

## **Использование математического пакета Maple при решении задач классической оптимизации**

**# 01, январь 2016**

**Шевченко А. С.**

УДК: 517, 519.67

Рубцовский институт (филиал) Алтайского государственного  
университета, Рубцовск, Россия  
[ibragimova.a.s@mail.ru](mailto:ibragimova.a.s@mail.ru)

### **Введение**

Одной из ключевых компетентностей будущего специалиста как экономического, так и технического профилей является способность применения математических методов в сочетании с информационными технологиями. Способность достижения значимых результатов в профессиональной деятельности часто напрямую связана с осведомленностью о методах и способах решения математических задач с использованием специального программного обеспечения. Владение хотя бы одним из математических пакетов (Maple, Mathematica, MathCAD, MatLab, Maxima и др.) позволяет будущему специалисту, не владеющему в полной мере техникой математических преобразований, самостоятельно выполнять громоздкие вычисления, решать сложные прикладные задачи.

Уникальные возможности и чрезвычайная простота интерфейса системы Maple, сделали ее одной из самых популярных и, безусловно, самым распространенным математическим пакетом. Она используется при решении различных задач. Возможности применения пакета аналитических вычислений Maple рассмотрены в многочисленных работах и пособиях, в частности [1-2].

Новые версии Maple дают новые средства для решения разнообразных задач, в том числе, и задач оптимизации. Мощным средством решения задач оптимизации (как линейной, так и нелинейной) в Maple является профессиональный (специализированный) пакет Global Optimization Toolbox, который, однако, не поставляется в базовой версии продукта и покупается отдельно, что требует значительных финансовых вложений. Поэтому в данной работе мы решили показать, как можно использовать возможности системы Maple для решения задач оптимизации, в частности, нелинейного программирования, без привлечения специализированных пакетов.

Приведем примеры исследования функции двух переменных с использованием системы Maple.

## 1. Задача на безусловный экстремум

**Пример 1.** Исследовать на экстремум функцию  $f = x^4 + y^4 - x^2 - 2 \cdot x \cdot y - y^2$ .

1) Зададим функцию с помощью функционального оператора:

```
> f := (x, y) -> x^4 + y^4 - x^2 - 2*x*y - y^2;
```

Каждый оператор или команда должны обязательно завершаться разделительным знаком: точкой с запятой (;) или двоеточием (:). Если ввод предложение завершается точкой с запятой, то в строке под предложением сразу будет отклик: результат исполнения команды или сообщение об ошибке. Для нашего примера получаем:

$$f := (x, y) \rightarrow x^4 + y^4 - x^2 - 2xy - y^2$$

Разделитель (:) используется для отмены вывода, когда команда выполняется системой, но ее результат не выводится на экран.

2) Прежде всего, нарисуем (трехмерный) график нашей функции. Для этого необходимо подключить пакет расширения **plots** с помощью команды:

```
> with(plots): plot3d(f(x,y), x=-2.5..2.5, y=-2.5..2.5, axes=frame);
```

В этой команде указана опция **axes**, которая задает вид координатных осей. На самом деле, в данной команде, можно указывать большое число различных опций, которые задают вид графиков: цвет, тип и толщину линий, угол поворота осей и др. Их подробное описание можно найти в Help.

Итак, мы получаем график, изображенный на рис. 1а.

Теперь построим контурный график, только в команде plot3d укажем опцию **style** и присвоим значение **patchcontour**:

```
> plot3d(f(x,y), x=-2.5..2.5, y=-2.5..2.5, style=patchcontour, axes=frame, contours=20);
```

Опция **contours** задает число линий уровня функций на поверхности.

Получаем график, изображенный на рис. 1б.

