Mehrdimensionale Optimierung ohne Nebenbedingungen

Sonja Biedermann, Felix Freynschlag, Bernhard Hayden,
Stepan Kharin, Christoph Pressler

10. Dezember 2015

Inhaltsverzeichnis

1 Das Nelder-Mead-Verfahren

1.1 Einführung

Das Nelder-Mead-Verfahren (auch Downhill-Simplex genannt) verfolgt einen geometrischen Ansatz zur iterativen Optimierung: ein Simplex bestehend aus $N + 1$ Punkten (im Falle $N = 2$, dem zweidimensionalen Raum, also ein Dreieck) wird fortwährend so transformiert, sodass sich der Simplex dem Minimum nähert, um welches er sich dann zusammenzieht. [1]

1.2 Funktionsweise

Der Algorithmus besteht im wesenstlichen aus drei Phasen [2], die in jeder Iteration durchlaufen werden:

1. Ordering
2. Centroid
3. Transformation

Ordering Zu Beginn besitzen wir $N + 1$ Punkte, die anhand irgendwelche Kriterien (durchaus auch zufällig) gewählt wurden. Diese müssen wir nun anhand ihrer Güte ordnen, also sortieren. Im Falle des zweidimensionalen Raumes erhalten wir also die drei mnemonisch benannten Punkte B (*best*), G (*good*) und W (*worst*). [3] B ist im Falle eine Minimierung der Punkt, dessen Funktionswert am geringsten ist, G der mit dem zweit geringsten Wert, und so weiter.

Centroid Als nächstes berechnen wir den Mittelpunkt der beiden besseren Punkte

$$M = \frac{B + G}{2}$$

und nehmen diesen als Basis für unsere weiteren Berechnungen.

Transformation Nun reflektieren wir den schlechtesten Punkt W am Mittelpunkt

$$R = M + \alpha(M - W)$$

und prüfen, ob dieser Punkt R die Relation $f(B) \leq f(R) \leq f(W)$ erfüllt.

Wenn ja, ersetzen wir den schlechtesten Punkt mit R und brechen die jetzige Iteration ab. Diesen Vorgang nennt man auch das „Akzeptieren“ eines Punktes.

Wenn nein, prüfen wir, ob der reflektierte Punkt R besser als unser bisher bester Punkt ist, also ob $f(R) < f(B)$ gilt. Wenn dem so ist, expandieren wir den Punkt R testweise weiter:

$$E = M + \gamma(M - W)$$

und prüfen wiederum, ob dieser Punkt E besser als unser reflektierte Punkt R ist. Wenn ja, akzeptieren wir E. Wenn nein, akzeptieren wir R.

Gilt $f(R) \geq f(G)$, so ist der reflektierte Punkt kein guter Kandidat. Wir müssen nun die zwei möglichen kontrahierten Punkte [2] C_1, C_2 berechnen. [3] Gilt $f(R) < f(W)$, berechnen wir den äußeren kontrahierten Punkt C_1, ansonsten den inneren kontrahierten Punkt C_2.

$$\begin{aligned} C_1 &= M + \rho(R - M) \\ C_2 &= M + \rho(W - M) \end{aligned}$$

Sind die berechneten kontrahierten Punkte besser als die zur Berechnung verwendeten Punkte (R und W respektive), so werden diese akzeptiert. Ist dem nicht so, müssen wir zum nächsten Schritt, der Komprimierung, fortfahren. Bei der Komprimierung werden jediglich die zwei schlechteren Punkte G und W in Richtung des besten Punktes B gezogen.

$$\begin{aligned} G &= B + \sigma(B - G) \\ W &= B + \sigma(B - W) \end{aligned}$$

Diese Schritte werden fortgeführt, bis eine Abbruchbedingung eintritt, wie etwa dass der Unterschied zwischen dem besten Funktionswert $f(B)$ und dem schlechtesten Funktionswert $f(W)$ oder der Umfang des Simplex' ein gegebenes ε unterschreitet.

1.3 Erläuterung der einzelnen Transformationsschritte

Folgend werden die einzelnen Transformationsschritte anhand eines Dreiecks als Simplex vorgeführt und erläutert. Für unsere Koeffizienten wählen wir die Werte $\alpha = 1$, $\gamma = 2$, $\rho = \frac{1}{2}$, $\sigma = \frac{1}{2}$. Unser Dreieck $\triangle BGW$ ist definiert mit $B = (1, 3)$, $G = (4, 6)$, $W = (6, 2)$.

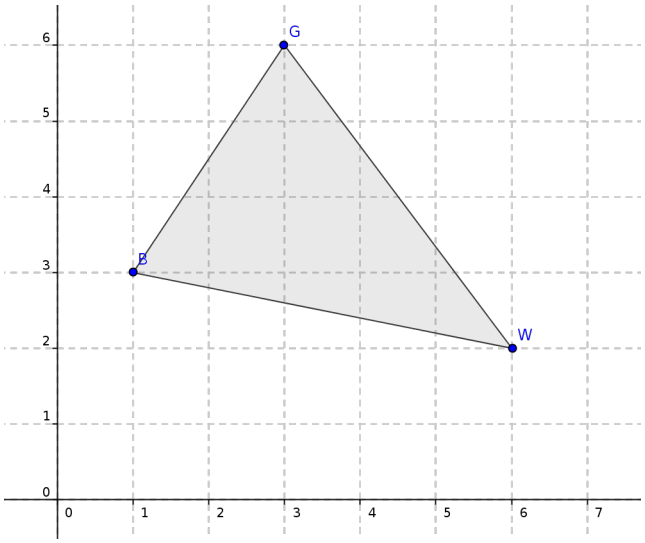

Abbildung 1: Unser Ursprungssimplex

1.3.1 Reflektion

Wir reflektieren den schlechtesten Punkt an dem Mittelpunkt in die vermeintlich bessere Richtung.

$$R = M + M - W = 2M - W = \begin{pmatrix} 5 \\ 9 \end{pmatrix} - \begin{pmatrix} 6 \\ 2 \end{pmatrix} = \begin{pmatrix} -1 \\ 7 \end{pmatrix}$$

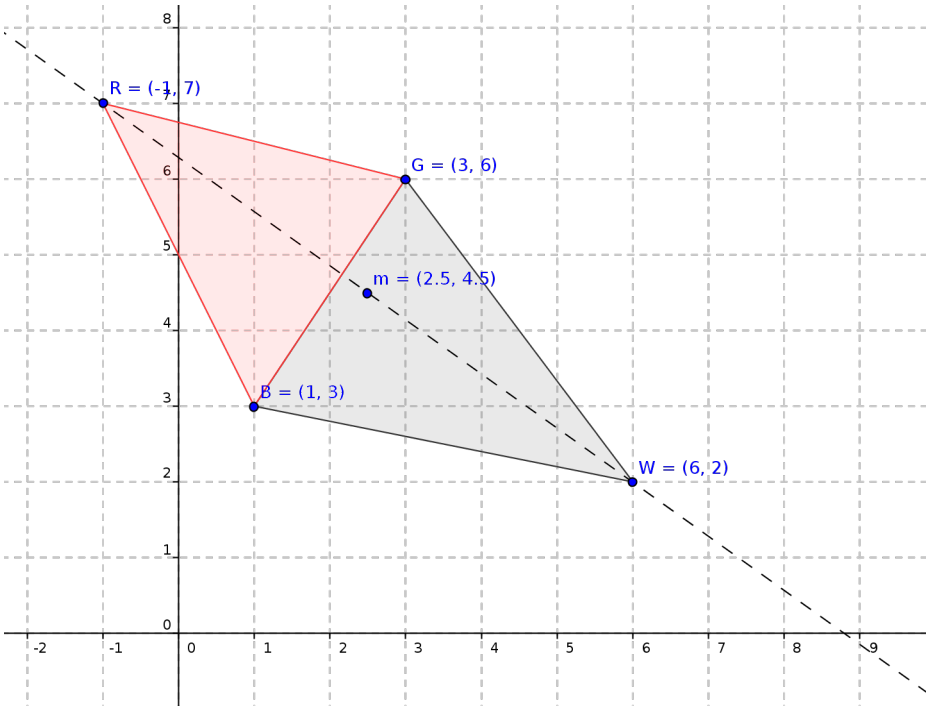

Abbildung 2: Nach der Reflektion

Dies entspricht dem „Heruntertaumeln“ des Simplex, welches die Optimierung in den ersten Iterationsschritten unter anderem antreibt.

