Doğrusal Kodların Spektrum Ağırlık Fonksiyonlarının Hesaplanması

Orhan Gazi [1], A. Özgür Yılmaz [2]

[1] Elektronik Haberleşme Mühendisliği Bölümü, Çankaya Üniversitesi
Balgat, 06530, Ankara. *e-posta: o.gazi@cankaya.edu.tr*
[2] Elektrik-Elektronik Mühendisliği Bölümü, Orta Doğu Teknik Üniversitesi
Balgat, 06530, Ankara. *e-posta: aoyilmaz@eee.metu.edu.tr*

ABSTRACT

We introduce a simple method to evaluate the weight spectrum functions of linear codes. Analytical upper bounds for bit error probabilities of turbo codes and serially concatenated codes are evaluated for different interleaver lengths.

Anahtar Sözcükler: *Ağırlık sıralama fonksiyonu, şartlı ağırlık sıralama fonksiyonu, düzgün serpiştirici, seri birleşik kod, paralel birleşik kod.*

1. GİRİŞ

Birleşik kodlar ilk olarak Forney tarafından 1966 yılında tanıtılmıştır [1]. Forney sonsal *(APP)* olasılıkları kullanarak seri birleşik konvolusyonel kodları (SBKK) çözümlemiştir. Forney'in çalışmasını takiben bir kaç tane *APP* algoritması geliştirilmiştir. Bu algoritmaların en bilinenlerinden birisi *BCJR* [2] algoritmasıdır ve de uzun zaman boyunca oldukça karmaşık olduğu düşünülmüştür. Birleşik kodlar için yinelemeli çözümleme çalışmaları 1990 yıllarının başlarında tekrar hız kazanmaya başlamıştır. Bu çalışmaların sonucunda 1993 yılında Turbo kodlar [3] geliştirilmiştir. Turbo kodların bulunması kodlama teorisi için bir dönüm noktası olmuştur. Turbo kodlar, paralel birleştirilmiş konvolusyonel kodlar (*PBKK*) olarak ta bilinmektedirler. Turbo kodların keşfiyle yinelemeli algoritmalar üzerine olan eğilim artmıştır ve de birleşik kod yapılarının yinelemeli çözümlenmesi üzerine yapılan çalışmalarda bir artış gözlenmiştir. Turbo kodların icadından sonra seri birleşik konvolusyonel kodlar (SBKK) icat edilmiştir [4]. SBKK'lar turbo kodlara göre daha iyi performans sağlayabilmektedir. Daha sonra hem paralel hemde seri yapıyı içeren hibrid birleşik kodlar geliştirilmiştir [5].

Turbo kodların performansları benzetim çalışmaları ile ispat edilse bile, analitik olarak turbo kodların şaşırtan performansları ilk olarak [6] çalışmasında açıklanmıştır. Bu çalışmada düzgün serpiştirici tanıtılmış ve de düzgün serpiştirici kullanarak turbo kodlar için analitik performans sınırları çizilmiştir. Düzgün serpiştirici kullanarak analitik peformans sınırlarını çizme yöntemi SBKK'lara da uygulanmıştır [6]. Düzgün serpiştirici ile bit hata olasılıkları için analitik sınırlar çizebilmemiz için birleşik kodun spektrum ağırlık fonksiyonlarına ihtiyaç vardır. Spektrum ağırlık fonksiyonlarının hesabı özellikle uzun veri sözcükleri için oldukça zor bir iştir.

Küçük oranlardaki konvolusyonel kodların spektrum ağırlık fonksiyonlarının hesaplanması ile ilgili çalışma [7]'de yapılmıştır. Bu çalışmada ileri-geri algoritması spektrum ağırlık fonksiyonunun hesaplanması için kullanılmıştr. Ayrıca Viterbi türü bir metod [8]'de tanıtılmıştır. Biz bu makalemizde spektrum ağırlık fonksiyonlarının hesaplanması için basit ve de kullanışlı bir metodu tanıtacağız ve de bu metodu kullanarak SBKK'lar ile PBKK'lar için bit hata olasılıkları için analitik sınırlar çizeceğiz.

