Kritik Bölge Savunma Planlaması İçin Koruma Modelleri Temelinde Bütünleşik Bir Çözüm Yaklaşımı

Orkun Başkan

DOKTORA TEZİ

Endüstri Mühendisliği Anabilim Dalı

Haziran 2020

An Integrated Solution Approach Based on Protection Models for Critical Zone Planning

Orkun Başkan

DOCTORAL DISSERTATION

Department of Industrial Engineering

June 2020

Kritik Bölge Savunma Planlaması İçin Koruma Modelleri Temelinde Bütünleşik Bir Çözüm Yaklaşımı

Orkun Başkan

Eskişehir Osmangazi Üniversitesi
Fen Bilimleri Enstitüsü
Lisansüstü Yönetmeliği Uyarınca
Endüstri Mühendisliği Anabilim Dalı
Endüstri Mühendisliği Bilim Dalında
DOKTORA TEZİ
Olarak Hazırlanmıştır

Danışman: Prof. Dr. Müjgan Sağır

Haziran 2020

ETİK BEYAN

Eskişehir Osmangazi Üniversitesi Fen Bilimleri Enstitüsü tez yazım kılavuzuna göre, Prof. Dr. Müjgan Sağır danışmanlığında hazırlamış olduğum "Kritik Bölge Savunma Planlaması İçin Koruma Modelleri Temelinde Bütünleşik Bir Çözüm Yaklaşımı" başlıklı DOKTORA tezimin özgün bir çalışma olduğunu; tez çalışmamın tüm aşamalarında bilimsel etik ilke ve kurallara uygun davrandığımı; tezimde verdiğim bilgileri, verileri akademik ve bilimsel etik ilke ve kurallara uygun olarak elde ettiğimi; tez çalışmamda yararlandığım eserlerin tümüne atıf yaptığımı ve kaynak gösterdiğimi ve bilgi, belge ve sonuçları bilimsel etik ilke ve kurallara göre sunduğumu beyan ederim. 15/06/2020

Orkun Başkan

İmza

ÖZET

Günümüzde kritik tesislerin korunması önemli problemlerden biridir. Özellikle 11 Eylül 2001'deki ve AB ülkelerinde gerçekleşen terör saldırıları ile birlikte kritik tesislerin güvenliğine yönelik ABD, AB ve gelişmiş diğer ülkelerde pek çok çalışma başlamıştır. ABD Başkanlık politika direktifinde (PPD-21) 16 kritik sektör belirtilmiştir. Bu çalışmada kritik tesislerden sayılan ve enerji sektörünün önemli bir bileşeni olan elektrik şebekeleri ele alınmıştır. Belirli bir bölgenin elektrik tedarikini kesintisiz olarak sağlayacak şekilde elektrik şebekesi bileşenlerinin hangilerinin korunacağının belirlenmesi olarak problemi özetleyebiliriz. Ayrıca kritik bölgelerin savunulmasında, karar vericileri destekleyecek bir çözüm yöntemi önerilmesi amaçtır. Bu çalışma kapsamında "saldırı" odaklı bir çerçeve üzerinde durulmuştur. Yaklaşım bazı düzenlemelerle ortaya çıkabilecek arızalara karşı gerçekleştirilebilecek eylemleri belirlemek için de kullanılabilir.

Çalışmada, daha önceki çalışmalarda göz ardı edilmiş olan farklı saldırı tipleri ve karşı gelen savunma tipleri göz önüne alınmıştır. Yine bu çalışmada yük atmanın yapılacağı bölgenin kritikliği amaca dâhil edilmiştir. Yukarıda tanımlanan problem için matematiksel modellere dayalı yeni çözüm önerileri geliştirilmiş ve deneysel sonuçlarla yöntemlerin kullanılabilirliği ortaya konmuştur. Gerçek boyutlu problemler ele alındığında ise çözüm uzayı büyümekte ve NP-zor bir probleme dönüşmektedir. Bu problemler için Genetik Algoritma temelli başlangıç popülasyonunun kural tabanlı oluşturulduğu k_GA ve başlangıç popülasyonunun rassal oluşturulduğu r_GA sezgisel algoritmaları geliştirilmiştir. Bu yaklaşımlar GAMS 24.0.2 ile etkileşimli olarak Python 2.6.1 programlama diliyle kodlanmıştır. Kritik sistemlerin korunma durumları, korunma kaynaklarının yoğunluğuna göre sıkı, orta ve gevşek olarak ayrılmıştır. Önerilen sezgisellerin hangi kategorilerde daha iyi bir performans gösterdikleri analiz edilmiştir. Özellikle elektrik sistemi gibi koruma kaynaklarının çok olduğu sıkı şekilde korunması muhtemel sistemlerde k_GA sezgiselinin kullanımının daha etkin olacağı değerlendirilmiştir.

Anahtar Kelimeler: Yasaklama problemleri, koruma problemleri, elektrik şebeke problemleri, kritik altyapılar, Genetik Algoritmalar, üç seviyeli programlama, Stackelberg oyun modeli, çoklu saldırı, arz talep dengesi

SUMMARY

Nowadays, protection of critical facilities is one of the important problems. Followed by terrorist attacks on September 11, 2001 and terrorist attacks in the EU countries, many studies have started related to the security of critical facilities in the USA, EU and other developed countries.16 critical sectors are specified in the US Presidential policy directive (PPD-21). In this study, electricity networks considered from critical facilities are discussed. We can summarize the problem as determining which of the electrical network components will be protected so as to ensure uninterrupted electricity supply for a particular regionThis study focuses on interruptions caused by "attack".

In this study, different types of attacks and corresponding defense types that are ignored in previous studies were taken into consideration. In addition to the previous studies on electrical networks, the criticality of the area where the load is shedding is included in purpose function by this study. This scope of work, for the problem described above, new solutions based on mathematical models were developed and the usability of the method was demonstrated with experimental results. In real size problems, the solution space is growing and it becomes an NP-hard problem. Genetic Algorithm based k_GA (initial population is rule-based) and r_GA (initial population is randomly) heuristics have been developed NP-hard problems. In particular, it has been evaluated that the use of k_GA heuristics will be more effective in systems where there are many protection sources such as electrical system and which are likely to be tightly protected.

Keywords: Interdiction Problem, protection problem, electric grid problem, critical infrastructures, Genetic Algorithm, tri-level programing, Stackelberg game model, multiple attack, supply-demand balance**.**

TEŞEKKÜR

Öncelikle doktora tez çalışmam sırasında vermiş olduğu bilimsel katkılardan ve desteklerden dolayı; yapılan işe tutku ile bağlanmamı sağlayan ve akademik çalışma disiplinini örnek aldığım Doktora tez danışmanım sayın Prof. Dr. Müjgan Sağır'a teşekkür ederim.

Tez izleme komitemde yer alan sayın Prof. Dr. Muzaffer Kapanoğlu ve sayın Prof. Dr. İbrahim Akgün'e her tez izleme toplantısında çalışmanın bir adım ileri taşınmasında sağladıkları katkılar ve örnek akademik kişilikleri ile yaptıkları yönlendirmeler için teşekkür ederim. Yine tezimin elektrik şebeke modellerini daha gerçekçi olarak yansıtmama sağladığı katkı için sayın Dr. Öğr. Üyesi Burak Urazel'e teşekkür ederim.

Bu zamana kadar yetişmem ve hiçbir zaman esirgemedikleri destekleri için Annem, Babam ve aileme teşekkür ederim. Bu tez, sevgili eşim Hülya ve çocuklarım Çınar ve İpek'ten esirgediğim zamanlarda tamamlandı. Kendilerine, bitip tükenmeyen sevgileri, destekleri ve hoşgörüleri için çok teşekkür ediyorum.

İÇİNDEKİLER

Sayfa

İÇİNDEKİLER (devam)

Sayfa

İÇİNDEKİLER (devam)

Sayfa

ŞEKİLLER DİZİNİ

Şekil **Sayfa**

ŞEKİLLER DİZİNİ (devam)

Şekil **Sayfa**

ÇİZELGELER DİZİNİ

Çizelge **Sayfa**

ÇİZELGELER DİZİNİ (devam)

Çizelge **Sayfa**

1. GİRİŞ VE AMAÇ

Günümüzde kritik tesislerin korunması önemli problemlerden biridir. Özellikle 11 Eylül 2001 ve AB ülkelerinde gerçekleşen terör saldırıları ile birlikte ABD, AB ve gelişmiş diğer ülkelerde kritik tesislerin güvenliğine yönelik pek çok çalışma başlamıştır.

Kritik tesisler, bir ülke için yüksek önem düzeyine sahip, herhangi bir sebeple, savaş, terörist saldırı, doğal afetler vb. gibi kendilerine verilecek zarar neticesinde ülkeyi önemli ölçüde etkileyebilecek tesislerdir. Bu tesislerde oluşabilecek zafiyet, güvenlik açığının oluşmasına, toplumsal yaşamın etkilenmesine ve ekonomik zararların oluşmasına yol açacaktır.

Ülkemizin bulunduğu coğrafya ve günümüz çıkar çatışmaları göz önüne alındığında çok çeşitli tehditler söz konusudur ve bu tehditlere karşı kritik tesislerin savunulması önemli konuların başında gelmektedir. NATO Genel Sekreteri Anders Fogh Rasmussen (2009-2014, NATO 12. Genel Sekreteri) günümüzdeki tehditleri 'küresel terörizm', 'balistik füzelerin yayılması' ve 'siber-güvenlik' olarak sınıflandırmıştır.

Bu çalışma ile terör tehdidine karşı kritik tesislerin etkin bir şekilde korunmasını sağlayacak yöntem geliştirilmesi amaçlanmaktadır. Böylece kritik tesislerin savunma planlamasında, karar vericilere, matematiksel modellere ve sezgisel yaklaşımlara dayalı bir destek sistemi sunulmaktadır.

Çalışma sürecinin başlangıcında ülkemiz için kritik olabilecek tüm tesislerin, ülkemiz çevresindeki bazı ülkelerin, mevcut ve olası silah sistemleri ile yapabilecekleri saldırılara karşı, nerede konuşlanacak hangi tip silahlarla korunabileceği sorularına yanıt aranması hedeflenmiştir. Ancak problemin geniş kapsamı kritik tesis savunma probleminin genel hatlarıyla ele alınmasının zorluğu ve gerekli "kritik" verinin teminindeki güçlük nedeni ile kapsamın daraltılarak belirli bir alt alana, korunacak belirli ve önemli bir birime yönelmenin yerinde olacağına karar verilmiştir.

ABD Başkanlık politika direktifinde (PPD-21) Çizelge 1.1'deki 16 kritik sektör belirtilmiştir. Bunlar kimya, iletişim, kritik imalat, bilgi teknolojileri vd. sektörlerdir. Bunların en önemlilerinden biri enerji sektörüdür. Elektrik sistemleri de enerji sektörünün en önemli parçalarından biridir.

Çizelge 1.1. Kritik Tesisler (ABD Başkanlık Politika Direktifleri PPD-21)

Kritik Sektörler	
Kimya Sektörü	Ticari Tesisler
İletişim Sektörü	Kritik İmalat Sektörü
Barajlar	Savunma Sanayi Temelli Sektörler
Acil Servisler	Enerji Sektörü
Finans Hizmetleri	Gıda Ve Tarım Sektörü
Hükümet Tesisleri	Sağlık Hizmetleri ve Kamu Sağlığı
Bilgi Teknolojileri	Nükleer Reaktörler, Maddeler Ve Atıklar
Ulaşım Sistemleri	Su ve Atıksu Sistemleri

Bu çerçevede ileriki bölümlerde literatür taramasında değinilecek olan ve yaygın tanımlama biçimi ile ağ veya tesis yasaklama olarak iki türü bulunan yasaklama (interdiction) modelleri ile problemin ele alınmasına karar verilmiştir. Ayrıca saldırılar bir ülkenin diğerine yaptığı saldırılar değil, kritik tesislerin hizmet vermesini engelleyici çeşitli saldırı/arıza vb. olarak da düşünülebilir. Ancak bu çalışma kapsamında ağırlıklı olarak zarar vermeyi hedefleyici bilinçli saldırılar dikkate alınmıştır. Kritik tesislerin yerleri ve ne oldukları bellidir ve problem çerçevesinde bunlar belirlenmeyecektir. Ancak bu kritik tesislerden hangilerinin daha önemli ve mevcut bir sistem içinde daha kırılgan oldukları, bu çalışma ile elde edilen modeller aracılığı ile belirlenebilecektir. Bu belirlenen noktalar gerek bir saldırıya gerekse doğal ya da sistemsel bir arızaya karşı korunması/güçlendirilmesi gereken noktalar olarak ortaya konacaktır.

Yasaklanması ile toplumu önemli derecede etkileyebilecek farklı hizmetler söz konusudur. Bu hizmetler doğalgaz sistemleri, ulaşım alt yapısı, acil servisler ve elektrik sistemleri gibi kritik tesis olarak adlandırılan sistemlerdir. Bu sistemlerde elektrik sağlayıcı birimleri tahrip etmeye dönük girişimler en önemli tehditler olarak görülmüştür.

Yasaklama ifadesi ileriki bölümlerde, faaliyetlerin bir sebeple durması ya da kesintiye uğraması anlamında kullanılmaktadır. Elektrik sistemlerinde oluşabilecek bir kesinti hem ekonomik hem de toplumda oluşturacağı etki açısından önemlidir. Bu tesislerin olası saldırılara karşı etkin bir şekilde savunulması gerekmektedir. Yukarıda örneklenen diğer birimler de son derece kritik olmakla birlikte hemen tümünde elektrik enerjisinin varlığı, ilgili sistemin aktif çalışabilmesi için "ortak" bir gereklilik olduğundan daha önemli bir zorunluluk olmaktadır. Örneğin elektrik sisteminde yaşanan bir sıkıntı, iletişim sektörü, güvenlik birimleri, bankacılık sektörü, ulaşım sistemi (tren, metro, vb. alt yapılar), sağlık sistemi (acil sağlık hizmetleri, ameliyathane, yoğun bakım üniteleri vb.), sınavlar, uçuş kontrol ve kule hizmetleri gibi pek çok sektörü direk etkilemektedir. Hatta salgın dönemlerinde eğitim sistemi, tedarik zinciri sistemleri gibi sistemlerin hizmet sunma biçiminde yaşanan değişiklerde olduğu gibi pek çok beklenmeyen başka sistemi etkileme potansiyeli olduğu görülmektedir. Elektrik sistemlerinin her hizmet sunulan toplumsal sistemde önemli rolü olduğundan, önerilen yöntemin önemli ölçüde yaygın etkisi bulunmaktadır.

Temel olarak ele alınan kritik alt yapı ve hizmetler ve kısıtları değişmekle birlikte çoğu problem saldırgan ve savunanın olduğu bir yapıda olup, saldırganın kritik sisteme en fazla zararı vermek üzere yasaklama yapmaya çalıştığı, savunma planlamacısının ise koruma planını, oluşabilecek zararı enküçükleyecek şekilde oluşturmaya çalıştığı bir yapıya dayanmaktadır.

İkinci bölümde yasaklama/koruma modelleri ele alınmıştır ve daha önce yapılan çalışmalar bu bölümde sunulmaktadır. Gerçek hayat problemlerinde saldırı tipinin ve bu saldırıya karşı alınan önlemin tek tip olmadığı açıktır. Günümüzdeki teknolojik gelişmeler ile sürekli değişen ve çok çeşitli araçları olan hibrit tehditler söz konusudur. Hagelstam ve Narinen'de (2018), NATO bünyesinde hibrit tehditlere karşı yapılan çalışmalar paylaşılmıştır. Bu yazıda belirtilen hibrit tehditlerin, tezin ilerleyen bölümlerde olduğu gibi, hedefin niteliğine ve istenen sonuca bağlı olarak ayrı ayrı veya kombinasyon halinde kullanılabileceği belirtilmiştir. Ardından genel bir yasaklama/koruma problemi için saldırı tiplerindeki değişim göz önüne alınarak geliştirilen iki yeni matematiksel model sunulmuştur. Bu kapsamda belirli bir bölgeye kesintisiz olarak hizmet sağlamak amaçlanmaktadır. Burada hizmet sunan tesisler kritik tesisler olarak kabul edilmiştir ve

bunlardan hangilerinin korunmasının en iyi karar olacağı araştırılmıştır. Bu birimlerin hizmet sunduğu müşteriler ise talep noktalarıdır.

Çalışmada ele alınan problem, Scaparra'da (2008-a) sunulan iki seviyeli RIMF (r-interdiction median problem with fortification) model temel alınarak modellenmiştir ve gerçek durumu daha iyi ortaya koymayı hedefleyen iki yeni model önerilmiştir. Önerilen ilk modelde saldırganın yasaklama çeşidinin tek olamayacağı, yasaklama çeşidine göre koruma çeşitlerinin de farklılaşacağı yönüyle bir eklenti yapılarak problem "İki Seviyeli Çoklu Saldırı Tipli Koruma Modeli (R_eIMF)" olarak 4.1'de sunulmuştur. Yine hizmet sunan tesislerin kapasitelerinin kısıtlı olacağı ve saldırganın arz ve talep arasındaki dengeyi bozmayı hedefleyeceği göz önüne alınarak "Arz Talep Dengesi Bozma Amaçlı İki seviyeli Çoklu Saldırı Tipli Koruma Modeli ($SD - R_e$IMF)" 4.2'de sunulmuştur. Her iki duruma da önceki çalışmalarda rastlanmamıştır.

Önerilen yeni koruma modelleri, kritik sistemlerin korunması problemini, farklı koruma ve saldırı tiplerini göz önüne alarak genel bir biçimde açıklamaktadır. Hizmet alan ve hizmet sunan sistemler kendi özel problemleri çerçevesinde ele alındığında ise problemin kendine özel yapı ve kısıtları nedeni ile ana çerçeve aynı kalmakla birlikte, matematiksel model problemden probleme önemli değişiklikler göstermektedir. Bu çalışmada kritik hizmet sunan sistemler içerisinden pek çok sistemi etkileme potansiyeli olan elektrik sistemleri seçilmiştir. Bu kapsamda 4. Bölümde önerilen çerçevede, gerçek bir elektrik sistemini modellemek amacıyla 5. Bölümde sunulan çalışmalar yapılmış ve "Üç Seviyeli Çoklu Saldırı Tipli Elektrik Şebekesi Koruma Modeli" önerilmiştir.

Beşinci bölümde, elektrik şebekelerinin genel yapısı hakkında bilgi verilmiştir ve elektrik şebekelerine yönelik geçekleştirilen yasaklama/koruma modelleri ve bu alanda yapılan çalışmalar incelenmiştir. 5.3.'te gerçek bir elektrik şebekesinin kısıtları göz önüne alınarak elektrik sistemi, 3 seviyeli şekilde modellenmiştir ve "Üç Seviyeli Çoklu Saldırı Tipli Elektrik Şebekesi Koruma Modeli" yeni bir model olarak sunulmuştur. Modelin üst seviyesi elektrik sistemi planlayıcısının seviyesidir. Planlamacı bu model ile hangi bileşenlerin ne tür saldırılara karşı korunacağının kararını vermektedir, amacı herhangi bir saldırı durumunda belirlenen bölgenin enaz zararı görmesidir. Orta seviye ise saldırganın seviyesidir. Saldırgan elektrik şebekesinde hangi bileşenlerin ne tür saldırılara karşı

savunulduğunu bilmektedir ve enbüyük zararı vermek amacı ile saldırı gerçekleştirmektedir. En alt seviye ise sistem operatörünün seviyesidir. Bu sevide ki operatör saldırı sonrası şebekede gerekli değişikleri yaparak sistemin enaz zararı görmesini sağlar. Sistem planlayıcısı ve sistem operatörü aynı amacı paylaşmakla beraber sistem planlayıcısı saldırı öncesi nerelerin korunacağının kararını verirken, sistem operatörü ise saldırı sonrası sistemde yaptığı değişikler ile sistemin an az zararı görmesini sağlar. Bu modelde saldırı tiplerine ek olarak, yine daha önceki elektrik şebeke çalışmalarında göz ardı edilen hizmet sağlanan bölgenin kritikliği göz önüne alınmıştır. 5.4.'de modelin çözümü için bir çözüm algoritması oluşturulmuş, küçük bir örnek seti ile denenmiştir ve sonuçları bu bölümde paylaşılmıştır. Örnek seti büyüdükçe NP-zor olan bu problemin çözülemediği görülmüş ve Genetik Algoritma (GA) tabanlı sezgisel çözüm yaklaşımları geliştirilmiştir. Ayrıca Bölüm 5.5'de karmaşıklık analizi yapılarak problemin NP-zor bir problem olduğu gösterilmiştir. Bileşen (santral, trafo merkezi ve hat) sayısındaki artış problemi polinom olarak büyütürken, saldırı tipi sayısındaki artış üssel olarak büyümesine sebep olmaktadır.

Altıncı bölümde önce GA metasezgisel yöntemi tanıtılmıştır ve ardından geliştirmiş olduğumuz GA tabanlı r_GA ve k_GA sezgisel yaklaşımları sunulmuştur. Ele alınan problemde, hangi hatların hangi trafo merkezlerinin ve hangi santrallerin daha kritik olabileceği, bir uzmanın mevcut verileri muhakeme ve tecrübesiyle değerlendirmesi ile sezilmektedir. Örneğin en çok bağlantının olduğu trafo merkezi, iki bölge arasında bağlantıyı sağlayan tek bir hat, sistemi besleyen elektrik santrallerinden en büyük kapasiteli olanı ya da en kritik bölgeyi besleyen bileşenleri gibi bazı göstergeler karar vericiye bu kritik noktaların önemli olduğu ve korunması gerektiği konusunda ipuçları vermektedir. Bu göstergeler ile bir kural seti oluşturulmuştur. Geliştirilen k_GA sezgisel yaklaşımında GA başlangıç popülasyonu içinde bu kurallara uygun olan bireylere daha fazla şans tanınmıştır. Bu bölümde deney tasarımı ile her iki algoritmanın eniyi parametre düzeyleri belirlenmiştir.

Yedinci bölümde, önerilen çözüm yaklaşımlarının deneysel sonuçları paylaşılmıştır. 7.1'de "Üç Seviyeli Çoklu Saldırı Tipli Elektrik Şebekesi Koruma Modeli" için beşinci bölümde önerilen çözüm yaklaşımı ile elde edilen çözüm sonuçları paylaşılmış, koruma ve saldırı parametrelerinde yapılan değişikliklerin problemin boyutunu nasıl etkilediği örneklerle ortaya konmuştur. 7.2'de öncelikle 7.1'de verilen problem setleri için GA'nın performansı verilmiş ve matematiksel modelin sonuçları ile karşılaştırılmıştır. Devamında

önerilen sezgisellerin karşılaştırılması için literatürde yer alan ve elektrik şebeke sistemleri ile ilgili bilimsel çalışmalarda test verisi olarak yaygın kullanılan IEEE'nin 14 baralı örnek problem seti (IEEE 14 Baralı Test sistemi, 1962 Şubat ayı itibariyle Amerikan Elektrik Güç Sisteminin (Orta Batı ABD'de) bir bölümünü temsil etmektedir (University of Washington - Electrical Engineering, 1962)) kullanılmıştır. Aynı problem seti koruma kaynakları miktarları değiştirilerek, sıkı orta ve gevşek koruma olmak üzere üç kategoriye ayrılmıştır. Bu örnek problemler üzerinden kural tabanlı önerilen sezgisel çözüm yaklaşımı k_GA ile başlangıç popülasyonun rassal olarak üretildiği r_GA sezgisel çözüm yaklaşımının performansları analiz edilmiş ve karşılaştırılmıştır. Tüm kategorilerde k_GA sezgisel yaklaşımı daha iyi sonuçlar elde etmiştir. Bu farkın anlamlılığı non-parametrik testler ile incelenmiştir. Koruma kaynaklarının yoğun olduğu sıkı kategoride olan problemlerde ve orta kategorideki problemlerde k_GA sezgiseli daha iyi performans göstermiştir. Özellikle elektrik sistemi gibi sıkı şekilde korunması muhtemel sistemlerde k_GA sezgiselinin kullanımının daha etkin olacağı değerlendirilmiştir.

Sekizinci ve dokuzuncu bölümlerde, elde edilen sonuçlar paylaşılmış ve sonraki çalışmalar için öneriler sunulmuştur.

