Семинар 9. Решение одномерного уравнения теплопроводности.

1. Постановка задачи.

$$\frac{\partial u}{\partial t} = \frac{\partial}{\partial x}\left(k(x)\frac{\partial u}{\partial x} \right) + f(x,t), \quad a < x < b,$$

$$u(x,0) = g_0(x), \quad u(a) = g_1(t), \quad u(b) = g_2(t),$$

$$k(x) = \begin{cases} k_1, & a \le x < x_k, \\ k_2, & x_k \le x \le b, \end{cases} \quad f(x,t) = Q_0 \exp\left[-(x-x_0)^2 / r_0^2\right]\left(1-\exp\left[-t/\tau_0\right]\right),$$

$$g_0(x) = u_0, \quad g_1(t) = u_0 + (u_1 - u_0)\left(1 - \exp\left[-t/\tau_1\right]\right), \quad g_2(t) = u_0.$$

2. Численный алгоритм.

Равномерная сетка $\Omega = \omega_x \times \omega_t$, $\omega_x = \left\{ x_i = a + ih_x, i = 0,\dots,N_x, h_x = \dfrac{b-a}{N_x} \right\}$, $\omega_t = \left\{ t_j = j\tau,\ j = 0,\dots,N_t, \tau = \dfrac{t_{\max}}{N_t} \right\}$.

Схема с весами:

$$\frac{y_i^{j+1} - y_i^j}{\tau} = \sigma\left[\frac{1}{\hbar_x}\left\{ k_{i+1/2}\frac{y_{i+1}^{j+1} - y_i^{j+1}}{h_x} - k_{i-1/2}\frac{y_i^{j+1} - y_{i-1}^{j+1}}{h_x} \right\} + f_i^{j+1} \right] +$$

$$+(1-\sigma)\left[\frac{1}{\hbar_x}\left\{ k_{i+1/2}\frac{y_{i+1}^{j} - y_i^{j}}{h_x} - k_{i-1/2}\frac{y_i^{j} - y_{i-1}^{j}}{h_x} \right\} + f_i^{j} \right], \quad 0 < i < N_x, \quad 0 \le j < N_t,$$

$$y_i^0 = g_0(x_i), \quad y_0^j = g_1(t_j), \quad y_{N_x}^j = g_2(t_j),$$

$$k_{i\pm1/2} = \frac{2k_i k_{i\pm1}}{k_i + k_{i\pm1}}, \quad k_i = k(x_i), \quad f_i^j = f(x_i, t_j), \quad \hbar_x = \begin{cases} 0.5h_x, & i = 0, N_x, \\ h_x, & 1 < i < N_x. \end{cases}$$

Выбор параметров сеток:

1) $h_x \sim \dfrac{r_0}{m}$, $m = 1,2,3,\dots$; 2) $t_{\max} \sim \max(\tau_0, \tau_1)\cdot M$, $M = 5,10,15,\dots$;

3) если имеется стационар, то величина $g_t = \max\limits_{0\le i\le N_x}\left| \dfrac{y_i^{j+1} - y_i^j}{\tau y_i^j} \right|$ уменьшается и можно поставить условие $g_t \le \varepsilon_t$;

4) $\tau \le \min\left(\tau^{(\sigma)}, \tau^{(f)}\right)$ – условия устойчивости для схемы с весами и ОДУ:

$$\tau^{(\sigma=0)} = \frac{0.5h_x^2}{\max(k_1,k_2)}, \ \tau^{(\sigma=0.5)} = \frac{h_x^2}{\max(k_1,k_2)}, \ \tau^{(\sigma=1)} = \frac{0.5h_x}{\sqrt{\max(k_1,k_2)}}, \ \tau^{(f)} = \frac{1}{\max\left|f(x,t)\right| L} = \frac{1}{Q_0 L}, \quad L = 1,5,10,\dots .$$

3. Параллельная реализация.

Разбиение пространственной сетки на равные интервалы. Линейная топология обменов.

