

APCO TELSİZ SİSTEMLERİNDE ORANSIZ KODLARIN
KULLANIMI

GÜLAY ÇIĞIRKAN

YÜKSEK LİSANS TEZİ
ELEKTRİK VE ELEKTRONİK MÜHENDİSLİĞİ

TOBB EKONOMİ VE TEKNOLOJİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

ARALIK 2014

ANKARA

Fen Bilimleri Enstitü onayı

Prof. Dr. Osman EROĞUL
Müdür

Bu tezin Yüksek Lisans derecesinin tüm gereksinimlerini sağladığını onaylarım.

Prof. Dr. Murat ALANYALI
Anabilim Dalı Başkanı

GÜLAY ÇİĞİRKAN tarafından hazırlanan APCO TELSİZ SİSTEMLERİNDE
ORANSIZ KODLARIN KULLANIMI adlı bu tezin Yüksek Lisans tezi olarak
uygun olduğunu onaylarım.

Doç. Dr. Tolga GİRİCİ
Tez Danışmanı

Tez Jüri Üyeleri

Başkan : Yrd. Doç. Dr. İsrail BAHÇECİ

Üye : Doç. Dr. Tolga GİRİCİ

Üye : Doç. Dr. Kemal BIÇAKCI

TEZ BİLDİRİMİ

Tez içindeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edilerek sunulduğunu, ayrıca tez yazım kurallarına uygun olarak hazırlanan bu çalışmada orijinal olmayan her türlü kaynağa eksiksiz atıf yapıldığını bildiririm.

Gülay ÇİĞİRKAN

Üniversitesi : TOBB Ekonomi ve Teknoloji Üniversitesi
Enstitüsü : Fen Bilimleri
Anabilim Dalı : Elektrik ve Elektronik Mühendisliği Anabilim Dalı
Tez Danışmanı : Doç. Dr. Tolga GİRİCİ
Tez Türü ve Tarihi : Yüksek Lisans – Aralık 2014

Gülay ÇIĞIRKAN

APCO TELSİZ SİSTEMLERİNDE ORANSIZ KODLARIN KULLANIMI

ÖZET

Kamu Güvenliği İletişimi Kurumu'nun (Association of Public-Safety Communications Officials-APCO) Proje25 standardı kamu güvenliği ve acil durum haberleşme sistemlerinde telsiz haberleşmesini tanımlamaktadır. Bu haberleşme standardı sayısal ses ve veri iletişimini desteklemektedir. Telsiz uygulamaları arasında teke gönderimin yanısıra çoğa gönderim gerektiren veri uygulamaları da vardır. Her bir kullanıcıdan her paket için onay mesajı almak mümkün olmadığından APCO P25 sisteminde çoğa gönderim genellikle onaysız olarak yapılır. Gönderilen bilginin önemine göre veri ya arttırılmış bir kodlama ile alındı onayı olmaksızın gönderilmekte ya da her kullanıcı sıraya alınarak teker teker gönderim yapılmaktadır. Onaylı veri iletiminde ortaya çıkan trafikte bütün kullanıcılardan onay almak zorunda olmak, noktadan noktaya sırayla aynı veriyi sistemden defalarca göndermek sistem yükünü aşırı derecede artırmaktadır.

Oransız kodlama özellikle çoğa gönderim uygulamalarında sistem yükünü azaltan bir tekniktir. En büyük avantajı her alınan paket için geri bildirim mesajı gönderilmesine gerek olmamasıdır. Alıcının sadece orijinal veriyi çözmeyi başardığı anda olumlu geri bildirim yapması yeterlidir. Diğer bir avantaj ise yeniden iletme gerek duyulmamasıdır. Bu avantajlardan dolayı bu çalışmada APCO P25 sisteminde çoğa gönderim uygulamalarında oransız kodların kullanımı incelenmiştir.

Anahtar Kelimeler: APCO, PMR, Oransız Kodlar, Çoğa Gönderim, Yayın.

University : TOBB University of Economics and Technology
Institute : Institute of Natural and Applied Sciences
Science Programme : Electrical and Electronics Engineering
Supervisor : Assoc. Prof. Tolga GİRİCİ
Degree Awarded and Date : M.Sc. – December 2014

Gülay ÇİĞİRKAN

USE OF FOUNTAIN CODES IN APCO RADIO SYSTEMS

ABSTRACT

Association of Public-Safety Communications Officials (APCO) Project25 is a standard that defines public safety and emergency wireless communications. It supports digital voice and data transmission. Among the wireless applications there are ones that require broadcast transmission besides unicast transmission. Broadcast transmission in APCO P25 is normally unconfirmed, as it is impossible to receive acknowledgement (ACK) from every user and for every packet. Depending on the importance of the information, either all packets are sent without any ACK and with stronger error correcting codes, or a separate confirmed session is established with each and every user. For confirmed transmission, requiring to get ACK from every user and sending whole information separately to each user make the system inefficient.

Rateless coding is a technique that is especially useful in broadcast applications. The greatest advantage is that receivers don't need to send a separate ACK for each received packet. They only need to send ACK, as they are able to recover the original data. The second advantage is that no retransmission is needed. Due to the above listed advantages, this thesis will address the use of rateless codes in APCO P25 broadcast applications.

Keywords: APCO, PMR, Rateless Codes, Multicast, Broadcast.

TEŐEKKÖR

Bu alıŐma Bilim, Sanayi ve Teknoloji Bakanlıđı'nın desteklediđi Aselsan A.Ő. ile birlikte yÖrÖtÖlen San-Tez 1225-STZ.2012-1 numaralı "APCO Telsiz Sistemlerinde Oransız Kodların Kullanımı" isimli proje kapsamındadır.

Bu alıŐmada emeđi geen; baŐta tez danıŐmanım Do. Dr. Tolga GİRİCİ'ye, bu sÖrete bana desteklerini esirgemeyen aileme ve kıymetli tecrÖbelerinden faydalandıđım TOBB Ekonomi ve Teknoloji Öniversitesi Elektrik ve Elektronik MÖhendisliđi BÖlÖmÖ Öđretim Öyelerine ve arkadaŐlarıma sonsuz teŐekkÖrler...

İÇİNDEKİLER

ÖZET	iv
ABSTRACT	v
TEŞEKKÜR	vi
İÇİNDEKİLER	vii
ŞEKİLLERİN LİSTESİ	x
ÇİZELGELERİN LİSTESİ	xi
SEMBOL LİSTESİ	xii
1 GİRİŞ	1
1.1 Problem Tanımı	1
1.2 Tezin İçeriği ve Katkıları	2
2 FOUNTAIN KODLAR	3
2.1 Luby Dönüşüm (LT) Kodları	4
2.1.1 Kodlama	4
2.1.2 Kod Çözümü	4
2.2 Raptor Kodları	5

2.2.1	Kodlama	5
2.2.2	Kod Çözümü	5
2.3	Rastgele Doğrusal Fountain Kodlar	6
2.3.1	Kodlama	6
2.3.2	Kod Çözümü	6
3	APCO PROJE25 SİSTEMİ	8
3.1	APCO P25 Standardında Paket Yapısı	8
3.2	Rastgele Doğrusal Fountain Kodlar'ın Kullanımı	10
3.3	Sistem Modeli	11
3.4	Benzetim Sonuçları	16
3.5	Sonuçlar	22
4	APCO PROJE25 SİSTEMLERİNDE GÜVENLİ VE VERİMLİ ÇOĞA GÖNDERİM	24
4.1	Kullanıcı Sayısı Kestirimi	25
4.1.1	Tek Aşamalı Kestirim	25
4.1.2	Yaklaşık Maksimum Olabilirlik (YMO) Kestirimi	27
4.1.3	Çok Aşamalı (ÇA) Kestirim	29
4.2	APCO Proje25 Sistemleri İçin Önerilen Çok Aşamalı Yöntem	30
4.3	Nümerik Sonuçlar	33
4.4	Sonuç	34

5 SONUÇLAR	36
KAYNAKLAR	38
ÖZGEÇMİŞ	41

ŞEKİLLERİN LİSTESİ

2.1	Rastgele Doğrusal Fountain Kodlar'ın Üretilmesi [6]	7
3.1	APCO P25 Standardında Paket Yapısı	9
3.2	Rastgele Doğrusal Fountain Kodlamada İletimde Gereken Ek Veri Yükünün Dağılım Grafiği	12
3.3	Vericide Yapılan İşlemler	13
3.4	C4FM Kipleme [19]	15
3.5	Yükseltilmiş Kosinüs Filtresinin Frekans Tepkisi	16
3.6	Darbe Şekillendirici Filtrenin Frekans Tepkisi	17
3.7	C4FM Kip Çözücü [19]	18
3.8	Farklı Hata Oranlarında Farklı Blok/Paket Oranları İçin Bit Yükü Grafiği	18
3.9	Paketteki Blok Sayısına (Nb) göre SGO Bitiş Zamani Grafiği	19
3.10	TBGG ve Rayleigh Kanallarında Sinyal Gürültü Oranı - Bit Hata Oranı Grafiği	20
3.11	TBGG ve Rayleigh Kanallarında Sinyal Gürültü Oranı - Blok Hata Oranı Grafiği	21
3.12	Değişik Blok Hata Oranları için 10KB Dosya İletim Süresi	22
4.1	Kullanıcı Sayısı - Ortalama Tamamlama Süresi	34
4.2	Kullanıcı Sayısı - Ortalama Tamamlama Süresi	35

ÇİZELGELERİN LİSTESİ

3.1	1/2 Trellis Kodlayıcı Durum Geçiş Tablosu	12
3.2	3/4 Trellis Kodlayıcı Durum Geçiş Tablosu	13
3.3	Serpiştirme Tablosu	14
3.4	Sembollerin Kipleme Fazlarına Eşlenişi	15
4.1	Maksimum Olabilirlik Kestirimi: Kestirim Hatasının Ortalama Kare Kökü	27
4.2	Yaklaşık Maksimum Olabilirlik Kestirimi: Kestirim Hatasının Ortalama Kare Kökü	28
4.3	Yaklaşık Maksimum Olabilirlik Kestirimi: Kestirim Hatasının Ortalama Kare Kökü	28
4.4	Yaklaşık Maksimum Olabilirlik Kestirimi: Çok Aşamalı Kestirim Performansının Ortalama Hata ve Ortalama Kare Kökü	29
4.5	Uyarlanır Çok Aşamalı Yaklaşık Maksimum Olabilirlik Kestirimi: Kestirim Süresi, Ortalama ve Ortalama Kare Kökü	31

Sembol Listesi

Değişken	Açıklaması	İlgili Kısım
s_i	$i \in \{1, 2, \dots, K\}$, Veri parçası	2
t_n	n . andaki oransız kodlanmış paket	2
G_{kn}	Üreteç matrisi	2
N_b	Bir paketteki blok sayısı	3
i	Erişim yapılmamış (boş) zaman aralık sayısı	4
s	Başarılı erişim yapılmış zaman aralığı sayısı	4
T	Toplam zaman aralığı sayısı	4
U	Toplam kullanıcı sayısı	4
$p_{T,U}(i, s)$	U kullanıcı aralıklara rastgele eriştiğinde, T aralıkta i boş ve s başarılı aralık olasılığı	4
U^{MO}	Maksimum olabilirlik kestirimine göre belirlenen kullanıcı sayısı	4
$\tilde{p}_{T,U}(i, s)$	U kullanıcı aralıklara rastgele eriştiğinde, T aralıkta i boş ve s başarılı aralık olduğunda yaklaşık olasılığı	4
U^{YMO}	Yaklaşık maksimum olabilirlik kestirimine göre belirlenen kullanıcı sayısı	4
U_k^{MO-CA}	Çok aşamalı maksimum olabilirlik kestirimine göre k . aşamada belirlenen kullanıcı sayısı	4
U_k^{YMO-CA}	Çok aşamalı yaklaşık maksimum olabilirlik kestirimine göre k . aşamada belirlenen kullanıcı sayısı	4
$p_{T_j,u}(i_j, s_j)$	Çok aşamalı maksimum olabilirlik kestirimine göre j . aşamada T_j aralıkta i_j boş ve s_j başarılı aralık olasılığı	4
$\tilde{p}_{T_j,u}(i_j, s_j)$	Çok aşamalı yaklaşık maksimum olabilirlik kestiriminde j . aşamada T_j aralıkta i_j boş ve s_j başarılı aralık olasılığı	4
L	Mesaj bit sayısı	4
D	N_b blok/paket için L veri biti gönderim süresi	4
$L_i^{r,0}$	Kullanıcı i 'nin bu ilk gönderimden sonra biriktirmeye ihtiyaç duyduğu geri kalan bit sayısı	4
$\hat{L}_{mean}^{r,k}$	k . aşamada ortalama geri kalan bit sayısının tahminini	4

1. GİRİŞ

1.1 Problem Tanımı

Proje25 (P25) Uluslararası Kamu Güvenliği İletişimi Kurumu (APCO International) ve diğer A.B.D. yerleşik kurumların çabalarıyla oluşturulan bir standartlar bütünüdür. Bu standartla karasal gezgin radyoların (Land Mobile Radio) haberleşmesi tanımlanmıştır ve bu standardı dikkate alan üretici firmalar Proje25 uyumlu ürünler üretebilir [1], [2]. APCO Proje25 standardı kamu güvenliği ve acil durum haberleşme sistemlerinde telsiz haberleşmesini tanımlamak için kullanılmaktadır. Kamu güvenliği için kullanılan telsiz sistemlerinde bilgiye hızlı ve zamanında ulaşılması, emniyet ve asayiş hizmetlerinin süratli bir şekilde yapılmasına imkan sağlar. Veri haberleşmesi bu telsiz sistemlerinde günden güne artmakta, hız ve süre kazandıran yöntemlerle iletişimi sağlamak, kaynakları verimli kullanabilmek önem kazanmaktadır.

APCO Proje25 standardı sayısal ses ve veri gönderimini desteklemektedir. Veri uygulamaları arasında acil durum mesajları, arama çağrıları gibi kısa mesajların yanında daha uzun (50-200KB hacminde) mesajlar da vardır. Veri iletimi merkezden tekil kullanıcılara olduğu gibi (unicast), çoklu kullanıcı gruplarına da olabilir (multicast/broadcast). Standart kapsamında veri gönderimi onaylı veya onaysız olarak gerçekleştirilmektedir. Onaysız gönderimde veri paketleri artırılmış bir kodlama ile kodlanarak ya da her kullanıcı sıraya alınarak teker teker gönderim yapılmaktadır. Ancak kamu güvenliği sistemi geniş bir alanda binlerce kullanıcıdan oluştuğunda çoğa gönderim noktadan noktaya yapılması nedeniyle aynı verinin sistemden defalarca geçmesine yol açar. Ayrıca onaylı veri iletimi durumunda alındı bilgilerinin gönderiminde çakışmalardan dolayı toplanamaması sistem yükünü aşırı derecede artırmaktadır.

