七、附录

附录 1：优化模型遗传算法求解代码及其适应度函数（yichuan.m 和 fitness.m）

（1）遗传算法主程序（文件名：yichuan.m）

```
%这里所求出旋转角度θ减去 180 度与文中对应 !!
%运行时间大约 1 min
%运行时需将 A 题附件放入相同目录中
global t
t=1; %对第 t 个方向的投影数据进行参数优化

x0=46.4010; %旋转中心 x0 初始值
y0=59.4376; %旋转中心 y0 初始值
dn=0.2768;  %光源距离初始值
k=0.6923;   %比例系数 K 初始值
d0=66.9896; %基准线距第一条 X 射线的距离

chushizhi=zeros(6,10);
theta0=zeros(1,10);
global data1;
global data2;
data1=xlsread('A 题附件.xls',1);
data2=xlsread('A 题附件.xls',2);

lb=[117;46;51.5;61.5;0.26;0.68];
ub=[119+t;48.9;55.7;70;0.29;0.7];

%产生初始种群
for m=1:10
theta0(m)=118+t*0.98+rand()*5;
chushizhi(:,m)=[theta0(m),x0+rand()*0.5,y0+rand()*0.5,d0+rand(),dn+rand()
*0.05,k+rand()*0.05]';
end
%chushizhi(:,10)=z;  %利用较优个体作为初始种群
chushizhi=chushizhi';
options=gaoptimset('PopulationSize',10,'Generations',200,'InitialPopulati
on',chushizhi);
[z,val]=ga(@fitness,6,[],[],[],[],lb,ub,[],options);

%这里所求出旋转角度θ减去 180 度与文中对应 !!
```

（2）适应度函数（文件名：fitness.m）

```
function obj=fitness(bianliang)
global data1;
global data2;
global t;

%bianliang=z; %检验
THETA=bianliang(1);
x0=bianliang(2)/100*256+54;
y0=bianliang(3)/100*256+54;
d0=x0-bianliang(4)/100*256;
dn=bianliang(5)/100*256;
k=bianliang(6);

IMG=data1;
[lt, wt] = size(IMG);
iDiag = sqrt(lt^2 + wt^2);
ld = ceil(iDiag - lt) + 2;
wd = ceil(iDiag - wt) + 2;
padIMG = zeros(lt+ld, wt+wd);
padIMG(ceil(ld/2):(ceil(ld/2)+lt-1), ...
    ceil(wd/2):(ceil(wd/2)+wt-1)) = k*IMG;

n = size(padIMG,1);
x = linspace(1,365,n);
[X1,Y1]=meshgrid(x,x);
s=zeros(1,512);
for i=1:512
    s(i)=d0+(i-1)*dn;
end
[sx,sy]=meshgrid(s,x);
PR = zeros(512,1);

%for i = 1:180
    i=1;
    theta = (90-THETA(i))*pi/180;
    X = cos(theta)*(sx-x0)+x0+ -sin(theta)*(sy-y0);
    Y = sin(theta)*(sx-x0)+ cos(theta)*(sy-y0)+y0;
    % 二维插值
    tmpimg = interp2(X1,Y1,padIMG,X,Y);
    tmpimg(isnan(tmpimg)) = 0;
    % 累加
    PR(:,i) = (sum(tmpimg))’;
%end

pr=PR;
pp=data2(:,t);
obj=norm(pp-pr);
```

