

FEHLERTOLERANTE VERFAHREN ZUR EINHALTUNG VON GEOFENCES FÜR UNBEMANNTE KIPPFLÜGELFLUGZEUGE

E. König*

* Institut für Flugsystemdynamik der RWTH Aachen University, Wüllnerstraße 7, 52062 Aachen, Deutschland

Zusammenfassung

In dieser Arbeit werden fehlertolerante Verfahren für unbemannte Kippflügelflugzeuge unter Berücksichtigung von flugmechanischen Beschränkungen, Hindernissen und dem Geofence vorgestellt. Die entwickelten Verfahren stellen sicher, dass das Kippflügelflugzeug in einem sicheren Flugzustand verbleibt oder in solch einen Zustand überführt wird, falls die initial geplante Flugbahn nicht bis zum Ende abgeflogen werden soll. Dies wird durch die Berechnung von Notfallflugbahnen zu vordefinierten Sicherheitspunkten entlang der Flugbahn sichergestellt. Sobald einer der Sicherheitspunkte erreicht wird, kann das Flugsgerät im Schwebeflug verbleiben oder eine Landung durchführen.

Keywords

Flugbahnplanung; UAV; Kippflügelflugzeug

1. EINLEITUNG

Kippflügelflugzeuge verfügen über ein breit gefächertes Anwendungsspektrum und werden in verschiedenen Forschungsprojekten eingesetzt [1, 2]. Die Einsatzmöglichkeiten reichen von der Unterstützung bei Rettungsmissionen über den Transport von Gütern bis hin zur Verkehrsbeobachtung. Neben den flugmechanischen Beschränkungen des Fluggerätes spielen auch festgelegte Flugbegrenzungen, sogenannte Geofences, eine entscheidende Rolle. Sie begrenzen den zulässigen Aufenthaltsbereich des Fluggeräts und trennen erlaubte von nicht erlaubten Fluggebieten ab. Für den Betrieb unbemannter Fluggeräte, der sogenannten „speziellen“ Kategorie ist laut EU-Verordnung 2019/947 sicherzustellen, dass die Fluggeräte die in der Betriebserlaubnis definierten Fluggebiete nicht verlassen. Daher werden Verfahren benötigt, die auch bei Systemdefekten oder -ausfällen garantieren, dass das Fluggerät im zulässigen Fluggebiet verbleibt.

Bei der Flugbahnplanung wird vor der Durchführung der jeweiligen Mission überprüft, ob die geplante Flugbahn zu jeder Zeit innerhalb des Geofences verläuft. Damit wird sichergestellt, dass die geplante Missionsflugbahn die Anforderung der EU-Verordnung erfüllt. Aufgrund von Systemdefekten oder -ausfällen können während des Fluges jedoch Fehler auftreten, sodass die Missionsflugbahn nicht wie geplant abgeflogen werden kann. Damit kann eine sichere Fortführung des Fluges nicht mehr gewährleistet werden. Auch in solchen Fällen muss jedoch garantiert werden, dass das Fluggerät unter Berücksichtigung der flugmechanischen Beschränkungen den Geofence nicht verlässt und der Flug

schnell und sicher unterbrochen bzw. kontrolliert eine Landung eingeleitet wird. Trotz des Systemdefekts des Fluggeräts wird im Rahmen der Arbeit davon ausgegangen, dass das Fluggerät noch flugfähig ist, d.h. kein Servo- oder Aktuatorausfall vorliegt und die Positionsbestimmungslösung noch verfügbar und die Hardware noch funktionsfähig ist. Systemdefekt- und -ausfälle beziehen sich in dieser Arbeit auf einen Ausfall von oder in den elektronischen Komponenten, in denen die Bahnplanung durchgeführt wird. Ein ungewöhnliches Batterieverhalten kann jedoch beispielsweise auch ein Grund sein, um einen Flug zum Sicherheitspunkt einzuleiten.

In dieser Arbeit wurde ein Verfahren entwickelt und validiert, das bei Ausfall von Systemen oder bei Aktivierung in Notsituationen eine Notfallflugbahn abfliegt. Der Endpunkt dieser Notfallflugbahn ist ein festgelegter Sicherheitspunkt, an dem ein sicherer Schwebeflug oder eine sichere Landung möglich ist. Die Sicherheitspunkte müssen vor dem Flug festgelegt werden. Durch ein Preprocessing wird sichergestellt, dass ausschließlich Sicherheitspunkte verwendet werden, die innerhalb des zulässigen Fluggebiets und außerhalb von Hindernissen liegen. Bei der Planung der Notfallflugbahnen wird überprüft, ob die generierte Notfallflugbahn zulässig ist, d.h. innerhalb des Fluggebiets liegt, frei von Hindernissen ist und bis zu dem Sicherheitspunkt eine Geschwindigkeit von $0 \frac{m}{s}$ erreicht werden kann. Auch die flugmechanischen Beschränkungen des Fluggeräts werden berücksichtigt. Die Notfallflugbahn wird periodisch während des Fluges auf Basis des aktuellen Flugzustands des Fluggeräts bestimmt. In dieser Arbeit werden dafür Dubinspfade und der Constrained Wavefront Expansion (COWEX) Algorithmus

genutzt. Für beide Methoden muss sichergestellt werden, dass die Flugbahn frei von Hindernissen ist und die Länge der Flugbahn von der aktuellen Position des Fluggeräts bis zum Sicherheitspunkt ausreichend lang ist, um bis zum Sicherheitspunkt auf eine Geschwindigkeit von $0 \frac{m}{s}$ zu entschleunigen. Wird eine solche Flugbahn gefunden, wird diese als Notfallflugbahn abgespeichert, die im Notfall automatisch abgeflogen werden kann.

Bevor die Planung der Notfallflugbahn im Detail erklärt wird, wird das Fluggerät sowie dessen Flugführungssystem und die verwendeten Flugbahnplanungsmethoden vorgestellt. Danach wird auf die Erweiterung des Flugführungssystems eingegangen, die notwendig ist, um bei Fehlern einen automatischen Flug entlang der Notfallflugbahn durchzuführen. Daraufhin werden die verwendeten Verfahren zur Planung einer Notfallflugbahn erläutert und die Anforderungen der jeweiligen Verfahren erklärt. Anschließend werden die entwickelten Sicherheitsverfahren mithilfe von Simulationen validiert, die implementierten Funktionalitäten bewertet und ihre Anwendbarkeit analysiert. Abschließend wird ein Fazit über die entwickelten und implementierten Verfahren gezogen.

2. GRUNDLAGEN

Bevor die Funktionsweise der fehlertoleranten Verfahren vorgestellt wird, erfolgt eine Übersicht über das verwendete Fluggerät, das Flugführungssystem sowie der verwendeten Flugbahnplanungsmethoden.

