

OPTIMIERTER ENDANFLUG IN EINER NOTFALLSITUATION (ENGINE OFF) MIT HILFE DES REINFORCEMENT LEARNINGS.

R. Titze, W. Schiffmann

Fakultät für Mathematik und Informatik, FernUniversität in Hagen, Deutschland
Wolfram.Schiffmann@FernUni-Hagen.de

Zusammenfassung

Ein Motorausfall während des Flugs erfordert einen Notlandeanflug auf ein geeignetes Landefeld. Während des Anflugs auf das Notlandefeld wird die Flughöhe abgebaut und soll idealerweise am Landepunkt Null betragen. Aufgrund von Unsicherheiten (bspw. Windgeschwindigkeit und -Richtung) bei der Kalkulation des Anflugs, wird häufig ein überhöhter Anflug berechnet bzw. gewählt. Diese verbleibende Sicherheitshöhe muss zu einem geeigneten Zeitpunkt abgebaut werden. Entsprechend der jeweiligen Luftfahrzeugbauweise können verschiedene Bauteile (Lande-, Bremsklappen) und Flugweisen (Slipping) zum Abbau der Flughöhe eingesetzt werden.

Im Beitrag wird beschrieben, wie mittels Reinforcement Learning (RL) ein optimiertes Landeverfahren für den oben beschriebenen Einsatzfall gefunden wird und es wird geprüft, ob ein solches Vorgehen - der Einsatz dieses verstärkenden Lernens - geeignet ist, um komplexe Aufgabenstellungen in der Luftfahrt zu lösen. Dazu wird als Algorithmus das „Deep Deterministic Policy Gradient“-Verfahren (DDPG) angewendet und ein Agent implementiert, der den kontinuierlichen Aktionsraum eines klassisch implementierten Autopiloten (PID-Regler) bis zur Flare-Position ansteuert. Zusätzlich wird ein Belohnungssystem spezifiziert, welches ein Flugverhalten fördert, das auch für den realen Flugbetrieb geeignet ist.

Die Ergebnisse zeigen, dass anreizbasiertes Lernen eine Möglichkeit ist, ein sicheres Anflugverfahren zu generieren und dieses Verfahren als Vorlage für reale Notverfahren einzusetzen. Darüber hinaus wird diskutiert, ob der Einsatz von RL grundsätzlich geeignet ist, Verfahren für komplexe Situationen zu finden bzw. diese zu optimieren ohne spezifisches Vorwissen zur Verfügung zu stellen.

1. EINFÜHRUNG

Ein Notfall während des Fluges führt aufgrund der Unsicherheit des aktuellen Flugzustands zu einer erhöhten Arbeitsbelastung und einer potenziell höheren mentalen Belastung für die Luftfahrzeugführer. Vor allem in niedriger Höhe (Privatpiloten), bei einer reduzierten Besatzungszahl (Single Pilot Cockpit) oder bei autonomen Luftfahrzeugen (LFZ) können Notlandeassistenzsysteme helfen, Komplexität zu reduzieren und Unsicherheiten zu kalkulieren. Das Ziel dieses aktuellen Forschungsgebiets ist es, Notlandungen ohne Personen- und Sachschäden mit Hilfe von solchen Assistenzsystemen durchzuführen. Dazu sind Pfadplanungen zu kalkulieren und geeignete Landefelder zu identifizieren. Eine der ersten umfassenderen Beschreibungen im Bereich Notlandeassistenz von Privatflugzeugen ist die Arbeit von [1]. Ergänzend dazu ist in [2] die Berechnung der Pfadplanung von der Notfall- bis zur Endanflugposition optimiert.

Die oben aufgeführten Arbeiten berechnen den Anflugpfad nicht bis zum Aufsetzpunkt. Aufgrund von Unsicherheiten und potenziellen Fehlprognosen (Windverhältnisse, Steuerung des LFZ, Abstraktion der Berechnungsmodelle) endet ein Endanflug typisch oberhalb des Landefelds (Sicherheitshöhe).

Diese Lücke wird in [3] geschlossen. Die Autoren beschreiben ein Verfahren mit einem mittleren Gleitwinkel, bei dem die Notlandung bei sich ändernden Windbedingungen bis zum Boden durchgeführt werden kann. Als Grundlage der Untersuchung wird dabei ein spezieller Flugzeugtyp mit Luftbremsen verwendet. Andere Möglichkeiten den Gleitpfad zu steuern (Seitenruder, Landeklappen) bleiben unberücksichtigt. In [3] wird das Setting des LFZ fest vorgegeben (Landeklappen 100%, Ruder neutral) und eine Exploration verschiedener Strategien (Choreografie verschiedener aufeinander folgender Settings) wird nicht untersucht.

