ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

Қ.И. Сәтбаев атындағы Қазақ ұлттық техникалық зерттеу университеті

Ақпараттық және телекоммуникациялық технологиялар институты

Электроника, телекоммуникациялар және ғарыштық технологиялар кафедрасы

Ізбасар А. І.

Сызықты блокты кодтарға арналған тордың анализы

ДИПЛОМДЫҚ ЖҰМЫС

5В071900 – «Радиотехника, электроника және телекоммуникация» мамандығы

Алматы 2019

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

Қ.И. Сәтбаев атындағы Қазақ ұлттық техникалық зерттеу университеті

Ақпараттық және телекоммуникациялық технологиялар институты

Электроника, телекоммуникациялар және ғарыштық технологиялар кафедрасы

ҚОРҒАУҒА ЖІБЕРІЛДІ
Кафедра меңгерушісі
техн. ғыл. канд-ы
__________ Е. Т. Таштай
«26» апреля 2019 ж.

ДИПЛОМДЫҚ ЖҰМЫС

Тақырыбы: «Сызықты блокты кодтарға арналған тордың анализы»

5В071900 – «Радиотехника, электроника және телекоммуникация» мамандығы

Орындаған: Ізбасар А. І.

Пікір беруші
ҚазҰАУ, ЭҮжА каф. меңгерушісі
доктор PhD.,
қауымдастырылған профессор
__________ Ж. С. Шыныбай
«25» 04 2019 ж.

Ғылыми жетекші
ЭТжҒТ кафедрасының
техн. ғыл. канд-ы
қауымдастырылған профессор
__________ Л. Б. Илипбаева
«24» 04 2019 ж.

Алматы 2019

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

Қ.И. Сәтбаев атындағы Қазақ ұлттық техникалық зерттеу университеті

Ақпараттық және телекоммуникациялық технологиялар институты

Электроника, телекоммуникация және ғарыштық технологиялар кафедрасы

5В071900 – Радиотехника, электроника және телекоммуникация

ҚОРҒАУҒА ЖІБЕРІЛДІ
Кафедра меңгерушісі,
техн.ғыл.канд.
_________ Е.Таштай
«20» 01 2019 ж.

Дипломдық жұмыс орындауға
ТАПСЫРМА

Білім алушы *Ізбасар Айбол Ізбасарұлы*

Тақырыбы «Сызықты блокты кодтарға арналған тордың анализы»

Университет ректорының *«16» қазан 2018 ж. № 1162-б бұйрығымен бекітілген.*

Аяқталған жұмысты тапсыру мерзімі "25" сәуір 2019 ж.

Дипломдық жұмыстың бастапқы берілістері:

1) Графтағы қысқа жолды іздейтін Витерби алгоритмы;

2) q=2, 3, 4 болғандағы GF(q) өрісінің мәндер кестесі;

3) F_i торының i=0 … 6 кезіндегі мәндер кестесі;

4) G алгебралық тобының келтірілмейтін көпмүшелер кестесі;

Дипломдық жұмыста қарастырылатын мәселелер тізімі:

а) Бөгеуілге тұрақты кодтар және модуляция туралы мағлұматтарға әдебиеттік шолу;

ә) Бөгеуілге тұрақтылықты арттыру мақсатындағы модуляция процесін және сызықты блокты кодтарды сараптап негіздеу;

б) Сызықты блокты кодтардың торын негіздеп сараптау және бағдарламалық түрде модельдеу;

Сызбалық материалдар тізімі (міндетті сызбалар дәл көрсетілуі тиіс)

Сызықты блокты кодтардан алынған тор сұлбасы көрсетілген.

Ұсынылатын негізгі әдебиет:

1) Журавлев В. Г. Помехоустойчивые коды. Владимир, 2013.
2) Кудряшов Б. Основы теории кодирования. Спб, 2016.
3) Shu L. Error control coding: fundamental and applications. New Jersey, 2012.

Дипломдық жұмысты (жобаны) дайындау
КЕСТЕСІ

Бөлімдер атауы, қарастырылатын мәселелер тізімі	Ғылыми жетекшіге және кеңесшілерге көрсету мерзімі	Ескерту
Бөгеуілге тұрақты сызықты блокты кодтар және модуляция процесіне аналитикалық шолу	20.01.2019 - 01.03.2019	
Сызықты блокты код торын негіздеп сараптау	02.03.2019-02.04.2019	
SystemView бағдарламасындағы Файер кодының кодтау және декодтау құрылғысының іске асырылуы	01.04.2019–15.04.2019	

Дипломдық жұмыс (жоба) бөлімдерінің кеңесшілері мен норма бақылаушының аяқталған жұмысқа(жобаға) қойған **қолтаңбалары**

Бөлімдер атауы	Кеңесшілер (аты, әкесінің аты, тегі, ғылыми дәрежесі, атағы)	Қол қойылған күні	Қолы
Норма бақылау	PhD докторы, ЭТжҒТ каф.сениор-лекторы Тайсариева К.Н.	25.04.19	

Ғылыми жетекшісі ______________________ Л. Б. Илипбаева
(қолы)

Тапсырманы орындауға алған білім алушы ______________ А. Ізбасар

Күні “25” ______04______ 2019 ж.

АҢДАТПА

Бұл дипломдық жұмыста сызықты блокты кодтар үшін код торының қолданылуы қарастырылған. Бөгеуілге тұрақты сызықты блокты кодтардың кодтық торларын сараптау арқылы, код торын құру және қолдану тиімділігін анықтау мәселесіне басты назар аударылған. Дипломдық жұмыс көлемінде код торын қолдану арқылы қысқа хабарламаларды декодтау процесі, Балл-Кук-Джелинека-Равив алгоритмының қолданылуы қарастырылған. SystemView бағдарламасының шегінде Файер кодының кодерінің және декодерінің жұмысы іске асырылған.

Нәтижелерді салыстырмалы түрде сараптау арқылы код торының артықшылықтары мен кемшіліктеріне баға берілген.

АННОТАЦИЯ

В данной дипломной работе рассматривается применение кодовых решеток для линейно-блочных кодов. Основное внимание обращено анализу кодовых решеток, при помощи методов построения и эффективности применения. В пределах дипломной работы даны декодирования коротких кодов с помощью алгоритмов Балла-Кука-Джелинека-Равива. Реализовано кодирующее и декодирующее устройство кода Файера в программе SystemView.

При помощи сравнительного анализа результатов, дана оценка преимущества и недостатка кодовых решеток.

ANNOTATION

In this thesis work discusses the use of code lattice for linear-block codes. The main attention is paid to the analysis of code lattice, using construction methods and application efficiency. Within the thesis, decoding of short codes is given using the Balla-Cook-Jelinek-Raviv algorithms. Implemented the coding and decoding device of Fire code in the SystemView program.

With the help of a comparative analysis of the results, the advantages and disadvantages of code arrays are estimated.

МАЗМҰНЫ

КІРІСПЕ

Цифрлы ақпарат тарату жүйелерінің мәселелері, заманауи шешімдерді талап етеді. Қазіргі таңда ақпаратты тарату жүйелеріндегі ең үлкен мәселе қолданушыларға тұрақты және сенімді байланысты орнату болып табылады. Байланыс сенімділігі тек қатесіз және бұрмалануларсыз жеткізілген хабар кезінде ғана артады. Мұндай сенімділікті бөгеуілге тұрақты кодтау теориясының жетістіктері бере алады. «Сенімділікті арттыру» соңғы уақыттағы қарқынды дамып келе жатқан графтар теориясындағы үрдіс, кез – келген қателерді болжамды таңдайтын процестерден, әсіресе ақпарат тарату жүйелерінен, нақтылықты және жедел шешімдерді талап етеді. Мұндай шешімнің бірін сызықты блокты кодтарды зерттеу саласына қатысты графтар теориясын кең көлемде қолдану бере алады. Бұл өте жылдам өзгерістер орын алып жатқан әлемдік телекоммуникациялық технологиялар нарығындағы, заманауи талаптар мен қағидаларға толығымен жауап бере алатын, жаңашыл көз қарасты туғыза алатын құралдың рөлін атқаратындығына дәлел бола алады

Дипломдық жұмысқа сызықты блокты кодтарға код торын қолдану тиімділігін анықтау мақсаты қойылған. Сызықты блокты кодтарды код торы арқылы көрсету, сонымен қатар код торын кодтау және декодтау кезінде қолдану арқылы оңтайлы алгоритмдерді анықтау және үйірткілі кодтарды декодтау үшін, белсенді қолданыс тапқын, тиімді алгоритмдердің бірін сызықты блокты кодтарға қолдану нұсқасын сараптау – дипломдық жұмыстың негізгі мәселелерін құрайды. Код торының тиімділігін анықтау салыстырмалы сараптау және бағдарламалық модельдеу көмегімен жүзеге асырылады. Зерттеу мақсатына жету үшін:

Бірінші, сызықты блокты кодтардың негізгі параметрлеріне және бөгеуілден қорғау үшін модуляция поцесіне қатысты жалпы мәліметтерге шолу жасалады.

Екінші, қарастырылып отырған тақырыпқа байланысты жүргізілген зерттеулерге және өзге де жұмыстарға шолу жасалу арқылы, сызықты блокты кодтарға қатысты код торының қолданылуына, құраушы және синдромды матрицасының көмегімен код торын құрастыру әдістері қарастырылады. Балл Кук Джелине Равив алгоритмі есептеледі және декодтау кезінде қолданылуына баға беріледі.

Үшінші, SystemView бағдарламасының мүмкіндіктерін пайдалану арқылы Файер кодының кодтаушы және декодтаушы құрылғыларының жұмысы іске асырылатын болады. Модельдеу нәтижелері теориялық мәліметтермен салыстырмалы түрде сарапталады.

1 Бөгеуілге тұрақты сызықты блокты кодтар және модуляция процесіне аналитикалық шолу

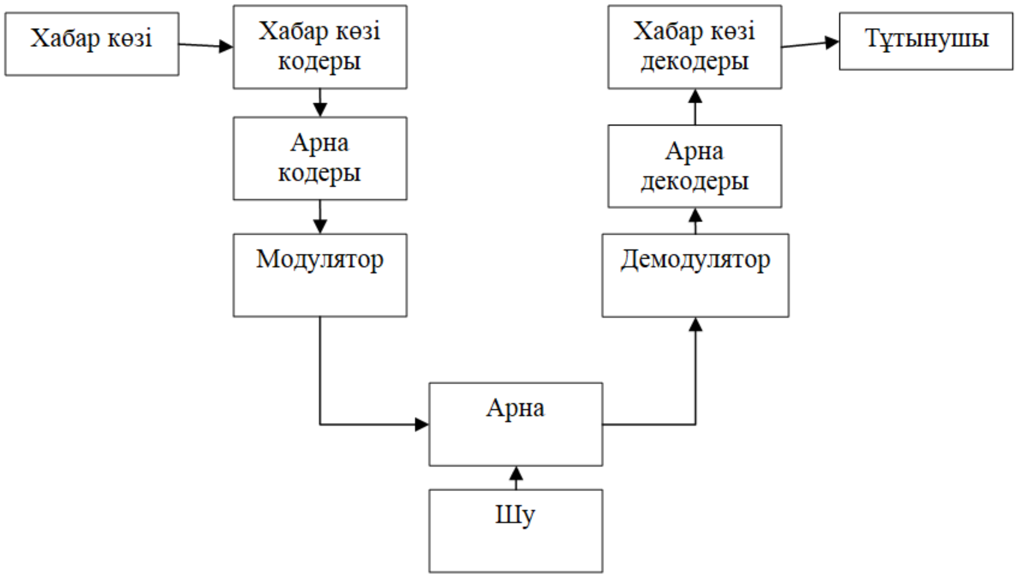
Телекоммуникациялық жүйелерде хабарламаларды жіберу сапасы – әлемдік ақпарат тарату жүйелеріндегі ақпарат көлемінің өсуімен тікелей байланысты болды. Бұл жағдайда кодтар және кодтау түсінігінің ауқымы хабарламаларды бөгде адамдардан қорғау түсінігінен де үлкен ауқымды иемдене бастады. Дәл қазіргі таңда код және кодтау түсінігі ақпаратты тиімді, ыңғайлы және ақпаратты қатесіз жеткізу мағынасымен қолданылады. Сол себепті ақпаратты тарату жүйесінің нақтылығын және дәлдігін арттыру мақсатында кодтау теориясының алар орны үлкен болып табылады. Себебі бөгеуілдер кез келген жүйе үшін кездейсоқ сипатқа ие болады және әсерін бағамдау қиын немесе көп жағдайда мүлдем мүмкін емес.

1.1 Бөгеуілге тұрақты сызықты блокты кодтарды сараптап негіздеу

Байланыс жүйесі ақпарат көзі мен қабылдаушыны байланыс арнасының көмегімен байланыстырады. Байланыс арнасы радиотолқындар арқылы, коаксиальды кабель немесе симметриалы кабель және т.б. жолдар арқылы іске асырылуы мүмкін. Жүйелерді жобалау барысында байланыс арнасына ақпаратты енгізетін және шығаратын құрылғыларды жасайды.

Арналық кодтарды қателерден қорғауына байланысты бөгеуілдерге тұрақты және бөгеуілдерге тұрақты емес деп екі топқа бөлуге болады. Кодтардың қателерді табуы және түзете алуы ең маңызды көрсеткіші болып табылады. Кодтау процесін жүргізу санау жүйесінің ыңғайлы болуын талап етеді, өз кезегінде мұндай санау жүйесі ретінде кодтау теориясында екілік жүйе жиі қолданылады. Екілік санау жүйесі позициялық принципте болады. Әр бір разряд келесі разрядтан екінің еселігіне үлкен болатын санау жүйесі цифрлық жүйенің негізі болып табылады. Қарапайым түрде құрылған бөгеуілден қорғалмаған код қатарына санды – импульсті, Морзе, Бодо кодтарын жатқызамыз. Бірлік импульстен тұратын және кодтық комбинациясы кодтық сөзбен тең болатын код арнадағы бөгеуілдерден ақпараттың дұрыс берілуін қамтамасыз ете алмайды. Сол секілді Морзе коды да, кодтық комбинацияларының ұзындықтары арқылы ғана айырмашылық жасайды және ақпараттың бөгеуілден қорғалуын немесе қателердің түзелуін қамтамасыз етпейді. [1]

Қазіргі таңдағы цифрлық байланыс арнасының кең таралған блок сұлбасы суретте көрсетілген.



Сурет 1.1 – Байланыс жүйесінің блок – сұлбасы

Хабар көзінен келетін ақпарат, 1.1 сурет бойынша, сигналды өңдеу, түрлендіру процесын оңтайландыру үшін хабар көзінің кодерінен өңдеуден өтеді. Бұл аралық түрлендіру символдардың тізбегі болып табылады, мұндай символдар тізбегін хабар көзінің кодалық сөзі деп атайды. Бұл түрлендіру ақпаратқа ешқандай артықшылық қоспайды. Артықшылық тек арна кодерінен кейін қосылады. Артықшылық бастапқы ақпараттың символдар тізбегіне қосылады. Яғни арнаның кодалық сөзі деп аталатын символдар тізбегін құрады. Әр бір сөздегі символ бит арқылы немес биттер тізбегімен беріле алады. Келесі қадам модуляция процесы және модуляцияланған сигналдың арнамен берілуі болып табылады. Арнада интерефеpенция, шуылдар немесе бөгеуілдердің әсерінің бар болуынан бастапқы берілген және қабылданған сигналда айырмашылық пайда болады. Қабылданған сигнал демодуляторға кіреді. Демодуляциядан кейін арна декодеры кодалық сөздің артықшылығын пайдалану арқылы қателіктерді түзетеді. Хабар көзінің декодері кодерге қарама карсы операцияны іске асырады. Барлық қателіктер түзетілгеннен кейін бастапқы символдар тізбегі мен қабылданған символдар тізбегі бірдей болуы тиіс, бұл ақпараттың толық дұрыс қабылданғандығын білдіреді.

Блок сұлбадан белгілі болғандай бөгеуілдерден ақпаратты сигналды қорғаудың ең тиімді жолы қатені табатын және түзете алатын кодтарды қолдану. Бұл қазіргі таңдағы электромагниттік спектрде жасанды сигналдар санының артып келе жатқанымен байланысты, коррекциялық қасиетке ие кодтардың қолданылуының маңыздылығын арттыруда. Мысалы, жерсеріктік байланыс жүйесінде таратқыштың қуатын арттыруға қарағанда код артықшылығын арттыру анағұрлым оңтайлы әдіс болып табылады, себебі қуатты арттыру қосымша құрылғыларды және қаржылық шығындарды қажет етеді. Демек, ақпаратты сигналға қосымша символдар, яғни тексеруші биттер немесе биттер тізбегін қосу арқылы қателікті табу және түзету кодтаудың негізгі мақсаты. Цифрлы каналдарда бұл кодтық сөздер арқылы, жоғарыда сипатталған сұлбамен іске асырылады.

Мысалы, ұзындығы n болатын, M комбинациялы E – екілік кодты қарастырайық. Мұндай код өте қарапайым болады, кодалық сөздері n ұзындықты. Бұл кодты $M=2^k$ арқылы сипатталады, мұндағы k-қандай да бір бүтін сан, мұндай код екілік (n, k) – код деп аталады.

Осы параметрлерін пайдаланып келесі кодты көрсетуге болады:

$$E = \begin{pmatrix} 1\,0\,1\,0\,1 \\ 1\,0\,0\,1\,0 \\ 0\,1\,1\,1\,0 \\ 1\,1\,1\,1\,1 \end{pmatrix} \tag{1.1}$$

Бұл код M = 4, n = 5 болатын жоғарыда айтылған шартты қанағаттандыратын екілік код болып табылады. Бұл кодты 2 бит ақпаратты көрсету мақсатында пайдалануға болады:

$$\begin{array}{l} 00 \leftrightarrow 10101 \\ 01 \leftrightarrow 10010 \\ 10 \leftrightarrow 01110 \\ 11 \leftrightarrow 11111 \end{array} \tag{1.2}$$

Егер қабылдағышқа осы 5 биттік кодтық сөздердің бірі қабылданса байланыс жүйесі ешқандай қатесіз 2 биттік ақпараттың бірін қабылдайды. Ал егер қандай да бір бит өзгеріске ұшыраса, ондай жағдайда қабылданған кодтық сөздің арасынан қабылдануы мүмкін ықтималдығы жоғары сөзді іздей бастаймыз. Мысалы, егер бізге 01100 кодтық сөзі келсе, онда 01110 кодтық сөзі қатемен берілді деген тұжырымға келеміз, демек қабылдануы тиіс ақпарат 10 болғаны. Бұл мысалдағы код өте қарапайым және түзете алатын қателер саны өте аз болғандықтан қолданылуға тиімсіз болып табылады. Сондықтан кодты таңдау кезінде кодалық сөздер бір бірінен ең үлкен шамаға айырмашылық жасауы қажет. Мұндай шаманы кодтау теориясында d Хэмминг арақашықтығы деп атайды. Хэмминг арақашықтығы екі q тізбектің n ұзындықты x және y комбинациясының бір бірінен айырмашылық жасайтын d(x, y) арақашықты атаймыз.

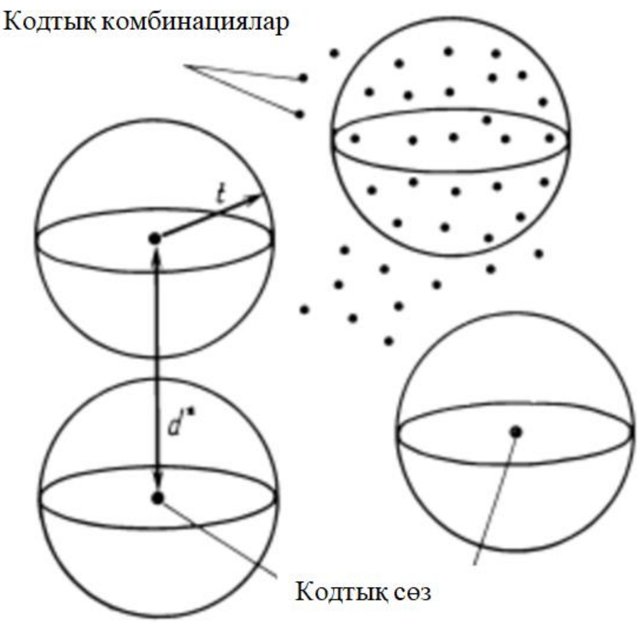
Мысалы, x = 10101 және y = 01100 комбинацияларын алайық. Онда Хэмминг арақашықтығы d (10101, 01100) = 3 болады.

