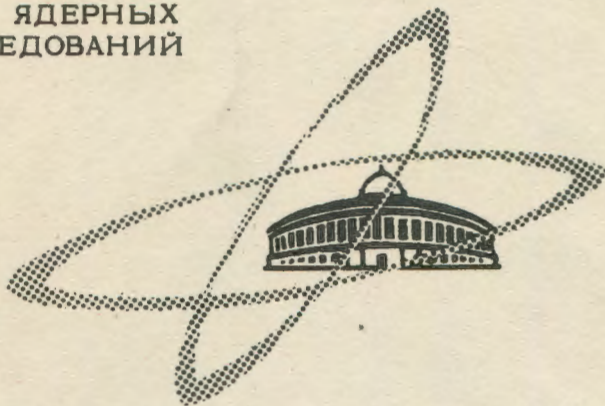


К-782

ОБЪЕДИНЕННЫЙ
ИНСТИТУТ
ЯДЕРНЫХ
ИССЛЕДОВАНИЙ

Дубна

2858



Ш. Красновски, Э. Надь, Ф. Телбис

ПРОГРАММА ДЛЯ ОПРЕДЕЛЕНИЯ КОНСТАНТ
ОПТИЧЕСКОЙ СИСТЕМЫ ПУЗЫРЬКОВЫХ КАМЕР
С ПОМОЩЬЮ ИЗМЕРЕНИЯ
СООТВЕТСТВУЮЩИХ ТОЧЕК

ЛАБОРАТОРИЯ ВЫСОКИХ ЭНЕРГИЙ

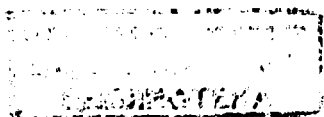
1966

2656

4181/2 чр.

Ш. Красновски, Э. Надь, Ф. Телбис

ПРОГРАММА ДЛЯ ОПРЕДЕЛЕНИЯ КОНСТАНТ
ОПТИЧЕСКОЙ СИСТЕМЫ ПУЗЫРЬКОВЫХ КАМЕР
С ПОМОЩЬЮ ИЗМЕРЕНИЯ
СООТВЕТСТВУЮЩИХ ТОЧЕК



1. В геометрической программе ^{/1/} пространственная координата, Z , вычисляется по следующей формуле

$$Z = \frac{z + z'}{2}, \quad (1)$$

где

$$z = \left[\frac{B \cdot r}{p \cdot \operatorname{tg} \beta(r)} + dt(r) \right] \operatorname{tg}(r) - [d_0 \operatorname{tg}(r) + d_1 \frac{\operatorname{tg}(r)}{\operatorname{tg}_1(r)} + d_2 \frac{\operatorname{tg}(r)}{\operatorname{tg}_2(r)}], \quad (2)$$

$$z' = \left[\frac{B \cdot r'}{p' \cdot \operatorname{tg}' \beta(r')} + dt(r') \right] \operatorname{tg}'(r') - [d'_0 \operatorname{tg}'(r') + d_1 \frac{\operatorname{tg}'(r')}{\operatorname{tg}'_1(r')} + d_2 \frac{\operatorname{tg}'(r')}{\operatorname{tg}'_2(r')}] .$$

Здесь

$$p = x + \frac{x'y}{y'}; \quad p' = x' + \frac{xy'}{y}; \quad r = \sqrt{x^2 + y^2}; \quad r' = \sqrt{x'^2 + y'^2};$$

$$\operatorname{tg}(r) = \sqrt{n^2 + (n^2 - 1) \operatorname{tg}^2 \beta(r)}; \quad \operatorname{tg}_1(r) = \sqrt{n_1^2 + (n_1^2 - 1) \operatorname{tg}^2 \beta(r)}; \quad \operatorname{tg}_2(r) = \sqrt{n_2^2 + (n_2^2 - 1) \operatorname{tg}^2 \beta(r)};$$

$d_0 = (37,15 + \bar{d}_0)$ мм, где \bar{d}_0 — расстояние между передними узловыми точками объективов и защитном стеклом камеры,

d_1 — суммарная толщина стекла, через которое производится фотографирование,

d_2 — толщина слоя воды,

n, n_1, n_2 — показатели преломления пропана, стекла и воды,

b — база фотографирования,

x, y, x', y' — координаты изображений некой пространственной точки на левом и правом снимках стереопары в системе координат объектива ^{/1/}. Снимок считается левым, если для координат реперных крестов 1 и 2 в системе координат микроскопа имеет место следующее соотношение:

$$x_2 > x_1.$$

Для правого снимка

$$x'_1 > x'_2. \quad (3)$$

При работе геометрической программы ^{/1/} требуется выполнение следующих условий:

$$z_k^{(1)} = z_k^{(2)} = z_k^{(3)} = z_k^{(1)'} = z_k^{(2)'} = z_k^{(3)'} \quad (4)$$

для всех соответствующих точек, k (индекс в скобках здесь и в дальнейшем обозначает номер стереопары). Из-за неточного знания всех параметров оптической системы камеры для выполнения условия (4) нужно подобрать соответствующим образом константы d_0 и d_0' . В работе /2/ разработан метод определения этих констант с помощью измерения крестов на защитном стекле рабочей камеры /3/. Здесь мы решили эту задачу с помощью измерения соответствующих точек в рабочем объеме камеры, итак, метод можно использовать и в том случае, когда крестов на рабочем стекле нет.

