SÖZDE RASTSAL SAYI ÜRETİMİNİN KRİPTOGRAFİK AÇIDAN İNCELENMESİ

[1]Fatma BÜYÜKSARAÇOĞLU, [2] Ercan BULUŞ
[1]Trakya Üniversitesi, Mühendislik Mimarlık Fakültesi, Bilgisayar Mühendisliği, EDİRNE
[2]Namık Kemal Üniversitesi, Çorlu Mühendislik Fakültesi, Bilgisayar Mühendisliği, Çorlu-TEKİRDAĞ
fbuyuksaracoglu@trakya.edu.tr, ercanbulus@corlu.edu.tr

ÖZET

Sözde Rastsal Sayı Üreteci (Pseudorandom Number Generator-PRNG), öğeleri arasında kolay kolay ilişki kurulamayacak bir sayı dizisi üreten algoritma türleridir. Kriptografik Sözde-Rastsal Sayı Üreteçleri şifreleme yöntemlerinin anahtar üretim sürecinde önemli rol oynamaktadır. Son dönemlerde akış şifreleme algoritmalarının da gelişmesiyle bu üreteçlerin kullanımı ve algoritmik yapıları üzerinde gelişmeler sağlanmıştır. Çalışmamızda bu üreteçler için kullanılan en yaygın ve önemli algoritmik yapı olan LFSR (Lineer Feedback Shift Register- Doğrusal Geri Beslemeli Öteleyici Saklayıcı)' lara ve bu yapıya uygun örneklere yer verilmiştir. Sayı üreteçlerinin kullanımının gelişmesi üretilen sayıların güvenilirliği sorusunu doğurmuştur. Şifrelemenin temeli olan güvenlik konusu uygulanan test yöntemleriyle giderilmeye çalışılmıştır. Çalışmamızda ayrıca üretilen sayıların nasıl test edileceği konusunda da örneklere yer verilmiştir.

1. GİRİŞ

Kriptografide kullanılan Rastgele Sayı Üreteçlerinde amaç kriptografik uygulamalar için örneğin anahtar üretimi rastgele sayılar üretmektir. Kriptografik açıdan sahte-rastgele (pseudo-random) sayılar kullanmak şifreleme gücünü arttırmaktadır. Bu tip sayıları üretmek için sözde rastsal sayı üreteçleri (pseudo-random number generator) kullanılmaktadır[9] .

Rastgelelik içermek amacıyla genelde (binlerce bitten oluşan) geniş bir havuz kullanılır ve havuzdaki her bir bit giriş gürültüsünün (input noise) her bir bitine ve havuzdaki diğer her bite kriptografik olarak güçlü bir yolla bağımlı yapılır. Önemli olan nokta, verinin herhangi bir dış gözlemci için kestirilemez olmasıdır.

Kriptografik Sözde rastsal Sayı Üreteçleri tipik olarak, rastgelelik içeren geniş bir havuza ('çekirdek değeri') sahiptir. Havuzdan veri alınarak havuza iade edilen bitler havuzun içeriğinin ortaya çıkmasını önlemek için (veriyi seçimlik olarak) bir şifreleme fonksiyonu üzerinden çalıştırırlar. Daha fazla bite ihtiyaç olduğunda havuzdaki her bir biti diğer her bite bağımlı yapan uygun bir rastgele anahtarla (havuzun hiç kullanılmamış tarafından olabilir) havuzun içeriğini şifreleyerek havuzu karıştırır. Önceki veya gelecekteki kestirimleri daha da zorlaştırmak için havuz karıştırılmadan önce yeni çevresel gürültüler havuza eklenmelidir. Rastgele Sayı Üreteci yetersiz yapılmışsa sistemin en zayıf noktaları olacaklardır [2].

