1 Inledning

Vi skall se på numerisk lösning av partiella differentialekvationer (PDE). För stationära problem kommer vi använda differensmetoder och för tidsberoende problem den s.k. linjemetoden som leder till system av ordinära differentialekvationer, som vi sedan löser med en ODE-lösare.

Vi såg tidigare att temperaturen $u(x)$ i en isolerad stav beskrevs av randvärdesproblemet

$$\begin{cases} -\kappa u'' = f, \ 0 \le x \le L \\ u(0) = g_0, \ u(L) = g_L \end{cases}$$

där g_0 och g_L är temperaturen i stavens ändpunkter, κ är stavens termiska diffusivitet och $f(x)$ är en värmekälla.

Nu skall vi se på motsvarande problem för en isolerad platta. Temperaturen $u(x, y)$ beskrivs av följande randvärdesproblem för partiell differentialekvation

$$\begin{cases} -\kappa\,(u''_{xx} + u''_{yy}) = f, \ \text{i } \Omega \\ u = g, \ \text{på } \partial\Omega \end{cases}$$

där g ger temperaturen längs plattans kanter, κ är plattans termiska diffusivitet, Ω är området $0 \le x \le L, 0 \le y \le L$ och $\partial\Omega$ dess rand.

Vi skall också se på hur temperaturen hos staven förändras med tiden, vilket beskrivs av följande partiella differentialekvation med både rand- och begynnelsevillkor

$$\begin{cases} u'_t - \kappa u''_{xx} = f, \ 0 \le x \le L, \ t > 0 \\ u(0, t) = g_0, \ u(L, t) = g_L, \ t > 0 \\ u(x, 0) = u_0(x), \ 0 \le x \le L \end{cases}$$

Här är $u_0(x)$ begynnelsetemperaturen i staven och vi kommer se hur temperaturen $u(x, t)$ utvecklas mot det stationära tillståndet.

Givetvis skall vi även se på motsvarande tidsutveckling för plattan med temperaturen $u(x, y, t)$ som lösning till

$$\begin{cases} u'_t - \kappa\,(u''_{xx} + u''_{yy}) = f, \ \text{i } \Omega, \ t > 0 \\ u = g, \ \text{på } \partial\Omega, \ t > 0 \\ u(x, y, 0) = u_0(x, y), \ \text{i } \Omega \end{cases}$$

Avslutningsvis skall vi kort se på egenvärdesproblem för partiella differentialekvationer.

Men allra först ser vi lite på differensapproximationer av partiella derivator. T.ex. $u'_x(x, y)$ kan approximeras av *framåt*-, *bakåt*- och *centraldifferenskvoterna*

$$u'_x(x, y) \approx \frac{u(x+h, y) - u(x, y)}{h}, \quad u'_x(x, y) \approx \frac{u(x, y) - u(x-h, y)}{h}$$

$$u'_x(x, y) \approx \frac{u(x+h, y) - u(x-h, y)}{2h}$$

där h som ett litet positivt tal.

På motsvarande sätt approximerar vi $u'_y(x,y)$ med differenskvoterna

$$u'_y(x,y) \approx \frac{u(x,y+k)-u(x,y)}{k}, \quad u'_y(x,y) \approx \frac{u(x,y)-u(x,y-k)}{k}$$

$$u'_y(x,y) \approx \frac{u(x,y+k)-u(x,y-k)}{2k}$$

där k som ett litet positivt tal.

För andraderivatorna $u''_{xx}(x,y)$, $u''_{yy}(x,y)$ och $u''_{xy}(x,y)$ använder vi centraldifferenskvoterna

$$u''_{xx}(x,y) \approx \frac{u(x+h,y)-2u(x,y)+u(x-h,y)}{h^2}$$

$$u''_{yy}(x,y) \approx \frac{u(x,y+k)-2u(x,y)+u(x,y-k)}{k^2}$$

$$u''_{xy}(x,y) \approx \frac{u(x+h,y+k)-u(x+h,y-k)-u(x-h,y+k)+u(x-h,y-k)}{4hk}$$

där h och k är små positiva tal.