2.1. Fluggerät

Die Verfahren wurden für ein Kippflügelflugzeug, welches in Bild 1 dargestellt ist, entwickelt.



BILD 1. Verwendetes Kippflügelflugzeug

Im Vergleich zu Starrflügelflugzeugen besitzen Kippflügelflugzeuge die Fähigkeit des senkrechten Startens und Landens und benötigen daher nur wenig Platz zum Starten und Landen. Ermöglicht wird das senkrechte Starten und Landen durch die Rotation der Tragfläche mit den daran angebrachten Antrieben um die Querachse. Somit kann die Schubrichtung in Abhängigkeit der Rotation der Tragfläche verändert werden. Die Einstellung der Schubrichtung ermöglicht dem Kippflügelflugzeug neben dem senkrechten Starten und Landen auch die Fähigkeit zu schwe-

ben. In diesen Flugzuständen ist die Tragfläche des Fluggeräts nach oben gekippt.

Im Vorwärtsflug ist die Tragfläche nach vorne gekippt und der Auftrieb wird durch die Tragflächen erzeugt. In dieser Konfiguration gleichen Kippflügelflugzeuge von der Anordnung der Tragfläche den konventionellen Starrflügelflugzeugen. Das Kippen der Tragfläche nach vorne ermöglicht einen effizienten Flug bei hohen Fluggeschwindigkeiten und stellt damit einen großen Vorteil gegenüber Multikoptern dar. Das verwendete Kippflügelflugzeug kann Fluggeschwindigkeiten von bis zu $30 \frac{m}{s}$ erreichen. Das Kippflügelflugzeug kann im gesamten Geschwindigkeitsbereich, also vom Schwebeflug bis zu seiner Maximalgeschwindigkeit, jeden Flugzustand stationär erfliegen [3].

2.2. Flugführungssystem

Das Flugführungssystem besteht wie in Bild 2 dargestellt aus einer Missionskontrollstation, einem Bahnplaner, einem Sicherheitsbahnplaner, einem Trajektorienplaner, einem Trajektorienregler und einem Flugzustandsregler, die auf verschiedenen Hardwarekomponenten aufgeteilt sind.

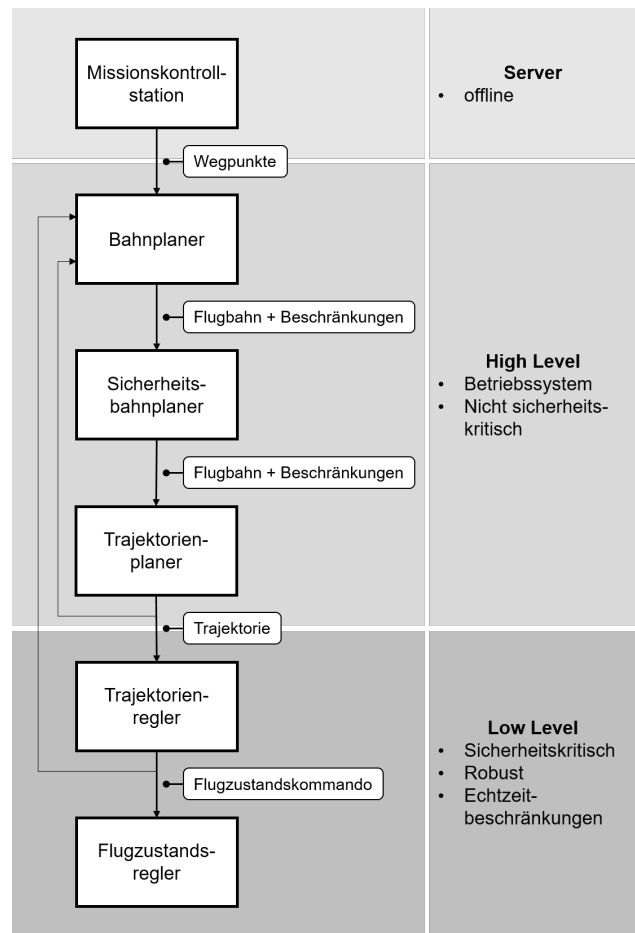


BILD 2. Flugführungssystem

Die Missionskontrollstation befindet sich auf einem Server, der Bahnplaner, der Sicherheitsplaner und der Trajektorienplaner auf dem High Level Board und

der Trajektorien- bzw. Flugzustandsregler auf dem Low Level Board des Fluggeräts.

In der Missionskontrollstation gibt der Anwender den Geofence, Hindernisse, Sicherheitspunkte und die gewünschten Wegpunkte vor. Dies erfolgt auf einem Server. Der auf dem High Level Board befindliche Bahnplaner erhält diese Wegpunkte sowie Datenrückführungen vom Trajektorienplaner und Trajektorienregler. Im Bahnplaner werden diese Wegpunkte in eine Flugbahn mit zugehörigen Beschränkungen überführt. Darauf wird die Flugbahn an den Sicherheitsbahnplaner übergeben, der für die Umsetzung der entwickelten Verfahren benötigt wird und die Flugbahn entsprechend anpasst. Diese Erweiterung wird in Unterabschnitt 2.4 erläutert. Im Trajektorienplaner werden die Flugbahnen mit ihren Beschränkungen in Trajektorien umgewandelt, sodass ein zeitabhängiger Verlauf der Flugbahn verfügbar ist. Vom Trajektorienplaner werden die vorausgeplanten Flugzustände über eine serielle Schnittstelle vom High Level Board an den Trajektorienregler auf das aus einem Mikrocontroller bestehenden Low Level Board übertragen. Hier werden die Flugzustände ausgewertet und neue Vorgaben an die Stellgrößen für die Aktuatoren des Fluggeräts vorgegeben, um der geplanten Flugbahn folgen zu können. Die Auswertung der Flugzustände auf dem Low Level Board erfolgt während des Fluges in Echtzeit und ist damit sicherheitskritisch.

2.3. Flugbahnplanungsmethoden

Im Rahmen der Arbeit wurden zwei verschiedene Flugbahnplanungsmethoden verwendet: der Dubinspfad [4] und der Constrained Wavefront Expansion (COWEX) Algorithmus [2, 5].

Der Dubinspfad ist der kürzeste Pfad zwischen zwei Punkten mit jeweiliger Richtung, die unter Einhaltung definierter Kurvenradien verbunden werden sollen. Der Dubinspfad besteht aus bis zu drei Kurvensegmenten und einem optionalen Geradensegment. Der Radius der Kurvensegmente wird durch den minimalen Kurvenradius des Fluggeräts bestimmt. Der Vorteil des Dubinspfades gegenüber anderen Bahnplanungsmethoden ist, dass die Pfade analytisch innerhalb kürzester Zeit berechnet werden können. Bei der Berechnung des Dubinspfades können jedoch keine Hindernisse oder Geofencegrenzen berücksichtigt werden.