Sözde rastsal sayı üreteçleri deterministik bir bilgisayarda çalıştıkları için deterministtik algoritmalardır ve bu tür bir algoritma ile üretilen sayı dizisinin gerçek bir rastsal dizide olmayan bir özelliği olacaktır: periyodiklik. Şurası kesindir ki, eğer üreteç sabit miktarda hafıza kullanıyorsa yeterli sayıda döngü adımından sonra aynı içsel duruma ikinci kez gelecektir ve ondan sonra da sonsuza dek tekrar edecektir. Periyodik olmayan bir üreteç tasarlanabilir ancak bu tür bir sistemin ihtiyaç duyduğu hafıza miktarı sistem çalıştıkça büyüyecektir. Buna ek olarak bir sözde rastsal sayı üreteci keyfi bir başlama noktasından, ya da çekirdek durumundan, başlatılabilir ve o andan itibaren özdeş bir sayı dizisi üretir. Periyodikliğin pratik önemi sınırlıdır. Eklenen her bir hafıza biti ile maksimum periyot iki katına çıkar. Herhangi bir bilgisayarın evrenin beklenen yaşam süresi boyunca hesaplayamayacağı kadar uzun periyoda sahip sözde rastsal sayı üreteçleri inşa etmek mümkündür.

Şifre bilimdeki cevaplanmamış sorulardan biri de iyi tasarlanmış bir sözde rastsal sayı üretecinin çıktısını, çekirdeğini (başlangıç parametrelerini) bilmeden, gerçek rastsal gürültüden ayırt etmenin mümkün olup olmayacağıdır. Şifre bilimdeki pek çok uygulama uygun bir sözde rastsal sayı üretecinin çıktısının gürültüden ayırt edilmeyeceği varsayımına dayanır. En basit örneği akış şifresidir. Bu algoritma gizli bir mesajı, rastsal sayı üretecinin çıktısı ile XOR işlemine tabi tutar. Bu tür rastsal sayı üreteçlerinin tasarımı bir hayli zordur ve çoğu program çok daha basit üreteçler kullanır. Uygulamada, pek çok sözde rastsal sayı üreteci istatistiksel olarak önemli testleri geçmelerini engelleyen bazı durumlar sergiler. Bunlardan sadece birkaçını söylemek gerekirse:

- Bazı çekirdek (başlangıç) durumları için beklenenden daha kısa periyotlar
- Kötü boyutsal dağılım
- Birbirini takip eden değerlerin bağımsız olmaması
- Bazı bitlerin diğerlerinden 'daha rastsal' olabilmesi
- Tekbiçimlilik eksikliği [9]

Şifre bilimsel olarak uygun olan bir sözde rastsal sayı üreteci rastsallık testlerini geçmeye ek olarak bazı ek şifre bilimsel koşulları da sağlamak zorundadır. Bazı şifre bilimsel olarak güvenli sözde rastsal sayı üretici algoritmalar şunlardır:

- Counter (sayıcı) modda veya çıktı besleme modunda çalışan akış veya blok şifreleri.
- Güvenlik kanıtı olan özel tasarımlar. Örn. Blum Blum Shub algoritmasının güçlü bir koşullu güvenlik kanıtı vardır ancak yavaş çalışmaktadır.
- Şifre bilimsel olarak güvenliğe dikkat ederek tasarlanmış özel sözde rastsal sayı üreteçleri. Örn. ISAAC algoritması [1],[2].

2. TASARIM MEKANİZMALARI

Bu bölümde çalışmamızın temelini kapsayan Sözde rastsal Sayı üretiminde kullanılan LFSR (Linear Feedback Shift Registers-Doğrusal Geri Beslemeli Öteleyici Saklayıcı)' ların tasarım ve matematiksel yapıları açıklanacaktır.

LFSR (Linear Feedback Shift Registers-Doğrusal Geri Beslemeli Öteleyici Saklayıcı)' lar birçok anahtar dizisi üretecinde kullanılmaktadır. Bunun nedeni olarak donanımsal uygulamalarda uygunlukları, geniş periyoda sahip olmaları, üretilen serinin iyi istatistiksel özellikler göstermesi ve cebirsel tekniklerle kolayca analiz edilebilmeleri gösterilebilir.

