

9. Minimum funkcji

9.1. Minimum funkcji jednej zmiennej - funkcje `fminbnd` i `fmin`

Funkcja `fminbnd` jest stosunkowo nową funkcją i nie występuje we wcześniejszych wersjach MATLAB-a (np. w wersji 5.0). We wszystkich natomiast wersjach można korzystać z funkcji `fmin`. Funkcja `fmin` zostanie prawdopodobnie w następnych wersjach MATLAB-a usunięta. W wersji obecnej wywołanie funkcji `fmin` generuje ostrzeżenie informujące o zastąpieniu funkcji `fmin` funkcją `fminbnd`.

Przedstawiony poniżej sposób korzystania z funkcji `fminbnd` dotyczy również funkcji `fmin`.

Funkcja `fminbnd` może być wywołana na kilka różnych sposobów. Najmniejsza liczba parametrów wejściowych podczas wywołania funkcji wynosi 3, najmniejsza liczba parametrów wyjściowych wynosi 1.

`fminbnd(funkcja, x1, x2)`

Funkcja `fminbnd` wywołana z 3 parametrami, oblicza minimum funkcji której nazwa podana jest w pierwszym parametrze, parametry drugi i trzeci wyznaczają przedział w którym poszukiwane jest minimum: $[x_1, x_2]$. Parametrem wyjściowym (wartością zwracaną przez funkcję) jest znaleziony x , dla którego w podanym przedziale funkcja osiąga wartość minimalną.

Przykład 1.

Należy obliczyć minimum funkcji $y = x^2 + 3$ w przedziale $x \in [-5, 5]$.

Rozwiązanie:

1. Należy utworzyć funkcję obliczającą dla x wartość y zgodnie ze wzorem $y = x^2 + 3$, zapisać w M-pliku np. pod nazwą `Funkcja1.m`
2. Wywołać w oknie **Command Window** polecenie minimalizujące funkcję.

1. Definicja funkcji

(funkcja zapisana w pliku „Funkcja1.m”)

```
function [y] = Funkcja1(x)
% Funkcja oblicza wartość  $y(x) = x^2 + 3$ 
y = x^2 + 3;
```



2. Wywołanie minimalizacji

```
fminbnd('Funkcja1', -5, 5)
```

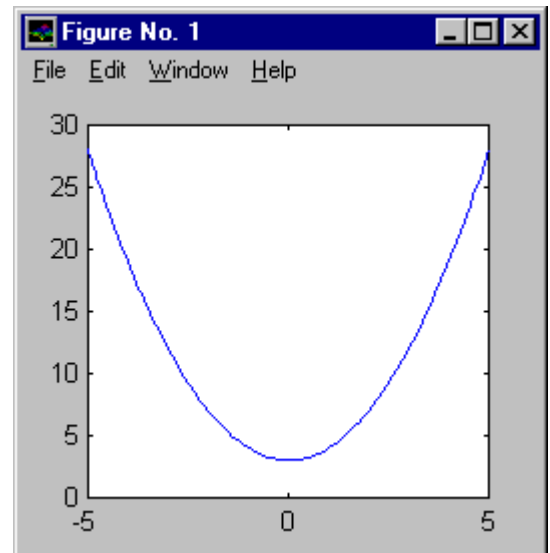
```
ans =
```

```
0
```

Wyniki możesz sprawdzić na wykresie:

```
fplot('Funkcja1', [-5 5])
```

patrz rysunek obok.



fminbnd(funkcja, x1, x2, opcje)

Dodatkowy (czwarty) parametr wejściowy jest wektorem, którego kolejne współrzędne są kolejnymi opcjami wpływającymi na przebieg obliczeń. Temat ten zostanie krótko omówiony na końcu rozdziału.

fminbnd(funkcja, x1, x2, opcje, p1, p2, ...)

Powyższe wywołanie zostało wprowadzone aby umożliwić użytkownikowi wywoływanie minimalizowanej funkcji z dodatkowymi parametrami (p1, p2, ...). Liczba tych parametrów zależy od potrzeb użytkownika.

Uwaga!

Parametry p1, p2, ... można podawać dopiero po wektorze opcje. Ze względu na to, że sposób definiowania wektora opcje nie został omówiony należy podczas wywoływania funkcji fminbnd podawać w jego miejsce wektor pusty [].

Przykład 2.

