# Extrémy funkce vázané podmínkou

## Funkce dvou proměnných

Následující procedura vypočte po zadání funkce f(x,y) a podmínky g(x,y) ( =0 ) příslušné lokální extrémy v bodech, kde existují druhé derivace f a g a kde *Mathematica* umí vyřešit příslušné rovnice. Procedura není ošetřena na všechny možné výskyty singularit.

```
f[x_, y_] := 2 * x^2 - 3 * y^3 + 4 * x^2 * y^2;
g[x_, y_] := x^2 + y^2 - 1;
```

```
F[x_, y_, a_] := f[x, y] - a g[x, y];
kritnul =
  Solve[{D[F[x, y, a], x] == 0, D[F[x, y, a], y] == 0,
     g[x, y] == 0}, {x, y, a}] // N;
diskr1 = D[F[x, y, a], x, x] -
   2 D[F[x, y, a], x, y] D[g[x, y], x] / D[g[x, y], y] +
   D[F[x, y, a], y, y] (D[g[x, y], x] / D[g[x, y], y]) ^2;
tab1 = Table[{{x, y}, f[x, y] // N, diskr1} /. kritnul[[i]],
   {i, 1, Length[kritnul]}];
diskr2 = D[F[x, y, a], y, y] -
   2 D[F[x, y, a], x, y] D[g[x, y], y] / D[g[x, y], x] +
   D[F[x, y, a], x, x] (D[g[x, y], y] / D[g[x, y], x]) ^2;
tab2 = Table[{{x, y}, f[x, y] // N, diskr2} /. kritnul[[i]],
   {i, 1, Length[kritnul]}];
Do[Print["V bodě " , tab2[[i]][[1]], " je ",
  If[Abs[tab1[[i]][[3]]] == ∞,
   If[tab2[[i]][[3]] < 0, "lokální maximum",
    If[tab2[[i]][[3]] > 0, "lokální minimum",
     "nerozhodnuto"]],
   If[tab1[[i]][[3]] < 0, "lokální maximum",
    If[tab1[[i]][[3]] > 0, "lokální minimum",
     "nerozhodnuto"]]]], {i, 1, Length[kritnul]}]
```

V bodě {0., −1.} je lokální minimum

V bodě {0., 1.} je lokální minimum

V bodě {−1., 0.} je lokální minimum

V bodě {1., 0.} je lokální minimum

V bodě {−0.518754, −0.854924} je lokální maximum

V bodě {0.518754, −0.854924} je lokální maximum

V bodě {−0.956289, 0.292424} je lokální maximum

V bodě {0.956289, 0.292424} je lokální maximum

Pro kontrolu lze vytisknout tabulky hodnot v kritických bodech.

