

Ergänzungen zu dem Buch  
**Numerische Verfahren zur Lösung  
unrestringierter Optimierungsaufgaben**  
von Carl Geiger und Christian Kanzow  
(Springer-Verlag, 1999)

## Das Nelder-Mead-Verfahren

Sei  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  eine (nicht notwendig differenzierbare) Funktion. Zur Bestimmung eines Minimums von  $f$  stellen wir in diesem Abschnitt das so genannte *Nelder-Mead-Verfahren* vor, bei dem es sich im Wesentlichen um eine heuristische Methode handelt, welches zu der Klasse der *ableitungsfreien Verfahren* gehört, da es keinerlei Informationen über den (evtl. gar nicht existierenden) Gradienten  $\nabla f(x)$  oder gar die Hesse-Matrix  $\nabla^2 f(x)$  benötigt.

Die Idee des Nelder-Mead-Verfahrens besteht darin, eine Folge von Simplexen zu erzeugen, die im Idealfall einen immer kleiner werdenden Durchmesser aufweisen und sich schließlich um das gesuchte Minimum herumscharen. Aus diesem Grunde wird die Methode von Nelder-Mead oft auch als *Simplex-Verfahren* bezeichnet. Es hat allerdings nichts mit dem bekannten Simplex-Verfahren für lineare Programme zu tun.

Zur Beschreibung der Nelder-Mead-Methode seien  $x^1, \dots, x^{n+1} \in \mathbb{R}^n$  paarweise verschiedene Punkte (ihre konvexe Hülle spannt anschaulich dann einen Simplex im  $\mathbb{R}^n$  auf). Seien

- $x^b$  der beste Punkt unter den  $x^i$ , also  $f(x^b) = \min_{i=1, \dots, n+1} f(x^i)$
- $x^s$  der schlechteste Punkt unter den  $x^i$ , also  $f(x^s) = \max_{i=1, \dots, n+1} f(x^i)$
- $x^z$  der zweitschlechteste Punkt, also  $f(x^z) = \max_{x^i \neq x^s} f(x^i)$ .

Die Punkte  $x^b, x^s$  und  $x^z$  sind hierbei nicht notwendig eindeutig bestimmt.

Formal lautet das Nelder-Mead-Verfahren dann wie folgt.

**Algorithmus 1** (*Nelder-Mead-Verfahren*)

Wähle Parameter  $\alpha > 0, \beta > 1$  und  $0 < \gamma < 1$  sowie ein Startpolyeder mit den Ecken  $x^1, x^2, \dots, x^{n+1} \in \mathbb{R}^n$ .

REPEAT

Bestimme  $x^b, x^s$  und  $x^z$ .

Bestimme das Zentrum der  $n$  besseren Punkte:  $\bar{x} := \frac{1}{n} \sum_{\substack{i=1 \\ i \neq s}}^{n+1} x^i$ .

Reflektiere den schlechtesten Punkt an  $\bar{x}$ :  $x^r := \bar{x} + \alpha(\bar{x} - x^s)$ .

Unterscheide 3 Fälle:

IF  $f(x^b) \leq f(x^r) \leq f(x^z)$

```

    Ersetze  $x^s$  durch  $x^r$ .
ELSEIF  $f(x^r) < f(x^b)$ 
    Bestimme  $x^e := \bar{x} + \beta(x^r - \bar{x})$ .
    IF  $f(x^e) < f(x^b)$ 
        Ersetze  $x^s$  durch  $x^e$ .
    ELSE
        Ersetze  $x^s$  durch  $x^r$ .
    END
ELSEIF  $f(x^r) > f(x^z)$ 
    IF  $f(x^r) \geq f(x^s)$ 
         $x^k := \bar{x} + \gamma(x^s - \bar{x})$ ;
    ELSE
         $x^k := \bar{x} + \gamma(x^r - \bar{x})$ ;
    END
    IF  $f(x^k) < f(x^s)$ 
        Ersetze  $x^s$  durch  $x^k$ .
    ELSE
        Ziehe Polyeder um  $x^b$  zusammen:  $x^i := \frac{1}{2}(x^i + x^b) \ \forall i = 1, \dots, n+1$ .
    END
END
UNTIL Abbruch.

