Random-Finite-Set-basiertes Multisensor-Multiobjekttracking für automatisierte und vernetzte Fahrzeuge: Aktueller Forschungsstand und offene Fragen

Martin Herrmann*, Charlotte Hermann*, Klaus Dietmayer†
und Michael Buchholz‡

Zusammenfassung: Die zeitliche und räumliche Verfolgung anderer Verkehrsteilnehmer ist ein essentieller Bestandteil automatisierter und vernetzter Fahrzeuge sowie vernetzter Infrastruktursensorsysteme und wird häufig mit Random-Finite-Set-basierten Methoden gelöst. Dabei stellt die Assoziation der beobachteten Objekte mit den Messungen insbesondere im Multisensorfall eine große algorithmische Herausforderung dar. Dieser Artikel fasst die wissenschaftlich beschriebenen Lösungsmethoden hierfür zusammen und ordnet insbesondere die in letzter Zeit an unserem Institut entwickelten parallelisierbaren Methoden auf Basis der Bayes-Parallel-Combination-Rule in den wissenschaftlichen Kontext ein. Letztere betrachten wir tiefergehend und diskutieren offene Fragestellungen und potenzielle Lösungsansätze.

Schlüsselwörter: Bayes-Parallel-Combination-Rule, Generalized-Labeled-Multi-Bernoulli, Labeled-Multi-Bernoulli, Product-Multi-Sensor-Multi-Object Tracking, Random-Finite-Sets

1 Einleitung

Jedem Autofahrer dürfte intuitiv bewusst sein, dass eine erfolgreiche Umgebungswahrnehmung zur Erfüllung der Fahraufgabe essentiell ist. Automatisierte Fahrzeuge sind für die Perzeption auf interne Sensorik angewiesen [1], die im Fall vernetzter Fahrzeuge durch externe Informationen anderer Fahrzeuge oder vernetzter Infrastruktur ergänzt werden kann. Die aus den Sensoren gewonnenen Daten werden in gängigen Architekturen unter anderem von Filteralgorithmen verarbeitet, die eine zeitliche und räumliche Zuordnung zwischen den Messungen und den beobachteten Verkehrsteilnehmern herstellen. Auf Basis der so gewonnenen Information über die vergangenen Trajektorien anderer Verkehrsteilnehmer kann ein automatisiertes Fahrzeug seine eigene Trajektorie planen und sogar Rückschlüsse auf das voraussichtliche zukünftige Verhalten der anderen ziehen.

*M. Herrmann und C. Hermann sind akademische Mitarbeitende am Institut für Mess-, Regel- und Mikrotechnik der Universität Ulm, Albert-Einstein-Allee 41, 89081 Ulm (martin.herrmann@uni-ulm.de; charlotte.hermann@uni-ulm.de).

†K. Dietmayer ist Leiter des Instituts für Mess-, Regel- und Mikrotechnik der Universität Ulm, Albert-Einstein-Allee 41, 89081 Ulm (klaus.dietmayer@uni-ulm.de).

‡M. Buchholz ist akademischer Oberrat am Institut für Mess-, Regel- und Mikrotechnik der Universität Ulm, Albert-Einstein-Allee 41, 89081 Ulm (michael.buchholz@uni-ulm.de).

Um die Messungen aller Sensoren zu verarbeiten, kommen sogenannte Multiobjekttracker zum Einsatz. Während die Literatur mit den Nearest-Neighbor- (NN), den Probabilistic-Data-Association- (PDA) und den Multi-Hypothesis-Tracking-Verfahren (MHT-Verfahren) [2] eine Vielzahl unterschiedlicher Methoden bereitstellt, beschränkt sich dieser Artikel im Folgenden nur auf die Random-Finite-Set-basierten (RFS-basierten) Methoden. Genauer gesagt betrachten wir ausschließlich das Generalized-Labeled-Multi-Bernoulli-Filter (GLMB-Filter) [3] sowie dessen Approximation, das Labeled-Multi-Bernoulli-Filter (LMB-Filter) [4]. Beide nutzen Mahlers Finite-Set-Statistics-Framework (FISST-Framework) [5, 6] und können über die Multiobjekt-Bayesformel hergeleitet werden. Überdies handelt es sich beim GLMB-Filter um das bisher einzige rechentechnisch implementierbare Multiobjektfilter, für das Bayes-Optimalität eindeutig gezeigt wurde [7].

Trotz geeigneter Approximationsverfahren ist das GLMB-Filter in vielen, insbesondere komplexen Szenarien rechentechnisch herausfordernd. Sowohl in der Prädiktion als auch im Update entstehen eine Vielzahl unterschiedlicher Hypothesen, die in der Folge betrachtet werden müssen. Folglich wird häufig, so auch in unseren Anwendungen im automatisierten und vernetzten Fahren [8–10], das LMB-Filter genutzt, welches besonders die Prädiktion stark vereinfacht. Es hat unter anderem deshalb enorme rechentechnische Vorteile bei nur sehr geringen Leistungseinbußen. Unglücklicherweise stößt aber auch das LMB-Filter im Multisensorfall schnell an seine Grenzen, was sich auch in unseren Versuchen an der Infrastrukturanlage in Ulm-Lehr zeigt, bei der zusätzlich ein relativ komplexes Messmodell für Objektreferenzpunktmessungen zur Anwendung kommt [11]. Die Gründe hierfür liegen insbesondere in der fehlenden Möglichkeit, Messungen mehrerer Sensoren parallel zu verarbeiten, und in der NP-Schwere des Messung-zu-Track-Assoziationsproblems [12].

Dieser Beitrag widmet sich den Methoden, die verschiedene Forschungsgruppen zur Lösung dieser Herausforderungen in den letzten Jahren entwickelt haben. Wir konzentrieren uns dabei ausschließlich auf zentralisierte Ansätze, da die dezentralen Varianten unter Verwendung der suboptimalen Track-to-Track-Fusion ein deutlich geringeres Leistungsniveau zeigen [2, 13]. Anhand praktischer Anwendungsfälle erörtern wir noch offene Fragestellungen, die einer praxistauglichen Implementierung entgegenstehen, und präsentieren potenzielle Lösungsansätze.

2 Zentralisiertes Multisensor-Multiobjekttracking

Nach Bar-Shalom [2] unterscheidet man grob drei unterschiedliche Fusionsansätze: Messdatenfusion, zentralisiertes Multisensortracking und Track-to-Track-Fusion. Diese unterscheiden sich grundlegend anhand der fusionierten Datentypen. Bei der Messdatenfusion kombiniert ein vorgelagertes Modul die Sensordaten zu einer Art Supermessung, welche dann zeitlich gefiltert wird. Demgegenüber übernimmt der zentralisierte Multisensortracker die Fusion der Sensordaten implizit bei der Verarbeitung der Sensormessungen. Im letzten Fall steht allen Sensoren ein eigener Tracker zur Verfügung, deren gefilterte Daten anschließend fusioniert werden.