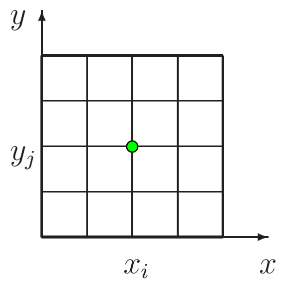
2 Stationära problem

Vi ser på följande randvärdeproblem för partiell differentialekvation

$$\begin{cases} -(u''_{xx}+u''_{yy}) = f, \text{ i } \Omega \\ u = g, \text{ på } \partial\Omega \end{cases}$$

där f och g är givna funktioner, Ω enhetskvadraten och $\partial\Omega$ dess rand.

Vi skall lösa problemet med differensmetod, dvs. ersätta derivator med differenskvoter över ett beräkningsnät eller gitter (mesh).



Vi inför ett likformigt nät på området:

$$x_i = ih,\ i = 0,\cdots,n+1, \quad y_j = jh,\ j = 0,\cdots,n+1, \quad h = \tfrac{1}{n+1}$$

Vi ersätter $-(u''_{xx}+u''_{yy})$ i differentialekvationen i punkten (x_i,y_j) med approximationen

$$-(u''_{xx}(x_i,y_j)+u''_{yy}(x_i,y_j)) \approx$$

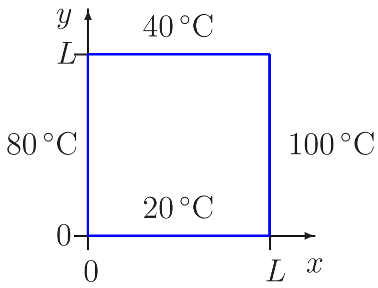
$$\approx \frac{1}{h^2}\left(-u(x_{i+1},y_j)-u(x_i,y_{j+1})+4u(x_i,y_j)-u(x_{i-1},y_j)-u(x_i,y_{j-1})\right)$$

Om vi låter $u_{i,j}$ beteckna approximationen av $u(x,y)$ i punkten (x_i,y_j) får vi ekvationerna

$$4\,u_{i,j}-u_{i+1,j}-u_{i-1,j}-u_{i,j+1}-u_{i,j-1} = h^2 f(x_i,y_j)$$

När vi träffar på randen plockar vi över dessa kända randvärden till högerledet. Vi kommer slutligen fram till ett linjärt ekvationssystem som vi löser.

Uppgift 1. Vi skall se på stationär värmeledning i en platta där randtemperaturerna ges av figuren. I det inre av plattan har vi även en värmekälla.



Temperaturen $u(x, y)$ beskrivs av följande randvärdesproblem för partiell differentialekvation

$$\begin{cases} -\kappa\,(u''_{xx} + u''_{yy}) = f, \text{ i } \Omega \\ u = g, \text{ på } \partial\Omega \end{cases}$$

där g ger temperaturen längs plattans kanter, κ är plattans termiska diffusivitet, Ω är området $0 \le x \le L, 0 \le y \le L$ och $\partial\Omega$ dess rand.

Tag randtemperaturer enligt figuren, $\kappa = 0.5$, $L = 1$ och värmekällan

$$f(x, y) = k(x-1)^2(y-1)^2, \quad k = 2500$$

Ersätt derivatorna med differenskvoter över ett beräkningsnät. Sätt upp det linjära ekvationssystem $\boldsymbol{A}\boldsymbol{u} = \boldsymbol{b}$ som beskriver temperaturen i de olika nodpunkterna på nätet. Matrisen blir gles så använd `spdiags`. Tag $n = 20$ och lös ekvationssystemet. Rita nivåkurvor, med `reshape` samt `contourf` och `colorbar`, som visar lika temperatur (isotermer), och funktionsyta till $u(x, y)$ med `surf`. (Börja med att ta $n = 3$.)

3 Tidsberoende problem

Vi ser på den s.k. värmelednings- eller diffusionsekvationen med både rand- och begynnelsevärden

$$\begin{cases} u'_t - u''_{xx} = 0, \; 0 < x < 1, \; 0 < t < T \\ u(0, t) = g_0(t), \; u(1, t) = g_1(t), \; 0 < t < T \\ u(x, 0) = u_{beg}(x), \; 0 < x < 1 \end{cases}$$

Inför nät med steglängden h i rumsled och ersätt u''_{xx} med D_+D_-u.

