

РЕШЕНИЕ ЗАДАЧ ТЕОРЕТИЧЕСКОЙ МЕХАНИКИ В СРЕДЕ MAPLE

П. А. Кузнецов
С. В. Масло ассистент

Жизнь без компьютера в современном мире невозможна. Большая часть вещей быта каждого человека либо является компьютерами, либо находится под управлением компьютерных программ. Но жизнь современного высококвалифицированного инженера в полной мере связана с непосредственной работой и разработками на персональном или специализированном компьютере. За прошедшее десятилетие было создано огромное множество прикладных программ и оборудования, помогающих инженеру в его разработках и расчетах, а также ускоряющих непосредственные прикладные расчеты и производство. Одной из таких программ является Maple.

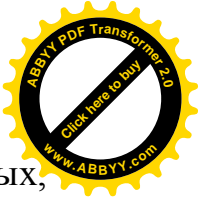
Maple — программный пакет, система компьютерной алгебры. Является продуктом компании Waterloo Maple Inc., которая с 1984 года выпускает программные продукты, ориентированные на сложные математические вычисления, визуализацию данных и моделирование. Система Maple предназначена для символьных вычислений, хотя имеет ряд средств и для численного решения дифференциальных уравнений и нахождения интегралов. Обладает развитыми графическими средствами. Имеет собственный язык программирования, напоминающий Паскаль.

Maple состоит из ядра - процедур, написанных на языке C и в высшей степени оптимизированных, библиотеки, написанной на Maple-языке, и интерфейса. Ядро выполняет большинство базисных операций. Библиотека содержит множество команд - процедур, выполняемых в режиме интерпретации. Программируя собственные процедуры, пользователь может пополнять ими стандартный набор и, таким образом, расширять возможности Maple.

Интерфейс Maple в настоящее время различается в зависимости от используемой техники и версии программы. В данной статье будет рассмотрен пример применения пакета для решения практических задач. Работа в Maple проходит в режиме сессии (session) - пользователь вводит предложения (команды, выражения, процедуры и др.), которые воспринимаются Maple.

Основные возможности пакета:

- ✓ Создание передаточных функций для моделей на основе дифференциальных уравнений, пространства состояний, полюсов и нулей коэффициентов усиления.
- ✓ Быстрое преобразование модели из одной формы в другую.
- ✓ Графический анализ: построение схем частотных характеристик, графиков корневых годографов, графическое изображение нулей и полюсов линейных систем.



- ✓ Генерации сигналов различных форм волн для создания импульсных, периодических, синусоидальных, шаговых, прямоугольных и треугольных тестовых сигналов.
- ✓ Имитация дискретных и непрерывных систем.
- ✓ Решение дифференциальных уравнений с помощью усовершенствованных алгоритмов решения стандартных дифференциальных уравнений (ODEs), дифференциальных уравнений с частными производными (PDEs) и дифференциальных алгебраических уравнений (DAEs).
- ✓ Использование новых алгоритмов для решения классов нелинейных стандартных дифференциальных уравнений 1 и 2 порядка и линейных дифференциальных уравнений 3 порядка.
- ✓ Применение новых алгоритмов для преобразования уравнений в формы, подходящие для решения в Maple.
- ✓ Усовершенствованные инструменты для работы с уравнениями с частными производными включают команды для работы с оператором Эйлера, сохраняющимися потоками и обобщенными интегрирующими множителями.
- ✓ Решение задач с задаваемыми пользователем событиями, параметрических задач, определение дискретных переменных при постановке задачи: в комбинации с событиями дискретные переменные могут быть использованы для определения критериев останова, условий возврата и многих других событий, имеющих место в процессе решений.
- ✓ Интерполяции кривых с возможностью просмотра и уточнения результатов благодаря команде ArrayInterpolation для многомерной интерполяции данных.
- ✓ Улучшенные опции для программирования.
- ✓ Возможность анализировать и решать системы параметрических полиномиальных уравнений и неравенств.
- ✓ Команда для вычисления тензора Кронекера произведения двух матриц.
- ✓ Преобразование кодов MATLAB в Maple.
- ✓ Интеграция с базами данных Microsoft SQL Server, Microsoft Access, Sybase, Oracle, IBM DB2 и MySQL.
- ✓ Возможность делать запросы, обновлять и создавать базы данных в среде Maple.