Özetle tezde yapılan çalışmalar ve tezin literatüre katkıları şu şekildedir:

- Çalışmada üç yeni model önerilmiştir.
    - Yeni model 1: İki Seviyeli Çoklu Saldırı Tipli Koruma Modeli (R_eIMF)
    - Yeni model 2: Arz Talep Dengesi Bozma Amaçlı İki seviyeli Çoklu Saldırı Tipli Koruma Modeli ($SD - R_e$IMF)
    - Yeni model 3: Üç Seviyeli Çoklu Saldırı Tipli Elektrik Şebekesi Koruma Modeli
- Literatürde saldırı ve savuma tiplerinin tek tip olduğu varsayılmıştır. Gerçek hayatta ise saldırı tipleri ve karşılığı savunma tipleri çeşitlidir.
    - Önerilen modellerde saldırı tipleri göz önüne alınmıştır.
    - Önerilen modellerde savunma tipleri göz önüne alınmıştır.
- Bir saldırganın saldırıdaki amacının maliyet değil, arz-talep dengesi bozma olduğu göz önüne alınmıştır ve saldırganın arz talep dengesi bozma öncelikli amacını göz önüne alan yeni model 2 geliştirilmiştir.

- Literatürde yapılan çalışmalarda hizmet sunan tesislerin kapasitelerinin sınırsız olduğu varsayılmaktadır. Yeni model 2'de tesislerin kapasite sınırı modele dahil edilmiştir.
- Yeni model 1 ve 2, RIMF modelini temel almıştır. RIMF modelinin, bir tesis hizmet dışı kaldığında talep noktasının hangi tesisten hizmet alacağını belirleyen kısıtı geliştirilmiştir.
- Elektrik şebekelerinin kendisine has kısıtları göz önüne alınarak literatürdeki elektrik şebeke modellerinden farklı yeni bir model geliştirilmiştir (Yeni Model 3). Bu model saldırganın farklı tiplerde saldırı yapabilme durumunun göz önüne alındığı ilk elektrik şebekesi yasaklama modelidir.
- Yeni model 3'te literatürde göz önüne alınmamış olan "elektrik kesintisi yapılacak bölgenin kritikliği" kavramı modele dahil edilmiştir.
- Modellerin test edilmesi amacıyla IEEE'nin 14 baralı test probleminden 12 yeni problem seti oluşturulmuştur.
- Koruma kaynağı yoğunluğuna göre sıkı orta ve gevşek koruma kategorileri olmak üzere problem setleri kategorize edilmiştir. Bu ayrımın istatistiksel olarak anlamlılığı gösterilmiştir.
- Çözüm için tam sayımlama temelinde bir yaklaşım geliştirilmiş ve kodlanmıştır.
- Problemin NP zor sınıfında olduğu, karmaşıklık analizi ile gösterilmiş ve modeldeki yasaklama tipi değişikliğinin, polinom olarak büyüyen bu problemi üssel olarak büyüttüğü gösterilmiştir.
- NP zor bu problem için GA tabanlı r_GA (rassal tabanlı) ve k_GA (kural tabanlı) sezgisel algoritmaları geliştirilmiştir.
- k_GA sezgisel yaklaşımının, literatürde yer alan yasaklama/koruma problemleri çözüm yaklaşımları arasında benzerine rastlanmamıştır.
- r_GA ve k_GA sezgisel yaklaşımlarının sıkı-orta ve gevşek kategoriler için performansları analiz edilmiştir. Sıkı ve Orta koruma kaynağı kategorisi için k_GA'nın daha iyi olduğu gösterilmiştir.

2. LİTERATÜR ARAŞTIRMASI

Tesisler ya da tesislerin sunduğu hizmetler; doğal afetler, terör saldırıları ya da başka bir ülkenin yaptığı saldırı gibi ataklar ile kesintiye uğrayabilir. Bu tarz sistemlere yönelik olarak yapılan kasıtlı saldırılar yasaklama (interdiction) olarak adlandırılır.

Literatürdeki örnekler incelendiğinde problemlerin iki şekilde ele alındığı görülmektedir. Bunlardan biri yasaklamayı yapan (saldırgan) tarafından problemlerin ele alındığı modellerdir, bu modelleri yasaklama modelleri olarak adlandıracağız. Diğeri ise korunacak tesisleri belirleyen savunucu tarafından ele alınan modellerdir ve bu modellere de koruma modelleri diyeceğiz. İki bakış açısının temel farkı modellerde eniyilenmeye çalışılan eğer saldırganın amacı ise modelin yasaklama modeli, eniyilenmeye çalışılan savunanın amacı ise koruma modeli olduğudur.

Genel olarak literatürde yasaklama/koruma modelleri iki kategoride sınıflandırılmaktadır. İlki "Ağ yasaklama modelleri", diğeri "tesis yasaklama modelleri"'dir.

Literatürde öncelikle ağ yasaklama modellerinin çalışıldığı görülmektedir. Wollmer (1964) makalesi bu alandaki ilk çalışmadır, çalışmada ağların (düğümler arası arkların) kesintiye uğraması modellenmiştir. Arkların kesintiye uğratılması ile akış kapasitesini düşürmek ve merkez ile hedef arasındaki en kısa mesafeyi enbüyüklemek amaçlanmıştır. Ağ yasaklama modellerinde tesis ile talep noktası arasındaki bağlantıların (arkların) kesilmesi/korunması ile en kısa yolun enbüyüklenmesi/enküçüklenmesi hedeflenir. Israeli (2002) çalışmasında en kısa yol problemlerinde en kısa yolun en büyüklenmesini amaçlayan bir model ve çözüm önerisi sunmuştur ve problemi saldırgan açısından ele almıştır. Bu modellerde savunmacının ve saldırganın amaçları Çizelge 2.1'deki gibidir:

Çizelge 2.1. Ağ Yasaklama/Koruma Modellerinde Savunan ve Saldıran Açılarından Amaçlar

Savunmacı Açısından Amaçlar	Saldırgan Açısından Amaçlar
Ağdan mümkün olan en kısa sürede geçmek, (Cappanera ve Scaparra, 2011).	En kısa yolun uzunluğunu enbüyüklemek, (Israeli ve Wood, 2002)
Ağdan geçen akış miktarını enbüyüklemek (Shimizu, 2012).	Ağdaki en büyük akışı en aza indirmek
Ağ üzerinde yakalanmadan hareket etmek (Wood, 1993; Cormican vd. 1998).	Ağdaki tespit olasılığını en üst düzeye çıkarmak

Bu modellerin, düşman akışlarını aksatıcı uygulamalar (McMasters ve Mustin, 1970), bulaşıcı hastalıkların kontrolü (Assimakopoulos, 1987), terörle mücadele (Farley, 2003), kaçakçılığın engellenmesi (Washburn ve Wood, 1995) ve nükleer materyallerin kaçakçılığının durdurulması (Morton vd., 2007) gibi çeşitli uygulamalarda kullanıldığı görülmektedir (Ramamoorthy, 2016).

Tesis yasaklama/koruma modelleri ise belirli bir ağdaki en hayati tesislerin tahrip edilmesine odaklanır (Aliakbarian, 2015). Yasaklama/koruma problemleri önceleri savaşlarda tedarik yollarının kesilmesine yönelik konularda çalışılırken, son yıllarda kritik altyapılara yönelik problemlerde ele alınmaktadır (Church, 2004). Yasaklama/koruma modellerinde sistemi savunan ve bu sisteme zarar vermek isteyen bir saldırgan bulunur. Sistemi savunan, sunulan hizmetlerin eniyi şekilde sağlanmasını ve sürekliliğini amaçlar, sisteme zarar vermek isteyen saldırgan ise en fazla zararı vermeyi amaçlar. Ayrıca yasaklama modelleri bir yapının içerisindeki zayıf noktaların çıkarılmasında kullanılan modellerdir. Ancak güvenliğin optimize edilmesini açıkça ele almamaktadır (Scaparra, 2006). Tesis yasaklama alanındaki ilk çalışma Church vd. nin (2004) çalışmasıdır. Bu çalışmada saldırganın bakış açısından problem ele alınmıştır ve RIM (r-interdiction median) ve RIC (r-interdiction covering) olarak isimlendirilen iki model sunulmuştur. Modeller p medyan ve kapsama modellerini temel almaktadır. Ayrıca Scaparra (2008) yasaklamanın etkilerini enküçükleyecek ve tahkimi eniyileyecek şekilde bir tamsayılı doğrusal program önermiştir. Bu model "r-interdiction median problem with fortification (RIMF)" olarak adlandırılan iki seviyeli bir modeldir. Bu makalede yasaklama ile tahkim arasındaki ilişki kullanılmıştır. Yasaklamanın etkilerini azaltmak için mevcut tesislerin güçlendirilmesi ve böylece güvenliğin arttırılması amaçlanmıştır.

Yasaklama/koruma problemlerinin büyük kısmı Stackelberg'in oyun teorisi (savunan (defender)-saldıran (attacker)) olarak ele alınmıştır. Problemde savunucu p kadar tesisten belli miktarını koruyabilir. Saldırgan ise korunmayan r kadar tesise saldırarak sistemin etkinliğini en yüksek oranda azaltmaya çalışır. Bu problemlerde koruma, yasaklamanın en kötü durumu göz önüne alınarak yapılır.

Oyun teorisi olarak ele alınan problemler, saldırganın lider olduğu ve bir ağ operatörünün (savunmacı) takipçi olduğu şekilde, ya da savunmacının lider olduğu (belirli koruma kaynağını tesisleri korumak üzere atadığı) ve saldırganın takipçi olduğu şekilde iki seviyeli olarak modellenmiştir. Ayrıca yapılan çalışmalarda, 3 seviyeli olarak modellemelerin yapıldığı da görülmektedir.

Salmeron vd. (2004) yaptıkları çalışmada bir elektrik şebekesinde bir atak karşısında kritik olan bileşenleri tanımlamaya yönelik olarak iki seviyeli bir model geliştirdiler. Model saldıran-savunan şeklinde modellenmiştir. Salmeron ve Wood'da (2015) yaptıkları çalışmada elektrik şebekelerinde yöneliktir ve model saldıran-savunan şeklinde modellenmiştir. Aynı şekilde Lezama vd. (2017) çalışmalarında saldıran-savunan şeklinde bir yasaklama problemi kurgulamıştır. Aksen vd. (2014) çalışmasında da model saldıran-savunan şeklinde modellenmiştir, saldırgan lider, savunucu ise takipçi durumundadır. Saldırgan hizmet kesintisini enbüyüklemek istemektedir. Savunmacı (takipçisi) ise tüm müşterilerin talebini karşılamaktan sorumludur ve en kötü durumdaki saldırı sonrasında toplam talebin ağırlıklı taşıma maliyeti ve dış kaynak kullanım maliyetini en aza indirmeyi amaçlar.

Losada (2010), (2012-a) yapılan çalışmada ise model zaman periyodlarını göz önüne almıştır ve zarar gören tesislerin yeniden düzelme süreleri modele dahil edilmiştir. Model ise savunan-saldırgan olarak ele alınan iki seviyeli bir modeldir. Aksen ve Aras (2011) çalışmalarında şarj istasyonlarının korunma planı ele alınmıştır, bu çalışmada da model savunan-saldırgan şeklindedir.

3 seviyeli modellerde ise savunan – saldırgan - sistem operatörü şeklinde modellemeler gerçekleştirilmektedir. Wu ve Conejo (2017) yaptığı çalışmada elektrik şebekelerinin savunması problemi için 3 seviyeli bir optimizasyon modeli geliştirdiler. 3

seviyeli ele alınan problemde üst seviye savunma planlamacısı, orta seviye saldırıyı yapanın seviyesi, alt seviye ise sistem operatörüdür. Alguacil vd. nin (2014) elektrik şebekelerinin savunmasını ele alındıkları çalışma da 3 aşamalı bir modeldir. Problemde öncelikle 3 seviye 2 seviyeli şekle dönüştürülmüştür. Jian vd. (2015) yaptıkları çalışmada kentsel demiryolu ağına yönelik saldırılara karşı eniyi korumayı sağlamaya yönelik üç seviyeli bir model geliştirmişlerdir. Problem savunan saldırgan ve kullanıcı olacak şekilde üç seviyeli olarak ele alınmıştır.

Ağ/tesis yasaklama/koruma problemlerinin çözümünde ise literatürde büyük boyutlu problemler için kesin çözüm algoritmalarının kullanıldığı, daha büyük boyutlu problemlerin çözümü için ise sezgisellerin kullanıldığı görülmektedir (Çizelge 2.2).

Kesin çözüm algoritmalarında genellikle tam sayım ve Benders Ayrıştırma Algoritmalarının kullanıldığını görülmektedir. Tam sayım algoritması, sınırlı sayıda düğüm değerlendirmesi ile eniyilemeyi garanti eden sistematik bir arama sağlar (Alguacil, 2014). Losade (2010) ise ayrıştırma metotlarının kullanıldığı iki adet çözüm yaklaşımı geliştirmiştir. İlki Super-Valid-Inequalities (SVI) temel almaktadır. İkinci metot Benders Ayrıştırmasıdır. Wu ve Conejo (2017) yaptığı çalışmada elektrik şebekelerinin savunması problemi için geliştirilen 3 seviyeli bir modelde, alt ve orta seviye birleştirilerek bir enbüyükleme problemi elde edilmiştir, burada Benders Algoritmasından ve ikilinden yararlanılmıştır.

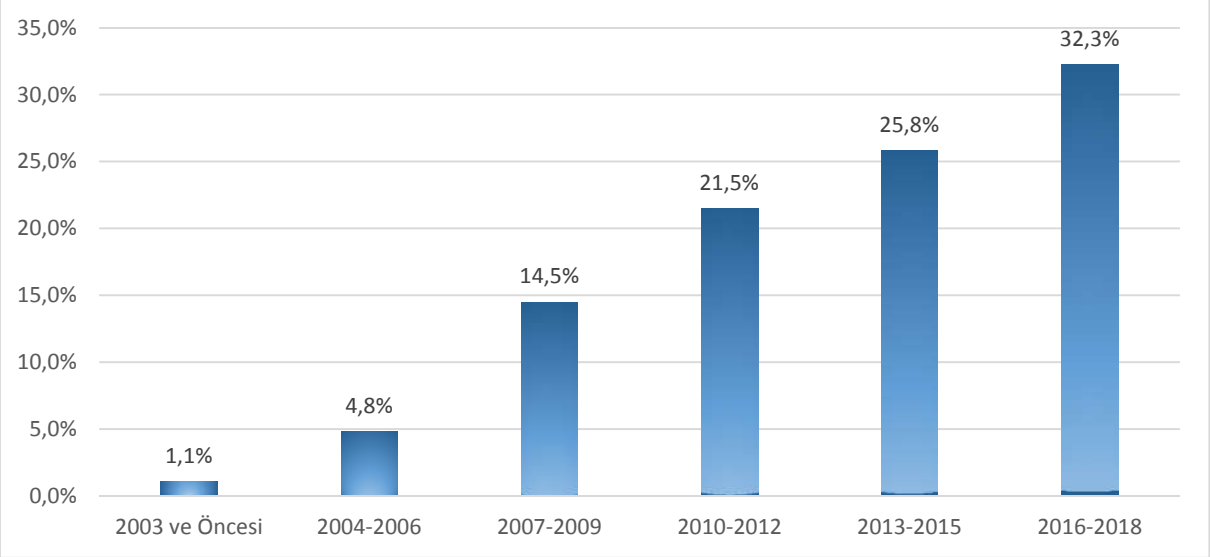
Sezgisellerde ise yerel arama, tabu arama gibi sezgisellerin kullanıldığı görülmektedir. Lezame'da (2017) yerel arama ile problem çözülmüştür ve sonuçlar GA ile karşılaştırılmıştır. Ayrıca Salmeron vd. (2004) Benders tabanlı sezgisel modeller geliştirilmiştir. Yasaklama/Koruma modellerinde kullanılan bazı çözüm yöntemlerin Çizelge 2.2'deki gibidir.

Çizelge 2.2. Yasaklama/Koruma Modellerinde Kullanılan Bazı Çözüm Yöntemleri (DA (Defender-Attacker), AD (Attacker- Defender), DAD (Defender-Attacker-Defender))

Yazar	Yıl	Model seviyesi	Kullanılan Çözüm Yöntemi	Çözüm Türü
Alguacil vd.	2014	3 (DAD)	Tam Sayım (Implict Enumeration)	Kesin Çözüm Yönt.
Wu ve Conejo	2017	3 (DAD)	Yerel Arama	Sezgisel
Ghaffarinasaba ve Atayi	2018	2 (DA)	Tam Sayım (Implict Enumeration)	Kesin Çözüm Yönt.
Salmeron vd.	2004	2 (AD)	Bender Ayrıştırma Temelli Sezgisel	Sezgisel
Salmeron ve Wood	2015	2 (AD)	(1) Benders Ayrıştırması (2) Yeni bir sayma yöntemi	Kesin Çözüm Yönt.
Lezama J.M.L. vd.	2017	2 (AD)	Yerel Arama (GA ile karşılaştırılmıştır)	Sezgisel
Losade	2010	2 (DA)	(1) Ayrıştırma Algoritması (2) Bender Ayrıştırması	Kesin Çözüm Yönt.
Losade	2012/a	2 (DA)	(1) Lower level SubProblem (SP), (2) Benders Ayrıştırması, (3) SVI Tabanlı Ayrıştırma (D-SVI), (4) D-Bend and D-SVI Kombine Edildiği Melez Ayrıştırma (D-H)	Kesin Çözüm Yönt.
Losade	2012/b		Matematiksel Model	Kesin Çözüm Yönt.
Aksen ve Aras	2011	2 (DA)	(1) Tabu Arama (2) Konum ve Koruma Kararlarının Ayrıldığı Sıralı Yöntem (Sequential method)	Sezgisel
Aksen vd.	2014	2 (AD)	(1) İlerici Şebeke Arama Yöntemi (progressive grid search)(Çok büyük prob.için) (2) Çok Başlangıçlı Simpleks Sezgisel (multi-start simplex search heuristic) (Büyük prob. için)	Sezgisel
Jing J.G. Vd.	2015	3 (DAD)	İç İçe Geçmiş Değişken Komşu Arama Yöntemi (nested variable neighborhood search method)	Sezgisel
Sacaparra vd.	2008	2 (DA)	Tam Sayım (Implict Enumeration)	Kesin Çözüm Yönt.
Ramamoorthy	2018	2 (AD)	Benders Ayrıştırması	Kesin Çözüm Yönt.

Literatürde yapılan çalışmaların ilerlemesine Şekil 2.1'de baktığımızda, alana yoğun bir ilginin olduğu görülmektedir. Problemlerin farklı yönleriyle ele alındığı ve gerçek hayat problemlerini daha iyi açıklayabilmek için çeşitli iyileştirmeler ile modellerin geliştirildiği ve çeşitlendirildiği görülmektedir. Aksen (2010), RIMF (Scaparra, 2008) modelini temel alarak kapasite genişletme maliyetini modele dahil etmiştir. Korunacak tesislerin sayısını ise

bütçe kısıtı ile belirlemiştir. Keçici (2012) yeni tesis açma maliyeti, yer değiştirme maliyeti ve koruma maliyetlerini göz önüne almıştır, sınırlı bir bütçe ile saldırı sonrası toplam hizmet kapsamını enbüyüklemiştir. Zhu (2013) yine RIMF modelini temel alarak, bir tesisin korunmuş olmasının, bir saldırıya karşı kesin olarak koruma sağlamayacağını ortaya koymuş ve koruma başarısını olasılıklı olarak modele dahil etmiştir. Losada (2010, 2012-a) yaptığı çalışmalarda zaman periyodlarını göz önüne almıştır ve zarar gören tesislerin yeniden düzelme süreleri modele dahil edilmiştir. Losada (2012-b) yaptığı çalışmada bozulma yoğunluk seviyelerinin belirsiz olduğunu göz önüne alarak stokastik bir yasaklama modeli önermiştir. Hangi tesislerin yasaklanacağı bilinemeyeceği gibi oluşacak bozulmanın yoğunluk seviyesinin belirsiz olacağı göz önüne alınmıştır. Liberatore (2011) da RIMF modelini temel alarak, yasaklanacak tesis sayısının kesin olmayacağını göz önüne alarak stokastik olarak yasaklama sayısını probleme dahil etmiştir. Aksen vd. (2014) çalışmalarında ise ilk kez tesislerin bir kapasitesi olduğunu ve dış kaynak kullanılabildiğini modele dahil etmişlerdir. Bu çalışmada saldırgan hizmet kesintisini enbüyüklemek istemektedir. Savunmacı (takipçisi) ise tüm müşterilerin talebini karşılamaktan sorumludur ve en kötü durumdaki saldırı sonrasında toplam talebin ağırlıklı taşıma maliyeti ve dış kaynak kullanım maliyetini en aza indirmeyi amaçlar. Jian vd (2015) üç aşamalı bir model geliştirdikleri çalışmalarında, problemde koruma amacı ile ayrılan kaynakların etkinliğini ölçmeyi amaçlamıştır. Bu alanda yapılan çalışmaların yıllara göre artarak devam ettiği görülmektedir.



Şekil 2.1. Yasaklama/Koruma Modeli Çalışmalarının Yıllara Göre Dağılımı

Daha önceki çalışmalar göz önüne alındığında saldırı tipinin belli bir sayıda ve tek tip olacağının varsayıldığı görülmektedir. Saldırı tiplerinin ve koruma tiplerinin de farklı olabileceğinin göz önüne alınmadığı görülmektedir. Gerçek hayat problemlerine bakıldığında ise saldırı tipinin ve bu saldırıya karşı alınan önlemin tek tip olmadığı açıktır. Örneğin bir terörist gurubun bir tesisi hedef alarak yapabileceği saldırılara bakıldığında birbirinden tamamen farklı saldırı tipleri mevcuttur. Günümüzdeki teknolojik gelişmeler ise bu çeşitliliği arttırmaktadır. Bir tesise canlı bomba, bombalı araç, uzaktan roket atışı, drone ile saldırı, siber saldırılar vb. farklı saldırı çeşitleri ile saldırılabilir.

Hagelstam ve Narinen'in (2018) NATO dergisinde yayınlanan "Hibrit tehditlere mücadele için işbirliği" başlıklı yazıda NATO'da hibrit tehditlere karşı yapılan çalışmalar paylaşılmıştır. Bu yazıda Hibrit tehditler, sürekli değişen ve çok çeşitli araçları olan tehditler olarak tanımlanmıştır. Kullanılan araçlar ise sahte sosyal medya profillerinden, sofistike siber saldırılardan, askeri güçlere kadar aradaki her şeyin kullanmasını kapsamaktadır. Bu araçlar, bizim çalışmamızda da belirttiğimiz gibi, hedefin niteliğine ve istenen sonuca bağlı olarak ayrı ayrı veya kombinasyon halinde kullanılabileceği yazıda belirtilmiştir.

Bu çalışmada yine Hibrit tehditlerin, farklı aktörlerin ve faaliyetlerin sinerjisinden yararlandığı gibi, bizimde hibrit savunmamızda aynı şeyi yapmamız gerektiği belirtilmektedir. Bu kapsamda 2016 yılından beri, NATO ve AB'nin hibrit tehditlerle mücadelede işbirliği konusu öncelikli bir konu olarak tanımlamaktadır (Hegelstam ve Narinen, 2018).

Aynı şekilde bu farklı çeşitteki saldırılara karşı alınacak önlemlerde birbirinden farklıdır ve farklı maliyetlere sahiptir. Bu çalışma çerçevesinde ise öncelikle saldırı tiplerinin tek tip olmayacağı, birbirinden farklı olacağı göz önüne alınmıştır. Ayrıca koruma tiplerinin de farklı olabileceği, bazı koruma şekillerinin birden fazla saldırı tipini önleyebileceği göz önüne alınmıştır ve **"İki Seviyeli Çoklu Saldırı Tipli Yasaklama/Koruma Modeli"** önerilmiştir. Yine yapılan çalışmaların bir kısmında hizmet sağlayan tesislerin kapasitelerinin sınırsız olduğu ya da kapasitesinin arttırılabileceği varsayılmaktadır. Önerilen ikinci model, "**Arz Talep Dengesi Bozma Amaçlı İki Seviyeli Çoklu Saldırı Tipli Yasaklama/ Koruma Modeli"**'nde ise hizmet sağlayan tesislerin bir kapasitesinin olduğu modele dahil edilmiştir ve saldırganın kapasite ile talep arasındaki

dengeyi bozmayı öncelikle hedefleyeceği göz önüne alınmıştır. Terör guruplarının hizmetleri kesintiye uğratma amacı ile yaptıkları tesis saldırılarına bakıldığında öncelikli amacın tesise zarar vermekten çok toplum üzerinde infial yaratmak olduğu görülmektedir. Bu amaç, önerilen ikinci modele eklenmiştir. Bu yeni amaçla birlikte Scaparra'da (2008) sunulan iki seviyeli RIMF modeli temel alınarak ve ilk modelde bazı değişikler yapılarak tekrar modellenmiştir.