附录 2：180 个方向角度

序号	θ/度	序号	θ/度	序号	θ/度	序号	θ/度	序号	θ/度
1	-60.2967	37	-24.2350	73	11.6747	109	47.5138	145	83.2617
2	-58.9435	38	-23.2202	74	12.7115	110	48.5060	146	84.1805
3	-58.4254	39	-22.2225	75	13.7154	111	49.5190	147	85.1118
4	-57.3341	40	-21.2857	76	14.5419	112	50.5208	148	86.2685
5	-56.3043	41	-20.1952	77	15.6462	113	51.4609	149	87.1449
6	-55.3107	42	-19.1463	78	16.8351	114	52.4504	150	88.6246
7	-54.3002	43	-18.1577	79	17.6130	115	53.4494	151	89.9509
8	-53.3078	44	-17.2781	80	18.5806	116	54.3845	152	90.9994
9	-52.3122	45	-16.5567	81	19.6723	117	55.3803	153	92.0000
10	-51.2495	46	-15.2909	82	20.6029	118	56.3509	154	93.0000
11	-50.22	47	-14.1651	83	21.5183	119	57.4295	155	94.0000
12	-49.3297	48	-13.1403	84	22.2612	120	58.4608	156	95.0000
13	-48.3173	49	-12.4825	85	23.6229	121	59.4169	157	96.0000
14	-47.2956	50	-11.3356	86	24.6145	122	60.4275	158	97.0000
15	-46.3161	51	-10.3155	87	25.4120	123	61.3523	159	98.0000
16	-45.0912	52	-9.2958	88	26.3273	124	62.3655	160	98.8263
17	-44.2357	53	-8.2242	89	27.2953	125	63.3932	161	99.8246
18	-43.2116	54	-7.2031	90	28.7691	126	64.3551	162	100.9172
19	-42.2584	55	-6.5694	91	29.7621	127	65.3908	163	101.9085
20	-41.2585	56	-5.2393	92	30.6384	128	66.4252	164	102.7732
21	-40.1874	57	-4.1605	93	31.6379	129	67.4632	165	103.8476
22	-39.0825	58	-3.2054	94	32.6321	130	68.4733	166	104.8412
23	-38.1944	59	-2.1560	95	33.6445	131	69.5159	167	105.8720
24	-37.2177	60	-1.2175	96	34.6421	132	70.5026	168	106.7154
25	-36.2455	61	-0.2833	97	35.7309	133	71.4110	169	107.6914
26	-35.1394	62	0.6962	98	36.7393	134	72.3557	170	108.6776

27	-34.2304	63	1.8200	99	37.7400	135	73.4359	171	109.6540
28	-33.1695	64	2.5516	100	38.7410	136	74.4892	172	110.6817
29	-32.2621	65	3.6171	101	39.6178	137	75.4860	173	111.6492
30	-31.2381	66	4.6285	102	40.6211	138	76.4309	174	112.6334
31	-30.3495	67	5.8428	103	41.6258	139	77.3292	175	113.6086
32	-29.4376	68	6.3163	104	42.5254	140	78.2936	176	114.6067
33	-28.2953	69	7.5960	105	43.5246	141	79.3708	177	115.6106
34	-27.2949	70	8.7032	106	44.4887	142	80.3892	178	116.6488
35	-26.2543	71	9.7148	107	45.5191	143	81.5161	179	117.6440
36	-25.2763	72	10.5256	108	46.5214	144	82.4647	180	118.5965

