



Hochschule für Angewandte Wissenschaften Hamburg  
*Hamburg University of Applied Sciences*

**Ausarbeitung Seminar**  
Kalman Filter zur Rekonstruktion  
von Messsignalen  
Denis Schetler

*Fakultät Technik und Informatik  
Department Informatik  
Betreuer: Prof. Dr.-Ing. Bernd Schwarz  
Datum: 15. Februar 2007*

*Faculty of Engineering and Computer Science  
Department of Computer Science*

# Denis Schetler

## Thema der Seminararbeit

Kalman Filter zur Rekonstruktion von Messsignalen

## Stichworte

Kalman Filter, FAUST, Messsignale, Sensordatenfusion, Objektverfolgung, Zustandsbeobachter, Luenberger, Regelschleife, Simulation, Matlab, Wurfparabel

## Kurzzusammenfassung

Diese Seminararbeit stellt das Kalman Filter zur Rekonstruktion von Messsignalen vor. Das Filter wird oft in Echtzeitsystemen zur Messsignalverarbeitung und Bildverarbeitung eingesetzt. Grund hierfür ist die rekursive Arbeitsweise in einer Regelschleife und eine optimale Schätzung der Systemzustände. Für die Anwendung des Kalman Filters wird die System- und Messgleichung benötigt und für jeden Rekursionsschritt ein aktueller Messvektor. Das Prinzip und die Arbeitsweise des Kalman Filters wird in dieser Ausarbeitung vorgestellt.

Es wird gezeigt, warum eine Sensordatenfusion bei Fahrerassistenzsystemen sinnvoll ist und in welchen Bereichen das Kalman Filter eingesetzt wird. Der Grundgedanke des Zustandsbeobachters nach Luenberger wird vorgestellt. Anschließend wird die Querverbindung zwischen dem Luenberger Beobachter und dem Kalman Filter aufgezeigt. Diese Betrachtungen werden zu Beginn im zeitkontinuierlichen Bereich vorgenommen, bevor in den zeitdiskreten Bereich übergegangen wird, damit das Filter auf den  $\mu C$ -Plattformen des FAUST Projekts einsetzbar ist. Dazu werden zunächst die System- und Messgleichung diskretisiert. Die Berechnungsformeln der Regelschleife des diskreten Kalman Filters werden vorgestellt und erklärt.

Ein Anwendungsbeispiel wird anhand einer Wurfparabel gegeben. Aus dem physikalischen System werden die diskreten Systemgleichungen entwickelt. Die Gleichungen und Lösungen für den ersten Berechnungsschritt des Kalman Filters für die Wurfparabel sind angegeben. Für die komplette Wurfparabel ist eine Matlab Simulation realisiert, deren Ergebnis präsentiert wird. Anschließend wird ein kurzer Exkurs in die Objektverfolgung am Beispiel der Wurfparabel gemacht. Dort wird die Reaktion des Filters bei einer kurzen Verdeckung simuliert und dargestellt, wie mit einem Suchfenster für die Messwerte das Kalman Filter im Prädiktionsmodus arbeitet.

Es wird ein Ausblick in die Industrie gewährt, für die die Thematik dieser Seminararbeit relevant ist. Abschließend wird das Thema für die Masterarbeit umrissen. In dieser muss zunächst das bestehende Fahrzeugmodell des FAUSTs analysiert und erweitert werden. Anschließend soll der Moving-Average-Filter bei der Messsignalverarbeitung der Laserscannerabstandswerte im FAUST Projekt durch das Kalman Filter ersetzt werden, um die Filterverzögerung zu beseitigen und um nicht mehrere Messwerte speichern zu müssen. Im weiteren Verlauf soll eine Objektverfolgung realisiert werden. Diese dient zum Einen, um andere Objekte zu verfolgen und somit Fahrerassistenzsysteme mit Informationen zu versorgen, als auch um bekannte Objekte im Raum zu verfolgen und anhand dieser eine Positionsbestimmung im Raum zu realisieren.

## Inhaltsverzeichnis

<b>Abbildungsverzeichnis</b>	<b>3</b>
<b>1 Einleitung</b>	<b>4</b>
<b>2 Sensordatenfusion für die Umgebungserkennung</b>	<b>4</b>
<b>3 Anwendungsbereiche des Kalman Filters</b>	<b>5</b>
<b>4 Einführung in das Kalman Filter</b>	<b>5</b>
<b>5 Zustandsbeobachter nach Luenberger</b>	<b>6</b>
<b>6 Querverbindung zum Kalman Filter</b>	<b>7</b>
<b>7 Diskrete System- und Messgleichung</b>	<b>7</b>
<b>8 Regelschleife des Kalman Filters</b>	<b>8</b>
<b>9 Simulationsbeispiel Ideale Wurfparabel</b>	<b>9</b>
9.1 Beispielrechnung für Schritt 0 . . . . .	10
9.2 Matlab Simulation . . . . .	11
<b>10 Objektverfolgung</b>	<b>11</b>
<b>11 Zusammenfassung und Ausblick</b>	<b>12</b>
11.1 Motivation aus der Industrie . . . . .	13
11.2 Thema für die Masterarbeit . . . . .	13
<b>Literatur</b>	<b>14</b>
<b>A Beispielrechnung für Schritt 0</b>	<b>15</b>
<b>B Matlab Simulationscode</b>	<b>16</b>

## Abbildungsverzeichnis

1 Rampenantwort des MA-Filters für sich nähernde Fahrzeuge . . . . .	4
2 Übersicht des Kalman Filters . . . . .	6
3 Luenberger Beobachter . . . . .	7
4 Regelschleife des Kalman Filters . . . . .	8
5 Ideale Wurfparabel . . . . .	9
6 Trapezintegration . . . . .	10
7 Rekonstruierte Wurfparabel . . . . .	11
8 Objektverfolgung Fehler . . . . .	12
9 Objektverfolgung OK . . . . .	12