L uzunluğunda bir LFSR (Linear Feedback Shift Register) 0'dan $L-1$'e kadar numaralanmış her biri bir bit depolayabilme yeteneği olan, bir giriş ve bir çıkışa sahip ve verinin hareketini kontrol eden bir saate sahip olan L tane gecikme ünitesi içerir.

LFSR başlangıçta keyfi olarak seçilen L eleman ile yüklenerek başlangıç fazı elde edilir. Geri besleme katsayıları ve başlangıç durumu XOR işlemine sokularak LFSR' ın karakteristik polinomunun elde edilmesi sağlanır [1], [2],[8].

L: LFSR'ın boyu

Başlangıç fazı: $\sigma_0 = a_{-L}, a_{-L+1}, ..., a_{-2}, a_{-1}$

Geri besleme katsayıları:

$c_1, c_2, ..., c_{L-1}, c_L \in Z_2 = \{0,1\}$

Buna göre;

$$a_0 = c_L a_{-L} \oplus c_{L-1} a_{-L+1} \oplus ... c_2 a_{-2} \oplus c_1 a_{-1}$$

$$\Rightarrow \sigma_1 : a_{-L+1}, a_{-L+2}, ..., a_{-1}, a_0$$

$$a_1 = c_L a_{-L+1} \oplus c_{L-1} a_{-L+2} \oplus ... c_2 a_{-1} \oplus c_1 a_0$$

Genel olarak:

$$a_n = c_L a_{n-L} \oplus c_{L-1} a_{n-L+1} \oplus ... c_2 a_{n-2} \oplus c_1 a_{n-1}$$

Bu recursive bağıntının polinomu aynı zamanda LFSR'ın karakteristik polinomudur.

Buna göre karakteristik polinom:

$$m(x) = x^L + c_1 x^{L-1} + c_2 x^{L-2} ... + c_{L\_1} x + c_L$$

LFSR' ın bağlayıcı polinomu:

$$C(D) = 1 + c_1 D + c_2 D^2 + ... + c_{L-1} D^{L-1} + c_L D^L$$

Bağlayıcı polinom ile karakteristik polinom arasındaki bağıntı şu şekildedir: $m(x) = x^L C\left(\frac{1}{x}\right)$

Bir LFSR, boyu L ve bağlayıcı polinomu C(D) ile belirlenir: $LFSR = \langle L, C(D) \rangle$

Örnek:

$$LFSR = \langle 4, C(D) = 1 + D + D^4 \rangle$$

$$c_1 = 1, c_2 = c_3 = 0, c_4 = 1$$

$$f(x_1, x_2, x_3, x_4) = x_1 \oplus x_4$$

Bu LFSR'ı çalıştırmak için başlangıç fazı $\sigma_0 = (0011)$ olarak alınırsa, periyodu 5 olan dizisi üretilir.

x_1	x_2	x_3	x_4	$x_1 \oplus x_4$
0	0	1	1	1
0	1	1	1	1
1	1	1	1	0
1	1	1	0	1
1	1	0	1	0
1	0	1	0	1
0	1	0	1	1
1	0	1	1	0
0	1	1	0	0
1	1	0	0	1
1	0	0	1	0
0	0	1	0	0
0	1	0	0	0
1	0	0	0	1
0	0	0	1	1
0	0	1	1	$= c_0$

LFSR'ın fonksiyonu doğrusal olduğundan $f(0,0,...,0) = 0$ 'dır. Dolayısıyla başlangıç fazı 0 vektörü alınırsa 0 geri beslenir. 0 vektöründen başka faz görülmez. 0'dan farklı bir vektörle başlanırsa 0 vektörü hiç görülmez.

Boyu L olan bir LFSR'ın ürettiği dizinin periyodu en fazla $2^L - 1$ olabilir. Çünkü LFSR'da faz olarak L uzunluğundaki vektörler alınır. L uzunluğunda $2^L - 1$ adet vektör vardır. LFSR'da 0 vektörü görülmezse en fazla $2^L - 1$ adet değişik vektör görülebilir. Yani; LFSR en fazla $2^L - 1$ adım sonra başlangıç noktasına geri döner.

