

## Bináris rádióátviteli csatornák modellezése

ETO 621.391.8:519.72

A hibajavító kód eljárások tervezésénél a zajos csatorna és a dekódoló algoritmus együttes modellezése igen előnyös. A dekódoló tényleges viselkedése ugyanis elméleti megfontolások alapján csak közelítőleg adható meg. Az egyes kódok emellett ugyanazon hibastruktúrákra különbözőképpen reagálnak. A kódok tulajdonságainak megítélése, ill. azok összehasonlítása modellezés után nagyobb biztonsággal történhet.

A rádiócsatornában az átvitel során némely jel meghibásodik. A tapasztalat szerint bizonyos rádiócsatornában a hibás jelek előfordulása független attól, hogy megelőzőleg mikor lépett fel hibás jel. Ezt a csatornát független csatornának nevezzük.

Más esetekben a hibás jel után nagyobb valószínűséggel következnek be újabb hibák, a hibák mintegy csomókban jelentkeznek.

A csatornák modellezésénél a független csatorna egy paraméterrel, a csomós hibákat tartalmazó pedig az alábbi két paraméterrel egyszerűen jellemezhető:

$$\begin{aligned} \Pr(0|0) &= p \\ \Pr(1|0) &= q = 1 - p \\ \Pr(1|1) &= P \\ \Pr(0|1) &= Q = 1 - P. \end{aligned} \quad (1)$$

A valószínű csatornában független és csomós hibák egyesén fordulnak elő, és arányuk igen különböző. Gyakorik az igen hosszú hibamentes intervallumok. Más esetekben több hibacsomó követi egymást, viszonylag rövid időn belül mintegy sűrűsödnek a hibacsomók. Egyszerű modellekkel a valószínű csatorna viselkedése nem adható meg.

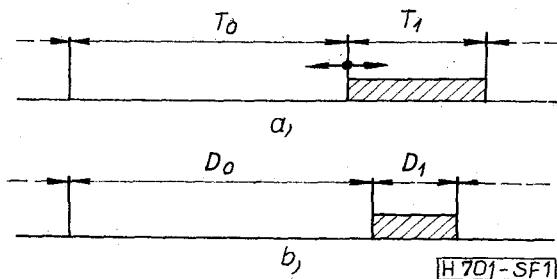
A tapasztalat szerint kétféle modell megfelelő kombinációjával a valószínűsághoz jobban igazodó modell szerkeszthető. A modell egyszerűsített Markov-láncot realizál. A modell ugyanis olyan Markov-mátrix-szal adható meg, amelynek fő átlójában különböző  $p_{11}, p_{22}, p_{33} \dots P$  átmenetvalószínűségek találhatóak. Az utolsó sor elemei  $Q_1, Q_2, Q_3 \dots P$ , az utolsó oszlopé pedig  $q_1, q_2, q_3 \dots P$ . A mátrix többi eleme zérus értékű. A mátrixhoz számítógépes algoritmus szerkeszthető.

Bizonyos konvolúciós kódrendszerek hibajavító képességét azzal jellemezhetjük, hogy

— a javítható hibákat vagy hibacsomókat mekkora hibamentes szakasznak (védőköznek) kell elválasztania,

— mekkora a javítható hibacsomó hosszúsága.

A fenti két jellemző egy-egy kódrendszerre kons-



1. ábra. A  $T_0, T_1$ , valamint  $D_0, D_1$  mennyiségek összehasonlítása

tans érték. Amennyiben valamely átviteli csatornát is úgy jellemzünk, hogy

— milyen hosszú hibamentes szakaszokat tartalmaz, továbbá

— milyen hosszú a hibacsomó, akkor a hibajavító kód eljárás tulajdonságait viszonylag egyszerűen vizsgálhatjuk.