In dieser Arbeit soll zunächst der Notlandeanflug um die oben angesprochenen Steuerelemente des LFZ erweitert werden. Hierbei wird untersucht, welche Kombination aus Landeklappen und Ruder in welcher Phase des finalen Anflugs zu setzen ist, damit trotz sich ändernder Windverhältnisse (Richtung, Geschwindigkeit), ein Aufsetzen am Beginn des Landefelds gewährleistet wird.

Weiterhin wird überprüft, ob eine Suche nach der richtigen Parameterkombination mittels verstärkenden Lernens zielführend eingesetzt und als generelles Verfahren abgeleitet werden kann. Der RL-Algorithmus wird als Actor-Critic mit vier künstlichen neuronalen Netzen (KNN) umgesetzt und entspricht im Grundsatz den Beschreibungen aus [4], [5], [6] und [7]. Ergänzungen des

Algorithmen sind im Kapitel 2 beschrieben und basieren auf [8].

Die Explorationseigenschaft des RL-Algorithmus unterstützt das Auffinden neuer Strategien (Gleitwinkelvarianten durch Slipping und Landeklappen).

Eine Besonderheit dieser vorliegenden Untersuchung im Zusammenhang mit RL ist die Nutzung einer zeitkontinuierlichen LFZ-Simulationsumgebung (X-plane 11).

2. VERSUCHSAUFBAU

Für die Untersuchung wird folgender Versuchsaufbau verwendet (Bild 1)

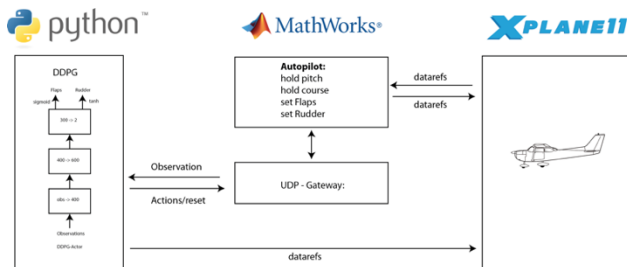


BILD 1. Versuchsaufbau

Als Simulations- und Testumgebung wird die Flugzeugsimulationssoftware X-plane 11 und ein Luftfahrzeug vom Typ Cessna 172 eingesetzt. Die Steueranweisungen für Quer-, Höhen-, Seitenruder (Ruder) und Landeklappen (Klappen) für das LFZ und die Umweltdaten/Observationsdaten (True Air Speed (TAS), Ground Speed (GS), Flughöhe, Distanz zum Zielpunkt, Vertikalgeschwindigkeit) für den RL-Algorithmus werden mit der X-plane-API (dataref) und mittels UDP-Protokoll übermittelt.

In Matlab/Simulink wurde je ein P/PD-Controller für den Pitch-Winkel und den Kurs implementiert, der die Steuerimpulse für das Quer- und Höhenruder berechnet. Damit wird der Pitch-Winkel auf 1° und der Kurs auf die Landerichtung von 50° (EDDH 05) geregelt.

Die Observationsdaten werden von Matlab/Simulink empfangen, teilweise umgerechnet (Geschwindigkeiten in horizontalen Richtungen zu geflogenem Kurs) und an eine Python-Umgebung als UDP-Pakete weitergeleitet.

Ein Algorithmus (Deep Deterministic Policy Gradient (DDPG)) bzw. der DDPG-Actor (siehe Kapitel 3) berechnet aus den Observations die entsprechenden Aktionen (Ruder und Landeklappen-Ausschlag) und sendet diese über Matlab/Simulink an X-plane.

Ziel des Versuchsaufbaus ist es, eine ähnliche Vorgehensweise wie in einer openAI-Umgebung¹ zu erhalten und neben „Observations“ und „Actions“ auch Zustände/Kommandos wie „done“ und „reset“ senden zu können.

¹ <https://openai.com/>

3. VERSUCHSABLAUF

Der Versuchsablauf besteht aus Episoden. Eine Episode beginnt indem das LFZ an den Startpunkt (Höhe: 304 m; Distanz zum Aufsetzpunkt: 1280 m) gesetzt wird. Die Windkonditionen werden zufällig und gleichverteilt gemäß Tabelle 1 initiiert. Im Höhenbereich 100 – 900 ft werden die Windkonditionen verändert. Gleichverteilt wird ein Gegen-/Rückenwind mit 0 – 5 Knoten eingestellt.