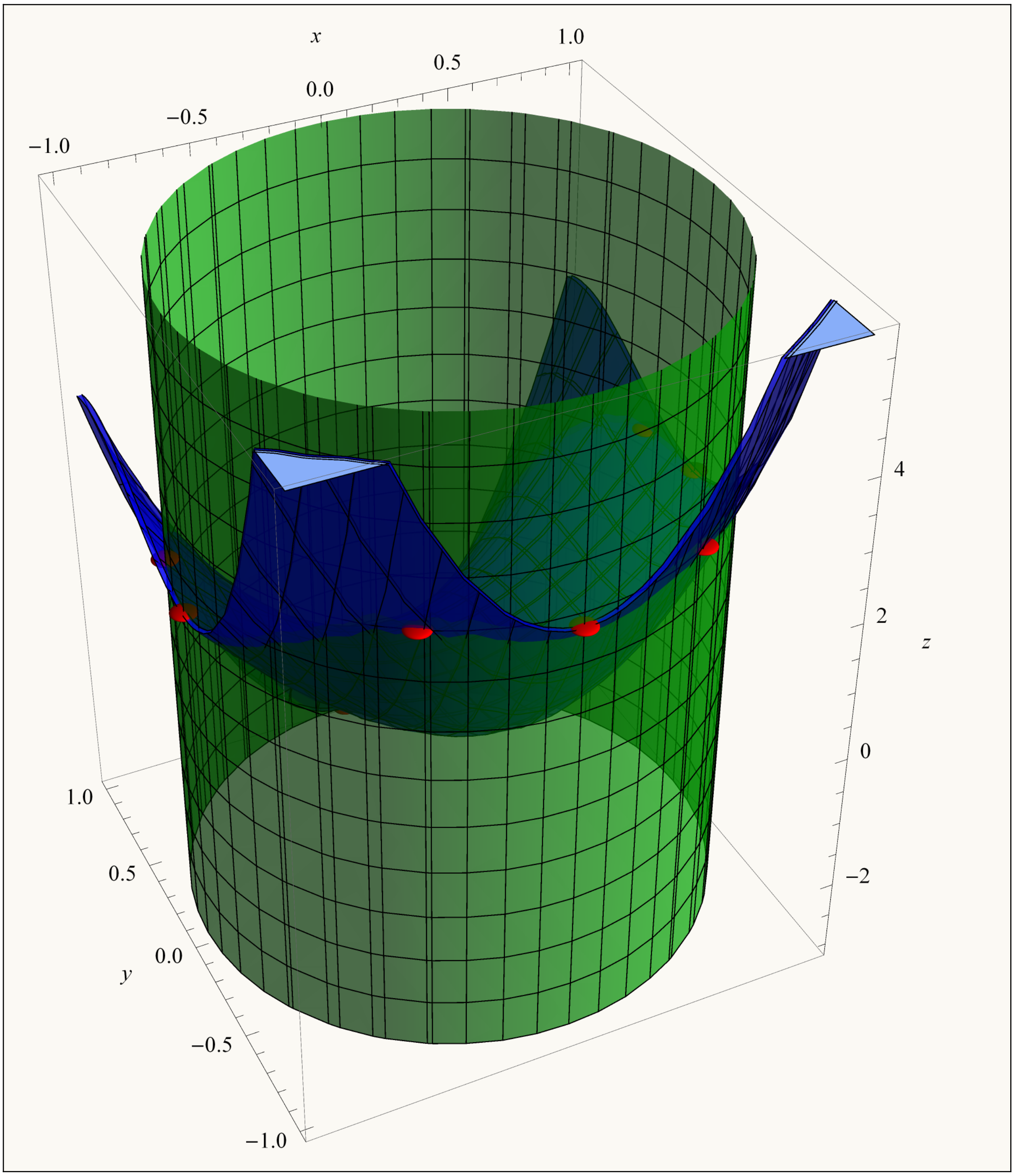
```
Labeled[
 Grid[tab1, Frame → All, ItemSize → All],
 Text[Row[{Style["(x,y)", Red], Style["f(x,y)", Red],
    Style["Kvadr.forma pro gy≠0", Red]}, Spacer[100]]], Top]

Labeled[
 Grid[tab2, Frame → All, ItemSize → All],
 Text[Row[{Style["(x,y)", Red], Style["f(x,y)", Red],
    Style["Kvadr.forma pro gx≠0", Red]}, Spacer[95]]],
 Top]
```

| (x,y) | f(x,y) | Kvadr.forma pro $g_y$≠0 |
|---|---|---|
| {0., −1.} | 3. | 3. |
| {0., 1.} | −3. | 21. |
| {−1., 0.} | 2. | ComplexInfinity |
| {1., 0.} | 2. | ComplexInfinity |
| {−0.518754, −0.854924} | 3.19954 | −5.77843 |
| {0.518754, −0.854924} | 3.19954 | −5.77843 |
| {−0.956289, 0.292424} | 2.06676 | −57.4091 |
| {0.956289, 0.292424} | 2.06676 | −57.4091 |

| (x,y) | f(x,y) | Kvadr.forma pro $g_x$≠0 |
|---|---|---|
| {0., −1.} | 3. | ComplexInfinity |
| {0., 1.} | −3. | ComplexInfinity |
| {−1., 0.} | 2. | 4. |
| {1., 0.} | 2. | 4. |
| {−0.518754, −0.854924} | 3.19954 | −15.6943 |
| {0.518754, −0.854924} | 3.19954 | −15.6943 |
| {−0.956289, 0.292424} | 2.06676 | −5.36819 |
| {0.956289, 0.292424} | 2.06676 | −5.36819 |