Az 1a, ill. 1b ábrán a csatorna, ill. kódrendszer ezen paramétere egymás mellett szerepelnek. A csatornában azonban a  $T_0$  hibamentes, ill.  $T_1$  hibás intervallumok széles határok között ingadozhatnak, a kódok megfelelő  $D_0$ , ill.  $D_1$  szakaszait viszont bizonyos egyszerűsítéssel) állandónak vehetjük. A dekódoló akkor téveszt,

— ha a hibamentes szakasz rövidebb, mint a védőtáv, vagy

— ha a hibacsomó hosszabb, mint a javítókészég.

Eddigi kísérleteink során számos csatornafajta és dekódoló eljárást tanulmányoztunk. A vizsgálatok egy része arra irányult, hogy téves dekódolásnál mi volt a hiba oka. Megállapítottuk, hogy

— független hibák esetén a téves dekódolás az esetek kb. 70–85%-ában azért következett be, mert a védőtávban hibás jel fordult elő (a  $D_0$  védőtáv rövidebb, mint a  $T_0$  hibamentes intervallum),

— csomós hibák esetén a téves dekódolás az esetek kb. 55–70%-ában azért következett be, mert a  $T_1$  hibacsomó hosszabb volt, mint a  $D_1$  mennyiség.

Az optimális dekódoló eljárásnak egyesén előforduló hibafajta esetén ellentétes feltételeket kell kielégítenie. Ilyen csatornák dekódolására az irodalom elsősorban

— bonyolultabb felépítésű dekódolókat,

— több, sorba kapcsolt egyszerű dekódolókból álló rendszert

javasol.

A továbbiakban olyan eljárásokat vizsgáltunk, ahol a dekódolásnál nem az egymás melletti, hanem a távolabbi ( $n$ -edik) jeleket vesszük figyelembe. Ekkor a csatornahibák struktúrája már eltérő lesz.

Jelöljük ugyanis az (1) szerinti stacionárius Markov-lánc kezdeti állapotának valószínűségét  $P_0(0)$ -val, ill.  $P_0(1)$ -gyel. A  $P, Q, p, q$  átmenetvalószínűségekkel leírható Markov-lánc állapotának valószínűsége a kezdeti állapot után  $n$  lépés múlva

$$P_n(1) = \frac{Q}{Q+q} + (1-q-Q)^n \left( P_0(1) - \frac{Q}{Q+q} \right) \quad (2)$$

és

$$P_n(0) = \frac{q}{Q+q} + (1-q-Q)^n \left( P_0(0) - \frac{q}{Q+q} \right) \quad (3)$$

lesz.

A (2) és (3) összefüggések arra mutatnak, hogy ha az eredeti sorozat minden második, harmadik,  $n$ -edik elemét kiemeljük, és egymás mellé helyezzük, akkor ennek az új sorozatnak valószínűség-eloszlása az eredetitől különbözni fog. Ugyanis  $n \rightarrow \infty$

esetén a (2) és (3) összefüggések jobb oldalán a kitevős tényező zérus felé tart. Ennek következtében az  $n$  lépés utáni állapotérték egyre kevésbé függ attól, hogy a kiindulási állapot 0 vagy 1 volt.

Növekvő  $n$  esetén a megelőző állapot befolyása a következő állapotra egyre csökken,  $n$  lépés után a modell „elfelejti” a régi állapotát, és az új állapot valószínűsége — a független modellhez hasonlóan — megelőzőtől (Markov-láncok esetén az  $1-q-Q$  kifejezés a két, egymástól  $n$  lépés távolságban levő esemény közötti korrelációs együtthatót adja. A távolság növelésével az események közötti korreláció csökken.)

A mintavételezett sorozatokban így a hibák csomós jellege egyre inkább eltűnik, és a független eloszláshoz közeledik.

A 2a és 2b ábrán számítógépes modellezés eredménye látható. A  $P_0(n)$  az abszcissaértéknél hosszabb hibátlan szakasz előfordulási valószínűségét, a görbék paramétere a mintavételezési távolságot adja. (Az 1 paraméterű görbe az eredeti jelsorozatot jelöli.) Látható, hogy  $n=14, 64, 512$  esetén a rövid hibátlan szakaszok előfordulási valószínűsége egyre jobban csökken. Az ilyen csatornában tehát nem szükséges csomós hibák javítására szolgáló dekódolót alkalmazni.

