

18 Klassische Numerische Methoden der Modell-diskriminierung

Aufgrund der Wichtigkeit der Parameterschätzung in Bezug auf Modellbewertung wollen wir hier die Herleitung des Gauss-Newton-Verfahrens sowie Eigenschaften des Verfahrens sowie seine Implementierung in Presto-Kinetics skizzieren.

18.1 Newton-Verfahren und Eigenschaften

Unser grundsätzliches Problem ist es, für ein gegebenes Modell und gegebene Messdaten die Parameter des Modells so zu bestimmen, dass die Abweichung des Modells bzw. seiner Trajektorien von den Messdaten möglichst gering wird. Dieses Problem der Parameterschätzung kann zuerst auf die Fragestellung eines Minimierungsproblems und dann auf die Frage der Nullstellenbestimmung eines geeignet definierten Problems zurückgeführt werden. Für diese Fragestellung gibt es ein adäquates Verfahren, das Newton-Verfahren. Die Herleitung, Konvergenzeigenschaften und Umsetzbarkeit dieses Verfahrens werden dargestellt; wir halten uns dabei eng an die Darstellung aus [7]. Die Abwandlung zum Gauss-Newton-Verfahren, welches wir dann in der Praxis benutzen werden, folgt im zweiten Teil.

Da wir uns mit echten Modellen aus dem chemisch-technischen Bereich beschäftigen, werden wir die gesamten theoretischen Problem- und Verfahrensbeschreibungen auf den n -dimensionalen realen Raum \mathbb{R}^n beschränken.

18.1.1 Newton-Verfahren

Folgendes Problem, welches sich nicht direkt algebraisch lösen läßt, sei gegeben: Finde eine Lösung der Gleichung

$$F(x) = 0, F : \mathbb{R}^n \rightarrow \mathbb{R}^n \quad (84)$$

mit F stetig differenzierbar. Es liege ein Startpunkt $x^0 \in \mathbb{R}^n$ vor.

Ansatz Die Taylor-Entwicklung von F um den Startpunkt x^0 ergibt:

$$0 = F(x) = \underbrace{F(x^0) + F'(x^0)(x - x^0)}_{:=\bar{F}(x)} + o(|x - x^0|) \text{ für } x \rightarrow x^0 \quad (85)$$

Die Nullstelle x^1 der linearen Ersatzabbildung \bar{F} , die durch Vernachlässigen aller Terme mit höherer Ordnung als 1 erhalten wird, ist gerade

$$x^1 = x^0 - F'(x^0)^{-1}F(x^0)$$

falls die Jacobimatrix $F'(x^0)$ an der Stelle x^0 invertierbar ist, und lässt sich leicht anhand vorliegender Werte berechnen. Liegt $F(x^1)$ dichter an 0 als $F(x^0)$, so ist man

der Lösung des Problems nähergekommen. Besitzt die Abbildung F diese (kontrahierende) Eigenschaft, können wir hoffen, dass sukzessives Anwenden dieses Linearisierungsverfahrens eine Folge von x -Werten liefert, die gegen die Lösung des Problems konvergiert.

Eine Regel der Numerischen Analysis besagt, dass das Invertieren von Matrizen möglichst vermieden werden soll. Daher wird die obige Nullstellenbestimmung durch Lösen des äquivalenten Gleichungssystems $F'(x^0)(x^1 - x^0) = -F(x^0)$ erledigt. Aus diesem Grund schreiben wir auch das resultierende **allgemeine Newton-Verfahren** in der folgenden Form:

$$F'(x^k)\Delta x_k = -F(x^k), \quad x^{k+1} = x^k + \Delta x_k, \quad k = 0, 1, \dots \quad (\text{allg. Newton-Verfahren})$$

Offensichtlich ist es das Ziel, die Lösung des allgemeinen nichtlinearen Problems (84) durch die Lösungen einer Folge von linearen Problemen zu bestimmen.

Dieses Verfahren wurde konstruiert, um die Nullstelle der Funktion F zu finden. Die herausstechenden Fragen sind nun:

1. welche Voraussetzungen müssen an den Startpunkt bzw. die Iterationspunkte des Verfahrens und auch an die Funktion gestellt werden, damit
 - (a) das Verfahren überhaupt angewendet werden kann, und
 - (b) das Verfahren tatsächlich gegen eine/die Nullstelle konvergiert, und
2. mit welcher Geschwindigkeit konvergiert es?
3. Kann man Aussagen zu 1 und 2 unabhängig von der zu untersuchenden Funktion F machen?

Ein weiterer wesentlicher Punkt ist die Einführungen von Skalierungen, die in Anwendungsbeispielen immer eine Rolle spielen werden, und zwar sowohl Skalierungen im Bildraum der Abbildung F als auch Skalierungen im Urbildraum. Beide Arten von Skalierungen sollten nicht dazu führen, dass die Lösung eines skalierten Problems von der Lösung des nichtskalierten Problems abweicht.