Расчетные формулы явной схемы:

$$y_i^{j+1} = y_i^j + \gamma\left\{ \frac{2k_i k_{i+1}}{k_i + k_{i+1}}\left(y_{i+1}^j - y_i^j\right) - \frac{2k_i k_{i-1}}{k_i + k_{i-1}}\left(y_i^j - y_{i-1}^j\right) \right\} + \tau f_i^j, \quad 0 < i < N_x,$$

$$y_0^{j+1} = g_0(t_{j+1}), \quad y_{N_x}^{j+1} = g_1(t_{j+1}), \quad \gamma = \frac{\tau}{\hbar_x h_x}.$$

Уравнения неявной схемы:

$$y_i^{j+1} - \gamma\left\{ \frac{2k_i k_{i+1}}{k_i + k_{i+1}}\left(y_{i+1}^{j+1} - y_i^{j+1}\right) - \frac{2k_i k_{i-1}}{k_i + k_{i-1}}\left(y_i^{j+1} - y_{i-1}^{j+1}\right) \right\} = y_i^j + \tau f_i^{j+1}, \quad 0 < i < N_x,$$

$$y_0^{j+1} = g_0(t_{j+1}), \quad y_{N_x}^{j+1} = g_1(t_{j+1}).$$

Симметричная схема:

$$y_i^{j+1} - \frac{\gamma}{2}\left\{ \frac{2k_i k_{i+1}}{k_i + k_{i+1}}\left(y_{i+1}^{j+1} - y_i^{j+1}\right) - \frac{2k_i k_{i-1}}{k_i + k_{i-1}}\left(y_i^{j+1} - y_{i-1}^{j+1}\right) \right\} =$$

$$= y_i^j + \frac{\gamma}{2}\left\{ \frac{2k_i k_{i+1}}{k_i + k_{i+1}}\left(y_{i+1}^{j} - y_i^{j}\right) - \frac{2k_i k_{i-1}}{k_i + k_{i-1}}\left(y_i^{j} - y_{i-1}^{j}\right) \right\} + \frac{\tau}{2}\left(f_i^{j+1} + f_i^j\right), \quad 0 < i < N_x,$$

$$y_0^{j+1} = g_0(t_{j+1}), \quad y_{N_x}^{j+1} = g_1(t_{j+1}).$$

4. Тестирование программы.

Использование вложенных сеток и стационарного решения.

Проводим расчеты для $h_x = h, \dfrac{h}{2}, \dfrac{h}{4}$. Погрешность $O\left(\tau^\alpha + h_x^2\right)$ в стационаре стремится к величине Ch_x^2.

В итоге: $\left\|y_h^\infty - u_h^\infty\right\| \le Ch^2$, $\left\|y_{h/2}^\infty - u_{h/2}^\infty\right\| \le C\dfrac{h^2}{4}$, $\left\|y_{h/4}^\infty - u_{h/4}^\infty\right\| \le C\dfrac{h^2}{16}$.

$\Delta_1 = \left\| y_h^\infty - y_{h/2}^\infty \right\| \le \left\| y_h^\infty - u_h^\infty \right\| + \left\| y_{h/2}^\infty - u_{h/2}^\infty \right\| \le Ch^2 \left(1 + \frac{1}{4}\right)$, $\Delta_2 = \left\| y_{h/2}^\infty - y_{h/4}^\infty \right\| \le \left\| y_{h/2}^\infty - u_{h/2}^\infty \right\| + \left\| y_{h/4}^\infty - u_{h/4}^\infty \right\| \le Ch^2 \frac{1}{4}\left(1 + \frac{1}{4}\right)$,

$\log_2 \Delta = \log_2 \left(\Delta_1 / \Delta_2\right) \approx 2$. Таким образом, проверяется порядок сходимости схемы по пространственной переменной. При этом результат (стационарное решение) не должен зависеть от веса схемы σ.

5. Реализация примеров.

