

Kalman-Filter und Partikelfilter zur Selbstlokalisierung – Ein Vergleich

Dirk Neumann

8. Januar 2002

Zusammenfassung

Eine präzise und stabile Selbstlokalisierung stellt eine der wichtigsten Voraussetzungen für autonome mobile Roboterplattformen dar. Ohne ein genaues und zuverlässiges Wissen über die Position und Orientierung (Pose) des Roboters sind Weltmodellierung und komplexere Handlungsplanungen nicht möglich. Für die verschiedenen Plattformen, Sensorsysteme und Einsatzgebiete werden dazu die unterschiedlichsten Algorithmen eingesetzt. Den meisten Verfahren gemeinsam ist dabei ein meist nicht unerheblicher Schätzfehler.

Ziel dieser Arbeit war die Verbesserung der Selbstlokalisierung durch Anwendung statistischer Filter. Verglichen wurden dabei die Genauigkeit und Robustheit des Kalman-Filters mit einem Bayes-Partikelfilters anhand von aufgezeichneten Testsequenzen, die durch ungenaue bzw. falsche Sensorinformation gekennzeichnet sind (partielle Verdeckung, Positionsveränderungen ohne Odometrie, starke Beschleunigungen, häufige Richtungswechsel, “kidnap robot problem”).

Zur Selbstlokalisierung standen auf den mobilen Roboterplattformen Algorithmen zur Verfügung, die jeweils die Daten des Laser-Scanners, der omnidirektionalen Kamera sowie der Odometrie auswerten. In einer etwas verkleinerten Variante des RoboCup-Spielfeldes wurden die Fehlerverteilungen

der Pose-Schätzungen in Abhängigkeit von der Position des Roboters an den Knotenpunkten eines 1x1-Meter-Gitters gemessen. Diese empirischen Fehlerverteilungen wurden dann durch multivariate Gauß-Funktionen approximiert. Zwischen den Messpunkten des 1x1-Meter-Gitters wurden die Parameter dieser Fehlerfunktionen interpoliert. Die Parameter des Odometrie-Fehlermodells wurden anhand einer Testfahrt mittels eines linearen Modells empirisch bestimmt. Die empirisch gewonnenen Fehlermodelle wurden für den Kalman-Filter und den Partikelfilter in identischer Form verwendet.

Obwohl beide Verfahren von gleichen Annahmen über die Fehlerverteilung ausgingen, zeigte sich für die aufgezeichneten Testsequenzen, bei denen die Voraussetzungen der Normalverteilung zum Teil massiv verletzt werden, ein deutlicher Unterschied in der Genauigkeit und der Konvergenzgeschwindigkeit zwischen beiden Verfahren. Der Partikelfilter konvergiert nach Ausreißern und mehreren falschen Messwerten deutlich schneller gegen den wahren Wert und liefert stabilere und genauere Schätzungen als der Kalman-Filter.

Die Annahme der Normalverteilung und die Verwendung des Kalman-Filters zur Sensordatenintegration erscheinen damit nicht zweckmäßig.

Inhaltsverzeichnis

1	Einleitung	4
2	Grundlagen	8
2.1	Wahrscheinlichkeit	8
2.2	Kalman-Filter	9
2.3	Bayes-Theorem	13
2.4	Partikelfilter	15
3	Implementierung des Partikelfilters	20
3.1	Gewichtung	21
3.2	Resampling	21
4	Fehlermodelle	22
4.1	Kovarianzberechnung	22
4.2	Fehlerkovarianz der Pose-Bestimmung	24
4.3	Odometrie-Fehlerkovarianzmodell	25
5	Evaluation	33
6	Fazit und Ausblick	40

1 Einleitung

In der mobilen Robotik wird zwischen globalen und lokalen Verfahren unterschieden. Globale Verfahren sind dadurch gekennzeichnet, dass sie Informationen verwenden und erzeugen, die in vom Roboter unabhängigen Räumen definiert sind, während bei lokalen Verfahren roboterzentrische Koordinaten benutzt werden.

Beide Informationsklassen haben unterschiedliche Vor- und Nachteile und damit ihre eigene Berechtigung. Globale Verfahren werden für globale Planungs- und Handlungsberechnungen genutzt. Der Vorteil lokaler Verfahren liegt dagegen häufig in einer deutlich geringeren Komplexität der Berechnung, da hier meist auf probabilistische Sensormodelle verzichtet wird und die Beziehung zwischen den Steuerungsgrößen und den Sensorinformationen meist auch eine deutlich geringe Komplexität aufweist, als dies bei globalen Verfahren der Fall ist.

Selbstlokalisationsverfahren stellen eine Brücke zwischen den beiden Informationsklassen her. Im Gegensatz zu globalen (Fremd)Lokalisierungsverfahren, bei denen wie zum Beispiel in der RoboCup *small size league* Deckenkameras verwendet werden, um die Position in globalen Koordinaten zu schätzen, sind Selbstlokalisationsverfahren in der mobilen Robotik dadurch charakterisiert, dass nur die Daten der Plattform zur Verfügung stehen.

Diese Daten müssen dann anhand von Zustandsinformationen und Modellen in Bezug zu den globalen Bezugssystemen gesetzt werden. Sowohl die Informationen über den aktuelle Zustände in globalen Räumen als auch die Sensor- und Bewegungsmodelle selbst sind dabei mit starken Unsicherheiten belastet. Probabilistische Verfahren stellen daher ein zentrales Element der meisten Selbstlokalisationsalgorithmen dar.

Die derzeit am häufigsten verwendeten statistischen Filter ist die Klasse der Kalman-Filter. Für multivariate Sensordaten, deren Fehler normalverteilt ist und deren Kovarianz bekannt ist, ist der Kalman-Schätzer optimal und stellt ein sehr effizientes Verfahren dar.

Die (theoriekonforme) Anwendung des Kalman-Filters ist an eine Reihe von

Voraussetzungen geknüpft. So wird verlangt, dass die empirische Fehlerverteilung tatsächlich durch eine theoretische gewonnene oder durch Schätzung parametrisierte Gaußverteilung modelliert werden kann. Diese Annahme wird jedoch selten explizit überprüft, auf eine Messung der tatsächlichen Fehlerverteilung und einen statistischen Anpassungstest, der die Annahme der Normalverteilung verwerfen könnte, wird in den meisten Arbeiten verzichtet. Für das hier verwendete visuelle Lokalisationsverfahren mittels omnidirektionaler Kamera sind die Fehler beispielsweise nicht normalverteilt.

Die Verletzung der Annahme der Normalverteilung stellt an sich noch nicht Brauchbarkeit des Kalman-Filters in Frage. Statistische Schätzverfahren für Normalverteilung sind in der Regel relativ robust gegen Verletzung dieser Voraussetzung. Schiefgipfligkeit, geringe Fehler bei der Varianzschätzung und einzelne Ausreißer haben bei hohem Aggregationsniveau auf die Genauigkeit und die Zuverlässigkeit der Schätzung nur eine geringe Auswirkung.

Entscheidend ist daher, wie häufig und wie groß die Abweichungen von der Normalverteilung tatsächlich ist. Daraus ließe sich dann der Einfluss auf den tatsächlichen Schätzfehler berechnen. Im Folgenden soll jedoch ein anderer Weg besprochen werden.

Verglichen wurden zwei Verfahren, der Kalman-Filter und ein Partikelfilter, die sich in ihren Annahmen nur hinsichtlich einer Komponente unterscheiden. In der vorgenommen Implementierung sind die Sensormodelle für beide Verfahren identisch und gehen von einer Normalverteilung aus, deren Parameter empirisch bestimmt wurden. Für beide Verfahren wurde dasselbe Bewegungsmodell verwendet. Als Schätzfunktion (Schätzer) wird für beide Methoden der Erwartungswert der Verteilung benutzt.

Der Unterschied zwischen dem Kalman-Filter und dem hier verwendeten Partikelfilter liegt in den Annahmen über die Verteilung des wahren Wertes. Der Kalman-Filter setzt voraus, dass alle Sensordaten denselben wahren Wert, allerdings mit unterschiedlicher Genauigkeit, messen. Daher wird gefolgert, dass das der Kalman-Schätzwert auch entsprechend normalverteilt ist und der (theoreti-

sche) Erwartungswert des Schätzers mit dem wahren Wert übereinstimmt. Die Schätzung kann dadurch sehr effizient durch den Mittelwert und die Kovarianzmatrix charakterisiert werden.

Beim Partikelfilter wird dagegen auf die Annahme der Normalverteilung des geschätzten Wertes verzichtet. Es wird nicht mehr von *einem* wahren Wert ausgegangen, sondern die Schätzung wird als Wahrscheinlichkeitsverteilung modelliert. Die Wahrscheinlichkeitsverteilung wird an ausgewählten Stellen repräsentiert (diskretisiert) und zwischen diesen Stellen wird die Verteilung (implizit) interpoliert. Für die Auswahl dieser Repräsentationspartikel, die Art der Interpolation und Propagation stehen verschiedene Verfahren zur Verfügung (siehe z.B. [2]). Sind weitere Nebenbedingungen wie zum Beispiel Stetigkeit erfüllt, dann konvergiert die empirische Verteilung der Partikelfilter gegen die wahre Verteilung.

Dies soll nun dazu genutzt werden, um die Angemessenheit der Annahme einer Normalverteilung des Fehlers zu beurteilen. Sind die Messfehler tatsächlich normalverteilt bzw. lassen sie sich in Annäherung durch eine Normalverteilung darstellen, dann sollte der Partikel- und der Kalman-Filter identische Schätzungen liefern, da dann beide Ansätze eine optimale Kombination der Sensordaten vornehmen. Für den Kalman-Filter wären in diesem Falle eine etwas genauere Schätzungen zu erwarten, da sich beim Partikelfilter durch die Diskretisierung der Verteilung und die implizite Interpolation eine etwas langsamere Konvergenzgeschwindigkeit ergibt.

Werden dagegen die Annahmen wesentlich verletzt, dann sollte die Schätzung durch den Partikelfilter genauer sein als die des Kalman-Filters. Treffen Annahmen der Fehlermodelle nicht zu, dann stellt die Verteilung der Positionsschätzung keine Normalverteilung mehr dar. Es sind dann zum Beispiel multimodale Verteilungen möglich, die einzelnen Modalwerte könnten durch systematische Fehler in den Messungen oder Modellen entstehen. Durch die explizite Modellierung des Zustandes als Wahrscheinlichkeitsverteilung würden die Schätzungen mittels Partikelfilter in diesem Fall genauer als die des Kalman-Filters sein.

