

Optymalizacja nieliniowa

Część II. Optima funkcji wielu zmiennych bez ograniczeń

- Metody gradientowe
- Metody bezgradientowe (bezpośrednich poszukiwań)

WPROWADZENIE

Zadania optymalizacji to najczęściej zagadnienia wielowymiarowe, w których funkcje celu są funkcjami wielu zmiennych, tzn.

$$f : \mathbb{R}^n \rightarrow \mathbb{R}.$$

Metody optymalizacji wielowymiarowej można podzielić na metody gradientowe i metody bezgradientowe.

Metody bezgradientowe wykorzystują jedynie informacje o wartości funkcji celu. Niektóre z nich do swojego działania nie potrzebują nawet założenia o ciągłości.

Metody gradientowe potrzebują natomiast założenia nie tylko o ciągłości, ale i o różniczkowalności funkcji celu w całej przestrzeni \mathbb{R}^n .

METODY GRADIENTOWE

Metody gradientowe w swoim działaniu wykorzystują informacje o gradiencie i hesjanie funkcji celu oraz o wielkościach z nimi związanych.

Przypomnienie. Gradientem funkcji $f : \mathbb{R}^n \rightarrow \mathbb{R}$ w punkcie $\mathbf{x} = (x_1, x_2, \dots, x_n)$ nazywamy wektor pochodnych cząstkowych względem każdej zmiennej, tzn.

$$\nabla f(\mathbf{x}) = \left[\frac{\partial f(\mathbf{x})}{\partial x_1}, \frac{\partial f(\mathbf{x})}{\partial x_2}, \dots, \frac{\partial f(\mathbf{x})}{\partial x_n} \right]^T .$$

Gradient $\nabla f(\mathbf{x})$ wskazuje kierunek największego wzrostu funkcji. Natomiast wektor $-\nabla f(\mathbf{x})$ wskazuje kierunek największego spadku analizowanej funkcji i jest to własność wykorzystywana przez wszystkie algorytmy gradientowe.

Długość wektora gradientu w punkcie \mathbf{x} jest miarą nachylenia (stromości) wykresu funkcji f (określa szybkość wzrostu wartości w danym kierunku).

Ogólny algorytm metod gradientowych

Niech dana będzie różniczkowalna funkcja celu $f : \mathbb{R}^n \rightarrow \mathbb{R}$, której minimum poszukujemy.

Większość metod gradientowych wykorzystuje ten sam schemat działania, mianowicie dla danego punktu $\mathbf{x}^{(i)}$ (przybliżenia szukanego minimum uzyskanego w i -tej iteracji) wyznaczamy

- kierunek dalszych poszukiwań $\mathbf{d}^{(i)}$,
- długość kroku $h^{(i)}$, który należy wykonać zgodnie ze znalezionym wcześniej kierunkiem.

Pozwala to wyznaczyć kolejne przybliżenie szukanego minimum za pomocą wzoru:

$$\mathbf{x}^{(i+1)} = \mathbf{x}^{(i)} + h^{(i)} \mathbf{d}^{(i)}.$$

Powyższy schemat jest powtarzany aż do uzyskania zadanej dokładności (warunku stopu).

Ze względu na sposób doboru kroku, metody optymalizacji dzieli się na *stałokrokowe* (długość kroku stała w każdej iteracji) i *zmienokrokowe* (długość kroku optymalizowana w każdej iteracji).

Optymalizacja długości kroku odbywa się na zasadzie minimalizacji wartości funkcji celu wzdłuż kierunku poszukiwań, tzn. minimalizacji funkcji

$$g(\alpha) = f(\mathbf{x}^{(i)} + \alpha \mathbf{d}^{(i)}).$$

Dla wyznaczonego minimum α^* przyjmujemy wówczas

$$h^{(i)} = \alpha^*.$$

Uwaga. Zauważmy, że funkcja g jest funkcją jednej zmiennej. Oznacza to, że minimalizację funkcji celu wzdłuż kierunku poszukiwań możemy wykonać za pomocą jednej z metod optymalizacji jednowymiarowej.

Uwaga. Metody gradientowe są zazwyczaj szybciej zbieżne niż metody bezgradientowe. Jednakże w sytuacjach, w których brak jest opisu analitycznego funkcji i gradient musi być obliczany numerycznie, bardziej efektywne może okazać się stosowanie metod bezgradientowych.