Пример 1. Решение задачи по явной схеме (ex13a.c):

```
#include <stdio.h>  #include <stdlib.h>  #include <string.h>  #include <math.h>
#include "mycom.h"  #include "mynet.h"  #include "myio.h"
static int np, mp, nl, ier, lp;  static int mp_l, mp_r;
static char pname[MPI_MAX_PROCESSOR_NAME];  static char vname[10] = "ex13a";
static char sname[20];  static MPI_Status status;  static union_t buf;
static double tick, t1, t2, t3;  static FILE *Fi = NULL;  static FILE *Fo = NULL;
static int nx, ntp, ntm, ntv;  static double xa, xb, xk, x0, r0, q0, u0, u1;
static double k1, k2, tau0, tau1, tmax, epst;  static double tv, u10, omg0, omg1, gt;
double k(double x); double k(double x) {if (x<xk) return k1; else return k2; }
double f(double x, double t); double f(double x, double t) {
  double s1 = (x-x0) / r0; double s2 = omg0 * t; return q0*exp(-s1*s1)*(1.0-exp(-s2)); }
double g0(double x); double g0(double x) { return u0; }
double g1(double t); double g1(double t) { double s1=omg1*t; return u0+u10*(1.0-exp(-
s1)); }
double g2(double t); double g2(double t) { return u0; }
int main(int argc, char *argv[])
{
  int i, j, ii, i1, i2, nc, ncm;
  double hx, hx2, tau, gam, s0, s1, s2, s3, y0m, y0p;
  double *xx, *aa, *bb, *y0, *y1;
  MyNetInit(&argc,&argv,&np,&mp,&nl,pname,&tick);
  fprintf(stderr,"Netsize: %d, process: %d, system: %s, tick=%12le\n", np,mp,pname,tick);
  sleep(1); sprintf(sname,"%s.p%02d",vname,mp);
  ier = fopen_m(&Fo,sname,"wt"); if (ier!=0) mpierr("Protocol file not opened",1);
  if (mp==0) {
    sprintf(sname,"%s.d",vname);
    ier = fopen_m(&Fi,sname,"rt");
    if (ier!=0) mpierr("Data file not opened",2);
    fscanf(Fi,"xa=%le\n",&xa); fscanf(Fi,"xb=%le\n",&xb); fscanf(Fi,"xk=%le\n",&xk);
    fscanf(Fi,"x0=%le\n",&x0); fscanf(Fi,"r0=%le\n",&r0); fscanf(Fi,"q0=%le\n",&q0);
    fscanf(Fi,"u0=%le\n",&u0); fscanf(Fi,"u1=%le\n",&u1); fscanf(Fi,"k1=%le\n",&k1);
    fscanf(Fi,"k2=%le\n",&k2); fscanf(Fi,"tau0=%le\n",&tau0);
    fscanf(Fi,"tau1=%le\n",&tau1); fscanf(Fi,"tmax=%le\n",&tmax);
    fscanf(Fi,"epst=%le\n",&epst); fscanf(Fi,"nx=%d\n",&nx);
    fscanf(Fi,"ntp=%d\n",&ntp); fscanf(Fi,"ntm=%d\n",&ntm); fscanf(Fi,"lp=%d\n",&lp);
    fclose_m(&Fi);
    if (argc>1) sscanf(argv[1],"%d",&nx); if (argc>2) sscanf(argv[2],"%d",&ntp);
    if (argc>3) sscanf(argv[3],"%d",&ntm);
  }
  if (np>1) {
    if (mp==0) {
      buf.ddata[0] = xa; buf.ddata[1] = xb; buf.ddata[2] = xk; buf.ddata[3] = x0;
      buf.ddata[4] = r0; buf.ddata[5] = q0; buf.ddata[6] = u0; buf.ddata[7] = u1;
      buf.ddata[8] = k1; buf.ddata[9] = k2; buf.ddata[10] = tau0; buf.ddata[11] = tau1;
      buf.ddata[12] = tmax; buf.ddata[13] = epst; buf.idata[28] = nx;
      buf.idata[29] = ntp; buf.idata[30] = ntm; buf.idata[31] = lp;
    }
    MPI_Bcast(buf.ddata,16,MPI_DOUBLE,0,MPI_COMM_WORLD);
    if (mp>0) {
      xa = buf.ddata[0]; xb = buf.ddata[1]; xk = buf.ddata[2]; x0 = buf.ddata[3];
      r0   = buf.ddata[4]; q0 = buf.ddata[5]; u0 = buf.ddata[6]; u1 = buf.ddata[7];
      k1   = buf.ddata[8]; k2 = buf.ddata[9]; tau0 = buf.ddata[10]; tau1 = buf.ddata[11];
