Interner Bericht

DLR-IB-FT-BS-2021-260

Multisensordatenfusion zur optischen Navigation für UAVs

Hochschulschrift

Bauer, Marius

Deutsches Zentrum für Luft- und Raumfahrt

Institut für Flugsystemtechnik
Braunschweig

Deutsches Zentrum
DLR für Luft- und Raumfahrt

Institutsbericht
DLR-IB-FT-BS-2021-260

Multisensordatenfusion zur optischen Navigation für UAVs

Bauer, Marius

Institut für Flugsystemtechnik
Braunschweig

114 Seiten
050 Abbildungen
010 Tabellen
058 Referenzen

Deutsches Zentrum für Luft- und Raumfahrt e.V.
Institut für Flugsystemtechnik
Abteilung Unbemannte Luftfahrzeuge

Stufe der Zugänglichkeit: I, Allgemein zugänglich: Der Interne Bericht wird elektronisch ohne Einschränkungen in ELIB abgelegt. Falls vorhanden, ist je ein gedrucktes Exemplar an die zuständige Standortbibliothek und an das zentrale Archiv abzugeben.

Braunschweig, den 02.02.2024

Institutsdirektor: Prof. Dr.-Ing. S. Levedag

Abteilungsleitung: Johann Dauer

Betreuer:in: Nikolaus Ammann

Verfasser:in: Marius Bauer

Masterarbeit

Multisensordatenfusion zur optischen Navigation für UAVs

Optimierung der Positionsschätzung durch Anpassung des Blickwinkels

Marius Bauer

Themensteller: Prof. Dr. Wolfram Schiffmann
Betreuer: Nikolaus Ammann
Lehrgebiet Rechnerarchitektur
Fachbereich Informatik

Inhaltsverzeichnis

1 Motivation und Zielsetzung **1**

2 Grundlagen und Testsystem **3**
2.1 Navigation für UAVs 3
2.2 Anpassung des Blickwinkels 10
2.3 superArtis Kleinhubschrauber 14

3 Strategien zur Landmarken-Auswahl und deren Implementierung **37**
3.1 Minimierung der Zustandsvarianz 39
3.2 Heuristiken 47
3.3 Statische Blickwinkel 52

4 Analyse und Vergleich der Strategien **54**
4.1 Rahmenbedingungen der Simulation 55
4.2 Laufzeiten 62
4.3 Performance nach Verlust des GPS Signals 65
4.4 Performance ohne GPS 79

5 Fazit **90**

A Ergänzende Abbildungen **93**

Kurzfassung

Das Ziel dieser Arbeit ist die Untersuchung von Strategien zur Anpassung des Blickwinkels von Kamera und Lidar-Sensor als optische Komponente der UKF-basierten Navigations- und Steuersoftware des superArtis Forschungshubschraubers des DLR. Es wurden sowohl statische Blickwinkel als auch Strategien zur Auswahl einer nachzuverfolgenden Landmarke implementiert. Letztere waren verschiedene Heuristiken sowie Strategien auf Basis der für eine Sensormessung erwarteten Fehler-Kovarianzmatrix. Alle Strategien wurden mithilfe von simulierten Flügen verglichen. Es konnte eine deutliche Verbesserung der Zustandsschätzung und der Filterinitialisierung durch den Einsatz von active vision im Vergleich mit allen statischen Blickwinkeln beobachtet werden, wobei die Performance aller Strategien maßgeblich von der Anzahl sichtbarer Landmarken abhängt. Die Performanceunterschiede zwischen den dynamischen Strategien sind dabei von konkreten Einsatzszenarios abhängig. Die implementierten und mittels Simulation evaluierten Strategien können bezüglich weiterer Einflussfaktoren untersucht und müssen in einem nächsten Schritt auch unter realen Bedingungen in Flugversuchen erprobt werden.

1 Motivation und Zielsetzung

Eine grundlegende Fähigkeit für diverse potentielle Einsatzmöglichkeiten unbemannter Luftfahrzeuge (*engl.* unmanned aerial vehicles, UAVs) ist die autonome Navigation. Um eine Schätzung der Position und des Flugzustands von hoher Güte zu erhalten, müssen die Daten verschiedener Sensoren fusioniert werden. Hierbei können zum Beispiel inertiale Messeinheiten (*engl.* inertial measurement units, IMUs), globale Navigationssatellitensysteme (GNSS), Magnetometer, Barometer, Lidar-Sensoren (light detection and ranging) sowie Kameras zum Einsatz kommen.[Con18] Die Kameras sind dabei häufig starr am UAV montiert, sodass das Blickfeld der Kamera durch die Position und Lage des UAVs im Raum bestimmt wird.

Dem Gegenüber steht der Einsatz beweglicher Kameras zur dynamischen Anpassung des Blickfelds, was als *active vision* bezeichnet wird. In der Literatur sind Navigationsverfahren unter dem Einsatz von active vision bisher vorwiegend für mobile Roboter und nicht für UAVs beschrieben.[BEF96][DM98][CLK11] Strategien zur dynamischen Auswahl eines zu beobachtenden Bildbereichs sind für UAVs zwar bekannt, jedoch in der Regel mit der entsprechenden Anpassung der Lage des Vehikels verknüpft.[BS05][BS08]

Durch den Einsatz von active vision für UAVs kann nun dieses starre Abhängigkeit zwischen Lage des Vehikels und Blickwinkel aufgelöst werden. In diesem Fall ist die geplant Trajektorie vom idealen Blickfeld für die Zustandsschätzung zumindest partiell losgelöst. Gleichzeitig kann die inhärente Abhängigkeit zwischen Blickfeld (*engl.* field of view, FOV) und Auflösung der Kamera insofern abgemildert werden, dass mit einer beweglichen Kamera trotz hoher Auflösung und kleinem FOV eines Einzelbilds ein Panorama der Umgebung zugänglich ist.[SR08][HFH+11]

Am Institut für Flugsystemtechnik des Deutschen Zentrums für Luft- und Raumfahrt in Braunschweig wird der unbemannte Kleinhubschrauber superArtis zur Forschung an robusten Navigationsverfahren eingesetzt. Der Hubschrauber ist zu diesem Zweck mit einer IMU, verschiedenen Kameras und weiteren Sensoren sowie einem Boardcomputer ausgestattet. Die Kameras sind dabei an einer Schwenk-Neige-Einheit (*engl.* pan tilt unit, PTU) angebracht, wodurch die Kameraausrichtung ohne Änderung der Lage des Hubschraubers über einen weiten Bereich variiert werden kann. Die Navigation beziehungsweise Zustandsschätzung des superArtis Hubschraubers findet mithilfe eines Uscented Kalman-Filters auf Basis der Messdaten der IMU mit Korrektur durch die Informationen der übrigen Sensoren sowie der Steuerbefehle statt. Dabei fließen auch die Positionen von Landmarken, die im Rahmen der Vorverarbeitung der Kamerabildern bestimmt wer-

den, in die Korrektur ein.[AAF+18] Als Landmarken werden so genannte AprilTags eingesetzt.[Ols11][WO16]

In dieser Arbeit wird die Optimierung der Zustandsschätzung durch die Anpassung des Blickwinkels der durch die PTU beweglichen Kameras des superArtis Hubschraubers untersucht. Es werden verschiedene Strategien zur dynamischen Anpassung des Blickwinkels der Kamera erarbeitet und implementiert. Die Implementierung erfolgt in C/C++ im Rahmen des Frameworks der Navigationssoftware des Hubschraubers. Diese Software kann sowohl auf dem Boardcomputer als auch in einem Testsystem ausgeführt werden. Als Testsystem wird dabei eine Matlab/Simulink-Simulation eingesetzt mit der virtuelle Testflüge durchgeführt werden können. Durch Auswertung der Testflüge wird die Performance der unterschiedlichen implementierten Strategien zur Anpassung des Kamerablickwinkels bestimmt und darauf basierend ein Vergleich dieser dargestellt.

2 Grundlagen und Testsystem

In diesem Kapitel werden zunächst die Grundlagen der Navigation von unbemannten Luftfahrzeugen mit Fokus auf die optische Navigation dargestellt. Dazu zählt insbesondere die Beschreibung von Kalman-Filtern, die häufig zur Datenfusionierung in der Navigation für UAVs eingesetzt werden. Danach werden Aspekte und Strategien für die dynamische Anpassung des Kamerablickwinkels während der Navigation beschrieben. Abschließend werden Konfiguration sowie Simulationsumgebung für den als Testsystem im Rahmen dieser Arbeit eingesetzten superArtis Kleinhaubschraubers des Instituts für Flugsystemtechnik des Deutschen Zentrums für Luft- und Raumfahrt (DLR) dargelegt.

2.1 Navigation für UAVs

Für die Navigation eines UAVs ist die Bestimmung der aktuellen Position beziehungsweise des gesamten Flugzustands von entscheidender Bedeutung. Der Zustand wird dabei durch die Position des UAVs, also einen Punkt im Raum, einen Geschwindigkeitsvektor sowie die räumliche Lage beschrieben. Der Zustand wird auf Basis der Messdaten einer inertialen Messeinheit zeitlich vorwärts propagiert, sodass eine Schätzung des aktuellen Zustands vorliegt. Dabei akkumulieren sich über die Zeit Abweichungen zwischen dem so geschätzten und dem tatsächlichen Zustand des UAVs durch Messfehler beziehungsweise Drift. Auch die Fähigkeit zur Navigation in unbekannten Umgebungen auf Basis dieser einfachen Zustandsschätzung ist stark eingeschränkt.[BVN16][Jha16][GG17][Con18]

Um die Abweichungen zwischen geschätztem und tatsächlichem Zustand des UAV zu charakterisieren und zu korrigieren, können weitere Sensoren eingesetzt werden. Die mit diesen erhaltenen zusätzlichen Informationen können zur Verbesserung der Zustandsschätzung genutzt werden (s. 2.1.2). Dabei können zum Beispiel Magnetometer, globale Navigationssatellitensysteme, Barometer oder optische Sensoren eingesetzt werden. Letztere sind hier besonders geeignet, wobei vor allem Kameras und Lidar-Sensoren zum Einsatz kommen. Im Rahmen der optischen Navigation müssen die Daten dieser Sensoren vorverarbeitet werden, um eine passende Eingabe für den Zustandsschätzer zu erhalten. Diese Vorverarbeitung liefert Positionen von Objekten beziehungsweise Positionsänderungen aus Bildsequenzen (s. 2.1.1). Letzteres wird als visuelle Odometrie bezeichnet. Auch ist es mit optischen Verfahren möglich Informationen über zuvor unbekannte Umgebungen zu erfassen. Werden die Objekte, deren Position in der Vorverarbeitung aus den Bilddaten extrahiert wird, erst während der Navigation entdeckt, so wird dieses Verfahren als simultane Po-

sitionsbestimmung und Kartierung (*engl.* Simultaneous Localization and Mapping, SLAM) bezeichnet.[BVN16][Jha16][GG17][Con18]

Im Folgenden wird zunächst dargestellt, wie Objekte in Kamerabildern erkannt und wiedererkannt werden und wie anhand dieser Referenzpunkte Informationen über den Zustand des UAVs ermittelt werden können. Anschließend wird diskutiert, wie diese Informationen zusammen mit anderen Sensordaten und den Steuereingaben mit der Zustandsschätzung aus den Messungen der IMU mithilfe eines Kalman-Filters fusioniert werden können, um die Schätzung des Zustands des Vehikels zu optimieren.

2.1.1 Detektion und Zustandsbestimmung durch Referenzpunkte

Im Kamerabild sichtbare Objekte können als Referenzpunkte für die optische Navigation verwendet werden und dabei in Landmarken und Feature unterteilt werden. Landmarken sind im Voraus bekannte Objekte, deren absolute Position in einem Weltkoordinatensystem bekannt ist, sodass ein Katalog der nutzbaren Landmarken vorliegt. Im Gegensatz dazu stellen Feature willkürliche, markante Bildbereiche dar, deren absolute Position im Vorhinein unbekannt ist, sodass Feature vor allem beim SLAM genutzt werden. Bei einem geeigneten Referenzpunkt handelt es sich um einen dreidimensionalen Punkt mit fester Position im Raum. Nur dann ist die Veränderung der Position des Referenzpunktes in aufeinanderfolgenden Kamerabildern allein auf die Zustandsänderung des UAVs zurückzuführen. Dies ist für Landmarken immer gegeben, während Feature als markante Bildbereiche auch durch eine visuelle Überlagerung mehrerer Objekte im Raum unter dem aktuellen Kamerablickwinkel entstehen und somit keinem fixen Punkt im dreidimensionalen Raum zugeordnet werden können. Solche Feature müssen im Verlauf der Navigation dann wieder aufgegeben werden.[BEF96]

Zur Nutzung von Referenzpunkten beider Typen für die Positionsschätzung ist deren initiale Identifikation in einem Kamerabild erforderlich. Typischerweise werden als Feature markante Bereiche mit starken Gradienten in einer oder zwei Richtungen genutzt. Für die Detektion solcher Kanten und Ecken existieren diverse effiziente Verfahren.[BEF96][Li17] Verfahren zur Erkennung von Landmarken müssen dem gegenüber auf die Charakteristika der jeweiligen Landmarken zugeschnitten werden. So können in begrenzten Bereichen zum Beispiel spezielle *Beacons* (*dt.* Leuchtfeuer) installiert werden, die mit einem Symbol wie etwa einem QR-Code markiert sind und anhand diesem in Bildern identifiziert werden können.[BEF96] AprilTags (Abb. 1) stellen eine Klasse speziell für die optische Navigation entwickelter Landmarken dar,

für die Verfahren zur Identifikation auf Bildern zur Verfügung stehen, mit denen auch die Lage, Orientierung und Identität der Landmarken präzise bestimmt werden können.[Ols11][WO16] Für UAVs bieten sich auch markante Geländestrukturen als Landmarken an. Insbesondere für Satelliten stellen Krater von Himmelskörpern geeignete Landmarken dar.[MKT11][Maa16] Auch Sternpositionen können als Landmarken für die Lagebestimmung eingesetzt werden.[Lie95]

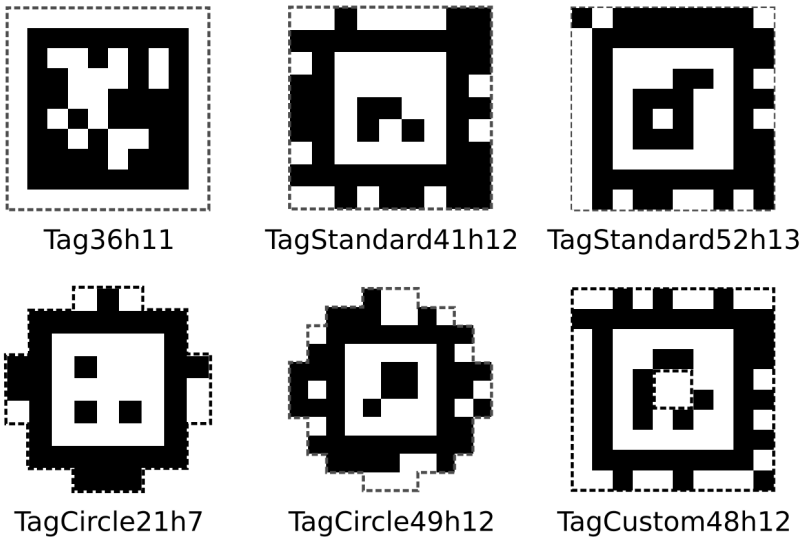

Abbildung 1: Exemplarische AprilTags in verschiedenen Formaten.[Ols10]

Mithilfe von Landmarken kann die Pose der Kamera durch Korrespondenzen zwischen deren Position in einem einzelnen Bild und deren bekannter dreidimensionaler Position im Weltkoordinatensystem ermittelt werden. Dieses als *Perspective-n-Point* bekannte Problem kann mit mindestens vier Korrespondenzen eindeutig gelöst werden und es existiert mit dem *Efficient Perspective-n-Point*-Algorithmus (EPnP) von Lepetit *et al.* ein effizienter Algorithmus für dessen Lösung.[LMNF08] Im Rahmen eines Kalman-Filters (s. 2.1.2) bietet jede identifizierte Landmarke zusätzliche Informationen über den Zustand des UAVs, da die tatsächlich gemessene Position im Kamerabild mit der erwarteten Messung auf Basis der bekannten absoluten Position verglichen wird.

Für Feature, deren Positionen im Weltkoordinatensystem nicht bekannt sind, kann aus einem einzigen Kamerabild hingegen keine Information über den Zustand des UAVs abgeleitet werden. Für beide Arten von Referenzpunkten kann jedoch die Veränderung der Position in aufeinanderfolgenden Bildern genutzt werden, um Informationen über die Zustandsänderung abzuleiten. Diese Nachverfolgung von Referenzpunkten in Bildsequenzen kann entweder über die Identifikation von Bildbereichen maximaler Ähnlichkeit erreicht werden (*engl.* Feature Tracking), oder dadurch,

dass Feature in Einzelbildern unabhängig voneinander erkannt und danach mit entsprechender Metrik verglichen werden, um wahrscheinliche Übereinstimmungen zu ermitteln (*engl.* Feature Matching).[ST94][Can08]

Anhand der Positionsänderungen korrespondierender Referenzpunkt in Sequenz von Bildern kann der optische Fluss berechnet werden. Der optische Fluss stellt ein Vektorfeld der Bewegung der auf den Kamerabildern sichtbaren Objekte dar. Da geeignete Referenzpunkte im Weltkoordinatensystem feste dreidimensionale Punkte sind, können aus dem optischen Fluss somit Rückschlüsse auf die relative Bewegung der Kamera und somit des UAVs in Bezug auf die Umgebung gezogen werden.[MA13]

2.1.2 Kalman-Filter

Kalman-Filter stellen eine Klasse von so genannten State-Observer-Verfahren dar, mit denen der Zustand eines Systems iterativ geschätzt werden kann. Dabei stellen fehlerbehaftete Messgrößen aus denen der Systemzustand abgeleitet werden kann, die Eingabe dar. Der klassische Kalman-Filter stellt einen idealen Schätzer für Systeme dar, in denen nur lineare Abhängigkeiten zwischen den einzelnen Größen existieren, die den Systemzustand beschreiben. Die Optimalität bezieht sich dabei auf die Minimierung der Varianz des Fehlers der Zustandsschätzung.[Kal60] Die Darstellung in diesem Abschnitt basiert, wenn nicht anderweitig angegeben, auf [Sim06] sowie zusätzlich auf [Sar07] für die Beschreibung des Unscented Kalman-Filters .

Der Zustand des Systems wird durch den Zustandsvektor x zum Zeitpunkt k, kurz x_k, beschrieben. Hierbei werden in der Regel diskrete Zeitpunkte mit festen Abständen Δt betrachtet. Im Falle linearer Abhängigkeiten zwischen den einzelnen Elementen des Zustandsvektors kann dieser mittels

$$x_k = F_{k-1}x_{k-1} + B_{k-1}u_{k-1} + w_{k-1} \tag{1}$$

beschrieben werden. In dieser linearen Differenzengleichungen beschreibt die Matrix F_{k-1} den Übergang zwischen den aufeinanderfolgenden Systemzuständen x_{k-1} und x_k und enthält somit die Zustandsgleichung, die die eigentliche Systemdynamik beschreibt. Darüber hinaus wirken deterministische Einflüsse, wie etwa Steuerbefehle, auf das System. Deren Ausmaß für den betrachteten Zeitpunkt wird durch den Vektor u_{k-1} beschrieben, während die Matrix B_{k-1} die zugehörigen Zustandsgleichungen beschreibt. Letztlich werden noch statistische Abweichungen durch den Rauschterm w_{k-1} definiert. Dieses Prozessrauschen wird hierbei als zeitlich unkorreliert modelliert und durch eine mehrdimensionale Normalverteilung mit Mittelwert 0 beschrieben. Entsprechend $w_{k-1} \sim \mathcal{N}(0, Q_{k-1})$ wird die Verteilung durch die ebenfalls zum Filter gehörende Kovarianzmatrix Q_{k-1} beschrieben.

Eine Beobachtung der messbaren Größen z_k findet im Rahmen dieses Modells entsprechend

$$z_k = H_k x_k + v_k \tag{2}$$

statt. Der wahre Systemzustand x_k wird durch die Matrix H_k in die Messgrößen überführt, wobei die Matrix das Beobachtungsmodell eben dieser Messungen repräsentiert. Hinzu kommt der statistische Fehler der Messung, welcher durch v_k beschrieben und durch weißes Rauschen entsprechend einer Normalverteilung mit Mittelwert 0 und der Kovarianzmatrix R_k nach $v_k \sim \mathcal{N}(0, R_k)$ modelliert wird. Die Komponenten des Ausgangszustands sowie die verschiedenen Rauschbeiträgen werden als paarweise unabhängig angenommen. Ein Schema der verschiedenen Größen und Abhängigkeiten des dem Kalman-Filter zugrunde liegenden Modells zeigt Abbildung 2.

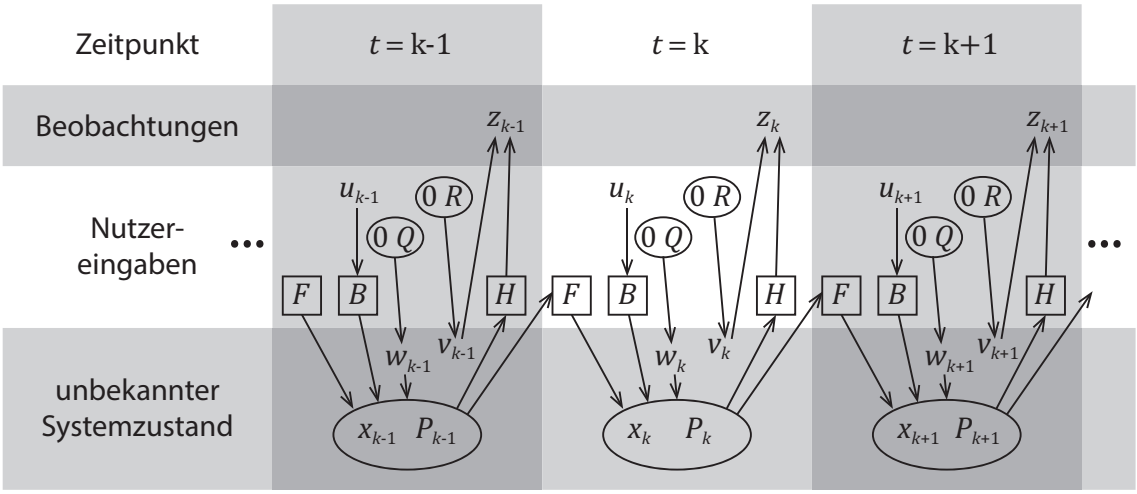

Abbildung 2: Schematischer Zeitverlauf der verschiedenen Größen und ihrer Abhängigkeiten im Modell des Kalman-Filters. Multidimensionale Normalverteilungen mit Mittelwert und Kovarianz sind in Ellipsen, Matrizen in Rechtecken und Vektoren ohne Umrandung dargestellt. Nach [Hea09].

Ziel des Filters ist es, den unbekannten wahren Systemzustand entsprechend dieses Modells auf Basis der Beobachtungen und der Beschreibung des Systems zu schätzen. Der Kalman-Filter ist ein rekursiver Schätzer, was bedeutet, dass der aktuelle Zustand ausschließlich auf Basis der aktuellen Messungen sowie der letzten Schätzung durchgeführt wird. Das Vorhalten von Informationen aus vorherigen Messungen ist somit nicht erforderlich. Hierbei wird die Schätzung von x zum Zeitpunkt n auf Basis der Beobachtungen bis zum Zeitpunkt m ($m \leq n$) als $\hat{x}_{n|m}$ dargestellt. Der Zustand des Filters wird dann durch die *a posteriori* Schätzung $\hat{x}_{k|k}$

sowie deren Kovarianz $P_{k|k}$ repräsentiert. Letztere ist ein Maß für die Genauigkeit der Zustandsschätzung.

Die Arbeitsweise des Filters kann in drei Phasen unterteilt werden. In der Initialisierungsphase werden $\hat{x}_{0|0} = 0$ und $P_{0|0} = \sigma^2 \cdot I$ initialisiert, wobei die zugeordneten Werte lediglich eine typische Initialisierung darstellen, bei der σ^2 eine geeignete Varianz ist.[SOL81] Darauffolgend werden die Prädikations- beziehungsweise Vorhersage- und die Korrekturphase abwechselnd durchlaufen. Für die Prädikation gelten die Gleichungen

$$\begin{aligned} \hat{x}_{k|k-1} &= F_{k-1}\hat{x}_{k-1|k-1} + B_{k-1}u_{k-1} \\ P_{k|k-1} &= F_{k-1}P_{k-1|k-1}F_{k-1}^{\top} + Q_{k-1} \end{aligned} \tag{3}$$

für die Vorhersage des *a priori* Zustands. Die Korrektur wird durch die Gleichungen

$$\begin{aligned} K_k &= P_{k|k-1}H_k^{\top}(H_kP_{k|k-1}H_k^T + R_k)^{-1} \\ \hat{x}_{k|k} &= \hat{x}_{k|k-1} + K_k(z_k - H_kx_{k|k-1}) \\ P_{k|k} &= (I - K_kH_k)P_{k|k-1} \end{aligned} \tag{4}$$

beschrieben. Hierbei stellt K_k die sogenannte Kalman-Gain-Matrix dar, welche die Gewichtung der Residuen für die Korrektur des Systemzustands angibt. Der Ausdruck $(H_kP_{k|k-1}H_k^T + R_k)$ in der ersten Korrekturgleichung wird auch Residualkovarianz S_k genannt. Der Ausdruck zur Gewichtung der Kalman-Gain-Matrix in der zweiten Korrekturgleichung $(z_k - H_kx_{k|k-1})$ wird als Innovation $\tilde{y}_k$ bezeichnet und gibt die Genauigkeit der Vorhersage des aktuellen Mittelwerts durch die Beobachtungsgleichungen an.

Um auch Systeme mit nicht-linearem Verhalten hinreichend modellieren zu können, sind Erweiterungen das Kalman-Filters erforderlich. Hierbei treten an die Stelle der Gleichungen 1 und 2 die Gleichungen

$$\begin{aligned} x_k &= f(x_{k-1}, u_k) + w_k \\ z_k &= h(x_k) + v_k \end{aligned} \tag{5}$$

mit den nicht-linearen Funktionen f und h. Eine solche Erweiterung stellt der Unscented Kalman-Filter (UKF) dar, bei dem mit einem deterministischen Verfahren eine Stichprobe von Punkten, den so genannten Sigma-Punkten, ermittelt wird. Durch die Anwendung der nicht-linearen Zustandsgleichungen auf die Sigma-Punkte werden Mittelwert und Kovarianz des Systemzustands bestimmt.[JU97] Die konkrete Ausprägung des UKF hängt von Verfahren ab, mit dem die Stichprobe ermittelt wird. Allgemein wird für einen L-komponentigen Vektor $x = (x_1, ..., x_L)$ eine Menge $\{s_0, ..., s_N\} = \{(s_{0,1}, ..., s_{0,L}), ..., (s_{N,1}, ..., S_{N,L})\}$ mit N L-komponentigen Vektoren als Sigma-Punkte bezeichnet, wenn dieser Menge Gewichte erster Ordnung

$W_a^0, ..., W_a^N$ zugeordnet sind, für die

$$\begin{aligned} \sum_{j=0}^{N} W_a^j &= 1 \\ E[x_i] &= \sum_{j=0}^{N} W_a^j s_{j,i} \; \forall i \in \{1, ..., L\} \end{aligned} \tag{6}$$

gilt sowie Gewichte zweiter Ordnung $W_c^0, ..., W_c^N$ für die

$$\begin{aligned} \sum_{j=0}^{N} W_c^j &= 1 \\ E[x_i, x_l] &= \sum_{j=0}^{N} W_c^j s_{j,i} s_{j,l} \; \forall (i,l) \in \{1, ..., L\}^2 \end{aligned} \tag{7}$$

gilt. Die Prädikation erfolgt unter Anwendung der Übergangsfunktion f auf die Sigma-Punkte $s_{k-1|k_1}$ bezüglich des Zustands zum Ausgangszeitpunkt $k-1$ entsprechend $s_{k|k-1}^j = f(s_{k-1|k-1}^j)$ mit

$$\begin{aligned} \hat{x}_{k|k-1} &= \sum_{j=0}^{2L} W_a^j s_{k|k-1}^j \\ P_{k|k-1} &= \sum_{j=0}^{2L} W_c^j (s_{k|k-1}^j - \hat{x}_{k|k-1})(s_{k|k-1}^j - \hat{x}_{k|k-1})^\top + Q_k \end{aligned} \tag{8}$$

als Berechnungsvorschrift, wobei typischerweise $N = 2L+1$ Sigma-Punkte generiert werden. Für die Korrektur wird die Funktion h auf die bereits transformierten Sigma-Punkte $s_{k|k-1}$ angewandt. Mit $g_k^j = h(s_{k|k-1}^j)$ ergeben sich die so genannten Gamma-Punkte, deren empirischer Mittelwert zu

$$\hat{z} = \sum_{j=0}^{2L} W_a^j g_k^j \tag{9}$$

und die zugehörige Kovarianz zu

$$\hat{S}_k = \sum_{j=0}^{2L} W_c^j (g_k^j - \hat{z})(g_k^j - \hat{z})^\top + R_k \tag{10}$$

berechnet wird, wodurch die erwartete Messung und deren Unsicherheit auf Basis des aktuellen Filterzustands beschrieben ist. Zusätzlich wird die Kreuzkovarianz-Matrix

$$C_{sz} = \sum_{j=0}^{2L} W_c^j (s_{k|k-1}^j - \hat{x}_{k|k-1})(g_k^j - \hat{z})^\top \tag{11}$$

zwischen Zustand und Messung berechnet. Damit kann mittels

$$\begin{aligned} K_k &= C_{sz}\hat{S}_k^{-1} \\ \hat{x}_{k|k} &= \hat{x}_{k|k-1} + K_k(z_k - \hat{z}) \\ P_{k|k} &= P_{k|k-1} - K_k\hat{S}_k K_k^\top \end{aligned} \tag{12}$$

die Korrektur für den UKF beschrieben werden.

2.2 Anpassung des Blickwinkels

In der optischen Navigation sind die eingesetzten Kameras typischerweise fest am zu navigierenden Vehikel angebracht. Entsprechend steht die Kamera in einem festen Bezug zum Vehikel und dessen Pose kann direkt in die Pose des Vehikels überführt werden. Die Transformation zwischen dem Vehikel-zentrischen Koordinatensystem (*engl.* body fixed frame, BFF) und dem Sensor-zentrischen Koordinatensystem (*engl.* sensor fixed frame, SFF) ist somit konstant. Im Gegensatz dazu besteht im Rahmen von active vision die Möglichkeit mit einer entsprechenden Mechanik den Blickwinkel beziehungsweise die Ausrichtung der Kamera relativ zum Vehikel dynamisch anzupassen, um somit interessante oder relevante Bereiche (*engl.* regions of interest, ROIs) gezielt ins Blickfeld der Kamera zu rücken oder sukzessive einen größeren Bereich der Szene betrachten zu können.[Dav98][DM98]

Auch für die Navigation unter Einsatz von active vision wird die Detektion von und Orientierung an Referenzpunkten als grundlegende Technik genutzt (s. 2.1.1). Hierbei muss die Ausrichtung der Kamera relativ zum Vehikel berücksichtigt werden, da die Transformation von BFF zu SFF nun variable ist. Durch die aktive Steuerung der Positionierungsmechanik ist diese jedoch im Rahmen der Stellgenauigkeit der Mechanik bekannt. Als fehlerbehaftete Größe kann die Ausrichtung der Kamera relativ zum Vehikel auch in das Modell des Kalman-Filters (s. 2.1.2) integriert werden. Eine zusätzliche Herausforderung stellt auch die Abwägung dar, ob bekannte Feature weiterhin im Blickfeld behalten werden sollen oder zu Gunsten anderer Referenzpunkte der Blickwinkel angepasst werden sollte. Dazu muss zunächst die Qualität einzelner sichtbarer Referenzpunkte bewertet werden. Maße dazu werden im folgenden Abschnitt dargestellt. Darauffolgend werden bekannte Strategien für die Wahl des Blickwinkels diskutiert.[Dav98][DM98]

2.2.1 Bewertung von Referenzpunkten

Bei der Bewertung von Referenzpunkten muss zwischen deren intrinsischer Qualität sowie deren Qualität für die Positionsschätzung unterschieden werden. Als intrin-

sische Qualität wird der geschätzte Fehler der Position eines Referenzpunkten im Weltkoordinatensystem bezeichnet. Solche Fehler sind in der Regel technisch bedingt und werden etwa durch die Kameraauflösung oder die Winkelauflösung der Positionierungsmechanik der Kamera bestimmt. Diese Fehler pflanzen sich dann entsprechend des Verfahrens zur Positionsbestimmung der Feature fort. Das wiederholte Beobachten eines Features kann dabei zu einer Verbesserung der intrinsischen Qualität führen. Folglich ist die intrinsische Qualität von Landmarken optimal, da deren Position im Weltkoordinatensystem bereits im Vorfeld bekannt ist.[BEF96][Dav98]

Die Qualität eines Referenzpunkts für die Positionsschätzung hingegen gibt an, wie sich die Betrachtung dieses Referenzpunkts auf die Positionsschätzung auswirkt. Dabei können zu jedem Zeitpunkt nur Referenzpunkte berücksichtigt werden, von denen erwartet wird, dass sie von der aktuellen Position aus sichtbar sind.[Dav98]

Ein möglicher Ansatz bei bekannten Messunsicherheiten beruht auf der Strategie immer die Messung durchzuführen, bei der die höchste Unsicherheit für das Messergebnis vorliegt. Mit der Durchführung dieser Messung kann somit die Gesamtunsicherheit voraussichtlich maximal reduziert werden.[WF97] Im Rahmen der Positionsschätzung mithilfe eines Kalman-Filters kann diese Unsicherheit bestimmt werden, da sich mithilfe des Messmodels H_k das erwartete Messresultat ergibt. Die Residualkovarianzmatrix S_k beschreibt die für eine solche Messung zu erwartende Abweichung zwischen dem tatsächlichen und dem zu erwartenden Messresultat (s. 2.1.2). Als Skalar zur Bewertung wird von Davison und Murray das Volumen V_S des 3σ-Ellipsoids von S_k vorgeschlagen, wobei für das Feature mit dem höchsten Wert V_S die größte Verbesserung der Positionsschätzung erwartet wird.[DM98]

Die Eignung eines Referenzpunkts für die Positionsbestimmung kann auch durch den Informationsgewinn beschrieben werden. Hierbei handelt es sich um ein informationstheoretisches Maß auf Basis der Entropie H, die für eine Variable x einer n-variaten Normalverteilung mit der Kovarianzmatrix P nach

$$H(x) = \frac{1}{2} \log \left((2\pi e)^n |P| \right) \tag{13}$$

gegeben ist. Der Informationsgewinn $I[x, z]$ durch eine Beobachtung z ergibt sich aus der Differenz der Entropien vor und nach der Beobachtung. Er kann somit nach

$$I[x, z] = H(x) - H(x|z) = -\frac{1}{2} \log \left(\frac{|P(x|z)|}{|P(x)|} \right) \tag{14}$$

berechnet werden. Der Beobachtung, die den Informationsgewinn maximiert, wird die höchste Qualität für die Positionsbestimmung zugeordnet.[BS05][BS08]

Eine zum Informationsgewinn ähnliche Strategie auf Basis bayesscher Statistik stellt der erwartete Nutzen EU (*engl.* expected utility) dar. Nach der Formel

$$EU(a_k|x_j) = \sum_{l=1}^{q} P(z_l|x_j)U(a_k, z_l) \tag{15}$$

ergibt dieser sich aus der Wahrscheinlichkeit P, dass der Zustand z_l entsprechend der Beobachtung x_j vorliegt sowie dem Nutzen U, den das Ausführen der Aktion a_k bei Vorliegen des Zustands z_l hat. Der Nutzen beinhaltet dabei auch die Kosten, die etwa für die Positionierung der Sensoren für die Aktion a_k notwendig sind.[Kri97][dCSKP09]

Auch das Konzept der Salienz kann zur Bewertung der Qualität von Referenzpunkten eingesetzt werden. Die Salienz oder visuelle Auffälligkeit eines Referenzpunkts steht dabei nicht direkt im Zusammenhang mit der Eignung für die Positionsbestimmung, sondern wird eingesetzt, um möglichst auffällige und somit robuste Referenzpunkte zu finden. Die Salienz kann in einer biologisch inspirierten Art auf Basis von Form und Farbkontrast bestimmt werden.[BMB01] Auch der Informationsgehalt eines Featuretyps gewichtet mit seiner Häufigkeit kann als Salienz interpretiert werden.[KE13][KE14]

2.2.2 Nachverfolgung von Referenzpunkten

In der Literatur beschriebene Strategien zur Anpassung der Blickwinkels basieren in der Regel auf der Nachverfolgung von Referenzpunkten über einen gewissen Zeitraum durch Nachführen der Kamera. Für eine solche Strategie ist nicht nur die Qualität eines Referenzpunkts für die Positionsschätzung von Bedeutung, sondern auch das Gesamtsystem. Entsprechend muss in jedem Fall das Positionierungssystem berücksichtigt werden, da Änderungen des Blickwinkels nicht instantan erfolgen können. Darüber hinaus kann beim Vorliegen eines übergeordneten Ziels auch eine Abwägung zwischen Zielerfüllung und Positionsschätzung notwendig sein. Besteht dieses übergeordnete Ziel in einem SLAM-Ansatz, so ist häufig eine Phase der Featureentdeckung in Situationen nötig, in denen die Positionsschätzung mit den aktuell bekannten Features nicht zufriedenstellend möglich ist oder in denen die aktuelle Karte erweitert werden soll. Diese Phasen können durch das Unterschreiten eines Grenzwerts von sichtbaren Features eingeleitet werden und folgen in der Regel einer festgelegten Strategie für das Absuchen von Bereichen nach neuen Features.[Dav98]

Eine mögliche Strategie zur Auswahl des Blickwinkels stellt die Berücksichtigung der Anzahl von Messungen dar, die nicht genutzt werden können, wenn die Kamera zu einem bekannten Referenzpunkt bewegt wird. Dazu wird der Zeitraum

in der Anzahl von Messung $N_{\max}$ berücksichtigt, der für die längste Bewegung benötigt wird, um die Kamera auf einen aktuell sichtbaren Referenzpunkt zu richten. Für den mit dieser Bewegung erreichten Referenzpunkt steht nach der Positionierung Zeit für genau eine Messung zur Verfügung. Wird der Blickwinkel hingegen nicht geändert, können vom aktuell beobachteten Referenzpunkt $N_{\max} + 1$ Messung durchgeführt werden. Für Referenzpunkte die nach einer kleineren Anzahl N_i von Messdauern erreicht werden, können entsprechend nach der Positionierung $N_{\max} - N_i + 1$ Messungen durchgeführt werden. Mithilfe eines entsprechenden Qualitätsmaßes können nun die jeweiligen Aktionen bestehend aus Kamerapositionierung und Messung für jeden Referenzpunkt bewertet werden. Für die unter 2.2.1 beschriebenen Qualitätsmaße kann zum Beispiel V_S nach der jeweiligen gesamten Aktion ausgewertet werden.[Dav98][DM98] Auch der Informationsgewinn kann auf Aktionen $a \in \{x(k), z(k), x(k+1), z(k+1), ..., x(k+n), z(k+n)\}$ als Folge von Zuständen x und Beobachtungen z angewandt werden.[BS08]

Eine Möglichkeit zur Abwägung zwischen der Positionsschätzung und der übergeordneten Strategie der Umgebungserkundung beim SLAM stellt die getrennte Betrachtung des Informationsgewinns für die Positionszustand $I[x_v, z]$ und des Kartenzustands $I[x_m, z]$ dar. Die als absolute Nützlichkeit

$$U = w_v I[x_v, z] + w_m I[x_m, z] \tag{16}$$

bezeichnete Bewertung einer Aktion kann somit durch eine gewichtete Summe der Informationsgewinne mit den Gewichten w_v und w_m bestimmt werden. Das Verhältnis der Gewichte ist dabei geeignet graduell zwischen einer explorativen und einer konservativen, die Positionsschätzung optimierenden, Strategie zu wechseln.[BS08]

Ein weiterer Strategietyp, der auch das übergeordnete Ziel der Exploration verfolgt, wird als *next-best-view* bezeichnet. Hierbei werden für die Entdeckung neuer Feature immer solche Blickwinkel ausgewählt, in denen die maximale zusätzliche Information erwartet wird. In einer gewichteten Abwägung gegenüber der Positionsschätzung kann mit dieser Strategie ebenfalls eine effiziente Exploration realisiert werden.[Con85][KE12]

Im Rahmen des erwarteten Nutzens (s. 2.2.1) werden die Kosten einer Aktion für die Bestimmung der besten Aktion bereits berücksichtigt. Für die Entscheidungsstrategie kann jedoch auch ein aus den Wirtschaftswissenschaften bekanntes Konzept der Optimierung des Nutzen-zu-Kosten-Verhältnisses eingesetzt werden. Dazu wird der Abstand des erwarteten Nutzens einer Aktion vom mittleren erwarteten Nutzen aller möglichen Aktionen durch die Kosten der Aktion geteilt. Der so

genannte *expected interst from sample information*-Wert liefert somit die Aktion, die den höchsten Nutzen pro eingesetzter Kosten verspricht.[Kri97]

2.3 superArtis Kleinhubschrauber

In diesem Abschnitt wird zunächst der superArtis Kleinhubschrauber beschrieben, der zur Erprobung von Navigationsverfahren am DLR genutzt wird. Dabei wird zuerst die Hardware mit Fokus auf das Sensorsystem des Hubschraubers vorgestellt. Daran anschließend wird die Steuer- sowie Navigationssoftware detailliert beschrieben. Abschließend wird die Funktionsweise der Test- beziehungsweise Simulationsumgebung erläutert, in der auch die im Rahmen dieser Arbeit erarbeiteten Verfahren zur Anpassung des Kamerablickwinkels während der Navigation evaluiert wurden.

