



Hochschule für Angewandte Wissenschaften Hamburg
Hamburg University of Applied Sciences

Ausarbeitungsarbeit

Andrej Rull

Fahrspur- und Odometrie-basierte
Selbstlokalisierung und Kartierung (SLAM)

Inhaltsverzeichnis

Tabellenverzeichnis	3
Abbildungsverzeichnis	4
1 Einführung	5
1.1 Wozu eine Karte?	6
1.2 Der Anwendungsfall	6
2 Problemstellung	8
2.1 Positionsbestimmung	8
2.1.1 Odometriebasierte Positionsbestimmung	9
2.1.2 Kartenbasierte Positionsbestimmung	11
2.2 SLAM - Simultaneous Localization and Mapping	12
2.3 Anforderung an den Anwendungsfall	14
3 Lösungsansätze	15
3.1 Extended Kalman Filter	16
3.2 ATLAS Framework	18
3.3 CaroloCup Ansatz	19
4 Resümee und Ausblick auf AW2	20
Literaturverzeichnis	21

Tabellenverzeichnis

1.1	Fehlerszenarien beim CaroloCup Fahrzeug	7
-----	---	---

Abbildungsverzeichnis

2.1	Odometriebasierte Positionsbestimmung	9
2.2	Fehler der Odometrie	10
2.3	Fehler der Odometrie	11
2.4	Kartenbasierte Positionsbestimmung	11
2.5	Datenabhängigkeit bei der Kartierung ¹	13
2.6	Zunahme des Fahlers im Laufe der Fahrt ²	14
3.1	Extended Kalman Filter	17
3.2	Fehlerkorrektur durch EKF ³	17
3.3	Aufteilung der globalen Karte ⁴	18

1 Einführung

Seit mehreren Jahren gewinnen autonome mobile Systeme zunehmend an Bedeutung. So sind solche Systeme nicht nur in der Forschung zu finden, sondern immer mehr im alltäglichen Leben. Dabei haben die Systeme die Aufgaben Menschen in verschiedenen Bereichen zu entlasten und die Schwächen der Menschen zu kompensieren. Autonome Systeme bedienen sich am Vorteil der Sensoren, die anderes als der Mensch gegenüber Ermüdung oder Emotionen stabil sind und dadurch zuverlässiger und länger arbeiten. Beispiele hierfür sind die Staubsaugerroboter, die sich eigenständig in einem Raum orientieren und diesen eigenständig säubern, oder die Nasa Erkundungsroboter "Opportunity" und "Spirit", die auf dem Mars eigenständig nach Wasser suchen.

Ein weiterer Forschungsschwerpunkt für autonome mobile Systeme ist im Automobilenbereich verankert. Solche Systeme haben die Aufgabe den Menschen bei der Fahrzeugführung zu entlasten und dadurch die Anzahl der Verkehrsunfälle zu reduzieren. Dabei wird unterschieden zwischen Autonomen- und Fahrerassistenz Systemen. Während Autonome Systeme die gesamte Fahrzeugführung übernehmen und das Fahrzeug ohne Fremdeinwirkung steuern (autonome Fahrzeuge), unterstützen Fahrerassistenzsysteme den Menschen bei der Fahrzeugführung, indem sie autonom bzw. teilautonom in die Steuerung oder den Antrieb des Fahrzeugs eingreifen und den Menschen in Gefahrensituationen darüber informieren.

Zu diesem Schwerpunkt organisiert die DARPA (Defense Advanced Research Projects Agency) in den USA jedes Jahr die DARPA Grand Challenge. Dabei träten verschiedene Universitäten, der ganzen Welt gegeneinander an und versuchen mit einem autonomen Fahrzeug eine festgelegte Kurs in der Umgebung in einer bestimmten Zeit abzufahren. Das Fahrzeug muss dabei ohne menschliche Hilfe eine Route finden. Um dies zu ermöglichen, wird dem Fahrzeug eine Karte vorgegeben anhand derer sich das Fahrzeug orientieren kann.

An der HAW Hamburg wurde diesbezüglich das Forschungsprojekt FAUST gegründet, in dem an unterschiedlichen Fahrerassistenzsystemen, wie Einparkassistenten, Spurführungsassistenten oder Ausweichassistenten geforscht wird. Diese Assistenten sollen unter anderem die autonome Fahrt eines Fahrzeugs ermöglichen.

Ein FAUST-Projekt, bei dem Anhand eines Modellfahrzeugs im Maßstab 1:10 Verfahren für die autonome Fahrt entwickelt werden, ist das CaroloCup Projekt. Dabei muss sich das Fahr-

zeug auf einer unbekanntem Fahrbahn anhand von Fahrbahnmarkierungen eigenständig orientieren und dabei Hindernissen ausweichen. Anders als bei der Grand Challenge wird dem Fahrzeug keine fertige Karte zu Verfügung gestellt. Damit das Fahrzeug die Vorteile eine Karte nutzen kann, muss diese autonom während der Fahrt aus den Kameradaten und den Sensordaten modelliert werden.

Diese Ausarbeitung beschäftigt sich mit der Modellierung von Umgebungskarten und der Problematik der autonomen Kartierung. Es wird auf die Fehlerquellen der Kartierung eingegangen und insbesondere die Positionsbestimmung des Fahrzeugs in der Umgebung untersucht.