3. MATERYAL VE YÖNTEM

Çalışmada Scaparra (2008)'nın iki seviyeli RIMF modeli temel alınarak, farklı saldırı ve koruma tiplerini göz önüne alan iki seviyeli R_eIMF modeli geliştirilmiştir. Bölüm 4.2'de ise amacın, savunan için öncelikle arz talep dengesini korumak, saldırgan için ise bozmak olduğu "Arz Talep Dengesi Bozma Amaçlı İki Seviyeli Çoklu Saldırı Tipli Yasaklama/ Koruma Modeli ($SD - R_e$IMF)" geliştirilmiştir..

Çalışmanın devamında problem çerçevesinin daraltılması için kritik sistemler içerisinden elektrik şebekeleri seçilmiştir. Elektrik şebekelerine özel kısıtlar göz önüne alınarak Bölüm 5.3'de, saldırı tipleri katkısı göz önüne alınarak, "Üç Seviyeli Çoklu Saldırı Tipli Elektrik Şebekesi Koruma Modeli" geliştirilmiştir. Bu modellerin çözümü için ise tüm olası koruma ve yasaklama kombinasyonlarını değerlendiren bir çözüm yaklaşımı geliştirilmiştir.

İki ve üç seviyeli modellerin literatürde NP-zor problemler olduğu gösterilmektedir. Bölüm 5.5'de ele aldığımız problemin karmaşıklık analizi yapılarak NP-zor bir problem olduğu gösterilmiştir. Problemde ele alınan bileşenlerin sayısındaki artış polinom büyümeye sebep olurken tip sayısında ki artışın üssel büyümeye sebep olduğu bu bölümde gösterilmiştir. NP-zor problemin çözümü için Genetik algoritmalar tabanlı iki sezgisel algoritma geliştirilmiştir. Bunlar başlangıç popülasyonun rassal belirlendiği r_GA ve başlangıç popülasyonun kural tabanlı olarak belirlendiği k_GA sezgiselidir. Bu algoritmaların eniyi performansı gösterdiği parametre düzeyleri belirlenmesi için deney tasarımı yapılmıştır. Bu iki algoritmanın bir sistemin korunmasında koruma kaynaklarının yoğunluğuna göre farklı durumlar için gösterdikleri performanslar yedinci bölümde istatistiksel olarak değerlendirilmiştir.

İlgili çözüm yaklaşımları Python 2.6.1 yazılımı ile kodlanmıştır ve GAMS 24.0.2 ile etkileşimlidir. İstatistiksel analizler için Minitab 19.2020.1 paket programı kullanılmıştır. Kullanılan bilgisayarın teknik özellikleri ise 8 GB RAM, Intel ® Core ™ i7, 4770 CPU, 3.40 Ghz işlemci ve 64 bit işletim sistemidir.

4. KRİTİK TESİSLER İÇİN SALDIRAN VE SAVUNANIN OLDUĞU YASAKLAMA/KORUMA PROBLEMLERİ

4.1. İki Seviyeli Çoklu Saldırı Tipli Yasaklama/Koruma Modeli (Yeni Model 1)

Yasaklama/koruma problemlerinde yasaklanmasına/korunmasına çalışılan farklı hizmet türleri ele alınabilir. Bu tez çerçevesinde elektrik hizmeti ele alınmıştır. Ancak bu bölümde genel olarak hizmet sunan bir tesis (kritik tesis) ve müşterinin (talep noktası) olduğu, tesisin yasaklanma/korunma durumunun göz önüne alındığı bir model önerilmiştir. Her hizmet türünün özel kısıtları olmakla birlikte, burada açıklananlar, doğalgaz sistemleri, silah depoları, acil servisler, ticari tesisler, su sistemleri vb. hizmet sunan ve bu hizmeti alan müşterilerin olduğu tüm sistemler için geçerli olacaktır.

Çözmeyi amaçladığımız problemde kritik tesisler aracılığı ile belirli bir bölgeye (talep noktası) hizmet sunulmaktadır. Sunulan hizmetin önemi nedeni ile talep noktalarının ihtiyacı olan hizmet kesintisiz olarak sunulmalıdır. Kritik tesislere yapılan saldırılar ile bu noktalara sunulan hizmet kesintiye uğrayabilir. Saldırganların kritik tesisleri hedef almaları ve bunlardan yeterli korunmaya sahip olmayanlarını devre dışı bırakacakları göz önüne alınacaktır. Kritik tesisin faaliyetlerini yerine getiremeyecek şekilde devre dışı bırakılması, durdurulması yasaklama olarak ele alınacaktır.

Şekil 4.1'de sunulan Scaparra (2008), iki seviyeli RIMF modeli temel alınarak iki seviyeli R_eIMF modeli geliştirilmiştir. Temel alınan RIMF modelinde *r* ile verilen değer yasaklanabilecek tesis sayısını ifade etmektedir.

Scaparra (2008) RIMF (r-interdiction median problem with fortification) Modeli:

Karar Değişkenleri;

$z_j = \begin{cases} 1, & j\ kritik\ tesisi\ \ korunursa \\ 0, & d.d. \end{cases}$

$s_j = \begin{cases} 1, & j\ kritik\ tesisi\ yasaklanırsa \\ 0, & d.d. \end{cases}$

$x_{ij} = \begin{cases} 1, yasaklama\ sonrası\ i\ talep\ noktası \\ \quad j\ kritik\ tesisinden\ hizmet\ alırsa \\ 0, \quad d.d. \end{cases}$

$T_{ij} = \{k \in J \text{ I } k \neq j\ ve\ d_{ik} > d_{ij}\}$ T_{ij} , j tesisi dışında kalan mevcut tesislerin setidir ve bu küme içindeki tüm tesislerin *i* talep noktasına hizmet sunma maliyeti, *j* kritik tesisinin *i* talep noktasına hizmet sunma maliyetinden fazladır.

r = yasaklama kaynak kapasitesi

a_i = i. talep noktasının talep miktarı

d_{ij} = i. talep noktası ile j. kritik tesis arası mesafe

Amaç Fonksiyonu (Üst Seviye)

$$\text{enk}\, H(z) \tag{4.1}$$

Kısıtları altında

$$\sum_{J \epsilon J} z_j = q \tag{4.2}$$

$$z_j \in \{0,1\} \quad \forall j \in J \tag{4.3}$$

Öyleki (Alt Seviye)

$$H(z) = \text{enb} \sum_{i \in I} \sum_{j \in J} a_i d_{ij} x_{ij} \tag{4.4}$$

Kısıtları altında

$$\sum_{j \in J} x_{ij} = 1 \qquad \forall i \in I \tag{4.5}$$

$$\sum_{j \in J} s_j = r \tag{4.6}$$

$$\sum_{k \in T_{ij}} x_{ik} \leq s_j \qquad \forall j \in J, \forall i \in I \tag{4.7}$$

$$s_j \leq 1 - z_j \qquad \forall j \in J \tag{4.8}$$

$$s_j \in \{0,1\} \qquad \forall j \in J \tag{4.9}$$

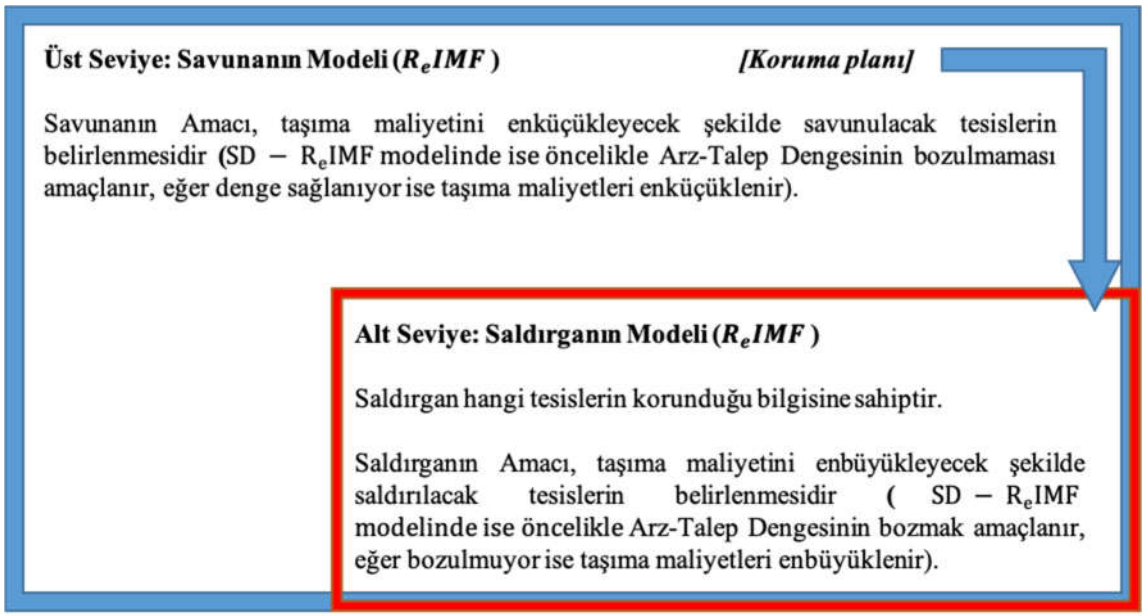
$$x_{ij} \in \{0,1\} \qquad \forall i \in I, \forall j \in J \tag{4.10}$$

(4.1) İki seviyeli modelin üst seviye (savunanın) amaç fonksiyonudur. p kadar tesisten r kadarı yasaklandığında ağırlıklı maliyeti enküçüklemeyi amaçlar.
(4.4) İki seviyeli modelin alt seviye (saldırganın) amaç fonksiyonudur. p kadar tesisten r kadarı yasaklandığında ağırlıklı maliyeti enbüyüklemeyi amaçlar.
(4.2) q kadar tesise tahkim yapılmasını sağlar.
(4.5) Her talep noktasının saldırı sonrası bir tesisten hizmet almasını sağlar.
(4.6) r kadar tesisin yasaklanmasını sağlar.
(4.7) i. tesisin, yasaklanan j tesislerinin dışında kalan en yakın tesisten hizmet almasını sağlar
(4.8) Korunan bir tesisin yasaklanmasını önler.
(4.3), (4.9) ve (4.10) karar değişkenlerinin işaret kısıtlarıdır.

Şekil 4.1. Scaparra (2008) RIMF Modeli

Yukarıda detayları verilen RIMF modelinde ve literatürdeki diğer modellerde saldırganın tek tip saldırı yapabileceği varsayılmaktadır. Gerçekte ise saldırgan farklı tiplerde saldırılar yapabilme kabiliyetine sahiptir. Bu saldırı tiplerini göz önüne alarak tahkim çeşitleri de farklılık göstermelidir. Bu farklılık göz önüne alınarak modelde yasaklama çeşidine göre koruma çeşitlerinin de farklılaşacağı göz önünde bulundurulmuştur.

Saldırı ve koruma tipleri göz önüne alınarak geliştirilen "İki Seviyeli Çoklu Saldırı Tipli Yasaklama/Koruma Modeli (R_eIMF) (Yeni Model 1) " modelinin üst seviyesi savunan seviyesidir ve koruma planı üretecek olan seviyedir. Alt seviye ise saldırganın seviyesidir. Saldırgan hangi tesislerin korunduğu bilgisine sahiptir ve korunan bu tesislere saldırı yapmamaktadır. Stackelberg oyun teorisine dayanan bu model Şekil 4.2'deki gibi savunan-saldıran (DA Defender-Attacker) şeklinde oluşturulmuştur. Bir sonraki bölümde sunulan "Arz Talep Dengesi Bozma Amaçlı İki Seviyeli Çoklu Saldırı Tipli Yasaklama/ Koruma Modeli (SD – R_eIMF) (Yeni Model 2)" modelinde ise savunanın amacı öncelikle arz talep dengesininin bozulmasını engellemektir. Saldırganın amacı ise bu dengeyi bozmaktır.



Şekil 4.2. İki Seviyeli Yeni R_eIMF ve SD – R_eIMF Koruma Modelleri

Varsayımlar:

1. Saldırgan, farklı tiplerde ve belirli sayılarda saldırı yapabilme yeteneği ve donanımına sahiptir ve bu bilinmektedir.
2. Saldırgan saldırı anında (1)'de belirtilen saldırı tiplerinin belli sayıdaki bir karmasını kullanacaktır.
3. Saldırgan hangi tesislerin korunduğunu bilmektedir.
4. Saldırganın farklı çeşitteki saldırı tiplerinde kaçar saldırı yapabilme kabiliyetine sahip olduğu bilinmektedir.
5. Bir saldırı periyodunda bu saldırılarından toplamda kaç adedini yapabileceği de bilinmektedir.
6. Saldırgan hangi tesislerin hangi saldırı tiplerine karşı korunduğunu bilmektedir ve bu tesislere ilgili saldırı tipi ile saldırmaz.
7. Bir tesis yasaklandığında hizmet alınacak alternatif fakat daha yüksek maliyetli bir tesis her zaman vardır.

İndisler :
$i \in I$: Talep noktası
$j \in J$: Kritik tesis
$e \in E$: Saldırı tipi
$f \in F$: Koruma tipi

Kümeler :
I : Talep noktaları $I = \{i|i = 1,2,..n\}$
J : Kritik tesisler $J = \{j|j = 1,2,..p\}$
E : Saldırı tipleri $E = \{e|e = 1,2,..g\}$
F : Koruma tipleri $F = \{f|f = 1,2,..t\}$

Parametreler:

d_{ij} = Kritik tesis j'den talep noktası i'ye sunulan hizmetin birim maliyeti

a_i = *i* talep noktasının talebi

b_{jf} = *j* kritik tesisinin f tipi ile tahkim (korunma) maliyeti

b_{tot} = Savunma için ayrılan toplam bütçe

R= Saldırganların yasaklayabileceği toplam kritik tesis sayısı

r_e = Saldırganın e. saldırı tipinde yapabileceği toplam yasaklama sayısı

L_{ijk} = *k* kritik tesisinin, *i* talep noktasına olan mesafesi, *j* kritik tesisinin i talep noktasına olan mesafesinden büyükse 1, eşit ve küçükse 0'dır. Bu parametre RIMF orijinal modelinde ($T_{ij} = \{k \in J \text{ I } k \neq j \; ve \; d_{ik} > d_{ij}\}\; T_{ij}$,*j* kritik tesisi dışında kalan mevcut kritik tesislerin setidir ve bu küme içindeki tüm kritik tesislerin *i* kritik tesisine hizmet sunma maliyeti, *j* kritik tesisinin *i* talep noktasına hizmet sunma maliyetinden fazladır) olarak açıklanan T_{ij} parametresi yerine ilave edilmiştir ve eğer bir tesis yasaklanırsa o tesisten hizmet alan talep noktasının en yakın tesisten hizmet almasını sağlayan (4.19) no.'lu kısıt bu değişiklik göz önüne alınarak revize edilmiş ve (4.20) numaralı kısıt oluşturulmuştur.

K_{ef} = e saldırı tipi *f* koruma tipi ile korunabiliyor ise 1, koruma sağlayamıyor ise 0 değerini almaktadır. (K_{ef} parametresi ile farklı koruma tipleri göz önüne alınmıştır ve (4.12), (4.13) ve (4.18) no.'lu kısıtlar koruma tipi farklılıklarını sağlayacak şekilde yeniden düzenlenmiştir).

Yasaklama ve koruma tiplerinin farklı olduğu göz önüne alınarak, yeni parametre tanımlamaları ve revize edilen kısıtlar doğrultusunda model aşağıdaki gibi elde edilmektedir.

Karar Değişkenleri;

$$z_{jf} = \begin{cases} 1, & j \; kritik \; tesisi \; f \; koruma \; tipi \; ile \; korunursa \\ 0, & d.d. \end{cases}$$

$$s_{je} = \begin{cases} 1, & j \; kritik \; tesisi \; e \; saldırı \; tipi \; ile \; yasaklanırsa \\ 0, & d.d. \end{cases}$$

$$x_{ij} = \begin{cases} 1, & yasaklama \; sonrası \; i \; talep \; noktası \; j \; kritik \; tesisinden \; hizmet \; alırsa \\ 0, & d.d. \end{cases}$$

Amaç Fonksiyonu

$$\text{enk}\, H(z) \tag{4.11}$$

Kısıtları altında

$$\sum_{j \in J} \sum_{f \in F} b_{jf} z_{jf} \leq b_{tot} \tag{4.12}$$

$$z_{jf} \in \{0,1\} \quad \forall j \in J, \forall f \in f \tag{4.13}$$

Yerine

$$H(z) = \text{enb} \sum_{i \in I} \sum_{j \in J} a_i d_{ij} x_{ij} \tag{4.14}$$

Kısıtları altında

$$\sum_{j \in J} x_{ij} = 1 \qquad \forall i \in I \tag{4.15}$$

$$\sum_{j \in J} s_{je} \leq r_e \qquad \forall e \in E \tag{4.16}$$

$$\sum_{j \in J} \sum_{e \in E} s_{je} = R \tag{4.17}$$

$$s_{je} \leq 1 - K_{ef} \cdot z_{jf} \qquad \forall j \in J, \quad \forall e \in E, \quad \forall f \in F \tag{4.18}$$

$$\sum_{k} L_{ijk} \cdot x_{ik} \leq \sum_{e} s_{je} \qquad \forall j \in J, \quad \forall i \in I \tag{4.19}$$

$$\sum_{i} x_{ij} \leq n - n \cdot s_{je} \qquad \forall j \in J, \quad \forall e \in E, \quad \forall i \in I \tag{4.20}$$

$$s_{je} \in \{0,1\} \qquad \forall j \in J, \forall e \in E \tag{4.21}$$

$$x_{ij} \in \{0,1\} \qquad \forall i \in I, \forall j \in J \tag{4.22}$$

(4.11) İki seviyeli modelin üst seviye amaç fonksiyonudur. p kadar tesisten r kadarı yasaklandığında ağırlıklı maliyeti enküçüklemeyi amaçlar.

(4.12) *p* kadar tesisten savunma için ayrılmış olan bütçe kadar tesise tahkim yapılmasını sağlar.

(4.14) İki seviyeli modelin alt seviye amaç fonksiyonudur. *p* kadar tesisten *r* kadarı yasaklandığında ağırlıklı maliyeti enbüyüklemeyi amaçlar.

(4.15) Her talep noktasının saldırı sonrası bir tesisten hizmet almasını sağlar.

(4.16) *e* saldırı tipi ile en fazla yasaklanabilecek tesis sayısını sınırlar.

(4.17) *R* kadar tesisin yasaklanmasını sağlar.

(4.18) Korunan bir tesisin yasaklanmasını önler.

(4.19) Yasaklama sonrası, yasaklı bir tesise talep noktasının atanmasını önler. Yasaklama sonrası yasaklanan tesisten sonraki en yakın tesise *i* talep noktasının atanmasını sağlar.

(4.20) no.'lu kısıt ise yeni eklenmiştir. Bu kısıt, eğer bir *j* santrali herhangi bir *e* saldırı tipi ile yasaklanırsa o tesisten hizmet alınmasını engellemektedir

(4.13), (4.21) ve (4.22) karar değişkenlerinin işaret kısıtlarıdır.

4.2. Arz Talep Dengesi Bozma Amaçlı İki Seviyeli Çoklu Saldırı Tipli Yasaklama/ Koruma Modeli (Yeni Model 2)

Kritik tesislerden sunulan hizmet, talep noktalarına iletilmektedir. Bu tesislerden bazıları sistem dışı kaldığında kalan tesisler talep noktalarının ihtiyaçlarını farklı maliyetlerle karşılamaya devam ederler.

İlk modelde kritik tesislerin kapasitelerinin sınırsız olduğu varsayılmaktadır. Gerçekte ise her tesisin bir kapasitesi vardır. Saldırgan tarafından gerçekleştirilen yasaklamalar ile kalan tesislerin kapasiteleri toplamı, talebi karşılamaya yetmeyebilir.

Saldıranın bu durumda iki amacı vardır. İlk amacı klasik r-interdiction modellerde olduğu gibi saldırı sonrası en yüksek maliyetle taleplerin karşılandığı durumu sağlamak. Diğer amacı ise yaptığı saldırılar ile talebi karşılanamayan noktaların oluşmasıdır. Bu durum bazı bölgelerin hizmet alamaz duruma gelmesine sebep olacaktır.

Terörizm belirli amaçlar için bir araç olarak kamunun provoke edilmesi, topluma korku salma ve yıldırmak amacıyla tasarlanan eylemlerin kasıtlı ve sistematik kullanımı olarak tanımlanmaktadır. Terör saldırılarında, toplum üzerinde büyük bir infial yaratma hedefi vardır. Bu açıdan ele aldığımızda bir terör saldırısında öncelikle arz talep dengesizliğini oluşturmak daha cazip görülmektedir.

Bu bağlamda modele saldırganın öncelikle arz talep dengesizliğine en fazla sebep olacak şekilde saldırılarını organize edeceği gerçeği eklenmek istenmiştir. Saldırgan eğer böyle bir dengesizliğe sebep olabiliyor ise önce bunu tercih edecektir. Ancak olamıyorsa maliyeti en fazla oluşturacak şekilde saldıracaktır.

Varsayımlar:

1. Saldırgan, farklı tiplerde ve belirli sayılarda saldırı yapabilme yeteneği ve donanımına sahiptir ve bu bilinmektedir.
2. Ancak bir saldırı anında bunların belli sayıdaki bir karmasını kullanacaktır.
3. Saldırgan hangi tesislerin korunduğunu iyi bilmektedir.
4. Saldırganın farklı çeşitteki saldırı tiplerinde kaçar saldırı yapabilme kabiliyetine sahip olduğu bilinmektedir.
5. Bir saldırı periyodunda bu saldırılarından toplamda kaç adedini yapabileceği de bilinmektedir.
6. Saldırgan hangi tesislerin hangi saldırı tiplerine karşı korunduğunu bilmektedir ve bu tesislere ilgili saldırı tipi ile saldırmaz.
7. Saldırgan öncelikle tesislerin kapasitesini talep miktarının altına düşürmeyi amaçlar, bunu yapamaz ise maliyeti enbüyüklemeyi hedefler.

İndisler :
i ∈ I : Talep noktası
j ∈ J : Kritik tesis
e ∈ E : Saldırı tipi
f ∈ F : Koruma tipi

Kümeler :
I : Talep noktaları $I = \{i|i = 1,2,..n\}$
J : Kritik tesisler $J = \{j|j = 1,2,..p\}$
E : Saldırı tipleri $E = \{e|e = 1,2,..g\}$
F : Koruma tipleri $F = \{f|f = 1,2,..t\}$

Parametreler:

d_{ij} = Kritik tesis j'den talep noktası i'ye sunulan hizmetin birim maliyeti

a_i = *i* talep noktasının talebi

b_{jf} = *j* kritik tesisinin *f* tipi ile tahkim (korunma) maliyeti

b_{tot} = Savunma için ayrılan toplam bütçe

R= Saldırganların yasaklayabileceği toplam kritik tesis sayısı

r_e = Saldırganın e. saldırı tipinde yapabileceği toplam yasaklama sayısı

L_{ijk} = *k* kritik tesisinin, *i* talep noktasına olan mesafesi, *j* kritik tesisinin i talep noktasına olan mesafesinden büyükse 1, eşit ve küçükse 0'dır. Bu parametre RIMF orijinal modelinde ($T_{ij} = \{k \in J \text{ I } k \neq j \; ve \; d_{ik} > d_{ij}\}$ T_{ij} , *j* kritik tesisi dışında kalan mevcut kritik tesislerin setidir ve bu küme içindeki tüm kritik tesislerin *i* kritik tesisine hizmet sunma maliyeti, *j* kritik tesisinin *i* talep noktasına hizmet sunma maliyetinden fazladır) olarak açıklanan T_{ij} parametresi yerine ilave edilmiştir ve eğer bir tesis yasaklanırsa o tesisten hizmet alan talep noktasının en yakın tesisten hizmet almasını sağlayan (4.31) no.'lu kısıt bu değişiklik göz önüne alınarak revize edilmiş ve (4.32) numaralı kısıt oluşturulmuştur.