Арнада t қателік орын алса және қабылданған сөзден әр кодтық сөзге дейінгі арақашықтық t-дан үлкен болса, декодер ең жақын кодтық сөзді дұрыс қабылданған деп қателерді түзетеді. Бұл шарт келесі түрде беріледі:

$$d \geq 2t + 1 \tag{1.3}$$

Бұл шартты геометриялық түрде көрсетуге болады. Барлық кодтық комбинациялар кеңістігінен қандай да бір n – тізбекті комбинациялар жиынтығы алынған және кодтық сөздер деп белгіленген. Егер d – минималды арақашықтығы болса, ал t – шартты қанағаттандыратын ең үлкен бүтін сан болса, онда әр кодтық сөзден өзара қиылыспайтын сфераны сызуға болады. Сфераның ішінде орналасқан кодтық комбинациялар қабылданған кезде, декодер бұл комбинацияларды сфераның центры болып табылатын кодтық сөздер ретінде декодтайды Егер қабылданған комбинацияда t қатеден көп болмаса, қабылданған комбинация тек сфера ішінде жатады және тек бір ғана кодтық сөзге тиесілі болуы мүмкін болады. Бұд жағдайда декодтау қатесіз жүзеге асады.

Жоғарыда көрсетілген t – қатеден көп қателер қабылданғанда, 1.2 сурет бойынша, комбинация басқа сфераның ішінде орналасады және басқа кодтық сөз ретінде декодермен қате қабылданады [2].



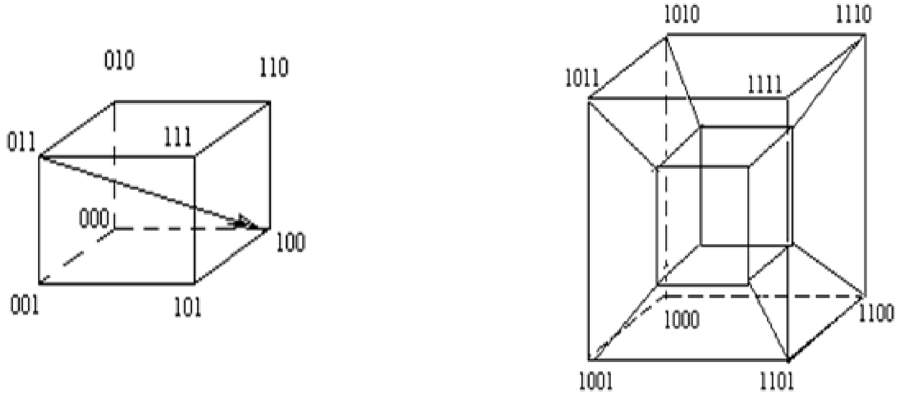
Сурет 1.2 – Декодтау сфералары

Осындай жағдайға байланысты декодерде қолданылатын екі әдісті қарастыруға болады.

Бірінші, тек қандай да бір сфераға тиісті болатын кодтық сөзді ғана қабылдап декодтайтын толық емес декодерды қолдану әдісі. Өзге комбинациялардың қателер саны көптігінен декодер бұл кодталған сөздерді ажырату мүмкін емес деп, түзетілмейтін сөздер қатарына жатқызады. Көптеген қолданыстағы декодерлер осындай тәсілге негізделген.

Екінші, толық декодерді қолдану. Бұл декодер кез келген комбинацияны ең жақын орналасқан сфераға тиесілі деп қабылда, сол сфераның кодтық сөзімен декодтайды. Бұл әдіс кездейсоқ нақты дұрыс кодтық сөзді тауып алу ықтималдығына сүйенеді. Код түріне байланысты қолданылатын кодер және декодер үйлесімі түрлі болуы мүмкін.

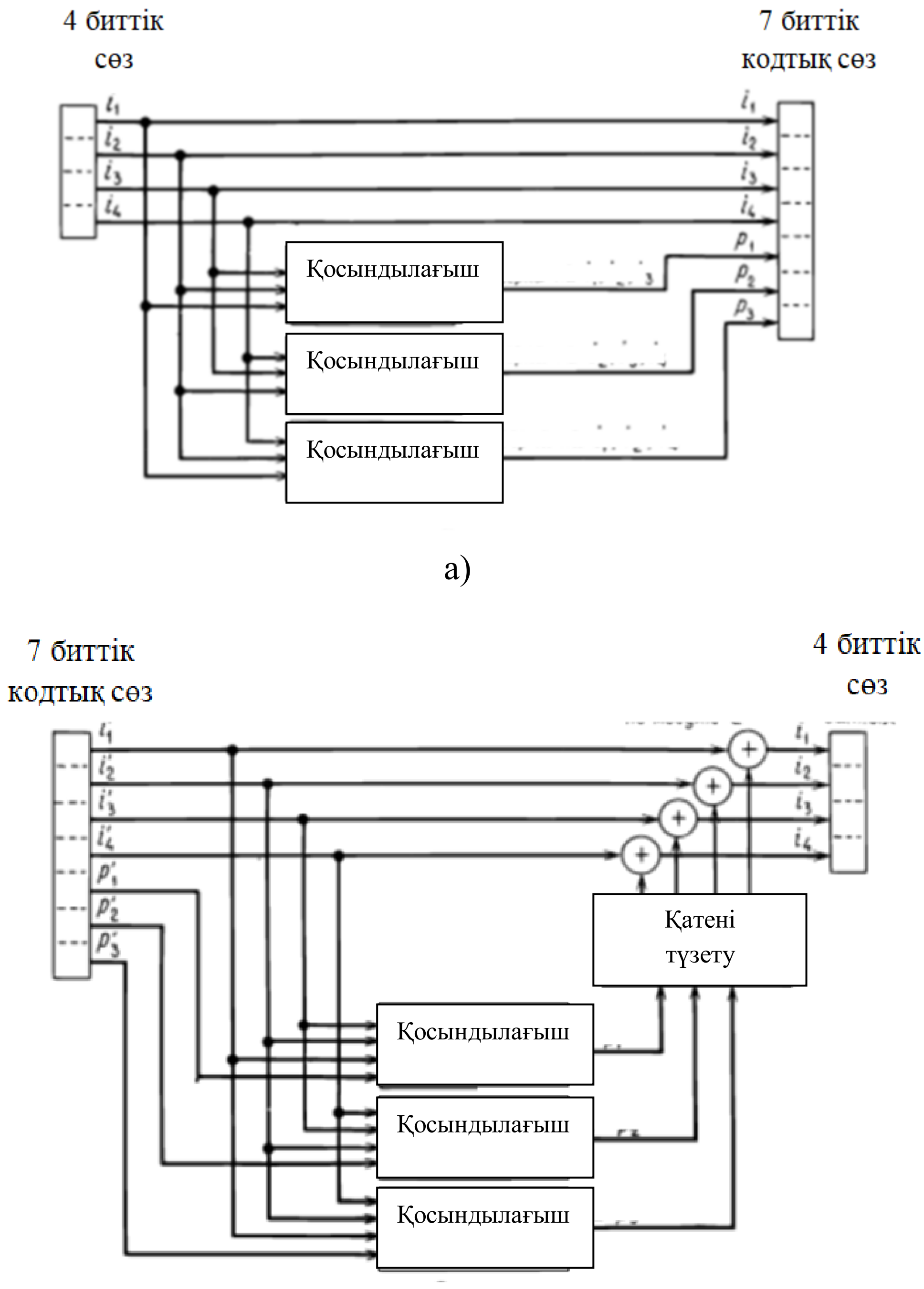
Сонымен қатар геометриялық интерпритацияны n – разрядты кодты комбинацияның n – еселі куб арқылы 1.3 – суреттегідей көрсетуге болады. Бұл жағдайда куб қырлары Хэмминг арақашықтығының ең кіші мәніне тең болады d = 1. Код ұзындығына байланысты куб өлшемі өзгереді, мысалы n=2 квадрат, n=3 бірлік, n=4 төрт өлшемді куб. Кубтың төбелері 2^n шамасымен анықталады, ол ең үлкен мүмкін болатын комбинация санына тең. Бұл геометриялық сұлба өзге сұлбалардан кодтық аралыққа қарапайым моделін береді. Ол келесі кодтық комбинацияға ауысу үшін немесе түрлендіру үшін қажетті куб төбесіне дейінгі өтетін кубтың қырларының санына тең болады.



Сурет 1.3 – Код арақашықтығының геометриялық интерпретациясы

Сызықты блокты код құрылу структурасына байланысты анықталады. Құрылу структурасына байланысты мұндай кодтар кодер және декодер құрылысын біршама жеңілдетеді, 1.4 – сурет бойынша, бұл кодтардың теория жүзіндегі әдістерді зерттеу ыңғайлылығы тағы да бір үлкен артықшылығы болып табылады. Қарапайым болып есептелетін жұптылыққа тексеру қағидасына негізделген ең алғаш қолданыс тапқан кодтардың бірі Хэмминг кодының кодері және декодері көрсетілен 1.4 суреттен, кірісіне төрт битті ақпарат беріліп, шығысында код артықшылығы қосылғаннан кейінгі жеті биттік сөздің пайда болғанын көруге болады. Бұл жылжымалы регистрлер көмегімен және қосындылағыш құрылғылардың көмегімен іске асырылады. Кодталған комбинация бірден байланыс арнасына беріледі және жоғарыда айтылғандай тексеруші символдар арнайы разрядтарда орналастырылады. Қателерді тауып түзету код арақашықтығына байланысты жүзеге асады.

Декодтау процесі де разряд бойынша қосу арқылы жүзеге асырылып, код арақашықтығына байланысты қателерді түзету орын алады. Шынайы сұлбада қателерді анықтау кезіндегі синдромдарды сақтау үшін буферлі регистрлер қолданылады. Құрылымдық сұлбадан екінің модулі бойынша қосу опециясының орындалатындығын байқауға болады Декодтау және кодтау осы операциямен байланысты болады.



Сурет 1.4 – Хэмминг кодер (а) және декодер (б) сұлбасы

Жоғарыда айтылғандай кодтық комбинациялардан тұратын кеңістікте n – код ұзындығынан тұратын комбинациялар векторлық кеңістікте жиындарды құрайды. Сызықты блокты кодтар тек барлық n – ұзындықты жиындардың ішкі векторлық кеңістіктің жиындарынан ғана құрылады. Себебі кез келген кодтық комбинацияның разряды келесі разрядпен байланысты болады. Екілік арналар үшін кез келген кодтық комбинациялар ішкі векторлық жиындар болып табылады. Сызықты екілік кодтар үшін топтық немесе блокты код деген атау берілген.

1.2 Сызықты блокты кодтардың математикалық ерекшеліктері

Жақсы параметрлерге ие болатын кодты құру, көп жағдайда таңдап алынған заманауи алгебралық структураға байланысты болады. Көптеген таыблаған маңызды кодтар көпмүшелер шеңберлерінің структурасына және Галуа өрісіне байланысты болады. Сонымен қатар мұндай алгебралық структуралар кодерлер және декодерлерді құрастырғанда пайдаланылатын құрал рөлін атқарады.

Бөгеуілге тұрақты сызықты блокты кодтардың торлары арқылы декодтауды жүргізу барысында көбейту, бөлу, қосу, азайту амалдарын жүргізуге болатын сандар жиыны ұғымы қолданылады. Алайда бұл жиындар қарапайым арифметикалық амалдардың нәтижесі болып табылмайды. Бұл өзіндік ережелері мен заңдылықтары бар өрістің сандар жиынын құрайды. Оларды анықтау үшін келесі анықтамаларды қабылдаймыз:

1) абелев тобы: «қосу» және «азайту» амалын жүргізуге болатын жиын;

2) шеңбер: «қосу», «азайту», «көбейту» амалын жүргізуге болатын жиын;

3) сандар өрісі: «қосу», «азайту», «көбейту», «бөлу» амалын жүргізуге болатын сандар жиыны. Өрістер белгілі бір заңдылыққа бағына отырып, шексіз сандардың жиынын құрайды. Бірақ кодтау теориясы тек шекті саны бар өрістерді қарастырады. q элементтерден ғана тұратын, жиындағы сандар саны шектелген өрісті Галуа өрісі деп атаймыз және келесідей белгілейміз GF(q).

Ең кіші өріс нольдік және бірлік элементтен тұрады, мұндай өріс GF(2) деп аталады және осы өрістен кіші өріс болуы мүмкін емес. Ол келесі заңдылыққа бағынады:

$$\begin{array}{c|cc} + & 0 & 1 \\ \hline 0 & 0 & 1 \\ 1 & 1 & 0 \end{array} \qquad \begin{array}{c|cc} \text{x} & 0 & 1 \\ \hline 0 & 0 & 0 \\ 1 & 0 & 1 \end{array} \tag{1.4}$$

Өріс көмегімен сызықты блокты кодты анықтау жеңіл болады.

Сызықты блокты код дегеніміз GF(q) өрісіне тиісті болатын кіші өрістің мүшесі. Осылайша, сызықты код өрісіндегі элементтер кодтық сөздер болып табылады, ал кез - келген орындалатын амал кодтық сөзді береді. Яғни, кез келген өріс элементінің қосындысы кодтық сөзге тең болады және бірлік элемент пен нольдік элемент те кодтық сөз болады.

Сызықты блокты кодты анықтайтын өріске анықтама бердік, енді өрісті құрайтын G алгебралық тобын анықтайық.

G тобы деп анықталған операцияларды жүргізуге болатын элементтер жиынын айтамыз және бұл топ үшін келесі аксиомалар ақиқат болуы тиіс:

1) кез келген топ элементтеріне операцияны қолдануға болады. Екі элементтің оперциясының нәтижесі үшінші элементті береді;

2) кез келген топқа тиісті элементтер үшін қосу және көбейту амалдарына (a +b) +c =a +(b+c) және (ab)c = (ac)b теңдігі орындалады;

3) топтың бірлік элементі болуы қажет. Егер топтағы операция «қосу» болса бірлік элемент 0 болады, келесі теңдік орындалады 0 + a = 0 + a = a. Ал егер топтағы операция «көбейту» болса, бірлік элемент 1 болады, келесі теңдік орындалады:1a = a1 = a.

4) топтағы әр бір элемент кері элементке ие болады. Егер топтағы операция «қосу» болса, a элементіне кері элемент минус a болады. Ал егер «көбейту» болса, a элементіне кері элемент a^{-1} болып белгіленеді.

Бұл аксиомалардан басқа өріске қатысты да айтылған шарт орындалады: топтағы кез келген элементке тек қана бір кері элемент тиісті болады.

1.3 Бөгеуілге тұрақтылықты арттыру мақсатындағы модуляция процесін сараптау

Бөгеуілдер байланыс арналарында тасымалданатын ақпараттың нақтылығына тигізетін әсері жайында зерттеу жұмыстары аз болмады. Тікелей сапаға әсері болатын бөгеуілдермен күресу жолдары қатарына ақпаратты бөгеуілге тұрақты кодтаумен қатар, модуляция процесін іске асыру жатады.

Модуляция – ақпаратты сигнал заңдылығымен тасымалдаушы сигнал параметрлерінің бірінің өзгерісі. Мұндағы ақпаратты сигнал төменгі жиіліктегі өзгеріс заңдылығын алдын ала болжауға болмайтын сигнал, ал тасушы сигнал бұл – жоғары жиілікті белгілі заңдылығы бар сигнал. Яғни модуляция екі сигналдың қабаттасуының нәтижесі болып табылады. Модуляцияның негізгі мақсаты төменгі жиілікті сигналды физикалық ортада таратуға тиімді болатын жоғары жиілікті сигналға түрлендіру. Сигнал таралатын физикалық орта сымды байланыс жолы болуы мүмкін немесе сымсыз байланыс жүйесі де болуы мүмкін. Модуляция ақпаратты сигналды осы ортада бөгеуілдердің әсерінен қатесіз таралуына икемдейді.

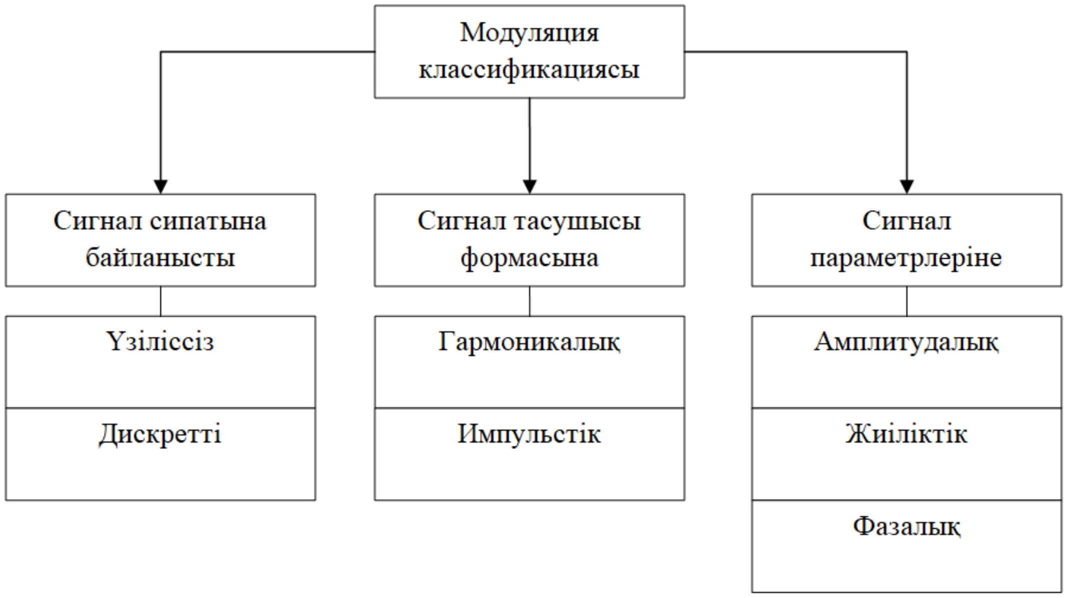
Тасушы сигнал – гармоникалық сигнал және егер синусоидалы формада болса, келесі (1.2) формула арқылы сипаттауға болады:

$$u = U_M sin\Omega t \tag{1.5}$$

Мұндағы, U_M- сигнал амплитудасы;

Ωt- уақыт бірлігіндегі сигнал фазасы.

Модуляцияның іске асырылу тәсілі байланыстың орнатылу ортасына, яғни физикалық ортаға және мақсатына байланысты болады. Модуляция түрлерінің классификациясын 1.5 суреттен көруге болады.



Сурет 1.5 – Модуляция классификациясы

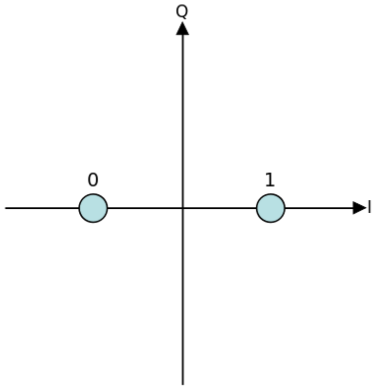
Сигналды қашықтыққа тасымалдаудың жолы келесі сигнал параметрлеріне байланысты бөліп қарастырылады: амплитудасына, фазасына, жиілігіне. Қазіргі таңда байланыс жүйелері қолданылу аясына байланысты модуляцияның өз ортасына тиімді түрін пайдаланады. Сигналға әсері кез келген жүйеде байқалатын шуыл қатарына аддитивті Гаустық шуылдарды жатқызуға болады. Гаусстық шу кездейсоқ әсерінен жүйеде сигналдың жоғалуы, яғни байланыстың үзілуі байқалады. Демодуляция процесі қабылдаушының ақпаратты дұрыс, қатесіз алуының бірінші қадамы.