1.1 Wozu eine Karte?

Ein großes Problem des CaroloCup Fahrzeugs ist, dass es sich nur anhand von Kameradaten und Sensordaten auf der Fahrstrecke orientiert. Dabei wird die Streckenführung mittels des TFALDA Algorithmus, der in der Diplomarbeit [Berger (2008)] entwickelt wurde, erfasst und der Regeleinheit in entsprechendem Datenformat zu Verfügung gestellt. Dies funktioniert jedoch nur, solange die Kamera richtige Daten liefert. Werden die Fahrbahnmarkierungen nicht richtig oder gar nicht erkannt, versagt das System und ist für eine autonome Weiterfahrt ungeeignet.

[Tab. 1.1] stellt weitere Fehlerszenarien beim CaroloCup Fahrzeug ohne ein Kartierungsmodul dar.

Die Idee der Kartierung ist solche Fehler zu beseitigen, indem dem Fahrzeug Umgebungsinformationen zu Verfügung gestellt werden. Diese könnten dazu verwendet werden um das Ausweichen zu planen, bevor ein Hindernis erkannt wurde oder um den Weg, beim Verlassen der Fahrbahn, zurück auf die Strecke zu finden.

1.2 Der Anwendungsfall

Das momentan implementierte Kartierungssystem auf dem CaroloCup Fahrzeug funktioniert zwar, ist jedoch gegenüber äußeren Einflüssen fehleranfällig. Ist die Fahrstrecke bspw. uneben oder kommt es zum Schlupf während der Fahrt, so werden die Fehler in die Fahrzeugposition mit eingerechnet, wodurch die Position mit der Zeit ungenauer wird. Da keine Fehlerkorrektur implementiert ist, wird die Position nach einer Zeit so ungenau, dass diese für die Weltmodellierung nicht mehr verwendet werden kann.

Fehlerszenarium	Problematik	Konsequenz
schlechte Lichtverhältnisse	inkorrekte Streckendaten	Totalausfall d. autonomen Fahrt
Sonneneinstrahlung	fehlerhafte TFALDA Daten, inkorrekte Streckendaten	Fahrzeugregelung in der Zeit τ nicht möglich
Kameraausfall	keine Kameradaten	Totalausfall d. autonomen Fahrt
Ultraschall-Sensorausfall	Hindernisdaten beschränkt vorhanden	Ausweichen nur mittels Linienlaserdaten möglich
Linienlaserausfall	Hindernisdaten beschränkt vorhanden	Ausweichen nur mittels Ultraschalldaten möglich
Abkommen von der Strecke	TFALDA Daten fehlerhaft	Totalausfall d. autonomen Fahrt

Tabelle 1.1: Fehlerszenarien beim CaroloCup Fahrzeug

Ziel dieser Ausarbeitung ist es, solche Fehler mit der Zeit zu korrigieren. Dabei soll das CaroloCup Fahrzeug zunächst die Strecke mit einer langsamen Geschwindigkeit durchfahren und ein Prototyp einer Karte modellieren. Die Karte soll zunächst nur der genauen Positionsbestimmung dienen und wäre anfangs ungenau. Durch den Abgleich der neu aufgenommenen Kameradaten und Sensordaten mit den zuvor aufgenommenen Daten soll die Karte aktualisiert und rekonstruiert werden. Dabei würde diese sukzessiv genauer, sodass man im optimalen Fall nach zehn Runden eine genaue Positionsabschätzung des Fahrzeugs in der Umgebung bekommt.

Zum Schluss soll ein genaues geometrisches Weltmodell aufgebaut werden, welches die Umgebungsmerkmale in Form von Rastern zur Verfügung stellt. Durch das Weltmodell ist das Fahrzeug in der Lage Fehlverhalten während der Fahrt (bspw. verlassen der Fahrbahn) zu kompensieren.

2 Problemstellung

In diesem Kapitel wird die grundsätzliche Problematik der Kartierung bei autonomen Fahrzeugen erläutert. Die Fehlerquellen, insbesondere durch die Positionsbestimmung, werden beschrieben und deren Auswirkung auf das modellierte Weltmodell erläutert. Zum Schluss werden Anforderungen an die Weltmodellierung gestellt.

2.1 Positionsbestimmung

Eine korrekte Positionsbestimmung ist maßgeblich für eine fehlerfreie Kartierung. Die Position wird durch den Vektor:

$$Pose = \begin{pmatrix} x_p \\ y_p \\ \varphi \end{pmatrix}$$

gekennzeichnet. Dabei geben (x_p, y_p) die Positionskordinaten im Bezug auf das Weltkoordinatensystem an und (φ) den Orientierungswinkel des Fahrzeugs im Koordinatensystem. Der Positionsvektor ist entscheidend für die Transformationsmatrix, mit der die lokalen Merkmalskoordinaten (x_q, y_q) der aufgenommenen Daten in globale (x_t, y_t) transformiert werden.

$$\begin{pmatrix} x_t \\ y_t \end{pmatrix} = \begin{pmatrix} \cos \varphi & \sin \varphi \\ -\sin \varphi & \cos \varphi \end{pmatrix} * \begin{pmatrix} x_q \\ y_q \end{pmatrix} + \begin{pmatrix} x_p \\ y_p \end{pmatrix}$$

Für die Fusionierung der lokalen Karte in die globale geben die Positionskordinaten die nötige Verschiebung an und die Fahrzeugorientierung die benötigte Verdrehung. Ist die Position fehlerhaft werden auch die aufgenommenen Sensor- und Kameradaten falsch in das globale Weltmodell eingetragen. Die Fusionierung der lokalen Karten in die globale wird in [Rull (2008)] und [Ebert (2008)] genauer erläutert.

