



ist  $\mu$  hier nicht explizit bekannt. Da aber ein direkter Zusammenhang zwischen  $\mu$  und dem Spektralradius  $\rho(H(\omega))$  der Iterationsmatrix  $H(\omega)$  des SOR-Verfahrens besteht, kann  $\mu$  aus den bei der Iteration berechneten Größen geschätzt werden. Damit soll der Relaxationsparameter  $\omega$  während der Rechnung korrigiert werden. Als Startwert ist  $\omega = 1$  (Einzelschrittverfahren) zu wählen. Eine Strategie zur Anpassung von  $\omega$  ist in [2,S.225 ff] beschrieben. Es ist darauf zu achten, dass kein (wesentlicher) zusätzlicher Speicherbedarf bei der Schätzung von  $\omega_b$  entsteht. Durch ein Protokoll (Graphik) der wichtigsten Größen ist der Verlauf des Verfahrens zu dokumentieren.

- Literatur:**
1. Stoer, Bulirsch: Numerische Mathematik II
  2. Hageman, Young: Applied iterative methods.

**Aufgabe 2 (Block-Iterationen):**

Auch für ganze Matrizen  $X \in \mathbb{R}^{m \times n}$  kann man lineare Gleichungssysteme betrachten, mit  $A \in \mathbb{R}^{m \times m}$ ,  $B \in \mathbb{R}^{n \times n}$  und  $C \in \mathbb{R}^{m \times n}$  ist die sogenannte Sylvestergleichung

$$AX + XB = C \tag{1}$$

ein lineares Problem. Speichert man die Spalten der Matrizen  $X$  und  $C$  in großen Vektoren  $x, c \in \mathbb{R}^{mn}$ , ist die Sylvestergleichung tatsächlich äquivalent mit einem linearen System  $(\tilde{A} + \tilde{B})x = c$  mit der Eigenschaft, dass die beiden Matrizen kommutieren,  $\tilde{A}\tilde{B} = \tilde{B}\tilde{A}$ . Diese Strukturen kann man sich bei Iterationsverfahren zu Nutze machen, etwa wenn  $A$  und  $B$  definit sind.

- E.** Block-Einzelschrittverfahren: Hier löst man der Reihe nach die  $j$ -te Spalte des Systems auf nach dem entsprechenden Spaltenvektor  $x^{(j)} = Xe^{(j)}$  und bekommt den Iterationsschritt

$$(A + b_{jj}I)x^{(j)} := c^{(j)} - \sum_{i \neq j} x^{(i)}b_{ij}, \quad j = 1, \dots, n,$$

bei dem nur ein kleineres  $m \times m$ -System zu lösen ist.

- A.** ADI-Verfahren: Mit einem Parameter  $r > 0$  formt man die Sylvestergleichung um in  $(A + rI)X + X(B - rI) = C$  bzw.  $X(B + rI) + (A - rI)X = C$  und nutzt diese als Iterationsvorschrift,

$$\begin{aligned} (A + rI)Y^{(k)} &= C - X^{(k-1)}(B - rI), \\ X^{(k)}(B + rI) &= C - (A - rI)Y^{(k)}, \end{aligned} \quad k \in \mathbb{N},$$

in der jeweils auch nur Systeme der Größe  $m$  bzw.  $n$  auftreten.

Bei Diskretisierung der Poissongleichung  $-\Delta u(t_1, t_2) = g(t_1, t_2)$  auf dem Einheitsquadrat durch Differenzen auf einem gleichförmigen Gitter mit  $n + 1$  Knoten in jeder Richtung sind die Gitterwerte  $x_{ij} \cong u(\frac{i}{n+1}, \frac{j}{n+1})$ ,  $i, j = 1, \dots, n = m$  die Unbekannten in einem System (1), vgl. Literatur §8. Die Matrizen  $A, B$  sind gleich und sogar symmetrisch und tridiagonal. Daher ist der Gesamtaufwand in jedem Schritt der beschriebenen Verfahren nur  $O(mn)$  und daher in der Größenordnung des einfachen Einzelschrittverfahrens, die Konvergenzgeschwindigkeit aber höher. Zur Implementierung benötigt man Unterprogramme, welche mit einer Choleskyzerlegung  $LL^T = A + rI$  Systeme der Formen  $LL^TY = R$  und  $XLL^T = S$  lösen. Vergleichen Sie beide Verfahren für  $n = 32$ .