Damit steht dann ein quantitatives und objektives Maß zur Verfügung, um die Auswirkungen der Verletzung der Normalverteilungsannahme zu beurteilen.

2 Grundlagen

Beide Verfahren sind statistische Verfahren und fassen die Messwerte als Wahrscheinlichkeitsverteilungen auf. Im Folgenden soll daher kurz der Wahrscheinlichkeitsbegriff dargestellt werden. Danach wird ausgehend von eindimensionalen und mehrdimensionalen (multivariaten) Normalverteilungen die Lösung der optimalen Gewichtung von Messwerten beim Kalman-Filter dargestellt.

Beim Partikelfilter wird statt mit einer parametrisierten Form direkt mit den punktuellen Wahrscheinlichkeitswerten (Partikel-Wahrscheinlichkeiten) gerechnet. Grundlage bildet das Bayes-Theorem, das im Abschnitt 2.3 dargestellt wird. Eine einfache Anwendung stellt die gitterbasierte Lokalisation dar, bei der die Wahrscheinlichkeit ausschließlich für die Punkte eines festen Gitters im Zustandsraum berechnet wird. Die Partikelfilter können (vereinfacht) als Erweiterung dieses Ansatzes betrachtet werden, bei dem sich die Knotenpunkte des Netzes, an denen die Wahrscheinlichkeitsverteilung repräsentiert wird, ständig neu gewählt werden. Die wichtigsten dabei verwendeten Algorithmen werden im Abschnitt 2.4 vorgestellt.

2.1 Wahrscheinlichkeit

Eine Wahrscheinlichkeitstheorie ist ein Axiomensystem, es beschreibt, welche Eigenschaften eine Wahrscheinlichkeit besitzen muss. Zudem sollte eine Wahrscheinlichkeitstheorie Regeln umfassen, um Wahrscheinlichkeiten zu konstruieren und zu interpretieren. Dabei werden neben empirischen auch logische und subjektive Theorien unterschieden [6, S. 1], auf die hier nicht weiter eingegangen werden soll.

Im Folgenden sollen die Axiome von Kolmogoroff verwendet werden, die Aussagen über Wahrscheinlichkeiten auf Aussagen über Mengen zurückführen (nach [4, S. 766]):

- Jeden zufälligen Ereignis E des Ereignisfeldes wird eine nichtnegative Zahl $P(E)$ zugeordnet, die als Wahrscheinlichkeit von E bezeichnet wird.

- Die Wahrscheinlichkeit des sicheren Ereignisses M ist $P(M) = 1$.
- Sind Ereignisse E_1, E_2, \dots, E_n paarweise unvereinbar, so ist

$$P(E_1 \cup E_2 \cup \dots \cup E_n) = P(E_1) + P(E_2) + \dots + P(E_n).$$

Problematisch ist hierbei, dass jedem Ereignis E eine Wahrscheinlichkeit E fest zugeordnet ist. Die Forderung, dass die Wahrscheinlichkeit des sicheren Ereignisses 1 ist, setzt voraus, dass das sichere Ereignis ($\cup E =: M$) bekannt ist. Dies ist z.B. beim Partikelfilter nicht der Fall, es wird nur eine endliche Teilmenge der Ereignisse betrachtet (die Menge der Partikel) und erst durch Normalisierung erreicht, dass die Summe der Wahrscheinlichkeiten über der betrachtete Ereignismenge 1 ist. Die Wahrscheinlichkeit bezieht sich daher nicht mehr auf die der empirischen Ereignisse, sondern nur auf die simulierten Partikel. Es ist daher keineswegs offensichtlich, dass die Partikelfilter gegen die wahre Verteilung konvergieren.

Bei der Betrachtung von Wahrscheinlichkeitsaussagen ist es demzufolge wichtig sicherzustellen, dass Berechnungen über einer Ereignismenge $M := \cup_i E_i$ auch mit der Verteilung über der tatsächlich betrachteten Menge M' übereinstimmen. Dies bedeutet, z.B. im trivialen Fall, dass überprüft werden muss, ob die Normalverteilung, mit der in M gerechnet wird auch in M' vorliegt. Auch für die Partikelfilter müssen Nebenbedingungen erfüllt sein, damit die berechneten Verteilungen über M überhaupt gegen die wahre Verteilung konvergiert.

2.2 Kalman-Filter

Der Kalman-Filter [7] stellt eine effiziente Methode zur Filterung dar, wenn von normalverteilten Fehlern ausgegangen wird und die Parameter der Verteilung a-priori bekannt ist. Weiterhin wird angenommen, dass die Veränderung des Zustandes linear und ebenfalls a-priori bekannt ist. Es wurde gezeigt, dass unter diesen Voraussetzungen der Kalman-Filter die optimale Lösung darstellt [9].

Die Messwerte werden durch Vektoren y dargestellt. Die Menge aller bis zu einem Zeitpunkt t gemessenen Werte stellen dann die multivariate Verteilung $Y|t$ dar, von der angenommen wird, dass sie normalverteilt ist. Die Punktwahrscheinlichkeit eines Messwertes ($P(y)$) kann durch eine Eigenwert-Transformation der multivariaten Verteilung berechnet werden. Dazu wird für jeden Eigenvektor die entsprechende Varianz λ_i aus Zerlegung der Kovarianzmatrix berechnet.

Multivariate Normalverteilung Die Wahrscheinlichkeitsfunktion einer eindimensionalen Messung ist wie folgt definiert:

$$P(y) = \frac{1}{\sqrt{2\pi\sigma^2}} \cdot e^{-\frac{(y-\mu)^2}{2\sigma^2}}$$

Im Folgenden kurz:

$$Y \sim N(\mu, \sigma^2)$$

Im mehrdimensionalen Fall kann man analog formulieren:

$$Y \sim N(\mu, \Sigma)$$

Wobei y den Messvektor, μ den Erwartungswert der Messvektorverteilung Y und Σ die Kovarianzmatrix von Y darstellt.

Die Wahrscheinlichkeit eines beliebigen Punktes y lässt sich mittels Eigenwertzerlegung berechnen:

$$P(\langle y, e_i \rangle) = \frac{1}{\sqrt{2\pi\lambda_i^2}} \cdot e^{-\frac{(\langle y, e_i \rangle - \langle \mu, e_i \rangle)^2}{2\lambda_i^2}}$$

$$P(y) = \prod_{i=1}^n P(\langle y, e_i \rangle)$$

Mahalanobis-Abstand Der Abstand r eines Punktes y von einer multivariaten Normalverteilung X wird meist durch den Mahalanobis-Abstand beschrieben.

Dieser gibt den Abstand des Punktes y vom Erwartungswert $E(X)$ in z-Werten (Vielfache der Standardabweichung) an (nach [5, S. 18]):

$$r^2 = (y - E(X))^T \Sigma^{-1} (y - E(X)) \quad (1)$$

Unabhängigkeitskriterium Beim Kalman-Filter wird ferner gefordert, dass der Erwartungswert des Fehlers dem wahren Wert entspricht, d.h. der Erwartungswert des Fehlers ist 0: ¹

$$E(Y) = E(X) = \mu$$

$$E(\varepsilon) = 0, \quad y = \mu + \varepsilon$$

Weiterhin wird gefordert, dass die einzelnen Messungen voneinander unabhängig sind:

$$E(\text{cov}(y_i, y_j)) = 0, \quad \forall i \neq j$$

Kovarianz bei konstanter Fehlerverteilung Durch diese Unabhängigkeitskriterien kann die Varianz der Verteilung Y einfach aus den Varianzen von beliebigen, disjunkten Teilmengen Y_1 und Y_2 berechnet werden. Für den speziellen Fall, dass die Kovarianz-Matrix konstant ist, reduziert sich der Kalman-Filter auf die Berechnung des Mittelwertes:

$$\hat{x} = E(Y|t) = \frac{1}{t} \sum_{i=1}^t y_i$$

Die Kovarianz des Schätzfehlers ist dabei eine Funktion der Anzahl der

¹Im folgenden soll der wahre Wert (die tatsächliche Pose) als x bezeichnet werden. \hat{x} kennzeichnet den Schätzer für x . Da eine Normalverteilung der Messwerte Y angenommen wird, wird x am besten durch den Mittelwert der Messwertverteilung Y geschätzt.

Messwerte t :

$$\hat{X} \sim N \left(E(Y), \frac{\Sigma}{t} \right) = N \left(E(X), \frac{\Sigma}{t} \right) = N \left(\mu, \frac{\Sigma}{t} \right)$$

Die Idee des Kalman-Filters ist, diese Berechnung schrittweise vorzunehmen. Für eine konstante Fehlerverteilung ergibt sich dann:

$$\underbrace{\hat{x}|t+1}_{a\text{-posteriori}} = \underbrace{\hat{x}|t}_{a\text{-priori}} + \underbrace{\frac{t}{t+1}}_{\text{Gewichtung}} \cdot \underbrace{(y|t+1 - \mu)}_{\text{Abweichung}}$$

Statt μ wird dabei der verfügbare Schätzer $\hat{x}|t$ eingesetzt:

$$\hat{x}|t+1 = \hat{x}|t + \frac{t}{t+1} \cdot (y|t+1 - \hat{x}|t)$$

Gewichtung bei variabler Fehlerverteilung Sind die Parameter der Fehlerverteilung nicht konstant, so müssen zur optimalen Filterung die einzelnen Messwerte gewichtet werden. Die optimale Gewichtung ist dann (siehe z.B. [3]):

$$\hat{x}|t+1 = \hat{x}|t + \underbrace{\hat{\Sigma}_{\hat{x}}|t \cdot \left(\hat{\Sigma}_{\hat{x}}|t + \hat{\Sigma}_Y|t+1 \right)^{-1}}_{=B} \cdot (y|t+1 - \hat{x}|t) \quad (2)$$