1.2 Tezin İeriđi ve Katkıları

Bu tez alıřması, T.C. Bilim, Sanayi ve Teknoloji Bakanlıđı'nın "Sanayi Tezleri Programı" kapsamında ASELSAN A.ř. tarafından desteklenen ve yrtclđ tez danıřmanı Do. Dr. Tolga GİRİCİ tarafından yapılan "APCO Telsiz Sistemlerinde Oransız Kodların Kullanımı" bařlıklı proje ıřıđında yapılmıřtır. Tez ieriđi řu řekildedir: 1) Fountain Kodlar, 2) APCO Proje25 Sistemi, 3)APCO Proje25 Sistemlerinde Gvenli ve Verimli ođa Gnderim olmak zere  kısımdan oluřmaktadır.

Birinci kısımda; Fountain Kodları'ndan Luby Dnřm, Raptor ve Rastgele Dođrusal Fountain Kodları'nın alıřma prensipleri ile ilgili kısaca bilgiler verilmiřtir.

İkinci kısımda; APCO P25 sisteminin MATLAB zerinde gereklenmesi ve nerilen sistemin uygulanması incelenmiřtir. Bu blmde standartta yer alan paket yapısı, kipleme ile ilgili bilgiler verilmiřtir. Ayrıca Rastgele Dođrusal Fountain Kodlar'mın kullanılması durumunda mesajın zlebilmesi iin gereken paket miktarı incelenmiřtir. nerilen yntemden bahsedilip nerilen yntem iin Rayleigh snml kanalda bit hata oranı, farklı hata oranlarında bit yk ve paket iletim sreleri incelenmiřtir.

nc kısımda ise gvenli ve verimli bir ođa gnderim yapılabilmesi iin gnderim yapılacak kullanıcı sayısının kestirimi ile ilgili ok ařamalı mekanizma nerilmiřtir. nerilen metot zaman dilimli rastgele eriřime dayalı olan kestirim ve geri bildirim evrelerinden oluřmaktadır. Baz istasyonu ođa gnderim yapacak kullanıcı sayısını nerilen kestirim mekanizması ile belirleyip belirlenen deđere gre gnderim iřlemine devam eder. Kestirim mekanizmasının dahil edilmesiyle; 1) baz istasyonunun kullanıcı sayısının nceden bildiđi, 2) son kullanıcıların mesajı zdđnde ođa gnderimin hemen sonlandırıldıđı, 3) kullanıcıların akıřma olmadan geri bildirim yaptıđı varsayımları ile belirlenen alt sınıra kıyasla ne kadar srede ođa gnderimin tamamlandıđı incelenmiřtir.

2. FOUNTAIN KODLAR

Fountain kodlar aynı zamanda oransız kodlar olarak da bilinirler. Kod oranı yani kodlanan veya gönderilen sembol sayısı iletim süresince belirsizdir. Bu nedenle bu kodlar oransız kodlar olarak isimlendirilirler.

Fountain kodlar silinti kod ailesindedir. Oransız kodlar, silinti kanallar üzerinden iletim yapmak amaçlı geliştirilmiştir, uygulama katmanı doğrusal kodlardır [3]. Bu kodlar, ilk 1990'lı yılların sonunda [4] çalışmasında yer aldı ve ardından ilk pratik uygulamaları olan Luby Dönüşüm Kodları [5] ve Raptor Kodları [7] gelişimi takip etti. Çok kısa bir zaman içerisinde Raptor Kodları [8] ve [9] standartlarında yer almıştır.

Oransız kodlar kullanan kablosuz ağlarla ilgili literatürde pek çok çalışma olsa da, bu çalışmaların büyük kısmı fazlasıyla kuramsaldır ve idealize edilmiş sistemleri ele almışlardır. Bu tür çalışmalar verinin parçalara bölünüp belli bir prosedüre göre aralarından rastgele paketler seçilip XOR işlemine tabi tutulması ve alıcıdaki kod çözme prosedürünü dikkate almamaktadırlar. Oransız kodlar iki ayrı haberleşme standardına alınmıştır, Dijital Video Yayını (DVB) [10], [11], [12], [13] ve Üçüncü Nesil Ortaklık Projesi (3GPP) [14], [15], [16]. Bu çalışmalar, fiziksel katmandaki hata düzelten kodlamayı ve uygulama katmanındaki oransız kodlamayı beraber ele alarak sistemi optimize etmeye çalışmışlardır. Bunların yanında [17] makalesi de 802.11 tabanlı kablosuz ağlar için benzer bir çalışma yapmıştır.

Oransız kodlama özellikle çoğa gönderim uygulamalarında sistem veri yükünü azaltan bir kodlama türüdür. İletici elindeki K adet paket içerisinde her seferinde rastgele seçerek bit bazında XOR işleminden geçirir ve bir oransız kodlanmış paket oluşturulur. Bu işlem her seferinde tekrarlanarak, oluşturulan oransız kodlanmış paketler alıcıya iletilir. Burada alıcı(lar)ın spesifik bir kodlanmış paketi alıp almamaları önemli değildir, çünkü kodlanmış paketler

birbirlerinin yerini tutabilir. Alıcı N adet, K 'dan biraz daha fazla kodlanmış paket aldığımda orijinal veri paketlerini çözebilmektedir ($N = K + \epsilon$). Oransız kodlama tekniğinin en büyük avantajı her kodlanmış paket için geri bildirim mesajı iletilmesine gerek olmamasıdır. Alıcının sadece orijinal veri paketlerini çözmeyi başardığı anda olumlu geri bildirim yapması yeterlidir. Diğer bir avantaj ise yeniden iletme gerek duyulmamasıdır. Klasik çoğa gönderim sistemlerinde her alıcının farklı paketleri alamadığı durumda hemen hemen bütün paketlerin yeniden iletilmesi gerekecektir. Oysa ki oransız kodlamalı iletimde rastgele XORlanmış paketler birbirinin eşdeğeridir ve belli sayıda alındıklarında orijinal veri paketleri çözülebilir.

2.1 Luby Dönüşüm (LT) Kodları

Pratik olarak gerçekleştirilebilen ilk oransız kod LT Kodları'dır [5]. Bu kod yüksek sayıda paket/parça iletildiği durumda minimum yük yaratacak şekilde optimize edilmiştir. LT Kodları'nın kod çözme karmaşıklığı $O(K \ln(K))$ 'dır ve K arttıkça zaman karmaşıklık problemi yaratacak düzeye gelir. Bu sorun Raptor Kodları'nın varolmasına sebep olmuştur.

2.1.1 Kodlama

Verici kodlanmış paket oluştururken kaç veri parçası kullanacağını Robust Soliton dağılımına göre belirler [6]. Her seferinde bu dağılıma göre üretilen sayı kadar parça K adet veri parçası içerisinden rastgele seçilir. Seçilen paketlere birbirleriyle XOR işlemi uygulanır ve kodlanmış paket oluşturulur. Robust Soliton dağılımının özelliği pek çok defa kodlanmış paketlerin sadece bir adet veri parçası içermesidir.

2.1.2 Kod Çözümü

Bir adet veri parçası içeren kodlanmış paketler ile diğer kodlanmış paketler XOR işlemi yapılarak birden fazla veri parçası içeren kodlanmış paketlerin belirlenmiş

veri parçasından arındırılması sağlanır. Parçalar arındırıldıkça, yeni tek veri parçası içeren paketler oluşur ve orijinal veri çözülür.

2.2 Raptor Kodları

LT Kodları'nda karmaşıklık K arttığında sorun oluşturacağı için Raptor Kodları önerilmiştir [7]. Burada karmaşıklığı azaltmak için paketlerin derecesi düşürülüyor. Ancak bu durum bazı paketlerin uzun süre çözülememesine neden olabilir. Bu paketleri çözebilmek için oransız kodlamadan önce yapılan bir ön kodlama, düşük yoğunlukta eşlik denetimi kodlaması (Low Density Parity Check-LDPC) [18] yapılır. Böylece alıcıda oransız kodlar çözüldükçe elde edilemeyen veri paketleri LDPC kod çözümü sonucunda elde edilir. Raptor Kodları doğrusal karmaşıklığa sahiptir.

2.2.1 Kodlama

Raptor Kodları'nın kod çözme karmaşıklığı $O(K)$ 'dir. Doğrusal karmaşıklık, kodlanmış paket oluştururken derecesini sabit ortalama bir sayı, yaklaşık 3 [6], kullanarak sağlanır. Oransız kodlama işlemi yapılmadan önce bir ön kodlama yapılır. Ardından ortalama derece sayısına göre rastgele seçilen paketler XOR işlemine tabi tutulur. Ortalama sabit bir sayı kullanmak Luby Dönüşüm Kodları'nın sahip olduğu yüksek performansı düşürse de ön kodlama sayesinde yükünü hafifletir. Kodlama tekniği [7] çalışmasında ayrıntıları ile yer almaktadır.

2.2.2 Kod Çözümü

Daha düşük derece ile kodlanmış paketlerin kod çözme işleminin başarısız olma ihtimali düşük gibi görünse de derecesi bir olan paket bu kodlama tekniği için de şarttır. Alıcı kaynak paket sayısından biraz daha fazla kodlanmış paket aldığı anda iki aşamalı kod çözme işlemi gerçekleştirilir ve orijinal veri elde edilir.

2.3 Rastgele Doğrusal Fountain Kodlar

Rastgele Doğrusal Fountain Kod daha az ek veri yüküne sahiptir, ancak karmaşıklığı daha fazladır. Bu kodun kod çözme karmaşıklığı ise $O(K^3)$ 'dir.

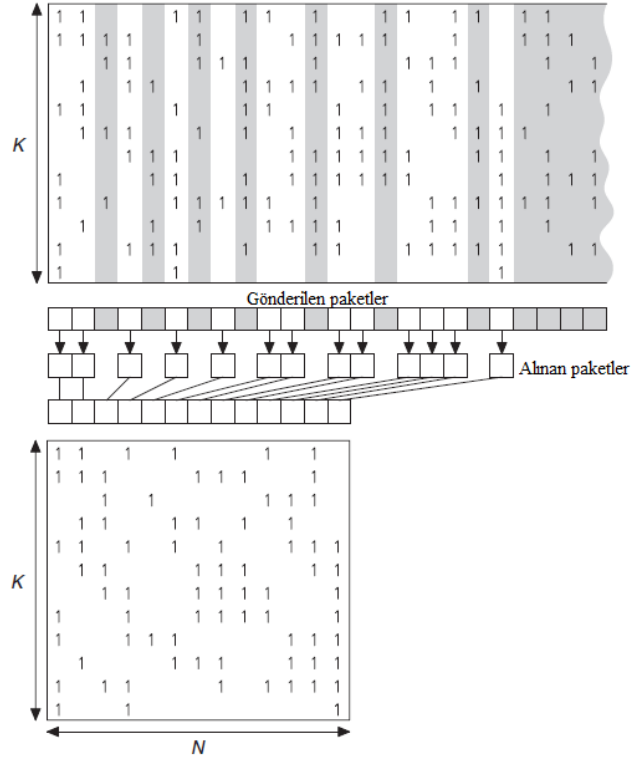
2.3.1 Kodlama

Rastgele Doğrusal Fountain Kodlama işleminde kodlanmış paket oluşturulurken her parça 0.5 ihtimalle dahil edilir. Yani Luby Dönüşüm Kodları'nın aksine pek çok parça kodlanmış pakette yer alabilir. Şekil 2.1'de K boyutlu verinin kodlanması gösterilmiştir. Her bir gönderimde (n) kodlayıcı K adet bit dizisi üreterek gönderilecek veri parçalarını belirler. Seçilen parçalara XOR işlemi uygulanarak kodlanmış paket elde edilir. (2.1)'de s_1, s_2, \dots, s_k veri parçalarından n . seferde gönderilecek kodlanmış paketin (t_n) elde edilmesi matematiksel olarak ifade edilmiştir. Buradaki toplam, ikilik tabanda yapılan bir toplamdır. G_{kn} , kodlayıcının ürettiği K adet bitlerden oluşturulan bir üreteç matrisidir. Bu matris gönderim süresince kodlanmış paket oluşturmak için üretilen K boyundaki bit dizilerinin eklenmesiyle uzar. Şekil 2.1'de en alttaki $K \times N$ 'lik matris G_{kn} 'dir. Burada kodlanmış paketlerde kullanılan veri parçaları 1, diğerleri 0 ile gösterilmektedir. Yine burada da alıcının kod çözme işlemi için hangi veri parçalarının kodlanmış pakette yer aldığına, yani G_{kn} bilgisine sahip olması gereklidir.

$$t_n = \sum_{k=1}^K (s_k G_{kn}) \quad (2.1)$$

2.3.2 Kod Çözümü

Daha öncede belirtildiği gibi oransız kodlarda alıcının hangi kodlanmış paketi kaybettiği bilgisi önemli değildir çünkü almış olduğu paketler birbirinin yerini tutabilir. Alıcı aldığı kodlanmış pakette hangi veri parçalarının kullanıldığı bilgisine sahip olmalıdır. Bu bilgi önceden alıcıya yüklenebileceği gibi, verici



Şekil 2.1: Rastgele Doğrusal Fountain Kodlar'ın Üretilmesi [6]

ve alıcının aynı şekilde rastgele sayı üretmesi sayesinde de elde edilebilir. Alıcı bir tür denklem çözme (Gauss eleme yöntemi) işlemi yapar. Alıcının oluşturduğu matris denkleminin derecesi (rank) K olana kadar yeni parçalar bekler. Alıcı orijinal veri parçalarının sayısından biraz daha fazla kodlanmış paket aldığı anda orijinal mesajı çözebilir.

