

Обзорная лекция по системам компьютерной математики

Алексей Орлов

9 сентября 2020 г.

Содержание

1	Введение	1
2	Компьютерные средства для проведения математических расчетов	3
3	MATLAB и его свободные реализации	4
3.1	Octave	10
3.2	Scilab	12
3.3	Freemat	14
4	Системы компьютерной алгебры	14
5	Maxima	16
6	Wolfram Mathematica	21
7	Maple	25
8	Mathcad	31
9	Средства на основе Python	34
9.1	NumPy	34
9.2	SymPy	36
9.3	SciPy	37

1 Введение

Задача проведения автоматизированных расчётов возникает еще на заре цивилизации. Из лекции «Архитектура ЭВМ и ОС» вы уже знаете, как развивались вычислительные приборы, начиная еще с древних времен. Однако

чем все это было мотивировано и ради чего требовались все большие вычислительные мощности? Что ж, задачи, стоящие перед человеком и человечеством становились всё более сложными и изощрёнными. И если раньше торговцы могли справиться с расчётами при помощи таких приспособлений, как счётные палочки либо, позднее, финикийские глиняные фигурки, то сейчас таким участникам рынка, как финансовым аналитикам, необходимо применять сложные вычислительные алгоритмы и математические модели.

Особенно остро вопрос решения вычислительных задач встал в эпоху, когда математика начала оформляться как наука. История математики до компьютерной эры содержит много примеров трудоемких вычислений. Некоторые вычисления сводились к сложным громоздким преобразованиям формул, другие вычисления использовали небольшие формулы, но требовали выполнения операций с большим количеством цифр в числах. Уж поверьте, Иоганну Кеплеру приходилось крайне нелегко, проделывая тот титанический набор вычислений, соотнося их с имеющимися данными наблюдений. Причём всё вручную, на бумаге. Однако именно это привело его к открытию законов движения планет Солнечной системы. На это уходили долгие-долгие годы. Целые десятилетия!

Впечатляющие вычисления с карандашом и бумагой проделал французский астроном Чарльз-Евгений Делоне для вычисления орбиты Луны. Он вывел около 40000 формул. На их вывод потребовалось 10 лет и еще 10 лет ушло на проверку формул. Окончательная формула занимала 128 страниц его книги с результатами работы. Проверка его аналитических преобразований была проведена двумя американскими математиками с помощью компьютера в 70-е годы XX века. Компьютеру потребовалось двое суток работы.

Большие усилия тратили математики на определение числа π , вручную вычисляя большое количество цифр. Так, например, наилучший результат к концу XIX века был получен англичанином Вильямом Шенксом. Он потратил 15 лет для того, чтобы вычислить 707 цифр, хотя из-за ошибки только первые 527 были верными.

Итак, зачем считать на компьютере теперь понятно. Это позволяет проводить громоздкие вычисления быстро и безошибочно, моделировать сложнейшие системы совершенно различной природы. Современный мир попросту был бы невозможен без компьютерных вычислений.

Так какими же программными средствами можно воспользоваться для того, чтобы «считать» на компьютере? Здесь мы подразумеваем наукоемкие вычисления. Такие, которые нельзя провести на поставляемом вместе с операционной системой калькуляторе даже в инженерном режиме. Такие вычисления включают в себя, например, решения задач из курса высшей математики и линейной алгебры, задачи математической физики и многие другие. Вас, как студентов, скорее всего, может заинтересовать вопрос: "Где мне делать матан или решать линейную алгебру?". И мы рассмотрим программные средства, которые позволят вам сделать это. Однако все приводимые программные пакеты способны на гораздо большее и используются

учёными и инженерами в их повседневной исследовательской деятельности.

Для каждого из них мы посмотрим, как можно решить одну и ту же задачу. А именно, в каждом из пакетов мы создадим матрицу A , вектор B и будем искать решение уравнения $Ax = B$ в виде $x = A^{-1}B$, где

$$A = \begin{vmatrix} 12 & 4.2 & -0.8 \\ -4.1 & 2.2 & -0.16 \\ -1.6 & -4.3 & 8.4 \end{vmatrix} \quad \text{и} \quad B = \begin{vmatrix} -5.4 \\ 1.6 \\ 12.2 \end{vmatrix}.$$

Также мы рассмотрим графические возможности, предоставляемые непосредственно «из коробки», построив трехмерную поверхность, задаваемую функцией двух аргументов $F(x, y)$.

2 Компьютерные средства для проведения математических расчетов

На данный момент существует множество решений, которые позволят вам проводить компьютерные вычисления разной степени сложности. Это может быть как коммерчески разрабатываемое программное обеспечение, так и свободное с открытым кодом. Такие системы называются *системами компьютерной математики* (СКМ).

СКМ – это комплекс программ, который обеспечивает автоматизированную, технологически единую и замкнутую обработку задач математической направленности при задании условия на специально предусмотренном языке. Каждая система компьютерной математики имеет нюансы в своей архитектуре или структуре. Зачастую СКМ обладают следующей типовой структурой: центральное место занимает ядро системы – коды множества заранее откомпилированных функций и процедур, обеспечивающих достаточно представительный набор встроенных функций и операторов системы.

Функции и процедуры, включённые в ядро, выполняются предельно быстро. Поэтому объём ядра ограничивают, но к нему добавляют библиотеки более редких процедур и функций. Кардинальное расширение возможностей систем и их адаптация к решаемым конкретными пользователями задачам достигаются за счёт пакетов расширения систем. Эти пакеты (нередко и библиотеки) пишутся на собственном языке программирования той или иной СКМ, что делает возможным их написание обычными пользователями.

В чём же состоят особенности численного счета на компьютере? Когда мы говорим о вычислительных методах, то считаем, что все вычисления выполняются в поле вещественных или комплексных чисел. В действительности же всякая программа для компьютера имеет дело только с конечным набором рациональных чисел, поскольку только такие числа представляются в компьютере. Для записи целого числа отводится обычно 16 или 32 двоичных символа (бита), для вещественного – 32 или 64 бита. Это множество не замкнуто относительно арифметических операций, что значит, что

мы можем получить неточный результат, также это может выражаться в различных переполнениях (например, при умножении достаточно больших чисел или при делении на маленькое число). Ещё более существенной особенностью вычислительной математики является то, что арифметические операции над этими числами, выполняемые компьютером, отличаются от арифметических операций в поле рациональных чисел.

Современные системы компьютерной математики представляют собой программы с многооконным графическим интерфейсом, развитой системой помощи, что облегчает их освоение и использование. Эти системы имеют дружелюбный интерфейс, реализуют множество стандартных и специальных математических операций, снабжены мощными графическими средствами и обладают собственными языками программирования. Всё это предоставляет широкие возможности для эффективной работы специалистов разных профилей, о чём говорит активное применение математических пакетов в научных исследованиях, обучении и преподавании.

Такие проприетарные СКМ высокого уровня, как MATLAB, Mathcad, Maple и Mathematica получили широкое распространения. Они оказывают влияние друг на друга. Так, символьные математические вычисления уже давно доступны не только в Maple и Mathematica. MATLAB обзавёлся собственными средствами компьютерной алгебры, поглотив проект MuPAD, который по возможностям сравним с Maple. Но MATLAB надолго останется первым среди них по возможностям получить численные результаты, поскольку он и многочисленные приложения к нему создавались прежде всего для этих целей.

Кроме закрытого коммерческого программного обеспечения, имеются и активно развиваются бесплатные системы компьютерной математики, например, *Maxima*, *Scilab*, *Octave* и другие.

3 MATLAB и его свободные реализации

MATLAB (MATrix LABoratory) — одна из самых передовых систем компьютерной математики – высокоэффективного программного обеспечения для научных и инженерных расчетов. Выполняя математические расчеты в программной среде MATLAB, пользователь имеет крайне комфортные возможности для решения различных вычислительных задач в областях линейной алгебры, общей теории систем, теории информации и обработки сигналов, теорий автоматического и автоматизированного управления и ряда иных дисциплин.

Пакет ориентирован в первую очередь на численные расчеты. В его набор входят реализации большого количества численных схем и алгоритмов. Такие пакеты численного счёта, как MATLAB особенно ценны, ведь наблюдаемая нами реальность устроена так, что только простейшие модели позволяют получить аналитическое решение, в большинстве остальных случаев решение получается путем использования численных методов. Впрочем, в MATLAB также имеется расширение для символьных вычислений, бывшее

в прошлом самостоятельным пакетом MuPAD.

Система фактически стала мировым стандартом в области современного математического и научно-технического программного обеспечения.

Для MATLAB имеется возможность подключать специальные наборы инструментов (так называемых «тулбоксов», от англ. toolbox), расширяющие его функциональность. Наборы инструментов представляют собой собрания функций, написанных на языке MATLAB для решения крайне широкого класса задач. Огромное число расширений делает её самой большой СКМ для персональных компьютеров. Кроме базового набора инструментов, предусматривается возможность применения и следующих специализированных тулбоксов:

- Signal Processing Toolbox «обработка сигналов»
- Optimization Toolbox «оптимизация»
- Neural Network Toolbox «нейронные сети»
- Control System Toolbox «системы управления»
- Spline Toolbox «сплайны»
- SIMULINK (Dynamic System Simulation Software) «программное обеспечение моделирования динамических систем»
- Symbolic Math Toolbox (упомянутый выше MuPAD)
- и многих десятков других.

В MATLAB удачно реализованы средства работы с многомерными массивами, большими и разреженными матрицами и многими типами данных. Система прошла многолетний путь развития от узко специализированного матричного программного модуля, используемого только на больших электронно-вычислительных машинах, до универсальной интегрированной СКМ, ориентированной на массовые персональные компьютеры, а также суперкомпьютеры. MATLAB имеет мощные средства создания графического интерфейса пользователя, построения графиков и комплексной визуализации вычислений.

MATLAB – это также название высокоуровневого языка программирования, используемого пакетом. MATLAB по сравнению с традиционными языками программирования (C/C++, Python, Java) позволяет на порядок сократить время решения типовых задач и значительно упрощает разработку новых алгоритмов, включая высокоуровневые с использованием концепций объектно-ориентированного программирования.

В нём имеются все необходимые средства интегрированной среды разработки, включая отладчик и профайлер.