18.1.2 Affine Transformation

In praktischen Beispielen ist es oft so, dass sowohl die Messdaten als auch die Parameter in sehr unterschiedlichen Größenordnungen bzw. Einheiten vorliegen und eine Skalierung nötig ist. Um einen allgemeingültigen Algorithmus aufzustellen ist es daher notwendig, dass zum einen das Verfahren selber unabhängig von Transformationen operiert, zum anderen aber auch die Beweistheorie von Transformationen solcher Art unabhängig gilt.

Seien $\mathbf{A}, \mathbf{B} \in \mathbb{R}^n$ nichtsinguläre Abbildungen. Das System

$$G(y) = \mathbf{A}F(\mathbf{B}y) = 0, \quad x = \mathbf{B}y$$

ist eine affin invariante Transformation des ursprünglichen Systems, denn

$$\begin{aligned} F(x) &= 0 \iff AF(x) = 0 \\ F(x) &= 0 \iff F(By) = 0 \text{ nur für } x = By \end{aligned}$$

Diese wird dem Newton-Verfahren unterworfen:

$$G(y^k)\Delta y^k = -G(y^k), y^k + 1 = y^k + \Delta y^k$$

Mit der Beziehung

$$G(y^k) = AF'(x^k)B$$

und

$$y^0 = B^{-1}x^0$$

folgt

$$\begin{aligned} y^1 \text{ ergibt sich aus der Lösung von } G(y^0)\Delta y^0 &= -G(y^0) \\ AF'(x^0)B\Delta y^0 &= -AF(By^0) \\ F'(x^0)B\Delta y^0 &= -F(By^0), \text{ aber auch} \\ F(x^0)\Delta x^0 &= -F(x^0), \text{ damit folgt sofort} \\ By^1 &= x^1 \end{aligned}$$

und man erhält für alle weiteren k in der Folge die Beziehung

$$By^k = x^k, k = 0, 1, \dots$$

Sowohl die Problemstellung als auch das Verfahren sind damit affin-invariant. Die Eigenschaft, dass man eine beliebige (invertierbare) Abbildung A auf F anwenden kann, ohne dass sich die Folge der Iterationspunkte ändert (setze $B=I$), nennt man affine Kovarianz, die Eigenschaft, dass die Iterierten sich bei Anwendung einer Transformation auf den Urbildraum im gleichen Sinne als Transformation der Ur-Iterierten darstellen, nennt man affine Kontravarianz. Eine Theorie zu den Konvergenzeigenschaften, die auch algorithmisch umsetzbar ist, sollte diese Eigenschaften berücksichtigen. In der Tat gibt zwei Richtungen: eine Theorie, die die affine Kovarianz aufgreift und Kriterien entwickelt, die auf den Iterierten x^k bzw. den Fehlern $\|x^k - x^*\|$ aufsetzt, und eine Theorie, die die affine Kontravarianz aufgreift. Bei dieser werden Kriterien entwickelt, die auf den Residuen $F(x^k)$ aufbauen.

18.1.3 Konvergenztheorie des Newton-Verfahrens

Damit das Verfahren

$$F'(x^k)\Delta x_k = -F(x^k), \quad x^{k+1} = x^k + \Delta x_k, k = 0, 1, \dots \quad (\text{allg. Newton-Verfahren})$$

überhaupt angewendet werden kann, muss als notwendige Bedingung gelten: $F'(x^k)^{-1}$ existiert in jedem x^k und $F'(x^k)^{-1}$ ist beschränkt.

Die Beschränktheit kann man sowohl global für $x \in D$ formulieren

$$\|F'(x)^{-1}\| \leq \beta < \infty, x \in D \quad (88)$$

als auch lokal

$$\|F'(x^0)^{-1}\| \leq \beta_0 \quad (89)$$

fordern.

Um Aussagen über die Konvergenz zu machen, muss man zusätzlich gewisse Kontraktionseigenschaften der Funktion F unter diesem Verfahren voraussetzen/nachweisen, was man i.A. über eine Lipschitzbedingung der Form

$$\|F'(x) - F'(\bar{x})\| \leq \gamma \|x - \bar{x}\|, x, \bar{x} \in D \subset \mathbb{R}^n \quad (90)$$

macht.

Ist diese kontrahierende Eigenschaft gegeben, so lässt sich zeigen (Banachsches Störungslemma), dass $\|F'(x)^{-1}\| < \beta \leq \frac{\beta_0}{1 - \beta_0 \gamma \|x - x^0\|}$ für $\|x - x^0\| < \frac{1}{\beta_0 \gamma}$.