Daher wird ein anderer Flugbahnplanungsalgorithmus verwendet, wenn ein berechneter Dubinspfad nicht frei von Hindernissen ist oder außerhalb des Geofences verläuft. Für solche Szenarien wird der COWEX Algorithmus verwendet. Der COWEX Algorithmus benötigt als Eingabeparameter einen Start- und einen Endpunkt mit jeweils definierter Ausrichtung, einen minimalen Kurvenradius, flugmechanische Beschränkungen und eine Zielgeschwindigkeit am Endpunkt. Das Prinzip hinter dem COWEX Algorithmus basiert auf der wellenartigen Ausbreitung von Kreisen variabler Größe, die in den hindernisfreien

Raum aufgespannt werden. Ausgehend von der Startposition werden Kreise aufgespannt und mittels einer Kostenfunktion evaluiert. Sobald ein Kreis den definierten Endpunkt erreicht, wird eine Flugbahn durch Verbindung der Kreismittelpunkte des letzten Kreises und seinen Vorgängerkreisen generiert. Eine detaillierte Beschreibung des COWEX Algorithmus ist in [2] und [5] zu finden.

2.4. Erweiterung des Flugführungssystems

Wie bereits in Unterabschnitt 2.2 erklärt wurde, wird das Flugführungssystem um den Sicherheitsbahnplaner erweitert, um das entwickelte Verfahren in das bestehende Flugführungssystem zu integrieren. Im Sicherheitsbahnplaner wird die initiale Missionsflugbahn immer wieder durch eine neu generierte Notfallflugbahn mit entsprechenden Beschränkungen ersetzt, die immer an einem Sicherheitspunkt endet. Hierbei werden der Geofence, Hindernisse und flugmechanische Beschränkungen berücksichtigt. Diese neu generierte Flugbahn wird dann wie in Unterabschnitt 2.2 erklärt, weiterverarbeitet.

Die Flugbahn von der aktuellen Position des Fluggeräts zu einem Sicherheitspunkt besteht immer aus zwei Teilen. Der erste Teil wird als B_1 und der zweite als B_2 bezeichnet. B_1 ist Teil der initialen Missionsflugbahn und B_2 ist eine mit einem der in Unterabschnitt 2.3 vorgestellten Flugbahnplanungsmethoden berechnete Flugbahn. In Bild 3 sind drei Beispielflugbahnen für die gleiche Missionsflugbahn, aber andere Startpositionen und Startzeitpunkte t_0 , t_1 und t_2 dargestellt. Die initiale Missionsflugbahn ist durch die gestrichelte Linie, die aktuelle Position des Fluggeräts durch das blaue Kreuz und der Sicherheitspunkt durch den roten Kreis dargestellt. Die Abzweigung von der Missionsflugbahn erfolgt an der Position, die das Fluggerät in fünf Sekunden erreichen haben wird. Diese Position wird als Offset-Position bezeichnet und ist mit einer magentafarbenen Raute markiert.

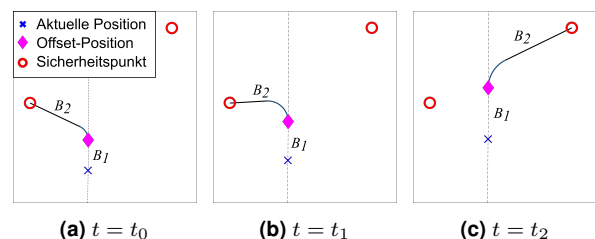


BILD 3. Notfallflugbahn für verschiedene Zeiten t_n auf der Missionsflugbahn

In Bild 3 ist zu erkennen, dass die B_1 Bahnen unterschiedlich lang sind, was durch die unterschiedlichen Geschwindigkeiten des Fluggeräts zu erklären ist. Im linken Beispiel (Bild 3a) ist die Geschwindigkeit des Fluggeräts kleiner als die des Fluggeräts in der Mitte (Bild 3b) bzw. im rechten Beispiel (Bild 3c), weshalb die Offset-Position in Bild 3a näher an der aktuellen Position des Fluggeräts ist als in Bild 3b und Bild 3c.

In diesem Beispiel fliegt das Fluggerät die initial geplante Missionsflugbahn ab. Mit fortschreitender Zeit verändern sich daher auch die Offset-Positionen und entsprechend auch die Bahnen B_1 und B_2 . Dabei wird B_2 kontinuierlich von der Offset-Position zu einem Sicherheitspunkt neu geplant. In dem Beispiel aus Bild 3 wurde dafür in allen Fällen der Dubinspfad verwendet. Nachdem B_2 generiert wurde, wird die aus B_1 und B_2 zusammengesetzte Notfallflugbahn an den Trajektorienplaner übergeben und nach der Erklärung in Unterabschnitt 2.2 weiterverarbeitet.

Solange kein Fehler auftritt, verschiebt sich der Planungshorizont sukzessive bis zum Ende der Missionsflugbahn. Es wird periodisch eine neue Notfallflugbahn geplant, bevor das Fluggerät die Offset-Position erreicht hat. In diesem Fall wird das Fluggerät nie auf die B_2 Bahn wechseln, sondern immer auf der initialen Missionsflugbahn bleiben. Sollte jedoch ein Fehler auftreten, kann auch keine Verschiebung des Planungshorizontes vorgenommen werden und es wird automatisch die zuletzt geplante Notfallflugbahn zur weiteren Steuerung des Fluggeräts verwendet. Da diese Flugbahn an einem Sicherheitspunkt endet, erfolgt an dieser Position die Landung des Fluggeräts. In Bild 4 ist dies beispielhaft dargestellt. Zum Zeitpunkt t_0 wird eine Notfallflugbahn berechnet. Dann kommt es zu einem Fehler und es kann keine neue Notfallflugbahn berechnet werden. Ab diesem Zeitpunkt bewegt sich das Fluggerät entlang der bei $t = t_0$ berechneten Notfallflugbahn bis zum Sicherheitspunkt (siehe Bild 4b und Bild 4c).

