

Kutupsal Kodlar ile Asimetrik ve Asimetrik Olmayan Slepian–Wolf Kodlama Asymmetric and Nonasymmetric Slepian–Wolf Coding with Polar Codes

Saygun Öney

Elektrik ve Elektronik Mühendisliği Bölümü
Bilkent Üniversitesi, Ankara
saygun@bilkent.edu.tr

ÖZETÇE

Bu bildiri, düzgün marjinal dağılıma sahip kaynaklar için Slepian–Wolf (SW) hız eğrisindeki herhangi bir noktaya ulaşabilen Gehrig tarafından bulunmuş bir asimetrik olmayan dağıtık kaynak kodlama (DKK) tasarımı ele alıyor ve kutupsal kodlamanın bu yöntem içerisinde nasıl kullanılabileceğini gösteriyoruz. Daha sonra önerilen yöntemin simülasyon sonuçlarıyla performansını sunuyoruz.

ABSTRACT

We consider a nonasymmetric distributed source coding (DSC) scheme invented by Gehrig which can achieve any point on the dominant face of the Slepian–Wolf (SW) rate region for sources with *uniform* marginals. And we show how to use polar codes in this scheme. We then present simulation results that exhibit the performance of the proposed method.

1. GİRİŞ

Slepian-Wolf (SW) kuramının [1] ispatı rassal kodlama savına dayanmaktadır ve yapıcı değildir. Wyner 1974'te [2], ikili doğrusal kanal kodları kullanarak asimetrik SW kodlaması inşa edilebileceğini ortaya koymuştur ve bu kurgunun optimalliğini göstermiştir. Eğer ki bir doğrusal blok kod kaynakların arasındaki ilişkiyi modelleyen ikili simetrik kanalın (BSC) kapasitesine erişebiliyorsa, o zaman aynı kod bu ilişki modeli için SW sınırına erişebilen bir koda dönüştürülebilir. Bu yöntem o zamandan beri (Wyner'ın) *sendrom* yaklaşımı olarak adlandırılmaktadır. SW problemine kanal kodları kullanarak pratik bir çözüm sunan ilk çalışma 1999'da Pradhan ve Ramchandran [3] tarafından yayımlanmıştır ve sendrom yaklaşımını benimsemektedir. Bu dönüm noktası olan makalenin yayımlanmasından bu yana, konu üzerinde oldukça fazla miktarda araştırma yapılmıştır. Çeşitli araştırmacılar tarafından hem asimetrik hem de asimetrik olmayan SW problemi için, LDPC ve turbo kodlar gibi en son teknoloji kodları kullanan ve sendrom yaklaşımını benimseyen birçok yöntem bulunmuştur.

Yakın zamanda Arıkan tarafından keşfedilmiş olan kutupsal kodlama [4], düşük kodlama ve kod çözme karmaşıklığına sahip olduğu halde bazı kanallar için kapasiteye ulaşabildiği ispatlanabilen ilk kodlama yöntemidir. Keşfinden kısa bir süre

sonra, kaynak kodlama ve asimetrik Slepian–Wolf ve Wyner–Ziv problemleri için kutupsal kodlamanın optimal olduğunu gösteren bir çok çalışma yapılmıştır [5, 6]. Kutupsal kodların asimetrik SW probleminde nasıl kullanılabilecekleri bu çalışmalarda gösterilmiş olsa da asimetrik olmayan durum için bir yöntemin tanımlanması açık bir problemdir.

