

Путешествие окружности в треугольнике, а треугольника в ложбине, или Сам себе компьютерный режиссер

В статье рассмотрена необычная возможность использования анимации для изучения численных методов решения математических задач. Под анимацией понимается не только и не столько создание анимационных клипов, но непосредственное «оживление» результатов реализации численных методов.

Ключевые слова: численные и аналитические методы решения математических задач, Mathcad, анимация, Первое апреля.

TRAVEL OF A CIRCLE IN A TRIANGLE, AND THE TRIANGLE IN THE HOLLOW OR SELF COMPUTER DIRECTOR

The article describes an unusual possibility to use animation to study numerical methods for solving mathematical problems. Animation is defined not only and not so much as the creation of animation clips, but direct «revival» of the results of the implementation of numerical methods for solving mathematical problems on your computer

Keywords: Numerical and analytical methods for solving mathematical problems, Mathcad, animation, April Fool's Day.

Сразу без особого вступления «берем быка за рога». Правда, небольшое вступление было дано в аннотации.

Задача: дан треугольник (декартовы координаты трех его вершин (x_1, y_1) , (x_2, y_2) и (x_3, y_3)). Найти параметры вписанной в этот треугольник окружности: ее радиус r и декартовы координаты ее центра (x_r, y_r) (рис. 1).

Можно попытаться поискать в справочниках или интернете гото-

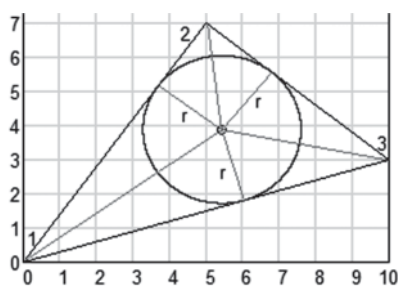


Рис. 1. Задача об окружности, вписанной в треугольник

вые формулы, по которым рассчитываются значения искомых величин r , x_r и y_r по значениям заданных переменных x_1, y_1, x_2, y_2, x_3 и y_3 . Но готовых формул для всех трех величин r , x_r и y_r найти не удастся (их и быть не может, вернее, пока не может: читаем статью дальше). Можно найти только одну очень изящную формулу, по которой рассчитывается радиус вписанной в треугольник окружности в зависимости от его площади (S) и полупериметра (p)¹ или в зависимости от

¹ Нормальный человек под переменной p будет понимать периметр (p – perimeter) треугольника – сумму длин его сторон. Но в этой формуле радиуса вписанной окружности для ее простоты математики под переменную p зарезервировали полупериметр. А сам периметр обозначается заглавной буквой P , так же как площадь обозначается заглавной буквой S . И это очень путает многих. Истинно сказано: «Иная простота хуже воровства». Одного из авторов, например, до сих пор смущает и путает тот факт, что в формуле площади круга стоит его радиус в квадрате, а в формуле

длин сторон треугольника (a, b и c) и полупериметра (площадь треугольника выражена формулой Герона $S_{\Delta} = \sqrt{p(p-a)(p-b)(p-c)}$, которую изучают в школе):

$$r = \frac{S}{p} = \sqrt{\frac{(p-a)(p-b)(p-c)}{p}}$$

Величины $S, p, a, b,$ и c нужно будет рассчитывать по другим формулам, которые опять же нужно где-то искать или выводить самому. Длины сторон треугольника можно, например, вычислить по значениям декартовых координат вер-

шины окружности этот радиус нужно помножить еще и на двойку, т.е. работать уже с диаметром, а не радиусом. Отсюда вывод. Вводя в расчет даже простейшую формулу (формулу для площади круга, например), чтобы не ошибиться, стоит себя подстраховать: искать формулу не в своей голове, а в справочниках – бумажных или интернетовских. Вернее, так: сначала попытаться найти в голове (это очень хорошо для тренировки памяти), а потом сравнить найденное с формулой из справочника.



Валерий Федорович Очков,
д.т.н., профессор
Тел.: (495) 362-71-71
Эл. почта: ochkov@twi.mpei.ac.ru
Национальный исследовательский
университет «Московский
энергетический институт»
<http://www.mpei.ru>

Valerij F. Ochkov
D. Sc. in Engineering, Professor
Тел.: (495) 362-71-71
E-mail: ochkov@twi.mpei.ac.ru
National research university «Moscow
Power Engineering Institute»
<http://www.mpei.ru>



Богомолова Елена Петровна,
к.ф.-м.н., доцент
Тел.: (495) 362-73-92
Эл. почта: epbogomolova@yandex.ru
Национальный исследовательский
университет «МЭИ»
<http://www.mpei.ru>

Elena P. Bogomolova,
Ph.D., Associate Professor
Тел.: (495) 362-73-92
E-mail: epbogomolova@yandex.ru
National Research University
“Moscow Power Engineering Institute”
<http://www.mpei.ru>

$$x_1 := 0m \quad y_1 := 0m \quad x_2 := 5m \quad y_2 := 7m$$

$$a := \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} = 8.602m$$

$$x_3 := 10m \quad y_3 := 3m$$

$$b := \sqrt{(x_2 - x_3)^2 + (y_2 - y_3)^2} = 6.403m$$

$$c := \sqrt{(x_1 - x_3)^2 + (y_1 - y_3)^2} = 10.44m$$

$$p := \frac{a + b + c}{2} = 12.723m$$

$$S := \sqrt{p \cdot (p - a) \cdot (p - b) \cdot (p - c)} = 27.5m^2$$

$$r := \frac{S}{p} = 2.161m$$

Рис. 2. Расчет в среде Mathcad
радиуса окружности,
вписанной в треугольник

шин с привлечением теоремы еще одного древнего грека – Пифагора (см. рис. 3).

Рисунок 2 – это протокол расчета радиуса вписанной в треугольник окружности, выполненный в среде Mathcad. Координаты вершин треугольника на рис. 2 даны в метрах. Это покорило многих «чистых» математиков. Они сразу скажут, причем тут единицы длины²? Почему здесь именно метры, а не сантиметры, километры или футы с дюймами?! А вот почему! Ввод в расчет единиц измерения позволяет вести контроль размерностей и выявлять возможные ошибки при вводе формул. Забыл человек, например, возвести переменную в квадрат или по ошибке возвел ее в куб. Расчет без метров эту оплошность «проглотит». А если ввести размерные величины, то такая ошибка компьютером сразу будет замечена и исправлена [1].