Wie bereits eingangs erwähnt, steht die Track-to-Track-Fusion den anderen Verfahren hinsichtlich der Leistungsfähigkeit stark nach, hat aber auch völlig andere Einsatzgebiete, beispielsweise in verteilten Architekturen mit wechselnden Netzwerktopologien und Kommunikationsteilnehmern. Die beiden anderen Methoden sind jedoch heutzutage nur noch schwer auseinanderzuhalten und oft in Mischformen präsent, weshalb generelle Aussagen

schwierig zu treffen sind. Jedoch sind Multisensorszenarien auf Basis reiner Messdatenfusion im betrachteten Anwendungsfeld automatisiertes und vernetztes Fahren eher untypisch, da sich beispielsweise Sensorsichtbereiche meist nicht vollständig überlappen oder die Modellierung für unterschiedliche Sensorprinzipien kompliziert und stark vom Sensortyp abhängig ist. Folglich hat man auch bei vorfusionierten Sensordaten oft einen Multisensortracker, der die Daten mehrerer Sensoren oder Sensorgruppen filtert und fusioniert.

In dieser Algorithmenklasse differenzieren wir erneut drei Methoden, nämlich solche mit Multisensor-Messmodell, serielle Ansätze und parallelisierbare Ansätze. Diese klassifizieren wir im Folgenden jeweils kurz und stellen die jeweiligen Vertreter mit entsprechender (Multiobjekt-)Wahrscheinlichkeitsdichtefunktion (WDF) [5] ausführlich vor, nachdem wir die grundlegende Funktionsweise der GLMB- und LMB-Filter zusammengefasst haben. Die Erläuterungen sind mit Absicht knapp und oberflächlich gehalten und nicht vollständig. Sie sollen auch fachlich weniger versierten Lesern einen groben Überblick erlauben und ein Grundverständnis für die Unterschiede der anschließend diskutierten Methoden vermitteln. Fachlich interessierte Leser verweisen wir auf [6] für tiefergehende Informationen.

2.1 Das GLMB- und das LMB-Filter

Eine GLMB-WDF $\boldsymbol{\pi}(\boldsymbol{X})$ mit Zustands-RFS $\boldsymbol{X}$ hat im Allgemeinen die folgende Form [3]:

$$\boldsymbol{\pi}(\boldsymbol{X}) = \Delta(\boldsymbol{X}) \sum_{c \in \mathbb{C}} w^{(c)}(\mathcal{L}(\boldsymbol{X})) \left[p^{(c)}(\cdot)\right]^{\boldsymbol{X}} . \tag{1}$$

Sehr vereinfacht gesagt modelliert diese alle möglichen Evolutionen eines Multiobjektzustands-RFS durch Hypothesen. Diese sind durch eine eindeutige Nummerierung gekennzeichnet und werden durch die Indexmenge $\mathbb{C}$ beschrieben. Die Summe in (1) bildet also diese möglichen Hypothesen ab, die, abhängig von der Realisierung, jeweils aus einer Eintrittwahrscheinlichkeit $w^{(c)}(\mathcal{L}(\boldsymbol{X}))$ und je einer WDF $p^{(c)}(\boldsymbol{X})$ je enthaltenem Objekt bestehen. Das Zustands-RFS $\boldsymbol{X}$ und dessen GLMB-WDF $\boldsymbol{\pi}(\boldsymbol{X})$ sind fettgedruckt um hervorzuheben, dass es sich um eine indizierte (*labeled*) Größe handelt. Dabei wird jedem Objekt eine eindeutige Kennzeichnung zugeordnet und man spricht dann häufig auch von geschätzten Trajektorien anstelle eines geschätzten Zustands.

Im Updateschritt des GLMB-Filters müssen nun alle Hypothesen mit den Sensormessungen aktualisiert werden. Hierzu erzeugt man sämtliche mögliche Assoziationen zwischen den beobachteten Objekten $\boldsymbol{X}$ des RFS mit den einzelnen Messungen des Mess-RFS Z. Hierbei wächst die Anzahl der modellierten Hypothesen exponentiell an, da dieser Schritt für alle prädizierten Hypothesen (künftig *Quellhypothesen* genannt) durchgeführt werden muss und deren Anzahl weder in der Prädiktion noch im Updateschritt verringert werden kann. Aus diesem Grund greift man typischerweise zu einem Trick und berechnet nur die k wahrscheinlichsten Hypothesen, was man durch Lösung des k-rangiges Zuordnungsproblem (engl.: k-ranked assignment problem) erreicht. Dieses wird für alle relevanten Quellhypothesen separat berechnet, was man als Truncation bezeichnet und einer Approximation entspricht. Im zweidimensionalen Fall kann dieses Zuordnungsproblem in polynomieller Zeit durch Murtys Algorithmus gelöst werden (kubische Komplexität) oder mithilfe des Gibbs-Samplers (quadratische Komplexität) angenähert werden [14]. Im mehrdimensionalen Fall dagegen ist die Lösung NP-schwer und ist damit im Allgemeinen nicht in endlicher Zeit berechenbar [12]. Abschließend muss jedoch in jedem Fall die Zahl der Hypothesen

insgesamt beschränkt werden, was man als Pruning bezeichnet. Hierfür sortiert man alle aktualisierten Hypothesen nach ihrem Gewicht und entfernt die unwahrscheinlichen.

Das LMB-Filter unterscheidet sich nur geringfügig davon, denn eine LMB-WDF stellt einen Spezialfall einer GLMB-WDF mit nur einer Komponente dar. Beschreibt nun also eine LMB-WDF den Multiobjektzustand zu einem Zeitpunkt k, für welche dann das Bayes-Update durchgeführt wird, so ergibt sich im Ergebnis eine GLMB-WDF. Damit wird klar, dass der Updateschritt des LMB-Filters prinzipiell dem des GLMB-Filters gleicht. Allerdings approximiert man anschließend die GLMB-WDF wieder als LMB-WDF. Kurz gesagt verliert man in diesem Schritt modellierte Abhängigkeiten zwischen den Objekten, behält aber die gesamte räumliche Information über die Objektzustände bei, indem man sie in eine Mixturdichte komprimiert. Im Ergebnis reduziert sich die Komplexität der WDF deutlich, da die Zahl der Hypothesen enorm abnimmt, weshalb das LMB-Filter im Allgemeinen deutlich recheneffizienter ist. Zusätzlich vereinfacht sich der Prädiktionsschritt, da die Objekte als unabhängig betrachtet werden.

2.2 RFS-basiertes Multisensor-Multiobjekttracking

Abbildung 1 zeigt die Strukturbilder der im Folgenden vorgestellten Methoden. Die Farben markieren den Typ der WDF [5], der in den jeweiligen Teilen genutzt wird. Dabei steht rot für GLMB- und blau für LMB-WDFs.