2.3.1 Hardware und Sensorsystem

Beim superArtis Kleinhubschrauber (Abb. 3) handelt es sich um das Modell SDO 50 V2 der Firma SwissDrones. In Tabelle 1 sind einige technische Spezifikationen des Hubschraubers zusammengefasst. Dabei ist insbesondere die bezüglich des Eigengewichts hohe Nutzlast hervorzuheben, die es ermöglicht die im Folgenden beschriebene Hardware zu transportieren.

Maße $l \times b \times h$	Leergewicht	Nutzlast	Flugzeit	Geschwindigkeit
$2.32\,\mathrm{m} \times 0.7\,\mathrm{m} \times 0.92\,\mathrm{m}$	$42\,\mathrm{kg}$	$45\,\mathrm{kg}$	$\leq 3.1\,\mathrm{h}$	$\leq 20\,\mathrm{m/s}$

Tabelle 1: Technische Spezifikationen des SwissDrones SDO 50 V2 Hubschraubers.[Swi20]

Das Kernstück der Sensorik stellt die IMU der Firma iMAR vom Typ iTraceRT-F400-Q-E dar. Der MEMS-Beschleunigungssensor sowie das Laserkreisel-Gyroskop sind mit der ebenfalls in der IMU integrierten GPS-Einheit gekoppelt. Die IMU liefert sowohl die Rohdaten der einzelnen enthaltenen Sensoren sowie den Flugzustand auf Basis einer internen Fusionierung dieser Rohdaten. Die Ausgabe dieser Informationen erfolgt mit einer Frequenz von bis zu 400 Hz. Wertebereiche und Genauigkeiten des Gyroskops sowie des Beschleunigungssensors zeigt Tabelle 2.

Für die Untersuchung zur Anpassung des Kamerablickwinkels für die Navigation ist das Kamerasystem entscheidend, welches in Abbildung 4 dargestellt ist. Es sind zwei Kameras des Herstellers Allied Vision vom Typ Prosilica GT1380 am Hubschrauber angebracht. Die Kameras haben eine maximale Auflösung von

Abbildung 3: Der superArtis Hubschrauber des DLR mit am Gestell montierter Hardware und Sensoren.

	Gyroskop	Beschleunigungssensor
Messbereich	$\pm 450°/\mathrm{s}$	$\pm 5\,\mathrm{g}$
Genauigkeit (1σ)	$0.75°/\mathrm{h}$	$2\,\mathrm{mg}$
Messrauschen	$< 0.1°/\sqrt{h}$	$50\,\mu\mathrm{g}/\sqrt{\mathrm{Hz}}$

Tabelle 2: Spezifikationen des Gyroskops und des Beschleunigungssensors in der iTraceRT-F400-Q-E IMU. Die angegebenen Werte werden ohne die Fusionierung der Messdaten mit denen des GPS Moduls erreicht.[iG13]

$1360 \times 1024\,\mathrm{px}$ bei einer maximalen Wiederholrate von $30.5\,\mathrm{Hz}$.[Ima15] Im Sensorsystem werden die Kameras über einen Trigger ausgelöst, sodass der Aufnahmezeitpunkt der Bilder genau bestimmt und synchronisiert werden kann.

Im Rahmen dieser Arbeit wird jedoch nur eine Kamera betrachtet, wobei als Linse XXX mit einer Brennweite von YY eingesetzt wird. Die Auflösung wird auf $800 \times 600\,\mathrm{px}$ festgelegt und die getriggerte Bildaufnahme findet mit einer Wiederholrate von $5\,\mathrm{Hz}$ statt. Zusätzlich zur Kamera ist ein Laserscanner der Marke Velodyne Lidar vom Typ Puck installiert. Dessen Messbereich ist auf ein FOV von horizontal $\pm 45°$ und vertikal ± 15 beschränkt. Mit dem Sensor sind Abstandsmessungen in einem Bereich von 1–100 m mit einer Genauigkeit von $\pm 0.03\,\mathrm{m}$ möglich.[Lid19] Die Kamera und der Laserscanner sind an einer Schwenk-Kipp-Einheit (*engl.* Pan-Tilt-Unit, PTU) montiert, sodass sich das Zentrum ihrer FOVs überlagert. Es wird eine PTU des Herstellers Teledyne FLIR vom Typ PTU-D48E eingesetzt. Diese ist so mon-

tiert, dass die Nullpunkte der Pan- und der Tilt-Achse entlang der Vorwärtsachse orientiert sind. Die PTU kann um je $\pm 120°$ nach links und rechts geschwenkt und um insgesamt 120° gekippt werden. Mit einem Tilt von 0° wird die Orientierung entlang der Vorwärtsachse, mit 90° entlang der Abwärtsachse beschrieben. Der Blick nach unten stellt auch die maximale Auslenkung der PTU in diese Richtung dar. Im Tilt-Winkelbereich von $[-30°, 0°)$ wird die Kamera über den Blick nach vorne leicht nach oben verkippt. Die PTU ist so konfiguriert, dass sie mit einer Maximalgeschwindigkeit von 60°/s in Pan- und 25°/s in Tiltrichtung bewegt werden kann.[Fli21]

Abbildung 4: Kameras und Laserscanner montiert an der PTU zur Anpassung des Blickwinkels.

Die auf dem superArtis aktuell nicht vorhandenen Sensoren Barometer und Magnetometer können während Testflügen emuliert werden. Dazu wird die fusionierte Ausgabe der IMU zur Position als wahr angenommen und auf Basis dieser Position Sensordaten berechnet. Die Barometerdaten werden dabei auf Basis der ICAO-Standardatmosphäre berechnet.[Org93] Der bestimmt Luftdruck wird mit weißem Rauschen sowie einer konstanten Abweichung in die simulierten Sensordaten umgerechnet, die mit einer Frequenz von 50 Hz geliefert werden. Darüber hinaus wird die Position mithilfe des *World Magnetic Model*s in Messdaten des Magnetometers umgerechnet, welche ebenfalls mit weißem Rauschen zur emulierten Sensorausgabe verrechnet werden.[fEI19] Die Ausgaberate des emulierten Magnetometers beträgt 20 Hz.

Die Steuerung des Hubschraubers und der Sensoren sowie das Abspeichern und Verarbeiten des Sensordaten (s. 2.3.2) wird auf dem Boardcomputer durchgeführt.

Dieser wurde speziell für den superArtis Hubschrauber konfiguriert und basiert auf dem Mainboard ADLQM87PC der Firma ADL Embedded Solutions, welches mit einer Intel i7 CPU der 4. Generation mit einer Taktrate von 2.4 GHz ausgestattet ist.[Sys18]

2.3.2 Steuer- und Navigationssoftware

Die Steuer- und Navigationssoftware des superArtis Hubschraubers wurde im Institute für Flugsystemtechnik des DLR in der Abteilung für Unbemannte Luftfahrzeuge entwickelt. Sie ist modular aufgebaut und wird per Datenfluss gesteuert. Die verschiedenen Module haben eine feste Anzahl von Ein- und Ausgängen mit dem sie zur Gesamtsoftware verbunden werden. Dazu gibt es für die Module jeweils Steuerparameter, die vor der Programmausführung angepasst werden können und während dieser konstant sind. Die Datenflusssteuerung äußert sich daran, dass die Berechnungen eines Moduls angestoßen werden, sobald neue Daten an seinen Eingängen vorliegen. Eine Übersicht über die einzelnen Module sowie den Datenfluss zwischen ihnen zeigt Abbildung 5.

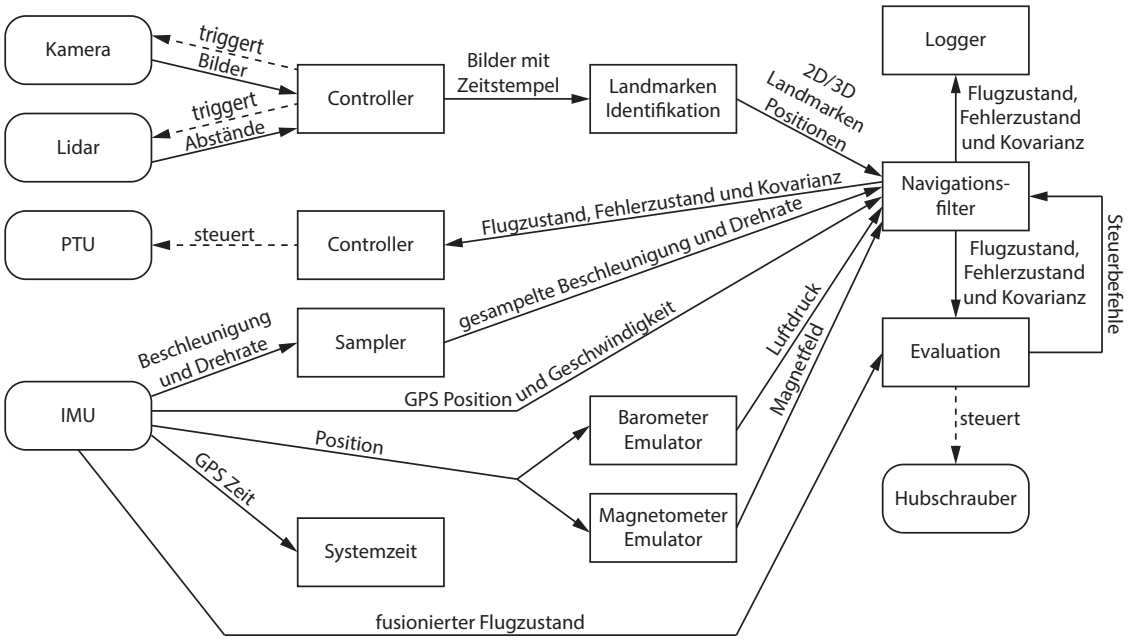

Abbildung 5: Modell des Datenflusses (durchgezogene Pfeile) zwischen den Modulen (Rechtecke) der Steuer- und Navigationssoftware. Auch die Sensoren und Aktoren (abgerundete Rechtecke) und Steuersignale (gestrichelte Pfeile) sind dargestellt.

Der Controller von Kamera und Lidar-Sensor ist dafür zuständig, die Sensoren in vorgegebenen Zeitabständen auszulösen (s. 2.3.1) und die zurückgelieferten Messdaten mit den Zeitstempel des Auslösens zu versehen. Durch die Verwendung von

AprilTags als Landmarken (s. 2.1.1) kann die zu deren Identifikation auf Kamerabildern verfügbare Software eingesetzt werden.[Ols11][WO16] Liegt die Landmarke im Messbereich des Lidar-Sensors, so kann der Abstand der Landmarke zusätzlich ermittelt werden. In diesem Fall ist die 3D Position der Landmarke im SFF bekannt, anderenfalls nur eine 2D Position in Polarkoordinaten ohne Abstand. Diese 2D und 3D Positionen gehen in die Verarbeitung im Navigationsfilter ein (s. 2.3.2.1).

Die IMU liefert verschiedene Daten (s. 2.3.1), die durch unterschiedliche Module weiterverarbeitet werden. Durch den GPS Empfänger wird die aktuelle Zeit erhalten, mit der die Systemzeit synchronisiert wird. Die Rohdaten des Beschleunigungssensors und des Gyroskops werden mit einer Frequenz von 400 Hz erzeugt. Durch den Sampler wird die Frequenz der Messungen, die an den Navigationsfilter weitergeleitet werden reduziert. Dabei wird in der Regel eine Ausgabefrequenz von 100 Hz gewählt, mit der auch der Navigationsfilter arbeitet (s. 2.3.2.1).

Der GPS-Empfänger der IMU liefert ebenfalls Rohdaten, die in den Navigationsfilter eingehen. Dabei wird die Position in der Regel im geographischen Koordinatensystem (Längengrad, Breitengrad und Höhe, *engl.* Latitude, Longitude, Altitude, LLA) und die Geschwindigkeit in North-East-Down-Koordiantensystem (*dt.* Nord-Ost-Abwärts, NED) erhalten. Typische Messraten für die Position sind 2 Hz und für die Geschwindigkeit 5 Hz.

Die in der IMU intern durch Sensorfusion erhaltene Position wird wie unter 2.3.1 beschrieben zur Emulation von Barometer und Magnetometer genutzt.

Der gesamte durch die IMU und ihre interne Datenfusion bestimmte Flugzustand wird im Evaluations-Modul mit dem Flugzustand, den der Navigationsfilter ermittelt hat, verglichen. Die Lösung des Navigationsfilters wird dabei nur zur Steuerung des Hubschraubers berücksichtigt, wenn sie innerhalb vorab gewählter Grenzen mit der Ausgabe der IMU übereinstimmt. Das Evaluationsmodul dient somit als Schutz vor Abstürzen oder ähnlichen katastrophalen Fehlschlägen. Die genutzten Steuerbefehle gehen vom Evaluationsmodul aus wieder in den Navigationsfilter ein, da diese für die zeitliche Propagation des Flugzustands benötigt werden (s. 2.3.2.1).

Der durch den Navigationsfilter ermittelte Flugzustand und dessen geschätzter Fehler gehen auch in den Controller der PTU ein. Das Entwickeln, Implementieren und Testen verschiedener Strategien zur Steuerung der PTU auf Basis dieser Eingabe und etwaiger zusätzlicher Parameter ist das Kernthema dieser Arbeit und wird in den folgenden Kapiteln beschrieben und diskutiert. Für das Datenflussmodell ist hier noch die implizite Abhängigkeit zwischen PTU und Kamera sowie Lidar zu erwähnen. Die Bewegung der PTU wirkt sich auf die von Kamera und Lidar ermittelten Daten durch die Änderung des Blickwinkels aus, auch wenn kein direkter

Datenfluss und auch keine Steuerung stattfindet. Die Kopplung beider Systeme ist durch deren physische Verbindung gegeben (s. 2.3.1).

2.3.2.1 Navigationsfilter Der Navigationsfilter basiert auf einer Kombination von inertialer Navigation und dem bereits in Abschnitt 2.1.2 theoretisch beschriebenen Unscented Kalman-Filter. Im Folgenden wird die konkrete Umsetzung des UKF in der Software des superArtis Hubschraubers erläutert.

Innerhalb des Navigationsfilters gibt es drei elementare Größen. Diese sind der Filterzustand, auch einfach Zustand genannt, der Fehlerzustand und die Kovarianz des Fehlerzustands, die in der Regel nur Kovarianz genannt wird. Diese drei Größen stellen auch die Ausgabe des Navigationsfilters dar.

Der Zustand x wird durch den 10-Vektor

$$x = \begin{bmatrix} r \\ v \\ q \end{bmatrix} \in \mathbb{R}^{10} \tag{17}$$

beschrieben, der die dreidimensionale Position r und die dreidimensionale Geschwindigkeit v enthält. Dazu ist die Lage q enthalten, welche durch ein Quaternion beschrieben wird. Mit jeder neuen Messung des IMU, die im Navigationsfilter eingeht, wird der Zustand in Form des geschätzten Zustands $\hat{x}$ aktualisiert.

Eine Korrektur des Zustands ist notwendig, da sich aufgrund von statistischen und systematischen Mess- und Modellfehlern Abweichungen zwischen dem tatsächlichen Zustand x und dem geschätzten Zustand $\hat{x}$ akkumulieren (s. 2.1). Zu diesem Zweck wird der Fehlerzustand

$$\delta x = \begin{bmatrix} \delta r \\ \delta v \\ \delta\theta \\ b_a \\ b_\omega \\ b_g \\ b_p \end{bmatrix} \in \mathbb{R}^{25} \tag{18}$$

eingesetzt. Dieser besteht aus den jeweiligen Fehlern der Komponenten des Zustands. Diese sind der Positionsfehler δr, der Geschwindigkeitsfehler δv sowie der Lagefehler δq. Um die Forderung nach unkorrelierten Komponenten des Fehlerzustands im Rahmen des UKF (s. 2.1.2) zu erfüllen, wird der Lagefehler durch die notwendige Drehung mit einem Winkel θ um eine Achse e ($e \in \mathbb{R}^3, e \cdot e^\top = 1$) beschrieben, die

dem zu beschreibenden Fehler δq entspricht. In dieser Form sind die drei Komponenten von $\delta\theta = \theta \cdot e$ unabhängig, wohingegen bei δq durch $\|q\| = 1$ eine Abhängigkeit zwischen den Komponenten besteht. Mit

$$\delta q = \begin{bmatrix} e \cdot \sin\left(\frac{\theta}{2}\right) \\ \cos\left(\frac{\theta}{2}\right) \end{bmatrix} \tag{19}$$

kann dieser Fehler $\delta\theta$ in den Lagefehler δq bezüglich der Quaternionen überführt werden.

Im Fehlerzustand werden darüber hinaus auch die systematischen Fehler des Beschleunigungssensors b_a und des Gyroskops b_ω in jeder der drei Raumrichtungen geschätzt. Pro Raumrichtung und Sensor werden ein additiver Fehleranteil (*engl.* bias) und ein proportionaler Fehleranteil (*engl.* scale factor error, SFE) berücksichtigt, sodass $b_a, b_\omega \in \mathbb{R}^6$ gilt. Abschließend gehen auch die Abweichungen zwischen modelliertem und tatsächlichem Gravitationsfeld $b_g \in \mathbb{R}^3$ und atmosphärischen Luftdruck $b_p \in \mathbb{R}^1$ in den Fehlerzustand ein.

Der Fehlerzustand wird auf Basis der aktuell verfügbaren Messdaten der Sensoren durch den UKF aktualisiert. Dies findet aufgrund der Unterschiedlichen Messfrequenzen der verschiedenen Sensoren mit einer deutlich geringeren Häufigkeit statt als die Aktualisierung des geschätzten Zustands. Der wahre Zustand

$$x := \begin{bmatrix} \hat{r} + \delta r \\ \hat{v} + \delta v \\ \hat{q} \otimes \delta q \end{bmatrix} \tag{20}$$

ergibt sich aus dem geschätzten Zustand $\hat{x}$ und den entsprechenden Komponenten des Fehlerzustands δx mithilfe der Umformung des Lagefehlers nach Gleichung 19. Die Kovarianz $P \in \mathbb{R}^{25\times25}$ des Fehlerzustands wird ebenfalls durch den UKF ermittelt und beschreibt die Unsicherheit der aktuellen Lösung des Navigationsfilters.

Die Berechnungen des Navigationsfilters können nochmals in einzelne Module unterteilt werden (Abb. 6). Auch hier ist der Ablauf gesteuert vom Datenfluss. Es werden zwischen den Messungen von Beschleunigung a und Drehrate ω durch die IMU und denen der übrigen Sensoren h mit zugehörigen Unsicherheiten σ_h unterschieden.

Für die mit höherer Frequenz eingehenden Messungen von Beschleunigung und Drehrate wird zunächst eine Fehlerkorrektur vorgenommen, wozu die letzte Schätzung der systematischen Sensorfehler aus dem Fehlerzustand genutzt wird. Auf Basis der korrigierten Sensordaten findet die inertiale Navigation statt. Zur Propagation des aktuellen Zustands vom Zeitpunkt t über ein Intervall Δt wird das explizite

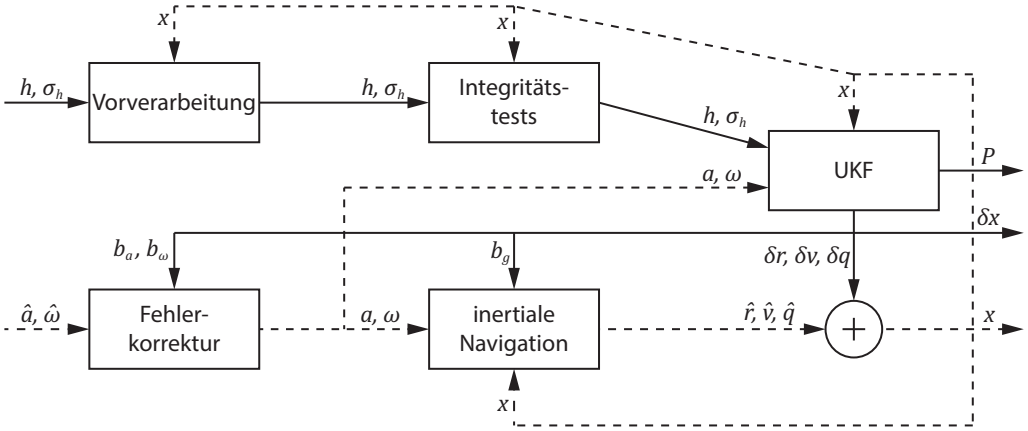

Abbildung 6: Modell des Datenflusses zwischen den Modulen (Rechtecke) des Navigationsfilters. Der Datenfluss ausgehend von Beschleunigungs- und Drehratenmessung der IMU mit höherer Frequenz ist durch gestrichelten Pfeilen dargestellt, während der Datenfluss auf Basis der übrigen Sensormessungen mit geringerer Frequenz durch durchgezogene Pfeilen dargestellt wird.

Euler-Verfahren entsprechend

$$x(x + \Delta t) = x(t) + \Delta t \cdot \dot{x}(t) + 2 \cdot \Delta t^2 \cdot \ddot{x}(t) \tag{21}$$

eingesetzt. Das Zeitdifferential 1. Ordnung wird nach

$$\dot{x}(t) = \begin{bmatrix} \dot{r} \\ \dot{v} \\ \dot{q} \end{bmatrix} = \begin{bmatrix} v \\ R \cdot a + g(r) + b_g - 2 \cdot \omega \times v \\ R \cdot \Omega - \Omega_{\oplus} \cdot R \end{bmatrix} \tag{22}$$

bestimmt. Hierbei ist R die 3×3-Rotationsmatrixdarstellung des Quaternions q, g der Gravitationsvektor an der aktuellen Position entsprechend des Gravitationsmodells mit dem aktuell geschätzten Fehler b_g und $2 \cdot \omega \times v$ die Coriolis-Beschleunigung. Die Repräsentation der Drehrate ω durch eine antisymmetrische Matrix gibt Ω wieder, wobei $\Omega_{\oplus} \cdot R$ aus der Drehrate der Erde resultiert.

Das Zeitdifferential 2. Ordnung wird lediglich für die Position berücksichtigt und so ergibt sich

$$\ddot{x}(t) = \begin{bmatrix} \ddot{r} \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} \dot{v} \\ 0 \\ 0 \end{bmatrix} \tag{23}$$

wobei das Zeitdifferential 1. Ordnung der Geschwindigkeit nach Gleichung 22 bestimmt wird.

Der im Rahmen der inertialen Navigation zeitliche propagierte geschätzte Zustand wird mit dem Fehlerzustand nach Gleichung 20 korrigiert, was in Abbildung 6

durch den Kreis mit dem Plus-Zeichen dargestellt ist. Somit steht nach dem Eingang neuer Messdaten der IMU ein aktualisierter Zustand zur Verfügung.

Beim Vorliegen von Messdaten der übrigen Sensoren werden zunächst ein Vorverarbeitungsschritt und ein Integritätstest durchlaufen. Ersteres kann etwa eine nötige Überführung der Daten in ein anderes Bezugssystem sein. Als Integritätstest wird die *Normalized Innovation Squared* Teststatistik, kurz NIS, eingesetzt. Diese ähnelt der χ^2-Teststatistik und wird genutzt, um die Abweichung zwischen tatsächlichem und bezüglich des aktuellen Zustands erwartetem Messwert zu bewerten. Damit wird eine konsistente Lösung des Navigationsfilters bis zu einem gewissen Grad ermöglicht, selbst wenn inkonsistente Messwerte vorliegen, beispielsweise durch Sensorfehler.[MP71][BSLK01]

Der UKF als zentrales Modul der Sensorfusion wird nicht auf den Zustand sondern auf den Fehlerzustand angewandt und mit einem Zeitversatz von 0.25 s auf den Daten aus der Vergangenheit ausgeführt. Durch diesen Zeitversatz wird sichergestellt, dass alle Sensordaten, die bis zum Zeitpunkt der Filterausführung aufgenommen wurden die Vorverarbeitungsschritte bereits durchlaufen haben und auch wirklich zur Verfügung stehen. Damit werden keine Sensordaten verworfen und die Resultate des Filters müssen auch nicht revidiert werden. Zur Umsetzung dieser Strategie ist es jedoch erforderlich die in der inertialen Navigation verwendeten Daten zwischenzuspeichern. Sobald ein neuer Fehlerzustand durch den UKF berechnet wurde, werden die Berechnungen der intertialen Navigation vom Zeitpunkt des berechneten Fehlerzustands aus bis zur aktuellsten Messung mit diesem neuen Fehlerzustand wiederholt, sodass stets der neuste Fehlerzustand berücksichtigt wird.

Für die Ausgabe bedeutet dies, dass es neben dem in der Vergangenheit geschätzten Fehlerzustand und dessen Kovarianz sowohl einen aktuell geschätzten Zustand gibt, der für die Steuerung des Hubschraubers zum Einsatz kommt und einen verzögert geschätzten Zustand, der durch den UKF ideal korrigiert wurde, jedoch erst mit einer Zeitverzögerung zur Verfügung steht. Deshalb kann dieser Zustand nicht zur Steuerung des Hubschraubers genutzt werden, ist jedoch für die Bewertung der Performance des UKF maßgeblich, da er die Schätzung des UKF repräsentiert, die dieser geliefert hätte, wären alle Sensordaten sofort zur Verarbeitung verfügbar gewesen. Fehler durch Verzögerungen werden nicht durch den UKF verursacht und sollten deshalb auch nicht in der Bewertung von dessen Performance eingehen.

Eingehende Sensordaten werden zusätzlich zur allgemeinen Zeitverzögerung Zeitslots von 0.1 s Breite zugeordnet und alle Daten aus einem Zeitslot gesammelt durch den UKF verarbeitet. Durch diese Unterteilung wird der Filter nicht mit jedem Datum eines einzelnen Sensors ausgeführt, sondern es werden mehrere Daten gleich-

zeitig in einem Filterdurchlauf verarbeitet, was zu einer effizienteren Nutzung der Hardware-Ressourcen führt.

Im Folgenden wird nun die konkrete Umsetzung des unter 2.1.2 allgemein beschriebenen Filters für die Navigationssoftware unter Verwendung der gleichen Indices der jeweiligen Zeitpunkte des aktuellen Zustands $k-1$ und des Folgezustands k dargestellt. Zunächst werden der aktuelle Fehlerzustand

$$\delta x_{k-1|k-1} = \begin{bmatrix} \delta x_{k-1|k-1} \\ E(w_k) \\ E(v_k) \end{bmatrix} \tag{24}$$

sowie die zugehörige Kovarianz

$$P_{k-1|k-1} = \begin{bmatrix} P_{k-1|k-1} & 0 & 0 \\ 0 & Q_k & 0 \\ 0 & 0 & R_k \end{bmatrix} \tag{25}$$

um die Mittelwerte und Kovarianzen des Prozess- ($E(w_k)$ und Q_k) und des Messrauschens ($E(v_k)$ und R_k) zum nächsten Zeitpunkt k erweitert. Die Mittelwerte der jeweiligen Rauschanteile sind dabei 0 (s. 2.1.2).

Die Kovarianz des Prozessrauschens wird durch eine Diagonalmatrix der Varianzen der einzelnen Komponenten des Fehlerzustands beschrieben. Für die Varianzen der Komponenten des Flugzustands können die Spezifikationen von Beschleunigungssensor und Gyroskop der IMU herangezogen werden, da diese Sensordaten in die inertialen Navigation eingehen. Aus diesen Spezifikationen ergeben sich auch direkt die Varianzen von Bias und SFE für beide Sensoren. Da nicht alle benötigten Varianzen für die eingesetzt iTraceRT-F400-Q-E IMU bekannt sind, werden diese den Spezifikationen der vergleichbaren 3DM-GX3-25 IMU der Firma LORD Sensing Systems entnommen.[Sys14] Für die Varianz des Gravitationsmodellfehlers sowie des Luftdruckmodellfehlers werden typische Werte gewählt. Tabelle 3 fasst die Standardabweichungen aller Werte zusammen, deren Quadrate die Varianz des Prozessrauschens ergeben. Die Varianzen einer Größe des Fehlerzustands (Gl. 18) sind dabei jeweils in alle Raumrichtungen identisch.

Der erste Block des Messrauschens R_k wird ebenfalls durch eine Diagonalmatrix der Varianzen beschrieben, die die statistischen Messfehler der jeweiligen Sensoren beschreibt. Es werden für Position und Geschwindigkeit, die per GPS bestimmt werden, typische Fehler dieser Messung angenommen. Gleiches gilt für Magnetometer und Barometer. Die jeweiligen Standardabweichungen aus denen sich die Varianzen ergeben, zeigt Tabelle 4.

Position / m		$7.5 \cdot 10^{-4}$
Geschwindigkeit / m/s		$7.5 \cdot 10^{-4}$
Lage / °		$3.67 \cdot 10^{-2}$
Beschleunigungs-	Bias / m/s	$1.31 \cdot 10^{-4}$
sensor	SFE	$1.67 \cdot 10^{-4}$
Gyroskop	Bias °/s	$1.67 \cdot 10^{-3}$
	SFE	$1.67 \cdot 10^{-4}$
Gravitationsmodell / m/s^2		0.01
Luftdruck / hPa		1

Tabelle 3: Standardabweichung zur Bestimmung der Kovarianzmatrix des Prozessrauschens Q_k. Die Daten resultieren aus den Spezifikationen der LORD Sensing Systems IMU 3DM-GX3-25[Sys14] sowie typischen Werten.

Für die Messung der Landmarken durch Kamera und Lidar ergibt sich die Kovarianz des Messrauschens jedoch nicht direkt aus den Sensorspezifikationen, da es zwischen den gemessenen Positionen der Landmarken eine signifikante Abhängigkeit gibt. Diese ist darauf zurückzuführen, dass sich die Unsicherheit der Positionierung der PTU nicht zufällig sondern systematisch auf die Positionsbestimmung aller Landmarken gleichzeitig auswirkt. Um dies zu berücksichtigen wird die Kovarianz durch eine Unscented Transformation ermittelt, auf der auch der UKF basiert. Es wird ein Messvektor sowie ein Diagonalmatrix der Varianzen dieser Messung erstellt. Dazu wird zunächst der geschätzte Positionsfehler der PTU mit 0 in den Messvektor sowie die zugehörige Varianz (Tab. 4) in die Diagonalmatrix eingetragen. Pro Landmarke wird die horizontale und vertikale Position der Landmarke auf dem Kamerabild in Pixeln in den Messvektor eingetragen sowie die für alle Landmarken identische Unsicherheit dieser Werte (Tab. 4) in die Diagonalmatrix der Varianzen. Ist für eine Landmarke eine Abstandsinformation vorhanden, wird diese ebenfalls individuell pro Landmarke zusammen mit einer allgemeingültigen Varianz (Tab. 4) berücksichtigt. Mit L als Dimensionalität des so erstellten Messvektors werden mit entsprechenden Gewichten (Gl.32) Sigma-Punkte nach Gleichung 8 erstellt. Mit Hilfe des Kameramodells der Pinhole-Kamera werden die in den Sigma-Punkten repräsentierten Pixel-Positionen über das SFF in Winkel in Kugelkoordinaten übertragen (Gl. 52). Etwaige Abstandsinformationen bleiben erhalten. Die gewichtete Mittelwertbildung der Sigma-Punkte (Gl. 9) liefert die erwartete Messung und nach Gleichung 10 wird auch die Kovarianz erhalten, die die Abhängigkeiten zwischen den Messwer-

ten berücksichtigt. Mit dieser Kovarianz als zweitem Block ergibt sich die gesamte Kovarianzmatrix R_k.

	1. Achse	2. Achse	3. Achse
Position (GPS) / m	0.5	0.5	1.5
Geschwindigkeit (GPS) / m/s	0.5	0.5	1.5
Magnetfeld / nT	100		–
Luftdruck / hPa	0.001	–	–
PTU Positionsfehler / °	0.01		–
Landmarken Positionsfehler / px	2		–
Landmarken Abstandsfehler / m	0	–	–

Tabelle 4: Standardabweichung zur Bestimmung der Kovarianzmatrix des Messrauschens R_k. Die Achsen beziehen sich für Position und Geschwindigkeit auf ein North-East-Down-Koordinatensystem. Für das Magnetometer auf die Achsen des SFF. Für den Luftdruck bezieht sich die Standardabweichung auf den Messfehler. Der Positionsfehler der PTU sind für Pan- und Tilt-Achse identisch, ebenso wie die der Landmarken für die vertikale und horizontale Position im Kamerabild. Der Abstandsfehler bezieht sich auf die Messunsicherheit des Lidar-Sensors, der als fehlerfrei angenommen wird.

Ausgehend vom so erweiterten Zustand und dessen Kovarianz (Gl. 24 und 25) werden $N = 2L+1$ Sigma-Punkte bestimmt, wobei sich mit dem aktuell vorliegenden Messvektor z_k die Anzahl $L = 2 \cdot \dim(x) + \dim(z_k)$ ergibt. Mit $\left[\sqrt{(L+\lambda)P_{k-1|k-1}}\right]_j$ als j-te Spalte der Quadratwurzel der Matrix $P_{k-1|k-1}$ ergeben sich nach

$$s^j_{k-1|k-1} = \begin{cases} \delta x_{k-1|k-1} & \text{für } j = 0 \\ \delta x_{k-1|k-1} + \left[\sqrt{(L+\lambda)P_{k-1|k-1}}\right]_j & \text{für } j \in [1, L] \\ \delta x_{k-1|k-1} - \left[\sqrt{(L+\lambda)P_{k-1|k-1}}\right]_j & \text{für } j \in [L+1, 2L] \end{cases} \tag{26}$$

die jeweiligen Sigma-Punkte $s^j_{k-1|k-1}$.

Als nächstes wird die Übergangsfunktion f auf die Sigma-Punkte angewandt. Hierzu wird die in Gleichung 24 beschriebene Dreiteilung genutzt. Die Übergangsfunktion wird nur auf den Teil der Sigma-Punkte angewandt, die den Fehlerzustand repräsentieren. Danach wird das im zweiten Teil der Sigma-Punkte enthalte Prozessrauschen zum Ergebnis addiert. Das Messrauschen bleibt unverändert und wird

an späterer Stelle angewandt, wie

$$s^j_{k|k-1} = f(s^j_{k-1|k-1}) = f\left(\begin{bmatrix}\delta x \\ w_k \\ v_k\end{bmatrix}\right) = \begin{bmatrix}f'(\delta x) + w_k \\ v_k\end{bmatrix} \tag{27}$$

zeigt. Für den Übergang des Fehlerzustands wird analog zum Euler-Ansatz (Gl. 22) für den Zustand vorgegangen. Demnach ergibt sich

$$f'(\delta x) = f'\left(\begin{bmatrix}\delta r \\ \delta v \\ \delta\theta \\ b_a \\ b_\omega \\ b_g \\ b_p\end{bmatrix}\right) = \begin{bmatrix}\delta r \\ \delta v \\ \delta\theta \\ b_a \\ b_\omega \\ b_g \\ b_p\end{bmatrix} + \Delta t \cdot \begin{bmatrix}\delta\dot{r} \\ \delta\dot{b} \\ \delta\dot{\theta} \\ 0 \\ 0 \\ 0 \\ 0\end{bmatrix} \tag{28}$$

mit

$$\begin{aligned}
\delta\dot{r} &= \delta v_m \\
\delta\dot{v} &= (\delta R - I) \cdot R \cdot a + \delta R \cdot R \cdot b_a + g(r + \delta r) - g(r) + b_g - 2 \cdot \omega \times v \\
\delta\dot{\theta} &= \delta R \cdot \left(R \cdot \Omega \cdot R^\top + \Omega_\oplus\right) - \omega_\oplus \cdot \delta R
\end{aligned} \tag{29}$$

als Übergangsfunktion.

Die Rekombination der Sigma-Punkte im Vorhersage-Schritt kann mit

$$\delta x_{k|k-1} = \sum_{j=0}^{2L} W^j_a s^j_{k|k-1} \tag{30}$$

und

$$P_{k|k-1} = \sum_{j=0}^{2L} W^j_c \left[s^j_{k|k-1} - \delta x_{k|k-1}\right]\left[s^j_{k|k-1} - \delta x_{k|k-1}\right]^\top \tag{31}$$

durchgeführt werden. Hierzu werden die Gewichte

$$\begin{aligned}
W^0_a &= \frac{1}{3} \\
W^0_c &= W^0_s \\
W^j_a &= \frac{1 - W^0_a}{2L} \\
W^j_c &= W^j_s \\
\lambda &= \frac{L}{1 - W^0_a}
\end{aligned} \tag{32}$$

nach Julier und Uhlmann angewandt.[JU04] Aus den Gewichten ergibt sich auch λ in Gleichung 26.

Zur Bestimmung der Gamma-Punkte wird nun die Beobachtungsfunktion h auf die bereits verarbeiteten Sigma-Punkte angewandt. Dazu wird die eigentliche Beobachtungsfunktion erneut nur auf den Teil der Sigma-Punkte angewandt, die den Fehlerzustand repräsentieren. Danach wird der bisher unverändert übertragene Teil des Messrauschens addiert, wie

$$g_k = h(s_{k|k-1}) = h\left(\begin{bmatrix}\delta x \\ v_k\end{bmatrix}\right) = h'(\delta x) + v_k \tag{33}$$

darstellt.

Für die durch den GPS-Sensor gemessene Position und Geschwindigkeit wird bereits in den Messvektor z_k nicht der tatsächliche Messwert sondern die Differenz aus Messwert und letzter Schätzung im Zustand eingetragen. Somit enthält der Messvektor einen Positions- und Geschwindigkeitsfehler. Damit können der Positions- und der Geschwindigkeitsfehler unverändert aus den Sigma-Punkten in die Gamma-Punkte übertragen werden.

Für das Magnetometer werden die, mit dem durch den jeweiligen Sigma-Punkt geschätzten Fehler korrigierte, Position $r+\delta r$ und Lage $\theta+\delta\theta$ mithilfe des World Magnetic Models in eine Magnetfeld-Messung übertragen. Position und Lage stammen dabei aus dem aktuellen Zustand x_{k-1}.

Analog wird die erwartete Barometermessung aus der mit dem Fehler des jeweiligen Sigma-Punktes korrigierten Position und dem durch diesen Sigma-Punkt geschätzten Barometerfehler anhand der Standardatmosphäre bestimmt und dem korrespondierenden Gamma-Punkt hinzugefügt. Für die Landmarken wird jede der aktuell sichtbaren Landmarken analog behandelt. Auf Basis der mit dem durch den jeweiligen Sigma-Punkt geschätzten Fehler korrigierten Lage und Position sowie der aktuellen Ausrichtung der PTU wird die Position der Landmarke in das SFF übertragen. Mit

$$\begin{aligned}\alpha &= \operatorname{arctan2}(x, z) \\ \beta &= \operatorname{arctan2}(y, z)\end{aligned} \tag{34}$$

ergeben sich der horizontale Winkel α und der vertikale Winkel β bezüglich des Zentrums des aktuellen FOV der Kamera aus der Position $\begin{bmatrix}x & y & z\end{bmatrix}^\top$ der Landmarke im SFF. Vor dem Hinzufügen der Winkel zum Gamma-Punkt werden diese mit dem Öffnungswinkel des FOV abgeglichen und nur berücksichtigt, falls sie innerhalb dieses FOV liegen. Im Falle einer verfügbaren 3D-Information wird der Abstand

zu $\sqrt{x^2+y^2+z^2}$ bestimmt und ebenfalls dem Gamma-Punkt zum betrachteten Sigma-Punkt hinzugefügt.

Aus den bestimmten Gamma-Punkten wird die erwartete Messung

$$\hat{z}_k = \sum_{j=0}^{2L} W_a^j g_k^j \tag{35}$$

sowie die zugehörige Kovarianz-Matrix

$$P_{z_k z_k} = \sum_{j=0}^{2L} W_c^j \left[g_k^j - \hat{z}_k\right] \left[g_k^j - \hat{z}_k\right]^\top \tag{36}$$

berechnet. Ebenso wird die Kreuzkovarianz-Matrix für die weitere Berechnung benötigt, die sich aus den verarbeiteten Sigma-Punkten sowie den Gamma-Punkten nach

$$P_{x_k z_k} = \sum_{j=0}^{2L} W_c^j \left[s_{k|k-1}^j - \delta x_{k|k-1}\right] \left[g_k^j - \hat{z}_k\right]^\top \tag{37}$$

ergibt. Damit kann die Kalman-Gain-Matrix zu

$$K_k = P_{x_k z_k} P_{z_k z_k}^{-1} \tag{38}$$

bestimmt werden. Mit diesen Komponenten ist schließlich die Propagation des Fehlerzustands und der Kovarianz um einen Zeitschritt nach

$$\begin{aligned} \delta x_{k|k} &= \delta x_{k|k-1} + K_k \left(z_k - \hat{z}_k\right) \\ P_{k|k} &= P_{k|k-1} - K_k P_{z_k z_k} K_k^\top \end{aligned} \tag{39}$$

möglich. Am Ende eines solchen Update-Schritts wird der Fehler von Position, Geschwindigkeit und Lage zum aktuellen Zustand hinzuaddiert und im Fehlerzustand zurückgesetzt, sodass zu jeder Zeit ein möglichst genauer Zustand vorliegt.

Damit ist der durch den Datenfluss gesteuerte Ablauf aller Schritte des Navigationsfilters beschrieben. Für den Betrieb ist jedoch noch die Initialisierung von Zustand, Fehlerzustand und Kovarianz nötigt. Die Initialisierung des Zustands erfolgt mit der vor dem Start des Navigationsfilters in der Regel ungefähr bekanntem Flugzustand. Dieser kann aus den Messwerten einzelner Sensor abgeleitet werden. Der Fehlerzustand wird mit 0 initialisiert, da der Fehler zunächst nicht bekannt ist und vom Filter erst geschätzt werden muss. Eine Ausnahme stellt der Barometerfehler b_p dar. Dieser ist häufig groß, da wetterabhängige Abweichungen von der Standardatmosphäre die Regel sind. Deshalb erfolgt die Initialisierung auf Basis der Höhe im initialen Zustand als Differenz zwischen tatsächlich gemessenem Luftdruck bei der Filterinitialisierung und dem durch die Standardatmosphäre bei dieser Höhe

zu erwartenden Luftdruck. Die Kovarianz wird zu Beginn als Diagonalmatrix der Varianzen der jeweiligen Werte festgelegt. Die Varianzen von Position, Geschwindigkeit und Lage werden etwas größer gewählt, als Werte die typischerweise durch den Filter ermittelt werden. Dadurch beginnt der Filter mit einer Konvergenzphase, bei der etwaige Ungenauigkeiten bei der Initialisierung des Zustands ausgeglichen werden können. Tabelle 5 fasst die Standardabweichungen σ zusammen, aus deren Quadrat sich die Varianzen für die Filterinitialisierung ergeben. Für Bias und SFE von Beschleunigungssensor und Gyroskop werden aus Gründen der Konstanz auch hier die Spezifikationen der IMU vom Typ 3DM-GX3-25 der Firma LORD Sensing Systems genutzt. Für den Fehler des Gravitationsmodels und des Luftdrucks werden sensortypische Werte angenommen.

