

NOTLANDEASSISTENT FÜR URBAN AIR MOBILITY LUFTFAHRZEUGE

F. Settele, F. Mothes, M. Dehm, A. Knoll

Hochschule München, Labor für Automatisierung und Dynamik, Lothstr. 64, 80335 München

Zusammenfassung

Im vorliegenden Paper wird ein Trajektorienplanungsalgorithmus für Urban Air Vehicles mit Flügeln vorgestellt. Der Flugpfad wird approximiert durch eine endliche Zahl quasi-stationärer Kreis- und Geradensegmente im dreidimensionalen Raum. Das zugrunde liegende numerische Optimierungsproblem, für dessen Initialisierung ein zweidimensionaler Dubins-Pfad herangezogen wird, wird schließlich mittels NLP-Solver gelöst. Der Algorithmus wird anhand des Szenarios eines vollständigen Antriebsausfalls beschrieben und demonstriert. Dabei werden drei verschiedene Konstellationen aus Start- und Zielpunkten sowie drei verschiedene Kostenfunktionen betrachtet. Durch geschickte Annahmen, die im hinterlegten Modell getroffen werden, und dank der Initialisierung mit Dubins-Pfad, welche das nichtholonom Flugsystem annähernd abbilden, konvergiert der Algorithmus schnell und zuverlässig.

Keywords

Geometrische Pfadplanung; Urban Air Mobility; Notlandeassistent; Urban Air Mobility; Nichtlineare Optimierung; rechen-effiziente Trajektorienplanung

1. EINLEITUNG

Urban Air Mobility Vehicles sind für kürzere Strecken in überwiegend geringer Höhe ausgelegt. Im Fehlerfall bleibt daher nur wenig Zeit, eine fliegbare Trajektorie zu einem geeigneten Notlandefeld zu ermitteln. Mithilfe zeitdiskreter Trajektorienoptimierung können mittels modellartiger Abbildung des physikalischen Verhaltens komplexe Trajektorien ermittelt werden. Deren Konvergenzfähigkeit und -geschwindigkeit hängt allerdings stark von der Güte der Initialisierung, auch „Initial Guess“ oder „Anfangswertschätzung“ genannt, und der Problemkomplexität ab. Um sowohl schnell als auch zuverlässig zu einer Konvergenz des Problems zu gelangen, muss das entsprechende Optimierungsproblem vereinfacht und ein sinnvoller Initial Guess gefunden werden.

Das vorliegende Paper stellt eine Methode der effizienten Trajektorienoptimierung vor. Hierzu wird die Trajektorie von der aktuellen Position zum Notlandefeld auf geometrische Formen reduziert, repräsentiert im vorliegenden Fall durch Kreis- und Geradensegmente. Die als konstant angenommene Flugleistung im Gleitflug wird über eine quadratische Polare angenähert. Der Initial Guess ist die jeweilige Lösung des zweidimensionalen Dubins-Pfad (DP) [1]. Das Trajektorienoptimierungsproblem wird schließlich mit Hilfe eines NLP-Solvers (`fmincon` in MATLAB) gelöst. In der Literatur finden sich zahlreiche Ansätze Landungen automatisiert durchzuführen. Hierfür müssen stets geeignete Trajektorien von der aktuellen Position zum Landefeld unter Berücksichtigung der Flugleistungen erzeugt werden. Für nicht-lineare Differentialgleichungen erzeugen [2] geeignete Vertikal-Trajektorien für die Phasen „Approach“, „Glideslope“ und „Flare“ getrennt voneinander. Für lineare und nicht-lineare Bewegungsgleichungen eines UAVs ermittelt [3] Notlande-Trajektorien mittels „NLP-Solving“ für

maximale Zeit und maximale Distanz. Die Autoren in [4] stellen zeit- und verbrauchsoptimierte Trajektorien unter Berücksichtigung von realen Anforderungen an ein eVTOL (electrical Vertical Takeoff and Landing Aircraft) vor. Zum Einsatz kommt hier ebenfalls sog. NLP-Solving. Dagegen verwenden die Autoren in [5] keine fest und vorab ermittelten Trajektorien als Eingang in die Trajektorien, sondern adaptiert online nur den als nächstes zu fliegenden Abschnitt der Trajektorie mittels vereinfachter Bewegungsdynamik und einer Entscheidungslogik. Diese evaluiert, ob der Landepunkt erreichbar ist. In ähnlicher Weise ermitteln die Autoren von [6] online mittels geometrischer Approximationen der real fliegbaren Flugbahnen (Trochoiden) unter Berücksichtigung des Windes.

Im folgenden Kapitel wird das Modell beschrieben, welches die Notfalltrajektorie sowie den Gleitflug approximiert. Anschließend wird das Optimierungsproblem definiert und dessen Lösung erörtert. Schließlich werden die resultierenden Ergebnisse werden gezeigt und diskutiert.

Die vorliegende Arbeit ist im Rahmen des Projektes „AUDEKI“ (Automatisierte Flugführung unterstützt durch eine Kombination intelligenter Algorithmen) entstanden.

2. MODELLIERUNG

In diesem Abschnitt wird zunächst die geometrische Darstellung der Trajektorie erklärt. Anschließend wird das quasi-stationäre mathematische Modell vorgestellt, das die Flugdynamik des eVTOLs approximiert.

Folgende Annahmen werden getroffen:

- 1) Die Trajektorie wird mithilfe von Geraden- und Kurvensegmenten vereinfacht,
- 2) konstante Krümmungsradien anstelle klothoidenähnlicher Übergängen von Geradeaus- zu Kurvenflug und

vice versa infolge masse- und trägheitsbedingter Änderungen des Querneigungswinkels und der Kurvengeometrie,

- 3) konstante Geschwindigkeit in jedem Segment anstatt massebehafteter Umwandlung zwischen potenzieller und kinetischer Energie,
- 4) abrupte Geschwindigkeitswechsel zwischen Segmenten statt realistischer Beschleunigungs- und Verzögerungsphasen,
- 5) konstante Luftdichte je Segment anstatt höhenabhängiger Variation,
- 6) Wind und Turbulenz werden vernachlässigt.