$\langle L, C(D) \rangle$ LFSR'ının ürettiği dizinin periyodu $C(D)$ polinomunun çarpanlarına ayrılabilir olup olmamasıyla ve başlangıç fazıyla ilişkilidir.

- Eğer $C(D)$ çarpanlarına ayrılabiliyorsa üretilen dizinin periyodu başlangıç fazına göre değişir.

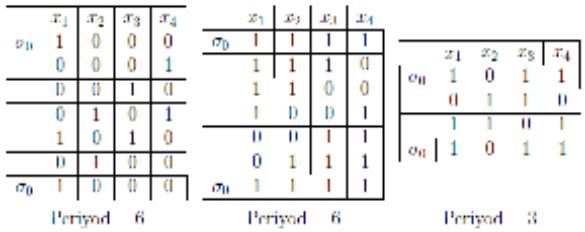
Örnek:

$$\left\langle L, C(D) = 1 + D^2 + D^4 \right\rangle$$

$$c_1 = 0, c_2 = 1, c_3 = 0, c_4 = 1$$

$$f(x_1, x_2, x_3, x_4) = x_2 \oplus x_4$$

$$C(D) = 1 + D^2 + D^4 = \left(1 + D + D^2\right)^2$$



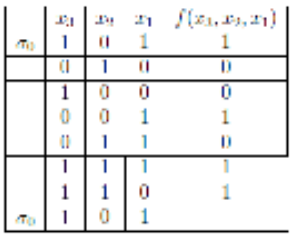
Periyod 6 Periyod 6 Periyod 3

- Maksimum periyotta dizi üretebilmek için $C(D)$ polinomunun çarpanlarına ayrılamaz olması gerekir. Eğer $C(D)$ çarpanlarına ayrılamıyorsa dizinin periyodu başlangıç fazına bağlı değildir ve $C(D)$ polinomunun böldüğü $1 + D^p$ polinomlarından en küçük dereceli olanın derecesi üretilen dizinin periyoduna eşittir. Buradaki p sayısı $2^L - 1$ sayısının bir bölenidir.
- Dizinin periyodunun maksimum yani $2^L - 1$ olması için $C(D)$ polinomunun böldüğü en küçük dereceli polinom $1 + D^{2^L - 1}$ olmalıdır. Bunu sağlayan $C(D)$ polinomuna *ilkel (primitive) polinom* denir [1], [2],[8].

Örnek:

$$\left\langle 3, 1 + D + D^2 \right\rangle$$

$$f(x_3, x_2, x_1) = x_1 \oplus x_1 \cdot x_2 \oplus x_2 \cdot x_3$$



$$z = (1001011)^{\infty}$$

3. SÖZDE-RASTSAL SAYI ÜRETEÇLERİNİN TEST EDİLMESİ

Sözde-Rastsal Sayı Üreteçlerini test etmek için istatistiksel testler kullanılır. Sayı üreteçlerinin rastsallığını kontrol etmek için bir istatistiksel test yeterli değildir. Bu konuda birçok test paketi üretilmiştir (FIBS 140 - Queensland University, DieHard - Florida State University, NIST). NIST paketinden seçilen bazı testlerin açıklaması aşağıda verilmiştir.

3.1. Frekans Testi

Verilen bir dizide bulunan 0 ve 1'lerin oranını kontrol eder. Testin herhangi bir parametresi yoktur. Testte kullanılan referans dağılım yarım normal dağılımdır. Testin sonunda elde edilen p-değerinin çok küçük çıkması dizideki 1' lerin ya da 0' ların sayısının beklenenden fazla olduğunu gösterir. Testin geçerli olabilmesi için dizi uzunluğunun en az 100 bit olması gerekir. Test denklemleri kullanılarak üretilen değer olan p > 0.01 ise dizi rastsal olarak kabul edilir [1], [4], [5].

