Aplikace matematiky

Josef Kateřiňák

Mini-programs for hidden lines on a surface

Aplikace matematiky, Vol. 32 (1987), No. 5, 355–363

Persistent URL: `http://dml.cz/dmlcz/104267`

32 (1987) APLIKACE MATEMATIKY No. 5, 355—363

MINI-PROGRAMS FOR HIDDEN LINES ON SURFACE

Josef Kateřiňák

(Received March 24, 1986)

Summary. Many programs for the hidden lines on a surface are known [1] but these programs are long and need a large memory of the computer. In this paper we show algorithms and the corresponding mini-programs in BASIC which can be implemented on minicomputers with plotters or plotting displays. We map the surface in the parallel or central projections.

Keywords: Euclidean geometry, hidden lines on surface, algorithms and programs in BASIC.

AMS classification: 51M05.

1. DESCRIPTION OF ALGORITHMS FOR HIDDEN LINES ON SURFACE

Let a surface $\mathbf{P}$ with parametric equations $x = f_1(r, s)$, $y = f_2(r, s)$, $z = f_3(r, s)$ have at least one seminet of parametric curves K_i in the planes ϱ_i parallel to a fixed plane ϱ and let each curve K_i divide the plane ϱ_i into the "interior" and "exterior" defined by relations $<$ and $>$.

A point A on the surface $\mathbf{P}$ and its image A' in the parallel projection in the direction of a constant non-zero vector $\mathbf{v} = (v_1, v_2, v_3)$ is "visible" if and only if the intersection of the semiline $A\mathbf{v}$ and the surface $\mathbf{P}$ is the one-point set $\{A\}$. A test for "visible" points is the following. We choose a small constant $\Delta > 0$ and consider the points $B_i = A + i\,\Delta\mathbf{v}$ and $B_{i+1} = A + (i + 1)\,\Delta\mathbf{v}$ in equidistant planes ϱ_i and ϱ_{i+1}, $i = 1, 2, \ldots, n$. The points B_i and B_{i+1} are elements of the "interior" or the "exterior", hence to these points there correspond certain negative or positive numbers u_i and u_{i+1}. If $u_i\, u_{i+1} > 0$ for all $i = 1, 2, \ldots, n$, then the point A is "visible". Otherwise, the point A is "hidden".

We approximate a curve on the surface by a sequence of line segments. If two points on the curve are both "visible", then we join these points by a line segment. Otherwise, we do not join the two points on the curve by a line segment (we do not draw "hidden" segments).

We approximate the surface by a seminet or a net of parametric or non-parametric curves on the surface.

The algorithm described above for the parallel projection can be used also for the central projection with centre $S = (r_0, r_1, r_2)$. In that case we have a variable vector $\mathbf{v} = S - A$ and a variable number $\Delta > 0$ depending on the point $A \in \mathbf{P}$ and on the distance of the equidistant planes ϱ_i. For the points $B_i = A + i\,\Delta\mathbf{v}$ we have the limit condition $i\Delta \leqq 1$.

We show the details on typical examples in Sections 2 and 3.

2. PARALLEL PROJECTION

The parallel projection has the transformation formulas

$$p = r_0 x + r_2 y + r_4 z\,, \quad q = r_1 x + r_3 y + r_5 z \tag{2.1}$$

where $r_0, r_1, r_2, r_3, r_4, r_5$ are constants and x, y, z are the orthogonal coordinates of a point in the space while p, q are the orthogonal coordinates of its image in the plane on the plotter, and four non-collinear points $(0, 0)$, (r_0, r_1), (r_2, r_3), (r_4, r_5) in the plane are the images of four points $(0, 0, 0)$, $(1, 0, 0)$, $(0, 1, 0)$, $(0, 0, 1)$ from the space.

A non-zero vector of the parallel projection $\mathbf{v} = (v_1, v_2, v_3)$ is the solution of the system of two equations

$$0 = r_0 v_1 + r_2 v_2 + r_4 v_3\,, \quad 0 = r_1 v_1 + r_3 v_2 + r_5 v_3\,. \tag{2.2}$$

Example 1. The graph $\mathbf{P}$ of a function $z = f(x, y)$ of two variables $x = r$, $y = s$ over a domain bounded by two functions $h_1(x) \leqq y \leqq h_2(x)$ and $\min x \leqq x \leqq \max x$ has parametric curves $K_i \subset \varrho_i \parallel yz$ (we do not use the parametric curves in the planes parallel to xz). We define the "interior" by $z < f(x, y)$ and the "exterior" by $z > f(x, y)$. We choose a constant $\Delta > 0$ such that $\max x - \min x$ is a natural multiple of $|\Delta v_1|$. For testing the "visible" points $A \in \mathbf{P}$ over the points $B_i = A + i\Delta\mathbf{v} = (x, y, t_6)$ we use the numbers $u_i = t_6 - f(x, y)$. The difference of the variable x is $|\Delta v_1|$. For the variable y we choose the difference $(h_2(x) - h_1(x))/k$ with a natural number k.