$tkap_j$ = *j* kritik tesisinin kapasitesi

toptalep = Korunması planlanan bölgenin ihtiyacı olan toplam talep miktarı

Karar Değişkenleri;

$$z_{jf} = \begin{cases} 1, & j \; kritik \; tesisi \; f \; koruma \; tipi \; ile \; korunursa \\ 0, & d.d. \end{cases}$$

$$s_{je} = \begin{cases} 1, & j \; kritik \; tesisi \; e \; saldırı \; tipi \; ile \; yasaklanırsa \\ 0, & d.d. \end{cases}$$

$$S_j = \begin{cases} 1\,, & j\ kritik\ tesisi\ yasaklanırsa \\ 0, & d.d. \end{cases}$$

$$x_{ij} = \begin{cases} 1\,, yasaklama\ sonrası\ i\ talep\ merkezi\ j\ santralinden\ hizmet\ alırsa \\ 0, & d.d. \end{cases}$$

$$y = \begin{cases} 1\,, yasaklama\ sonrası\ kalan\ toplam\ kapasite\ yetersiz\ ise \\ 0, & d.d. \end{cases}$$

Amaç Fonksiyonu

$$\text{enk}\, H(z) \tag{4.23}$$

Kısıtları altında

$$\sum_{j\in J}\sum_{f\in F} b_{jf} z_{jf} \le b_{tot} \tag{4.24}$$

$$z_{jf} \in \{0,1\} \quad \forall j \in J, \forall f \in F \tag{4.25}$$

Yerine

$$H(z) = \text{enb} \sum_{i\in I}\sum_{j\in J} a_i d_{ij} x_{ij} + M.y \tag{4.26}$$

Kısıtları altında

$$\sum_{j\in J} x_{ij} \le 1 \qquad \forall i \in I \tag{4.27}$$

$$\sum_{j\in J} x_{ij} \ge 1 - y \qquad \forall i \in I \tag{4.28}$$

$$\sum_{j\in J} s_{je} \le r_e \qquad \forall e \in E \tag{4.29}$$

$$\sum_{j\in J}\sum_{e\in E} s_{je} = R \tag{4.30}$$

$$s_{je} \le 1 - K_{ef}\,.z_{jf} \qquad \forall j \in J, \quad \forall e \in E, \quad \forall f \in F \tag{4.31}$$

$$\sum_{k} L_{ijk}.x_{ik} \le \sum_{e} s_{je} \qquad \forall j \in J\,, \quad \forall i \in I \tag{4.32}$$

$$\sum_{i\in I} a_i = toptalep \tag{4.33}$$

$$\frac{\sum_e s_{je}}{1+\sum_e s_{je}} + \frac{1}{2} \ge S_j \ge \frac{\sum_e s_{je}}{1+\sum_e s_{je}} \qquad \forall j \in J \tag{4.34}$$

$$\sum_{j\in J} tkap_j - \sum_{j\in J} S_j * tkap_j < toptalep + M(1-y) \tag{4.35}$$

$$\sum_{j\in J} tkap_j - \sum_{j\in J} S_j * tkap_j \ge toptalep - M.y \tag{4.36}$$

$$s_{je} \in \{0,1\} \qquad \forall j \in J, \forall e \in E \tag{4.37}$$

$$x_{ij} \in \{0,1\} \qquad \forall i \in I, \forall j \in J \tag{4.38}$$

$$S_j \in \{0,1\} \qquad \forall j \in J \tag{4.39}$$

$$y \in \{0,1\} \tag{4.40}$$

Yukarıda önerilen modele y değişkeni eklenmiştir. Eğer saldırgan elindeki kaynaklar ile tesislerin sunduğu hizmeti talep miktarının altına düşürürse y=1 değerini alır. Saldırganın öncelikli amacı budur. Amaç fonksiyonunda y değişkenini kaysayısı büyük M'dir. Bu y değişkeninin modelde önceliklenmesini sağlar. Saldırgan tesis kapasitesini talep miktarının altına düşüremiyorsa y=0 olur ve amaç fonksiyonu ağırlıklı maliyeti enbüklemeyi amaçlar.

(4.23) İki seviyeli modelin üst seviye amaç fonksiyonudur. Öncelikle arz miktarının talep miktarının altına düşmemesini amaçlar, bu sağlandığında, *p* kadar tesisten *r* kadarı yasaklandığında ağırlıklı maliyeti enküçüklemeyi amaçlar.
(4.24) *p* kadar tesisten savunma için ayrılmış olan bütçe kadar tesise tahkim yapılmasını sağlar.
(4.26) İki seviyeli modelin alt seviye amaç fonksiyonudur. Öncelikle arz miktarının talep miktarının altına düşmesini amaçlar, bu sağlanamaz ise *p* kadar tesisten *r* kadarı yasaklandığında ağırlıklı maliyeti enbüyüklemeyi amaçlar.
(4.27) Her talep noktasının saldırı sonrası en fazla bir tesisten hizmet almasını sağlar.
(4.28) Saldırı sonrası eğer santral kapasitesi yeterli ise her talep noktasının bir santralden hizmet almasını sağlar. Eğer kapasite yetersiz ise *i.* talep noktasının talebi karşılanmayabilir.
(4.29) *e* saldırı tipi ile en fazla yasaklanabilecek tesis sayısını sınırlar.
(4.30) *R* kadar tesisin yasaklanmasını sağlar.
(4.31) Yasaklama sonrası, yasaklı bir tesise talep noktasının atanmasını önler. Yasaklama sonrası yasaklanan tesisten sonraki en yakın tesise atanmasını sağlar.
(4.32) Korunan bir tesisin yasaklanmasını önler.
(4.33) Toplam talebin değerini belirler.
(4.34) Bir santralin yasaklı olup olmadığını belirler
(4.35) ve (4.36) saldırı sonrası kalan santrallerin toplam kapasitesi, toplam talepten az ise y = 1, diğer durumda y = 0 olmasını sağlar.
(4.15), (4.36), (4.38) , (4.39) ve (4.40) karar değişkenlerinin işaret kısıtlarıdır.

Bu bölümde saldırı/koruma tipi eklentisi genel yasaklama/koruma modellerine yapılarak modellerdeki değişimler incelenmiştir. RIMF modelinde ve literatürdeki diğer modellerde saldırganın tek tip saldırı yapabileceği varsayılmaktadır. Gerçekte ise saldırgan farklı tiplerde saldırılar yapabilme kabiliyetine sahiptir. Bu saldırı tiplerini göz önüne alarak tahkim çeşitleri de farklılık göstermelidir. Bu farklılık göz önüne alınarak geliştirilen "İki

Seviyeli Çoklu Saldırı Tipli Yasaklama/Koruma Modeli (R_eIMF) (Yeni Model 1)" Stackelberg oyun teorisine dayanan bir model olarak savunan-saldıran (DA Defender-Attacker) şeklinde oluşturulmuş ve Bölüm 4.1'de sunulmuştur.

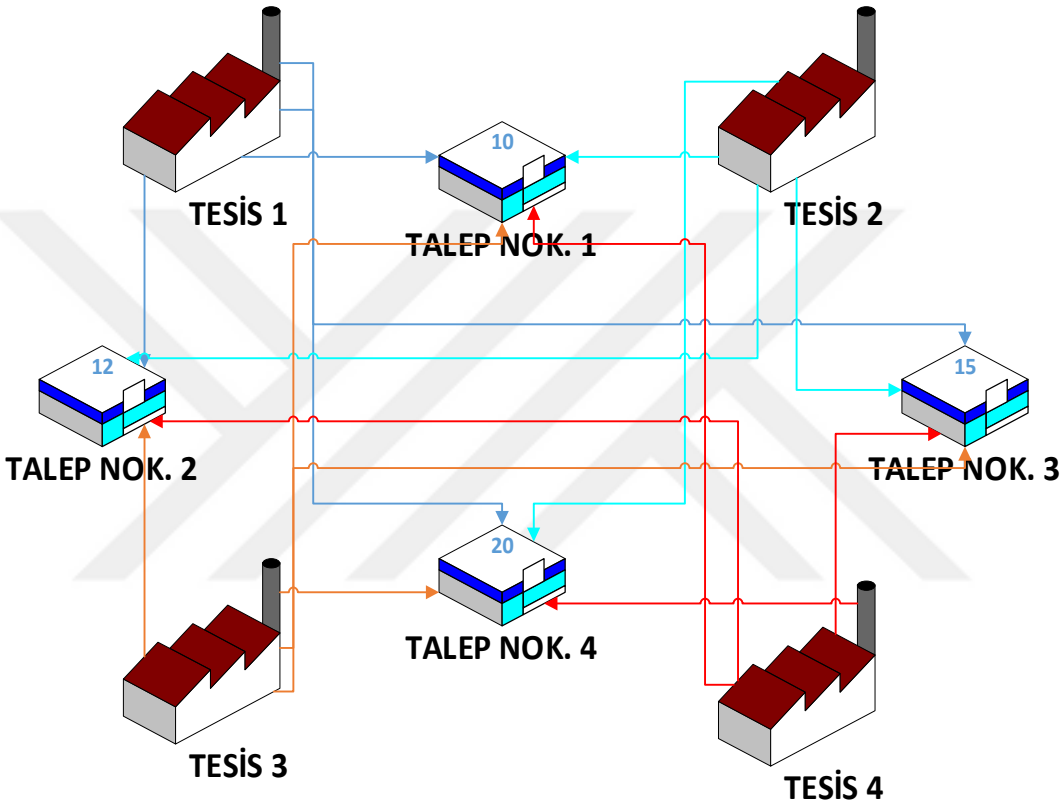
Bölüm 4.2'de ise saldırganın gerçekleştirdiği terör saldırılarının amaçlarından birinin yaptığı saldırılar ile talebi karşılanamayan noktaların oluşmasını sağlamak ve bu şekilde kamunun provoke edilmesi, topluma korku salma ve yıldırarak toplum üzerinde infial yaratma hedefi göz önüne alınmıştır. Bu hedef göz önüne alınarak, "Arz Talep Dengesi Bozma Amaçlı İki Seviyeli Çoklu Saldırı Tipli Yasaklama/ Koruma Modeli ($SD - R_e$IMF) (Yeni Model 2)" oluşturulmuştur, bu modelde ise amaç, savunan için öncelikle arz talep dengesini korumak, saldırgan için ise bozmaktır. Sunulan bu modeller genel hatları ile hizmet alan ve sunan kritik sektörlerin genelini açıklamak ve çerçeveyi oluşturmak üzere sunulmuştur.

Tezin bundan sonraki bölümünde, ele alınan genel problem sınırları daraltılmış daha spesifik bir konu olan elektrik şebeke sistemlerine yönelik modeller geliştirilmektedir. Bu problem ilgilendiği konu özelinde çok spesifik yeni parametreleri ve yeni kısıtları olduğu ortaya çıkmıştır, bağlı olarak da geliştirilen çözüm yaklaşımı ve modelin kendisi oldukça farklı bir yapıya bürünmüştür. Bu sebeple bundan sonra geliştirilen elektrik şebeke modeli ve çözüm yaklaşımları 5. Bölümde ve 6. Bölümde, gerçekleştirilen testler ve elde edilen sonuçlar 7. Bölümde topluca değerlendirilmektedir. Şuana kadar üzerinde çalışılan farklı saldırı ve koruma tiplerinin olduğu gelinen noktadaki modellerin ise bu nedenle test sonuçları hemen bu aşamada ele alınmıştır.

Bölüm 4.1 ve 4.2'de önerilen matematiksel modellerin çözümleri bir uygulama örneği ile 4.3'de sunulmaktadır. Modeller askeri ve kritik tesisler göz önüne alınarak tasarlanan modellerdir, ancak gerçek veri setlerine ulaşmak, özellikle konunun güvenlik ile ilgili olması nedeni ile güçtür. Bu sebeple veri türetme zorunluluğu bulunmaktadır.

4.3. Yeni Model 1 ve Yeni Model 2'nin Çözümleri

Önerilen Yeni Model 1 ve Yeni Model 2'yi test etmek için hizmet sunan dört kritik tesis (j), hizmet alan dört talep merkezi (i) ve iki farklı yasaklama (e) ve karşılık gelen iki farklı koruma tipinin (f) olduğu bir test problemi kullanılmıştır. Koruma kaynağı beş, saldırı kaynağı ise iki olarak alınan test problemi Şekil 4.3'de şematik olarak gösterilmiştir.



Şekil 4.3. Dört Tesis ve Dört Talep Noktasından Oluşan Örnek Problemin Şematik Gösterimi

Modeller askeri ve kritik tesisler göz önüne alınarak tasarlanmıştır. Ancak bu alanda gerçek veri setlerine ulaşmanın zorluğu nedeni ile veri türetilerek bu bölümde çözülen örnek problem oluşturulmuştur.

Probleme ilişkin parametreler Ek-A'de verilmiştir. Verilen örnek problemde kritik tesislerin kapasitelerinin sınırsız olarak kabul edildiği R_eIMF modeli ile elde edilen çözüm sonucu Çizelge 4.1'de verilmiştir. 5 farklı koruma planı için en iyi değer elde edilmiştir.

Çizelge 4.1. R_eIMF Modeli İçin Amaç Fonksiyonu Değeri En İyi Olan Koruma Planları

Plan No	Amaç Fonksiyonu Değeri	Korunan Tesislerin Bilgisi z_{jf}	Yasaklanan Tesislerin Bilgisi s_{je}	Yasaklama Sonrası Talep Noktalarının Hizmet Aldığı Tesis Bilgisi x_{ij}
1	742	$z_{11}, z_{12}, z_{22}, z_{31}, z_{32}$	s_{21}, s_{42}	$x_{13}, x_{21}, x_{33}, x_{43}$
2	742	$z_{11}, z_{12}, z_{21}, z_{31}, z_{32}$	s_{22}, s_{41}	$x_{13}, x_{21}, x_{33}, x_{43}$
3	742	$z_{11}, z_{12}, z_{31}, z_{32}, z_{41}$	s_{21}, s_{42}	$x_{13}, x_{21}, x_{33}, x_{43}$
4	742	$z_{11}, z_{12}, z_{31}, z_{32}, z_{42}$	s_{22}, s_{41}	$x_{13}, x_{21}, x_{33}, x_{43}$

1 no.'lu koruma planında 5 adet koruma kaynağının $z_{11}, z_{12}, z_{22}, z_{31}, z_{32}$ şeklinde tesisleri koruduğu görülmektedir. Bu koruma planına göre 1 ve 3 no.'lu tesisler iki saldırı tipine karşı korunmaktadır. Bu durum saldırganın 1 ve 3 no.'lu tesislere saldırmasını engellemektedir. 2 no.'lu tesis yalnızca 2 no.'lu saldırı tipine karşı (z_{22}) korunmaktadır. 4 no.'lu tesisin ise 1 ve 2 no.'lu saldırı tiplerine karşı korunmadığı görülmektedir. Saldırgan sadece s_{21}, s_{42} saldırı planı ile 2 ve 4 no.'lu tesislere saldırmıştır. Saldırgan 2 adet saldırı kaynağını da herhangi bir tesise saldırmak için kullanmamış ve hem 1 no.'lu hem de 4 no.'lu tesisi yasaklayacak şekilde saldırısını planlamıştır.

Plan 1'de verilen koruma planının, koruma ve saldırı planları sonucu olarak i talep noktaları $x_{13}, x_{21}, x_{33}, x_{43}$ şeklinde hizmet almaya devam etmişlerdir. 1 no.'lu talep noktası 3 no.'lu tesisten (x_{13}), 2 no.'lu talep noktası 1 no.'lu tesisten (x_{21}), 3 no.'lu talep noktası 3 no.'lu tesisten (x_{33}), 4 no.'lu talep noktası 3 no.'lu tesisten (x_{43}) hizmet alarak 742 birimlik bir maliyet oluşmuştur.

Ayrıca en iyi amaç fonksiyonu değerini veren tüm koruma planları birlikte incelendiğinde özellikle 1 ve 3 no.'lu tesisin, tüm koruma planlarında her iki saldırı tipine karşı da korunduğu görülmektedir ve böylece hangi tesislerin kritik olduğu konusunda bilgi alınmaktadır.

Arz talep dengesi bozmaya yönelik SD – R_eIMF modeli için ise Çizelge 4.2'deki parametreler modele eklenmiştir. M değeri 5000 olarak kabul edilmiştir.

Çizelge 4.2. Kritik Tesislerin Kapasiteleri ($tkap_j$)

j	1	2	3	4
Kapasite ($tkap_j$)	30	40	27	25

Tesislerin belirli bir kapasitesinin olduğunu göz önüne alan SD − R_eIMF modelinin çözüm sonuçları ise şu şekilde gerçekleşmiştir.

1 – 4 no.'lu tesislerin birlikte korunduğu ve 3 – 4 no.'lu tesislerin birlikte korunduğu durumlar için arz talep dengesinin bozulduğu görülmektedir. Ancak 1 – 2 , 1 – 3 , 2 – 3 ve 2 – 4 no.'lu tesis çiftlerinin birlikte korunduğu koruma planlarında arz talep dengesinin korunduğu ve makul değerler aldığı Çizelge 4.3'de görülmektedir. Bunlarda 1 – 3 no.'lu tesisin birlikte korunduğu koruma planlarının ise en iyi amaç değerini (748) almaktadır.

Çizelge 4.3. Kapasite Kısıtlı Örnek Problem için Olası Çözüm Değerleri

Korunan Tesisler	Çözüm Sonucu
1 ve 2 no.'lu tesisler korunursa	834
1 ve 3 no.'lu tesisler korunursa	748
1 ve 4 no.'lu tesisler korunursa	Arz-Talep Dengesi Bozuk
2 ve 3 no.'lu tesisler korunursa	844
2 ve 4 no.'lu tesisler korunursa	964
3 ve 4 no.'lu tesisler korunursa	Arz-Talep Dengesi Bozuk

SD − R_eIMF Modelinin çözümü ile elde edilen eniyi amaç değerine sahip koruma planları Çizelge 4.4'de verilmiştir. Buna göre yine 1 ve 3 no.'lu tesislerin her iki saldırı tipine karşı korunduğu görülmektedir (z_{11} , z_{12} , z_{31} , z_{32}). Örnek problemde kapasite kısıtlarının göz önüne alınması sebebi ile elde tesis – talep merkezi eşleştirmelerine bakıldığında ise 2 no.'lu talep merkezinin kapasite kısıtlarından kaynaklı olarak hem 1 no.'lu hem de 3 no.'lu tesisten hizmet aldığı görülmektedir (x_{21}, x_{23}).

Çizelge 4.4. SD − R_eIMF Modeli İçin Amaç Değeri en iyi olan Koruma Planları

Plan No	Amaç Fonksiyonu Değeri	Korunan Tesislerin Bilgisi z_{jf}	Yasaklanan Tesislerin Bilgisi s_{je}	Yasaklama Sonrası Talep Noktalarının Hizmet Aldığı Tesis Bilgisi x_{ij}
1	748	z_{11} , z_{12} , z_{31} , z_{32} , z_{41}	s_{21} , s_{42}	x_{13}, x_{21}, x_{23}, x_{33}, x_{41}
2	748	z_{11} , z_{12} , z_{31} , z_{32} , z_{42}	s_{22} , s_{41}	x_{13}, x_{21}, x_{23}, x_{33}, x_{41}
3	748	z_{11} , z_{12} , z_{21} , z_{31} , z_{32}	s_{22} , s_{41}	x_{13}, x_{21}, x_{23}, x_{33}, x_{41}
4	748	z_{11} , z_{12} , z_{22} , z_{31} , z_{32}	s_{21} , s_{42}	x_{13}, x_{21}, x_{23}, x_{33}, x_{41}

Model öncelikle arz talep dengesini bozulmasını engelleyecek şekilde koruma planlarını belirlemiş ve bu çerçevede en iyi amaç fonksiyonu değerini veren koruma planını seçmiştir.

5. ELEKTRİK ŞEBEKELERİNDE YASAKLAMA/KORUMA PROBLEMLERİ

Terörist saldırıların amaçlarından birinin toplumun önemli işlevlerinin devre dışı bırakılması olduğu önceki bölümlerde ifade edilmişti. Çalışmada özellikle elektrik sistemlerinin seçilmesi, pek çok kritik tesisin, sektörün ve birimin elektriğe olan ihtiyacıdır. Elektrik sistemlerine planlanan saldırılar ile farklı alanlarda toplumun ihtiyacı olan hizmetler engellenebilir. Bu durum, ekonomik bir etkinin yanında korku, endişe, hızlı bir şekilde toplumun ilgisini çekmek, normal yaşamı etkileyerek insanların psikolojilerinde travma yaratmak ile sonuçlanabilir.

Bilindiği gibi metro alt yapısı, internet altyapısı, hastane vb. tesisler, yanı sıra otoyol vb. pek çok birim doğrudan elektrik şebekesinin güvenliğinden etkilenir. Örneğin metro sisteminin elektriğinin kesilmesi önemli bir kesimin ulaşımını etkileyecektir ve bu durum ülke gündeminde önemli konulardan biri olacaktır. Ya da bir sanayi bölgesinin elektriksiz kalması ekonomik olarak olumsuzluklara sebep olacaktır. Bu nedenle problem alanı olarak elektrik şebekeleri seçilmiştir ve elektrik sistemleri kısıtları da göz önünde bulundurularak bu bölümde sunulan "Üç Seviyeli Çoklu Saldırı Tipli Elektrik Şebekesi Koruma Modeli" oluşturulmuştur.

5.1 Literatür İncelemesi (Elektrik Şebekeleri Yasaklama/Koruma Modelleri)

Yasaklama modelleri pek çok farklı kritik alt yapı için çalışılmaktadır. Bunlardan elektrik şebekelerine yönelik yapılan çalışmalar dikkat çekmektedir. Özellikle ABD'de elektrik sistemlerine yapılan saldırılar ve terörle mücadeleye yönelik çalışan birimlerin konuya dikkat çekmeleri ile bu alanda çalışmaların yapılmaya başladığı görülmektedir. Elektrik sistemlerine yönelik olarak bu alanda yapılan ilk çalışma Salmoron vd. (2004)'ne aittir. Çalışmada bir terörist atak sonrası elektrik sisteminin direncini belirlemeye ve kritik olan bileşenleri tanımlamaya yönelik iki seviyeli yeni bir model ve çözüm önerilmiştir. Model üst seviyesi, saldıran tarafından ele alınmıştır, yük atma maliyeti ve elektrik üretim maliyeti toplamını enbüyüklemeyi amaçlamaktadır. Elektrik sistemlerinde, yük atma elektriğin tümden kaybedilmesini engellemek amacıyla, sistemin elektrik arzı ve talebinde oluşan dengesizliği gidermek için elektrik arzında düştüğünde, bazı bölgelerin elektriğini keserek (yük atma) sistemi dengede tutmak üzere gerçekleştirilen sistem müdahalesidir.

Elektrik şebekelerinde elektrik arz ve tüketim dengesi sistemin sürdürülebilirliği için gerekli şarttır. Motto vd. (2005), elektrik şebekelerine yönelik oluşturulmuş iki seviyeli bu modelin doğrusallaştırılması ve tek seviyeye indirilmesini incelemiştir. Tek seviyeli karma tamsayılı doğrusal modeli ise dal-kesme (branch-cut) yöntemi ile çözmüştür.

Salmeron ve Wood (2015) elektrik şebekelerinde yüksek voltaj trafolarının yedeklerinin hangilerinin saldırılara karşı yedek olarak kullanılabileceğinin belirlenmesine yönelik bir çalışma yapmıştır ve bu saldıran-savunan şeklinde modellemiştir. Aynı şekilde Lezama J.M.L. vd. (2017) elektrik güç şebekelerine yönelik olarak saldıran-savunan şeklinde iki seviyeli bir yasaklama problemi kurgulamıştır. Problemde elektrik sisteminin doğasına uygun kısıt ve amaç fonksiyonları kullanmıştır. Problem yerel arama ile çözülmüş ve GA ile karşılaştırılmıştır.

Wu ve Conejo (2017) elektrik şebekelerinin savunması problemi için 3 seviyeli bir optimizasyon modeli geliştirmiştir. Üst seviye savunma planlamacısı, orta seviye saldırıyı yapanın seviyesi, alt seviye ise sistem operatörüdür (sistem operatörü saldırı sonrası sistemde yaptığı değişiklikler ile yük atma miktarını enküçüklemeyi amaçlamaktadır). Alguacil vd. (2014)'in elektrik şebekelerinin savunmasının ele alındığı çalışmasında 3 aşamalı bir model oluşturulmuştur. Üst seviyede savunma planlayıcısı hangi bileşenleri koruyacağının tespitini yapmaktadır. Orta seviyede ise saldırgan enbüyük zararı vermek üzere hangi bileşenleri etkisiz hale getireceğini belirlenmektedir. Alt seviyede sistem operatörü verilen zarar sonrası şebekenin çalışma sisteminde yapılan değişikler ile hasarı enaza indirmeye çalışmaktadır. Problemde öncelikle 3 seviye, 2 seviyeli şekle dönüştürülmüştür.

Elektrik sistemlerine yönelik yapılan 3 seviyeli modellerde, savunan – saldırgan - sistem operatörü şeklinde modellerle karşılaşılmaktadır.

Delgadillo (2010) terör tehdidi altında olan elektrik şebekelerinin güvenlik açığı (vulnerability) analizine yönelik bir çalışma yapmıştır. Oluşturulan model karma tamsayılı doğrusal olmayan iki seviyeli bir modeldir, modelin üst seviyesi saldırgan açısındandır. Modelin ayırt edici özelliği sistem operatörünün düzeltici eylemler içerisinden ağ topolojisini değiştirme kabiliyetine sahip olmasıdır. Model doğrusal olmaması nedeni ile tek

seviyeye düşürülememiş, bu nedenle Benders Ayrıştırmasına dayanan bir çözüm yöntemi kullanılmıştır.