Типичное использование MATLAB – это:

- Математические вычисления

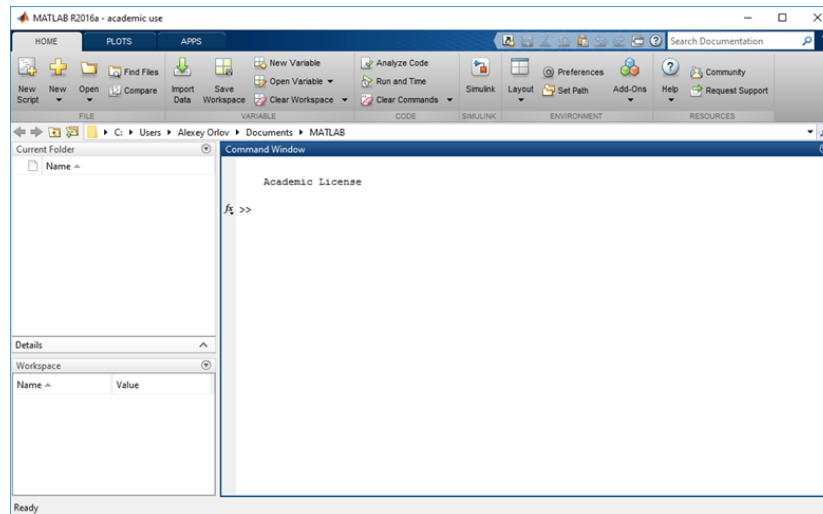


Рис. 1: Главное рабочее окно MATLAB'a непосредственно после его открытия. Видно, что используется академическая лицензия для образовательных учреждений.

- Создание алгоритмов
- Моделирование
- Анализ данных, их исследование и визуализация
- Научная и инженерная графика
- Разработка приложений, включая создание графического интерфейса

Типичное научное применение включает решение таких задач как: табулирование функций, решение нелинейных уравнений, поиск оптимальных решений, решение задач Коши, численное интегрирование и других задач, традиционно включаемых в курс численных методов.

Работа в среде MATLAB может осуществляться в двух режимах:

- в интерактивном режиме, когда вычисления осуществляются сразу после набора очередного оператора или команды MATLAB; при этом значение результатов вычисления могут присваиваться некоторым переменным, или результаты выводятся непосредственно в командное окно, без присваивания;
- путем вызова имени программы, написанной на языке MATLAB, предварительно составленной и сохранённой на диске, которая содержит все необходимые команды, обеспечивающие ввод данных, организацию вычислений и вывод результатов на экран).

В обоих режимах пользователю доступны практически все вычислительные возможности системы, в том числе по выводу информации в графической форме. Работа с написанными скриптами позволяет сохранять разработанные вычислительные алгоритмы и, таким образом, повторять вычисления при других входных данных.

Интерпретатор MATLAB позволяет в одном и том же сеансе работы выполнять несколько самостоятельных программ, причем все переменные, используемые в этих программах, являются общими для этих программ и образуют общее рабочее пространство (Work Space). Это позволяет более рационально организовывать сложные громоздкие вычисления.

MATLAB, в отличие от многих языков программирования высокого уровня, содержит в себе очень простую в пользовании встроенную арифметику комплексных чисел. Большинство элементарных математических функций допускают в качестве аргументов комплексные числа, а результаты формируются как комплексные числа. Эта особенность языка делает его очень удобным и полезным для научных работников и инженеров.

Ввод векторов и матриц можно осуществлять путем поэлементного ввода. Для этого следует сначала указать имя вектора, потом поставить знак присваивания '=', затем - открывающую квадратную скобку, а за ней ввести заданные значения элементов вектора или строки матрицы, отделяя их пробелами или запятыми. Строки матрицы при этом отделяются друг от друга точкой с запятой. Заканчивается запись закрывающей квадратной скобкой.

Помимо памяти, необходимой для хранения числовых элементов, MATLAB автоматически при создании массивов выделяет еще и память для управляющей информации. В этой области памяти хранится размерность массива, количество элементов по каждой размерности, тип элементов (вещественные или комплексные) и т. д. MATLAB может даже динамически перестраивать структуру массива.

Например, код

```
B = [-5.4 1.6 12.2]
```

создаёт вектор-строку B , содержащий три элемента, а строка

```
B = [-5.4; 1.6; 12.2]
```

создаст вектор-столбец B из этих же значений.

Наша матрица A будет вводиться следующим образом:

```
A = [12 4.2 -0.8; -4.1 2.2 -0.16; -1.6 -4.3 8.4]
```

Функция обращения матрицы `inv(M)` вычисляет матрицу, обратную заданной матрице M . Исходная матрица M должна быть квадратной, а ее определитель не должен равняться нулю.

Оригинальными в языке MATLAB являются две новые, неопределяемые в математике функции деления матриц. При этом вводятся понятия деления матриц слева направо и деления матриц справа налево. Первая операция записывается с помощью знака «/», а вторая - «\».

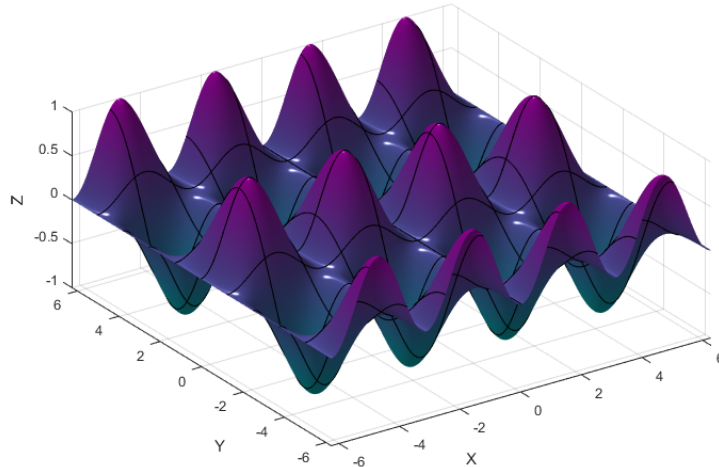


Рис. 2: Поверхность построенной нами двумерной функции $\sin(x)^2 \cos(y)$. MATLAB позволяет установить освещение и выбрать, как материал поверхности будет отражать свет. В данном примере для поверхности использован тип материала «металл».

Операция B / A эквивалентна последовательности действий $B * \text{inv}(A)$, где функция $\text{inv}()$ осуществляет обращение матрицы. Ее удобно использовать для решения матричного уравнения: $X * A = B$.

Аналогично, операция $A \setminus B$ равносильна совокупности операций $\text{inv}(A) * B$, которая представляет собой решение матричного уравнения $A * X = B$.

Итого, наша СЛАУ в MATLAB'е будет решаться в командном окне следующим образом:

```
>>> A = [12 4.2 -0.8; -4.1 2.2 -0.16; -1.6 -4.3 8.4];
>>> B = [-5.4; 1.6; 12.2];
>>> x = A \ B
x =
```

```
-0.3907
0.1032
1.4308
```

В результате вычислений в системе MATLAB обычно получается большой массив данных, который трудно анализировать без наглядной визуализации. Поэтому система визуализации, встроенная в MATLAB, придает этому пакету особую практическую ценность. Графические возможности системы MATLAB являются мощными и разнообразными. MATLAB пока-

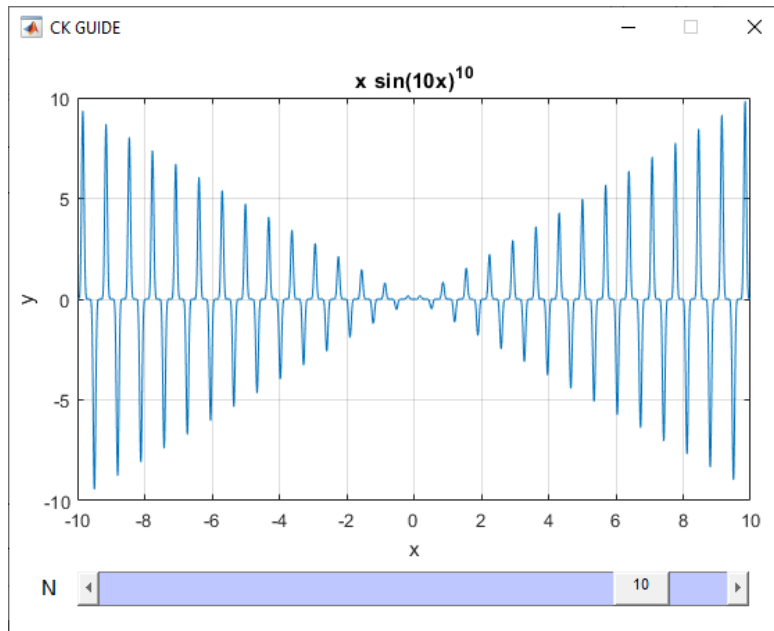


Рис. 3: Приложение с графическим интерфейсом, разработанное в пакете MATLAB. За изменение параметра N отвечает слайдер внизу окна. Индикация его значения не является стандартной возможностью GUIDE, а достигается при помощи написанного нами кода.

зывает графические объекты в специальных графических окнах, имеющих в заголовке слово *Figure* (фигура, изображение). Графический интерфейс редактирования графики позволяет тонко настроить все его свойства. Они также могут быть прописаны в коде скрипта, в итоге на выходе мы получаем полностью оформленный готовый к публикации график, такой как изображенный на рис. 2.

В MATLAB также можно создавать полноценные приложения с графическим интерфейсом. Для этого в состав пакета входит специализированная среда GUIDE. Работа в этой среде достаточно проста – элементы управления (кнопки, раскрывающиеся списки и т.д.) размещаются при помощи мыши, а затем программируются события, которые возникают при обращении пользователя к данным элементам управления.

Приложение может состоять как из одного основного окна, так и нескольких окон и осуществлять вывод графической и текстовой информации в основное окно приложения либо в отдельные окна. Ряд функций MATLAB предназначен для создания стандартных диалоговых окон открытия и сохранения файла, печати, выбора шрифта, окна для ввода данных и других, которыми можно пользоваться в собственных приложениях.

Для демонстрации создания приложений с графическим интерфейсом

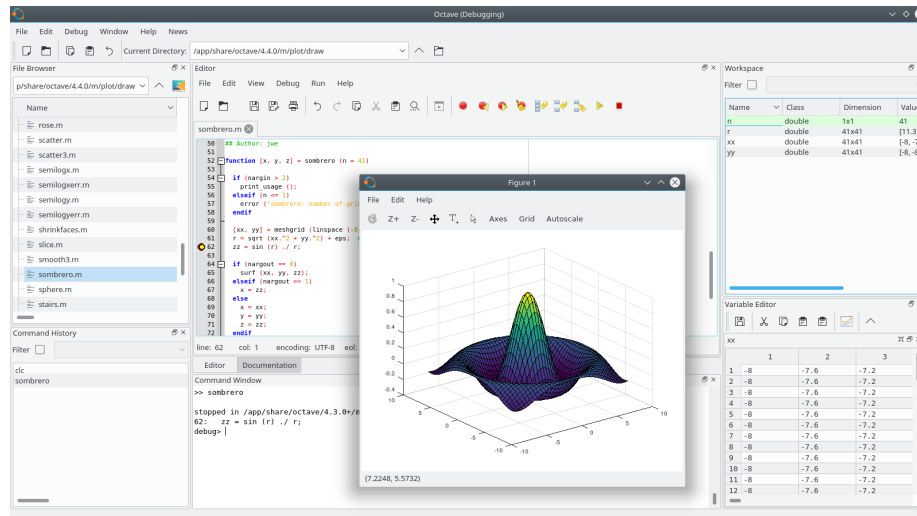


Рис. 4: Интерфейс и небольшая демонстрация возможностей пакета Octave. В целом интерфейс пакета обнаруживает крайнее сходство с интерфейсом MATLAB прошлых версий.

