Systemidentifikation und Regelung in der Medizin

5. Vorlesung

Prediction Error Method

Sommersemester 2020

2. Juni 2020

Thomas Schauer

Technische Universität Berlin
Fachgebiet Regelungssysteme

Literaturempfehlung: 1) E. Ikonen und K. Najim, Advanced Process Identification and Control, Marcel Dekker, Inc., 2002;
2) David T. Westwick, Robert E. Kearney, Identification of Nonlinear Physiological Systems, Wiley-IEEE Press, 2003 (Kapitel 8)

5. Parameterschätzung / Prediction Error Method (PEM)

- Bestimmung von Modellparametern aus Messdaten
- Least Squares nur anwendbar für lineare statische Abbildungen und einige dynamische Systeme (ARX) (Systemausgang ist linear abhängig von den Parametern sowie Parametervektor und Regressionsverktor sind unabhängig voneinander)
- Für nichlineare Modelle: iterative Optimierungsverfahren (Iterative Least Squares Methods)
  - wiederholte Anwendung des gesamten Datensatzes zur Parameterschätzung
    $\Rightarrow$ Batch-Adaptation

Gütefunktional basierend auf der Einschrittvorausprädiktion:

$$J(\boldsymbol{\Theta}) = \sum_{k=1}^{K} \frac{1}{2}(y(k) - \hat{y}(k, \boldsymbol{\Theta}|k-1))^2 \tag{1}$$

- Anfangswert: $\hat{\boldsymbol{\Theta}}(1)$
- In jedem Updateschritt l (Achtung: l ist nicht der Abtastindex k) Bestimmung einer Richtung $\boldsymbol{u}(l)$ und Schrittweite $\eta(l)$
- Update der Parameter mittels

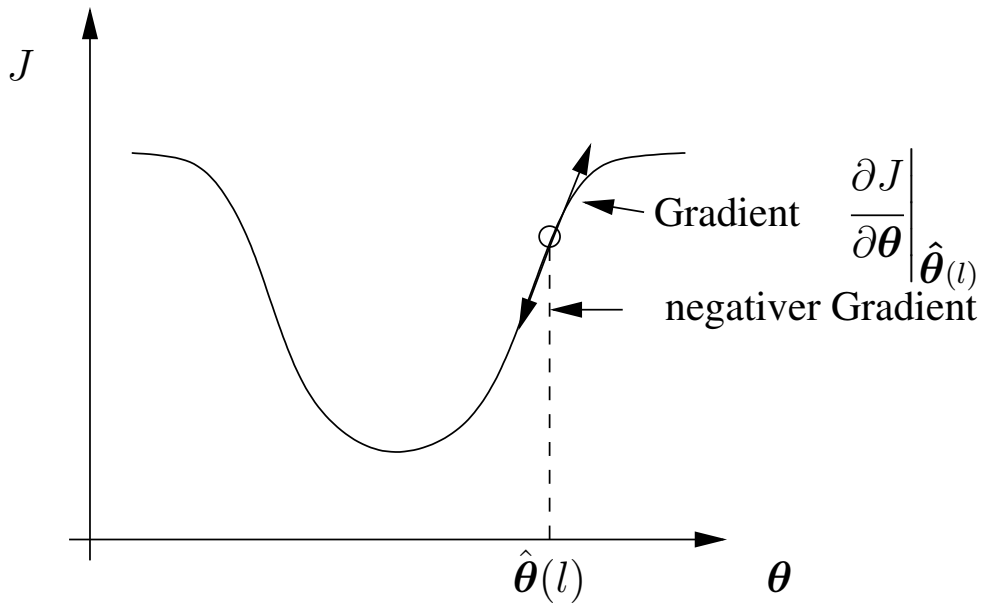
$$\hat{\boldsymbol{\Theta}}(l+1) = \hat{\boldsymbol{\Theta}}(l) + \eta(l)\boldsymbol{u}(l) \tag{2}$$

Wichtig: In $J \longrightarrow \hat{y}(k, \hat{\boldsymbol{\Theta}}(l))$ ist immer eine Funktion der aktuellen Parameterschätzung $\hat{\boldsymbol{\Theta}}(l)$
Ziel: $\boldsymbol{u}(l)$, $\eta(l)$ so wählen, dass

$$\hat{\boldsymbol{\Theta}}(l) \rightarrow \arg\min_{\boldsymbol{\Theta}} J(\boldsymbol{\Theta}) \text{ für } l \longrightarrow \infty$$