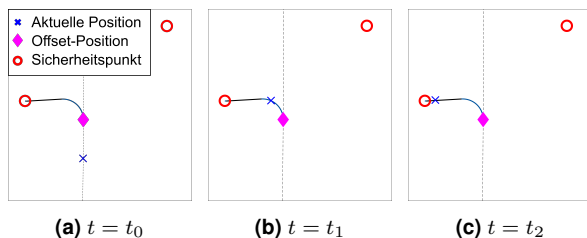


BILD 4. Position des Fluggerätes auf der Notfallflugbahn für verschiedene Zeitpunkte t

Um den Wechsel auf eine Notfallflugbahn bei einem einmaligen Fehler zu verhindern, wird die oben eingeführte Offset-Position verwendet. Der Bahnplaner und der Sicherheitsplaner werden mit einer Frequenz von 1Hz ausgeführt. Falls ein Fehler auftritt und eine Funktion ein weiteres Mal aufgerufen oder eine Komponente neu gestartet werden muss, wird zusätzliche Zeit benötigt. Darüber hinaus muss bei der Datenübertragung über die serielle Schnittstelle zwischen den Systemkomponenten mit einer erhöhten Fehlerwahrscheinlichkeit gerechnet werden. Unter der Annahme, dass eine Datenübertragung ein- oder zweimal fehlschlägt und die Daten ein weiteres Mal gesendet werden, wird eine Zeit von zwei bis drei Sekunden für diesen Vorgang eingeplant. Ein Zeitrahmen von fünf Sekunden ermöglicht daher eine neue Berechnung inklusive Übertragung der Daten. Somit wird verhindert, dass durch einen einmaligen Fehler,

der durch einen Komponentenneustart behoben wird, ein Flug zu einem Sicherheitspunkt eingeleitet wird. Falls ein Fehler aufgetreten ist, aber durch einen Neustart einer Komponente behoben werden konnte, gibt es keinen Grund, zum Sicherheitspunkt zu fliegen. Dies verhindert, dass das Fluggerät direkt zu einem Sicherheitspunkt fliegt, obwohl kein permanenter Fehler vorliegt. Sobald sich das Fluggerät auf der Notfallflugbahn befindet, kann nicht mehr zur initialen Missionsflugbahn gewechselt werden.

3. GENERIERUNG VON NOTFALLFLUGBAHNEN

Im Folgenden wird die Generierung der Notfallflugbahn genauer erklärt. Der Startpunkt der Notfallflugbahn ist die Offset-Position und der Endpunkt ist ein Sicherheitspunkt. Für die Flugbahnplanung können beliebige Flugbahnplanungsmethoden verwendet werden. Im Rahmen dieser Arbeit werden der in Unterabschnitt 2.3 vorgestellte Dubinspfad und der COWEX Algorithmus genutzt.

3.1. Anforderungen an die Sicherheitspunkte

Bevor der Ablauf der Notfallflugbahnplanung erklärt wird, werden zuerst die Anforderungen an die Sicherheitspunkte erläutert, die den Endpunkt jeder Notfallflugbahn darstellen.

An einem Sicherheitspunkt muss ein sicherer Schwebeflug oder eine sichere und hindernisfreie Landung möglich sein. Aus diesem Grund wird zunächst in einem Preprocessing überprüft, welche Sicherheitspunkte innerhalb des Geofences und außerhalb der Hindernisse liegen. Der vermeintliche Widerspruch, dass ein Sicherheitspunkt außerhalb des Geofences und damit gegebenenfalls in einem nicht sicheren bzw. nicht erlaubten Fluggebiet liegt, kann zum einen durch die beliebige Definition des Geofences durch den Anwender, aber auch z. B. durch temporäre Flugverbotszonen im realen Anwendungsfall begründet werden. Für die Hindernisse gilt diese Begründung ebenso. Hindernisse können z. B. nur für eine bestimmte Zeit gültig sein. Zudem werden Sicherheitspunkte ausgeschlossen, die sich zu nah an einem Hindernis oder am Geofence befinden. Bei der Planung der Notfallflugbahn für den Fehlerfall werden ausschließlich die zulässigen Sicherheitspunkte verwendet.

3.2. Ablauf der Notfallflugbahnplanung

In Bild 5 ist der Ablauf der Notfallflugbahnplanung grafisch in Form eines Ablaufdiagramms dargestellt, das im Folgenden detailliert beschrieben wird.

Ausgehend von der aktuellen Offset-Position werden die Sicherheitspunkte vorverarbeitet. Um zu vermeiden, dass ein Sicherheitspunkt angefliegen wird, der von der aktuellen Offset-Position im Vergleich zu anderen Sicherheitspunkten weiter entfernt liegt als andere Sicherheitspunkte, werden die zulässigen Sicherheitspunkte entsprechend der folgenden Logik sortiert: Es wird der Abstand von der aktuellen

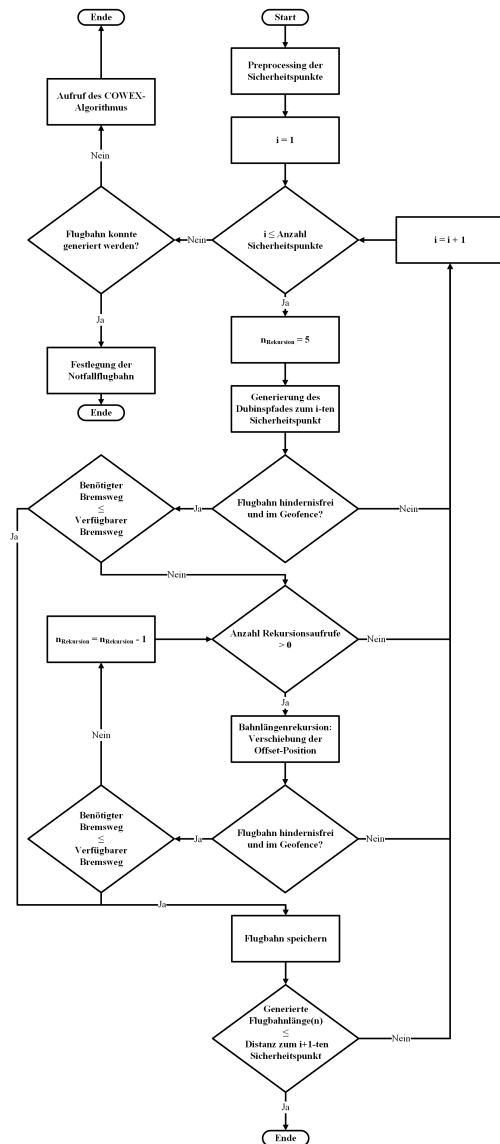


BILD 5. Ablaufdiagramm für die Generierung der Notfallflugbahn

Offset-Position zu allen zulässigen Sicherheitspunkten bestimmt und die Sicherheitspunkte dann nach zunehmender Distanz geordnet und in einer Liste gespeichert. Ziel ist es, Flugbahnen zu nahe liegenden Sicherheitspunkten zu generieren, damit die Notfallflugbahn möglichst kurz ist.

