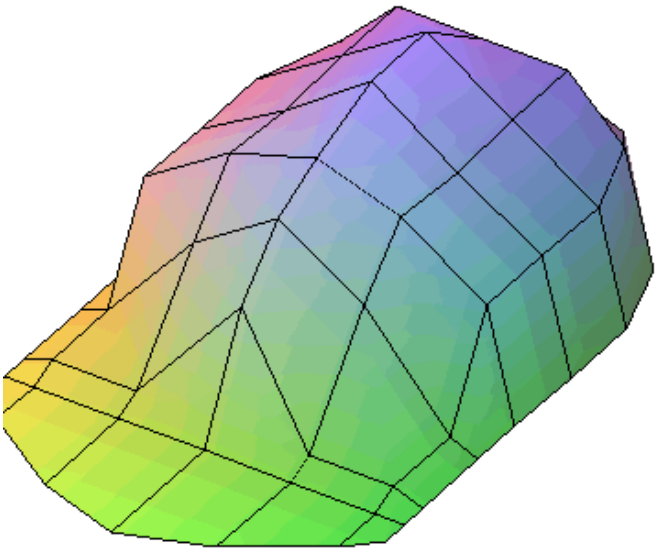
Грахов Павел, вариант 2

Объект



Выполняемые преобразования

Бикубическая интерполяция: По 16 точкам:

$$P(t) = \frac{1}{6}\begin{bmatrix} s^3 & s^2 & s & 1 \end{bmatrix}\begin{pmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 0 & 3 & 0 \\ 1 & 4 & 1 & 0 \end{pmatrix}\begin{bmatrix} P_{i,j} & P_{i,j+1} & \cdots \\ \vdots & \ddots & \\ P_{i+3,j} & & P_{i+3,j+3} \end{bmatrix}\begin{pmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 0 & 3 & 0 \\ 1 & 4 & 1 & 0 \end{pmatrix}^T\begin{bmatrix} t^3 \\ t^2 \\ t \\ 1 \end{bmatrix}$$

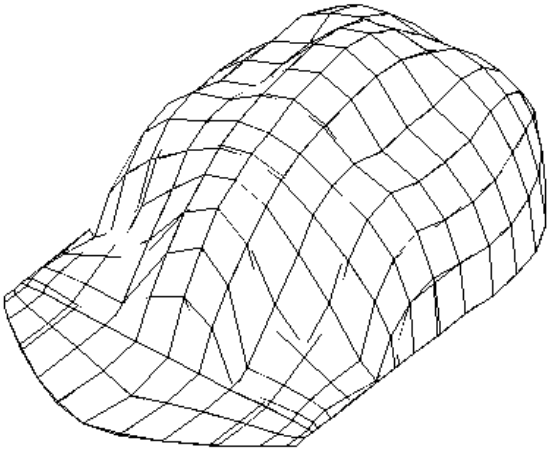
Параболическая интерполяция: По 9 точкам:

$$P(t) = \begin{bmatrix} s^2 & s & 1 \end{bmatrix}\begin{pmatrix} 1 & -4 & 2 \\ -3 & 4 & -1 \\ 1 & 0 & 0 \end{pmatrix}\begin{bmatrix} P_{i,j} & P_{i,j+1} & \cdots \\ \vdots & \ddots & \\ P_{i+2,j} & & P_{i+2,j+2} \end{bmatrix}\begin{pmatrix} 1 & -4 & 2 \\ -3 & 4 & -1 \\ 1 & 0 & 0 \end{pmatrix}^T\begin{bmatrix} t^2 \\ t \\ 1 \end{bmatrix}$$