нами была написана программа, выводящая график функции $x \sin(Nx)^N$. Значение N при этом задаётся перемещением ползунка, расположенного внизу окна программы (см. рис. 3. Также обратим внимание на то, что значение, выставленное ползунком, отображается непосредственно на нём. Это не стандартный функционал, предоставляемый GUIDE. Значение отображается в текстовом поле, положение которого совпадает с позицией слайдера и динамически изменяется при его перемещении. Таким образом, в MATLAB можно разрабатывать по-настоящему гибкие графические приложения и интерфейсы.

3.1 Octave

Пакет MATLAB, как говорилось выше, является столпом в численных вычислениях и обработке сигналов. Однако цена, по которой он продаётся, довольно высока. С этим не могли смириться определённые группы людей, математиков и программистов, в результате чего возникли проекты по написанию свободной и бесплатной реализации MATLAB'a.

Из них крайне популярным является проект GNU Octave - свободной системы для математических вычислений, использующей совместимый с MATLAB язык высокого уровня. Из всех доступных свободно распространяемых альтернатив Octave предоставляет наибольшую совместимость кода с языком MATLAB.

Octave — интерпретирующий язык программирования, похож на C и поддерживает большинство основных функций стандартной библиотеки C.

Синтаксис языка очень похож на MATLAB, и грамотно написанные скрипты будут запускаться как в Octave, так и в MATLAB.

В конце 1990-х гг. (2-я серия выпусков) отмечалось, что производительность Octave существенно уступает производительности MATLAB. К концу 2000-х гг. (3-я серия выпусков) производительности этих пакетов «почти сравнялись». Остающуюся разницу приписывают наличию в составе MATLAB интерпретатора-компилятора (Just in Time — JIT), которого нет в Octave. На настоящий момент производительность MATLAB все же выше, чем у Octave.

Следует учесть, что Octave хоть и написан на C++, при установке он будет искать, не присутствует ли в системе Java Runtime Environment (JRE). Связано это с тем, что пакет оснащен интерфейсом, который позволяет использовать классы Java прямо в Octave. Впрочем, наличие его совершенно необязательно. В этом случае использовании Java в Octave будет недоступно.

Octave был написан с учётом совместимости с MATLAB и реализует многие его возможности:

- матрицы в качестве основных типов данных
- встроенная поддержка комплексных чисел
- мощные встроенные математические функции и большие библиотеки функций
- расширяемость, благодаря возможности создания пользовательских функций

Но есть и отличия:

- комментарии могут начинаться как с символа `#`, так и с символа `%`
- поддерживаются C-подобные операторы `++`, `-`, `+=`, `*=`, `/=`
- элементы могут быть адресованы без создания новой переменной, например `[1:10](3)`
- строки могут быть заданы как символом `«"»`, так и символом `«'»`.

Octave по-прежнему сохраняет свою приверженность командной строке, отчего не имеет полноценных инструментов разработки графического интерфейса пользователя. Сама система всё таки обзавелась стабильным графическим интерфейсом, написанным с использованием библиотеки Qt. А вот средств для разработки программ с пользовательским графическим интерфейсом практически не имеется. В наличии, например, есть прогресс-бар и диалоги открытия/сохранения файлов, а также некоторые элементы пользовательского интерфейса вроде меню, кнопок, слайдеров, выпадающих списков и т. д. Но все эти элементы придется прописывать вручную в коде, если вы вдруг соберётесь их использовать.

Интересным фактом является то, что система находит применение не только на домашних пользовательских машинах. Octave используется в науке и промышленности. Так, Octave был развернут на массивном высокопроизводительном кластере в Питтсбургском суперкомпьютерном центре, чтобы найти уязвимости, связанные с угадыванием номера социальной страховки.

Octave доступен для работы онлайн по следующей ссылке: <https://octave-online.net/>. После несложной регистрации там позволяют даже запускать скрипты.

3.2 Scilab

Пакет прикладных математических программ, предоставляющий открытое окружение для инженерных (технических) и научных расчетов. Из всех проектов, создающих бесплатную замену MATLAB'у, считается наиболее полной общедоступной его альтернативой. В том числе потому, что содержит пакет моделирования объединенных объектов Scicos (Scilab Connected Object Simulator), который служит для имитационного моделирования систем, состоящих из блоков с заданными свойствами (параметрами). Это функциональный аналог пакета Simulink из MATLAB.

Содержит сотни математических функций, и есть возможность добавления новых. Полной совместимости с MATLAB'ом Scilab, впрочем, не имеет. Это можно видеть уже хотя бы из того, что в Scilab отсутствует функция `plot3`, которая является трехмерным аналогом функции `plot` для построения двумерных графиков. Впрочем, для построения трехмерных поверхностей возможности Scilab достаточно богаты.

Имеются разнообразные структуры данных (списки, полиномы, рациональные функции, линейные системы), интерпретатор и язык высокого уровня.

При установке пакета, в отличие от Octave, установка окружения Java является обязательной. Хотя у вас и получится снять галочку с "Java Environment" при выборе компонентов для установки, Scilab вам немедленно заявит, что без Java он устанавливаться не будет.

Для Scilab доступен модуль символьных вычислений который, однако, работает на базе пакета Maxima. Последний будет более подробно рассмотрен в главе 5.

В Scilab присутствуют средства разработки графического пользовательского интерфейса. Причем имеются целых две опции:

- Интерфейс TCL/Тк. Эти средства предназначены для конструирования графического окна с использованием набора управляющих компонентов
- Графический пользовательский интерфейс и диалоговые окна. Эти средства предназначены для работы с графическим редактором Scilab

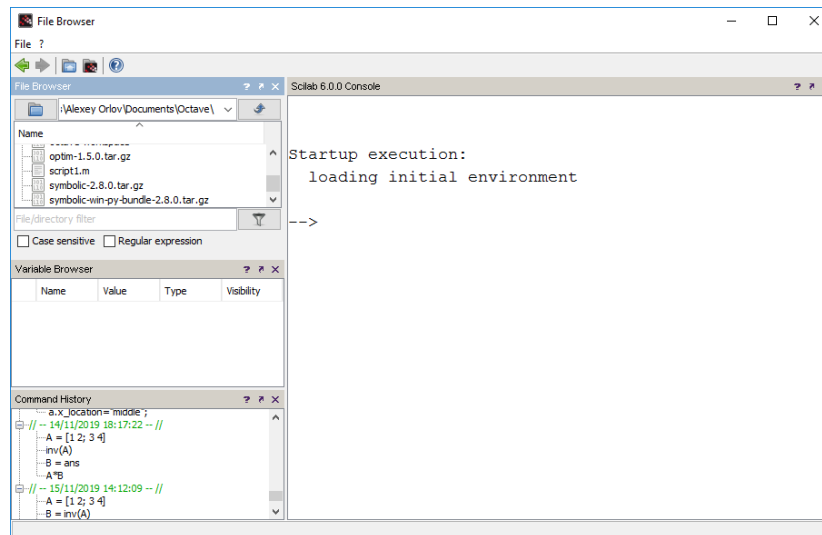


Рис. 5: Главное рабочее окно пакета Scilab. Видно командное окно, окно истории команд, а также окно с рабочей директорией и окно переменных. Ровно так выглядел MATLAB версии 6.0.

```
--> A = [12 4.2 -0.8; -4.1 2.2 -0.16; -1.6 -4.3 8.4]
A =

    12.    4.2   -0.8
   -4.1    2.2  -0.16
   -1.6   -4.3    8.4

--> B = [-5.4; 1.6; 12.2]
B =

   -5.4
    1.6
   12.2

--> x = A^-1*B
x =

   -0.3907227
    0.1031635
    1.4307675
```

Рис. 6: Выполненное нами задание по решению СЛАУ методом обратной матрицы в командной консоли пакета Scilab .

TCL (от англ. Tool Command Language — «Командный язык инструментов», читается «тикль») — скриптовый язык высокого уровня. TCL часто применяется совместно с графической библиотекой Tk (Tool Kit). Связку TCL/Tk по-русски иногда называют «Так-тикль».

Scilab позволяет работать с элементарными и большим числом специальных функций (Бесселя, Неймана, интегральные функции), имеет мощные средства работы с матрицами, полиномами (в том числе и символично), производить численные вычисления (например, численное интегрирование) и решение задач линейной алгебры, оптимизации и симуляции, мощные статистические функции, а также средство для построения и работы с графиками.

3.3 Freemat

Вы можете также обнаружить, что существует и третий бесплатный аналог MATLAB'a — пакет Freemat. Распространяется он под свободной лицензией GPL и бесплатно предоставляет часть функциональности MATLAB. Хотя по совместимости с ним серьёзно уступает даже Scilab'у: пакет не поддерживает некоторые функции, в том числе крайне важные, такие как `mesh`, что ограничивает его возможности по работе с 3D графиками. Также в нём отсутствует функционал для решения дифференциальных уравнений. Последняя версия на момент написания настоящего текста датируется 2013 годом. Неизвестно, будет ли его разработка поддерживаться дальше.

4 Системы компьютерной алгебры

В 60-х годах XX века стали создаваться первые системы компьютерной алгебры (СКА). Система компьютерной алгебры (computer algebra system) — программа для выполнения символьных (математических) вычислений. Основная определяющая функциональность таких систем — это операции с выражениями в символьной форме. Хотя компьютеры начали применяться для решения таких задач символьных преобразований, как, например, символьное дифференцирование, ещё в 50-х годах прошлого века, активная разработка систем компьютерной алгебры началась именно в конце 60-х годов.

В чём основные отличия символьных вычислений от численных и почему возник термин «компьютерная алгебра»?

Символьные операции обычно включают в себя: вычисление символьных либо числовых значений для выражений, преобразование, изменение формы выражений, нахождение производной одной или нескольких переменных, решение линейных и нелинейных уравнений, решение дифференциальных уравнений, вычисление пределов, вычисление определённых и неопределённых интегралов, работа с множествами, вычисления и работа с матрицами.

В компьютерной алгебре вещественные и комплексные числа практически не применяются, зато широко используются алгебраические числа. Алгебраическое число задаётся своим минимальным многочленом, а иногда для его задания требуется указать интервал на прямой или область в комплексной плоскости, где содержится единственный корень данного многочлена. Многочлены играют в символьных вычислениях исключительно важную роль. На использовании полиномиальной арифметики основаны теоретические методы аналитической механики, они применяются во многих областях математики, физики и других наук. Кроме того, в компьютерной алгебре рассматриваются такие объекты, как дифференциальные поля (функциональные поля), допускающие показательные, логарифмические, тригонометрические функции, матричные кольца (элементы матрицы принадлежат кольцам достаточно общего вида) и другие. Даже при арифметических операциях над такими объектами происходит разбухание информации, и для записи промежуточных результатов вычислений требуется значительный объём памяти компьютера.