Цифрлы байланыс жүйелерінде бір символ көмегімен бірнеше бит ақпаратты беруге болатын көппозициялы модуляция қолданылады. Бұл аналогты жүзеге асырыалтын модуляциямен салыстырғанда оңтайлы. Әр символ көппозициялы модуляцияны іске асырғанда 2 биттен бастап 256 немесе одан да көп бит ақпаратты тасымалдай алады. Көппозициялы модуляция негізінде сигналдың синфазалық және квадратуралық құраушылары негізгі орын алады.

$$I(t) = A(t)\cos(\omega t),$$
$$Q(t) = A(t)\sin(\omega t) \qquad (1.6)$$

Мұндағы $I(t)$ – синфазалық құраушы;
$Q(t)$ – квадратуралық құраушы;
$A(t)$ – ақпаратты сигнал.

Мұндай модуляция түріне ең қарапайым екі позициялы фазалық манипуляцияны жатқызуға болады.



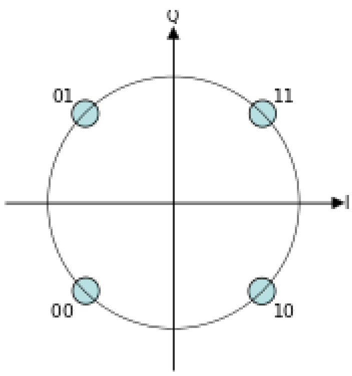
Сурет 1.6 – Екі позициялы фазалық манипуляция

Модуляцияның жұмысы тасушы сигналдың фазасының ығысуна негізделген. Бұл манипуляция түрін квадратуралық манипулцияның дербес жағдайы деп те есептейді. Фазасы ығысқан манипуляцияны сипаттауға болады:

$$s_m(t) = g(t)\cos[2\pi f t + \varphi(t)] \quad (1.7)$$

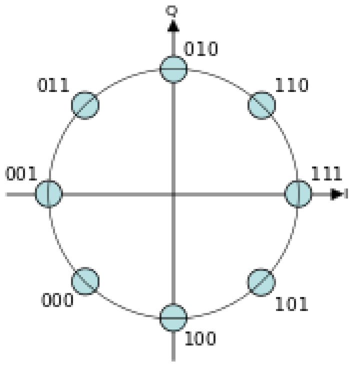
Мұндағы $g(t)$ - сигналды сипаттайды;
$\varphi(t)$ - модуляциялаушы сигнал;
f - тасушы жиілігі.

Модуляциялаушы сигнал M – дискретті шаманы қабылдай алады. Екі позициялы манипуляция үшін M = 2 BPSK, M = 4 болған жағдайда квадраттық фазалық манипуляция QPSK. Әр бір фазаның ығысуы арқылы берілетін бит хабардың саны екілік жүйеге сәйкес екі санының дәрежесі болады. Сәйкесінше төртпозициялы фазалық модуляцияда ақпаратты беру үшін суретте көрсетілгендей тасушы сигналдың фазасының 4 позицияға ығысуы қолданылады.



Сурет 1.7 – Квадраттық фазалық модуляция

Сегіз позициялық 8 – PSK бір символ көмегімен 3 бит ақпаратты бере алады. Яғни, тасушы сигналдың фазасы 1.8 суретте көрсетілгндей сегіз тұрақты фаза көлемінде өзгеріске ұшырайды. Бұл модуляция кезінде биттік жылдамдық үш есе үлкен болады. Суретте көрсетілгендей әр фазаның тұйық позициясына үш бит ақпарат келеді, бұл бөгеуілге тұрақтылықты арттыра түседі. Демек, қабылданған ақпаратта әр битке келетін қателік саны азаяды.



Сурет 1.8 – 8-PSK модуляция

Көппозициялы модуляция қатарына квадраттық амплитудалы модуляцияны немесе QAM модуляцияны жатқызуға болады. Квадраттық амплитудалы модуляция амплитудалы – фазалық модуляция түрі болып табылады. QAM цифрлы телевидение және цифрлы радиохабар таратуда кең қолданыста. 16 позициялы QAM – 16 модуляцияда сигналдың төрт тұрақты күйі болады. Осы арқылы бір символ көмегімен 4 бит ақпарат беріле алады.

Бұл өз кезегінде арнадағы символдық жылдамдықтың биттік жылдамдықтан 4 есе кіші болатындығын көрсетеді, демек мұндай модуляция түрі спектралды түрде алғанда өте тиімді болатын ақпарат тарату жүйесін ұйымдастыра алады. Мысалы, екі позициялы, төрт позициялы өзге көппозициялы модуляция түрлерімен салыстырғанда бұл модуляция түрі арнадағы жылдамдық бойынша тиімдірек болады. Салыстыру кезінде төрт позициялы және 4-QAM бір модуляция түрі екендігін ескерілген. Егер сигнал саны q екінің дәрежесіне тең деп алсақ, онда:

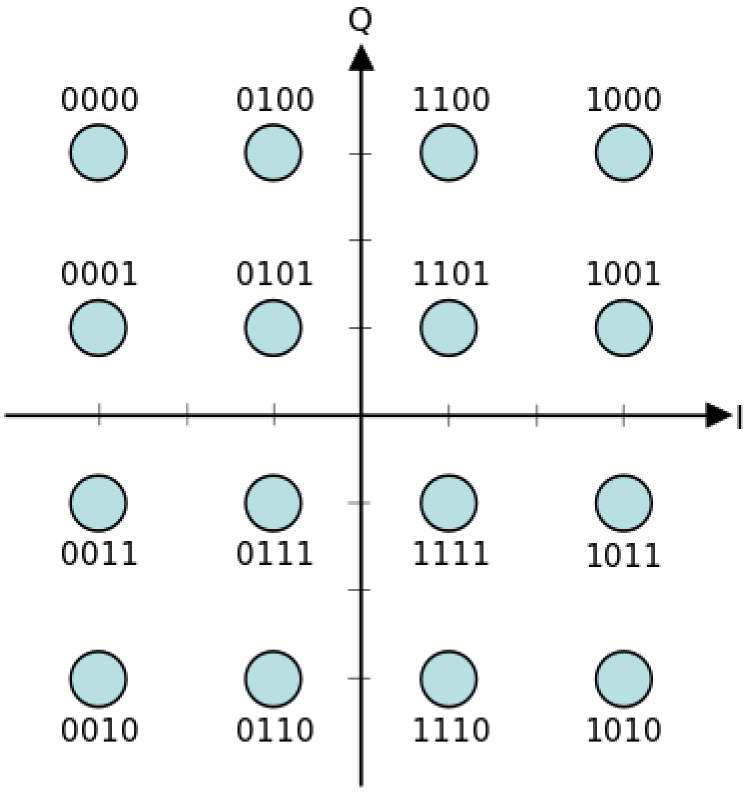
$$\sqrt{q} = 2^k \tag{1.8}$$

Мұндағы k – бүтін натурал сан.

Фазалардың ауытқуы қандай да бір аралыққа тиісті екендігінен, әр бір бит ақпараттың бір бірінен минималды арақашықтығын есептеуге болады. [4]

$$\Delta = \frac{2A}{\sqrt{q} - 1} \tag{1.9}$$

Мұндағы A – сигнал амплитудасы.



Сурет 1.9 – QAM- 16 көрінісі

QAM модуляцияның тағы бір дербес түрі бұл – QAM-32. Оның сипаттамасы келесідей болады. Әр бір позиция 6 сигналдан құралады, демек барлығы мүмкін болатын позиция болғаны, алайда келесі теңсіздікті байқауға болады.[5]

$$36 > 32 \tag{1.10}$$

Бұл мүмкін болатын төрт шеткі позицияны әлсірету керектігін білдіреді.

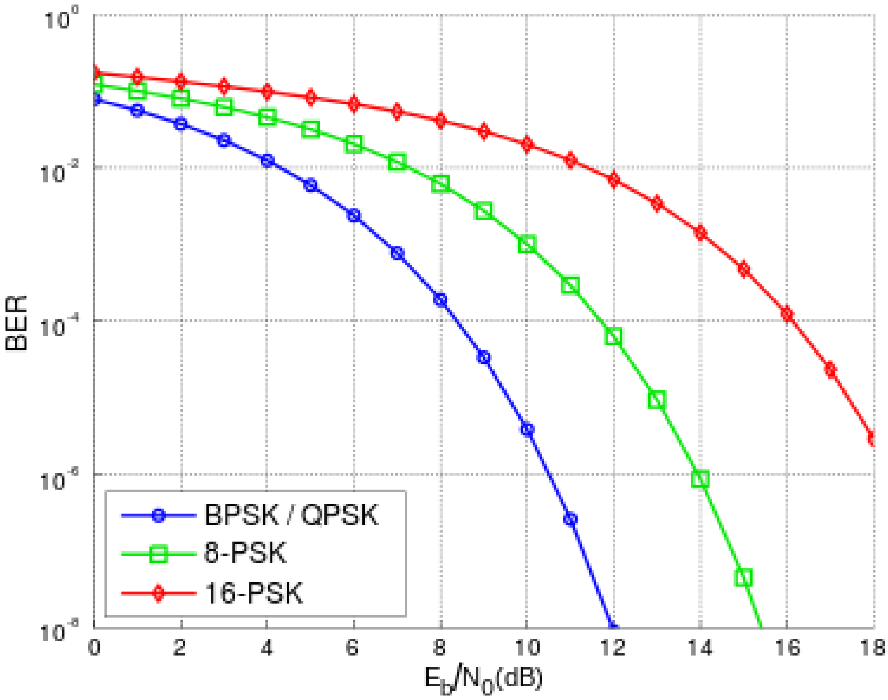
Осылайша, таратқыш туғызатын шығыс сигналдың қуаты төмендейді. Алайда биттік жылдамдық 5 бит/с-қа тең болады. Битке тиісті қателіктің пайда болу ықтималдылығы, егер арнада аддитивтң гаустық шуылдар бар деп есептесек келесі формула бойынша есептеуге болады.

$$P_b = Q\left(\sqrt{\frac{2E_b}{N_0}}\right) \tag{1.11}$$

Мұндағы Q – үлестірімділікке байланысты коэфициент.

$$Q(x) = \frac{1}{\sqrt{2\pi}}\int_x^{\infty} e^{-\frac{t^2}{2}dt} \tag{1.12}$$

Әр символға келетін бит саны қолданылатын модуляцияның түріне байланысты болатындықтан, битке тиісті қателіктің пайда болу ықтималдылығының арнадағы сигнал/шу қатыгасына тәуелділік графигі түрінде көрсетуге болады.



Сурет 1.10 – Қателіктердің пайда болуын салыстыру

Арнада тасушы сигналдың өздігінен фазасының ауысуы байқалса демодулятор бұл қателікті анықтай алмайды. [6]

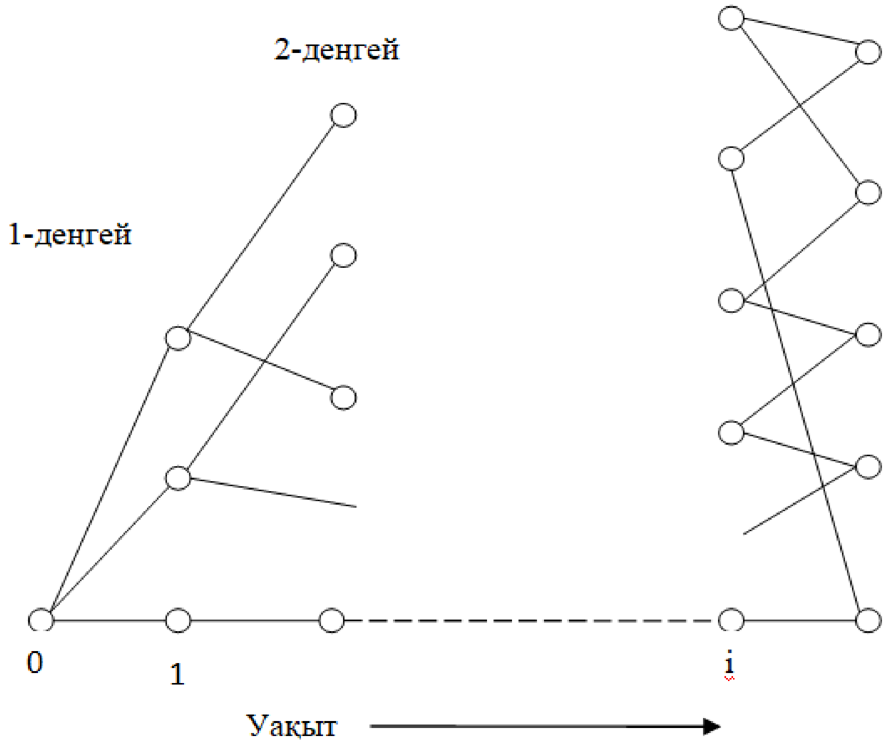
Салыстырмалы түрде талдау жүргізетін болсақ графиктен әр фазасының ауысуы тек бір бит ақпаратты тасымалдайтын BPSK және салыстырмалы түрде бит ақпараты екіге тең болатын QPSK бөгеуілге тұрақтылығы жағынан өзге модуляция түрімен салыстырған төзімді екендігін байқауға болады. Яғни, берілетін ақпарат көлемі аздығынан қабылдау кезінде қателік туындаудың ықтималдылығының төмендігін көрсетеді. Алайда, әр символ тек бір бит ақпаратты тасымалдайтындығынан өзге модуляция түрімен салыстырғанда ақпараттың берілу жылдамдығы төмен болатындығын байқауға болады. Бұл теңдіктен байланыс арнасынан қабылданған амплитудалық сигналға әсер ететін бөгеуілдің аддитивті сипатта болатынын анықтауға болды, алайда сигнал/шу қатынасынан бұл шудың өтпелі әсерге ие болатынын анықтауға болады. Мұндай әсер өтпелі деп аталады. Өтпелі шекарадан сигнал/шу қатынасы асқан кезде аддитивті бөгеуілдің мультипликативті сипатқа ие болатынын байқауға болады. Өтпелі әсерді практикалық жолмен анықтау мүмкіндігі аз, бұл модуляцияланған сигналды демодуляциялау қиындығын және қателердің пайда болуын туғызады. Сигнал/шу қатынасын үлкейту үшін таратқыш қуатын арттыру қажет болады. Бұл шара қосымша құрылғылардың және қосымша шығындардың болуын немесе арналардың өткізу жолағын арттыруды талап етеді.

Модуляция кезіндегі энергия шамасының көп бөлігі тасушы сигналға тиісті болады, бірақ байланыс жүйесі үшін мұндай энергия бөлігі бос жұмсалады. Тасушының қуатын арттыру қателіктердің алдын алу үшін жасалады және сигналдың таралу қашықтығын арттырады [7]. Мысалы, телевизиялық көрсетілімдер кезінде шығындардың көп бөлігі тасушы сигнал қуатына беріледі. Мұндай шарттарды қанағаттандыру үшін заманауи байланыс жүйелері бөгеуілдерден қорғалған кодтармен қатар модуляцияның артықшылықтарын қоса пайдаланады.

2 Сызықты блокты код торын негіздеп сараптау

Код торы ұғымы бағытталған граф болып табылады. Әр бір түйіннің, яғни граф төбелерінің кодтық комбинациямен байланысын алдағы мәліметтермен түсіндіруге болады [8]. Граф өзара байланысқан түйіндерден және сол түйіндерді байланыстыратын бағытталған қабырғалардан тұрады. Қабырғалар граф түйіндерін байланыстырушы, ал екінші өлшем ретінде уақытты алатын болсақ граф түйіндерінің күйінің уақыт өту барысындағы күйлерінің немесе мәндерінің, шамаларының өзгерісін көрсететін тор екендігін байқауға болады. Мұндай граф бір түйіннен тарайды, бағыты тек оңнан солға қарай болып шектелген болуы мүмкін. Бұл заңдылық уақыттың тек бір бағытта тарайтындығына негізделген.

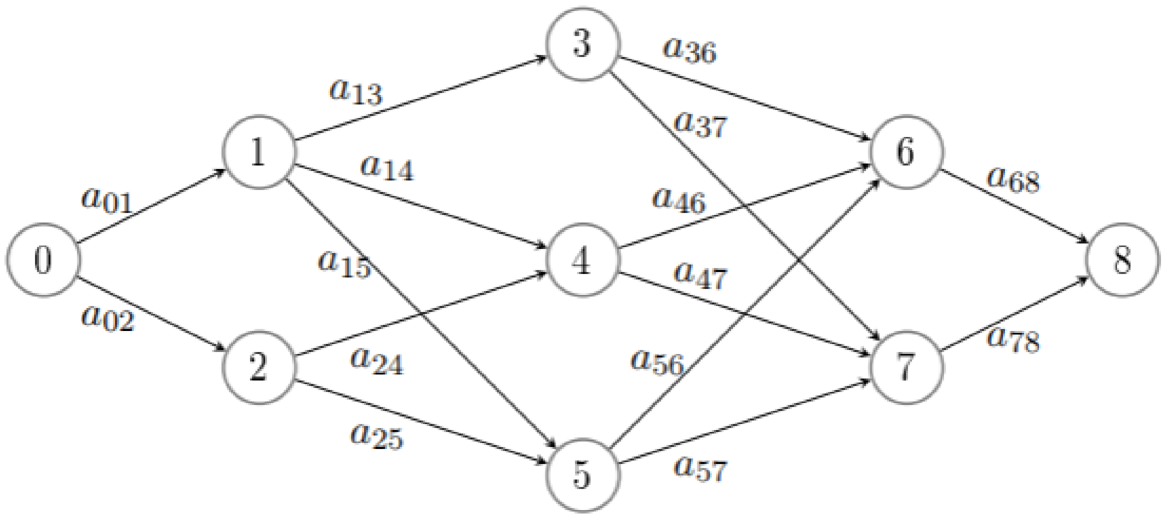
Графтағы әр уақыт мезеті қандай да бір тактқа сәйкес алынады. Әр граф төбесі кодтық комбинациялар үшін бір символ болып есептеледі, демек бір бит ақпарат, екілік жүйе үшін негіздеп жазсақ – «0» немес «1» жағдайы ғана болуы мүмкін.



Сурет 2.1 – Уақыт күйі көрсетілген граф

Логарифмдік шындыққа ұқсастық функциясын кодтық сөздердің қосындысы ретіндегі көрсету бұл шаманы кодтық комбинациялардың ара қашықтығы ретінде немесе сөздің аддитивті метрикасы ретінде қарастыруға болады. Егер кодталған комбинацияны бағытталған граф ретінде көрсете алсақ, декодтау процесы осы графтағы ең қысқа жолды іздеумен ғана шектеледі.

Мұндай тәсілді ең алғаш рет Витерби үйірткілі кодтарды декодтау үшін қолданған, ал сараптаулар арқасында кейін мұндай тәсіл блокты кодтарға да қолданылу мүмкіндігі анықталды.



Сурет 2.1 – Код торының мысалы

Кодтау теориясындағы графты торлы диаграмма немесе тор деп атайды. Кодтың торы – бұл келесі қасиеттерге ие болатын граф:

1. Графтың төбелері өзара қиылыспайтын жиындарға бөлінген, олар деңгейлер деп аталады. Әр бір граф төбесі тек өзге жиынның деңгейімен ғана байланысқан

2. Нөлдік деңгей және соңғы деңгей тек бір ғана түйіннен тұрады.

3. Граф бағытталған және номері төменгі деңгейден жоғары деңгейлі түйінге қарай ғана қозғалады.