1.3.2 Expansion

Nehmen wir an, dass $f(R)$ besser als unser bis jetzt bester Wert ist. Wir expandieren also weiter:

$$E = M + 2(M - W) = \begin{pmatrix} 2.5 \\ 4.5 \end{pmatrix} + \begin{pmatrix} -7 \\ 5 \end{pmatrix} = \begin{pmatrix} -4.5 \\ 9.5 \end{pmatrix}$$

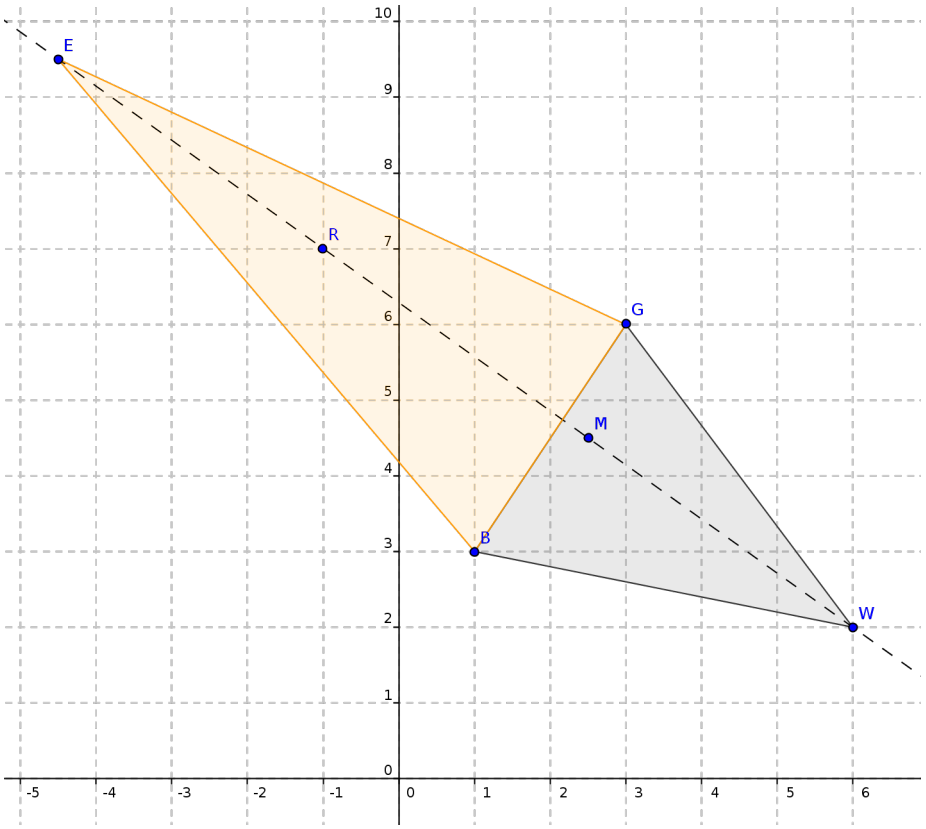

Abbildung 3: Nach der Expansion

Dies entspricht einem *greedy approach* – dieser Schritt ist nicht notwendig per se, minimiert aber zusätzlich die Anzahl der nötigen Iterationen. In schwer zu optimierenden Funktionen, wie etwa Rosenbrock's Bananenfunktion, kann der Expansionsfaktor γ dazu genutzt werden, den Simplex in flachem Gelände schneller wandern zu lassen.

1.3.3 Kontraktion

Gehen wir jetzt vom Gegenteil aus – Punkt R war also kein geeigneter Punkt und Punkt E wurde gar nicht berechnet – so müssen wir die beiden Kontraktionspunkte C_1, C_2 berechnen.

$$
\begin{aligned}
C_1 &= M + \frac{R - M}{2} = \begin{pmatrix} 2.5 \\ 4.5 \end{pmatrix} + \frac{\binom{-1}{7} - \binom{2.5}{4.5}}{2} = \begin{pmatrix} -1.75 \\ 1.25 \end{pmatrix} \\
C_2 &= M + \frac{W - M}{2} = \begin{pmatrix} 2.5 \\ 4.5 \end{pmatrix} + \frac{\binom{6}{2} - \binom{2.5}{4.5}}{2} = \begin{pmatrix} 4.25 \\ 1.25 \end{pmatrix}
\end{aligned}
$$

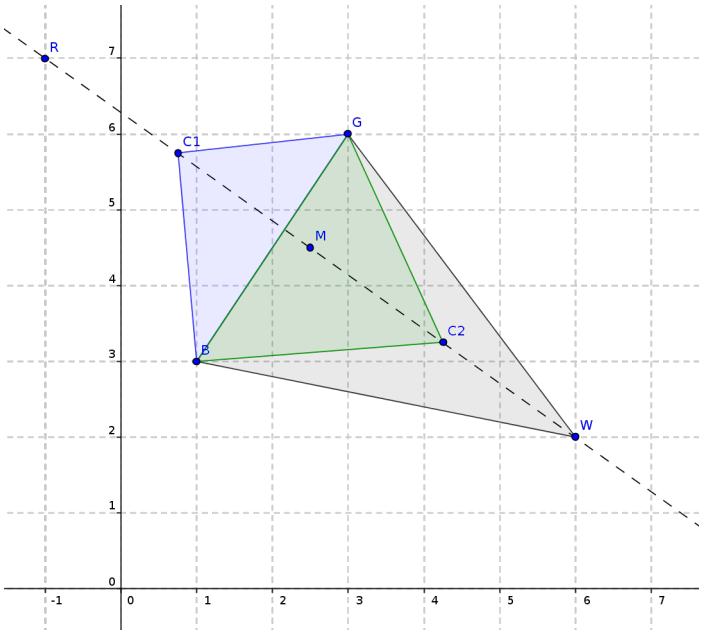

Abbildung 4: Nach der Kontraktion

1.3.4 Komprimierung

Wir komprimieren die Punkte G und W in Richtung B.

$$G' = B + \frac{G-B}{2} = \begin{pmatrix}1\\3\end{pmatrix} + \begin{pmatrix}1\\1.5\end{pmatrix} = \begin{pmatrix}2\\4.5\end{pmatrix}$$

$$W' = B + \frac{W-B}{2} = \begin{pmatrix}1\\3\end{pmatrix} + \begin{pmatrix}2.5\\-0.5\end{pmatrix} = \begin{pmatrix}3.5\\2.5\end{pmatrix}$$

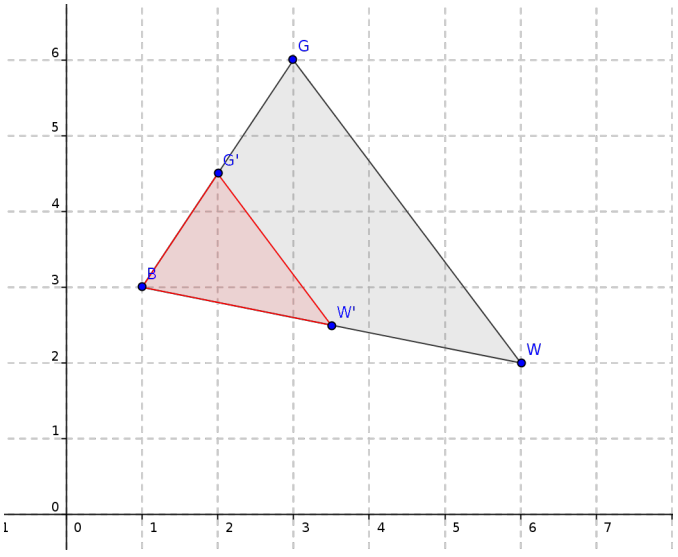

Abbildung 5: Nach der Komprimierung Richtung B

Dies entspricht dem Zusammenziehen des Simplex' in den letzten Iterationsschritten des Algorithmus'.