Основной класс выражений, используемых в символьных вычислениях – это многочлены, понимаемые в обобщенном смысле. Для того чтобы прояснить ситуацию, приведем пример. Преобразование

$$(x - y)(x + y) \rightarrow x^2 - y^2$$

является многочленным (полиномиальным) вычислением, но таким же является и преобразование

$$(\cos a - \sin b)(\cos a + \sin b) \rightarrow \cos 2a - \sin 2b,$$

в котором фактически произведено то же вычисление, что и в предыдущем преобразовании с заменой x на $\cos a$ и y на $\sin b$.

В большинстве систем компьютерной алгебры все выражения в диалоге с пользователем и при символьных вычислениях преобразуются в контексте автоматического упрощения. Это означает, что при вводе все операнды математических операторов автоматически упрощаются перед тем как операторы применяются, и то, что результат вычисления выражения представлен в автоматически упрощенном виде.

Автоматическое упрощение описывается множеством упрощающихся правил преобразования, применяемых без участия пользователя. Существует множество упрощающих правил. Но только часть из них следует применять во всех ситуациях. Применять или нет других правила – решает пользователь. Во-первых, во многих случаях «упрощенный вид выражения» зависит от контекста вычисления; во-вторых, применение некоторых упрощающих правил математически допустимо только при определенных условиях. Таким образом, кроме автоматического упрощения, должны существовать явно используемые правила упрощения.

Возрастающий интерес к алгебраическим алгоритмам возник в результате осознания центральной роли алгоритмов в информатике. Их легко описать на формальном и строгом языке и с их помощью обеспечить решение

задач, давно известных и изучавшихся на протяжении веков. В то время как традиционная алгебра имеет дело с конструктивными методами, компьютерная алгебра интересуется ещё и эффективностью, реализацией, а также аппаратными и программными аспектами таких алгоритмов. Оказалось, что при принятии решения об эффективности и определении производительности алгебраических методов требуются многие другие средства, например, теория рекурсивных функций, математическая логика, анализ и комбинаторика.

В начальный период применения вычислительных машин в символьной алгебре быстро стало очевидным, что непосредственные методы из учебников часто оказывались весьма неэффективными. Вместо обращения к методам численной аппроксимации компьютерная алгебра систематически изучает источники неэффективности и ведёт поиск иных алгебраических методов для улучшения или даже замены таких алгоритмов.

5 Maxima

Maxima является потомком системы DOE-Macsyma, которая начала своё существование в конце 1960 года в MIT. Macsyma первая создала систему компьютерной алгебры, произведя в свое время переворот в компьютерной алгебре и оказала влияние на многие другие системы, в числе которых Maple и Mathematica. По сути, Macsyma проложила для них путь.

Maxima позволяет решать достаточно широкий круг задач, относящихся к различным разделам математики. Возможности системы включают действия по преобразованию выражений, работу с частями выражений, дифференцирование, интегрирование, решение задач линейной алгебры, математического анализа, комбинаторики, разложение в ряд, преобразование Лапласа, обыкновенные дифференциальные уравнения, системы линейных уравнений, многочлены, множества, списки, векторы, матрицы и тензоры, задачи теории чисел, тензорного анализа, статистики. Maxima производит численные расчеты высокой точности, используя точные дроби, целые числа и числа с плавающей точкой произвольной точности. Система позволяет строить графики функций и статистических данных в двух и трех измерениях.

Главный вариант Maxima разрабатывался Вильямом Шелтером с 1982 по 2001 год, вплоть до того, как его с нами не стало. В 1998 году он получил разрешение на открытие исходного кода системы под лицензией GPL. Это было важным шагом, так как Maxima являлась ответвлением от DOE-Macsyma и содержала её код, который был закрытым. Благодаря его стараниям Maxima сумела выжить и сохранить свой оригинальный код. Вскоре Вильям передал Maxima группе пользователей и разработчиков, которые донесли её до наших дней и поддерживают в рабочем состоянии. На настоящий момент пакет достаточно активно развивается и во многих отношениях не уступает продвинутым системам компьютерной математики. Система обновляется с завидной регулярностью, становясь только лучше, совершен-

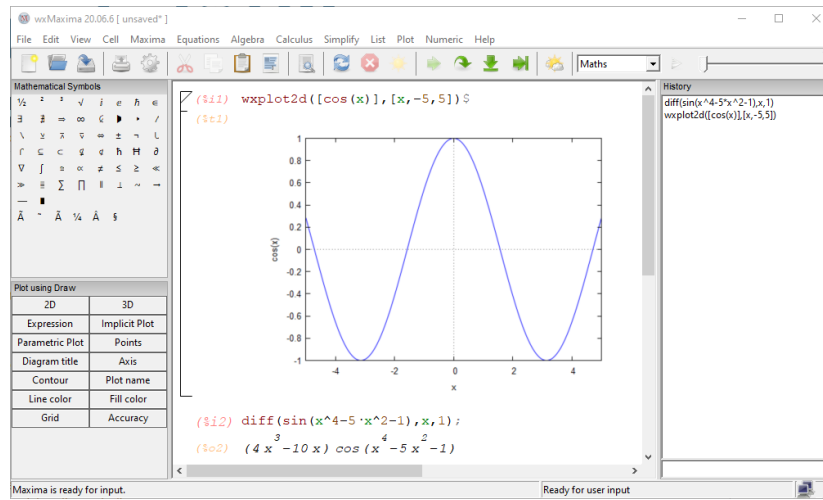


Рис. 7: wxMaxima - графическая оболочка для пакета Maxima.

ствуя код и документацию.

Пакет Maxima состоит из интерпретатора макроязыка, написанного на Lisp, и нескольких поколений пакетов расширений, написанных на макроязыке пакета или непосредственно на Lisp. Исходный код Maxima может компилироваться на многих системах, включая Windows, Linux и MacOS X. На SourceForge доступны исходные коды и исполняемые файлы для Windows и Linux.

Области математики, поддерживаемые в Maxima:

- Операции с полиномами (манипуляция рациональными и степенными выражениями, вычисление корней и т.п.)
- Вычисления с элементарными функциями, в том числе с логарифмами, экспоненциальными функциями, тригонометрическими функциями
- Вычисления со специальными функциями, в т.ч. эллиптическими функциями и интегралами
- Вычисление пределов и производных
- Аналитическое вычисление определённых и неопределённых интегралов
- Решение интегральных уравнений
- Решение алгебраических уравнений и их систем
- Операции со степенными рядами и рядами Фурье

- Операции с матрицами и списками, большая библиотека функций для решения задач линейной алгебры
- Операции с тензорами
- Теория чисел, теория групп, абстрактная алгебра

Опробовать Maxima можно без установки к себе на компьютер, протестировав систему онлайн по адресу <http://maxima.cesga.es>.

В настоящее время у системы Maxima есть мощный, эффективный и довольно дружелюбный кроссплатформенный графический интерфейс, который называется wxMaxima. Достоинствами wxMaxima являются: возможность графического вывода формул, упрощенный ввод наиболее часто используемых функций (через диалоговые окна), а не набор команд, как в классической Maxima. В текстовом меню wxMaxima находятся функции для решения большого количества типовых математических задач, разделённые по группам: уравнения, алгебра, анализ, упростить, графики, численные вычисления.

Создать матрицу в системе можно как воспользовавшись графическим интерфейсом, так и при помощи команды `matrix()`. Внутри последней матрица помещается следующим образом: элементы строки матрицы группируются при помощи квадратных скобок, перечисление элементов и сгруппированных строк идёт через запятую. Далее необходимо назначить матрице переменную, воспользовавшись оператором присвоения «:». При этом ещё одной копии матрицы не возникнет, вместо этого появится указатель на созданную матрицу.

Суммируя вышесказанное, получим следующие команды для ввода матриц A и B :

```
A:matrix([12, 4.2, -0.8], [-4.1, 2.2, -0.16],
          [-1.6 -4.3 8.4]);
B:matrix([-5.4], [1.6], [12.2]);
```

Чтобы посчитать обратную матрицу, необходимо воспользоваться командой `invert()`. Матричное умножение выполняется оператором точки. Теперь у нас есть всё для получения решения нашей СЛАУ. Результатом выполнения

```
(% i1) A:matrix([12,4.2,-0.8],[-4.1,2.2,-0.16],[-1.6,-4.3,8.4]);
```

$$\begin{pmatrix} 12 & 4.2 & -0.8 \\ -4.1 & 2.2 & -0.16 \\ -1.6 & -4.3 & 8.4 \end{pmatrix} \quad (\% \text{ o1})$$

```
→ B:matrix([-5.4],[1.6],[12.2]);
```

$$\begin{pmatrix} -5.4 \\ 1.6 \\ 12.2 \end{pmatrix} \quad (\% \text{ o2})$$

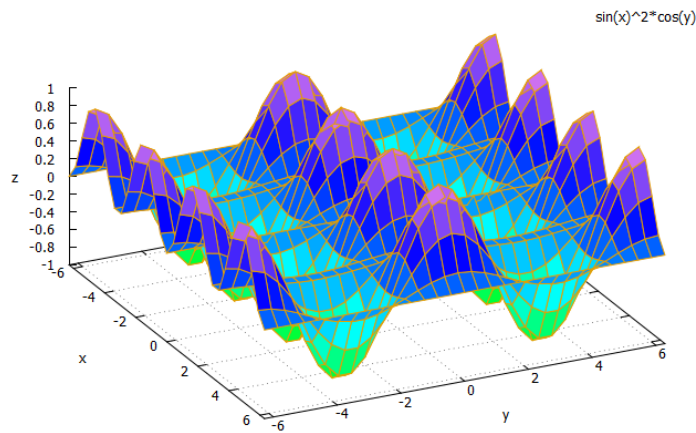


Рис. 8: График функции $\sin^2(x) \cos(y)$, построенный в пакете Maxima одной командой.