n: Bit dizisinin boyutu

ε: RNG (Random Number Generator-Rastsal Sayı Üreteci) veya PRNG ile üretilen bit dizisi

$$S_{obs} = \frac{|S_n|}{\sqrt{n}}$$

*erfc:*Hata fonksiyonu $erfc(z) = \frac{2}{\sqrt{\pi}} \int_z^{\infty} e^{-u^2} du$

P-değeri= $erfc\left(\frac{S_{obs}}{\sqrt{2}}\right)$

Örnek:

ε=10110110101101100010101010101011100111000
11011000101000110010011010101001101010010101100110011100110

$n = 100$ $S_{100} = 2$ $S_{obs} = 0.2$

P-değeri = 0.843053 >0.01 olduğundan dizi rastsal kabul edilir.

3.2. Blok Frekans Testi

Verilen bir dizide bulunan 0 ve 1'lerin oranını M bitlik bloklar içinde kontrol eder. Testin tek parametresi blok uzunluğudur (M). Blok uzunluğu 1 olarak alındığında blok frekans testi, frekans testine dönüşür. Her bir bloktaki 1'lerin beklenen oranı $M/2$ 'dir. Testte kullanılan referans dağılımı ki-kare dağılımıdır. Testin sonunda elde edilen p-değeri çok küçük çıkması, dizideki bloklarda 1'lerin ve 0'ların oranının 1/1'den fazlasıyla saptığını gösterir. Testin geçerli olabilmesi için blok uzunluğunun en az 20 bit, dizi uzunluğunun da en az 100 bit olması gerekir [1], [5], [7].

$$\pi_i = \frac{\sum_{j=1}^{M} \varepsilon_{(i-1)M+j}}{M}$$

$$\chi^2(obs) = 4M\sum_{i=1}^{N}\left(\pi_i - \frac{1}{2}\right)^2 \qquad N = \left\lfloor \frac{n}{M} \right\rfloor$$

P-değeri = *igamc* (gamma fonksiyonu) $\left(\frac{N}{2}, \frac{\chi^2}{2}\right)$

Örnek:

ε=1011011010110110001010101010101110011100011011000101000110010011010101001101010010101100110011100110

$\pi_1 = \frac{3}{5}$, $\pi_2 = \frac{1}{2}$, $\pi_3 = \frac{1}{2}$, $\pi_4 = \frac{3}{5}$, $\pi_5 = \frac{1}{2}$,

$\pi_6 = \frac{2}{5}$, $\pi_7 = \frac{1}{2}$, $\pi_8 = \frac{2}{5}$, $\pi_9 = \frac{3}{5}$, $\pi_{10} = \frac{1}{2}$

$\chi^2(obs) = 2$

$$P - value = igamc\left(\frac{N}{2}, \frac{\chi^2}{2}\right)$$

$P - value = igamc(5,1) = 0.99634015$

P-değeri $\geq$ *0.01* olduğundan dizi rastsal kabul edilir.

3.3. Akış Testi (Runs Test)

Bu test bit dizisindeki akışların toplam sayısıyla ilgilidir. Akış ardışık aynı bit sıralamasını ifade eder. Böylece 0'lar ve 1'ler arasındaki dalgalanmaların kontrolü sağlanarak üretilen bit dizisinin yavaş ya da hızlı olabileceği konusunda fikir verir [2], [5].

$V_n(obs)$ *:* Akışların sayısı

π *:*Bit dizisindeki 1'lerin sayısı

$$\pi = \frac{\sum j\varepsilon_j}{n} \qquad \left|\pi - \frac{1}{2}\right| \geq \tau \qquad \tau = 2/\sqrt{n}$$

$$V_n(obs) = \sum_{k=1}^{n-1} r(k) + 1$$

$\varepsilon_k = \varepsilon_{k+1}$ ise $r(k) = 0$ diğer durumlarda $r(k) = 1$ olarak kabul edilir.

$$P - değ. = erfc\left(\frac{\left|V_n(obs) - 2n\pi(1-\pi)\right|}{2\sqrt{2n}\pi(1-\pi)}\right)$$