3. APCO PROJE25 SİSTEMİ

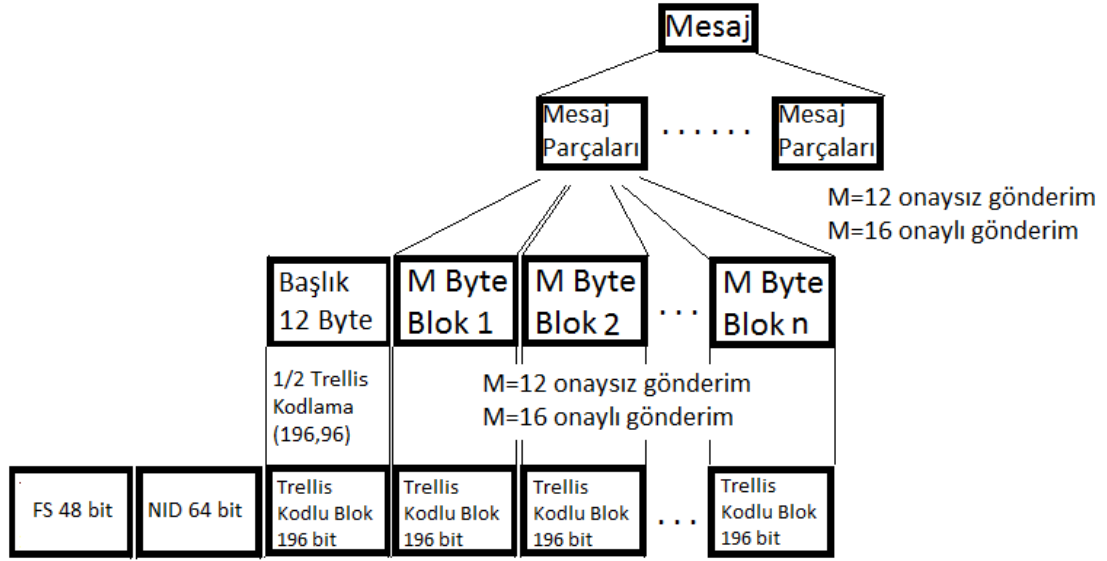
Kamu güvenliği telsiz ağları standardı olan APCO P25 dar bantlı veri bağlantısı tanımlar. Sayısal veri iletimi yapan bu sistemde ses iletiminin yanısıra veri iletimi de (resim, harita, form, kısa bilgiler) yapılabilir. Veri iletimi noktadan noktaya yapıldığı gibi çoğa gönderim (multicast/broadcast) şeklinde de olabilir.

APCO P25 standardında iletim yapan telsiz sistemleri çoğa gönderimi, bir paketi hedefteki tüm kullanıcılara teker teker gönderim yaparak gerçekleştirir. Bir kullanıcının dahi alımı başarısız olursa gönderme işlemi tekrar edilir. Bu durum sistemin verimsiz olmasına neden olmaktadır. Eğer haberleşme türü onaylı olursa alıcıların aldığı veya almadığı her paket için geri bildirim yapmaları gerekir. Bu durum da kanalda çakışmalara ve ekstra yüke neden olur. Bu standarda göre haberleşen telsizler üzerinde sistem yükünü azaltacak bir çalışma yapılması hedeflenmiştir ve oransız kodların kullanımı incelenmiştir.

Bu sistemde oransız kodlamanın sağladığı en büyük avantaj şudur: Alıcının hangi paketi alıp almadığı önemli değildir. Alıcı aldığı kodlanmış paketin hangi mesaj parçalarından oluştuğunu bildiği sürece paketler birbirinin yerini tutabilir. Alıcı mesajı oluşturan parçaların sayısından biraz daha fazla paket aldığında mesajı çözebilir. Bu sayede, alıcının aldığı her paket için geri bildirim yapmasına gerek yoktur, sadece çözme işlemini başardığına dair geri bildirim yapması yeterlidir. Bu çalışmada sistem için bir oransız kodlama tekniği olan Rastgele Doğrusal Fountain Kodlama uygulanmıştır.

3.1 APCO P25 Standardında Paket Yapısı

APCO P25'e göre gönderilecek paketlerin belli bir forma getirilmesi gerekir. Öncelikle mesaj parçalara ayrılır. Mesaj parçaları onaylı gönderim için 16 Byte,



Şekil 3.1: APCO P25 Standardında Paket Yapısı

onaysız gönderim için 12 Byte'lık bloklara bölünür. Her blok gönderim türüne göre sırasıyla 3/4 veya 1/2 Trellis kodlama yapılarak korunur. Trellis kodlaması sonunda bloklar 196 bitlik bloklar haline gelir.

Şekil 3.1'de görüldüğü gibi n bloğun başına 196 bit Trellis kodlanmış başlık, 64 bit ağ kimlik kodu (NID) ve 48 bit senkronizasyon bloğu eklenir ve bir paket oluşturulur.

Örneğin paketteki blok sayısı üç olacaksa bir paketteki bit yükü $3 \times 196 + 196 + 64 + 48 = 896$ bittir. Paket başarılı gönderilse de gönderilmese de bu yük gerçekleşir. Eğer gönderilen 3 blok Viterbi algoritmasıyla başarılı bir şekilde çözülebilirse 3×96 (onaysız iletimde) bitlik veri gönderilmiş olur. Blok sayısı artarsa paket hata oranı artar ama başlıklardan doğan ek veri yükü yüzdesi azalmış olur. Olabilecek en iyi sonuç kanalın verimli kullanılması için oldukça büyük bir önem taşır.

3.2 Rastgele Doğrusal Fountain Kodlar'ın Kullanımı

Sistem üzerinde uygulanan teknik kısaca şu şekildedir. Öncelikle, gönderilmek istenen mesajın uygun bir boyutta parçalamp K adet parçadan oluşturduğunu kabul edelim. Bu K adet mesaj parçasından her gönderim için rastgele parçalar seçilir. Seçilen parçalara bit bazından XOR işlemi yapılarak kodlanmış mesaj parçası oluşturulur. Alıcıya gönderilecek mesaj parçası XOR işleminden elde edilen parçadır. Alıcının K adet parçadan hangilerinin seçildiği bilgisine sahip olduğu kabul edilmiştir. Bu rastgele seçim, önceden belirlenen bir listeye göre veya verici ve alıcının aynı rastgele üretici kullanmasıyla yapılabilir. Alıcı kodlanmış pakete dahil olan mesaj parçaları için 1, diğerleri için 0 ile belirtilen bit dizilerinden bir matris oluşturur [6]. Bu matrisin birbirinden doğrusal bağımsız satır/sütun sayısı K olduğunda yani kertesesi K olduğunda alıcı kodlanmış paketlerden mesajı elde edebilir. Alıcı mesajı çözebilecek şekilde paket aldığıında bu paketlerin çözülmesi Gauss eleme yöntemine göre yapılmaktadır.

Kodlanmış pakette yer alan parçaların seçimi ve kod çözümü için farklı metodlar önerilmiştir. Örneğin LT Kodlarında [5] Robust Soliton Dağılımı ile genelde az sayıda parça seçilir. Bu sayede kod çözümü basitleşir. Raptor kodlamasında [7] ise oransız kodlamadan başka LDPC kodlama da yapılır. Bu sayede kod çözme karmaşıklığı daha da azalır. Ancak bu yöntemde verimliliğin yüksek (ek veri yükünün düşük) olması için mesaj çok büyük olmalıdır. Rastgele Doğrusal Fountain Kodlama'da ise her parçanın kodlanmış pakette yer alma ihtimali 0.5'tir. Kod çözme işlemi için ise Gauss eleme yöntemi kullanılır. Karmaşıklığı oldukça yüksek olmakla birlikte her durumda ek veri yükü çok azdır. APCO senaryosunda paket iletim süresi oldukça uzun ve mesaj boyutları oldukça kısa olduğu için kod çözme için yeterli zaman vardır, dolayısıyla karmaşıklık problem değildir. Ek veri yükünün de az olması nedeniyle Rastgele Doğrusal Kodlama yöntemi benimsenmiştir.

Paketlerin telsizlere sırayla gönderildiği ve her paket için geri bildirimlerin yapıldığı bir sistemde, sadece mesaj parça sayısından biraz daha fazla paket gönderimi/alımı ile sistemin daha verimli kullanılması sağlanabilir. Önerilen yeni

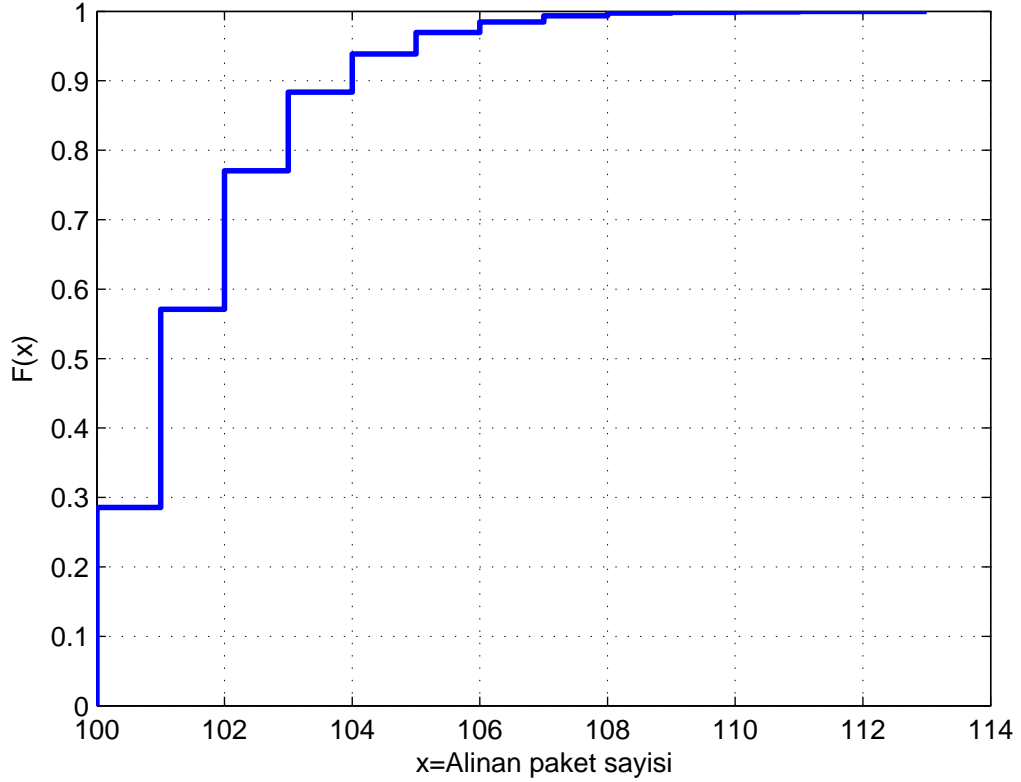
sistem ile alıcı tarafından olumlu veya olumsuz geri bildirimlerin ihtiyacı ortadan kalkacaktır. Alıcının sadece tüm mesajı çözdüğünde geri bildirim yapması yeterli olur. Çünkü vericiden gönderilen paketlerden hangisinin alınıp alınmadığı önemli değildir. Herhangi bir paketten eksik olan bir mesaj parçası elde edilebilir. Bu durum 1000 kullanıcı bir sistem için düşünüldüğünde oldukça büyük bir avantaj sağlar.

Rastgele Doğrusal Fountain Kodlama'ya göre alıcının paketi çözmesi için alması gereken paket sayısı 10000 deneme yapılarak incelenmiştir. Bu incelemede 4000 bit uzunluğunda 100 adet mesaj parçası oluşturulup 10000 kere deneme yapılmıştır. Örneğin Şekil 3.2'de görüldüğü üzere 6 adet ekstra paket alındığında yaklaşık olarak denemelerden %98'i mesajı oluşturulabilmiştir. Şekil 3.2'deki grafikten yola çıkarak %4-%8 gibi ekstra paket ile çoğa gönderim yapıldığı düşünelim. Bir de eski yöntemle, hangi paketlerin alınmadığı bilgisine ihtiyaç duyularak yapılan onaylı çoğa gönderimi düşünelim. Dar bantlı ve çok kullanıcı bir sistemde çoğa gönderim yeni yöntemle gerçekleştirildiğinde kanalda oluşabilecek yüklerin oldukça hafifleyeceği ön görülmektedir.

MATLAB'da yapılan zaman ölçümlerinde paket başına yaklaşık 12 msn süre gerektiği görülmüştür. Kod çözümü için APCO P25 sistemlerinde 1000 bitlik bir paketin bile 100 mili saniyeden fazla zamanda gönderildiği düşünülürse paket başına kod çözme süresi paket iletim süresinden çok daha azdır. Önerilen yeni sistem kod çözümünden dolayı ekstra zaman kaybına neden olmaz.

3.3 Sistem Modeli

Önerilen yöntemde varolan sistemde mesaj parçalara ayrıldıktan sonra, henüz hiçbir şifreleme ve kodlama işlemi gerçekleşmeden önce oransız kodlama işlemi yapılır. İletilecek dosya Şekil 3.3'de verilmiş olan sıralama doğrultusunda uygun büyüklükteki parçalara ayrılır. Bu parçalar 96 bitin katları şeklindedir. Mesaj parçaları üzerinden oransız kodlama işlemi uygulanır. Bu mesaj parçaları şifrelenir. Kanal kodlaması, gönderme yönünde hata düzeltimi (Forward Error Correction-FEC) yapılır.



Şekil 3.2: Rastgele Doğrusal Fountain Kodlamada İletimde Gereken Ek Veri Yükünün Dağılım Grafiği

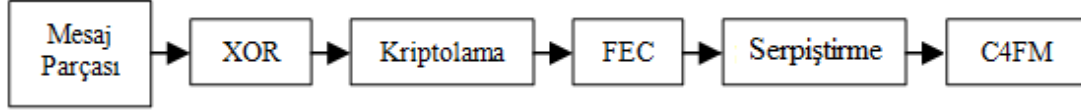
FEC aşamasında gönderim türünün onaylı veya onaysız olmasına göre 3/4 veya 1/2 Trellis kodlama yapılır. Trellis kodlama bit dizisinin 3/4 oranı için üç, 1/2 oranı için iki bite ayrılması ile başlar. Trellis kodlama sonlu durum makinesi olarak uygulanır. Bit dizisinin sonuna 000 veya 00 ekleyerek son durumunu başlangıç durumuna 0'a getirir. Trellis kodlayıcı 3/4 için 8-durumlu, 1/2 için 4-durumlu bir sonlu durum makinesidir. Her iki kodlama türü için durum geçiş tabloları Çizelge 3.1 ve Çizelge 3.2'de verilmiştir.

Çizelge 3.1: 1/2 Trellis Kodlayıcı Durum Geçiş Tablosu

	0	1	2	3
0	0	15	12	3
1	4	11	8	7
2	13	2	1	14
3	9	6	5	10

Çizelge 3.2: 3/4 Trellis Kodlayıcı Durum Geçiş Tablosu

	0	1	2	3	4	5	6	7
0	0	8	4	12	2	10	6	14
1	4	12	2	10	6	14	0	8
2	1	9	5	13	3	11	7	15
3	5	13	3	11	7	15	1	9
4	3	11	7	15	1	9	5	13
5	7	15	1	9	5	13	3	11
6	2	10	6	14	0	8	4	12
7	6	14	0	8	4	12	2	10



Şekil 3.3: Vericide Yapılan İşlemler

Trellis kodlaması ile 144 (3/4 için) veya 96 (1/2 için) bitlik bloklar 196 bitlik bloklara dönüştürülür ve bu kodlanmış bloklara serpiştirme (interleaving) işlemi uygulanır. Serpiştirme işleminin amacı 98 adet dibit bloklarının Rayleigh sönümlü kanalda toplu bir şekilde bozulmasını engellemektedir. Serpiştirme işleminde, ikili bit dizeleri Çizelge 3.3’de verilen tabloya göre yeniden sıralanır.