Initiale Windgeschwindigkeit	Windrichtung
[0, 10] kn	50 / 230 °

TAB 1. Initiale Windkonditionen

Dieser Versuchsablauf soll die Unsicherheit der Windvorhersage innerhalb der Berechnung des finalen Landeanflugs simulieren. Das mögliche Anflugfenster in Anlehnung an [3] ist im Bild 2 dargestellt. Die Begrenzung des Anflugfensters ist durch die möglichen minimalen/maximalen Gleitwinkel bzw. durch die Geschwindigkeit über Grund und der maximalen bzw. minimalen Sinkgeschwindigkeit (V_{smin} , V_{smax}) definiert. Außerhalb des dargestellten Anflugfensters ist mit den eingestellten Nebenbedingungen (Pitch-Winkel: 1°) und den veränderbaren Parameter (Seitenruder, Landeklappen) ein Erreichen des Aufsetzpunkts nicht mehr möglich.

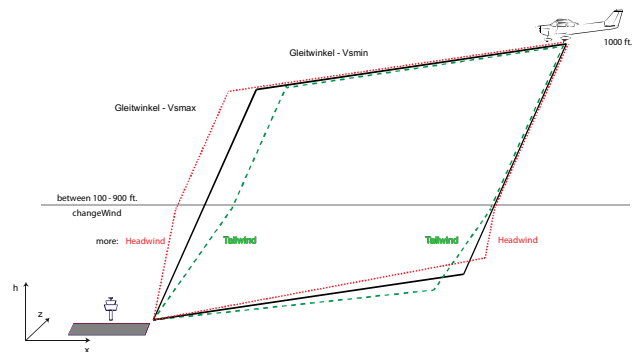


BILD 2. Versuchsablauf mit Anflugfenster

Unterschreitet das LFZ eine Flughöhe von 10 ft (über Grund) oder überfliegt das LFZ den Landepunkt, wird das LFZ erneut am Startpunkt initialisiert und eine weitere Episode wird gestartet.

In der vorliegenden Arbeit liegt der Schwerpunkt der Untersuchung in der Choreografie der Steuerelemente, außerdem soll geklärt werden, ob RL als Methode zum Optimieren von Lande- und Anflugverfahren grundsätzlich geeignet ist. Inwiefern eine Verbesserung des RL-Algorithmus möglich ist, um bspw. schneller und stabiler zu konvergieren, ist Gegenstand zukünftiger Untersuchungen.

Der Algorithmus DDPG nach [5] wird an die Problemstellung und die spezifische Entwicklungsumgebung angepasst. Die Grundelemente des Algorithmus bestehen aus den vollvermaschten „feed-forward“ KNNs: Actor, Critic, Actor-Target und Critic-Target

(Anzahl der versteckten Schichten und Neuronen siehe Bild 3), den Speicherobjekten (im Folgenden auch als replay Buffer bezeichnet) für die Transitionen (state0, reward0, action0, state1) und einem Explorationsalgorithmus nach [9]. Die Elemente sind in Python 3.6, TensorFlow 1.14.0 und Keras 2.2.4 implementiert und sind angelehnt an [10].

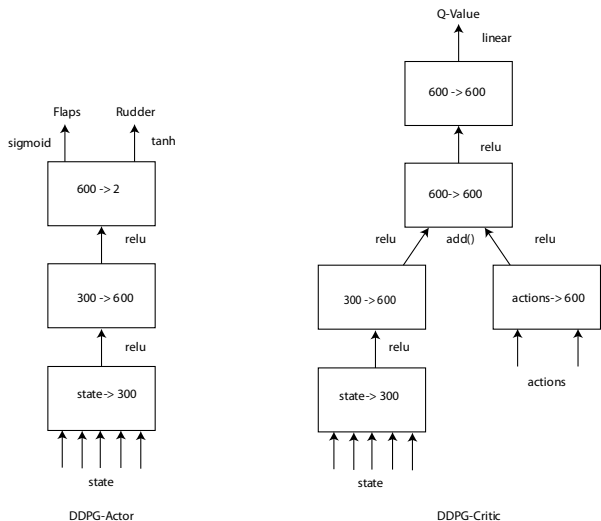


BILD 3. Struktur DDPG-Aktor und DDPG-Critic

Die Implementierung des DDPG-Aktors (im Folgende Aktor genannt) ermöglicht es, Aktionen für zwei Steuerelemente mit unterschiedlichen Wertebereichen (Ruder [-1, 1], Klappen [0, 1]) zu berechnen. Dazu sind zwei Ausgangsneuronen mit den jeweiligen Aktivierungsfunktionen (Bild 3 - links) definiert.