B wird häufig als Kalman-Gain-Matrix bezeichnet. Die Kovarianzmatrix des Schätzers $\hat{\Sigma}_{\hat{x}}$ kann mit Hilfe von B wie folgt geschätzt werden:

$$\hat{\Sigma}_{\hat{x}}|t+1 = (I - B) \cdot \hat{\Sigma}_{\hat{x}}|t = \left(1 - \hat{\Sigma}_{\hat{x}}|t \left(\hat{\Sigma}_{\hat{x}}|t + \hat{\Sigma}_Y|t+1 \right)^{-1} \right) \cdot \hat{\Sigma}_{\hat{x}}|t \quad (3)$$

Lineare Propagation Zustandsveränderungen müssen beim Kalman-Filter linear beschrieben werden. Bei linearer Transformation des Zustandes ändert sich die Form der Verteilung nicht.

$$\hat{X}|t+1 = A \hat{X}|t \quad (4)$$

Die Transformation des Schätzfehlers ist dann bilinear und kann mit der Propagationsmatrix A berechnet werden:

$$\hat{\Sigma}_{\hat{x}}|t+1 = A \hat{\Sigma}_{\hat{x}}|t A^T \quad (5)$$

Implementierung Die Berechnung des Kalman-Filters kann in drei Gruppen eingeteilt werden (siehe Abbildung 1). Die Initialisierung (0.) erfolgt mit dem ersten Messwert und dessen Kovarianzmatrix:

$$\hat{x}|t=0 \leftarrow y|t=0$$

$$\hat{\Sigma}_{\hat{x}}|t=0 \leftarrow \Sigma_y|t=0$$

Für die Berechnung der neuen Verteilung (1.) wurden die Gleichungen (4) und (5) verwendet. Die Transformationsmatrix wurde von der Steuerungssoftware des Roboters zur Verfügung gestellt [10].

Nach jedem Propagationsschritt wurde überprüft, ob eine Positionsschätzung der omnidirektionalen Kamera vorliegt. War dies der Fall, dann wurde anhand des Fehlermodells (siehe Abschnitt 4) die neue a-posteriori-Wahrscheinlichkeit (2.) berechnet. Die Kalman-Gain-Matrix B wurde berechnet (Gleichung 2), \hat{x} und $\hat{\Sigma}_{\hat{x}}$ aktualisiert (Gleichung 3). Dabei wurde für die aktuelle Position die Kovarianzmatrix der Messwertverteilung $\Sigma_Y|y_t$ geschätzt (siehe Abschnitt 4).

2.3 Bayes-Theorem

Das Bayes-Theorem beschreibt die Beziehung zwischen den bedingten Wahrscheinlichkeiten zweier Ereignismengen x und y . Es erlaubt die Berechnung der bedingten Wahrscheinlichkeit $P(x|y)$ aus $P(y|x)$ und den Grundwahrscheinlichkeiten $P(x)$ und $P(y)$.

$$P(x|y) = P(y|x) \cdot P(x) \cdot P(y)^{-1} \quad (6)$$

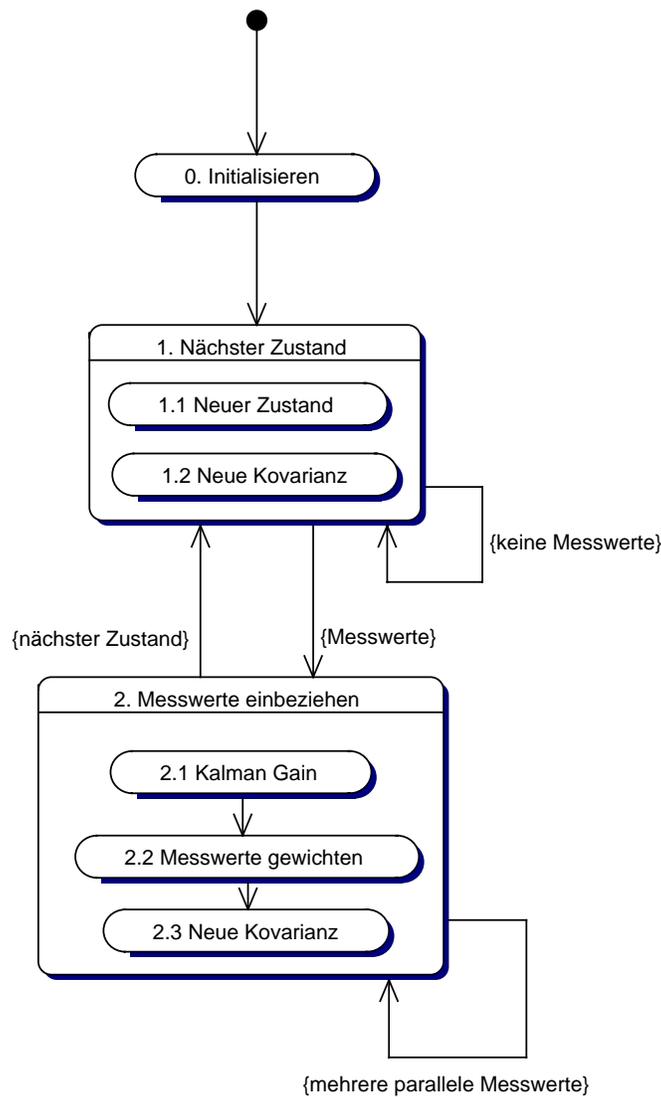


Abbildung 1: UML Zustandsdiagramm der Berechnungsschritte des Kalman-Filters. Die Initialisierung (0.) erfolgte mit dem ersten Messwert. Für die Propagation des Zustandes (1.) wurde das Modell der Steuerungssoftware übernommen [10]. Die Berechnung der a-posteriori-Wahrscheinlichkeit (2.) wurde mit dem in Abschnitt 4 gewonnenen, positionsabhängigen Parametern durchgeführt.

Beweis:

$$\begin{aligned}
 P(x|y) &= P(x \cup y) \cdot P(x) \\
 \Leftrightarrow P(x|y) \cdot P(x)^{-1} &= P(x \cup y) \\
 P(y|x) &= P(y \cup x) \cdot P(y) \\
 \Leftrightarrow P(y|x) \cdot P(y)^{-1} &= P(y \cup x) \\
 \hline
 \Rightarrow P(x|y) \cdot P(x)^{-1} &= P(y|x) \cdot P(y)^{-1} \\
 \Leftrightarrow P(x|y) &= P(y|x) \cdot P(x) \cdot P(y)^{-1}
 \end{aligned}$$

Markov-Ketten Die Markov-Ketten sind eine einfache Anwendung des Bayes-Theorems. Dabei wird fortlaufend die Wahrscheinlichkeit für ausgewählte x_i anhand der a-priori-Wahrscheinlichkeiten für diese Werte $P(x_i)$ und die (bedingte) Wahrscheinlichkeit für eine Messung $P(y|x_i)$ berechnet.

$$P(\hat{x}|Y) = \int P(y|\hat{x}) \cdot P(\hat{x}) dy$$

Für die sequentielle Berechnung ergibt sich folgende Formel:

$$P(\hat{x}_{t+1}) = P(y_{t+1}|\hat{x}_t) \cdot P(\hat{x}_t)$$

Ein Beispiel ist die gitterbasierte Monte-Carlo-Lokalisierungsmethode. \hat{X} kann dabei die Punkte eines festgelegten Gitters (\hat{x}_i) annehmen. Für jeden Gitterpunkt wird dann fortlaufend die Wahrscheinlichkeit berechnet, dass der wahre Wert dem Gitterpunkt x_i entspricht.

$$P(\hat{x}_{i,t+1}) = P(y_{t+1}|\hat{x}_{i,t}) \cdot P(\hat{x}_{i,t}) \quad (7)$$

2.4 Partikelfilter

Die gitterbasierte Lokalisation hat den Nachteil, dass viele Gitterpunkte berechnet werden müssen, die eventuell eine sehr geringe Wahrscheinlichkeit besitzen und der Rechenaufwand mit der Auflösung des Gitters enorm ansteigt ($O(n^{dim})$).

Partikelfilter betrachten dagegen nicht den gesamten Zustandsraum X , sondern berechnen die Wahrscheinlichkeitsverteilung nur für eine Teilmenge $\hat{X} \subset X$. Die Auswahl Teilmenge $\hat{X}|t$ ist selbst wieder eine Zufallsgröße. In der Art und Weise wie diese Teilmengen generiert werden, unterscheiden sich die einzelnen Ansätze zum Teil beträchtlich.

Die Präzision der Schätzung kann erhöht werden, wenn vor allem die Punkte mit hoher Wahrscheinlichkeit berücksichtigt werden und wenn sich die a-posteriori-Wahrscheinlichkeit nicht stark verändert, d.h. die Messwerte nicht stark von der erwarteten Verteilung abweichen [8].

Sampling Als *Sampling* bezeichnet man die Auswahl der betrachteten Zustandsmenge \hat{X} , den Partikeln, aus dem Zustandsraum X . Diese Auswahl wird bei jeder Integration von Sensordaten oder bei der Propagation des Zustandsraumes neu berechnet und dient bei den meisten Partikelfilter-Algorithmen gleichzeitig zur Modellierung dieser Prozesse.

Sampling/importance resampling (SIR) Diese Methode geht auf Rubin ([11] und [12]) zurück. Sie setzt voraus, dass aus der a-priori-Verteilung $P(\hat{X})$ gezogen werden kann und die Likelihood-Funktion $P(Y|\hat{x})$ ausgewertet werden kann.

Die Idee ist, aus der Verteilung $P(\hat{X})$ mit einer Wahrscheinlichkeit π_i zu ziehen, wobei π_i proportional zur Likelihood des Messwertes y ist:

$$\begin{aligned}\omega_i &= P(y|\hat{x}_i) \\ \pi_i &= \frac{\omega_i}{\sum_{i=1}^N \omega_i}\end{aligned}\tag{8}$$

Mit steigender Anzahl der *Samples* N konvergiert $P(\hat{X}|y)$ gegen eine Teilmenge aus $P(y)$. Diese Teilmenge ist statistisch abhängig, da statt eines Integrals die diskrete Summe $\sum_i \omega_i$ zur Normalisierung verwendet wird. Nach [8] ist die Varianz, unter der Voraussetzung, dass sich die Wahrscheinlich-

keitsschätzung $P(\hat{X})$ nicht stark mit \hat{X} ändert, proportional zu:

$$\frac{E\left(\pi(\hat{X})^2\right)}{N} \quad (9)$$

Der SIR-Algorithmus ist also wenig robust in Bezug auf Ausreißer und stark gipflige Verteilungen, da dann $\left|\frac{P(\hat{X})}{d\hat{X}}\right|$ recht gross werden kann.