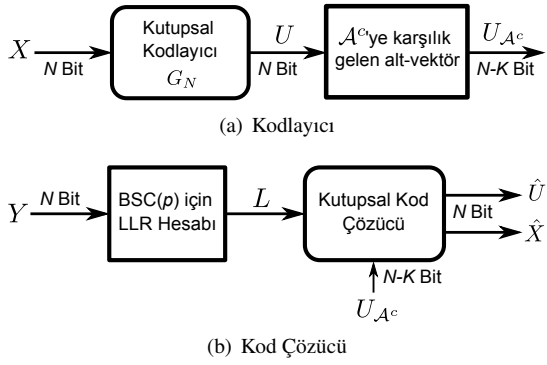
Kutupsal kodların ortaya konduğu makalede [4] önerilen kod çözme yöntemi “sıralı eleme” (SE) yöntemidir. Kutupsal kodların bu kod çözücüyle performansları kısa blok uzunlukları için hiç de tatmin edici değildir. Yakın zamanda, SE kod çözme için verimli bir liste kod çözme yöntemi [7]'de sunulmuştur. Ancak, liste boyutuyla birlikte performans artıyor olsa da bit hata oranı (BHO) eğrisi belirgin bir “şelale” davranışı göstermektedir. Bu durumun temel sebebi, kutupsal kodların inşasının özünde olan düşük asgari kod uzaklığıdır. Bu duruma çözüm getirmek için basit kod zincirleme yöntemleri kullanılabilir. Bu bağlamda, [7]'nin yazarları tarafından, bilgi bitlerine 16 bitlik bir CRC eklenmesi ve SE liste (SEL) kod çözme işleminin bu CRC yardımıyla yapılması önerilmiştir. Kısa ve orta blok uzunlukları için bu yöntem kod performansını, neredeyse hiç ek yük getirmeden, oldukça artırmaktadır.

Bu bildiriye ilk önce, kutupsal kodlayıcı / SE kod çözücü ikilisinin pratik asimetrik SW ve tek kaynak sıkıştırma problemlerinde nasıl kullanıldığını kısaca gözden geçiriyoruz. Daha sonra düzgün marjinal dağılıma sahip kaynaklar için asimetrik olmayan SW kodlama probleminin çözümü için Gehrig'in yönteminin [8] kutupsal kodlara nasıl uygulanabileceğini gösteriyoruz. En son olarak bu yöntemlerin performansını ortaya koymak için simülasyon sonuçları sunuyoruz. En iyi performanslara ulaşabilmek için CRC'li SEL kod çözücü kullanıyoruz. Okuyucunun kutupsal kodlara, özellikle [4, 6]'deki konulara ve SW problemine aşina olduğunu kabul etmekte ve yer darlığı nedeniyle temel konularda fazla detaya girmemekteyiz.

2. ASİMETRİK SLEPIAN–WOLF ve TEK KAYNAK SIKIŞTIRMA

Bu bildiriye, ilintili X ve Y kaynaklarının marjinal dağılımlarının düzgün olduğu kabul edilmektedir. Bu durumda, $H(X|Y) = H(Y|X) = \mathcal{H}(p)$ 'dir ve $\mathcal{H}(\cdot)$ ikili entropi fonksiyonudur. Bu durum aslında kanal kodlama probleminden çok da farklı değildir. x^N 'in bir $BSC(p)$ 'den geçirilmiş hali olan y^N gözleminin verildiği durumda x^N 'i tahmin etmek

Bu araştırma TÜBİTAK tarafından 110E243 nolu proje kapsamında desteklenmiştir.



Şekil 1: Kutupsal kodların asimetrik SW için kullanımı.

bir kanal kod çözme probleminden başka birşey değildir. Normal bir kanal kod çözme işleminden tek farkı, çözümlemenin yapılacağı eşkümenin (coset) sıfır sendromuna karşılık gelmemesi, onun yerine x^N 'in sendromu ile belirlenmesidir.