Az 1. táblázatban két különböző jelsorozaton végzett modellezés eredménye található. Az eredeti jelsorozatokat a számítógép egy rövidhullámú rádiócsatornára jellemző Markov-mátrix szerint állította elő. Az eredeti jelsorozat mintavételezése ezután  $n=64$  és  $n=700$  értékkel következett be. Mind az eredeti, mind a mintavételezett sorozatokat 4-féle önortogonális és 6-féle ortogonalizálható dekódoló eljárással dekódoltuk. A táblázatok a dekódolás után ki nem javított hibák számát tartalmazzák.

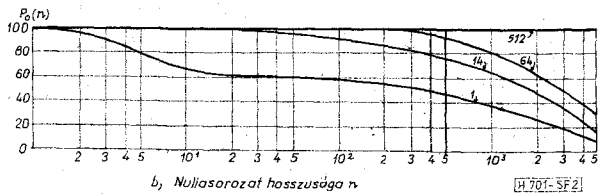
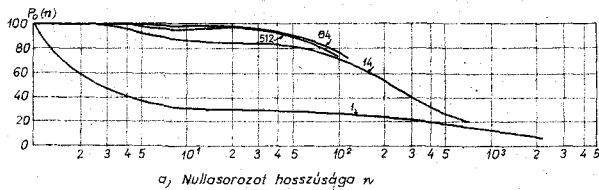
Megfigyelhető, hogy  $n=64$  esetében a fennmaradó hibák száma az  $n=1$  esethez képest többnyire jóval kisebb, de  $n=700$  már nem hoz minden esetben lényeges javulást. Ez arra mutat, hogy egy-egy valóágos összeköttetés esetén a mintavételezési távolságot nem érdemes túlzottan megnövelni. Az optimális távolság beállítása kísérleti úton, történhet.

A modellezés alapján a hibajavító kódrendszerek konstrukciójával kapcsolatosan az alábbi tapasztalatokat nyertük:

1. A rádiócsatornák hibajavító kódolásához egyszerűbb kód eljárás és mintavételezés együttes alkalmazása célszerű.

2. A mintavételezést célszerű változtatható paraméterű áramkörrel megoldani. Az optimális értéket egy-egy csatornára a valós üzem során kísérletileg kell megállapítani.

Fentiek figyelembevételével a Mikrohullámú Híradástechnika Tanszéken egy kísérleti mikrovezérelt hibajavító berendezés készült. Ebben 4-féle,  $n=1, 16, 64, 192$  távolságú mintavételezés állítható be. A hibajavító részben egyszerű önortogonális kódoló, ill. dekódoló algoritmust alkalmaztunk.



2. ábra. A mintavételezés hatása a hibátlan szakaszok gyakoriságára

1. táblázat

Hiba-sorozat	Kódtípusok	Javítatlan hibák száma		
		n=1	n=64	n=700
1	Önortogonális 1	23	5	7
	Önortogonális 2	29	16	5
	Önortogonális 3	29	11	0
	Önortogonális 4	19	33	11
	Ortogonalizálható 1	34	8	4
	Ortogonalizálható 2	29	16	14
	Ortogonalizálható 3	32	5	3
	Ortogonalizálható 4	28	7	8
	Ortogonalizálható 5	41	12	13
	Ortogonalizálható 6	34	12	4
2	Önortogonális 1	29	7	9
	Önortogonális 2	31	3	8
	Önortogonális 3	32	6	7
	Önortogonális 4	31	2	3
	Ortogonalizálható 1	37	17	4
	Ortogonalizálható 2	46	20	28
	Ortogonalizálható 3	59	10	11
	Ortogonalizálható 4	26	5	4
Ortogonalizálható 5	45	2	3	
Ortogonalizálható 6	43	10	3	