In der Fachliteratur wird generell zwischen der absoluten und der relativen Positionsbestimmung unterschieden. Bei der absoluten Positionsbestimmung ist die Fahrzeugposition anfangs unbekannt. Dem Fahrzeug wird jedoch eine fertige, fehlerfreie Karte mitgegeben anhand derer das Fahrzeug eine Selbstlokalisierung durchführt, indem die Umgebung mittels

der Sensorik analysiert wird und bestimmte Umgebungsmerkmale in der vorgegebenen Karte wiederkannte werden.

Die relative Positionsbestimmung wird ohne eine vorgegebene Karte durchgeführt. Dem Fahrzeug wird zu Beginn der Fahrt eine Anfangsposition mitgegeben, die sich durch die Odometriedaten (Wegmessungen) während der Fahrt aktualisiert. Das Problem der Odometrie ist, dass diese zwar auf kurzen Strecken genau ist, sich aber auf langen Strecken ein Fehler bildet welcher im System mitgeführt wird und sich immer weiter hochschaukelt.

2.1.1 Odometriebasierte Positionsbestimmung

In der Praxis hat sich für kurze Strecken eine Methode etabliert, durch welche sich die Position nur anhand von Wegmessungen errechnen lässt. Um den gefahrenen Weg zu bestimmen, werden im Fahrzeug Inkrementalgeber eingesetzt mit deren Hilfe sich die Umdrehung der einzelnen Räder messen lässt. Bei der Positionsberechnung wird die Differenz der gefahrenen Wegstrecken des linken Rades und des rechten Rades pro Zeiteinheit betrachtet. Dabei ist der Unterschied abhängig vom Radabstand.

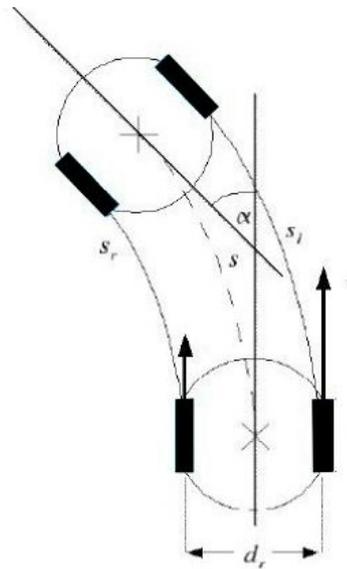


Abbildung 2.1: Odometriebasierte Positionsbestimmung

Betrachtet man eine Linksfahrt [Abb. 2.1], so legt das rechte Rad einen längeren Weg zurück als das linke, wodurch sich die Orientierung pro Zeiteinheit über die Differenz der Wegstrecken errechnet. Um die aktuelle Fahrzeugorientierung zu erhalten, wird diese zu der alten globalen Fahrzeugorientierung hinzuaddiert.

$$\delta\varphi = \frac{s_l - s_r}{d}$$

Die neue Pose errechnet sich wiederum aus der alten Pose, dem neuen Orientierungswinkel und der Wegstrecke die im Mittelzurückgelegt wurde.

$${}^{pO}\vec{s}_{k+1} = \begin{pmatrix} x_k + s_k \sin \omega_k \\ y_k + s_k \cos \omega_k \\ \omega_k + \delta\varphi_k \end{pmatrix}$$

Wie schon erwähnt ist die odometriebasierte Selbstlokalisierung zwar auf kurzen Strecken genau, auf langen Strecken jedoch kaum zu gebrauchen, da Hindernisse, Schlupf oder Bodenwellen zu Fehlern in der Berechnung führen [Abb. 2.2]. Da es keinerlei Korrektur gibt, werden die Berechnungsfehler die ganze Zeit mitgeführt und summieren sich mit der Zeit auf. [Abb. 2.3] zeigt ein Beispiel so eines Fehlerszenariums. Das rechte Bild zeigt die gewünschte Strecke, während das linke Bild die tatsächlich gefahrene Strecke beim Einsatz der odometriebasierten Selbstlokalisierung darstellt.

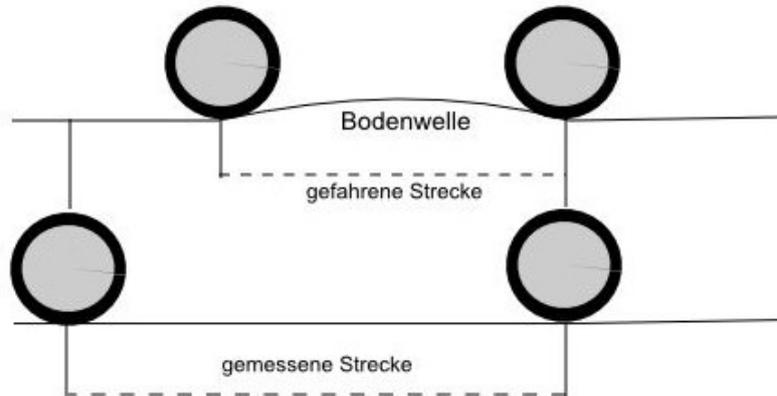


Abbildung 2.2: Fehler der Odometrie

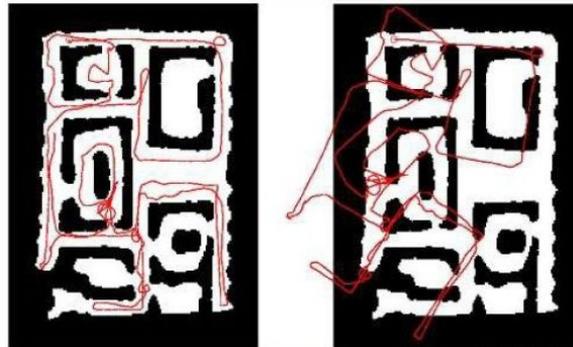


Abbildung 2.3: Fehler der Odometrie

2.1.2 Kartenbasierte Positionsbestimmung

Anders als bei der odometriebasierten Positionsbestimmung wird bei der kartenbasierten Selbstlokalisierung die Position nicht mit Hilfe der internen Sensorik (Inkrementalgeber) berechnet, sondern unter Zunahme der externen Sensoren (Ultraschallsensoren, Kamera) in der Umgebung abgeschätzt. Dabei nimmt das Fahrzeug die Umgebungsmerkmale während der Fahrt auf und vergleicht diese mit den Daten im Weltmodell. Durch das Wiedererkennen der Merkmale kann das Fahrzeug mit einer gewissen Wahrscheinlichkeit sagen wo es sich auf der Karte befindet. Danach muss nur noch die Verdrehung bzw. Verschiebung zu dem Merkmal berechnet werden um die aktuelle Position zu erhalten.

