İSTANBUL TEKNİK ÜNİVERSİTESİ ★ BİLİŞİM ENSTİTÜSÜ

SÜPERELİPTİK EĞRİLER İLE ÇARPANLARA AYIRMA YÖNTEMİ

YÜKSEK LİSANS TEZİ

Kübra Nari

Bilişim Uygulamaları Anabilim Dalı

Bilgi Güvenliği Mühendisliği ve Kriptografi Programı

ARALIK 2018

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ BİLİŞİM ENSTİTÜSÜ

SÜPERELİPTİK EĞRİLER İLE ÇARPANLARA AYIRMA YÖNTEMİ

YÜKSEK LİSANS TEZİ

Kübra Nari
(707151014)

Bilişim Uygulamaları Anabilim Dalı

Bilgi Güvenliği Mühendisliği ve Kriptografi Programı

Tez Danışmanı: Doç. Dr. Enver Özdemir

ARALIK 2018

İTÜ, Bilişim Enstitüsü'nün 707151014 numaralı Yüksek Lisans öğrencisi Kübra Nari, ilgili yönetmeliklerin belirlediği gerekli tüm şartları yerine getirdikten sonra hazırladığı "SÜPERELİPTİK EĞRİLER İLE ÇARPANLARA AYIRMA YÖNTEMİ" başlıklı tezini aşağıda imzaları olan jüri önünde başarı ile sunmuştur.

Tez Danışmanı :	**Doç. Dr. Enver Özdemir** İstanbul Teknik Üniversitesi	.....
Jüri Üyeleri :	**Doç. Dr. Ergün Yaraneri** İstanbul Teknik Üniversitesi	.....
	Dr. Öğr. Üyesi Elif Segah Öztaş Karamanoğlu Mehmetbey Üniversitesi	.....

Teslim Tarihi : **16 Kasım 2018**
Savunma Tarihi : **14 Aralık 2018**

Aileme,

ÖNSÖZ

Öncelikle tez konumu seçme, araştırma ve geliştirme aşamasında bana her türlü desteği sağlayan bilgi ve tecrübelerini benimle paylaşan tez danışmanım Doç. Dr. Enver Özdemir'e teşekkür ederim. Bununla birlikte, her zaman beni motive eden ve destekleyen arkadaşlarıma ve ayrıca eğitim - öğretim hayatım boyunca benden hiçbir maddi - manevi desteği esirgemeyen, her türlü başarımın ardında bulunan başta babam Ramazan Nari ve sevgili annem Dilek Nari olmak üzere, tüm aileme sonsuz teşekkürlerimi sunarım.

Aralık 2018

Kübra Nari
Yazılım Geliştirme Uzmanı

İÇİNDEKİLER

Sayfa

KISALTMALAR

SQUFOF **:** Kare Formlar ile Çarpanlara Ayırma (Square Forms Factorization)

CFRAC **:** Devamlı Kesir Yöntemi (Continued Fractions)

NFS **:** Sayı Alan Kalburu **(**Number Field Sieve)

QS **:** Kuadratik Eleme (Quadratic Sieve)

HPC **:** Yüksek Başarımlı Hesaplama (High Performance Computing)

ÇİZELGE LİSTESİ

Sayfa

ŞEKİL LİSTESİ

Sayfa

SÜPERELİPTİK EĞRİLER İLE ÇARPANLARA AYIRMA

ÖZET

Günümüzde yaşanan teknolojik gelişmeler sonucunda gerek günlük hayatta gerekse profesyonel iş hayatında kullanılan tüm sistemler internet ve makinalara bağımlı hale gelmiştir. Her türlü bankacılık, hastane randevu, tahlil görüntüleme, üniversite kayıt vb. gibi işlemleri artık internet aracılığı ile gerçekleştiriyoruz. İnternet hayatımızı bu kadar kolaylaştırıp bize zaman kazandırırken, aynı zamanda güvenlik sorununu da gün yüzüne çıkartmaktadır.

İnternet üzerinde iletişimi sağlamak için belirli adımlar bulunmaktadır. Bir ağ üzerindeki iki makinanın haberleşmesi için en temel olarak birbirlerinini tanımaları gerekmektedir. Bu tanıma işleminin de güvenli bir şekilde gerçekleştirilmelir. Kimlik doğrulama işleminin (authentication) güvenli bir şekilde gerçekleştirilmesinden sonra iletişime devam edilebilir ve yapılmak istenen ilgili işlemler tamamlanabilir. Ancak kimlik doğrulama aşaması internet üzerinde güvenli iletişim için ilk ve temel adımdır. Bu aşamanın güvenli bir şekilde kontrol edilmemesi durumunda kötü niyetli kişilere davetiye çıkartılmış olur. Günümüzde bir çok veri kaybı yaşanmasına ve kişisel bilgilerin çalınmasına sebep olarak güvenilir olmayan web sitelerde önemli bilgilerin paylaşılmasıdır. Bu tarz saldırı ve kayıpların önüne geçmek adına güçlü kriptografik algoritmalar ile güvenlik sağlanmalıdır.

Hali hazırda internet tarayıcılarının güvenliği sağlamak adına kullandığı algoritmalar bulunmaktadır. Bu algoritmalardan en çok kulanılanı RSA algoritmasınıdır. RSA şifreleme tekniği günümüzde özellikle internet tarayıcılarının SSL protokollerinde kullanılan bir tekniktir ve bu algoritmanın zorluğu temelde bir tam sayının çarpanlarına ayrılmasının zorluğuna dayanmaktadır.

Çarpanlara ayırma konusu yüzyıllardır merak konusu olmuştur ve bir çok kişi tarafından bu alanda çalışmalar yapılmıştır ve halen devam etmektedir. Pek çok algoritma geliştirilmiş olmasına rağmen günümüz şartları gözönüne alındığında bu algoritmalar RSA'de kullanılan sayılar için yetersiz kalmaktadır. RSA'in en az 1024

bitlik anahtarlar kullandığı düşünülürse, bu denli büyük bir sayının çarpanlarına ayrılması için bilgisayarların ve özellikle yüksek başarımlı hesaplama sistemlerinin devreye girmesi zorunlu bir hal almıştır.

Bu çalışmada çeşitli çarpanlara ayırma algoritmalarını inceleyerek temelde Lenstra'nın eliptik eğriler ile çarpanlara ayırma yöntemi açıklanacaktır. Sonrasında bu yöntemin süpereliptik eğriler kullanılarak geliştirdiğimiz versiyonu anlatılacaktır. Geliştirdiğimiz yöntemin avantaj ve dezavantajları tartışılacaktır. Bu tezdeki algoritma tamamen paralelleşebilen bir yapıya sahip olduğundan HPC sistemlerine kolayca adapte edilebilir ve verimlilik sağlanabilecektir. Yapılan çalışmaların kodlama aşamasında PARI kütüphanesinden faydalanılmış ve C++ programlama dili kullanılmıştır.

INTEGER FACTRIZATION METHOD WITH SUPERELLIPTIC CURVES

SUMMARY

As a result of technological developments in this digital age, all systems that are used in professional bussines life and daily life have become dependent on internet. Almost all kind of operations in our daily life such as banking, hospital appointments, university registration is carried out via internet. Although internet makes our life so easy and saves our time, it also brings out the security problem.

There are some certain steps to ensure communication on the Internet. Basically, two machines on a network need to know each other for communication. This recognition process should also be performed safely. After the authentication step is carried out safely, the communication can be continued and related operations can be completed. As long as authentication step is not controlled and completed securely, the system will be vulnerable against adversaries. Therefore, if important informations are shared through suspicious and probably unsecure network systems, a large amount of data loss can be occured and personel information can be stolen. So, security should be ensured with strong cryptographic algorithms to prevent such attacks and data losses.

Currently, there are algorithms used by internet browsers to provide security. RSA algorithm is the most commonly used algorithm for especially authentication by the browsers. For example RSA method is used in transportation protocol with SSL. The vulnerability of this algorithm mainly depends on hardness of factorization of large integers.

Integer factorization has been popular topic for centuries and many people have been worked on this field and also many are still working. Although many algorithms have been developed, these algorithms are weak considering today's conditions. Considering that RSA uses at least 1024-bit keys, computers and especially high-performance computing systems have to be become a part of this research area in order to factor out such a large number.

In this work, many different methods about factorization have been investigated and Lenstra 's elliptic curves method have been explained. After this general explanations, a factorization algorithm has been presented. The algorithm is similar to Lenstra’s method but it mainly uses super elliptic curves. The elements of super elliptic curve come from jacobian groups of curve and they are named as ideal. As we know from Lenstra’s method, to find a factor we have to compute a certain power of chosen ideal. If there exist any difference between the elements of ideal when trying to compute, a factor is found. In the superelliptic curves, the number of ideal elements increase in direct proportion to the degree of curve. For example, there exist 6 polynomials in the ideal of 3th degree curves and 15 in the ideal of 5th degree curves. In the hyperelliptic curve method, ideals occur the pairs, so the chance to catch a degree difference is lower than superelliptic curves. Hence, superelliptic curves are investigated and used in this work to increase the possibility of detection a factor.

Then, advantages and disadvantages of our algorithm, which employes supereliptic curves as an improved version of Lenstra's method, has been explained. C ++ programming language with PARI library have been used in our experiments. In addition, the improved algorithm is embarrasingly parallel, so it can be easily adapted to the HPC environments.