1.1. Ağırlık Sıralama Fonksiyonu (ASF)

Ağırlık sıralama fonksiyonu kod sözcükleri hakkında bilgi taşır (belirli bir Hamming ağırlığındaki kod sözcüklerinin sayısı),

$$A(X) = \sum_{x=d_{min}}^{n} A_x X^x$$

Bu ifade de ki A_x Hamming ağırlığı x olan kod sözcüklerinin sayısını ifade etmektedir. X ise bir değişkendir.

1.2. Girdi Çıktı Ağırlık Sıralama Fonksiyonu (GÇASF)

ASF kod sözcükleri hakkında bilgi verir. Girdi veri sözcükleri hakkında herhangi bir bilgi vermez. *GÇASF* ise hem kod sözcükleri hakkında hemde kod sözcüklerini oluşturan veri sözcükleri hakkında bilgi içerir.

$$A(W, X) = \sum_{w,x} A_{w,x} W^w X^x$$

Bu ifade de $A_{w,x}$ Hamming ağırlığı w olan veri sözcükleri tarafından üretilen Hamming ağırlığı x olan kod sözcüklerinin sayısını vermektedir. W ve X ise göreceli değişkenlerdir.

1.3. Girdi Fazlası Ağırlık Sıralama Fonksiyonu (GFASF)

GFASF kod sözcüklerindeki pariti bitlerinin Hamming ağırlıkları hakkında bilgi içerir.

$$A(W,Z) = \sum_{w,z} A_{w,z} W^w Z^z,$$

Bu ifadede ki $A_{w,z}$ Hamming ağırlığı w olan veri sözcükleri tarafından üretilen parity ağırlığı z olan kod sözcüklerinin sayısını belirtir.

1.4. Şartlı Ağırlık Sıralama Fonksiyonu (ŞASF)

ŞASF ise Hamming ağırlığı w olan veri sözcükleri tarafından üretilen kod sözcüklerindeki Hamming ağırlığı z olan parity sözcükleri hakkında bilgi verir.

$$A_w(Z) = \sum_{z} A_{w,z} Z^z$$

1.5. Spektrum Ağırlık Fonksiyon Katsayıları Arasındaki İlişkiler

Spektrum ağırlık fonksiyonlarının katsayıları arasındaki ilişkiler aşağıda özetlenmiştir.

GFASF-GÇASF $A_{w,z} = A_{w,x}\big|_{z=x-w}$
ASF-GFASF $A_x = \sum_{x=w+z} A_{w,z}$
ASF-GÇASF $A_x = \sum_{i=x} A_{w,i}$

1.6. Spektrum Ağırlık Fonksiyonları Arasındaki İlişkiler

Spektrum ağırlık foksiyon katsayılarında olduğu gibi spektrum ağırlık fonksiyonları arasında da benzer ilişkiler vardır ve de bu ilişkiler aşağıdaki gibi özetlenmiştir.

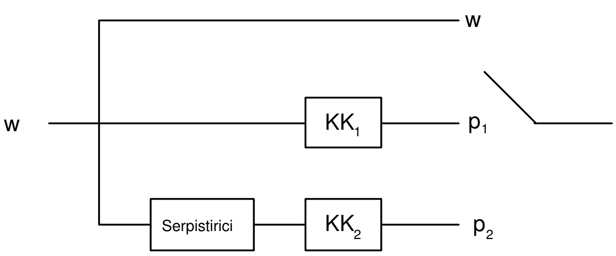
GÇASF-ŞASF $A(W,X) = \sum_w W^w A_w(X)$
ŞASF-GFASF $A_w(Z) = \frac{1}{w!} \frac{\partial^w A(W,Z)}{\partial W^w}\Big|_{W=0}$
ASF-ŞASF $A(X) = A(W,Z)|_{\substack{W=X \\ Z=X}}$

2. Birleşik Kodların Spektrum Fonksiyonları

Birleşik kodlar üç değişik yol izlenerek oluşturulabilirler. Seri birleştirme metodu ile, paralel birleştirme metodu ile ya da hem seri hem de paralel birleştirme metodu ile, en son ifade edilen metod hibrid tekniği olarakta bilinmektedir. Birleşik kodların spektrum ağırlık fonksiyonları bir kez bulunduktan sonra kodların performansları için analitik üst sınırlar çizmek mümkün olmaktadır.