Metoda Cauchy'ego największego spadku

Niech dana będzie różniczkowalna funkcja celu $f : \mathbb{R}^n \rightarrow \mathbb{R}$, której minimum poszukujemy.

Metoda Cauchy'ego jest najprostszą metodą optymalizacji wielowymiarowej wykorzystującą informację o gradiencie. Algorytm tej metody wynika wprost z własności gradientu mówiącej, że wektor $-\nabla f(\mathbf{x})$ wyznacza kierunek, w którym wartości funkcji f maleją najszybciej.

W metodzie największego spadku wektor kierunku, w i -tej iteracji, wyznaczamy ze wzoru:

$$\mathbf{d}^{(i)} = -\nabla f(\mathbf{x}^{(i)}).$$

Zgodnie z ogólnym algorytmem metod gradientowych, ciąg kolejnych przybliżeń szukanego minimum wyznaczamy ze wzoru

$$\mathbf{x}^{(i+1)} = \mathbf{x}^{(i)} - h^{(i)} \nabla f(\mathbf{x}^{(i)}), \quad i = 0, 1, 2, \dots,$$

gdzie $\mathbf{x}^{(0)}$ jest wybranym punktem startowym (pierwszym przybliżeniem).

Długość kroku $h^{(i)}$ może być optymalizowana w każdej iteracji (metoda zmiennokrokowa) lub może być stała w każdej (metoda stałokrokowa).

Uwaga. Metoda stałokrokowa, nazywana również metodą gradientu prostego, jest prostszą wersją metody Cauchy'ego. Może ona jednak nie być zbieżna, gdyż wykonywanie kroków o stałej długości nie zawsze przynosi spadek wartości funkcji celu w kolejnych iteracjach.

Na korzyść metody gradientu prostego przemawia fakt, że ustalona z góry długość kroku, powoduje, że nie musimy wykonywać dodatkowych obliczeń związanych z poszukiwaniem minimum na kierunku.

Uwaga. Metoda Cauchy'ego zmiennokrokowa jest zazwyczaj szybciej zbieżna.

Uwaga. Wektory kierunkowe $d^{(i)}$ wyznaczone w kolejnych iteracjach zmiennokrokowej metody Cauchy'ego są wzajemnie prostopadłe. Mianowicie, jeżeli przyjmiemy oznaczenie

$$g(\alpha) = f(\mathbf{x}^{(i)} + \alpha \mathbf{d}^{(i)}),$$

to wówczas

$$g'(\alpha) = \sum_{j=1}^n \frac{\partial f(\mathbf{x}^{(i)} + \alpha \mathbf{d}^{(i)})}{\partial x_j} d_j^{(i)} = \langle \nabla f(\mathbf{x}^{(i)} + \alpha \mathbf{d}^{(i)}), \mathbf{d}^{(i)} \rangle.$$

Po zoptymalizowaniu długości kroku możemy napisać, że

$$g'(h^{(i)}) = 0$$

oraz

$$\nabla f(\mathbf{x}^{(i)} + h^{(i)} \mathbf{d}^{(i)}) = \nabla f(\mathbf{x}^{(i+1)}) = -\mathbf{d}^{(i+1)}.$$

Co ostatecznie prowadzi do $\langle \mathbf{d}^{(i+1)}, \mathbf{d}^{(i)} \rangle = 0$.

Uwaga. Stałokrokowa metoda największego spadku może być użyta do optymalizacji funkcji jednej zmiennej.

Zaletą tej metody jest to, że nie musimy ustalać wstępnego przedziału poszukiwań.

Wadą natomiast to, że nie mamy wpływu na położenie kolejnych przybliżeń i w związku z tym nie możemy ustalić z góry potrzebnej liczby iteracji (długość wektora $h\mathbf{d}$ może być duża, nawet gdy samo h jest małe).

Metoda Newtona

Schemat działania metody Newtona jest podobny jak w metodzie największego spadku, z tym że poszukiwania prowadzone są w innym kierunku. Wektor kierunkowy w i -tej iteracji obliczamy według wzoru:

$$\mathbf{d}^{(i)} = -H^{-1}(\mathbf{x}^{(i)})\nabla f(\mathbf{x}^{(i)}),$$

gdzie $H(\mathbf{x})$ oznacza macierz Hessego (hesjan) funkcji celu f w punkcie \mathbf{x} .

