

Автор д.т.н., профессор НИУ "МЭИ" Солодов А.П.

9 Численные методы теплопроводности

9.1 Введение

В последнее время все более актуальными становятся задачи, связанные с воздействием сверхмощных потоков энергии на элементы конструкций. Например, в проблеме управляемого термоядерного синтеза плотность теплового потока на тепловоспринимающих твердых поверхностях может достигать 10^8 Вт/м². В тяжелых температурных условиях работают графитовые электроды плазмотронных установок, применяемых для высокотемпературной обработки различных материалов. Большие потоки и высокие температуры возникают при лазерной или электронно-лучевой обработке деталей с целью упрочнения их поверхности. Похожие процессы имеют место при изготовлении микросхем.

Почти всегда мощные воздействия имеют импульсный, периодический характер, и в твердых телах возникают и распространяются температурные волны.

Математические модели таких процессов часто оказываются нелинейными, и по этой (и другим причинам, например, из-за геометрической сложности) недоступными для аналитических методов предыдущей главы. Универсальным способом решения являются численные методы, требующие компьютерной реализации.

Рассмотрим далее в качестве модели описанных выше процессов одномерную нестационарную задачу теплопроводности с внутренними источниками теплоты (Рис. 9.1).

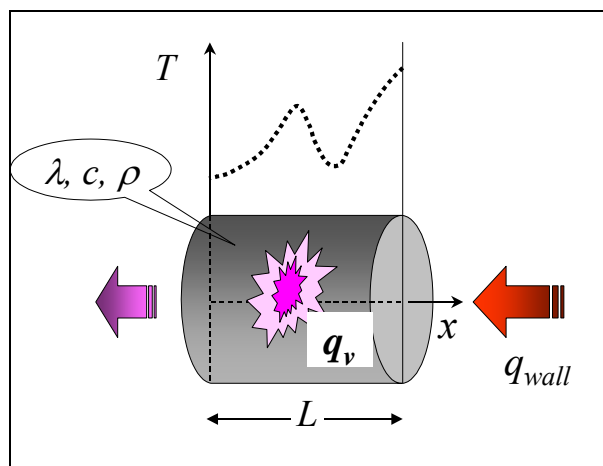


Рис. 9.1. Одномерная нестационарная задача теплопроводности

Предположим, что пространственные изменения происходят только вдоль оси координат x . Боковую поверхность будем считать адиабатически изолированной. Если потребуется учесть теплообмен на боковой поверхности, то можно имитировать теплосъем посредством внутреннего стока.

Чтобы обеспечить универсальность модели, воспользуемся численным методом и разработаем для этой цели специальную Mathcad-программу. В ее основе будет лежать знаменитый в численном анализе *метод прогонки* – сверхбыстрый алгоритм для решения больших систем уравнений со специальной диагональной структурой.

9.2 Математическая формулировка задачи

В качестве исходной формулировки можно принять уравнение энергии (9.1), в котором следует исключить оператор конвективного переноса и учесть одномерность задачи, $t = t(x, \tau)$. В результате получится

$$\rho c \frac{\partial t}{\partial \tau} = \frac{\partial}{\partial x} \left(\lambda \frac{\partial t}{\partial x} \right) + q_v. \quad (9.1)$$

На левом и правом торцах объекта (см. Рис. 9.1) происходит тепловое взаимодействие со средой, и здесь необходимо задать граничные условия. Универсальным способом описать самые различные ситуации будет применение *граничных условий третьего рода* на левом ($x=0$) и правом ($x=L$) торцах объекта:

$$\begin{aligned} -\lambda \left. \frac{\partial t}{\partial x} \right|_{x=+0} &= \alpha_1 (t_{f1} - t(x=0, \tau)); \\ -\lambda \left. \frac{\partial t}{\partial x} \right|_{x=L-0} &= \alpha_2 (t(x=L, \tau) - t_{f2}), \end{aligned} \quad (9.2)$$

где α и t_f – коэффициенты теплоотдачи и температуры окружающей среды на торцах стержня.

В этих соотношениях приравнены значения плотности теплового потока,

- поступающего из окружающей среды и вычисленного по уравнению Ньютона–Рихмана (правые части)
- и отводимого внутрь тела посредством теплопроводности и вычисленного по закону Фурье (левые части).

Следует подчеркнуть, что такое равенство справедливо *при отсутствии* фазовых превращений на поверхности раздела. Поскольку мы не собираемся сейчас решать сложные задачи с фазовыми превращениями, ограничимся этим предупреждением. В общем случае разность потоков по обеим сторонам границы раздела расходуется на плавление, отвердевание, испарение и т.п.

9.3 Дискретное представление

Для численного интегрирования дифференциального уравнения в частных производных (9.1) необходимо подготовить его конечно-разностную аппроксимацию. Предварительно полезно просмотреть еще раз аналогичные вычисления, которые проводились ранее в п..

Запишем закон сохранения энергии для малого, но *конечного контрольного объема*, сформированного в окрестности узла P (Рис. 9.2). Последний находится сейчас в фокусе нашего внимания, но выведенные далее соотношения будут годиться для любого *внутреннего* узла.

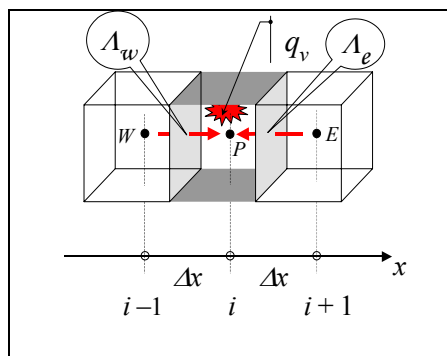


Рис. 9.2. Контрольный объем и потоки энергии

От соседних, западного W и восточного E узлов поступают потоки теплоты за счет теплопроводности. Внутри контрольного объема выделяется теплота при действии внутреннего источника. В результате произойдет увеличение тепловой энергии в рассматриваемом объеме, что будет обнаружено по повышению температуры – от T_{0P} до T_P за время $\Delta\tau$. Нулевой баланс bal для всех составляющих, перенесенных в одну сторону уравнения, запишется следующим образом:

$$bal = \frac{\rho c \Delta x (T_P - T_{0P})}{\Delta\tau} - \lambda \Lambda_w \frac{T_W - T_P}{\Delta x} - \lambda \Lambda_e \frac{T_E - T_P}{\Delta x} - q_V \Delta x. \quad (9.3)$$

Величина λ без индекса означает некоторое характерное значение теплопроводности. Большими греческими «лямбда» Λ обозначены относительные значения коэффициента теплопроводности, вычисленные для контрольных поверхностей, проходящих между узлами. Например, для восточной поверхности:

$$\Lambda_e \equiv \frac{\lambda_e}{\lambda} = \frac{1}{2} \frac{\Lambda_P + \Lambda_E}{\Lambda_P \Lambda_E}.$$

(9.4)

Если коэффициент теплопроводности постоянен, то достаточно указать значение λ , а величинам Λ присписать единичные значения. Если же коэффициент теплопроводности сильно зависит от температуры или даже испытывает разрывы из-за слоистой структуры материала, *среднегармонический* способ осреднения (9.4) обеспечивает вычисление тепловых потоков с хорошей точностью.

Уравнение (9.3) записано для внутренних узлов. Ниже будет получено аналогичное уравнение для граничных узлов (9.8).

Неявная схема

Подчеркнем, что температура T_p в центральном узле и температуры T_w и T_E в окружающих узлах – неизвестные величины из «будущего», в то время как T_{0p} – известное значение, взятое с *предыдущего* шага по времени либо из начальных условий. Поэтому (9.3) представляет собой уравнение с тремя неизвестными. Всего таких уравнений можно записать столько же, сколько имеется неизвестных температур в узлах сетки. Для нахождения температур необходимо решить систему уравнений.

Можно повторить то же другими словами: *не существует* явного выражения для неизвестных, а требуется решать уравнения для их определения. Такие схемы в численном анализе называются *неявными*. Они обладают важным *свойством устойчивости* при счете, хотя и сложны из-за необходимости решать систему уравнений.

Явная схема

Если в уравнении (9.3) разности температур во втором и третьем операторах (теплопроводности) записать для «прошлого» и снабдить обозначением «0», то получится *явная* схема: в каждом уравнении содержится одно единственное неизвестное значение T_p (в первом операторе, описывающем изменение энергии во времени). Программа и вычисления будут очень простыми, но если временной шаг превысит некоторое значение, при счете возникнут прогрессирующие паразитные осцилляции. Ограничения на шаг довольно обременительны, поэтому в настоящее время предпочитают неявные схемы.

Подробное обсуждение явной и неявной схем и демонстрационный вычислительный пример приведены в п..

Коэффициенты неявной схемы

Вернемся к анализу уравнения баланса (9.3), задающего неявную численную схему. Упростим форму его записи, поделив почленно на $(\rho c \Delta x / \Delta \tau)$:

$$bal = (T_p - T_{0p}) - Fo \Lambda_w (T_w - T_p) - Fo \Lambda_e (T_E - T_p) - Fo \frac{q_V \Delta x^2}{\lambda},$$

(9.5)

где Fo – безразмерной шаг по времени, так называемое сеточное число Фурье

$$Fo \equiv \frac{\lambda \Delta \tau}{\rho c \Delta x^2}.$$

Перепишем (9.5) в стандартной форме линейного уравнения для неизвестных T_w, T_p, T_E :

$$CT_W + BT_P + AT_E + D = 0 \quad (9.6)$$

или

$$C_k T_{k-1} + B_k T_k + A_k T_{k+1} + D_k = 0.$$

Первое из этих уравнений представлено в мнемонической форме, второе – в индексной, необходимой для составления программы.

Сравнивая выражения (9.5) и (9.6), получают значения коэффициентов A , B , C , D . Необходимые символьные операции можно выполнить в Mathcad, как показано на Рис. 9.3. В первом блоке вычислений применяется оператор `collect`, собирающий вместе члены с каждой из перечисленных неизвестных. Во втором блоке приводится альтернативный вариант с оператором `coeffs`, выписывающим коэффициенты полинома (в нашем случае – первой степени) для указанной переменной.