(% i7) invert(A);

$$\begin{pmatrix} 0.0519767039664956 & -0.09301586411270343 & 0.003178431537519515 \\ 0.1013592468986921 & 0.2907330024025203 & 0.01519103308373297 \\ 0.06178660571556776 & 0.1311103009226799 & 0.1274293967523908 \end{pmatrix} \quad (\% \text{ o7})$$

(% i8) x:%B;

$$\begin{pmatrix} -0.3907227192416637 \\ 0.1031634742126372 \\ 1.43076745099139 \end{pmatrix} \quad (\% \text{ o8})$$

(% i11) A.x-B;

$$\begin{pmatrix} -8.88178419700125210^{-16} \\ 4.44089209850062610^{-16} \\ 0.0 \end{pmatrix} \quad (\% \text{ o11})$$

Графические возможности системы богаты, хотя и могут выглядеть несколько архаично без дополнительной настройки. Построим график функции $F(x, y) = \sin^2(x) \cos(y)$, выполнив следующую команду:

(% i1) plot3d(sin(x)^2*cos(y),[x,-2*%pi,2*%pi],[y,-2*%pi,2*%pi]);

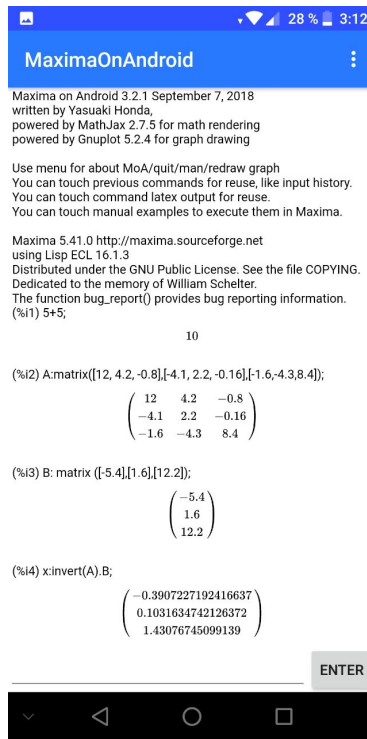


Рис. 9: Maxima в версии для Android. Показано, как ввод системы может служить в качестве простого калькулятора. Далее представлено решение нашей СЛАУ. Обратите внимание на вывод программы: представлены отформатированные в \LaTeX матрицы.

Для системы Maxima необходимо задать границы интервалов для каждой из величин. При этом операции умножения должны быть обозначены явно либо через звездочку $*$, либо через пробел. Границы интервалов для обеих переменных необходимо проставить вручную. При этом стоит учесть, что такие константы, как π , записываются через предваряющий их знак процента, то есть $\%pi$. В MATLAB'e, к примеру, к этой же константе можно обратиться просто через pi , без знака процента. Полученный график изображен на рис. 8.

Для устройств на платформе Android доступна мобильная портированная версия системы, представленная на рис. 9. Порт включает в себя помимо самой системы Maxima интерпретатор языка Lisp и библиотеку MathJax, благодаря которой вывод красиво форматируется средствами \LaTeX .

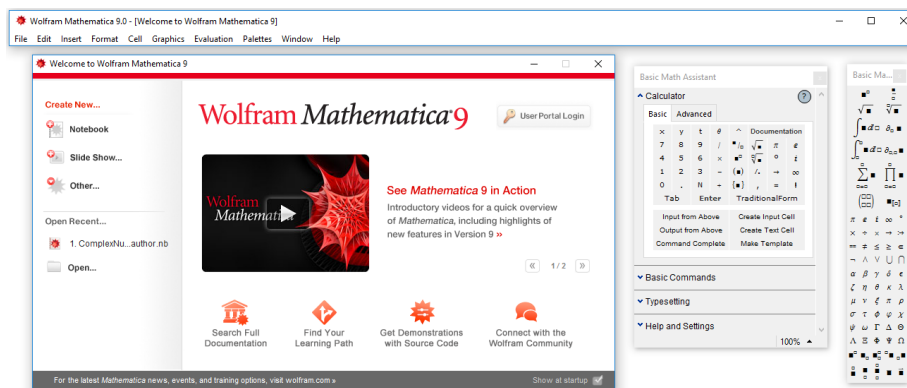


Рис. 10: Система Wolfram Mathematica непосредственно после запуска. Видно окно приветствия, предоставляющее пользователю варианты, такие как создать новый блокнот либо ознакомиться с документацией, а также другие. На рисунке также представлены открытые палитры ввода Basic Math Assistant и Basic Math.

6 Wolfram Mathematica

Mathematica — это широко используемая СКА. Изначально разработана физиком, математиком и программистом Стивеном Вольфрамом. Сейчас продаётся компанией Wolfram Research. Он начал работу над Mathematica в 1986 году, а выпустил в 1988 году. Mathematica не только СКА, но и мощный язык программирования. Этот язык включает себя несколько парадигм программирования: символьные вычисления, а также функциональное программирование. Реализован он на основе объектно-ориентированного варианта языка C, расширяемого при помощи так называемых библиотек кода. Эти библиотеки представляют собой текстовые файлы, написанные на языке Mathematica.

Архитектура Mathematica представлена ядром и пользовательским интерфейсом (фронтенд). Ядро программы отвечает за интерпретацию программ, написанных на языке Mathematica, и непосредственно занимается вычислениями. Пользовательские интерфейсы предназначены для выводов результатов в форме, понятной пользователю. По мнению компании-разработчика, большая часть пользователей Mathematica — это технические профессионалы. Также Mathematica широко используется в образовании. Сейчас несколько тысяч курсов на основе этого продукта читаются во многих учебных заведениях, начиная от средней школы и заканчивая аспирантурой.

Ядро системы является по существу идентичным на различных платформах, оболочка к ядру отличается способом, которым вы вводите и получаете информацию, и приспособлен обычно к конкретному виду компьютерной системы. Ядро и оболочка могут функционировать совершенно отдельно.

```
CK Lecture.nb

In[54]:= a := {{12, 4.2, -0.8}, {-4.1, 2.2, -0.16}, {-1.6, -4.3, 8.4}}

In[55]:= a // MatrixForm
Out[55]/MatrixForm=

$$\begin{pmatrix} 12 & 4.2 & -0.8 \\ -4.1 & 2.2 & -0.16 \\ -1.6 & -4.3 & 8.4 \end{pmatrix}$$


In[56]:= b =  $\begin{pmatrix} -5.4 \\ 1.6 \\ 12.2 \end{pmatrix}$ 
Out[56]= {{-5.4}, {1.6}, {12.2}}

In[62]:= x = Inverse[a].b
Out[62]= {{-0.390723}, {0.103163}, {1.43077}}

In[65]:= a.x // MatrixForm
Out[65]/MatrixForm=

$$\begin{pmatrix} -5.4 \\ 1.6 \\ 12.2 \end{pmatrix}$$


In[68]:= a.x == b
Out[68]= True
```

Рис. 11: Блокнот Wolfram Mathematica с выполненным заданием по решению СЛАУ методом обратной матрицы. Обратите внимание на операторы матричного умножения, а также на синтаксис функции взятия обратной матрицы. В конце выполняется проверка.

Они не обязательно должны выполняться на одном и том же компьютере, а могут работать на отдельных компьютерах, связанных коммуникационным стандартом высокого уровня, называемым MathLink.

Несмотря на 25-летнюю историю системы Mathematica, её язык программирования не имел общепринятого названия. Часто его называли языком Mathematica. Но в настоящее время этот язык стал использоваться и в других программных продуктах компании Wolfram Research, и поэтому с июня 2013 года он получил официальное название язык Wolfram. Этот мультипарадигмальный язык программирования спроектирован как максимально универсальный язык, с акцентом на символьные вычисления, функциональное программирование и программирование, основанное на правилах.

Аналитические возможности системы крайне обширны. Mathematica позволяет производить манипулирование алгебраическими формулами, то есть разлагать на множители, раскрывать скобки и производить упрощение полиномов и рациональных выражений. Она также позволяет находить алгебраические решения полиномиальных уравнений и систем уравнений, вычислять интегралы и производные, решать дифференциальные уравнения, представлять функции в виде разложения в ряд, а также вычислять пределы в символьной форме.

Mathematica также может производить расчеты с использованием большого числа специальных функций. Кроме того, Mathematica может производить численные вычисления с любой точностью. С помощью пакета Mathematica можно численно вычислять интегралы, решать алгебраические и дифференциальные уравнения и системы уравнений, обрабатывать численные данные, производя их статистический анализ, производить Фурье-анализ, интерполяцию и аппроксимацию данных с помощью метода наименьших квадратов. Mathematica может работать не только с числами, но и с матрицами, обеспечивая выполнение всех операций линейной алгебры.

Язык Wolfram является интерпретирующим языком – это означает, что команды выполняются в реальном времени без необходимости компилировать их в программы. Вы печатаете выражение, нажимаете комбинацию Shift+Enter, Mathematica отметит ввод выражения надписью `In[n] :=` и после этого ядро вычисляет значение выражения. Соответствующий вывод результата располагается под вводом и отмечается надписью `Out[n] =`.

Наряду со стандартной текстовой формой ввода, Mathematica поддерживает общую, нетекстовую форму ввода, такую как графика и управление интерактивными моделями, произвольно смешиваемую с вводом текста.

Документы в Mathematica называются блокнотами. Блокноты могут содержать обычный текст, графику, а также ввод и вывод системы. Блокнот состоит из ячеек (cells). Характеристика ячейки зависит от её функции и от вида информации, которую она содержит. В окне блокнота ячейки указаны скобками различных форм (полученных из квадратных скобок). Скобки с правой стороны окна блокнота показывают размер каждой ячейки. Вид

скобки соответствует атрибутам ячейки. Так, например, различаются скобки ячеек ввода и вывода. Ячейки могут быть объединёнными в группы; группам соответствуют групповые скобки, объемлющие другие скобки.

В интерфейсе блокнота ввод (передача информации из оболочки ядру) производится нажатием клавиш Enter с зажатой клавишей Shift. Это общий способ для платформ Macintosh и Microsoft Windows.

Mathematica использует символы верхнего и нижнего регистра и различает заглавные и строчные буквы. Все встроенные имена в системе Mathematica таких объектов, как функции, переменные, опции и т. д., начинаются с заглавных букв. Рекомендуется, чтобы вы всегда использовали символы нижнего регистра для имен переменных, которые вы определяете сами.

Также в системе различают немедленное и отложенное присваивание. Если выражение использует знак « \equiv » или функцию `Set[]`, Mathematica немедленно заменяет переменную на установленное значение. Отложенное присваивание « $:=$ » или `SetDelayed` возвращает значение, когда функция вызывается.

Это в корне отличается от того, что мы могли видеть до этого. В рассмотренных нами до этого системах имелся лишь оператор немедленного присваивания.

В пакете Mathematica оператор точки перегружен, это значит, что в зависимости от контекста, может выполнять разные операции: матричное умножение или скалярное умножение двух векторов. Это необходимо учитывать при выполнении нашего задания поиска решения СЛАУ методом обратной матрицы. Блокнот с командами, задающими и решающими наше матричное уравнение представлен на рис. 11.

В системе Mathematica реализовано огромное количество графических функций. Для построения графика поверхности, задаваемой используемой нами функцией $\sin(x)^2 \cos(y)$, применим команду `Plot3D`:

```
In[1]:= Plot3D[Sin[x]^2*Cos[y],{x,-2Pi,2Pi},{y,-2Pi,2Pi}]
```

Первый аргумент – выражение $F(x, y)$, график которого должен быть построен. Второй и третий аргументы определяют границы изменения переменных x и y , задаваемые в виде списков. Mathematica размещает график в коробку, ориентация которого определяется положением точки, из которой он рассматривается. С помощью опций функции `Plot3D` ориентацию можно изменить. Также для этого можно использовать графические средства редактирования, которые отобразятся прямо в блокноте. Всплывающие помогают гибко настроить отображения непосредственно самой поверхности, сетки и осей.