Należy obliczyć minimum funkcji $y = a x^2 + b x + c$ w przedziale $x \in [-5, 5]$, dla trzech zbiorów parametrów **a**, **b**, **c**:

- $a = 1, b = 2, c = 1;$
- $a = 1, b = 3, c = 1;$
- $a = 1, b = 4, c = 1;$

Rozwiązanie:

1. Należy utworzyć funkcję obliczającą dla x wartość y zgodnie ze wzorem $y = a x^2 + b x + c$. Parametry równania kwadratowego a , b , c są dodatkowymi (oprócz x) parametrami wejściowymi funkcji. Funkcję zapisać w M-pliku np. pod nazwą Funkcja2.m.
2. Wywołać w oknie **Command Window** polecenie minimalizujące funkcję dla wymienionych zbiorów parametrów a , b , c .

1. Definicja funkcji

(funkcja zapisana w pliku „Funkcja2.m”)

```
function [y] = Funkcja1(x, a, b, c)
% Funkcja oblicza wartość y(x) = ax2 + bx + c
y = a*x2 + b*x + c;
```

2.1. Wywołanie minimalizacji dla pierwszego zbioru parametrów a, b, c:

```
fminbnd('Funkcja2', -5, 5, [], 1, 2, 1)
ans =
    -1
```

2.2. Wywołanie minimalizacji dla drugiego zbioru parametrów a, b, c:

```
fminbnd('Funkcja2', -5, 5, [], 1, 3, 1)
ans =
   -1.5000
```

2.2. Wywołanie minimalizacji dla drugiego zbioru parametrów a, b, c:

```
fminbnd('Funkcja2', -5, 5, [], 1, 4, 1)
ans =
    -2
```



9.2. Minimum funkcji wielu zmiennych – funkcje *fminsearch* i *fmins*

Podobnie jak w przypadku poszukiwania minimum funkcji jednej zmiennej istnieją dwie funkcje minimalizacji funkcji wielu zmiennych:

- funkcja **fminsearch** (nowa funkcja występująca w MATLAB-ie 6, nie występuje w poprzednich wersjach MATLAB-a),
- funkcja **fmins** (przestarzała funkcja można jej używać we wszystkich wersjach MATLAB-a, zostanie w następnych wersjach usunięta o czym informuje ostrzeżenie pojawiające się w trakcie wywołania tej funkcji w MATLAB-ie 6.5).

Przedstawiony poniżej sposób korzystania z funkcji **fminsearch** dotyczy również funkcji **fmins**.

Funkcja **fminsearch** może być wywołana na kilka różnych sposobów. Najmniejsza liczba parametrów wejściowych podczas wywołania funkcji wynosi 2, najmniejsza liczba parametrów wyjściowych wynosi 1.

fminsearch(funkcja, x0)

Funkcja **fminsearch** wywołana z 2 parametrami, oblicza minimum funkcji której nazwa podana jest w pierwszym parametrze, parametr drugi wyznacza punkt startowy, od którego poszukiwania minimum zostaną rozpoczęte. Parametrem wyjściowym (wartością zwracaną przez funkcję) jest znaleziony punkt z dziedziny, dla którego w podanym przedziale funkcja osiąga wartość minimalną.

Przykład 1.

Należy obliczyć minimum funkcji $y = (x_1 - 1)^2 + (x_2 - 2)^2$ poszukując minimum począwszy od punktu $x_0 = [10, 10]$.

Rozwiązanie:

1. Należy utworzyć funkcję obliczającą dla x_1 i x_2 wartość y zgodnie ze wzorem $y = (x_1 - 1)^2 + (x_2 - 2)^2$, zapisać w M-pliku np. pod nazwą Funkcja3.m
2. Wywołać w oknie **Command Window** polecenie minimalizujące funkcję.

Uwaga:

Funkcja minimalizująca **fminsearch** wymaga aby minimalizowana przez nią funkcja była funkcją o nagłówku: **function [y] = fun (x)** (użytkownik może oczywiście zmienić nazwę parametrów: wejściowego – **x**, wyjściowego – **y**, czy nazwę funkcji – **fun**).



Parametr wyjściowy musi być wartością liczbową (nie wektorem).

