Erc.Ünv.Fen Bil. Derg., 7, 1, (1991), 1147-1163

ARDIŞIL KOD ÇÖZME ALGORİTMASINI VE KOD BİRLEŞTİRME TEKNİĞİNİ KULLANAN I. TÜRDEN KARMA ARQ SİSTEMLERİNİN BAŞARIM ANALİZİ

Necmi TAŞPINAR

Erciyes Üniversitesi Mühendislik Fakültesi Elektronik Böl. KAYSERİ

ÖZET

Karma ARQ sistemleri güvenilir ve etkin bir haberleşme sağlarlar. Bu makalede, devre-dışı (time-out) şartı altında ardışıl kod çözme algoritmasını ve kod birleştirme tekniğini kullanan I. türden karma ARQ sistemlerinin başarım analizi, ikili simetrik kanal için bilgisayar simülasyonu ile yapılmış ve elde edilen sonuçlar yorumlanmıştır. Sonuçta, kod birleştirme tekniğini kullanan I. türden karma seçici-tekrar (SR) ARQ sisteminin en iyi sistem başarımını sağladığı görülmüştür.

THE THROUGHPUT ANALYSIS OF TYPE-I HYBRID ARQ SYSTEMS USING SEQUENTIAL DECODING AND CODE COMBINING TECHNIQUE

SUMMARY

Hybrid ARQ systems allow reliable and efficent communication. In this paper, the throughput analysis of type-I hybrid ARQ schemes using sequential decoding and code combining technique under the time-out condition and binary symmetric channel were done by computer simulation. As a result, it is seen that the type-I hybrid selective-repeat (SR) ARQ scheme using code combining technique yields the best throughput.

1.GİRİŞ

Gürültülü kanallar üzerinden güvenilir ve etkin bir haberleşme sağlamak için kullanılan teknikler iki sınıfta toplanabilirler: Otomatik tekrar isteği (ARQ) ve ileri hata düzeltme sistemleri [1].

Herhangi bir ARQ hata kontrol sisteminde hata sezen bir koda göre oluşturulan kod kelimesi kanal çıkışında iletim ha-

taları içerebilir. Alıcıda kod kelimesinin paritesine bakılarak mesajın doğru olup olmadığına karar verilir. Mesajın doğru olmadığı anlaşıldığında alıcı, bir geri besleme kanalı üzerinden mesajı tekrar göndermesi için vericiyi uyarır. ARQ sistemlerinin dur ve bekle (SW), N kelime geri git (GBN) ve seçici tekrar (SR) modunda olmak üzere üç ayrı çalışma şekli vardır [1-2].

Bir FEC hata kontrol sisteminde ise bir mesaj bloğu hata düzeltici bir kodla kodlanarak alıcıya iletilir. Eğer alınan kod kelimesinde hatalar varsa düzeltilmeye çalışılır; düzeltilemediği takdirde " yeniden gönderme " nin söz konusu olmaması nedeniyle kullanıcıya hatalı veri verilir [1-2].

ARQ ve FEC tekniklerinin uygun bir şekilde birleştirilmesi ile elde edilen " karma ARQ " sistemlerinde FEC alt sistemi kod kelimesindeki hataları düzelterek yeniden gönderme sayısını azaltırken, ARQ alt sistemi yanlış veri kelimelerinin kullanıcıya iletilmesini önler. Karma ARQ sistemleri I. ve II. türden karma ARQ sistemleri olmak üzere iki sınıfta incelenirler [1-2].

I. türden karma ARQ sistemlerinde bir veri paketinin yeniden iletilmesi gerektiğinde, alıcıda bu paket çıkarılır ve yerine yeniden iletilen kopyası yerleştirilir. Böylece kod çözücü herhangi bir zamanda bir paketin yanlızca bir tek kopyasını kullanır ve önceki bütün kopyaları çıkarır. Kanal çok gürültülü olduğunda böyle sistemler arzu edilen sistem başarımını sağlayamazlar. Bir ARQ sisteminin başarımı; birim zamanda doğru alınan bit sayısının birim zamanda iletilen bit sayısına oranı olarak tanımlanır. Son yıllarda Chase, Mullers ve Wolf [3], çok yüksek kanal bit hata oranlarında faydalı bir sistem başarımı sağlamak için, bir paketin bütün alınan kopyalarını birleştirme fikrini ortaya attılar. Gürültülü paketleri birleştirme tekniği olan bu teknik kod birleştirme tekniği olarak adlandırılır [4].