Literatur: Stoer/Bulirsch, Numerische Mathematik 2

### Aufgabe 3 (Iteration mit Tensorproduktstruktur):

Für die Neumannreihe einer Matrix  $B \in \mathbb{R}^{n \times n}$  mit  $\rho(B) < 1$  gilt folgende Faktorisierung

$$(I - B)^{-1} = \sum_{j=0}^{\infty} B^j = (I + B)(I + B^2)(I + B^4) \cdots$$

Durch fortgesetztes Quadrieren von  $B$  bekommt man schnell sehr lange Partialsummen der Reihe und gute Näherungen einer Lösung  $(I - B)^{-1}r$  durch die Iteration

$$x^{(k+1)} := (I + B_k)x^{(k)}, \quad B_{k+1} := B_k^2, \quad k = 0, 1, \dots,$$

mit  $x^{(0)} := r$ ,  $B^{(0)} := B$ . Im Allgemeinen ist das aber nicht effizient wegen des hohen Aufwands  $O(n^3)$  bei der Matrix-Quadrierung. Dies ändert sich bei speziellen Strukturen.

Man kann Gleichungssysteme auch für ganze Matrizen  $X \in \mathbb{R}^{m \times m}$  betrachten, z.B. mit  $A, D, R \in \mathbb{R}^{m \times m}$  und  $\rho(A)\rho(D) < 1$  ein lineares System der Form

$$X - AXD = R. \tag{2}$$

Offensichtlich ist  $X \mapsto X - AXD$  eine lineare Abbildung auf dem  $\mathbb{R}^{m \times m}$ , deren Matrix  $I - B \in \mathbb{R}^n$ ,  $n = m^2$ , man sogar explizit angeben kann. Für dieses System lautet das obige Verfahren

$$X_{k+1} := X_k + A_k X_k D_k, \quad A_{k+1} := A_k^2, \quad D_{k+1} := D_k^2, \quad k = 0, 1, \dots,$$

mit  $X_0 := R$ ,  $A_0 = A$ ,  $D_0 = D$ . Der Aufwand pro Schritt ist jetzt nur noch  $O(m^3) = O(n^{3/2})$ . Verifizieren Sie die Methode zunächst.

Bei Diskretisierung der erweiterten Poissongleichung  $-\Delta u(t_1, t_2) + au(t_1, t_2) = 1$  auf dem Einheitsquadrat  $0 \leq t_1, t_2 \leq 1$  durch Differenzen auf einem gleichförmigen Gitter mit  $m + 1$  Knoten in jeder Richtung sind die Gitterwerte  $x_{ij} \cong u(\frac{i}{n+1}, \frac{j}{m+1})$ ,  $i, j = 1, \dots, m$  die Unbekannten in einem linearen System der Form  $bX - TX - XT = G$  mit  $b = a + 4/(m + 1)^2$ , vgl. Literatur §8. Dieses läßt sich leicht in die Gestalt (2) bringen. Programmieren Sie hierfür das angegebene Iterationsverfahren, welches so lange durchgeführt werden soll, bis für den Defekt gilt  $\|bX_k - TX_k - X_kT - G\|_{\infty} \leq 10^{-10}$ . Vergleichen Sie die Laufzeit mit der Auflösung des zugehörigen Bandsystems mit dem Gauß-Algorithmus, jeweils für  $a \in \{0, 4\}$  und  $m = 32, 128$ .

Literatur: Stoer/Bulirsch, Numerische Mathematik 2

### Aufgabe 4 (Vergleich von iterativen Verfahren):

Vergleichen Sie die verschiedenen iterativen Verfahren, welche in der Marburg Software Library (MSL) implementiert sind. Machen Sie sich dazu zuerst mit der MSL und insbesondere mit den Dateien `MathTL/numerics/iteratsolv.h/cpp` und `MathTL/numerics/test_iteratsolv` vertraut. Alle hier enthaltenen Algorithmen dienen zur Lösung eines linearen Gleichungssystems

$$Ax = b.$$

Der Dokumentation ist zu entnehmen, welche Anforderungen an die Matrix  $A$  die jeweiligen Verfahren verlangen. Die Konvergenzgeschwindigkeit des Richardson-Verfahren, welches nach der Vorschrift