Let in particular $z = f(x, y) = (1{\cdot}5/4)\, x^2 - (2{\cdot}5/4)\, y^2 + 2{\cdot}5$, $h_1(x) = -h_2(x) \leqq y \leqq h_2(x) = \sqrt{((4/2{\cdot}5)\; ((1{\cdot}5/4)\, x^2 + 2{\cdot}5))}$, $-4{\cdot}2 \leqq x \leqq 4{\cdot}2$, $r_0 = 1 = r_5$, $r_1 = 0 = r_4$, $r_2 = -{\cdot}7 = r_3$.

We choose $v_2 = 1$ and from (2.2) we compute $v_1 = v_3 = {\cdot}7$. We choose $\Delta = {\cdot}5$. Hence $\Delta\mathbf{v} = ({\cdot}35,\ {\cdot}5,\ {\cdot}35)$, $|\Delta v_1| = {\cdot}35$, $\max x - \min x = 8{\cdot}4 = 24\,|\Delta v_1|$. We choose $k = 10$ and the quadrangle scale $\min p = -8{\cdot}5$, $\max p = 7{\cdot}5$, $\min q = -2{\cdot}5$, $\max q = 9{\cdot}5$ on the plotter.

Table 1 presents a program in BASIC for the seminet of parametric curves $K_i \subset \varrho_i \parallel yz$.

Line 1 commands the radian measure of angles.

Lines from 10 to 31 are the input data R0 = r_0, R1 = r_1, R2 = r_2, R3 = r_3, R4 = r_4, R5 = r_5, R6 = min p, R7 = max p, R8 = min q, R9 = max q, S0 = = min x, S1 = max x, S2 = $|\Delta v_1|$, S7 = $|\Delta v_1|$, S8 = Δv_1, S9 = Δv_2, T0 = Δv_3, K = k.

Line 70 limits the drawing on the plotter into the quadrangle with horizontal coordinates from R6 to R7 and vertical coordinates from R8 to R9.

Lines from 800 to 810 represent the subroutine for the plotting by formulas (2.1).

Command PLOT P, Q moves the pen on the line segment to the point (P, Q) so that if the pen is down it draws a segment, and if the pen is up it does not draw a segment, but it puts pen down at the point (P, Q) and draws this point (horizontal coordinate P, vertical coordinate Q).

Command PENUP lifts the pen up.

Lines from 900 to 910 represent the subroutine for the computation of values of the function $z = f(x, y)$.

Lines from 950 to 960 represent the subroutine for the computation of values of the two functions S4 = $h_1(x)$ and S5 = $h_2(x)$.

Lines from 100 to 670 represent the main program for the plotting of the "visible" parametric curves $K_i \subset \varrho_i \parallel yz$.