[3]'deki FEC alt sistemi optimum Viterbi kod çözücüsü ile konvolüsyon kodlayıcısından [2] oluşmuştur. Daha sonraki yıllarda Kallel ve Haccoun [5] tarafından yapılan bir çalışmada FEC alt sistemi için ardışıl kod çözme algoritması kullanılmıştır.

2.ARDIŞIL KOD ÇÖZME ALGORİTMASI

Ardışıl kod çözme algoritması, en büyük olasılıklı iletilen yolu belirleme çalışmasında, konvolüsyonel olarak kodlanmiş ağacın [2] sadece bir bölümünün kullanıldığı bir kod çözme işlemidir [5]. Temelde iki ardışıl kod çözme algoritması bulunmaktadır. Bunlar; Fano algoritması [6] ve Zigangirov - Jelinek (Z-J) veya yığın (stack) algoritmasıdır [7]. Ayrık belleksiz bir kanalı (DMC) [8] göz önüne alırsak, kodçözücü herhangi bir zaman biriminde ağacın bir dalını inceler ve

$$m_j = \log_2 \left\{ \frac{P\left[y_j \mid x_j \right]}{P\left[y_j \right]} \right\} - R \qquad (1)$$

şeklinde verilen logaritmik olasılık fonksiyonunu veya Fano sembol metriğini [6] kullanır. Burada X_j , j. kanal giriş sembolünü; y_j , buna karşı düşen alınan sembolü ve R ise kullanılan kodun oranını temsil etmektedir. Bu durumda U sembol uzunluklu bir yol için toplam metrik,

$$M_U = \sum_{j=1}^{U} m_j \qquad (2)$$

şeklindedir ve bir ardışıl kod çözücü sürekli olarak en büyük metrik değerli yolu araştırmaya devam edecektir. Ağacın sonunda, en büyük metrik değerine sahip olan yol kodçözülmüş yol olarak kabul edilecektir. Ardışıl kod çözmenin dezavantajlarından birisi, verilen bir veri paketini çözmek için farklı sayıda hesaplama gerektirmesidir. Bir biti çöz-

mek için gerekli hesaplama sayısının dağılımı Pareto fonksiyonu [8] vasıtasıyla sınırlandırılır:

$$P(C \geq N) \leq \beta . N^{-\alpha} \qquad , N >> 1 \qquad (3)$$

Burada β, N'den bağımsız sınırlı bir sabit ve α,

$$R = \frac{E_0(\alpha)}{\alpha} \qquad (4)$$

parametrik denklemi ile verilen Pareto üssüdür. (4) denkleminde $E_0(\alpha)$, sadece kanala bağlı olan Gallager fonksiyonudur. DMC için $E_0(\alpha)$,

$$E_0(\alpha) = -\log_2 \sum_j \left[\sum_k P(k) . \left\{ P\left(j \mid k\right) \right\}^{1/1+\alpha} \right]^{1+\alpha} \qquad (5)$$

şeklinde olup, burada j ve k kanal çıkış ve giriş bilgileridir [8]. P(k) ise giriş bilgisi k'nın olasılık fonksiyonudur. Pareto üssü $\alpha \leq 1$ olduğu zaman, bir biti çözmek için gerekli ortalama hesaplama sayısı teorik olarak sınırsızdır. $\alpha = 1$ durumunda $E_0(\alpha)$, ardışıl kod çözmenin hesaplamayı kesme oranı (R_{comp}) olarak adlandırılır. Böylece $R = R_{comp}$ 'da çalışma, ardışıl kod çözücüler için pratik limiti temsil eder. Eğer α bu limitten daha büyük değerler alırsa, bir biti çözmek için gerekli hesaplama sayısı düşer, bu ise bir veri paketinin kod çözme süresini azaltır.

Kanal gürültüsü nedeniyle bazen kod çözme işlemi başarısız olabilir. Bu durumda bir geri besleme kanalı üzerinden yeniden iletim yapılır. Yeniden iletim işleminin birkaç türü vardır [9]. Bunlardan birinde kod çözücü, verilen bir paketin çözülmesi için gereken süreyi bir T_{max} değerine sınırlandırabilir. T_{max} değerinin aşılması halinde kod çözme işlemi durdurulur, son paket çıkarılır ve onun yeniden iletimi istenir [9]. Bu yeniden iletim işlemi devre-dışı (time