2) Программа STAR(1) χ^2 вычисляет $d_0^{(1)}$, $d_0^{(1)'}$, $d_0^{(2)}$, $d_0^{(2)'}$, причем в первом приближении итерации для $d_0^{(1)'}$ и $d_0^{(2)'}$ берутся измеренные значения.

Программа STAR(2) χ^2 вычисляет те же величины, что и STAR(1), с той лишь разницей, что в первой итерации для $d_0^{(1)}$ и $d_0^{(2)}$ берутся измеренные значения.

Программы (STAR(3) χ^2 и STAR(4), χ^2 работают аналогичным образом для $d_0^{(3)}$ и $d_0^{(3)'}$.

Полученные таким образом $d_0^{(p)}$, $d_0^{(p)'}$ ($p = 1, 2, 3$) дают минимальные значения для следующих выражений:

$$\begin{aligned} \Phi_1 &= \sum_{k=4}^{np} (z_k^{(1)} - z_k^{(1)'})^2 + \sum_{k=4}^{np} (z_k^{(2)} - z_k^{(2)'})^2 + \sum_{k=4}^N (z_k^{(1)} - z_k^{(2)})^2, \\ \Phi_2 &= \sum_{k=4}^{np} (z_k^{(1)} - z_k^{(1)'})^2 + \sum_{k=4}^{np} (z_k^{(2)} - z_k^{(2)'})^2 + \sum_{k=4}^N (z_k^{(1)' - z_k^{(2)'})^2, \quad (5) \\ \Phi_3 &= \sum_{k=4}^{np} (z_k^{(2)} - z_k^{(2)'})^2 + \sum_{k=4}^N (z_k^{(3)} - z_k^{(2)})^2, \\ \Phi_4 &= \sum_{k=4}^{np} (z_k^{(2)} - z_k^{(2)'})^2 + \sum_{k=4}^N (z_k^{(1)' - z_k^{(2)'})^2. \quad (6) \end{aligned}$$

Здесь np_1 - число измеренных соответствующих точек на первой,
 np - на третьей (второй) стереопаре,
 N - число общих точек на первой и третьей (первой и второй) стереопарах.

χ^2 Алгоритмическая программа на транслятор ТА1 имеется в ВЦ ОИЯИ.

Полученные через STAR(1) и STAR(2) ((STAR(3) и STAR(4)) $d_0^{(p)}$ и $d_0^{(p)'$ надо усреднить и повторить вычисления до тех пор, пока результаты, полученные через STAR(1) и STAR(2) (STAR(3) и STAR(4)), не будут совпадать.

3. Обозначения, использованные на алгольном языке:

p - номер стереопары, p = 2 - средняя стереопара;
 p = 1, если $y_3 > y_1$;
 p = 3, если $y_3 < y_1$, относительно направления пучка ;

$$nr - n^2 ; nr1 - n_1^2 ; nr2 - n_2^2 ; par - p ;$$

$$x[1,k] - x_k ; x[2,k] - x'_k ;$$

$$\ell_x, \ell_y, dx, dy, d0[p,1] - \ell_x, \ell_y, \Delta x, \Delta y, d_0^{(x)}$$

$$\ell'_x, \ell'_y, dx', dy', d0[p,2] - \ell'_x, \ell'_y, \Delta'x, \Delta'y, d_0^{(y)}$$

tbl[1] - $\text{tg}\beta(r)$, , dt1[1] - $dt(r)$ - табличная зависимость,

$\text{tg}\beta(r)$ - разная, $dt(r)$ - общая для всех объективов. Последовательность в массиве tbl[1] (dt1[1]) следующая:

$$r_1^{(1)}, \text{tg}\beta^{(1)}(r_1), \left. \frac{d\text{tg}\beta(r)}{dr} \right|_{r=r_1}^{(1)}, r_2^{(1)}, \text{tg}\beta^{(1)}(r_2), \left. \frac{d\text{tg}\beta(r)}{dr} \right|_{r=r_2}^{(1)}, \dots$$

$$r_1^{(2)}, \text{tg}\beta^{(2)}(r_1), \left. \frac{d\text{tg}\beta(r)}{dr} \right|_{r=r_1}^{(2)}, \dots$$

здесь r_1 - точки, где значения $\text{tg}\beta(r)$ и его производное задано.

