FunKI

Funkkommunikation mit Künstlicher Intelligenz

Entwurf und Implementierung von Hardware-Architekturen für KI-Algorithmen in Sende- und Empfängerstrukturen von Funkkommunikationssystemen

FunKI

Sachbericht zum Verwendungsnachweis

Teil 1: Kurzbericht

Förderkennzeichen: 16KIS1185

Förderzeitraum: 15.05.2020 bis 14.05.2023

GEFÖRDERT VOM

Bundesministerium
für Bildung
und Forschung

CREONIC ip cores & system solutions

Deutsches Forschungszentrum für Künstliche Intelligenz GmbH

intel

MOTIUS WE R&D.

NOKIA

RP TU Rheinland-Pfälzische Technische Universität Kaiserslautern Landau

Universität Bremen

University of Stuttgart Germany

1. Aufgabenstellung und Ziele

Ziel des Projekts FunKI war es eine grundlegende Untersuchung von KI-getriebenen Technologien für die Funkkommunikation durchzuführen, die sich auf die 5. Mobilfunkgeneration (5G) und deren Weiterentwicklung konzentriert. Im Speziellen wurden Konzepte zur Verbesserung des Modellverständnisses basierend auf Messkampagnen erarbeitet, KI-basierte Methoden zur Parameterschätzung entwickelt, die datengetriebene Optimierung von Sende- und Empfangsverfahren durchgeführt und KI-basierte Algorithmen in Hardware implementiert.

Der Lehrstuhl „Entwurf Mikroelektronischer Systeme“ der RPTU konzentrierte sich in diesem Projekt auf das enge Zusammenspiel zwischen Algorithmen und deren effizienter Umsetzung auf fortschrittlicher ASIC-Technologie und FPGAs. Dabei spielte der Trade-Off zwischen nachrichtentechnischer Performanz, Algorithmen und Implementierungseffizienz eine wichtige Rolle, was als Entwurfsraumexploration bezeichnet wird. Die Aufgabe der RPTU in Zusammenarbeit mit Partnern, die ihren Schwerpunkt auf die nachrichtentechnische und algorithmische Seite legten, bestand darin, effiziente NN-Accelerator-Architekturen zu entwickeln, die hohe Durchsatzraten, geringen Latenz und hohe Energieeffizienz aufweisen.

2. Ablauf des Vorhabens

Das Projekt FunKI ist am 15.05.2020 gestartet und lief bis zum 14.05.2023. In der ersten Phase des Projekts lag der Fokus der RPTU auf der Auswahl von zwei Basiskomponenten zur effizienten Hardware Implementierung. Diese Auswahl erfolgte in enger Zusammenarbeit mit den Projektpartnern UST und UB. Im weiteren Verlauf wurde eine umfassende Entwurfsraumexploration beider Acceleratoren durchgeführt, um Parameter wie Speicheranforderung und nachrichtentechnische Performanz zu evaluieren. Basierend auf der Entwurfsraumexploration wurden parametrisierbare Hardware-Architekturtemplates erstellt um verschiedene Trade-Offs wie die Quantisierung zu explorieren. Im Anschluss erfolgten prototypische Implementierungen der Acceleratoren, sowohl auf Virtual Silicon als auch auf FPGAs. In einem nächsten Schritt wurden die neuartigen Implementierungen mit klassischen Ansätzen hinsichtlich der Implementierungskomplexität verglichen. In Kooperation mit Creonic wurde abschließend ein Demonstrator entwickelt, der die Ergebnisse des Projekt visualisiert und für ein breiteres Publikum zugänglich macht.

3. Ergebnisse

Innerhalb von AP 4 wurden in Zusammenarbeit mit den Partnern Creonic sowie der Universität Stuttgart und Bremen folgende Transceiver Komponenten zur Implementierung ausgewählt:

1) **LDPC-Decoder**: Die Dekodierung von LDPC-Codes basiert auf einem iterativen Austausch von Nachrichten über die Kanten Tanner-Graphen, der aus der H-Matrix des Codes abgeleitet ist. Die Anzahl der Nachrichten ist durch die Topologie des Graphen vorgegeben, wobei die Quantisierung als wichtigster Optimierungsparameter für die Implementierungseffizienz verbleibt. Aus diesem Grund wurde die Information-Bottleneck-Methode (IBM) zur effizienten Quantisierung von Kanten-Nachrichten untersucht.
2) **Autoencoder**: Im Kontext von Kommunikationssystemen bezeichnet Autoencoder (AE) ein System, das Teile des traditionellen Senders und Empfängers durch künstliche neuronale Netzwerke (ANNs) ersetzt. Dadurch kann das System, weitestgehend unabhängig vom Kanalmodell, global optimiert werden, wodurch die Nachrichtentechnische Performanz im Vergleich zu konventionellen Ansätzen gesteigert werden kann. Aus diesem Grund wurde in diesem Projekt ein Autoencoder-basierter Demapper implementiert.

3.1 LDPC Decoder

Die Auswahl des Kanaldecoders erfolgte aufgrund seiner Komplexität und Bedeutung in der Basisbandsignalverarbeitung. LDPC Codes werden aufgrund ihrer guten Performanz in Funkkommunikationssystemen für hohe Datenraten wie im FunKI-Anwendungsfall eingesetzt. Die Quantisierung wurde als entscheidende Stellschraube zur effizienten Implementierung von hochparallelen LDPC Decodern identifiziert. Dabei wurde die Information Bottleneck Methode (IBM) zur effizienten Quantisierung von Kantennachrichten erforscht, diese erfordert jedoch den Ersatz elementarer Knotenoperationen durch Look-Up Tabellen (LUTs).

Eine einfache IBM Decoder-Version ist der mLUT Decoder, aber seine exponentiell wachsenden LUTs bei höheren Knotengraden sind problematisch. Eine alternative Methode zur Reduzierung der LUT Größe ist die Serialisierung, die die Anzahl der Literale linear mit dem Knotengrad erhöht. Implementierungsergebnisse zeigen, dass IBM Decoder bei niedriger Quantisierung keine signifikanten Verbesserungen bieten, während sie bei höherer Quantisierung an Effizienz verlieren.

Die Analyse des IBM Decoders auf Architektur-/Mikroarchitekturebene zeigt die höhere Komplexität im Vergleich zu konventionellen Decodern. Der Minimum Integer Computation (MIC) Decoder wurde entwickelt, um die Knotenkomplexität zu reduzieren. Dieser Ansatz verbessert die Implementierungseffizienz erheblich, bietet bessere Skalierbarkeit und ermöglicht die Verarbeitung größerer Blockgrößen, was die Fehlerkorrekturfähigkeit und den Durchsatz von hochparallelen Decoder-Architekturen steigert.

Die im Projekt erlangten Erkenntnisse bezüglich des LDPC Decoders waren Grundlage für drei Publikationen (siehe Tabelle 4 und 5 der Eingehenden Darstellung).

3.2 Autoencoder

Im Rahmen des FunKI Projekts wurde eine effiziente Hardware-Architektur eines Autoencoder-basierten Kommunikationssystems implementiert, die in der Lage ist, sich Kanalschwankungen anzupassen, indem das Empfänger-ANN zur Laufzeit nachtrainiert wird. Die entwickelte FPGA-Implementierung ist äußerst flexibel, da sie variable Quantisierung und einen flexiblen Parallelisierungsgrad unterstützt, der individuell für Inferenz und Training konfiguriert werden kann. Darüber hinaus wurde ein Framework entworfen, das die Lücke zwischen der Kommunikationstechnischen-Ebene und der Hardware-Ebene schließt, indem es den Parallelisierungsgrad je nach Anwendungsanforderung anpasst. Des Weiteren, wurden die Vorteile unserer Hardware-Architektur und des FPGA als Implementierungsplattform gezeigt, indem wir unsere Implementierung mit verschiedenen General-Purpose-Prozessoren verglichen haben. Dabei erreichte unsere FPGA-basierte AE-Implementierung einen um das 2000-fache höhere Durchsatzrate als eine leistungsstarke GPU, verbrauchte 5-mal weniger Energie als eine eingebettete CPU und ist im Vergleich zu einer eingebetteten GPU für kleine Batch-Sizes um das 5800-fache energieeffizienter. Außerdem wurde der neuartige Autoencoder-basierte Ansatz mit einem klassischen Demapper hinsichtlich der Implementierungskomplexität verglichen.

Des Weiteren diente die Implementierung des Autoencoders als Grundlage für den Demonstrator, der in Zusammenarbeit mit Creonic im Rahmen von AP5 entwickelt wurde. Dieser demonstriert anschaulich wie das Training des Autoencoder-basierten Demappers erfolgt und wurde auf verschiedenen Projektmeetings und Messen ausgestellt.

Die im Projekt erlangten Erkenntnisse bezüglich des Autoencoders waren Grundlage für drei Publikationen (siehe Tabelle 4 und 5 der Eingehenden Darstellung).

FunKI

Funkkommunikation mit Künstlicher Intelligenz

Entwurf und Implementierung von Hardware-Architekturen für KI-Algorithmen in Sende- und Empfängerstrukturen von Funkkommunikationssystemen

Sachbericht zum Verwendungsnachweis
Teil 2: Eingehende Darstellung

Förderkennzeichen: 16KIS1185
Förderzeitraum: 15.05.2020 bis 14.05.2023

GEFÖRDERT VOM
Bundesministerium für Bildung und Forschung

CREONIC ip cores & system solutions
Deutsches Forschungszentrum für Künstliche Intelligenz GmbH
intel
MOTIUS WE R&D.

Inhaltsverzeichnis

4. Projektübersicht

Im Rahmen des Projekts FunKI (Funkkommunikation mit Künstlicher Intelligenz) wurde eine grundsätzliche Untersuchung KI-getriebener Technologien für Funkkommunikation verfolgt, die sich an der 5. Mobilfunkgeneration (5G) und deren Weiterentwicklung orientierte. Speziell wurden Konzepte zum verbesserten Modellverständnis basierend auf Messkampagnen erarbeitet, KI-basierte Methoden zur Parameterschätzung entwickelt, die datengetriebene Optimierung von Sende- und Empfangsverfahren vorgenommen und KI-basiertes Training der Hardware-Implementierungen durchgeführt. FunKI führte richtungsweisende Unternehmen und Forscher aus den Bereichen 5G-Funkkommunikation, künstlicher Intelligenz und dem Entwurf mikroelektronischer Systeme zusammen, um die anstehenden Herausforderungen bei der Umsetzung zukünftiger Kommunikationssysteme zu bewältigen und auch auf internationaler Ebene weiterhin eine Führungsrolle einzunehmen.