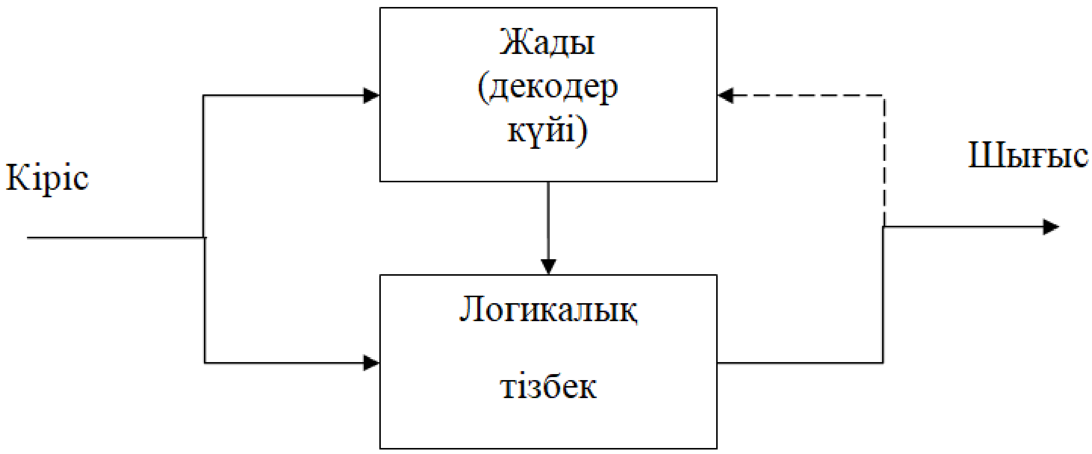
4. Граф қабырғаларына метрикалар жазылған. Метрикалар оң таңбалы немесе теріс таңбалы болуы мүмкін. Жолдың ұзындығы граф қабырғаларының метрикасының қосындысы арқылы табылады.

Осындай қасиеттерге негізделген графта ең қысқа жолды анықтау негізгі мәселе болып табылады.

Негізгі шешім бұл барлық мүмкін болатын жолдарды қарастыру және ең қысқасын тауып алу. Мысалы, келтірілген суретте барлығы 10 жол әр жолдың ұзындығын табу үшін 3 рет қосу амалын орындау қажет, яғни барлығы 30 амалды орындауымыз қажет. Сонымен қатар ең қысқа жолды анықтау үшін 9 мәрте салыстыру және таңдау операциясын орындау қажет.

Негізгі емес әдіс бұл бағдарламалау арқылы өзге алгоритм көмегімен таңдау. Яғни, бірнеше жолдар біріккен түйіндерде ең тиімді жолды таңдап метрикасы нашар жолды қарастырмауға да болады. Қарастырылған мысалда бұл 4, 5, 6, 7 түйіндері. Ең тиімді жол тек 8 түйінде ғана анықталады.

Мысалы, 5 түйінде екі жол ұзындығы салыстырылады: $a_{01} + a_{15}$ және $a_{02} + a_{25}$. Келесі деңгейлерде екі жолға да бірдей шамадағы метрика қосылады, сондықтан тиімділікті жоғалтпай екі жолдың бірін таңдауға болады. Негізгі әдіспен салыстырғанда, мұнда 13 қосу амалы және 7 салыстыру және таңдау амалы орындалады.



Сурет 2.2 – Қабылдағыш құрылғы құрылымы

Ақырғы қабылдаушы құрылғы құрылымы 2.2 сурет бойынша қолданылатын кодтың құраушы полиномына байланысты болады. Қабылданған кодтық комбинациялар қателері буфферлік регистрлер көмегімен анықталады. Суретте бейнеленген ақырғы құрылғы жұмысы логикалық тізбек арқылы қабылданған кодтық комбинацияларды декодтау процесін іске асырады. Кері байланыс арқылы қабылданған кодтық комбинациялардың синдромдары декодерлердің күйіне сақталады, яғни жады да сақталады. Бұл уақытша сақталатын мәліметтер болып табылады. Мұндай декодер жылжымалы регистрлер көмегімен іске асады, ал декодтаудың тиімділігін арттыру үшін сызықты блокты кодтар үшін тор көмегімен іске асыруға болады. Яғни тиімді жолды іздеу арқылы кодтық комбинацияның шындыққа жанасқан декодтау процесін іске асырамыз. Бұл өз кезегінде көп жиындар ішінен кодтық комбинацияға тиісті болатын жолды іздеу амалына алып келеді. Мұндай декодтау үйірткілі кодтар үшін тиімді пайдаланылып келеді.

Бұл мәселені шешу үшін кодтау теориясында Витерби алгоритмын қолданады. Витерби алгоритмы графтағы қысқа жолды іздеу кезіндегі есептеудің тиімді ережесін қалыптастырады.

Алгоритмды қолдану үшін сызықты блокты кодтың торын тұрғызып, графтың қабырғаларын кодтық сөздермен байланыстырамыз. Кодтық комбинацияларды декодер қабылдағанда кодтық комбинацияларды олардың логарифмдік шындыққа ұқсастық функциясымен алмастырады. Егер декодтау Хэммингтік арақашықтық арқылы жүзеге асатын болсаң, онда кодтың торы кодтық комбинациялардың Хэмминг арақашықтығы бойынша құрылады және канал шығысында символмен жазылады, ал гаусстық арна үшін шығыс арнадағы кодтық комбинация сол шама қалпында жазылады. Әр бір арна моделі өзіндік қасиетке ие болатындықтан, дипломдық жұмыста байланыс арнасы жалпы жағдайда қабылданады.

Витерби алгоритмы келесі қадамдарға жіктеп көрсетейік:

1. Әр бір жолдың метрикасын табамыз, метрика әр бір алдыңғы жолдардың қосындысы ретінде табылады.

2. Барлық табылған жолдардың метрикаларын салыстырамыз және ең кіші жолды анықтаймыз.

3. Түйінге баратын жол қабырғаның ұзындығы арқылы беріледі және түйінге баратын кез келген жол осы қабырғалар арқылы баратын жолға тең болады.

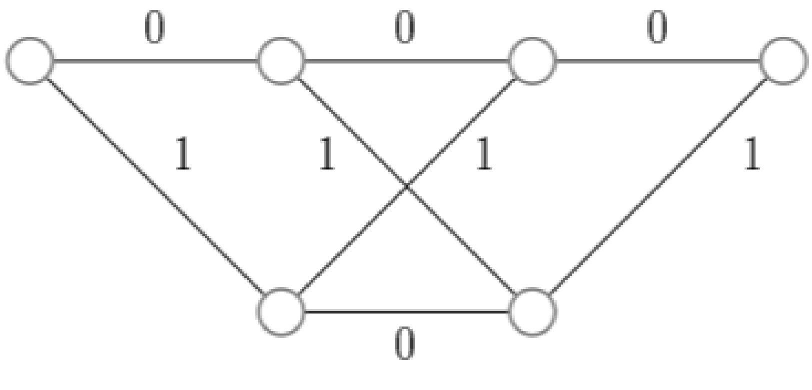
2.1 Кодтың ең кіші торы туралы түсінік

Сызықты блокты кодтың кодтық сөздерінің жиынын қарастырайық. Жиын үшін кодтың ең кіші торын құрастырамыз. Кодтың торында қабырғаларға кодтық символдың шамасы жазылады. Бірінші әр бір қабырғаға тек бір символ сәйкес келетін мысалды қарастырамыз. Алайда әр қабарғаға блокты кодтың комбинациясын жазуға болатын тор бар, мұндай торды секцияландырылған деп аталады. Секцияландырылмаған торда барлығы n + 1 деңгей болады.

Келесі құраушы матрицамен берілген (3, 2) кодын қарастырайық:

$$G = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix} \qquad (2.1)$$

Келесі 2.3 суретте тордың бағыттары көрсетілмейді, граф шартты түрде солдан оңға қарай бағытталған деп есептеледі. Бұл код үшін келесі тор тиісті болады:



Сурет 2.3 – (3, 2) кодының торы

Декодер қабылданатын комбинациялар реттілігін біліп, символдарды логарифмдік шындық функциясымен алмастырып жазсақ болады. Яғни, жоғарыда айтылғандай Витерби алгоритмын қолдану арқылы декодтау процесын ең қысқа жолды іздеу арқылы шешуге болады.

Декодтау күрделілігі тордағы түйіндердің санымен анықталады. Демек, декодерды құрастырудың тиімділігі кодтың торындағы түйін санының ең аз болуымен сипатталады. Яғни, ең кіші кодтық торды тұрғызумен шешіледі.

Жалпы жағдайда бір код үшін күрделілігі әр түрлі бірнеше торды тұрғызуға болады. Тордың күрделілігі $\xi = (\xi_0 \ldots \xi_i)$ тордағы түйін санымен анықталатын, күрделілік профилымен анықталады.

Тор ең кіші болып есептеледі, егер кез келген тормен салыстырғанда таңдап алынған тор үшін $\xi = (\xi_0 \ldots \xi_i)$ кіші болса. Яғни, теңсіздік $\xi_i > \xi$ орындалса. Ең кіші торға қойылатын негізгі талап тұрғызылатын торлардың ішіндегі оңайы болуы керек. Тек осы талап арқылы декодтаудың оңай жолын анықтауға болады.

Код торына қатысты жоғарғы және төменгі индекстер ұғымын енгіземіз. Жоғарғы индекстер түйіннің графтағы жолына қатыстылығын білдіреді.

Бұл ұғымдар келесідей түсіндіріледі: бір түйінге енетін жолдар «ортақ болашаққа» ие болады, ал бір түйіннен шығатын жолдар «ортақ өткен шаққа» ие болады. Сәйкесінше, жоғары индекстер «p» және «f» - өткен шақ «past» және болашақ «future». Мұндай параметр кез келген торда кездеседі. Келесі қандай да бір c^p үшін жиынды қарастырайық:

$$F_i = \{c^f : (c^p, c^f)\} \in C \tag{2.2}$$

Бұл жиынның тиістілігінен әр бір c^p жиынына бір немесе бірнеше c^f тиісті болуы мүмкін. Демек, F_i жиынын келесі жиындарға бөлуге болады, $F_i(c^p)=\{c^f : (c^p, c^f) \in C\}$. Келесі құраушы матрицаны қарастырайық:

$$G = \begin{pmatrix} 1\,0\,1\,1\,0\,1 \\ 1\,0\,1\,0\,1\,0 \\ 1\,1\,0\,1\,0\,0 \end{pmatrix} \tag{2.3}$$

Код келесі кодтық сөздерден құралады:

$c_0 = (000000)$ $c_1 = (110100)$ $c_2 = (101010)$ $c_3 = (011110)$

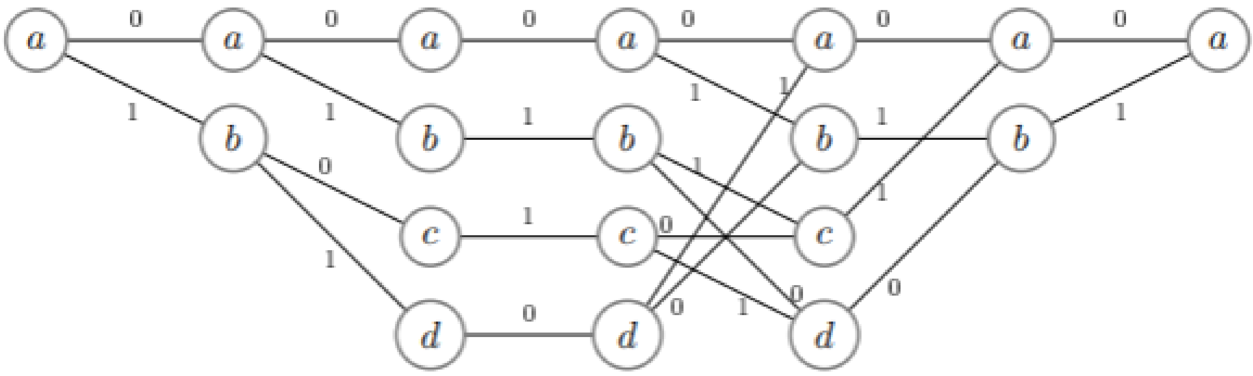
$c_4 = (101101)$ $c_5 = (011001)$ $c_6 = (000111)$ $c_7 = (110011)$

Кестеде барлық деңгейлердегі түйіндер жиыны берілген. Әр бір жиынға берілген деңгейге сәйкес барлық түйіндер берілген. Түйіндер өсу реті бойынша әріптермен белгіленген. Тордың күрделілік профилі осы тор үшін (1, 2, 4, 4, 4, 2, 1) күрделілік профилі анықталған.

Бұл тұрғызылған кодтың торы ең кіші екендігіне көз жеткізуге болады. Ол үшін осы кодқа тиісті болатын кез келген бір торын тұрғызуға болады. Тиісінше кодтың торын салыстыратын болсақ, онда тордағы деңгейлердің санының айырмашылық жасайтынын байқауға болады. Сәйкесінше, торды граф деп есептесек мұндағы жолдар саны көп екендігін байқауға болады. Кодтық сөздердің саны көп болған сайын, оның күрделілігінің артатындығымен байланысты.

Кесте 1.1 – Ең кіші торды құрастыру

i	c^p	$F_i(c^p)$	v_i	v_{i-1}	c_i
0	0	$\{c_m, m = 0, \ldots, 7\}$			
1	0	00000,11110,11001,00111	a	a	0
	1	10100,01010,01101,10011	b	a	1
2	00	0000,0111	a	a	0
	01	1110,1001	b	a	1
	10	1010,1101	c	b	0
	11	0100,0011	d	b	1
3	000	000,111	a	a	0
	011	110,001	b	b	1
	101	010,101	c	c	1
	110	100,011	d	d	0
4	0000,1101	00	a	a,d	0,1
	0001,1100	11	b	a,d	1,0
	0111,1010	10	c	b,c	1,0
	0110,1011	01	d	b,c	0,1
5	00000,11010,10101,01111	0	a	a,c	0,1
	10110,01100,00011,11001	1	b	b,d	1,0
6	$\{c_m, m = 0, \ldots, 7\}$		a	a,b	0,1



Сурет 2.4 – Кестеге сәйкес кодтың торы

Кез келген код ең кіші торға ие болады. Барлық кіші торлардағы деңгейлердің түйіндерінің нумерациясы мен орналасуы сәйкес болады. [11]

Яғни, кіші торды құрастыру үшін өткен шағы бірдей жолдарды біріктіру жеткілікті болады. Дәл сол секілді тордың бағытын өзгертіп, болашағы ортақ болатын жолдарды біріктіру арқылы кіші торды құрастыруға болады.

Қорытындылап, егер әр бір деңгейде барлық жолдарға ортақ болашақ және өткен шақ болса және бұл жолдар тордың бір түйінінен өтсе, онда осы код үшін бұл тор ең кіші тор болып есептеледі.

2.2 Құраушы матрица арқылы код торын тұрғызу

Жоғарыда қарастырылған кодты тұрғызу принципі барлық кодтық сөздерінің анализін қажет етеді. Демек, код торын құрастыру қиындығы кодтық сөздер санына қарай пропорционалды түрде өседі. Бұл шама кодтың ұзындығына экспоненционалды түрде байланыста болады. Сызықты кодтар үшін бұл принциптен құраушы матрица және түзеуші матрица арқылы жүзеге асыруға болады.

Сызықты блокты код $x = (x_1, \dots, x_n)$ реттілікпен берілген деп есептесек, қабылданатын кодтық комбинация басы болып $b(x)$ бірінші нольдік емес элемент есептеледі, соңғы элемент болып соңғы нольдік емес элемент $e(x)$ есептеледі. Осы позициядағы элементтерді $b(x), \dots, e(x) - 1$ активті элементтер деп атайық. Ал барлық активті элементтер санын спэн деп атаймыз.

$x = (0,0,1,0,1,1,1,0,0,0)$ қабылданған коды үшін $b(x) = 3$, $e(x) = 7$, спэн 4 – ке тең, үшіншіден алтыншы позицияға дейін. [12]

Құраушы матрица минималды спэнге келтірілген болып есептеледі, егер барлық жолдардың басы және аяғы әр түрлі болса. Ыңғайлылық үшін матрица жолдары өсу ретімен орналасады. Матрицаны реттелген түрге келтіруді қарастырайық.

Құраушы матрицаны минималды спэнды түрге келтіруге болады:

$$G = \begin{pmatrix} 1\,0\,1\,1\,0\,1 \\ 1\,0\,1\,0\,1\,0 \\ 1\,1\,0\,1\,0\,0 \end{pmatrix} \tag{2.4}$$

Ол үшін бірінші жолды екінші және үшінші жолға қосып, екінші және үшінші жолдардың орнын ауыстырамыз. Келесі матрицаны аламыз:

$$G = \begin{pmatrix} 1\,0\,1\,1\,0\,1 \\ 0\,1\,1\,0\,0\,1 \\ 0\,0\,0\,1\,1\,1 \end{pmatrix} \tag{2.5}$$

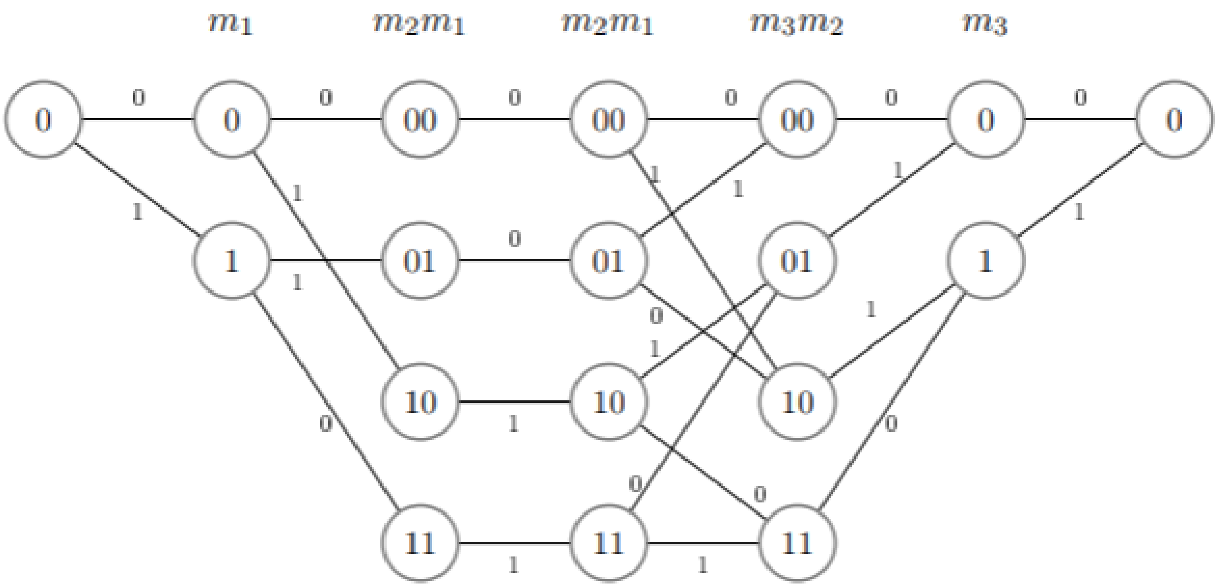
Жолдар басы бойынша реттелген. Яғни, қарастырылып отырған матрица бірлік матрица болатын қалыпты жағдайға ұмтылады немесе толықтай бірлік матрица болады.

Ендігі кезекте екінші жолды бірінші жолға қосамыз, соңғы жолды екінші жолға қосамыз. Минималды спэнды матрицаны аламыз:

$$G = \begin{pmatrix} 1\,1\,0\,1\,0\,0 \\ 0\,1\,1\,1\,1\,0 \\ 0\,0\,0\,1\,1\,1 \end{pmatrix} \qquad (2.6)$$

Нөлдік деңгейде тек бір бастапқы түйін орналасады. Бірінші деңгейде түйіннің номері қабылданған символға байланысты болады. Яғни, бірінші қабылданған символ 0 болса, онда бірінші деңгейдегі бірінші түйін номері 0 болып есептеледі. Ал бір түйіннен екінші деңгейдің түйініне байланыстыратын жол екінші болып қабылданатын символмен белгіленеді.