out) yeniden iletim işlemi olarak da adlandırılır. Kod çözme süresinin hesaplama sayısıyla orantılı olduğunu kabul edersek, bu durumda, verilen bir paketin çözülmesi için yapılan toplam hesaplama sayısı C_L, C_{max} değerini aşarsa kod çözme işlemi durdurulur ve paketin yeniden iletimi istenir. Bu durumda kod çözme başarısızlığı,

$$P(F) = P(C_L \geq C_{max}) \qquad (6)$$

şeklinde verilen P(F) olasılığı ile meydana gelir. L bitlik bir bilgi paketi için P(F) olasılığı (3) ifadesinden,

$$P(F) \simeq L.\beta.C_{max}^{-\alpha} \qquad (7)$$

şeklinde yaklaşık olarak elde edilebilir [9].

3.ARQ SİSTEMLERİNDE KOD BİRLEŞTİRME TEKNİĞİ

Kod birleştirme tekniği; daha düşük bir kodlama oranı ve böylece çok daha güçlü bir hata düzelten kod elde etmek için, bir R oranlı kodla kodlanmış herhangi bir sayıdaki paketleri birleştirmek için kullanılan bir tekniktir [4].

Kod birleştirme tekniğini kullanan ARQ sistemlerinde bir paketin yeniden iletimi istendiğinde, bu paket genel ARQ'da olduğu gibi çıkarılmaz. Kod çözücü yeniden iletilen kopyayı aldıktan sonra, iki kopyayı birbiriyle sıralı karıştırır ve bu birleştirilmiş kopyayı R/2 (bit/sembollük) kod oranına sahip iki tekrarlı bir koddan türetilmiş bir paket şeklinde ele alıp tekrar kod çözme işlemine tabi tutar. İkinci defa bir yeniden iletim gerekirse, bu durumda yeni kopyanın iletimi gerçekleştirilir. Kod çözücü bu kopyayı önceki iki kopya ile birleştirir ve birleştirilmiş paketi R/3 (bit/sembollük) kod oranına sahip üç tekrarlı bir koddan türetilmiş bir paket şeklinde ele alıp kod çözme işlemine tabi tutar [5].

4.KOD BİRLEŞTİRMELİ ARDIŞIL KOD ÇÖZME ALGORİTMASI

Bu kısımda, bir kod kelimesinin n tane kopyasının kod birleştirme tekniği vasıtasıyla birleştirilmesi sonucu elde edilen paketin ardışıl kod çözme algoritması vasıtasıyla çözülmesi ele alınacaktır. $I = (I_1, I_2, \ldots\ldots, I_L)$ enformasyon dizisi, $x = (x_1, x_2, \ldots\ldots, x_U)$ dizisine kodlanmış ve x kod dizisi n defa tekrar edilerek ardışıl şekilde bir ayrık belleksiz kanal (DMC) üzerinden iletilmiş olsun (DMC her tekrar için farklı olabilir. Bu durumda her tekrarda bir geçiş olasılık kümesi göz önüne alınmalıdır). Alınan n tane dizi,

$$y^{(i)} = \left(y_1^{(i)}, y_2^{(i)}, \ldots\ldots, y_U^{(i)} \right), \qquad i=1,2,\ldots,n \qquad (8)$$

şeklinde olsun. $i = 1,2,\ldots\ldots,n$ için kod çözücü $y^{(i)}$ dizilerini sıralı karıştırır ve

$$Y = \left(y_1^{(1)}, y_1^{(2)}, \ldots\ldots, y_1^{(n)}, \ldots\ldots, y_j^{(1)}, y_j^{(2)}, \ldots\ldots, y_j^{(n)}, \ldots \right.$$
$$\left. \ldots\ldots\ldots, y_U^{(1)}, y_U^{(2)}, \ldots\ldots, y_U^{(n)} \right) \qquad (9)$$

dizisini oluşturur. $\left(y_j^{(1)}, y_j^{(2)}, \ldots\ldots, y_j^{(n)} \right)$ 'den oluşan her grup x_j 'nin n defa iletimine karşı düşer. Bu durumda bu n sembol istatistiksel olarak birbirlerine bağımlıdır ve böylece ardışıl kod çözücü onları tek bir bilgiden ibaret bir seri şeklinden ziyade, herhangi bir zamandaki n bilgiden oluşmuş bir grup olarak göz önüne almalıdır [5].