1.4 Ein Beispiel: die Himmelblau-Funktion

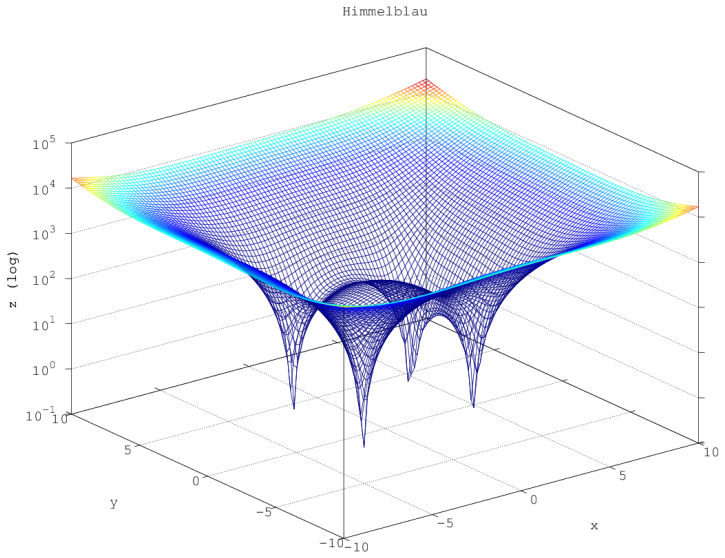

Abbildung 6: Die Himmelblau Funktion auf logarithmischer z-Skala

Um unseren Algorithmus exemplarisch vorzuführen, wollen wir die Himmeblau- Funktion minimieren. Dazu verwenden wir die beiliegende `C++`-Implementierung, um die hier aufgelisteten Graphiken zu generieren.

Die Himmelblau-Funktion ist definiert als

$$f(x, y) = (x^2 + y - 11)^2 + (x + y^2 - 7)^2$$

und hat 4 gleiche lokale Minima an den Stellen

$$\begin{aligned} f(3.2, 2.0) &= 0.0 \\ f(-2.805118, 3.131312) &= 0.0 \\ f(-3.779310, -3.283186) &= 0.0 \\ f(3.584428, -1.848126) &= 0.0 \end{aligned}$$

Wir wollen unseren Algorithmus auf sie loslassen, um eines davon zu finden. Als Abbruchbedingung wählen wir

$$\Delta f_{W,B} \leq 0.1,$$

also dass der Unterschied des besten und schlechtesten Funktionswert kleiner gleich 0.1 ist. Wir wählen diesen so groß, weil wir aus Platzgründen lieber schnell konvergieren als ein möglichst genaues Resultat wollen.

Unsere Startpunkte sind folgende:

$$\begin{aligned} P_1 &= (-1, -5) \\ P_2 &= (3, -8) \\ P_3 &= (8, 8) \end{aligned}$$

Wie bereits besprochen, sind die Startpunkte relativ egal und wurden auch hier recht zufällig gewählt. Die einzige Einschränkung des Startsimplex ist, dass es sich hierbei auch um einen tatsächlichen Simplex handeln muss (ein degenerierter Simplex wäre nutzlos). Weiters ist ein großer Simplex wünschenswert, da gegebenenfalls zu Beginn einer Optimierung große Schritte in Richtung Minimum gemacht werden müssen, um schnell konvergieren zu können. Ein kleiner Simplex benötigt weit mehr Zeit um zu konvergieren – dieses Problem kann man beispielsweise bei Rosenbrock's Banenenfunktion beobachten, die in der beiliegenden Implementierung ebenfalls vorhanden ist.

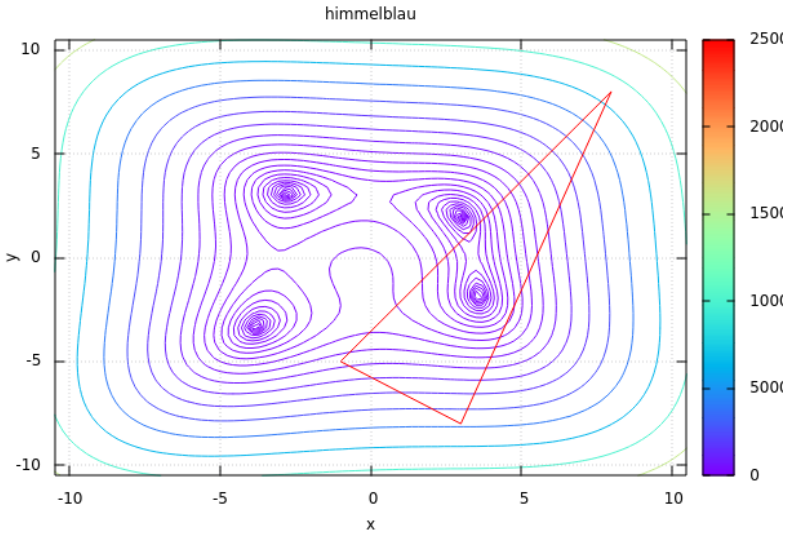

Abbildung 7: Unser Anfangssimplex aufgetragen auf einen Contourplot der Himmelblau-Funktion.

Der erste Schritt ist das Ordnen der Punkte nach ihrem Funktionswert. Wir berechnen:

$$\begin{aligned} f(P_1) &= 512 \\ f(P_2) &= 3700 \\ f(P_3) &= 7964 \end{aligned}$$

Netterweise sind unsere Punkte also schon geordnet und P_1, P_2 und P_3 entsprechen jeweils B, G, und W. Wir fahren fort und berechnen den Mittelpunkt

$$M = \frac{B+G}{2} = \begin{pmatrix} 1 \\ -6.5 \end{pmatrix}$$

Als nächstes berechnen wir den reflektierten Punkt

$$R = 2M - W = \begin{pmatrix} -6 \\ -21 \end{pmatrix}$$

Berechnen wir nun $f(R) = 183200$, sehen wir, dass der reflektierte Punkt keine gute Idee ist. Da R bei weitem schlechter als unser schlechteste Punkt W ist, berechnen wir den inneren kontrahierten Punkt

$$C_2 = M + \frac{W - M}{2} = \begin{pmatrix} 4.5 \\ 0.75 \end{pmatrix}$$

Berechnen wir nun $f(C_2) = 103.75391$, sehen wir, dass wir einen guten Fortschritt gemacht haben. Wir ersetzen W und ordnen unsere Punkte neu.

$$\begin{aligned} B &= (4.5,\ 0.75) \\ G &= (-1,\ -5) \\ W &= (3,\ -8) \end{aligned}$$