$\text{bal} := (T_P - T_{0P}) - F_0 \cdot \Lambda_W \cdot (T_W - T_P) - F_0 \cdot \Lambda_e \cdot (T_E - T_P) - F_0 \cdot \frac{q_V \cdot \Delta x^2}{\lambda}$ $\text{bal collect, } T_W, T_P, T_E \rightarrow$ $-F_0 \cdot \Lambda_W \cdot T_W - F_0 \cdot \Lambda_e \cdot T_E + (F_0 \cdot \Lambda_W + F_0 \cdot \Lambda_e + 1) \cdot T_P - q_V \cdot F_0 \cdot \frac{\Delta x^2}{\lambda} - T_{0P}$ $C := -F_0 \cdot \Lambda_W \quad A := -F_0 \cdot \Lambda_e \quad B := 1 - A - C \quad D := -T_{0P} - \frac{q_V \cdot D x^2}{\lambda} \cdot F_0$
$\text{coeffsTP} := \text{bal coeffs, } T_P \rightarrow \left(\begin{array}{c} -T_{0P} - F_0 \cdot \Lambda_W \cdot T_W - F_0 \cdot \Lambda_e \cdot T_E - F_0 \cdot q_V \cdot \frac{\Delta x^2}{\lambda} \\ 1 + F_0 \cdot \Lambda_W + F_0 \cdot \Lambda_e \end{array} \right)$ $B := \text{coeffsTP}_1 \rightarrow 1 + F_0 \cdot \Lambda_W + F_0 \cdot \Lambda_e$

Рис. 9.3. Коэффициенты неявной схемы

Структура получившейся линейной системы уравнений с коэффициентами A , B , C , D оказывается *трехдиагональной*, как в примере для сетки из пяти узлов:

$$(9.7) \quad \begin{pmatrix} B_1 & A_1 & 0 & 0 & 0 \\ C_2 & B_2 & A_2 & 0 & 0 \\ 0 & C_3 & B_3 & A_3 & 0 \\ 0 & 0 & C_4 & B_4 & A_4 \\ 0 & 0 & 0 & C_5 & B_5 \end{pmatrix} \begin{pmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \\ T_5 \end{pmatrix} = \begin{pmatrix} -D_1 \\ -D_2 \\ -D_3 \\ -D_4 \\ -D_5 \end{pmatrix}$$

При большом числе узлов матрица коэффициентов окажется почти пустой. Например, для 100 узлов всего 3% ячеек будет занято числами, остальные останутся нулевыми. Ясно, что при использовании классического Гауссова исключения в основном будут перемножаться и складываться нули. Существует специальный вариант метода исключения – *метод прогонки* [], учитывающий трехдиагональную структуру матрицы, и мы применим его в разрабатываемой вычислительной программе.

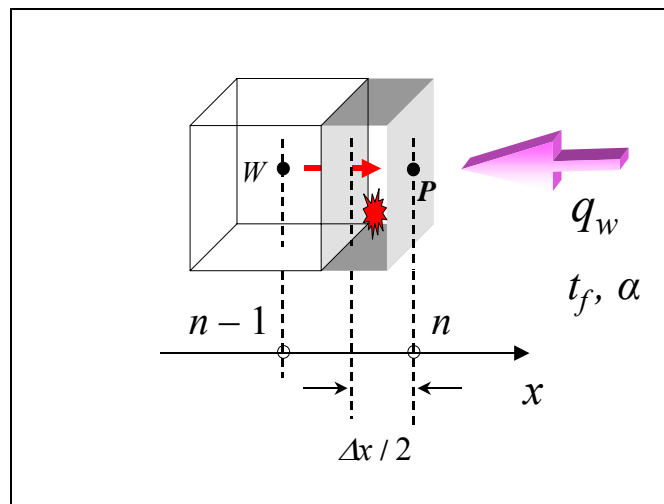


Рис. 9.4. Баланс энергии для поверхностного узла

Для *граничных* узлов контрольные объемы оказываются половинного размера, как показано на Рис. 9.4. Кроме того, для контрольной грани, попадающей на поверхность объекта, необходимо специальным образом записать тепловой поток:

$$bal = \frac{1}{2} \frac{\rho c \Delta x (T_P - T_{0P})}{\Delta \tau} - \alpha (T_f - T_P) - \lambda \Lambda_{in} \frac{T_{in} - T_P}{\Delta x} - \frac{1}{2} q_w \Delta x.$$

(9.8)

Это уравнение является дискретным аналогом граничных условий, заданных выше соотношениями **Ошибка! Источник ссылки не найден..** Индекс «in» означает ближайший внутренний узел: $(n-1)$ правого торца, 2 – левого торца. Мы не будем повторять вычисления для коэффициентов в уравнениях для поверхностных узлов. Они полностью аналогичны тем, что выполнены для внутренних узлов. Окончательные выражения можно прочитать в тексте программы Coef (Рис. 9.5).

9.4 Метод прогонки. Вычислительные программы Coef и SysTRD

Mathcad-программа Coef

Функция Coef (см. Рис. 9.5) вычисляет коэффициенты A, B, C, D для системы уравнений, структура которой иллюстрируется соотношением (9.7).

Величина T0 в списке формальных параметров означает вектор начальных значений температуры. Индексация начинается с единицы, т.е. значение переменной Origin среды Mathcad следует установить в единицу. Размерность вектора T0 равна числу узлов сетки.

Величины Tf и Bi – векторы из двух элементов, задающие соответственно температуры среды и числа Био на левом и правом торцах объекта (Рис. 9.1). Число Био – это безразмерный коэффициент теплоотдачи: $Bi = \alpha \cdot \Delta x / \lambda$.

Цикл for обеспечивает вычисления коэффициентов во внутренних узлах. Отдельно рассчитываются коэффициенты для граничных узлов, исходя из соотношения (9.8).

Параметр iTime сообщает подпрограмме, на каком шаге по времени находится процесс вычислений. Это важно для вычисления температуры среды, контактирующей с правым торцом. Задавая в векторе Pulse ненулевую амплитуду Ampl и некоторое значение частоты ν , мы сможем имитировать периодические тепловые воздействия на объект.

Программа Coef возвращает массив, собранный посредством функции augment из векторов A, B, C, D.


```

Qv := Fo * Δx2 *  $\frac{qv}{\lambda}$ 
Coef(Fo, Λ, Qv, T0, Tf, Bi, Pulse, iTime) :=
  iUp ← last(T0)
  for i ∈ 2.. iUp - 1
    (Ai ← -Fo * Λi  Ci ← -Fo * Λi-1)
    Bi ← 1 - Ai - Ci
    Di ← T0i + Qv
  A1 ← -2 * Fo * Λ1
  B1 ← 1 - A1 + 2 * Fo * Bi1
  C1 ← 0
  D1 ← T01 + Qv + 2 * Fo * Tf1 * Bi1
  (Ampl ← Pulse1  v ← Pulse2)
  Bi2 ← Bi2
  Tf2 ← Tf2 * (1 + Ampl * sin(v * iTime))
  CiUp ← -2 * Fo * ΛiUp
  BiUp ← 1 - CiUp + 2 * Fo * Bi2
  AiUp ← 0
  DiUp ← T0iUp + Qv + 2 * Fo * Tf2 * Bi2
  augment(A, B, C, D)

```

Рис. 9.5. Подпрограмма для вычисления матрицы коэффициентов

Mathcad-программа SysTRD

Функция SysTRD (см. Рис. 9.6) решает трехдиагональную систему линейных уравнений методом прогонки.

SYSTRD(A, B, C, D) :=	$iUp \leftarrow \text{last}(A)$ $\text{for } i \in 2..iUp$ $\left \begin{array}{l} \text{temp} \leftarrow \frac{C_i}{B_{i-1}} \\ (B_i \leftarrow B_i - \text{temp} \cdot A_{i-1} \quad D_i \leftarrow D_i - \text{temp} \cdot D_{i-1}) \end{array} \right.$ $D_{iUp} \leftarrow \frac{D_{iUp}}{B_{iUp}}$ $\text{for } j \in iUp - 1, iUp - 2..1$ $D_j \leftarrow \frac{(D_j - A_j \cdot D_{j+1})}{B_j}$ D
-----------------------	--

Рис. 9.6. Метод прогонки

Матрицы коэффициентов A , B , C , D подготовлены подпрограммой **Coef**. В теле процедуры значения коэффициента D замещаются вычисленными значениями температуры в узлах сетки. Процедура **SySTRD** возвращает вектор решений системы.

Mathcad-программа TimeHistory

Программа **TimeHistory** организует вычисления по временным шагам (см. Рис. 9.7) и возвращает распределения температуры по координате x для последовательных моментов времени $iTime$ как столбцы матрицы F . Параметр $nTime$ задает число шагов по времени.

$\text{TimeHistory}(Fo, \Lambda, Q_V, T_0, Tf, Bi, Pulse, nTime) :=$ $\text{for } iTime \in 1..nTime$ $\left \begin{array}{l} S \leftarrow \text{Coef}(Fo, \Lambda, Q_V, T_0, Tf, Bi, Pulse, iTime) \\ T \leftarrow \text{SYSTRD}(S^{(1)}, S^{(2)}, S^{(3)}, S^{(4)}) \\ T_0 \leftarrow T \\ F^{(iTime)} \leftarrow T \end{array} \right.$

Рис. 9.7. Главная программа

На каждом шаге по времени вызывается процедура **Coef** для вычисления коэффициентов A , B , C , D . Затем происходит обращение к решателю системы уравнений методом прогонки **SySTRD**. Обновляется вектор значений температуры T_0 на предыдущем временном шаге и добавляется

столбец к матрице F , хранящей распределения температуры для последовательных моментов времени.

9.5 Компьютерное моделирование периодических тепловых воздействий

Подготовка исходных данных для расчета показана на Рис. 9.8. Предполагается исследовать температурные колебания в латунном стержне длиной 39 мм, если на правом торце задана температура, пульсирующая около среднего значения 800°C с амплитудой 320°C , а на левом торце – постоянная нулевая температура.

Из вербальной формулировки следует, что на обоих торцах ставятся граничные условия Дирихле (граничные условия первого рода). В программном коде реализованы смешанные условия (условия третьего рода), но их легко приспособить для любой ситуации. Присваивая нужные значения температурам жидкости T_f ($T_{f1}=0$, $T_{f2}=800$ и $\text{Ampl}=320$) и задавая большие значения сеточных чисел Био Bi_1 , Bi_2 , соответствующие большим значениям коэффициентов теплоотдачи, мы обеспечим примерное равенство температур поверхности и жидкости.

Подготовительные вычисления на Рис. 9.8 понятны без комментариев. В конце представленного фрагмента с помощью функции `matrix` формируется вектор начальных температур T_0 , задающий равномерное начальное распределение 100°C , а также вектор значений относительного коэффициента теплопроводности Λ .

В данном примере величина Λ принимается постоянной. Если теплопроводность зависит от координат и/или температуры, то понадобится дополнительная процедура или вставка в программу `Coef`. Лучшим способом осреднения коэффициента теплопроводности будет вычисление среднегармонических значений по формуле (9.4). Мы оставляем эти усовершенствования программы как упражнение для читателей.

