KIEZ

AIRBUS

Gefördert durch:

Bundesministerium für Wirtschaft und Klimaschutz

aufgrund eines Beschlusses des Deutschen Bundestages

Exemplar-Nr./ Copy No.:

Ausgabe / Issue: **1**

Projekt / Project: **KIEZ4-0Fs-20D1914A**

Art des Dokuments / Type of Document: Abschlussbericht

Project	Firma Company	- Nr. - No.	Ausgabe Issue
KIEZ	**ADS**	**1000**	**1**

Verfasser / Author: Alasdair Cook

Titel / Title: Künstliche Intelligenz Europäisch Zertifizieren unter Industrie 4.0 Aspekten für Flugsteuerung

Datum / Date: 05.06.2025

Berichtsumfang / Number of Pages: **56**

Der Inhalt dieser Unterlage ist geistiges Eigentum der Airbus Defence and Space GmbH. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor.

Das diesem Bericht zugrundeliegende Vorhaben wird mit Mitteln des Bundesministeriums für Wirtschaft und Energie gefördert. Alle Rechte für den Fall der Patenterteilung oder Gebrauchsmuster-Eintragung vorbehalten.

The content of this document is the intellectual property of Airbus Defense and Space GmbH. The responsibility for the content of this publication rests with the author.

The project which forms the basis for this report is endorsed and funded by the Federal Ministry for Economic Affairs and Energy. All rights are reserved in the event of the grant of a patent or the registration of a utility model or design.

	Kennblatt / Identification Sheet
Dokument-Nummer / Report No:	KIEZ-ADS-1000-1
Projekt / Project:	KIEZ
Auftraggeber / Customer:	Bundesministerium für Wirtschaft und Energie
Vertrags-Nr. / Contract No.:	20D1914A
Vertragsthema / Object of Contract:	Luftfahrtforschungsprogramm der Bundesrepublik Deutschland
Vertragsdatum / Date of Contract: von-bis	01.07.2020 - 30.09.2024
Vertragsdauer / Contract Termination:	4 Jahre 3 Monaten
Interne Auftragsnummer / Internal Order No.:	Netzplan Nr. 510675
PSP-Nr. / WBS No.:	-
Schlüsselworte / Keywords:	KIEZ – Certification of AI

	Bearbeiter/ Prepared by:			
Unterschr. / Sign	Redacted			
Datum / Date	05.06.2025			
Name / Name	Cook, Alasdair			
Abtlg. / Dept.	TADIR2			
	Projektleitung/ Technical Management:	**Programmleitung/** Project Management:	**Akquisitionsbereich/** Acquisition:	
Unterschr. /Sign.	Redacted	Redacted	Redacted	
Datum / Date	05.06.2025	05.06.2025	05.06.2025	
Name / Name	Redacted	Redacted	Redacted	
Abtlg. / Dept.	Redacted	Redacted	Redacted	

Inhalt

1 Wissenschaftliche / technische Ergebnisse 6

1.1 HINWEIS ZUR BERICHTERSTATTUNG, DER NUR FÜR AIRBUS GILT 6

1.2 ERGEBNISSE DES HAP 1 – KONZEPTE 6

1.2.1 AP1.1 Synthese des bestehenden Know-hows für die Zertifizierung 6

1.2.1.1 Untersuchungsgegenstand und Notwendigkeit der Zertifizierung 6

1.2.1.2 Identifizierte Zertifizierungskonzepte und beteiligte Akteure 6

1.2.1.3 Analyse des regulatorischen Rahmens für Lufttüchtigkeit 7

1.2.1.4 Ergebnisse zur Anwendung von CS-25 und AMC 25.1309 7

1.2.1.5 Analyse des Entwicklungsprozesses gemäß ED-79A 8

1.2.1.6 Erkenntnisse zur Softwareentwicklung nach ED-12C/DO-178C 8

1.2.1.7 Untersuchung moderner Softwaretechnologien und Abweichungen von Standards .. 8

1.2.1.8 Zusammenfassende Bewertung der Ergebnisse und Ausblick 9

1.2.2 AP1.2 Zertifizierungskonzepte für KI 9

1.2.2.1 Grundlagen der Zertifizierungsanforderungen in der Luftfahrt 9

1.2.2.2 Stand der Technik: KI und Softwareentwicklungsparadigmen 11

1.2.2.3 Zukunft der Künstlichen Intelligenz in der Luftfahrtindustrie und Herausforderungen der Zertifizierung 12

1.2.2.4 Vorschlag für zertifizierbare Künstliche Intelligenz (Kern des Arbeitspakets) 12

1.3 ERGEBNISSE DES HAP 2 – ANWENDUNGEN VON KI 13

1.3.1 AP2.1 KI Potentiale für Luftfahrzeug, Flugsicherung, Flughafen 13

1.3.1.1 Einleitung 13

1.3.1.2 Potenziale der Künstlichen Intelligenz in der Luftfahrt 14

1.3.1.3 Technologien der Künstlichen Intelligenz als Enabler 15

1.3.2 AP2.2 Anwendungen mit KI für Luftfahrzeug, Flugsicherung, Flughafen 16

1.3.2.1 Hintergrund und Projektziele 17

1.3.2.2 Das KIEZ-System: Fähigkeiten und Änderungen gegenüber bestehenden Systemen 18

1.3.2.3 Systemübersicht, Architektur und Modi 19

1.3.2.4 Betriebliche Prozesse 21

1.3.2.5 Analyse, Auswirkungen und zukünftige Erweiterungen des vorgeschlagenen Systems 29

1.4 ERGEBNISSE DES HAP 3 – KI DEMONSTRATOREN / USE CASE 31

1.4.1 AP 3.2 Komponenten mit KI 31

1.4.1.1 Einleitung und Zielsetzung 31

1.4.1.2 Theoretische Grundlagen von Temporal Planning Networks 31

1.4.1.3 Anwendungsfall: Flugkorridore und dynamische Zonen 31

1.4.1.4 Softwarearchitektur und Funktionsweise 32

1.4.1.5 Analyse der Systemleistung 33

1.4.1.6 Zusammenfassende Bewertung 35

1.4.2 AP 3.3 Komplexe Anwendungen 35

1.4.2.1 Szenario 1: Automatische Routenwahl und Reaktion auf statische NFZs 35

1.4.2.2 Szenario 2: Umgang mit beweglichen NFZs und POI-Berücksichtigung mit RP-Genehmigung 37

1.4.2.3 Szenario 3: Mehrfache Neuplanung bei statischen NFZs und variablen Automatisierungsgraden 39

1.4.2.4 Szenario 4: Manuelle Eingabe von Zeitbeschränkungen und RP-Ablehnung von Routenoptionen 42

1.5 ERGEBNISSE DES HAP 4 – ANALYSE 45

1.5.1 AP 4.2 Verifikation & Validierung / 4.3 Methodik der Zertifizierung 45

1.5.2 AP 4.4/4.5 Abgleich mit EASA / Neue Methodik für die Zertifizierung von KI 46

1.5.2.1 System- und Softwareüberblick 46

1.5.2.2 Software-Lebenszyklus und Zertifizierungsaktivitäten 47

1.5.2.3 Zusätzliche Überlegungen 48

1.5.2.4 Anhänge: Geplante Konformität 48

1.6 ERGEBNISSE DES HAP 6 49

1.6.1 Zweck und Umfang 49

1.6.2 Überblick über Symbolische KI 49

1.6.3 Anwendbarkeit regulatorischer Rahmenbedingungen 49

1.6.4 Bewertung der Ziele des EASA-Konzeptpapiers 49

1.6.5 Besondere Überlegungen zu Expertensystemen 50

1.6.6 Fazit 50

2 Projektverlauf 51

2.1 PROJEKTMANAGEMENT (HAP 0) 51

2.2 AP 1.1: SYNTHESE DES BESTEHENDEN KNOW-HOWS FÜR DIE ZERTIFIZIERUNG 51

2.3 AP 1.2: ERSTELLUNG EINES ZERTIFIZIERUNGSKONZEPTS 51

2.4 AP 2.1: ERARBEITUNG EINES ANWENDUNGSKATALOGS UND ANALYSE EXISTIERENDER KI-TECHNOLOGIEN 51

2.5 AP 2.2: ENTWICKLUNG EINES PFADPLANUNGS-SYSTEMS, DEMONSTRATIONSFRAMEWORKS UND CONOPS 52

2.6 AP 3.2: ENTWICKLUNG UND INTEGRATION DER KI-KOMPONENTE, SYSTEMANFORDERUNGEN, FUNKTIONALE ARCHITEKTUR UND SICHERHEITSBEWERTUNG 52

2.7 AP 3.3: INTEGRATION DES TPN-SYSTEMS IN FCS/FMS UND DURCHFÜHRUNG DER DEMONSTRATIONSKAMPAGNE 52

2.8 AP 3.4: INTEGRATION DER DREI DEMONSTRATOREN DER PARTNER 53

2.9 AP 4.2/4.3/4.4: ZERTIFIZIERUNGSASPEKTE UND -METHODEN (HAP 4) 53

2.10 AP 5: QUERSCHNITTSTHEMEN 53

2.11 AP 6: ARBEITEN IM RAHMEN DER PROJEKTVERLÄNGERUNG 2024 53

3 Verwertung der Ergebnisse / Fortschreibung des Verwertungsplans 55

3.1 ERFINDUNGEN/SCHUTZRECHTE DIE IN ANSPRUCH GENOMMEN WURDEN 55

3.2 WISSENSCHAFTLICHE UND TECHNISCHE ERFOLGSAUSSICHTEN NACH PROJEKTENDE 55

1 Wissenschaftliche / technische Ergebnisse

1.1 Hinweis zur Berichterstattung, der nur für Airbus gilt

In diesem Dokument wird über die Erkenntnisse und den Projektstatus aller Arbeitspakete berichtet, zu denen Airbus einen Beitrag leistete. Sämtliche Beiträge der anderen Partner sind in deren jeweiliger Berichterstattung zu finden.

1.2 Ergebnisse des HAP 1 – Konzepte

1.2.1 AP1.1 Synthese des bestehenden Know-hows für die Zertifizierung

Das vorliegende Arbeitspaket (AP) 1.1, hat die komplexen Zertifizierungsanforderungen in der Luftfahrtindustrie untersucht.

1.2.1.1 Untersuchungsgegenstand und Notwendigkeit der Zertifizierung

Die Untersuchung im Rahmen dieses APs bestätigte, dass der sichere Betrieb von Luftfahrzeugen gesetzlich eine nachgewiesene Lufttüchtigkeit erfordert. Mit zunehmender Größe, Komplexität oder dem Risiko eines Luftfahrzeugs steigt der Aufwand für diesen Nachweis exponentiell an. Die Zertifizierung wurde als der Prozess identifiziert, bei dem Hersteller die zuständigen Behörden überzeugen, dass ihr Produkt sicher im Luftraum betrieben werden kann. Die Ergebnisse zeigen, dass dieser Nachweis auf drei Säulen ruht: dem Fehlen von Designfehlern, der konformen Fertigung gemäß dem zertifizierten Design und der technischen Zuverlässigkeit des Produkts. Industriestandards wie DO-178C für die Softwareentwicklung spielen hierbei eine unterstützende Rolle.

Das AP zielte darauf ab, darzulegen, wie Sicherheit in der zivilen Luftfahrt erreicht und nachgewiesen wird, von der Gesetzgebung bis zu konkreten Entwicklungsstandards. Ein besonderer Fokus der Untersuchung lag auf modernen Softwaretechnologien, die bereits in Zertifizierungsanforderungen Erwähnung finden, jedoch noch nicht umfassend in der Industrie etabliert sind. Das Thema Künstliche Intelligenz wurde explizit aus der Untersuchung ausgeklammert, da aktuelle Standards dies noch nicht abbilden und eine Konformitätsbewertung noch nicht möglich ist.

1.2.1.2 Identifizierte Zertifizierungskonzepte und beteiligte Akteure

Die Untersuchung ergab, dass für jedes einzelne Luftfahrzeug ein Lufttüchtigkeitszeugnis notwendig ist, welches bei Serienprodukten auf einem für das Design erteilten Musterprüfzeugnis basiert. Auf internationaler Ebene wurden Schlüsselinstitutionen und deren Zusammenspiel identifiziert: In den USA erarbeiten SAE International und die RTCA technische Leitlinien, während die FAA als Regulierungs- und Zertifizierungsbehörde fungiert. In Europa ist EUROCAE das Pendant zur RTCA und arbeitet eng mit dieser zusammen, was sich in identischen Dokumenten wie ED-12C und DO-178C zeigt. Die EASA agiert als zentrale EU-Agentur für Regulierung, Sicherheit und Zertifizierung. Nationale Behörden wie das LBA sind weiterhin für die Ausstellung individueller Lufttüchtigkeitszeugnisse zuständig.

Das AP stellte fest, dass Dokumente von Industriegremien wie RTCA oder EUROCAE formell nur technische Vorschläge sind. Verbindliche Standards werden durch offizielle

Verlautbarungen der FAA (Advisory Circulars) oder der EASA (Executive Director Decisions) geschaffen. Diese erkennen Industriedokumente häufig als "Acceptable Means of Compliance" (AMC) an – also als akzeptable, aber nicht alleinige Methoden zum Nachweis der Konformität. Die Befolgung dieser AMCs reduziert das Projektrisiko erheblich. Die Zertifizierung selbst ist ein flexibler Prozess, der auf der Überzeugung der Behörden basiert und eine enge Kommunikation und Kooperation erfordert ("certification liaison"). Im Vergleich zu anderen Sektoren wie der Automobilindustrie ist die Luftfahrt durch eine deutlich strengere und direktere Einbindung der Behörden in den Entwicklungsprozess charakterisiert. Militärische Regularien folgen eigenen, hier nicht weiter untersuchten Prinzipien.

1.2.1.3 Analyse des regulatorischen Rahmens für Lufttüchtigkeit

Die Untersuchung des regulatorischen Rahmens zeigte eine mehrstufige Struktur. International bildet das Abkommen von Chicago und die ICAO mit Anhang 8 die Grundlage, welche Lufttüchtigkeit definiert und den Zertifizierungsprozess (Musterzulassung, Lufttüchtigkeitszeugnis) vorgibt. Nationale deutsche Gesetze wie das LuftVG sind weitgehend durch EU-Recht abgelöst worden. Auf europäischer Ebene wurde durch Verordnungen (zuletzt (EU) Nr. 2018/1139) die EASA etabliert und mit der Verantwortung für die Zertifizierung von Luftfahrzeugen betraut.

Die EASA veröffentlicht Zertifizierungsspezifikationen (z.B. CS-25 für Großflugzeuge) und zugehörige AMCs. Eine zentrale Erkenntnis ist die Verknüpfung dieser Regularien: CS 25.1309 legt grundlegende Sicherheitsanforderungen für Ausrüstung fest. Die zugehörige AMC 25.1309 beschreibt akzeptable Nachweisverfahren und verweist für den übergeordneten Systementwicklungsprozess auf ED-79A (bzw. ARP4754A). ED-79A wiederum referenziert für die Softwareentwicklung explizit den Standard ED-12C (identisch zu DO-178C). Die AMC 20-115D bestätigt ED-12C/DO-178C auch als eigenständige Methode für die Softwarezertifizierung. Die Hierarchie und Verknüpfung dieser Dokumente ist in einer Abbildung im Quelldokument visualisiert.

1.2.1.4 Ergebnisse zur Anwendung von CS-25 und AMC 25.1309

Die Detailanalyse der CS-25 für Großflugzeuge ergab, dass diese primär technische und physikalische Anforderungen an das Produkt spezifizieren. Software wird dabei nicht als eigenständige Entität, sondern als integraler Bestandteil von Ausrüstung betrachtet und gemäß deren Anforderungen qualifiziert und zertifiziert. Besonders relevant ist hier CS 25.1309, der fordert, dass katastrophale Fehlerzustände extrem unwahrscheinlich sein und nicht aus einem einzelnen Fehler resultieren dürfen.

Die AMC 25.1309 konkretisiert den Nachweis dieser Anforderungen. Sie definiert eine Ausfallwahrscheinlichkeit für katastrophale Fehler von weniger als 10^{-9} pro Flugstunde. Da Softwarefehler als systematische Entwicklungsfehler betrachtet werden, deren Wahrscheinlichkeit nicht wie bei Hardware durch Tests quantifiziert werden kann, wurde das Konzept der "Development Assurance" (Entwicklungssicherung) als zentrales Ergebnis identifiziert. Dieses zielt darauf ab, durch einen rigorosen und qualifizierten Entwicklungsprozess die Wahrscheinlichkeit unentdeckter Fehler zu minimieren. Die AMC 25.1309 erkennt ED-79A als eine akzeptable Methode an, um diese Entwicklungssicherung zu gewährleisten.

1.2.1.5 Analyse des Entwicklungsprozesses gemäß ED-79A

Das AP hat ED-79A als einen Standard für den generischen Entwicklungsprozess von Luftfahrzeugen und -systemen, oft als V-Modell dargestellt, untersucht. Dieser Prozess verfolgt einen Top-Down-Ansatz, bei dem abstrakte Anforderungen sukzessive verfeinert werden, um Entwicklungsfehler frühzeitig zu erkennen und zu korrigieren. Ein wesentliches Ergebnis ist die Definition von Development Assurance Levels (DALs), die den erforderlichen Grad an Strenge für den Nachweis der Fehlerfreiheit festlegen. Die DAL-Zuweisung (A als höchste Stufe) basiert auf einer Funktionsgefährdungsanalyse.

Von herausragender Bedeutung ist das Prinzip der Unabhängigkeit, das oft durch Redundanz erreicht wird. Werden jedoch identische Systeme redundant eingesetzt, muss der Entwicklungsprozess selbst durch höhere DALs und unabhängige Verifikationsschritte die notwendige Sicherheit gegen gemeinsame Entwicklungsfehler bieten. Die Zuweisung von Functional DALs (FDALs) hängt von der Kritikalität der Funktion und der Unabhängigkeit von anderen Funktionen ab. Werden Funktionen auf Hardware-Items (Ausrüstung) zusammengeführt, muss der Item DAL (IDAL) sicherstellen, dass keine funktionale Unabhängigkeit durch die Kolokation verloren geht, was ggf. eine Erhöhung des IDAL erfordert. Die Verifikation der Implementierung erfolgt gemäß ED-79A durch Inspektion, Analyse, Tests oder Betriebserfahrung, wobei für Software auf ED-12C/DO-178C verwiesen wird.

1.2.1.6 Erkenntnisse zur Softwareentwicklung nach ED-12C/DO-178C

Die Untersuchung hat DO-178C (bzw. ED-12C) als das zentrale Rahmenwerk für die Entwicklung sicherheitskritischer Software in der Luftfahrt identifiziert. Ein Kernaspekt von DO-178C ist die detaillierte Dokumentation aller Prozessschritte und Artefakte, um Nachweise für Qualitätssicherung, Rückverfolgbarkeit, Testabdeckung und Unabhängigkeit zu liefern. Der Standard beschreibt einen V-Modell-basierten Top-Down-Entwicklungsprozess, beginnend mit High-Level Requirements, über Software-Design (Architektur und Low-Level Requirements) bis hin zur Code-Implementierung.

Parallel dazu fordert DO-178C umfassende Verifikationsaktivitäten auf jeder Ebene. Dies beinhaltet Reviews, Analysen und Tests zur Sicherstellung von Korrektheit, Vollständigkeit, Konsistenz und Rückverfolgbarkeit. Besondere Aufmerksamkeit gilt der Verifikation des Softwareverhaltens hinsichtlich Timing, Speicherbedarf, Robustheit und korrekter Integration. Obwohl Tests primär gegen Low-Level Requirements durchgeführt werden, ist eine strukturelle Codeabdeckung (z.B. MC/DC) zwingend erforderlich. Das AP identifizierte zudem ein Ökosystem wichtiger ergänzender Standards, darunter DO-248C (FAQs), DO-278A (Bodensoftware), DO-330 (Werkzeugqualifizierung), DO-331 (modellbasierte Entwicklung), DO-332 (objektorientierte Technologien), DO-333 (formale Methoden) und die DO-200-Serie für aeronautische Daten.

1.2.1.7 Untersuchung moderner Softwaretechnologien und Abweichungen von Standards

Das AP hat auch untersucht, inwieweit DO-178C und seine Ergänzungen moderne Softwareentwicklungspraktiken abdecken. Es wurde festgestellt, dass für Software-Partitionierung, vom Benutzer modifizierbare Software, die Verarbeitung von aeronautischen Daten (geregelt in separaten DO-2xx Standards), objektorientierte

Technologien (OOT) in Verbindung mit DO-332, modellbasierte Entwicklung (MBD) mit DO-331 und formale Methoden mit DO-333 spezifische Leitlinien und Akzeptanzkriterien existieren. Diese ermöglichen den Einsatz innovativer Ansätze unter Wahrung der Sicherheitsanforderungen.

Allerdings zeigte die Analyse auch, dass Abweichungen von diesen etablierten Standards und Prozessen nur schwer zu rechtfertigen sind und ein hohes Projektrisiko darstellen. Jeder alternative Prozess muss mindestens das gleiche Sicherheitsniveau und die gleiche Unabhängigkeit gewährleisten und umfassend begründet werden. Die Komplexität solcher Abweichungen wird am Beispiel von OOT deutlich, für das mit DO-332 ein umfangreiches Zusatzdokument notwendig wurde, um die spezifischen Herausforderungen dieser Technologie adressieren zu können.

1.2.1.8 Zusammenfassende Bewertung der Ergebnisse und Ausblick

Die Ergebnisse des Arbeitspakets 1.1 zeigen, dass die Regulierung in der Luftfahrtindustrie zu den strengsten weltweit gehört und technische, operationelle sowie prozessuale Aspekte abdeckt. Die EASA hat mit der CS-25 und den zugehörigen AMCs ein präzises Zertifizierungssystem etabliert, das ED-79A und ED-12C (DO-178C) als empfohlene Entwicklungspraktiken für Systeme und Software verankert. DO-178C formalisiert den V-Modell-Ansatz und spezifiziert detaillierte Entwicklungs- und Verifikationsaktivitäten, um mittels Development Assurance Levels (DALs) die Wahrscheinlichkeit unentdeckter Entwicklungsfehler zu minimieren. Die Untersuchung bestätigte, dass dieser Standard auch "State-of-the-Art"-Technologien wie MBD und formale Methoden berücksichtigt. Künftige Untersuchungen müssen klären, inwieweit dieser Rahmen für KI- und ML-Ansätze adaptierbar ist. Die EU-Verordnung Nr. 2018/1139, die erstmals unbemannte Luftfahrzeuge reguliert, wurde als wichtiger rechtlicher Rahmen für zukünftige Entwicklungen in diesem Sektor identifiziert

1.2.2 AP1.2 Zertifizierungskonzepte für KI

Die Ergebnisse des Arbeitspakets 1.2 zeigen den aktuellen Stand der Technik intelligenter Softwaresysteme mit einem starken Fokus auf die Luftfahrtindustrie. Es wurden vielversprechende Softwaretechnologien beleuchtet und Vorschläge für einen Entwicklungs- und Zertifizierungsprozess präsentiert, der auch Aspekte der Künstlichen Intelligenz (KI) berücksichtigt. Das Arbeitspaket verfolgte dabei bewusst eine konservative Herangehensweise an KI, die auf Technologien fokussiert, deren Potenzial in der Industrie noch nicht voll ausgeschöpft ist und die als ausreichend reif für sicherheitskritische Anwendungen und deren Zertifizierung angesehen werden. Ansätze wie Maschinelles Lernen wurden daher nicht primär behandelt.

1.2.2.1 Grundlagen der Zertifizierungsanforderungen in der Luftfahrt

Die Zertifizierungsanforderungen in der Luftfahrt zielen darauf ab, die Zuverlässigkeit eines Systems hinsichtlich seiner Sicherheit bis zu einem akzeptablen Niveau nachzuweisen. Dies impliziert spezifische Entwicklungsmethoden und Verifikationsaktivitäten, abhängig von den eingesetzten Technologien. Zwei Hauptbereiche sind hierbei zentral:

- Development Assurance (Entwicklungsabsicherung): Dieser Aspekt stellt sicher, dass der Entwicklungsprozess eine akzeptable Strenge aufweist. Es soll

gewährleistet werden, dass Entwickler ein angemessenes Verständnis des Systems, seiner Anforderungen, seiner Nutzung und seiner Umgebung haben, um alle relevanten Aspekte zu berücksichtigen. Kernkonzepte sind Kompetenz, Unabhängigkeit und Nachverfolgbarkeit (Traceability). Softwarefehler werden im Grunde als Entwicklungsfehler betrachtet, weshalb der Qualität des Entwicklungsprozesses eine hohe Bedeutung zukommt.
- Verification (Verifizierung): Hierbei geht es darum, dass die Systemeigenschaften ein akzeptables Niveau an Zuverlässigkeit aufweisen. Die Verifizierung attestiert gewünschte und spezifizierte Systemeigenschaften mit dem Ziel, die Wahrscheinlichkeit eines Systemausfalls akzeptabel klein zu halten. Grundlegende Konzepte umfassen die Wahl der Technologien, Tests und Analysen.