- lokale Verfahren: suche nach lokalen Minimum

5.1 Verfahren 1. Ordnung



$\hat{\boldsymbol{\Theta}}(l)$: aktuelle Schätzung des Parametervektors

Taylorentwicklung von $J(\boldsymbol{\Theta})$ bis zur 1. Ordnung:

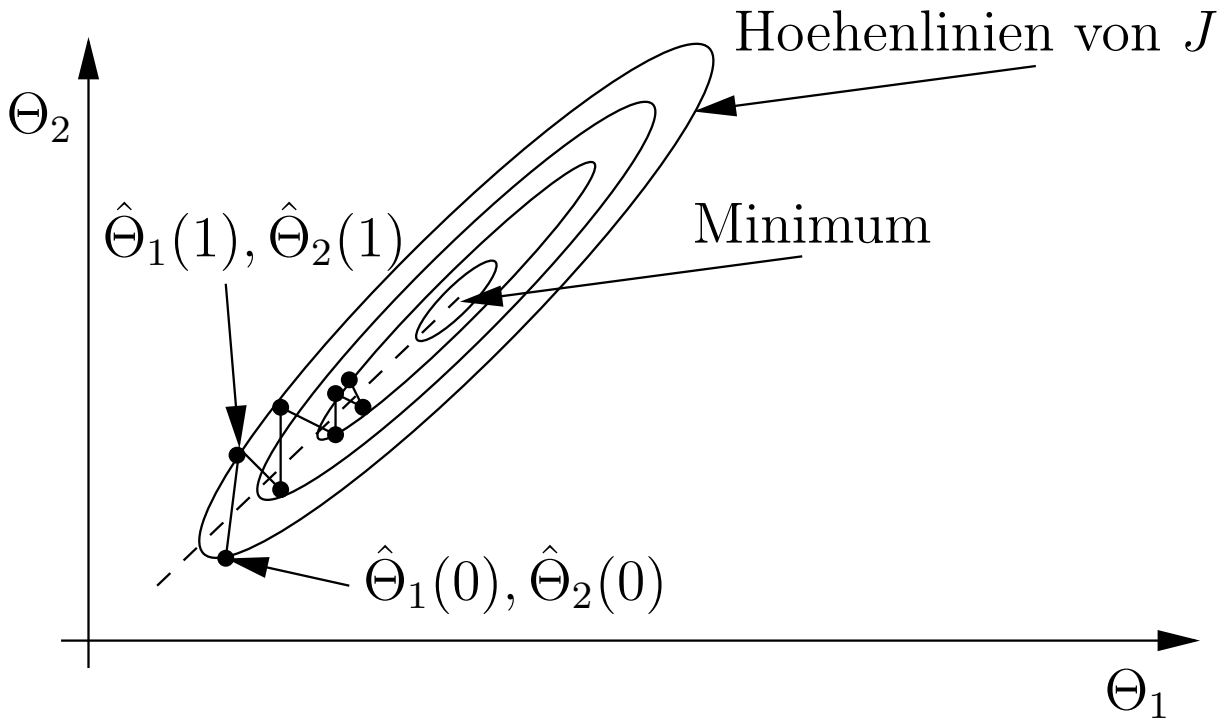
$$J(\boldsymbol{\Theta}) \simeq J(\hat{\boldsymbol{\Theta}}(l)) + \sum_{i=1}^{I} \left[\frac{\partial J}{\partial \Theta_i}\right]\Bigg|_{\boldsymbol{\Theta}=\hat{\boldsymbol{\Theta}}(l)} \tilde{\Theta}_i, \qquad \tilde{\boldsymbol{\Theta}} = \boldsymbol{\Theta} - \hat{\boldsymbol{\Theta}}(l)$$

mit

$$J(\mathbf{\Theta}) = \sum_{k=1}^{K} \frac{1}{2}(y(k) - \hat{y}(k, \mathbf{\Theta}|k-1))^2$$

$$\left[\frac{\partial J}{\partial \Theta_i}\right]\Bigg|_{\mathbf{\Theta}=\hat{\mathbf{\Theta}}(l)} = -\sum_{k=1}^{K}[y(k) - \hat{y}(k, \hat{\mathbf{\Theta}}(l))]\frac{\partial \hat{y}(k)}{\partial \Theta_i}\Bigg|_{\mathbf{\Theta}=\hat{\mathbf{\Theta}}(l)}$$

$$\hat{\mathbf{\Theta}}(l+1) = \hat{\mathbf{\Theta}}(l) - \eta \left[\frac{\partial J}{\partial \mathbf{\Theta}}\right]\Bigg|_{\mathbf{\Theta}=\hat{\mathbf{\Theta}}(l)}$$



