(19) **Bundesrepublik Deutschland**
Deutsches Patent- und Markenamt

(10) **DE 695 31 482 T2** 2004.07.01

(12) # Übersetzung der europäischen Patentschrift

(97) **EP 0 728 393 B1**
(21) Deutsches Aktenzeichen: **695 31 482.3**
(86) PCT-Aktenzeichen: **PCT/US95/11755**
(96) Europäisches Aktenzeichen: **95 933 822.9**
(87) PCT-Veröffentlichungs-Nr.: **WO 96/008904**
(86) PCT-Anmeldetag: **14.09.1995**
(87) Veröffentlichungstag der PCT-Anmeldung: **21.03.1996**
(97) Erstveröffentlichung durch das EPA: **28.08.1996**
(97) Veröffentlichungstag der Patenterteilung beim EPA: **13.08.2003**
(47) Veröffentlichungstag im Patentblatt: **01.07.2004**

(51) Int Cl.[7]: **H04L 25/03**

(30) Unionspriorität:
305787 **14.09.1994** **US**

(84) Benannte Vertragsstaaten:
AT, BE, CH, DE, DK, ES, FR, GB, GR, IE, IT, LI, LU, NL, PT, SE

(73) Patentinhaber:
Ericsson Inc., Research Triangle Park, N.C., US

(72) Erfinder:
DENT, W., Paul, S-240 36 Stehag, SE

(74) Vertreter:
HOFFMANN · EITLE, 81925 München

(54) Bezeichnung: **GLEICHZEITIGE DEMODULATION UND DEKODIERUNG EINES DIGITAL MODULIERTEN FUNKSIGNALS**

Anmerkung: Innerhalb von neun Monaten nach der Bekanntmachung des Hinweises auf die Erteilung des europäischen Patents kann jedermann beim Europäischen Patentamt gegen das erteilte europäische Patent Einspruch einlegen. Der Einspruch ist schriftlich einzureichen und zu begründen. Er gilt erst als eingelegt, wenn die Einspruchsgebühr entrichtet worden ist (Art. 99 (1) Europäisches Patentübereinkommen).

Die Übersetzung ist gemäß Artikel II § 3 Abs. 1 IntPatÜG 1991 vom Patentinhaber eingereicht worden. Sie wurde vom Deutschen Patent- und Markenamt inhaltlich nicht geprüft.

Beschreibung

HINTERGRUND

[0001] Die vorliegende Erfindung betrifft allgemein ein Verfahren und eine Vorrichtung zum gleichzeitigen Demodulieren und Decodieren eines digital modulierten Funksignals bzw. Radiosignals, wie beispielsweise eines Signals eines zellularen drahtlosen bzw. schnurlosen Telefons gemäß dem Oberbegriff des Anspruchs 1 und des Anspruchs 16.

[0002] In EP 0 594 551 A1 ist eine Demodulation von digital modulierten Signalen beschrieben, und zwar insbesondere von denjenigen, die über Kanäle empfangen werden, die an einer oder mehreren Beeinträchtigungen leiden, wie beispielsweise einem Fading, einer Zwischensymbolinterferenz, einem Frequenzfehler, einer Verzerrung, und zwar entweder temporär konstant oder mit der Zeit variierend.

[0003] Weiterhin ist in Vol. 2 aus 4, 29. November 1993, Institute of Electrical and Electronic Engineers, Seiten 814–818, Jung-Fu Cheng et al.: "Complexity-reduced Multilevel Coding with Rate-compatible Punctured Convolutional Codes" ein Codierungsentwuf basierend auf einer Mehrfachpegel-Codierkonstruktion mit ratenkompatiblen durchbrochenen Faltungscodes als Aufbaublöcke beschrieben.

[0004] Weiterhin ist in US-A-5,331,666 ein Demodulator für eine adaptive maximale Ähnlichkeit zum Demodulieren von digitalen Funksignalen auf einem Kanal beschrieben, der Beeinträchtigungen enthält, ohne Verwendung von Kanalmodellen für eine Erzeugung von Signalvorhersagen.

[0005] Eine Kombination aus Modulation und Codierung, und gegensätzlich dazu aus Demodulation und Decodierung, ist eine Technik, die mit Veröffentlichungen von Ungerboeck assoziiert worden ist. So genante Ungerboeck-Codes können verwendet werden, um Datenbits für eine Übertragung in einer Reihe von Funksignal-Phasenänderungen zu codieren. Mehr als eine Phasenänderung pro übertragenem Datenbit wurde allgemein ins Auge gefasst, und ebenso hing jede Phasenänderung von mehr als einem Datenbit ab. Ein Demodulieren und ein Decodieren eines solchen Signals enthielt eine Hypothesenbildung einer Sequenz von Bits, die bei dem Sender codiert werden, unter Verwendung eines Modells des Codierprozesses zum Vorhersagen, welche Phasenänderungen bei dem Empfänger entsprechend einer jeweiligen Hypothese beobachtet werden sollten, und unter Vergleichen des tatsächlich empfangenen Signals mit dem Hypothesesignal, um die Fehlanpassung zu bestimmen und somit "Stimmen" für und gegen eine jeweilige Hypothese zu sammeln. Die Hypothesesequenz, die mit den meisten Stimmen endet, wurde dann dafür bestimmt, dass sie die übertragene ist.

[0006] Ein Problem bei einer Ungerboeck-Codierung besteht darin, dass die Daten in einer Reihe von sequentiellen Signaländerungen codiert werden. Auf Fading-Kanälen kann eine Schwundresonanz eine gesamte Reihe von solchen Signaländerungen auswischen und somit ein gesamtes Datenbit. Ein solcher nicht verschachtelter Code ist nicht dazu fähig, sogar kurze Fadings bzw. Schwunde zu überbrücken.

[0007] Zum Bekämpfen eines Fadings ist es wünschenswert, ein Verschachteln zu verwenden, so dass benachbarte Symbole aus dem Codierer zeitlich gut beabstandet übertragen werden, und somit dann, wenn sie beim Decodierer wieder angeordnet werden, benachbarte Symbole in den Decodierer von Signalsegmenten gekommen sind, die nicht im selben Schwundbereich sind. Nach einem Entschachteln beim Empfänger wird eine Schwundresonanz bezüglich der Zeit ausgebreitet werden, so dass Fehler nicht angehäuft werden. Unglücklicherweise ist ein Ungerboeck-Decodierer für verschachtelte, codierte Signale gewöhnlich zu kompliziert, um berücksichtigt zu werden, da seine Komplexität bei Symbolen, die der Code überspannt, exponentiell von der Gesamtzeit abhängt.

[0008] Eine alternative herkömmliche Technik für codierte, verschachtelte und modulierte Signale ist eine separate Demodulation, der ein Entschachteln und ein separates Decodieren folgt. Wenn der Übertragungskanal an einer Zeitdispersion leidet, d. h. an verzögerten Echos des übertragenen Signals, kann ein Entzerrer dazu verwendet werden, eine adäquate Demodulationsleistung zu ergeben. Ein Entzerrer für eine sequentielle maximale Wahrscheinlichkeit, wie beispielsweise derjenige, der zum Beispiel in dem US-Patent US-A-5,331,666 mit dem Titel "Adaptive Maximum Likelihood Demodulator", eingereicht am 8. Juni 1992, und in dem US-Patent US-A-5,335,250, das zu der Patentfamilie der europäischen Patentanmeldung EP-A-594 551 gehört, mit dem Titel "Bidirectional Demodulation", eingereicht am 22. Oktober 1992, hat eine Komplexität, die exponentiell davon abhängt, wie viele Symbole es gibt, die die Echos verzögerten. Ein solcher Entzerrer kann eher "weiche" als "harte" Entscheidungen liefern, die nach einem Entschachteln dem Fehlerkorrektur-Decodierer präsentiert werden. Ein Fehlerkorrektur-Decodierer für Faltungscodes hat auch eine Komplexität, die exponentiell von der Beschränkungslänge des Codes abhängt.

[0009] Der Nachteil von einer separaten Demodulation und Decodierung, und zwar selbst mit einem Entzerrer, der weiche Entscheidungen liefert, besteht darin, dass die Demodulatorleistung bei niedrigen Signal-zu-Rausch-Verhältnissen fehlschlagen kann. Dies entsteht insbesondere in dem Fall von zeitvariierenden Kanälen, wo die Phase des Signals durch beispielsweise Dopplereffekte und eine Mehrwegeausbreitung zerstört ist. Für solche Kanäle kann es erwünscht sein, Daten durch differentielles Ändern der Phase des Signals von einer Symbolperiode zur nächsten auf das Funksignal zu modulieren, so dass es durch Vergleichen von

Signalwerten demoduliert werden kann, die nur um ein Symbol voneinander beabstandet sind, für eine ausreichend kurze Zeitspanne, so dass die Kanalphase sich über dieser Zeit nicht signifikant ändert. Unglücklicherweise gibt es normalerweise einen Verlust in der Größenordnung von 3 dB beim Vergleichen eines rauschbehafteten empfangenen Signalwerts mit einem anderen rauschbehafteten Signalwert, um das übertragene Symbol zu bestimmen, wenn eine differentielle Demodulation verwendet wird. Dieser Verlust kann im Fehlerkorrektur-Decodierprozess nach dem Demodulationsprozess nicht entdeckt werden, wenn herkömmliche separate Demodulations- und Decodierprozesse verwendet werden.

[0010] Ein weiterer Defekt eines herkömmlichen separaten Demodulierens und Decodierens besteht darin, dass der separate Demodulator aufgrund eines Rauschens Fehler machen kann und eine falsche Sequenz von Bits als die "beste" Ausgabe auswählen kann. Dieses einzelne Fehlerereignis wird, wenn es entschachtelt wird, zu mehreren separaten Bitfehlerereignissen, so dass der Fehlerkorrektur-Decodierer nun mit einer größeren Anzahl von Fehlern fertig werden muss. Somit können Demodulatoren für eine maximale Wahrscheinlichkeit, denen ein Entschachteln folgt, und dann Faltungsdecodierer für eine maximale Wahrscheinlichkeit, an einer Verschlechterung der Leistungsfähigkeit leiden.

[0011] Demgemäß wäre es für den Fehlererfassungs/Korrektur-Decodierprozess wünschenswert, mit dem Demodulationsprozess in einem einzigen Demodulator für eine maximale Wahrscheinlichkeit enthalten zu sein, so dass die Demodulation durch das Decodieren unterstützt werden kann, um Fehler aufgrund eines Rauschens zu vermeiden.

ZUSAMMENFASSUNG

[0012] Diese und andere Probleme, Nachteile und Beschränkungen von herkömmlichen Demodulations- und Decodierverfahren und Systemen werden durch Verwenden von Verfahren und Systemen gemäß der vorliegenden Erfindung überwunden. Dabei wird zuerst ein zu übertragender digitaler Datenstrom zuerst einer Fehlerkorrektur-Codierung unter Verwendung von beispielsweise einer Faltungscodierung unterzogen. Die codierten Datenbits werden dann gemäß einem Verschachtelungsalgorithmus bezüglich der Zeit verteilt, der auch eine beschränkte Anzahl von Symbolen einfügen kann, die dem Empfänger im Voraus bekannt sind, wie beispielsweise Synchronisationssymbole. Die codierten und verschachtelten Datenbits werden dann auf ein Funksignal moduliert, das in Bursts übertragen werden kann.

[0013] Diese und andere Probleme, Nachteile und Beschränkungen von herkömmlichen Demodulations- und Decodierverfahren und -systemen werden durch Verwenden eines Verfahrens gemäß der vorliegenden Erfindung überwunden, wie es im unabhängigen Anspruch 1 definiert ist, und einer Vorrichtung gemäß der vorliegenden Erfindung, wie sie im unabhängigen Anspruch 16 definiert ist. Dabei wird ein zu übertragender digitaler Datenstrom zuerst einer Fehlerkorrektur-Codierung unter Verwendung von beispielsweise einer Faltungscodierung unterzogen. Die codierten Datenbits werden dann gemäß einem Verschachtelungsalgorithmus bezüglich der Zeit verteilt, der auch eine beschränkte Anzahl von Symbolen einfügen kann, die dem Empfänger im Voraus bekannt sind, wie beispielsweise Synchronisationssymbole. Die codierten und verschachtelten Datenbits werden dann auf ein Funksignal moduliert, das in Bursts übertragen werden kann.

[0014] Der erfinderische Demodulator weist eine Viterbi-Abschätzeinheit für eine Sequenz maximaler Länge auf, die Datenbitsequenzen einer Hyothesenbildung unterzieht. Mit der Hilfe eines Modells des Codier- und Modulationsprozesses (Codulationsprozesses) werden Funksignal-Abtastungen ("Symbole"), die beobachtet werden sollten, vorhergesagt. Diese werden mit der tatsächlich empfangenen komplexen Signalwellenform verglichen, und eine Fehlanpassung wird berechnet. Die Viterbi-MLSE-Maschine fährt dann damit fort, die Datenbithypothese zu bestimmen, die die geringste Fehlanpassung ergibt, und gibt diese Hypothese aus. Der erfinderische Decodulationsprozess gemäß einem beispielhaften Ausführungsbeispiel berechnet zuerst Fehlanpassungen für Symbole, die an bekannte Symbole angrenzend liegen, und sammelt die Fehlanpassungen in eine gesamte Fehlanpassung oder ein Pfadmaß bzw. Wegemaß für die Hypothese-Symbolsequenz. Dieser Prozess wird hierin "Decodulieren eines Symbols" genannt. Die bekannten Symbole können beispielsweise Synchronisationssymbole sein.

[0015] Wenn alle Positionen, die bekannten Symbolen am nächsten sind, ausgeschöpft sind, können fehlerkorrigierte Daten beginnen, aus dem Decodierer aufzutauchen, und somit kann das erste decodierte Symbol nun mit relativ hoher Sicherheit als ein bekanntes Symbol angesehen werden. Der Prozess fährt dann unter Behandlung dieses Symbols als bekanntes Symbol fort, um mit ihm die Decodulation eines benachbarten Symbols zu unterstützen. Dies räumt ein anderes fehlerkorrigiertes Datensymbol aus, usw., bis alle Daten decodiert worden sind. Dieser Prozess basiert teilweise auf einer Kenntnis des Empfängers bezüglich der Reihenfolge einer Platzierung von Symbolen im übertragenen Strom und auf der Platzierung von irgendwelchen bekannten Synchronisationssymbolen, d. h. bezüglich des Verschachtelungsmusters. Eine Auswahl von Sende-Verschachtelungsformaten zum Erleichtern eines Decodulierens ist hierin detailliert beschrieben.

[0016] Ein weiterer der Vorteile von Techniken gemäß der vorliegenden Erfindung besteht darin, zuzulassen, dass Daten differentiell moduliert werden, d. h. eher bei den Übergängen zwischen zwei Signalwellenformstel-

len codiert zu werden, als bei dem Absolutwert der Wellenform, ohne dass der gewöhnliche differentielle Verlust von 3 dB auftritt. Eher eine differentielle als eine absolute (kohärente) Modulation ist für Funkkanäle wünschenswert, die über dem Raum von einigen wenigen Symbolen schnell variieren bzw. sich ändern, was eine kohärente Demodulation behindern würde. Die vorliegende Erfindung liefert den Vorteil für zeitvariierende Kanäle einer differentiellen Modulation ohne den normalen Leistungsnachteil von 3 dB. Ein diagonal verschachteltes, codiertes und differentiell moduliertes Signalformat ist offenbart, das wenigstens die Hälfte bis zu dem gesamten des differentiellen Modulationsverlustes von 3 dB wiedergewinnen kann.

KURZE BESCHREIBUNG DER ZEICHNUNGEN

[0017] Die vorangehenden und andere Aufgaben, Merkmale und Vorteile der vorliegenden Erfindung werden einfacher beim Lesen der folgenden detaillierten Beschreibung in Zusammenhang mit den Zeichnungen verstanden werden, wobei:
[0018] **Fig.** 1 ein Signalübertragungsformat gemäß einem beispielhaften Ausführungsbeispiel der vorliegenden Erfindung darstellt;
[0019] **Fig.** 2(a) ein Sender-Blockdiagramm zum Erzeugen eines Signals darstellt, wie beispielsweise dasjenige der **Fig.** 1;
[0020] **Fig.** 2(b) einen beispielhaften Codierer mit der Rate 1/3 darstellt, der bei dem Sender der **Fig.** 2(a) verwendet werden kann;
[0021] **Fig.** 3 ein Empfänger-Blockdiagramm zum Decodieren eines empfangenen Signals, wie beispielsweise desjenigen, das in **Fig.** 1 gezeigt ist, darstellt;
[0022] **Fig.** 4 die innere Struktur des in **Fig.** 3 implementierten beispielhaften Decodulationsprozesses detaillierter darstellt;
[0023] **Fig.** 5 eine modifizierte interne Struktur zum Berechnen von Referenzvektoren im Vorübergehen bzw. im Fluge darstellt;
[0024] **Fig.** 6 ein blockdiagonal verschachteltes und differentiell moduliertes binäres Übertragungsformat darstellt;
[0025] **Fig.** 7(a) und 7(b) ein Ablaufdiagramm zur Verfügung stellen, das ein beispielhaftes Mehrwegeverfahren gemäß der vorliegenden Erfindung darstellt; und
[0026] **Fig.** 8 eine Modifikation einer differentiellen Modulation zum Überspringen nicht codierter Symbole darstellt.

DETAILLIERTE BESCHREIBUNG

[0027] Gemäß **Fig.** 1 ist eine Anzahl von codierten Symbolen, die zu einem Empfänger zu befördern sind, mit "U" für unbekannt markiert, und mit einer Anzahl von bekannten Symbolen, die mit "K" markiert sind, verschachtelt. Die Symbole sind zwischen eine Anzahl von Bursts in einem angenommenen Zeitvielfachzugriffs-(TDMA = Time Division Multiple Access)-Übertragungssystem verteilt gezeigt, aber dies ist nur ein Beispiel, und die Erfindung kann gleichermaßen auf kontinuierliche (z. B. FDMA oder CDMA) sowie auf Burst-Übertragungen angewendet werden.
[0028] Als Beispiel enthält ein TDMA-Frame von 40 mS 300 Symbolperioden und ist in drei Zeitschlitze aufgeteilt. Ein Funksender ist einem Zeitschlitz pro Frame zugeteilt, in welchem er nahezu 100 Symbole zu senden hat. Einige der Symbolperioden zu Beginn und am Ende von Übertragungsbursts werden für eine Leistungs-Aufwärts/Abwärts-Rampenbildung reserviert und werden somit nicht zur Datenübertragung verwendet. Die in **Fig.** 1 mit 1 bis 4 nummerierten Bursts beziehen sich auf entsprechende Zeitschlitze, die demselben Sender in aufeinanderfolgenden TDMA-Frames von 40 mS zugeteilt sind.
[0029] Alternativ dazu kann ein Hybrid-TDMA/FDMA-Zugriffsformat, wie es im US-Patent US-A-5,539,730 mit dem Titel "TDMA/FDMA/CDMA Hybrid Radio Access Methods), eingereicht am 11. Januar 1994, eschrieben ist, verwendet werden. Diese Alternative enthält, dass jeder Sender für zwei aus den drei Zeitschlitzen sendet, so dass sich in jedem Zeitschlitz zwei Übertragungen überlagern. Jedoch verwenden die zwei gleichzeitig sendenden Vorrichtungen jeweils die oberen und unteren Hälften des zugeteilten Frequenzkanalspektrums, wenn sie sich bezüglich der Zeit überlagern, um eine wechselseitige Interferenz zu vermeiden. Wenn dieser Typ von Zugriffsformat verwendet wird, wird sich die Bezeichnung (Burst 1, Burst 2), die in **Fig.** 1 verwendet wird, jeweils auf eine Senderübertragung in einem ersten von drei Zeitschlitzen unter Verwendung einer unteren Hälfte des Kanalspektrums und eine Übertragung in einem zweiten Zeitschlitz unter Verwendung einer oberen Hälfte des Kanalspektrums beziehen. Der Burst 3, der Burst 4 würden sich dann auf dasselbe Auftreten in dem folgenden TDMA-Frame von 40 mS beziehen.
[0030] Ein erwünschtes Ziel ist beispielsweise ein Übertragungsformat mit einer Effizienz von 90%, das 90 aus den 100 Symbolperioden pro Burst einer Datenübertragung zuteilt. **Fig.** 1 stellt vier Symbole, die mit K markiert sind, in jedem Burst dar, die Synchronisationssymbolen gewidmet sind, und die übrigen sechs Sym-

bolperioden pro Burst werden für eine Aufwärts- und Abwärts-Rampenbildung einer Dauer von jeweils drei Symbolen verwendet. Die bekannten Symbole werden keiner Gruppenbildung bzw. Clusterbildung unterzogen, da es keinen Vorteil für Kanäle gibt, dies zu tun, die sich über dem Raum einiger weniger Symbole ändern können. Stattdessen werden unbekannte Datensymbole abgewägt zwischen den bekannten Symbolen platziert, und zwar aus den Gründen, die unten erklärt sind.

[0031] Eine differentielle Modulation wird so verwendet, dass Symbole dem Träger durch Ändern ihrer Phase über einer Symbolperiode eingeprägt werden. Beispielsweise kann das als Pi/4-DQPSK bekannte Modulationsschema verwendet werden, bei welchem die Phasenänderung einer der vier Werte +/–45 oder +/–135° ist, um dadurch zwei Informationsbits zu codieren.

[0032] Wenn das Symbol und daher die Phasenänderung bekannt ist, gibt es eine bekannte Beziehung zwischen dem startenden und dem endenden Signalvektor. Durch Drehen des bei dem Start eines bekannten Symbols empfangenen Vektors über dieses bekannte Ausmaß, kann er mit dem Endvektorwert ausgerichtet werden, der empfangen wird, und konstruktiv zu ihm addiert werden, um dadurch das Signal-zu-Rausch-Verhältnis beim Ergebnis um 3 dB zu verbessern. Dieser verbesserte Endvektor ist der Startvektor für das folgende, unbekannte Symbol, und die Verbesserung des Signal-zu-Rausch-Verhältnisses lässt somit zu, dass das unbekannte Symbol durch einen Vergleich mit einem Hypothesesymbol zuverlässiger bestimmt wird.