1. GİRİŞ

Günümüzde internet kullanımının her alanda yaygınlaşmasıyla birlikte güvenlik problemlerini de su yüzüne çıkmaktadır. Sanal alemde güvenliğin sağlanması ve gizliliğin korunması gerek kişisel gerekse kamusal alanlarda önem verilmesi gereken hususlardır. Güvenliği ve gizliliği sağlamak için kullanılan çeşitli yöntemler bulunmaktadır ve bu kriptografik yöntemler matematiksel temellere dayanmaktadır. En yaygın olarak kullanılanı RSA açık anahtarlı şifreleme sistemidir. Bu sistem p ve q asal sayı olmak üzere $n = pq$ şeklinde iki asal sayının çarpımından oluşan anahtarlara sahiptir. Bu anahtarlar en az 1024 bit (309 basamaklı) olarak seçilmektedir. Bu durumda bu kadar büyük sayıların çarpanlarına ayrılması kolay değildir. RSA kripto sisteminin güvenliği de bu büyük sayıların çarpanlarına ayrılamamasından kaynaklanmaktadır. Pek çok RSA anahtarı çarpanlarına ayrılmış olsa da günümüzde kullanılan anahtarların çarpanları henüz tespit edilmemiştir.

Çarpanlara ayırma konusu yıllar boyu merak konusu olmuştur ancak 1970'lerde bilgisayarların devreye girmesi ve 1990'lar internetin yaygınlaşmasıyla daha çok önem kazanmıştır. Özellikle günümüzde tüm kişisel verilerimizi paylaştığımız internet ortamlarında her türlü bilgi aktarımının güvenliğinin sağlanması oldukça önemlidir.

Kötü niyetli kişilerce bilgiler ele geçirebileceği gibi bilgiler erişilemez hale getirilebilir yada maddi zarara eden olabilecek kayıplar yaşanabilir. Ayrıca büyük kurumlar için sistemlerin işleyiş ve güvenliğindeki en ufak bir aksama ciddi maddi kayıplara ve itibar zedelenmelerine yol açabilir. Bu nedenle gerek kişisel bilgisayarların gerekse büyük kurumların sistemlerinin güvenliğini sağlamak için güçlü kriptografik sistemler kullanılmalıdır.

Bu çalışmada da makineler arası iletişimde kimlik doğrulama ve başka pek çok amaç için kullanılan RSA metoduna yönelik atak niteliğinde bir çarpanlara ayırma algoritması geliştirilmiştir. Lenstra'nın eliptik eğri yöntemi temel alınarak geliştirilen yöntem bazı olasılıksal avantajlar taşımaktadır. Ayrıca algoritma yapısı gereği YBH ortamlarına uygun bir şekilde kodlanabilecek ve algoritmanın verimliliğini artırılabilecektir.

2. RSA AÇIK ANAHTARLI ŞİFRELEME SİSTEMİ

RSA sistemi Ron L. Rivest, Adi Shamir, ve Len Adleman tarafından 1978 yılında yayınlanan bir açık anahtarlı şifreleme sistemidir [1]. Günümüzde bilgisayar sistemlerinde güvenliği sağlamak amacıyla pek çok alanda kullanılmaktadır(Kimlik doğrulama, imzalama algoritmaları vb.). RSA sisteminin temelleri basit bir şekilde bir tamsayının çarpanlarına ayrılmasının zorluğuna dayanmaktadır. RSA'nın anahtarları ikililerden oluşmaktadır ve bunlardan biri iki sayının çarpımından oluşur. RSA algoritmasında anahtarların oluşturulması işlemi aşağıda ki adımları içermektedir:

(1) p ve q aynı bit uzunlukta olmak ve belirli özellikleri[2] sağlamak üzere iki büyük asal sayı olarak seçilir.
(2) $n = pq$ ve $\varphi(n) = (p-1)(q-1)$ hesaplanır.
(3) $1 < e < \varphi(n)$ koşuluna uyacak bir e tamsayısı seçilir. $ebob(e, \varphi(n)) = 1$ olmalıdır.
(4) $ed \equiv 1 \, mod \, \varphi(n)$ denkliğinden d hesaplanır. $(1 < d < \varphi(n))$
(5) (n, e) ikilisi açık anahtar olarak belirlenirken (n, d) ikilisi özel anahtar olarak belirlenir.

Göndericinin (n, e) açık anahtarı kullanılarak mesaj şifrelenir ve şifreli mesaj ancak alıcının özel anahtarı ile okunabilir. Mesaj şifreleme ve deşifre etme işlemleri ise aşağıdaki şekilde gerçekleşmektedir.

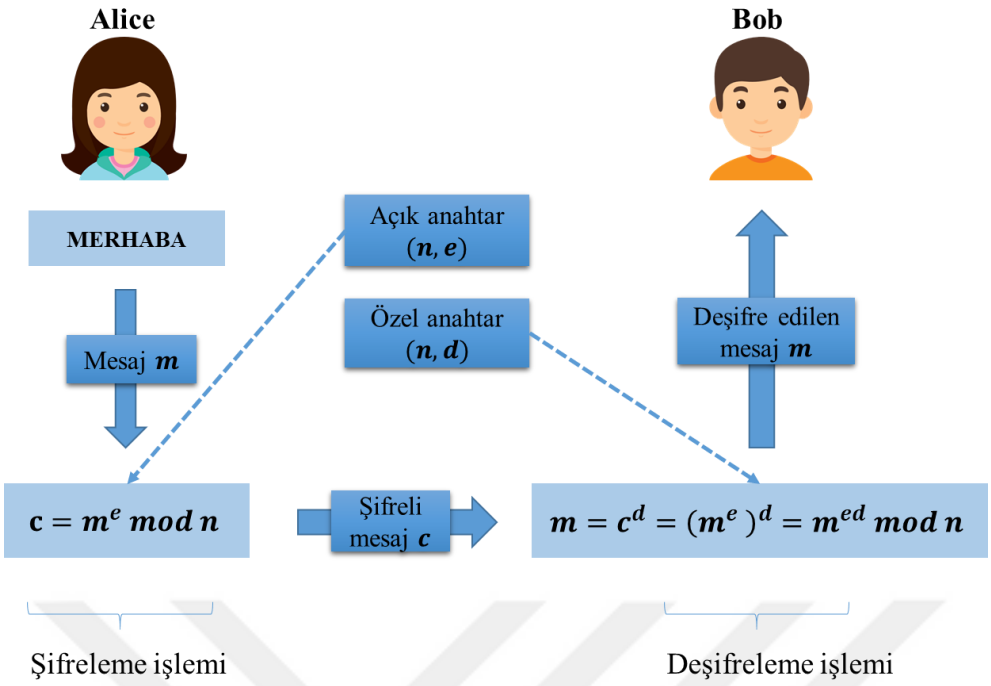
(1) m gönderilecek olan mesaj metni olmak üzere şifreli metin

$$c \equiv m^e \, mod \, n$$

olarak hesaplanır.
(2) Alıcı özel anahtarı (n, d) ile

$$m \equiv c^d \equiv m^{ed} \, mod \, n$$

değerini hesaplayarak mesajı deşifre eder ve m metnini elde eder.



Şekil 2.1 : RSA açık anahtarlı şifreleme sistemi çalışma şeması

Günümüzde literatürde RSA için kullanılabilecek sayılar arasında en çok 232 basamaklı sayılar çarpanlarına ayrılmıştır. En günceli ise 15 Ağustos 2018 tarihinde 230 basamağın çarpanlarına ayrılmasıdır [3]. Şuan ki sistemlerde kullanılan anahtar n'yi oluşturan p ve q değerleri en az 1024 bitlik asal sayılardan oluşmaktadır ve günümüzde 2048 bitlik anahtarlar da kullanılmaktadır. Bu durumda $n = pq$ değerinin çarpanlara ayrılması için işlem gücü gerektirmektedir. Örneğin; 232 basamaklı RSA anahtarının NFS yöntemi ile çarpanlarına ayrılması için kullanılan işlem gücü zamanın kişisel bir bilgisayarın 2000 senelik çalışmasına denktir [3]. Ancak YBH kullanımı ile bu süre makul süreçlere kısaltılmıştır.

3. ÇARPANLARA AYIRMA ALGORTİMALARI

3.1 Giriş

Euclid'in asal sayıların tanımını yapması ve çarpanlara ayırma fikrini ortaya atmasıyla çarpanlara ayırma kavramı aritmetiğin temel ilgi alanlarından biri haline gelmiştir. 1970'lere kadar tam sayıların iş hayatında veya gündelik hayatta çok yer almamasından kaynaklı olarak bu ilgi yalnızca teorik olarak kalmıştır. Ancak 1970'lerin sonlarına doğru bu teorik ilgi yerini güvenlikle ilgili kavramlara bırakmıştır. Fermat döneminden önce Bölenleri Deneme Yöntemi (Trial Division) dışında bilinen bir çarpanlara ayırma yöntemi bulunmamaktadır. Bu sebeple çarpanlara ayırma tarihinin Fermat ile başladığını kabul etmek mümkündür.

1643'te Fermat tam sayıların çarpanlara ayrılması konusunda günümüze dek ışık tutacak bir fikir ortaya atmıştır. Fermat'a göre herhangi bir sayı n iki tamsayının kareleri farkı şeklinde ifade edilebilir. Yani, öyle iki x ve y sayılar bulunur ki; $n = x^2 - y^2$ eşitliğini sağlar. Bu durumda $(x + y)$ ve $(x - y)$ sayıları n'nin birer çarpanı olacaktır. Fermat'ın yöntemi n'nin çarpanlarının küçük ve birbirine çok yakın olması durumunda çok hızlı çalışmaktadır. Ancak aksi halde; pek verimli bir yöntem sayılmaz.

