

# Srovnání evolučních algoritmů pomocí vybraných testovacích funkcí

Bc. Michal Andresík



# Obsah

1.	Postup řešení a použité nástroje .....	3
2.	Vybrané testovací funkce.....	4
2.1.	Ackleyho funkce.....	4
2.2.	Alpine01 funkce .....	5
2.3.	Alpine02 funkce .....	6
2.4.	Bohachevskyho funkce .....	7
2.5.	CosineMixture funkce.....	8
2.6.	DeflectedCorrugatedSpring funkce .....	9
2.7.	Dixon-Price funkce.....	10
2.8.	Griewankova funkce .....	11
2.9.	InvertedCosineWave funkce .....	12
2.10.	Levyho funkce.....	13
2.11.	Michalewiczova funkce s parametrem $m = 1$ .....	14
2.12.	Michalewiczova funkce s parametrem $m = 10$ .....	15
2.13.	Mishra07 funkce .....	16
2.14.	Mishra11 funkce .....	17
2.15.	Patologická funkce .....	18
2.16.	Rastriginova funkce .....	19
2.17.	Rosenbrockovo sedlo.....	20
2.18.	Salomonova funkce .....	21
2.19.	Schwefel22 funkce .....	22
2.20.	Schwefel26 function .....	23
2.21.	Sinová obálka.....	24
2.22.	Roztažená sinová obálka.....	25
2.23.	StyblinskiTang funkce .....	26
2.24.	Trigonometrická funkce .....	27
2.25.	Vincentova funkce.....	28
3.	Výsledky a pořadí algoritmů .....	29
3.1.	Dimenze 2 .....	29
3.2.	Dimenze 10.....	31
3.3.	Dimenze 30.....	33
4.	Friedmanův rank test.....	35

# 1. Postup řešení a použité nástroje

Tato seminární práce hodnotí 5 zadaných evolučních algoritmů na 25 vybraných testovacích funkcích.

Vybrané funkce pro hodnocení algoritmů jsou představeny v kapitole 2. U každé uvádíme vzorec, 2D a 3D graf pro ukázkou chování funkce a zdroj. U funkcí, které jsou definovány pro 2 a více proměnných nebo je jejich průběh pro jednu proměnnou zdegenerován (např. Mishra11, kde je každý bod globálním minimem), 2D graf neuvádíme.

V kapitole 3 vyšetříme výsledky algoritmů v dimenzích 2, 10 a 30 a uvedeme nejlepší dosažená minima, společně také s pořadím algoritmů pro každou funkci. Pomocí průměrného umístění pro každý algoritmus vypíšeme 5 funkcí, které se od očekávaného pořadí odlišují nejvíce. Tuto skutečnost se pokusíme zdůvodnit.

V kapitole 4 otestujeme, zda se jednotlivé algoritmy na námi zvolených funkcích umísťují konzistentně ve stejném nebo velmi podobném pořadí.

Všechny nástroje můžeme nalézt na <https://github.com/mandresik/Evolutionary-computation>.

Ve složce `benchmark_functions` máme grafy funkcí, společně s matlab skripty, pomocí kterých jsou grafy vykresleny. Zdroje k jednotlivým funkcím jsou uvedeny v kapitole 2.

V souboru `benchmark_functions.py` jsou tyto funkce implementovány v pythonu.

V souboru `algorithms_fes.py` jsou implementovány zadané algoritmy. Zdrojem byly prezentace k přednáškám a drobné dohledávání na internetu, zejména různých forem pseudokódů.

V souboru `main.py` jsou dané algoritmy na uvedených funkcích vyšetřeny. Každé vyšetření se provádí 30krát. Průměrné hodnoty nejlepších minim jsou uloženy do .csv souboru. Z těchto nalezených hodnot je vytvořeno umístění algoritmů pro každou funkci, a to je taktéž uloženo jako .csv soubor.

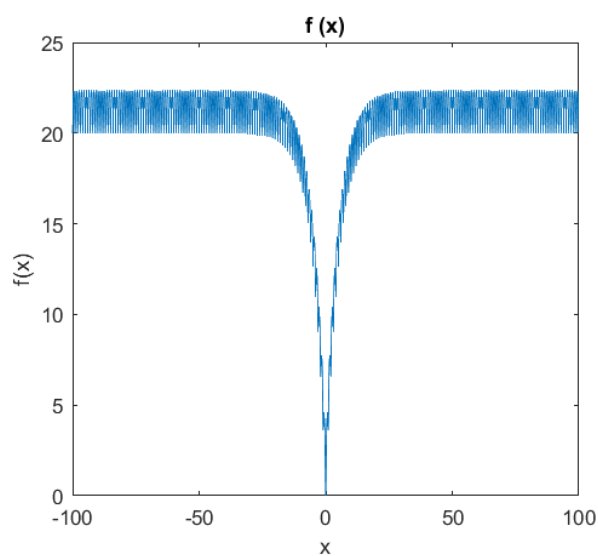
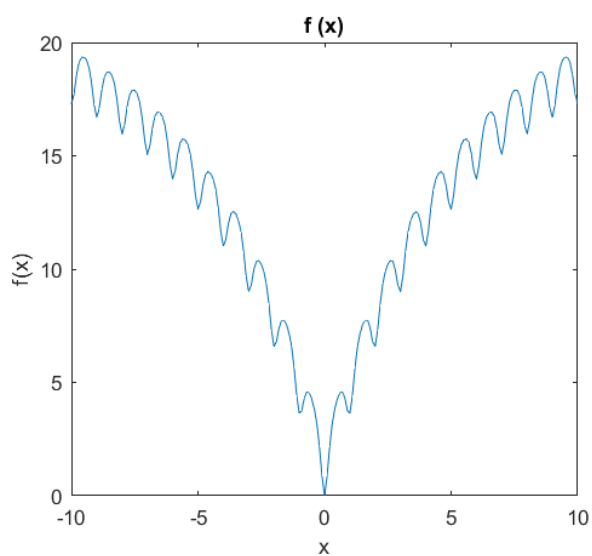
Ve složce `results` pak můžeme nalézt všechny tyto uložené soubory. Dále zde máme soubor `friedman.py`, pomocí kterého počítáme testovací hodnotu Friedmanova rank testu.

## 2. Vybrané testovací funkce

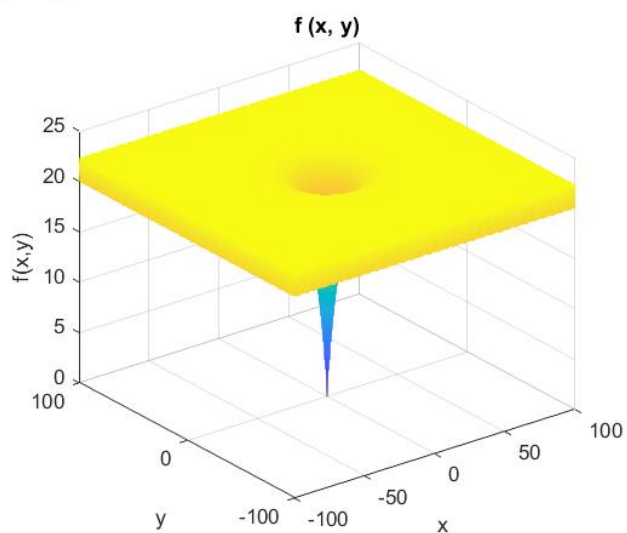
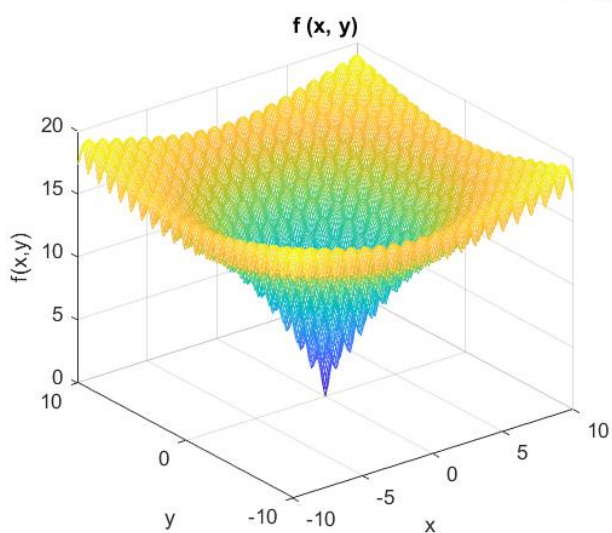
### 2.1. Ackleyho funkce

$$f(\mathbf{x}) = -20e^{-0.2\sqrt{\frac{1}{n}\sum_{i=1}^n x_i^2}} - e^{\frac{1}{n}\sum_{i=1}^n \cos(2\pi x_i)} + 20 + e$$

Ackleyho funkce



Ackleyho funkce

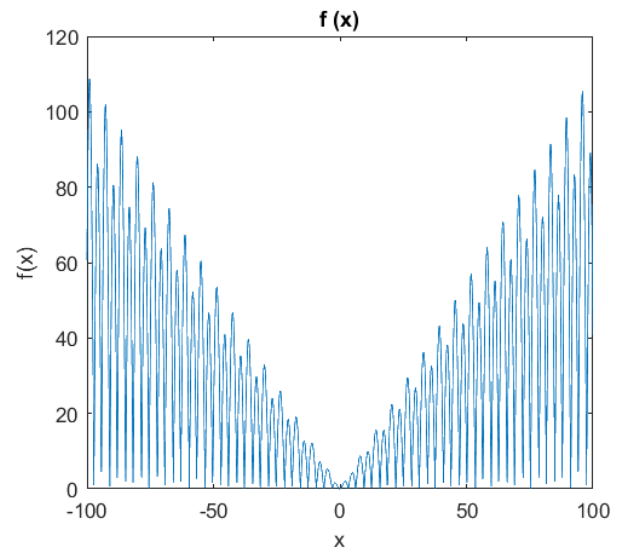
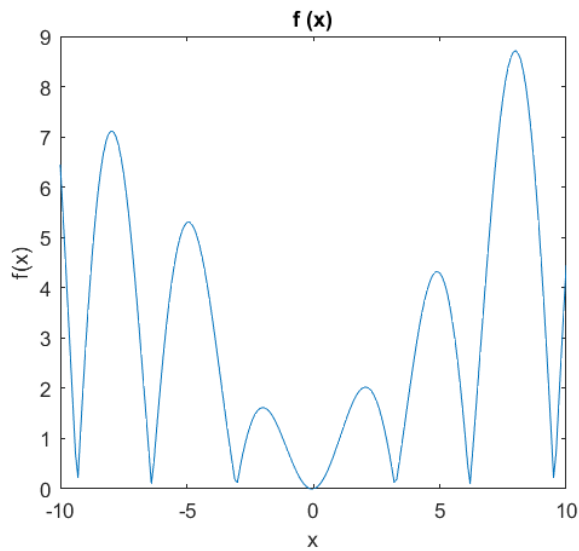


Zdroj: [https://infinity77.net/global\\_optimization/test\\_functions\\_nd\\_A.html#go\\_benchmark.Ackley](https://infinity77.net/global_optimization/test_functions_nd_A.html#go_benchmark.Ackley)

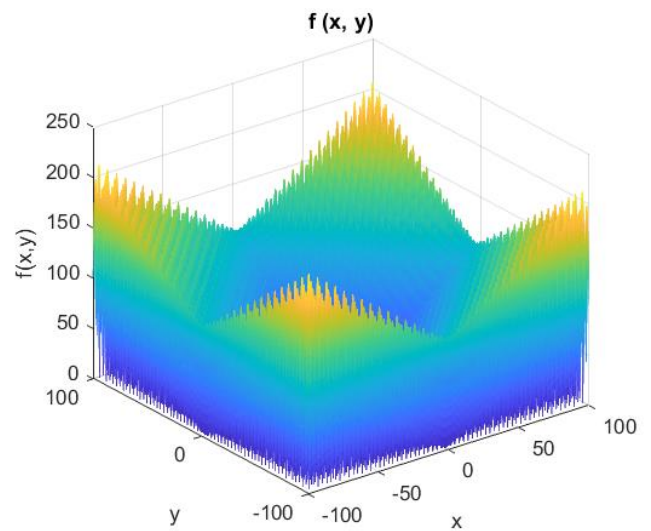
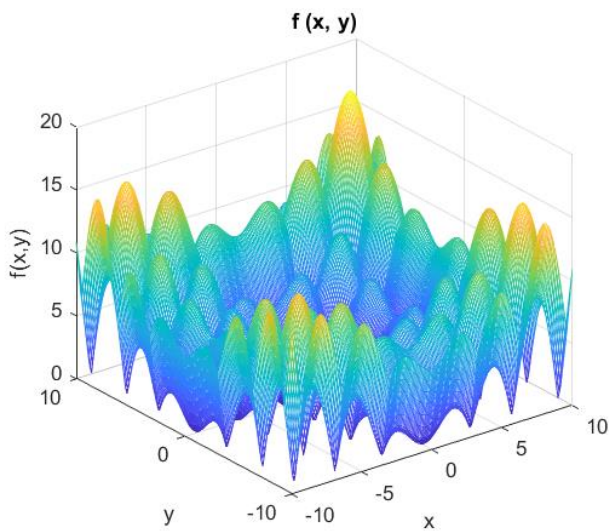
## 2.2. Alpine01 funkcce

$$f(\mathbf{x}) = \sum_{i=1}^n |x_i \sin(x_i) + 0.1x_i|$$

Alpine01 funkcce



Alpine01 funkcce

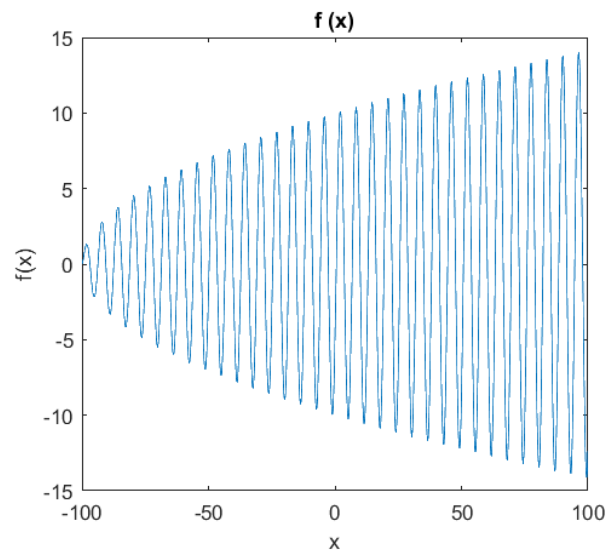
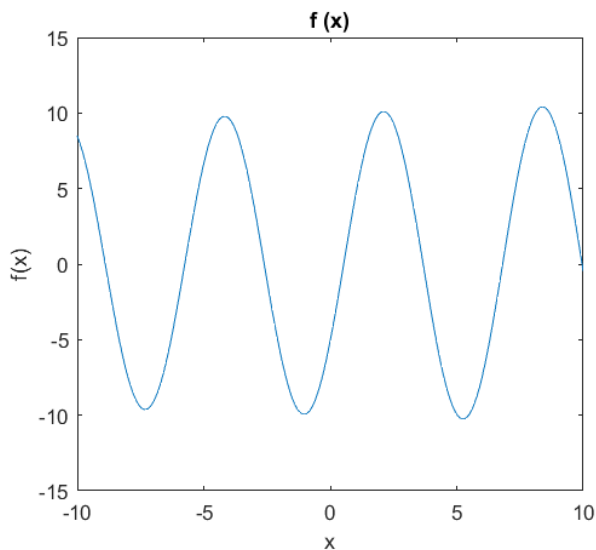


Zdroj: [https://infinity77.net/global\\_optimization/test\\_functions\\_nd\\_A.html#go\\_benchmark.Alpine01](https://infinity77.net/global_optimization/test_functions_nd_A.html#go_benchmark.Alpine01)