Z określenia wektora kierunku wynika, że aby można było stosować tą metodę hesjan musi być określony w każdym punkcie. Co oznacza, że muszą istnieć wszystkie pochodne cząstkowe drugiego rzędu w każdym punkcie.

Przypomnienie. Hesjanem (macierzą Hessego) funkcji $f : \mathbb{R}^n \rightarrow \mathbb{R}$ w punkcie $\mathbf{x} = (x_1, x_2, \dots, x_n)$ nazywamy macierz drugich pochodnych (jeżeli istnieją) funkcji f w punkcie \mathbf{x} , tzn.

$$H(\mathbf{x}) = \begin{bmatrix} \frac{\partial^2 f(\mathbf{x})}{\partial x_1^2} & \frac{\partial^2 f(\mathbf{x})}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f(\mathbf{x})}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f(\mathbf{x})}{\partial x_2 \partial x_1} & \frac{\partial^2 f(\mathbf{x})}{\partial x_2^2} & \cdots & \frac{\partial^2 f(\mathbf{x})}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f(\mathbf{x})}{\partial x_n \partial x_1} & \frac{\partial^2 f(\mathbf{x})}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 f(\mathbf{x})}{\partial x_n^2} \end{bmatrix}.$$

Uwaga. We wzorze na wektor kierunkowy $H^{-1}(\mathbf{x})$ oznacza macierz odwrotną hesjanu. Metoda działa poprawnie, gdy jesteśmy w stanie obliczyć $H^{-1}(\mathbf{x}^{(i)})$. A to ma miejsce wtedy, gdy hesjan jest macierzą nieosobliwą w każdym z punktów $\mathbf{x}^{(i)}$.

Uwaga. Metoda Newtona nie będzie działać poprawnie, jeżeli

- hesjan $H(\mathbf{x}^{(i)})$ jest macierzą nieodwracalną,
- gradient $\nabla f(\mathbf{x}^{(i)}) = \mathbf{0}$.

Obydwa warunki oznaczają, że kierunku poszukiwań dla i -tej iteracji nie da się ustalić.

Uwaga. Metoda Newtona jest metodą stałokrokową z krokiem $h = 1$. Istnieje możliwość optymalizacji długości kroku w każdej iteracji, w wyżej opisany sposób - procedura nosi wtedy nazwę **metody Newtona-Raphsona**.

Metody gradientów sprzężonych

Metody gradientów sprzężonych narodziły się jako antidotum na podstawową wadę metod Cauchy'ego i Newtona. Mianowicie obie te metody w i -tej iteracji, minimalizują funkcję f w kierunku $\mathbf{d}^{(i)}$ a w kolejnej iteracji w kierunku $\mathbf{d}^{(i+1)}$, nie zachowując jednak optymalności względem poprzedniego kierunku. Metody gradientów sprzężonych, optymalizując funkcję celu w kierunku $\mathbf{d}^{(i+1)}$ biorą pod uwagę optymalność względem poprzedniego kierunku.

W metodach gradientów sprzężonych kierunki poszukiwań wyznacza się według zasady:

$$\mathbf{d}^{(0)} = -\nabla f(\mathbf{x}^{(0)}),$$

$$\mathbf{d}^{(i+1)} = -\nabla f(\mathbf{x}^{(i+1)}) + \beta^{(i+1)}\mathbf{d}^{(i)}, \text{ gdzie}$$

- w metodzie *Fletcher-Reevesa*:

$$\beta^{(i+1)} = \frac{\nabla f(\mathbf{x}^{(i+1)})^T \nabla f(\mathbf{x}^{(i+1)})}{\nabla f(\mathbf{x}^{(i)})^T \nabla f(\mathbf{x}^{(i)})} = \frac{(\|\nabla f(\mathbf{x}^{(i+1)})\|_2)^2}{(\|\nabla f(\mathbf{x}^{(i)})\|_2)^2};$$

- w metodzie *Polaka-Ribiere'a*:

$$\beta^{(i+1)} = \frac{\nabla f(\mathbf{x}^{(i+1)})^T (\nabla f(\mathbf{x}^{(i+1)}) - \nabla f(\mathbf{x}^{(i)}))}{\nabla f(\mathbf{x}^{(i)})^T \nabla f(\mathbf{x}^{(i)})};$$

- w metodzie *Hestenesa-Stiefela*:

$$\beta^{(i+1)} = \frac{\nabla f(\mathbf{x}^{(i+1)})^T (\nabla f(\mathbf{x}^{(i+1)}) - \nabla f(\mathbf{x}^{(i)}))}{(\mathbf{d}^{(i)})^T (\nabla f(\mathbf{x}^{(i+1)}) - \nabla f(\mathbf{x}^{(i)}))}.$$

Metody quasi-newtonowskie

W metodzie Newtona wektor kierunku obliczany jest w każdym kroku jako iloczyn odwrotności hesjanu przez gradient. W zastosowaniach praktycznych obliczanie odwrotności macierzy Hessego może być skomplikowane i pracochłonne.

