескольких секунд до нескольких минут.

Допустимые значения точности находятся в диапазоне от 0.1 до 10^{-13} .

Команда Диапазоны

Команда **Диапазоны** предназначена для задания областей определения переменных решаемой системы. По команде **Диапазоны** на экране появляется окно диалога, в котором требуется указать значения областей определения (см. Рис. 4.5).

Окно диалога содержит два поля:

- 1-й квадрант,
- Тригонометрические функции.

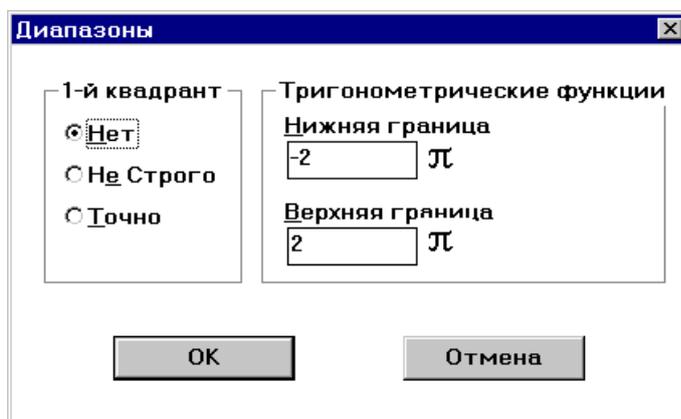


Рис. 4.5. Окно диалога команды **Диапазоны**

Поле *Первый квадрант* предназначено для задания области определения переменных и параметров системы. Выбор альтернативы **Нет** означает, что никакие дополнительные ограничения на систему не накладываются, т.е. переменные и параметры рассматриваются на всей действительной оси. Выбор альтернативы **Не строго** приводит к тому, что все переменные и параметры системы рассматриваются на неотрицательной полуоси. Задание этого режима равносильно добавлению к системе набора неравенств вида $x \geq 0$ для каждой переменной и каждого параметра. Если переменная задана в виде интервала, то ее левая граница определяется как максимум из нуля и заданной левой границы. В режиме **Строго** переменные и параметры рассматриваются на положительной полуоси.

Поле *Тригонометрические функции* предназначено для задания области определения аргументов тригонометрических функций. В UniCalc считается, что аргументы всех тригонометрических функций определены в одном диапазоне. Область определения задается в виде интервала, имеющего нижнюю и верхнюю границы, кратные числу π .

Ввод значений осуществляется в виде $x\pi$, где x — любое действительное число. Выбор подтверждается нажатием кнопки ОК, кнопкой Отмена можно отказаться от ввода. По умолчанию областью определения тригонометрических функций считается интервал $[-2\pi, 2\pi]$.

Команда **Степенная функция**

Команда **Степенная функция** предоставляет возможность интерпретировать выражение 0^0 следующими способами:

$$\begin{array}{ll} 0^0 = 1, & 0^0 = \text{противоречие,} \\ 0^0 = \text{бесконечность,} & 0^0 = [0, 1]. \end{array}$$

Команда Аналитика

Команда **Аналитика** дает возможность установить следующие режимы работы:

- решать линейные уравнения,
- показать упрощенную модель,
- не запускать вычисления.

Для решения системы линейных уравнений рекомендуется включить режим **Решать линейные уравнения**. Это позволяет сэкономить время счета и повысить точность решения.

В режиме **Показать упрощенную модель** создается окно, в которое выводится текст модели после аналитических преобразований.

Если Вас интересуют только результаты символьных преобразований (например упрощения, дифференцирования, трансформации линейной системы), то Вы можете сэкономить немного памяти и времени, установив режим **Не запускать вычисления**.

Команда Решения

Команда **Решения** предназначена для установки стилей записи и просмотра решений. По этой команде на экране появляется окно диалога *Настройки решения* (Рис. 4.6.).

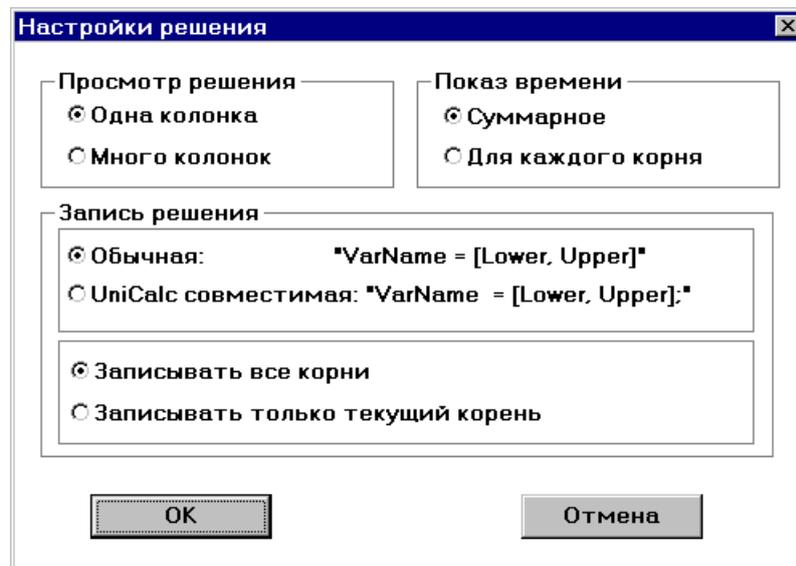


Рис. 4.6. Окно диалога команды **Решения**

Окно диалога содержит три поля:

- Просмотр решения,
- Запись решения,
- Показ времени.

Поле *Просмотр решения* содержит варианты представления решения в окне результатов в случае, когда список переменных, помещенных в одну колонку, выходит за рамки экрана, что соответствует

альтернативе **Одна колонка**. Выбор альтернативы **Много колонок** приводит к тому, что решение представляется в виде нескольких колонок переменных таким образом, чтобы их высота по возможности не превосходила высоту экрана.