Stellt das Gütefunktional ein langgestrecktes Tal dar, so kann es bei einem Optimierungsverfahren 1. Ordnung zu Oszillationen kommen, der denen die Lösung immer von einer Seite des Tales zur anderen Seite des Tales springt.

Verfahren 1. Ordnung - Eigenschaften:

- Verfahren 1. Ordnung (wie „steilster Abstieg“) sind einfach zu realisieren
- schlechte (langsame) Konvergenz
- Siehe Beispiel: langes Tal
  - Oszillationen
  - kleine Schrittweite damit man das Tal nicht verlässt
  - Richtung immer orthogonal zu den Höhenlinien

5.2 Verfahren 2. Ordnung

- Ausnutzung der lokalen Krümmung der Gütefunktionals (Fläche)
- Beispiel langes Tal:
  - Krümmung am größten orthogonal zum Tal
  - Krümmung am kleinsten entlang des Tals

    $\longrightarrow$ optimale Suchrichtung
- Annahme: lokales Aussehen des Gütefunktionals entspricht einer quadratischen Funktion der Parameter

Taylorreihenentwicklung

$$
\begin{aligned}
J(\boldsymbol{\Theta}) &= J(\hat{\boldsymbol{\Theta}}(l)) + \sum_{i=1}^{I} \left.\frac{\partial J}{\partial \Theta_i}\right|_{\boldsymbol{\Theta}=\hat{\boldsymbol{\Theta}}(l)} \tilde{\Theta}_p + \sum_{i=1,i^*=1}^{I} \left.\frac{\partial^2 J}{\partial \Theta_i \partial \Theta_{i^*}}\right|_{\boldsymbol{\Theta}=\hat{\boldsymbol{\Theta}}(l)} \tilde{\Theta}_i \tilde{\Theta}_{i^*} + \cdots \\
&\simeq J(\hat{\boldsymbol{\Theta}}(l)) - \boldsymbol{b}^T \tilde{\boldsymbol{\Theta}} + \frac{1}{2} \tilde{\boldsymbol{\Theta}}^T \boldsymbol{H}^T \tilde{\boldsymbol{\Theta}}, \quad \tilde{\boldsymbol{\Theta}} = \boldsymbol{\Theta} - \hat{\boldsymbol{\Theta}}(l)
\end{aligned}
$$

mit $\boldsymbol{b} = -\left.\dfrac{\partial J}{\partial \boldsymbol{\Theta}}\right|_{\boldsymbol{\Theta}=\hat{\boldsymbol{\Theta}}(l)}$ (negativer Gradient)

$\boldsymbol{H}$ - Hession Matrix (Hessische Matrix):

$$
\left.\frac{\partial^2 J}{\partial \Theta_p \partial \Theta_{i^*}}\right|_{\boldsymbol{\Theta}=\hat{\boldsymbol{\Theta}}(l)} = h_{i,i^*}
$$

Nullsetzen der Ableitung von J:

$$\frac{\partial J}{\partial \tilde{\boldsymbol{\Theta}}} = -\boldsymbol{b}^T + \tilde{\boldsymbol{\Theta}}^T \boldsymbol{H}^T = 0 \quad \Rightarrow \quad \tilde{\boldsymbol{\Theta}} = \boldsymbol{H}^{-1}\boldsymbol{b}$$

- Wahre Funktion ist nicht quadratisch; daher verwendet man $\boldsymbol{H}^{-1}\boldsymbol{b}$ als Richtung (meistens mit Schrittweite versehen) und macht dann iterativ weiter.

