

Der Viterbi- Dekoder

Table of Contents

1. Vorwort.....	3
2. Arten der Kodierung.....	3
2.1 Blockkodierungen.....	3
2.2 Faltungskodierungen.....	4
2.3.1 Historisches.....	4
3. Struktur von Faltungskodierungen.....	5
4. Der Viterbi-Dekoder.....	6
4.2 Beschreibung des Registerinhalts.....	6
4.3 Die Dekodierung.....	8
4.3.1 Die Dekodierung im fehlerfreien Fall.....	8
4.3.2 Die Dekodierung bei Übertragungsfehlern.....	11
4.3.3 Das Dekodierungsergebnis.....	12
5. Die Schwächen des Viterbi Dekoders.....	13
5.1. Die Durchlaufzeit / Korrekturqualität.....	13
5.2 Das Verhalten bei Bündelfehlern.....	13
5.3 Der Rechenaufwand.....	14
6. Mögliche Strukturen der Faltungs-Enkoder.....	14
7. Quellenverzeichnis.....	15
.....	15

1. Vorwort

Im Vortrag über „Digitale Betriebsarten im Amateurfunk – wie die Bits laufen“ wurde der Viterbi-Dekoder schon einmal kurz beschrieben – allerdings zu kurz um verständlich erklären zu können wie dieser Dekoder funktioniert. Deshalb soll der Dekoder in diesem Vortrag genauer beleuchtet werden.

2. Arten der Kodierung

Um per Funk Daten sicher übertragen zu können ist eine spezielle Kodierung dieser Nutzdaten nötig die es auf der Empfangsseite erlaubt Übertragungsfehler zu erkennen und wenn möglich zu korrigieren. Gibt es nur einen Sender und gleichzeitig mehrere Empfänger, scheiden Verfahren aus bei denen der Empfänger bei erkannten Fehlern beim Sender eine Wiederholung anfordert. Die Informationen zur eventuell nötigen Fehlerkorrektur müssen statt dessen in der gesendeten Information mit enthalten sein. Kodierungen die die Informationen zur Fehlererkennung und Fehlerkorrektur mitliefern werden FEC Kodierungen (forward error correction)-Kodierungen genannt.

FEC-Kodierungen lassen sich wiederum in zwei Klassen aufteilen:

- die Blockkodierungen und
- die Faltungskodierungen.

2.1 Blockkodierungen

Bei Blockkodierungen werden die Informationen (wie aus dem Namen zu erkennen) in zwei Blöcken übertragen, einem Block mit den Nutzdaten und einem mit einer Sicherungsinformation. Mit Hilfe dieser Sicherungsinformation soll es dann möglich sein, Fehler bei der Übertragung der Nutzdaten zu erkennen und zu korrigieren. Im einfachsten Fall kann die Sicherungsinformation aus einem sogenannten Paritybit bestehen. Dieses Bit wird z.B. so gesetzt, dass die Anzahl der binären Einsen geradzahlig wird (gerade Parity). Tritt bei einer Übertragung ein Fehler auf (aus einer Null wird eine Eins oder umgekehrt), so ist die Gesamtzahl von Einsen auf der Empfängerseite ungerade – ein Indikator dass in der Übertragung mindestens ein Fehler aufgetreten sein muss. Das hier beschriebene einfache Verfahren lässt sich allerdings nur zur Fehlererkennung verwenden, eine Fehlerkorrektur ist damit nicht möglich. Für eine mögliche Fehlerkorrektur muss die Sicherungsinformation weit aufwendiger sein. Blockkodierungen werden im Amateurfunk bei Übertragungsverfahren wie JT65 verwendet. Bei den Fehlerkorrekturen muss außerdem noch bedacht werden, dass die Übertragungsfehler ja nicht nur die Nutzdaten sondern auch die Sicherungsblöcke betreffen können und dass auch in solchen Fällen eine Korrektur möglich sein muss.

Der Viterbi Dekoder

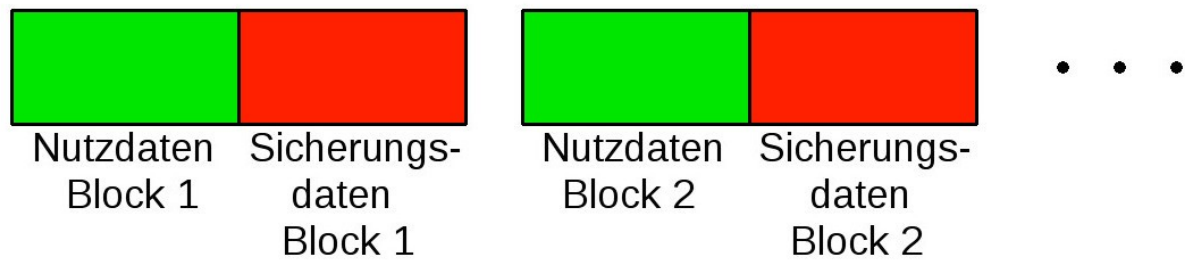


Bild1: Struktur eines Blockcodes

2.2 Faltungskodierungen

Bei Faltungskodierungen gibt es keine Aufteilung in Nutz- und Sicherungsinformation. Die Sicherungsinformation ist in den übertragenen Daten mit enthalten. In den meisten Fällen taucht die Nutzinformation im übertragenen Datenstrom deshalb nicht mehr direkt auf sondern wird erst nach der Dekodierung sichtbar. Ebenso wie es eine große Zahl möglicher Blockkodierungen gibt, existiert eine große Zahl möglicher Faltungskodierungen. In diesem Vortrag möchte ich mich aber auf die im Amateurfunk verwendeten Kodierungen beschränken.