Kodlayıcı ve n defa tekrar edici tek bir cihaz şeklinde modellenebilir. x paketinin n defa tekrarlanması yerine x'in her sembolü n defa tekrarlanmış olsun. Böylece $x^{(n)}$ ile gösterilen sonuçtaki dizi,

$$x^{(n)} = \Big(\underbrace{x_1, x_1, \ldots, x_1}_{(\text{n tekrar})}, \ldots\ldots\ldots, \underbrace{x_j, x_j, \ldots, x_j}_{(\text{n tekrar})}, \ldots\ldots\ldots, \underbrace{x_0, x_0, \ldots, x_0}_{(\text{n tekrar})} \Big) \qquad (10)$$

seklinde elde edilir [5]. Eğer $x^{(n)}$ dizisi bir DMC üzerinden iletilirse (9) ifadesindeki aynı Y dizisi elde edilecektir ki, burada her bir n tekrarlı $\left(x_j, x_j, \ldots, x_j \right)$ sembolü $\left(y_j^{(1)}, y_j^{(2)}, \ldots, y_j^{(n)} \right)$ şeklindeki n bilgiye karşı düşmektedir.

n ardışıl tekrar için kullanılan kanal, her bir n tekrarlı $\left(x_j, x_j, \ldots, x_j \right)$ giriş sembolü için $\left(y_j^{(1)}, y_j^{(2)}, \ldots, y_j^{(n)} \right)$ çıkış bilgisi ürettiğinden dolayı bir " zahiri kanal " olarak düşünülebilir. Bu zahiri kanalın bir DMC olduğu açıktır.

Kod birleştirmeli ardışıl kod çözme algoritması için Fano metriği,

$$m_j^{(n)} = \log_2 \left\{ \frac{\prod_{i=1}^{n} P\left(y_j^{(i)} \,\middle|\, x_j \right)}{P\left(y_j^{(1)}, y_j^{(2)}, \ldots, y_j^{(n)} \right)} \right\} - R \qquad (11)$$

seklinde elde edilir [5]. Örnek olarak Şekil-1(a)'da görülen p geçiş olasılıklı ikili simetrik kanal (BSC) için, karşı düşen n=2 tekrarlı zahiri kanal Şekil-1(b)'de gösterilmiştir. BSC için, R kodlama oranlı kodun ardışıl kod çözülmesinde kullanılan iki sembol metriği,

$$m_1^{(1)} = \log_2 (p/0.5) - R = \log_2 \left[2.p \right] - R$$

$$m_2^{(1)} = \log_2 \left[(1-p)/0.5 \right] - R = \log_2 \left[2.(1-p) \right] - R \qquad (12)$$

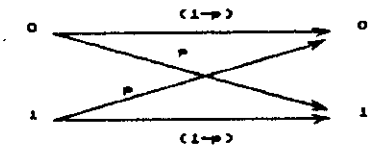
şeklinde elde edilir [8]. Şekil-1(b)'de görülen zahiri kanala göre, kod birleştirmeli ardışıl kod çözme algoritması kullanıldığında üç tane sembol metriği elde edilir. Bunlar:

$$m_1^{(2)} = \log_2 \left[2.p^2 / \left\{ (1-p)^2 + p^2 \right\} \right] - R$$

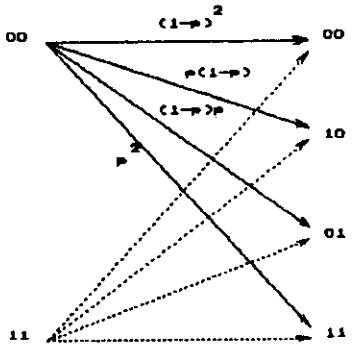
$$m_2^{(2)} = -R \qquad (13)$$

$$m_3^{(2)} = \log_2 \left[2.(1-p)^2 / \left\{ (1-p)^2 + p^2 \right\} \right] - R$$

şeklinde elde edilir [5].



Şekil-1: (a) İkili simetrik kanal.



(b) İki tekrarlı zahiri kanal.

