Program pro řešení PDR parabolického typu metodou predictor-corrector

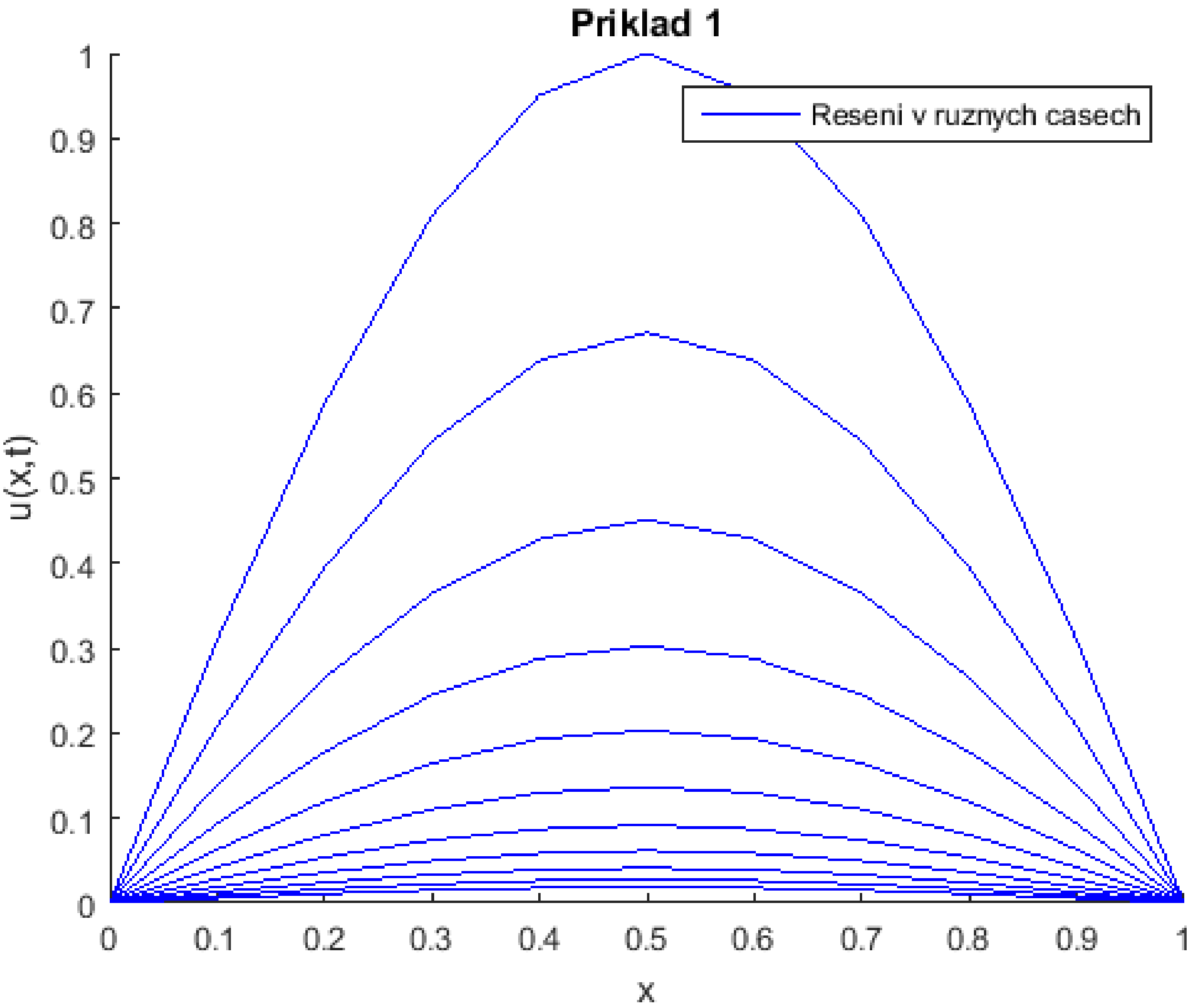
```
function u = PDEParabPC(n,m,k,a,b,g,e,f,alpha1,alpha2,beta1,beta2,gamma1,gamma2,phi)
% Reseni parcialni diferencialni rovnice parabolickeho typu metodou siti.