Die klassischen Konvergenzsätze Newton-Kantorovitch bzw. Newton-Mysovskikh besagen nun im Wesentlichen: Setzt man die Lipschitzbedingung sowie die Beschränktheit von $F'(x)^{-1}$ voraus, und ist der Startpunkt x^0 so gewählt, dass der nächste Schritt des Verfahrens der Bedingung

$$h_0 := \|x^1 - x^0\| \beta_0 \gamma < \frac{1}{2} \quad (\text{Newton} - \text{Kantorovitch}) \quad (91)$$

bzw.

$$h_0 := \|x^1 - x^0\| \beta \gamma < 2 \quad (\text{Newton} - \text{Mysovskikh}) \quad (92)$$

genügt, so konvergiert das Newton-Verfahren quadratisch gegen die eindeutige Nullstelle der Funktion F und alle Iterationspunkte verbleiben innerhalb der Konvergenzkugel um x^0 mit dem Radius $\rho_0 \sim 1/\beta_0 \gamma$ bzw. $\rho \sim 1/\beta \gamma$

Für die praktische Implementierung eines Algorithmus sind diese theoretischen Größen nicht zugänglich und, selbst wenn sie zugänglich wären, nicht hilfreich, weil sie nicht invariant unter Transformationen sind. Daher werden diese Konvergenzsätze nicht ausführlicher dargestellt; statt dessen sollen sie auf unsere Anforderung zugeschnitten formuliert und bewiesen werden. Wir beschränken uns auf die kontravariante Beschreibung, da die Betrachtung der Residuen weit natürlicher erscheint.

18.1.4 Affin kontravariante Konvergenztheorie des Newton-Verfahrens

Betrachtet man die Klasse von Problemen

$$G(y) = F(By), x = By, B \in GL(n)$$

mit $GL(n)$ nicht-singuläre Matrizen, so kann man die klassischen Bedingungen (Beschränktheit und Lipschitzbedingung) folgendermassen formulieren:

$$\|(F'(\bar{x}) - F'(x))(\bar{x} - x)\| < \omega \|F'(x)(\bar{x} - x)\|^2$$

Beide Seiten sind unabhängig von B. Dieser Ansatz wurde von A.Hohmann in seiner Dissertation entwickelt und kann für residuenbasierte inexakte Newton-Verfahren benutzt werden.

Mit dieser Formulierung können wir nun den Konvergenzsatz (siehe [7], Theorem 2.12) angeben:

Satz 31 Sei $F : D \rightarrow \mathbb{R}^n$ eine differenzierbare Abbildung, $D \in \mathbb{R}^n$ offen und konvex. Sei $F'(x)$ invertierbar für alle $x \in D$. Die folgende affin kontravariante Lipschitzbedingung gelte:

$$\|(F'(y) - F'(x))(y - x)\| \leq \omega \|F'(x)(y - x)\|^2 \text{ für alle } x, y \in D \quad (93)$$

Definiere $L_\omega = \{x \in D \mid \|F(x)\| < \frac{2}{\omega}\}$; sei $\overline{L_\omega} \subset D$ beschränkt. Für einen gegebenen Vorschlag x^0 einer (unbekannten) Nullstelle x^* von F sei

$$h_0 := \omega \|F(x^0)\| < 2, \text{ d.h. } x^0 \in L_\omega. \quad (94)$$

Dann gilt für die Folge der Newton-Iterierten $\{x^k\}$, dass jedes Folgenglied in L_ω liegt und dass die Folge gegen eine Lösung $x^* \in \overline{L_\omega}$ mit $F(x^*) = 0$ konvergiert. Die iterierten Residuen $\{F(x^k)\}$ konvergieren gegen Null mit einer geschätzten Rate

$$\|F(x^{k+1})\| \leq \frac{1}{2} \omega \|F(x^k)\|^2 \quad (95)$$

Für den Beweis verweisen wir auf das Buch [7], Seite 77. Dort wird gezeigt, dass der gesamte Weg zwischen x^k und x^{k+1} aus L_ω ist. Hier wollen wir nur die anwendungstechnischen wichtigen Teile aufführen.

Für die Folge der residuenorientierten Größen

$$h_k := \omega \|F(x^k)\| \quad (96)$$

lässt sich die quadratische Rekursion herleiten

$$h_{k+1} \leq \frac{1}{2} h_k^2 = \left(\frac{1}{2} h_k\right) h_k \quad (97)$$

Die Voraussetzung (94) ($h_0 < 2$) liefert dann $h_{k+1} < h_k < 2, k = 0, 1, \dots$ und damit folgt sofort $\lim_{k \rightarrow \infty} h_k = 0$ und damit auch $\lim_{k \rightarrow \infty} \|F(x^k)\| = 0$.

18.1.5 Abbruchkriterium

Da der Konvergenzsatz auf einer affin kontravarianten Lipschitzbedingung aufbaut, kann man ein Abbruchkriterium einführen in der Form

$$\text{Ende, wenn } \|F(x^k)\| \leq TOL \quad (98)$$

wobei TOL ein vom Benutzer vorgegebener Residuentoleranzwert ist.