$\lambda := 100$	$c_p := 400$	$\rho := 9000$	$a := \frac{\lambda}{c_p \cdot \rho}$
$\Delta x := 0.001$	$nX := 41$	$\Delta \tau := 1$	$nTime := 201$
$Fo := a \cdot \frac{\Delta \tau}{\Delta x^2}$	$Fo = 27.778$	$q_v := 0$	$Q_v := Fo \cdot \Delta x^2 \cdot \frac{q_v}{\lambda}$
$Ampl := 0.4$	$v := 0.1$	$Pulse := \begin{pmatrix} Ampl \\ v \end{pmatrix}$	
$Tf := (0 \ 800)^T$	$Bi := (1 \ 1)^T$		
$FT(i, j) := 100$	$T0 := matrix(nX, 1, FT)$		
$F\Lambda(i, j) := 1$	$\Lambda := matrix(nX, 1, F\Lambda)$		

Рис. 9.8. Пример расчета: исходные данные

Результаты расчетов показаны на Рис. 9.9. Трехмерный график дает полное представление о температурном поле. Координатная плоскость в основании диаграммы образована продольной координатой x и временной осью. На вертикальной оси отложена температура. Проследивая линии сетки на волнообразной поверхности в каком-либо нормальном к оси x сечении, мы наблюдаем временную зависимость температуры в фиксированной точке стержня. В сечениях, нормальных к оси времени, получаются распределения температуры вдоль стержня для различных моментов времени. Для построения такой диаграммы в Mathcad следует в окне 3-D Plot Format выбрать закладку Appearance и опцию Wireframe; на закладке Special необходимо отключить опцию Draw Line для z-contour и включить ее для x и y -contours.

Итак, на диаграмме Рис. 9.9 наблюдаются пульсации температуры стержня, вынуждаемые периодическим изменением температуры горячей жидкости на правом торце стержня. Пульсации затухают по мере приближения к левому торцу, интенсивно охлаждаемому жидкостью с постоянной температурой.

Следующая диаграмма (Рис. 9.10) представляет серию температурных распределений вдоль оси x стержня для нескольких последовательных моментов времени. Можно заметить, что имеется начальная стадия (кривая $T_{iX,1}$), когда внутри стержня еще сохраняется область с начальным значением температуры, а затем устанавливаются периодические вынужденные колебания в соответствии с внешним тепловым воздействием.

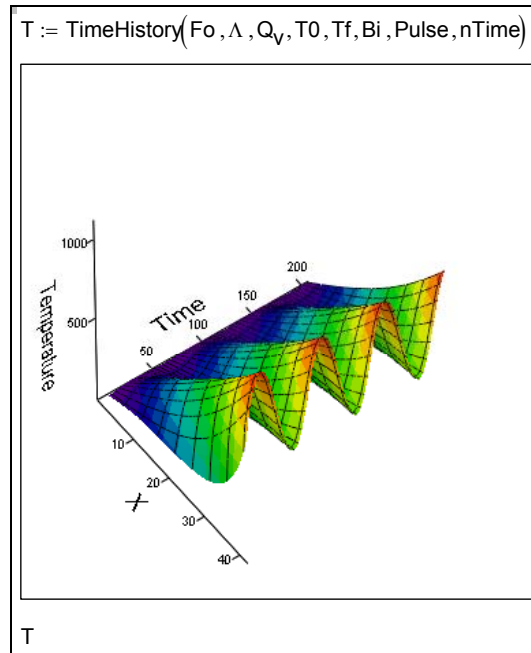


Рис. 9.9. Результаты расчета: температурные волны в трехмерном представлении

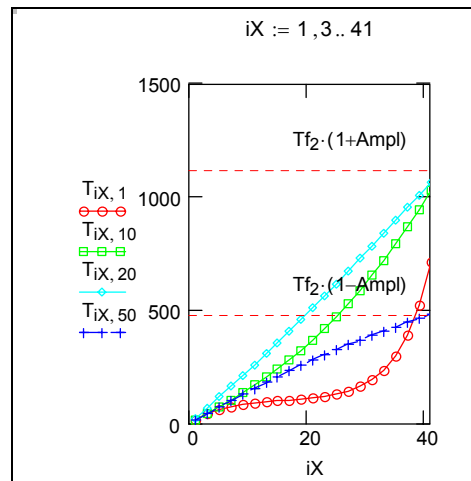


Рис. 9.10. Температурные волны: временные серии распределения температуры по продольной координате

Особенно наглядное представление нестационарного температурного поля получают посредством анимации, как это показано на скриншоте рабочего листа Mathcad (Рис. 9.11).

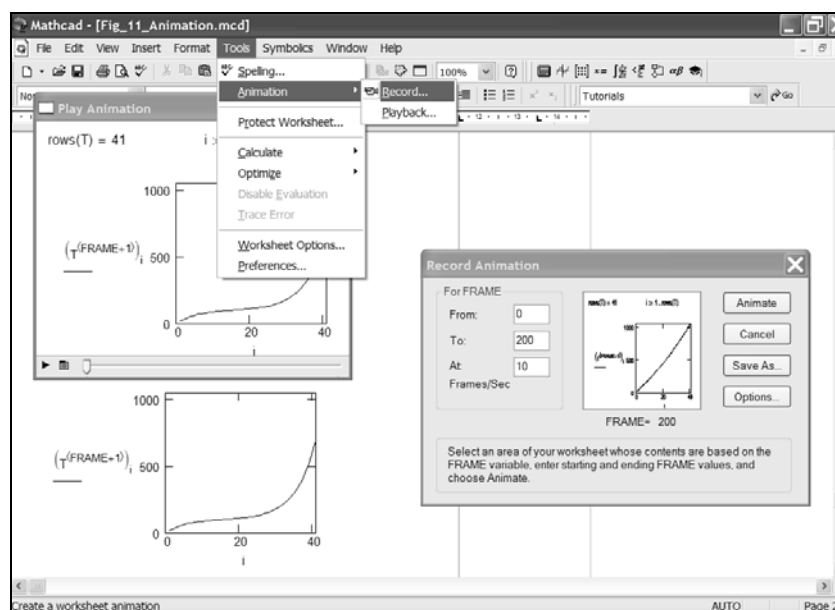


Рис. 9.11. Создание анимированного графика распределения температуры по длине стержня

Демонстрацию можно запустить, щелкнув мышью на гиперссылке на файл видео-клипа:

[..\Mathcad_HMT\9_Num_HeatCond\Fig_11_Animation.avi](#)

Еще один способ презентации тех же расчетных результатов с помощью анимации демонстрируется следующим видео-клипом:

[..\Mathcad_HMT\9_Num_HeatCond\Rod_Pulse_TDMA_Anim.avi](#)

Прямоугольный объект есть изображение стержня с отметками продольной координаты. Цветные поля показывают области с одинаковой температурой.

Далее, в параграфе 9.7, можно увидеть живые примеры создания анимаций, обратившись непосредственно к встроенному Mathcad-документу.

9.6 Встроенная функция Pdesolve

В математическом пакете Mathcad 11 появилась возможность решать одномерные нестационарные уравнения в частных производных посредством новой встроенной функции:

$$\text{pdesolve} \left(u, x, \begin{pmatrix} 0 \\ xMax \end{pmatrix}, t, \begin{pmatrix} 0 \\ tMax \end{pmatrix}, [xpts], [tpts] \right)$$

при обращении к которой возвращается функция пространственной координаты x и времени t как решение уравнения в частных производных, записанного для функции с именем u . Дифференциальное уравнение и краевые условия вводятся в привычной математической нотации между служебным словом **Given** («дано») и обращением к интегратору **pdesolve()** («решить уравнение в частных производных») (см. пример на **Рис. 9.13**).

При записи уравнения и краевых условий для указания частной производной используется буквенный подстрочный индекс. Например, $u_{xx}(x,t)$ есть вторая частная производная от u по x . Буквенный (литеральный) индекс создают нажатием клавиши точки.

Перечислим остальные аргументы функции:

2-элементный вектор-столбец $(0, xMax)^T$ содержит граничные значения x

2-элементный вектор-столбец $(0, tMax)^T$ содержит граничные значения t

$xpts$ (необязательный параметр) задает число точек пространственной дискретизации

$tpts$ (необязательный параметр) задает число точек временной дискретизации.

Результат вычисления в **pdesolve()** присваивается функции с заданным пользователем именем, как в примере на **Рис. 9.13**.

Если **pdesolve** применяется к системе уравнений, то u как формальный аргумент должен быть вектором-столбцом с именами искомых функций в качестве компонентов. Обращение к интегратору выглядит в этом случае следующим образом:

$$\begin{pmatrix} w \\ u \end{pmatrix} = \text{pdesolve} \left(\begin{pmatrix} w \\ u \end{pmatrix}, x, \begin{pmatrix} 0 \\ xMax \end{pmatrix}, t, \begin{pmatrix} 0 \\ tMax \end{pmatrix}, [xpts], [tpts] \right)$$

Дополнительные сведения могут быть получены в Mathcad Help, в том числе о применении в *командной строке* (не в вычислительном блоке **Given**) решателя **pumol** для систем дифференциальных уравнений в частных производных.

Pdesolve базируется на численном *методе линий* (MOL), который применяется для гиперболических и параболических уравнений в частных производных. Для эллиптических уравнений, таких как уравнение Пуассона,

следует использовать встроенные функции `relax` или `multigrid`, а также пользовательскую функцию `Plate` (глава).

Как пример для `pdesolve` рассмотрим ту же задачу, что и в предыдущем параграфе – задачу о температурных волнах в стержне, но с некоторыми изменениями в граничных условиях.

Будем считать физические параметры стержня постоянными и внутренние источники тепла отсутствующими. Рассматриваемая одномерная нестационарная задача описывается дифференциальным уравнением теплопроводности параболического типа:

$$\frac{\partial t}{\partial \tau} = a \frac{\partial^2 t}{\partial x^2}, \quad (9.9)$$

где a – коэффициент температуропроводности.

Следующие три уравнения определяют граничные и начальные условия:

$$\begin{aligned} \left. \frac{\partial t}{\partial x} \right|_{x=+0} &= 0; \\ -\lambda \left. \frac{\partial t}{\partial x} \right|_{x=L-0} &= \alpha_2 (t(x=L, \tau) - t_{f2}); \\ t(x, \tau = 0) &= t_0. \end{aligned} \quad (9.10)$$

Первое из них есть условие Неймана на левом теплоизолированном торце стержня, где задается нулевой тепловой поток.

Второе – условие Роббина (условие третьего рода) на правом торце стержня, причем температура жидкости t_{f2} будет задана как синусоидальная функция времени, а коэффициент теплоотдачи α_2 – как большое число, чтобы обеспечить примерное равенство температур жидкости и правого торца стержня.

Начальное распределение задано в простейшей форме – как постоянная по длине стержня величина.