[0033] Wenn das Symbol, das dem unbekannten Symbol folgt, auch ein bekanntes Symbol ist, kann derselben Prozedur umgekehrt gefolgt werden, um die Phase zu Beginn des zweiten Synchronisationssymbols/des Endes des unbekannten Symbols zu bilden. Da sowohl der Start- als auch der Endvektor des unbekannten Symbols dann ein um 3 dB höheres Signal-zu-Rausch-Verhältnis haben, setzt sich dies direkt zu einer Verbesserung von 3 dB bei einer Demodulation des unbekannten Symbols um. Mathematisch wird dies durch das folgende beschrieben:

Man lasse Z(i – 1) den komplexen Wert eines Signals sein, der zu Beginn eines ersten bekannten Symbols abgetastet wird;

man lasse Z(i) die Abtastung am Ende des ersten bekannten Symbols und zu Beginn des unbekannten Datensymbols sein;

man lasse die bekannte differentielle Phasendrehung, die durch das erste bekannte Symbol angewendet wird, Thetal sein; dann gilt

U(i) = (Z(i – 1)Exp(j·Theta1) + Z(i))/2 ist eine bessere Abschätzung des Signalvektors beim Start des unbekannten Symbols.

[0034] Man lasse Z(i + 1) den Signalvektor am Ende des unbekannten Symbols und beim Start des zweiten bekannten Symbols sein und man lasse Z(i + 2) den Signalvektor am Ende des zweiten bekannten Symbols sein;

man lasse Theta2 die bekannte differentielle Phasendrehung sein, die durch das zweite bekannte Symbol angewendet wird; dann gilt:

U(i + 1) = (Z(i + 1) + Z(i + 2) Exp(–j·Theta2))/2 ist eine bessere Abschätzung des Signalvektors am Ende des unbekannten Symbols.

[0035] Daher ist eine verbesserte Abschätzung des unbekannten Symbols:

$$U(i+1)\cdot U^* \qquad \text{(i)}$$

wobei * komplex-konjugiert bedeutet und wobei die Real- und Imaginärteile jeweils weiche Werte für die 2 Bits des quaternären Symbols ergeben. Diese weichen Werte werden mit einer geeigneten Skalierung bei einem Faltungsdecodierprozess verwendet, der in **Fig.** 4 dargestellt ist. Vor einem Beschreiben von **Fig.** 4 werden Blockdiagramme eines beispielhaften Senders und eines beispielhaften Empfängers mit der Hilfe der **Fig.** 2(a), 2(b) bzw. 3 beschrieben werden.

[0036] Gemäß **Fig.** 2(a) wird ein zu übertragendes analoges Sprachsignal durch einen Sprachcodierer **20** mit niedriger Bitrate in einen digitalen Bitstrom umgewandelt. Dieser Codierer kann beispielsweise einen A/D-Wandler für 8 Kiloabtastungen/zweite PCM, gefolgt durch einen resterregten, linearen Vorhersagecodierer (RELP = Residual Excited, Linear Predictive Coder), einen adaptiven Delta-PCM (ADPCM), eine Vektor-Codebuch-erregte lineare Vorhersageschaltung (VSELP = Vector Code-Book Excited Linear Prediction) oder eine ähnliche Schaltung sein. Der Codierer **20** wandelt Sprache zu einer niedrigen Bitrate um, während eine subjektive Qualität beibehalten wird, einschließlich beispielsweise einer getreuen Wiedergabe von Hintergrundklängen.

[0037] Die durch den Codierer **20** erzeugten Bits haben nicht typischerweise alle dieselbe Signifikanz beim Beeinträchtigen der wahrgenommenen Sprachqualität. Bits können somit gemäß ihrer Wahrnehmungssignifikanz sortiert werden, so dass ein Fehlerkorrektur-Codierer **21** seine Anstrengungen am effektivsten ausrichten kann.

[0038] Der Fehlerkorrektur-Codierer **21** kann eine größere Anzahl von Ausgangsbits als Eingangsbits erzeugen, um eine Redundanz zur Übertragung zur Verfügung zu stellen. Ebenso sind Ausgangsbits eine Funktion

von mehreren Eingangsbits, und umgekehrt beeinflusst jedes Eingangsbit mehrere Ausgangsbits, so dass ein Verlust von einem Ausgangsbit bei einer Übertragung nicht notwendigerweise bedeutet, dass ein Informationsbit verloren wird. Ein beispielhafter Codierer zur Verwendung als Codierer **21** ist ein Codierer mit durchbrochener Faltung. Ein Faltungscodierer weist ein Schieberegister einer Länge von L Bits auf, durch welches die Eingangsdatenbits geführt werden. Eine erste Gruppe von Abgriffen an dem Register wählt eine Anzahl von verzögerten Bits aus und kombiniert sie durch eine Exklusiv-ODER-Operation, um ein erstes codiertes Bit oder ein Paritätsbit P1 zu erzeugen. Eine zweite Gruppe von Abgriffen wählt eine andere Kombination von verzögerten Bits aus, um P2 zu erzeugen, usw. Wenn die Gesamtanzahl von Paritätsbits, die für jedes Eingangsdatenbit erzeugt sind, N ist, dann wird der Codierer ein Codierer mit der Rate 1/N genannt. Beispielsweise dann, wenn zwei Paritätsbits für jedes Eingangsbit erzeugt werden, das hereingeschoben wird, ist der Codierer ein Codierer mit der "Rate einer Hälfte". Ein gegebenes Eingangsbit kann mehr als ein P1 oder P2 beeinflussen und beeinflusst somit mehrere übertragene Bits. Die Anzahl von Bits, die beeinflusst werden, kann wie folgt berechnet werden: wenn P1 eine Kombination von N1 Registerabgriffen ist, dann werden N1 P1-Werte beeinflusst werden, wenn ein gegebenes Datenbit die verwendeten Abgriffe durchläuft. Wenn P2 unter Verwendung von N2 Abgriffen erzeugt wird, dann werden N2 P2-Werte durch ein gegebenes Datenbit beeinflusst werden. Insgesamt hängen daher N1 + N2 Ausgangswerte von einem jeweiligen Eingangsbit ab.

[0039] Ein Code mit einer durchbrochenen Faltung kann unterschiedliche Schutzpegel für codierte Sprachbits von unterschiedlicher Wahrnehmungssignifikanz ergeben. Es kann angenommen werden, dass das Ausmaß an Schutz proportional zu der Anzahl von gesendeten Codierer-Ausgangsbits ist, die von diesem Sprachbit abhängen. Dies kann durch Weglassen variiert werden, bestimmte der Paritätsbits P1, P2, P3, ... gemäß einer Lösch- oder "Durchbrechungs"-Strategie zu übertragen. Beispielsweise dann, wenn nur P1 nach einer bestimmten Registerverschiebung übertragen wird, wird die Anzahl von Ausgangsbits, die von den durch die Registerabgriffe ausgewählten N1 Eingangsbits abhängen, um 1 reduziert. N1 Eingangsbits erhalten somit einen etwas geringeren Schutz. Diese können derart angeordnet werden, dass sie Bits einer etwas niedrigeren Wahrnehmungssignifikanz sind. Andererseits kann es dann, wenn ein drittes Paritätsbit P3, das von N3 Registerabgriffen abhängt, berechnet wird, aber nicht normal übertragen wird, zusätzlich übertragen werden, wann immer es erwünscht ist, N3 der Eingangssprachbits einen größeren Schutz zuzuteilen. Somit kann durch Weglassen, bestimmte Paritäten zu einigen Zeitpunkten zu übertragen, und durch Übertragen von zusätzlichen Paritäten zu anderen, ein kontinuierlich abgestuftes Ausmaß an Fehlerschutz für unterschiedliche Eingangsbits zur Verfügung gestellt werden, während noch dieselbe Gesamtanzahl von übertragenen Bits gesendet wird. Die Schutzstufen können gemäß dem Schutz sortiert werden, der zur Verfügung gestellt ist (d. h. gemäß der Anzahl von übertragenen Bits, die von einem jeweiligen Eingangsbit abhängen), und dann auf die Sprachbits abgebildet werden, die in einer Reihenfolge einer Wahrnehmungssignifikanz sortiert sind. Eine detailliertere Erklärung einer Codierung mit punktierter Faltung gemäß einem beispielhaften Ausführungsbeispiel der vorliegenden Erfindung ist nachfolgend beschrieben.

[0040] Wenn Sprache durch fortentwickelte Techniken mit niedriger Bitrate digitalisiert wird, haben nicht alle Bits die gleiche Wahrnehmungssignifikanz. Daher kann beim Übertragen der digitalen Sprache über einen Schwund- bzw. Fading-Funkkanal ein Anpassen des Ausmaßes an Fehlerkorrektur-Codierschutz an jedes Bit gemäß seiner Wahrnehmungssignifikanz die optimale Ausnutzung von Bandbreite sein.

[0041] Beispielsweise soll angenommen sein, dass Codes mit der Rate 1/3, 1/2 und 2/3 rds verfügbar sind, sowie keine Codierung (Rate 1), und dass 163 Bits zu 288 Bits für eine Übertragung alle 40 mS codiert werden müssen.

[0042] Ein Bezeichnen der Bits, die jeweils durch die Rate 1/3, 1/2 und 2/3 rds und 1 codiert sind, durch a, b, c und d liefert das folgende System von Gleichungen:

$$a + b + c + d = 163$$

und

$$3a + 2b + 1{,}5c + d = 288$$

daher gilt:

$$1a + b + 0{,}5c = 125$$

[0043] Unter der Annahme, dass nur die signifikantesten 11 Bits den extremen Schutz der Rate 1/3 garantieren, d. h. a = 11; gilt dann folgendes:

$$b + 0{,}5\,c = 103$$

oder

2b + c = 206

[0044] Somit ist die absolute maximale Anzahl b der mit der Rate 1/2 codierten Bits 103, wenn c = 0. Somit existieren die folgenden Möglichkeiten:

a	b	c	d
11	103	2	49
11	102	2	48
11	101	4	47
11	100	6	46
.		.	.
11	54	98	0

[0045] Es existieren andere Möglichkeiten mit unterschiedlichen Werten von "a". Der maximale mögliche Wert von "a" ist 62, was folgendes ergibt:

a	b	c	d
62	0	2	99

[0046] Wenn einige der 99 nicht codierten Bits zu wahrnehmungssignifikant sind, um keine Codierung zu tolerieren (d. h. 8–10% BER), dann kann "a" beispielsweise durch Transferieren von "a" Bits zu "b" Bits reduziert werden:

	61	1	4	97	
	60	2	6	95	
	59	3	8	93	
	.	.	.	.	
	31	31	64	37	
	.	.	.	.	
oder	36	36	34	57	
	35	36	38	54	
	34	36	42	51	
	33	36	46	48	.....(1)

usw.

[0047] Die letztere Möglichkeit stellt vielleicht die gleichmäßigste Verteilung von Bits zwischen den vier möglichen Schutzpegeln dar.

[0048] Viele unterschiedliche Arten zum Anwenden der obigen ungleichen Schutzebenen können erdacht werden. Beispielsweise dann, wenn eine Faltungscodierung verwendet wird, ist es äußerst wünschenswert, den Zusatz von Endbits zu vermeiden. Endbits können entweder durch Verwenden eines kontinuierlichen Faltungs-Codierens/Decodierens oder durch ein Beschneiden des Endes bzw. ein Abbeißen des Endes vermieden werden.

[0049] Das in der Zeile (1) einer Bitzuteilung gezeigte letztere Beispiel wird zum Beschreiben eines kontinuierlichen Faltungscodierens verwendet werden. Dabei sind 33 Bits pro 40 mS mit einer Rate von 1/3 rd zu schützen. Die 33 Bits von jedem Sprachframe mit 40 mS werden aneinandergekettet, um einen kontinuierlichen Eingangsbitstrom zu einem Faltungscodierer mit optimierter Rate 1/3 rd zu bilden. Die 99 Bits pro 40 mS, die aus dem Codierer ausgegeben werden, werden sowohl innerhalb von TDMA-Bursts als auch zwischen Bursts zur Übertragung verschachtelt. Die Verschachtelung sollte nur durch die Sprachverzögerung beschränkt sein, die toleriert werden kann.

[0050] Gleichermaßen werden die 36 Bits von jedem Frame für einen Schutz der Rate 1/2 verkettet, um eine kontinuierliche Bitstromeingabe zu einem Codierer mit einer optimierten Rate von 1/2 zu bilden. Die 72 Bits pro 40 mS, die ausgegeben werden, sind sowohl innerhalb als auch zwischen Frames mit den anderen Codie-

rer-Ausgangsbits verschachtelt.

[0051] Die 46 mit einer Rate von 2/3rds geschützten Bits werden mit gleichen bzw. ähnlichen Blöcken verkettet, um die Bitstromeingabe zu einem Codierer mit einer Rate von 2/3rds zu bilden, der eine Ausgabe von 69 Bits pro 40 mS erzeugt. Diese werden auch mit den vorherigen Codiererausgaben zusammen mit der nicht codierten Information von 48 Bits/40 mS verschachtelt, was insgesamt übertragene 88 Bits pro 40 mS ergibt.

[0052] Ein potentieller Nachteil dieses Verfahrens ohne CRCs besteht darin, dass der Decodierer mit einer Rate von 1/3rd beispielsweise nicht sicherstellen kann, dass er einen 33-Bit-Block richtig decodiert hat, bis er einige Bits vom nächsten Frame empfangen hat, um den Decodierer auszuräumen. Somit kann eine Zusatzverzögerung beim Empfänger eingeführt werden. Dies kann entweder durch bloßes Verwenden der Bits entsprechend dem besten Maß vermieden werden, das nicht optimal sein kann, oder durch Hinzufügen eines CRC, der auf jeden Viterbi-Zustand geprüft wird, um ein festeres Indiz für den richtigen Zustand zu ergeben, was unglücklicherweise einen Zusatz hinzufügt und das Ausmaß an möglichem Codierschutz reduziert, oder durch Verwenden von einem "Abschneiden eines Endes", wie es als nächstes beschrieben wird.

[0053] Gemäß einem weiteren beispielhaften Ausführungsbeispiel der vorliegenden Erfindung werden unter Verwendung eines Faltungscodierens mit einem Abschneiden eines Endes die 33 Bits, die mit einer Rate von 1/3rd zu codieren sind, derart angesehen, dass sie in einem geschlossenen Kreis angeordnet sind. Ein Faltungscodierer mit der Beschränkungslänge 6 würde beispielsweise mit den Bits 0 bis 5 beginnen bzw. starten und aus ihnen drei codierte Bits erzeugen. Der Prozess würde sich dann für eine Stufe um den Kreis bewegen, um drei weitere Bits unter Verwendung der Bits 1 bis 6 zu erzeugen, usw., bis 99 Ausgangsbits erzeugt worden sind, wobei die letzten drei vom Bit 31 und wieder von den Bits 0 bis 4 abhängen. Die 99 Bits werden dann mit der Ausgabe der anderen Codierer für ein Abschneiden eines Endes und den nicht codierten Bits verschachtelt, um einen Frame mit einer Gesamtheit von 288 pro 40 mS zu ergeben. Der Decodierer arbeitet an den empfangenen 99 codierten Bits auf gleiche Weise, die in einem Kreis angeordnet sind, und fährt damit fort, bis alle Bits decodiert worden sind, und etwas darüber hinaus, bis ein zweites Mal decodierte Bits mit bei der ersten Umdrehung decodierten Bits übereinstimmen. Somit kann unter Verwendung eines Abschneidens eines Endes ein Decodieren beendet werden, ohne darauf zu warten, den nächsten Frame zu empfangen.

[0054] In Wirklichkeit sind Codierer für ein Abschneiden eines Endes eine Form einer Blockcodierung, welche jedoch im Sinne einer Hamming-Distanz nicht optimal sein kann. Die 33 Bit-Blöcke sollten natürlich Idealerweise auf 99 Bits codiert werden, und zwar unter Verwendung eines optimalen Blockcodes (99,33). Decodierer für ein Abschneiden eines Endes und Faltungsdecodierer sind jedoch viel einfacher zu implementieren, als große Blockcodierer und -decodierer.

[0055] Während ein beispielhaftes Verfahren zum Implementieren einer Mehrfachratencodierung unter Verwendung eines durchbrochenen Faltungscodierens und -decodierens beschrieben worden ist, wird nun gezeigt werden, dass durch ein intelligentes Verschachteln der unterschiedlichen Raten selbst die Fehlerrate der "nicht codierten" Bits wesentlich reduziert werden kann. Ein Faltungscodierer besteht konzeptmäßig aus einem abgegriffenen Schieberegister, durch welches zu codierende Bits verschoben werden. Bei jeder Verschiebung wird eine erste Gruppe von Abgriffen einer Exklusiv-ODER-Verknüpfung unterzogen, um ein erstes codiertes Bit zu erzeugen, wird eine zweite Gruppe von Abgriffen einer Esklusiv-ODER-Verknüpfung unterzogen, um ein zweites codiertes Bit zu erzeugen, etc., bis zu dem erwünschten Ausmaß an Redundanz (d. h. dem reziproken der Rate). Somit ist ein Codierer mit einer Beschränkungslänge von 6 und einer Rate von 1/3rd in **Fig.** 2(b) dargestellt.

[0056] Es kann bestimmt werden, dass zu dem in **Fig.** 2(b) gezeigten Zeitpunkt die Ausgangsbits gegeben sind durch:

```
P11 = B1 + B2 + B3       + B5 + B6
P12 = B1           + B4 + B5
P13 = B1 + B2      + B4      + B6
```

[0057] Bei darauffolgenden Zeitpunkten werden die Ausgangsbits gegeben sein durch:

```
P21  =  B2  +  B3  +  B4                    +  B6  +  B7
P22  =  B2                        +  B5  +  B6
P23  =  B2  +  B3                 +  B5        +  B7

P31  =  B3  +  B4  +  B5                    +  B7  +  B8
P32  =  B3                        +  B6  +  B7
P33  =  B3  +  B4                 +  B6        +  B8

etc.
```

[0058] Die übertragenen Bits Pij bei sequentiellen Registerverschiebungen i = 1, 2, 3, etc. können in drei Zeilen j = 1, 2, 3 wie folgt angeordnet werden:

```
P11  P21  P31  P41  P51  P61  ...
P12  P22  P32  P42  P52  P62  ...
P13  P23  P33  P43  P53  P63  ...
```

[0059] Welche von diesen durch B7 beeinflusst werden, wenn es sich beispielsweise durch das Codierungs-Schieberegister bewegt, kann durch ein Kreuz in dem geeigneten Kasten dieser 3-mal-5-Matrix angezeigt werden:

```
x    x         x    x    x
     x    x              x
x         x         x    x
```

[0060] Somit beeinflusst irgendein Bit, wie beispielsweise B6 eine Anzahl von übertragenen Bits gleich der Anzahl der obigen Kreuze, d. h. 12. Ein Fehler beim Decodieren von B6 würde daher eine Fehlanpassung zwischen 12 der empfangenen Bits und ihre entsprechenden erwarteten Werte verursachen, und es ist somit weniger wahrscheinlich, dass ein Fehler gemacht werden wird.

[0061] Die Reduzierung bezüglich einer Fehlerrate wird selbst dann noch deutlich groß sein, wenn nicht alle der 12 Pij's, die durch B6 bewirkt werden, übertragen werden. Beispielsweise dann, wenn P11, P22 und P33 nicht übertragen werden, von welchen alle von B6 abhängen, wird die Anzahl von übertragenen Bits, die von B6 abhängen, 9 anstelle von 12 sein. Zur gleichen Zeit reduziert sich die Anzahl von übertragenen Bits, die von B1 abhängen, auf 11, die von B2 abhängen, auf 10, die von B3 abhängen, auf 10, die von B4 abhängen, auf 11 und die von B5 abhängen, auf 10.

[0062] Allgemein kann es durch Abzielen auf ein Löschen von übertragenen Bits, die von B6 abhängen, eingerichtet werden, dass das Ausmaß an Codierung, welches B6 zugeteilt wird, reduziert wird, während ein geringerer Verlust eines Codierschutzes für benachbarte Bits veranlasst wird. Der einem jeweiligen Datenbit zugeteilte Schutz, das durch einen solchen punktuellen Code codiert ist, kann letztlich auf die Anzahl von übertragenen Bits bezogen sein, die von ihm abhängen, und somit kann dieses Datenbit für ein Sprachbit von entsprechender Wahrnehmungssignifikanz verwendet werden.

[0063] Wenn ein regelmäßiges Muster von Löschungen verwendet wird, wie beispielsweise durch Löschen von einem Pij aus den dreien in jeder Spalte, wird ein Code mit der Rate 1/2 erzeugt. Das gelöschte P-Bit sollte nicht notwendigerweise immer in derselben Zeile sein, sondern könnte gemäß einem regelmäßigen Durchbrechungsmuster zugelassen sein.

[0064] Gleichermaßen kann ein Code mit einer Rate von 2/3 durch Löschen von zwei P-Bits aus einer Hälfte der Spalten und von einem P-Bit aus der anderen Hälfte erzeugt werden, so dass drei Bits für alle zwei Codier-Registerverschiebungen übertragen werden. Eine Rate 1 kann durch Löschen von allem außer einem P-Bit in jeder Spalte erzeugt werden. Jedoch ist es interessant, anzumerken, dass die Anzahl von übertragenen Bits, die von einem gegebenen Datenbit abhängen, sich nicht auf Eins reduzieren, wie es für überhaupt keine Codierung der Fall wäre.

[0065] Wenn beispielsweise die übertragenen Bits folgende sind:
P11, P22, P23, P41, P53, P62; P71, P82, P93, etc.,
dann beeinflusst B6 noch sechs übertragene Bits. Wenn diese mit unterschiedlichen TDMA-Frames oder Frequenzsprüngen eines Funkübertragungsformats verschachtelt sind, oder mit unterschiedlichen Positionen innerhalb desselben Frames oder Sprungs, wo ein Fading im Wesentlichen unkorreliert ist, dann wird eine Maß-

nahme eines Schutzes von B6 gegenüber einem Fading erhalten werden, ohne die Anzahl von übertragenen Bits erhöht zu haben. Eine Viterbi-Abschätzeinheit für eine Sequenz mit maximaler Wahrscheinlichkeit kann noch verwendet werden, um einen solchen Code mit einer "Rate 1" zu decodieren.