		1. Achse	2. Achse	3. Achse
Position / m		5	5	10
Geschwindigkeit / m/s		1	1	1.5
Lage / °		5	5	10
Beschleunigungs-sensor	Bias / m/s	$6.54 \cdot 10^{-3}$		
	SFE	$3.33 \cdot 10^{-4}$		
Gyroskop	Bias °/s	$8.33 \cdot 10^{-2}$		
	SFE	$1 \cdot 10^{-4}$		
Gravitationsmodell / m/s^2		0.1		
Luftdruck / hPa		50	–	–

Tabelle 5: Standardabweichung zur Bestimmung der initialen Kovarianzmatrix des UKF. Die Achsen beziehen sich für Position, Geschwindigkeit und Lage auf ein North-East-Down-Koordinatensystem, für Beschleunigungssensor und Gyroskop auf die drei Achsen des jeweiligen SFF und für das Gravitationsmodell auf des geozentrische Koordinatensystem. Für das Barometer bezieht sich der Wert nicht auf eine Raumachse sondern auf den Modellfehler.

2.3.3 Test- und Simulationsumgebung

Um Flüge des superArtis zu simulieren und dabei den Einfluss von Änderungen an den Modulen der Steuer- und Navigationssoftware zu testen, wird eine Integration der Software in eine Matlab/Simulink-Umgebung genutzt. Diese Umgebung bietet die Möglichkeit in C/C++ programmierte Module, wie die der Navigationssoftware, als so genannte *s-functions* einzubinden. Damit kann sowohl zur Flugsteuerung des superArtis als auch für die Simulation die selbe Code-Basis eingesetzt werden.

Eine Simulink-Simulation basiert auf einem gerichteten Graphen, der den Datenfluss zwischen den verschiedenen Modulen beschreibt und die Eingaben in Ausgaben transformiert. Dies entspricht somit auch der Software-Architektur, die auf dem superArtis zum Einsatz kommt (s. 2.3.2). Die Simulation wird dabei in festen Zeitschritten ausgeführt, wobei keine Echtzeitanforderungen erfüllt werden müssen, da der nächste Zeitschritt erst nach Abschluss der Berechnungen des vorhergehenden ausgeführt wird. Abbildung 7 zeigt das Simulink-Schaltbild, welches der Test- und Simulationsumgebung zugrunde liegt.

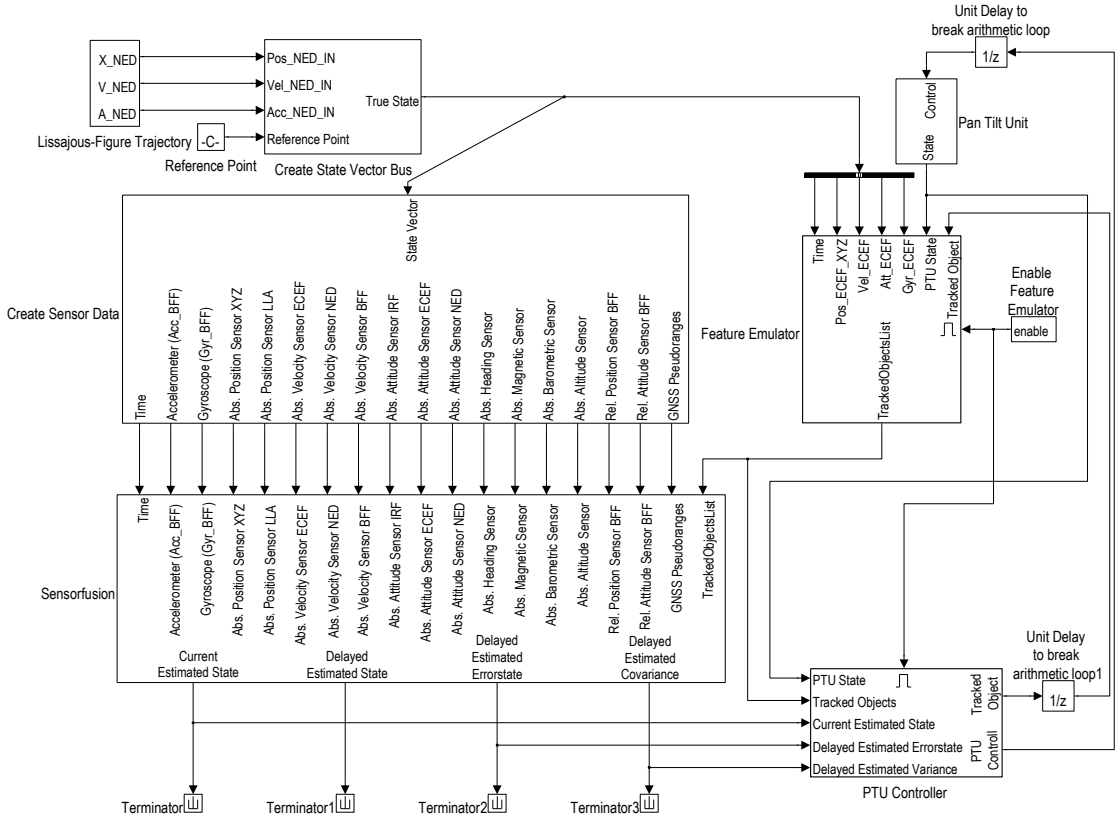

Abbildung 7: Simulink-Schaltbild der Test- und Simulationsumgebung.

Zur Erzeugung der Sensordaten wird der wahre Zustand des UAVs für jeden Zeitschritt der Simulation benötigt. Dazu wird eine Trajektorie genutzt sowie die Zeit der Simulation und damit die Zeit die zum Zurücklegen der Trajektorie benötigt wird. Als Trajektorie dient eine Lissajous-Figur bei der die Position $r(t)$ im NED-Kooridnatensystem als Funktion der Zeit nach

$$r(t) = \begin{bmatrix} A_N \cdot \sin\left(2\pi \cdot \omega_N \cdot t + \phi_N\right) + b_N \\ A_E \cdot \sin\left(2\pi \cdot \omega_E \cdot t + \phi_E\right) + b_E \\ A_D \cdot \sin\left(2\pi \cdot \omega_D \cdot t + \phi_D\right) + b_D \end{bmatrix} \tag{40}$$

genutzt wird, da so vielfältige Variationen anhand der weniger Parameter Amplitude A_i, Frequenz $omega_i$, Phase ϕ_i und Bias b_i pro Raumrichtung möglich sind und die Trajektorie differenzierbar ist. Mit den Parametern in Tabelle 6 ergibt sich die im Rahmen dieser Arbeit genutzt Flugbahn (Abb. 8). Diese enthält sowohl längere Bereiche ruhigen Flugs als auch abrupte Flugmanöver und ist geschlossen, sodass Start- und Endpunkt des Fluges einander entsprechen. Zusammen mit einem Startpunkt kann aus der Trajektorie somit zu jedem Zeitpunkt der Simulation eine Position r, Geschwindigkeit v und Beschleunigung a im NED-Kooridnatensystem als wahrer Zustand generiert werden.

Raumrichtung	Amplitude A_i/m	Frequenz ω_i/rad/s	Phase ϕ_i/rad	Bias b_i/m
North	100	2	0	0
East	1000	2	$\pi/2$	0
Down	-100	1	$3\pi/2$	-150

Tabelle 6: Parameter zur Berechnung der Trajaktorie in Abbildung 8 nach Gleichung 40.

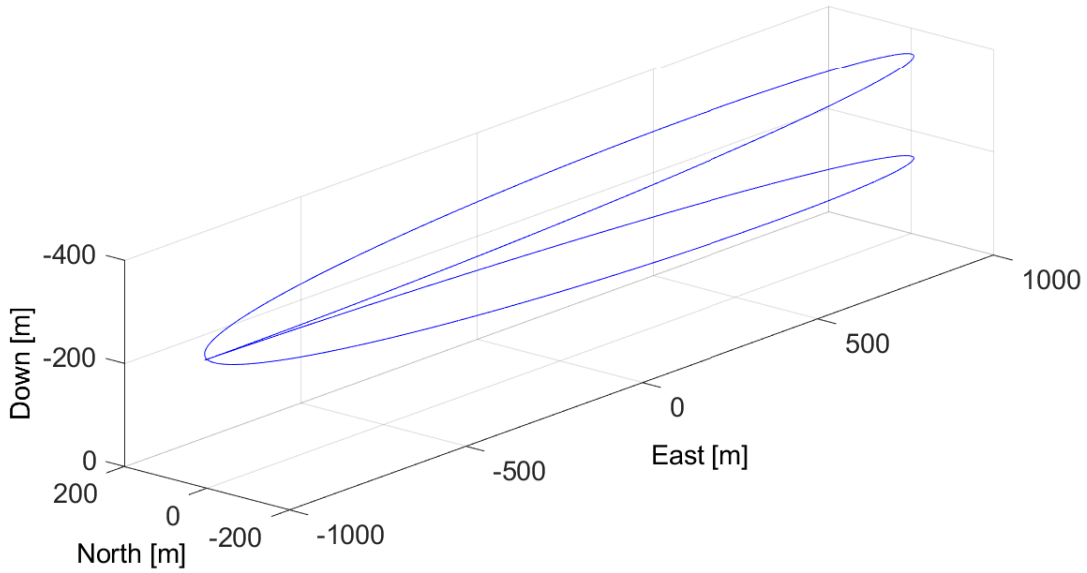

Abbildung 8: Im Rahmen der Simulation als Trajektorie eingesetzt Lissajous-Figur nach Gleichung 40 mit den Parameterin in Tabelle 6.

Zur Beschreibung der Simulation einzelner Sensordaten werden an mehreren Stellen Transformationen zwischen Koordinatensystemen eingesetzt. Koordinatentransformationen einer Größe y von einem Ausgangskoordinatensystem A in ein Zielkoordinatensystem Z werden dabei durch $T_{A\to Z}(y)$ dargestellt. Auch werden

Zufallszahlen aus einer Normalverteilung NRN mit einer Varianz σ^2 zur Simulation von Rauschprozessen benötigt. Eine solche Zufallszahl wird damit als $NRN(\sigma^2)$ dargestellt.

Um einen Messwert des Beschleunigungssensors zu simulieren, wird der anhand der Trajektorie bestimmte wahre Zustand vom NED in das BFF übertragen. Als Zwischenschritt der Übertragung wird das geozentrische Koordinatensystem (*engl.* earth centered earth fixed, ECEF) gewählt, wobei die Coriolis-Beschleunigung aus dem Drehraten-Vektor der Erde $\omega_\oplus$, die Graviationsbeschleunigung aus dem Gravitationsmodell $g(a)$, die Zentrifugal-Beschleunigung der Erde ebenfalls auf Basis von $\omega_\oplus$ und der wahre Fehler des Gravitationsmodells b_g berücksichtigt werden. Damit wird die Beschleunigung im ECEF

$$\begin{aligned} a_{ECEF} = T_{NED\to ECEF}(a) - 2 \cdot \omega_\oplus \times T_{NED\to ECEF}(v) - g(T_{NED\to ECEF}(a)) \\ - \omega_\oplus \times (\omega_\oplus \times T_{NED\to ECEF}(r)) + b_g \end{aligned} \quad (41)$$

als Zwischenwert erhalten. Unter Berücksichtigung des SFE des Sensors SFE_a, des Bias B_a und einem Rauschen entsprechender Varianz σ_a^2 ergibt sich daraus nach

$$\hat{a} = T_{ECEF\to BFF}(a_{ECEF}) \cdot SFE_a + B_a + NRN(\sigma_a^2) \quad (42)$$

der simulierte Messwert des Beschleunigungssensors $\hat{a}$.

Für die Drehrate werden die ins ECEF transformierte Beschleunigung (Gl. 41) sowie die ebenfalls ins ECEF transformierte Geschwindigkeit $v_{ECEF} = T_{NED\to ECEF}(v)$ benötigt. Aus beiden kann die Lage mithilfe von

$$\begin{aligned} x_\omega &= \frac{v_{ECEF}}{\|v_{ECEF}\|} \\ y_\omega &= \frac{-\frac{a_{ECEF}}{\|a_{ECEF}\|} \times x_\omega}{\left\|-\frac{a_{ECEF}}{\|a_{ECEF}\|} \times x_\omega\right\|} \\ z_\omega &= \frac{x_\omega \times y_\omega}{\|x_\omega \times y_\omega\|} \end{aligned} \quad (43)$$

als Rotationsmatrix

$$\Omega_{ECEF} = \begin{bmatrix} x_\omega & y_\omega & z_\omega \end{bmatrix} \quad (44)$$

bestimmt werden. Durch Differenzbildung zur Lage aus dem vorherigen Zeitschritt der Simulation ergibt sich die Lageänderung $\Delta\Omega_{ECEF}$. Der wahre Drehraten-Vektor ergibt sich aus

$$\Omega_t = \Delta\Omega_{ECEF} \times (\Omega_{ECEF})^\top \quad (45)$$

anhand der Komponenten $\omega_{i,j}$

$$\omega_t = \begin{bmatrix} \omega_{2,3} & \omega_{3,1} & \omega_{1,2} \end{bmatrix} \quad (46)$$

der Matrix Ω_t. Unter Berücksichtigung der Erdrotation wird ω_t in das SFF überführt. Unter analoger Berücksichtigung von SFE, Bias und Rauschen wie für den Beschleunigungssensor wird schließlich der Messwert $\hat{\omega}$ des Gyroskops nach

$$\hat{\omega} = T_{ECEF \to BFF}\left(\omega_t\right) \cdot SFE_\omega + B_\omega + NRN(\sigma_\omega^2) \tag{47}$$

erhalten.

Zur Simulation der Positionsmessung durch einen GPS-Sensor in LLA-Koordinaten ist lediglich eine Transformation zwischen den Koordinatensystemen sowie die Berücksichtigung des Rauschens pro Koordinate durch

$$\hat{r}_{GPS} = T_{NED \to ECEF}(r) + NRN(\sigma_r^2) \tag{48}$$

nötig. Ganz ohne Transformation ergibt sich die Beschleunigungsmessung im NED-Koordinatensystem, bei der nur das Messrauschen nach

$$\hat{a}_{GPS} = a + NRN(\sigma_a^2) \tag{49}$$

addiert werden muss.

Der wahre Wert einer Magnetfeldmessung wird aus der Position im ECEF und dem World Magnetic Model $WMM(r)$ bestimmt. Diese wird in das BFF übertragen und mit einem Rauschen entsprechend der Varianz versetzt, sodass

$$\hat{B} = T_{ECEF \to BFF}\left(WMM(r_{ECEF})\right) + NRN(\sigma_B^2) \tag{50}$$

den simulierten Messwert repräsentiert.

Aus der Position im ECEF ergibt sich die Höhe in der Atmosphäre, sodass der Luftdruck auf Basis der Standardatmosphäre $SA(r)$ bestimmt werden kann. Zusammen mit dem konstanten Modellfehler b_p und einem Rauschen ergibt sich nach

$$\hat{p} = SA(r_{ECEF}) + b_p + NRN(\sigma_p^2) \tag{51}$$

der Messwert des Barometers.

Zur Simulation der Messung von Landmarken durch Kamera und Lidar wird zunächst ein Katalog der Landmarken benötigt. Dieser wird vor Beginn der Simulation generiert, wobei zunächst ein diskretisiertes Terrain in NED-Koordinaten erzeugt wird. Dieses ist in 0.01 m Einheiten unterteilt und wird in Nord und Ost durch die maximale Ausdehnung der Trajektorie begrenzt, wobei diese Grenzen um $\tan(80°) \cdot h_{\max}$ erweitert werden, wobei $h_{\max}$ der maximalen Flughöhe über dem Boden entspricht. Damit wird der durch die Kamera sichtbare Bereich bei einer

Blickrichtung weg von der Trajektorie berücksichtigt. In der Höhe reicht die Begrenzung vom Boden bis zu einer zur mittleren Flughöhe der Trajektorie relativen Höhe h_{rel}, die einen Eingabeparameter der Simulation darstellt. Für $h_{\mathrm{rel}} = 0$ ist das Terrain absolut eben, während für $h_{\mathrm{rel}} = 1$ das Terrain bis zur mittleren Flughöhe reicht und für $h_{\mathrm{rel}} > 1$ auch Landmarken oberhalb der mittleren Flughöhe simuliert werden, was zum Beispiel bei einem Flug durch ein Gebäude zu erwarten ist.

Auf Basis dieser maximalen Höhe wird im diskretisierten Terrain die Höhe der einzelnen Punkte durch ein Perlin-Rauschen generiert, sodass sich graduelle Änderungen der Höhe zwischen benachbarten Punkten ergeben.[Per85] Die Position der Landmarken in Nord und Ost wird nun durch Zufallszahlen aus einer Gleichverteilung über die Ausdehnung des Terrains in der jeweiligen Richtung bestimmt. Die Höhe ergibt sich dann aus der Höhe des Terrains an dieser Stelle. Abschließend wird die Größe der Landmarken anhand von χ^2-verteilten Zufallszahlen festgelegt. Für die Verteilung wird ein Verteilungsparameter von 14 gewählt, wodurch die meisten Landmarken eine maximale Größe von 30 m aufweisen.

Zur Simulation der Landmarkenbeobachtung wird noch das Kamera-Modell sowie der Zustand der PTU benötigt. Als Kamera-Modell wird das einer Lochkamera eingesetzt. Anhand des der Brennweite der Linse ($f = 4\,\mathrm{mm}$), der Pixelgröße ($6.45\,\mu\mathrm{m}$) und Auflösung der Kamera ($800 \times 600\,\mathrm{px}$) kann aus der Größe einer Landmarke bestimmt werden, in welchem Abstandsbereich diese sichtbar ist. Dabei wird festgelegt, dass die Abbildung auf mindestens 10 Pixel der Kamera nötig ist, um die Landmarke zu identifizieren und dass sie vollständig ins Bild der Kamera passen muss. Für die PTU wird stets deren fehlerfreie Position im Verlauf der Simulation verfolgt. Für die Simulation der Messung wird die Position der PTU dann jedoch jeweils mit einem statistischen Fehler mit entsprechender Varianz versehen.

Nun wird über den zu Beginn der Simulation generierten Landmarken-Katalog iteriert und dabei die Position der Landmarke anhand der Position der PTU beßüglich des UAVs und der Position und Lage des UAVs im Raum in das SFF überführt. Durch den Vergleich des Abstands zwischen Sensor und Landmarke mit den Abstandsbereich der Sichtbarkeit wird festgestellt, ob die Landmarke potentiell sichtbar ist. In diesem Fall, wird die Landmarke aus ihrer Position im SFF $\begin{bmatrix} x & y & z \end{bmatrix}^\top$ nach

$$\begin{bmatrix} p_{\mathrm{h}} \\ p_{\mathrm{v}} \end{bmatrix} = \begin{bmatrix} \frac{f \cdot x}{z} \\ \frac{f \cdot y}{z} \end{bmatrix} + c \tag{52}$$

in die Position im Kamerabild in Pixeln überführt. Dabei ist f die Brennweite der Kamera, c das Zentrum des Sensors und p_{h} beziehungsweise p_{v} die horizontale und vertikale Position des Landmarke im Bild, jeweils in Pixeln. Für $z > 0$ liegt die

Landmarke vor und nicht hinter der Kamera und kann somit tatsächlich sichtbar sein. Liegen p_h und p_v im Bildbereich der Kamera, so ist die Landmarke tatsächlich im Bild sichtbar und wird berücksichtigt. Der Abstand des Objekts im SFF wird als fehlerfrei angesehen und stellt somit den Messwert dar. Liegt dieser innerhalb des Messbereichs des Sensors (s. 2.3.1), so wird zur Messung der Landmarke auch die Abstandsinformation zurückgegeben. Für jede messbare Landmarke wird so die Position im Kamerabild in Pixeln sowie die Position der Landmarke im ECEF, bekannt aus dem Katalog, und gegebenenfalls der Abstand als Messwert des Sensors ausgegeben.

Die so simulierten Daten der jeweiligen Sensoren werden mit der Frequenz des Sensors zur Verfügung gestellt, sodass der UKF wie unter 2.3.2.1 beschreiben auch in der Simulation arbeitet, wo er als s-function eingebunden wird.

Als letztes Modul der Simulation wird auch die Steuerung der PTU nachgebildet. Hierbei stehen Zustand, Fehlerzustand und Kovarianz als Eingabe zur Verfügung. Im Steuermodul ist zusätzlich auch die aktuelle Position sowie Geschwindigkeit mit der die PTU bewegt wird bekannt. Letztlich stellt die Liste der aktuell sichtbaren Landmarken, die wie zuvor erläutert erhalten wird, eine Eingabe dar. Mit dieser Information ist es möglich, die Kamera zur Nachverfolgung einer Landmarke mit der Kamera einzusetzen. Dazu wird ein PID-Regler zur Steuerung der Geschwindigkeit der PTU eingesetzt und die Position, Lage und Bewegung des Sensors relativ zur Landmarke berücksichtigt. Es kann eine maximale Nachverfolgungsdauer als Parameter festgelegt werden, ebenso wie eine Dauer bis zu der die Nachverfolgung einer aktuell nicht in den Sensordaten identifizierten Landmarke weiter fortgesetzt wird. Die Strategie zur Auswahl der Landmarke, die nachverfolgt werden soll, ist ebenfalls per Parameter steuerbar.

Bei einem Simluationslauf werden die bisher beschriebenen Daten wie wahrer Zustand direkt und verzögert geschätzter Zustand, Fehlerzustand mit Kovarianz aber auch Sensordaten und der Zustand der PTU zu jedem Zeitschritt der Simulation gespeichert. Zur Verbesserung der Datenbasis und um zu vermeiden, das System auf eine spezielle Anfangskonfiguration zu optimieren, ist es dabei möglich die Simulation mit unterschiedlich initialisiertem Zufallszahlengenerator bei sonst gleicher Konfiguration entsprechend einer Monte-Carlo-Simulation mehrfach auszuführen. Darüber hinaus können die diversen Parameter der Steuerung im Rahmen der Simulation variiert werden.

Zum Abschluss jedes Simulationslaufs werden noch etwaige doppelt gespeicherte Informationen der Zustände mit identischen Zeitstempeln entfernt sowie im Falle von mehreren Monte-Carlo-Simulationsläufen die Zeitstempel dieser Durchläufe an-

gepasst und etwaige fehlende Zwischenwerte interpoliert, sodass alle Durchläufe die gleiche Anzahl an Datenpunkten der gespeicherten Daten enthalten.

3 Strategien zur Landmarken-Auswahl und deren Implementierung

Die Änderung der Kameraausrichtung mithilfe der PTU (s. 2.3.1) hat das Ziel, die Schätzung des Flugzustands durch den Navigationsfilter (s. 2.3.2.1) gegenüber solchen auf Basis einer festen Kameraposition zu verbessern. Dazu wird das Feature Tracking (s. 2.2.2) eingesetzt und eine gewählte Landmarke über einen vorgegebenen Zeitraum oder bis zum Verlust ihrer Sichtbarkeit möglichst im Zentrum des FOV der Kamera gehalten. Auf Basis des Flugzustands und der bekannten Position der Landmarken in ECEF Koordinaten ist die Nachführung der Kamera mithilfe einer PID-Regelung möglich (s. 2.3.3). Beim Vorliegen neuer Messdaten der Kamera, wird stets geprüft, ob die aktuell nachzuverfolgende Landmarke noch im Blickfeld liegt und ob die maximale Zeit der Nachverfolgung bereits verstrichen ist und gegebenenfalls die Auswahl einer neuen Landmarke gestartet.

Die Auswahl aus dem vorliegenden Landmarkenkatalog ist dadurch beschränkt, dass nur Landmarken für die Nachverfolgung in Frage kommen, welche in einem Abstand zur aktuellen Position des UAVs liegen, sodass sie mit der Kamera beobachtbar sind. Die Sichtbarkeit wird ausgehend von der Größe der Landmarke d_L sowie ihrer Position r_L^{ECEF} und der des UAVs r_{UAV}^{ECEF}, jeweils in ECEF Koordinaten, überprüft. Das Kameramodell und die Hardwareparameter Brennweite f und Anzahl von Pixeln N_{px} entlang der kürzeren Sensorachse werden innerhalb der Prozedur (Alg. 1) als bekannt angenommen. Wie unter 2.3.3 beschrieben, wird auch hier angenommen, dass eine Landmarke dann auf der Kamera identifizierbar ist, wenn sie mindestens auf 10 Pixel abgebildet wird und gleichzeitig vollständig ins Bild passt.

Landmarken, für deren Beobachtung die PTU an den Rand ihres Stellbereichs bewegt werden muss, liegen dann nicht im Zentrum des FOV der Kamera und sind häufig nach kurzer Zeit nicht mehr sichtbar, da die Nachführung nicht weiter möglich ist. Deshalb wird für die Kandidaten von nachzuverfolgenden Landmarken im Allgemeinen gefordert, dass die PTU zu Beginn der Nachverfolgung maximal auf das 0.75-fache der maximalen Auslenkungen in jede Richtung bewegt werden muss. Damit hat die Regelung zum Nachführen der Kamera noch Bewegungsspielraum in jede Richtung. Gleichzeitig wird damit garantiert, dass die Landmarke überhaupt im Rahmen der Bewegung der PTU von der aktuellen Position und Lage des UAVs aus anvisierbar ist. Zusammen mit der Prüfung des Abstands (Alg. 1) wird so die Messbarkeit der Landmarke sichergestellt.

Algorithmus 1 Abstandsbereich der Sichtbarkeit von Landmarken

Eingabe: Landmarkengröße d_L, Landmarkenposition r_L^{ECEF}, UAV-Position r_{UAV}^{ECEF}

Ausgabe: Wahrheitswert ob die Landmarke im Sichtbarkeitsabstand liegt

$d_{max} := d_L \cdot (f/10)$
$d_{min} := d_L \cdot (f/N_{px})$
$d := \left\| r_L^{ECEF} - r_{UAV}^{ECEF} \right\|$
if $d_{min} < d < d_{max}$ **then**
    return true
else
    return false
end if

Zur Überprüfung dieser Bedingung zur Positionierung der PTU muss die Position der Landmarke im BFF (Gl. 54) in Kugelkoordinaten übertragen werden (Gl. 55). Der horizontale und vertikale Winkel entsprechen dann aufgrund der Anbringung der PTU am UAV (s. 2.3.1) direkt dem Stellwinkel der PTU, unter dem die Landmarke im Zentrum des FOV der Kamera liegt. Die Prozedur zur Durchführung dieses Winkeltests zeigt Algorithmus 2, wobei die von der Hardware bedingten maximalen Stellwinkel als bekannte Konfigurationsparameter vorausgesetzt werden.

Algorithmus 2 Akzeptierter Winkelbereich der PTU

Eingabe: Landmarkenposition im BFF r_L^{BFF}

Ausgabe: Wahrheitswert ob Position im akzeptierten Winkelbereich liegt

$[\alpha, \beta] :=$ Kugelkoordinaten(r_L^{BFF})
if $\alpha < 0.75 \cdot \text{maxPan}$ **and**
  $\alpha > 0.75 \cdot \text{minPan}$ **and**
  $\beta < 0.75 \cdot \text{maxTilt}$ **and**
  $\beta > 0.75 \cdot \text{minTilt}$ **then**
    return true
else
    return false
end if

Die Berechnungen des Steuermoduls der PTU werden durch den Datenfluss (s. 2.3.2 und Abb. 5) angestoßen, wenn ein neuer Zustand gegebenenfalls mit Fehlerzustand und Kovarianz als Ausgabe des Navigationsfilters am Eingang vorliegt. Ebenso werden zur Nachverfolgung wie bereits erläutert neue Kameramessungen

als Eingangsdaten berücksichtigt. Entweder wird dann das Nachverfolgen der Landmarke fortgesetzt oder es muss eine neue Landmarke ausgewählt werden. In den folgenden Algorithmen wird der Beginn der Nachverfolgung der Landmarke l durch die Prozedur InitialisereNachverfolgung(l) angegeben. Dabei existiert auch eine spezielle leere Landmarke l_{leer}. Wir die Prozedur zur Initialisierung der Nachverfolgung mit dieser Landmarke aufgerufen, so wird die PTU in die Default-Position mit einem Pan- und Tilt-Winkel von jeweils 0° und damit einer Blickrichtung orthogonal zur Seitwärts- und entlang der Vorwärtsachse ausgerichtet, was einem Blick noch vorne entspricht. Dieses Default-Verhalten ist deshalb notwendig, damit auch für den Fall, dass aktuell keine Landmarke aus dem Katalog sichtbar ist oder die sonstigen Kriterien zur Auswahl nicht erfüllt sind, ein definiertes Verhalten der PTU-Steuerung vorliegt. Sobald am Eingang des Steuermoduls neue Informationen vorliegen, wird bei aktuell aktiven Default-Verhalten die Auswahl einer Landmarke zur Nachverfolgung erneut gestartet.

Die im Rahmen dieser Arbeit entwickelten und implementierten Strategien zur Auswahl der nachzuverfolgenden Landmarke werden im Folgenden dargestellt. Dazu wird zunächst der Ansatz beschrieben, der die Minimierung der Zustandsvarianz zum Ziel hat. Da die dazu notwendigen Berechnungen aufwändig sind, wurden ebenfalls verschiedene Heuristiken umgesetzt, die im darauffolgenden Abschnitt beschrieben werden. Abschließend wird kurz auf die Umsetzung der Positionierung der Kamera bei einem festen Blickwinkel eingegangen.

3.1 Minimierung der Zustandsvarianz

Der UKF, der das Kernelement des Navigationsfilters darstellt und auch die Daten von Kamera- und Lidar-Sensor zur Bestimmung des Flugzustands berücksichtigt, ist insofern ein optimaler Schätzer, dass auf Basis der zur Verfügung stehenden Messdaten die Varianz der Schätzung minimiert wird (s. 2.1.2). Dies stellt auch die Motivation dar, die Minimierung der Zustandsvarianz als Ziel der Auswahl des Blickwinkels der Kamera anzustreben. In Kombination mit dem Ansatz des Feature Trackings besteht also die Anforderung darin, die Landmarke zu identifizieren, deren Positionierung im Zentrum des Kamera FOV die zu erwartende Zustandsvarianz minimiert. Dazu wird zunächst die Berechnung der durch eine Messung und deren Berücksichtigung durch den UKF erwarteten Kovarianzmatrix benötigt. Danach müssen die Kovarianzmatrizen der verschiedenen möglichen Messungen verglichen werden, wozu verschiedene Möglichkeiten bestehen. Diese beiden Aspekte werden in den zwei folgenden Abschnitten dargestellt.

3.1.1 Berechnung der erwarteten Kovarianz

Zur Berechnung der Kovarianzmatrix, die durch eine potentielle Messung und deren Berücksichtigung im UKF erhalten wird, muss diese Messung emuliert werden. Dazu kann die Unscented Transformation genutzt werden, die auch im UKF zum Einsatz kommt (s. 2.3.3). Als potentielle Eingabe für das Feature Tracking kommt dabei die Ausrichtung der Kamera auf jede der Landmarken l aus dem Landmarkenkatalog L in Fragen. Algorithmus 3 beschreibt den Ablauf der Berechnung der zu erwartenden Kovarianzmatrix $\hat{P}$, wenn die Kamera auf die Landmarke l ausgerichtet ist, für jede Landmarke, die innerhalb des Sichtbarkeitsbereichs bezüglich der aktuellen Position des UAVs liegt (Alg. 1). Als Eingabe werden der aktuelle Zustand x, Fehlerzustand δx sowie dessen Kovarianzmatrix P und der Landmarkenkatalog L benötigt. Es wird weiterhin vorausgesetzt, dass die Hardware-Eigenschaften von PTU, wie minimale und maximale Pan- und Tilt-Winkel (minPan, minTilt, maxPan, maxTilt), Kamera, wie Brennweite f, und Messbereich des Lidar-Sensors (minLidar, maxLidar) bekannt sind. Genauso stellen die Standardabweichungen der Messungenauigkeit einer Kameramessung in Pixel σ_{px} sowie einer Abstandmessung σ_d innerhalb der Prozedur bekannte Werte dar. Als Ausgabe wird eine Menge von Tupeln aus Landmarke l zur Zuordnung, zugehöriger Kovarianzmatrix $\hat{P}$ und den Werten sumDist und PTUDist erhalten. Letztere stellen Prüfwerte (*engl.* Tiebreaker) dar, um zwischen Messungen mit identischen erwarteten Kovarianzmatrizen zu entscheiden. Die Details der Berechnungen werden im Folgenden dargestellt.

Zunächst gilt, wie für die im Rahmen der Simulation eingesetzte Emulation einer Kamera-Messung (s. 2.3.3), dass die Unsicherheit der Position der Landmarke im Bild sowie die der Abstandsmessung für jede Landmarke identisch ist. Diese kann somit durch den 3-Vektor der Varianzen σ_l^2 beschrieben werden. Die Umrechnung des Fehlers von einer Ungenauigkeit im Kamerabild in Pixeln σ_{px} in einen Positionsfehler der Winkel in Kugelkoordinaten σ_α kann durch

$$\sigma_\alpha = 2 \cdot \arctan\left(\frac{\sigma_{px}}{2 \cdot f}\right) \tag{53}$$

in analoger Weise für die vertikale und die horizontale Bildachse erfolgen. Dazu wird das Kameramodell der Lochkamera mit der Brennweite f zugrunde gelegt.

Die Unscented Transformation wird für alle Landmarken des Katalogs als Messpunkte zusammen durchgeführt, da die Sichtbarkeit der Landmarken und die Messbarkeit der Abstände im Rahmen des genutzten Modells erst für die zu erwartende Messung $\hat{z}$ bezüglich des erwarteten Zustands $\hat{x}$ bestimmt werden kann. Anderenfalls würden gegebenenfalls Landmarken ignoriert, die bei Berücksichtigung der aktuel-

Algorithmus 3 Erwartete Kovarianzmatrix der Kameramessungen

Eingabe: Zustand x, Fehlerzustand δx, Kovarianz P, Landmarkenkatalog L

Ausgabe: Menge $\{P\}$ von Tupeln (l, $\hat{P}$, sumDist, PTUDist)

Bestimme σ_l^2 aus σ_{px} und σ_d
Erweitere δx und P um $l \in L$ zu δx_a und P_a
Bestimme $N = 2 \cdot \dim(\delta x_a) + 1$ Sigma-Punkte s für δx_a und P_a
for $s^j \in s$ **do**
    for $l \in L$ **do**
        $r_L^{BFF} := \Omega_{UAV}^{-1} \cdot \left(r_L^{ECEF} - r_{UAV}^{ECEF}\right)$
        $g^j(l) :=$ Kugelkoordinaten(r_L^{BFF})
    end for
end for
g += v_k aus s
Bestimme $\hat{z}$, P_{zz}, P_{xz}, $\hat{x}$
for $l \in L$ **do**
    if not SichtbarerAbstand(l) **then**
        Entferne l aus L und aus $\hat{z}$, P_{zz}, P_{xz}
    end if
end for
for $l^z \in L$ **do**
    Dupliziere P_{zz}, P_{xz} zu P_{zz}^d, P_{xz}^d
    PTUDist := $d_{geo}(PTU_z, l^z)$
    for $l^a \in L$ **do**
        if CoSichtbar(l^z, l^a) **then**
            sumDist += $d_{geo}(l^z, l^a)$
            if not minLidar $< d \in l^a <$ maxLidar **then**
                Entferne $d \in l^a$ aus P_{zz}^d und P_{xz}^d
            end if
        else
            Entferne l^a aus P_{zz}^d und P_{xz}^d
        end if
    end for
    Bestimme $K = zx^d \cdot (P_{zz}^d)^{-1}$
    Bestimmt $\hat{P}$ aus P, K und P_{zz}^d
    Füge (l^z, $\hat{P}$, sumDist, PTUDist) zur Ausgabemenge $\{P\}$ hinzu
end for
return $\{P\}$

len Unsicherheit des geschätzten Fehlerzustands doch messbar sind. Darüber hinaus muss die Transformation zwar auf einem großen Datensatz, jedoch nur einmal durchgeführt werden, was in einer Mehrzahl der Fälle vermutlich eine effizientere Berechnung erlaubt. Zur Durchführung der Transformation werden der aktuelle Fehlerzustand und die Kovarianz, die als Eingabe vorliegen, um alle Landmarken des Katalogs erweitert. Pro Landmarken wird der Fehlerzustand um einen 3-Vektor mit 0 als Werten erweitert und die Kovarianz um einen Block, der sich aus der Diagonalmatrix mit der Varianz σ_l^2 für horizontalen und vertikalen Winkel sowie Abstand ergibt. Mit $L = \dim(\delta x_a)$, der Größe des so erweiterten Fehlerzustands δx_a, werden mit den Gewichten nach Gleichung 32 $N = 2L + 1$ Sigma-Punkte generiert (Gl. 26), wie auch bei Anwendung des UKFs beim Vorliegen realer Messdaten.

Pro Sigma-Punkt s^j wird mit dem in diesem enthaltenen Fehlerzustand und dem zuletzt geschätzten Zustand x aus der Eingabe ein für diesen Sigma-Punkt korrigierter Zustand nach Gleichung 20 bestimmt. In diesem Zustand enthalten sind die Position des UAV in ECEF-Koordinaten r_{UAV}^{ECEF} sowie die Lage, welche in Form der Rotationsmatrix Ω_{UAV} zur Berechnung der Position der Landmark im BFF entsprechend

$$r_L^{BFF} = \Omega_{UAV}^{-1} \cdot \left(r_L^{ECEF} - r_{UAV}^{ECEF} \right) \tag{54}$$

genutzt wird. Diese Position $r_L^{BFF} = \begin{bmatrix} x & y & z \end{bmatrix}^\top$ kann nach

$$\begin{bmatrix} \alpha \\ \beta \\ r \end{bmatrix} = \begin{bmatrix} -\arctan2\left(y, x\right) \\ -\arctan2\left(z, \sqrt{(x+y)^2}\right) \\ \sqrt{x^2 + y^2 + z^2} \end{bmatrix} \tag{55}$$

in Kugelkoordinaten überführt werden, womit die Messung dieser Landmarke abgebildet wird. Diese Berechnung wird durch die Prozedur Kugelkoordinaten(r) in Algorithmus 3 dargestellt. Durch die Bearbeitung aller Landmarken für jeden Sigma-Punkt ergeben sich so die Gamma-Punkte g^j. Zu diesen wird noch das Messrauschen addiert, welches beim Erstellen der Sigma-Punkte nach Gleichung 33 als Block v_k in diesen enthalten ist.

Mit den bereits bestimmten Gewichten kann nach Gleichung 35 der zu erwartende Messvektor $\hat{z}$ und nach den Gleichungen 36 und 37 die Kovarianz der Messung P_{zz} beziehungsweise die Kreuzkovarianz zwischen Fehlerzustand und Messung P_{xz} bestimmt werden. Aus dem Sigma-Punkt s^0 kann ausgehend von den gewählten Gewichten aus dem zuletzt geschätzten Zustand x auch ein mittlerer korrigierter Zustand $\hat{x}$ berechnet werden.

Durch eine weitere Iteration über alle Landmarken wird so zunächst geprüft, ob diese aufgrund ihrer Größe und ihres Abstands detektierbar sind (Alg. 1). Dabei

wird die Größe der Landmarke d_L aus dem Landmarkenkatalog L, die Landmarkenposition aus der erwarteten Messung $\hat{z}$ und die Position des UAV aus dem mittleren korrigierten Zustand $\hat{x}$ entnommen. Nach dem Ergebnis dieser Prüfung werden nicht detektierbare Landmarken aus dem Katalog L und dem Messvektor $\hat{z}$ gelöscht. Ebenso werden aus den Kovarianzmatrix P_{zz} und P_{xz} die Spalten und Zeilen entfernt, die zu dieser Landmarke korrespondieren.

Über den auf die detektierbaren Landmarken reduzierten Katalog wird nun in einer äußeren und einer inneren Schleife iteriert. Zu Beginn eines Durchlaufs der äußeren Schleife werden jeweils P_{zz} und P_{xz} dupliziert, um die Kopien in jedem Durchlauf neu verändern zu können. Die Landmarke aus der äußeren Schleife stellt dabei die potentiell nachzuverfolgende Landmarke l^z dar. Es wird somit angenommen, dass diese im Zentrum des Kamerabilds liegt. Für alle Landmarken wird nun in der inneren Schleife geprüft, ob diese bezüglich des Öffnungswinkels der so positionierten Kamera ebenfalls sichtbar sind und so simultan mit der nachzuverfolgenden Landmarke beobachtet würden. Dieser Test wird durch die Prozedur CoSichtbar(l_1, l_2) nach Algorithmus 4 durchgeführt. Hierbei wird erneut vorausgesetzt, dass das Kameramodell und die Kameraparameter Brennweite f und Anzahl der Pixel in horizontaler N^h_{px} und vertikaler Richtung N^v_{px} innerhalb der Prozedur bekannt sind. Ebenfalls werden die zur Bestimmung der Position der Landmarke im BFF benötigte Position und Lage des UAVs aus dem mittleren korrigierten Zustand $\hat{x}$ als bekannt angenommen. Das FOV der Kamera in Kugelkoordinaten wird analog zu Gleichung 53 berechnet, wobei in der Prozedur halbe Öffnungswinkel als $\Delta\alpha$ und $\Delta\beta$ bestimmt werden.

