

Experimentelle Validierung eines Georeferenzierungssystem für ein helikoptergestütztes Bodenradarsystem

Bachelor Thesis

Author(s):

Pohl, Benedikt Raphael; Rischatsch, Fadri

Publication date:

2023-06-15

Permanent link:

<https://doi.org/10.3929/ethz-b-000617068>

Rights / license:

[In Copyright - Non-Commercial Use Permitted](#)

Benedikt Pohl, Fadri Rischatsch

**Experimentelle Validierung eines
Georeferenzierungssystem für ein
helikoptergestütztes
Bodenradarsystem**

Bachelorarbeit

Institut für Geodäsie und Photogrammetrie (IGP)
Eidgenössische Technische Hochschule Zürich

Betreuung

Prof. Dr. Benedikt Soja
Dr. Gregor Möller, Raphael Moser, Dr. Andreas Bauder

15. Juni 2023

Abstract

Diese Bachelorarbeit befasst sich mit der Weiterentwicklung des an der ETH Zürich entwickelten Airborne Ice Radar Systems (AIR-ETH), einem Messsystem zur genauen Bestimmung der Gletschermächtigkeit. Das AIR-ETH liefert georeferenzierte Daten als Helikopter-gestütztes Bodenradarsystem. Diese Bachelorarbeit baut auf der Arbeit von Lardon (2022) auf. Insbesondere wird die Implementierung des auf Sensorfusion basierenden Georeferenzierungssystems unter Verwendung eines Extended-Kalman-Filters behandelt.

Zwei von Lardon (2022a) identifizierte, kritische Herausforderungen werden in dieser Arbeit behandelt. Erstens wurde ein Problem mit Ausreissern bei bestimmten Richtungsänderungen untersucht, das bisher zu wiederholten Konvergenzzeiten für die Navigationslösung geführt hat. Zweitens wurde die ausschliessliche Verwendung von Epochen mit vier Antennen überarbeitet, da theoretisch nur drei Beobachtungen für die Bestimmung der Lage und Position notwendig sind. Ausserdem wurde ein neuer, von VAW entwickelter, gewichtsoptimierter GFK-Rahmen experimentell validiert.

Durch eine Modifikation der Behandlung von Richtungsänderungen in der Innovationsmatrix des Kalman-Filters konnte das Konvergenzzeitproblem erfolgreich gelöst werden, was zu einer verbesserten Navigationslösung insbesondere in bestimmten Flugrichtungen (Nord/Süd) führt. Durch die Berechnung der Position mit nur drei gleichzeitigen GNSS-Beobachtungen konnte die Anzahl der verfügbaren Epochen erhöht und damit die Verfügbarkeit des Systems verbessert werden. Auch die Benutzerfreundlichkeit der Software wurde durch die Einführung eines Kontrollfensters und die Erstellung einer umfassenden Dokumentation verbessert.

Zusammenfassend stellt diese Arbeit einen wichtigen Schritt zur Optimierung der Leistungsfähigkeit von AIR-ETH dar und bereitet den Weg für effektivere und genauere Gletscherdickenmessungen.

Danksagung

An dieser Stelle möchten wir uns bei allen Personen bedanken, die uns während unserer Bachelorarbeit unterstützt haben.

An erster Stelle bedanken wir uns bei **Prof. Dr. Benedikt Soja** und **Dr. Gregor Möller** für die Betreuung und Unterstützung der Arbeit sowie für das entgegengebrachte Vertrauen und die dadurch ermöglichte selbständige Arbeitsweise.

Des Weiteren bedanken wir uns bei den Beteiligten der VAW. Namentlich sind dies **Dr. Andreas Bauder**, **Raphael Moser** und **Barthélémy Anhorn**. Sie haben uns bei der Durchführung der Experimente unterstützt, uns in das System eingeführt und sind uns mit Rat und Tat zur Seite gestanden. Wir danken auch **Alexander Wolf** für seine Unterstützung in technischen Fragen und sein Engagement bei der Suche nach Lösungen.

Wir möchten uns ausserdem herzlich bei allen Personen bedanken, die uns bei der Überarbeitung dieser Arbeit unterstützt haben. Insbesondere möchten wir **Anna Minor**, **Andreas Bernhard**, **Dr. phil. Deborah Bernhard**, **Marisa Rischatsch** und **Johanna Pfäffli** nennen. Ihre wertvollen Beiträge zum Text haben es uns ermöglicht, das hier vorgestellte Endergebnis zu erzielen.

Zuletzt, aber keineswegs weniger bedeutsam, möchten wir uns bei **Fabrice Lardon** bedanken. Seine hilfreichen Ratschläge, Unterstützung und die Bereitschaft, uns mit zusätzlichen Informationen zu versorgen, haben wesentlich zu unserer Arbeit beigetragen.

Inhaltsverzeichnis

Abstract	i
Danksagung	iii
Abbildungsverzeichnis	ix
Tabellenverzeichnis	x
Abkürzungsverzeichnis	xi
1 Einleitung	1
1.1 Zielsetzung	1
1.2 Weitere Problemstellungen	2
2 Theorie	3
2.1 Kalman-Filter	3
2.1.1 Einführung	3
2.1.2 Bestandteile	4
2.2 Koordinatensysteme	9
3 Methodik	11
3.1 Bodyframe Vermessung	11
3.1.1 Setup und Durchführung	11
3.1.2 Berechnung und Transformation der Daten	12
3.1.3 Integration von Plausibilitätsprüfungen in die Datenverarbeitung	13
3.2 Experimentelle Validierung des Systems	13
3.2.1 Dynamik-Test: Kran Experiment	13

3.2.2	Analyse von Hardware-Problemen mit ublox ANN-MB01 und ZED-F9P	14
3.2.3	Anwendungstest: Helikopterflug Wallis	15
3.2.4	Analyse nach Anwendungstest	15
	Plausibilitätschecks von Gyro-Raten, Alignment und Beschleunigungen	16
3.2.5	Behandlung des IMU Offsets	16
	IMU-Test: Kran Experiment	16
3.3	Softwareentwicklung und DevOps	17
3.3.1	Behebung von Azimutwinkel-Orientierungsfehlern im Kalman-Filter	18
3.3.2	Erhöhung der Systemverfügbarkeit durch Redundanz in GNSS-Antennennutzung	18
	Tests der Implementierung	19
3.3.3	Implementierung der rückwärtslaufenden Verarbeitung	19
3.3.4	Implementierung der Korrektur von IMU-Offset	20
3.3.5	Überarbeitung der Transformation ins LV95	21
3.3.6	Kontrollfenster für Verarbeitung	21
3.3.7	Implementierung von DevOps-Prinzipien	22
	Etablierung von Git für zentrale Versionskontrolle und Zusammenarbeit	23
	Dokumentation des Prozesses und der Anwendung	23
4	Ergebnisse	24
4.1	Ergebnisse der Bodyframe-Vermessung	24
4.2	Ergebnisse der experimentellen Validierung des Systems	25
4.2.1	Ergebnisse des Dynamik-Tests: Kran Flugexperiment	25
4.2.2	Erkenntnisse aus der Analyse von Hardware-Problemen mit ublox ANN-MB01 und ZED-F9	25
4.2.3	Ergebnisse des Anwendungstests: Helikopterflug Wallis	26
4.2.4	Analyse nach Anwendungstest	27
	Analyse des SNR und der Multipath-Effekte	27
	RTKPOST Processing	29
	Plausibilität von Gyro-Raten, Alignment und Beschleunigungen	30
4.2.5	Effekte der IMU-Umplatzerung im IMU-Test	32
4.3	Ergebnisse der Softwareentwicklung & DevOps	34

4.3.1	Wirksamkeit der Behebung von Azimutwinkel- Orientierungsfehlern im Kalman-Filter	34
4.3.2	Effektivität der erhöhten Systemverfügbarkeit durch Redun- danz in GNSS-Antennennutzung	35
4.3.3	Effekte der Beschleunigungskorrektur	37
4.3.4	Effekte der Überarbeitung der Transformation ins LV95	37
4.3.5	Entwicklung des <i>AIR-ETH Control Window</i>	37
4.3.6	Verwendung von Git für zentrale Versionskontrolle und Doku- mentation	38
5	Diskussion	39
5.1	Experimentelle Validierung des Systems	39
5.1.1	Wichtigkeit der GNSS-Daten	41
5.1.2	IMU-Drift	41
5.1.3	IMU-Alignment	41
5.2	Softwareentwicklung	42
5.2.1	Wirkung der Änderung im Kalman-Filter	42
5.2.2	Redundanz in GNSS-Antennennutzung	43
5.2.3	Verbesserungen in Software und DevOps	44
6	Fazit und Ausblick	45
6.1	Fazit	45
6.2	Ausblick	46
	Literaturverzeichnis	49
A	Arbeitsaufteilung	I
B	Weitere Formeln und Plots	II
B.1	Weitere Formeln	II
B.2	Weitere Plots	III
C	Dokumentation	IV
C.1	Home	IV
C.1.1	Prerequisites	IV
C.1.2	Installation	IV
C.1.3	Usage	V
C.2	Bodyframe Determination	V

C.2.1	Input	V
C.2.2	Processing	V
C.2.3	Additional information	VI
C.2.4	Bodyframe IMU origin offset	VI
C.3	Material list	VI
C.4	Measuring the baselines	VII
C.4.1	IMU Placement	XVIII
C.5	Hardware Bugs	XX
C.5.1	Bugs with the Loggers of the AIR-ETH Project	XX
C.6	Processing Procedure	XX
C.6.1	Pre-requisites	XXI
	Software	XXI
	Files	XXI
	GNSS-related files	XXI
	Other Files	XXI
C.6.2	Processing Procedure	XXII
C.6.3	Converting uBlox Files to .obs Files	XXIV
C.6.4	Baseline Processing using RTKPOST	XXV
C.7	Data Processing with geoAIR-ETH	XXVII
C.7.1	Using AIR-ETH Processing Control Window	XXVII
C.7.2	Using AIR-ETH Processing Procedure	XXVII
C.7.3	Troubleshooting	XXXII
C.8	Software Documentation	XXXII
C.9	AIR-ETH Software Documentation	XXXII
C.9.1	Structure	XXXIII
C.9.2	Contributions	XXXIII
C.9.3	Overview	XXXIII
C.9.4	Application Class: MyApp	XXXIV
C.9.5	Main Window Class: Window	XXXIV
	GUI Elements	XXXIV
	Functionality	XXXIV
C.9.6	ProcessingParams Class	XXXV

Abbildungsverzeichnis

2.1	Schematische Darstellung des verwendeten Kalman-Filters	4
2.2	Verwendete Koordinatensysteme im AIR-ETH Rahmen. Quelle: (Lardon, 2022a)	10
3.1	Brückenkran (weiss) bei HIA Gebäude mit Materiallager. Bilddaten: CNES, Spot Image, swisstopo, NPOC	17
4.1	Berechnete Position nach der Verarbeitung im Navigationssystem . .	27
4.2	SNR & MP von allen Antennen	28
4.3	RTKPLOT von Flug mit altem Rahmen am 23. Mai 2022	29
4.4	Yaw über die Zeit	32
4.5	Beschleunigung des Rahmens in Navigationsrahmen	33
4.6	Vergleich der Yaw-Winkel vor (blau) und nach (rot) der Anpassung des Wertebereichs der Innovationmatrix	34
4.7	Pitch-Winkel Lösungen aus verschiedenen Antennenkombinationen mit 60° Versatz	35
4.8	Vergrosserter Ausschnitt der Pitch-Winkel Lösungen aus verschiedenen Antennenkombinationen mit 60° Versatz	36
4.9	Fluss-Diagramm des Control Windows	38
4.10	Control Window	38
B.1	Yaw-Winkel Lösungen aus verschiedenen Antennenkombinationen mit 60° Versatz	III
B.2	Roll-Winkel Lösungen aus verschiedenen Antennenkombinationen mit 60° Versatz	III
C.1	Overview	VII
C.2	Workflow for processing	XXI
C.3	Settings_xsens	XXIII

Tabellenverzeichnis

4.1	Maximale Grössen der Winkelgeschwindigkeiten	30
4.2	Standardabweichungen der Beschleunigungen während 15 Minuten stabile Periode	31
4.3	Standardabweichungen der Orientierung während 15 Minuten stabile Periode	31
4.4	Kenngrossen der gemessenen Beschleunigung ohne Gravitation	32

Abkürzungsverzeichnis

AGNES	Automatisches GNSS-Netz Schweiz
AIR-ETH	Airborne Ice Radars of ETH Zurich
CNR	Carrier-to-noise-Verhältnis
DevOps	Software development and IT operations
DGNSS	Differential Global Navigation Satellite System
GFK	Glasfaserverstärkter Kunststoff
GNSS	Globales Navigationssatellitensystem
GPR	<i>Ground-penetrating radar</i> bzw. Bodenradar
IGP	Institut für Geodäsie und Photogrammetrie
IPA	Interdisziplinäre Projektarbeit
IMU	<i>Inertial Measurement Unit</i> bzw. Inertiale Messeinheit
LV95	Landesvermessung 95
MP	<i>Multipath</i> bzw. Mehrwegeeffekt
TS	Totalstation
VAW	Versuchsanstalt für Wasserbau, Hydrologie und Glaziologie
WGS84	World Geodetic System 84

Kapitel 1

Einleitung

Die vorliegende Bachelorarbeit befasst sich mit der Weiterentwicklung des Georeferenzierungssystems des *Airborne Ice Radars of ETH Zurich* (AIR-ETH). Das AIR-ETH ist ein luftgestütztes Bodenradarsystem, zur Bestimmung der Gletscherdicke. Das ursprüngliche Georeferenzierungssystem des AIR-ETH wurde von Fabrice Lardon im Rahmen einer IPA (Lardon, 2022b) und einer anschliessenden Masterarbeit (Lardon, 2022a) entwickelt. Es wurde mit einem geodätischen Referenzsystem (*JAVAD*) verglichen, um Datenqualität, RTKPOST-Verarbeitung und Softwarenutzung zu evaluieren (Lardon, 2022a). Die Software ist derzeit funktionsfähig, weist jedoch Verbesserungspotential auf.

1.1 Zielsetzung

Das Hauptziel dieser Bachelorarbeit ist die Weiterentwicklung der bestehenden Software. Dabei sollen folgende Probleme behoben werden:

- Der Kalman-Filter verliert gelegentlich seine Orientierung bei Übergängen von -180° zu 180° (und umgekehrt) im Azimutwinkel. Eine Anpassung der Verarbeitung ist erforderlich, um eine zuverlässige Orientierung zu gewährleisten.
- Obwohl lediglich drei GNSS-Antennen für Positionierung und Lageschätzung notwendig sind, werden derzeit durchgehend vier Antennen verwendet. Die zusätzliche Antenne dient als Redundanz. Um die Systemverfügbarkeit zu erhöhen, soll die Lageschätzung und Positionsbestimmung mit nur drei verfügbaren Antennen implementiert werden.

Zudem soll ein neuer, gewichtsoptimierter Rahmen von der Versuchsanstalt für Wasserbau, Hydrologie und Glaziologie (VAW) für zukünftige Kampagnen vorbereitet werden, einschliesslich:

- Vermessung der GNSS-Antennen und der IMU mittels Tachymetrie
- Berechnung der Baselines und der Abweichung der IMU zum Schwerpunkt des Rahmens (Offset)
- Unterstützung bei der Systemintegration mit dem neuen Rahmen

1.2 Weitere Problemstellungen

Während der Bearbeitung traten verschiedene Herausforderungen auf. Die Software-Dokumentation war teils unzureichend und verteilt über IPA, Masterarbeit und Dokumentationsdateien. Die Software war nicht zentralisiert und versioniert, was den Zugriff auf unterstützende Code-Teile erschwerte. Eine umfassende Systemänderung führte dazu, dass die Ergebnisse Datenverarbeitung nicht mehr plausibel waren, obwohl die vorherigen Daten des Holzrahmens korrekt verarbeitet wurden. Zudem verursachten Hardware-Probleme beim Austausch von ublox ANN-MB zu ANN-MB01 GNSS-Antennen fehlgeschlagene Experimente.

Um diese Probleme zu bewältigen, wurden folgende Massnahmen umgesetzt:

- Einrichtung eines Git-Repositories zur zentralen Ablage, Versionierung und Kollaboration
- Erstellung einer umfassenden Dokumentation im Git-Wiki für asynchronen Wissenstransfer
- Systemanalyse zur Korrektur der fehlschlagenden Datenverarbeitung
- Hardware-Analyse und Workaround für ublox ANN-MB01 mit ZED-F9P

Durch diese Massnahmen konnte diese Arbeit umgesetzt und die identifizierten Probleme gelöst werden. Die verbesserte Software und die Systemintegration des gewichtsoptimierten Rahmens tragen zur Weiterentwicklung des Georeferenzierungssystems des AIR-ETH bei. Dies ermöglicht eine effizientere und zuverlässigere Nutzung des Systems in zukünftigen Kampagnen und Forschungsprojekten.

Kapitel 2

Theorie

In diesem Kapitel erläutern wir die Theorie hinter der Sensorfusion, welche ein zentrales Element des vorliegenden Georeferenzierungssystems darstellt. Besonders diskutiert wird die aktuelle Weise Implementierung und die Rolle der auftretenden Koordinatensysteme. Vorwissen zum Kalman-Filter, einem Kernaspekt der Sensorfusion, wird vorausgesetzt. Dieses Kapitel dient als Grundlage für weitere Diskussionen.

2.1 Kalman-Filter

Der Kalman-Filter ist ein wesentlicher Bestandteil dieser Arbeit. Seine Funktionsweise und seine wichtigsten Komponenten werden daher im Folgenden theoretisch erläutert.

2.1.1 Einführung

Im bestehenden Georeferenzierungssystem bildet der Kalman-Filter das Bindeglied zwischen den Messungen der IMU und den vier GNSS-Antennen. Der verwendete Extended-Kalman-Filter besteht aus zwei Komponenten, einem Prediction-Step und einem Update-Step. Im Prediction-Step werden die IMU-Daten und ein dynamisches Modell verwendet, um den Systemzustand und die Kovarianzmatrix vorherzusagen. Im anschließenden Update-Step werden diese Vorhersagen durch Einbezug der gemessenen GNSS-Lösungen aktualisiert. Dadurch kann die Übereinstimmung des prädizierten Systemstatus und der Kovarianzmatrix mit den realen Messungen maximiert werden und der mean squared prediction error minimiert werden.

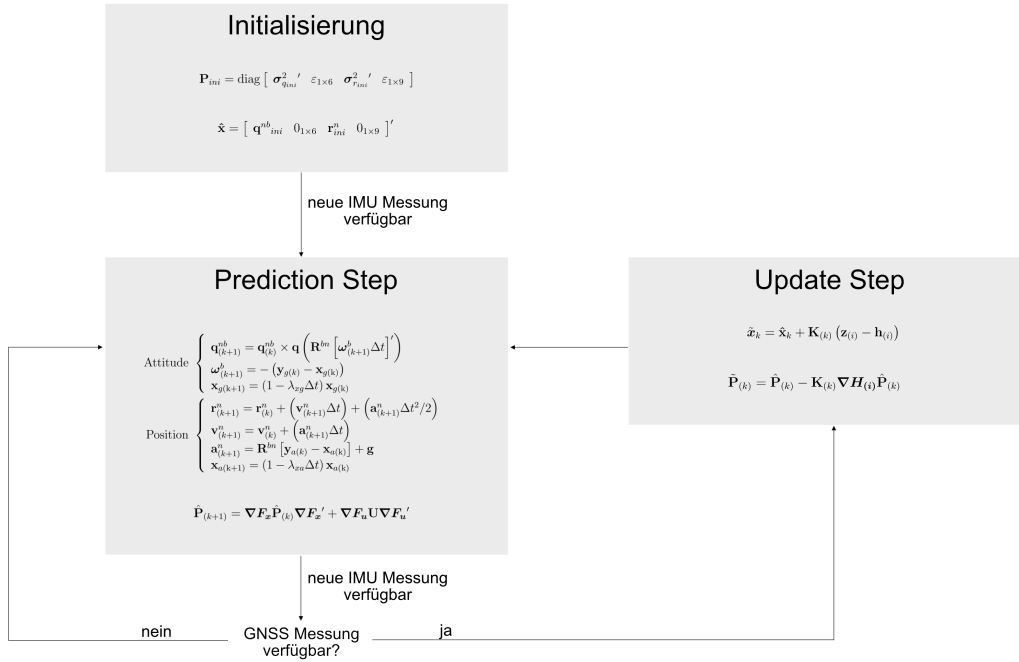


Abbildung 2.1: Schematische Darstellung des verwendeten Kalman-Filters

(Montenbruck and Teunissen, 2017, p 657ff.) Dabei ist zu beachten, dass aufgrund der meist deutlich höheren Abtastrate der IMU auch mehrere Prediction-Steps zwischen zwei Update-Steps liegen können. Der im Code verwendete Kalman-Filter ist ein loosely-coupled Kalman-Filter. Das bedeutet, dass nur die ausgewerteten GNSS-Daten mit den IMU-Daten verarbeitet werden und nicht wie beim Tightly-Coupled Kalman-Filter auch unverarbeitete Messdaten wie z.B. Pseudo-Ranges. (Munguía, 2014)

2.1.2 Bestandteile

Der Systemstatus wird im state Vektor in den Kalman-Filter eingegeben. Dieser kann folgendermassen beschrieben werden:

$$\mathbf{x} = \left[\mathbf{q}^{nb} \quad \boldsymbol{\omega}^b \quad \mathbf{x}_g \quad \mathbf{r}^n \quad \mathbf{v}^n \quad \mathbf{a}^n \quad \mathbf{x}_a \right]' \quad (2.1)$$

Wobei:

\mathbf{q}^{nb} : Einheitsquaternion der Lage (Roll, Pitch, Yaw) [1x4]

$\boldsymbol{\omega}^b$: Biaskompensierte Drehgeschwindigkeiten im Bodyframe [1x3]

\mathbf{x}_g : Bias der Gyroskope [1x3]

\mathbf{r}^n : Position des Ursprungs des Bodyframes im Navigationsrahmen [1x3]

\mathbf{v}^n : Geschwindigkeit im Navigationsrahmen [1x3]

\mathbf{a}^n : Bias kompensierte Beschleunigung im Navigationsrahmen [1x3]

\mathbf{x}_a : Bias der Beschleunigungsmesser [1x3]

Zu Beginn wird der state Vektor mit der ersten verfügbaren GNSS-Beobachtung initialisiert. Dies geschieht während der Aufzeichnung der stable period, in der die Geschwindigkeiten und Beschleunigungen Null sind. Daher werden nur die Quaternion und die Position in den Zustandsvektor geschrieben. Dieser sieht dann wie folgt aus. (Lardon, 2022a)

$$\hat{\mathbf{x}} = \left[\mathbf{q}^{nb}_{ini} \quad 0_{1 \times 6} \quad \mathbf{r}^n_{ini} \quad 0_{1 \times 9} \right]' \quad (2.2)$$

Zu jedem Zeitpunkt k, zu dem Beschleunigungs- und Rotationsmessungen ($\mathbf{y}_a, \mathbf{y}_g$) der IMU zur Verfügung stehen, kann ein weiterer Prediction-Step mit Gleichung 2.3 durchgeführt werden (Munguía, 2014).

$$\begin{array}{l} \text{Attitude} \\ \text{Position} \end{array} \left\{ \begin{array}{l} \mathbf{q}^{nb}_{(k+1)} = \mathbf{q}^{nb}_{(k)} \times \mathbf{q} \left(\mathbf{R}^{bn} \left[\boldsymbol{\omega}^b_{(k+1)} \Delta t \right]' \right) \\ \boldsymbol{\omega}^b_{(k+1)} = -(\mathbf{y}_g^{(k)} - \mathbf{x}_g^{(k)}) \\ \mathbf{x}_g^{(k+1)} = (1 - \lambda_{xg} \Delta t) \mathbf{x}_g^{(k)} \\ \mathbf{r}^n_{(k+1)} = \mathbf{r}^n_{(k)} + \left(\mathbf{v}^n_{(k+1)} \Delta t \right) + \left(\mathbf{a}^n_{(k+1)} \Delta t^2 / 2 \right) \\ \mathbf{v}^n_{(k+1)} = \mathbf{v}^n_{(k)} + \left(\mathbf{a}^n_{(k+1)} \Delta t \right) \\ \mathbf{a}^n_{(k+1)} = \mathbf{R}^{bn} \left[\mathbf{y}_a^{(k)} - \mathbf{x}_a^{(k)} \right] + \mathbf{g} \\ \mathbf{x}_a^{(k+1)} = (1 - \lambda_{xa} \Delta t) \mathbf{x}_a^{(k)} \end{array} \right. \quad (2.3)$$

Mit:

\mathbf{R}^{bn} : Rotationsmatrix vom Bodyframe in den Navigationsrahmen

λ_{xa} : Korrelationszeit Faktor der Beschleunigung

λ_{xg} : Korrelationszeit Faktor der Winkelmessung

Wobei das Quaternion mit

$$\mathbf{q} = \left[\cos\left(\frac{\|\boldsymbol{\omega}\|}{2}\right) \quad \sin\left(\frac{\|\boldsymbol{\omega}\|}{2}\right) \left[\frac{\boldsymbol{\omega}}{\|\boldsymbol{\omega}\|}\right]' \right] \quad (2.4)$$

aus den Drehraten $\boldsymbol{\omega}$ berechnet werden kann (Lardon, 2022a).

