

# TURBO KODOVI

## Kodiranje poruke

Na strani odašiljača od informacije, poruke stvara se kodni zapis za koji je karakteristično povećanje redundancije. To znači da uz bitove koji predstavljaju samu poruku postoje i paritetni bitovi kojima je cilj povećati energetska učinkovitost kod bežičnih komunikacijskih sustava i to na način da se na prijemnoj strani vrši korekcija pogrešaka, nastalih pri prijenosu (razuman broj takvih), iz samog koda (FEC). Postoje još i korekcijski kodovi s automatskim zahtjevom za ponavljanjem čim se detektira pogreška (ARQ). Budući se kod FEC kodova tolerira veći broj grešaka, mogu si priuštiti rad s manjom energijom odašiljanja, prijenos na veće udaljenosti, veću toleranciju smetnji, upotrebu manjih antena i brži prijenos.

Binarni FEC koder uzima  $k$  bitova da bi na izlazu imao kodnu riječ od  $n$  bitova, gdje je  $n-k$  broj paritetnih bitova. Znači postoji ukupno  $2^n$  različitih kodnih riječi od kojih  $2^k$  njih pripada samoj poruci. Omjer  $k/n$  se naziva *kodni omjer* i označava se sa  $r$ . Kod kodova gdje je njegov iznos manji, dopušten je veći broj pogrešaka pri prijenosu i time se postiže veća *energetska učinkovitost*. U obrnutom slučaju imamo manji broj paritetnih bitova što vodi većoj *iskoristivosti frekvencijskog područja*.

Za svaku kombinaciju kodnog omjera ( $r$ ), duljine kodne riječi ( $n$ ), modulacijske tehnike, tipa kanala i snage šuma postoji minimalni iznos energije koja se mora utrošiti da bi se jedan bit prenio od odašiljača do prijemnika. On se naziva *kapacitet kanala* ili *Shannon-ov kapacitet* (*Teorija informacije*, 1948.). Mnogi znanstvenici su pokušavali konstruirati kodove koji će polučiti uspjeh te teorije. Napredak je ipak postojao, tako da je ranih 90-ih godina razlika između teorije i prakse za binarnu modulaciju bila 3dB, ali uz jednostavni model kanala (uključuje samo AWGN šum) što znači da je to optimalni slučaj, teško izvediv u praksi.

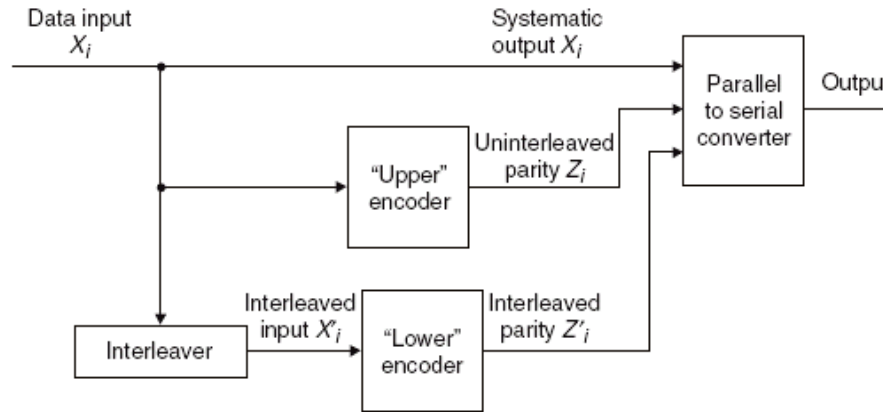
## Turbo kod

Grupa istraživača koji su radili u Francuskoj 1993. razvili su turbo kodove. Početni rezultati što se tiče energetske učinkovitosti pokazali su približavanje na 0.5dB od Shannon-ovog kapaciteta. U početku skeptično dočekan, ovaj rezultat je ipak bio potvrđen s tim da su karakteristike koda s vremenom poboljšane.

Do kraja 90-ih turbo kod je bio nadaleko poznat i implementiran u različite sustave. Sad se nalazi u NASA standardima za svemirsku komunikaciju, digitalni video prijenos i oba (cdma2000 i UMTS) standarda za mobilne uređaje treće generacije.