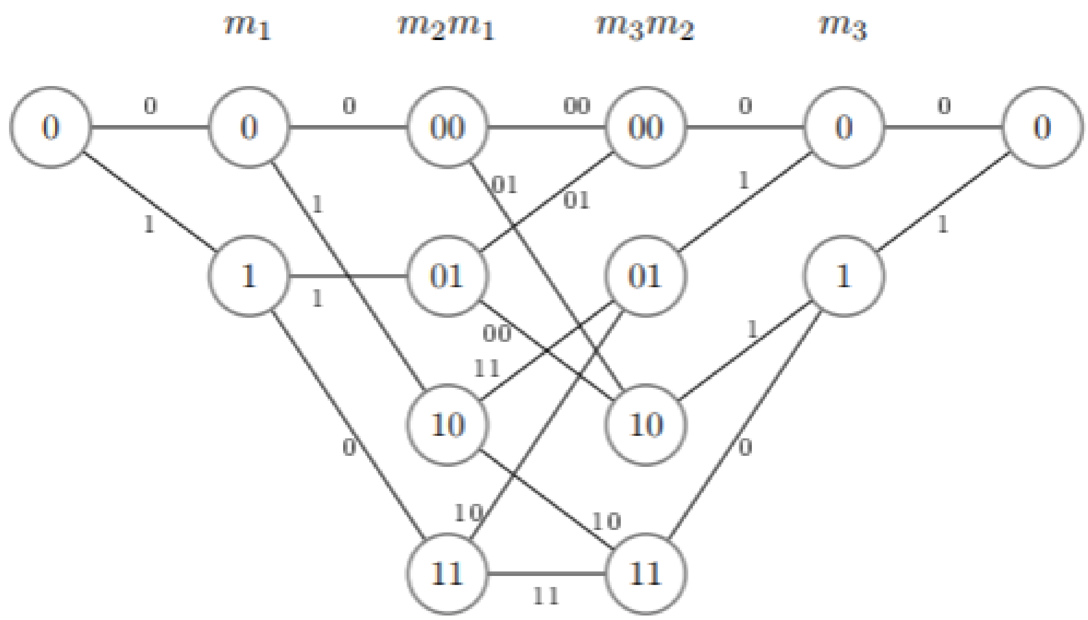
Тордың төртінші деңгейінің құрастырылу принципін қарайық: осы деңгейге дейін түйін номері ақпаратты символдардың m_2, m_1 символдарына сәйкес келеді. Келесі деңгейдің түйініне m_3, m_2 символдарына сәйкес келеді. Егер m_2=1, m_1=0 болса алдыңғы жағдайда (1, 0). Онда m_3=1 болғанда келесі ауысу жүзеге асады (1, 1) және тор қабырғаларына кодтық комбинацияның төртінші мүшесіне сәйкес номер жазылады. Яғни құраушы матрицаның төртінші бағанына сәйкес келеді. (1, 0)-ден ауысу (0, 1) орын алған кезде ақпаратты символдар m_1=0, m_2=1, m_3=0 сәйкес көбейту амалдарын орындағаннан кейін баған $(111)^T$ болады, ал қосу амалын орындағаннан кейін кодтық символ 1 болады. Сол секілді (1, 1) ауысуы жүзеге асқан кезде ақпаратты символдар m_1=0, m_2=1, m_3=1 құраушы $(011)(111)^T = 0$ элементіне сәйкес келеді. [13]



Сурет 2.5 – Құраушы матрица арқылы құрылған тор

Тордың күрделілік профилі (1, 2, 4, 4, 4, 2, 1), бұл профиль қарастырылған құраушы матрица үшін ең тиімді болып табылады. Ең кіші торды алу үшін екінші және үшінші деңгейді қосуға болады. Ондай жағдайда тордың күрделілік профилі (1, 2, 4, 4, 2, 1) болады. Мұндай операция секциаландыру деп аталады.

Көп жағдайда осындай операцияны орындау арқылы тордың күрделілік профилін екі есе кішірейтуге болады және соның арқасында тордағы деңгейлер мен түйіндердің санын азайтуға болады.



Сурет 2.6 – Секциаландырылған кодтың торы

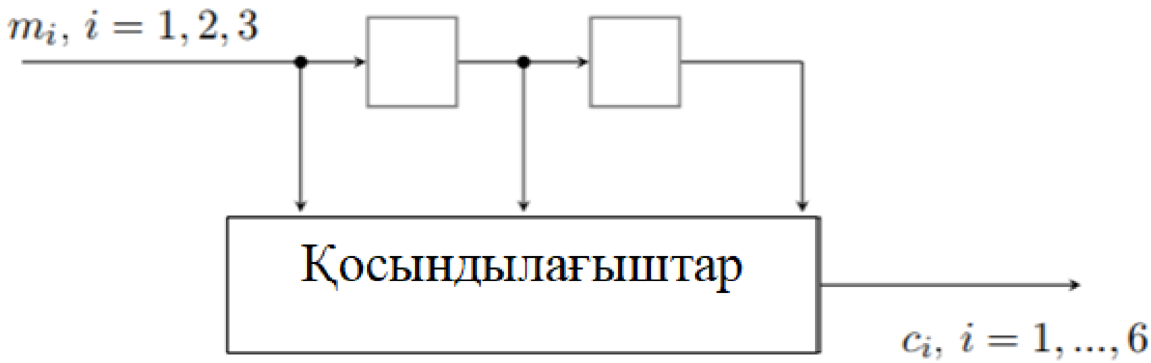
Тордағы түйіндердің номерлері ақпараттық символдардың шамасымен анықталады. Олар активты бағандар арқылы жүзеге асады. Ауысуларға сәйкес келетін символдар құраушы матрицадағы сызықты жүйе ретінде қарастырылып анықталады.

Құраушы матрица арқылы құрастырылған тордың ең кіші екендігін анықтадық. Ол үшін келесі тұжырымды қарастырамыз. C^f ортақ болатын жолдардың қандай да бір деңгейде тек бір түйіннен өтетіндігін көруге болады. Ал бұл өз кезегінде ең кіші тордың негізгі шарты болып табылады.

Қабылданатын символдар бір түйіннен l деңгейі арқылы өтеді деп есептесек, бұл символдардың шамасы осы деңгейдегі активты жолдарға сәйкес келеді. [14]

Егер осындай активты жолдардан қабылданған екі бірдей символдар бар болса және олар сәйкесінше бір деңгейдегі бірдей түйіндерден өтіп жататын болса, онда бұл символдардың қосындысы l деңгейінде активты болатын сөзді тудырады, және бұл сөз l деңгейінен үлкен деңгейде 0 – ге тең болады. Мұндай бірдей сөздер жоқ болатындықтан, мұндай тор минималды болады.

Жоғарыда құрылған торлар келесі интерпритацияны көрсете алады. Тор арқылы декодер немесе кодер құрылысын анықтауға болады. Тордың түйіндерін регистрдың жағдайына байланысты жазуға болады. Регистрге символдар реттілікпен беріледі деген шартпен, құраушы матрица арқылы құрылған кодтың торын сараптайық. Бастапқыда кодерде нольдер жазылады.



Сурет 2.7 – Код торының интерпритациясы арқылы құрылған кодер сұлбасы

Ары қарай әр бір такт өткен сайын, регистр кірісіне ақпаратты символдар беріледі, ал қосындылағыштар модуль бойынша қосу амалын орындап сәйкес кодтық сөздердің символдарын анықтайды. Код торының байланысы регистрдің бір күйден екінші күйге ауысуын дәл көрсетеді, ал тордағы жолдар толығымен регистр ұяшықтары мен қосындылағыштың байланысын көрсетеді. Мұндай интерпретация блоктық және үйірткілі кодтарының арасындағы байланысты түсіну үшін маңызды және үйірткілі кодтарынан тордың күрделілігі аз болатын блокты кодтарды алу мүмкіндігін береді.

2.3 Тексеруші матрица арқылы құрылған кодтың торы

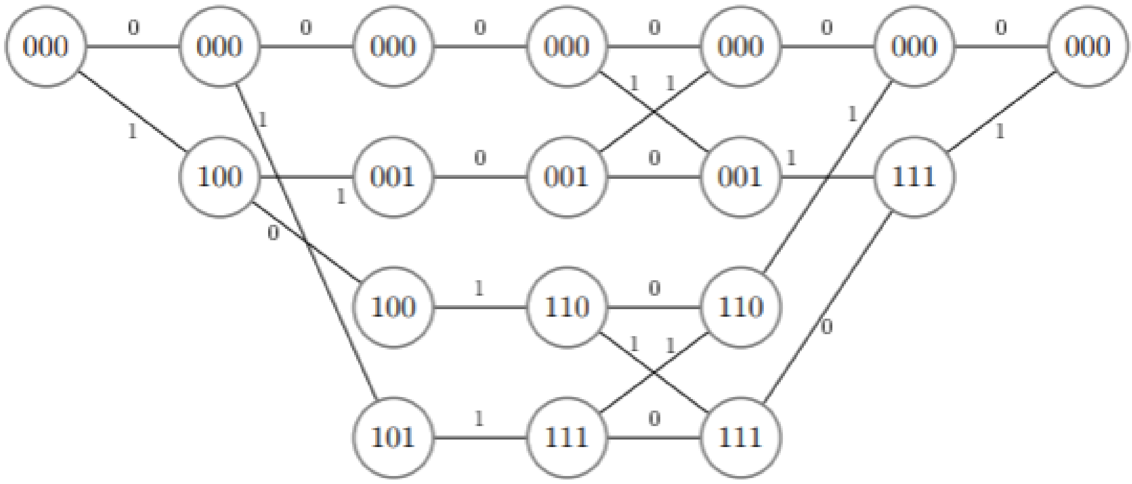
Құраушы матрицаға қарағанда тексеруші матрица арқылы жасалатын кодтың торы жоғары жылдамдықты кодтар үшін ыңғайлырақ болады. Мұндай жолмен құрылған тордың декодтау процесі тексеруші матрицаның параметрлерімен байланысты болады. Сол себепті, мұндай тор жеңіл болуы қажет. [15]

Ең кіші тордың шартына байланысты деңгейлердегі түйіндердің нумерациясы бірдей болады. Жалпы кодтау теориясының бастамасы торлар үшін синдромды матрицалар арқылы жүзеге асты. Бұл кодтардың жылдамдығы ½. Тек бірнеше зерттеулердің қорытындыларынан кейін сызықты блокты кодтардың торлар арқылы берілуі кодтардың декодталуын жеңілдете түседі.[16]

Құраушы матрица арқылы жүзеге асыру мысалын қарастырғаннан кейін тексеруші матрица көмегімен де құрастырылатын кодтың торын қарастырып сараптаймыз. Берілген тексеруші матрица:

$$H=\begin{pmatrix}1\ 1\ 0\ 0\ 1\ 1\\0\ 0\ 1\ 0\ 1\ 1\\0\ 1\ 0\ 1\ 0\ 1\end{pmatrix} \qquad (2.7)$$

Матрица арқылы кодтың торын құрастыру құраушы матрица арқылы жүзеге асатын код торына ұқсас болады. Алайда бұл жағдайда кодтың торының қабырғаларына және сәйкесінше түйіндеріне қабылданатын кодтық комбинациялардың синдромдары жазылады.



Сурет 2.8 – Түзетуші матрица арқылы құрылған кодтың торы

Алғашқы нольдік деңгейдегі түйінге нөлдік синдром тиісті болады. Бірінші деңгейге ұзындығы 1 болатын келесі синдромдар келесі жиындар векторының біріне тиісті болуы мүмкін: {(0,0,0), (1,0,0)}. Ұзындығы 2 болатын синдром келесі деңгейдегі түйіндерге сәйкес келеді және келесі вектор жиынынан тұрады: {(0,0,0), (1,0,0), (1,0,1), (0,0,1)}. Торды құрастыру кезінде әр бір келесі деңгейде түйіндер саны екі еселеніп отырады. Бұл заңдылықты тек бастапқы торды құрастыру кезінде байқауға болады. Синдромдар саны $2^{n-k} = 2^{r} = 8$, бұл шарттан синдром саны асып кетсе кодтық комбинациялары бірдей болатын синдромдар пайда бола бастайды. Осыған байланысты барлық деңгейдегі түйіндер саны шектелген.

Синдромды кодтың торы әр дайым минималды немесе ең кіші болады. Бұл тұжырымға көз жеткізу үшін бірдей реттілікпен c^{f} әр дайым тек қана бір түйіннен өтетіндігіне көз жеткізсек болғаны. Кодтық сөздердің бір түйіннен шығатындығын немесе бір түйінге ұмтылатындығын ескерсек бұл тұжырымның синдромды тор үшін дұрыс екендігіне көз жеткіземіз.[17]

Түрлі жолдармен құрылған кодтың торлары әр деңгейдегі түйіндердің номерлері бойынша бірдей болады, ал эквивалентті кодтар үшін құрылған торлар өзара әр түрлі болады. Басты себебі кодтық сөздердің әр түрлі реттілікте болуы, яғни кодтық комбинациялардың бір бірінен айырмашылық жасауында. Кодтардың негізінде құрылатын торлар үшін жоғарғы қиындықтың шегі болады:

$$\xi\text{max} = \max i\ \xi i \leq \min\{2\ k\ ,\ 2\ n{-}k\ \}. \qquad (2.8)$$

Бұл шарттың басты мағынасы: тордың деңгейлеріндегі мүмкін болатын түйіндер санын шектеу. Алайда осындай шартты қанағаттандыратын кодтар құрастыру үлкен қиындық туғызады, себебі тор бойынша декодтау процесіне қатесіз жіберілетін кодты құру қиындығы үлкен және ол код үшін ең кіші торды анықтау процесі де, ол торды іздеу алгоритмдері де қиындық туғызады.

2.4 Балл Кук Джелинека Равив алгоритмы

Кодтық сөздердің дұрыс қабылдануы арқылы нақтылығын арттыру шарасына шартты ықтималдықтармен декодтау тәсілі жатады. Бұл әдіспен декодтау процесы жоғары шынайылықпен декодтау тәсіліне қарағанда тиімдірек екендігін көрсетеді. Қабылданатын хабар ретінде ақпараттық символдарды қарастыруға болады. Мұндай жағдайда шартты ықтималдықпен декодтау тәсілі қабылданатын әр бір символ үшін жеке жағдай қарастырады, ал жоғары шынайылықпен декодтау тәсілі кодтық сөздер толық қабылданған сәтте ғана толық кодтық комбинация жайында ақпарат бере алады. Сол себепті әр бит ақпарат үшін декодтау процесы қателік жіберу ықтималдығын біршама төмен екендігін көрсетеді.

Шартты ықтималдықпен декодтау артықшылығы кодтың торын құру арқылы пайдалану кезінде арта түседі. Бұл каскадты кодтардың пайда болуынан кейін кең қолданыс тапқан алгоритм Балл Кукк Джелинека Равив алгоритмы болды. Каскадты кодтардың қолданыс табуы алгоритмның негізгі қолданыс табуына себеп болды. [18]

Екі немесе одан да көп кодер арқылы бірдей ақпаратты символдар кодталады. Байланыс арнасы арқылы берілгеннен кейін, демодулятордан шыққан ақпарат декодер кірісіне беріледі. Декодтау процесінен кейінгі нәтиже екінші бір декодер кірісіне беріледі. Екінші декодердің нәтижесі қайтадан бірінші декодер кірісіне беріледі. Осындай операция орындалғаннан кейін шартты түрде ақпаратты символдардағы қателер толық жойылғанға дейін итерация жүргізіледі, демек операция қажетті шартты ықтималдылықта код қателіктері жойылғанға дейін қайталанады. Әдетте мұндай операциялар саны онға жетеді.

Шартты қабылданған ықтималдық бойынша декодтау тиімді екендігін біле отыра, осындай итеративті сұлбадан тұратын декодтау процесіне қатысатын әр бір декодер қабылданған символдар бойынша тек қана символдарға қатысты шешім шығармай, ол қабылданған шешімнің қаншалықты нақты екендігіне баға беруі қажет.

Осындай шартты қанағаттандыратын декодер SISO деп аталады, және осындай шартты шешетін БКДР декодерлері де жүзеге асырылады.

Екілік сызықты блокты кодты қарастырайық $C = \{c_m, m = 1, \dots, M\} \in\in \{0, 1\}$. Бұл кодқа тиісті кодтың ең кіші торы бар деп есептейік. Декодер кірісіне келесі реттіліктегі хабар келеді:

$$y = (y_1, \dots, y_n) \in Y^n \quad (2.9)$$

Қабылданатын символ үшін шартты ықтималдық келесі түрде жазылады:

$$p(c_t = c|\boldsymbol{y}) = \frac{p(c_t = c, \boldsymbol{y})}{p(\boldsymbol{y})}, \quad (2.10)$$

$$p(c_t = c, \boldsymbol{y}) = \sum_{\boldsymbol{c} \in C_t(c)} p(\boldsymbol{c}, \boldsymbol{y}), \quad (2.11)$$

Кодтағы кодтық сөздердің күрделілігі алгоритмы жүзеге асыру күрделілігімен тура пропорционал.

Бұл жағдайда кодты тор түрінде көрсету алгоритмды қолдану жұмысын жеңілдетеді. Витерби алгоритмі секілді, БКДР алгоритмы үшін қабылданған символдарға қатысты орын алған тордың активты бөлігіндегі есептеулерді ең тиімді жолды табу арқылы есептеу. Кодтық сөздерді қосу операциясын тор бойынша жиындарды қосу операциясымен алмастырамыз.

Ең алдымен шартты ықтималдықты есептейміз:

$$\Pr\left(s_{t-1} = m', s_t = m|\boldsymbol{y}\right) = \frac{\Pr\left(s_{t-1} = m', s_t = m, \boldsymbol{y}\right)}{p\left(\boldsymbol{y}\right)}. \quad (2.12)$$

Мұндағы Pr – шартты ықтималдылық.