2.1. Geometrische Trajektorienrepräsentation

Im Notfall (z.B. Antriebsausfall) wird ein erreichbarer (Not-)Landeplatz und eine fliegbare Trajektorie dorthin benötigt. Da die Zeit zum Ermitteln einer geeigneten Notlandetrajektorie im Ernstfall sehr kurz sein sollte, muss ein geeignetes Verfahren kurze Berechnungszeiten aufweisen. Für eine schnelle und robuste Berechnung ist es quasi unabdingbar, gewisse Vereinfachungen zugunsten der Schnelligkeit und Robustheit in Kauf zu nehmen. Flugzeugtrajektorien können durch geometrische Formen wie Geraden, Segmente konstanter Krümmung sowie Klo-thoiden approximiert werden [7, 8]. In der vorliegenden Arbeit kommen, wie bereits erwähnt, Geraden- und Kurvensegmente zum Einsatz. Der Zustand an Start- und Endpunkt der Trajektorie (\mathbf{x}_0 bzw. \mathbf{x}_{fin}) wird über den Zustandsvektor

$$(1) \quad \mathbf{x} = \begin{bmatrix} x_n & x_e & x_{alt} & \Psi \end{bmatrix}^T,$$

mit den Größen nördliche Koordinate x_n , östliche Koordinate x_e , der Höhe x_{alt} und der Flugzeugausrichtung und Flugrichtung Ψ („Heading“) beschrieben.

Innerhalb der Flugleistungen im Gleitflug kann ein Landeplatz durch die Kombination von vier geometrischen Segmenten erreicht werden (siehe Abb. 1):

- ein Kreissegment s_1 ,
- ein lineares Segment s_2 ,
- ein weiteres Kreissegment s_3 und
- ein abschließendes lineares Segment s_4 ¹.

Jedes Kreissegment ist definiert über seinen Radius² r sowie über seine Segmentlänge l . Nach [14] hängt der Kurvenradius einer stationär geflogenen Kurve von der Geschwindigkeit V und dem Hängewinkel μ ab:

$$(2) \quad r(V, \mu) = \frac{V^2}{g \tan(\mu)}$$

Zur Vermeidung numerischer Singularitäten (für den Einsatz von Euler-Winkeln) bei $\mu = 90^\circ$ sowie zur realisierbaren Darstellung der Kurvensegmente wird der Querneigungswinkel bei der späteren Optimierung auf den Bereich $-45^\circ \leq \mu \leq +45^\circ$ begrenzt. Der Zusammenhang zwischen der Kursänderung in einem Kurvensegment $\Delta\Psi$ hängt von der Bogenlänge l und dem Radius r des Kreisbogensegments ab:

¹Das letzte lineare Segment s_4 wird zwar nicht zum Erreichen des Zielpunktes \mathbf{x}_{targ} benötigt, jedoch zur Stabilisierung des Endanflugs berücksichtigt.

²Das Vorzeichen des Radius legt hier die Drehrichtung fest.

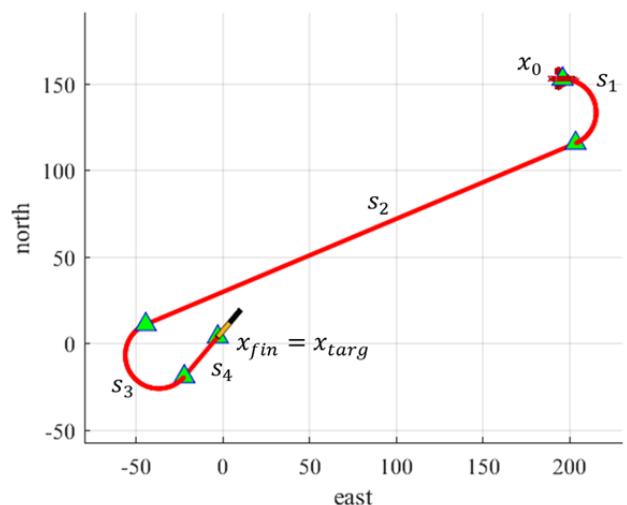


Abb. 1. Beispielhaft eine geometrische Notlandetrajektorie.

ments ab:

$$(3) \quad \Delta\Psi = \frac{l}{2\pi r} \cdot 360^\circ$$

Lineare Segmente werden durch ihre Länge l beschrieben. Alle linearen Segmente sind im Übergangspunkt tangential an die Kreissegmente angeschlossen, und das erste Kreissegment beginnt tangential zur aktuellen Flugrichtung. Ein Parametersatz \mathbf{p} definiert somit eine eindeutige geometrische (noch zweidimensionale) Trajektorie $\mathbf{T} = f(\mathbf{p}, \mathbf{x}_0)$, die trigonometrisch berechnet und dargestellt werden kann, wie in Abb. 1 dargestellt ist:

$$(4) \quad \mathbf{p} =$$

$$\left[\mu_{s_1} \quad V_{s_1} \quad l_{s_1} \quad l_{s_2} \quad V_{s_2} \quad \mu_{s_3} \quad V_{s_3} \quad l_{s_3} \quad l_{s_4} \quad V_{s_4} \right]^T$$

2.2. Flugleistungsmodell

Das verwendete Modell approximiert das mechanische Verhalten des Flugzeugs. Vor allem die Annahme fester Geschwindigkeiten in jedem Segment ermöglicht die individuelle Berechnung der abgebauten Höhendifferenz in jedem Segment.

2.3. Flugleistung

Um den Höhenverlust über alle Segmente im Gleitflug abschätzen zu können, werden quasi-stationäre Gleitflugleistungen entlang einer quadratischen, aerodynamischen Polare angenommen. Der Widerstandsbeiwert c_D hängt dabei von dem aktuellen Auftriebsbeiwert c_L über den dimensionslosen Parameter k_{aero} und den dimensionslosen Nullwiderstandsbeiwert $c_{D,0}$ ab:

$$(5) \quad c_D(c_L) = c_{D,0} + k_{aero} c_L^2$$

Der Auftriebsbeiwert c_L ergibt sich für quasi-stationär konstante Geschwindigkeit, unter Annahme kleiner Bahnwin-

kel γ , zu

$$(6) \quad c_L(q) = \frac{2 m g}{q S_{\text{ref}}} \frac{1}{\cos(\mu)},$$

mit der Flugzeugmasse m , der Flügelfläche S_{ref} und dem Staudruck $q(V) = \frac{\rho(h)}{2} V^2$. Im stationären Kurvenflug erhöht sich der notwendige Auftrieb infolge des Hängewinkels um den Faktor $\frac{1}{\cos(\mu)}$.