% Metoda Predictor-Corrector
% n...pocet kroku v prostoru
% m...pocet kroku v case
% k...casovy krok
% reseni u(x,t) = u(j,i), j = 0,...,m; i = 0,...,n;
% alpha1,alpha2,beta1,beta2,gamma1,gamma2 ... okrajove podminky

h = (b-a)/n; % krok site
u = zeros(m+1,n+1);
x = a:h:b ;
t = (0:k:m*k)';

% pomocne konstanty
alpha = k/(h*h);
beta = k/(2*h);

% pocatecni podminka
u(1,:) = phi(x);

% reseni na (j+1) casove vrstve
for j = 1:m
    % vypocet reseni na (j+1)-ni casove vrstve
    % PREDICTOR (explicitni metoda):
    u_pred = zeros(n+1);
    for i = 2:n
        d1 = alpha*g(x(i),t(j)) - beta*e(x(i),t(j));
            d2 = 1 - 2*alpha*g(x(i),t(j));
            d3 = alpha*g(x(i),t(j)) + beta*e(x(i),t(j));
        u_pred(i) = d1*u(j,i-1) + d2*u(j,i) + d3*u(j,i+1) + k*f(x(i),t(j),u(j,i));
    end

    % CORRECTOR (implicitni metoda):
    % vypocet diagonal d1,d2,d3 a prave strany rhs
    for i = 2:n
        d1(i-1)  = -alpha * g(x(i),t(j+1)) + beta * e(x(i),t(j+1));
            d2(i-1)  = 1 + 2 * alpha * g(x(i),t(j+1));
            d3(i-1)  = -alpha * g(x(i),t(j+1)) - beta*e(x(i),t(j+1));
        rhs(i-1) = u(j,i) + k*f(x(i),t(j+1),u_pred(i));
    end

    % dosazeni okrajovych podminek do prvni a posledni rovnice
    p = 1/(2*h*alpha1-3*beta1(t(j+1)));
    d2(1)  = d2(1)  - (4 *  beta1(t(j+1))  * d1(1)) * p;
    d3(1)  = d3(1)  + (     beta1(t(j+1))  * d1(1)) * p;
    rhs(1) = rhs(1) - (2*h* gamma1(t(j+1)) * d1(1)) * p;

    q = 1/(2*h*alpha2+3*beta2(t(j+1)));
    d2(n-1)  = d2(n-1)  + (4 *  beta2(t(j+1))  * d3(n-1)) * q;