附录3：SART算法程序（文件名：SART.m）

```
x0=54.9014;
y0=46.6736;
dn=0.2765;
di=64.0266;
k=0.6973;
theta=reshape([ 29.7033    47.7416    65.7650    83.4306   101.6747   119.7621
137.5138   155.3908   173.2617   191.9085
    31.0565    48.7415    66.7798    84.7607   102.7115   120.6384   138.5059
156.4252   174.1805   192.7732
    31.5746    49.8126    67.7775    85.8395   103.7154   121.6379   139.5190
157.4632   175.1118   193.8476
    32.6659    50.9175    68.7143    86.7946   104.5419   122.6321   140.5209
158.4733   176.2685   194.8412
    33.6957    51.8056    69.8048    87.8440   105.6461   123.6444   141.4609
159.5159   177.1449   195.8720
    34.6893    52.7823    70.8537    88.7825   106.8351   124.6421   142.4504
160.5026   178.6246   196.7154
    35.6998    53.7545    71.8423    89.7167   107.6130   125.7309   143.4494
161.4110   179.9509   197.6914
    36.6922    54.8606    72.7219    90.6962   108.5806   126.7393   144.3845
162.3557   180.9994   198.6776
    37.6878    55.7696    73.4433    91.8200   109.6723   127.7400   145.3803
163.4359   182.0000   199.6540
    38.7505    56.8305    74.7091    92.5516   110.6029   128.7410   146.3509
164.4892   183.0000   200.6817
    39.7800    57.7379    75.8349    93.6171   111.5183   129.6177   147.4295
165.4860   184.0000   201.6492
```

```
   40.6703    58.7619    76.8597    94.6285   112.2612   130.6211   148.4608
166.4309   185.0000   202.6334
   41.6827    59.6505    77.5175    95.8428   113.6229   131.6258   149.4169
167.3292   186.0000   203.6086
   42.7044    60.5624    78.6644    96.3163   114.6145   132.5254   150.4275
168.2936   187.0000   204.6067
   43.6839    61.7047    79.6845    97.5960   115.4120   133.5246   151.3523
169.3708   188.0000   205.6106
   44.9088    62.7051    80.7042    98.7032   116.3273   134.4887   152.3655
170.3892   188.8263   206.6488
   45.7643    63.7457    81.7758    99.7148   117.2953   135.5191   153.3932
171.5161   189.8246   207.6440
   46.7884    64.7237    82.7969   100.5256   118.7691   136.5214   154.3551
172.4647   190.9172   208.5965],1,[]);%读入θ度数
theta=theta*2.86/180-1.3;
x_start=zeros(256,256);
x_current=x_start;
x_new=x_current;
x_current0=reshape(x_current,1,[]);
x_new0=reshape(x_new,1,[]);
 kongzhi=1;
  %构建贡献矩阵
for j0=1:1
    for i0=1:512
   d=(i0-1)*dn-di;
   Mi0=zeros(256,256);
   for i=1:256
    for j=1:256
         xp=j*100/256;
         yp=100-100*i/256;
       if abs((1*yp-
tan(theta(j0))*xp+tan(theta(j0))*x0+tan(theta(j0))*d*sin(theta(j0))+d*cos(theta(j
0))-y0)...
/sqrt(1^2+(tan(theta(j0)))^2))<=100/512
           Mi0(i,j)=1;
       end
    end
   end
   mi0_0=reshape(Mi0,1,[]);
    a(kongzhi,:)=mi0_0;
   kongzhi=kongzhi+1;
    end
end
```

```
p0=reshape(fujian6,1,[]);
 %迭代过程，diedai为循环变量，循环100次
for diedai=1:100
  for i=1:1*512
    p_star(i)=sum(a(i,:).*x_current0);
  end
  delta=p_star-p0;
 for j=1:65536
     X(j)=0;
     for i=1:512
         X(j)=X(j)+(a(i,j)*delta(i)/sum(a(i,:)))/sum(a(:,j));
     end
 end
 x_new0=x_current0-X;
 x_current0=x_new0;
end
    shuchu=reshape(x_current0,256,[]);
```