Örnek:

ε=1011011010110110001010101010101110011100011011000101000110010011010101001101010010101100110011100110

$\pi = 0.51$ $\tau = 0.63246$

$r(1)=1$	$r(2)=1$	$r(3)=0$	$r(4)=1$	$r(5)=1$
$r(6)=0$	$r(7)=1$	$r(8)=1$	$r(9)=1$	$r(10)=1$
$r(11)=0$	$r(12)=1$	$r(13)=1$	$r(14)=0$	$r(15)=1$
$r(16)=0$	$r(17)=0$	$r(18)=1$	$r(19)=1$	$r(20)=1$
$r(21)=1$	$r(22)=1$	$r(23)=1$	$r(24)=1$	$r(25)=1$
$r(26)=1$	$r(27)=1$	$r(28)=1$	$r(29)=1$	$r(30)=1$
$r(31)=0$	$r(32)=0$	$r(33)=1$	$r(34)=0$	$r(35)=1$
$r(36)=0$	$r(37)=0$	$r(38)=1$	$r(39)=0$	$r(40)=0$
$r(41)=1$	$r(42)=0$	$r(43)=1$	$r(44)=1$	$r(45)=0$
$r(46)=1$	$r(47)=0$	$r(48)=0$	$r(49)=1$	$r(50)=1$
$r(51)=1$	$r(52)=1$	$r(53)=0$	$r(54)=0$	$r(55)=1$
$r(56)=0$	$r(57)=1$	$r(58)=0$	$r(59)=1$	$r(60)=1$
$r(61)=0$	$r(62)=1$	$r(63)=0$	$r(64)=1$	$r(65)=1$
$r(66)=1$	$r(67)=1$	$r(68)=1$	$r(69)=1$	$r(70)=1$
$r(71)=0$	$r(72)=1$	$r(73)=0$	$r(74)=1$	$r(75)=1$
$r(76)=1$	$r(77)=1$	$r(78)=1$	$r(79)=0$	$r(80)=1$
$r(81)=1$	$r(82)=1$	$r(83)=1$	$r(84)=1$	$r(85)=0$
$r(86)=1$	$r(87)=0$	$r(88)=1$	$r(89)=0$	$r(90)=1$
$r(91)=0$	$r(92)=1$	$r(93)=0$	$r(94)=0$	$r(95)=1$
$r(96)=0$	$r(97)=1$	$r(98)=0$	$r(99)=1$	

$V_n(obs) = 0.420184$

P-değeri $\geq$ *0.01*, olduğundan dizi rastsal kabul edilir.

3.4. Bloktaki En Uzun Birler Testi

Test, M-bitlik bloklarda bulunan en uzun birler grubu üzerinde odaklaşır. Testin tek parametresi blok uzunluğudur (M). Dizi M-bitlik n tane bloğa bölünür ve her blok içerisindeki en uzun birler öbeğinin uzunluğuna bakılır. Bu değerlerin frekansları beklenen değerlerle kıyaslanır ve ciddi bir sapma olup olmadığı kontrol edilir. Testte kullanılan referans dağılım ki-kare dağılımıdır. Dizi uzunluğuna göre blok uzunluğu ve blok sayısına karar verilir [2], [5].

Minimum *n*	***M***
128	8
6272	128
750000	10^4

v_i	***M=8***	***M=128***	***M=10⁴***
v_0	≤ 1	≤ 4	≤ 10
v_1	2	5	11
v_2	3	6	12
v_3	≥ 4	7	13
v_4		8	14
v_5		≥ 9	15
v_6			≥ 16

$$P(v \leq m) = \sum_{r=0}^{M}\binom{M}{r} P(v \leq m|r)\frac{1}{2^M}$$

$$\chi^2(obs) = \sum_{i=0}^{K}\frac{(v_i - N\pi_i)^2}{N\pi_i}$$

Tablo1. K ve M değerlerinin aldığı değere göre hesaplanan olasılık tablosu

K=5 , M=128

Sınıflar	$v \le 1$	$v = 2$	$v = 3$	$v \ge 4$
Olasılıklar	$\pi_0 = 0{,}2148$	$\pi_1 = 0{,}3672$	$\pi_2 = 0{,}2305$	$\pi_3 = 0{,}1875$