Der klassische Entwicklungsprozess, oft nach dem V-Modell strukturiert, und Programmiersprachen wie C/C++, Ada oder Java sind in sicherheitskritischen Anwendungen etabliert. Jede alternative Herangehensweise, insbesondere für KI-Systeme, muss mindestens das gleiche Sicherheitsniveau und die gleiche Qualität der Sicherheitsnachweise erbringen.

Ein zentraler Einstiegspunkt im klassischen Prozess ist eine umfassende Sicherheitsbewertung (Safety Assessment). Diese umfasst:

- Functional Hazards Assessment: Identifizierung von Risiken durch fehlerhafte Planung, Auslegung oder Implementierung von Systemfunktionen. Daraus resultieren funktionale Sicherheitsanforderungen und die Festlegung des Development Assurance Levels (DAL).
- Non-Functional Hazard Assessment: Betrachtung von Risiken, bei denen die Systemfunktionen zwar korrekt geplant und implementiert sind, die Art und Weise der Ausführung jedoch nicht mit der geplanten Nutzung kompatibel ist (z.B. Zeitverhalten, Präzision, Ressourcenverbrauch). Dieser Aspekt wurde im Rahmen des Arbeitspakets als oft kritischer und weniger formalisiert in Entwicklungsprozessen identifiziert als funktionale Gefahren.

Das Requirements Engineering bildet die Basis der Zertifizierung. Besonders in sicherheitskritischen Systemen sind granulare Anforderungen unerlässlich, um ein vollständiges Bild des Systems zu erhalten und eine lückenlose Verifizierung zu ermöglichen.

Die Softwareimplementierung umfasst mehr als nur das Kodieren; Design, Dokumentation, Einhaltung von Standards und Ressourcenmanagement sind ebenso wichtig und müssen nachweisbar sein. Jede Codezeile muss einem unabhängigen Review unterzogen werden.

Die Softwareverifizierung erfolgt primär durch Tests gegen Anforderungen. DO-178C fordert Tests gegen Low-Level-Anforderungen anstelle von klassischen Unit-Tests gegen den Code. Darüber hinaus sind verschiedene Analysen notwendig, um unter anderem die Vorhersagbarkeit, Reproduzierbarkeit und Korrektheit der Softwarekonstruktion sicherzustellen:

- Timing Analysis: Überprüfung des Zeitverhaltens.
- Memory Analysis: Sicherstellung der korrekten Speichernutzung (z.B. Vermeidung von Stack Overflows).

- Mathematical Analysis: Vermeidung von arithmetischen Fehlern, Überläufen und Gewährleistung numerischer Stabilität.
- Compilation Analysis: Umgang mit Compilerfehlern.
- Data Flow Analysis: Überprüfung des Datenflusses im Programmcode.

1.2.2.2 Stand der Technik: KI und Softwareentwicklungsparadigmen

Die Entwicklung der Künstlichen Intelligenz verlief in Zyklen von hohem Interesse und Phasen der Ernüchterung ("AI Winter"). Aktuell befinden wir uns in einer Phase hohen Interesses.

Im Rahmen des Arbeitspakets wurde der Begriff "Künstliche Intelligenz" breit gefasst und bezieht sich auf Systeme, deren Funktionen über klassisch programmierte, fixe Algorithmen hinausgehen und Mechanismen bereitstellen, für die Menschen typischerweise natürliche Intelligenz verwenden. Es wurde eher der Begriff "(hoch) adaptive Systeme" als passender erachtet.

Bei den Programmiersprachen für KI wurde LISP als historisch und konzeptionell besonders geeignet hervorgehoben, aufgrund seines Minimalismus, seiner Homoikonizität (Code ist Datenstruktur) und seiner interpretierten Natur, was dynamische Anpassungen erleichtert. Ada hingegen bietet durch statische Typisierung und klare Abbildung auf Objektcode Vorteile für sicherheitskritische Systeme, erfordert aber eine stärkere Anpassung der Domäne an die Sprache. C++ ist mächtig und flexibel für abstrakte Programmierung, erfordert aber hohe Entwicklerkompetenz, um Fallstricke zu vermeiden. Python ist populär und effizient für KI, aber nicht für sicherheitskritische zertifizierbare Anwendungen geeignet. Entscheidend ist, dass die Wahl der Programmiersprache die Entwicklung eines abstrakten Programmiermodells der technischen Domäne unterstützen und nicht behindern darf.

Moderne Softwareentwicklungstechniken, die für dynamische Systeme und KI relevant sind und bereits von DO-178C und ergänzenden Dokumenten (DO-331, DO-332, DO-333) adressiert werden, umfassen:

- Objektorientierte Technologie (OOT): Ermöglicht dynamisches Verhalten durch Datenstrukturen, die den Programmfluss steuern. DO-332 bietet einen Zertifizierungsrahmen. OOT ist intuitiv, birgt aber bei falscher Anwendung Risiken der Komplexitätssteigerung.
- Modellbasierte Entwicklung (MBD): Verwendet formale Modelle zur Spezifikation von Verhalten oder Struktur, aus denen Code generiert werden kann. DO-331 adressiert MBD. Ein vielversprechender Ansatz für KI sind Laufzeitmodelle, bei denen Modelle zur Laufzeit interpretiert oder modifiziert werden, um adaptives Verhalten zu realisieren.
- Formale Methoden: Nutzen mathematische Modelle zur präzisen Spezifikation. Sie ermöglichen den Nachweis von Systemeigenschaften durch mathematisches Schließen, was über die Aussagekraft von Tests hinausgeht. DO-333 behandelt formale Methoden. Sie sind besonders relevant für Systeme mit dynamisch änderndem Verhalten, da sie helfen können, Invarianten über alle möglichen Konfigurationen nachzuweisen.
- Programmdaten: Bezieht sich auf Daten, die zur Laufzeit geladen oder modifiziert werden und das Systemverhalten beeinflussen (z.B. Field Loadable Software, User Modifiable Software, Konfigurationsdaten, aeronautische Daten). Bestehende

Standards (DO-178C, DO-200A) decken Aspekte davon ab, aber eine vollständige Abdeckung für Laufzeitmodelle ist noch offen.
- Software-Partitionierung: Trennung von Softwarekomponenten unterschiedlicher Kritikalität, um zu verhindern, dass weniger kritische Teile die sicherheitskritischen Funktionen beeinträchtigen. DO-178C adressiert Mechanismen zur Software-Partitionierung.

1.2.2.3 Zukunft der Künstlichen Intelligenz in der Luftfahrtindustrie und Herausforderungen der Zertifizierung

Moderne KI-Ansätze, insbesondere solche, die auf Machine Learning basieren, stellen traditionelle Zertifizierungsansätze (wie DO-178C) vor erhebliche Herausforderungen. Oftmals ist das genaue Verhalten solcher Systeme schwer vorhersagbar oder nachweisbar, insbesondere wenn das Problemdomänenverständnis der Ingenieure unvollständig ist, was eine Kernanforderung der klassischen Entwicklungsprozesse untergräbt. Aktivitäten wie Unit-Tests oder bestimmte Analysen (z.B. Division-by-Zero-Analyse in einem neuronalen Netz) sind möglicherweise nicht direkt anwendbar. Strukturelle Softwareeigenschaften (Speicherverbrauch, Zeitverhalten) sind bei sich selbst modifizierenden oder lernenden Systemen schwer zu garantieren.

Das KIEZ-Projekt zielt daher nicht darauf ab, völlig neue Zertifizierungsstandards für alle Arten von KI zu entwickeln, sondern einen Pfad aufzuzeigen, der mit aktuellen Standards wie DO-178C kompatibel ist. Dies bedeutet eine Fokussierung auf eine konservativere Interpretation von KI.

1.2.2.4 Vorschlag für zertifizierbare Künstliche Intelligenz (Kern des Arbeitspakets)

Das Arbeitspaket schlägt einen Ansatz vor, der auf der Abstraktion durch Modelle, deren algorithmischer Verarbeitung und der Nutzung von Domänenspezifischen Sprachen (DSLs) sowie Formalen Methoden basiert.

- Modelle als Kern: Abstrakte Repräsentationen von Fakten und Systemzuständen (Modelle) sind die Basis für sichere Entscheidungen. Diese Modelle sind reine Datenstrukturen, auf denen Algorithmen operieren. Ihre Entwicklung und Spezifikation unterliegt klassischen Engineering-Disziplinen und Qualitätskriterien (Konsistenz, Genauigkeit etc.) und muss den Anforderungen von DO-178C und DO-331 genügen.
- Von Modellen zu Algorithmen: Es wird eine Trennung vorgeschlagen: generische Algorithmen, die spezifiziert, entwickelt und zertifiziert werden, operieren auf Daten (Modellen), die zur Laufzeit geladen oder verändert werden können. Dies ermöglicht "intelligenzartiges" Verhalten, da das Systemverhalten durch die Daten und nicht durch einen fixen Algorithmus bestimmt wird.
- Domänenspezifische Sprachen (DSLs): DSLs dienen dazu, Modelle menschenlesbar und handhabbar zu machen. Sie definieren eine Sprache für eine spezifische Domäne und ermöglichen es, Modelle zu erstellen, zu laden, zu transformieren und zu verifizieren. DSL-Werkzeuge (Editoren, Parser) sind Entwicklungswerkzeuge und unterliegen der Werkzeugqualifizierung (DO-330). Eine DSL erzwingt Disziplin bei der Algorithmenentwicklung, da nur die im Modell explizit gemachten Aspekte verwendet werden können.

- Formale Methoden: Die Definition einer präzisen mathematischen Semantik für die Modelle und DSLs ermöglicht es, Eigenschaften rigoros nachzuweisen. Dies ist besonders wichtig für die Verifizierung von Modelltransformationen und die Gewährleistung von Sicherheitseigenschaften.
- Werkzeuge (Language Workbenches): Moderne Werkzeuge wie Jetbrains MPS oder Racket unterstützen die Definition und Nutzung von DSLs erheblich und bieten eine Grundlage für den Engineering-Prozess und die Zertifizierung.
- Modelltransformationen: Adaptives Verhalten wird durch Laufzeitmodifikationen des zugrundeliegenden Modells erreicht, die als strukturierte Datentransformationen implementiert werden. Diese Transformationen können mathematisch definiert, analysiert und getestet werden, wenn eine formale Semantik existiert.
- Funktionale Architektur: Es wurde eine konzeptionelle funktionale Architektur vorgeschlagen, bestehend aus Blöcken für PERCEPTION (Sensorik, Kommunikation, Status), AWARENESS Model (reine Datenrepräsentation des Systemzustands), BEHAVIOUR Model (Verhaltensregeln als Daten), DECISION (generische Algorithmen, die auf Awareness- und Behaviour-Modellen operieren) und AIRCRAFT (Aktuatoren, Flugsteuerung). Die ADAPTATION-Funktionalität (Anpassung der Modelle) ist hierbei noch flexibel einzuordnen, abhängig von der Komplexität und den Anforderungen.
- Vorgeschlagene Erweiterungen zu DO-178C: Um diesen Ansatz im Rahmen bestehender Zertifizierungsstandards abzubilden, wurden konkrete neue oder erweiterte Ziele für verschiedene Tabellen in DO-178C vorgeschlagen. Diese betreffen u.a. die Entwicklung und Verifizierung von Modellanforderungen (New Objective 1, 5, 6, 7, 8), die Ableitung von Low-Level-Anforderungen aus Modellanforderungen (New Objective 3, 9) und die Entwicklung und Verifizierung der Modellanpassungsmechanismen (New Objective 10, 11, 12, 13), wobei hier oft Formale Methoden zum Einsatz kommen. Auch der Zertifizierungs-Liaison-Prozess wurde um ein neues Ziel ergänzt (New Objective 14).

Zusammenfassend stellt das Arbeitspaket fest, dass ein Weg zur Zertifizierung von Systemen mit intelligenten, adaptiven Fähigkeiten in der Luftfahrt durch einen stark modellbasierten Ansatz unter Nutzung von Domänenspezifischen Sprachen und Formalen Methoden denkbar ist. Dieser Ansatz erfordert eine Anpassung und Erweiterung der bestehenden Zertifizierungsleitlinien wie DO-178C, um die spezifischen Aspekte der Modellentwicklung, -verifizierung und -transformation adäquat zu berücksichtigen.

1.3 Ergebnisse des HAP 2 – Anwendungen von KI

1.3.1 AP2.1 KI Potentiale für Luftfahrzeug, Flugsicherung, Flughafen

1.3.1.1 Einleitung

Das Arbeitspaket 2.1 hatte zum Ziel, Ideen und Potenziale zu identifizieren, wie Künstliche Intelligenz (KI) für vielfältige Funktionen in Luftfahrzeugen eingesetzt werden kann. Der Fokus lag dabei explizit auf Software und Funktionen, die direkt an Bord des Flugzeugs ausgeführt werden. Softwarelösungen für Bodenstationen, Flugverkehrsmanagement oder andere nicht bordgestützte Anwendungen waren nicht Gegenstand dieser Untersuchung. Die im Folgenden dargestellten Erkenntnisse fassen die identifizierten Anwendungsbereiche und die dafür relevanten KI-Technologien zusammen.

1.3.1.2 Potenziale der Künstlichen Intelligenz in der Luftfahrt

Im Rahmen des Arbeitspakets wurden zahlreiche Bereiche identifiziert, in denen KI als Wegbereiter für zukünftige Produktinnovationen und -verbesserungen in der Luftfahrt dienen kann.

Ein zentrales Anwendungsfeld ist der Systemdiagnose-Assistent. Flugzeuge sind hochkomplexe Systeme, deren korrekte Funktion für die Flugsicherheit unerlässlich ist. Während einzelne Systeme oft über eigene Überwachungsmechanismen verfügen, erfordern manche Abweichungen eine übergreifende Analyse von Daten aus mehreren Komponenten, um korrekt interpretiert zu werden. Herkömmliche technische Überwachungssysteme stoßen hier an Grenzen, da nicht alle Abweichungen automatisch erkannt werden können, insbesondere wenn Messwerte zwar plausibel, aber dennoch fehlerhaft sind. Die Komplexität der Systeme und die Menge an Daten können Piloten, die letztendlich für die Beurteilung des Systemverhaltens zuständig sind, überfordern. Dies zeigte sich beispielsweise beim Air-France-Flug 447, wo die Piloten trotz klarer Prozeduren und Warnsysteme die Situation nicht korrekt erfassten, was zu einem Absturz führte. Ein KI-basierter Assistent könnte hier Abhilfe schaffen, indem er große Datenmengen analysiert, Muster erkennt und den Piloten wichtige Informationen zur Interpretation des Systemzustands und zur Ursachenfindung bei Fehlermeldungen liefert. Ein solcher Assistent könnte mit dem Expertenwissen von Systemingenieuren gespeist werden und, da er nicht direkt in bestehende Systeme eingreift, potenziell mit einem niedrigeren Sicherheitslevel entwickelt werden.

Die Flugzeugüberwachung stellt ein weiteres Potenzialfeld dar. Feste Wartungsintervalle sind oft konservativ und spiegeln nicht immer den tatsächlichen Zustand eines Bauteils wider, da Verschleiß auch von der Nutzung abhängt. KI könnte durch die Analyse von Betriebsdaten präzisere Vorhersagen für Wartungsaktivitäten ermöglichen und so die Instandhaltung optimieren. Ähnlich wie Autofahrer ein Gefühl für Materialalterung entwickeln, könnten KI-Systeme lernen, subtile Veränderungen im Verhalten eines Flugzeugs zu erkennen und zu interpretieren.

Für autonome Fluggeräte ist die Missionsplanung ein entscheidender Bereich. KI kann hier unterschiedliche Autonomiegrade unterstützen, von ferngesteuerten Systemen, bei denen ein Operateur eine Sequenz von Aktivitäten vorgibt, bis hin zu vollständig autonomen Systemen, die Ziele eigenständig in Handlungspläne umsetzen und auf unvorhergesehene Ereignisse reagieren. Solch hohe Autonomiegrade sind besonders dort relevant, wo eine kontinuierliche menschliche Überwachung nicht möglich oder wirtschaftlich ist.

Auch Start- und Landehilfen können von KI profitieren. Während der Start technisch relativ einfach ist, können KI-Systeme insbesondere die Landung unterstützen, vor allem unter Sichtflugbedingungen oder auf Flughäfen ohne Instrumentenlandesysteme. Es wurde jedoch auch bedacht, dass eine zunehmende Automatisierung von Routineaufgaben wie der Landung dazu führen kann, dass Piloten an Übung verlieren, was in unerwarteten Situationen, in denen auch technische Systeme versagen könnten, kritisch sein kann. Ein automatisches System wäre beispielsweise kaum in der Lage, eine Notwasserung wie die des Fluges AWE 1549 ohne entsprechende Programmierung und Tests durchzuführen.

Das Konzept des Single Pilot Aircraft, also Flugzeugen, die nur von einem Piloten geflogen werden, wird von Fluggesellschaften aus Kostengründen angestrebt. Technisch ist der Betrieb mit nur einem Piloten möglich, jedoch stellt das Risiko eines medizinischen Notfalls des einzigen Piloten eine große Herausforderung dar. Eine Realisierung ist daher nur denkbar, wenn das Flugzeug im Notfall vollautonom weiterfliegen und landen kann, was einen sehr hohen Autonomiegrad erfordert. Dies könnte erreicht werden, indem die autonome Flugfähigkeit auf Notfallprozeduren beschränkt wird,

wobei speziell ausgerüstete Flughäfen als Ausweichziele dienen könnten und das Luftraummanagement im Notfall für freie Korridore sorgt.

Der autonome Flug von Passagierflugzeugen stellt die technisch anspruchsvollste Anwendung dar und umfasst zahlreiche Teilfunktionen wie autonomes Rollen am Boden, Kommunikation mit der Flugsicherung, Durchführung aller Flugphasen und die angemessene Reaktion auf jegliche Abweichungen.

Im Bereich der Navigation kann KI unterstützen, insbesondere dort, wo der Luftverkehr nicht stark reguliert ist, um Kollisionen zu vermeiden und No-Fly-Zonen zu berücksichtigen. Dies gilt auch für die Navigation in komplexen Umgebungen wie zwischen Gebäuden in Städten, in Wäldern oder auf Baustellen.

Der Nutzlastbetrieb erfordert oft spezifische Flugmanöver, beispielsweise für Kameras, wissenschaftliche Sensoren oder Luftbetankungssysteme. KI kann hierbei helfen, auch wenn der genaue Flugweg nicht im Voraus festgelegt ist, etwa bei der Identifizierung eines geeigneten Abwurfbereichs für eine Lieferdrohne oder bei der Erkundung unbekannter Gebiete für wissenschaftliche Messungen.

Ein weiteres interessantes Feld ist die Anwendung von KI bei instabilen Flugzeugen. Zivile Flugzeuge sind üblicherweise aerodynamisch stabil ausgelegt, was jedoch Energie kostet, da das Heckleitwerk kontinuierlich einen negativen Auftrieb erzeugen muss. Instabile Flugzeugdesigns könnten signifikant treibstoffeffizienter sein, erfordern aber eine kontinuierliche aktive Regelung durch Flugsteuerungssysteme. KI könnte Mechanismen bereitstellen, um solche Flugzeuge flexibler und selbstlernender hinsichtlich verschiedener Konfigurationen und Massenverteilungen zu machen, was die Akzeptanz solcher Designs im zivilen Bereich fördern könnte.

Schließlich wurde das ökonomische Potenzial autonomer Drohnen beleuchtet. Während bei großen Kampfflugzeugen die Personalkosten für Piloten im Vergleich zu den Gesamtkosten einer Flugstunde relativ gering erscheinen mögen, und auch bei unbemannten Drohnen wie der Heron noch erhebliches Bodenpersonal benötigt wird, sieht die Rechnung bei kommerziellen Fluggesellschaften anders aus. Hier können die Personalkosten für Piloten einen signifikanten Anteil der Gewinnmarge ausmachen. Für kleinere und kostengünstigere Flugsysteme wird das wirtschaftliche Potenzial autonomer Systeme noch deutlicher, beispielsweise für Transportdienste, fliegende Taxis, Erdbeobachtung oder landwirtschaftliche Anwendungen.

1.3.1.3 Technologien der Künstlichen Intelligenz als Enabler

Für die Umsetzung der genannten Potenziale wurden verschiedene KI-Technologien als vielversprechend identifiziert.

Die Verhaltensmodellierung, die formale Konzepte wie Zustandsautomaten, Entscheidungsbäume oder Planungsnetzwerke umfasst, eignet sich gut für technische Prozesse und Algorithmen. Ihre strukturierte Natur und die Möglichkeit der formalen Verifikation machen sie attraktiv für sicherheitskritische Anwendungen wie die Missionsplanung oder die Planung von Notfallprozeduren beim Single Pilot Aircraft.

Die Stochastische Modellierung verwendet strukturierte mathematische Modelle wie Bayes'sche Netze, um Wahrscheinlichkeiten und Korrelationen zu definieren, was für Inferenzalgorithmen zur Entscheidungsfindung oder zum logischen Schließen genutzt wird. Diese eignen sich gut für Systemdiagnose-Assistenten, da der Systemzustand oft nur mit Unsicherheiten aus Sensordaten abgeleitet werden kann. Für sicherheitskritische Navigationsaufgaben sind sie weniger geeignet, es

sei denn, die Sicherheitsanforderungen können durch andere Methoden, wie ein übergeordnetes Sicherheitssystem, reduziert werden.

Verstärkendes Lernen (Reinforcement Learning) ist ein algorithmisches Konzept, bei dem eine Maschine Entscheidungen trifft, deren Auswirkungen bewertet und daraus lernt, um zukünftige Entscheidungen zu verbessern. Dieses Konzept könnte für selbstlernende Regelungsketten in Flugsteuerungssystemen, insbesondere bei instabilen Flugzeugen, verwendet werden, indem das System schrittweise sein Wissen über die Flugzeugcharakteristika erweitert und so den sicheren Flugbereich vergrößert. Auch für nutzlastspezifische Funktionen, bei denen unbekannte Objekte oder Gebiete schrittweise erkundet werden sollen, erscheint dieser Ansatz vielversprechend.

Bilderkennung und Neuronale Netze haben in den letzten Jahren beeindruckende Erfolge in der Mustererkennung und Datenverarbeitung gezeigt. Ihre Stärke liegt darin, dass sie aus großen Datenmengen lernen können, ohne dass die zu identifizierenden Muster explizit spezifiziert werden müssen. Dies bietet Vorteile bei Problemen, für die keine algorithmische Lösung bekannt ist. Mögliche Anwendungen in der Luftfahrt wären Systemdiagnose-Assistenten, die Überwachung von Flugzeugen zur Wartungsprädiktion und visuelle Landehilfen. Allerdings stellt die Komplexität und die fehlende Nachvollziehbarkeit der internen Funktionsweise Neuronaler Netze eine große Herausforderung für die Verifikation und Zertifizierung in sicherheitskritischen Bereichen dar. Die Demonstration extrem niedriger Fehlerraten, wie sie in der Luftfahrt gefordert werden (z.B. 1/109), ist durch reines Testen unrealistisch. Daher ist hier noch Grundlagenforschung notwendig, bevor Neuronale Netze für kritische Flugsteuerungsalgorithmen eingesetzt werden können.

Neben diesen fortgeschrittenen KI-Technologien wurde auch die fortwährende Bedeutung von Standardalgorithmen und klassischen imperativen Programmiersprachen wie C oder Ada hervorgehoben. Viele Anwendungsfälle, wie der Systemdiagnose-Assistent (sofern keine fortgeschrittenen Machine-Learning-Aspekte involviert sind), die Missionsplanung oder Navigationsaufgaben mit bekannten Algorithmen, können hiermit umgesetzt werden. Auch die meisten nutzlastspezifischen Steuerungen fallen in diese Kategorie. Für autonome Drohnen mit einfachen, vordefinierten Flugpfaden sind ebenfalls keine neuen KI-Konzepte zwingend erforderlich. Bei sehr komplexen Aufgaben wie der optischen Bildverarbeitung für Landehilfen oder einem vollständig autonomen Flug stoßen klassische Algorithmen jedoch an ihre Grenzen.