Im Fokus des Projektes stand die Entwicklung und Erprobung lern- und anpassungsfähiger Verfahren der physikalischen Übertragungsschicht und deren Umsetzung und Implementierung. Dabei wurde die Frage adressiert, welche Module der Signalverarbeitungskette im Sender und im Empfänger vorteilhaft mit KI-Verfahren umgesetzt werden können, oder ob sogar vollständig erlernte Sender- und Empfänger ohne Vorgabe spezieller Strukturen zu effizienteren Transceivern führen können. Da aufgrund der begrenzten Ressourcen nicht alle Einzelkomponenten in FunKI zu betrachtet werden konnten, werden vielversprechende Komponenten ausgewählt, die optimiert werden sollen. Im Anschluss wurde die Rückwirkung dieser KI-basierten Optimierung oder Implementierung auf das Gesamtsystem analysiert und diskutiert.

4.1 Ziele

Die möglichen Vorteile von KI-Verfahren auf der PHY-Ebene sind vielfältig und reichen von der Entwicklung effizienter Empfängerkomponenten (Decoder, MIMO-Detektoren, Entzerrer) mit geringerer Bit-Auflösung, kontrollierbarer Latenz und Komplexität, bis hin zu weniger quantifizierbaren, methodischen Vorteilen, etwa der inhärenten Generalisierbarkeit von KI-Methoden, was zum schnelleren, kostengünstigeren Entwurf von robusten Übertragungsverfahren führt. Die Industrie kann damit schnell und flexibel auf die Freigabe neuer Frequenzbänder reagieren oder neue Störszenarien (wie etwa in Industrie 4.0-Anwendungen) ohne vollständiges Re-Design in ihren PHY-Systemen berücksichtigen.

Insbesondere umfassten die Ziele von FunKI den datengetriebenen Entwurf von Sendestrukturen, z.B. zum optimierten Design von Sendeimpulsformen und Synchronisationssequenzen, den datengetriebenen Entwurf von Empfängern zur Kompensation von nichtlinearen Verzerrungen, und den gemeinsamen Entwurf von Sendern und Empfängern. Weiterhin genannt seien die Parameterextraktion aus MIMO-Kanalmessungen und Herleitung robuster Prädiktoren, die Verbesserung der Qualität der zurzeit eingesetzten Verfahren zur Link Adaption basierend auf KI/ML zur bestmöglichen dynamischen Anpassung der zur Übertragung gewählten Modulationsart und Coderate, sowie des MIMO-Kanalranges und der Sendegewichte.

Ausgehend von ausgewählten 5G Anwendungsfällen wurden in FunKI somit lern- und anpassungsfähige Kommunikationssysteme mit Fokus auf dem PYH- und MAC-Layer entwickelt und bis hin zur effizienten Hardware-Umsetzungen analysiert und optimiert.

4.2 Rolle der RPTU im Projekt FunKI

Der Lehrstuhl Entwurf Mikroelektronischer Systeme legte den Schwerpunkt seiner Arbeiten in diesem Projekt auf das engen Zusammenspiel zwischen den Algorithmen und deren effizienter

Implementierung auf fortgeschrittenen ASIC-Technologien und FPGAs. Hierzu spielte der Trade-Off zwischen nachrichtentechnischer Performanz, entsprechenden Algorithmen und Implementierungseffizienz eine zentrale Rolle. Die Untersuchung dieses Zusammenspiels mit dem Ziel einer bestmöglichen Implementierung mit kleiner Fläche, hohem Durchsatz und großer Energieeffizienz unter nachrichtentechnischen Performanzvorgaben wird als Entwurfsraumexploration bezeichnet. Aufgabe der RPTU in diesem Projekt war es, in Zusammenarbeit mit den Partnern, die ihren Fokus mehr auf der nachrichtentechnischen und algorithmischen Seite haben (insbesondere Universität Stuttgart und Bremen), in einem Ebenen-übergreifenden iterativen Ansatz effiziente NN-Acceleratorarchitekturen zu entwickeln, die einen hohen Durchsatz, kleine Fläche und eine hohe Energieeffizienz aufweisen. Zur Bewertung der Implementierungseffizienz war ein Virtual Silicon Ansatz notwendig, da valide Daten bezüglich Durchsatz (d.h. Frequenz) und Leistungsverbrauch nur dann möglich sind, wenn das Layout einer ASIC-Implementierung vorliegt. Des Weiteren galt es, unter besonderer Berücksichtigung der effizienten Implementierung, herauszuarbeiten in welchen Bereichen diese Ansätze den herkömmlichen Ansätzen überlegen sind und wie herkömmliche Ansätze mit NN-basierten Ansätzen effizient kombiniert werden können. Darüber hinaus war es für den Projekterfolg wichtig, anhand eines konkreten Demonstrators die Performanz der NN-basierten Ansätze zu zeigen.

Die in diesem Projekt durchzuführten Arbeiten und Untersuchungen lieferten ein tiefergreifendes Verständnis von NN-basierten Acceleratorarchitekturen in Kommunikationssystemen und werden deshalb sicher auch weitere Impulse für die Grundlagenforschung auf diesem hochaktuellen Gebiet liefern, was Ausgangspunkt für weitere Forschungsarbeiten sein wird. Mit den Erkenntnissen dieses Projektes kann eine Brücke zwischen den beiden sehr wichtigen Gebieten des maschinellen Lernens und der Kommunikationstechnik erfolgreich geschlagen werden.

5. Arbeitsplan und Rolle der RPTU in den einzelnen Arbeitspaketen

Für das Projekt FunKI wurden die Arbeiten in 5 Arbeitspakete (APs) entsprechend Abbildung 1 Aufgeteilt.

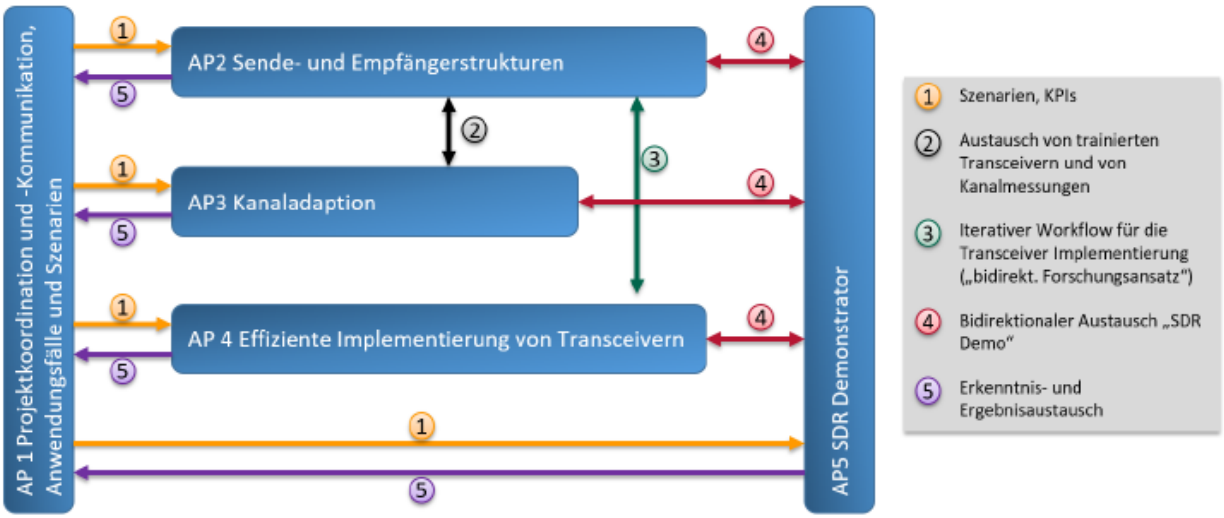

Abbildung 1: Arbeitspakete

Die RPTU war dabei hauptsächlich in AP-4 „Effiziente Implementierung von Transceivern" und AP-5 „SDR Demonstrator" beteiligt.

5.1 Rolle der RPTU in AP-4

Ziel von AP-4 war die Implementierung von ausgewählten Transceiver-Komponenten als sogenannte „Hardware-Acceleratoren“. Für die Implementierung wurden insbesondere die Komponenten der Basisbandsignalverarbeitung (z. B. Demodulation, Entzerrung, Kanaldecodierung) betrachtet.

Die RPTU beteiligte sich an der Hardwareimplementierung, d.h. der Architekturentwicklung, sowie der entsprechenden Implementierung und Optimierung auf fortgeschrittener ASIC-Technologie. Zudem wurde eine ausführliche Analyse der Implementierungseffizienz und der verschiedenen Trade-Offs erstellt. Die RPTU übernahm die Leitung dieses Arbeitspakets. Zudem bildete sie die Schnittstelle zu AP-5.

5.2 Rolle der RPTU in AP-5

Das Ziel von AP-5 war die Entwicklung eines Demonstrators, welcher die Machbarkeit von lernfähigen Algorithmen im Bereich der Basisbandsignalverarbeitung in existierender Hardware aufzeigt. Hierzu wurden ausgewählte Komponenten klassischer Funkkommunikationssysteme durch KI-Komponenten, basierend auf den Ergebnissen der vorherigen Arbeitspakete, ausgetauscht. Eine Selektion von Demonstrationsszenarien im Hinblick auf Systemspezifikationen des 5G Standards stellten die von FunKI angestrebte Verwertung und Anwendung im Bereich von 5G sicher. Folglich wurden in AP-5 die drei Anwendungsfälle in Abhängigkeit der ausgewählten Komponenten adressiert.

Die RPTU beteiligte sich an der Auswahl und Definition geeigneter Demonstrationsszenarien. Zudem wurden, gemeinsam mit den beteiligten Partnern, Konzeptionierung, Aufbau und Optimierung einer FPGA-Hardwareplattform/GPU-Softwareplattform zur Umsetzung der KI-Komponenten aus AP-4 durchgeführt. Nach dem Einbinden aller Module in die Demonstrator-Plattform wurden abschließend Vergleiche mit Referenzmodellen gezogen. Die RPTU bildete hier die Schnittstelle zu AP-4.

6. Notwendigkeit und Angemessenheit der geleistete Projektarbeiten

Die geleisteten Projektarbeiten entsprechen im Wesentlichen den im Antrag vorgesehenen Arbeiten. Aufgrund der Corona-Pandemie konnten weitgehend nicht wie ursprünglich geplant Präsenz-Aktivitäten (beispielsweise Projektmeetings in Präsens) durchgeführt werden und es musste in diesen Fällen auf Online-Formate (beispielsweise Online-Meetings und Workshops) ausgewichen werden. Aufgrund der hohen Kompetenz und der Vorarbeiten der beteiligten Partner sowie des fokussierten Projektplanes konnten die Projektziele trotz dessen weitestgehend erreicht werden.