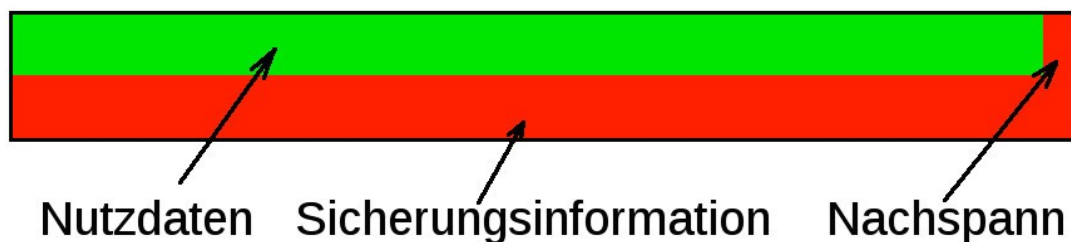


Bild 2: Struktur eines Faltungscodes.

Der im Bild zu sehende Nachspann ist erforderlich, dass auch die letzten Nutzdatenbits dieselbe Korrekturqualitate haben wie die anderen.

2.3.1 Historisches

Die erste Faltungskodierung wurde bereits 1955 von Peter Elias beschrieben. Ein Verfahren zur effektiven Dekodierung solcher Codes wurde aber erst 1967 also 12 Jahre spater von Andrew Viterbi beschrieben. Eine der bekanntesten Anwendungen der Faltungskodierung ist die im Voyager-Projekt. Im Jahr 1977 wurden zwei Sonden zu den Planeten unseres Sonnensystems geschickt. Mit wachsender Entfernung wurden naturlich auch die Signalpegel schwacher und schwacher. Um trotzdem Bildubertragung zu ermoglichen wurden die Sonden so programmiert dass die Daten mit einer Faltungskodierung versehen gesendet wurden. Diese Kodierung hat sich als so effizient erwiesen dass sogar Bilder uber eine Entfernung empfangen und dekodiert werden konnten die auerhalb unseres Sonnensystems aufgenommen wurden. (mehr als 20 Milliarden km Entfernung).

3. Struktur von Faltungskodierungen

Der zentrale Teil eines Faltungsenkoders ist ein Schieberegister. Ein solches Schieberegister besteht aus einer Reihe von Zellen durch die die zu kodierende Nutzinformation von links nach rechts durchgeschoben wird. Die Information rückt also bei jedem Taktschritt um eine Schieberegisterzelle nach rechts. Die Inhalte jeder Zelle können zu jedem Zeitpunkt abgegriffen werden.

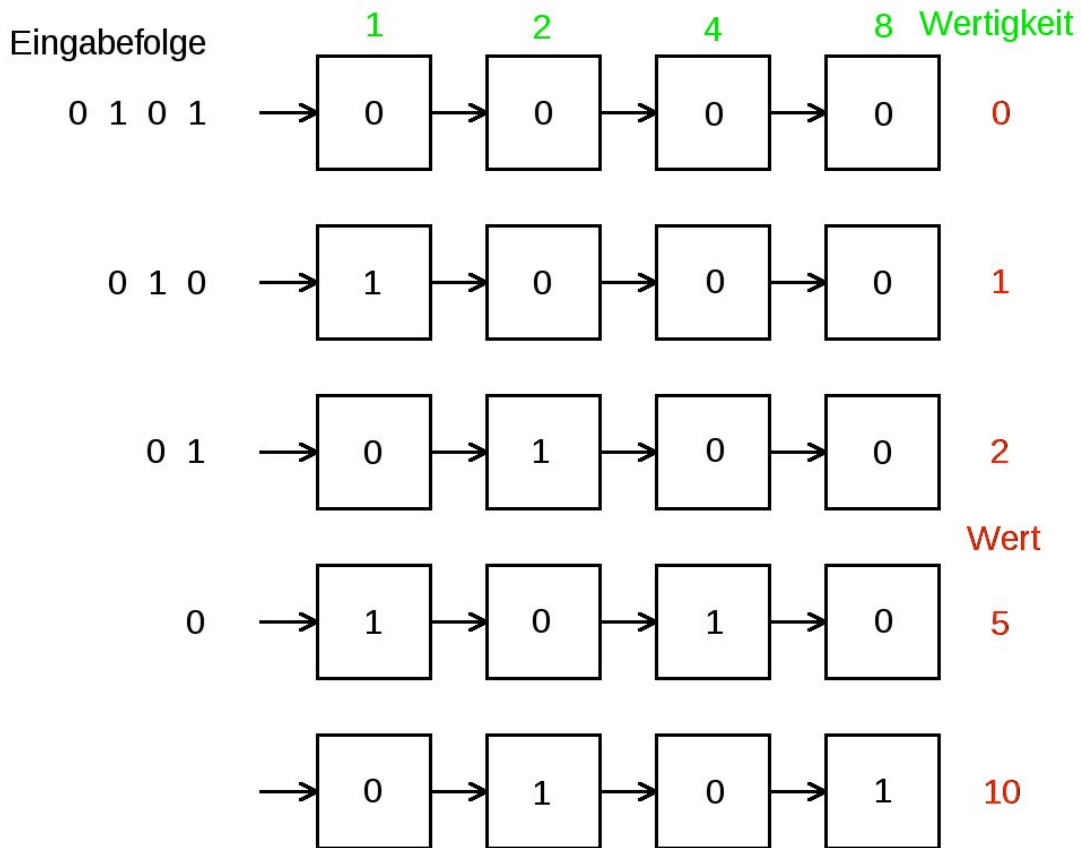


Bild 3: Schieberegister in das die Beispielfolge 0101 eingeschoben wird

Um den Schieberegisterinhalt (für die späteren Erklärungen) eindeutig beschreiben zu können wird jedem Einzelregister eine Wertigkeit zugewiesen (grün). Der Gesamtwert des Schieberegisterinhalts ist dann die Summe der mit „1“ besetzten Registerzellen (rot).