## 1 Einleitung

In dieser Seminararbeit wird das Kalman Filter zur Rekonstruktion von gestörten Messsignalen vorgestellt. Bei Sensoren in technischen Systemen, wie dem Fahrzeug im Projekt Fahrerlose Autonome Transportsysteme (FAUST), treten immer Messfehler auf, z.B. bei der Abstandsmessung des Laserscanners. Zur Unterdrückung dieser Fehler im System werden Filter eingesetzt. Das Kalman Filter liefert optimale Schätzwerte für die Systemzustandsgrößen und somit indirekt die optimalen Schätzwerte für die Messsignale. Es wird außerdem zur Objektverfolgung und Ortsbestimmung eingesetzt. Im Masterprojekt FAUST der Technischen Informatik an der Hochschule für Angewandte Wissenschaften Hamburg sollen aktuelle Forschungsthemen der Industrie umgesetzt werden. Im Automobilbereich sind dieses unter anderem Assistenzsysteme zur Unfallvermeidung. Zu nennen sind z.B. Brems-, Ausweich-, Spur halten und Kreuzungsassistent, vgl. [1], [6], [3]. Zur Realisierung dieser Systeme sind Sensoren zur Umgebungserkennung notwendig, wobei die folgenden Sensorarten alleine oder in Kombination zur Anwendung kommen: Ultraschallsensoren, Kameras und Laserscanner. Jeder dieser Sensortypen hat seine Vor- und Nachteile, eine Sensordatenfusion soll die Umgebungserkennung und Informationsauswertung möglichst robust, zuverlässig und echtzeitfähig machen. Das Ziel im FAUST Projekt ist es, die Laserscanner- und Kameradaten zu kombinieren. Die Rekonstruktion der gestörten Messsignale mit Hilfe des Kalman Filters ist dazu ein notwendiger Vorverarbeitungsschritt, denn erst die gefilterten Werte eignen sich für die Sensordatenfusion. In der Masterarbeit [1] wurde der Moving-Average-Filter verwendet. Dieser Filter hat allerdings den Nachteil, dass der Mittelwert über mehrere Messwerte gebildet wird, d. h. es müssen immer mehrere Messwerte im Speicher vorgehalten werden. Zudem hat das Filter eine zeitliche Verzögerung, vgl. Abb. 1. Nach [1] ist bei sich nähernden Fahrzeugen der Filterausgangswert  $y$  größer als die aktuell gemessene Entfernung des Laserscanners  $u$ . Die angegebene zeitliche Differenz von Eingangs- und Ausgangswert ist  $\Delta t = 225ms$ , d. h. die maximale Differenz der Entfernung ist  $\Delta sm = 63cm$ . Für die Realisierung der Vorverarbeitung der Sensordatenfusion und der Verbesserung der bisher implementierten Assistenzsystemem durch das Kalman Filter, anstatt des Moving-Average-Filters, wird in dieser Seminararbeit die Funktionsweise des Kalman Filters vorgestellt.

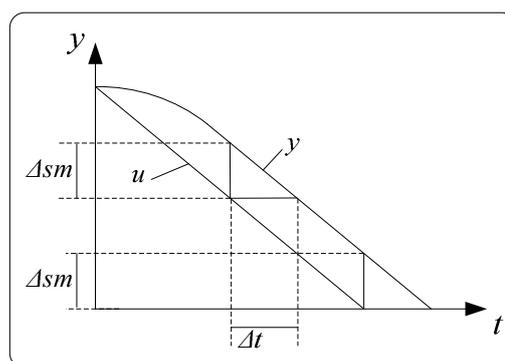


Abb. 1: Rampenantwort des MA-Filters für sich nähernde Fahrzeuge

## 2 Sensordatenfusion für die Umgebungserkennung

Für eine robuste Objektverfolgung und Umgebungswahrnehmung reicht ein Sensordatentyp nicht aus. Hierzu ist eine Sensordatenfusion von verschiedenen Sensortypen notwendig, womit die Vorteile eines jeweiligen Sensortyps genutzt werden.

Eine Kamera bietet alle benötigten Informationen der Umgebung des Fahrzeugs, schließlich nimmt der menschliche Fahrer seine Umgebung mit den Augen wahr. Allerdings ist die Bildverarbeitung aufwendig, um die benötigten Informationen, wie Fahrbahnmarkierungen und die anderen Fahrzeuge, aus den Rohdaten zu extrahieren. Außerdem besitzt eine Kamera zumeist nur einen Öffnungswinkel von  $30^\circ$  bis  $40^\circ$ , der aber mit einer entsprechenden Optik vergrößert werden kann. Veränderungen bei den Witterungs- und Lichtverhältnissen, wie z.B. bei Tunnel

Ein- und Ausfahrten, stellen bei der Bildverarbeitung besondere Herausforderungen dar. Zudem nimmt die Abstandsbestimmung mit Hilfe der Kamera mit dem Quadrat der Entfernung ab [6]. Der Laserscanner liefert eine Abstandsgenauigkeit bis zu 1cm und ist somit genauer als eine Kamera. Er liefert einen horizontalen Schnitt der Fahrbahn und liefert als Rohdaten gemessene Entfernungspunkte. Es ist möglich, die Umgebung bis zu einem Bereich von  $360^\circ$  zu scannen. Neuere Laserscanner, wie der ALASCA XT [5], liefern auch bis zu vier Messpunkte für einen Winkel, so wird z.B. von der Windschutzscheibe eines Autos ein Teil des Laserlichts reflektiert und gemessen, ein Teil geht aber durch die Scheibe hindurch und wird im Fahrzeuginneren reflektiert, so gibt es für einen Winkel zwei Messpunkte. Zudem bietet der Laserscanner auch die Möglichkeit in vertikaler Richtung bis zu einem Winkel von  $3,2^\circ$  zu scannen. Ein genereller Nachteil des Laserscanners ist, dass Straßenmarkierungen und Schilder nicht erkannt werden. Zudem ist ein Fußgänger und ein Ampelpfahl nur im zeitlichen Verlauf mit mehreren Messungen unterscheidbar. Denn der Fußgänger bewegt sich, die Ampel nicht, dies ist aber nicht mit einem Scan feststellbar.

### 3 Anwendungsbereiche des Kalman Filters

Im Folgenden sind einige Themengebiete aufgelistet, in denen das Kalman Filter zur Anwendung kommt. Für das FAUST Projekt sind vor allem die Punkte Messsignalverarbeitung und Bildverarbeitung interessant, wobei sich diese Seminararbeit auf den ersten Punkt konzentriert.

- Messsignalverarbeitung
  - Zur Unterdrückung von System- und Messrauschen.
  - Zur Schätzung des „wahren“ Systemzustands, der nicht direkt messbar ist.
  - Zur Objektverfolgung anhand von Messsignalen des Laserscanners.
- Bildverarbeitung
  - Zur Objektverfolgung anhand von Kanten im Bild.
  - Zur Positionsbestimmung anhand von Landmarken.
  - Für Fahrer-Assistenz-Systeme, wie z. B. Fahrbahnverfolgung.
- Bilddatencodierung
  - Um einen höheren Datenkompressionsfaktor bei MPEG4, durch Segmentierung der bewegten Objekte und deren präzisen Bewegungsschätzung, zu erreichen.

### 4 Einführung in das Kalman Filter

Das Kalman Filter liefert optimale Schätzwerte für die Systemzustandsgröße  $x$ . Nach [6] beinhaltet der Zustandsvektor bei dem Objektmodell die Werte für die Position  $x_0, y_0$ , die Geschwindigkeit  $v_x, v_y$  und die Ausdehnung  $b_0, l_0$ . Da dieser Zustandsvektor zu Beginn nicht bekannt ist, wird dieser geschätzt, vgl.  $\hat{x}_0^-$  in Abb. 2. Für jeden Rekursionsschritt des Kalman Filters muss eine aktuelle Laserscannermessung in  $y_k$  für die Berechnung der Messfehlerkorrektur anliegen. Die Berechnung ergibt aus der „a priori“ Schätzung für  $\hat{x}_k^-$  und der aktuellen Messung  $y_k$  die „a posteriori“ Schätzung  $\hat{x}_k^+$ , d. h. die Schätzung der Systemzustände ist mit der aktuellen Messung korrigiert. Der Prädiktion, die die zeitliche Entwicklung des Systems berechnet, wird zusätzlich

die „a posteriori“ Schätzfehlerkovarianz  $P_k^+$  übergeben. Diese Berechnet sich aus dem Erwartungswert der Multiplikation der Differenz von „wahrem“ und geschätzten Systemzustand  $e_k^+$  und deren transponierten Matrix. Mit  $P$  wird der erwartete Fehler für die Abweichung der Systemzustände ausgedrückt, vgl. Gl. 1 bis 4.

$$e_k^+ = x_k - \hat{x}_k^+ \quad (1) \quad P_k^+ = E \left[ e_k^+ \cdot e_k^{+T} \right] \quad (2)$$

$$e_k^- = x_k - \hat{x}_k^- \quad (3) \quad P_k^- = E \left[ e_k^- \cdot e_k^{-T} \right] \quad (4)$$

Aus der Prädiktion ergibt sich der „a priori“ Systemzustand und die „a priori“ Schätzfehlerkovarianz für den nächsten Rekursionsschritt. Über die Messmatrix  $H$  sind zudem die rekonstruierten Messwerte abgreifbar. Die Messgenauigkeit des Laserscanners fließt durch die angegebene Varianz des Messfehlers  $R_k$  in die Berechnung der Messfehlerkorrektur ein. Das Systemrauschen, z. B. eine Wasserfütze auf dem Boden, wird durch  $Q_k$  berücksichtigt.