Oluşan bit dizisine kipleme işlemi uygulanır. Daha sonra oluşan paketlerin başına 196 bitlik (kodlanmış) başlık, 64 bit NID ve 48 bitlik senkronizasyon bloğu eklenir ve böylece kanaldan gönderilecek paketler oluşturulur. Alıcıda ise bu işlemlerin tersi yapılır ve oransız kod çözümünün ardından orijinal mesaj dosyası elde edilir.

Kodlama sonucunda oluşan veri C4FM kiplemesi ile kanaldan iletilir. Bu kipleme tipi bir tür 4-frekans kaydırmalı anahtarlama kiplemesidir ve her bir sembolde iki bit iletilir. Dört ayrı frekans vardır ve aralarında 1.2KHz boşluk mevcuttur. Öncelikle iki bitlik gruplar Çizelge 3.4’de görüldüğü gibi ∓ 1 , ∓ 3 sayılarına eşlenirler. Şekil 3.4’de verilen sıralamadaki gibi sembol dizisi yükseltilmiş kosinüs filtresinden geçirilir. Bunun ardından darbe şekillendirici filtreden geçen sinyal en sonunda bildiğimiz frekans kipleycisinden geçirilir. Frekans kipleycisi nedeniyle C4FM aslında bir sürekli faz kiplemesi (continuous phase modulation-CPM) çeşididir.

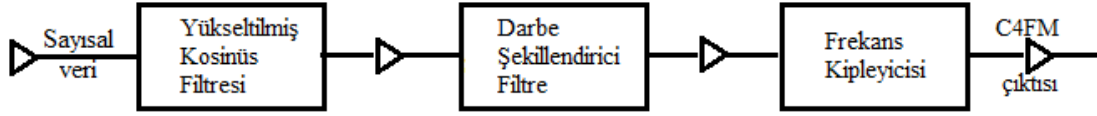
Çizelge 3.3: Serpiştirme Tablosu

Çıkış	Giriş	Çıkış	Giriş	Çıkış	Giriş	Çıkış	Giriş
0	0	26	2	50	4	74	6
1	1	27	3	51	5	75	7
2	8	28	10	52	12	76	14
3	9	29	11	53	13	77	15
4	16	30	18	54	20	78	22
5	17	31	19	55	21	79	23
6	24	32	26	56	28	78	30
7	25	33	27	57	29	79	31
8	32	32	34	56	36	78	38
9	33	33	35	57	37	79	39
10	40	32	42	56	44	78	46
11	41	33	43	57	45	79	47
12	48	34	50	58	52	80	54
13	49	35	51	59	53	81	55
14	56	36	58	60	60	82	62
15	57	37	59	61	61	83	63
16	64	38	66	62	68	84	70
17	65	39	67	63	69	85	71
18	72	40	74	64	76	86	78
19	73	41	75	65	77	87	79
20	80	42	82	66	84	88	86
21	81	43	83	67	85	89	87
22	88	44	90	68	92	90	94
23	89	45	91	69	93	91	95
24	96	-	-	-	-	-	-
25	97	-	-	-	-	-	-

Şekil 3.5’de yükseltilmiş kosinüs filtresinin frekans tepkisi görülmektedir. Filtre 40. dereceden bir filtredir ve dereceli azalma faktörü 0.2’dir; yani frekans bandı normalden %20 daha fazladır. Bu sayede semboller arası girişimin önüne geçilmiş olur. Yükseltilmiş kosinüs filtresinde sembol başına dört örnek alınmaktadır. Normalde 4800 örnek/sn olan örnek hızı, bu filtreden sonra dört katına çıkar. Yükseltilmiş kosinüs filtresinin frekans tepkisi (3.1) denkleminde verilmiştir.

Çizelge 3.4: Sembollerin Kipleme Fazlarına Eşlenişi

Bilgi Bitleri	Sembol	C4FM Sapması
01	+3	+ 1.8 KHz
00	+1	+ 0.6 KHz
10	-1	- 0.6 KHz
11	-3	- 1.8 KHz



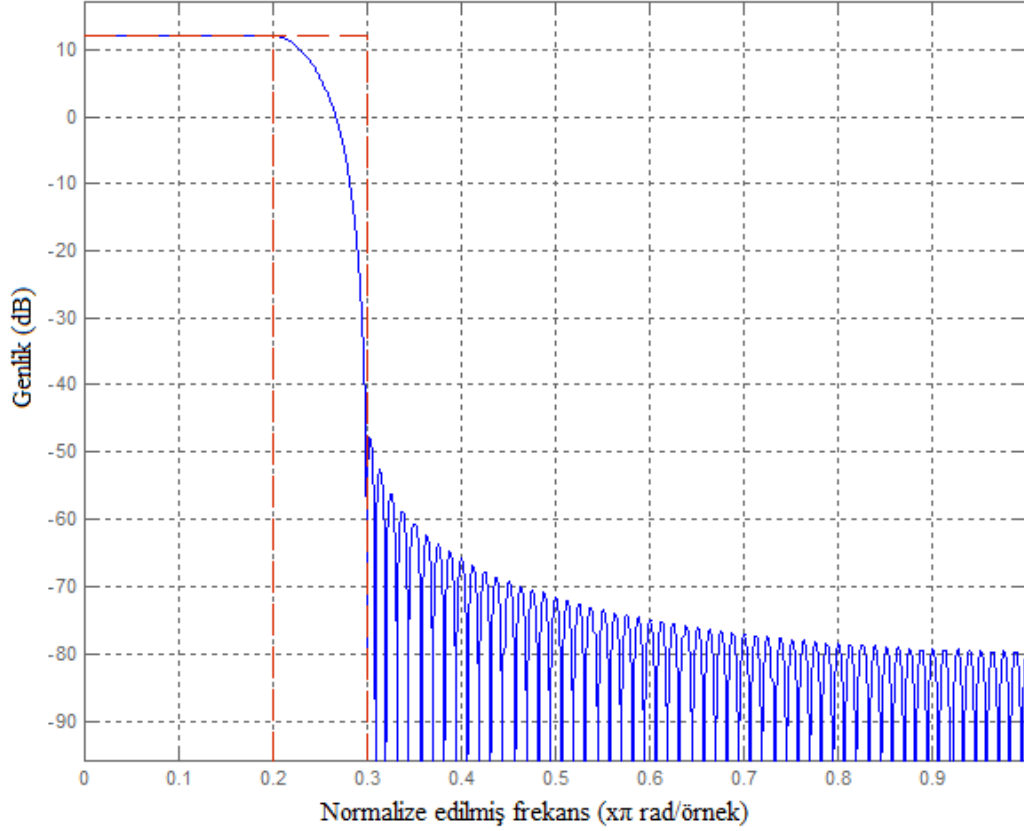
Şekil 3.4: C4FM Kiplemesi [19]

$$|H(f)| = \begin{cases} 1 & |f| < 1920Hz \\ 0.5 + 0.5 \cos(2\pi f) & 1920Hz < |f| < 2880Hz \\ 0 & |f| > 2880Hz \end{cases} \quad (3.1)$$

C4FM kiplemesinde sinyal yükseltilmiş kosinüs filtresinin ardından darbe şekillendirici filtresinden geçmektedir. (3.2)'de frekans tepkisi verilen bu filtre, geçirme bandında ters sinc fonksiyonu şeklindedir. Filtre frekans tepkisi Şekil 3.6'da verildiği gibidir. Bu filtrenin amacı, alıcıda kullanılan biriktir-boşalt (integrate-dump) filtresinin sinyali bozmasını engellemektir. Darbe şekillendirici filtreden geçen sinyal daha sonra 600 Hz frekans sapmalı bir frekans kipleyicisinden geçirilir.

$$|P(f)| = \begin{cases} \frac{\pi f}{4800} \sin\left(\frac{\pi f}{4800}\right) & |f| \leq 2880Hz \\ 0 & |f| > 2880Hz \end{cases} \quad (3.2)$$

Toplanır Beyaz Gauss Gürültülü (TBGG) kanaldan geçen C4FM sinyali alıcıda Şekil 3.7'de işlem sıralaması verilen C4FM kip çözücünden geçirilir. Biriktir-boşalt filtresi her ardışık dört örneğin ortalamasını alır. Böylece elde edilen örnekler darbe genlik kip çözücüye iletilir. Elde edilen örnekler en yakınlık kriterine göre [-3, -1, 1, 3] sembollerine eşlenir. İletilen ve alınan sembollere karşılık gelen bitler bulunur ve buradan hata oranları hesaplanabilir. Stokastik gradyan sembol toplama aşaması önce bahsedilen darbe genlik kip çözümünü yaparken



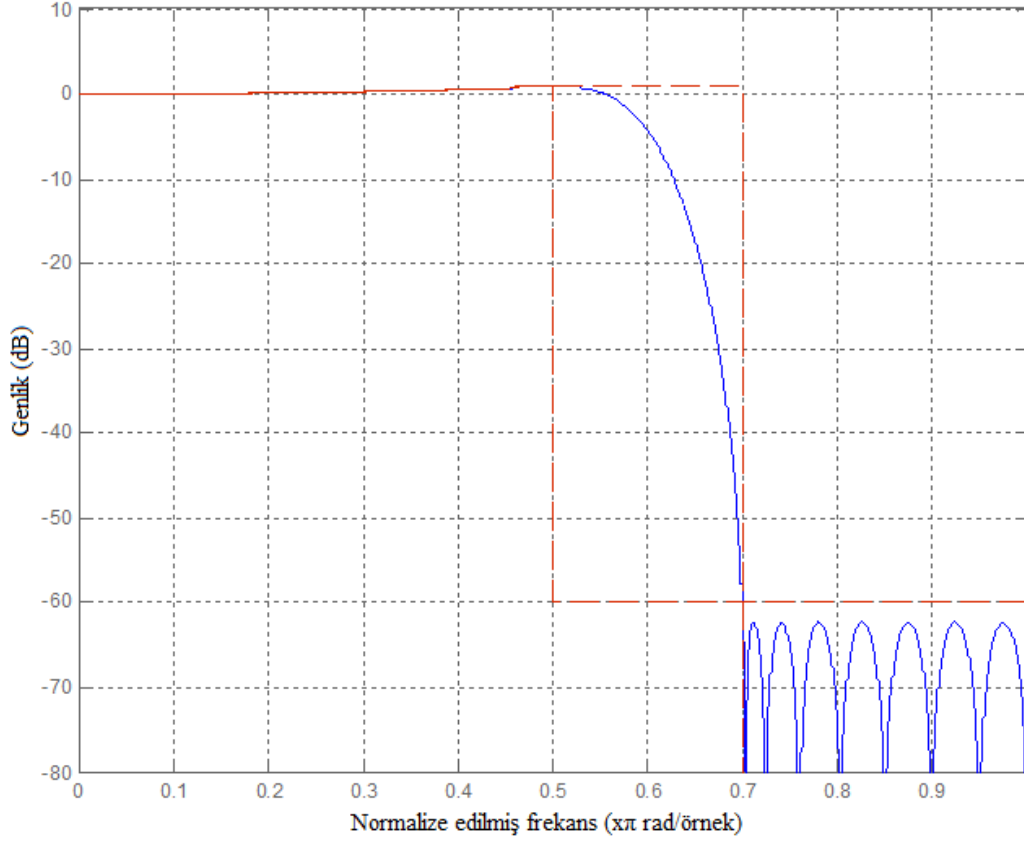
řekil 3.5: Ykseltilmiř Kosins Filtresinin Frekans Tepkisi

aynı zamanda senkronizasyonu da saęlar. Kurduęumuz benzetim modelinde senkronizasyon problemi olmadıęı varsayılmıřtır. Frekans kipleyci ve kip czc tabanbantta gerekleřtirilmiřtir.

Kip czm yapıldıktan sonra serpiřtirme iřleminin tersi olarak yine izelge 3.3'e gre ikili bitler sıralanır. Tekrar sıralanan bu bit dizelerine Trellis kod czm uygulanır. Bu kod czme iřlemi Viterbi algoritmasına dayalıdır. Ardından Fountain kodlu paketleri de Gauss eleme yntemi ile kod czm yaparak orijinal paketler elde edilir.

3.4 Benzetim Sonuęları

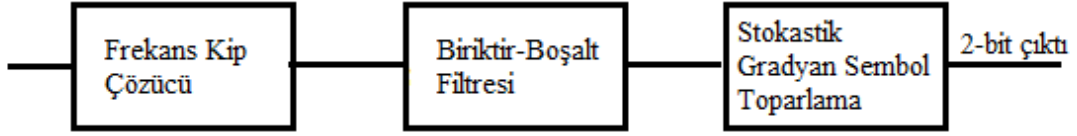
Daha nce APCO P25 modelinde herhangi uzunlukta bir veri mesajının tařıma katmanında paralara ayrılıp, oransız kodlamanın bu paralar zerinde yapıldıęı



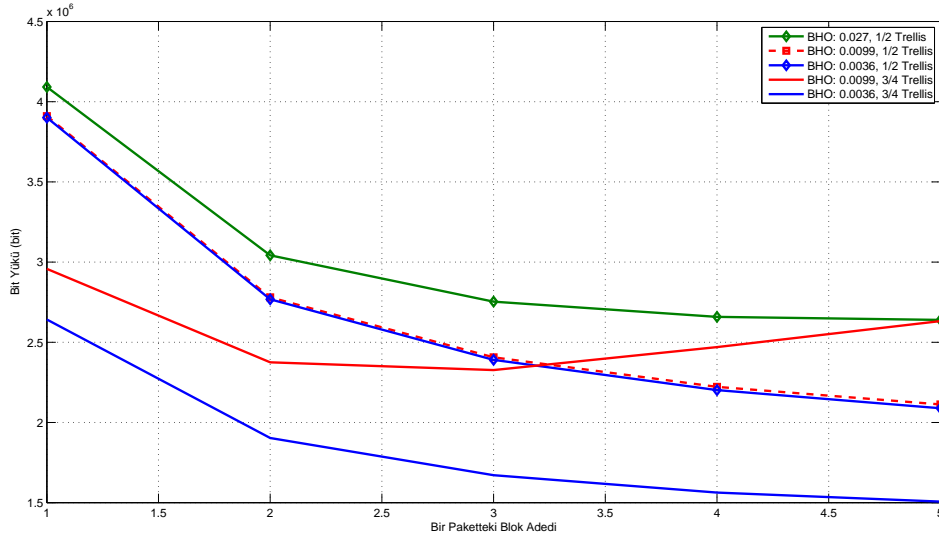
řekil 3.6: Darbe řekillendirici Filtrenin Frekans Tepkisi

belirtilmiřti. Daha sonra her bir (kodlanmıř) para n adet 196 bitlik bloklara ayrılır. Eęer onaylı trafikte iletiliyorsa (mesela veri) kod oranı $3/4$ 't r ve Trellis kodlaması kullanılarak 144 veri biti 196 bitlik bloęa evrilir. Onaysız trafikte iletiliyorsa (mesela ses), kod oranı $1/2$ 'dir ve Trellis kodlaması kullanılarak 96 bitlik veri 196 bitlik bloęa evrilir. n adet veri bloęunun bařına 196 bitlik (kodlanmıř) bařlık, 64 bit NID ve 48 bitlik senkronizasyon bloęu eklenir ve paket oluřur.