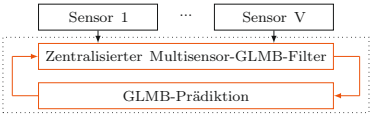
2.2.1 Ansätze mit Multisensor-Messmodell

Im allgemeinen Fall kann das Multisensor-Multiobjektupdate mit RFS-basierten Methoden mit der Multisensor-Bayesgleichung [5] berechnet werden:

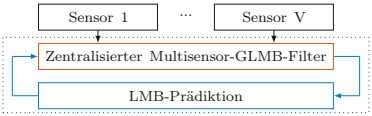
$$\boldsymbol{\pi}(\boldsymbol{X}|Z) \propto \boldsymbol{\pi}(\boldsymbol{X}) \cdot g(Z|\boldsymbol{X}). \tag{2}$$

Dabei bezeichnet $g(Z|\boldsymbol{X})$ das Multisensor-Multiobjektmessmodell, welches zumindest theoretisch auch für stark korrelierte Multisensormessungen immer gebildet werden kann [2]. Jedoch führt dies unausweichlich auf ein NP-schweres, mehrdimensionales Zuordnungsproblem ohne gemeingültige, technisch realisierbare Lösungsmethode [12]. Theoretisch betrachtet lassen sich sowohl das GLMB- als auch das LMB-Filter mit einem Multisensor-Messmodell darstellen, wie in den Abbildungen 1a und 1b dargestellt. In der Praxis sind nur Implementierungen mit stark begrenzter Sensoranzahl V in bestimmten Szenarien möglich, beispielsweise in nicht echtzeitkritischen Anwendungen wie der Nachbearbeitung von Datensätzen.

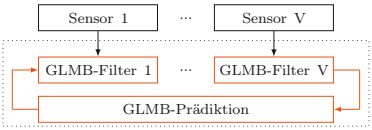
Das **GLMB-Filter mit Multisensor-Messmodell** wurde erstmals für zwei Sensoren in [15] präsentiert, kann aber theoretisch leicht um beliebige Sensoren erweitert werden [12, 17]. Zumindest theoretisch ist es das leistungsstärkste Filter, verliert diese Eigenschaft aber aufgrund der notwendigen Approximationen in der Praxis oft. Nicht wirklich anders stellt sich die Situation für das **LMB-Filter mit Multisensor-Messmodell** dar. Es ist, vermutlich deshalb, auch in der Literatur nicht beschrieben, aber in logischer Konsequenz aus dem LMB-Filter und dem GLMB-Filter mit Multisensor-Messmodell ableitbar. Anders ausgedrückt berechnet das LMB-Filter mit Multisensor-Messmodell die Prädiktion auf der LMB-WDF und nutzt für das Multisensorupdate das Update des GLMB-Filters mit Multisensor-Messmodell. Das Multisensorupdate muss also wie



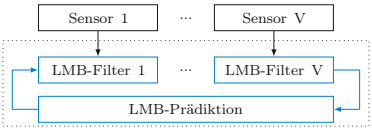
(a) GLMB-Filter mit Multisensor-Messmodell [15]



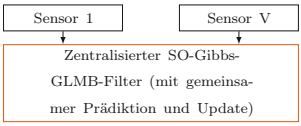
(b) LMB-Filter mit Multisensor-Messmodell



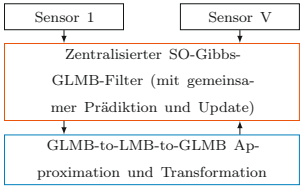
(c) IC-GLMB-Filter



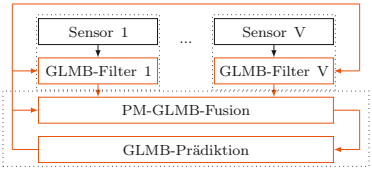
(d) IC-LMB-Filter



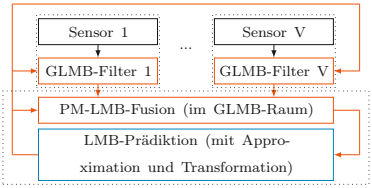
(e) SO-Gibbs-GLMB-Filter [12]



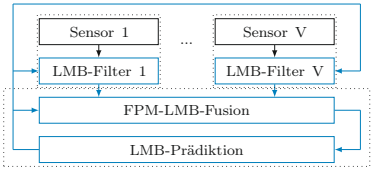
(f) SO-Gibbs-LMB-Filter [16]



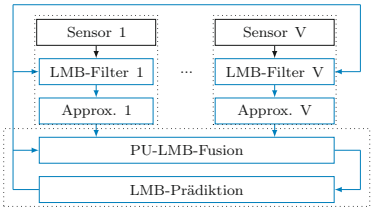
(g) PM-GLMB-Filter [17, 18]



(h) PM-LMB-Filter [19]



(i) FPM-LMB-Filter [20]



(j) PU-LMB-Filter [16]

Abbildung 1: Übersicht der zentralisierten Multisensor-Multiobjekttracking-Ansätze, wobei die Farben den Typ der verwendeten WDF auf der Schnittstelle oder innerhalb der Module indizieren: Rot steht für GLMB-WDFs und blau für LMB-WDFs.

üblich im GLMB-Raum durchgeführt werden. Auch dieser Filteransatz liefert zumindest theoretisch die bestmögliche Güte, ist aber ebenso nur in speziellen Fällen überhaupt anwendbar. Zusätzlich leiden beide Ansätze unter dem Problem, dass Lösungsmethoden für das mehrdimensionale Zuordnungsproblem nicht sinnvoll parallelisierbar sind.

2.2.2 Serielle Ansätze

Serielle Ansätze können angewendet werden, wenn sich das Multisensor-Messmodell $g(Z|\boldsymbol{X})$ als Produkt der einzelnen Sensormessmodelle schreiben lässt, also die Messungen unabhängig voneinander sind [12]. Diese Annahme kann in der Praxis häufig getroffen werden, da die Unabhängigkeit der Messungen einzig vom Messrauschen der Sensoren abhängt. Dann gilt für das Bayes-Update

$$\boldsymbol{\pi}(\boldsymbol{X}|Z) \propto \boldsymbol{\pi}(\boldsymbol{X}) \cdot \prod_{s=1}^{V} g^{(s)}(Z^{(s)}|\boldsymbol{X}), \tag{3}$$

wodurch die Klasse der Iterated-Corrector-Ansätze (IC-Ansätze) sowie der Multisensor-Gibbs-Sampler begründet wird.

Abbildung 1c und 1d zeigen die Filterstrukturen der **IC-Ansätze Iterated-Corrector-GLMB-Filter (IC-GLMB-Filter)** und **Iterated-Corrector-LMB-Filter (IC-LMB-Filter)**. Beide ähneln sich in ihrer Struktur insofern, als dass alle Sensormessungen sequentiell verarbeitet werden. Das IC-GLMB-Filter ist theoretisch betrachtet äquivalent zum GLMB-Filter mit Multisensor-Messmodell und damit optimal, wohingegen das IC-LMB-Filter das LMB-Filter mit Multisensor-Messmodell approximiert, indem die GLMB-zu-LMB-Approximation nach jedem Sensorupdate durchgeführt wird. Aber auch im IC-GLMB-Filter müssen die gängigen Approximationen nach jedem Update durchgeführt werden, da die Zahl der Hypothesen ansonsten zu stark ansteigen würde. Damit erfordern beide Ansätze maximal Lösungsalgorithmen für das zweidimensionale Zuordnungsproblem und sind technisch realisierbar. Allerdings wird die Filterleistung abhängig von der Verarbeitungsreihenfolge der Sensormessungen und sinkt im Allgemeinen erkennbar [12]. Zudem bleibt der Nachteil der fehlenden Parallelisierbarkeit der Messdatenverarbeitung bestehen.