Области применения системы стремительно расширяются и включают в себя сейчас научные вычисления в бесконечном множестве разнообразных наук, вплоть до молекулярной биологии и секвенирования человеческого

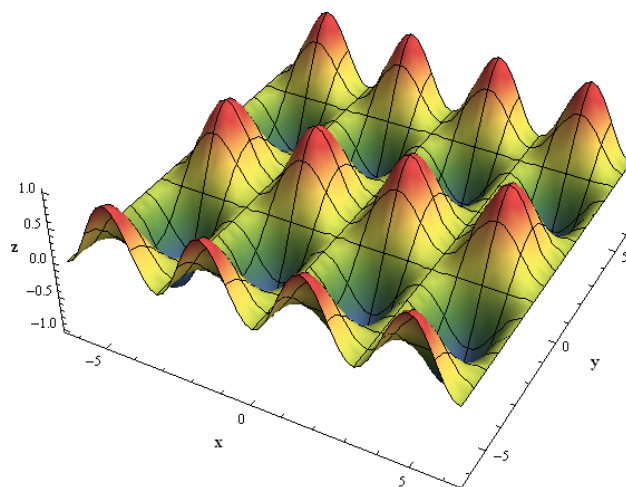


Рис. 12: График нашей двумерной функции, построенный в Wolfram Mathematica.

ДНК. Так, функция `GenomeLookup["seq"]` позволит установить точное положение ДНК-последовательности “seq” в референсном человеческом геноме.

Однако нельзя сказать, что подобная экспансия в новые предметные области благоприятно сказывается на развитии приложения. На Wolfram Research поступают жалобы на то, что они плохо воспринимают критику от сообщества пользователей, и не удосуживаются исправлять ошибки, которые обнаруживаются юзерами в самой основе системы, связанной именно с компьютерной алгеброй.

В этом плане у Maple, например, все гораздо оптимистичнее, и найденные баги в конце концов исправляются, а производитель больше внимания уделяет совершенствованию системы, нежели частоте новых релизов и агрессивному маркетингу. Хотя финансовый модуль, надо заметить, появился и там. Ничего не поделаешь: финансисты и экономисты запрашивают сейчас в своей деятельности самые мощные средства для аналитических исследований, и производители СКА им дружно отвечают.

7 Maple

Система Maple, созданная в 80-х годах прошлого века в Канаде (о чём недвусмысленно говорит само её название и логотип), с самого начала была задумана как система для персональных компьютеров, учитывающая их особенности. Её алгебраические средства существенно расширяют диапазон проблем, которые могут быть решены на качественном уровне. Она

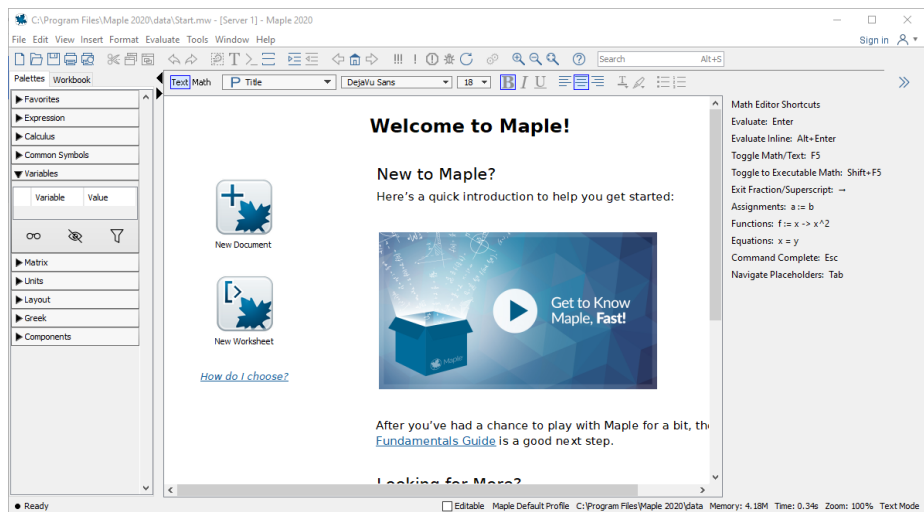


Рис. 13: Окно системы компьютерной алгебры Maple 2020 непосредственно после первого запуска. После загрузки предлагается создать либо новую интерактивную среду, либо новый рабочий лист. Слева можно видеть панели инструментов для ввода математических символов. Справа содержится контекстная панель, которая позволит вам быстро обращаться к необходимым вам функциям и выражениям.

развивается «вширь и вглубь», даже её ядро переписывалось с одного алгоритмического языка на другой. В настоящее время Maple широко применяется во многих странах в учебном процессе, а также в различных областях научных и технических исследований.

Предназначена система главным образом для выполнения аналитических (символьных) вычислений и имеет для этого один из самых мощных в своем классе арсенал специализированных процедур и функций (более 3000). Такая комплектация для большинства пользователей, которые сталкиваются с необходимостью выполнения математических расчетов среднего уровня сложности, является избыточной. Возможности Maple ориентированы в том числе и на пользователей — профессиональных математиков; решения задач в среде Maple требуют не только умения оперировать какой-либо функцией, но и знания методов решения, в неё заложенных: во многих встроенных функциях Maple фигурирует аргумент, задающий метод решения.

Maple является системой аналитических вычислений, предназначенной для облегчения решения не только математических задач, но также инженерных, финансовых, экономических и технических, требующих привлечения разнообразного математического аппарата. Пользователем данной системы может быть любой человек, владеющий математикой в пределах необходимых ему знаний для решения задач проблемной области. Это может быть

инженер, занимающийся расчетом конструкций, финансист, следящий за потоками денежных масс на предприятии, математик, использующий Maple для облегчения своей работы, и т.д.

Основной компонент Maple - рабочий лист, в который пользователь заносит не только команды Maple, но и комментарии в виде форматированного текста. Развитая система графики позволяет встраивать в рабочий лист двумерные и трехмерные графические отображения, поясняющие и помогающие решению конкретной поставленной задачи.

Также имеется режим работы в Maple в виде интерактивного сеанса: пользователь вводит команды в рабочем поле интерфейса и передает их на выполнение исполняющей системе (ядру) Maple. Все вводимые команды и отображаемые результаты вычислений представляют собой рабочий лист - основной документ, который создает Maple, и с которым он работает. При завершении сеанса работы его можно сохранить на диске в файлах разных форматов, а при очередном сеансе открыть и снова выполнить все команды, содержащиеся в нем, или произвести его корректировку.

Команды можно отображать либо в форме операторов Maple, либо в виде привычной математической записи. Например, для вычисления интеграла от функции $f(x)$ можно в поле ввода отобразить команду `int(f(x),x)`, либо установить режим, при котором эта команда отобразится в привычной математической записи $\int f(x)dx$.

Будучи системой символьных вычислений, Maple должен иметь средства для преобразования алгебраических выражений: приведения подобных членов (это делается по умолчанию), раскрытия скобок (`expand`), группировки (`collect`), разложения на множители (`factor` для многочленов и `ifactor` для целых чисел) упрощения выражений (`simplify`) и т.д. Действие перечисленных выше функций станет понятно из следующих примеров:

```
> collect(x^2+2*x*y-5*y-y^2+x*y^2+3,y);
(-1 + x) y^2 + (-5 + 2 x) y + x^2 + 3
```

```
> expand((x^2+x-1)*(x^2+3));
x^4 + 2 x^2 + x^3 + 3 x - 3
```

```
> expand(cos(3*x));
4 cos(x)^3 - 3 cos(x)
```

```
> factor(a^3-b^3);
(a - b) (a^2 + a b + b^2)
```

```
> ifactor(22!+1);
(23) (521) (93799610095769647)
```

```
> a:=sin(u)^2+cos(u)^2; simplify(a);
a := sin(u)^2 + cos(u)^2
```

Численные вычисления - альтернативный путь нахождения решения в тех случаях, когда символьный метод слишком медленно работает над данной задачей или решение в символьном виде вообще не существует. Maple поддерживает почти все существующие численные методы. Все символьные константы могут быть приближены с точностью до любого знака, так как среда Maple имеет "бесконечную точность".

Примеры численных вычислений:

```
> evalf(Pi, 25);
3.141592653589793238462643
```

```
> 0.5*A/3;
16666666667
```

```
> sum((-1)^i/i!, i=0..20);
```

$$\frac{4282366656425369}{11640679464960000} \quad (1)$$

Рассмотрим теперь, как можно решить нашу задачу о нахождении решения СЛАУ методом обратной матрицы в пакете Maple. Среда Maple позволяет выполнять все стандартные операции, определенные в линейной алгебре. Они становятся доступными при подключении библиотеки «**LinearAlgebra**». Библиотеки подключаются через команду **with** с указанием ее имени. В нашем случае запишем:

```
> with(LinearAlgebra):
```

Зададим матрицы A и B :

```
> A := <<<12, 4.2, -0.8> | <-4.1, 2.2, -0.16> |
      <-1.6, -4.3, 8.4>>>;
```

$$A = \begin{bmatrix} 12 & -4.1 & -1.6 \\ 4.2 & 2.2 & -4.3 \\ -0.8 & -0.16 & 8.4 \end{bmatrix}.$$

Мы видим, что ввод матрицы способом, задаваемым библиотекой LinearAlgebra, подразумевает ввод матрицы не по строкам, а по столбцам. В итоге нам нужно транспонировать матрицу A , чтобы привести её к изначальному виду:

```
> A := Transpose(A)
```

$$\begin{bmatrix} 12 & 4.2 & -0.8 \\ -4.1 & 2.2 & -0.16 \\ -1.6 & -4.3 & 8.4 \end{bmatrix}$$

Далее, вбиваем матрицу B :

`> B := <<-5.4, 1.6, 12.2>>`

Одной из доступных операций над матрицами, которые предоставляет LinearAlgebra, является операция нахождения обратной матрицы. Сделать это можно следующим образом:

`> MatrixInverse(A)`

$$\begin{bmatrix} 0.0519767039664955979 & -0.0930158641127034336 & 0.00317843153751951307 \\ 0.101359246898692135 & 0.290733002402520258 & 0.0151910330837329421 \\ 0.0617866057155677662 & 0.131110300922679973 & 0.127429396752390844 \end{bmatrix}$$

Далее, командой `%`.В получаем искомое нами решение:

$$\begin{bmatrix} -0.390722719241663685 \\ 0.103163474212636705 \\ 1.43076745099139013 \end{bmatrix}.$$

Графические возможности системы крайне богаты и позволяют строить все возможные типы графиков, визуализируя функции, векторные поля и другие объекты. Нарисуем график функции $F(x, y) = \sin^2(x) \cos(y)$, выполнив следующую команду:

`> plot3d(sin(x)^2*cos(y))`

Maple автоматически проставит границы интервалов для обеих переменных от -2π до 2π . Полученный график изображен на рис. 14. Дополнительные настройки графика проводились при помощи графического интерфейса. В частности, свойство “Color” было изменено на “Z (Hue)”.