Поле *Запись решения* определяет вид и содержание файла с результатами, создаваемого на диске. Выбор альтернативы **Обычная** означает, что между именами переменных и их значениями в этом файле ставится знак равенства. В UniCalc-совместимой форме в конце ставится точка с запятой. Такой файл в дальнейшем можно использовать для задания диапазонов переменных в моделях UniCalc. Выбор альтернативы **Записывать все корни** приводит к тому, что в созданном файле содержатся все найденные решения. Они располагаются одно под другим в порядке их нахождения. Перед каждым корнем стоит комментарий, указывающий номер корня. Выбор альтернативы **Записывать только текущий корень** означает, что в файл записывается только тот корень, который находится в окне результатов в момент записи.

Поле *Показ времени* определяет, какое время счета записывается в файл с результатами. Выбор альтернативы **Суммарное** означает, что после списка всех корней, сохраняемых в этом файле, приводится общее время счета, записанное в виде комментария. Выбор альтернативы **Для каждого корня** приводит к тому, что после каждого корня указывается время, затраченное на его вычисление.

Меню Справка

Если Вы не можете вспомнить, какое действие соответствует той или иной команде системы или затрудняетесь правильно написать математическое выражение, воспользуйтесь системой **Справка**.

Меню содержит три команды:

- Содержание,
- Использование Справки,
- О программе.

Команда Содержание

Команда **Содержание** предоставляет информацию о наименованиях разделов Справки.

Команда Использование справки

Команда **Использование Справки** предоставляет информацию о работе с разделами Справки.

Команда О программе

Команда **О программе** используется для получения общих сведений о решателе UniCalc (номере версии, разработчике, адресе электронной почты, адресе WWW-страницы, времени создания).

Локальные меню

Для удобства работы с решателем существуют локальные меню. Они появляются в рабочей области экрана при нажатии правой кнопки мыши либо нажатием *Alt+F10*.

Локальное меню для окна модели содержит команды:

- Показать решение,
- Показать упрощенную модель.

Если после вычислений Вы закрыли окно с решением или окно с упрощенной моделью, то при необходимости обращения к ним воспользуйтесь этими командами, чтобы не запускать вычисления снова.

Локальное меню для окна результатов содержит команды:

- Просмотр,
- Сохранить,
- Сохранить как,
- Предыдущее решение **Alt+F7**,
- Следующее решение **Alt+F8**,
- Редактировать исходный файл,
- Показать упрощенную модель,
- Параметры решения.

Команда **Просмотр** подразделяется на следующие команды:

- Показать только выбранные,
- Показать все,
- Одна колонка,
- Много колонок.

Команда **Показать только выбранные** показывает решения только для выбранных переменных (см. стр. 2-3).

Команда **Показать все** показывает все решения в окне результатов.

Команды **Одна колонка** и **Много колонок** соответствуют уже описанным командам поля **Просмотр решения** окна *Настройки решения* (см. стр. 4-10).

Команды **Сохранить** и **Сохранить как** осуществляют запись файла с решениями на диск.

С помощью команд **Предыдущее решение** и **Следующее решение** можно просматривать решения в окне результатов.

Командой **Редактировать исходный файл** осуществляется переход из окна результатов в окно модели.

Команда **Показать упрощенную модель** вызывает окно с упрощенной моделью.

Команда **Параметры решения** вызывает окно с настройками решения.

Локальное меню для окна упрощенной модели имеет вид:

- Редактировать исходный файл,
- Показать решение.

Командой **Показать решение** осуществляется переход из окна с упрощенной моделью в окно результатов.

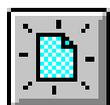
Локальное меню для окна сообщений:

- Редактировать исходный файл,
- Удалить все сообщения.

Последней командой удаляются все сообщения из окна *Сообщение*.

Пиктограммы

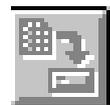
На рабочей панели решателя, под строкой главного меню, имеется строка пиктограмм — кнопок, эквивалентных некоторым командам главного меню. При указании стрелкой на пиктограмму в самой нижней строке панели появляется информация о ней. Использование пиктограмм удобно для ускорения действий. Строка пиктограмм содержит следующие кнопки:



Создание нового файла



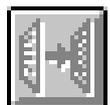
Открытие существующего файла



Сохранение файла на диске



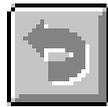
Перемещение выделенного фрагмента в буфер обмена и стирание его из модели



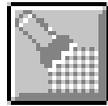
Перемещение копии выделенного фрагмента в буфер обмена



Вставка информации из буфера обмена



Отмена самого последнего из действий



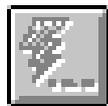
Контекстный поиск



Продолжить поиск



Запустить вычисления



Продолжить вычисления после прерывания



Запустить процесс поиска корней



Продолжить поиск корней после прерывания



Показать предыдущее решение



Показать следующее решение



Графика



Печать содержимого активного окна



Вызов справочной информации

Функциональные клавиши

Для работы с решателем UniCalc Вы можете использовать специальные клавиши, которые дублируют команды главного меню решателя и позволяют ускорить темп работы:

Клавиша	Действия
F1	вызов подсказки, содержащей краткие сведения для пользования решателем
F7	автоматическое закомментирование (раскомментирование) выражения
F8	запуск процесса поиска корней
F9	запуск процесса вычислений
F10	вход в главное меню решателя
Esc	выход из подменю в меню или выход из главного меню в окно редактора
PgUp, PgDn	просмотр результатов в окне результатов, просмотр файлов с описанием моделей, файла подсказки
←, →	переход к соседнему элементу главного меню, соседнему подменю или пунктам подсказки, сдвиг окна в режиме перемещения окна
↑, ↓	переход к соседнему элементу подменю, пунктам подсказки или просмотр результатов в окне результатов, сдвиг окна в режиме перемещения окна
Клавиша	Действия
Alt-F7	выход на предыдущую ошибку либо переход к предыдущему корню
Alt-F8	выход на следующую ошибку либо переход к следующему корню
Alt-буква	быстрый переход к подменю по ключевой букве (заглавная буква в имени действия или команды в подменю)
Ctrl-F4	закрыть активное окно
Ctrl-F6	переход из окна решений или упрощенной модели в окно уравнений и из окна уравнений в окно решений
Ctrl-F8	продолжить поиск корней
Ctrl-F9	продолжить счет
Ctrl-Ins	копировать отмеченный текст
Ctrl-Del	удалить весь текст
Ctrl- →, Ctrl- ←	быстрый переход к следующему или предыдущему идентификатору
Ctrl-K-H, Ctrl-H, Del	удалить отмеченный текст
Shift-Ins	вставить отмеченный текст
Shift-Del	забрать
Shift- ↑, Shift- ↓	отметка блока
Shift- ←, Shift- →	отметка блока

Глава 5

Некоторые рекомендации по решению задач

В этой главе на конкретных примерах будут даны рекомендации по решению задач в UniCalc. Особо будут выделены нестандартные ситуации, требующие интерактивного участия пользователя в процессе решения задачи. Это необходимо сделать для того, чтобы человек, привыкший работать со стандартными вычислительными пакетами, лучше разбирался в получаемых с помощью UniCalc результатах, правильно понимал область применимости решателя и мог осознанно предпринимать действия для получения необходимых результатов.