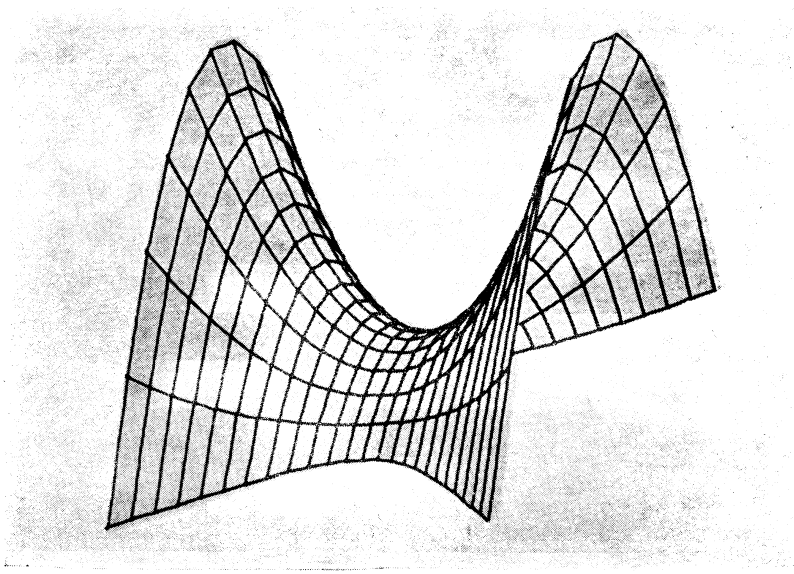

Fig. 1

For the seminet of non-parametric curves $K_j \subset \boldsymbol{P}$ for $y = h_1(x) + j(h_2(x) - - h_1(x))/k$, min $x \leqq x \leqq$ max x, $z = f(x, y)$, $j = 0, 1, 2, \ldots, k$, we add and change in Table 1 the following lines:

```
90 J=0                          600 PENUP
113 Y=S4+J*(S5-S4)/K            601 J=J+1
580 X=X+S2                      602 IF J<=K THEN 100
590 IF X<=S1 THEN 110           603 GOTO 660
```

In Figure 1 we have both seminets.

For the surface of revolution $\boldsymbol{P}$ from Example 2 of the next Section 3 in the parallel projection, the procedure is similar to Example 1.

3. CENTRAL PROJECTION

The central projection has the transformation formulas

$$
\begin{aligned}
(3.1)\qquad & w_0 = -v_1x - v_2y - v_3z\,, \quad w_1 = v_1(r_0 - x) + v_2(r_1 - y) + v_3(r_2 - z)\,, \\
& w_2 = w_0/w_1\,, \quad w_3 = x + w_2(r_0 - x)\,, \quad w_4 = y + w_2(r_1 - y)\,, \\
& w_5 = z + w_2(r_2 - z)\,, \\
& p \;= w_3v_4 + w_4v_5 + w_5v_6\,, \quad q = w_3v_7 + w_4v_8 + w_5v_9
\end{aligned}
$$

where $S = (r_0, r_1, r_2)$ is the centre of projection, the plane of images is $v_1x + v_2y + v_3z = 0$ for $v_1r_0 + v_2r_1 + v_3r_2 > 0$, (v_7, v_8, v_9) is the unit vector corresponding to the vector (w_3, w_4, w_5) for $x = 0$, $y = 0$, $z = 1$, (v_4, v_5, v_6) is the unit vector corresponding to the vector $(v_1, v_2, v_3) \times (v_7, v_8, v_9) = (v_2v_9 - v_3v_8, v_3v_7 - v_1v_9, v_1v_8 - v_2v_7)$.

Example 2. The surface of revolution $\mathbf{P}$: $x = g(z)\cos(s)$, $y = g(z)\sin(s)$, $z = r$, $\min z \leqq z \leqq \max z$, $0 \leqq s \leqq 2\pi$ has the circles $K_i \subset \varrho_i \parallel xy$ as parametric curves. We define the "interior" by $x^2 + y^2 < (g(z))^2$ and the "exterior" by $x^2 + y^2 > (g(z))^2$ (i.e., the distance between a point in the plane ϱ_i and the centre of the circle K_i is $<$ radius or $>$ radius, respectively). For testing the "visible" points $A \in \mathbf{P}$ over the points $B_i = A + \mathrm{i}\Delta\mathbf{v} = (t_4, t_5, z)$ we use the numbers $u_i = t_4^2 + t_5^2 - (g(z))^2$.

Let in particular $g(z) = \sqrt{(1 + (1/1{\cdot}2)^2 z^2)}$, $-4{\cdot}9 \leqq z \leqq 4{\cdot}9$, $r_0 = 21$, $r_1 = 30$, $r_2 = 23$, $v_1 = -{\cdot}5$, $v_2 = 9{\cdot}5$, $v_3 = 1$.

We substitute $x = 0$, $y = 0$, $z = 1$ into (3.1) and compute (w_3, w_4, w_5) and the corresponding unit vector $(v_7, v_8, v_9) = (-{\cdot}07583, -{\cdot}10833, {\cdot}99122)$. We compute $(v_1, v_2, v_3) \times (v_7, v_8, v_9)$ and the corresponding unit vector $(v_4, v_5, v_6) = ({\cdot}99575, {\cdot}04388, {\cdot}08097)$.

We choose differences $\Delta'_z = \Delta''_z = {\cdot}7$ and differences $\Delta'_s = \Delta''_s = 2\pi/30$ and $\min p = -11$, $\max p = 11$, $\min q = -9{\cdot}5$, $\max q = 7$.