Ein weiterer Teil des Enkoders besteht aus einem Verknüpfungsnetzwerk von Exklusiv-Oder-Gattern. Exklusiv-Oder-Gatter geben am Ausgang eine binäre „1“ aus wenn sich die beiden Eingänge unterscheiden, in den anderen Fällen eine binäre „0“ also $00 \rightarrow 0$, $01 \rightarrow 1$, $10 \rightarrow 1$ und $11 \rightarrow 0$. In den Strukturplänen werden diese Gatter oft mit einem „+“ bezeichnet weil die Exklusiv-Oder-Funktion einer binäre Addition (ohne Übertrag) entspricht.

Schließlich wird noch ein Inverter benutzt der aus einer „1“ eine „0“ macht und umgekehrt. Dargestellt wird dieser Inverter als Kästchen mit der Bezeichnung „-1“

Der Viterbi Dekoder

Der 5-Bit Faltungsenkoder der bei PSK31/QPSK verwendet wird sieht damit so aus:

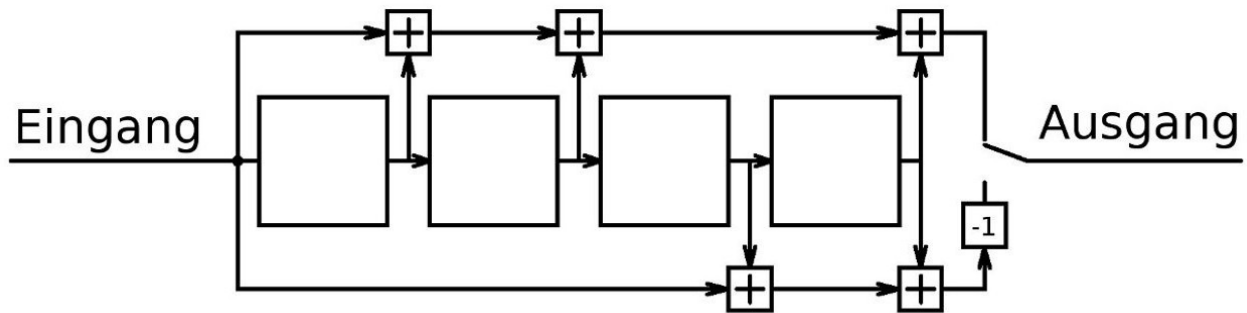


Bild 4: Struktur des bei PSK31/QPSK verwendeten Faltungsenkoders

In dem Bild erkennt man dass die Schieberegisterinhalte an verschiedenen Stellen abgegriffen und in zwei Zeilen (über und unter dem Schieberegister) verknüpft werden. Am rechten Ende ist ein Schalter zu sehen mit dessen Hilfe das obere und das untere Verknüpfungsergebnis ausgegeben werden kann. Um nun einen Datensatz zu kodieren wird das gesamte Schieberegister

- zu Beginn gelöscht
- das erste zu übertragende Bit aufgeschaltet
- die beiden Verknüpfungsergebnisse nacheinander ausgegeben
- das aufgeschaltete Bit in das Schieberegister geschoben
- das nächste zu übertragende Bit aufgeschaltet

Wie man aus der Beschreibung erkennen kann werden für jedes zu kodierende Bit zwei Bits ausgegeben. Der kodierte Datenstrom ist also genau doppelt so lang wie der originale Datenstrom.

4. Der Viterbi-Dekoder

Im Viterbi-Dekoder wird dieselbe Struktur nachgebaut wie sie auch im Encoder existiert. Weil auf der Sendeseite die Originaldaten ohne Veränderung Taktschritt für Taktschritt in das Schieberegister geschoben werden kennt man die zu übertragenden Originaldaten wenn man den Schieberegisterinhalt der Sendeseite zu jedem Zeitpunkt kennt. Die Grundidee der Dekodierung ist nun die, dass man die Originaldaten auch auf der Empfangsseite kennt wenn es einem gelingt die Schieberegisterinhalte auf der Empfangsseite zu jedem Zeitpunkt mit denen auf der Sendeseite synchron zu halten. Die Dekodierung konzentriert sich deshalb darauf den Schieberegisterinhalt auf der Empfangsseite mit dem auf der Sendeseite identisch zu halten.

4.2 Beschreibung des Registerinhalts

Eigentlich ist es trivial – der Registerinhalt lässt sich beschreiben indem man die Bits wie eine normale binäre Zahl liest. Man kann damit jedem möglichen Schieberegisterinhalt genau eine Zahl zuordnen die den Inhalt eindeutig beschreibt. Genau dieses Verfahren ist im Bild 5 dargestellt, die roten Zahlen sind die Wertigkeiten nach jedem Schieberegisterinhalt. Schiebt bei dem in Bild 5 gezeigten Beispiel noch „0000“ nach um den gesamten Registerinhalt wieder gelöscht zu bekommen ergibt

Der Viterbi Dekoder

sich die Zahlenfolge 0, 1, 2, 5, 10, 4, 8, 0, 0. Weit anschaulicher ist es natürlich, wenn man die Schieberegisterinhalte nicht nur durch Zahlen sondern graphisch darstellt. Das Trellis-Diagramm ist genau eine solche graphische Darstellung: In der Y-Achse werden die Zahlen, also die möglichen Registerinhalte aufgetragen, in der X-Achse die Taktschritte. Geht man davon aus, dass das Schieberegister zu Beginn den Inhalt 0 hatte (alle Zellen haben den Inhalt 0) dann sieht der Durchlauf eines einzelnen Bits wie folgt aus. Verbindet man die einzelnen Punkte erhält man das sogenannte Trellis-Diagramm oder genauer einen „Pfad“ im Trellis-Diagramm.