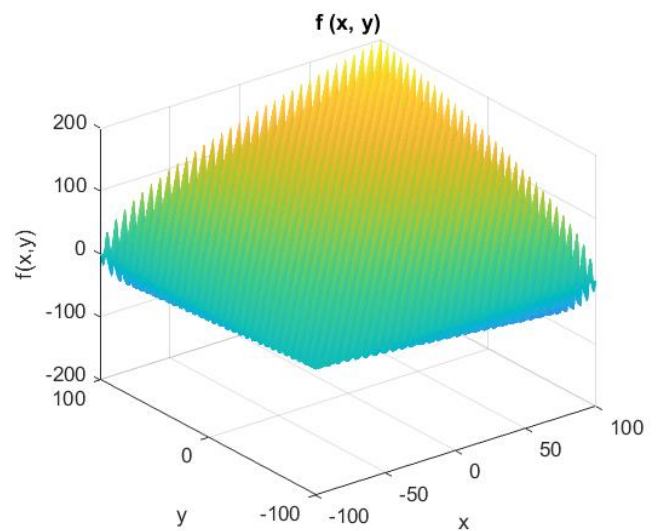
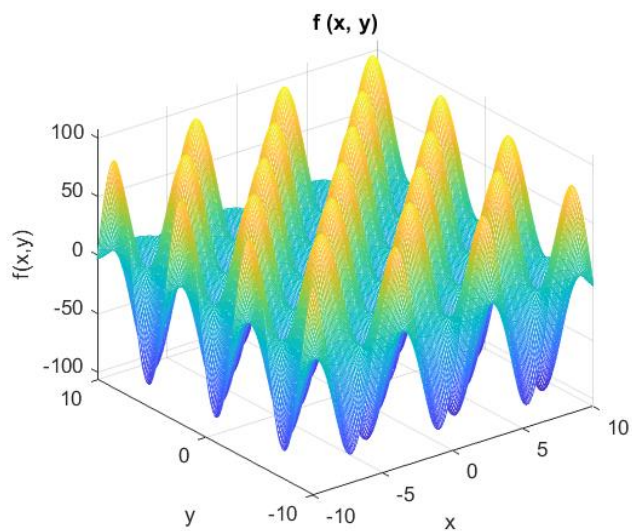
### 2.3. Alpine02 funkce

$$f(\mathbf{x}) = \prod_{i=1}^n \sqrt{x_i + 100} \sin(x_i + 100)$$

Alpine02 funkce



Alpine02 funkce



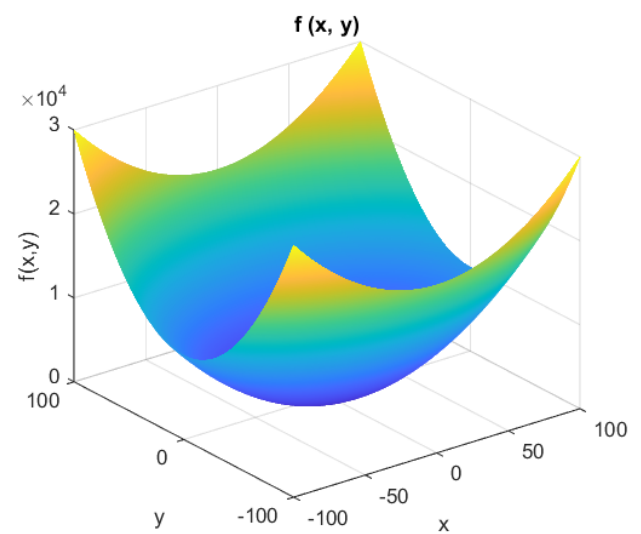
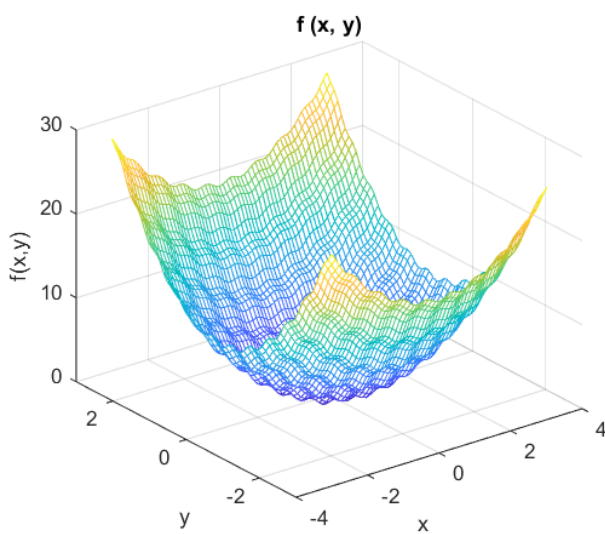
Pozn.: Definiční obor Alpine02 funkce ze zdroje byl rozšířen z  $D(f) = \langle 0, \infty \rangle$  na  $D(f) = \langle -100, \infty \rangle$ .

Zdroj: [https://infinity77.net/global\\_optimization/test\\_functions\\_nd\\_A.html#go\\_benchmark.Alpine02](https://infinity77.net/global_optimization/test_functions_nd_A.html#go_benchmark.Alpine02)

## 2.4. Bohachevskyho funkce

$$f(\mathbf{x}) = \sum_{i=1}^{n-1} [x_i^2 + 2x_{i+1}^2 - 0.3 \cos(3\pi x_i) - 0.4 \cos(4\pi x_{i+1}) + 0.7]$$

Bohachevskyho funkce



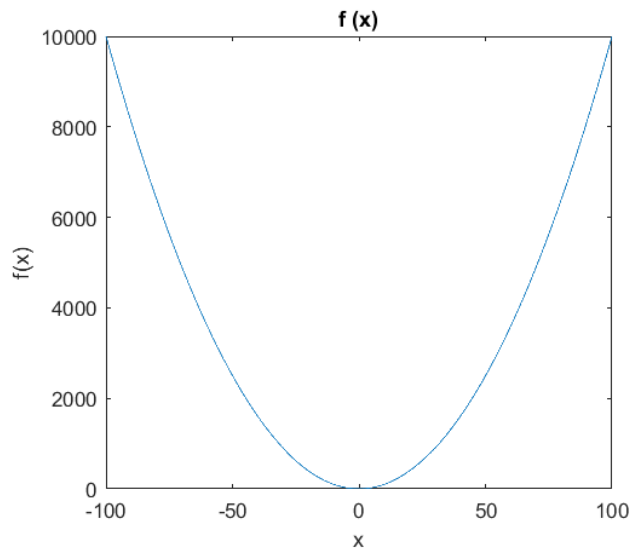
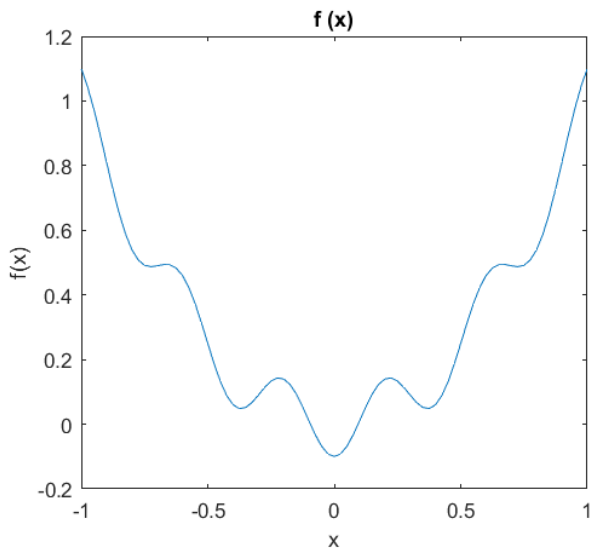
Pozn.: Funkce je definována pro 2 a více proměnné. 2D graf pro jednu proměnnou není možné vytvořit.

Zdroj: [https://infinity77.net/global\\_optimization/test\\_functions\\_nd\\_B.html#go\\_benchmark.Bohachevsky](https://infinity77.net/global_optimization/test_functions_nd_B.html#go_benchmark.Bohachevsky)

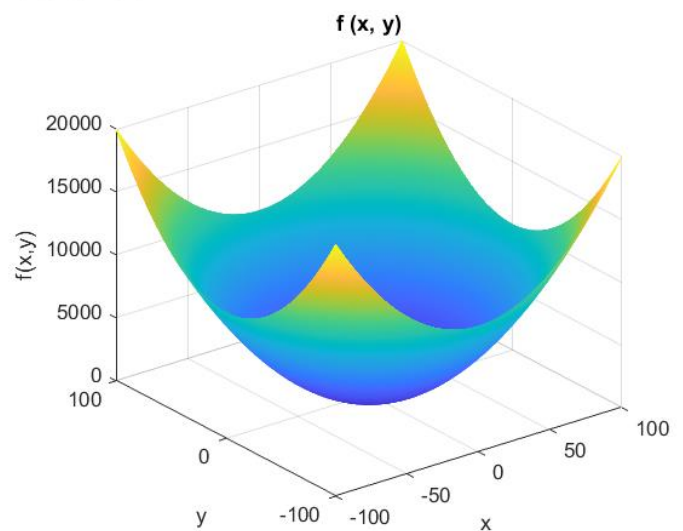
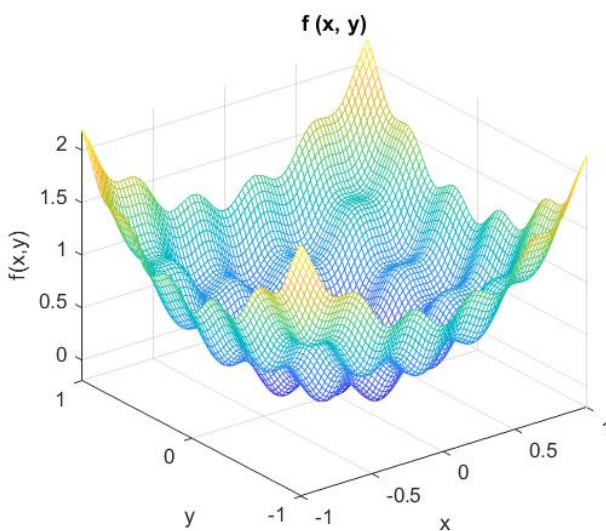
## 2.5. CosineMixture funkce

$$f(\mathbf{x}) = -0.1 \sum_{i=1}^n \cos(5\pi x_i) + \sum_{i=1}^n x_i^2$$

CosineMixture funkce



CosineMixture funkce



Pozn.: Uvedená funkce ze zdroje je pozměněna. Před druhou sumací je obrácené znaménko a jedná se tak o minimalizační úlohu.

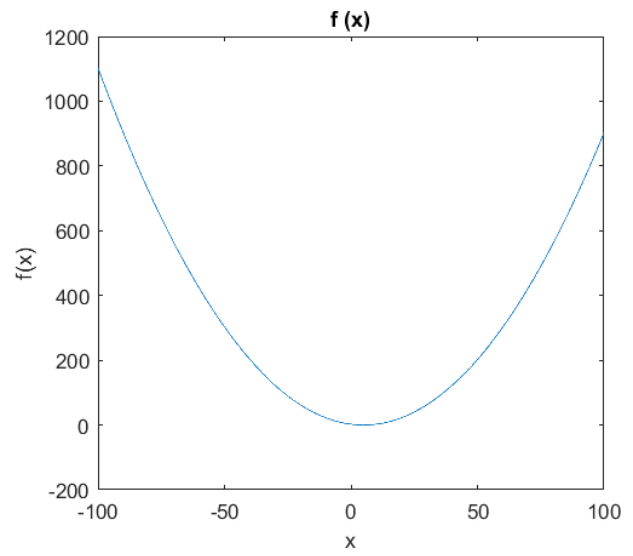
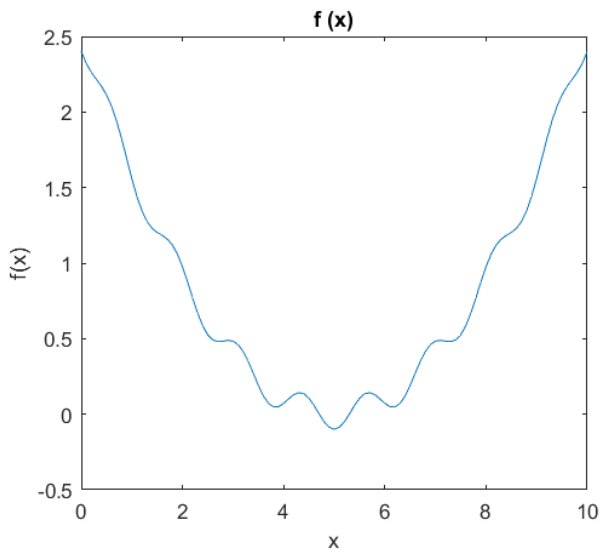
Zdroj: [https://infinity77.net/global\\_optimization/test\\_functions\\_nd\\_C.html#go\\_benchmark.CosineMixture](https://infinity77.net/global_optimization/test_functions_nd_C.html#go_benchmark.CosineMixture)



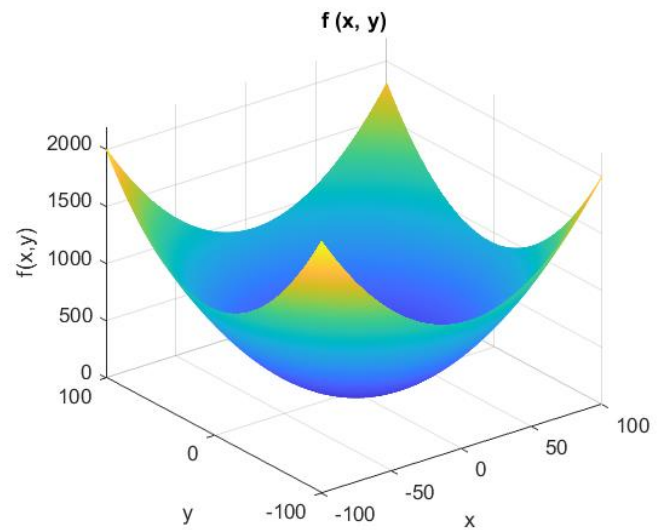
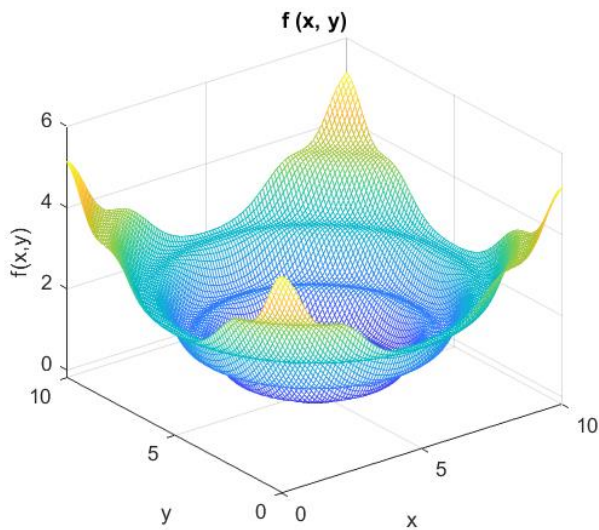
## 2.6. DeflectedCorrugatedSpring funkcce

$$f(\mathbf{x}) = 0.1 \sum_{i=1}^n \left[ (x_i - \alpha)^2 - \cos \left( K \sqrt{\sum_{i=1}^n (x_i - \alpha)^2} \right) \right], \quad \alpha = K = 5$$

DeflectedCorrugatedSpring funkcce



DeflectedCorrugatedSpring funkcce



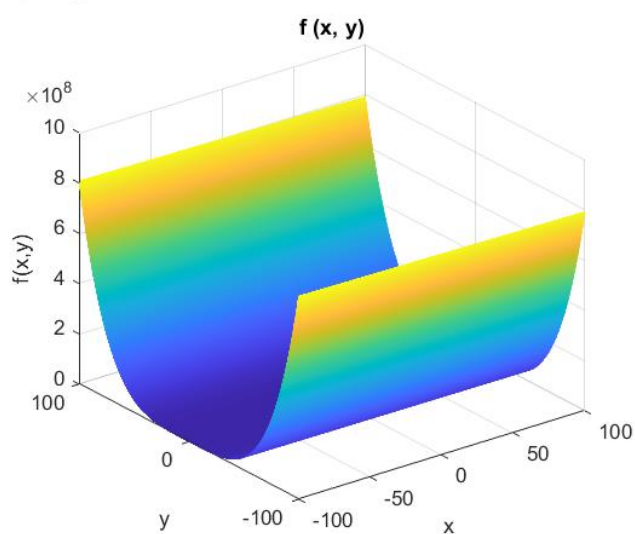
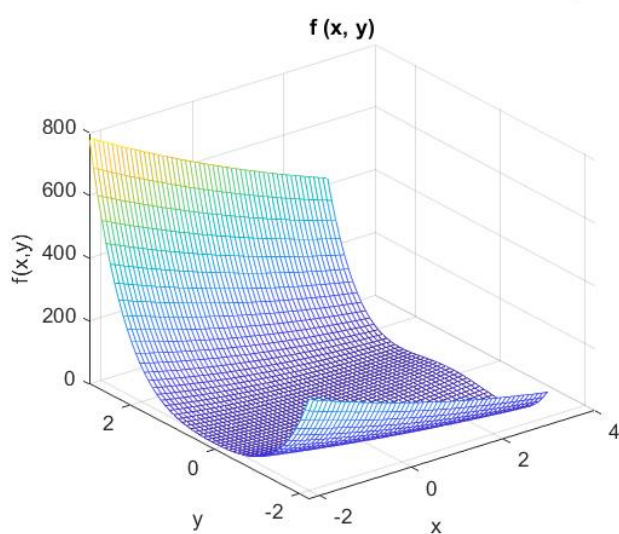
Zdroj:

[https://infinity77.net/global\\_optimization/test\\_functions\\_nd\\_D.html#go\\_benchmark.DeflectedCorrugatedSpring](https://infinity77.net/global_optimization/test_functions_nd_D.html#go_benchmark.DeflectedCorrugatedSpring)

## 2.7. Dixon-Price funkce

$$f(\mathbf{x}) = (x_1 - 1)^2 + \sum_{i=2}^n i(2x_i^2 - x_{i-1})^2$$

Dixon-Price funkce



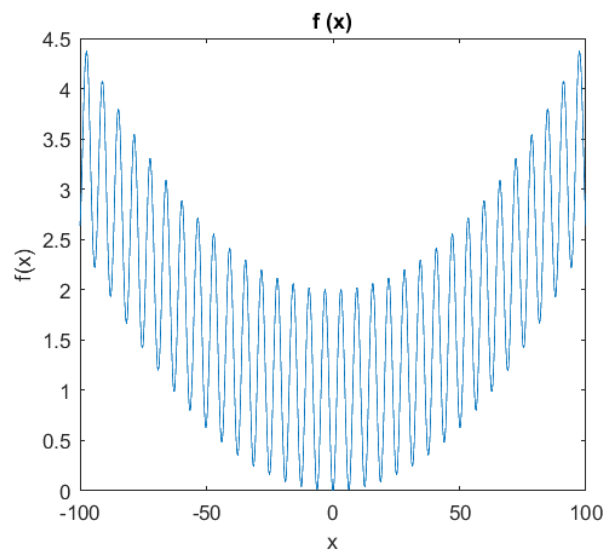
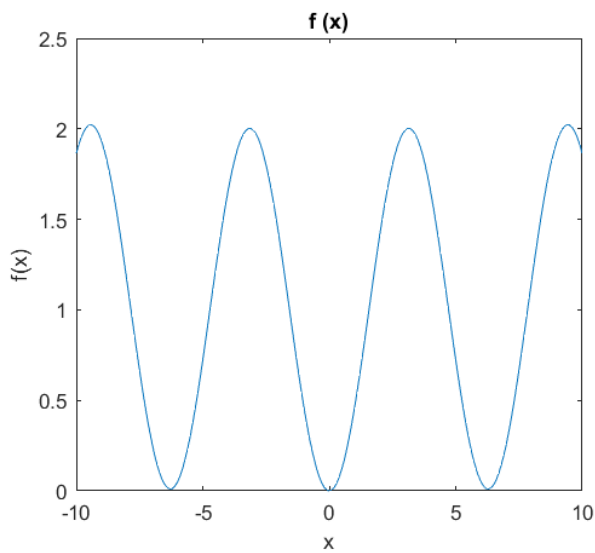
Pozn.: Funkce obsahuje sumaci, která k sobě vztahuje každé 2 po sobě jdoucí proměnné. Funkce pro jednu proměnnou nevystihuje chování, ke kterému dochází u 2 a více proměnných. 2D graf je tedy vynechán.

Zdroj: <https://www.sfu.ca/~ssurjano/dixonpr.html>

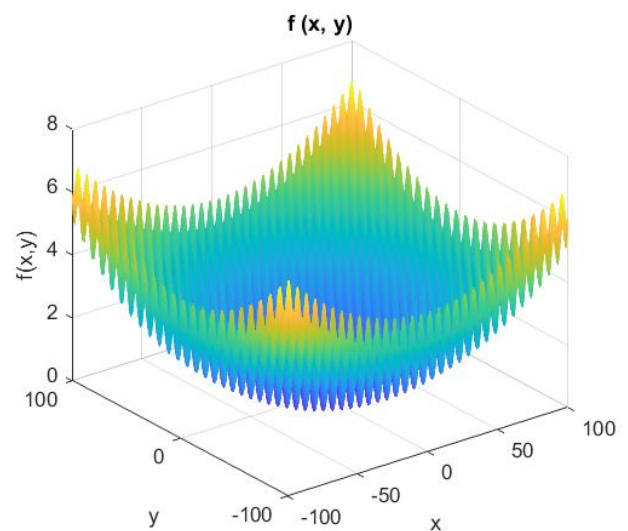
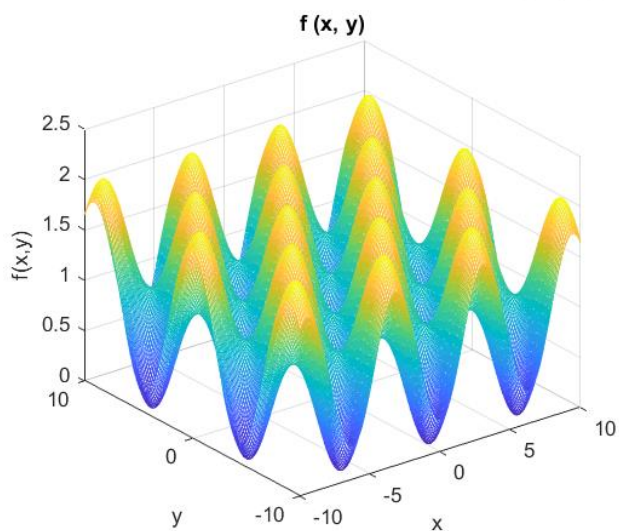
## 2.8. Griewankova funkce

$$f(\mathbf{x}) = 1 + \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right)$$

Griewankova funkce



Griewankova funkce

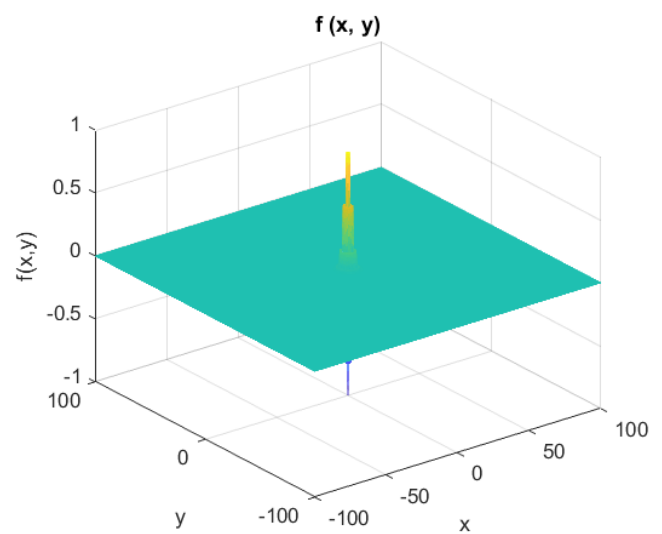
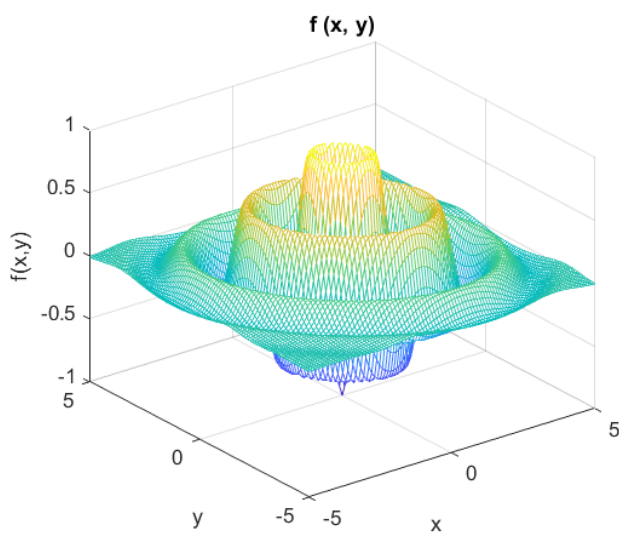


Zdroj: prezentace k přednáškám z matematické informatiky od prof. Šenkeříka

## 2.9. InvertedCosineWave funkce

$$f(\mathbf{x}) = - \sum_{i=1}^{n-1} e^{-\frac{x_i^2 + x_{i+1}^2 + 0.5x_i x_{i+1}}{8}} \cos\left(4\sqrt{x_i^2 + x_{i+1}^2 + 0.5x_i x_{i+1}}\right)$$

InvertedCosineWave funkce



Pozn.: Funkce je definována pro 2 a více proměnných. 2D graf pro jednu proměnnou není možné vytvořit.

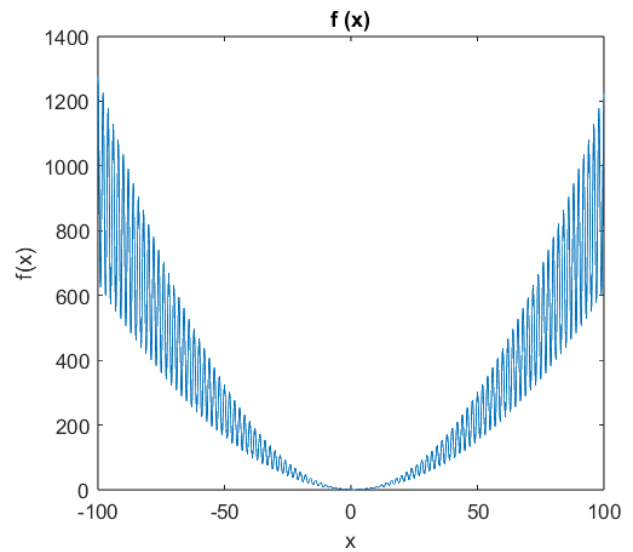
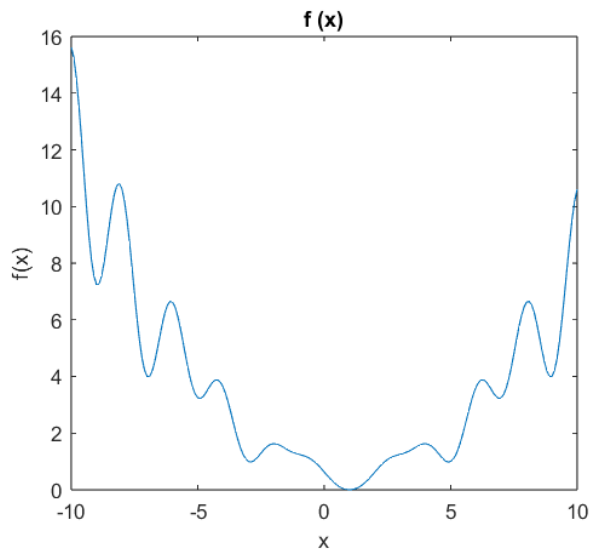
Zdroj: prezentace k přednáškám z matematické informatiky od prof. Šenkeříka

## 2.10. Levyho funkce

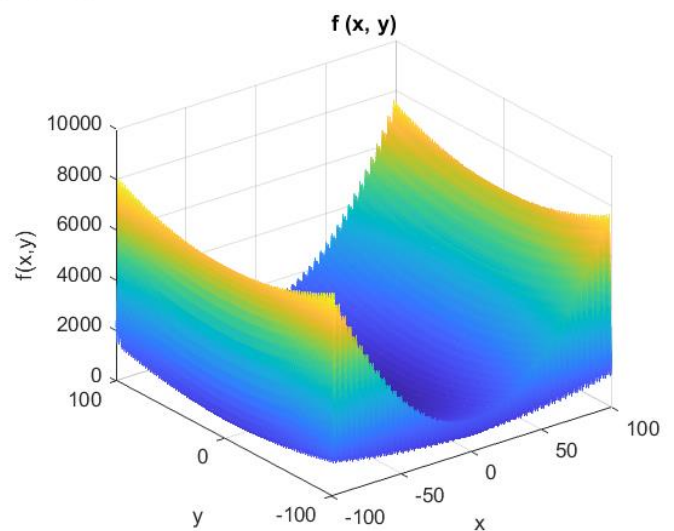
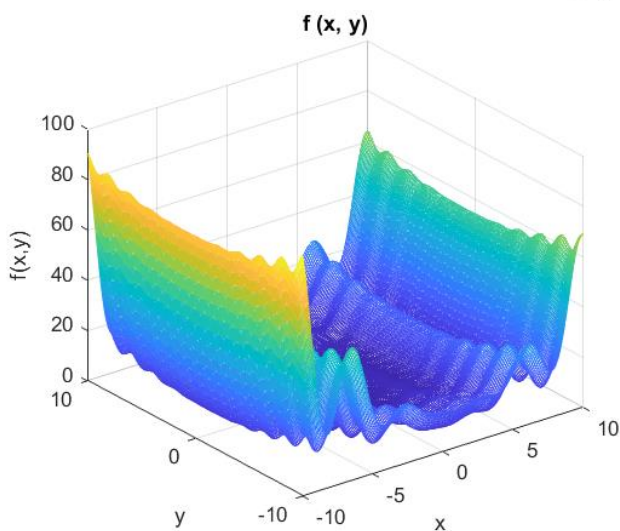
$$f(\mathbf{x}) = \sin^2(\pi w_1) + \sum_{i=1}^{n-1} [(w_i - 1)^2 [1 + 10 \sin^2(\pi w_{i+1})]] + (w_n - 1)^2 [1 + \sin^2(2\pi w_n)]$$

$$w_i = 1 + \frac{x_i - 1}{4}$$

Levyho funkce



Levyho funkce

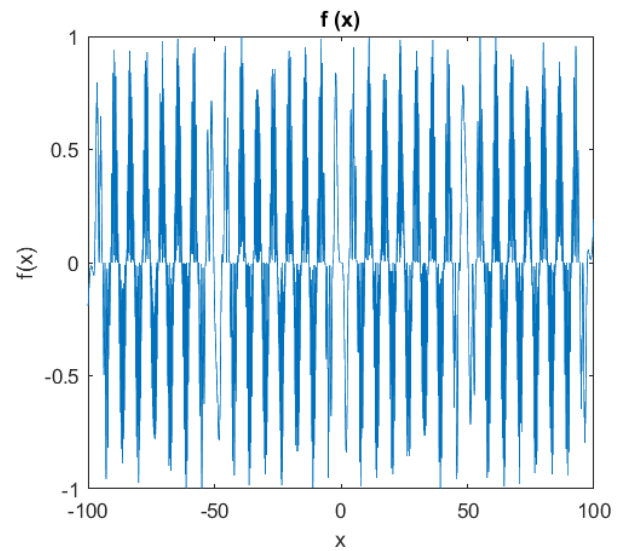
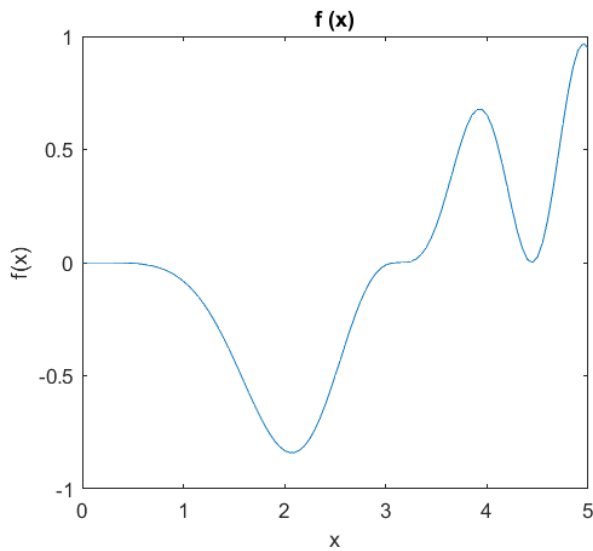