Um auch Aussagen über die Genauigkeit der Prädiktion machen zu können, wird zusätzlich die Kovarianzmatrix $\hat{\mathbf{P}}_{(k+1)}$ geschätzt. Diese wird zunächst wie in der Gleichung 2.5 beschrieben initialisiert. Da nur die Quaternion und die Position im initialen state Vektor von null verschieden sind, werden die restlichen Werte, um Verarbeitungsprobleme zu vermeiden, mit sehr kleinen, positiven Werten beschrieben (Lardon, 2022a).

$$\mathbf{P}_{ini} = \text{diag} \left[\boldsymbol{\sigma}_{qini}^2 \quad \varepsilon_{1 \times 6} \quad \boldsymbol{\sigma}_{rini}^2 \quad \varepsilon_{1 \times 9} \right] \quad (2.5)$$

Der darauf folgende Prediction-Step kann nun mit der Gleichung 2.6 erreicht werden (Munguía, 2014).

$$\hat{\mathbf{P}}_{(k+1)} = \nabla \mathbf{F}_x \hat{\mathbf{P}}_{(k)} \nabla \mathbf{F}_x' + \nabla \mathbf{F}_u \mathbf{U} \nabla \mathbf{F}_u' \quad (2.6)$$

Mit:

$\nabla \mathbf{F}_x$: Jakobimatrix des state Vektors

\mathbf{U} : Kovarianzmatrix des Prozessrauschens

$\nabla \mathbf{F}_u$: Jakobimatrix des Prozessrauschens

Die Matrix \mathbf{U} beinhaltet dabei die geschätzten Rauschterme der IMU und hat die Form von Gleichung 2.7.

$$\mathbf{U} = \text{diag} \left[\sigma_g^2 I_{3 \times 3} \quad \sigma_{xg}^2 I_{3 \times 3} \quad \sigma_a^2 I_{3 \times 3} \quad \sigma_{xa}^2 I_{3 \times 3} \right] \quad (2.7)$$

Nach der Durchführung des Prediction-Steps wird, sobald GNSS-Messungen verfügbar sind, ein Update-Step durchgeführt. In diesem Schritt wird der updated state Vektor aus dem predicted state Vektor und der gewichteten Differenz zwischen der aktuellen GNSS-Messung und der vorhergesagten Messung berechnet. Die Berechnung erfolgt wie in Gleichung 2.8 beschrieben.

$$\tilde{\mathbf{x}}_k = \hat{\mathbf{x}}_k + \mathbf{K}_{(k)} (\mathbf{z}_{(i)} - \mathbf{h}_{(i)}) \quad (2.8)$$

Mit:

$\hat{\mathbf{x}}_k$: Predicted state Vektor

$\mathbf{K}_{(k)}$: Kalman-Gain-Matrix

$\mathbf{z}_{(i)}$: Aktuelle GNSS-Messungen

$\mathbf{h}_{(i)}$: prädizierte Messung

Dabei wird die Differenz aus $\mathbf{z}_{(i)}$ und $\mathbf{h}_{(i)}$ auch Innovationmatrix genannt, wobei $\mathbf{z}_{(i)}$ und $\mathbf{h}_{(i)}$ wie folgt aufgebaut sind.

$$\mathbf{z}_{(i)} = \left[z_{yaw(i)} \quad z_{pitch(i)} \quad z_{roll(i)} \quad \mathbf{z}_{r(i)}^n \quad \mathbf{z}_{v(i)}^n \right]' \quad (2.9)$$

$$\mathbf{h}_{(i)} = \left[h_{yaw(i)} \quad h_{pitch(i)} \quad h_{roll(i)} \quad \mathbf{h}_{r(i)}^n \quad \mathbf{h}_{v(i)}^n \right]' \quad (2.10)$$

Die genaue Berechnung von $\mathbf{h}_{(i)}$ kann in Lardon (2022a, p 9) nachgelesen werden. Diese Innovationmatrix wird mit Eulerwinkeln gerechnet (Lardon, 2022a).

Um die in der Gleichung 2.8 benötigte Kalman-Gain-Matrix zu berechnen, werden die predicted Kovarianzmatrix $\hat{\mathbf{P}}_{(k)}$, die Matrix der partiellen Ableitungen des Messungs-Prädiktions-Modells bezüglich $\hat{\mathbf{x}}_k$, $\nabla \mathbf{H}_{(i)}$ und die Kovarianzmatrix des Messrauschens $\mathbf{R}_{(i)}$ benötigt. Diese werden wie in Gleichung 2.11 respektive Gleichung 2.12 beschrieben berechnet (Munguía, 2014).

2.2 Koordinatensysteme

Im folgenden Bericht werden verschiedene Koordinatensysteme verwendet. Der Grund dafür ist einerseits, dass die verschiedenen Systeme die Verarbeitung erheblich erleichtern. Andererseits wurde dies bereits in der vorangehenden Masterarbeit so implementiert. Insgesamt werden in der vorliegenden Arbeit sechs verschiedene Koordinatensysteme verwendet. (Lardon, 2022a) Dies sind:

- **IMU-Bezugsrahmen**

Dieser Rahmen wird von der IMU selbst definiert und hat seinen Ursprung in der IMU. Es handelt sich dabei um ein Linkssystem, was bereits vom Hersteller so vorgegeben wird. (Lardon, 2022a)

- **Vermessungssystem**

Als Vermessungssystem wird das Koordinatensystem bezeichnet, das von der Totalstation bei der Messung erzeugt wird. Die erste Achse zeigt in eine beliebige Richtung, die von der Orientierung und damit von der Aufstellung der Totalstation abhängt. Die zweite Achse wird senkrecht zur ersten Achse in der horizontalen Ebene erzeugt.

- **Bodyframe**

Der Bodyframe ist ein Rechtssystem mit Ursprung im Schwerpunkt des Rahmens. Die x-Achse zeigt in Flugrichtung und die z-Achse nach unten. Alle Positionen der GNSS-Antennen sowie der IMU sind im Bodyframe bekannt. (Lardon, 2022a)

- **Navigationsrahmen**

Der Navigationsrahmen hat seinen Ursprung im Phasenzentrum der Referenzstation und ist ein Rechtssystem mit den Achsen in Richtung Nord, Ost, Unten (N, E, D). Es wird vor allem während der Verarbeitung verwendet und wenn kein übergeordnetes Koordinatensystem gewählt wird, werden die Koordinaten nach der Verarbeitung im Navigationsrahmen ausgegeben. (Lardon, 2022a)

- **Landesvermessung 95 (LV95)**

Das LV95 ist der lokale Referenzrahmen der Schweiz. Er beruht auf 29 GNSS-Permanentstationen des AGNES und bildet das Fundament für sämtliche raumbezogenen Daten der Schweiz (Brockmann et al., 2005). In der vorliegenden Arbeit wird er als übergeordnetes Koordinatensystem für Flüge in der Schweiz verwendet, um die Daten weiterverarbeiten zu können.

- **World Geodetic System 84 (WGS84)**

Das World Geodetic System 84 ist ein weltweit definiertes Referenzsystem, das auf dem amerikanischen GPS System basiert (Kresse and Danko, 2022, p 167). Es dient im Projekt AIR-ETH als übergeordnetes Koordinatensystem, mit dessen Hilfe die prozessierten Daten in andere Programme eingelesen und weiterverarbeitet werden können.

Die verschiedenen Koordinatensysteme sind in Abbildung 2.2 visualisiert.

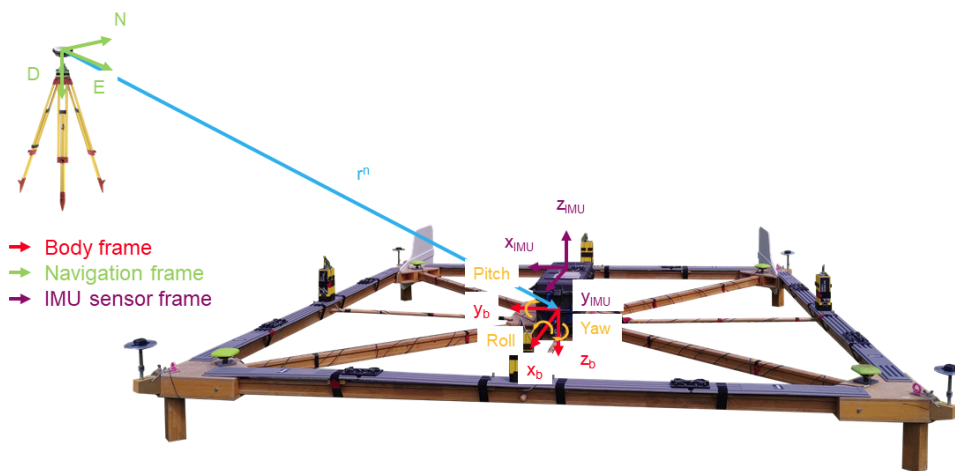


Abbildung 2.2: Verwendete Koordinatensysteme im AIR-ETH Rahmen. Quelle: (Lardon, 2022a)

Kapitel 3

Methodik

In diesem Abschnitt entwickeln wir basierend auf theoretischem Wissen verschiedene Methoden, um die gestellten Aufgaben zu lösen. Hauptaufgaben sind die Überprüfung der Systemintegration des neuen Rahmens der VAW. Zusätzlich adressieren wir die Probleme bezüglich des Kalman-Filters, wie in Kapitel 1 beschrieben. Unser Ansatz umfasst die Vermessung des neuen Rahmens, die Validierung der Systemintegration nach dem Neubau und die Weiterentwicklung der Georeferenzierungs-Software. Die experimentelle Validierung erfolgt schrittweise, während die Softwareentwicklung parallel durchgeführt wird.

3.1 Bodyframe Vermessung

Wie bereits in der Einleitung erwähnt, wurde der ursprünglich schwere Holzrahmen durch einen leichteren Rahmen aus glasfaserverstärktem Kunststoff ersetzt. Dadurch sind auch die Positionen der Antennen und der IMU im Rahmen verändert. Die Positionen sind folglich nicht bekannt, was aber für die Bestimmung der Lage des Rahmens von grundlegender Bedeutung ist. Dementsprechend wurde der neue Rahmen tachymetrisch vermessen.

3.1.1 Setup und Durchführung

Zur Vermessung des Rahmens wurde dasselbe Messkonzept verwendet, welches bereits zur Vermessung des alten Rahmens von Lardon (2022a) bestand. Dieses sieht vor, dass die Koordinaten der Antenne 1 gesetzt werden, um anschliessend die relativen Koordinaten der anderen Antennen respektive der IMU zur Antenne 1 mit

der Leica Nova TS-60 oder einer anderen Totalstation zu bestimmen. Nach erfolgreicher Durchführung des Messkonzepts wurden die Daten exportiert und in einem CSV-File abgespeichert.

3.1.2 Berechnung und Transformation der Daten

Das erzeugte CSV wurde anschliessend mittels eines von Lardon (2022b) entwickelten Skripts prozessiert, um die Punkte vom Vermessungsrahmen der Totalstation in das x, y, z System des Bodyframes zu transformieren. Bei einem Test dieses Skripts gab es jedoch Schwierigkeiten, da der vom bisherigen Code erzeugte Output keine plausiblen Ergebnisse lieferte. Anhand dieser Erkenntnisse wurde der Code genauer betrachtet, wodurch einige nicht nachvollziehbare Berechnungen aufgedeckt werden konnten. Da der Code nicht kommentiert und auch keine dazugehörige Dokumentation verfügbar war, wurde die Umwandlung vom Vermessungs-Koordinatensystem in den Bodyframe eigenständig implementiert.

Der grundlegende Prozess zur Bestimmung der Koordinaten im Bodyframe umfasst folgende Schritte:

1. **Einlesen der Koordinaten:** Die gemessenen Antennenkoordinaten und die Position der IMU im Vermessungsrahmen werden eingelesen.
2. **Bodyframe-Initialisierung:** Für die Bestimmung des Ursprungs (der sogenannte *Pivot Point*) werden zuerst die Koordinaten der diagonal gegenüberliegenden Antennen komponentenweise gemittelt. Anschliessend werden die beiden Ergebnisse erneut gemittelt, um den Ursprung zu erhalten. Anschliessend wird die relative Ausrichtung zum Vermessungsrahmen in Roll-, Pitch- und Yaw-Winkel bestimmt.
3. **Offset-Bestimmung:** Der relative IMU-Offset zum Ursprung des Vermessungsrahmens wird durch Subtraktion des Pivot-Points von den eingemessenen IMU-Koordinaten berechnet.
4. **Koordinatentransformation:** Der Mittelpunkt des Rahmens wird anschliessend als Ursprung des Bodyframes definiert. Mittels einer Translation und Rotation werden die Koordinaten im Bodyframe berechnet.

Mit den erhaltenen Koordinaten im Bodyframe war damit der Grundstein für die Verarbeitung der Daten gelegt.

3.1.3 Integration von Plausibilitätsprüfungen in die Datenverarbeitung

Die Bodyframe-Koordinaten der Antennen für die Lagebestimmung sind von fundamentaler Bedeutung, insofern ist es wichtig, dass die Transformation vom Vermessungsrahmen in den Bodyframe keine Fehler enthält. Da der Code neu entwickelt werden musste, wurden einige Plausibilitätsprüfungen eingeführt. Bereits bestehend gab es eine Prüfung, bei der die Diagonalen verglichen wurden. Wenn sich die beiden Diagonalen zu stark unterscheiden, deutet dies darauf hin, dass der Rahmen nicht richtig zusammengesetzt wurde oder die einzelnen Teile ungenügend fixiert wurden. Zusätzlich dazu werden neuerdings auch die Winkel der Ecken des Rahmens vor und nach der Transformation verglichen, was ein Qualitätsmerkmal für die Transformation liefert. Um eine richtige Reihenfolge der Transformation zu gewährleisten, werden die Rohdaten mit Daten verglichen, welche einmal in den Bodyframe und anschliessend mit der inversen Transformation wieder zurück in den Vermessungsrahmen transformiert wurden.

3.2 Experimentelle Validierung des Systems

In diesem Kapitel wird die Konzeption und Durchführung der vorgesehenen Experimente erläutert. Die Experimente zielen darauf ab, das System kostengünstig und zeitnah zu testen und vorläufige Daten zu sammeln, um die Plausibilität der verschiedenen Sensordaten zu bewerten. Zu diesem Zweck wurden kranbasierte Experimente durchgeführt.

Darüber hinaus wurde ein erster Flugversuch als Helikopter-Unterlast durchgeführt, der vor allem die Anwendbarkeit des neuen Rahmens für die VAW demonstrieren sollte. Während dieses Versuchs wurden erste Daten gesammelt, um die Leistung der Georeferenzierungssoftware zu bewerten.

3.2.1 Dynamik-Test: Kran Experiment

Da die VAW den Rahmen neu konstruiert hat, sollte nach der Bestimmung der Basislinien des Rahmens (siehe 3.1) ein erster dynamischer Test durchgeführt werden. Das Hauptziel dieses Experiments war es, zu beurteilen, ob der Rahmen angemessen auf Richtungsänderungen reagiert und ob er seine allgemeine Integrität beibehält, während er am Kran aufgehängt ist. Dieses Experiment sollte für die vorliegende Ba-

chelorarbeit ausserdem erste Daten aus einem dynamischen Versuchsaufbau liefern und helfen, die Systemabhängigkeiten zu verstehen.

Zu diesem Zweck wurde der Rahmen an einem Turmdrehkran auf dem Campus Höggerberg der ETH Zürich befestigt und mehreren Richtungsänderungen unterzogen. Um eine ausreichende Geschwindigkeit zu erreichen, wurde die Laufkatze am äusseren Ende des Auslegers positioniert, wodurch die Drehrate des Krandrehtellers in eine hohe Rahmengeschwindigkeit umgesetzt wurde. Als Referenzstation wurde die Permanentstation der ETH auf dem HPV Gebäude verwendet.

Vor Beginn des 'Fluges' mussten alle Sensoren eine stabile Periode von etwa 10 Minuten aufzeichnen, um die initiale Lage zu bestimmen. Dazu wurde der Rahmen auf einen stabilen Untergrund ohne Krafteinwirkung gestellt und die Sensordaten aufgezeichnet. So konnte gleichzeitig ein Funktionstest der Kommunikationsschnittstellen und Sensoren durchgeführt werden.

3.2.2 Analyse von Hardware-Problemen mit ublox ANN-MB01 und ZED-F9P

Wie folgend im Unterabschnitt 4.2.1 näher erläutert, erreichten zwei der vier GNSS-Logger mit den zugehörigen Antennen während des Dynamiktests am Turmdrehkran keine stationäre Lösung. Somit standen keine auswertbaren Messdaten des Kranversuchs zur Verfügung. Um dies für zukünftige Flüge zu vermeiden und die Funktionsfähigkeit des Gesamtsystems zu gewährleisten, musste die Fehlerquelle evaluiert und nach Möglichkeit behoben werden. Zur Evaluierung der Fehlerquelle wurden verschiedene Kombinationen der verwendeten Antennen und Logger getestet. Zusätzlich wurden weitere Logger und ältere Antennen für die Tests zur Verfügung gestellt. So konnten alle möglichen Antennen-Logger-Kombinationen - auch mit älteren Antennen - getestet und die daraus resultierenden Probleme generiert werden. Um einen systematischen Ansatz für die Tests zu haben, wurde jede Logger-Antennen-Kombination mit Hilfe einer zuvor erstellten Testtabelle geprüft und die auftretenden Probleme notiert.

Wissend der festgestellten Probleme, die in Unterabschnitt 4.2.2 genauer erläutert werden, wurde versucht, die vier eigentlich baugleichen Logger so zu modifizieren, dass auch Logger 2 und 3 eine Fix-Lösung erhalten. Dies war jedoch mit grösseren Aufwänden verbunden. Es wurde entschieden, die vier ublox ANN-MB01 Antennen gegen vier des Vorgängertyps ublox ANN-MB00 auszutauschen.

3.2.3 Anwendungstest: Helikopterflug Wallis

Nach dem Durchführen des Dynamiktests gemäss Abschnitt 3.2.1 sollte ein umfassender Anwendungstest erfolgen. Dieser Test sollte zeigen, dass das System unter realen Betriebsbedingungen einsatzfähig ist und bleibt. Ausserdem sollte das Verhalten des Rahmens in Extremsituationen, mit hohen Geschwindigkeiten und steilen Kurven, untersucht werden.

Der Versuchsaufbau entsprach dem regulären Einsatzverfahren. Zunächst wurde eine 'stabile Periode' aufgezeichnet, die zur initialen Lagebestimmung dient. Anschliessend wurde der Rahmen mithilfe eines 20 m langen Seils an einer AS-350-B3 (Airbus Helicopters H125) als Unterlast angehoben und in Bereichen des Langsamflugs ($<20 \text{ km h}^{-1}$), der Betriebsgeschwindigkeit (ca. 50 km h^{-1}) sowie bei extremen Geschwindigkeiten (bis zu 130 km h^{-1}) bewegt. Daraufhin wurde ein längeres Seil eingehängt (40 m), um den Einfluss auf das Flugverhalten des Rahmens und des Helikopters auf die GNSS-Signale zu untersuchen. Als Referenzstation sollte das Leica-GNSS-Receiver-System der VAW verwendet werden, jedoch wurde dies in Zürich vergessen. Damit trotzdem die Baselines berechnet werden konnten, wurde eine virtuelle Referenzstation mithilfe des *swipos-GIS/GEO*-Dienstes am Startplatz verwendet.

3.2.4 Analyse nach Anwendungstest

Nach Abschluss des Anwendungstests führte eine Analyse durch, um die Leistung des Systems zu bewerten. Unstimmigkeiten bei den erfassten Positionen und Geschwindigkeiten mussten im Detail untersucht werden.

Zur Sicherstellung der Qualität der Rohdaten wurden diese einer Analyse unterzogen. Dazu wurde das Signal-zu-Rausch-Verhältnis (SNR) und die Multipath-Effekte (MP) während der stabilen Periode und des Fluges analysiert, um mögliche Störquellen zu identifizieren. Zusätzlich wurde ein Vergleich mit den Daten von Lardon (2022a) durchgeführt, um beurteilen zu können, ob es systematische Abweichungen bei allen Antennen gibt. Die Implementierung der IMU-Offset-Korrektur aus Kapitel 3.3.4 wurde erstmals untersucht, um den Einfluss der Inertialsensoren zu überprüfen. Um mögliche Einflüsse auf die Rohdatenverarbeitung zu identifizieren, wurden verschiedene Versionen der RTKLIB-Software verglichen.

Plausibilitätschecks von Gyro-Raten, Alignment und Beschleunigungen

Die von der IMU aufgezeichneten Gyro-Raten und Beschleunigungen mussten dabei auf Plausibilität geprüft werden. Diese Daten wurden dafür mit den erwarteten Werten verglichen.

Besondere Aufmerksamkeit war auf die initiale Lagebestimmung der IMU zu widmen. Eine korrekte Lagebestimmung ist für die weitere Verarbeitung der von ihr aufgezeichneten Daten von entscheidender Bedeutung. Damit können die Beschleunigungen gegenüber der vorherrschenden Erdbeschleunigung korrigiert werden. Das verwendete Verfahren und die Qualität der Ausrichtung wurden überprüft, wodurch die Plausibilität der aufgezeichneten Daten verifiziert werden konnte.

3.2.5 Behandlung des IMU Offsets

Nach dem Helikopter-Experiment gemäss Abschnitt 3.2.3 und anschliessender Plausibilitätsanalyse in Abschnitt 3.2.4 wurden einige Probleme aufgedeckt (siehe Ergebnisse in Kapitel 4). Es wurde entschieden, die IMU in den Ursprung zu platzieren, um den Einfluss auf die Beschleunigungsmessungen zu minimieren.

IMU-Test: Kran Experiment

Aufgrund der Umplatzierung der IMU an den Ursprung wurde darauf ein Test der IMU sowie der gesamten Systeme nach der Umstellung durchgeführt. Dafür wurde aus Gründen der Verfügbarkeit ein Brückenkran ausserhalb des HIA Gebäudes der ETH Zürich verwendet. Dieser steht in Nordwest-Südost Ausrichtung und kann sich linear in Südwest-Nordost Richtung bewegen. Eine Gebäudefassade ist in Nordost-Richtung, eine weitere in Ost-Richtung (siehe Abbildung 3.1). Weiter ist auf dem Areal ein Materiallager rund um den Kran.



Abbildung 3.1: Brückenkran (weiss) bei HIA Gebäude mit Materiallager. Bilddaten: CNES, Spot Image, swisstopo, NPOC

Es war daher zu erwarten, dass die GNSS-Beobachtungen mit Multipath-Effekten und Abschattungen gestört werden. Als Referenzstation wurde die Permanentstation der ETH auf dem HPV Gebäude verwendet.

Das Experiment sollte die Systemfunktionalität aufzeigen und einen ersten Datensatz zur Prüfung des Systems nach der Umplatzierung liefern. Die Sensordaten der IMU wurde auf Plausibilität geprüft und eine erste Verarbeitung durchgeführt.

3.3 Softwareentwicklung und DevOps

Das bestehende Georeferenzierungssystem liefert bereits zufriedenstellende Ergebnisse, die mit dem *JAVAD*-System vergleichbar sind. Dennoch hat Lardon (2022a) Optimierungsmöglichkeiten identifiziert. Im folgenden Abschnitt beschreiben wir die Weiterentwicklung der Software zur Behebung dieser Probleme und stellen zusätzlich Lösungsansätze für neu entdeckte Probleme vor.

3.3.1 Behebung von Azimutwinkel-Orientierungsfehlern im Kalman-Filter

Eine dieser Optimierungsmöglichkeiten stellte der Übergang von -180° auf 180° oder umgekehrt dar. Im bisherigen Code führte dieser Übergang zu grossen Schwierigkeiten, da der Kalman-Filter ausbrach und danach wieder Konvergenzzeit benötigte. Diese Konvergenzzeit betrug meist nur wenige Epochen, jedoch führten die Ausreisser zu grösseren Abweichungen im Vergleich mit den *JAVAD* Daten (Lardon, 2022a).

Da dieses Problem im Kalman-Filter entstand, wurde dieser genauer betrachtet, wodurch auffiel, dass die allermeisten Berechnungen für die Lage im Kalman-Filter auf Einheitsquaternionen beruhten. Diese Berechnungen konnten dementsprechend nicht Auslöser des Problems sein, da das Problem aufgrund der Eigenschaften von Quaternionen nicht auftreten kann. Es stellte sich jedoch heraus, dass die Innovationmatrix mit Euler-Winkeln gerechnet werden. Dies führte dazu, dass die Yaw-Winkel, welche den -180° zu 180° Übergang, oder umgekehrt, durchlaufen, eine betragsmässig zu grosse Innovationmatrix zur Folge hatten. So wurden beispielsweise für den Übergang von 178° auf -174° 352° in die Innovationmatrix eingetragen, statt der tatsächlichen 8° . Dies führte dazu, dass im Update-Step eine zu grosse Korrektur zum Predicted-State hinzugefügt und so ein Ausreisser erzeugt wurde.

Dieses Problem konnte jedoch gelöst werden, indem der Wertebereich angepasst wurde, sodass sich der Yaw-Winkel beliebig ausweiten kann und auch Werte über 180° bzw. unter -180° annehmen kann. Das konnte dadurch erreicht werden, dass Yaw-Winkel-Differenzen, welche grösser als 180° sind, mit 360° addiert bzw. subtrahiert werden. Dadurch gibt es keine Probleme mehr mit dem Übergang und da die Korrektur nur die Innovationmatrix betrifft, hatte diese auch keine weiteren ungewollten Auswirkungen auf den Rest der Verarbeitung.

