

Trust region method and the Dogleg method

```
function [x,k] = trust_region1(varargin)%建立函数
if nargin < 1, help(mfilename),
    f = @(x1,x2)100*(x2-x1^2)^2 + (1-x1)^2;
    options = struct('f',f,'x0',[0;0],'Delta_hat',10,'Delta0',0.1,'eta',0);%罗列初始值及参数
    addpath(genpath(pwd));%添加路径
end

f = options.f;
syms x1 x2;
f_s = f(x1,x2);
x0 = options.x0;
Delta_hat = options.Delta_hat;
Delta0 = options.Delta0;
eta = options.eta;
df = diff_handle(f_s);
B = hessian(f_s);
Delta = Delta0;
x = x0;
step = 100;
for k = 0:step-1
    x;
    fk = f(x(1),x(2));
    dfk = df(x(1),x(2));
    Bk = subs(B,{x1,x2},{x(1),x(2)});
    Bk = double(Bk);
    sk = cal_sk(dfk,Bk,Delta);
    m = @(p) fk + dfk'*p + 1/2*p'*Bk*p;
    rho_k = cal_rho_k(f,m,x,sk);
    if rho_k < 1/4
        Delta = 1/4*Delta;
    elseif rho_k > 3/4 && abs(norm(sk,2) - Delta)<1.0e-10
        Delta = min(2*Delta,Delta_hat);
    else
        %不做任何操作;
    end

    if rho_k > eta
        x = x + sk;
    else
        %不做任何操作;
    end
end
```

```

        if k <= 1
            fprintf('第%d 次迭代后 x 的值为: \n',k+1) ;%对结果进行打印输出
            disp(x)
            fprintf('第%k 次迭代后 x 值为: %x',k,x);
        end
    end

end

end

```

```

function df = diff_handle(f_s)
syms x1 x2;
df = [diff(f_s,x1); diff(f_s,x2)];
df = matlabFunction(df);
end

```

```

function tau = cal_tau(pB,pU,Delta)
npB = sqrt(pB'*pB);
npU = sqrt(pU'*pU);
if npB <= Delta
    tau = 2;
elseif npU >= Delta
    tau = Delta/npU;
else
    pB_U = pB-pU;
    tau = (-pU'*pB_U+sqrt((pU'*pB_U)^2-pB_U'*pB_U*(pU'*pU-Delta^2)))/(pB_U'*pB_U);
    tau = tau + 1;
end
end

```

```

function sk = cal_sk(dfk,Bk,Delta)
pU = -(conj(dfk')*dfk)/(conj(dfk')*Bk*dfk).*dfk;
pB = -Bk^(-1)*dfk;
tau = cal_tau(pB,pU,Delta);
if tau >=0 && tau <=1
    sk = tau*pU;
elseif tau >= 1 && tau <=2
    sk = pU + (tau-1)*(pB-pU);
end

```

```
else
    error('tau 的值不能为%f',tau);
end
```

```
end
```

```
function rho_k = cal_rho_k(f,m,x,sk)
rho_k = (f(x(1),x(2)) - f(x(1)+sk(1),x(2)+sk(2)))/((m([0;0])-m(sk)));
end
```