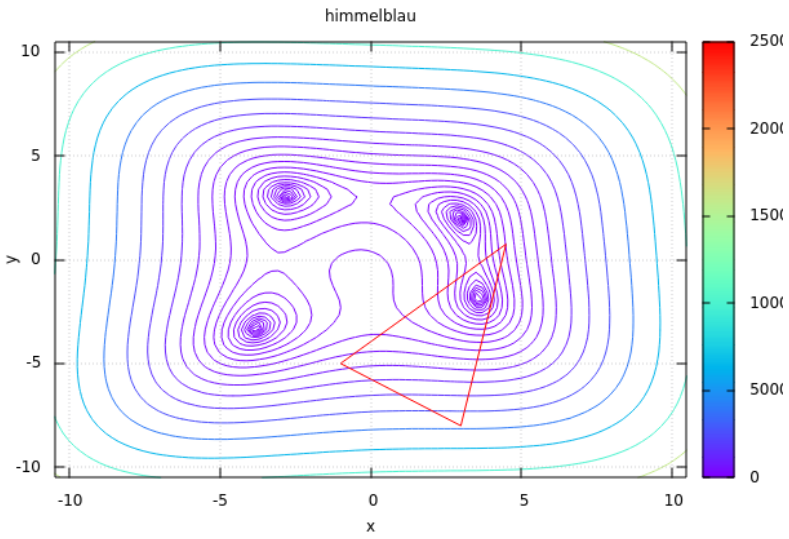

Abbildung 8: Unser Simplex nach dem ersten Transformationsschritt

Nun berechnen wir den neuen Mittelpunkt und reflektierten Punkt

$$M = \frac{B+G}{2} = \begin{pmatrix} 1.75 \\ -2.125 \end{pmatrix}$$
$$R = 2M - W = \begin{pmatrix} 0.5 \\ 3.75 \end{pmatrix}$$

Nun berechnen wir $f(R) = 106.19141$ und sehen, dass dies Relation $f(B) \leq f(R) \leq f(W)$ erfüllt ist. Wir akzeptieren also den reflektierten Punkt und ordnen unsere Werte neu.

$$\begin{aligned} B &= (4.5,\ 0.75) \\ G &= (0.5,\ 3.75) \\ W &= (-1,\ -5) \end{aligned}$$

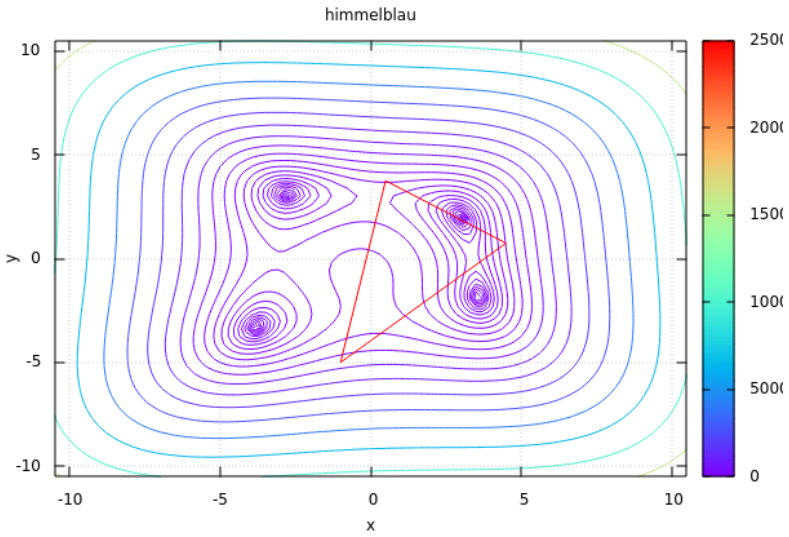

Abbildung 9: Unser Simplex nach dem zweiten Transformationsschritt

Wir berechnen den neuen Mittelpunkt $M = \binom{2.5}{2.25}$ und den neuen reflektierten Punkt $R = \binom{6}{9.5}$. $f(R) = 9155.815$, wir erreichen also keine Verbesserung. Wir berechnen wiederum den inneren kontrahierten Punkt $C_2 = \binom{0.75}{-1.375}$ und sehen, dass $f(C_2) = 158.53931$, also eine Verbesserung gegenüber $f(W) = 514$. Wir akzeptieren also C_2 und ordnen unsere Punkte neu.

$$\begin{aligned} B &= (4.5,\ 0.75) \\ G &= (0.5,\ 3.75) \\ W &= (0.75,\ -1.375) \end{aligned}$$

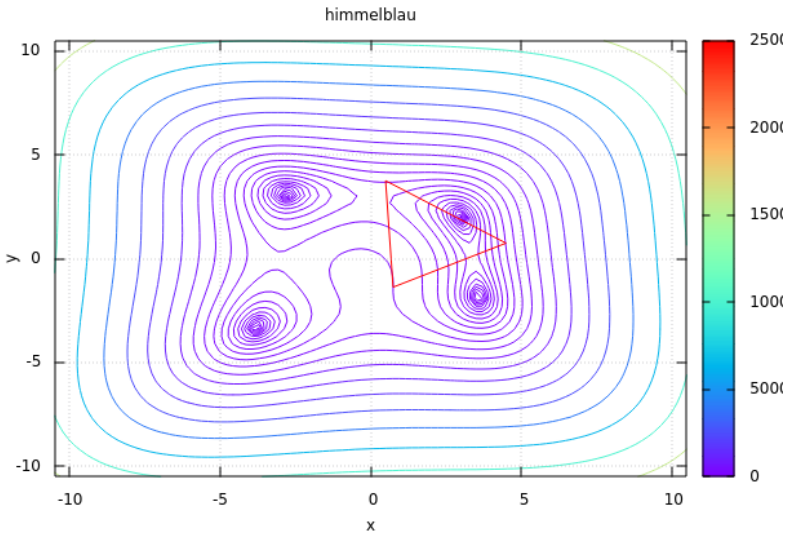

Abbildung 10: Nach dem dritten Transformationsschritt

Die beiden besten Punkte haben sich nicht geändert, also bleibt auch der Mittelpunkt gleich. Wir berechnen den reflektierten Punkt als $R = \binom{4.25}{5.875}$. Da $f(R) = 1176.43384$ haben wir keine Verbesserung erreicht und sind sogar schlechter als der schlechteste Punkt unseres Simplex', demnach berechnen wir wiederum den inneren kontrahierten Punkt $C_2 = \binom{1.625}{0.4375}$. Der Funktionswert an der Stelle C_2 ist 89.62574 und damit besser als B. Wir akzeptieren diesen Punkt daher und ordnen unsere Punkte neu.

$$\begin{aligned} B &= (1.625,\ 0.4375) \\ G &= (4.5,\ 0.75) \\ W &= (0.5,\ 3.75) \end{aligned}$$

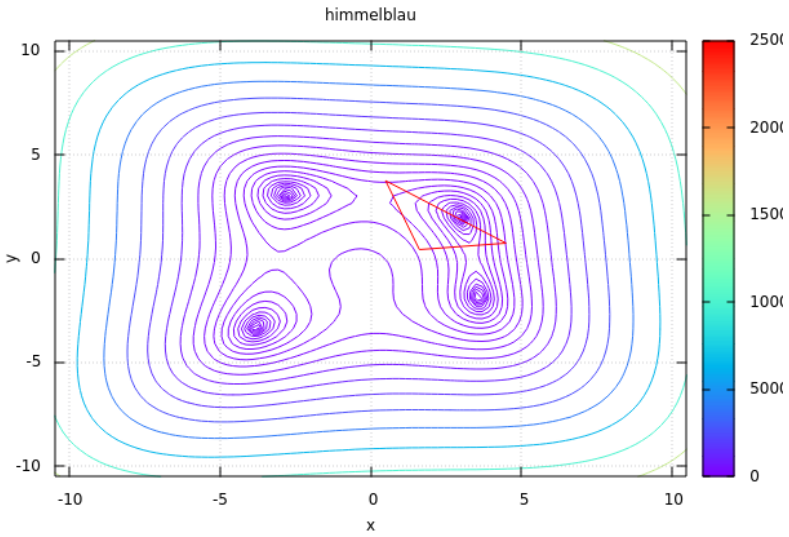

Abbildung 11: Nach dem vierten Transformationsschritt

Es ist mittlerweile klar, dass wir im Punkt $\binom{3}{2}$ konvergieren werden. Das beliegende `C++`-Programm konvergierte mit den angegebenen Parametern in 23 Iterationsschritten. Die weiteren Schritte werden hier mit dem Verweis auf die Implementierung und die `.gif`-Datei auf der Projektwebseite ausgelassen.