Орындалатын функцияны келесі σ_t- шартты функциямен алмастырамыз:

$$\sigma_t(m', m) = \Pr(s_{t-1} = m', s_t = m, \boldsymbol{y}). \quad (2.13)$$

Қабылданатын символдар үшін шартты ықтималдықты бір ғана формула көмегімен табуға болатындығына көзіміз жетті, келесі кезекте символдардың қабылдану реттілігіне байланысты ортақ болашағы және өткен шағы бар символдарды тордан анықтап келесі заңдылықты аламыз:

$$\sigma_t(m', m) = \Pr\left((s_{t-1} = m', \boldsymbol{y}_1^{t-1}), (s_t = m, y_t), (\boldsymbol{y}_{t+1}^n)\right). \quad (2.14)$$

Бұл символдар ортақ жолда ортақ түйіндерден өтетіндігін ескеріп, символдардың белгілі бір кодтық сөздерді қабылдаған уақытта өзара тәуелді екендігін анықтауға болады. Ал арнаның жадысы жоқ деп және қабылданған кодтық комбинация белгілі десек, онда символдар шартты түрде өзара тәуелсіз күйде болады. [19]

Кодтың торы белгілі болған жағдай с символдар жиыны үшін ортақ өткен шағы бар жолдарда орналасқан символдар, ендігі кезекте өзара тәуелсіз қабылданады. Сондықтан ықтималдық келесі түрде болады:

$$\Pr(A_1) = \Pr(s_{t-1} = m', \boldsymbol{y}_1^{t-1});$$

$$\Pr(A_2|A_1) = \Pr(s_t = m, y_t|s_{t-1} = m', \boldsymbol{y}_1^{t-1}) = \Pr(s_t = m, y_t|s_{t-1} = m') \quad (2.15)$$

$$\Pr(A_3|A_1A_2) = \Pr(\boldsymbol{y}_{t+1}^n|s_{t-1} = m', s_t = m, \boldsymbol{y}_1^t) = \Pr(\boldsymbol{y}_{t+1}^n|s_t = m).$$

Келесідей белгілеулерді енгізсек, ықтималдық анықтау ыңғайлы түрге ие болады:

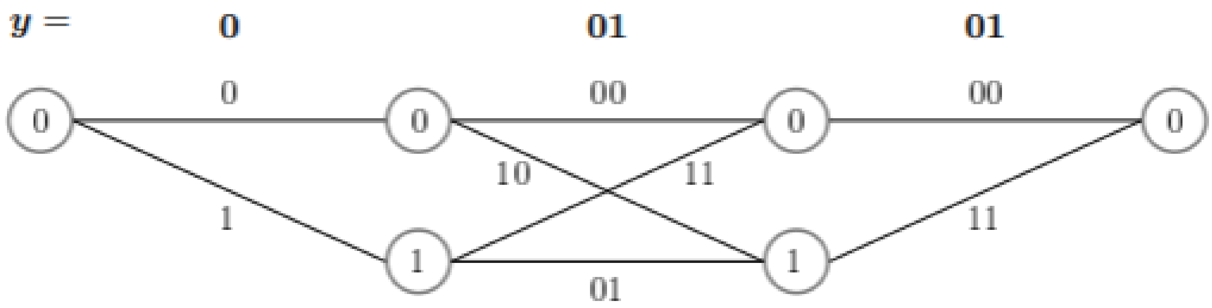
$$\begin{aligned} \alpha_t(m) &= \Pr(A_1) = \Pr(s_t = m, \boldsymbol{y}_1^t); \\ \gamma_t(m', m) &= \Pr(s_t = m, y_t|s_{t-1} = m'); \\ \beta_t(m) &= \Pr(\boldsymbol{y}_{t+1}^n|s_t = m). \end{aligned} \quad (2.16)$$

Мұндағы $\alpha_t, \gamma_t, \beta_t$ – шартты ауысулар функциясы.

Алгоритмды бес циклдан тұрады деп есептесек болады. Шартты ықтималдықтар белгілі болғандықтан кодтың торында екі бағытта жүріп өту де жеткілікті болады. Соның ішінде γ және α тура бағытта, яғни солдан оңға қарай, ал σ және λ кері бағытпен қозғалған кезде есептеледі.

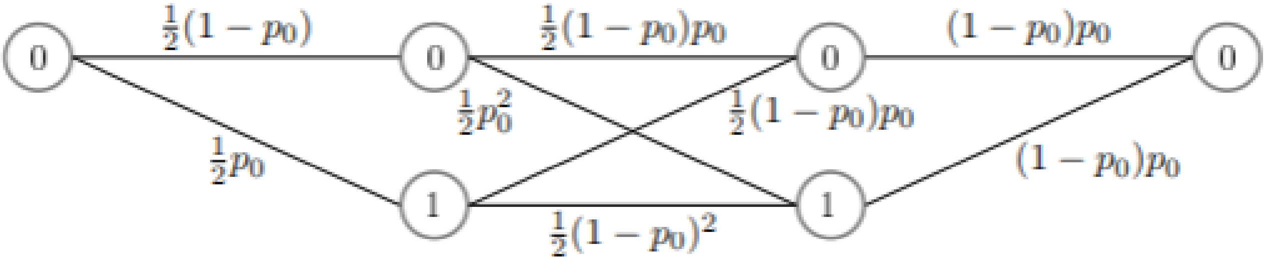
$$G = \begin{pmatrix} 1\ 1\ 1\ 0\ 0 \\ 0\ 1\ 0\ 1\ 1 \end{pmatrix} \quad (2.17)$$

Құраушы матрицасы арқылы берілген код байланыс арнасы арқылы келесі реттіліктегі кодты жіберді: $y = (0, 0, 1, 0, 1)$. Жіберілген символдарды қабылдау ықтималдығы бірдей деп есептеп, шартты қабылдану ықтималдығын есептейміз, кодтық сөз үшін және әр бір қабылданатын символ үшін. Алынған мәліметтер негізінде ықтималдылықты ескеріп 2.9 суретте секцияландырылған торды құруға болады.

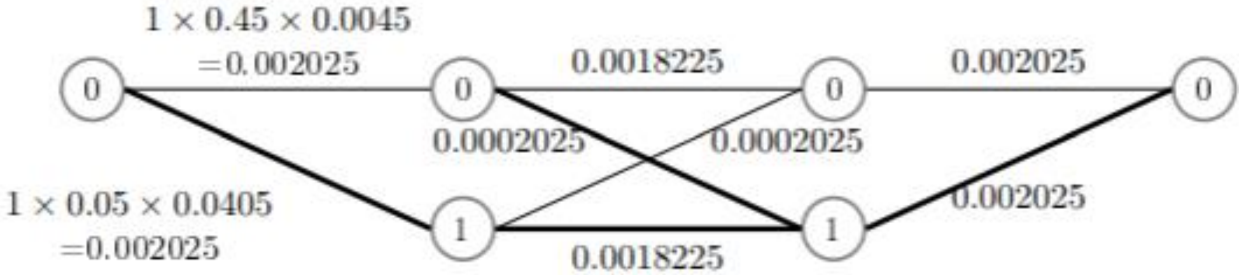


Сурет 2.9 – Кодтың торы және қабылданған кодтық сөз

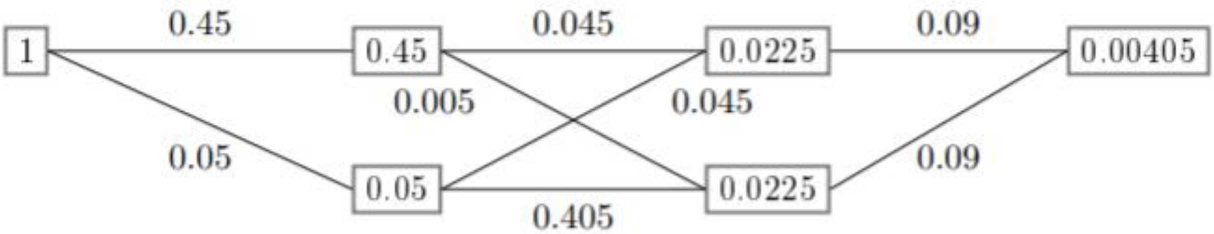
Шартты ықтималдылық белгілі болғандықтан кодтың торында тура және теріс бағыттар үшін есептеулерді жүргізуге болады. [20]



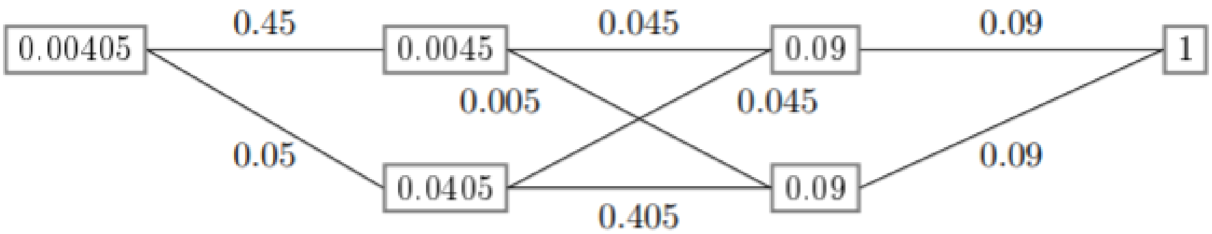
а) γ есептеу



ә) σ есептеу



б) α есептеу



в) β есептеу

Сурет 2.10 – БКДР алгоритмы бойынша кодтың торында декодтау

2.5 Сызықты блокты кодтар үшін тор көмегімен декодтау процесінің қиындығы

Жоғарыда қарастырылған торлардың кодтау теориясындағы қолданылуы мүмкіндікі шектелген. Кодтарды тор түрінде көрсету декодтау процесін тек графтағы жолдарды табумен ғана шектесе де, күрделі есептеулерден құтқармайды. Қазіргі таңдағы микроэлектроника және есептеуіш техниканың даму деңгейіне қарамастан, кодтау теориясы бұл жүйелерден 2^8-2^{12} шамасындағы арифметикалық операцияларды орындау деңгейін ең тиімді деп есептейді.

Бұл тұжырымнан кодтардың торын пайдаланып декодтау тек жылдамдығы k/n = 1/2 болатын, ал ұзындығы 50 – ден аспайтын кодтар үшін ғана тиімді болатындығын көрсетеді. Бұл кодтар өз кезегінде күрделілігі төмен кодтар қатарына жатады.

2.6 Ең кіші код торының қасиеттері

С кодтық сөз сызықты кодтың мүшесі болатын жиынға тиісті болады. Сызықты код жиыны кіші жиындардан тұратындықтан, барлық кодтық сөздер үшін ортақ болады.

Сызықты код үшін $Fi(0)$ жиыны, ұзындығы n – і болатын сызықты код.

Сызықты код үшін кез келген бос емес F_i(a) жиыны $a \neq 0$ болғанда, F_i(0) болған жағдаймен бірдей немесе мүлдем өзгеше болады. Мысалы, (a, 0) кодқа тиісті деп есептейік. $b \in Fi(a)$ болса, онда $(a, b) \in$ C деп есептеледі, код сызықты болғандықтан $(0, b) \in$ C, $b \in Fi(0)$. Демек жоғарыда айтылғандай $Fi(a) = Fi(0)$.

Өзге жағдайда, кодқа $(a, 0)$ тиісті емес болса, (a, b) кодтық сөз болса, ал $(0, b)$ кодқа тиісті емес, онда екі жиындар қиылыспайды немесе ұқсас емес деп есептеледі $Fi(a) \neq Fi(0)$. [21]

Келесі қасиет осы тұжырымнан шығады, сызықты код үшін $Fi(a) \neq Fi(0)$ болған жағдайда, екі жиынды бірдей шамаға кодтық комбинацияларға b векторының көмегімен жылжыту арқылы (a, b) кодтық сөз болатындай амал қолдансақ болады.

Кодтық арақашықтығы d болатын $Fi(0)$ жиыны үшін – n – і ұзындықты код арақашықтығы d арақашықтықтан кіші болмайды. Сызықты (n, k) коды, барлық i = 1, …, n-1 үшін келесі күрделілік профиліне ие болса $\xi = (\xi 0, \xi 1, ..., \xi n)$, онда бұл код үшін келесі теңдік орындалады:

$$k = k_i^p + \log_2 \xi_i + k_{n-i}^f. \tag{2.18}$$

2.7 Код торының қиындық шегі

Код ұзындығының код торының күрделілік профилімен байланысын анықтадық. Кодтық комбинациялардың ұзындығына байланысты тордағы деңгейлердің түйіндерінің саны анықталады. Кодтың мүмкін болатын кодтық арақашықтығымен манипуляциялар жасау арқылы, құрастыруға мүмкін болатын код торының күрделілігіне әсер етуге болады. Мұндай манипуляция әдісін асимптотикалық қиындығы деп атайды.

Сызықты блокты кодтар үшін тордағы деңгейлердің түйіндер саны екінің дәрежесіне тең болып есептеледі. Сондықтан, код торын құрау кезінде логарифмдік функциян пайдалану тиімді.

Код торының күрделілік профилі логарифмдік функция арқылы жазуға болады. Логарифмдік күрделілік тор үшін келесі шаманың максимал мәні болып есептеледі:

$$\zeta = \max\ \zeta. \tag{2.19}$$

Тұрақты ұзындығы бар кодтар үшін K(n, d), ұзындығы тұрақты n болатын, ара қашықтығы d минимал болатын жағдайда логарифмдік күрделілігі, осы шартты қанағаттандыратын кез келген сызықты блокты код үшін:

$$\zeta \geq \max i\ k - K(i, d) - K(n - i, d). \tag{2.20}$$

Мысалы, (23, 12) болатын Голей коды, код ұзындығы d=7. Кодтың әр түрлі параметрлерін пайдалану арқылы ең тиімді күрделілік деңгейін есептеп аламыз. Тордың параметрін есептеу құрылатын тордың ең кіші параметрге ие болуы, мүмкін болатын тордың бастапқы параметрін анықтайды. Ол параметрлер қатарына деңгейлерді қамтитын түйіндерің саны, әр деңгейдегі түйіндер саны, түйіндер мен басқа деңгейде орналасатын түйіндерді байланыстыратын жолдар саны.

Қарастырылып отырған код үшін i = 7 … 12 болған кездегі күрделілік профилін анықтайық:

$$\begin{aligned}
\zeta &\geq 12 - K(7,7) - K(16,7) = 12 - 1 - 5 = 6 \\
\zeta &\geq 12 - K(8,7) - K(15,7) = 12 - 1 - 5 = 6 \\
\zeta &\geq 12 - K(9,7) - K(14,7) = 12 - 1 - 4 = 7 \\
\zeta &\geq 12 - K(10,7) - K(13,7) = 12 - 1 - 3 = 8 \\
\zeta &\geq 12 - K(11,7) - K(12,7) = 12 - 2 - 2 = 8
\end{aligned} \tag{2.21}$$

Осы аралықты қамтитын күрделілік профилін есептеулерінің нәтижесінен, осы код үшін максималды мүмкін болатын кодтың күрделілілік деңгейінің шегі $\zeta \geq 8$. Бұл шарт осы код үшін кодтың торын құрастыру кезіндегі ең тиімді шаманың көрсеткіші болып есептеледі, күрделілік шегінен үлкен кодтың торын құру мағынасы жоқ болып есептеледі.

Кодтың торының күрделілігін анықтаудың маңыздылығы келесіде: қазіргі таңдағы есептеуіш машиналардың, құрылғылардың мүмкіндігінің шектеулі болуынан, код торының сызықты блокты кодтар үшін шектеулі параметрлерге ғана ие болатынын анықтайды. Үйірткілі кодтармен салыстырғанда сызықты блокты кодтар ақпаратты символдар мен тексеруші символдар арасындағы бөлек кодталу процесін қарастырады. Өз кезегінде мұндай айрмашылыққа байланысты кодтың торын тұрғызу әдісі де әр түрлі мәнге ие болады. Сызықты кодтар үшін құраушы матрица негізгі кодтың торын тұрғызуға мүмкіндік берсе, үйірткілі код үшін бұл кодтың торы тікелей кодтау процесінің ажырамас бөлігі болып есептеледі. Сонымен қатар кодер немесе декодердің белгілі уақыт мезетіндегі күйін көрсетеді, ал сызықты блокты код үшін, бұл тек құраушы матрицадан белгілі болған, рұқсат етілген кодтық комбинациялар үшін мүмкін болатын қадамдар болып есептеледі. Декодтау кезінде мұндай қадамдарды пайдалану өзіндік артықшылықты береді. Шынайылыққа жақын декодтау кезінде бұл кодтық комбинацияны қатесіз қабылдауға мүмкіндік береді, алайда мұндай декодер құрылымын жасау, ұзын болатын хабарламалар үшін мүмкін емес болып табылады.

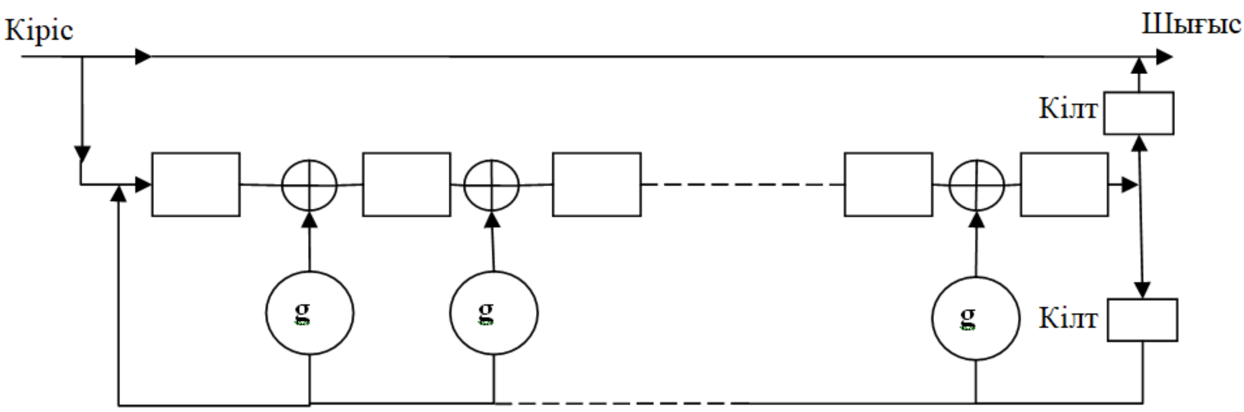
3 SystemView бағдарламасындағы Файер кодының кодтау және декодтау құрылғысының іске асырылуы

3.1 Файер кодының кодерін және декодерін бағдарламалық модельдеу

SystemView бағдарламасы электронды құрылғылардың жұмысын модельдеуге, бақылауға және сараптауға арналған арнаулы блоктар жиынтығынан тұратын бағдарлама. Бағдарлама мүмкіндіктерін пайдаланып сызықты блокты кодтың кодтаушы және декодтаушы құрылғыларының құрылысын қарастыруға болады.

Сызықтық блокты кодтардың қосалқы санаты болып табылатын циклдық кодтар, қарапайым түрлендірулерде қолданылатын жүйенің талаптарына сәйкес болуы мүмкін, олардың арасында кодты кеңейту және кодты қысқарту іске асырылу мүмкіндігі. Кейбір жағдайларда, түрлендіру циклдық кодының циклдік жылжуы, әрқашан рұқсат етілген код комбинациясына әкелмейтін кодқа әкелуі мүмкін. Яғни циклдық жылжу рұқсат етілмеген кодтық комбинацияға алып келеді. Бұл кодер немесе декодердің бастапқы да қандай мақсатта құрылғандығына байланысты болады. Кодер және декодер кодтық комбинациядағы қателерді тек анықтау үшін ғана немесе кодтық комбинацияда қатені анықтап, түзету мақсатында қолданылады. Буфферлік регистр кодтың қолданысына байланысты орналастырылады, ол қабылданған кодтық комбинацияның синдромын анықтау үшін қолданылады.

Циклдық Файер коды үшін кодердің жалпы құрылымы 3.1 суретте құраушы полином дәрежесінен алынады. Келесі құрылым циклдық кодтың кодері үшін жалпы ортақ болып есептеледі.

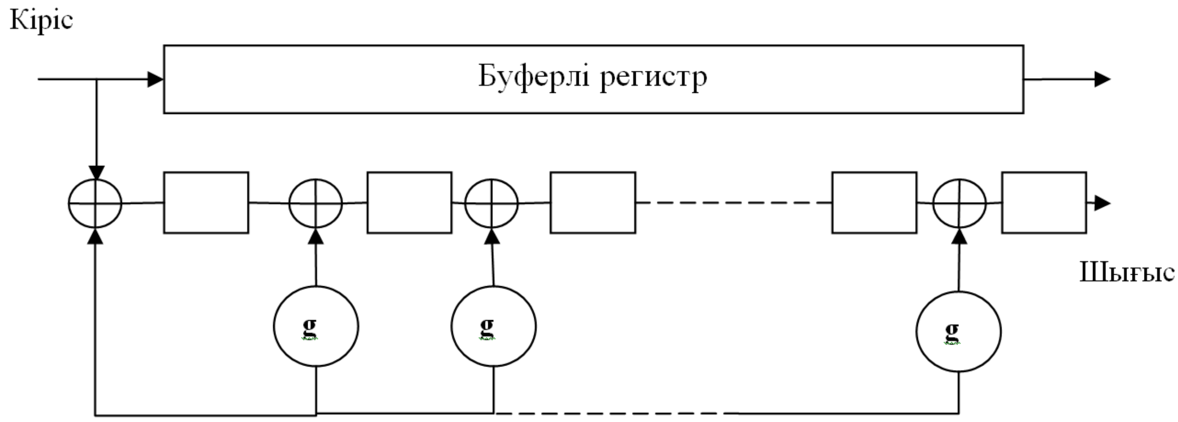


Сурет 3.1 – Файер кодерының құрылымдық сұлбасы

Бастапқыда бірінші кілт жабық күйде орналасады, екінші кілт ашық болады. Екінші кілт арқылы барлық ақпаратты символдар байланыс тізбегіне беріледі. Екінші кілт жабық күйге, бірінші кілт ашық күйге ауысады. Бұл сәтті бірінші регистрда бірінші тексеруші символ пайда болады және бірден байланыс арнасына беріледі. Сонымен бірге екінші тексеруші символ бірінші регистрда орналасады. Осылайшы барлық тексеруші символ жылжымалы регистр арқылы байланыс арнасына беріледі. Келесі қадамда бірінші кілт жабық күйге ауысады, ал екінші кілт бастапқы күйіне келеді. Келесі ақпараттық символдар байланыс арнасына беріледі. Бұл алгоритм және кодер құрылымы барлық циклдық кодтар үшін біртектілікке ие болады.