Для мобильных платформ сейчас доступен Maple Companion. Этот крайне продвинутый калькулятор позволяет вводить условия задач, используя стандартные математические обозначения, сканировать примеры, написанные вручную на листе бумаги, решать системы уравнений, решать задачи по математическому анализу, работать с матрицами, производить многие виды символьных математических вычислений и преобразований.

К сожалению, мобильная версия Maple не позволяет работать с набором выражений, допуская ввод только одной строки. Несмотря на это приложение позволяет решить нашу СЛАУ, записав решение следующим образом:

$$\begin{bmatrix} 12 & 4.2 & -0.8 \\ -4.1 & 2.2 & -0.16 \\ -1.6 & -4.3 & 8.4 \end{bmatrix}^{-1} \cdot \begin{bmatrix} -5.4 \\ 1.6 \\ 12.2 \end{bmatrix}.$$

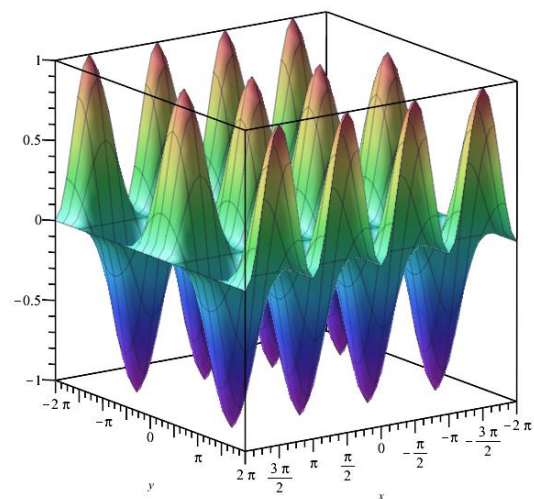


Рис. 14: График функции $\sin^2(x) \cos(y)$, построенный в пакете Maple всего одной короткой командой.

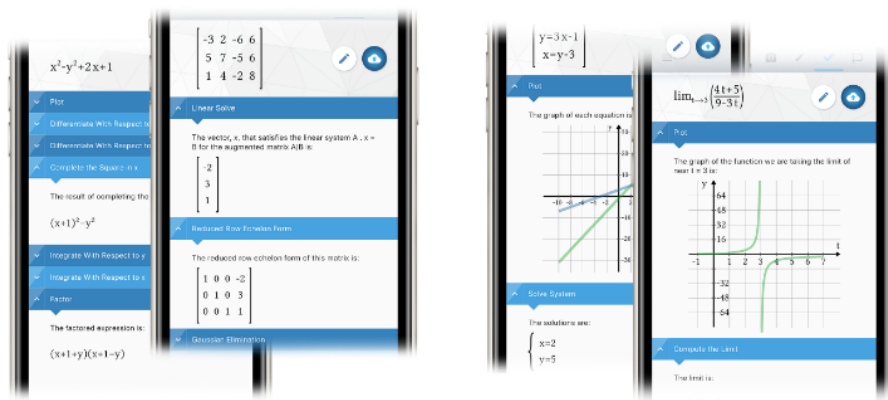


Рис. 15: Maple Companion – крайне интересное решение для мобильных платформ. Видны некоторые возможности системы: матричное исчисление, нахождение пределов функций, построение графиков функций, решение систем уравнений.

8 Mathcad

Mathcad позиционируется как система для проведения инженерных расчётов, которая способна производить математические вычисления. При этом она также обладает функционалом для символьных вычислений, хотя и довольно ограниченным.

Mathcad сейчас распространяется компанией РТС в новой версии Prime, во многом отличной от предыдущих версий, которые производила компания Mathsoft. По сути, сейчас в версии Prime это инженерно-научный блокнот, который ориентирован на поддержку концепции интерактивного рабочего листа. Уравнения и выражения отображаются на рабочем листе так, как они выглядели бы на какой-нибудь презентации, а не так, как выглядят на языке программирования.

Некоторые задачи, которые выполняет программа: решение дифференциальных уравнений, графики на плоскости и в пространстве, символьные вычисления, операции с векторами и матрицами, решение систем уравнений, построение графиков, набор статистических функций и вероятностных распределений.

Mathcad полезен для решения тех проблем, где требуется гибкость и творчество: при изучении какой-либо дисциплины, выполнении домашнего задания, проектировании, разработке нового инженерного продукта. В Mathcad можно даже написать книгу, с живыми расчетами и графиками. Освоить Mathcad достаточно легко, но все же на это требуется некоторое время. Интерфейс интуитивно понятен, но существует масса деталей, которыми нужно овладеть для успешной работы.

Примечательной чертой данной системы, выделяющей её из ряда других СКМ является работа с размерными величинами. В Mathcad числовые переменные и функции могут обладать размерностью. Сделано это для упрощения инженерных и физических расчетов. В Mathcad встроено большое количество единиц измерения, с помощью которых и создаются размерные переменные.

Над размерными переменными можно производить любые разумные с физической точки зрения расчеты. Работая с размерными переменными, приготовьтесь к тому, что Mathcad будет постоянно контролировать корректность расчетов. Например, нельзя складывать переменные разной размерности, в противном случае будет получено сообщение об ошибке "The units in this expression do not match" (Размерности в этом выражении не совпадают). Тем не менее, позволяет складывать, например, амперы с килоамперами и величины, размерность которых выражена в разных системах измерения (например, СИ и СГС).

Существенным недостатком данного пакета является его платформозависимость: работает он только под управлением семейства операционных си-

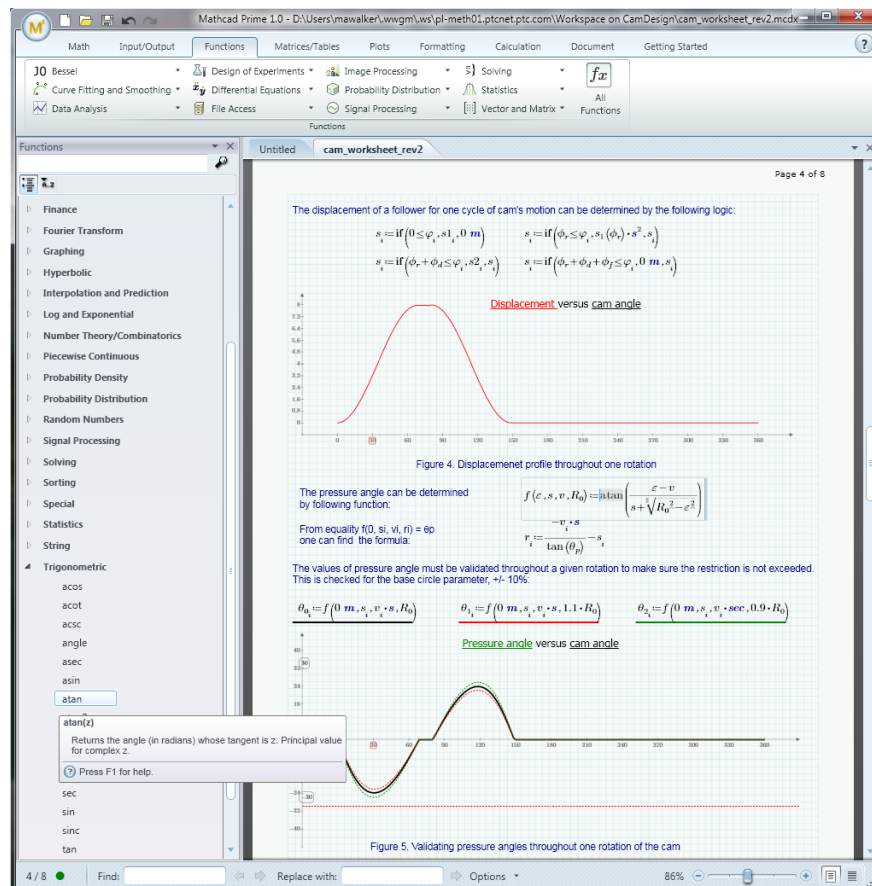


Рис. 16: Главное окно пакета Mathcad Prime. Показано, как можно работать с программой в её стихии - инженерном проектировании. При этом видно, что прямо в рабочем блокноте можно писать техническую документацию. Поддерживается символьный ввод формул с последующим их вычислением и выводом графиков.

$$A := \begin{bmatrix} 12 & 4.2 & -0.8 \\ -4.1 & 2.2 & -0.16 \\ -1.6 & -4.3 & 8.4 \end{bmatrix} \quad B := \begin{bmatrix} -5.4 \\ 1.6 \\ 12.2 \end{bmatrix} \quad x := A^{-1} B \quad x = \begin{bmatrix} -0.391 \\ 0.103 \\ 1.431 \end{bmatrix}$$

Рис. 17: Решение СЛАУ методом обратной матрицы в пакете Mathcad Prime.

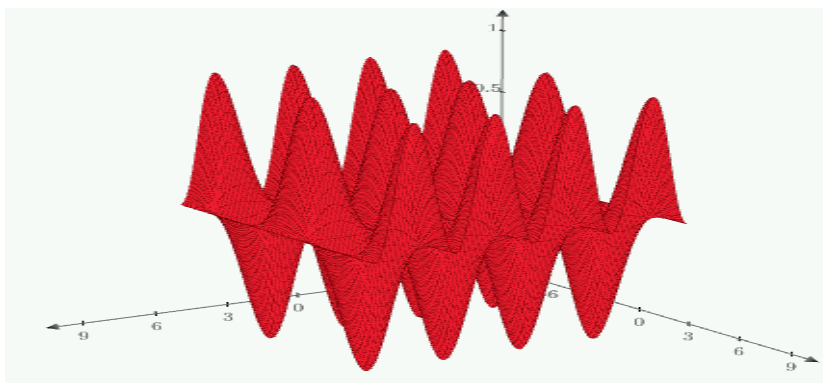


Рис. 18: Поверхность, задаваемая функцией $\sin(x)^2 \cos(y)$, построенная в пакете Mathcad Prime.

стем Windows. Также в разное время были обнаружены довольно серьёзные ошибки, которые допускает программа.