1.5 Analyse

Die formale mathematische Analyse des Nelder-Mead-Algorithmus' ist nicht möglich. Es kann nicht eindeutig bewiesen werden, dass der Algorithmus beim Bearbeiten einer gegebenen Funktion f_x konvergiert. Die Annahme, dass der Simplex konvergiert, beruht auf zwei Gegebenheiten:

Der Simplex degeneriert nicht während der Laufzeit.

Irgendeine Art eines Abstiegskriteriums wird verwendet.

Beides ist *nicht* gegeben – der Simplex kann und wird in einigen Fällen beinahme degenerieren, sofern die Landschaft der Funktion dies zulässt. [1] Weiters ist das Abstiegskriterion mehr zufällig – es ist nicht mathematisch fundiert wie etwa die Verwendung des negativen Gradienten, der mathematisch beweisbar immer in die Richtung des steilsten Abstiegs zeigt. [2]

Nichtsdestotrotz ist der Nelder-Mead-Algorithmus ein guter ableitungsfreier Optimierungsalgorithmus, der zumeist zuverlässig korrekte Resultate liefert. Tatsächlich gibt es aber eine ganze Familie von Funktionen, für die das Nelder-Mead-Verfahren immer zu einem Punkt konvergiert, der nicht das Minimum ist [4].

In vielen naturwissenschaftlichen Bereichen sind Funktionswerte fehlerbehaftet und damit genaue Lösungen unnötig. In diesen Fällen ist der Nelder-Mead-Algorithmus von großem Vorteil, da er schnell Ergebnisse liefert die oft bereits „gut genug" sind. Dazu kommt selbstverständlich die Tatsache, dass

[1] Man betrachte Rosenbrock's Bananenfunktion, in deren Tal der Simplex rapide schrumpft und sich somit nur mehr sehr langsam bewegen kann.

der vorliegende Algorithmus anschaulich und relativ leicht verständlich ist, und viele natürlich mehr inkliniert sind, Methoden zu verwenden, die sie auch verstehen.

2 Das Gradientenverfahren

2.1 Die Idee und der Zweck des Gradientenverfahren

Grundsätzlich ist das Gradientenverfahren ein Optimierungsverfahren, welches das Minimum einer mehrdimensionalen Funktion errechnen. Der Gradient ($\nabla f(x,y)$) einer Funktion $f(x)$ zeigt, aus später gezeigten [2] Gründen, immer in die Richtung des steilsten Anstiegs, folglich zeigt -$\nabla f(x,y)$ in die Richtung des stärksten Abstiegs. Das Gradientenverfahren macht sich dies zunutze und man "wandert" eine gewisse Dauer in diese Richtung, wie "lange" man in die errechnete Richtung "wandert" hängt unter anderem auch von der gewählten Methode ab.

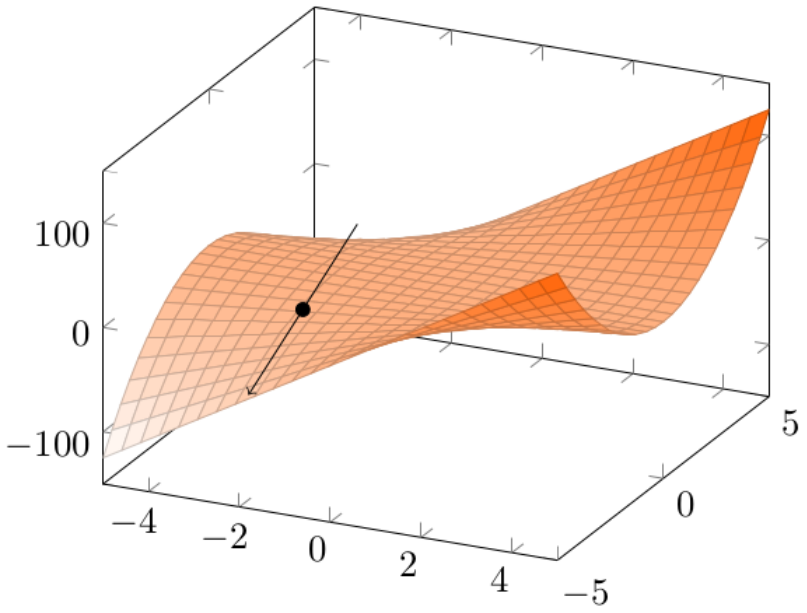

Abbildung 12: Hier zu sehen ist -$\vec{\nabla} f(-2,-2)$ der Funktion $f(x,y) = x * y^2$

2.2 Der Ablauf des Gradientenverfahrens

1. Es wird ein beliebiger Startpunkt $P = (x,y)$ ausgewählt.
2. Es wird die Ableitung der Funktion $f(x,y)$, also $\nabla f(x)$ gebildet. Sollte dieser Gradient bereits den Wert null haben sind wir bereits fertig und haben ein lokales Minimum.
3. Der Punkt P wird in den Gradienten eingesetzt und der negative Wert berechnet, also $-\nabla f(x_P, y_P)$.
4. Um eine neuen Punkt $P^{[n+1]}$ zu finden stellen wir nun folgende Gleichung auf: $P^{[n+1]} = P^{[n]} + \alpha_{[n]} * s^{[n]}$. Nun haben wir die Richtung in die optimiert werden muss gefunden ($s^{[n]}$ $(= -\nabla f(x_P, y_P))$), jetzt muss noch entschieden werden wie weit wir in diese Richtung "wandern"

[2] HIER VERWEIS EINFÜGEN

($\alpha^{[n]}$, liegt zwischen 0 und 1) (Hinweis: n ist die aktuelle Iteration).

Das kann man einerseits mit Hilfe eines fixen Werts machen, andererseits auch mit einer Folge (wie z.B $\frac{1}{\sqrt{n}}$) oder einer jedes mal neu errechneten Schrittweite. Zu beachten ist, dass die ausgewählte Folge auf jeden Fall divergent sein muss, da man sonst bis zu einer maximalen Entfernung "wandern" kann und die Wahrscheinlichkeit groß ist das Optimum nie zu erreichen. Sollte man jedes mal einen fixen Wert verliert man zwar den Aufwand für die Berechnung der Schrittweite, jedoch ist die Wahrscheinlichkeit hoch, dass man viele Iterationen benötigt oder in einer endlosen Schleife festhängt.

Es gibt verschieden Arten wie man die Schrittweite berechnen kann, im Folgenden werde ich die Schrittweitenbestimmung nach Armijo erklären. Dieses Verfahren ist eines der sogenannten "line-search" Verfahren, das von mir gezeigte ist jedoch kein exaktes line-search Verfahren sondern lediglich eine einfache Heuristik.

Die Armijo-Bedingung: $\varphi(\alpha) \leq \varphi(0) + c * \alpha^{[n]} * \varphi(0)'$.