[0066] Der potentielle Vorteil von Codes mit einer Rate 1 von diesem Typ mit einer langen Beschränkungslänge besteht im Ergeben derselben Bitfehlerratenkurve für einen Relais-Fading-Kanal wie für einen statischen Kanal, da die Summe der Relais-Schwundrauschbeiträge zu dem Maß eines gegebenen decodierten Bits durch das zentrale Grenztheorem sich einfach einem Gauß-Rauschen mit derselben mittleren Varianz nähern muss. Ein solches Codieren, das eine Verstärkung ohne eine Bandbreitenexpansion ergeben kann, ist ein neuer Aspekt der vorliegenden Erfindung. Zum effizienten Arbeiten sollten solche mit einer "Rate 1" codierten Bits durch schwerer bzw. stärker codierte Bits begrenzt werden, und allgemein wird ein kontinuierlicher Bereich eines Codierschutzes durch die Erfindung erzeugt, indem Bits mit unterschiedlichen Ausmaßen an Codierung verschachtelt werden. Dies ist gegensätzlich zu der Verwendung nach dem Stand der Technik von durchbrochenen Faltungscodes, die typischerweise alle mit einer Rate von 1/3 codierten Bits miteinander gruppieren, alle mit der Rate 1/2 codierten Bits miteinander, etc., und die "nicht codierten Bits" überhaupt nicht codieren.

[0067] Gemäß diesem beispielhaften Ausführungsbeispiel wird jedoch ein Durchbrechungsmuster ausgewählt, das zwischen einem Übertragen von allen Paritäten für die am stärkten bzw. schwersten codierten Bits und einem Weglassen von einer oder mehreren Paritäten für die weniger codierten Bits abwechselt. Beispielsweise kann die Anzahl von übertragenen Paritäten für jede Verschiebung in das Codierregister eines zu codierenden neuen Bits gemäß der Sequenz 3,3,2,3,2,1,2,3,2,3,1,1,2,1,2,2,3, ... usw. gewählt werden. Bei Vorgabe eines solchen Durchbrechungsmusters kann die Anzahl von übertragenen Paritäten, die von einem jeweiligen der Ursprungsinformationsbits abhängen, berechnet werden.

[0068] Beispielsweise kann das

Informationsbit 1 12 übertragene Paritäten beeinflussen;
Bit 2 10 übertragene Paritäten beeinflussen;
Bit 3 7 übertragene Paritäten beeinflussen;
Bit 4 11 übertragene Paritäten beeinflussen;
Bit 5 9 übertragene Paritäten beeinflussen; usw.

[0069] Weiterhin werden gemäß diesem beispielhaften Ausführungsbeispiel diese Bits demgemäß sortiert, wie viele übertragene Paritäten von ihnen abhängen, wodurch beispielsweise die folgende Tabelle erhalten wird: Bits, die 12 11 10 9 8 7 6 5 4 3 2 1 Paritäten beeinflussen:

```
b1  b4  b2  b5      b3
b6  b9      b7  b10 b8
b11 b14 b13 b15 b12 b16
b21 b12 b17 b23 b20 b25
b24 b18 b22 b28 b27     etc.
```

[0070] In der obigen Tabelle ist angenommen, dass nur sechs Pegel bzw. Ebenen einer Redundanz erzeugt werden, d. h. Informationsbits nicht weniger als 7 und nicht mehr als 12 Paritäten beeinflussen. Dies soll jedoch nicht bedeuten, dass eine Beschränkung impliziert ist, sondern nur dass in der Praxis ein solcher Bereich von Optionen ausreichen kann.

[0071] Fährt man gemäß diesem beispielhaften Ausführungsbeispiel fort, werden die Informationsbits, wie sie durch den Sprachdigitalisierungsprozess erzeugt werden, gemäß ihrer Wahrnehmungssignifikanz sortiert, d. h. demgemäß, wie störend es für den Hörer klingt, wenn ein bestimmtes Bit fehlerhaft ist. Dies kann im Voraus während des Entwicklungsprozesses bestimmt werden, und zwar vorzugsweise durch subjektive Hörtests unter Verwendung einer Vielzahl von Sprechern und Hörern. Unter der Annahme, dass ein solcher Hörtest das folgende Ergebnis erzeugt, wenn Sprachcodiererbits, die mit c1, c2, c3, etc. bezeichnet sind, durch eine Wahrnehmungssignifikanz von höchster Signifikanz zur niedrigsten sortiert werden:

c1, c8, c10, c2, c7, c11, c12, c4 c5, c3, c6, c23, c34, c15, c9, ...

[0072] Das beispielhafte Codiersystem ordnet nun die erste Gruppe von c-Bits (d. h. die am meisten wahrnehmungssignifikante) zu den b-Bits zu, die die größte Anzahl von Paritäten (z. B. 12) beeinflussen, so dass c1, c8, c10, c2 und c7 bei dem obigen Beispiel durch b1, b6, b11, b21 und b24 dargestellt werden würden.

[0073] Dann werden, wenn der Bitzuteilungsprozess die Anzahl von Bits ausgeschöpft hat, die 12 Paritäten beeinflussen, die Sprachbits, die in einer Signifikanzreihenfolge die nächsten sind, denjenigen zugeordnet, die 11 Paritäten beeinflussen, so dass die Sprachbits c11, c12, c4, c5 und c3 durch den Fehlerkorrektur-Codierprozess durch b4, b9, b14, b18 und b19 dargestellt werden würden. Der Prozess fährt damit fort, Sprachbits von nach und nach niedrigerer Signifikanz zu b-Bits zuzuordnen, die nach und nach weniger übertragene Paritäten beeinflussen.

[0074] Das obige beispielhafte Ausführungsbeispiel für ein durchbrochenes Codieren konzentriert somit die Leistung des Codierens darauf, wo es am meisten benötigt wird, aber mit einem Abbilden von Sprachbits zu codierten Bits, das einen kontinuierlicheren Bereich eines Codierschutzes als Systeme nach dem Stand der Technik erkennt, die Bits einfach in diejenigen gruppieren, die geschützt werden sollen, und zwar mit einer Codierung mit einer Rate von 1/3, diejenigen, die geschützt werden sollen, mit einer Codierung von einer Rate von 1/2, etc., und diejenigen, die überhaupt nicht codiert werden sollen. Insbesondere kann es gesehen werden, dass gemäß diesem beispielhaften Ausführungsbeispiel keinerlei Sprachbits nötig sind, die nur ein übertragenes Bit beeinflussen.

[0075] Ein strengeres Entwicklungsverfahren zum Implementieren dieses beispielhaften Ausführungsbeispiels enthält am effizientesten ein Simulieren der Übertragung der codierten Bits, einschließlich eines Verschachtelns, einer Modulation, von Übertragungskanalfehlern, eines Rauschens und einer Interferenz. Viele Frames einer Übertragung werden simuliert, und die mittlere Fehlerwahrscheinlichkeit für jedes codierte Informationsbit b1, b2, b3, etc. berechnet. Diese Bits werden dann durch eine mittlere Fehlerrate vom niedrigsten zum höchsten sortiert, und diejenigen mit der niedrigsten Wahrscheinlichkeit eines Fehlers werden zugeordnet, um Sprachcodierer-Ausgangsbits von größter Wahrnehmungssignifikanz darzustellen, usw., bis herunter zu am wenigstens wahrnehmungssignifikanten Bits, die den codierten Bits zugeordnet sind, die am häufigsten als fehlerhaft beobachtet wurden.

[0076] Nimmt man wiederum Bezug auf **Fig.** 2(a), werden nach einem Fehlerkorrektur-Codieren die Codierer-Ausgangsbits durch eine Verschachtelungseinheit **22** bezüglich der Zeit verteilt. Die Verschachtelungseinheit **22** stellt sicher, dass benachbarte bzw. angrenzende Bits aus dem Fehlerkorrektur-Codierer **21** nicht benachbart über die Luft übertragen werden, um eine Fehleranhäufung zu vermeiden, die das Risiko eines Verlierens von allen übertragenen Bits erhöhen würde, die von einem bestimmten Sprachbit abhängen, um somit zu verhindern, dass ein Sprachbit entdeckt wird. Weiterhin kann das Verschachtelungsmuster gemäß dem Prinzip gewählt werden, das mit der Hilfe der **Fig.** 1 erklärt ist, so dass die ersten Bits aus dem Codierer **21** am nächsten zu (und vorzugsweise zwischen) bekannten Bits (z. B. Synchronisationsbits) platziert werden, und dann, wenn alle solchen Positionen ausgeschöpft sind, das nächste Bit aus dem Codierer **21** am nächsten zu dem ersten Bit platziert wird, usw. Dieser Typ von Verschachtelungseinheit ist als Blockverschachteler bekannt, weil ein Block von M. N Datenbits in einem Block M Frames platziert wird.

[0077] Jedoch kann es auch von Vorteil sein, einen sogenannten Diagonalverschachteler oder einen Blockdiagonalverschachteler zu verwenden. Der Blockdiagonalverschachteler kann mit einer Pi/4-DQPSK-Quaternärmodulation verwendet werden, die zwei codierte Datenbits pro übertragenem Symbol trägt, und zwar mit einer diagonalen Verschachtelung über M = 2 Frames. Wenn die zwei Bits eines Symbols mit "ungeradzahligen" und "geradzahligen" Bits bezeichnet sind, kann es vorteilhaft sein, einen Block von codierten Bits über den geradzahligen Bits eines ersten der M Frames und den ungeradzahligen Bits des zweiten der M Frames anzuordnen. Dann wird der nächste Block von codierten Bits über den geradzahligen Bits des zweiten Frames und den ungeradzahligen Bits des nächsten Frames angeordnet, usw.

[0078] Der Verschachteler **22** kann einen Pufferspeicher enthalten, der einen oder zwei Frames von codierten Sprachbits hält, und wird dann eine Hälfte der codierten Bits eines vorherigen Sprachframes mit einer Hälfe der codierten Bits eines aktuellen Sprachframes für eine Übertragung im aktuellen TDMA-Burst verschachteln. Eine Verwendung eines Zeitvielfachzugriffs ist lediglich beispielhaft, und Fachleute auf dem Gebiet werden erkennen, dass die vorliegende Erfindung auch beispielsweise mit einem Frequenzvielfachzugriff oder einem Codevielfachzugriff verwendet werden kann. Das übertragene Signal kann auch vorteilhaft bezüglich der Frequenz übersprungen werden, so dass die unterschiedlichen Bursts oder Segmente, über welche die Codebits verschachtelt sind, auf unterschiedlichen Frequenzen mit unterschiedlichen, nicht korrelierten Störern sind.

[0079] Verschachtelte Bits vom Verschachteler **22** werden zur Übertragung unter Verwendung einer gewählten Modulationstechnik in einem Wellenformgenerator **23** in eine Wellenform umgewandelt. Beispielsweise kann eine einer Gauß-Filterung unterzogene minimale Phasenumtastung (GMSK) verwendet werden, wobei eine binäre 1 als eine Phasenänderung von +90 Grad codiert ist und eine binäre 0 als eine Phasenänderung von –90 Grad gegenüber der vorherigen Phase. Bei einer GMSK erfolgen die Phasenänderungen stufenweise, um so viel wie möglich von dem übertragenen Spektrum innerhalb eines zugeteilten Kanals zu enthalten. Die stufenweisen Phasenänderungen werden durch Anlegen eines positiven oder eines negativen Impulses erzeugt, und zwar gemäß der Datenbitpolarität, an ein Gauß-Filter, dessen Ausgabe als eine Rate einer Änderung einer Phase (d. h. eine Frequenzverschiebung) angesehen wird. Diese Rate einer Änderung einer Phase wird dann Modulo-2Pi integriert, und der resultierende absolute Phasenwert unter Verwendung einer Nachschautabelle in Kosinus- oder Sinuswerte umgewandelt, um I- und Q-Quadraturmodulationswerte zu erzeugen. Diese werden durch eine D/A-Umwandlung in analoge Wellenformen umgewandelt, der eine Tiefpassfilterung **24** folgt. Wenn die Gauß-Filterung mit einem finiten Impulsantwortfilter (FIR-Filter) von vernünftig kurzer Impulsantwort angenähert werden kann, z. B. drei Bitperioden, dann hängen die ausgegebenen I,Q-Wellenformen über jeder Bitperiode von einer vergangenen Vorgeschichte der drei Bits ab, und sind somit anzahlmäßig nur acht. Alle acht Wellenformen können vorberechnet werden und in einer Nachschautabelle im Modulati-

ons-Wellenformgenerator **23** gespeichert werden und zur Ausgabe gemäß den letzten drei Eingangsbits vom Verschachteler **22** ausgewählt werden.

[0080] Alternativ kann der Wellenformgenerator **23** eine vier Pegel-Modulation implementieren, wie beispielsweise eine einer Gauß-Filterung unterzogene 4-MSK, bei welcher Paare von Eingangsbits zu einer Zeit in eine der vier Phasenänderungen +45, –45, +135 oder –135 Grad codiert werden. Ein solcher Modulator kann gemäß demselben Prinzip arbeiten, wie es oben zum Erzeugen von 90-Grad-Phasenänderungen im GMSK-Fall umrissen ist.

[0081] Eine Modulation mit nicht konstanter Amplitude, wie beispielsweise Pi/4-DQPSK, kann auch verwendet werden, bei welcher Paare von Bits zu einer Zeit in einen Vektorübergang von einer vorherigen I,Q-Vektorposition zu einer neuen Position codiert werden, die gegenüber der alten Position um +45 oder +135 Grad gedreht ist. Jedoch werden Übergänge von der alten zu der neuen Position in diesem Fall nicht durch stufenweises Anwenden der Phasendrehung geglättet, sondern vielmehr durch ein Filtern der I,Q-Wellenformen unter Verwendung von beispielsweise einem Kosinusfilter mit Wurzelziehung, wie es im US-T. I. A. standard IS54 für digitale zellulare Telefonsysteme beschrieben ist.

[0082] Es ist auch möglich, die vorliegende Erfindung mit Modulationsschemen zu implementieren, die eine größere Anzahl von Pegeln als vier haben, wie beispielsweise 8-DPSK oder 16 DQAM. Eine Verwendung einer differentiellen Modulationstechnik ist kein Erfordernis der Erfindung, und ein Fachmann auf dem Gebiet kann auf einfache Weise die Technik an den Fall anpassen, bei welchem Information in einer Sequenz von absoluten Phasenwerten getragen wird, wie beispielsweise P5K, QPSK, 8-PSK oder 16 QAM. Vielmehr lässt die vorliegende Erfindung die Verwendung einer differentiellen Modulation ohne den für gewöhnlich zugehörigen Verlust an Leistung von 3 dB im Vergleich mit einer kohärenten Demodulation zu, während die Analogie bezüglich des Falls einer absoluten oder einer kohärenten Phasenmodulation darin besteht, dass die Erfindung eine Auflösung der für gewöhnlich zugehörigen Phasenvieldeutigkeit ermöglicht bzw. erleichtert.

[0083] Die durch irgendeines der obigen Verfahren erzeugten I,Q-Modulationswellenformen werden auf einen Quadraturmodulator angewendet, der zusammengesetzt ist aus einem 90-Grad-Phasenteiler **27**, Ausgleichsmodulatoren **25**, **26**, einem Kombinierer **29** und einem Filter **30**. Diese Schaltung wendet die I-Modulationswellenform multiplikativ auf eine Kosinus-Trägerwelle bei einer Frequenz an, die durch einen Frequenzsynthesizer **28** bestimmt wird, und die Q-Wellenform auf einen Sinusträger bei derselben Frequenz. Der Kombinierer **29** bildet $I \cdot \cos(\omega t) + Q \cdot \sin(\omega t)$, welches die erwünschte modulierte Wellenform ist, aber bei einer Zwischenfrequenz ω. Diese wird dann zu der erwünschten Übertragungsfrequenz unter Verwendung einer heterodynen Mischung in einem Aufwärtswandler **31** mit einem Lokaloszillatorsignal vom Synthesizer **28** umgesetzt. Die Lokaloszillatorfrequenz kann durch eine Steuerung (nicht gezeigt) variiert werden, um eine Übertragung auf unterschiedlichen erwünschten Kanälen zu bewirken.

[0084] Das Übertragungsfrequenzsignal aus dem Aufwärtsmischer **31** wird durch ein Filter **32** ausgewählt, das andere unerwünschte Frequenzen unterdrückt, und wird dann in einem Leistungsverstärker **33** zu einem erwünschten Übertragungsleistungspegel verstärkt. In einem TDMA-System dirigiert ein TDMA-Zeitgabegenerator die Zeitgabe von Datenflüssen zwischen den unterschiedlichen Blöcken und steuert auch den Leistungsverstärker beim Einschalten und Ausschalten auf eine ruhige Weise mittels eines Aufwärts- und Abwärts-Rampenbildens des Leistungspegels, um spektrale Spritzer in Nachbarkanäle zu vermeiden.

[0085] Die I,Q-Modulatoren können durch ausgeglichene I,Q-Filter betrieben werden, wie sie beispielsweise im US-Patent US-A-5,530,722 mit dem Titel "Quadrature Modulator with Integrated Distributed RC Filters" beschrieben ist, das am selben Tag wie die vorliegende Anmeldung eingereicht wurde. Ein beispielhaftes Empfänger-Blockdiagramm zum Implementieren der vorliegenden Erfindung wird nun mit der Hilfe der **Fig.** 3 beschrieben werden.

[0086] Ein Superheterodyn-Empfänger mit einen Antennenfilter **40**, einem Verstärker **41** mit niedrigem Rauschen, einem Bildzurückweisungsfilter **42**, einem Mischer **43** und einem programmierbar synthetisierten Lokaloszillator **44**, **45**, **56** wird zum Abwärtsmischen der Frequenz des empfangenen Signals zu einer geeigneten Zwischenfrequenz verwendet, wo sie mit einem Festfrequenz-IF-Filter **46** gefiltert werden kann. Ein integrierter Schaltungs-IF-Chip **48** kann dann dazu verwendet werden, das Signal unter Verwendung eines zweiten Lokaloszillators weiter abwärts zu mischen, welches auch mit der Hilfe einer Frequenzsynthesizer-Steuerschaltung **56** und einem Schleifenfilter **47** synthetisiert werden kann. Das weiter abwärts gemischte Signal wird weiterhin durch den Chip **48** verstärkt und unter Verwendung von zweiten IF-Filtern **49**, **50** weiter gefiltert. Das verstärkte Signal wird dann unter Verwendung eines A/D-Wandlers **51** digitalisiert, der vorteilhafterweise die logische Polaritätstechnik verwenden kann, die im US-Patent Nr. 5,048,059 beschrieben ist. Unter Verwendung des logischen Polaritätsverfahrens wird die Endverstärkerstufe zu einer Sättigung getrieben, so dass ein Signal mit einer im Wesentlichen konstanten Amplitude auftaucht, was Information über die Funksignalphase bewahrt, nicht aber seine Amplitude. Die Amplitudeninformation wird separat durch Detektoren erzeugt, die an jeder der internen Verstärkerstufen im IF-Verstärkerchip **48** angebracht sind, um ein Signal zu erzeugen, das ungefähr proportional zum Logarithmus der Amplitude ist, was allgemein als Funksignalstärkeindiz (RSSI = Radio Signal Strength Indication) bekannt ist.

[0087] Die Phaseninformation wird von der gesättigten Ausgabe unter Verwendung einer A/D-Umwandlungstechnik, wie beispielsweise einer Tiefpassfilterung, gefolgt durch eine Quadraturabtastung und eine A/D-Umwandlung der Abtastungen zum Erhalten von numerischen Werten, die ungefähr proportional jeweils zum Kosinus und zum Sinus der Phase sind, extrahiert, oder durch Verwendung eines direkten Phasendigitalisierers, wie er beispielsweise im US-Patent Nr. 5,084,669 offenbart ist, das den Titel "Direct Phase/Frequency Digitization" hat. Wenn das letztere Verfahren verwendet wird, kann dann der so erhaltene Phasenwert unter Verwendung von beispielsweise Nachschautabellen in Kosinus- und Sinuswerte umgewandelt werden. Der Kosinus- und der Sinuswert werden dann mit dem entdeckten Amplitudensignal multipliziert, um die kartesische komplexe Darstellung des Funksignals zu rekonstruieren. Das Amplitudensignal wird aus dem RSSI-Signal beispielsweise mittels einer antilogarithmischen Nachschautabelle entdeckt.

[0088] Es ist manchmal wünschenswert, zuerst einen Skalierungswert von den RSSI-Werten zu subtrahieren, wie beispielsweise den größten Wert in einem Block von Signalabtastungen (Blockskalierung), um sicherzustellen, dass die kartesischen Werte innerhalb eines bestimmten Bereichs liegen, z. B. zwischen +1 und –1. Solche Operationen für eine Blockskalierung und eine logarithmische-kartesische Umwandlung werden durch eine digitale Signalverarbeitungseinheit **53** an digitalisierten Signalabtastungen durchgeführt, die in einem Pufferspeicher **52** gesammelt sind. Die digitale Signalverarbeitungseinheit **53** kann auch programmiert sein, um die zum Implementieren der vorliegenden Erfindung aus einer Kombination einer Demodulation und eines Decodierens erforderliche Verarbeitung durchzuführen. Die digitale Signalverarbeitungseinheit **53** kann dann auch einen VSELP- oder einen RELP-Sprachdecodierer zum Umwandeln decodulierter Daten in PCM-Sprachabtasten implementieren, welche dann unter Verwendung eines D/A-Wandlers **54** in eine analoge Sprachwellenform umgewandelt werden.

[0089] **Fig.** 4 stellt eine beispielhafte Implementierung eines Demodulators gemäß der vorliegenden Erfindung dar. Jede Größe Z in der Spalte von komplexen Matrizen, die mit Z0 bis Z15 markiert sind, ist eine Vielzahl von komplexen Werten, d. h. eine komplexe Vektormatrix, wobei jeder Wert die beste aktuelle Abschätzung des Funksignalvektors genau vor einer zu verarbeitenden Signalabtastung darstellt. Die Anzahl von Z-Werten in jeder Matrix kann gleich dem Zweifachen der Anzahl von bekannten Symbolen sein, die in den Signalbursts enthalten sind, die gerade verarbeitet werden, und zwar in dem Fall einer differentiellen Modulation, oder gleich der Anzahl bekannter Symbole in dem Fall einer absoluten Phasenmodulation.