W metodach quasi-newtonowskich odwrotność hesjanu jest zastąpiona jego pewną aproksymacją. Wektor kierunku optymalizacji dany jest wzorem:

$$\mathbf{d}^{(i)} = -V^{(i+1)}\nabla f(\mathbf{x}^{(i)}),$$

gdzie $V^{(i)}$ jest symetryczną i dodatnio określoną macierzą kwadratową przybliżającą odwrotność hesjanu funkcji celu f , tzn. $V^{(i+1)} \approx H^{-1}(\mathbf{x}^{(i)})$. Różnice między metodami polegają na odmiennym sposobie wyznaczania macierzy $V^{(i)}$.

Przypomnienie. Macierz kwadratową A nazywamy symetryczną, gdy zachodzi $A^T = A$.

Macierz kwadratową A nazywamy dodatnio (ujemnie) określoną, jeżeli dla każdego wektora $\mathbf{x} \neq \mathbf{0}$ zachodzi $\mathbf{x}^T A \mathbf{x} > 0$ ($\mathbf{x}^T A \mathbf{x} < 0$).

Definicja. Niech dana będzie macierz kwadratowa stopnia n

$$A = \begin{bmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nn} \end{bmatrix}.$$

Wyznacznikami Sylwestera dla macierzy A nazywamy liczby $\Delta_1, \dots, \Delta_n$,
gdzie

$$\Delta_i = \det \begin{bmatrix} a_{11} & \cdots & a_{1i} \\ \vdots & \ddots & \vdots \\ a_{i1} & \cdots & a_{ii} \end{bmatrix}$$

dla $i = 1, 2, \dots, n$.

Twierdzenie Sylwestera. Niech będzie macierz kwadratowa A stopnia n oraz niech $\Delta_1, \dots, \Delta_n$ będą jej wyznacznikami Sylwestera. Zachodzą następujące warunki:

- macierz A jest dodatnio określona wtedy i tylko wtedy, gdy

$$\Delta_i > 0 \text{ dla } i = 1, 2, \dots, n;$$

- macierz A jest ujemnie określona wtedy i tylko wtedy, gdy

$$(-1)^i \Delta_i > 0 \text{ dla } i = 1, 2, \dots, n.$$

Metoda Davidona-Fletcher-Powella (DFP)

W metodzie Davidona-Fletcher-Powella macierz V wyznaczamy rekurencyjnie przyjmując:

$$V^{(0)} = I,$$

gdzie I oznacza macierz jednostkową, a następnie

$$V^{(i+1)} = V^{(i)} + A^{(i)} + B^{(i)},$$

gdzie macierze $A^{(i)}$ oraz $B^{(i)}$ obliczane są następująco:

$$A^{(i)} = \frac{\mathbf{r}^{(i)} (\mathbf{r}^{(i)})^T}{(\mathbf{r}^{(i)})^T \mathbf{s}^{(i)}}, \quad B^{(i)} = -\frac{V^{(i)} \mathbf{s}^{(i)} (\mathbf{s}^{(i)})^T V^{(i)}}{(\mathbf{s}^{(i)})^T V^{(i)} \mathbf{s}^{(i)}},$$

przy czym

$$\mathbf{s}^{(i)} = \nabla f(\mathbf{x}^{(i)}) - \nabla f(\mathbf{x}^{(i-1)}), \quad \mathbf{r}^{(i)} = \mathbf{x}^{(i)} - \mathbf{x}^{(i-1)}.$$

Uwaga. Zauważmy, że

$$\mathbf{r}^{(0)} = \mathbf{x}^{(0)} - \mathbf{x}^{(-1)}.$$

Zatem, aby rozpocząć proces iteracyjny potrzebujemy dwóch punktów $\mathbf{x}^{(0)}$ i $\mathbf{x}^{(-1)}$. W praktyce przyjmujemy $\mathbf{x}^{(-1)}$ jako punkt startowy, a punkt $\mathbf{x}^{(0)}$ otrzymujemy wykonując jeden krok innej metody, np. największego spadku.