Zdroj: <https://www.sfu.ca/~ssurjano/levy.html>

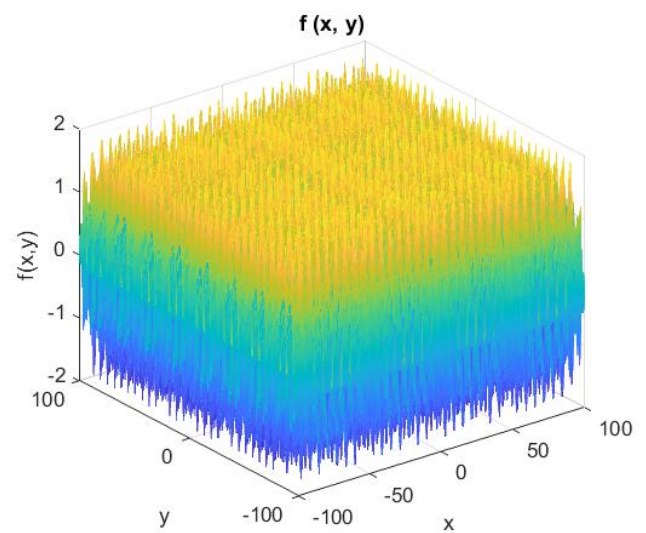
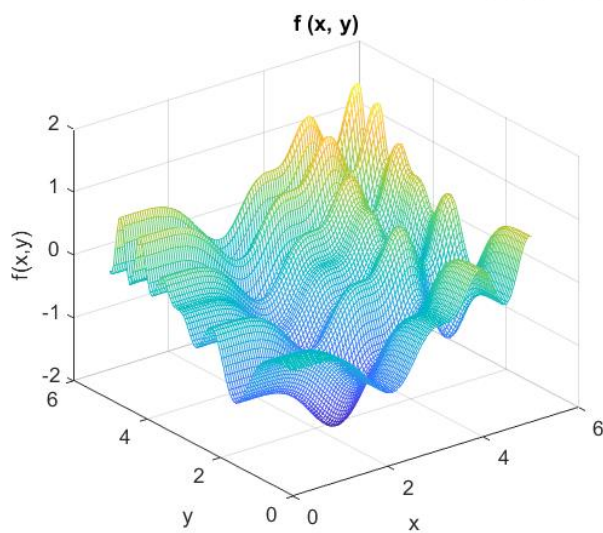
## 2.11. Michalewiczova funkce s parametrem $m = 1$

$$f(\mathbf{x}) = -\sum_{i=1}^n \sin(x_i) \sin^{2m}\left(\frac{ix_i^2}{\pi}\right), \quad m = 1$$

Michalewiczova funkce ( $m=1$ )



Michalewiczova funkce ( $m=1$ )



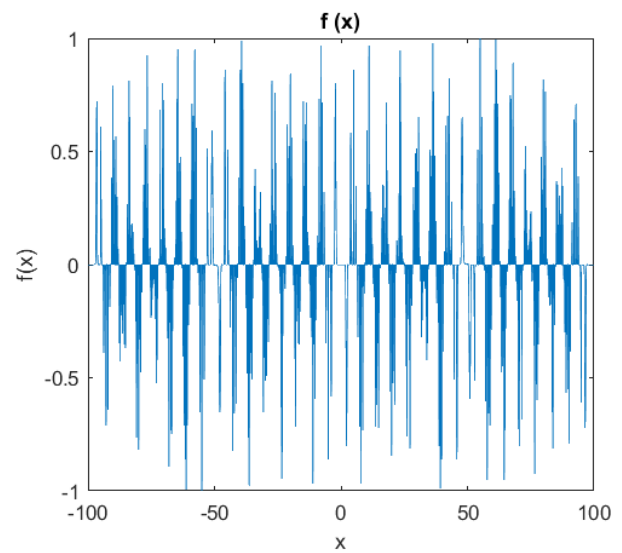
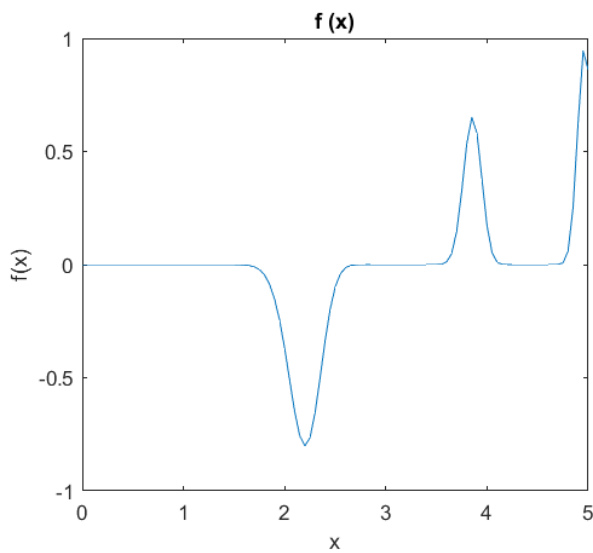
Zdroj: <https://www.sfu.ca/~ssurjano/michal.html>



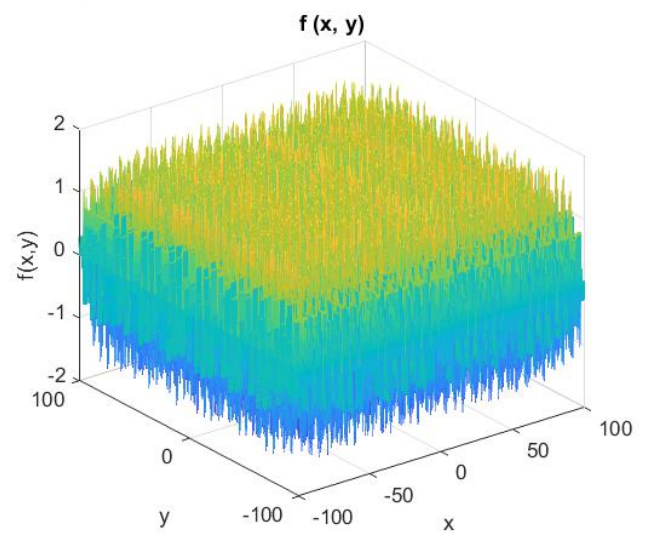
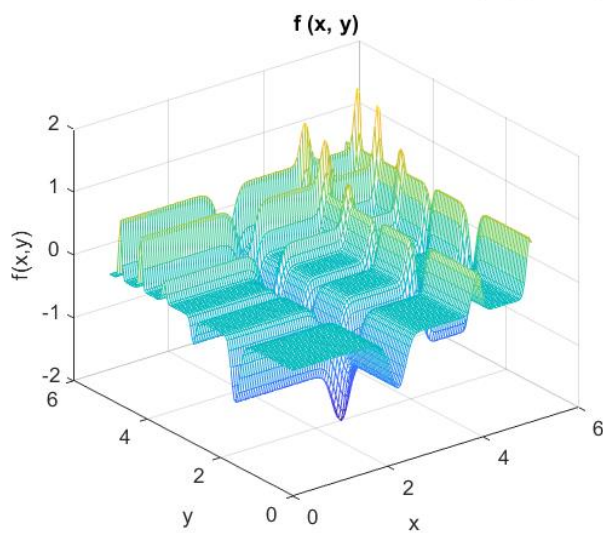
## 2.12. Michalewiczova funkce s parametrem $m = 10$

$$f(\mathbf{x}) = - \sum_{i=1}^n \sin(x_i) \sin^{2m}\left(\frac{ix_i^2}{\pi}\right), \quad m = 10$$

Michalewiczova funkce ( $m=10$ )



Michalewiczova funkce ( $m=10$ )

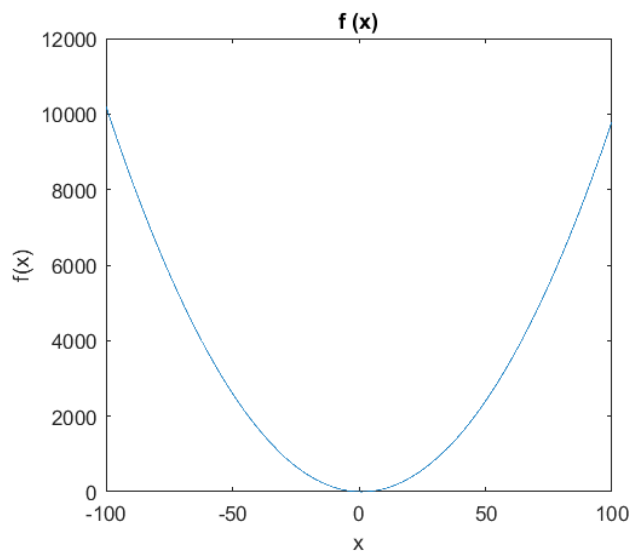
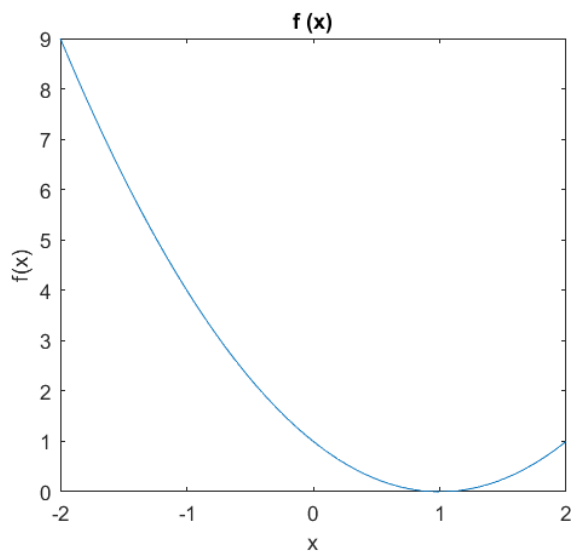


Zdroj: <https://www.sfu.ca/~ssurjano/michal.html>

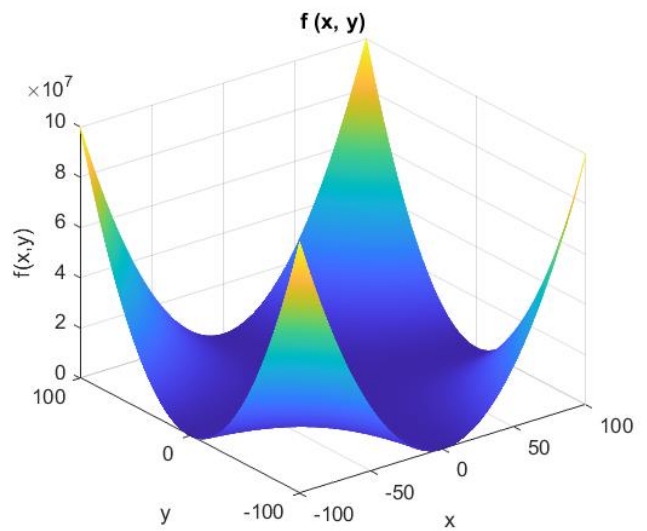
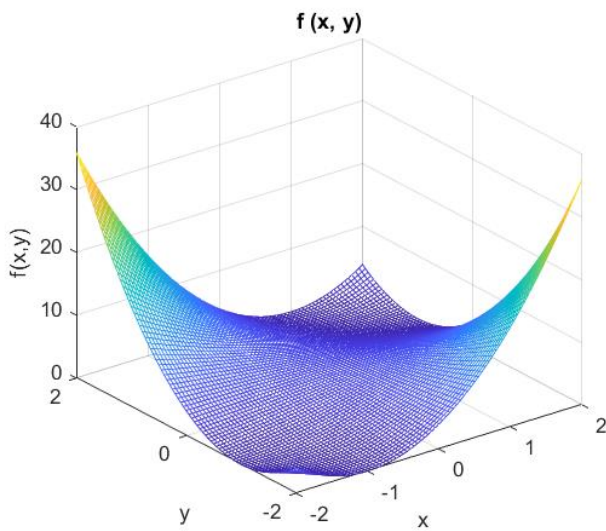
### 2.13. Mishra07 funkcce

$$f(\mathbf{x}) = \left[ \prod_{i=1}^n x_i - n! \right]^2$$

Mishra07 funkcce



Mishra07 funkcce



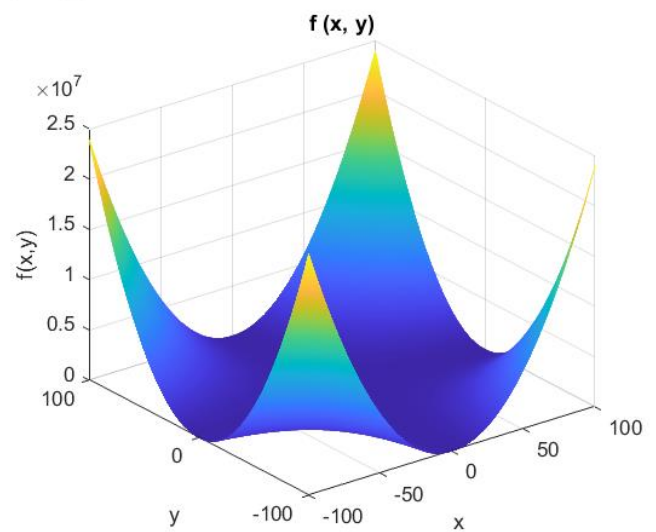
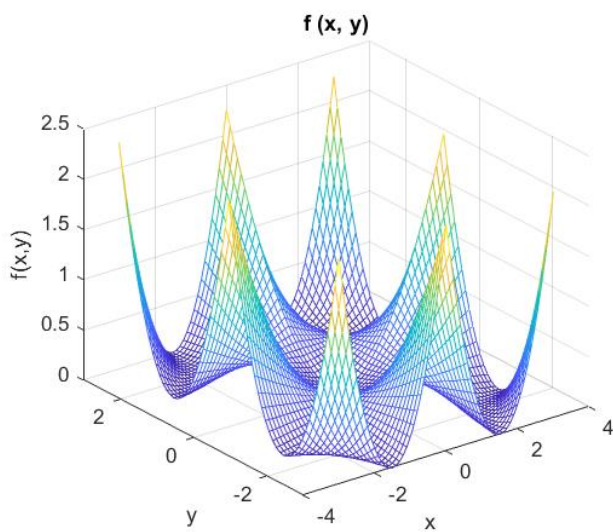
Zdroj: [https://infinity77.net/global\\_optimization/test\\_functions\\_nd\\_M.html#go\\_benchmark.Mishra07](https://infinity77.net/global_optimization/test_functions_nd_M.html#go_benchmark.Mishra07)



## 2.14. Mishra11 funkce

$$f(\mathbf{x}) = \left[ \frac{1}{n} \sum_{i=1}^n |x_i| - \left( \prod_{i=1}^n |x_i| \right)^{\frac{1}{n}} \right]^2$$

Mishra11 funkce



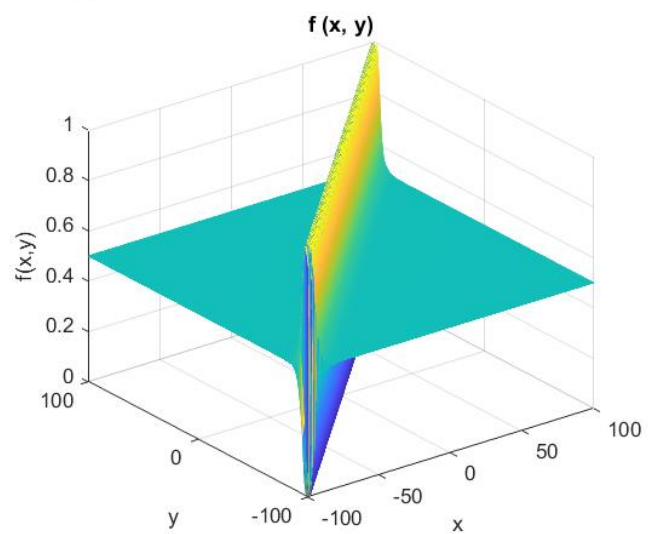
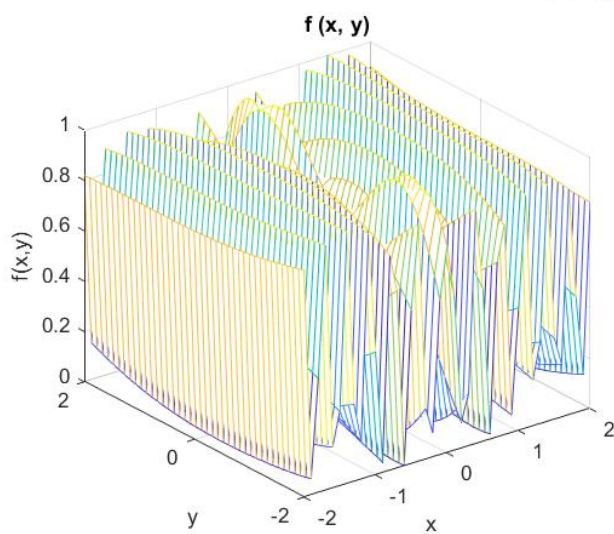
Pozn.: Funkce měří rozdíl mezi aritmetickým a geometrickým průměrem. Pro jednu vstupní proměnnou je aritmetický a geometrický průměr totožný. 2D graf funkce  $f(x) = 0$  tedy není uveden.