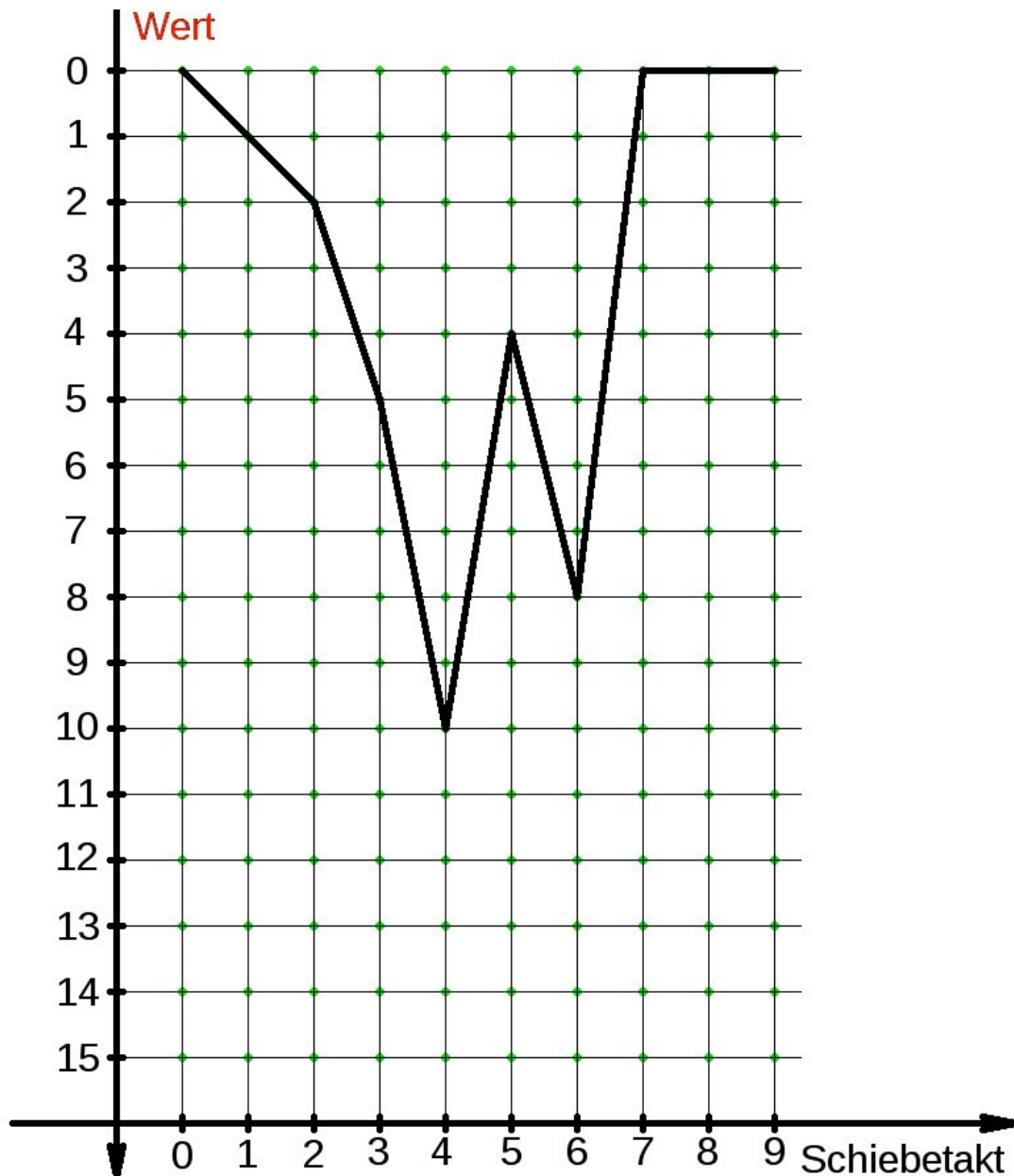


Bild 5: Trellis-Diagramm für das durch das Schieberegister geschobene Bitmuster 0000 0101

4.3 Die Dekodierung

Um die Dekodierung zu verstehen ist es am einfachsten sich zu erst einen Fall anzusehen bei dem es keine Übertragungsfehler gegeben hat.

4.3.1 Die Dekodierung im fehlerfreien Fall

Wie schon erwähnt besteht die Dekodierung aus der Aufgabe den Schieberegisterinhalt auf der Empfangsseite zu jedem Zeitpunkt so zu setzen wie den auf der Sendeseite. In ersten Schritt ist das einfach denn jede Übertragung beginnt vereinbarungsgemäß mit dem Registerinhalt „0“. Also setzt man das Empfangsschieberegister zuerst auf „0“ zurück.

Danach wartet der Empfänger auf die ersten beiden Bits. Die ersten **beiden** deshalb weil mit jedem Bit aus den Originaldaten **zwei** Bits zur Übertragung generiert werden. Die Kombination dieser beiden Bits wird in der Faltungskodierung auch als Symbol bezeichnet. Nach dem Empfang dieser beiden Bits (also dem Symbol) probiert der Empfänger aus welches Originaldatenbit also eine „0“ oder eine „1“ im eigenen nachgebildeten Enkoder zu derselben Bitkombination führt wie die empfangene. Das Datenbit das zur selben Bitkombination führt ist damit das erste auf der Sendeseite zur Kodierung angestandene Originaldatenbit.