5.2.1 Newton Verfahren

$$\hat{\boldsymbol{\Theta}}(l+1) = \hat{\boldsymbol{\Theta}}(l) + \eta \boldsymbol{H}^{-1}(\hat{\boldsymbol{\Theta}}(l))\boldsymbol{b}(\hat{\boldsymbol{\Theta}}(l))$$

η: Lernrate

- erfolgreicher Schritt: $J(\hat{\boldsymbol{\Theta}}(l+1)) < J(\hat{\boldsymbol{\Theta}}(l)) \rightarrow \eta$ vergrössern ($\approx$ Faktor 10)
- nicht erfolgreicher Schritt: $J(\hat{\boldsymbol{\Theta}}(l+1)) \geq J(\hat{\boldsymbol{\Theta}}(l)) \rightarrow$ letzten Schritt verwerfen und η verkleinern ($\approx$ Faktor 10)

Problem: Berechnen von $\boldsymbol{H}^{-1}(\hat{\boldsymbol{\Theta}}(l))$

5.2.2 Approximative Verfahren

K - Anzahl der Messungen, I - Anzahl der Parameter, $e(k)$ Prädiktionsfehler für Abtastzeitpunkt k

$$e(k, \hat{\boldsymbol{\Theta}}(l)) = y(k) - \hat{y}(k, \hat{\boldsymbol{\Theta}}(l)), \quad \boldsymbol{E}(\hat{\Theta}(l)) = \begin{bmatrix} e(1, \hat{\boldsymbol{\Theta}}(l)) & \cdots & e(K, \hat{\boldsymbol{\Theta}}(l)) \end{bmatrix}^T$$

Gütefunktional

$$J(\hat{\boldsymbol{\Theta}}(l)) = \frac{1}{2} \boldsymbol{E}(\hat{\boldsymbol{\Theta}}(l))^T \boldsymbol{E}(\hat{\boldsymbol{\Theta}}(l))$$

$$\frac{\partial J(\hat{\boldsymbol{\Theta}}(l))}{\partial \hat{\boldsymbol{\Theta}}(l)} = \sum_{k=1}^{K} e(k, \hat{\boldsymbol{\Theta}}(l)) \frac{\partial e(k, \hat{\boldsymbol{\Theta}}(l))}{\partial \hat{\boldsymbol{\Theta}}(l)} = \boldsymbol{G}(\hat{\boldsymbol{\Theta}}(l))^T \boldsymbol{E}(\hat{\boldsymbol{\Theta}}(l))$$

$\boldsymbol{G}(\hat{\boldsymbol{\Theta}}(l)) \in \mathbb{R}^{K \times I}$: Jakobimatrix im Iterationsschritt l

$$g_{k,i} = \frac{\partial e(k, \hat{\boldsymbol{\Theta}}(l))}{\partial \hat{y}(k, \hat{\boldsymbol{\Theta}}(l))} \frac{\partial \hat{y}(k, \hat{\boldsymbol{\Theta}}(l))}{\partial \hat{\Theta}_i(l)} = -\frac{\partial \hat{y}(k, \hat{\boldsymbol{\Theta}}(l))}{\partial \hat{\Theta}_i(l)}$$

$$\boldsymbol{G}(\hat{\boldsymbol{\Theta}}(l)) = \begin{bmatrix} -\frac{\partial \hat{y}(1,\hat{\boldsymbol{\Theta}}(l))}{\partial \hat{\Theta}_1(l)} & \cdots & -\frac{\partial \hat{y}(1,\hat{\boldsymbol{\Theta}}(l))}{\partial \hat{\Theta}_I(l)} \\ \vdots & \ddots & \vdots \\ -\frac{\partial \hat{y}(K,\hat{\boldsymbol{\Theta}}(l))}{\partial \hat{\Theta}_1(l)} & \cdots & -\frac{\partial \hat{y}(K,\hat{\boldsymbol{\Theta}}(l))}{\partial \hat{\Theta}_I(l)} \end{bmatrix}$$

$$\begin{aligned} \frac{\partial^2 J((\hat{\boldsymbol{\Theta}}(l)))}{\partial(\hat{\boldsymbol{\Theta}}(l))^2} &= \sum_{k=1}^{K} \left[\frac{\partial e(k,\hat{\boldsymbol{\Theta}}(l))}{\partial \hat{\boldsymbol{\Theta}}(l)} \left[\frac{\partial e(k,\hat{\boldsymbol{\Theta}}(l))}{\partial \hat{\boldsymbol{\Theta}}(l)} \right]^T + e(k,\hat{\boldsymbol{\Theta}}(l)) \frac{\partial^2 e(k,\hat{\boldsymbol{\Theta}}(l))}{\partial(\hat{\boldsymbol{\Theta}}(l))^2} \right] \\ &= \boldsymbol{G}(\hat{\boldsymbol{\Theta}}(l))^T \boldsymbol{G}(\hat{\boldsymbol{\Theta}}(l)) + \boldsymbol{S}(\hat{\boldsymbol{\Theta}}(l)) = \boldsymbol{H}(\hat{\boldsymbol{\Theta}}(l)) \end{aligned}$$