Schließlich wurde auf Optimierungsprobleme als eine Klasse von Algorithmen hingewiesen, die darauf abzielen, optimale Lösungen für mathematisch spezifizierte Probleme zu finden. Diese sind oft Bestandteil der Verhaltensmodellierung oder können für spezifische Aufgaben wie die Berechnung der Anflugkurve bei einer Landehilfe eingesetzt werden.

Diese Ergebnisse des Arbeitspakets 2.1 zeigen ein breites Spektrum an Möglichkeiten für den Einsatz von KI in der Luftfahrt auf, verdeutlichen aber auch die technologischen und zertifizierungstechnischen Herausforderungen, die insbesondere bei sicherheitskritischen Anwendungen noch zu bewältigen sind.

1.3.2 AP2.2 Anwendungen mit KI für Luftfahrzeug, Flugsicherung, Flughafen

Für Arbeitspaket 2.2 (AP2.2) wurden zwei Hauptschwerpunkte verfolgt. Der erste war die Erstellung eines Betriebskonzepts (Concept of Operations, CONOPS), der zweite die Vorbereitung eines Demonstrationssystems. Die Entwicklung des Letzteren erfolgte in Zusammenarbeit mit Arbeitspaket 3.2 (AP3.2), weshalb hierüber im Abschnitt von AP3.2 berichtet wird. Für das CONOPS bestand das Hauptziel darin, zu definieren, wie der gewählte Anwendungsfall (Vermeidung von Flugverbotszonen) in einem bemannten Luftfahrzeug realisiert werden könnte.

Dies bildet die Grundlage für die Anforderungen, welche die Entwicklung des Systems in AP3.2 steuerten. Nachfolgend sind die Erkenntnisse aus dem CONOPS-Dokument dargestellt.

Der CONOPS beschreibt aus Anwendersicht das Betriebskonzept (Concept of Operations) eines generischen ferngesteuerten Luftfahrzeugsystems (Remotely Piloted Aircraft System - RPAS), das mit einem auf einem Temporal Planning Network (TPN) basierenden Navigations-Routing-Werkzeug namens KIEZ ausgestattet ist. Ziel des Dokuments ist es, die Betriebsumgebung, die Fähigkeiten und die Nutzung des KIEZ-Systems zu definieren und somit eine Grundlage für den Systementwurf und die -definition zu schaffen. Es dient dazu, die Systemfähigkeiten herauszuarbeiten und Bereiche zu identifizieren, die für ein nach EASA-Sicherheitszielen zertifizierbares Produkt gelöst werden müssen. Dies stellt die erste Ausgabe des CONOPS dar.

1.3.2.1 Hintergrund und Projektziele

Das KIEZ-Programm zielt darauf ab, Methoden zur Zulassung eines Routenplanungs-Softwaremoduls zu entwickeln, das auf einem Temporal Planning Network (TPN) basiert. KIEZ selbst ist als ein KI-"Expertensystem" ohne maschinelles Lernen konzipiert, das bei identischen Eingabedaten konsistente und wiederholbare Ergebnisse liefert (siehe Abbildung 1 zur KI-Taxonomie). Die Autonomieebenen von KIEZ sind variabel und hängen von den Entscheidungen des verantwortlichen Piloten (Pilot in Command - PIC) ab. Sie können von Level 1 (Unterstützung des Menschen) bis Level 3 (autonome Maschine, potenziell nicht übersteuerbar in Szenarien mit Verlust der Steuerungsverbindung – Command and Control Link Loss, C2LL) reichen. Die Grundlagen für die Fähigkeiten des Systems wurden aus dem KIEZ 4.0 Statement of Work, Standardisierungsdokumenten, Piloteninterviews und vorheriger Forschung abgeleitet.

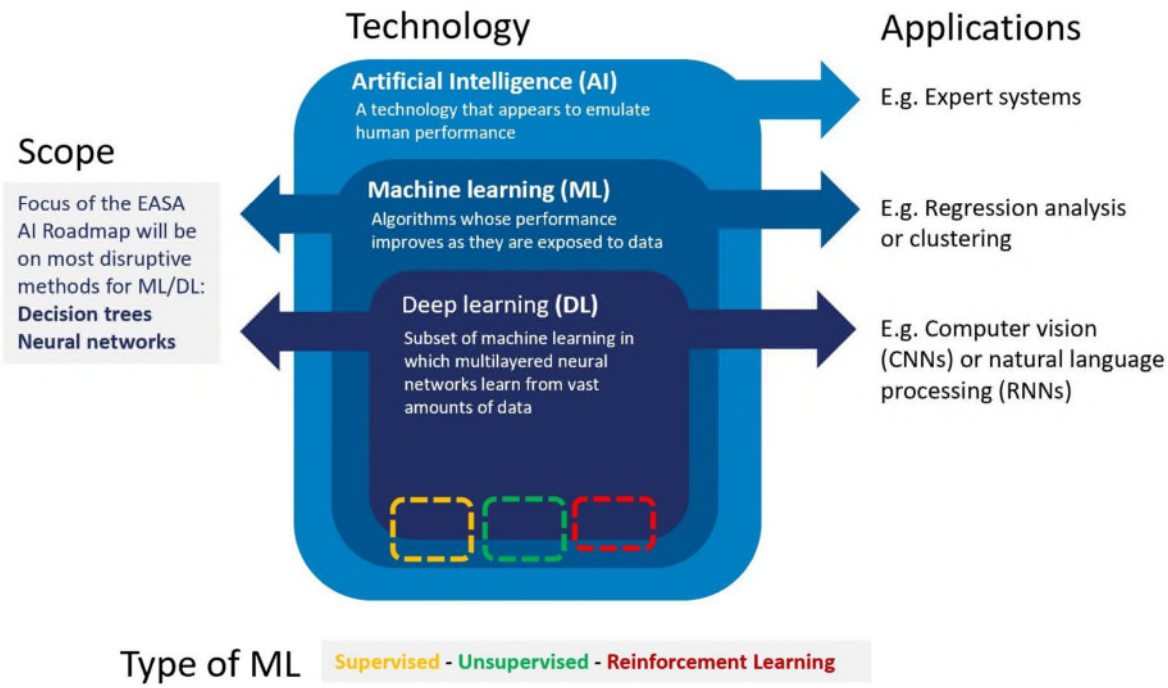


Abbildung 1 - KI-Taxonomie

Der Umfang der KIEZ-Software beschränkt sich auf Routenplanung, Navigation (sowohl individuell als auch kollaborativ) und Routenüberwachung unter der Kontrolle eines „Pilot-In-Command“ (PIC). KIEZ fungiert als eine auf TPN basierende Routenplanungs- und Managementhilfe, die eine schnelle Definition von Flugrouten ermöglicht, um vordefinierte Zeitfenster einzuhalten, Routen nach

Abweichungen neu zu planen, Sperrgebiete (z.B. Wetter, No Fly Zones - NFZ) zu berücksichtigen, die Arbeitslast des PIC zu reduzieren und eine Vielzahl von Faktoren jenseits menschlicher Fähigkeiten zu bewerten. KIEZ kann während der Reiseflugphasen aktiv die Routenführung übernehmen (dargestellt in Abbildung 2). Das System berücksichtigt die ICAO-Integrationsprinzipien für RPAS und zielt auf eine nahtlose Integration durch Einhaltung der Verfahren für bemannte Luftfahrzeuge und Gewährleistung der Sicherheit ab. Die Betriebsumgebung umfasst IFR-Bedingungen (IMC/VMC), Luftraumklassen A-G, segregierten/nicht-segregierten Luftraum, Betrieb am Boden und während des Reiseflugs, einzelne oder im Schwarm agierende (KIEZ-ausgestattete) RPAS, einen einzelnen PIC und RPAS der EASA-zertifizierten Kategorie. Durchgeführte Piloteninterviews unterstrichen den Bedarf an flexibler Routeneingabe, Pilotenaufsicht, gestufter Automation, multiplen Routenoptionen und klarem Verhalten bei C2LL.

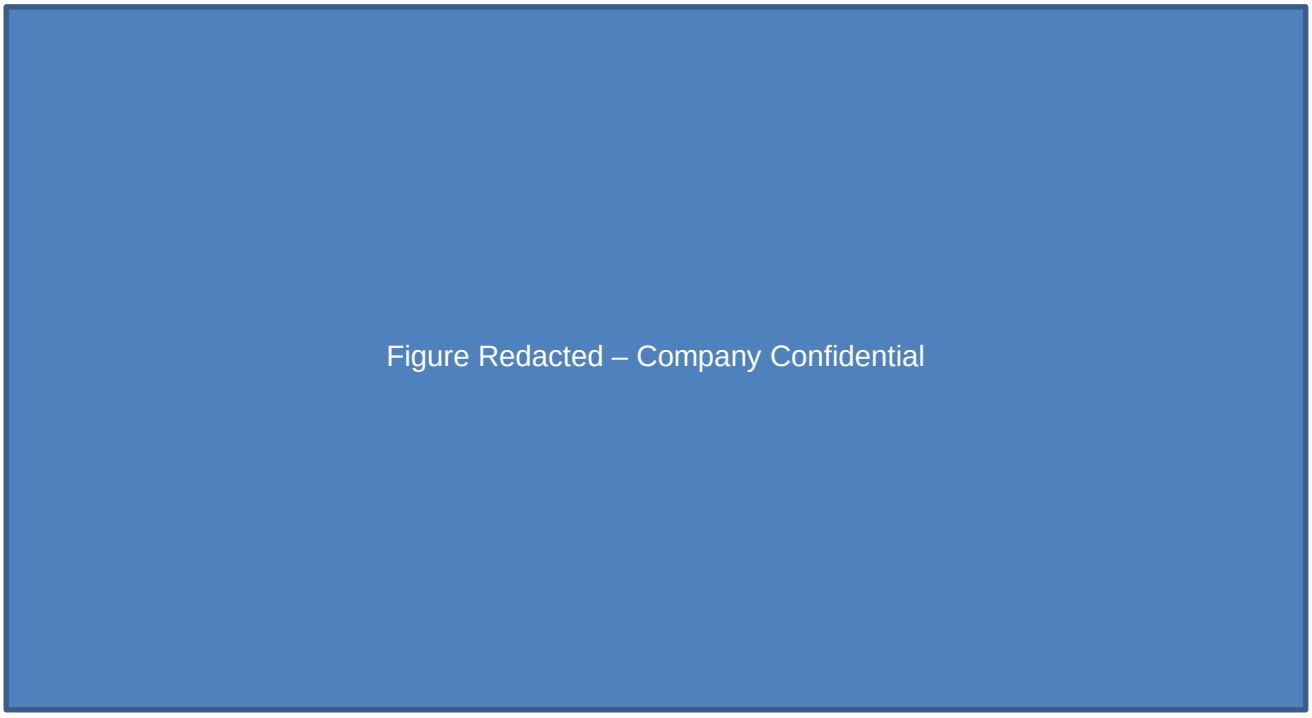


Abbildung 2 – KIEZ PoF

1.3.2.2 Das KIEZ-System: Fähigkeiten und Änderungen gegenüber bestehenden Systemen

KIEZ hat zum Ziel, die Fähigkeiten der bemannten Luftfahrt auf unbemannte Systeme zu übertragen, um die Gemeinsamkeiten zu wahren und die Arbeitsbelastung des Fernpiloten (Remote Pilot - RP) nachweislich zu reduzieren. Die wichtigsten vorgesehenen Fähigkeiten umfassen:

- Missionsplanung vor dem Flug unter Berücksichtigung von NFZ und zeitlichen Vorgaben.
- Überprüfung vor dem Start und Validierung der Route.
- Neuplanung der Route aufgrund von Änderungen der Luftraumverfügbarkeit oder durch bordeigene Systeme gemeldete NFZ.
- Aktivierung der von KIEZ geplanten Route als aktive Route.
- Bereitstellung von halbautomatischer und vollautomatischer Routenplanung.
- "RP Route Mode", bei dem die vom RP definierte Route beibehalten wird, während KIEZ das TPN und NFZ verwaltet.
- Koordination mehrerer kollaborativ agierender KIEZ-ausgestatteter RPAS.
- Management von Systemausfällen, einschließlich C2LL-Szenarien.

Die Änderungen gegenüber aktuellen Systemen beinhalten eine Missionsdateneingabe, die sich auf Ziele, Points of Interest (POIs) und Wegpunkte konzentriert, anstatt auf eine fest vorgegebene Route. Dies führt zu einer geringeren Arbeitsbelastung des RP während des Fluges. KIEZ bietet sowohl zeitliche als auch positionsbezogene Steuerung und kann vom Automation and Emergency Recovery (A&ER) System in Ausfallszenarien genutzt werden. Es ermöglicht flexible C2LL-Routingoptionen, die über fest vordefinierte Routen hinausgehen. Während ADS-B Out bei C2LL den Standort übermittelt, nicht aber die präzise von KIEZ bestimmte Route, wird das KIEZ-Bodenelement die RPAS-Position und -Route unter Verwendung bekannter NFZ und vom RPAS gemeldeter Daten vorhersagen. Die automatische bzw. halbautomatische Neuplanung reduziert die Arbeitsbelastung, wobei der Pilot entweder "in the loop" oder "on the loop" bleibt.

1.3.2.3 Systemübersicht, Architektur und Modi

KIEZ besteht aus zwei Hauptkomponenten: einer Bodenkomponente (KIEZ Ground - KIEZg) und einer Bordkomponente (KIEZ airborne - KIEZa) (siehe Abbildung 3 für eine generische RPAS-Architektur mit KIEZ). Das System bietet eine autonome Routenplanung und -verwaltung unter Verwendung eines reaktiven modellbasierten Ansatzes. Es plant Flüge bei Änderungen der Umgebung oder der Flugziele neu, um die Missionserfüllung, Flugsicherheit und Einhaltung zeitlicher Ziele zu gewährleisten.

- Vom RP wählbare Funktionsmodi (bei aktiver C2-Verbindung und wenn KIEZ das aktive Routingsystem ist) umfassen:
- RP Advice Mode: KIEZ plant und überwacht, berät den RP und benötigt dessen Zustimmung für neue Routen und NFZ-Aktualisierungen.
- Semi-Automatic Mode: KIEZ verwaltet NFZ automatisch und schlägt bei Neuplanungsereignissen Routen zur RP-Genehmigung vor.
- RP Route Mode: Die vom RP definierte Route wird geflogen; KIEZ automatisiert TPN/NFZ-Aktionen und berät, falls die RP-Route betroffen ist. Routenänderungen durch KIEZ erfordern RP-Genehmigung.
- Fully-Automatic Mode: KIEZ plant Routen automatisch neu und aktiviert sie; der RP wird informiert und kann innerhalb eines Zeitlimits übersteuern. Eine Zusammenfassung der Aktionsautomatisierung pro Modus findet sich in Tabelle 1.

Tabelle 1 - Aktionsautomatisierung pro Modus

Nicht vom RP wählbare Systemmodi beinhalten:

- KIEZ (Aircraft part) C2LL Mode: Automatischer Modus, falls A&ER KIEZ für das C2LL-Routing auswählt. Nutzt ein C2LL-Zielset, falls definiert, operiert im vollautomatischen Modus, fügt NFZ von anderen Systemen hinzu, ändert aber keine NFZ im eigenen Datenspeicher. Wenn keine Routen verfügbar sind, werden optionale Ziele entfernt; falls weiterhin keine Routen existieren, wird A&ER gewarnt.
- KIEZ (Ground part) C2LL Mode: Nutzt ADS-B Out zur Positionsschätzung und Routenvorhersage für den RP.
- KIEZ Reduced Data Mode: Alarmiert den RP, wenn eine NFZ-Datenquelle ausfällt, und fordert eine Bestätigung/manuelle Aktualisierung der NFZ durch den RP.
- KIEZ Routing Error Mode: Wird aktiviert, wenn der RP alle Routen ablehnt; kehrt in den RP Advice Mode zurück, das RPAS bleibt auf der zuletzt akzeptierten Route.
- KIEZ Failed Mode: Sendet eine Warnung und gibt keine Routen aus. Eine Wiederherstellung beinhaltet das Neuladen von Daten und die Reaktivierung durch den RP.

Zu den Datenschnittstellenanforderungen gehören ein geladenes Zielset (Goal Set), TPN-Daten (Wegpunkte, Flugstreckenabschnitte, POIs, Einschränkungen, NFZ), der Status von DAA/TAWS-Manövern, die Autorisierung durch den RP, die Bestätigung der Reiseflugphase, Informationen zur Erreichbarkeit von Zielen und Benachrichtigungen über Situationsänderungen.

Abbildung 3 - Generische RPAS-Architektur mit KIEZ

1.3.2.4 Betriebliche Prozesse

Der CONOPS beschreibt detailliert verschiedene betriebliche Abläufe anhand von Flussdiagrammen:

KIEZ01: Missionsdaten- und Flugplanerstellung (Abbildung 4): Der RP gibt Missionsdaten ein, KIEZg erstellt das TPN und eine initiale Route. Der RP prüft und akzeptiert/verwirft diese. Die akzeptierte Route wird bei der Flugsicherung (ATC) eingereicht. Bei Ablehnung durch die ATC erfolgt ein iterativer Prozess.

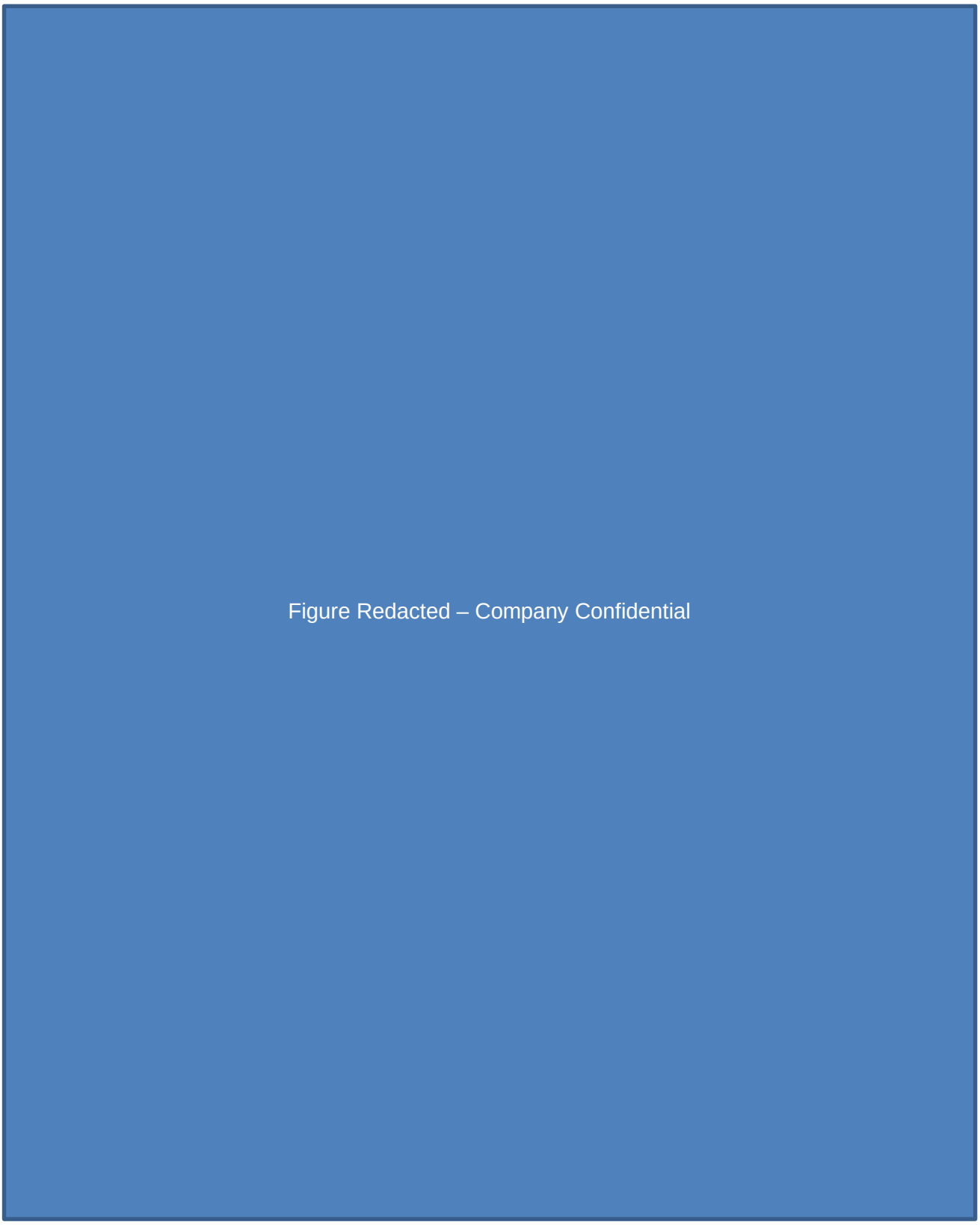


Abbildung 4 - KIEZ01: Missionsdaten- und Flugplanerstellung

KIEZ02: Flugvorbereitung, einschließlich Routenüberprüfung (Abbildung 5): Missionsdaten werden in das RPAS geladen. KIEZa vergleicht die Missionsdatenroute mit dem aktuellen Status-TPN und berechnet bei Bedarf neu. Der RP prüft, ändert ggf. und reicht den Plan bei der ATC ein. Nach ATC-Freigabe wählt der RP vor dem Start die KIEZ-Parameter.

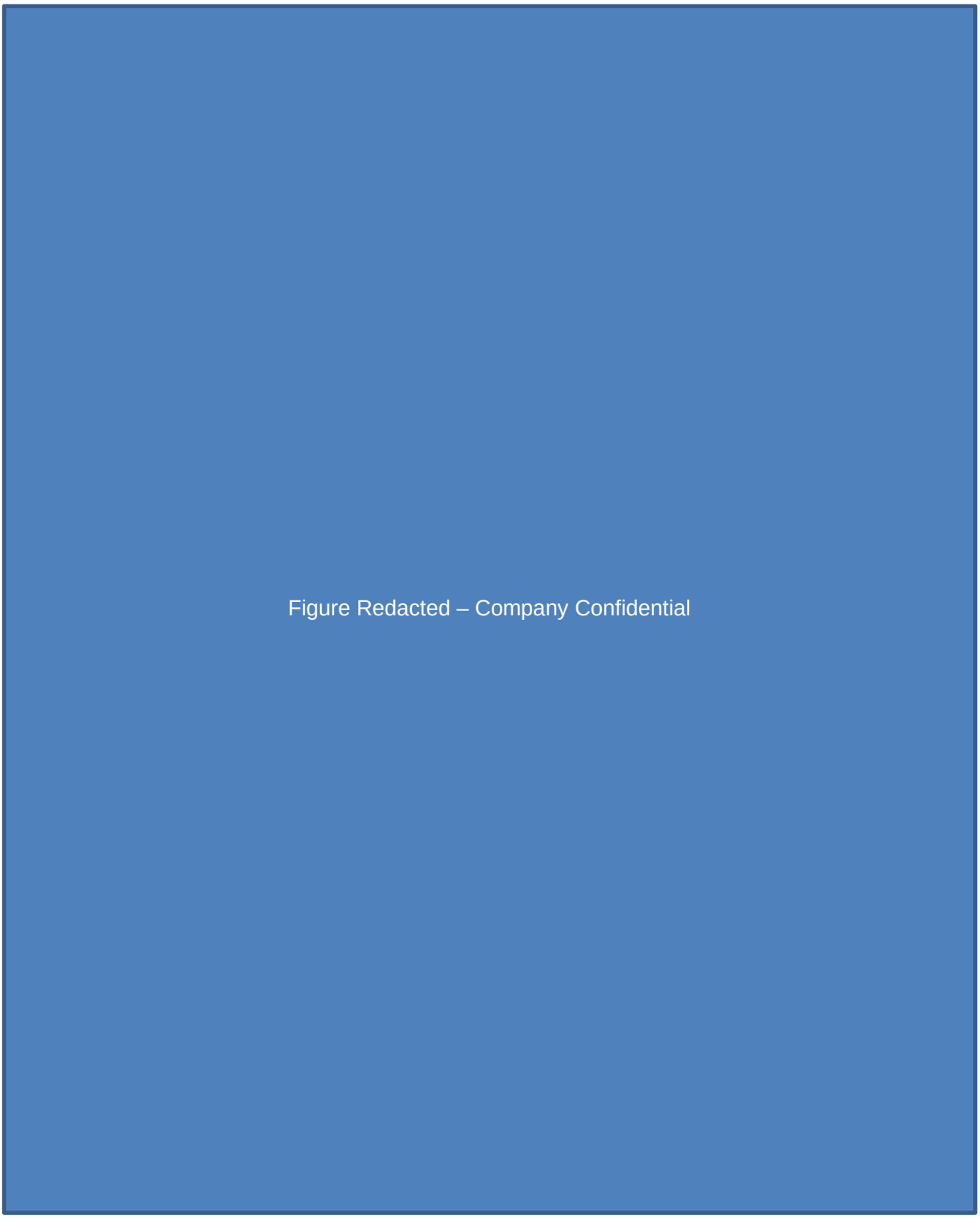


Abbildung 5 - KIEZ02: Flugvorbereitung, einschließlich Routenüberprüfung

KIEZ03: Übergabe am Steigflugende (Top of Climb - ToC) und optimaler Transitflug (Abbildung 6): Am ToC signalisiert das AVMS (Air Vehicle Management System) KIEZ die Übernahme der Routenführung. KIEZ sendet die aktive Route an das AVMS. KIEZ überwacht die Erreichbarkeit der

Ziele. Bei Änderungen erfolgt ein Übergang zu KIEZ04/05. Am Initial Approach Fix (IAF) übergibt das AVMS die Führung von KIEZ.