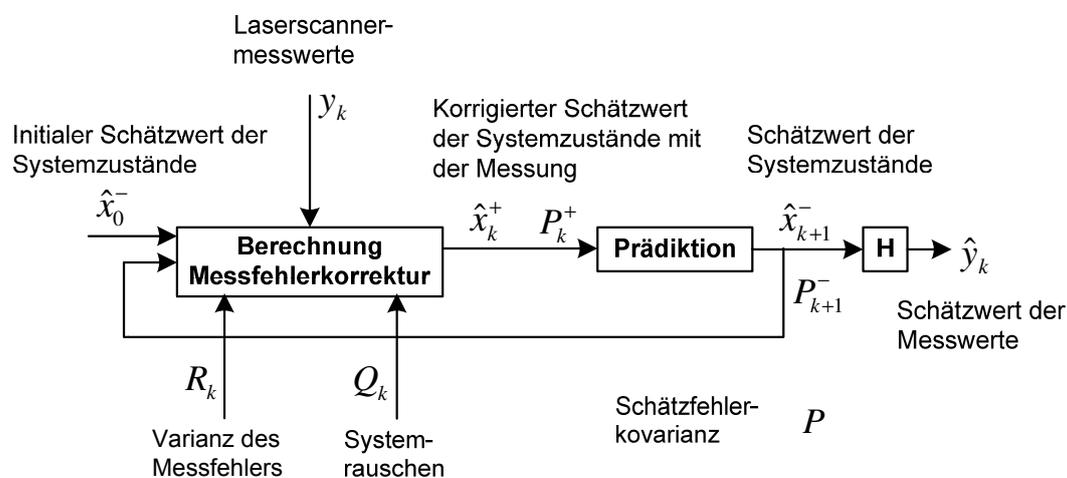


Abb. 2: Übersicht des Kalman Filters

## 5 Zustandsbeobachter nach Luenberger

Der Luenberger Beobachter eignet sich für die asymptotische Annäherung des mathematischen Modells an das Fahrzeug bei Anfangsstörungen, aber nicht bei laufenden stochastischen Störungen. In diesem Fall wird das Kalman Filter benutzt.

Der Zustandsbeobachter nach Luenberger von 1964 berücksichtigt kein System- und kein Messrauschen. Das Systemrauschen ist z.B. ein feuchter Untergrund, auf dem die Räder eine andere Haftreibung haben als auf dem trockenen übrigen Boden. Das Messrauschen wird u.a. durch die Messungenauigkeit des Laserscanners verursacht.

Bei der Beschreibung des Fahrzeugs als dynamisches System mit Zustandsvariablen beinhaltet der Zustandsvektor  $\underline{x}$  die Geschwindigkeit, vgl. Abb. 3. Als Eingangsvektor  $\underline{u}$  dient die Soll Geschwindigkeit und als Messvektor  $\underline{y}$  die Abstands- und Drehgebermesswerte. Die Messwerte werden über die Messmatrix  $\underline{H}$  von dem Zustandsvektor abgegriffen. Bei dem Fahrzeug kann nicht auf den „wahren“ Zustandsvektor zugegriffen werden und zudem ist der Startzustand  $\underline{x}_0$  unbekannt. Deshalb wird der Eingangsvektor zusätzlich dem mathematischen Modell zugeführt. Hier wird der Startzustand  $\hat{\underline{x}}_0$  geschätzt. Somit ist der Zustandsvektor  $\hat{\underline{x}}$  und der Messvektor  $\hat{\underline{y}}$  nicht gleich mit den entsprechenden Vektoren aus dem Fahrzeug.

Aber durch die Rückführung der Differenz von  $\underline{y} - \hat{\underline{y}}$ , verstärkt über die Matrix  $\underline{L}$ , wird das mathematische Modell asymptotisch dem Fahrzeug angenähert. Dabei wird  $\underline{L}$  so bestimmt, dass der Schätzfehler  $\tilde{\underline{x}}(t) = \underline{x}(t) - \hat{\underline{x}}(t)$  für  $t \rightarrow +\infty$  gegen Null strebt. Die Eigenwerte der Dynamikmatrix  $\underline{F} = \underline{A} - \underline{L} \underline{H}$  müssen dazu links der j-Achse liegen.

## 6 Querverbindung zum Kalman Filter

Der Luenberger Beobachter und das Kalman Filter unterscheiden sich nur in der Bestimmung der Rückführmatrix  $\underline{L}$ . Bei dem Luenberger Beobachter wird hierzu, wie oben beschrieben, die Polvorgabe für die Berechnung der Matrix  $\underline{L}$  verwendet. Im Gegensatz dazu wird bei dem Kalman Filter die Optimierung des quadratischen Gütemaßes  $J$  genutzt. Das Gütemaß  $J$  wird mit der Formel 5 bestimmt. Danach berechnet sich das Gütemaß aus der Summe der Erwartungswerte von den Quadraten der Schätzfehler.

$$J = \sum_{i=1}^n E(\tilde{x}_i^2) \quad (5)$$

Auf die Berechnung der Matrix  $\underline{L}$  für das Kalman Filter mit Hilfe des Riccati Entwurfs wird in dieser Seminararbeit nicht eingegangen, hierfür sei auf [2] verwiesen.

## 7 Diskrete System- und Messgleichung

Die diskrete System- und Messgleichung lassen sich aus den zeitkontinuierlichen Gleichungen ableiten. Die Vektoren und Matrizen erhalten jeweils einen Index für den Berechnungsschritt, somit ergeben sich die Gleichungen 6 und 7.

Das Kalman Filter berücksichtigt, wie schon erwähnt, das System- und Messrauschen, folglich fließen diese in die jeweilige Gleichung mit ein. Das Systemrauschen, z.B. der feuchte Untergrund, geht als Vektor  $\underline{w}_k$  ein. Das Messrauschen, z.B. die Messgenauigkeit, fließt als Vektor  $\underline{v}_k$  ein.