[6]'de ortaya konan kutuplaşmanın kaynak kodlamaya uygulanması aşağıdaki gibi özetlenebilir. x^N girdi vektörüne uygulanan G_N kutupsal dönüşüm matrisi, çıktı vektörü u^N 'nin koşullu entropi terimlerinin kutuplaşmasını sağlar. Yani blok uzunluğu N büyüdükçe $H(U_i|U^{i-1}, Y^N)$ terimleri 0 veya 1'e yakınsarlar. [6]'te sunulduğu gibi verilen bir N ve oran R için, koşullu entropi terimlerinin 1'e daha yakın olduğu ($N - K$) uzunluktaki yüksek entropi kümesi $E_{X|Y}$ oluşturulabilmektedir. Bu durumda, kodlama işlemi, $u^N = x^N G_N$ dönüşümünün yapılması ve indisleri yüksek entropi setine dahil olan bit vektörü $u_{E_{X|Y}}$ 'nin sıkıştırılmış kelime olarak çıktıya verilmesinden ibarettir. Koşullu entropi terimleri, Bhattacharyya parametreleri $Z(U_i|U^{i-1}, Y^N)$ ile aynı anda kutuplaşmaktadırlar. Yani yüksek entropi kümesi aslında kanal kodlama açısından bakıldığında dondurulmuş indis kümesi \mathcal{F} (A^c) ile aynıdır. Aslında kanal kodlama açısından sıkıştırma, x^N girdi vektörünün sendromunun hesaplanmasına karşılık gelir. Dondurulmuş bit vektörü $u_{\mathcal{F}}$, üretici matrisi $(G_N)_{\mathcal{A}}$ olan doğrusal blok kodun bir eşkümesini (coset) tanımlar [4]. Bu nedenle $u_{\mathcal{F}}$ 'nin bu kodun sendromu olduğu uygun bir parite matrisi bulunabilir. $G_N^{-1} = G_N$ olduğu için sendrom hesabı aslında kutupsal kodlama işleminin ardından dondurulmuş indislere karşılık gelen bitlerin alınmasıdır:

$$s^{N-K} = u_{\mathcal{F}} = (x^N G_N)_{\mathcal{F}}. \quad (1)$$

Kodlama ve kod çözme işlemleri şekil 1'de özetlenmiştir. Kodlama işlemi (1)'de tanımlanan işlemi gerçekleştirmektedir. Kod çözme ise \mathbf{x} ve \mathbf{u} 'nin tahminlerini çıktı olarak veren bir SEL kod çözücüsü ile gerçekleşmektedir. $\hat{\mathbf{x}}$, kaynak vektörünün tahminidir. "LLR hesabı" kutusu X ve Y kaynakları arasındaki BSC(p) olduğu kabul edilen ilintiye göre LLR değerlerini hesaplar:

$$L_i = \begin{cases} (-1)^{y_i} \cdot \log \frac{1-p}{p}, & i \in \{1, \dots, N - l_{crc}\} \\ 0, & i \in \{N - l_{crc} + 1, \dots, N\}. \end{cases} \quad (2)$$

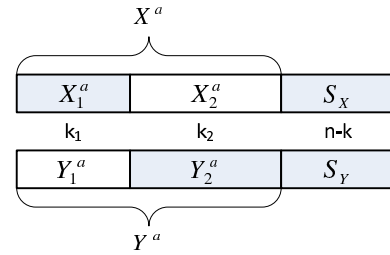
l_{crc} , eğer varsa CRC bitlerinin miktarını göstermektedir. İlk $(N - l_{crc})$ bitin istatistiği bilinmekte (Ber(p)) ve kod çözme için kullanılmaktadır. CRC bitleri ise buna dahil değildir ve düzgün

dağılımı olarak kabul edilmişlerdir. CRC kullanıldığı durumda, bloktaki bilgi bitlerinin miktarı $N' = N - l_{crc}$ 'ye düşmektedir fakat sıkıştırılmış bitlerin uzunluğu K olarak sabit kalmaktadır. Bu nedenle, sıkıştırma oranında ufak bir azalma olmaktadır.

Tek kaynak sıkıştırmanın yukarıda anlatılanlardan tek farkı, yan bilgi vektörü y^N 'nin sıfır vektörü 0^N ile değiştirilmesidir.