Die Luftdichte $\rho(h)$ wird aufgrund des kleinen betrachteten Höhenintervalls linear mit der Höhe nach [9] mit dem Luftdruck auf Meeressniveau ρ_0 und dem linearisierten Dichteabfall $\frac{d\rho}{dh}$ approximiert:

$$(7) \quad \rho(h) = \rho_0 - \frac{d\rho}{dh} h$$

Der aerodynamische Widerstand ergibt sich schließlich zu

$$(8) \quad D(V) = q(V) S c_D(c_L)$$

Der stationäre Gleitpfadwinkel im aktuellen (antriebslosen) Segment lautet

$$(9) \quad \gamma(V, h) = \arcsin \left(\frac{-D(V)}{m g} \right),$$

während die im Segment abgebaute Höhe

$$(10) \quad \Delta h(V, h) = l \sin(\gamma(V, h))$$

beträgt.

Alle relevanten Parameter sind in Tab. 1 zusammengefasst.

TAB 1. Verwendete Modellparameter.

m	435 kg
S_{ref}	10 m ²
$c_{D,0}$	0.0107
k_{aero}	0.062
ρ_0	1.225 kg/m ³
$\frac{d\rho}{dh}$	1.144×10^{-4} kg/m ⁴
g	9.81 N/kg

3. METHODEN

In diesem Abschnitt wird das zu lösende Optimierungsproblem spezifiziert, eine numerische Lösungsmethode vorgestellt und die Strategie zur Anfangswertschätzung erläutert.

3.1. Definition des Optimalsteuerungsproblems

Wie in Abschnitt 2.2 beschrieben, definiert ein eindeutiger Parametersatz \mathbf{p} (vgl. Gl. (4)) eine eindeutige Trajektorie $\mathbf{T} = f(\mathbf{x}_0, \mathbf{p})$ vom Anfangspunkt \mathbf{x}_0 zum Endpunkt $\mathbf{x}_{\text{fin}}(\mathbf{x}_0, \mathbf{p})$. Die Parameter \mathbf{p} bilden die Optimierungsvariablen.

Der Aufsetzpunkt des anvisierten Notlandeplatzes weist die Koordinaten bzw. den Zustand \mathbf{x}_{targ} (Zielzustand, vgl. (1)) auf. Gesucht wird in der Parameteroptimierung der Parametersatz, welcher die Trajektorie \mathbf{T} zu diesem Aufsetzpunkt

beschreibt. Daher wird für die Parameteroptimierung folgende Nebengleichheitsbedingung gesetzt:

$$(11) \quad \mathbf{x}_{\text{fin}}(\mathbf{x}_0, \mathbf{p}) - \mathbf{x}_{\text{targ}} = 0$$

Es werden drei verschiedene Kostenfunktionen betrachtet. Die erste Kostenfunktion J_{sp} minimiert die Gesamtpfadlänge, wobei der Index _{sp} für „shortest path“ steht:

$$(12) \quad \min_{\mathbf{p}} J_{\text{sp}} = \sum_{n=1}^4 l_{s_n}(\mathbf{p}).$$

Die zweite Kostenfunktion J_{lp} maximiert die Pfadlänge zur Erzielung eines möglichst flachen Gleitwinkels, wobei der Index _{lp} für „longest path“ steht:

$$(13) \quad \max_{\mathbf{p}} J_{\text{lp}} = \sum_{n=1}^4 l_{s_n}(\mathbf{p}).$$

Eine dritte Zielfunktion J_{mV} basiert auf der Summe der Differenzen der Geschwindigkeiten zwischen den vier Segmenten s_1, s_2, s_3 und s_4 , wobei der Index _{mV} für „minimal velocity difference“ steht:

$$(14) \quad \min_{\mathbf{p}} J_{\text{mV}} = \sum_{n=1}^3 |V_{s_{n+1}}(\mathbf{p}) - V_{s_n}(\mathbf{p})|$$

Mit dieser Kostenfunktion werden in der Flugtrajektorie die Geschwindigkeitsunterschiede zwischen den Segmenten minimiert.

Wie zuvor in Kap. 2.1 beschrieben, wird der Hängewinkel auf den Bereich $-45^\circ \leq \mu \leq +45^\circ$ begrenzt. Des Weiteren ist die Geschwindigkeit auf Werte zwischen $V_{\min} \leq V \leq V_{\max}$ limitiert. Für das letzte Geradensegment vor der Landung s_4 wird zudem die Länge auf ein Mindestmaß $l_{s_4} \geq 50\text{m}$ gehalten, um eine Stabilisierung des Landeanflugs zu ermöglichen. Diese Begrenzungen werden direkt als \mathbf{p}_{\min} bzw. \mathbf{p}_{\max} als Grenzen für den Parametervektor \mathbf{p} an den Optimierungsalgorithmus übergeben.

Zur Lösung wird ein gradientbasierter, nichtlinearer Optimierungsalgorithmus verwendet. Als Simulations- und Darstellungsumgebung dient MATLAB [10], insbesondere die Optimierungsfunktion `fmincon` von MATLAB. Zur Reduktion der Rechenzeit werden Modell und Nebenbedingungen als `mex`-Funktionen (MATLAB-Executables) kompiliert.

3.2. Initial Guess

Für den initial guess \mathbf{p}_0 (vgl. Gleichung (4)) wird in vorliegender Arbeit der DP [11] herangezogen. Der Dubins-Pfad bezeichnet den kürzesten (zweidimensionalen) Pfad zwischen Start- und Endkonfiguration (blaue bzw. grüne Pfeile), der vereinfachend ausschließlich aus maximal zwei Kreisbögen mit festem Radius (jeweils schwarze Bögen) und einem Geradensegment (rote Linien) besteht (vgl. Abb. 2).