Dieses korrekt dekodierte Originaldatenbit wird jetzt auf der Empfangsseite in das Schieberegister geschoben und das Empfangsregister damit für den nächsten Übertragungsschritt auf denselben Stand gebracht wie das Senderegister. Danach wartet der Empfänger auf die nächsten beiden Bits.

Mit den nächsten beiden Bits wird nun genauso verfahren wie mit den ersten beiden. Auch hier muss der Empfänger jeweils ausprobieren ob eine „0“ oder „1“ zur empfangenen Bitkombination führt denn jeder Schieberegisterinhalt führt zu einer anderen Kodierung. Diese Schleife

- warten auf die nächsten beiden Bits
- ausprobieren welches Originalbit zur selben Ergebnis führt
- und aktualisieren des Empfangsschieberegisters mit dem neu dekodierten Originalbit

lässt sich jetzt bis zum Ende der Übertragung fortsetzen.

Die Abfolge der Schieberegisterinhalte lässt sich mit einer Zickzack-Linie im Trellis-Diagramm darstellen. Im fehlerfreien Fall sind die Trellis-Diagramme der Schieberegister von Sende und Empfangsseite natürlich identisch. In unserem Beispielfall - Bitmuster 0000 0101 – sieht das Diagramm dann so aus wie im Bild 6 dargestellt.

Bei kleineren Schieberegistern (wie in unserem Fall) kann man das eben erwähnte „probieren“ vorwegnehmen und eine Tabelle erstellen in der man einträgt wie das Ergebnis der Kodierung einer „0“ oder einer „1“ abhängig vom Schieberegisterinhalt aussieht. In unserem QPSK-Enkoder besteht das Schieberegister aus vier Zellen, es gibt also nur 16 Möglichkeiten:

Der Viterbi Dekoder

Wert des Schieberegisters	Ergebnis der Kodierung einer „0“ 0- Symbol	Ergebnis der Kodierung einer „1“ 1-Symbol
0	01	10
1	10	01
2	00	11
3	00	11
4	11	00
5	00	11
6	10	01
7	01	10
8	11	00
9	00	11
10	10	01
11	01	10
12	01	10
13	10	01
14	00	11
15	11	00

Der Viterbi Dekoder

Wie man aus in der Tabelle 1 auch erkennen kann wurde die Kodierlogik so gewählt dass sich die Kodierergebnisse (Symbole genannt) für den gleichen Registerinhalt immer um 2 Bits unterscheiden (es gibt also nur die Symbole 01 / 10 und 00 / 11). Damit ist sichergestellt dass Übertragungsfehler erkannt werden denn jedes veränderte einzelne Bit führt zu einer fehlerhaften Bitkombination.

Ein vollständiger Dekodierdurchlauf sieht im Trellis-Diagramm dann so aus wie im folgenden Bild dargestellt. Zusätzlich zur bekannten Kurve (dem Pfad) wurden noch die Symbole eingetragen, die zum jeweils nächsten Punkt im Diagramm geführt haben und die Anzahl der detektierten Fehler (im hier fehlerfreien Fall natürlich an allen Stellen Null).