Wobei c eine Konstante zwischen null und eins ist und dazu dient das die Bedingung nicht zu restriktiv ist, $\varphi(\alpha) = f(P^{[n]} + \alpha^{[n]} s^{[n]})$ und $\varphi(0)' = \nabla f(P^n) * s^{[n]}$

5. Für den Anfang wird $\alpha = 1$ gewählt und errechnet ob die Armijo-Bedingung erfüllt ist - sollte sie erfüllt sein haben wir ein passende α gefunden, wenn nicht wird α mit einem β, welches ebenfalls zwischen null und eins liegt, multipliziert und wieder geprüft ob die Ungleichung erfüllt ist. (Hinweis: In der Praxis liefern die Werte $c = 0.01$ und $\beta = 0.9$ häufig gute Ergebnisse. [3]
6. Wenn man einen neuen Punkt gefunden hat wird bei Punkt 3 fortgesetzt. Da es oft einen großen Rechenaufwand erfordert das genaue Minimum zu finden wird das Verfahren meist beendet wenn der Gradient einen akzeptablen Wert erreicht hat (So kann man als Abbruchbedingung z.B. die stärke des Anstiegs in einem Punkt (also den Betrag des Gradienten) nehmen).

2.3 Beispiele

2.3.1 Beispiel 1

Zu minimierende Funktion: $f(x, y) = y^4 + 2x^2 - 3xy + 1$, als Abbruchbedingung wählen wir $|\nabla f(x, y)| \leq 0.3$.

[3] https://www.unibw.de/lrt1/gerdts/lehre/lpnlp/lp-nlp.pdf

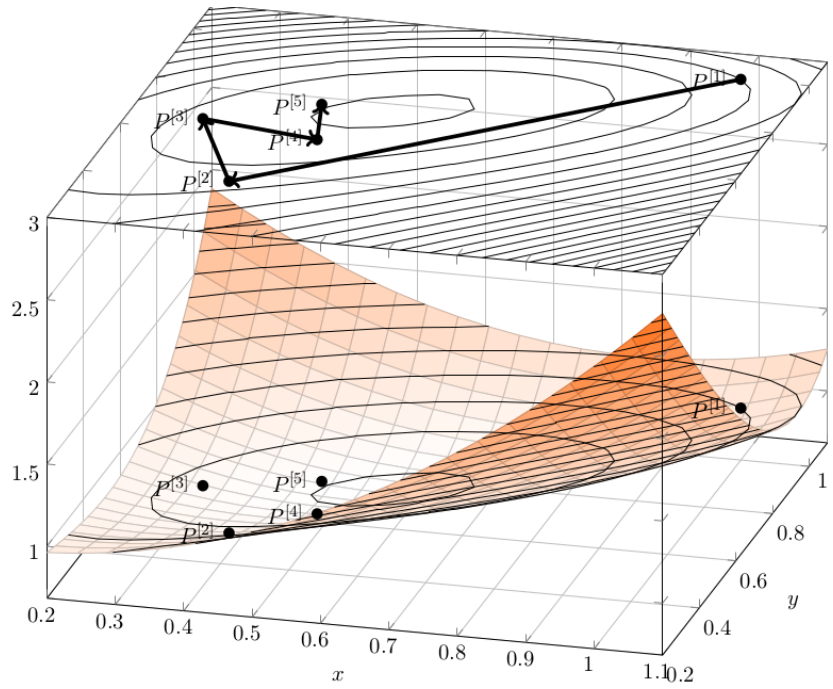

Abbildung 13: Beispiel 1. Zusätzlich sind hier die Höhenschichtlinien eingezeichnet.

Wie man in Abbildung 13 sieht, nähert man sich schrittweise dem Minimum an.

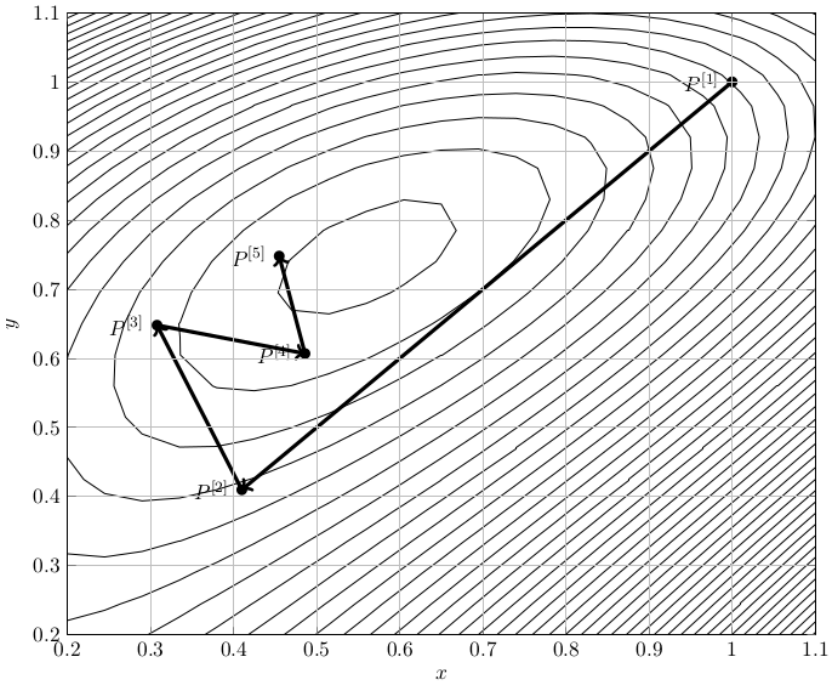

Abbildung 14: Die Höhenschichtlinien genauer gezeigt.

1. Startpunkt $P = (1, 1)$

2. Ableitung: $\nabla f(x,y) = \begin{pmatrix} 4x - 3y \\ 4y^3 - 3x \end{pmatrix}$ ($\rightarrow$ Der Gradient)

3. $-\nabla f(1,1) = \begin{pmatrix} -4+3 \\ -4+3 \end{pmatrix} = \begin{pmatrix} -1 \\ -1 \end{pmatrix}$ ($\rightarrow$ Richtung des stärksten Abstiegs)

4. Die Armijo-Bedingung (mit $\alpha = 1$, $c = 0.1$ und $\beta = 0.9$):
   $f(\begin{pmatrix} 1 \\ 1 \end{pmatrix} + 1 * \begin{pmatrix} -1 \\ -1 \end{pmatrix}) \leq f(\begin{pmatrix} 1 \\ 1 \end{pmatrix}) + 0.1 * 1 * \begin{pmatrix} 1 \\ 1 \end{pmatrix} * \begin{pmatrix} -1 \\ -1 \end{pmatrix}$
   $= 1 \leq 0.8$
   $\rightarrow \alpha^{[2]} = \alpha^{[1]} * \beta = 0.9$

   $f(\begin{pmatrix} 1 \\ 1 \end{pmatrix} + 0.9 * \begin{pmatrix} -1 \\ -1 \end{pmatrix}) \leq f(\begin{pmatrix} 1 \\ 1 \end{pmatrix}) + 0.1 * 0.9 * \begin{pmatrix} 1 \\ 1 \end{pmatrix} * \begin{pmatrix} -1 \\ -1 \end{pmatrix}$
   $= 0.99 \leq 0.82$
   $\rightarrow \alpha^{[3]} = \alpha^{[2]} * \beta = 0.81$

   $f(\begin{pmatrix} 1 \\ 1 \end{pmatrix} + 0.81 * \begin{pmatrix} -1 \\ -1 \end{pmatrix}) \leq f(\begin{pmatrix} 1 \\ 1 \end{pmatrix}) + 0.1 * 0.81 * \begin{pmatrix} 1 \\ 1 \end{pmatrix} * \begin{pmatrix} -1 \\ -1 \end{pmatrix}$
   $= 0.965 \leq 0.838$
   $\rightarrow \alpha^{[4]} = \alpha^{[3]} * \beta = 0.729$