$$x^{(k+1)} = x^{(k)} + \omega(b - Ax^{(k)}), \quad x^{(0)} := 0,$$

iteriert, hängt insbesondere von der Wahl des Parameters  $\omega > 0$  ab. Testen Sie die in `MathTL/numerics/test_iteratsolv` angegebene Matrix mit verschiedenen Werten von  $\omega$ . Welches ist der optimale Wert und wie lässt er sich bestimmen? Plotten Sie das Residuum  $Ax^{(k)} - b$  gegen die Anzahl der Iterationen in doppelt logarithmischer Skala. Hierzu lassen Sie sich die benötigten Werte in einer Matlab-Datei ausgeben. Vergleichen Sie das Richardson-Verfahren mit dem Jacobi- und dem CG-Verfahren. Plotten Sie die Residuen dieser Verfahren ebenfalls in logarithmischer Skala und schauen Sie für welche Wahl der  $\omega$  das Richardson-Verfahren mit den anderen Verfahren mithalten kann.

**Link** <https://wavelet-and-multiscale-library.github.io/>

### Aufgabe 5 (Spline-Approximation verstreuter Daten):

In vielen Anwendungen will man mehrdimensionale Daten durch Funktionen approximieren, auch wenn zugehörige Messdaten nur in unregelmäßig verteilten Punkten bekannt sind. Zum Beispiel kann man bei Zeitreihen mit vielen Daten,  $(t_k, x_k, y_k, z_k, \dots)$ ,  $k \in \mathbb{N}$ , versuchen, einen funktionalen Zusammenhang in der Form  $z_k \cong \phi(x_k, y_k)$  herzustellen bzw. zu prüfen.

Programmieren Sie dazu folgenden Ansatz für Daten, die von 2 Parametern  $(x, y)$  abhängen. Zur Approximation auf einem Rechteck  $\Omega := [a, b] \times [c, d]$  ist ein Produktansatz mit B-Splines zu verwenden in der Form

$$\phi(x, y) = \sum_{i,j=1}^{\ell} \alpha_{ij} B_i(x) B_j(y), \quad (x, y) \in \Omega.$$

Der Ansatz besitzt also  $n = \ell^2$  Freiheitsgrade. Die Standard-B-Splines aus der Numerik-1 sind dazu durch geeignete Rand-Splines zu ergänzen (vgl. Literatur). Mit diesem Ansatz ist eine Kleinste-Quadrate-Approximation von  $m > n$  Meßdaten  $(x_k, y_k, z_k) : k = 1, \dots, m$ , zu bestimmen, welche das Residuum

$$\sum_{k=1}^m (z_k - \phi(x_k, y_k))^2$$

minimiert. Stellen Sie dazu das überbestimmte Gleichungssystem für die Koeffizienten  $\alpha_{ij}$  auf, lösen Sie dieses mit einer QR-Zerlegung und stellen Sie den Funktionsgraphen in einer einfachen 3D-Graphik dar (z.B. Gitterfunktion).

Testen Sie das Verfahren mit  $\ell^2 = 16$  Freiheitsgraden an Zeitreihen von Finanzdaten und stellen Sie den DAX-Kurs dar als Funktion der Umlaufrendite und des Goldpreises.

**Literatur** de Boor: A practical guide to splines

### Aufgabe 6 (Newton-Liniensuche):

Die Konvergenz des einfachen Newtonverfahrens  $x^{(k+1)} := x^{(k)} - f'(x^{(k)})^{-1} f(x^{(k)})$ ,  $k = 0, 1, \dots$  zur Bestimmung von Nullstellen einer genügend glatten Funktion  $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$  ist nur in genügender Nähe zu einer Nullstelle sicher. Zur Verbesserung des globalen Konvergenzverhaltens kann man eine Liniensuche durchführen. Dabei betrachtet man mit einer Näherung  $\tilde{x}$  den Vektor  $s := -f'(\tilde{x})^{-1} f(\tilde{x})$  nur als Suchrichtung und sucht eine bessere Näherung auf dem Strahl  $y(t) := \tilde{x} + ts$ . Die Güte der Näherung auf dem Strahl kann man über den Wert der Kontrollfunktion

$$g(x) := \frac{1}{2} \|f(x)\|^2, \text{ also mit } \varphi(t) := g(y(t))$$

prüfen. Für genügend kleine  $t > 0$  ist eine Verbesserung möglich, da  $\varphi'(0) = -2\varphi(0) < 0$  gilt,  $s$  also eine Abstiegsrichtung von  $g$  ist. Wenn allerdings die Ableitung  $f'$  eine große Konditionszahl besitzt, ist  $s$  eine schlechte Richtung, da sie beinahe orthogonal zum Gradienten der Kontrollfunktion  $g$  ist. Eine Verbesserung bekommt man, indem man nicht die Suchrichtung ändert, sondern eine bessere Kontrollfunktion