Zdroj: [https://infinity77.net/global\\_optimization/test\\_functions\\_nd\\_M.html#go\\_benchmark.Mishra11](https://infinity77.net/global_optimization/test_functions_nd_M.html#go_benchmark.Mishra11)

## 2.15. Patologická funkce

$$f(\mathbf{x}) = \sum_{i=1}^{n-1} \left[ 0.5 + \frac{\sin^2 \left( \sqrt{100x_i^2 + x_{i+1}^2} \right) - 0.5}{1 + 0.001(x_i^2 - 2x_i x_{i+1} + x_{i+1}^2)^2} \right]$$

Patologická funkce



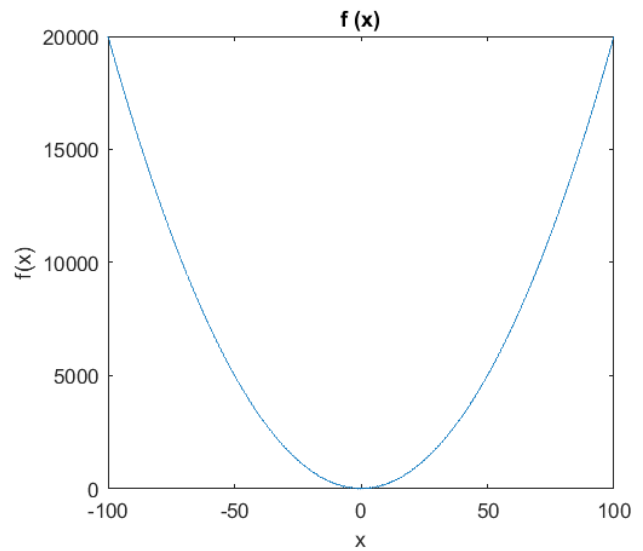
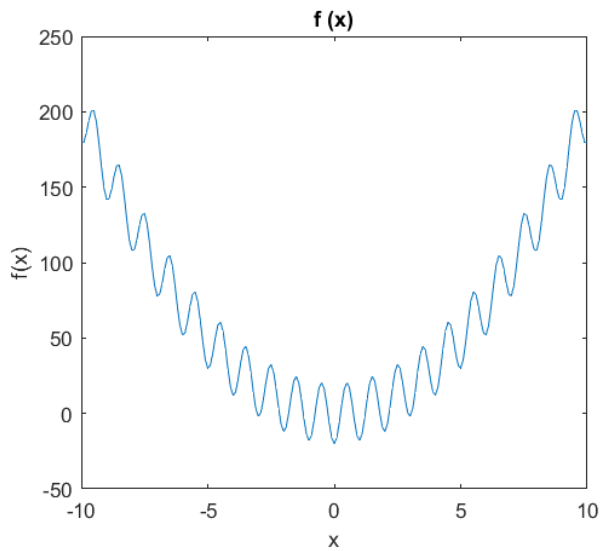
Pozn.: Funkce je definována pro 2 a více proměnných. 2D graf pro jednu proměnnou není možné vytvořit.

Zdroj: prezentace k přednáškám z matematické informatiky od prof. Šenkeříka

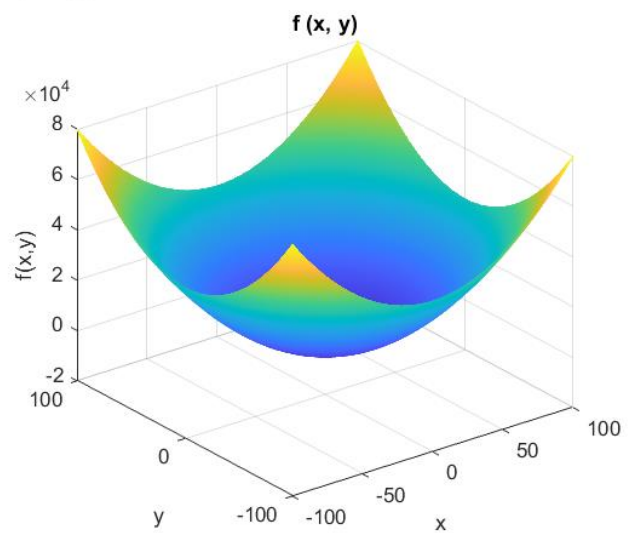
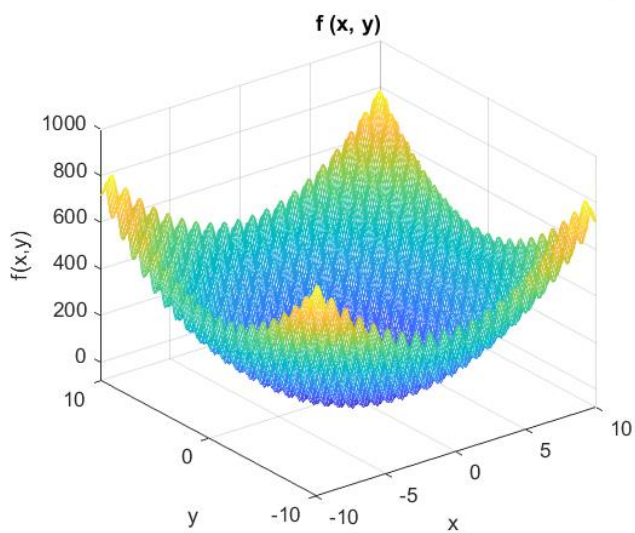
## 2.16. Rastriginova funkce

$$f(\mathbf{x}) = 2n \sum_{i=1}^n x_i^2 + 10 \cos(2\pi x_i)$$

Rastriginova funkce



Rastriginova funkce

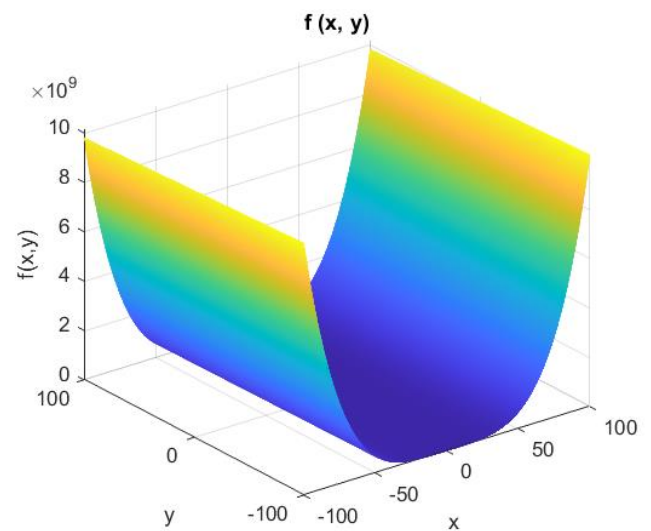
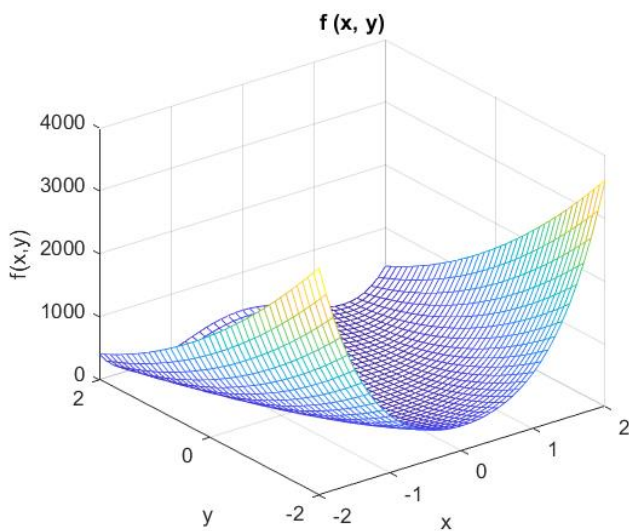


Zdroj: prezentace k přednáškám z matematické informatiky od prof. Šenkeříka

## 2.17. Rosenbrockovo sedlo

$$f(\mathbf{x}) = \sum_{i=1}^{n-1} \left[ 100(x_i^2 - x_{i+1})^2 + (1 - x_i)^2 \right]$$

Rosenbrockovo sedlo



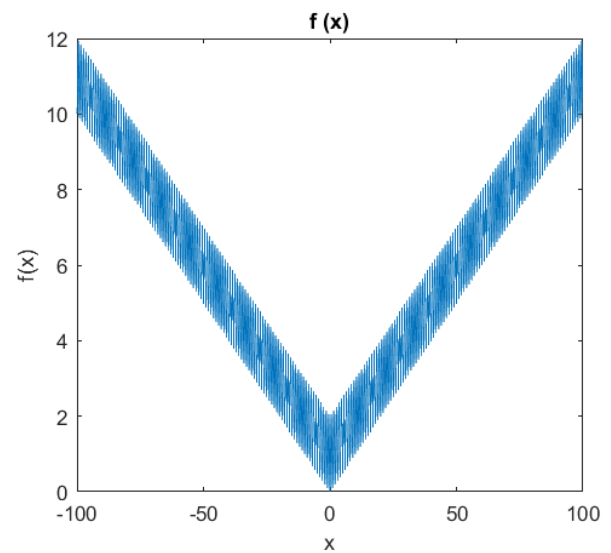
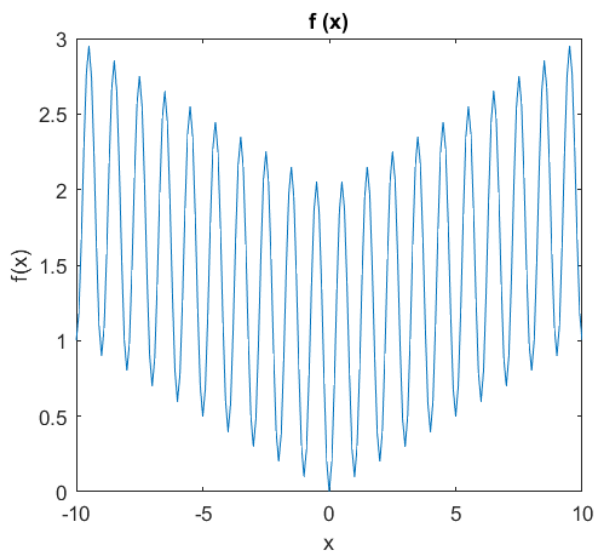
Pozn.: Funkce je definována pro 2 a více proměnných. 2D graf pro jednu proměnnou není možné vytvořit.

Zdroj: prezentace k přednáškám z matematické informatiky od prof. Šenkeříka

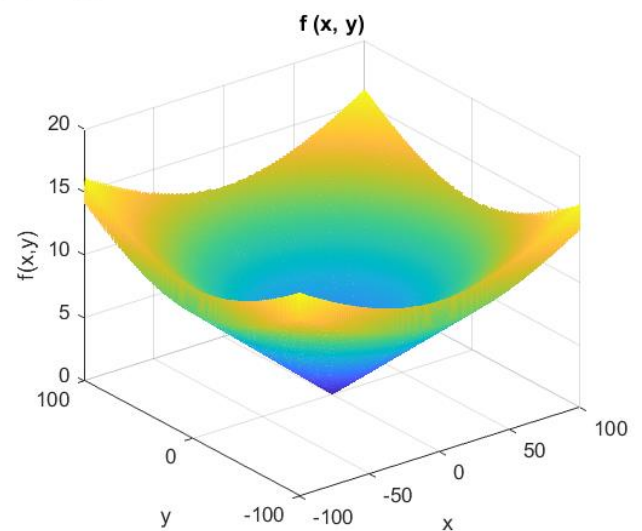
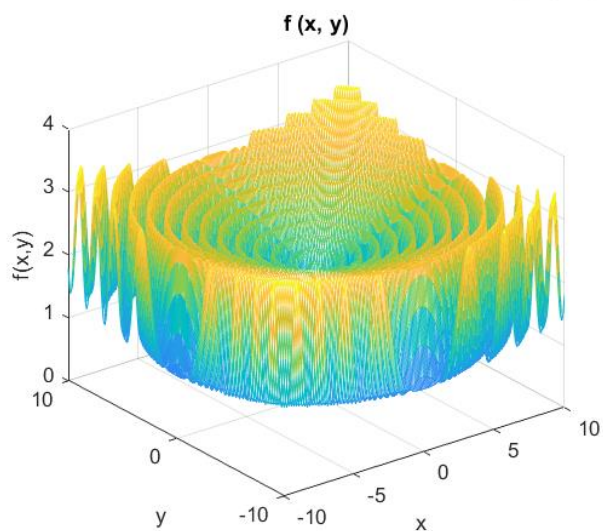
## 2.18. Salomonova funkce

$$f(\mathbf{x}) = 1 - \cos\left(2\pi \sqrt{\sum_{i=1}^n x_i^2}\right) + 0.1 \sqrt{\sum_{i=1}^n x_i^2}$$

Salomonova funkce



Salomonova funkce

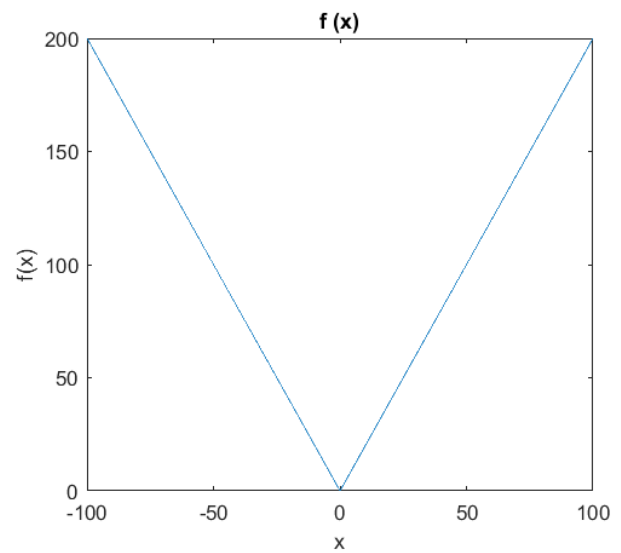
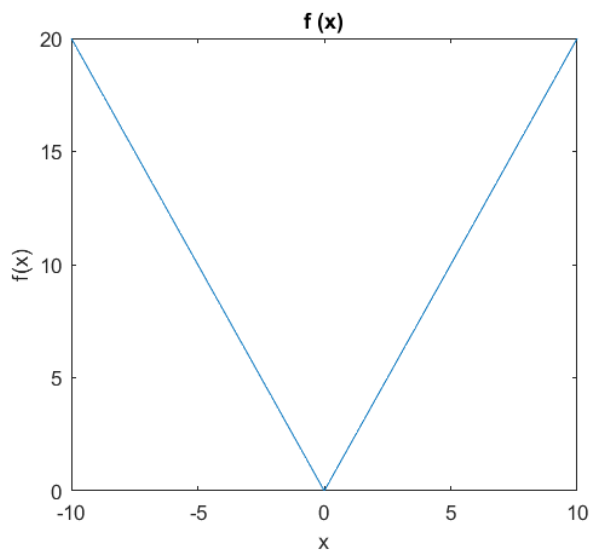


Zdroj: [https://infinity77.net/global\\_optimization/test\\_functions\\_nd\\_S.html#go\\_benchmark.Salomon](https://infinity77.net/global_optimization/test_functions_nd_S.html#go_benchmark.Salomon)