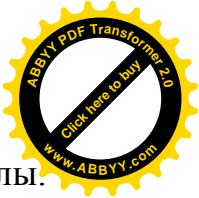
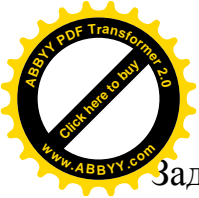
Рассмотрим программу в действии на примере решения задачи из курса теоретической механики.

Пример: Определить усилие в стержнях плоской фермы.

Решение: В основу расчёта положен метод вырезания узлов. В статически определимой неизменяемой ферме с N узлами, стержней всего $2N-3$. Для N узлов составляем $2N$ уравнений проекций на оси x , y . Включим в число стержней 3 опорные стержня. Таким образом, количество уравнений совпадает с числом неизвестных.

Описание программы. Сама программа состоит из 3-х блоков.

1. Ввод данных



Зададим число узлов и число стержней фермы, величины нагрузок (в кН), углы.

Оператор `evalf` переводит символ P_i в формат вещественного числа π .

Для алгоритмизации процесса и сомтавления уравнений пронумеруем стержни и узлы. Шарниры необходимы для ориентации опорных стержней. Для этих шарниров уравнения равновесия не составляются.

Координаты узлов x , y задаются в виде векторов (массивов) с $N+3$ компонентами. Начало координат помещаем в опоре A . Вектор заключён в скобки $\langle \rangle$, элементы отделены вертикальными чёрточками $|$.

Указываем номера точек приложения ($Nbeg$) векторов усилий в стержнях и номера их концов ($Nend$). Точка приложения выбирается условно, так как стержень подходит к двум узлам, и в качестве начала вектора усилия можно взять любой из них. На рис. 1 показана часть программы в среде Maple, где вводятся расчётные данные.