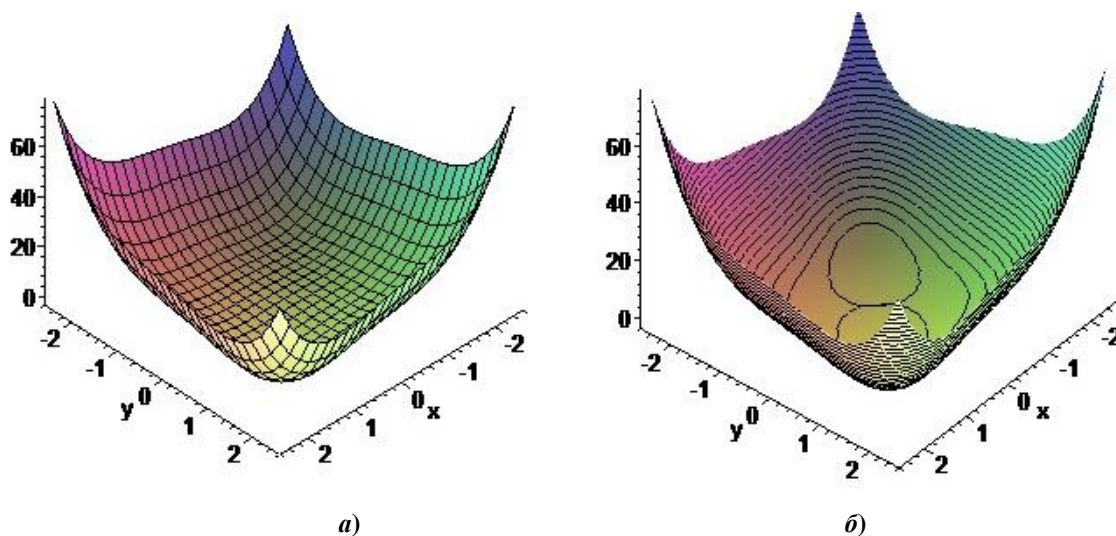


Рис. 1. Трехмерный график функции  $f(x,y)$

На контурном графике чуть лучше виден характер функции, но все равно для определения экстремумов удобнее использовать двумерные линии уровня функции. Их можно получить таким образом (рис. 2):

```
> contourplot(f(x,y), x=-2.5..2.5, y=-2.5.. 2.5, axes=frame,
filled=true, contours=50, coloring=[yellow, green]);
```

В этой команде опция **filled** равна **true**. Это означает, что Maple автоматически закрашивает области между линиями уровня в разные цвета в зависимости от значения функции  $f$  на этих линиях уровня. Это делает график более выразительным и позволяет визуально оценить характер изменения значений функции. Если положить опцию **filled** равной **false** (или вообще не указывать), то получим не закрашенный график. Напротив, если положить **filled=true**, то можно использовать опцию **coloring**, которая позволяет управлять цветами закрашки. Например, если указана **coloring=[yellow, green]**, то линии уровня, отвечающие наименьшему значению  $f$ , будут желтыми, а линии уровня, отвечающие наибольшему значению  $f$  – зелеными, а все промежуточные линии уровня Maple закрасит в промежуточные цвета.

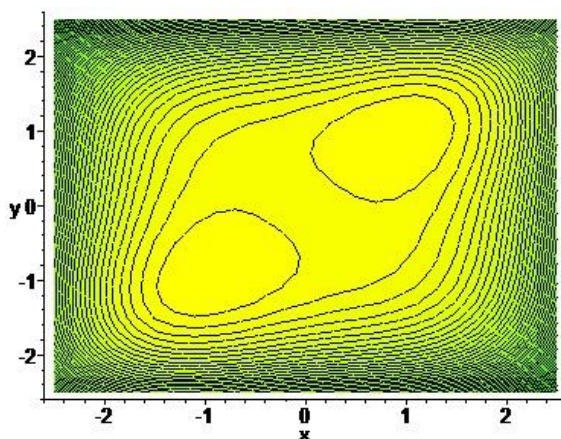


Рис. 2. Двумерный график линий уровня функции  $f(x,y)$

Глядя на эти рисунки, можно высказать предположение, что наша функция имеет два экстремума, причем из характера графика, полученного выше, следует, что это точки минимума.