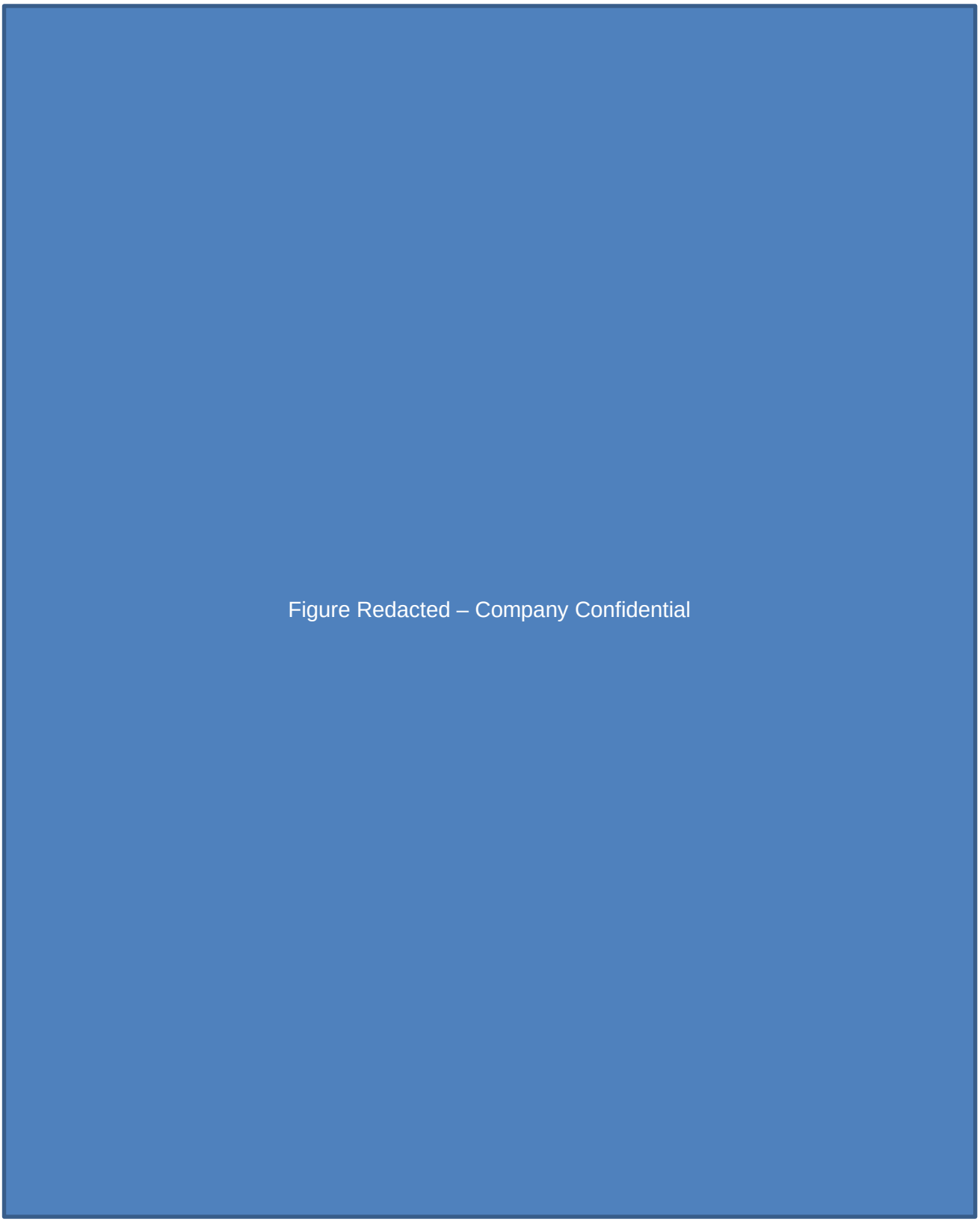


Abbildung 6 - KIEZ03: Übergabe am Steigflugende (Top of Climb - ToC) und optimaler Transitflug

KIEZ04: Management geänderter Bedingungen im halbautomatischen Modus (Abbildung 7): Eine Änderung wird vom AVMS gemeldet. KIEZ plant neu und sendet die optimale Route zur

Genehmigung an den RP. Bei Annahme informiert der RP die ATC und ändert ggf. den Flugplan. KIEZ sendet die neue Route an das AVMS.

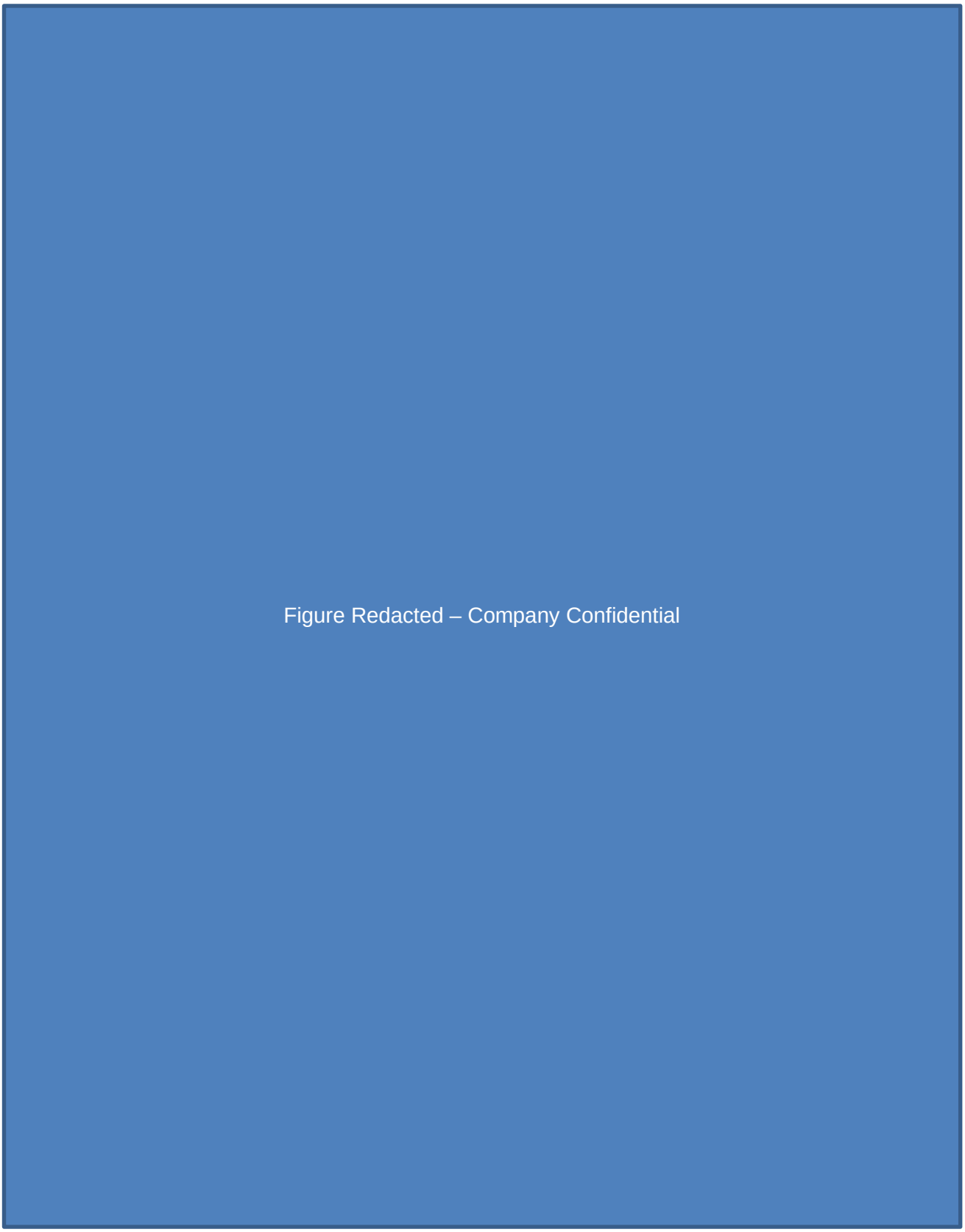


Abbildung 7 - KIEZ04: Management geänderter Bedingungen im halbautomatischen Modus

KIEZ05: Management geänderter Bedingungen im automatischen Modus (Abbildung 8): Eine Änderung wird vom AVMS gemeldet. KIEZ plant neu, setzt die optimale Route als aktuell, sendet sie an AVMS und RP. Der RP kann innerhalb eines Zeitfensters widersprechen. Bei Akzeptanz/keinem Widerspruch informiert der RP die ATC.

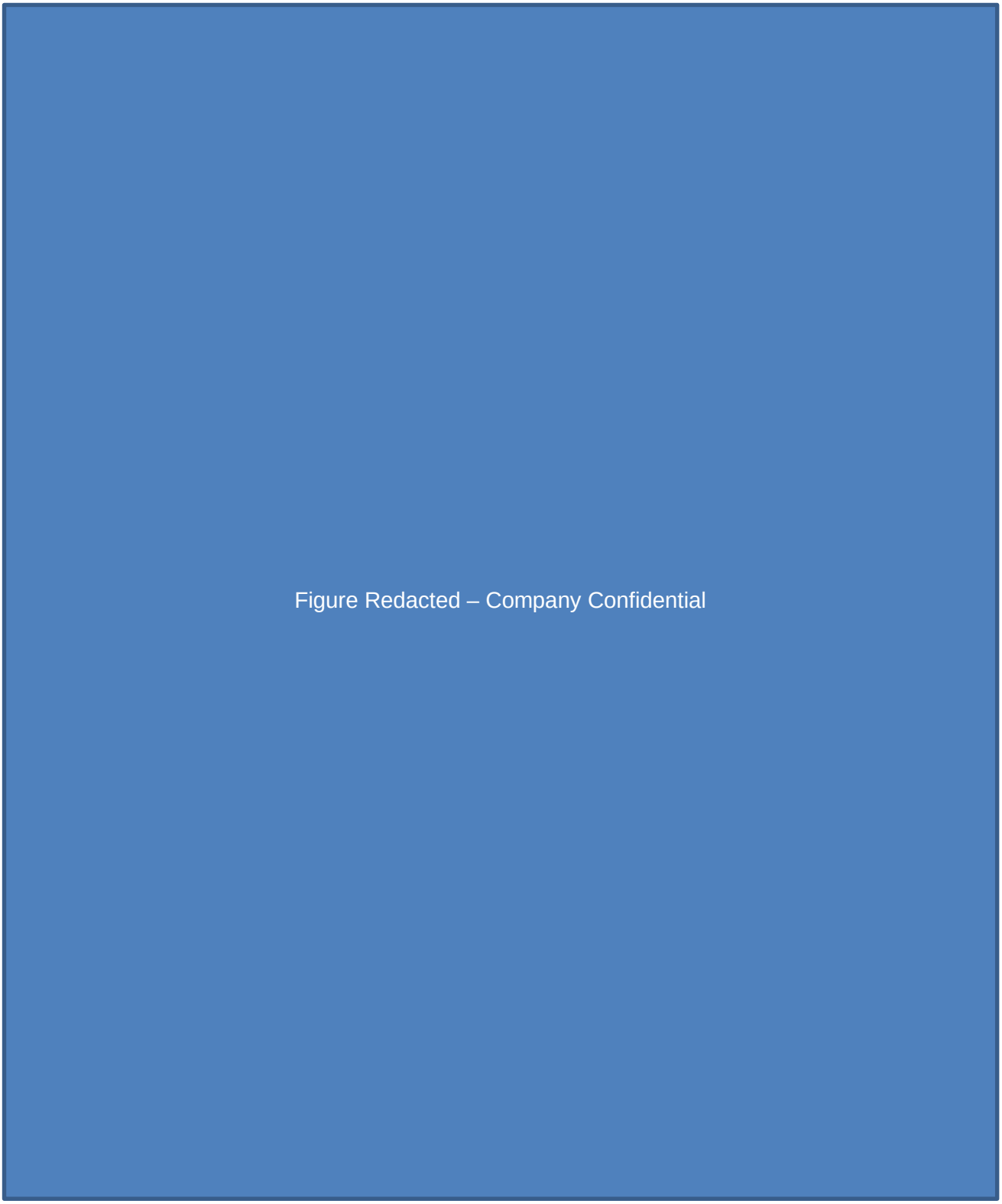


Abbildung 8 - KIEZ05: Management geänderter Bedingungen im automatischen Modus

KIEZ06: Reaktion auf KIEZ-Ausfälle (Abbildung 9): KIEZ-Ausfall erkannt. Wenn KIEZ die aktuelle Routenquelle ist, sendet es eine WARNUNG an das AVMS und gibt keine Routen aus. Das AVMS behält die Route bei, der RP wählt eine Alternative. Ist KIEZ nicht die aktuelle Quelle, sendet es eine ACHTUNG. Der RP folgt der Checkliste. Die Wiederherstellung beinhaltet eine Reinitialisierung und Reaktivierung durch den RP.

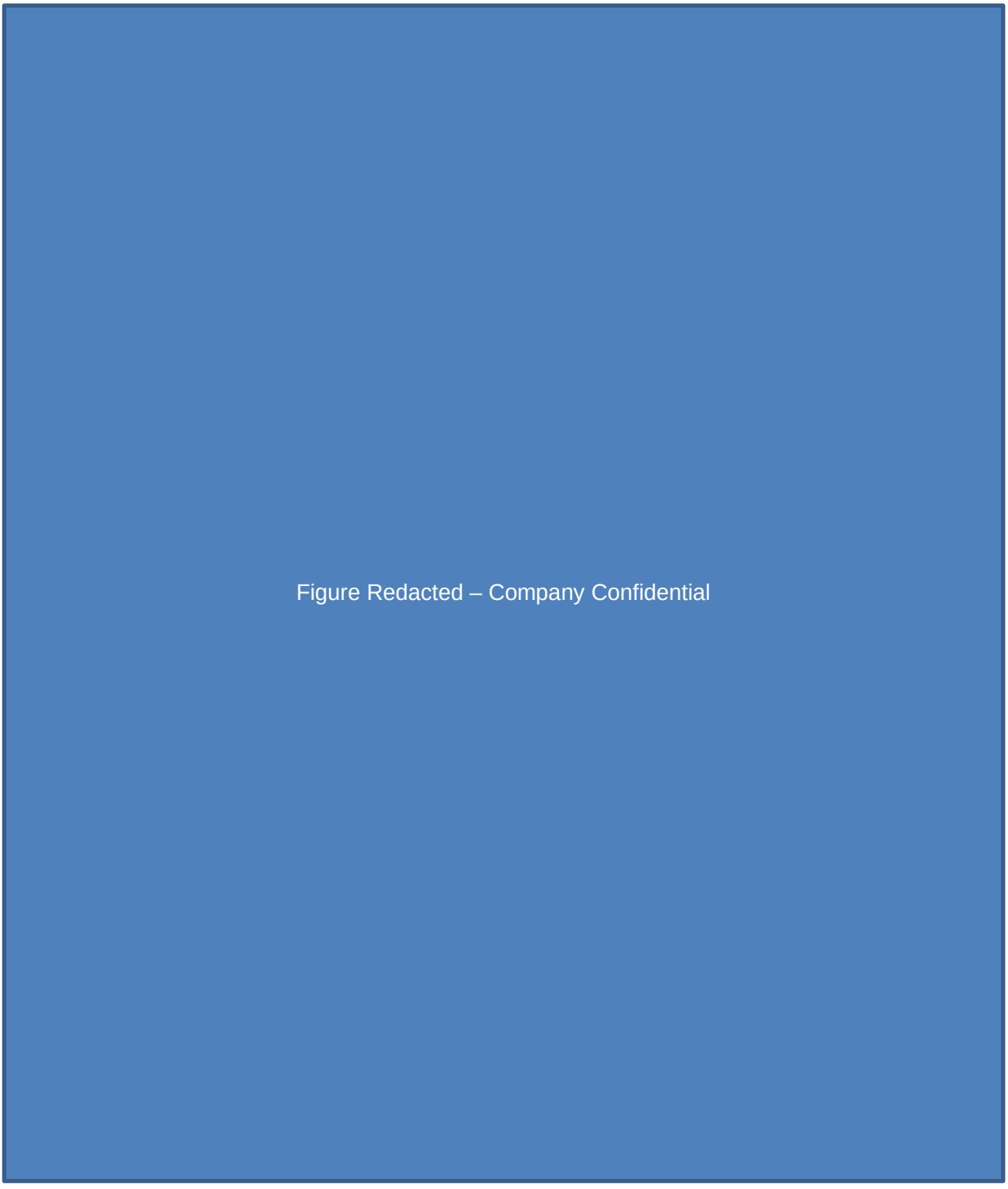


Abbildung 9 - KIEZ06: Reaktion auf KIEZ-Ausfälle

KIEZ07: Reaktion auf Ausfälle zugehöriger Systeme (Abbildung 10): NFZ-Datenquelle nicht verfügbar. KIEZ informiert das AVMS, dass manuelle Aktualisierungen erforderlich sind. Der RP folgt der Checkliste, ändert bei Bedarf KIEZ-Daten und initiiert eine manuelle Neuplanung. Der RP führt manuelle Aktualisierungen durch.

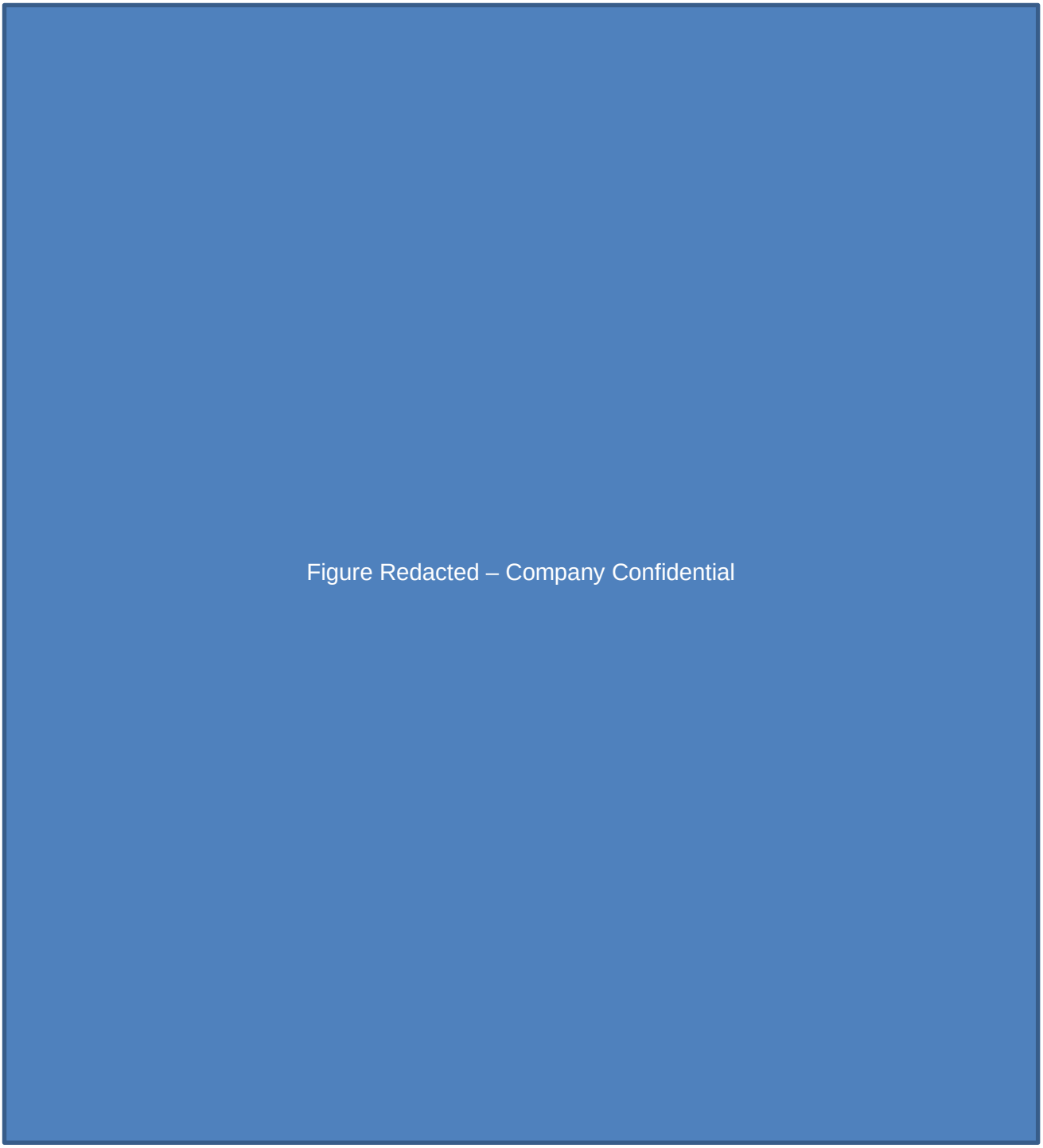


Abbildung 10 - KIEZ07: Reaktion auf Ausfälle zugehöriger Systeme

KIEZ08: KIEZ-Verhalten bei C2LL (Abbildung 11): C2LL tritt ein. KIEZa arbeitet im vollautomatischen Modus, falls von A&ER ausgewählt. Nutzt C2LL-Zielset, falls vorhanden. KIEZg prognostiziert die Route für den RP. Bei Wiederherstellung der C2-Verbindung bleibt KIEZ im C2LL-Modus, bis der RP die Steuerung wieder übernimmt und eine manuelle Neuplanung initiiert.