$1/2$ ve $3/4$ Trellis kodlama iin toplam bit y k  sonuları farklı blok sayısına g re performansı řekil 3.8'de verilmiřtir. Grafiđ 100 KB saf veri bařarılı iletilene kadar oluřan bit y k n  g stermektedir. Grafiđin yatay eksenini bir pakette yer alan blok sayısı, dikey eksenini ise oluřan bit y k d r. Paket bařarılı g nderilse de g nderilmese de kanalda bir y k  oluřturur. Pakette 3 blok olduęundaki  rneęi d ř n rssek, paketin bařarılı olması durumunda (yani 3 blokta da Viterbi kod  z m  bařarılı sonulanırsa) $3 \times 96 = 288$ bitlik veri g nderilmiř olur. Blok sayısı



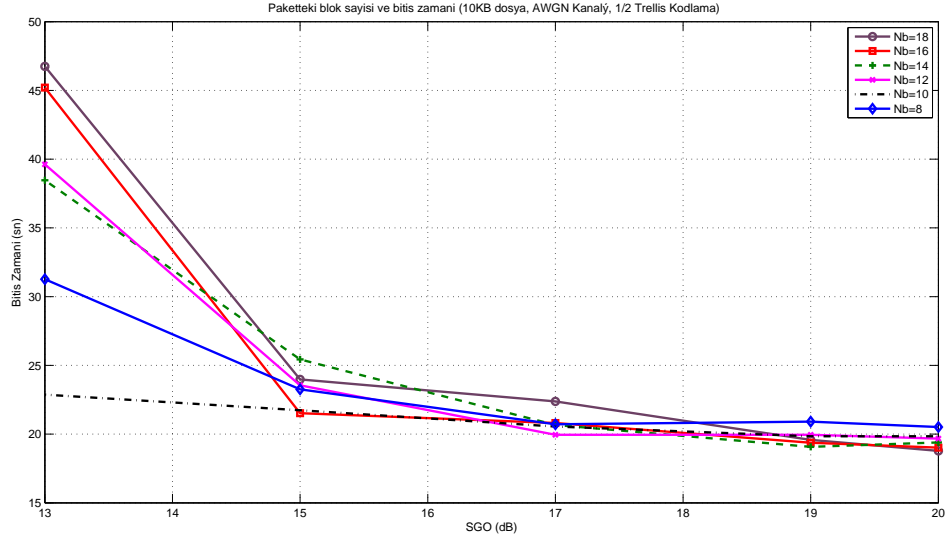
Şekil 3.7: C4FM Kip Çözücü [19]



Şekil 3.8: Farklı Hata Oranlarında Farklı Blok/Paket Oranları İçin Bit Yükü Grafiği

artarsa paket hata oranı artar ama başlıklardan doğan ek veri yükünün yüzdesi azalmış olur. 0.0036 (düşük) bit hata oranı için blok sayısı arttıkça verimin arttığı görülmektedir (her iki kod oranı için). Ancak 3/4 kod oranı, kodlamadan doğan yük az olduğundan, daha az bit yüküne sahiptir. 0.0099 hata oranında ise düşük blok sayıları için 3/4 kod oranı daha iyidir, ancak yüksek blok sayılarında 1/2 kod oranı ile verimi yükseltmek mümkündür. Yüksek hata oranında ise kesinlikle 1/2 kod oranı daha iyidir.

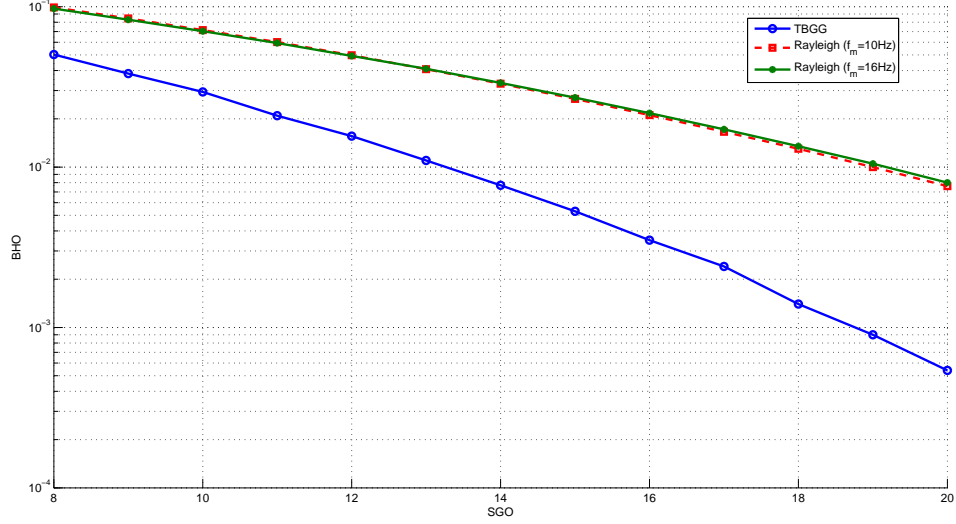
Paket iletim süresi sistemdeki en düşük SGO değerine sahip kullanıcı tarafından belirlenir. Yapılan çalışmalarda 10KB dosyanın APCO P25 sisteminde iletim süresi farklı SGO değerleri için incelenmiştir. Sistem benzetimi TBGG kanalında 1/2 Trellis kodlama için gerçekleştirilmiştir. Benzetim sonunda elde farklı SGO değerlerinde elde edilen bitiş süresi grafiği Şekil 3.9'da verilmiştir.



Şekil 3.9: Paketteki Blok Sayısına (N_b) göre SGO Bitiş Zamanı Grafiği

Şekil 3.9'da görüldüğü üzere, SGO 13dB'den küçük olduğunda gecikme çok artmakta ve sistem çalışamaz hale gelmektedir. Bu değer APCO ile ilgili geçmişte yapılmış analizlerle uyumludur. Ayrıca, paketteki blok sayısı (N_b) arttığında ek yük azalır ve dolayısıyla daha iyi SGO değerlerinde gecikme süresinde iyileşme olur. SGO düşük olduğunda ise paketteki blok sayısını artırmak pakette hatalı blokların olma ihtimalini artırır. Paketteki bir blok bile hatalı olduğunda paket değerlendirilemeyeceğinden dolayı gecikme süresi uzar. Çoğa gönderimde parametreleri belirlerken SGO değeri en kötü kullanıcıyı göz önünde bulundurup blok sayısını düşük tutmak daha iyi bir tercih olabilir.

Şekil 3.10 ve Şekil 3.11'de Rayleigh sönümlü kanal için C4FM bit hata oranı (BHO) ve blok hata oranı (BLHO) performansları görülmektedir. Elde edilen sonuçlar [20] ve [21] makalelerindekiyle benzerdir. [21] çalışmasında belirtildiği üzere APCO'da bit hata oranı 0.05'i aşmamalıdır. TBGG kanalı için minimum çalışabilir sinyal gürültü oranı (SGO) 8dB'dir. Rayleigh kanalında ise SGO 13dB'ye eşit veya büyük olmalıdır. Şekil 3.11'de görüldüğü üzere, TBGG kanalında SGO 20dB olduğunda neredeyse hiçbir blok hatası olmamaktadır. Rayleigh kanalında ise aynı SGO değerinde her 10 blokta birden fazla blok hatalı olmaktadır. Rayleigh sönümlenmesinde 10Hz ve 16Hz Doppler frekansları arasında başarımlı açısından fazla bir fark görülmemektedir. [21] makalesinde de



Şekil 3.10: TBGG ve Rayleigh Kanallarında Sinyal Gürültü Oranı - Bit Hata Oranı Grafiği

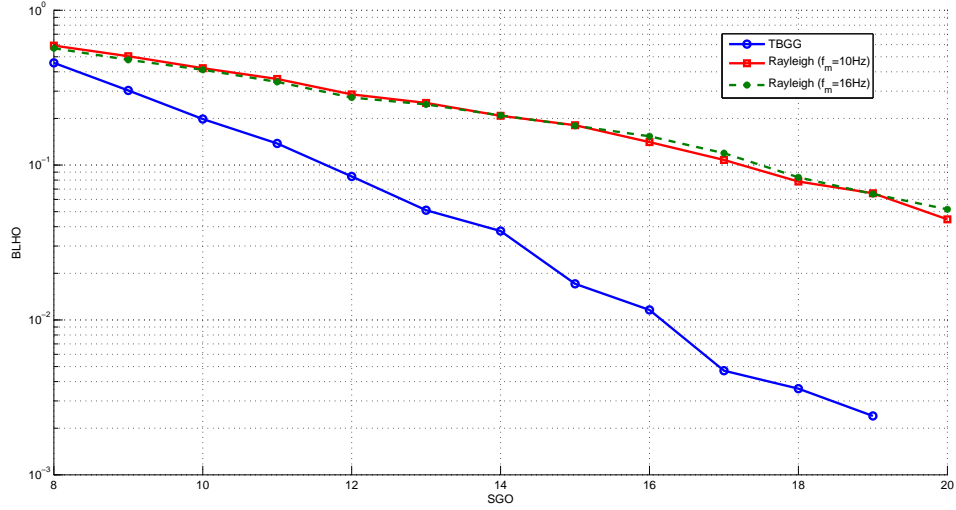
8-20dB SNR aralığı için benzer sonuçlar gözlemlenmiştir.

Şekil 3.12'de ise 10KB dosyanın APCO P25 sisteminde iletim süresinin paket blok sayısına (N_b) göre değişimi görülmektedir. Dosya iletim süresi, paketteki blok sayısı ve blok hata oranı cinsinden (3.3)'de verildiği gibidir.

$$D = \frac{10KB}{96N_b} \times \frac{64 + 196 + 196N_b}{9600 \times (1 - BLHO)^{N_b}} \quad (3.3)$$

Burada $96N_b$ paket başına düşen veri biti sayısıdır. $64 + 196 + 196N_b$ ise paketin toplam boyudur. $(1 - BLHO)^{N_b}$ ise bir paketin doğru alınma ihtimalidir (yani bütün blokların doğru alınma ihtimali). Bir paket en fazla 512B olacağından, N_b en fazla 19 olabilir.

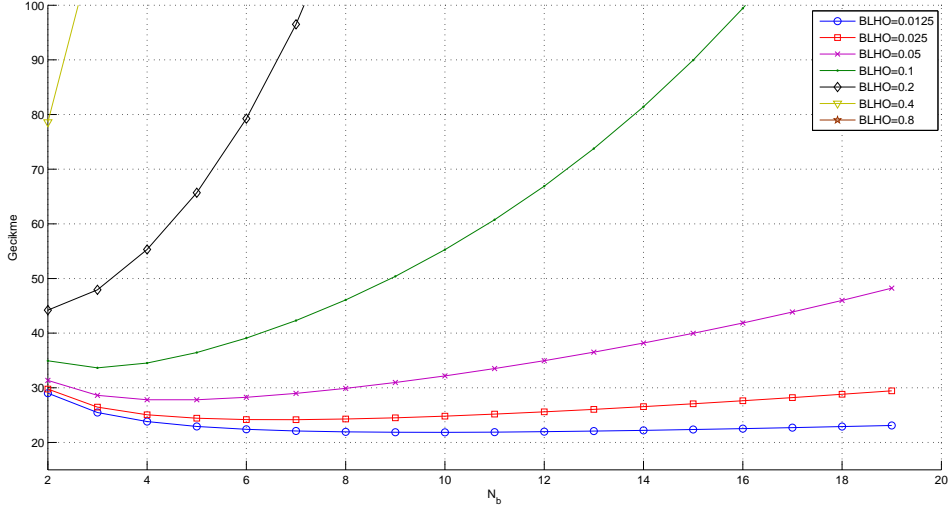
Blok sayısı N_b arttığı zaman ek yük azalmakta, dolayısıyla SGO iyi olduğunda gecikme iyi olmaktadır. Ancak, SGO düşük olduğunda blok sayısını artırmak, bloklardan en az birinin hatalı olma ihtimalini arttırdığından gecikmeyi arttırmaktadır. Dolayısıyla çoğa gönderimde en kötü kullanıcıya göre parameteleri belirlemek gerektiğinden blok sayısını düşük tutmak daha iyi bir tercih olabilir. Şekil 3.11 ve Şekil 3.12'ye bakarak aşağıdaki gözlemleri yapabiliriz.



Şekil 3.11: TBGG ve Rayleigh Kanallarında Sinyal Gürültü Oranı - Blok Hata Oranı Grafiği

- Blok hata oranı (BLHO) 0.01'den küçük olduğunda N_b 'yi mümkün olduğu kadar artırmak gerekir. Örneğin; TBGG kanalında en uzak kullanıcı için SGO 16dB değerinden büyük olduğunda BLHO 0.01'den küçük olur. Rayleigh sönümlenmeli kanalda ise BLHO 0.01'den küçük olması için SGO'nun çok yüksek olması gerekmektedir. Çoğa gönderimde en uzaktaki kullanıcının böyle bir sinyal girişim oranına sahip olması pek mümkün değildir.
- BLHO; 0.025, 0.05, 0.1 değerlerine sahip olması durumunda ise optimal N_b değeri 2 ile 19 arası bir değerdir.
- BLHO; 0.2, 0.4, 0.8 gibi değerlere sahip iken N_b 'yi 1 seçmek gerekmektedir. Örneğin; TBGG kanalında SGO 8, 9, 10dB olduğu durumlarda N_b 'yi 1 seçmek uygundur. Rayleigh kanalında ise SGO 16dB'den küçük olduğu durumlarda (en uzak kullanıcı çok büyük ihtimalle böyle bir SGO değerine sahiptir) N_b değerini 1 olarak seçmek gerekmektedir.