5.ARDIŞIL KOD ÇÖZME ALGORİTMASINI VE KOD BİRLEŞTİRME TEKNİĞİNİ KULLANAN 1. TÜRDEN KARMA SR ARQ SİSTEMİ

Devre-dışı (time-out) şartı altında ardışıl kod çözme algoritmasını ve kod birleştirme tekniğini kullanan I. türden karma SR ARQ sisteminde k bitlik I(x) bilgi dizisi bir (n,k,m) C_1 konvolüsyon koduyla bir V(x) kod kelimesine kodlanır ve kanal üzerinden gönderilir. Alınan dizi V'(x), devre-dışı şartı altında çalışan bir ardışıl kod çözücü vasıtasıyla çözülür. Kod çözme işlemi başarılı ise çözülen dizi alıcı tarafından kabul edilir. Eğer kod çözme işlemi başarılı değilse kod kelimesinin yeniden iletimi istenir ve hatalı kod kelimesi alıcıda saklanır. Yeniden iletilen kod kelimesi alındıktan sonra kod çözücü bu iki kopyayı kod birleştirme tekniğini kullanarak birleştirir ve kod çözme işleminden geçirir. Kod çözme işlemi başarılı ise çözülen dizi alıcı tarafından kabul edilir. Kod çözme işlemi başarısız ise kod kelimesinin iki kopyası alıcıda saklanır ve kod kelimesinin yeniden iletimi istenir. Yeniden iletilen kopya alındıktan sonra alıcıda saklanan iki kopya ile birleştirilir ve tekrar kod çözme işlemine tabi tutulur. Yeniden iletim ve kod birleştirme işlemi kod çözme başarılı oluncaya kadar devam eder [5].

Verilen bir paketin n birleştirilmiş kopyasının çözülmesi işleminde $F^{(n)}$ ve $S^{(n)}$, sırasıyla " başarısız kod çözme " ve " başarılı kod çözme " olaylarını ifade etsinler. Bu durumda,

$$P\left(F^{(n)} \right) = 1 - P\left(S^{(n)} \right) = P\left(C_L^{(n)} \geq C_{max} \right) \qquad (14)$$

yazılabilir. Burada $C_L^{(n)}$, n kopyanın birleştirilerek ardışıl kod çözme algoritması vasıtasıyla çözülmesi işlemindeki toplam hesaplama sayısıdır. (7) ifadesinden faydalanılarak kod çözmenin başarısız olma olasılığı $P\left(F^{(n)} \right)$ yaklaşık olarak,

$$P\left(F^{(n)} \right) = L.\beta_n.C_{max}^{-\alpha_n} \qquad (15)$$

şeklinde yazılabilir. Burada β_n , N'den bağımsız sınırlı bir sabittir. α_n ise n tekrarlı zahiri kanala ilişkin zahiri Pareto üssüdür ve

$$R = \frac{E_0^{(n)}(\alpha_n)}{\alpha_n} \qquad (16)$$

parametrik denklemi ile verilir ki, burada R, bit/sembol olarak orijinal kodun oranı ve $E_0^{(n)}$, zahiri kanalın Gallager fonksiyonu olup,

$$E_0^{(n)}(\alpha_n) = -\log_2 \sum_{j_1,j_2} \sum \cdots \sum_{j_n} \left[\sum_k P(k).\left\{ \prod_{i=1}^{n} P\left(j \mid k\right) \right\}^{1/1+\alpha_n} \right]^{1+\alpha_n} \qquad (17)$$

ile verilir. (17) ifadesindeki k ve j harfleri kanal giriş ve çıkış sembollerini, n ise her sembolün tekrarlanma sayısını temsil etmektedir.

Verilen bir paketin ortalama iletim sayısı:

$$E\,[V] = 1.P\left(S^{(1)} \right) + 2.P\left(F^{(1)},S^{(2)} \right) + 3.P\left(F^{(1)},F^{(2)},S^{(3)} \right) + \ldots\ldots + n.P\left(F^{(1)},F^{(2)},\ldots\ldots,F^{(n-1)},S^{(n)} \right) + \ldots\ldots \qquad (18)$$

şeklinde ifade edilir. (18) ifadesini,

$$E\,[V] = 1.P\left(S^{(1)} \right) + 2.P\left(F^{(1)},S^{(2)} \right) + 2.P\left(F^{(1)},F^{(2)} \right) - 2.P\left(F^{(1)},F^{(2)} \right) + 3.P\left(F^{(1)},F^{(2)},S^{(3)} \right) + 3.P\left(F^{(1)}, \right.$$

$$F^{(2)},F^{(3)}\Big) - 3.P\Big(F^{(1)},F^{(2)},F^{(3)}\Big) + \ldots\ldots\ldots\ldots + n.P\Big(F^{(1)},F^{(2)},\ldots\ldots,F^{(n-1)},S^{(n)}\Big) + n.P\Big(F^{(1)},F^{(2)},\ldots\ldots,F^{(n-1)},F^{(n)}\Big) - n.P\Big(F^{(1)},F^{(2)},\ldots\ldots\ldots,F^{(n-1)},F^{(n)}\Big) + \ldots\ldots\ldots\ldots \quad (19)$$