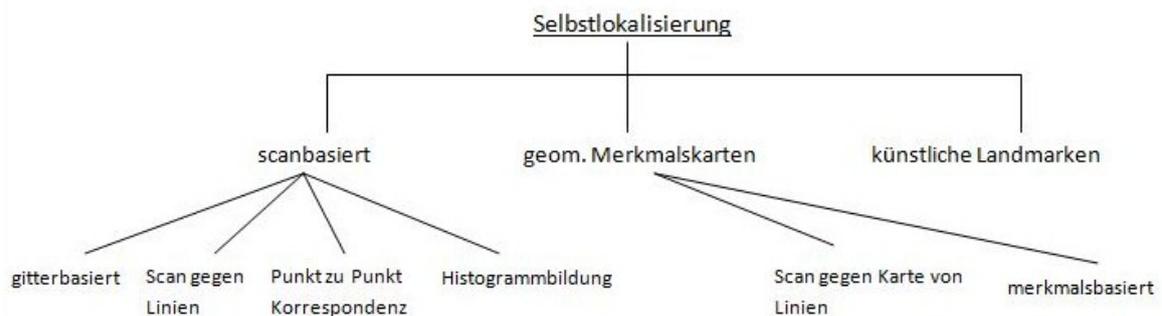


Abbildung 2.4: Kartenbasierte Positionsbestimmung

In der Fachliteratur gibt es mehrere Ansätze für die Schätzung der Position in der Umgebung. [Abb. 2.4] zeigt wie diese aufgeteilt werden. Das einfachste Lokalisierungsverfahren ist die Position anhand von künstlichen Landmarken abzuschätzen. Dabei werden in der Umgebung kodierte Marken aufgestellt, welche die Position des Fahrzeugs angeben. Entdeckt das Fahrzeug während der Fahrt eine Landmarke, so wird diese detektiert und dekodiert. Da

jeder Landmarke eindeutig gekennzeichnet ist und das Fahrzeug den Aufbau der Landmarken kennt kann die Position auch ohne eine Karte bestimmt werden.

Scan- und merkmalskartenbasierte Verfahren stellen eine dynamische Positionsbestimmung dar, die eine vorhandene Karte für die Abschätzung verwenden. Dabei werden natürliche Landmarken in der Umgebung (z.B. Kreuzungen, Linien, Kurven, Ecken) gesucht und mit den aktuell-aufgenommenen Daten abgeglichen.

Ein Problem der kartenbasierten Positionsbestimmung ist die eindeutige Kennzeichnung von natürlichen Landmarken. Insbesondere beim CaroloCup Fahrzeug gibt es auf der Fahrbahn keine künstliche Landmarken, anhand derer sich die Position bestimmen lässt. Auch können Kurven nicht als eindeutige Landmarken verwendet werden, da mehrere Kurven für das Fahrzeug identisch aussehen und sich dadurch eine eindeutige Landmarkenzuweisung nicht realisieren lässt. Ein weiteres Problem der kartenbasierten Positionsbestimmung ist der enorme Rechenaufwand. Um eine korrekte Positionsabschätzung durchführen zu können müssen alle Kartendaten analysiert werden und aktuell aufgenommene Merkmale gefunden werden, die mit den Kartendaten korrelieren.

2.2 SLAM - Simultaneous Localization and Mapping

Das größte Problem des CaroloCup Fahrzeugs ist, dass das Fahrzeug zu Beginn der Fahrt keinerlei Umgebungsinformationen in Form einer Karte besitzt, sodass ein Weltmodell zunächst autonom aufgebaut werden muss. Die Problematik dabei ist, dass die Umgebung bekannt sein muss um eine korrekte Fahrzeugposition zu erhalten. Andererseits muss eine genaue Position bekannt sein damit ein Weltmodell korrekt modelliert werden kann. Dieser Konflikt wird in der Fachliteratur auch als "Simultaneous Localization and Mapping" bezeichnet, heißt eine Selbstlokalisierung und eine Kartenmodellierung zu selben Zeit.

Laut [[Sebastian Thrun \(2008\)](#)] hängt die Weltmodellierung von der Position x_t , der Odometrie u_t und der Messung z_t ab. Diese Größen ändern sich während der Fahrt und stehen in Beziehung zu einander.

$$X_T = \{x_0, x_1, x_2, \dots, x_T\}$$

$$U_T = \{u_1, u_2, u_3, \dots, u_T\}$$

$$Z_T = \{z_1, z_2, z_3, \dots, z_T\}$$

Das eigentliche Weltmodell m wird aus allen Positionsdaten X_T und den Messdaten Z_T aufgebaut.

$$m = \{X_T, Z_T\}$$

[Abb. 2.5] stellt die Abhängigkeit der einzelnen Größen dar. Dabei hängt die Position des Fahrzeugs nicht nur von den Odometriedaten ab, sondern wird zusätzlich durch Abgleich der Messdaten mit den Daten, die in der Karte m abgelegt sind kontrolliert und korrigiert.