Diskrete Systemzustandsgleichung:

$$x_{k+1} = A_k \cdot x_k + B_k \cdot u_k + w_k \quad (6)$$

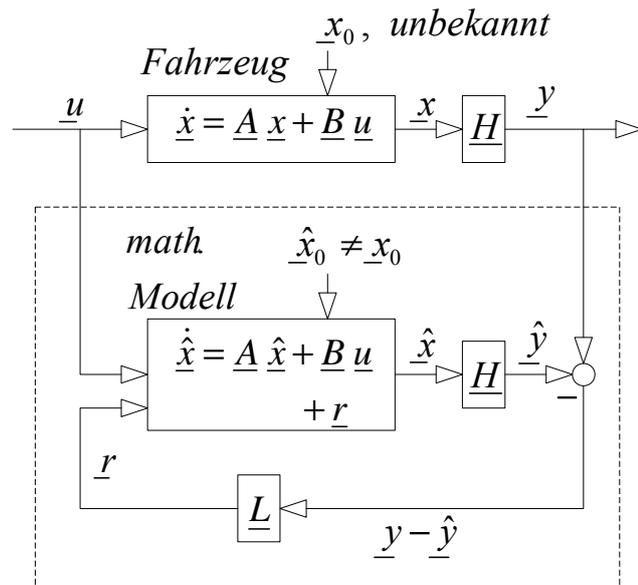


Abb. 3: Luenberger Beobachter

Bedeutung	Var.	Dimension
Zustand Fahrzeug	$x_k$	$n$ -dim.
Soll Geschwindigkeit	$u_k$	$l$ -dim.
Systemmatrix	$A_k$	$(n \cdot n)$ -Matrix
Eingabematrix	$B_k$	$(n \cdot l)$ -Matrix
Systemrauschen	$w_k$	$n$ -dim.

Messgleichung:

$$y_k = H_k \cdot x_k + v_k \quad (7)$$

Bedeutung	Var.	Dimension
Abstandswerte, Drehgebermesswerte	$y_k$	$m$ -dim.
Messmatrix	$H_k$	$(m \cdot n)$ -Matrix
Messrauschen	$v_k$	$m$ -dim.

## 8 Regelschleife des Kalman Filters

Die bisherige Betrachtung für das Kalman Filter in den Kapiteln 5 und 6 ist für den zeitkontinuierlichen Bereich. Für die Rekonstruktion der Messsignale auf  $\mu$ C-Plattformen wird allerdings ein zeitdiskretes Kalman Filter benötigt. Für die Herleitung des diskreten Kalman Filters aus dem kontinuierlichen sei auf [7] verwiesen.

Im diskreten Bereich arbeitet das Kalman Filter in einer Regelschleife, wie in Abb. 4 dargestellt. Am Anfang wird der Startzustand des Fahrzeugs  $\hat{x}_0^-$  geschätzt. Im ersten Schritt der rekursiven Schleife wird unter Berücksichtigung der aktuellen Abstandsmessung  $y_k$ , die Messfehlerkorrektur berechnet. Anschließend wird die zeitliche Entwicklung des Systems (Prädiktion) ermittelt. Die Kalman Verstärkung  $K_k$  wird in (1) so gewählt, dass die Schätzfehlerkovarianz  $P_k^+$  mini-

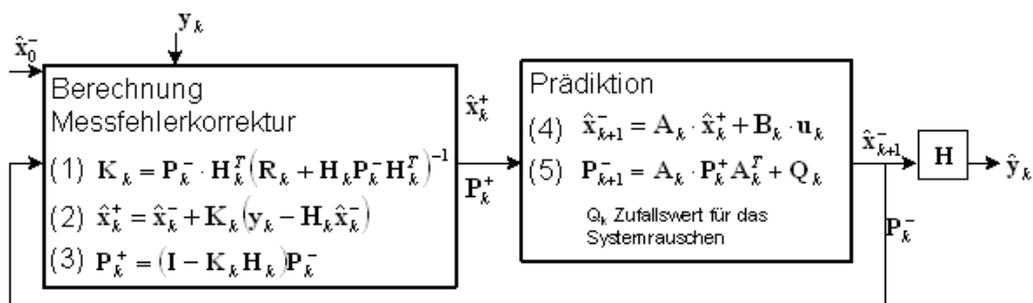


Abb. 4: Regelschleife des Kalman Filters

mal wird. Hierbei gilt, wenn die Varianz des Messfehlers  $R_k$  gegen null strebt, dann wird  $K_k$  maximal und somit der aktuellen Abstandsmessung  $y_k$  mehr vertraut. Strebt allerdings die Fehlerkovarianz  $P_k^-$  gegen null, so geht auch die Kalman Verstärkung gegen null. Die Korrektur der Systemzustandsschätzung  $\hat{x}_k^+$  mit der aktuellen Messung  $y_k$  wird in (2) berechnet. Dabei ist  $\hat{x}_k^-$  der vorherige Schätzwert und die Kalman Verstärkung ein Gewichtungsfaktor, der ausdrückt, zu welchem Anteil die Differenz des aktuellen Messwerts  $y_k$  und dem vorhergesagten Messwert  $H_k \cdot \hat{x}_k^-$  einfließt. Zuletzt wird die Fehlerkovarianzmatrix  $P_k^+$  korrigiert (3).

Bei der Prädiktion wird zunächst der Systemzustand  $\hat{x}_{k+1}^-$  des Fahrzeugs (4) und anschließend die Fehlerkovarianzmatrix  $P_{k+1}^-$  (5) prädiert.

## 9 Simulationsbeispiel Ideale Wurfparabel

In diesem Kapitel wird die Funktionsweise des Kalman Filters anhand der Idealen Wurfparabel Abb. 5 gezeigt. Beispielhaft sei hier an einen kleinen Jungen gedacht, der einen Ball 1,27m weit wirft. Die Geschwindigkeit in x-Richtung ist konstant und beträgt 1,9m/s. In y-Richtung nimmt die Geschwindigkeit vom Startpunkt mit 3,268m/s zunächst ab und wird am Wendepunkt negativ, bis schließlich der Ball den Boden berührt und die Simulation abbricht. Der Ball benötigt für diese Flugbahn mit den entsprechenden Startgeschwindigkeiten 0,665s.

Physikalisch lässt sich die Wurfparabel mit den Gleichungen 8, 9 und 10 beschreiben. Die

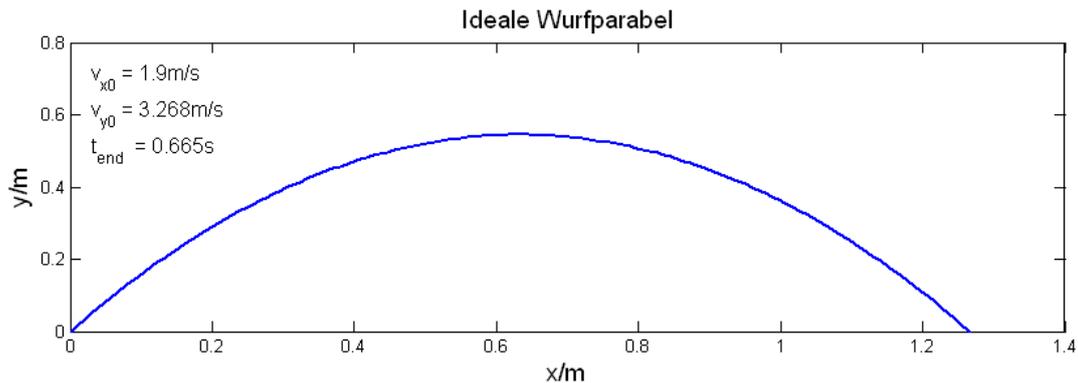


Abb. 5: Ideale Wurfparabel

Ableitung des Ortes über die Zeit ist die Geschwindigkeit, Gln. 8 und 9. Zudem ist die Geschwindigkeit in x-Richtung konstant, Gl. 8, d.h. der Luftwiderstand wird vernachlässigt. Die Ableitung der Geschwindigkeit in y-Richtung ist die Erdbeschleunigung  $g$ , Gl. 10.