Қателердің жиынын анықтап, ол қателерді түзетуге арналған декодердің құрылымы бірлік қателерді анықтайтын циклдық кодтар декодерінен айырмашылық жасайды. Себебі мұндай кодтарда кодтық арақашықтық ұғымы қолданылмайды.

Декодер 3.2 сурет бойынша қабылданған кодтық комбинацияны сақтауға арналған буферлік регистрдан және осы қабылданған комбинацияны құраушы полиномға бөлуге арналған сұлбадан тұрады.



Сурет 3.2 – Файер декодерының құрылымдық сұлбасы

Егер қабылданған кодтық комбинацияны құраушы полиномға бөлу нәтижесі нөлге тең болса, комбинация қатесіз қабылданған болып есептеледі және бірден келесі құрылғыға немес тұтынушыға беріледі.

Қатені анықтау кодтық комбинацияның толықтай іске жарамсыз екендігін білдірді және қатені тек анықтайтын кодтар үшін кері байланыс арқылы хабарламаны қайта жіберуді сұратумен аяқталады. Қатені тауып, түзететін кодтар үшін синдромды есептеумен жалғасады. [22]

Бұл бөлімде Файер кодының кодерын және декодерын қарастыру арқылы сызықты блокты кодтар үшін жалпы ортақ болатын жағдайды сараптаймыз. Файер кодының келесі параметрлі кодеры және декодеры қарастырылады (120, 108). Бұл параметрлі код бірнеше қатені түзетеді.

Файер коды қателердің жиындарын түзетуге арналған және тек берілген ақпараттағы кодтық комбинациясы 1 ден басталып 1 ден аяқталатын қателердің жиындарын анықтайды, яғни пакеттерін. Мысалы, 1001, 1011, 1101, 1111.

Циклдық кодтардың өзге кодтау әдістерімен салыстырғанда Файер кодтарында кодтық арақашықтық шамасы қолданылмайды.

Файер коды келесі сипаттаушы шамалардан анықталады:

b_r –табылатын қателер пакетінің ұзындығын;

b_s – түзелетін қателер пакетінің ұзындығын.

Осы шамаларды қолдана отырып келесі алгоритммен кодтау жүзеге асады.

1) Файр кодының құраушы полиномын анықтаймыз, ол үшін келесі қадамдарды жасаймыз:

$$P(X)_{\text{ф}} = P(X)^l (X^c + 1) \tag{3.1}$$

Мұндағы $P(X)$- келтірілмейтін полином.

Құраушы полином сызықты код үшін, соның ішінде циклдық кодтың негізгі параметрін анықтайтын полином болып табылады. Файр коды үшін полином дәрежесі бірнеше қосымша параметр көмегімен анықталады.

2) Келесі теңсіздік арқылы келтірілмейтін полиномның дәрежесін l – анықтаймыз:

$$l \geq b_s \tag{3.2}$$

3) Кодтың құрылу принципінен келесі шамалард табамыз:

Сөз ұзындығын анықтау үшін c – өзара байланысқан аралықты бүтін санды және e келтірілмейтін полиномға тәуелді бүтін шаманы анықтаймыз

$$c \geq b_r + b_s - 1 \tag{3.3}$$

$$e = 2^l - 1 \tag{3.4}$$

4) Сөздің ұзындығы e және c сандарының Ең кіші ортақ еселігі арқылы табылады:

$$n = \text{ЕКОЕ}(e, c) \tag{3.5}$$

c және e сандары келесі шартты қанағаттандыруы қажет: таңдап алынған сандар қатынасы оң бүтін емес сан болуы қажет.

5) Тексеруші символдар санын табамыз:

$$m = c + l \tag{3.6}$$

6) Ақпараттық символдар санын анықтаймыз

$$k = n - m \tag{3.7}$$

7) Құраушы полиномды қажетті шамаларды орнына қойып есептейміз.

(3.1) формулаға сәйкес (120, 108) Файр кодының құраушы полиномы келесі түрде болады:

$$P(X)_{\Phi} = (X^4 + X + 1)(X^8+1) = X^{12} + X^9 + X^8 + X^4 + X + 1 \qquad (3.8)$$

(3.3) және (3.4) сәйкес құрылатын кодтың ұзындығы ең кіші ортақ еселік арқылы табылады, яғни келесі қадамдарды есептейміз:

$$c \geq 5 + 4 - 1 \qquad (3.9)$$

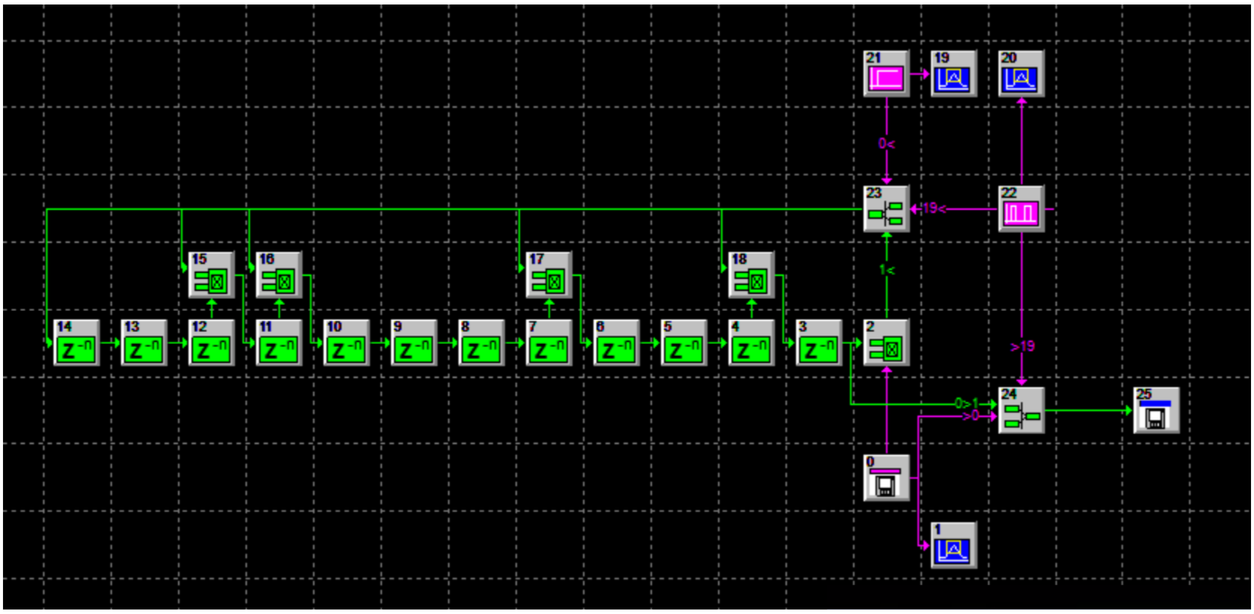
$$e = 2^4 - 1 = 15 \qquad (3.10)$$

$$n = \text{ЕКОЕ}(15, 8) = 120 \qquad (3.11)$$

Ақпаратты символдар белгілі болғандықтан, кодтың артықшылығын енгіземіз. Бұл келесі ұзындықтағы тексеруші символдарды (3.6) қажет етеді:

$$\mathrm{m} = c + l = 8 + 4 = 12 \qquad (3.12)$$

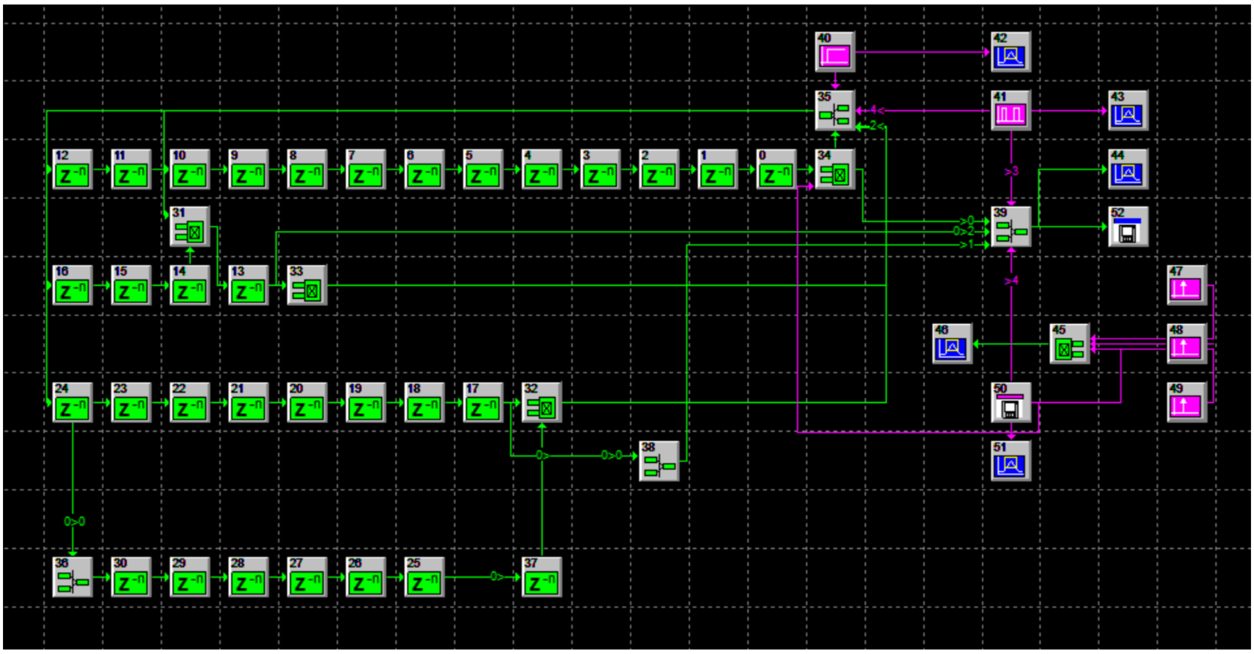
Файр кодын іске асыратын құрылғының негізін 3.3 суреттегідей кері байланысы бар жылжымалы регистр құрайды. Кері байланыс көпмүшелерді модуль 2 бойынша бөлуге мүмкіндік береді. Регистрлердегі әр бір тактыге сәйкес келетін қадам импульс көмегімен басқарылады. Бұл кодтау құрылғысы тексеруші символдарды құраушы полиномға тікеле бөлу арқылы есептеп анықтайды. Бұл үшін регистрдің бастапқы жағдайы (n-k) разрядта болады. Кодтаушы көпмүше коэффициенттері бірінші уақыт мезетінен кері байланысқа қатысады.



Сурет 3.3 – Файр кодының (120, 108) кодеры

Бастапқыда бірінші кілт қосылған күйде орналасады. Ақпараттық символдар бір мезетте регистрге және де байланыс арнасына беріледі. Сол себепті тексеруші символдар бір мезетте есептеледі. Осылайша k – тактта қалдық қалыптасады. Келесі уақыт мезетінде бірінші кілт ажыратылады, екінші кілт қосылады және тексеруші символдар байланыс арнасына беріледі.

Декодтаушы құрылғының құрылымы 3.4 суреттен байқауға болатындай кодтаушы құрылғының құрылымына ұқсас болады. Тек қана буферлік регистрлердің болуымен ғана айырмашылық жасайды. Буферлік регистрлер кодтық комбинациялардың қабылданған кезінде болатын қателердің синдромын сақтау үшін қолданылады.



Сурет 3.4 – Файр кодының (120, 108) декодеры

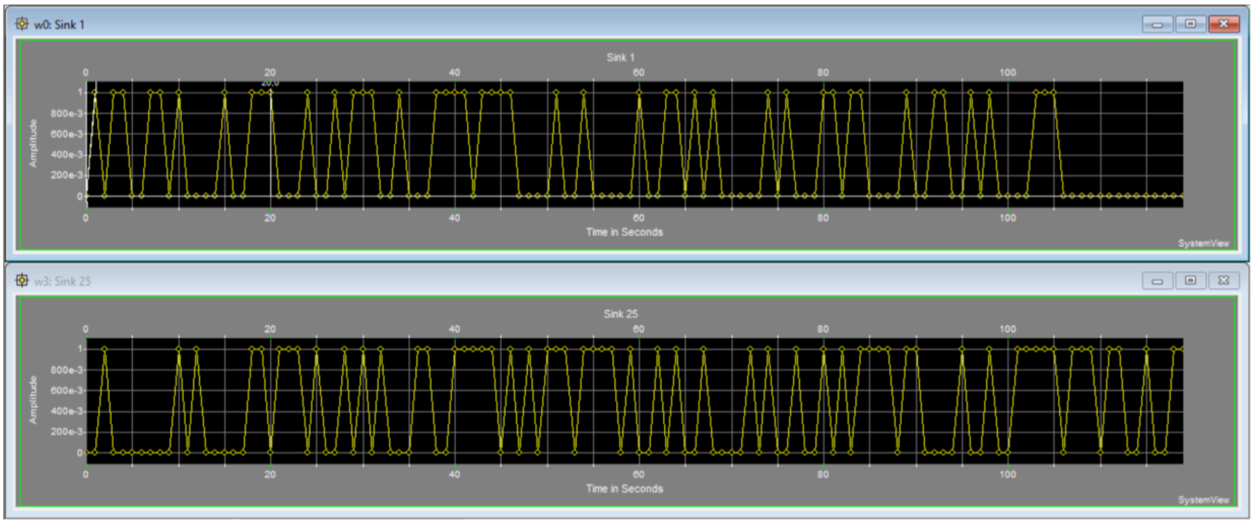
Егер кодтық комбинация қатесіз қабылданған болса, декодтау барысындағы символдар, яғни ақпараттық символдар бірден тұтынушыға беріледі.

Бірлік және жұп қателерді анықтау үшін, декодер сұлбасында екі синдромды қабылдауға мүмкіндік беретін буфферлік регистрды ұйымдастырамыз. Қабылдауға мүмкін болатын синдром ұзындығы артып кеткен жағдайда мүмкін болатын буфферлік регистр құрылымы өте күрделі түрге ие болады. Себебі, бұл декодердің құрылысын толықтай өзгертуді талап етеді. Қарастырылған модель бір уақыт интервалында ұзындығы 120-ға тең кодтық комбинацияны қабылдап декодтайды.

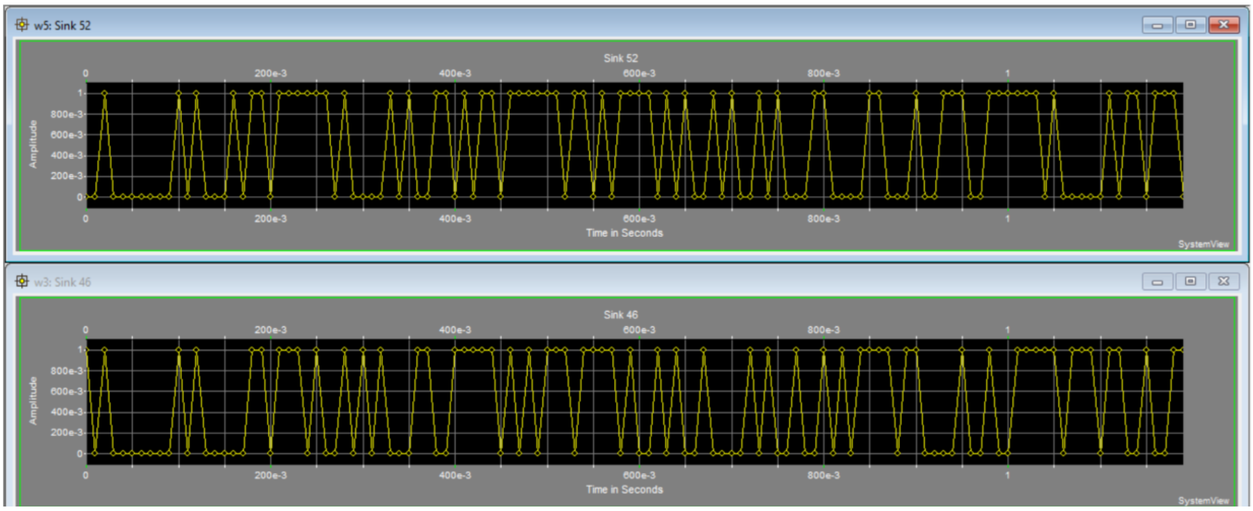
Осыған байланысты арнайы кодтар есептелген, бұл кодтардың декодерлері қабылданатын синдромға арнайы бапталады. Бұл екінші көпмүшелі полиномға бөлу арқылы іске асады.

3.2 Алынған нәтижелерді салыстырмалы түрде сараптау

Бағдарламадағы алынған модельдеу нәтижесі Файер кодының кодталған және декодталған комбинацияларының көрінісі болып табылады. Кодтық комбинацияға модель шегінде жасанды түрде қателіктер жіберілген. Қателіктер импульстер түріндегі бөгеуілдер арқылы іске асырылған. Қателіктерді түзету Файер кодының бастапқыда берілген параметрлеріне сәйкес беріледі. Декодер кірісінен алынатын анализдік терезеден қателердің жиынын 3.5 және 3.6 суреттен көруге болады. Қателер жиыны импульстік бөгеуілдердің нәтижесінде пайда болады.



Сурет 3.5 – Кодер кірісі және шығысы

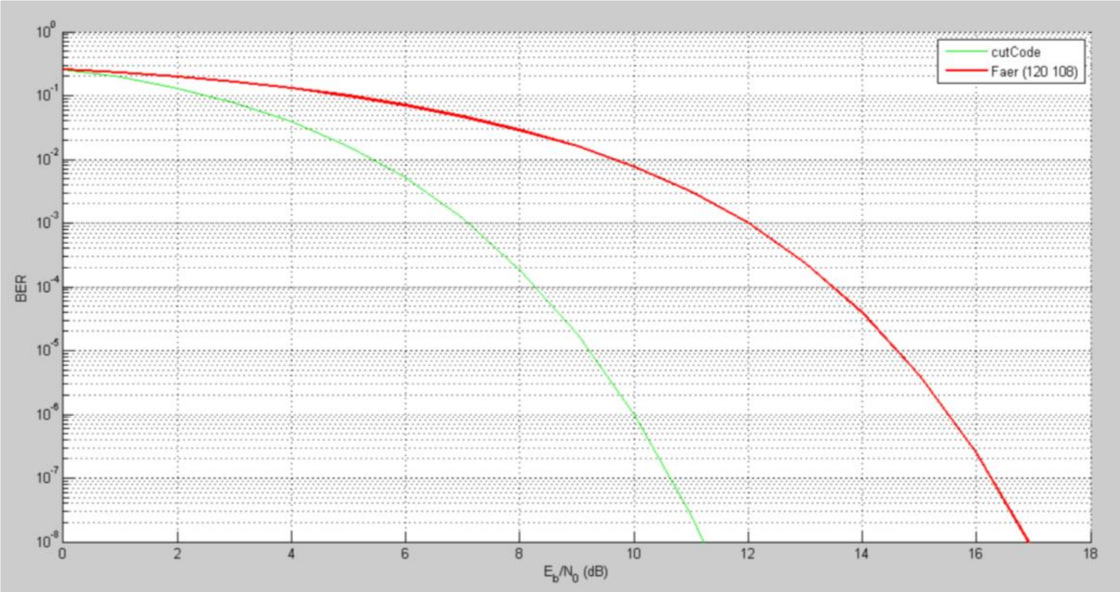


Сурет 3.6 – Декодер кірісі және шығысы

Кодер шығысындағы комбинация 3.5 суретте толықтай берілген параметрге сәйкес кодталған комбинация болып есептеледі. Кодтаушы құрылғы регистрлері барлық ақпаратты символдарды өз мәнінде қабылдайды, тек соңғы символдарды 0 ретінде есептейді.