Основные параметры задачи такие же как на Рис. 9.8. Дополнительные подготовительные вычисления показаны на Рис. 9.12. Коэффициент теплоотдачи на правом торце обозначен как α ; осциллирующая температура жидкости – $Tf2$. Начальная температура $T0$ по длине стержня постоянная и равна средней температуре жидкости $Tf2$.

$\alpha := 100000$	
$T0 := Tf2$	$Tf2(\tau) := Tf2 \cdot (1 + \text{Ampl} \cdot \sin(v \cdot \tau))$
$L := \Delta x \cdot (nX - 1)$	$L = 0.04$
$\tau\text{Max} := \Delta \tau \cdot (n\text{Time} - 1)$	$\tau\text{Max} = 200$

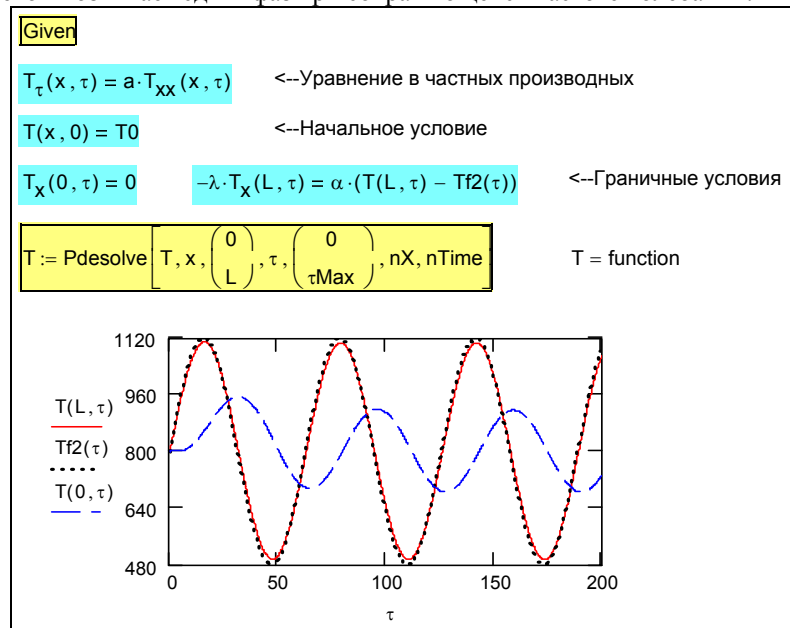
Рис. 9.12. Дополнительные исходные данные для Pdesolve

Итак, мы предполагаем исследовать, как распространяются температурные пульсации вдоль стержня и как выглядят колебания температуры на левом теплоизолированном конце стержня по сравнению с вынужденными пульсациями на правом обогреваемом торце. Краевая задача (9.9), (9.10) вводится в блок **Given...Pdesolve** и результат численного интегрирования присваивается функции T (Рис. 9.13).

Графики отображают:

- Рассчитанную температуру обогреваемого (правого) конца стержня $T(L, \tau)$,
- Заданную пульсирующую температуру жидкости $Tf2(\tau)$,
- Рассчитанную температуру изолированного (левого) конца $T(0, \tau)$.

Температуры $T(L, \tau)$ and $Tf2(\tau)$ практически совпадают благодаря большой величине коэффициента теплоотдачи α . Пульсации затухают вдоль стержня. При этом возникает сдвиг фаз при сохраняющейся частоте колебаний.

Рис. 9.13. Вычислительный блок **Given...Pdesolve**

Полное представление процесса дается трехмерным графиком на Рис. 9.14, созданном встроенной функцией **CreateMesh**. Эта функция генерирует сетку в основании трехмерного графика и для каждого узла этой сетки определяет значения координат $(x, Time)$ и функции T . Получается три матрицы, из которых как из элементов составляется результирующий массив

xTimeT_array. Это замечание полезно для тех, кто хочет детально разобраться с техникой построения графиков, но сейчас достаточно увидеть, что по вертикальной оси откладывается температура, а в горизонтальной плоскости - координата вдоль стержня x и время $Time$.

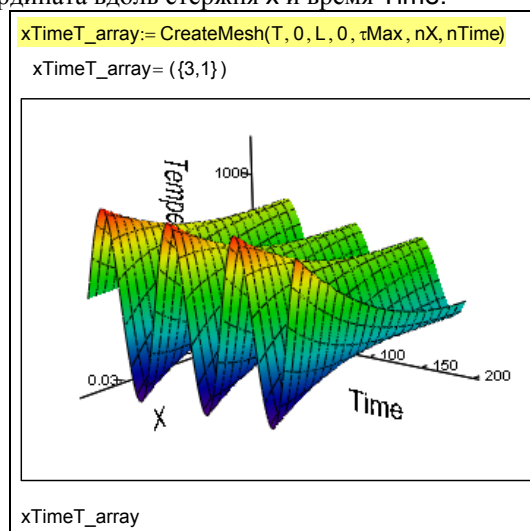


Рис. 9.14. Температурные волны в стержне с изолированным торцем

Анимированное двумя способами распределение температуры вдоль стержня можно увидеть по гиперссылкам:

..\Mathcad_HMT9_Num_HeatCond\Rod_sinPulse_pdesolve_Curve_Anim.avi

..\Mathcad_HMT9_Num_HeatCond\Rod_sinPulse_pdesolve_Anim.avi.

В заключительном численном эксперименте мы стократно уменьшим коэффициент теплоотдачи α на обогреваемом торце. Теперь амплитуда колебаний температуры на обогреваемом торце стержня резко уменьшилась. По-прежнему наблюдаются явления затухания и сдвига фаз при перемещении вдоль стержня (Рис. 9.15).

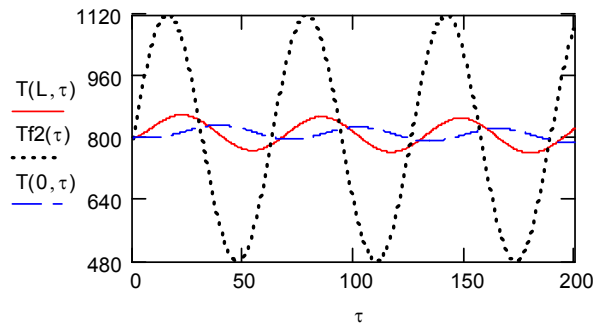


Рис. 9.15. Температурные волны в стержне при малых значениях коэффициента теплоотдачи на обогреваемом торце

Рассмотрим коротко идею численного метода линий (MOL), реализованного в Pdesolve. Возвращаясь к выводу дискретного уравнения (9.3), мы сделаем теперь только половину работы: мы проведем дискретизацию для пространственной координаты x , но мы оставим исходное непрерывное представление для временной переменной τ (для упрощения выкладок мы положили постоянными свойства и сделали нулевым внутренний источник). В результате вместо (9.3) получается следующее уравнение:

$$\frac{dT_P}{d\tau} = \frac{a}{\Delta x^2} (T_W - 2T_P + T_E) \quad (9.11)$$

иногда называемое семидискретизацией (буквально, «полудискретизацией»).

Уравнения вида (9.11) можно записать для всех узлов сетки, с некоторыми модификациями для поверхностных узлов. Тогда вместо уравнения в частных производных получается *система обыкновенных дифференциальных уравнений*, порядок которой равен числу узлов сетки. Решение этой редуцированной задачи, т.е. интегрирование системы обыкновенных дифференциальных уравнений, является хорошо изученной проблемой. Исторически семидискретизация применялась для интегрирования уравнений в частных производных на электронных аналоговых машинах (см., например, []).

9.7 Работа с компьютерными программами

По гиперссылке

[..\Mathcad_HMT\9_Num_HeatCond\Rod_sinPulse_TDMA_Anim.mcd](#)

вы попадете в Mathcad-программу, базирующуюся на методе прогонки (см. параграф 9.4).

Вы можете

- подробно ознакомиться с кодом программы;
- модифицировать этот код в соответствии со своими задачами, например, изменить в программе Coef в блоке Right boundary вид температурных колебаний, заменив синусоиду какой-либо другой функцией;
- провести численный эксперимент, изменяя параметры задачи, например, значения коэффициентов теплоотдачи (параметров Bi);
- потренироваться в создании анимаций.

По гиперссылке

[..\Mathcad_HMT\9_Num_HeatCond\Rod_sinPulse_pdesolve_Anim.mcd](#)

будет вызвана программа решения задачи о волнах в стержне посредством встроенной функции Pdesolve. Граничные условия изменены по сравнению с предыдущим примером: теперь левый торец стержня теплоизолирован, и теплообмен со средой происходит только на правом торце. Тепловой поток попеременно то поступает в стержень, то отводится обратно в жидкость. По такому принципу работают так называемые *регенеративные* теплообменники.

В этом Mathcad-документе можно во всех подробностях проследить за способами визуализации температурного поля и подготовкой анимаций.

В следующем примере

[..\Mathcad_HMT\9_Num_HeatCond\Rod_squarePulse_pdesolve_Anima.mcd](#)

изменена форма вынужденных колебаний на правом торце: теперь это попеременное обтекание горячей и холодной жидкостью со ступенчатым временным профилем температуры.

Численные эксперименты на компьютерных моделях могут быть эффективным средством оптимизации температурных режимов при специальной термической обработке деталей с целью упрочнения поверхности, как показано в программе

[..\Mathcad_HMT\9_Num_HeatCond\case-hardening_pdesolve.mcd](#)

Такую обработку проводят посредством мощного лазерного импульса, обеспечивающего нагрев тонкого приповерхностного слоя до высокой температуры. Затем происходит быстрое охлаждение этого слоя благодаря теплоотводу внутрь материала. Мощность импульса (или плотность потока излучения q_0) и его длительность (τ_{Pulse}) должны быть подобраны таким образом, чтобы обеспечить нужный температурный режим закалки в слое заранее заданной (порядка нескольких сотен микрометров) толщины.

Программирование начинается с формирования лазерного импульса, как показано на Рис. 9.16.

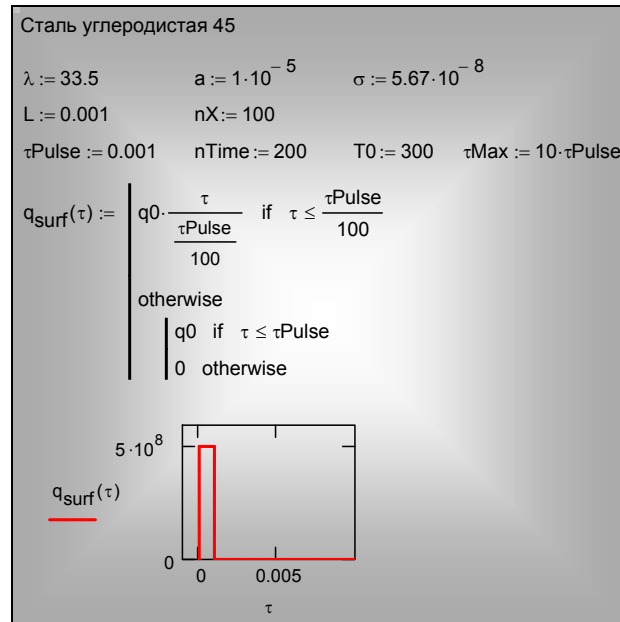


Рис. 9.16. Формирование ступенчатого теплового импульса

При постановке краевой задачи теплопроводности (Рис. 9.17) на левой граничной поверхности задается условие адиабатической изоляции, а на правой границе, подверженной лазерному воздействию, ставится условие второго рода (условие Неймана), причем тепловой поток определяется лазерным импульсом и тепловым излучением стержня в окружающую среду.