Далее продемонстрируем, как можно решить задачу нахождения экстремумов нашей функции аналитически [5,6].

3) Применим необходимое условие первого порядка. Для нахождения критических точек функции зададим систему уравнений  $\frac{\partial f}{\partial x}(x,y)=0$ ,  $\frac{\partial f}{\partial y}(x,y)=0$ :

```
> SysEqn:={diff(f(x,y), x)=0, diff(f(x,y), y)=0};
```

Получаем:

$$\text{SysEqn} := \{ 4x^3 - 2x - 2y = 0, 4y^3 - 2x - 2y = 0 \} .$$

4) Решаем данную систему с помощью команды **solve**:

```
> rez:=solve(SysEqn,{x, y});
```

Maple выдает следующий результат:

```
rez := { x = 0, y = 0 }, { x = 0, y = 0 }, { x = 0, y = 0 }, { x = 1, y = 1 }, { x = -1, y = -1 }, { x
      = RootOf(-RootOf(_Z^2 + _Z + 1, label = _L29) + 2 _Z^2 - 1, label = _L28)
      RootOf(_Z^2 + _Z + 1, label = _L29),
      y = RootOf(-RootOf(_Z^2 + _Z + 1, label = _L29) + 2 _Z^2 - 1, label = _L28) }
```

Видим, что результат содержит функцию **RootOf**. Это означает, что аналитические решения системы нельзя выразить в радикалах. В таком случае можно попытаться получить численные, приближенные значения решений с помощью команды **evalf**. Эта команда служит для принудительного приближенного вычисления результата: **evalf(%)**.

После применения этой команды получаем:

```
{ x = 0., y = 0. }, { x = 0., y = 0. }, { x = 0., y = 0. }, { x = 1., y = 1. }, { x = -1., y = -1. },
{ x = 0.6123724357 + 0.3535533906 I, y = -0.6123724357 + 0.3535533906 I }
```

Другой способ «бороться» с функцией **RootOf** заключается в том, чтобы в ответ исходной системы искусственно внести неточность, чтобы Maple не пыталась найти точное решение. Это можно сделать, например, так: вместо какого-нибудь целого числа в записи уравнений системы, например, числа 0, написать число 0.0. Maple воспринимает это число как нецелое и автоматически ищет приближенное решение такой системы.

Итак, мы получили такие (вещественные) корни: точки (0,0) (кратности три), (1,1) и (-1,-1).

5) Далее применим условия экстремума второго порядка. Загружаем пакет линейной алгебры **linalg** и вычисляем Гессиан с помощью команды **hessian**:

```
> with(linalg): MatrixGesse:=hessian(f(x,y), [x,y]);
```

Гессиан имеет следующий вид:

$$MatrixGesse := \begin{bmatrix} 12x^2 - 2 & -2 \\ -2 & 12y^2 - 2 \end{bmatrix}.$$

6) Подставляем в Гессиан координаты стационарных точек с помощью оператора **subs** и вычисляем его главные диагональные миноры:

```
> MatrixGesse:=subs(rez[1], hessian(f(x,y), [x,y]));
M11:=MatrixGesse[1,1]; M12:=det(MatrixGesse);
```

Результат подстановки точки (0,0):

$$MatrixGesse := \begin{bmatrix} -2 & -2 \\ -2 & -2 \end{bmatrix}$$

$$M11 := -2$$

$$M12 := 0$$

Из результата видно, что точка (0,0) не является точкой экстремума функции.