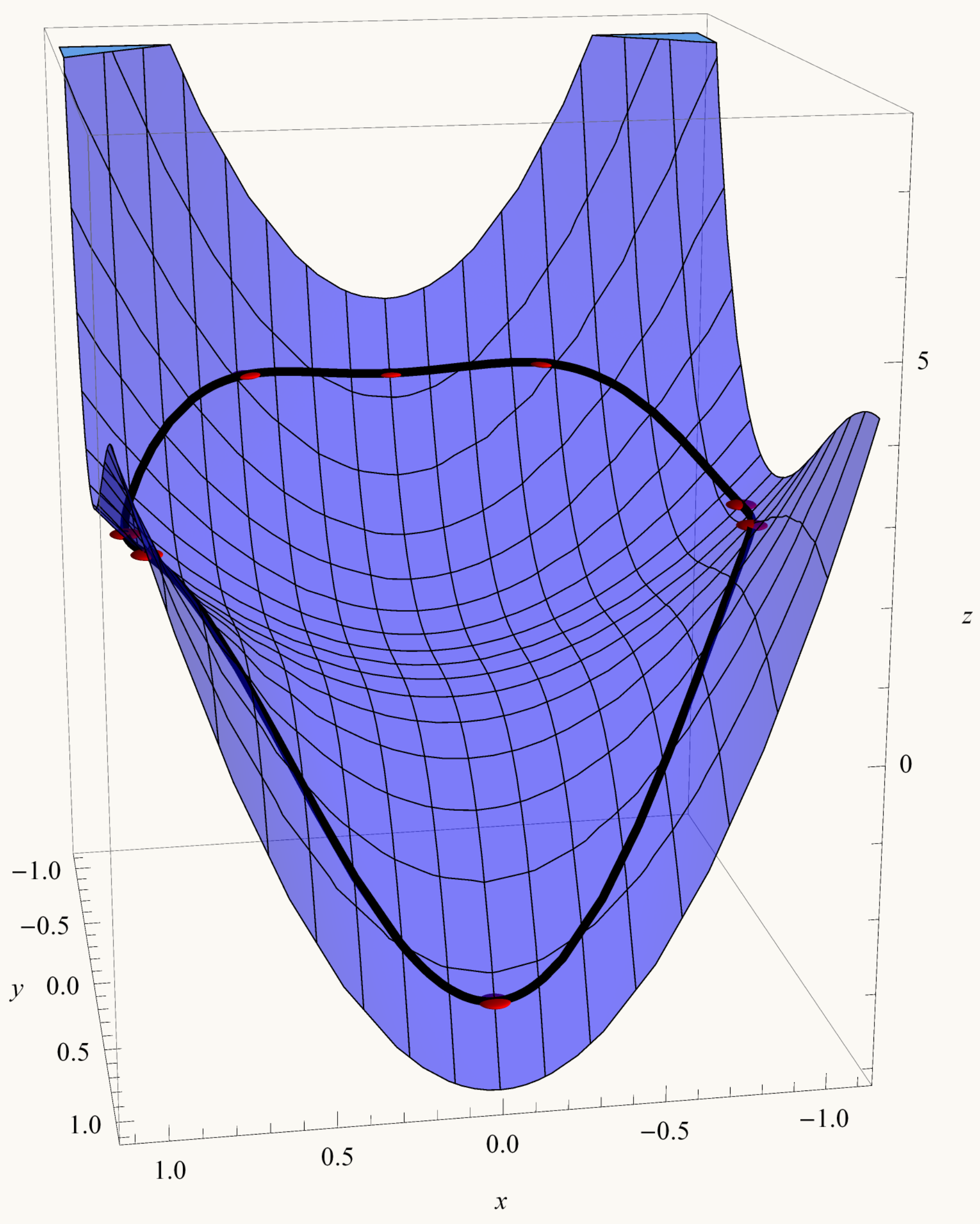
Pro kontrolu je vhodné nakreslit příslušný graf. Obecně lze nakreslit hodnoty funkce f za podmínky g=0 jen jako průnik dvou ploch. Červeně jsou vyznačeny kritické body.

```
p1 = Plot3D[f[x, y], {x, -1, 1}, {y, -1, 1},
   AxesLabel → {x, y, z},
   PlotStyle → {Opacity[.6], Blue, Thickness[.02]}];
p2 = ContourPlot3D[g[x, y] == 0, {x, -1, 1}, {y, -1, 1},
   {z, -3, 8}, ContourStyle → {Opacity[.7], Green}];
body = Table[{x, y, f[x, y]} /. kritnul[[i]],
   {i, 1, Length[kritnul]}];
extr =
  Graphics3D[
   {Red, Table[Sphere[body[[j]], .05],
     {j, 1, Length[kritnul]}]}] ;
Show[p1, p2, extr, BoxRatios → {1, 1, 1.3}]
```