Der replay Buffer ist in drei Teile gegliedert (siehe auch [8]). Die Transitionen der aktuellen Episode werden in einen temporären Puffer gespeichert. Am Ende der Episode wird geprüft, wie nahe das LFZ am Landepunkt aufsetzt. Die Transitionen eines guten Versuchs (> 70% vom bisherigen Maximum) werden sowohl in den normalen Speicher als auch in einen H-Puffer (High-Quality) gespeichert, während der temporäre Speicher gelöscht wird. Für das Training der KNNs werden Transitionen aus dem H-Puffer im Verhältnis 10:1 genutzt.

Die Belohnungsfunktion des DDPG-Algorithmus ist an den Anflugrahmen wie in Bild 2 angelehnt. Innerhalb des Rahmens ist die Belohnung 1.000, außerhalb berechnet sich die Belohnung aus dem Abstand zum Rahmen begrenzt auf -10.000. Beim Zurücksetzen des LFZ auf die Startposition wird auf die Belohnungsfunktion noch das Negative der Resthöhe addiert.

Das Training der KNNs entspricht dem Vorgehen aus [4]. Das Training wird beendet, wenn 30 Episoden in Folge den Aufsetzpunkt im Umkreis von 10 m erreichen. Die Distanz zum Aufsetzpunkt ergibt sich aus der Distanz (berechnet über die X-, Z-Koordinaten der X-plane-Umgebung) und der Resthöhe (Y-Koordinate). Die Resthöhe wird über den

Tangens eines mittleren Gleitwinkels (8 °) in eine ausfliegbare Strecke umgerechnet und von der oben genannten Distanz abgezogen.

4. ERGEBNISSE

4.1. Anflüge ohne Wind

Im ersten Unterabschnitt sind die Ergebnisse von 502 Anflügen zusammengefasst. Es wird festgestellt, dass das Landefeld bzw. der Aufsetzpunkt mit einer hohen Präzision erreicht werden kann. Zusätzlich wird in Abschnitt 4.2. dargestellt, welche Einstellungen von Ruder und Landeklappen im Verlauf des finalen Landeanflugs durch den Aktor berechnet werden.

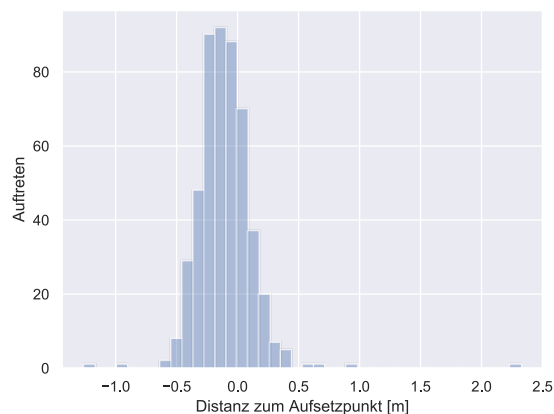


BILD 4. Verteilung: Distanz zum Ziellandepunkt ohne Wind

Anzahl Anflüge/ Statistische Werte [m]	502
mean	-0.102932
std	0.229141
min	-1.259565
25%	-0.237618
50%	-0.111753
75%	0.012543
max	2.321365

TAB 2. Statistik: Genauigkeit des Anflugs ohne Wind [m]

Im Durchschnitt verfehlt der Aktor den Landepunkt um 0.1 m (TAB 2 – mean). Das Bild 4 stellt dar, wie häufig der Ziellandepunkt verfehlt wurde und wie weit der tatsächliche Landepunkt vom Ziellandepunkt entfernt ist. In Tabelle 2 sind die entsprechenden statistischen Werte, bei der Annahme einer Normalverteilung, aufgeführt.

Die Variationsbreite der Zielgenauigkeit (Max – Min) beträgt 3,58 m und kann als qualitatives Maß für die Anflugpräzision genutzt werden. Wird die Landebahn um diesen Wert verlängert und der Ziellandepunkt verschoben, können alle Landungen im Notlandefeld erfolgen. Bezogen auf die Landestrecke einer C 172 (400 m)² und dem oben

angegebenen Gütemaß ergibt sich eine Anflugpräzision von 98,5 %.