Nach der Vorverarbeitung wird durch die sortierte Liste der Sicherheitspunkte iteriert. Die im folgenden beschriebenen Schritte der Iteration zur Generierung einer Notfallflughahn werden durchgeführt, solange der Index i des aktuell ausgewählten Sicherheitspunktes kleiner ist als die Gesamtanzahl der zulässigen Sicherheitspunkte. Die Anzahl der Rekursionsaufrufe $n_{\text{Rekursion}}$ wird für jeden Sicherheitspunkt zu Beginn des Prozesses auf fünf gesetzt. Diese Variable wird für die in Unterabschnitt 3.4 beschriebene Bahnlangenrekursion benötigt.

Als erster Schritt innerhalb des Prozesses wird eine Notfallflughahn mit der Methode des Dubinspfades zum i -ten Sicherheitpunkt generiert (siehe Unterabschnitt 3.3). Nach Bestimmung der Flughahn

wird überprüft, ob die berechnete Flugbahn zum Sicherheitspunkt innerhalb des Geofences liegt und hindernisfrei ist. Wird diese Bedingung nicht erfüllt, wird die Flugbahn verworfen und es wird eine Flugbahn zum nächsten Sicherheitspunkt geplant. Befindet sich die Flugbahn im Geofence und ist zudem frei von Hindernissen, wird anschließend überprüft, ob der benötigte Bremsweg zum Sicherheitspunkt ausreichend groß ist, um am Sicherheitspunkt eine Bahngeschwindigkeit V_k von $0 \frac{m}{s}$ zu erreichen. Hierbei haben die aktuelle Geschwindigkeit und flugmechanische Beschränkungen, insbesondere die maximal mögliche Bremsverzögerung, einen großen Einfluss auf den benötigten Bremsweg: Je höher die aktuelle Bahngeschwindigkeit, desto größer ist der benötigte Bremsweg. Reicht die Notfallflugbahnlänge aus, um mit der maximal möglichen Bremsleistung mit der Bahngeschwindigkeit an der Offset-Position am Sicherheitspunkt eine Bahngeschwindigkeit V_k von $0 \frac{m}{s}$ zu erreichen, ist diese Flugbahn ein möglicher Kandidat für eine Notfallflugbahn.

Genügt der verfügbare Bremsweg nicht aus, wird eine Funktion zur Bahnlängenrekursion aufgerufen. Ziel ist es, mit dieser Funktion eine Bahngeschwindigkeit V_k von $0 \frac{m}{s}$ am Sicherheitspunkt zu erreichen, indem das Fluggerät länger als bis zur Offset-Position auf der Missionsflugbahn verbleibt. Diese Bahnlängenrekursion wird in Unterabschnitt 3.4 genauer erläutert. Gelingt es mit der Bahnlängenrekursion, die Anforderung eine Geschwindigkeit von $0 \frac{m}{s}$ und zudem hindernisfrei und innerhalb des Geofences zu bleiben, wird die Notfallflugbahn gespeichert.

Insgesamt wird jeweils die Bahnlänge von der Bahnlänge von der Offset-Position bis zum Sicherheitspunkt gespeichert, um die Notfallflugbahnen anhand ihrer Länge vergleichbar zu machen. Der beschriebene Prozess der Notfallflugbahngenerierung wird für alle Sicherheitspunkte durchgeführt, kann aber unter geeigneten Bedingungen frühzeitig abgebrochen werden. Diese möglichen Abbruchkriterien werden im Folgenden erläutert: Ist die Gesamtlänge der aktuellen Flugbahn zum Sicherheitspunkt kleiner als die Distanz von der Offset-Position zum Sicherheitspunkt, der in der sortierten Liste der Sicherheitspunkte folgt, wird der Prozess abgebrochen. In diesem Fall kann keine kürzere Flugbahn von der aktuellen Offset-Position zu einem anderen Sicherheitspunkt gefunden werden. Das zweite Abbruchkriterium ist erfüllt, wenn innerhalb der Iteration eine der vorher generierten Flugbahnen kürzer ist als die Distanz von der Offset-Position bis zum Sicherheitspunkt, die in der sortierten Liste folgt. Aus diesem Grund ist die Sortierung der Sicherheitspunkt nach dem Abstand notwendig.

Ist der Index i größer als die Anzahl aller zulässigen Sicherheitpunkte und konnte die Flugbahnplanung nicht verfrüht beendet werden, wird überprüft, ob innerhalb der Iteration eine zulässige Flugbahn gefunden wurde. Falls keine Flugbahn gespeichert wurde, konnte mit der Methode des Dubinspfades keine Flugbahn gefunden werden, die die geforderten

Bedingungen erfüllt. In diesem Fall wird im Anschluss der COWEX Algorithmus aufgerufen (siehe Unterabschnitt 3.5). Konnte im Rahmen der Iteration durch die Sicherheitspunkte mindestens eine Flugbahn generiert, aber der Ablauf der Notfallflugbahngenerierung nicht verfrüht beendet werden, tritt der Fall ein, dass der Index i größer als die Anzahl der Sicherheitspunkte ist. Das heißt, die Flugbahnen waren nicht kürzer als die Distanz zum jeweils nächsten Sicherheitspunkt bzw. nicht kürzer als die vorher generierten Flugbahnen, die die geforderten Kriterien erfüllen. Es wird nun die Flugbahn als Notfallflugbahn ausgewählt, die die kürzeste Flugbahnlänge aufweist.

3.3. Notfallflugbahnplanung mit dem Dubinspfad

Eine Möglichkeit, eine Notfallflugbahn zum Sicherheitspunkt zu generieren, ist die Flugbahnplanung mit der Methode des Dubinspfades. Wie bereits in Unterabschnitt 2.4 beschrieben wurde, wird die Flugbahn nicht von der aktuellen Position des Fluggerätes geplant, sondern von der Offset-Position. Die Bestimmung der Position, der Ausrichtung und der minimale zulässigen Kurvenradius am Startpunkt erfolgt durch die Auswertung der Flugzustandsplanung. Mithilfe dieser Auswertungsfunktion können die benötigten Variablen an der Offset-Position berechnet und zur weiteren Nutzung verwendet werden. Als Endpunkt des Dubinspfades wird der ausgewählte Sicherheitspunkt definiert. Es wird hier durch die in Unterabschnitt 3.2 sortierte Liste der Sicherheitspunkte iteriert.

Nach Bestimmung der Flugbahn wird entsprechend der Erklärung aus Unterabschnitt 3.2 bzw. aus Bild 5 die generierte Flugbahn bewertet und entschieden, ob die Flugbahn die definierten Kriterien erfüllt oder nicht.