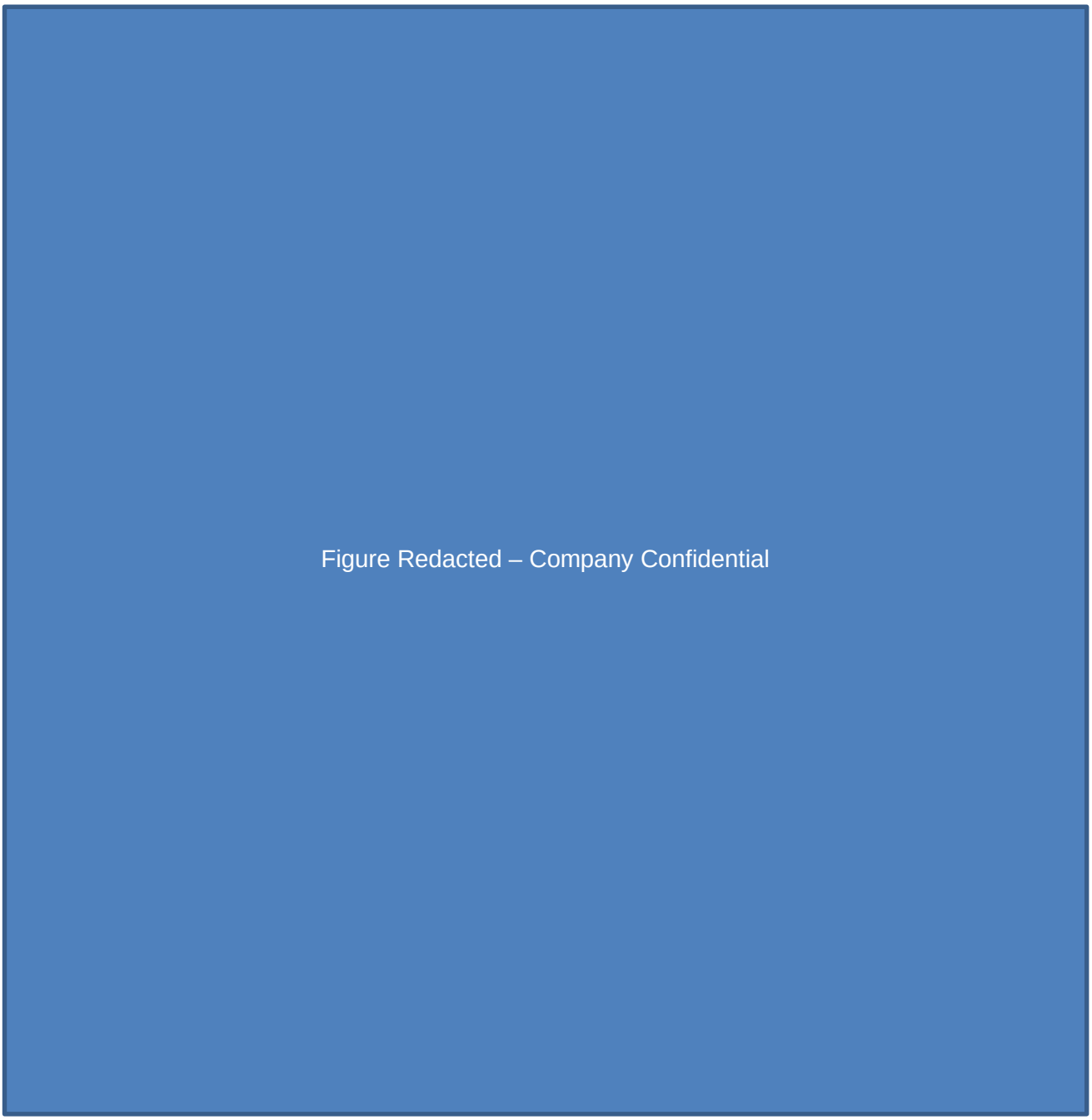


Abbildung 11 - KIEZ08: KIEZ-Verhalten bei C2LL

1.3.2.5 Analyse, Auswirkungen und zukünftige Erweiterungen des vorgeschlagenen Systems

Negative Auswirkungen: Die Akzeptanz durch die Flugsicherung (ATM) für ein automatisiertes Routing als Alternative zu festen Routen bei Verbindungsverlust stellt die einzige identifizierte negative Auswirkung dar.

Positive Auswirkungen: Zu den positiven Aspekten zählen eine reduzierte Arbeitsbelastung des RP, eine verbesserte Routenplanung und Zeitüberwachung (was potenziell zu geringerem Treibstoffverbrauch und besserer Einhaltung von Landeslots führen kann), die Fähigkeit, sehr komplexe Planungssituationen zu bewältigen und somit Missionen zu ermöglichen, die andernfalls

eine übermäßige Arbeitsbelastung verursacht oder zu einem Flugabbruch geführt hätten, sowie die zukünftige Fähigkeit zur Koordination mehrerer Agenten.

Beispiele für Neuplanung: Der CONOPS illustriert die Komplexität der Routenplanung und die Vorteile von KIEZ anhand von Beispielen wie der einfachen Umfahrung statischer NFZ der Erweiterung um Zeitfaktoren, der Umfahrung beweglicher NFZ und der Umfahrung mehrerer beweglicher/fester NFZ. Diese Szenarien verdeutlichen, wie KIEZ durch die Berücksichtigung von Wegpunkten im TPN, zeitlichen Beschränkungen, dynamischen NFZ und deren Bewegungen optimale Routen berechnet und somit den RP entlastet. Die Notwendigkeit, Haltepunkte sorgfältig zu definieren (z.B. durch ein "Holding permitted"-Attribut für Wegpunkte), wird ebenfalls hervorgehoben.

Zukünftige Erweiterungen:

Situational Awareness (SA) Mode: Ermöglicht es dem RP, von der geplanten Route abzuweichen, während KIEZ im Hintergrund die zeitgebundene Route und NFZ entlang des aktuellen Flugpfades überwacht und warnt, falls Missionsziele gefährdet sind.

Multi-Agent (MA) Mode: Ein KIEZ-System koordiniert die Routen mehrerer kollaborativ operierender, mit KIEZ ausgestatteter RPAS (sogenannte Agenten). Bei einer Routenänderung eines Agenten bewertet das koordinierende KIEZ, ob andere Agenten ebenfalls eine Neuplanung benötigen, und sendet gegebenenfalls neue Ziele.

Formating-Agent (FA) Mode: Eine Erweiterung des MA-Modus, bei dem Agenten sich auf einen führenden Agenten ausrichten und zu einer Formation unter einem PIC zusammenschließen können. Dies beinhaltet Protokolle für sichere Staffelung und Kommunikation gemäß.

1.4 Ergebnisse des HAP 3 – KI Demonstratoren / Use Case

1.4.1 AP 3.2 Komponenten mit KI

Arbeitspaket 3.2 (AP3.2) fokussierte auf die Entwicklung der KI-Komponente in zertifizierbarer Weise, die anschließend für Tests und Validierung in AP3.3 zur Verfügung steht. Die Erkenntnisse zu den Zertifizierungsprozessen aus AP1.1 und AP1.2 fanden in diesem Entwicklungsschritt Anwendung, wobei insbesondere die Einhaltung von DO-178C berücksichtigt wurde. Während der CONOPS-Entwicklung in AP2.2 wurden Anforderungen aus dem Dokument abgeleitet, welche anschließend in ein Dokument der obersten Systemanforderungen (Top Level Systems Requirement Document, TLSRD) aufgenommen wurden. Dieser Anforderungssatz wurde in eine bestehende Spezifikation eines Flugmanagementsystems (FMS) integriert. Dies ermöglicht eine einfache Wiederverwendung des entwickelten Systems. Das TLSRD diente anschließend als Spezifikation für die Entwicklung des TPN-Systems. In den folgenden Abschnitten werden das entwickelte System erläutert und die Ergebnisse dieser Entwicklung dargestellt. Das Endergebnis des Arbeitspakets war ein vollständiges System, bereit für die Test- und Demonstrationsphase, die in AP3.3 durchgeführt wurde.

1.4.1.1 Einleitung und Zielsetzung

Im Rahmen des KIEZ-Projekts, das auf die Demonstration neuartiger Algorithmen und vielfältiger Anwendungen künstlicher Intelligenz (KI) abzielt, konzentrierte sich dieses Arbeitspaket auf die Entwicklung eines Pfadfindungsalgorithmus. Dieser Algorithmus, basierend auf Temporal Planning Networks (TPNs), dient der Vermeidung von Nicht-Flug-Zonen (NFZs) und Wetterzonen. Die theoretischen Grundlagen der TPNs sowie die spezifischen Anwendungsfälle und die Softwareimplementierung werden im Folgenden detailliert beschrieben.

1.4.1.2 Theoretische Grundlagen von Temporal Planning Networks

Temporal Planning Networks (TPNs) stellen eine Erweiterung von Simple Temporal Networks (STNs) dar. Sie ergänzen STNs um Entscheidungsknoten und symbolische Nebenbedingungen, zusätzlich zu den bereits vorhandenen zeitlichen Nebenbedingungen. Zeitliche Nebenbedingungen definieren die erlaubte Dauer einer Aktivität oder den zeitlichen Abstand zwischen zwei Aktivitäten. Symbolische Nebenbedingungen hingegen schränken das Verhalten von Aktivitäten durch beliebige Zusicherungen oder Anforderungen ein. Entscheidungsknoten ermöglichen Verzweigungen und somit unterschiedliche Ausführungspfade von Aktivitäten.

1.4.1.3 Anwendungsfall: Flugkorridore und dynamische Zonen

Der primäre Anwendungsfall ist die Berechnung des kürzesten Flugpfades unter Umgehung von NFZs und Wetterzonen. In einem Graphennetzwerk, das aus Wegpunkten und diese verbindenden Kanten besteht, soll der Algorithmus den optimalen Pfad vom Start- zum Zielknoten finden und dabei definierte Zonen meiden. Während NFZs als statisch betrachtet werden, bewegen sich Wetterzonen mit konstanter Geschwindigkeit und Richtung. Beide Zonenarten können mehrere Übertretungsstufen aufweisen, wobei das Durchdringen äußerer Stufen mit Strafkosten verbunden sein kann und das Passieren innerer Stufen strikt verboten sein kann. Zusätzlich können bestimmte Wegpunkte als Points of Interest (POI) definiert werden, die angeflogen werden müssen, oder es können zeitliche Vorgaben für das Passieren von Wegpunkten existieren. Abbildung 12 illustriert ein einfaches TPN mit einem POI und einer sich bewegenden NFZ.

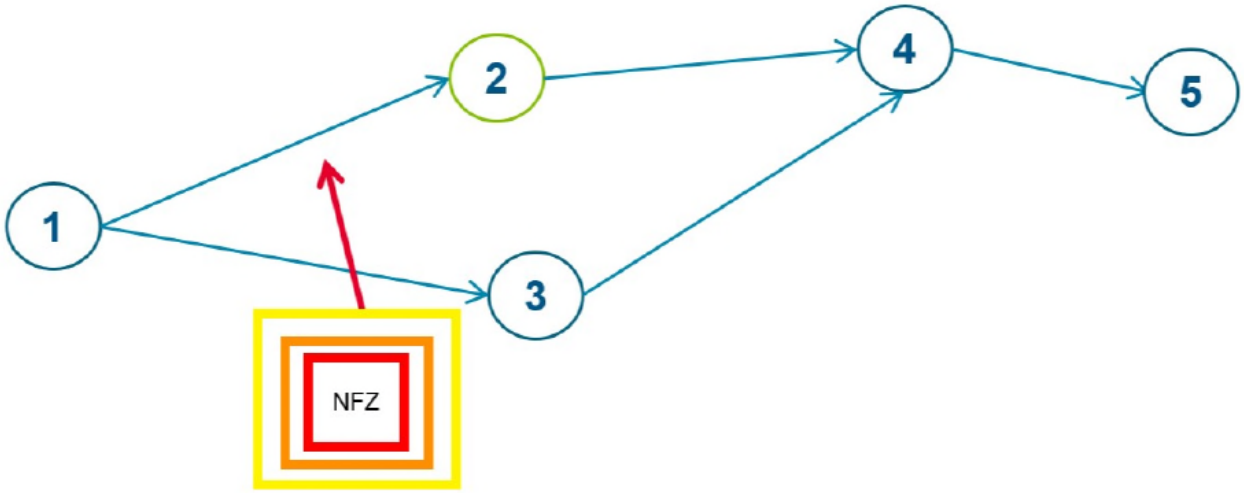


Abbildung 12 - TPN mit einem POI und einer sich bewegenden NFZ

1.4.1.4 Softwarearchitektur und Funktionsweise

Die Architektur des implementierten TPN-Systems ist in Abbildung 13 dargestellt. Ein zentraler Aspekt ist die Konsistenzprüfung des Graphen. Obwohl Algorithmen wie Bellman-Ford negative Zyklen erkennen können, ist dies im vorliegenden Anwendungsfall nicht notwendig, da Kantengewichte die positive Reisedauer zwischen Wegpunkten repräsentieren, wodurch negative Zyklen ausgeschlossen sind. Entsprechend sind der Graph Consistency Checker und der Bellman-Ford-Algorithmus zwar implementiert, werden aber nicht aktiv genutzt.

Figure Redacted – Company Confidential

Abbildung 13 - Die Architektur des implementierten TPN-Systems

Das Herzstück des Algorithmus zur Bestimmung des kürzesten Pfades ist der A∗ Algorithmus. Als Heuristik dient die lineare Distanz zum Zielknoten. Bei der Expansion eines neuen Knotens prüft ein Konsistenz-Checker, ob die mit dem Knoten verbundenen Nebenbedingungen (zeitliche Beschränkungen, POIs) erfüllt sind. Ist dies der Fall, wird die Suche fortgesetzt. Andernfalls wird ein rekursives Backtracking zum letzten möglichen Entscheidungsknoten ausgelöst, von wo aus die Suche neu aufgenommen wird. Der A∗-Algorithmus terminiert, sobald der Zielknoten erreicht ist und alle POIs besucht wurden. Ist letzteres nicht erfüllt, wird ebenfalls das Backtracking aktiviert.

Umgang mit Wetterzonen und NFZs

Statische NFZs, die Kanten blockieren, werden durch unerfüllbare zeitliche Nebenbedingungen modelliert (z.B. Passieren des Knotens vor 0 Sekunden oder nach ∞ Sekunden). Dies führt dazu, dass der blockierte Pfad bei der Suche nicht weiter berücksichtigt wird. Bewegliche Wetterzonen blockieren eine Kante nur für ein bestimmtes Zeitintervall. Der Beginn und das Ende dieser Blockade können mathematisch berechnet werden, indem die Bewegungsgleichung der Zone mit der konstanten Bewegung des Agenten zwischen zwei Wegpunkten gleichgesetzt wird. Um die Zone zu meiden, kann der Agent die Kante entweder vor dem Eintreffen der Zone passieren (durch Erhöhung der Geschwindigkeit) oder nachdem die Zone vorbeigezogen ist (durch Reduzierung der Geschwindigkeit oder Einführen von Wartepunkten). Diese Optionen werden durch temporale Nebenbedingungen und gegebenenfalls eingefügte Zwischenknoten im TPN abgebildet.

Multi-Agenten-Szenarien

In einem Multi-Agenten-Szenario wird der Suchalgorithmus für jeden Agenten ausgeführt, um den jeweils kürzesten Pfad zu finden. Eine zusätzliche Anforderung kann sein, dass die Agenten kollaborieren, um gemeinsam alle definierten POIs zu besuchen. Dieses Problem ist als Multiple Traveling Salesman Problem (MTSP) bekannt und gehört zur Klasse der NP-schweren Probleme. Da die Lösung des MTSP nicht Ziel dieses Projekts war, können Multi-Agenten-Szenarien derzeit nur ohne die Berücksichtigung von POIs gelöst werden.

1.4.1.5 Analyse der Systemleistung

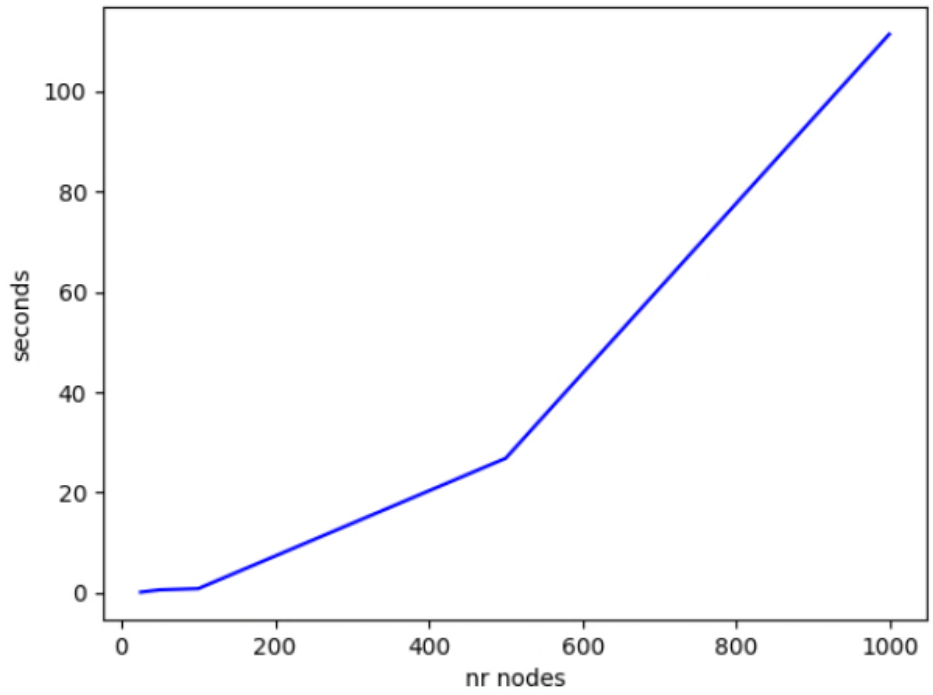
Skalierbarkeit

Der TPN Solution Finder ist für bis zu 999 Knoten ausgelegt. Diese Begrenzung ist primär durch die Implementierung der Ein- und Exportroutinen bedingt, die aktuell nur dreistellige Wegpunkt-IDs unterstützen. Eine Anpassung dieser Routinen würde eine Erhöhung der Knotenzahl ermöglichen. Die Anzahl der NFZs und der zeitlichen Nebenbedingungen ist prinzipiell nicht limitiert, jedoch wirkt sich eine hohe Anzahl negativ auf die Performanz aus. Die Anzahl der Kanten ist ebenfalls nicht begrenzt, sollte aber in einem sinnvollen Verhältnis zur Knotenzahl stehen.

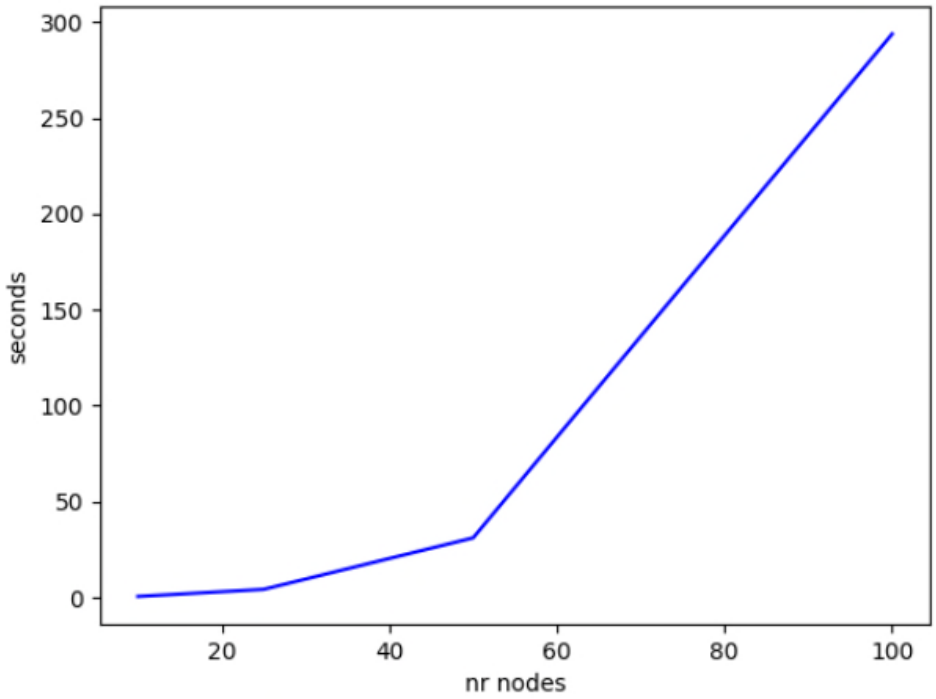
Berechnungsleistung

Der TPN Solution Finder wurde als Technologiedemonstrator und Prototyp entwickelt, wobei der Fokus nicht primär auf der Optimierung der Rechenleistung lag. Die Wahl der Programmiersprache Python, bekannt für schnelle Entwicklung und gute Lesbarkeit statt für maximale Speicher- oder Zeiteffizienz, trägt hierzu bei. Im Vergleich zu C++ ist die Ausführungszeit in Python für denselben Algorithmus typischerweise um einen Faktor 10 bis 100 langsamer. Darüber hinaus ist das rekursive Backtracking auf eine Weise implementiert, die teure tiefe Kopien des gesamten Suchraums verwendet, was die Performanz weiter beeinträchtigt. Eine manuelle Verwaltung des Backtrackings durch Zurücksetzen der Änderungen im selben Suchraum könnte hier Verbesserungen bringen. Auch Optimierungen des A∗-Algorithmus selbst bieten weiteres Potenzial.

Die Performanz skaliert exponentiell mit der Anzahl der Knoten, wie in den Abbildungen 14 und 15 dargestellt. Szenarien mit NFZs zeigen eine schlechtere Performanz, da blockierte Kanten durch NFZs nicht in der Heuristik berücksichtigt werden, was zu vermehrtem Backtracking führt.



Abbildungen 14 - Performanz ohne NFZs



Abbildungen 15 - Performanz mit NFZs

1.4.1.6 Zusammenfassende Bewertung

Die Architektur, der Algorithmus und die unterstützten Funktionen des TPN Solution Finders wurden erfolgreich implementiert. Die Analyse der Berechnungsleistung und Skalierbarkeit zeigt, dass der aktuelle prototypische Aufbau des Systems zu Leistungseinschränkungen führt, insbesondere bei komplexen Szenarien.

1.4.2 AP 3.3 Komplexe Anwendungen

Dieser Abschnitt fasst die Ergebnisse der Demonstrationskampagne zusammen, die durchgeführt wurde, um die Machbarkeit der Hauptanforderungen an das TPN-System im Rahmen des Arbeitspakets 3.3 zu validieren. Die Szenarien, die diese Anforderungen abdecken und diesen Abschnitt strukturieren, basieren auf vordefinierten Beschreibungen.

1.4.2.1 Szenario 1: Automatische Routenwahl und Reaktion auf statische NFZs

Beschreibung des Szenarios

Im ersten Szenario wurde der Automatisierungsgrad auf 4 eingestellt. Der Demonstrationsfall begann mit drei verfügbaren Routen, aus denen KIEZ die beste Route (Route 1) auswählen sollte. Während des Fluges auf dem ersten Streckenabschnitt wurden dem KIEZ-System durch das Wettervermeidungssystem zwei statische No-Fly-Zones (NFZs) übermittelt. Diese NFZs machten zwei der drei ursprünglichen Routen (Route 1 und Route 2) nicht mehr verfügbar. Route 1 wurde durch eine Verletzung der Stufe 2 und Route 2 durch eine Verletzung der Stufe 1 blockiert. Es wurde erwartet, dass KIEZ automatisch eine Neuplanung initiiert und die Route wählt, die die geringere Verletzungsstufe aufweist (Route 2). Eine schematische Darstellung der Routen und NFZs ist Teil der Szenariendefinition (siehe Abbildung 16).

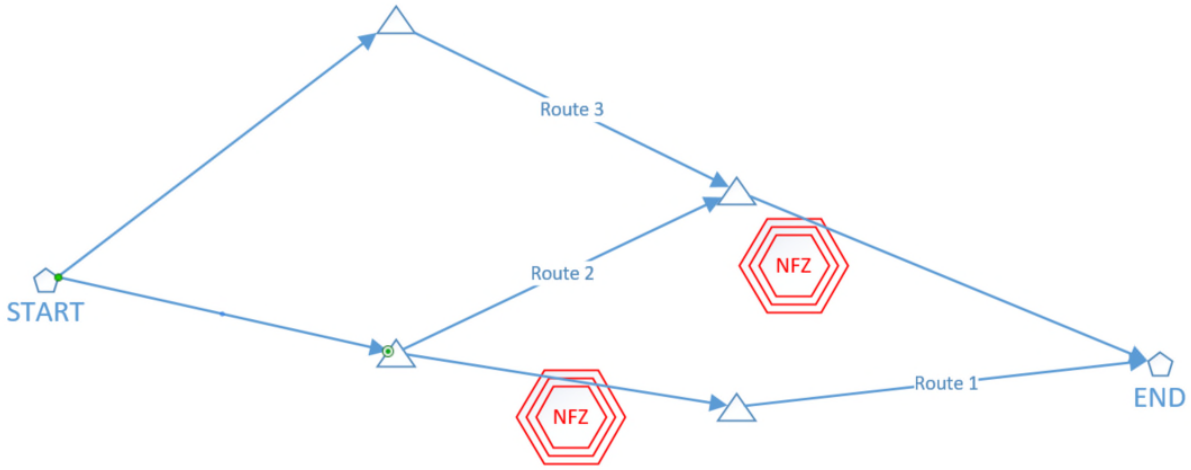


Abbildung 16 - Szenario 1

Ergebnisse der Demonstration

Ursprüngliche Planung:

Zu Beginn wählte das System den Pfad 001-003-005-006 als initialen Plan aus. Diese Wahl basierte darauf, dass dieser Pfad die kürzeste Distanz aufwies und keine weiteren Einschränkungen wie Points of Interest (POIs), NFZs oder zeitliche Nebenbedingungen vorhanden waren. Die initiale Planung mit dem Netzwerk und der simulierten Flugroute ist dargestellt (siehe Abbildung 17).