Положительную или отрицательную определенность матрицы можно выяснить при помощи команды **definite** (*A*, *param*), где *param* может принимать значения: 'positive\_def' – положительно определена ( $A > 0$ ), 'positive\_semidef' – не отрицательно определенная ( $A \geq 0$ ), 'negative\_def' – отрицательно определенная ( $A < 0$ ), 'negative\_semidef' – не положительно определенная ( $A \leq 0$ ). Результатом действия будет константа true – подтверждение, false – отрицание сделанного предположения:

```
> definite(MatrixGesse, 'positive_def'); definite(MatrixGesse,
'negative_def');
```

*false*

*false*

Подставляем точку (1;1). У матрицы Гессе вычисляем главные диагональные миноры:

```
> MatrixGesse:=subs(rez[4], hessian(f(x,y), [x,y]));
```

```
M12:=MatrixGesse[1,1]; M22:=det(MatrixGesse);
```

```
definite(MatrixGesse, 'positive_def');
```

Результат подстановки точки (1,1):

$$MatrixGesse := \begin{bmatrix} 10. & -2 \\ -2 & 10. \end{bmatrix}$$

*M12 := 10.*

*M22 := 96.*

*true*

Т.к. все главные диагональные миноры матрицы Гессе положительны ( $M12=10>0$ ,  $M22=96>0$ ), то точка (1,1) является точка минимума функции.

Вычисляем значение функции в данной точке:

```
> F:=subs(rez[4], f(x, y));
```

Получаем:

*F := -2.*

Подставляем точку (-1; -1):

```
> MatrixGesse:=subs(rez[5], hessian(f(x,y), [x,y]));
```

```
M13:=MatrixGesse[1,1]; M23:=det(MatrixGesse);
```

```
definite(MatrixGesse, 'positive_def');
```

Результат подстановки точки (-1,-1):

$$MatrixGesse := \begin{bmatrix} 10. & -2 \\ -2 & 10. \end{bmatrix}$$

*M13 := 10.*

*M23 := 96.*

*true*

Т.к. все главные диагональные миноры матрицы Гессе положительны ( $M13=10>0$ ,  $M23=96>0$ ), то точка (-1,-1) точка минимума функции.

Вычисляем значение функции в данной точке.

```
> F:=subs(rez[5], f(x, y));
```

Получаем:

*F := -2.*

## 2. Задача на условный экстремум

**Пример 2.** Найти условный экстремум для функции двух переменных при наличии одного ограничения:

$$f(x, y) := (x-2)^2 + (y-1)^2 \rightarrow \text{extr},$$

$$g(x, y) = x^2 + y^2 - 1.$$

1) Зададим нашу функцию и ограничение:

```
> restart: f:=(x-2)^2+(y-1)^2; g:=x^2+y^2-1.;
```

2) Изобразим на плоскости  $xOy$  линии уровня функции  $f(x, y)$  и ограничение  $g(x, y) = 0$ :

```
>with(plots):
>g1:=implicitplot(g(x,y)=0, x=-1..1, y=-1..1, scaling=constrained):
>g2:=contourplot(f(x,y), x=-1.5..1.5, y=-1.5..1.5, scaling=constrained, filled=false, coloring=[red, green], contours=30):
>display(g1,g2);
```

Опция **scaling=constrained** означает, что масштабы по осям  $Ox$  и  $Oy$  на графике будут выбраны одинаковыми. Этот фрагмент программы выдает рисунок, приведенный на рис.3. Полученный график позволяет локализовать точки минимума и максимума, которые являются точками касания окружности  $g(x, y) = 0$  и линий уровня функции  $f(x, y)$ .