Декодер толықтай 120 символды кодталған ақпаратты қабылдайды. Көрсетілген 3.6 суретке сәйкесінше кодтық комбинацияны қабылданған символдарға сәйкес декодтау процессы осы кодтық комбинацияларға сәйкес жүзеге асады. Декодтаушы құрылғы қабылданған қателерді түзетуді, бөлек жүзеге асырады. Бұл қабылданған кодтық комбинацияны құраушы полнномға бөліп, синдромдарды циклдық жылжыту арқылы жүзеге асады.

Қарастырылған сызықты блокты кодтармен модельдеу нәтижесіндегі мәліметтерді салыстыратын болсақ, әр бит ақпаратқа тиісті болатын қателердің санымен және таралатын ортадағы сигнал/шу қатынасының мәні арасындағы тәуелділікпен график тұрғызамыз. Есептеу MatLab бағдарламасының bertool командасының көмегімен, қажетті кодтық комбинацияны және негізгі сипаттаушы параметрлерді енгізу арқылы жүзеге асырылады. [23]



Сурет 3.7 – Салыстыру нәтижесі

Аддитивті гаустық шуылы бар арнаны бағдарлама баптауында қолдансақ, 3.7 суретте көрсетілген нәтижені аламыз. Бағдарламаның нәтижесінен қысқа ұзындықты комбинациялардың биттік қателер орын алуының төмен екендігін байқай аламыз, бұл декодтау процесіндегі код торының қысқа ұзындықты кодтық комбинация үшін қолданылуының оңтайлылығын білдіреді. Мұндай қысқа кодтар энергияны үнемдеу қағидатын ұстанатын құрылғылар мен технологиялар негізінде қолданылатын байланыс жүйелері үшін өте тиімді болып есептеледі, ал код торы бұл жүйелер үшін жұмыс барысын жеңілдететін құрал рөлін атқара алады.

ҚОРЫТЫНДЫ

Дипломдық жұмыста қарастырылған мәселелер қойылған мақсатқа сәйкес орындалды. Сызықты блокты кодтар үшін код торының анализы жүргізіліп, жалпы аналитикалық шолулардан бастап талданды. Сызықты блокты кодтарды сипаттайтын параметрлер анықталды және оларды есептеу жолдары көрсетілді.

Дипломдық жұмыстың көлемінде сызықты блокты кодтардың торын құраушы және синдромды матрицалар көмегімен тұрғызылды және әр әдістің өзіндік ерекшеліктері қарастырылды. Код торы жиі қолданылатын үйірткілі код торларының құрылу және қолданылу әдістерімен салыстырлып, үйірткілі кодтар үшін декодтау барысында қолданылатын Витерби алгоритмына ұқсас, шындыққа жақын қабылданатын ықтималдықтың шамасына байланысты Балл – Кук – Джелинека – Равив декодтау алгоритмы есептелді және алгоритм сызықты блокты кодтарға код торының метрикасы арқылы қолданылып, қабылданатын кодтық комбинациялардың пайда болу, қабылдану ықтималдылығы арқылы декодтау әдісінің тиімділігі жоғары екендігі дәлелденді.

Теориялық және іс жүзіндегі мәліметтерді салыстыру мақсатында SystemView бағдарламасында келесі құраушы (3.8) формуладағы полином арқылы берілген Файер (120, 108) кодының кодерінің және декодерінің жұмысы іске асырылды. Модельдеу нәтижесі кодер және декодердің шығысы мен кірісін сипаттайтын график түрінде берілді.

Дипломдық жұмыста қарастырылған код торы кодтау теориясында өте жиі қолданыс табатын зерттеу құралы, ал кейбір кодтар үшін негізгі кодтау және декодтау құралы. Код торы толықтай сызықты блокты код қамтитын кодтық комбинацияларды қамти алады, күрделі есептеу жұмыстарын жүргізбей-ақ код кодтау және декодтау процесінің оңтайлылығын көрсете алады, бірақ сызықты блокты кодтардың торы ұзын кодтық комбинациялар үшін тиімсіз болып табылатындығын ескеріп келесі қорытындыны нақтылауға болады: код торын қолдану тиімділігі және тиімсіз болатын тұстарын салыстыру арқылы код торы өте тиімді зерттеу моделі бола алатындығына және сараптау жүргізудің керемет әдісінің бірі болып табылатындығына көз жеткізуге болады.

Жұмысты орындау барысында алынған мәліметтер және қорытынды нәтижелер «цифрлы байланыс технологиялары» және «бөгеуілге тұрақты кодтау» пәндерінің тәжірибелік және зертханалық жұмыстарында қолданыла алады.

ПАЙДАЛАНЫЛҒАН ӘДЕБИЕТТЕР ТІЗІМІ

1 Питерсон У., Уэлдон Э. Коды, исправляющие ошибки.— М.: Мир, 1976.

2 Скляр Б. Цифровая связь. — Вильямс, 2003.

3 Томаси У. Электронные системы связи. — Техносфера, 2007.

4 Кутьин А.М. Проблемы теории кодирования и теория геометрий. — Красноярск, 2001.

5 Тутевич В. Н. Телемеханика. — Москва, 1985.

6 Moreira J. C., Farrell P. G. Essentials of Error-Control coding. — London: Wiley, 2016.

7 Блейхут Р. Теория и практика кодов, контролирующих ошибки. – Мир, 1986.

8 Варбанец С. П. Теория кодирования. — Одесса, 2013.

9 Аршинов М. Н., Садовский Л. Е. Коды и математика (Рассказы о кодировании) — М.: Наука, 1983.

10 Федоренко С. В. Методы быстрого декодирования линейных блоковых кодов.— СПб.: ГУАП, 2008.

11 Кудряшов Б. Д. Основы теории кодирования. — БХВ-Петербург, 2016.

12 Загидуллин Р., Карутин С. SystemView – системотехническое моделирование обработки сигналов. — Горячая Линия, 2015.

13 Гоноровский И. С. Частотная модуляция и её применение. — Связь, 2003

14 Иосиф З., Разевиг В. SystemView – средство системного проектирования радиоэлектронных устройств. — Спб.: Горячая Линия, 2002.

15 Золотарев В. В. Коды и кодирование. — Москва, 1990.

16 Кутьин А. М. Коды, композиции и решетки. — Красноярск, 2014.

17 Вернер М. Основы кодирования. — М.: Техносфера, 2005.

18 Морелос – Сарагоса Р. Искусство помехоустойчивого кодирования: Методы, алгоритмы, применения. Учебное пособие для студентов. — М.: Техносфера, 2005.

19 Журавлев В. Г., Куранова Н. Ю., Евсеева Ю. Ю. Помехоустойчивые коды. — Владимир, 2013.

20 Robert H. The art of error correcting coding. — London, 2006.

21 Shu Lin, Daniel J. Costello, Error control coding: Fundamental and Applications. — New Jersey, 2000.

22 Yuan Jiang. A practical guide to error-control coding using Matlab. — London, 2010.

23 Tood K. Moon. Error correction coding. Mathematical methods and algorithms. — London: Wiley, 2015.

А Қосымша

Ақпаратты символдар k = 108

0,00000000000000e+000	0,00000000000000e+000	0,00000000000000e+000
1,00000000000000e+000	0,00000000000000e+000	0,00000000000000e+000
0,00000000000000e+000	1,00000000000000e+000	1,00000000000000e+000
1,00000000000000e+000	1,00000000000000e+000	0,00000000000000e+000
1,00000000000000e+000	1,00000000000000e+000	1,00000000000000e+000
0,00000000000000e+000	1,00000000000000e+000	0,00000000000000e+000
0,00000000000000e+000	0,00000000000000e+000	0,00000000000000e+000
1,00000000000000e+000	1,00000000000000e+000	0,00000000000000e+000
1,00000000000000e+000	1,00000000000000e+000	1,00000000000000e+000
0,00000000000000e+000	1,00000000000000e+000	1,00000000000000e+000
1,00000000000000e+000	1,00000000000000e+000	0,00000000000000e+000
0,00000000000000e+000	0,00000000000000e+000	1,00000000000000e+000
0,00000000000000e+000	0,00000000000000e+000	1,00000000000000e+000
0,00000000000000e+000	0,00000000000000e+000	0,00000000000000e+000
0,00000000000000e+000	0,00000000000000e+000	0,00000000000000e+000
1,00000000000000e+000	1,00000000000000e+000	0,00000000000000e+000
0,00000000000000e+000	0,00000000000000e+000	0,00000000000000e+000
0,00000000000000e+000	0,00000000000000e+000	1,00000000000000e+000
1,00000000000000e+000	1,00000000000000e+000	0,00000000000000e+000
1,00000000000000e+000	0,00000000000000e+000	0,00000000000000e+000
1,00000000000000e+000	0,00000000000000e+000	1,00000000000000e+000
0,00000000000000e+000	0,00000000000000e+000	1,00000000000000e+000
0,00000000000000e+000	0,00000000000000e+000	0,00000000000000e+000
0,00000000000000e+000	0,00000000000000e+000	0,00000000000000e+000
1,00000000000000e+000	1,00000000000000e+000	1,00000000000000e+000
0,00000000000000e+000	0,00000000000000e+000	0,00000000000000e+000
0,00000000000000e+000	0,00000000000000e+000	1,00000000000000e+000
1,00000000000000e+000	1,00000000000000e+000	0,00000000000000e+000
0,00000000000000e+000	1,00000000000000e+000	0,00000000000000e+000
1,00000000000000e+000	0,00000000000000e+000	0,00000000000000e+000
1,00000000000000e+000	1,00000000000000e+000	0,00000000000000e+000
1,00000000000000e+000	0,00000000000000e+000	1,00000000000000e+000
0,00000000000000e+000	1,00000000000000e+000	1,00000000000000e+000
0,00000000000000e+000	0,00000000000000e+000	1,00000000000000e+000
1,00000000000000e+000	0,00000000000000e+000	0,00000000000000e+000
0,00000000000000e+000	0,00000000000000e+000	

Б Қосымша

Файр кодының (120, 108) нәтижесі

0,00000000000000e+000	0,00000000000000e+000	0,00000000000000e+000
0,00000000000000e+000	1,00000000000000e+000	0,00000000000000e+000
1,00000000000000e+000	1,00000000000000e+000	1,00000000000000e+000
0,00000000000000e+000	0,00000000000000e+000	0,00000000000000e+000
0,00000000000000e+000	0,00000000000000e+000	1,00000000000000e+000
0,00000000000000e+000	1,00000000000000e+000	0,00000000000000e+000
0,00000000000000e+000	1,00000000000000e+000	0,00000000000000e+000
0,00000000000000e+000	1,00000000000000e+000	1,00000000000000e+000
0,00000000000000e+000	1,00000000000000e+000	0,00000000000000e+000
0,00000000000000e+000	1,00000000000000e+000	0,00000000000000e+000
1,00000000000000e+000	0,00000000000000e+000	1,00000000000000e+000
0,00000000000000e+000	1,00000000000000e+000	0,00000000000000e+000
1,00000000000000e+000	0,00000000000000e+000	1,00000000000000e+000
0,00000000000000e+000	1,00000000000000e+000	0,00000000000000e+000
0,00000000000000e+000	0,00000000000000e+000	1,00000000000000e+000
0,00000000000000e+000	1,00000000000000e+000	1,00000000000000e+000
0,00000000000000e+000	1,00000000000000e+000	1,00000000000000e+000
0,00000000000000e+000	1,00000000000000e+000	1,00000000000000e+000
1,00000000000000e+000	0,00000000000000e+000	0,00000000000000e+000
1,00000000000000e+000	1,00000000000000e+000	1,00000000000000e+000
0,00000000000000e+000	1,00000000000000e+000	1,00000000000000e+000
1,00000000000000e+000	1,00000000000000e+000	0,00000000000000e+000
1,00000000000000e+000	1,00000000000000e+000	0,00000000000000e+000
1,00000000000000e+000	0,00000000000000e+000	0,00000000000000e+000
0,00000000000000e+000	1,00000000000000e+000	0,00000000000000e+000
1,00000000000000e+000	0,00000000000000e+000	1,00000000000000e+000
0,00000000000000e+000	0,00000000000000e+000	0,00000000000000e+000
0,00000000000000e+000	1,00000000000000e+000	0,00000000000000e+000
1,00000000000000e+000	0,00000000000000e+000	1,00000000000000e+000
0,00000000000000e+000	1,00000000000000e+000	0,00000000000000e+000
1,00000000000000e+000	0,00000000000000e+000	0,00000000000000e+000
0,00000000000000e+000	0,00000000000000e+000	1,00000000000000e+000
1,00000000000000e+000	1,00000000000000e+000	1,00000000000000e+000
0,00000000000000e+000	0,00000000000000e+000	1,00000000000000e+000
0,00000000000000e+000	0,00000000000000e+000	1,00000000000000e+000

Б қосымшасының жалғасы

1,00000000000000e+000	0,00000000000000e+000	1,00000000000000e+000
0,00000000000000e+000	1,00000000000000e+000	0,00000000000000e+000
1,00000000000000e+000	1,00000000000000e+000	0,00000000000000e+000
1,00000000000000e+000	0,00000000000000e+000	1,00000000000000e+000
1,00000000000000e+000	0,0000000000000e+000	1,00000000000000e+000

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ және ҒЫЛЫМ МИНИСТРЛІГІ
Қ.И. СӘТБАЕВ атындағы ҚАЗАҚ ҰЛТТЫҚ ТЕХНИКАЛЫҚ ЗЕРТТЕУ УНИВЕРСИТЕТІ

ҒЫЛЫМИ ЖЕТЕКШІНІҢ
СЫН-ПІКІРІ

дипломдық жұмысына

Ізбасар Айбол Ізбасарұлы

5В071900-Радиотехника, электроника және телекоммуникациялар

Тақырыбына: «Сызықты блокты кодтарға арналған тордың анализы»

Бұл дипломдық жұмыста сызықты блокты кодтарға қатысты код торының қолданылу тиімділігін сараптап анықтау мақсаты қойылған.

Қойылған мақсатка сәйкес дипломдық жұмыс көлемінде келесі мәселелер қарастырылған:

–бөгеуілге тұрақтылыкты арттыру мақсатындағы модуляция процесінің қолданылуына және сызықты блокты кодтарға аналитикалық шолу жасалған;

–код торын сипаттайтын параметрлер анықталған;

–код торын матрица көмегімен құру тәсілдері көрсетілген;

–Файер кодының бағдарламалық моделі сипатталған және алынған нәтижелер сарапталған.

Бөгеуілге тұрақты сызықты блокты кодтарға және ақпаратты тарату жүйелеріндегі модуляция процесіне аналитикалық шолу жасалған. Сызықты блокты кодтардың параметрлеріне және математикалық ерекшеліктеріне сипаттама берілген.

Кодтау теориясындағы графтардың колданысы қарастырылып, сызықты блокты кодтардың торларын тұрғызу тәсілі зерттелініп, код торының сипаттамасы жасалынды. Сызықты блокты кодтарды декодтаудың Балл – Кук – Джелинека – Равив алгоритмы қарастырылып, торға қатысты метрикасы есептелді.

SystemView бағдарламасының мүмкіншілік деңгейіне сәйкес Файер (120, 108) кодының кодері және декодерінің моделі іске асырылған. Алынған нәтижелерді салыстырмалы сараптау арқылы код торының мүмкіншілігіне баға берілген.

Бұл дипломдық жұмыс жоғарғы оқу орындарының талаптарына сай жеткілікті жоғары дәрежеде жазылған, нәтижелер ақпаратты бөгеуілден қорғау және тарату жүйелерінің ғылыми бағытына жауап береді.

Жалпы, дипломдық жұмысты "95" (А+) өте жақсы деген бағаға, ал студент Ізбасар А. І. 5В071900 – Радиотехника, электроника және телекоммуникация мамандығы бойынша техника және технологиялар «бакалавр» академиялық дәрежесіне ұсынылады.

Ғылыми жетекші
ЭТжҒТ кафедрасының
қауымдастырылған профессоры
__________Л.Б. Илипбаева
«24» 04 ______ 2019 ж.

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ және ҒЫЛЫМ МИНИСТРЛІГІ
Қ.И. СӘТБАЕВ атындағы ҚАЗАҚ ҰЛТТЫҚ ТЕХНИКАЛЫҚ ЗЕРТТЕУ УНИВЕРСИТЕТІ

СЫН – ПІКІР

дипломдық жұмыс

Ізбасар Айбол Ізбасарұлы

5В071900 - Радиотехника, электроника және телекоммуникациялар

Тақырыбына: «Сызықты блокты кодтарға арналған тордың анализы»

Орындалды:
а) графикалық бөлімі 26 бет;
ә) түсіндірме жазбасы 52 бет.

ЖҰМЫСҚА ЕСКЕРТУ ЖАСАУ

Бұл дипломдық жұмыста Ізбасар Айбол Ізбасарұлы бөгеуілге тұрақты сызықты блокты кодтар үшін код торының қолданылуын сараптаған. Дипломдық жұмыс келесі мәселелерді қарастырады:

Біріншіден, бөгеуілге тұрақтылықты арттыру мақсатындағы бөгеуілге тұрақты сызықты блокты кодтарға және модуляция процесіне аналитикалық шолу жасау арқылы негізгі параметрлері анықталып, сызықты блокты кодтардың математикалық ерекшеліктері сипатталған.

Екіншіден, кодтау теориясындағы код торына және код торын сипаттайтын негізгі параметрлерге анықтама берілген. Код торын құраушы және синдромды матрицалар көмегімен тұрғызу тәсілдері есептеліп көрсетілген. Ең кіші код торы, спэндік матрица, код торының күрделілік коэфициентін есептеу және шынайылыққа жақын болатын Балл-Кук-Джелинека-Равив декодтау алгоритмы қарастырылған. Жиналған мәліметтерді сараптау арқылы код торының қолданылу артықшылығы және кемшілігіне баға берілген.

Үшіншіден, Файер (120, 108) коды есептеліп, SystemView бағдарламасында Файер кодының кодерін және декодерін модельдеу арқылы бағдарламалық деңгейде кодтау және декодтау процесі іске асырылған. Код мүмкіншілігін бағалау үшін модельдеуден алынған нәтижелер салыстырмалы түрде сараптау арқылы, код торының сызықты блокты кодтарға қатысты қолдану тиімділігі және кемшілігі жайында жазылған.

Дипломдық жұмыста стилистикалық қателіктер орын алған, бірақ бұл дипломдық жұмыстың өзектілігін төмендетпейді.

Дипломдық жұмыс жоғарғы оқу орындарының талаптарына сай жеткілікті жоғары дәрежеде жазылған, алынған нәтижелер мамандығы бойынша ғылыми бағытқа жауап береді.

Жұмыс бағасы

Жалпы, дипломдық жұмыс "95/А/ өте жақсы" деген бағаға, ал студент Ізбасар А. І. 5В071900 – Радиотехника, электроника және телекоммуникация мамандығы бойынша техника және технологиялар «бакалавр» академиялық дәрежесіне ұсынылады.

Пікір беруші
ҚазҰАУ, ЭҮжА каф. меңгерушісі,
доктор PhD.,
қауымдастырылған профессор
_______ Ж. С. Шыныбай
«25» 04 2019 ж