18.1.6 Konvergenzkriterium

Bei der Anwendung eines solchen iterativen Verfahrens in der Praxis ist es meist nicht möglich, die Voraussetzungen für eine Konvergenz nachzuweisen, z.B. das ω eines gegebenen Problems anzugeben. Um möglichst schnell eine Antwort darauf zu bekommen, ob das Verfahren konvergiert, wird das Residuum der Funktion (also die Abweichung des ermittelten Funktionswertes von dem gesuchten Funktionswert 0) untersucht in der Erwartung, dass dieses im Verlaufe des Verfahrens monoton kleiner wird.¹

Der einfache Monotonietest

$$\|F(x^{k+1})\| \leq \bar{\theta} \|F(x^k)\| \quad \text{für } k = 0, 1, \dots \quad \text{für ein } \bar{\theta} < 1 \quad (99)$$

ist sehr suggestiv formuliert, hat aber den Nachteil, dass er in dieser Form nicht affin-invariant ist, d.h. eventuelle Abbruchkriterien (Ende, wenn $\bar{\theta} < tol$) können nicht eindeutig formuliert werden.

Der *natürliche Monotonietest* dagegen

$$\|F'(x^k)^{-1}F(x^{k+1})\| \leq \bar{\theta} \underbrace{\|F'(x^k)^{-1}F(x^k)\|}_{\|\Delta x^k\|} \quad \text{für ein } \bar{\theta} < 1 \quad (100)$$

ist affin-invariant. Er unterscheidet sich vom einfachen Monotonietest zusätzlich darin, dass der erstere die Auswertung von F an zwei aufeinanderfolgenden Punkten der Iterationsfolge untersucht, während der letztere die Länge des vorangegangenen Schrittes zum Vergleich heranzieht: Wenn der einfache Monotonietest erfüllt ist, bedeutet er zugleich, dass der nächstfolgende Iterationsschritt kürzer sein wird als der vorangegangene.

In dieser Form ist der natürliche Monotonietest ohne weiteren Aufwand auszuwerten. Innerhalb der Iterationsfolge ist bereits Δx^k bekannt, $F(x^{k+1})$ muss für den folgenden Schritt sowieso berechnet werden, so dass höchstens bei Erfüllung eines Abbruchkriteriums diese letzte Auswertung von $F(x^{k+1})$ ausschließlich für den Monotonietest gemacht wird. Bleibt die Bestimmung von $F'(x^k)^{-1}$. Diese wird wiederum nicht explizit durchgeführt. Führt man ein \hat{x} ein:

$$\begin{aligned} \hat{x} & : &= -F'(x^k)^{-1}F(x^{k+1}) \\ F'(x^k)\hat{x} & = &-F(x^{k+1}) \end{aligned}$$

so kann das Gleichungssystem nach x gelöst werden und stellt damit die linke Seite aus (100) zur Verfügung. Beschreibt man \hat{x} als ein $\hat{\Delta}x^{k+1}$ (es ist noch nicht der nächste Schritt aus der Iterationsfolge!) so kann man (100) schreiben als

$$\|\hat{\Delta}x^{k+1}\| \leq \bar{\theta} \|\Delta x^k\|$$

¹ dies impliziert, dass nur lokale Minima gefunden werden können

Numerische Erfahrung hat gezeigt, dass ein Wert von $\bar{\theta} := 0.5$ für das Newton-Verfahren verlangt werden kann. Das bedeutet, dass das Newton-Verfahren bei einer Verletzung der Bedingung $\|\widehat{\Delta x}^{k+1}\| \leq \frac{1}{2} \|\Delta x^k\|$ abgebrochen werden sollte, weil Konvergenz nicht mehr zu erwarten ist. Als Konsequenz sollte ein möglichst besserer Startwert gewählt werden.

18.1.7 Gauss-Newton-Verfahren

Nachdem im vorangegangenen Kapitel die Lösung einer Nullstellenbestimmung diskutiert wurde, wird hier nun beschrieben, in welcher Form das Problem der Parameterschätzung auf ein Nullstellenproblem zurückgeführt werden kann. Gegeben sei ein Modell mit Parametern $p \in \mathbb{R}^n$, welches den Verlauf seiner Zustandsgröße $x \in \mathbb{R}^m$ durch m Trajektorien $\Phi_i(t, p)$ beschreibt, sowie Messdaten an bestimmten Messpunkten $(t_i, y_i), i = 1, l$. Das Ziel, Modell und Messdaten müssten „möglichst gut zueinander passen“, lässt sich mit dieser Modelldarstellung mit Hilfe der kleinsten Quadrate folgendermassen formulieren (Carl Friedrich Gauss):

$$\sum_{i=1}^l \left(\frac{y_i - \Phi_i(t_i, p)}{\delta y_i} \right)^2 = \min \quad (101)$$