$$v_x = v_0 = \frac{dx}{dt} = \text{const.} \quad (8)$$

$$v_y(t) = \frac{dy}{dt} \quad (9)$$

$$\frac{dv_y}{dt} = g = -9,81 \frac{m}{s^2} \quad (10)$$

Für das digitale Kalman Filter müssen die Systemgleichungen diskretisiert werden. Folglich ergeben sich vier Formeln, für die Geschwindigkeit und für den Ort, jeweils für  $x$  und  $y$ . Für die ersten drei Herleitungen  $v_x$  in 11,  $v_y$  in 12 und  $x$  in 13 werden die Differentialquotienten zu Differenzenquotienten umgewandelt und diese entsprechend aufgelöst.

$$\begin{aligned} v_x &= v_0 \\ v_x(k+1) &= v_x(k) \\ \Rightarrow v_x &= \text{const.} \end{aligned} \quad (11)$$

$$\begin{aligned} \frac{dv_y}{dt} &= a \Leftrightarrow \frac{\Delta v_y}{\Delta t} = a = -g \\ \Delta v_y &= -g \cdot \Delta t \\ v_y(k+1) - v_y(k) &= -g \cdot \Delta t \\ v_y(k+1) &= v_y(k) - g \cdot \Delta t \end{aligned} \quad (12)$$

$$\begin{aligned}
\frac{dx}{dt} &= v_x \Leftrightarrow \frac{\Delta x}{\Delta t} = v_x \\
\Delta x &= \Delta t \cdot v_x \\
x(k+1) - x(k) &= \Delta t \cdot v_x \\
x(k+1) &= x(k) + \Delta t \cdot v_x
\end{aligned} \tag{13}$$

Die Herleitung der diskreten Systemgleichung für  $y$  wird mit Hilfe der Trapezintegration gezeigt. Die gesamte Fläche unter der Geraden in Abb. 6 entspricht dabei  $y(k+1)$  in der Gl. 14, wobei  $\Delta t$  das Abtastintervall ist. Folgend die Flächen aus Abb. 6 und die entsprechenden Summanden aus Gl. 14:

- (1) :  $y(k)$
- (2) :  $\Delta t v_y(k)$
- (3) :  $\Delta t \frac{1}{2} \{v_y(k+1) - v_y(k)\}$

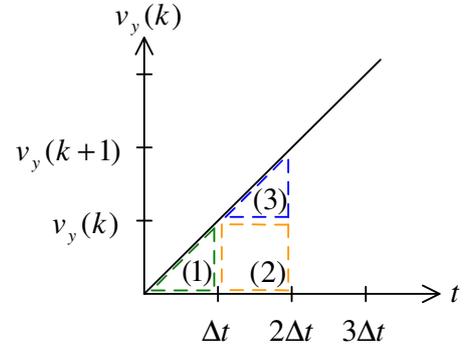


Abb. 6: Trapezintegration

$$\begin{aligned}
y(k+1) &= y(k) + \left[ \Delta t v_y(k) + \Delta t \frac{1}{2} \{v_y(k+1) - v_y(k)\} \right] \\
y(k+1) &= y(k) + \Delta t v_y(k) - \frac{1}{2} g \Delta t^2
\end{aligned} \tag{14}$$

Zusammengefasst ergibt sich aus Gl. 15 durch Einsetzen die stochastische Systemzustandsgleichung 16. Die stochastische Eigenschaft ergibt sich aus dem Systemrauschen  $w_k$ .

$$x_{k+1} = A_0 \cdot x_k + B_0 \cdot u_0 + w_k \tag{15}$$

$$\begin{pmatrix} x \\ y \\ v_x \\ v_y \end{pmatrix}_{k+1} = \begin{pmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ v_x \\ v_y \end{pmatrix}_k + \begin{pmatrix} 0 & 0 \\ 0 & \frac{1}{2} \Delta t \\ 0 & 0 \\ 0 & \Delta t \end{pmatrix} \cdot \begin{pmatrix} 0 \\ -g \end{pmatrix} + w_k \tag{16}$$

Bei der Wurfparabel ist es ausschließlich möglich, den Ort des Balls in x- und y-Richtung zu messen. Die Geschwindigkeiten sind nicht direkt messbar. Daraus folgt die entsprechende Messmatrix  $\underline{H}$  in der Messgleichung 18. Der Vektor  $v_k$  repräsentiert das Messrauschen.

$$y_k = H_0 \cdot x_k + v_k \tag{17}$$

$$\begin{pmatrix} x_m \\ y_m \end{pmatrix}_k = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ v_x \\ v_y \end{pmatrix}_k + v_k \tag{18}$$

## 9.1 Beispielrechnung für Schritt 0

Zur Verdeutlichung der Arbeitsweise des Kalman Filters sind im Anhang A die Berechnungen für den ersten Rekursionsschritt aufgezeigt. Die Startschätzungen für die Wurfparabel sind den Gln. 19 für den Systemzustand und 20 für die Fehlerkovarianzmatrix zu entnehmen. Die

Varianz des Messfehlers ist  $R_k$  und das Systemrauschen, welches hier Luftwirbel sind, ist  $Q_k$  vgl. 21 und 22. Der Messvektor bei  $k = 0$  sei  $y_0 = (0 \ 0,08)^T$  und das Abtastintervall  $\Delta t = 0,01s$ .

$$\hat{x}_0^- = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \end{pmatrix} \quad (19) \quad P_0^- = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (20)$$

$$R_k = \begin{pmatrix} 0,01 & 0 \\ 0 & 0,01 \end{pmatrix} \quad (21) \quad Q_k = \begin{pmatrix} 10^{-5} & 0 & 0 & 0 \\ 0 & 10^{-5} & 0 & 0 \\ 0 & 0 & 10^{-5} & 0 \\ 0 & 0 & 0 & 10^{-5} \end{pmatrix} \quad (22)$$

Bei den Berechnungen werden ausschließlich die entsprechenden Matrizen in die Gleichungen der Regelschleife eingesetzt und die Lösungen berechnet. Für den nächsten Rekursionsschritt  $k = 1$  muss selbstverständlich ein neuer Messvektor zur Verfügung stehen.

## 9.2 Matlab Simulation

Zur Veranschaulichung der Arbeitsweise des Kalman Filters ist in diesem Kapitel die Simulation mit Matlab zu der Wurfparabel aufgezeigt. Hierzu wird auf die Ideale Wurfparabel bei jedem Messpunkt in y-Richtung ein Zufallswert addiert, welcher im Bereich von -0,1m bis +0,1m liegt. Sollte der „gemessene“ Punkt negativ sein, so wird der y-Wert auf null gesetzt. Diese Fallbetrachtung ist sowohl zu Beginn als auch zum Ende der Wurfparabel nötig. Das Abtastintervall ist  $\Delta t = 0,005s$ . Die Erdbeschleunigung wird als  $g = 9,80665 \frac{m}{s^2}$  angenommen. In Abb. 7 ist die rekonstruierte Wurfparabel als Linie durch das Kalman Filter mit Hilfe der Messsignale dargestellt. Die Ideale Wurfparabel ist zum Vergleich als gestrichelte Linie und die Messpunkte als Kreuze gezeigt.