şeklinde ifade edebiliriz. (19) ifadesindeki terimleri yeniden düzenlersek,

$$E[V] = 1 + P\Big(F^{(1)}\Big) + P\Big(F^{(1)},F^{(2)}\Big) + P\Big(F^{(1)},F^{(2)},F^{(3)}\Big) + \ldots\ldots\ldots + P\Big(F^{(1)},F^{(2)},\ldots\ldots\ldots,F^{(n-1)},F^{(n)}\Big) + \ldots\ldots\ldots \quad (20)$$

elde edilir.

(j-1). adımda başarısız kod çözmenin meydana gelme olasılığı verildiğinde j. adımda da başarısız bir kod çözmenin meydana gelmesi olasılığı, bir paketin j tane kopyasının birleştirilmesi sonucu elde edilen paketin çözülmesi işleminin başarısız olma olasılığından büyüktür; yani,

$$P\Big(F^{(j)} \,\Big|\, F^{(j-1)}\Big) \geq P\Big(F^{(j)}\Big) \quad (21)$$

dir. Şimdi,

$$P\Big(F^{(1)},F^{(2)},\ldots\ldots\ldots,F^{(j)}\Big) \leq P\Big(F^{(j)}\Big) \quad (22)$$

gerçeğini kullanarak E [V] için alt ve üst sınırları,

$$1 + \sum_{n=1}^{\infty} \prod_{i=1}^{n} P\Big(F^{(i)}\Big) \leq E[V] \leq 1 + \sum_{n=1}^{\infty} P\Big(F^{(i)}\Big) \quad (23)$$

şeklinde yazabiliriz [5]. Ardışıl kod çözme algoritmasını ve kod birleştirme tekniğini kullanan I. türden karma SR ARQ sisteminin başarımı,

$$\eta_{SR} = \frac{1}{E\ [V]} \cdot \frac{k}{k + m - 1} \cdot R \qquad (24)$$

ifadesinden elde edilir. E_s / N_0, sinyal gücünün gürültü gücüne oranını göstermek üzere, E_s / N_0'ın büyük değerleri için veya eşdeğeri olan $\alpha > 1$ için (23) ifadesindeki her iki toplamada hakim terim $1 + P\left(F^{(1)} \right)$ olup, diğer terimler ihmal edilebilir; böylece her iki toplama aynı değerleri verir. (20) ifadesindeki her $P\left(F^{(1)}, F^{(2)}, \ldots\ldots, F^{(j)} \right)$ terimi,

$$P\left(F^{(1)}, F^{(2)}, \ldots\ldots, F^{(j)} \right) = P\left(F^{(1)} \mid F^{(2)}, \ldots\ldots, F^{(j)} \right) \cdot P\left(F^{(2)} \mid F^{(3)}, \ldots\ldots, F^{(j)} \right) \cdot \ldots\ldots P\left(F^{(j-1)} \mid F^{(j)} \right) \qquad (25)$$

şeklinde yazılabilir. $P\left(F^{(j-1)} \mid F^{(j)} \right)$ terimi, (j-1) kopyanın ilave bir kopya ile birleştirilmesi durumu için, başarısız bir kod çözmenin meydana gelme olasılığı verildiğinde, bir paketin (j-1) kopyasının birleştirilmesi halinde başarısız bir kod çözmenin meydana gelme olasılığını tanımlar. E_s / N_0'ın küçük değerleri için, yani $\alpha_j \leq 1$ için, bir paketin j kopyasının birleştirilerek çözülmesi büyük bir olasılıkla başarısız olacaktır. Tersine, eğer $\alpha_j > 1$ ise, bir paketin j kopyasının birleştirilerek çözülmesi büyük bir olasılıkla başarılı olacaktır. Bu durumda $\alpha_{j-1} < \alpha_j$ olduğu kabul edilir. Bu nedenle, bir paketin j kopyasının birleştirilmesi durumunda başarısız kod çözmenin meydana gelme olasılığı verildiğine göre, bu j kopyanın sadece (j-1) tanesinin birbiriyle birleştirildiği durumda başarısız bir kod çözmenin meydana gelme olasılığı yüksektir; yani,

$$P\left(F^{(j-1)} \mid F^{(j)} \right) \simeq 1 \qquad (26)$$

dir. Üstelik, $\alpha_{j-1} \leq \alpha_j \leq 1$ olduğundan büyük bir olasılıkla

$$P\left(F^{(j-1)} \mid F^{(j)} \right) \simeq P\left(F^{(j)} \right) \simeq 1 \qquad (27)$$

olduğu teyit edilebilir. Yukarıdaki tartışma (25) ifadesindeki bütün koşullu olasılıklara genelleştirilebilir. Bu nedenle (25) ifadesi,

$$P\left(F^{(1)}, F^{(2)}, \ldots\ldots, F^{(j)} \right) \simeq \prod_{i=1}^{j} P\left(F^{(1)} \right) \simeq P\left(F^{(j)} \right) \qquad (28)$$