[0090] Beispielsweise dann, wenn eine Verschachtelung differenziell codierte Symbole über zwei TDMR-Bursts streut, die mit sechs bekannten Synchronisationssymbolen pro Burst durchsetzt sind, ist die Anzahl von Z-Werten in jeder Matrix, wie beispielsweise Z0, 24. Die Werte stellen Abschätzungen des Signalvektors bei einer Position zur linken bzw. zur rechten eines bekannten Symbols dar. Die Z-Werte werden zu Abschätzungen des Vektors auf jeder Seite des jeweiligen differentiellen Synchronisationssymbols initialisiert. Beispielsweise dann, wenn für das erste differentielle bekannte Symbol im ersten TDMA-Burst bekannt ist, dass es eine +90-Grad-Drehung hat, und die Empfangssignalvektoren vor und hinter dieser bekannten Drehung jeweils Z(i – 1) und Z(i) sind, ist die beste Abschätzung der Phase auf der linken Seite des (d. h. vor dem) Synchronisationssymbols (Z(i – 1) – jZ(i))/2 und ist die beste Abschätzung des Signalvektors auf der rechten Seite des (d. h. hinter dem) Synchronisationssymbols (jZ(i – 1) + Z(i))/2. Das erstere wird durch Drehen von Z(i) in Rückwärtsrichtung über die bekannte 90-Grad-Phasenverschiebung erhalten, um Z(i) mit dem vorherigen Wert Z(i – 1) auszurichten, und darauffolgendes Bilden eines Durchschnitts der beiden. Das letztere wird durch Drehen von Z(i – 1) in Vorwärtsrichtung über die bekannten 90-Grad-Phasenverschiebung erhalten, um Z(i – 1) mit Z(i) auszurichten, und dann durch Bilden eines Durchschnitts der beiden. Später wird beschrieben werden, wie eine Variablenskalierung der Summe eher als ein einfaches Teilen der Summe durch die beiden vorteilhaft sein kann. Die Skalierung sollte tatsächlich in einer gleichen Rauscherwartung bei allen kombinierten Werten resultieren.

[0091] Ein Wiederholen dieses Prozesses für alle bekannten Werte, wie es in **Fig.** 1 angezeigt ist, stellt Startwerte für die Referenzvektoren Z zur Verfügung. An dieser Stelle sind die Werte Z0, Z1, ..., Z15, die zu unterschiedlichen Hypothesen für die decodierte Sequenz gehören, alle gleich.

[0092] Eine Decodulation gemäß diesem beispielhaften Ausführungsbeispiel geht nun weiter, wie es folgt. Eine Hypothese für eine übertragene Symbolsequenz, die alle Positionen im Faltungscodier-Schieberegister füllt, wird durchgeführt. Beispielsweise dann, wenn ein Code mit einer Beschränkungslänge von 5 verwendet wird, wird das Schieberegister fünf Stufen haben. Ein Auswählen der Zeile in **Fig.** 4 entsprechend der Hypothese 0000 und ein Bilden einer Hypothese für ein neues 0-Bit ist äquivalent zu einem Bilden einer Hypothese, dass das übertragende Codierregister fünf Nullen bei dieser Stelle enthielt. Die Hypotheseinhalte des Schieberegisters werden dann zum Erzeugen der codierten Bits P1, P2, etc. verwendet, die übertragen worden sein sollten, wenn die Hypothese richtig war. Dies wird durch Einbauen eines Modells des Faltungscodierers in ein Übertragermodell bzw. Sendermodell **60** ausgeführt. Das Übertragermodell **60** enthält auch die bekannten Modulationsparameter, so dass komplexe Signalwerte, die repräsentativ für P1 und P2 sind, vorhergesagt werden können, sowie die Verschachtelungs- oder Bitplatzierungs-Information zum Vorhersagen, wo erwartet werden würde, dass die TDMA-Signalbursts P1, P2, etc. gesehen werden.

[0093] Die ersten Bits P1, P2, die mit der Hilfe des Modells **60** einer Hypothetisierung unterzogen sind, sind gemäß beispielhaften Ausführungsbeispielen der vorliegenden Erfindung diejenigen, die angrenzend bzw. benachbart zu bekannten Symbolen (z. B. Synchronisationssymbolen) angeordnet sind, und sind die U-Bits mit den Zahlen 1 und 2 in **Fig.** 1. Bei diesem Beispiel liegt U1 zwischen zwei bekannten Symbolen K, und somit entspricht eine der Vektorabschätzungen, die hierin mit ZR bezeichnet sind, in Z0 (weil die aktuelle Hypothese 0000-x ist) dem Vektor auf der rechten Seite von U1 und auf der linken Seite des folgenden bekannten Symbols, und eine weitere, nämlich ZL, entspricht dem Vektor auf der linken Seite von U1 und auf der rechten Seite des vorangehenden bekannten Symbols. Unter der Annahme einer binären Modulation, wie beispielsweise MSK, für dieses Beispiel, wobei eine binäre "1" durch eine +90-Grad-Phasendrehung von einer Abtastung zur nächsten und eine binäre "0" durch eine –90-Grad-Drehung dargestellt ist, würde dann, wenn P1 als "1" hypothetisiert wird, erwartet werden, dass ZR relativ zu ZL um +90 Grad gedreht werden wird. Daher wählt das Modell 60 die geeignete aktuelle Vektorreferenz aus, nämlich ZL aus der Matrix Z0, wendet eine 90-Grad-Drehung auf die Vekaorreferenz an und vergleicht das gedrehte ZL bei diesem ersten darstellenden Beispiel mit dem besten erwarteten Wert ZR auch aus der Referenzmatrix. Später bei der Verarbeitung, wenn alle unbekannten Symbole, die zwischen bekannten Symbolen liegen, verarbeitet worden sind, wird ein Vergleich mit empfangenen Abtastungen durchgeführt werden, die durch die Entschachtelungseinheit vom Pufferspeicher **52** ausgewählt sind, die benachbart zu bereits verarbeiteten Abtastungen liegen.

[0094] Durch ein Vergleichen von jZL mit ZR in einem Komparator **61** wird der quadratische Fehler $|ZR - jZL|^2$ berechnet. Dieser Fehler ist die Fehlanpassung zwischen dem Signalwert, der erwartet wird, wenn die Hypothese wahr wäre, und dem Signalwert, der aktuell empfangen wird. Die neue Fehlanpassung wird in einem Addierer **62** zu irgendeiner existierenden kumulativen Fehlanpassung (d. h. dem Pfadmaß) addiert, die zu dieser Hypothese gehört, wobei die Pfadmaße anfangs auf Null eingestellt worden sind, um eine neue kumulative Fehlanpassung oder ein neues Pfadmaß zu finden. Der obige Prozess wird für P2 für eine Codierung mit der Rate 1/2 wiederholt, was darin resultiert, dass zwei Bits für jedes Informationsbit übertragen werden. Gleichermaßen wird dieser Prozess für P3, P4, etc. wiederholt, wenn die Codierrate 1/3, 1/4 oder niedriger ist, wobei berücksichtigt wird, wo früher unbekannte Werte durch den Verschachteler platziert wurden, und in Bezug auf ihre nächsten Vektorreferenzen. Beispielsweise wird P2 gemäß **Fig.** 1 in einem TDMA-Burst 2 platziert und entspricht dem mit "2" bezeichneten U-Bit. Seine Fehlanpassung wird analog zu derjenigen für P1 berechnet, aber unter Verwendung der Elemente eines Vektorreferenzfeldes Z0, das aus den bekannten Symbolen K auf einer Seite von U2 berechnet ist.

[0095] Das neue Pfadmaß wird an dieser Stelle als Kandidatenwert für einen neuen Zustand/Hypothese 0000 angesehen, wenn die Hypothese 1000, der ein neues Bit 0 folgt, ein gleichermaßen möglicher Kandidat ist. Wenn der Faltungscodierer sein Codierungsregister um eine Stelle nach links verschiebt, wird sich 1 oder 0 am weitesten links aus dem Register herausbewegen, was in beiden Fällen 0000 in den vier Positionen am weitesten links lässt. Somit wird ein Kandidatenmaß auf dieselbe Weise wie oben berechnet, aber beginnend mit der Hypothese 1000-0 und unter Verwendung der Z-Werte von Z8 als ihren entsprechenden Referenzvektor. Das zweite Kandidatenmaß vom Addierer **64** wird dann mit dem ersten vom Addierer **62** im Komparator **63** verglichen, und das niedrigere der beiden wird ausgewählt. Wenn der niedrigere Wert vom Addierer **64** kommt, dann wird für die Hypothese 1000-0 bestimmt, dass sie eine bessere Hypothese als 0000-0 ist, und Zustand/Hypothese 1000 wird der "beste Vorgänger" für einen neuen Zustand 0000 genannt. Die Inhalte des "bereits verarbeiteten" Speichers, der zu dem besten Vorgänger gehört, werden dann derart ausgewählt, dass sie die Inhalte des bereits verarbeiteten Symbolspeichers des neuen Zustands 0000 sind, und "1" oder "0" wird von der rechten Seite dieses Speichers aus nach links verschoben, und zwar in Abhängigkeit davon, ob der beste Vorgänger 1000 oder 0000 war. Das niedrigere der Kandidatenmaße wird auch das neue Pfadmaß, und das zu dem besten Vorgänger gehörende Referenzvektorfeld wird nach einem Aktualisieren ausgewertet, um das Referenzvektorfeld für den neuen Zustand zu sein. Das ausgewählte Referenzvektorfeld wird aktualisiert, wie es folgt.

[0096] Es soll angenommen werden, dass der beste Vorgänger eine Vorhersage für P1 von Binär 0 entsprechend einer Phasendrehung von –90 Grad ergab, und zwar unter der Annahme von beispielsweise einer MSK-Modulation. Dann sollte der Signalvektor auf der rechten Seite U1 um –90 Grad relativ zu dem Signalvektor auf der linken Seite gedreht sein. In der Tat ist im speziellen Fall eines unbekannten Symbols, das zwischen zwei bekannten Symbolen in Sandwichbauweise liegt, wie beispielsweise U1, wobei die Sequenz KUK aus komplexen Signalabtastungen Z(i – 2) auf der linken Seite des ersten K-Symbols, Z(i – 1) zwischen dem ersten K-Symbol und dem U-Symbol, Z(i) auf der rechten Seite des U-Symbols und Z(i + 1) hinter dem zweiten K-Symbol besteht, nun bekannt. Beispielsweise dann, wenn das erste K-Symbol eine MSK-"1" ist, d. h. eine +90-Grad-Phasendrehung, und das zweite K-Symbol, wie U1, eine –90-Grad-Drehung ist, sollten die folgenden Beziehungen Gültigkeit haben:

```
Z(i-1) = jZ(i-2)
Z(i)   = -jZ(i-1) = Z(i-2)
Z(i+1) = -jZ(i) = -Z(i-1) = -jZ(i-2)
```

[0097] Somit ist eine bessere Abschätzung von Z(i + 1) (zur Rechten des zweiten K-Symbols):

$$Z'(i + 1) = (Z(i + 1) – jZ(i) – Z(i – 1) – jZ (i – 2))/4 \quad (1)$$

[0098] Dann folgt, dass bessere Abschätzungen der anderen Werte einfach folgende sind:

$$Z'(i) = jZ'(i + 1); Z'(i – 1) = jZ'(i); Z'(i – 2) = –jZ'(i – 1)$$

[0099] Die entsprechenden Elmente in dem Feld bzw. in der Matrix, z. B. Z8, wenn 1000 der beste Vorgänger ist, oder Z0, wenn 0000 der beste Vorgänger war, werden somit zu den neuen Werten aktualisiert, bevor sie zu dem neuen Z0-Referenzvektorfeld für einen Nachläuferzustand 0000 werden.
[0100] Ein Wiederholen des obigen Prozesses für eine neue Bit-Hypothese von "1" anstelle von "0" erzeugt einen neuen Zustand 0001 auf dieselbe Weise. Ein Wiederholen des obigen beginnend mit wiederum Paaren von Hypothesen 0001-1001; 0010-1010; etc. erzeugt dann den Rest der neuen Zustände, so dass es am Ende einer Iteration dieselbe Anzahl von neuen Zuständen (16 bei diesem Beispiel) wie zu Beginn gibt.
[0101] Die obige Prozedur wird zum sequentiellen Verarbeiten der empfangenen Abtastungen in der in **Fig.** 1 bezeichneten Reihenfolge verwendet, nämlich U1, U2, ..., etc. Es kann gesehen werden, dass dieser beispielhafte Prozess Abtastungen sorgfältig verarbeitet, die zwischen zwei bekannten Symbolen vor anderen Abtastungen liegen, jedoch ist dies nur ein möglicher Aufbau. Wenn Symbole, die in diesen Positionen liegen, ausgeschöpft worden sind, wie es beispielsweise hinter der Symbolnummer U12 in **Fig.** 1 erfolgt, werden die ersten hypothetisierten Datenbits sich um eine Anzahl von Positionen von rechts nach links bewegt haben, um in den Speicher für ein "bereits verarbeitetes Symbol" einzutreten und mehr oder weniger fest bestimmt zu werden. Wenn beispielsweise ein entsprechendes Symbol bei derselben Position in allen Speichern übereinstimmt, dann ist es fest entschieden. Je länger ein Symbol in seinem Durchlauf von rechts nach links überlebt hat, umso größer ist die Chance, dass entsprechende Symbole übereinstimmen werden, und zwar aufgrund des kontinuierlichen Überschreibens der Speicherinhalte durch den besten von zwei möglichen Vorgängern. Auf dieselbe Weise werden die unter Verwendung dieser Symbolhypothese berechneten Vektorreferenzen sicherer geworden sein, so dass der Referenzwert, der bei einer Verarbeitung von U13 verwendet werden wird, welcher sich von bereits verarbeiteten Werten von U1, U5, U9 und vier bekannten Symbolen K ableitet, für die letzten 12 verarbeiteten Symbole überlebt haben wird, und somit eine geringere Wahrscheinlichkeit dafür haben wird, dass er nicht richtig ist. Auf diese Weise enthält die zum Decodieren des Symbols U13 verwendete Vektorreferenz weniger Unsicherheit aufgrund eines Rauschens, als sie es basierend auf dem einzigen K-Symbol zu ihrer Rechten gehabt hätte. Auf dieselbe Weise leitet sich die zum Decodieren von U14 im TDMA-Burst 2 verwendete Vektorreferenz von bereits verarbeiteten Symbolen U2, U6, U10 und den vier K-Symbolen im Burst 2 ab, usw., für Signalabtastungen bis zu U20.
[0102] An dieser Stelle gibt es keine weiteren unbekannten Symbole, die neben den bekannten Symbolen liegen, aber ein Symbol U21 liegt zur linken einer bereits verarbeiteten Symbolkette (beim Lesen von links nach rechts im Burst 1), nämlich U13, K, U1, K, U5, K, U9, K, U17. Somit existiert eine Vektorreferenz in jedem Zustand, der auf den vorherigen Hypothesen basiert, die zum Gelangen zu diesem Zustand durchgeführt sind. Wenn die Hypothesen wahr sind, wird die Vektorreferenz genau sein, was zu einer genauen Vorhersage von U21 führt, und somit zu einem geringen Hinzufügen zu dem Pfadmaß beim Verarbeiten von U21. Wenn andererseits die Hypothesen Fehler enthalten, wird die Vektorreferenz nicht so genau sein, und wird U2l nicht so genau vorhergesagt werden, und eine größere quadratische Fehlanpassung wird hinzugefügt werden, was die Wahrscheinlichkeit reduziert, dass der Zustand als "bester Vorgänger" überleben wird.
[0103] Ein herkömmliches differentielles Decodieren leidet normalerweise an einem Verlust von ungefähr 3 dB aufgrund dessen, dass der Referenzvektor ein gleiches Ausmaß an Unsicherheit aufgrund eines Rauschens wie die Signalabtastung, die gerade verarbeitet wird, hat. Unter Verwendung des oben beschriebenen erfinderischen Demodulators wird dieser Verlust in großem Ausmaß vermieden, da der Referenzvektor auf einer Anzahl von bereits verarbeiteten Signalwerten basiert, deren Unsicherheit durch die Faltungsdecodierung reduziert worden ist. Eine intelligente Auswahl des Verschachtelungsmusters, so dass ein nächstes Symbol, das zu verarbeiten ist, angrenzend bzw. benachbart zu einem bereits verarbeiteten Symbol liegt, lässt eine progressive Erweiterung des Decodulationsprozesses auf irgendeine Anzahl von unbekannten Symbolen zu, während der Vorteil eines reduzierten Rauschens beibehalten wird. Es ist wünschenswert, eine Decodulation durch Verarbeiten unbekannter Symbole zu beginnen, die am nächsten zu bekannten, z. B. Synchronisationssymbolen, liegen, da die Reduzierung von Rauschen für diese Symbole direkt erhalten wird.

[0104] Eine der Aufgaben der vorliegenden Erfindung besteht im Ermöglichen bzw. Erleichtern einer differentiellen Modulationstechnik, die weniger empfindlich gegenüber einer Zeitvariation des Kanals als die alternative kohärente Demodulationstechnik ist. Wenn der Kanal schnell variiert, und zwar über dem Raum von einigen Symbolen, ist eine gewisse Modifikation für das Verfahren zum Aktualisieren der Vektorreferenzen nötig, um die Tatsache zu berücksichtigen, dass die angenommenen Vektorbeziehungen, die beispielsweise in Gleichung (1) ausgedrückt sind, ungenau sein können. Aufgrund eines Mehrwegeschwunds bzw. Mehrwege-Fadings werden sich der Ausbreitungsverlust und eine Phasenverschiebung durch den Funkkanal mit der Zeit ändern, so dass Gleichungen, wie beispielsweise (1), modifiziert werden sollten, um irgendeine Form eines exponentiellen Vergessens und/oder Einbauens der abgeleiteten Abschätzungen zuzulassen.
[0105] Eine Ableitungsabschätzung kann beispielsweise durch Annehmen eingebaut werden, dass sich der Kanal-Ausbreitungsverlust um dA Knoten pro Symbol ändert (d. h. die empfangene Amplitude wird ein Symbol später um den Faktor EXP (dA) höher sein), und dass sich die Phase pro Symbol um dTHETA ändert. Diese Annahmen können dadurch kombiniert werden, dass gesagt wird, dass der Kanal ein Symbol später durch den komplexen Multiplikationsfaktor Q = EXP(dA + jdTHETA) modifiziert werde wird. Die Gleichung (1) sollte dann wie folgt gelesen werden:

$$Z'(j + 1) = (Z(i + 1) – jQ \cdot Z(i) – Q^2 \cdot Z(i – 1) – jQ^3 \cdot Z(1 – 2)/4 \quad (2)$$

[0106] Es kann auch wünschenswert sein, dass weitere Amplitudengewichte in der obigen Kombination enthalten sind, um die stärker empfangenen Abtastungen höher als die weniger stark empfangenen Abtastungen zu gewichten, und die Ableitung von solchen Gewichten zum Maximieren der Signal-zu-Rausch-Erwartung bei dem Ergebnis wird von Fachleuten auf dem Gebiet ableitbar sein, und dies wird demgemäß hierin nicht weiter diskutiert.
[0107] Ein geeigneter Wert von Q kann für jeden Zustand berechnet werden und zusammen mit den entsprechenden Referenzvektorfeldwerten für diesen Zustand gespeichert werden. Der Wert von Q für einen Zustand wird dadurch abgeschätzt, dass angenommen wird, dass die bereits verarbeiteten Symbole in diesem Zustand und die aktuelle Hypothese wahr sind. Die nominellen Vektordrehungen, die für diese Symbole erwartet werden, können dann entfernt werden, wobei nicht modulierte Werte zurückgelassen werden, die nur von der Variation aufgrund des Kanals abhängen.
[0108] Das exakte Verfahren zum Bestimmen der Kanalvariation ist nicht wesentlich für die vorliegende Erfindung und könnte beispielsweise ein bestes Anpassen einer einheitlichen Driftrate an die Phase, eine Tiefpassfilterung der nicht modulierten Werte unter Verwendung eines Filters mit einer Bandbreite, die nur zum Durchlassen der maximalen erwarteten Dopplerfrequenz ausreichend breit ist, oder eine Kalman-Filterung enthalten.
[0109] Ein weiteres beispielhaftes Ausführungsbeispiel der vorliegenden Erfindung vermeidet den Schritt zum Speichern und Aktualisieren von Referenzvektorwerten durch Berechnen von ihnen "im Vorübergehen". **Fig.** 5 zeigt eine beispielhafte Modifikation des Blockdiagramms der **Fig.** 4 für dieses Ausführungsbeispiel.
[0110] Gemäß **Fig.** 5 ist eine Referenzvektor-Abschätzeinheit **65** im Vergleich zu **Fig.** 4 hinzugefügt worden, und die Spalte von Referenzvektormatrizen ist entfernt worden. Bei dem vorangehenden Betriebsbeispiel war ein Decodieren eines unbekannten Symbols U13 mit der Hilfe eines Referenzvektors durchzuführen, der unter Verwendung von bereits verarbeiteten Symbolen U1, US und U9 sowie einigen bekannten Symbolen abgeleitet ist. Der Einfachheit halber ist die Referenzvektor-Abschätzeinheit **65** derart gezeigt, dass sie bereits verarbeitete Symbole U1, U5 und U9 von dem Speicher für "bereits verarbeitete Symbole" erhält, obwohl dieser Speicher nicht notwendigerweise U1, U5 und U9 enthält, sondern eher decodierte Symbole. Unter Verwendung von beispielsweise eine Faltungscodierung mit einer Rate von 1/2 können die "bereits verarbeiteten Symbole" nur halb so viele decodierte Symbole wie codierte (U-)Symbole enthalten. Nichts desto weniger können die codierten (U-)Symbole unter Verwendung eines Modells des Sendercodierers aus den decodierten Symbolen rekonstruiert werden. Alternativ kann dieses erneute Codieren kontinuierlich durchgeführt werden, so dass die erforderlichen U-Werte in den Speichern für "bereits verarbeitete Symbole" gespeichert werden. Genauso wie die U-Werte verwendet die Referenzvektor-Abschätzeinheit **65** die entsprechenden empfangenen Abtastwerte. Diese sind der Einfachheit halber nicht derart gezeigt, dass sie an einem Block **65** angelegt sind, sondern können beispielsweise in einer 17-ten Zeile des Symbolspeichers unterhalb entsprechender U-Symbole gespeichert sein. Die Abschätzeinheit **65** bildet die Referenzvektorabschätzungen durch Drehen von Signalabtastungen unter Verwendung eines entsprechenden U-Symbols, um die Richtung und das Ausmaß einer Drehung zu bestimmen, und dann unter Bildung einer gewichteten Summe, einschließlich irgendwelcher Signalabtastungen in Bezug auf umfasste bekannte Synchronisationssymbole. Die verwendeten Gewichtungen können ein progressives exponentielles Vergessen von älteren Werten berücksichtigen, um die Zeitvariation des Kanals zu berücksichtigen.
[0111] Einigen der Konzepte, die bei beispielhaften Decodulationssystemen und -verfahren gemäß der vorliegenden Erfindung verwendet werden, sind individuell Namen zugeteilt worden, die von Fachleuten auf dem

Gebiet erkannt werden. Wenn beispielsweise ein U-Wert, der zum Berechnen einer Referenzvektorabschätzung verwendet wird, nur auf Symbolen in dem Abschnitt basiert, der mit "aktuelle Symbolhypothese" markiert ist, sind alle möglichen Hypothesen innerhalb dieses Abschnitts noch offen, und dieser U-Wert bildet einen Teil des sequentiellen Abschätzungsprozesses für eine maximale Wahrscheinlichkeit der Maschine. Ein solcher U-Wert wird demgemäß gemeinhin als "Abgriff für eine maximale Wahrscheinlichkeit" oder "Viterbi-Abgriff" bei dem Symbolspeicher, der als Schieberegister angesehen wird, genannt. Wenn ein U-Wert, der zum Berechnen einer Referenzabschätzung verwendet wird, aus der Seite am weitesten links des Symbolspeichers genommen wird, wo alle entsprechenden hypothetisierten Symbole zu einer einzelnen Entscheidung konvergiert sind, wird es ein Entscheidungsrückkopplungsabgriff (DFE-Abgriff) genannt. Wenn jedoch der U-Wert auf der linken Seite von "aktuelle Symbolhypothesen" genommen wird, aber zur Rechten von der Stelle, wo Symbolwerte zu demselben Wert konvergiert sind, wird er ein "Nach-Überlebender"-Abgriff genannt, und seine Verwendung wird "Verarbeitung für ein Überleben" genannt, um anzuzeigen, dass jede Hypothese oder jeder Zustand einen anderen "späteren Überlebenden" hat, der in die aktuelle Maßberechnung eintritt. Die systematische Kombination von Viterbi-Abgriffen, DFE-Abgriffen und Nach-Überlebender-Abgriffe werden, wie es oben beschrieben ist, bei beispielhaften Ausführungsbeispielen der vorliegenden Erfindung zum Berechnen von Referenzvektoren für eine differentielle Demodulation und Decodierung verwendet.