Görüldüğü üzere SGO'nun ve kanal tipinin ne olduğuna bağlı olarak paketin içindeki optimal blok sayısı büyük değişiklik göstermektedir.



Şekil 3.12: Değişik Blok Hata Oranları için 10KB Dosya İletim Süresi

3.5 Sonuçlar

Bu bölümde APCO P25 protokolüne göre haberleşen telsiz sistemlerinde oransız kodların kullanılması incelenmiştir. Varolan bir sistem üzerinde mesaj parçalarının oransız kodlama algoritması uygulanarak paketlerin gönderimi gerçekleştirilmiştir. Onaylı gönderimde her paket için geri bildirim ve alınamayan paketlerin tekrar gönderimini gerektiren durumlarda kanalda oluşan yüklenmelerin azaltılması amaçlanmıştır. Sistem yeni haliyle sadece mesaj alıcı tarafından tamamen çözüldüğünde onay mesajı göndermesini gerektirecektir. Ayrıca, alıcının hangi paketi alıp alınmadığıyla da ilgilenilmesine gerek kalmayacaktır.

Yapılan çalışmalarda pakette bulunan blok sayısına karşılık kanal üzerinde oluşacak bit yükü incelenmiştir. Yüksek hata oranında ise 1/2 kod oranı ile Trellis kodlama daha iyi bir performans göstermiştir. Diğer bir çalışma olarak, farklı SGO'lara göre paketlerin iletim süresi incelenmiştir. Yüksek SGO değerlerinde paketteki blok sayısının artışı ek yükü azaltacağından gecikme süresinde azalma sağlar. Düşük SGO değerlerinde ise paketteki blok sayısının artışı hatalı paket sayısını artıracığından gecikmeyi de artıracaktır. Sistemin verimli kullanılması açısından ağda bulunan en kötü SGO değerine sahip kullanıcıya göre blok sayısı belirleyerek gecikme süresinin optimum bir değerde kalması sağlanabilir. Ayrıca

TBGG ve Rayleigh kanallarında bit ve blok hata oranları elde edilmiştir. Önerilen sistemin verimli çalışması için SGO için bir alt sınır belirlenmiştir ve önceki çalışmalarla sonuçların uyumlu olduğu görülmüştür. Kanal modeline ve SGO değerine bağlı olarak paket içerisinde yer alacak blok sayısı sistem verimliliği üzerinde büyük ölçüde etkiye sahip olduğu gözlenmiştir.

4. APCO PROJE25 SİSTEMLERİNDE GÜVENLİ VE VERİMLİ ÇOĞA GÖNDERİM

Çoğa gönderim günümüz haberleşme sistemlerinde önemli bir uygulamadır. Özellikle kablosuz sistemlerde çoğa gönderimin uygulanması bir takım sebeplerden dolayı zordur. Hepsinden önce, gönderilen her paketten sonra onay beklemek çok verimsizdir. İkinci olarak, çoğa gönderim hızı/süresi en kötü kanal kalitesine sahip kullanıcı tarafından belirlenir. Fountain kodlar bu verimsizlikleri önlemek için önerilmiştir. Fountain kodlama özellikle yayın uygulamalarında verimli olan bir tekniktir. Bu teknikte, verici mesajı K bloğa böler ve bu K blok kümesinde rastgele bloklar seçer. Seçilen bloklara XOR işlemi uygulanarak elde edilen kodlanmış paket gönderilir. Bu süreç, alıcının orijinal veriyi elde etmesine kadar tekrar eder. Burada, alıcılar belli bir bloğu almak zorunda değillerdir. Yeterli sayıda kodlanmış paket biriktirmiş olması gerekir. En büyük avantajı, alıcıların aldıkları her paket için ayrı bir onay göndermelerine gerek yoktur. İkinci avantajı ise tekrar gönderime gerek yoktur. Eğer Fountain kodlama kullanılmazdı, çok kullanıcı kablosuz sistemde neredeyse her paketin tekrar gönderilmesi gerekebilirdi.

Çoğa gönderim içeren bir diğer kablosuz teknoloji Kamu Güvenliği Telsiz Teknolojileri'dir (Public Safety Radio Technology) (APCO P25 [2], TETRA, DMR). Bu hücreli ağlar geniş kapsama alanlı, çok kullanıcı ve dar bantlı olarak nitelendirilirler. Ayrıca kullanıcıların kanal durumu kullanıcıdan kullanıcıya büyük farklılıklar gösterebilir. Normalde baz istasyonu kullanıcıların kanal durum bilgisine sahip değildir. Fountain kodlar APCO sistemlerde tekrar gönderimlerden ve her pakette geri bildirim mesajlarından kurtulmak için kullanılabilir. Baz istasyonunun kodlanmış blokları bir süre gönderdiğini ve sonrasında düğümlerin o ana kadar toplanmış olduğu paket sayısını içeren geribildirim mesajı gönderdiğini varsayılmıştır. Farklı düğümlerden gönderilen onay paketleri birbirleriyle çarpışabilirler. Baz istasyonunun kullanıcı sayısını ve

kimliklerini bilmediğini kabul edilmiştir. Bu nedenle zaman dilimli rastgele erişim mekanizmasının kurulması iyi bir çözüm olabilir. Rastgele erişim mekanizması sistemdeki kullanıcı sayısını tahmin etmek için gönderimden önce kullanılabilir. Rastgele erişim mekanizması çoğa gönderimden sonra geribildirim mesajları için de kullanılabilir.

Bu bölümde öncelikle, çoğa gönderim grubunda kullanıcı sayısını tahmin etmek için çok aşamalı mekanizma önerilmiştir. Bunu veri gönderme evresi takip eder. Sonrasında geri bildirim evresine dayalı başka bir rastgele erişim mekanizması kod çözümü yapmamış kullanıcı sayısını ve kullanıcıların güncel durumunu tahmin etmek için başlatılır. Veri gönderim ve geribildirim aşamaları tüm kullanıcıların başarılı şekilde kod çözümü yapmasına kadar döngüsel olarak gerçekleştirilir.

4.1 Kullanıcı Sayısı Kestirimi

4.1.1 Tek Aşamalı Kestirim

Baz istasyonunun T zaman aralığı bildirdiğini ve her kullanıcının bu aralıklardan rastgele birine eriştiklerini kabul edin. Eğer sadece bir kullanıcı bir zaman aralığına erişirse, o zaman aralığı başarılı olarak ifade edilir. Eğer hiç bir kullanıcı bir zaman aralığına erişemezse, o zaman aralığı boş olarak belirtilir. i ve s , boş ve başarılı aralık sayısı olsun. Baz istasyonunun amacı i , s ve T 'den kullanıcı sayısı U 'yu tahmin etmektir.

[22] çalışmasında gönderilecek pakete sahip kullanıcı sayısı, sistemi kararlı hale getirmek için tahmin ediliyor. [23]'de başarılı ve çakışan kullanıcı sayısını elde eden bir model öneriliyor. Bu çalışmalardan farklı olarak, kullanıcı sayısı tahmin edilirken boş zaman aralık sayısı da hesaba katılmıştır.

4.1.1.1 Maksimum Olabilirlik (MO) Kestirimi

Öncelikle $i + s$, T 'ye eşit veya küçük olmalıdır. İkinci olarak, başarılı ve çakışmış aralıklarda toplam gönderim sayısı en az $s + 2(T - s - i)$ olmalıdır. U kullanıcı aralıklara rastgele eriştiğinde, T aralıkta i boş ve s başarılı aralık olasılığı şu şekilde yazılabilir:

$$p_{T,U}(i, s) = 0, \text{ eğer } i + s > T \text{ veya } s + 2(T - s - i) > U \quad (4.1)$$

$$p_{T,U}(i, s) = \frac{C_i^T C_s^{T-i} \frac{U!}{(U-s)!}}{T^U} \times \sum_{k_1, k_2, \dots, k_{T-i-s} \in \mathcal{K}} \sum \dots \sum \frac{(U-s)!}{k_1! \times k_2! \times \dots \times k_{T-i-s}!} \quad (4.2)$$

Burada, $C_a^b = \frac{b!}{a!(b-a)!}$.

\mathcal{K} $T - s - i$ kutu kümesine karşılık gelmektedir. Bu ifade U topun T kutuya atılmasıyla (T^U olası kombinasyon) kutulardan i tanesi boş (C_i^T kombinasyon), kalan kutulardan s tanesi tek topa sahip ($C_s^{T-i} \frac{U!}{(U-s)!}$ olası kombinasyon) ve geri kalan $U - s$ topların $T - s - i$ kutuda her kutuda en az iki top (\mathcal{K} set olarak tanımlanan) olacak şekilde yer almasına denk gelir.

T aralıkta i boş ve s başarılı aralıklar verildiğinde, maksimum olabilirlik kestirimine göre kullanıcı sayısı,

$$U^{MO} = \arg \max_{u \geq s + 2(T - s - i)} \{p_{T,u}(i, s)\} \quad (4.3)$$

Baz istasyonunun kullanıcı sayısı ile ilgili herhangi bir istatistiksel bilgiye sahip olmadığı kabul edilmiştir.

Çizelge 4.1'de maksimum olabilirlik kestirimi ile elde edilen sonuçlarda tahmin hatasının ortalama kare kökü gösterilmiştir. Bu kestirim yöntemi birçok kullanıcı sayısı ve zaman aralıklarında test edilmiştir. Her 1000 durum için benzetimler gerçekleştirilmiştir ve tahmin hatasının ortalama kare kökü hesaplanmıştır.

Çizelge 4.1: Maksimum Olabilirlik Kestirimi: Kestirim Hatasının Ortalama Kare Kökü

	T=10	T=15	T=20	T=25
U=10	1.169	0.634	0.521	0.438
U=12	1.568	0.967	0.667	0.534
U=15	2.153	1.426	1.132	0.799
U=16	2.286	1.552	1.221	0.885
U=20	1.935	×	×	×

Kullanıcı sayısı, zaman aralık sayısından küçük olduğunda elde edilen sonuçlar kestirimin doğru olduğunu göstermiştir. Bu koşullar altında ortalama kare kökü genellikle birden küçüktür. Çizelge 4.1’de × ile gösterilen yerlerde olasılık hesaplanması oldukça uzun bir süre aldığından sonuçlar yazılamamıştır.

4.1.2 Yaklaşık Maksimum Olabilirlik (YMO) Kestirimi

Yukarıdaki kestirici hatasız olmasına rağmen, kombinatorik hesaplama gerektirir. Bu hesaplama da kullanıcı sayısı ve zaman aralık sayısı arttıkça karmaşık bir işlem haline dönüşür. [23]’deki çalışma, her zaman aralığına Poisson hızı U/T erişim ile dilimli rastgele erişim sistemine başarılı bir şekilde yaklaşır. Bir aralığın boş olma olasılığı yaklaşık olarak $e^{-U/T}$ ve bir aralığın başarılı olma olasılığı ise yaklaşık olarak $\frac{U}{T}e^{U/T}$, dir. U kullanıcı aralıklara rastgele eriştiğinde, T aralık içerisinde i boş ve s başarılı aralık olasılığı yaklaşık olarak,

$$\tilde{p}_{T,U}(i, s) = \frac{T!}{i!s!(T-s-i)!} \times \left(\frac{U}{T}e^{-\frac{U}{T}}\right)^s \left(e^{-\frac{U}{T}}\right)^i \left(1 - \frac{U}{T}e^{-\frac{U}{T}} - e^{-\frac{U}{T}}\right)^{T-s-i} \quad (4.4)$$

$\tilde{p}_{T,U}(i, s)$ yaklaşık olasılığı kullanılarak, U^{YMO} değerini bulan yaklaşık maksimum olabilirlik kestirici şu şekildedir:

$$U^{YMO} = \arg \max_{u \geq s+2(T-s-i)} \{\tilde{p}_{T,u}(i, s)\} \quad (4.5)$$

Çizelge 4.2: Yaklaşık Maksimum Olabilirlik Kestirimi: Kestirim Hatasının Ortalama Kare Kökü

	T=10	T=15	T=20	T=25
U=10	1.036	0.95	0.645	0.443
U=12	1.520	0.901	0.810	0.703
U=15	2.255	1.380	1.003	0.864
U=16	2.494	1.515	1.114	0.894
U=20	x	2.152	1.676	1.38

Çizelge 4.2’de yaklaşık maksimum olabilirlik kestirimi ile elde edilen sonuçlarda tahmin hatasının ortalama kare kökü gösterilmiştir. Sonuçlar, YMO kestiriminin MO kestirimi kadar iyi, hatta bazı durumlarda daha iyi olduğunu göstermiştir. Ayrıca sonuçlar özellikle zaman aralık sayısının kullanıcı sayısından büyük olduğunda kestirimin başarılı olduğunu göstermiştir. En önemlisi, yaklaşımsal teknik kombinatorik olarak daha basittir ve önemli ölçüde kısa zaman almaktadır.

Çizelge 4.3: Yaklaşık Maksimum Olabilirlik Kestirimi: Kestirim Hatasının Ortalama Kare Kökü

	T=100	T=200	T=300	T=400	T=500
U=100	3.532	1.841	1.315	0.952	0.888
U=200	10.424	5.074	3.444	2.678	2.206
U=300	19.74	9.481	6.426	4.790	3.852
U=400	37.113	14.077	9.661	7.485	6.013
U=500	88.132	20.766	13.291	10.241	8.060

Kombinatorik basitliğinin sayesinde, yaklaşımsal metot daha çok kullanıcı ve zaman aralığı için test edilebilmiştir. Çizelge 4.3, yaklaşık maksimum olabilirlik kestiriminin 100’den 500’e kadar kullanıcı sayısı ve 100’den 500’e kadar zaman aralığı için sonuçları göstermektedir. Zaman aralığı sayısı kullanıcı sayısından büyük olduğunda ortalama kare kökü hatası kullanıcı sayısının %1’ine yaklaşır ki bu da neredeyse hatasız bir kestirimdir.

İyi bir kestirim için, sonuçlar zaman aralık sayısının kullanıcı sayısından fazla olması gerektiğini göstermiştir. Aksi takdirde kullanıcı sayısını bilemeyiz.