1750'li yıllara gelindiğinde şüphesiz en üretken matematikçi olan Euler'in çarpanlara ayırma konusunda da fikirleri vardı [4]. Euler sadece belirli formlardaki tam sayılar ile ilgilendi. Yöntemlerinden biri aynı D değeri için iki farklı şekilde yazılabilen $n = a^2 + Db^2$ formundaki sayılar üzerinde etkilidir. Yöntem n'nin çarpanlarını ayırmak için Fermat'ın yöntemi şeklinde ilerler. Ayrıca Euler bu metodu bazı büyük sayıların çarpanlarını bulmak içinde kullanmıştır.

1798 yılında Legendre'nin uyumlu karaler teorisi ile (congruent squares) çarpanlara ayırma konusunda farklılık sunan bir fikir ortaya atılmış oldu. Bu yöntem tüm modern çarpanlara ayırma algoritmalarının temelini oluşturacaktır. Bu yıllarda hesaplama

gücününün yoksunluğu bu algoritmayı belirli sayılarla kısıtlı tutsa da bilgisayar gücü kullanılmaya başladığında en verimi algoritmalardan biri haline gelecektir.

Legendre kendi fikrini yayınladığı sırada Gauss'da sayılar teorisindeki en önemli çalışması olan Disquisitiones Arithmeticae'yı 1801'de yayınladı [4]. Bu kitap içerisinde tamsayıların çarpanlara ayrılması ile ilgili fikirler ve metodlar barındırmaktadır. Gauss'un metodu $mod\ n$'e ait kuadratik rezidüleri bularak bunların arasından çarpan olabilecek asal sayıları elemek üzere çalışır. Bu yöntem tüm modern çarpanlara ayırma algoritmalarında eleme yöntemlerinin önemli bir elemanını oluşturacaktır.

Gauss ve Legendre'den sonra ki yıllarda bu alanda yeni fikirlerin ortaya çıktığı söylenemez. Bu iki yöntemle bile 10-15 basamaktan daha fazla büyüklükte olan sayıların çarpanları bulunamıyordu. Ayrıca 10-15 basamaklı bir sayının çarpanlara ayrılması işlemi bile sayfalarca kağıt, mürekkep ile günlerce süren işlemler ile ancak hesaplanabiliyordu. Bu sebeplerden ötürü, bu alandaki çalışmalar eleme işlemlerini yapacak bir makina icat edilene kadar bi süreliğine askıda kalmış oldu [4].

1900'lü yıllarda birbirinden bağımsız olarak bir kaç kişi tarafından eleme temelli algoritmaların işlemlerininin gerçekleştirileceği çeşitli makinalar icat edildi. Bunların arasında en başarılları Lehmer ve Kraitchik'in geliştirdiğini makinalar olmuştur. Bu yıllarda en hızlı çarpanlara ayırma yöntemi olarak kabul edilmiştir.

1970'li yıllarda bilgisayarların işlem gücünün devreye girmesiyle birlikte buna uygun algoritmalar üzerinde çalışılmaya başlanmıştır. Daniel Shanks kuadratik formları kullanarak SQUFOF algoritması [5,6] üzerinde çalışmış ve farklı versiyonları yıllarca kullanılmıştır. SQUFOF algoritması Gauss'un ortaya koymuş olduğu ikili ikinci dereceden kuadratik formlardan yararlanır. x ve y tam sayı olmak üzere ikili ikinci dereceden formlar $f(x,y) = ax^2 + bxy + cy^2$ şeklinde yazılan fonksiyonlardır. Genelde bir form (a,b,c) şeklinde gösterilir ve diskriminantlarıyla ayırt edilir öyle ki; diskriminant $D = b^2 - 4ac$ şeklinde hesaplanır. Aynı diskriminanta sahip formlar arasında denklik bağlantısı vardır. Bu denklik bağlantısının yanında, denklik sınıfları için Gauss bir grup işlemi tanımlamıştır. Her bir denklik sınıfını tanımlayan eşsiz bir eleman vardır ve buna indirgenmiş form denir. İndirgenmiş formlar içerisinde

mertebesi 2 olan form (p, kp, c) şeklinde olmak zorundadır. (p, kp, c) şeklindeki bir formun ilk elemanı p'nin diskriminan değerini böldüğü açıkça görülmektedir. Dolayısıyla ikili ikinci dereceden diskriminantı D olan formların içerisinde mertebesi 2 olan bir eleman bulunduğunda D sayısının bir çarpanı elde edilmiş olur. Shanks'in SQUFOF algoritması bu fikrin üzerinde inşa edilmiştir.

John Pollars 1974 yılında $p-1$ yöntemini ve sonrasında p yöntemini geliştirmiştir. Morrison ve Brillhart temellerinin Legendre'nin atmış olduğu CFRAC algoritmasını ilk genel amaçlı çarpanlara ayırma algoritması olarak geliştirmişlerdir.1980'li yıllara gelindiğinde Brent p metodunu iyileştirerek daha verimli bir hale getirecek ve Carl Pomerance kuadratik eleme algoritması ile 71 basamaklı sayıların çarpanlara ayrılmasında başarıya ulaşacaktır[4].

Lenstra 1987 yılında tamamiyle farklı bir yol izleyerek çarpanlara ayırmada yeni bir yöntem geliştirmiştir. Bu yöntemde eliptik eğrileri kullananılmış olup, küçük çarpanlara sahip tamsayıları çarpanlara ayırlamayı hedefleyen bir metotdur. Daha sonraki yıllarda ise NFS olarak bilinen Sayı Alan Kalburu yöntemi gerçeklenecektir[7].
Aşağıda bazı çarpanlara ayırma algoritmaları detaylandırılmıştır.

3.2 Bölenleri Deneme Yöntemi(Trial Division)

Bu yöntem en basit çarpanlara ayırma yöntemidir. $S = \{p_1, p_2, p_3, \ldots., p_n\}$ asal sayı kümesi $\sqrt{n}$'den küçük tüm asal sayıları içeriyor olsun. S kümesi içindeki elemanları tek tek deneyerek n'i tam bölüp bölmediğini kontrol eder. n için düşündüğümüzde öncelikle p_1 asal sayısının n'yi bölüp bölmediği kontrol edilir. Eğer p_1, n'yi bölerse n'nin bir çarpanı elde dilmiş olur. Sonraki adımda bu işlem n bölünmeyene kadar tekrarlanır. Bu aşamadan sonra yeni bir asal sayı ile işlem devam ettirilir. Bu şekilde küçük asal çarpanlara sahip asal olmayan sayılar için hızlı bir şekilde çalışan bir method ortaya konmuştur. Ancak günümüz sayıları (RSA modülleri) düşünüldüğünde bu yöntemin uygulanabilirliği mümkün olmayacaktır.

3.3 Pollard Rho Metodu

Küçük çarpanlı sayılar için özelleşmiş bir algoritma olan Pollard Rho metodu 1975 yılında John Pollard tarafından sunulmuştur. Algoritma sonlu bir dizi üzerinde kendini

tekrar edecek şekilde tasarlanmıştır. Çarpanlarına ayıracağımız sayı $n = pq$ olsun. $x_{n+1} = x_n^2 + a(\mathrm{mod}\, n)$ iterasyonu kullanılarak bir x değerlerine ait bir S dizisi elde edilmiş olur. Bu durumda herhangi bir başlangıç x_0 değeri için de bu iterasyon kendini tekrar edecektir. Metot basitçe aşağıdaki şekilde çalışmaktadır:

(1) $x, y = 2$ ve $a = 1$ olarak seçilir.

(2) $f_a(x) = x^2 + a\ (mod\ n)$ hesaplanır.

(3) $x = f_a(x)$ ve $y = f_a(f_a(y))$ değerleri hesaplanır.

(4) $d = ebob(x - y, n)$ olmak üzere;

- $1 < d < n$ ise bir nontriviyal çarpan bulunur $p = d$.
- $d = 1$ ise 3. adıma dönülür.
- $d = n$ ise $a = a + 1$ olarak hesaplanır ve 2. adıma dönülür.

1980 yılında Richard P. Brent bu algoritmayı geliştirmiştir ve algoritmanın bu versiyonu orjinaline göre %25 daha hızlı çalışmaktadır [4]. Orijinal versiyona göre farklılık gösteren durum ise dizinin tespit yöntemindeki geliştirmelerdir.

3.4 Pollard p-1 Metodu

Pollard Rho metodunun bir türevi olan bu metot Küçük Fermat Teoremini baz almaktadır. p değeri bir çarpan olmak üzere, p-1'in çarpanlarının bir sınır olarak belirtilen B değerinden küçük olduğu durumlarda başarılı bir şekilde çalışmaktadır. Aksi halde algoritma başarısız olacaktır.

Fermat'ın teoreminden herhangi bir a ve asal p değerleri için $a^{(p-1)} \equiv 1(mod\ p)$ olduğunu biliyoruz. Bu durumda $a^{(p-1)} - 1 \equiv 0\ (mod\ p)$ denkliği elde edebiliriz. Böylece p, $a^{(p-1)} - 1$ 'in bir çarpanıdır. Pollard $p - 1$ metoduda bu denkliği kullanarak $a^k - 1$ şeklinde değerler bulmaya çalışır ve n ile ortak bir bölenleri olup olmadığına bakar. Metod basitçe aşağıdaki şekilde çalışmaktadır:

1. $1 < a < n$ olacak şekilde bir n, B sınırı ve $k \neq 2$ olmak üzere bir k değeri seçilir.($k = B!$)
2. $ebob(a, n) \neq 1$ ise bir çarpan bulunmuş olur.
3. $ebob(a, n) = 1$ ise $t = a^k (mod\ n)$ ve $d = ebob(t - 1, n)$ hesaplanır.