3.3.2 Erhöhung der Systemverfügbarkeit durch Redundanz in GNSS-Antennennutzung

Eine weitere Optimierungsmöglichkeit wird ebenfalls bereits in (Lardon, 2022a) beschrieben. Es handelt sich hierbei um die Erhöhung der Systemverfügbarkeit durch Lagebestimmung aus nur drei verfügbaren GNSS-Messungen. Dies hat den Vorteil, dass so auch mit nur drei verfügbaren Antennen eine Lösung gerechnet werden kann, mit welcher ein Update-Step durchgeführt werden kann. Dadurch kann auch bei Ausfällen einer Antenne eine präzise Lagebestimmung gewährleistet werden, welche

nicht von den über die Zeit stark ansteigenden Fehlertermen der IMU abhängig ist. Um dies zu ermöglichen, wurde der Umgang mit drei verfügbaren Antennen in der Verarbeitung überarbeitet. Statt diese Messungen wie bis anhin zu überspringen, wird zuerst geprüft, welche drei Antennen verfügbar sind. Von den verfügbaren Antennen werden die Daten in eine neue flexiblere Datenklasse übergeben, mit welcher anschliessend die Lageschätzung durchgeführt wird. Dafür wird, wie bereits in Lardon (2022a) ein vermittelndes Ausgleichungsmodell aus Piringer (2018) verwendet. Um dieses für nur drei Antennen anzupassen, wurde die Koeffizientenmatrix der Beobachtungsgleichungen und das funktionale Modell durch Verändern der Koeffizientenmatrix der Beobachtungsfunktion angepasst. Das restliche Modell konnte aus dem Fall für vier verfügbare Antennen übernommen werden. Anschliessend musste der Mittelpunkt des Rahmens für die weitere Verarbeitung geschätzt werden. Dies wurde für jede mögliche Kombination aus verfügbaren Antennen implementiert. Dafür wird mit den vorhandenen Baselines der drei verfügbaren Antennen die Position der vierten Antenne geschätzt. Aus diesen vier Positionen wird anschliessend der geschätzte Mittelpunkt des Rahmens ermittelt.

Tests der Implementierung

Um die neue Implementierung für eine Lageschätzung mit nur drei verfügbaren Antennen zu prüfen, wurde dieses Szenario künstlich erzeugt. Dafür wurden zuerst die benötigten Daten manipuliert, indem Beobachtungen aus den GNSS-Files gelöscht wurden. Daraufhin wurde die Verarbeitung erneut durchlaufen. Die resultierenden Daten wurden verglichen, indem die geschätzten Roll-, Pitch- und Yaw-Winkel aus den vier verschiedenen möglichen Kombinationen (Ausfall je einer Antenne) mit den Winkeln der Lösung mit vier verfügbaren Antennen verglichen wurden.

3.3.3 Implementierung der rückwärtslaufenden Verarbeitung

Um die Ausreisser des Yaw-Winkels im Kalman-Filter zu verhindern, hat Lardon (2022a) empfohlen, einen kombinierten Kalman-Filter aus vorwärts und rückwärtslaufendem Filter zu implementieren. Dies soll die resultierenden Yaw-Winkel glätten und somit die Ausreisser bereinigen. Dieser Ansatz wurde ebenfalls parallel zu der Korrektur der Innovationmatrix verfolgt. Nachdem der Ansatz mit der Innovationmatrix das bestehende Problem jedoch behoben hatte, wurde entschieden diesen Ansatz nicht mehr weiterzuverfolgen, da andere Bestandteile der Arbeit vonseiten der VAW weiter oben auf der Prioritätenliste standen. Aufgrund der begrenzten Zeitverfüg-

barkeit wurde entschieden, die Fertigstellung der rückwärtslaufenden Verarbeitung zu verschieben, obwohl bereits grosse Teile davon bestehen.

3.3.4 Implementierung der Korrektur von IMU-Offset

In der Masterarbeit (Lardon, 2022a) wurde die IMU im Zentrum des Rahmens platziert. Der Offset der IMU wurde nicht berücksichtigt. Durch einen Offset werden die gemessenen Beschleunigungen am Ort der IMU durch Rotationen vergrössert. Dies lässt sich mit der Formel zur Transformation von Beschleunigungen in einem beliebigen Punkt A zum Massenschwerpunkt abschätzen:

$$\vec{a}_C = \vec{a}_A + \dot{\vec{\omega}} \times \vec{c} + \vec{\omega} \times \vec{\omega} \times \vec{c}$$

Wobei:

\vec{a}_C : Beschleunigung im Massenzentrum

\vec{a}_A : Beschleunigung am Punkt A

$\vec{\omega}$: Drehraten entlang der Achsen

\vec{c} : Offset von Massenzentrum zu Punkt A

$\dot{\vec{\omega}}$: Winkelbeschleunigungen

Zur Abschätzung können maximale Winkelgeschwindigkeiten 2 rad s^{-1} angenommen werden, da bei höheren Geschwindigkeiten die Integrität des Rahmens sehr wahrscheinlich nicht mehr gewährleistet ist. Durch die Vermessung des Bodyframes in Kapitel 3.1 ist der aktuelle IMU-Offset bekannt ($\|\vec{c}\| \approx 10 \text{ cm}$). Damit konnte abgeschätzt werden, dass die maximalen Korrekturen der Beschleunigungen des Rahmens der VAW vor der Umplatzierung der IMU in Kapitel 3.2.5 um 0.1 m s^{-2} betragen und entsprechend korrigiert werden sollten.

Wenn die Achsen der IMU und des Bodyframes ähnlich ausgerichtet sind, d.h. die Fehlausrichtung (*Misalignment*) sehr klein ist, können die Beschleunigungen wie folgt korrigiert werden (Gainer et al., 1972, S. 69, Case II):

$$\vec{a}_C = \vec{a}_A + \begin{vmatrix} -\omega_y^2 - \omega_z^2 & \omega_x \omega_y - \dot{\omega}_z & \omega_x \omega_z + \dot{\omega}_y \\ \omega_x \omega_y + \dot{\omega}_z & -\omega_x^2 - \omega_z^2 & \omega_y \omega_z + \dot{\omega}_x \\ \omega_x \omega_z - \dot{\omega}_y & \omega_y \omega_z + \dot{\omega}_x & -\omega_x^2 - \omega_y^2 \end{vmatrix} \vec{c}$$

Da die Fehlausrichtung der IMU bezüglich der Bodyframe-Achsen unbekannt und im Vergleich zur allgemeinen Starrheit vernachlässigbar ist, wurde auf eine Korrektur der Winkelgeschwindigkeiten verzichtet. Diese können wie in Gainer et al. (1972, S. 69) implementiert werden.

3.3.5 Überarbeitung der Transformation ins LV95

Ein weiterer Punkt, in welchem die bestehende Verarbeitung noch Optimierungsbedarf aufwies, war die Transformation vom Navigationssystem in das LV95. Im bisherigen in Lardon (2022a) beschriebenen Code wurden die Koordinaten der Referenzstation dem Ursprung des Navigationsrahmens zugewiesen und alle gemessenen Koordinaten wurden mit den Koordinaten der Referenzstation addiert. Dies lieferte näherungsweise gute Werte, jedoch sind gerade in der Höhenkomponente Fehler zu erwarten. Aus diesem Grund wurde die bestehende Transformation überarbeitet.

Die erste Implementierung nutzte für die Transformation die Library `Swissreframe`, welche von `Swisstopo` entwickelt wurde. Aufgrund von daraus resultierenden, enorm langen Verarbeitungszeiten wurde entschieden eine Alternative zu finden, welche mit `GeoPandas` zur Verfügung stand.

3.3.6 Kontrollfenster für Verarbeitung

In der bestehenden Verarbeitung mussten die Benutzenden während der Datenverarbeitung mehrere Schritte ausführen und verschiedene Werte manuell eingeben. Um diesen Prozess zu vereinfachen und die Benutzerfreundlichkeit zu erhöhen, sollte ein Parameterfenster implementiert werden, in dem der Benutzer vor Beginn der Verarbeitung alle erforderlichen Werte einstellen kann.

Im Parameterfenster werden die folgenden Eingabefelder und Optionen bereitgestellt:

- **Arbeitsordner:** Auswahl des Ordners, in dem die Ergebnisse gespeichert werden.
- **IMU-Datei:** Auswahl der IMU-Datei für die Verarbeitung.
- **Parameter für stabile Periode:** Ermöglicht die Eingabe einer benutzerdefinierten stabilen Periode.
- **GNSS-Dateien:** Auswahl der Positionsdateien der vier GNSS-Antennen in der richtigen Reihenfolge.

- **Bodyframe-Koordinaten:** Auswahl der Datei mit den Koordinaten der Antennen im Bodyframe.
- **GNSS sampling rate:** Eingabefeld zur Einstellung der GNSS-Abtastrate.
- **Altimeterdatei:** Auswahl der Altimeter-Datei.
- **Jahreszeit:** Auswahl, ob die Messungen während der Sommer- oder Winterzeit durchgeführt wurden.
- **Ausgabe-Koordinatensystem:** Auswahl des gewünschten Koordinatensystems (LV95/WGS84) für die Datentransformation.
- **Details der Output-Datei:** Angabe von Flugort und Flugsegment, die im Dateinamen enthalten sein sollen.

Im Parameterfenster ist eine Schaltfläche 'Verarbeitung starten' vorgesehen. Diese Schaltfläche startet die Datenverarbeitung, falls alle erforderlichen Parameter eingestellt sind. Nachdem die Verarbeitung angelaufen ist, ist von Seite des Nutzens keine Eingabe mehr nötig. Durch die Reduktion der manuellen Eingaben und einem besser organisierten Arbeitsablauf bei der Datenverarbeitung wird die Benutzerfreundlichkeit erheblich verbessert.

3.3.7 Implementierung von DevOps-Prinzipien

DevOps ist ein organisatorischer Ansatz für die effektive Zusammenarbeit in und zwischen Teams (insbesondere zwischen Development und Operations) in Softwareentwicklungsunternehmen oder -projekten, um die Bereitstellung von Änderungen und zuverlässigen Systemen zu gewährleisten (Dyck et al., 2015). Obwohl DevOps eine Abkürzung für die Bereiche Entwicklung und Betrieb ist, sollte der Begriff nicht ausgeweitet werden, da er per Definition nicht ausschliesslich auf diese Bereiche beschränkt ist (Dyck et al., 2015). Wir orientierten uns an gängigen Praktiken, um die Codebasis des Projekts in eine Form zu bringen, die die Anwendung von Methoden moderner Developer Operations ermöglicht.

Um die zukünftige Zusammenarbeit am Code und die nutzerunabhängige Anwendung zu ermöglichen, müssen alle Softwarekomponenten verfügbar und dokumentiert sein (Carver et al., 2016). Dazu eignet sich ein zentral gehostetes, hochverfügbares Git-Repository, wie es die ETH Zürich mit der GitLab-Infrastruktur zur Verfügung stellt (Winters et al., 2020).

Etablierung von Git für zentrale Versionskontrolle und Zusammenarbeit

Der bestehende Code wurde von Fabrice Lardon als GitLab Repository zur Verfügung gestellt. Leider war dieser Code nicht ausführbar. Die VAW hatte ein Laufwerk mit funktionierendem Code, aber es fehlten einige Codebestandteile. Zur Verbesserung für zukünftige Arbeiten wurde ein GitLab-Repository erstellt, in dem alle Programmteile gesammelt und die Änderungen ab dem Zeitpunkt dieser Bachelorarbeit versioniert werden. Die Benutzerrollen wurden so verteilt, dass die Eigentümer:innen über die Institute verteilt sind, so dass der Zugriff bei Personalabgängen gewährleistet bleibt. Die Ordner-/Dateistruktur wurde bereinigt, um die Einarbeitungszeit zu reduzieren.

Dokumentation des Prozesses und der Anwendung

Eine vollständige Dokumentation für die Anwender:innen war nicht vorhanden. Es waren einzelne Dateien in den verschiedenen Instituten verteilt. Eine zentrale Dokumentation ist jedoch sinnvoll, um die Nutzung auch nach einer längeren Pause (z.B. im Sommer) zu ermöglichen. Zu diesem Zweck wurde die Wiki-Funktion von GitLab genutzt. Diese erlaubt die Anzeige von Markdown Dateien im Browser. Sobald der Zugriff auf das Repository gegeben ist, soll direkt auf die Dokumentation zugegriffen werden können. Die Inhalte sollen nicht nur den Prozess der Verarbeitung vermitteln, sondern auch alle Voraussetzungen und Schritte, die für die Vorverarbeitung notwendig sind:

- **Bodyframe Vermessung:** Anweisungen für die Bestimmung des Bodyframes bzw. für Messungen mit der Totalstation.
- **Datenerfassung und -verarbeitung:** Anweisungen für den Download von Ephemeriden und den Export von Daten aus den verschiedenen Sensoren.
- **Datenverarbeitung:** Anweisungen zur Datenverarbeitung mit RTKLIB und anschließender Verarbeitung mit der Georeferenzierungssoftware.

Darüber hinaus wurde im Rahmen der Bachelorarbeit ein Wissenstransfer für die VAW umgesetzt, der sie in die Lage versetzen soll, die Auswertung mit Hilfe der besprochenen Dokumentation in RTKLIB und der Verarbeitung durchzuführen.

Kapitel 4

Ergebnisse

In diesem Kapitel werden die Ergebnisse der behandelten Methoden aufgezeigt. Dabei wird detailliert auf die Resultate der einzelnen Arbeitsprozesse eingegangen.

4.1 Ergebnisse der Bodyframe-Vermessung

Die Vermessung des Bodyframes, wie in Abschnitt 3.1 genauer beschrieben, wurde im neu entwickelten Skript ausgewertet. Dabei wurde festgestellt, dass die beiden Diagonalen des Rahmens mit einer Abweichung von unter 2 cm übereinstimmen. Dies ist ein Indiz dafür, dass der Rahmen richtig aufgestellt und die Fixierungen korrekt befestigt wurden. Weiter stimmen die Seitenlängen mit Abweichungen unter 2 cm überein.

Die Plausibilitätsprüfung durch die Winkelbestimmung vor und nach der Transformation zeigte, dass die Winkel sich nur um numerische Instabilitäten ($< 10^{-13}$) unterscheiden. Dies zeigt, dass die Transformation die gewünschten Effekte hat. Auch nach der Rücktransformation traten ausschliesslich numerische Instabilitäten auf. Daraus kann ebenfalls auf eine korrekte Transformation geschlossen werden.

Die Verformung des Rahmens während des Fluges ist nicht bekannt. Optisch konnte jedoch erkannt werden, dass die Verformung durch die Seilauflängung deutlich grösser ist, als die Genauigkeit der eingemessenen Punkte. Daraus folgend ist die erforderliche Messgenauigkeit genügend. Aufgrund dieser Ergebnisse wurde entschieden, die gemessenen Koordinaten und die daraus durch Transformation gewonnenen Bodyframe-Koordinaten als ausreichend anzusehen und weiter zu verwenden.

4.2 Ergebnisse der experimentellen Validierung des Systems

Im nachfolgenden Abschnitt werden die Ergebnisse sämtlicher durchgeführter Experimente und Analysen erläutert.

4.2.1 Ergebnisse des Dynamik-Tests: Kran Flugexperiment

Durch den Dynamiktest am Turmdrehkran am Hönningerberg wurde eine Fehlfunktion der GNSS-Sensorik aufgedeckt. Die Logger 2 und 3 mit den dazugehörigen Antennen konnten keine Fix-Lösung rechnen. Diese Schwachstelle konnte auch nicht durch gängige Problemlösungsverfahren wie beispielsweise das Neustarten der Logger behoben werden. Daher musste der Rahmen nach diesem Test als nicht funktionsbereit bezeichnet werden.

Nichtsdestotrotz lieferte der Test positive Ergebnisse. Der Rahmen zeigte auch bei dynamischen Versuchen ein stabiles Flugverhalten und richtete sich innert kurzer Zeit weitgehend in Flugrichtung aus. Des Weiteren konnte durch die im Turmdrehkran eingebaute Waage evaluiert werden, dass eine markante Gewichtsreduktion vom ursprünglichen Holzrahmen zum neuen GFK-Rahmen erreicht wurde. Aufgrund der Ablesbarkeit der im Kran montierten Waage von 100 kg konnte die Gewichtsreduktion nicht genau beziffert werden. Dennoch kann gesagt werden, dass es der VAW gelungen ist, bei der Neugestaltung des Rahmens eine Gewichtsreduktion zu erreichen.

4.2.2 Erkenntnisse aus der Analyse von Hardware-Problemen mit ublox ANN-MB01 und ZED-F9

Aufgrund der Durchführung eines Dynamiktests am Kran konnten viele Erkenntnisse gewonnen werden. Bei der anschliessenden Prüfung der verschiedenen Bestandteile fiel auf, dass die Logger 1 und 4 mit allen verfügbaren Antennen keine Probleme bereiteten. Logger 2 und 3 erreichten jedoch nur bei gewissen Antennen eine Fix-Lösung. Da aus den Antennen, welche mit den Loggern 2 und 3 keine Fix-Lösung erhielten, keine weiteren Muster, wie beispielsweise dass nur die neuen Antennen Probleme bereiten, erkennbar waren, wurden die Kombinationen erneut getestet. Dabei fiel auf, dass gewisse Kombinationen, welche bisher nur Float-Lösungen erreichten, plötzlich eine Fix-Lösung erreichen und umgekehrt. Nach mehreren Tests und Versuchen zur Reproduzierbarkeit der Ergebnisse fiel auf, dass die Logger 2

und 3 nur eine Fix-Lösung erhielten, wenn die Antenne nach dem Aufstarten des Loggers angeschlossen wurde. Falls die Antenne bereits im ausgeschalteten Zustand angeschlossen wurde, konnte keine Fix-Lösung erreicht werden. Das Problem kann somit zuverlässig umgangen werden, indem die Logger zuerst angeschaltet und erst danach die Antennen angeschlossen werden. Des Weiteren stellte sich heraus, dass dieses Problem nur mit den ublox ANN-MB01 Antennen besteht, nicht aber mit dem Vorgängermodell, den ublox ANN-MB00 Antennen.

4.2.3 Ergebnisse des Anwendungstests: Helikopterflug Wallis

Bei dem in Unterabschnitt 3.2.3 erwähnten Experiment konnten verschiedenen Ergebnisse erzielt werden. Zuerst konnte durch den Testflug bestätigt werden, dass der Rahmen den Kräften, welche mit einem Unterlastflug unter dem Helikopter verbunden sind, gewachsen ist. Der Rahmen liegt stabil in der Luft und richtet sich bereits bei geringer Fluggeschwindigkeit in Flugrichtung aus. Des Weiteren können auch mit angehängtem Rahmen Reisegeschwindigkeiten von bis zu 130 km h^{-1} zurückgelegt werden, um zu dem zu messenden Gebiet zu gelangen. Auch bei diesen hohen Geschwindigkeiten funktioniert die Georeferenzierung. Jedoch sind die Daten des Altimeters nicht mehr aussagekräftig, da der Rahmen sich aufgrund des Luftwiderstands stark in Pitch-Richtung neigt. Bei der deutlich geringeren Betriebsgeschwindigkeit von maximal 55 km h^{-1} wird der Rahmen nur geringfügig gekippt, weshalb dieser Einfluss vernachlässigt werden kann. Zudem konnte durch den Testflug festgestellt werden, dass die vorhandenen Seitenflossen des Rahmens verstärkt werden müssen, um zu verhindern, dass sie bei Start und Landung aufgrund des Abwindes des Helikopters abknicken.

Die verwendeten Sensoren stellten ihre Funktionstüchtigkeit beim Testflug unter Beweis. So gab es, dank der aus Lardon (2022a) bewährten Kombination aus ANN-MB00 Antennen und ZED-F9P Loggern keine Komplikationen mit dem GNSS-Empfang. Auch der Altimeter bereitete keine Schwierigkeiten. Die IMU löste keine unmittelbaren Komplikationen aus, jedoch zeigte sich in den von ihr aufgezeichneten Daten ein starker Drift im Yaw-Winkel, welcher ungefähr 3° min^{-1} beträgt und in Unterabschnitt 4.2.4 genauer untersucht wird.

Für den Testflug im Wallis wurde geplant, eine Leica GS14 als Referenzstation einzusetzen. Diese wurde jedoch in Zürich vergessen, weshalb für die Auswertung

der Daten eine virtuelle Referenzstation gerechnet wurde. Aufgrund der virtuellen Referenzstation konnten jedoch nur noch Genauigkeiten in den Baselines im Bereich von 3-5 cm erreicht werden. Ein weiterer Faktor, welcher die Genauigkeit der GNSS-Messungen beeinflussen könnte, ist die Abschattung durch den Helikopter. Dies wurde durch verschiedene Seillängen von 20 m und 40 m untersucht, wobei kein bemerkenswerter Unterschied festgestellt wurde. Die Qualität der aus RTKPOST erhaltenen Baselines war bei beiden Seillängen in ähnlichen Größenordnungen.

Der Testflug lieferte des Weiteren die für die Prüfung der Verarbeitung wichtigen Daten. Erstmals wurden vollständige Daten mit dem neuen GFK-Rahmen aufgenommen, welche nun ausgewertet werden konnten. Dabei fiel auf, dass die bisherige Verarbeitung mit den Daten des neuen Rahmens nicht mehr funktionierte. Stattdessen wurden Werte ausgegeben, welche nicht realistisch waren. So wurden in der Verarbeitung Geschwindigkeiten von rund 100 m s^{-1} berechnet und die Position war, wie in Abbildung 4.1 sichtbar, nicht plausibel.

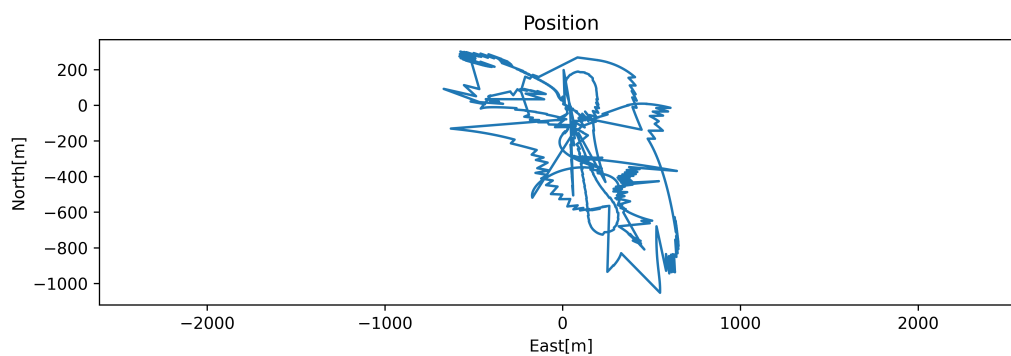


Abbildung 4.1: Berechnete Position nach der Verarbeitung im Navigationssystem

4.2.4 Analyse nach Anwendungstest

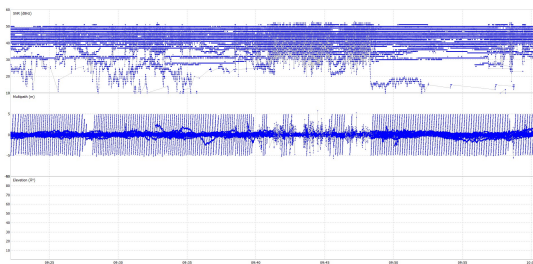
Analyse des SNR und der Multipath-Effekte

Zur Beurteilung der Qualität der GNSS-Loggeraufzeichnungen wurden das Signal-Rausch-Verhältnis (SNR) und der geschätzte Multipath wie in Unterabschnitt 3.2.4 beschrieben auf Auffälligkeiten untersucht. Diese Parameter wurden mit RTKLIB aus den konvertierten Rinex-Dateien der GNSS-Logger ausgewertet. In Abbildung 4.2 sind diese Plots für jede Antenne dargestellt.

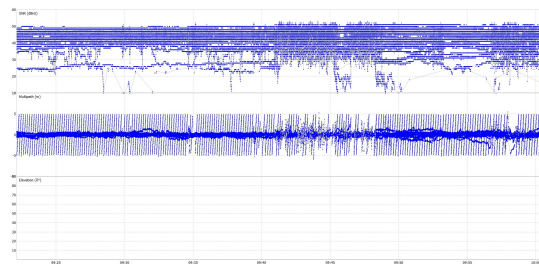
Während der **stabilen Periode** war das SNR (hier spezifisch das *Carrier-to-noise-Verhältnis CNR*) der gut sichtbaren Satelliten konstant um 35 bis 50 dB Hz. Bei

schlechter Geometrie können auch Satelliten mit einem sehr niedrigen CNR von unter 15 dB Hz noch verfolgt werden. Multipath-Effekte werden mit -5 bis 5 m geschätzt. Dies ist auf den Ort der Aufzeichnung zurückzuführen, an dem die Signale von einer Hauswand reflektiert wurden.

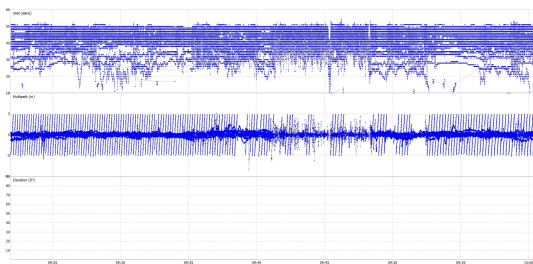
Während des **Fluges** ist der Wertebereich des CNR weiter gestreut (20 bis 50 dB Hz). Unterhalb von 20 dB Hz werden jedoch kaum mehr Signale aufgezeichnet. Die MP-Effekte nehmen während des Fluges auf unter -2 bis 2 m ab. Wenn zwischen den Flugperioden gelandet wird, um das längere Seil einzuhängen, sind die Einflüsse der Hauswand sofort wieder zu beobachten.