[0112] Eine alternative Offenbarung bzw. Erscheinung der Erfindung wird nun mit der Hilfe der **Fig.** 6 dargestellt werden. **Fig.** 6 stellt die Anordnung eines ersten Blocks von Datenbits a1, a2, a3, ... zwischen zwei übertragenen Blöcken A und B dar. Die ungeradzahligen Bits a1, a3, a5, ... sind in einem ersten übertragenen Block A zwischen geradzahligen Bits x2, x4, x6, ... eines vorherigen Datenblocks angeordnet. Die geradzahligen Bits a2, a4, a6, ... sind zwischen ungeradzahligen Bits eines Datenblocks b1, b2, b3, b4, ... in einem zweiten übertragenen Block B angeordnet. Die übrigen, nämlich die ungeradzahligen Bits des Datenblocks b1, b2, b3, ... sind in einem dritten übertragenen Block zwischen ungeradzahligen Bits eines neuen Datenblocks c1, c2, c3, ... angeordnet usw.

[0113] Die Datenbits x1, x2, x3, ..., a1, a2, a3, ..., b1, b2, b3, ..., c1, c2, c3, ... sind die sequentiellen Ausgangsbits eines Fehlerkorrektur-Codierers vom vorgenannten Typ, z. B. eines Codierers für eine durchbrochene Faltung. Die Bits werden auf eine Funkfrequenzwelle durch eine differentielle Phasenmodulationstechnik mit entweder einer konstanten oder einer variablen Amplitude moduliert, wie beispielsweise GMSK, DPSK, DQPSK, etc., um komplexe Vektorwellenformen für eine Übertragung mit Abtastwerten z1, z2, z3, ... bei datensymbolbeabstandeten Intervallen zu erzeugen. Somit wird das Symbol a2 in **Fig.** 6 in die Phasenänderung zwischen Vektorabtastungen z1 und z2 codiert, wird das Symbol b2 in die Phasenänderung zwischen Vektorabtastungen z2 und z3 codiert, etc.

[0114] Eine Decodulation dieses Formats geht gemäß diesem beispielhaften Ausführungsbeispiel der Erfindung weiter, wie es folgt. Es soll angenommen werden, dass der vorherige Block von Bits x1, x2, x3, ... durch den Fehlerkorrektur-Codierer gelassen worden ist und decodiert worden ist. Es soll weiterhin angenommen sein, dass eine richtige Decodierung zu einer hohen Wahrscheinlichkeit verifiziert worden ist, und zwar durch Verwenden einer Fehlererfassungstechnik, wie beispielsweise eines Testens eines sorgfältig eingefügten zyklischen Redundanz-Codes (CRC). Dann sind die Bits x2, x4, x6, ... im übertragenen Block A bekannt. Da X2 bekannt ist, ist somit die Phasenänderung von der komplexen Abtastung u1 zu u2 bekannt, und ein empfangener Vektorwert u1 kann durch diese bekannte Phasenänderung in Vorwärtsrichtung gedreht werden, um u1 mit u2 auszurichten. Die beiden werden dann addiert, um eine bessere Abschätzung des Vektors am Anfang eines differentiellen Symbols a1 zu erhalten.

[0115] Gleichermaßen ist deshalb, weil x4 bekannt ist, die differentielle Phase zwischen u3 und u4 bekannt, was zulässt, dass eine empfangene Abtastung u4 über diesen bekannten Winkel in Rückwärtsrichtung gedreht wird, um sie mit u3 auszurichten und zu dieser zu addieren, um dadurch eine bessere Abschätzung des Vektors u3 am Ende des Symbols a1 zu erhalten. Unter Verwendung dieser verbesserten Abschätzungen des Vektors auf jeder Seite von a1 lässt zu, dass a1 mit einer Verstärkung von beispielsweise 3 dB codiert wird, d. h. der normale differentielle Verlust wird um diese Verstärkung von 3 dB kompensiert, so dass die ungeradzahligen Symbole a1, a3, a5, ... so zu verlässig decodiert werden, als ob der Prozess ein Prozess einer kohärenten Modulation und Demodulation wäre. Unglücklicherweise ist diese Verstärkung noch nicht für die geradzahligen Symbole a2, a4, a6 verfügbar, da ihre Nachbarn b1, b3, b5, ... noch nicht decodiert sind. Jedoch wird die Hälfte der Symbole a1, a3, a5 in den Fehlerkorrektur-Decodierer durch das zuverlässigere Verfahren abgeschätzt worden sein, während die abwechselnden bzw. die anderen Symbole a2, a4, a6 die normale differentielle Leistung zeigen werden. Das Ergebnis besteht darin, dass der Datenblock a1, a2, a3, a4, ... mit einer Zuverlässigkeit decodiert werden wird, die etwa auf dem halben Weg zwischen derjenigen einer kohärenten und differentiellen Modulation und Demodulation ist.

[0116] Nach einem Decodieren des Blocks wird er infolge zu dem Fehlererfassungstest geführt, um zu bestimmen, ob er als ein bekannter Block angesehen werden, und zwar zum Zwecke eines Decodierens der Bits b1, b3, b5, etc.

[0117] Gemäß einem weiteren beispielhaften Ausführungsbeispiel können die "a" Bits decodiert werden,

nachdem die "b" Bits decodiert worden sind. Zu dieser Zeit werden die geradzahligen Bits "a" auch durch bekannte Bits b1, b3, b5, ... umgeben sein, so dass das zuverlässigere Decodierverfahren, das oben beschrieben ist, für sowohl ungeradzahlige als auch geradzahlige Bits von "a" verfügbar ist, und eine vollständige Verbesserung von 3 dB bei einer Leistung ist durch eine solche Retrospektive oder "Mehrfachdurchlass"-Demodulation erreichbar. Es ist zu beachten, dass dieser Prozess iterativ erweitert werden kann, so dass eine Zwerfachdurchlass-Demodulationstechnik "a" Bits wieder nach einem Decodieren von "b" Bits decodiert, und eine Dreifachdurchlass-Demodulationstechnik "b" Bits wieder nach einem Decodieren von "a" Bits für ein erstes Mal decodiert, und dann "a" Bits von einem dritten Mal nach einem Decodieren von "b" Bits für ein zweites Mal, etc. Wenn ein CRC-Fehlererfassungscode bei dem gesamten Datenblock verwendet wird, kann es wertvoll sein, eine Demodulation für nur diejenigen Blöcke von Bits zu wiederholen, die bei der CRC-Prüfung bei einer früheren Gelegenheit fehlerhaft waren. Somit ist ein beispielhaftes Ablaufdiagramm für einen Mehrfachdurchlass-Decodulationsprozess in **Fig.** 7 dargestellt.

[0118] Die Eintrittsstelle zu einem Schritt 100 erfolgt dann, wenn ein zuvor decodierter Block die CRC-Prüfung durchlaufen hat. Der nächste Block kann somit unter Verwendung des vorherigen Blocks, der nun bekannt ist, decodiert werden. Wenn der neue Block bei einem Schritt 102 auch die CRC-Prüfung durchläuft, dann wird der neue Block bei einem Block **104** einfach der vorherige Block, und der Algorithmusprozess springt zurück zum Schritt 100, um den nächsten Block zu decodieren.

[0119] Wenn jedoch die CRC-Prüfung im Schritt 102 fehlschlägt, dann wird die Blocknummer (k), bei welcher der Fehler erfasst wird, auf eine Markierervariable bei einem Block **106** eingestellt. Es ist zu beachten, dass der vorherige Block zu der Blocknummer (k) die CRC-Prüfung durchlief. Bei einem Schritt 108 wird der nächste Block (j) ohne den Vorteil decodiert, dass der vorherige Block (j – 1) (was für die erste Iteration gleich k ist) bekannt ist. Dann wird bei 110 bestimmt, ob der Block (j) richtig decodiert ist. Wenn der Block (j) nicht richtig decodiert ist, wird j im Schritt 112 inkrementiert, und der nächste Block wird bei **108** decodiert. Wenn schließlich ein Block (j) richtig decodiert ist, dann wird der frühere Block j – 1 durch Dekrementieren von j in einem Schritt 114 decodiert. Wenn der frühere Block (j) nicht der ursprünglich erfasste schlechte Block (k) ist, wie es in einem Block **116** bestimmt wird, gibt es einen vorangehenden schlechten Block (j – 1), so dass der frühere Block (j) mit dem Vorteil decodiert werden kann, dass der darauffolgende Block (j + 1) bekannt ist, aber nicht der vorangehende Block (j – 1), und zwar in einem Schritt 118. Wenn andererseits erfasst wird, dass die Variable j zu der Zahl k des ursprünglichen, zuerst erfassten schlechten Blocks im Schritt 116 zurückgekehrt ist, dann tritt eine Verzweigung zu einem Schritt 120 auf, wobei der ursprüngliche schlechte Block (k) mit dem Vorteil decodiert wird, dass sowohl der folgende als auch der vorangehende Block bekannt sind, und somit ist eine vollständige Verstärkung von 3 dB verfügbar, um eine höhere Wahrscheinlichkeit zum richtigen Decodieren für das zweite Mal durch diesen Prozess zu ergeben. Gleichgültig ob die CRC-Prüfung am Block (k) im Schritt **120** eine Prüfung durchführt oder nicht, kann keine weitere Verbesserung erwartet werden, und in den Schritten **124** und **126** springt der Algorithmus nun hinter den höchsten Block voraus, der bis jetzt richtig decodiert ist, und springt zurück zum Schritt 100.

[0120] Gleichermaßen springt in einem Block **128**, wenn ein zuvor schlechter Block, der wiederum mit der Hilfe eines nun bekannten folgenden Blocks decodiert ist, keine weitere Verbesserung zeigt, der Algorithmus zum Block **124** voraus und dann voraus hinter den richtig decodierten Block mit der höchsten Nummer, und springt zurück zum Schritt 100.

[0121] Es wird erkannt werden, dass eine Verwendung einer CRC-Prüfung einfach ein beispielhaftes Kriterium dafür ist, zu bestimmen, ob ein decodierter Block dazu verwendet werden kann, beim Decodieren eines folgenden Blocks oder beim erneuten Block eines vorherigen Blocks zu helfen, oder um zu bestimmen, ob ein zu vor decodierter Block überhaupt erneut decodiert werden muss. Andere Kriterien können jedoch verwendet werden, wie beispielsweise das Pfadmaß-Wachstum über einem Block. Wenn das Pfadmaß-Wachstum, das die Differenz zwischen dem besten Pfadmaß beim Beginn eines Decodierens eines Blocks und dem besten Pfadmaß am Ende eines Decodierens eines Blocks ist, innerhalb einer ersten Schwelle ist, kann bestimmt werden, dass der Block gut genug decodiert worden ist, um derart angesehen zu werden, bekannte Daten zum Helfen beim Decodieren eines weiteren Blocks darzustellen. Wenn das Pfadmaß-Wachstum außerhalb einer zweiten Schwelle ist, kann bestimmt werden, dass es Wert ist, den Block wieder zu decodieren, wenn ein Pfadmaß-Wachstum eines darauffolgenden Blocks innerhalb der ersten Schwelle liegt. Irgendein Kriterium einer Decodierqualitätssicherung kann verwendet werden, wie beispielsweise die oben beschriebenen beispielhaften Verfahren.

[0122] Die Verwendung von zyklischen Redundanzprüfcodes ist Fachleuten auf dem Gebiet adäquat bekannt, ohne sie hier detailliert zu beschreiben. CRC-Prüfungen enthalten allgemein ein Berechnen eines Prüfworts oder eines Syndroms, das aus nur Nullen bestehen sollte, wenn kein Fehler vorhanden ist. Bestimmte Nichtnulll-Syndrome können korrigierbaren Fehlern entsprechen, wie beispielsweise Einzelbitfehlern. Ob eine solche weitere Fehlerkorrekturmöglichkeit von CRC-Codes ausgenutzt wird oder nicht ist optional. Die CRC-Prüfung kann für die Zwecke des obigen Algorithmus als richtig bestimmt sein, wenn entweder das Syndrom Null ist oder wenn es gleich einem Wert entsprechend einem korrigierten Fehlermuster ist, wenn eine

Fehlerkorrektur durch CRC implementiert ist.

[0123] Bei der obigen Beschreibung ist angenommen, dass die Fehlerkorrektur-Decodierung nach jeder Stufe einen Schluss erreicht, bei welcher Stelle eine CRC-Prüfung durchgeführt werden kann. Wenn die Fehlerkorrektur-Codierung/Decodierung eine Blockcodierung ist, oder eine Faltungsdecodierung mit Endbits, oder die Technik eines Abschneidens eines Endes verwendet wird, um einen definierten Decodierschluss für jeden Block zu erreichen, dann kann CRC in der Tat an den Ergebnissen geprüft werden. Andererseits erreicht der Decodierer beim Verwenden einer kontinuierlichen Faltungscodierung, wobei jeder nachfolgende Datenblock dem vorherigen Block sequentiell in den Codierer folgt kein konklusives Ergebnis, wenn das letzte übertragene Bit, das von einem codierten Datenbit abhängt, verarbeitet worden ist. An dieser Stelle enthält ein Faltungscodierer typischerweise eine Anzahl von Kandidatenergebnissequenzen, die das beste Ergebnis für jede mögliche Kombination einer Anzahl von bisher unvollständig decodierten folgenden Bits darstellen. Nur dann, wenn ein weiterer Block, der mehrere Hinweise auf diese folgenden Bits enthält, verarbeitet worden ist, wird der wahrscheinlichste Kandidat offensichtlich werden. Es ist jedoch für den Empfänger unerwünscht, auf weitere Daten warten zu müssen, bevor er eine CRC-Prüfung an dem vorherigen Block verarbeiten kann, da dies eine Ausgabe von dem Demodulator verzögern kann, was eine unerwünschte Sprachverzögerung in beispielsweise einem digitalen Sprachübertragungssystem verursacht. Eine tolerierbare Verzögerung ist ein Faktor beim Beschreiben, eine wie große Tiefe für die retrospektive Decodierung, die in **Fig.** 7 beschrieben ist, angenommen werden kann, weshalb eine Leistungsfähigkeit auch von einem Minimieren unnötiger Verarbeitungsverzögerungen abhängt.

[0124] Andererseits wird eine der verfügbaren Kandidatensequenzen ausgewählt werden, und da diese bezüglich der Anzahl endlich sind und alle innerhalb des Vorgeschichtenspeichers des Viterbi-Decodierers gespeichert sind, kann eine CRC-Prüfung an allen Kandidatensequenzen durchgeführt werden. Dies kann erreicht werden, wie es folgt. Die CRC-Prüfung wird zuerst an dem Viterbi-Zustands-Vorgeschichtenspeicher mit dem besten zugehörigen Pfadmaß durchgeführt. Wenn die CRC prüft, können die Vorgeschichtenspeicherinhalte als das definitive Ergebnis angesehen werden, und sie werden zu einer weiteren Verarbeitung ausgegeben (z. B. einem Sprachdecodieralgorithmus oder einem D/A-Wandler). Dieses Ergebnis kann auch als die bekannte Bitsequenz genommen werden, um beim Decodieren des nächsten Blocks zu helfen.

[0125] Alternativ dazu wird, wenn die CRC nicht prüfte, der Zustand mit dem zweitbesten Maß geprüft, usw., bis ein Zustand, der die CRC durchläuft, gefunden wird. Bei einem kontinuierlichen Faltungsdecodieren resultiert dies in einer einzigen Startstelle, bei welcher Viterbi-Zustände initialisiert werden, bevor ein Verarbeiten des nächsten Blocks beginnt. Das bedeutet, dass der Vorgeschichtenspeicher und die Maße aller Zustände auf den Wert des Kandidaten eingestellt werden, der die CRC-Prüfung durchlief. Wenn kein Kandidat die CRC-Prüfung durchlief, wird der Frame gemäß dem Ablaufdiagramm in **Fig.** 7 als schlecht bezeichnet, und dann werden entweder die Daten von dem Zustand mit dem besten Maß zur weiteren Verarbeitung ausgegeben, oder wird eine Löschanzeige geliefert, wenn es nicht möglich ist, für eine längere Verzögerung zu warten. Im letzteren Fall kann es natürlich sein, dass es nicht wieder eine Stelle in einer retrospektiven Decodierung eines schlechten Frames gibt, da ein retrospektives oder Zweifachdurchlass-Decodieren insbesondere dann nützlich ist, wenn es zulässig ist, zu warten, bis wenigstens einige neue übertragene Daten empfangen werden.

[0126] Jedoch selbst dann, wenn ein retrospektives Decodieren nicht nützlich ist, kann die frühe CRC-Prüfung beim Begrenzen der Startzustände des Decodierers zum Decodieren des nächsten Blocks auf diejenigen sein, die die CRC durchlaufen. Somit wird gemäß einem beispielhaften Ausführungsbeispiel der Erfindung eine CRC-Prüfung bei allen Zuständen durchgeführt, und diejenigen, die nicht durchgehen, werden weggeworfen, bis nicht keiner die CRC-Prüfung durchläuft, in welchem Fall alle als Startzustände für jeweilige Viterbi-Zustände des nächsten Decodierzyklus zurückgehalten werden können. Ein einfaches Verfahren zum Wegwerfen eines Zustands besteht im Einstellen seines Pfadmaßes auf einen "schlechten" Wert, z. B. auf einen sehr hohen Wert, was sicherstellt, dass er nicht als ein "bester" Vorgänger für einen Nachläuferzustand ausgewählt werden wird. Somit werden dann, wenn irgendein Zustand die CRC durchläuft, die Pfadmaße von denjenigen, die die CRC nicht durchlaufen, auf hoch eingestellt. Wenn kein Zustand die CRC durchläuft, dann werden ihre Pfadmaße nicht modifiziert als die Startstellen des nächsten Decodierzyklus akzeptiert.

[0127] Eine weitere Variation modifiziert die Decodierersequenz, so dass nur Zustände, die die CRC-Prüfung durchlaufen, schließlich erzeugt werden können. Wenn die Anzahl von CRC-Prüfbits beispielsweise N ist, dann wird die Decodierersequenz modifiziert, nachdem nur M-N der Ausgangsbits von diesem Block erzeugt worden sind. An dieser Stelle können die linearen Beschränkungen, die die CRC darstellt, zum Berechnen für jeden Zustand von dem verwendet werden, in welchem die übrigen N Bits für die CRC-Prüfung sein müssen. Somit kann dann, wenn das nächste Bit für einen Zustand 00001 sein muss, dieser Zustand kein Vorgänger für einen Zustand 0000 sein, sondern nur für einen neuen Zustand 0001. Wenn der Zustand 1000 auch ein nächstes Bit gleich 1 haben muss, dann kann es entweder kein Vorgänger für den neuen Zustand 0000 sein, sondern nur für den neuen Zustand 0001, so dass es keinen möglichen Vorgänger für den neuen Zustand 0000 gibt, und er nicht erzeugt werden wird. Allgemein ist es nicht möglich zu sagen, wie viele Endzustände, die die CRC

durchlaufen, erzeugt werden, sondern es wird immer wenigstens einen geben. Diejenigen, die nicht erzeugt werden, sind diejenigen, die die CRC nicht durchlaufen würden. Das in diesem Fall erzeugte Ergebnis ist nicht dasselbe, das zulässt, dass der Prozess bis zum Ende eines Blocks fortfährt, und dann die Zustände wegwirft, die die CRC nicht durchlaufen. Dies ist deshalb so, weil bei dem letzteren Verfahren ein unrichtiger Zustand 0000 einen richtigen Zustand 1000 als Vorgänger für einen nachfolgenden Zustand 0000 oder 0001 aufgrund dessen überlebt haben kann, dass der unrichtige Zustand nachteiligerweise ein besseres Maß hat. Bei der Alternative akzeptiert der Decodierer nur das schlechtere Maß, weil ein Akzeptieren des besseren Maßes eine Sequenz erzeugen würde, die die CRC-Prüfung nicht durchlaufen könnte.