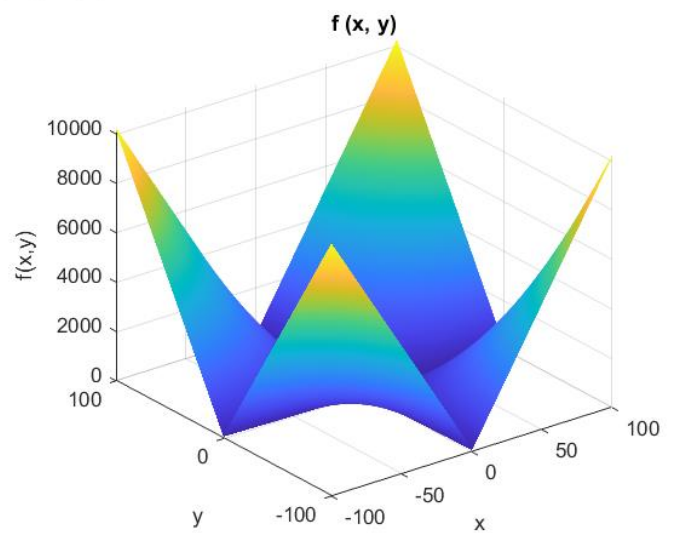
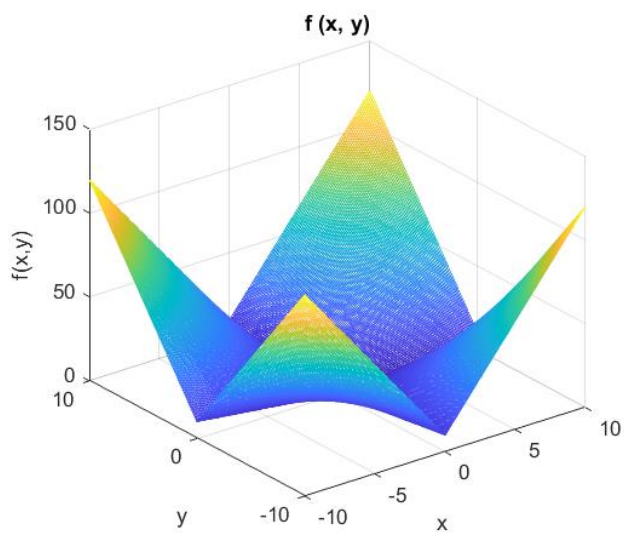
## 2.19. Schwefel22 funkce

$$f(\mathbf{x}) = \sum_{i=1}^n |x_i| + \prod_{i=1}^n |x_i|$$

Schwefelova funkce N22



Schwefelova funkce N22

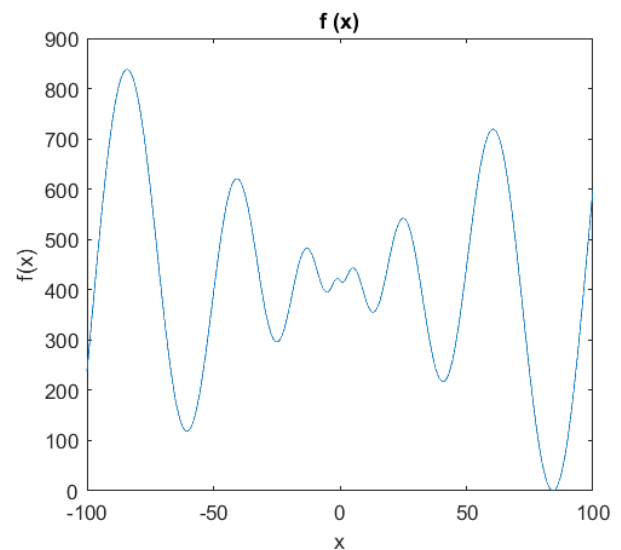
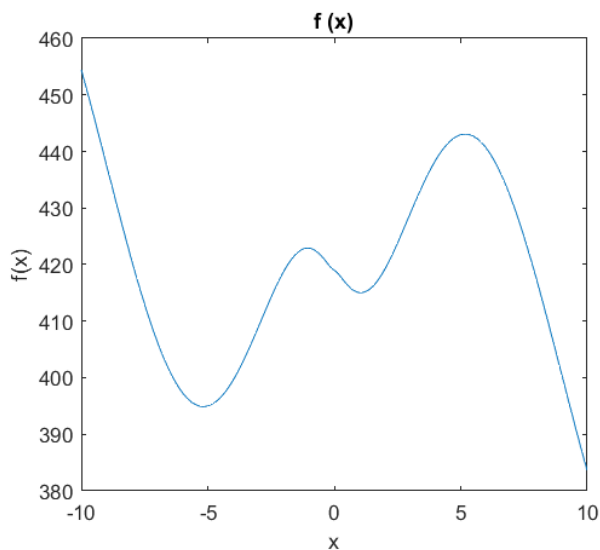


Zdroj: [https://infinity77.net/global\\_optimization/test\\_functions\\_nd\\_S.html#go\\_benchmark.Schwefel22](https://infinity77.net/global_optimization/test_functions_nd_S.html#go_benchmark.Schwefel22)

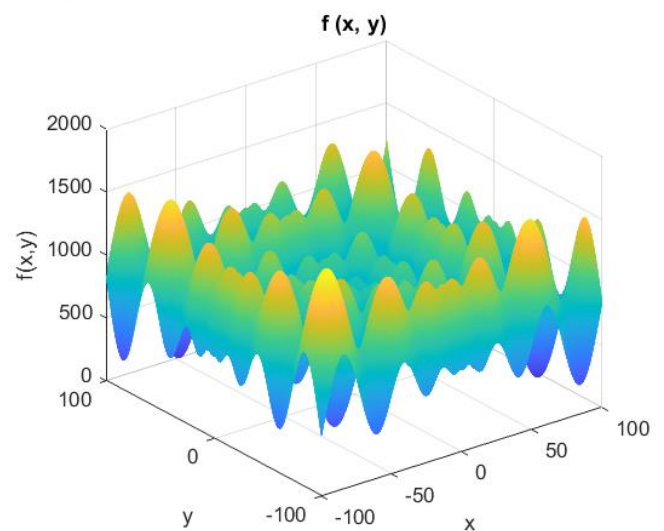
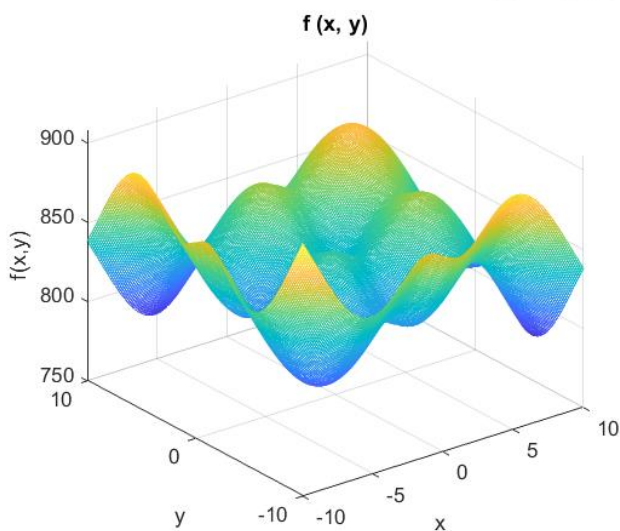
## 2.20. Schwefel26 function

$$f(\mathbf{x}) = 418.9829n - \sum_{i=1}^n 5x_i \sin(\sqrt{5|x_i|})$$

Schwefelova funkce N26



Schwefelova funkce N26



Pozn.: Schwefelova funkce je typicky prošetřována na intervalu  $x_i \in \langle -500, 500 \rangle$ . Pro zachování stejného průběhu na intervalu  $x_i \in \langle -100, 100 \rangle$  platí  $f(x_i) = f_{zdroj}(5x_i)$ .

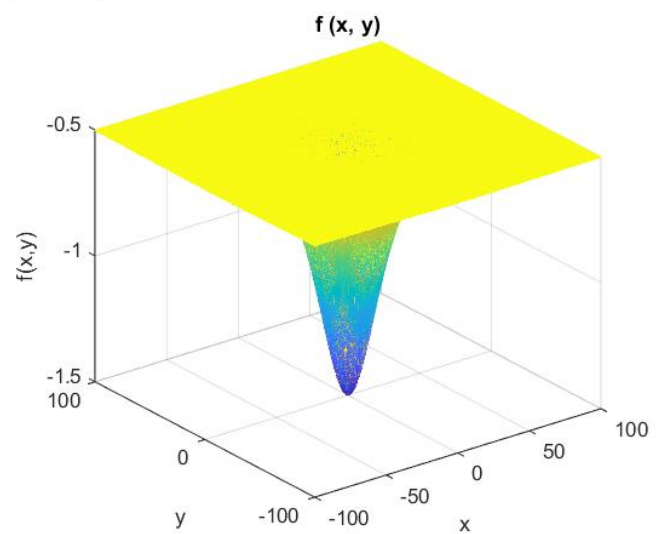
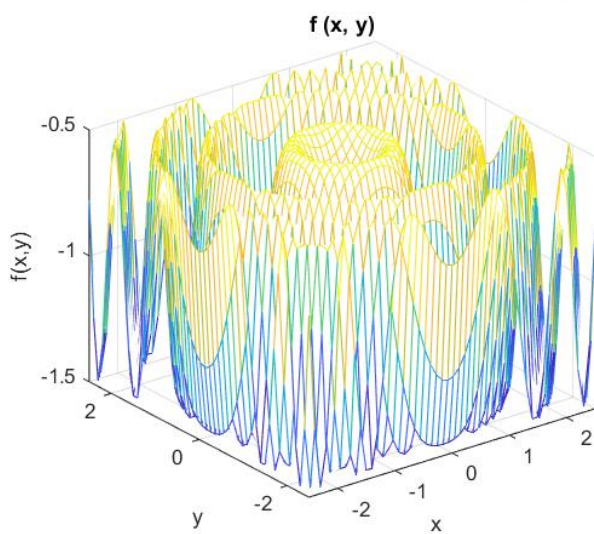
Zdroj: [https://infinity77.net/global\\_optimization/test\\_functions\\_nd\\_S.html#go\\_benchmark.Schwefel26](https://infinity77.net/global_optimization/test_functions_nd_S.html#go_benchmark.Schwefel26)



## 2.21. Sinová obálka

$$f(\mathbf{x}) = - \sum_{i=1}^{n-1} \left[ \frac{\sin^2 \left( \sqrt{x_{i+1}^2 + x_i^2} - 0.5 \right)}{(0.001(x_{i+1}^2 + x_i^2) + 1)^2 + 0.5} \right]$$

SineEnvelope funkce



Pozn.: Funkce je definována pro 2 a více proměnných. 2D graf pro jednu proměnnou není možné vytvořit.

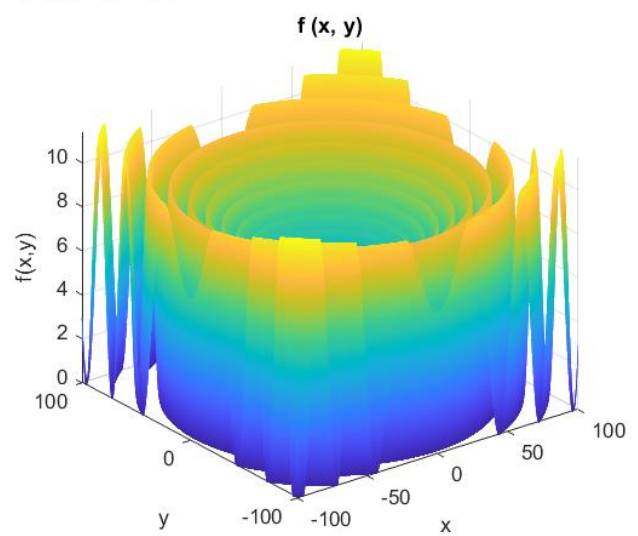
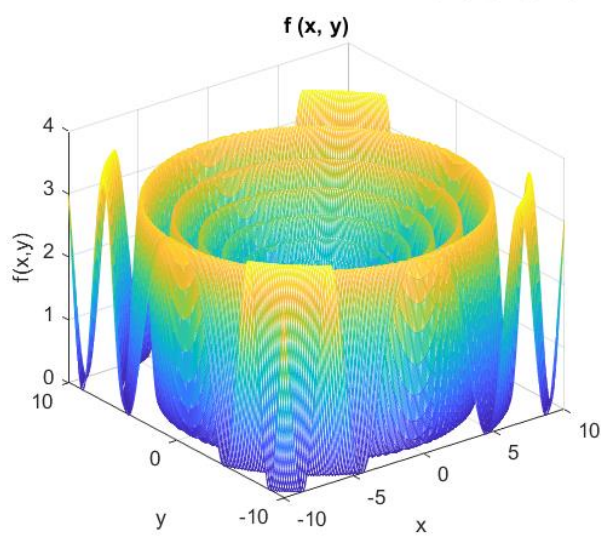
Zdroj: prezentace k přednáškám z matematické informatiky od prof. Šenkeříka



## 2.22. Roztažená sinová obálka

$$f(\mathbf{x}) = \sum_{i=1}^{n-1} t^{1/4} [\sin(50t_i^{0.1}) + 1]^2$$
$$t_i = x_{i+1}^2 + x_i^2$$

StretchedVSineEnvelope funkce



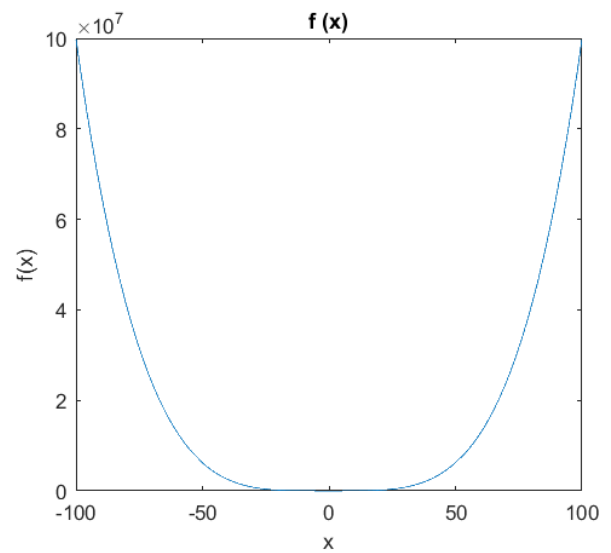
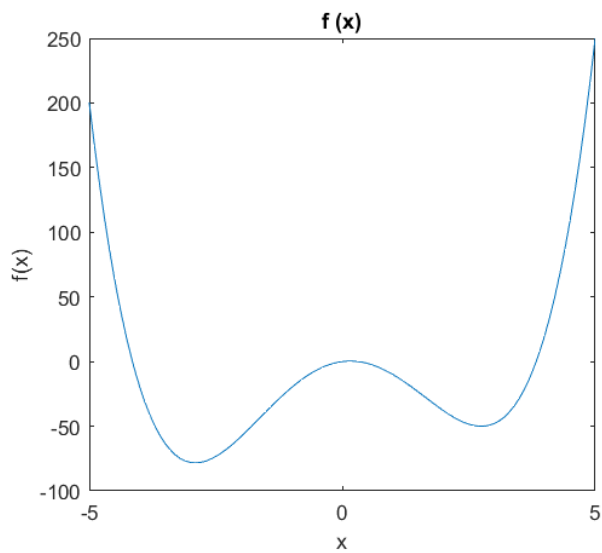
Pozn.: Funkce je definována pro 2 a více proměnných. 2D graf pro jednu proměnnou není možné vytvořit.