Температурное поле $T(x, \tau)$ рассчитывается посредством функции Pdesolve. На графике (Рис. 9.17) видно, что температура облучаемой поверхности $T(L, \tau)$ достигает высокого максимального значения в конце импульса и далее очень быстро убывает во времени, что и обеспечивает закалку.

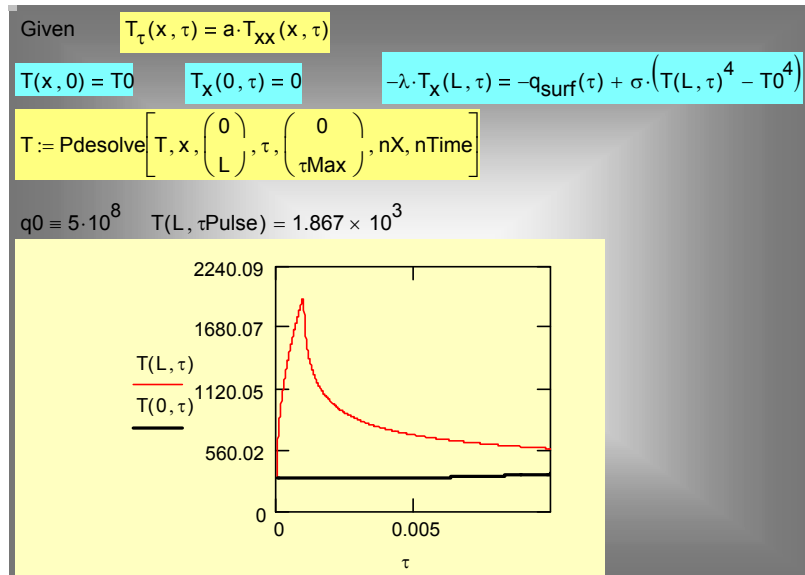


Рис. 9.17. Численное интегрирование задачи об импульсном воздействии

Представление о глубине закаливаемого слоя дает пространственно-временная диаграмма на Рис. 9.18. Изменяя при численных экспериментах параметры импульса, можно подобрать необходимые технологические параметры термической обработки.

Анимацию температурного поля при импульсном воздействии можно посмотреть по ссылке:

..\Mathcad_HMT\9_Num_HeatCond\case-hardening_pdesolve.avi.

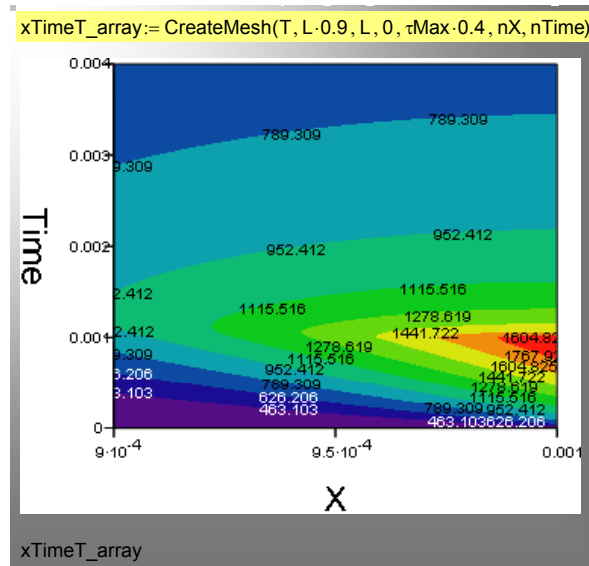


Рис. 9.18. Пространственно-временная диаграмма распределения температуры в стержне при импульсном воздействии

Мы привели несколько примеров решения одномерного нестационарного уравнения теплопроводности. Численные эксперименты можно продолжить в следующих направлениях:

- Исследовать влияние коэффициента теплоотдачи на температуру горячего торца (на ее максимальное и среднее значение).
- Исследовать влияние свойств материала на максимальную температуру поверхности и теплоотвод через стержень.
- Исследовать глубину проникания температурных волн, взяв длинный стержень и задав адиабатические условия на левом торце.
- Исследовать теплопередачу через стенку, если коэффициент теплоотдачи пульсирует во времени (для этого понадобится несколько модифицировать программу `Coef`, обеспечив вариации числа Био так же, как сейчас это сделано с температурой жидкости).
- Исследовать температурные режимы стенок цилиндра двигателя внутреннего сгорания с воздушным охлаждением.
- Исследовать температурные режимы стен здания при погодных и сезонных изменениях температуры.
- Решить обратную задачу, определяя температуру на поверхности по измеренным значениям в глубине.

- И т.д.

Большинство классических задач теплопроводности, рассматриваемых в учебных курсах тепломассообмена, можно проанализировать в режиме численного эксперимента на разработанной в параграфе 9.4 компьютерной модели. Методом *счета на установление* решаются стационарные задачи теплопроводности вдоль ребра. Комплекс учебных исследований может быть выполнен для стационарных и нестационарных задач с внутренними источниками теплоты – с целью имитации работы тепловыделяющих элементов ядерного реактора, как показано на анимации:

[..\Mathcad_HMT\9_Num_HeatCond\Fuel_El.avi](#)

Возвращаясь к Рис. 9.13, на котором показано применение встроенного в Mathcad решателя уравнения в частных производных, отметим, что дифференциальное уравнение и краевые условия записаны в обычной математической нотации, а обращение к интегратору Pdesolve занимает всего одну строчку. В действительности за обращением к интегратору стоит сложная вычислительная программа (см. пояснения к формуле (9.11)), но это остается *скрытым* от пользователя. По существу, численный метод переводится в категорию рутинных автоматизированных операций, благодаря чему внимание исследователя может быть полностью сосредоточено на главной творческой задаче – разработке адекватной математической модели.

Мы привели здесь краткое описание тенденции в развитии компьютерного моделирования. Она отчетливо видна для относительно простых задач, подобных рассмотренной выше, и в меньшей степени заметна для более сложных, таких как трехмерное нестационарное моделирование конвективного тепломассообмена в областях сложной геометрии.

По-видимому, полная автоматизация и стандартизация построения моделей тепломассообмена еще далеко впереди. В настоящее время многие исследователи и расчетчики предпочитают разрабатывать собственные программы и сохранять полный контроль над кодом вычислительных программ, выбором численных алгоритмов и т.д., не без основания полагая, что это обеспечит надежность получаемых результатов (и/или избавит от проблем лицензионной чистоты). Мы поступили таким образом, разработав в параграфе 9.4 свои программы в Mathcad. Прямой доступ к коду программ – с целью усовершенствования, постановки других краевых условий и т.п. – можно получить в параграфе 9.7, где помещены гиперссылки на Mathcad-документы.

9.8 Численные методы для многомерных задач

Образ платы компьютера как пластины, нагретой в одних местах довольно сильно и холодной в других – хороший логотип для двумерного температурного поля (Рис. 9.19). Обеспечение необходимого температурного режима настолько важно для надежного функционирования компьютера, что предусматривается установка датчиков температуры, показания которых можно проконтролировать, запустив утилиту аппаратной диагностики типа Hardware Doctor (Рис. 9.19). Шум работающих в корпусе системного блока микровентиляторов – постоянное напоминание о проблеме охлаждения компьютера.

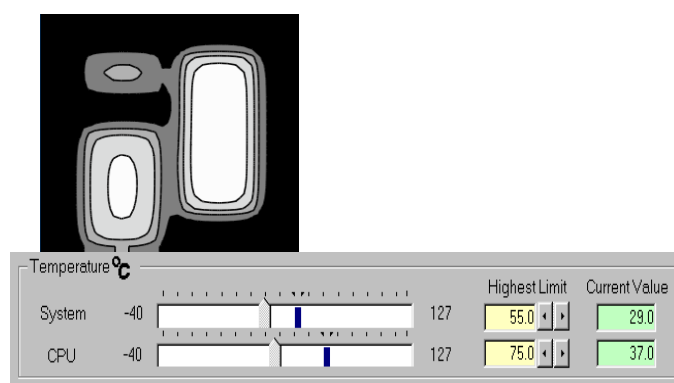


Рис. 9.19. Температурное поле платы компьютера и показания датчика для системного блока (29°C) и процессора (37°C)

Показания датчиков температуры дают важную, но все же очень ограниченную информацию о температурном режиме компьютера, поскольку измерения произведены только в нескольких точках. Далее мы рассчитаем и сделаем видимым *температурное поле* компьютерной платы, т.е. распределение температуры по всей поверхности, включая различные микросхемы и сильно греющийся процессор.

Схематическое модельное представление компьютерной платы как теплового объекта поясняется на Рис. 9.20, Рис. 9.21.



Рис. 9.20. Схематический образ платы компьютера

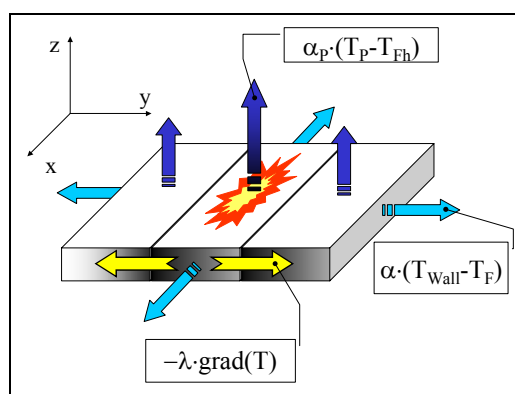


Рис. 9.21. Схема процессов переноса теплоты в плате компьютера

В микросхемах выделяется теплота, которая распространяется вдоль платы посредством теплопроводности. С верхней и нижней поверхностями происходит теплоотдача в окружающую среду (см. Рис. 9.21). Следует учитывать также теплоотвод с боковых (торцевых) граней.

При включении питания компьютерная плата будет разогреваться, но затем температура установится на некотором уровне, а внутреннее тепловыделение будет полностью компенсироваться теплоотводом в окружающую среду. Такой режим называется *стационарным*, не меняющимся во времени.