    d1(n-1)  = d1(n-1)  - (     beta2(t(j+1))  * d3(n-1)) * q;
    rhs(n-1) = rhs(n-1) - (2*h* gamma2(t(j+1)) * d3(n-1)) * q;
    %reseni soustavy linearnich rovnic s tridiagonalni matici => reseni u na (j+1)-casove vrstve
    up = TriDiagonalSolve(d1,d2,d3,rhs); % reseni u ve vnitrnich uzlech site
    for i = 2:n
        u(j+1,i) = up(i-1);
    end

    % vypocet krajnich hodnot reseni u na (j+1)-casove vrstve
    u(j+1,1)   = (2*h*gamma1(t(j+1)) - 4*beta1(t(j+1))*u(j+1,2) + beta1(t(j+1))*u(j+1,3)) * p;
    u(j+1,n+1) = (2*h*gamma2(t(j+1)) + 4*beta2(t(j+1))*u(j+1,n) - beta2(t(j+1))*u(j+1,n-1)) * q;
end



end
```

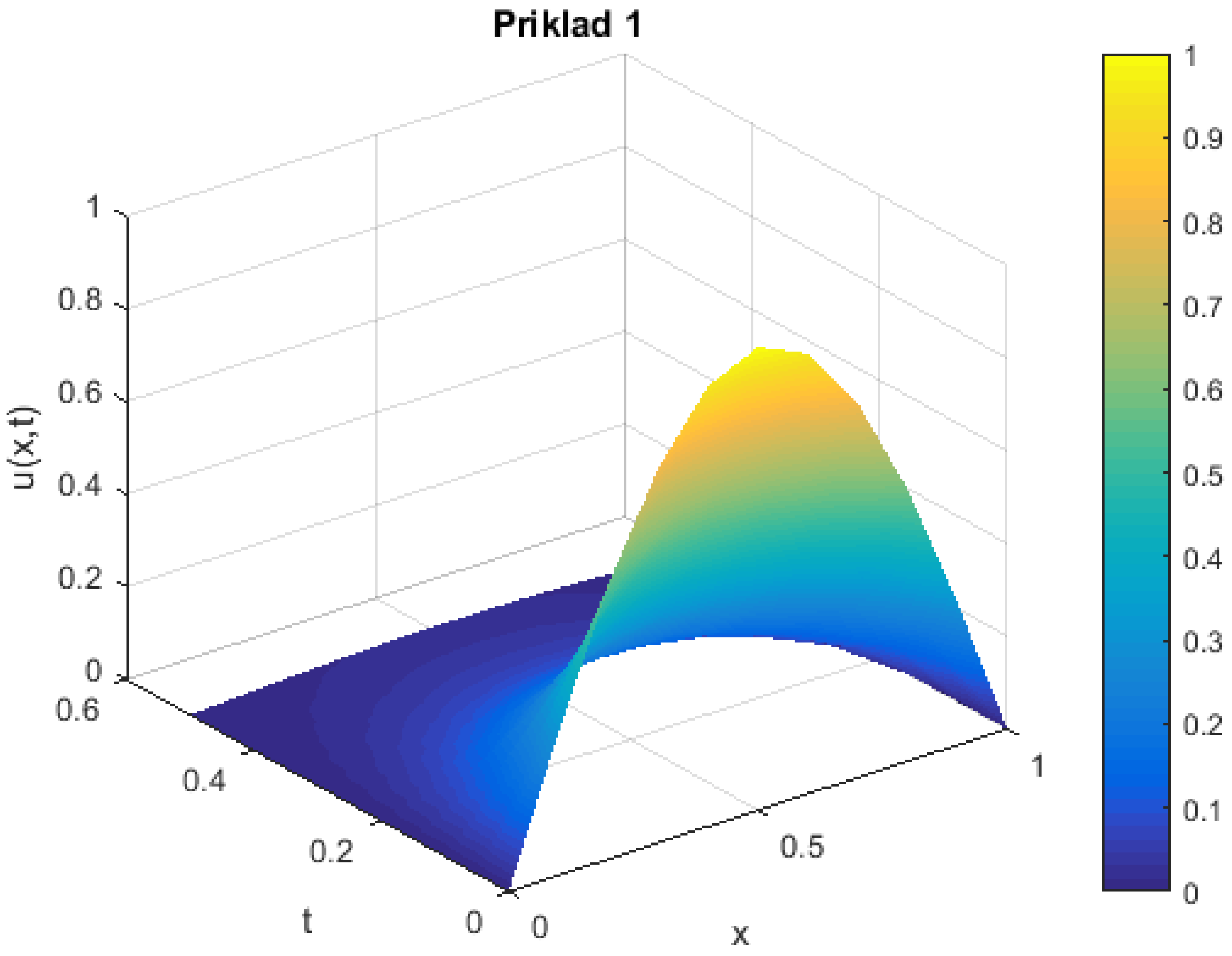
Published with MATLAB® R2015a

Použití metody PDEParabPC pro daný příklad:

```
                % du/dt = d2u/dx2
% u(0,t) = 0, u(1,t) = 0
% u(x,0) = sin(pi*x)

a = 0;
b = 1;
n = 10;
h = (b-a)/n;
m = 10;
k = 0.05;
g = inline('1','x','t');
e = inline('0','x','t');
f = inline('0','x','t','u');
alpha1 = 1;
beta1 = inline('0','t');
gamma1 = inline('0','t');
alpha2 = 2;
beta2 = inline('0','t');
gamma2 = inline('0','t');
phi = inline('sin(pi*x)','x');

u = PDEParabPC(n,m,k,a,b,g,e,f,alpha1,alpha2,beta1,beta2,gamma1,gamma2,phi);

figure;
hold on;
x = a:h:b;
for j = 1:m+1
  plot(x,u(j,:),'b');
end
title('Priklad 1');
xlabel('x');
ylabel('u(x,t)');
legend('Reseni v ruznych casech');
hold off;
hold off;

figure;
x = a:h:b;
t = (0:k:m*k)';
surf(x,t,u);
shading interp;
colorbar;
xlabel('x');
```

```
ylabel('t');
zlabel('u(x,t)');
title('Priklad 1');
```





Published with MATLAB® R2015a