4. Eğer $d|n$ ise bir çarpan bulunmuş olur.
5. d,n'i bölmüyorsa yeni a ve k değerleri seçilir ve 2. adıma dönülür.

Örneğin; $n = 4088459$ sayısını ele alalım ve Pollard $p-1$ metodu ile çarpanlarına ayıralım. $a = 2$ ve $B = 7$ olsun ($k = B!$).
Bu durumda

$$t = 2^{7!} \ (mod\ 4088459)$$

işleminin sonucunda $t = 377180$ olarak hesaplanır. $d = ebob(t-1, n)$'den

$$d = ebob(377179, 4088459) = 2017$$

olarak bulunur ve 2017 , 4088459 sayısını böler. 4088459 sayısı 2017 x 2027 olacak şekilde çarpanlarına ayrılmış olur.

3.5 Kongrü Kareler (Congruent Squares)

Kongrü karaler metodu 1798 yılında Legendre'nin yayınladığı bir yöntemdir. Legendre kongrüansı olarak da bilinir.

Tanım 3.5.1 *Legendre Kongrüansı*

$$x^2 \equiv y^2 \ \ mod\ n, \text{ öyle ki } \ 0 \le x \le y \le n, \ \ x \ne y, \ \ x + y \ne n.$$

x ve y bu kongrüansı sağlayan iki sayı olmak üzere $ebob(n, x-y)$ ve $ebob(n, x+y)$ değerlerinin n'nin birer çarpanları olması muhtemeldir. Yani $n = pq$ olmak üzere;

$$x^2 \equiv y^2 \ mod\ n$$

Ve bu denklik bize gösterir ki;

$$\begin{aligned} x^2 \equiv y^2 \ mod\ n \ &\Leftrightarrow n|x^2 - y^2 \\ &\Leftrightarrow n|(x-y)(x+y) \\ &\Leftrightarrow pq|(x-y)(x+y) \\ &\Leftrightarrow p|(x-y) \text{ ya da } p|(x+y) \\ &\Leftrightarrow q|(x-y) \text{ ya da } q|(x+y) \end{aligned}$$

Buradan $ebob(n, x \pm y)$ hesaplanarak p ve q değerleri bulunabilir. Ancak her zaman $x^2 \equiv y^2 \, mod \, n$ denkliği sağlanamayacağından n'nin bir çarpanını bulmak mümkün olmayabilir [4].

Aşağıda hangi durumlarda bir çarpan bulunabileceğine dair bir çizelge verilmiştir.

Çizelge 3.1 : Kongrü kare yönteminin tüm olasılıkları tablosu.

$p\|(x-y)$	$p\|(x+y)$	$q\|(x-y)$	$q\|(x+y)$	$ebob(n, x-y)$	$ebob(n, x+y)$	Çarpan var mı?
+	-	+	-	n	0	-
+	-	-	+	p	q	+
+	-	+	+	n	q	+
-	+	+	-	q	p	+
-	+	-	+	0	n	-
-	+	+	+	q	n	+
+	+	+	-	n	p	+
+	+	-	+	p	n	+
+	+	+	+	n	n	-

Çizelgeye göre tüm durumlar göz önüne alındığında bir çarpan bulma olasılığı $2/3$ olup, yöntem daha çok küçük çarpanlı sayılar etkilidir.

3.6 Kuadratik Eleme Yöntemi (Quadratic Sieve)

QS yöntemi Carl Pomerance tarafından 1981 yılında geliştirilmiştir [7]. Temelinde Legendre'nin kongrü kareler yöntemi vardır ve karelerin bulunması için farklı bir yaklaşımda bulunmuştur. Yöntemde kuadratik polinomlar ve kuadratik rezidüler kullanılmaktadır. Çarpanlarına ayrılmak istenen sayı n olsun. QS metodu aşağıdaki $x \not\equiv \mp y \, (mod \, n)$ ve $x^2 \equiv y^2 \, (mod \, n)$ denkliklerini sağlayan birer x ve y değeri bulmaya çalışır. Buradan $(x-y)(x+y) \equiv 0 \, (mod \, n)$ denkliği yazılabilir. Euclidean algoritması kullanılarak (x-y,n) hesaplanabilirse n'nin bir non-trivial çarpanı bulunmuş olur. Yöntem basitçe aşağıdaki şekilde çalışmaktadır [8]:

1. $Q(x) = \left(x + \lfloor \sqrt{n} \rfloor\right)^2 - n = \tilde{x}^2 - n$ formunda bir polinom tanımlanır.
2. $Q(x_1), Q(x_2), \dots, Q(x_k)$ hesaplanır.
3. $Q(x_i)$'lerden tam kare olanlar üzerinden bir alt küme $Q(x_{i_1})Q(x_{i_2}) \dots Q(x_{i_r})$ seçilir.

4. 1. adımda tanımlanmış olan polinomdan $Q(x) \equiv \tilde{x}^2 \ (mod\ n)$ denkliğini yazabiliriz. Buradan isteden kongrüler aşağıdaki şekilde elde edilir.

$$Q(x_{i_1})Q(x_{i_2}) \dots Q(x_{i_r}) \equiv (x_{i_1}x_{i_2} \dots x_{i_r})^2 \ mod\ n$$

5. Böylece 3.5. bölümde tanımlanmış yöntem ile bir non-trivial çarpan bulunabilir.

3.7 Sayı Alan Kalburu (Number Field Sieve)

NFS olarak bilinen Sayı Alan Kalburu 1990'lı yıllarda geliştirilmiş eleme temelli bir yöntemdir. Algoritmanın birkaç aşaması bulunmaktadır. Temelinde Legendre'nin kongrü kareler yöntemi bulunmaktadır. RSA'nin 130, 140, 150, 155, 160, 170, 174, 180 basamaklı anahtarları bu yöntem ile çarpanlarına ayrılmıştır [3]. Algoritmanın çalışma adımları aşağıda açıklanmıştır [4]:

i. *(Polinom seçme)* Bir m köküne sahip indirgenemez bir f(x)polinomu seçilir.Örneğin; $f(m) \equiv\ 0\ (mod\ n)$, $f(x) \in \mathbb{Z}[x]$.

ii. *(Çarpan bazları tespiti)* Çarpan bazlarının boyutu belirlenir. Rasyonel çarpan bazı, cebirsel çarpan bazı ve kuadratik karakter bazı tanımlanır.

iii. *(Eleme adımı)* Aşağıdaki koşulları sağlayan bir (a,b) tamsayı ikilisi seçilir.

1. $ebob(a, b) = 1$
2. $a + bm$ rasyonel çarpan bazı üzerinden düzgündür(smooth).
3. $b^{\deg(f)} f(a/b)$ cebirsel çarpan bazı üzerinde düzgündür(smooth).

Bu koşulları taşıyan (a, b) ikilisi bağıntı olarak adlandırılır. Eleme adımının temel amacı olabildiğince fazla bağıntı elde etmektir. Bu aşamanın sonuçları S kümesine tanımlanır.

iv. *(Lineer cebir adımı)* Eleme işleminde biriktirilen bağıntılar filtrelenir. Çakışık bağıntılar ve başka bir bağıntıyla ilişkisi olmayan asal ideal içeren bağıntılar elenir.

Bağıntılar, bağıntı kümelerine yerleştirilir ve GF(2) üzerinde tanımlı büyük bir seyrek(sparse) matrise konumlandırılır.

Matris indirgenerek sonuçlar elde edilir. Örneğin; $mod\ n$'de tam kare olan elemanlar.

v. *(Karekök adımı)* Rasyonel karekök hesaplanır.

$$y^2 = \prod_{(a,b)\in S} (a - bm)$$

α, $f(x)$'in bir kökü olmak üzere; cebirsel karekök hesaplanır.

$$x^2 = \prod_{(a,b)\in S} (a - b\alpha)$$

$ebob(n, x - y)$ ve $ebob(n, x + y)$ hesaplanarak bir p çarpanı bulunmuş olur.

Oldukça etkili olan bu yöntemin yalnızca matris aşaması YBH ortamlarına uygun bir şekilde paralele kodlanabilmektedir.

3.8 Eliptik Eğri Metodu

Lenstra'nın geliştirmiş olduğu eliptik eğri metodu Pollard $p - 1$ yönteminin temellerinden faydalanmaktadır. Ancak yöntemde belirtilen işlemler $(\mathbb{Z}/n\mathbb{Z})$ üzerinde tanımlı E eliptik eğrisi üzerinde yapılmaktadır. Bu yöntem 4. bölümde daha detaylı açıklanmıştır.

4. ELİPTİK EĞRİLER

4.1 Giriş

Elliptik eğriler 1980'li yıllarda Miller ve Koblitz tarafından kriptografi alanında kullanılmaya elverişli olduğu, yayınlarda konu edilmiştir. Sonrasında Lenstra tarafından pek çok kriptografik temel oluşturacak yöntemlere dönüştürülmüştür [9]. Tezde sunulan yönteminde temellerini barındıran Lenstra'nın eliptik eğriler ile çarpanlara ayırma yöntemini açıklamak için aşağıda eliptik eğrilerin tanımı ve grup işlemleri ve sonlu cisimler üzerinde tanımlı eliptik eğriler açıklanmıştır.