Ist eine Landmarke nicht gemeinsam mit der nachzuverfolgenden Landmarke sichtbar, werden die zugehörigen Einträge in den kopierten Kovarianzmatrizen entfernt. Anderenfalls wird noch geprüft, ob die Abstandsmessung innerhalb des Messbereichs des Lidar-Sensors möglich ist. Falls dies nicht der Fall ist, wird nur die Abstandsinformation für diese Landmarke aus den Kovarianzen entfernt. Für jede so generierte Messung von gleichzeitig sichtbaren Landmarken kann nach den Gleichungen 38 und 39 die zu erwartende Kovarianz dieser Messung bestimmt werden. Diese wird zusammen mit der Identifikation der für diese Messung nachzuverfolgenden Landmarke abgespeichert, sodass am Ende der verschachtelten Iteration die Menge der Kovarianzmatrizen ausgegeben werden kann.

Während der Iteration werden noch die Tiebreaker ermittelt und zur Ausgabe abgespeichert. Dazu wird zunächst die Summe der geodätischen Abstände aller gleichzeitig sichtbaren Landmarken zum Zentrum des FOV der Kamera und somit der Position der nachzuverfolgenden Landmarke als sumDist gespeichert, wobei ein

Algorithmus 4 Bestimmung gleichzeitig sichtbarer Landmarken

Eingabe: Landmarke l_1 im Zentrum des Blickfeld, Landmarke l_2

Ausgabe: Wahrheitswert ob l_2 im Kamerablickfeld mit l_1 im Zentrum sichtbar ist

$\Delta\alpha := \arctan\left(N_{px}^{h}/(2 \cdot f)\right)$
$\Delta\beta := \arctan\left(N_{px}^{v}/(2 \cdot f)\right)$
$r_1^{BFF} = \Omega_{UAV}^{-1} \cdot \left(r_1^{ECEF} - r_{UAV}^{ECEF}\right)$
$[\alpha_1,\, \beta_1] :=$ Kugelkoordinaten(r_1^{BFF})
$r_2^{BFF} = \Omega_{UAV}^{-1} \cdot \left(r_2^{ECEF} - r_{UAV}^{ECEF}\right)$
$[\alpha_2,\, \beta_2] :=$ Kugelkoordinaten(r_2^{BFF})
if $\alpha_2 \geq \alpha_1 - \Delta\alpha$ **and**
  $\alpha_2 \leq \alpha_1 + \Delta\alpha$ **and**
  $\beta_2 \geq \beta_1 - \Delta\beta$ **and**
  $\beta_2 \leq \beta_1 - \Delta\beta$ **then**
    return true
else
    return false
end if

geringerer Wert eine Messung repräsentiert, bei der die beobachteten Landmarken bei einer Positionsänderung unwahrscheinlicher aus dem Kamerabild verschwinden, da sie im Mittel näher am Zentrum des Bildes liegen. Falls auch dieser Prüfwert für zwei Kovarianzen identisch ist, wird eine geringere Auslenkung der PTU von ihrer mittleren Stellung bevorzugt, da hier ein größerer Bewegungsspielraum für die Nachverfolgung besteht. Die geodätischen Abstände

$$d_{geo} = \arccos\left(\sin\left(\alpha_z\right) \cdot \sin\left(\alpha\right) \cdot \cos\left(\beta_z - \beta\right) + \cos\left(\alpha_z\right) \cdot \cos\left(\alpha\right)\right) \tag{56}$$

ergeben sich für beide Tiebreaker aus dem horizontalen Winkel α und dem vertikalen Winkel β in Kugelkoordinaten wobei der Index z jeweils für die Zentralposition steht. Im Falle der PTU ergibt sich der Wert PTU_z als Schwerpunkt des Winkelbereichs in dem die PTU bewegt werden kann.

3.1.2 Skalar zum Vergleich

Die für die potentiellen Messungen erwarteten Kovarianzmatrizen können als solche nicht direkt verglichen werden, da sie mehrere Werte unterschiedlicher Bedeutung enthalten. Vielmehr wird ein Skalar benötigt, welches die Qualität der Messung bewertet und dessen Wert direkt mit dem anderer Messungen verglichen werden kann.

Algorithmus 5 stellt dabei das allgemeine Vorgehen unter Berücksichtigung der unter 3.1.1 beschriebenen Tiebreaker vor. Als Eingabe wird die mithilfe von Algorithmus 3 erzeugte Menge $\{P\}$ benötigt, die die Kovarianzen und Tiebreaker enthält. Ebenso muss eine Strategie zur Bestimmung des Skalar ausgewählt werden. Durch Vergleich der Skalare und Durchführung des Winkeltests kann am Ende die Nachverfolgung der nach der gewählten Strategie optimalen Landmarke initialisiert werden. Wird hierbei keine geeignete beziehungsweise sichtbare Landmarke gefunden, wird durch die Initialisierung von l_{min} mit der leeren Landmarke das Default-Verhalten ausgelöst.

Algorithmus 5 Vergleich von Kovarianzmatrizen

Eingabe: Menge $\{P\}$ von Tupeln (l, $\hat{P}$, sumDist, PTUDist), Strategie S
Ausgabe: Nachverfolgung der gewählten Landmarke

```
Initialisiere l_min = l_leer, k_min, sumDist_min, PTUDist_min
for l, P̂ ∈ {P} do
    if Winkeltest(l) then
        k := Skalar(S, P̂)
        if (k < k_min) or
            (k = k_min and sumDist < sumDist_min) or
            (k = k_min and sumDist = sumDist_min and PTUDist < PTUDist_min)
        then
            l_min := l
            k_min := k
            sumDist_min := sumDist
            PTUDist_min := PTUDist
        end if
    end if
end for
InitialisiereNachverfolgung(l_min
```

Zunächst kann zur Bestimmung des Skalars als Strategie die Bildung der Summe der Varianzen nach

$$\mathrm{Tr}(\hat{P}) = \sum_{i=1}^{25} p_{i,i} \tag{57}$$

aus den Diagonalelementen $p_{i,i}$ der erwarteten Kovarianzmatrix des Fehlerzustands $\hat{P}$ genutzt werden. Dadurch wird die Spur der Kovarianzmatrix $\mathrm{Tr}(\hat{P})$ minimiert.

Diese Minimierung ist optimal im Sinne des Kalman-Filters, der auch zu einer Varianzminimierung führt (s. 2.1.2).[BSLK01][Sor70]

Die Problemstellung ist hierbei jedoch nicht identisch, da nicht die optimale Schätzung auf Basis einer verfügbaren Messung gesucht wird, sondern eine optimale Messung für die nächste Schätzung. Die Minimierung der Spur der Kovarianzmatrix wird in der Literatur jedoch auch für die Wahl zwischen verschiedenen Sensoren oder unterschiedlichen Sensorpositionen eingesetzt.[KS78][GCHM06] Als Nachteil für diesen Ansatz ist bekannt, dass verschiedene Größenordnungen und Skalen der verschiedenen Größen, deren Varianzen in der Kovarianzmatrix dargestellt werden, dazu führen, dass diese unterschiedlich gewichtet berücksichtigt werden.[HH17]

Unter 2.2.1 wurde bereits die von Davison und Murray beschriebene Auswertung des Volumens des 3σ-Ellipsoids der Residualkovarianzmatrix für das SLAM diskutiert.[DM98] Einen ähnlichen Ansatz stellt die Berechnung der Determinante der Kovarianzmatrix $\det(\hat{P})$ als Skalar zum Vergleich der Kovarianzmatrizen dar. Diese beschreibt das Volumen des so genannten Konfidenz-Ellipsoids, der die Abhängigkeiten zwischen den verschiedenen Variablen des Fehlerzustands berücksichtigt, welche bei $\mathrm{Tr}(\hat{P})$ vernachlässigt werden.[HH17][MCM09] Darüber hinaus wird der Einfluss verschiedener Größenordnungen auf das Resultat abgemildert, da die Stärke des Einflusses einer Größe auf das Volumen des Ellipsoids auch von der Ausprägung der jeweils anderen Größen abhängt. Einen technischen Nachteil der Betrachtung der Determinante stellt der Fakt dar, dass diese bei vollständiger Korrelation zweier Variablen immer 0 beträgt. Dies ist jedoch bei einer Anwendung mit realen Daten unter Rauscheinflüssen nicht zu erwarten.

Um nur die Varianzen zu Berücksichtigen und trotzdem die verschiedenen Größenordnungen der unterschiedlichen Größen des Fehlerzustands auszugleichen, wurde als dritter Ansatz zur Berechnung des Skalars zum Vergleich der Kovarianzmatrizen eine Normierung eingesetzt. Dieser Ansatz wird im Folgenden als spannweitennormierte Varianz bezeichnet. Dabei wurde pro Varianzeintrag $p_{i,i}$ der maximale $\max(p_{i,i})$ und minimale Wert $\min(p_{i,i})$ aus allen zu vergleichenden Kovarianzmatrizen $\hat{P} \in \{P\}$ erfasst. Diese können vor der Berechnung nach Algorithmus 3.1.2 durch eine Iteration über die Menge der Kovarianzmatrizen und elementweise Bildung des Minimums und Maximums bestimmt werden. Durch eine Normalisierung nach

$$t_i = \frac{p_{i,i} - \min(p_{i,i})}{\max(p_{i,i}) - \min(p_{i,i})} \in [0, 1] \tag{58}$$

ergibt sich pro Varianz die relative Abweichung von der besten möglichen Messungen ($t_i = 0$) in der jeweiligen Dimension i des Fehlerzustands. Mit der Summe der nor-

mierten Werte $\sum_{i=1}^{25} t_i$ ergibt sich ein Vergleichswert, der die gesamte Abweichung von einer, in der Regel fiktiven, in jeder Dimension bestmöglichen Messung widerspiegelt. Damit wird ein Vergleich auf Basis eines pro Varianz identischen Wertebereichs erreicht. Der erhaltene Wert gibt jedoch nicht wieder, ob es sich um eine Verbesserung oder Verschlechterung gegenüber vorangegangenen Messungen handelt. Diese Vernachlässigung scheint jedoch sinnvoll, da ohnehin keine bessere Messung zum aktuellen Zeitpunkt möglich ist, selbst wenn es sich um eine Verschlechterung handelt.

Damit stehen drei Strategien zur Verfügung, die alle auf der Schätzung der erwarteten Kovarianzmatrix möglicher Messungen basieren (s. 3.1.1) und diese mit einem Skalar bewerten. Dieses Skalar ergibt sich entweder aus der Spur, der Determinante oder der spannweiten-normierten Varianzen dieser Kovarianzmatrizen. Bei identischen Skalaren werden Tiebreaker zur Bestimmung der nächsten nachzuverfolgenden Landmarke eingesetzt. Dazu wird zunächst die Messung gewählt, bei der sich die gemessenen Landmarken näher am Schwerpunkt des Kamerabilds befinden. Wird auch hierdurch keine Entscheidung erreicht, wird die Messung gewählt, bei der die PTU am wenigsten weit von ihrer Schwerpunktposition weg bewegt werden muss.

3.2 Heuristiken

Die in Abschnitt 3.1 dargestellte Auswahl der nachzuverfolgenden Landmarke anhand der zu erwartenden Messung stellt einen rechenintensiven Ansatz dar, selbst wenn die Unscented Transformation nur einmal für alle Landmarken des Katalogs durchgeführt werden muss. Dies ist auf die Vielzahl von Matrixoperationen wie Multiplikationen, Inversionen und Transpositionen zurückzuführen, die für die Berechnungen notwendig sind. Insbesondere die Cholesky-Zerlegung, die zur Bestimmung der Matrixwurzel (Gl. 26) eingesetzt wird[AAF+18], skaliert mit $\mathcal{O}(n^3)$ für die Zerlegung einer $n \times n$-Matrix.[Mei14]

Wie in der Analyse der Resultate (s. 4.2) dargestellt, ist die Auswahl auf Basis der zu erwartenden Kovarianz wie unter 3.1.1 beschrieben für umfangreiche Landmarkenkataloge auf der Hardware des superArtis tatsächlich nicht mehr echtzeitfähig. Deshalb wurden ebenfalls verschiedene Heuristiken für die Auswahl der nachzuverfolgenden Landmarke implementiert und getestet. Die Implementierungen der Heuristiken für eine zufällige Auswahl, der Wahl der Landmarke mit dem kleinsten Abstand zum UAV und der Landmarke mit den meisten gleichzeitig im Kamerabild sichtbaren Landmarken werden im Folgenden beschrieben.

3.2.1 Zufällige Auswahl

Die zufällige Auswahl einer Landmarke zur Nachverfolgung ist keine Heuristik von der eine Nachbildung der Auswahl auf Basis der Kovarianz zu erwarten ist. Vielmehr kann durch die Performance dieser Strategie abgeschätzt werden, wie gut eine andere Strategie beziehungsweise Heuristik mindestens sein muss, um tatsächlich nützlich zu sein. Anderenfalls wäre die zufällige Auswahl und somit keine Strategie besser geeignet als die zu vergleichende Strategie.

Algorithmus 6 Auswahl einer zufälligen Landmarke

Eingabe: Landmarkenkatalog L, Zustand x

Ausgabe: Nachverfolgung der gewählten Landmarke

```
Initialisiere l_out := l_leer, i := 0
while i < V_max do
    l := rand(L)
    if SichtbarerAbstand(l) then
        if Winkeltest(l) then
            l_out := l
            break
        end if
    end if
end while
InitialisiereNachverfolgung(l_out)
```

Zur Umsetzung der zufälligen Auswahl wird eine Landmarke mithilfe der rand()-Prozedur zufällig aus dem Katalog bestimmt. Auf Basis des aktuellen Zustands des UAVs, der für diese Strategie als Eingabe ebenfalls benötigt wird, wird die Position der Landmarke im BFF bestimmt und damit die Sichtbarkeit bezüglich des Abstands mithilfe von Algorithmus 1 geprüft. Aus dieser Position ergibt sich auch die nötige Positionierung der PTU zur Nachverfolgung der Landmarke. Somit kann auch der Winkeltest durchgeführt werden. Ist die zufällig gewählte Landmarke somit tatsächlich messbar, wird die nachverfolgung dieser initialisiert. Anderenfalls wird der Auswahlprozess wiederholt. Dabei wird eine maximale Anzahl von Versuchen V_{max} gewählt, nach denen die Suche abgebrochen wird. Hierfür wurde in der Implementierung 100 als Wert festgelegt. Dieses Abbruchkriterium verhindert eine Endlosschleife, falls aktuell keine sichtbaren Landmarken zur Verfügung stehen. Gleichzeitig muss nicht Nachverfolgt werden, welche Landmarken aus dem Katalog bereits zufällig ausgewählt wurden. Falls keine Auswahl erfolgt, wird die zur Initia-

lisierung genutzt leere Landmarke zur Nachverfolgung gesetzt, woraus das Default-Verhalten erfolgt (s. 3). Algorithmus 6 fasst die Prozedur zur zufälligen Auswahl zusammen.

3.2.2 Kleinster Abstand zur Landmarke

Die Landmarke mit dem geringsten Abstand nachzuverfolgen, scheint aus mehreren Gründen eine geeignete Heuristik zu sein. Zunächst ist für die Landmarke mit geringstem Abstand die höchste Wahrscheinlichkeit gegeben, dass neben der Positionsinformation auch die Abstandsinformation erhalten werden kann und so zusätzliche Messdaten für die Zustandsschätzung zur Verfügung stehen. Dies gilt, da aufgrund des Messbereichs des Abstandssensors und der Größe des superArtis (s. 2.3.1) davon auszugehen ist, dass Landmarken, wenn sie außerhalb des Messbereichs des Lidar-Sensors liegen, in der Regel zu weit entfernt und nicht zu nah sind. Darüber hinaus können an den Landmarken mit geringem Abstand häufig genauere Messung vorgenommen werden, da diese aufgrund der größeren Abbildung sicherer im Kamerabild identifizierbar sind.

Eine weitere Motivation stammt aus dem typischen Anwendungsfall des Überflugs eines Terrains in einer gewissen Höhe oberhalb aller Landmarken. Daraus resultiert, dass die Landmarken mit geringstem Abstand eher unterhalb des UAVs lokalisiert sind. Damit wird durch diese Strategie zwar immer eine Landmarke in einer ähnlichen Lage bezüglich des UAVs verfolgt, gleichzeitig sind aber weite Bewegung der PTU unwahrscheinlicher, sodass beim Umschwenken der Kamera weniger Messungen an Bereichen, die nicht das Ziel der Messung sind, durchgeführt werden.

Zur Bestimmung der Landmarke mit den geringsten Abstand wird Algorithmus 7 eingesetzt. Dazu wird über den gesamten Landmarkenkatalog iteriert. Der Abstand zwischen Landmarke und UAV ergibt sich aus der Norm der Position der Landmarke im BFF (Gl. 54), wozu erneut Position und Lage des UAVs aus dem zuletzt geschätzten Zustand x benötigt werden. Wird eine Landmarke mit potentiell geringstem Abstand gefunden, so wird geprüft ob diese im Bereich der sichtbaren Abstände liegt (Alg. 1) und im Rahmen des Winkeltests der PTU-Ausrichtung akzeptiert wird (Alg. 2). In diesem Fall wird die Landmarke als Kandidat zwischengespeichert. Am Ende der Iteration ist die sichtbare Landmarke mit geringstem Abstand zwischengespeichert und deren Nachverfolgung wird initialisiert. Auch hier wird durch die Initialisierung mit l_{leer} das Default-Verhalten garantiert, falls keine Auswahl erfolgt.

Algorithmus 7 Landmarke mit geringstem Abstand

Eingabe: Landmarkenkatalog L, Zustand x

Ausgabe: Nachverfolgung der gewählten Landmarke

```
Initialisiere l_min := l_leer, d_min
for l ∈ L do
    Berechne r_l^BFF
    if ||r_l^BFF|| < d_min then
        if SichtbarerAbstand(l) then
            if Winkeltest(l) then
                d_min := ||r_l^BFF||
                l_min := l
            end if
        end if
    end if
end for
InitialisiereNachverfolgung(l_min)
```

3.2.3 Höchste Anzahl sichtbarer Landmarken

Die Auswahl der Landmarke zur Nachverfolgung, mit der gemeinsam die größte Anzahl von Landmarken im Blickfeld der Kamera liegt, kann dadurch motiviert werden, dass eine größere Anzahl von Messpunkten wahrscheinlicher geeignet ist, die Varianz der Zustandsschätzung stärker zu minimieren als eine geringere Anzahl von Messpunkten. Dabei liegt der konkrete Einfluss jedoch an der in der Messung enthaltenen Information, sodass es sich bei dieser Strategie nur um eine Heuristik handelt.

Wie durch Algorithmus 4 dargestellt, kann das FOV der Kamera in ein Winkelinkrement und die Position der Landmarken in Kugelkoordinaten übertragen werden (Gl. 55), um die gleichzeitige Sichtbarkeit von Landmarken in einem Kamerabild zu prüfen. Hierzu muss in Algorithmus 8 zur Implementierung der Strategie wieder der zuletzt geschätzte Zustand x des UAVs sowie das Kameramodell und die Kameraparameter bekannt sein.

Zur Implementierung der Strategie (Alg. 4) wird der Katalog der Landmarken zweifach in einer äußeren und einer inneren Schleife iteriert. Es werden dabei nur solche Landmarken weiter berücksichtigt, die im Bereich der sichtbaren Abstände liegen (Alg. 1). Die in der äußeren Schleife aktuell ausgewählte Landmarke l^z stellt dabei die potentiell nachzuverfolgende Landmarke dar. Im Falle der gleichzeitigen

Algorithmus 8 Auswahl der Landmarke mit größter Anzahl gleichzeitig sichtbarer Landmarken

Eingabe: Landmarkenkatalog L, Zustand x

Ausgabe: Nachverfolgung der gewählten Landmarke

```
Initialisiere l_max := l_leer, N_max := 0, sumDist_max, PTUDist_max
for l^z ∈ L do
    if SichtbarerAbstand(l^z) then
        Initialisiere N := 0, sumDist := 0
        for l^a ∈ L do
            if SichtbarerAbstand(l^a) and CoSichtbar(l^z, l^a) then
                N += 1
                sumDist += d_geo(l^z, l^a)
            end if
        end for
        PTUDist := d_geo(PTU_z, l^z)
        if (N > N_max) or
           (N = N_max and sumDist < sumDist_max) or
           (N = N_max and sumDist = sumDist_max and PTUDist < PTUDist_max)
        then
            if Winkeltest(l^z) then
                N_max := N
                l_max := l^z
                sumDist_max := sumDist
                PTUDist_max := PTUDist
            end if
        end if
    end if
end for
InitialisiereNachverfolgung(l_max)
```

Sichtbarkeit mit der Landmarke l^a der inneren Schleife (Alg. 4) wird die Anzahl gleichzeitig sichtbarer Landmarken inkrementiert und der auch für diese Strategie genutzt Tiebreaker sumDist (s. 3.1.2) aktualisiert. Der Tiebreaker PTUDist wird am Ende des inneren Schleifendurchlaufs ebenfalls bestimmt.

Liegt am Ende des inneren Schleifendurchlaufs ein neues Maximum vor oder wird das aktuelle Maximum eingestellt und einer der Tiebreaker fällt zu Gunsten der aktuell betrachteten Landmarke aus, so wird abschließend geprüft, ob diese nach dem Winkeltest (Alg. 2) nachverfolgt werden sollten. In diesem Fall wird das neue Maximum als Kandidat mit den zugehörigen Werten der Tiebreaker zwischengespeichert.

Am Ende der Iteration ist die Landmarke mit der größten Anzahl gleichzeitig sichtbarer Landmarken zwischengespeichert und deren Nachverfolgung wird initiiert.

3.3 Statische Blickwinkel

Auch statische Blinkwinkel können mithilfe der PTU-Steuerung eingestellt werden. Der Vergleich der Resultate unter Anpassung des Blickwinkels mit den unter einem statischen Blickwinkel erhaltenen ist deshalb interessant, da statisch am UAV angebrachte Kameras den Regelfall für die optische Navigation darstellen (s. 2.1). Entsprechend ist hier, ähnlich wie im Vergleich zur zufälligen Auswahl der nachzuverfolgenden Landmarke (s. 3.2.1) eine bessere Performance erforderlich, um den technischen Mehraufwand für den Einsatz von active vision gegenüber einer fest installierten Kamera zu rechtfertigen.

Darüber hinaus bietet die Steuerung mithilfe der PTU auch die Möglichkeit, verschiedene statische Blickwinkel zu vergleichen, da diese mithilfe der Software reproduzierbar eingestellt werden können.

Zur Steuerung der PTU wird auch für statische Blinkwinkel der PID-Regler eingesetzt. Es werden die Zielwinkel der PTU-Achsen jedoch nicht anhand einer Position relativ zu einer Landmarke mithilfe der Prozedur InitialisiereNachverfolgung(l) gewählt, sondern direkt vorgegeben. Für einen ausreichend kalibrierten Regler ist somit zu erwarten, dass die gewählte statische Position ohne große Abweichungen beziehungsweise Schwankungen gehalten werden kann. Die Nutzung des PID-Reglers gegenüber einer einmaligen Bewegung in die gewünschte Position hat dabei den praktischen Vorteil, dass keine zwei Ansteuermechanismen für die PTU implementiert werden müssen.

Es wurden vier verschiedene statische Position gewählt und als Steuermodi der PTU implementiert. Alle vier Position haben dabei einen Pan-Winkel von 0° ge-

mein. Die Kamera steht somit immer orthogonal zur Seitwärtsachse des UAVs und die Strategie kann durch ihren Tilt-Winkel eindeutig beschrieben werden. Als Tilt-Winkel wurden 0°, 45°, 67.5° und 90° gewählt. 0° entspricht dabei einer Ausrichtung entlang der Vorwärtsachse, was auch dem Default-Verhalten (s. 3) bei Nachverfolgung der leeren Landmarke entspricht. 90° Grad hingegen stellt die Blickrichtung entlang der Abwärtsachse und somit einem Blick nach unten dar. 45° und 67.5° sind somit Blickrichtung schräg nach vorne.

4 Analyse und Vergleich der Strategien

Um die Charakteristika der verschiedenen implementierten Strategien zur Auswahl der nachzuverfolgenden Landmarken (s. 3) zu analysieren und die Strategien untereinander zu vergleichen, wurde die unter 2.3.3 beschriebene Simulationsumgebung eingesetzt.

Im Folgenden werden zunächst die gewählten Rahmenbedingungen der durchgeführten Simulationen beschrieben. Dazu gehörigen insbesondere die variierten Parameter sowie die Bestimmung der Dauer, für die eine Landmarke maximal nachverfolgt werden soll, bis der Auswahlprozess von neuem gestartet wird. Daran anschließend wird die Auswertung der Laufzeiten der Simulationen für die verschiedenen Strategien im Hinblick auf die Echtzeitfähigkeit des Systems für die Flugsteuerung des superArtis Hubschraubers vorgestellt. Abschließend folgt die Auswertung und der Vergleich der Performance der Strategien zur Landmarkenauswahl unter verschiedenen Bedingungen.

Um eine übersichtlichere Diskussion sowie kompaktere Darstellungen zu ermöglichen, werden die verschiedenen Strategien zur Auswahl der nachzuverfolgenden Landmarke im Folgenden an einigen Stellen durch Nummern abgekürzt. Diese Nummerierung gibt Tabelle 7 wieder.

	Strategie	Nummer
statisch	0° (nach vorne)	1
	45°	2
	67.5°	3
	90° (nach unten)	4
dynamisch	Zufällige Auswahl	5
	Geringer Abstand zum UAV	6
	Größte Anzahl im FOV	7
	Minimerung von $\mathrm{Tr}(\hat{P})$	8
	Minimierung von $\det(\hat{P})$	9
	Minimierung der spannweiten-normierten Varianz	10

Tabelle 7: Untersuchte Strategien zur Auswahl der nachzuverfolgenden Landmarke mit zugeordneter Nummer zur einfacheren Identifikation.

4.1 Rahmenbedingungen der Simulation

Eine Mehrzahl der Rahmenbedingungen ist während aller Simulationsdurchläufe identisch und fest vorgegeben. Diese festen Simulationsparameter wurden im Wesentlichen bereits unter 2.3 beschrieben und werden im Folgenden nochmals knapp rekapituliert.

Als Trajektorie wird die Lissajous-Figur in Abbildung 8 gewählt, wobei der Startpunkt auf die Koordinaten N51°51'21.4" E11°25'04.8" bei einer Höhe von 183 m festgelegt wurde. Dies entspricht der Startbahn des Flugplatzes Magdeburg-Cochstedt, auf dem vom DLR das nationale Erprobungszentrum für Unbemannte Luftfahrtsysteme betrieben wird. Die Trajektorie wird in einer Gesamtflugzeit von 300 s absolviert. Als absolute Zeit wurde für den Simulationsbeginn 0:00 Uhr am 1. Januar 2021 festgesetzt. Die Simulation erfolgt in Zeitschritten entsprechend einer Frequenz von 100 Hz, wobei das Logging der Sensordaten und Zwischenergebnisse mit an die Messfrequenzen angepassten Raten erfolgt.

Zur Beschreibung der Form der Erde sowie der Gravitation wird das WGS84-EGM96 Modell eingesetzt.[Age21] Das Magnetfeld der Erde wird entsprechend des World Magnetic Model modelliert.[fEI19] Der atmosphärischen Luftdruck wird auf Basis der ICAO-Standardatmosphäre berechnet.[Org93]

Die Initialisierung der UKF erfolgt wie unter 2.3.2.1 beschrieben mit den Werten entsprechend Tabelle 5. Auch die Bestimmung von Mess- und Prozessrauschen wird, wie dort beschrieben, durchgeführt. Für das Messrauschen werden die Werte entsprechend Tabelle 3 und für das Prozessrauschen entsprechend Tabelle 4 gewählt, wobei die diskreten Ausprägungen von Zufallswerten anhand von Normalverteilungen mit den jeweils in den Tabellen angegebenen Standardabweichungen bestimmt werden.

Neben Beschleunigungssensor und Gyroskop der IMU (Abtastrate je 100 Hz) sowie der Kamera mit Lidar (5 Hz) werden zusätzlich noch Barometer (50 Hz) und Magnetometer (20 Hz) als Sensoren simuliert. Letztlich wurde die Positions- und Geschwindigkeitsmessung per GPS (2 Hz bzw. 5 Hz) als Sensormessung berücksichtigt. Für diese beiden Messungen auf Basis des GPS-Sensors besteht die Besonderheit, dass diese nicht während der gesamten Simulation erhalten werden. Vielmehr wird die zusätzlichen Information aus der GPS-Messung nur zu Beginn des simulierten Fluges genutzt, damit die initial zu groß geschätzten Fehler des UKF zuverlässig konvergieren.

Zu diesem Zweck wurde für alle unter 4.3 diskutierten Resultate die Simulation mit allen beschriebenen Sensoren gestartet und die Positions- und Geschwindig-

keitsdaten des GPS-Sensors nur für die ersten 45 s des Fluges berücksichtigt. Die Abschätzung dieser Zeitdauer ist unter 4.1.1 beschrieben. Diese Situation stellt insofern ein realistisches Szenario dar, dass Flüge häufig in einem bekannten Gebiet starten und somit zusätzliche Informationen zur Initialisierung zur Verfügung stehen. Der Ausfall des GPS-Sensors durch eine Unterbrechung des Kontakts zu den Satelliten bei einem explorativen Flug ist dabei durchaus typisch, etwa wenn in einem bebauten Gebiet geflogen wird. In diesem Fall wird dann die optische Navigation besonders wichtig, um den Sensorausfall zu kompensieren. Im Rahmen des simulierten Szenarios kann also nach dem Wegfall der Messdaten des GPS-Sensors die tatsächliche Performance der Strategien zur Landmarkenauswahl in der optischen Navigation beurteilt werden, wobei zu Beginn ein zuverlässig initialisierter Navigationsfilter vorliegt. Für die unter 4.4 beschriebenen Ergebnisse wurde dahingegen vollständig auf den Einsatz der Daten des GPS-Sensors verzichtet, sodass hier die initiale Konvergenz des Navigationsfilters ausschließlich auf Basis der übrigen Sensoren erfolgen muss.

Die Simulation der Kamera- und Lidar-Messungen auf Basis des Landmarkenkatalogs wird wie unter 2.3.3 beschrieben durchgeführt. Dabei sind die Sensorparameter Brennweite ($f = 4\,\mathrm{mm}$), Auflösung ($800 \times 600\,\mathrm{px}$) und Pixelgröße ($6.45\,\mu\mathrm{m}$) der Kamera und der Messbereich des Lidar-Sensors (1 – $100\,\mathrm{m}$) festgelegt. Der Landmarkenkatalog wird für jeden Simulationslauf auf Basis neuer Zufallszahlen generiert. Die Größe der Landmarken wird für alle Simulationsläufe aus einer χ^2-Verteilung mit Verteilungsparameter 14 ermittelt. Die Anzahl und mittlere Höhe des Terrains wurden im Rahmen der Simulation variiert, ebenso wie die eingesetzte Strategie zur Nachverfolgung der Landmarken.

Die Anzahl der Landmarken im überflogenen Gebiet wurde dabei zwischen 25, 50, 100 und 200 variiert. Für die Simulation ohne GPS-Signal (s. 4.4) wurden zusätzlich Kataloge mit 400 Landmarken generiert. Die Anzahl der Landmarken wurde variiert, da zu erwarten ist, dass sie einen entscheidenden Einflussfaktor auf die Performance darstellt. Im Grenzfall einer geringen Anzahl liegen Situationen vor, in denen nur eine oder sogar keine Landmarke zu sehen ist. Hierdurch kann die Robustheit der unterschiedlichen Strategien bewertet werden. Im Falle einer möglichen Auswahl ist zu erwarten, dass diese besonders relevant für die Performance der Strategie sein wird. Dieser Grenzfall wird für 25 und 50 Landmarken abgebildet. Sind demgegenüber stets mehrere Landmarken sichtbar, so kann ermittelt werden, mit welcher Strategie insgesamt eine ideale Auswahl getroffen werden kann, wenn dauerhaft eine Wahlmöglichkeit besteht.

Ebenfalls wurde die maximale Höhe des Terrains und somit der Landmarken relativ zur mittleren Flughöhe variiert. Dabei wurden für den Parameter h_{rel} die Werte 0, 0.25, 0.5, 1 und 2 gewählt. Somit wird ein Übergang von einem Flug in großer Höhe bis zu einem Flug durch einen Raum abgebildet. Entsprechend kann durch die Variation dieses Parameter die Robustheit gegenüber verschiedenen Einsatzszenarien überprüft werden.

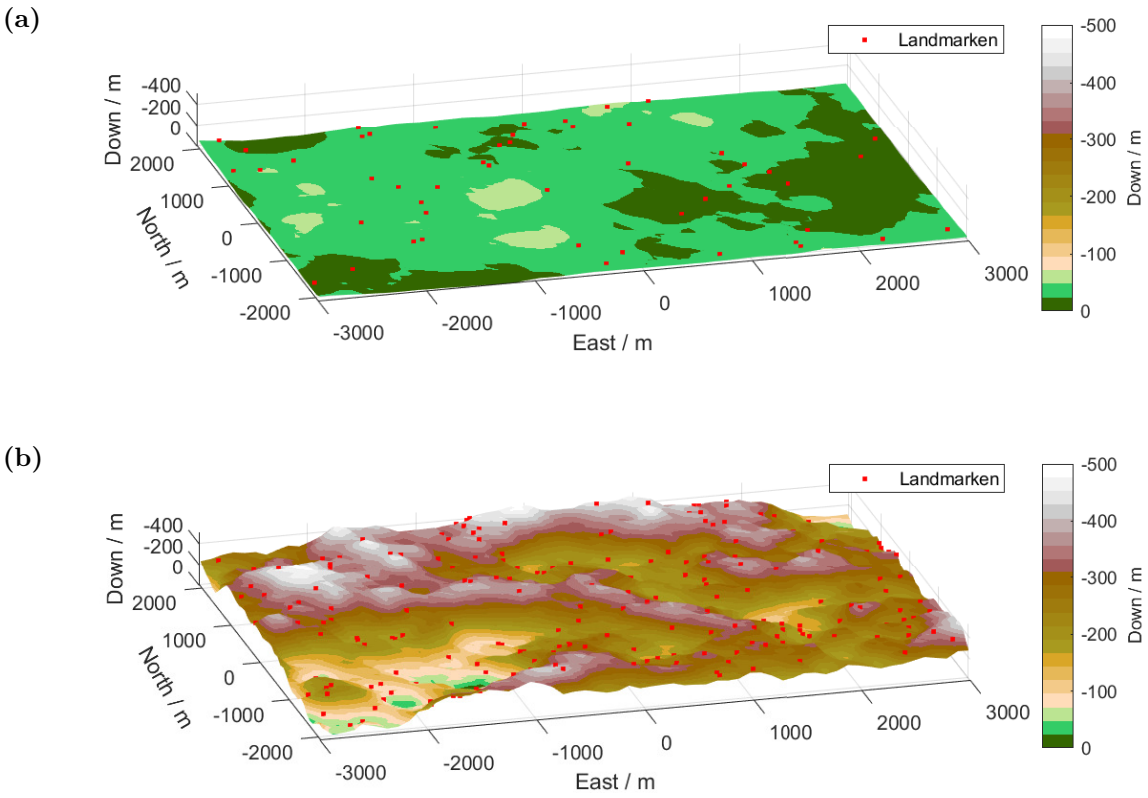

Abbildung 9: Exemplarische Verteilungen von Landmarken im Raum mit **(a)** 25 Landmarken und einer relativen Höhe von 0.5 und **(b)** 200 Landmarken und einer relativen Höhe von 2.0.

Aus der Bildung aller Paare dieser beiden Parameter ergeben sich 20 beziehungsweise 25 Parameterkombinationen, die pro Strategie simuliert wurden. Es wurden pro Parameterkombination jeweils 5 Simulationsdurchläufe mit unterschiedlich initialisiertem Zufallszahlengenerator durchgeführt, sodass eine gewisse Statistik über die Ergebnisse erzeugt wird und etwaige Einflüsse durch die Optimierung auf eine Zufallskonfiguration ausgeschlossen werden können. Die Landmarkenkataloge für gleiche Simulationsparameter bei unterschiedlichen Strategien entsprechen sich jedoch, um die Vergleichbarkeit der Strategien untereinander zu verbessern. Es wurden somit 100 beziehungsweise 125 Flüge pro Strategie simuliert. Abbildung 9 stellt

zwei illustrative Beispiele von Terrains dar, die auf Basis unterschiedlicher Parameterkombinationen erzeugt wurden. Das Terrain stellt für das UAV kein Hindernis dar, sondern wird lediglich als Stütze für die Höhenverteilung der Landmarken im Raum genutzt. Es kann also durch das Terrain hindurchgeflogen werden.

4.1.1 Dauer der Filterkonvergenz mit GPS-Sensor

Um den Zeitpunkt der Konvergenz der geschätzten Varianz des Fehlerzustands ausgehend von den initial zu groß geschätzten Varianzen mithilfe des GPS-Sensors zu ermitteln, wurde die Simulation ohne das Abschalten des GPS-Sensors durchgeführt. Es wurde dabei sowohl die statische Strategie mit einem Winkel von 0° sowie die dynamische Strategie mit zufälliger Landmarkenauswahl genutzt, da für letztere möglicherweise eine langsamere Konvergenz auftritt als bei einer gezielten Auswahl. Dadurch wird mithilfe der zufälligen Auswahl vermutlich eine obere Grenze der Konvergenzdauer für alle dynamischen Strategien ermittelt. Für beide Simulationen wurde 200 Landmarken mit einer relativen Höhe von maximal 0.5 als Rahmenbedingungen gewählt.

In den Abbildungen 10 und 11 sind die Verläufe der geschätzten Standardabweichungen von einem Teil der Komponenten des Fehlerzustands aus der Simulation mit zufälliger Landmarkenauswahl dargestellt. Diese ergeben sich aus der geschätzten Kovarianzmatrix. Die vollständigen Daten für beide Simulationen sind im Anhang A.1 aufgeführt.

Es zeigt sich, dass die Varianz des Positionsfehlers (Abb. 10**(a)**), ebenso wie die des nicht dargestellten Lagefehlers, rasch konvergieren. Auch die Unsicherheit des geschätzten Geschwindigkeitsfehlers ändert sich nach 20 s nicht mehr signifikant (Abb. 10**(b)**).

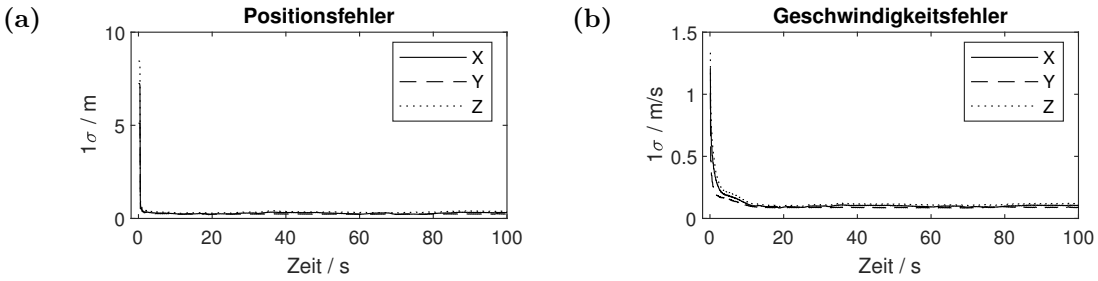

Abbildung 10: Verlauf der geschätzten Unsicherheit von **(a)** Positionsfehler und **(b)** Geschwindigkeitsfehler in einer Simulation mit zufälliger Landmarkenauswahl und 200 Landmarken sowie einer relativen Höhe von 0.5 und während der gesamten Simulation verfügbaren Messdaten des GPS-Sensors.

Die Unsicherheiten des Gravitationsmodellfehlers (Abb. 11**(b)**) und des nicht dargestellten Luftdruckmodellfehlers konvergieren ebenfalls innerhalb dieses Zeitraums. Der Bias des Gyroskops (Abb. 11**(a)**) hingegen ist erst nach etwa 45 s vollständig konvergiert. Der Bias des Beschleunigungssensors (Abb. 11**(c)**) konvergiert hingegen über den gesamten Flugzeitraum nicht zu einem annähernd konstanten Wert. Dies ist darauf zurückzuführen, dass aufgrund der geringen Anzahl an raschen Änderungen der Flugrichtung in der Trajektorie im Navigationsfilter nicht klar zwischen Abweichungen im Gravitationsmodell und im Beschleunigungssensor unterschieden werden kann.

Für den SFE des Beschleunigungssensors (Abb. 11**(d)**) sowie des nicht dargestellten Gyroskops hingegen ist die initale Schätzung bereits so gut, dass die Unsicherheit im Laufe des Fluges auf stets niedrigem Niveau ansteigt.

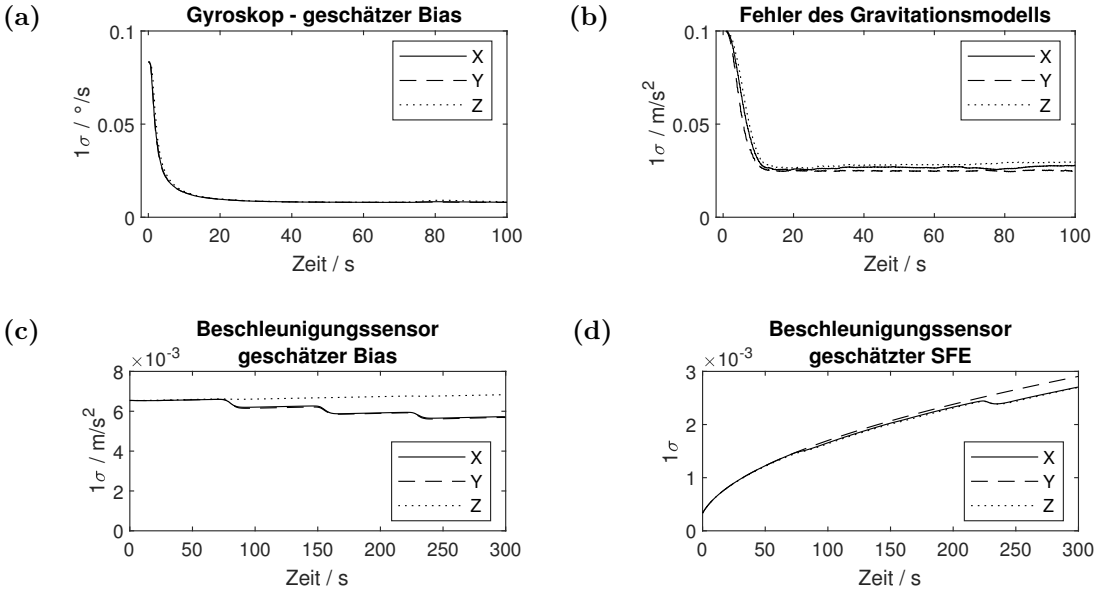

Abbildung 11: Verlauf der geschätzten Unsicherheit des **(a)** Gyroskop Bias, **(b)** Fehlers des Gravitationsmodells, **(c)** Bias und **(d)** SFEs des Beschleunigungssensors in einer Simulation mit zufälliger Landmarkenauswahl und 200 Landmarken mit maximaler, relativer Höhe von 0.5 sowie während der gesamten Simulation verfügbaren Messdaten des GPS-Sensors.

