

# Bestimmung von Position und Orientierung des Eigenschiffes mittels Korrelation von AIS-und Radar-Informationen

Jannik Knopp

# Abstract

lore ipsum

# Inhaltsverzeichnis

<b>Abstract</b>	<b>i</b>
<b>1 Einleitung</b>	<b>1</b>
1.1 Problemstellung . . . . .	1
1.2 Zielsetzung . . . . .	1
1.3 Abgrenzung . . . . .	2
1.4 Aufbau der Arbeit . . . . .	2
<b>2 Grundlagen</b>	<b>3</b>
2.1 Allgemeines . . . . .	3
2.1.1 NMEA 0183 . . . . .	3
2.1.2 WGS 84 . . . . .	4
2.1.3 Berechnungen auf dem Erdellipsoid . . . . .	4
2.2 Point Set Registration . . . . .	5
2.2.1 Rigid und Non-rigid Registration . . . . .	5
2.2.2 Stochastisches Point Set Registration . . . . .	5
2.2.3 Gaussian Mixture Model . . . . .	5
2.2.4 EM-Algorithmus . . . . .	5
2.2.5 Singulärwertzerlegung . . . . .	5
2.2.6 Coherent Point Drift . . . . .	5
<b>3 Literaturvergleich</b>	<b>6</b>
3.1 Fusion von AIS- und ARPA-Signalen . . . . .	6
3.2 Point Set Registration Methoden . . . . .	6
3.3 Position Fix Methoden / Sicherheitssteigerung der Positionsbestimmung . . . . .	6
<b>4 Umsetzung</b>	<b>7</b>
4.1 Ansätze . . . . .	7
4.1.1 Iterative Closest Point . . . . .	7
4.1.2 Graph-Matching . . . . .	7
4.1.3 Coherent Point Drift . . . . .	7
4.2 Implementierung . . . . .	8

4.2.1	Programmablauf . . . . .	8
4.2.2	Ungenauigkeiten durch den Erdellipsoiden . . . . .	8
4.2.3	Rotationsschwäche von Coherent Point Drift . . . . .	8
4.3	Position-Fix . . . . .	9
4.3.1	Bestimmung der Eigenposition . . . . .	9
4.3.2	Bestimmung der Sicherheit der Eigenposition und Eliminierung von Ausreißern . . . . .	9
4.3.3	Erstellung des Ergebnisses . . . . .	9
<b>5</b>	<b>Bewertung</b>	<b>10</b>
5.1	Genauigkeit . . . . .	10
5.2	Robustheit . . . . .	10
5.3	Ausblick . . . . .	10
<b>6</b>	<b>Schluss</b>	<b>11</b>
6.1	Zusammenfassung . . . . .	11

# 1. Einleitung

## 1.1 Problemstellung

Die Eigenposition eines Schiffes gehört zu den wichtigsten navigatorischen Daten. Diese wird meistens über einen GPS-Empfänger bestimmt. Das GPS-Signal ist manipulierbar, somit besteht die Möglichkeit einer böswilligen Manipulation durch einen Angreifer. Dadurch fällt der GPS-Empfänger aus oder gibt eine falsche Position zurück. In diesen Fällen ist die Eigenposition nicht mehr verfügbar oder nicht vertrauenswürdig, somit ist es notwendig eine Methode zur Positionsbestimmung zu entwickeln, welche unabhängig von dem eigenen GPS-Empfänger arbeitet und somit in der Lage ist dessen Daten zu verifizieren.

Zur Bestimmung der Eigenposition können die Daten aus dem Radar, die GPS-Daten der Schiffe in unmittelbarer Nähe (über AIS bekannt) und die Daten der Seekarte genutzt werden. Das Automatic Identification System (AIS) ist ein unverschlüsseltes Funksystem, zum Austausch von Schiffs- und Navigationsdaten. Durch die AIS-Daten ist die absolute Position von Schiffen und ATONs (*Aid to navigation*) bekannt. ATONs sind markante Landmarken oder Bojen zur Unterstützung der Navigation. Durch das Radar erhält man die relative Position der Schiffe und der ATONs zu der Eigenposition. Durch diese beiden Daten lässt sich ein Rückschluss auf die Eigenposition ziehen. Die Zuordnung zwischen AIS- und Radar-Signalen ist aktuell abhängig von der Eigenposition. Da diese bestimmt werden soll, muss ein Algorithmus entwickelt werden, welcher unabhängig von dieser die Zuordnung finden kann. Durch das Finden einer Zuordnung findet sich auch die Transformation, die beide Punktwolken unterscheidet.

## 1.2 Zielsetzung

Das Ziel der Thesis ist die Entwicklung eines Algorithmus, welcher die Eigenposition und den Kurs (Heading) des Eigenschiffes möglichst genau bestimmen kann. Dieser soll unabhängig der eigenen GPS-Daten funktionieren. Der Algorithmus soll später als eine Art Sensor für die Brücke eingesetzt werden, das heißt die Brücke sendet die benötigten Daten an den Algorithmus und dieser sendet die errechnete Eigenposition und eine Darstellung zurück an die Seekarte.