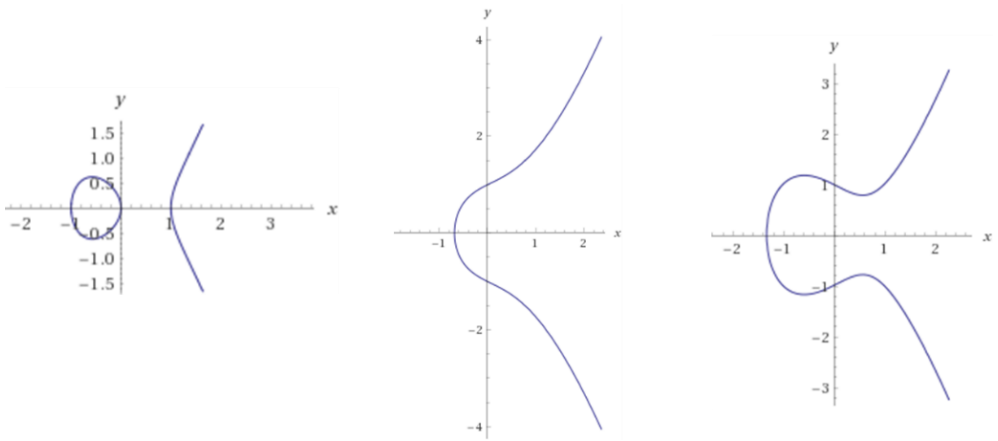
Eliptik eğriler bir K cismi üzerinde tanımlı $a, b \in K$ olmak üzere

$$\{(x, y)|x, y \in K, E: y^2 = x^3 + ax + b\}$$

formunda ki noktalar kümesinden oluşan eğrilerdir. Bu eğrilerin üzerindeki noktalar ve sonsuzdaki bir nokta(etkisiz eleman) abelian bir grup oluşturur. K cismi reel sayılar, rasyonel sayılar vb.den oluşabilir yada sonlu bir cisim olabilir. Sonlu cisimler üzerindeki eliptik eğri grupları sonlu sayıda eleman içermektedir. Genel gösterimleri $a, b, c, d, e \in K$ olmak üzere

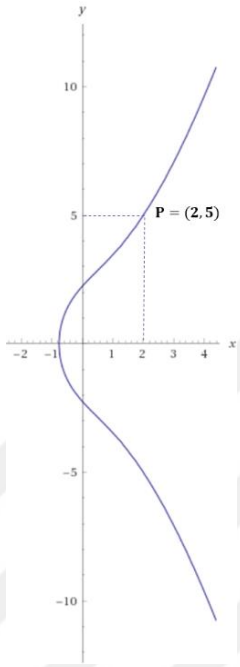
$$y^2 + axy + by = x^3 + cx^2 + dx + e$$

şeklindedir ve bu gösterime Weisstrass gösterimi denir.



Şekil 4.1 : Eliptik eğriler genel gösterimleri.

Bir E eliptik eğrisi üzerindeki noktanın gösterimi P=(x,y) şekildedir. Örneğin: $E: y^2 = x^3 + 6x + 5$ şekildinde bir eliptik eğri olsun. P=(2,5) noktası bu eğri üzerindedir ve gösterimi aşağıdaki şekildedir.



Şekil 4.2 : $E: y^2 = x^3 + 6x + 5$ eğrisi üzerindeki (2,5) noktasının gösterimi.

4.2 Grup İşlemleri

$E: y^2 = x^3 + ax + b$ bir eliptik eğri olsun. Bu eğri üzerinde tanımlı iki nokta $P_1 = (x_1, y_1)$ ve $P_2 = (x_2, y_2)$ olmak üzere bu iki noktanın toplamı aşağıdaki şekilde bulunmaktadır.

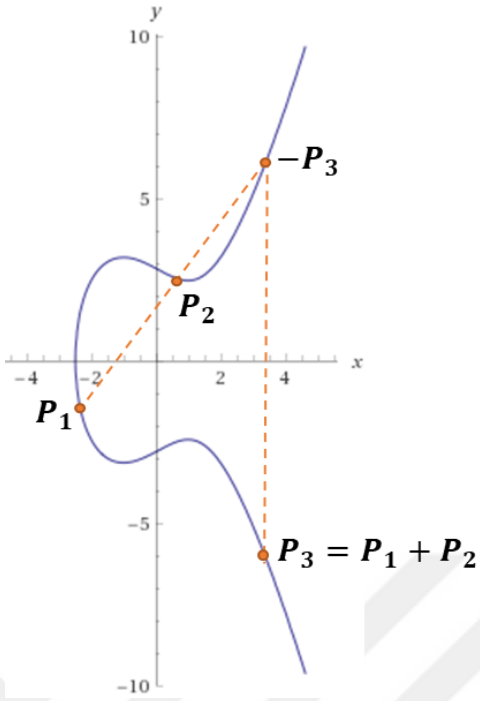
$P_1 + P_2 = P_3 = (x_3, y_3)$ olsun.

$$m = \begin{cases} P_1 \neq P_2 \; ise; & (y_2 - y_1)/(x_2 - x_1) \\ P_1 = P_2 \; ise; & (3x_1^2 + b)/(2y_1) \end{cases}$$

olmak üzere;

$$x_3 = m^2 - x_1 - x_2$$
$$y_3 = m(x_1 - x_3) - y_1$$

eşitliklerinden x_3 ve y_3 hesaplanır.



Şekil 4.3 : Eliptik eğriler üzerinde iki noktanın toplanması.

m'nin değerinin sonsuz olması durumunda $P_3 = \infty$ olur. Ayrıca tüm P noktarı için $\infty + P = P$'dir.

4.3 Sonlu Cisimler Üzerinde Eliptik Eğriler

Sonlu bir cisim $F_p = (Z_p, +, \cdot)$ ve p≠ 2,3 olmak üzere üzerinde tanımlı bir eliptik eğri $E: y^2 \equiv x^3 + ax + b\ (mod\ p)$ şeklinde gösterilmektedir. P eliptik eğri üzerinde bir nokta olsun. Herhangi bir t sayısı için tP'nin birim elemanı olması demek $(t-1)P + P$ hesaplanırken eğimin (4.2 de belirtilen formüldeki m değeri) paydasının p asal sayısına bölünebilmesidir. Yani paydanın $mod\ p$ 'de 0 olması demektir. Bu gözlem Lenstra'nın çarpanlara ayırma algoritmasının temelini oluşturmaktadır.

F_5 üzerinde tanımlanmış $E: y^2 \equiv x^3 + x + 2$ eğrisini ele alalım. Bu eğrinin 4 noktası vardır ve bunlar $\{\infty, (1,2), (1,3), (4,0)\}$ şeklinde gösterilir. İki noktanın toplanması işlemi aynı grup işlemleri ile gerçekleştirilmektedir.

Örneğin; $P = (1,2)$ ve $Q = (4,0)$ olmak üzere $P + Q = K$ işleminde K noktasını bulmak istersek aşağıdaki işlemler yapılır.

$$m = \frac{0-2}{4-1} = \frac{-2}{3} = -4 = 1\ (mod\ 5),\ \ (\frac{1}{3} \equiv 2\ mod\ 5)$$

$$x_3 = 1 - 1 - 4 = -4 = 1\ (mod\ 5)$$
$$y_3 = 1(1-1) - 2 = 3\ (mod\ 5)$$

$K = (1,3)$ olarak hesaplanır.

$E: y^2 \equiv x^3 + ax + b$, F_p sonlu cismi üzerinde tanımlı bir eliptik eğri olsun. Hasse-Weil teoremi eliptik eğri grubundaki eleman sayısının en az $\left(\sqrt{p}-1\right)^2$ ve en fazla $\left(\sqrt{p}+1\right)^2$ olabileceğini söyler. İlginç olan gözlem ise rastgele seçilen a ve b değeri için $E: y^2 \equiv x^3 + ax + b$ eliptik grubunun mertebesi $\left(\sqrt{p}-1\right)^2$ ve $\left(\sqrt{p}+1\right)^2$ eşit olmaksızın bu değerlerin arasında bulunmaktadır.[10]

4.4 Eliptik Eğriler ile Çarpanlara Ayırma Yöntemi

H.W. Lenstra tarafından 1987'de sunulan eliptik eğri metodu aslında Pollard's $p-1$ algoritmasından elde edilmektedir. Pollard $p-1$ yönteminden farklı olarak işlemler çarpımsal bir $(\mathbb{Z}/p\mathbb{Z})^*$ grubu yerine $(\mathbb{Z}/n\mathbb{Z})$ üzerinde tanımlı E eliptik eğrisi üzerinde yapılmaktadır [10]. Eliptik eğriler cisimler üzerinde tanımlı iken halkalar üzerinde de tanımlanabilmektedir. Teorik tanımdan ziyade $(\mathbb{Z}/n\mathbb{Z})$ üzerinde tanımlı eliptik eğriler için Lenstra sadece grup işlemini kullanımıştır. Bir $n > 1$ tam sayısının non-triviyal bir çarpanını bulmak için $(\mathbb{Z}/n\mathbb{Z})$ üzerinde tanımlı bir E eliptik eğrisi ve yine koordinatları $(\mathbb{Z}/n\mathbb{Z})$' de tanımlı E üzerinde bir P noktası seçilir. Aşağıda belirtilen toplama kurallarına göre k bir tamsayı olmak üzere kP hesaplanır. Bu aşamada seçilen E eğrisi ile n'nin bir asal çarpanı p'nin ilişkisine göre n'nin bir çarpanı bulunur. E eliptik eğrisinin F_p'deki mertebesi k'nın bir çarpanı olması aranan ilişki için yeterlidir. Dolayısıyla $kP\ mod\ n$'de hesaplanırken işlem devam edemez çünkü belli bir noktada elde edeceğimiz eğimin paydası p'yi bölen bir sayı olacaktır. Bu durumda paydanın ters alma işlemi gerçekleştirilemez ve n'nin bir non-triviyal çarpanı bulunmuş olur. Eliptik eğri metodu aşağıdaki şekilde çalışmaktadır.