При решении задачи о треугольнике, вписанном в окружность,² в старых версиях Mathcad была возможность оперировать не конкретными единицами измерения (метрами, секундами, килограммами и т.д.), а физическими величинами (длинной, временем, массой и т.д.), что более логично для нашей задачи. Но это уже будет «коробить» инженеров, любящих конкретику в расчетах. Так что всем не угодишь. Кого-то, например, может даже «коробить» такое «чисто математическое» выражение $x + x_2$: нельзя складывать переменную и ее квадрат!

можно не искать готовые решения, а поступить так, как было описано в [2], – проанализировать задачу, вспомнить, например, что радиус, проведенный в точку касания, перпендикулярен стороне треугольника (см. ниже). В результате такого анализа нетрудно составить систему трех алгебраических уравнений, описывающих положение окружности в треугольнике, и решить ее на компьютере аналитически (символьно – см. рис. 3) или численно (приблизительно – см. рис. 4). Раньше по понятным причинам нельзя было и помыслить о такой «лобовой атаке» на задачу – не было доступных компьютерных средств решения систем нелинейных алгебраических уравнений³. Теперь же – другое дело! Почти у каждого студента или школьника под рукой есть компьютер, в который несложно загрузить соответствующие программные средства решения такого рода задач!

Три алгебраических уравнения получаются, если исходный треугольник разбить на шесть прямоугольных треугольников (вернемся к рис. 1). В каждом из них одна из вершин упирается в центр окружности, а один из катетов является радиусом окружности. Гипотенузами этих прямоугольных треугольников будут отрезки прямых, соединяющих искомый центр окружности с вершинами треугольника. Наши уравнения, показанные на рис. 2 и 3, фиксируют тот факт, что длины сторон исходного треугольника – это суммы длин пар катетов смежных прямоугольных треугольников.

На рис. 3 показано использование в среде Mathcad аналитического (символьного) оператора solve при поиске формул для переменных r , x_2 и y_2 , подстановка которых в исходные уравнения превращает их строго в тождества. Mathcad решил поставленную задачу, но не смог выдать «на печать» очень объемный ответ – матрицу с двумя столбцами (два решения системы) и тремя строками (три формулы

³ Эти уравнения не алгебраические (состоящие из полиномов), а иррациональные [3]. Термин «алгебраический» тут означает, что это не дифференциальное уравнение.

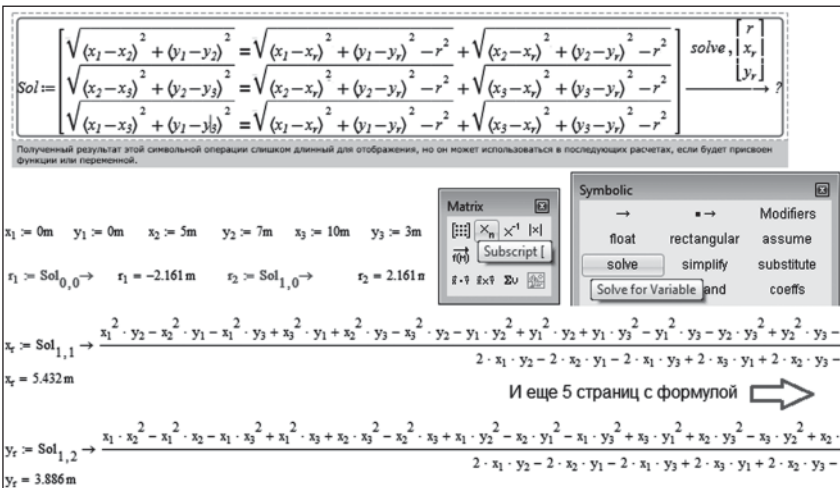


Рис. 3. Аналитическое решение задачи об окружности, вписанной в треугольник

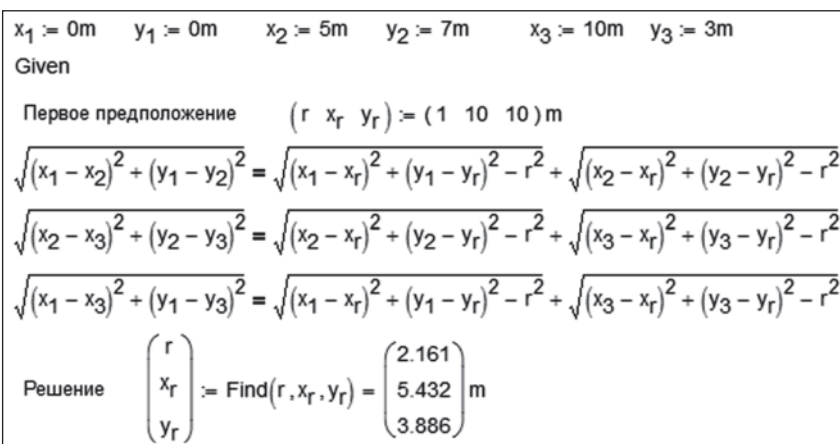


Рис. 4. Численное решение задачи об окружности, вписанной в треугольник

для трех неизвестных системы). Сообщение об этой ошибке «вывалилось» из «покрасневшего» оператора solve. Руководствуясь этим сообщением, мы вывели «на печать» отдельные элементы матрицы-ответа Sol, да и то только малые части формул для x_2 и y_2 . Формулу для r пакет Mathcad «на печать» выдать так и не смог. Эти формулы можно попытаться упростить, но они все равно останутся слишком объемными. Поэтому-то их и нет в справочниках или интернете. Но в принципе, несмотря на их очень большой размер, в настоящее время их можно поместить в интернет-справочники в виде, например, очень длинных текстов, но не для изучения в образовательных целях, а для копирования и вставки в компьютерные программы, где необходимо рассчитать положение вписанной в треугольник окружности по координатам вершин треугольника.

По трем формулам (двум видимым и одной невидимой – см. рис. 3) были подсчитаны значения искомых величин r , x_r и y_r при заданных ранее значениях координат вершин треугольника.

Если же нужно решить нашу задачу уже не на плоскости, а в объеме (найти параметры сферы, вписанной в тетраэдр, например), то аналитические подходы тут уже не сработают. В этом случае придется прибегать к численным методам решения задач. Проиллюстрируем их на уже нами решенной «плоской» задаче об окружности и треугольнике (см. рис. 4).