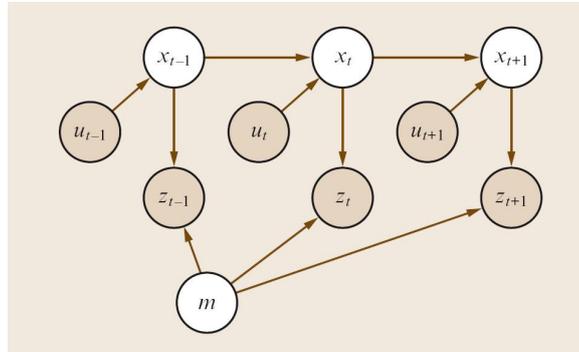


Abbildung 2.5: Datenabhängigkeit bei der Kartierung¹

Fährt das Fahrzeug von der Position x_{t-1} zu der Position x_{t+1} wird zunächst die Position mittels der Odometriedaten u_{t-1} , u_t , und u_{t+1} berechnet. Durch die Aufnahme von neuen Messdaten z_{t-1}, z_t und z_{t+1} und dem Abgleich dieser mit den Messdaten, die in der Karte gespeichert sind, wird die Fahrzeugposition überprüft und aktualisiert.

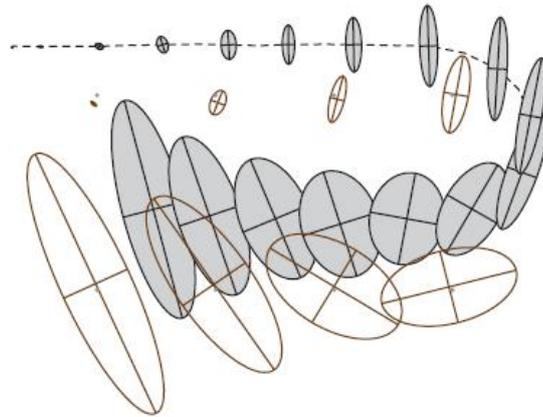
[Abb. 2.5] veranschaulicht auch die SLAM Problematik, denn zu Beginn der Fahrt ist die Karte m leer und beinhaltet keine Information über die Umgebung, sodass die Positionierung des Fahrzeug nur anhand der odometriedaten durchgeführt werden kann. Was dazu führt, dass die Karte wie in [Abb. 2.6] falsch modelliert wird. Die grauen Fehlerellipsen, stellen die mögliche Positionsabweichung von der tatsächlichen Position dar, während die weißen Fehlerellipsen den Transformationsfehler der Messdaten im Weltmodell darstellen.

[Sebastian Thrun (2008)] unterscheidet zwischen dem Offline SLAM Problem und dem Online SLAM Problem. Beim Offline SLAM Problem werden alle Wegmessungen sowie alle Umgebungsmessungen betrachtet und eine Verbindung zwischen diesen hergestellt. Dabei wird für jede Messung zu einem späteren Zeitpunkt eine Positionsabgeschätzt durchgeführt, sodass am Ende jede Messung eine Position zugewiesen bekommt und man ein Pfad von Positionen erhält, mit dem sich ein Weltmodell konstruieren lässt. Dies wird definiert durch:

$$p(X_T, m | Z_T, U_T)$$

Die Variablen rechts von Balken sind durch interne und externe Sensorik gegeben, während die links vom Balken abgeschätzt werden. Beim online SLAM Problem wird nicht der ganze Positionspfad im nahhinein abgeschätzt, sondern nur die aktuelle Fahrzeugposition zum

¹ Sebastian Thrun (2008)

Abbildung 2.6: Zunahme des Fehlers im Laufe der Fahrt²

momentanen Zeitpunkt.

$$p(x_t, m | Z_T, U_T)$$

2.3 Anforderung an den Anwendungsfall

Durch die Problemstellung ergeben sich mehrere Anforderungen, die bei der Weltmodellierung auf dem CaroloCup Fahrzeug berücksichtigt werden müssen. Zunächst müssen Landmarken in der Umgebung definiert werden, die während der Fahrt wiedererkannt werden können. Dies ist auf einer Strecke schwerer zu realisieren als in einem Raum mit vielen Ecken und Kanten, denn viele Streckenabschnitte ähneln sich, was dazu führt, dass keine genaue Positionsaussage getroffen werden kann.

Als nächstes muss das online SLAM Problem gelöst werden, denn die Landmarkenposition wird bei der ersten Fahrt durch die Odometrie bestimmt. Der daraus resultierende Fehler muss beim Wiedererkennen von Landmarken korrigiert werden. Die Korrektur kann jedoch erst beginnen, nachdem ein Kreis geschlossen wurde, das heißt wenn die erste Landmarke wiedererkannt wird, da die Positionierung der ersten Landmarke durch die Startposition bekannt ist und keinen Fehler aufweist. Die Abweichung der berechneten Position zur der Landmarkenposition muss in entsprechendem Fall korrigiert und die Korrektur auf andere Landmarken im System angewandt werden.