şeklinde yazılabilir ve böylece (28) ifadesi kullanılarak (23) ifadesindeki her iki toplamanın hemen hemen aynı değerleri verdiği görülebilir.

Kod birleştirmenin olmadığı durumlarda verilen bir paketin başarılı şekilde çözülmesi istatistiksel olarak bağımsızdır. Bu durumda ortalama iletim sayısı $E[V]^*$, $P\left(F^{(1)} \right)$, nin $P\left(F^{(1)} \right)$ ile yerdeğiştirdiği (23) ifadesinin sol tarafındaki toplama ile verilir. Buna göre,

$$E[V]^* = \frac{j}{1 - P\left(F^{(1)} \right)} \qquad (29)$$

elde edilir [5].

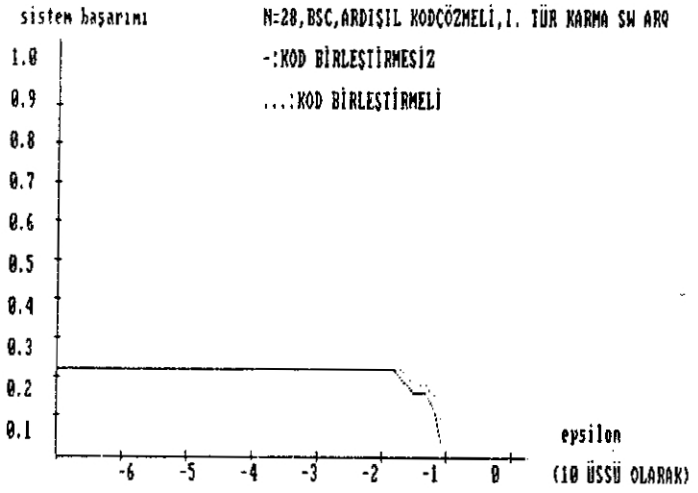
6.SİMÜLASYON ÇALIŞMALARI

Ardışıl kod çözme algoritmasını ve kod birleştirme tekniğini kullanan I. türden karma ARQ sistemlerinin dur ve bekle (SW), N kelime geri git (GBN) ve seçici tekrar (SR) mo-

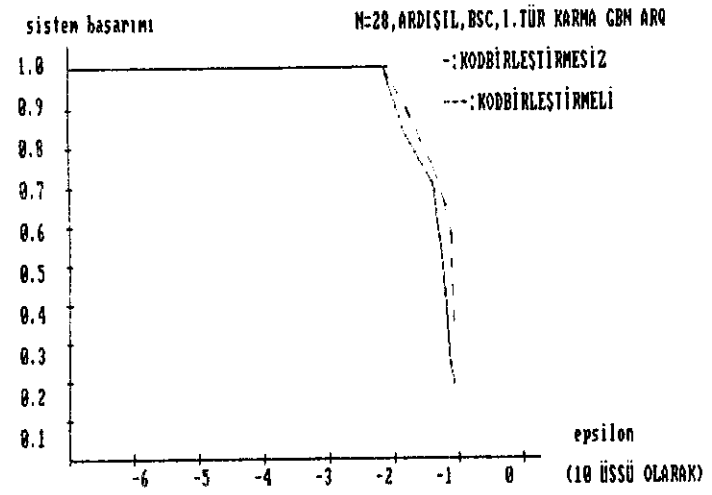
dundaki çalışmalarının bilgisayar simülasyonu ikili simetrik kanal (BSC) için devre-dışı (time-out) şartı altında yığın (stack) algoritması kullanılarak yapılmıştır. Söz konusu sistemlerde veri gönderme hızı 1 Mb/s, bekleme süresi 1.68 ms ve kod uzunluğu n=28 bit olarak alınmıştır. Simülasyon programlarında kısıt uzunluğu 2, oktal generatörü 57 olan sistematik olmayan konvolüsyon kodu kullanılmıştır [2]. Her üç sistemin başarımı Şekil-2, Şekil-3 ve Şekil-4' de verilmiştir.