7. Wichtigste Positionen des zahlenmäßigen Nachweises

Für die Durchführung von FunKI waren sowohl für die Maßnahmen zur technischen Umsetzung als auch für die inhaltlichen Arbeitspakete Personalressourcen und Reisekosten notwendig. Insgesamt betrug die Summe der beantragten Fördermittel (incl. Projektpauschale) 516.742,63 Euro.

7.1 Positionen im Einzelnen

Der größte Teil der Gelder wurde für wissenschaftliche Mitarbeiter verausgabt. Hier entstanden Ausgaben in Höhe von 427.744,97 Euro (Pos. 0812). Für Dienstreisen ergab sich die zweite Ausgabenposition (Pos. 0822) mit 3.614,94 Euro. Hierrunter fallen die Konferenzgebühr des International Symposiums on Field-Programmable Gate Arrays (62,41 Euro) sowie die Reisekosten

auf die SOCC Konferenz in Belfast (2.006,18 Euro) und die EUSIPCO Konferenz in Belgrad (1.546.35 Euro). Durch diese Konferenzen, mit dem Fokus auf Mikroelektronik und Signalverarbeitung, konnten im Projekt wichtige Kooperationen verstärkt und neue Erkenntnisse gewonnen werden.

Die Projektpauschale betrug 86.123,77 Euro. Die geringe Abweichung vom Gesamtfinanzierungsplan resultierte zum einen aus den tatsächlichen Beschäftigungsentgelten, abhängig von den entsprechenden Erfahrungsstufen, und zum anderen aus dem Corona-Pandemie bedingten Reiseverbot.

7.2 Änderung bei der Ausgabenplanung

Im ursprünglichen Zeitplan wurde davon ausgegangen, dass die notwendigen Stellen zügig besetzt werden können. Allerdings konnten die geplanten Stellenbesetzungen erst verzögert vorgenommen werden. Dadurch kam es zu Beginn zu einer kurzfristigen Zeitversetzung bei der Bearbeitung der Arbeitspakete. Diese Zeitverzögerung konnte jedoch durch verdichtete Arbeit kompensiert werden. Die geplanten Stellen mussten zudem aufgrund der veränderten individuellen beruflichen Planung der vorgesehenen Personen letztlich auf mehrere Personen verteilt werden. Dadurch entstand vor Ort insgesamt ein Mehraufwand in der Koordination. Dieser Mehraufwand tangierte allerdings nicht die finanziellen Rahmenbedingungen des Projekts.

8. Beschreibung der Ergebnisse

Im Folgenden werden die von der RPTU erzielten wissenschaftlichen Ergebnisse im Projekt FunKI detailliert beschrieben.

8.1 Komponentenauswahl und Methodik

Innerhalb von AP-4 wurden in Zusammenarbeit mit den Partnern Creonic sowie der Universität Stuttgart und Bremen folgende Transceiver Komponenten zur Implementierung ausgewählt:

1) LDPC-Decoder: Die Dekodierung von LDPC-Codes basiert auf einem iterativen Austausch von Nachrichten über die Kanten Tanner-Graphen, der aus der H-Matrix des Codes abgeleitet ist. Dieser Nachrichtenaustausch ist ein kritischer Faktor für die effiziente Implementierung von LDPC-Decodern, insbesondere für hohe Datenraten, da er die Fläche des Decoders durch die Zuweisung von Verdrahtungsressourcen, den kritischen Pfad und den Stromverbrauch beeinflusst. Die Anzahl der Nachrichten ist durch die Topologie des Graphen vorgegeben, wobei die Quantisierung als wichtigster Optimierungsparameter für die Implementierungseffizienz verbleibt. Aus diesem Grund wurde die Information-Bottleneck-Methode (IBM) zur effizienten Quantisierung von Kanten-Nachrichten untersucht.
2) Autoencoder: Im Kontext von Kommunikationssystemen bezeichnet Autoencoder (AE) ein System, das Teile des traditionellen Senders und Empfängers durch künstliche neuronale Netzwerke (ANNs) ersetzt. Dadurch kann das System, weitestgehend unabhängig vom Kanalmodell, global optimiert werden, wodurch die nachrichtentechnische Performanz im Vergleich zu konventionellen Ansätzen gesteigert werden kann.

Um eine gleichzeitige Untersuchung sowohl der Algorithmen als auch der Implementierungskomplexität zu ermöglichen, wurden sogenannte Entwurfsraumexplorations-Frameworks (DSE-Framework) entwickelt. Diese ermöglichen eine schnelle Adaptierung der Implementierung an algorithmische Anpassungen und Änderungen der Designparameter.

Für den LDPC-Decoder umfasste das DSE-Framework eine C++-Software-Simulationsumgebung für Kommunikationssysteme sowie VHDL-Bibliotheken mit Beschreibungen elementarer Hardwarekomponenten. Basierend auf diesen Komponenten und der Designparameter erstellen sogenannte Hardwaregeneratoren eine VHDL-Beschreibung der Decoder-Architektur.

Für den Autoencoder lädt das DSE-Framework vor-synthetisierte Bitstreams und lieft direkt auf der CPU der FPGA-Plattform. Dies ermöglichte eine Anpassung der Parallelisierung von Inferenz und Training basierend auf den Anwendungsanforderungen wie Latenz oder Bitfehlerrate.

8.2 Finite Alphabet Message Passing (FA-MP) LDPC Decoder

Als *Komponente 1* wurde wie eingangs erwähnt der Finite Alphabet Message Passing (FA-MP) LDPC Decoder ausgewählt, da dieser die komplexeste Komponente in der Basisbandsignalverarbeitungskette darstellt und dessen Implementierung als besonders kritisch zu bewerten ist. Weiterhin finden LDPC Codes aufgrund ihrer guten Performanz bereits breite Anwendung in Funkkommunikationssystemen wie beispielsweise Wifi oder 5G-NR. Da die zugrundeliegende Belief Propagation (BP) Decodierung gut parallelisierbar ist, eignen sich LDPC Codes insbesondere auch für hohe Datenraten wie sie beispielsweise für den FunKI Anwendungsfall „Kosteneffiziente flächendeckende Breitbandversorgung“ angestrebt wurden.

Für hochparallele LDPC Decoder Architekturen wurde die Quantisierung als wichtige Stellschraube für die Implementierungseffizienz identifiziert. Aus diesem Grund wurde in AP-4 die Information Bottleneck Methode (IBM) zur effizienten Quantisierung der Kantennachrichten untersucht. Eine Konsequenz dieses Verfahrens in der Decoder-Implementierung ist, dass die elementaren, algebraischen Knotenoperationen durch gelernte Abbildungsvorschriften, in Form von Look-Up Tabellen (LUT) ersetzt werden müssen. Daraus ergibt sich ein Trade-Off zwischen der Einsparung von Datentransfers als Ergebnis einer geringeren Quantisierung gegenüber den Kosten, die sich aus der höheren Komplexität in der Implementierung der Knotenoperationen ergeben.

Die simpelste Form eines IBM Decoders ist der mLUT Decoder, dessen Abbildungsvorschriften in Form mehrdimensionaler LUTs vorliegen. Da die Größe der LUTs und somit die Anzahl der Literale in den repräsentierten Abbildungsvorschriften exponentiell mit der Anzahl der Eingangskanten des entsprechenden Knoten und der Quantisierung der Kantennachrichten anwächst, führt dies schnell zu praktischen Problemen in der Implementierung. So besitzen State-of-the-Art Synthese Tools ein oberes Limit für die Anzahl der Literale einer Logikfunktion, die in der Logiksynthese optimiert werden kann, beispielsweise $2^{24} = 16.777.216$ im Fall von Synopsys Design Compiler. Dies entspricht beispielsweise einem Variablenknoten vom Grad 6 bei einer Quantisierung von 4 bit. Während dies im Fall des Variablenknotens für viele gängige Codes ausreicht, ist dies insbesondere für den Checkknoten, der in der Regel deutlich höhere Knotengrade aufweist, ein Problem. State-of-the-Art IBM Decoder nutzen daher eine Minimum Approximation im Checkknoten, die zu einer unwesentlichen Verschlechterung der Performanz führt. Diese Art von Decoder erhält im Folgenden ein „Min“ Präfix, z.B. hier Min-mLUT.

Ein weiteres State-of-the-Art Verfahren zur Reduzierung der LUT Größe ist deren Serialisierung, d.h. an Stelle einer einzigen mehrdimensionalen LUT wird eine Kaskade von zweidimensionalen LUTs verwendet. Damit wächst die Anzahl der Literale der repräsentierten Logikfunktion zwar weiterhin exponentiell mit der Quantisierung, jedoch nur noch linear mit dem Knotengrad. Für das obige Beispiel eines Variablenknoten vom Grad 4 und 4 bit Quantisierung reduziert sich die Anzahl der Literale von $2^{6\cdot4} = 16.777.216$ auf $6 \cdot 2^{2\cdot4} = 1536$. In Verbindung mit der Minimum Approximation im Checkknoten, wird dieser Decoder im Folgenden als Min-sLUT bezeichnet.

Die Untersuchung der beschriebenen Trade-Offs erfolgte basierend auf einem Design Space Exploration (DSE) Framework für LDPC Decoder (siehe Abbildung 2). Dieses umfasst eine C++

Software Simulationsumgebung für Kommunikationssysteme, sowie VHDL Bibliotheken mit Beschreibungen elementarer Hardwarekomponenten. Sogenannte Hardwaregeneratoren generieren aus diesen Komponenten entsprechend der gewählten Design Parameter eine VHDL Beschreibung der Decoderarchitektur. Dieses DSE Framework wurde zunächst um erste Versionen der o.g. Ansätze, d.h. IBM und WBP, erweitert. Dazu wurden neue, bitgenaue, parametrisierbare Softwaremodelle, Hardwarebausteine und entsprechende Hardwaregeneratoren in das Framework integriert.