О существовании решения

Вопрос о существовании решения должен быть предметом изучения пользователя. В некоторых случаях UniCalc дает ответ на этот вопрос. Если система не имеет действительных корней, то решатель, как правило, выдает сообщение о несовместности системы. Иногда при отсутствии решения UniCalc может выдать для каждой переменной интервал, но никогда для несовместной системы не будет выдано точное значение в качестве решения.

Если для рассчитываемой математической модели вопрос существования решения не изучен или представляет значительные теоретические трудности, то с помощью UniCalc можно провести некоторые исследования в этой области. Например, можно провести расчеты для точно заданных значений параметров с подключением режима **Поиск корней**, и если решения не существует, то будет выдано сообщение о несовместности системы.

О решении, полученном на UniCalc

Математические модели с точными значениями параметров

Если математическая модель с точными параметрами имеет одно или более решений, то в качестве решения будет выдан многомерный параллелепипед, гарантированно содержащий все корни системы. В случае единственного решения этот параллелепипед, как правило, будет стянут в точку. Иногда для единственного решения UniCalc выдает интервальное значение, а для решения с несколькими корнями может быть выдан интервал, левая граница которого меньше, а правая соответственно больше любого из корней. Этот эффект связан с использованием интервальной математики, что приводит к расширению точных границ интервальной функции.

Если получено интервальное решение для системы с точными коэффициентами, нужно использовать встроенную процедуру **Поиск корней**. С помощью этой процедуры будут найдены все корни или выдано сообщение о несовместности системы.

Математические модели с интервальными параметрами

Системы уравнений с интервальными параметрами имеют, как правило, интервальное решение. UniCalc решает внешнюю интервальную задачу, т.е. определяет область допустимых значений, в которой лежит решение. Внутри полученной области решения могут существовать подобласти, не удовлетворяющие математической модели. Полностью исключить подобласти, где решения не существует, не всегда удается даже с использованием процедуры **Поиск корней**, так как область решения не всегда является интервалом, прямоугольником или параллелепипедом.

Ниже на конкретных примерах и более подробно будет показано использование процедуры **Поиск корней** для нахождения одного корня и в случае существования нескольких корней.

Зависимость решения от вида представления системы

Близость полученного в результате вычислений параллелепипеда к наименьшему, содержащему все решения, зависит от вида представления системы. Этот эффект является следствием применения интервальной арифметики, в которой нарушается закон дистрибутивности. Так, задание выражения в виде произведения нескольких сомножителей, зависящих от одних и тех же переменных, приведет к получению достаточно широких интервалов. Ниже приводятся примеры решения полиномиального уравнения, имеющего один действительный корень, для различных форм его представления.

Форма представления полинома	Решение, выдаваемое UniCalc
a. $x \cdot (6 \cdot x^3 + 1) \cdot (4 \cdot x - 1) = 4$	[-1185630, 1185630]
b. $(6 \cdot x^4 + x) \cdot (4 \cdot x - 1) = 4$	[-35930.4, 35930.4]
c. $6 \cdot x^4 \cdot (4 \cdot x - 1) + 4 \cdot x^2 - x = 4$	[-56234.1, 56234.1]
d. $24 \cdot x^5 + 4 \cdot x^2 - x \cdot (6 \cdot x^3 + 1) = 4$	[-0.286474, 0.782224]
e. $24 \cdot x^5 + 4 \cdot x^2 - 6 \cdot x^4 - x = 4$	[-0.43906, 0.782224]
g. $24 \cdot x^5 - x \cdot (6 \cdot x^3 - 4 \cdot x + 1) = 4$	[-0.324672, 0.812077]

Результаты этих экспериментов хорошо иллюстрируют вышесказанное. Раскрытие произведений существенно сужает интервал, в котором содержится решение.

Классы задач, для которых могут быть получены слишком широкие интервалы

Если в процессе вычислений получены слишком большие интервалы, то может потребоваться этап корректировки модели. На этом этапе можно на основе полученных результатов уточнить начальные данные и наложить дополнительные ограничения, после чего

повторно запустить процесс вычислений, а затем сравнить полученные результаты. Ограничения формулируются в виде неравенств, уравнений или логических связок, добавляемых к исходной системе (можно, например, ввести условие на знак некоторого выражения и т.п.). Для математических моделей с точно заданными значениями параметров рекомендуется подключить процедуру **Поиск корней**.

Приведем примеры классов систем, для которых могут быть получены достаточно широкие интервалы:

- системы уравнений, области определения которых содержат особые точки;
- системы уравнений, в которых одна из частей записана в виде:

$$f(x)(x - c_1)(x - c_2) \dots (x - c_k),$$
 где $f(x)$ — произвольная функция, а c_1, c_2, \dots, c_k — константы.
- система линейных алгебраических уравнений, записанная без диагонального преобладания;
- некоторые квазилинейные системы;
- слабосвязанные системы;
- поиск минимума функционала, записанного по формулам метода наименьших квадратов.

Вычисление корней полиномов

При нахождении действительных корней полиномов, если полином записан в виде:

$$a_0 x^n + a_1 x^{n-1} + \dots + a_{n-1} x + a_n,$$

UniCalc всегда стягивает бесконечный интервал до конечного, в котором содержатся все корни полинома. Полученный интервал полностью содержится в интервале, нижняя и верхняя границы которого удовлетворяют известной оценке границ полинома

$$- \left[-1 - \left(\frac{q_1}{a_0} \right)^{\frac{1}{k}}, 1 + \left(\frac{q_2}{a_0} \right)^{\frac{1}{k}} \right],$$

где k — номер первого отрицательного коэффициента, q_1 — наибольшее из значений положительных коэффициентов a_i , q_2 — наибольшая из абсолютных величин отрицательных коэффициентов a_i . Для того чтобы выделить корни, необходимо либо применить процедуру **Поиск корней**, либо использовать дополнительные ограничения.