1. $(\mathbb{Z}/n\mathbb{Z})$ üzerinde rastgele bir $E: y^2 = x^3 + ax + b$ eğrisi, E üzerinde $P(x, y)$ noktası, B sınırı ve bir k tamsayısı seçilir.($k = B!$)
2. $Q = kP \in E(\mathbb{Z}/\mathbb{Z})$ noktası hesaplanır.
3. I birim eleman olmak ve p de n'nin herhangi bir çarpanı olmak üzere $Q \equiv I(mod\ p)$ sağlanırsa yani Q birim elemana eşit olursa bir non-triviyal çarpan bulunmuş olur.
4. Eğer 3. Adım gerçekleşmezse tekrar 1. Adıma dönülür ve algoritma tekrarlanır.

p herhangi bir asal sayı olmak üzere, seçilen E eğrisinin eleman sayısı p olan bir cisim üzerindeki mertebesinin m olduğu düşünelim; m'nin tüm asal çarpanları bir B sayısından küçükse m sayısına B düzgündür (B-smooth) denir. Lenstra'nın eliptik eğri metodunun zaman karmaşıklığı bu B sayısına bağlıdır. Bu sebeple bu B sayısı oldukça küçük seçilmelidir. Ancak p sayısı büyüdükçe E'nin mertebesinin B-düzgün olma olasılığı düşmektedir. Bu yüzden Lenstra'nın eliptik eğri metodu genellikle küçük asal çarpanlara sahip tam sayılar için daha etkili bir yöntemdir.

Basit bir örnekle açıklamak gerekirse; $n = 104921$ sayısını ele alalım. Ve bu sayıyı çarpanlarına ayırmak için bir $E: y^2 = x^3 + 6x - 3\ (mod\ 104921)$ eliptik eğrisi ve bu eğri üzerinde bir $P = (1,2)$ noktası seçelim. $k = 10!$ olsun ve $10!\,P$ yi hesaplamaya başlayalım. Bu aşamada öncelikle 4.2'de gösterilen grup işlemleri kullanılarak $2P$ hesaplanır.

$$2!\,P = 2P = [45906, 27863]$$

Sonrasında sırasıyla aşağıdaki noktalar hesaplanarak $10!\,P$ noktasını hesaplamak için işlemlere devam edilir.

$$2(2P) = 4P$$

$$3!\,P = 3(2P) = 2P + 4P = [66051, 92107]$$

$$2(6P) = 12P$$

$$4!\,P = 2(12P) = [20678, 7346]$$

$$2(24P) = 48P$$

$$2(48P) = 96P$$

$$5!\,P = 24P + 96P = [82270,42210]$$

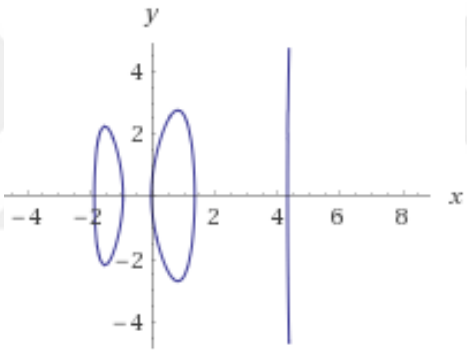
$$2(120P) = 240P$$

$$2(240P) = 480P$$
$$6!\,P = 240P + 480P = [3250, 41639]$$

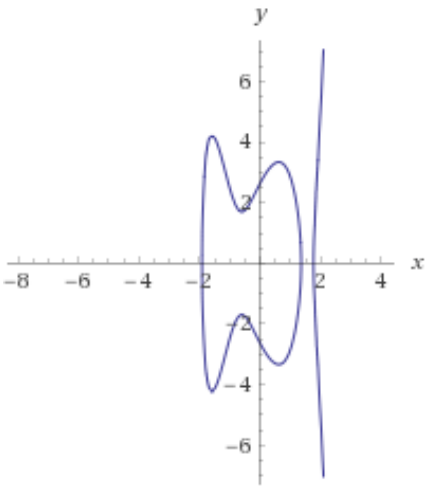
$6!\,P$'ye kadar işlemlerimizi devam ettirebilir ve $6!\,P$ noktasını hesaplayabiliriz. Ancak $7!\,P$'yi hesaplarken herhangi bir aşamada 37315 değerinin $(mod\ 104921)$'te tersini hesaplamamız gerekir ancak 37315'in $(mod\ 104921)$ üzerinde tersi bulunamaz çünkü $ebob(37315, 104921) = 439$'dur. Böylece 104921 sayısı çarpanlarına 439×239 olarak ayrılmış olur.

4.5 Hipereliptik Eğriler

$H: y^2 = f(x)$ formunda tanımlanan eğrilere hipereliptik eğriler denir. $f(x)$'in derecesi $2g + 1$ olarak tanımlanır. Yapılarına göre aşağıda belirtilen şekillerde ki gibi gösterimlere sahip olabilirler.



Şekil 4.4 : $H: y^2 = x^5 - 3x^2 - 8x^3 + 7x^2 + 10x$ hipereliptik eğrisinin grafiği.



Şekil 4.5 : $H: y^2 = x^7 - 2x^5 - 8x^3 + 10x + 7$ hipereliptik eğrisinin grafiği.

Her bir eğrinin kendine ait soyut bir grubu vardır ve bu grup Jacobian grubu olarak adlandırılır. $Jac(H)$ grubun elemanlar $(u(x), v(x))$ ikilisi ile gösterilir. Bu ikililer aşağıdaki koşulları sağlamalıdır.

1. $u(x)$, $F_q[x]$'e ait monik bir polinomdur.
2. $deg(v(x)) < deg(u(x)) \leq g$.
3. $v(x)^2 - f(x)$, $u(x)$'in bir katı olmalıdır.

Jacobian gruplarının grup işlemleri D. Cantor [11] tarafından tanımlanmıştır. Algoritmanın detayları aşağıda açıklanmaktadır.

4.5.1 Cantor Algoritması

$H: y^2 = f(x)$ hipereliptik eğrisine ait Jacobian grubunun iki elemanı $D_1 = (u_1(x), v_1(x))$ ve $D_2 = (u_2(x), v_2(x))$ olmak üzere $D = D_1 + D_2$ işlemi aşağıdaki adımlar izlenerek gerçekleştirilmektedir.

Composition-Toplama.

(1) Genişletilmiş Euclid algoritması kullanılarak $d_1 = ebob(u_1, u_2)$, $d = (d_1, v_1 + v_2) = (u_1, u_2, v_1 + v_2)$ hesaplanır. Bu adımda $d = h_1u_1 + h_2u_2 + h_3(v_1 + v_2)$ eşitliğini sağlayan üç polinom elde edilmiş olur.

(2) $u = \frac{u_1u_2}{d^2}$ ve
$v = \frac{h_1u_1v_2 + h_2u_2v_1 + h_3(v_1v_2 + f)}{b} \ (mod\ u)$ hesaplanır.

Reduction-İndirgeme.

(3) $\deg u \leq g$ koşulu sağlanana kadar aşağıdaki işlemler tekrarlanır.
   a. $u' = \frac{f - v^2}{u}$ ve $v' = -v \ (mod\ u')$ hesaplanır.
   b. $u = u'$ ve $v = v'$ olarak atanır.

(4) e, u'nun başkatsayısı olmak üzere $u = e^{-1}u$ işlemi yapılır.(u monik yapılır.)

(5) $D = (u(x), v(v))$ hesaplanmış olur.

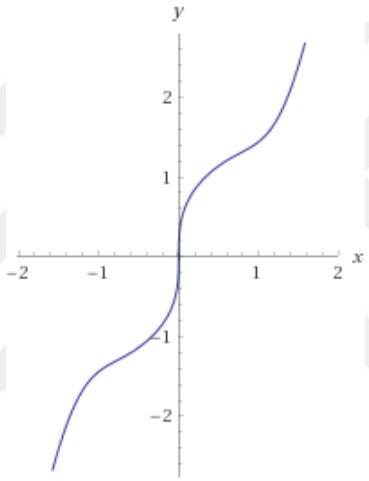
Hipereliptik eğriler, $n = pq$ formundaki bir sayının çarpanlarına ayrılmasında kullanılacak olursa, eliptik eğriler metoduna benzer şekilde $kD\ mod\ n$ hesaplanır. n'nin bir çarpanını bulmak içinse $kD\ mod\ q$ ya da $kD\ mod\ p$ 'den herhangi birinin etkisiz elemanı vermesi gerekmektedir.

Hipereliptik eğriler ile de çarpanlara ayırma konusunda çalışılmış olsa da standart eliptik eğri metotundan daha verimli olduğu gözlemlenememiştir [12].

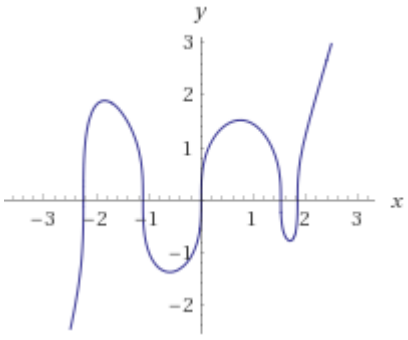
5. SÜPERELİPTİK EĞRİLER

5.1 Giriş

K cismi üzerinde tanımlanmış $\deg(f(x)) \nmid m$ olmak üzere $y^m = f(x)$ şeklindeki çakışık kökü bulunmayan eğrilere süper elliptik eğriler denir. Yapılarına göre bazı süperliptik eğri örneklerinin grafikleri aşağıda gösterilmiştir.