Abgesehen von den parallelisierbaren Methoden, stellen das IC-GLMB- und insbesondere das IC-LMB-Filter die Implementierungen mit den kürzesten Rechenzeiten dar. Dies gilt umso mehr, wenn zur Lösung des Zuordnungsproblems Sampling-basierte Methoden, wie z.B. der Gibbs-Sampler mit quadratischer [14, 21] bzw. linearer [22] Komplexität, verwendet werden. Zudem ist die Umsetzung der IC-Varianten, beispielsweise auf Basis von Implementierung für Einzelsensoren, sehr einfach. Schließlich erlauben diese Ansätze auch die Anwendung von *Grouping*-Strategien, also der unabhängigen und parallelen Berechnung der Messupdates unabhängiger Objektgruppen [4, 23].

Auswertungen in [12, 17–20] zeigen aber eindrücklich die Schwierigkeiten der iterativen Ansätze, die vor allem bei ungünstiger Verarbeitungsreihenfolge kritische Folgen haben können. Deutlich besser schlägt sich das **Sub-optimal Gibbs-Sampling-based GLMB-Filter (SO-Gibbs-GLMB-Filter)** [12], dessen Filterstruktur in Abbildung 1e abgebildet ist. Trotz eines Multisensorupdates zählen wir es zu den seriellen Ansätzen, da es auf derselben Annahme unabhängiger Sensormessungen beruht. Zusätzlich verwendet dieses Filter einen kombinierten Prädiktions- und Updateschritt (engl. *joint-prediction-and-update*), der das Filter sehr leistungsfähig und effizient hinsichtlich der benötigten Hypothesen zur Abbildung der mehrdeutigen Messassoziation macht.

Gegenüber den iterativen Varianten benötigt das SO-Gibbs-GLMB-Filter allerdings eine deutlich höhere Rechenzeit, was insbesondere für die ebenfalls bekannte optimale Variante, das **Optimal Gibbs-Sampling-based GLMB-Filter (O-Gibbs-GLMB-Filter)** gilt, das aber ebenfalls in die Kategorie der NP-schweren Probleme fällt [12] und deshalb hier nur am Rand erwähnt wird. Es ist bisher auch nicht bekannt, inwiefern Grouping-Strategien mit dem Multisensor-Gibbs-Ansatz kombiniert werden können oder ob eine Multisensorvariante des Gibbs-Samplers mit linearer Komplexität möglich ist. Beide Erweiterungen dürften bei erfolgreicher Realisierung die Bedeutung des Ansatzes aufgrund der zu erwartenden Beschleunigung der Rechenzeit deutlich erhöhen. Bekannt dagegen ist die entsprechende Anwendung auf LMB-WDFs, also das **Sub-optimal Gibbs-Sampling-based LMB-Filter (SO-Gibbs-LMB-Filter)**, dessen Struktur in Abbildung 1f dargestellt ist [16].

2.2.3 Parallelisierbare Ansätze

Ebenfalls unter der Bedingung unabhängiger Sensormessungen kann die Bayes-Updategleichung geschickt so umgestellt werden, sodass ein paralleles und unabhängiges Update der Sensormessungen möglich wird. Diese Formulierung ist bekannt als Bayes-Parallel-Combination-Rule (BPCR) [24] und lautet für RFSs

$$\boldsymbol{\pi}(\boldsymbol{X}|Z) \propto \left(\boldsymbol{\pi}\left(\boldsymbol{X}\right)\right)^{1-V} \prod_{s=1}^{V} \boldsymbol{\pi}^{(s)}\left(\boldsymbol{X}|Z^{(s)}\right). \tag{4}$$

Auf ihr basiert beispielsweise auch das Information-Matrix-Fusion-Filter (IMF-Filter) [2], weshalb wir die folgenden Filter auch als Multisensor-Informationsfilter oder Multisensor-IMF-Äquivalent bezeichnen. Ganz generell zeichnet sich die BPCR durch die Parallelisierbarkeit der Sensorupdates und damit durch eine mögliche verteilte Implementierung aus. Damit ist auch ein Intellectual-Property-Schutz (IP-Schutz) der Messmodelle möglich. Allerdings kann die BPCR nur unter zwei Bedingungen angewendet werden: Es wird erstens eine globale Fusionsinstanz benötigt, die allen beteiligten Sensoren eine global gültige Prädiktion bereitstellt. Und zweitens muss das Produkt der lokal aktualisierten WDFs durch das $(V-1)$-fache Produkt der prädizierten WDF teilbar sein, was die Anwendung der BPCR auf GLMB- und LMB-WDFs herausfordernd macht. Dennoch konnten sowohl die Forschungsgruppe um Robertson [16] als auch wir in den vergangenen Jahren entsprechende Lösungen für beide Dichten finden.

Den ältesten der vier Ansätze stellt das in Abbildung 1g skizzierte **Product-Multi-Sensor-GLMB-Filter (PM-GLMB-Filter)** [17, 18] dar. Bei diesem berechnen die lokalen Filter ihr Sensorupdate jeweils im GLMB-Raum auf Basis der prädizierten globalen GLMB-WDF. Die Filterformulierung kann grundsätzlich exakt hergeleitet werden, weshalb das Filter neben dem GLMB-Filter mit Multisensor-Messmodell die einzige Bayes-optimale Multisensor-Implementierung darstellt. Jedoch ignoriert man in der Praxis sinnvollerweise die Kreuzkovarianzen der einzelnen Sensorinnovationen, da deren Berechnung numerisch instabil ist [18].

Dennoch zeigt das Filter in Simulationen sehr gute Leistungen (wobei ein direkter Vergleich bisher nur mit dem IC-GLMB- und dem IC-LMB-Filter durchgeführt wurde) und stellt aufgrund der Parallelisierbarkeit die schnellste bekannte Multisensor-GLMB-Filtervariante dar. Nachteilig ist aber die in der Praxis recht komplizierte Parametrisierung der Truncation- und Pruningparameter, die deutliche Auswirkungen auf die Filterleistung

haben. Insbesondere zeigt das PM-GLMB-Filter Schwierigkeiten mit der Geburt neuer Objekte, wenn die zentrale Instanz ein Pruning auf die prädizierte WDF anwendet. Im Gegenzug zu den vorigen Filtern ist dies aber in der Praxis oft notwendig.

Das **Product-Multi-Sensor-LMB-Filter (PM-LMB-Filter)** [19], dessen Strukturdiagramm in Abbildung 1h dargestellt ist, bildet eine mögliche Alternative auf Basis der LMB-WDFs ab. In diesem Fall wird die Prädiktion vereinfacht im LMB-Raum durchgeführt, aber die Messupdates und die Filterfusion finden im GLMB-Raum statt. Dies entspricht mathematisch betrachtet dem LMB-Filter mit Multisensor-Messmodell, und zeigt dementsprechend gute Ergebnisse in der Praxis. Außerdem ist das PM-LMB-Filter noch einmal deutlich schneller als das PM-GLMB-Filter. Einschränkend muss allerdings erwähnt werden, dass die Rechenzeiten des IC-LMB-Filters nur dann erreicht werden, wenn dieser kein Grouping verwendet, da ein Grouping für die parallelisierbaren Methoden noch nicht entwickelt wurde, wenngleich es zumindest für die LMB-Varianten möglich erscheint.