Birleşik kodlar doğrusal kodlar olarak düşünülebilirler. Doğrusal kodların performans üst sınırları için matematiksel ifadeler mevcuttur ve de bu ifadeler birleşik kodlar için de kullanılabilirler. Birleşik kodların spektrum fonksiyonları hesaplanabilir ve de analitik performans sınırlarının çizilmesi amacı ile kullanılabilir. PBKK'un düzgün serpiştirici kullanılarak analitik analizi [6] çalışmasında yer almaktadır. Düzgün serpiştirici kullanarak birleşik kodların bit hata performansları için analitik performans sınırlarını çizebilmek önemli bir gelişmedir. Örgü sonlandırmaya tabi tutulan konvolusyonel kodlar blok kodlar gibi düşünülebilir. Ve de blok kodlar için yapılan çalışmalar örgü sonlandırılmış konvolusyonel kodlara da uygulanabilir. Birleşik kodların spektrum fonksiyonlarını takip eden bölümlerde anlatacağız.

2.1. Paralel birleşik kodun GÇASF



Şekil 1: Paralel birleşik kod. CC1, CC2 parça kodlardır. Bu kodlar blok, konvolusyonel ya da bunların kombinezonu olabilirler.

Düzgün serpiştirici yaklaşımı kullanarak PBKK'un *GÇASF* katsayıları aşağıdaki gibi bulunabilir.

$$A_{w,x}^{C_p} = \sum_{\substack{x_1,x_2 \\ x_1+x_2=x}} \frac{A_{w,x_1}^{C_1} A_{w,x_2}^{C_2}}{\binom{N}{w}},$$

bu eşitlikteki $A_{w,x_1}^{C_1}$ ve $A_{w,x_2}^{C_2}$ ifadeleri parça konvolusyonel kodlar KK_1 ve KK_2'nin *GÇASF* katsayıları olmaktadır, $A_{w,x}^{C_p}$ ise paralel birleşik kodun *GÇASF* katsayılarıdır. N serpiştiricinin uzunluğu, w, x ise veri ve de kod sözcüklerinin Hamming ağırlığı olmaktadır. Düzgün serpiştirici kullanarak, paralel birleşik kodların *ŞASF* aşağıdaki ifade ile bulunmaktadır,

$$A_w^{C_p}(Z) = \frac{A_w^{CC_1}(Z).A_w^{CC_2}(Z)}{\binom{N}{w}}.$$

C_p kodunun *GFASF* şöyle elde edilir,

$$A^{C_p}(W,Z) = \sum_{w=1}^{k} W^w A_w^{C_p}(Z)$$

bu ifadede k veri dizilerinin uzunluğunu belirtmektedir. Eğer blok kodlar birleşik kod da parça kodlar (KK_1 bir (n,k) blok koddur) olarak kullanılmışsa, parça kodun l kod sözcüğü için *GFASF*

$$A^{C_1^l}(W,Z) = [A^{C_1}(W,Z)]^l$$

ifadesiyle hesaplanır. Paralel birleşik kodların spektrum fonksiyonları biliniyorsa, en yüksek bit hata olasılıkları için analitik sınırlar aşağıdaki ifade ile hesaplanmaktadır,