7.SONUÇ

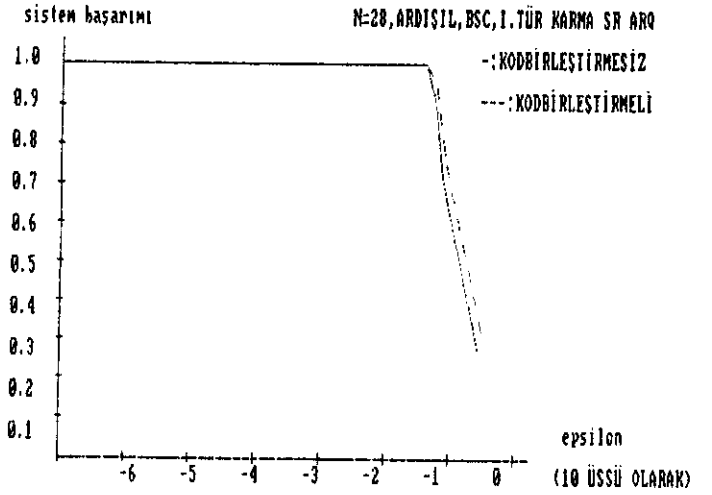
Şekil-2'de verilen I. türden dur ve bekle (SW) ARQ sisteminin başarım grafiğine göre sistemin başarımının oldukça düşük olduğu, fakat kod birleştirme tekniğinin kullanılması halinde başarımın biraz daha iyileştiği görülmektedir. I.



Şekil-2. Ardışıl kod çözme algoritmasını ve kod birleştirme tekniğini kullanan I. türden karma SW ARQ sisteminin başarım grafiği.



Şekil-3. Ardışıl kod çözme algoritmasını ve kod birleştirme tekniğini kullanan I. türden karma GBN ARQ sisteminin başarım grafiği.



Şekil-4. Ardışıl kod çözme algoritmasını ve kod birleştirme tekniğini kullanan I. türden karma SR ARQ sisteminin başarım grafiği.

türden N kelime geri git (GBN) ARQ sisteminin başarımı ise Şekil-3'de verilmiştir. Bu sistemin I. türden dur ve bekle (SW) ARQ sistemine göre çok daha iyi başarım sağladığı; kod birleştirme tekniğinin başarımı daha çok arttırdığı görülmektedir. Şekil-4'de verilen I. türden seçici tekrar (SR) ARQ sisteminin ise diğer türden sistemlere göre daha iyi başarım sağladığı; kod birleştirme tekniğinin kullanılması halinde en iyi sistem başarımının sağlandığı gözlenmiştir.

REFERANSLAR

[1] S. LIN, D.J. COSTELLO,JR, M.J. MILLER, Automatic repeat request error control schemes, IEEE Commmunications Magazine, pp. 5-15, Dec. 1984.

[2] S. LIN, D.J. COSTELLO, JR, Error control coding: Fundamentals and applications , New Jersey: Prentice-Hall, 1983.

[3] D. CHASE, P.D. MULLERS, J.K. WOLF, Application of code combining to a selective-repeat ARQ link, Proc. MILCOM' 85, Conf. Rec. vol.1, 1985, pp.247-252.

[4] D. CHASE, Code combining: A maximum-likelihood decoding approach for combining an arbitrary number of noisy packets, IEEE Trans. Commun., vol. COM-33, pp. 385-393, May 1985.

[5] S. KALLEL, D. HACCOUN, Sequential decoding with ARQ and code combining: A robust hybrid FEC / ARQ system, IEEE Trans. Commun., vol. COM-36, pp. 773-780, July 1988.

[6] R.M. FANO, A heuristic discussion of probabilistic decoding, IEEE Trans. Inform. Theory, vol. IT-9, pp. 64-73, 1963.

[7] F. JELINEK, Fast sequential decoding using a stack, IBM J. Res. Develop., vol. 13, pp. 675-685, Nov.1969.

[8] A.J. VITERBI, J.K. OMURA, Principles of digital communication and coding, New-York: Mc Graw-Hill, 1979.

[9] A. DRUKAREV, D.J. COSTELLO, JR, Hybrid ARQ error control using sequential decoding, IEEE Trans. Inform. Theory, vol. IT-29, pp. 521-535, July 1983.