Durch die zusätzliche Reduktion des in Kap. 3.1 beschriebenen Optimierungsproblems auf zwei laterale Dimensionen (x und y) wird der kürzeste Pfad zwischen den Punkten \mathbf{x}_0 und \mathbf{x}_{targ} ermittelt. Die in Abb. 1 gezeigte Trajektorie wird um den Abschnitt s_4 gekürzt. Mit anderen Worten wird der

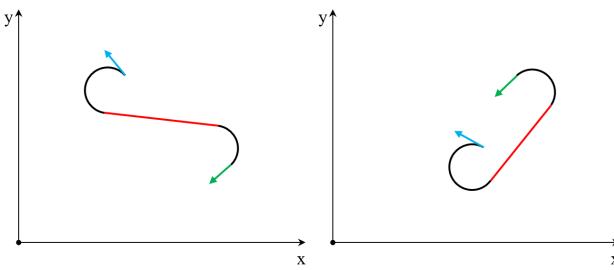


Abb. 2. Zwei beispielhafte Dubins Pfade.

Beginn der Landebahn und damit der Zielpunkt der Trajektorie um einen fixen Betrag entgegen der Landebahnrichtung nach vorn verschoben. Des Weiteren werden im Parametervektor \mathbf{p}_0 alle Geschwindigkeiten $V_1 = V_2 = V_3 = V_{\text{ini}}$ mit der Initialisierungsgeschwindigkeit V_{ini} , alle Hängewinkel $\mu_1 = \mu_2 = \mu_3 = \mu_{\text{max}}$ und alle (Bogen-)Längen l_1 , l_2 und l_3 direkt auf Ergebnis-Längen aus dem Algorithmus für den initial guess gesetzt. Bogenlänge l , Geschwindigkeit und Hängewinkel hängen über Gleichung (3) zusammen. Zur Verwendung des DP kommt die Toolbox [12, 13] zum Einsatz. Mit dem so erzeugten „initial guess“ wird schließlich die in Abschnitt 3.1 erläuterte Lösung des Optimierungsproblems initialisiert.

4. ERGEBNISSE

In diesem Abschnitt werden verschiedene Problemkonstellationen vorgestellt, um die Leistungsfähigkeit der vorgestellten Flugbahnoptimierungs-Methode zu demonstrieren. Jede Konstellation ist definiert durch einen Anfangszustand \mathbf{x}_0 und einen Zielpunkt \mathbf{x}_{targ} , für die mindestens eine gültige Trajektorienlösung ermittelt wird.

4.1. Konstellation 1

Für die erste Konstellation werden die in Tab. 2 aufgeführten Start- und Zielzustände verwendet:

TAB 2. Konstellation 1 (Einheiten [m] bzw. [°]).

	x_n	x_e	x_{alt}	Ψ
\mathbf{x}_0	-2100	250	450	110
\mathbf{x}_{targ}	4	-3	0	285

Die Fluggeschwindigkeiten in den vier Abschnitten werden $V_{\min} = 40 \frac{\text{m}}{\text{s}}$ und $V_{\max} = 150 \frac{\text{m}}{\text{s}}$ festgelegt. Für den Hängewinkel in den beiden Kreissegmenten gilt $|\mu_1|, |\mu_3| \leq 45^\circ$.

Die angewendete Kostenfunktion ist die Minimierung der Gesamtpfadlänge (shortest path) nach Gleichung (12).

In Abb. 3 ist das Ergebnis der vorgestellten Methode für Konstellation 1 zu sehen. Darin ist der DP in blau und die Ergebnistrajektorie nach der Optimierung in grün dargestellt. Dazwischen zeigen graue Linien die Zwischentrajektorien während der Konvergenz des Optimierungsverfahrens. Die Wegpunkte (dargestellt mit grünen Dreiecken mit blauer Umrandung) stellen die Verbindungspunkte zwischen den Streckensegmenten und den Aufsetzpunkt auf der Landebahn dar.

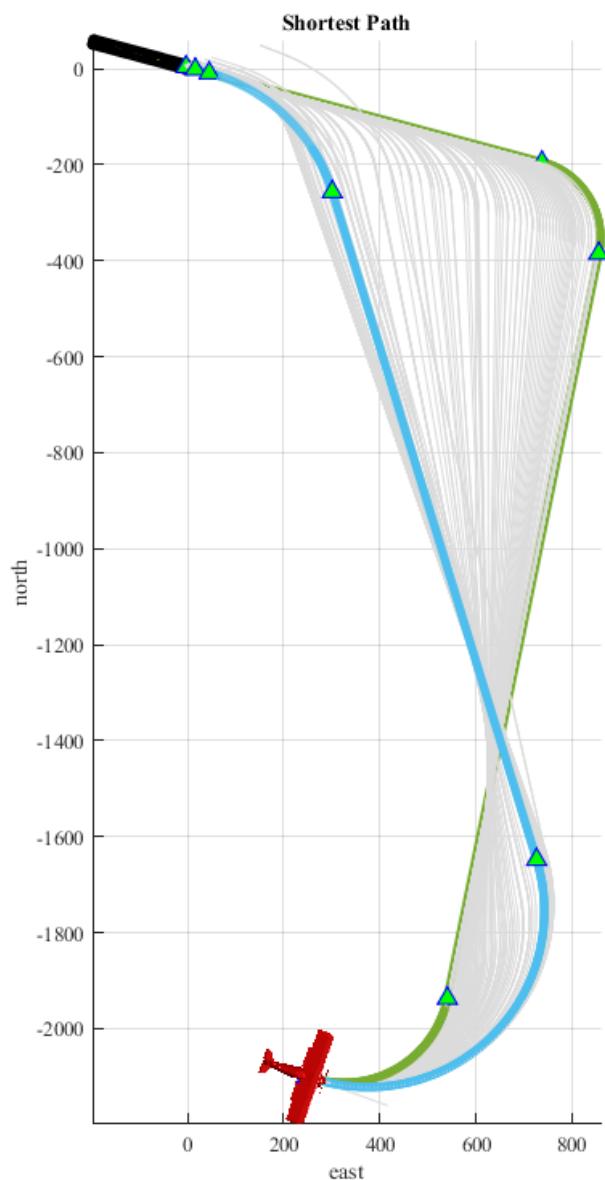


Abb. 3. Trajektorie für die Konstellation 1.