Uwaga. Proces iteracyjny możemy rozpocząć od macierzy $V^{(0)}$ innej niż jednostkowa. Należy przy tym pamiętać, aby przyjęta macierz $V^{(0)}$ była symetryczna i dodatnio określona.

Metoda Broydena-Fletcher-Goldfarba-Shanno (BFGS)

W metodzie Broydena-Fletcher-Goldfarba-Shanno macierz V wyznaczamy rekurencyjnie przyjmując:

$$V^{(0)} = I, \quad V^{(i+1)} = V^{(i)} + (1 + C^{(i)})A^{(i)} + D^{(i)},$$

gdzie macierze $A^{(i)}$, $B^{(i)}$ i $D^{(i)}$ obliczane są następująco:

$$A^{(i)} = \frac{\mathbf{r}^{(i)} (\mathbf{r}^{(i)})^T}{(\mathbf{r}^{(i)})^T \mathbf{s}^{(i)}}, \quad C^{(i)} = \frac{(\mathbf{s}^{(i)})^T V^{(i)} \mathbf{s}^{(i)}}{(\mathbf{r}^{(i)})^T \mathbf{s}^{(i)}},$$

$$D^{(i)} = -\frac{\mathbf{r}^{(i)} (\mathbf{s}^{(i)})^T V^{(i)} + V^{(i)} \mathbf{s}^{(i)} (\mathbf{r}^{(i)})^T}{(\mathbf{r}^{(i)})^T \mathbf{s}^{(i)}}$$

przy czym

$$\mathbf{s}^{(i)} = \nabla f(\mathbf{x}^{(i)}) - \nabla f(\mathbf{x}^{(i-1)}), \quad \mathbf{r}^{(i)} = \mathbf{x}^{(i)} - \mathbf{x}^{(i-1)}.$$

Uwaga. Szczegóły dotyczące punktów startowych i początkowej macierzy V są identyczne jak dla metody DFP.

METODY BEZGRADIENTOWE **(bezpośrednich poszukiwań)**

Metody bezgradientowe poszukują minimum funkcji celu bazując jedynie na porównywaniu jej wartości w wybranych punktach. Różnice między metodami polegają na doborze punktów w których testowane są wartości funkcji.

Metoda Hooke'a-Jeeves'a

Metoda Hooke'a-Jeeves'a jest jedną z najprostszych metod bezgradientowych optymalizacji wielowymiarowej. Kierunki poszukiwań w tej metodzie są zgodne w wersorami układu współrzędnych i są niezmiennie w całym procesie poszukiwań, natomiast długości kroków wyznaczone są na zasadzie prostych zależności.

Metoda Hooke'a-Jeeves'a nie jest typową metodą iteracyjną, gdyż nie da się wskazać momentu w którym kończy się jedna iteracja i zaczyna następna.

Metoda składa się z dwóch etapów: próbnego i roboczego. Algorytm rozpoczyna działanie w wybranym punkcie początkowym x , który następnie jest przekształcany w kolejnych etapach.

Etap próbny. Celem tego etapu jest zbadanie lokalnego zachowania się funkcji w otoczeniu bieżącego punktu \mathbf{x} , z którego wykonujemy krok o wybranej długości s , kolejno w każdym z kierunków \mathbf{e}^j bazy kanonicznej \mathbb{R}^n , gdzie \mathbf{e}^j dla $j = 1, 2, \dots, n$ są wersorami.

Celem etapu próbnego jest wyznaczenie tzw. *punktu bazowego* \mathbf{x}^B dla etapu roboczego. Jako początkowy punkt bazowy przyjmujemy \mathbf{x} , tzn. $\mathbf{x}^B = \mathbf{x}$.