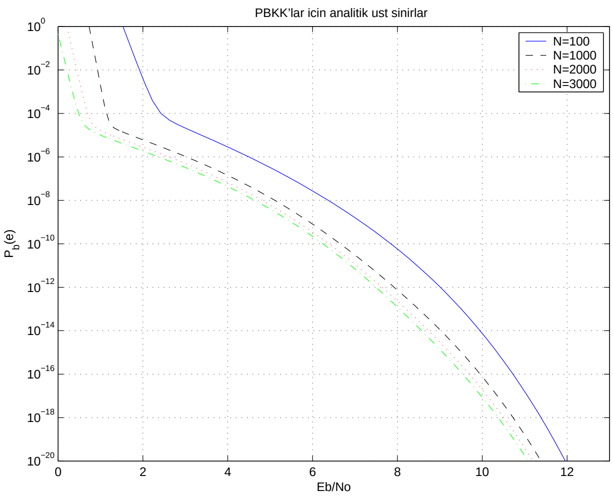
$$P_b(e) \leq \sum_{w=1}^{k} \frac{w}{k} W^w A_w^C(Z) \mid_{W=Z=e^{-R_c E_b/N_o}} . \quad (1)$$

Bu ifade daha fazla basitleştirilerek (2) deki şekliyle verilebilir,

$$P_b \cong \frac{1}{2} \sum_m D_m erfc(\sqrt{m \frac{R_c E_b}{N_o}}) \quad (2)$$

bu ifadedeki R_c kod oranı olmaktadır, $\frac{E_b}{N_o}$ ise AWGN kanal için bit enerjisinin gürültü miktarına oranıdır. D_m ifadesi ise şöyle elde edilir,

$$D_m = \sum_{z+w=m} \frac{w}{k} A_{w,z}. \quad (3)$$

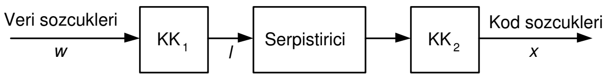


Şekil 2: Paralel birleşik kod bit hata olasılıkları için üst sınırlar. KK_1, KK_2 üretici matriksi $(1, 5/7)_{oktal}$ olan parça konvolusyonel kodlardır.

2.2. Seri birleşik kodların GÇASF

Seri birleşik kodun *GÇASF* $A^{C_s}(W,X)$ parça kodların *ŞASFs* çarpımlarının normalize edilmiş şekliyle elde edilir.

$$A^{C_s}(W,X) = \sum_{l=0}^{N} \frac{A_l^{C_o}(W) \times A_l^{C_i}(X)}{\binom{N}{l}}$$

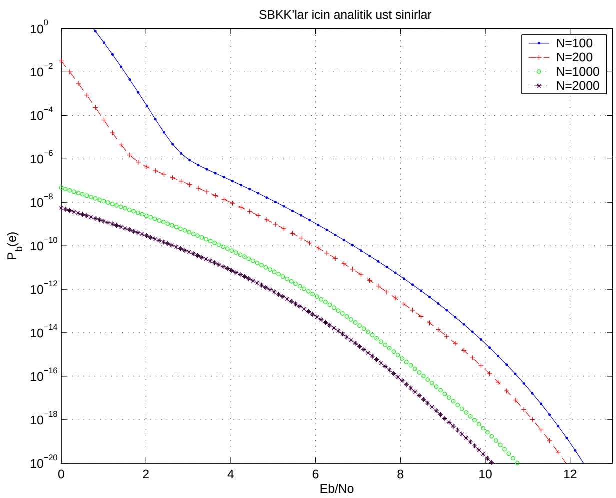


Şekil 3: Seri birleşik kod. KK_1, KK_2 parça kodlar. N serpiştiricinin uzunluğunu belirtir.

seri birleşik kodun *GÇASF* katsayıları $A_{w,x}^{C_s}$ Hamming ağırlıpı w olan veri sözcüklerinin ürettiği Hamming ağırlığı x olan kod sözcüklerinin sayısını ifade eder.

$$A_{w,x}^{C_s} = \sum_{l=0}^{N} \frac{A_{w,l}^{C_o} \times A_{l,x}^{C_i}}{\binom{N}{l}},$$

bu ifadedeki l dış kod sözcüklerinin Hamming ağırlığını ifade eder x ise iç kod sözcüklerinin Hamming ağırlığını gösterir. Birleşik kodların *GÇASF* elde edildikten sonra denklemler (2) ve (3) kullanılarak bit hata olasılıkları için üst sınırlar çizmek mümkün olmaktadır. Şekil 4'te SBKK'lar için farklı uzunlukta şerpiştiriciler için bit hata oranları için üst sınırlar çizilmiştir.