Çizelge 4.4: Yaklaşık Maksimum Olabilirlik Kestirimi: Çok Aşamalı Kestirim Performansının Ortalama Hata ve Ortalama Kare Kökü

	K=6	K=7	K=8
U=100	[0.973, 5.442]	[0.551, 3.416]	[0.590, 2.094]
U=200	[3.722, 17.127]	[1.351, 8.069]	[0.643, 4.865]
U=300	[16.42, 60.232]	[1.624, 14.662]	[0.792, 8.055]
U=400	×	[2.733, 23.234]	[0.844, 11.505]
U=500	×	[5.376, 37.973]	[1.539, 15.427]

4.1.3 Çok Aşamalı (ÇA) Kestirim

Yukarıdaki ilk benzetim sonuçları, eğer rastgele erişim aralıkları yeterli ise kullanıcı sayısının yeterli doğrulukta tahmin edilebileceğini gösterir. Aksi takdirde baz istasyonu kullanıcı sayısı üzerine bir fikri olmadığından yukarıdaki strateji başarılı olamaz.

Çok aşamalı stratejinin k . aşamasında, baz istasyonu $T_k = 2^k$ rastgele erişim aralığı açar. Kullanıcılar bunlardan bir tanesini rastgele seçer ve gönderir. i_k ve s_k k . aşamadaki boş ve başarılı aralık sayısı olsun. k . aşamanın sonunda, önerilen çok aşamalı kestirici şöyledir:

$$U_k^{MO-CA} = \arg \max_{u \geq \max_j \{s_j + 2(T_j - s_j - i_j)\}} \left\{ \prod_{j=1}^k p_{T_j, u}(i_j, s_j) \right\} \quad (4.6)$$

$$U_k^{YMO-CA} = \arg \max_{u \geq \max_j \{s_j + 2(T_j - s_j - i_j)\}} \left\{ \prod_{j=1}^k \tilde{p}_{T_j, u}(i_j, s_j) \right\} \quad (4.7)$$

$p_{T_j, u}(i_j, s_j)$ ve $\tilde{p}_{T_j, u}(i_j, s_j)$ yukarıda tanımlanmıştı.

Çizelge 4.4 kullanıcı sayısı 100'den 500'e ve aşama sayısı K=6 (126 aralık), 7 (254 aralık), 8 (510 aralık) için ortalama hata ve ortalama kare kökü hatalarını göstermektedir. Yukarıdaki sonuçlarda görüldüğü gibi, eğer çok aşamalı kestirimde aralık sayısı kullanıcı sayısından fazla ise kestirim başarılıdır. Algoritma 1'de çok aşamalı kestirim algoritması verilmiştir. k . aşamada baz istasyonu 2^k rastgele erişim aralığı açar ve kullanıcılar bunlara rastgele erişir.

Eğer k . aşamanın sonunda tahmin edilen kullanıcı sayısı o ana kadar ki toplam zaman aralığından az ise algoritma biter.

Algoritma 1 Çok Aşamalı Kestirim Algoritması $[\hat{U}] = \text{cokasamali}(U, \text{metot})$

```

1:  $U$  kullanıcı ( $i = 1, \dots, U$ ).
2: Başlat  $k = 0$ , toplamaralık = 0
3: while dur demedikçe do
4:    $k = k + 1$ 
5:    $T_k = 2^k$ 
6:   Kullanıcılar rastgele  $T_k$  aralığa erişir.  $k$ . aşamada  $i_k$  boş aralık sayısı,  $s_k$  başarılı aralık sayısıdır.
7:   if metot = MO-CA then
8:     (4.6) denklemini kullanarak  $\hat{U} = U_k^{MO-CA}$  bul.
9:   else
10:    (4.7) denklemini kullanarak  $\hat{U} = U_k^{YMO-CA}$  bul.
11:   end if
12:   if  $\hat{U} < \sum_{j=1}^k T_j$  then
13:     dur
14:   else
15:     devam
16:   end if
17: end while

```

Çizelge 4.5 uyarlanır çok aşamalı yaklaşık maksimum olabilirlik kestirimi performans sonuçlarını göstermektedir. Sonuçlar, kestirimin beklenen süresinin kullanıcı sayısı ile kabaca lineer bir şekilde arttığını gösteriyor. Ortalama hata sifıra çok yakın ve ortalama kare kökü kullanıcı sayısının %2-%5'i civarındadır.

4.2 APCO Proje25 Sistemleri İçin Önerilen Çok Aşamalı Yöntem

APCO Proje25 standardı kamu güvenliği telsiz ağları için bir haberleşme yapısı tanımlar. Dijital ses ve veri gönderimini destekler. Sistem dar bantlı olduğundan bit hızı 9600 bps ile sınırlıdır. Fiziksel katmanda her 96 bitlik bloklar kodlanarak 196 bitlik bloklar elde edilir. Serpiştirmeden sonra, belli bir blok sayısı N_b bir araya getirilir ve başlık eklenerek APCO paketi oluşturulur. Bir tip frekans kiplemesi olan C4FM bitleri göndermek için kullanılır. [24]'de APCO sisteminin

Çizelge 4.5: Uyarlanı Çok Aşamalı Yaklaşık Maksimum Olabilirlik Kestirimi: Kestirim Süresi, Ortalama ve Ortalama Kare Kökü

	Beklenen Süre	$E[U - \hat{U}]$	$std[U - \hat{U}]$
U=100	126 [6 aşama]	0.973	5.442
U=150	253.7 [6,7 aşama]	0.823	5.698
U=200	254 [7 aşama]	1.351	8.069
U=250	360 [7,8 aşama]	1.120	8.267
U=300	510 [8 aşama]	0.792	8.055
U=350	510 [8 aşama]	1.152	9.320
U=400	510 [8 aşama]	0.844	11.505
U=450	510 [8 aşama]	0.665	13.344
U=500	662 [8,9 aşama]	2.002	12.176

fiziksel katmandaki benzetimi gerçeklenmiştir. Buradan blok hata olasılıkları (BLHO) elde edilmiştir. Güvenilir bir gönderim için SGO 12 dB'den büyük olmalıdır.

L bit uzunluklu bir mesaj gönderilir. Bu mesaj 96 bit bloklara bölünür ve Fountain kodlanır. Rastgele Doğrusal Fountain Kodlama uygulanmıştır. Alıcının $\lceil \frac{L}{96} \rceil$ blok çözebildiği kabul edilsin. Bir paketdeki N_b bloğun hepsi doğru iletildi ise paket başarılıdır. Verilen N_b blok/paket için L veri biti gönderim süresi;

$$D = \left\lceil \frac{L}{96N_b} \right\rceil \times (48 + 64 + 196 + 196N_b) \times \frac{1}{9600} \quad (4.8)$$

Yukarıdaki gönderim süresi hatasız kanal içindir. Sınırlı bir SGO için bu değerden daha büyük olacağı açıktır.

Önerilen Algoritma 2 kestirim evresi ile başlar (2. satır). Buradaki çoğa gönderim grubundaki kullanıcı sayısı Algoritma 1'deki çok aşamalı rastgele erişim sürecinden tahmin edilmiştir. Sonra $\lceil \frac{L}{96N_b} \rceil$ paket gönderilir (3. satır). $L_i^{r,0}$ kullanıcı i 'nin bu ilk gönderimden sonra biriktirmeye ihtiyaç duyduğu geri kalan bit sayısı olsun.

Bu ilk gönderimden sonra çoğa gönderim yöntemi geribildirim ve gönderim aşamaları ile döngüsel devam eder. Bu aşama kod çözümü yapmamış kullanıcı sayısı sıfır olana kadar devam eder. $[\hat{U}, \hat{L}_{mean}^{r,k}] = \text{feedbackdecoded}(\hat{U}^k, \mathcal{U}^{fu})$

Algoritma 2 : Önerilen Güvenilir Çoğa Gönderim Planı

- 1: L veri biti U kullanıcıya gönderilir ($i = 1, \dots, U$). Her bir gönderilen paket N_b bloktan oluşur.
 - 2: Çalıştır: $[\hat{U}] = \text{cokasamalı}(U, \text{metot})$
 - 3: $\left\lceil \frac{L}{96N_b} \right\rceil$ paketi gönder. $L_i^{r,0}$, i kullanıcısının toplamaya gerek duyduğu bit sayısı olsun.
 - 4: Başla $\text{toplampaket} = \left\lceil \frac{L}{96N_b} \right\rceil$
 - 5: **while** $\hat{U} > 0$ **do**
 - 6: $k = k + 1$, $\hat{U}^k = \hat{U}$
 - 7: Çalıştır: $[\hat{U}, \hat{L}_{mean}^{r,k}] = \text{feedbackundecoded}(\hat{U}^k, \mathcal{U}^{fu})$
 - 8: $\max\left(1, \left\lceil \frac{\hat{L}_{mean}^{r,k}}{L - \hat{L}_{mean}^{r,k}} \times \text{toplampaket} \right\rceil\right)$ paketi gönder.
 - 9: Güncelle $\text{toplampaket} = \text{toplampaket} + \max\left(1, \left\lceil \frac{\hat{L}_{mean}^{r,k}}{L - \hat{L}_{mean}^{r,k}} \times \text{toplampaket} \right\rceil\right)$
 - 10: **end while**
-

fonksiyonu (7. satır) kod çözümü yapmamış kullanıcı sayısını ve onların geri kalan ortalama bit sayısını kestirir. Bu basitçe (4.3) veya (4.5)'in kullanıldığı tek aşamalı kestirimdir. Baz istasyonu \hat{U}^k geribildirim aralıklarını bildirir (\hat{U}^k tahmin edilen kullanıcı sayısı) ve kod çözümü yapmamış kullanıcılar (\mathcal{U}^{fu} kümesi ile belirtilen) bu aralıklardan rastgele seçer. Geribildirim mesajı düğüm ID ve alınması gereken geri kalan bit sayısını içerir. Eğer bir aralıkta birden fazla kullanıcı gönderiyorsa, o geribildirim çözülemez. Boş ve başarılı aralık sayısına dayanarak, baz istasyonu kod çözümü yapmamış kullanıcı sayısını kestirebilir ve \hat{U}^k günceller. Ayrıca başarılı aralıklarda geri kalan bit bilgisine dayanarak, baz istasyonu k . aşamada ortalama geri kalan bit sayısının tahminini $\hat{L}_{mean}^{r,k}$ hesaplar. Sonrasında baz istasyonu gönderilecek paket sayısını hesaplar (9. satır). Baz istasyonu yollamış olduğu paket sayısını ve toplanacak geri kalan bit sayısını bilir. Böylece gönderilecek paket sayısını $\max\left(1, \left\lceil \frac{\hat{L}_{mean}^{r,k}}{L - \hat{L}_{mean}^{r,k}} \times \text{toplampaket} \right\rceil\right)$ ile tahmin eder. Ortalama geri kalan bit sayısı kullanımı maksatlıdır. Eğer en yüksek geri kalan bit sayısı kullanılsaydı, aşırı gönderim sayısına yol açardı. En küçük geri kalan bit sayısı kullanımı ise çok sayıda geribildirim aşamasının oluşmasına yol açar. Bu iki durumda tamamlanma süresini artırır. Ortalama geri kalan bit sayısının kullanımı her aşamada kod çözmemiş kullanıcı sayısını yarılar.

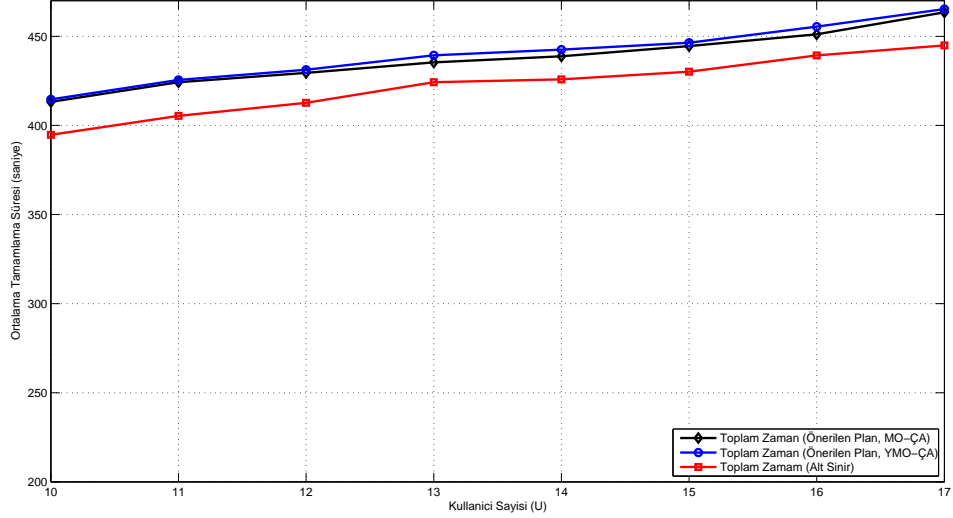
4.3 Nümerik Sonular

SGO deęerleri $\{13, 14, 15, 16, 17, 18, 19, 20\}$ kümesinden rastgele bir deęer alan, N adet oęa gönderim alıcısı durumu deęerlendirilmiřtir. Doppler frekansı 10 Hz olan Rayleigh kanalı düşünölmüřtür. Bölüm 3’de yer alan benzetimlere dayanarak SGO deęerlerine karřılık gelen blok hata olasılıkları 0.0511, 0.0375, 0.0171, 0.0116, 0.0047, 0.0036, 0.0024, 0.0008’dir. Benzetim alıřması 30KB dosya gönderimini dikkate alır. Bir paketteki blok sayısı 40’dır. Bir pakette 3840 veri biti vardır ve bir paket toplamda kodlama ve ek yükü 8100 bittir. Bir paketteki blokların tümü başarılı alınırsa, paket başarılıdır. Her gönderilen paketin başarı/hata baęımsızdır. Her geribildirim paketi 20B uzunluęunda olduęu kabul edilmiřtir. Ayrıca akıřmalar dıřında geribildirim paketlerinin başarılı alındıęı kabul edilmiřtir.

Önerilen oęa gönderim yönteminin performansı test edilmiřtir. İki kestirim metodu düşünölmüřtür, ok aşamalı maksimum olabilirlik kestirimi ve yaklaşık ok aşamalı maksimum olabilirlik kestirimi. Karřılařtırma olarak, tamamlama süresinde alt sınır dikkate alınacaktır. Bu alt sınır 1) baz istasyonu kullanıcı sayısını önceden biliyor, 2) son kullanıcılar mesajı özdüęünde oęa gönderim hemen biter, 3) her kullanıcı akıřma olmadan geribildirim mesajlarını yollar varsayımlarına dayalıdır.