Исходные данные вводятся с помощью стандартной подпрограммы p 0042. При вводе координат точек, полученных на микроскопе, надо обратить внимание на следующую последовательность:

$$\underbrace{x_1, x_2, x_3, x_4, x_5, \dots, x_N, x_{N+1}}_{\text{кресты}}, \underbrace{x_{np}, x'_1, x'_2, x'_3, x'_4, \dots, x'_N}_{\text{общие точки}}, \dots$$

соответствующие точки

$$x'_{N+1}, \dots, x'_{np}, y_1, \dots, y_N, \dots, y_{np}, y'_1, \dots, y'_N, \dots, y'_{np} .$$

Выдача на бумаге происходит с помощью стандартной подпрограммы p 1041. Выдаются все исходные данные, кроме того, преобразованные в систему координат объективов координаты, $x_k^{(1)'} , x_k^{(1)}, x_k^{(2)'} , x_k^{(2)}, x_k^{(3)'} , x_k^{(3)}, x_k^{(4)'} , x_k^{(4)}, \dots, x_k^{(np)'} , x_k^{(np)}, x_k^{(1)'} , x_k^{(1)}, x_k^{(2)'} , x_k^{(2)}, x_k^{(3)'} , x_k^{(3)}, x_k^{(4)'} , x_k^{(4)}, \dots, x_k^{(np)'} , x_k^{(np)}$, но в нашей программе ℓ'_x имеет отрицательный знак.

до [р, ℓ] (р = 1, 2, 3; ℓ = 1, 2).

* * *

Авторы благодарны Э. Феньвешу за его постоянный интерес к работе, И. Паточке за ценные дискуссии по этой теме.

Л и т е р а т у р а

1. Н.А. Буздавина, З.М. Иванченко, В.Г. Иванов, И. Паточка, М.И. Попов. Препринт ОИЯИ, 2085, Дубна, 1985.
2. В.Г. Иванов. Препринт ОИЯИ, 2648, Дубна, 1986.
3. А.В. Богомолов, Ю.А. Будагов, А.Т. Василенко, В.П. Джелелов, Н.И. Дьяков, В.Г. Иванов, В.С. Кладницкий, З.И. Лешлюв, Ю.Ф. Ломакин, В.И. Москалев, В.Б. Флягин, Т.И. Шетет, П.В. Шляпников. ПТЭ, № 1, 61 (1984).

Рукопись поступила в издательский отдел
31 марта 1986 г.

```

begin integer p,np; L: p0042(p,np); p1041(p,np);
Begin real d1,d2,nr,nr1,nr2,B; integer m,n,N,np1;
  array x,y,x1,y1[1:2,1:np],lx,ly,dx,dy,d0[1:3,1:2],
    tb1[1:270], dt1[1:51];
procedure film(p); value p; integer p;
begin real sfi,cfi,epsx,epsy,dxm,dym,r; integer l,n;
for l:=1,2 do begin
dxm:=if l=1 then x[1,2]-x[1,1] else x[1,1]-x[1,2];
dym:=if l=1 then y[1,2]-y[1,1] else y[1,1]-y[1,2];
r:=sqrt(dxm2+dym2); sfi:=dym/r; cfi:=dxm/r; epsx:=lx[p,l]/r;
dxm:=x[1,3]-x[1,1]; dym:=y[1,3]-y[1,1]; epsy:=ly[p,l]/sqrt
(dxm2+dym2);
for n:=4 step 1 until np do begin
dxm:=x[1,n]-x[1,1]; dym:=y[1,n]-y[1,1];
x[1,n]:= (dxm*cfi+dym*sfi)*epsx-dx[p,1];
y[1,n]:= (-dxm*sfi+dym*cfi)*epsy-dx[p,1]
end n; end l; end film;
procedure H(n); value n; integer n;
begin real e; integer p,i,j,k;
  array a[1:2,1:3],b[1:6,1:3],c[1:2];
for k:=4 step 1 until N do begin for p:=1,3 do begin i:=1;
e:=f(p,i,k); for i:=n+2,n+4 do b[i,p]:=f(p,i,k) end p;
for i:=1,2 do for j:=1,2 do
a[i,j]:= (if k=4 then 0 else a[i,j]) + (-1)i+j *
  b[n+2, 2xi-1] * b[n+2, 2xj-1];
for i:=1,2 do a[i,3]:= (if k=4 then 0 else a[i,3]) + b[n+2, 2xi-1] *
  (b[n+4, 2xi-1] - b[n+4, 5-2xi]) end k;
for p:=1,3 do begin for k=4 step 1 until (if p=1 then np1 else
np) do begin
i:=1; e:=f(p,i,k);
for i:=n,n+2,n+4 do b[i,p]:=if i=1 then e else f(p,i,k);
a[0.5xp+0.5, 0.5xp+0.5]:=a[0.5xp+0.5, 0.5xp+0.5]+b[n+2,p]2;
a[0.5xp+0.5, 3]:=a[0.5xp+0.5, 3]+b[n+2,p]*x(b[n+4,p]-b[n,p])
end k; end p; i:=2; j:=1; k:=2; p1074(i,j,k,a,c);
d0[1,n]:=a[1,3]; d0[3,n]:=a[2,3] end H;