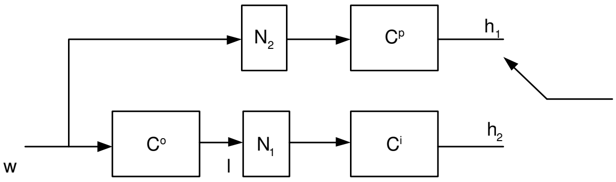
Şekil 4: Seri birleşik konvolusyonel kodun bit hata olasılığı için analitik üst sınırlar. Parça kodların üretici matriksi $(1, 5/7)_{oktal}$'dır. Serpiştiricinin uzunluğu N'dir.

2.3. Hibrid Birleşik Kodlar

Seri ve de paralel kod yapıları kombine edilerek hibrid kodlar üretilirler. Şekil 5'te paralel birleşik kod içeren bir hibrid yapı gösterilmektedir. $R_c^p = k/n_1, R_c^o = k/p, R_c^i = p/n_2$ ifadeleri paralel konvolusyonel kod, dış ve de iç kodlar için sırası ile kod oranlarını göstermektedir. Sistemde uzunlukları N_1 ve N_2 olan iki tane serpiştirici mevcuttur. Hibrid kodun kod oranı $R_c = k/(n_1 + n_2)$ eşitliği ile bulunur. Hibrid kodun GÇASF katsayıları aşağıdaki gibi bulunurlar.

$$A^{H}_{w,h} = \sum_{\substack{l=0 \\ h_1+h_2=h}}^{N_2} \frac{A^{C_p}_{w,h_1} \times A^{C_o}_{w,l} \times A^{C_i}_{l,h_2}}{\binom{N_1}{w}\binom{N_2}{l}}$$

bu ifadede $A^{H}_{w,h}, A^{C_p}_{w,h_1}, A^{C_o}_{w,l}, A^{C_i}_{l,h_2}$ Hibrid, paralel, dış ve de iç kod için GÇASF katsayılarıdır. İç kod yinelemeli kodlardan seçilebilir [5].



Şekil 5: Hibrid birleşik kod.

3. Spektrum Ağırlık Fonksiyonlarının Hesaplanması

Herhangi bir kodun spektrum fonksiyonunu bulmak için iki yol izlenebilir. İlk izlenecek yolda, belirli bir uzunluk değeri için var olan bütün veri sözcükleri üretilir, ve de bu veri sözcükleri kodlayıcıdan geçirilerek kod sözcükleri elde edilir. Daha sonra elde edilen kod sözcüklerinin ve de onları üreten veri sözcüklerinin Hamming ağırlıkları hesaplanır. Aynı Hamming ağırlığa sahip veri sözcüklerinin ürettiği kod sözcüklerinin sayısı bulunularak spektrum fonksiyonlar oluşturulurlar. İkinci metod ise bizim burda önericeğimiz polinom metodudur. Bu metodu bir örnekle açıklayalım.

Örnek: Üretici matriksi $(1, 5/7)_{oktal}$ olan konvolusyonel kodun *GFASF*'nu bulalım. Konvolusyonel kodlayıcı üç şekilde gösterilebilir, bunlar blok çizenek, durum çizeneği ve de durum geçiş matriksidir. Şekil 6'da durum ve de blok çizenekleri gösterilmiştir. Durum çizeneğindeki bilgiler durum geçiş matriksine aşağıdaki gibi aktarılmıştır. Durum çizeneğinin her dalındaki bilgi $W^w Z^z$ şeklinde bir polinom ile ifade edilmiştir bu polinomdaki W ve Z göstermelik değişkenlerdir, w ve z ise veri ve de pariti sözcüklerinin Hamming ağırlık değerleridir. $T(W, Z)$ matriksinin (i, j) koordinatındaki elemanı durum çizeneğindeki i dalından j dalına geçişteki bilgileri taşır.