Im Gegensatz zu den beiden vorigen Varianten führt das **Parallel-Update LMB-Filter (PU-LMB-Filter)** [16] den Filterzyklus vollständig im LMB-Raum durch (wobei wir die kurzzeitige Konversion in den GLMB-Raum für das Einzelsensorupdate hier der Einfachheit halber vernachlässigen), wie in Abbildung 1j dargestellt. Damit müssen anstelle der Hypothesen nur die jeweiligen Objekte mit demselben Label fusioniert werden, deren Anzahl üblicherweise deutlich geringer ist als die der Hypothesen bei der Fusion im GLMB-Raum. Allerdings müssen die lokal aktualisierten räumlichen WDFs der Objekte vor ihrer Fusion approximiert werden um die Division durch die prädizierte WDF durchführen zu können. Dabei müssen aus den normalverteilten Mixturdichten Normalverteilungen geschätzt werden. Im einfachsten Fall nimmt man dazu die wahrscheinlichste Mixturkomponente oder das gewichtete Mittel, die Autoren stellen aber auch komplexere Methoden zur Approximation zur Verfügung.

Das neueste und schnellste Filter, ist das **Fast-Product-Multi-Sensor-LMB-Filter (FPM-LMB-Filter)** [20], dessen Struktur in Abbildung 1i dargestellt ist. Es baut direkt auf der Filterformulierung des PU-LMB-Filters auf, löst aber das Divisionsproblem, indem zur Fusion der räumlichen Dichten auf das IMF zurückgegriffen wird. So bleibt auch das FPM-LMB-Filter vollständig im LMB-Raum und ist deshalb wie das PU-LMB-Filter noch einmal deutlich weniger rechenintensiv als die anderen parallelisierbaren Methoden.

2.3 Diskussion

In der Praxis spielen die Ansätze mit Multisensor-Messmodell aufgrund ihrer rechentechnischen Komplexität eigentlich keine Rolle. Die seriellen Verfahren dagegen sind in Multisensor-Anwendungen quasi Standard und profitieren von ihrer programmatischen Einfachheit sowie ihrer relativ guten Geschwindigkeit und ihrer oft ausreichenden Leistungsfähigkeit.

Werden jedoch hohe Echtzeit- oder Leistungsanforderungen gestellt, kommen die alternativen Ansätze ins Spiel, und hier stellen die Product-Multi-Sensor-Ansätze (PM-Ansätze) einen guten Kompromiss dar. Ihre Eigenschaften erhalten sie zu großen Teilen aus ihrem differenzierten Herrsche-und-Teile-Ansatz, durch den sie sich von den anderen Varianten unterscheiden. Einerseits ermöglicht der Ansatz eine echte Multisensorfusion, die gleichzeitig alle Sensormessungen berücksichtigt und dabei enorm schnell ist. Dies erkauft man sich allerdings dadurch, dass die Truncation der lokalen Updates isoliert und

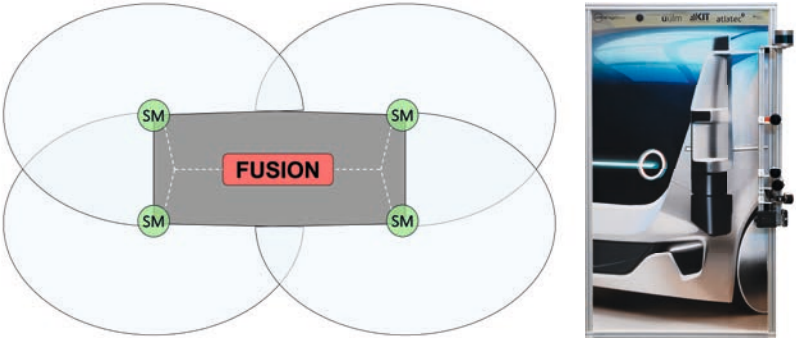


Abbildung 2: Perzeptionskonzept (links) und SM (rechts) aus UNICAR*agil*

unabhängig durchgeführt wird. Allerdings sieht die Situation bei den IC-Filtern sehr ähnlich aus, die bei der Truncation ebenfalls nur begrenzte Informationen zur Verfügung haben, genauso wenig auf ein Multisensorupdate bauen können und nach jedem Sensorupdate ein Pruning benötigen. Einzig das SO-Gibbs-GLMB- und das SO-Gibbs-LMB-Filter haben hier Vorteile, die sich jedoch in einer deutlich höheren Rechenzeit niederschlagen. Andererseits ermöglichen die parallelisierbaren Filter als einzige eine verteilte Implementierung und den Schutz geistigen Eigentums der Sensorhersteller an deren Messmodellen.

Zusammengefasst lässt sich also sagen, dass die PM-Ansätze enorme Vorteile bieten und dabei kaum praktisch relevante Nachteile haben.

3 Offene Forschungsfragen und zukünftige Arbeiten

Gegenüber den anderen Methoden sind die parallelisierbaren Ansätze noch relativ jung und entsprechend wenig verbreitet. Einer solchen Verbreitung stehen neben fehlender Bekanntheit allerdings auch noch einige offene Forschungsfragen im Weg, die bisher noch nicht abschließend gelöst werden konnten. Sie sollen im Folgenden anhand zweier Multisensor-Anwendungsfälle aus unseren Forschungsprojekten erläutert und potenzielle Lösungsstrategien aufgezeigt werden.

3.1 Modulares Sensormodulkonzept aus UNICAR*agil*

Modularität und Redundanz haben im Fahrzeugkonzept der im Rahmen des Projekts UNICAR*agil* entwickelten Fahrzeuge einen großen Stellenwert [10]. Auch die Perzeption folgt diesen Projektzielen und besteht, wie in Abbildung 2 dargestellt, aus vier unabhängigen Sensormodulen (SMs) und einer zentralen Fusionseinheit. Es handelt sich damit um eine zentralisierte Netzwerktopologie, für welche die PM-Filter ideal geeignet wären. In den realisierten Fahrzeugen kommt allerdings eine suboptimale Track-to-Track-Fusion zum Einsatz, da einige offene Forschungsfragen bisher nicht beantwortet wurden.

Da die Sensoren auf einem beweglichen Fahrzeug montiert sind, benötigt man im Multiobjekttracking ein **adaptives Geburtenmodell**. Leider ist das gängige Modell für LMB- und GLMB-Filter sensorspezifisch und hauptsächlich für die IC-Ansätze geeignet [4].

Bei den PM-Ansätzen dagegen generieren alle Sensoren gleichzeitig mögliche Geburtskandidaten, wobei sich mit wachsender Sensoranzahl mit an absoluter Sicherheit grenzender Wahrscheinlichkeit einige davon doppeln. Um die Filtergüte nicht zu sehr zu schmälern benötigt es geeignete Methoden zur Vermeidung solcher Mehrfachgeburten. Eine potentiell vielversprechende Option wäre die Verwendung LMB-basierter Track-to-Track Methoden, welche diese Geburtskandidaten kombinieren. Im besten Fall würde man dann nicht nur die Mehrfachgeburt verhindern, sondern sogar von der Multisensorinformation profitieren.