Rejection Sampling Beim *Rejection sampling* wird aus $P(\hat{X})$ gezogen und das *Sample* mit einer Wahrscheinlichkeit von $\pi(\hat{x})$ akzeptiert:

$$\pi(\hat{x}) = \frac{P(y|\hat{x})}{P(y|\hat{x}_{max})}$$

$$\hat{x}_{max} = \arg \max_{\hat{x}} P(y|\hat{x})$$

Der Vorteil dieser Methode gegenüber SIR ist die Zufälligkeit der gezogenen a-posteriori Partikel. Problematisch kann sich die Berechnung von \hat{x}_{max} und eine hohe Ablehnungsrate bei stark gipfligen Verteilungen erweisen.

Stochastische Propagation Eine beliebige Zustandsänderung a kann als Wahrscheinlichkeitsverteilung $P(X_{t+1}|X_t, a)$ modelliert werden. Für die Propagation des Zustandsraumes ergibt sich aus dem Bayes-Theorem (6) folgende Formel, die in Markov-Ketten verwendet wird ([1]):

$$P(\hat{x}_{t+1}) = \int_{\hat{x}_t} P(\hat{x}_{t+1}|\hat{x}_t, a_{t+1}) \cdot P(\hat{x}_t) d\hat{x}_t \quad (10)$$

Um dieses statistische Modell zu simulieren, wäre es nicht sinnvoll, wie im vorigen Abschnitt beschrieben aus $P(\hat{X})$ mit einer Wahrscheinlichkeit von $P(a|\hat{X})$ neue Partikel zu ziehen. Denn wenn sich der Roboter bewegt, dann wird sich $P(\hat{X}|t+1)$ in aller Regel stark von $P(\hat{X}|t)$ unterscheiden. Stattdessen ist es günstiger die Wahrscheinlichkeit der Zustandsänderung $P(dx) = P((\hat{x} + dx) - \hat{x})$ zu betrachten.

$P(\hat{X}|t + 1)$ erhält man dann durch ziehen aus $P(\hat{X}|t)$ und $P(dx)$, wobei $x_{t+1} = x_t + dx$ ($\hat{x} \in \hat{X}|t, \hat{x}|t + 1 \in \hat{X}|t + 1$).

Implementierung Die Berechnung des Partikelfilters erfolgt ebenso wie die des Kalman-Filters in drei Schritten (siehe Abbildung 2).

Der Partikelfilter wurde mit 1000 zufälligen Partikeln initialisiert (0.), denen die gleiche Wahrscheinlichkeit zugewiesen wurde.

Die Propagation der Wahrscheinlichkeitsverteilung erfolgte mit demselben linearen Modell, das auch für den Kalman-Filter verwendet wurde und von der Ansteuerungssoftware berechnet wird [10]. Während beim Kalman-Filter direkt mit der Transformationsmatrix A gerechnet wurde, muss beim Partikelfilter aus der Verteilung $N(a, \Sigma_a)$ gezogen werden (siehe Abschnitt 2.4). Dies wurde mittels *rejection sampling* implementiert.

Für die Berechnung der a-posteriori-Wahrscheinlichkeit (2.) wurde die SIR-Methode verwendet.

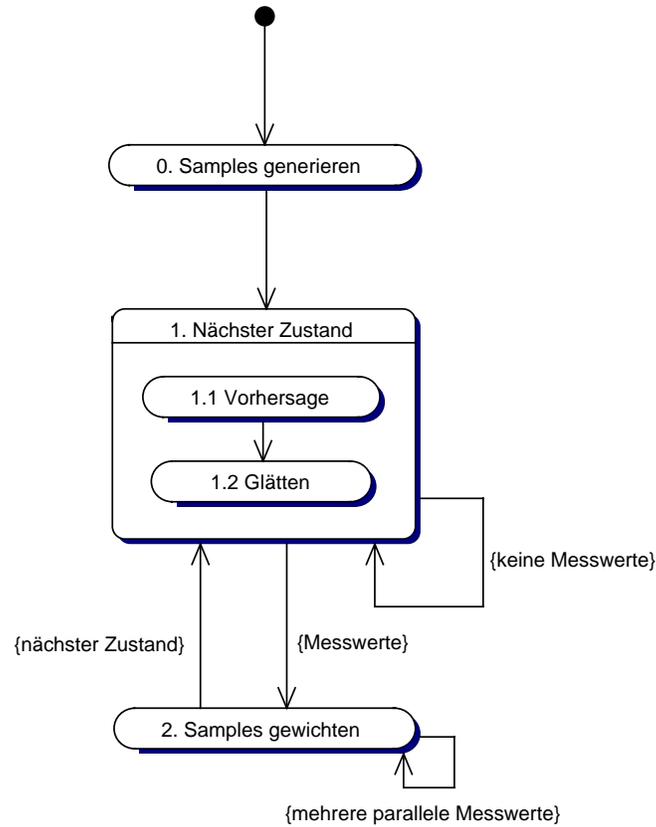


Abbildung 2: UML Zustandsdiagramm der Berechnungsschritte des Partikelfilters. Die Initialisierung (0.) erfolgte mit einer Gleichverteilung. Für die Propagation des Zustandes (1.) wurde das lineare Modell der Steuerungssoftware übernommen [10]. Die Berechnung der a-posteriori-Wahrscheinlichkeit (2.) wurde mit dem in Abschnitt 4 gewonnenen, positionsabhängigen Parametern durchgeführt.

3 Implementierung des Partikelfilters

Für einen Partikelfilter-Algorithmus ist neben den schon beschriebenen Sampling/Resampling-Algorithmen die effiziente Erzeugung von multivariaten Zufallsgrößen notwendig. Betrachten wir zum Beispiel den Propagationsschritt. Ausgehend von einem Partikel $\hat{x}|t$ wird ein neuer Zustand $\hat{x}|t+1$ aus der a-posteriori-Verteilung $P(Y|\hat{x}) \cdot P(\hat{x})$ gezogen:

$$P(\hat{x}|t+1) = P(Y|\hat{x}) \cdot P(\hat{x})$$

In der Implementierung für die Selbstlokalisierung wurde die Propagation durch Normalverteilungen modelliert (siehe Abschnitt 4):

$$X|t+1 = X|t + N(\Delta x, \Sigma) = X|t + \Delta x + N(0, \Sigma)$$

Um eine multivariat normalverteilte Zufallsvariable $X \sim N(0, \Sigma)$ zu erzeugen, bestimmt man als erstes den Eigenraum von Σ . Dazu wurde eine Eigenwertzerlegung ("single value decomposition", svd) benutzt.

$$\Sigma = U S V^t$$

S ist eine Diagonalmatrix mit den Eigenwerten λ_i , $\det(U)$ und $\det(V)$ haben jeweils den Betrag 1.

Da die Kovarianzmatrix Σ symmetrisch ist, gilt: $U = V$. Im Eigenraum ist die Kovarianzmatrix der Zufallsverteilung die Diagonalmatrix S . U^t transformiert die Zufallsverteilung X in den entsprechenden Eigenraum:

$$E(U^t X U^t X) = E(U X X^t U) = U^t \Sigma U = U^t U S V^t U = S$$

3.1 Gewichtung

3.2 Resampling

4 Fehlermodelle

Einen Schwerpunkt dieser Arbeit bildete die Gewinnung eines angemessenen Fehlermodells für die Selbstlokalisierung. Ein gängiges Verfahren ist hierbei, Annahmen über die Ungenauigkeit der Kameraauflösung (1-2 Pixel) und die Entfernung zu berücksichtigen und dann eine (Standard)Fehlerverteilung zu skalieren. Dabei wird die Varianz der (Standard)Fehlerverteilung meist so optimiert, dass für die Selbstlokalisierung ein möglichst optimaler Kompromiss zwischen Robustheit und Genauigkeit hergestellt wird.

Diese heuristische Herangehensweise hat den Vorteil, dass ungenau, falsch oder nicht modellierte Komponenten zur Selbstlokalisierung benutzten Sensorsysteme ausgeglichen werden können. Die tatsächliche Fehlerverteilung wird dagegen kaum gemessen. Abbildung 3 zeigt das Fehler-Histogramm an verschiedenen Positionen. Dort zeigt sich, dass die Fehler nicht normalverteilt sind und dass die Varianz nicht mit der Entfernung ansteigt oder abfällt.

Die Fragestellung der Arbeit war jedoch nicht, ob eine einfache (linear) Beziehung zwischen den Fehlerverteilungen vorliegt, sondern ob die Verteilungen selbst ohne Einbußen der Robustheit oder Genauigkeit des Verfahrens durch Normalverteilungen approximiert werden können.

Dazu wurde an den Punkten eines 1x1-Meter-Gitters die Position mit der omnidirektionalen Kamera für ca. 1 Minute gemessen und daraus die Kovarianzmatrix berechnet. Zwischen diesen Punkten wurde dann bilinear interpoliert, so dass für jeden Punkt innerhalb des Gitters die Fehler-Kovarianzmatrix geschätzt werden kann.