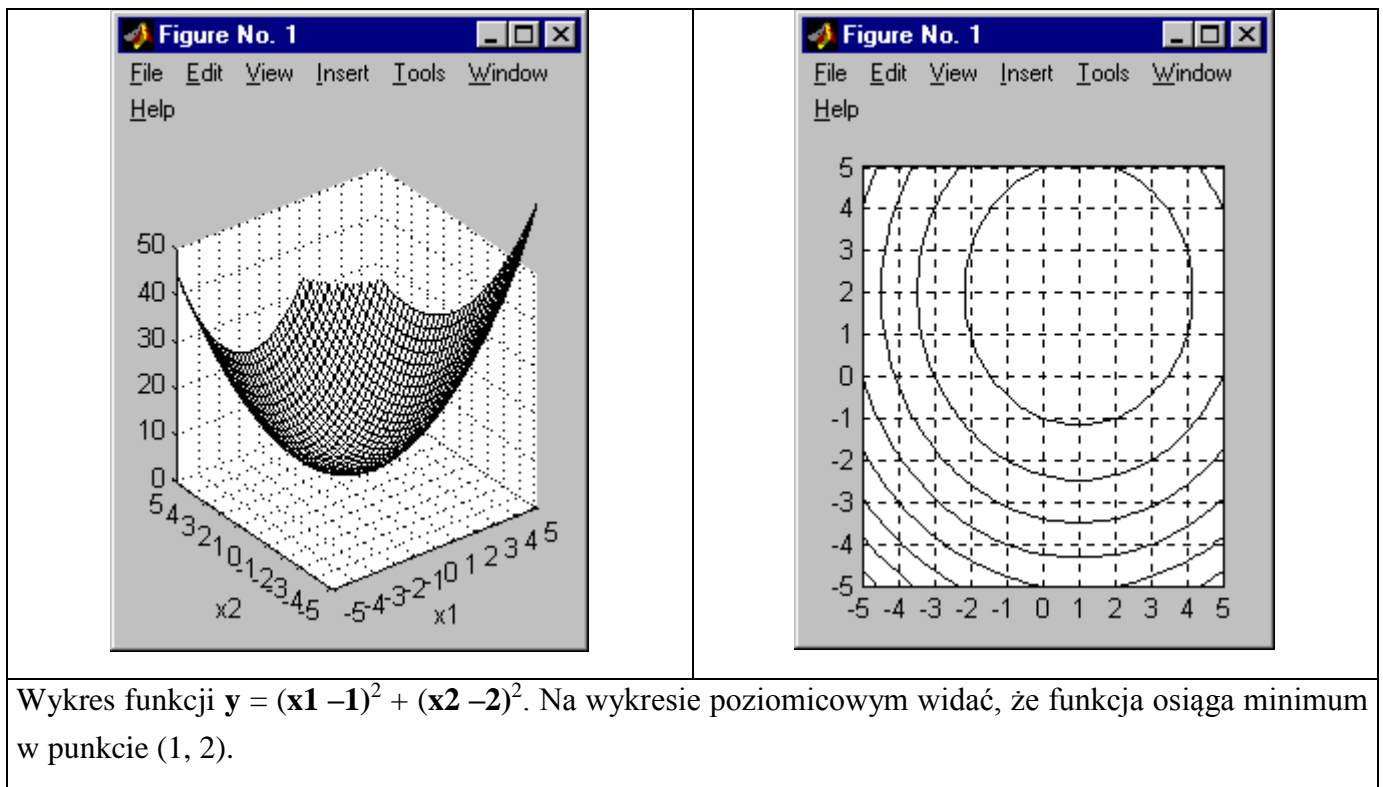
Parametr wejściowy może być wartością liczbową – jeżeli minimalizowana funkcja jest funkcją jednej zmiennej. Jeżeli minimalizowana funkcja jest funkcją wielu zmiennych to poszczególne elementy wektora wejściowego należy traktować jako kolejne zmienne od których funkcja jest zależna.

Np.: Jeżeli funkcja jest funkcją dwóch zmiennych x_1 i x_2 to pierwszy element parametru wejściowego funkcji (np. x) jest pierwszą zmienną ($x(1)$ to zmienna x_1), drugi element jest drugą zmienną ($x(2)$ to zmienna x_2).

Podsumowując: Jeżeli minimalizowana funkcja jest funkcją np.: $y = x_1^2 + x_2^2$ to funkcja ta powinna być zapisana jako:

```
function [y] = fun (x)
```

```
y = x(1)^2 + x(2)^2;
```



1. Definicja funkcji

(funkcja zapisana w pliku „Funkcja3.m”)

```
function [y] = Funkcja3(x)
% Funkcja oblicza wartość
% y(x1, x2) = (x1 - 1)^2 + (x2 - 2)^2

y = (x(1)-1)^ 2 + (x(2)-2)^ 2;
```

2. Wywołanie minimalizacji

```
fminsearch('Funkcja3', [10 10])
ans =
1.0000 2.0000
```



Funkcja **fminsearch** może być również wykorzystana do poszukiwania minimum funkcji jednej zmiennej.

Przykład 2.