Строго говоря, температурное поле будет *трехмерным*, т.е. температура будет меняться как вдоль платы (вдоль осей x , y , которые мы расположим в плоскости платы), так и по ее толщине (по оси z). Однако изменениями по оси z можно пренебречь, если плата тонкая, ее теплопроводность достаточно велика, а коэффициенты теплоотдачи на верхней и нижней поверхностях относительно малы. Примем эти предположения и будем считать температурное поле *двумерным*, а *теплоотвод с верхней и нижней поверхностей* (т.е. в направлении оси z) имитируем *внутренним стоком*

теплоты, т.е. отрицательным источником. Это замечание станет более понятным позже, при записи уравнения баланса (9.18).

Как было показано в гл. 1, стационарное двумерное температурное поле $T(x,y)$ в объекте с коэффициентом теплопроводности λ , Вт/(м К) и с действующими внутренними источниками теплоты мощностью q_V , Вт/м³ описывается следующим дифференциальным уравнением:

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} = -\frac{q_V}{\lambda} .$$

(9.12)

Мы рассчитаем далее температурное поле в плате компьютера посредством численного интегрирования этого уравнения.

Встроенные функции для решения уравнений в частных производных

Как уже говорилось, Mathcad располагает весьма ограниченными средствами для решения уравнений в частных производных. Используя встроенные функции `multigrid` и `relax`, можно получить решение уравнения Пуассона

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = \rho(x, y)$$

(9.13)

в *квадратной* области, если на границах задано

- постоянное *нулевое* значение искомой величины u (используется `multigrid`);
- *некоторое распределение* вдоль границы (используется `relax`).

Таким образом, в обоих случаях задача решается при граничных условиях первого рода (задача Дирихле).

Записав обращение к функции `multigrid`,
`u:=multigrid(M,ncycle)`,

получают квадратную матрицу значений искомой функции $u(x,y)$ в квадратной области. Матрица M размером $(n+1) \times (n+1)$ с элементами M_{ij} , $i := 0..n$, $j := 0..n$, задает внутренние источники (стоки) в узлах сетки $(n+1) \times (n+1)$. Рекомендуемые значения расчетного параметра `ncycle` есть 1 или 2. Напомним еще раз, что по умолчанию предполагается задание нулевых граничных условий.

Обращение к функции `relax` выглядит сложнее:

`u := relax(A, B, C, D, E, F, U, rjac)`.

(9.14)

Квадратные матрицы A, B, C, D, E определяют коэффициенты линейной аппроксимации лапласиана (оператора в левой части уравнения Пуассона) в

каждой точке расчетной сетки. Матрица F дает значения функции источника. Матрица U предписывает граничные значения вдоль четырех сторон, а также задает начальное приближение во внутренних точках. Расчетный параметр τ_{jac} контролирует сходимость итераций; его оптимальное значение из интервала $0..1$ зависит от индивидуальности задачи.

В документации, непосредственно доступной в системе Help, не содержится никакой дополнительной информации о смысле матриц коэффициентов A, B, C, D, E , а из примеров можно выяснить только то, что их стандартными элементами являются соответственно 1, 1, 1, 1 и (-4) .

Целью дальнейшего изложения являются необходимые пояснения того, как формируются указанные матрицы и как их можно модифицировать в зависимости от постановки задачи. Первым результатом будет существенное расширение возможностей встроенной функции `relax`: будет указан способ модификации матриц E и F (см. (9.14)), который позволит учесть теплосъем с поверхностей пластины. Кроме того, будет показано, как построить алгоритм и Mathcad-программу для решения задач с граничными условиями более общего вида, чем только условия Дирихле, и будет рассмотрен практический пример: решение задачи о температурном поле компьютерной платы.

Представление дифференциального уравнения теплопроводности в конечно-разностной форме

При численном интегрировании уравнения (9.12) задачу сводят к расчету температуры в конечном числе точек – узлов сетки, вокруг которых формируют так называемые *контрольные объемы*. Фрагмент сетки и типичный контрольный объем показаны на Рис. 9.22. Конечно-разностную аппроксимацию дифференциального уравнения получают, записывая тепловой баланс для контрольного объема $\delta x \delta y \delta z$.

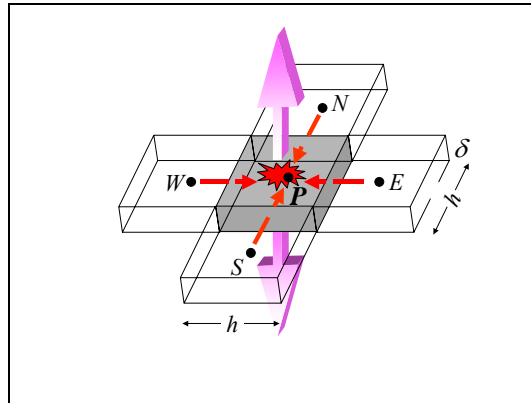


Рис. 9.22. Контрольный объем и потоки теплоты

Внутренняя энергия контрольного объема будет изменяться во времени по следующим причинам:

- Происходит поступление теплоты через южную, северную, восточную и западную границы из соседних объемов, температуру в которых мы для определенности считаем более высокой; например через западную грань

$$\lambda \frac{T_W - T_P}{h} (\delta \cdot h).$$

(9.15)

- Имеется теплоотдача в окружающую среду с температурой T_{jh} через верхнюю и(или) нижнюю поверхности

$$\alpha_P (T_P - T_{jh}) (h \cdot h).$$

(9.16)

- Действует внутреннее тепловыделение

$$q_V (\delta \cdot h \cdot h).$$

(9.17)

Суммируя составляющие баланса, получим:

$$\underbrace{\lambda \frac{T_S - T_P}{h} (\delta \cdot h) + \lambda \frac{T_N - T_P}{h} (\delta \cdot h) + \lambda \frac{T_E - T_P}{h} (\delta \cdot h) + \lambda \frac{T_W - T_P}{h} (\delta \cdot h)}_{\text{Теплоподвод посредством теплопроводности}} +$$

$$+ \underbrace{\left[q_V (\delta \cdot h \cdot h) - \alpha_P (T_P - T_{fn}) (h \cdot h) \cdot 2 \right]}_{\substack{\text{Внутреннее тепловыделение за вычетом} \\ \text{теплоотдачи с верхней и нижней поверхностей}}},$$

$$= \underbrace{\rho c (\delta \cdot h \cdot h) \left(\frac{T_P - T_{0P}}{\Delta \tau} \right)}_{\text{Увеличение внутренней энергии}}$$

(9.18)

где T_{0P} – температура в предыдущий момент времени, $\Delta \tau$ – временной шаг.

Уравнение баланса (9.18) следует теперь представить в форме линейного алгебраического уравнения относительно неизвестных температур:

$$AT_S + BT_N + CT_E + DT_W + ET_P = S.$$

(9.19)

Чтобы обратиться к какой-либо программе-решателю, необходимы выражения для коэффициентов A, B, C, D, E, S . Мы получим их, сопоставляя форму представления (9.19) и оригинал (9.18). Ограничимся случаем *стационарного* состояния ($T_P = T_{0P}$) и примем $\lambda = \text{const}$. Mathcad выполняет необходимые выкладки, если к (9.18) применить оператор `collect`. Заметим, что имя `balance` присвоено левой части (9.18), равной нулю для стационарной задачи.

$$\text{balance} := \frac{\lambda \cdot (T_W - T_P)}{h} \cdot (\delta \cdot h) + \frac{\lambda \cdot (T_E - T_P)}{h} \cdot (\delta \cdot h) + \frac{\lambda \cdot (T_N - T_P)}{h} \cdot (\delta \cdot h) \dots$$

$$+ \frac{\lambda \cdot (T_S - T_P)}{h} \cdot (\delta \cdot h) + q_V \cdot (\delta \cdot h \cdot h) - \alpha_P \cdot (T_P - T_{fn}) \cdot h \cdot h \cdot 2$$

$$\frac{\text{balance}}{\lambda \cdot \delta} \begin{array}{l} \text{expand} \\ \text{collect, } T_P \\ \text{(collect, } T_W, T_E, T_N, T_S) \end{array} \rightarrow$$

$$T_W + T_E + T_N + T_S + \frac{(-4 \cdot \lambda \cdot \delta - 2 \cdot \alpha_P \cdot h^2)}{\lambda \cdot \delta} \cdot T_P + \frac{(q_V \cdot h^2 \cdot \delta + 2 \cdot \alpha_P \cdot h^2 \cdot T_{fn})}{\lambda \cdot \delta}$$

Рис. 9.23. Коэффициенты конечно-разностного уравнения

В результате основное уравнение (9.18) приводится к виду:

$$T_S + T_N + T_E + T_W - (4 + \beta)T_P = S,$$

(9.20)

где

$$\beta \equiv \frac{\alpha_P h^2 \cdot 2}{\lambda \delta}; \quad S \equiv -\frac{q_V h^2}{\lambda} - \beta T_{fh}.$$

(9.21)

Коэффициенты A, B, C, D равны единице, а для E получаем выражение:

$$E = -(4 + \beta).$$

(9.22)

Добавочное слагаемое β учитывает теплоотдачу на горизонтальных поверхностях пластины (см. рис. 10.3, 10.4). Необязательно коэффициент теплоотдачи α_P и температура среды T_{fh} должны быть одинаковы на обеих (верхней и нижней) сторонах пластины или постоянны по поверхности. Необходимые изменения нетрудно ввести в (9.18) и в вычисления на Рис. 9.23. Например, величину α_P можно рассматривать как среднее значение коэффициента теплоотдачи на верхней (“up”) и нижней (“down”) поверхностях $(\alpha_{Pup} + \alpha_{Pdown})/2$, а температуру T_{fh} – как средневзвешенное значение $(\alpha_{Pup} T_{fh,up} + \alpha_{Pdown} T_{fh,down}) / (\alpha_{Pup} + \alpha_{Pdown})$.

Для граничных узлов, расположенных на боковых гранях (Рис. 9.21), температуры либо заданы, если ставятся граничные условия первого рода (условия Дирихле), либо для них записываются специальные уравнения, как будет показано ниже (см. уравнение (9.24)). Так или иначе, получается система линейных уравнений вида (9.19) с порядком, равным числу узлов сетки.

Например, если взять по каждой стороне пластины по 33 узла, то всего получится $33 \cdot 33 = 1089$ узлов сетки. Матрица коэффициентов такой системы будет содержать $1089 \cdot 1089 = 1185921$ элементов. Это много даже для современных компьютеров. В настоящее время трудности при решении больших систем в значительной степени преодолены, однако численное моделирование многомерных задач тепломассообмена остается искусством и требует применения самых больших вычислительных мощностей.