Låt $u_i(t)$ beteckna approximationen av $u(x_i, t)$.

Vi får följande begynnelsevärdesproblem för ODE-system

$$\begin{cases} u'_i(t) - \frac{u_{i+1}(t) - 2u_i(t) + u_{i-1}(t)}{h^2} = 0, \; i = 1, \cdots, n, \; 0 < t < T \\ u_0(t) = g_0(t), \; u_{n+1}(t) = g_1(t) \\ u_i(0) = u_{beg}(x_i), \; i = 1, \cdots, n \end{cases}$$

Med matrisbeteckningar kan detta skrivas

$$\begin{cases} \boldsymbol{U}'(t) = \boldsymbol{F}(t, \boldsymbol{U}), \; 0 < t < T \\ \boldsymbol{U}(0) = \boldsymbol{U}_0 \end{cases}$$

där $\boldsymbol{F}(t,\boldsymbol{U}) = \frac{1}{h^2}(\boldsymbol{b}(t) - \boldsymbol{A}\boldsymbol{U}(t))$, $\boldsymbol{A}$ är matrisen som motsvarar $-u''$ och

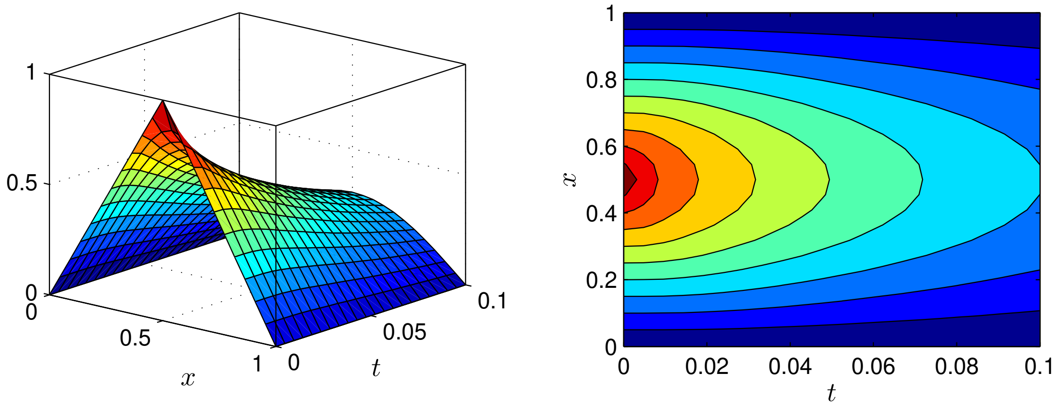
$$\boldsymbol{U}(t) = \begin{bmatrix} u_1(t) \\ u_2(t) \\ \vdots \\ u_{n-1}(t) \\ u_n(t) \end{bmatrix} \quad \boldsymbol{b}(t) = \begin{bmatrix} g_0(t) \\ 0 \\ \vdots \\ 0 \\ g_1(t) \end{bmatrix} \quad \boldsymbol{U}_0 = \begin{bmatrix} u_{beg}(x_1) \\ u_{beg}(x_2) \\ \vdots \\ u_{beg}(x_{n-1}) \\ u_{beg}(x_n) \end{bmatrix}$$

Vi tar t.ex. följande rand- och begynnelsevillkor

$$g_0 = g_1 = 0, \qquad u_{beg}(x) = \begin{cases} 2x, \ 0 \le x \le 0.5 \\ 2-2x, \ 0.5 < x \le 1 \end{cases}$$

och beräknar lösningen med linjemetoden i MATLAB enligt

```
>> T=0.1; n=21; h=1/(n+1); xi=h*[1:n]'; tspan=linspace(0,T,n+1);
>> g0=@(t)zeros(size(t)); g1=@(t)zeros(size(t));
>> u_beg=@(x)1-abs(2*(x-0.5)); U0=u_beg(xi);
>> A=spdiags(ones(n,1)*[-1 2 -1],[-1 0 1],n,n);
>> b=@(t)[g0(t);zeros(n-2,1);g1(t)];
>> F=@(t,U)(b(t)-A*U)/h^2;
>> [t,U]=ode45(F,tspan,U0);
>> x=[0;xi;1]; U=[g0(t),U,g1(t)];   % Kantar lösningen med randvärdena
>> surf(x,t,U), xlabel('x'), ylabel('t')
```