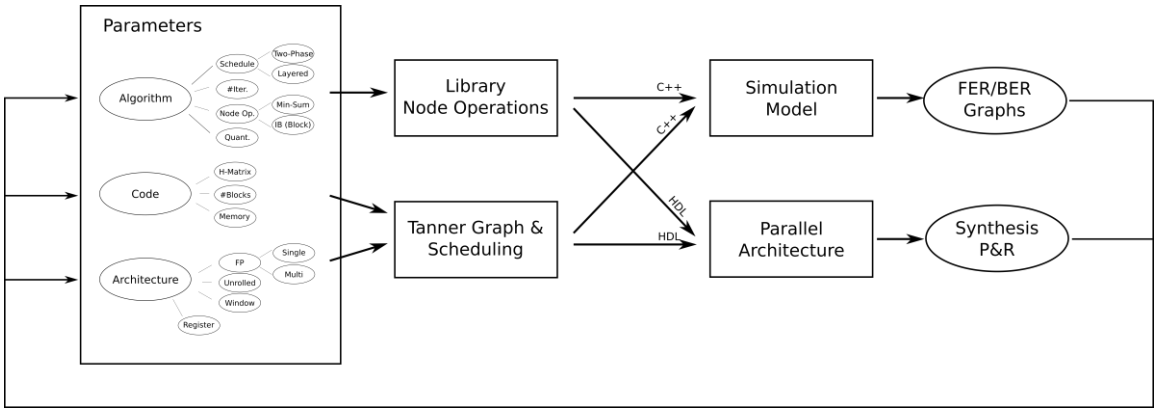

Abbildung 2: LDPC Decoder Design Space Exploration Framework.

In Tabelle 1 und Tabelle 2 werden Implementierungsergebnisse (22nm Technologie) der diskutierten Decoder, Min-mLUT und Min-sLUT, sowie einem konventionellen Normalized Min-Sum (NMS) Decoder gegenübergestellt ($(816,406)$-Code, vollparallele Architektur mit 8 abgerollten Iterationsstufen). Die Quantisierung (n_E, n_Q, n_R) wurde so gewählt, dass sich eine vergleichbare Performanz der jeweiligen Decoder ergibt (Abbildung 3, Abbildung 4). Beim Vergleich der Energieeffizienz fällt auf, dass sich bei 3 bit Auflösung (Tabelle 1) keine signifikante Verbesserung der IBM Decoder gegenüber NMS ergibt, bei 4 bit Auflösung (Tabelle 2) sogar eine Verschlechterung einstellt, d.h. die komplexeren Logikfunktionen überwiegen im Trade-Off gegenüber den Datentransfers.

Tabelle 1 *Implementierungsergebnisse und Vergleich in 22nm, IBM/NMS Decoder* ($n_E = n_Q = 3$, $n_R = 5$)

	MIC	Min-mLUT	Min-sLUT	NMS
n_E, n_Q	3	3	3	4
E_b/N_0 @ FER 10^{-4} $[dB]$	4.20	4.16	4.35	4.26
Utilization [%]	70	68	71	71
Frequency $[MHz]$	775	662	670	595
Coded Throughput $[Gb/s]$	633	540	547	486
Area $[mm^2]$	2.73	4.23	2.86	3.04
Area Efficiency $[Gb/s/mm^2]$	231.6	128	190	159.7
Latency $[ns]$	33.5	39.3	35.8	43.7
Power $[W]$	4.49	5.07	4.38	4.39
Energy Efficiency $[pJ/bit]$	7.10	9.4	8.0	9.0

Tabelle 2 *Implementierungsergebnisse und Vergleich in 22nm IBM/NMS Decoder* ($n_E = n_Q = 3$, $n_R = 5$)

	MIC	Min-mLUT	Min-sLUT	NMS

	MIC	Min-mLUT	Min-sLUT	NMS
n_E, n_Q	4	4	4	5
E_b/N_0 @ FER 10^{-4} $[dB]$	3.94	3.87	3.93	4.01
Utilization [%]	69	49	66	69
Frequency $[MHz]$	633	267	492	183
Coded Throughput $[Gb/s]$	516	218	401	149
Area $[mm^2]$	3.66	40.51	7.82	3.99
Area Efficiency $[Gb/s/mm^2]$	141.1	5.4	51.3	37.4
Latency $[ns]$	41.1	97.2	48.0	142.0
Power $[W]$	5.61	11.85	8.68	2.25
Energy Efficiency $[pJ/bit]$	10.9	54.3	21.6	15.1

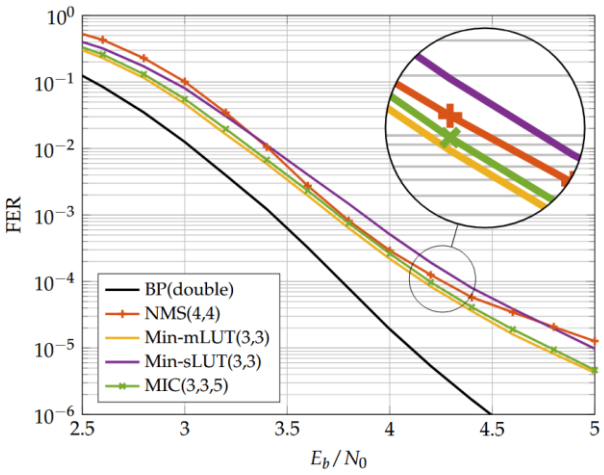

Abbildung 3: Performanz $n_e = 3$ bit IBM Decoder.

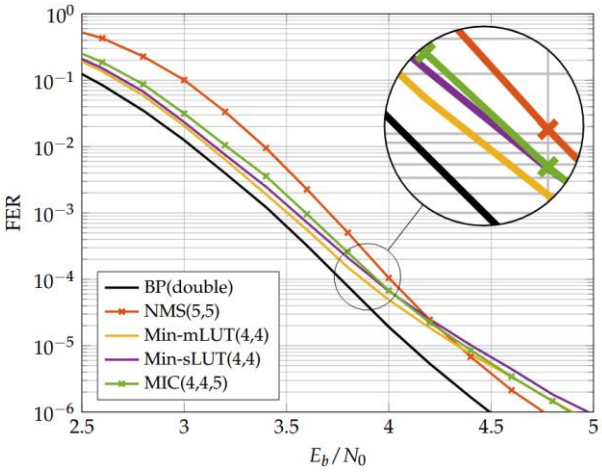

Abbildung 4: Performanz $n_e = 4$ bit IBM Decoder.

Weiterhin wurden Optimierungen des IBM Decoders auf Architektur-/Mikroarchitekturebene untersucht. Zu diesem Zweck wurde zunächst das DSE Framework um ein Tool zur grafischen Aufbereitung der Netzliste ergänzt, das eine detaillierte Analyse der synthetisierten Schaltung ermöglicht. Abbildung und Abbildung zeigen im direkten Vergleich die Schaltpläne eines IBM und eines MS Variablen Knoten. Deutlich zu erkennen ist die höhere Komplexität der IBM Schaltung. Darüber hinaus zeigt die Analyse der verwendeten Zellen eine Limitierung in der Logiksynthese der LUT-basierten Abbildungsvorschriften. Die verwendete Mapping-Heuristik kann ausschließlich single-output Zellen und keine optimierten multi-output Zellen verwenden, beispielsweise Volladdierer, wie sie beim MS Knoten eingesetzt werden.

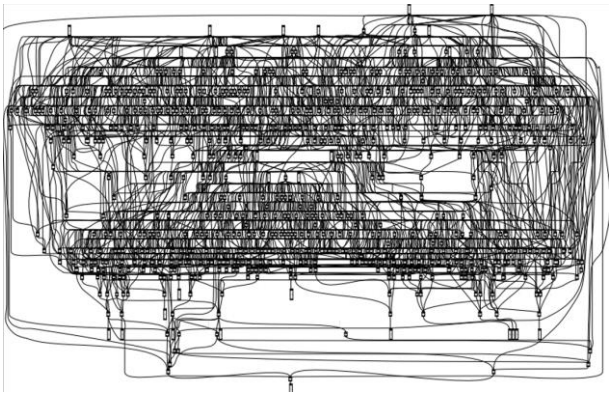

Abbildung 5 - Schaltplan eines IBM Variablen Knoten vom Grad 6 (4 Bit Quantisierung).

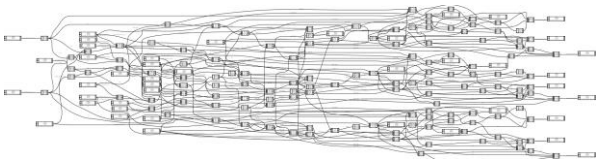

Abbildung 6 - Schaltplan eines Min-Sum Variablen Knoten vom Grad 6 (6 Bit Quantisierung).

Aus diesem Grund wurde in Zusammenarbeit mit dem Partner UB ein neuer Typ von IBM Decoder mit deutlich geringerer Knotenkomplexität entwickelt, im Folgenden Minimum Integer Computation (MIC) Decoder genannt. Dieser kombiniert konventionelle Abbildungsvorschriften und LUTs zur Reduzierung der Knotenkomplexität. Für eine detaillierte Beschreibung des Verfahrens verweisen wir auf zwei gemeinschaftliche Publikationen[1,2], die 2022 erschienen sind. Tabelle 1 und Tabelle 2 zeigen die Überlegenheit des MIC Decoders gegenüber den State-of-the-Art IBM Decoders und dem konventionellen NMS Decoder in praktisch allen relevanten Implementierungsmetriken bei gleichbleibender nachrichtentechnischer Performanz. Abbildung 7 zeigt Chip Layouts des MIC und NMS Decoders in 22nm Technologie im direkten Größenvergleich.

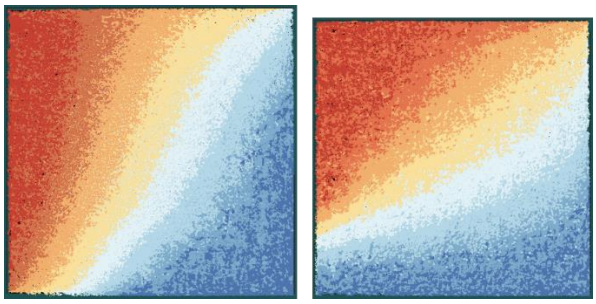

Abbildung 7: Chip Layout des NMS(5,5) und des MIC(4,4,5) im direkten Größenvergleich; jede Farbe zeigt eine Iterationsstufe des abgerollten Decoders.

Unsere Analyse zeigt, dass der neue MIC-Ansatz die Implementierungseffizienz erheblich verbessert. Dieser weist eine bessere Skalierung im Vergleich zu State-of-the-Art Min-mLUT-, Min-sLUT- und NMS-Implementierungen von hochparallelen LDPC Decoder Architekturen auf. Dies ermöglicht die Verarbeitung größerer Blockgrößen, was hauptsächlich auf die reduzierte Verdrahtungskomplexität zurückzuführen ist. Größere Blockgrößen verbessern die Fehlerkorrekturfähigkeit und erhöhen den Durchsatz von hochparallele Decoder-Architekturen weiter.