In Bild 6 ist eine mit dem Dubinspfad generierte Notfallflugbahn abgebildet. Der Geofence ist in grün dargestellt und trennt den zulässigen (weiß) vom unzulässigen Flugbereich (grau) ab. Die Hindernisse und ihr dazugehöriger Sicherheitsbereich sind mit Kreisen in schwarz bzw. grau dargestellt. Die Sicherheitspunkte sind mit roten Kreisen, die Offset-Position mit einer magentafarbenen Raute und die aktuelle Position des Fluggerätes mit einem blauen Kreuz gekennzeichnet.

In dem dargestellten Szenario kann die Suche nach einer geeigneten Notfallflugbahn verfrüht beendet werden, da die Länge der berechneten Notfallflugbahn mit dem Dubinspfad kleiner ist als die Distanz zum nächsten Sicherheitspunkt.

3.4. Anpassung der Notfallflugbahn durch Bahnlängenrekursion

Wie bereits in Unterabschnitt 3.2 erwähnt wurde, wird die rekursive Verschiebung der Offset-Position nur dann aufgerufen, wenn die generierte Flugbahn mit dem Dubinspfad innerhalb des Geofences liegt, aber der verfügbare Bremsweg zum Sicherheitspunkt nicht zum Abbremsen auf $0 \frac{m}{s}$ am Sicherheitspunkt

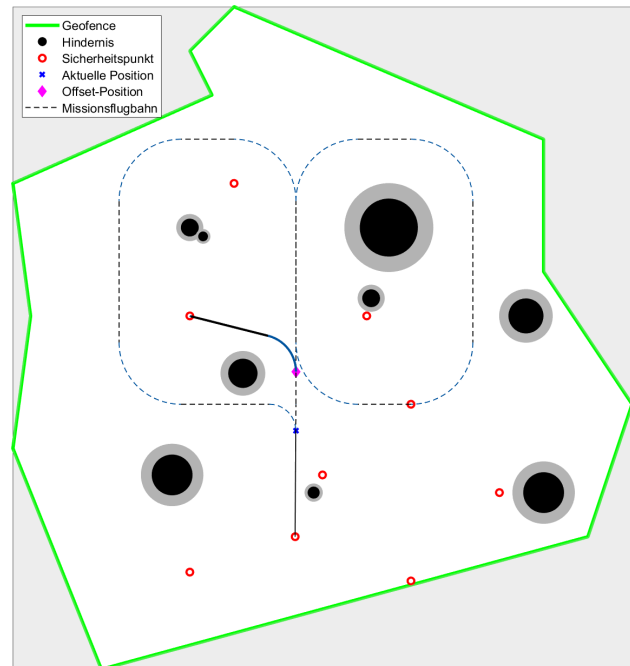


BILD 6. Beispielnotfallflugbahn generiert durch den Dubinspfad

ausreicht. Folglich ist der benötigte Bremsweg größer als der verfügbare Bremsweg. Die Anpassung der Flugbahn mithilfe der Bahnlängenrekursion wird ausschließlich für die Anpassung der durch die mit dem Dubinspfad generierten Notfallflugbahnen verwendet und ist auch Teil des Schaubildes in Bild 5. Für die Bahnlängenrekursion wird die Variable $n_{\text{Rekursion}}$ benötigt, die bei der Iteration durch die Sicherheitspunkte für jeden Sicherheitspunkt auf fünf gesetzt wird. Mit dieser Variable wird die maximale Anzahl der Rekursionsaufrufe für einen Sicherheitspunkt festgelegt. Die Anzahl der Rekursionsaufrufe auf fünf zu begrenzen stellt einen Kompromiss aus der benötigten Rechenzeit und einer sinnvollen Verschiebung der Offset-Position dar.

Die Idee hinter der Bahnlängenrekursion besteht darin, länger als bis zur Offset-Position auf der Missionsflugbahn zu verbleiben. Ausgehend von der verschobenen Offset-Position erfolgt dann die in Unterabschnitt 3.3 beschriebene Notfallflugbahnplanung mit dem Dubinspfad. Durch die Verschiebung der Offset-Position verändert sich auch der Startpunkt des Dubinspfades und somit auch der entsprechende Dubinspfad. Für diesen neu generierten Dubinspfad werden dieselben Kriterien wie für eine zulässige Notfallflugbahn in Unterabschnitt 3.3 überprüft. Falls die generierte Flugbahn nicht hindernisfrei ist oder den Geofence verletzt, wird die Rekursionsfunktion verlassen und zum nächsten Sicherheitspunkt übergegangen. Ist die Flugbahn innerhalb des Geofences, aber der benötigte Bremsweg ist immer noch kleiner als der verfügbare Bremsweg, wird die Anzahl an möglichen Rekursionsaufrufen um eins verringert und die Bahnlängenrekursion ein weiteres Mal aufgerufen. Ist die maximale Anzahl an Rekursionsaufrufen erreicht oder wurde eine Flugbahn im

Rahmen des Rekursionsaufrufes gefunden, wird die Bahnlängenrekursion beendet.

In Bild 7 ist beispielhaft eine mit der Bahnlängenrekursion und Dubinspfad generierte Notfallflugbahn abgebildet. Das schwarze Dreieck stellt die durch die Rekursion verschobene Startposition des Dubinspfades dar. Da diese Position auf der Missionsflugbahn liegt, kann über die Vorausplanung der Flugzustände die Position und Ausrichtung des Fluggeräts an dieser Position bestimmt und als Eingangsparameter für den Dubinspfad genutzt werden.

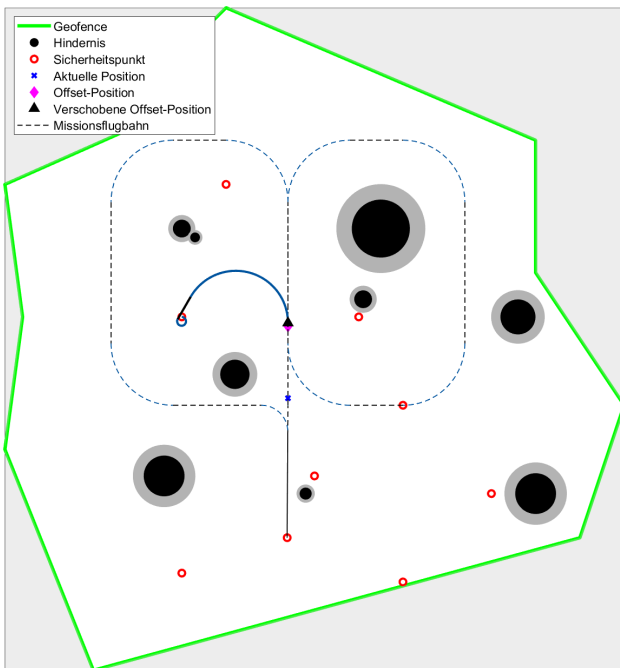


BILD 7. Beispielnotfallflugbahn generiert durch den Dubinspfad bei Bahnlängenrekursion

Auch bei der Bahnlängenrekursion muss die Länge der resultierenden Notfallflugbahn bestimmt werden. Bei der Bahnlängenrekursion muss zu der Notfallflugbahnlänge die Distanz, die von der Offset-Position länger auf der Missionsflugbahn verbracht wurde, addiert werden. Dies ist erforderlich, um Notfallflugbahnen, die mit bzw. ohne rekursive Verschiebung der Offset-Position generiert wurden, vergleichbar machen zu können. Die Entscheidung, welche der zulässigen Notfallflugbahnen ausgewählt wird, wird anhand der Bahnlänge getroffen.