² Sebastian Thrun (2008)

3 Lösungsansätze

Zur Lösung des online SLAM Problems werden in der Literatur meist Verfahren verwendet, die die Position während der Fahrt schätzen. Hierzu werden stochastische Karten aufgebaut, die mit der Zeit eine höhere Genauigkeit der Positionsabschätzung gewährleisten. Die Karten werden durch verschiedene Verfahren, die Filter genannt werden, aufgebaut und rekonstruiert. Der Extended Kalman Filter [Sebastian Thrun (2008)] ist die meist verbreitete Methode, die das online SLAM Problem ansatzweise lösen kann. Ein anderes Verfahren ist der Rao Blackwellised Particle Filter [Kwok und Dissanayake (2007)] oder auch das ATLAS Framework [Bosse u. a. (2003)], das Hypothesen für jeden lokalen Scan aufstellt um die Fahrzeugpositionierung den lokalen Scans zuzuordnen.

Alle Verfahren vertrauen zunächst der Odometrie, da man sonst keine Möglichkeit hat die Position zu bestimmen. Bei der ersten Fahrt werden Landmarken bzw. Features aus der Umgebung extrahiert oder lokale Scans der Umgebung zwischengespeichert und mit der odometrisch berechneten Position versehen. Erkennt das Fahrzeug ein Merkmal wieder bzw. eine Übereinstimmung zwischen zwei Scans und wird eine Abweichung der odometrischen Position zur Landmarkenposition bzw. der Scanposition entdeckt, korrigieren die oben erwähnten Methoden zum einen den Positionsvektor des Fahrzeugs und zum anderen die Positionskoordinaten der Landmarken bzw. der Scans im Weltmodell, wodurch sich die Karte eigenständig rekonstruiert. Durch diese Art der Weltmodellierung, wird die stochastische Karte als ein Graph von lokalen Sensorkarten gesehen [Bosse u. a. (2003)] oder insbesondere bei den Filtermethoden als eine Menge von Landmarken in der Umgebung.

Für die Lösung des SLAM Problems implementieren alle Verfahren zwei Modelle, das mathematische Modell:

$$p(x_t | x_{t-1}, u_t)$$

und das messungsbasierte Modell:

$$p(z_t | x_t, m)$$

Während das mathematische Modell die odometriebasierte Positionsbestimmung implementiert, hat das messungsbasierte Modell die Aufgabe die aktuellen Messungen einer bestimmten Position im Weltmodell zuzuordnen. Diese Zuordnung wird durch Wahrscheinlichkeitswerte gekennzeichnet, die mit der Zeit genauer werden.

3.1 Extended Kalman Filter

Ein Kalman Filter ist ein Verfahren, das einen Zustand eines linearen Systems vorhersagt. Dabei wird der geschätzte Ausgangswert mit dem tatsächlich gemessenen Ausgangswert verglichen. Die Differenz bzw. die Abweichung wird linear gewichtet und dient zur Verbesserung der Zustandsschätzung.

Für nicht lineare Systeme wird der Extended Kalman Filter verwendet, der auf einer Taylor-Näherung erster Ordnung beruht [Sebastian Thrun (2008)]. Bei der Weltmodellierung wird das Weltmodell als ein Zustandsvektor betrachtet, der die Positionskordinaten des Fahrzeugs, sowie alle Koordinaten der Landmarken beinhaltet. Der Vektor beinhaltet die beste Abschätzung der Fahrzeugposition sowie der Landmarkenpositionen und hat die Größe $(3+2N)$, wobei N die Anzahl der Landmarken angibt.

$$\vec{\mu}_t = (x_p, y_p, \varphi_p, x_1, y_1, \dots, x_n, y_n)^T$$

Die Elemente im Vektor werden auch Zufallsvariablen genannt, da diese sich während der Fahrt ändern. Der erwartete Schätzfehler im Zustandsvektor wird durch die Kovarianzmatrix angegeben, in der alle Landmarken in Beziehung zu einander, sowie in Beziehung zu der Fahrzeugposition gestellt werden, wodurch die Matrixgröße $(3+2N) \times (3+2N)$ beträgt. Die Zufallsvariablen des Zustandsvektors befinden sich auf der Diagonale der Matrix.

$$\Sigma_t = \begin{bmatrix} C_{pp} & C_{p1} & \dots & C_{pn} \\ C_{1p} & C_{11} & \dots & C_{1n} \\ \cdot & \cdot & \dots & \cdot \\ \cdot & \cdot & \dots & \cdot \\ C_{np} & C_{n1} & \dots & C_{nn} \end{bmatrix}$$

Am Anfang einer Fahrt beinhaltet der Zustandsvektor nur die Startkoordinaten und die Orientierung des Fahrzeugs. Beim Detektieren neuer Landmarken in der Umgebung, werden die Koordinaten der Landmarken im Zustandsvektor angelegt und in die Kovarianzmatrix eingetragen, wodurch die Kovarianzmatrix quadratisch wächst [Sebastian Thrun (2008)]. Die Aktualisierung der Zustandsvariablen erfolgt wie in [Abb. 3.1] dargestellt in drei Schritten. Zunächst wird die Position mittels der Odometriedaten aktualisiert, danach werden neue Messdaten mit den schon vorhandenen Daten im System abgeglichen. Wird eine Landmarke im System wiedererkannt und eine Abweichung in der Position festgestellt, so werden die Positionskordinaten im Zustandsvektor sowie in der Kovarianzmatrix aktualisiert. Wird eine neue Landmarke detektiert, werden deren Koordinaten in den Zustandsvektor eingetragen.