## Kodiranje paralelnim ulančavanjem sa miješanjem

Zanimljivost turbo koda je to da je on zapravo kombinacija dva koda koje krasi sinergijsko djelovanje, a ne bi bilo moguće upotrebom samo jednog. Turbo kod se zapravo formira *paralelnim ulančavanjem* dvaju kodova koji su odvojeni miješalom. U praksi su ta dva FEC koda koja su najčešće identična. Kao što se vidi na Sl.1 paralelni izlaz koji se sastoji od podatkovnog i dva paritetna bita se serijalizira.



Sl.1 turbo koder

Uloga miješala je promijeniti poredak bitova poruke na propisani, ali opet nepravilan način. Znači, isti su bitovi na ulazu i na izlazu miješala ali ne na istom mjestu u kodnoj riječi (algoritam miješala oponaša miješanje karata). Bez miješala uz iste kodere i isti ulaz imali bi isti izlaz što ne bi proizvelo nikakav učinak. Doduše upotrebom miješala podatak  $\{X_i\}$  je preuređen tako da ulaz drugog koder postaje  $\{X'_i\}$ . Dakle njegov izlaz je gotovo sigurno različit od prvog osim u stvarno rijetkim slučajevima (npr. sve nule ili sve jedinice).

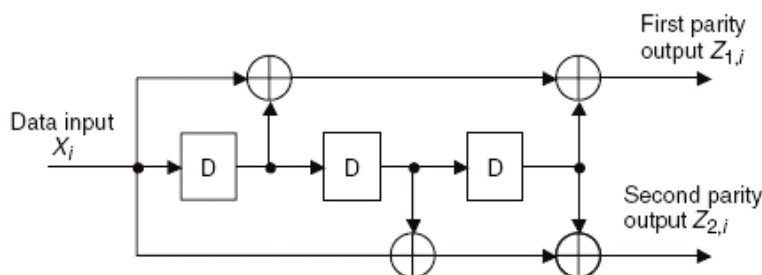
## Zašto je toliko dobar ?

Da bi se o ovom moglo govoriti potrebno je prvo uvesti dva važna pojma. Za kod možemo kazati da je *linearan* ako modul-2 suma dvije valjane kodne riječi daje također valjanu kodnu riječ (XOR operator). *Hammingova težina* ili samo *težina* kodne riječi je jednaka broju jedinica koje ona sadrži. Treba primjetiti da linearni kod mora sadržavati kodnu riječ sastavljenu od samih nula inače će riječ koja je sumirana sama sa sobom, koristeći već spomenuti operator, dati nultu kodnu riječ.

Kvaliteta linearnog koda se očituje u već spomenutoj težini koda. Kod veće težine poželjniji je jer je mnogo jasniji, te će dekodner imati lakši posao. To nameće zadatak koderu da mora proizvesti kod koji će minimizirati pojavu kodnih riječi manje težine. Ali to se i događa kod turbo koda. Čak i ako jedan koder izbacuje kod manje težine vjerojatnost da će to učiniti i drugi je vrlo mala. Razlog tome je miješalo te se zbog toga to poboljšanje u odnosu na slučaj kad ga nemamo zove *pojačanje miješala* i jedan je od važnih razloga zašto je turbo kod toliko dobar.

## Konvolucijski kodovi

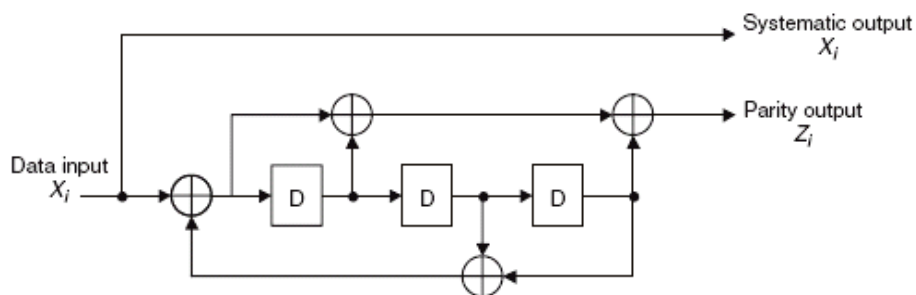
Iako se kao koder može upotrijebiti bilo koji, u praksi se gotovo uvijek koristi *rekurzivni sistematični konvolucijski* (RSC) koder. RSC kodovi su vrlo slični standardnim *kovolucijskim kodovima* NSC kodovima koji se koriste za mobilne sustave 2G, setelitskoj komunikaciji i u drugim aplikacijama. Na Sl.2 vidimo jedan takav.