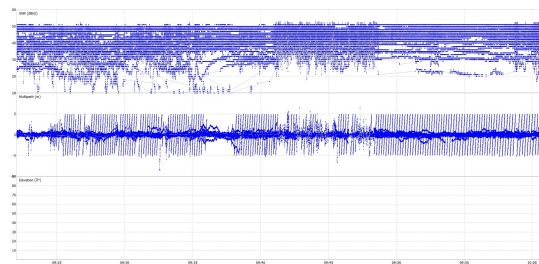
(a) Antenne 1



(b) Antenne 2



(c) Antenne 3



(d) Antenne 4

Abbildung 4.2: SNR & MP von allen Antennen

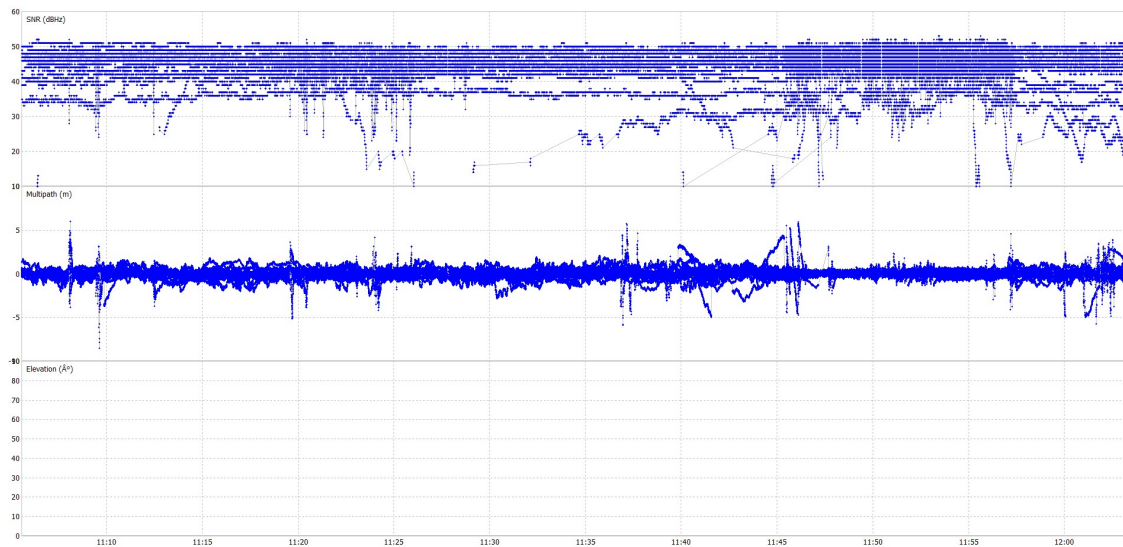


Abbildung 4.3: RTK PLOT von Flug mit altem Rahmen am 23. Mai 2022

Eine Periode aus der Arbeit von Lardon (2022a) wurde betrachtet und mit den obigen Ergebnissen verglichen, um Veränderungen im System abschätzen zu können. Dieser Flug wurde im Bereich des Morteratschgletschers am 23.5.2022 durchgeführt. Dabei wurde ein ähnliches CNR und kaum Multipath-Effekte festgestellt, da im Gegensatz zum Ort keine Gebäude vorhanden waren. Wie in Abbildung 4.3 zu sehen ist, nimmt die Streuung des CNR während des Fluges zu. Die MP-Effekte bleiben klein -1 bis 1 m. Dies ist vergleichbar mit den aufgezeichneten Daten des Anwendungstests.

RTKPOST Processing

Für die Berechnung der DGNSS-Baselines zwischen Antennen und Basisstation wird RTKPOST verwendet. Hier gibt es verschiedene Versionen, welche je nach Anwendungsfall gewählt werden sollten. Zu Beginn verwendeten wir das RTKLIB-b34 für Windows von rktlib.com. Diese Version ist nicht für low-cost Receiver optimiert, weshalb sich für diese Anwendung RTKLIB-demo5 anbietet.

Im Vergleich können ähnlich viele Baselines gerechnet werden, die Qualitätsparameter und auch das Verhalten sind aber mit demo5 deutlich besser. So waren die Standardabweichungen mit unseren Verarbeitungseinstellungen in Lage 1 bis 2 cm und in Höhe um Faktor 2 besser. Ausserdem war das Verhalten mit dem nicht optimierten RTKLIB fehleranfällig. Manchmal verschob sich die Position einer Antenne während der stabilen Periode um mehrere Meter (unabhängig von den anderen),

so dass die gerechnete GNSS-Rahmenlage stark von der Realität abweicht. Damit führen die durch Lardon (2022a) sehr hoch gewichteten Update-Steps zu starken Abweichungen und die gesamte Navigationslösung wird schlecht. Dieses Problem ist mit demo5 nicht mehr aufgetreten.

In RTKLIB können ausserdem die Verarbeitungsparameter aus Lardon (2022a) an die Umstände angepasst werden. Beispielsweise ist es wichtig, bei möglicher Umgebung für Multipath-Effekten eine andere Elevations-Maske zu wählen, oder die SNR-thresholds anzupassen. Eine vollständige Beschreibung der Optionen findet sich in Takasu (2021).

Plausibilität von Gyro-Raten, Alignment und Beschleunigungen

Um sicherzustellen, dass die schlechte Positionslösung nicht aus unzuverlässigen oder gar fehlerhaften IMU-Aufzeichnungen zurückzuführen ist, wurde die Plausibilität der Daten wie in Abschnitt 3.2.4 beschrieben überprüft.

Dabei wurden die **Drehraten der Gyrosensoren** entlang der Koordinatenachsen des IMU-Frames auf ihre Grössenordnung geprüft. Diese betragen in ihrer Magnitude maximal 1 rad s^{-1} , was ca. 60° pro Sekunde entspricht (siehe Tabelle 4.1). Aus den aufgezeichneten Videos des Fluges ist dieser Wert nachvollziehbar. In schnellen Steilkurven sowie auch bei Start und Landung können diese Drehraten durchaus erreicht werden.

Parameter	Max. Magnitude	Einheit
X_{max}	14.15	$^\circ \text{ s}^{-1}$
Y_{max}	9.16	$^\circ \text{ s}^{-1}$
Z_{max}	55.18	$^\circ \text{ s}^{-1}$

Tabelle 4.1: Maximale Grössen der Winkelgeschwindigkeiten

Bezüglich den **Beschleunigungen** muss zwischen *gravitations-unkorrigierten* und *effektiven* Beschleunigungen unterschieden werden. Um eine Aussage bezüglich ersteren zu treffen, wurden die Aufzeichnungen in der stabilen Periode untersucht. Wäre der IMU-Rahmen in z-Richtung parallel dem Schwerevektor ausgerichtet, müssten die Beschleunigungen in x- und y-Richtung nur Sensorrauschen beinhalten und in z-Richtung zusätzlich die gemessene Erdbeschleunigung. Da diese Ausrichtung nicht gegeben ist, wird die Erdbeschleunigung mit allen Richtungen gemessen. Um trotzdem eine Aussage machen zu können, wurde das Messrauschen untersucht. Während

der stabilen Periode ist das Messrauschen aller Achsen vergleichbar. Aus Tabelle 4.2 ist zu entnehmen, dass das Signalrauschen im $1\sigma_{gyro}$ Bereich von $<0.07 \text{ m s}^{-2}$ liegt.

Parameter	1σ	Einheit
X_{acc}	0.010 89	m s^{-2}
Y_{acc}	0.010 87	m s^{-2}
Z_{acc}	0.010 04	m s^{-2}

Tabelle 4.2: Standardabweichungen der Beschleunigungen während 15 Minuten stabile Periode

Parameter	1σ	Einheit
$Pitch$	0.067 59	$^\circ$
$Roll$	0.045 51	$^\circ$
Yaw	22.481 56	$^\circ$

Tabelle 4.3: Standardabweichungen der Orientierung während 15 Minuten stabile Periode

Während der stabilen Periode kann die *initiale Lage* bestimmt werden (Alignment) und damit die Erdbeschleunigung korrigiert werden, so dass nur noch die *effektiven Beschleunigungen* der IMU zur Navigation verwendet werden. Während der stabilen Periode sollten durch die Differenzbildung mit dem Gravitationsvektor die Beschleunigungsdaten in alle Richtung nur noch Messrauschen beinhalten. Ist diese Korrektur fehlerhaft, finden sich Einflüsse der Erdbeschleunigung in den Daten. Ausserdem kann je nach Verfahren die Orientierung bezüglich der Erdrotationsachse bzw. magnetisch Nord festgestellt werden.

Um eine Aussage bezüglich der *effektiven Beschleunigungen* zu machen, wurde zuerst das **Alignmentverfahren** analysiert. In der Implementierung von Lardon (2022a) wird der Alignment-Algorithmus des Herstellers der IMU für die weitere Verarbeitung genutzt. Standardmässig wird die Orientierung der z-Achse anhand des lokalen Erdmagnetfeld bestimmt (Movella, 2020, Kapitel 4.4.3). Die initiale Inklination (Pitch und Roll) wird mit der Annahme bestimmt, dass die durchschnittliche *effektive Beschleunigung* null ist, was während der stabilen Periode auch gegeben ist. Mithilfe der Erdbeschleunigung können dann die Achsen stabilisiert werden (Movella, 2020, Kapitel 4.4.2). In der stabilen Periode sollten mit dieser IMU also alle Winkel konstant und nur mit dem Messrauschen der Gyrosensoren bzw. des Magnetometers gemäss Varianzfortpflanzungsgesetz behaftet sein.

Um die **Qualität des Alignments** zu prüfen, wurden die Winkel auf ihre Stabilität untersucht. Wie in Tabelle 4.3 zu erkennen, fällt sofort der Yaw-Winkel mit einem $\sigma_z = 22.5^\circ \text{ s}^{-1}$ auf. Die anderen Winkel sind ähnlich, aber haben eine deutlich höhere Standardabweichung als beim Flug mit dem alten Rahmen am Morteratschgletscher von Lardon (2022a). Bei genauerer Betrachtung des Yaw-Winkels fällt ein starker

Drift auf. Während den ersten 15 Minuten der stabilen Periode driftet dieser knapp 100° , wie in Abbildung 4.4 zu erkennen ist.

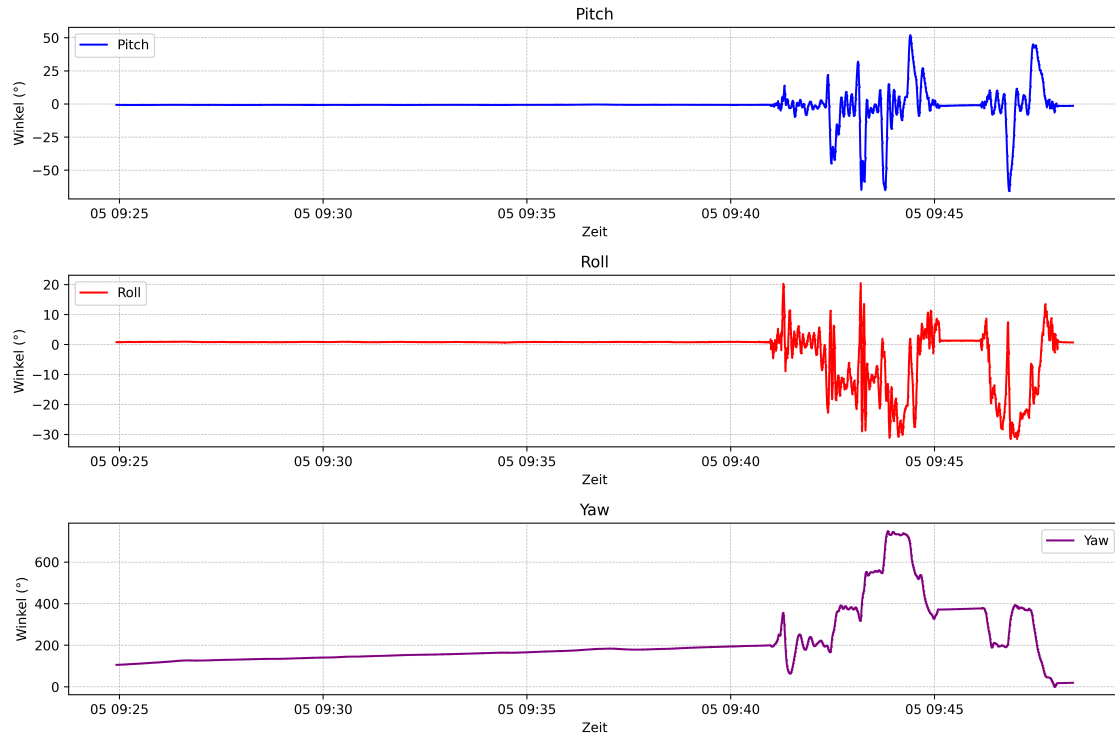


Abbildung 4.4: Yaw über die Zeit

Um zu überprüfen ob die Gravitationskorrektur richtig angebracht wurde, wurden die Beschleunigungen im Navigationsrahmen nach Abzug von \mathbf{g} geprüft. Wie in Abbildung 4.5 zu erkennen, ist die Korrektur entlang der z-Achse gelungen. Vor allem aber in x- und z-Richtung gibt es im Mittel deutlich zu hohe Beschleunigungen. Dies ist in Tabelle 4.4 klar zu sehen.

Parameter	Mean (m s^{-2})	σ (m s^{-2})
X_{stable}	0.0803	0.1710
Y_{stable}	-0.3491	0.2677
Z_{stable}	0.0053	0.0122

Tabelle 4.4: Kenngrößen der gemessenen Beschleunigung ohne Gravitation

4.2.5 Effekte der IMU-Umplatzierung im IMU-Test

Durch die Umplatzierung wie in Unterabschnitt 3.2.5 beschrieben, sollten die Abweichungen durch den Offset der IMU minimiert werden. Mit der Platzierung im Bodyf-

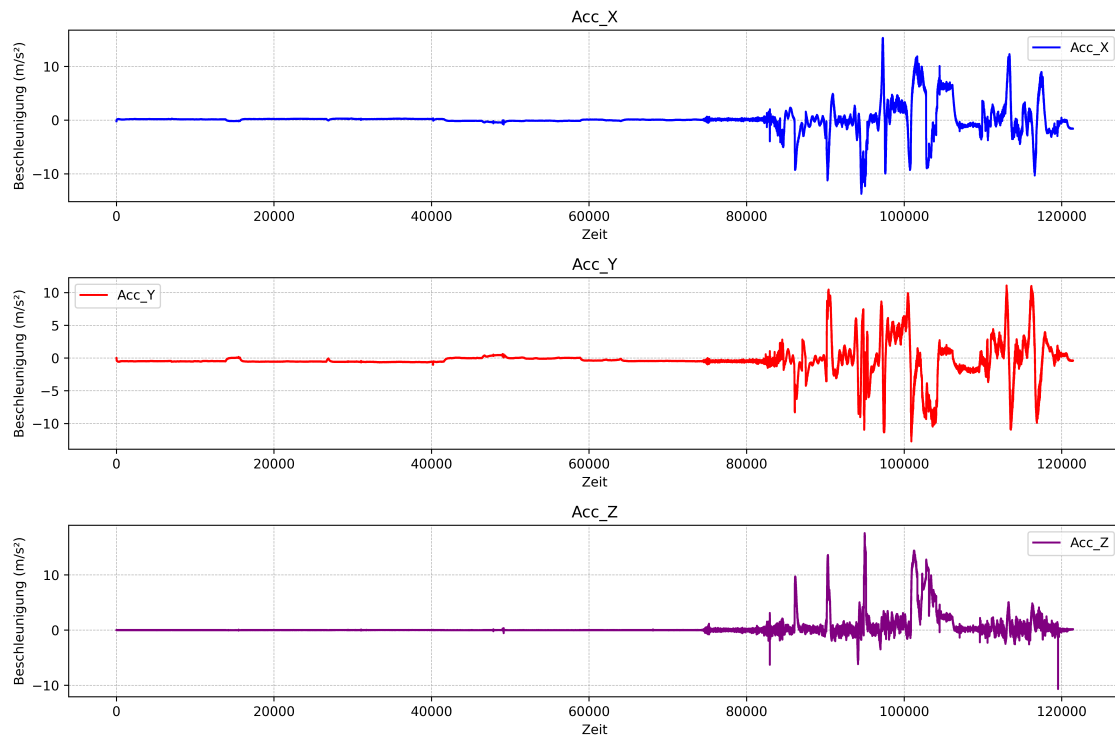


Abbildung 4.5: Beschleunigung des Rahmens in Navigationsrahmen

rame konnte in x- und y-Richtung eine Abweichung vom gerechneten Pivot-Punkt von -0.2 mm respektive 4.3 mm erreicht werden. In z-Richtung sind es -12.7 cm , wobei in dieser Achse physisch keine weitere Verbesserung vorgenommen werden kann.

Beim Kranexperiment aus Abschnitt 3.2.5 konnte leider keine Positionslösung gerechnet werden, da die GNSS-Rohdaten unzureichend waren. Es konnten nur sehr wenige Baselines gerechnet werden, welche für die Update-Steps in zu kleiner Anzahl vorhanden und mit grosser Unsicherheit behaftet waren, wodurch grosse Sprünge in der Orientierung resultieren. Die IMU-hatte einen ähnlichen Drift in Yaw wie bei der Analyse in 3.2.4 festgestellt. Nichtsdestotrotz konnte die Funktionstüchtigkeit der Sensoren bestätigt werden.

4.3 Ergebnisse der Softwareentwicklung & DevOps

In diesem Abschnitt werden die Ergebnisse der Softwareentwicklung und den angewendeten DevOps-Methoden vorgestellt.

4.3.1 Wirksamkeit der Behebung von Azimutwinkel-Orientierungsfehlern im Kalman-Filter

Die Anpassung des Wertebereichs in der Innovationmatrix führte dazu, dass beim Übergang von 180° auf -180° keine Ausreisser mehr berechnet wurden. Dies ist in Abbildung 4.6 zu sehen. Dabei fällt auf, dass in der bisherigen Implementierung nicht jeder Übergang des Winkels über die 180° -Grenze einen Ausreisser zur Folge hat. Dies kann dadurch erklärt werden, dass nur bei den zwei Übergängen, welche Ausreisser zur Folge haben, GNSS-Daten verfügbar waren. Bei den anderen Übergängen war dies nicht der Fall, weshalb keine Innovationmatrix gerechnet wird und keine Probleme entstehen.

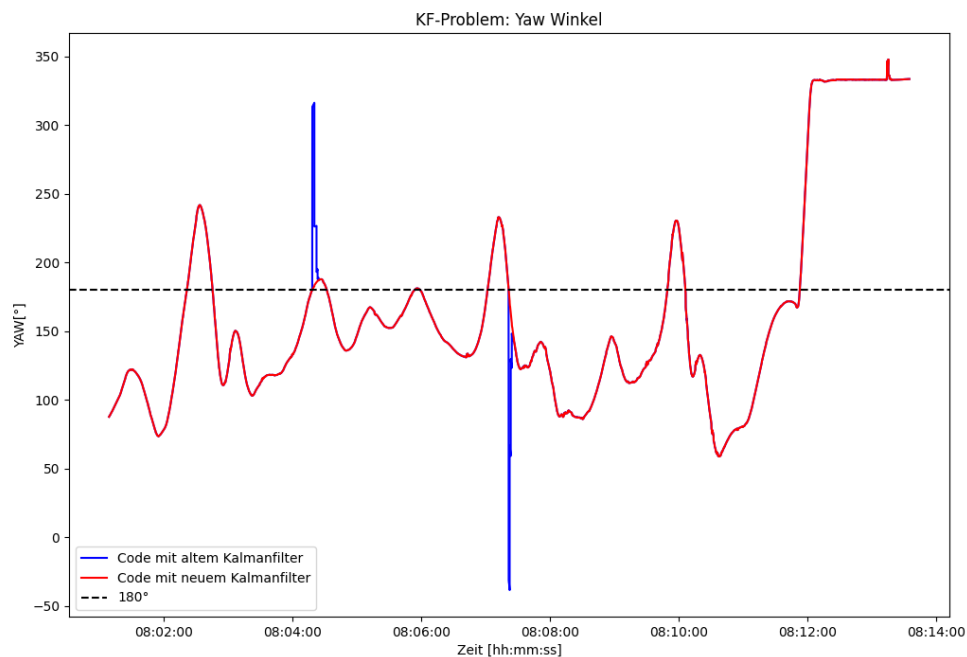


Abbildung 4.6: Vergleich der Yaw-Winkel vor (blau) und nach (rot) der Anpassung des Wertebereichs der Innovationmatrix

4.3.2 Effektivität der erhöhten Systemverfügbarkeit durch Redundanz in GNSS-Antennennutzung

Durch die in Unterabschnitt 3.3.2 beschriebenen Verbesserungen ist die Lageschätzung mit nur 3 verfügbaren GNSS-Antennen möglich. Dies wurde, wie in Kapitel 3 beschrieben, überprüft und in Abbildung 4.7 visualisiert. Im Plot sind dabei die verschiedenen Lösungen mit den verschiedenen Kombinationen mit nur 3 verfügbaren Antennen gezeigt. Dabei fällt auf, dass sich der Pitch-Winkel mit 4 verfügbaren GNSS-Antennen ähnlich verhält wie derjenige mit den Antennen 1, 3 und 4. Bei genauerer Betrachtung der Daten stellte sich heraus, dass Antenne 2 in den ersten 20 Minuten nicht verfügbar war und so nur mit drei Antennen gerechnet wurde. Dies liegt daran, dass die aktuelle Version der RTKLIB-demo5 Software, die für das Post-processing der GNSS-Aufnahmen verwendet wird, erst dann Daten ausgibt, nachdem einmal eine Fix-Lösung erreicht wurde. Die Antenne 2 jedoch erreichte während der ersten 20 Minuten keine Fix-Lösung, weshalb von ihr für den betreffenden Zeitraum keine Daten verfügbar sind. Dies erklärt auch das Verhalten der Lösungen, welche Antenne 2 beinhalten, da diese Lösungen nur auf IMU-Daten beruhen und somit deren Drift ausgesetzt sind.

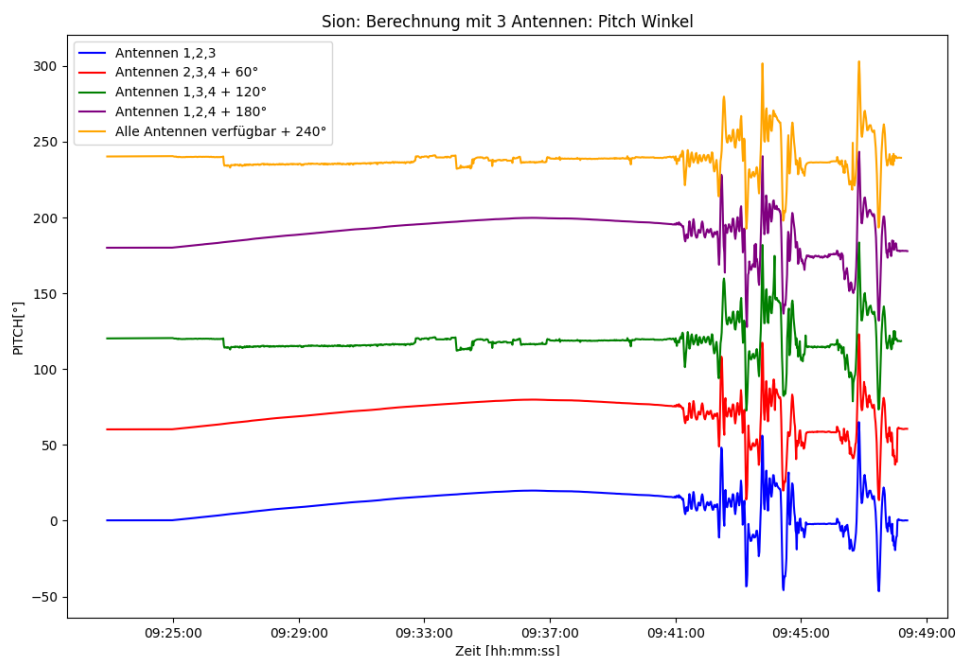


Abbildung 4.7: Pitch-Winkel Lösungen aus verschiedenen Antennenkombinationen mit 60° Versatz

Bei der Untersuchung der verschiedenen Pitch-Winkel fällt auf, dass die verschiedenen Lösungen bei gutem GNSS-Empfang bis auf wenige Grad übereinstimmen. Sobald jedoch der GNSS-Empfang beispielsweise durch Abschattung oder Multipath beeinträchtigt ist, häufen sich die Ausreisser in den Lösungen, welche mit 3 verfügbaren Antennen gerechnet werden. Diese Effekte sind in Abbildung 4.8, welche nur einen Ausschnitt des Zeitraums darstellt, besser sichtbar. Für die anderen Winkel liegt die Genauigkeit bei gutem GNSS-Empfang ebenfalls in einem ähnlichen Bereich von einigen Grad. Die Plots der Auswertung sind in Anhang B.2 in Abbildung B.1 und Abbildung B.2 dargestellt.