3.5. Notfallflugbahnplanung mit dem COWEX Algorithmus

Falls mit dem Dubinspfad inklusive der Bahnlängenrekursion keine Flugbahn zum Sicherheitspunkt gefunden wurde, wird der COWEX Algorithmus zur Generierung einer Notfallflugbahn genutzt.

Wie in Unterabschnitt 2.3 beschrieben, benötigt der COWEX Algorithmus die Eingabeparameter Start- und Endpunkt mit zugehöriger Ausrichtung, den minimalen Kurvenradius, die Hindernisse, die flugmechanischen Beschränkungen, die Zielgeschwindigkeit am Endpunkt und den Geofence.

Anhand der Eingabeparameter ist zu erkennen, dass der COWEX Algorithmus die Bahngeschwindigkeit am Endpunkt, den Geofence und die Hindernisse bei der Bahnplanung bereits berücksichtigt. Eine zusätzliche Überprüfung, ob der Bremsweg auf eine Bahngeschwindigkeit V_k von $0 \frac{m}{s}$ ausreichend ist und ob die Flugbahn frei von Hindernissen und innerhalb des Geofences verläuft, erfolgt innerhalb des Algorithmus und bedarf keiner zusätzlichen Kontrolle. Der Startpunkt entspricht der Offset-Position und der Endpunkt dem jeweiligen Sicherheitspunkt. Die Reihenfolge, in der Flugbahnen zu den jeweiligen Sicherheitspunkten geplant werden, ist identisch zu dem in Unterabschnitt 3.2 beschriebenen Ablauf beim Aufruf des Dubinspfades. Es wird zuerst versucht, Flugbahnen zu den Sicherheitspunkten zu generieren, die von der Offset-Position die geringste Distanz aufweisen. Anders als bei der Verwendung des Dubinspfades wird die Generierung weiterer Flugbahnen abgebrochen, sobald eine Flugbahn gefunden wurde, die die Kriterien zur Einhaltung des Geofences, eine Bahngeschwindigkeit V_k von $0 \frac{m}{s}$ und zudem hindernisfrei ist, erfüllt. Grund dafür sind die längeren Laufzeiten des Algorithmus im Vergleich mit der Berechnungsdauer eines Dubinspfades.

Im Rahmen dieser Arbeit wird davon ausgegangen, dass mithilfe des COWEX Algorithmus immer eine zulässige Flugbahn unter Berücksichtigung der Umweltbedingungen in Form des Geofences und von Hindernissen und der Forderung einer Bahngeschwindigkeit V_k am Sicherheitspunkt von $0 \frac{m}{s}$ gefunden werden kann.

In Bild 8 ist eine mit dem COWEX Algorithmus generierte Flugbahn dargestellt.

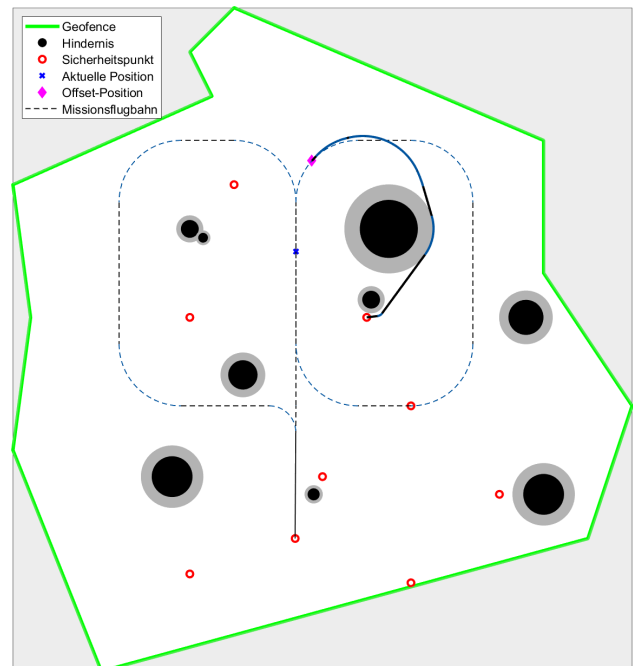


BILD 8. Beispielnotfallflugbahn generiert durch den COWEX Algorithmus

In dem dargestellten Szenario konnte mit dem Dubinspfad keine hindernisfreie Flugbahn, die zudem

die Geofencegrenzen einhält, gefunden werden. Aus diesem Grund wurde der COWEX Algorithmus zur Planung der Notfallflugbahn aufgerufen. Es ist zu erkennen, dass die Notfallflugbahn im Gegensatz zu den mit dem Dubinspfad generierten Flugbahnen aus mehr als drei Bahnelementen besteht. Zu dem von der Offset-Position näherliegenden Sicherheitspunkt konnte durch den COWEX Algorithmus keine Flugbahn generiert werden, die die geforderten Bedingungen erfüllt.

In Bild 9 sind beispielhaft zwei nicht zulässige Dubinspfade zu den zwei naheliegendsten Sicherheitspunkten abgebildet. Ein Dubinspfad verletzt hierbei den Geofence, während der Dubinspfad zum anderen Sicherheitspunkt durch ein Hindernis verläuft.

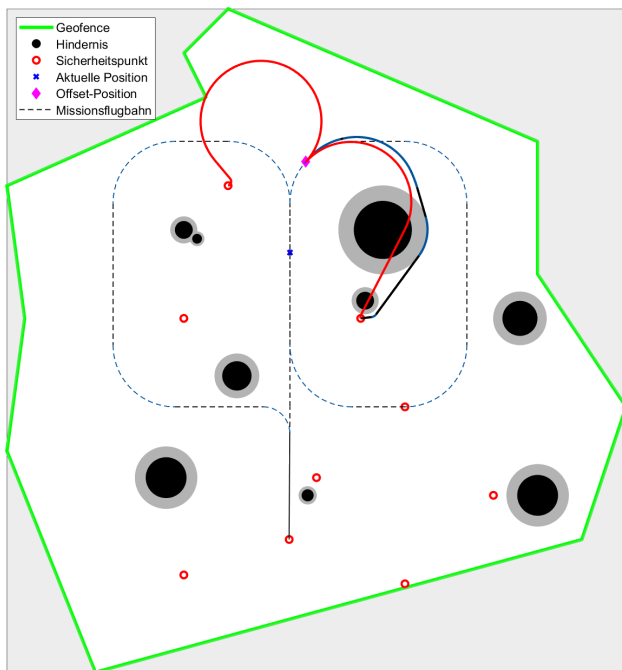


BILD 9. Beispielflugbahn generiert durch den COWEX Algorithmus und nicht zulässige Notfallflugbahnen generiert durch Dubinspfade

Sobald eine Flugbahn gefunden wurde, wird diese als Notfallflugbahn festgelegt und es werden keine Vergleiche zu Flugbahnen zu anderen Sicherheitspunkten gemacht, da die Berechnungszeit des COWEX Algorithmus im Vergleich zum Dubinspfad hoch ist.