      tmax = buf.ddata[12]; epst = buf.ddata[13]; nx = buf.idata[28];
      ntp = buf.idata[29]; ntm  = buf.idata[30]; lp = buf.idata[31];
    }
  }
  fprintf(Fo,"Netsize: %d, process: %d, system: %s, tick=%12le\n", np,mp,pname,tick);
  fprintf(Fo,"xa=%le xb=%le xk=%le x0=%le r0=%le\n",xa,xb,xk,x0,r0);
  fprintf(Fo,"q0=%le u0=%le u1=%le k1=%le k2=%le\n",q0,u0,u1,k1,k2);
  fprintf(Fo,"tau0=%le tau1=%le tmax=%le epst=%le\n",tau0,tau1,tmax,epst);
```

```
fprintf(Fo,"nx=%d ntp=%d ntm=%d lp=%d\n",nx,ntp,ntm,lp);
t1 = MPI_Wtime();
u10 = u1 - u0; omg0 = 1.0 / tau0; omg1 = 1.0 / tau1;
hx = (xb-xa)/nx; hx2 = hx * hx;
tau = 0.5 * hx2 / dmax(k1,k2); tau = dmin(tau,1.0/q0); gam = tau / hx2;
s0 = dmin(tmax/tau,1000000000.0); ntm = imin(ntm,(int)s0);
fprintf(Fo,"u10=%le omg0=%le omg1=%le\n",u10,omg0,omg1);
fprintf(Fo,"hx=%le tau=%le ntm=%d\n",hx,tau,ntm);
if (mp == 0) fprintf(stderr,"nx=%d hx=%le tau=%le ntm=%d\n",nx,hx,tau,ntm);
if (mp ==    0) mp_l = -1; else mp_l = mp - 1;
if (mp == np-1) mp_r = -1; else mp_r = mp + 1;
MyRange(np,mp,0,nx,&i1,&i2,&nc); ncm = nc-1;
fprintf(Fo,"i1=%d i2=%d nc=%d\n",i1,i2,nc);
xx = (double*)(malloc(sizeof(double)*nc)); y0 = (double*)(malloc(sizeof(double)*nc));
y1 = (double*)(malloc(sizeof(double)*nc)); aa = (double*)(malloc(sizeof(double)*nc));
bb = (double*)(malloc(sizeof(double)*nc));
for (i=0; i<nc; i++) xx[i] = xa + hx * (i1 + i); // grid
for (i=0; i<nc; i++) {
  ii = i1 + i;
  if ((ii==0) || (ii==nx)) { aa[i] = 0.0; bb[i] = 0.0; }
  else { s0 = k(xx[i]); s1 = k(xx[i]-hx); s2 = k(xx[i]+hx);
    aa[i] = gam * 2.0 * s0 * s1 / (s0 + s1); bb[i] = gam * 2.0 * s0 * s2 / (s0 + s2);
  }
}
ntv = 0; tv = 0.0; gt = 1.0;
for (i=0; i<nc; i++) y1[i] = g0(xx[i]); // initial profile
do {
  ntv++; tv += tau;
  for (i=0; i<nc; i++) y0[i] = y1[i];
  if (np>1) BndAExch1D(mp_l,1,y0+0,&y0m,mp_r,1,y0+ncm,&y0p);
  else { y0m = 0.0; y0p = 0.0; }
  for (i=0; i<nc; i++) {
    ii = i1 + i;
    if (ii==0)       y1[i] = g1(tv);
    else if (ii==nx) y1[i] = g2(tv);
    else {
      if (i==0) s1 = aa[i] * (y0[i] - y0m);
      else      s1 = aa[i] * (y0[i] - y0[i-1]);
      if (i==ncm) s2 = bb[i] * (y0p - y0[i]);
      else        s2 = bb[i] * (y0[i+1] - y0[i]);
      s3 = tau * f(xx[i],tv-tau);
      y1[i] += (s2 - s1 + s3);
    }
  }
  if (ntv % ntp == 0) {
    gt = 0.0;
    for (i=0; i<nc; i++) { s0 = (y1[i]/y0[i]-1.0); gt = dmax(gt,dabs(s0)); }
    gt = gt / tau;
    if (np>1) { s0 = gt; MPI_Allreduce(&s0,&gt,1,MPI_DOUBLE,MPI_MAX,MPI_COMM_WORLD); }
    if (mp == 0) {
      t2 = MPI_Wtime() - t1;
      fprintf(stderr,"ntv=%d tv=%le gt=%le tcpu=%le\n",ntv,tv,gt,t2);
    }
  }
  if (lp>0) { fprintf(Fo,"ntv=%d tv=%le gt=%le\n",ntv,tv,gt);
    for (i=0; i<nc; i++) fprintf(Fo,"i=%8d x=%12le y1=%12le\n",(i1+i),xx[i],y1[i]);
  }
} while ((ntv<ntm) && (gt>epst));
t1 = MPI_Wtime() - t1;
sprintf(sname,"%s_%02d.dat",vname,np);
OutFun1DP(sname,np,mp,nc,xx,y1);
fprintf(Fo,"ntv=%d tv=%le gt=%le time=%le\n",ntv,tv,gt,t1);
if (mp == 0) fprintf(stderr,"ntv=%d tv=%le gt=%le tcpu=%le\n",ntv,tv,gt,t1);
ier = fclose_m(&Fo);
MPI_Finalize(); return 0; }
```