Analoge Resultate werden ebenso für die statische Strategie (s. A.1) beobachtet. Somit sind nach spätestens 45 s alle Unsicherheiten, die im Laufe der Messung konvergieren tatsächlich konvergiert.

4.1.2 Maximale Nachverfolgungsdauer

Zwei weitere Parameter müssen für die Durchführung der Simulation noch festgelegt werden. Dabei handelt es sich zum einen um die maximale Dauer, für die eine Landmarke nachverfolgt wird, bis der Auswahlprozess von neuem beginnt. Zum anderen muss eine Zeitspanne gewählt werden, die gewartet wird, bis der Auswahlprozess neugestartet wird, falls die aktuell nachverfolgte Landmarke nicht mehr im Sichtfeld der Kamera liegt beziehungsweise nicht erkannt wird. Letzteres kann in der praktischen Anwendung vorkommen, tritt jedoch in den simulierten Szenarien nicht auf.

Diese Zeitdauern beziehungsweise die aus ihnen resultierende Häufigkeit der Neuauswahl der nachzuverfolgenden Landmarke hat einen signifikanten Einfluss auf die Laufzeit der Berechnungen der PTU Steuerung. Darüber hinaus ist jedoch auch ein Einfluss auf die Performance in der Art zu erwarten, dass eine häufigere Neuauswahl mit einer idealen Strategie zu besseren Ergebnissen führt, da in diesem Szenario eine Fortsetzung der Nachverfolgung der aktuell nachverfolgten Landmarke möglich ist. Umgekehrt wird die Nachverfolgung bei langen Zeitdauern jedoch nicht abgebrochen, wenn eigentlich die Nachverfolgung einer anderen Landmarke geeigneter wäre. Entsprechend ist eine Abwägung zwischen Performance der Zustandsschätzung und Laufzeit der Berechnung notwendig.

Um eine Abschätzung der Zeitdauer zu erhalten, in der die Nachverfolgung einer gegebenen Landmarke fortgesetzt wird, bevor die Beobachtung einer anderen Landmarke besser geeignet ist, wurde die Simulation ohne Zeitbeschränkung der Nachverfolgung durchgeführt. Entsprechend wurde für jede Messung neu berechnet, welche Landmarke idealerweise nachzuverfolgen ist.

Als Szenario wurde dabei ein vollständig ebenes Terrain gewählt, auf dem 204 Landmarken auf den Eck- und Kreuzungspunkten eines quadratischen Netzes und somit äquidistant zueinander verteilt waren. Dieses Szenario wurde gewählt, da so die Auswahl der nachzuverfolgenden Landmarke nicht von einer zufälligen Verteilung abhängt, sondern in jede Blickrichtung der Kamera Landmarken sichtbar sind. Für diesen möglichst fairen Vergleich wurde ebenfalls die Größe aller Landmarken auf einen für jede Landmarke identischen Wert von 30 m gesetzt. Das so generierte Terrain wurde in einer Ellipse mit einer großen Halbachse von 500 m und einer kleinen Halbachse von 50 m in konstanter Höhe überflogen (Abb. 12). Es wurde zum einen eine Höhe von 75 m gewählt. Bei dieser niedrigen Überflughöhe war für die Mehrzahl der sichtbaren Landmarken auch eine Abstandsinformation verfügbar, da diese im Messbereich des Lidar-Sensor lagen. Zum anderen wurde ein Überflug

in einer Höhe von 275 m simuliert, sodass die Abstandsinformation in keinem Fall verfügbar war, da alle Landmarken stets außerhalb des Messbereichs des Lidar-Sensors lagen.

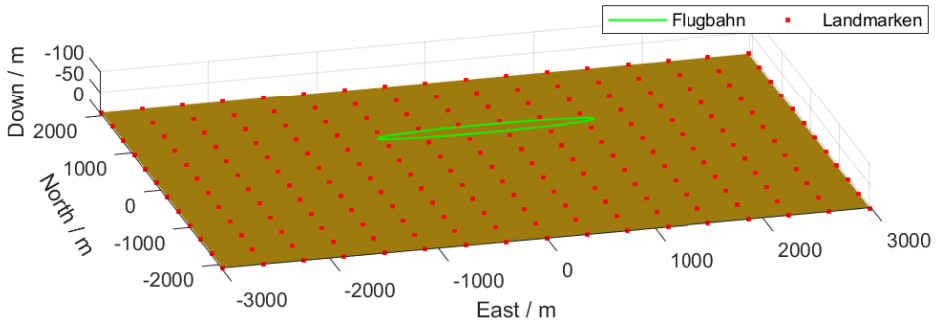

Abbildung 12: Terrain mit äquidistanter Verteilung von Landmarken und der elliptischen Flugbahn in geringerer Höhe.

Aufgrund der Analogie zur Minimierung der Zustandsvarianz durch den Kalman-Filter wurde für die Nachverfolgung die Strategie unter Minimierung der Spur der Kovarianzmatrix gewählt. Für die übrigen Strategien mit Bezug zur Kovarianzmatrix wurden jedoch vergleichbare Resultate, wie die im Folgenden beschriebenen, erhalten (s. A.2). Entsprechend werden die bestimmten Zeitdauern auch auf die übrigen Strategien übertragen, da von den Heuristiken eine möglichst gute Nachbildung der Resultate der Kovarianz-basierten Strategien erwünscht ist.

In Abbildung 13 sind die anhand dieser Simulation erhaltenen Verteilungen der Dauern, die eine Landmarke am Stück nachverfolgt wurde, dargestellt. Die geringste mögliche Zeitdauer ist die Zeit zwischen zwei Messungen und beträgt 0.2 s. Bei der niedrigeren Flughöhe wurden die Landmarken im Mittel für 0.4 ± 0.4 s nachverfolgt, was lediglich 2 aufeinanderfolgenden Messungen entspricht. Bei der höheren Flughöhe waren die Zeiten mit im Mittel 0.5 ± 0.4 s nur etwas länger. Der Unterschied ist vermutlich darauf zurückzuführen, dass durch die fehlende Abstandsinformation eine etwas höhere Messdauer erforderlich ist, um den Zustand bezüglich der aktuellen Blickrichtung zu optimieren.

Anhand dieser Resultate erscheint es zunächst sinnvoll, die Nachverfolgungsdauer kurz zu wählen und beim Verlust der Sichtbarkeit der aktuell nachverfolgten Landmarke sofort einen neuen Auswahlprozess zu starten. Dem gegenüber muss aber die erhöhte Laufzeit berücksichtigt werden (s. 4.2). Zusätzlich existieren noch Hardwarebeschränkungen, die sich ebenfalls in der Nachverfolgungsdauer wiederfinden sollten. So benötigt die PTU für eine Änderung des Blickwinkels von einer in die andere maximale Position 4.8 s (s. 2.3.1). Die Relevanz dieser endlichen Bewe-

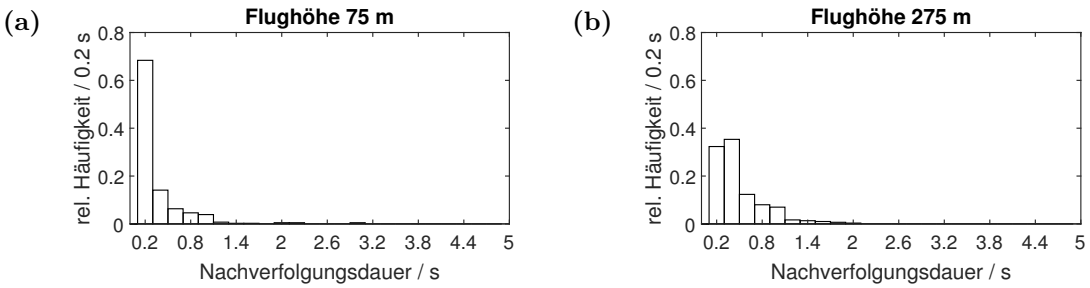

Abbildung 13: Verteilungen der Dauern bis zur Wahl einer neuen Landmarke mit der Strategie zur Minimierung von $\mathrm{Tr}(\hat{P})$ entsprechend des im Text beschriebenen Szenarios bei **(a)** geringer Flughöhe von 75 m und **(b)** hoher Flughöhe von 275 m.

gungsgeschwindigkeit wurde mit Bezug auf die Arbeiten von Davison und Murray unter 2.2.2 bereits beschrieben.[Dav98][DM98]

Auf Basis dieser drei Faktoren wurde eine maximale Nachverfolgungsdauer von 5.0 s gewählt. Damit ist sichergestellt, dass eine ausgewählte Landmarke unabhängig von der aktuellen Position der PTU zumindest für eine Messung ins Blickfeld der Kamera gerückt wird, solange sie bis zu diesem Zeitpunkt noch im Sichtbarkeitsbereich liegt. Gleichzeitig führt die gegenüber den Resultaten in Abbildung 13 erhöhte Nachverfolungsdauer zu einer Reduktion der nötigen Rechenzeit. Gleichzeitig wird die Zeit, bis zu der der Verlust der Landmarke aus dem Sichtfeld toleriert wird auf 0.1 s gesetzt. Damit wird die Landmarke nicht weiter verfolgt, wenn sie nach erfolgter Anpassung des Blickwinkels nicht sofort beobachtet wird oder während der Nachverfolgung für eine Messung nicht sichtbar ist. Damit sollen zumindest keine Messungen auf eine möglicherweise gar nicht mehr sichtbare Landmarke verwandt werden.

4.2 Laufzeiten

Um die Laufzeiten der zur Ausführung der unterschiedlichen Strategien notwendigen Berechnungen zu vergleichen, wurden die Ausführungszeiten der verschiedenen Simulationsläufe ausgewertet. Die Simulation wurde auf einem Notebook mit einer Intel Core i7 6600U CPU der 6. Generation mit 2 Kernen und einer Basistaktfrequenz von 2.6 GHz ausgeführt. Die Rechenleistung dieser CPU ist ungefähr mit der des Flugrechners der superArtis Hubschraubers vergleichbar (s. 2.3.1).

Ein Unterschied in der Ausführungszeit der Simulation im Vergleich zur Ausführung der Navigationssoftware im tatsächlichen Flug liegt darin, dass in der Simulation auch die Sensordaten synthetisiert werden müssen. Dieser Anteil der Laufzeit sollte im Wesentlichen von der Anzahl der Landmarken des Katalogs abhängen, da dieser

für die Simulation der Kameradaten in jedem Zeitschritt vollständig berücksichtigt wird (s. 2.3.3). Für die übrigen Sensordaten ist keine wesentliche Veränderung durch die Variation der Parameter zu erwarten. In Tabelle 8 sind typische Laufzeiten für diesen Anteil der Simulation in Abhängigkeit von der Landmarkenanzahl dargestellt. Diese wurden mit dem *Profiler* von Matlab ermittelt. Es zeigt sich der erwartete Trend einer signifikanten Zunahme mit steigender Anzahl der Landmarken. Die absolute Laufzeit ist gegenüber der Gesamtlaufzeit der Simulation selbst für die größte Anzahl der Landmarken vernachlässigbar.

Anzahl Landmarken	25	50	100	200
Laufzeit / s	0.16	0.11	0.86	1.65

Tabelle 8: Typische Laufzeiten für die Synthese der Sensordaten in einem Simulationslauf in Abhängigkeit von der Anzahl der Landmarken im Katalog. Die Laufzeiten wurden für die Strategie zur Minimierung der Spur der Kovarianzmatrix und einer relativen Höhe von 0.5 bestimmt, wobei der GPS-Sensor in den ersten 45 s der Simulation aktiv war.

Der Rest der Gesamtlaufzeit setzt sich hauptsächlich aus der Laufzeit des Navigationsfilters und damit der Sensorfusion durch den UKF sowie der Laufzeit der PTU Steuerung zusammen. Die Laufzeit des Navigationsfilters ist im Wesentlichen konstant, da die Häufigkeit der Ausführung in der Simulation für alle Simulationsläufe identisch ist und bezüglich der Dimensionalität der Eingabe lediglich die Anzahl der Landmarken im Sichtfeld der Kamera variiert. Hierbei ist jedoch zu erwarten, dass diese stets klein gegenüber der Größe des Landmarkenkatalogs ist und als annäherend konstant angenommen werden kann. Dies zeigen auch die exemplarischen Laufzeiten von $\approx 400\,\mathrm{s}$ des Navigationsfilters in Abhängigkeit von der Anzahl der Landmarken im Katalog in Tabelle 9. Die Ausführungszeiten des Navigationsfilters in der Größenordnung der Flugzeit legen bereits nahe, dass die zusätzlich benötigte Rechenzeit der PTU Steuerung einen relevanten Einflussfaktor für die Echtzeitfähigkeit des Gesamtsystems zur Flugsteuerung darstellt.

Da für die statischen Strategien keine Auswahl einer Landmarke stattfindet, bilden die mit diesen Strategien ermittelten mittleren Gesamtlaufzeit der Simulationsausführung einen geeigneten Ausgangspunkt zur Ermittlung der reinen Laufzeit, die für die Auswahl der Landmarke bei gegebener dynamischer Strategie in der PTU Steuerung erforderlich ist. Tabelle 10 stellt diesen, aus den mittleren Laufzeiten der statischen Strategien ermittelten Overhead sowie die durch die Korrektur mit dem Overhead ermittelten Laufzeiten für die Landmarkenauswahl mit den verschiede-

Anzahl Landmarken	25	50	100	200
Laufzeit / s	398	397	384	406

Tabelle 9: Typische Laufzeiten des Navigationsfilters in einem Simulationslauf in Abhängigkeit von der Anzahl der Landmarken im Katalog. Die Laufzeiten wurden für die Strategie zur Minimierung der Spur der Kovarianzmatrix und einer relativen Höhe von 0.5 bestimmt, wobei der GPS-Sensor in den ersten 45 s der Simulation aktiv war.

nen dynamischen Strategien gegenüber. Hierbei wurden die Mittelwerte über alle Simulationsläufe pro Landmarkenanzahl gebildet, da diese auch hier den einzigen signifikanten Einflussfaktor darstellt.

Strategie	Overhead	5	6	7	8	9	10
25 Landmarken	443 s	0 s	2 s	0 s	0 s	6 s	0 s
50 Landmarken	451 s	0 s	−3 s	−4 s	5 s	5 s	6 s
100 Landmarken	442 s	10 s	1 s	10 s	**21 s**	**21 s**	**21 s**
200 Landmarken	447 s	−7 s	2 s	−2 s	**84 s**	**86 s**	**83 s**

Tabelle 10: Mittlere Laufzeiten der Berechnungen zur Auswahl der nachzuverfolgenden Landmarke in Abhängigkeit von der gewählten dynamischen Strategie und der Anzahl der Landmarken im Katalog. Von der Gesamtlaufzeit der Simulation wurden der anhand der statischen Strategien ermittelte Overhead subtrahiert.

Es zeigt sich eine Unsicherheit von etwa ±10 s, was zur Streuung des Overheads, der nahezu unabhängig von der Anzahl der Landmarken sein sollte, und auch zu den negativen korrigierten Zeiten in Tabelle 10 korrespondiert. Diese Unsicherheit entspricht auch der typischen Standardabweichung der dargestellten Mittelwerte. Die Abweichungen sind vermutlich durch die unterschiedlichen Systemauslastungen des verwendeten PCs durch andere Prozesse während der Ausführung der Simulationsläufe zu erklären.

Im Rahmen dieser Messgenauigkeit zeigt sich nur für die Strategien 8–10, die auf der Vorhersage und Auswertung der Kovarianzmatrix basieren, eine signifikante Erhöhung der Laufzeit. Diese Erhöhung ist darüber hinaus auch signifikant und überlinear von der Anzahl der Landmarken abhängig. Diese stärker als lineare Abhängigkeit konnte insbesondere auch bei den Simulationen mit 400 Landmarken im Katalog (s. 4.4) beobachtet werden, wobei Laufzeiten von über 7 h aufgetreten sind.

Dem gegenüber führt der Einsatz der Heuristiken der Strategien 5 – 7 nicht zu einer signifikanten Erhöhung der Laufzeit. Auch ist keine klare Abhängigkeit von der Größe des Landmarkenkatalogs zu beobachten. Dies stellt bereits einen klaren Vorteil der Heuristiken gegenüber den Kovarianzmatrix-basierten Strategien dar und ist gleichzeitig die Hauptmotivation für deren Entwicklung, da so praktisch unabhängig von der Größe des Landmarkenkatalogs und ohne weitere Optimierung der Strategie die Echtzeitanforderungen für die Flugsteuerung erfüllt werden könnten.

4.3 Performance nach Verlust des GPS Signals

In einer ersten Parameterstudie wurden die verschiedenen Strategien zur Landmarkenauswahl sowie die Anzahl der Landmarken im Katalog und die relative Höhe der Landmarken im Raum variiert (s. 4.1). Es wurden Positions- und Geschwindigkeitsmessungen des GPS Sensors für die ersten 45 s jedes 300 s langen Fluges berücksichtigt und danach ein Verlust des GPS Signals simuliert, sodass die Unsicherheit des Navigationsfilters vor dem Verlust des GPS Signals konvergiert war (s. 4.1.1). Ab diesem Zeitpunkt und bis zum Ende des Fluges basiert die Zustandsschätzung nur noch auf den Daten von Beschleunigungssensor und Gyroskop der IMU sowie Barometer, Magnetometer und der optischen Navigation mit Kamera und Lidar.

Um nun eine erste Bewertung der Performance der verschiedenen Strategien zu erhalten, wurden die wahre und die geschätzte Position des UAVs entlang der Trajektorie ausgewertet. Wie die Beispiele in Abbildung 14 zeigen, traten hierbei sowohl Fälle auf, in denen die Position entlang der gesamten Trajektorie mit kaum sichtbaren Abweichungen zum wahren Wert geschätzt wurde als auch Fälle, in denen es im Laufe des Fluges zu immer stärkeren Abweichungen zwischen der tatsächlichen und geschätzten Position kam, wobei diese zunehmenden Abweichungen stets nach dem Verlust des GPS Signals auftraten.

Es wurde in den meisten Fällen beobachtet, dass diese stärkeren Abweichungen sobald, sie aufgetreten waren, mit fortschreitender Zeit bis zum Ende der Simulation immer weiter zunahmen. Darauf basierend wurde der Abstand zwischen tatsächlicher und geschätzter Position am Ende der Simulation genutzt, um die Simulationsläufe entweder als Fehlschlag oder Erfolg zu klassifizieren. Abbildung 15 zeigt ein für alle Simulationsläufe unabhängig von der Parameterkombinationen kombiniertes Histogramm dieser Abstände, anhand dessen die Grenze zwischen Erfolg und Fehlschlag für die binäre Klassifikation abgeschätzt werden kann. Der Abstand liegt bei 70% aller Simulationsläufe bei 50 m oder weniger. Die übrigen

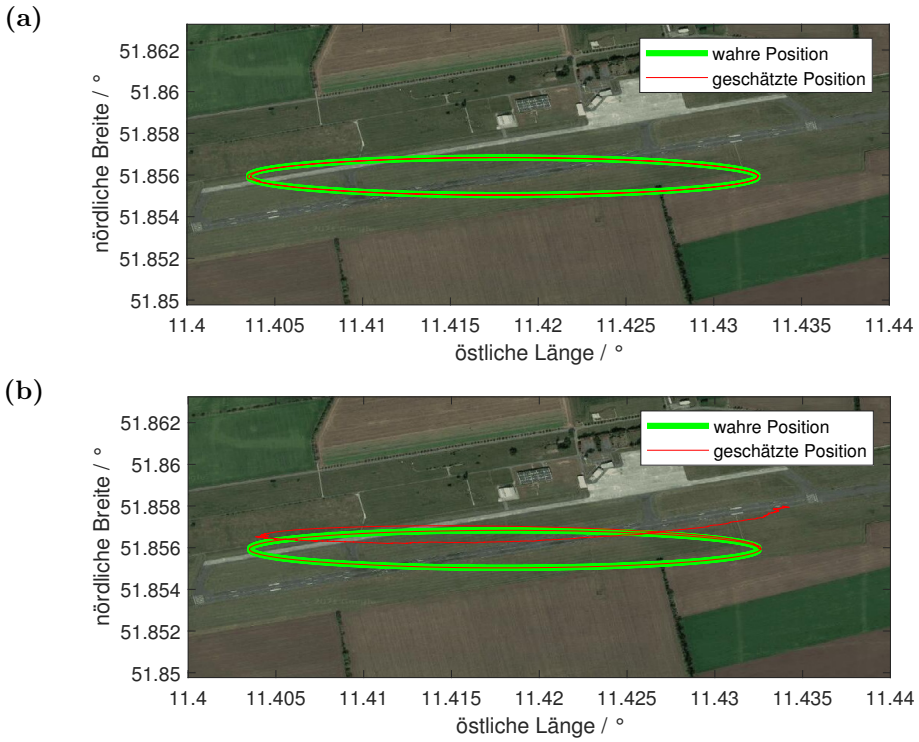

Abbildung 14: Aufsicht auf die Karte des Flughafens Cochstedt (aus Google Maps) mit der tatsächlichen und geschätzten Positionen aus der Simulation. Während des gesamten Fluges **(a)** tritt nur eine geringe Abweichen zwischen beiden Trajektorien auf, während die Abweichungen für **(b)** kontinuierlich zunehmen. Beide Beispiele stammen aus unterschiedlichen Simulationsläufen mit jeweils zufälliger Landmarkenauswahl, 25 Landmarken im Katalog und einer relativen Höhe von 0.0.

30% der Werte sind über einen großen Wertebereich verstreut und es gibt keine besonders auffälligen Häufungen. Dem entsprechend wurde der Wert von 50 m als Grenze zur Unterscheidung von Erfolgen und Fehlschlägen gewählt. Dies ist auch deshalb sinnvoll, da der größte initiale Positionsfehler bei etwa 50 m liegt und damit nach dieser Klassifikation ein Fehlschlag vorliegt, wenn der initiale Positionsfehler am Ende der Simulation wieder überschritten wurde.

Mithilfe des Grenzwerts wurden nun die jeweils 5 Monte-Carlo-Läufe pro Parameterkombination klassifiziert. Den Anteil der Fehlschläge in Abhängigkeit von relativer Höhe und Landmarkenanzahl für jede Strategie zeigt Abbildung 16. Anhand dieser Auswertung werden bereits mehrere Trends deutlich. Zunächst unterscheiden sich die statischen und dynamischen Strategien signifikant voneinander. Während für die dynamischen Strategien Fehlschläge die Ausnahme darstellen, sind diese für die statischen Strategien die Regel. Alle Strategien haben jedoch gemein, dass die Simulation mit einem Katalogumfang von 25 Landmarken die meisten Fehl-

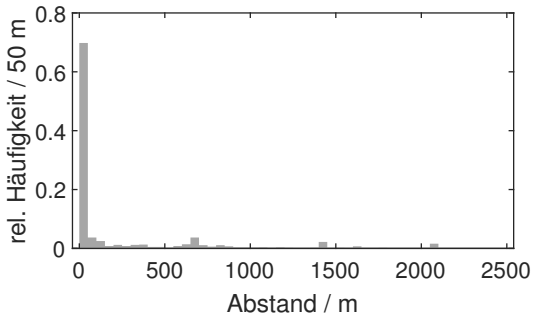

Abbildung 15: Kombinierte Verteilung der Abstände zwischen tatsächlicher und geschätzter Position am Ende des Fluges für alle Simulationsläufe.

schläge aufweisen. Dies lässt darauf schließen, dass entlang der Trajektorie in diesen Fällen häufig schlicht zu wenige oder sogar keine Landmarken für eine geeignete Zustandsschätzung im Sichtbereich der Kamera mit und ohne active vision liegen. Des Weiteren sind für die statischen Strategien deutliche Unterschiede zwischen den verschiedenen Blickwinkeln erkennbar, während der Anteil und die Parameterabhängigkeit der Fehlschläge für die dynamischen Strategien nahezu identisch ist. Bei den statischen Strategien weist der Blick nach vorne (0°) insgesamt weniger Fehlschläge auf als die übrigen statischen Blickwinkel, bei denen die Kamera mehr nach unten gerichtet ist. Dazu wirkt sich eine steigende relative Höhe negativ auf die letztgenannten Strategien aus. Dies ergibt sich direkt daraus, dass höher liegende Landmarken bei einem Blick nach unten in keinem Fall ins Blickfeld gelangen und somit für die Navigation nicht genutzt werden können. Aber auch beim Blick nach vorne werden mehr Landmarken benötigt als bei jeder dynamischen Strategie, bis keine Fehlschläge mehr auftreten. Hieran ist die Erweiterung des zugänglichen FOVs durch active vision im Allgemeinen bereits als vorteilhaft zu erkennen.

Um einen detaillierteren Einblick in die Performance der verschiedenen Strategien zu gewinnen und insbesondere auch Unterschiede zwischen den verschiedenen dynamischen Strategien herauszuarbeiten, wurden die Abstände zwischen geschätzter und tatsächlicher Position, Geschwindigkeit und Lage untersucht. Dabei wurden nicht nur die Endpunkte der Simulationsläufe, sondern alle simulierten Zeitschritte (10 Hz) betrachtet. Da in den ersten 45 s die Performance maßgeblich von den Messdaten des GPS-Sensors abhängt, wurden nur die Resultate der Simulation im Zeitbereich von 100 – 300 s für die folgenden Analysen berücksichtigt. Dadurch wurde nach dem Verlust des GPS Signals nochmals gewartet, um einen ähnlichen Effekt wie bei der initialen Filterkonvergenz für die neue Sensorsituation zu erreichen. Es wird davon ausgegangen, dass die Schätzungen ab dem Zeitpunkt von 100 s die Per-

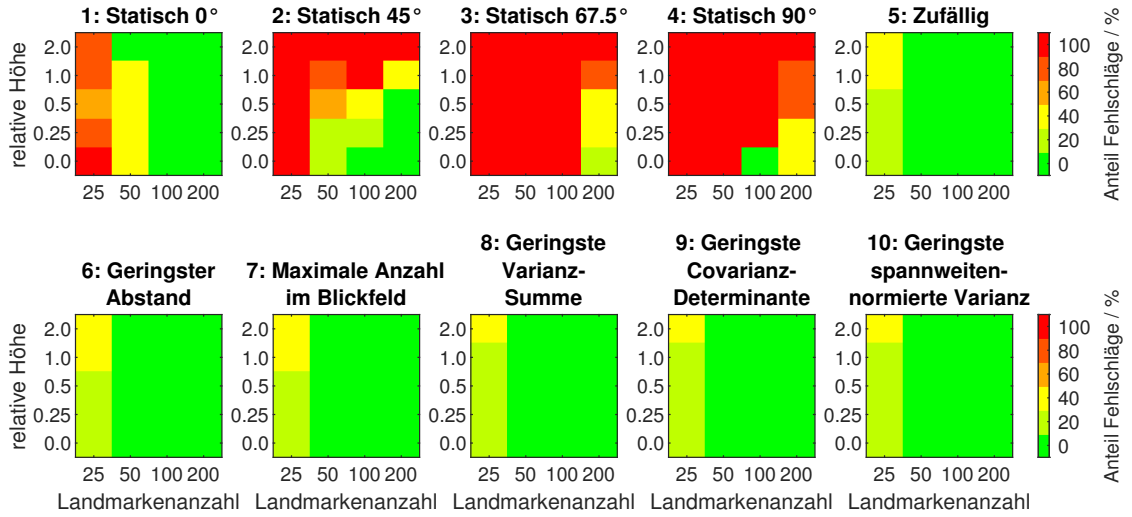

Abbildung 16: Anteil der Simulationsläufe mit Abständen von mehr als 50 m zwischen tatsächlicher und geschätzter Endposition in Abhängigkeit von den Simulationsparametern.

formance des ausreichend initialisierten Systems ohne Einfluss der GPS Messungen repräsentieren.

In Abbildung 17 sind die über alle Simulationsläufe einer Strategie, jeweils über das Intervall von 100 – 300 s, gemittelten Abweichungen zwischen geschätzter und tatsächlicher Position, Geschwindigkeit und Lage dargestellt. Hierbei bestätigt sich das bereits anhand der binären Klassifikation erhaltenen Bild, nach dem mithilfe von active vision deutlich bessere Schätzungen des Zustands möglich sind als mit den statischen Strategien. Bei letzteren sind für die simulierten Szenarien auch bei dieser Betrachtung für Position, Geschwindigkeit und Lage ein Blick nach vorne besser für die Zustandsschätzung geeignet als ein Blick nach unten.

Im Unterschied zur binären Klassifikation kann bei dieser Betrachtung über die gesamte Simulationszeit auch ein Unterschied zwischen den dynamischen Strategien festgestellt werden. Wie der Vergleich der mittleren Fehler in Abbildung 17**(b)**, 17**(d)** beziehungsweise 17**(f)** zeigt, werden die Größen des Zustands durch die unterschiedlichen Strategien unterschiedlich genau geschätzt.

Die Position wird durch Strategie 8, also durch die Minimierung der Spur der Kovarianzmatrix, mit dem geringsten Fehler geschätzt. Nur wenig schlechter ist die Schätzung durch die Heuristik der Maximierung der Anzahl gleichzeitig sichtbarer Landmarken (Strategie 7). Im Mittel etwas größere Positionsfehler werden durch die zufällige Auswahl der Landmarken (Strategie 5) und die Minimierung der Determinante der Kovarianzmatrix (Strategie 9) erhalten. Die größten mittleren Positionsfehler unter den dynamischen Strategien werden durch die Auswahl der Landmarke

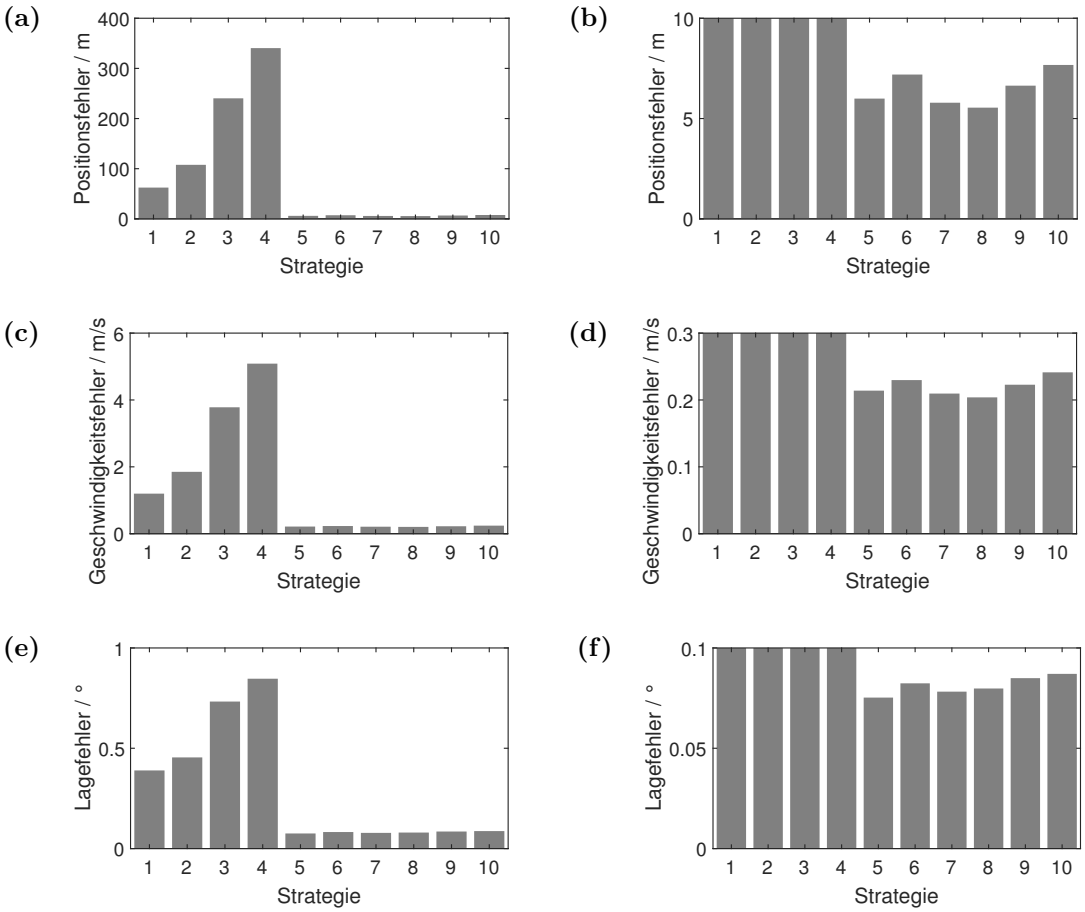

Abbildung 17: Mittlere Abstände zwischen geschätzter und tatsächlicher **(a)**, **(b)** Position, **(c)**, **(d)** Geschwindigkeit und **(e)**, **(f)** Lage. Es ist jeweils eine vollständige Übersicht (**(a)**, **(c)** und **(e)**) sowie eine Reskalierung mit Fokus auf die dynamischen Strategien (**(b)**, **(d)** und **(f)**) dargestellt. Der aufgetragene Mittelwert wurde aus allen Messpunkten im Zeitbereich von 100 – 300 s über alle Simulationsläufe jeder Strategie gebildet.

mit geringstem Abstand zum UAV (Strategie 6) sowie Strategie 10, die Minimierung der spannweiten-normierten Varianz, erzeugt. Die Qualität der Schätzung der Geschwindigkeit weist die selbe Rangfolge auf, wie die der Positionsschätzung.

Interessant ist die Änderung der Rangfolge für die Schätzung der Lage des UAVs, auch wenn die Unterschiede zwischen den Strategien insgesamt geringer sind. Von der Schätzung der Lage wird zunächst erwartet, dass diese einen geeigneten Indikator für die Schätzung der übrigen Größen des Zustands darstellt. Dies ist darauf zurückzuführen, dass sich ein Fehler in der Lageschätzung signifikant auf die übrigen Zustandsgrößen auswirkt. Ist zum Beispiel die Geschwindigkeit fehlerfrei bekannt, wird auf Basis einer falschen Lageschätzung trotzdem ein signifikanter Fehler der Positionsschätzung resultieren, wenn beide Werte zur Propagation des Zustands

herangezogen werden, wie es durch den Navigationsfilter der Fall ist (s. 2.3.2.1). Die Rangfolge der dynamischen Strategien bezüglich des Lagefehlers von geringsten zum größten Fehler ist zufällige Auswahl, größte Anzahl gleichzeitig sichtbarer Landmarken, minimale Spur der Kovarianzmatrix, geringster Abstand des UAVs zur Landmarke, minimale Determinante der Kovarianzmatrix und minimale spannweitennormierte Varianz.

Insgesamt ist es zunächst überraschend, dass Position und Geschwindigkeit von anderen Strategien besser geschätzt werden, obwohl hier der Lagefehler etwas größer ist. Da abgesehen von der zufälligen Auswahl auch für die Lage die Strategien 7 und 8 die beste Performance aufweisen, scheinen beide im Mittel über alle Simulationsparameter eine ausgewogenere Schätzung bezüglich aller Zustandsgrößen zu liefern.

Um etwaige Trends bei der Performance der unterschiedlichen Strategien in Abhängigkeit von den variierten Simulationsparametern zu untersuchen, wurden die mittleren Fehler auch für die einzelnen Parameterkombinationen separat und nicht über alle Parameter gemittelt ausgewertet. Abbildung 18 zeigt eine Auftragung dieser Abhängigkeiten bezüglich des Positionsfehlers, Abbildung 19 bezüglich des Geschwindigkeits- und Abbildung 20 bezüglich des Lagefehlers. Für die statischen Strategien zeigen sich für alle Fehler im Wesentlich die Trends, die bereits durch die binäre Klassifikation (Abb. 16) sichtbar geworden sind.

Für die dynamischen Strategien zeichnet sich eine deutliche Abnahme aller drei Fehler mit zunehmender Anzahl von Landmarken im Katalog ab. Dieser Trend ist für alle dynamischen Strategien in ähnlichem Ausmaße vorhanden. Hieran wird deutlich, dass eine größere Auswahl von Landmarken und damit potentiellen Blickrichtungen in allen Fällen positiv für die Performance ist, selbst wenn die Auswahl zufällig erfolgt. Auch zeigt sich im Vergleich mit der statischen Strategie mit Blickrichtung nach vorne, dass sich die Erweiterung des FOV selbst dann noch signifikant positiv auf die Performance auswirkt, wenn stets viele Landmarken bei einem statischen Blickwinkel sichtbar sind. Dieser Trend lässt die Vermutung zu, dass die Verwendung von active vision, unabhängig von der gewählten Strategie auch geeignet wäre, Zustandsschätzungen bei einer SLAM-Navigation signifikant zu verbessern.

Ebenfalls ist für alle drei Fehlertypen ein schwacher Trend bezüglich der relativen Höhe zu erkennen, insbesondere wenn eine größere Anzahl an Landmarken (≥ 100) im Katalog vorhanden ist. Mit steigender relativer Höhe werden die Fehler kleiner, obwohl die Dichte der Landmarken im betrachteten Raum durch die steigende relative Höhe abnimmt. Damit ist zunächst zu erwarten, dass im Mittel weniger Landmarken gleichzeitig beobachtet werden können. Es ist somit davon auszugehen, dass die Information aus den verschiedenen Raumrichtungen wichtiger für die

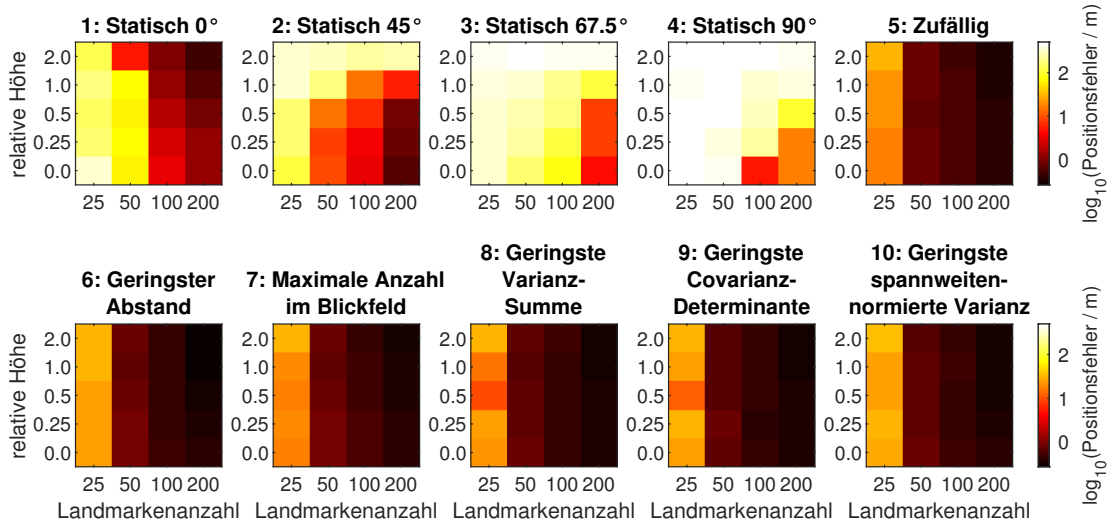

Abbildung 18: Abstand zwischen geschätzter und tatsächlicher Position in Abhängigkeit von den Simulationsparametern. Der Mittelwert wurde aus allen Messpunkten im Zeitbereich von 100 – 300 s der jeweils 5 Monte-Carlo-Läufe pro Parameterkonfiguration gebildet.

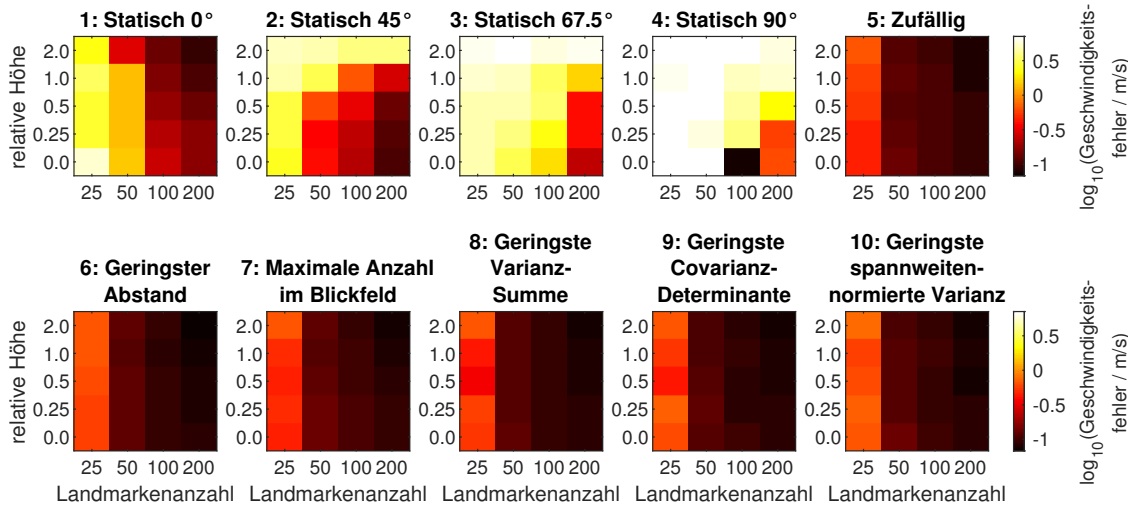

Abbildung 19: Abstand zwischen geschätzter und tatsächlicher Geschwindigkeit in Abhängigkeit von den Simulationsparametern. Der Mittelwert wurde aus allen Messpunkten im Zeitbereich von 100 – 300 s der jeweils 5 Monte-Carlo-Läufe pro Parameterkonfiguration gebildet.