```

Wir diskutieren im Folgenden etwas genauer den anschaulichen Hintergrund der einzelnen Schritte im Algorithmus 1 und verweisen hierzu insbesondere auch auf die Abbildung 1. Im Fall  $f(x^b) \leq f(x^r) \leq f(x^z)$  ist der reflektierte Punkt  $x^r$  zwar nicht optimal, aber doch ziemlich gut, so dass wir den bisher schlechtesten Punkt  $x^s$  durch  $x^r$  ersetzen.

Ist dagegen  $f(x^r) < f(x^b)$ , so handelt es sich bei  $x^r$  um einen ganz hervorragenden Punkt. Daher besteht die Hoffnung, dass in Richtung  $x^r$  noch mehr zu holen ist. Aus diesem Grund definiert man den Punkt  $x^e$ , der anschaulich einer Expansion des Simplex entspricht. Ist dann  $f(x^e) < f(x^b)$ , so ersetzt man  $x^s$  durch  $x^e$ , anderenfalls verzichtet man auf die Expansion und nimmt den Punkt  $x^r$  an Stelle von  $x^s$  in den Simplex auf.

Kritisch ist die Situation im Fall  $f(x^r) > f(x^z)$ . Der reflektierte Punkt  $x^r$  liefert also keine gute Reduktion des Funktionswertes. Daher liegt die Vermutung nahe, dass das aktuelle Polyeder zu groß ist, was eine Kontraktion erforderlich macht. Hierbei wird entweder eine *innere Kontraktion* durchgeführt in Richtung  $x^s$  (falls  $f(x^r) \geq f(x^s)$ ), der Punkt  $x^r$  also ganz schlecht ist) oder eine *äußere Kontraktion* in Richtung  $x^r$  (falls  $f(x^r) < f(x^s)$ ). Ist der auf diese Weise neu berechnete Punkt  $x^k$  ganz brauchbar im Sinne von  $f(x^k) < f(x^s)$ , so ersetzen wir den bisher schlechtesten Punkt  $x^s$  durch  $x^k$ . Anderenfalls fällt uns nichts besseres ein, als das bisherige Polyeder um den bislang besten Punkt  $x^b$  zusammenzuziehen.

Noch einige Bemerkungen zur Implementation: Zur Konstruktion eines Startpolyeders mit Ecke  $x^1$  wähle man sich neben dem Punkt  $x^1 \in \mathbb{R}^n$  ein  $c > 0$  und