На рис. 4 показано численное решение нашей системы трех нелинейных уравнений. Для этого записываются исходные данные (координаты вершин треугольника), вводится ключевое слово Given (Дано), задаются значения первого предположения решения и записывается сама система уравнений.

Встроенная в Mathcad функция Find (Найти) начинает по особому численному алгоритму менять значения своих аргументов (неизвестных r , x_r и y_r) так, чтобы прийти в конце концов к решению – найти значения этих переменных, подстановка которых в исходные уравнения превращает их в тождества. Вернее, почти (примерно, приближенно) в тождества: правые и левые части уравнений должны по модулю отличаться друг от друга на наперед заданное число STOL. По умолчанию $STOL = 10^{-3}$, а единицы измерения при этом равны базовыми единицам СИ – метрам в нашем случае. То есть наша геометрическая задача решена с точностью до миллиметра (10^{-3} m). Если бы мы наш треугольник и окружность измеряли не метрами, а сантиметрами (что более естественно – вспомним, как мы вычерчивали геометрические фигуры в линованной тетрадке по математике), то ответ был бы слишком грубым. Тут нужно было бы менять значение системной переменной STOL на значение 10^{-7} , например.

А по какому алгоритму функция Find «толкает» значения неизвестных от начального предположения, заданного пользователем, к искомому решению?! Можно узнать название этого алгоритма (метода) через нажатие правой кнопки мыши на слове Find и поискать информацию по конкретному методу в том же интернете (рис. 5). А можно поступить несколько иначе – не так скучно.

Пакет Mathcad 15, в среде которого численно решалась задача (см. рис. 4), имеет очень полезную возможность записи промежуточных значений искомых величин при реализации тех или иных встроенных инструментов численного решения задач. В нашем случае можно вспомнить крылатую фразу и сказать: «Цель ничто – движение всё!» Нас особо не интересует цель — место «вписывания» окружности в треугольник (мы его уже нашли аналитически – см. рис. 3 и численно — см. рис. 4). Но нас интересует, как окружность туда «закатывается численно», т.е. ее движение.

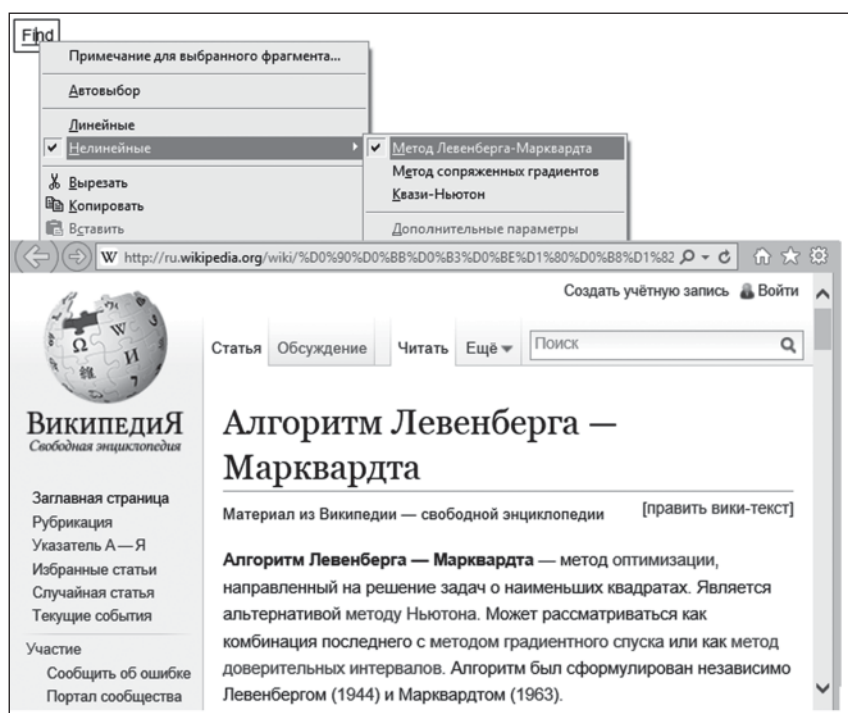


Рис. 5. Выбор метода решения системы уравнений в среде Mathcad 15 и справка в интернете

На рис. 6 показана траектория движения центра окружности от точки начального предположения ($x = 10 \text{ m}$, $y = 10 \text{ m}$) до «цели» – до решения ($x = 5.432 \text{ m}$, $y = 3.886 \text{ m}$). Мы специально задали пакету Mathcad довольно трудное задание – поместили точку начального предположения вне треугольника, чтобы посмотреть, как пакет «будет идти к цели»: «Много ты компьютер о себе воображаешь – посмотрим, проглотишь ли ты вот это!» А нужно было бы, конечно, задать значения первого предположения вблизи решения – где-то в центре исходного треугольника. Но, повторяем, «Цель ничто – движение всё!».

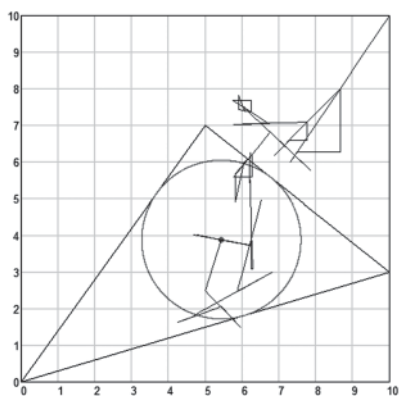


Рис. 6. Траектория движения центра окружности в треугольник

Но интересно посмотреть это движение окружности в **анимации**, инструменты создания которой есть в Mathcad 15 [4]. На рис. 7 показан начальный кадр этой анимации ($n = 0$): окружность (видна только ее дуга-четверть) с радиусом 1 m находится в точке $x = 10 \text{ m}$, $y = 10 \text{ m}$. Затем (первые 94 кадра анимации) окружность срывается с места и ударяется о ближайшую сторону треугольника (рис. 8), подскокивает и ударяется о треугольник снова и снова, как бы пытается пробить в нем дыру.