Замечание 1. В случае полиномов рекомендуется всегда раскрывать старший член, т.е. член с максимальной степенью (см. стр. 5-3).

Замечание 2. При нахождении корней полиномов очень высокой степени необходимо помнить, что в UniCalc существует свое максимальное вещественное число (в настоящей версии это 10^{19}). Это может привести к потере корня.

Рассмотрим два полинома:

$$x^{19} - 12x^{18} - x + 12 \quad \text{и} \quad x^{19} - 10x^{18} - x + 10.$$

Корнем первого полинома является $x = 12$, т.к.

$$x^{19} - 12x^{18} - x + 12 = (x - 12)(x^{18} - 1),$$

корнем второго — $x = 10$, т.к.

$$x^{19} - 10x^{18} - x + 10 = (x - 10)(x^{18} - 1).$$

Если запустить на счет первый полином, то получим

$$x = [-1.00941, 1.00941].$$

Число 12 не вошло в интервал результата вычислений. Это связано с тем, что любое промежуточное значение не должно превышать 10^{19} , т.е. из неравенства $x^{19} \leq 10^{19}$ следует, что $x \leq 10$. Поэтому и произошла потеря корня.

При запуске на счет второго полинома получаем интервал $[-1.13524, 10]$, который содержит все действительные корни.

Замечание 3. В некоторых случаях получается эффект «ползущего корня», т.е. вместо одного корня UniCalc выдает несколько корней, отличающихся между собой на величину, меньшую пользовательской точности. Эта ситуация возникает в следующих случаях:

- корень является кратным,
- ось абсцисс является касательной к графику полинома,
- если на интервале длины порядка $O(1)$ полином изменяется в пределах величины, не превышающей 0.01, и на этом интервале существует более одного корня.

Решение систем линейных уравнений

При решении систем линейных уравнений с точно заданными коэффициентами нужно установить режим **Аналитика / Решать линейные уравнения**.

Есть один класс систем линейных уравнений, которые UniCalc может решить и без установки этого режима — системы линейных алгебраических уравнений с диагональным преобладанием. Это системы, у которых на главной диагонали матрицы коэффициентов стоят значения, абсолютная величина которых не меньше суммы абсолютных величин коэффициентов соответствующей строки, причем хотя бы в одном случае должно быть строгое неравенство.

Рассмотрим систему линейных уравнений Гильберта:

$$\frac{x_1}{1} + \frac{x_2}{2} + \frac{x_3}{3} + \frac{x_4}{4} + \frac{x_5}{5} + \frac{x_6}{6} + \frac{x_7}{7} = 7$$

$$\frac{x_1}{2} + \frac{x_2}{3} + \frac{x_3}{4} + \frac{x_4}{5} + \frac{x_5}{6} + \frac{x_6}{7} + \frac{x_7}{8} = 5.2821428571428$$

$$\frac{x_1}{3} + \frac{x_2}{4} + \frac{x_3}{5} + \frac{x_4}{6} + \frac{x_5}{7} + \frac{x_6}{8} + \frac{x_7}{9} = 4.3420634920634$$

$$\frac{x_1}{4} + \frac{x_2}{5} + \frac{x_3}{6} + \frac{x_4}{7} + \frac{x_5}{8} + \frac{x_6}{9} + \frac{x_7}{10} = 3.713095238095$$

$$\frac{x_1}{5} + \frac{x_2}{6} + \frac{x_3}{7} + \frac{x_4}{8} + \frac{x_5}{9} + \frac{x_6}{10} + \frac{x_7}{11} = 3.25382395382$$

$$\frac{x_1}{6} + \frac{x_2}{7} + \frac{x_3}{8} + \frac{x_4}{9} + \frac{x_5}{10} + \frac{x_6}{11} + \frac{x_7}{12} = 2.9006132756$$

$$\frac{x_1}{7} + \frac{x_2}{8} + \frac{x_3}{9} + \frac{x_4}{10} + \frac{x_5}{11} + \frac{x_6}{12} + \frac{x_7}{13} = 2.619197469$$

Это классический пример линейной системы, плохо решаемой обычными методами.

Решение этой системы:

$$x_1 = 1, x_2 = 2, x_3 = 3, x_4 = 4, x_5 = 5, x_6 = 6, x_7 = 7.$$

Для этой системы не выполнено условие диагонального преобладания, поэтому UniCalc также не может решить эту систему без использования символьных преобразований и качестве ответа выдает бесконечные интервалы. Установка режима **Решать линейные уравнения** позволяет получить точные решения.

Замечание. С помощью режима **Решать линейные уравнения** можно решать только системы линейных алгебраических уравнений с точно заданными коэффициентами. Для систем с интервальными коэффициентами этот режим использовать нельзя.

Решение обратных задач

Вычислительный алгоритм в UniCalc организован таким образом, что нет строгого деления на входные и выходные параметры. Математическая модель трансформируется в вычислительную сеть, над которой производятся вычисления. Поэтому с помощью UniCalc можно решать как прямые, так и обратные задачи. Если обратная задача поставлена некорректно, то исходные интервальные значения переменных не будут стянуты.