Sl.2 Nesistematični konvolucijski (NSC) koder,  $r = 1/2$

Podatak ulazi s lijeva i pohranjuje se u linearnom posmačnom registru (D bistabil). Na svaki idući ciklus i dolazak novog bita poruke postojeći bit se pomiče udesno. Svaki od dva izlazna bita se dobije *isključivo ili* operacijom između pojedinih bitova u registrima. Budući se za jedan podatkovni bit dobiju dva paritetna, kodni omjer  $r = 1/2$ . Uvodi se i pojam *granične dužine* koji odgovara maksimalnom broju bitova o kojima ovisi izlaz kodera. U ovom slučaju  $K = 4$  (tri u registrima i jedan bit na ulazu utječu na izlaz).

Problem sa navedenim koderom je njegova *nesistematičnost*, odnosno kodni zapis samog podatka se ne pojavljuje na izlazu. Kodna riječ sadrži zbog toga samo paritetne bitove i ne može biti podijeljena na polje podatkovnog koda i polje pariteta potrebnog za turbo kod. Zato koristimo *sistematičan* kod koji pak možemo dobiti iz postojećeg NSC kodera tako da jedan njegov paritetni izlaz spojimo na ulaz kao što vidimo na Sl.3 a prenosimo onaj drugi.

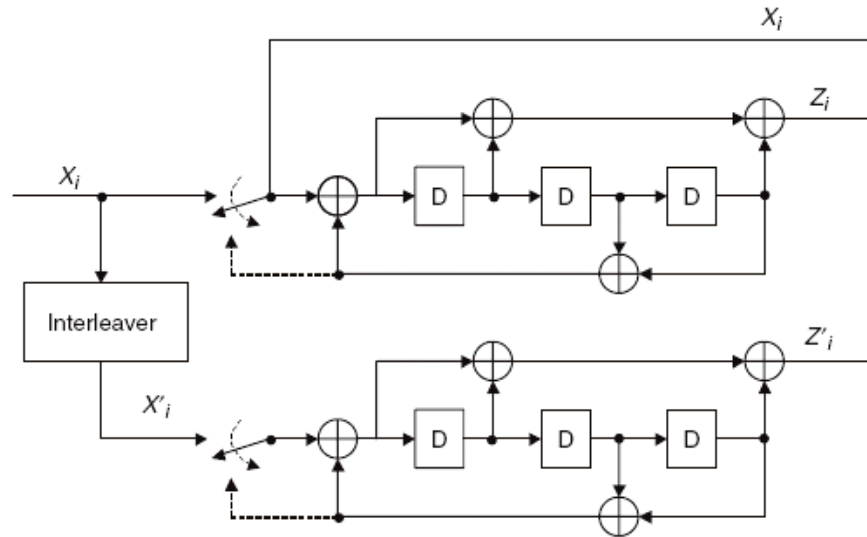


Sl.3 rekurzivni sistematični (RSC) koder koji koristi UMTS koder

Sad je moguće prenositi uz paritetni i podatkovni bit a još uvijek zadržavamo isti kodni omjer  $r = 1/2$ . Izlazi također da je povratna veza u koderu potrebna da bi se moglo postići maksimalno pojačanje miješala. Sistematični izlazi kodera su nepotrebni jer su identični (iako jedan promiješan) ulazu u sami turbo koder.

## UMTS turbo koder

Uz cdma2000 UMTS je najprihvaćeniji standard za mobilni standard 3G. Koder za UMTS turbo kod sadrži dva kodera granične duljine  $K=4$  identičnih onima na Sl.3. Kao što se vidi na Sl.4 izlaz ovog kodera je serijalizirana kombinacija sistematičnih bitova  $\{X_i\}$ , paritetnog izlaz prvog kodera  $\{Z_i\}$  i drugog kodera  $\{Z'_i\}$ .