Uppgift 2. Vi skall se på det *tidsberoende* värmeledningsproblemet för staven, som beskrivs av

$$\begin{cases} u'_t - \kappa u''_{xx} = f, \ 0 \le x \le L, \ t > 0 \\ u(0,t) = g_0, \ u(L,t) = g_L, \ t > 0 \\ u(x,0) = u_{beg}(x), \ 0 \le x \le L \end{cases}$$

Här är $u_0(x)$ begynnelsetemperaturen i staven och vi kommer se hur temperaturen $u(x,t)$ utvecklas mot det stationära tillståndet.

Låt $\kappa = 2$, $L = 1$, $g_0 = 40$, $g_L = 60$ samt $f(x) = 200\exp(-(x - \frac{L}{2})^2)$. Tag tidsintervallet $0 \le t \le 0.7$ och begynnelsetemperaturen $u_{beg}(x) = 5$. Använd linjemetoden. Lös ODE-systemet med `ode45` och gör en visualisering av tidsförloppet med `contourf` och `surf`. Tag $n = 30$. Jämför med laboration 2, uppgift 7.

Uppgift 3. Det *tidsberoende* värmeledningsproblemet för plattan, beskrivs av

$$\begin{cases} u'_t - \kappa\,(u''_{xx} + u''_{yy}) = f, \text{ i } \Omega, \ t > 0 \\ u = g, \text{ på } \partial\Omega, \ t > 0 \\ u(x,y,0) = u_{beg}(x,y), \text{ i } \Omega \end{cases}$$

där vi har samma område och randvillkor som i uppgift 1.

Lös problemet med linjemetoden, som består i att vi göra samma diskretisering i xy-planet men behåller tidsberoendet. Tag tidsintervallet $0 \le t \le 0.7$ och begynnelsetemperaturen $u_{beg}(x,y) = 5$. Lös ODE-systemet med `ode45` och gör en visualisering av tidsförloppet med `contourf` och `surf`.

Som ytterligare ett exempel på ett tidsberoende problem ser i på den s.k. vågekvationen

$$\begin{cases} u''_{tt} - c^2 u''_{xx} = 0, \ 0 < x < 1, 0 \le t \le T \\ u(0,t) = g_0(t), \ u(1,t) = g_1(t), \ 0 \le t \le T \\ u(x,0) = u_0(x), \ u'_t(x,0) = v_0(x), \ 0 < x < 1 \end{cases}$$

Inför nät med steglängden h i rumsled och ersätt u''_{xx} med D_+D_-u. Låt $u_i(t)$ beteckna approximationen av $u(x_i,t)$.

Får begynnelsevärdesproblem för ODE-system.

$$\begin{cases} u''_i(t) - c^2 \frac{u_{i+1}(t) - 2u_i(t) + u_{i-1}(t)}{h^2} = 0, i = 1, \cdots, n, 0 < t < T \\ u_0(t) = g_0(t), u_{n+1}(t) = g_1(t) \\ u_i(0) = u_0(x_i), i = 1, \cdots, n \\ u'_i(0) = v_0(x_i), i = 1, \cdots, n \end{cases}$$

$$\begin{cases} \boldsymbol{U}''(t) = \frac{c^2}{h^2}(\boldsymbol{b}(t) - \boldsymbol{A}\boldsymbol{U}(t)), 0 < t < T \\ \boldsymbol{U}(0) = \boldsymbol{U}_0, \ \boldsymbol{U}'(0) = \boldsymbol{V}_0 \end{cases}$$

där $\boldsymbol{A}$ är matrisen som motsvarar $-u''$ och

$$\boldsymbol{U}(t) = \begin{bmatrix} u_1(t) \\ u_2(t) \\ \vdots \\ u_{n-1}(t) \\ u_n(t) \end{bmatrix} \quad \boldsymbol{b}(t) = \begin{bmatrix} g_0(t) \\ 0 \\ \vdots \\ 0 \\ g_1(t) \end{bmatrix} \quad \boldsymbol{U}_0 = \begin{bmatrix} u_0(x_1) \\ u_0(x_2) \\ \vdots \\ u_0(x_{n-1}) \\ u_0(x_n) \end{bmatrix} \quad \boldsymbol{V}_0 = \begin{bmatrix} v_0(x_1) \\ v_0(x_2) \\ \vdots \\ v_0(x_{n-1}) \\ v_0(x_n) \end{bmatrix}$$