[0128] Somit kann das obige beispielhafte Ausführungsbeispiel der Erfindung zum Liefern einer verbesserten Leistungsfähigkeit verwendet werden, wenn ein Fehlerkorrektur-codiertes und differentiell moduliertes Signal decodiert wird. Dieses beispielhafte Ausführungsbeispiel basiert nicht auf einem Einfügen von explizit bekannten Symbolen in das übertragene Signal, mit welchem ein Abschätzen der Phasen- und Amplitudenvariationen des Kanals zu unterstützen sind. Bei dem ersteren Verfahren jedoch, bei welchem explizite bekannte Symbole eingefügt werden, können diese, wie es beschrieben ist, zum Abschätzen der RF-Vektoren auf einer oder einer anderen Seite eines unbekannten Symbols und somit zum Verbessern einer Zuverlässigkeit eines Decodierens von unbekannten Symbolen verwendet werden. Wenn ein benachbartes unbekanntes Symbol decodiert wird, kann es zusammen mit vorherigen bekannten oder decodierten Symbolen zum Abschätzen der Rate verwendet werden, mit welcher der Funkkanal sich bezüglich der Phase und der Amplitude ändert (oder bezüglich eines kartesischen komplexen Werts), und somit zum Helfen bei einem Vorhersagen des Funkkanaleffekts auf die erwarteten Werte einer als nächstes zu decodierenden Abtastung.

[0129] Obwohl oben mehrere beispielhafte Verfahren zum Schließen auf eine Decodierung eines Datensegments und zum Durchführen einer Fehlerprüfung vor einem Fortfahren beschrieben worden sind, können die Verfahren gemäß der vorliegenden Erfindung auch die Fehlerprüfung weglassen und ein Reduzieren des gerade decodierten Blocks zu einem einzelnen Kandidaten vermeiden. Da jeder Zustand des Decodierers nach einem Verarbeiten von beispielsweise allen "x" Bits ein möglicher Kandidat für das "x"-Ergebnis ist, können diese alle zurückgehalten und als "Nach-Überlebender"-Eingaben für ein Fortführen zum Verarbeiten der "a" Bits verwendet werden. Die Verwendung eines CRC ist somit optional.

[0130] Verfahren zum Aktualisieren der Kanalmodelle aus dem empfangenen Signal sind, wie es beispielsweise in der schwedischen Patentanmeldung mit der Nummer 90850301.4 beschrieben ist. Diese Verfahren unterhalten ein separates Kanalmodell für jeden Viterbi-Zustand. Wenn einer einer Anzahl von möglichen Vorgängerzuständen ausgewählt wird, um als der beste Vorgänger für einen neuen Zustand zu überleben, wird das Kanalmodell, das zu dem Überlebenden gehört, aktualisiert, um das Kanalmodell für den neuen Zustand zu werden. Auf diese Weise wird sichergestellt, dass die überlebenden Kanalmodelle immer von den besten demodulierten Datensequenzen, die aktuell sind, abgeleitet wurden.

[0131] Das US-Patent US-A-5 331 666 mit dem Titel "Adaptive Maximum Likelihood Demodulator", das am 08. Juni 1992 vom selben Erfinder und gleichen Zessionär eingereicht ist, beschreibt eine Variation eines adaptiven Viterbi-Entzerrers, der keine Kanalmodelle zum Erzeugen der Vorhersagen verwendet, außer vielleicht für ein anfängliches Hochfahren, und somit wird kein Aktualisieren von Kanalmodellparametern verwendet. Stattdessen verwendet diese Variation ein direktes Aktualisieren der Signalvorhersagen für jeden Zustand, ohne durch den Zwischenschritt von zuerst einem Aktualisieren eines Kanalmodells zu gehen.

[0132] Das gemeinsam zessionierte US-Patent US-A-5 335 250 mit dem Titel "Bidirectional Demodulation Method and Apparatus", eingereicht am 22. Oktober 1992, beschreibt ein Verfahren zum Minimieren eines Verlustes von intervenierenden bzw. dazwischenliegenden Daten, wenn ein temporärer Signalschwund einen Verlust von einem der Übungsmuster verursacht. Intervenierende Daten können nicht nur in Vorwärtsrichtung aus einem Übungsmuster demoduliert werden, sondern auch in Rückwärtsrichtung aus dem nächsten Übungsmuster, indem zuerst empfangene Signalabtastungen in einem Speicher gespeichert werden und dann die Sequenz bezüglich der Zeit umgekehrt wird, und um Qualitätsabschätzungen einer Demodulation in jeder Richtung herzustellen, indem bestimmt wird, wie viele Datensymbole durch eine Vorwärtsdemodulation und wie viele durch eine Rückwärtsdemodulation decodiert werden sollen.

[0133] Es sollte beachtet werden, dass die vorliegende Erfindung sowohl auf eine Vorwärtsdecodulation von bekannten "Übungs"-Symbolen als auch auf eine Rückwärtsdecodulation anwendbar ist, aber die Richtung nicht von der Signalqualität abhängig ist, sondern durch das gewählte Verschachtelungsmuster bestimmt wird, wie es in Bezug auf **Fig.** 1 diskutiert ist, wo die Reihenfolge einer Verarbeitung von Rechts nach Links von U_{13} war, aber von Links nach Rechts von U_{17}, gemäß dem festen Verschachtelungsmuster und nicht variabel in Abhängigkeit von einer Signalqualität. Tatsächlich ist ein Vorteil der vorliegenden Erfindung, dass eine Richtung einer Demodulation nicht mehr ein Punkt ist, da der Demodulationsprozess integriert mit dem Decodieren ist.

[0134] Alle obigen Verfahren können beispielsweise mit einem Faltungscodieren einer konstanten Rate (z. B. der Rate 1/2 oder der Rate 1/3) oder mit einer durchbrochenen Faltungscodierung einer variablen Rate verwendet werden. Im letzteren Fall können einige Datenbits überhaupt nicht codiert werden, wie beispielsweise

dann, wenn ihre Wahrnehmungssignifikanz beim Beeinflussen der Qualität einer digitalen Sprachübertragung nicht hoch ist. Wenn bestimmte Bits nicht durch eine Fehlerkorrekturcodierung geschützt werden, wird ihre Sicherheit durch ein Decodieren nicht verbessert, und somit können sie nicht als bekannte Symbole bei einer späteren Verarbeitung von benachbarten Symbolen angesehen werden. Es ist daher wünschenswert, ein Decodieren eines nachfolgenden Blocks unter Verwendung von nur denjenigen Symbolen zu verbessern, die durch eine Fehlerkorrektur geschützt sind. Dies kann gemäß einem weiteren Aspekt der vorliegenden Erfindung erreicht werden, wobei der differentielle Modulationsprozess zu einem "Überspringen" nicht codierter Symbole modifiziert wird. Dies ist in **Fig.** 8 dargestellt.

[0135] In **Fig.** 8 zählt im Vergleich mit **Fig.** 7 die Verbindung von u4 zu einem Symbol a3 und von z4 zu einem Symbol b3. Stattdessen ist a3 bezüglich der Phasendifferenz zwischen beispielsweise einem Vektor u2 und u5 codiert und ist b3 bezüglich der Differenz zwischen einem Vektor z2 und z5 codiert. Auf diese Weise werden nicht codierte Symbole x4 und a4 beim Decodulieren codierter Symbole a3 bzw. b3 nicht benötigt. Stattdessen wird die Kombination von Vektoren u5 und u6, die mit der Hilfe eines bereits bekannten Symbols x6 hergestellt ist, mit der Kombination von u1 und u2 verglichen, die mit der Hilfe eines bereits bekannten x2 hergestellt ist. Das Symbol a3 wird somit decodiert, ohne dass x4 bekannt sein muss, und x4 kann somit ein nicht codiertes Bit sein, das durch den Prozess "Übersprungen" wird, der x1, x2, x3, x5, x6, etc. decodiert. Später kann eine bessere Abschätzung des nicht codierten x4-Bits erzeugt werden, nachdem wenigstens das Bit a1 bekannt ist. Auf diese Weise wird eine Verstärkung selbst für nicht codierte Bits erzeugt, während die größere Unsicherheit von nicht codierten Bits verhindert wird, die die Fehlerrate von codierten Bits verschlechtern.

[0136] Es ist für a3 auch zulässig, dass es bezüglich der Vektordifferenz zwischen u3 und u5 codiert wird, wobei das weniger sichere u4 übersprungen wird. Dann wird ein bekanntes a1 zum Decodieren von a3 verwendet; dies ist jedoch ein Beispiel einer bereits angegebenen "Verarbeitung pro Überlebendem", wobei eine separate Hypothese des a1-Bits in jedem Viterbi-Zustand verfügbar ist und dann verwendet wird, wenn eine a3-Hypothese für den jeweiligen Zustand verarbeitet wird.

[0137] Beispielsweise soll eine Verarbeitung der Hypothese betrachtet werden, dass ein Zustand 1010 durch ein neues a3 = 0 gefolgt wird. Die vier Bits 1010 stellen für diesen Zustand Hypothesen dar, dass x(n – 1) = 1; x(n) = 0; a1 = 1 und a2 = 0. Somit wird beim Vorhersagen des Werts des komplexen Vektors u5, der empfangen werden sollte, wenn a3 = 0, ein verbesserter u3-Wert unter der Annahme berechnet, dass a1 = 1 so ist, wie es in diesem Zustand gegeben ist, und durch Drehen des zuvor verbesserten u2-Werts in Vorwärtsrichtung über den bekannten Phasenwinkel für a1 = 1, um ihn mit dem empfangenen Wert von u3 auszurichten und zu diesem zu addieren.

[0138] Bei dem obigen Beispiel war das Verschachtelungsmuster so, dass der erforderliche a1-Wert noch nur eine Hypothese war, die separat für jeden Zustand verfügbar ist, wodurch er eine "maximale Wahrscheinlichkeit" oder einen "Viterbi"-Abgriff abstellt, während andere Verschachtelungsmuster ins Auge gefasst werden können, so dass a1 lange Zeit hatte, seit es aus dem Hypothesebereich des Vorgeschichtenspeichers (der Zustandsnummer) heraus und in den "Nach-Überlebender"-Bereich geführt ist, in welchem Fall unter Verwendung von a1 zum Helfen beim Verarbeiten von a3 ein "Nach-Überlebender-Abgriff" dargestellt haben würde. Unter Verwendung der x-Werte, die eine lange Zeit haben, seit sie decodiert worden sind, um beim Decodieren der "a"-Symbole zu helfen, stellt "Entscheidungsrückkopplungsabgriffe" dar.

[0139] Die obigen Decodulationsprozesse können auch zum Kompensieren von zeitverzögerten Echos im Funkausbreitungspfad durch Abschätzen der Amplitude und der Phase von Echos verwendet werden, die um eine oder mehrere Symbolperioden verzögert sind, und durch Verwenden der abgeschätzten Echowerte beim Vorhersagen von komplexen Abtastungen, die für eine gegebene Symbolsequenzhypothese empfangen werden sollten, die in dem Viterbi-Zustands-Pfadspeicher enthalten ist. Die Technik, durch welche Abschätzungen von Echoamplituden und -phasen zum Helfen beim Verbessern einer Vorhersage von empfangenen Werten und somit zum Reduzieren von Fehlern aufgrund von Echos verwendet werden, ist allgemein in Offenbarungen beschrieben worden, die eine Kanalaktualisierung und eine Signalvorhersageaktualisierung beschreiben, auf die oben Bezug genommen ist, und die hier nicht weiter ausgeführt werden müssen.

[0140] Gemäß einem weiteren beispielhaften Ausführungsbeispiel existiert ein Verfahren zum Verbessern der Demodulation von nicht codierten Bits durch Vorsehen eines zweiten Durchlassprozesses nach einem Demodulieren und Decodieren von fehlerkorrekturcodierten Bits. Dieses beispielhafte Ausführungsbeispiel der vorliegenden Erfindung ist sogar dann anwendbar, wenn das Verschachtelungsmuster nicht spezifisch ausgewählt wird, um ein kombiniertes Demodulieren und Decodieren zu ermöglichen bzw. zu erleichtern, sondern hält separate Prozesse für eine Demodulation und ein Decodieren zurück. Beispielsweise kann ein erster Demodulationsprozess zum Demodulieren von Pi/4-DQPSK bei dem Vorhandensein von zeitverzögerten Echos gemäß den Techniken ausgeführt werden, die in den US-Patenten US-A-5 331 666 und US-A-5 335 250 offenbart sind.

[0141] Weiche Entscheidungswerte von einem solchen Entzerrungsdemodulator können dann entschachtelt werden, um Werte zusammenzubauen bzw. zu bilden, die repräsentativ für codierte Bits bei einer Präsentationsreihenfolge zu einem Fehlerkorrekturdecodierer sind, der Fehler bei codierten Bits korrigiert. Der Ent-

schachteler lässt nicht codierte Bits weg oder überspringt diese, die von dem Decodierer nicht benötigt werden.

[0142] Nach einem Decodieren und optionalen CRC-Prüfen werden die nun zuverlässiger bekannten decodierten Bits mit einem Modell des Codierprozesses verwendet, um die codierten Werte zu ergeben, wie sie übertragen sind. Diese codierten Werte wurden mit den nicht codieren Bits zur Übertragung verschachtelt und ursprünglich unter Verwendung des beispielhaften Viterbi-Entzerrers der vorgenannten Offenbarung demoduliert. Gemäß diesem beispielhaften Ausführungsbeispiel wird nun eine zweite Durchlass-Demodulationsoperation unter Verwendung desselben Viterbi-Entzerrers durchgeführt, aber unter Beschränkung der Zustände, durch welche sie gemäß den bereits decodierten Bits läuft. Eine Computersimulation zeigt, dass dies gleiche Leistungsverstärkungen wie eine zuvor beschriebene Decodulationstechnik liefert. Ein nicht codiertes Bit, das durch zwei codierte Bits begrenzt ist, die nun bekannt sind, zieht einen Vorteil aus der Viterbi-Pfadsammelinformation über das nicht codierte Bit, wenn es durch das nun bekannte vorherige Symbol und das nun bekannte folgende Symbol läuft. Je größer die Lauflänge von decodierten Symbolen ist, die ein nicht codiertes Symbol umgeben, umso größer wird die Verstärkung für das nicht codierte Symbol sein.

[0143] Dieses beispielhafte Ausführungsbeispiel der vorliegenden Erfindung kann auch auf eine quaternäre Modulation angewendet werden, bei welcher das Verschachtelungsmuster und die Codieroperationen so sind, dass eines der zwei Bits eines quaternären Symbols bereits während einer Demodulation des anderen decodiert wird, das Gitter dann Beschränkungen auferlegt, die die Anzahl von Pfaden von bereits decodierten Symbolen zu darauffolgenden Symbolen von vier auf zwei mit gleichen Vorteilen reduziert, oder auferlegt werden können, wenn eines der zwei Bits eines Symbols ein codiertes Bit ist und das andere ein nicht codiertes Bit ist.

[0144] Wie bei irgendeinem digitalen Datenempfänger, der ein Codieren und Verschachteln verwendet, ist es nötig, eine Signalabtastung bei dem Empfänger zu synchronisieren, so dass beispielsweise die Signalabtastungen entsprechend geradzahligen und ungeradzahligen übertragenen Symbolen richtig identifiziert werden können. Eine Synchronisation wird durch das Einschließen von bekannten (z. B. Synchronisations-)Symbolen erleichtert, die zwischen Informationssymbole bei einer bekannten Stelle in übertragenen Signalsegmenten vermischt sind. Wenn solche bekannten Symbole differentiell auf den Funkfrequenzträger durch beispielsweise Anwenden einer definierten Phasenänderung über dem Symbolintervall moduliert werden, können diese bekannten Symbole ungeachtet der absoluten Phase des Kanals erfasst werden. In dem Fall, in welchem Informationssymbole durch eine absolute Vektormodulation (z. B. ein kohärentes Phasenumtasten) getragen werden, ist es nötig, Gruppen von wenigstens zwei bekannten Symbolen zu verwenden, um die Möglichkeit zur Verfügung zu stellen, sie unabhängig von einer Kanalphase zu erfassen. Eine Gruppe von N aufeinanderfolgenden kohärent modulierten Synchronisationssymbolen kann dann als Gruppe von N-1 differentiell modulierten Symbolen behandelt werden.

[0145] Wenn Symbolperioden im Vergleich mit einer Zeitdispersion oder Echoverzögerungen lang sind, die auf dem Ausbreitungspfad erfahren werden, variiert die optimale Empfänger-Abtastzeitgabe um nicht mehr als einen Bruchteil einer Symbolperiode von einem Signalsegment zum nächsten, so dass es nur erforderlich ist, eine langzeitige durchschnittliche Abtastzeitgabe zu erhalten, die eine Durchschnittsbildung über viele Signalsegmente ist. Die bekannten Symbole über viele Signalsegmente können somit alle zum Bestimmen einer optimalen Empfängerzeitgabe verwendet werden. Beispielsweise dann, wenn jedes Signalsegment **136** Informationssymbole aufweist, und acht verstreute, isolierte und differentiell modulierte bekannte Symbole, wird ein Bestimmen einer optimalen Zeitgabe über einem sich bewegenden Fenster von 16 solchen Signalsegmenten insgesamt 128 bekannte Symbole verwenden. Das bevorzugte Verfahren zum Bestimmen der optimalen Zeitgabe besteht darin, zuerst eine Anfangszeitgabe bei dem Start einer Kommunikation unter Verwendung von Signalsegmenten zu erhalten, die mit bekannten Symbolen aufgefüllt sind, um eine schnelle Synchronisation zu ermöglichen, dann eine Synchronisation durch Durchführen eines Tests für eine geringe Anzahl von nahen Zeitgabepositionen auf jeder Seite der nominalen Position beizubehalten, die beispielsweise bei 1/8 Symbolperiodenintervallen beabstandet sind. Der Test kann durch Berechnen des differentiellen Symbolwerts Z(i), Z(i – 8)* für die getestete Zeitgabeposition i durch das Signalsegment (unter der Annahme von acht Abtastungen pro Symbol) durchgeführt werden, durch Drehen dieses komplexen Werts über das inverse des Winkels der für das bekannte Symbol erwartet wird, um es mit der realen Achse auszurichten, und durch Akkumulieren der gedrehten Werte in einem binären Histogramm entsprechend i. Die Zeitgabeposition i kann beispielsweise Werte von –8 bis +8 in Bezug auf eine nominale Zeitposition annehmen, d. h. +/– einer Symbolperiode. Das binäre Histogramm, das den größten Wert akkumuliert, zeigt die Abtastzeitgabe i an, die verwendet werden sollte. Die binären Histogramme können leckig gemacht werden, um den Einfluss von alten empfangenen Signalsegmenten exponentiell zu vergessen, um eine Anpassung der Zeitgabe an Änderungen aufgrund einer relativen Bewegung des Senders und des Empfängers zuzulassen, oder aufgrund von relativen Referenztaktfehlern.

[0146] Es ist mit einer zusätzlichen Signalverarbeitungsanstrengung auch möglich, eine Synchronisation zu prüfen und beizubehalten, ohne die bekannten Symbole zu verwenden, indem herkömmliche Taktbildungstechniken verwendet werden. Diese Technik kann mit einer Einrichtung zum erneuten Erlangen einer Framesynchronisation kombiniert werden, wenn ein Decodieren damit beginnt, häufig fehlzuschlagen. Beispielswei-

se kann ein Decodieren für mehr als eine postulierte Zeitgabeposition versucht werden, und wenn eine zuvor bevorzugte Zeitgabeposition damit beginnt, häufige Decodierfehler zu ergeben, während eine benachbarte Zeitgabeposition ein erfolgreiches Decodieren ergibt, dann wird die benachbarte Zeitgabeposition die bevorzugte Zeitgabeposition und wird die Abtastzeitgabe neu um die neue bevorzugte Zeitgabe zentralisiert.

[0147] Es wird aus dem obigen gesehen, dass der Empfänger vorzugsweise Abtastungen bei einem Bruchteil der Symbolbeabstandung nimmt, wie beispielsweise bei acht Abtastungen pro Symbolperiode, und dass ein Synchronisationsalgorithmus bestimmt, welche der acht Abtastphasen beim Auswählen einer Abtastung pro Symbol für eine Decodierung zu verwenden ist.

[0148] Zwei andere Variationen können angegeben werden. Es ist möglich, mehr als eine Abtastung pro Symbol bei dem kombinierten Demodulations- und Decodierprozess zu verwenden, indem der Algorithmus für eine maximale Wahrscheinlichkeit auf ein Verwenden des Vorhersagemechanismus zum Vorhersagen von nicht nur einer empfangenen Wellenformabtastung pro Symbol für eine hypothetisierte Symbolsequenz erweitert wird, sondern von zwei oder mehreren Abtastungen pro Symbol der erwarteten Wellenform. Die Maßberechnungseinrichtung berechnet dann eine Fehlanpassung zwischen allen vorhergesagten und beobachteten Wellenformabtastungen und akkumuliert sie. Es ist klar, dass dann, wenn mehrere Abtastungen pro Symbol im Decodulationsprozess enthalten sind, eine Leistungsfähigkeit nicht empfindlich gegenüber einer genauen Abtastzeitgabe sein wird.