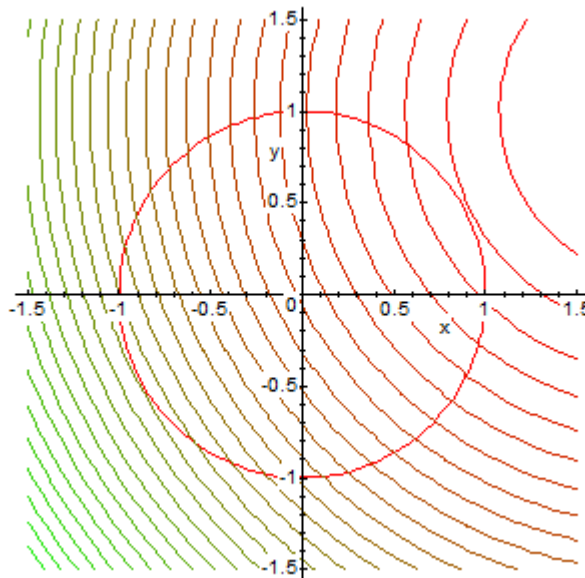


Рис.3. Линии уровня функции  $f(x, y)$  и ограничение  $g(x, y) = 0$

Найдем экстремальные точки нашей функции аналитически [5,6].

3) Зададим функцию Лагранжа  $L(x, y, \lambda) = f(x, y) + \lambda g(x, y)$ :

```
>L:=f+lambda*g;
```

Ее вид:

$$L := (x-2)^2 + (y-1)^2 + \lambda (x^2 + y^2 - 1)$$

4) Найдем частные производные функции Лагранжа и приравняем их к нулю:

```
>SysEqn:={diff(L,x)=0, diff(L,y)=0, diff(L,lambda)=0};
```

В результате получаем такую систему уравнений:

$$\text{SysEqn} := \{ 2x - 4 + 2\lambda x = 0, 2y - 2 + 2\lambda y = 0, x^2 + y^2 - 1 = 0 \}$$

5) Решаем данную систему с помощью команды solve:

```
> rez:=solve(SysEqn, {x,y,lambda});
```

Получаем два различных решения данной системы:

$$\text{rez} := \{ \lambda = -3.236067978, y = -.4472135955, x = -.8944271910 \}, \\ \{ \lambda = 1.236067978, x = .8944271910, y = .4472135955 \}$$

В этой команде можно, вообще говоря, и не указывать набор переменных, относительно которых решается система.

Значит, одна из найденных точек минимум, а вторая максимум, достаточно просто подставить их в функцию:

```
> F1:=subs(rez[1], f); F2:=subs(rez[2], f);
```

Значения функции  $f(x, y)$  в найденных точках:

$$F1 := 1.527864045 \\ F2 := 10.47213596$$

В данном случае оказалось, что задачу нахождения экстремумов можно решить, не применяя необходимых условий условного экстремума второго порядка. Однако для полноты изложения покажем, как применить эти условия. Вначале построим второй дифференциал функции Лагранжа. Можно действовать так.

6) Построим второй дифференциал функции Лагранжа, как функцию четырех переменных  $x, y, dx, dy$ :

```
>DDL:=diff(L, x$2)*dx^2+2*(diff(L, x, y))*dx*dy+diff(L, y$2)*dy^2;
```

Получаем:

$$DDL := (2 + 2\lambda) dx^2 + (2 + 2\lambda) dy^2$$

7) Находим связь между дифференциалами  $dx, dy$ , которая выражается равенством

$$dg(x, y) = \frac{\partial g(x, y)}{\partial x} dx + \frac{\partial g(x, y)}{\partial y} dy = 0:$$

```
>dg:=(diff(g, x))*dx+(diff(g, y))*dy=0;
```

Полученная связь имеет следующий вид:

$$dg := 2x dx + 2y dy = 0$$

8) Из равенства  $dg(x, y) = \frac{\partial g(x, y)}{\partial x} dx + \frac{\partial g(x, y)}{\partial y} dy = 0$  выражаем  $dy$  через  $dx$  с помощью команды solve:

```
>dy:=solve(dg, dy);
```

Выражение для  $dy$  следующее:

$$dy := -\frac{x dx}{y}$$



9) Теперь возвращаем функцию DLL от трех переменных  $x, y, dx$  (выражение для  $dy$  уже подставлено):

```
>eq:=DDL;
```