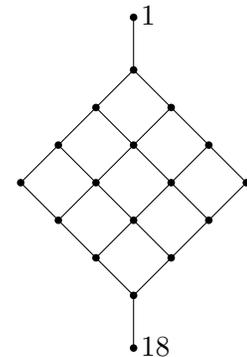
$$G(x) := \frac{1}{2} \|f'(\tilde{x})^{-1}f(x)\|^2, \text{ also } \Phi(t) := G(y(t))$$

auf dem Strahl bei  $\tilde{x} = x^{(k)}$  betrachtet. Hier ist die Newtonrichtung tatsächlich die Richtung des stärksten Abstiegs. Mit diesen Kontrollfunktionen führt man die Liniensuche so durch, dass man mit einem  $\sigma < 1$  die Schrittweite  $\hat{t}$  so bestimmt, dass

$$\varphi(\hat{t}) \leq (1 - \sigma\hat{t})\varphi(0), \text{ bzw. } \Phi(\hat{t}) \leq (1 - \sigma\hat{t})\Phi(0) \quad (3)$$

gilt. Dazu kann man, ausgehend vom ersten Versuch  $t_0 = 1$ ,  $\hat{t}$  z.B. als Minimalstelle des Interpolationspolynoms zu  $\varphi(0), \varphi'(0)$  und  $\varphi(t_0)$  bzw.  $\Phi(0), \Phi'(0)$  und  $\Phi(t_0)$  wählen. Falls  $\hat{t}$  die Bedingung (3) nicht erfüllt, wird diese Konstruktion mit  $t_0 := \hat{t}$  wiederholt. Eine Lösung wird bei  $\|s\| \leq 10^{-6}$  akzeptiert.

In einem nichtlinearen Netzwerk (z.B. Rohrnetz) mit Knoten  $1, \dots, n$  ergibt sich der Fluß durch eine Kante  $i \rightarrow j$  aus der Differenz der Drücke  $x_i$  und  $x_j$  in der Form  $a_{ij} \cdot d(x_i - x_j)$ . Dabei ist  $d(t)$  eine ungerade und monoton wachsende Funktion und  $a_{ij} = a_{ji} > 0$ , wenn eine Kante zwischen  $i$  und  $j$  existiert. Es sei nun  $n = 18$  und das skizzierte Netzwerk gegeben. In jedem inneren Knoten gilt dann die Massenerhaltung



$$f_i(x) := \sum_{k=1}^n a_{ik}d(x_i - x_k) = 0, \quad i = 2, \dots, n-1,$$

im Anfangs- und Endknoten sei  $x_1 := 0$  und  $x_n := 100$ . Weiter sei  $d(t) := t/(1 + |t|)$ , die Sättigungsgrenze des Flusses liegt also bei 1.

Vergleichen Sie die Liniensuche mit beiden Kontrollfunktionen  $g$  und  $G$  und dem Startwert  $x_i^{(0)} = 0, 1 \leq i < n$  bei dem gezeigten Netzwerk. Dabei sollen jeweils 20 zufällige erzeugte Belegungen der Gewichte  $a_{ij} \in (0, 1]$  der gezeigten Kanten verwendet werden.

Literatur: Freund/Hoppe: Stoer/Bulirsch – Numerische Mathematik 1

### Aufgabe 7 (Halbglatt-Newton-Verfahren):

Seien  $X, Y$  vollständige normierte Vektorräume (sogenannte Banach-Räume) und sei  $D \subset X$  eine offene Menge. Eine lokale Lipschitz-stetige Abbildung  $\mathcal{F} : D \rightarrow Y$  heißt NEWTON-DIFFERENZIERBAR, falls eine Familie beschränkter linearer Abbildungen  $\mathcal{G} : U \rightarrow \mathcal{L}(X, Y)$  existiert, so dass

$$\|\mathcal{F}(x+h) - \mathcal{F}(x) - \mathcal{G}(x+h)h\|_Y = o(\|h\|), \quad \forall x \in U$$

gilt. Dabei ist  $U \subset D$  eine offene Menge und  $\mathcal{L}(X, Y)$  der Raum aller beschränkter linearer Abbildungen von  $X$  nach  $Y$ .  $\mathcal{G}$  heißt NEWTON-ABLEITUNG von  $\mathcal{F}$  in  $U$ . Beachten Sie den Unterschied zum klassischen Ableitungskonzept, indem der Basispunkt von der Ableitung im

Gegensatz zur Newton-Ableitung nicht variiert.