Erkennbar sind die fixen Radien des DP, die durch die angehängte Optimierung noch angepasst werden. Zudem wird der Abschnitt s_4 durch die Optimierung verlängert – wie später gezeigt wird – um Höhe abzubauen. Die Länge des DP (zweidimensional, ohne Höhenverlauf) beträgt 2,72km, die Länge der optimierten, dreidimensionalen Trajektorie schließlich 2,98km. Die Trajektorie wird durch die Optimierung vor allem im Endteil (Segmente s_3 und s_4) verlängert.

Abb. 4 zeigt den Höhenverlauf der Trajektorie, sowie die Verläufe der Steuergrößen Geschwindigkeit in allen vier Segmenten V_1, V_2, V_3, V_4 und die Hängewinkel in den beiden Kurvensegmenten μ_1 und μ_3 . Gegenüber den Kurvensegmenten weisen die beiden Geradensegmente s_2 und s_4 einen steileren Gleitwinkel auf. Der Höhenabbau erfolgt also vor allem im Geradeausflug, und folglich ist auch die Geschwindigkeit in den Geradensegmenten deutlich höher als in den Kurvensegmenten. Dabei springt die Geschwindigkeit zwischen den Kurven- und den Geradensegmenten zwischen V_{\min} und V_{\max} . Ein Beschleunigen beziehungsweise

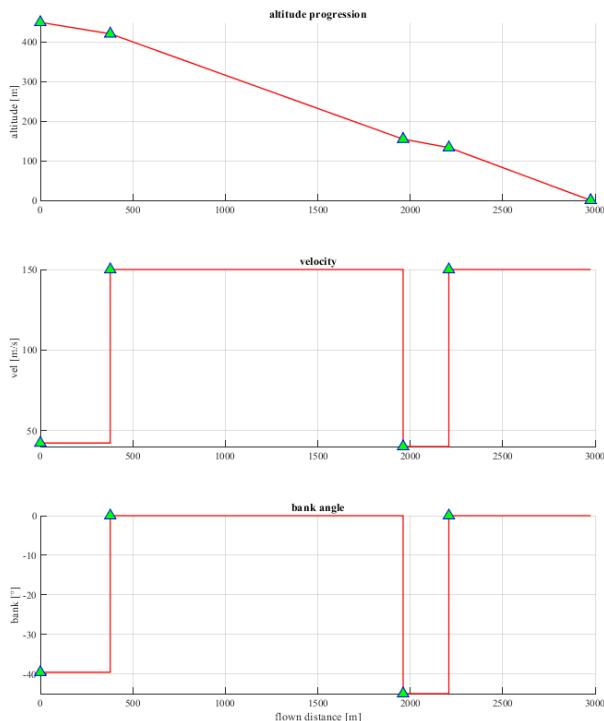


Abb. 4. Verläufe Konstellation 1.

ein Verzögern um gut $100 \frac{m}{s}$ zwischen den Segmenten, wie durch die Vereinfachung auf stationäre Segmente vorgesehen ist, bedarf in der Realität starker Höhenänderungen und ist damit in der Realität nicht für eine Notfalltrajektorie anwendbar. Der Hängewinkel in den beiden Kurvensegmenten liegt bei jeweils ca. -45° und damit nahe am Maximalwert, der für die Optimierung erlaubt ist. Für Konstellation 1 lässt sich vermuten, dass ein Kurvenflug möglichst kurz gehalten werden sollte. Die Fluggeschwindigkeit ist in den Kurven gering, der Hängewinkel groß, was mit Gleichung (2) einen kleinen Radius bewirkt. Da der effektive Anteil der Kurvensegmenten an der Gesamttrajektoriellänge gering ist, findet der Höhenabbau vor allem in den Geradensegmenten statt. Für Konstellation 1 wird im Versuch für die Ermittlung des DP rund 0,04s und für die Konvergenz der Optimierung im Mittel 0.18s mit 110 Iterationen benötigt³. Ohne Initialisierung des Problems mittels DP konvergiert das Problem nicht.

4.2. Konstellation 2

Für die zweite Konstellation werden folgende Start- und Zielzustände angenommen (vgl. Tab. 3):

TAB 3. Konstellation 2 (Einheiten [m] bzw. [°]).

	x_n	x_e	x_{alt}	Ψ
x_0	-2100	120	500	295
x_{targ}	4	-3	0	170

³Die verwendete Hardware ist dabei in Intel Core Ultra 7 mit 12 Kernen (1,7GHz), 32GB RAM und MATLAB R2024a. Die Modell- und die Nebenbedingungsfunktionen sind zur schnelleren Ausführung zu „mex“-Funktionen [10] kompiliert.

Die angewendete Kostenfunktion ist nun die Maximierung der Gesamtpfadlänge (longest path) nach (13) – gleichzusetzen mit einem Flug nahe der Fluggeschwindigkeit des besten Gleitens $V_{E_{max}}$. Schon vorab sei darauf hingewiesen, dass der DP im Gegensatz zu dieser Kostenfunktion steht. Dennoch funktioniert er gut als „initial guess“. Ohne DP konvergiert dieses Problem nicht.

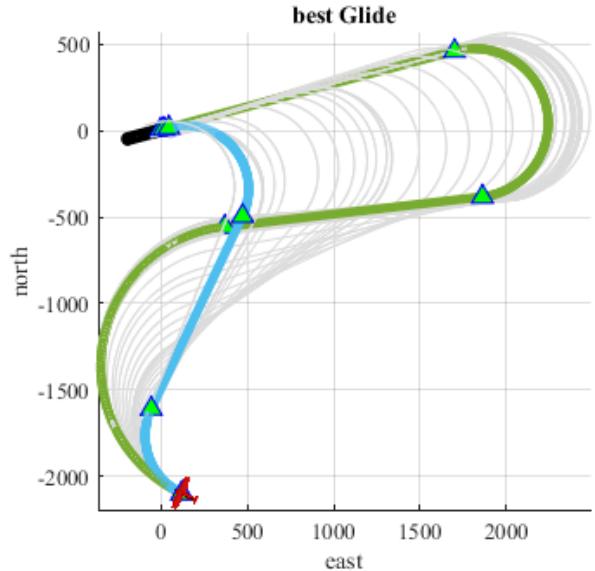


Abb. 5. Trajektorie für die Konstellation 2.