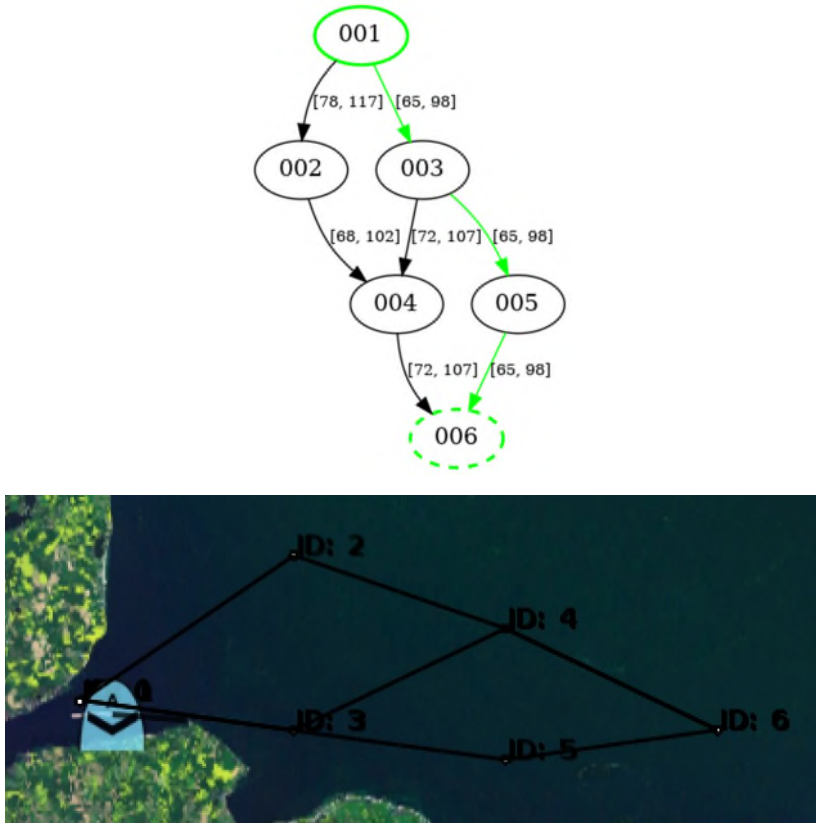


Abbildung 17 - Initiale Planung Szenario 1

Neuplanung:

Während der Simulation, auf dem ersten Streckenabschnitt, erschienen zwei neue NFZs, die vom Wettervermeidungssystem bereitgestellt wurden. Die größere NFZ blockierte die Kante 003-005 mit Verletzungen der äußeren und mittleren Ebene, während die kleinere NFZ die Kante 004-005 nur mit einer Verletzung der äußeren Ebene beeinträchtigte. Da kein anderer, unblockierter Pfad zum Ziel existierte und die Strafe für die Verletzung lediglich einer äußeren Ebene deutlich geringer war als die für eine mittlere Ebene, wurde der Pfad 001-003-004-006 als optimaler neuer Plan gewählt. Interessanterweise flog der Agent in der Simulation ein Warteschleifenmuster (Loiter Pattern) am Wegpunkt WP3, obwohl dies nicht explizit vom Planungssystem befohlen wurde, was auf leichte Abweichungen zwischen geplanter und simulierter Zeit hindeutet. Auch diese Neuplanung mit der aktualisierten Netzwerkansicht und der Simulation ist visualisiert (siehe Abbildung 18).

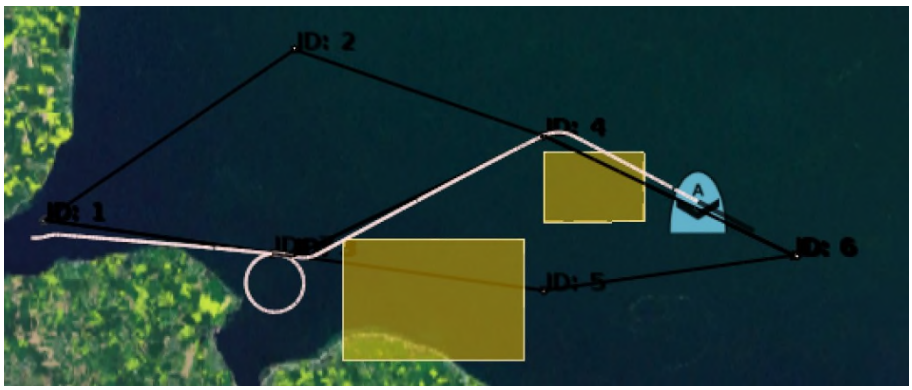


Abbildung 18 - Neuplanung Planung Szenario 1

1.4.2.2 Szenario 2: Umgang mit beweglichen NFZs und POI-Berücksichtigung mit RP-Genehmigung

Beschreibung des Szenarios

Für das zweite Szenario wurde der Automatisierungsgrad auf 2 festgelegt. Der Ausgangsfall umfasste vier Routen und einen zu besuchenden POI. KIEZ sollte die beste Route (Route 2) auswählen, die den Besuch des POIs sicherstellt, und diese Route dem Remote Pilot (RP) zur Genehmigung vorlegen. Nach erfolgter Genehmigung durch den RP und während des Fluges auf dem ersten Abschnitt wurde eine sich bewegende NFZ durch das Wettervermeidungssystem an KIEZ übermittelt, die alle vier Routen zu einem bestimmten Zeitpunkt kreuzte. Diese NFZ sollte ebenfalls dem RP zur Genehmigung für die Aufnahme in das TPN vorgelegt werden. Es wurde erwartet, dass die Genehmigung des RP die gewählte Route 2 zu dem Zeitpunkt, an dem das RPA (Remotely Piloted Aircraft) den betroffenen Abschnitt passieren würde, unbenutzbar macht. KIEZ sollte daraufhin automatisch eine Neuplanung auslösen, eine Route finden, die die NFZ nicht schneidet, aber den POI-Besuch sicherstellt, und diese neue Route dem RP zur Genehmigung senden. Diese neue Route erforderte ein Halten an einem Wegpunkt für eine gewisse Zeit. Nach Genehmigung sollte das RPA die Route wechseln. Eine schematische Darstellung der Routen, des POI und der beweglichen NFZ ist gegeben (siehe Abbildung 19).

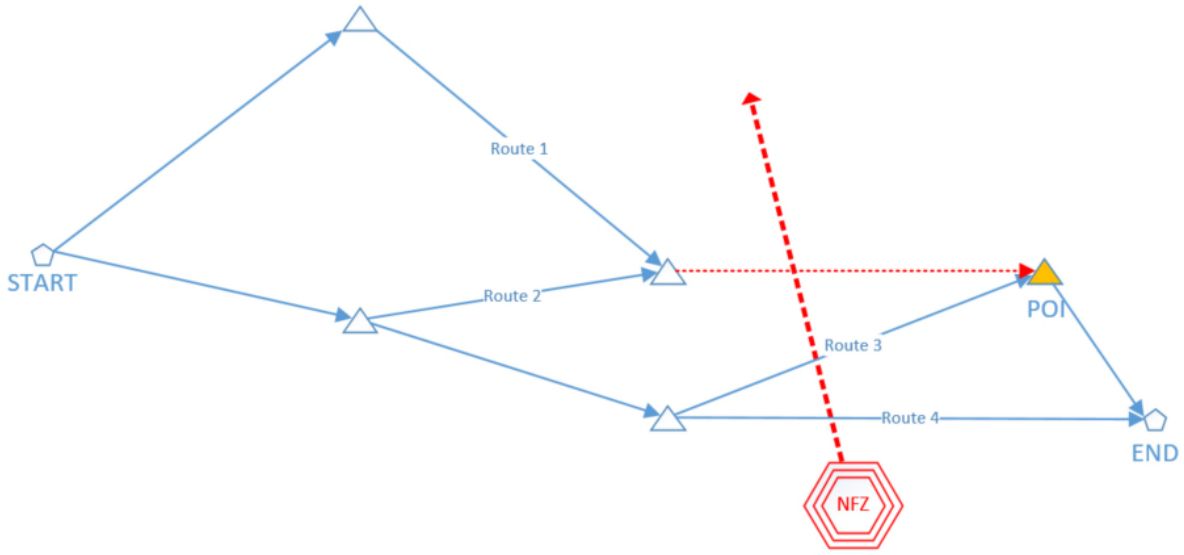


Abbildung 19 - Szenario 2

Ergebnisse der Demonstration

Ursprüngliche Planung:

Da anfangs keine NFZ vorhanden war, wurde der kürzeste Pfad 001-003-004-006, der auch den POI am Wegpunkt WP6 besuchte, ausgewählt. Die initiale Routenwahl und der POI sind im Plan und in der Simulation dargestellt (siehe Abbildung 20).

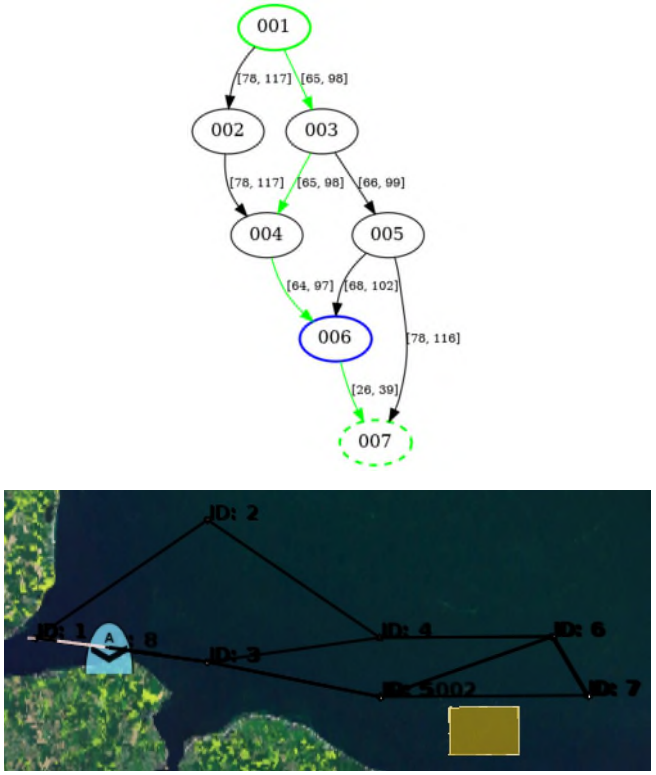


Abbildung 20 - Initiale Planung Szenario 2

Neuplanung:

Während der Simulation, beim Passieren des ersten Streckenabschnitts, tauchte eine sich bewegende NFZ auf, die die Kanten 005-006 und 004-006 temporär blockierte. Da der Agent am Wegpunkt WP4 länger hätte warten müssen als am WP5, um die blockierte Kante davor zu passieren, wurde der distanzmäßig längere Pfad über WP5 als bester Pfad gewählt. Um die Kante ohne Verletzung der NFZ zu passieren, befahl das Planungssystem ein Warten am WP5 (referenziert als 005w006_2 im Plan). Diese geplante Wartezeit wurde in der Simulation ausgeführt, indem der Agent ein Warteschleifenmuster am WP5/5002 flog. Dies ist in einer Simulationsansicht gezeigt (siehe Abbildung 21). Der komplexe Plan mit den Warteknoten ist ebenfalls grafisch dargestellt (siehe Abbildung 22).

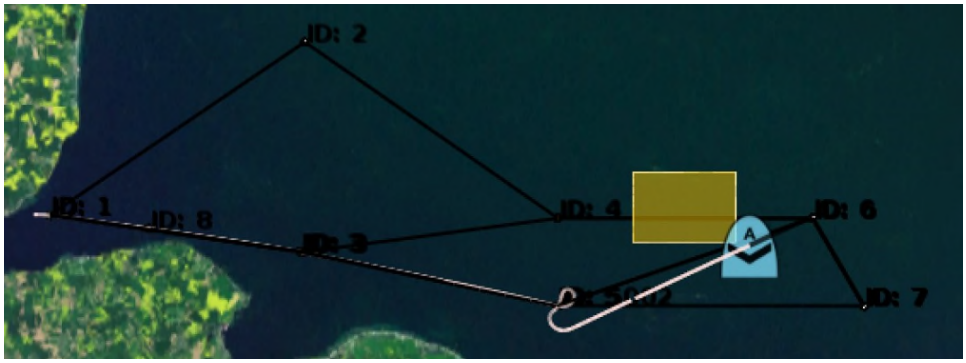


Abbildung 21 - Neuplanung Planung Szenario 2

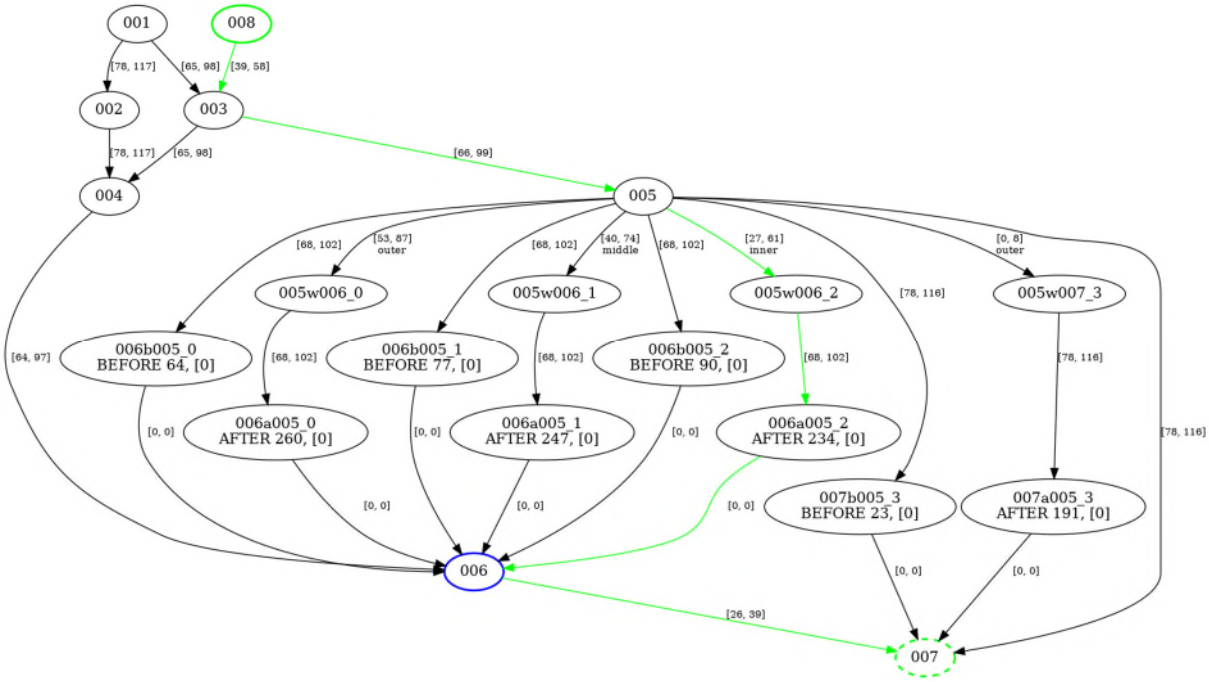


Abbildung 22 - Komplexe Plan mit den Warteknoten

1.4.2.3 Szenario 3: Mehrfache Neuplanung bei statischen NFZs und variablen Automatisierungsgraden

Beschreibung des Szenarios

Das dritte Szenario begann mit einem Automatisierungsgrad von 0 und drei Routen. KIEZ wählte die beste Route (Route 3) und sendete sie zur Genehmigung an den RP, welcher sie genehmigte. Auf dem ersten Flugabschnitt wurde eine statische NFZ übermittelt, die Route 3 schnitt. Nach Genehmigung der NFZ-Aufnahme durch den RP wurde Route 3 unbenutzbar. KIEZ sollte automatisch neu planen, eine nicht-kollidierende Route finden und diese dem RP zur Genehmigung senden. Sobald die Genehmigung erteilt war, sollte das RPA die Route wechseln. Anschließend änderte der RP den Automatisierungsgrad auf 4. Daraufhin wurde eine weitere NFZ an KIEZ gesendet, die die aktuelle Route schnitt. Diese NFZ wurde dem RP zur Genehmigung für die Aufnahme in das TPN gesendet. KIEZ sollte dann automatisch eine Neuplanung auslösen, eine neue Route finden, die alle Einschränkungen erfüllt, und diese automatisch genehmigen, woraufhin das RPA auf Route 1 wechseln sollte. Das Szenario-Setup mit den Routen, dem POI und den NFZs ist illustriert (siehe Abbildung 23).

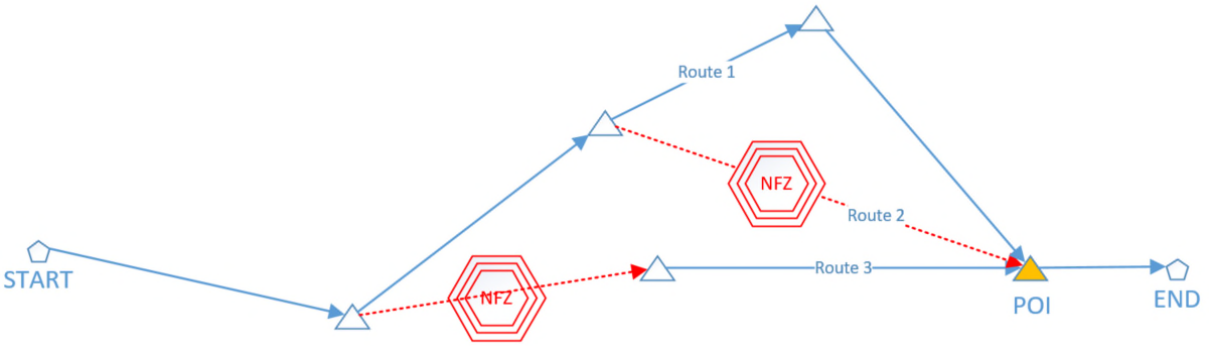


Abbildung 23 - Szenario 3

Ergebnisse der Demonstration

Ursprüngliche Planung:

Der Pfad 001-002-004-006-007 wurde als bester Plan gewählt, da er die kürzeste Distanz aufwies und im initialen Szenario keine weiteren Einschränkungen bestanden. Eine grafische Darstellung dieses Plans ist vorhanden (siehe Abbildung 24).

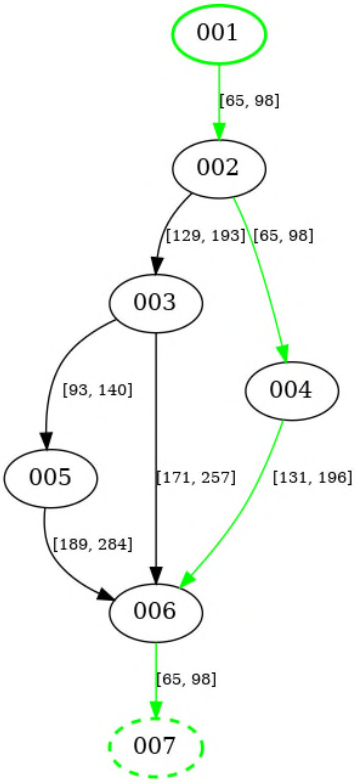
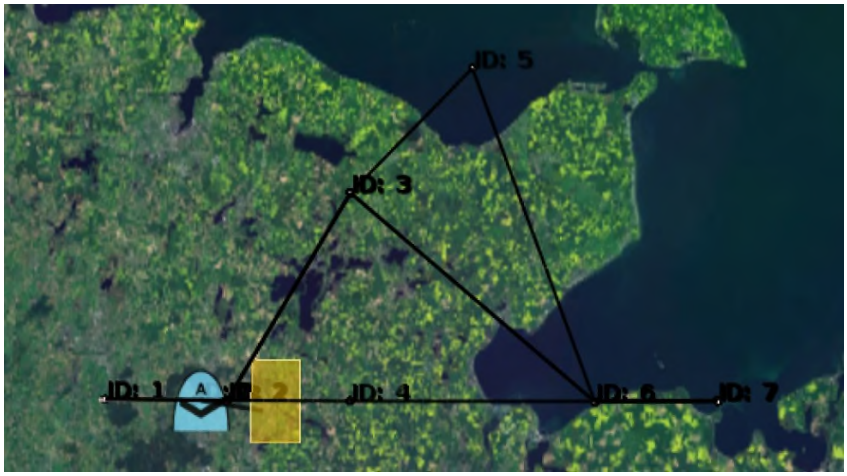


Abbildung 24 - Initiale Planung Szenario 3

1. Neuplanung:

Beim Passieren der ersten Kante wurde eine NFZ vom Wettervermeidungssystem bereitgestellt, die die Kante 002-004 blockierte. Dies löste die erste Neuplanung aus. Der Pfad 008-002-003-006-007 wurde nun als bester Plan gewählt, da er die NFZ umging und eine kürzere Distanz als der Pfad um WP005 bot. Der Wegpunkt WP008 wurde während der Neuplanung als neue Startposition des Agenten erstellt. Der Plan und die Simulation sind visualisiert (siehe Abbildung 25).



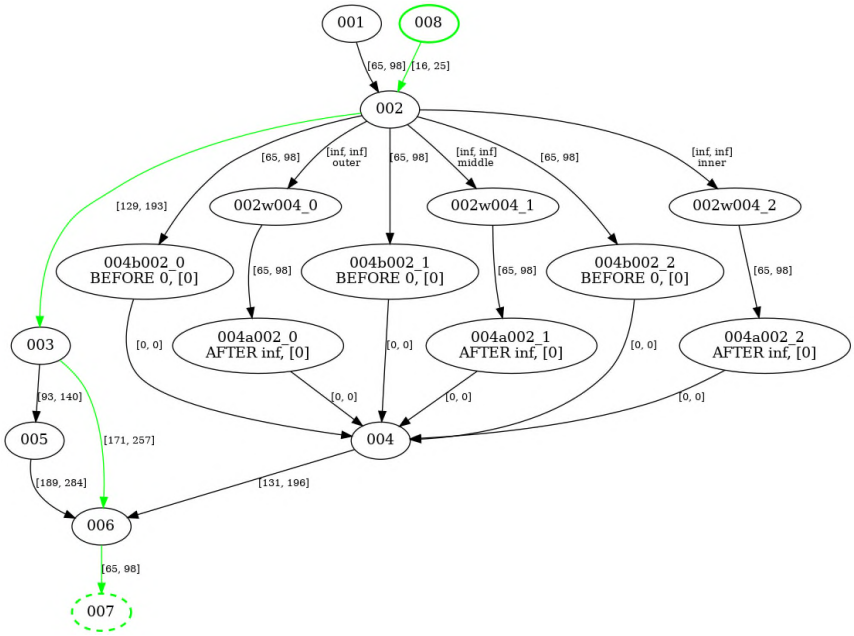
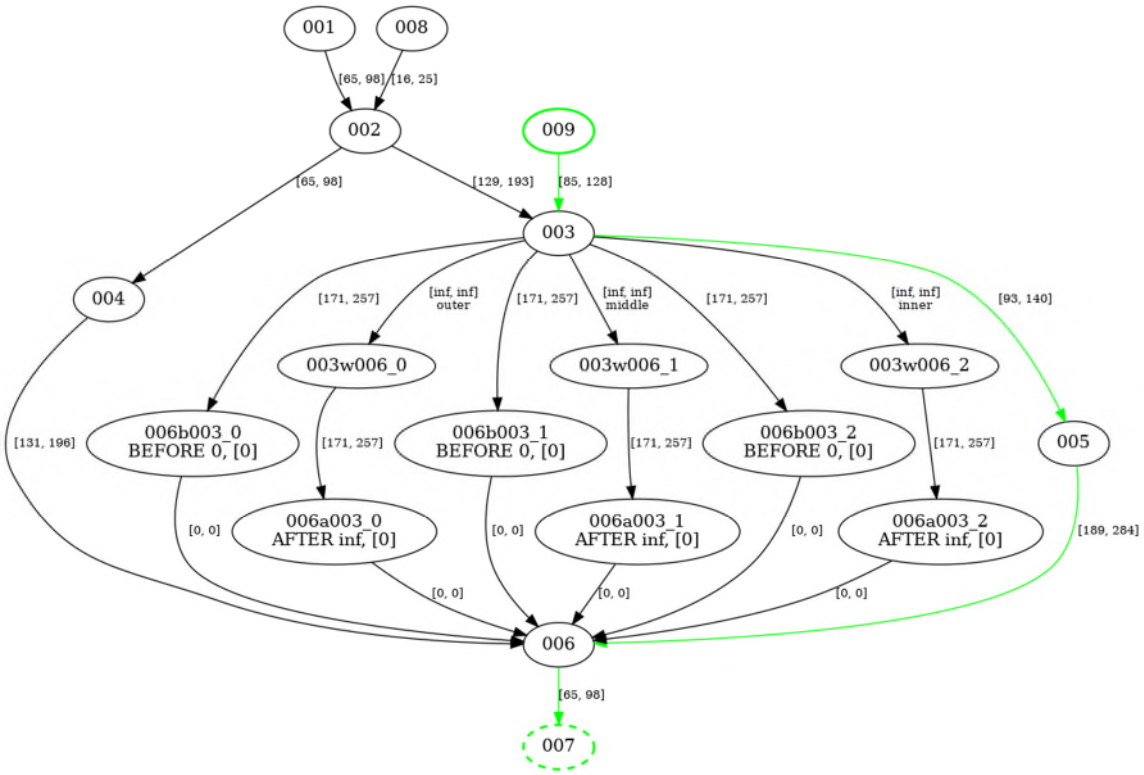


Abbildung 25 – 1. Neuplanung Planung Szenario 3

2. Neuplanung:

Während des Passierens der Kante 002-003 wurde eine zweite statische NFZ bereitgestellt, die die Kante 003-006 blockierte. Dies löste eine weitere Neuplanung aus. Nun wurde der längste Pfad um WP005 als bester Plan gewählt, da er beide statischen NFZs umging. Die Ausführung dieses Plans ist in einer Abbildung der Simulation und des Plans gezeigt (siehe Abbildung 26).