K=5 , M=128

Sınıflar	$v \le 4$	$v = 5$	$v = 6$	$v = 7$	$v = 8$	$v \ge 9$
Olasılıklar	$\pi_0 = 0{,}1174$	$\pi_1 = 0{,}2430$	$\pi_2 = 0{,}2493$	$\pi_3 = 0{,}1752$	$\pi_4 = 0{,}1027$	$\pi_5 = 0{,}1124$

K=5 , M=512

Sınıflar	$v \le 6$	$v = 7$	$v = 8$	$v = 9$	$v = 10$	$v \ge 11$
Olasılıklar	$\pi_0 = 0{,}1174$	$\pi_1 = 0{,}2460$	$\pi_2 = 0{,}2523$	$\pi_3 = 0{,}1755$	$\pi_4 = 0{,}1015$	$\pi_5 = 0{,}1077$

K=5 , M=1000

Sınıflar	$v \le 7$	$v = 8$	$v = 9$	$v = 10$	$v = 11$	$v \ge 12$
Olasılıklar	$\pi_0 = 0{,}1307$	$\pi_1 = 0{,}2437$	$\pi_2 = 0{,}2452$	$\pi_3 = 0{,}1714$	$\pi_4 = 0{,}1002$	$\pi_5 = 0{,}1088$

K=6 , M=10000

Sınıflar	$v \le 10$	$v = 11$	$v = 12$	$v = 13$	$v = 14$	$v = 15$	$v \ge 16$
Olasılıklar	$\pi_0 = 0{,}0882$	$\pi_1 = 0{,}2092$	$\pi_2 = 0{,}2483$	$\pi_3 = 0{,}1933$	$\pi_4 = 0{,}1208$	$\pi_5 = 0{,}0675$	$\pi_5 = 0{,}0727$

Çalışmamızda kullandığımız K ve M değerleri için ilgili tablo şu şekildedir:

M	K	N
8	3	16
128	5	49
10^4	6	75

$$P - değ. = igamc\left(\frac{K}{2}, \frac{\chi^2(obs)}{2}\right)$$

Örnek:

$K = 3$ ve $M = 8$

ε = 1100110000010101011011000100111001111000000
0000010010011010101000100010011111010110100000
0011010111110011001110011011011000010110010

n = 128

Alt Blok	Max-Run	Alt Blok	Max-Run
11001100	(2)	00010101	(1)
01101100	(2)	01001100	(2)
11100000	(3)	00000010	(1)
01001101	(2)	01010001	(1)
00010011	(2)	11010110	(2)
10000000	(1)	11010111	(3)
11001100	(2)	11100110	(3)
11011000	(2)	10110010	(2)

$v_0 = 4 , v_1 = 9 , v_2 = 3 , v_3 = 0 \quad \chi^2 = 4.882457$

P-değeri = 0.180609 ≥ 0.01 olduğundan dizi rastsal kabul edilir.

3.5. Dizi Testi

Bu test bit dizisindeki örtüşen m-bitlik öbekleri test eder. $\nabla\psi_m^2(obs)$ ve $\nabla^2\psi_m^2(obs)$ değerleri m-bitlik öbeklerin frekansını hesaplamak için kullanılır [2], [5], [6].