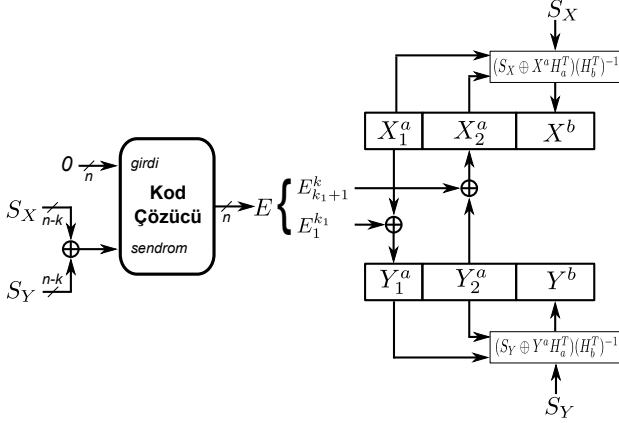
3. ASİMETRİK OLMAYAN SLEPIAN-WOLF

Düzenli dağılıma sahip kaynaklar için bir asimetrik olmayan SW tasarımı oluşturma yöntemi [8]'de önerilmiştir. Bu yöntem yakın zamanda, turbo kodlar kullanarak hem asimetrik olmayan hem de hız-uyarlamalı SW tasarımı oluşturmak üzere [9]'da kullanılmıştır. Bu bölümde, [8] yöntemini kullanarak kutupsal kodlar ile asimetrik olmayan SW kodlamanın nasıl inşa edilebileceğini gösteriyoruz.



Şekil 2: [8] yöntemiyle asimetrik olmayan SW kodlaması.

Tek bir kanal kodundan, sendrom yaklaşımını kullanarak, asimetrik olmayan SW kodlama oluşturan [8] yöntemi aşağıdaki gibi özetlenebilir. İki adet bağımsız özdeşçe dağılım, n uzunlukta, ilişkili vektör olan $\mathbf{x} = [\mathbf{x}^a \ \mathbf{x}^b]$ ve $\mathbf{y} = [\mathbf{y}^a \ \mathbf{y}^b]$ 'yi ele alalım. \mathbf{x}^a , \mathbf{x} vektörünün ilk k bitini, \mathbf{x}^b ise son $(n - k)$ bitini temsil etmektedir (aynı durum \mathbf{y} için de geçerlidir). Ayrıca, \mathbf{x}^a ve \mathbf{y}^a 'nın $\mathbf{x}^a = [\mathbf{x}_1^a \ \mathbf{x}_2^a]$ ve $\mathbf{y}^a = [\mathbf{y}_1^a \ \mathbf{y}_2^a]$ şeklinde ayrıştırlarını kabul edelim. \mathbf{x}_1^a , \mathbf{x}^a vektörünün ilk k_1 bitini, \mathbf{x}_2^a ise son k_2 bitini temsil etmektedir (aynı durum \mathbf{y}^a için de geçerlidir) ve $k_1 + k_2 = k$ 'dir. Kabul edelim ki kanal kodunun, $k \times n$ üretici matrisi G ve $(n - k) \times n$ parite matrisi $H = [H_a \ H_b]$ olsun. H_a , $(n - k) \times k$ boyutlu ve H_b $(n - k) \times (n - k)$ boyutlu alt-matrisleridir. Ayrıca kabul edelim ki H_b tersi var olacak şekilde seçilmiştir. Bir kodun sistematik halinin, bu tanımın özel bir durumu olduğuna dikkat edelim ($H_b = I_{n-k}$). Bu durumda, X kodlayıcısı, k_1 uzunluktaki \mathbf{x}_1^a vektörünü ve $(n - k)$ uzunluktaki $\mathbf{s}_x = \mathbf{x} H^T = \mathbf{x}^a H_a^T \oplus \mathbf{x}^b H_b^T$ vektörünü yollar. Y kodlayıcısı ise k_2 uzunluktaki \mathbf{y}_2^a vektörünü ve $(n - k)$ uzunluktaki $\mathbf{s}_y = \mathbf{y} H^T = \mathbf{y}^a H_a^T \oplus \mathbf{y}^b H_b^T$ vektörünü gönderir. X ve Y kodlayıcıları tarafından gönderilen bilgiler şekil 2'deki gölgeli kısımlara denk gelmektedir. Yani, X kodlayıcısı ($\mathbf{x}_1^a, \mathbf{s}_x$) ve Y kodlayıcısı ($\mathbf{y}_2^a, \mathbf{s}_y$) vektörlerini gönderir. k_1 'in değerini değiştirerek (ve $k_1 + k_2 = k$ eşitliği korunarak) SW eğrisi üzerindeki herhangi bir noktaya ulaşmak mümkün olmaktadır. X ve Y kaynakları düzgün dağılıma sahip ve bağımsız özdeşçe dağılımı oldukları kabul edildiği için $H(X) = 1$ ve $H(Y) = 1$ 'dir. Bu durumda kod, $(n - k) \geq nH(X|Y)$ olacak şekilde ayarlanır. Böylece toplam hız, entropi sınırını sağlar: $R = R_X + R_Y = k + 2 \times (n - k) = 2n - k \geq n[H(X) + H(Y|X)] = nH(X, Y)$.