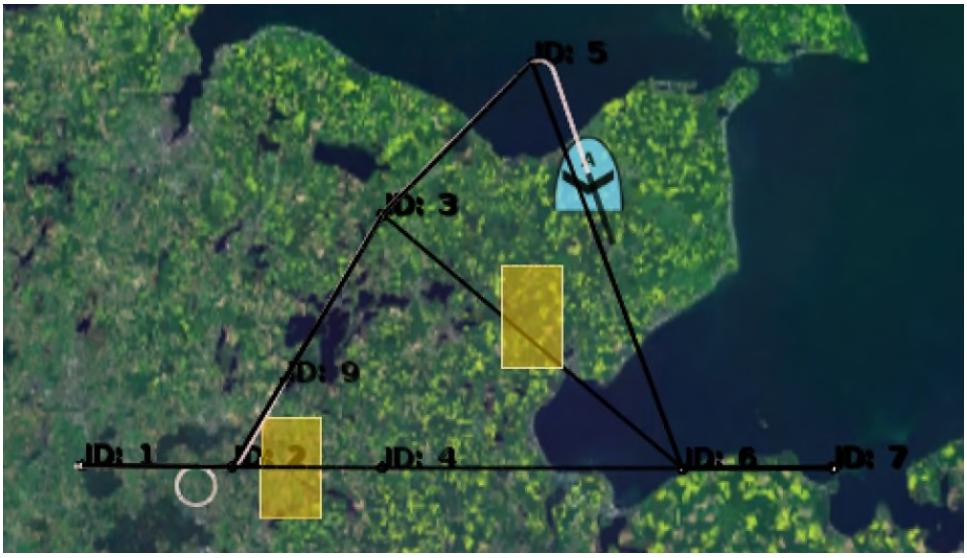


Abbildung 26 – 2.Neuplanung Planung Szenario 3

1.4.2.4 Szenario 4: Manuelle Eingabe von Zeitbeschränkungen und RP-Ablehnung von Routenoptionen

Beschreibung des Szenarios

Das vierte Szenario startete mit einem Automatisierungsgrad von 0 und fünf Routen. KIEZ wählte eine Route (Route 2), die alle Einschränkungen erfüllte, und sendete sie zur Genehmigung an den RP, der sie genehmigte. Das RPA flog die Route. Dann gab der RP manuell eine Zeitbeschränkung (Temporal Constraint, TC) in das TPN ein. KIEZ sollte eine Neuplanung auslösen und dem RP mehrere Routenoptionen zur Genehmigung senden (Route 2 war nicht mehr möglich). Der RP lehnte alle diese Routenoptionen (Routen 3, 4 und 5) ab. KIEZ sollte diese aus dem Lösungsraum entfernen, eine Neuplanung auslösen und eine Route (Route 1) dem RP zur Genehmigung senden. Der RP genehmigte die Route, woraufhin das RPA die Route wechseln sollte. Eine schematische Darstellung des Szenarios mit dem Wegpunkt mit TC ist gegeben (siehe Abbildung 27).

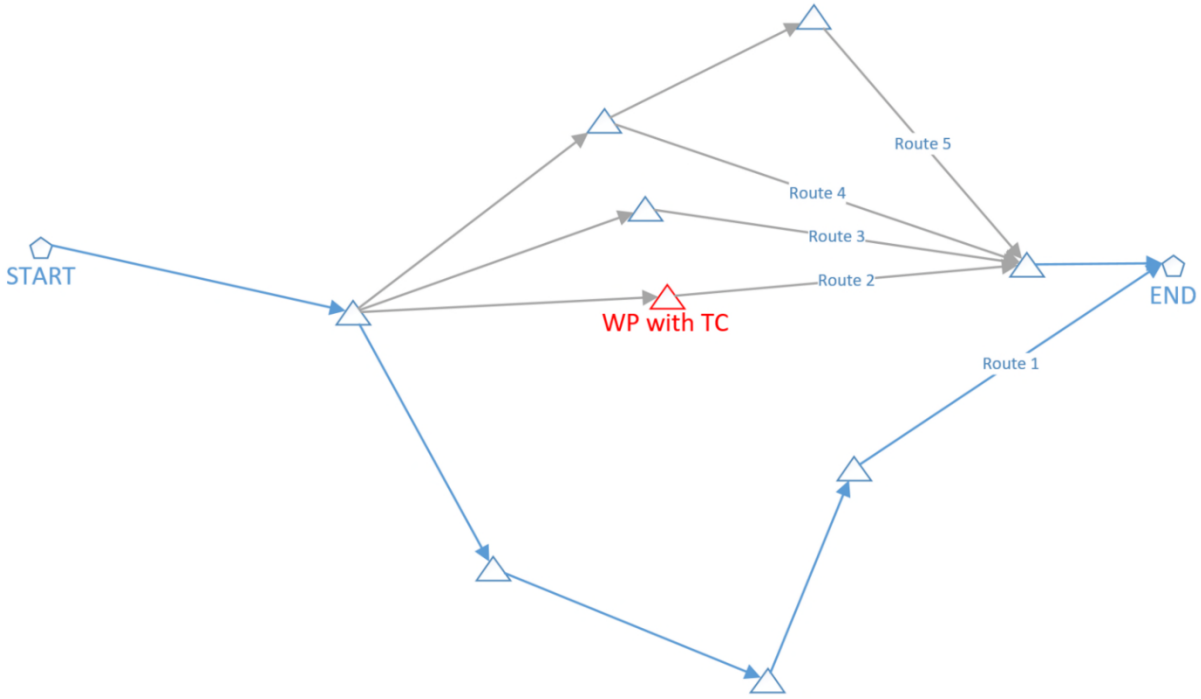


Abbildung 27 - Szenario 4

Ergebnisse der Demonstration

Ursprüngliche Planung:

Der Pfad 001-002-011-006-007 wurde als bester Plan durch den Graphen berechnet, da er die kürzeste Distanz aufwies. Diese Planung ist grafisch dargestellt (siehe Abbildung 28).

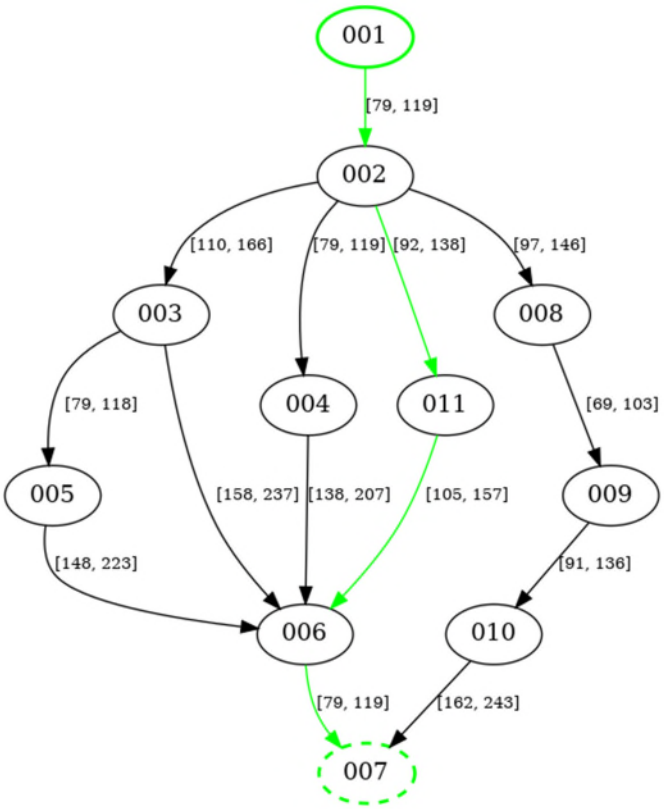


Abbildung 28 - Initiale Planung Szenario 4

Neuplanung:

Während der Simulation, als der Agent die erste Kante passierte, wurde vom RP eine neue zeitliche Einschränkung zum WP011 hinzugefügt. Diese besagte, dass WP011 vor 100 Sekunden passiert werden müsse, was in diesem Szenario nicht machbar war. Deshalb wurden die drei Pfade um die Wegpunkte WP003, WP004 und WP005 als beste Pläne ausgewählt. Der RP lehnte jedoch all diese Pfade ab, was sofort eine weitere Neuplanung auslöste. Nun wurde der längste Pfad um die Wegpunkte WP008-WP009 und WP010 als bester Plan zurückgegeben, da er der einzig verbleibende Pfad war, der machbar und nicht abgelehnt worden war. Die finale Routenwahl und der resultierende Flugpfad sind abgebildet (siehe Abbildung 29).

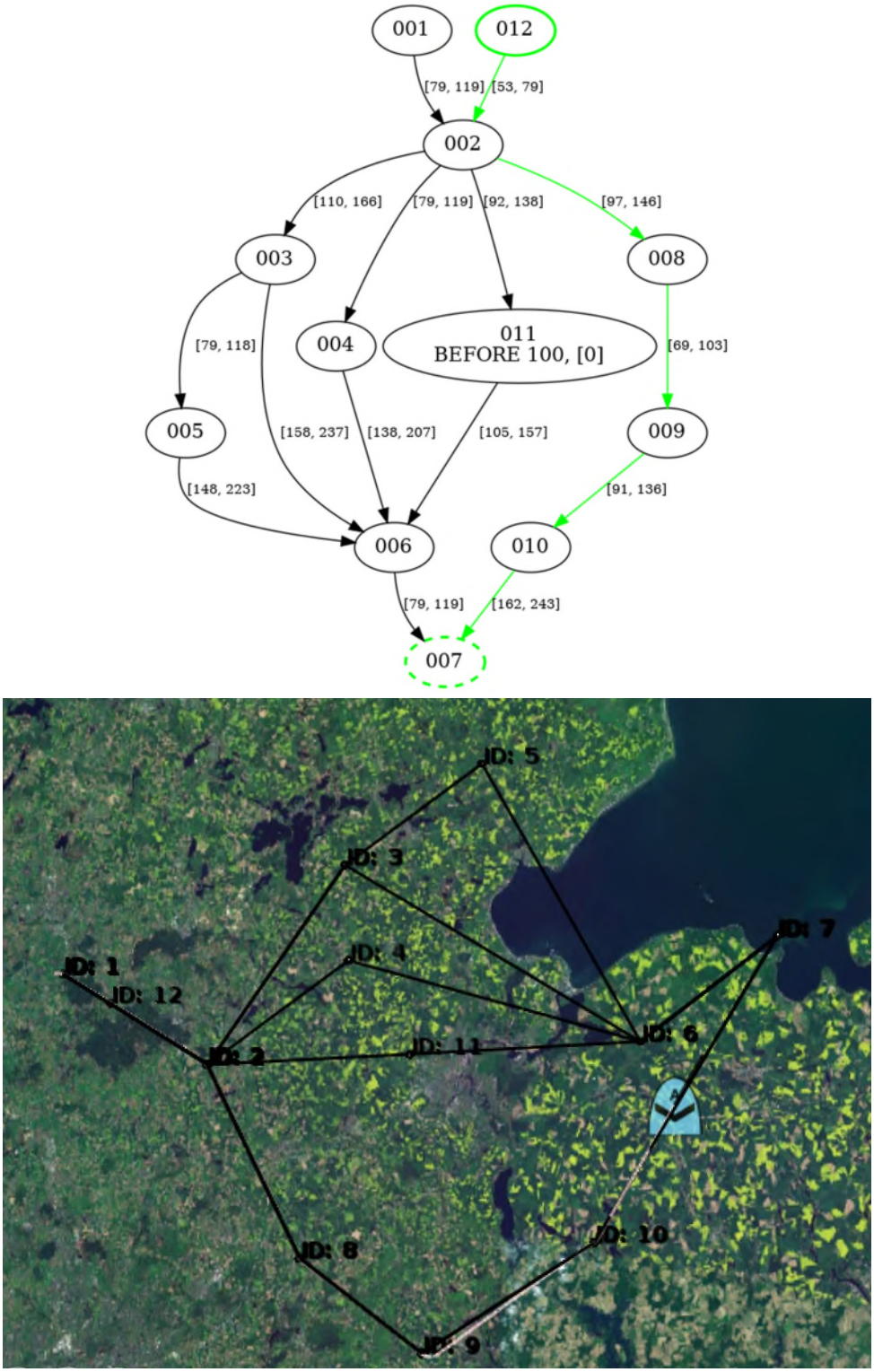


Abbildung 29 – Neuplanung Planung Szenario 4

1.5 Ergebnisse des HAP 4 – Analyse

1.5.1 AP 4.2 Verifikation & Validierung / 4.3 Methodik der Zertifizierung

Wie in Abschnitt 2.9 dargelegt, konzentrierten sich die Arbeitspakete 4.2 und 4.3 (AP4.2 und AP4.3) zunächst auf das KI-Konzeptpapier der EASA und dessen Nützlichkeit zur Unterstützung unserer Zertifizierungsbemühungen. Airbus lieferte Input für die Zielsetzungen, die der EASA präsentiert

werden sollten. Diese wurden anschließend während drei Abstimmungstreffen mit der EASA (EASA Alignment Meetings) detailliert diskutiert. Während dieser Diskussionen stellte sich heraus, dass dieser Ansatz bei Weitem zu kompliziert und nicht erforderlich war. Stattdessen wurde beschlossen, einen traditionelleren Zertifizierungsansatz zu verfolgen. Die von der EASA erhaltene Information lautete, dass der traditionelle Zertifizierungsansatz für unsere Anforderungen angemessen sei. Dieser Ansatz wurde der EASA in diesen Abstimmungstreffen vorgestellt, und die strategische Ausrichtung der Zertifizierung wurde gemeinsam von Airbus und der EASA vereinbart. Diese Vereinbarung bildete die Grundlage für die Arbeitspakete AP4.4 und AP4.5.

1.5.2 AP 4.4/4.5 Abgleich mit EASA / Neue Methodik für die Zertifizierung von KI

Das Arbeitspaket 4.4/4.5 konzentrierte sich auf die Erstellung eines Plans für die Softwareaspekte der Zertifizierung (PSAC) für ein zukünftiges, auf Temporal Planning Networks (TPN) basierendes Softwareprodukt. Ziel war es, die grundlegenden Aktivitäten darzulegen, die erforderlich sind, um eine Zertifizierung gemäß DO-178C und relevanter Ergänzungen wie DO-330 (Software Tool Qualification Considerations), DO-332 (Object-Oriented Technology) und DO-333 (Formal Methods) zu erreichen. Obwohl die KIEZ-TPN-Softwarekomponenten zunächst als Prototyp für Demonstrationszwecke implementiert wurden und daher nicht direkt zertifizierungsfähig sind, können Teile der erzeugten Software-Lebenszyklusdaten, insbesondere Systemanforderungen und Verifikationsartefakte, für zukünftige Zertifizierungsbemühungen wiederverwendet werden.

1.5.2.1 System- und Softwareüberblick

Das KIEZ TPN-System dient primär dazu, ein temporales Planungsnetzwerk zu erstellen, das eine Auswahl von Routen beschreibt, und daraus die optimale Route auszuwählen. Dieses Netzwerk berücksichtigt nicht nur räumliche, sondern auch zeitliche Aspekte und diverse Randbedingungen wie statische und dynamische No-Fly-Zones (NFZs) oder zeitliche Vorgaben an Wegpunkten. Das System gliedert sich in drei Hauptkomponenten:

- Die KIEZ Airborne Komponente: Befindet sich im Luftfahrzeug und ist zuständig für die Verwaltung des TPN, die Interaktion mit dem Piloten (Remote Pilot – RP) bezüglich Änderungen und Freigaben, die Kommunikation mit dem TPN Solution Finder und das Management verschiedener Automatisierungslevel, insbesondere auch in Szenarien mit Verbindungsverlust (C2LL).
- Die KIEZ Ground Komponente: Befindet sich in der Bodenkontrollstation (GCS) und ermöglicht dem RP die Überwachung und Steuerung des KIEZ Airborne Systems. Sie präsentiert Anfragen zur Freigabe, übermittelt Entscheidungen an die Airborne Komponente und verfügt über einen eigenen Solution Finder zur Unterstützung bei Verbindungsverlust.
- Der TPN Solution Finder: Ebenfalls auf der Flugzeugseite angesiedelt, löst dieser das TPN mithilfe eines modifizierten A*-Algorithmus und einer Heuristik. Ein Konsistenzprüfer stellt sicher, dass temporale und POI-bezogene (Point of Interest) Randbedingungen erfüllt sind, und initiiert bei Bedarf ein rekursives Backtracking.

Die funktionale Architektur beschreibt das Zusammenspiel dieser Komponenten, wobei Blöcke wie "Process Changes to TPN", "Consolidate Automation Level", "Provide Temporal Planning Network" und "Decide on Optimal Route Execution" im Mittelpunkt der Softwareimplementierung stehen. Diese Blöcke handhaben die Aktualisierung des TPN basierend auf externen Systemeingaben (z.B. Wetterradar) und RP-Anfragen, passen den Automatisierungsgrad an (auch bei Verbindungsverlust) und entscheiden über die Ausführung oder Freigabe der optimalen Route.

Softwareseitig steht der TPN Solution Finder mit seinem Suchalgorithmus im Fokus. Die Softwarearchitektur ist als UML-Klassendiagramm dargestellt, das die Beziehungen zwischen Kernklassen wie dem TPN Solution Finder, Graphknoten, Graphkanten und temporalen Randbedingungen aufzeigt.

1.5.2.2 Software-Lebenszyklus und Zertifizierungsaktivitäten

Für die Entwicklung wird ein Software-Lebenszyklusprozess vorgeschlagen, der sich am generischen Prozess von Airbus Defence and Space (A1) orientiert. Dieser traditionelle, sequenzielle Ansatz eignet sich für sicherheitskritische Anwendungen, kann aber auch für inkrementelle Funktionslieferungen angepasst werden. Agile Entwicklungsmethoden sind ebenfalls denkbar, müssten aber im PSAC und im Softwareentwicklungsplan detailliert beschrieben werden. Der Prozess umfasst typische Phasen wie Software-Anforderungsanalyse, Design, Kodierung und Integration, jeweils mit Verifikationsschritten und Übergangskriterien.

Die für die Zertifizierung geplanten Software-Lebenszyklusdaten umfassen eine Reihe von Dokumenten, die gemäß DO-178C, DO-333 und DO-332 erstellt werden müssten. Dazu gehören unter anderem der Plan für Softwareaspekte der Zertifizierung (PSAC), der Softwareentwicklungsplan (SDP), der Softwareverifikationsplan (SVP), der Softwarekonfigurationsmanagementplan (SCMP), der Softwarequalitätssicherungsplan (SQAP), Anforderungs- und Designdokumente (SRD, SDD), Verifikationsberichte und der Software Accomplishment Summary (SAS).

Die Software-Level-Klassifizierung geht basierend auf einer vorläufigen Sicherheitsbewertung von DAL C aus, da KIEZ-TPN SW-Komponenten potenziell zu "Major Failure Cases" beitragen könnten. Für eine umfassende Analyse wurden jedoch die Ziele bis DAL A berücksichtigt.

Die Software-Zertifizierungsaktivitäten folgen den Vorgaben des DO-178C. Dies beinhaltet:

- Softwareplanung: Definition der Prozesse, des Lebenszyklus und des Kritikalitätslevels.
- Softwareentwicklung: Erstellung von High-Level- und Low-Level-Anforderungen, Design der Architektur und Implementierung des Codes.
- Softwareverifikation: Durchführung von Unit-, Integrations- und Systemtests zur Sicherstellung der Anforderungserfüllung.
- Konfigurationsmanagement: Verwaltung von Software-Konfigurationselementen und Änderungen.
- Softwarequalitätssicherung: Überwachung und Auditierung der Entwicklungs- und Verifikationsaktivitäten.
- Rückverfolgbarkeit: Sicherstellung der Nachvollziehbarkeit zwischen Anforderungen, Design und Code.
- Werkzeugqualifizierung: Bewertung und Qualifizierung der eingesetzten Entwicklungs- und Verifikationswerkzeuge.
- Reviews und Audits: Regelmäßige Überprüfungen zur Einhaltung des DO-178C.
- Software Configuration Index (SCI) und Software Accomplishment Summary (SAS): Dokumentation des Konfigurationsstatus und Zusammenfassung der Zertifizierungsaktivitäten.
- Zertifizierungsdokumentation: Erstellung und Einreichung der Nachweisdokumentation.

1.5.2.3 Zusätzliche Überlegungen

Besondere Aufmerksamkeit gilt spezifischen technologischen Aspekten und deren Implikationen für die Zertifizierung:

Formale Methoden (DO-333): Es ist geplant, formale Methoden als Nachweismittel für die TPN Solution Finder Implementierung einzusetzen. Dies betrifft insbesondere die Verifikation der Ausgaben der Softwarekodierungs- und Integrationsprozesse (DO-333 A-5).

- SMT-basierte Verifikation: Unter Verwendung des Z3 SMT Solvers sollen Low-Level-Anforderungen der Kernfunktionen des TPN Solution Finders verifiziert werden. Hierbei werden die Funktionen und Anforderungen als Prädikate und Constraints der Linearen Reellen Arithmetik (LRA) formalisiert.
- Theorembeweis: Mit dem Program Verification System (PVS) sollen mathematische Beweise für Eigenschaften der Planungsalgorithmen erbracht werden, z.B. für Korrektheit der Heuristiken oder Behandlung von NFZs. PVS-Bibliotheken können diesen Prozess unterstützen.

Software-Werkzeuge und Werkzeugqualifizierung (DO-330): Die eingesetzten Werkzeuge, insbesondere Z3 und PVS, müssen qualifiziert werden. Basierend auf den Kriterien des DO-330 (Criteria 1, 2 oder 3) und dem Software-Level wird ein Tool Qualification Level (TQL) von TQL-1 bis TQL-5 bestimmt. Z3 und PVS fallen voraussichtlich unter TQL-4 oder TQL-5. Für TQL-5 Werkzeuge müssen Tool Operational Requirements (TOR) definiert und validiert werden, um deren korrekte Funktionsweise im Entwicklungsumfeld sicherzustellen. Für Z3 wird ein Ansatz ähnlich dem für Kind2 erwogen, der die Korrektheit der Evaluierungsergebnisse analysiert. Für PVS ist eine Mitigationsstrategie für den Fall von Fehlern im Werkzeug (z.B. manuelle Überprüfung der Beweise) notwendig.

Programmiersprache und Kodierungsstandard: Der aktuelle Prototyp ist in Python 3.8 implementiert und orientiert sich am PEP8-Standard. Für ein zukünftiges Produkt ist der Einsatz einer Sprache wie C++ vorgesehen, verbunden mit strengeren Kodierungsstandards (z.B. ANSI oder ISO).

Verwendung von objektorientierten Techniken (DO-332): Der Python-Prototyp nutzt grundlegende OO-Konzepte (Klassen, Methoden, Attribute), jedoch ohne intensiven Einsatz von z.B. Vererbung. Bei einer zukünftigen Implementierung in C++ würden die spezifischen Ziele des DO-332 für objektorientierte Technologien relevant. Es wurden bereits Anstrengungen unternommen, den Prototypcode stärker an OO-Prinzipien und Standards, wie sie aus C++ bekannt sind, auszurichten, z.B. durch Typ-Annotationen und Strukturierung.