Pokud máme k dispozici parametrické vyjádření množiny g=0, pak lze snadno vykreslit množinu hodnot f na této množině. Nutno však dodat, že v tomto případě je často snadnější počítat extrémy bez Lagrangeovývh multiplikátorů dosazením parametrického vyjádření do f.

```
p1 = Plot3D[f[x, y], {x, -1.1, 1.1}, {y, -1.1, 1.1},
    AxesLabel → {x, y, z}, PlotStyle → {Opacity[.5], Blue}];

p3 = ParametricPlot3D[{Cos[u], Sin[u], f[Cos[u], Sin[u]]},
    {u, 0, 2 Pi}, AxesLabel → {x, y, z},
    PlotStyle → {Thickness[.01]}];
Show[{p1, p3, extr}, BoxRatios → {1, 1, 1.3}]
```




V předchozím případě lze množinu bodů vyhovujících podmínce popsat parametricky. Pak můžeme parametry dosadit do dané funkce a řešit úlohu nalezení extrémů funkce jedné proměnné. Porovnejte oba výsledky.

```
f[x_, y_] := 2*x^2 - 3*y^3 + 4*x^2*y^2;
g[u_] := Cos[u]; h[u_] := Sin[u];
ff[u_] = f[x, y] /. {x → g[u], y → h[u]};
nulder =
  Sort[Solve[{Derivative[1][ff][u] == 0, 0 ≤ u, u ≤ 2 Pi}, u] //
    N];
krit = u /. nulder;
tab =
  Table[
   {{Round[g[u] * 10^7] / 10^7 // N,
      Round[h[u] * 10^7] / 10^7 // N}, ff[u] // N} /.
    nulder[[i]], {i, 1, Length[krit]}];
Labeled[
 Grid[tab, Frame → All, ItemSize → All],
 Text[Row[{Style["(x,y)", Red], Style["f(x,y)", Red]},
   Spacer[100]]], Top]
```

| (x,y) | f(x,y) |
|---|---|
| {1., 0.} | 2. |
| {0.956289, 0.292424} | 2.06676 |
| {0., 1.} | −3. |
| {−0.956289, 0.292424} | 2.06676 |
| {−1., 0.} | 2. |
| {−0.518754, −0.854924} | 3.19954 |
| {0., −1.} | 3. |
| {0.518754, −0.854924} | 3.19954 |
| {1., 0.} | 2. |