[1] T. Monsees, D. Wübben, A. Dekorsy, O. Griebel, M. Herrmann and N. Wehn, "Finite-Alphabet Message Passing using only Integer Operations for Highly Parallel LDPC Decoders," 2022 IEEE 23rd International Workshop on Signal Processing Advances in Wireless Communication (SPAWC), Oulu, Finland, 2022, pp. 1-5, doi: 10.1109/SPAWC51304.2022.9833953.

[2] T. Monsees, O. Griebel, M. Herrmann, D. Wübben, A. Dekorsy, N. Wehn, "Minimum-Integer Computation Finite Alphabet Message Passing Decoder: From Theory to Decoder Implementations towards 1 Tb/s", Entropy 2022, 24, 1452, doi: https://doi.org/10.3390/e24101452.

8.3 Autoencoder

Kommunikationssysteme haben sich in den letzten Jahrzehnten rasant weiterentwickelt. Während 2G/GSM Datenraten von etwa 10 kbps erreichte, unterstützt 5G über 10 Gbps und es wird eine Datenrate von über 1 Tbps für die nächsten Standards erwartet.

Das Design eines digitalen Basisband-Funkkommunikationssystems folgt traditionell zwei Schritten:

1) Aufteilung des Systems in Verarbeitungsblöcke wie Quellencodierung, Kanalcodierung und Modulation.
2) Optimierung jeder Unterkomponente für das Kanalmodell und die Anwendungsanforderungen.

Ein Nachteil dieses Ansatzes ist, dass eine Optimierung der einzelnen Komponenten nicht zwangsläufig die ideale Performanz des Gesamtsystems gewährleistet[3]. Um dieses Problem zu lösen, werden ANN-basierte Methoden in der Kommunikationstechnik untersucht[4,5,6].

Eine vielversprechender Ansatz ist der Autoencoder (AE), welcher Teile des klassischen Senders und Empfängers durch künstliche neuronale Netzwerke (ANNs) ersetzt. Der AE liefert erfolgsversprechende Ergebnisse[7,8], da das Gesamtsystem global optimiert werden kann. Daher wurde der AE in diesem Projekt als *Komponente 2* ausgewählt.

Ein Nachteil dieses Ansatzes, im Vergleich zu klassischen Verfahren, ist jedoch die hohe Implementierungskomplexität von ANNs. Eine Herausforderung innerhalb dieses Projekts war es daher, den ANN-basierten Ansatz effizient auf Ressourcenlimitierten FPGA zu implementieren ohne die nachrichtentechnische Performanz zu beeinträchtigen.

8.3.1 System Modell

Im Folgenden wird das System Modell des in diesem Projekt betrachteten AEs beschrieben, welcher auf den Arbeiten der Universität Stuttgart basiert. In diesem Fall beschränkt sich der AE auf den Mapper, Demapper und den Kommunikationskanal, wie in Abbildung 8 dargestellt.

[3] E. Zehavi, „8-PSK trellis codes for a Rayleigh channel“, *IEEE Trans. Commun.*, Bd. 40, Nr. 5, S. 873–884, Mai 1992, doi: 10.1109/26.141453.

[4] T. V. Sethuraman, S. Elias, und A. Ashok, „Demo: A Machine Learning based M-ary Amplitude Modulated Visible Light Communication System“, in *2020 International Conference on COMmunication Systems & NETworkS (COMSNETS)*, Bengaluru, India: IEEE, Jan. 2020, S. 694–695. doi: 10.1109/COMSNETS48256.2020.9027292.

[5] B. Gao, B. Bu, W. Zhang, und X. Li, „An Intrusion Detection Method Based on Machine Learning and State Observer for Train-Ground Communication Systems“, *IEEE Trans. Intell. Transp. Syst.*, Bd. 23, Nr. 7, S. 6608–6620, Juli 2022, doi: 10.1109/TITS.2021.3058553.

[6] M. R. Mahmood und M. A. Matin, „A Design of Extreme Learning Machine Based Receiver for 2×2 MIMO-OFDM System“, in *2021 IEEE International Conference on Communication, Networks and Satellite (COMNETSAT)*, Purwokerto, Indonesia: IEEE, Juli 2021, S. 367–370. doi: 10.1109/COMNETSAT53002.2021.9530798.

[7] T. J. O'Shea und J. Hoydis, „An Introduction to Deep Learning for the Physical Layer“, *ArXiv170200832 Cs Math*, Juli 2017, Zugegriffen: 14. September 2021. [Online]. Verfügbar unter: http://arxiv.org/abs/1702.00832

[8] B. Karanov *u. a.*, „End-to-End Deep Learning of Optical Fiber Communications“, *J. Light. Technol.*, Bd. 36, Nr. 20, S. 4843–4855, Okt. 2018, doi: 10.1109/JLT.2018.2865109.

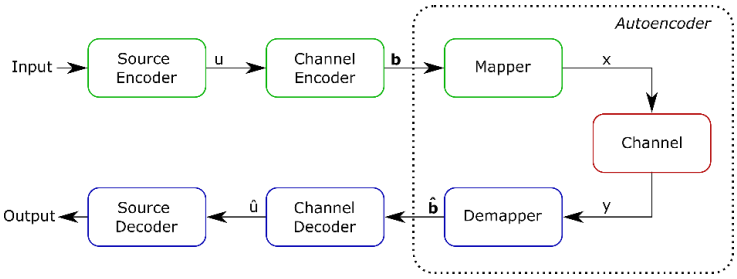

Abbildung 8: System Modell

Ziel des Mappers ist es, eine Symbol-Konstellation zu finden, die möglichst robust gegenüber dem Rauschen des Kanals ist. Ziel des Demappers ist es, die übertragenen Bits mit möglichst höher Zuverlässigkeit zu bestimmten. Beim AE Ansatz werden Mapper und Demapper durch ANNs repräsentiert und gemeinsam Ende-zu-Ende über ein Kanalmodell trainiert. Dabei bestehen Mapper und Demapper aus drei Fully-Connected Layern, jeweils gefolgt von einer ReLU Aktivierungsfunktion. Die Layer des Mappers bestehen dabei jeweils aus 128 Neuronen. Die Anzahl der Neuronen des Demappers hingegen wurden in einer Entwurfsraumexploration bestimmt. Basierend auf einem Framework, wurden die Anzahl an Neuronen soweit reduziert, bis eine weitere Reduzierung einen Einbruch der Nachrichtentechnischen Performanz bedeuten würde. Durch diese Methodik, konnte ein ANN gefunden werden, welches lediglich 16 Neuronen pro Layer besitzt und damit eine vergleichsweise geringe Berechnungskomplexität aufweist. Zusätzlich erreicht das Modell im Vergleich zu einer konventionellen 16-QAM Konstellation eine höhere Mutual Information im niedrigen SNR-Bereich.

8.3.2 Trainings Ergebnisse

Der im vorherigen Kapitel vorgestellt AE wurde für Kanalmodelle mit verschiedenen SNRs trainiert, um verschiedene, SNR-optimierte Symbolkonstellationen zu erhalten. Die SNRs befanden sich im Bereich von -6 bis 4 dB. In Abbildung 9 sind die trainierten Konstellationen und die Mutual Information für die verschiedenen SNRs dargestellt. Der informationstechnische Gewinn des AEs kann anhand der Mutual Information verdeutlicht werden. Es wird deutlich, dass die jeweiligen Konstellationen für genau das SNR am besten performen, für das sie trainiert wurden. In diesem Bereich übertreffen die AE-basierten Konstellationen auch die konventionelle 16-QAM Konstellation.

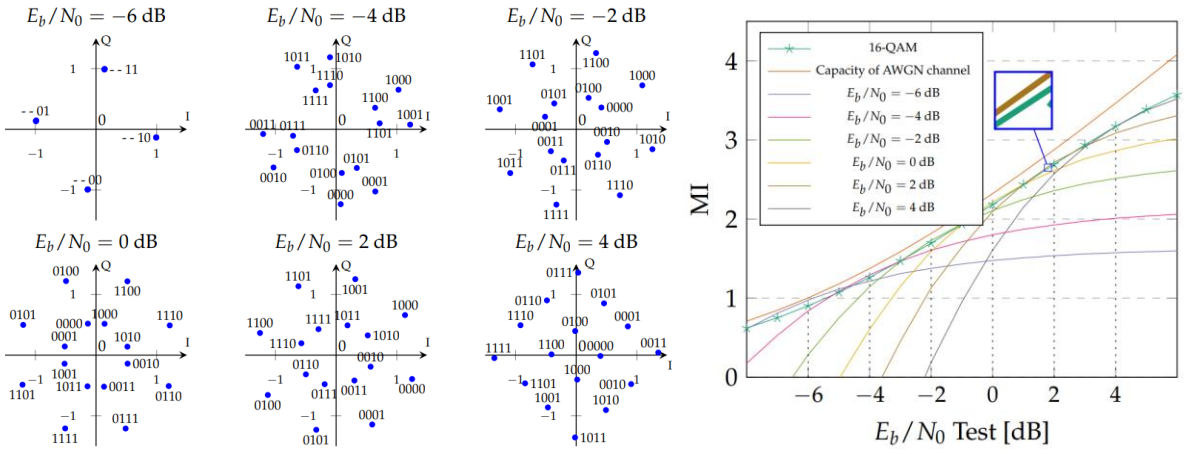

Abbildung 9: AE Konstellationen für verschiedene SNRs

8.3.3 Hardware Implementierung

Hauptaufgabe der RPTU im Projekt FunKI war die effiziente Hardware-Implementierung der ANN-basierten Kommunikationsblöcke. Daher wurde im Rahmen dieses Projekts eine optimierte Hardware-Architektur des AEs entworfen, die auf dem FPGA nachtrainiert werden kann, um sich an veränderliche Kanalbedingungen anzupassen. Um die Bandbreite zu reduzieren und den Kommunikationsaufwand eines Rückkanals vom Empfänger zum Sender zu eliminieren, fixierten wir die Konstellationen des Senders nach dem Training in der Software und implementierten nur den Empfänger als trainierbares künstliches neuronales Netz (ANN) auf dem FPGA. Zusammengefasst wird das AE-System in der Software gemeinsam trainiert, kann aber durch erneutes Training des Demappers in der Hardware feinabgestimmt und an reale Kanalbedingungen angepasst werden. Als Zielplattform wurde der Ultra96-V2 gewählt, da er energieeffizient sowie kostengünstig ist und einen ARM-Cortex A53-CPU bietet. Im Folgenden bezeichnen wir die CPU als Programmable-System (PS) und den ZU3EG-FPGA als Programmable-Logic (PL). Das PS wird verwendet um Eingabesequenzen zu generieren indem zufällige Bit-Vektoren generiert werden, die auf ein komplexes Symbol abgebildet werden, das von der Konstellation des trainierten Mappers bestimmt wird und über ein Kanalmodell übertragen wird. Darüber hinaus stellt das PS Labels für das Training bereit und berechnet die erreichte Bitfehlerrate (BER) und die Mutual Information. Die Hardware-Architektur besteht aus zwei separaten Modulen für Inferenz und Training. Das Inferenzmodul erhält als Eingabe die verrauschte Sequenz und detektiert die gesendeten Symbole, während das Trainingsmodul zusätzlich das Label als Eingabe erhält und die Gewichte mithilfe von Backpropagation und Gradientenabstieg anpasst. Für das Design der beiden Module wurden Vivado High-Level Synthesis (HLS) 2019.2 und Teile der Xilinx FINN-Bibliothek verwendet. Die Hardware-Architektur des Trainingsmoduls ist in Abbildung 10 dargestellt. Wichtige Merkmale dieses Moduls sind:

- Separat einstellbare Parallelisierungsgrade (DOPs) des Inferenz- und Trainingsmoduls, welche entsprechend den Anwendungsanforderungen angepasst werden können
- Eine vollständig gepipelinete on-chip Hardware-Architektur, die geringe Latenz und hohen Durchsatz ermöglicht

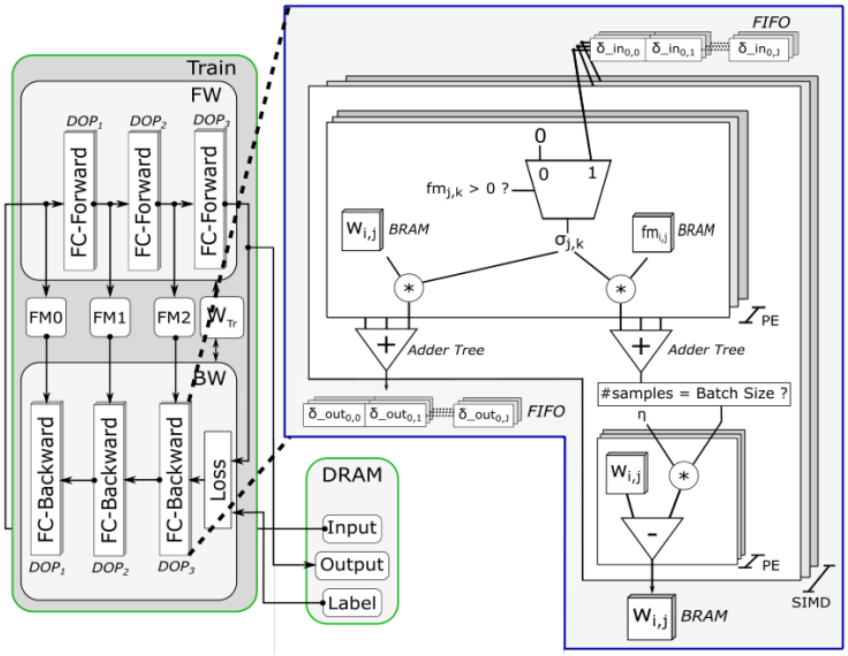

Abbildung 10: Hardware-Architektur des Trainings-Moduls

8.3.4 Cross-Layer Explorierungs Framework

Um die Flexibilität von FPGAs und unserer anpassbaren Architektur effizient zu nutzen, wurde ein Framework implementiert, das auf dem PS des FPGA ausgeführt werden kann, um automatisch den optimalen DOP für das Inferenz- und Trainingsmodul für spezifische Anwendungsanforderungen zu bestimmen. Dies wird dadurch ermöglicht, dass das Framework zur Laufzeit den Bitstream mit dem Parallelisierungsgrad lädt, der den aktuellen Anforderungen entspricht. Als Eingabe erhält das Framework:

- Das Ziel, z.B. minimale Inferenzlatenz, minimale Konvergenzlatenz, minimaler Inferenzleistungsverbrauch oder minimaler Trainingsleistungsverbrauch
- Anwendungsanforderungen in Form von Latenz- und Leistungsanforderungen sowie optional eine Ziel-BER
- Hardware-Beschränkungen basierend auf den verfügbaren Ressourcen der Zielplattform.

Die Ausgaben des Frameworks sind die DOPs für das Inferenz- und das Trainingsmodul, die für das gegebene Ziel optimiert sind und die Anforderungen der Anwendung erfüllen. Um die kommunikationsbezogenen Metriken wie SNR und BER mit den tatsächlichen Eigenschaften der zugrunde liegenden Hardware zu verknüpfen, greift das Framework intern auf eine Datenbank von Charakteristiken der Hardware-Module zu. Diese Merkmale umfassen hardwarebezogene Eigenschaften wie Ressourcennutzung, Latenz und Leistungsverbrauch sowie kommunikationstechnische Eigenschaften wie Konvergenzzeiten und SNRs. Das Framework ist Teil der Task T4.b.TUK-7 "Frameworks zur semi-automatisierten Generierung der Beschleuniger" in AP-4.

8.3.5 Implementierungsergebnisse

Im Folgenden vergleichen wir die Ressourcennutzung und die Latenz des Inferenz- und Trainingsmoduls für verschiedene DOPs, welche verschiedenen Anforderungen und Zielen in unserem Framework entsprechen. Die Latenz wird auf dem Board für mehrere Iterationen gemessen, um die durchschnittliche Latenz für die Verarbeitung eines Samples zu berechnen. Der DOP wird soweit erhöht bis das Modul vollständig parallelisiert ist oder das Ressourcenlimit auf dem Ultra96-V2 erreicht ist. Die Implementierungsergebnisse entsprechen der Task T.4.b.TUK-7 "Prototypische Implementierung von Accelerator 2".

In Abbildung 11 wird die Ressourcennutzung für DOPs bis zu 256 für das Inferenzmodul und 32 für das Trainingsmodul dargestellt. Es ist zu erkennen, dass DSPs die meistgenutzte Ressource sind, da sie zur Berechnung der MAC-Operationen verwendet werden. Während das Inferenzmodul mit einem DOP von 256 parallelisiert werden kann, kann das Trainings-Modul lediglich mit einem DOP von 32 parallelisiert werden. Die höhere Komplexität dieses Moduls lässt sich hauptsächlich durch das Vorhandensein von Forward- und Backwardpass für das Training, eine höhere Quantisierung der Gewichte und der Speicherung von Featuremaps für die Gradientenberechnung erklären. Zusammenfassend lässt sich feststellen, dass das Inferenz- und das Trainingsmodul des ANNs mit hoher Parallelität auch auf einem kostengünstigen FPGA implementiert werden können.

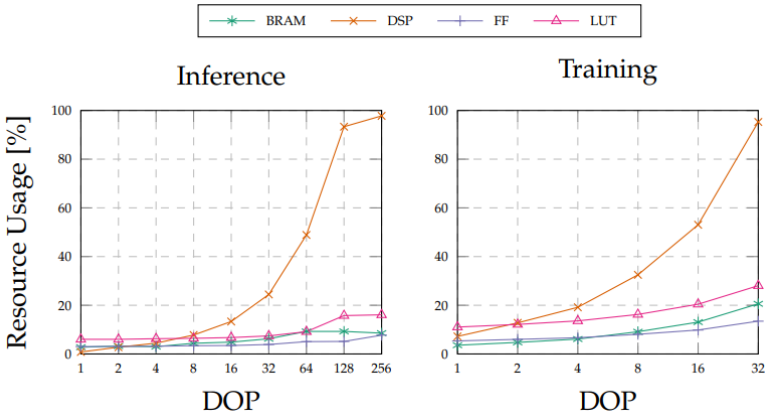

Abbildung 11: Ressourcennutzung auf dem Ultra96-V2 für verschiedene DOPs

Wie sich die verschiedenen Parallelisierungsgrade auf Latenz, Leistungsaufnahme und Energieverbrauch auswirken, ist in Abbildung 12 dargestellt.

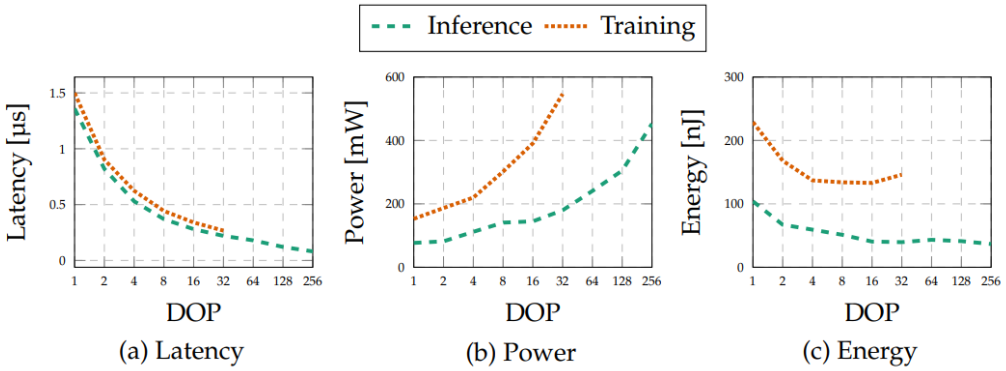

Abbildung 12: Latenz (a), Leistungsaufnahme (b) und Energieverbrauch (c) auf dem Ultra-96V2

Es wird deutlich, dass sich die Latenz mit höherem DOP bis auf 81 ns für das Inferenz- und auf 267 ns für das Trainingsmodul reduzieren lässt. Dadurch können auch strenge Latenzanforderungen, wie die von ultra reliable low-latency communicaiton (uRLLC) erfüllt werden. Die in Abbildung 12 dargestellte dynamische Leistungsaufnahme wurde mit Hilfe des externen Leistungsmessers Voltcraft VC870 als Differenz zwischen statischer Leistung und Gesamtleistung bestimmt. Der Energieverbrauch baut auf dieser Größe aus und ergibt sich aus dem Produkt zwischen Leistungsaufnahme und Verarbeitungszeit pro Symbol. Die Graphen zeigen, dass die Leistungsaufnahme mit steigendem DOP zunimmt, während sich der Energieverbrauch leicht reduziert. Die hier dargestellten Ergebnisse waren Grundlage für das Cross-Layer Explorierungs Framework. Insbesondere der Trade-off zwischen geringer Latenz und niedriger Leistungsaufnahme kann von dem Framework genutzt werden, um Anwendungsanforderungen durch Anpassung der DOPs zu erfüllen.