Zustandsschätzung ist, als eine höhere Informationsdichte bezüglich einer stärker eingeschränkten Raumrichtung.

Um den Einfluss der relativen Höhe genauer zu analysieren, wurden die Positions-, Geschwindigkeits- und Lagefehler in NED-Koordinatensystem pro Raumrichtung für die Simulation mit einem Katalog von 200 Landmarken bei jeder relativen Höhe über

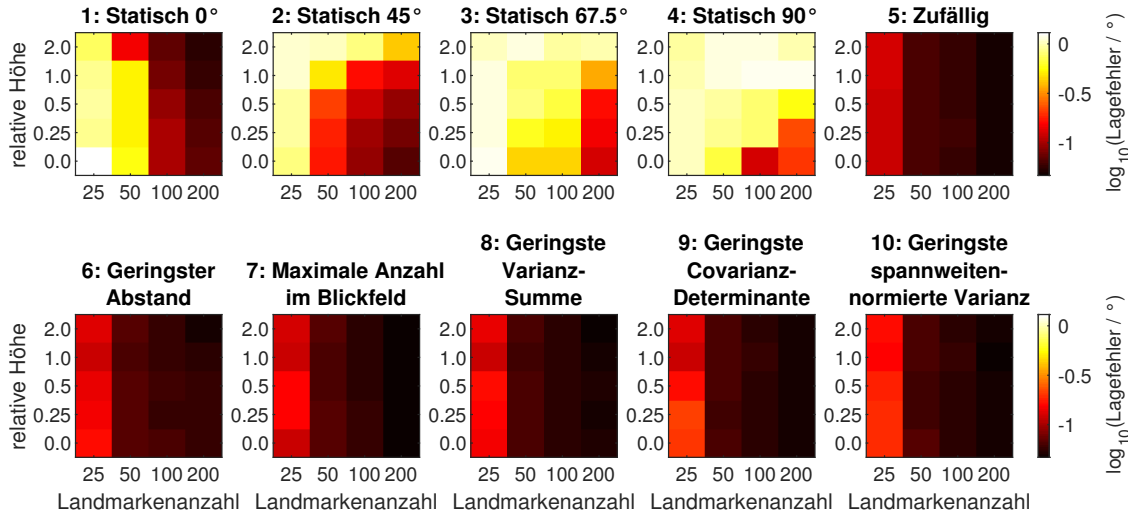

Abbildung 20: Drehwinkel zwischen geschätzter und tatsächlicher Lage in Abhängigkeit von den Simulationsparametern. Der Mittelwert wurde aus allen Messpunkten im Zeitbereich von 100 – 300 s der jeweils 5 Monte-Carlo-Läufe pro Parameterkonfiguration gebildet.

den Simulationszeitraum von 100 – 300 s ermittelt und daraus Boxplots der Fehlerverteilungen bestimmt. Abbildung 21 zeigt die jeweils fünf so erhaltenen Boxplots pro Raumrichtung exemplarisch für den Positionsfehler und Strategie 8. Die entsprechenden Abbildungen für Geschwindigkeits- und Lagefehler finden sich im Anhang A.3. Hierbei zeigt sich, dass durch eine weitere Verteilung der Landmarken entlang der Höhenachse die Fehler in allen Raumrichtungen abnehmen. Die zusätzliche Information wirkt sich somit nicht nur singulär auf die Schätzung entlang der Höhenachse aus, sondern ist positiv für die Schätzung bezüglich jeder Richtung. Dieses Verhalten ist prototypisch für alle Strategien und Fehler, die hier nicht dargestellt sind.

Da ein Vergleich der Details der unterschiedlichen dynamischen Strategien anhand der Abbildungen 18 bis 20 visuell schwer zu überblicken ist, wurden die Strategien in den Abbildung 22 für die mittleren Positionsfehler pro Parameterkonfiguration nochmals paarweise gegenübergestellt. Identische Gegenüberstellungen für die Geschwindigkeits- und Lagefehler sind im Anhang A.4.1 zu finden.

Abbildung 22 zeigt eine Auftragung für jede Kombination von 2 Strategien. So wird zum Beispiel in der Kachel ganz unten links Strategie 5 als Strategie A mit Strategie 10 als Strategie B verglichen. Für jede Parameterkombination wird dabei der mittlere Positionsfehler der jeweiligen Strategie nach Abbildung 18 berücksichtigt. Wichtig ist dabei hervorzuheben, dass für beide Strategien exakt identische Szenarien in der Simulation vorlagen und sich tatsächlich nur die Strategie zur Landmarkenauswahl unterscheidet. Aus den mittleren Fehlern pro Parameterkombination

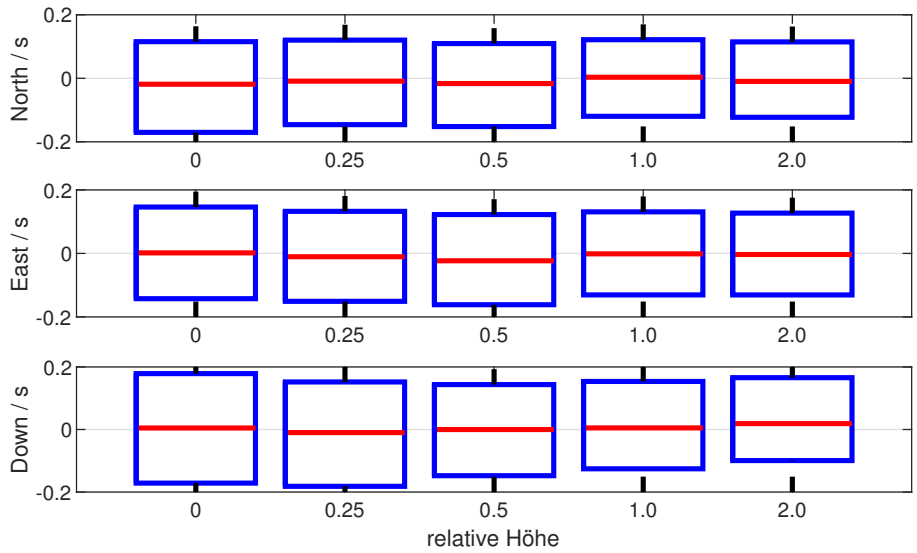

Abbildung 21: Boxplots der Positionsfehler im Zeitbereich 100 – 300 s für die Simulationsläufe mit 200 Landmarken am Beispiel der Strategie zur Minimierung der Spur der Kovarianzmatrix in Abhängigkeit von der relativen Höhe. Die Positionsfehler sind pro Koordinatenachse im NED-Koordinatensystem aufgetragen. Die roten Markierungen zeigen den Mittelwert die blauen Boxen die Standardabweichung an.

wird die Differenz gebildet. Für eine bessere Skalierung wurde von der Differenzbildung der Logarithmus der einzelnen mittleren Fehlerwerte zur Basis 10 gebildet. Die Differenz wird jeweils so gebildet, dass sie negativ ist, wenn Strategie B (auf der y-Achse) den besseren beziehungsweise geringeren Fehlerwert liefert und positiv wenn Strategie A (auf der x-Achse) den besseren Fehlerwert liefert. Ersteres ist in der Abbildung in Blautönen, letzteres in Gelb- und Rottönen kodiert. Grüne Farbwerte zeigen nahezu identische Fehlerwerte an.

Um einen übersichtlicheren Vergleich trotz unterschiedlicher Größenordnungen der Fehler für die unterschiedlichen Simulationsparameter zu erreichen, wurden die Fehlerdifferenzen pro Parameterkombination separat auf das Intervall $[-1, 1]$ skaliert. Damit wird für jede Parameterkombination in einem der Strategievergleiche der Maximalwert- und in einer der Minimalwert dieser Skala erreicht. Die jeweils beste und schlechteste Strategie bezüglich einer Parameterkombination wird auf Basis dieses Maximal- und Minimalwerts auf der entsprechenden Kachel auf der Diagonalen markiert, die den Vergleich einer Strategie mit sich selbst darstellt. Eine rote nach oben gerichtete Pfeilspitze zeigt die beste Strategie für eine Parameterkombination an, eine blaue nach unten gerichtete Pfeilspitze die schlechteste. Die detaillierte

Rangfolge der Strategien und ein Trend für die Stärke der Abweichungen zwischen ihnen ergibt sich dann aus der Farbkodierung der übrigen Kacheln.

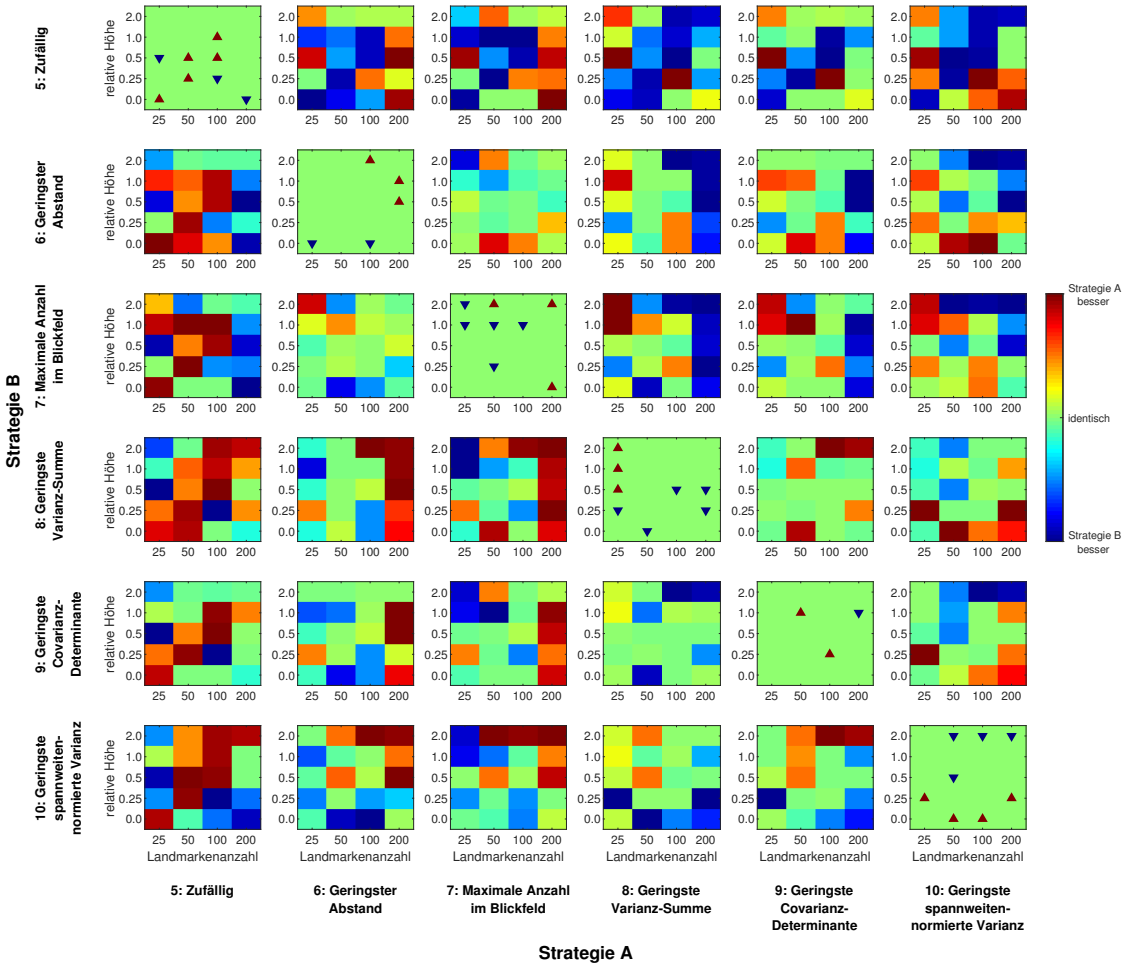

Abbildung 22: Pro Parameterkombination normierte Unterschiede der mittleren Positionsfehler mit den verschiedenen Strategien. Die Farben zeigen im Vergleich bessere oder schlechtere Performance der Strategien an. Auf der Diagonalen sind die Parameterkombinationen markiert, für die eine dynamische Strategie im Vergleich zu allen anderen am besten (rote Pfeilspitze nach oben) oder am schlechtesten (blaue Pfeilspitze nach unten) bezüglich der Minimierung der Fehler arbeitet.

Insgesamt zeigt sich kein eindeutiger Trend und auch keine in der Mehrzahl der Situationen überlegene Strategie. Für alle Strategien gibt es zumindest eine Parameterkombination für die diese den geringsten mittleren Fehler aufweist und auch zumindest eine für die der Fehler im Vergleich maximal ist. Strategie 10 zur Minimierung der spannweiten-normierten Varianz zeigt eine Häufung guter Performances für die kleineren Werte der relativen Höhe (≤ 0.5). Strategie 8, die die Minimierung der Spur der Kovarianzmatrix zum Ziel hat, zeigt dagegen für die kleinste Landmarkenanzahlen die geringsten Fehler, für größere Kataloge hinge-

gen im Vergleich deutlich reduzierte Performance. Für den Landmarkenkatalog mit dem größten Umfang von 200 zeigen die beiden Heuristiken der Strategien 6 und 7 hingegen überdurchschnittliche Performance. Jedoch ist in einigen Fällen auch die zufällige Landmarkenauswahl überlegen, wobei sich hier kein klarer Cluster zeigt.

Das Fehlen von klaren Trends ist möglicherweise darauf zurückzuführen, dass mit 5 Simulationsläufen pro Parameterkombination ein zu starker Einfluss der gewählten Zufallszahlen vorliegt und die der Auswertung zu Grunde liegende Statistik nicht ausreicht, um solche Trends zu erkennen. Weiterheit deutet darauf hin, dass die beste und schlechteste Performance teilweise für Parameterkombinationen vorliegen, bei denen nur ein Parameter um eine Stufe verändert wurde.

Sollten sich die beschriebenen schwachen Trends bei einer besseren Datengrundlage jedoch erhärten, so muss die Strategie in Abhängigkeit vom erwarteten Einsatzszenario gewählt werden. Für eine reine Landmarken-Navigation ist eine geringe Anzahl von gleichzeitig sichtbaren Landmarken ein wahrscheinliches Szenario, sodass Strategie 8 einen geeigneten Kandidaten darstellt. Ist hingegen ein hoher Flug über eine eher flache Szenerie zu erwarten, ist möglicherweise Strategie 10 besser geeignet. Strategie 7 und gegebenenfalls auch 6 könnten hingegen für eine SLAM-Anwendung interessant sein, da stets Feature entdeckt werden können, sodass ein umfangreicher Katalog zur Verfügung steht.

Hervorzuheben ist zusätzlich, dass die naive Erwartung, dass Strategie 8 stets den geringsten Fehler liefert, da die gleiche Minimierung stattfindet wie im Rahmen des UKF, offensichtlich nicht erfüllt wird. Dies ist darauf zurückzuführen, dass sowohl vom UKF als auch von der Strategie neben den Größen des Flugzustands auch die Sensor- und Modellfehler berücksichtigt werden, die im Rahmen dieser Arbeit aufgrund der hohen Komplexität der Abhängigkeiten jedoch nicht detailliert analysiert wurden. Darüber hinaus wird die zu erwartende Kovarianzmatrix berechnet, sodass hierbei je nach aktuellem Fehler des Zustands auch signifikante Abweichungen zur tatsächlich erhaltenen Kovarianzmatrix durch die Messung auftreten können. Die Strategie ist somit auf einen bereits geringen beziehungsweise gut geschätzten Fehlerzustand angewiesen, um zuverlässige Vorhersagen treffen zu können. Gleiches gilt für die beiden anderen auf der erwarteten Kovarianzmatrix basierenden Strategien.

Bezüglich Position und Geschwindigkeit sind die Vergleiche der Strategien weitestgehend identisch. Beim Lagefehlers gibt es etwas größere Abweichungen (s. A.4.1), wobei die bereits beschriebenen Trends im Wesentlichen erhalten bleiben.

Um auszuschließen, dass die ähnliche Performance der dynamischen Strategien lediglich darauf zurückzuführen ist, dass die simulierten Szenarien zu wenig unter-

schiedliche Wahlmöglichkeiten für die nachzuverfolgende Landmarke bieten, wurden die zu jedem Zeitpunkt der Simulation sichtbaren Landmarken verglichen.

Abbildung 23 zeigt dazu den mittleren Jaccard-Koeffizienten $Jac(M_1, M_2)$ der sich aus den Mengen M_1 und M_2 als

$$Jac(M_1, M_2) = \frac{\|M_1 \cap M_2\|}{\|M_1 \cup M_2\|} \tag{59}$$

ergibt. Dabei wurden alle Simulationsläufe einer Strategie jeweils über den gesamten Zeitbereich berücksichtigt und zu jedem Zeitpunkt der Jaccard-Koeffizient aus den beiden Mengen der sichtbaren Landmarken gebildet. Diese Jaccard-Koeffizienten wurden dann über alle Zeitpunkte und Simulationsläufe gemittelt. Anschaulich gibt ein hoher Koeffizient nahe 1 an, dass in vielen Fällen bei beiden Strategien die gleichen Landmarken im FOV der Kamera sichtbar waren. Für die statischen Strategien reflektieren die Jaccard-Koeffizienten letztlich den Überlapp des FOVs der Kamera bei den unterschiedlichen statischen Winkeln. Bei den dynamischen Strategien gibt es eine gewisse Überlappung der Mengen der beobachteten Landmarken, jedoch liegen die mittleren Werte deutlich unterhalb von 1 und es liegt somit keine dominierende Überlappung vor. Der statische Blickwinkel von 0° weist dabei noch die größte Ähnlichkeit zu den dynamischen Strategien auf, sodass davon auszugehen ist, dass von diesen häufiger nach vorne als nach unten geblickt wird.

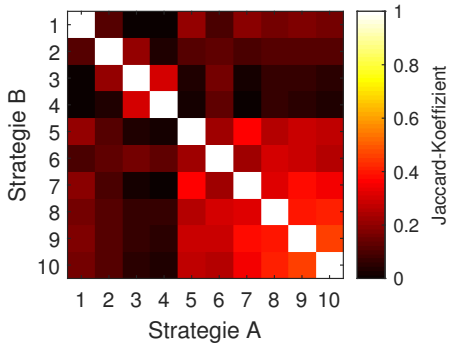

Abbildung 23: Mittlere Überlagerung der Menge sichtbarer Landmarken zwischen je zwei Strategien anhand der Jaccard-Koeffizienten aus den Mengen der gemessenen Landmarken zu jedem Simulationszeitpunkt bei jeweils gleichem Landmarkenkatalog und gleicher Trajektorie. Es wurden die gesamte Zeitdauer jedes Simulationslauf ausgewertet und über alle Simulationsläufe gemittelt.

Die Unterscheidung der Jaccard-Koeffizienten nach der Anzahl der Landmarken im Katalog in Abbildung 24 zeigt deutlich, dass die Strategien für den Katalog mit 25 Landmarken selten eine Wahlmöglichkeit hatten, da nur eine Landmarke sichtbar war und diese dann auch nachverfolgt wurde. Mit steigendem Umfang des

Katalogs und somit steigender Auswahl nimmt auch die Überlappung der beobachteten Landmarkenmengen ab. Die Ähnlichkeit der Performance ist somit nicht auf die geringe Auswahl oder gar eine ähnliche Auswahl der beobachteten Landmarken zurückzuführen. Vielmehr werden häufig unterschiedliche Landmarken ausgewählt und trotzdem eine ähnliche Performance erzielt. Lediglich unter den Strategien auf Basis der zu erwartenden Kovarianzmatrix gibt es eine geringfügig erhöhte Ähnlichkeit. Die unterschiedlichen Normierungen scheinen somit einen geringeren Einfluss auf die Auswahl zu haben als der Einsatz einer Heuristik. Trotzdem liegt auch für die Strategien 8 – 10 die Überlappung bei den höheren Landmarkenanzahlen bei unter 50%. Das eine gewisse Unabhängigkeit der Performance von der konkreten Wahl der nachzuverfolgenden Landmarke besteht, zeigt sich auch daran, dass die zufällige Auswahl der Landmarke eine deutliche bessere Performance aufweist, als alle statischen Strategien.

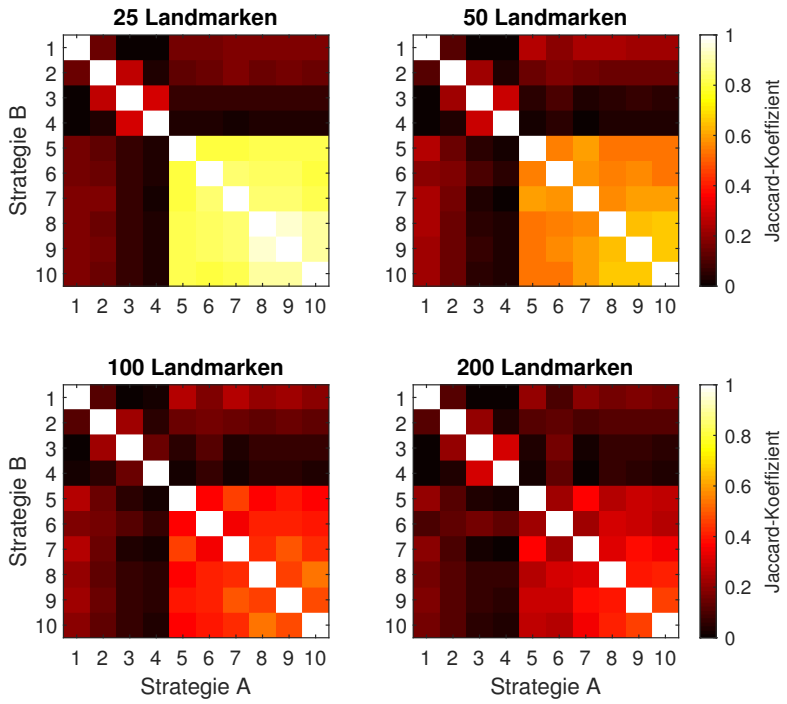

Abbildung 24: Mittlere Überlagerung der Menge sichtbarer Landmarken zwischen je zwei Strategien anhand der Jaccard-Koeffizienzen aus den Mengen der gemessenen Landmarken zu jedem Simulationszeitpunkt bei jeweils gleichem Landmarkenkatalog und gleicher Trajektorie. Es wurde die gesamte Zeitdauer jedes Simulationslauf ausgewertet und über die Simulationsläufe mit gleicher Landmarkenanzahl im Katalog gemittelt.

Zu einem ähnlichen Resultat führt auch der Vergleich der mittleren Ausrichtung der PTU, wozu der Abstand zwischen den PTU-Positionen zu jedem Simulationszeitpunkt nach Gleichung 56 ermittelt und ein mittlerer Wert über alle Simulationsläufe und Zeitpunkte berechnet wurde (Abb. 25). Für die statischen Strategien ist die PTU-Position natürlich fest und die Ausrichtung zu 0° ist am ähnlichsten zu denen der dynamischen Strategien. Von diesen werden zwar keine gegensätzlichen Blickrichtungen gewählt, jedoch ist die mittlere Ausrichtung der PTU paarweise jeweils signifikant unterschiedlich.

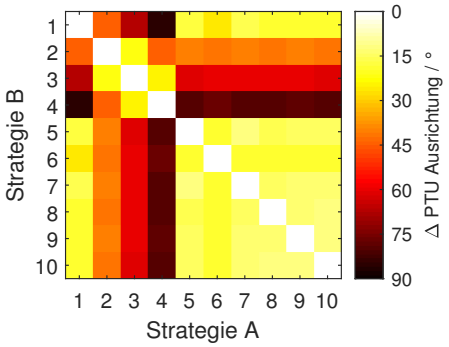

Abbildung 25: Mittlerer Abstand der PTU Ausrichtungen bei jeweils gleichem Landmarkenkatalog und gleicher Trajektorie. Es wurde die gesamte Zeitdauer jedes Simulationslauf ausgewertet und über alle Simulationsläufe gemittelt.

Für Strategie 6, welche die Maximierung der Anzahl von Landmarken im Blickfeld zum Ziel hat, wird hier die größte Abweichung unter den dynamischen Strategien beobachtet. Diese Abweichung zeigt sich bei Berücksichtigung des Einflusses der Landmarkenanzahl im Katalog (Abb. 26) insbesondere für Anzahlen ≥ 100. Auf diese stärkere Abweichung des Blickwinkels von den übrigen Strategien ist vermutlich auch die erhöhte Performance der Strategie für Situationen mit vielen Landmarken zurückzuführen (s. A.4.1 und Abb. 22). Dabei ist es nicht verwunderlich, dass sich diese Unterschiede erst dann manifestieren, wenn auch genügend Landmarken vorhanden sind, sodass sich die Blickwinkel in der Anzahl von sichtbaren Landmarken signifikant unterscheiden.

Zusammenfassend zeigt sich als klarer Trend, dass active vision eine deutliche Verbesserung gegenüber statischen Blickwinkeln für die Zustandsschätzung bringt. Innerhalb der dynamischen Strategien konnte hingegen kein klarer Trend bezüglich deren Performance ermittelt werden, was vermutlich auf die mangelnde Statistik der Simulation zurückzuführen ist. Das Verhalten der einzelnen Strategien ist jedoch so deutlich verschieden, dass mit einer besseren Statistik klarere Resultate zu erwarten sind. Als Trends deutet sich bereits an, dass die Berücksichtigung der erwarteten Ko-

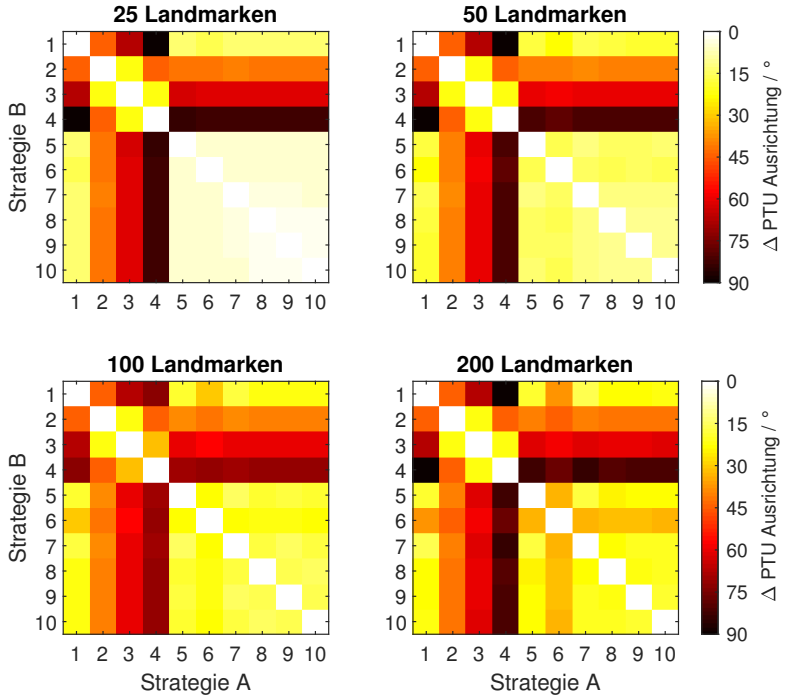

Abbildung 26: Mittlerer Abstand der PTU Ausrichtungen bei jeweils gleichem Landmarkenkatalog und gleicher Trajektorie. Es wurde die gesamte Zeitdauer jedes Simulationslauf ausgewertet und jeweils über die Simulationsläufe mit gleicher Landmarkenanzahl im Katalog gemittelt.

varianzmatrix mit Minimierung der Spur oder der spannweiten-normierten Varianz für verschiedene Szenarien geeignete Ansätze darstellen könnten. Bei einer hohen Anzahl von gleichzeitig sichtbaren Landmarken und somit vermutlich auch in einem SLAM-Szenario könnte die Auswahl des Blickwinkels mit der größten Anzahl von sichtbaren Landmarken einen vielversprechenden Ansatz darstellen.

4.4 Performance ohne GPS

In einer zweiten Analyse der Performance der verschiedenen Strategien wurden erneut die relative Höhe der Landmarken und deren Anzahl im Katalog variiert. Im Gegensatz zu den unter 4.3 beschriebenen Resultaten wurde hier jedoch keine initiale Filterkonvergenz durch die Unterstützung des GPS-Sensors herbeigeführt. Vielmehr standen während der gesamten Simulation ausschließlich die Messdaten von Beschleunigungssensor und Gyroskop der IMU sowie Barometer, Magnetometer und Kamera mit Lidar zur Verfügung.

Für die Performance ist somit zu erwarten, dass die initiale Konvergenz des Filters (s. 4.1.1) auf Basis der zur Verfügung stehenden Sensordaten einen entschei-

denden Einfluss hat. Für die Auswertung der Daten kann bei diesen Szenarien die gesamte Flugzeit berücksichtigt werden, da keine Änderung der Sensorkonfiguration im Laufe der Simulation eintritt.

Wie bereits für die erste Performance-Analyse wurde auch hier zunächst der Abstand zwischen tatsächlicher und geschätzter Endposition des UAVs für eine binäre Klassifikation herangezogen. Für diesen Simulationslauf gibt es analog zu Abbildung 14 geschätzte Flugbahnen, die den tatsächlichen Verlauf ohne größere Abweichung nachbilden und solche bei denen signifikante Abweichungen auftreten. Es treten dazu jedoch auch Situationen auf, in denen die tatsächliche Flugbahn von Beginn an ungenau geschätzt wird und sich der Positionsfehler stets weiter vergrößert, sodass die tatsächliche Flugbahn kaum noch zu erkennen ist (Abb. 27). Dies lässt den Rückschluss zu, dass in solchen Fällen vermutlich die initiale Konvergenz des Navigationsfilters nicht erfolgreich war und daraus der deutliche Fehlschlag resultiert. Ab einem gewissen Abstand zwischen geschätzter und tatsächlicher Position kann dann die optische Navigation auch nicht mehr zur Stützung der Zustandsschätzung beitragen, da die tatsächlich beobachteten Landmarken nicht erwartet und diese somit in der Fusion auf Basis der vorherigen Überprüfung mit der NIS-Teststatistik nicht berücksichtigt werden (s. 2.3.2.1).

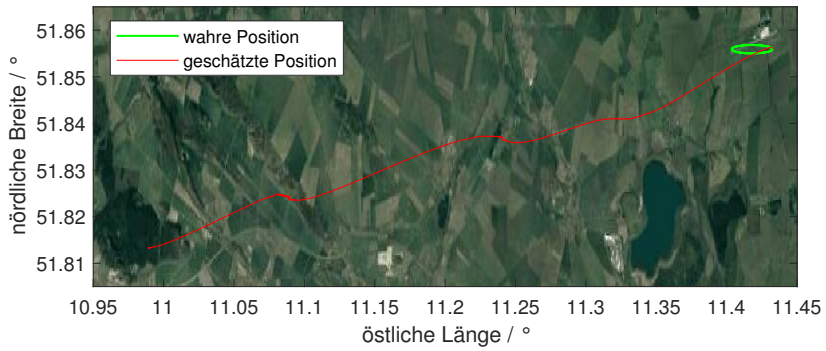

Abbildung 27: Aufsicht auf die Karte der Region um den Flughafen Cochstedt (aus Google Maps) mit der tatsächlichen und geschätzten Flugbahn aus der Simulation. Von Beginn des Fluges an tritt eine kontinuierlich zunehmende Abweichung der Schätzung von der wahren Position auf. Das Beispiel stammt aus einem Simulationslauf mit zufälliger Landmarkenauswahl, 25 Landmarken im Katalog und einer relativen Höhe von 0.0.

In Abbildung 28 ist die Verteilung aller Abstände zwischen geschätzten und tatsächlichen Endpunkt über alle Simulationsparameter dargestellt. Etwa 37% aller Simulationsläufe weisen einen Fehler von 1 km oder weniger auf. Nahezu alle Einträge in diesem Intervall resultieren von Simulationsläufen, deren Fehler kleiner als 50 m

ist, dem Grenzwert der unter 4.3 zur Klassifikation eingesetzt wurde. An dieser Stelle wird als Grenzwert zur Klassifikation 1 km genutzt, wobei die Unterschiede dementsprechend vernachlässigbar sind. Darüber hinaus zeigt die Verteilung neben einer Häufung von Fehlern um $\approx$ 10 km singuläre Ausschläge in den Intervallen von $[19, 20]$ km, $[22, 23]$ km und $[30, 32]$ km. Diese Häufungen resultieren aus einer fehlgeschlagenen initialen Filterkonvergenz und darauffolgender Zustandsschätzung vorwiegend auf Basis der inertialen Navigation. Da sich die initialen Fehlerwerte zwischen den Monto-Carlo-Läufen unterscheiden, jedoch pro Parameterkombination immer wieder wiederholen, kommt es zu diesen Häufungen.

Diese drastischen Abweichungen zwischen tatsächlicher und geschätzter Position zeigen nochmals deutlich, wieso zusätzliche Stützungen der Zustandsschätzung durch die Sensorfusion unerlässlich sind, da bereits innerhalb eines Fluges von 300 s Dauer eine immense Akkumulation der Sensorfehler stattfindet.

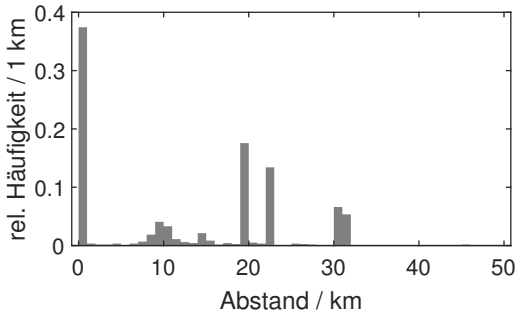

Abbildung 28: Kombinierte Verteilung der Abstände zwischen tatsächlicher und geschätzter Position am Ende des Fluges für alle Simulationsläufe.

Die Häufigkeit der Fehlschläge auf Basis der binären Klassifikation zeigt Abbildung 29, welche im Vergleich zu Abbildung 16 deutlich mehr Fehlschläge ausweist. Hieran wird deutlich, dass eine gute initiale Schätzung der Fehler oder eine geeignete Konvergenzphase entscheidend für eine andauernd zuverlässige Zustandsschätzung des Navigationsfilters ist, selbst wenn im weiteren Verlauf Sensoren ausfallen.

Im Vergleich weisen die statischen Blickwinkel von 45°, 67.5° und 90° erneut deutlich mehr Fehlschläge auf als die übrigen Strategien. Es zeigt sich darüber hinaus, dass für alle dynamischen Strategien erst für Situationen mit 100 oder mehr Landmarken im Katalog der Anteil der Erfolge überwiegt. Die Anzahl der sichtbaren Landmarken ist also auch hier ein entscheidender Faktor. Insbesondere für die initiale Filterkonvergenz kann vermutet werden, dass eine präzise Lokalisation des UAVs relativ zu den Landmarken zum Beginn des Fluges entscheidend ist.

Darüber hinaus lassen sich die bereits vermuteten Trends in Abbildung 29 zum Teil wiedererkennen. So ist der Anteil der Fehlschläge für Strategie 7 insbesondere

für eine hohe Anzahl von Landmarken im Katalog gegenüber den übrigen Strategien reduziert, während Strategie 10 insbesondere für eine geringe relative Höhe weniger Fehlschläge aufweist.

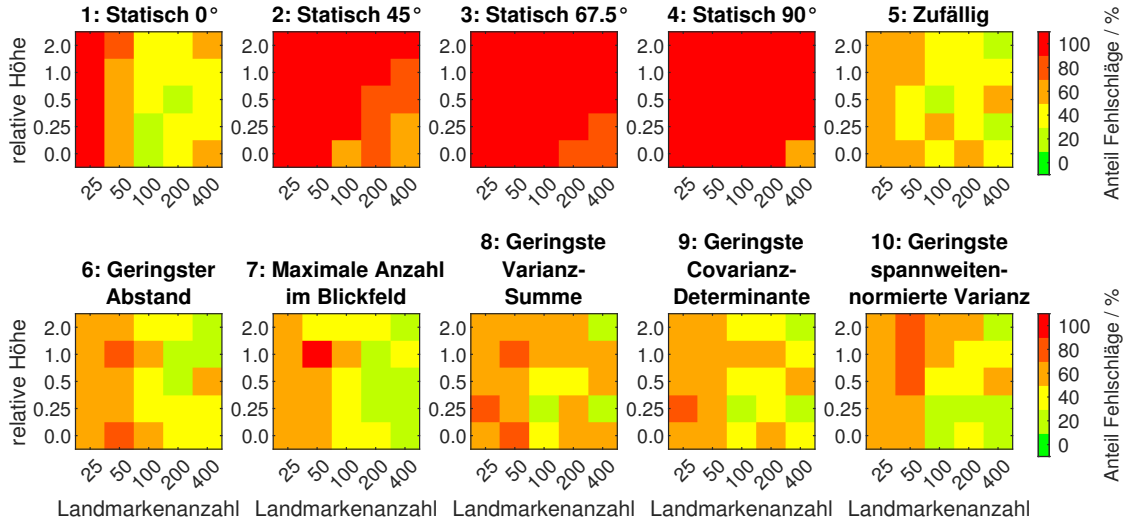

Abbildung 29: Anteil der Simulationsläufe mit Abständen von mehr als 1 km zwischen tatsächlicher und geschätzter Endposition in Abhängigkeit von den Simulationsparametern.

Werden nun für eine detailliertere Analyse erneut die Verteilungen der Positions-, Geschwindigkeits- und Lagefehler über die gesamten Simulationsläufe herangezogen (Abb. 30), zeigen sich zum Teil ähnliche aber zum Teil auch abweichende Trends im Vergleich mit dem Szenario mit dem Ausfall des GPS Sensors.

Zunächst zeigt sich, dass die statischen Strategien für alle drei Fehlerarten erneut deutlich größere Werte aufweisen als die dynamischen Strategien. Auch bleibt die Rangfolge unter den statischen Strategien erhalten, sodass auch hier ein Blick nach vorne zu geringeren Fehlern führt als ein Blick nach unten. Dies ist vermutlich wie zuvor auf die aus dem Blickwinkel resultierende Anzahl von Landmarken im FOV der Kamera zurückzuführen, da sich die simulierten Szenarien gegenüber des Szenarios mit Initialisierung bei eingeschaltetem GPS-Sensor nicht wesentlich geändert haben.

Weiterhin sind die Fehler der dynamischen Strategien untereinander deutlich ähnlicher. Hierbei tritt jedoch die erste Abweichung auf, da die zufällige Auswahl der nachzuverfolgenden Landmarke eine Ausnahme darstellt. Der durch diese Strategie erzielte Fehler ist gegenüber den übrigen dynamischen Strategien so stark erhöht, dass er in den reskalierten Darstellungen in Abbildung 30 nicht mehr erkennbar ist. Offenbar ist die zufällige Auswahl im Mittel deutlich schlechter geeignet die initiale Filterkonvergenz zu erreichen als die übrigen dynamischen Strategien, die eine gezielte Auswahl der nachzuverfolgenden Landmarke treffen.

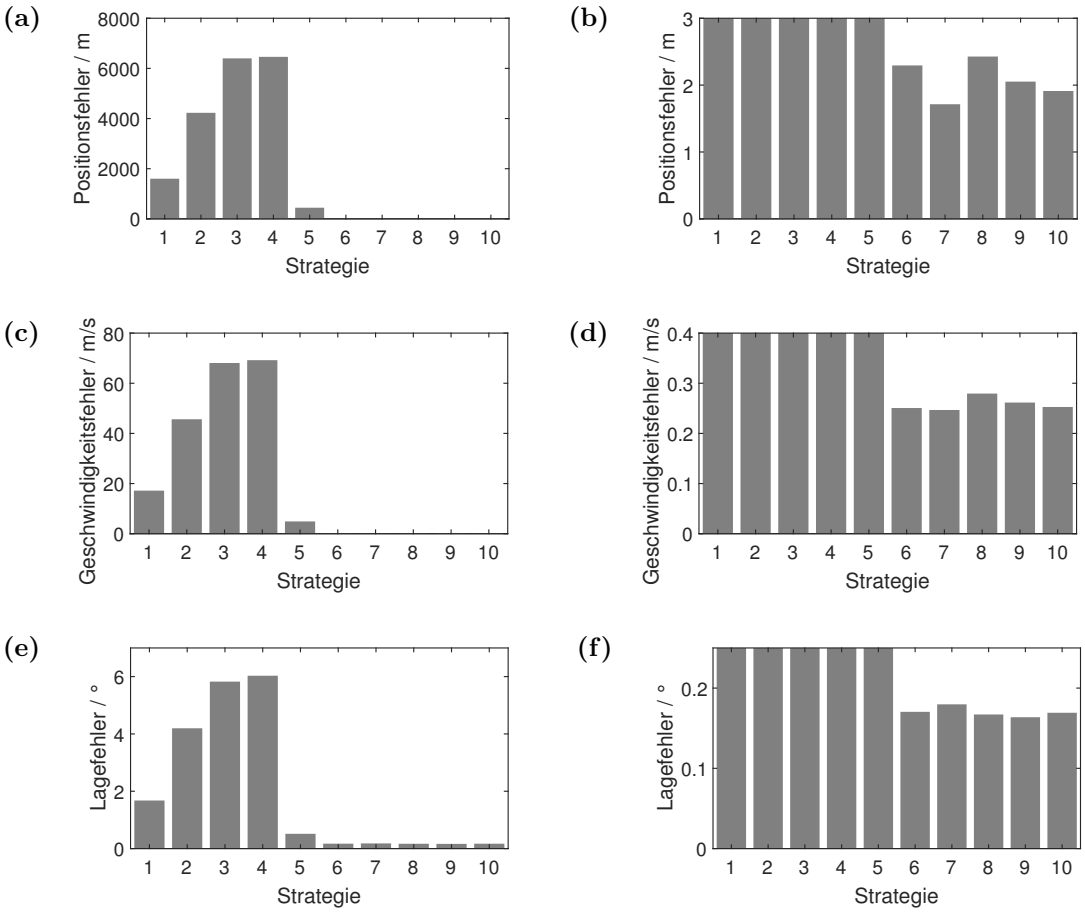

Abbildung 30: Mittlere Abstände zwischen geschätzter und tatsächlicher **(a)**, **(b)** Position, **(c)**, **(d)** Geschwindigkeit und **(e)**, **(f)** Lage. Es ist jeweils eine vollständige Übersicht (**(a)**, **(c)** und **(e)**) sowie eine Reskalierung mit Fokus auf die dynamischen Strategien (**(b)**, **(d)** und **(f)**) dargestellt. Der Mittelwert wurde aus allen Messpunkten im gesamten Zeitbereich über alle Simulationsläufe pro Strategie gebildet.