Slutligen får vi formulera om vår ODE som ett första ordningens system

$$\begin{cases} \boldsymbol{W}'(t) = \boldsymbol{f}(t, \boldsymbol{W}), \ 0 < t < T \\ \boldsymbol{W}(0) = \boldsymbol{W}_0 \end{cases}$$

där

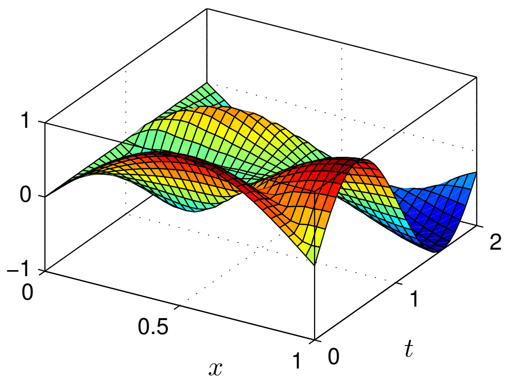
$$\boldsymbol{W} = \begin{bmatrix} \boldsymbol{U} \\ \boldsymbol{V} \end{bmatrix}, \quad \boldsymbol{f}(t, \boldsymbol{W}) = \begin{bmatrix} \boldsymbol{V}(t) \\ \frac{c^2}{h^2}(\boldsymbol{b}(t) - \boldsymbol{A}\boldsymbol{U}(t)) \end{bmatrix}, \quad \boldsymbol{W}_0 = \begin{bmatrix} \boldsymbol{U}_0 \\ \boldsymbol{V}_0 \end{bmatrix}$$

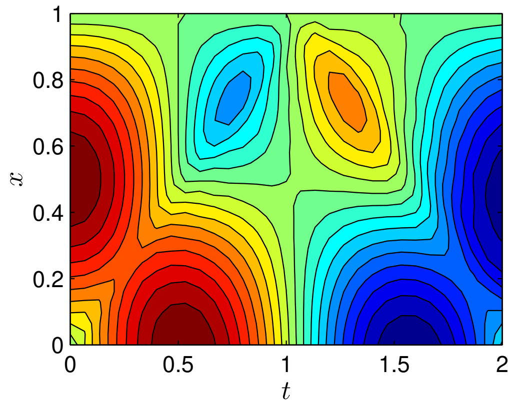
Vi tar $c = 1$ och följande rand- och begynnelsevillkor

$$g_0(t) = 0, \ g_1(t) = \sin(3t), \ u_0(x) = \sin(\pi x), \ v_0(x) = 0$$

och beräknar lösningen med MATLAB enligt

```
>> T=2; L=1; c=1; u0=@(x)sin(pi*x);
>> n=30; m=30; h=L/(n+1); xi=h*[1:n]'; tspan=linspace(0,T,m+1);
>> U0=u0(xi); V0=zeros(size(xi)); W0=[U0; V0];
>> A=spdiags(ones(n,1)*[-1 2 -1],[-1 0 1],n,n); b=@(t)[0;zeros(n-2,1);sin(3*t)];
>> f=@(t,w)[w(n+1:2*n);c/h^2*(b(t)-A*w(1:n))];
>> [t,W]=ode45(f,tspan,W0);
>> x=[0;xi;L]; U=[zeros(size(t)),W(:,1:n),sin(3*t)];  % Kantar med randvärden
>> surf(x,t,U)
>> xlabel('x'), ylabel('t')
```





4 Egenvärdesproblem

Vi har i en tidigare laboration sett på *egenvärdesproblem* för ordinär differentialekvation. Som exempel tog vi

$$\begin{cases} -u''(x) = \lambda u(x), \ 0 \leq x \leq 1 \\ u(0) = u(1) = 0 \end{cases}$$

med följande egenfunktioner och motsvarande egenvärden

$$u_n(x) = B_n \sin(n\pi x), \quad \lambda_n = n^2\pi^2, \quad n = 1, 2, \cdots$$

Vanligtvis finns det ingen enkel formel för lösningen utan vi får beräkna approximationer.