4.2. Beschreibung der Parameter im Landeanflug (ohne Wind)

Wie in Abschnitt 4.1. beschrieben, steuert der Aktor ein LFZ mit einer hohen Genauigkeit zum Landepunkt. Dazu berechnet der Aktor die Ruder- und Landeklappenstellungen. Durch die Vorgabe eines konstanten Anstellwinkels (1°) durch den PD-Regler ändert sich bei Verstellung von Ruder und Landeklappen das Verhältnis von Auftrieb zu Widerstand und damit der Gleitwinkel des LFZ. In den Bildern 5 – a bis f ist auf der X-Achse die Distanz zum Aufsetzpunkt (siehe Bild 2) aufgetragen. Die Distanz ist in 22 äquidistante Segmente aufgeteilt und die entsprechenden Werte sind im jeweiligen Segment als Boxplot³ dargestellt.

Der Aktor kalkuliert für das Seitenruder⁴ (Bild 5 - b) einen konstanten Wert von 1. Die Landeklappen⁵ (Bild 5 – a) variieren von 0 – 0,75. In einigen Anflügen erkennt man kurzzeitig Werte von 1. Der Gleitwinkel ist im Bild 5 - d dargestellt.

Im Vergleich der Bilder 5 - a und 5 - d ist erkennbar, dass eine größere Landeklappenstellung mit einem steileren Gleitwinkel korrespondiert und vice versa. Im Verlauf des Anflugs vergrößert der Aktor die Landeklappeneinstellung und damit den Gleitwinkel. Die Werte der Sinkgeschwindigkeit (Bild 5 – e) korrespondieren mit den Werten des Gleitwinkels.

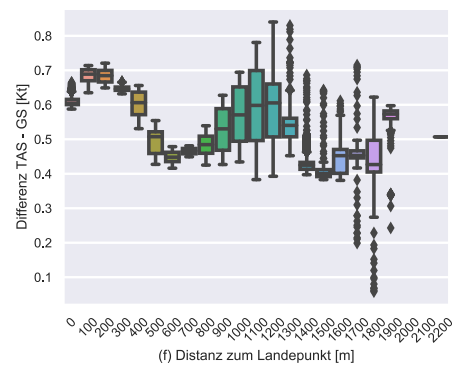
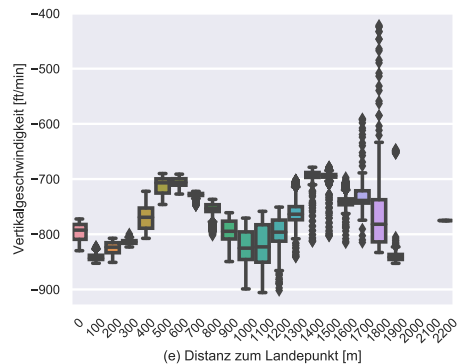
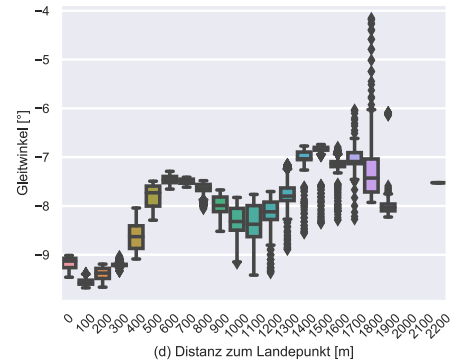
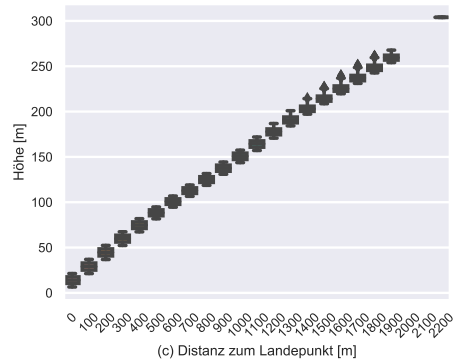
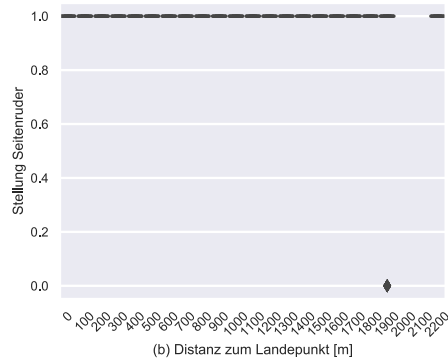
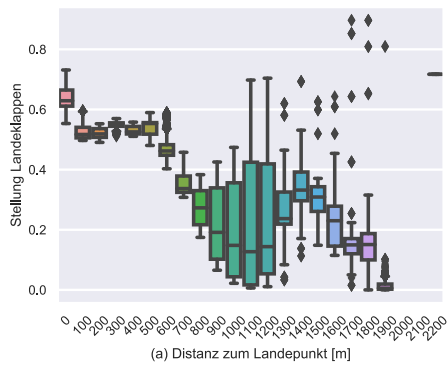