wobei δy_i die Fehlertoleranz der Messdaten sei. Dieses Problem lässt sich mithilfe der Bezeichnung

$$F(p) = (f_i(p)) = \left(\frac{y_i - \Phi_i(t_i, p)}{\delta y_i} \right), i = 1, ..l \quad (102)$$

umformulieren in

$$\|F(p)\| = \min \quad (103)$$

Taylor-Entwicklung liefert

$$\|F(p^*)\| = \|F(p^0) + F'(p^0)(p^* - p^0) + .quadr.Terme...\| \quad (104)$$

Vernachlässigt man quadratische und höhere Terme (entsprechend der Entwicklung des Newton-Verfahrens), so erhält man die Iterationsvorschrift

$$\|F'(p^k)\Delta p^k + F(p^k)\| = \min, \quad p^{k+1} = p^k + \Delta p^k, \quad (105)$$

so dass man (auch entsprechend der Entwicklung des Newton-Verfahrens) die Lösung des ursprünglichen Problems durch die Lösungen einer Folge von linearisierten Problemen anzunähern versucht. Diese Lösungen kann man direkt ablesen zu

$$F'^T(p^k)F'(p^k)\Delta p^k = -F'^T(p^k)F(p^k) \quad (106)$$

18.2 Implementierter Algorithmus

Hier stellen wir die tatsächliche Implementierung des Gauss-Newton-Verfahrens im Programmpaket Presto-Kinetics vor.

18.2.1 Initialisierungen

int $k_{\max} = 200$, double $h_{\min} = 0.01$, double $first = 0.1$, double $f_{first} = 1$, double
 $damp = 1$, double $eps_{\max} = 0.125$, double $delta = 1$, double $\rho_{end} = 0.5$, double $\beta = 0$, double
 $h_{\max} = \frac{2-\beta}{1+\beta}$, $h_0 = h_{first} = 0.1$

$$F = f_{Newton} = f_{first}, damp = \min(1, \frac{1}{h_0})$$

18.2.2 Subroutine Accuracies(int k)

$$\epsilon = \min(\epsilon_{\max}, \beta * h_k / 2)$$

$$\delta = \max(\epsilon * F^k, \rho_{end} * tol)$$

$$\epsilon_k = \min(\frac{\beta}{4}, \frac{\delta}{F^k})$$

18.2.3 Startschritt $k = 0$

Call Accuracies(k=0)

Bestimme $F_0 = F(p_0; y(p_0))$

if ($F_0 \leq tol$) dann fertig.

18.2.4 Iterationsschleife $k=0,1,..k_{\max}$

1. Dämpfungsfaktor festlegen $damp = \min(1, \max(damp, \frac{1}{h_k}))$
if ($damp < 1e - 10$) dann fertig, Meldung „Dämpfungsfaktor zu klein“
2. Call Accuracies(k)
3. Bestimme eine Approximation der Jacobi-Matrix $\tilde{J}_k(p^k) = F_p^k(y(p^k))$ durch
Bildung des einseitigen Differenzenquotienten mit $\Delta = 1e - 8$ (bei IND) oder
 $\Delta = \sqrt{eps}$
4. Löse durch eine QR-Zerlegung der Jacobi-Matrix die Gleichung $\tilde{J}(p^k)\Delta p^k =$
 $-F(p^k; y(p^k))$ und bestimme daraus die Änderung der Parameter Δp^k .
Bilde a priori-Werte

$$F^{k+1} = F_{apriori} = F^k \cdot h_k \cdot (1 + \beta) / 2 / (1 - \epsilon_k), \text{ wenn } damp = 1$$

$$F^{k+1} = F_{apriori} = F^k / 2, \text{ wenn } damp < 1$$

$$h_{k+1} = h_{apriori} = h_k^2 \cdot (1 + \beta) / 2 / (1 - \epsilon_k), \text{ wenn } damp = 1$$

$$h_{k+1} = h_{apriori} = h_k / F^k \cdot F_{apriori}, \text{ wenn } damp < 1$$

5. Accuracies(k+1) auf Basis der apriori bestimmten Werte bestimmen

6. Bestimme neue Parameter $p^{k+1} = p^k + damp \cdot \Delta p^k$.
Bei Vorzeichenwechsel eines Parameters: $p^{k+1} = p^k + damp \cdot \frac{1}{2} \cdot \Delta p^k$ mit 1 so klein wie möglich