## 1.3 Abgrenzung

Das Produkt der Thesis ist ein externes Softwaremodul, welches in Zusammenarbeit mit Herrn André Becker entwickelt wird. Dieser verfasst ebenfalls seine Thesis über einen Teil dieses Softwaremoduls. Die Thesen lassen sich wie folgt abgrenzen, Herr Becker entwickelt einen Teil der Vorfilterung der Daten, einen Point-Matching Algorithmus und eine Netzwerkschnittstelle zur Kommunikation des Softwaremoduls mit der Schiffsbrücke. Diese Thesis umfasst einen weiteren Teil der Vorfilterung der Daten, ebenfalls einen Point-Matching Algorithmus und einen Algorithmus zur Positionsbestimmung. Das Einlesen der Daten wird über einen Daten Parser realisiert, welcher von Raytheon Anschütz gestellt wurde und somit nicht Teil der Thesis ist. Ebenso wird die Funktionalität über Daten aus einem Daten Generator verifiziert, welcher ebenfalls von Raytheon Anschütz gestellt wurde.

## 1.4 Aufbau der Arbeit

Die folgende Arbeit behandelt in Kapitel 2 wichtige Grundlagen die benötigt werden, um im weiteren die Problemstellungen und die Algorithmen zu verstehen. Das dritte Kapitel zeigt andere Ansätze aus der Literatur zu diesem Thema und wie diese zur aktuellen Arbeit in Verbindung stehen. In Kapitel 4 werden verschiedene Ansätze und die Umsetzung beschrieben. Danach wird in Kapitel 5 das erarbeitete Ergebnis bewertet und es wird ein Ausblick über mögliche Optimierungen gegeben.

## 2. Grundlagen

### 2.1 Allgemeines

.

#### 2.1.1 NMEA 0183

Der NMEA 0183 ist ein Standard zur Kommunikation zwischen Geräten auf einer Schiffsbrücke. Dieser wurde von der National Marine Electronics Association 1983 veröffentlicht. In diesem ist unter anderem die Struktur der Nachrichten definiert.

Die Nachrichten, welche über NMEA gesendet werden enthalten z.B. Daten von Systemen auf dem Schiff wie dem Radar, Daten von Sensoren wie Kompass und GPS-Empfängern. Hier rüber werden auch zusätzlich AIS-Nachrichten empfangen aus der nahen Umgebung

Eine solche NMEA - Nachricht ist wie folgt aufgebaut:

Sie beginnt mit einem „!“ oder einem „\$“ als Startzeichen. Darauf folgen zwei Zeichen Geräte ID, dadurch ist erkennbar woher die gesendeten Daten kommen. Auf die Geräte ID folgen drei Zeichen die den Nachrichtentyp bezeichnen, an welchen auch die folgenden Datenfelder zu identifizieren sind. Nun folgen die Datenfelder, welche für jeden Nachrichtentyp spezifisch sind. Am Ende der Nachricht folgt eine optionale Prüfzahl, welche mit einem „\*“ abgetrennt wird. Die Nachricht wird mit einem Zeilenumbruch beendet.

Im folgenden beschreibt Abbildung 1.1 eine Tracked Target Management (TTM) NMEA - Nachricht. Dieser Nachrichtentyp enthält Schiffe aus dem Radar die als solche erkannt wurden und nun kontinuierlich über das Radar verfolgt werden.



- Abbildung mit Genauigkeiten



## **2.2 Point Set Registration**

### **2.2.1 Rigid und Non-rigid Registration**

Unterschied zwischen Rigid und Non-rigid registration erklären und kurze Erwähnung der Probleme welche bei Rigid Registration auftreten

### **2.2.2 Stochastisches Point Set Registration**

Vergleich von Stochastischen Point Set Registration Methoden( basieren meist auf GMMs) und anderen Methoden z.B. Iterative Closest point

### **2.2.3 Gaussian Mixture Model**

Erklärung von Gaussian Mixture Models im Zusammenhang mit Point Set Registration

### **2.2.4 EM-Algorithmus**

Erklärung des EM Algorithmus und welchen nutzen er für Point Set Registration hat

### **2.2.5 Singulärwertzerlegung**

Erklärung von Singulärwertzerlegung

### **2.2.6 Coherent Point Drift**

Erklärung der Methodik des Coherent Point Drift Algorithmus und des mathematischen Hintergrunds anhand der letzten drei Punkte

## 3. Literaturvergleich

### 3.1 Fusion von AIS- und ARPA-Signalen

Es existieren mehrer Arbeiten zum Thema Fusion von Arpa und Ais Signalen, diese werden betrachtet und kurz bewertet wie sie zu dieser Aufgabenstellung passen

### 3.2 Point Set Registration Methoden

Mehrere Rigid Point Set Registration Methoden werden betrachtet.