4.1 Kovarianzberechnung

Die Varianz ist definiert als der Erwartungswert der quadrierten Abweichung vom Erwartungswert einer Verteilung X :

$$\sigma^2(X) := E \left[(X - E(X))^2 \right]$$

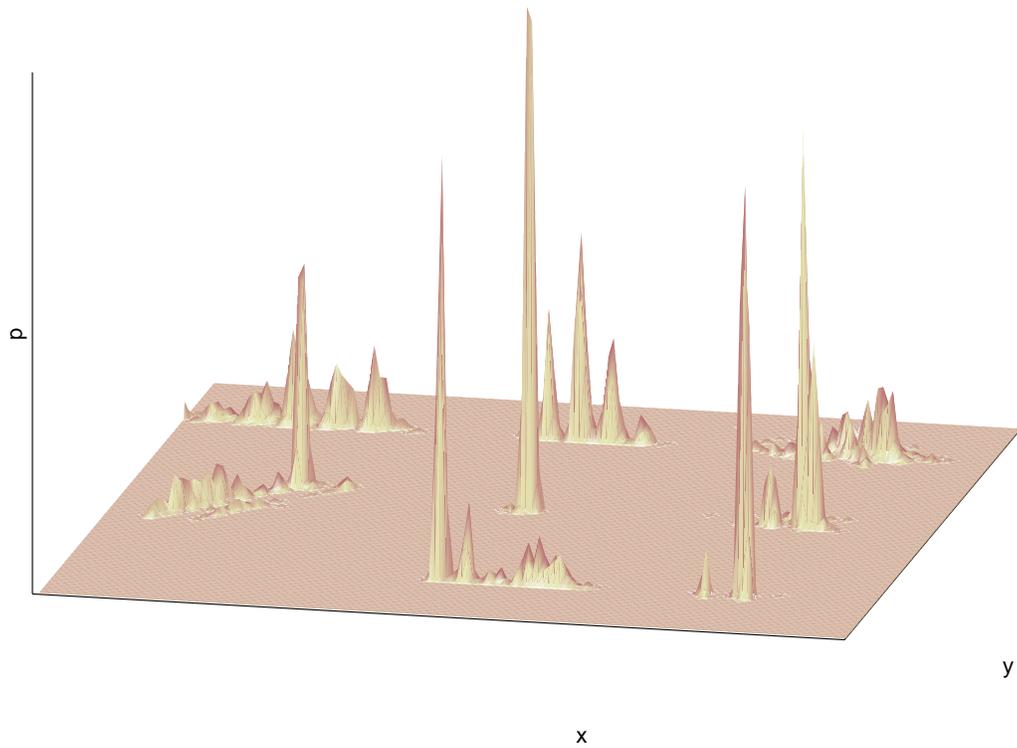


Abbildung 3: Histogram der Positionsschätzung an den Punkten des 1x1-Meter-Gitters. Die Abweichungen von der tatsächlichen Position (Messfehler) sind vergrößert dargestellt.

Im multivariaten Fall entsprechend die Kovarianz:

$$\text{cov}(X, Y) = E[(X - E(X)) \cdot (Y - E(Y))]$$

Durch ausmultiplizieren erhält man ein einfache Berechnungsformel:

$$\begin{aligned} \text{cov}(X, Y) &= E[(X - E(X)) \cdot (Y - E(Y))] \\ &= E[XY] - E[X \cdot E(Y)] - E[Y \cdot E(X)] + E(X) \cdot E(Y) \\ &= E[XY] - 2 \cdot E(X) \cdot E(Y) + E(X) \cdot E(Y) \\ &= E[XY] - E(X) \cdot E(Y) \end{aligned}$$

Die Kovarianzmatrix zweier Variablen kann demnach recht einfach durch Summen berechnet werden:

$$\text{cov}(X, Y) = \frac{1}{n} \sum_{t=1}^n x_t \cdot y_t - \frac{1}{n} \sum_{t=1}^n x_t \cdot \frac{1}{n} \sum_{t=1}^n y_t$$

Für den mehrdimensionalen Fall entsprechend:

$$\Sigma_X = \frac{1}{n} \sum_{t=1}^n x_t \cdot x_t^T - \left(\frac{1}{n} \sum_{t=1}^n x_t \right) \cdot \left(\frac{1}{n} \sum_{t=1}^n x_t \right)^T \quad (11)$$

4.2 Fehlerkovarianz der Pose-Bestimmung

Die Fehlerverteilung der Positionsschätzungen wurde für die Kreuzungspunkte des 1x1-Meter-Gitters bestimmt. Dazu wurde der Roboter auf die entsprechenden Koordinaten gestellt und auf 180° ausgerichtet, so dass der Laser-Scanner auf das Tor zeigt. An der vorderen linken Position (1, 1) konnte keine Positionsbestimmung mit der omnidirektionalen Kamera durchgeführt werden, da der durch die Tiefe des Tores nur ein kleiner Teil der hinteren farbigen Torwand detektiert wird.

Anschließend wurde mit einer Sampling-Rate von 20 Hz die Messwerte des Laser-Scanners, der Odometrie und der omnidirektionalen Kamera aufgenommen. Zusätzlich wurden für die Positionsschätzung mittels omnidirektionalen Ka-

mera die Länge der erkannten Torlinie und der Öffnungswinkel protokolliert. Sie wurden jedoch nicht weiter ausgewertet.

Interpolation der Kovarianz Gemessen wurden die Verteilung der Fehler der Positionsschätzung jeweils für die Kreuzungspunkte eines Rasters mit 1m Gitterabstand. Zwischen diesen Punkten wurden die Elemente der Kovarianz-Matrix bilinear interpoliert. Abbildung 4b) zeigt das Ergebnis der Interpolation für die Summe der x- und y-Standardabweichungen. Vor dem Tor ist der Fehlervarianz gering und steigt auch mit der Entfernung nur relativ leicht an. Rechts vom Tor ist der Standardfehler jedoch wesentlich größer, da hier ein Teil der nach hinten versetzten Torwand nicht sichtbar ist.

4.3 Odometrie-Fehlerkovarianzmodell

Für den Kalman-Filter wurden die Parameter des Odometrie-Fehlermodells gemessen statt die theoretisch gewonnenen Kovarianz-Matrizen der Robotersoftware zu verwenden.

Messung Zur Bestimmung der Fehlerverteilung der Odometrie wurden die Daten einer Testfahrt verwendet. Für den RoboCup ist es wichtig, ein konservatives Fehlermodell zu verwenden, da Extremsituationen wie schnelles Beschleunigen (z.B. bei Kollisionen) recht häufig auftreten. Die Testfahrt wurde manuell gesteuert und enthält starke Beschleunigungen, schnelle Richtungsänderungen und eine Passage mit Rückwärtsfahren.

Von der Odometrie wurde diese Testfahrt als glatte Trajektorien gemessen (Abbildung 5). Die Positionsschätzung mittels Laser-Scanner war jedoch stark fehlerbehaftet (Abbildung 6). Deshalb wurden vor der weiteren Verwendung 10% Messwerte mit dem größten Mahalanobis-Abstand herausgefiltert, um die entfernten Fehlmessungen (Ausreißer) zu entfernen. In Abbildung 6 sind diese Punkte rot dargestellt. Zusätzlich wurde die Daten zwischen der 20. und 25. Sekunde ausgeschlossen, da der Laser-Scanner in der unteren rechten Spielfeldecke z.T.

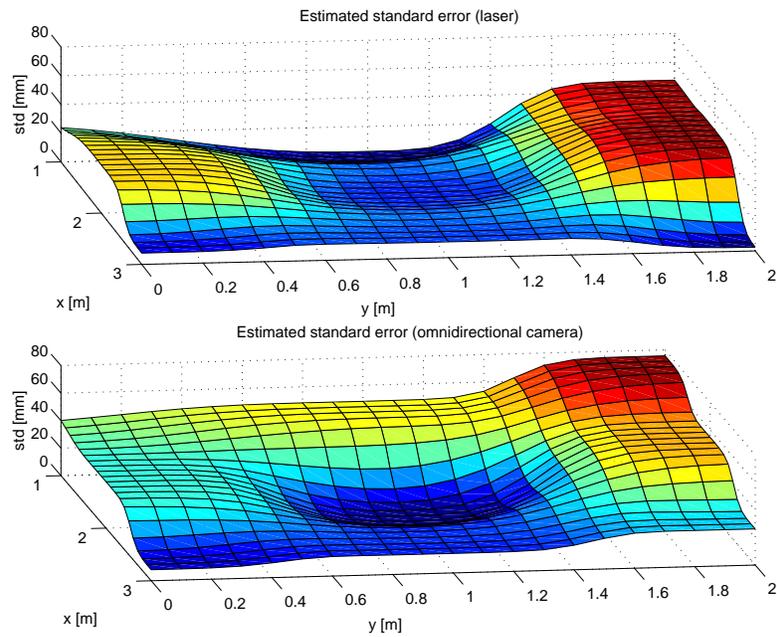


Abbildung 4: Summe der Standardschätzfehler in Abhängigkeit von der Position (interpoliert). Dargestellt ist die Summe der Standardschätzfehler der Pose-Schätzung für den Laser-Scanner (a) und die omnidirektionale Kamera (b). Zwischen den Punkten des 1x1-Meter-Gitters an denen die Kovarianzmatritzen gemessen wurde, wurde der Standardmessfehler interpoliert.

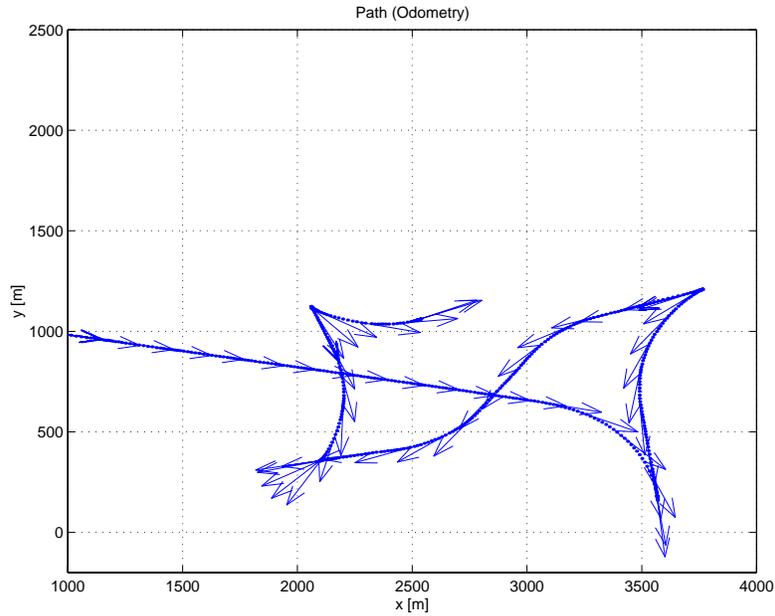


Abbildung 5: Pfad der Testfahrt (Odometrie). Dargestellt ist der Pfad der Testfahrt, der aus den Pose-Daten der Odometrie rekonstruiert wurde.