In Abb. 5 ist das Ergebnis der vorgestellten Methode für Konstellation 2 dargestellt. Es ist zu sehen, dass die Länge der Ergebnis-Trajektorie durch die Optimierung ausgehend vom DP noch deutlich vergrößert wird. Im Vergleich zu Konstellation 1 ist die Ausweitung der Trajektorie erheblich, was grundsätzlich an der veränderten Kostenfunktion in Konstellation 2 liegt. Die Länge des DP beträgt 2,73km, die Länge der maximierten Trajektorie mit 6,80km. Wie zu erwarten war, werden zugleich (und im ebenfalls im Gegensatz zu Konstellation 1) durch die Optimierung die Radien der Kurvensegmente erhöht. Vor allem die Geradensegmente s_2 und s_4 verlängern sich durch die Maximierung.

Abb. 6 zeigt den Höhenverlauf der Trajektorie, sowie die Verläufe der Steuergrößen Geschwindigkeit in allen vier Segmenten V_1, V_2, V_3, V_4 , und die Hängewinkel in den beiden Kurvensegmenten μ_1 und μ_3 . Gegenüber Konstellation 1 ist der Höhenverlauf weitestgehend linear, der Gleitwinkel in allen vier Segmenten ist annähernd gleich, der Abbau der Höhe findet also gleichmäßig über die gesamte Trajektorie statt. Sprünge in der Fluggeschwindigkeit sind weiterhin vorhanden, insgesamt aber mit deutlich geringeren Sprunghöhen. Der gesamte Geschwindigkeitsverlauf liegt zwischen $V_1 = V_2 = V_3 = V_4 = 58 \dots 65 \frac{m}{s}$ und damit nahe an der Geschwindigkeit für das beste Gleiten⁴ $V_{E_{max}}$. Diese liegt für die quadratische Polare mit den Parametern aus Tab. 1 bei ca $60 \frac{m}{s}$ (vgl. Abb. 7). Im Gegensatz zu Konstellation 1 ist die Geschwindigkeit in den Kurvensegmenten etwas höher als in den Geradensegmenten. Der Abstand von der Geschwindigkeit des besten Gleitens ist aber klein, der Einfluss auf den Gleitwinkel vernachlässigbar, wie auch im

⁴Die Maximale Strecke im Gleitflug kann bei der Fluggeschwindigkeit des besten Gleitens erlangt werden.

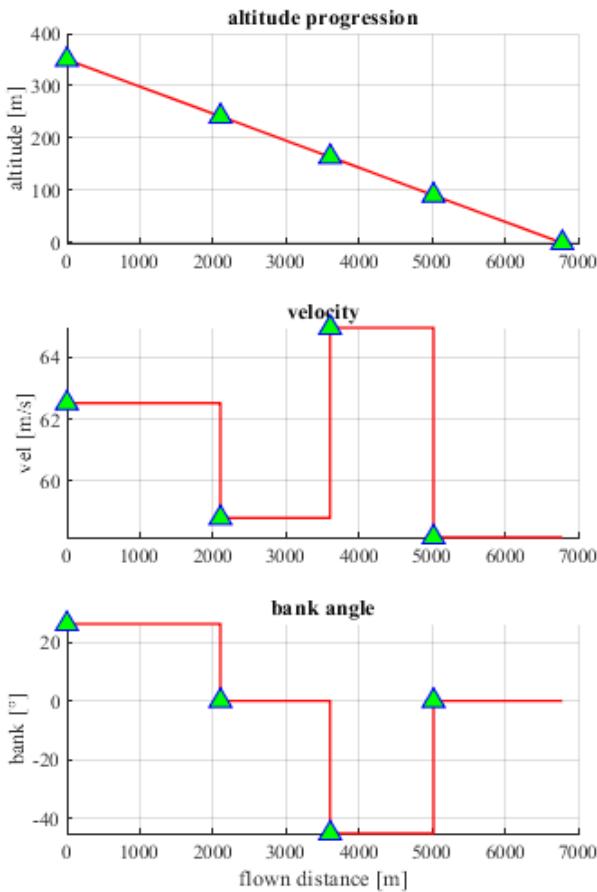


Abb. 6. Verläufe Konstellation 2.

Höhenverlauf zu sehen ist. Die Hängewinkel in den beiden Kurvensegmenten liegt zwischen $\mu_{13} = -45 \dots 26^\circ$. Durch diese etwas höhere Geschwindigkeit ist der Kurvenradius im Vergleich zu Konstellation 1 nach (2) größer.

Für Konstellation 2 wird im Versuch für die Ermittlung des DP im Mittel 0,033s und für die Konvergenz der Optimierung 0.30s mit 87 Iterationen benötigt.

4.3. Konstellation 3

Mit diesem Beispiel werden die drei Kostenfunktionen (12), (13) und (14) miteinander verglichen. Für die dritte Konstellation werden folgende Start- und Zielzustände angenommen (vgl. Tab. 4):

TAB 4. Konstellation 1 (Einheiten [m] bzw. [°]).

	x_n	x_e	x_{alt}	Ψ
x_0	-2100	250	450	285
x_{targ}	4	-3	0	285

Abb. 8 zeigt die Verläufe der Ergebnistrajektorien für die drei Kostenfunktionen *kürzester Pfad* bzw. *shortest path* (*sp*) nach (12), *längster Pfad* bzw. *longest path* (*lp*) nach (13) und *minimale Differenzen der Geschwindigkeiten* *minimum velocity difference* (*mV*) nach (14). Die mit Abstand kürzeste Trajektorie ist erwartungsgemäß die für die Kostenfunktion *sp*. Sie verläuft auf der nahezu kürzesten Verbindung zur Landebahn. Am weitesten holt die Trajek-