Zusätzlich stellen aber auch die verbauten Radarsensoren eine Herausforderung dar. Prinzipbedingt sind sie nicht exakt zeitlich synchronisierbar, im Gegensatz zu den Kameras und Lidarsensoren. Um entsprechende Sensordaten fusionieren zu können, wäre eine Möglichkeit, die Sensordaten auf die synchronisierten Zeitpunkte des Filter-Updates zu prädizieren. Allerdings ist dies aufgrund der Unsicherheiten in den Messdaten sowie deren Unvollständigkeit in der Praxis oft schwierig. Alternativ könnte eine Fusion sogenannter Out-of-Sequence-Messungen realisiert werden. Hierzu existieren bisher jedoch noch keine Lösungen.

Letztlich leiden außerdem sowohl das PM-GLMB- als auch das PM-LMB-Filter in der Praxis unter der Vielzahl an zu berücksichtigenden Hypothesen. Beispielsweise führt die Vernachlässigung einer Quellhypothese durch nur einen einzigen Sensor in der Fusion zur vollständigen Auslöschung aller darauf aufbauenden Hypothesen. Gerade mit steigender Objekt- und Sensoranzahl nimmt die Wahrscheinlichkeit solcher Auslöschungen prinzipiell zu, und es braucht Methoden zur Steuerung. In unseren Implementierungen erreichen wir dies aktuell nur unzureichend durch eine Kombination mehrerer Strategien: Einerseits parametrisieren wir die Truncation der Sensoren so, dass deutlich mehr Hypothesen in den lokalen Updates berechnet werden als Hypothesen das Pruning nach der Fusion überleben. Andererseits beschränken wir oft auch die Anzahl der Hypothesen nach der Prädiktion durch einen zusätzlichen Pruningschritt. Und letztlich zwingen wir die lokalen Filter dazu, einen bestimmten Anteil ihrer verfügbaren Rechenzeit so auf die Quellhypothesen zu verteilen, dass möglichst alle mindestens einmal berücksichtigt werden. Dieser Ansatz ist natürlich rechentechnisch und leistungsmäßig fraglich und sollte verbessert werden, beispielsweise durch eine **verbesserte Koordination der zu berücksichtigenden Hypothesen, (also einer koordinierten Truncation)**, in den lokalen Filtern.

3.2 Multiobjekttracking für verteilte Infrastruktursensorsysteme

Insbesondere im unübersichtlichen und unsignalisierten innerstädtischen Verkehr tun sich automatisierte Fahrzeuge derzeit noch sehr schwer [9]. Im Projekt Lokales Umfeldmodell für das kooperative, automatisierte Fahren in komplexen Verkehrssituationen (LUKAS) nutzen wir ein Infrastruktursensorsystem zur Unterstützung vernetzter Fahrzeuge in unserer Versuchsanlage in Ulm-Lehr. Abbildung 3 zeigt auf der linken Seite eine Karte der Kreuzung und auf der rechten Seite zwei der exponiert angebrachten Kamerasensoren in der nördlichen Kreuzungszufahrt. Eine Besonderheit stellt dabei das verwendete Referenzpunktmessmodell zur Vermeidung systematischer Messfehler dar [11], bei dem Sensoren flexibel diejenigen Messdaten übermitteln können die auch real inferierbar sind. Typischerweise ist dies die Position eines Fahrzeugeckpunkts (sogenannter Referenzpunkt) und gegebenenfalls noch die Objektausdehnung in eine oder zwei Richtungen. Insgesamt besteht das System aus mehreren Kamera-, Radar- und Lidarsensoren, deren Messungen drahtlos an einen Multiaccess-Edge-Computing-Server (MEC-Server) gesendet und dort fusioniert werden. Folglich

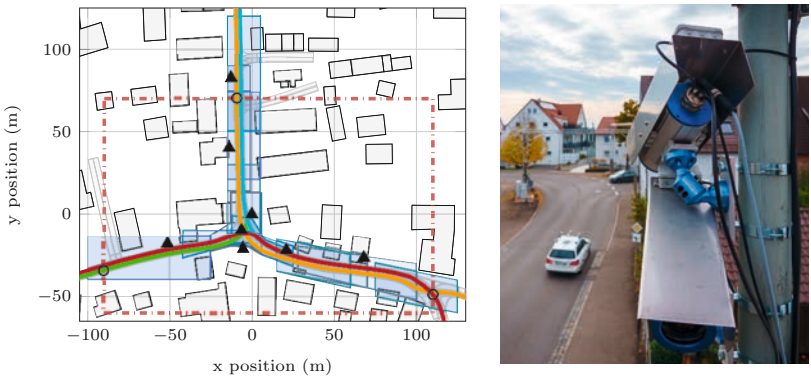


Abbildung 3: Simuliertes Verkehrsszenario an der Infrastrukturanlage in Ulm-Lehr mit sieben Fahrzeugen (farbige Linienzüge), 14 Sensoren (deren Sichtbereich blau dargestellt sind), allen Sensormasten (markiert durch ▲), den statischen Geburtsorten (markiert durch ○) und dem ausgewerteten Bereich (rot gestrichelter Bereich) [25, 26].

besitzt auch dieses System eine zentralisierte Netzwerktopologie, die sich hervorragend für die PM-Filter eignet.

Im Gegensatz zum UNICAR*agil*-Fahrzeug ist deren Anwendung mit statischem Geburtenmodell in einer Digital-Twin-Simulation [27] hier aufgrund der statischen Sensorpositionierung möglich. Im gezeigten Szenario simulieren wir ausschließlich mehrere Kamerasensoren an den mit ▲ markierten Montagepositionen. Die jeweiligen Sensorsichtbereiche sind blau hervorgehoben und die rot umrandete Zone markiert den ausgewerteten Bereich. In der Simulation beschränken sich die Sensoren auf die Detektion der Eckpunkte, die sie mit einer normalverteilten und unabhängigen Messunsicherheit σ generieren. Damit fällt dem Multiobjekttracker die Aufgabe der Ausdehnungsschätzung der Objekte zu, was möglich ist, solange mehrere Sensoren dasselbe Objekt an verschiedenen Referenzpunkten beobachten. Zudem generieren die Sensoren durchschnittlich einen zusätzlichen Falschalarm pro Zeitschritt und übersehen Fahrzeuge mit einer Wahrscheinlichkeit von 10 %.

Dieses Szenario haben wir in einer Monte-Carlo-Simulation mit 100 Wiederholungen ausgeführt und die Ergebnisse der vier Filtervarianten IC-GLMB-, IC-LMB-, PM-GLMB- und PM-LMB-Filter mit der OSPA$^{(2)}$-Metrik [28] ausgewertet. Die Ergebnisse sind im oberen Diagramm der Abbildung 4 für die Messunsicherheiten $\sigma^2 = 0.5\text{m}^2$ und $\sigma^2 = 2\text{m}^2$ dargestellt. Im unteren Diagramm sieht man die Kardinalitätsschätzung für den zweiten Fall. Hierbei zeigen sich einige Besonderheiten. Entgegen der gängigen Annahme schlagen sich die LMB-Filter besser als die GLMB-Filter. Dies liegt daran, dass die Anzahl der eigentlich benötigten Hypothesen in den GLMB-Filtern derart hoch ist, dass eine Ausführung rechentechnisch nicht mehr realisierbar und schon gar nicht echtzeitfähig ist. Daher muss auf eine geringere Anzahl von Hypothesen reduziert werden. Konsequenterweise steigt mit steigender Messunsicherheit dann auch der Leistungsunterschied zwischen den beiden Filtertypen, weil die Messung-zu-Track-Assoziation mehrdeutiger wird.