Рассмотрим задачу восстановления коэффициентов функции по известным значениям функции и ее первых двух производных в некоторых точках. Пусть эти значения даны с некоторой погрешностью. Требуется найти интервалы значений параметров, для которых известны лишь пределы их изменения. Подобного рода задачи часто возникают при обработке результатов наблюдений и измерений,

например при поиске параметров регрессионных зависимостей. Выберем для рассмотрения функцию

$$f(x) := a * x^k + b * \exp(0.1 * x + 1.2) - c.$$

Тогда описанная выше задача на языке UniCalc будет выглядеть так (см. файл с примером POL_REG.UNI):

```
f(x) := a * x^k + b * exp(0.1*x + 1.2) - c;
f1(x) := a * k * x^(k-1) + 0.1*b * exp(0.1*x + 1.2);
f2(x) := a*k*(k-1)*x^(k-2) + 0.01*b*exp(0.1*x + 1.2);
a = [1,100]; b = [2,16]; c = [0,100]; k = [1,10];
f(2.13) = t1; f(14.91) = t6;
f1(2.13) = t2; f1(4.06) = t4;
f2(2.13) = t3; f2(10.28) = t5; f2(14.91) = t7;
t1 = [10.8,10.9]; t2 = [32.5,32.7]; t3 = [29.4,29.6];
t4 = [115.2,115.4]; t5 = [142.1,142.3];
t6 = [7646.1,7646.3]; t7 = [206.1,206.3];
```

После запуска вычислений UniCalc выдает следующее решение:

```
a = [2.29865, 2.30134]; b = [2.86548, 3.44134];
c = [23.0854, 25.5771]; k = 3;
```

Решение тригонометрических уравнений

По умолчанию область определения тригонометрических функций есть интервал $[-2\pi, 2\pi]$. Если значения аргументов функций лежат вне области определения, то решателем выдается сообщение о несовместности системы. Для преодоления этой ситуации следует с помощью опции **Диапазоны** в меню **Параметры** расширить область определения тригонометрических функций.

Приведем несколько примеров, иллюстрирующих использование опции **Диапазоны**.

Пример 1

Пусть необходимо решить уравнение:

$$a = \sin(7x);$$

$$x = 1;$$

Если область определения тригонометрических функций не содержит числа 7 (например $[-2\pi, 2\pi]$), UniCalc выдаст сообщение «Система несовместна». Для получения верного ответа достаточно правую границу увеличить до 3π .

Пример 2

Пусть в опциях **Диапазоны** для тригонометрических функций установлен интервал $[-2\pi, 2\pi]$.

Для уравнения

$$\sin(x) = 1$$

будет получен ответ $x = [-4.71239, 1.5708]$, а для уравнения

$$\sin(6x) = 1$$

получим значение $x = [-0.785398, 0.261799]$. Последнее связано с тем, что аргумент $6x$ изменяется в пределах $[-2\pi, 2\pi]$.

Пример 3

Рассмотрим уравнение (см. файл с примером SINSUM.UNI):

$$\sin(x) + \sin(2x) + \sin(3x) + \sin(4x) + \sin(5x) + \sin(6x) = 0$$

В интервале $-2\pi < 6x < 2\pi$ это уравнение имеет решение:

$$x_1 = -1.0471975, \quad x_2 = -0.8975979, \quad x_3 = 0,$$

$$x_4 = 0.8975979, \quad x_5 = 1.0471975.$$

После запуска на счет с точностью 10^{-7} UniCalc выдает следующий интервал для значения x : $[-1.04719755, 1.04719755]$. Далее можно выделить отдельные решения с помощью механизма поиска корней или изменением диапазона определения тригонометрических функций.

Решение задач оптимизации

Специализированных средств решения задач оптимизации в UniCalc нет, но тем не менее с его помощью можно решить многие оптимизационные задачи. Универсальным средством решения таких задач является *механизм поиска корней*. Начинать поиск корней нужно с имени той переменной, по которой ищется оптимальное решение, причем для нахождения минимума поиск необходимо начинать слева направо, а для нахождения максимума — справа налево. На все дальнейшие запросы «Искать по другой переменной?», следует отвечать «да», при этом далее уже не важно, по какой переменной продолжать поиск и в каком направлении — слева или справа. Удачный выбор переменной и направления поиска может сократить время вычислений, но на результат не повлияет.

Рассмотрим задачу нахождения точек минимума для функции двух переменных (файл с примером OPT_COS.UNI):

$$f(x, y) := \frac{(x^2 + y^2)}{200} - \cos(x) \cos\left(\frac{y}{\sqrt{2}}\right) + 1.$$

Для нахождения точек минимума функции двух переменных в математическую модель расчетов нужно добавить необходимые условия: равенство нулю первых частных производных и условие положительной определенности квадратичной формы.

Запишем эту модель на языке UniCalc:

```
f(x,y) := (x^2 + y^2)/200 - cos(x)*cos(y/(2^0.5))+1;
fx(x,y) := dif(f(),x); fy(x,y) := dif(f(),y);
fxx(x,y) := dif(f(),x:2); fyy(x,y) := dif(f(),y:2);
fxy(x,y) := dif(f(),x,y);
det = fxx(x,y)*fyy(x,y) - fxy(x,y)^2;
(*необходимые условия существования минимума:*)
fx(x,y)=0; fy(x,y)=0; det>0; fxx(x,y)>0;
fmin = f(x,y);
```

После запуска на счет получим следующие интервалы:

```
fmin = [1,1.60218], x = [-6.28319,6.28319],
y = [-8.88577,8.88577]
```

Запустив поиск корня по переменной f_{min} слева (ищется минимум!), получим:

```
fmin = [1,1.0001]
x = [-6.28319,6.28319]
y = [-8.88577, 8.88577].
```

Это действительное решение, и оно единственно. Если ответить «да» на запрос «Искать следующий корень?», то UniCalc выдаст значение $f_{min} = [1.0001, 1.0002]$, а значения x и y будут интервальными. Вести дальнейший поиск не имеет смысла, так как значение f_{min} будет не меньше 1.0001. Из этого следует, что других точек минимума нет.