Liegt eine Newton-Ableitung für eine nicht-glatte, lokale Lipschitz-stetige Abbildung  $\mathcal{F}$  vor, dann kann man unter weiteren Annahmen folgende Iteration zur Lösung des nichtglatten Gleichungssystems  $\mathcal{F}(x) = 0$  verwenden:

1. Sei  $x^0 \in U$ ,  $k := 0$ .

2. Löse

$$\mathcal{G}(x^k)\delta x^k = -\mathcal{F}(x^k)$$

nach  $\delta x^k$  auf.

3. Setze  $x^{k+1} := x^k + \delta x^k$  und  $k := k + 1$ .

4. Falls ein Abbruchkriterium gilt, dann STOP. Sonst GO TO 2.

Dieses Verfahren konvergiert lokal superlinear falls gilt:

1. Es existiert ein  $x^* \in U$  mit  $\mathcal{F}(x^*) = 0$ .

2.  $\mathcal{G}(x)$  ist für alle  $x \in U$  nicht singulär.

3. Die Menge  $\{\|\mathcal{G}(x)^{-1}\|_{\mathcal{L}(Y,X)} : x \in U\}$  ist beschränkt.

4. Der Abstand  $\|x^0 - x^*\|_X$  ist hinreichend klein.

### Anwendung: Halbglatt-Newton für Variationsungleichungen

Der wichtigste nichtglatte Operator in der Optimierung ist der komponentenweise  $\max(0, \cdot)$ -Operator, der wie folgt definiert ist. Sei  $x \in \mathbb{R}^n$ , dann ist

$$(x)^+ := \begin{bmatrix} \max(0, x_1) \\ \vdots \\ \max(0, x_n) \end{bmatrix}$$

In diesem Fall ist  $\mathcal{G} : \mathbb{R}^n \rightarrow \mathbb{R}^{n \times n}$  mit

$$\mathcal{G}_{ii}(y) = \begin{cases} 1 & \text{if } y_i > 0, \\ 0 & \text{if } y_i \leq 0, \end{cases}$$

$\mathcal{G}_{ij}(y) = 0$ ,  $i \neq j$ , für jede  $y \in \mathbb{R}^n$  eine Newton-Ableitung von  $(\cdot)^+$  im Punkt  $y$ .

Seien  $x, \lambda \in \mathbb{R}^n$ . Zeigen Sie, dass die Komplementaritätsrelation

$$x_i \geq 0, \quad \lambda_i \geq 0, \quad x_i \lambda_i = 0, \quad \forall i = 1, \dots, n$$

äquivalent zum nichtglatten Gleichungssystem ist:  $\lambda = (\lambda - cx)^+$ . Dabei ist  $c$  eine beliebige positive Konstante.

Sei  $V := H_0^1(0,1)$  der Sobolev-Raum aller  $L^2$ -Funktionen  $u$  mit  $u(0) = u(1) = 0$ , deren (schwache) Ableitungen auch in  $L^2(0,1)$  sind. Wegen der Poincaré-Friedrichs-Ungleichung können wir die Norm

$$\|u\|_V := \left( \int_0^1 |u_x(x)|^2 dx \right)^{1/2}$$

nehmen. Damit ist

$$(u, v)_V = (u_x, v_x)_{L^2} = \int_0^1 u_x(x)v_x(x)dx$$

ein inneres Produkt auf  $V$ .

Betrachtet wird folgendes unilaterales variationelles Problem:

$$\text{Finde } u \in K : (u, v - u)_V \geq \nu(f(u), v - u)_{L^2}, \quad v \in K. \quad (4)$$

Dabei sei  $K := \{u \in H_0^1(0, 1) : u(x) \leq \Psi(x), \text{ f.f.a } x \in (0, 1)\}$  mit  $\Psi \in C[0, 1]$   $\nu > 0$  und  $f : \mathbb{R} \rightarrow \mathbb{R}$  eine zweimal stetig differenzierbare Funktion.