SI.4 UMTS turbo koder

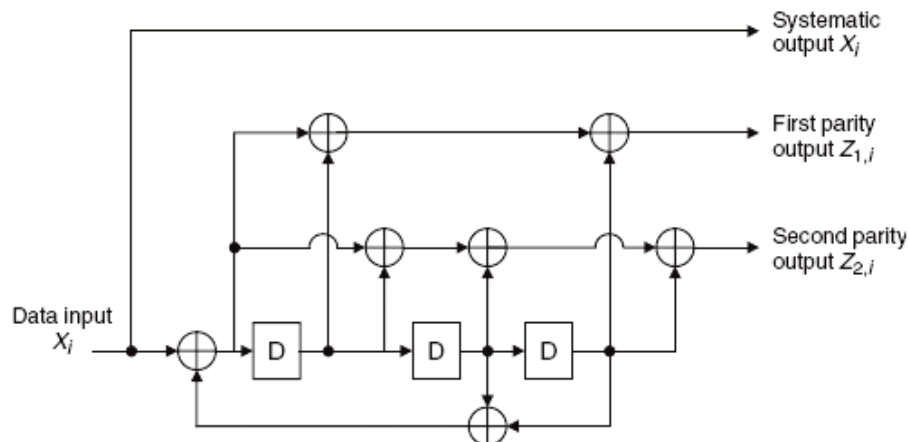
Ukupni kodni omjer iznosi približno  $r = 1/3$ . Zašto približno? Duljina ulazne riječi može biti od 40 do 5114 bitova. Prije samog kodiranja inicijalno stanje svih posmačnih registara je nula. Nakon kodiranja  $k$  bitova ulazne riječi svaki koder bi bio u jednom od osam stanja. Međutim koder funkcioniše puno bolje ako zna početno i krajnje stanje registara. Očit izbor bi bio i da završno stanje karakteriziraju tri nule.

Za NSC koder to dobijemo ako na kraju kodne riječi dodamo još *tri nule*. Ali kod RSC-a to nije moguće. Tu koristimo povratnu vezu jer *isključivo ili* između dva ista bita daje nulu te nakon tri ciklusa takta u svim registrima će biti upisana nula. Fizički je to izvedeno tako što se nakon  $k$  bitova poruke preklopnik spoji na povratnu vezu. Ali zbog miješala će ta tri *dodana bita* biti različita za dva enkodera. Odsalani okvir uključuje dodane bitove za gornji  $\{X_{k+1}, X_{k+2}, X_{k+3}\}$  i donji koder  $\{X'_{k+1}, X'_{k+2}, X'_{k+3}\}$  te također paritetne bitove za oba kodera  $\{Z_{k+1}, Z_{k+2}, Z_{k+3}\}$ ,  $\{Z'_{k+1}, Z'_{k+2}, Z'_{k+3}\}$ . Dakle stvarni kodni omjer iznosi  $r = k/(3k+12)$ , ali treba uzeti u obzir da je utjecaj dodatnih bitova zanemariv za kodne riječi veće duljine.

### Cdma2000 turbo kod

Ovaj koder razlikuje se od prethodnog po algoritmu miješala, dopuštenoj duljini ulaznog koda i kodnom omjeru korištenih RSC kodera. Dok je kod UMTS dozvoljeni raspon duljine ulazne veličine iznosi  $40 \leq k \leq 5114$ , kod cdma2000 postoje točno određene duljine ulaznog koda: 378, 570, 762, 1146, 1530, 2398, 3066, 4602, 6138, 9210, 12282 ili 20730. Već spomenuto miješalo odlikuje se drugačijim radom nego kod UMTS-a.

Dva konstituirajuća kodera koja se koriste kod cdma2000 su prikazana na SI.5.