附录4：FBP算法程序（文件名：FBP.m）

```
fujian1=xlsread('A题附件.xls',1);
fujian3=xlsread('A题附件.xls',3);
fujian4=xlsread('A题附件.xls',4);
fujian2=xlsread('A题附件.xls',2);
fujian5=xlsread( A题附件.xls',5);
R=fujian3;
theta=reshape(...
[ 29.7033    47.7416    65.7650    83.4306   101.6747   119.7621   137.5138
155.3908   173.2617   191.9085
   31.0565    48.7415    66.7798    84.7607   102.7115   120.6384   138.5059
156.4252   174.1805   192.7732
   31.5746    49.8126    67.7775    85.8395   103.7154   121.6379   139.5190
157.4632   175.1118   193.8476
   32.6659    50.9175    68.7143    86.7946   104.5419   122.6321   140.5209
158.4733   176.2685   194.8412
   33.6957    51.8056    69.8048    87.8440   105.6461   123.6444   141.4609
159.5159   177.1449   195.8720
   34.6893    52.7823    70.8537    88.7825   106.8351   124.6421   142.4504
160.5026   178.6246   196.7154
   35.6998    53.7545    71.8423    89.7167   107.6130   125.7309   143.4494
161.4110   179.9509   197.6914
   36.6922    54.8606    72.7219    90.6962   108.5806   126.7393   144.3845
162.3557   180.9994   198.6776
   37.6878    55.7696    73.4433    91.8200   109.6723   127.7400   145.3803
163.4359   182.0000   199.6540
```

```
   38.7505    56.8305    74.7091    92.5516   110.6029   128.7410   146.3509
164.4892   183.0000   200.6817
   39.7800    57.7379    75.8349    93.6171   111.5183   129.6177   147.4295
165.4860   184.0000   201.6492
   40.6703    58.7619    76.8597    94.6285   112.2612   130.6211   148.4608
166.4309   185.0000   202.6334
   41.6827    59.6505    77.5175    95.8428   113.6229   131.6258   149.4169
167.3292   186.0000   203.6086
   42.7044    60.5624    78.6644    96.3163   114.6145   132.5254   150.4275
168.2936   187.0000   204.6067
   43.6839    61.7047    79.6845    97.5960   115.4120   133.5246   151.3523
169.3708   188.0000   205.6106
   44.9088    62.7051    80.7042    98.7032   116.3273   134.4887   152.3655
170.3892   188.8263   206.6488
   45.7643    63.7457    81.7758    99.7148   117.2953   135.5191   153.3932
171.5161   189.8246   207.6440
   46.7884    64.7237    82.7969   100.5256   118.7691   136.5214   154.3551
172.4647   190.9172   208.5965],1,[]);%读入θ的度数

kuandu=2^nextpow2(size(R,1)); bianhuan=fft(R,kuandu);
filter=2*[0:(kuandu/2-1),kuandu/2:-1:1]'/kuandu;
pinlvhanshu=zeros(kuandu,180);
for i=1:180
pinlvhanshu(:,i)=bianhuan(:,i).*filter;
end
fanhanshu=real(ifft(pinlvhanshu));
jieguo=zeros(384,384);
for i=1:180
rad=theta(i)*pi/180;
for x=(-384/2+1):384/2
for y=(-384/2+1):384/2
t=round(x*1.413*cos(rad+pi/2)+y*1.413*sin(rad+pi/2));
if t+round(size(R,1)/2)>=1&&t+round(size(R,1)/2)<=size(fanhanshu,1)
jieguo(x+384/2,y+384/2)=jieguo(x+384/2,y+384/2)+fanhanshu(t+...
round(size(R,1)/2),i);
else

end
end
end
end
jieguo=jieguo/180;
jieguo=jieguo*10/3;%根据附件2的数据对每点的吸收系数进行修正
%滤掉杂点
```

```
for i=1:384
    for j=1:384
        if jieguo(i,j)>=0.1
        else
            jieguo(i,j)=0;
        end
    end
end
for i=1:10
    chazhi3=jieguo(79:334,89:344);
z(i)=interp2(chazhi3,fujian4(i,1)*256/100,(100-fujian4(i,2))*256/100,...
'nearest');
end
subplot(1,1,1),imshow(jieguo(79:334,89:344)/1.5)
%原图像作图过于明亮，此处乘以一个小系数以保证图像清晰
%最终结果的 jieguo 矩阵为 384*384，实际的 256*256 矩阵存放于 chazhi3
```