řekil 4.1 30KB’lık bir dosyanın ortalama gönderim süresini gösterir. Geri bildirim mesaj boyutu 20B’dır. Kullanıcı sayısı 10’dan 17’ye deęiřir ve kullanıcıların sinyal güröltü oranı $\{13, 14, \dots, 20\}$ kümesinden bir deęerdir. Maksimum olabilirlik kestirimi zaman alıcı olduęundan, benzetimler düşük sayıda kullanıcı durumları için incelenmiřtir. Sonular, tamamlanma süresinin kullanıcı sayısı ile neredeyse doğrusal olarak arttıęını göstermiřtir. ok aşamalı yaklaşık maksimum olabilirlięin tamamlanma süresi ok aşamalı maksimum olabilirlięinkine ok yakındır. Önerilen oęa gönderim taslaęının tamamlanma süresi ise alt sınırdan sadece %4-5 daha fazladır. Düşük sayıda kullanıcı için geribildirim mesajlarına harcanan süre gözardı edilebilir.

řekil 4.2 30KB’lık bir dosyanın ortalama gönderim süresini gösterir. Geri bildirim mesaj boyutu 20B’dır. Kullanıcı sayısı 100’den 500’e deęiřir ve kullanıcıların

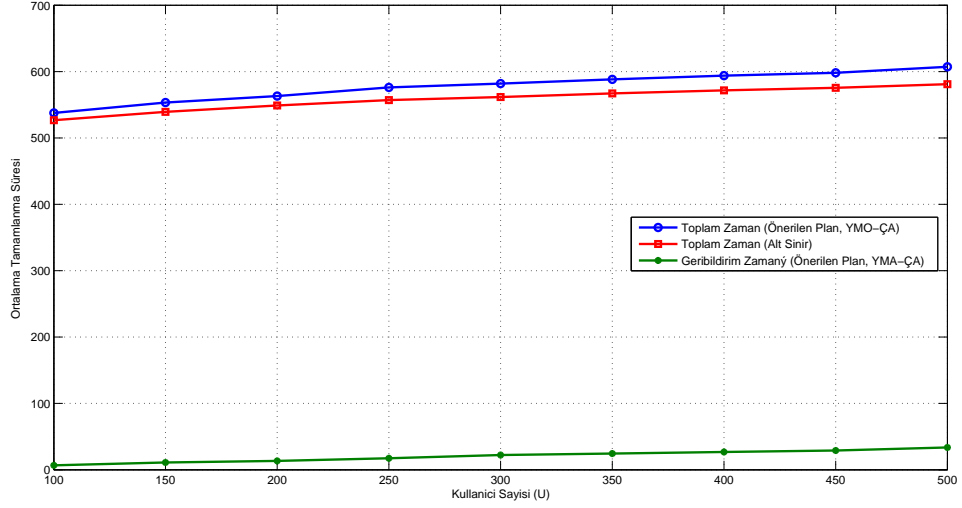


Şekil 4.1: Kullanıcı Sayısı - Ortalama Tamamlama Süresi

sinyal gürültü oranı $\{13, 14, \dots, 20\}$ kümesinden bir değerdir. Sonuçlar, önerilen yaklaşık maksimum olabilirlik kestirim yönteminin alt sınıra çok yakın bir tamamlama süresinde büyük sayıda kullanıcıya güvenilir çoğa gönderimin yapabildiğini göstermektedir. Önerilen yöntemin ortalama tamamlama süresi alt sınırdan sadece %2.5-5 daha fazladır. Alt sınır geribildirim için harcanan zaman açısından fazlaca optimistik olduğunu düşürsek, bu çok iyi bir verimliliğe işaret eder. Alttan üçüncü eğri kestirim ve gönderilen geribildirim paketleri için harcanan ortalama zamanı gösterir. Geribildirim için harcanan zaman tamamlama süresinin küçük bir parçası olduğu görülmektedir.

4.4 Sonuç

Burada, APCO P25 tabanlı telsiz sistemlerinde güvenilir çoğa gönderim incelenmiştir. Baz istasyonununun 1) kullanıcı sayısını tahmin ettiği, 2) Fountain kodlu paket gönderdiği, 3) kod çözümü yapmamış kullanıcılardan onların süreçleri hakkında geribildirim aldığı, 4) geribildirim mesajına dayalı olarak gönderimi sürdürdüğü durum için çoğa gönderim yöntemi sunulmuştur. Kestirim ve geribildirim evreleri dilimli rastgele erişime dayalıdır. Yüksek doğrulukta kullanıcı



Şekil 4.2: Kullanıcı Sayısı - Ortalama Tamamlama Süresi

sayısı kestiren bir çok aşamalı metot önerilmiştir. Geribildirim evresi kod çözümü yapmamış kullanıcı sayısını kestirir ve gönderilecek ilave Fountain kodlu paket sayısını belirler. Önerilen çoğa gönderim yöntemi önceki kullanıcı sayısı bilgisine gerek duymaz. Geribildirim ve gönderim yöntemi herhangi bir kanal durum bilgisine gerek duymaz. Geribildirim paketleri sadece düğüm ID ve toplanacak geri kalan bit sayısını içermek zorundadır. Kestirim yöntemi olarak, maksimum olabilirlik kestirim metodu ve yaklaşımsal metot önerilmiştir. Benzetim sonucunda yaklaşımsal metodun maksimum olabilirlik kestirimine çok yakın olduğu görülmüştür. Nümerik değerlendirmeler önerilen çok aşamalı yöntemin verimliliğini ortaya koyar. Önerilen yöntemin tamamlanma süresi optimistik alt sınıra oldukça yakındır.

5. SONUÇLAR

Bu çalışma konu itibariyle daha önce üzerinde çalışılmamış bir konudur. Literatür analizinde belirtildiği gibi oransız kodlar DVB ve 3G çoğa gönderim uygulamalarında kullanılmak üzere standartlara dahil edilmiş bir tekniktir, ancak APCO P25 gibi telsiz sistemlerinde uygulanması henüz düşünülmemiştir. APCO P25, dar bant, geniş kapsama alanı, yüksek hızlara ulaşabilen gezgin kullanıcılar nedeniyle özgün zorlukları olan bir sistemdir. Bu standarda uygun üretilen telsiz sisteminde ortaya çıkan çok açık bir verimsizliği çözmeyi hedeflemektedir, dolayısıyla sanayi ile ilgili teknik bir problemin (verimsizliğin) giderilmesini sağlayacaktır.

Bu tez çalışmasında APCO P25 protokolüne göre haberleşen telsiz sistemlerinin verimli ve güvenli çoğa gönderim imkanı sağlayan yeni bir uygulama incelenmiştir. Bu çalışmayla telsiz sistemlerinde varolan uzun süren çoğa gönderimi kısa sürede ve daha verimli hale getirilmesi amaçlanmıştır. Yapılan çalışmalar iki aşamalı olarak incelenmiştir.

İlk aşamada, oransız kodlamanın uygulandığı APCO P25 sistemi MATLAB üzerinde gerçekleştirilmiştir. Kurulan sistemin Rayleigh ve TBGG kanallarındaki bit hata oranının önceki çalışmalarla uygun olduğu görülmüştür. Oransız kodlamanın sistemde herhangi bir olumsuz etkiye sahip olmadığı görülmüştür. Aynı sistemde blok hata oranları da incelenip SGO'nun ve kanal model tipine bağlı olarak paketin sahip olması gereken optimal blok sayısının sistem verimliliği üzerindeki etkisi incelenmiştir.

İkinci aşamada ise aynı sistem için çok aşamalı çoğa gönderim incelenmiştir. Zaman dilimli rastgele erişime dayalı olan kestirim ve geri bildirim evreleri önerilmiştir. Önerilen metotta çoğa gönderim başladıktan sonra belli aralıklarla bu evreler uygulanır. Kestirim evresinde baz istasyonunun gönderim yapması gereken kullanıcı kalmayana kadar sürekli döngüsel olarak devam eder. Yapılan

incelemelerde, önerilen yöntemin ortalama tamamlanma süresi oldukça optimist bir yaklaşımla kabul edilen alt sınırdan oldukça yakındır. Sonuçlar, önerilen yaklaşık maksimum olabilirlik kestirim yönteminin alt sınıra yakın bir tamamlama süresinde büyük sayıda kullanıcıya güvenilir çoğa gönderim yapılabildiğini göstermektedir.

KAYNAKLAR

- [1] TIA STANDARD Project25, "FDMA-Common Air Interface," New Technology Standards Project - Digital Radio Technical Standards, 2003.
- [2] Daniel's Electronics, Daniel's P25 Training Guide, 2009.
- [3] P. Cataldi, M. Gerla, F. Zampognaro, "Rateless Codes for File Transfer over DVB-S," First International Conference on Advances in Satellite and Space Communication, 2009.
- [4] J. Byers, M. Luby, M. Mitzenmacher, A. Rege, "A Digital Fountain Approach to Reliable Distribution of Bulk Data," ACM SIGCOMM Computer Communication Review, vol.28, no.4, pp.56-67, 1998.
- [5] M. Luby, "LT Codes," The 43rd Annual IEEE Symposium on Foundations of Computer Science, 2002.
- [6] D.J.C. MacKay, "Fountain Codes," IEEE Proc.-Commun., vol.152, no.6, Dec. 2005.
- [7] A. Shokrollahi, "Raptor Codes," IEEE Transactions on Information Theory, vol.52, pp.2551-2567, 2006.
- [8] "Digital Video Broadcasting (DVB); IP Datacast over DVB-H: Content delivery protocols," Tech. Rep. ETSI TS 102 472 V1.2.1, European Broadcasting Union, 2006.
- [9] "Multimedia Broadcast/Multicast Service (MBMS); protocols and codecs," Tech. Rep. TS 26.346 V7.7.0, 3rd Generation Partnership Project (3GPP), 2008.

- [10] S. W. Kim, S. Y. Kim, S. Kim, J. Heo, "Performance analysis of forward error correcting codes in IPTV," *Consumer Electronics, IEEE Transactions on*, vol.54, no.2, pp.376-380, May 2008.
- [11] D. Gomez-Barquero, D. Gozalvez, N. Cardona, "Application Layer FEC for Mobile TV Delivery in IP Datacast Over DVB-H Systems," *Broadcasting, IEEE Transactions on*, vol.55, no.2, pp.396-406, June 2009.
- [12] J. Lei, M.A. Vazquez-Castro, T. Stockhammer, "Link-Layer FEC and Cross-Layer Architecture for DVB-S2 Transmission With QoS in Railway Scenarios," *Vehicular Technology, IEEE Transactions on*, vol.58, no.8, pp.4265-4276, Oct. 2009.
- [13] C. Hellge, D. Gomez-Barquero, T. Schierl, T. Wiegand, "Layer-Aware Forward Error Correction for Mobile Broadcast of Layered Media," *Multimedia, IEEE Transactions on*, vol.13, no.3, pp.551-562, June 2011.
- [14] M. Luby, T. Gasiba, T. Stockhammer, M. Watson, "Reliable Multimedia Download Delivery in Cellular Broadcast Networks," *Broadcasting, IEEE Transactions on*, vol.53, no.1, pp.235-246, March 2007.
- [15] D. Gomez-Barquero, A. Fernandez-Aguilella, N. Cardona, "Multicast Delivery of File Download Services in Evolved 3G Mobile Networks With HSDPA and MBMS," *Broadcasting, IEEE Transactions on*, vol.55, no.4, pp.742-751, Dec. 2009.
- [16] T. Mladenov, S. Nooshabadi, K. Kim, "Efficient Incremental Raptor Decoding Over BEC for 3GPP MBMS and DVB IP-Datacast Services," *Broadcasting, IEEE Transactions on*, vol.57, no.2, pp.313-318, June 2011.
- [17] H. Chen, R. Maunder, L. Hanzo, "Fountain-Code Aided File Transfer in 802.11 WLANs," *Vehicular Technology Conference, IEEE 70th*, pp.1-5, 20-23 Sept. 2009.
- [18] R. Gallager, "Low-density parity-check codes," *IEEE Transactions on Information Theory*, vol.8, no.1, pp.21-28, 1962.
- [19] TIA Standard Project 25, "FDMA – Common Air Interface," *Telecommunications Industry Association, ANSI/TIA-102*, Sept. 2003.

- [20] H. E. Zorlu, M. Ceven, H. Arslan, "Reception of C4FM and CQPSK Signals in APCO Project-25," Signal Processing and Communications Applications Conference, pp.745-749, April 2009.
- [21] S. E. Bartlett, K. M. Syed, "Modeling Study to Determine the Realistic Constraints of the Wireless Land Mobile Radio Narrowband CAI Interface Specified in the TIA-102 Standard," Wireless Personal Communications, vol.536, pp.149-160, 2000.
- [22] D. Shen, V. O. K. Li, "Performance Analysis for A Stabilized Multi-channel Slotted ALOHA Algorithm," 14th IEEE Proceedings on Personal, Indoor, and Mobile Radio Communications, vol.1, pp.249-253 Sept. 2003.
- [23] C. H. Wei, R. G. Cheng, and S. L. Tsao, "Modeling and Estimation of One-Shot Random Access for Finite-User Multichannel Slotted ALOHA Systems," IEEE Communications Letters, vol.16, no.8, Aug. 2012.
- [24] G. Cigirkan, T. Girici, M. Tas, S. Bengur, "Usage of rateless codes in APCO radio systems," Signal Processing and Communications Applications Conference (SIU), 2014 22nd , pp.1295-1298, 23-25 April 2014.

ÖZGEÇMİŞ

Kişisel Bilgiler

Soyadı, Adı : ÇIĞIRKAN, Gülay
Uyruđu : T.C.
Dođum tarihi ve yeri : 01.05.1987 Çorum
Medeni hali : Bekar
Telefon : +90 506 594 3838
Faks : +90 312 292 4180
e-mail : gulaycigirkan@gmail.com

Eđitim

Derece	Eđitim Birimi	Mezuniyet Tarihi
Y. Lisans	TOBB Ekonomi ve Teknoloji Üniversitesi	2014
Lisans	TOBB Ekonomi ve Teknoloji Üniversitesi	2012

İş Deneyimi

Yıl	Yer	Görev
2012-2014	TOBB Ekonomi ve Teknoloji Üniversitesi	Burslu Y. L. Öğrencisi Proje Asistanı

Yabancı Dil

İngilizce (Çok iyi)
İspanyolca (Orta)

Yayımlar

- G. Cigirkan, T. Girici, M. Tas, S. Bengur, “Usage of rateless codes in APCO radio systems,” IEEE 22nd Signal Processing and Communications Applications Conference, 23-25 April 2014.

- G. Cigirkan, T. Girici, A. M. Yuksel Turgut, “Efficient and Reliable Multicast of Data in APCO P25 Systems,” Deęerlendirilme ařamasında, VTC 2015.