Nu skall vi se på egenvärdesproblem för partiella differentialekvationer och som exempel tar vi

$$\begin{cases} -(u''_{xx} + u''_{yy}) = \lambda u, \ \text{i} \ \Omega \\ u = 0, \ \text{på} \ \partial\Omega \end{cases}$$

Tar vi Ω som enhetskvadraten kan vi skriva upp alla egenfunktioner och motsvarande egenvärden

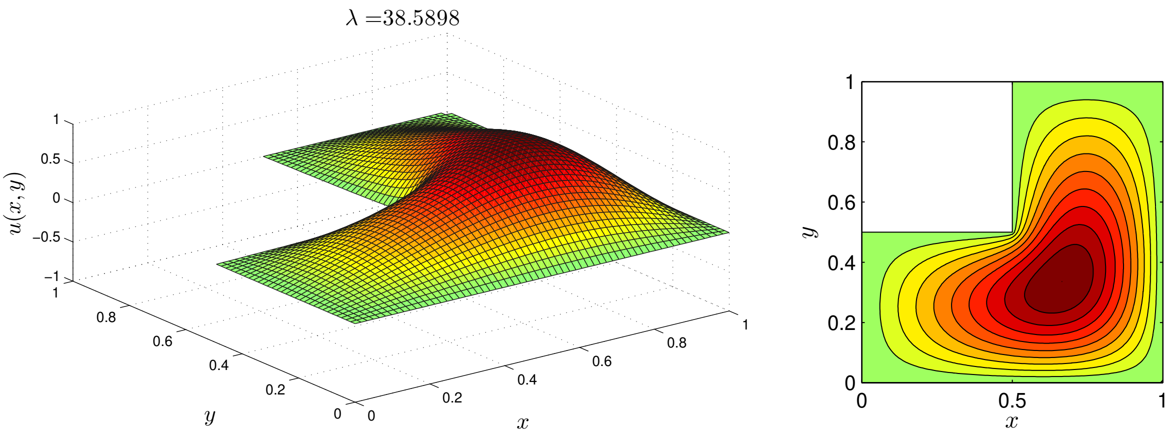
$$u_{m,n}(x, y) = B_{m,n} \sin(m\pi x) \sin(n\pi y)$$