$$\psi_m^2 = \frac{2^m}{n}\sum_{i_1\ldots i_m} v_{i_i\ldots i_m}^2 - n$$

$$\psi_{m-1}^2 = \frac{2^{m-1}}{n}\sum_{i_1\ldots i_{m-1}} v_{i_i\ldots i_{m-1}}^2 - n$$

$$\psi_{m-2}^2 = \frac{2^{m-2}}{n}\sum_{i_1\ldots i_{m-2}} v_{i_i\ldots i_{m-2}}^2 - n$$

$$\nabla\psi_m^2 = \psi_m^2 - \psi_{m-1}^2 \qquad \nabla^2\psi_m^2 = \psi_m^2 - 2\psi_{m-1}^2 + \psi_{m-2}^2$$

$$P - değ.1 = igamc\left(2^{m-2}, \nabla\psi_m^2\right)$$
$$P - değ.2 = igamc\left(2^{m-3}, \nabla^2\psi_m^2\right)$$

Örnek:

ε=10110110101101100010101010101011100111000 1
1011000101000110010011010101001101010010101100 1 10011100110

$$v_{000}=4 \quad v_{001}=12 \quad v_{010}=18 \quad v_{011}=15$$
$$v_{100}=12 \quad v_{101}=20 \quad v_{110}=15 \quad v_{111}=2$$

$$v_{00}=16 \quad v_{01}=32 \quad v_{10}=33 \quad v_{11}=18$$

$$v_0=49 \quad v_1=51$$

$$\psi_3^2=\frac{2^3}{100}(16+144+324+225+144+400+225+4)-100=18.56$$

$$\psi_2^2=\frac{2^2}{100}(256+1024+1089+324)-100=7.72$$

$$\psi_1^2=\frac{2^1}{100}(2401+2601)-100=0.04$$

$$\nabla\psi_3^2=10.84$$
$$\nabla^2\psi_3^2=3.16$$

$$P-de\breve{g}.1=igamc(2,10.84)=0.0002319$$
$$P-de\breve{g}.2=igamc(1,3.16)=0.04242574$$

$P-de\breve{g}.1<0.01$ olduğundan dizi rastsal kabul edilmez.
$P-de\breve{g}.2>0.01$ olduğundan dizi rastsal kabul edilir.

4. SONUÇ

Bu çalışmada sözde rastsal sayı üretiminin LFSR' lar kullanılarak elde edilmesi ve üretilen sayı dizilerinin test edilmesi üzerinde durulmuştur. Sözde rastsal sayılar rastsal sayılar üzerindeki tartışmalara alternatif olarak geliştirilen ve şifrelemede halen önemini korumaya devam eden yapılardır. Çalışmamızda sunduğumuz örnekler ve tasarım mekanizmaları sözde rastsallığa yapılan yaklaşımlardır.
Gerçekleştirmiş olduğumuz çalışma temel analizler ve tasarım mekanizmaları üzerinde daha önce gerçekleştirilen yaklaşımlarla paralellik içermektedir. Bu bilgiler ışığında kriptografik uygulamalarda kullanılmak üzere sözde rastsal sayı üretimi için yeni tasarım mekanizmaları ve yaklaşımlar geliştirilmeye devam edilecektir.

KAYNAKLAR

[1] A. Menezes, P. v. Oorschot, and S. Vanstone, *Handbook of Applied Cryptography*, CRC Press, 1997.
[2] Uygulamalı Matematik Enstitüsü Kriptografi Bölümü, "*Kriptografiye Giriş Ders Notları*", ODTÜ, 2004
[3] Anne Canteaut, *Stream Ciphers*, Encyclopedia of Cryptography and Security, 2005.
[4] Kai Lai Chung, "*Elementary Probability Theory with Stochastic Processes*", New York: Springer-Verlag, 1979
[5] Revised NIST Special Publication 800-22, "*A Statistical Test Suite for the Validation of Random Number Generators and Pseudo Random Number Generators for Cryptographic Applications"*, 2001.
[6] I. J. Good, "*The serial test for sampling numbers and other tests for randomness*," Proc. Cambridge Philos. Soc. 47, pp. 276-284, 1953.
[7] Nick Maclaren, "*Cryptographic Pseudo-random Numbers in Simulation*", Cambridge Security Workshop on Fast Software Encryption. Dec. 1993.
[8] P. Ekdahl, *"On LFSR Based Stream Ciphers"*, PHd Thesis, November 2003.
[9] www.wikipedia.org