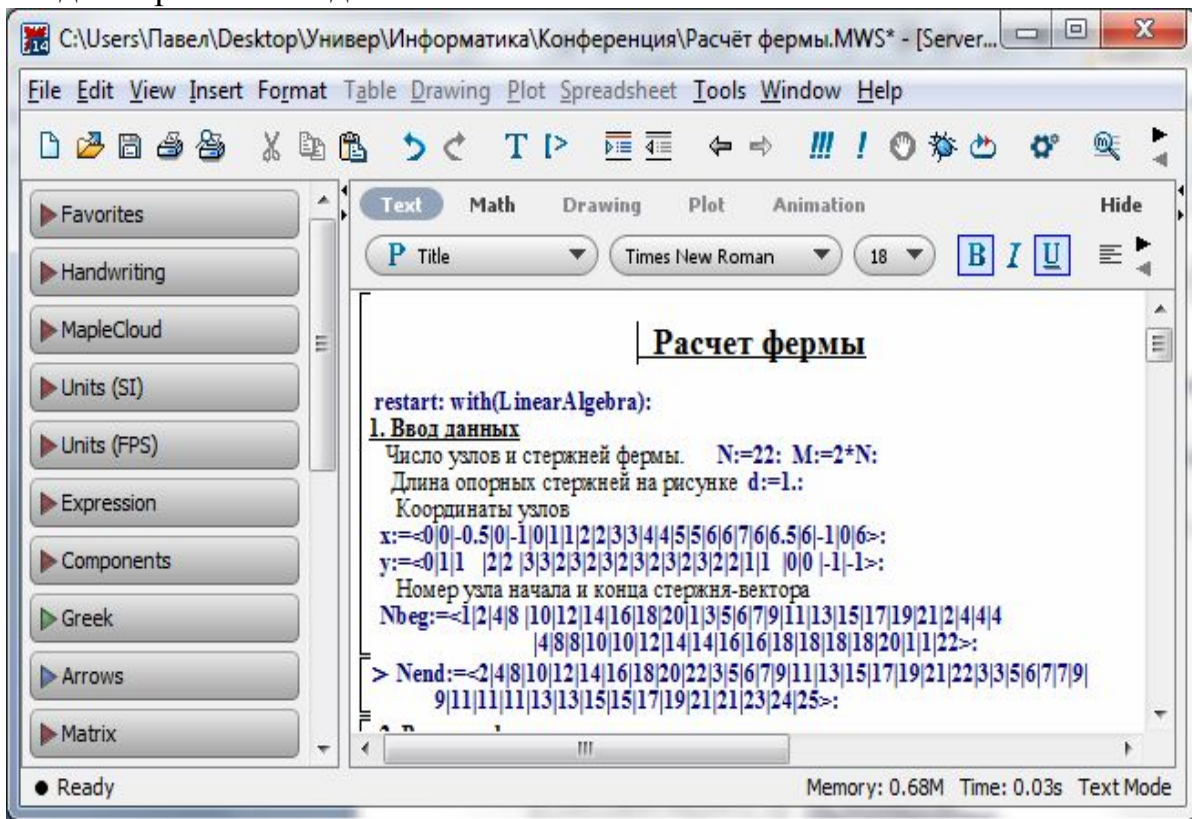


Рис 1. Ввод данных

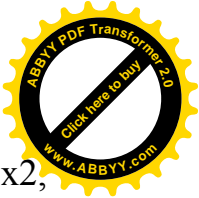
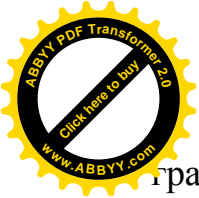
2. Рисунок фермы

Для контроля ввода геометрических величин изображаем ферму средствами графики Maple. Для этого задействуем оператор вывода графических объектов на экран `display`.

Изображение стержня с номером i присваиваем переменной $R[i]$ в цикле по числу стержней, включая опорные.

Одна из форм оператора цикла в Maple имеет вид `for i from i1 by i2 to i3 do ...od`.

Шаг `by i2` и начальное значение `from i1` по умолчанию приняты равными 1 и могут опускаться. Оператор `PLOT (CURVES ([[x1, y1],[x2, y2]]))` создает



графический объект – отрезок с концами в точках с координатами (x_1, y_1) и (x_2, y_2) .

Для вывода номеров стержней и узлов выбираем шрифт и размер. Эти значения заносим в переменную **Шрифт**. В Maple для переменных можно использовать русские буквы, программы при этом читаются легче. В отдельных циклах создаются графические объекты **Шарнир [i]** – номера узлов, **Стержень [i]** – номера стержней. Цвет номеров задаёт оператор **COLOR (HUE, n)**. Выбор цвета определяет дробная часть числа n. Можно также использовать оператор **COLOR (RGB, n1, n2, n3)** и выбирать цвет оттенками красного, зелёного и голубого, $0 < n_i < 1$. Оператор для изображения нагрузок – **arrow**. Параметры стрелок (толщина стрелки, толщина острия, отношение длины острия к длине стрелки и цвет) присвоены переменной **arw**.

Чтобы не писать длинный список объектов, для вывода изображения используется запись **seq(R[i], i=1..M)** – краткая форма простого перечисления $R[1], R[2], \dots, R[M]$.

В операторе **display** можно использовать различные режимы (опции); **scaling=constrained** – опция, задающая одинаковый по осям координат масштаб рисунка. Если после оператора **display** поставить не двоеточие, а точку с запятой, то на экране появится изображение фермы. На рис.2 изображена часть программы в среде Maple, где строится ферма.