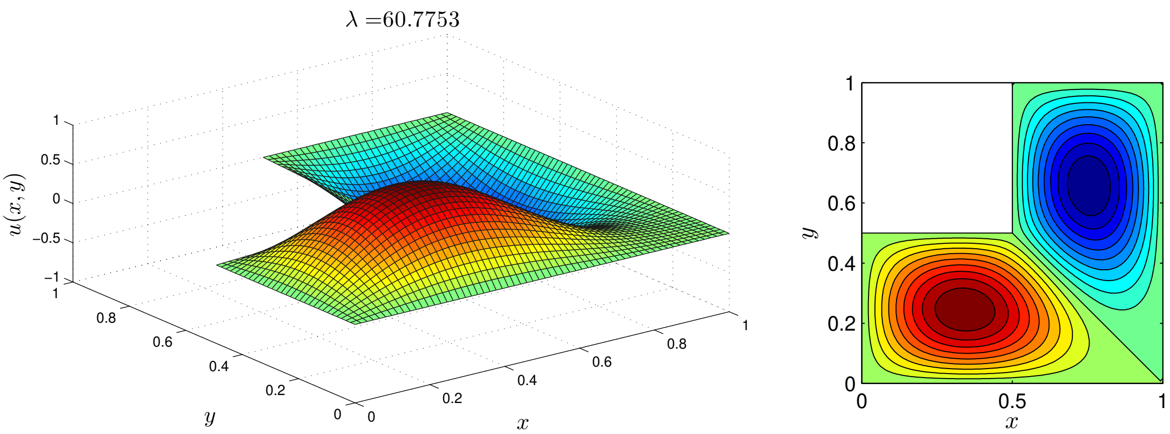
$$\lambda_{m,n} = (n^2 + m^2)\pi^2$$

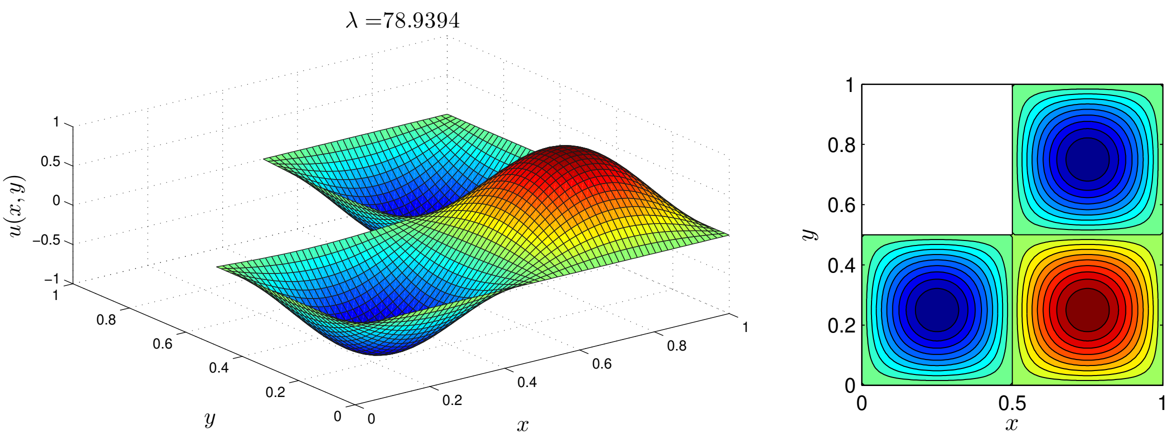
för $m = 1, 2, \cdots$ och $n = 1, 2, \cdots$.

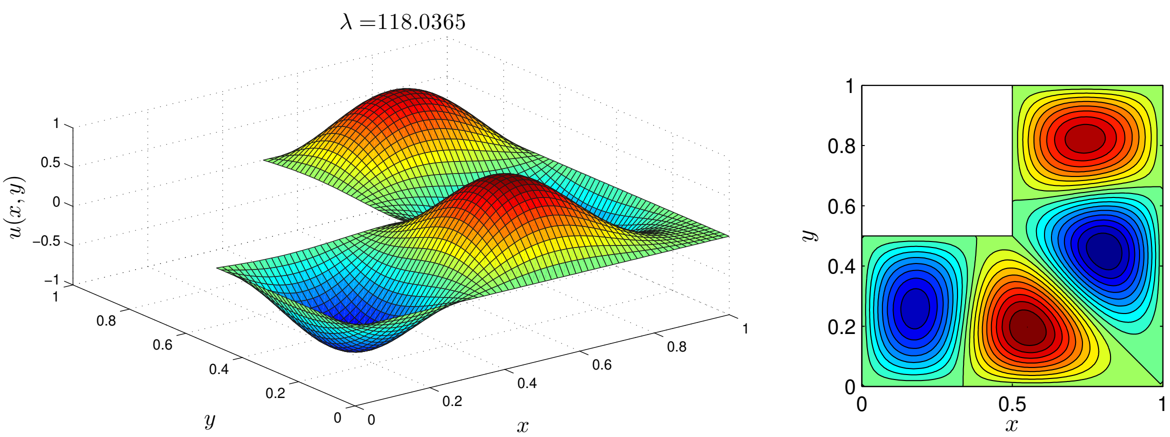
Om vi däremot tar Ω som exempelvis ett L-formigt område måste vi beräkna approximationer. Vi ersätter derivator med differensapproximationer (på samma sätt som motsvarande stationära problem) och får ett stort och glest matrisegenvärdesproblem som vi löser med `eigs`.

Här ser vi beräkningsresultatet. De fyra lägsta egenvärdena och motsvarande egenfunktioner.