Im Vergleich der übrigen dynamischen Strategien gibt es keinen klaren Trend in der Rangfolge der verschiedenen Fehlertypen. Während Strategie 7 einen deutlich geringeren mittleren Positionsfehler und auch beim Geschwindigkeitsfehler mit geringem Abstand den kleinsten mittleren Fehlerwert aufweist, liegt sie in der Rangfolge bezüglich des Lagefehlers auf dem 5. Platz. Ähnlich gibt es auch für die übrigen Strategien keinen klaren Trend, sodass im Folgenden eine Detailbetrachtung der Fehler in Abhängigkeit von den variierten Parametern durchgeführt wird. Es lässt sich jedoch aus diesem mittleren Fehler schließen, dass die initiale Konvergenz des Filters von allen dynamischen Strategien mit Ausnahme der zufälligen Auswahl ähnlich gut beziehungsweise ähnlich häufig gelingt.

In der Analyse der Fehler in Abhängigkeit von den variierten Parametern wurden nur solche Monte-Carlo-Läufe berücksichtigt, die entsprechend der Abbildun-

gen 28 und 29 als Erfolge klassifiziert wurden. Damit wird zwar der zu Grunde liegende Datensatz zu Ungunsten der Statistik weiter eingeschränkt, jedoch kann die Performance der Strategien besser miteinander verglichen werden. Dies ist darauf zurückzuführen, dass die Fehler nicht durch große Werte verfälscht werden, die bei Fehlschläge vorliegen und maßgeblich von der zufälligen Filterinitialisierung (Abb. 28) und nicht der gewählten Strategie abhängen. Die Daten repräsentieren somit den mittleren Fehler der durch Stützung der Zustandsschätzung mittels optischer Navigation erhalten wird, wenn mit dieser die Initialisierung des Filters gelingt. Parameterkombinationen für die keine erfolgreichen Monte-Carlo-Läufe vorlagen, sind in den Abbildung 31 – 33 ausgegraut.

Bei der Betrachtung der Positions- und Geschwindigkeitsfehler zeigt sich erneut deutlich, dass ein umfangreicherer Landmarkenkatalog zu einer signifikanten Reduktion der Fehler für alle dynamischen Strategien führt. Dieser Trend ist auch für die statische Strategie mit einem Winkel von 0° zu erkennen. Hierbei treten für die dynamischen Strategien jedoch auch einzelne Ausreißer von diesem Trend auf. Dies lässt sich vermutlich auf Situationen zurückführen, in denen die initiale Konvergenz des Filters zwar gelungen ist, während des Fluges jedoch Situationen mit wenigen sichtbaren Landmarken vorkommen, sodass intermediär stärkere Fehler auftreten. Diese können dann in Folge einer erneut höheren Anzahl sichtbarer Landmarken wieder kompensiert werden konnte. Abbildung 34 zeigt eine Trajektorie, die eine solche Situation nahelegt.

Ein klarer Trend zur geringeren Fehlern bei größeren relativen Höhen ist im Gegensatz zum unter 4.3 beschriebenen Szenario nicht zu erkennen. Vielmehr scheinen häufig geringere relative Höhen zu kleineren Positions- und Geschwindigkeitsfehlern zu führen. Dieser Effekt könnte dadurch zustande kommen, dass für die initiale Konvergenz eine höhere Dichte von Landmarken und somit die Möglichkeit viele Landmarken gleichzeitig im FOV der Kamera zu beobachten vorteilhaft ist. Dieser Effekt könnte den Vorteil der erhöhten räumlichen Information bei größeren relativen Höhen aufwiegen, der für die Situation der Konvergenz mit Hilfe des GPS-Sensors noch dominierte.

Die Betrachtung der Lagefehler ergibt ein weniger einheitliches Bild, da insbesondere für die höchste Anzahl von Landmarken im Katalog alle dynamischen Strategien höhere Lagefehler aufweisen. Der Einfluss dieser Abweichungen sollte jedoch nicht überschätzt werden, da die Fehler in einem Bereich von lediglich $\approx 0.3°$ variieren. Wie schon in Abbildung 30**(f)** ersichtlich ist, ist der Unterschied im Lagefehler zwischen den Strategien insgesamt äußert gering, währen der Positionsfehler deutlich stärkere Variationen aufweist (Abb. 30**b)**. Dies ist möglicherweise darauf

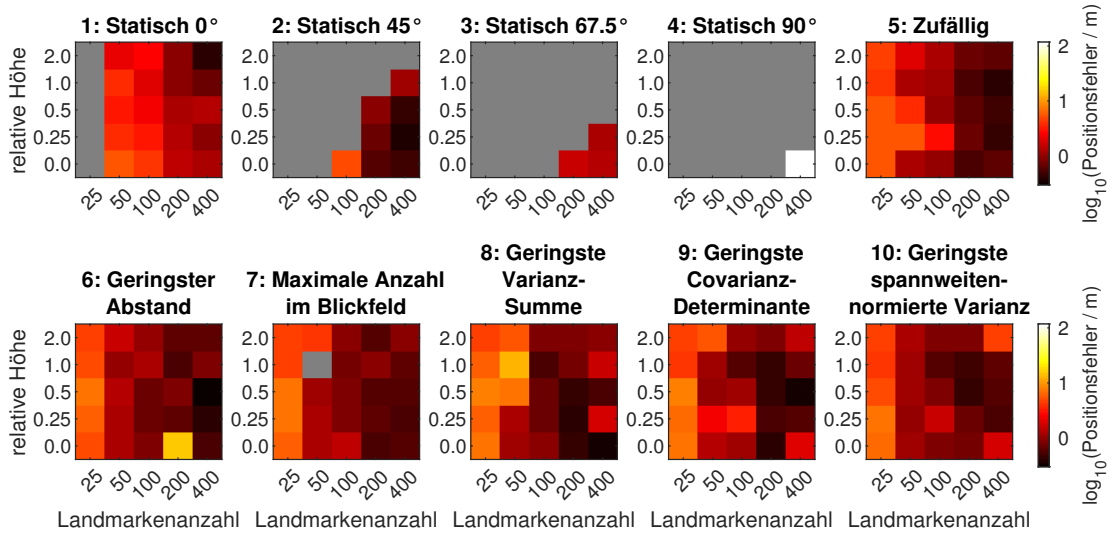

Abbildung 31: Abstand zwischen geschätzter und tatsächlicher Position in Abhängigkeit von den Simulationsparametern. Der Mittelwert wurde aus allen Messpunkten im gesamten Zeitbereich über alle Monte-Carlo-Läufe pro Parameterkonfiguration gebildet, die entsprechend der Abbildungen 28 und 29 als Erfolge klassifiziert wurden.

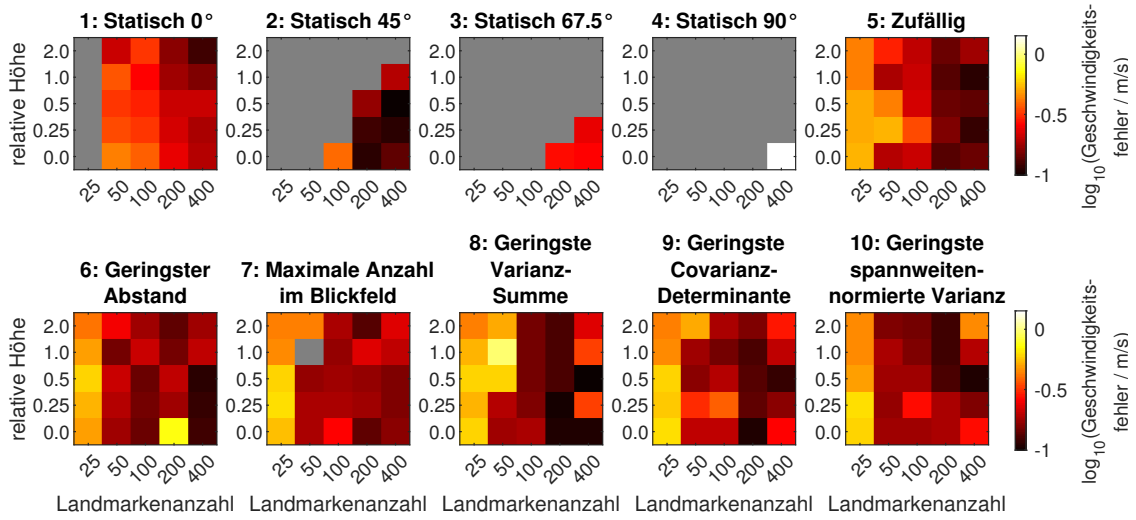

Abbildung 32: Abstand zwischen geschätzter und tatsächlicher Geschwindigkeit in Abhängigkeit von den Simulationsparametern. Der Mittelwert wurde aus allen Messpunkten im gesamten Zeitbereich über alle Monte-Carlo-Läufe pro Parameterkonfiguration gebildet, die entsprechend der Abbildungen 28 und 29 als Erfolge klassifiziert wurden.

zurückzuführen, dass Position und Geschwindigkeit im Vergleich zum Szenario unter 4.3 nicht per GPS-Messung gestützt werden können, während das Magnetometer

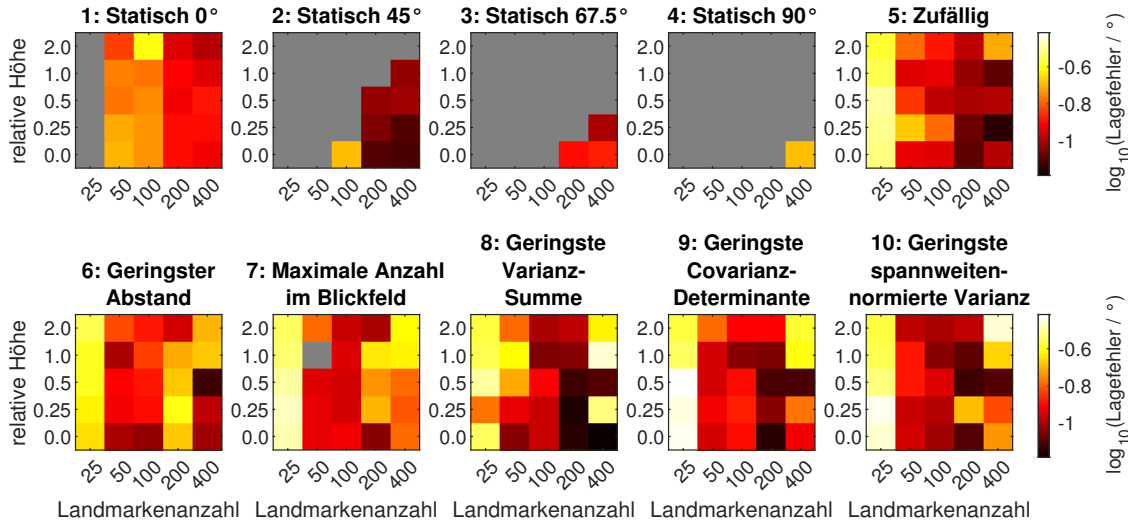

Abbildung 33: Drehwinkel zwischen geschätzter und tatsächlicher Lage in Abhängigkeit von den Simulationsparametern. Der Mittelwert wurde aus allen Messpunkten im gesamten Zeitbereich über alle Monte-Carlo-Läufe pro Parameterkonfiguration gebildet, die entsprechend der Abbildungen 28 und 29 als Erfolge klassifiziert wurden.

zur Lageschätzung in beiden Fällen beiträgt und die optische Navigation vor allem die Positionsschätzung stützen muss.

Für den Vergleich der Performance der einzelnen dynamischen Strategien wurde erneut eine Gegenüberstellung der Positionsfehler in Form von Abbildung 35 ermittelt. Eine genauere Beschreibung zur Bestimmung der Daten für diese Abbildung findet sich unter 4.3. Hierfür wurden in diesem Fall erneut nur erfolgreiche Läufe entsprechend der binären Klassifikation berücksichtigt und Parameterkombinationen ohne in beiden Strategien korrespondierende erfolgreiche Monte-Carlo-Läufe ausgegraut. Eine gleichartige Gegenüberstellung von Geschwindigkeits- und Lagefehler findet sich im Anhang A.4.2.

Im Gegensatz zu den Szenarien mit zu Beginn aktivem GPS-Sensor zeigt sich hier in vielen Situationen die zufällige Auswahl der nachzuverfolgende Landmarke als Strategie mit dem größten Fehler, selbst wenn im Gegensatz zu Abbildung 30**(e)** hier lediglich erfolgreiche Simulationsläufe berücksichtigt werden. Es gibt auch hier Situationen in denen diese Strategie den geringsten Fehler aufweist, was erneut auf eine zu geringe statistische Grundlage zur klaren Analyse von Trends hinweist.

Für den Positions- und auch Geschwindigkeitsfehler stellt hier, in einem klaren Unterschied zum Szenario mit initial aktivem GPS-Sensor, die Auswahl der Landmarke mit dem geringsten Abstand für Situationen mit 100 und mehr Landmarken fast immer die Strategie mit dem geringsten Fehler dar. Die Auswahl des Blick-

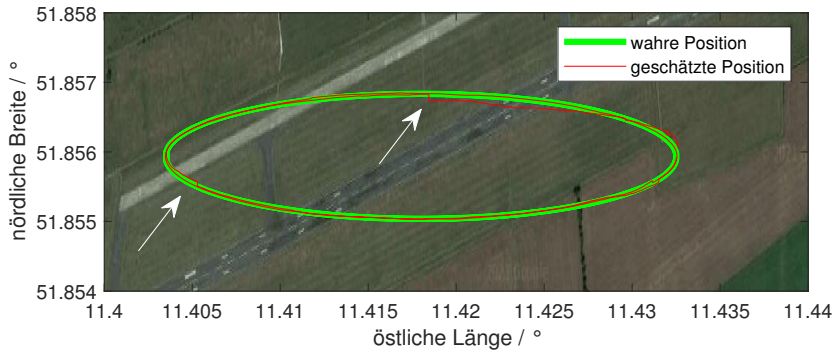

Abbildung 34: Aufsicht auf die Karte des Flughafens Cochstedt (aus Google Maps) mit der tatsächlichen und geschätzten Flugbahn aus der Simulation. Zwei sprunghafte Änderungen der Schätzung sind mit Pfeilen markiert. Das Beispiel stammt aus einem Simulationslauf mit zufälliger Landmarkenauswahl, 25 Landmarken im Katalog und einer relativen Höhe von 0.0.

winkels mit der größten Anzahl von Landmarken im FOV ist hingegen für diese beiden Fehlertypen in keinem Fall optimal. Die auf der erwarteten Kovarianzmatrix basierenden Strategien hingegen zeigen in den meisten Fällen keine besonders großen aber auch keine besonders kleinen Fehler. Lediglich Strategie 10 ist bei einer geringen Anzahl von Landmarken etwas häufiger die Strategie mit den geringsten Positions- und Geschwindigkeitsfehlern. Das hiervon auch Situationen mit geringen relativen Höhen betroffen sind, ist der einzige Trend der im Vergleich zum Szenario mit zunächst aktiviertem GPS-Sensor erhalten bleibt.

Bezüglich des Lagefehlers (s. A.4.2) bleiben die Strategien auf Basis der Kovarianzmatrix insgesamt ebenfalls im mittleren Fehlerbereich. Für diesen Fehler werden eher Trends erhalten, die den vorherigen Resultaten entsprechen, da Strategie 8 für eine geringe Landmarkenanzahl und Strategie 10 für geringere relative Höhen häufiger die Lagefehler im Vergleich der Strategien minimieren. Überraschenderweise sind die Rollen der Heuristiken bezüglich des Lagefehlers gerade vertauscht. Dieser wird für viele Fälle von Strategie 7, also der Auswahl des FOV mit den meisten sichtbaren Landmarken, minimiert, während die Auswahl der nächsten Landmarke (Strategie 6) den größten Lagefehler verursacht. Dies ist möglicherweise auf die insgesamt geringe Streuung der Lagefehler zurückzuführen. Jedoch ist es auch möglich, dass Landmarken mit dem geringsten Abstand aus geometrischen Gründen eher zentral vor dem UAV liegen, während für die maximale Anzahl keine solche Präferenz existiert. Daraus könnte folgen, dass eine größere Variabilität der Blickrichtungen sich besonders positiv auf die Lageschätzung auswirkt, für die Postions- und Geschwindigkeitsschätzung jedoch nicht relevant sind.

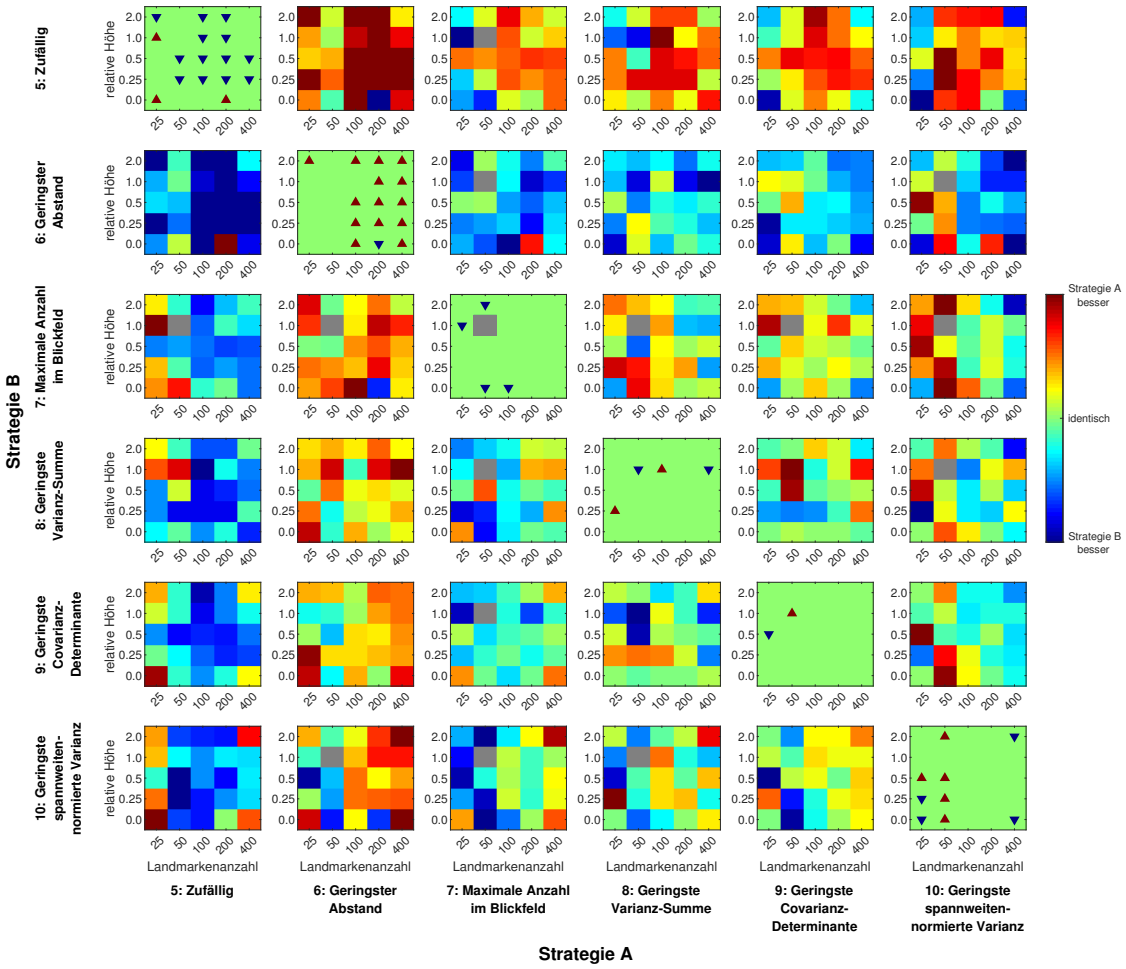

Abbildung 35: Pro Parameterkombination normierte Unterschiede der mittleren Positionsfehler mit den verschiedenen Strategien. Die Farben zeigen im Vergleich bessere oder schlechtere Performance der Strategien an. Auf der Diagonalen sind die Parameterkombinationen markiert, für die eine dynamische Strategie im Vergleich zu allen anderen am besten (rote Pfeilspitze nach oben) oder am schlechtesten (blaue Pfeilspitze nach unten) bezüglich der Minimierung der Fehler arbeitet. Es wurden jeweils nur Monte-Carlo-Läufe berücksichtigt, die entsprechend der Abbildungen 28 und 29 als Erfolge klassifiziert wurden.

Insgesamt liegen für die hier dargestellten Situationen keine eindeutigen Trends vor und es bleibt die Frage der Signifikanz der statischen Grundlage, die in jedem Fall für weitere Analysen ausgeweitet werden sollte. Dies ist insbesondere wichtig, da sich die beobachteten Trends zwischen den zwei Szenarien mit und ohne GPS-Sensordaten zur initialen Konvergenz teilweise deutlich unterschieden. Es bleibt jedoch auch hier die klare Feststellung, dass active vision einen signifikanten Vorteil gegenüber statischen Blickwinkeln, auch im Hinblick auf die initiale Konvergenz des Filters, darstellt und vermutlich unterschiedliche Strategien für die Auswahl der

nachzuverfolgenden Landmarke in Abhängigkeit vom geplanten Einsatzszenario und der verfügbaren Rechenleistung gewählt werden müssen.

5 Fazit

Im Rahmen dieser Arbeit wurde die Integration von Strategien zur Anpassung des Blickwinkels als Erweiterung der optische Navigation des superArtis Forschungshubschraubers des DLR durchgeführt und die Performance dieser Strategien analysiert.

Der Flugzustand des Hubschraubers, also dessen Position, Geschwindigkeit und Lage, wird mittels inertialer Navigation zeitlich propagiert und durch die Fusionierung mit verschiedenen Sensordaten kontinuierlich korrigiert. Zur Fusionierung wird ein Unscented Kalman-Filter eingesetzt. Als Sensoren stehen neben Beschleunigungssensor und Gyroskop der IMU zur inertialen Navigation Barometer, Magnetometer und GPS-Sensor sowie die hier im Fokus stehenden Sensoren zur optische Navigation zur Verfügung. Mithilfe einer Kamera und eines Lidar-Sensors werden die Positionen von Landmarken relativ zum UAV bestimmt, wobei die Positionen der Landmarken in Weltkoordinaten bekannt sind. Die beiden optischen Sensoren sind an einer Pan-Tilt-Einheit befestigt, wodurch die Ausrichtung der Sensoren relativ zur Lage des UAVs angepasst werden kann. Das Ziel der Arbeit war es eine Strategie zu finden und zu implementieren, mit der die optischen Sensoren so positioniert werden können, dass die Zustandsschätzung durch den UKF mithilfe der Messdaten insgesamt optimiert wird.

Es wurden insgesamt zehn verschiedene Strategien implementiert und getestet. Davon waren vier statisch und sechs dynamisch. Als statische Strategien wurden die Positionierung der PTU bei einem festen Pan-Winkel von 0° und unterschiedlichen festen Tilt-Winkeln von 0° (vorwärts), 45°, 67.5° und 90° (abwärts) umgesetzt. Im Rahmen der dynamischen Strategien wurde jeweils eine Landmarke ausgewählt und für einen Zeitraum von maximal 5 s im Zentrum des FOV der optischen Sensoren gehalten und somit entsprechend nachverfolgt. Als dynamische Strategien wurden drei Strategien auf Basis der durch eine Messung erwarteten Kovarianzmatrix und drei Heuristiken umgesetzt. Die Kovarianzmatrix wurde mithilfe der Unscented Transformation basierend auf dem aktuellen Zustand und der bekannten Positionen der Landmarken im Weltkoordinatensystem ermittelt. Für die unterschiedlichen auf dieser Kovarianzmatrix basierenden Strategien wurde jeweils die Landmarke zur Nachverfolgung ausgewählt, für deren Positionierung im Zentrum des FOV der optischen Sensoren entweder die Spur, die Determinante oder die so genannte spannweitennormierte Varianz der erwarteten Kovarianzmatrix minimiert wird. Als Heuristiken wurden die zufällige Landmarkenauswahl als Referenz, die Auswahl der Landmarke mit dem geringsten Abstand zum UAV sowie mit der größten Anzahl gleichzeitig

mit der gewählten Landmarke sichtbarer Landmarken im FOV der Kamera implementiert.

Zur Analyse der Performance der Strategien wurden Simulationen von Flügen des superArtis Hubschraubers in einer in Matlab/Simulik implementierten Simulationsumgebung durchgeführt, wobei neben den Strategien auch die Anzahl der Landmarken im Katalog sowie die mittlere Höhe der Landmarken im Raum relativ zur mittleren Flughöhe variiert wurde. Es zeigt sich, dass in allen untersuchten Szenarien die Schätzung des Flugzustands mithilfe der dynamischen Strategien der Zustandsschätzung den statischen Strategien überlegen ist, da insgesamt deutlich geringere Zustandsfehler auftreten. Im Allgemeinen wirkt sich eine größere Anzahl von Landmarken positiv auf die Performance aller Strategien aus. Bezüglich der relativen Höhe der Landmarken sind die Ergebnisse weniger eindeutig. Liegt zu Beginn ein gut initialer Zustand des Navigationsfilter vor, so kann eine größere relative Höhe und damit eine höhere räumliche Variabilität der Landmarkenpositionen zu besseren Zustandsschätzungen führen. Für die Initialisierung des Navigationsfilters ist jedoch eine höhere Dichte von Landmarken und somit die erhöhte Wahrscheinlichkeit mehrere Landmarken im FOV der optischen Sensoren zu beobachten vorteilhaft.

Unter den dynamischen Strategien konnten keine eindeutigen Trends identifiziert werden, in welchen Situationen welche Strategie überlegen ist. Es zeigt sich auch, dass sich der Einsatz unterschiedlicher Strategien unterschiedlich auf die Schätzung der verschiedenen Komponenten des Flugzustands auswirken. Es deutet sich insgesamt an, dass für Situationen mit wenigen Landmarken die Strategie zur Minimierung der Spur der Kovarianzmatrix vielversprechend sein könnte. Dieses Szenario ist für eine reine Landmarken-Navigation realistisch, wo für einen großen Bereich nur wenige markante Punkte wie hohe Bauwerke oder ähnliches katalogisiert sind. Für eine größere Anzahl von Landmarken zeigen die beiden Heuristiken der Auswahl der Landmarke mit dem geringsten Abstand und der größten Anzahl gleichzeitig sichtbarer Landmarken hingegen Potenzial zur Optimierung der Zustandsschätzung. Somit könnten beide für SLAM-Anwendungen interessant sein, bei denen stets neue Objekte zur Nachverfolgung entdeckt werden können. Bei insgesamt geringer relativen Höhe und somit einer eher hohen Flugbahn über der Szenerie zeigt auch die Strategie zur Minimierung der spannweiten-normierten Varianz eine teilweise überdurchschnittliche Performance.

Alle diese Trends sind jedoch nicht stark ausgeprägt, was vermutlich auch auf die geringe vorliegende Datengrundlage zurückzuführen ist, auf der die beschriebene Auswertung fußt, da pro Parameterkonfiguration nur je fünf Simulationsläufe durchgeführt wurden.

Um die durchgeführten Analysen zu verbessern, sollten die zu Grunde liegenden Simulationen somit zunächst mit deutlich mehr Wiederholungen durchgeführt werden, sodass die Datengrundlage verbessert beziehungsweise vergrößert wird. Dadurch könnten die beschriebenen Trends entweder mit höherer Sicherheit bestätigt oder auf Grundlage der neuen Datenlage angepasst werden. Ebenfalls wäre die Variation weiterer Parameter von Interesse, wobei insbesondere die Untersuchung des Einflusses einer veränderten maximalen Nachverfolgungsdauer der gewählten Landmarke einen vielversprechenden Ansatzpunkt darstellt, da sich diese wie bereits in dieser Arbeit festgestellt signifikant auf die Laufzeit der PTU Steuerung auswirkt. Diese sollte so lang wie möglich gewählt werden, ohne die Performance des Navigationsfilters signifikant zu verschlechtern, während die Echtzeitfähigkeit der Navigations- und Steuersoftware weiterhin gewährleistet bleibt. Auch der im Rahmen dieser Arbeit nicht im Detail untersuchte Einfluss der verschiedenen Strategien auf die geschätzten Sensor- und Modellfehler stellt einen Ansatzpunkt dar, die hier vorgestellten Resultate weiter zu vertiefen.

Letztlich muss es auch ein Ziel sein, die mithilfe der auf verschiedener Annahmen und Vereinfachungen basierenden Simulation erprobten Strategien im realen Flugversuch zu testen. Die Resultate aus diesen Flugversuchen können dann mit den auf Basis der Simulation erhaltenen Ergebnissen verglichen und etwaige Abweichungen als Ansatzpunkt weiterer Untersuchungen genutzt werden.

A Ergänzende Abbildungen

A.1 Konvergenz der Unsicherheiten des Fehlerzustands

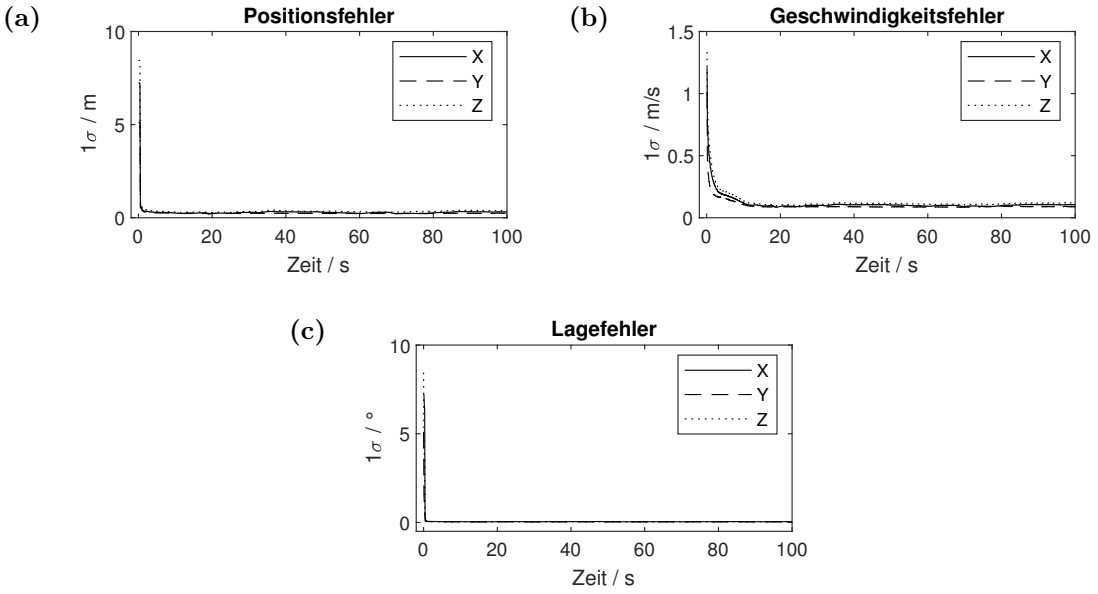

Abbildung 36: Verlauf des geschätzten Unsicherheit von **(a)** Positionsfehler, **(b)** Geschwindigkeitsfehler und **(c)** Lagefehler in einer Simulation mit zufälliger Landmarkenauswahl und 200 Landmarken mit maximaler relativer Höhe von 0.5 sowie während der gesamten Simulation verfügbaren Messdaten des GPS-Sensors.

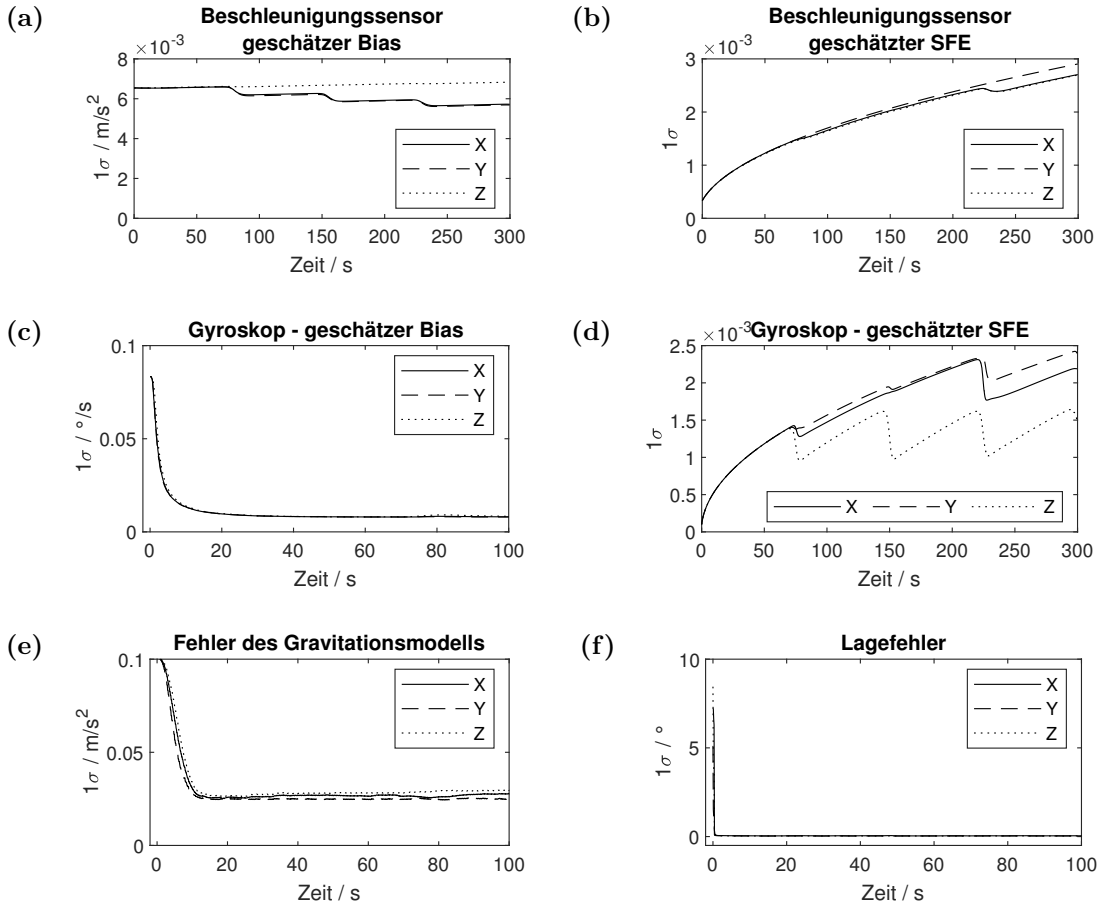

Abbildung 37: Verlauf des geschätzten Unsicherheit des **(a)** Bias und **(b)** SFEs des Beschleunigungssensors sowie des **(c)** Bias und **(d)** SFEs des Gyroskops ebenso wie der Fehler **(e)** des Gravitationsmodells und **(f)** des Luftdruckmodells in einer Simulation mit zufälliger Landmarkenauswahl und 200 Landmarken mit maximaler relativer Höhe von 0.5 sowie während der gesamten Simulation verfügbaren Messdaten des GPS-Sensors.

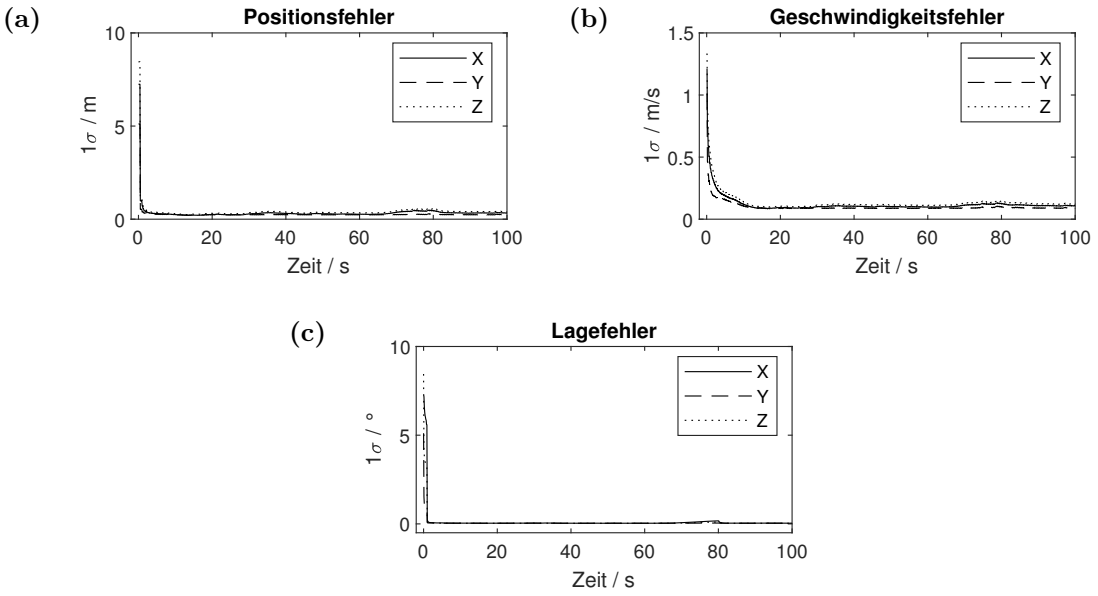

Abbildung 38: Verlauf des geschätzten Unsicherheit von **(a)** Positionsfehler, **(b)** Geschwindigkeitsfehler und **(c)** Lagefehler in einer Simulation mit statischem Blickwinkel von 0° und 200 Landmarken mit maximaler relativer Höhe von 0.5 sowie während der gesamten Simulation verfügbaren Messdaten des GPS-Sensors.

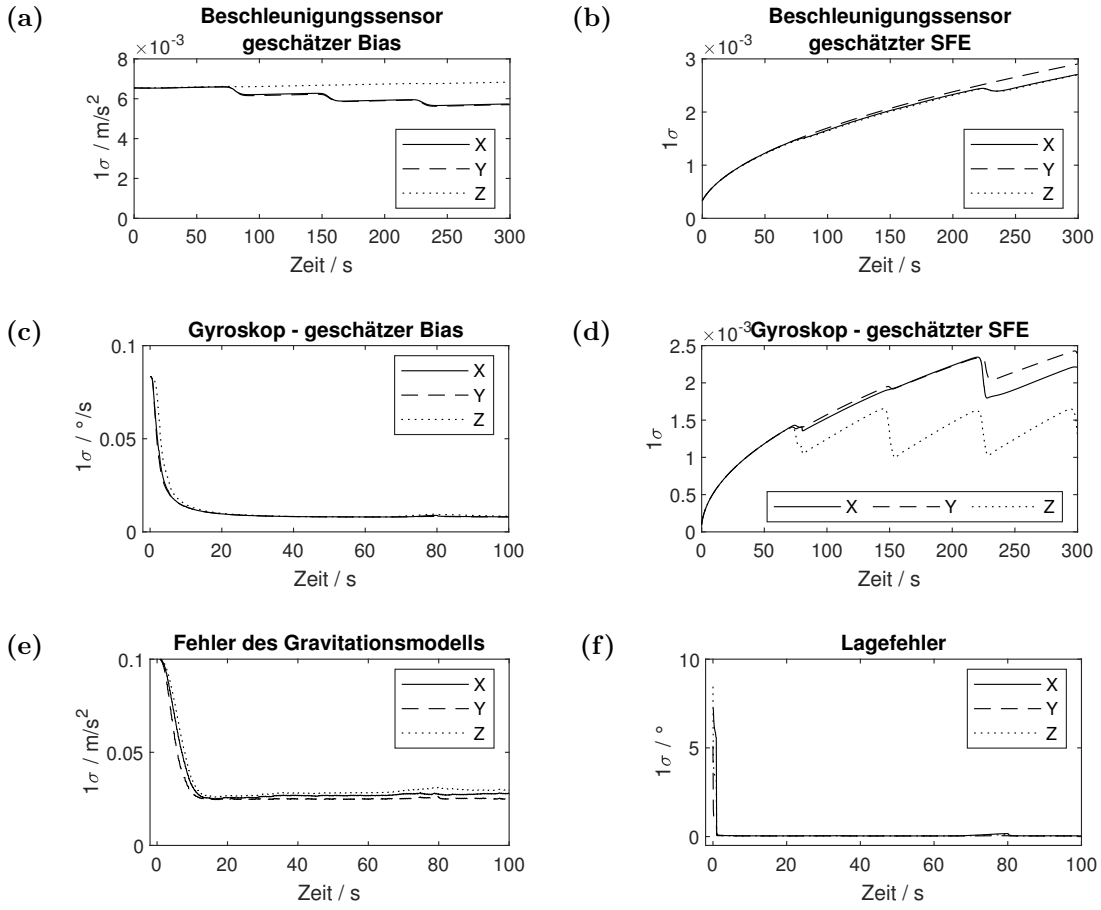

Abbildung 39: Verlauf des geschätzten Unsicherheit des **(a)** Bias und **(b)** SFEs des Beschleunigungssensors sowie des **(c)** Bias und **(d)** SFEs des Gyroskops ebenso wie der Fehler **(e)** des Gravitationsmodells und **(f)** des Luftdruckmodells in einer Simulation mit mit statischem Blickwinkel von 0° und 200 Landmarken mit maximaler relativer Höhe von 0.5 sowie während der gesamten Simulation verfügbaren Messdaten des GPS-Sensors.

A.2 Verteilungen der Nachverfolungsdauern

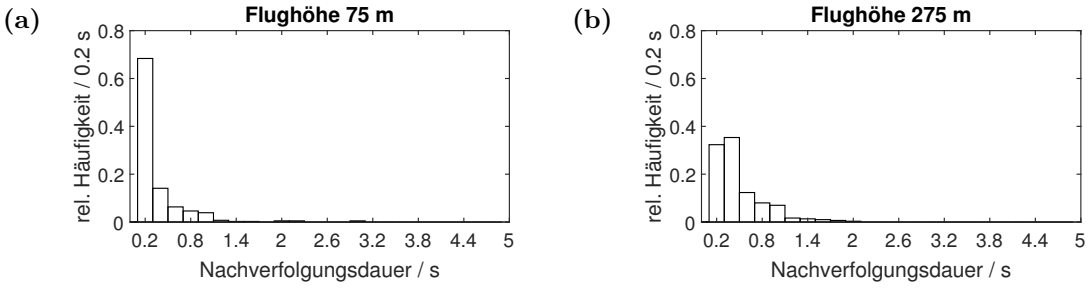

Abbildung 40: Verteilung der Dauern bis zur Wahl einer neuen Landmarke mit der Strategie zur Minimierung von $\mathrm{Tr}(\hat{P})$ entsprechend des unter 4.1.2 beschriebenen Szenarios bei **(a)** geringer Flughöhe von 75 m und **(b)** hoher Flughöhe von 275 m.