   $f(\begin{pmatrix} 1 \\ 1 \end{pmatrix} + 0.729 * \begin{pmatrix} -1 \\ -1 \end{pmatrix}) \leq f(\begin{pmatrix} 1 \\ 1 \end{pmatrix}) + 0.1 * 0.729 * \begin{pmatrix} 1 \\ 1 \end{pmatrix} * \begin{pmatrix} -1 \\ -1 \end{pmatrix}$
   $= 0.932 \leq 0.854$
   $\rightarrow \alpha^{[5]} = \alpha^{[4]} * \beta = 0.656$

   $f(\begin{pmatrix} 1 \\ 1 \end{pmatrix} + 0.656 * \begin{pmatrix} -1 \\ -1 \end{pmatrix}) \leq f(\begin{pmatrix} 1 \\ 1 \end{pmatrix}) + 0.1 * 0.656 * \begin{pmatrix} 1 \\ 1 \end{pmatrix} * \begin{pmatrix} -1 \\ -1 \end{pmatrix}$
   $= 0.896 \leq 0.869$
   $\rightarrow \alpha^{[6]} = \alpha^{[5]} * \beta = 0,59$

   $f(\begin{pmatrix} 1 \\ 1 \end{pmatrix} + 0,59 * \begin{pmatrix} -1 \\ -1 \end{pmatrix}) \leq f(\begin{pmatrix} 1 \\ 1 \end{pmatrix}) + 0.1 * 0,59 * \begin{pmatrix} 1 \\ 1 \end{pmatrix} * \begin{pmatrix} -1 \\ -1 \end{pmatrix}$
   $= 0.86 \leq 0.882$
   Nun haben wir ein passendes α gefunden und können zum nächsten Punkt wandern.

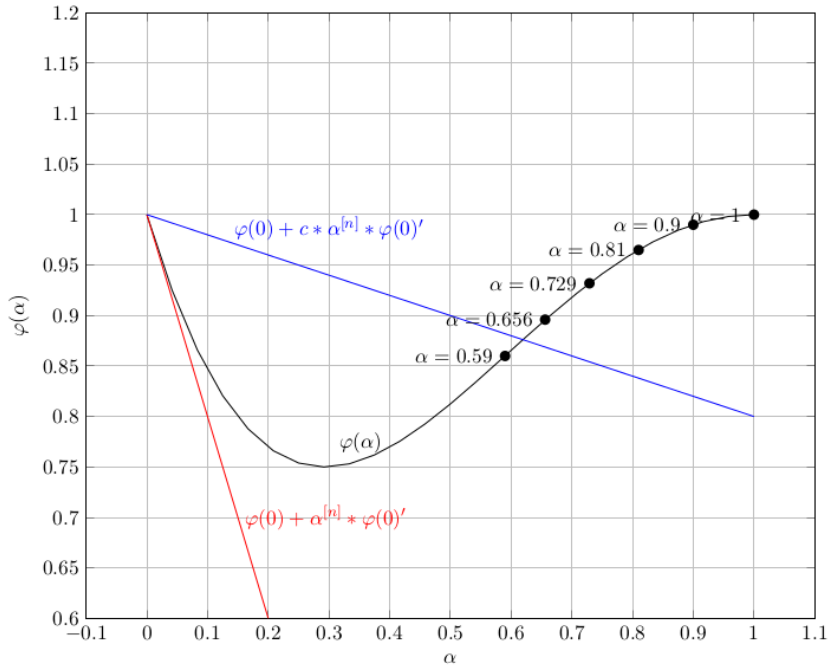

Abbildung 15: Hier sieht man die Funktion $\varphi(\alpha) = (1-\alpha)^4 + 2*(1-\alpha)^2 - 3*(1-\alpha)*(1-\alpha) + 1)$, sie ist sozusagen ein Durchschnitt durch die bei Beispiel 1 gegebene Funktion. Diese Funktion wird erzeugt, wenn man vom Punkt $P^{[1]} = (1,1)$ in Richtung des Gradienten wandert. Die blaue Gerade zeigt die Armijo-Bedingung bei $c = 0.1$, die rote bei $c = 1$ bzw. ohne c.

5. $P^{[2]} = \begin{pmatrix} 1 \\ 1 \end{pmatrix} + 0,59 * \begin{pmatrix} -1 \\ -1 \end{pmatrix} = \begin{pmatrix} 0.41 \\ 0.41 \end{pmatrix}$

6. $-\nabla f(0.41, 0.41) = \begin{pmatrix} -0.41 \\ 0.95 \end{pmatrix}$,
   $|\nabla f(0.41, 0.41)| = \sqrt{0.41^2 + 0.95^2} = 1.071$

7. Die Armijo-Bedingung (mit $\alpha = 1$, $c = 0.1$ und $\beta = 0.5$):
   $f(\begin{pmatrix} 0.41 \\ 0.41 \end{pmatrix} + 1 * \begin{pmatrix} -0.41 \\ 0.95 \end{pmatrix}) \leq f(\begin{pmatrix} 0.41 \\ 0.41 \end{pmatrix}) + 0.1 * 1 * \begin{pmatrix} -0.41 \\ 0.95 \end{pmatrix} * \begin{pmatrix} 0.41 \\ -0.95 \end{pmatrix}$
   $= 4.421 \leq 0.753 \rightarrow \alpha^{[2]} = \alpha^{[1]} * \beta = 0.5$
   $\rightarrow = 1.153 \leq 0.807 \rightarrow \alpha^{[3]} = \alpha^{[2]} * \beta = 0.25$
   $\rightarrow = 0.768 \leq 0.833$

8. $P^{[3]} = \begin{pmatrix} 0.41 \\ 0.41 \end{pmatrix} + 0.25 * \begin{pmatrix} -0.41 \\ 0.95 \end{pmatrix} = \begin{pmatrix} 0.308 \\ 0.648 \end{pmatrix}$

9. $-\nabla f(0.308, 0.648) = \begin{pmatrix} 0.712 \\ -0.164 \end{pmatrix}$,
   $|\nabla f(0.308, 0.648)| = \sqrt{0.712^2 + 0.164^2} = 0.534$

10. Die Armijo-Bedingung (mit $\alpha = 1$, $c = 0.1$ und $\beta = 0.5$):
    $f(\begin{pmatrix} 0.308 \\ 0.648 \end{pmatrix} + 1 * \begin{pmatrix} 0.712 \\ -0.164 \end{pmatrix}) \leq f(\begin{pmatrix} 0.308 \\ 0.648 \end{pmatrix}) + 0.1 * 1 * \begin{pmatrix} 0.712 \\ -0.164 \end{pmatrix} * \begin{pmatrix} -0.712 \\ 0.164 \end{pmatrix}$
    $= 1.655 \leq 0.714 \rightarrow \alpha^{[2]} = \alpha^{[1]} * \beta = 0.5$
    $\rightarrow = 0.857 \leq 0.741 \rightarrow \alpha^{[3]} = \alpha^{[2]} * \beta = 0.25$

$\rightarrow = 0.723 \leq 0.754$

11. $P^{[4]} = \begin{pmatrix} 0.308 \\ 0.648 \end{pmatrix} + 0.25 * \begin{pmatrix} 0.712 \\ -0.164 \end{pmatrix} = \begin{pmatrix} 0.486 \\ 0.607 \end{pmatrix}$

12. $-\nabla f(0.486, 0.607) = \begin{pmatrix} -0.123 \\ 0.563. \end{pmatrix}$,
$|\nabla f(0.486, 0.607)| = \sqrt{0.123^2 + 0.563^2} = 0.332$