BILD 5 a-f. Parameter während des finalen Anflugs ohne Windeinfluss

³ <https://seaborn.pydata.org/generated/seaborn.boxplot.html>

⁴ Wert 1 entspricht Vollausschlag in X-plane

⁵ Wert 1 entspricht Vollausschlag in X-plane

Die 502 Anflüge bzw. die Episoden starten immer mit der gleichen Einstellung (Startpunkt) und ohne Störungen durch den Wind (Bild 5 – f). Dennoch ist in Bild 5-a auffällig, dass die Bandbreite der Klappenstellung im Distanzbereich 1400 – 800 m höher als zu Beginn oder am Ende der Episode ist. Die Ursache für die stärkere Varianz wird in dieser Arbeit nicht näher untersucht. Im nächsten Abschnitt 4.3. wird jedoch ersichtlich, dass die Bandbreite des Klappenausschlags durch die Unsicherheit der Windkonditionen eingegrenzt wird.

Die Abstraktion des Modell-LFZs durch die Simulationsumgebung könnte ursächlich für die schwankungsintensivere Ansteuerung der Landeklappen durch den Aktor sein. Innerhalb der Simulationsumgebung X-plane lassen sich die Landeklappen augenblicklich setzen und zurücknehmen. Im realen LFZ werden die Landeklappen mit Stellmotoren gefahren und das Setzen der Landeklappen beinhaltet eine zeitliche Verzögerung.

4.3. Anflüge mit Wind

Im Folgenden kann gezeigt werden, dass präzise Notlandeanflüge unter unsicheren Windbedingungen durchgeführt werden können. Dazu landet der Aktor das LFZ, trotz einer falschen Windvorhersage von bis zu 15 kn (auftreten bis zu einer Flughöhe von 100 ft), in ein entsprechendes Landefeld. Die dabei berücksichtigte Fehlprognose des Winds kann als sehr hoch interpretiert werden (mittlere Windgeschwindigkeit (in den Jahren 1981-2010) 6 in der Region Schleswig-Holstein u. Hamburg bei 5.1 m/s entsprechend 9.9 kn). Die resultierenden Windkonditionen als Differenz aus Fluggeschwindigkeit im Windfenster (TAS) und Fluggeschwindigkeit über Grund (GS) ergeben sich wie im Bild 7 - f dargestellt.

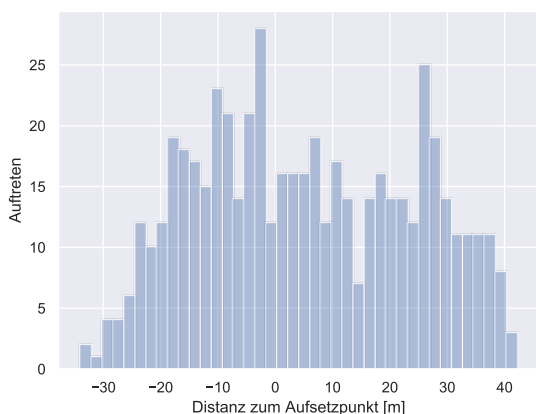


BILD 6. Verteilung: Distanz zum Ziellandepunkt mit Wind

Anzahl Anflüge/ Statistische Werte [m]	539
mean	5.304778
std	18.633797
min	-34.007470
25%	-10.072355
50%	3.652484
75%	21.977014
max	42.161449

TAB 3. Statistik: Genauigkeit des Anflugs mit Wind [m]

In Bild 6 wird die Anzahl des Auftretens der Distanz zwischen Ziellandepunkt und tatsächlichem Aufsetzpunkt des finalen Anflugs dargestellt. In Tabelle 3 sind, unter der Annahme einer Normalverteilung der Abweichungen vom Ziellandepunkt, die entsprechenden Werte aufgeführt. Die Abweichungen sind erwartungsgemäß höher als bei Anflügen ohne Windeinfluss.

Die mittlere Entfernung vom tatsächlichen Landepunkt zum Ziellandepunkt beträgt 5,3 m. Der negative Wert beim Wert „min“ ergibt sich, da eine Resthöhe oberhalb des Aufsetzpunkts vorhanden ist, welche noch abzufliegen ist. Damit ist ein Aufsetzen nur nach dem Zielpunkt möglich.