Betrachtet man aber jeweils die Gruppe der GLMB-und LMB-Filter unabhängig voneinander fällt auf, dass die PM-Variante immer besser als die jeweilige IC-Variante abschneidet. Dies verdanken die PM-Filter ihrer überlegenen Kardinalitätsschätzung auf-

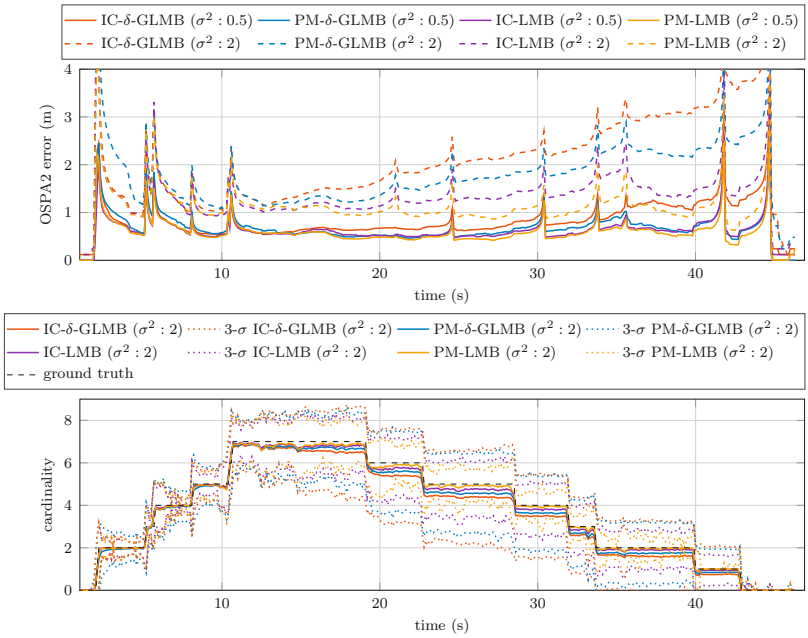


Abbildung 4: Optimal-Sub-pattern-Assignment$^{(2)}$ (OSPA$^{(2)}$)-Fehler aller Filter für beide simulierten Messunsicherheiten (oben), sowie die geschätzte Kardinalität der Filter für den Fall mit höherem Messrauschen (unten), jeweils gemittelt über 100 Simulationsdurchläufe.

grund des echten Multisensorupdates, wie man im unteren Diagramm deutlich sehen kann. Im Verlauf können die PM-Filter die Kardinalität immer etwas präziser schätzen, wobei die großen Fehler natürlich daher rühren, dass aufgrund des statischen Geburtenmodells einmal verlorengegangene Tracks dauerhaft verloren sind.

Anhand dieses Beispiels und der Ergebnisse lassen sich aber weitere noch offene Forschungsfragen ableiten, die in naher Zukunft beantwortet werden sollten. So ist beispielsweise das Multisensor-Update des PM-Filters aufgrund der idealen Parallelisierbarkeit sehr schnell berechenbar, aber in diesem Beispielszenario dennoch nicht ganz echtzeitfähig. Der Grund: Für die IC-Filter gibt es sogenannte **Groupingansätze**, wobei anhand der Unsicherheiten der Sensormessungen die WDF in unabhängige Teil-WDFs aufgeteilt wird. Dies führt, fast verlustfrei, zu einer enormen Verkürzung der Rechenzeit, da die Größe des Assoziationsproblems deutlich reduziert wird. Eine Erweiterung auf den Multisensor-Fall wäre folglich sehr attraktiv und könnte zumindest für das FPM-LMB- als auch das PU-LMB-Filter aufgrund der modellierten Objektunabhängigkeit möglich sein.

Anhand entsprechender effizienter Implementierungen wäre abschließend vor allem ein **fairer Vergleich der unterschiedlichen PM-Filteransätze sowohl untereinander, als auch gegenüber den Multisensor-Gibbs-Filtern** interessant. Denn es ist bisher nicht bekannt wie die Ansätze gegeneinander hinsichtlich verschiedener Kriterien, wie die Schätzgüte, Rechenzeit, etc., abschneiden. Hierfür wären allerdings vergleichbare und

rechentechnisch effiziente Implementierungen notwendig, die es leider bisher weder in der Open-Source-Szene, noch bei uns am Institut gibt.

Aufgrund der **numerisch instabilen Berechnungsvorschrift für die Innovationskovarianz in den PM-Filtern** müssen diese approximiert werden [18]. Praktisch vernachlässigt man diese Werte, da sie nachweislich kaum Relevanz haben. Dennoch könnte ihre Berechnung in spezifischen Situationen von Vorteil sein, weshalb die Entwicklung einer geschlossenen Berechnungsvorschrift vorteilhaft wäre.

Mit den Poisson Multi-Bernoulli Mixture (PMBM)-Filtervarianten für Trajektorien (siehe [29] für einen Überblick) existieren noch weitere vielversprechende RFS-basierte Filtertypen in der einschlägigen Literatur. Diese profitieren wie das LMB-Filter von einer vereinfachten Prädiktion, können aber dennoch Objektabhängigkeiten modellieren und werden in praktischen Untersuchungen oft als dem GLMB-Filter überlegen dargestellt [29]. Eine Anwendung der PMBM-WDF auf die BPCR und die entsprechende Entwicklung des **Product-Multi-Sensor-Poisson-Multi-Bernoulli-Mixture-Filters (PM-PMBM-Filters)** erscheint also vielversprechend.

Durch die PM-Filter gibt es nun erstmals die Möglichkeit, die Update-Berechnung im Multisensor-Multiobjekt-Fall parallel durchzuführen. Möchte man diese jedoch in einem Netzwerk verteilt berechnen lassen, müssen die Verbindungen breitbandig genug sein, um die jeweiligen WDFs zu übertragen. Hier stößt man sowohl bei GLMB- als auch bei LMB-WDFs derzeit noch schnell an praktische Grenzen, insbesondere in drahtlosen Netzwerken. Entsprechend müssen **geeignete Komprimierungsverfahren für WDFs** gefunden werden und der jeweilige Einfluss auf die Filterleistung untersucht werden.

4 Fazit

In diesem Artikel haben wir die aktuell wissenschaftlich beschriebenen RFS-basierten Multisensoransätze für GLMB- und LMB-WDFs vorgestellt und anhand ihrer Vor- und Nachteile verglichen. Im Ergebnis zeigt sich, dass die diskutierten PM-Filter aufgrund ihrer Laufzeit und Leistungsfähigkeit eine ernstzunehmende Alternative für die gängigen IC-, aber auch die Multisensor-Gibbs-Varianten darstellen. Allerdings stehen mit den Themen wie Grouping und adaptive Geburt noch gewichtige offene Fragen im Raum, deren Beantwortung wir in der kommenden Zeit angehen und auch andere Wissenschaftler dazu motivieren möchten. Denn im Multiobjektfall treten die Vorteile des parallelen Multisensor-Updates umso mehr zu Tage, die insbesondere in Anwendungen wie dem automatisierten Fahren mit vielen zu beobachtenden Verkehrsteilehmern zum Tragen kommen.