13. Die Armijo-Bedingung (mit $\alpha = 1$, $c = 0.1$ und $\beta = 0.5$):
$f(\begin{pmatrix} 0.486 \\ 0.607 \end{pmatrix} + 1 * \begin{pmatrix} -0.123 \\ 0.563 \end{pmatrix}) \leq f(\begin{pmatrix} 0.486 \\ 0.607 \end{pmatrix}) + 0.1 * 1 * \begin{pmatrix} -0.123 \\ 0.563 \end{pmatrix} * \begin{pmatrix} 0.123 \\ -0.563 \end{pmatrix}$
$= 1.863 \leq 0.689 \rightarrow \alpha^{[2]} = \alpha^{[1]} * \beta = 0.5$
$\rightarrow = 0.852 \leq 0.707 \rightarrow \alpha^{[3]} = \alpha^{[2]} * \beta = 0.25$
$\rightarrow = 0.706 \leq 0.715$

14. $P^{[5]} = \begin{pmatrix} 0.486 \\ 0.607 \end{pmatrix} + 0.25 * \begin{pmatrix} -0.123 \\ 0.563 \end{pmatrix} = \begin{pmatrix} 0.455 \\ 0.748 \end{pmatrix}$

15. $-\nabla f(0.455, 0.748) = \begin{pmatrix} 0.424 \\ -0.309. \end{pmatrix}$,
$|\nabla f(0.455, 0.748)| = \sqrt{0.424^2 + 0.309^2} = 0.275$
Die Abbruchbedingung ist erfüllt und somit ist ein Punkt, welcher nahe genug am Optimum liegt gefunden.

3 Die Koordinatenabstiegsmethode

3.1 Einführung

Bei der Koordinatenabstiegsmethode wird bei jedem Schritt eine Koordinate i gewählt, nach der man optimiert. Es wird bei einem beliebigen Startpunkt begonnen und in jedem Iterationsschritt setzt man alle Koordinaten des aktuellen Punktes in die Funktion ein , nur die Koordinate , nach der man minimieren will , behält man als Variable bei.

Es wird immer in Richtung einer Achse minimiert und die Stelle , an der das Minimum ist , als neue Koordinate beibehalten und so gelangt man mit einer Anzahl an Iterationen zum Minimum der Funktion.

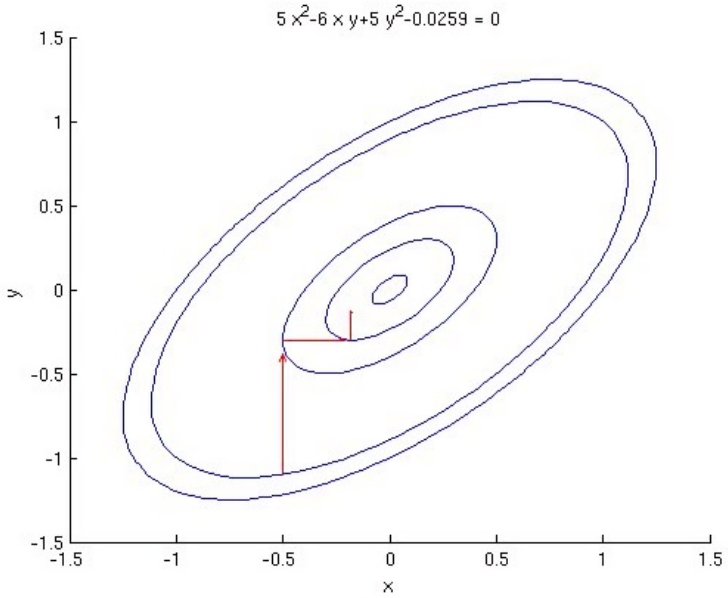

Abbildung 16: Die Koordinatenabstiegsmethode

Wenn zum Beispiel die Funktion $f(x, y) = 2x^2 + 2xy + 1.5y^2$ und der Startpunkt $P = (1, 1)$ gegeben ist, so muss man, wenn entlang der x-Achse minimiert werden soll folgendes eingesetzt werden:

$$\begin{aligned} x &= 1 + 1d \\ y &= 1 + 0d \end{aligned}$$

Soll entlang der y-Achse optimiert werden, so muss folgendes eingesetzt werden:

$$\begin{aligned} x &= 1 + 0d \\ y &= 1 + 1d \end{aligned}$$

3.2 Beispiel

Wir möchten die Funktion $f(x, y) = 2x^2+2xy+1.5y^2$ optimieren und wählen den Startpunkt $P_0 = (1, 1)$. Wir minimieren zuerst entlang der x-Achse und setzen ein

$$\begin{aligned} f(d) &= 2(1 + d)^2 + 2(1 + d) + 1.5 \\ &= 2d^2 + 6d + 5.5 \end{aligned}$$

Diese soeben erhaltene Funktion müssen wir nun noch minimieren.

$$\begin{aligned} f'(d) &= 4d + 6 \\ 4d + 6 &= 0 \\ d &= \frac{3}{2} \end{aligned}$$

Setzen wir dies in die Gleichung für die x-Koordinate ein und verwenden die alte y-Koordinate erhalten wir den neuen Punkt $P_1 = (-\frac{1}{2}, 1)$. Nun minimieren wir nach y, wir setzen also ein:

$$\begin{aligned} x &= -\frac{1}{2} + 0d \\ y &= 1 + 1d \end{aligned}$$

und erhalten

$$f(d) = 1.5d^2 - d + 6$$

Dies minimieren wir wiederum und erhalten die neue y-Koordinate $\frac{1}{3}$. Damit haben wir den neuen Punkt $(-\frac{1}{2}, \frac{1}{3})$. Dieser Vorgang wird wiederholt, bis sich die errechneten Werte nurnoch marginal unterscheiden. Danach wird abgebrochen.

3.3 Problemfälle

Zeigen Sie, dass die Koordinatenabstiegsmethode für die stetige Funktion

$$f(x, y) = \begin{cases} (x + y - 5)^2 + (x - y - 2)^2 & falls\ x \leq y \\ (x + y - 5)^2 + (x - y + 2)^2 & sonst \end{cases}$$

beginnend beim Startpunkt $(0, 0)$ nicht funktioniert.

Wir setzen ein:

$$\begin{aligned} x &= d \\ y &= 0 \end{aligned}$$

und erhalten die Funktion

$$\begin{aligned} f(d) &= (d - 5)^2 + (d - 2)^2 \\ &= 2d^2 - 14d + 29 \end{aligned}$$

welche ein Minimum an der Stelle $d = \frac{7}{2}$ hat. Dies fortgesetzt für y erhalten wir den neuen Punkt $P = (\frac{7}{2}, \frac{7}{2})$. Bei weiteren Iterationen merken wir, dass d immer 0 beträgt – was bedeutet, dass sich der Punkt nicht weiterbewegt. Wir sitzen auf einem lokalen Minimum fest. Führen wir das mit dem anderen Zweig der Funktion ebenfalls durch, so sehen wir das gleiche Problem.

Literatur

[1] Wikipedia. Nelder–mead method — wikipedia, the free encyclopedia, 2015. [Online; accessed 10-December-2015].

[2] S. Singer and J. Nelder. Nelder-Mead algorithm. *Scholarpedia*, 4(7):2928, 2009. revision 91557.

[3] Unknown. The nelder-mead method (proof), 2015. [Online excerpt of book; accessed 10-December-2015].

[4] K. I. M. Mckinnon and K. I. M. Mckinnon. Convergence of the nelder-mead simplex method to a non-stationary point. Technical report, SIAM J. Optim, 1996.