7. Monotonietest: Bestimme $F^{k+1} = F(p_{k+1}; y(p_{k+1}))$

$$\begin{aligned}
 h_k &= h_{aposteriori} = 2 * (1 - \epsilon_{k+1}) / (1 + \beta) \cdot F^{k+1} / F^k, \text{ wenn } damp = 1 \\
 h_k &= h_{aposteriori} = \max(1, 2/damp^2 / \left| \left| F^{k+1} - (1 - damp) \cdot F^k \right| \right| / \left| \left| F^k \right| \right|), \\
 &= \text{Kantorovich-Schätzung, wenn } damp < 1 \tag{107} \\
 h_{k+1} &= h_{apriori} = h_k^2 \cdot (1 + \beta) / 2 / (1 - \epsilon), \text{ wenn } damp = 1 \\
 h_{k+1} &= h_{apriori} = h_k / F^k \cdot F^{k+1}, \text{ wenn } damp < 1
 \end{aligned}$$

if($F^{k+1} \leq tol$) dann fertig

if($(F^{k+1} - F^k) / F^k < tol$) dann Meldung „Minimum der Zielfunktion erreicht“

8. sonst: nächster Schritt k+1

18.3 Direkte Verfahren (Boxsuche und Simulated Annealing)

In (nahezu) allen relevanten Problemen steht die Ableitung des Zielfunktional nicht explizit zur Verfügung. Daher muss diese innerhalb der Anwendung ableitungsorientierter Verfahren numerisch bestimmt werden. Das bringt sowohl Richtungsungenauigkeiten als auch erheblichen Rechenaufwand mit sich. Ausserdem ist das Auffinden der globalen Lösung nicht immer gewährleistet. Neben den ableitungsorientierten Optimierungsverfahren gibt es aber eine Vielzahl von direkten Verfahren, die nur mit Hilfe der Auswertung des Zielfunktional eine Näherung der optimalen Lösung finden wollen. Durch den Verzicht auf Ableitungen (geometrisch Krümmungen und Steigungen) umgehen diese Verfahren das Problem der Eindeutigkeit der Lösung: auch multimodale Probleme mit mehreren Minima können behandelt werden. Auch in Hinblick auf den Rechenaufwand sind diese Verfahren meist günstiger als ableitungsorientierte, die das Zielfunktional vielfach auswerten müssen. Dieser Vorteil wird allerdings dadurch zunichte gemacht, dass die Anzahl Schritte, die diese Verfahren machen müssen, um ein Minimum zu finden, meist erheblich größer ist die von ableitungsorientierten. Wenn also der Aufwand für die Ableitungsberechnung vertretbar hoch ist, sollten direkte Verfahren nicht als Hauptwerkzeug, sondern nur als Hilfsmittel gesehen werden, um Startwerte für ein ableitungsorientiertes Verfahren zu liefern.

Direkte Verfahren lassen sich nach verschiedenen Kriterien unterscheiden, zum Beispiel

1. ob sie mit vordefinierten Punktmengen arbeiten (passives Verfahren) oder selber Punkte bestimmen, an denen das Zielfunktional ausgewertet wird (aktive Verfahren), oder
2. ob sie Auswertungspunkte nach Regeln (deterministisch) bestimmen oder (stochastisch) mit Hilfe eines Zufallsgenerators, oder

3. ob sie (aktive, stochastische Verfahren) Entscheidungen für neue Punkte rein zufällig oder aufgrund von vorhergegangenen Auswertungen treffen, oder
4. ob lokale Optimierungsverfahren miteingebunden werden (können)

Ein wirklich erfolgversprechendes direktes globales Optimierungsverfahren untersucht zum Einen möglichst den gesamten zur Verfügung stehenden Raum auf Minima, zum Anderen nimmt es aber bei Punkten mit „guten“ Zielfunktionalwerten weitere Auswertungen in der Nähe dieses Punktes vor in der Annahme, dass dort möglicherweise noch bessere Punkte zu erreichen sind.

Es ist sicher sinnvoll, innerhalb einer Problemuntersuchung verschiedene Verfahren anzuwenden; allerdings sollte man sich aus Gründen der Aufwandsminimierung auf zwei bis drei unterschiedliche, dafür aber erfolgversprechende Verfahren beschränken. Aus diesem Grund werden im folgenden nur die Verfahren ausführlich dargestellt, die sich in der Untersuchung der zugrundeliegenden Probleme als sehr hilfreich erwiesen haben.