Należy obliczyć minimum funkcji $y = x^2 + 3$ poszukując minimum począwszy od punktu $x_0 = -5$.

Rozwiązanie:

Wywołać w oknie **Command Window** polecenie minimalizujące funkcję. Skrypt definiujący funkcję został już napisany we wcześniejszym przykładzie: „Funkcja1.m”

```
fminsearch('Funkcja1', -5)
```

```
ans =
```

```
0
```

fminsearch(funkcja, x0, opcje)

Dodatkowy (trzeci) parametr wejściowy jest wektorem, którego kolejne współrzędne są kolejnymi opcjami wpływającymi na przebieg obliczeń. Temat ten zostanie krótko omówiony na końcu rozdziału.

fminsearch(funkcja, x0, opcje, p1, p2, ...), fmins(funkcja, x0, opcje, [], p1, p2, ...),

Powyższe wywołanie zostało wprowadzone aby umożliwić użytkownikowi wywoływanie minimalizowanej funkcji z dodatkowymi parametrami (p1, p2, ...). Liczba tych parametrów zależy od potrzeb użytkownika.

Przykład 3

Należy obliczyć minimum funkcji $y = (x_1 + a)^2 + (x_2 + b)^2$, poszukując minimum począwszy od punktu $x_0 = [10, 10]$ dla dwóch zbiorów parametrów **a**, **b**:

- $a = -1, b = -2$
- $a = 1, b = 2$



Rozwiązanie:

1. Należy utworzyć funkcję obliczającą wartość y na podstawie wartości x_1 i x_2 zgodnie ze wzorem $y = (x_1 + a)^2 + (x_2 + b)^2$. Funkcję zapisać w M-pliku np. pod nazwą Funkcja4.m.
2. Wywołać w oknie **Command Window** polecenie minimalizujące funkcję dla wymienionych zbiorów parametrów a, b .

1. Definicja funkcji

(funkcja zapisana w pliku „Funkcja4.m”)

```
function [y] = Funkcja4(x, a, b)
```

```
% Funkcja oblicza wartość  $y(x_1, x_2) = (x_1 + a)^2 + (x_2 + b)^2$ 
```

```
y = (x(1) + a)^2 + (x(2) + b)^2;
```

2.1. Wywołanie minimalizacji dla pierwszego zbioru parametrów a, b :

```
fminsearch('Funkcja4', [10, 10], [], -1, -2)
```

```
ans =
```

```
1.0000 2.0000
```

2.2. Wywołanie minimalizacji dla drugiego zbioru parametrów a, b :

```
fminsearch('Funkcja4', [10, 10], [], 1, 2)
```

```
ans =
```

```
-1.0000 -2.0000
```

9.3. Znaczenie wektora opcje w funkcjach minimalizujących**9.3.1. funkcje *fminbnd*, *fminsearch***

Ustawienia wektora opcji wpływają zarówno na przebieg obliczeń jak i na sposób wyświetlania wyników. Do generowania wektora opcji należy wykorzystać funkcję MATLAB-a: **optimset**. Funkcje minimalizujące wykorzystują opcje:



Opcja	Znaczenie
Display	Opcja wpływa na sposób wyświetlania wyników. Może przyjmować jedną z wartości: off , iter , notify , final . Ustawienie opcji Display wartości iter powoduje wyświetlanie wyników pośrednich. Domyślnie wyniki pośrednie nie są wyświetlane.
TolX	Opcja wpływa na wymaganą dokładność poszukiwanego rozwiązania. Można ustawić np. na 0.1 (rozwiązanie zostanie znalezione z dokładnością do 0.1) czy na 1.0 e-4 (rozwiązanie zostanie znalezione z dokładnością do 0.0001). Zwiększanie dokładności wpływa na wydłużenie czasu obliczeń. Domyślnie opcja ta ma ustawioną wartość 1.0 e-4.
TolFun	Opcja wpływa na wymaganą dokładność wyznaczenia wartości funkcji w punkcie optymalnym.
MaxFunEvals	Opcja definiuje maksymalną liczbę wywołań minimalizowanej funkcji.
MaxIter	Opcja definiuje maksymalną liczbę iteracji procesu minimalizacji.