Table 2 presents a program in BASIC for the seminet of parametric curves — circles $K_i \subset \varrho_i \parallel xy$.

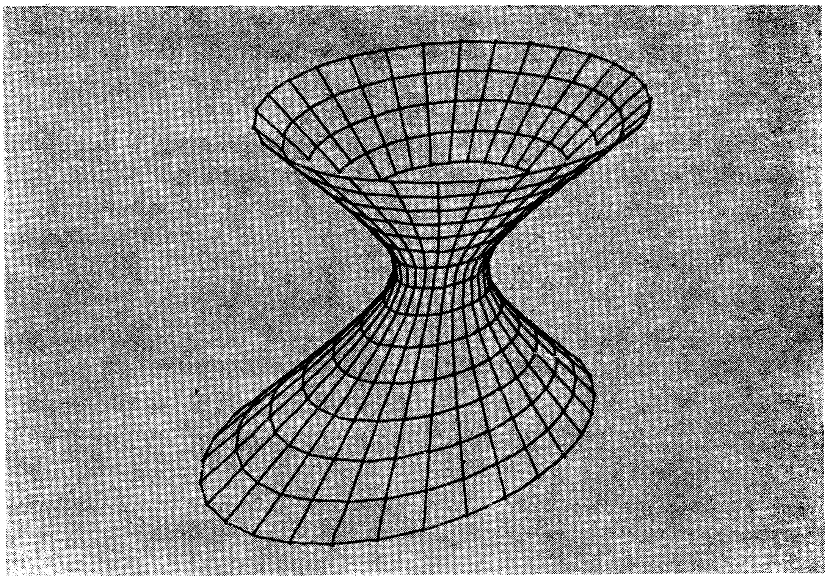

Fig. 2

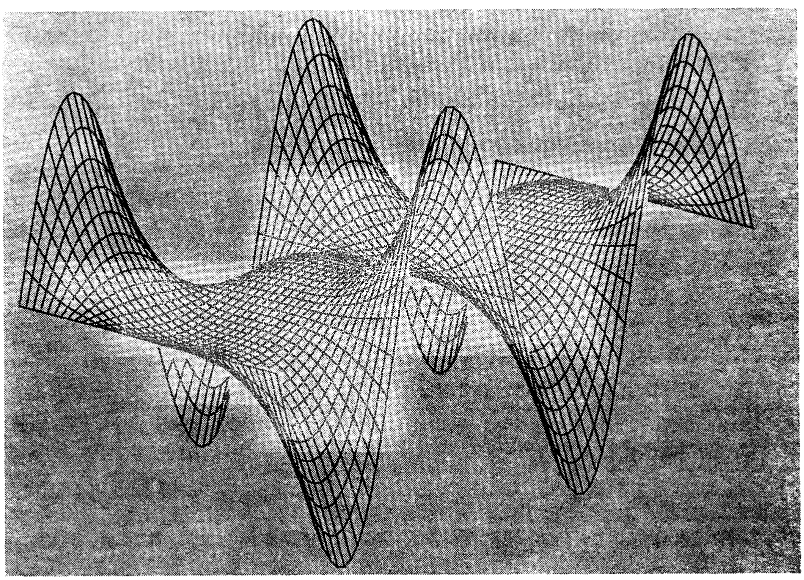

Fig. 3

Lines from 10 to 59 are the input data R0 = r_0, R1 = r_1, R2 = r_2, R6 = min p, R7 = max p, R8 = min q, R9 = max q, S0 = min z, S1 = max z, S2 = Δ'_z, S3 = = Δ''_s, S4 = 0, S5 = 2π, S6 = Δ'_s, S7 = Δ''_z, V1 = v_1, V2 = v_2, V3 = v_3, V4 = v_4, V5 = v_5, V6 = v_6, V7 = v_7, V8 = v_8, V9 = v_9.

Lines from 800 to 810 represent the subroutine for the plotting by formulas (3.1).

Lines from 900 to 910 represent the subroutine for the computation of values of the function U3 = $g(z)$.

Lines from 920 to 940 represent the subroutine for the computation of the coordinates x, y of points on the surface $\boldsymbol{P}$ (B substitutes s).

Lines from 152 to 156 are computations of the variable number C = $\Delta > 0$ and the three coordinates S8, S9, T0 of the variable vector $\Delta\boldsymbol{v} = \Delta(S - A)$ corresponding to the variable point $A \in \boldsymbol{P}$ and the distance Δ''_z of equidistant planes ϱ_i.