1.5.2.4 Anhänge: Geplante Konformität

Der PSAC detailliert die geplante Konformität für jedes der Ziele in den Tabellen A-1 bis A-10 des DO-178C sowie die durch DO-333 (Formale Methoden) und DO-332 (Objektorientierte Technologie) modifizierten bzw. ergänzten Ziele. Für DO-333 liegt der Fokus der Anwendung formaler Methoden auf der Verifikation der Ausgaben der Softwarekodierungs- und Integrationsprozesse für die TPN Solution Finder Komponente. Zusätzliche Ziele im Kontext symbolischer KI, wie die Zulässigkeit, Vollständigkeit und Genauigkeit von Heuristiken und Domänenwissen, werden ebenfalls betrachtet. Für DO-332 wird die Anwendbarkeit geprüft, falls C++ für das zukünftige Softwareprodukt verwendet wird, insbesondere hinsichtlich lokaler Typkonsistenz. Die Nutzung von dynamischem Speichermanagement ist derzeit nicht geplant; statische Speicherallokation wird bevorzugt.

Zusammenfassend legt dieses Arbeitspaket eine solide Grundlage für die zukünftige Zertifizierung eines TPN-basierten Softwareprodukts dar, indem es einen detaillierten Plan und die notwendigen Überlegungen gemäß den einschlägigen Luftfahrtstandards aufzeigt.

1.6 Ergebnisse des HAP 6

1.6.1 Zweck und Umfang

Dieses Dokument zielt darauf ab, die Anwendbarkeit der Ziele des EASA-Konzeptpapiers auf Symbolische KI-Anwendungen zu bewerten. Es bietet zudem Empfehlungen für deren Integration in solche Systeme, entsprechend dem im Dokument „Certified Autonomy for UAS and FCAS“, Ausgabe 1, vom 15.12.2023, definierten Statement of Work.

1.6.2 Überblick über Symbolische KI

Symbolische KI zeichnet sich in Bereichen aus, die präzise Logik und strukturiertes Problemlösen erfordern, wie z. B. mathematisches Theorembeweisen, Expertensysteme und die Verarbeitung natürlicher Sprache, insbesondere in Kontexte, in denen Grammatik- und Syntaxregeln streng definiert sind. Ein wesentlicher Vorteil ist die hohe Interpretierbarkeit, die es Forschern und Ingenieuren ermöglicht, Entscheidungsprozesse einfach zu analysieren und zu debuggen.

Symbolische KI weist jedoch Einschränkungen bei der Verarbeitung von mehrdeutigen, rauschbehafteten oder unstrukturierten Daten auf. Ihre starre Struktur macht sie weniger effektiv für Aufgaben, die Anpassungsfähigkeit oder die Verarbeitung großer Mengen an nicht gekennzeichneten Daten erfordern. Trotz dieser Einschränkungen bleibt Symbolische KI ein unverzichtbarer Bestandteil des breiteren KI-Ökosystems.

Um diese Einschränkungen zu überwinden, gewinnen hybride Ansätze, die Symbolische KI mit Maschinellem Lernen (ML) kombinieren – sogenannte Hybride KI – zunehmend an Bedeutung. Diese Methoden nutzen die Stärken beider Paradigmen und schaffen robuste und vielseitige KI-basierte Systeme.

1.6.3 Anwendbarkeit regulatorischer Rahmenbedingungen

Symbolische KI-Anwendungen werden typischerweise mithilfe traditioneller Algorithmen implementiert. Folglich sind bestehende Vorschriften und Standards, insbesondere RTCA DO-178C, auf diese Systeme anwendbar. Über die allgemeine Implementierungsrichtlinien von DO-178C hinaus wurden die im EASA-Konzeptpapier dargelegten Ziele auf ihre Relevanz für Symbolische KI hin geprüft.

1.6.4 Bewertung der Ziele des EASA-Konzeptpapiers

Die Anwendbarkeit der Ziele des EASA-Konzeptpapiers auf Symbolische KI-Anwendungen lässt sich wie folgt zusammenfassen:

1. Nicht anwendbare Ziele

Kategorien wie Entwicklungsabsicherung (Development Assurance, DA), Datenmanagement (Data Management, DM), Lernmanagement (Learning Management, LM) und Implementierung von ML-Modellen (ML Model Implementation, IMP) sind spezifisch auf Maschinelles Lernen zugeschnitten und daher nicht relevant für Symbolische KI.

2. Anwendbare Ziele

Ziele in den Kategorien Konzept (Concept, CO), Klassifikation (Classification, CL), Sicherheitsbewertung (Safety Assessment, SA), Ethik und Vertrauenswürdigkeit (Ethics and Trustworthiness, ET), Konfigurationsmanagement (Configuration Management, CM), Qualitätsmanagement (Quality Assurance, QA), Menschliche Faktoren (Human Factors, HF), Sicherheitsrisikominderung (Safety Risk Mitigation, SRM) und Organisation (Organization, ORG) sind auf Symbolische KI anwendbar. Diese Kategorien setzen nicht den Einsatz von Maschinellem Lernen voraus.

Während einige Ziele dieser Kategorien mit bestehenden Standards wie DO-178C (z. B. CO, CM, SA, QA, SRM) übereinstimmen, führen andere neue Anforderungen ein, die für KI-Anwendungen im Allgemeinen relevant sind. Kategorien wie CL, ET und HF bieten einzigartige Beiträge, die in den Zertifizierungsprozess für Symbolische KI-Anwendungen einfließen sollten.

1.6.5 Besondere Überlegungen zu Expertensystemen

Die Hauptspezifik für Expertensysteme innerhalb der Symbolischen KI liegt in der Validierung von Anforderungen, die aus Heuristiken abgeleitet wurden. Eine Heuristik ist eine domänenspezifische Regel, die in der Regel in natürlicher Sprache ausgedrückt wird und wie eine Standardanforderung behandelt werden kann. Abhängig von ihrer Komplexität können zusätzliche abgeleitete Anforderungen für die Implementierung und Validierung erforderlich sein.

1.6.6 Fazit

Die Validierung von Heuristiken stellt einen spezifischen Aspekt von Symbolischen KI-Anwendungen dar, insbesondere bei Implementierungen als Expertensysteme. Die Anerkennung und Berücksichtigung dieser Spezifik ist entscheidend für die effektive Zertifizierung solcher Systeme. Die Ziele des EASA-Konzeptpapiers, insbesondere jene aus den Kategorien CL, ET und HF, sollten in den Zertifizierungsprozess integriert werden, um eine umfassende Compliance zu gewährleisten.

2 Projektverlauf

2.1 Projektmanagement (HAP 0)

Fortschritt: Das Projekt begann mit einer Verzögerung von etwa 5 Monaten und einem gestaffelten Start der Partner zwischen August und Oktober 2020. Das Kick-off-Meeting fand im Oktober 2020 statt. Die Unterbeauftragung der EASA gestaltete sich anfangs schwierig und führte zu Verzögerungen. Es wurden Projektverlängerungen gewährt, zunächst bis Ende 2023 und dann eine kostenneutrale Verlängerung bis zum 30.09.2024. Die Zusammenarbeit mit der EASA wurde etabliert, obwohl Ressourcenengpässe bei der EASA den Umfang ihrer Beteiligung beeinflussten.

Status am Projektende (30.09.2024): Das Projektmanagement hat die anfänglichen Verzögerungen und Herausforderungen bei der EASA-Koordination erfolgreich gemeistert und notwendige Verlängerungen gesichert. Die letzte Phase im Jahr 2024 (abgedeckt durch WP6) schloss das Projekt ab. Das Projekt wurde trotz der Herausforderungen adaptiv gesteuert.

2.2 AP 1.1: Synthese des bestehenden Know-Hows für die Zertifizierung

Fortschritt: Die Hauptarbeiten fanden von August bis Dezember 2020 statt. Die finalen Dokumente waren im Entwurfsstadium und sollten Anfang 2021 finalisiert werden. Der Meilenstein MS1 (Ergebnisse von AP1.1) wurde wegen des verspäteten Projektbeginns auf den 31. Januar 2021 verschoben. Airbus Defence and Space (ADS) schloss seine Dokumente Ende Dezember 2020 ab und versandte sie zum internen Review. Dieses AP fasste die bestehenden Zertifizierungsrahmenbedingungen zusammen.

Status am Projektende: Abgeschlossen im Januar 2021. Die Ergebnisse bildeten die Grundlage für AP 1.2 und die weiteren Arbeiten zur Zertifizierung.

2.3 AP 1.2: Erstellung eines Zertifizierungskonzepts

Fortschritt: Aufbauend auf AP 1.1 wurde in diesem AP ein Zertifizierungskonzept entwickelt. Es wurde festgestellt, dass bestehende Zertifizierungsanforderungen zwar Innovationen ermöglichen, die Zertifizierung echter KI jedoch eine große Herausforderung darstellt und ein wirklicher Durchbruch in naher Zukunft nicht zu erwarten ist. Der Fokus wurde auf modellbasierte Ansätze und formale Methoden gelegt. Das Hauptarbeitspaket 1 (HAP 1, umfassend AP 1.1 und AP 1.2) wurde bis Sommer 2021 abgeschlossen.

Status am Projektende: Abgeschlossen in 2021 als Teil von HAP 1. Die Erkenntnisse flossen maßgeblich in die APs 2 und 4 ein.

2.4 AP 2.1: Erarbeitung eines Anwendungskatalogs und Analyse existierender KI-Technologien

Fortschritt: ADS erarbeitete im Rahmen von HAP 2 einen Anwendungskatalog. Dieser wurde mit Experten abgestimmt. Es wurden existierende KI-Technologien aufgelistet und deren Anwendbarkeit auf die Punkte des Katalogs erörtert. Es zeigte sich, dass der größte Nutzen bei weniger neuartigen Technologien zu erwarten ist. Dieses AP wurde 2021 abgeschlossen, da HAP 2 in 2021 abgeschlossen wurde.

Status am Projektende: Abgeschlossen in 2021. Lieferte eine Übersicht über potenzielle KI-Anwendungen und Technologien als Basis für die Demonstratorentwicklung.

2.5 AP 2.2: Entwicklung eines Pfadplanungs-Systems, Demonstrationsframeworks und CONOPS

Fortschritt: Die Entwicklung des TPN-basierten Pfadplanungssystems zur Umfliegung von Unwetter- und No-Fly-Zones begann 2021. Als Datenstrukturgrundlage dienten Temporal Planning Networks (TPNs). Ein A*-Algorithmus mit Consistency Checker und rekursivem Backtracking wurde implementiert. Als Demonstrationsumgebung wurde das MDE (MUM-T Development Environment) gewählt, was Zeit sparte. AP 2.2 wurde Anfang 2022 abgeschlossen. Ein funktionierender Demonstrator wurde bis Ende 2022 validiert. Im Rahmen dieses Arbeitspakets und der späteren TRL4-Aufgaben wurde auch das Concept of Operations (CONOPS) entwickelt. Das CONOPS wurde geschrieben, um Benutzerbedürfnisse zu bewerten und Fähigkeiten sowie Anforderungen zu generieren; hierzu wurden Pilotenинterviews durchgeführt.

Status am Projektende: Das AP 2.2 selbst wurde Anfang 2022 abgeschlossen, der Demonstrator Ende 2022 validiert. Das TPN-System und sein Demonstrator waren zentrale Beiträge von ADS. Das CONOPS, das im Rahmen der TRL4-Bestrebungen (welche großteils AP3.3 zugeordnet sind) in 2023 finalisiert wurde, lieferte eine wichtige Grundlage für die Systemdefinition und trug maßgeblich zur Erreichung von TRL4 bei.

2.6 AP 3.2: Entwicklung und Integration der KI-Komponente, Systemanforderungen, Funktionale Architektur und Sicherheitsbewertung

Fortschritt: Dieses AP umfasste die Entwicklung und Integration der KI-Komponente (TPN-Planer) in das MDE-Framework. Es wurde Mitte 2022 abgeschlossen. Ein funktionierender Demonstrator lag Ende 2022 vor. Das technische Beschreibungsdokument (TDD) für TPN war ein wichtiger Liefergegenstand aus diesem AP. Im Rahmen der Ausreifung der TPN-Technologie und als Beitrag zu TRL4 wurden in diesem AP auch das Systemanforderungsdokument (SRD bzw. TLSRD) erstellt, eine funktionale Architektur generiert und eine vorläufige Sicherheitsbewertung durchgeführt. Diese Dokumente wurden großteils 2022 begonnen und 2023 im TRL4-Kontext finalisiert.

Status am Projektende: Das AP wurde Mitte 2022 bezüglich der KI-Komponente und des TDD abgeschlossen. Die Arbeiten an SRD, funktionaler Architektur und Sicherheitsbewertung wurden im Kontext von TRL4 in 2023 finalisiert und trugen maßgeblich zur erfolgreichen TRL4-Bewertung bei.

2.7 AP 3.3: Integration des TPN-Systems in FCS/FMS und Durchführung der Demonstrationskampagne

Fortschritt: Aufbauend auf dem CONOPS (aus AP 2.2) und den Systemdefinitionen aus AP3.2 lag der Schwerpunkt dieses APs auf der Integration des TPN-Systems in Flugführungssysteme (FCS) und Flight Management Systeme (FMS). Ein wesentliches Ziel war die Durchführung einer Demonstrationskampagne zur Validierung der Systemanforderungen. Diese Demonstrationskampagne wurde im Jahr 2023 erfolgreich durchgeführt, wobei Tests zeigten, dass das System wie erwartet funktionierte. Die Ergebnisse trugen maßgeblich zur Erreichung von TRL 4 für die TPN-Technologie bei. Der Großteil der übrigen TRL4-Dokumente und -Aktivitäten, die nicht AP3.2 zugeordnet wurden, wurde unter AP3.3 geliefert.

Status am Projektende: Weitgehend abgeschlossen in 2023 mit dem Erreichen von TRL 4 für die TPN-Technologie und einer erfolgreichen Demonstrationskampagne.

2.8 AP 3.4: Integration der drei Demonstratoren der Partner

Fortschritt: Begann 2022, wobei der Großteil der Arbeiten für 2023 geplant war. Ziel war es, die Interoperabilität der von den verschiedenen Partnern entwickelten Demonstratoren zu untersuchen und zu definieren. Die inhaltlichen Arbeiten waren Ende 2023 zu ca. 90% abgeschlossen.

Status am Projektende: Die inhaltliche Arbeit war Ende 2023 weitgehend abgeschlossen. Allerdings konnte das finale zusammenfassende Berichtsdokument für AP 3.4 aufgrund von Ressourcenengpässen nicht erstellt werden. Dies war der einzige nicht abgeschlossene Liefergegenstand des Projekts zum Ende des Berichtszeitraums 2023.

2.9 AP 4.2/4.3/4.4: Zertifizierungsaspekte und -methoden (HAP 4)

Fortschritt: Vorbereitende Arbeiten begannen 2022.

AP 4.2/4.3 (EASA Concept Paper Methods): Es wurde zunächst untersucht, inwieweit das EASA-Konzeptpapier zur KI-Zertifizierung für TPN (symbolische KI) anwendbar ist. Da das Papier primär auf Machine Learning (ML) fokussiert war, wurden diese Arbeiten als vorläufig bewertet und später eingestellt und archiviert, als der Fokus auf traditionelle Methoden verlagert wurde und EASA eigene Pläne für symbolische KI für 2024/2025 ankündigte.

AP 4.4 (Traditionelle Zertifizierungsmethoden): Im Jahr 2023 wurde ein "KIEZ-TPN SW-Kit" entwickelt. Dieses Kit, strukturiert als Plan für Softwareaspekte der Zertifizierung (PSAC), illustriert die potenzielle Zertifizierung des TPN-basierten Softwareprodukts mittels etablierter Standards wie DO-178C und dessen Ergänzungen (DO-330, DO-332, DO-333). Der Ansatz wurde der EASA vorgestellt und grundsätzlich positiv bewertet.

Status am Projektende: AP 4.2/4.3 wurden archiviert. AP 4.4 lieferte erfolgreich das "KIEZ-TPN SW-Kit" für traditionelle Zertifizierungswege mit positiver Resonanz von der EASA Ende 2023. Die ursprünglich für 2024 geplante Weiterführung der EASA-Abstimmung zu symbolischer KI wurde in WP6 überführt.

2.10 AP 5: Querschnittsthemen

Die Arbeitspakete 5.1 und 5.2 (AP5.1 und AP5.2) wurden von anderen Partnern geleitet, und Airbus leistete hierzu keinen Beitrag. Informationen zu diesen Arbeitspaketen werden in deren Dokumentation dargestellt.

Arbeitspaket 5.3 (AP5.3) wurde von der Fortiss GmbH zum Thema Humanfaktoren geleitet. Airbus trug zu diesem Arbeitspaket bei, indem die Entwicklung eines Humanfaktoren-Demonstrators unterstützt wurde. Diese Arbeit wird von der Fortiss GmbH in ihrer Dokumentation dargestellt.

Arbeitspaket 5.4 (AP5.4) wurde aufgrund eines Vertragsfehlers in Abstimmung mit dem Kunden gestrichen.

2.11 AP 6: Arbeiten im Rahmen der Projektverlängerung 2024

Fortschritt: Dieses Arbeitspaket deckte die kostenneutrale Projektverlängerung von Januar bis September 2024 ab. Die Verlängerung war notwendig, da die EASA-Konzeptpapiere zunächst nur ML-Anwendungen und nicht symbolische KI-Anwendungen (wie TPN) abdeckten. Airbus setzte seine Vertretung in der EASA WG-114 fort, die die Arbeiten an den Konzeptpapieren koordiniert. Die Erwartung war, dass sich die Diskussionen 2024 auf symbolische KI konzentrieren würden, und Airbus wollte diese Präsenz nutzen, um Informationen zu sammeln und spezifische Beiträge für den TPN-Anwendungsfall zu liefern. Ursprünglich waren für 2024 zwei Leistungen geplant, die die in

dieser Erweiterung geleisteten Arbeiten abdecken sollten; diese wurden zu einer kombinierten Leistung zusammengefasst.

Status am Projektende (30.09.2024): Dieses Arbeitspaket schloss die Arbeiten der Projektverlängerung 2024 ab. Die geplante kombinierte Leistung wurde termingerecht geliefert. WP6 führte die wichtige Schnittstellenarbeit mit der EASA zu den sich entwickelnden Richtlinien für die Zertifizierung symbolischer KI fort und brachte die Projekterkenntnisse ein.

3 Verwertung der Ergebnisse / Fortschreibung des Verwertungsplans

Die beiden Hauptprodukte der von Airbus durchgeführten Arbeiten sind:

1. Der Pfadplanungsalgorithmus (entwickelt gemäß den Zertifizierungsgrundsätzen aus Punkt 2)
2. Das mit der EASA abgestimmte Zertifizierungskonzept und -verfahren, um eine Zertifizierung des Algorithmus aus Punkt 1 zu ermöglichen

Der Algorithmus zeichnet sich durch seine Vorhersagbarkeit aus, d.h. die vom System gewählte Route stammt stets aus einer größeren Menge, die der Operator definiert. Dies bietet Vorteile sowohl für zivile als auch für militärische Anwendungsfälle:

> **Militärisch**: In einem militärischen Szenario kann der Operator zahlreiche sichere Routen für das Luftfahrzeug definieren. Dies bedeutet, dass das Luftfahrzeug in einem C2LL-Szenario (Command and Control Link Loss) eine höhere Wahrscheinlichkeit hat, sicher zu bleiben. Die Zertifizierung im militärischen Friedensbetrieb ist der zivilen Zertifizierung ähnlich.
>
> **Zivil**: In einem zivilen Szenario hilft die Vorhersagbarkeit des Algorithmus dem Piloten und der Flugsicherung (ATC) zu verstehen, wohin sich das Luftfahrzeug in einem C2LL-Szenario bewegen könnte. Dies ist für die Luftraumintegration von großer Bedeutung. Auch hier wird die Zertifizierung des Algorithmus in beiden Fällen ähnlich sein.

Als Nachfolge zu diesem Projekt stehen, da die Bemühungen zur Integration unbemannter Luftfahrzeuge in den zivilen Luftraum wesentlich langsamer als im militärischen Bereich verlaufen, nur sehr wenige zivile Folgeprojektoptionen zur Verfügung. Zudem wird die WG-114 (mit Fokus auf die Zertifizierung von KI) ihre Ergebnisse für symbolische KI erst in vielen Jahren abstimmen und implementieren. Dies bedeutet, dass eine Wiederverwendung der KIEZ-Ergebnisse in einem zivilen Projekt einige Jahre lang nicht möglich sein wird.

Jedoch muss eine Weiterentwicklung der Integration des Algorithmus durchgeführt werden. Viele Integrationsprobleme müssen gelöst und zahlreiche Anwendungsfälle konzipiert und getestet werden. Aus diesem Grund werden die Projektergebnisse im Projekt „Enhanced RPAS Automation 2.1“ (ERA2.1) wiederverwendet und weiterentwickelt. Obwohl dieses Projekt militärisch ausgerichtet ist, sind die im Zusammenhang mit dem Algorithmus durchgeführten Arbeiten nicht spezifisch militärisch. Die Anwendungsfälle der Arbeiten in ERA2.1 sind gleichermaßen nützlich sowohl für zivile als auch für militärische Anwendungen.

3.1 Erfindungen/Schutzrechte die in Anspruch genommen wurden

Es wurden im Berichtszeitraum keine Erfindungen/Schutzrechte in Anspruch genommen. Im Projekt wurden keine Patente angemeldet.

3.2 Wissenschaftliche und technische Erfolgsaussichten nach Projektende

Die nach Projektende verbleibenden technischen Fragen betreffen die Integration des entwickelten Algorithmus in ein luftgestütztes System unter Einbeziehung der Ergebnisse des gemäß den EASA-Vorgaben entwickelten Zertifizierungskonzepts. Das Projekt ERA2.1 zielt darauf ab, diese Fragen (unter anderem) zu beantworten, insbesondere:

- Wie kann der Autonomiegrad von fernferngesteuerten Luftfahrzeugsystemen (Remotely Piloted Aircraft Systems, RPAS) signifikant erhöht werden, um den Einsatz von Fernpiloten zu reduzieren und neue Betriebskonzepte zu ermöglichen?
    - Das Projekt strebt den Übergang „Von der Automatisierung zur Autonomie“ an, wodurch Anwendungen wie Manned-Unmanned Teaming (MUT) und Flottenbetrieb

mit reduzierter Besatzung ermöglicht werden, die bisher als unwirtschaftlich galten. Dies beinhaltet die Bewertung aktueller und sich in Entwicklung befindlicher Technologien, mit besonderem Fokus auf „modellbasierte Autonomie“ (“Model Based Autonomy”) für autonomes Notfallmanagement (Contingency/Emergency Management) und „Multi-Agenten-Systeme“ (“Multi-Agent Systems”) für die Koordination und Kollaboration von unbemannten Luftfahrzeugsystemen (UAS).
    - Um den Autonomiegrad zu erhöhen, ist ein vorhersagbarer Routenwahlalgorithmus erforderlich, weshalb die Ergebnisse aus KIEZ in das Projekt einfließen werden.
- Was ist die erforderliche Zertifizierungsgrundlage und was sind die akzeptablen Nachweisverfahren (Acceptable Means of Compliance, AMCs), um autonome Funktionen in unbemannten Luftfahrzeugen zu ermöglichen, insbesondere solche, die Künstliche Intelligenz (KI) einbeziehen?
    - Ein Kernziel von ERA2.1 ist die Entwicklung der Zertifizierungsgrundlage gemeinsam mit den Behörden, der Nachweis der Sicherheit (Safety Case) sowie die Erstellung von akzeptablen Nachweisverfahren (AMCs) und Leitmaterial (Guidance Material, GM) für autonome Funktionen. Dies beinhaltet die Bewältigung der einzigartigen Herausforderungen bei der Qualifizierung und Zertifizierung von KI-basierten autonomen Flugsystemen, was derzeit weltweit ein ungelöstes Problem darstellt.
    - Die im Rahmen von KIEZ im Bereich der Zertifizierung durchgeführten Arbeiten werden direkt in die Arbeitspakete zur Zertifizierung und Lufttüchtigkeit einfließen, insbesondere in Bezug auf akzeptable Nachweisverfahren (Acceptable Means of Compliance).