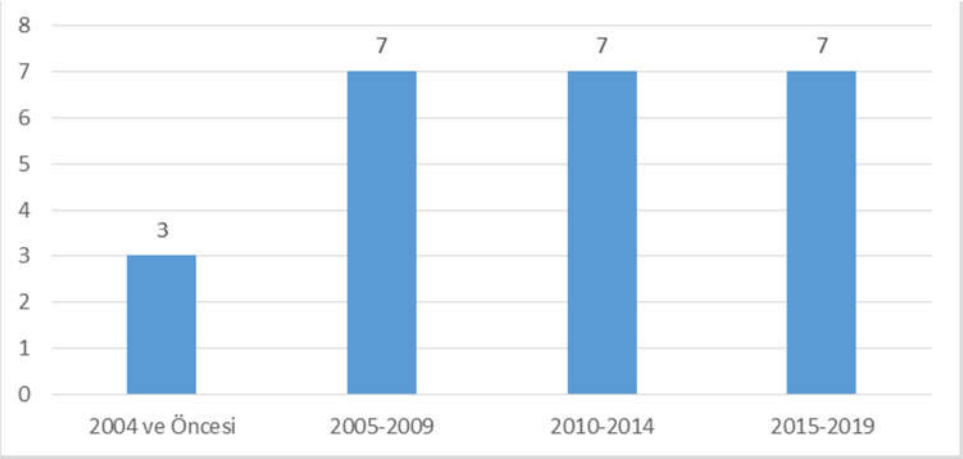
Elektrik şebekelerine yönelik yapılan çalışmaları incelendiğinde benzer amaç fonksiyonları içerdiği görülmektedir (Çizelge 5.1). Genel olarak yük atma miktarının enbüyüklenmeye ya da enküçüklenmeye çalışıldığı görülmektedir. Yük atma bir şebekedeki elektrik tüketim miktarı ile sisteme verilen elektrik miktarını dengede tutmak amacı ile yapılan bir müdahaledir. Amaç fonksiyonunda yük atmanın yanında elektrik maliyetlerinin ele alındığı çalışmalar da yapılmıştır.

Çizelge 5.1. Elektrik Şebekelerine yönelik yapılan yasaklama/koruma modellerinin yapıları

Yazar	Yıl	Model seviyesi	Üst Seviye	Amaç Fonksiyonu
Salmeron vd.	2004	2 (AD)	Saldırgan	Elektrik üretim maliyeti ve yük atma maliyeti toplamının enbüyüklenmesi
Motto	2005	2 (AD)	Saldırgan	Elektrik üretim maliyeti ve yük atma maliyeti toplamının enbüyüklenmesi
Arroyo ve Galiana	2005	2 (AD)	Saldırgan	Saldırıya uğramamış hat sayısını enbüyüklemek
Yao Y.	2007	3 (DAD)	Savunan	Elektrik üretim maliyeti ve yük atma maliyeti toplamının enküçüklenmesi
Delgadillo	2010	2 (AD)	Saldırgan	Yük atma miktarının enbüyüklenmesi
Alguacil vd.	2014	3 (DAD)	Savunan	Yük atma miktarının enküçüklenmesi
Salmeron ve Wood	2015	2 (AD)	Saldırgan	t periyodunda elektrik üretim maliyeti ve t periyodunda atılan yükün maliyeti toplamının enbüyüklenmesi
Wu ve Conejo	2017	3 (DAD)	Savunan	Yük atma miktarının enküçüklenmesi
Lezama J.M.L. vd.	2017	2 (AD)	Saldırgan	Yük atma miktarının enbüyüklenmesi
Zhao ve Zeng	2013	2 (AD)	Saldırgan	Servis edilen yük miktarının en küçüklenmesi

İki ya da üç seviyeli yasaklama/koruma modeli çalışmalarının yanında elektrik hatlarının güvenlik açığı analizi için farklı çalışmalar da yapılmıştır. Bier (2007)'in yasaklama modeline alternatif olarak geliştirdiği model her bir yinelemede şebekede en yüksek yüke sahip iletim hattının engellendiği açgözlü bir algoritmaya dayanmaktadır.

Salmoron vd. (2004) ile başlayan Elektrik sistemlerine yönelik bu alandaki çalışmalar Şekil 5.1'den de görüleceği gibi güncelliğini korumaktadır**.**



Şekil 5.1. Elektrik Şebekeleri Yasaklama/Koruma Çalışmalarının Yıllara Göre Dağılımı

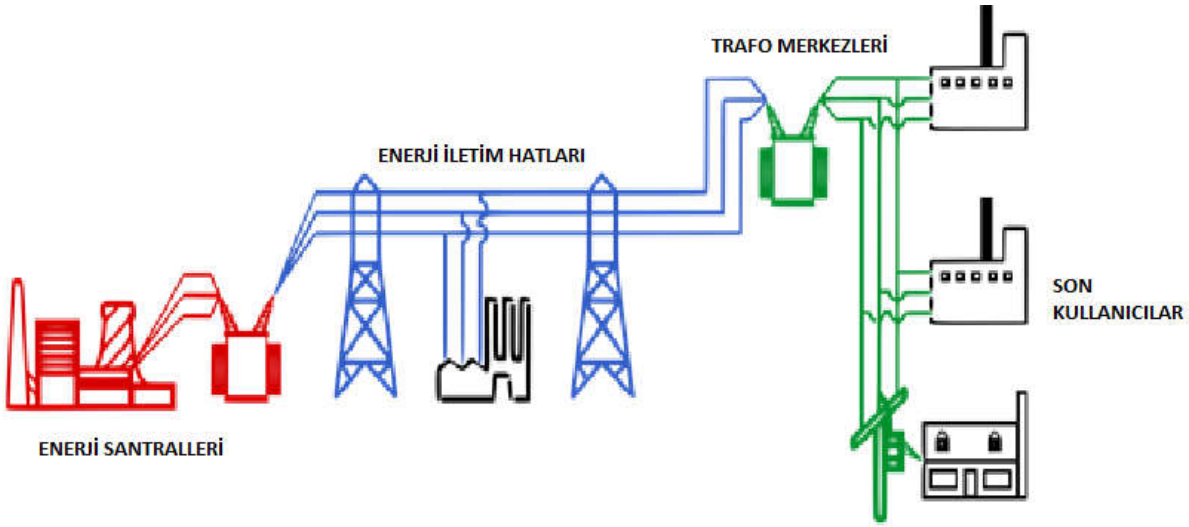
Bu tezde ise problem, kritik tesis olarak seçilen elektrik şebekesi özelinde üç seviyeli bir koruma modeli olarak modellenmiştir. Modelin amaç fonksiyonu yük atma miktarı ile ilişkilidir. Ancak sadece yükün miktarı değil aynı zamanda yük atılan bölgenin önemi göz önüne alınacak şekilde amaç fonksiyonu oluşturulmuştur. Akıllı bir saldırgan yasaklama hamlesini yapmadan önce öncelikle hangi bölgeleri elektriksiz bırakmanın kendi çıkarları için önemli olacağını düşünecektir. Bir bölgenin kritiklik seviyesini ise saldırganın hedef alma potansiyeli ve savunma planlamacısı için önemi belirleyecektir.

5.2. Elektrik Şebekeleri

Elektrik enerjisi elektrik santrallerinde (hidroelektrik, termik, doğalgaz, rüzgar vs.) üretilir, iletim hatları vasıtası ile ülkenin çeşitli bölgelerinde kurulu trafo merkezlerine iletilir ve bu merkezlerden son tüketiciye, sanayi kuruluşlarına vb. elektrik ulaştırılır. Bu bileşenler, elektriği kullanıcılara iletmek üzere oluşturulmuş ağlardır. Elektrik, depolanmasının zorluğu nedeni ile kullanıcılara hemen ulaştırılmalıdır. Elektrik talebi ile üretiminin denge içinde olması gerekmektedir. Bu denge ülkemizde TEİAŞ (Türkiye Elektrik İletim A.Ş.) tarafından sağlanmaktadır.

Türkiye elektrik iletim şebekesi 68.204 km uzunluğunda enerji iletim hattı, 736 trafo merkezi, 172.276 MVA trafo gücü ve komşu ülkelerle 12 enterkoneksiyon hattından meydana gelmektedir. 88.550,8 MW santral kurulu gücü, 47.660 MW ani puantı, 304.801,9 GWh yıllık elektrik enerjisi üretimi gerçekleştirilmektedir (TEİAŞ, 2019).

Şekil.5.2'de elektriğin son kullanıcıya nasıl ulaştığı basitçe gösterilmektedir. Elektrik santrallerinde üretilen elektrik, trafo merkezlerine kadar iletim hatları üzerinden ulaştırılır. Trafo merkezlerinden ilgili bölgeye elektrik iletilir.



Şekil 5.2. Elektrik Şebekesi

Santrallerde üretilen elektriğin en az kayıpla uzak mesafelere ulaştırılabilmesi için gerilimi yükseltilir, çok yüksek gerilimli (380 kV) ve yüksek gerilimli (154 kV) şebeke iletim hatları ile elektrik trafo merkezlerine taşınır. Trafo merkezlerinde çok yüksek gerilim ve yüksek gerilim, orta gerilime (34 kV) düşürülür ve tüketim bölgelerine dağıtılmak üzere dağıtım trafolarına gönderilir. Dağıtım trafoları ile düşük gerileme (0,4 kV) düşürülen elektrik son kullanıcılara sunulur. Problem çerçevesinde elektrik şebekeleri ele alınırken saldırıya uğraması muhtemel noktalar; Santraller, çok yüksek ve yüksek gerilim iletim hatları ve trafo merkezleri olarak belirlenmiştir. Dağıtım trafolarına ve sonraki elektrik dağıtım sürecine yapılan saldırıların yeterli etkiyi sağlamayacağı göz önüne alınarak göz ardı edilmiştir.

Elektrik şebekeleri herhangi bir sebeple (doğa koşulları, sabotaj vd.) saldırıya uğradığında elektrik arzının istenilen seviyenin altına düşmesine sebep olunmaktadır. Bu durumlarda sistem elektrik üretimini arttırmaktadır ve/veya bazı bölgelerin elektriğini keserek (yük atma) denge sağlamaktır.

Ülkemizde elektrik frekans değeri 50 hz civarında olmalıdır. Elektrik üretim ve tüketiminde oluşacak dengesizlikler frekans değerlerinde sapmalara, bu değerlerde oluşacak sapmalar ise belli bir aralığın üzerine çıktığında tüm sistemin kapanmasına sebep olmaktadır. Bu durumlarda bazı bölgelerin elektriği belirli bir sıra ile kesilerek yük atma işlemi yapılmaktadır. Yük atma ile üretim ve tüketim dengesi sağlanabilmekte ve sistemin tamamının bloke olması engellenmektedir. Yük atma işlemi trafo merkezlerinin beslediği bölgelerden bazılarının elektriğinin kesilmesi ile sağlanmaktadır. Burada önemli konulardan biri öncelikle hangi bölgelerin elektriğinin kesileceğidir. Bu da ilgili bölgenin önemine göre belirlenecektir. Bu çalışma kapsamında ileride de görüleceği üzere amaç fonksiyonu yük atma ile ilişkili tanımlanmıştır. Elektrik kesintisi yapılacak bölgelerin önemi amaç fonksiyonuna yansıtılmıştır.

Enterkonnekte Şebekeler ve Elektrik Şebekelerinin Ağ Yapısı:

Ülkemiz elektrik şebekesi, yurt içinde ve yurtdışında oluşturmuş olduğu bağlantılar ile yüksek miktarlı enerji alışverişi sağlayacak şekilde, bölgeler arası ve uluslararası bağlantı olanağı sağlayan enterkonnekte bir sistemin parçasıdır. Bir ülkenin tamamının veya belli bölgelerinin elektrik enerji ihtiyacını karşılayacak üretim ve tüketim merkezleri arasındaki enerji alışverişini temine yarayan enerji nakil hatlarının teşkil ettiği sisteme enterkonnekte sistem denir (Çakır, 1989). Enterkonnekte sistemlerde şebekenin belli bir bölgesinde gerçekleşen bir değişiklik, ağın başka bir bölgesini etkiler. Enterkonnekte sistemde bir arıza olduğunda, sadece arıza olan yerin enerjisi kesilir, diğer kısımlarda enerjinin sürekliliği bozulmaz. Sistem içerisinde bir bölgede arızalanan santral veya trafolar devre dışı bırakıldığında diğer santral ve trafolar bu bölgeleri beslemeye devam eder. Bu tip şebekelerde, o bölgedeki bütün elektrik üretim ve tüketim araçları büyük küçük ayrımı yapılmaksızın sisteme dahil edilmektedir, Enterkonnekte şebekenin kesintisiz elektrik sağlayabilme, yüksek verim, ekonomik olması gibi avantajları vardır. Bununla birlikte kısa devre akımlarının yüksek oluşu ve sistemin kararlılığının sağlanmasının zor oluşu gibi sakıncaları vardır (Üstünel, 2012).

Enterkonnekte sistemi üretim ve tüketim yönünden emniyette tutan, kaliteli ve ekonomik olarak işletilmesine devamlı olarak nezaret eden, işletme manevralarının koordinasyon ve kumandasını yapan işletme merkezine, yük tevzi merkezi adı verilir

(Bergen A.R.,1986). Enterkonnekte sistemler, elektriğin taşınmasında yüksek verimli ve ekonomik olması nedeni ile daha karlıdır ve ağın güvenilirliği daha yüksektir. Ülkemiz ulusal elektrik sistemi, Yunanistan, Bulgaristan, Gürcistan, Ermenistan, Nahcivan, İran, Irak ve Suriye ile enterkonnekte sistem ile birbirine bağlantılıdır. Yurt içinde ise 10 yük tevzi merkezi ile enterkonnekte sistem işletilmektedir.

Enerji sisteminin güvenli ve karalı bir şekilde işletilebilmesi için elektriksel niceliklere ait bazı kısıtların sağlanması gerekmektedir. Bu kısıtlar, elektrik şebekesi koruma probleminin modellenmesinde, modelin probleme özel kısıtlarını oluşturmaktadır. Probleme ait kısıtların açıklanmasında ise elektrik şebekelerine ait kullanılan terimler ise özetle şunlardır; "Aktif Güç" bir santralin sisteme verdiği gücü gösterirken, "Aktif Güç Sınırı" bir santralin fiziksel özelliklerinden dolayı sisteme verebileceği enaz ve enfazla gücü gösterir. "Aktif Güç Talebi" ise kritik bölgeye bağlı trafo merkezinden talep edilen gücün miktarını göstermektedir. Elektrik İletim hattı üzerinden çekilen güce ise "Görünen Güç" denir. Görünen gücün bir üst sınırı mevcuttur ve herhangi bir hat için bu limit aşıldığında, yüksek akımın neden olduğu aşırı ısınmadan dolayı hat fiziksel olarak zarar görecek ve devre dışı kalacaktır. Baralar aynı gerilim ve frekans değerinde elektriğin toplandığı ve dağıtıldığı birimlerdir. Her baranın ise bir gerilim alt ve üst sınırı mevcuttur. Bir baranın elektrik toplama ve dağıtması ise baraya giriş yapan akım ve çıkış yapan akımlar ile gerçekleşmektedir. Bu akımlar dalgalı bir şekilde gelmektedir, zaman farkları ile giriş ve çıkış yapan bu akımların zamansal farkları ise faz açılarını oluşturmaktadır. Baralarda faz açısının, alt üst limiti mevcuttur. Kullanıcıların elektrik enerjisini kararlı bir şekilde kullanabilmeleri için bara gerilim ve faz açılarının belirlenen sınırlar içerisinde kalması gerekmektedir. Kondüktans (conductance) ise bir malzemenin akımı iletme derecesidir, iletim hatlarının elektrik iletkenliğini ifade eder. Bir malzemenin iletkenliği, eğer karmaşık sayılarla matematiksel olarak ifade edilebiliyorsa, bunun imajinel kısmı süseptans (susceptance)'dır.

5.3. Üç Seviyeli Çoklu Saldırı Tipli Elektrik Şebekesi Koruma Modeli (Yeni Model 3)

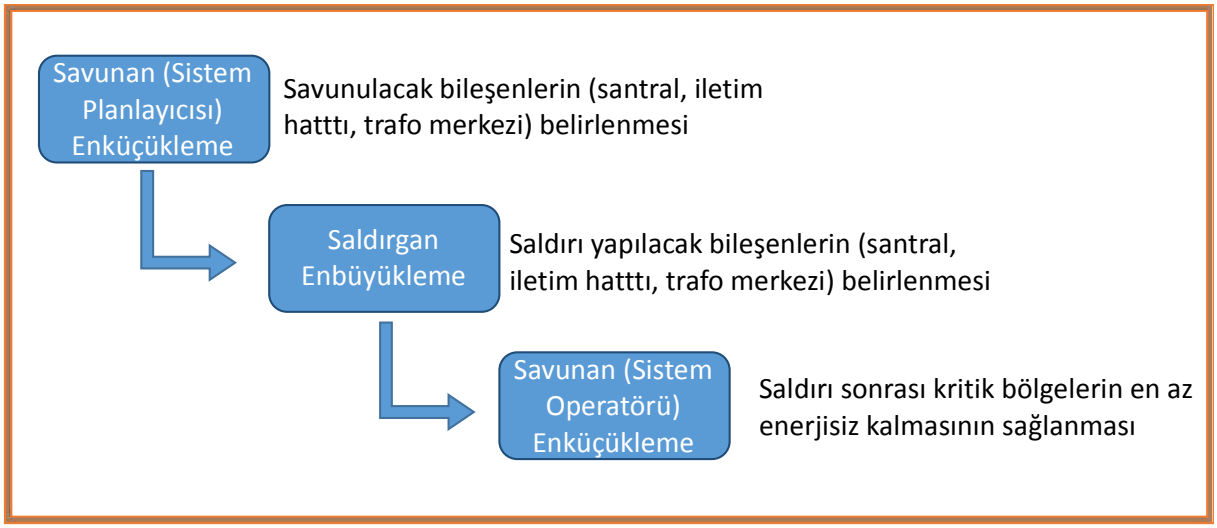
Elektrik şebekelerinde, enerjinin üretildiği santraller, enerjiyi trafo merkezine taşıyan iletim hatları ve trafo merkezleri, terör vb. saldırılara karşı korunması gereken bileşenlerdir. Bu bileşenlerden son nokta olan trafo merkezlerinde, elektriğin gerilimi düşürülmekte ve bağlı bölgelerin elektriği verilmektedir. Bu trafo merkezlerinin taleplerini ilgili bölgedeki son kullanıcılar (haneler) ve sanayi kurumları oluşturmaktadır. Trafo merkezleri çeşitli santrallerden elektrik alarak ilgili bölgenin ihtiyacını karşılamaktadır. Bu nedenle bu trafo merkezlerinin elektrik taleplerini kesintisiz olarak karşılanması gerekmektedir. Elektrik üretim santrallerine, santraller ile trafo merkezleri arasındaki iletim hatlarına ve trafo merkezlerine yapılan saldırılar ile bu bölgelerin elektrik iletimi kesintiye uğrayabilir ve elektriksiz kalma problemi yaşanabilir.

Bölgelere elektrik enerjisini sağlayan enerji üretim santralleri, iletim hatları ve trafo merkezleri elektrik şebekesinin kritik bileşenleridir. Saldırganların bu kritik tesisleri hedef almaları ve bunlardan yeterli korunmaya sahip olmayanlarını devre dışı bırakacakları beklenir. Kritik tesisin faaliyetlerini yerine getiremeyecek şekilde devre dışı bırakılması, durdurulması "yasaklama" olarak ifade edilecektir.

Elektrik sistemlerinde yapılan bir saldırıda sistemin tamamın bloke olmaması için yük atma fonksiyonu çok önemlidir. Bu fonksiyon yerine getirilirken yük atılacak bölgenin seçilmesi, göz ardı edilemeyecek konuların başında gelmektedir. Bu nedenle modelde yük atmanın nerelerden yapılacağının belirlenmesine yönelik değişikler yapılmıştır. Diğer çalışmaların büyük çoğunluğundan farklı olarak sadece iletim hatları değil, santrallerin ve trafo merkezlerinin saldırıya uğrama ihtimalleri de bu çalışmada göz önüne alınmıştır. Gerçekte de bakıldığında özellikle trafo merkezleri hedef olma açısından çok cazip noktalardır. Hem çoğunluğunun saldırıya açık olarak kırsal alanlarda konumlandığı hemde verilen zararın telafisinin ya da bu hizmeti yerine koymanın uzun sürecek olması nedeni ile en kırılgan noktalardır ve bu bileşenlerde gerçekleşen bir yasaklama, bağlı iletim hatlarını da etkiyecektir. Çünkü bu bileşenlerin bazıları hub konumundadır ve bu hubların zarar görmesi bunlara bağlı iletim hatlarınında devre dışı kalmasına sebep olacaktır.

Holmgren vd. (2007) de saldırı stratejileri göz önüne alınarak bazı atak senaryoları belirlenmiştir. Bu tezde ise bölgelerin kritiklik değerleri tanımlanırken benzer atak stratejileri göz önüne alınarak bu farklılaşma probleme dahil edilebilir. Saldırı stratejisi esasen saldırganın nerelere öncelikle saldırmaya karar vermesi ile ilgilidir. Benzer şekilde ilgili bölgelerin kritiklik değerleri belirlenerek saldırı stratejisi belirlemede de bu bilgi kullanılabilir.

"Üç Seviyeli Çoklu Saldırı Tipli Elektrik Şebekesi Koruma Modeli" gerçek bir elektrik şebekelerinin kısıtları göz önüne alınarak çoklu saldırı tipli ve 3 seviyeli olarak modellenmiştir. Modelin üst seviyesi elektrik sistemi planlayıcısının seviyesidir. Planlamacı hangi bileşenlerin ne tür saldırılara karşı korunacağının kararını vermektedir, amacı herhangi bir saldırı durumunda sistem operatörünün yaptığı değişikler neticesinde bölgenin enaz zararı görmesidir. Orta seviye ise saldırganın (terörist) seviyesidir. Saldırgan elektrik şebekesinde hangi bileşenlerin hangi saldırı tiplerine karşı savunulduğunu bilir ve savunmasız olanlarına enbüyük zararı vermek amacı ile saldırır. En alt seviye ise sistem operatörünün seviyesidir. Bu sevideki operatör saldırı sonrası şebekede gerekli değişikleri, sistem kısıtlarını göz önüne alarak yapar ve sistemin enaz zararı görmesini sağlar. Sistem Planlayıcısı – Saldırgan – Sistem Operatörü Modelinin şematik gösterimi Şekil 5.3 görülmektedir. Şekilde görüldüğü gibi sistem planlayıcısı öncelikle korunacak bileşenleri belirliyor ve koruma kaynaklarını tahsis ediyor, saldırgan korunmayan bileşenler içinden saldıracağı bileşenleri belirliyor. Sistem operatörü ise saldırı gerçekleştikten sonra sistemde gerekli değişikleri yaparak sistemiz en az zararı görmesini sağlıyor. Önerilen model ile eniyi koruma planını belirlemek amaçlanmaktadır.



Şekil 5.3. Savunan (Defender - Sistem Planlayıcısı) – Saldırgan (Attacker) – Savunan (Defender - Sistem Operatörü) Modeli

Varsayımlar:

1. Saldırgan, farklı tiplerde ve belirli sayılarda saldırı yapabilme yeteneği ve donanımına sahiptir ve bu bilinmektedir. Ancak bir saldırı anında bunların belli sayıdaki bir karmasını kullanacaktır.
2. Saldırgan hangi tesislerin hangi saldırı tiplerine karşı korunduğunu bilmektedir ve bu tesislere ilgili saldırı tipi ile saldırmaz.
3. Saldırganın farklı çeşitteki saldırı tiplerinde kaçar saldırı yapabilme kabiliyetine sahip olduğu bilinmektedir.
4. Bir saldırı periyodunda, saldırganın mevcut saldırı kaynaklarından toplamda kaç adedini yapabileceği bilinmektedir.
5. Koruma kaynakları net olarak bilinmektedir. Koruma kaynakları ile hangi tip saldırıya karşı hangi tür bileşenin (santral, iletim hattı ve trafo merkezi) korunabileceği bilinmektedir. (Örneğin, sistem planlayıcısı drone saldırısına karşı kaç santrali koruyabileceğini, ya da el yapımı patlayıcı saldırısına karşı kaç trafo merkezini tahkim edebileceğini bilmektedir).