Метод Гаусса–Зейделя. Программа Plate

Уравнение для внутренних узлов

Один из простейших алгоритмов решения получающейся системы уравнений основан на итерационной процедуре Гаусса–Зейделя [], когда последующие приближения получаются из предыдущих по формуле (9.20), переписанной относительно температуры в рассматриваемом узле P :

$$T_P = \frac{S - T_S - T_N - T_E - T_W}{E}, \quad E = -(4 + \beta).$$

(9.23)

Эта формула годится только для внутренних узлов сетки. Если температура в узлах на границе задана, т.е. поставлены граничные условия *первого рода*, то больше ничего не требуется.

Уравнение для поверхностных узлов

Чаще всего на практике ставятся *граничные условия третьего рода*. При этом задается температура окружающей среды T_f и коэффициент теплоотдачи α на поверхности (на тонких боковых гранях $\delta \times h$ пластины).

Приближенная математическая формулировка условия *третьего рода* будет следующей:

$$\alpha(T_f - T_{wall})(\delta \cdot h) \cong \lambda \frac{T_{wall} - T_{inner}}{h} (\delta \cdot h) \quad (9.24)$$

Вводя безразмерный коэффициент теплоотдачи (сеточное число Био) как

$$Bi \equiv \frac{\alpha h}{\lambda}, \quad (9.25)$$

получим удобное для итераций выражение температуры в узлах на (боковой) поверхности T_{wall} через температуру окружающей среды T_f и температуру ближайшего внутреннего узла T_{inner} :

$$T_{wall} = \frac{Bi \cdot T_f + T_{inner}}{1 + Bi}. \quad (9.26)$$

На основе соотношений (9.23) и (9.26) построена Mathcad-программа **Plate** для расчета температуры в прямоугольной (не обязательно квадратной) области с учетом теплоотдачи на *верхней и нижней* поверхностях. На четырех *боковых* гранях ставятся граничные условия третьего рода.

Эта программа позволяет решать широкий круг практически важных задач. В дополнение к примеру с компьютерной платой укажем на расчет пластинчатого оребрения, применяемого с целью интенсификации теплопередачи в теплообменных устройствах, или на расчет пластинчатого радиатора космической энергетической установки.

В следующем пункте приведен подробный протокол Mathcad-документа, который в оригинале имеет название `relax_vs_Plate.mcd` и в котором сопоставляется работа встроенной функции `relax` и пользовательской функции `Plate`.

Тепловая модель платы компьютера

Вычислительная программа `Plate`, представленная на Рис. 9.24, возвращает

- стационарное температурное поле платы T как двухмерный массив с индексами $i=0..m$, $j=0..n$, т.е. с размерностью $(m+1) \times (n+1)$;
- относительную ошибку $relErr$ как разность между двумя последними итерациями,
- число итераций n_iter ;

Точность tol и максимальное число итераций $maxiter$ заданы непосредственно в тексте процедуры, поскольку эти параметры редко приходится изменять.

Перечислим входные параметры процедуры **Plate**:

- E – коэффициент в определяющем уравнении (9.23). Задается как матрица той же размерности, что и температура T в узлах сетки. Вычисляется через значения β – безразмерного коэффициента теплоотдачи на верхней и нижней поверхности платы (см. формулы (9.21) и (9.22) и рисунки: Рис. 9.21 и Рис. 9.22).
- S – правая часть определяющего уравнения (9.19) или (9.20), матрица такой же размерности, что и T . Вычисляется по формуле (9.21) через мощность внутренних источников q_V (т.е. Джоулеву теплоту, выделяющуюся в микросхеме) и теплоотдачу β .
- IC – начальное распределение температуры, матрица такой же размерности, что и T .
- Bi – число Био, безразмерный коэффициент теплоотдачи на боковых торцах платы (см. формулу (9.25)); матрица размерности четыре, по числу боковых граней платы.
- Tf – температура среды на боковых торцах платы (см. Рис. 9.21 и формулу (9.26)); матрица размерности четыре, по числу боковых граней платы.

Структура программы **Plate** следующая. Внешний цикл **for** по переменной **iter** организует итерации. Если максимальная разность в двух последовательных итерациях становится меньше величины tol , то по команде **break** вычисления в цикле прекращаются и выводятся результаты. Результат предыдущей итерации хранится в массиве **T0**.

Внутренний двойной цикл по переменным i, j использует формулу (9.23) для итераций во *внутренних* точках области. В двух последних циклах производится обход по границам, а для итераций применяется формула (9.26).

```

Plate(E, S, IC, Bi, Tf) ≡ [(m ← rows(IC) - 1) (n ← cols(IC) - 1)]
[(tol ← 0.001) (maxIter ← 100)]
T ← IC
for iter ∈ 1..maxIter
  for i ∈ 0..m
    for j ∈ 0..n
      T0i,j ← Ti,j
    for i ∈ 1..m - 1
      for j ∈ 1..n - 1
        Ti,j ←  $\frac{(S_{i,j} - T_{i,j+1} - T_{i+1,j} - T_{i,j-1} - T_{i-1,j})}{E_{i,j}}$ 
    for i ∈ 0..m
      [ [ T1,0 ←  $\frac{(T_{1,1} + Bi_0 \cdot Tf_0)}{(1 + Bi_0)}$  ] [ T1,n ←  $\frac{(T_{1,n-1} + Bi_1 \cdot Tf_1)}{(1 + Bi_1)}$  ] ]
    for j ∈ 0..n
      [ [ T0,j ←  $\frac{(T_{1,j} + Bi_2 \cdot Tf_2)}{(1 + Bi_2)}$  ] [ Tm,j ←  $\frac{(T_{m-1,j} + Bi_3 \cdot Tf_3)}{(1 + Bi_3)}$  ] ]
    Tm ← max( $\vec{T}$ )
    Tm ← 0.01 if Tm < 0.01
    [ [ relErr ←  $\frac{\max(\vec{|T - T_0|})}{T_m}$  ] (n_iter ← iter) ]
    break if relErr < tol
(T relErr n_iter)T

```

Рис. 9.24. Функция Plate для численного решения уравнения Пуассона

Поскольку задача будет параллельно решаться с помощью встроенной функции `relax`, обращение к которой записывается как (9.14), напомним, что для равномерной сетки и для постоянного значения теплопроводности коэффициенты A, B, C, D имеют единичные значения (см. комментарий к определяющему уравнению (9.18)). Коэффициенты E вычисляются с учетом теплоотдачи на горизонтальных (верхней и нижней) поверхностях пластины, так же как для программы `Plate`. Матрицы коэффициентов F и U в (9.14) суть то же, что S и IC в списке параметров для `Plate`.

После этих предварительных замечаний приведем протокол работы с Mathcad-документом (Рис. 9.25). В блоках программы производятся следующие вычисления:

- 1) Задается разбиение квадратной области n , вводятся значения безразмерного коэффициента теплоотдачи β на поверхности пластины и параметра тепловыделения S_0 (в градусах К).
- 2) Формируются матрицы коэффициентов уравнения (9.19), задается начальное условие как массив значений температуры `Tinit`, очищается массив `Source`, в котором будут записаны значения внутреннего тепловыделения.

- 3) задается внутреннее тепловыделение в микросхемах – трех подобластях платы.
- 4) Для программы **Plate** задаются граничные условия, имитирующие отсутствие теплосъема с тонких боковых граней пластины, т.е. вводятся нулевые безразмерные коэффициенты теплоотдачи вдоль четырех боковых граней и значения температуры среды (последние не обязательно нулевые).

(1)	$n := 32$	$\beta := 0.5$	$S_0 := -50$
(2)	$i := 0..n$	$j := 0..n$	
	$A_{i,j} := 1$	$B_{i,j} := 1$	$C_{i,j} := 1$
	$D_{i,j} := 1$	$E_{i,j} := -(4 + \beta)$	
	$T_{init,j} := 0$	$Source_{i,j} := 0$	
(3)	$i := 18..26$	$j := 6..n-6$	$Source_{i,j} := S_0$
	$i := 6..14$	$j := 2.. \frac{n}{2}$	$Source_{i,j} := \left(\frac{S_0}{1.2} \right)$
	$i := 6..14$	$j := 22..26$	$Source_{i,j} := \frac{S_0}{2}$
(4)	$T_f := (0 \ 0 \ 0 \ 0)^T$	$B_i := (0 \ 0 \ 0 \ 0)^T$	

Рис. 9.25. Подготовка исходных данных

Далее производится обращение к пользовательской функции **Plate** и к встроенной функции **relax**. Результаты решения представлены на Рис. 9.26 и Рис. 9.27.

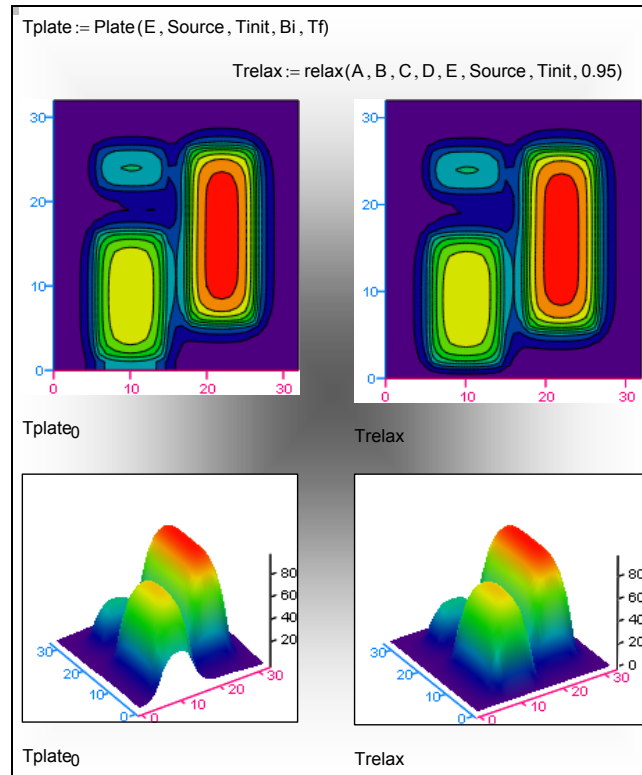


Рис. 9.26. Результаты численного моделирования температуры платы компьютера (Plate – слева, relax – справа)

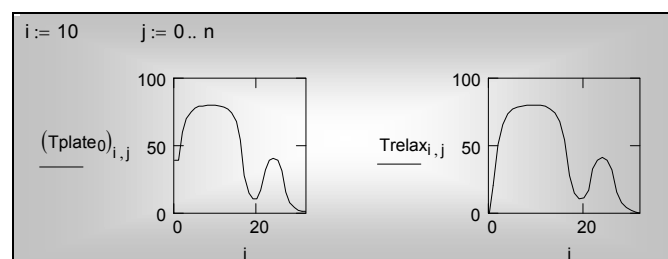


Рис. 9.27. Распределение температуры вдоль линии $i=10$

Цвет изображения (или градации серого) воспроизводит различные значения температуры. Области с высокой температурой – это три

микросхемы. Процессор компьютера (самая большая микросхема) нагреет наиболее сильно.