- $\boldsymbol{G}(\hat{\boldsymbol{\Theta}}(l))^T \boldsymbol{G}(\hat{\boldsymbol{\Theta}}(l))$: leicht zu bestimmen, da nur erste Ableitungen vorkommen
- $\boldsymbol{S}(\hat{\boldsymbol{\Theta}}(l))$: schwer zu berechnen, da zweite Ableitungen vorkommen

5.2.2.1 Gauß-Newton-Verfahren

$$\hat{\boldsymbol{\Theta}}(l+1) = \hat{\boldsymbol{\Theta}}(l) - \eta \underbrace{\left[\boldsymbol{G}(\hat{\boldsymbol{\Theta}}(l))^T \boldsymbol{G}(\hat{\boldsymbol{\Theta}}(l))\right]^{-1}}_{\hat{\boldsymbol{H}}^{-1}(\hat{\boldsymbol{\Theta}}(l))} \underbrace{\boldsymbol{G}(\hat{\boldsymbol{\Theta}}(l))^T \boldsymbol{E}(\hat{\boldsymbol{\Theta}}(l))}_{-\boldsymbol{b}(\hat{\boldsymbol{\Theta}}(l))}$$

η : Schrittweite $\rightarrow$ anpassen ähnlich zum Newton-Verfahren

5.2.2.2 Levenberg-Marquardt-Verfahren

$$\hat{\boldsymbol{\Theta}}(l+1) = \hat{\boldsymbol{\Theta}}(l) - \underbrace{\left[\boldsymbol{G}(\hat{\boldsymbol{\Theta}}(l))^T \boldsymbol{G}(\hat{\boldsymbol{\Theta}}(l)) + \eta \boldsymbol{I}\right]^{-1}}_{\hat{\boldsymbol{H}}^{-1}(\hat{\boldsymbol{\Theta}}(l))} \underbrace{\boldsymbol{G}(\hat{\boldsymbol{\Theta}}(l))^T \boldsymbol{E}(\hat{\boldsymbol{\Theta}}(l))}_{-\boldsymbol{b}(\hat{\boldsymbol{\Theta}}(l))}$$

- $\eta(l)$ erhöhen für schlechten Schritt: $\eta(l) = \eta(l)\gamma, \quad \gamma > 1$

  und Verwerfen des letzten Schritts!
- $\eta(l)$ erniedrigen für erfolgreichen Schritt: $\eta(l+1) = \eta(l)/\gamma, \gamma > 1, l = l+1$
- $\eta(l)$ sehr groß: $\hat{\boldsymbol{\Theta}}(l+1) \approx \hat{\boldsymbol{\Theta}}(l) - \frac{1}{\eta}\boldsymbol{G}(\hat{\boldsymbol{\Theta}}(l))^T \boldsymbol{E}(\hat{\boldsymbol{\Theta}}(l))$ (steilster Abstieg)

  $\longrightarrow$ Verfahren erster Ordnung
- $\eta(l)$ sehr klein: $\hat{\boldsymbol{\Theta}}(l+1) \approx \hat{\boldsymbol{\Theta}}(l) - \left[\boldsymbol{G}(\hat{\boldsymbol{\Theta}}(l))^T \boldsymbol{G}(\hat{\boldsymbol{\Theta}}(l))\right]^{-1} \boldsymbol{G}(\hat{\boldsymbol{\Theta}}(l))^T \boldsymbol{E}(\hat{\boldsymbol{\Theta}}(l))$

  $\longrightarrow$ Gauß-Newton-Verfahren

5.3 Gradientenberechnung für die PEM

$$J(\hat{\boldsymbol{\Theta}}(l)) = \frac{1}{2}\sum_{k=1}^{K}\Big(y(k) - \hat{y}(k, \hat{\boldsymbol{\Theta}}(l)|k-1)\Big)^2$$

$$\frac{\partial J(\hat{\boldsymbol{\Theta}}(l))}{\partial \hat{\Theta}_i(l)} = -\sum_{k=1}^{K}\left([y(k) - \hat{y}(k, \hat{\boldsymbol{\Theta}}(l)|k-1)]\frac{\partial \hat{y}(k, \hat{\boldsymbol{\Theta}}(l)))}{\partial \hat{\Theta}_i(l)}\right)$$