Lines from 100 to 670 represent the main program for the plotting of "visible" parametric curves – circles $K_i \subset \varrho_i \parallel xy$.

For the second seminet of parametric curves we change in Table 2 the following lines:

```
100 B=S4            590 IF Z<=S1 THEN 120
110 Z=S0            640 B=B+S3
580 Z=Z+S2          650 IF B<=S5 THEN 110
```

In Figure 2 we have both seminets.

For the surface $\boldsymbol{P}$ from Example 1 of the previous Section 2 in the central projection, the procedure is similar to Example 2.

4. TIME OF COMPUTATION AND ACCURACY

On older minicomputers based on the LSI-electronics, the time of computation for the seminet or net with the number of points less than 800 is roughly 30 minutes (one of Figures 1,2). On new minicomputers based on the microelectronics, the time of computation is less than 30 minutes. We performed the computation for the test surface $z = f(x, y) = (x * x * x/8 + 1) * \sin(y)$, $-4{\cdot}12477 \leqq x \leqq 4{\cdot}12477$, $-2 * \pi \leqq y \leqq 2 * \pi$, $39 * 71 = 2769$ points, parallel projection $r_0 = {\cdot}43$, $r_1 = -{\cdot}37$, $r_2 = -{\cdot}52$, $r_3 = -{\cdot}42$, $r_4 = 0$, $r_5 = 1$, (for the quadrangle domain we change in Table 1 the jump in the four lines 270, 290, 420, 440 IF ... THEN 510), Figure 3, on three minicomputers with plotters Hewlett-Packard with the following times of computation:

Minicomputer	Plotter	Produced	One seminet
HP 9830 A	HP 9862 A	1973	140 minutes
HP 9825 A	HP 9872 A	1976	20 minutes
HP 1000	HP 9872 B	1979	4 minutes

Evidently, small distance of equidistant planes ϱ_i gives more exact testing of "visible" points on the surface, but time of computation is large.

For some computers it is necessary to add in the program a small number $\pm$ D, to the boundary number (in Tables 1,2 we have D = ·001).

Table 1. Program in BASIC language — Example 1

```
1 RAD
10 R0=1
11 R1=0
12 R2=-·7
13 R3=-·7
14 R4=0
15 R5=1
16 R6=-8·5
17 R7=7·5
18 R8=-2·5
19 R9=9·5
20 S0=-4·2
21 S1=4·2
22 S2=·35
27 S7=·35
28 S8=·35
29 S9=·5
30 T0=·35
31 K=10
70 SCALE R6, R7, R8, R9

100 X=S0
110 GOSUB 950
111 U4=S4
112 U5=S5
113 Y=U4
120 GOSUB 900
130 T1=X
140 T2=Y
150 T3=Z
180 T6=T3
190 X=X+S8
200 Y=Y+S9
210 T6=T6+T0
212 GOSUB 950
230 IF S0>X THEN 510
250 IF X>S1 THEN 510
270 IF S4>Y THEN 190
290 IF Y>S5 THEN 190
310 GOSUB 900
320 U1=T6-Z
330 U0=U1
340 X=X+S8
350 Y=Y+S9
360 T6=T6+T0
362 GOSUB 950
380 IF S0>X THEN 510
400 IF X>S1 THEN 510
420 IF S4>Y THEN 340
440 IF Y>S5 THEN 340
460 GOSUB 900
470 U1=T6-Z
480 IF U0*U1>0 THEN 330
490 PENUP
500 GOTO 550
510 X=T1
520 Y=T2
530 Z=T3
540 GOSUB 800
550 X=T1
560 Y=T2
580 Y=Y+(U5-U4)/K
590 IF Y<=U5+·001 THEN 120
600 PENUP
610 X=T1
620 Y=T2
640 X=X+S7
650 IF X<=S1 THEN 110
660 PENUP
670 STOP
800 PLOT R0*X+R2*Y+R4*Z,
    R1*X+R3*Y+R5*Z
810 RETURN
900 Z=1·5/4*X*X-2·5/4*Y*Y+2·5
910 RETURN
950 S5=SQR(4/2·5*(1·5/4*X*X+2·5))
951 S4=-S5
960 RETURN
999 END
```