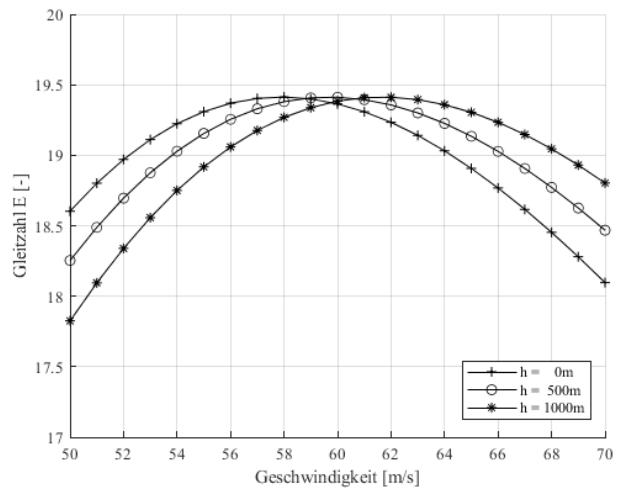
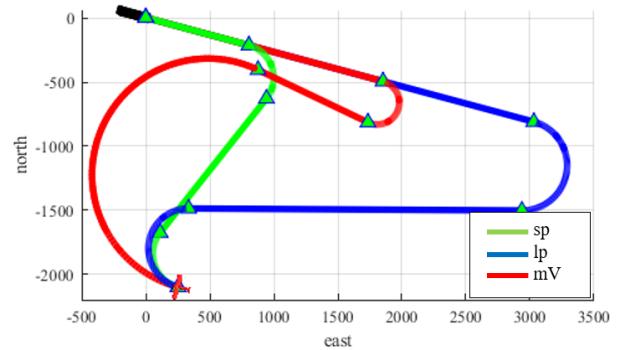
Abb. 7. Gleitzahl über der Fluggeschwindigkeit (die Fluggeschwindigkeit ist die wahre Fluggeschwindigkeit gegenüber den Luftteilchen (V_{TAS})) für verschiedene Flughöhen (0 . . . 500 . . . 1000m).

Abb. 8. Trajektorie für die Konstellation 3.

torie für *lp* aus auf dem Weg zur Landebahn. Dazwischen liegt die Trajektorie für *mV* beginnend mit einem auffallend weiten Bogensegment zu Beginn der Trajektorie. Die Gesamtlängen für den DP und die drei Trajektorien belaufen sich dabei auf (vgl. Tab. 5):

TAB 5. Konstellation 3 Vergleich der Trajektorienlängen.

Trajektorie	Länge
Dubins	2,47km
<i>sp</i>	3,22km
<i>lp</i>	7,67km
<i>mV</i>	6,46km

Abb. 9 zeigt den Höhenverlauf für die drei Vergleichstrajektorien, sowie die Verläufe der Steuergrößen Geschwindigkeit in allen vier Segmenten V_1, V_2, V_3, V_4 und die Hängewinkel in den beiden Kurvensegmenten μ_1 und μ_3 . Auch wie in Konstellation 1 sind die Geschwindigkeitssprünge für *sp* erheblich, der zugehörige Höhenverlauf entspricht mit variierendem Bahnwinkel. Die Geschwindigkeit in den Kurvensegmenten s_1 und s_3 ist gering, dagegen in den Geradensegmenten s_2 und s_4 hoch, wie schon in Konstellation 1 zu sehen war.

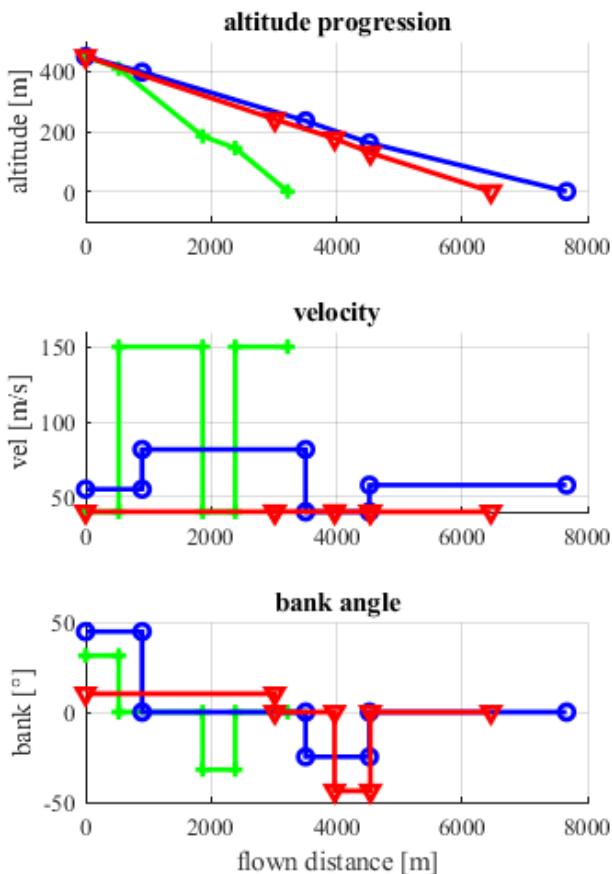


Abb. 9. Verläufe Konstellation 3.

Die Trajektorie für **lp** überwindet die längste Strecke. Der Höhenverlauf bleibt dabei weitestgehend linear, nur das letzte Segment s_4 ist etwas flacher. Geschwindigkeitssprünge sind zwar vorhanden, aber im Vergleich zur Trajektorie für **sp** deutlich geringer ausgeprägt. Außerdem befindet sich die Geschwindigkeit während des Verlaufs näher an der Geschwindigkeit des besten Gleitens $V_{E_{\max}} \approx 60 \frac{m}{s}$. Auffällig geradlinig ist der Höhenverlauf bei der Trajektorie **mV**, die Geschwindigkeit ist, wie zu erwarten, konstant über die ganze Trajektorie. Die Kostenfunktion (14) eignet sich im Einsatz folglich gut, um Geschwindigkeitssprünge zwischen allen Segmenten zu minimieren. Es sei jedoch darauf hingewiesen, dass die Geschwindigkeit sich genau am unteren Rand V_{\min} befindet. Aus Sicherheitsgründen, sollte diese Grenze angehoben werden.

Abschließend werden in Tab. 6 für Konstellation 3 und alle drei Kostenfunktionen die Berechnungs- und Konvergenzdaten verglichen. Es ist zu sehen, dass das Problem für Kostenfunktion **mV** am schnellsten konvergiert und dafür am wenigsten Iterationen benötigt werden. Die Dauer zur Ermittlung für den DP ist rein theoretisch für alle drei Kostenfunktionen gleich. Insgesamt liegt die Dauer für die vorgestellte Methode für alle drei Kostenfunktionen bei unter 0,3s. Die optimalen Flugtrajektorien können ausreichend schnell berechnet werden, um im Rahmen von schnellen Entscheidungsprozessen in Notsituationen in vertretbarer Zeit sicher angewendet werden zu können, da es sich bei einer Notlandung um eine strategische und nicht reaktive Maßnahme handelt.