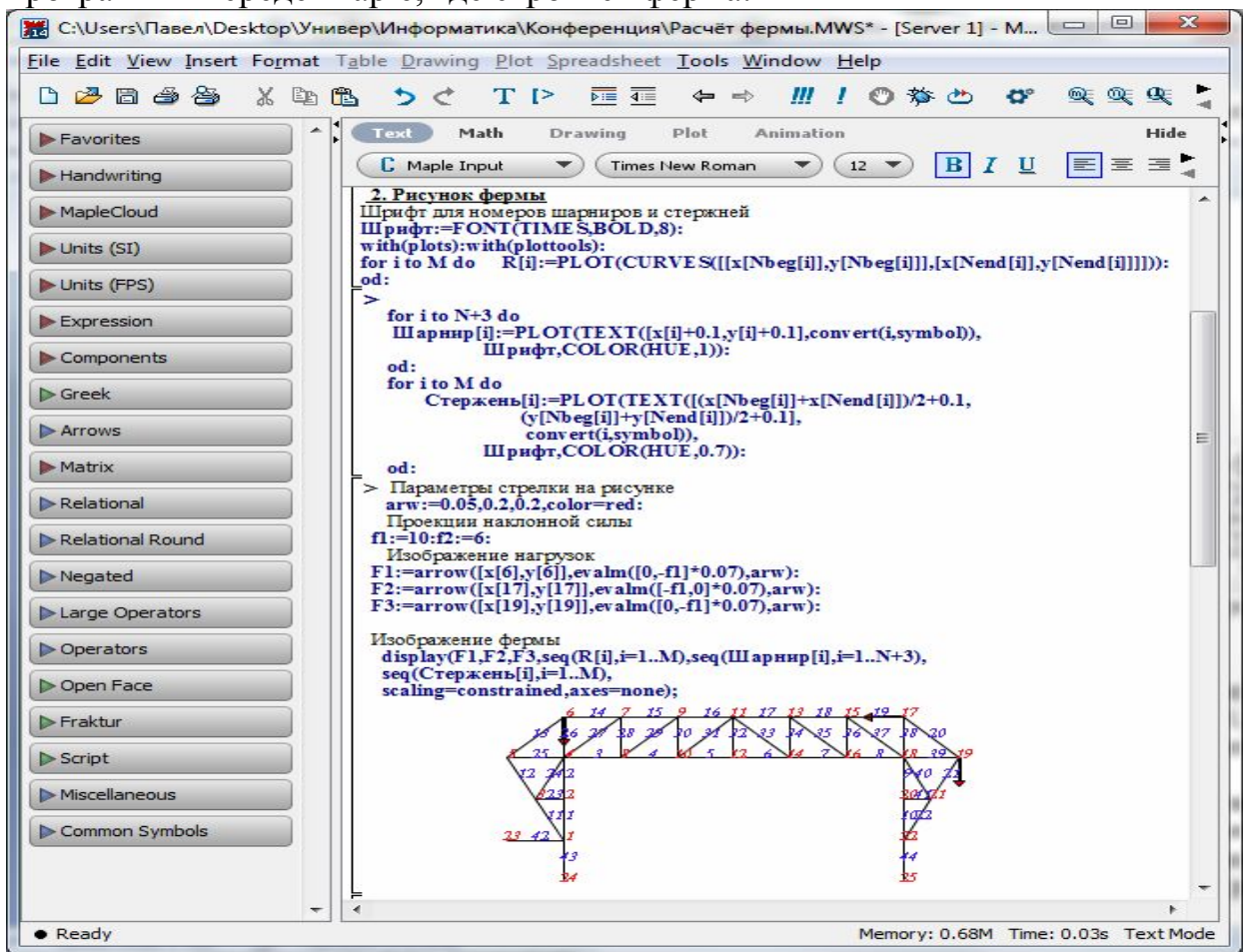


Рис. 2. Построение фермы



3. Вычисления

Косинус угла усилия $S[i]$, $i=1, \dots, M$ с осью x , приложенного к узлу j , поместим в переменную типа двумерного массива $G[2j-1, i]$. Косинус угла этого же усилия с осью y – в $G[2j, i]$. Элементы G предварительно обнуляются командой `restart`.

В теле цикла вычисляются $G[i, j]$. Каждое из неизвестных усилий S_i входит в систему уравнений дважды: в уравнений равновесия узлов $Nbeg[i]$ и $Nend[i]$ с разными знаками. Только усилия опорных стержней с номерами $M-2, M-1, M$ появляются в уравнениях равновесия по одному разу., отсюда необходимость условного оператора `if < M-2... fi`.

Проекции нагрузки F_{jx} и F_{jy} , приложенной к узлу j заносятся соответственно в элементы $2j-1$ и $2j$ вектора B в правой части системы уравнений.

`LinearSolve (A, B)` - оператор решений системы уравнений с матрицей A и вектором правой части B . Для того, чтобы этот оператор работал, необходимо загрузить специальную библиотеку программ командой `with (Linear Algebra)`. Величины $G[i, j]$ не образуют матрицу, поэтому для обращения к оператору решения `LinearSolve` используем преобразование `Matrix (M, G)`. В результате решения значения усилий присваиваются элементам вектора S .

Для вывода на печать чисел и символов используем оператор форматного вывода `printf`. На рис.3 приведена часть программы в среде Maple, где происходит вычислительный процесс.