Пример 2. Решение задачи по неявной схеме (ex13b.c), отличие в расчетном цикле:

```
  do {
    ntv++; tv += tau;
    for (i=0; i<nc; i++) y0[i] = y1[i];
    for (i=0; i<nc; i++) {
      ii = i1 + i;
      if (ii==0)        ff[i] = g1(tv);
      else if (ii==nx)  ff[i] = g2(tv);
      else              ff[i] = y0[i] + tau * f(xx[i],tv);
    }
    if (np<2) ier = prog_right(nc,aa,bb,cc,ff,al,y1);
    else      ier = prog_rightpm(np,mp,nc,ntv,aa,bb,cc,ff,al,y1,y2,y3,y4);
    if (ier!=0) mpierr("Bad solution",1);
    ***
  } while ((ntv<ntm) && (gt>epst));
```

Трансляция:

```
>mpicc -o ex13a.px -O2 ex13a.c mycom.c mynet.c myio.c -lm
>mpicc -o ex13b.px -O2 ex13b.c mycom.c mynet.c myio.c myprog.c -lm
```

Результаты расчетов:

```
>mpirun -np 1 -nolocal -machinefile hosts ex13a.px   (запуск в классе)
>mpirun -np 1 ex13a.px                               (запуск на сервере)
nx=200 hx=5.000000e-03 tau=1.250000e-06 ntm=1000000
ntv=50000 tv=6.250000e-02 gt=2.608438e+00 tcpu=7.621830e-01
ntv=100000 tv=1.250000e-01 gt=1.819795e-01 tcpu=1.525041e+00
ntv=150000 tv=1.875000e-01 gt=1.287988e-02 tcpu=2.287851e+00
ntv=200000 tv=2.500000e-01 gt=9.140317e-04 tcpu=2.872474e+00
ntv=250000 tv=3.125000e-01 gt=6.488034e-05 tcpu=3.444287e+00
ntv=300000 tv=3.750000e-01 gt=4.605373e-06 tcpu=4.016293e+00
ntv=350000 tv=4.375000e-01 gt=3.269213e-07 tcpu=4.588261e+00
ntv=400000 tv=5.000000e-01 gt=2.354150e-08 tcpu=5.160074e+00
>mpirun -np 1 -nolocal -machinefile hosts ex13b.px   (запуск в классе)
>mpirun -np 1 ex13b.px                               (запуск на сервере)
nx=200 hx=5.000000e-03 tau=7.905694e-04 ntm=2529
ntv=100 tv=7.905694e-02 gt=1.324355e+00 tcpu=2.376000e-03
ntv=200 tv=1.581139e-01 gt=4.849999e-02 tcpu=4.717000e-03
ntv=300 tv=2.371708e-01 gt=1.802814e-03 tcpu=7.061000e-03
ntv=400 tv=3.162278e-01 gt=6.707688e-05 tcpu=9.401000e-03
ntv=500 tv=3.952847e-01 gt=2.495812e-06 tcpu=1.172200e-02
ntv=600 tv=4.743416e-01 gt=9.286492e-08 tcpu=1.404600e-02
```