Но такой «лобовой таран» не удается, и окружность начинает кататься по стороне треугольника,

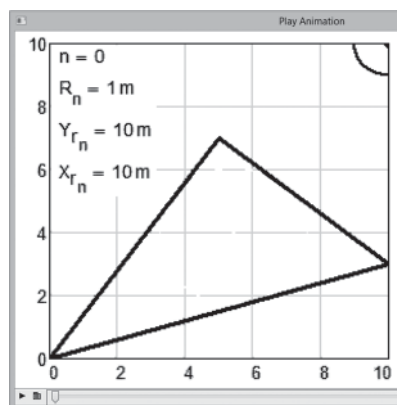


Рис. 7. Начальный кадр анимации движения окружности в треугольник

пытаясь как-то проникнуть в него (рис. 8). Эти попытки оказываются также безуспешными. Тогда наша «живая и настырная» окружность начинает уменьшаться в размерах почти до точки (рис. 9), находит некую «дырочку» в стороне треугольника и проникает внутрь его (рис. 10).

Затем окружность начинает «биться» внутри треугольника (рис. 11), меняя свое положение и радиус, пока не «успокоится» – не найдет в нем свое место: решение задачи (рис. 6 и 12).

Такое «живое» поведение окружности при «численном» ее вписывании в треугольник, отображенном на рис. 7–12, видно только при хорошем пространственном воображении. Но можно рисунки с кадрами анимации вырезать из журнала, сшить их в блокнот, быстро перелистывать его, перегнув и зажав пальцами страницы и отпуская их. Такие книжки-игрушки с незамысловатыми сюжетами были очень популярны до изобретения кино и телевидения. Так примерно и поступает пакет Mathcad 15 при создании анимации, «сшивая в блокнот» отдельные кадры, а затем показывая их со скоростью, достаточной для анимирования. Анимация движения окружности в треугольник (рис. 7–12) опубликована на сайте PlanetPТС под номером 3972⁴.

⁴ Полный адрес этой анимации <http://communities.ptc.com/videos/3972>. PlanetPТС это форум, где пользователи Mathcad и других программ фирмы PTC задают вопросы и обмениваются информацией.

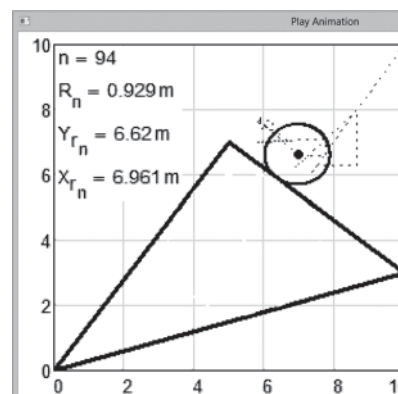


Рис. 8. 94-й кадр анимации движения окружности в треугольник

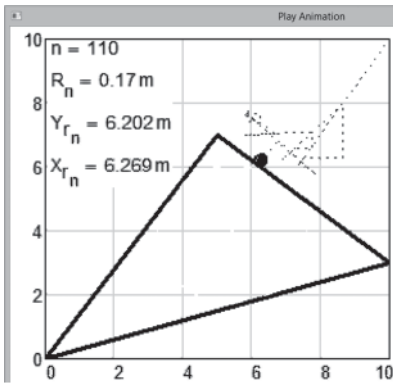


Рис. 9. 110-й кадр анимации движения окружности в треугольник

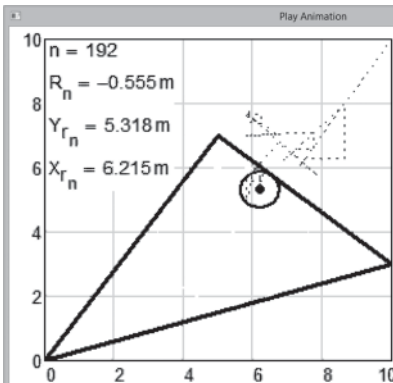


Рис. 10. 192-й кадр анимации движения окружности в треугольник

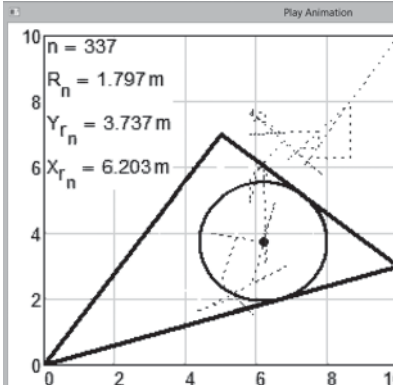


Рис. 11. 337-й кадр анимации движения окружности в треугольник

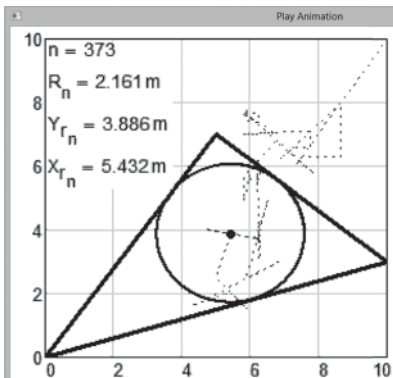


Рис. 12. Финальный кадр анимации движения окружности в треугольник

Как можно выводить промежуточные данные при запуске встроенных в Mathcad 15 инструментов численного решения задач?

Для этого в среде Mathcad есть три инструмента: встроенная функция trace, режим работы Toggle Debugging (Режим отладки) и окно Trace Window, куда заносятся промежуточные данные. Все они показаны на рис. 13.

Функция trace должна быть вставлена в одно из алгебраических уравнений (см. верхний левый угол на рис. 13). Аргументами функции trace будут поочередно: радиус окружности (как на рис. 13) и координаты центра, начальные, промежуточные и конечные значения которых будут автоматически заноситься в окно Trace Window (левый нижний угол рис. 13), если включен режим отладки (правая часть рис. 13). Эти массивы данных нужно будет скопировать, перенести в расчет, отобразить на графике и создать по ним анимацию движения окружности от точки первого предположения до конечной точки. При этом можно менять начальные размер и положение окружности, а также метод численного решения задачи и точность поиска корня алгебраического уравнения. Так можно «снимать довольно увлекательное кино», т.е. заниматься своего рода компьютерным изобразительным искусством [5], базирующимся на прикладной математике.

На сайте PlanetPТС можно увидеть анимацию (номер 5015) численного решения «обратной» задачи – задачи об окружности, описанной вокруг треугольника. Окружность срывается с места в центр треугольника почти к решению, а

затем «в сомнениях» (опять живой организм!) начинает рыскать в нем, отыскивая окончательное решение.

Ну, а если особо не увлекаться «съемками кино с окружностью и треугольником в главных ролях», а решать задачу о вписывании окружности в треугольник традиционными методами, то нам понадобятся три теоремы («Нет ничего практичнее хорошей теории!»):

Теорема 1. Все три биссектрисы треугольника пересекаются в одной точке.