Dla każdego kierunku \mathbf{e}^j , gdzie $j = 1, 2, \dots, n$ prowadzimy następujące rozumowanie:

- jeżeli zachodzi $f(\mathbf{x}^B + s\mathbf{e}^j) < f(\mathbf{x}^B)$ (krok pomyślny), to otrzymujemy nowy punkt bazowy:

$$\mathbf{x}^B \leftarrow \mathbf{x}^B + s\mathbf{e}^j;$$

- w przeciwnym wypadku (krok niepomyślny) sprawdzamy czy zachodzi $f(\mathbf{x}^B - s\mathbf{e}^j) < f(\mathbf{x}^B)$. Jeżeli zachodzi to otrzymujemy nowy punkt bazowy:

$$\mathbf{x}^B \leftarrow \mathbf{x}^B - s\mathbf{e}^j.$$

Jeżeli etap próbny nie przynosi poprawy wartości funkcji celu w żadnym kierunku (wartości nie zmniejszają się), w stosunku do wartości w początkowym punkcie bazowym (tzn. punkt bazowy pozostaje bez zmian), wówczas zmniejszamy krok poszukiwań s , tzn. przyjmujemy $s \leftarrow \alpha s$, gdzie $\alpha \in (0, 1)$. Następnie etap próbny jest powtarzany z nową długością kroku.

W przypadku, gdy etap próbny kończy się znalezieniem nowego punktu bazowego, stary punkt bazowy z początku etapu oznaczamy $\bar{\mathbf{x}}^B$.

Uwaga. Etap próbny kończy się, gdy poszukiwanie było pomyślne w chociaż jednym z kierunków, tzn. został znaleziony nowy punkt bazowy lub na skutek skracania długości kroku, jego długość jest mniejsza niż zadana dokładność obliczeń.

Etap roboczy. Etap rozpoczyna się w bieżącym punkcie bazowym. Wyznaczamy kolejny punkt według wzoru:

$$\mathbf{x} = \mathbf{x}^B + (\mathbf{x}^B - \bar{\mathbf{x}}^B) = 2\mathbf{x}^B - \bar{\mathbf{x}}^B,$$

w którym wykonujemy pojedynczy krok etapu próbnego. W przypadku kroku niepomyślnego nie zmniejszamy długości kroku i nie kontynuujemy, ale pozostajemy w punkcie \mathbf{x} . Dalej postępujemy według zasady:

- jeżeli punkt zwrócony w przeprowadzonym etapie próbnym przynosi zmniejszenie wartości funkcji celu w stosunku do wartości w ostatnim punkcie bazowym, znaleziony punkt staje się nowym punktem bazowym dla którego ponownie wykonujemy etap roboczy;
- jeżeli punkt zwrócony w przeprowadzonym etapie próbnym nie przynosi zmniejszenia wartości funkcji celu w stosunku do $f(\mathbf{x}^B)$, wracamy do punktu bazowego dla którego rozpoczynamy pełny etap próbny.

Metoda sympleksowa Neldera-Meada

Metoda sympleksowa Neldera-Meada działa na zasadzie porównywania wartości funkcji celu jednocześnie w kilku wybranych punktach (innych dla każdej iteracji). Wybrane punkty są wierzchołkami sympleksu.

Przypomnienie. Sympleksem n -wymiarowym o $n + 1$ wierzchołkach $\mathbf{x}^1, \mathbf{x}^2, \dots, \mathbf{x}^{n+1}$ nazywamy najmniejszy zbiór wypukły zawierający te punkty, przy czym zbiór $\{\mathbf{x}^j - \mathbf{x}^1 : 2 \leq j \leq n\}$ musi składać się z wektorów liniowo niezależnych.

Uwaga. W przestrzeni \mathbb{R}^2 sympleksem jest dowolny trójkąt, a w przestrzeni \mathbb{R}^3 dowolny czworościan.

Zgodnie z ideą metody Nelder-Mead, startowy sympleks jest przekształcany za pomocą pewnych elementarnych operacji geometrycznych nazywanych: odbiciem, ekspansją, zawężeniem i redukcją. W wyniku każdej z nich „najgorszy” wierzchołek sympleksu (w którym wartość funkcji celu jest największa) jest zastępowany przez inny - „lepszy”. W ten sposób kolejne sympleksy, coraz lepiej lokalizują lokalne minimum badanej funkcji.

Procedura działa iteracyjnie i kończy działanie, jeżeli różnica między największą i najmniejszą wartością w wierzchołkach aktualnego sympleksu jest nie większa od przyjętej dokładności.

Rozważmy funkcję $f : \mathbb{R}^n \rightarrow \mathbb{R}$ unimodalną w minimum. W przestrzeni \mathbb{R}^n sympleks ma $n + 1$ wierzchołków: $\mathbf{x}^1, \mathbf{x}^2, \dots, \mathbf{x}^{n+1}$.