Sei  $\Omega := (0, 1)$ ,  $N \in \mathbb{N}$ ,  $h := \frac{1}{N+1}$  und  $\Omega_h$  eine Zerlegung von  $\Omega$  in  $N + 1$  Intervalle der Breite  $h$ . Sei dazu  $\{\psi_1, \dots, \psi_N\}$  eine entsprechende nodale Basis. Diskretisieren Sie (4), indem Sie den Ansatz

$$u_h(x) := \sum_{i=1}^N U_i^h \psi_i(x),$$

mit  $U_h := (U_1^h, \dots, U_N^h) \in \mathbb{R}^N$  in die Variationsungleichung (4) einsetzen. Nach Einführung von Schlupfvariablen  $S_1^h, \dots, S_N^h$ , sollten Sie ein System der Gestalt:

$$A_h U_h + S_h = \nu F(U_h), \quad (5a)$$

$$S_h = (S_h - c(\Psi_h - U_h))^+. \quad (5b)$$

erhalten.

Leiten Sie eine explizite Form für eine Halbglatt-Newton-Iteration mit der gegebenen Formel für die Newton-Ableitung von  $(\cdot)^+$  her. Sie dürfen annehmen, dass die übliche Kettenregel für die Newton-Ableitung gelte. Hinweis: für ein beliebiges Paar  $(U_h, S_h)$  führen Sie folgende Mengen ein:

$$\begin{aligned} \mathcal{A}(U, S) &:= \left\{ i \in \{1, \dots, N\} : S_i^h - c(\Psi_i^h - U_i^h) > 0 \right\}, \\ \mathcal{I}(U, S) &:= \left\{ i \in \{1, \dots, N\} : S_i^h - c(\Psi_i^h - U_i^h) \leq 0 \right\}. \end{aligned}$$

Wenden Sie nun die Halbglatt-Newton-Iteration zuerst nur auf (5b) an. Sie sollten die Regel

$$U_i^h + \delta U_i^h = \Psi_i^h, \quad i \in \mathcal{A}(U, S), \quad S_i^h + \delta S_i^h = 0, \quad i \in \mathcal{I}(U, S)$$

erhalten. Wenden Sie nun die Halbglatt-Newton-Iteration auf (5a) an und vereinfachen Sie mit dieser Regel das daraus entstehendem System. Sie sollten am Ende ein Verfahren der Gestalt:

1. Gegeben  $(U_h, S_h) \in \mathbb{R}^N \times \mathbb{R}^N$
2. Bestimme  $\mathcal{A}$  and  $\mathcal{I}$ .
3. Setze  $U_i^{+,h} = \Psi_i^h$ ,  $i \in \mathcal{A}$ ,  $S_i^{+,h} = 0$ ,  $i \in \mathcal{I}$ .
4. Löse nach  $U_{\mathcal{I}}^{+,h}$  auf:  $L_{\mathcal{I},\mathcal{I}} U_{\mathcal{I}}^{+,h} = b_{\mathcal{I}} - L_{\mathcal{I},\mathcal{A}} \Psi_{\mathcal{A}}^h$
5. Setze  $S_{\mathcal{A}}^{+,h} = b_{\mathcal{A}} - [L U_h^+]_{\mathcal{A}}$ .

für eine bestimmte Matrix  $L$  und einen bestimmten Vektor  $b$ . Dabei ist  $L_{\mathcal{I},\mathcal{A}}$  die Teilmatrix mit Zeilen aus  $\mathcal{I}$  und Spalten aus  $\mathcal{A}$ ; analog für  $b_{\mathcal{I}}$  usw.

Implementieren Sie dieses Verfahren in einer Programmiersprache Ihrer Wahl und lösen Sie (4) für die Fälle

1.  $f(u) \equiv 1, \Psi \equiv 1, \nu = 1$
2.  $f(u) = e^u, \Psi \equiv 7, \nu = 0.25, 0.35, 0.45, 0.5$

Als Abbruchkriterium für das Verfahren verwenden Sie das Residuum des Systems (5). Verfassen Sie eine Tabelle mit dem Verhalten dieses Residuums über alle Iterationen. Geben Sie dazu die (experimentelle) Konvergenzrate der Iterierten  $\{U_h^k\}$  und  $\{S_h^k\}$  an. Plotten Sie für jedes Experiment die Menge  $\mathcal{A}$ , die Sie beim Abbruch bekommen, sowie die Lösung  $U_h^*$  und Schlupfvariablen  $S_h^*$ .