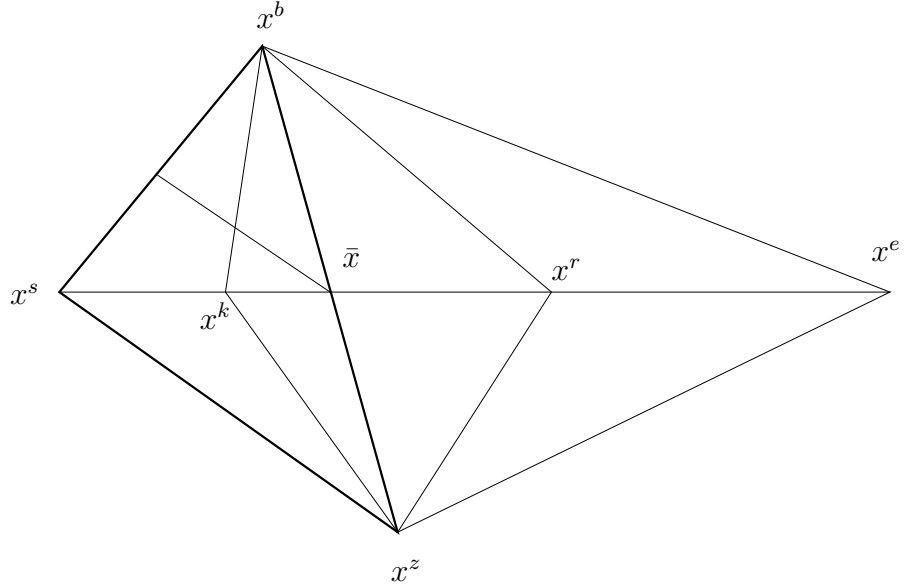


Abbildung 1: Veranschaulichung des Nelder–Mead–Verfahrens

setze

$$x^i := x^1 + p + (p_1 - p_2)e^{i-1} \quad \text{für } i = 2, \dots, n+1,$$

wobei  $e^i$  den  $i$ -ten Einheitsvektor im  $\mathbb{R}^n$  bezeichnet,  $p \in \mathbb{R}^n$  ein Vektor mit Komponenten  $p = (p_2, \dots, p_2)^T$  ist und die skalaren Größen  $p_1, p_2 \in \mathbb{R}$  definiert sich durch

$$p_1 := c \frac{\sqrt{n+1} + n - 1}{\sqrt{2n}}, \quad p_2 := c \frac{\sqrt{n+1} - 1}{\sqrt{2n}}.$$

Man verifiziert sehr leicht, dass die so konstruierten Punkte  $x^1, \dots, x^{n+1}$  dann ein regelmäßiges Polyeder mit  $\|x^i - x^j\|_2 = c$  für alle  $i \neq j$  bilden.

Als Abbruchkriterium kann man mittels einer vorgegebenen Toleranz  $\varepsilon > 0$  etwa testen, ob die Ungleichung

$$\sqrt{\sum_{i=1}^{n+1} \frac{(f(x^i) - f(\bar{x}))^2}{n}} < \varepsilon \quad (1)$$

erfüllt ist. Zusätzlich sollte auch die Zahl der Funktionsauswertungen nach oben beschränkt werden, um einen Notausstieg für das Verfahren zu haben, da die Konvergenz der Nelder–Mead–Methode nicht gewährleistet ist. Für Gegenbeispiele und entsprechende Hinweise sei etwa auf [?] verwiesen.

In der Praxis ist das Nelder–Mead–Verfahren trotzdem eines der beliebtesten Optimierungsverfahren überhaupt. Dies dürfte insbesondere daran liegen, dass es sich recht einfach implementieren lässt. Dennoch sollte man das Nelder–Mead–Verfahren nur für Probleme von sehr kleiner Dimension (etwa  $n \leq 10$ ) verwenden.

Als Beispiel illustrieren wir das Verhalten des Nelder–Mead–Verfahrens für die so genannte Crescent–Funktion

$$f(x) := \max \{x_1^2 + (x_2 - 1)^2 + x_2 - 1, -x_1^2 - (x_2 - 1)^2 + x_2 + 1\}$$

mit dem (Start–) Vektor  $x^1 := (-1.5, 2)^T$  sowie den Parametern

$$c = 1, \quad \varepsilon = 10^{-6}, \quad \alpha = 1, \quad \beta = 2, \quad \gamma = \frac{1}{2}.$$

Die zugehörigen numerischen Resultate sind in der Tabelle 1 enthalten (die letzte Spalte enthält dabei den Wert des Ausdrucks vom Abbruchkriterium (1)). Das Verfahren bricht zwar relativ schnell ab, die Genauigkeit ist allerdings nicht allzu hoch. So ist beispielsweise

$$x^b \approx (0.00094048275, -0.00000036076)^T$$

der am Ende der Iteration beste Näherungsvektor für die Lösung, die durch  $x^* := (0, 0)^T$  gegeben ist.