Danksagung

Diese Arbeit wurde finanziell vom Bundesministerium für Wirtschaft und Klimaschutz (BMWK) im Rahmen des Programms „Hoch- und vollautomatsiertes Fahren in anspruchsvollen Fahrsituationen“ (Projekt LUKAS, FKZ 19A16010I) und vom Bundesministerium für Bildung und Forschung (BMBF) (Projekt UNICAR*agil*, FKZ 16EMO0290) unterstützt.

Literatur

[1] H. Winner, S. Hakuli, F. Lotz, and C. Singer, Eds., *Handbuch Fahrerassistenzsysteme*. Springer Fachmedien, Wiesbaden, 2015.

[2] Y. Bar-Shalom, P. K. Willett, and X. Tian, *Tracking and Data Fusion: A Handbook of Algorithms*. Connecticut: YBS Publishing, 2011.

[3] B. T. Vo and B. N. Vo, "Labeled random finite sets and multi-object conjugate priors," *IEEE Trans. Signal Process.*, vol. 61, no. 13, pp. 3460–3475, 2013.

[4] S. Reuter, B. T. Vo, B. N. Vo, and K. Dietmayer, "The Labeled Multi-Bernoulli Filter," *IEEE Trans. Signal Process.*, vol. 62, no. 12, pp. 3246–3260, 2014.

[5] R. P. S. Mahler, *Statistical Multisource-Multitarget Information Fusion*. Artech House, Norwood, 2007.

[6] ——, *Advances in Statistical Multisource-Multitarget Information Fusion*. Artech House, Norwood, 2014.

[7] ——, "Exact closed-form multitarget bayes filters," *Sensors*, vol. 19, no. 12, 2019.

[8] F. Kunz et al., "Autonomous driving at Ulm University: A modular, robust, and sensor-independent fusion approach," in *Proc. IEEE Intell. Veh. Symp. (IV)*, 2015.

[9] M. Buchholz et al., "Handling occlusions in automated driving using a multiaccess edge computing server-based environment model from infrastructure sensors," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 3, pp. 106–120, 2022.

[10] ——, "Automation of the UNICARagil vehicles," *Open Access Repositorium der Universität Ulm*, 2020, doi: 10.18725/OPARU-34024.

[11] M. Herrmann, A. Piroli, J. Strohbeck, J. Müller, and M. Buchholz, "LMB filter based tracking allowing for multiple hypotheses in object reference point association," in *Proc. IEEE Int. Conf. Multisens. Fusion Integr. (MFI)*, 2020, pp. 197–203.

[12] B. N. Vo, B. T. Vo, and M. Beard, "Multi-Sensor Multi-Object Tracking with the Generalized Labeled Multi-Bernoulli Filter," *IEEE Trans. Signal Process.*, vol. 67, no. 23, pp. 5952–5967, 2019.

[13] K. Da, T. Li, Y. Zhu, H. Fan, and Q. Fu, "Recent advances in multisensor multitarget tracking using random finite set," *Frontiers of Information Technology & Electronic Engineering*, vol. 22, no. 1, pp. 5–24, 2021-01.

[14] B.-N. Vo, B.-T. Vo, and H. G. Hoang, "An efficient implementation of the generalized labeled multi-Bernoulli filter," *IEEE Trans. Signal Process.*, vol. 65, no. 8, pp. 1975–1987, 2017.

[15] B. Wei, B. Nener, W. Liu, and L. Ma, "Centralized multi-sensor multi-target tracking with labeled random finite sets," in *Proc. Int. Conf. Control, Autom. Inf. Sci. (ICCAIS)*, 2016, pp. 82–87.

[16] S. Robertson, C. van Daalen, and J. du Preez, "Efficient approximations of the multi-sensor labelled multi-Bernoulli filter," *Signal Processing*, vol. 199, 2021.

[17] M. Herrmann, C. Hermann, and M. Buchholz, "Distributed implementation of the centralized generalized labeled multi-Bernoulli filter," *IEEE Trans. Signal Process.*, vol. 69, pp. 5159–5174, 2021.

[18] M. Herrmann, T. Luchterhand, C. Hermann, T. Wodtko, J. Strohbeck, and M. Buchholz, "Notes on the product multi-sensor generalized labeled multi-Bernoulli filter and its implementation," in *Proc. Int. Conf. Inf. Fusion (FUSION)*, 2022.

[19] M. Herrmann, T. Luchterhand, C. Hermann, and M. Buchholz, "The product multi-sensor labeled multi-Bernoulli filter," in *Proc. Int. Conf. Inf. Fusion (FUSION)*, 2023.

[20] C. Hermann, M. Herrmann, T. Griebel, and M. Buchholz, "The fast product multi-sensor labeled multi-Bernoulli filter," in *Proc. Int. Conf. Inf. Fusion (FUSION)*, 2023.

[21] S. Reuter, A. Danzer, M. Stubler, A. Scheel, and K. Granstrom, "A fast implementation of the Labeled Multi-Bernoulli filter using Gibbs sampling," in *2015 IEEE Intell. Veh. Symp. (IV)*, 2017, pp. 765–772.

[22] C. Shim, B.-T. Vo, B.-N. Vo, J. Ong, and D. Moratuwage, "Linear complexity gibbs sampling for generalized labeled multi-bernoulli filtering," *IEEE Trans. Signal Process.*, vol. 71, pp. 1981–1994, 2023.

[23] M. Beard, B. T. Vo, and B. Vo, "A solution for large-scale multi-object tracking," *IEEE Trans. Signal Process.*, vol. 68, pp. 2754–2769, 2020.

[24] R. Kruse, E. Schwecke, and J. Heinsohn, *Uncertainty and Vagueness in Knowledge Based Systems*. Springer Berlin, Heidelberg, 1991.

[25] OpenStreetMap contributors. (2017) Planet dump. Retrieved from https://planet.osm.org.

[26] F. Poggenhans et al., "Lanelet2: A high-definition map framework for the future of automated driving," in *Proc. IEEE Intell. Transp. Syst. Conf.*, 2018.

[27] J. Strohbeck, J. Muller, A. Holzbock, and M. Buchholz, "DeepSIL: A software-in-the-loop framework for evaluating motion planning schemes using multiple trajectory prediction networks," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, 2021.

[28] M. Beard, B. T. Vo, and B.-N. Vo, "OSPA(2): Using the OSPA metric to evaluate multi-target tracking performance," in *Proc. Int. Conf. Control, Autom. Inf. Sci. (ICCAIS)*, 2017, pp. 86–91.

[29] A. F. Garcia-Fernandez, L. Svensson, J. L. Williams, Y. Xia, and K. Granstrom, "Trajectory Poisson multi-Bernoulli filters," *IEEE Trans. Signal Process.*, vol. 68, pp. 4933–4945, 2020.