(konsistent) falsche Daten lieferte. In Abbildung 7 ist der y-Koordinate des Roboters während der Testfahrt dargestellt, man erkennt deutlich die Fehlmessungen ab der 19. und 24. Sekunde.

Berechnung der Kovarianz Für die Schätzung der Kovarianzmatrix wurden für die verbleibenden Messwerte jeweils die Differenz zwischen der Schätzung der Odometrie und des Laser-Scanners bestimmt:

$$\varepsilon_x = x_{odo} - x_{laser}$$

Anschließend wurde die Kovarianz zwischen diesen Werten für verschiedene Zeitdifferenzen berechnet:

$$cov_{dt} = cov(\varepsilon_x|t, \varepsilon_x|t + dt)$$

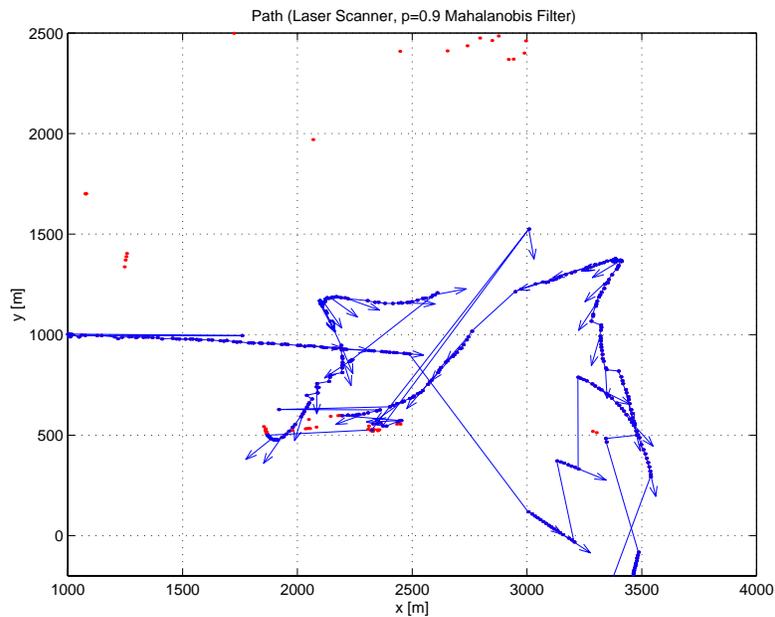


Abbildung 6: Pfad der Testfahrt (Laser-Positionsschätzung). Dargestellt ist der Pfad der Testfahrt, der aus den Daten der Laser-Selbstlokalisierung rekonstruiert wurde. 10% der Messwerte mit dem größten Mahalanobis-Abstand vom Gesamtmittelwert der Pose wurden von der weiteren Analyse ausgeschlossen, sie sind als einzelne Punkte dargestellt.

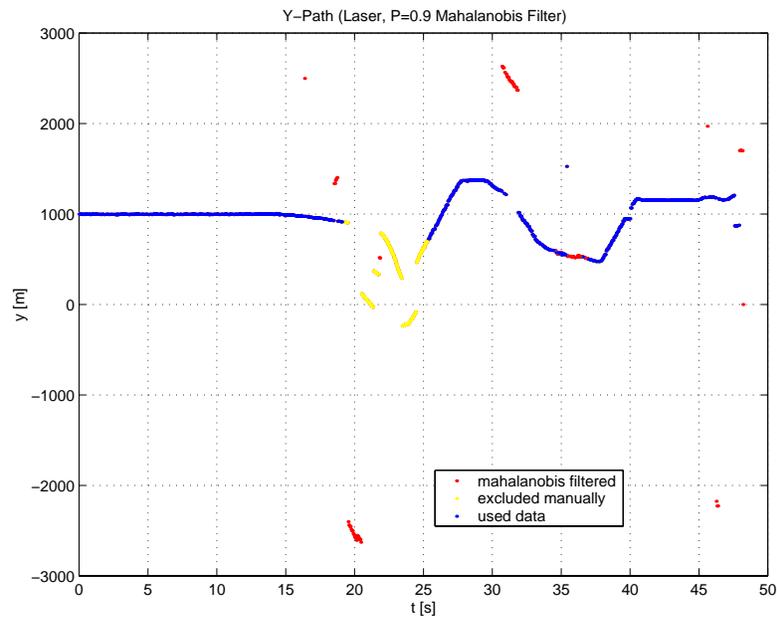


Abbildung 7: y-Position der Testfahrt (Laser-Positionsschätzung). Dargestellt ist geschätzte Position in y-Richtung während der Testfahrt. 10% der Messwerte mit dem größten Mahalanobis-Abstand vom Gesamtmittelwert der Pose-Schätzung sind als einzelne Punkte dargestellt und wurden wie die Messwerte zwischen der 19. und 24. Sekunde von der weiteren Analyse ausgeschlossen.

Die cov_{dt} gibt die durchschnittliche Kovariation zwischen dem Fehler einer Messung und einer zweiten, um dt versetzten Messung an. Die Kovarianz dieser Differenz ε_x ist, bei angenommener Unabhängigkeit der Fehler, gleich der Summe der Kovarianzen der Einzelmessgrößen:

$$\Sigma_{\varepsilon_x} = \Sigma_{odo} + \Sigma_{laser}$$

Für die Berechnung von Σ_{odo} wurde die Kovarianzmatrix des Laserscanner-Fehlers Σ_{laser} von der Position (2, 2) verwendet (siehe Abbildung 4a).

$$\Sigma_{odo} = \Sigma_{\varepsilon_x} - \Sigma_{laser}$$

Diese Korrektur wurde für alle dt durchgeführt.

Lineare Regression Diese Fehlerverteilung wurde nun als linear Funktion in Abhängigkeit von beliebigen Zeitdifferenzen dt betrachtet:

$$\Sigma|dt = B dt + A$$

Gesucht ist allerdings nicht ein Modell in Abhängigkeit von der Zeit. Der Fehler der Odometrie hängt nicht von der Zeitdifferenz zwischen zwei Messungen, sondern von dem zurückgelegten Weg ab. Da der Roboter während der Testfahrt ständig in Bewegung war, ist der zurückgelegte Weg stark mit der Zeit korreliert. Daher wurde der Weg als Funktion der Zeitdifferenz aufgefasst und der Erwartungswert des Weges $E(s|dt)$ für die Regression verwendet:

$$\Sigma|ds = B ds + A$$

$$s = E(s|dt)$$

Der zurückgelegte Weg wurde mittels euklidischer Metrik über den x- und y-Koordinaten berechnet. Um den Einfluss der Geschwindigkeitsvarianz (und damit

der Wegvarianz) zu minimieren, wurden nur diejenigen Messpaare $(x|t, x|t + dt)$ verwendet, bei denen der zurückgelegte Weg nicht allzu stark vom jeweiligen Erwartungswert $E(s|dt)$ abweicht. Dieses Intervall wurde so gewählt dass für jeden Zeitdifferenz dt 50% der Messwerte der Testfahrt verwendet wurden (siehe Erläuterung des “Mahalanobis-Filters” im vorangegangenen Abschnitt).

Trotz des extremen Charakters der Testfahrt zeigt sich ein deutlicher linearer Zusammenhang zwischen der zurückgelegten Strecke $E(x_{laser}|t + dt - x_{laser}|t)$ und den bedingten Kovarianzen der Fehler $\Sigma|dt$ (siehe Abbildung 8).

Die Koeffizienten der linearen Funktion von dx auf Σ wurden mittels linearer Regression bestimmt. Im Partikelfilter wurde dann für jeden Messzeitpunkt t bestimmt, wie groß die zurückgelegte Strecke seit der letzten Messung war und dann die Kovarianzmatrix $\Sigma|s = B s + A$ für den Propagationsschritt verwendet.

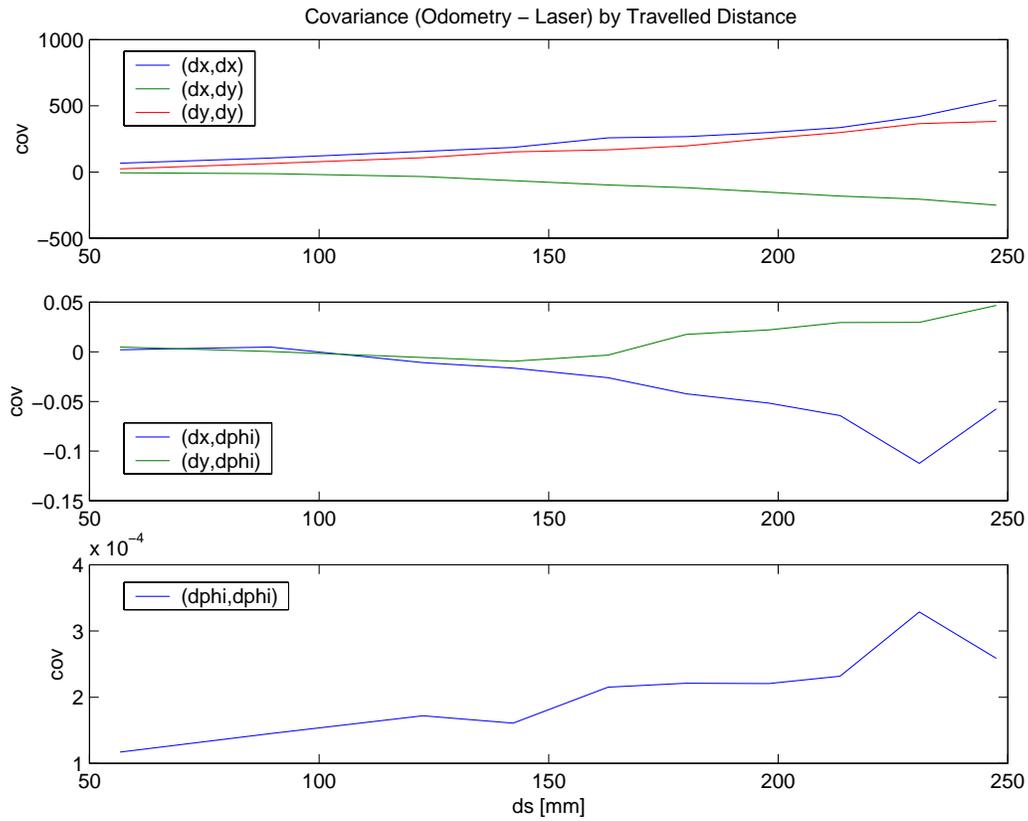


Abbildung 8: Abhängigkeit der Koeffizienten der Fehlerkovarianzmatrix der Odometriefehler von der durchschnittlich zurückgelegten Wegstrecke. Erläuterungen im Text.