Für die als Testbeispiel sehr beliebte *Rosenbrock–Funktion*

$$f(x_1, x_2) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2 \tag{2}$$

mit Startvektor  $x^0 := (-1.2, 1)^T$  bricht das Nelder–Mead–Verfahren nach 71 Iterationen mit dem Näherungsvektor

$$(0.999955467, 0.999911049)^T$$

für das Minimum  $x^* := (1, 1)^T$  ab. Der Iterationsverlauf mit den verschiedenen Simplexen ist anschaulich in der Abbildung 2 angegeben.

$k$	$f(x^b)$	Abbruchkrit.
0	3.1286796564	—
1	3.1286796564	1.9395747542
2	0.8373750326	1.1902771324
3	0.8373750326	0.8610631278
4	0.3327381104	0.4582248468
5	0.3327381104	0.1694732721
6	0.1863483455	0.1885290796
7	0.1191843435	0.1679864578
8	0.0881941768	0.1092758947
9	0.0881941768	0.0776551049
10	0.0705837321	0.0502593319
11	0.0338425004	0.0244614717
12	0.0338425004	0.0174033442
13	0.0239779761	0.0121112016
14	0.0154701253	0.0174691231
15	0.0154701253	0.0073244071
16	0.0046681390	0.0057731916
17	0.0046681390	0.0053946483
18	0.0046681390	0.0032826795
19	0.0046681390	0.0019365801
20	0.0046681390	0.0026966119
21	0.0012391289	0.0020602342
22	0.0012391289	0.0015441629
23	0.0012391289	0.0014741671
24	0.0007379904	0.0010501970
25	0.0007379904	0.0006089000
26	0.0000326095	0.0004789462
27	0.0000326095	0.0003799246
28	0.0000326095	0.0002497455
29	0.0000326095	0.0002317109
30	0.0000326095	0.0001355594
31	0.0000326095	0.0000641765
32	0.0000326095	0.0000450861
33	0.0000170346	0.0000188569
34	0.0000170346	0.0000091766
35	0.0000170346	0.0000068166
36	0.0000096025	0.0000140048
37	0.0000077768	0.0000056551
38	0.0000044492	0.0000034788
39	0.0000012453	0.0000027582
40	0.0000012453	0.0000019562
41	0.0000012453	0.0000015244
42	0.0000012453	0.0000008678

Tabelle 1: Numerische Resultate zum Nelder–Mead–Verfahren

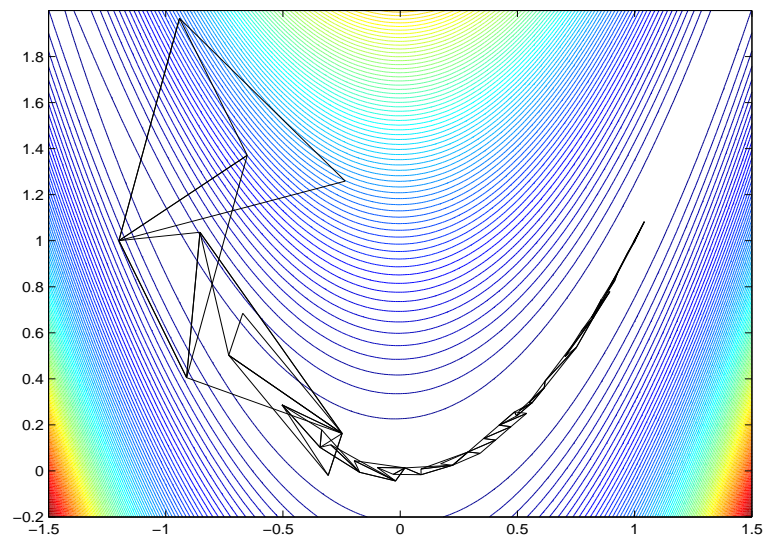


Abbildung 2: Iterationsverlauf des Nelder-Mead-Verfahrens bei der Rosenbrock-Funktion