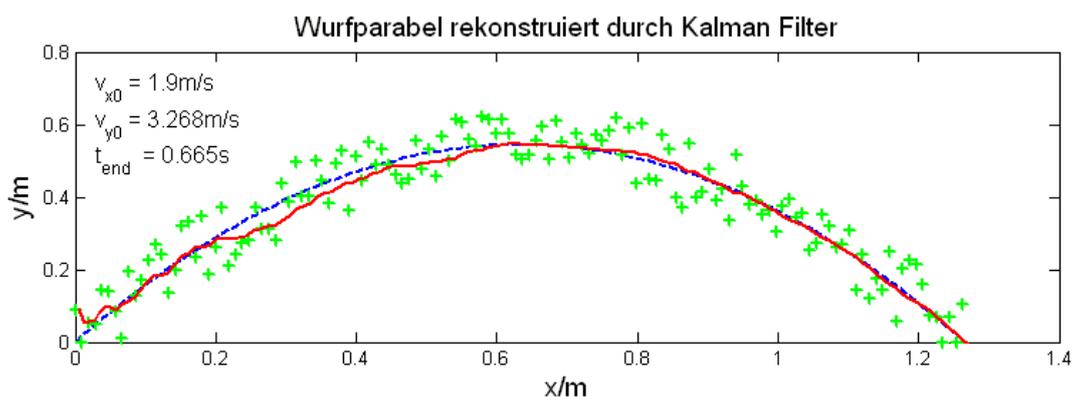


Abb. 7: Rekonstruierte Wurfparabel

## 10 Objektverfolgung

In diesem Kapitel erweitern wir das Simulationsbeispiel um einen kleinen Baum, der die Flugbahn des Balls zu seinem Beobachter verdeckt. Dieser Fall tritt häufig bei der Objektverfolgung auf. Ein Objekt das verfolgt wird, ist kurzfristig von einem anderen Objekt verdeckt. In Abb. 8 ist dieses zwischen 0,7m und 0,8m der Fall. In der Simulation ist dazu der y-Wert auf 0,1m gesetzt und anschließend mit dem Zufallswert verwechselt. Es ist zu erkennen, dass die geschätzte

Position des Balls den falschen Messwerten schnell folgt und sich anschließend bei korrekten Messwerten nur langsam der idealen Kurve annähert. Dieses liegt an der konstanten Varianz des Messfehlers  $R_k = 0,01$ , d. h. die Kalman Verstärkung wird von  $R_k$  immer gleich beeinflusst. Die Lösung für das Problem, dass die geschätzte Position des Balls den falschen Messwerten

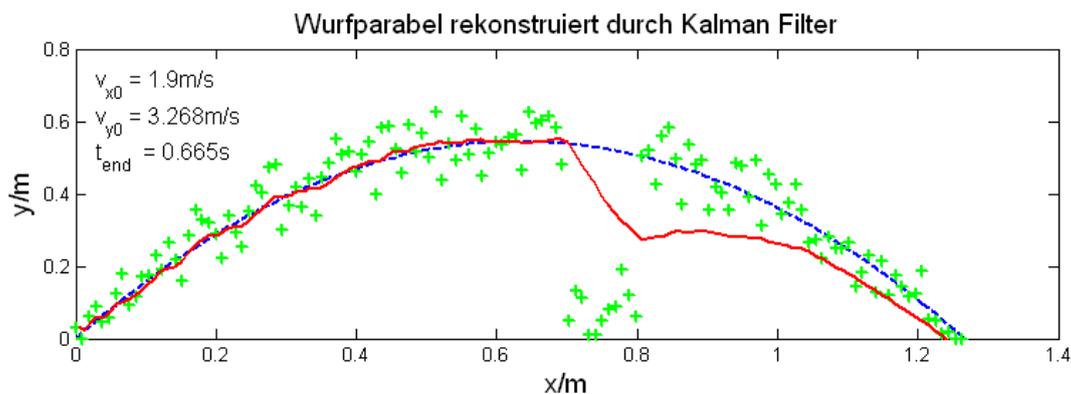


Abb. 8: Objektverfolgung Fehler

relativ schnell folgt, ist ein Suchfenster für gültige Messwerte. Bei Messpunkten innerhalb des Suchfensters bleibt die Varianz des Messfehlers bei  $R_k = 0,01$ . Wenn der Messpunkt außerhalb des Suchfensters ist, dann wird  $R_k$  auf 1 gesetzt. In diesem Fall geht die aktuelle Messung praktisch nicht in den Filter ein, d. h. der Messung wird nicht vertraut. Das Kalman Filter läuft in diesem Fall im Prädiktionsmodus. Sind bereits vor der Störung gute Schätzungen über Position und Geschwindigkeit des Balls verfügbar, so funktioniert dieses für eine gewisse Zeit gut. Wenn die Messwerte wieder im Suchfenster liegen, wird die Varianz des Messfehlers auf den normalen Wert gesetzt und somit wird die Schätzung wieder mit Messwerten gestützt. Das Ergebnis der Simulation mit Suchfenster ist in Abb. 9 gezeigt, der Matlab Simulationscode hierzu befindet sich im Anhang B.

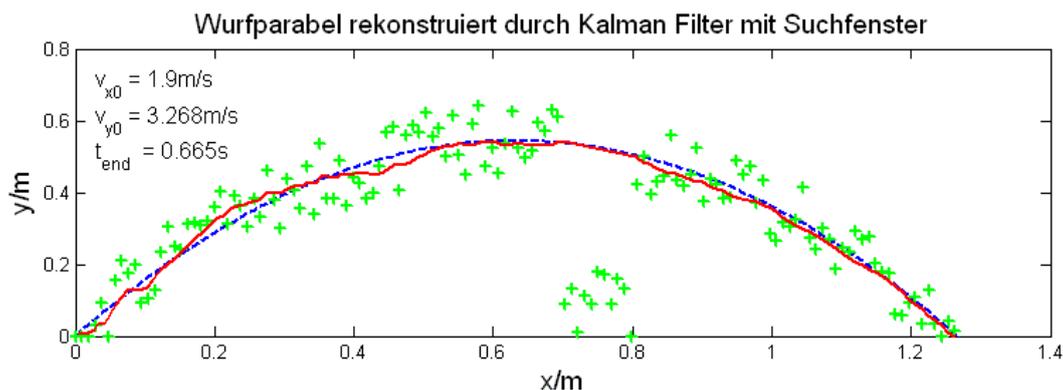


Abb. 9: Objektverfolgung OK

## 11 Zusammenfassung und Ausblick

Diese Seminararbeit bietet eine Einführung in das diskrete Kalman Filter und ein Simulationsbeispiel anhand der Wurfparabel. Somit ist das Handwerkszeug gegeben, um gestörte Messsignale in einem System zu rekonstruieren. Außerdem wurde ein Einblick in die Objektverfolgung mit einem Suchfenster gewährt.