Sl.5 1/3 RSC koder kojeg koristi cdma200 turbo koder

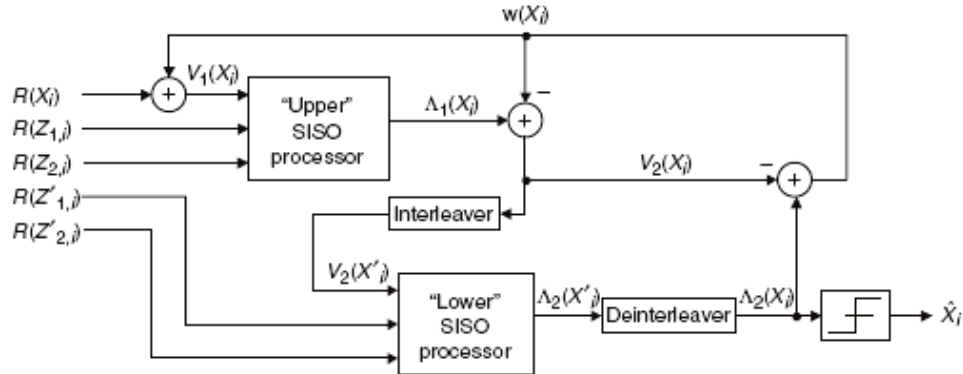
Svaki ima tri izlaza (jedan sistematičan i dva paritetna). Dakle kodni omjer iznosi  $r = 1/3$  (opet približno). Prvi paritetni bit  $Z_{1,i}$  generira se na isti način kao kod UMTS koda  $Z_i$ . Znači razlikuju se samo po prisutnosti drugog paritetnog bita  $Z_{2,i}$ . Ovaj koder također se upotrebljava u strukturi paralelnog ulančavanja prikazanog na Sl.1. Budući da svaki od dva koda proizvodi dva paritetna bita kodni omjer je sada  $r = 1/5$ . Za mnoge aplikacije tako niska vrijednost kodnog omjera je nepoželjna te zato cdma2000 posjeduje mogućnost transformiranja u manji kodni omjer  $r = 1/2$ ,  $1/3$  i  $1/4$ . Taj mehanizam naziva se *kodiranje s prekidanjem*. Npr. turbo koder briše drugi paritetni bit svakog koda pa onda cdma2000 funkcionira isto kao UMTS. Mehanizam za postizanje ostalih kodnih omjera nešto je kompliciraniji. Uz to cdma2000 također ima dodatne bitove da bi vratio posmačne registre u nulto stanje.

## Turbo dekodiranje

Nakon kodiranja cijela n-bitna riječ se uokviruje u odgovarajući sinkronizacijski okvir, modulira, prenosi kanalom i naposljetku dekodira. Neka je  $U_i$  modulacijski bit (sistematičan ili paritetni) a  $Y_i$  isti taj ali na prijemničkoj strani (nakon izlaza iz korelatora ili prilagođenog filtra). Dok  $U_i$  može biti samo 0 ili 1 dotle  $Y_i$  može poprimiti bilo koju vrijednost. Turbo dekodeoer zahtijeva ulaz ima sljedeći oblik:

$$R(U_i) = \ln \frac{P(Y_i | U_i = 1)}{P(Y_i | U_i = 0)}$$

gdje  $P(Y_i | U_i = j)$  predstavlja uvjetnu vjerojatnost primanja  $Y_i$  signala ako je poslan bit  $U_i = j$ . Ovakvi izrazi predstavljaju *logaritam kvocijenta vjerojatnosti* ili *LLR* vrijednost poslanog bita te se koriste u procesu dekodiranja.



Sl.6 Aritektura dekodera kod cdma2000 i UMTS

Ovakav oblik ulaza zahtijeva ne samo poznavanje signala  $Y_i$  već i karakteristike prijenosnog puta. Primljene vrijednosti se moraju transformirati u LLR oblik kako za sistematične tako i za paritetne bitove i kao takve su spremne za turbo dekodiranje (Sl.6). Ovaj dekodera je pogodan za oba standarda s tim da se za UMTS ulazi  $R(Z_{2,i})$  i  $R(Z'_{1,i})$  postavljaju u nulu.