4. ZUSAMMENFASSUNG

Im Rahmen dieser Arbeit wurden fehlertolerante Verfahren zur Einhaltung des Geofences für unbemannte Kippflügelflugzeuge bei Fehlern im Flugführungssystem entwickelt und implementiert. Durch Systemdefekte oder -ausfälle können Fehler in der Flugführung des unbemannten Fluggeräts auftreten, sodass die Missionsflugbahn nicht wie geplant abgeflogen werden kann. Um auch in solchen Fällen den Verbleib des Fluggeräts im zulässigen Fluggebiet zu garantieren, wurde das bereits bestehende Flugführungssystem um den Sicherheitsbahnplaner

erweitert, in dem Notfallflugbahnen zu definierten Sicherheitspunkten generiert werden.

Die Generierung der Notfallflugbahn erfolgt periodisch ausgehend von den jeweiligen Flugzuständen auf der Missionsflugbahn bis zu einem Sicherheitspunkt. Dabei werden die flugmechanischen Beschränkungen des Fluggeräts berücksichtigt. Diese Notfallflugbahn wird automatisch abgeflogen, sobald ein Fehler im Flugführungssystem vorliegt und keine neue Flugbahn mehr an den Trajektorienregler gesendet oder das Abfliegen der Notfallflugbahn durch den Anwender kommandiert wird. An einem Sicherheitspunkt ist ein sicherer Schwebeflug oder eine sichere Landung möglich. Die Notfallflugbahn wird mithilfe von zwei verschiedenen Flugbahnplanungsmethoden generiert, die jeweils die flugmechanischen Beschränkungen des Fluggeräts in der Planung berücksichtigen.

Zunächst wurde die Logik des Sicherheitsbahnplaners erläutert. Im Anschluss wurden die beiden unterschiedlichen Ansätze zur Bahnplanung vorgestellt und ihre Unterschiede diskutiert sowie beispielhaft Notfallflugbahnen für verschiedene Szenarien gezeigt. In beiden Verfahren werden die flugmechanischen Beschränkungen berücksichtigt. Darüber hinaus wird sichergestellt, dass die Notfallflugbahn hindernisfrei ist, innerhalb des Geofences verläuft und eine Bahngeschwindigkeit V_k am Sicherheitspunkt von $0 \frac{m}{s}$ erreicht werden kann.

Die vorgestellten Verfahren wurden in Simulationen getestet und es wurde gezeigt, dass es möglich ist, mit beiden Methoden zulässige Notfallflugbahnen zu generieren und abzufliegen. Das Kippflügelflugzeug folgte in den Simulationen präzise der vorgegebenen Notfallflugbahn und war in der Lage, am Sicherheitspunkt eine vertikale Landung durchzuführen.

Kontaktadresse:

koenig@fsd.rwth-aachen.de

Literatur

- [1] Sebastian David Seitz, Philipp Hartmann, Isabelle Barz, and Dieter Moormann. Fault-Tolerant Flight Guidance System for UAS Enabling Full U-space Integration. In *AIAA SCITECH 2022 Forum : January 3-7, 2022, San Diego, CA & Virtual*, pages AIAA 2022–1146, Reston, VA, Jan 2022. AIAA Science and Technology Forum and Exposition, AIAA SciTech Forum 2022, San Diego, CA (USA), 3 Jan 2022 - 7 Jan 2022, American Institute of Aeronautics and Astronautics. Hybride Konferenz. DOI: 10.2514/6.2022-1146.
- [2] Lennart Danielmeier, Sebastian David Seitz, Isabelle Barz, Philipp Hartmann, Max Hartmann, and Dieter Moormann. Modified Constrained Wavefront Expansion Path Planning Algorithm for Tilt-Wing UAV. In *2022 International Conference on Unmanned Aircraft Systems (ICUAS) : 21-24 June 2022 / [technical co-sponsors: IEEE; IEEE Con-*

trol Systems Society; IEEE Robotics & Automation Society; MCA, Mediterranean Control Association], pages 676–685, Piscataway, NJ, Jun 2022. 2022 International Conference on Unmanned Aircraft Systems, Dubrovnik (Croatia), 21 Jun 2022 - 24 Jun 2022, IEEE. Datenträger: USB-Stick. DOI: [10.1109/ICUAS54217.2022.9836126](https://doi.org/10.1109/ICUAS54217.2022.9836126).

- [3] Marten Schütt, Tobias Islam, Philipp Hartmann, and Dieter Moormann. Scalable Design Approach to Analyze Flight Mechanical Performance for Tilt-wing UAVs; 31st. In *[International Council of Aeronautical Sciences, ICAS, 2018-09-09 - 2018-09-14, Belo Horizonte, Brazil]*. International Council of Aeronautical Sciences, Belo Horizonte (Brazil), 9 Sep 2018 - 14 Sep 2018, Sep 2018.
- [4] L. E. Dubins. On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents. *American Journal of Mathematics*, 79(3):497–516, 1957. ISSN: 00029327, 10806377.
- [5] Isabelle Barz, Lennart Danielmeier, Sebastian David Seitz, and Dieter Moormann. Kollisionsfreie Flugbahnen für Kippflügelflugzeuge auf Basis des Wavefront-Algorithmus unter Berücksichtigung flugmechanischer Beschränkungen. In *Luft- und Raumfahrt - Verantwortung in allen Dimensionen : Deutscher Luft- und Raumfahrtkongress 2020 : 1.-3. September 2020, Online : Publikationen zum DLRK 2020 / DGLR*, pages 530192, 1–9, Bonn e.V., Sep 2022. 69. Deutscher Luft- und Raumfahrtkongress, online, 1 Sep 2020 - 3 Sep 2020, Deutsche Gesellschaft für Luft- und Raumfahrt - Lilienthal-Oberth. Konferenzort: Aachen, Germany. DOI: [10.25967/530192](https://doi.org/10.25967/530192).