Gesucht: $\dfrac{\partial \hat{y}(k, \hat{\boldsymbol{\Theta}}(l))}{\partial \hat{\Theta}_i(l)} \longrightarrow$ für lineare dynamische Systeme (OE,ARMAX,BJ)

5.3.1 Beispielhafte Gradientenberechnung für ein OE-Modell

Parametervektor:

$$\boldsymbol{\Theta} = \begin{bmatrix} a_1 & a_2 & \cdots & a_{n_A} & b_0 & b_1 & b_2 & \cdots & b_{n_B} \end{bmatrix}^T \tag{3}$$

Prädiktor:

$$\hat{y}(k, \hat{\boldsymbol{\Theta}}(l)|k-1) = \hat{\mathsf{B}}(q^{-1}, l)u(k-d) - \hat{\mathsf{A}}_1(q^{-1}, l)\hat{y}(k-1|k-2)$$

Um die Gleichungen kompakter zu halten, wird $|\ldots$ im folgenden weggelassen.

Berechnung der Gradienten:

$$\frac{\partial \hat{y}(k|k-1)}{\partial \hat{a}_n(l)} = \underbrace{\frac{\partial}{\partial \hat{a}_n(l)} \left(\hat{\mathsf{B}}(q^{-1}, l)u(k-d) \right)}_{0} - \frac{\partial \hat{\mathsf{A}}_1(q^{-1}, l)\hat{y}(k-1)}{\partial \hat{a}_n(l)} \tag{4}$$

$$\begin{aligned} \frac{\partial \hat{y}(k)}{\partial \hat{a}_n(l)} &= -\hat{a}_1(l)\frac{\partial \hat{y}(k-1)}{\partial \hat{a}_n(l)} \cdots - \frac{\partial(\hat{a}_n(l)\hat{y}(k-n))}{\partial \hat{a}_n(l)} \cdots - \hat{a}_{n_A}(l)\frac{\partial \hat{y}(k-n_A)}{\partial \hat{a}_n(l)} \\ \frac{\partial \hat{a}_n(l)\hat{y}(k-n)}{\partial \hat{a}_n(l)} &= \hat{y}(k-n) + \hat{a}_n(l)\frac{\partial \hat{y}(k-n)}{\partial \hat{a}_n(l)} \end{aligned} \tag{5}$$

$$\frac{\partial \hat{y}(k)}{\partial \hat{a}_n(l)} = -\hat{y}(k-n) - \sum_{m=1}^{n_A} \hat{a}_m(l) \frac{\partial \hat{y}(k-m)}{\partial \hat{a}_n(l)}$$

$$\frac{\partial \hat{y}(k)}{\partial \hat{b}_n(l)} = \underbrace{\frac{\partial(\hat{\mathsf{B}}(q^{-1},l)u(k-d))}{\partial \hat{b}_n(l)}}_{*} - \underbrace{\frac{\partial(\hat{\mathsf{A}}_1(q^{-1},l)\hat{y}(k-1))}{\partial \hat{b}_n(l)}}_{**}$$

$$* = u(k-d-n)$$

$$** = \sum_{m=1}^{n_A} \hat{a}_m(l) \frac{\partial \hat{y}(k-m)}{\partial \hat{b}_n(l)}$$