Za svaki podatkovni bit turbo dekodera mora proračunati sljedeći LLR:

$$\Lambda(X_i) = \ln \frac{P(X_i = 1 | Y_1 \dots Y_n)}{P(X_i = 0 | Y_1 \dots Y_n)}$$

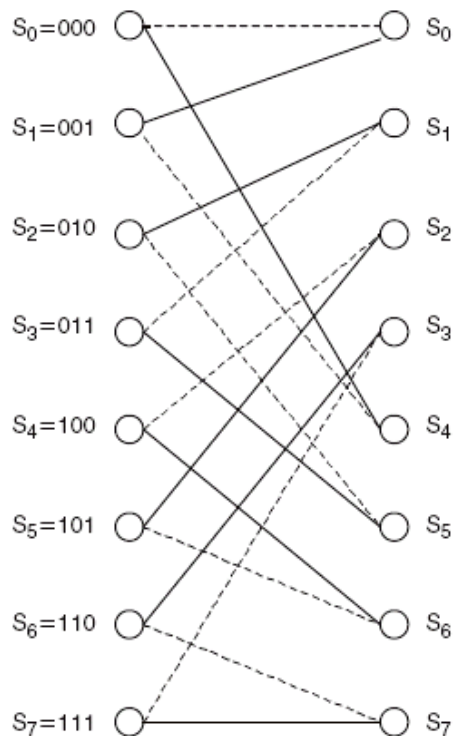
LLR dakle u ovom slučaju uspoređuje vjerojatnost da je konkretni bit 1 nasuprot vjerojatnosti da je 0 s obzirom na cijelu kodnu riječ. Kad se izračuna LLR odluka o vrijednosti  $X_i$  bita prepušta se komparatoru s tim ako  $\Lambda(X_i) > 0$  onda se pretpostavlja  $\hat{X} = 1$  i ako  $\Lambda(X_i) < 0$ ,  $\hat{X} = 0$ .

Turbo dekodera koristi primljenu kodnu riječ i znanje o strukturi samog koda da bi izračunao  $\Lambda(X_i)$ . Ali to nije jednostavno budući miješalo komplicira strukturu koda. Zato dekodera obavlja LLR estimaciju u dva koraka. U prvom računa jednadžbu koristeći strukturu gornjeg koda a u drugom donjeg. Odgovarajući rezultati označavaju se sa  $\Lambda_1(X_i)$  odnosno  $\Lambda_2(X_i)$ . Za dobivanje ovih veličina koristi se SISO procesor. Budući dva procesora računaju LLR za iste podatke (iako izmješane) karakteristike dekodera se mogu poboljšati razmjenom informacija između njih. Dakle izlaz prvog treba dovesti na ulaz drugog i obrnuto. Zbog ove razmjene informacija između procesora algoritam dekodiranja je iterativan. Nakon svake iteracija estimacija podatka je bolja, iako svaka sljedeća iteracija tome manje doprinosi. Zbog te razmjene informacija turbo kod je i dobio ime, jer to podsjeća na povratnu vezu između usisnog i isisnog kompresora turbo motora.

Kao što je poznato, kod sustava sa povratnom vezom mora se paziti na oscilacije i stoga se samo ona informacija jedinstvena za pojedini procesor može predati drugom procesoru. Dok procesori primaju različite paritetne bitove sistematični bitovi su jednaki. Dakle sistematični ulazi svakog procesora moraju se oduzeti od njegova izlaza prije nego će njegov izlaz ići u drugi procesor. Ta razlika između LLR izlaza i sistematičnog ulaza pojedinog procesora naziva se *nepravna informacija* i označava se sa  $w(X_i)$ .

## SISO processor

Srce turbo dekodera je algoritam implementiran u SISO procesore. SISO algoritam koristi *rešetkasti dijagram* gdje su prezentirane sve moguće vremenske sekvence kodera. Praktično taj dijagram prikazuje sva stanja moguća na kraju  $i$ -tog vremenskog ciklusa gdje je  $i$  ide od 1 do  $k+3$  (zbog dodatnih bitova). Budući RSC koderi imaju tri registra broj mogućih stanja je osam. Na prijelazu iz jednog ciklusa u drugi dolazi do promjene stanja (trenutak  $i$ ) tako da dijagram prikazuje moguće novonastalo stanje (u trenutku  $i+1$ ). Primjer možemo vidijeti na Sl.7 gdje pune crte predstavljaju prijelaz uz ulazni bit  $X_i = 1$  a isprekidane crte za  $X_i = 0$ . Te crte predstavljaju *ogranke*. Svaka promjena kodne riječi je predstavljena jednim putom kroz rešetku..