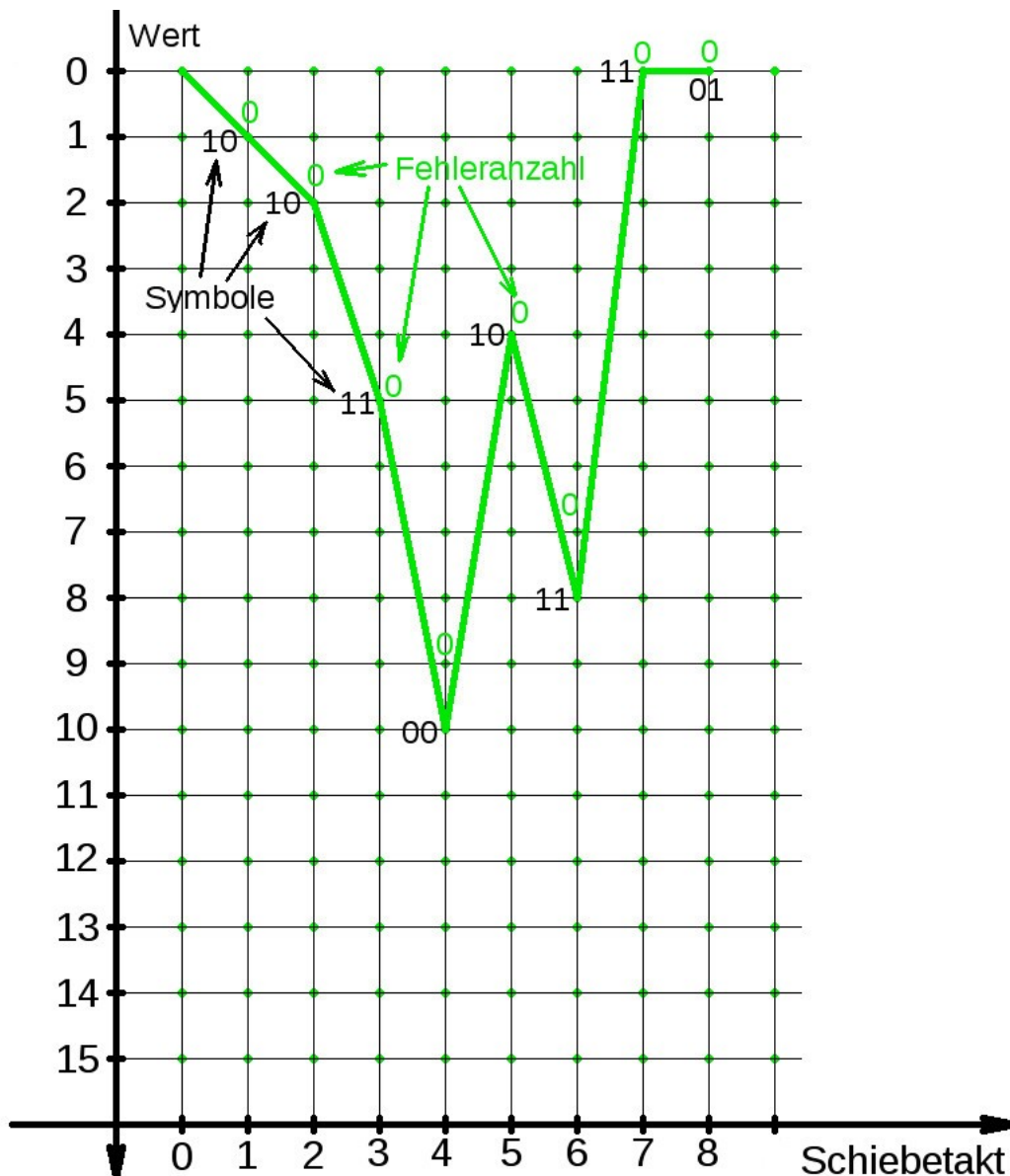


Bild 6: Trellis-Diagramm für eine fehlerfreie Dekodierung

4.3.2 Die Dekodierung bei Übertragungsfehlern

Mit zwei Bits lassen sich vier Zustände darstellen, mit einem Bit nur zwei. Damit sind in jedem Übertragungsschritt jeweils nur zwei der vier Möglichkeiten gültig. Treten bei der Übertragung Fehler auf die dazu führen, dass auf der Empfangsseite eine Bitkombination auftaucht, die sich durch Probieren ob das Originalbit eine „0“ oder ein „1“ gewesen ist nicht erzeugen lässt, lässt sich auf dem Empfangsseite nicht mehr feststellen ob das Originalbit auf der Sendeseite „0“ oder eine „1“ ist. Nachdem sich das richtige Bit nicht mehr ermitteln lässt trägt man jetzt im Trellis-Diagramm beide Möglichkeiten ein. Graphisch teilt sich der Pfad damit in zwei Äste auf. Zusätzlich vermerkt man an der Stelle der Verzweigung dass ein Fehler stattgefunden hat.

Wir haben jetzt den Zustand, dass einer der beiden vermerkten Registerinhalte der richtige sein muss, der jeweils andere falsch. Welcher stimmt und welcher nicht, lässt sich auf der Empfängerseite vorerst nicht mehr feststellen.

Nach dem Empfang der nächsten beiden Bits probiert der Dekoder nun mit beiden Registerinhalten getrennt aus ob sich die empfangene Bitkombination erzeugen lässt. Gehen wir davon aus, dass die Übertragung diesmal fehlerfrei war, wird sich die Kombination mit dem korrekten Registerinhalt sicher erzeugen lassen, mit dem fehlerhaften Registerinhalt im Mittel aber nur in 50% der Fälle. Der Dekoder geht nun so vor, dass er fehlerfreie Pfade weiterführt und im Fall eines Fehlers den betroffenen Pfad ein weiteres mal verzweigt. An den Verzweigungspunkten wird die jeweilige Fehleranzahl vermerkt die bis zu diesem Punkt aufgetreten ist.

Bleiben wir gedanklich bei nur einem Übertragungsfehler, wird sich folgendes Bild ergeben:

- ein Pfad wird in Trellis-Diagramm ohne weitere Verzweigung durchlaufen. Dieser Pfad ist mit der Kennung „1 Fehler“ versehen.
- zusätzlich wird es ein ganzes Bündel von Pfaden geben, die sich mehr und mehr verzweigen, weil die Registerinhalte der beiden Schieberegister (wegen der fehlerhaften Annahme eines Inhalts) unterscheiden und es deshalb im Mittel bei jeder zweiten Übertragung zu einer weiteren Verzweigung kommen wird.
- Besonders bei kurzen Schieberegistern kann es vorkommen, dass ein mit mehreren Fehlern versehener Pfad zufällig wieder auf den korrekten Pfad trifft und die beiden Register ab dem Zeitpunkt wieder synchron laufen. Dieser Pfad unterscheidet sich von dem fehlerärmeren aber durch die Zahl aufgetretener Fehler.

In dem folgenden Bild ist zu sehen, wie sich der Pfad der nach dem erkannten Fehler von Wert 2 nach 9 verläuft im weiteren Verlauf immer weiter aufteilt weil das Schieberegister im Empfänger nicht mehr mit dem im Sender synchron ist. Die in den nicht synchronen Pfaden für die weitere auffaserung verantwortlichen als fehlerhaft vermuteten Symbole wurde der Übersichtlichkeit wegen im Diagramm nicht eingetragen.