### 3.3 Position Fix Methoden / Sicherheitssteigerung der Positionsbestimmung

Andere Ansätze zur Sicherstellung der Positionsdaten werden erklärt (Hier ist noch keine konkrete Literatur recherchiert kann sich deswegen noch ändern)

## 4. Umsetzung

### 4.1 Ansätze

#### 4.1.1 Iterative Closest Point

- Allgemeine beschreibung
  - Minimierung der Distanz zwischen Punktwolken
  - fixed, moving Cloud
  - Translation und Rotation
  - error metric: distance between points / sum squarred distances
  - Viele Varianten (alle Verlinken)
  - Stärken: schnell und simpel
  - Schwächen: outliers, rotation, noise in Data, hard matching(0 or 1)?

#### 4.1.2 Graph-Matching

- Allgemeine Beschreibung
  - finden von Subgraphen
  - Funktionsweise kurz beschreiben
  - Stärken: schnell und genau
  - Schwächen: outlier können nicht gefunden werden => falsches Ergebnis
  - Grafik? (wenn Platz ist ja)

#### 4.1.3 Coherent Point Drift

- Einleitung passend zu Grundlagen
  - Begründung warum CPD verwendet wurde:
  - outlier erkennen
  - noise Data
  - soft matching(0.0 - 0.5 - 1.0)? => Falsche matchings können besser erkannt werden
  - Performanz Vergleich? (Wieder finden im Paper im Vergleich zu ICP)

## 4.2 Implementierung

### 4.2.1 Programmablauf

Beschreibung was das Programm machen soll (kurz, einleitend)

- Grafik zum Ablauf des Softwaremoduls
- Daten einlesen => Daten die benötigt werden kurz nennen
- Daten Filtern => Filter nennen
- Algorithmus auswählen
- AIS Interpolieren kurz beschreiben und warum
- Arpa zu WGS 84 Koordinaten
- Start des Algorithmus
- Verschiebung beider Punktwolken auf den Ursprung des Koordinatensystems
- Ergebnis = Zuordnung zwischen den Punkten und Rotation
- Position Fix Eigenposition bestimmen
- Eigenposition und Heading als NMEA-Nachricht
- Ergebnis SVG senden

### 4.2.2 Ungenauigkeiten durch den Erdellipsoiden

Wie entstehen ungenauigkeiten durch den Algorithmus? => Rotation und Translation sorgen für Ungenauigkeiten da mit WGS 84 Koordinaten gerechnet wird

- Rotation und Translation auf einer Ellipse/Kreis (Grafik kartesische Koordinaten System Vergleich mit Kugel Koordinatensystem)
- Verschiebung auf gemeinsamen Punkt (Ursprung da Position im worst case sowieso unbekannt ist) Rotation sorgt für genaue Ergebnisse
- Arpa Verschieben mit initial Position auf 0,0
- AIS Verschieben, Abstände speichern zu aisMittelwert und dann mittels dieser Abstände die punktwolke an 0,0 neu konstruieren
- Abbildung für Ungenauigkeiten? (Müsste noch erstellt werden)

### 4.2.3 Rotationsschwäche von Coherent Point Drift

- Experimente beschreiben und Visualisieren (Tabelle oder so) über die Versuche bis zu welchem grad die Punktwolke rotiert sein darf (Möglicherweise nicht nur für WGS 84 Koordinaten sondern auch für kartesische Koordinaten)

- Paper verlinken welches dasselbe beobachten konnte
- Lösung für die Rotation:
- Mehrfachausführung mit initialem Drehwinkel für die Arpa Wolke
- Ergebnis mit dem geringstem Fehler auswählen und zurückgeben (Methodik hier genauer beschreiben)
- Erklärung das es nur einmalig ist, da nach dem ersten Durchlauf wieder ein Heading existiert an dem man sich orientieren kann.

## **4.3 Position-Fix**

### **4.3.1 Bestimmung der Eigenposition**

- Matches aus den Daten erstellen (Zuordnung verwenden und True bearing bestimmen)
- Zurückrechnen der Eigenpositionen
- Berechnung gewichteter und arithmetischer Mittelwert
- Grafik für die Mittelwerte um die Auswirkung beschreiben zu können.

### **4.3.2 Bestimmung der Sicherheit der Eigenposition und Eliminierung von Ausreißern**

Bestimmung der Mahalanobis IDstanz von einem Punkt zum Rest als relative Abstandsmetrik

- für jeden Punkt wiederholen und speichern
- mit % Wert abgleichen und entfernen falls er zu weit von der Punktwolke entfernt ist
- Sicherheitsbereich über Standardabweichung bestimmen
- Grafik aus der EMS Demo einfügen

### **4.3.3 Erstellung des Ergebnisses**

Kurze Erklärung wie das Ergebniss erstellt wird (NMEA Telegram und Grafik) um von der Netzwerkschnittstelle gesendet werden zu können.

## 5. Bewertung

### 5.1 Genauigkeit

Tests mit Generatordaten werden dargelegt, erklärt und ausgewertet. Es werden außerdem Methoden zur Genauigkeitssteigerung des Algorithmus bewertet.

### 5.2 Robustheit

Tests mit Generatordaten mit zunehmenden Ausreißern und Auswertung der Ergebnisse

### 5.3 Ausblick

Mögliche Optimierungen für den Algorithmus werden beschrieben.

## 6. Schluss

### 6.1 Zusammenfassung

# Nomenklatur

ICP    Iterative Closest Point