[0149] Eine zweite Variation, die unempfindlich gegenüber einer Abtastzeitgabe ist, verwendet nur eine Abtastung pro Symbol. Wenn die übertragene Modulation eine konstante Hüllkurvenmodulation ist, wie beispielsweise GMSK oder 4-CPFSK, dann wird die Energie der empfangenen Abtastungen nicht von der ausgewählten Zeitgabe abhängen. Jedoch hängt die Abhängigkeit der Abtastung von dem vorherigen und von dem kommenden Symbol von der Zeitgabe ab. Die empfangene Wellenform kann entweder bei Positionen abgetastet werden, bei welchen die Abtastwerte primär von einzelnen Symbolen abhängen werden, d. h. in der Mitte von Symbolen, oder bei einem andern Extrem kann sie bei Positionen abgetastet werden, bei welchen die Abtastungen gleichermaßen von zwei benachbarten Symbolen abhängen werden, d. h. innerhalb zwischen den zwei benachbarten Symbolen. Die Leistungsfähigkeit eines Algorithmus für eine maximale Wahrscheinlichkeit, der empfangene Abtastungen basierend auf allen Symbolen vorhersagt, von welchen sie abhängen, führt näherungsweise gleich gut bei allen Abtastzeitgaben aus, selbst wenn nur eine Abtastung pro Symbol verwendet wird. Die Symbole, von welchen eine Abtastung primär abhängt, sollten jedoch vorzugsweise eher ML-Abgriffe als DFE- oder Nach-Überlebender-Abgriffe darstellen, um sicherzustellen, dass alle möglichen Kombinationen der in Frage stehenden Symbole getestet werden. Dies kann durch eine geeignete Auswahl eines Codier- und Verschachtelungsmusters gemäß den folgenden Richtlinien sichergestellt werden.

[0150] Ein Codierer mit der Rate 1/n wird n Ausgangssymbole für jedes Eingangssymbol erzeugen. Gegensätzlich dazu wird ein Decodierer mit der Rate 1/n n empfangene Symbole verarbeiten, um ein decodiertes Symbol auszugeben. Die n empfangenen Symbole werden vorzugsweise bezüglich der Zeit über mehrere übertragene Segmente verteilt, um den Effekt eines Schwunds bzw. Fadings zu reduzieren. Bei den vorangehenden beispielhaften Ausführungsbeispielen ist beispielsweise vorgeschlagen worden, dass aufeinanderfolgende Symbole vom Codierer zuerst am nächsten zu bekannten Synchronisationssymbolen platziert werden, und dann am nächsten zu den ersten platzierten Symbolen, usw. Wenn n Positionen, die bekannten Synchronisationssymbolen am nächsten sind, verfügbar sind, dann werden die n codierten Symbole für jedes nicht codierte Symbol diese Positionen ausschöpfen, und die nächsten (n + 1)-ten codierten Symbole werden benachbart bzw. angrenzend zu dem ersten codierten Symbol platziert werden. Jedoch deshalb, weil der Decodierer nur ein decodiertes Symbol in seinem Pfad-Vorgeschichtenspeicher für jedes codierte Symbol ausgibt, das verarbeitet wird, hängen codierte Symbole 1 und (n + 1) von Symbolen ab, die im Pfad-Vorgeschichtenspeicher um ein Symbol weiter weg sind. Wenn die Codebeschränkungslänge m Symbole ist, dann sind die entfernten decodierten Symbole m – 1 noch innerhalb des ML-Abgriffs, was garantiert, dass alle Kombinationen von diesen zwei Symbolen getestet werden. Der Verschachteler kann somit n(m – 1) codierte Symbole in zeitlich verstreuten Positionen platzieren, bevor er die nächsten n am nächsten zu den ersten n platziert, während zwei Symbole in benachbarten Positionen in übertragenen Signalsegmenten innerhalb des Bereichs von Abgriffen für eine maximale Wahrscheinlichkeit gehalten werden, um dadurch sicherzustellen, dass alle Kombinationen der zwei Symbole hypothetisiert werden. Dies stellt wiederum sicher, dass alle möglichen Signalabtastwerte, die von den zwei benachbarten Symbolen abhängen, vorhergesagt und mit empfangenen Abtastwerten verglichen werden, was eine Leistungsfähigkeit ergibt, die dann nicht empfindlich demgegenüber ist, ob eine Abtastzeitgabe in der Mitte eines Symbols oder zwischen Symbolen liegt. Auf diese Weise ist es auch möglich, mit Kanalungenauigkeiten fertig zu werden, die verursachen, dass empfangene Abtastungen von mehr als einem benachbart übertragenen Symbol abhängen, d. h. Echos, die um bis zu einer Symbolperiode verzögert sind. Ein solches Übertragungsschema kann somit auf vorteilhafte Weise über Mobilfunkkanäle verwendet werden, die an einer Mehrwegeausbreitung leiden.

Patentansprüche

1. Verfahren zum Übertragen und Empfangen digitaler Daten, enthaltend die Schritte:
Fehlerkorrektur-Codierung beliebiger Informationssymbole (**21**), Verschachteln (**22**) der codierten, beliebigen Informationssymbole mit Symbolen, die einem Empfänger bekannt sind, und Übertragen der verschachtelten Symbole;
Empfangen der übertragenen Symbole;
Fehlerkorrektur-Decodierung der empfangenen Symbole entsprechend den codierten, beliebigen Informationssymbolen (**100**), die an die bekannten Symbole angrenzen;
Fehlerkorrektur-Decodierung der codierten, beliebigen Informationssymbole (**100**), die an die ersten decodierten Symbole angrenzen, unter Verwendung der ersten decodierten Symbole und so weiter, unter Wegbewegung in jeder Richtung von den bekannten Symbolen.

2. Verfahren nach Anspruch 1, ferner gekennzeichnet durch den Schritt zum Modellieren der beliebigen Informationssignale und der bekannten Symbole auf ein Funkfrequenzsignal.

3. Verfahren nach Anspruch 2, dadurch gekennzeichnet, dass die Schritte zum Modellieren den weiteren Schritt wie folgt enthalten:
Darstellen der Information und bekannten Symbole durch Vektoränderungen zwischen eine Symbolperiode beabstandeten Signalabtastwerten.

4. Verfahren nach Anspruch 3, ferner gekennzeichnet durch den Schritt:
Platzieren eines codierten Bits bei einer von zwei Bitpositionen der quasi ternären Symbole in einem ersten übertragenen Signalsegment und eines anderen codierten Bits bei der anderen Bitposition der quasi ternären Symbole in einem nachfolgend übertragenen Signalsegment.

5. Verfahren nach Anspruch 4, dadurch gekennzeichnet, dass der Schritt zum Modellieren unter Verwendung einer I, Q gefilterten Pi/4-DQPSK Technik ausgeführt wird.

6. Verfahren nach Anspruch 4, dadurch gekennzeichnet, dass der Schritt zum Modellieren unter Verwendung einer fortlaufenden Phasenmodulationstechnik unter Erzeugung näherungsweise konstanter Amplitudensignale ausgeführt wird.

7. Verfahren nach Anspruch 1, dadurch gekennzeichnet, dass die Codierung eine Blockfehlerkorrektur-Codierung ist.

8. Verfahren nach Anspruch 1, dadurch gekennzeichnet, dass die Codierung eine Faltungscodierung ist.

9. Verfahren nach Anspruch 1, dadurch gekennzeichnet, dass die Codierung eine punktierte Faltungscodierung ist.

10. Verfahren nach Anspruch 3, dadurch gekennzeichnet, dass der Schritt zum Modulieren ferner den folgenden Schritt enthält:
Verwenden einer M-wertigen Symbolmodulation, bei der die Vektoränderungen irgendeine aus einer Zahl von M vordefinierten Änderungen sein können, gleich einer ganzzahligen Potenz N von zwei.

11. Verfahren nach Anspruch 10, dadurch gekennzeichnet, dass die Vektoränderungen jeweils eine Zahl von Information oder bekannter Bits gleich der ganzzahligen Zahl N codieren.

12. Verfahren nach Anspruch 11, dadurch gekennzeichnet, dass es ferner folgende Schritte enthält:
Platzieren codierter Informationsbits zunächst bei einem ersten der ganzzahligen Zahl N von Bits, befördert durch jedes der M-wertigen Symbole, als ein erstes Signalsegment; und
nachfolgendes Platzieren codierter Informationsbits bei einer zweiten der ganzzahligen Zahl N von Bitpositionen der M-wertigen Symbole, geführt durch ein zweites übertragenes Signalsegment.

13. Verfahren nach Anspruch 12, dadurch gekennzeichnet, dass es ferner der Schritt enthält:
Fortführen der Platzierung codierter Informationsbits bei Bitpositionen der M-wertigen Symbole gemäß dem Signalsegment, das die M-wertigen Symbole enthält, bis eine Zahl von Signalsegmenten gleich der ganzen Zahl N verwendet wurde.

14. Verfahren nach Anspruch 1, dadurch gekennzeichnet, dass es den Schritt enthält:
Verschachteln nicht codierter Symbole mit den beliebigen Informationssymbolen und den bekannten Symbolen.

15. Verfahren nach Anspruch 14, dadurch gekennzeichnet, dass es ferner folgende Schritte enthält:
bei dem Empfänger, Überspringen nicht codierter Symbole während der Decodierung der codierten Symbole; und
dann Rückkehr zu der Schätzung der nicht codierten Symbole unter Verwendung bereits decodierter Symbole.

16. Gerät zum Übertragen und Empfangen digitaler Informationssymbole, enthaltend:
eine Übertragungsvorrichtung (**60**) für eine Fehlerkorrekturcodierung zumindest einiger der digitalen Informationssymbole für die Übertragung, Verschachteln (**22**) der codierten Symbole mit nicht codierten Symbolen und Modulieren (**23**) der verschachtelten Symbole auf ein Trägersignal;
eine Vorrichtung zum Empfangen des modulierten Trägersignals, zum Filtern (**40**), Verstärken (**41**), Abtasten und Umsetzen der Signalabtastwerte in eine geeignete Form für die Verarbeitung; und
eine Vorrichtung zum Demodulieren (**53**) von Signalabtastwerten gemäß den codierten, verschachtelten und modulierten Symbolen und zum Fehlerkorrektur-Decodieren der demodulierten Abtastwerte zum Erhalten decodierter Symbole; und
eine Vorrichtung zum Demodulieren (**53**) von Abtastwerten gemäß den nicht codierten, modulierten Symbolen unter Verwendung der bereits decodierten Symbole angrenzend zu den nicht codierten modulierten Symbolen zum Reproduzieren der nicht codierten Symbole mit einer reduzierten Fehlerwahrscheinlichkeit.

17. Gerät nach Anspruch 16, dadurch gekennzeichnet, dass die Verarbeitungsvorrichtung zum Demodulieren von Signalabtastwerten gemäß zumindest den codierten, verschachtelten und modulierten Symbolen auch die Demodulation und die Decodierung in einem Decodulationsprozess kombiniert, der die Signalabtastwerte zum Erzeugen decodierter Symbole decoduliert.

Es folgen 8 Blatt Zeichnungen

Anhängende Zeichnungen

TDMA-BURST 1	. .	U	U	U	U	U	K	U	K	U	K	U	K	U	U	U	U	U
DEMODULATIONSREIHENFOLGE					21	13		1		9		5		17	25	33	41	49
TDMA-BURST 2	. .	U	U	U	U	U	K	U	K	U	K	U	K	U	U	U	U	U
DEMODULATIONSREIHENFOLGE		46	38	30	22	14		2		10		6		18	26	34	42	50
TDMA-BURST 3	. .	U	U	U	U	U	K	U	K	U	K	U	K	U	U	U	U	U
DEMODULATIONSREIHENFOLGE		47	39	31	23	15		3		11		7		19	27	35	43	51
TDMA-BURST 4	. .	U	U	U	U	U	K	U	K	U	K	U	K	U	U	U	U	U
DEMODULATIONSREIHENFOLGE		48	40	32	24	16		4		12		8		20	28	36	44	52

Fig. 1

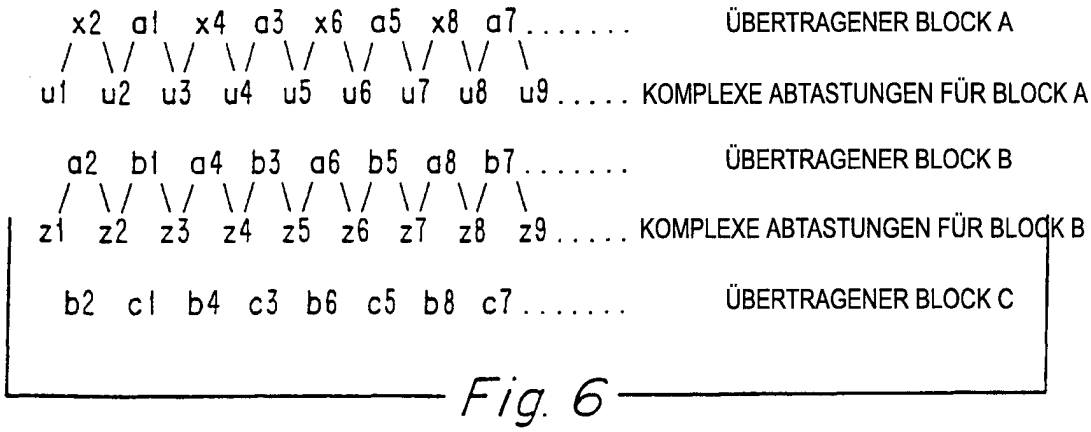


Fig. 6

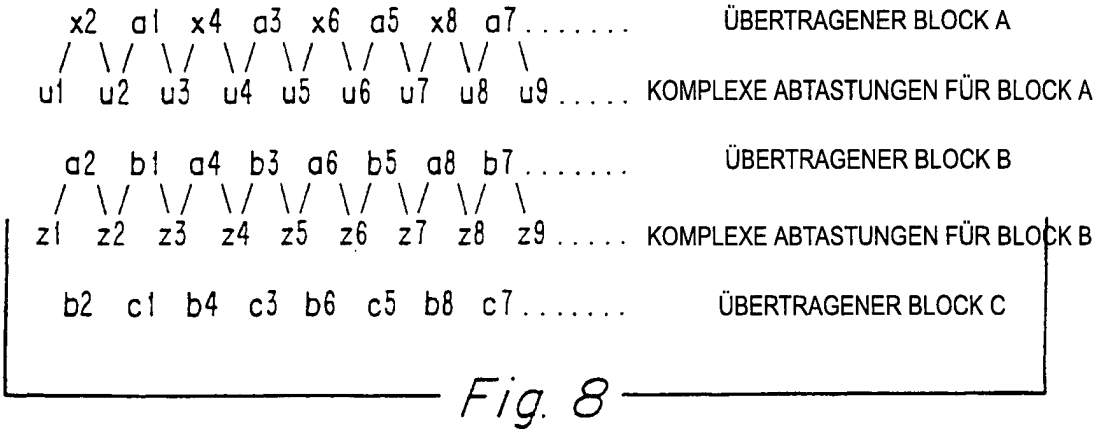


Fig. 8

Fig. 2(a)

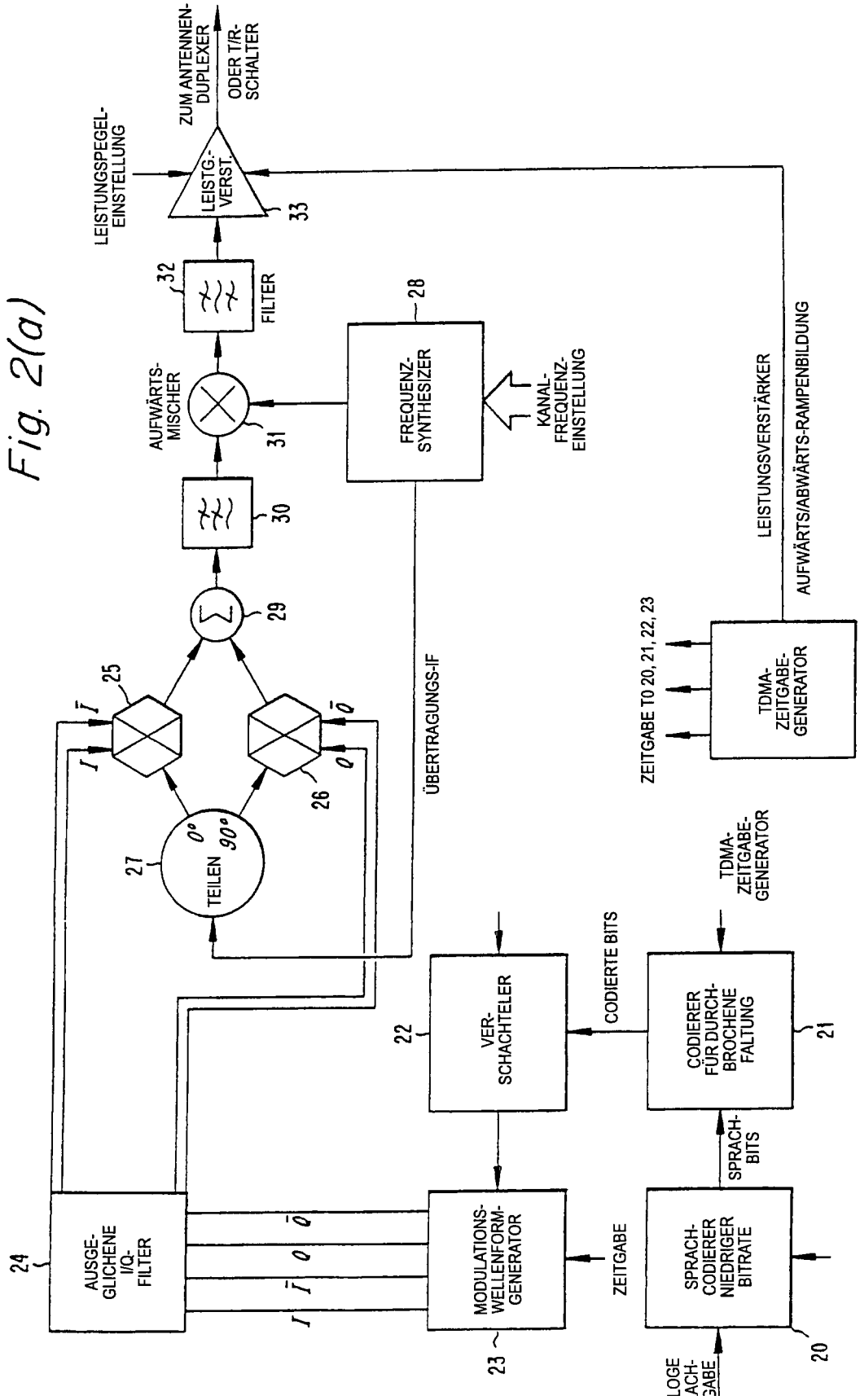

ZUM ANTENNEN-DUPLEXER ODER T/R-SCHALTER
LEISTUNGSPEGEL-EINSTELLUNG
LEISTG.-VERST.
33
32
FILTER
AUFWÄRTS-MISCHER
31
28
FREQUENZ-SYNTHESIZER
KANAL-FREQUENZ-EINSTELLUNG
30
29
25
26
27
TEILEN
0°
90°
ÜBERTRAGUNGS-IF
LEISTUNGSVERSTÄRKER
AUFWÄRTS/ABWÄRTS-RAMPENBILDUNG
ZEITGABE T0 20, 21, 22, 23
TDMA-ZEITGABE-GENERATOR
TDMA-ZEITGABE-GENERATOR
CODIERTE BITS
22
VER-SCHACHTELER
CODIERER FÜR DURCH-BROCHENE FALTUNG
21
SPRACH-BITS
24
AUSGE-GLICHENE I/Q-FILTER
MODULATIONS-WELLENFORM-GENERATOR
ZEITGABE
23
SPRACH-CODIERER NIEDRIGER BITRATE
20
LOGE ACH-GABE

Fig. 2(b)