Şekil 3: [8] yöntemiyle asimetrik olmayan SW kod çözümü.

Yukarıda anlatılan yöntemin kod çözme işlemi şekil 3'te gösterilmiştir. Kabul edelim ki $\mathbf{e} = \mathbf{x} \oplus \mathbf{y}$ hata vektörüdür, o zaman $\mathbf{s}_e = \mathbf{e}H^T = (\mathbf{x} \oplus \mathbf{y})H^T = \mathbf{s}_x \oplus \mathbf{s}_y$. Kanal kod çözücüsüne sıfır vektörü girdi olarak, \mathbf{s}_e vektörü ise eşküme (coset) indisi olarak verilir ve hata kestirimi $\hat{\mathbf{e}}$ çıktı olarak alınır. Daha sonra, $\hat{\mathbf{e}}$ 'nin yardımıyla \mathbf{x}_2^a ve \mathbf{y}_1^a sırasıyla \mathbf{y}_2^a ve \mathbf{x}_1^a kullanılarak hesaplanabilir. En son olarak, $\mathbf{x}^b = (\mathbf{s}_x \oplus \mathbf{x}^a H_a^T)(H_b^T)^{-1}$ ve $\mathbf{y}^b = (\mathbf{s}_y \oplus \mathbf{y}^a H_a^T)(H_b^T)^{-1}$ eşitlikleri kullanılarak \mathbf{x}^b ve \mathbf{y}^b elde edilir. Şunu da belirtmek gerekir ki, şekil 3'te açıkça gösterilmemiş olsa da kod çözücüyü giren sıfır vektörünün LLR hesabı X ve Y kaynakları arasındaki varsayımsal BSC(p) kanalına göre yapılır.

Genel hali yukarıda anlatılan bu yöntem kutupsal kodlara [10]'de sunulmuş olan sistematik halini kullanarak uygulanabilir. Sistematik kutupsal kodlar aşağıdaki gibi tanımlanmıştır. \mathbf{x} kod kelimesi $\mathbf{x} = (\mathbf{x}_B, \mathbf{x}_{B^c})$ şeklinde iki parçaya ayrılır. B , $|B| = |A|$ olacak şekilde $\{1, \dots, N\}$ 'nin bir alt kümesidir ve G_{AB} alt-matrisi terslenebilir. O zaman, \mathbf{x} kod kelimesinin "sistematik" ve "parite" kısımları sırasıyla şu şekilde yazılabilir

$$\mathbf{x}_B = \mathbf{u}_A G_{AB} + \mathbf{u}_{A^c} G_{A^c B} \quad (3)$$

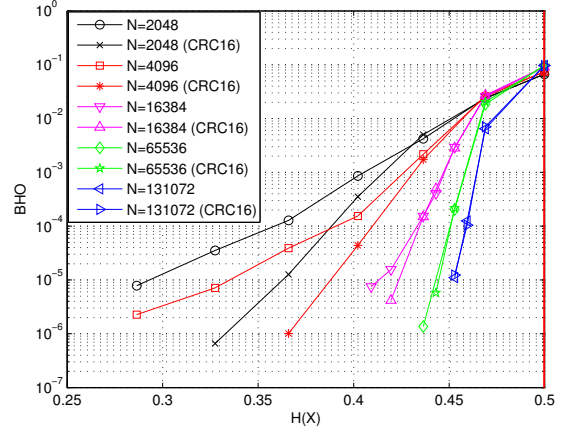
$$\mathbf{x}_{B^c} = \mathbf{u}_A G_{AB^c} + \mathbf{u}_{A^c} G_{A^c B^c}. \quad (4)$$