Das Kalman Filter ist ein rekursives Filter und eignet sich deshalb gut für Echtzeitanwendungen. Es müssen nicht wie beim Moving-Average-Filter mehrere Messwerte vorgehalten werden und es gibt keine zeitliche Verzögerung. Sind die Varianzen des Mess- und Systemrauschen konstant, so ist es möglich, die Kalman Verstärkung  $K_k$  und die Kovarianz des Schätzfehlers  $P_k$  vorab zu berechnen, da diese sich sehr schnell konstanten Werten annähert. Eine Voraussetzung für das Kalman Filter ist die Beobachtbarkeit der Systemzustände durch die Messung, andernfalls divergiert das Filter. Zudem muss auf die numerische Genauigkeit des Systems geachtet werden, damit keine Rundungsfehler bei den Berechnungen entstehen.

## 11.1 Motivation aus der Industrie

Auf das Kalman Filter wird in diversen wissenschaftlichen Artikeln des Transportation Days und in einigen Dissertationen verwiesen, exemplarisch seien hierfür [3] und [6] genannt. Es wird demnach in verschiedenen Assistenzsystemen und Umgebungserkennungssystemen eingesetzt. Zur Motivation außerhalb der Hochschule für Angewandte Wissenschaften Hamburg und den Forschungsarbeiten zu diesem Thema, sind z. B. verschiedene Arbeitsplatzangebote zu sehen. Zum Einen ist dort die Firma Siemens VDO Automotive [8] zu nennen, die im Dezember Entwickler mit den Aufgabenbereichen „Konzeption, Entwicklung und Implementierung von Algorithmen zur Fahrumgebungserfassung für Fahrerassistenzsysteme“ sowie für „Embedded Systeme (Bildverarbeitung, Radar, Laser)“ suchte. Zum Zweiten ist die Firma Ibeo in Hamburg zu nennen, mit folgender Stellenbeschreibung: „Entwicklung von Algorithmen für Sensoren bis hin zur Fahrerassistenzapplikation auf den Versuchswagen.“ Diese beiden Firmenbeispiele zeigen die Aktualität des behandelten Themenkomplexes in der Industrie.

## 11.2 Thema für die Masterarbeit

In der Masterarbeit muss zunächst das bestehende Fahrzeugmodell des FAUSTs analysiert und erweitert werden, um die Systemgleichungen zu erhalten. Diese sind für die Anwendung des Kalman Filters zwingend erforderlich. Anschließend soll der Moving-Average-Filter in dem Brems- und dem Ausweichassistent durch das Kalman Filter ersetzt werden. Somit werden die beiden Fahrerassistenzsysteme verbessert.

Im weiteren Verlauf der Masterarbeit soll eine Objektverfolgung realisiert werden. Dazu werden die Daten des Laserscanners benutzt, um die Position und Geschwindigkeit anderer Objekte zu ermitteln. Die Sensordaten werden zu Objekten zusammengefasst und deren Eigenschaften über mehrere Scans geschätzt. Damit ist eine weitere Grundlage für Assistenzsysteme gelegt. Darauf aufbauend soll eine ständige Ortsbestimmung des FAUSTs im Raum vorgenommen werden, denn wenn bekannte Objekte im Raum erkannt werden, so ist es möglich, mit Hilfe des Kalman Filters die aktuelle Position im Raum optimal zu schätzen. Ziel ist es, eine Karte bzw. ein Koordinatensystem zu erstellen, in dem die aktuelle Position des FAUSTs abgebildet ist. Zur Erarbeitung dieses Themenkomplex dienen neben den wissenschaftlichen Artikeln und Dissertationen aus dem Automotive Bereich zusätzlich Forschungsarbeiten und Literatur aus der Robotik. Beispielhaft sei hierfür auf [4] verwiesen. Diese Dissertation stellt unter anderem Verfahren zur exakten Positionsbestimmung und der Navigation in sich veränderten Umgebungen für Roboter vor. Es handelt sich somit um die gleichen Themengebiete, die für das FAUST Projekt interessant sind. Der Vorteil bei den Arbeiten aus der Robotik und hier speziell Roboter-Fußball ist, dass die Forschungsergebnisse meist offen liegen, im Gegensatz dazu der Automotive Bereich, in denen meist industrielle Interessen vertreten sind.

---

## Literatur

- [1] CORDES, Stefan: *Automatischer Bremsassistent auf Basis einer Laserscanner-Abstandserfassung für ein fahrerloses Transportsystem*, Hochschule für Angewandte Wissenschaften Hamburg, Masterarbeit, 2006
- [2] FÖLLINGER, Otto: *Regelungstechnik*. Hüthig, 1994. – ISBN 3-7785-2336-8
- [3] FUERSTENBERG, Kay: *Intersection Safety*, IBEO, Abstract, 2005
- [4] GUTMANN, Jens-Steffen: *Robuste Navigation autonomer mobiler Systeme*. Akademische Verlagsgesellschaft Aka, 2000. – ISBN 3-89838-241-9
- [5] IBEO: *IBEO*. 2006. – [www.ibeo-as.com](http://www.ibeo-as.com)
- [6] KIRCHNER, Alexander: *Sensordatenverarbeitung eines Laserscanners für autonome Fahrfunktionen von Kraftfahrzeugen*, Universität der Bundeswehr Hamburg, Dissertation, 2000
- [7] SCHLITT, Herbert: *Systemtheorie für stochastische Prozesse*. Springer Verlag, 1992. – ISBN 3-540-54288-4
- [8] SIEMENS: *Siemens*. 2006. – [www.siemensvdo.de](http://www.siemensvdo.de)

## A Beispielrechnung für Schritt 0

Berechnung der Kalman Verstärkung

$$K_k = P_k^- \cdot H_0^T (R_0 + H_0 P_k^- H_0^T)^{-1} \quad (23)$$

$$\begin{aligned} K_0 &= \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{pmatrix} \cdot \\ &\cdot \left[ \begin{pmatrix} 0,01 & 0 \\ 0 & 0,01 \end{pmatrix} + \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{pmatrix} \right]^{-1} \\ &= \underline{\underline{\begin{pmatrix} 0,99 & 0 \\ 0 & 0,99 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}}} \end{aligned} \quad (24)$$

Korrektur der Schätzung mit der Messung  $y_k$

$$\hat{x}_k^+ = \hat{x}_k^- + K_k (y_k - H_0 \hat{x}_k^-) \quad (25)$$

$$\begin{aligned} \hat{x}_k^+ &= \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \end{pmatrix} + \begin{pmatrix} 0,99 & 0 \\ 0 & 0,99 \\ 0 & 0 \\ 0 & 0 \end{pmatrix} \cdot \left[ \begin{pmatrix} 0 \\ 0,08 \\ 0 \\ 0 \end{pmatrix} - \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \end{pmatrix} \right] \\ &= \underline{\underline{\begin{pmatrix} 0 \\ 0,0792 \\ 1 \\ 1 \end{pmatrix}}} \end{aligned} \quad (26)$$

Korrektur der Fehlerkovarianz

$$P_k^+ = (I - K_k H_0) P_k^- \quad (27)$$

$$\begin{aligned} P_k^+ &= \left[ \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} - \begin{pmatrix} 0,99 & 0 \\ 0 & 0,99 \\ 0 & 0 \\ 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \right] \cdot \\ &\cdot \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} = \underline{\underline{\begin{pmatrix} 0,01 & 0 & 0 & 0 \\ 0 & 0,01 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}}} \end{aligned} \quad (28)$$