Der minimale/maximale Abstand (Variationsbreite) zum Aufsetzpunkt beträgt 76 m. Um diesen Wert muss die Landebahn in Richtung Anflug verlängert werden, um sicher innerhalb des Landefelds landen zu können.

4.4. Beschreibung der Parameter im Landeanflug (mit Wind)

Wie in Abschnitt 4.2. beschrieben wurde, berechnet der Aktor für die Rudereinstellung durchgängig einen Vollausschlag. Damit wird die Sinkgeschwindigkeit erhöht und die Sicherheitshöhe (siehe Kapitel 1) der originären Anflugberechnung (beschrieben u.a. in [2]) abgebaut. Durch Vergleich der Bilder 7 - a und 7 - d kann abgelesen werden, dass sich die Vergrößerung des Gleitwinkels von ca. 8 ° am Startpunkt des finalen Anflugs auf 9,5 ° am Landepunkt, durch die Erhöhung des Landeklappenausschlags von 0 bei maximaler Distanz bis zu einem Wert von 50 % (Mittelwerte) am Aufsetzpunkt ergibt. Die durch den Aktor kalkulierten Werte für die Landeklappen im Anflug (Distanz zum Ziellandepunkt) sind im Bild 7 – a als „Boxplots“ dargestellt.

⁶ <https://www.norddeutscher-klimamonitor.de/klima/1981-2010/jahr/mittlere-windgeschwindigkeit/schleswig-holstein->

[hamburg/coastdat-2.html](https://www.norddeutscher-klimamonitor.de/klima/1981-2010/jahr/mittlere-windgeschwindigkeit/schleswig-holstein-hamburg/coastdat-2.html)

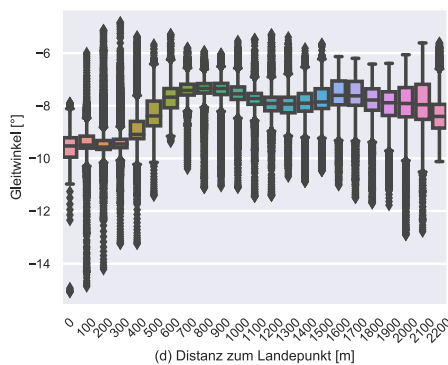
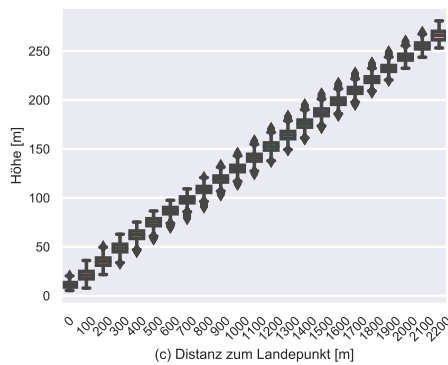
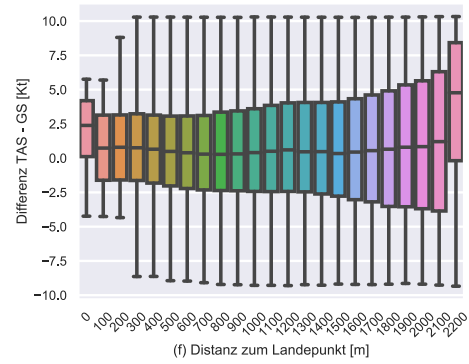
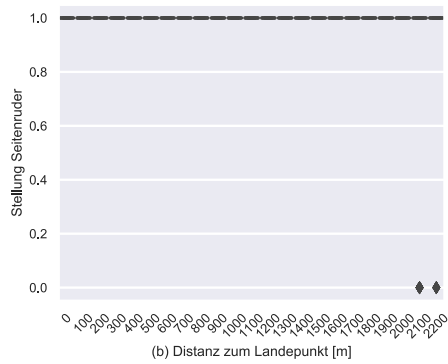
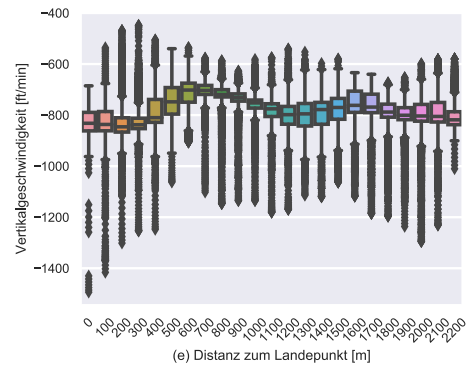
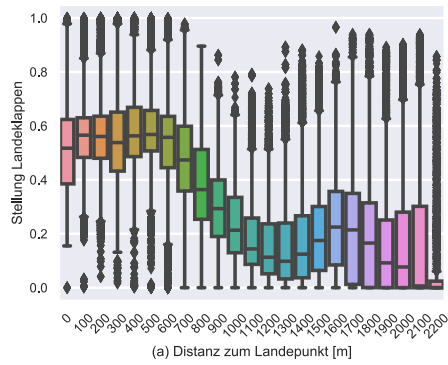