Zdroj: prezentace k přednáškám z matematické informatiky od prof. Šenkeříka

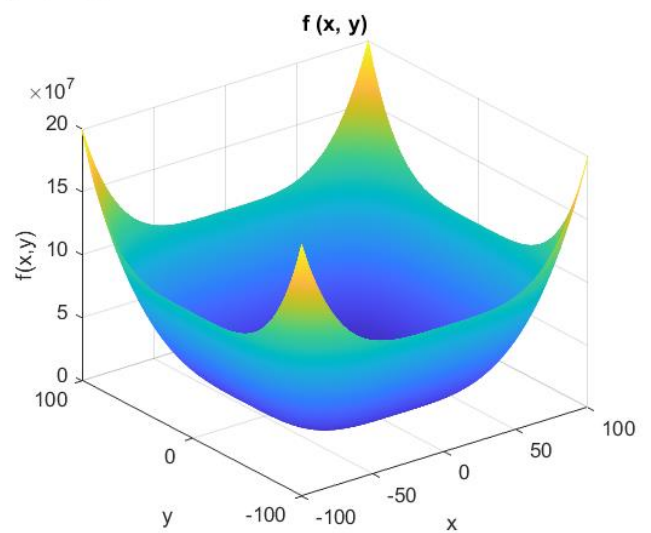
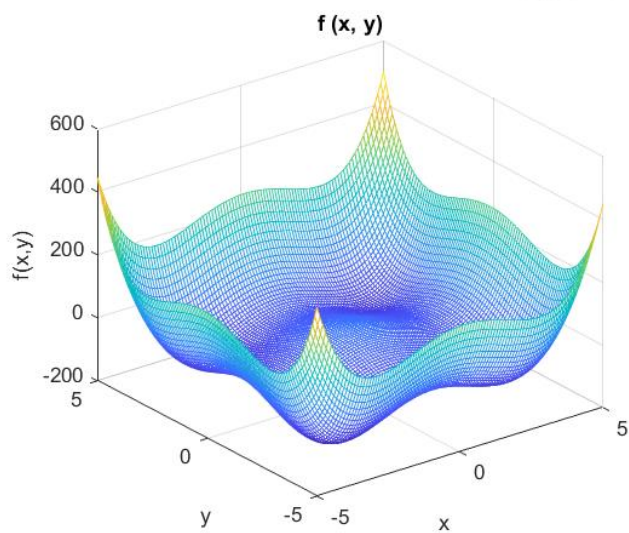
## 2.23. StyblinskiTang funkcje

$$f(\mathbf{x}) = \sum_{i=1}^n (x_i^4 - 16x_i^2 + 5x_i)$$

StyblinskiTang funkcje



StyblinskiTang funkcje

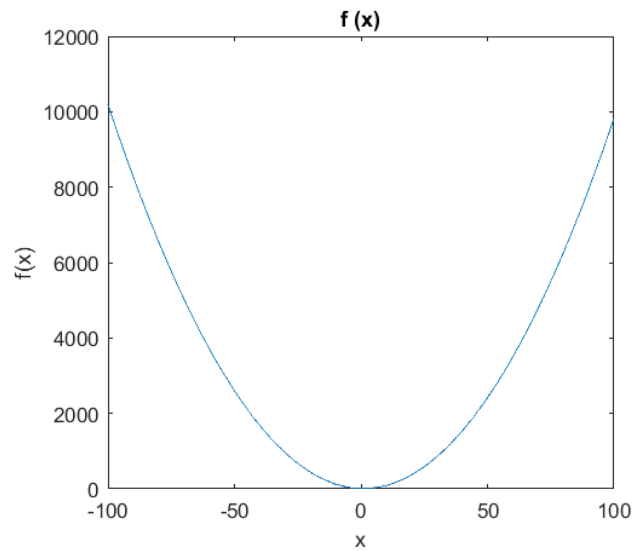
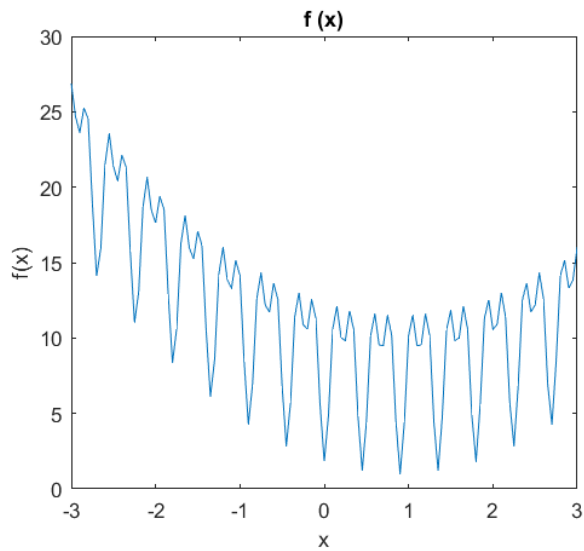


Zdroj: [https://infinity77.net/global\\_optimization/test\\_functions\\_nd\\_S.html#go\\_benchmark.StyblinskiTang](https://infinity77.net/global_optimization/test_functions_nd_S.html#go_benchmark.StyblinskiTang)

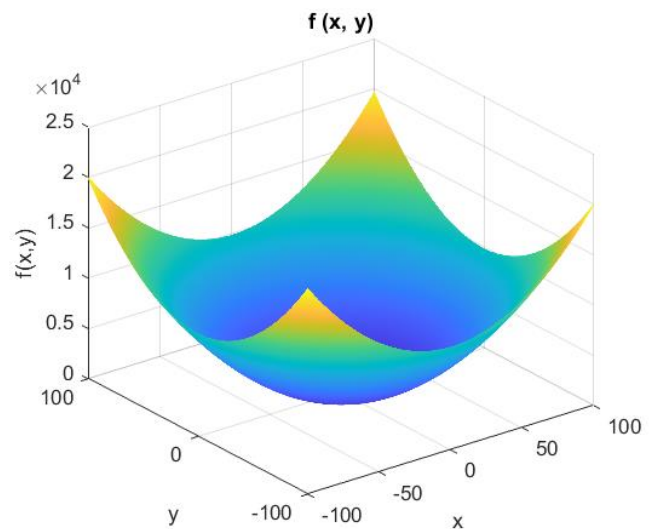
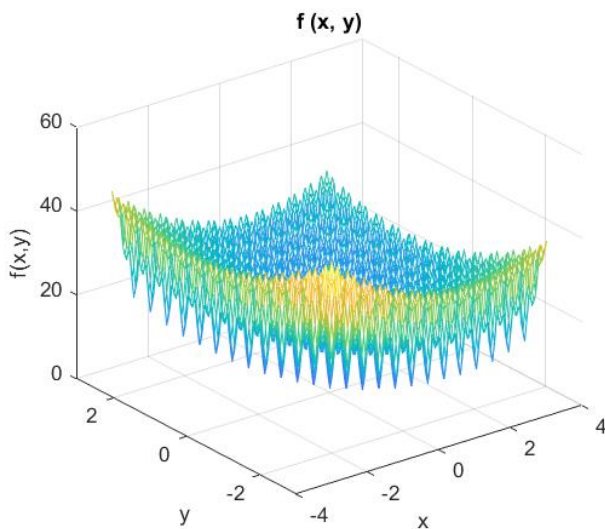
## 2.24. Trigonometrická funkce

$$f(\mathbf{x}) = 1 + \sum_{i=1}^n 8 \sin^2[7(x_i - 0.9)^2] + 6 \sin^2[14(x_i - 0.9)^2] + (x_i - 0.9)^2$$

Trigonometrická funkce N2



Trigonometrická funkce N2

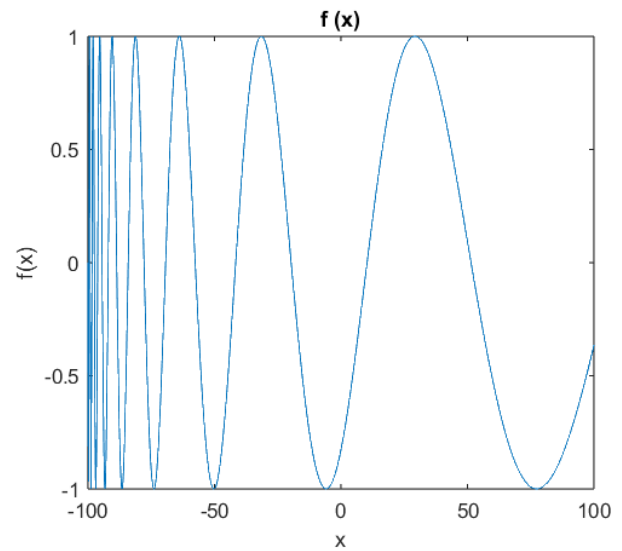
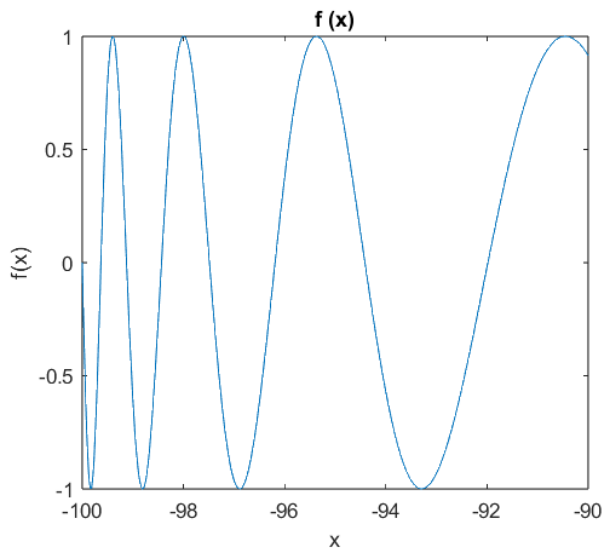


Zdroj: [https://infinity77.net/global\\_optimization/test\\_functions\\_nd\\_T.html#go\\_benchmark.Trigonometric02](https://infinity77.net/global_optimization/test_functions_nd_T.html#go_benchmark.Trigonometric02)

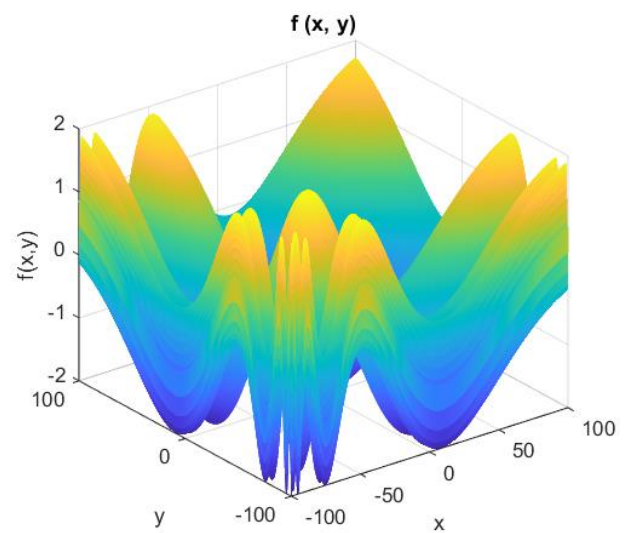
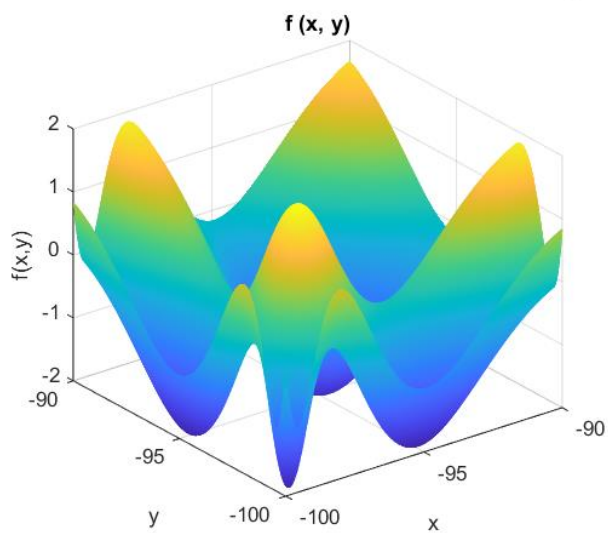
## 2.25. Vincentova funkce

$$f(\mathbf{x}) = -\sum_{i=1}^n \sin(10 \log(x_i + 101))$$

Vincentova funkce



Vincentova funkce



Pozn.: Kvůli logaritmu byl definiční obor funkce ze zdroje  $D(f) = (0, \infty)$  posunut na  $D(f) = (-101, \infty)$ .

Zdroj: [https://infinity77.net/global\\_optimization/test\\_functions\\_nd\\_V.html#go\\_benchmark.Vincent](https://infinity77.net/global_optimization/test_functions_nd_V.html#go_benchmark.Vincent)

### 3. Výsledky a pořadí algoritmů

V této části představíme výsledky algoritmů ve vyšetřovaných dimenzích a následně z nich vycházející pořadí, kterého se algoritmům v jednotlivých funkcích podařilo dosáhnout.

Dále se podíváme na průměrné pořadí každého algoritmu  $\bar{r}_j$  a na míru odlišnosti každého algoritmu od jeho průměrného pořadí  $e_i$ . K tomu využijeme následující vztahy.

$$\bar{r}_j = \frac{1}{n} \sum_{i=1}^n r_{ij}$$

$$e_i = \sum_{j=1}^k |r_{ij} - \bar{r}_j|$$

#### 3.1. Dimenze 2

Následující tabulka obsahuje nejlepší hodnoty dosažené jednotlivými algoritmy na daných funkcích.

	DE_rand_1_bin	DE_best_1_bin	PSO	SOMA_all_to_one	SOMA_all_to_all
Ackley	4.440892098500626e-16	4.440892098500626e-16	3.0252037911774705e-11	2.2344448957767325e-09	1.9858248019530853
Alpine01	0.004168389861824527	0.08077641013048172	5.261113250129751e-10	0.0052227672928335965	0.11863330931698937
Alpine02	-197.6691649632342	-196.20634161858058	-189.68224547273343	-196.74438080550905	-190.33777300148262
Bohachevsky	0.0	0.013764227675860498	0.0	7.031412489292658e-17	0.6089862979269309
CosineMixture	-0.20000000000000001	-0.20000000000000001	-0.20000000000000001	-0.20000000000000007	0.15184228669135394
DefCorrSpring	-0.20000000000000001	-0.027228428113977025	-0.20000000000000001	-0.17469814042902476	-0.08458813282629045
DixonPrice	1.849409136525733e-12	3.7388719987037537e-32	1.028748284403875e-25	4.354085519787079e-16	0.6115414614524524
Griewank	0.006052711799941842	0.021934071977454877	0.0077281806453088795	0.0078068372391915586	0.025073727057527313
InvCosineWave	-0.999999999280017	-0.9301577517188063	-0.9214274706836568	-0.8777760655079097	-0.7249824055291932
Levy	1.4997597826618565e-32	1.4997597826618565e-32	3.019960485431515e-32	1.5036561994576574e-17	0.04504184060736036
Michalewicz_m1	-1.9673647882765217	-1.9057989311279004	-1.9769648150166184	-1.9822556259746809	-1.9453314889876545
Michalewicz_m10	-1.8988553350424886	-1.8624667774768588	-1.944109943559927	-1.9459516561109393	-1.9123013551005223
Mishra07	4.238791434617831e-06	1.868196999871901e-08	7.034141215038226e-27	1.3132024278090682e-24	0.0005500848656159114
Mishra11	0.0	0.0	3.0884935365814903e-18	0.0	6.163323362539023e-15
Pathological	3.416991326828563e-06	2.987218617970244e-05	2.2988162546496347e-06	2.3722735021536322e-05	0.00026217151329807435
Rastrigin	-79.9999999999709	-78.430446838711	-79.60201637716268	-79.07137154671283	-67.23088960644658
Rosenbrock	0.026650564931311585	3.463296917838591e-26	8.280759740159551	8.337279438741882	4.617090774417533
Salomon	0.02497543960088521	0.05389406508996524	0.056829608006448866	0.08322784881713896	0.1500119138733694
Schwefel22	2.3404450821136666e-18	3.125801091323877e-28	3.6095671537046333e-16	7.98598457435784e-10	0.43879281727587155
Schwefel26	15.791803403724073	75.01097071094303	63.16713724950187	35.53152583946374	1.2164477980829966
SineEnvelope	-1.4958689031945385	-1.495869638954356	-1.4958722282713097	-1.495664973158267	-1.491409688173044
StretchedVSine	2.9720027612538687e-06	2.189969837869725e-05	2.530975720544296e-07	2.1353759505700342e-06	2.0401090294925274e-06
StyblinskiTang	-156.66466281508568	-151.00997519569066	-155.72221487851985	-154.77976694195402	-149.94615181083626
Trigonometric02	1.0	1.0	1.0	1.0000000000000002	3.4271309810194803
Vincent	-2.0	-2.0	-2.0	-1.9999999999990439	-1.9998773270556582

Následující tabulka obsahuje umístění algoritmů pro každou testovací funkci, průměrné umístění algoritmů  $\bar{r}_j$  a míru odlišnosti každé funkce  $e_i$ .