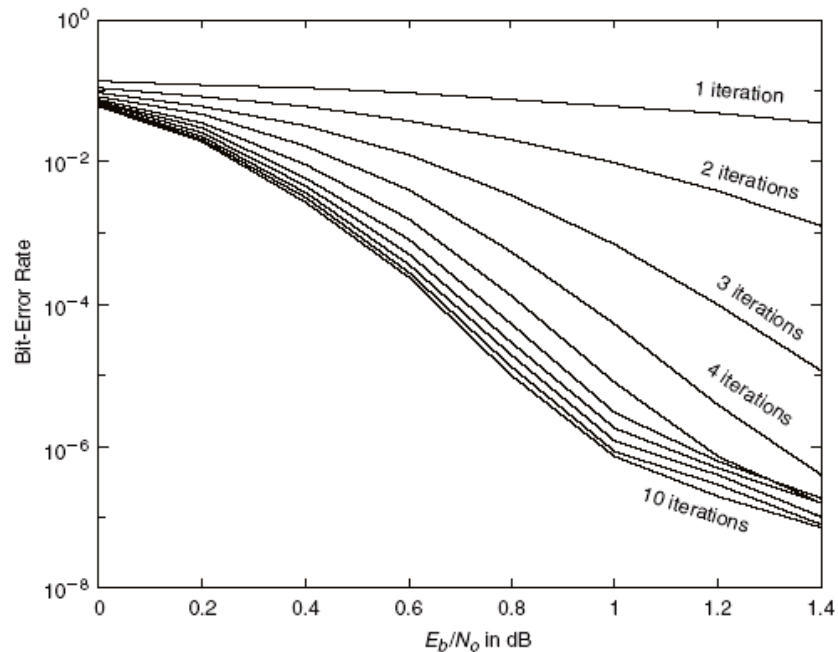


Sl.7 tipična sekcija rešetkastog dijagrama

SISO algoritam mjeri težinu svake pojedine grane (učestalost pojave određenog stanja) i može dobiti LLR estimaciju svakog podatkovnog bita  $\Lambda(X_i)$  propisanom obradom izmjerene težine ogranaka. Ta obrada je moguća uz korištenje jednog od dva nova algoritma, *SOVA* ili *MAP* kriterij. Ovi algoritmi sastavni su dio teorije detekcije gdje je problem odlučiti se za jednu od dvije hipoteze (0 ili 1) na temelju raspoloživih podataka (primljena zašumljena vrijednost). Oba su vezana uz Viterbijev algoritam za dekodiranje konvolucijskih kodova a koriste se mogu dati izlaz u LLR formi dok Viterbijeb ne može.

## Karakteristike turbo koda 3G

Simulacije su provedene uz AWGN kanal koristeći BPSK modulaciju. Upotrebljvan je MAP kriterij odlučivanja. Rezultati su prikazani u obliku BER-SNR krivulje. Sl.8 prikazuje karakteristike UMTS turbo koda uz kodnu riječ  $k = 1530$  bita.