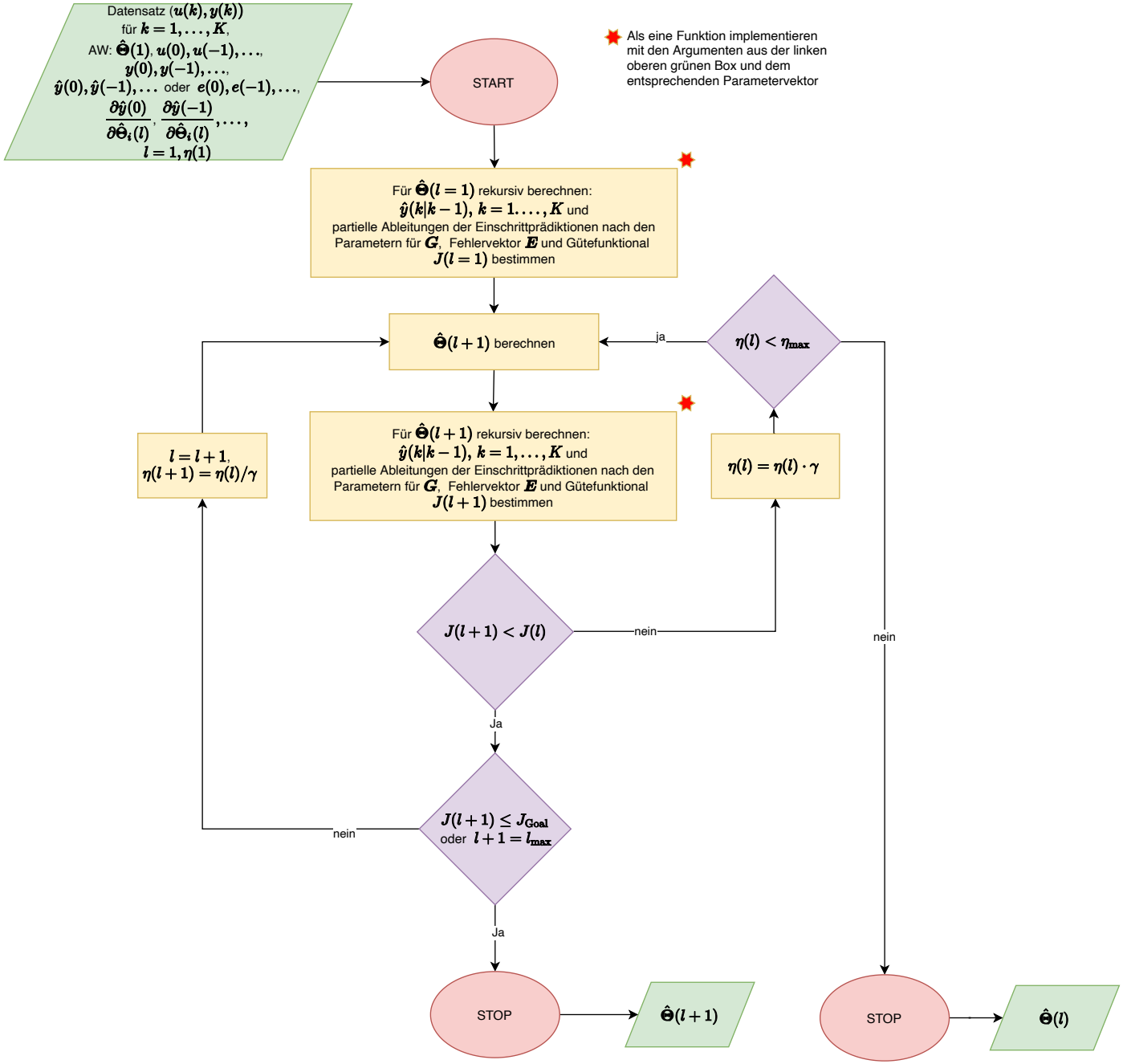
$$\frac{\partial \hat{y}(k)}{\partial \hat{b}_n(l)} = u(k-d-n) - \sum_{m=1}^{n_A} \hat{a}_m(l) \frac{\partial \hat{y}(k-m)}{\partial \hat{b}_n(l)}$$

Rekursiver Algorithmus:

$$\frac{\partial \hat{y}(k)}{\partial \hat{a}_n(l)} \Rightarrow \Psi_n^a(k), \quad \frac{\partial \hat{y}(k)}{\partial \hat{b}_n(l)} \Rightarrow \Psi_n^b(k)$$

$$\Psi_n^b(k) = u(k-d-n) - \sum_{m=1}^{n_A} a_m \Psi_n^b(k-m), \quad k = 1 \cdots K$$

$$\Psi_n^a(k) = -\hat{y}(k-n) - \sum_{m=1}^{n_A} a_m \Psi_n^a(k-m), \quad k = 1 \cdots K$$