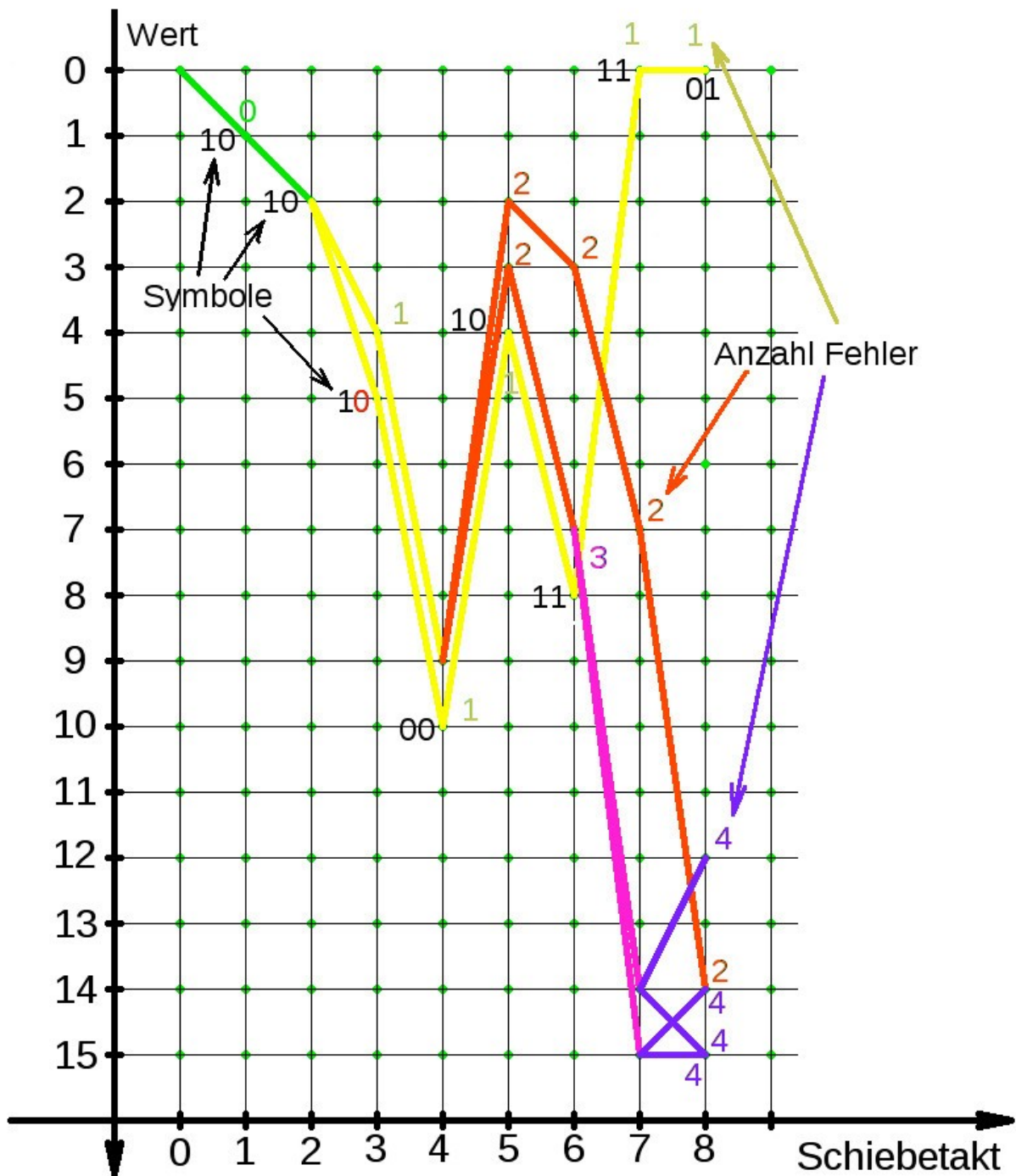


Bild 7: Trellis-Diagramm einer Dekodierung mit einem Übertragungsfehler im 3. Symbol

4.3.3 Das Dekodierungsergebnis

Am Ende der Übertragung sucht man sich aus dem Trellis-Diagramm den Pfad mit den wenigsten Fehlern raus. Die Bitfolge die zu dem ausgesuchten Pfad geführt hat entspricht mit der größten Wahrscheinlichkeit der Originalbitfolge wie sie im Sender kodiert wurde. Die während der Dekodierung auf dem ausgewählten Pfad vermerkte Fehleranzahl ist ein Maß für die Güte der

Datenrekonstruktion. Bei nur sehr wenigen Übertragungsfehlern ist das Ergebnis eindeutig (ein Pfad mit sehr kleiner Fehlerrate, alle anderen mit einer deutlich höheren), bei sehr vielen Übertragungsfehlern können die möglichen Ergebnisse nahe beieinander liegen – natürlich mit einer entsprechend hohen Anzahl von detektieren Übertragungsfehlern versehen. Im dem in Bild x dargestellten Fall ist der gelbe Pfad mit der Kennung „1 Fehler“ der, der die korrekte Dekodierung beinhaltet.

5. Die Schwächen des Viterbi Dekoders

Natürlich gibt es einige Eigenschaften die Viterbi-Dekoders die sich in machen Fällen als Problem erweisen. Einige lassen sich durch geeignete Maßnahmen kompensieren, andere nicht.

5.1. Die Durchlaufzeit / Korrekturqualität

Die Dekodierergebnisse sind nicht schon beim ersten Bit aussagekräftig. Um ein Dekodierergebnis mit der maximal möglichen Aussagekraft zu erhalten müssen mindestens so viele Schritte durchlaufen worden sein wie die Zellenanzahl des Schieberegisters. Die Durchlaufzeit entspricht damit der Zeit eines Taktes mal der Anzahl der Bitstellen im Schieberegister. Dasselbe gilt auch für die letzten Bits. Um auch diese Bits mit derselben Korrekturqualität zu dekodieren wie alle anderen ist es üblich nach dem Ende des Nutzdatenstroms auf der Sendeseite in den Originaldaten so viele „0“ nachzuschieben wie das Schieberegister Bitstellen hat. Aus diesem Grund ist das in Bild x gezeigte Trellis-Diagramm nicht ganz korrekt. Wenn bekannt ist, dass die letzten vier Originaldatenbits (vier weil es vier Schieberegisterzellen gibt) Nullen sind, müssen auch die letzten vier Symbole jeweils Symbole sein die Nullen in den Originalbits kodieren. Sind bei den nicht synchronen Paden am Ende scheinbar korrekte Symbole eingetroffen die aber zu „1“ gehören, müssten auch die als zusätzliche Fehler gezählt werden und statt den „1“ Nullen nachgeschoben werden.