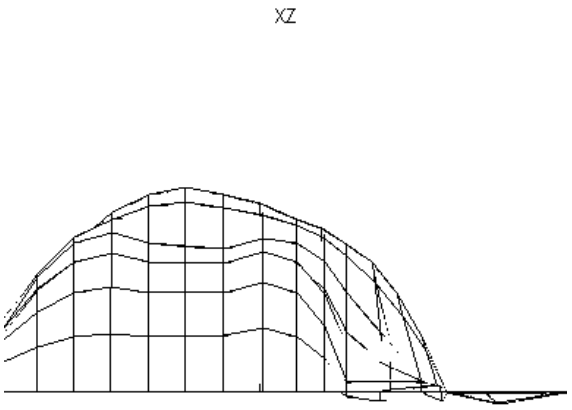
Листинг Maple-программы

```
> restart; with(plots); L[1] := [[−3.5, 10.5, 0], [−3.5, 10, 0], [−3.5, 9.5, 0], [−3.5, 8, 0], [−3.5, 7, 0], [−3.5, 5.5, 0],
    [−3.5, 4, 0], [−3, 2.5, 0], [−1.5, 1, 0], [−.5, .5, 0]]; L[2] := [[−3, 11, 0], [−3, 10, 0], [−3, 9.5, .2], [−3, 8, .2],
    [−3, 7, 2], [−3, 5.5, 2], [−3, 4, 2], [−2.5, 2.5, 2], [−1.5, 1, 1], [−.4, .5, .1]]; L[3] := [[−1.5, 12, 0], [−1.5, 10,
    0], [−1.5, 9.5, .2], [−1.5, 8, 2], [−1.5, 7, 3], [−1.5, 5.5, 2.9], [−1.5, 4, 3], [−1.5, 2.5, 3], [−1, 1, 1.2], [−.2, .3,
    .2]]; L[4] := [[0, 12.5, 0], [0, 10, 0], [0, 9, 2], [0, 8, 3], [0, 7, 3.5], [0, 5.5, 4], [0, 4, 4], [0, 2.5, 3], [0, 1,
    1.5], [0, .4, .3]]; L[5] := map(proc (x) options operator, arrow; [−x[1], op(x[2 .. 3])] end proc, L[3]); L[6] := map(
    proc (x) options operator, arrow; [−x[1], op(x[2 .. 3])] end proc, L[2]); L[7] := map(proc (x) options operator, arrow;
    [−x[1], op(x[2 .. 3])] end proc, L[1]);
> P := seq(seq([L[k][i], L[k+1][i], L[k+1][i+1], L[k][i+1]], i = 1 .. 9), k = 1 .. 6);
> Model := PLOT3D(POLYGONS(P), SCALING(CONSTRAINED)); display(Model);

> T := Vector[row]([t^2, t, 1]); S := Vector[row]([s^2, s, 1]); T4 := Vector[row]([t^3, t^2, t, 1]); S4 := Vector[row]([s^3,
    s^2, s, 1]); BsplineM := (1/6)*Matrix(4, 4, [−1, 3, −3, 1, 3, −6, 3, 0, −3, 0, 3, 0, 1, 4, 1, 0]); ParabolicM := Matrix
    (3, 3, [2, −4, 2, −3, 4, −1, 1, 0, 0]);
> Px := proc (m, n) options operator, arrow; Matrix(4, 4, [seq(L[m][n+i−1][1], i = 1 .. 4), seq(L[m+1][n+i−1][1], i = 1 .. 4)
    , seq(L[m+2][n+i−1][1], i = 1 .. 4), seq(L[m+3][n+i−1][1], i = 1 .. 4)]) end proc; Py := proc (m, n) options operator,
    arrow; Matrix(4, 4, [seq(L[m][n+i−1][2], i = 1 .. 4), seq(L[m+1][n+i−1][2], i = 1 .. 4), seq(L[m+2][n+i−1][2], i = 1 ..
    4), seq(L[m+3][n+i−1][2], i = 1 .. 4)]) end proc; Pz := proc (m, n) options operator, arrow; Matrix(4, 4, [seq(L[m][n+i
    −1][3], i = 1 .. 4), seq(L[m+1][n+i−1][3], i = 1 .. 4), seq(L[m+2][n+i−1][3], i = 1 .. 4), seq(L[m+3][n+i−1][3], i = 1 ..
     4)]) end proc; P3x := proc (m, n) options operator, arrow; Matrix(3, 3, [seq(L[m][n+i−1][1], i = 1 .. 3), seq(L[m+1][n+i
    −1][1], i = 1 .. 3), seq(L[m+2][n+i−1][1], i = 1 .. 3)]) end proc; P3y := proc (m, n) options operator, arrow; Matrix(3,
    3, [seq(L[m][n+i−1][2], i = 1 .. 3), seq(L[m+1][n+i−1][2], i = 1 .. 3), seq(L[m+2][n+i−1][2], i = 1 .. 3)]) end proc; P3z
     := proc (m, n) options operator, arrow; Matrix(3, 3, [seq(L[m][n+i−1][3], i = 1 .. 3), seq(L[m+1][n+i−1][3], i = 1 .. 3)
    , seq(L[m+2][n+i−1][3], i = 1 .. 3)]) end proc;
> X3st := proc (m, n) options operator, arrow; T.ParabolicM.P3x(m, n).ParabolicM^%T.S^%T end proc; Y3st := proc (m, n)
    options operator, arrow; T.ParabolicM.P3y(m, n).ParabolicM^%T.S^%T end proc; Z3st := proc (m, n) options operator, arrow;
     T.ParabolicM.P3z(m, n).ParabolicM^%T.S^%T end proc;
> Xst := proc (m, n) options operator, arrow; S4.BsplineM.Px(m, n).BsplineM^%T.T4^%T end proc; Yst := proc (m, n) options
    operator, arrow; S4.BsplineM.Py(m, n).BsplineM^%T.T4^%T end proc; Zst := proc (m, n) options operator, arrow; S4.BsplineM
    .Pz(m, n).BsplineM^%T.T4^%T end proc;
>
> ParabolicSmooth := seq(seq(plot3d([X3st(m, n), Y3st(m, n), Z3st(m, n)], s = 0 .. 1, t = 0 .. 1, grid = [5, 5], scaling =
    constrained, color = white), m = 1 .. 5), n = 1 .. 7);
> display(ParabolicSmooth);

> ParabolicSmoothProjXY := seq(seq(plot3d([X3st(m, n), Y3st(m, n), Z3st(m, n)], s = 0 .. 1, t = 0 .. 1, grid = [5, 5],
    scaling = constrained, color = white, orientation = [90, 0]), m = 1 .. 5), n = 1 .. 7); ParabolicSmoothProjXZ := seq(seq(
    plot3d([X3st(m, n), Y3st(m, n), Z3st(m, n)], s = 0 .. 1, t = 0 .. 1, grid = [5, 5], scaling = constrained, color = white,
     orientation = [0, 90]), m = 1 .. 5), n = 1 .. 7); ParabolicSmoothProjYZ := seq(seq(plot3d([X3st(m, n), Y3st(m, n), Z3st(
    m, n)], s = 0 .. 1, t = 0 .. 1, grid = [5, 5], scaling = constrained, color = white, orientation = [90, 90]), m = 1 .. 5)
    , n = 1 .. 7); display(ParabolicSmoothProjYZ, title = "YZ"); display(ParabolicSmoothProjXZ, title = "XZ"); display(
    ParabolicSmoothProjXY, title = "XY");

> BsplineSmooth := seq(seq(plot3d([Xst(m, n), Yst(m, n), Zst(m, n)], s = 0 .. 1, t = 0 .. 1, grid = [5, 5], scaling =
    constrained, color = white), m = 1 .. 5), n = 1 .. 6);
> display(BsplineSmooth);
```