5 Evaluation

Die empirisch bestimmten Kovarianzmatrizen stellen eine optimale Parametrisierung der Fehlerverteilungen für die stationäre Selbstlokalisierung und die Testfahrt dar. Es ist daher zu erwarten, dass der Kalman- und der Partikelfilter optimale Vorhersagen der tatsächlichen Position schätzen sollten.

Im folgenden Abschnitt werden nun die gefilterten Positionsschätzungen mit den ursprünglichen, verrauschten Messwerten verglichen. Dies soll beispielhaft an für die stationäre Selbstlokalisierung an der Position (1,1) erfolgen.

Anschließend werden die Ergebnisse für die aufgezeichnete Testfahrt und ein “kidnap robot”-Sequenz vorgestellt.

Stationäre Selbstlokalisierung Die Filter-Algorithmen wurden auf die Originalmessreihen, die an den Gitterpunkten aufgenommen wurden, angewendet. Da die Kovarianz-Matrizen auch aus diesen Daten geschätzt wurden (allerdings wurden die Ausreisser entfernt), überschätzen die Simulationsergebnisse vermutlich die Genauigkeit der Algorithmen.

Wahlweise seien in den Abbildungen 9-11 die Filterung der Zeitreihen für die Positionen (1,1) dargestellt. Man erkennt, dass beide Filter innerhalb von 20 s (400 Iterationen) konvergieren. Die geringere Differenz zwischen dem Schätzer und den wahren Wert beim Kalman-Filter resultiert aus der Tatsache, dass der Kalman-Filter den ersten Messwert als Startpunkt benutzt, der Partikelfilter jedoch von einer gleichförmigen Verteilung ausgeht. Allerdings stimmt die Konvergenzgeschwindigkeit, wie theoretisch zu erwarten, zwischen beiden Filtern in etwa überein. Würde man den Partikelfilter mit $P(\hat{X}|y_0)$ starten, würde man identische Differenzen erwarten.

Deutlich sichtbar ist auch die erhöhte Varianz des Partikelfilters, die durch die begrenzte Anzahl der *Samples* bedingt ist.

Testfahrt Bei der Testfahrt zeigt sich, dass der Kalman-Filter nicht geeignet ist, große Diskrepanzen zwischen den Messwerten und der Schätzung zu integrieren

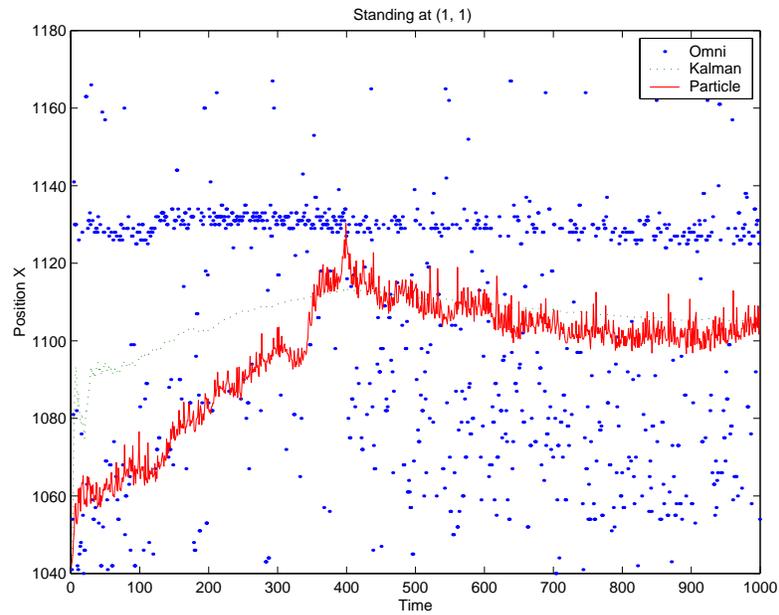


Abbildung 9: x-Position der stationären Pose-Schätzung mittels omnidirektionaler Kamera an Gitterposition (1,1). Dargestellt sind die originalen Positionsschätzungen und die Schätzungen durch den Kalman- und den Partikelfilter.

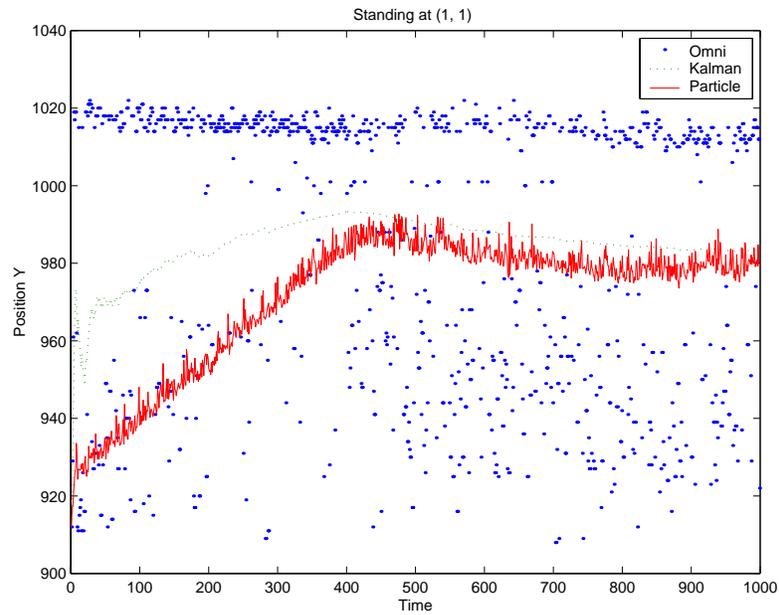


Abbildung 10: y-Position der stationären Pose-Schätzung mittels omnidirektionaler Kamera an Gitterposition (1,1). Dargestellt sind die originalen Positionsschätzungen und die Schätzungen durch den Kalman- und den Partikelfilter.

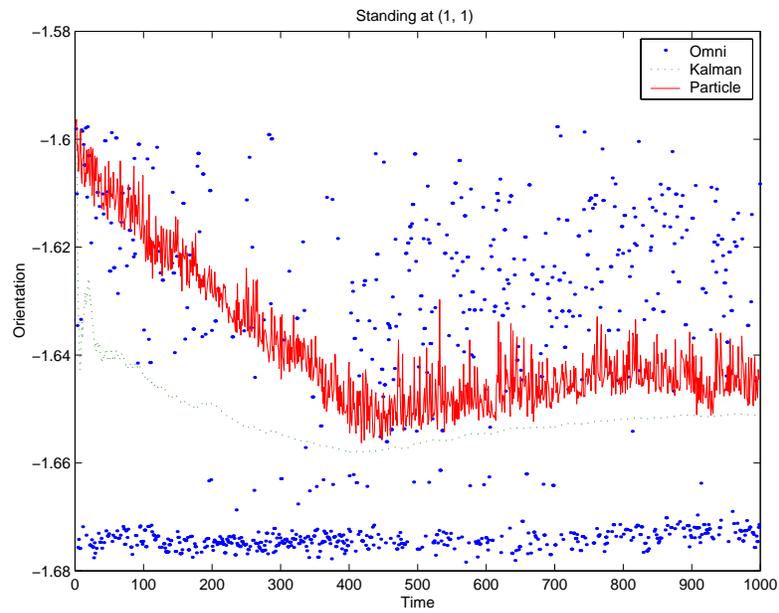


Abbildung 11: θ -Komponente (Orientierung) der stationären Pose-Schätzung mittels omnidirektionaler Kamera an Gitterposition (1,1). Dargestellt sind die originalen Orientierungsschätzungen und die Schätzungen durch den Kalman- und den Partikelfilter.

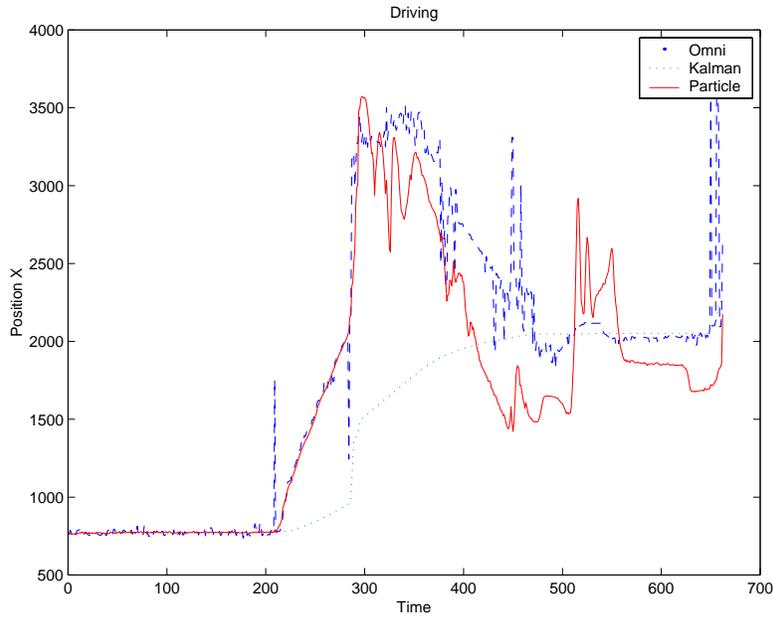


Abbildung 12: x-Position der Pose-Schätzung mittels omnidirektionaler Kamera während der Testfahrt. Dargestellt sind die originalen Positionsschätzungen und die Schätzungen durch den Kalman- und den Partikelfilter in Abhängigkeit von der Zeit. Eine Zeiteinheit entspricht 50 ms.

(siehe Abbildung 12). Die Abweichungen betragen bis zu 2 Metern, wohingegen der größte Fehler beim Partikelfilter ca. 50 cm beträgt.