8.3.6 Vergleich mit anderen Plattformen

Um die Vorteile unseres FPGA-Designs zu demonstrieren, wurde unsere FPGA-Implementierung mit Implementierungen des selben Netzes auf der GPU *Nvidia RTX 2080*, der eingebetteten GPU *Nvidia Jetson AGX* und der eingebetteten CPU *ARM Cortex-A53* verglichen. Dabei wurden sowohl Durchsatz, Energieeffizienz und Leistungsverbrauch für Inferenz- und Trainingsmodul verglichen. Für einen fairen Vergleich wurde die Batch-Size von GPU und CPU erhöht, bis der Speicher der Plattformen voll ausgenutzt war. Es ist jedoch zu beachten, dass für die Anwendung des

Demappings von Kommunikationssymbolen in Echtzeitkommunikationssystemen solch hohe Batch-Sizes nicht praktikabel sind, da das Buffern einer solch großen Datenmenge am Empfänger zu hohen Latenzen führen würde.

Durchsatz und Energieverbrauch der verschiedenen Plattformen werden in Abbildung 13 für verschiedene Batch-Sizes verglichen.

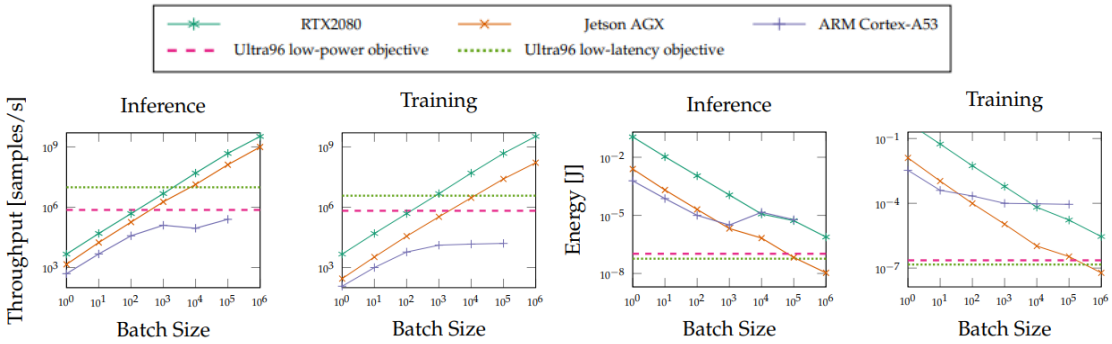

Abbildung 13: Vergleich von Durchsatz und Energieverbrauch von FPGA, GPU und CPU

Der FPGA erzielte für ein Batch-Size von 1 einen Durchsatz von $1{,}23 \cdot 10^7$ Samples/s für die Inferenz und $3{,}75 \cdot 10^6$ Samples/s für das Training und übertrifft damit die Nvidia RTX 2080 GPU um den Faktor 2000 bzw. 3800 für die gleiche Batch-Size. Die hohe Parallelität der GPU wurde erst bei sehr hoher Batch-Size voll ausgenutzt und führte erst ab einer Batch-Size von 10.000 zu einem höheren Durchsatz im Vergleich zum FPGA.

In Bezug auf den Energieverbrauch übertraf der FPGA alle anderen Plattformen bis zu einer Batch-Size von 100.000, bei der nur die Nvidia Jetson AGX einen geringeren Wert lieferte. Für sehr kleine Batch-Sizes gibt es einen erheblichen Performanz Unterschied zwischen dem Ultra96 und allen anderen Plattformen. So verbrauchte er etwa 5000 Mal weniger Energie als die eingebettete CPU. Diese Ergebnisse zeigen, dass General Purpose-Prozessoren nicht immer die optimale Plattform für die Inferenz und das Training von ANNs bieten. Im Gegensatz zu CPUs und GPUs ermöglicht die Flexibilität von FPGAs eine hohe Spezialisierung, da der DOP an die Einschränkungen und Anforderungen angepasst werden kann. Insbesondere in Fällen von energiebeschränkten eingebetteten Systemen oder Anwendungen in denen die Datenverarbeitung sequentiell erfolgen muss und somit großen Batch-Sizes nicht praktikabel sind, bieten FPGAs eine hervorragende Performanz in Bezug auf Durchsatz und Energieeffizienz.

8.3.7 Vergleich mit konventionellem Demapping

Im Rahmen von Task T4.c.TUK-1 "Vergleich der Acceleratoren mit klassischen Ansätzen" erfolgte in diesem Projekt ein Vergleich des ANN-basierten Demappings mit klassischen Demapping Methoden. Dazu wurde der von Robertson et al. präsentierte Soft-Demapping-Algorithmus implementiert[9]. Dieser ersetzt exponentielle und logarithmische Funktionen durch einfache arithmetische Operationen und basiert auf folgender Gleichung:

$$llr(b_k|s_r) = \frac{1}{2\sigma^2}\left(\min_i\left(s_r - c_{i,k=0}\right)^2 - \min_i\left(s_r - c_{i,k=1}\right)^2\right)$$

Wobei s_r das empfangene Symbol beschreibt, b_k das k-te Bit und $c_{i,k=0}$ sowie $c_{i,k=1}$ die Konstellationssymbole deren k-te Bits *0* oder *1* sind darstellen.

[9] P. Robertson, E. Villebrun, und P. Hoeher, „A comparison of optimal and sub-optimal MAP decoding algorithms operating in the log domain“, in *Proceedings IEEE International Conference on Communications ICC '95*, Seattle, WA, USA: IEEE, 1995, S. 1009–1013. doi: 10.1109/ICC.1995.524253.

Der konventionelle Ansatz wird in Tabelle 3 mit dem ANN-basierten Demapper mit geringer Parallelisierung (AE low-power) und hoher Parallelisierung (AE low-latency) verglichen.

Tabelle 3: Vergleich des AE mit klassischem Demapping

Ansatz	BRAM	DSP	FF	LUT	Latency [s]	Throughput [bit/s]	Power [W]
AE low-power	6	3	4283	4248	$1.36 \cdot 10^{-6}$	$2.94 \cdot 10^{6}$	$7.68 \cdot 10^{-2}$
AE low-latency	18.5	352	10895	11343	$8.10 \cdot 10^{-8}$	$4.92 \cdot 10^{7}$	$4.53 \cdot 10^{-1}$
Conventional	0	1	1042	1107	$5.33 \cdot 10^{-8}$	$3.00 \cdot 10^{8}$	$5.50 \cdot 10^{-2}$

Aus der Tabelle wird deutlich, dass die Komplexität eines herkömmlichen Demappers auf einem FPGA im Vergleich zu einem System basierend auf ANNs viel geringer ist. Dennoch zeigen die Ergebnisse, dass der AE vergleichbare Leistung erzielen kann, wenn er für die jeweilige Metrik optimiert wurde. Die auf Latenz optimierte AE-Implementierung erreichte eine Latenz in derselben Größenordnung wie der herkömmliche Soft-Demapper. Das Gleiche gilt für den auf Leistung optimierte AE bezüglich der Leistungsaufnahme.

Der erzielte Durchsatz des Soft-Demappers ist jedoch im Vergleich zum AE viel höher, und könnte sogar noch weiter gesteigert werden, indem mehrere Module parallel instanziiert werden, was aufgrund der geringen Ressourcennutzung möglich ist. Dies zeigt, dass FPGA-Implementierungen von ANN-basierten Systemen derzeit hinsichtlich der Implementierungskomplexität beim Demapping hinter hochoptimierten herkömmlichen Systemen zurückbleiben.

Allerdings kann das ANN-basierte System in der Theorie viel komplexere Aufgaben ausführen als nur das Demapping. Beispielsweise wurde in von Caciularu et al. gezeigt, dass ein kleines ANN mit nur zwei Faltungsschichten in der Lage ist, blinde Kanalentzerrung durchzuführen[10]. Darüber hinaus haben Fischer et al. gezeigt, dass selbst ein komplexer Wiener-Filter von einem ANN-basierten Ansatz hinsichtlich der nachrichtentechnischen Performanz übertroffen werden kann[11]. Diese Ergebnisse zeigen, dass auch komplexere nachrichtentechnische Algorithmen von einem Ende-zu-Ende AE-System durchgeführt werden können und das reine Demapping nur einen ersten Schritt darstellt.

Eine Analyse welche Verarbeitungsblöcke in ein AE-basiertes System einbezogen werden können, war nicht Teil dieses Projekts, könnte jedoch für zukünftige Arbeiten sehr interessant sein. Dadurch könnte ein komplexeres AE-basiertes System implementiert und mit klassischen Algorithmen verglichen werden, um das volle Potenzial von AEs im Bereich der Kommunikationstechnik aufzuzeigen.

8.3.8 Hybrider Ansatz

Ein weiteres Ziel des Projekts war es ANN-basierte Ansätze mit klassischen Algorithmen zu kombinieren. Hierzu wurde eine Methodik entwickelt, die es ermöglicht, die hohe Anpassungsfähigkeit des ANN-basierten Ansatzes mit der Effizienz des konventionellen Demappings zu verbinden. Diese Methodik ist in Abbildung 14 dargestellt.