Datensatz $(u(k), y(k))$
für $k = 1, \ldots, K$,
AW: $\hat{\Theta}(1), u(0), u(-1), \ldots,$
$y(0), y(-1), \ldots,$
$\hat{y}(0), \hat{y}(-1), \ldots$ oder $e(0), e(-1), \ldots,$
$\frac{\partial \hat{y}(0)}{\partial \hat{\Theta}_i(l)}, \frac{\partial \hat{y}(-1)}{\partial \hat{\Theta}_i(l)}, \ldots,$
$l = 1, \eta(1)$
START
Als eine Funktion implementieren mit den Argumenten aus der linken oberen grünen Box und dem entsprechenden Parametervektor
Für $\hat{\Theta}(l=1)$ rekursiv berechnen:
$\hat{y}(k|k-1), k = 1, \ldots, K$ und
partielle Ableitungen der Einschrittprädiktionen nach den Parametern für G, Fehlervektor E und Gütefunktional $J(l=1)$ bestimmen
$\hat{\Theta}(l+1)$ berechnen
ja
$\eta(l) < \eta_{max}$
Für $\hat{\Theta}(l+1)$ rekursiv berechnen:
$\hat{y}(k|k-1), k = 1, \ldots, K$ und
partielle Ableitungen der Einschrittprädiktionen nach den Parametern für G, Fehlervektor E und Gütefunktional $J(l+1)$ bestimmen
$l = l+1,$
$\eta(l+1) = \eta(l)/\gamma$
$\eta(l) = \eta(l) \cdot \gamma$
$J(l+1) < J(l)$
nein
Ja
$J(l+1) \leq J_{Goal}$
oder $l+1 = l_{max}$
nein
Ja
nein
STOP
$\hat{\Theta}(l+1)$
STOP
$\hat{\Theta}(l)$

5.4 Batch versus Sample Adaptation (Rekursive Prädiktionsfehlermethode)

Normalerweise wird ein ganzer Datensatz (Batch) mit vielen Messungen für die iterativen Parameter-Updates verwendet. Alle Daten fließen in das Gütefunktional und dessen Ableitungen ein. Die allgemeine Update-Gleichung lautet

$$\hat{\boldsymbol{\Theta}}_{l+1} = \boldsymbol{f}\left(\hat{\boldsymbol{\Theta}}_j, J(\hat{\boldsymbol{\Theta}}_j), \frac{\partial J(\hat{\boldsymbol{\Theta}}_j)}{\partial \hat{\boldsymbol{\Theta}}_j}, \ldots\right) \quad \text{mit } j = l, l-1, \ldots,$$

wobei in den meisten Fällen nur $j = l$ verwendet wird:

$$\hat{\boldsymbol{\Theta}}_{l+1} = \boldsymbol{f}\left(\hat{\boldsymbol{\Theta}}_l, J(\hat{\boldsymbol{\Theta}}_l), \frac{\partial J(\hat{\boldsymbol{\Theta}}_l)}{\partial \hat{\boldsymbol{\Theta}}_l}, \ldots\right).$$

Einen Sonderfall erhält man, wenn man den Datensatz auf eine Messung reduziert und $l = k$ und $J = J(k) = \frac{1}{2}e(k)^2$ ansetzt. Man spricht dann von der sogenannten *Sample Adaptation* oder *Rekursiver PEM*.

$$\hat{\boldsymbol{\Theta}}_{k+1} = \boldsymbol{f}\left(\hat{\boldsymbol{\Theta}}_k, \frac{1}{2}e(k, \hat{\boldsymbol{\Theta}}_k)^2, \frac{\partial \frac{1}{2}e(k, \hat{\boldsymbol{\Theta}}_k)^2}{\partial \hat{\boldsymbol{\Theta}}_k}, \ldots\right)$$

- Es liegt die Versuchung nahe, ein solches Verfahren für die Onlineschätzung von Parametern zu verwenden. Jedoch muss bedacht werden, dass es sich hier um ein iteratives Verfahren und nicht um ein rekursives Verfahren wie RLS handelt.
- RLS liefert das gleiche Ergebnis wie LS
- Sample Adaptation liefert nicht das gleiche Ergebnis wie ein Batch Adaptation-Schritt der gleichen Daten, da in jedem Schritt der Sample Adaptation ein anderes Gütefunktional optimiert wird.
- Auch ist die Konvergenz bei der Sample Adaptation extrem langsam, da über alle Messdaten nur ein Iterationsschritt im Sinne des Batchverfahrens näherungsweise durchgeführt wird.