Приведем пример задачи Голомба. Эта задача целочисленной оптимизации в комбинаторике используется в радио-астрономических приложениях. Требуется найти минимальное значение последнего члена монотонно возрастающей целочисленной последовательности, такой что разности между любыми двумя членами этой последовательности должны отличаться друг от друга. Рассмотрим задачу для последовательности, состоящей из 5 членов. На языке UniCalc эта задача запишется следующим образом (файл с примером GOLOMB5.UNI):

```
golomb(n) := for( i = 1,1,n-1; ix[i] < ix[i+1])
and ix[1]=0 and ix[2]-ix[1] > ix[n]-ix[n-1]
and
for( i1=1,1,n-1;
for( i2=i1+1,1,n;
```

```
for( i3=i1+1,1,n-1;
    for( i4=i3+1,1,n;
        i4>i2 -> ix[i1]-ix[i2] <> ix[i3]-ix[i4])));
golomb(7);
```

Так как нам необходимо найти минимальное значение i_7 , запустим поиск корней по переменной i_7 слева. После запроса «Искать по другой переменной?» ответим «Да». Поскольку дальнейшая стратегия поиска не влияет на значение i_7 , воспользуемся режимом автоматического поиска корней. Первый же найденный корень будет искомым оптимальным решением:

```
ix[1] = 0
ix[2] = 3
ix[3] = 4
ix[4] = 12
ix[5] = 18
ix[6] = 23
ix[7] = 25
```

Заметим, что решение для i_2, i_3, i_4, i_5, i_6 не единственно и оно зависит от последовательности выбора переменных, по которым осуществляется поиск корня, а также от направления поиска. Однако последнее не важно, так как мы ищем минимальное значение для i_7 , которое единственно и не зависит от выбранной стратегии поиска.

Глава 6

Сообщения системы

Сообщения UniCalc

При обнаружении ошибок в записи формул или в процессе счета пользователю выдаются сообщения об ошибках. Ниже приводится список выдаваемых сообщений с указанием возможных причин возникновения ошибок. При обнаружении синтаксической ошибки нужно исправить ее с помощью средств редактора. Для ошибок вычислений приводятся рекомендации по их устранению.

Синтаксические ошибки

Деление на нуль

В системе присутствует знаменатель, обращающийся в нуль.

Индексы должны быть целыми

Не целое значение индексного выражения.

Мало операндов

Ошибка в записи стандартных математических функций **max** или **min**: количество аргументов функции меньше двух.

Неверное выражение for (...)

Нарушен синтаксис языка в описании оператора цикла **for(...)**.

Неверное выражение prod (...)

Нарушен синтаксис языка в описании операции **prod(...)**.

Неверное выражение sum (...)

Нарушен синтаксис языка в описании операции **sum(...)**.

Недифференцируемое выражение

Нельзя дифференцировать функции **sign**, **min**, **max**.

Недопустимая точность

Заданная точность выходит за пределы допустимого диапазона ($0.1 — 10^{-13}$).

Неконстантное выражение

При инициализации используется неконстантное выражение.

Некорректная инициализация

В интервальной константе, задающей начальное значение, правая граница меньше левой или синтаксическая ошибка.

Неправильное определение функции

В описании функции отсутствует оператор присваивания.

Несоответствие числа аргументов определению функции

При обращении к пользовательской функции число фактических параметров не соответствует числу формальных, указанных в описании функции.

Ожидалось неотрицательное число

Требуется неотрицательное число.

Ожидалось положительное число

Требуется положительное число.

Ожидается символ ","

При задании интервала инициализации для переменной пропущен разделительный знак ", ".

Ошибка в выражении

Некорректно составлено выражение (ошибка в имени функции или нарушен синтаксис языка); система не нашла определения функции, возможно, оно стоит после обращения к функции.

Плохой символ

Встретился незнакомый символ.

Повторное определение функции

Множественное определение функции.

Пропущен символ ";"

В конце выражения отсутствует символ ";" либо допущена ошибка в записи выражения (использован недопустимый знак операции, отсутствует символ "(" или "[" и т.д.).

Пропущен символ ")"

Нечетное количество круглых скобок.

Пропущен символ "]"

Нечетное количество квадратных скобок.

Слишком длинное индексное выражение

Длина индексного выражения вместе с длиной имени массива превышает 32 символа.

Слишком много ошибок

Количество синтаксических ошибок превысило число 10.

Слишком сложное выражение

Рекомендуется разбить его на более простые.

Типы операндов не соответствуют операции

Используемая операция не применима для данного типа операндов.

Функция не определена

Ошибка возникает при обращении к пользовательской функции, которая не определена или определяется после обращения к ней.

Ошибки в процессе вычислений**Недостаточно свободной памяти**

Для получения окончательного результата не хватило оперативной памяти.

Рекомендация: сделать систему более компактной.

Несовместная система

Система несовместна (уравнение не имеет решений или имеет комплексные корни) либо аргументы тригонометрических функций лежат вне заданного интервала.

Рекомендация: проверить правильность записи системы; провести анализ описанной модели на совместность; воспользоваться рекомендациями по решению задач (см. главу Глава 5); увеличить диапазон определения тригонометрических функций (см. стр. 5-7).

Переполнение целого

Целое число вышло за допустимые границы (от -2147483647 до $+2147483647$).

Приложение А

Справочник по пользовательскому интерфейсу

Пользовательский интерфейс решателя UniCalc оформлен в виде системы «спускающихся» меню. В таблице приведены команды интерфейса и действия, которые выполняются при использовании этих команд.

В системе возможен альтернативный метод задания команд или действий — нажатием определенной функциональной клавиши либо сочетания клавиш. Таблица функциональных клавиш приведена на стр. 4-14.

Таблица 14

Меню	Команда меню	Действие
Файл	Создать	Создать рабочий файл
	Открыть	Считать файл из архива
	Сохранить	Сохранить редактируемый файл
	Сохранить Как...	Сохранить редактируемый файл под новым именем
	Печать	Вывести на печать файлы с результатами и моделью
	Выбор Принтера	Выбрать принтер, на котором будет выполняться печать
	Выход Alt+F4	Выход из UniCalc
Редактирование	Отменить Alt+BkSp	Отменить последнюю команду
	Вырезать	Запомнить фрагмент, удалив его из текста
	Копировать	Запомнить фрагмент, не удаляя его из текста
	Вставить	Вставить фрагмент в редактируемый текст
	Очистить все...	Очистить экран
	Удалить	Удалить выделенный текст
Поиск	Найти...	Найти фрагмент текста
	Заменить	Заменить фрагмент
	Повторить Поиск	Повторить предыдущую команду (Найти или Заменить)