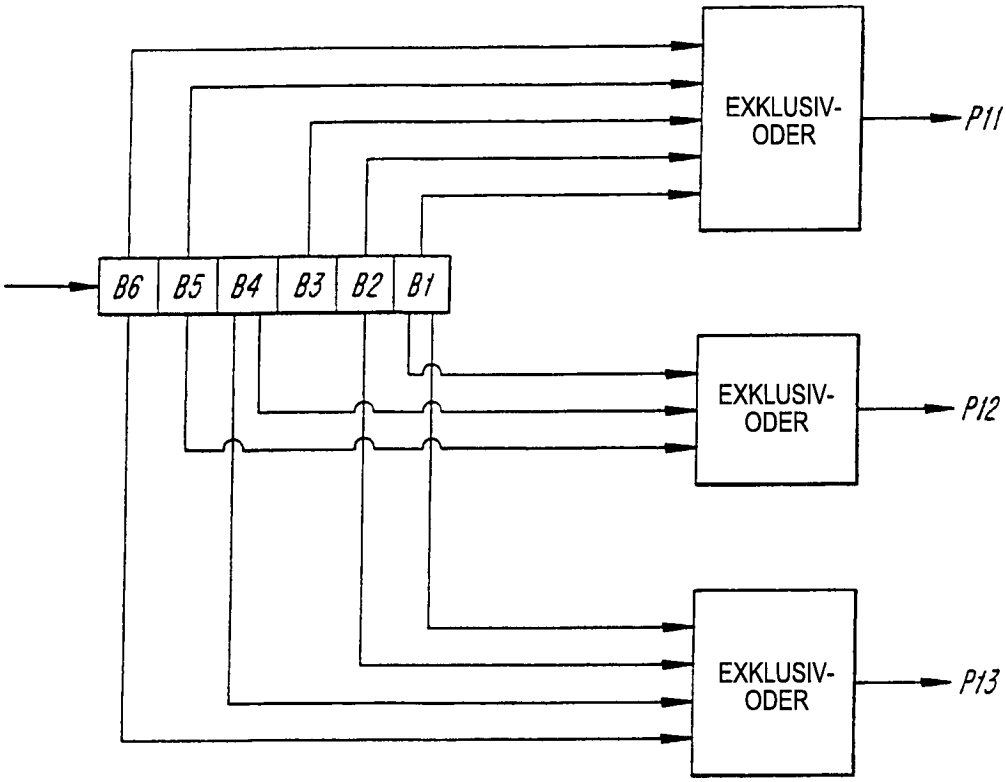

EXKLUSIV-
ODER
P11
B6
B5
B4
B3
B2
B1
EXKLUSIV-
ODER
P12
EXKLUSIV-
ODER
P13

Fig. 3

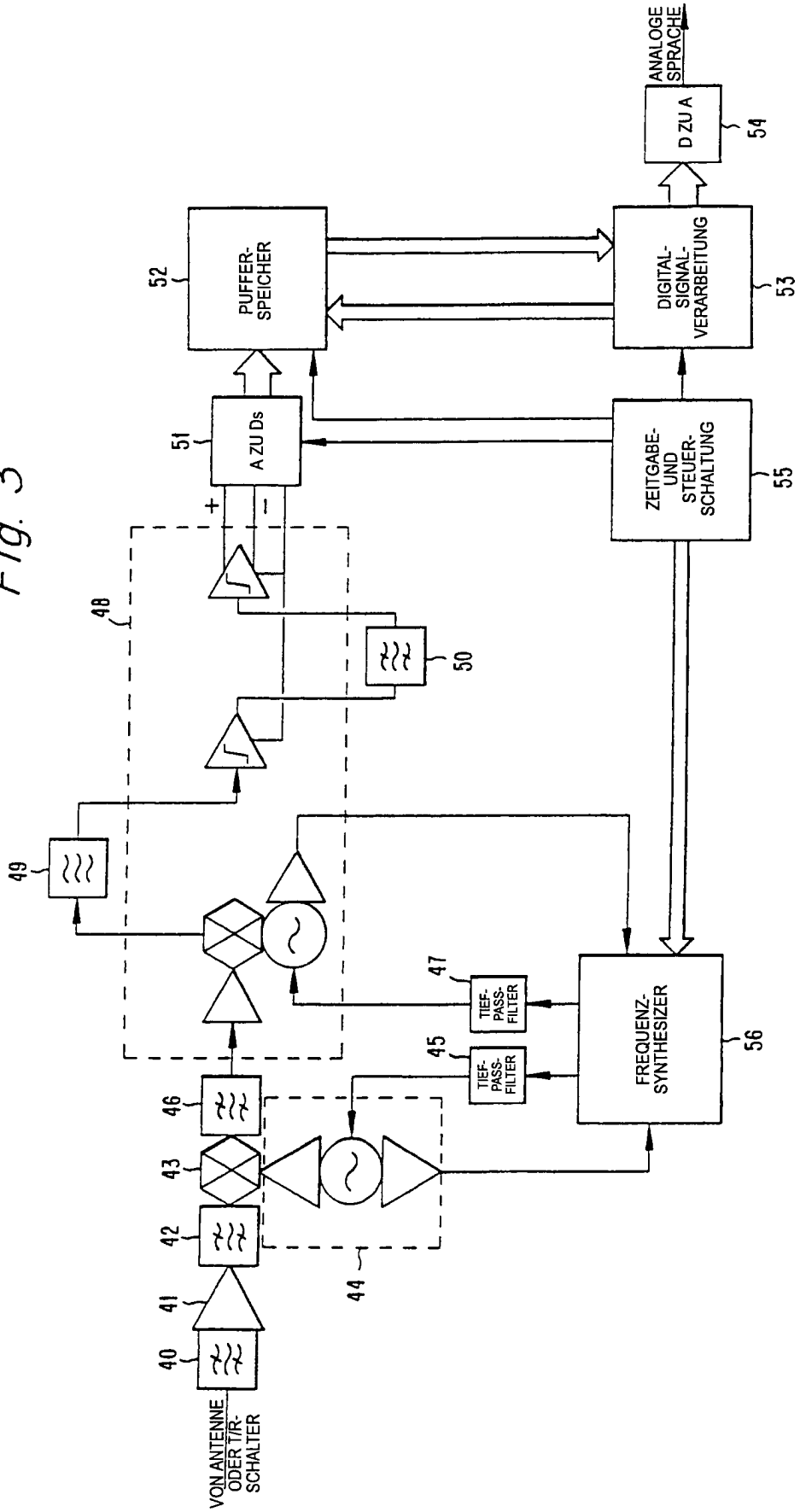

ANALOGE SPRACHE
D ZU A
54
PUFFER-SPEICHER
52
DIGITAL-SIGNAL-VERARBEITUNG
53
A ZU Ds
51
ZEITGABE-UND STEUER-SCHALTUNG
55
48
50
49
47
TIEF-PASS-FILTER
45
TIEF-PASS-FILTER
FREQUENZ-SYNTHESIZER
56
46
43
42
44
41
40
VON ANTENNE ODER T/R-SCHALTER

Fig. 4

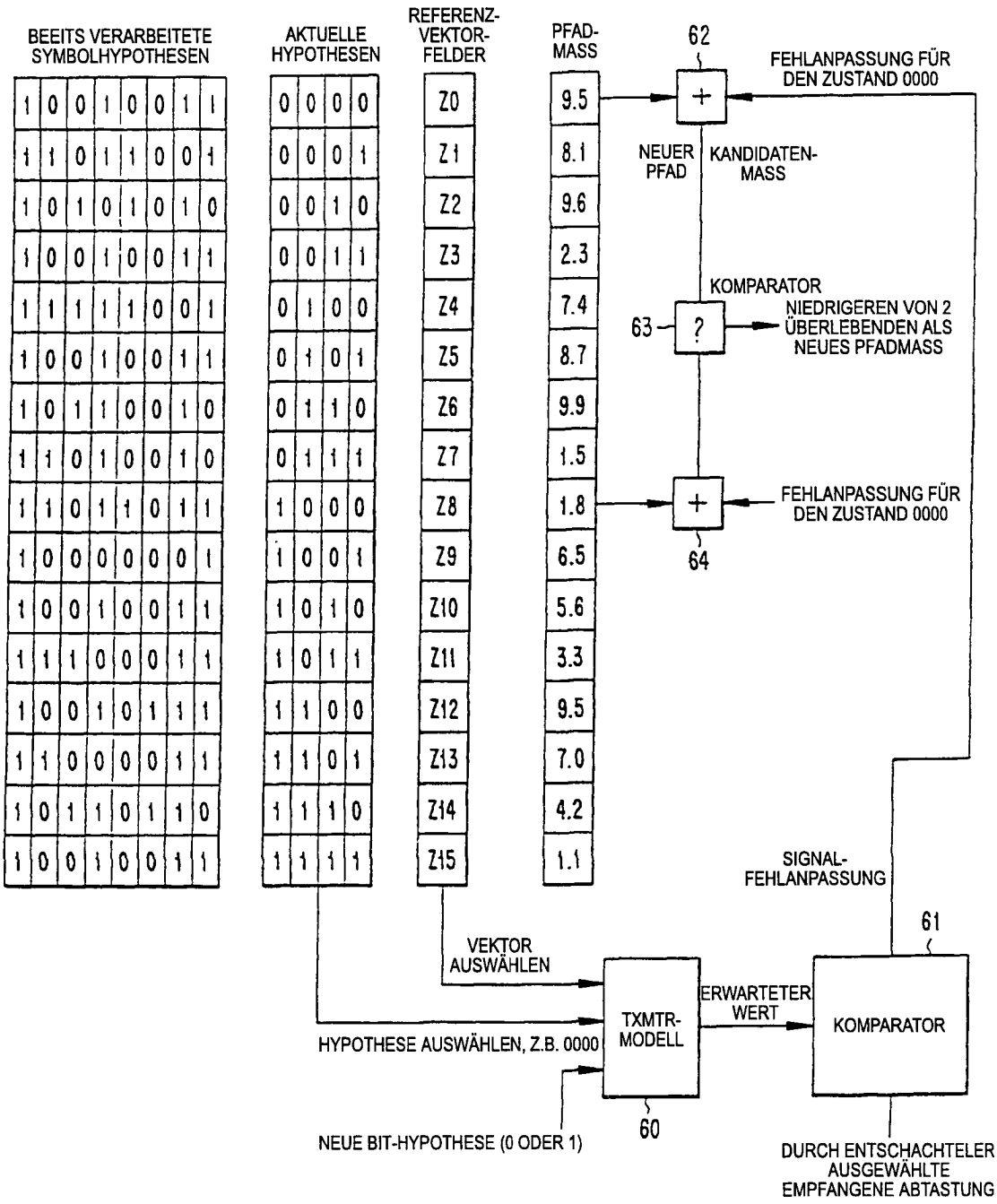
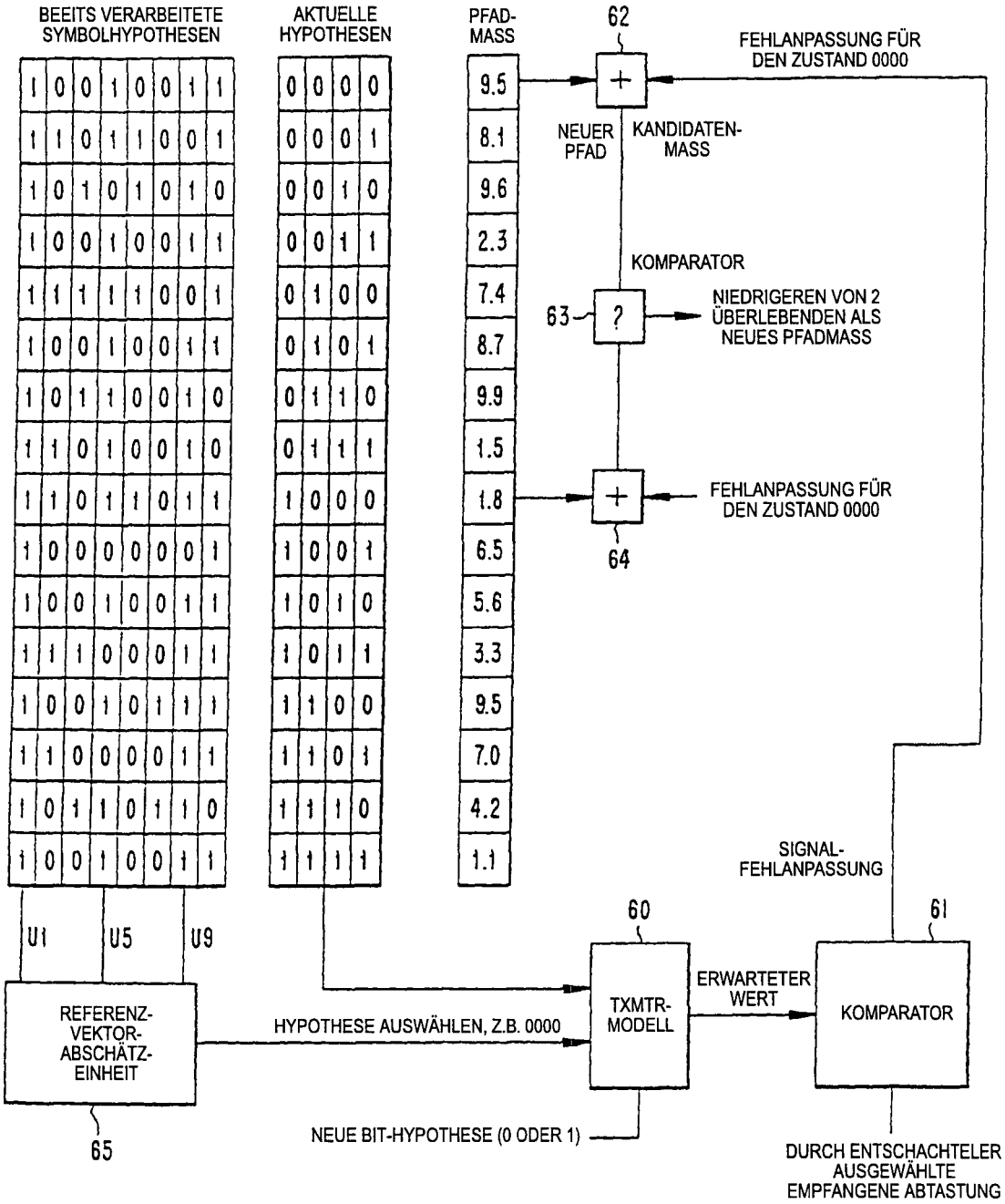

BEEITS VERARBEITETE SYMBOLHYPOTHESEN
AKTUELLE HYPOTHESEN
REFERENZ-VEKTOR-FELDER
PFAD-MASS
62
FEHLANPASSUNG FÜR DEN ZUSTAND 0000
NEUER PFAD
KANDIDATEN-MASS
KOMPARATOR
63
?
NIEDRIGEREN VON 2 ÜBERLEBENDEN ALS NEUES PFADMASS
FEHLANPASSUNG FÜR DEN ZUSTAND 0000
64
SIGNAL-FEHLANPASSUNG
61
VEKTOR AUSWÄHLEN
HYPOTHESE AUSWÄHLEN, Z.B. 0000
TXMTR-MODELL
ERWARTETER WERT
KOMPARATOR
NEUE BIT-HYPOTHESE (0 ODER 1)
60
DURCH ENTSCHACHTELER AUSGEWÄHLTE EMPFANGENE ABTASTUNG

Fig. 5


BEEITS VERARBEITETE SYMBOLHYPOTHESEN
AKTUELLE HYPOTHESEN
PFAD-MASS
62
FEHLANPASSUNG FÜR DEN ZUSTAND 0000
NEUER PFAD
KANDIDATEN-MASS
KOMPARATOR
63
?
NIEDRIGEREN VON 2 ÜBERLEBENDEN ALS NEUES PFADMASS
FEHLANPASSUNG FÜR DEN ZUSTAND 0000
64
SIGNAL-FEHLANPASSUNG
60
61
U1
U5
U9
REFERENZ-VEKTOR-ABSCHÄTZ-EINHEIT
65
HYPOTHESE AUSWÄHLEN, Z.B. 0000
TXMTR-MODELL
ERWARTETER WERT
KOMPARATOR
NEUE BIT-HYPOTHESE (0 ODER 1)
DURCH ENTSCHACHTELER AUSGEWÄHLTE EMPFANGENE ABTASTUNG

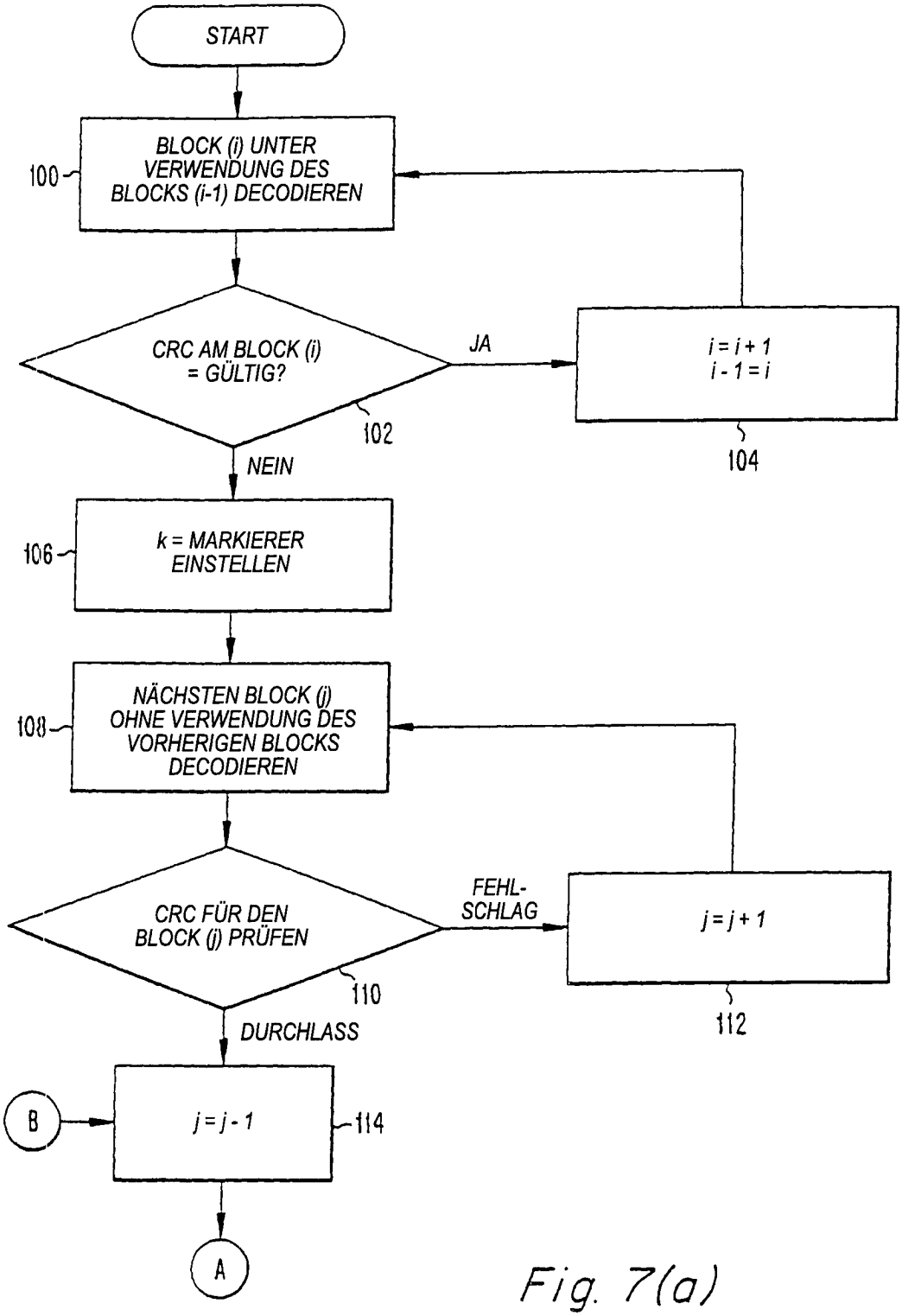

START
BLOCK (i) UNTER VERWENDUNG DES BLOCKS (i-1) DECODIEREN
100
CRC AM BLOCK (i) = GÜLTIG?
JA
i = i + 1
i - 1 = i
102
104
NEIN
k = MARKIERER EINSTELLEN
106
NÄCHSTEN BLOCK (j) OHNE VERWENDUNG DES VORHERIGEN BLOCKS DECODIEREN
108
CRC FÜR DEN BLOCK (j) PRÜFEN
FEHL-SCHLAG
j = j + 1
110
112
DURCHLASS
B
j = j - 1
114
A

Fig. 7(a)

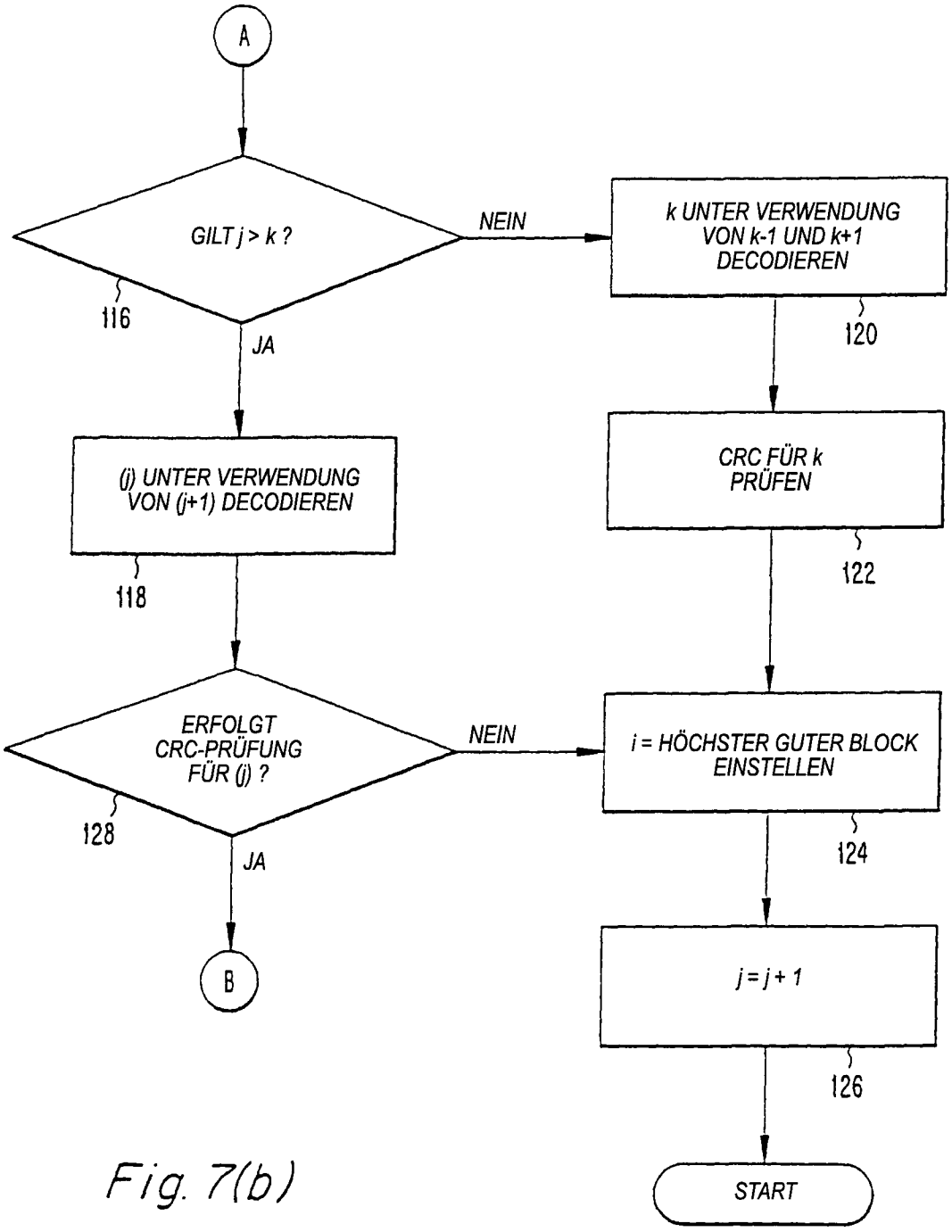

A
GILT j > k ?
NEIN
k UNTER VERWENDUNG VON k-1 UND k+1 DECODIEREN
116
120
JA
(j) UNTER VERWENDUNG VON (j+1) DECODIEREN
CRC FÜR k PRÜFEN
118
122
ERFOLGT CRC-PRÜFUNG FÜR (j) ?
NEIN
i = HÖCHSTER GUTER BLOCK EINSTELLEN
128
124
JA
B
j = j + 1
126
START

Fig. 7(b)