Имеем:

$$eq := (2 + 2 \lambda) dx^2 + \frac{(2 + 2 \lambda) x^2 dx^2}{y^2}$$

10) Для удобства можно привести подобные с помощью команды collect:

```
>collect(eq, dx);
```

Тогда второй дифференциал примет вид:

$$\left( 2 + 2 \lambda + \frac{(2 + 2 \lambda) x^2}{y^2} \right) dx^2$$

11) Найдем значения второго дифференциала в найденных точках:

```
>for i from 1 to 2 do assign(rez[i]); X[i]:=x; Y[i]:=y; Lambda[i]:=lambda; print('DDL'=DDL): x:='x': y:='y': lambda:='lambda': od;
```

Результаты решения данного фрагмента:

$$X_1 := 0.8944271910$$

$$Y_1 := 0.4472135955$$

$$\Lambda_1 := 1.236067978$$

$$DDL = 22.36067978 dx^2$$

$$x := x$$

$$y := y$$

$$\lambda := \lambda$$

$$X_2 := -0.8944271910$$

$$Y_2 := -0.4472135955$$

$$\Lambda_2 := -3.236067978$$

$$DDL = -22.36067978 dx^2$$

$$x := x$$

$$y := y$$

$$\lambda := \lambda$$

Получаем, что при  $x = 0.894427191, y = 0.4472135955, \lambda = 1.236067978$  второй дифференциал положителен (т.е. этому значению соответствует минимум функции  $f(x, y)$ ), а при  $x = -0.894427191, y = -0.4472135955, \lambda = -3.236067978$  – отрицателен (т.е. этому значению соответствует максимум функции  $f(x, y)$ ).



## Заключение

Участие одновременно педагога и компьютера в процессе обучения значительно улучшает качество образования. Использование возможностей математического пакета Maple активизирует процесс преподавания, повышает интерес студентов к изучаемой дисциплине и эффективность учебного процесса, позволяет достичь большей глубины понимания учебного материала.

Применение Maple при решении задач по нелинейной оптимизации может представлять интерес не только для студентов и аспирантов, но и для преподавателей, поскольку освобождает от больших интеллектуальных усилий и неоправданных затрат времени.

Кроме того, сходство синтаксиса команд Maple и Maxima (распространяется свободно) позволяет достаточно легко использовать данный подход для решения задач на свободно распространяемом (GPL) продукте wx Maxima.

## Список литературы

- [1]. Алексеев Е.Р., Чеснокова О.В. Решение задач вычислительной математики в пакетах Mathcad 12, Matlab 7, Maple 9 (самоучитель). М.: НТ Пресс. 2006. 496 с.
- [2]. Дьяконов В.П. Maple 9.5/10/11 в математике, физике и образовании. 2-е изд., перераб. и доп. М.: ДМК Пресс, Солон-Пресс. 2011. 752 с.
- [3]. Аладьев В.З., Бойко В.К., Ровба Е.А. Программирование и разработка приложений в Maple. Таллинн: Межд. Акад. Ноосферы. 2007. 458 с.
- [4]. Сдвижков О.А. Математика на компьютере: Maple 8. М.: Солон-Пресс. 2003. 176 с.
- [5]. Горбунова Р.И., Курганова М.В., Макаров С.И., Мищенко М.В., Нуйкина Е.Ю., Севастьянова С.А., Сизиков А.П., Уфимцева Л.И., Фомин В.И., Чупрынов Б.П., Черкасова Т.Н. Экономико-математические методы и модели. Задачник: учебно-практическое пособие / под ред. С.И. Макарова, С.А. Севастьяновой. 2-е изд., перераб. М.: КноРус. 2009. 208 с.
- [6]. Пантелеев А.В., Летова Т.А. Методы оптимизации в примерах и задачах: Учебное пособие. 2-е изд., исправл. М.: Высш. Шк., 2005. 544 с.