Prädiziere Systemzustand

$$\widehat{x}_{k+1}^- = A_0 \cdot \widehat{x}_k^+ + B_0 \cdot u_0 \quad (29)$$

$$\begin{aligned} \widehat{x}_{k+1}^- &= \begin{pmatrix} 1 & 0 & 0,01 & 0 \\ 0 & 1 & 0 & 0,01 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 0 \\ 0,0792 \\ 1 \\ 1 \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & \frac{1}{2} \cdot 0,01^2 \\ 0 & 0 \\ 0 & 0,01 \end{pmatrix} \cdot \begin{pmatrix} 0 \\ -g \end{pmatrix} \\ &= \underline{\underline{\begin{pmatrix} 0,01 \\ 0,08915 \\ 1 \\ 0,9019 \end{pmatrix}}} \end{aligned} \quad (30)$$

Prädiziere Fehlerkovarianz

$$P_{k+1}^- = A_0 \cdot P_k^+ A_0^T + Q_0 \quad (31)$$

$$\begin{aligned} P_{k+1}^- &= \begin{pmatrix} 1 & 0 & 0,01 & 0 \\ 0 & 1 & 0 & 0,01 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 0,01 & 0 & 0 & 0 \\ 0 & 0,01 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \\ &\cdot \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0,01 & 0 & 1 & 0 \\ 0 & 0,01 & 0 & 1 \end{pmatrix} + \begin{pmatrix} 10^{-5} & 0 & 0 & 0 \\ 0 & 10^{-5} & 0 & 0 \\ 0 & 0 & 10^{-5} & 0 \\ 0 & 0 & 0 & 10^{-5} \end{pmatrix} \\ &= \underline{\underline{\begin{pmatrix} 0,01001 & 0 & 0,01 & 0 \\ 0 & 0,01001 & 0 & 0,01 \\ 0,01 & 0 & 1,01001 & 0 \\ 0 & 0,01 & 0 & 1,01001 \end{pmatrix}}} \end{aligned} \quad (32)$$

## B Matlab Simulationscode

Listing 1: Matlab Simulationscode

```
g = 9.80665
H = 0.005
k = 0
n = 134

random = randint(n,1,[-10,10])
% -----
% Fuer ideales System
t = [0]

x_ideal = [0];
y_ideal = [0];

v_0 = 3.8
vx_0 = 0.5 * v_0
vy_0 = 0.86 * v_0
```

---

```

vx_ideal = [vx_0];
vy_ideal = [vy_0];
% -----
% Fuer Kalman Filter
Mat_K = zeros(4,4);
% Startschaeztung
Mat_Pm = [1 0 0 0; 0 1 0 0; 0 0 1 0; 0 0 0 1]

Mat_R = [0.01 0; 0 0.01]
Mat_H = [1 0 0 0; 0 1 0 0]

% Startschaeztung
Vec_Xm = [0 0 1 1].';
Array_Xm = zeros(4,n+1);
Array_Xm(1) = Vec_Xm(1);
Array_Xm(2) = Vec_Xm(2);
Array_Xm(3) = Vec_Xm(3);
Array_Xm(4) = Vec_Xm(4);
x_kal = [0];
y_kal = [0];
x_mes = [0];
y_mes = [0];

Vec_Xp = [0 0 0 0].';

Vec_Y = [0 0].';

Mat_Pp = zeros(4,4);
Mat_I = [1 0 0 0; 0 1 0 0; 0 0 1 0; 0 0 0 1]

Mat_A = [1 0 H 0; 0 1 0 H; 0 0 1 0; 0 0 0 1]
Mat_B = [0 0; 0 ( 0.5 * H*H); 0 0; 0 H]

Vec_u = [0 -g].';

Mat_Q = [0.00001 0 0 0; 0 0.00001 0 0; 0 0 0.00001 0; 0 0 0
0.00001]

for k = 1:n
% -----
% ideales System
t(k+1) = t(k) + H

x_ideal(k+1) = x_ideal(k) + H * vx_ideal(k)
y_ideal(k+1) = y_ideal(k) + H * vy_ideal(k) - 0.5 * g * H *
H

vx_ideal(k+1) = vx_ideal(k)
vy_ideal(k+1) = vy_ideal(k) - g * H

```

---

```

% -----
% Messung

Vec_Y(1) = x_ideal(k)
rand      = (random(k)/100)
Vec_Y(2) = y_ideal(k) + rand

    % Wurfparabel mit Stoerung
    if( 0.7 < x_ideal(k) & x_ideal(k) < 0.8 )
        Vec_Y(2) = 0.1 + rand
        Mat_R(1) = 1
        Mat_R(4) = 1
    else
        Mat_R(1) = 0.01
        Mat_R(4) = 0.01
    end
    % Ende Wurfparabel mit Stoerung

if (Vec_Y(2) < 0)
    Vec_Y(2) = 0
end
x_mes(k) = Vec_Y(1);
y_mes(k) = Vec_Y(2);

% -----
% Korrektur mit der Messung

Inverse = inv(Mat_R + Mat_H * Mat_Pm * Mat_H')
Mat_K   = Mat_Pm * Mat_H' * Inverse
Vec_Xp  = Vec_Xm + Mat_K * (Vec_Y - Mat_H * Vec_Xm)

Mat_Pp  = (Mat_I - Mat_K * Mat_H) * Mat_Pm

% -----
% Prädiktion
Vec_Xm = Mat_A * Vec_Xp + Mat_B * Vec_u

    Array_Xm( (k*4)+1 ) = Vec_Xm(1);
    Array_Xm( (k*4)+2 ) = Vec_Xm(2);
    Array_Xm( (k*4)+3 ) = Vec_Xm(3);
    Array_Xm( (k*4)+4 ) = Vec_Xm(4);
    x_kal(k) = Vec_Xm(1);
    y_kal(k) = Vec_Xm(2);

    Mat_Pm = Mat_A * Mat_Pp * Mat_A' + Mat_Q
end

% ideal
subplot(2,1,1);

```

```
plot(x_ideal ,y_ideal , 'LineWidth',2 );
axis([0,1.4, 0,0.8])
title('Ideale_Wurfparabel', 'FontSize',14)
xlabel('x/m', 'FontSize',14)
ylabel('y/m', 'FontSize',14)
str1 = ['v_{x0}=_', num2str(vx_0) 'm/s'];
str2 = ['v_{y0}=_', num2str(vy_0) 'm/s'];
str3 = ['t_{end}=_', num2str(t(k)) 's'];
text(0.03,0.7, str1 , 'FontSize',12)
text(0.03,0.6, str2 , 'FontSize',12)
text(0.03,0.5, str3 , 'FontSize',12)

% Kalman Filter
subplot(2,1,2);
plot( x_ideal ,y_ideal ,'--', x_mes,y_mes,'g+', x_kal ,y_kal ,'r'
, 'LineWidth',2);
axis([0,1.4, 0,0.8])
xlabel('x/m', 'FontSize',14)
ylabel('y/m', 'FontSize',14)
title('Wurfparabel_rekonstruiert_durch_Kalman_Filter_mit_
Suchfenster', 'FontSize',14)
str1 = ['v_{x0}=_', num2str(vx_0) 'm/s'];
str2 = ['v_{y0}=_', num2str(vy_0) 'm/s'];
str3 = ['t_{end}=_', num2str(t(k)) 's'];
text(0.03,0.7, str1 , 'FontSize',12)
text(0.03,0.6, str2 , 'FontSize',12)
text(0.03,0.5, str3 , 'FontSize',12)
```