	DE_rand_1_bin	DE_best_1_bin	PSO	SOMA_all_to_one	SOMA_all_to_all	$e_i$
Ackley	1	1	3	4	5	4,92
Alpine01	2	4	1	3	5	3,2
Alpine02	1	3	5	2	4	5,44
Bohachevsky	1	4	1	3	5	3,88
CosineMixture	1	1	1	4	5	5,24
DefCorrSpring	1	5	1	3	4	4,76
DixonPrice	4	1	2	3	5	4,8
Griewank	1	4	2	3	5	2,88
InvCosineWave	1	2	3	4	5	3,92
Levy	1	1	3	4	5	4,92
Michalewicz_m1	3	5	2	1	4	6,08
Michalewicz_m10	4	5	2	1	3	8,08
Mishra07	4	3	1	2	5	5,2
Mishra11	1	1	4	1	5	7,16
Pathological	2	4	1	3	5	3,2
Rastrigin	1	4	2	3	5	2,88
Rosenbrock	2	1	4	5	3	7,12
Salomon	1	2	3	4	5	3,92
Schwefel22	2	1	3	4	5	4,24
Schwefel26	2	5	4	3	1	7,76
SineEnvelope	3	2	1	4	5	4,56
StretchedVSine	4	5	1	3	2	8,08
StyblinskiTang	1	4	2	3	5	2,88
Trigonometric02	1	1	1	4	5	5,24
Vincent	1	1	1	4	5	5,24
$\bar{r}_j$	<b>1,84</b>	<b>2,8</b>	<b>2,16</b>	<b>3,12</b>	<b>4,44</b>	<b>5,024</b>

Největší rozdíly v pořadí máme pro funkce Michalewicz\_m10, Mishra11, Rosenbrockovo sedlo, Schwefelova funkce a Roztaženou sinovou obálku.

U Mishra11 a Roztažené sinové obálky vychází rozdíly nalezených minim řádově  $10^{-15}$  a  $10^{-5}$ . Pro tyto funkce tak nemá smysl, kvůli malým rozdílům ve výsledcích, uvažovat nad chováním algoritmů.

U Rosenbrockova sedla je situace zajímavější. Velká oblast sedla s malou změnou funkčních hodnot, zdá se, vyhovuje diferenciální evoluci. Jedinci zřejmě mohou lépe prohledávat okolní prostor a postupovat tak rychleji ke globálnímu minimu. U PSO může být postup zpomalen chováním hejna. Jedinci upravují své polohy vzájemně na sobě a postup ke globálnímu minimu může být více usměrněný. U SOMA je situace podobná. Vyšetřováním trajektorií mezi jedinci je postup pravděpodobně zpomalen.



U Schwefelovy a Michalewicz\_m10 funkce vypadá situace obráceně. Obě tyto funkce mají mnoho lokálních extrémů a prudký průběh. Zde naopak situaci lépe zvládají hejna, která mohou v okolním prostoru lépe určit postup pro další migraci.

Dále si můžeme všimnout, že nejlepšího průměrného pořadí dosahuje DE s rand/1 mutací. Pro dimenzi 2 se tedy zdá, že je prvek náhody v mutaci výhodný.

### 3.2. Dimenze 10

Následující tabulka obsahuje nejlepší hodnoty dosažené jednotlivými algoritmy na daných funkcích.

	DE_rand_1_bin	DE_best_1_bin	PSO	SOMA_all_to_one	SOMA_all_to_all
Ackley	20.385611310459975	20.244356816002608	18.74024583958154	17.474778409338608	19.936393016986997
Alpine01	22.405725121262194	1.3682023441190443	1.2834505434446087e-08	0.6074046266723825	28.21735969744521
Alpine02	-49089965007.499245	-87916029188.06705	-49068441575.94983	-141566227925.2047	-22702885336.00723
Bohachevsky	3.2917055789952063	0.01566274716408151	0.3212350973567219	1.343064391077472	2182.840468101238
CosineMixture	-0.25328260348707643	-0.990147716618615	-0.9113294495675739	-0.7774095088406333	876.404747573301
DefCorrSpring	-0.7195087112570903	-0.8433397703260412	-0.6292394543953838	0.13835807718421625	91.72568613138709
DixonPrice	2.875361644208761	0.6000001637902999	0.5555555555555552	0.5520787614432972	2097211.1000695694
Griewank	0.641113925423507	0.2084953729782003	0.07937861393322174	0.16382217516840036	1.182386223474738
InvCosineWave	-1.5784037334482117	-4.6742204390675965	-4.049931927074064	-3.4675065041836537	-1.5685317880849396
Levy	4.463543339969312	0.04543240185962583	0.2071235633941873	17.275657046828755	218.4392602036986
Michalewicz_m1	-6.0350551506319565	-7.325221423784094	-8.81733876458018	-9.459263249360166	-6.439412899574905
Michalewicz_m10	-4.414333749371736	-6.017324186030933	-6.2293711432058405	-7.5586469567809385	-4.494877746868026
Mishra07	10146273301.15858	253664486.43132558	8.357969987401089e-16	918.216217439968	98224794270.47835
Mishra11	0.040137371562334954	1.635044821873942e-15	0.00013257823527339948	3.318204525440943e-15	0.04926720379541994
Pathological	3.008145383840719	2.2767779076250743	1.6207413042976868	1.57399623876059	2.85560588332502
Rastrigin	-924.6679023845526	-1669.6572848513472	-1838.1534542451516	-1536.8648461904186	16511.320002594737
Rosenbrock	367.0007012344892	2.041321568172982	32.87528318019496	266.2912912315926	10834251.15799522
Salomon	0.6575219600261054	0.11023286648241774	0.19320667918014428	0.32764433406711224	3.4514290443604834
Schwefel22	168.38044395003217	2.5874359511836174e-05	5.828575821168369e-20	3.785789628276363	34690.66774554831
Schwefel26	770.6556839917755	256.6670875906831	584.428205525726	418.62408169514674	1141.8764779530636
SineEnvelope	-9.901012880086501	-11.437939607762122	-12.81254781074742	-12.705149144226299	-9.122695119564868
StretchedVSine	4.59200578233565	4.187621764095541	0.06876648537364402	0.038863948688617006	3.6980246004419564
StyblinskiTang	-618.8196522525504	-739.9707089928919	-711.6972708964253	-693.7589445122891	192595.13969261936
Trigonometric02	30.813661598708475	2.650952538666705	2.3574751102243026	9.631443209033904	927.2583572942733
Vincent	-9.706079959572204	-9.99999999971003	-10.0	-9.999999782980769	-9.413799063865143

Následující tabulka obsahuje umístění algoritmů pro každou testovací funkci, průměrné umístění algoritmů  $\bar{r}_j$  a míru odlišnosti každé funkce  $e_i$ .

	DE_rand_1_bin	DE_best_1_bin	PSO	SOMA_all_to_one	SOMA_all_to_all	$e_i$
Ackley	5	4	2	1	3	6
Alpine01	4	3	1	2	5	2,4
Alpine02	3	2	4	1	5	4,8
Bohachevsky	4	1	2	3	5	2,24
CosineMixture	4	1	2	3	5	2,24
DefCorrSpring	2	1	3	4	5	6,24
DixonPrice	4	3	2	1	5	2,64
Griewank	4	3	1	2	5	2,4
InvCosineWave	4	1	2	3	5	2,24
Levy	3	1	2	4	5	4,24
Michalewicz_m1	5	3	2	1	4	4
Michalewicz_m10	5	3	2	1	4	4
Mishra07	4	3	1	2	5	2,4
Mishra11	4	1	3	2	5	2,8
Pathological	5	3	2	1	4	4
Rastrigin	4	2	1	3	5	2
Rosenbrock	4	1	2	3	5	2,24
Salomon	4	1	2	3	5	2,24
Schwefel22	4	2	1	3	5	2
Schwefel26	4	1	3	2	5	2,8
SineEnvelope	4	3	1	2	5	2,4
StretchedVSine	5	4	2	1	3	6
StyblinskiTang	4	1	2	3	5	2,24
Trigonometric02	4	2	1	3	5	2
Vincent	4	2	1	3	5	2
$\bar{r}_j$	<b>4,04</b>	<b>2,08</b>	<b>1,88</b>	<b>2,28</b>	<b>4,72</b>	<b>3,1424</b>

V dimenzi 10 začíná SOMA\_all\_to\_all zaostávat. To může být způsobeno vyšetřováním jednotlivých trajektorií mezi všemi jedinci. Ostatní algoritmy nebo strategie zřejmě dokáží využít vyhodnocování prostoru lépe. Diferenciální evoluce s rand/1 mutací se také umísťuje špatně. Zdá se, že se ve vyšší dimenzi začíná vyplácet strategie s leaderem a povědomí o nejlepším současném řešení.

Největší rozdíly v pořadí vypadají obdobně jako ve dvou dimenzích. U funkcí s hladším průběhem, jako je DeflectedCorrugatedSpring a Levy, se diferenciální evoluci daří umístit lépe.

Funkce Ackley, Alpine02 a Roztažená sinová obálka mají průběh prudší a hejna se zde umísťují lépe. K větším rozdílu zde přispělo průměrně špatné umístění SOMA\_all\_to\_all.



### 3.3. Dimenze 30

Následující tabulka obsahuje nejlepší hodnoty dosažené jednotlivými algoritmy na daných funkcích.

	DE_rand_1_bin	DE_best_1_bin	PSO	SOMA_all_to_one	SOMA_all_to_all
Ackley	20.941051120819182	20.928816085862003	20.039193261048332	20.05368238751081	20.997347573527648
Alpine01	332.14055515591264	177.50727793597383	1.1858930208349772e-05	27.287518964144073	351.09502081163856
Alpine02	-2.9349501527256113e+28	-1.1058466150274335e+30	-1.532276205283951e+31	-5.818282610221445e+31	-1.5780089578245314e+27
Bohachevsky	37119.16439358625	80.1710323014245	3.2052372649325163	260.57096315022	69993.15210558941
CosineMixture	13227.080425340442	24.317277374964455	-2.330296757518664	97.66804202420379	24777.40667580812
DefCorrSpring	1278.073299189688	2.008540419777351	-0.8633380652437765	38.70859792535776	2440.0604131560262
DixonPrice	858996734.9868045	49536.80562733046	0.6666666666667773	263635.8834513362	2553859281.6490073
Griewank	4.275666463571634	0.715651690535374	0.013272784297659967	0.6511456451912803	7.25282912986469
InvCosineWave	-1.6551389114553492	-2.1488140333763273	-13.91683947678352	-5.833605731451053	-1.3299136634637707
Levy	5921.716728689112	599.3729348016963	106.40650639470825	477.06784839408886	6818.700238534768
Michalewicz_m1	-10.899633716875687	-12.029088049499626	-21.141854566470432	-23.896416856575684	-10.262169196684205
Michalewicz_m10	-7.406573900609985	-8.065353114456892	-10.60228185177442	-14.917409319280335	-6.36300411520386
Mishra07	8.227246865677234e+63	3.902756818241737e+61	4.967286399193678e+55	9.599352863102326e+55	4.93327792957882e+65
Mishra11	4.562251752351004	1.658547281503003e-11	0.00045907658194486685	0.0007370854210073956	7.035141224399782
Pathological	12.130082234346377	11.963099537652726	8.537341756103558	8.557002072308071	12.212932041659466
Rastrigin	737466.5372713939	68.98738939922168	-12696.894405800154	19326.929013311616	1447400.9143498833
Rosenbrock	1707414225.702424	86317.80195684623	83.03797495724994	288250.7423018239	4484133878.8525915
Salomon	12.776608922411492	2.2156560770489206	0.42654003727006445	2.4465806322235997	16.305552766554356
Schwefel22	8.589980928956524e+30	1.8688225967257487e+23	9.338715783562291e-08	174.57091842069636	3.0023401288858014e+32
Schwefel26	7334.6755678581085	2937.965956610562	3210.5222072875154	2897.6317247273355	8048.701811744535
SineEnvelope	-20.861617156844943	-24.413206649860836	-34.64099513821149	-33.38452222033233	-20.709732427041796
StretchedVSine	52.10216201088897	50.03780226805141	0.8776318022421703	3.497119657827512	50.71436942323419
StyblinskiTang	19441083.12251244	-778.6640335837127	-2050.2714983983506	6045.444621144051	51042828.312546
Trigonometric02	12862.575151984052	199.08602696083568	35.983932500062096	421.0169255917938	25452.80095996478
Vincent	-19.023100788815153	-27.51010643498117	-30.0	-29.74938979152097	-18.93835409733511

Následující tabulka obsahuje umístění algoritmů pro každou testovací funkci, průměrné umístění algoritmů  $\bar{r}_j$  a míru odlišnosti každé funkce  $e_i$ .

	DE_rand_1_bin	DE_best_1_bin	PSO	SOMA_all_to_one	SOMA_all_to_all	$e_i$
Ackley	4	3	1	2	5	1,04
Alpine01	4	3	1	2	5	1,04
Alpine02	4	3	2	1	5	2,56
Bohachevsky	4	2	1	3	5	1,6
CosineMixture	4	2	1	3	5	1,6
DefCorrSpring	4	2	1	3	5	1,6
DixonPrice	4	2	1	3	5	1,6
Griewank	4	3	1	2	5	1,04
InvCosineWave	4	3	1	2	5	1,04
Levy	4	3	1	2	5	1,04
Michalewicz_m1	4	3	2	1	5	2,56
Michalewicz_m10	4	3	2	1	5	2,56
Mishra07	4	3	1	2	5	1,04
Mishra11	4	1	2	3	5	3,12
Pathological	4	3	1	2	5	1,04
Rastrigin	4	2	1	3	5	1,6
Rosenbrock	4	2	1	3	5	1,6
Salomon	4	2	1	3	5	1,6
Schwefel22	4	3	1	2	5	1,04
Schwefel26	4	2	3	1	5	3,6
SineEnvelope	4	3	1	2	5	1,04
StretchedVSine	5	3	1	2	4	2,88
StyblinskiTang	4	2	1	3	5	1,6
Trigonometric02	4	2	1	3	5	1,6
Vincent	4	3	1	2	5	1,04
$\bar{r}_j$	<b>4,04</b>	<b>2,52</b>	<b>1,24</b>	<b>2,24</b>	<b>4,96</b>	<b>1,6832</b>

V dimenzi 30 se potvrzuje strategie leadera a povědomí o nejlepším současném řešení. Se zvyšující se dimenzí se vysoké výpočetní nároky SOMA\_all\_to\_all umísťují na posledním místě a náhodné mutování DE\_rand\_1\_bin na místě předposledním. Srovnávání algoritmů tedy zbývá pouze pro DE\_best\_1\_bin, PSO a SOMA\_all\_to\_one, ze kterých nejlépe vychází PSO.

To se také projevuje v průměrné hodnotě míry odlišnosti, která je z daných vyšetřovaných dimenzí nejmenší.

Pro funkce s nejvyšším rozdílem v pořadí platí stejný fakt, jako v dimenzích předchozích. U klidnější funkce, Mishra11, lépe vychází evoluce. U Michalewiczovy, Schwefelovy funkce a Roztažené sinové obálky opět vítězí hejna.

## 4. Friedmanův rank test

Pomocí Friedmanova rank testu vyhodnotíme, zda se na námi zvolených funkcích daří lépe vycházejícím algoritmům opakovaně umisťovat lépe než algoritmům ostatním.

K zamítnutí, nebo přijetí hypotézy použijeme hladinu významnosti 0,05 (95% kvantil).

Hypotéza: Umístění algoritmů je konzistentní.

Hypotézu zamítáme, pokud je vypočtená testovací hodnota větší než hodnota z chí kvadrát rozdělení.

Vypočtené testovací hodnoty:

```
-----  
Ranks_2D.csv  
Friedman Test Statistic: 35.49999999999986  
P-value: 3.6666902039719866e-07  
-----  
Ranks_10D.csv  
Friedman Test Statistic: 66.5919999999998  
P-value: 1.1884321747159952e-13  
-----  
Ranks_30D.csv  
Friedman Test Statistic: 88.28800000000001  
P-value: 3.041616308517811e-18  
-----
```

Jelikož je hodnota z chí kvadrát rozdělení 9,49, hypotézu zamítáme pro všechny dimenze.

Pro dosažení lepší testovací hodnoty by bylo vhodné založit „úspěšnost“ algoritmů. Díky znalosti globálních minim můžeme porovnávat nejlepší nalezené hodnoty s těmi opravdu nejlepšími. V některých případech se nám stává, že rozdíl mezi nalezenými hodnotami a také globálními minimy nepřesahuje řádově  $10^{-10}$  a přesto má některý algoritmus rank 5.

Tomu bychom se mohli vyhnout zvolením velikosti chyby, např.  $10^{-3}$ . Pokud by byl rozdíl nalezeného a globálního minima menší, než je velikost chyby, daný algoritmus bychom považovali za úspěšný a jeho rank by byl automaticky 1.