Теорема 2. Центр вписанной в треугольник окружности лежит в точке пересечения биссектрис.

Теорема 3. Биссектриса угла треугольника делит противоположную сторону на отрезки, пропорциональные прилежащим сторонам.

Кстати, в предыдущих расчетах мы опирались еще на одну важнейшую теорему: радиус окружности, проведенный в точку касания, перпендикулярен касательной. Именно это и дало нам возможность нарисовать шесть прямоугольных треугольников и составить для них нелинейную систему алгебраических уравнений.

Итак, опираясь на три вышеперечисленные теоремы, наметим план решения поставленной задачи. Нам достаточно найти точку пересечения лишь двух каких-либо биссектрис (теорема 1). Для этого нам понадобятся уравнения этих биссектрис. Уравнение прямой (биссектрисы) получим, зная две точки, лежащие на этой прямой.

Мы исходим из того, что длины сторон исходного треугольника уже найдены. Обозначим a_{12} длину стороны A_1A_2 , a_{13} длину стороны A_1A_3 и a_{23} длину стороны A_2A_3

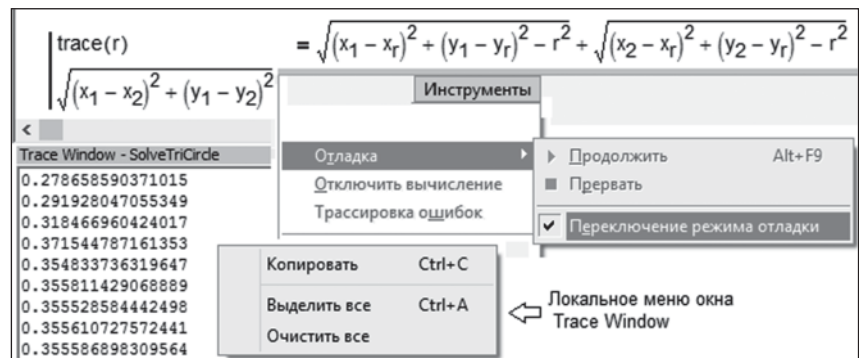


Рис. 13. Инструменты вывода промежуточных результатов численного поиска решения

(см. рис. 1). Пусть A_3B_3 – биссектриса угла A_3 , а точка B_3 лежит на стороне A_1A_2 . Аналогично A_2B_2 – биссектриса угла A_2 , а точка B_2 лежит на стороне A_1A_3 . Тогда (теорема 3) имеем соотношения

$$\frac{A_2B_3}{A_1B_3} = \frac{a_{23}}{a_{13}} \text{ и } \frac{A_3B_2}{A_1B_2} = \frac{a_{23}}{a_{12}}.$$

Если известны две точки плоскости $A(x_A, y_A)$, $B(x_B, y_B)$, то координаты точки $M(x_M, y_M)$, которая делит отрезок AB в отношении

$$\lambda = \frac{AM}{BM},$$

ищутся по формулам

$$x_M = \frac{x_A + \lambda \cdot x_B}{1 + \lambda}, y_M = \frac{y_A + \lambda \cdot y_B}{1 + \lambda}.$$

Используя указанные формулы, легко вычисляем координаты точек $B_3(x_{B3}, y_{B3})$ и $B_2(x_{B2}, y_{B2})$.

Уравнение прямой в общем случае имеет вид $Kx + Ly + P = 0$. Но поскольку в начале координат находится вершина A_1 , то искомые биссектрисы A_2B_2 и A_3B_3 через начало координат не проходят, а поэтому уравнение каждой биссектрисы будем искать в упрощенном виде $kx + ly + 1 = 0$. Искомые коэффициенты найдем из системы, полученной при подстановке в уравнение прямой координат двух точек, лежащих на этой прямой. Так, уравнение для биссектрисы A_2B_2 будет иметь вид $k_2x + l_2y + 1 = 0$, где

$$\begin{cases} k_2x_2 + l_2y_2 + 1 = 0 \\ k_2x_{B2} + l_2y_{B2} + 1 = 0 \end{cases}$$

а для биссектрисы A_3B_3 будет иметь вид $k_3x + l_3y + 1 = 0$, где

$$\begin{cases} k_3x_3 + l_3y_3 + 1 = 0 \\ k_3x_{B3} + l_3y_{B3} + 1 = 0 \end{cases}$$

Останется найти точку (x, y) пересечения этих биссектрис, поскольку (теорема 2) она и является центром окружности. Координаты (x, y) таким образом находятся из системы шести линейных уравнений

$$\begin{cases} k_2x_2 + l_2y_2 + 1 = 0 \\ k_2x_{B2} + l_2y_{B2} + 1 = 0 \\ k_3x_3 + l_3y_3 + 1 = 0 \\ k_3x_{B3} + l_3y_{B3} + 1 = 0 \\ k_2x + l_2y + 1 = 0 \\ k_3x + l_3y + 1 = 0 \end{cases}$$

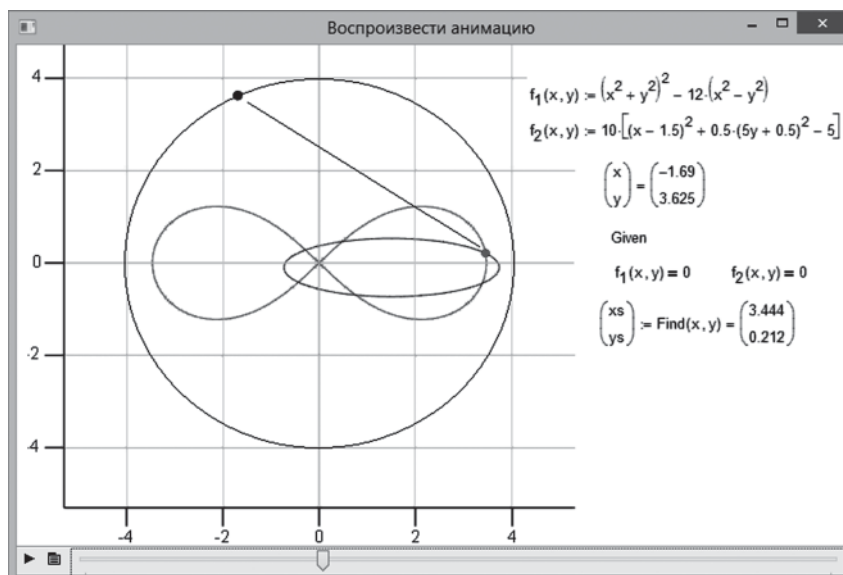


Рис. 14. Кадр анимации связи первого предположения с корнем системы уравнений

в которой числа x_2, x_3, y_2, y_3 известны по условию, а числа $x_{B2}, x_{B3}, y_{B2}, y_{B3}$ уже найдены. Можно сразу решить указанную систему. А можно решить последовательно три системы (из двух уравнений), на которые и распадается данная система. Заметим, что Mathcad справится с такой задачей очень легко. Но это потому, что всю тяжесть решения мы взвалили на свои плечи.