Algorytm w każdej iteracji rozpoczyna działanie od

1) uporządkowania wierzchołków sympleksu (aktualizacja indeksów) zgodnie ze wzrostem wartości funkcji celu, tzn.

$$f(\mathbf{x}^i) \leq f(\mathbf{x}^{i+1}),$$

co oznacza, że w \mathbf{x}^1 funkcja przyjmuje najmniejszą wartość, a w \mathbf{x}^{n+1} - największą;

oraz

2) wyznaczenia środka ciężkości sympleksu (z wyłączeniem najgorszego wierzchołka \mathbf{x}^{n+1}), według wzoru:

$$\mathbf{x}^b = \frac{1}{n} \sum_{i=1}^n \mathbf{x}^i.$$

Procedura wykorzystuje trzy dodatnie parametry α, β, γ spełniające warunki: $\alpha \in (0, 1]$, $\beta < 1$ i $\gamma > 1$, będące współczynnikami odbicia, zawężenia i ekspansji. Standardowo i dla zapewnienia dobrej zbieżności przyjmuje się $\alpha = 1$, $\beta = 0.5$ i $\gamma = 2$.

Algorytm w każdej iteracji stara się poprawić lokalizację najgorszego punktu \mathbf{x}^{n+1} , tzn. poszukuje punktu który daje mniejszą wartość funkcji celu. Poszukiwanie tego punktu prowadzone jest na prostej przechodzącej przez punkty \mathbf{x}^b i \mathbf{x}^{n+1} i jest realizowane w następujących krokach:

Krok 1. Odbicie. Wyznaczamy punkt

$$\mathbf{x}^r = (1 + \alpha)\mathbf{x}^b - \alpha\mathbf{x}^{n+1}$$

będący odbiciem punktu \mathbf{x}^{n+1} względem punktu \mathbf{x}^b . Jeżeli

$$f(\mathbf{x}^r) < f(\mathbf{x}^1),$$

to przechodzimy do kroku 2, w przeciwnym wypadku do kroku 3.

Krok 2. Ekspansja. Wartość w punkcie \mathbf{x}^r jest mniejsza niż wartość w najlepszym wierzchołku, co sugeruje dalsze poszukiwania w tym kierunku. Wyznaczamy punkt

$$\mathbf{x}^e = \gamma\mathbf{x}^r + (1 - \gamma)\mathbf{x}^b,$$

oddalony γ razy dalej od punktu \mathbf{x}^b niż punkt \mathbf{x}^r . Jeżeli

$$f(\mathbf{x}^e) < f(\mathbf{x}^r),$$

to wierzchołek \mathbf{x}^{n+1} zastępowany jest przez \mathbf{x}^e , a w przeciwnym wypadku przez \mathbf{x}^r .

Krok 3. Jeżeli

$$f(\mathbf{x}^r) < f(\mathbf{x}^{n+1}),$$

to wierzchołek \mathbf{x}^{n+1} zastępowany jest przez \mathbf{x}^r , a w przeciwnym wypadku przechodzimy do kroku 4.

*Krok 4. **Zawężenie.*** Wyznaczamy punkt

$$\mathbf{x}^c = (1 - \beta)\mathbf{x}^{n+1} + \beta\mathbf{x}^b.$$

Jeżeli

$$f(\mathbf{x}^c) < f(\mathbf{x}^{n+1}),$$

to wierzchołek \mathbf{x}^{n+1} zastępowany jest przez \mathbf{x}^c , a w przeciwnym wypadku przechodzimy do kroku 5.

Krok 5. Redukcja. Krok ten jest wykonywany, gdy nie udało się znaleźć punktu o mniejszej wartości niż $f(\mathbf{x}^{n+1})$. W takiej sytuacji wszystkie punkty są przybliżane do najlepszego zgodnie ze wzorem:

$$\mathbf{x}^i \leftarrow (1 - \beta)\mathbf{x}^1 + \beta\mathbf{x}^i, \quad i = 2, 3, \dots, n + 1.$$

Procedura działa iteracyjnie powtarzając powyższe kroki do momentu kiedy różnica między największą i najmniejszą wartością w wierzchołkach sympleksu staje się mniejsza lub równa od założonej dokładności.