İndisler :

$i \in I$: Kritik bölge

$n, b \in N$: Trafo merkezi

$j \in J$: Elektrik enerjisi üretim santrali

$l \in L$: İletim hattı

$e \in E$: Saldırı çeşidi

Kümeler :

I : Kritik bölgeler $I = \{i|i = 1,2,..s\}$

N : Trafo merkezleri $N = \{n|n = 1,2,..t\}$

N_i : i kritik bölgesini besleyen trafo merkezleri $N_i \in N$

N_j : j. santralden beslenen trafo merkezleri $N_j \in N$

N_l : l iletim hattına bağlı trafo merkezleri $N_l \in N$

N_n : n trafo merkezine bağlı diğer trafo merkezleri $N_n \in N$

J : Elektrik enerjisi üretim santralleri $J = \{j|j = 1,2,..u\}$

L : İletim hatları $L = \{l|l = 1,2,..u\}$

E : Saldırı çeşitleri $E = \{e|e = 1,2,..v\}$

Parametreler:

$Yük_n$: Kritik bölgelere hizmet veren n. trafo merkezinin ihtiyacı olan yük miktarı

KD : i. kritik bölgenin kritiklik değeri

$\theta_n^{min}, \theta_n^{max}$: n. trafo merkezinin faz açısı sınırları

V_n^{min}, V_n^{max} : n. trafo merkezinin gerilim sınırları

PG_j^{min}, PG_j^{max} : j. santralin aktif güç sınırı

P_n : n. trafo merkezinin aktif güç talebi

S_{bn}^{max} : b ve n trafo merkezleri arasındaki hattın görünür üst güç sınırları

con_{bn} : b ve n trafo merkezlerini birbirine bağlayan iletim hattının kondüktansı (conductance)

sus_{bn} : b ve n trafo merkezlerini birbirine bağlayan iletim hattının suseptansı (susceptance)

MJ_e : Santralleri e tipi saldırıya karşı korumada kullanılabilecek kaynak kapasitesi

ML_e, : İletim hatlarını e tipi saldırıya karşı korumada kullanılabilecek kaynak kapasitesi

MN_e : Trafo merkezlerini e tipi saldırıya karşı korumada kullanılabilecek kaynak kapasitesi

KJ_e, : Saldırganın e saldırı tipinde yasaklayabileceği santral sayısı kapasitesi

KL_e, : Saldırganın e saldırı tipinde yasaklayabileceği iletim hattı sayısı kapasitesi

KN_e, : Saldırganın e saldırı tipinde yasaklayabileceği trafo merkezi sayısı kapasitesi

R : Saldırganların yasaklayabileceği bileşen (santral, iletim hattı ve trafo merkezi) sayısı kapasitesi

Karar Değişkenleri:

$$ZJ_{je} = \begin{cases} 1, & j\ santrali\ e\ saldırı\ tipine\ karşı\ korunursa \\ 0, & d.d. \end{cases}$$

$$SJ_{je} = \begin{cases} 1, & j\ santrali\ e\ saldırı\ tipi\ ile\ yasaklanırsa \\ 0, & d.d. \end{cases}$$

$$ZL_{le} = \begin{cases} 1, & l\ iletim\ hattı\ e\ saldırı\ tipine\ karşı\ korunursa \\ 0, & d.d. \end{cases}$$

$$SL_{le} = \begin{cases} 1, & l\ iletim\ hattı\ e\ saldırı\ tipi\ ile\ yasaklanırsa \\ 0, & d.d. \end{cases}$$

$$ZN_{ne} = \begin{cases} 1, & n\ trafo\ merkezi\ e\ saldırı\ tipine\ karşı\ korunursa \\ 0, & d.d. \end{cases}$$

$$SN_{ne} = \begin{cases} 1, & n\ trafo\ merkezi\ e\ saldırı\ tipi\ ile\ yasaklanırsa \\ 0, & d.d. \end{cases}$$

ΔP_n $= n\ trafo\ merkezinden\ atılan\ yük\ miktarı$

$$x_i = \begin{cases} 1, & yasaklama\ sonrası\ i\ kritik\ bölgesinde\ yük\ atma\ gerçekleşiyor\ ise \\ 0, & d.d. \end{cases}$$

θ_n $= n\ trafo\ merkezinin\ yasaklama\ sonrası\ faz\ açısı$

V_n $= n\ trafo\ merkezi\ yasaklama\ sonrası\ gerilimi$

PG_j $= j.santralinden\ yasaklama\ sonrası\ sağlanan\ aktif\ güç$

S_{nb} $= n\ ve\ b\ trafo\ merk.arasındaki\ hattın\ yasaklama\ sonrası\ görünen\ gücü$

Amaç Fonksiyonu

$$\text{enk}\, H(z) \tag{5.1}$$

Kısıtları altında

$$\sum_{j \in J} ZJ_{je} \leq MJ_e \qquad \forall e \in E \tag{5.2}$$

$$\sum_{l \in L} ZL_{le} \leq ML_e \qquad \forall e \in E \tag{5.3}$$

$$\sum_{n \in N} ZN_{ne} \leq MN_e \qquad \forall e \in E \tag{5.4}$$

$$ZJ_{je}, ZL_{le}, ZN_{ne} \in \{0,1\} \tag{5.5}$$

Yerine

$$H(z) = \text{Enb}\, H(z)^* \tag{5.6}$$

Kısıtları altında

$$\sum_{i \in I} SJ_{je} \leq KJ_e \qquad \forall e \in E \tag{5.7}$$

$$\sum_{l \in L} SL_{le} \leq KL_e \qquad \forall e \in E \tag{5.8}$$

$$\sum_{n \in N} SN_{ne} \leq KN_e \qquad \forall e \in E \tag{5.9}$$

$$SJ_{je} \leq 1 - ZJ_{je} \quad \forall j \in J, \quad \forall e \in E \tag{5.10}$$

$$SL_{le} \leq 1 - ZL_{le} \quad \forall l \in L, \quad \forall e \in E \tag{5.11}$$

$$SN_{ne} \leq 1 - ZN_{ne} \quad \forall n \in N, \quad \forall e \in E \tag{5.12}$$

$$\sum_{i \in I} \sum_{e \in E} SJ_{je} + \sum_{l \in L} \sum_{e \in E} SL_{le} + \sum_{n \in N} \sum_{e \in E} SN_{ne} \leq R \tag{5.13}$$

$$SJ_{je}, SL_{le}, SN_{ne} \in \{0,1\} \tag{5.14}$$

Yerine

$$H(z)^* = \text{Enk} \sum_{i} \sum_{n \in N_i} \Delta P_n . KD_i \tag{5.15}$$

Kısıtları altında

$$\sum_{n \in N_j} PG_n - \sum_{b \in N_n} S_{nb} + \Delta P_n = P_n \qquad \forall n \in N \tag{5.16}$$

$$S_{nb} = (1 - SL_{le}).[V_n^2 con_{nb} - V_n V_b\,(con_{nb}.\cos(\theta_n - \theta_b) - sus_{nb}.\sin(\theta_n - \theta_b))] \tag{5.17}$$

$$\forall l \in L, \quad \forall n, b \in N_l, \quad \forall e \in E$$

$$\theta_n^{min} \leq \theta_n \leq \theta_n^{max} \qquad \forall n \in N \tag{5.18}$$

$$V_n^{min} \leq V_n \leq V_n^{max} \qquad \forall n \in N \tag{5.19}$$

$$PG_n^{min} \leq PG_n \leq PG_n^{max} \qquad \forall n \in N_j \tag{5.20}$$

$$0 \leq S_{nb} \leq S_{nb}^{max} \qquad \forall n, b \in N \tag{5.21}$$

$$0 \leq \Delta P_n \leq P_n \qquad \forall n \in N \tag{5.22}$$

$$PG_j \leq PG_j^{max} . (1 - SJ_{je}) \qquad \forall j \in J, \quad \forall e \in E \tag{5.23}$$

$$V_n \leq V_n^{max} . (1 - SN_{ne}) \qquad \forall n \in N, \quad \forall e \in E \tag{5.24}$$

$$S_{nb} \leq S_{nb}^{max} . (1 - SL_{le}) \qquad \forall n, b \in N_l, \quad \forall l \in L, \quad \forall e \in E \tag{5.25}$$

(5.1) Üç seviyeli modelin üst seviye amaç fonksiyonudur. Sistem planlayıcısının amaç fonksiyonudur. Yasaklama sonrası kritik bölgelerden atılan yük miktarını enküçüklemeyi amaçlar.

(5.2) e tipi saldırı çeşidine karşı santralleri korumak için ayrılmış olan kaynak kadar santrallere tahkim yapılmasını sağlar.

(5.3) e tipi saldırı çeşidine karşı iletim hatlarını korumak için ayrılmış olan kaynak kadar iletim hatlarına tahkim yapılmasını sağlar.

(5.4) e tipi saldırı çeşidine karşı trafo merkezlerini korumak için ayrılmış olan kaynak kadar trafo merkezlerine tahkim yapılmasını sağlar.

(5.6) Üç seviyeli modelin orta seviye amaç fonksiyonudur. Saldırganın amaç fonksiyonudur. Yasaklama sonrası kritik bölgelerden atılan yük miktarını enbüyüklemeyi amaçlar.

(5.7) e saldırı tipi ile en fazla yasaklayabileceği santral sayısını sınırlar.

(5.8) e saldırı tipi ile en fazla yasaklayabileceği iletim hattı sayısını sınırlar.

(5.9) e saldırı tipi ile en fazla yasaklayabileceği trafo merkezi sayısını sınırlar.

(5.10) e tipi saldırıya karşı korunan santrale e tipi saldırı yapılmasını önler.

(5.11) e tipi saldırıya karşı korunan iletim hattına e tipi saldırı yapılmasını önler.

(5.12) e tipi saldırıya karşı korunan trafo merkezine e tipi saldırı yapılmasını önler.

(5.13) Aynı anda kullanılabilecek yasaklama kaynağını sınırlar.

(5.15) Üç seviyeli modelin alt seviye amaç fonksiyonudur. Sistem operatörünün amaç fonksiyonudur. Yasaklama sonrası kritik bölgelerden atılan yük miktarını enküçüklemeyi amaçlar. Bu amaçla saldırı sonrası, bu seviyede belirtilen kısıtlara göre sistem operatörü şebekede gereken değişikleri yapar.

(5.16) Enerji sistemi aktif güç üretim tüketim dengesini sağlar. Sistemdeki üretim birimlerinin aktif güç üretim değerlerinin toplamı, sistemden talep edilen toplam aktif güç değerleri ile iletim hatlarındaki aktif güç kayıplarının toplamına eşit olmalıdır.

(5.17) İletim hatlarında taşınan aktif güç miktarını belirler. Burada yer alan con_{nb} ve sus_{nb} parametreleri *n* ve *b* trafo merkezleri arasına bağlı iletim hattının, sırasıyla, kondüktans (*conductance*) ve suseptans (*susceptance*) değeridir. Ancak pratikte bir iletim hattının parametreleri olarak rs_{nb} direnci (resistance) ve rc_{nb} reaktansı (*reactance*) verilmektedir. Bu durumda iletim hattının kondüktans ve suseptans değerleri aşağıda verilen eşitlikle elde edilir. Bu eşitliğin karmaşık sayılar içerdiğine dikkat ediniz.

$$(con_{nb} + jsus_{nb}) = {}^{1}/_{(rs_{nb} + jrc_{nb})} \qquad (5.26)$$

(5.18) Yasaklama sonrası n. trafo merkezinin faz açısının faz açısı limitleri içerisinde olmasını sağlar.

(5.19) Yasaklama sonrası n. trafo merkezinin geriliminin gerilim limitleri içerisinde olmasını sağlar

(5.20) Yasaklama sonrası j. santralinden sağlanan aktif gücün santral aktif güç sınırları içerisinde olmasını sağlar.

(5.21) Yasaklama sonrası n ve b trafo merkezleri arasındaki l. iletim hattının görünen gücünün, hattın güç sınırları içerisinde olmasını sağlar.

(5.22) Yasaklama sonrası n. trafo merkezinden atılacak yük, n. trafo merkezinin yük talebinden fazla olamaz.

(5.23) j. santrali eğer yasaklanıyorsa yasaklama sonrası o santralden aktif güç alınmayacaktır.

(5.24) n. trafo merkezi yasaklanıyorsa o trafo merkezi üzerinden kritik bölgeye gerilim akışı olmayacaktır.

(5.25) l. iletim hattı yasaklanıyorsa o hat üzerinden aktif güç akışı olmayacaktır.

(5.5) ve (5.14) karar değişkenlerinin işaret kısıtlarıdır.

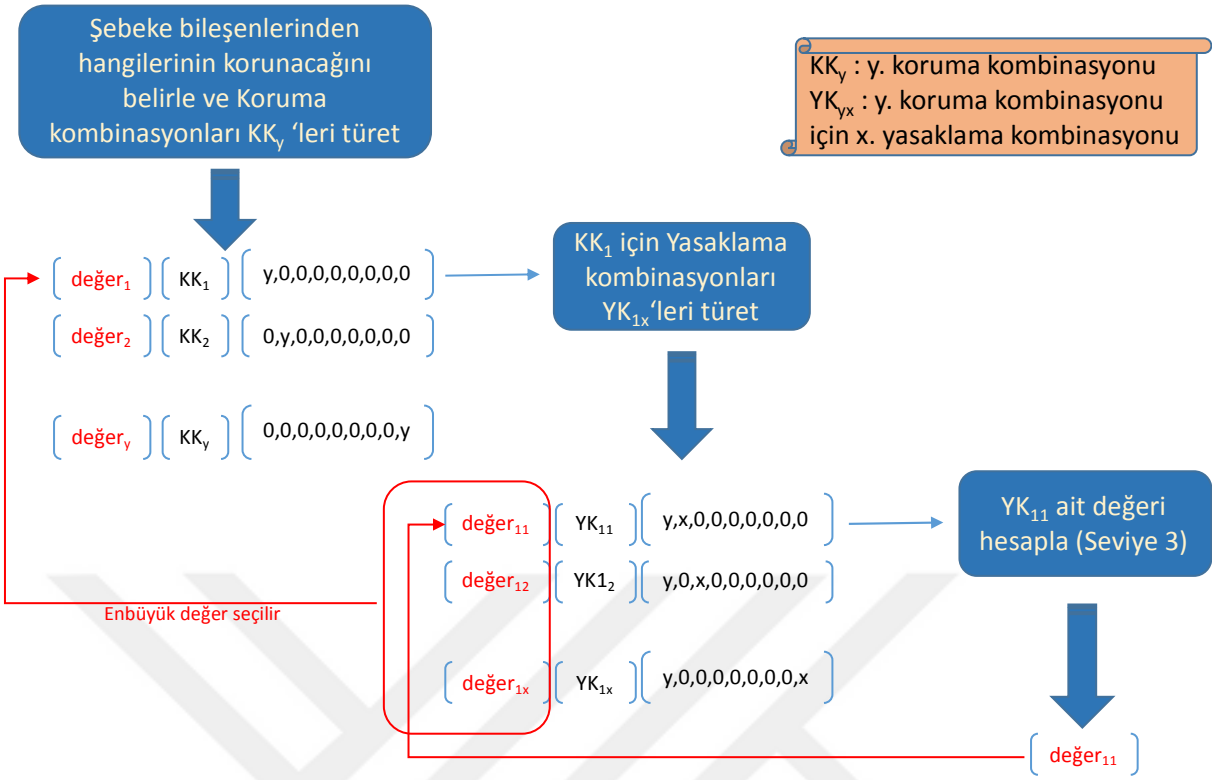
İleriki bölümlerde geliştirilen bu modelin çözüm yaklaşımı ele alınmaktadır. "Üç Seviyeli Çoklu Saldırı Tipli Elektrik Şebekesi Koruma Modeli" için geliştirilen çözüm algoritması bölüm 5.4'de sunulmaktadır. Matematiksel model ve çözüm yaklaşımının deneysel değerlendirmesi ise bölüm 7.1'de örnek veri setleri ile gerçekleştirilmektedir.

5.4. Üç Seviyeli Çoklu Saldırı Tipli Elektrik Şebekesi Koruma Modeli Çözüm Algoritması

Üç seviyeli çoklu saldırı tipli koruma modelinin üst seviyesi, sistem planlayıcısının seviyesidir ve saldırı gerçekleşmeden önce sistem planlayıcısı koruma kararını vermiştir. Orta seviye saldırganın seviyesidir ve korunan bileşenleri iyi bilen saldırgan tarafından en büyük zararı vermek üzere yasaklanacak bileşenler belirlenir. Alt seviye ise sistem operatörünün seviyesidir ve saldırı sonrası sistemin en az zararı görmesi için gerekli değişikleri yapar. Bu değişikler iletim sisteminin kesintisiz ve güvenli bir şekilde işletilmesi için gerçek zamanlı üretim tüketim dengesini sağlamak, sistem frekans ve gerilim regülasyonunu gerçekleştirmek gibi müdahalelerdir. Bu müdahaleler primer, sekonder frekans kontrolü ve tersiyer yedek arıza hallerinde yapılan her türlü manevraların, jeneratörlerin, hatların, oto-trafoların, güç trafolarının, reaktörlerin, sönt ve seri kapasitörlerin gerilim regülasyonu yaparak, bu kapsamda jeneratörlerin reaktif yüklenmesi,

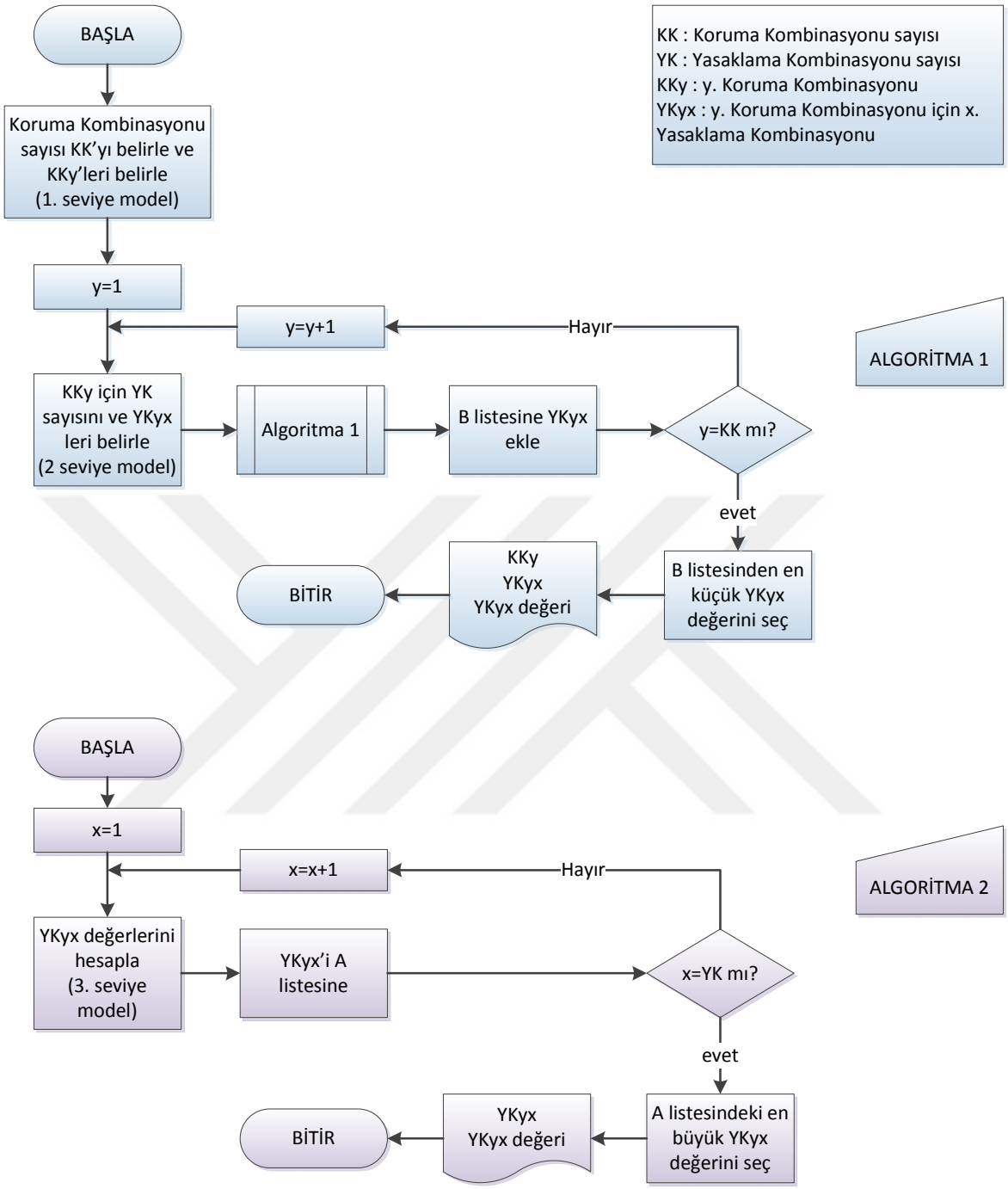
senkron kompansatör alınıp reaktif yüklenmesi, reaktör ve kapasitörlerin devreye alınıp çıkartılmasını sağlamak gibi faaliyetlerdir. Tanımlamalara dikkat edilirse, sistem planlayıcısı saldırı öncesi kararını vermiştir, sistem operatörü ise saldırı sonrası gerekli değişikleri yapacaktır. Saldırganın aktif olduğu ara aşamada, korunan yerleri, saldırı sonrası yapılacak değişiklerin bilen saldırgan, enbüyük zararı vermek üzere saldırı planlar.

Bu üç seviyeli problem için önerilen çözüm yaklaşımı şu şekildedir: Sistem planlayıcısının elindeki koruma kaynakları ile oluşturabilecek tüm alternatif koruma planları türetilir. Türetilen her bir koruma planı için saldırganın, elindeki saldırı kaynakları ile yapabileceği olası yasaklama kombinasyonları türetilir. Amacı saldırı sonrası verilen zararı enküçüklemek olan sistem operatörü bu set içerisindeki her bir yasaklama kombinasyonu için elde edebileceği eniyi amaç fonksiyonu değerini hesaplar (Şekil 5.4). Bu değer aynı zamanda ilgili saldırgana (seviye 2) ait yasaklama kombinasyonun amaç fonksiyonu değeridir. Bu şekilde, bir koruma kombinasyonuna ait yasaklama kombinasyonu setindeki, tüm yasaklama kombinasyonlarının amaç fonksiyonu değerleri hesaplanır. Bu set içerisindeki en büyük amaç fonksiyonu değeri ilgili koruma kombinasyonun amaç değeridir. Bu şekilde hesaplanan tüm koruma kombinasyonları değerlerinin içerisinden enküçük değere sahip kombinasyon, sistem planlayıcısı tarafından aranan koruma planıdır ve problemin (seviye1) amaç fonksiyonu değeri bu kombinasyonun değeridir.



Şekil 5.4. Olası Tüm Koruma ve Yasaklama Kombinasyonlarının Değerlendirildiği Çözüm Yaklaşımı

Modelin çözümü için eldeki kaynaklar ile gerçekleştirilebilecek koruma kombinasyonlarının türetildiği, sonrasında her bir koruma kombinasyonu için saldırganın saldırı kaynakları ile oluşturabileceği yasaklama kombinasyonlarını türeten bir algoritma önerilmiştir (Algoritma 1). Diğer bir algoritma ise her bir saldırı kombinasyonu için modelin 3. seviyesinin çalıştırıldığı ve bu saldırı kombinasyonlarından enbüyük amaç fonksiyonu değerine sahip yasaklamayı seçen bir başka algoritmadır (Algoritma 2). İkinci algoritmada elde edilen sonuçlar birinci algoritmanın bir alt sürecidir. Birinci algoritmadaki her bir koruma kombinasyonu için ikinci algoritma en büyük amaç fonksiyonu değerine sahip yasaklama kombinasyonunu ve değerini döndürmektedir. Birinci algoritmada, her bir koruma kombinasyonu için dönen bu değerler, aynı zamanda koruma kombinasyonun amaç fonksiyonu değeridir. Algoritma 1 bu koruma kombinasyonları içinden en küçük amaç fonksiyonu değerine sahip koruma kombinasyonunu seçer, nihai problemin sonucu elde edilir (Şekil 5.5).



Şekil 5.5. Üç Seviyeli Koruma Modelinin Çözüm Algoritmaları

Sistem planlayıcısı elindeki koruma kaynakları ile bileşenleri pek çok farklı şekilde koruyabilir. Bu koruma kombinasyonlarının her biri için ise saldırgan elindeki kaynaklar çerçevesinde saldıracağı bileşenleri çeşitli şekillerde yasaklayabilir. Bu yasaklama kararını sistem operatörünün saldırı sonrası yapacağı düzenlemeyi göz önünde bulundurarak yapar.