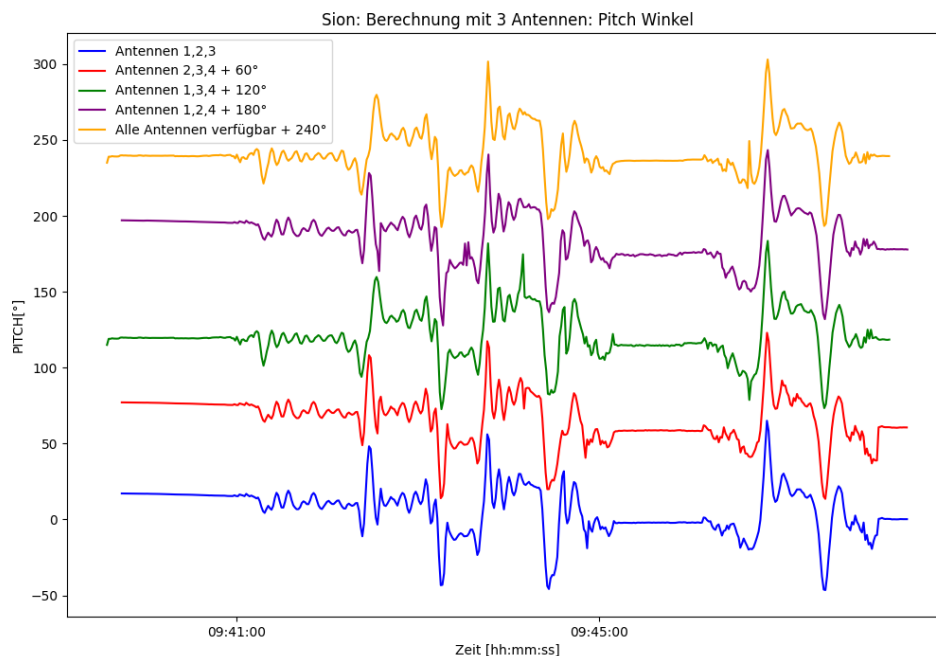


Abbildung 4.8: Vergrößerter Ausschnitt der Pitch-Winkel Lösungen aus verschiedenen Antennenkombinationen mit 60° Versatz

Während dem Flug kann die Neuerung in der Software dazu führen, dass vermehrt GNSS-Lageschätzungen berechnet werden können, auch wenn nicht alle Antennen verfügbar sind. Dies führt dazu, dass mehr Update-Steps durchgeführt werden können und die Lösung nicht nur auf den Daten der IMU basiert.

4.3.3 Effekte der Beschleunigungskorrektur

Mit der Implementierung aus Unterabschnitt 3.3.4 wurden die Auswirkungen der Beschleunigungskorrektur auf die Daten des Anwendungstest in Sion untersucht. Dabei wurde eine maximale Korrektur der Beschleunigung von 0.12 m s^{-2} festgestellt. Bei hohen Drehraten mit starken Beschleunigungen in Steilkurven ist der Einfluss am grössten.

4.3.4 Effekte der Überarbeitung der Transformation ins LV95

Mit der in Unterabschnitt 3.3.5 beschriebenen Überarbeitung konnten folgende Ergebnisse erzielt werden. Die rohen Koordinaten, welche im Navigationsrahmen gegeben sind, werden nun immer ins WGS84 transformiert, von wo sie anschliessend, falls gewünscht, ins LV95 weiter transformiert werden. Dies hat auch den Vorteil, dass die Verarbeitung bei einer falschen Eingabe des Nutzens keine Koordinaten im Navigationssystem ausgibt, welche für die weitere Verarbeitung nicht verwendet werden können.

4.3.5 Entwicklung des *AIR-ETH Control Window*

Das Hauptziel bei der Entwicklung des Kontrollfensters war die Vereinfachung der Datenverarbeitungspipeline für die Verarbeitung. Für die Entwicklung des Kontrollfensters wurde das Python-Framework PyQt5 verwendet. Es enthält Schaltflächen zur Auswahl vordefinierter Optionen, Zeileneingabefelder für Benutzereingaben und eine Anzeige eines Logos.

Was das Design betrifft, so ist das Kontrollfenster mit einem Layout versehen, um die Widgets in einem Raster zu positionieren. Das Logo befindet sich oben, unmittelbar gefolgt von den Widgets (siehe Abbildung 4.9). Ausgewählte Dateien werden hervorgehoben, indem die Farbe der entsprechenden Schaltflächen auf grün geändert wird. Die Benutzereingabe ermöglicht die Auswahl von Dateien wie IMU-, Altimeter-, Bodyframe- und GNSS-Antennendateien.

Im Kontrollfenster steht auch eine IMU-Plotfunktion zur Verfügung. Durch Anklicken der entsprechenden Schaltfläche werden die rohen Beschleunigungsdaten aus der ausgewählten IMU-Datei angezeigt, wodurch die richtigen Indizes der stabilen Periode gefunden werden können.

Vor der Verarbeitung wird geprüft, ob alle notwendigen Parameter korrekt eingegeben wurden. Ist dies der Fall wird die Verarbeitung gestartet.

Für die Verwaltung dieser Parameter wurde eine eigene Klasse *ProcessParams* erstellt. Diese Klasse dient als Container für die ausgewählten Eingabeparameter, wobei jede Instanzvariable einem einzelnen Parameter entspricht. Das Flow-Diagramm in Abbildung 4.9 zeigt den Prozess des Programmablaufs. Eine genauer Beschreibung der Klasse findet sich in Anhang C.9.6.

Zusammenfassend bietet die Implementierung des Kontrollfensters eine robuste und benutzerfreundliche Schnittstelle für die Datenverarbeitungspipeline des AIR-ETH-Projekts.

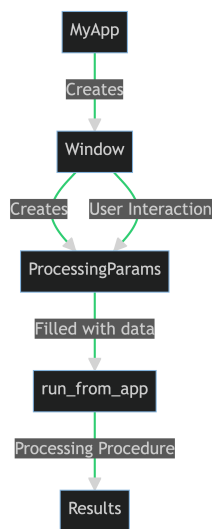


Abbildung 4.9: Fluss-Diagramm des Control Windows

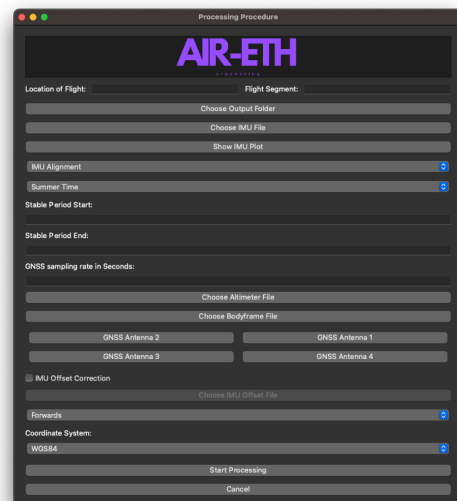


Abbildung 4.10: Control Window

4.3.6 Verwendung von Git für zentrale Versionskontrolle und Dokumentation

Wie in Abschnitt 3.3.7 besprochen, wurde ein Git-Repository erstellt. Zugriff haben die VAW via Raphael Moser, vom IGP sind Gregor Möller, Benedikt Soja und Alexander Wolf ermächtigt.

Im Repository findet sich nun der aktuellste, versionierte Code aller Bestandteile. Die Dokumentation ist als Gitlab-Wiki gepflegt. Der Stand vom 13.06.2023 ist im Anhang C in dieser Arbeit angehängt. Weiter sind dort offene Issues getrackt, die in Zukunft bearbeitet werden können.

Kapitel 5

Diskussion

In diesem Kapitel werden die Ergebnisse der experimentellen Validierung des Systems und der Softwareentwicklung diskutiert. Dabei werden verschiedene Aspekte wie die Leistung des GNSS-Systems, das IMU-Verhalten und das Software-Setup analysiert. Es wird aufgezeigt, wie die gewonnenen Erkenntnisse zur Verbesserung des Systems beigetragen haben. Insbesondere werden die Effektivität der Behebung von Azimutwinkel-Orientierungsfehlern im Kalman-Filter, die Systemverfügbarkeit durch Redundanz in der GNSS-Antennennutzung sowie die Verbesserungen in der Software und im DevOps-Bereich besprochen.

5.1 Experimentelle Validierung des Systems

Das Experiment am Turmdrehkran der ETH Höggerberg zeigte, wie bereits in Unterabschnitt 3.2.1 erläutert, dass einige der GNSS-Logger keine Fix-Lösung erreichen konnten. Bei der Analyse der Ursache wurde dabei herausgefunden, dass dieses Problem nur mit den neuen *ublox ANN-MB01* auftrat. Daher wurde nach möglichen Gründen für dieses Verhalten gesucht. Auffällig war, dass sich die verschiedenen Logger unterschiedlich verhielten, obwohl es sich um baugleiche, industriell gefertigte Logger handelt. Daher wurde die These aufgestellt, dass die Antennen gerade so viel Strom benötigen, wie die Logger während des Hochfahrens maximal zur Verfügung stellen können. Aufgrund geringfügiger Unterschiede in der maximalen Leistung der Logger liegt der Strombedarf jedoch bei zwei der vier Logger genau über der maximalen Leistung. Daher wird bei den Loggern mit geringerer maximaler Leistung angenommen, dass die Antenne einen Defekt aufweist, weshalb sie nicht

weiter mit Strom versorgt wird. Wird die Antenne jedoch erst nach dem Hochfahren aller Loggersysteme angeschlossen, ist die maximale Leistung gerade ausreichend, so dass die Antenne mit genügend Strom versorgt werden kann. Diese Erklärung ist jedoch nur eine Vermutung, die nicht weiter untersucht wurde. Wir empfehlen daher, diese Zusammenhänge genauer zu analysieren, um mögliche Probleme bei weiteren Experimenten zu verhindern und damit die Funktionsfähigkeit des Rahmens zu gewährleisten.

Bezüglich des dynamischen Verhaltens des Rahmens in der Luft ist zu erwähnen, dass die Ausrichtung in Flugrichtung in der Luft bereits sehr gut funktioniert. Daher sind in diesem Punkt derzeit keine weiteren Verbesserungsmaßnahmen notwendig. Ein grosser Fortschritt ist auch die Gewichtsreduktion, die durch den von der VAW neu entwickelten GFK-Rahmen erreicht werden konnte. Dadurch kann der Rahmen besser transportiert werden und man kommt dem Ziel eines bodengestützten, von einer Person gezogenen Radarsystems einen Schritt näher. Zusammenfassend kann festgestellt werden, dass zwar keine verwertbaren Daten aus dem Versuch gewonnen werden konnten, der Versuch aber sehr wertvolle Ergebnisse lieferte, da viele Schwachstellen durch die Durchführung entdeckt und behoben werden konnten.

Beim Testflug im Unterwallis konnte, wie in Unterabschnitt 4.2.3 beschrieben, bestätigt werden, dass der neu entwickelte Rahmen auch den Belastungen unter dem Helikopter standhält. Auch die hohen Geschwindigkeiten, die dieser Rahmen ermöglicht, sind ein grosser Vorteil, da das Messgebiet so in kurzer Zeit erreicht werden kann. Bei hohen Geschwindigkeiten werden jedoch die Daten des Altimeters aufgrund der Neigung des Rahmens sehr unzuverlässig. Dies stellt eine weitere Optimierungsmöglichkeit dar, da der gleiche Effekt auch bei geringeren Geschwindigkeiten auftritt und somit die Qualität der Ergebnisse beeinträchtigt. Es könnte daher eine Altimeter-Optimierung entwickelt werden, die entweder rechnerische Korrekturen vornimmt oder aufgrund der Fixierung immer vertikal nach unten misst. Ausserdem sollten die Seitenflossen des bestehenden Rahmens verstärkt werden, damit sie dem Abwind des Helikopters standhalten können. Die GNSS-Logger haben im Versuch zuverlässig aufgezeichnet und lieferten Daten, die gut ausgewertet werden konnten. Grundlage dafür ist jedoch eine Referenzstation, die für zukünftige Messkampagnen immer mitgeführt und aufgebaut werden sollte. Sie ist für eine genaue Positionierung und Lagebestimmung unerlässlich. Der Versuch mit verschiedenen Seillängen zeigte, dass im geflogenen Szenario keine signifikanten Unterschiede zwischen 20 m und 40 m festgestellt werden konnten. Da jedoch bei einer angenäherten Grundfläche des Hubschraubers von 3×3 m ein Winkel von ca. 10° bzw. 5° durch den Hubschrauber

verdeckt wird, ist dies nicht in jeder Umgebung zu erwarten. In Tälern, wo die Verdeckung durch Berge bereits gross ist, kann die Seillänge einen Unterschied machen. Dieser Effekt sollte daher genauer untersucht werden.

5.1.1 Wichtigkeit der GNSS-Daten

Um eine gute Positionslösung zu erhalten, müssen die GNSS-Daten von hoher Qualität sein. Dafür muss, wie in Abschnitt 4.2.4 beschrieben, eine für die Empfänger und Antennen optimierte Verarbeitungsbibliothek verwendet werden. Treten Probleme auf, müssen die Rohdaten untersucht werden (CNR, MP, Visibility) und die Parameter in RTKPOST an die Umstände angepasst werden.

5.1.2 IMU-Drift

Das IMU-Alignment wurde in Abschnitt 4.2.4 untersucht, wobei ein IMU-Drift festgestellt wurde. Die Grössenordnung mit 3° min^{-1} ist deutlich über der Spezifikation der *In-run-bias-stability* des Herstellers von 8° h^{-1} (Movella, 2020). Dies könnte an verschiedenen Änderungen liegen. Einerseits ist die IMU neu im Koffer statt wie früher auf dem Koffer. Dabei ist sie direkt neben der GPR-Steuereinheit platziert. Das Metall oder elektromagnetische Strahlung aus der Elektronik könnten den Magnetometer beeinflussen. Da der Alignment-Algorithmus des Herstellers verwendet wird, welcher den Magnetometer zur Fixierung der Orientierung nutzt, könnte dies zum beobachteten Yaw-Drift führen. Es konnte nicht untersucht werden, ob die IMU ausserhalb des Koffers nicht mehr mit diesem Drift behaftet ist. Daher empfiehlt es sich, eine Kalibrierung wie in Movella (2022) beschrieben durchzuführen, um diesen Drift ggf. zu beseitigen. Dies könnte zum gestörten Alignment-Algorithmus (Abschnitt 4.2.4) führen. Durch die Gewichtung der Update-Steps im Kalman-Filter hat dieser Effekt keinen allzu grossen Einfluss auf die Navigationslösung. Andererseits führt dies zu einer Abhängigkeit von genügend GNSS-Beobachtungen, was zu Problemen bei ungünstigen Verhältnissen für GNSS führen kann.

5.1.3 IMU-Alignment

Die Beschleunigungen sind nach der Differenz mit dem Gravitationsvektor noch deutlich zu hoch für eine gute Integration der Beschleunigungen zur Positionsbestimmung. Das System funktioniert trotzdem, da Lardon (2022a) die Update-Steps mit hoher Gewichtung durchführt und die GNSS-Baselines mit 5 Hz oder mehr gerechnet werden. Um eine gute Sensorfusion auch bei längeren GNSS-Ausfällen zu er-

reichen, muss das Alignment verbessert werden. Wegen der Nutzung des Alignment-Algorithmus des Herstellers, muss regelmässig eine Kalibration durchgeführt oder die Störeinflüsse minimiert werden. Ein weiterer Ansatz wäre, das Alignment während der Verarbeitung zu rechnen, und so die äusseren Einflüsse unter eigener Kontrolle zu haben. Es konnte im Rahmen dieser Arbeit nicht nachvollzogen werden, wie die Werte der Beschleunigungssensoren mit dem Drift des Yaw-Winkels zusammenhängen. Dies sollte genauer analysiert werden.

5.2 Softwareentwicklung

In diesem Kapitel werden die verschiedenen Verbesserungen und deren Auswirkungen auf die Leistung des Systems diskutiert. Zunächst wird die Effektivität der Behebung von Azimutwinkel-Orientierungsfehlern im Kalman-Filter betrachtet. Es wird erläutert, wie das Problem der Ausreisser durch Anpassung des Wertebereichs gelöst wurde und welche Auswirkungen dies auf die Flugplanung hat. Zudem wird die Systemverfügbarkeit durch Redundanz in der GNSS-Antennennutzung analysiert und es wird erläutert, wie diese Implementierung kurze Antennenausfälle überbrücken kann. Schliesslich werden die Verbesserungen in der Software und im DevOps-Bereich betrachtet, darunter die IMU-Beschleunigungskorrektur, das Control Window und das zentrale Repository. Diese Verbesserungen tragen zur Steigerung der Effizienz, Fehlerreduktion und vereinfachten Weiterentwicklung des Systems bei.

5.2.1 Wirkung der Änderung im Kalman-Filter

Die in Unterabschnitt 4.3.1 beschriebenen Ergebnisse zeigen, dass das Problem der Ausreisser durch zu grosse Einträge in der Innovationsmatrix gelöst wurde. Dies konnte durch eine simple Anpassung des Wertebereichs erreicht werden. Obwohl ein Lösungsansatz mit Einheitsquaternionen wahrscheinlich eleganter gewesen wäre, ist der bestehende Ansatz für diese Anwendung ausreichend. Dies, da die ursprünglichen Ausreisser Folge von leicht unterschiedlichen Messungen des Yaw-Winkels der IMU und der GNSS-Lösung sind. Da dieser kleine Unterschied nur beim 180°-Übergang von Bedeutung ist, ist es ausreichend, den in Unterabschnitt 3.3.1 beschriebenen Ansatz durchzuführen. Damit dieser Ansatz nicht mehr zufriedenstellend wäre, müssten sich die Werte aus den IMU- und den GNSS-Daten für denselben Zeitpunkt fundamental unterscheiden. Wenn dies der Fall ist, liegt jedoch ein anderes, weitaus grösseres Problem bei der Lagebestimmung oder dem Zeitabgleich der beiden Messungen vor, welches für eine funktionierende Messung behoben werden muss. Daher

wurde beschlossen, die anschaulichere Methode zu wählen. Ausserdem wurde für diesen Fall nur der Yaw-Winkel korrigiert, obwohl das gleiche Problem auch für Roll und Pitch auftreten könnte. Aufgrund der Anwendung als Unterlast unter einem Hubschrauber wurde dies jedoch als unrealistisches Szenario erachtet, weshalb diese Korrekturen nicht implementiert wurden.

Die oben beschriebenen Verbesserungen und die daraus resultierenden Ergebnisse wirken sich auch auf die Flugplanung aus. So können nun auch Flüge in Nord-Süd-Richtung, bei denen der Rahmen oft den 180° Übergang durchläuft, geplant werden, ohne weitere Probleme bei der Auswertung befürchten zu müssen. Dies ist ein grosser Vorteil, da das zu vermessende Gebiet auf dem effizientesten Weg überflogen werden kann und somit die Kosten durch Verkürzung der Flugzeit reduziert werden können. Ausserdem kann durch die Anpassung des Kalman-Filters mit einer höheren GNSS-Abtastrate geflogen werden, da diese Daten in der Verarbeitung zu zusätzlichen Aktualisierungsschritten und nicht zu mehr Ausreissern führen.

5.2.2 Redundanz in GNSS-Antennennutzung

Die in Unterabschnitt 4.3.2 dargestellten Ergebnisse zeigen, dass mehr Ergebnisse aus der GNSS-Lageschätzung zur Verfügung stehen. Dies liegt daran, dass nun auch Ergebnisse aus nur drei verfügbaren Antennen gerechnet werden. Diese Ergebnisse unterscheiden sich bei gutem GNSS-Empfang nur um wenige Grad von der Lösung, die mit vier Antennen berechnet wurde. Daher ist diese Implementierung sehr gut geeignet, um kurze Antennenausfälle zu überbrücken. Für längere Ausfälle oder für eine Lösung mit nur drei Antennen ist die derzeit vorhandene Implementierung jedoch nicht ausgelegt, da insbesondere bei schlechtem GNSS-Empfang schnell grosse Ausreisser auftreten. Dies kann auf die fehlende Redundanz in der Lageschätzung und durch die geringere Genauigkeit der GNSS-Positionen bei schlechtem Empfang zurückgeführt werden. Diese Ausreisser könnten in einem weiterführenden Projekt identifiziert und entfernt werden, so dass auch längere Ausfälle ohne Ausreisser überbrückt werden könnten. Die Redundanz sollte jedoch unbedingt beibehalten werden, um einerseits sicherzustellen, dass die Daten auch bei Ausfall eines Loggers verarbeitet werden können. Andererseits hilft die Redundanz auch, die Genauigkeitsanforderungen bei schlechtem GNSS-Empfang zu gewährleisten. In diesem Zusammenhang ist es wichtig anzumerken, dass die Verarbeitung mindestens eine Minute an Daten benötigt.

Durch die erhöhte Verfügbarkeit der GNSS-Lageschätzungen werden im Kalman-Filter häufiger Update-Steps durchgeführt, wodurch die Ergebnisse genauer werden und nur in kürzeren Intervallen das Fehlerverhalten der IMU aufweisen. Dies gewährleistet eine genauere Positionierung und Bestimmung der Lage des Rahmens.

5.2.3 Verbesserungen in Software und DevOps

Die IMU-Beschleunigungskorrektur anhand des Offsets bietet eine gute Grundlage für eine höhere Gewichtung der IMU bei besserem Alignment und vor allem die Möglichkeit in Zukunft auch Rahmen zu bauen, bei denen die IMU aus baulichen Gründen weiter vom Drehpunkt abweicht.

Auch führt das *Control Window* zu einer deutlichen Vereinfachung der Bedienung und zusätzlich zu einer geringeren Fehleranfälligkeit, da es ein visuelles Feedback und eine Kontrolle der Eingaben gibt. Mit dem alten Vorgehen brach die Verarbeitung jeweils bei nicht zulässigen Eingabeparameter ab. Nun werden zu Beginn alle Eingaben geprüft. Dies vereinfacht einerseits das Testen, zudem ist die Verarbeitungs-Pipeline für eine Automatisierung vorbereitet.

Das zentrale Repository ermöglicht eine kürzere Einarbeitungszeit und vor allem versionierten Code, wodurch zukünftige Weiterentwicklungen vereinfacht werden. Durch die bereitgestellte Dokumentation kann ein zeitunabhängiger Informationsaustausch erfolgen.

Kapitel 6

Fazit und Ausblick

In diesem Kapitel wird ein Fazit aus den Ergebnissen gezogen und ein Ausblick auf zukünftige Arbeiten gegeben. Es wird festgestellt, dass durch die vorgenommenen Softwareverbesserungen und die Überarbeitung des Rahmens wichtige Fortschritte erzielt wurden. Im Ausblick werden verschiedene Aspekte der Sensorfusion identifiziert, die weiter verbessert und optimiert werden könnten.

6.1 Fazit

Dieses Kapitel gibt einen Überblick über die experimentelle Validierung des Systems und den Fortschritt der Softwareentwicklung im Rahmen dieser Arbeit.

Im Bereich der **Softwareentwicklung** wurden Verbesserungen vorgenommen, um Anomalien im Kalman-Filter zu beheben, indem der Yaw-Winkelbereich geöffnet wurde, so dass auch bei einem 180°-Übergang des Systems ein zuverlässiger Betrieb gewährleistet ist. Dies verbessert die Positionsauflösung und verkürzt die Konvergenzzeit bei einer grossen Anzahl von Aktualisierungsschritten.

Durch die Implementierung einer Drei-Antennen-Lösung können kurze Antennen-ausfälle überbrückt und die GNSS-Lageschätzungen erhöht werden. Dies ermöglicht häufigere Aktualisierungsschritte des Kalman-Filters, was zu einer besseren Positionsbestimmung und Ausrichtung des Rahmens führt.

Schliesslich haben Software- und DevOps-Verbesserungen wie die Beschleunigungskorrektur und das Hinzufügen eines Kontrollfensters die Zuverlässigkeit und Benutzerfreundlichkeit des Systems erheblich verbessert. Das zentrale Repository hat

den Entwicklungsprozess rationalisiert, was zu einer schnelleren Einarbeitung und einfacheren zukünftigen Weiterentwicklungen führt. Die zentrale Dokumentation ermöglicht eine selbstständige Einarbeitung und Anwendung des Systems.

Auf der **experimentellen Seite** wurde festgestellt, dass einige GNSS-Logger nicht in der Lage sind, eine Fix-Lösung zu erzielen, wobei Diskrepanzen zwischen Loggern desselben Herstellers festgestellt wurden. Dieses Problem wurde durch einen Rollback auf ältere Antennen gelöst, da der Workaround zu einer zusätzlichen Komplexität bei der Inbetriebnahme geführt hätte.

Das dynamische Verhalten des Rahmens in der Luft, insbesondere seine Ausrichtung in Flugrichtung, erwies sich als sehr gut, so dass keine unmittelbaren Verbesserungen erforderlich waren. Ein wesentlicher Fortschritt war die Gewichtsreduzierung des neuen, von VAW entwickelten GFK-Rahmens. Dadurch wurde eine bessere Transportfähigkeit erreicht.

Bei den Testflügen im Unterwallis hielt der von der VAW neu entwickelte Rahmen den Flugbelastungen stand und ermöglichte den Betrieb des Helikopters bei hohen Geschwindigkeiten. Allerdings gab es Probleme mit der Neigung des Rahmens bei hohen Geschwindigkeiten, was zu unzuverlässigen Höhenmessungen führte. Dies ist vor allem während des Reisefluges relevant und könnte ein Sicherheitsrisiko darstellen, falls diese Höhen zur Navigation verwendet werden.