Şekil 5.1 : S: $y^3 = x^7 - x^5 + 3x$ süpereliptik eğrisinin grafiği.



Şekil 5.2 : S: $y^3 = x^5 - 6x^3 + x^2 + 7x$ süpereliptik eğrisinin grafiği.

Süpereliptik eğrilerin elemanaları ait oldukları Jacobian gruplarından gelmektedir. Jacobian grupları ideal sınıf gruplarına denk gelmekte ve böylece elemanlar da ideallerden oluşmaktadır.

S: $y^m = f(x)$ olmak üzere S eğrisinin Jacobian gruplarına ait elemanları($I \in Jac(S)$) Hermite normal form [13] olarak adlandırılan gösterim ile aşağıdaki şekilde gösterilmektedir [14]:

$$[a_{1,1}(x), a_{2,2}(x)y + a_{2,1}(x), \dots, a_{m,m}(x)y^{m-1} + \cdots + a_{m,1}(x)]$$

öyle ki $a_{i,j}(x) \in K[x]$ olmak üzere $\deg(a_{i,j}) < \deg(a_{i,i})$ ve $a_{i+1,i+1}(x) | a_{i,i}(x)$ koşullarını sağlar.

Tasarlanan süpereliptik eğriler ile çarpanlara ayırma algoritmasında 3. dereceden süpereliptik eğriler kullanılmakta olup aşağıda 3. dereceden ideallerin grup işlemleri gösterilmektedir.

$K[C]$ bir ideal sınıf grubu ve I bu grubun içinden herhangi bir ideal olmak üzere; $m = 3$ için I ideali $[s, s'(u + y), v + wy + y^2]$ formundadır. s, s', u, v, w polinomları $K[x]$ üzerinden seçilir ve aşağıdaki koşulları sağlamalıdır[5].

$$s' | s, \qquad u^3 \equiv -f \ mod(s/s'), \qquad v \equiv w^2 \ mod(s'),$$
$$v - uw - u^2 \equiv 0 \ mod(s/s'), \qquad uv - uw^2 \equiv f - vw \ mod(s)$$

Bu koşullar altında ideallerin grup işlemleri aşağıdaki şekilde yapılmaktadır:

5.2 Grup İşlemleri

Süpereliptik eğriler üzerinde iki idealin çarpım işlemi aşağıda belirtilen algoritma ile gerçekleştirilmektedir [15].

1. $I_1 = [s_1, s_1'(u_1 + y_1), v_1 + w_1 y_1 + y_1^2]$ ve $I_2 = [s_2, s_2'(u_2 + y_2), v_2 + w_2 y_2 + y_2^2]$ olmak üzere iki ideal seçilir.
2. Yarı-genişletilmiş Euclid algoritması kullanılarak d hesaplanır.

$$d = ebob(s_1/s_1', s_2/s_2') = r_1 s_1/s_1' + r_2 s_2/s_2'$$

3. Euclid algoritması ile d_1 hesaplanır.

$$d_1 = \frac{ebob(d, u_1 - u_2)}{ebob(d, f)}$$

4. $s_3 = s_1 s_2 \frac{d_1}{d}, \quad s_3' = s_1' s_2' \frac{d}{d_1}, \quad u = u_1 - (u_1 - u_2)(r_1 \frac{s_1}{s_1' d})$ değerleri hesaplanır.

5. d_2 ve r_3 yarı-genişletilmiş Euclid algoritması ile hesaplanır.

$$ebob(d_1, 3u^2) = d_2 = 3r_3 u^2 + r_4 d_1$$

6. $U' = u - r_3(\frac{u^3 + f}{d_2})$ olarak atanır.

7. $\deg U < \deg(S/S')$ olmak üzere; $U \equiv U'(mod\ S/S')$ hesaplanır.

8. Genişletilmiş Euclid algoritması ile aşağıdaki eşitlik hesaplanır.

$$\begin{aligned} 1 = ebob(s_1, s_1' s_2', s_1'(u_1 + w_2), s_2, s_2'(u_2 + w_1), v_1 + v_2 + w_1 w_2) \\ = \quad a_1 s_1 + a_2 s_1' s_2' + a_3 s_1'(u_1 + w_2) + a_4 s_2 + a_5 s_2'(u_2 + w_1) \\ + a_6(v_1 + v_2 + w_1 w_2) \end{aligned}$$

9. $V' = a_1 s_1 v_2 + a_2 s_1' s_2' u_1 u_2 + a_3 s_1'(u_1 v_2 + f) + a_4 s_2 v_1 + a_5 s_2'(u_2 v_1 + f) + a_6(v_1 v_2 + w_1 f + w_2 f)$ hesaplanır.

10. $W' = a_1 s_1 w_2 + a_2 s_1' s_2'(u_1 + u_2) + a_3 s_1'(u_1 w_2 + v_2) + a_4 s_2 w_1 + a_5 s_2'(u_2 w_1 + v_1) + a_6(w_1 v_2 + v_1 w_2)$ hesaplanır.

11. $\deg W < \deg S'$ olmak üzere; $W = W' + qS'$ hesaplanır.

12. $\deg V < \deg S$ olmak üzere; $V \equiv V' + qS'U\ (mod\ S)$ hesaplanır.

13. $I_3 = [S, S'(U + y), V + Wy + y^2]$ olarak bulunur.

6. SÜPERELİPTİK EĞRİLER İLE ÇARPANLARA AYIRMA

Teorem 5.3.1 n asal olmayan bir tam sayı ve H bir hipereliptik eğri olsun. D, $Jac(H)$'nin bir elemanı ve k bir pozitif tamsayı olmak üzere, p ve q asal sayılarının n'yi böldüğünü varsayalım. Eğer $kD\ mod\ p$, $(u_p(x), v_p(x))$ ikilisini ve $kD\ mod\ q$, $(u_q(x), v_q(x))$ ikilisini verirse, öyleki $deg(u_p(x)) \neq deg(u_q(x))$ eşitsizliği sağlansın. Bu durumda $kD\ mod\ n$ hesaplanırken n'nin bir çarpanı (p veya q) bulunur.

Verilen teorem süpereliptik eğrilere uygulandığında çarpanlara ayırma algoritması aşağıdaki şekilde çalışmaktadır:

(1) $S: y^3 = (x-1)g(x)$ formunda bir eğri seçilir.

(2) S'nin ideal sınıf grubu üzerinden rastgele bir

$$I = [a_{1,1}(x), a_{2,2}(x)y + a_{2,1}(x), \dots, a_{m,m}(x)y^{m-1} + \cdots + a_{m,1}(x)]$$

elemanı seçilir. $(m = 3)$

(3) $I^{B!}\ mod\ n$ hesaplanır.

(4) $\deg\left(a_{i,j,p}(x)\right) \neq \deg\left(a_{i,j,q}(x)\right)$ koşulunda algoritma durur ve p veya q bir çarpan olarak bulunmuş olur. Aksi halde (1). adım tekrarlanır.

Algoritma ve yukarıdaki teoreme istinaden hipereliptik eğrilerdeki iki bileşen $(u_q(x), v_q(x))$ üzerinde çarpanlara ayırma işlemi yapıldığında $\deg\left(u_q(x)\right) \neq \deg\left(v_q(x)\right)$ eşitsizliğinin sağlanması, 3 bileşen içeren S süpereliptik eğrisinde $\deg\left(a_{i,j,p}(x)\right) \neq \deg\left(a_{i,j,q}(x)\right)$ eşitsizliğinin sağlanmasına göre daha düşük bir olasılıktadır. Seçilen eğrinin derecesi (m) değeri artırılırsa bir çarpan bulma olasılığı da artmış olacaktır. Aşağıdaki çizelgede eğrinin derecesine (m) göre oluşacak ideallerdeki polinom sayıları belirtilmiştir. Bu durumda bir ideal ne kadar farklı polinom içerirse derece farklılığı elde etme olasılığıda aynı orantıda artmış olacaktır.

Çizelge 6.1 : Süpereliptik eğrilerin derecesine göre idealleri ve içerdikleri polinom sayıları.

Süpereliptik Eğri	İdeal	Polinom sayısı
$y^3 = f(x)$	$[a_{1,1}(x), a_{2,2}(x)y + a_{2,1}(x), a_{3,3}(x)y^2 + a_{3,2}(x)y + a_{3,1}(x)]$	6
$y^5 = f(x)$	$[a_{1,1}(x), a_{2,2}(x)y + a_{2,1}(x), \dots, a_{5,5}(x)y^4 + a_{5,4}(x)y^3 + a_{5,3}(x)y^2 + a_{5,2}(x)y + a_{5,1}(x)]$	15
$y^7 = f(x)$	$[a_{1,1}(x), a_{2,2}(x)y + a_{2,1}(x), \dots, a_{7,7}(x)y^6 + \cdots + a_{7,1}(x)]$	28
…	…	…
$y^m = f(x)$	$[a_{1,1}(x), a_{2,2}(x)y + a_{2,1}(x), \dots, a_{m,m}(x)y^{m-1} + \cdots + a_{m,1}(x)]$	$\frac{m(m+1)}{2}$

Yukarıdaki çizelgeden de görülebileceği gibi hipereliptik eğriler metodunun bir çarpan bulma olasılığı, m bileşen içeren bir S süpereliptik eğrisine göre daha düşük bir olasılıktadır. Bu sebeple önerdiğimiz yöntemde bu olasılık avantajından yararlanmak amacıyla süpereliptik eğriler temel alınmıştır. Ayrıca algoritma tamamen paralleşebilen bir yapıya sahiptir bu durumda her biri farklı düğümlere dağıtılabilecek yüzlerce farklı eğri kullanılarak hızlı bir yöntem elde etmek mümkündür.