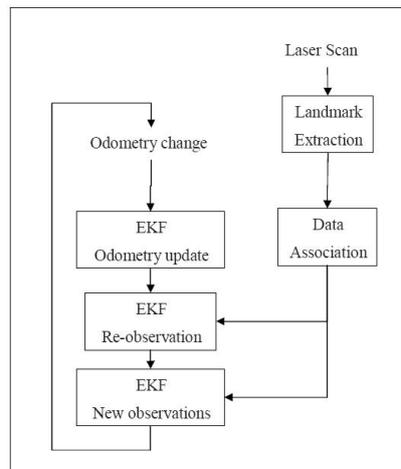
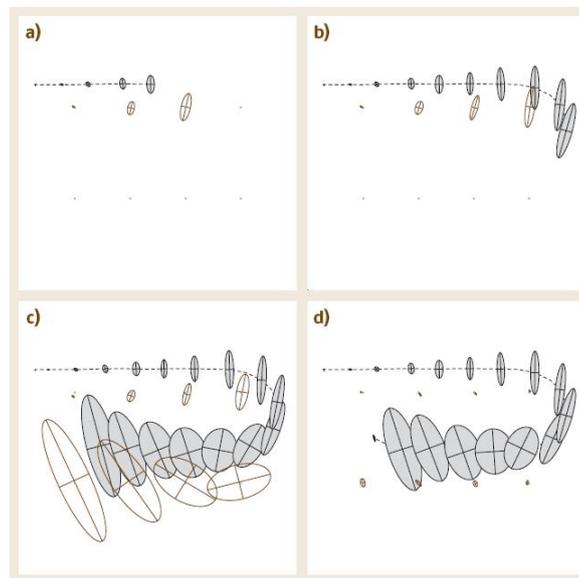


Abbildung 3.1: Extended Kalman Filter

[Abb. 3.2] veranschaulicht die Arbeitsweise des Extended Kalman Filters. Auf dem Bild a), b) und c) wird der Zustandsvektor nur anhand der Odometriedaten aktualisiert, wodurch der Fehler bei der Positionsschätzung im Laufe der Fahrt zunimmt. Auf Bild d) wird eine Landmarke vom System wiedererkannt und die Koordinaten des Fahrzeugs, sowie der Landmarken im Zustandsvektor aktualisiert, wodurch die Fehlerellipsen der Fahrzeugposition und der Landmarken kleiner werden.

Abbildung 3.2: Fehlerkorrektur durch EKF³

³ Sebastian Thrun (2008)

3.2 ATLAS Framework

Ein anderer Ansatz um das SLAM Problem zu lösen ist das ATLAS Framework. Anders als bei Extended Kalman Filter handelt es sich hierbei um kein featurebasiertes- sondern um ein scanbasiertes Verfahren. Dabei wird für jeden aktuellen Scan ein lokaler Frame angelegt, sodass während einer Fahrt mehrere lokale Frames im System erzeugt werden, die miteinander Verbunden sind und einen Graph von Frames bilden. Da die Framegröße bekannt ist, weiß das Fahrzeug wann der Framebereich überschritten wird, sodass ein neuer Frame erzeugt werden kann. Das ATLAS Framework betrachtet das Weltmodell nicht in einem globalen Koordinatensystem, sondern als eine Menge von lokalen Frames, die jeweils ein lokales Koordinatensystem implementieren.

Für die kartenbasierte Positionsbestimmung wird jedem Frame außer den Sensordaten die Fahrzeugposition zugewiesen. Die Startposition ist genau wie beim Extended Kalman Filter zu Beginn bekannt. Da die Folgepositionen sich zunächst aus der Odometrie berechnen, erhalten die Frames, die zu einer späteren Zeit erzeugt werden eine Position mit einem höherem Fehler als die Frames die zu Beginn erzeugt wurden. [Abb. 3.3] veranschaulicht ein Weltmodell, welches durch mehrere lokale Karten m_1, m_2, m_3 definiert wird.

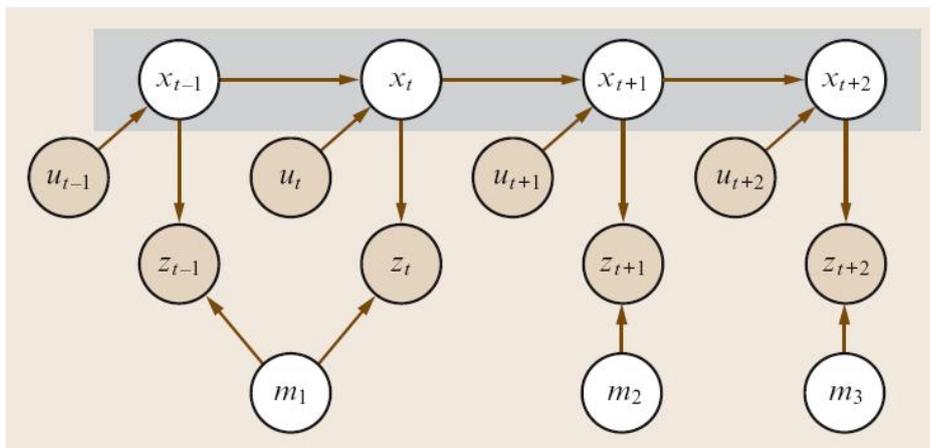


Abbildung 3.3: Aufteilung der globalen Karte⁴

⁴ Sebastian Thrun (2008)

Die Fehler in der Position der Frames werden durch Scan-Match Verfahren korrigiert. Der aktuelle Umgebungsscan wird dabei mit den Frames abgeglichen. Jeder Frame bildet für sich eine Hypothese, ob die aktuelle Fahrzeugposition zu dem Frame passt und ob die Daten im aktuellen Scan mit den Daten im Frame eine hohe Übereinstimmung haben. Hat ein Frame mit dem aktuellen Scan eine hohe Überdeckung, so wird durch Scan-Match Verfahren, wie in [Rull (2008)] beschrieben, nach der Transformation gesucht, die die Verschiebung und Verdrehung der Scans angibt. Die Position wird dann entsprechend der Transformation in allen Frames korrigiert, indem sich die Korrektur zurück am Pfad entlang pflanzt.