Расчеты на эффективность:

```
>mpirun -np <1-12> -nolocal -machinefile hosts ex13a.px 10000 2000 10000  (в классе)
>mpirun -np <1-12> ex13a.px 10000 2000 10000                              (на сервере)
np=1  nx=10000 ntv=10000 tv=5.000000e-06 gt=9.990008e+01 tcpu=1.977835e+01
np=2  nx=10000 ntv=10000 tv=5.000000e-06 gt=9.990008e+01 tcpu=1.249927e+01
np=3  nx=10000 ntv=10000 tv=5.000000e-06 gt=9.990008e+01 tcpu=9.079243e+00
np=4  nx=10000 ntv=10000 tv=5.000000e-06 gt=9.990008e+01 tcpu=7.574098e+00
np=5  nx=10000 ntv=10000 tv=5.000000e-06 gt=9.990008e+01 tcpu=6.698560e+00
np=6  nx=10000 ntv=10000 tv=5.000000e-06 gt=9.990008e+01 tcpu=5.972717e+00
np=7  nx=10000 ntv=10000 tv=5.000000e-06 gt=9.990008e+01 tcpu=5.401055e+00
np=8  nx=10000 ntv=10000 tv=5.000000e-06 gt=9.990008e+01 tcpu=4.887469e+00
np=9  nx=10000 ntv=10000 tv=5.000000e-06 gt=9.990008e+01 tcpu=4.596798e+00
np=10 nx=10000 ntv=10000 tv=5.000000e-06 gt=9.990008e+01 tcpu=4.400385e+00
np=11 nx=10000 ntv=10000 tv=5.000000e-06 gt=9.990008e+01 tcpu=4.325092e+00
np=12 nx=10000 ntv=10000 tv=5.000000e-06 gt=9.990008e+01 tcpu=4.053824e+00
>mpirun -np <1-12> -nolocal -machinefile hosts ex13b.px 10000 2000 10000  (в классе)
>mpirun -np <1-12> ex13b.px 10000 2000 10000                              (на сервере)
np=1  nx=10000 ntv=10000 tv=1.581139e-01 gt=3.932917e-02 tcpu=2.308190e+01
np=2  nx=10000 ntv=10000 tv=1.581139e-01 gt=3.932917e-02 tcpu=1.442830e+01
np=3  nx=10000 ntv=10000 tv=1.581139e-01 gt=3.932917e-02 tcpu=1.231996e+01
np=4  nx=10000 ntv=10000 tv=1.581139e-01 gt=3.932917e-02 tcpu=1.032698e+01
np=5  nx=10000 ntv=10000 tv=1.581139e-01 gt=3.932917e-02 tcpu=1.147053e+01
np=6  nx=10000 ntv=10000 tv=1.581139e-01 gt=3.932917e-02 tcpu=1.144447e+01
np=7  nx=10000 ntv=10000 tv=1.581139e-01 gt=3.932917e-02 tcpu=1.367407e+01
np=8  nx=10000 ntv=10000 tv=1.581139e-01 gt=3.932917e-02 tcpu=1.132352e+01
np=9  nx=10000 ntv=10000 tv=1.581139e-01 gt=3.932917e-02 tcpu=1.392584e+01
np=10 nx=10000 ntv=10000 tv=1.581139e-01 gt=3.932917e-02 tcpu=1.387531e+01
np=11 nx=10000 ntv=10000 tv=1.581139e-01 gt=3.932917e-02 tcpu=1.445106e+01
np=12 nx=10000 ntv=10000 tv=1.581139e-01 gt=3.932917e-02 tcpu=1.452619e+01
```