TAB 6. Vergleich Performance Konstellation 3.

Konstellation	Dauer Dubins	Dauer Opti.	Iterationen
1	0,030s	0,22s	169
2	0,025s	0,16s	154
3	0,028s	0,11s	104

5. DISKUSSION

In der Arbeit wurde eine Approximation von Trajektorien durch eine Abfolge einfacher geometrischer Formen, Geraden und Kreissegmente vorgestellt. Zur Erzeugung einer Notfalltrajektorie werden dabei zwei Kurvensegmente und zwei Geradensegmente verwendet mit jeweils konstanter Geschwindigkeit und Hängewinkel. Mithilfe eines Optimierungsalgorithmus können die Parameter der approximierten Trajektorie so bestimmt werden, dass ein Zielpunkt erreicht wird. Das Optimierungsproblem muss initialisiert werden, wofür ein zweidimensionaler DP angewendet wird. Ohne die Initialisierung ist eine Konvergenz deutlich unwahrscheinlicher. Es wurden verschiedene Kostenfunktionen vorgestellt und miteinander verglichen. Dabei zeigte sich, dass eine Initialisierung mit DP das Konvergenzverhalten des Algorithmus überhaupt erst ermöglicht. Trajektorien für den kürzesten Weg zum Zielpunkt weisen starke Geschwindigkeitssprünge auf und sind daher für reale Anwendungen ungeeignet. Trajektorien für den weitesten Weg zum Zielpunkt liefern deutlich bessere, jedoch ebenfalls nicht optimale Ergebnisse. Vielversprechend ist hingegen eine Kostenfunktion zur Minimierung der Geschwindigkeitsunterschiede zwischen den einzelnen Segmenten. Für die untersuchten Konstellationen und Kostenfunktionen konnte gezeigt werden, dass der vorgestellte Algorithmus mit DP-Initialisierung ausreichend schnell ($< 0,3$ s) und stabil konvergiert und damit für die kurzfristige Erzeugung von Notlandetrajektorien gut geeignet ist.

Um die Konvergenz-Fähigkeit sicherstellen zu können, werden im nächsten Schritt Monte-Carlo-Simulationen mit zufällig erzeugten Konstellationen durchgeführt. Dabei ist von großem Interesse, zu welchen Anteilen welche der drei Kostenfunktionen zu sinnvollen Ergebnissen führt und welche der drei Kostenfunktionen für die Anwendung bevorzugt zum Einsatz kommen sollte. Des Weiteren sollen analytisch beschriebene Geschwindigkeitsübergänge zwischen den Segmenten und Übergänge des Hängewinkels untersucht und implementiert werden.

Kontaktadresse:ferdinand.settele@hm.edu**FÖRDERUNG**

Die vorliegende Arbeit ist im Rahmen des Projektes „AUDEKI“ (Automatisierte Flugführung unterstützt durch eine Kombination intelligenter Algorithmen) entstanden. Dieses ist gefördert durch „Bayerisches Staatsministerium für Wirtschaft, Landesentwicklung und Energie“:

Gefördert durch



Bayerisches Staatsministerium für
Wirtschaft, Landesentwicklung und Energie

Literatur

- [1] Dubins, L. E., “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 (1957): 497-516.
- [2] Sun, Z., Liaoni Wu, and Yancheng Y. “Automatic Landing System Design for Unmanned Fixed-Wing Vehicles via Multivariable Active Disturbance Rejection Control.” *International Journal of Aerospace Engineering* 2023.1 (2023): 9395447.
- [3] Fang, X., et al. “Emergency landing trajectory optimization for fixed-wing UAV under engine failure”, *AIAA Scitech 2019 Forum*. 2019.
- [4] Krammer, C., et al. “Requirements-based generation of optimal vertical takeoff and landing trajectories for electric aircraft”, *33rd Congress of the International Council of the Aeronautical Sciences (ICAS)*. 2022.
- [5] Liang, J., L., Wang, S., Wang, B. “Online motion planning for fixed-wing aircraft in precise automatic landing on mobile platforms.” *Drones* 7.5 (2023): 324.
- [6] Warren, M., et al. “An Automated Emergency Landing System for Fixed-Wing Aircraft: Planning and Control”, *2015 Journal of Field Robotics* 32.8 (2015): 1114-1140.
- [7] Mothes, F., “Trajectory planning in time-varying adverse weather for fixed-wing aircraft using robust model predictive control”. *Aerospace* 6.6 (2019): 68.
- [8] Mothes, F.. “Trajectory Planning in Time-Varying Adverse Weather for Fixed-Wing Aircraft applying a Model Predictive Approach”, Diss. Technische Universität München, 2021.
- [9] United States Committee on Extension to the Standard Atmosphere. “US standard atmosphere”, National Oceanic and Atmospheric [sic] Administration, 1976.
- [10] The MathWorks, Inc. “MATLAB Documentation”, version: R2024b. Zuletzt abgerufen am 08.01.2026 auf <https://de.mathworks.com/>, 2024.
- [11] Lugo-Cárdenas, I., Flores G., Salazar S., Lozano, R. “Dubins path generation for a fixed wing UAV”, 2014. International Conference on Unmanned Aircraft Systems (ICUAS 2014), Orlando.
- [12] Shkel, A. M. and Lumelsky, V. “Classification of the Dubins set”, 2001. *Robotics and Autonomous Systems* 34 179 V202, zuletzt abgerufen am 08.01.2026 auf <https://github.com/AndrewWalker/Dubins-Curves#shkel01>
- [13] Walker, A. “Dubins-curves: an open implementation of shortest paths for the forward only car”, 2008, , zuletzt abgerufen am 08.01.2026 auf <https://github.com/AndrewWalker/Dubins-Curves>.
- [14] Brockhaus, R., “Flugregelung: physikalische Grundlagen, mathematisches Flugzeugmodell, Auslegungskriterien-Regelungsstrukturen, Entwurf von Flugregelungssystemen, Entwicklungslinien”. Springer-Verlag, 2013.