### Aufgabe 8 (Parameter-Schätzung):

Es sei folgendes Modell für die Verschiebung  $y$  eines ein-dimensionalen elastischen Körpers von einem Restzustand gegeben:

$$-\frac{\partial}{\partial x}\left(a\frac{\partial}{\partial x}y(x)\right) + \beta(b(x))y(x) = f(x), \quad x \in (0, 1); \quad y(0) = y(1) = 0. \quad (6)$$

Hier ist  $a > 0$  ein bekannter Materialparameter und  $f \in C[0, 1]$  eine bekannte verteilte Kraft auf den Körper. Der Nemytski-Operator  $\beta$  ist durch die Funktion  $\hat{\beta} : \mathbb{R} \rightarrow \mathbb{R}$  mit

$$\hat{\beta}(z) := \varepsilon + \begin{cases} z - \frac{\varepsilon}{2}, & z \geq \varepsilon, \\ \frac{z^3}{\varepsilon^3} - \frac{z^4}{2\varepsilon^4}, & z \in (0, \varepsilon), \\ 0, & \text{sonst} \end{cases}$$

mit  $\varepsilon > 0$  gegeben. Die Funktion  $b : (0, 1) \rightarrow \mathbb{R}$  ist unbekannt und sollte durch ein Inverse-Problem bestimmt werden. Wir gehen davon aus, dass Messungen  $y_1, \dots, y_m$  an den Stellen  $x_1, \dots, x_m \in (0, 1)$  gegeben sind. Mit diesem Datensatz versuchen wir  $b$  anzunähern, indem wir folgendes Funktional betrachten:

$$J(y, b) := \sum_{i=1}^m |y(b)(x_i) - y_i|^2 + \frac{\nu}{2} \int_0^1 |b(x)|^2 dx, \quad \nu > 0,$$

wobei  $y(b)$  eine Lösung von (6) für ein gegebenes  $b : (0, 1) \rightarrow \mathbb{R}$  ist. Das ergibt folgendes Optimierungsproblem

$$\min \frac{1}{2} \sum_{i=1}^m |y(x_i) - y_i|^2 + \frac{\nu}{2} \int_0^1 |b(x)|^2 dx \quad \text{über } (b, y) \in B \times Y \quad (7a)$$

$$\text{U.d.N.} \quad (7b)$$

$$-\frac{\partial}{\partial x}\left(a\frac{\partial}{\partial x}y(x)\right) + \beta(b(x))y(x) = f(x), \quad x \in (0, 1); \quad y(0) = y(1) = 0, \quad (7c)$$

$$0 \leq b(x) \leq \bar{b}, \quad \text{f.f.a } x \in (0, 1), \quad (7d)$$

wobei  $\bar{b} > 0$  eine fixe obere Schranke ist.

Die Aufgaben dieses Projekts sind wie folgt (einige Details finden Sie unten, alle weitere Details und Erklärungen bekommen Sie im persönlichen Gespräch mit Prof. Surowiec):

1. Bestimmen Sie passende Räume  $B$  und  $Y$  damit (7) einen zulässigen Punkt hat.
2. Verwenden Sie den Satz über implizite Funktionen, um eine Sensitivität des Zustands  $y$  bezüglich der Unbekannten  $b$  herzuleiten. Dabei sollen Sie eine stetig (Fréchet) differenzierbare Funktion  $y(b)$  bekommen, deren Ableitung eine verwandte lineare Differentialgleichung löst
3. Leiten Sie eine Formel für die Ableitung von  $J(y(b), b)$  her.
4. Leiten Sie notwendige Bedingungen erster Ordnung in der Form einer Variationsungleichung her.
5. Lösen Sie (7) mit Hilfe des Verfahrens der projizierten Gradienten.

### Details

1. Offenbar sollte  $b$  mindestens messbar und integrierbar sein. Die Verschiebung  $y$  löst eine Differentialgleichung und sollte deshalb mehr Regularität haben. Hier sollen Sie für die Differentialgleichung eine verwandte,  $b$ -abhängige Bilinearform finden, damit eine schwache Lösung für (6) definiert werden kann. Schauen Sie zum Beispiel in das Buch von Evans "Partial Differential Equations" nach.
2. Lesen Sie im Buch von E. Zeidler „Applied Functional Analysis“ Sec. 4.8 mehr zu diesem Thema nach. Wenn man (6) als Operator-Gleichung der Gestalt

$$e(y, b) := Ay + \beta(b)y - f = 0$$

betrachtet, dann kann man die Bedingungen des Satzes direkt überprüfen und anwenden.