Zusammenfassend lässt sich sagen, dass trotz der noch offenen Punkte sowohl bei der Systemintegration des neuen Rahmens als auch bei den Softwareverbesserungen wichtige Fortschritte erzielt wurden, die eine bessere Performance im zukünftigen Betrieb erzielen.

6.2 Ausblick

Im Zuge zukünftiger Arbeiten könnten mehrere Aspekte der Sensorfusion weiter verbessert und optimiert werden. Ein Fokus sollte dabei auf der Untersuchung und Optimierung der Gewichtung innerhalb des Kalman-Filters liegen. Hier relevant ist auch eine eingehende Prüfung und mögliche Anpassung des Alignmentverfahrens, um eine zuverlässigere Datenverarbeitung zu gewährleisten.

Eine weitere Entwicklung könnte die Fertigstellung einer kombinierten Vorwärts-/Rückwärtsverarbeitung darstellen, die bereits in dieser Arbeit begonnen wurde. Bei deren Weiterentwicklung ist es wichtig, die Genauigkeit der Schätzung der Rah-

menlage zu überprüfen und entsprechend bei jedem Update-Step zu gewichten. Dadurch könnten potenzielle Ausreisser in der Lage erkannt und eliminiert werden, was letztendlich die Genauigkeit des Systems verbessert.

Zudem sollte der zugrundeliegende Code überarbeitet werden. Durch eine PEP8-konforme (*Python Enhancement Proposal: 8, Style Guide for Python Code*) Schreibweise und eine sorgfältige Bereinigung des Codes kann die Lesbarkeit erhöht und die Einarbeitungszeit verkürzt werden. Eine kontinuierlich aktualisierte Dokumentation, die alle Weiterentwicklungen berücksichtigt, ist dabei unerlässlich.

Die Beurteilung des Nutzens neuer Antennen könnte durch eine Untersuchung zur Qualitätsverbesserung der Navigationslösung erfolgen. Eine Integration der Steuerungssoftware der Logger mit den anderen Sensoren könnte ausserdem die Datenaufzeichnung erheblich vereinfachen, indem ein *one-click Start* ermöglicht wird.

Literaturverzeichnis

- Brockmann, E., Ineichen, D., and Wiget, A. (2005). Neumessung und Auswertung des GPS-Landesnetzes der Schweiz LV95. Publisher: SIGImedia AG.
- Carver, J., Hong, N., and Thiruvathukal, G. (2016). *Software Engineering for Science*. Chapman & Hall/CRC Computational Science. CRC Press.
- Dyck, A., Penners, R., and Lichter, H. (2015). Towards Definitions for Release Engineering and DevOps. In *2015 IEEE/ACM 3rd International Workshop on Release Engineering*, pages 3–3.
- Gainer, T., Hoffman, S., and Center, L. R. (1972). *Summary of Transformation Equations and Equations of Motion Used in Free-flight and Wind-tunnel Data Reduction and Analysis*. NASA.
- Kresse, W. and Danko, D., editors (2022). *Springer handbook of geographic information*. Springer handbooks. Springer, Cham, 2nd edition edition.
- Lardon, F. (2022a). Development of a georeferencing system for a helicopter-borne radar system. Master’s thesis, ETH Zurich, Zurich.
- Lardon, F. (2022b). Development of a low-cost system for direct georeferencing of a helicopter-borne radar system. Place: Zurich Type: Student Paper.
- Montenbruck, O. and Teunissen, P., editors (2017). *Springer Handbook of Global Navigation Satellite Systems*. Springer Handbooks. Springer International Publishing : Imprint: Springer, Cham, 1st ed. 2017 edition.
- Movella (2022). Reasons why the heading/yaw is not stable or incorrect. Publication Title: Xsens knowledge base.
- Movella, X. (2020). MTi Family Reference Manual. Publication Title: Document MT1600P.
- Munguía, R. (2014). A GPS-aided inertial navigation system in direct configuration. *Journal of applied research and technology*, 12(4):803–814. Publisher: Universidad Nacional Autónoma de México, Instituto de Ciencias Aplicadas y . . .
- Piringer, F. (2018). *Schätzung von Flugparametern mit GNSS-Beobachtungen*. Bachelor thesis, ETH Zurich.
- Takasu, T. (2021). RTKLIB Manual: Demo5 version. Technical report.

Winters, T., Manshreck, T., and Wright, H. (2020). *Software Engineering at Google: Lessons Learned from Programming Over Time*. O'Reilly Media.

Anhang A

Arbeitsaufteilung

Die Arbeit wurde als Gruppenarbeit geschrieben. Daher wurden einige Teile gemeinsam erarbeitet. Dies betrifft insbesondere die Einleitung, die Bodyframe-Vermessung, Teile der experimentellen Validierung, die Diskussion und das Fazit (sofern weiter unten nicht anders vermerkt).

Benedikt Pohl hat des weiteren die folgenden Kapitel selbstständig erarbeitet:

- Integration von Plausibilitätsprüfungen in die Datenverarbeitung (3.1.3)
- Analyse nach Anwendungstest (3.2.4) (4.2.4)
- IMU Offset (3.2.5), (4.2.5)
- Implementierung der Korrektur von IMU-Offset (3.3.4), (4.3.3)
- Kontrollfenster für Prozessierung (3.3.6), (4.3.5)
- Implementierung von DevOps-Prinzipien (3.3.7), (4.3.6)

Fadri Rischatsch erarbeitete folgende Kapitel selbstständig:

- Theorie (2)
- Behebung von Azimutwinkel-Orientierungsfehlern im Kalman-Filter (3.3.1),(4.3.1)
- Erhöhung der Systemverfügbarkeit durch Redundanz in GNSS-Antennennutzung (3.3.2),(4.3.2)
- Implementierung der rückwärtslaufenden Prozessierung (3.3.3)
- Überarbeitung der Transformation ins LV95 (3.3.5), (4.3.4)
- Analyse von Hardware-Problemen mit ublox ANN-MB01 und ZED-F9P (3.2.2), (4.2.2)

Anhang B

Weitere Formeln und Plots

B.1 Weitere Formeln

$$\begin{aligned}t^{\mathbf{q}.yaw} &= \arctan\left(\frac{2q_1q_2 + 2q_0q_3}{2q_0^2 + 2q_1^2 - 1}\right) \\t^{\mathbf{q}.pitch} &= \arcsin(2q_0q_2 - 2q_1q_3) \\t^{\mathbf{q}.roll} &= \arctan\left(\frac{2q_2q_3 + 2q_0q_1}{2q_0^2 + 2q_3^2 - 1}\right)\end{aligned}\tag{B.1}$$

B.2 Weitere Plots

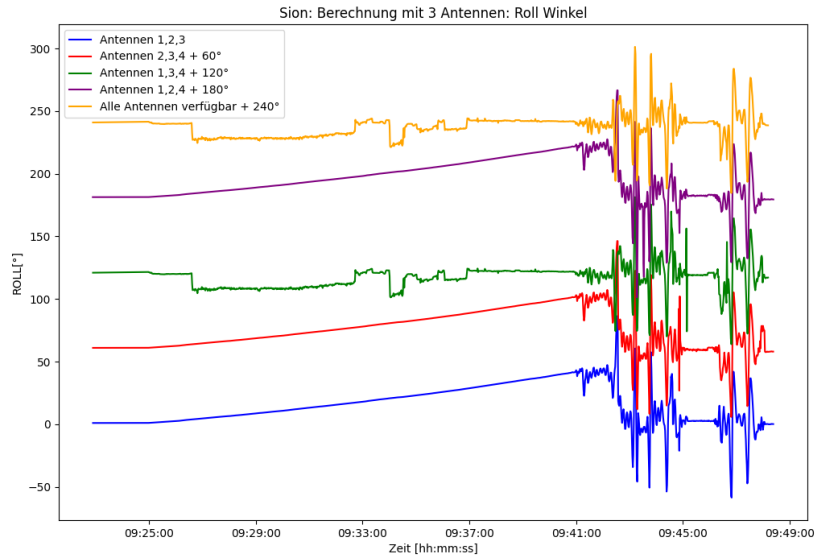


Abbildung B.1: Yaw-Winkel Lösungen aus verschiedenen Antennenkombinationen mit 60° Versatz

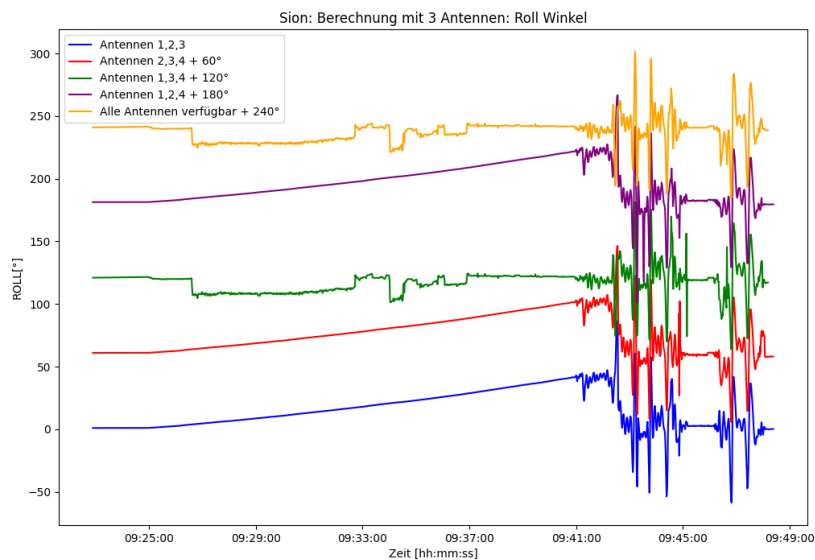


Abbildung B.2: Roll-Winkel Lösungen aus verschiedenen Antennenkombinationen mit 60° Versatz

Anhang C

Dokumentation

C.1 Home

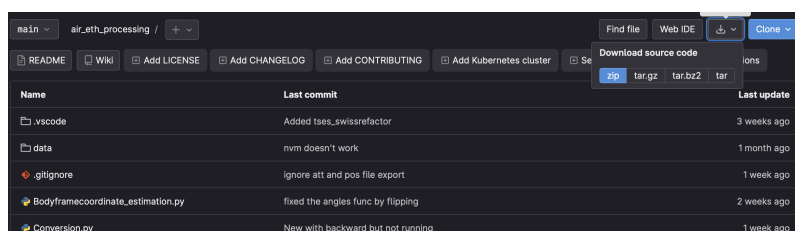
Welcome to the AIR-ETH Processing Wiki! This guide will help you set up and use the AIR-ETH processing library.

C.1.1 Prerequisites

1. Install Python 3.11 from the official Python website (be sure to add Python to PATH and disable path length limit, *this only matters to win users*).
2. Download and install the latest RTKLIB demo5 release for GNSS data processing capabilities. Download the demo5_xx.zip file and unpack to your desired location to install.

C.1.2 Installation

1. Download the repository:



2. In your command-line shell (cmd, PowerShell, bash) navigate to the `air_eth_processing` directory and run the following command to install the required modules:


```
pip install -r requirements.txt
```

C.1.3 Usage

Follow the processing procedure to process your data. If you want to have an update, either download it from the repository again, or use git to pull changes when there are some.

We recommend Visual Studio Code IDE that offers integrated Git functionality. It provides a seamless Git experience with features like staging changes, committing, and pushing directly from the editor.

You can download Visual Studio Code from the official website: [Download Visual Studio Code](#)

Once you have Visual Studio Code installed, you can easily manage Git repositories by using the built-in Git integration or installing Git-related extensions from the Visual Studio Code marketplace.

C.2 Bodyframe Determination

This is the script to get your coordinates in the Bodyframe.

C.2.1 Input

The input is a CSV-File, with the Header " Pointname, East, North, Height "

Important: due to the given code from Fabrice Points must be in reverse order (so the order is 4,3,2,1)

There have to be 2 different CSV-Files, one with Point names and another with numbers only instead of names.

C.2.2 Processing

The Code converts the coordinates from the TS (E,N,U) system in a lefthanded E,N,D system, which is then translated to the frame's origin and further rotated into the body-frame **with y,x,z**. These coordinates are finally written in `ublox_bodyframe_coords.txt`.

C.2.3 Additional information

Fabrice calculated something with the ublox.txt files. I didn't manage to find out, what he did. However, the code now runs and should be correct. The original code from Fabrice is still as comment in the script.

The tt 1-6 values are the length of different sides of the frame. It is compared if the different sides are similar. If the deviation is more than about 5cm it is likely that the frame was not set up correctly. **the tt values say nothing about the accuracy of the measurement itself**

In the original case, the IMU was placed in the origin after the measurement. The actual version of the code is only to get the coordinates of the antennas in the body frame.

C.2.4 Bodyframe IMU origin offset

The script calculates the origin offset of the AIRETH system with respect to the navigation frame by finding the average of the two diagonal points of the AIRETH system, which is taken to be the origin of the body frame. The offset of the IMU in the navigation frame is then calculated by subtracting the origin of the body frame from the IMU coordinates.

The offset CSV file contains the offset of the IMU in the body frame after rotation. It is a single row file with three values representing the displacement of the IMU in the y, x, and z directions. The header of the file indicates the order of the values. This documentation offers a brief manual on how to determine the body frame of the AIR-ETH ground penetrating radar system with the aid of a total station and a prism. Prior (basic) knowledge of surveying is assumed for comprehending and following these instructions.

C.3 Material list

- Frame** AIR-ETH
- Antenna mounts with **antennas** from AIR-ETH
- Total station** (for example Leica TS60 from GSEG) + **tripod**

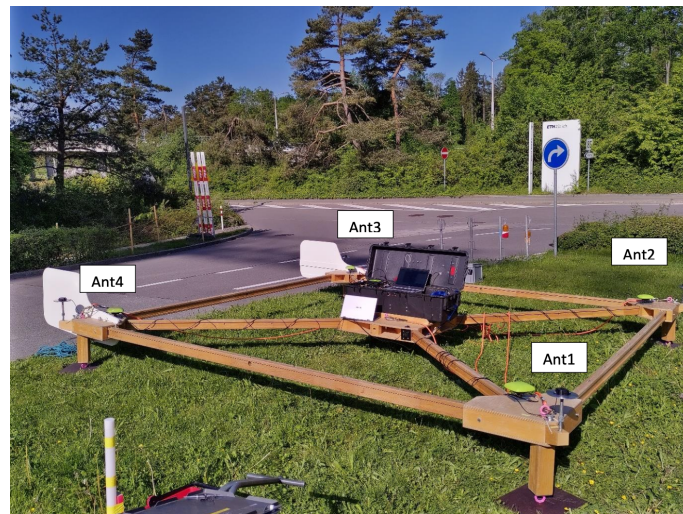


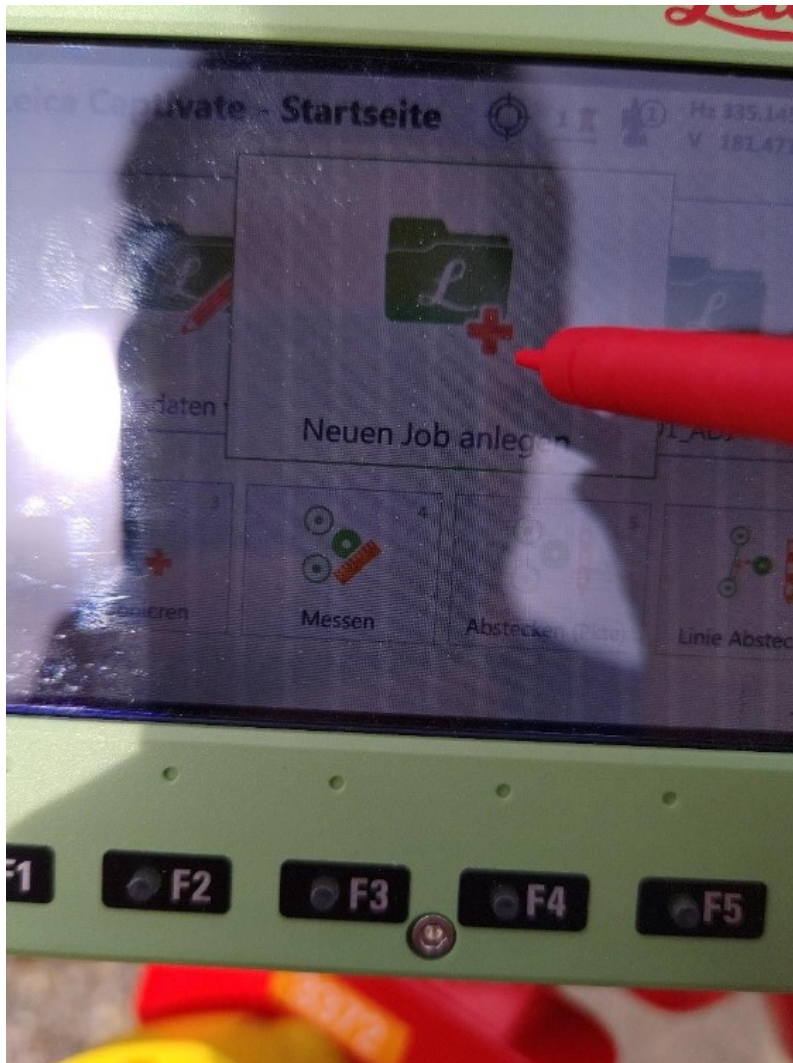
Abbildung C.1: Overview

- Miniprisma** with round level (DE: Miniprisma mit Absteckspitze und Dosenlibelle)
- Meterstick**

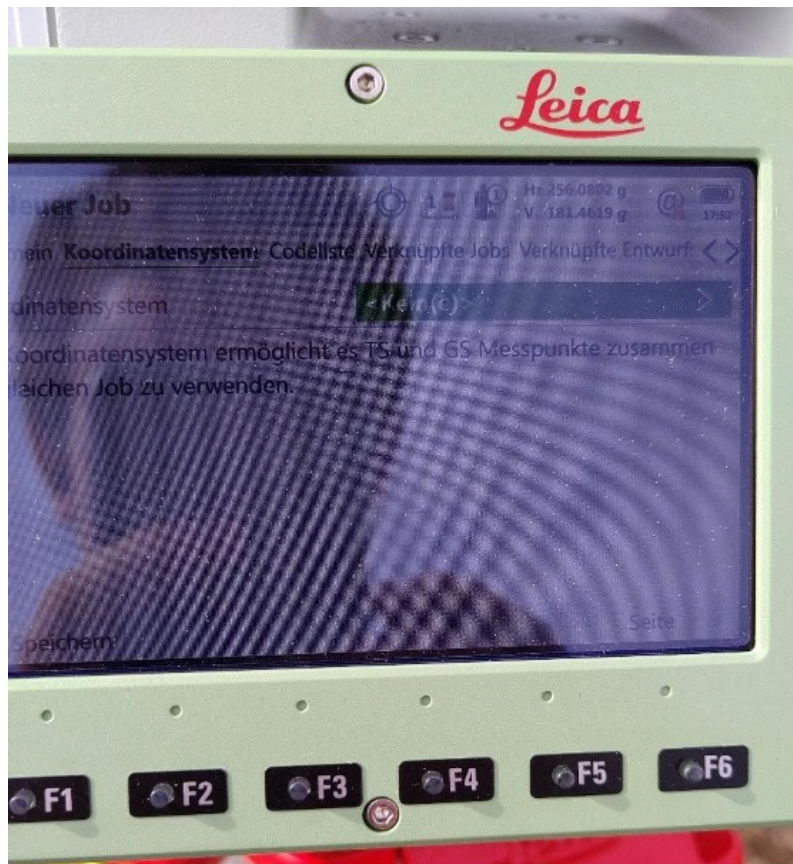
C.4 Measuring the baselines

This documentation outlines the procedure using a Leica total station. Other total stations will have comparable settings but a different UI.

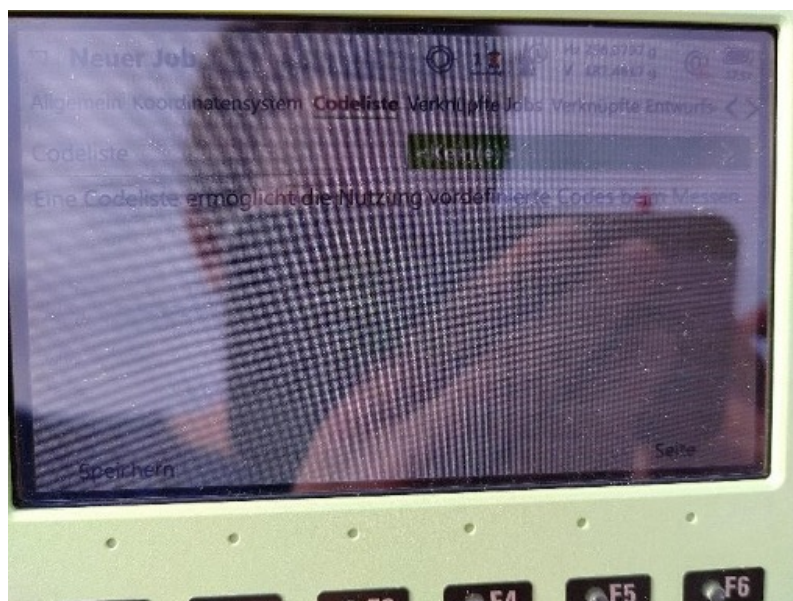
1. Create a new job in TS60



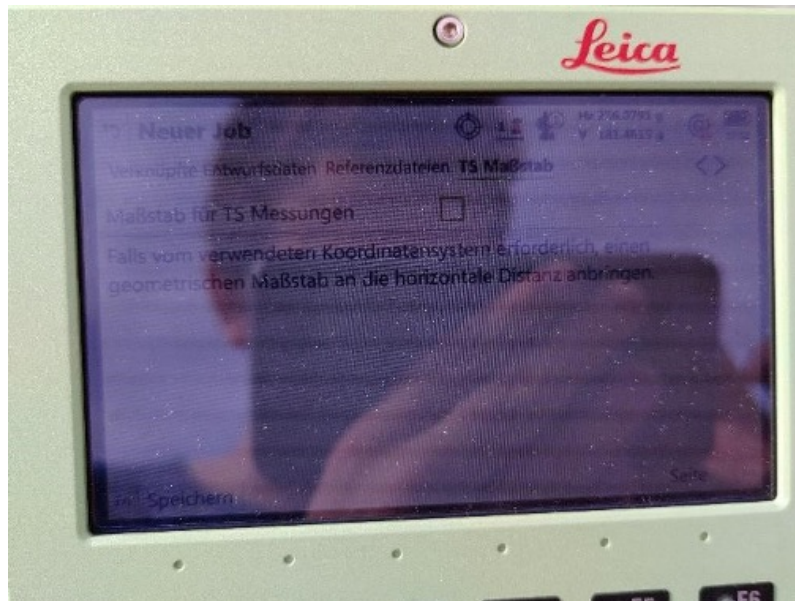
2. Select no coordinate system



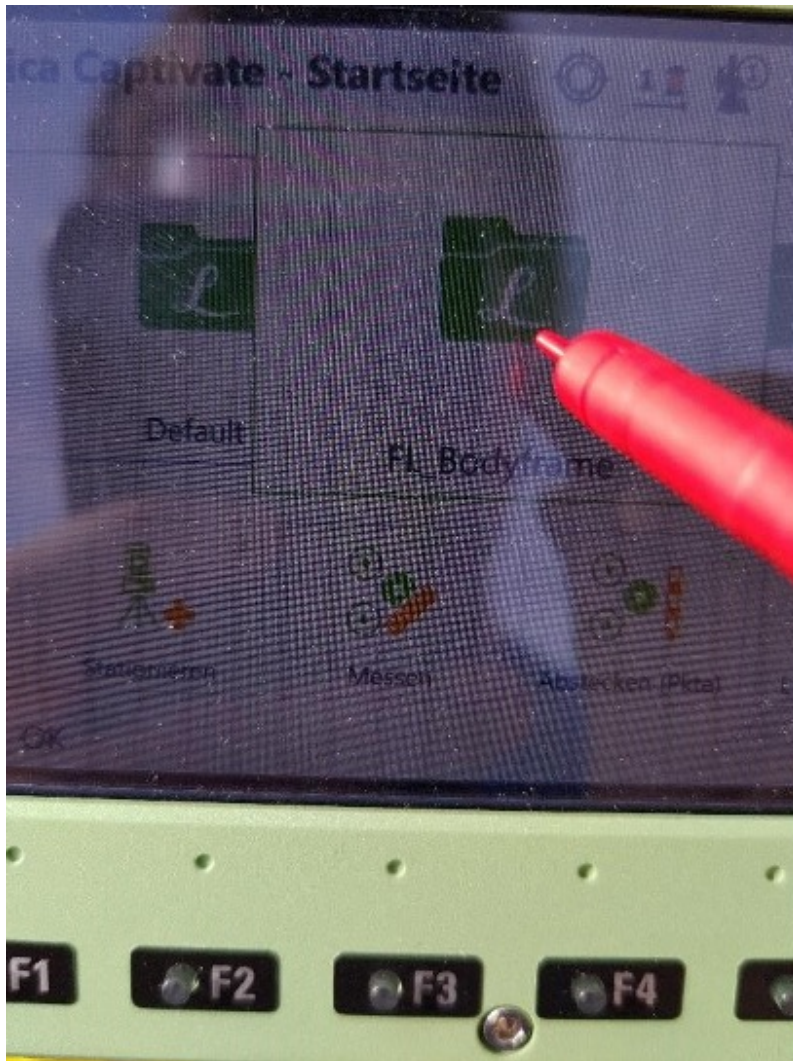
3. Select no code list



4. Don't apply a scale



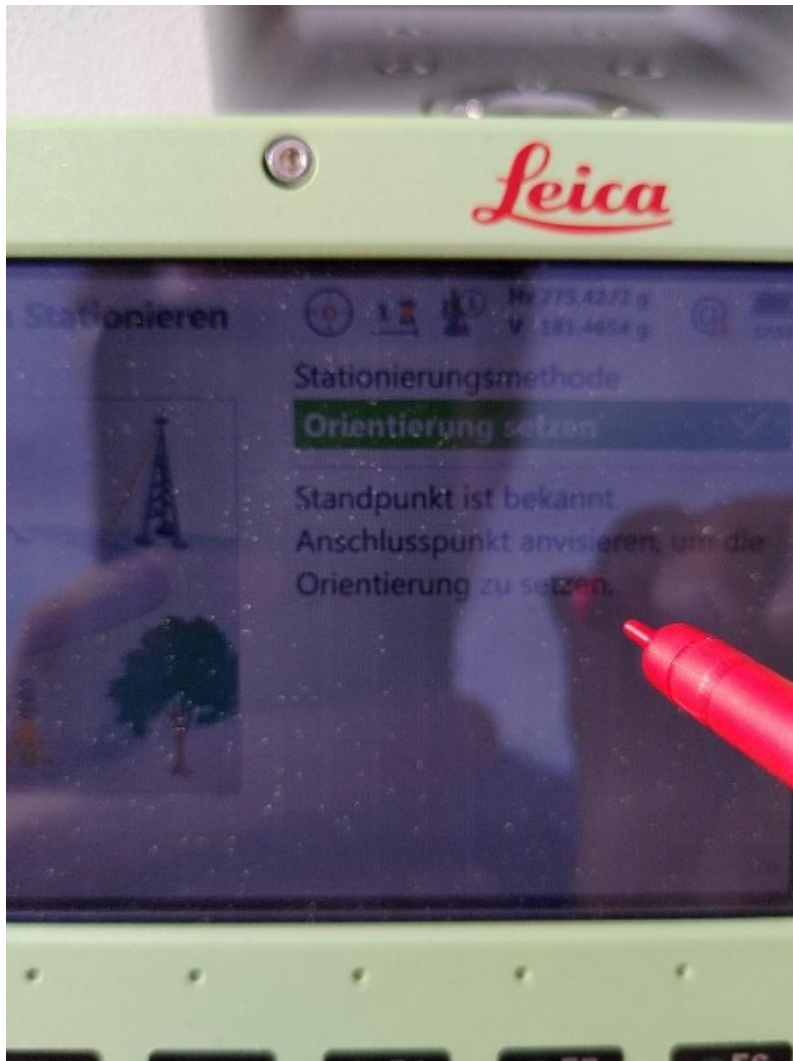
5. Check if the newly created job is selected