Ух! Вернемся лучше к «съемкам фильмов» с нелинейными уравнениями.

Система алгебраических уравнений, показанная на рис. 3 и 4, как мы уже отметили, имеет два решения: одно с положительным значением r и второе с таким же по модулю, но отрицательным значением r . На рис. 14 показан кадр еще одного «кинофильма» — решения системы двух алгебраических уравнений: уравнения эллипса и уравнения лемнискаты Бернулли, которая имеет уже не два, а шесть действительных корней. Вокруг этих двух замкнутых кривых и точек их пересечения (корни системы двух уравнений) в анимации вращается по окружности точка первого предположения. Другая точка (текущее решение системы уравнений – точка пересечения эллипса с лемнискатой) перескакивает с одного места на другое. Эта авторская анимация на сайте PlanetPTC имеет номер 2418. Она показывает,

как точка первого предположения влияет на решение системы – на выбор того или иного корня. Эта анимация акцентирует внимание на важности выбора начальной точки поиска решения.

Иногда вместо термина «**первое предположение**», очень важного при использовании численных методов решения задач, используют термин «**начальное приближение**». Это не совсем верно, что и доказывает анимация, показанная на рис. 14. Очень часто решение – это не точка, ближайшая к исходной, а совсем другая. А в ряде случаев – даже одна из самых дальних. Все зависит от системы уравнений и метода, примененного для ее численного решения. В статье [5], которая также описывает «пересечение» прикладной математики (а конкретнее, численных методов решения задач) с изобразительным искусством, области соответствия стартовой точки тому или иному корню системы двух нелинейных алгебраических уравнений были закрашены в разные цвета. Получилась картина, очень напоминающая одно из полотен Ван Гога.

Наблюдая в анимации движение окружности (рис. 1–13), мы видим, что она ведет себя как некий **живой организм**, забивающийся в свою «треугольную норку». Тут как раз впору вспомнить, что некоторые численные методы решения задач

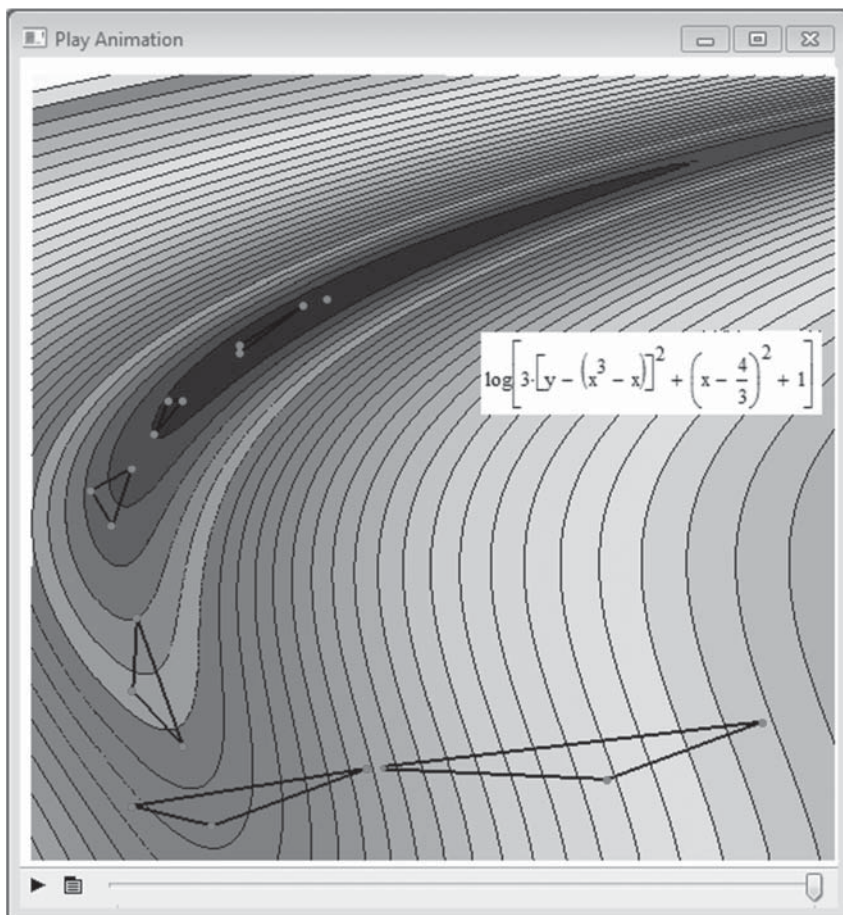


Рис. 15. Сползание треугольника в ложбину

ции, но эффективен: точка минимума находится быстро (за малое число ходов) и довольно точно. Если у анализируемой функции не две, а три переменные, то треугольник заменяется тетраэдром. Если переменных больше трех, то для реализации метода выбирается соответствующий многогранник.

Хитроумные сложные алгоритмы, подобные тому, какой «наследил» на рис. 15, создавались во времена тихоходных компьютеров для ускорения счета: одно дело найти минимум функции за час, а другое – за несколько минут. В настоящее время быстродействие компьютеров существенно повысилось и стало возможным использовать и простые алгоритмы: нет большой разницы в том, считаем ли мы одну тысячную доли секунды или одну сотую доли секунды.

На рис. 16 показан след менее быстрого, но довольно простого в описании и реализации авторского

имеют «животные» названия. Так, один из методов поиска минимума функции называется «Амеба». Понять это название можно, решая математическую задачу, и видя, как уже не окружность, а треугольник может вести себя, словно **живое существо**, и переползает куда-то.

Кадры анимации «переползания» треугольника показаны на рис. 15 (анимация под номером 4984): футуристическая треугольная амеба, переваливаясь с боку на бок, ползет в ложбину (минимум функции двух аргументов), уменьшаясь при этом в размерах (сжимаясь), пока не найдет искомую точку, обреченную на рис. 15 линиями одного уровня.