5.2 Das Verhalten bei Bündelfehlern

Speziell bei Kurzwellenverbindungen treten durch Schwund bedingte Übertragungsausfälle auf, die nicht nur einzelne Bits betreffen sondern ganze zusammenhängende Reihen von Bits. Fehler die nicht nur einzelne Bits sondern zusammenhängende Folgen von Bits betreffen werden als Bündelfehler bezeichnet.

Der Viterbi-Dekoder reagiert auf solche Bündelfehler recht empfindlich. Der Grund ist dass der Dekoder nur so lange funktioniert, so lange mindestens ein Pfad mit der Sendeseite Synchron ist. Fallen mehrere Bits hintereinander aus, entstehen Pfade, die alle von der Sendeseite abweichen und damit nicht mehr auf den synchronen Pfad hin konvergieren.

Bei den Übertragungsverfahren die man für Kurzwellen verwendet behilft man sich indem man die Daten nach der Kodierung „verwürgelt“. Dabei werden die Bits so umsortiert, dass Bits die bei der Übertragung direkt hintereinander lagen nach der Umsortierung im Datensatz einen großen Abstand

haben. Bündelfehler werden damit in viele Einzelbitfehler „aufgelöst“ womit der Viterbi-Dekoder wiederum recht gut umgehen kann.

5.3 Der Rechenaufwand

Bei Dekodern mit kurzen Schieberegistern ist der Rechenaufwand minimal – entsprechend klein sind auch die Fähigkeiten bei der Fehlerkorrektur. Bei einem Schieberegister mit 5 Bit länge sind z.B. maximal 32 Pfade möglich. Ganz anders sieht es bei längeren Schieberegistern weil die Zahl möglicher Pfade $z=2^n$ (z = Bitzahl des Schieberegisters) ansteigt. Bei 16 Bits liegt z schon bei 65536, bei 32 Bit (wie bei WSPR benutzt) sind sogar 4294967296 Pfade möglich. Bei größeren Datenmengen und fehlerhafter Übertragung kann die Zahl zu bearbeitender Pfade sehr schnell ansteigen und einen erheblichen Teil der Anzahl der möglichen Pfade erreichen. Hat man nicht gerade beliebig viel Zeit und muss die übertragenen Daten „online“ dekodieren, muss man Kompromisse schließen und Pfade die eine bestimmte Fehlerzahl erreicht haben löschen. Auch beim Ergebnis muss entschieden werden, ab welcher Fehleranzahl die Daten als „nicht mehr dekodierbar“ eingestuft werden sollen. Setzt man diese Grenze zu optimistisch an, kann es passieren, dass eine vermeintlich korrekt abgelaufene Dekodierung nur absolute Zufallsdaten liefert die mit dem originalen Datensatz nichts gemein haben. Aus diesem Grund kommt es bei WSPR gelegentlich vor dass „Spots“ mit völlig unsinnigen Daten ausgegeben werden.

6. Mögliche Strukturen der Faltungs-Encoder

Die im Amateurfunk (bei QPSK und WSPR) verwendeten Strukturen sind so aufgebaut, dass um die Schieberegisterzeile zwei Netzwerke aus Exklusiv-Oder-Gattern existieren. Die Abgriffe aus den Schieberegistern sind nicht zufällig; vielmehr gibt es zu jeder Schieberegisterlänge nur eine kleine Anzahl möglicher Netzwerke. Alle anderen führen zu sogenannten „katastrophalen Kodierungen“ - das sind Kodierungen die nicht mehr eindeutig sind und sich deshalb nicht mehr dekodieren lassen.

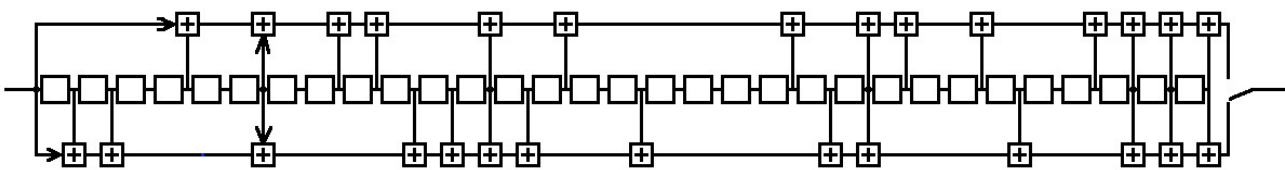


Bild 8: Struktur des bei WSPR benutzten Faltungsencoders

Im Laufe der Zeit haben sich bei anderen Anwendungen eine große Zahl anderer Strukturen etabliert. So sind anstatt der nur zwei Netzwerke um das Schieberegister herum auch mehrere möglich. Weiterhin gibt es Strukturen die Rückkopplungen enthalten.

7. Quellenverzeichnis

de.wikipedia.org/wiki/Faltungscode

TU Chemnitz, Proseminar Kodierverfahren, Thema Faltungscode

www.n sarc.ca/jt-modes.pdf