3. Hier können Sie die Kettenregel anwenden. Denn nach 2. ist  $y(b)$  stetig Fréchet-differenzierbar. Es hilft auch an dieser Stelle Auswertungsoperatoren  $E_{x_i} : Y \rightarrow \mathbb{R}$  zu definieren, sodass

$$J(y(b), b) = \frac{1}{2} \sum_{i=1}^m |E_{x_i}(y(b)) - y_i|^2 + \frac{v}{2} \int_0^1 |b(x)|^2 dx$$

ist.

4. Nach 1.-3. können Sie klassische Ansätze aus der Optimierung verwenden, um diese Bedingungen herzuleiten. Für mehr über Variationsungleichungen können Sie im Buch "An Introduction to Variational Inequalities and Their Applications" von Kinderlehrer und Stampacchia lesen.
5. Das Verfahren der projizierten Gradienten ist eine der einfachsten numerischen Optimierungsalgorithmen. In diesem Projekt können Sie ohne weitere Schwierigkeiten die Projektion auf die bilateralen Schranken an  $b$  berechnen. Allerdings ist es notwendig, die Gradienten der Funktion  $\mathcal{J}(b) := J(y(b), b)$  zu bestimmen. Letzteres erfolgt über die Lösung einer sogenannten adjungierten Gleichung.

### Aufgabe 9 (Trust-Region-Verfahren):

Betrachtet wird folgendes optimale Steuerungsproblem:

$$\min f(u) := \frac{1}{2} \int_0^1 |y(x, T; u) - z(x)|^2 dx + \frac{\alpha}{2} \int_0^1 |u(t)|^2 dt \text{ über } (u, y) \in U_{\text{ad}} \times Y. \quad (8a)$$

Dabei ist  $y(x, T; u)$  die Lösung des nichtlinearen parabolischen Problems

$$y_t(t, x) = y_{xx}(t, x), \quad x \in (0, 1), \quad t \in (0, T), \quad (9a)$$

$$y(0, x) = y_0(x), \quad x \in (0, 1), \quad (9b)$$

$$y_x(t, 0) = 0, \quad (9c)$$

$$y_x(t, 1) = g(y(t, 1)) + u(t), \quad t \in (0, T) \quad (9d)$$

für ein gegebenes  $u$  aus der Menge

$$U_{\text{ad}} := \{u \in L^\infty([0, T]) \mid u_{\min}(t) \leq u(t) \leq u_{\max}(t), \text{ f.f.a. } t \in [0, T]\}$$

mit  $u_{\min}, u_{\max} \in L^\infty([0, T])$  und  $g \in C^2(\mathbb{R})$ ,  $g', g'' \in L^\infty(\mathbb{R})$ .

Das Ziel dieses Projekts ist die Implementierung des Trust-Region-Verfahrens aus der Arbeit:

#### *A Trust Region Method for Parabolic Boundary Control Problems*

von C. T. Kelley and E. W. Sachs (SIAM J. Optim., 9(4), 1064–1081) zur Lösung dieses Problems. Sie können die Arbeit unter <http://epubs.siam.org/doi/10.1137/S1052623496308965> aus dem Uni-Netz herunterladen.

### Details

1. Im Gegensatz zu den anderen Optimierungsprojekten enthält dieses Projekt keine theoretische Aufgaben. Dafür ist das Arbeitspensum an Programmieraufgaben wesentlich größer.
2. Der zu implementierende Algorithmus samt Subroutines befinden sich in der oben genannten Arbeit. Dabei finden Sie alle Details zum effizienten Programmieren des Verfahrens sowie eine sinnvolle Diskretisierung des unendlich dimensional Problems. Sie zeigen die erfolgreiche Implementierung des Verfahrens durch das Lösen des in der Arbeit angegebenen Beispiels.
3. Damit  $f(u)$ ,  $\nabla f(u)$  und  $\nabla^2 f(u)$  (oder eine Approximation) berechnet werden können, sind auch Solver für parabolischen pDG benötigt. Diese können entweder mit bestehenden Software-Paketen (wie in der Arbeit) oder mit Ihren eigenen Codes gelöst werden.
4. In Ihrer schriftlichen Zusammenfassung des Projektes sollen Sie zzgl. Ihrer numerischen Ergebnisse die Details der Arbeit und deren wichtigsten Punkte zusammenfassen.
5. Wie bei den anderen Optimierungsprojekten erfolgt die Betreuung in enger Absprache mit Prof. Surowiec.