Saldırgan, saldırı kombinasyonları içinden en büyük zararı veren saldırı kombinasyonunu tercih edecektir. Bu saldırı kombinasyonunun değeri esasen sistem planlayıcısının ilgili koruma kombinasyonun değeridir. O halde sistem planlayıcısı olası koruma kombinasyonları içerisinden enküçük amaç fonksiyonu değerine sahip olan koruma kombinasyonunu seçecektir. Bu bakış açısı ile Şekil 5.5'te ki algoritmalar oluşturulmuştur. Algoritma 2 her bir yasaklama kombinasyonu için sistem operatörü modelinin (modelin 3. seviyesi) çalıştığı ve enküçük sonucu ürettiği ve bunlardan enbüyük amaç fonksiyon değerini sağlayan yasaklama kombinasyonunu belirleyen algoritmadır. Algoritma 1 ise her bir koruma kombinasyonu için amaç fonksiyonu değerlerini belirleyen ve bunlardan enküçük değere sahip koruma kombinasyonunu belirleyen algoritmadır.

Algoritma 1: Koruma kaynakları çerçevesinde tüm koruma kombinasyonları türetilir. Koruma kombinasyonu sayısı (KK) belirlenir. Her bir koruma kombinasyonu için saldırganın yasaklama kombinasyonları seti ve yasaklama kombinasyon sayısı (YK) belirlenir. Algoritma 2 çalıştırılır ve sonuçları B listesine kayıt edilir (B listesi, koruma ve yasaklama kombinasyonlarının ve değerlerinin tutulduğu listedir). Tüm koruma kombinasyonları için bu işlem tekrarlanır. Döngü tamamlandıktan sonra B listesi içinden enküçük değerli kombinasyon seçilir. Bu kombinasyonda hangi bileşenlerin korunduğu, amaç değeri ve bu koruma planında saldırganın verebileceği en kötü durumu oluşturacak saldırı planı görülecektir.

Algoritma 2: Algoritma 1'den YK yasaklama kombinasyonu sayısı ve KK_y koruma kombinasyonuna ait YK_{yx}yasaklama kombinasyonlarının seti algoritma 2'ye gelir. Algoritma 2'de her YK_{yx} yasaklama kombinasyonu için 3. seviye model çalıştırılır ve amaç fonksiyonu değeri belirlenir, elde edilen sonuçlar A listesine eklenir (A listesi, bir koruma planına ait tüm yasaklama kombinasyonlarının ve değerlerinin tutulduğu listedir). Tüm yasaklama kombinasyonları tamamlanıncaya kadar işlem tekrarlanır. Döngü tamamlandıktan sonra A listesinden enbüyük amaç değerine sahip yasaklama kombinasyonu ve değeri belirlenir ve süreç bitirilir.

Önerilen bu çözüm algoritması ile "Üç Seviyeli Çoklu Saldırı Tipli Elektrik Şebekesi Koruma Modeli" deneysel değerlendirmesi, Bölüm 7.1'de örnek veri setleri ile gerçekleştirilmektedir. Bölüm 5.5'de problemin karmaşıklık analizi yapılarak NP-zor bir problem olduğu gösterilmiş ve sezgisel bir algoritmaya olan ihtiyaç görülmektedir.

5.5 Karmaşıklık Analizi

Bu bölümde ele alınan problem için önerilmiş olan çözüm algoritmasının karmaşıklığı NP-zor olarak belirlenmiş ve bu problemin çözümü için sezgisel algoritmaya olan ihtiyaç gösterilmiştir.

Algoritmalar belirli bir görevi yerine getiren sonlu sayıdaki işlemler dizisidir. Yapılan çalışmalara göre algoritmaların işlev derecelerine göre büyüme hızları (karmaşıklığı) yavaştan hızlıya doğru Çizelge 5.2'deki gibidir:

Çizelge 5.2. Fonksiyon Derecelerine Göre Algoritmaların Yavaştan Hızlıya Büyüme Hızları

1	log n	n	n.log n	n^2	n^3	n^c	2^n	$n!$

Karmaşıklık analizi, algoritma analizinin ve karşılaştırılmasının yapıldığı çalışmalardır. Algoritmaların teorik analizinde, asimptotik anlamda karmaşıklıklarının tahmini, yani büyük girdiler için karmaşıklık işlevini tahmini yaygın olarak yapılmaktadır ve bu amaçla genellikle, büyük O notasyonu, omega notasyonu ve teta notasyonları kullanılmaktadır (Santoso, 2019).

Büyük O notasyonu, n bağımsız değişkeni sonsuza doğru yöneldiğinde, bir fonksiyonun sınırlayıcı davranışını açıklayan matematiksel bir gösterimdir. Big O notasyonu Paul Bachmann (1894) ve Edmund Landau (1909) tarafından bulunmuş, Bachmann-Landau notasyonu veya asimptotik notasyon olarak isimlendirilen notasyonlardan biridir.

Problem 3 seviyeli bir model olarak ele alınmıştır. Çözüm için önerilen algoritmada, koruma ve yasaklama kaynaklarına göre ilk seviyede olası tüm koruma kombinasyonları türetilmekte ve ikinci seviyede her bir koruma kombinasyonu için yasaklama kombinasyonları belirlenmektedir. Yasaklama kombinasyonları 3. seviyede çözülmektedir.

Çözüm algoritmasının karmaşıklığını analiz etme amacı ile öncelikle yasaklama kombinasyonu sayısına ait fonksiyon şu şekilde belirlenmiştir;

Problem, santral, trafo merkezi ve iletim hattı olmak üzere üç bileşenden oluşmaktadır. Her bileşen kendisini yasaklayacak saldırı çeşidi kadar koruma (saldırı) tipine sahiptir.

n santral sayısı, j koruma kaynağı sayısı, i yasaklama kaynağı sayısı olmak üzere tek bir bileşenin (örneğin santrallerin), tek bir tipi için yasaklama kombinasyonu sayısı;
Yasaklama Kombinasyonu Sayısı = Koruma Kombinasyonu sayısı x Korunmayan santrallerin olası yasaklanma sayısı eşitliğe göre aşağıda ki gibidir.

$$\text{Yasaklama Kombinasyonu Sayısı} = \frac{n!}{(n-j)!\,.\,j!} \text{ x } \frac{(n-j)!}{i!\,.\,(n-j-i)!} = \frac{n!}{j!\,.\,i!\,.\,(n-j-i)!} \qquad (5.27)$$

Bu eşitliğe göre çeşitli n değerlerine göre hesaplanan yasaklama kombinasyonu sayıları aşağıdaki Çizelge 5.3'de gösterilmiştir.

Çizelge 5.3. Farklı *n* (santral sayısı) Değerleri için Oluşacak Yasaklama Kombinasyonu Sayıları (koruma kaynağı sayısı, *j* =2, yasaklama kaynağı sayısı *i* =1, saldırı tipi tek alınmıştır)

	Yasaklama Kombinasyonu Sayısı
n	$\frac{n!}{j!\,.\,i!\,.\,(n-j-i)!}$
3	$\frac{3!}{2!\,.\,1!\,.\,(3-2-1)!} = \frac{3.2!}{2!\,.\,1!\,.\,(3-2-1)!} = \frac{3.2!}{2!} = 3$
4	$\frac{4!}{2!\,.\,1!\,.\,(4-2-1)!} = \frac{4.3.2!}{2!\,.\,1!\,.\,(4-2-1)!} = \frac{4.3.2!}{2!} = 4.3 = 12$
5	$\frac{5!}{2!\,.\,1!\,.\,(5-2-1)!} = \frac{5.4.3.2!}{2!\,.\,1!\,.\,(5-2-1)!} = \frac{5.4.3.2!}{2!\,.\,2!} = 5.3.2 = 30$
6	$\frac{6!}{2!\,.\,1!\,.\,(6-2-1)!} = \frac{6.5.4.3.2!}{2!\,.\,1!\,.\,(6-2-1)!} = \frac{6.5.4.3!}{2!\,.\,3!} = 6.5.2 = 60$
7	$\frac{7!}{2!\,.\,1!\,.\,(7-2-1)!} = \frac{7.6.5.4!}{2!\,.\,1!\,.\,4!} = \frac{7.6.5}{2!} = 7.5.3 = 105$
8	$\frac{8!}{2!\,.\,1!\,.\,(8-2-1)!} = \frac{8.7.6.5!}{2!\,.\,1!\,.\,5!} = \frac{8.7.6}{2!} = 8.7.3 = 168$
9	$\frac{9!}{2!\,.\,1!\,.\,(9-2-1)!} = \frac{9.8.7.6!}{2!\,.\,1!\,.\,6!} = \frac{9.8.7}{2!} = 9.7.4 = 252$

Çizelge 5.3. Farklı *n* (santral sayısı) Değerleri için Oluşacak Yasaklama Kombinasyonu Sayıları (koruma kaynağı sayısı, *j* =2, yasaklama kaynağı sayısı *i* =1, saldırı tipi tek alınmıştır) (devam)

10	$\frac{10!}{2!\,.\,1!\,.\,(10-2-1)!} = \frac{10.9.8.7!}{2!\,.\,1!\,.\,7!} = \frac{10.9.8}{2!} = 10.9.4 = 360$
. .	. .
n	$\frac{n!}{2!\,.\,1!\,.\,(n-2-1)!} = \frac{n.(n-1).(n-2).(n-3)!}{2!\,.\,1!\,.\,(n-3)!} = \frac{n.(n-1).(n-2)}{2}$

Çizelge 5.3'den yararlanarak $f(n)$ fonksiyonu (5.28)'de verildiği gibi elde edilir. *n*'in artan değerlerinde *f(n)*'nin maksimum büyüme oranı *g(n)* = n^3'dür. Burada n^3 değerinden düşük *n* değerlerinin performansı önemsiz kabul edilmektedir.

$$f(n) = \frac{n.(n-1).(n-2)}{2} = \frac{(n^3-3n^2+2n)}{2} = O(g(n^3)) \qquad (5.28)$$

f(n), *O(g(n))* ya da O(n^3) derecesindedir diyebiliriz. Tanıma göre, tüm n>1 değerleri için ve *C* bir sabit iken, $|f(n)| \leq C\,|g(n)|$ ifadesi geçerlidir.

Çizelge 5.3'de belirtildiği gibi tek bir bileşenin sayısının değiştiği yani koruma kaynağı ve yasaklama kaynağı sayıları sabit saldırı tipi de tek iken santral sayısının (*n*) farklı değerlerde olabildiği durum yerine; koruma ve yasaklama kaynakları yine sabit ancak iki tip saldırının olması halinde santral sayısının (*n*) farklı değerleri için fonksiyonun karmaşıklığı $n^3.n^3$ olacaktır. Tip sayısı *a* olsun, aynı durum için bu bileşenin karmaşıklığı n^{3a}'dır. Problemde bulunan 3 bileşen (santral, trafo merkezi, iletim hattı) için ise karmaşıklık değeri $n^{3a}.n^{3a}.n^{3a} = n^{9a}$'dır. Bu fonksiyon polinom bir fonksiyondur. Tip sayısındaki (*a*) artış görüldüğü gibi fonksiyonu üssel olarak arttıracaktır. Çizelge 5.4'de fonksiyonların büyüme oranları farklı *n* sayıları için verilmiştir.

Çizelge 5.4. Fonksiyon Derecelerine Göre Büyüme Oranları

Fonksiyon	*n*=10	*n*=100	*n*=1000	*n*=10000	*n*=100000	*n*=1000000
1	1	1	1	1	1	1
$log_2 n$	3	6	9	13	16	19
n	10	10^2	10^3	10^4	10^5	10^6
n^2	10^2	10^4	10^6	10^8	10^{10}	10^{12}
n^3	10^3	10^6	10^9	10^{12}	10^{15}	10^{18}
n^{9a}	10^9	10^{18}	10^{27}	10^{36}	10^{45}	10^{54}
2^n	10^3	10^{30}	10^{301}	10^{3010}	10^{30103}	10^{301030}

İki seviyeli eniyileme problemleri NP-zor problemlerdir, tüm fonksiyonları sürekli ve doğrusal olduğunda bile bu problemlerin NP-zor problemler olduğu Hansen, Jaumar ve Saward tarafından kanıtlanmıştır (Bard, 1991, Zhang, 2014, Scaparra ve Church, 2008). Üç seviyeli eniyileme problemleri ise iki seviyeli problemlerin genelleştirilmiş halidir ve NP zor problemlerdir (Mahmoodjanloo, 2016).

6. ÜÇ SEVİYELİ ÇOKLU SALDIRI TİPLİ ELEKTRİK ŞEBEKESİ KORUMA PROBLEMİ İÇİN GENETİK ALGORİTMA TABANLI SEZGİSEL YAKLAŞIMLAR

Bölüm 5.4'de verilen Üç Seviyeli Çoklu Saldırı Tipli Elektrik Şebekesi Koruma Modeli Çözüm Algoritması kullanılarak küçük boyutlu problemler için kesin çözüm elde edilmiştir. Ancak problem boyutu büyüdükçe makul sürelerde çözüm bulmak güçleşir, nitekim yedinci bölümde yapılan deneysel çalışmalar da bu gösterilmiştir. Büyük boyutlu problemler için Genetik Algoritma tabanlı iki sezgisel yaklaşım önerilmiştir. 6.1 Bölümü'nde Genetik Algoritmalar hakkında genel bir bilgi verilmekte, 6.2'de önerilen Genetik Algoritmalar tabanlı r_GA ve k_GA sezgisel yaklaşımları tanıtılmaktadır. Önerilen her iki yaklaşımın performansına katkı sağlaması amacı ile çaprazlama işlemi sonrası yerel arama tabanlı bir komşu arama operatörü çözüm sürecinde yer almaktadır. Bu operatör ile çaprazlama sonrası elde edilen en iyi bireyin komşuları aranmaktadır.

6.3 bölümünde ise r-GA ve k_GA sezgisel algoritmalarının eniyi parametre değerlerini belirlemek üzere deney tasarımı yapılmıştır. Her bir parametre için en iyi performansı veren düzeyler belirlenmiştir. 6.4'de ise bu algoritmaların kodlanmasında kullanılan yazılım, yazılım dilleri ve çözücüler hakkında bilgi verilmiştir.

6.1. Genetik Algoritmalar

İncelediğimiz problemi gerçek boyutlu bir sistem için ele aldığımızda kabul edilebilir süre içerinde tüm alternatif çözümlerin amaç fonksiyonu değerlerinin hesaplanması mümkün değildir, bu nedenle problemin çözümü için genetik algoritmalar tabanlı sezgisel bir algoritma kullanılmıştır.

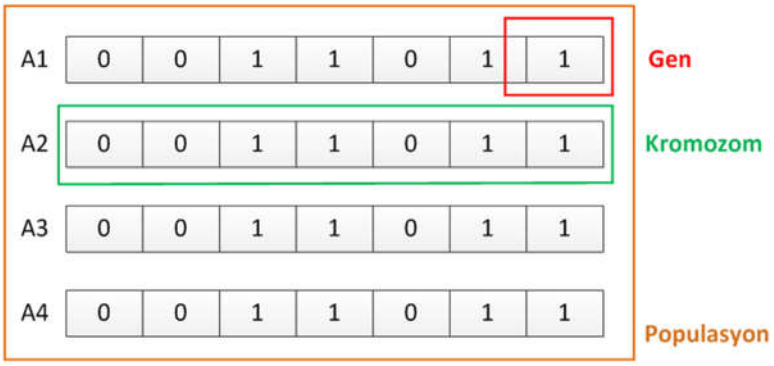
Genetik algoritmalar (GA) pek çok gerçek yaşam probleminde kullanılan iyi bilinen bir metasezgiseldir. GA'nın temelleri Holland (1975) tarafından ortaya konmuştur. GA Darwin teorisinin en güçlünün hayatta kalması bakış açısı temel alınarak geliştirilmiştir.

GA'nın eniyileme problemlerinde tercih edilen bir teknik olmasının temel nedenleri şunlardır (Gen ve Cheng, 1997);

1. Özel matematiksel yapılar içermemesi
2. Bütünsel en iyiyi araştırmada etkin bir yöntem olması
3. Özel problemler için daha güçlü araştırma yapılarının geliştirilebileceği, melez yöntemlerin kullanımına uygun esnek bir yapıya sahip olması.

GA'lar diğer eniyileme yöntemlerinden farklı olarak, arama işlemini tek bir aday çözüm ile gerçekleştirmek yerine birden fazla aday çözümün oluşturduğu bir topluluk ile gerçekleştirir. Bundan dolayı çözüm uzayının birden çok başlangıç noktasıyla paralel olarak taranması sağlanmış olur. Bu özellik, GA'ların yerel eniyi değerlere takılmadan, global eniyi değerleri bulabilmesinde en büyük etken olmaktadır (Mitchell, 1999).

GA'da kullanılan temel kavramlar gen, kromozom ve popülasyondur. Gen, kendi başına anlamı olan ve genetik bilgi taşıyan GA'nın en küçük genetik birimdir (Şekil 6.1). Bir gen A, B gibi bir karakter olabileceği gibi 0 veya 1 ile ifade edilen bir bit veya bit dizisi olabilir. Eniyilenecek değişkene aittir ve her bir genin aldığı değer "allel" olarak ifade edilir.



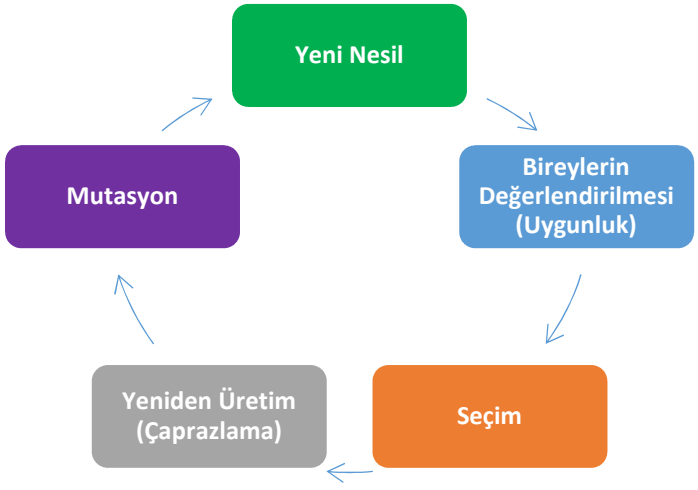
Şekil 6.1. Bir Genetik Problemde, Örnek Gen, Kromozom ve Popülasyon Gösterimi

Kromozomlar ise bir ya da birden fazla genin bir araya gelmesiyle oluşurlar ve probleme ait tüm bilgileri içerirler. Kromozom popülasyon içindeki bireylere karşılık gelir. Problem için eniyi çözüm için alternatif adaylardan biridir. Bir kromozom belirlenirken problemin yapısına uygun olarak kodlaması gerekir. Kodlama, çözümlerin veya popülasyondaki bireylerin nasıl temsil edileceğine karar verilmesidir. Bu ikili kodlama,

değer kodlama, ağaç kodlama şeklinde ya da bir gezgin satıcı problemi gibi problemler için permütasyon kodlama olabilir.

Şekil 6.1'deki gibi popülasyon, kromozomlar veya bireyler topluluğudur. Popülasyon üzerinde durulan problem için alternatif çözümler kümesidir. Popülasyon büyüklüğü probleme göre belirlenir ve popülasyon içindeki kromozom sayısı sabittir. n adet kromozomun seçilerek oluşturulan başlangıç setine başlangıç popülasyonu denir. Bazı araştırmalar ideal başlangıç popülasyon büyüklüğünün 20-30 civarı olması gerektiğini belirtirken bazı araştırmalarda 50-100 civarı popülasyon büyüklüğünün en ideal olduğunu söylemektedir (Reeves ve Rowe, 2002; Pasia vd. 2005).

GA'da probleme uygun olarak kromozomlar kodlandıktan ve başlangıç popülasyonu oluşturulduktan sonra her bir bireyin (kromozomun) ne kadar iyi olduğunu belirlemek üzere uygunluk fonksiyonu belirlenir. Uygunluk Fonksiyonu (Fitness Function), bir popülasyon oluşturulduktan sonra, popülasyondaki her bireyin uygunluk (fitness) değerinin hesaplandığı fonksiyondur. Bu fonksiyon genetik algoritmanın beynini oluşturmaktadır. GA da probleme özel çalışan tek kısım bu fonksiyondur. Uygunluk değerleri göz önüne alınarak seçme, çaprazlama ve mutasyon işlemleri gerçekleştirilerek yeni nesil (popülasyon) oluşturulur (Şekil 6.2).



Şekil 6.2. GA'da Bir Nesilden Yeni Bir Neslin Üretimi

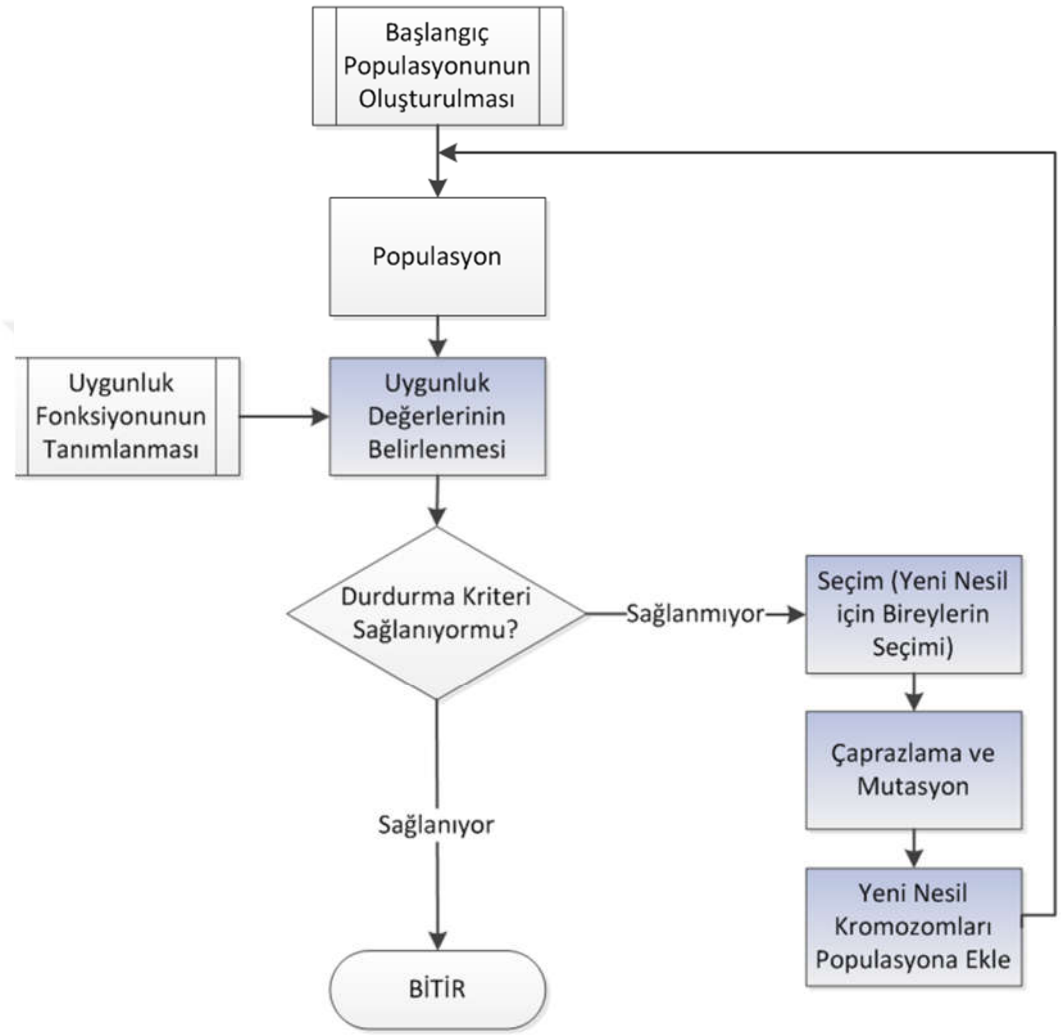
Bir popülasyondaki kromozomların uygunluk değerleri belirlenip ne kadar iyi olduğu tespit edildikten sonra seçim işlemi, yeni nesillerde daha yüksek uygunluk değerlerine sahip bireylerin oluşması için, eski popülasyondaki bir kromozomun, uygunluk değerine bağlı olarak bazı yöntemlerle yeni oluşturulacak bir popülasyon içine seçme işlemidir. Seçim işlemi için Turnuva seçimi, Rulet tekerleği seçimi, Sıralama seçimi, Sabit durum seçimi ve Stokastik seçim yöntemleri kullanılabilir (Goldberg ve Deb, 1991). Yeni nesillerin bir öncekinden farklı nesiller üreterek çeşitliliği arttırmak ve böylece daha geniş bir çözüm uzayında çalışarak arzu edilen sonuca ulaşmak amacıyla çaprazlama (crossover) yapılır. Çaprazlama, yeni kromozomların eski kromozomların iyi genlerini alıp daha iyi olacakları düşüncesiyle yapılır. Çaprazlama operatöründen kromozomların iyi özelliklerini birleştirerek daha iyi kromozomlar oluşturması beklenir. GA'da yapılan aramanın yerel eniyilerde takılmasının önüne geçmek için ise mutasyon operatörü kullanılır. Mutasyon genellikle çaprazlamaya göre daha az olasılıkta kullanılır. Bunun nedeni çaprazlama sonucu elde edilen uyum değeri yüksek dizileri kaybetmemektir (Reeves ve Rowe, 2002). Yeni kromozomlara yer açmak için eski kromozomlar ortadan kaldırılır. Çaprazlama ve Mutasyon GA için esastır, çaprazlama, algoritmanın farklı bireylerden en iyi genleri çıkarmasını ve bunları potansiyel olarak üstün çocuklara birleştirmesini sağlar. Mutasyon ise bir popülasyonun çeşitliliğine katkıda bulunur ve böylece algoritmanın daha iyi uygunluk değerlerine sahip bireyler üretme olasılığını arttırır (https://www.mathworks.com/help/gads/how-the-genetic-algorithm-works.html). Yeni oluşan neslin uygunluk değerleri hesaplanarak yeni neslin başarısı bulunur. Bu süreç durma noktasına gelinceye kadar defalarca tekrarlanır. Durma sonrası bulunan en iyi kromozom ise GA'nın sonuç değerini oluşturur.