Bu tanımın bilindik bir sistematik kod tanımına göre tek farkı sistematik bitlerin kod kelimesinin ilk k biti yerine indisleri B kümesi ile belirlenmiş farklı pozisyonlara sahip olmasıdır. Bu şekilde (B, \mathbf{u}_{A^c}) parametresi ile tanımlanmış sistematik bir kodlayıcı $\mathbf{x}_B \mapsto \mathbf{x} = (\mathbf{x}_B, \mathbf{x}_{B^c})$ eşlemesini

$$\mathbf{u}_A = (\mathbf{x}_B \oplus \mathbf{u}_{A^c} G_{A^c B})(G_{AB})^{-1} \quad (5)$$

işlemini yaparak ve daha sonra \mathbf{u}_A 'yı (4)'de kullanarak \mathbf{x}_{B^c} 'yi hesaplamak suretiyle yapabilir.

Şimdi, yukarıda anlatılan [8] yöntemiyle asimetrik olmayan SW oluşturma işlemine geri dönerek şu eşleme yapılır: $\mathbf{x}^a = \mathbf{x}_B$, $\mathbf{x}^b = \mathbf{x}_{B^c}$ ve $\mathbf{s}_x = \mathbf{u}_{A^c}$. Bu şekilde yöntemin gerekleri yerine getirilmiş olur. Yani hata vektörü $\hat{\mathbf{e}}$ kullanılarak \mathbf{x}^a çözüldükten sonra, geri kalan kısım \mathbf{x}^b , \mathbf{x}^a ve \mathbf{s}_x kullanılarak hesaplanabilir. $\mathbf{x}^a = \mathbf{x}_B$ ve $\mathbf{s}_x = \mathbf{u}_{A^c}$ verildiği takdirde $\mathbf{x}^b = \mathbf{x}_{B^c}$ ve \mathbf{u}_A 'nın hesaplanması yukarıda anlatılmış olan sistematik kutupsal kodlama işleminden farklı değildir ve bu işlem [10]'te anlatıldığı üzere SE kutupsal kod çözücü kullanılarak etkin bir şekilde yapılabilir. [4]'te sunulan standart kutupsal kod tanımı için B , A 'nın farklı bir dizilimi (permutation)



Şekil 4: Asimetrik SW Bit Hata Oranı ($R = 0.5$).

olarak seçilebilir. Bu dizilim, bit tersine çevirme (bit reversal) işlemine karşılık gelir.

Bu asimetrik olmayan yöntemde kutupsal kodlarla CRC kullanımı da mümkündür. Her iki kaynak tarafından da kodlama blokları, $N' = N - l_{crc}$ uzunluğundaki bilgi bitlerine l_{crc} 'lik CRC eklenerek N bite tamamlanır. CRC işlemi doğrusal olduğu için $\mathbf{e} = \mathbf{x} \oplus \mathbf{y}$ vektörünün CRC'si de tutmak zorundadır. Böylece, SEL kanal kod çözücüsü hata vektörünü çözerken bu bilgiyi kullanabilir.

4. SAYISAL SONUÇLAR

Bu bölümde SEL kod çözücü ile gerçekleştirilen kaynak kodlama yöntemlerinin simülasyon sonuçlarıyla performanslarını sunmaktayız. X ve Y kaynaklarının kabul edilen bağlantı modeli $Y = X \oplus Z$ şeklindedir. Z , Ber(p) dağılımına sahiptir. Bütün simülasyonlarda kodlayıcıların hızları belli bir sabit değerde tutulmuş ve p değeri değiştirilerek farklı $H(X|Y)$ veya $H(X, Y)$ noktalarına erişilmiştir. Simülasyonlarda [7]'de sunulmuş olan SEL kod çözücü kullanılmıştır. Performansı artırmak için 16 bitlik CRC (CCITT) de eklenmiştir. Ancak kanal kodlamadan farklı olarak CRC \mathbf{u}_A bitlerine değil, \mathbf{x} "kod kelimesi" bitlerine eklenmektedir. Liste kod çözücü, listesi içerisinde en son kararı verirken CRC'den faydalanır. Bütün simülasyonlarda liste boyutu 32'ye ayarlanmıştır. Kod tasarımı ise [11]'de sunulan yöntem ile $R = 0.5$ için $p = 0.09$ 'a, $R = 0.25$ için $p = 0.03$ 'e optimize edilmiştir.