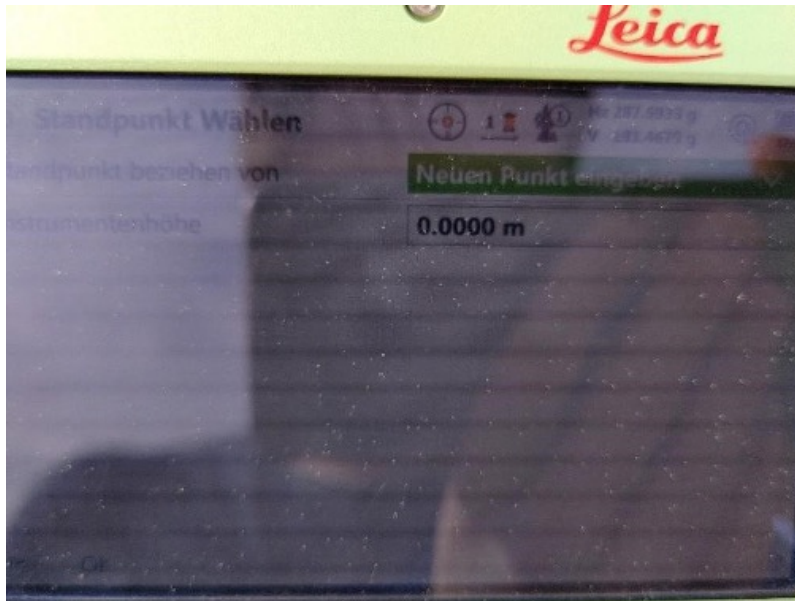
6. Click on "Stationieren"



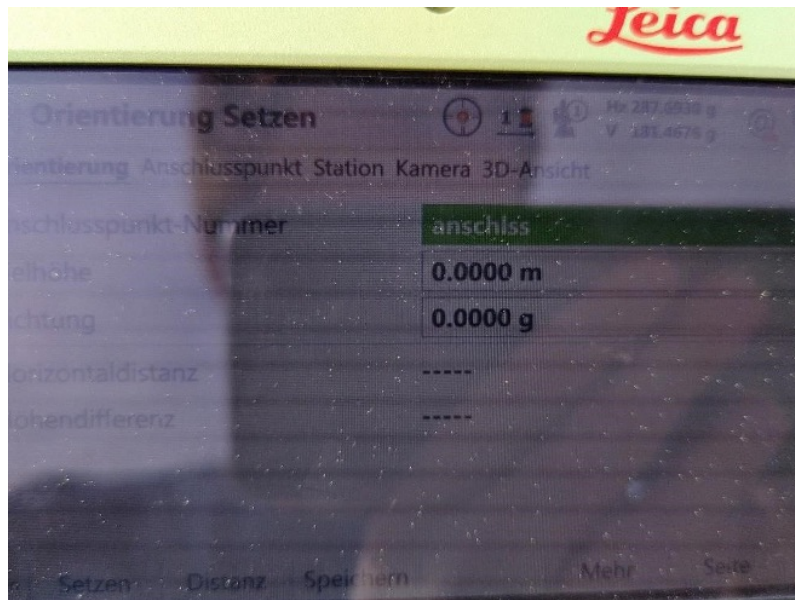
7. Select mode "Orientierung setzen" and click OK



8. Create a **new point** and set coordinates to (100, 100, 100)



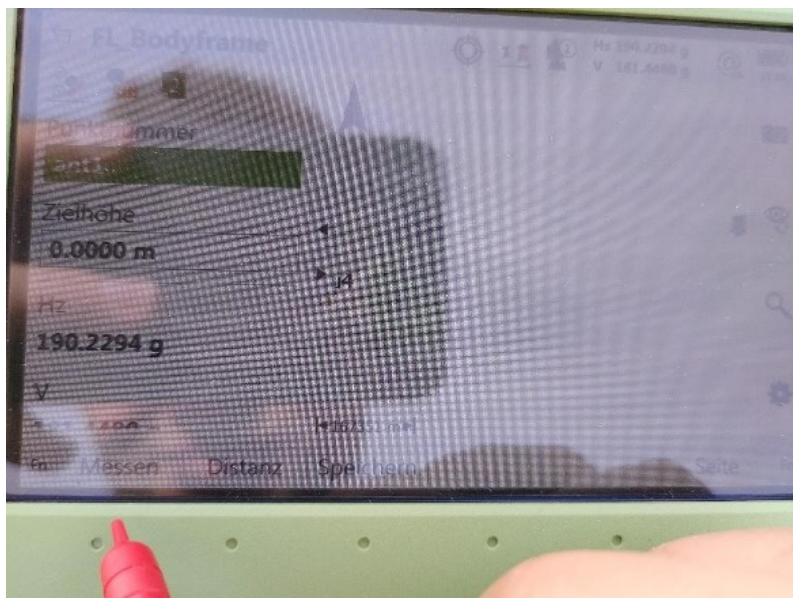
9. For the «Anschlusspunkt» it must be specified what the target height is and how the TS is oriented to it: point the total station to the Anschlusspunkt (eg. antenna 1) and click on “setzen”



10. In the main menu click on “Messen”



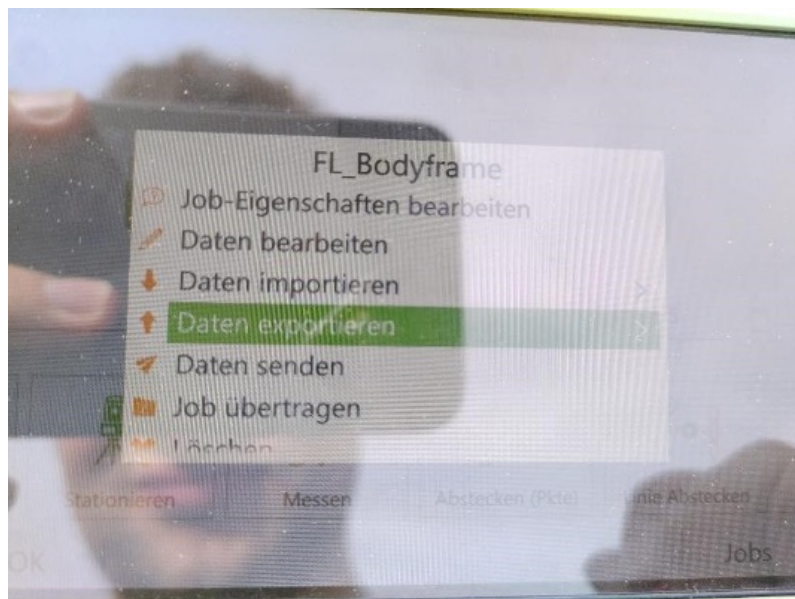
11. Define a good antenna center (on the ublox we used the opening in the **b** as a common center) and hold the mini prism on it, click on *measure*



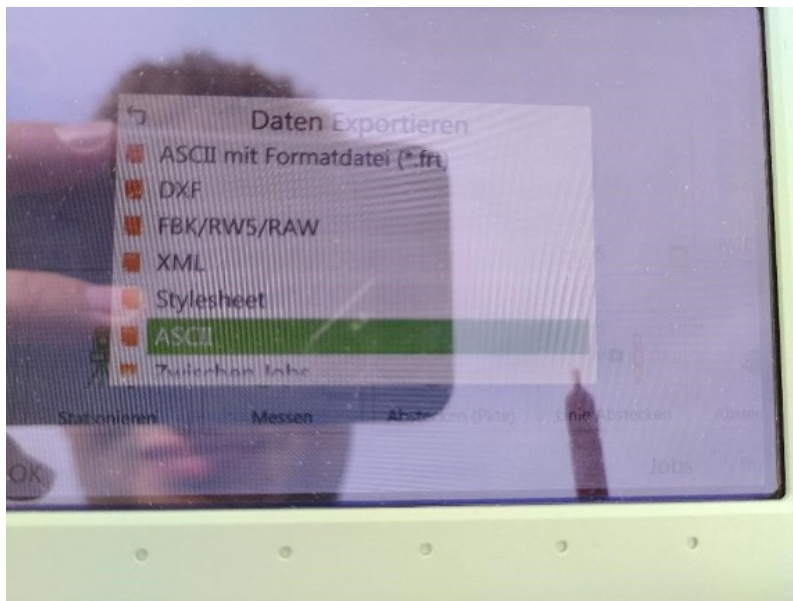
12. Repeat this for all four eyelets at the four corner. Follow the order of the antennas displayed in overview image. Afterwards go to main menu and click on the job



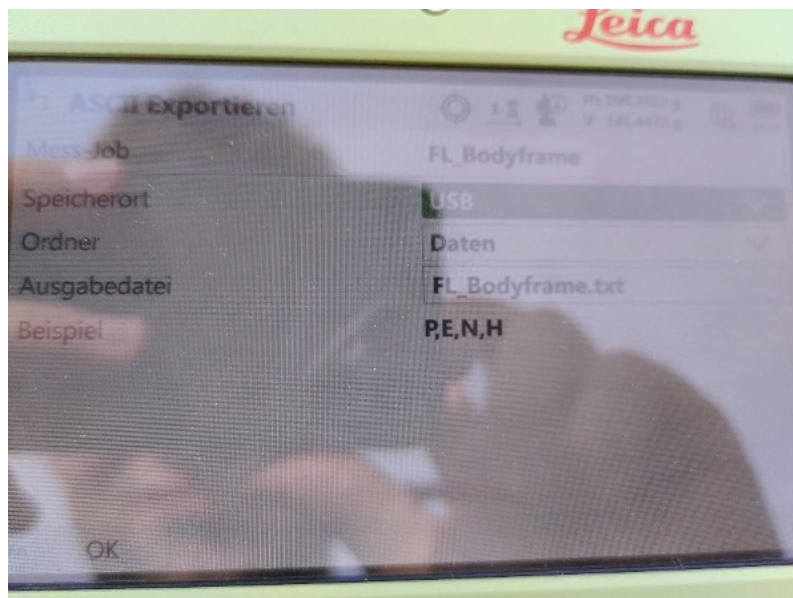
13. Click export data



14. Select ASCII-format



15. Select storage location, storage folder Daten and specify a meaningful name (eg. Eyelets_epochxx.txt)



You can also do *Satzmessungen* for redundancy of the antennas. Due to the short measuring distances and the required accuracy it is non-essential but nice to have for redundancy.

C.4.1 IMU Placement

Follow these additional steps if IMU placement is needed:

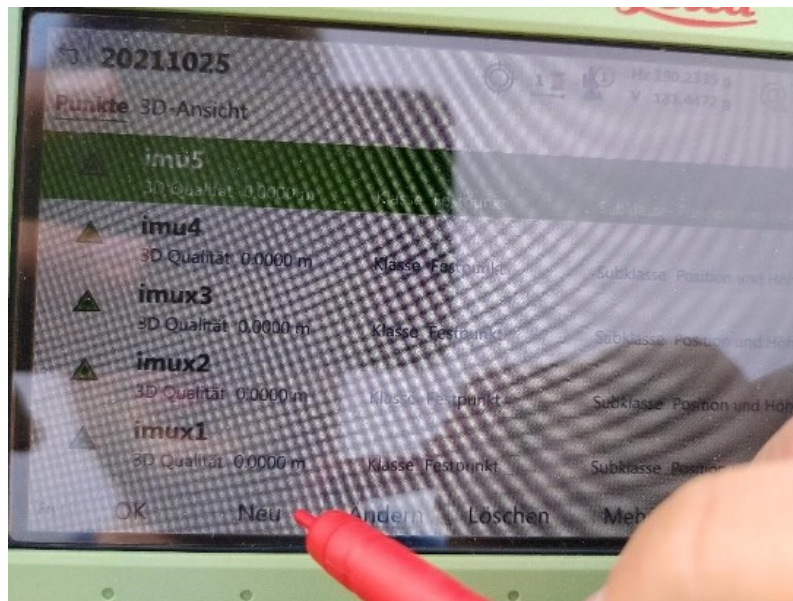
- Get the origin coordinates in the total station coordinate system from line 23 of main script.
 - On the total station main menu select the mode “Abstecken”



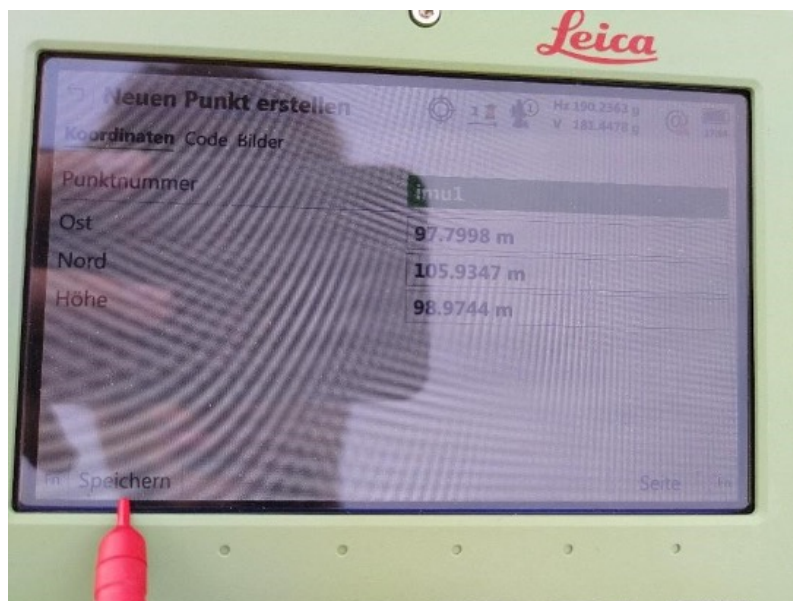
- Click on the point number box



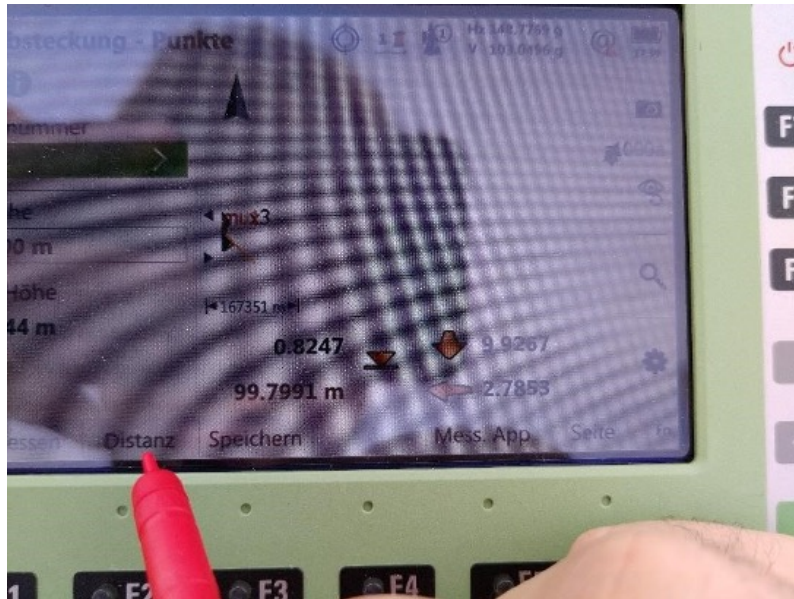
- In the point list create a new point



- Set the point name, insert the coordinates of the origin for East, North and Height



- Place the prism and point with the total station to it and click on distance to measure the distance to the prism



- Search the position where the differences of the two values to the right are minimal. The Height value can be ignored if the IMU cannot be mounted at the origin

If IMU is already placed, measure its position and compare it to the script's origin for validation.

C.5 Hardware Bugs

C.5.1 Bugs with the Loggers of the AIR-ETH Project

The AIR-ETH Project's loggers have a minor issue whereby, when used in conjunction with the u-Blox ANN-MB-01, all loggers (excluding SN21 and SN27) fail to obtain a fixed solution if the antenna is connected during the device's startup process. To remedy this issue, either utilize the ANN-MB-00-00 or connect the antenna after the logger has completed its startup process.

C.6 Processing Procedure

To obtain a geo-referenced solution for your GPR data, it's crucial to have your data files prepared and correctly processed. This guide outlines the necessary prerequisites and steps to achieve a geo-referenced solution for the recorded GPR data.

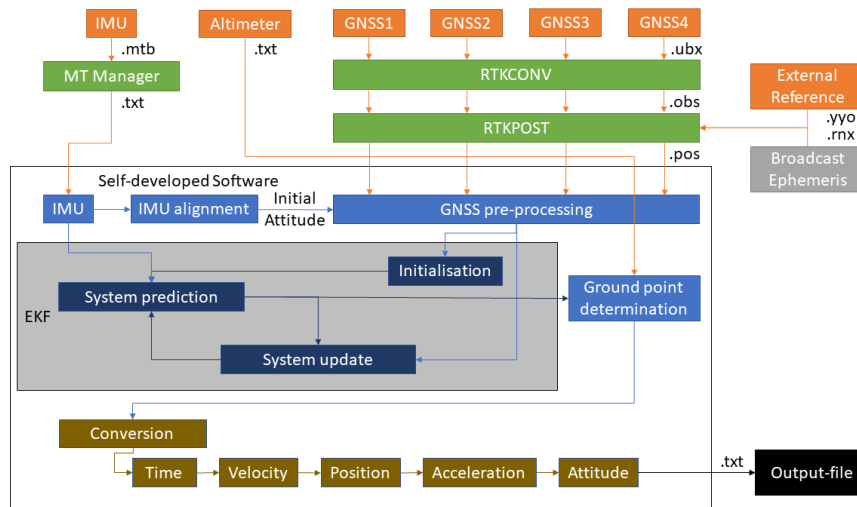


Abbildung C.2: Workflow for processing

C.6.1 Pre-requisites

Software

Refer to the [Home](#) page for the necessary software packages and their installation instructions.

Files

During the data recording, ensure you have the following files to obtain a geo-referenced solution for the GPR data:

Best Practice: gather all the files in one folder for each flight, as this makes it faster to choose the files for processing.

GNSS-related files

- 4 files from the **GNSS-loggers** (.ubx, .ub01, or similar)
- 1 Base-station **observation** file (.XXo, where XX is the year, e.g., 42 for 2042)
- 1 precise RINEX **ephemerides** file of all satellite orbits of the measurement day (see Download Ephemerides, named like BRDC00IGS_R_20231150000_01D_MN.rnx)
- 1 RTKPOST configuration file for demo5. This is our standard configuration.

Other Files

- **1 IMU** file saved as a .txt from XSens MT Manager (see Export IMU File)
- **1 Altimeter** file saved as a .csv as a log from the altimeter control software
- **1 Bodyframe coordinate** file from Bodyframe Determination

Optional:

- **1 IMU offset** file from Bodyframe Determination

C.6.2 Processing Procedure

Once you have all your files ready, perform the following steps:

1. GNSS processing with RTKLIB: Process raw measurements from the loggers into baselines between the base station and antennas.
2. Process the data using the AIR_ETH_processing routine.

To process your observations, you need precise orbit ephemerides, which are calculated daily and distributed by NASA. Follow the steps below to download the ephemerides available the day after your measurement:

1. Visit the NASA CDDIS archive at: <https://cddis.nasa.gov/archive/gnss/data/daily/>.
2. Log in with your username and password. If you do not have an account, click “Register” to create one.
3. Navigate to the folder that corresponds to the measurement year.
4. Open the folder for the corresponding day of the year. You can use this calendar to find the corresponding day of the year.
5. Enter the “yyp” folder, where “yy” represents the last two digits of the year (e.g., for 2022, the folder is “22p”).
6. Look for the file named “BRDC00IGS” and download it. This file will be available the day after your measurement.
7. After downloading the file, right-click on it and extract its contents using your preferred archive client (e.g., 7-Zip on Windows).

By following these steps, you will have successfully downloaded the precise orbit ephemerides required to process your observations. Export the IMU file using MT Manager with following settings:

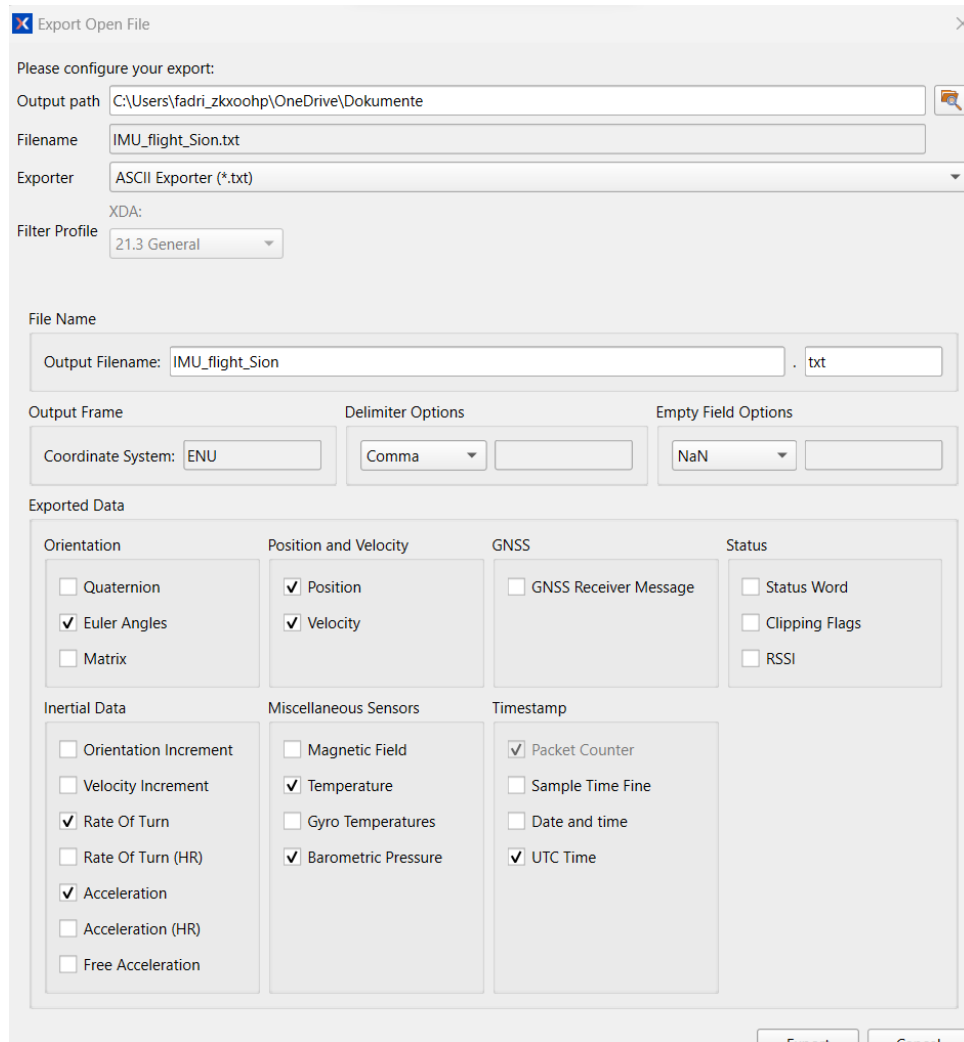


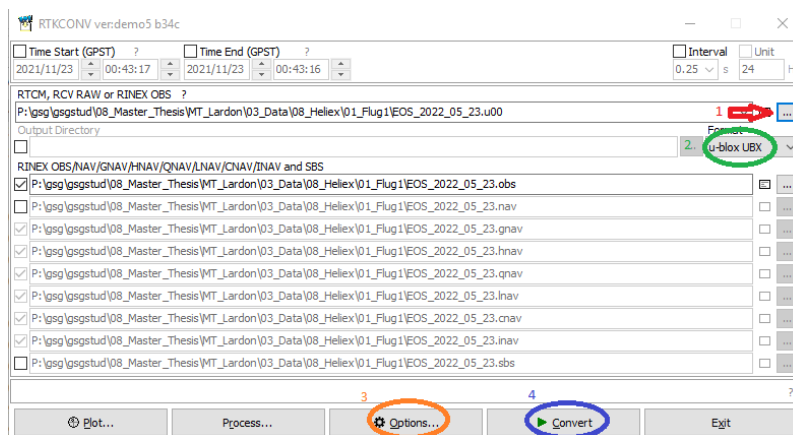
Abbildung C.3: Settings_xsens

To convert raw data from GNSS loggers into .pos files containing measured baselines from the reference station, follow the two-step process using RTKLIB. This process involves converting uBlox files to .obs files using RTKCONV and processing the .obs files using RTKPOST. Follow the steps below to successfully complete this process.