Süpereliptik eğrilerin pratikte dezavantajlı olmasının en büyük nedeni ilgili grup işlemlerinin çok fazla sayıda polinom ebob(polynomial gcd)'larını içermesidir. Eliptik eğriler çok uzun zamandan itibaren pratikte birçok amaç için kullanıldıkları için grup operasyonu eliptik eğriler için çok hızlı yapılmaktadır. Hipereliptik eğriler için ise birçok güncelleme yapılmış ve grup işlemleri iyileştirilmiştir. Aşağıda verilen tabloda bu özel eğriler için gerekli olan polinomu ebob sayıları verilmiştir.

Çizelge 6.2 : Eğrilerin grup işlemlerinin teorik sonuçları.

İşlemler	Süpereliptik Eğriler	Hipereliptik Eğriler	Eliptik Eğriler
1 Toplama İşlemi (Addition)	8 Polinom Ebobu	2 Polinom Ebobu	Sadece Sayısal İşlemler

Bir polinomu ebobunun maliyeti; derecesi d olan bir polinomun mod p üzerindeki ebobu bulunurken zaman karmaşıklığı $O(d^2\ (log\ p))$'dir. Dolayısıyla zamandan kaybedilen anavtajı tersine çevirmek için süpereliptik eğrinin derecesinin en az 5 olması gerekmektedir. Yani kullanılacak süpereliptik eğri $y^5 = f(x)$ formunda olmalıdır.

Yapılan testler sonucunda grup işlemlerinin harcadığı zaman aralıklarının veri tablosu aşağıda verilmiştir.

Çizelge 6.3 : Eğrilerin grup işlemlerinin deneysel sonuçları.

İşlemler	Süpereliptik Eğriler	Hipereliptik Eğriler	Eliptik Eğriler
2P Hesaplanması	≈0,005 sn	≈0,001 sn	≈0,001 sn
100!P Hesaplanması	≈6,24 sn	≈0,183 sn	≈0,093 sn

$y^3 = f(x)$ formunda bir süpereliptik eğri 5. dereceden bir hipereliptik eğri ve 3. dereceden bir eliptik eğri kullanılmış olup, n değeri 87181 olarak alınmıştır. Her bir eğri üzerindeki nokta/ideal'in grup işlemleri gerçekleştirilmiştir. İşlem sırasında nokta/ideal olarak seçilen P'den yola çıkarak $100!\ P$ hesaplanmıştır. Yapılan işlemler İntel Core i5 3.8 GHz işlemciye ve 16 Gb RAM'e sahip makine üzerinde gerçekleştirilmiştir.

7. SONUÇ VE ÖNERİLER

Önerilen algoritma PARI [16] kürüphanesi kullanılarak C++ programa diliyle gerçeklenmiştir. Süpereliptik eğriler ile çarpanlara ayırma algoritması Lenstra'nın eliptik eğriler ile çarpanlara ayırma yöntemine göre teoride olasılıksal olarak daha avantajlı bir yöntemdir. NFS algoritmasının yalnızca matris işlemlerinin paraleleşebildiği göz önüne alınırsa, eliptik eğri algortiması tamamen paralelleşebilen yapısı sayesinde daha avantajlı olarak görülebilir. Fakat, olasılıksal bir metot olması sebebiyle, algoritma geliştirilerek süper eliptik eğriler ile bu olasılığın artırılması hedeflenmiştir. Ancak, algoritmanın başarıya ulaşması için pek çok farklı eğri için algoritma çalıştırılmalıdır. Süpereliptik eğrilerdeki grup işlemlerinin detaylı olması nedeniyle yukarıda belirtilen avantajlar pratikte çok fazla fayda sağlayamayabilir. Seri olarak kodlanması durumunda, yapılan test sonuçları Çizelge 6.3'te görüleceği gibi zaman açısından olcukça maliyetlidir. Ancak algoritmanın tamamen paralelleşebilmesinden dolayı çalışmaların HPC ortamlarında kodlanması ve algoritmanın optimize edilmesi ile daha farklı sonuçlara ulaşılması hedeflenmektedir.

Gelecek çalışmalar kapsamında algoritmanın paralel olarak kodlanması ve GPU hesaplama kartları [17] ile daha düşük maliyetle çarpanların ayrılması hedeflenmektedir.

KAYNAKLAR

[1] **Rivest, R., Shamir, A. ve Adleman, L.** (1978). A Method for Obtaining Digital Signatures and Public-Key Cryptosystems. *Communications of the ACM*. 21 (2), 120–126.

[2] **Boneh, D.** (1999). Twenty years of attacks on the RSA cryptosystem. *Notices of the AMS*, 46.

[3] **Url-1**<*https://en.wikipedia.org/wiki/RSA_Factoring_Challenge*>, erişim tarihi 07.11.2018.

[4] **Jensen, P.L.** (2005). *Integer Factorization* (Master Thesis), Department of Computer Science University of Copenhagen.

[5] **Shanks, D.** (1975). Analysis and Improvement of the Continued Fraction Method of Factorization. *Unpub. circa 1975*. Latexed by Stephen McMath March 2004.

[6] **Shanks, D.** (1975). SQUFOF Notes. *circa 1975. Unpub.* Latexed by Stephen McMath March 2004.

[7] **Pomerance, C**. (1943-1993). The number field sieve, *Mathematics of Computation.*

[8] **Landquist, E.** (2001). The Quadratic Sieve Factoring Algorithm, *MATH 488: Cryptographic Algorithms*.

[9] **Trappe, W. ve Washington, L.C.** (2006). *Introduction to Cryptogrsphy with Coding Theory*, Pearson, ISBN 0-13-198199-4.

[10] **Lenstra, H.W.** (1987). Factoring Integers with Elliptic Curves, *Annals of Mathematics*, 126, 649-673.

[11] **Cantor, D.** (1987). Computing in the jacobian of a hyperelliptic curve, *Math.Comp.*, 48, 95-101.

[12] **Lenstra, H.W., Pila, Jr.J. ve Pomerance,C.** (1993). A Hyperellictic Smoothness Test. I, *Philos. Trans. Roy. Soc. London*, Vol.345, pp.397–408.

[13] **Cohen, H.** (2000). *A Course in Computational Algebraic Number Theory*, Springer- Verlag.

[14] **Galbraith, S.D., Paulus, S.M. ve Smart, N.P.** (2000). Arithmetic on Superelliptic Curves, *Math.Comp*, (71), 393-405.

[15] **Baurer, M.** (2003). The arithmetic of certain cubic function fields, *Math. Comp*., 73, 387-413.

[16] **Url-2 <***https://pari.math.u-bordeaux.fr/***>**, erişim tarihi 07.11.2018.

[17] **Url-3**<*http://www.nvidia.com/object/tesla-workstations.html*>, erişim tarihi 07.11.2018.

ÖZGEÇMİŞ

Ad Soyad: Kübra Nari

Doğum Yeri ve Tarihi: BURSA, 06.07.1992

E-Posta: nari15@itu.edu.tr

ÖĞRENİM DURUMU:

Lisans **:** 2015, İstanbul Kültür Üniversitesi, Fen-Edebiyat Fakültesi, Matematik-Bilgisayar (Tam Burslu)

Yüksek Lisans **:** 2019, İstanbul Teknik Üniversitesi, Bilişim Uygulamaları, Bilgi Güvenliği Mühendisliği ve Kriptografi

MESLEKİ DENEYİM VE ÖDÜLLER:

11.2015 - Halen	İstanbul Teknik Üniversitesi,Ulusal Yüksek Başarımlı Hesaplama Merkezi (UHeM), Ayazağa, İstanbul Veritabanı ve Yazılım Geliştirme Uzmanı (PHP,C,C++)
08.2014 -11.2015	Argosia Business Solutions, Gümüşsuyu, Istanbul Yazılım Geliştirme Uzmanı (.NET,C#,MS SQL)
06.2014 - 08.2014	Argosia Business Solutions, Gümüşsuyu, Istanbul Stajyer

YAYIN VE PATENT LİSTESİ:

- **Nari K., Özdemir E., Yaraneri E.** 2018. İkili Kuadratik Formlar ile Çarpanlara Ayırma, *Journal of Engineering Technology and Applied Sciences* 3(3),165-171.

TEZDEN TÜRETİLEN YAYINLAR/SUNUMLAR

- **Ayar G., Nari K., Özdemir E.** 2019. Primality Test with Singular Curves, *AMS Joint Mathematics Meetings 2019,* Ocak 16-19, 2019 Baltimore, USA.

- **Nari K., Özdemir E., Yaraneri E.** 2018. Binary Quadratic Forms and Integer Factorization, *International Conference on Mathematical Studies and Applications(ICMSA2018),* Ekim 4-6, 2018 Karaman,Turkiye.

- **Nari K., Özdemir E., Yaraneri E.** 2018. An Application of Binary Quadratic Forms, *5th International Congress on Fundamental and Applied Sciences(ICFAS2018),* Haziran 18-22, 2018 Üsküp, Makedonya.

- **Nari K., Özdemir E., Yaraneri E.** 2018. An Integer Factorization Algorithm, *AMS Joint Mathematics Meetings 2018,* Ocak 10-13, 2018 San Diego, USA.

- **Nari K., Özdemir E.** 2016. Süpereliptik Eğriler ile Çarpanlara Ayırma, *9. Uluslararası Bilgi Güvenliği ve Kriptoloji Konferansı(ISC Turkey 2016),* Ekim 25-26, 2016 Ankara, Türkiye.