5 Färdiga program i MATLAB

Det finns en funktion `pdepe` i MATLAB för lösning av system av paraboliska och elliptiska ekvationer i en rumsvariabel på formen

$$c(x,t,u,\frac{\partial u}{\partial x})\frac{\partial u}{\partial t} = x^{-m}\frac{\partial}{\partial x}\left(x^m f(x,t,u,\frac{\partial u}{\partial x})\right) + s(x,t,u,\frac{\partial u}{\partial x})$$

Åtminstone en ekvation måste vara parabolisk.

Som exempel tar vi det paraboliska problemet

$$\begin{cases} \frac{\partial u}{\partial t} = \frac{\partial}{\partial x}\left((1+x^2)\frac{\partial u}{\partial x}\right),\ 0 < x < 1, t > 0 \\ \frac{\partial u}{\partial x}(0,t) = 0, u(1,t) = 0,\ t > 0 \\ u(x,0) = 1 - x^2,\ 0 \le x \le 1 \end{cases}$$

Så här använder vi `pdepe` för att beräkna lösningen och rita upp den.

```
>> T=0.1; nx=20; nt=30; m=0;
>> x=linspace(0,1,nx); t=linspace(0,T,nt);
>> sol=pdepe(m,@pde,@ic,@bc,x,t);
>> u=sol(:,:,1);
>> subplot(1,2,1), surf(x,t,u)
>> subplot(1,2,2), contourf(x,t,u)
```

Följande funktion beskriver differentialekvationen

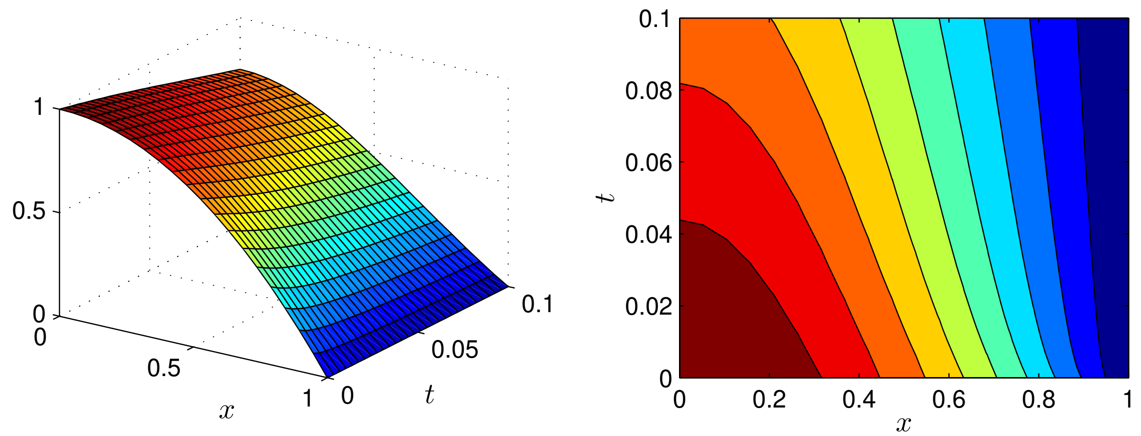
```
function [c,f,s]=pde(x,t,u,DuDx)
c=1; f=(1+x.^2).*DuDx; s=0;
```

Här beskrivs begynnelsevillllkoren (initial conditions)

```
function u0=ic(x)
u0=(1-x.^2);
```

och randvillkoren (boundary conditions)

```
function [pl,ql,pr,qr]=bc(xl,ul,xr,ur,t)
pl=0; ql=1/(1+xl^2); pr=ur; qr=0;
```



Se hjäptexterna för mer detaljer.

Det finns även en verktygslåda PDE Toolbox för lösning av PDE med finita elementmetoden som kan användas via Graphical User Interface (GUI) eller direkt som funktioner.

Lösningen sker i följande steg:

- Draw mode – Geometribeskrivning med solidmodellering och mängdoperationer
- Boundary mode – Randvillkor anges
- PDE mode – Differentialekvationen anges
- Mesh mode – Nätgenerering och förfining av nät
- Solve mode – Lösning beräknas
- Plot mode – Visualisering av beräkningsresultat