Ustawienia opcji **TolX**, **TolFun**, **MaxFunEvals**, **MaxIter** wyznaczają warunek zakończenia procesu poszukiwania minimum. Proces ten jest więc zatrzymywany:

- jeżeli kolejne przybliżenia punktu w którym znajduje się minimum będą mniejsze od **TolX** lub
- jeżeli kolejne przybliżenia wartości funkcji będą mniejsze od **TolFun** lub
- jeżeli algorytm minimalizacji wykona większą liczbę przebiegów niż **MaxIter** lub
- jeżeli podczas procesu poszukiwania minimum minimalizowana funkcja została wywołana więcej razy niż **MaxFunEvals**

Aby wygenerować wektor opcji, w którym:

- opcja **Display** otrzyma wartość **iter**,
- opcja **TolX** otrzyma wartość **0.01**

należy wykonać polecenie:

```
>> optimset('Display', 'iter', 'TolX', 0.01)
```

Z tak zdefiniowanymi opcjami można teraz np. wywołać minimalizację funkcji z przykładu 1:

```
>> fminsearch('Funkcja3', [10 10], optimset('Display', 'iter', 'TolX', 0.01))
```



1	3	145	initial	18	32	0.356261	contractinside
2	5	140.563	expand	19	34	0.0968294	contractinside
3	7	122.266	expand	20	36	0.0968294	contractinside
4	9	114.285	expand	21	38	0.0613201	contractoutside
5	11	84.6963	expand	22	40	0.00740013	contractinside
6	12	84.6963	reflect	23	42	0.00740013	contractoutside
7	13	84.6963	reflect	24	44	0.00740013	contractinside
8	15	51.2288	expand	25	46	0.00314787	reflect
9	16	51.2288	reflect	26	48	0.00235711	contractoutside
10	18	22.0156	expand	27	50	0.00152579	contractinside
11	20	14.1338	reflect	28	52	0.000100887	contractinside
12	22	2.52954	reflect	29	54	0.000100887	contractinside
13	23	2.52954	reflect	30	55	0.000100887	reflect
14	25	2.52954	contractoutside	31	57	0.000100887	contractinside
15	27	1.48232	contractinside	32	59	0.000100887	contractoutside
16	28	1.48232	reflect	33	61	3.32384e-005	contractinside
17	30	0.528996	contractinside	34	63	8.42377e-006	contractinside

ans =

1.0025 2.0025

9.3.2. funkcje *fmin*, *fmins*

Funkcje **fmin** i **fmins** wymagają aby opcje były wprowadzane do wektora. Poniżej przedstawiono tabelę zawierającą zestaw opcji dla funkcji **fminbnd** i **fminsearch** wraz z ich odpowiednikami dla funkcji **fmin** i **fmins**.

Opcja (funkcje: fminbnd , fminsearch)	Opcja (funkcje: fmin , fmins)
Display	Należy ustawić pierwszy element wektora opcji. Ustawienie wartości dodatniej odpowiada ustawieniu opcji Display na iter , ustawienie wartości 0 odpowiada ustawieniu opcji Display na off .
TolX	Dokładność poszukiwanego rozwiązania należy wprowadzić do drugiego elementu wektora opcji.
TolFun	Dokładność wyznaczenia wartości funkcji w punkcie optymalnym należy



	wprowadzić do trzeciego elementu wektora opcji.
MaxFunEvals	Maksymalną liczbę wywołań minimalizowanej funkcji należy wprowadzić do czternastego elementu wektora.

Aby wykonać minimalizację funkcji z przykładu 1 (z wyświetlaniem wyników pośrednich i tolerancją $x = 0.01$) za pomocą funkcji **fmins** należy wykonać polecenie:

```
>> fmins('Funkcja3', [10 10], [1 0.01])
```

Ćwiczenia

1. Znajdź minimum funkcji $y = x^2 + 5x + 3$. Jako punkt startowy przyjmij wartość **5**. Wygeneruj wykres funkcji i sprawdź czy wykres potwierdza otrzymany przez Ciebie rezultat.
2. Znajdź minimum funkcji $y = 100 (x_2 - x_1^2)^2 + (1 - x_1)^2$. Jako punkt startowy przyjmij punkt: **[-10, -10]**. Poszukiwane minimum znajduje się w punkcie [1, 1]. Wartość funkcji w tym punkcie wynosi 0. Wywołaj funkcję minimalizującą tak aby były widoczne rezultaty pośrednich kroków obliczeń. Sprawdź czy wartość funkcji w punkcie [1, 1] wynosi 0.
3. Znajdź minimum funkcji $y = a (x_2 - x_1^2)^2 + (b - x_1)^2$ dla dwóch zbiorów parametrów **a, b**:
 - $a = 100, b = -1$ (minimum w punkcie [-1, 1])
 - $a = 100, b = 1$ (minimum w punkcie [1, 1])

Wartości **a** i **b** powinny być parametrami wejściowymi funkcji. Jako punkt startowy przyjmij punkt: **[-10, -10]**.