Table 2. Program in BASIC language — Example 2

```
1 RAD
10 R0=21
11 R1=30
12 R2=23
16 R6=-11
17 R7=11
18 R8=-9·5
19 R9=7
20 S0=-4·9
21 S1=4·9
22 S2=·7
23 S3=2*PI/30
24 S4=0
25 S5=2*PI
26 S6=2*PI/30
27 S7=·7
51 V1=-·5
52 V2=9·5
53 V3=1
54 V4=·99575
55 V5=·04388
56 V6=·08097
57 V7=-·07583
58 V8=-·10833
59 V9=·99122
70 SCALE R6, R7, R8, R9

100 Z=S0
110 B=S4
120 GOSUB 900
121 GOSUB 920
130 T1=X
140 T2=Y
150 T3=Z
152 IF ABS(R2-Z)<=·001
    THEN 510
153 C=ABS(S7/(R2-Z))
154 S8=C*(R0-X)
155 S9=C*(R1-Y)
156 T0=C*(R2-Z)
157 T=0
160 T4=T1
170 T5=T2
180 T6=T3
190 T4=T4+S8
200 T5=T5+S9
210 T6=T6+T0
211 T=T+C
220 Z=T6
221 IF T>1 THEN 510
230 IF S0>Z THEN 510
250 IF Z>S1 THEN 510
310 GOSUB 900
320 U1=T4*T4+T5*T5-U3*U3
330 U0=U1
340 T4=T4+S8
350 T5=T5+S9
360 T6=T6+T0
361 T=T+C
370 Z=T6
371 IF T>1 THEN 510
380 IF S0>Z THEN 510
400 IF Z>S1 THEN 510
460 GOSUB 900
470 U1=T4*T4+T5*T5-U3*U3
480 IF U0*U1>0 THEN 330
490 PENUP
500 GOTO 550
510 X=T1
520 Y=T2
530 Z=T3
540 GOSUB 800
550 X=T1
560 Y=T2
570 Z=T3
580 B=B+S6
590 IF B<=S5+·001 THEN 120
600 PENUP
610 X=T1
620 Y=T2
630 Z=T3
640 Z=Z+S7
650 IF Z<=S1 THEN 110
660 PENUP
670 STOP
800 W0=-V1*X-V2*Y-V3*Z
801 W1=V1*(R0-X)+V2*(R1-Y)+
    +V3*(R2-Z)
802 IF ABS(W1)>·001 THEN 805
803 PENUP
804 GOTO 810
805 W2=W0/W1
806 W3=X+W2*(R0-X)
807 W4=Y+W2*(R1-Y)
808 W5=Z+W2*(R2-Z)
809 PLOT W3*V4+W4*V5+W5*V6,
    W3*V7+W4*V8+W5*V9
810 RETURN
900 U3=SQR(1+1/1·2*(1/1·2)*Z*Z)
910 RETURN
920 X=U3*COS(B)
930 Y=U3*SIN(B)
940 RETURN
999 END
```

References

[1] *J. G. Griffith:* Bibliography of Hidden-Line and Hidden Surface Algorithms. Computer-Aided Design, 3, 1978, 2, 3—8.

[2] *J. Kateřiňák:* Geometrické struktury a počítačová grafika (Geometric Structures and Computer Graphic). Výzkumná zpráva státního vědeckého úkolu I-5-4/13. VŠDS Žilina 1981—1985, 51 p.

Souhrn

MINI-PROGRAMY PRO SKRYTÉ ČÁRY NA PLOŠE

JOSEF KATEŘIŇÁK

Je známo mnoho programů pro skryté čáry na ploše [1], avšak tyto programy jsou dlouhé a potřebují velkou paměť počítače. V této práci jsou odvozeny krátké programy pro skryté čáry na ploše, která má aspoň jednu polosíť parametrických křivek v rovnoběžných rovinách. Plocha je zobrazená v rovnoběžném nebo ve středovém promítání. Jsou uvedeny časy výpočtu na starých LSI-počítačích a na nových mikropočítačích.

Резюме

МИНИ-ПРОГРАММЫ ДЛЯ СКРЫТЫХ ЛИНИЙ НА ПОВЕРХНОСТИ

JOSEF KATEŘIŇÁK

Известны многие программы для скрытых линий на поверхности [1], но эти программы длинные и требуют большой памяти вычислительной машины. В работе показываются краткие алгорифмы и программы в языке BASIC, которые пригодны для малых вычислительных машин и координатных самописцей. Поверхность изображается в параллельной или в центральной проекции.

Author's address: RNDr. *Josef Kateřiňák,* CSc, Puškinova 18, 010 01 Žilina.