$$T(W,Z) = \begin{pmatrix} 1 & 0 & WZ & 0 \\ WZ & 0 & 1 & 0 \\ 0 & W & 0 & Z \\ 0 & Z & 0 & W \end{pmatrix}$$

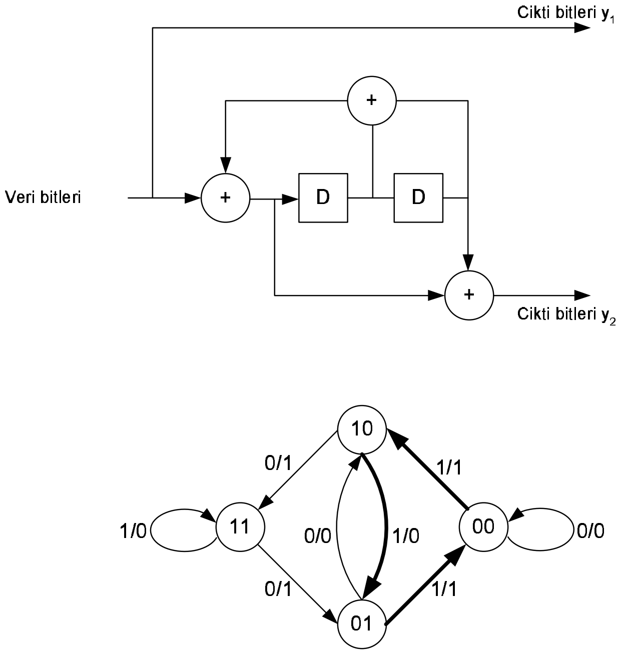
k uzunluğundaki veri sözcükleri için, blok geçiş matriksi $F(W, Z)$ şöyle tanımlanmıştır:

$$F(W,Z) = T(W,Z)^k$$

$F(W, Z)$ matriksinin her elemanı bir polinomdur ve de bütün elemanlarının toplamı konvolusyonel kodun *GFASF* $A(W, Z)$ yi verir.

$$A(W,Z) = \sum_{i,j} F_{i,j}(W,Z)$$

$F(W, Z)$ matriksinin $(2, 3)$ koordinatlarındaki eleman ise örgü çizelgesi durum 2 den başlayıp durum 3'te biten kod sözcüklerinin *GFASF*'nu verir. Bizim kullanacağımız konvolusyonel kodlar her zaman için örgü sonlandırma işlemine tabi tutulacaklardır (Diğer bir deyişle örgü çizelgeleri durum 0 dan başlar ve de durum 0 da sonlanırlar). Bundan ötürü $F(W, Z)$'nin (0,0) bölgesindeki elemanı konvolusyonel kod hakkındaki bütün bilgiyi taşımaktadır, yani kodun *GFASF*'nu $F_{0,0}(W, Z)$'dir.



Şekil 6: Üreticisi $(1, 5/7)_{oktal}$ kod için durum makinesi.

4. Spektrum Ağırlık Fonksiyonlarının Hesaplanması

Küçük uzunluktaki veri sözcükleri için bile doğrusal kodların *GFASF* hesabı oldukça güç bir iştir. *GFASF* hesapları için genellikle *Matlab* kullanılır. İki polinomum çarpımı şu şekilde yapılabilir. Önce polinom katsayıları değişkenlerin artan kuvvetlerine göre sıralanır, elde edilen bu iki katsayı vektörünün konvolusyoneli hesaplanır, sonuç vektöründeki katsayılar kullanarak tekrar bir polinom yazılır. Eğer polinomumuz iki değişken içeriyorsa bu durumda iki boyutlu konvolusyonel işlemi kullanılır (yani *conv2 matlab* fonksiyonu). Eğer polinomlarımızda n tane değişken varsa bu durumda n boyutlu konvolusyonel kullanılır (yani *convN*).