Основное отличие Mathcad от аналогичных программ — это графический, а не текстовый режим ввода выражений. Для набора команд, функций, формул можно использовать как клавиатуру, так и кнопки на многочисленных специальных панелях инструментов. В любом случае — формулы будут иметь привычный, аналогичный книжному, вид. То есть особой подготовки для набора формул не нужно. Вычисления с введенными формулами осуществляются по желанию пользователя: или мгновенно, одновременно с набором, либо по команде. Обычные формулы вычисляются слева направо и сверху вниз (подобно чтению текста). Любые переменные, формулы, параметры можно изменять, наблюдая воочию соответствующие изменения результата. Для использования Mathcad можно вообще не быть знакомым с программированием в том или ином виде.

Рассмотрим, как можно создать вектор или матрицу. Для этого необходимо открыть вкладку «Матрицы/таблицы». Когда курсор находится в пустой области щелчок по самой левой кнопке «Вставить матрицу» обеспечит появление сетки с маленькими белыми квадратами.

Перемещением указателя на сетку, можно выбрать желаемый размер матрицы. После щелчка левой кнопкой мыши появится пустая матрица. Мат-

рице можно присвоить имя, щелкнув на левую скобку, нажав `[:]` для оператора присваивания и введя имя. Вставку и удаление строк и столбцов легко осуществлять с помощью команд из меню «Операторы с векторами/матрицами» на вкладке «Матрицы и таблицы».

А вот графические возможности системы версии Prime по неведомой причине уступают предыдущим версиям и присутствуют только в базовой версии с минимально возможным набором свойств, поддающихся редактированию. Так, для поверхности нельзя задать цветовую схему, можно выбрать только однотонный цвет. Надеемся, в будущих версиях пользователю будет предоставлен гораздо более полный арсенал возможностей по настройке отображения графиков.

9 Средства на основе Python

Если вы знакомы с основами программирования на Python, то при освоении разделов математики можно с успехом использовать предназначенные для этого библиотеки и пакеты. Хотя изначально Python не является средством, ориентированным именно на проведение сколь-либо наукоемких вычислений, некоторые особенности языка привлекли разработчиков, и были созданы обширные наборы библиотек для осуществления математических расчётов различного уровня сложности. К плюсам языка программирования Python можно отнести динамическую типизацию, а также то, что язык является интерпретирующим и выполняется на лету, не требуя компиляции.

9.1 NumPy

Главным препятствием для научных вычислений на языке Python является отсутствие возможности работать со сложными структурами данных. Так, в языке нет такого понятия, как матрица или многомерный массив. Пакет NumPy решает эту проблему. Будучи библиотекой для языка Python, он добавляет поддержку больших многомерных массивов и матриц вместе с большой библиотекой высокоуровневых (и быстрых) математических функций для операций с этими массивами. NumPy является незаменимым помощником Python. Он приносит с собой анализ данных, машинное обучение и научные вычисления, а также существенно облегчает обработку векторов и матриц. Некоторые ведущие пакеты Python используют NumPy как основной элемент своей инфраструктуры. К их числу относится и SciPy, который будет рассмотрен в разделе 9.3.

Перед тем как использовать NumPy не забудьте импортировать библиотеку в проект:

```
>>>import numpy as np
```

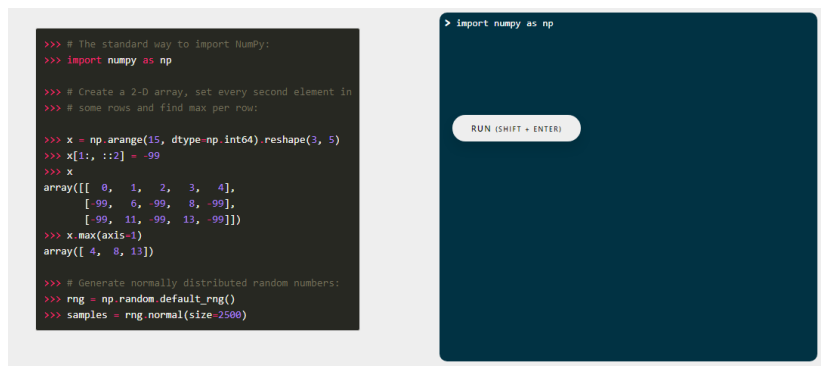


Рис. 19: Интерактивная оболочка, предоставляемая на сайте NumPy



Рис. 20: Создание матрицы при помощи NumPy. Матрица в данном случае представима в виде списка списков.

Основным объектом NumPy является однородный многомерный массив (в NumPy называется `numpy.ndarray`). Это многомерный массив элементов (обычно чисел) одного типа. Создается он при помощи функции `numpy.array()`:

```
>>> import numpy as np
>>> a = np.array([1, 2, 3])
>>> a
array([1, 2, 3])
>>> type(a)
<class 'numpy.ndarray'>
```

Функция `array()` трансформирует вложенные последовательности в многомерные массивы. Матрицу в NumPy можно создать, используя вложенные списки, таким образом матрица представляется в виде списка из списков:

```
>>> A = np.array([[12, 4.2, -0.8], [-4.1, 2.2, -0.16],
                  [-1.6, -4.3, 8.4]])
>>> A
array([[12.    ,  4.2    , -0.8    ],
       [-4.1    ,  2.2    , -0.16   ],
       [-1.6    , -4.3    ,  8.4    ]])
```

Для того чтобы посчитать обратную матрицу, можно воспользоваться функцией `numpy.linalg.inv()`. Умножение матриц выполняется оператором точки, который, однако, записывается не просто символом точки, а как `.dot()`. Таким образом, чтобы решить нашу задачу со СЛАУ, используем следующий код:

```
>>> A = np.array([[12, 4.2, -0.8], [-4.1, 2.2, -0.16],
                  [-1.6, -4.3, 8.4]])
>>> B = np.array([[-5.4], [1.6], [12.2]])
>>> x = np.linalg.inv(A).dot(B)
>>> x
array([[ -0.38142113],
       [ 0.07409017],
       [ 1.41765642]])
```

9.2 SymPy

SymPy представляет собой библиотеку символьных вычислений, которая в конечной цели стремится стать полноценной системой компьютерной алгебры, сохраняя при этом как можно более простой код, ясный для понимания и дальнейших изменений и дополнений. SymPy написан исключительно на Python и не требует никаких других библиотек. Она является реальной альтернативой таким математическим пакетам как Mathematica или Maple и обладает очень простым и легко расширяемым кодом.

Решение уравнений и систем, интегрирование и дифференцирование, вычисление пределов, разложение в ряд и суммирование рядов, поиск решения дифференциальных уравнений и систем, упрощение выражений – вот далеко не полный перечень аналитических возможностей пакета SymPy.

В библиотеке SymPy есть три встроенных численных типа данных: Real, Rational и Integer. С Real и Integer всё понятно, а класс Rational представляет рациональное число как пару чисел: числитель и знаменатель рациональной дроби. Таким образом, `Rational(1, 2)` представляет собой $1/2$, а, например, `Rational(5, 2)` — соответственно $5/2$.

Считается, что вычисление первообразных на первых курсах обучения является одной из наиболее сложных задач математического анализа, да и не все СКА справляются с отдельными примерами. Как показывает практика, во многих случаях вычисление первообразных в среде Python предпочтительнее, так как оно просто и наглядно, но в сложных примерах нужно пробовать использовать другие программы, например, Maxima, Maple или Mathematica. То же самое можно сказать и о пределах: результаты вычисления в Python достаточно простых пределов весьма наглядны, однако в более сложных случаях нужно использовать возможности других СКМ, рассмотренных в настоящей лекции.

Попробуем вычислить в SymPy предел $\lim_{x \rightarrow \infty} \left(1 + \frac{5}{x}\right)^{3x}$. Его решение сводится ко второму замечательному пределу.

Для аналитического вычисления пределов в SymPy имеется функция `limit()`; для записи символа ∞ (бесконечность) используется запись `oo` (две буквы «о»); символьные переменные объявляются при помощи команды `symbols()`:

```
>>> x=symbols('x')
>>> limit((1+5/x)**(3*x), x, oo)
e^15
```

Напомним, что в языке Python оператор «**» есть оператор возведения в степень.

Если функция `limit()` не может вычислить предел, например, он не существует, то в зависимости от режима `init_printing` она возвращает на экран либо строку с невычисляемым эквивалентом `Limit()`, либо невычисленное исходное выражение.

Онлайн-шелл для работы с библиотекой SymPy доступен по адресу <https://live.sympy.org/>.

9.3 SciPy

SciPy – это библиотека прикладных математических процедур, основанная на расширении NumPy. Реализует множество численных схем, включая поиск минимумов и максимумов функций, вычисление интегралов функций, поддержка специальных функций, работа с генетическими алгоритмами, решение обыкновенных дифференциальных уравнений и другие. Также позволяет обрабатывать сигналы и изображения.

Как видно, библиотека привносит в Python некоторые возможности MATLAB’a. Целевой аудиторией SciPy являются, соответственно, пользователи продуктов MATLAB и его свободных альтернатив. Специально для них SciPy предлагает даже функцию `loadmat()`, позволяющую загружать данные из файлов `.mat`, в которых MATLAB сохраняет данные в нативном формате. Такие файлы не являются текстовыми, поэтому их нельзя просто прочитать строка за строкой. Данная функция SciPy позволяет корректно открывать и работать с этими файлами.

Также именем SciPy назван проект, объединяющий уже рассмотренные нами библиотеки NumPy и SymPy, дающие возможность языку Python работать со сложными структурами данных и математическими объектами, с набором других библиотек, расширяющих возможности Python, включая, конечно же, и саму библиотеку SciPy.

Например, для визуализации результатов расчётов применяется библиотека Matplotlib, являющаяся аналогом средств вывода графики MATLAB.

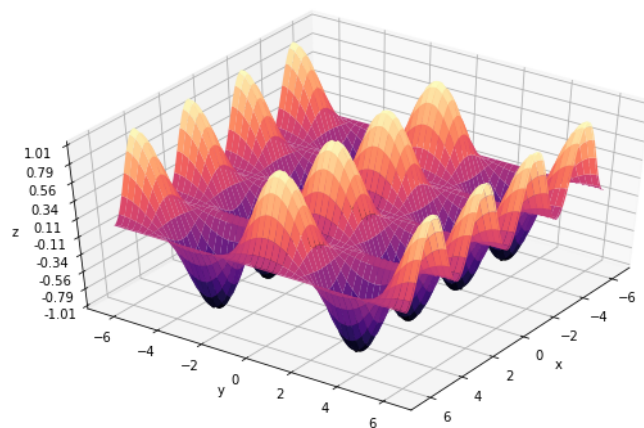


Рис. 21: Поверхность, задаваемая функцией $\sin(x)^2 \cos(y)$, построенная с использованием библиотеки Matplotlib.

Построенный с её помощью график трёхмерной поверхности изображен на рис. 21.

Также в набор включен пакет IPython – интерактивная консоль, работать с которой можно, как с командным окном, сразу получая ответ на введенное выражение. Таким образом благодаря SciPy в окружении Python выстраивается такая же полноценная система компьютерной математики, как MATLAB, Maple и Mathematica.