Можно найти в интернете описание этого метода оптимизации (другие его названия – метод Нелдера – Мида, метод деформируемого многогранника, симплекс-метод) и понять его суть. Метод довольно сложен и в описании, и в реализа-

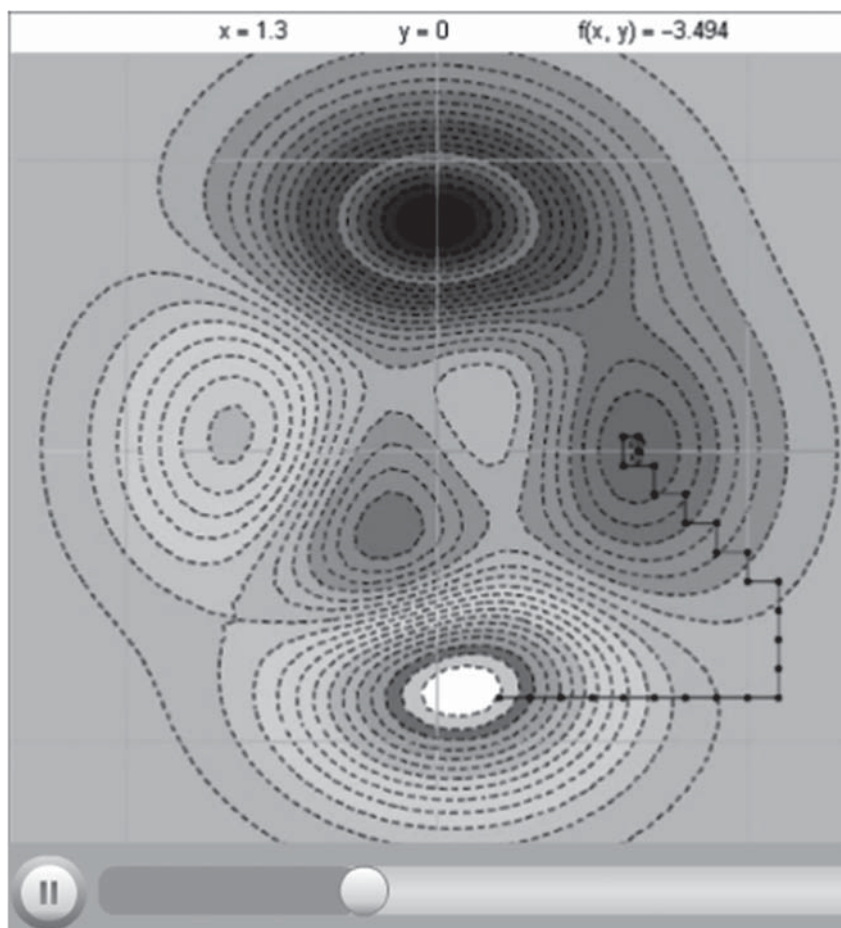


Рис. 16. Анимация метода «Два шага»

```

MinTwoStep(f, x, D) := ( L ← last(x) f_min ← f(x) j ← p ← n ← 0 M(0) ← x )
while D > CTOL
  p ← 1
  while p
    p ← 0
    for i ∈ ORIGIN..L
      for X ∈ -D, D
        ( x_i ← x_i + X F ← f(x) )
        ( p ← X j ← i f_min ← F ) if F < f_min
        x_i ← x_i - X
      ( x_j ← x_j + p n ← n + 1 M(n) ← x )
    D ← D / 2
  M

```

Рис. 17. Mathcad-программа «Два шага»

метода оптимизации, который условно можно назвать «Два шага» (номер анимации 1460).

Программа, еще раз подчеркиваем, довольно проста: больше времени ушло на поиск названия для нее, чем на ее создание. Можно было бы, конечно, тут применить стандартный метод градиентного спуска, при котором шаги делаются по направлению градиента (при поиске максимума) или минус градиента (при поиске минимума), но была поставлена цель написать самую простую в реализации программу, особо не думая о ее скорости. Программа была названа «**Два шага – Two step**». В начале XX в. был очень популярен такой бальный танец. Один персонаж кинофильма «Свадьба в Малиновке» называл его «В ту степь». Программа при поиске минимума делает два шага от очередной точки приближения (от «печки», если опять же вспомнить танцы) и переносит опорную точку туда, где значение оптимизируемой функции минимально. Эти два шага длиной X делаются по всем координатам (их число L) в отрицательном (-D) и положительном (D) направлениях. По мере приближения к минимуму шаги укорачиваются. Такие танцевальные па повторяются до тех пор, пока шаг не станет короче заданного значения погрешности $D < CTOL$ – рис. 17).

Программа «выводит» на печать не только финальную точку, а весь путь к этой точке (матрица M), который несложно отобразить на графике и анимировать (см. рис. 16).

В программе интересен не сам алгоритм, а след, оставляемый программой на плоскости X-Y («в степи») при оптимизации некоторой функции двух переменных (рис. 18 и 19).

Форма следа программы «Тустеп» позволила дать ей и другие названия. Считается, что чем больше у человека имен, тем больше у него небесных покровителей. Это, наверное, можно отнести и к программам (методам) решения задач.

Если наполнить ванну водой, бросить в нее, к примеру, перышко и выдернуть пробку, то перышко

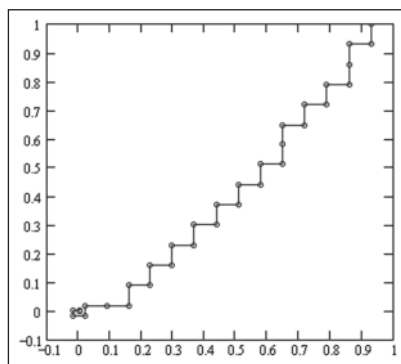


Рис. 18. Начальные шаги приближения к минимуму

сначала будет более-менее спокойно двигаться к сливному отверстию, а потом закрутится в водовороте. Примерно так, как показано на рис. 18 и 19: при начальных шагах приближения (рис. 18) промежуточные точки спокойно дрейфуют к минимуму, а затем «закручиваются в вихре танца» (рис. 19), который позволяет называть данный метод оптимизации не только «Тустеп», но и «Вальс». Но если быть совсем точным, то его следует именовать также и методом *наискорейшего спуска воды*. И вот почему.