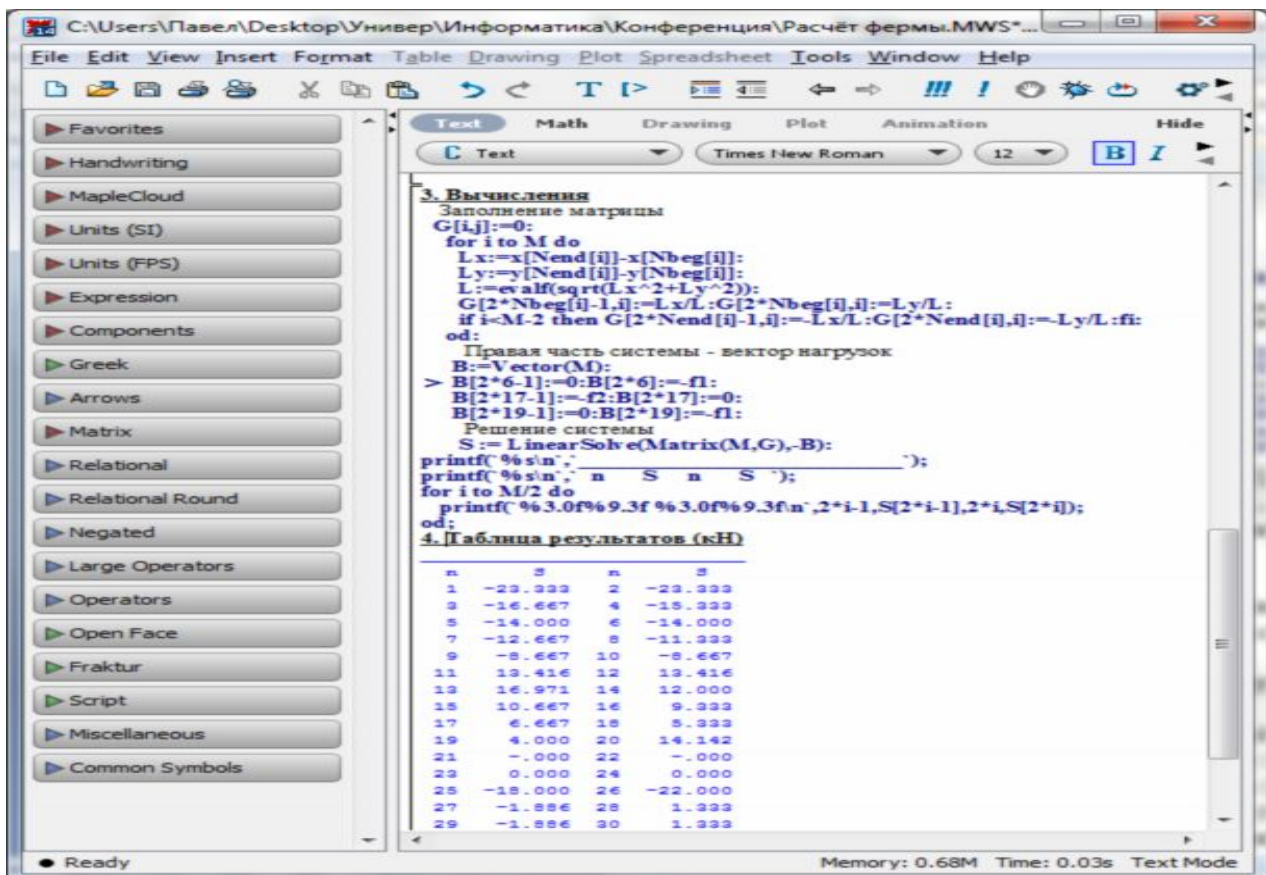




Рис. 3. Вычисление результата.

Таким образом, в данной статье проанализировано решение задачи из курса теоретической механики в среде Maple. Эта программа также применима не только для решения статических задач. Её возможности могут быть использованы для решения инженерных задач любого рода: статических, динамических, задач электромеханики и электродинамики. В этом пакете находится множество инструментов, которые делают инженерные расчёты простыми и безошибочными.

Список использованных источников

1. Теоретическая механика. Решебник./ Кирсанов М.Н.- М.- ФИЗМАТЛИТ, 2002.
2. <http://ru.wikipedia.org>
3. Использование среды Maple для решения задач квантовой механики./ Чернушкин В. В. Воронежский Государственный Университет. Методические указания к специальному практикуму «Решение задач по квантовой механике» для студентов физического факультета дневного отделения обучающихся по направлению 010400 «Физика».