4.1. Asimetrik Slepian-Wolf ve Tek Kaynak Sıkıştırma

Bölüm 2'de değinildiği gibi tek kaynak kodlama, asimetrik Slepian-Wolf kodlamanın özel bir durumu olarak düşünülebilir ve dolayısı ile beklendiği gibi performans sonuçları tamamen aynı çıkmaktadır. Şekil 4'te $R = 0.5$ için BHO grafiği verilmiştir. Tablo 1'de ise farklı blok uzunlukları ve kod oranları için sonuçlar verilmiştir. Tablodaki değerler BHO 10^{-5} 'e düştüğündeki kaynak ilinti değerinin entropisini ($\mathcal{H}(p) = H(X|Y)$) göstermektedir. Örneğin $N = 65536$ için Slepian-Wolf sınırına olan 0.5 'e uzaklık 0.056 'dır.

Tablo 1: Asimetrik SW performansı ($H(X|Y)$ değerleri)

N	2048	4096	16384	65536	131072
$R = 0.5$	0.360	0.386	0.423	0.444	0.453
$R = 0.25$	0.118	0.153	0.190	0.205	0.210

4.2. Asimetrik Olmayan Slepian–Wolf

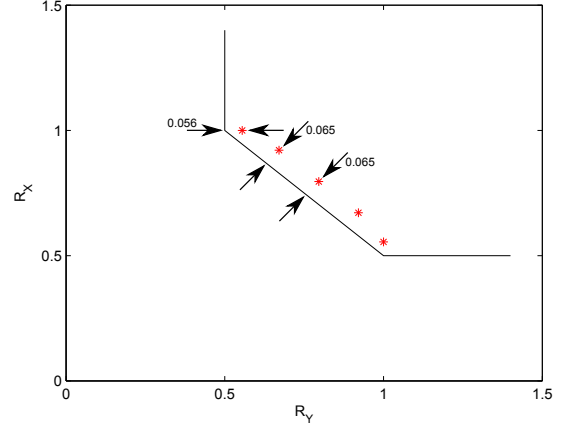
Asimetrik olmayan durumun $N = 65536$ için performansı şekil 5'te sunulmuştur. Şekildeki noktalar, BHO'nun 10^{-5} 'e düştüğü yerde alınmıştır. Bölüm 3'te anlatılan asimetrik olmayan yöntem ile yapılmış ancak asimetrik bir duruma karşılık gelen hız paylaşımını ($R_X = 0.5, R_Y = 1$) için ortaya çıkan performans değerleri bölüm 2'de sunulan asimetrik yöntem ile yapılmış simülasyonlarla tamamen aynıdır. Bu beklenen bir durumdur çünkü, bölüm 3'teki yöntemde asimetrik hız paylaşımını yapıldığı durumda, Y vektörü hatasız hesaplanabilmektedir. Buna ek olarak, aynı durumda, X kodlayıcısı tarafından da sadece sendrom bitleri gönderilmektedir. Sonuç olarak, bölüm 3 yöntemi bölüm 2 yöntemine indirgenmiş olmaktadır. Şekil 5'te, asimetrik hız paylaşımına ek olarak, simetrik ($R_X = 0.75, R_Y = 0.75$) ve bir asimetrik olmayan ($R_X = 0.875, R_Y = 0.625$) hız paylaşımına karşılık gelen noktalar da işaretlenmiştir. Şekilden de görülebileceği gibi bu noktaların performansları asimetrik duruma göre biraz kötüdür. Bu beklenen bir durumdur çünkü, asimetrik durumda Y kaynağı için hiç hata yapılmamaktayken, asimetrik olmayan durumda Y 'nin tahmini de hataya açıktır, buna ek olarak bu hatalar X 'nin tahminini de kötü yönde etkilemektedir. Ayrıca şekilden asimetrik olmayan tüm durumlar için hata oranlarının sabit olduğu da gözlemlenebilmektedir.