BILD 7 a-f. Parameter während des finalen Anflugs mit Windeinfluss

Für eine Landepfadberechnung ist das Notlandefeld nur dann geeignet sein, wenn der Ziellandepunkt - um die maximale Abweichung verschoben - noch eine sichere Landung zulässt. Für das Notlandeverfahren – Emergency Landing/engine off – kann abgeleitet werden, dass das anzufliegende Notlandefeld näher am LFZ positioniert sein muss, als es über den besten Gleitwinkel zu erreichen ist. Nur somit können Unsicherheit bei der Pfadplanung durch eine Gleitwinkeländerung ausgeglichen werden.

5. DISKUSSION UND AUSBLICK

In der vorliegenden Arbeit wurde eine RL-basierte Methode präsentiert, welche eine automatisierte Landeeinteilung eines LFZ bei Notfällen (engine off) zum Ziellandepunkt ermöglicht. Es konnte gezeigt werden, dass der Gleitwinkel durch die Einstellung von Seitenrudern und Landeklappen an die unsicheren Windbedingungen so angepasst werden kann, dass ein Landefeld sicher erreicht wird. Das Verfahren ermöglicht es, dass automatische Anflüge bei Notfällen bis zum Boden berechnet bzw. durchgeführt werden können. Gegenstand zukünftiger Untersuchungen ist es, zusätzlich die Fluggeschwindigkeit zu variieren und damit den Ziellandepunkt auch bei stärkeren Fehlprognose der Windkonditionen zu erreichen.

Darüber hinaus wurde eine Methode dargestellt (DDPG), welche das Auffinden einer optimalen Strategie für die Einstellung der eingesetzten Steuerorgane (Ruder, Landeklappen) ermöglicht. Damit konnte gezeigt werden, dass eine komplexe Problemstellung ohne Modellinformationen mit Hilfe des verstärkenden Lernens gelöst werden kann. In Folgeuntersuchungen soll die

gesamte Pfadplanung (vom Notfall- bis zum Ziellandepunkt inklusive Hindernisse) mit Hilfe des verstärkenden Lernens gelöst werden.

Literatur:

- [1] M. Coombes, “Landing Site Reachability and Decision Making for UAS Forced Landings,” 2016.
- [2] M. Klein, A. Klos, J. Lenhardt, and W. Schifffmann, “Moving target approach for wind-aware flight path generation,” *Int. J. Netw. Comput.*, vol. 8, no. 2, pp. 351–366, 2018.
- [3] M. Lamp and R. Luckner, “Automatic Landing of a High-Aspect-Ratio Aircraft without Using the Thrust,” in *Advances in Aerospace Guidance, Navigation and Control*, Cham: Springer International Publishing, 2015, pp. 549–567.
- [4] R. S. Sutton and A. G. Barto, *Reinforcement learning : an introduction*, 2. Auflage. MIT Press, 2018.
- [5] D. Silver, G. Lever, N. Heess, T. Degris, D. Wierstra, and M. Riedmiller, “Deterministic policy gradient algorithms,” *31st Int. Conf. Mach. Learn. ICML 2014*, vol. 1, pp. 605–619, 2014.
- [6] T. P. Lillicrap *et al.*, “Continuous control with deep reinforcement learning,” Sep. 2015.
- [7] V. Mnih *et al.*, “Human-level control through deep reinforcement learning,” *Nature*, vol. 518, no. 7540, pp. 529–533, Feb. 2015.
- [8] Z. Zhang, J. Chen, Z. Chen, and W. Li, “Asynchronous Episodic Deep Deterministic Policy Gradient: Towards Continuous Control in Computationally Complex Environments,” Mar. 2019.
- [9] G. E. Uhlenbeck and L. S. Ornstein, “On the Theory of the Brownian Motion,” *Phys. Rev.*, vol. 36, no. 5, pp. 823–841, Sep. 1930.
- [10] B. Lau, “DDPG-Keras-Torcs,” *DDPG-Keras-Torcs*, 2016. [Online]. Available: <https://github.com/yanpanlau/DDPG-Keras-Torcs>.