# *Funkce tří proměnných s jednou podmínkou*

Následuje obecnější procedura.která ale nevyčerpává všechny případy, kdy např. podmínku nelze vyřešit vzhledem k proměnné z (tj., kdy parciální derivace podmínky podle z je 0). Pak je nutné vzít jinou proměnnou

```
Needs["VectorAnalysis`"]
Clear[f, g, a]; f[{x_, y_, z_}] := x y + 2 y z - 2 z^2;
g[{x_, y_, z_}] := x^2 + y^2 + z^2 - 1;
```

```
(* Mathematica nezná gradienty vyšších než 3
 proměnných (lze je snadno dodefinovat) a tak vezmeme
 v dalším postupu jen 3 proměnné a proměnnou "a"
 budeme přidávat, kde je třeba. *)
F[{x, y, z}] := f[{x, y, z}] - a g[{x, y, z}];
 (* Nyní vypočteme kritické body spolu s příslušnými
 hodnotami multiplikátoru *)
kritnul =
  Solve[{Grad[F[{x, y, z}], Cartesian[x, y, z]] == {0, 0, 0},
     g[{x, y, z}] == 0}, {x, y, z, a}, WorkingPrecision → 7] //
   N;
kritbody = Table[{x, y, z} /. kritnul[[i]],
   {i, 1, Length[kritnul], 1}];
 (* Mathematica neumí požít Derivative v obecnějším
 smyslu
 (v našem případě pro f=
   ag pomocí obecně daného vektoru s proměnnou) a
 tak si musíme vypomoci definicí. Pomocí ní spočteme
 matice druhých parciálních derivací *)
der[list_] := Derivative[list][f][{x, y, z}] -
   a Derivative[list][g][{x, y, z}];
m = Table[Grad[der[UnitVector[3, i]], Cartesian[x, y, z]],
   {i, 1, 3, 1}];

(* Nyní dosadíme do kvadratické formy pro F jednu
   proměnnou z rovnice grad(g)..=
 0 (označenou jako varz). Dostaneme kvadratickou
  formu (kvadr) dvou proměnných x,
y a vypočteme její vlastní čísla v kritických bodech. *)
varz =
  z /.
   NSolve[Grad[g[{x, y, z}], Cartesian[x, y, z]].{1, 1, 1} == 0,
      z][[1]][[1]];
kvadr = Simplify[{x, y, varz}.m.{x, y, varz}];
mm =
  Normal[CoefficientArrays[kvadr, {x, y},
```

```
      "Symmetric" → True]][[3]];
vlastnivektory = Table[Eigenvalues[mm /. kritnul[[i]]],
    {i, 1, Length[kritnul]}];
(* Použijeme upravenou rozhodovací funkci z extrémů
 funkce tří proměnných na otevřené množině a vypíšeme
 výsledky. *)
dec[{a_, b_}, n_] :=
   If[a b == 0, Print["V bodě ", kritbody[[n]],
     " nelze o extrému rozhodnout"],
    If[a + b == Abs[a] + Abs[b],
     Print["V bodě ", kritbody[[n]],
      " je lokální minimum s hodnotou ", f[kritbody[[n]]]],
        If[-a - b == Abs[a] + Abs[b],
      Print["V bodě ", kritbody[[n]],
        " je lokální maximum s hodnotou ",
        f[kritbody[[n]]]],
      Print["V bodě ", kritbody[[n]], " není extrém"]]]];
(*Výpis zjištěných extrémů*)
Table[dec[vlastnivektory[[i]], i], {i, 1, Length[kritnul]}];
```

V bodě {−0.547679, −0.785189, −0.289009}
je lokální maximum s hodnotou 0.716832

V bodě {0.83276, −0.478122, −0.279125} není extrém

V bodě {0.0809845, −0.393546, 0.915731}
je lokální minimum s hodnotou −2.42976

V bodě {−0.0809845, 0.393546, −0.915731}
je lokální minimum s hodnotou −2.42976

V bodě {−0.83276, 0.478122, 0.279125} není extrém

V bodě {0.547679, 0.785189, 0.289009} je lokální maximum s hodnotou 0.716832

# *Funkce tří proměnných s dvěma podmínkami*

Upravíme předchozí postup pro dvě podmínky. Hledáme lokální extrémy funkce f za podmínek g=0,h=0. Zjednodušíme si postup použitím nestandardního balíčku Miscellaneous`RealOnly`.