Kidnap Robot Gänzlich ungeeignet ist der Kalman-Filter, wenn die Propagation des Zustandes (z.B. durch Fehler der Odometrie-Daten) nicht möglich ist (siehe Abbildung 13). Während der Partikel-Schätzer der tatsächlichen Position mit geringer zeitlicher Verzögerung folgt, ist die Konvergenzgeschwindigkeit des Kalman-Filters sehr langsam.

Ein weiterer Vorteil der Partikelfilter ist hierbei, dass die Varianz des Schätzfehlers $E[(\hat{X} - X)^2]$ proportional zur Varianz des Schätzers $E[\hat{X}^2]$ ist (Gleichung 9) (in Abbildung 14 für das *Kidnap-Robot-Problem*). Für Handlungsplanungsalgorithmen wäre dieser Parameter sicherlich sehr von Nutzen, zumal er

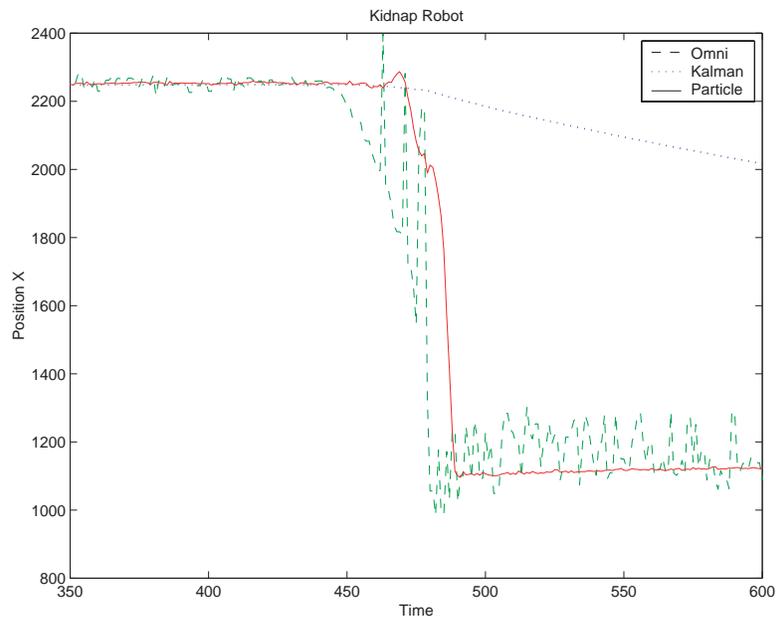


Abbildung 13: x-Position der Pose-Schätzung mittels omnidirektionaler Kamera während des Roboter-Kidnappings. Dargestellt sind die originalen Positionsschätzungen und die Schätzungen durch den Kalman- und den Partikelfilter in Abhängigkeit von der Zeit. Eine Zeiteinheit entspricht 50 ms.

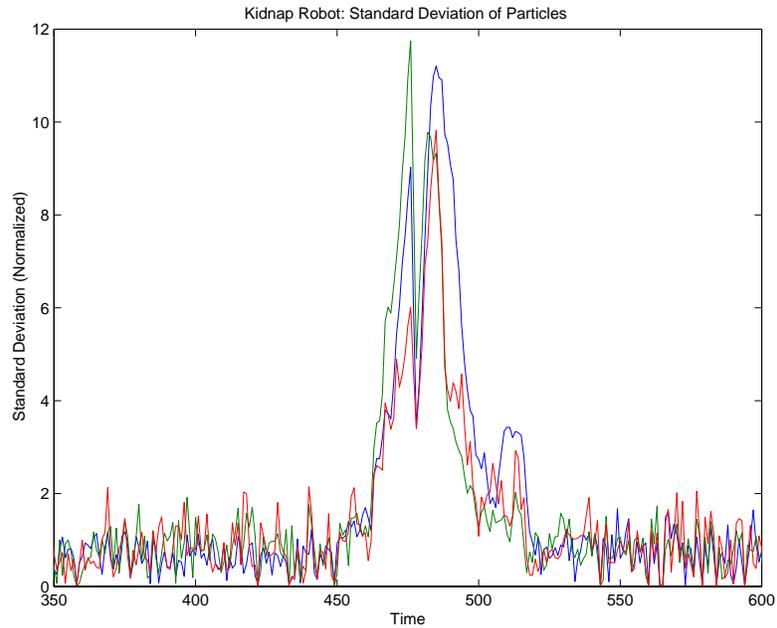


Abbildung 14: Standardabweichungen der Pose-Schätzung (x, y, θ) des Partikel-filters während des Roboter-Kidnappings. Dargestellt ist der Verlauf der norma-lisierten Standardabweichungen der Partikel entlang der Dimensionen x, y, θ in Abhängigkeit von der Zeit. Eine Zeiteinheit entspricht 50 ms.

im vorliegenden Fall nur eine geringe Latenz zeigt.

6 Fazit und Ausblick

Für den Fall der stationären Selbstlokalisierung zeigte sich ein schnelles Konvergenzverhalten bei beiden Verfahren. Sowohl der Kalman-Filter als auch der Partikelfilter konvergieren nach 400 Messwerten (20 s) von einer anfangs um etwa 10 cm falschen Position (Abbildungen 9, 10).

Auch während der Testfahrt konvergieren beide Verfahren gegen die korrekte Position. Allerdings erreicht die größte Abweichung beim Kalman-Filter bis zu 2 Meter während die Positionsschätzung des Partikelfilters nur um maximal 50 cm abweicht (Abbildung 12). Der Nachteil des Partikelfilters ist hier eine erhöhte Varianz des Schätzers. Zwischen einzelnen Messzeitpunkten schwanken die Schätzungen um bis zu einem Meter. Dies wird wahrscheinlich durch eine mehrgipflige Verteilung bedingt, die dadurch zustande kommt, dass die propagierten Verteilung und die aktuellen Messwerte nicht mehr übereinstimmen. Der Mittelwert ist hier keine geeignete Schätzfunktion und es wäre unbedingt notwendig hier einen stabilen Maximum-Likelihood-Schätzer zu verwenden.

Im Falle des Roboter-Kidnapping konvergiert der Kalman-Filter nichts mehr innerhalb der Testsequenz von etwas einer Minute gegen die wahre Position, die Abweichung beträgt über einen Meter. In einem solchen Fall müsste man den Roboter manuell aus dem Spielfeld entfernen, wohingegen der Partikelfilter mit einer Latenz von nur etwa 10 Samples (500 ms) der Kameraschätzung auch ohne Odometriedaten folgen kann.

Die eingangs aufgestellten Fragestellungen nach der Angemessenheit der Normalverteilungsannahme und dem quantitativen Abweichung der Schätzung durch die Fehlmessungen lassen sich damit eindeutig beantworten. Hinsichtlich der Modellierung der Fehlerverteilung für die stationäre Lokalisationsschätzung kann das hier vorgestellte Verfahren der Kovarianzschätzung als hinreichend betrachtet werden. Beide Verfahren konvergieren innerhalb derselben Zeit gegen die gleichen Werte. Sie unterscheiden sich in den unterschiedlich implementierten Anfangsbedingungen und darin, dass beim Partikelfilter stärker lokale Schwankungen durch die endliche Partikelanzahl auftreten.

Den entgegengesetzt zeigt sich in den Bedingungen, in denen die Odometrie zur Propagation der Pose eingesetzt wird, dass hier der Kalman-Filter keine zuverlässige Schätzung mehr liefern kann. Selbst für die von der Odometrie als steige Kurve aufgezeichnete Testfahrt (Abbildung 5) ist die Konvergenzgeschwindigkeit des Kalman-Filter absolut unzureichend. Jedoch ist auch die Stabilität der Schätzung des Partikelfilters nicht zufrieden stellend.

Es erscheint daher nicht sinnvoll den Kalman-Filter als alleinigen statistischen Filter für die visuelle Selbstlokalisierung einzusetzen, da hier die Propagation mit dem linearen Modell nicht ausreichend ist.

Dagegen ist die Latenz der Pose-Schätzung mittels Partikelfilters beeindruckend. Als Weiterentwicklungen wäre es hier wünschenswert, die Partikelanzahl in Abhängigkeit von der Varianz der Partikel zu wählen (vergleiche Abbildung 14). Zudem ist es notwendig, zukünftig eine Schätzfunktion zu verwenden, die auch bei multimodalen Verteilungen zuverlässige und stabile Schätzungen liefert.

Literatur

- [1] K. Chung. *Markov Chains with Stationary Transition Probabilities*. Springer, 1960.
- [2] A. Doucet. On sequential simulation-based methods for bayesian filtering. Technical Report CUED/F-INFENG/TR.310, Department of Engineering, University of Cambridge, CB2 1PZ Cambridge, 1998.
- [3] R.M. du Plessis. *Poor Man's Explanation of Kalman Filtering*. Taygeta Scientific Incorporated: Monterey CA, 1967.
- [4] W. Gellert, H. Küstner, M. Hellwich, and H. Kästern, editors. *Kleine Enzyklopädie Mathematik*. VEB Bibliographisches Institut Leipzig, second edition, 1967.

- [5] J.-S. Gutman. Vergleich von algorithmen zur selbstlokalisierung eines mobilen roboters. Master's thesis, Universität Ulm, Fakultät für Informatik, 1996.
- [6] J.A. Hartigan. *Bayes Theory*. Springer, first edition, 1983.
- [7] R.E. Kalman. A new approach to linear filtering and prediction problems. In *Trans. ASME, Series D*, number 82 in J. Basic Eng., 1960.
- [8] J. Liu. Metropolized independent sampling with comparison to rejection sampling importance sampling. *Statistics and Computing*, 6:113–119, 1996.
- [9] P. Maybeck. *Stochastic Models, Estimation and Control*, volume 1. Academic Press, 1979.
- [10] M. Plagge. Personal communication: Pioneer baseserver, 2000.
- [11] D.B. Rubin. A noniterative sampling/importance resampling alternative to the data augmentation algorithm for creating a few imputations when the fraction of missing information is modest: the sir algorithm. discussion of tanner and wong (1987). *J. Am. Statist. Assoc.*, 23:543–546, 1987.
- [12] D.B. Rubin. Using the sir algorithm to simulation posterior distributions. In J.M. Bernardo, M.H. DeGroot, D.V. Lindley, and A.F.M. Smith, editors, *Bayesian Statistics 3*, pages 395–402. Oxford University Press, 1988.