18.3.1 Residuenminimierung deterministisch

Boxsuche Das einfachste solcher direkter (passiver) Verfahren ist das Abscreenen eines definierten Bereichs. Für jeden Parameter $p_k, k = 1, ..m$ werden sowohl Grenzen als auch eine Unterteilungsanzahl u_k oder u_k fest definierte Punkte innerhalb dieser Grenzen angegeben. (In der tatsächlichen Implementierung in Presto-Kinetics wird noch die Achseneinteilung linear oder logarithmisch abgefragt, da die Größenordnungen vieler Reaktionsparameter eine logarithmische Einteilung erfordert) Aus diesen Angaben werden kombinatorisch alle möglichen Parametertupel zusammengestellt, und das Funktional wird an diesen Stellen ausgewertet. Dieses Verfahren, als Boxsuche in Presto-Kinetics implementiert, gibt einen guten Überblick über die Residuenlandschaft und zeigt auch mögliche verschiedene Minima an. Nach einem Durchlauf können die Grenzen nachgestellt und eine feinere Boxsuche durchgeführt werden. Nachteil dieser Vorgehensweise ist, dass die Auswertung bei Vorliegen von vielen Parametern sehr (zeit-)aufwändig ist, und auch, dass die visuelle Darstellung der Residuenlandschaft als eine Projektion auf 2 Richtungen bei vielen Parametern nicht mehr zu überschauen ist. Dennoch sollte dieses Verfahren dann, wenn die Funktionsauswertung nicht zu aufwändig ist und nicht zuviele Parameter geschätzt werden, benutzt werden, um gute Startwerte für ein ableitungsorientiertes Verfahren oder für ein zusätzliches, aktives Verfahren zu erhalten.

Ein offensichtlicher Mangel der Boxsuche ist der, dass dieses Verfahren keine intelligente, zielgerichtete Verfeinerungsvorschrift enthält. Dennoch ist es aufgrund seiner Zuverlässigkeit in der Aussagekraft und dem programmtechnisch geringen Aufwand (kein Vorliegen von heuristischen Parametern) anderen deterministischen Verfahren wie etwa dem Simplex-Verfahren vorzuziehen.

Simplex-Verfahren Die Simplex-Methode von Nelder und Mead ist ein direktes, iteratives Verfahren, welches Informationen aus vorangegangenen Schritten benutzt,

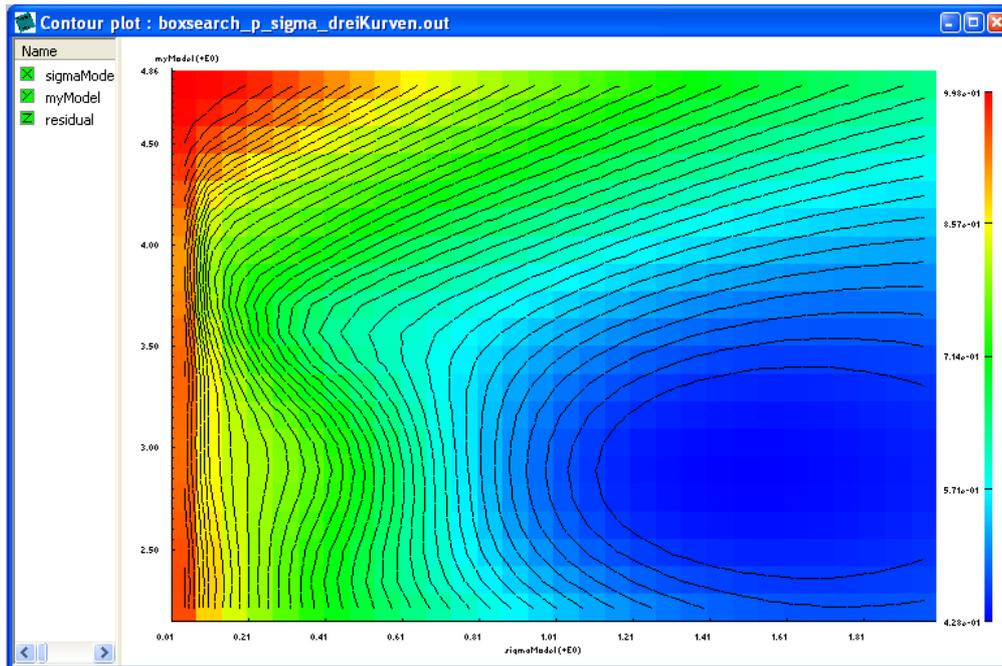


Abbildung 37: Projektierung der Ergebnisse einer Boxsuche führt im günstigen Fall zur erkennbaren Einsicht in die Residuenlandschaft, rot Gebiete mit großem Residuum, dunkelblau Gebiet mit niedrigem Residuum

um auf effektive Art und Weise neue Punkte zu bestimmen. Ausgehend von einem $m + 1$ -dimensionalen Start-Simplex werden die Funktionswerte $F(p_i)$ aller Eckpunkte berechnet und der Größe ihres Funktionswertes nach geordnet: $F_1 \leq \dots F_k \leq F_{m+1}$. Auf jeder Stufe des Algorithmus wird nun der Punkt p_{m+1} mit dem jeweils schlechtesten Ergebnis durch einen neuen Punkt p_N ersetzt. Damit erreicht man, dass sich in jedem Schritt ein neuer Simplex ergibt. Dieser neue Punkt ergibt sich zunächst durch Reflektion des schlechtesten Punktes am Schwerpunkt des (Rest-)Simplex.

$$c = \frac{1}{m} \sum_{j=1}^m p_j \quad (108)$$

Dann wird nach gewissen Regeln entschieden, wie weiter mit diesem Punkt verfahren wird. Abhängig von dem Funktionswert im neuen Punkt ergeben sich verschiedene Vorschriften:

1. $F(p_1) \leq F(p_N) \leq F(p_m)$: der neue Funktionswert lässt sich einsortieren, der schlechteste Punkt wird durch den neuen Punkt p_N ersetzt
2. $F(p_N) \leq F(p_1)$: die Suchrichtung ist gut, es wird eine Expansion in dieser Richtung vorgenommen, und abhängig von dem Funktionswert wird entweder der neue Punkt oder der expandierte Punkt übernommen. Für diese Regel ist die Einführung einer Konstanten α erforderlich, die regelt, wie groß die Expansion

sein darf. Für die Wahl dieser Konstanten wiederum ist eine Heuristik nötig, ein typischer Wert ist $\alpha = 2$.

3. $F(p_N) \geq F(p_{km})$ das Simplex ist zu groß und wird in Richtung des schlechtesten Punktes verkleinert. Ist der dann erzeugte neue Punkt besser, siehe Regel 1. Gilt noch immer $F(p_N) \geq F(p_m)$, so wird das Simplex um den besten Punkt p_1 verkleinert, indem man alle anderen Punkte durch neue Punkte $p_j = p_1 + \delta(p_j - p_1)$ ersetzt. Auch für δ ist eine Belegungsvorschrift nötig, $\delta = 0.5$ ist ein typischer Wert. Durch die Einführung der neuen Punkte sind m zusätzliche Funktionsauswertungen notwendig.

Konvergenzkriterien sind zum Beispiel ein kleiner Abstand zwischen $F(p_1)$ und $F(p_{m+1})$ oder die Streuung der Punkte p_j . Das Verfahren garantiert nicht das Auffinden eines Minimums. In der Literatur finden sich viele Modifikationen dieses Algorithmus.

18.3.2 Residuenminimierung stochastisch

Ein anderer Ansatz, das Minimum eines Funktionals zu finden, ist der, zufällig ausgewählte Punkte eines gegebenen Bereiches auszuwerten. Der Metropolis-Algorithmus zum Beispiel wählt einen beliebigen Punkt p^k und wertet die Zielfunktion dort aus. Ist der erreichte Wert kleiner als der bisher beste, wird dieser Punkt als optimaler Punkt akzeptiert. Ist der erreichte Wert nicht kleiner als der bisher beste, so wird der Punkt unter Umständen trotzdem akzeptiert.

Solche heuristischen Verfahren garantieren weder das Auffinden der optimalen Lösung noch sind sie (aufgrund der Benutzung eines Zufallszahlengenerators) wiederholbar, d.h. sie verletzen die Annahme der Vorhersagbarkeit, wie sie im Allgemeinen für Algorithmen gefordert wird. Dieser Nachteil ist nicht nur ein dialektischer: die fehlende Reproduzierbarkeit von Ergebnissen führt durchaus zu Akzeptanzeinbußen beim Anwender.

Die meisten gebräuchlichen stochastischen Verfahren basieren auf Prinzipien, die der Natur abgeschaut sind. Das Simulated Annealing zum Beispiel ergibt sich aus der Beobachtung der Abkühlung von Flüssigkeiten oder Kristallstrukturen:

Simulated Annealing Ausgangspunkt ist, wie bei der Boxsuche, die Angabe eines Bereiches, innerhalb dessen das Simulated Annealing durchgeführt werden soll. Für jeden Parameter $p_k, k = 1, \dots, m$ werden demnach Grenzen angegeben. Ausgehend von einem Startpunkt p^0 wertet das Verfahren das Zielfunktional am aktuellen Punkt p^k aus ($f(p^k)$), springt dann mit Hilfe einer zufallsgenerierten Zahl zu einem anderen Punkt p^{k+1} innerhalb des Bereiches und wertet erneut aus $f(p^{k+1})$. Ist das neue Residuum $f(p^{k+1})$ kleiner als das bisher beste ($f(p^{opt})$), so wird dieses mit dem zugehörigen Punkt p^{k+1} als bester Wert gesichert und der nächste Punkt wird zufällig gewählt. Ist das neue Residuum nicht kleiner, so wird das Wertepaar $(p^{k+1}, f(p^{k+1}))$ nicht unbedingt verworfen: wenn der Quotient aus $f(p^{opt}) - f(p^{k+1})$ und einem „Temperaturwert“ T unter der Exponentialfunktion kleiner ist als ein zufällig gewählter Wert $\in [0, 1]$, so wird mit einer (vorgegebenen) Anzahl an Gauss-Newton-Schritten

untersucht, welcher bestmögliche Residuumswert von diesem Punkt aus zu erreichen ist. Damit unterscheidet sich das Simulated Annealing deutlich vom vorher angesprochenen Metropolis-Algorithmus und macht es zu einem effektiven Verfahren.