3.3 CaroloCup Ansatz

Der Extended Kalman Filter, sowie das ATLAS Framework stellen interessante Ansätze zur Lösung des SLAMs Problems dar, wobei sich diese nicht ohne weiteres auf das CaroloCup Fahrzeug übertragen lassen. Wie schon erwähnt gibt es auf der Strecke keine eindeutige Features, was den Einsatz des EKF erschwert. Auch der Ansatz mit den lokalen Frames hat die Problematik, dass sich mehrere Frames ähneln, sodass für zwei unterschiedliche Geraden durch den ATLAS Ansatz nur ein Frame erzeugt wird.

Die Idee beim CaroloCup Fahrzeug ist es sich am ATLAS Framework zu orientieren, diesen aber in manchen Bereichen abzuändern. So soll sich der Framegraph nicht selbst aufbauen sondern von einem Zustandsgenerator erzeugt werden. Der Zustandsgenerator soll die Frames, die bei der Fahrt als nächstes erwartet werden als aktiv markieren. Zu Beginn der Fahrt würden lokale Frames erzeugt, die sich um einen gewissen Grad unterscheiden, wie bspw. Aufnahmen von Geraden, Kurven, Kreuzungen oder Startlinien. Da mehrere Geraden bzw. Kurven auf einer Strecke auftauchen soll der Zustandsgenerator einen Abgleich des aktuellen Scans mit zwei gleichen Frames verhindern. Die Transformation zwischen zwei Scans könnte durch den Punk-zu-Punkt Korrespondenz Algorithmus IDC berechnet werden, der in [Lu und Milios (1997)] erläutert wird.

Eine Positionskorrektur würde, wie schon beim ATLAS und EKF erst starten, wenn das Fahrzeug den Frame mit den Startkoordinaten wiedererkennt. Die Korrektur würde entsprechend alle Framepositionen, außer der Position im Startframe (da korrekt), aktualisieren.

4 Resümee und Ausblick auf AW2

In dieser Ausarbeitung wurde die grundsätzliche Problematik der Kartierung erläutert, die verschiedene Forschungsprojekte zu lösen versuchen. Auch an der HAW Hamburg, wird an der Problematik geforscht, indem eine Karte von einer Fahrbahn mittels eines Modellfahrzeugs modelliert wird. Die Kartierung und Positionsbestimmung zur Selben Zeit (Simultaneous Location and mapping) stellt dabei das größte Problem der Weltmodellierung dar. Denn für ein korrektes Weltmodell wird eine korrekte Position vorausgesetzt, für eine korrekte Positionsbestimmung ist jedoch ein korrektes Weltmodell unausweichlich. In der Forschung versucht man die Problematik durch stochastische Karten in den Griff zu bekommen. Dabei werden Karten zunächst ungenau modelliert und mit der Zeit, durch das Wiedererkennen von Umgebungsmerkmalen, korrigiert und rekonstruiert.

In Hinblick auf Anwendung 2 soll das ATLAS Framework genauer analysiert werden. Für das CaroloCup Fahrzeug müssen Regeln definiert werden, die den Aufbau von lokalen Frames beschreiben, damit nicht jede lokale Sensorkarte auch einen Frame bildet. Durch den Frameansatz soll die Fahrzeugposition nach mehrfachem durchfahren einer Fahrstrecke so genau bestimmt werden, dass die momentan implementierte Weltmodellierung verbessert wird. Die Weltmodellierung soll dann ein genaues geometrisches Weltmodell anderen Fahrassistenzsystemen bereitstellen.

Literaturverzeichnis

- [Berger 2008] BERGER, Denis: *Fahrspurerkennung mit Three Feature Based Lane Detection Algorithm(TFALDA)*, Hochschule für Angewandte Wissenschaften Hamburg, Studienarbeit, 2008
- [Bosse u. a. 2003] BOSSE, Michael ; NEWMAN, Paul ; LEONARD, John ; TELLER, Seth: An atlas framework for scalable mapping. (2003), S. 1899–1906
- [Ebert 2008] EBERT, Michael: *Aktives Mapping und Positionsbestimmung eines autonomen Modellfahrzeugs*, Hochschule für Angewandte Wissenschaften Hamburg, Bachelorarbeit, 2008
- [K.E. Bekris 2006] K.E. BEKRIS, L.E. K.: Evaluation of algorithms for bearing-only SLAM. In: *IEEE International Conference on Robotics and Automation* (2006), S. 1937–1943
- [Kuipers 2000] KUIPERS, Benjamin: The spatial semantic hierarchy. In: *Artificial Intelligence* 119 (2000), Februar, S. 191–233
- [Kwok und Dissanayake 2007] KWOK, N. M. ; DISSANAYAKE, G.: Bearing-only slam in indoor environments using a modified particle filter. In: *In Australasian Conference on Robotics and Automation* (2007), S. 1–8
- [Lu und Milios 1997] LU, Feng ; MILIOS, Evangelos: Robot Pose Estimation in Unknown Environments by Matching 2D Range Scans. In: *Journal of Intelligent and Robotic Systems* 18 (1997), S. 249–275
- [Rull 2008] RULL, Andrej: *Sensorbasierte Umgebungskartierung mit lokaler Positionskorrektur für autonome Fahrzeuge*, Hochschule für Angewandte Wissenschaften Hamburg, Bachelorarbeit, 2008
- [Sebastian Thrun 2008] SEBASTIAN THRUN, John J. L.: Simultaneous Localization and Mapping. In: *Handbook for Robotics* (2008), S. 871–889