Способ визуализации отражает особенности профессионального – для специалиста в области теплообмена – видения платы компьютера. Мы отслеживаем изменения температуры от точки к точке, внимание концентрируется на областях с *разной температурой* и в особенности там, где температура может превысить допустимое значение.

Как видно из рисунков, решение двумя способами дает в целом близкие результаты. Однако имеются заметные отличия вблизи краев пластины (платы). Это следствие различной постановки граничных условий.

Для встроенной функции **relax** имеется единственная возможность, а именно, задать температуру границы. Какова будет эта температура, заранее не известно. Можно только предполагать, что если тепловыделение не слишком велико и микросхемы расположены на удалении от кромок, то температура по краям платы должна быть близка к температуре окружающей среды. В примере с **relax** температура границ была задана нулевой. Заметим сразу, что высказанные предположения могут не соответствовать действительности, и тогда моделирование методом **relax** приведет к грубым ошибкам.

Для программы **Plate** имеется возможность задавать любые граничные условия. Самым подходящим выглядит предположение о пренебрежимо малом теплоотводе с узких боковых граней платы. Такое условие можно имитировать, задавая нулевые значения коэффициента теплоотдачи, что и сделано в примере. Обратите внимание, температура платы на краях действительно не нулевая. Особенно выделяется область повышенной температуры там, где одна из микросхем расположена близко к границе (см. Рис. 9.26 и Рис. 9.27).

Таким образом, решение методом **relax** привело к заметной ошибке из-за неправильной постановки граничного условия, а решение методом **Plate** обеспечило реалистичный результат.

Умение правильно (адекватно реальности) формулировать краевые условия есть основа искусства математического моделирования задач тепломассообмена. Соответственно, хороший инструмент решения этих задач должен предоставлять такую возможность. Встроенной функции **relax** недостает средств постановки граничных условий, позволяющих адекватно представлять взаимодействие объекта с окружающей средой. Программа **Plate** (Рис. 9.24) восполняет этот пробел. Численные эксперименты с температурным полем платы компьютера можно провести, обратившись к программе:

[..\Mathcad_HMT\9_Num_HeatCond\Computer_Board.MCD](#)

9.9 Заключение

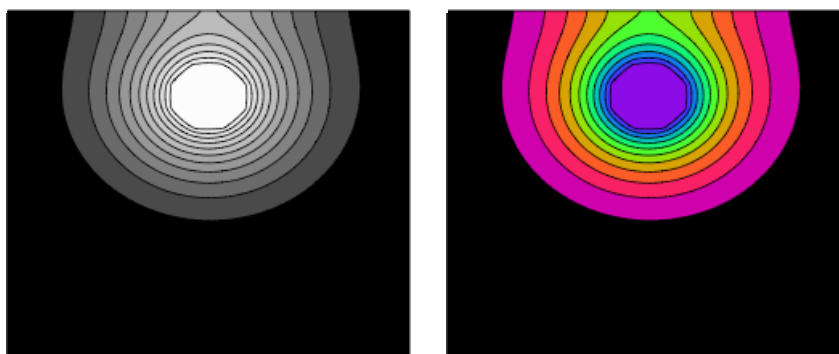
Задачи, возникающих в инженерной практике, могут потребовать более мощных вычислительных программ, чем применявшиеся нами. Рассмотрим в качестве примера компьютерное моделирование с целью управления тепловыми режимами магистральных газопроводов [1].

Тысячекилометровые нити газопроводов, проложенные на Севере, в полной мере испытывают на себе воздействие проблемы, которая называется *неустойчивостью грунтовых оснований в условиях вечной мерзлоты*. При вспучивании грунта, протаивании или промерзании на отдельных участках возникают недопустимые деформации вплоть до разрыва газопроводов.

Компьютерная модель должна обеспечить расчет полей температур и тепловых потоков в системе газопровод–грунт на участках большой продольной протяженности и для больших промежутков времени, с учетом климатических, сезонных и погодных колебаний внешних условий.

По крайней мере двумя необходимыми свойствами *не обладают* находящиеся в нашем распоряжении *relax* и *Plate*, чтобы решать такие задачи:

- Они неспособны рассчитывать нестационарные температурные поля.
- Трудно задать необходимую геометрическую конфигурацию – «проложить» газопровод, т.е. сделать в массиве круглое отверстие и задать на получившейся внутренней поверхности нужные граничные условия третьего рода.



T

T

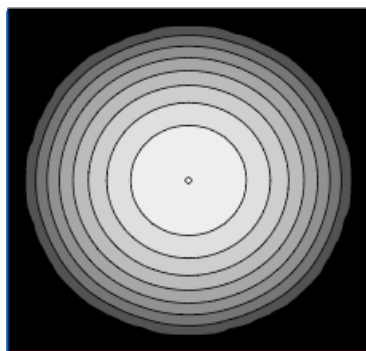
Рис. 9.28. Температурное поле в грунте вокруг трубопровода

Мы модифицировали программу *Plate* следующим образом, чтобы все же провести демонстрационные расчеты такого уровня сложности:

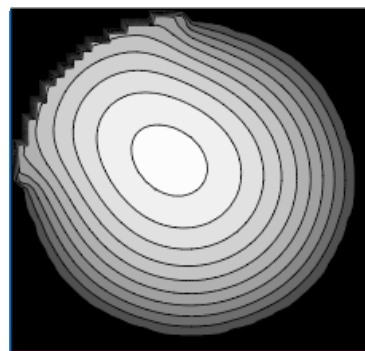
- Во-первых, пришлось вернуться к исходному уравнению баланса (9.18) и сохранить в нем нестационарный оператор. В связи с этим изменилось основное расчетное уравнение (9.23).
- Во-вторых, потребовалось идентифицировать узлы сетки, а именно, определить, какие из них остаются в расчетной области, а какие попадают в отверстие. Среди попадающих в расчетную область было необходимо отметить те из узлов, которые оказываются на внутренней (круглой) границе области, и применить для расчета температур в этих узлах специальное граничное условие.

Указанные усовершенствования сделали Mathcad-программу довольно громоздкой, и мы не приводим ее здесь. Демонстрационный пример расчета по этой программе (Рис. 9.28) показывает, каким образом зимой над неглубоко заложенной теплотрассой может оказаться оттаявшая земля.

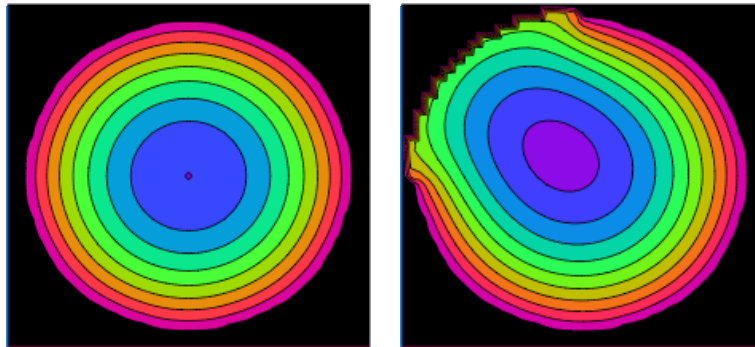
Еще один пример работы модифицированной программы **Plate** – расчет температурного поля в тепловыделяющем элементе (ТВэле) ядерного реактора (Рис. 9.29). Задача о ТВэле была решена ранее в одномерной, осесимметричной постановке. Однако на практике симметрия может быть нарушена, и тогда необходимо рассчитывать двумерное температурное поле. Результаты численного интегрирования представлены на Рис. 9.29. Слева показана штатная ситуация с достаточным и равномерным по окружности охлаждением. Справа продемонстрирована аварийная ситуация, когда условия охлаждения в левой верхней четверти окружности ТВэла ухудшились, и изотермы с высокой температурой вплотную приблизились к защитной циркониевой оболочке, допустимая температура которой относительно невелика – около 400°C. При более высокой температуре в контакте с охлаждающей водой быстро развивается разрушительная коррозия.



Т



Т



T

T

Рис. 9.29. Температурное поле в тепловыделяющем элементе ядерного реактора

Для расчетов полей в сложных геометрических областях часто применяют метод конечных элементов (МКЭ) [1]. Одним из главных отличительных признаков МКЭ является нерегулярная, адаптирующаяся к геометрии объекта и особенностям температурного поля сетка (см. Рис. 9.30).

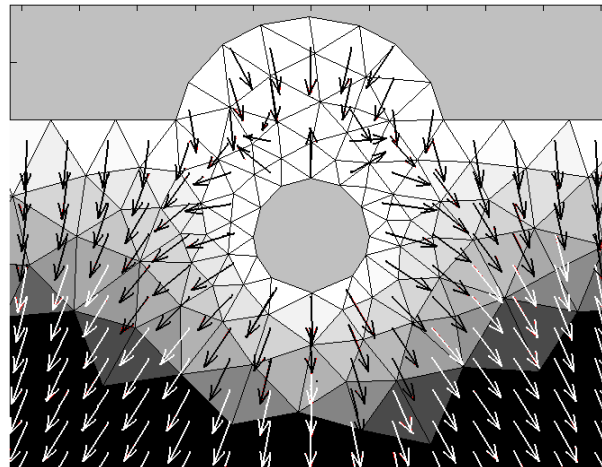


Рис. 9.30. Расчет температурного поля методом конечных элементов

Большие инженерные задачи, такие как полный расчет ядерного энергетического реактора или моделирование аэродинамики автомобиля, требуют применения специальных математических пакетов. На базе МКЭ построен универсальный вычислительный комплекс ANSYS

(<http://www.cadfem.ru>, <http://www.ansys.com>), предназначенный для решения задач различной физической природы (прочность, теплофизика, гидрогазодинамика, тепломассообмен, электромагнетизм). Вычислительные пакеты STAR-CD (<http://www.cd.co.uk>), PHOENICS (<http://www.cham.co.uk>, <http://213.210.25.174/website/new/top.htm>) специально ориентированы на компьютерное моделирование в области динамики жидкости и тепломассообмена.

STAR-CD, PHOENICS, ANSYS принадлежат к числу так называемых «тяжелых» пакетов. Они велики по объему, оснащены эффективными численными алгоритмами, удобным интерфейсом, мощными графическими средствами для формирования сложной геометрии объектов и для визуализации результатов. Самое главное, практически отсутствуют ограничения на сложность задач, за исключением памяти и быстродействия компьютера.

Однако работа с такими программами требует специальной подготовки, а сами пакеты являются дорогостоящими коммерческими продуктами. Нашей задачей является изучение основных идей компьютерного моделирования, и здесь более эффективны простые программы с открытым кодом, доступным для изучения и модификации.