附录5：迭代FBP算法程序（文件名:IAFBP.m）

```
fujian1=xlsread('A题附件.xls',1);
fujian3=xlsread('A题附件.xls',3);
fujian4=xlsread('A题附件.xls',4);
fujian2=xlsread('A题附件.xls',2);
fujian5=xlsread( A题附件.xls',5);
P=fujian1;
R=fujian5;
theta=reshape([ 29.7033   47.7416   65.7650   83.4306   101.6747   119.7621
137.5138   155.3908   173.2617   191.9085
   31.0565   48.7415   66.7798   84.7607   102.7115   120.6384   138.5059
156.4252   174.1805   192.7732
   31.5746   49.8126   67.7775   85.8395   103.7154   121.6379   139.5190
157.4632   175.1118   193.8476
   32.6659   50.9175   68.7143   86.7946   104.5419   122.6321   140.5209
158.4733   176.2685   194.8412
   33.6957   51.8056   69.8048   87.8440   105.6461   123.6444   141.4609
159.5159   177.1449   195.8720
   34.6893   52.7823   70.8537   88.7825   106.8351   124.6421   142.4504
160.5026   178.6246   196.7154
   35.6998   53.7545   71.8423   89.7167   107.6130   125.7309   143.4494
161.4110   179.9509   197.6914
   36.6922   54.8606   72.7219   90.6962   108.5806   126.7393   144.3845
162.3557   180.9994   198.6776
   37.6878   55.7696   73.4433   91.8200   109.6723   127.7400   145.3803
163.4359   182.0000   199.6540
   38.7505   56.8305   74.7091   92.5516   110.6029   128.7410   146.3509
164.4892   183.0000   200.6817
    39.7800     57.7379     75.8349     93.6171    111.5183   129.6177   147.4295
165.4860   184.0000   201.6492
    40.6703     58.7619     76.8597     94.6285    112.2612   130.6211   148.4608
166.4309   185.0000   202.6334
    41.6827     59.6505     77.5175     95.8428    113.6229   131.6258   149.4169
167.3292   186.0000   203.6086
    42.7044     60.5624     78.6644     96.3163    114.6145   132.5254   150.4275
168.2936   187.0000   204.6067
    43.6839     61.7047     79.6845     97.5960    115.4120   133.5246   151.3523
169.3708   188.0000   205.6106
    44.9088     62.7051     80.7042     98.7032    116.3273   134.4887   152.3655
170.3892   188.8263   206.6488
    45.7643     63.7457     81.7758     99.7148    117.2953   135.5191   153.3932
171.5161   189.8246   207.6440
    46.7884     64.7237     82.7969   100.5256   118.7691   136.5214   154.3551
172.4647   190.9172   208.5965],1,[]);

kuandu=2^nextpow2(size(R,1)); %为FFT变换指定宽度
%FFT变换
bianhuan=fft(R,kuandu);
filter=2*[0:(kuandu/2-1),kuandu/2:-1:1]'/kuandu;
 pinlvhanshu=zeros(kuandu,180);
for i=1:180
pinlvhanshu(:,i)=bianhuan(:,i).*filter;
end
%反变换
fanhanshu=real(ifft(pinlvhanshu));
jieguo=zeros(384,384);
for i=1:180
rad=theta(i)*pi/180;%化为弧度制
 for x=(-384/2+1):384/2
  for y=(-384/2+1):384/2
     t=round(x*1.413*cos(rad+pi/2)+y*1.413*sin(rad+pi/2));
       if t+round(size(R,1)/2)>=1&&t+round(size(R,1)/2)<=size(fanhanshu,1)
          jieguo(x+384/2,y+384/2)=jieguo(x+384/2,y+384/2)+...
          fanhanshu(t+round(size(R,1)/2),i);
       else
       end
  end
 end
end
jieguo=jieguo/180;
jieguo1=jieguo;
```

```
shuchu=zeros(384,384);
for i=1:384
    for j=1:384
        if jieguo(i,j)>=0
         shuchu(i,j)=jieguo(i,j);
        else
        end
    end
end


 for xunhuan=1:5
   A=radon(shuchu,theta,768);
   A0=zeros(768,180);
   for i=1:768
       for j=1:180
           if A(i,j)>=0
               A0=A;
           else
           end
       end
   end
 pd=R-A0(129:640,:);
 kuandu=2^nextpow2(size(pd,1)); bianhuan=fft(pd,kuandu);
 filter=2*[0:(kuandu/2-1),kuandu/2:-1:1]'/kuandu;
pinlvhanshu=zeros(kuandu,180);
 for i=1:180
 pinlvhanshu(:,i)=bianhuan(:,i).*filter;
 end
fanhanshu=real(ifft(pinlvhanshu));
jieguo=zeros(384,384);
 for i=1:180
rad=theta(i)*pi/180;
  for x=(-384/2+1):384/2
   for y=(-384/2+1):384/2
    t=round(x*1.413*cos(rad+pi/2)+y*1.413*sin(rad+pi/2));
    if t+round(size(R,1)/2)>=1&&t+round(size(R,1)/2)<=size(fanhanshu,1)
        jieguo(x+384/2,y+384/2)=jieguo(x+384/2,y+384/2)+...
       fanhanshu(t+round(size(R,1)/2),i);
    else
    end
   end
  end
end
```

```
  jieguo=jieguo/180;
%保持每次结果非负
jieguo0=zeros(384,384);
  for i=1:384
      for j=1:384
          if jieguo(i,j)>=0
            jieguo0(i,j)=jieguo(i,j);
          else
          end
      end
end

  shuchu=jieguo0+shuchu;
  end
shuchu=shuchu*100/85;
%滤掉杂点
for i=1:384
      for j=1:384
          if shuchu(i,j)>=0.1
          else
              shuchu(i,j)=0;
          end
      end
end
%画出图象，此处除以三也是由于原图像太亮无法看清细节
subplot(1,1,1),imshow(shuchu(79:334,89:344)/3)
%插值
for i=1:10
      chazhi5=shuchu(79:334,89:344);
y(i)=interp2(chazhi5,fujian4(i,1)*256/100,(100-fujian4(i,2))*256/100,'nearest');
end
%最终得到的 256*256 矩阵存放于 chazhi5 中
```