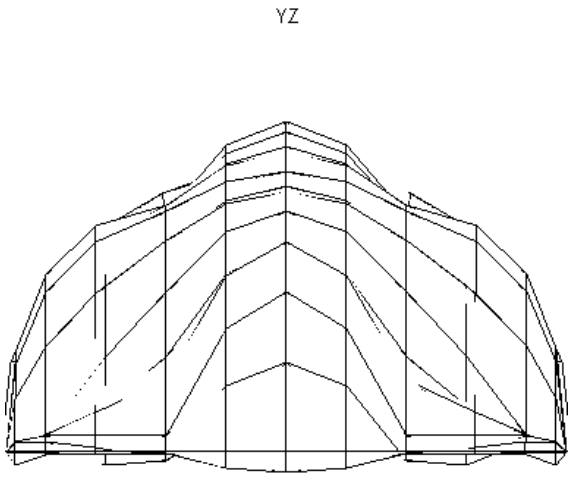
Результаты преобразований (вывод)



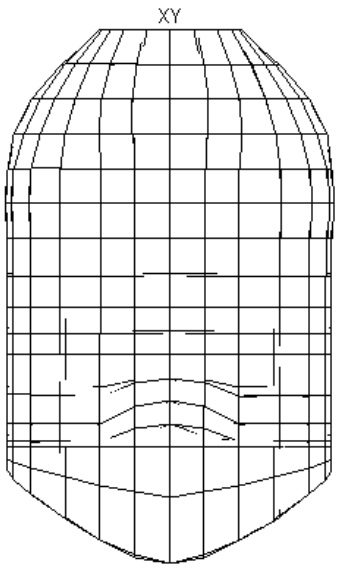
(a) Параболическая интерполяция



(b) XZ



(a) YZ



(b) XY

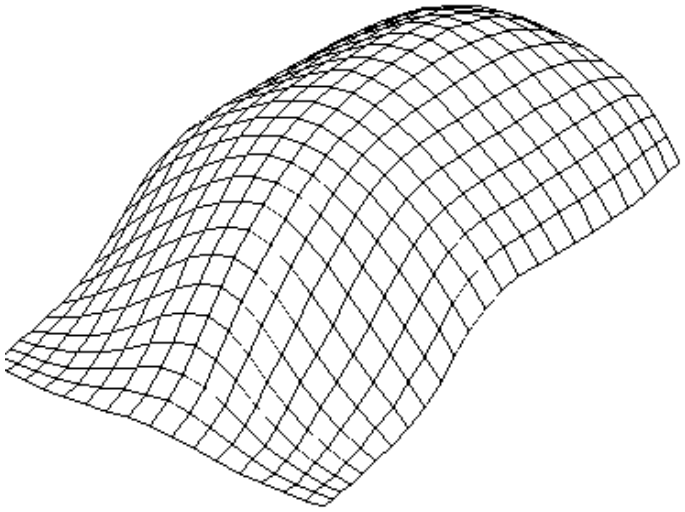


Рис. 3: Бикубическая интерполяция