[10] A. Caciularu und D. Burshtein, „Blind Channel Equalization using Variational Autoencoders“, *ArXiv180301526 Cs Eess Math*, März 2018, Zugegriffen: 14. September 2021. [Online]. Verfügbar unter: http://arxiv.org/abs/1803.01526

[11] M. B. Fischer *u. a.*, „Wiener Filter versus Recurrent Neural Network-based 2D-Channel Estimation for V2X Communications“, *ArXiv210203163 Cs Math*, Mai 2021, Zugegriffen: 14. September 2021. [Online]. Verfügbar unter: http://arxiv.org/abs/2102.03163

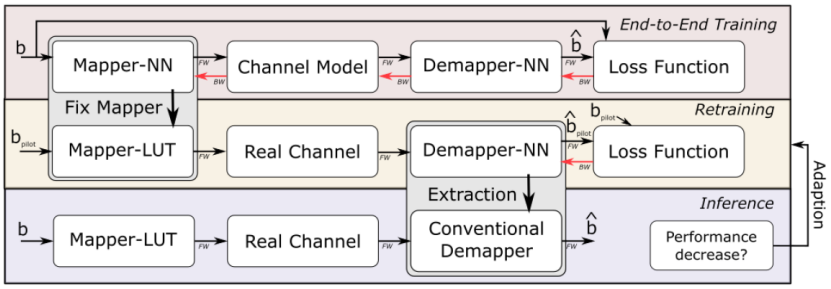

Abbildung 14: Hybrider Demapping Ansatz

Der hybride Ansatz lässt sich in drei Schritte unterteilen: Das End-to-End Training, das Retraining und die Inferenz. Im ersten Schritt wird das ANN, welches den Mapper repräsentiert und das ANN des Demappers gemeinsam Ende-zu-Ende über ein Kanalmodell trainiert. Das Ergebnis ist eine gelernte Symbolkonstellation, die möglichst robust gegenüber dem Rauschen des Kanalmodells ist. Im nächsten Schritt wird diese Konstellation in Form einer Lookup-Tabelle auf dem FPGA implementiert. Im Gegensatz dazu wird der Demapper inklusive der gelernten Gewichte als trainierbares ANN auf dem FPGA implementiert. Dadurch kann der Demapper für einen echten Kanal nachtraininert werden, um die nachrichtentechnische Performanz zu erhöhen. Im letzten Schritt werden die gelernten Informationen über den realen Kanal aus dem Demapper-ANN extrahiert. Basierend auf den gelernten Entscheidungsschwellen werden Konstellationspunkte ermittelt, welche in einen konventionellen Demapper eingebunden werden, um eine effiziente Inferenz zu ermöglichen.

Zusammengefasst dient das Demapper-ANN also dazu, Diskrepanzen zwischen Kanalmodell und echtem Kanal auszugleichen, während gleichzeitig konventionelle Algorithmen für die effiziente Inferenz genutzt werden können.

Die Ergebnisse dieser neuartigen Methodik wurden im Jahr 2022 auf dem Reconfigurable Architectures Workshop publiziert[12].

8.3.9 Demonstrator

Im Rahmen von AP5 wurde in enger Zusammenarbeit mit Creonic auf Basis der AE-Implementierung ein Demonstrator entwickelt. Hierzu wurde zunächst eine geeignete FPGA Plattform ausgewählt (Xilinx XCZU7EV-2FFVC1156) und die entsprechenden Schnittstellen bestimmt. Des Weiteren wurden die Randbedingungen des Demonstrator-Szenarios festgelegt: Zur Visualisierung der Adaption des Demapper-ANNs wird dieses zunächst offline für einen AWGN Kanal trainiert, anschließend wird die Adaption auf dem FPGA für einen Phasen-Offset Kanal demonstriert.

Eine Funktion des Demonstrators ist die Visualisierung von Entscheidungsschwellen des Demappers, wie in Abbildung 15 dargestellt. Hiermit lässt sich visualisieren, wie sich der Demapper dem Kanal anpasst und wie die Entscheidungsschwellen mit dem Phasenoffset rotieren. In der Abbildung sind die Entscheidungsschwellen vor und nach dem Nachtrainieren für den Phasenoffsetkanal dargestellt. Es ist zu erkennen, dass die Entscheidungsschwellen nach dem Nachtrainieren um $\pi/4$ rotiert sind. Dies entspricht dem Phasenoffset des Kanals, der somit kompensiert werden konnte.

[12] J. Ney, B. Hammoud and N. Wehn, "A Hybrid Approach combining ANN-based and Conventional Demapping in Communication for Efficient FPGA-Implementation," 2022 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW), Lyon, France, 2022, pp. 92-95, doi: 10.1109/IPDPSW55747.2022.00024

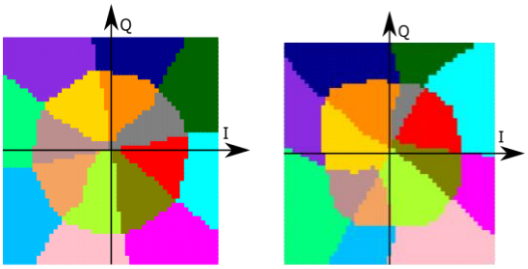

Abbildung 15: Entscheidungsschwellen des Demappers. Links: Vor dem Nachtrainieren, Rechts: Nach dem Nachtrainieren

Der entwickelte Demonstrator wurde zum einen in internen FunKI Meetings zu Visualisierungszwecken verwenden und diente zum anderen der Außendarstellung auf weiteren BMBF Events wie beispielsweise der Berlin 6G Conference.

8.3.10 Zusammenfassung

Im Rahmen des FunKI Projekts wurde eine effiziente Hardware-Architektur eines AE-basierten Kommunikationssystems implementiert, die in der Lage ist, sich Kanalschwankungen anzupassen, indem das Empfänger-ANN zur Laufzeit nachtrainiert wird. Unsere FPGA-Implementierung ist äußerst flexibel, da sie variable Quantisierung und einen flexiblen Parallelisierungsgrad unterstützt, der individuell für Inferenz und Training konfiguriert werden kann. Darüber hinaus wurde ein Framework entworfen, das die Lücke zwischen der Kommunikationssystem-Ebene und der Hardware-Ebene schließt, indem es den Parallelisierungsgrad je nach Anwendungsanforderung anpasst. Des Weiteren, wurden die Vorteile unserer Hardware-Architektur und des FPGA als Implementierungsplattform gezeigt, indem wir unsere Implementierung mit verschiedenen General-Purpose-Prozessoren verglichen haben. Dabei erreichte unsere FPGA-basierte AE-Implementierung einen um das 2000-fache höhere Durchsatzrate als eine leistungsstarke GPU, verbrauchte 5-mal weniger Energie als eine eingebettete CPU und ist im Vergleich zu einer eingebetteten GPU für kleine Batch-Sizes um das 5800-fache energieeffizienter. Zudem war die AE-Implementierung Grundlage für einen FPGA-basierten Demonstrator, welcher im Rahmen von AP-5 entwickelt und zur Außendarstellung des Projekts auf diversen Events genutzt wurde.

9. Voraussichtlicher Nutzen, insbesondere Verwertbarkeit der Ergebnisse

Die Ergebnisse des Projekts wurden durch wissenschaftlichen Publikationen und durch Präsentationen auf Konferenzen verbreitet. Somit wurde die wissenschaftliche Verwertung der durchgeführten Untersuchungen und deren Ergebnissen sichergestellt. Des Weiteren erfolgte innerhalb des Projekts eine enge Zusammenarbeit mit Industriepartnern. Im Fall der RPTU fand vor allem eine enge Kooperation mit der Firma Creonic statt. Die innerhalb diesen Projektes gesammelten Erkenntnisse können damit auch in die Entwicklung zukünftiger Produkte im nachrichtentechnischen Bereich einfließen.

Im speziellen können die Ergebnisse bezüglich des LDPC-Decoders dazu genutzt werden effizientere Decoder Architekturen zu implementieren, was von hoher Bedeutung für die Basisbandverarbeitung zukünftiger Kommunikationsstandards ist. Die Ergebnisse der Autoencoder-Implementierung können vor allem in zukünftige Forschungsarbeiten einfließen, da sie aufzeigen in welchen Bereichen noch Forschungsbedarf besteht. Hier sollte vor allem eine Reduzierung der Berechnungskomplexität im Vordergrund stehen, um im Vergleich mit klassischen Algorithmen nicht nur aus algorithmischer Sicht sondern auch aus Sicht der Hardware-

Implementierung konkurrenzfähig zu sein. Der im Projekt in Zusammenarbeit mit Creonic entwickelte Demonstrator wird auch zukünftig für die Präsentation der wissenschaftlichen Erkenntnisse des Projekts auf Messen und Konferenzen genutzt und trägt damit zur wissenschaftlichen Verwertung der durchgeführten Untersuchungen bei.

10. Fortschritte auf dem Gebiet des Vorhabens bei anderen Stellen

Im Projekt FunKI wurde regelmäßig der Stand der Wissenschaft ermittelt und in die Forschungsarbeiten der einzelnen Partner integriert. Hierzu haben regelmäßig Informationsrecherchen auf den einschlägigen Onlineplattformen stattgefunden.

Ein Austausch mit anderen Stellen fand im Rahmen von projektinternen Workshops sowie wissenschaftlichen Konferenzen statt.

11. Erfolgte und geplante Veröffentlichungen

Im Rahmen des Projekts FunKI wurde von Seiten der RPTU eine Vielzahl an wissenschaftlichen Publikationen sowohl in Magazinen als auch auf Konferenzen veröffentlicht.

Die erfolgten Konferenzbeiträge werden in Tabelle 4 aufgelistet.

Tabelle 4: Konferenzbeiträge der RPTU im Rahmen von FunKI

Autoren	Titel	Konferenz	Datum
J. Ney, S. Dörner, M. Herrmann, M. H. Sadi, J. Clausius, S. ten Brink, N. Wehn	FPGA-based Trainable Autoencoder for Communication Systems	International Symposium on Field-Programmable Gate Arrays	Februar 2022
J. Ney, B. Hammoud, N. Wehn	A Hybrid Approach combining ANN-based and Conventional Demapping in Communication for Efficient FPGA-Implementation	29th Reconfigurable Architectures Workshop	Mai 2022
T. Monsees, D. Wübben, A. Dekorsy, O. Griebel, M. Herrmann, N. Wehn	Finite-Alphabet Message Passing using only Integer Operations for Highly Parallel LDPC Decoders	IEEE SPAWC 2022	Juni 2022

Die Publikationen in Magazinen sind in Tabelle 5 gelistet.

Tabelle 5: Beiträge der RPTU in wissenschaftlichen Magazinen im Rahmen von FunKI

Autoren	Titel	Magazin	Datum
J. Ney, B. Hammoud, S. Dörner, M. Herrmann, J. Clausius, S. ten Brink, N. Wehn	Efficient FPGA Implementation of an ANN-Based Demapper using Cross-Layer Analysis	Electronics, MDPI	April 2022
L. Johannsen, C. Kestel, O. Griebel, T. Vogt, N. Wehn	Partial Order-Based Decoding of Rate-1 Nodes in Fast Simplified Successive-Cancellation List Decoders for Polar Codes	Electronics, MDPI	Februar 2022
T. Monsees, O. Griebel, M. Herrmann, D. Wübben, A. Dekorsy, N. Wehn	Minimum-Integer Computation Finite Alphabet Message Passing Decoder: From Theory to Decoder Implementations towards 1 Tb/s	Entropy, MDPI	Oktober 2022