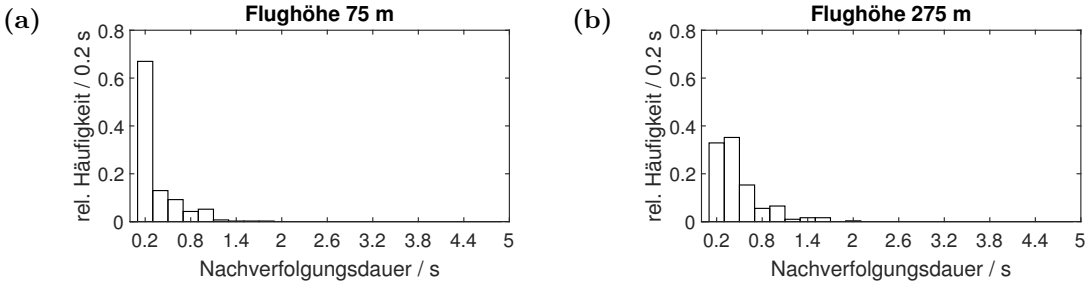

Abbildung 41: Verteilung der Dauern bis zur Wahl einer neuen Landmarke mit der Strategie zur Minimierung von $\det(\hat{P})$ entsprechend des unter 4.1.2 beschriebenen Szenarios bei **(a)** geringer Flughöhe von 75 m und **(b)** hoher Flughöhe von 275 m.

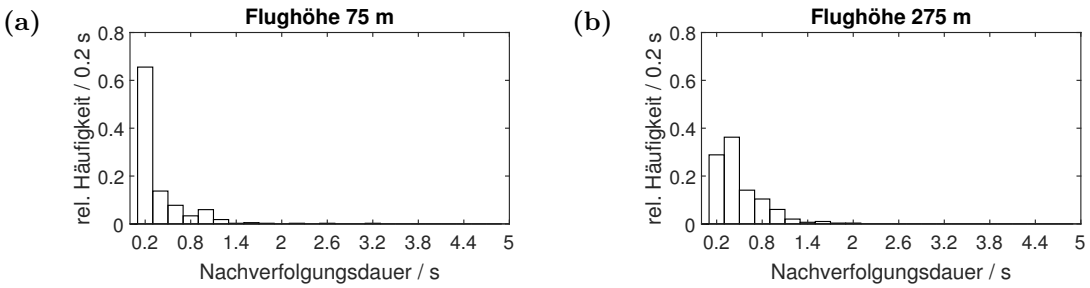

Abbildung 42: Verteilung der Dauern bis zur Wahl einer neuen Landmarke mit der Strategie zur Minimierung der spannweiten-normierten Varianz entsprechend des unter 4.1.2 beschriebenen Szenarios bei **(a)** geringer Flughöhe von 75 m und **(b)** hoher Flughöhe von 275 m.

A.3 Einfluss der relativen Höhe

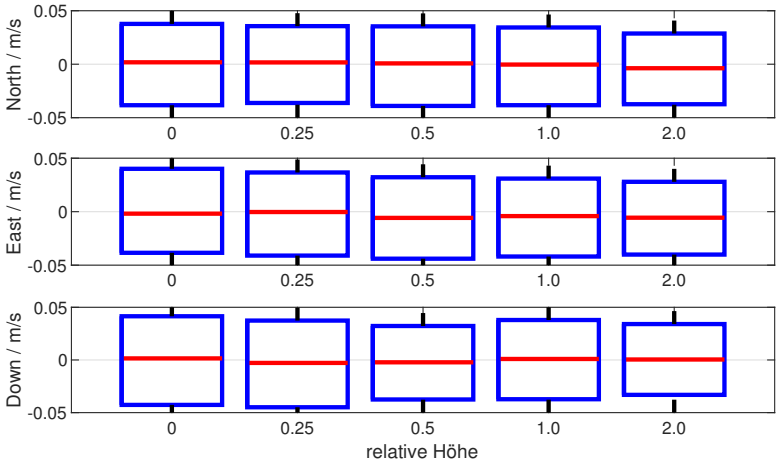

Abbildung 43: Boxplots der Geschwindigkeitsfehler im Zeitbereich 100 – 300 s für die Simulationsläufe mit 200 Landmarken am Beispiel der Strategie zur Minimierung der Spur der Kovarianzmatrix in Abhängigkeit von der relativen Höhe. Die Geschwindigkeitsfehler sind pro Koordinatenachse im NED-Koordinatensystem aufgetragen. Die roten Markierungen zeigen den Mittelwert die blauen Boxen die Standardabweichung an.

A.4 Detailvergleich der Strategien

A.4.1 Performance nach Verlust des GPS Signals

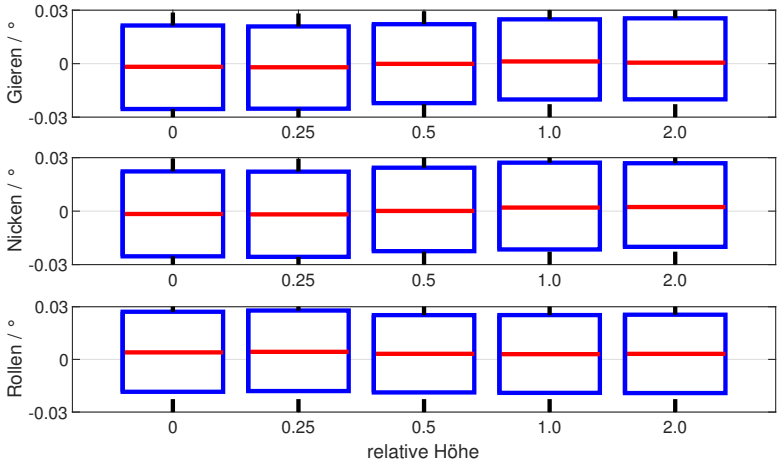

Abbildung 44: Boxplots der Lagefehler im Zeitbereich 100 – 300 s für die Simulationsläufe mit 200 Landmarken und der Strategie zur Minimierung der Spur der Kovarianzmatrix in Abhängigkeit von der relativen Höhe. Die Lagefehler sind als Gierungs-, Nick- und Rollwinkel bezüglich des NED-Koordinatensystem aufgetragen. Die roten Markierungen zeigen den Mittelwert die blauen Boxen die Standardabweichung an.

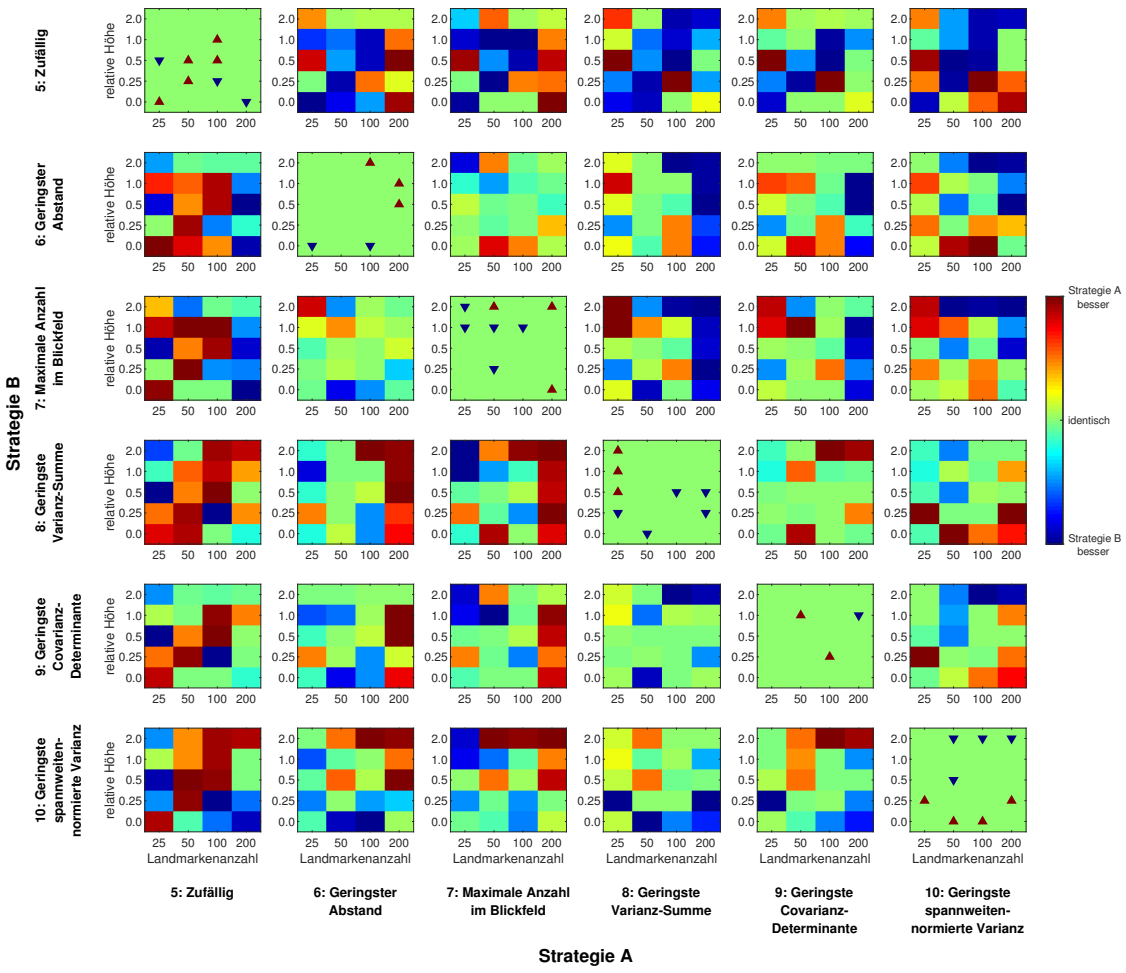

Abbildung 45: Pro Parameterkombination normierte Unterschiede der mittleren Positionsfehler mit den verschiedenen Strategien. Die Farben zeigen im Vergleich bessere oder schlechtere Performance der Strategien im Verglich an. Auf der Diagonalen sind die Parameterkombinationen markiert, für die eine dynamische Strategie im Vergleich zu allen anderen am besten (rote Pfeilspitze nach oben) oder am schlechtesten (blaue Pfeilspitze nach unten) performt.

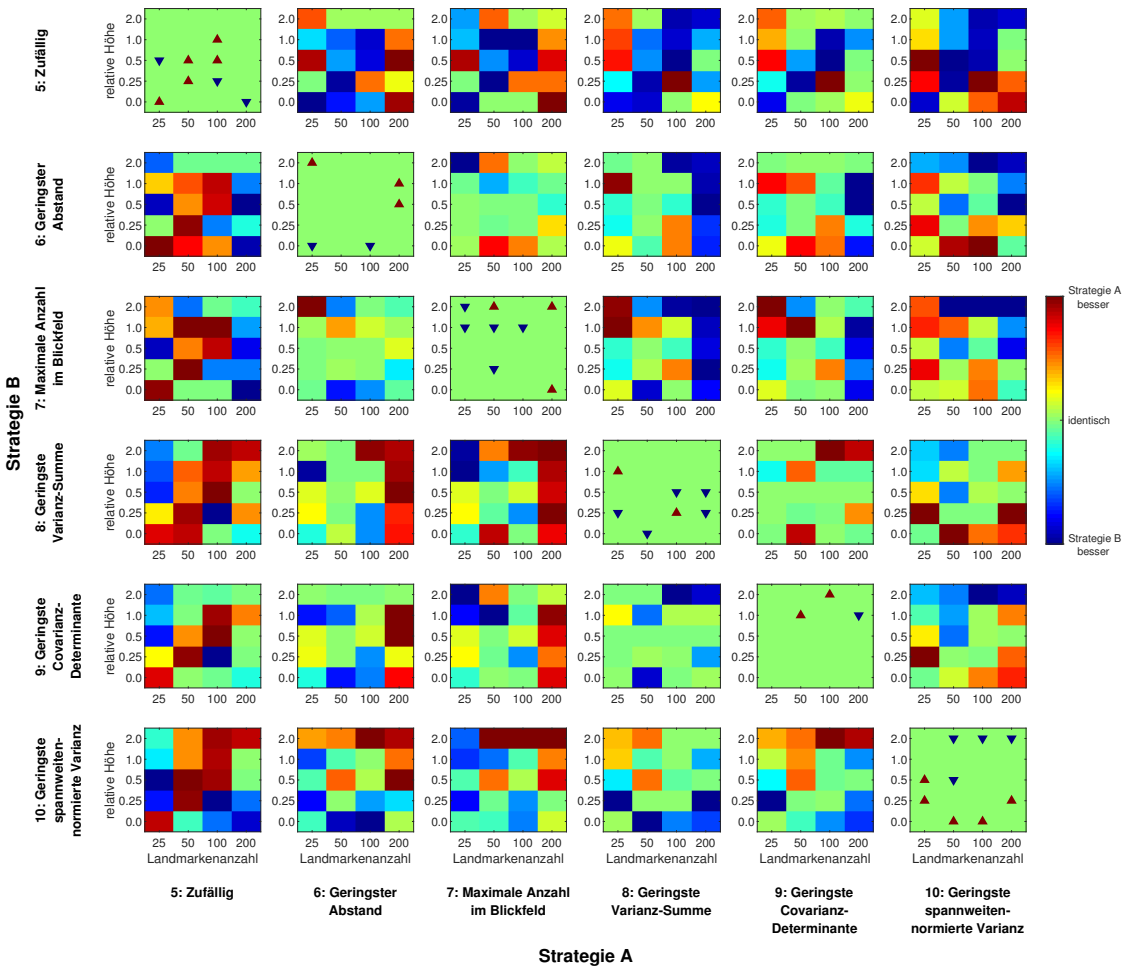

Abbildung 46: Pro Parameterkombination normierte Unterschiede der mittleren Geschwindigkeitsfehler mit den verschiedenen Strategien. Die Farben zeigen im Vergleich bessere oder schlechtere Performance der Strategien im Verglich an. Auf der Diagonalen sind die Parameterkombinationen markiert, für die eine dynamische Strategie im Vergleich zu allen anderen am besten (rote Pfeilspitze nach oben) oder am schlechtesten (blaue Pfeilspitze nach unten) performt.

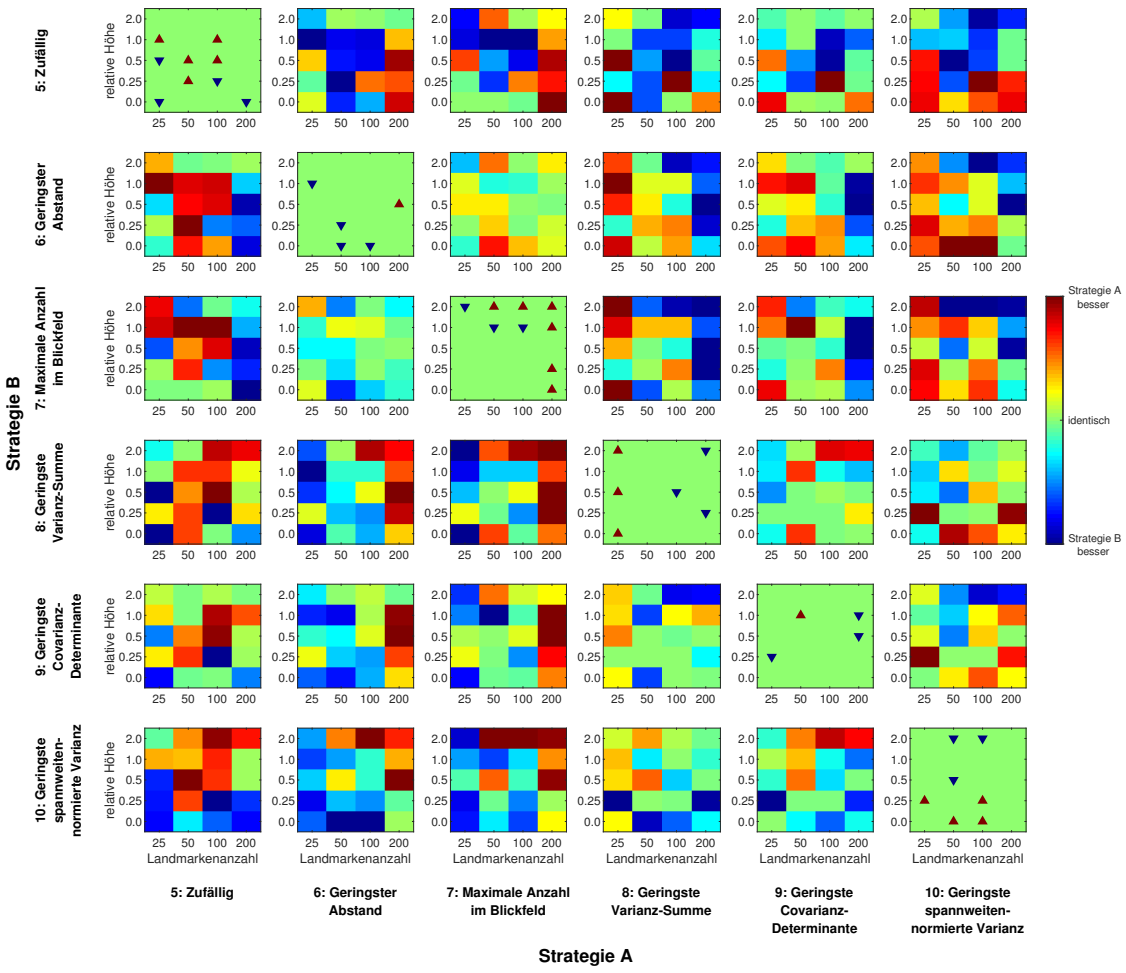

Abbildung 47: Pro Parameterkombination normierte Unterschiede der mittleren Lagefehler mit den verschiedenen Strategien. Die Farben zeigen im Vergleich bessere oder schlechtere Performance der Strategien im Verglich an. Auf der Diagonalen sind die Parameterkombinationen markiert, für die eine dynamische Strategie im Vergleich zu allen anderen am besten (rote Pfeilspitze nach oben) oder am schlechtesten (blaue Pfeilspitze nach unten) performt.

A.4.2 Performance ohne GPS

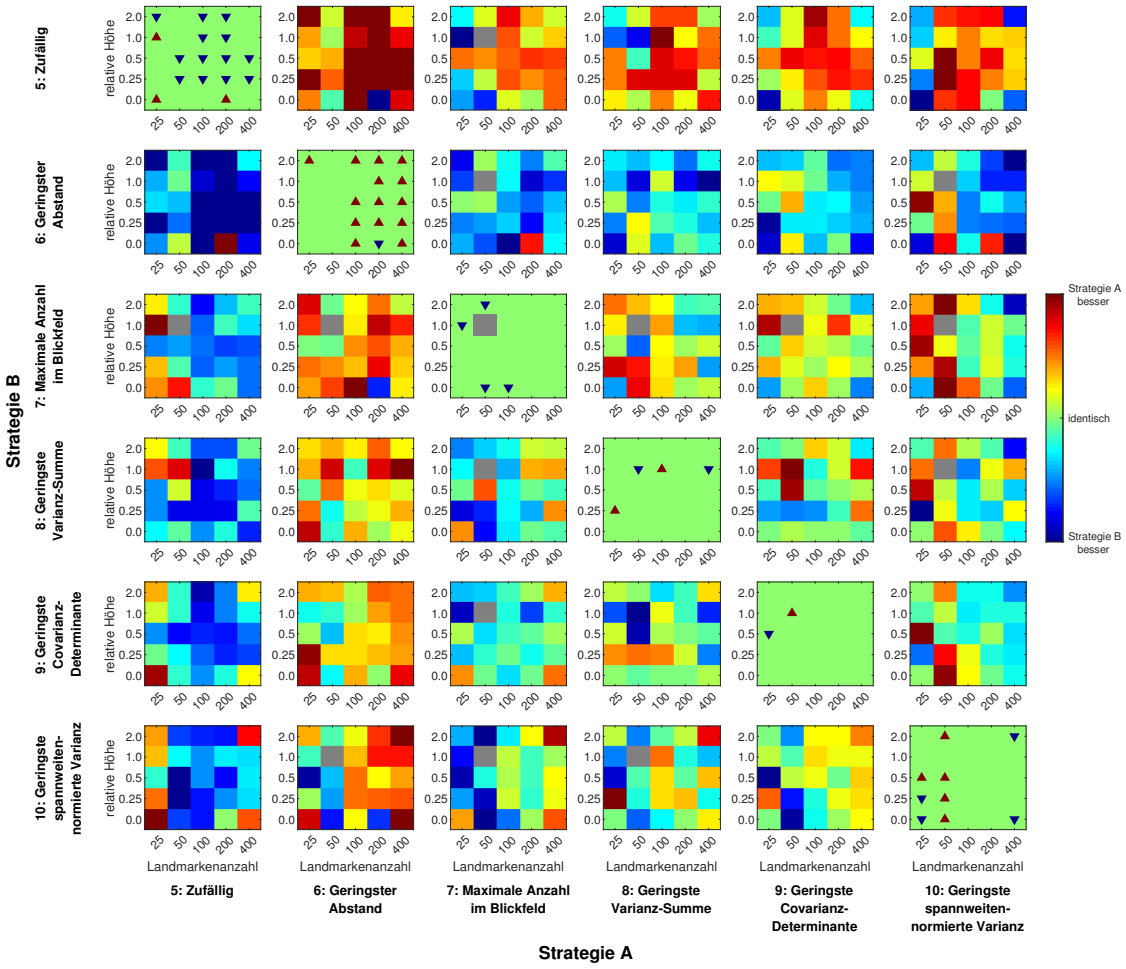

Abbildung 48: Pro Parameterkombination normierte Unterschiede der mittleren Positionsfehler mit den verschiedenen Strategien. Die Farben zeigen im Vergleich bessere oder schlechtere Performance der Strategien im Verglich an. Auf der Diagonalen sind die Parameterkombinationen markiert, für die eine dynamische Strategie im Vergleich zu allen anderen am besten (rote Pfeilspitze nach oben) oder am schlechtesten (blaue Pfeilspitze nach unten) performt.

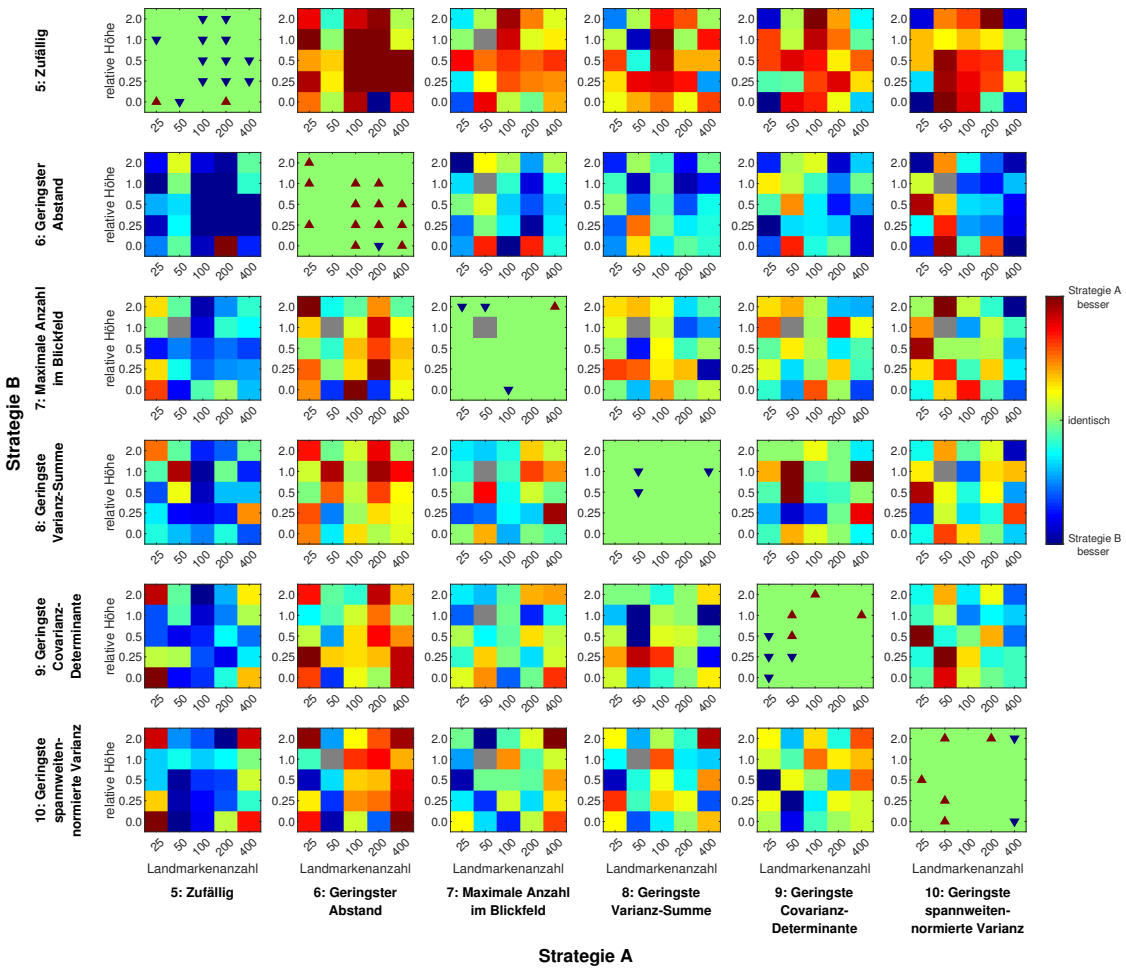

Abbildung 49: Pro Parameterkombination normierte Unterschiede der mittleren Geschwindigkeitsfehler mit den verschiedenen Strategien. Die Farben zeigen im Vergleich bessere oder schlechtere Performance der Strategien im Verglich an. Auf der Diagonalen sind die Parameterkombinationen markiert, für die eine dynamische Strategie im Vergleich zu allen anderen am besten (rote Pfeilspitze nach oben) oder am schlechtesten (blaue Pfeilspitze nach unten) performt.

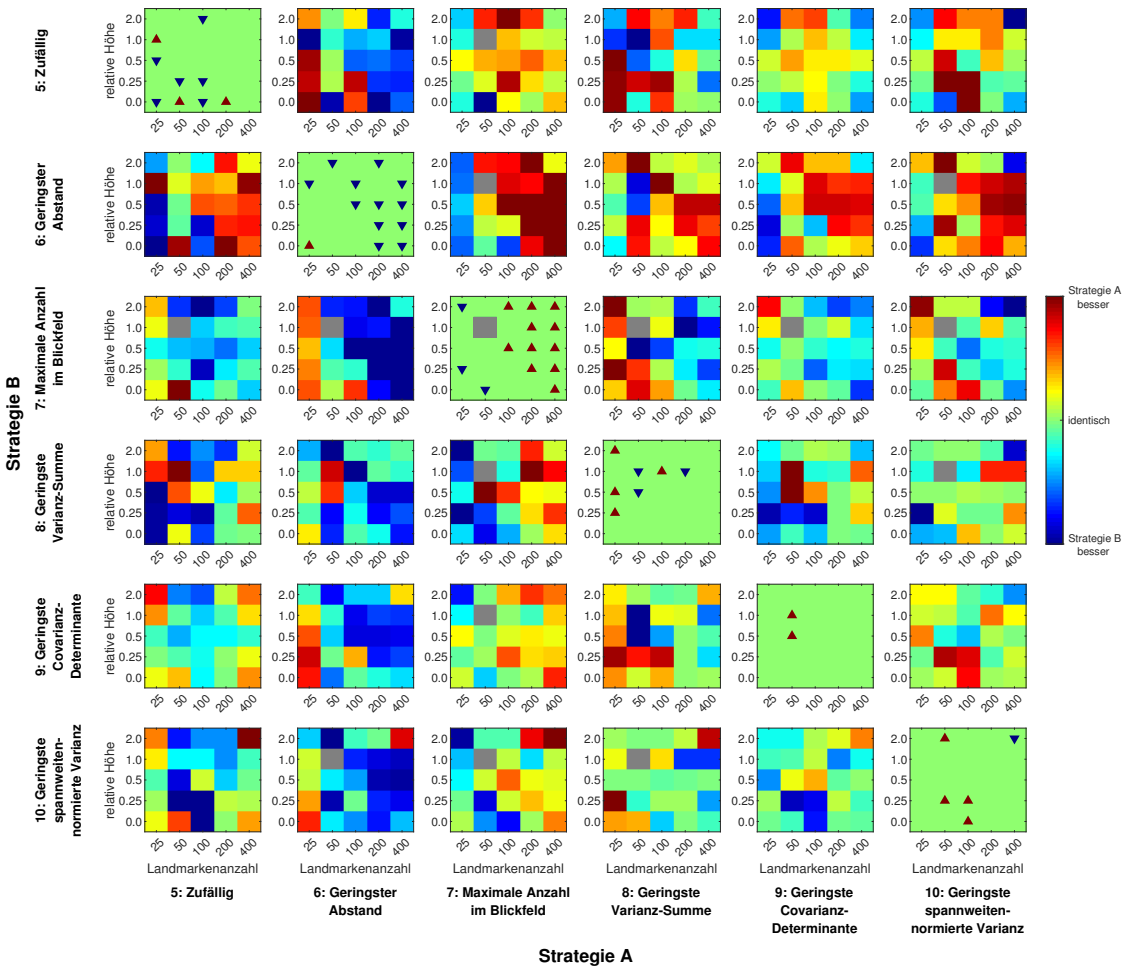

Abbildung 50: Pro Parameterkombination normierte Unterschiede der mittleren Lagefehler mit den verschiedenen Strategien. Die Farben zeigen im Vergleich bessere oder schlechtere Performance der Strategien im Verglich an. Auf der Diagonalen sind die Parameterkombinationen markiert, für die eine dynamische Strategie im Vergleich zu allen anderen am besten (rote Pfeilspitze nach oben) oder am schlechtesten (blaue Pfeilspitze nach unten) performt.

Literatur

[AAF+18] N. Ammann, F. Andert, T. Franz, H. KrÃ¼ger, M. Lingenauber, D. LÃ¼dtke, B. Maass, C. Paproth, S. Theil, and J. Wohlfeil. Project aton – autonomous terrain-based optical navigation – final report. Technischer bericht, Deutsches Zentrum fÃ¼r Luft- und Raumfahrt, Braunschweig, 2018.

[Age21] National Geospatial-Intelligence Agency. Wgs 84 data & apps. `https://earth-info.nga.mil/` (abgerufen am 16.10.2021), 2021.

[BEF96] J. Borenstein, H. R. Everett, and L. Feng. *Navigating Mobile Robots: Systems and Techniques*. A. K. Peters, 1996.

[BMB01] G. Backer, B. Mertsching, and M. Bollmann. Data- and model-driven gaze control for an active-vision system. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(12):1415–1429, 2001.

[BS05] M. Bryson and S. Sukkarieh. An information-theoretic approach to autonomous navigation and guidance of an uninhabited aerial vehicle in unknown environments. In *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2005.

[BS08] M. Bryson and S. Sukkarieh. Observability analysis and active control for airborne SLAM. *IEEE Transactions on Aerospace and Electronic Systems*, 44(1):261–280, 2008.

[BSLK01] Yaakov Bar-Shalom, X.-Rong Li, and Thiagalingam Kirubarajan. *Estimation with Applications to Tracking and Navigation*. John Wiley & Sons, Inc., 2001.

[BVN16] G. Balamurugan, J. Valarmathi, and V. P. S. Naidu. Survey on UAV navigation in GPS denied environments. In *2016 International Conference on Signal Processing, Communication, Power and Embedded System*. IEEE, 2016.

[Can08] K. Cannons. A review of visual tracking. Technical report, York University, 2008.

[CLK11] Shengyong Chen, Youfu Li, and Ngai Ming Kwok. Active vision in robotic systems: A survey of recent developments. *The International Journal of Robotics Research*, 30(11):1343–1377, aug 2011.

[Con85] C. Connolly. The determination of next best views. In *1985 IEEE International Conference on Robotics and Automation*. Institute of Electrical and Electronics Engineers, 1985.

[Con18] J.-P. Condomines. *Nonlinear Kalman Filtering for Multi-Sensor Navigation of Unmanned Aerial Vehicles*. ISTE Press, 2018.

[Dav98] A. J. Davison. *Mobile Robot Navigation Using Active Vision*. PhD thesis, University of Oxford, 1998.

[dCSKP09] G. C. H. E. de Croon, I. G. Sprinkhuizen-Kuyper, and E. O. Postma. Comparing active vision models. *Image and Vision Computing*, 27(4):374–384, 2009.

[DM98] A. J. Davison and D. W. Murray. Mobile robot localisation using active vision. In *Proceedings of the 5th European Conference on Computer Vision - Volume II*, 1998.

[fEI19] National Centers for Enviromental Information. The world magnetic model. `https://www.ngdc.noaa.gov/geomag/WMM/DoDWMM.shtml` (abgerufen am 23.09.2021), 2019.

[Fli21] Teledyne Flir. Flir ptu-d48e. `https://www.flir.com/products/ptu-d48e/?model=D48E-SS-SS-000-SS` (abgerufen am 23.09.2021), 2021.

[GCHM06] Vijay Gupta, Timothy H. Chung, Babak Hassibi, and Richard M. Murray. On a stochastic sensor selection algorithm with applications in sensor scheduling and sensor coverage. *Automatica*, 42(2):251–260, feb 2006.

[GG17] P. C. Garcia and L. E. M. Hernandez; P. G. Gil. *Indoor Navigation Strategies for Aerial Autonomous Systems*. Butterworth-Heinemann, 2017.

[Hea09] Headlessplatter. Kalman filter model 2. `https://en.wikipedia.org/wiki/Kalman_filter#/media/File:Kalman_filter_model_2.svg` (abgerufen am 09.04.2021), 2009.

[HFH+11] Murad Al Haj, Email Carles FernÃ¡ndez, Zhanwu XiongIvan Huerta, Jordi GonzÃ lez, and Xavier Roca. *Visual Analysis of Humans*. Springer-Verlag GmbH, 2011.

[HH17] Chingiz Hajiyev and Ulviye Hacizade. Testing the determinant of the innovation covariance matrix applied to aircraft sensor and actuator/surface fault detection. In *Advances in Sustainable Aviation*, pages 269–283. Springer International Publishing, nov 2017.

[iG13] iMAR GmbH. itracert-f400-q-e. `http://www.imar-navigation.de/downloads/TraceRT-F400Q-E_en.pdf` (abgerufen am 23.09.2021), 2013.

[Ima15] Stemmer Imaging. Allied vision prosilica gt 1280. `https://www.stemmer-imaging.com/media/uploads/cameras/avt/11/113221-Allied_Vision_Prosilica_GT-1380.pdf` (abgerufen am 23.09.2021), 2015.

[Jha16] A. R. Jha. *Theory, Design, and Applications of Unmanned Aerial Vehicles*. CRC Press, 2016.

[JU97] S. J. Julier and J. K. Uhlmann. New extension of the kalman filter to nonlinear systems. In Ivan Kadar, editor, *Signal Processing, Sensor Fusion, and Target Recognition VI*. SPIE, 1997.

[JU04] S.J. Julier and J.K. Uhlmann. Unscented filtering and nonlinear estimation. *Proceedings of the IEEE*, 92(3):401–422, mar 2004.

[Kal60] R. E. Kalman. A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 82(1):35–45, 1960.

[KE12] A. Kim and R. Eustice. Next-best-view visual slam for bounded-error area coverage. online, 2012.

[KE13] A. Kim and R. M. Eustice. Real-time visual SLAM for autonomous underwater hull inspection using visual saliency. *IEEE Transactions on Robotics*, 29(3):719–733, 2013.

[KE14] A. Kim and R. M. Eustice. Active visual SLAM for robotic area coverage: Theory and experiment. *The International Journal of Robotics Research*, 34(4-5):457–475, 2014.

[Kri97] S. Kristensen. Sensor planning with bayesian decision theory. *Robotics and Autonomous Systems*, 19(3-4):273–286, 1997.

[KS78] S. Kumar and J. Seinfeld. Optimal location of measurements for distributed parameter estimation. *IEEE Transactions on Automatic Control*, 23(4):690–698, aug 1978.

[Li17] S. Li. A review of feature detection and match algorithms for localization and mapping. *IOP Conference Series: Materials Science and Engineering*, 231:012003, 2017.

[Lid19] Velodyne Lidar. Puck. `https://velodynelidar.com/wp-content/uploads/2019/12/63-9229_Rev-K_Puck-_Datasheet_Web.pdf` (abgerufen am 23.09.2021), 2019.

[Lie95] C. C. Liebe. Star trackers for attitude determination. *IEEE Aerospace and Electronic Systems Magazine*, 10(6):10–16, 1995.

[LMNF08] V. Lepetit, F. Moreno-Noguer, and P.l Fua. EPnP: An accurate o(n) solution to the PnP problem. *International Journal of Computer Vision*, 81(2):155–166, 2008.

[MA13] A. Mitiche and J. K. Aggarwal. *Computer Vision Analysis of Image Motion by Variational Methods*. Springer Verlag, 2013.

[Maa16] B. Maass. Robust approximation of image illumination direction in a segmentation-based crater detection algorithm for spacecraft navigation. *CEAS Space Journal*, 8(4):303–314, 2016.

[MCM09] Nima Moshtagh, Lingji Chen, and Raman Mehra. Optimal measurement selection for any-time kalman filtering with processing constraints. In *Proceedings of the 48h IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference*. IEEE, dec 2009.

[Mei14] Andreas Meister. *Numerik linearer Gleichungssysteme*. Springer-Verlag GmbH, 2014.

[MKT11] B. Maass, H. KrÃ¼ger, and S. Theil. An edge-free, scale-, pose-and illumination-invariant approach to crater detection for spacecraft navigation. In *Image and Signal Processing and Analysis*, pages 603–608, 2011.

[MP71] R.K. Mehra and J. Peschon. An innovations approach to fault detection and diagnosis in dynamic systems. *Automatica*, 7(5):637–640, sep 1971.

[Ols10] E. Olson. Apriltag: Technical description. Technical report, Robotics Laboratory at the University of Michigan, 2010.

[Ols11] E. Olson. AprilTag: A robust and flexible visual fiducial system. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 3400–3407, 2011.

[Org93] International Civil Aviation Organization. *Manual of the ICAO Standard Atmosphere: Extended to 80 Kilometres (262 500 Feet)*. Doc (International Civil Aviation Organization). International Civil Aviation Organization, 1993.

[Per85] Ken Perlin. An image synthesizer. *ACM SIGGRAPH Computer Graphics*, 19(3):287–296, jul 1985.

[Sar07] Simo Sarkka. On unscented kalman filtering for state estimation of continuous-time nonlinear systems. *IEEE Transactions on Automatic Control*, 52(9):1631–1641, sep 2007.

[Sim06] D. Simon. *Optimal State Estimation*. John Wiley & Sons, 2006.

[SOL81] R. H. Shumway, D. E. Olsen, and L. J. Levy. Estimation and tests of hypotheses for the initial mean and covariance in the kalman filter model. *Communications in Statistics - Theory and Methods*, 10(16):1625–1641, 1981.

[Sor70] H. W. Sorenson. Least-squares estimation: from gauss to kalman. *IEEE Spectrum*, 7(7):63–68, jul 1970.

[SR08] Eric Sommerlade and Ian Reid. Information-theoretic active scene exploration. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, jun 2008.

[ST94] J. Shi and Tomasi. Good features to track. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*. IEEE Comput. Soc. Press, 1994.

[Swi20] SwissDrones. Sdo 50 v2 vtol. `https://www.swissdrones.com/img/layout/brochure/SDO-Brochure-v2.5-2020.pdf` (abgerufen am 23.09.2021), 2020.

[Sys14] LORD Sensing Systems. 3dm-gx3-25. http://files.microstrain.com/3DM-GX3-25-Attitude-Heading-Reference-System-Data-Sheet.pdf (abgerufen am 03.10.2021), 2014.

[Sys18] ADL Embedded Systems. Adlqm87pc pcie/104 sbc. https://www.adl-usa.com/wp-content/uploads/2018/03/Datasheet_ADLQM87PC-March12-2018.pdf (abgerufen am 23.09.2021), 2018.

[WF97] P. Whaite and F.P. Ferrie. Autonomous exploration: driven by uncertainty. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(3):193–205, 1997.

[WO16] J. Wang and E. Olson. AprilTag 2: Efficient and robust fiducial detection. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2016.

Selbstständigkeitserklärung

Name:	Marius Bauer
Matrikel-Nr.:	5679460
Fach:	Informatik
Modul:	Masterarbeit
Thema:	Multisensordatenfusion zur optischen Navigation für UAVs - Optimierung der Positionsschätzung durch Anpassung des Blickwinkels

Ich erkläre, dass ich die Abschlussarbeit selbstständig und ohne unzulässige Inanspruchnahme Dritter verfasst habe. Ich habe dabei nur die angegebenen Quellen und Hilfsmittel verwendet und die aus diesen wörtlich oder sinngemäß entnommenen Stellen als solche kenntlich gemacht. Die Versicherung selbstständiger Arbeit gilt auch für enthaltene Zeichnungen, Skizzen oder graphische Darstellungen. Die Arbeit wurde bisher in gleicher oder ähnlicher Form weder derselben noch einer anderen Prüfungsbehörde vorgelegt und auch noch nicht veröffentlicht. Mit der Abgabe der elektronischen Fassung der endgültigen Version der Arbeit nehme ich zur Kenntnis, dass diese mit Hilfe eines Plagiatserkennungsdienstes auf enthaltenen Plagiate geprüft werden kann und ausschließlich für Prüfungszwecke gespeichert wird.

Datum: XX. November 2021 Unterschrift: _____