5. SONUÇLAR

Bildiriye öncelikle tek kaynak ve asimetrik Slepian–Wolf sıkıştırmanın kutupsal kodlar kullanılarak nasıl yapıldığını gözden geçirerek başladık. Daha sonra, [8]'te sunulan genel çerçeveyi ve kutupsal kodların sistematik halini [10] kullanarak kutupsal kodlayıcı ve kod çözümlerinin asimetrik olmayan Slepian–Wolf problemi için nasıl uygulanabileceklerini gösterdik. En iyi sonuçlara ulaşabilmek için CRC'li sıralı eleme liste kod çözümleri kullanılmıştır. Elde edilen performans sonuçları oldukça iyi olmasına rağmen literatürde turbo ve LDPC kodlar kullanılarak ulaşılmış en iyi sonuçlara [12, 13, 9] göre SW sınırına biraz daha uzaktır. Ancak kutupsal kodların bazı avantajları bulunmaktadır. Sendrom uzunluğu (sıkıştırma oranı) kod performansında fazla kayba neden olmadan kolayca değiştirilebilmektedir. Bu da değişen ilinti koşullarında kutupsal kodların kullanılması için önemli bir teşvik olabilir. Ayrıca kod gerçekleştirme karmaşıklığı ve kod çözme gecikmesi bakımından da LDPC ve turbo kodlara göre avantajları olabilir. Bu da gelecek için bir araştırma konusudur.

6. KAYNAKÇA

- [1] J. D. Slepian and J. K. Wolf, "Noiseless coding of correlated information sources," *IEEE Trans. Inform. Theory*, vol. IT-19, pp. 471–480, July 1973.
- [2] A. Wyner, "Recent results in the shannon theory," *In-*



Şekil 5: Asimetrik olmayan SW performansı ($N = 65536$).

- [1] *formation Theory, IEEE Transactions on*, vol. 20, no. 1, pp. 2–10, 1974.
- [3] S. S. Pradhan and K. Ramchandran, "Distributed source coding using syndromes (DISCUS): design and construction," in *Proc. of the IEEE International Data Compression Conference (DCC)*, pp. 158–167, Mar. 1999.
- [4] E. Arıkan, "Channel polarization: A method for constructing Capacity–Achieving codes for symmetric Binary–Input memoryless channels," *IEEE Trans. Inform. Theory*, vol. 55, pp. 3051–3073, July 2009.
- [5] S. B. Korada, *Polar codes for channel and source coding*. PhD thesis, EPFL, Lausanne, Switzerland, July 2009.
- [6] E. Arıkan, "Source polarization," in *Proc. of the IEEE International Symp. Inform. Theory*, (Austin, Texas, U.S.A.), pp. 899–903, June 2010.
- [7] I. Tal and A. Vardy, "List decoding of polar codes," in *Proc. of the IEEE International Symp. Inform. Theory*, (St. Petersburg, Russia), pp. 1–5, Aug. 2011.
- [8] N. Gehrig and P. L. Dragotti, "Symmetric and asymmetric Slepian–Wolf codes with systematic and nonsystematic linear codes," *IEEE Communications Letters*, vol. 9, pp. 61–63, Jan. 2005.
- [9] M. Zamani and F. Lahouti, "A flexible rate slepian-wolf code construction," *IEEE Transactions on Communications*, vol. 57, pp. 2301–2308, Aug. 2009.
- [10] E. Arıkan, "Systematic polar coding," *IEEE Communications Letters*, vol. 15, pp. 860–862, Aug. 2011.
- [11] I. Tal and A. Vardy, "How to construct polar codes," *arXiv:1105.6164*, May 2011.
- [12] A. D. Liveris, Z. Xiong, and C. N. Georghiades, "Compression of binary sources with side information at the decoder using LDPC codes," *IEEE Communications Letters*, vol. 6, pp. 440–442, Oct. 2002.
- [13] V. Stankovic, A. D. Liveris, Z. Xiong, and C. N. Georghiades, "Design of Slepian–Wolf codes by channel code partitioning," in *Proc. of the IEEE International Data Compression Conference (DCC)*, pp. 302–311, Mar. 2004.