Procedura je upravena jen pro jednu možnost  nenulového determinantu v matici (grad(g),grad(h)). V mnoha případech program soustavu rovnic nevyřeší nebo nenajde všechna řešení.

```
Needs["VectorAnalysis`"]
<< Miscellaneous`RealOnly`
Clear[f, g, a]; f[{x_, y_, z_}] := x y + 2 y z  - 2 z^2;
g[{x_, y_, z_}] := x^2 + y^2 + z^2 - 1;
h[{x_, y_, z_}] := x - 3 y + 4 z^2 + 1;
```

— *General::obspkg :*

*Miscellaneous`RealOnly` is now obsolete. The legacy version being loaded may conflict with current Mathematica functionality. See the Compatibility Guide for updating information.* ≫

```
(* Mathematica nezná gradienty vyšších než 3
 proměnných (lze je snadno dodefinovat) a tak vezmeme
 v dalším postupu jen 3 proměnné a proměnnou "a"
 budeme přidávat, kde je třeba. *)
F[{x, y, z}] := f[{x, y, z}] - a g[{x, y, z}] - b h[{x, y, z}];
 (* Nyní vypočteme kritické body spolu s příslušnými
 hodnotami multiplikátoru *)
kritnul =
   Solve[{Grad[F[{x, y, z}], Cartesian[x, y, z]] == {0, 0, 0},
      g[{x, y, z}] == 0, h[{x, y, z}] == 0}, {x, y, z, a, b},
     WorkingPrecision → 7] // N;
kritbody = Table[{x, y, z} /. kritnul[[i]],
    {i, 1, Length[kritnul] - 1, 1}];
 (* Spočteme matici druhých parciálních derivací *)
der[list_] := Derivative[list][f][{x, y, z}] -
    a Derivative[list][g][{x, y, z}] -
    b Derivative[list][h][{x, y, z}];
m = Table[Grad[der[UnitVector[3, i]], Cartesian[x, y, z]],
    {i, 1, 3, 1}];
 (* Nyní dosadíme do kvadratické formy pro F derivace
   proměnných y a z jako funkce x z rovnic g=0,
h=0 (označené jako vary, varz). Dostaneme kvadratickou
   formu (kvadr)
 jedné proměnné x a z ní snadno rozhodneme o extrémech
  (po dosazení hodnot v kritických bodech dostaneme
     seznam mm a podle parity jeho čísel se rozhodne). *)
varzy =
   Solve[{Grad[g[{x, y, z}], Cartesian[x, y, z]].{1, 1, 1} == 0,
     Grad[h[{x, y, z}], Cartesian[x, y, z]].{1, 1, 1} == 0},
    {y, z}];
vary = y /. varzy[[1]][[1]];
varz = z /. varzy[[1]][[2]];
kvadr = Simplify[{x, vary, varz}.m.{x, vary, varz}];
mm = Table[kvadr /. kritnul[[i]],
    {i, 1, Length[kritnul] - 1, 1}];
decision[u_] := If[u > 0, "lokální minimum",
    If[u < 0, "lokální maximum", If[u == 0, "nerozhodnuto"]]];
Do[Print["V bodě " , kritbody[[i]], " je ",
    decision[mm[[i]]]], {i, 1, Length[kritnul] - 1, 1}];
```

V bodě $\{-0.305436,\ 0.729977,\ -0.611426\}$ je lokální minimum

V bodě $\{-1.,\ 0.,\ 0.\}$ je lokální maximum

V bodě $\{0.642127,\ 0.692159,\ 0.329527\}$ je lokální maximum

V bodě $\{-0.758719,\ 0.415753,\ 0.501492\}$ je lokální minimum