```

```

real procedure f(p,i,k); value p,i,k; integer p,i,k;
begin real tb,tf,r,par,dit; integer j,l,q;
    array fix[1:10],g[1:45],h[1:51];
    switch s:=s1,s2,s3,s4,s5,s6,s7;
if i#1 then go to s[i]; for l:=1,2 do begin
r:=if p=1 then sqrt(x1[l,k]2+y1[l,k]2)
    else sqrt(x[l,k]2+y[l,k]2); q:=2x(p+1)+1;
for j:=1 step 1 until 45 do g[j]:=tb1[45x(q-1)+j]; p1054(x,tb,g);
for j:=1 step 1 until 51 do h[j]:=dt1[j]; p1054(r,dit,h);
tf:=sqrt(nr+(nr-1)xtb2);
par:=if p=1 then x1[l,k]+x1[3-l,k]xy1[l,k]/y1[3-l,k]
    else x[l,k]+x[3-l,k]xy[l,k]/y[3-l,k];
fix[1]:= (Bxr/par/tb+dit)xtf; fix[1+2]:=tf;
fix[1+4]:=tf/sqrt(nr1+(nr1-1)xtb2);
fix[1+6]:=tf/sqrt(nr2+(nr2-1)xtb2);
fix[1+8]:=fix[1]-(d0[p,1]xfix[1+2]+d1xfix[1+4]+d2xfix[1+6])
end l; go to s[i];
s1: f:=fix[10]; go to s7; s2: f:=fix[9]; go to s7;
s3: f:=fix[3]; go to s7; s4: f:=fix[4]; go to s7;
s5: f:=fix[9]+d0[p,1]xfix[3]; go to s7;
s6: f:=fix[10]+d0[p,2]xfix[4]; go to s7; s7: end f;
p0042(x,y); p1041(x,y); if p=3 then go to M;
p0042(lx,ly,dx,dy,d0,d1,d2); p1041(lx,ly,dx,dy,d0,d1,d2);
p0042(tb1,dt1); p1041(tb1,dt1);
nr:=1.5129; nr1:=2.23502; nr2:=1.7609; B:=400;
M: film(p); p1041(x,y);
if p=1 then begin np1:=np; p0042(N); p1041(N);
for n:=1,2 do for m:=1 step 1 until np do begin
x1[n,m]:=x[n,m]; y1[n,m]:=y[n,m] end n; go to L end p;
for n:=1,2 do H(n);
begin real s; integer i; array u[1:2];
for p:=1,3 do begin for m:=4 step 1 until (if p=1 then np1 else
np) do begin for i:=1,2 do u[i]:=f(p,i,m); p1041(u);
s:=(if m=4 then 0 else s)+(u[1]-u[2])2 end m; p1041(s); end end;
p1041(d0) end stop end

```


S T A R (2)

Replace the corresponding row in STAR (1) by the following one :

for n:=2,1 do H(n);

S T A R (3)

Replace the corresponding parts of STAR (1) by the following ones :

procedure H(n); value n; integer n;
begin real e,a,c; integer p,i,k; array b[1:2,1:6];
for k:=4 step 1 until N do begin for p:=1,2 do begin i:=1;
e:=f(p,i,k); for i:=3-n,n+2,n+4 do b[p,i]:=f(p,i,k) end p;
a:=(if k=4 then 0 else a)+b[2,n+2]²;
c:=(if k=4 then 0 else c)+b[2,n+2]x(b[2,n+4]-b[1,3-n]) end k;
for k:=4 step 1 until np do begin p:=2; i:=1; e:=f(p,i,k);
for i:=n,n+2,n+4 do b[p,i]:=f(p,i,k); a:=a+b[2,n+2]²;
c:=c+b[2,n+2]x(b[2,n+4]-b[2,n]) end k; dO[2,n]:=c/a end H;

if p=2 then go to M;

for p:=1,2 do

S T A R (4)

Replace the corresponding row in STAR (3) by the following one :

for n:=2,1 do H(n);