Uzunluğu k olan veri sözcükleri için blok geçiş matriksini (i.e. $T(W,Z)^k$) hesaplamak için şu yol takip edilir. $T(W,Z)$ matriksinin her elemanı $p \times p$ matrik şeklinde ifade edilir. Burdaki p rasgele bir değerdir. Büyük p değerleri daha iyi sonuçlar üretirler. Ama orta düzeyde bir p değeride yeterlidir. $T(W,Z)$'nin her elemanı matriks olarak ifade edildikten sonra, matriksin kuvveti alınabilir. $T(W,Z)$'in kuvvetini alırken normal matriks çarpma işlemi uygulanır. Matrikslerin elemanlarını çarparken, 2 boyutlu konvolusyonel işlemini kullanacağız, çünkü biz $T(W,Z)$ elemanlarını matriks olarak ifade ettik, polinom çarpımıda konvolusyonel alma işlemine denk gelir. Yalnız her iki boyutlu konvolusyonel alma işlemi sonucunda sonuç matriksinin boyutları iki katına çıkar, bu yüzden çarpım sonuç matriksi eski boyutlarına kırpılır.

Örnek: $T_{2,1}(W,Z) = WZ$ elemanı matrik olarak aşağıdaki gibi ifade edilmiştir:

$$T_{(2,1)}(W,Z) = \begin{array}{cccccc c} W^0 & W^1 & W^2 & W^3 & W^4 & W^5 & \\ 0 & 0 & 0 & 0 & 0 & 0 & Z^0 \\ 0 & 1 & 0 & 0 & 0 & 0 & Z^1 \\ 0 & 0 & 0 & 0 & 0 & 0 & Z^2 \\ 0 & 0 & 0 & 0 & 0 & 0 & Z^3 \\ 0 & 0 & 0 & 0 & 0 & 0 & Z^4 \\ 0 & 0 & 0 & 0 & 0 & 0 & Z^5 \end{array}$$

Şekil 7: WZ polynomunun matriks gösterimi.

5. SONUÇLAR

Spektrum ağırlık fonksiyonlarının hesaplanmaları esnasında pratikte karşımıza çıkacak zorluklardan kurtulmak için spektrum ağırlık fonksiyonlarının hesabı için basit bir yöntem önerdik ve de bu yöntemi kullanarak seri ve de paralel birleştirilmiş konvolusyonel kodların değişik serpiştirici uzunlukları için spektrum ağırlık fonksiyonlarını hesapladık.

KAYNAKLAR

[1] G. D. Forney, Jr., Concatenated Codes, M.I.T. Press, Cambridge, MA, USA, 1966.

[2] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate", *IEEE Trans. Inf. Theory*, Vol. IT-20, March 1974, pp. 284-287.

[3] C. Berrou, A. Galavieux, and P. Thitimajshima, "Near Shannon Limit Error-Correcting Coding and Decoding: Turbo Codes", *Proceeding of IEEE International Communications Conference '93.*

[4] S. Benedetto, D. Divsalar, G. Montorsi, F. Pollara, "Serial Concatenation of Interleaved Codes: Performance Analysis, Design, and Iterative Decoding.", *IEEE Transactions on Information Theory,* Vol. 44, No. 3, May 1998.

[5] D. Divsalar and F. Pollara, "Serial and hybrid concatenated codes with applications", *Proceedings of the International Symposium on Turbo Codes and Related Topics,* pp. 80-87, Brest, France, 3-5 September, 1997.

[6] S. Benedetto, G. Montorsi, "Unveiling Turbo Codes: Some Results on Parallel Concatenated Coding Schemes", *IEEE Transactions on Information Theory,* Vol. 42, No. 2, March 1996.

[7] J. Conan, "The Weight Spectra of Some Short Low-Rate Convolutional Codes", *IEEE Transaction on Communications,* Vol. Com-32, N0. 9, September 1984.

[8] H. Lajos, T. H. Liew and B. L. Yeap, Turbo Coding, Turbo Equalisation and Space-Time Coding for Transmission over Fading Channels, John Wiley and Sons.