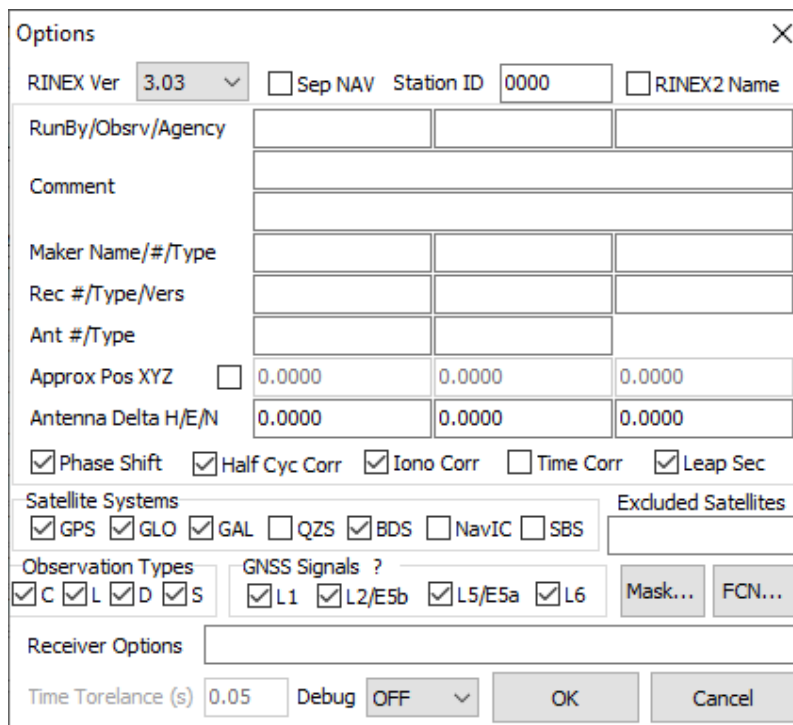
Make sure you are using RTKLIB **demo5** from the repository (see [Home](#) for reference on how to install).

C.6.3 Converting uBlox Files to .obs Files

1. Launch `rtkconv.exe`.
2. Select the raw GNSS measurement file (supported formats: `.ubx`, `.u00`, `.u01`, etc.).
3. Ensure that the file format is set to “u-blox UBX”.



4. Review the options and configure them according to the reference image provided in the documentation.

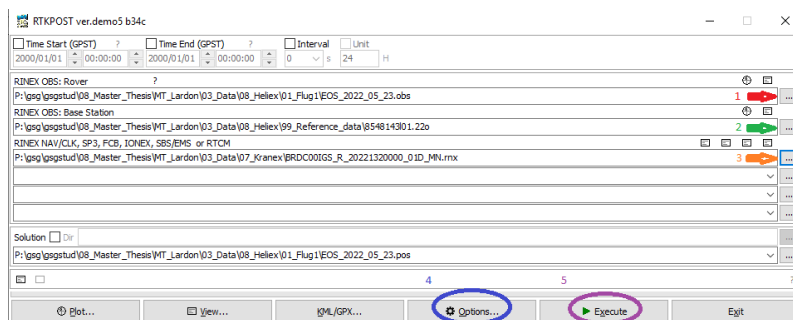


5. Click “Convert” to generate the converted GNSS measurements in a RINEX file.
6. Do this for all remaining antennas

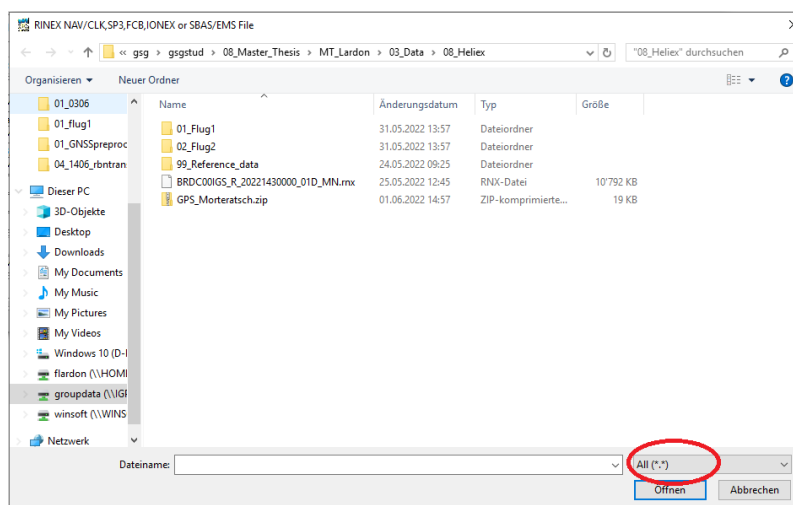
After completing the conversion, verify the presence of observations (denoted as O=#ofObservations). The number of observations should be similar across all antennas.

C.6.4 Baseline Processing using RTKPOST

1. Open `rtkpost.exe`.
2. Select the converted GNSS measurement file (file format: `.obs`, `.yjo`).



3. Select the RINEX file of the base station (file format: `.obs`, `.yjo`).
4. Select the Broadcast ephemerides file (file format: `.rnx`) and refer to the image below.



5. Check and configure options as follows:

1. Click “Load” and select `std_config.conf`.
2. Navigate to “Positions” and refer to the image in the below.
3. Ensure that “Lat/Lon/Height (deg/m)” is selected. Add the instrument height of the GNSS antenna to the height component.
4. Enter the WGS84 coordinates of the base station. If only LV95 coordinates are known, convert them using the Swisstopo Coordinate Converter. If you are using a rinex file where the position is saved in the header, you can choose “Use rinex header position”.

The screenshot shows the 'Options' dialog box with the 'Positions' tab selected. The 'Rover' section has 'Lat/Lon/Height (deg/m)' selected, with values 90.000000000, 0.000000000, and -6335367.6285. The 'Base Station' section also has 'Lat/Lon/Height (deg/m)' selected, with values 47.407067577, 8.510529418, and 594.8420. The 'Station Position File' field is empty. Buttons for 'Load...', 'Save...', 'OK', and 'Cancel' are at the bottom.

6. Click “Execute”.
7. Verify if GNSS measurements are processed (Q=1 fix, Q=2 float). If errors occur, check the following possible reasons:
 - The time span of the base station and GNSS measurements do not overlap (UTC time is 1-2 hours ahead).
 - The Broadcast ephemerides file is for the wrong day.

8. Do this for the remaining antennas

C.7 Data Processing with geoAIR-ETH

There's a new way to start the processing procedure using one Input-Window.

C.7.1 Using AIR-ETH Processing Control Window

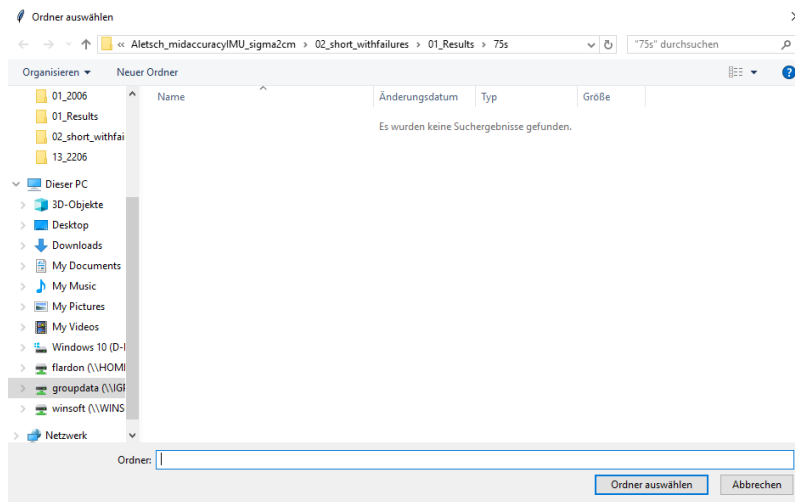
1. Open your Python IDE (IDLE or VSCode), navigate to the AIR-ETH project folder (eg. the one you downloaded from the repository).
2. If necessary add a directory where the workspace is saved (e.g., folder for results)
3. Run the `startup_window.py` script
4. Put in all the values that are needed, it will light up green if you provided a file
5. Click on "Start Processing"

C.7.2 Using AIR-ETH Processing Procedure

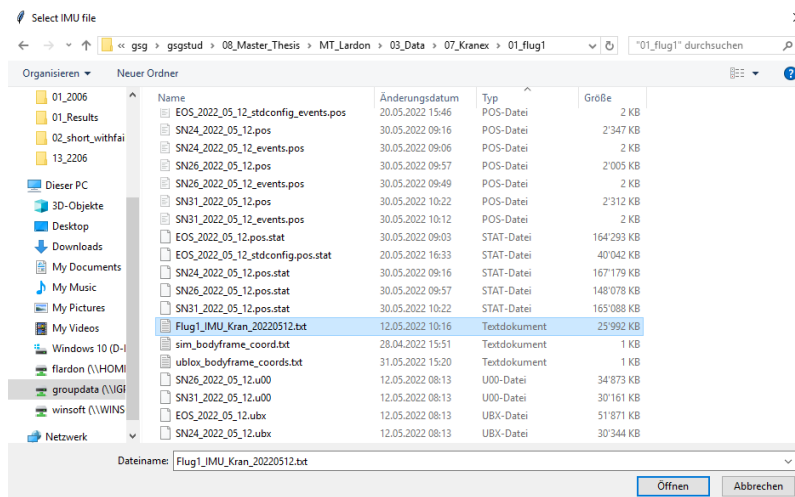
It should not be required to use this routine, but if you want to try it, this still works.

1. Open your Python IDE and open the geoAIR-ETH project folder (eg. the one you downloaded from the repository).
2. If necessary add a directory where the workspace is saved (e.g., folder for results).
3. Run the main script.

4. Select the folder where the results will be stored.

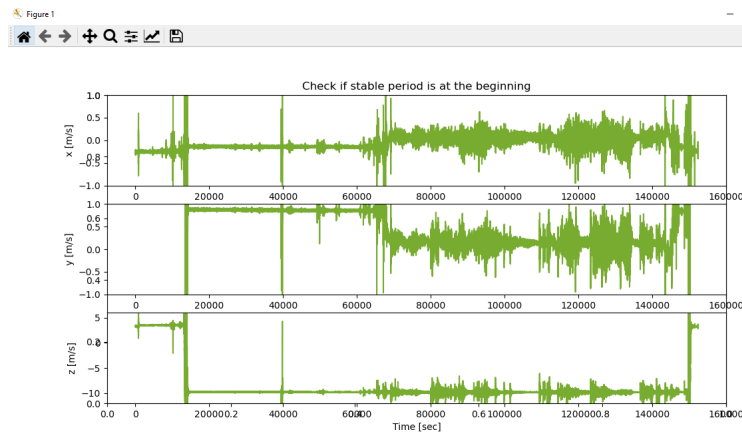


5. Choose the IMU file.

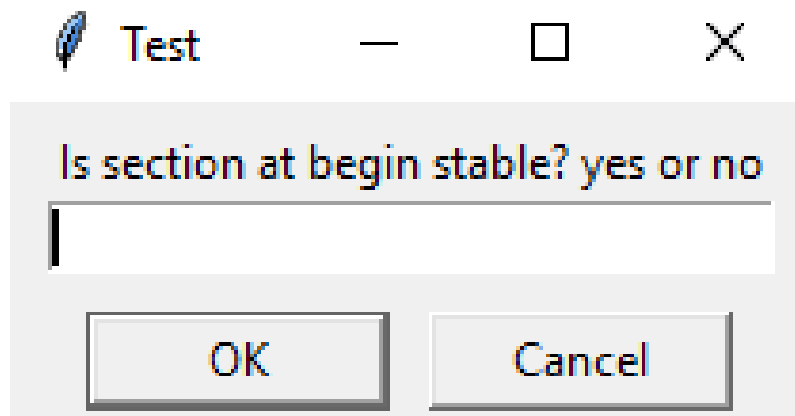


6. Specify a stable period:

- Check if the period is at the beginning, then close the plot.

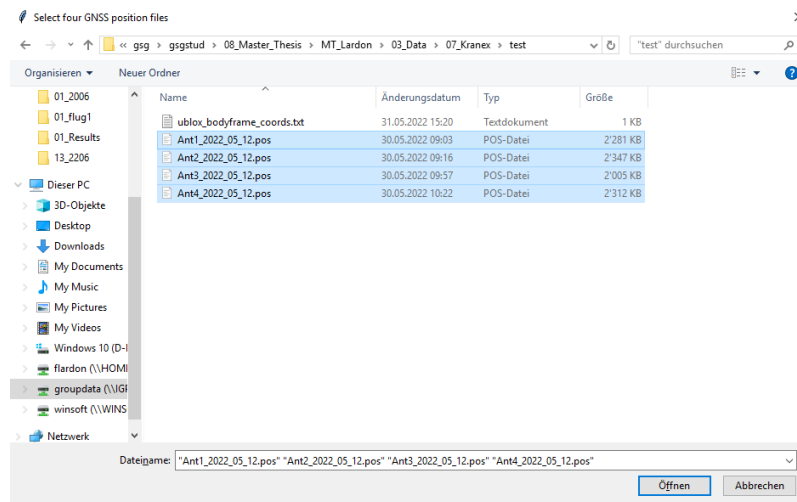


- Specify whether the period at the start is stable.

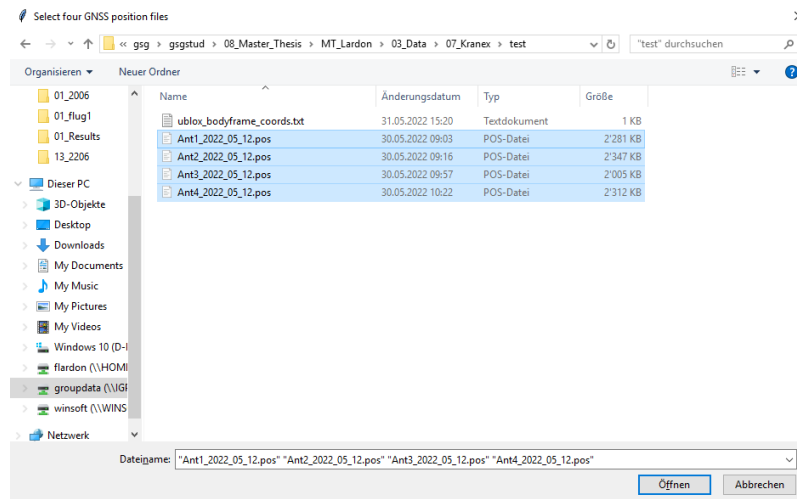


- If yes, the first 2 minutes are used.
- If no, confirm with OK.
- Reopen the plot with raw acceleration data, check the start and end of the stable period, and close the plot.
- Enter the start and end of the stable period.

7. Select the position files from the four GNSS antennas in the correct order.

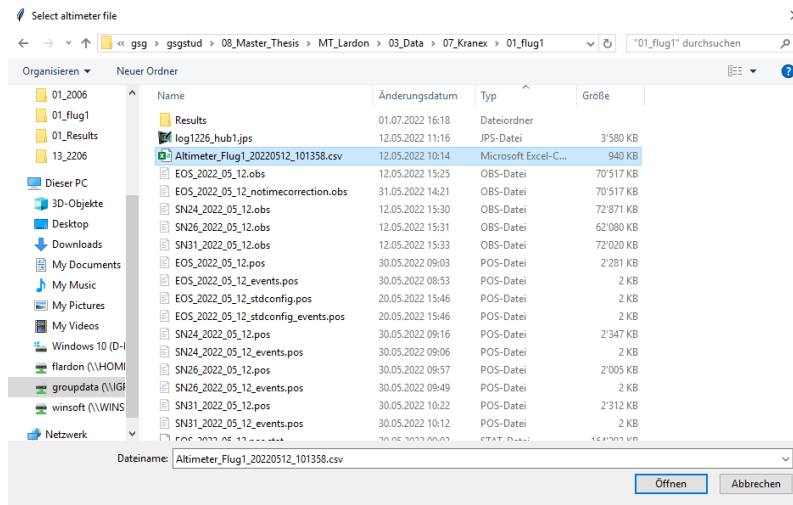


8. Choose the body frame coordinate file of the antennas.



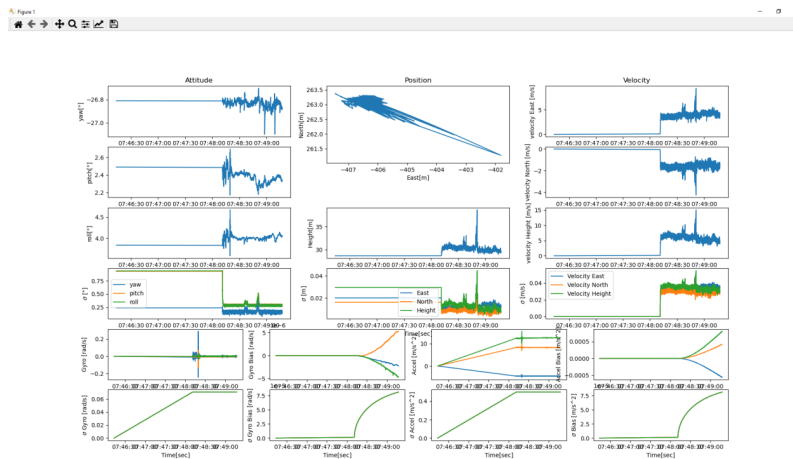
9. Set the GNSS sampling rate.
10. Perform GNSS preprocessing.

11. Select the altimeter file.



12. Specify whether the measurements were collected during summer or winter time (Switzerland).

13. Execute the Kalman filter. During the execution, the results are visualized periodically. Do not close this figure.



14. Perform ground point determination.

15. Select the desired coordinate system (LV95/WGS84) for data conversion.

- If WGS84 is selected, no further action is needed.
- If LV95, enter the coordinates of the reference station.
- If neither, the results are processed in the local navigation frame.

16. For output file generation:

- Specify the flight location (appears in the name of the result).
- Specify the flight segment (number of the flight on that day).

17. Generate and save the output file to the specified folder.

C.7.3 Troubleshooting

Not all possible errors are listed, but the most important ones are:

- GNSS preprocessing attitude estimation is not converging:
 - For only a few epochs: This occasionally happens and is not a problem.
 - For every epoch: Typically, the order of the GNSS files or body frame is incorrect.
- GNSS preprocessing or Kalman filter stopped working:
 - This can sometimes happen; restart the processing.
- Ground point and distance are “nan” in the output file:
 - For some epochs: No problem, as the sampling rate of the altimeter is only about 20 Hz.
 - For all epochs: Typically, there is an issue with time correction.
 - * If flights are in Switzerland: The summer/winter time correction may not have been correctly applied.
 - * For flights in other countries with different time zones: Time correction must be adjusted accordingly.

C.8 Software Documentation

C.9 AIR-ETH Software Documentation

Welcome to the AIR-ETH Software Documentation! This is the central hub for all documentation related to the AIR-ETH **code**. The aim is to provide a comprehensive guide and reference for every **function**, **module**, and component utilized within

our project. Whether you are a collaborator, contributor, or user, we hope that this detailed documentation helps you understand the inner workings of this project and facilitates your interaction with the codebase.

C.9.1 Structure

The documentation should be organized to cover the following areas:

- **Modules Overview:** A high-level overview of all the main modules in our codebase and their functionalities.
- **API Reference:** A detailed description of all functions, classes, methods, and their respective parameters used in the code.
- **GUI Documentation:** Comprehensive information about our application's Graphical User Interface (GUI), detailing the layout, functionalities, and user interactions.
- **FAQs & Troubleshooting:** Commonly asked questions and troubleshooting advice for common issues.

We encourage you to explore these sections to understand how the AIR-ETH software is structured, and how it can be utilized and extended.

C.9.2 Contributions

Your contributions and suggestions are always welcome! If you come across any inconsistencies, errors, or areas that are not adequately explained, feel free to create an issue or just edit the part. This is about the implementation of the `startup_window.py` File. Also there's files for controlling the processing (`processing_control.py`).

C.9.3 Overview

This code implements a graphical user interface (GUI) application for data processing related to flight parameters. The GUI lets users input various types of data (including IMU files, GNSS antenna files, altimeter and bodyframe files) and set several parameters for processing flight data.

C.9.4 Application Class: MyApp

The `MyApp` class is the entry point of the application. It creates a Qt application and a main window (`Window` class). The `run` method starts the Qt event loop.

C.9.5 Main Window Class: Window

The `Window` class represents the main window of the application. It contains a multitude of GUI widgets allowing the user to interact with the application.

GUI Elements

- **Logo:** Displays a logo using `QGraphicsView`, `QGraphicsScene`, and `QPixmap`.
- **Text Fields:** User can input the “Location of Flight” and “Flight Segment” through the provided `QLineEdit` widgets.
- **File Selection Buttons:** Opens a `QFileDialog` to allow users to select necessary files for data processing.
- **Dropdown Menus:** User can select various processing options such as the alignment method, processing method, time of flight, and coordinate system.
- **Stable Period Inputs:** User can input the stable period start and end times for IMU processing.
- **GNSS sampling rate:** User can input the GNSS sampling rate.
- **Checkbox:** User can choose whether to apply IMU Offset Correction or not.
- **Start Processing and Cancel Buttons:** User can start the processing procedure or cancel the application.

Functionality

The user fills in all necessary information and selects the required files. If any field is incorrectly filled or missed, the application will display an error message. If all information is correct, the user can start the data processing by pressing the “Start Processing” button. The processing runs using the parameters provided by the user in the GUI.

C.9.6 ProcessingParams Class

The `ProcessingParams` class is used to store the parameters for the data processing procedure. It includes the following:

- **File Paths**
 - Input Files
 - * Required Input Files Path
- **Alignment and Propagation**
 - Alignment Method
 - Propagation Method
- **GNSS Sampling**
 - Sampling Rate
- **Stable Period**
 - Period Length
- **Location**
 - Location Name
- **Flight Segment**
 - Segment Name
- **Coordinate System**
 - Coordinate System Name



Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Eigenständigkeitserklärung

Die unterzeichnete Eigenständigkeitserklärung ist Bestandteil jeder während des Studiums verfassten Semester-, Bachelor- und Master-Arbeit oder anderen Abschlussarbeit (auch der jeweils elektronischen Version).

Die Dozentinnen und Dozenten können auch für andere bei ihnen verfasste schriftliche Arbeiten eine Eigenständigkeitserklärung verlangen.

Ich bestätige, die vorliegende Arbeit selbständig und in eigenen Worten verfasst zu haben. Davon ausgenommen sind sprachliche und inhaltliche Korrekturvorschläge durch die Betreuer und Betreuerinnen der Arbeit.

Titel der Arbeit (in Druckschrift):

Experimentelle Validierung eines Georeferenzierungssystem für ein helikoptergestütztes Bodenradarsystem

Verfasst von (in Druckschrift):

Bei Gruppenarbeiten sind die Namen aller Verfasserinnen und Verfasser erforderlich.

Name(n):

Pohl

Rischatsch

Vorname(n):

Benedikt Raphael

Fadri Curdin

Ich bestätige mit meiner Unterschrift:

- Ich habe keine im Merkblatt „Zitier-Knigge“ beschriebene Form des Plagiats begangen.
- Ich habe alle Methoden, Daten und Arbeitsabläufe wahrheitsgetreu dokumentiert.
- Ich habe keine Daten manipuliert.
- Ich habe alle Personen erwähnt, welche die Arbeit wesentlich unterstützt haben.

Ich nehme zur Kenntnis, dass die Arbeit mit elektronischen Hilfsmitteln auf Plagiate überprüft werden kann.

Ort, Datum

Sant'Abbondio, 15.06.2023

Unterschrift(en)

Bei Gruppenarbeiten sind die Namen aller Verfasserinnen und Verfasser erforderlich. Durch die Unterschriften bürgen sie gemeinsam für den gesamten Inhalt dieser schriftlichen Arbeit.