Меню	Команда меню	Действие
Вычисления	Запустить F9	Запустить модель на счет
	Продолжить Ctrl+F9	Продолжить счет с прерванного места
	Поиск корней F8	Запустить процедуру выделения корней из интервала значений, полученных в результате вычислений
	Искать дальше Ctrl+F8	Продолжить поиск корней после прерывания процесса поиска клавишей Esc
Параметры	Точность	Задать точность вычислений
	Диапазоны	Задать интервалы определения параметров модели
	Степенная функция	Определить значение 0^0
	Аналитика	Задать режим решения
	Решения	Настроить показ и запись решения, показ времени счета
Окно	Каскад	Расположить все открытые окна уступами
	Мозаика	Расположить все открытые окна так, чтобы они не перекрывались
	Упорядочить значки	Упорядочить значки свернутых окон
	Закреть Все	Закреть все открытые окна
	Решение	Показать окно результатов
	Упрощенная модель	Показать окно с упрощенной моделью
	Сообщение	Открыть окно сообщений
Справка	Содержание	Выдать содержание Справки
	Использование Справки	Вызвать информацию о пользовании Справкой
	О программе	Высветить заставку об UniCalc

Приложение Б

Примеры решения задач

Процесс решения задач с помощью UniCalc можно продемонстрировать на следующих примерах.

Пример 1

В системе UniCalc нет специализированных средств для решения задач оптимизации и поиска экстремальных точек, однако в ряде случаев можно найти решение для математической модели в исходной постановке с помощью процедуры поиска корней.

Рассмотрим модель, записанную на языке решателя UniCalc (файл с примером INTOPTIM.UNI):

(* Задача целочисленной оптимизации.

Необходимо найти минимум целевой функции

$$ihval = 20 * i + 25 * j + 10 * k + 12 * l;$$

Для его нахождения следует использовать поиск корней по переменной *ihval*. Известное решение: *ihval* = 22.

*)

$$ifval = 3 * i + 6 * j + 3 * k + 5 * l;$$

$$ifval \geq 8;$$

$$igval = 2 * i + 3 * j + 4 * k + 4 * l;$$

$$igval \geq 7;$$

(* Целевая функция *)

$$ihval = 20 * i + 25 * j + 10 * k + 12 * l;$$

$$i = [0, 1];$$

$$j = [0, 1];$$

$$k = [0, 1];$$

$$l = [0, 1];$$

Приведенная модель описывает задачу, в которой требуется найти такие значения параметров *i*, *j*, *k*, *l*, при которых функция *ihval* принимает минимальное значение. При этом параметры равны либо 0, либо 1 и должны удовлетворять неравенствам $ifval \geq 8$ и $igval \geq 7$.

Запустив модель на счет, получим интервалы, ограничивающие решение:

$$i = [0, 1];$$

$$ifval = [8, 17];$$

$$igval = [7, 13];$$

$$ihval = [0, 67];$$

$$j = [0, 1];$$

$$k = [0, 1];$$

$$l = [0, 1].$$

Подключим поиск корней по переменной *ihval* слева направо. В результате получим:

```
i = 0;
ifval = 8;
igval = 8;
ihval = 22;
j = 0;
k = 1;
l = 1.
```

Первый найденный корень слева и есть минимум функции цели. Так как решение задачи найдено, то на запрос «Искать следующий корень?» можно нажать кнопку Отмена.

Пример 2

Свободно падающий с высоты $h = 2$ метра шарик ударяется о неподвижно наклонную плоскость, расположенную под углом α к горизонту. Известно, что угол наклона α находится в интервале от 25 до 35 градусов, а коэффициент восстановления A_k лежит в диапазоне 0.75 – 0.85.

Определить возможные направления скорости шарика в конце удара, при которых высота подъема после удара h_1 будет превосходить 0.5 метра.

Используя теорему об изменении количества движения, запишем соответствующую систему уравнений и неравенств. Эта система на языке решателя выглядит следующим образом (файл с примером FALL.UNI):

```
V = sqrt(2 * @g * h);
Vt = V * sin(alpha);
Un = -Ak * V * cos(alpha);
Ut = Vt;
ctg(beta) = abs(Un) / abs(Vt);
(Ut^2 + Un^2) * cos(alpha + beta)^2 = 2 * @g * h1;
h1 > 0.5;
```

(* Инициализация *)

```
alphagr = [25, 35];
Ak = [0.75, 0.85];
h = 2; (* метра *)
betagr = [0, 90];
```

(* Формула перевода из радиан в градусы *)

```
convert(angle) = angle * 180 / @pi;
alphagr = convert(alpha);
betagr = convert(beta);
```

В результате решения системы мы находим для каждой интересующей нас величины интервал допустимых значений. Особо следует

отметить изменение начальных интервалов для Ak и $alfa$ ($alfagr$): UniCalc уточняет и исходные данные, сужая их до диапазонов, при которых задача имеет решение.

$$\begin{aligned} Ak &= [0.782877, 0.85]; \\ Un &= [-4.82486, -4.44385]; \\ Ut &= [2.64691, 2.88065]; \\ V &= 6.26311; \\ Vt &= [2.64691, 2.88065]; \\ alpha &= [0.436332, 0.477927]; \\ alphagr &= [25, 27.3832]; \\ beta &= [0.501766, 0.54336]; \\ betagr &= [28.7491, 31.1323]; \\ h &= 2; \\ h1 &= [0.5, 0.562958]. \end{aligned}$$

Пример 3

Система, содержащая уравнения, неравенства и логические выражения (знак ' \rightarrow ' означает импликацию):

$$\begin{aligned} x^2 + 6.0x &= y - 2^k; \\ kx + 7.7y &= 2.4; \\ (k - 1)^2 &< 4; \\ (\ln(y + 2.0x + 12.0) < (k + 5)) \text{ or } (y > k^2) &\rightarrow (x < 0.0) \text{ and } (y < 1.0); \\ (x < 0.0) &\rightarrow (k > 1), \end{aligned}$$

где k — целое, x, y — вещественные.

Решения, полученные на UniCalc:

$$\begin{aligned} k &= 2, \\ x &= -0.658479, \\ y &= 0.482722. \end{aligned}$$

$$\text{Точность} = 10^{-4}.$$

Ниже приводится несколько целочисленных проблем. Проблемы такого типа достаточно часто возникают на практике.