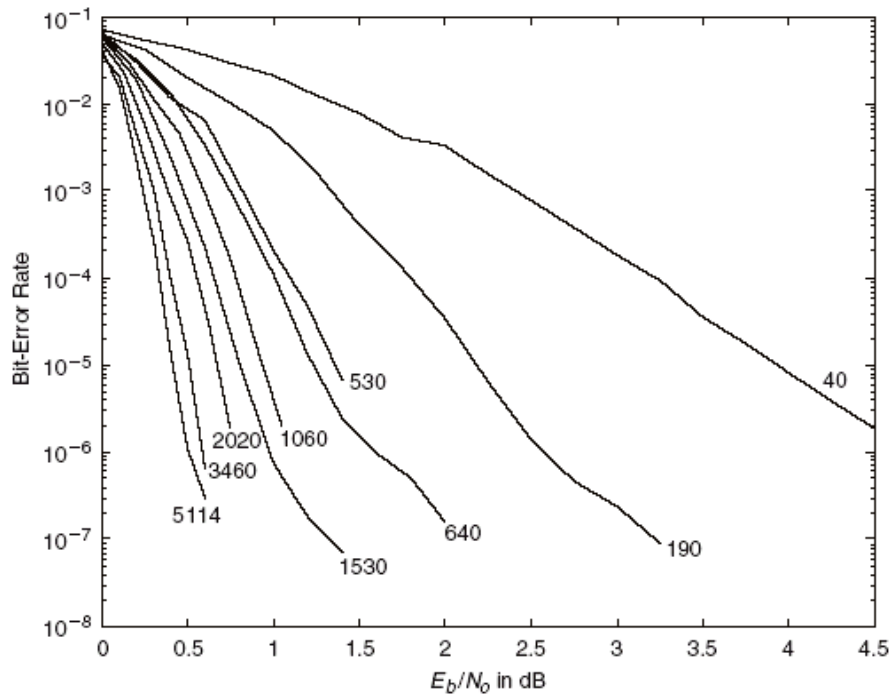


Sl.8 Karakteristika UMTS turbo koda pri promjeni broja iteracija od 1 do 10

Slika pokazuje kako se poboljšavaju karakteristike turbo koda kako broj iteracija raste. Ali ipak nakon pete iteracije to poboljšanje je minimalno tako da veći broj iteracija nije poželjan zbog duljeg vremena dekodiranja i veće kompleksnosti postupka.

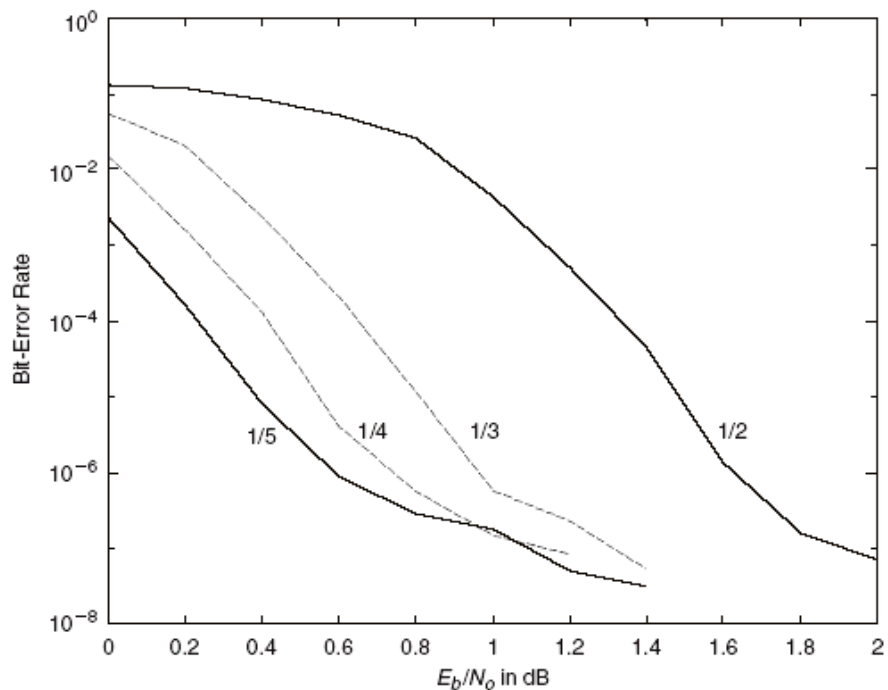
Sl.9. pokazuje karakteristike koda uz duljinu okvira kao parametar. Kao što je vidljivo povećanjem okvira postizemo bolju učinkovitost postupka, a sve zbog većeg pojačanja miješala.





SI.9 Karakteristike UMTS turbo koda za različite duljine ulazne riječi

SI.10 pak pokazuje osobine cdma2000 turbo koda za okvir  $k = 1530$  bita uz korištenje BPSK modulacije i AWGN kanala ali ovaj put uz kodni omjer kao parametar. Rezultat je očekivan jer veći broj paritetnih bitova jamči manju vjerojatnost pogreške ne samo kod turbo koda nego i kod bilo kojeg drugog FEC koda. Još se može uočiti da za  $r = 1/3$  turbo kod ima iste osobine kako za cdma2000 tako i za UMTS što je bilo i očekivano. Već spomenuta razlika u miješalu, dakle ne utječe znatno na krivulju.



SI.10 Karakteristike cdma2000 turbo koda uz različite kodne omjere