Вода в ванне или в каком-либо ином сосуде с дыркой на дне всегда закручивается в одну сторону. Даже если воду раскрутить рукой в другом направлении, то, преодолев насилие, она вернет свою природную закрутку. Это связано с силой Кориолиса, которая, в свою очередь, зависит от вращения Земли вокруг своей оси. В северном полушарии вода закручивается по часовой стрелке, в южном – против, на экваторе – сливается без вихрей (а на полюсах – совсем не течет: замерзает – шутка!). Это явление основательно изучалось в alma mater авторов статьи (в Московском энергетическом институте) применительно к барабанным паровым котлам. Дело в том, что вода из барабана котла, поступая в опускные трубы, а затем в подъемные трубы, расположенные в топке котла, может закручиваться и захватывать вследствие этого большие порции водяного пара. Это ухудшает теплообмен и приводит к авариям – разрывам паровых труб от перегрева. Подробнее об этом здесь: <http://twt.mpei.ru/ochkov/KotelVikhr.html>.

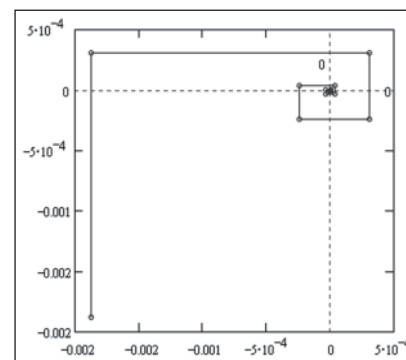


Рис. 19. Конечные шаги приближения к минимуму

Для проверки этой гипотезы авторы переслали по интернету файл с программой оптимизации коллеге в Австралию (чем черт не шутит) и к полному своему удивлению, переходящему в удовлетворение, узнали, что след поиска минимума в южном полушарии закручивается против часовой стрелки. «Австралийский» след оптимизации из-за экономии места здесь не приводится. Но его можно увидеть, посмотрев на рис. 18 и 19 через зеркало.

Причина иной закрутки следа поиска минимума южнее экватора, конечно, не в воде, хотя сила Кориолиса здесь присутствует: при пересылке файла в Австралию в код программы (рис. 17) вкралось маленькое изменение. Вместо строки `for X ∈ -D, D` появилась другая – `for X ∈ D, -D`. На поиск минимума это не повлияло (перебор координат оптимизируемой функции ведется всего лишь с другого конца), но тем не менее след стал закручиваться по-иному. А виной тому все та же сила Кориолиса, заставляющая жесткие диски серверов и маршрутизаторов в южном полушарии Земли вращаться несколько иначе, чем в северном полушарии. Отсюда и незначительные сбои при

пересечении файлами экватора. Их выловить довольно трудно, так как при контрольной обратной пересылке файла ошибка исправляется по принципу минус на минус дает плюс. (Продолжение темы закрутки следа поиска минимума – <http://twt.mpei.ac.ru/ochkov/SI>.)

Выводы

Современные программные средства позволяют легко и быстро анимировать численные методы решения математических задач. Такая анимация позволяет лучше понимать эти методы, особенности и ограничения при их применении для решения прикладных задач на компьютере. Кроме того, такая анимация позволит оживить, сделать более интересными занятия по изучению данных методов на семинарах и уроках информатики и прикладной математики.

Авторы не устают повторять своим студентам, что от учебы нужно стараться в первую очередь получать не знания и навыки, а... удовольствие. Учеба – это работа, очень трудная работа, а без удовольствия, без любви к труду даже необременительная, престижная и

высокооплачиваемая работа может превратиться в каторгу. Удовольствие же немислимо без шутки (см. последнее ключевое слово статьи), без легкого и ироничного отношения к самым серьезным вещам. А что может быть «серьезнее» математики? Наша средняя школа со своей зубрежкой ответов для ЕГЭ напрочь отбивает у многих школьников тягу к получению знаний, к «получению удовольствия от процесса получения знаний». Эту статью можно рассматривать, как попытку вернуть учащимся интерес к этой самой главной человеческой способности – к получению знаний. Многие студенты тянутся к компьютеру, но будущая их инженерная специальность – это не всегда компьютерные технологии (IT), а, например, теплотехника [8]. Авторы надеются, что данная статья поможет этим студентам хотя бы через компьютер и интернет возродить в себе способность к получению удовольствия от учебы в техническом вузе, основой которой является математика. Недаром Иммануил Кант говаривал, что «в каждой естественной науке заключено столько истины, сколько в ней есть математики».

Литература

1. *Очков В.Ф.* Физические и экономические величины в Mathcad и Maple. – М.: Финансы и статистика, 2002. – 192 с. – (Серия «Диалог с компьютером»). – Доступно на: http://twt.mpei.ac.ru/ochkov/Units/Forword_book.htm
2. *Очков В.Ф.* Преподавание математики и математические пакеты // Открытое образование. – 2013. – № 2. – С. 23–34. – Доступно на: <http://twt.mpei.ac.ru/ochkov/Mathcad-15/OchkovMath.pdf>
3. *Богомолова Е.П., Очков В.Ф.* Решатели, или Великолепная семерка Mathcad // Открытое образование. – 2015. – № 1. – С. 27–33. – Доступно на: <http://twt.mpei.ac.ru/ochkov/Solvers-OE.pdf>
4. *Очков В.Ф.* Живые кинематические схемы в Mathcad // Открытое образование. – 2013. – № 3. – С. 27–33. – Доступно на: <http://twt.mpei.ac.ru/ochkov/Mathcad-15/kinematic.html>
5. *Очков В.Ф.* Решение алгебраических уравнений и систем, или Ван Гог в среде Mathcad // КомпьютерПресс. – 2001. – № 9. – Доступно на: <http://twt.mpei.ac.ru/ochkov/Carpet/index.htm>
6. *Очков В.Ф.* Программирование и сила Кориолиса, или Необычайное приключение файла в Интернете // Компьютерра. – 1997. – № 13. – Доступно на: <http://twt.mpei.ac.ru/ochkov/Koriolis.htm>
7. *Яворский Б.М., Детлаф А.А.* Справочник по физике для инженеров и студентов вузов. – М.: Наука, 1977.
8. Теплотехнические этюды с Excel, Mathcad и Интернет / под общ. ред. В.Ф. Очкова. – Изд. 2-е, испр. и доп. – СПб.: БХВ-Петербург, 2015. – 336 с. – Доступно на: <http://twt.mpei.ac.ru/ochkov/TTMI>