Пример 4

Разложение на сумму квадратов:

$$\begin{aligned} i^2 + j^2 &= 61^2; \\ i &> 1; \\ j &> i. \end{aligned}$$

Решение, полученное на UniCalc:

$$i = 11; \quad j = 60.$$

Пример 5

Разложение на целочисленные множители:

$$k * l = 1333;$$

$$k > 1;$$

$$l > k.$$

Решение, полученное на UniCalc:

$$k = 31; \quad l = 43.$$

Пример 6

Решение Диофантовых уравнений:

$$31m + 43n = 997;$$

$$m > 0;$$

$$n > 0.$$

Решение, полученное на UniCalc:

$$m = 28, \quad n = 3.$$

Пример 7

Нахождение показателя степени разложения:

$$11^n + 60^n = 61^n;$$

$$n > 0.$$

Решение, полученное на UniCalc:

$$n = 2.$$

Ниже, на примере нескольких задач, приведен сравнительный анализ результатов, полученных с помощью UniCalc с результатами, полученными аналитически или другими методами.

Пример 8

Решение системы линейных алгебраических уравнений, для которой выполнено условие диагонального преобладания:

$$1.1161x_1 + 0.1254x_2 + 0.1397x_3 + 0.1490x_4 = 1.5471;$$

$$0.1582x_1 + 1.1675x_2 + 0.1768x_3 + 0.1871x_4 = 1.6471;$$

$$0.1968x_1 + 0.2071x_2 + 1.2168x_3 + 0.2271x_4 = 1.7471;$$

$$0.2368x_1 + 0.2471x_2 + 0.2568x_3 + 1.2671x_4 = 1.8471.$$

Известные решения системы:

$$x_1 = 1.04059, \quad x_2 = 0.98697, \quad x_3 = 0.93505, \quad x_4 = 0.88130.$$

$$\text{Точность} = 10^{-4}.$$

Решения, полученные на UniCalc:

$$x_1 = 1.04058, \quad x_2 = 0.986956, \quad x_3 = 0.935053, \quad x_4 = 0.881297.$$

$$\text{Точность} = 10^{-4}.$$

Пример 9

Степенное уравнение:

$$5^t + 6^t = 7^t.$$

Известное решение:

$$t = 2.97354894.$$

$$\text{Точность} = 10^{-5}.$$

Решение, полученное на UniCalc:

$$t = [2.97354891, 2.97354892].$$

$$\text{Точность} = 10^{-7}.$$

Пример 10

Решение неравенства:

$$-3 \frac{\ln(x+1)}{\ln(0.7)} < 6.$$

Известные решения:

$$-1 < x < 1.0408163.$$

$$\text{Точность} = 10^{-6}.$$

Решения, полученные на UniCalc:

$$-1 < x < 1.0408163.$$

$$\text{Точность} = 10^{-6}.$$

Пример 11

Нахождение корней полинома:

$$f(x) := x^4 - 12x^3 + 47x^2 - 60x + a;$$

$$f(x) = 0.$$

Известные решения при некоторых значениях свободного члена a :

при $a = 0$ полином имеет четыре корня:

$$x_1 = 0, \quad x_2 = 3, \quad x_3 = 4, \quad x_4 = 5;$$

при $a = 24.0$ полином имеет два корня:

$$x_1 = 1, \quad x_2 = 0.888305779072;$$

при $a = 24.1$ корней нет.

Решения, полученные на UniCalc, совпадают с вышеуказанными.

Пример 12

Решение сильно нелинейной системы:

$$3\ln(x) + \sin^3(x) + \cos(y) = 1.0;$$

$$0.5\operatorname{tg}(e^{y-x})x^3 - \cos(x) = 4.$$

Известные решения:

$$x_1 = [1.064636, 1.06464], \quad y_1 = [1.427239, 1.427241];$$

$$x_2 = [1.399863, 1.399868], \quad y_2 = [2.88024, 2.880253].$$

Точность 10^{-7} .

Решения, полученные на UniCalc:

$$x = [0.785800579, 1.73095492],$$

$$y = [0.762434474, 3.27016143].$$

Точность 10^{-7} .

Решения, полученные с помощью команды **Поиск корней**:

$$x_1 = 1.06463755, \quad y_1 = 1.42724098;$$

$$x_2 = 1.3998655, \quad y_2 = 2.88024679.$$

Пример 13

Нахождение локальных максимумов функции:

$$f(x) := x^6 - 26x^3 + 45x^2 - 10x + 1.$$

Воспользуемся необходимыми условиями максимума функции — равенством нулю первой производной и отрицательностью второй.

Первая производная:

$$g(x) := 6x^5 - 78x^2 + 90x - 10.$$

$$g(x) = 0.$$

Вторая производная:

$$h(x) = 30x^4 - 156x + 90.$$

$$h(x) < 0.$$

Известное решение:

$$f_{\max} = [11.8627, 118640] \text{ в точке } x = [1.21767, 1.21768].$$

Точность 10^{-4} .

Решение, полученное на UniCalc:

$$f_{\max} = 11.8633479,$$

$$x = 1.21767322.$$

Точность 10^{-7} .

Пример 14

Нахождение минимума для функции Розенброка:

$$y := 100((x_2 - x_1^2)^2 + (x_3 - x_2^2)^2 + (x_4 - x_3^2)^2 + (x_5 - x_4^2)^2 + (x_6 - x_5^2)^2 + (x_7 - x_6^2)^2 + (x_8 - x_7^2)^2) + (1 - x_1)^2;$$

$$y = [0.0, 1.0].$$

Известные решения:

$$x_1 = x_2 = x_3 = x_4 = x_5 = x_6 = x_7 = x_8 = 1,$$

$$y = 0.$$

Решение, найденное с помощью **Поиска корней**, совпадает с точным.

Приложение В

Термины

Модель

Система математических выражений, описывающая какой-либо процесс.

Метод недоопределенных вычислений (МНВ)

Метод, который позволяет оперировать переменными, имеющими недоопределенные значения и получать либо точные значения, либо интервал значений, содержащий все решения, либо сообщение о несовместности системы.

Недоопределенные значения (н-значения)

Значения, обусловленные недостаточностью знаний природы изучаемого явления или полученные в результате неточного измерения.

Недоопределенные переменные (н-переменные)

Переменные, которым можно сопоставить совокупность значений.

Недоопределенные целые (н-целые)

Целые переменные, которым можно сопоставить совокупность целых значений.