GA temel olarak Şekil 6.3'de de verilen aşağıdaki adımları içerecek şekilde kullanılır:

1. Uygunluk (fitness) fonksiyonu tanımlanması.
2. Genetik kodlama.
3. Rastgele bireylerden oluşacak şekilde başlangıç popülasyonun türetilmesi.
4. Tekrarla (yeterince iyi bir çözüm bulana kadar)
    a. Populasyondaki tüm bireylerin uygunluk değerlerinin hesaplanması
    b. Yeni nesil için en iyi bireylerin belirlenmesi
    c. Çaprazlama ve mutasyon ile yeni neslin oluşturulması

d. Yeni nesil (kromozomları) popülasyona ekle

5. En iyi çözümü döndür.



Şekil 6.3. GA'nın Temel Adımları

Yasaklama/koruma modellerinde ve elektrik şebekelerine yönelik olarak ele alınan problemlerde GA yaklaşımının kullanıldığı görülmektedir.

Agudelo vd.'nin (2015) çalışması, elektrik şebekelerine yönelik yapılan ve çözümünde özelleştirilmiş bir Genetik Algoritma yaklaşımının kullanıldığı bir çalışmadır. GA'nın uygunluk fonksiyonu olarak yük atma miktarı kabul edilmiştir. Popülasyonun

bireyleri 1-0 şeklinde ikili bir kromozom olarak sunulmuştur. Başlangıç popülasyonu her bireyde aynı sayıda saldırıya uğramış gen olacak şekilde rastgele türetilmiştir.

Lezama vd. (2017), elektrik şebekelerine yönelik iki seviyeli doğrusal olmayan bir yasaklama modeli kurmuşlardır, bu çalışmada yinelenen yerel arama ve Genetik algoritma kullanılarak karşılaştırılması yapılmıştır. Yapılan çalışma sonucunda hem sonuç hem de süre açısından yinelenen yerel arama sezgiselinin daha iyi sonuç verdiği belirlenmiştir.

Mahmoodjanloo vd. (2016) üç seviyeli modellerinde Genetik algoritmaları kullanmışlardır. Çalışmada tesislerin kurulacağı yerleri belirleyen birinci seviyede Genetik algoritma kullanılmış ve elde edilen tesis yerlerini gösteren kromozom 2. Seviyeye gönderilmiştir. Burada eldeki saldırı kaynaklarına göre olası tüm saldırı kombinasyonları belirlenerek değerlerinin hesaplanması için 3 seviyeye gönderilmiştir. 2 ve 3 seviyede tam sayım metodu kullanılmıştır. Aynı çalışmada GA yerine biyocoğrafya temelli bir sezgisel (BBO) kullanılmıştır. Performans sonuçları benzerdir, bazen GA, bazen BBO iyi sonuç göstermiştir, CPU süresi açısından GA daha iyi performans göstermiştir

Yine Fard ve Keshteli'de (2018)' iki amaçlı iki seviyeli bir yasaklama modeli oluşturmuşlardır. Problemin çözümünde su dalgası optimizasyonunun ve genetik algoritmanın melez bir yapısı kullanılmıştır. Yine bu çalışmada balina algoritması ile parçacık sürüsü yöntemleri melez olarak çalışılmıştır. Çalışmada, bu dört algoritmanın ayrı ayrı sonuçları ve iki melez algoritmanın sonuçları karşılaştırılmıştır. Çözüm süreleri dışında en iyi performansı melez balina optimizayonu ve parçacık sürüsü optimizasyonunun gösterdiği gösterilmiştir.

Jiang ve Liu (2018) su şebekelerinin yasaklaması problemi için çok amaçlı olarak önerdikleri modelin çözümünde genetik algoritmaları kullanmışlardır. İç içe geçmiş bir sezgisel şeklinde kullanılan Genetik algoritmalar, Savunma-seviyesi sezgisel genetik algoritma Yasaklama-seviyesi sezgisel genetik algoritma olmak üzere iç içe geçmiş şekilde oluşturulmuştur.

6.2. Üç Seviyeli Çoklu Saldırı Tipli Elektrik Şebekesi Koruma Problemi için Geliştirilen Genetik Algoritma Tabanlı Sezgisel Yaklaşımlar

Bu bölümde Üç Seviyeli Çoklu Saldırı Tipli Elektrik Şebekesi Koruma probleminin yapısına uygun olarak kromozom yapıları ve operatörler tanımlanmaktadır. Öncelikle 0-1 yapılı kromozom yapısı oluşturulmuştur. Sonrasında başlangıç popülasyonu, önce rassal olarak oluşturulan r_GA algoritması, sonrasında ise başlangıç popülasyonu kural tabanlı bir yapı ile oluşturulan k_GA algoritması önerilmiştir. Çaprazlama işlemi üç farklı öğe (Santral, Trafo Merkezi ve İletim Hatları) barındıran problemin yapısına uygun olarak üç farklı noktadan gerçekleştirilmektedir. Mutasyon işlemi ise yine her bir öğe için gerçekleştirilerek üç farklı noktadan gerçekleştirilmektedir. Başlangıç Popülasyonu Büyüklüğü, Çaprazlama Oranı, Çaprazlama Nokta Sayısı, Mutasyon Oranı, En İyi Birey Oranı ve Kural Tabanlı Birey Oranı gibi parametreler ilgili algoritmaların performansını etkileyen parametrelerdir. Bu parametrelerin değerleri ise 6.3'de yapılan deney tasarımı ile belirlenmiştir.

Kromozomların Kodlanması:

Genetik algoritma için öncelikle kromozom yapısı oluşturulmuştur. Kromozomlar 0-1 ikili kodlama yöntemi ile kodlanmıştır. Bir kromozom üç bölümden oluşmaktadır. Şekil 6.4'de görüldüğü gibi ilk bölüm santralleri ve santrallerin hangi saldırı tiplerine karşı korunduğu bilgisini, ikinci bölüm trafo merkezlerini ve hangi saldırı tiplerine karşı korunduğu bilgisini, son bölüm ise iletim hatlarını ve hangi saldırı tiplerine karşı korunduğu bilgisini göstermektedir.

Santraller				Trafo Merk.						İletim Hatları														
Tip1		Tip2		Tİp1			Tip2			Tip 1							Tip2							
1	2	1	2	1	2	3	1	2	3	1	2	3	4	5	6	7	1	2	3	4	5	6	7
1	0	1	1	1	1	0	1	1	0	1	1	0	1	1	1	0	1	1	1	1	1	1	0

Şekil 6.4. Üç Seviyeli Elektrik Şebeke Problemlerine Ait Kromozom Yapısı Örnek Kromozom

Şekil 6.4'de verilen örnekte tüm santral, trafo merkezleri ve iletim hatları için Tip 1 ve Tip 2 olmak üzere 2 tip saldırının olabileceği bir kromozom yapısı verilmiştir. Renkli numaralar sırasıyla santral (2 adet), trafo merkezi (3 adet) ve iletim hattı (7 adet) numaralarıdır. Bu numaralar için ayrılmış genler aslında söz konusu birimin söz konusu (Tip 1 veya Tip 2) tehdide karşı korunup korunmadığına göre aldığı 0 ya da 1 değerini barındıracaktır. Buna göre örneğin pembe ile gösterilmiş ilk 1 ve 2 değerlerinin altındaki örnek dizide yer alan 1 ve 0 sırasıyla 1. santralin Tip 1 saldırısına karşı korunduğunu, 2. santralin ise Tip 1 saldırısına karşı korunmadığını göstermektedir.

Ele alınan problemde koruma ve saldırı kapasiteleri sonludur. Bu nedenle çaprazlama ve mutasyon işlemleri sonucunda ortaya çıkan kromozom yapılarının bu kısıtı (uygunluğu) sağlayacak şekilde kalması mümkün kılınmaktadır.

Başlangıç Popülasyonu:

Başlangıç popülasyonu iki şekilde belirlenmiştir. İlkinde başlangıç popülasyonu mevcut koruma kombinasyonları içinden rassal olarak seçilerek oluşturulmuştur ve bu sezgisel yaklaşım, r_GA sezgisel algoritması olarak adlandırılmıştır. Diğer yöntemde ise koruma kombinasyonları içinden daha başarılı olması beklenen bireylerden bir seçim yapılarak bir popülasyon üretilmiştir. Bu sezgisel yaklaşım da k_GA sezgisel algoritması olarak adlandırılmıştır. Her iki yöntemde de koruma kapasitesi kısıtı göz önüne alınmıştır.

Başarılı olması beklenen bireylerden oluşan popülasyonda bireylerin belli bir kısmı kural tabanlı bir yöntem ile diğer bireyler ise yine rassal olarak oluşturulmuştur. Kural tabanlı seçimde Şekil 6.5'de görüleceği gibi öncelikle rassal olarak üretilen kromozomlardan belirlenmiş olan kurallara göre bir değerlendirme yapılarak bir birey seti oluşturulmuştur.

Şekil 6.5. Kural Tabanlı Seçim

Bunun için aşağıdaki kurallar uygulanmıştır:

- En fazla bağlantıya sahip ve kapasitesi en yüksek olan trafo merkezleri,
- Kritiklik değeri en yüksek olan bölgeler ile bağlantıyı sağlayan iletim hatları,
- Kapasitesi en yüksek olan santraller,
- Herhangi bir bileşen eğer bir tipe karşı korunuyor ise diğer saldırı tiplerine karşıda korunan bileşenlerin (tam koruma)

korunması önceliklidir. Bu kurallar elektrik dağıtımı konusunda uzman kişiler ile yapılan görüşmeler göz önüne alınarak belirlenmiştir.

Bu kurallara göre oluşturulan bireylere öncelik verilerek, başlangıç popülasyonunun bir kısmı bu şekilde oluşturulmuş kalanı ise rassal olarak seçilerek, başlangıç popülasyonuna dahil edilmiştir (Şekil 6.6)

Şekil 6.6. İmtiyazlı Bireylere Sahip Başlangıç Popülasyonunun Oluşturulması

Her iki yöntem içinde başlangıç popülasyon büyüklüğü ve k_GA için kural tabanlı seçilen birey oranı ileride bölüm 6.3'de gerçekleştirilen deney tasarımı çalışmasının sonuçları ile belirlenecektir.

Seçim:

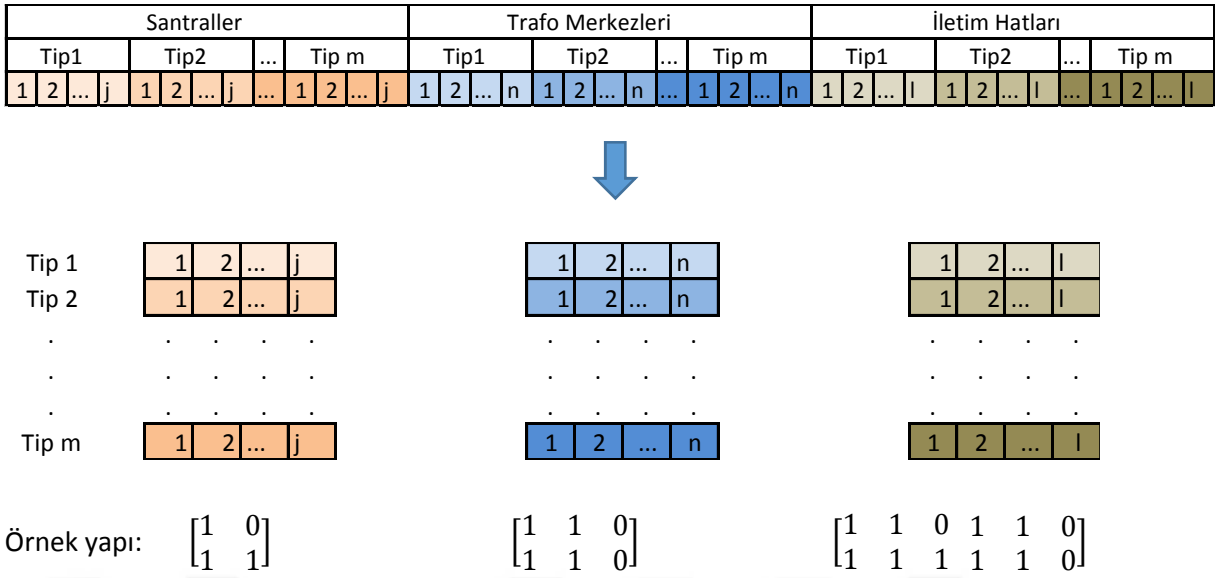
Başlangıç popülasyonunda hangi bireylerin çaprazlamaya ve mutasyona tabi tutulacağının belirlenmesi için rassal bir seçim uygulanmıştır. Rassal seçim yöntemine ek olarak Rulet tekerleği seçimi (Holand, 1975), deterministik seçim, turnuva yöntemi gibi

genetik algoritmalarda çok kullanılan yöntemler mevcuttur. Problemde, yeni popülasyon oluşturulmasında en iyi bireylerin belli bir kısmının popülasyon içinde tutulması nedeni ile çaprazlama ve mutasyon için rassal seçim yapılması tercih edilmiştir.

Çaprazlama:

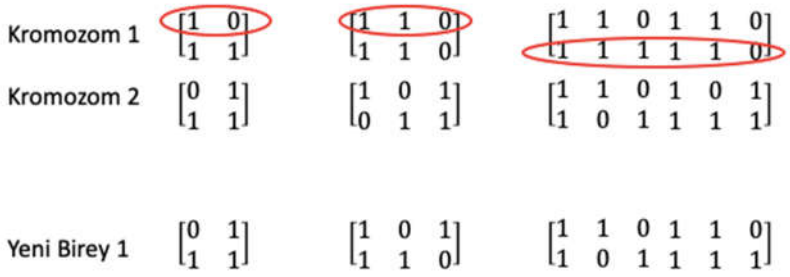
Çaprazlama, ata kromozomların iyi parçalarından yeni kromozomlar oluşturulması ve böylelikle daha iyi kromozomlar elde edilmesi beklentisine dayanmaktadır (Obitko, 1998). Çaprazlama işlemi ile kromozomların mevcut potansiyelini inceleyebiliriz. Çalışmada ikili kodlama şeklinde oluşturulan kromozomların çaprazlama sonrası özel işlemci geliştirmesine ve genetik onarım işlemine tabii tutulmasına gerek kalmaması amacıyla çaprazlama işleminin bir kromozomda üç farklı noktadan ve kendi içlerinde olacak şekilde yapılmasına karar verilmiştir. Birden çok çaprazlama noktası kullanılması genetik algoritmanın performansını azaltmakta ancak problem uzayının baştan sona incelenmesini sağlamaktadır (Sivanandam ve Deepa, 2008).

Kromozomlar problemin yapısına uygun olarak matris şeklinde ele alınmıştır. Her bir kromozomda santral, trafo merkezi ve iletim hattı ile temsil edilen üç bölüm vardır, bunların her birisi matris olarak düşünülmüştür ve her bir matrisin satırlarını koruma/saldırı tipleri temsil etmektedir (Şekil 6.7).



Şekil 6.7. Kromozom Yapısının Matris Gösterimi

Matris şeklinde gösterilen kromozomlar her bir öğeye ait matris için belirlenen bir noktadan çaprazlanarak ve ilgili gen bölümü yer değiştirilerek yeni bireyler oluşturulmaktadır. Şekil 6.8'de görüldüğü gibi kromozom 1 üzerinde rassal olarak seçilen noktalar, kromozom 2'den alınarak yeni birey oluşturulmaktadır.



Şekil 6.8. Çaprazlama İşlemi Örnek Şematik Gösterimi

Santraller, trafo merkezleri ve iletim hatları bölümlerini temsil eden kromozomun 3 bölümünden, kaçında çaprazlama yapılacağı (Çaprazlama Nokta Sayısı) gerçekleştirilen deney tasarımı ile belirlenmiştir. Çaprazlama sayısı kadar ilgili bölümlerde rassal olarak çaprazlama işlemi gerçekleştirilir.

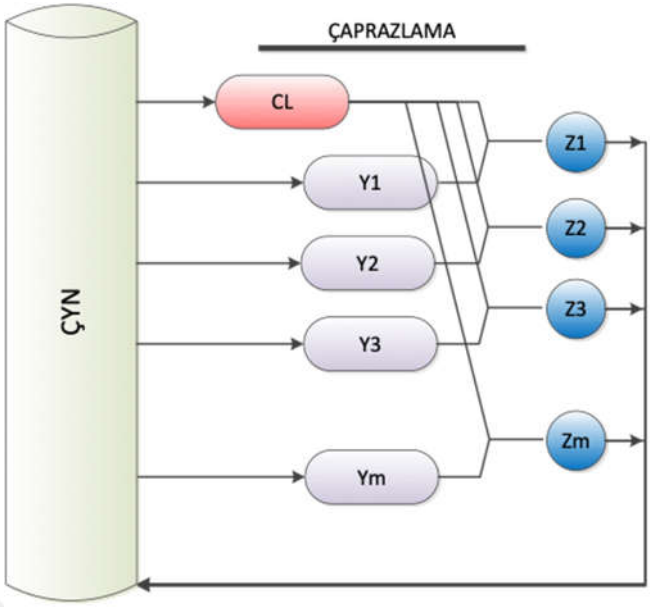
Yerel Arama Algoritması:

Önerilen Genetik Algoritma temelli sezgisel algoritmaların bir yerel arama algoritması ile daha etkin çalışabilmesi ve yerel arama yaklaşımının üstünlüklerinden yararlanmak amacıyla çaprazlama aşaması sonrası literatürde yer alan Binary-coded Local Genetic Algorithm (BLGA) tabanlı bir yaklaşımı (Martinez C.G. ve Lozano M., 2010,) kullanılmıştır. Böylece en iyi bireyin komşuları yerel arama yöntemi ile taranmaktadır. BLGA Algoritmasının adımları aşağıdaki gibidir:

1. Popülasyon içinden rastgele bir birey seçer (C^L)
2. Eş Seçimi: Popülasyon içinden C^L'ye benzer m tane birey (Hamilton mesafesi yöntemi ile benzer bireyler belirlenir) seçilir.
3. Popülasyondan seçilen eş ile C^L çaprazlanarak yeni birey türetilir. Yeni birey kısa süreli hafızaya atılır ve tekrar eden birey ise tekrar çaprazlanır.
4. Yeni birey eğer C^L'den daha iyi ise C^L ile yer değiştirir.
5. Durdurma kriterine kadar 3 ve 4'ü tekrarla ve C^L'yi popülasyona ekle.

Bu çalışmada komşu arama için önerilen algoritma adımları ise aşağıda verilmiştir. Genetik algoritmaların yerel arama ile bazı çözümlerin etrafının taranarak daha iyi çözümlere ulaşması amacıyla önerilen bu algoritma, hem r_GA hem de k_GA sezgisel algoritmalarında kullanılmıştır. Şekil 6.9'da önerilen komşu arama algoritmasının şematik gösterimi sunulmuştur.

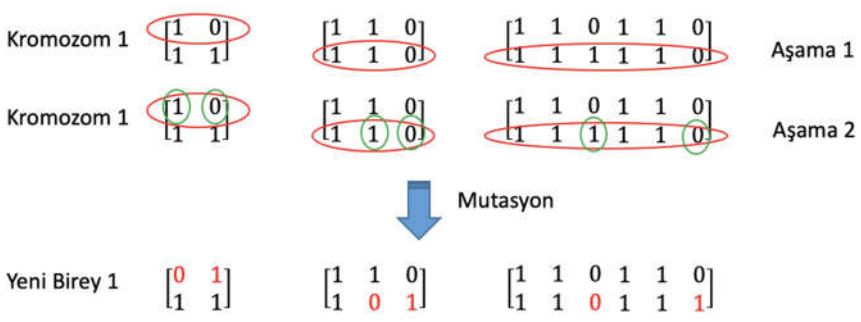
1. Çaprazlama sonrası elde edilen yeni nesil (ÇYN) içinden en iyi birey seçilir (C^L).
2. Eş Seçimi: Çaprazlama sonrası elde edilen bireyler (ÇYN) içinden C^L'ye benzer m tane birey (Hamilton mesafesi yöntemi ile benzer bireyler belirlenir) seçilir.
3. Popülasyondan seçilen eş ile C^L çaprazlayanarak yeni birey türetilir.
4. Yeni birey, ÇYN'deki bireyler ile aynı ise 3 tekrarlanır.
5. Yeni birey ÇYN'ye ilave edilir.
6. 3 ve 4'ü m kez tekrarla ve bitir.



Şekil 6.9. Önerilen Yerel Arama Algoritması

Mutasyon:

Rastgele iki genin seçilerek, bu genlerdeki bitlerin değiştirilmesi ya da rastgele bir gen seçerek, bu genden sonraki bitlerin terslerinin alınması gibi mutasyon yaklaşımları bulunmaktadır (Sivanandam ve Deepa, 2008). Bu çalışmada, mutasyon işlemi, çaprazlama işleminde olduğu gibi yine her bir öğe (Santral, Trafo Merkezi, İletim Hattı) için gerçekleştirilebileceği 3 farklı nokta belirlenmiştir. Şekil 6.10'da görüldüğü gibi ilk aşamada her bir matris için değişikliğin yapılacağı satır seçilir. Sonrasında o satırda bir adet 0 değeri 1'e, bir adet 1 değeri de 0'a dönüştürülür.



Şekil 6.10. Mutasyon İşlemi Örnek Şematik Gösterimi

GA' da mutasyon, seçim süreci sırasında popülasyonda kaybolmuş genleri yerine koyma veya başlangıç popülasyonunda bulunmayan gen dizilimlerini ortaya çıkarma gibi kritik görevleri yerine getirmektedir (Gen ve Cheng,1997). Hamilton ve Ridley (2005) GA'da mutasyon oranının doğada olduğu gibi genellikle oldukça düşük olduğunu belirtmektedir. r_GA ve k_GA algoritmaları için mutasyon oranını belirlemek üzere deney tasarım analizinde ilgili parametre için üç düzey belirlenmiştir. Bu düzeyler % 0,2, % 1 ve % 1,5' gibi düşük mutasyon oranlarıdır. Bölüm 6.3'de gerçekleştirilen Deney Analizi sonucunda her iki sezgisel algoritma için de daha iyi performans veren düzey ayrı ayrı belirlenmiştir.

Yeni Popülasyonun Oluşturulması:

Yeni popülasyon oluşturulurken çaprazlama ve mutasyon sonucu oluşan yeni bireylerin bir kısmı ya da tamamı yeni popülasyona aktarılır. Burada en iyi değerleri kaybetmemek adına bazı elitist seçimler yapılmaktadır. Elitist seçim ile en iyi kromozomlar, yeni topluma aktarılmakta, ardından toplumun geri kalanı oluşturulmaktadır (Obitko, 1998). Çok sıkı bir seçim gerçekleştirilmesi, toplumda en iyinin altında, oldukça uygun bireylerin sayıca artmasına ve ilerleme ve değişim için gerekli çeşitliliğin azalmasına neden olacaktır. Zayıf bir seçim gerçekleştirilmesi durumunda ise evrim oldukça yavaşlayacaktır (Mitchell, 1999).

Çaprazlama ve mutasyon işlemlerinden sonra mevcut popülasyonun en iyi değerlerinin bir sonraki nesle aktarılmama riskine karşılık, yeni popülasyonun oluşturulmasında öncelikle yeni bireylerin uygunluk değerleri hesaplanır ve başlangıç popülasyonu ile birlikte yukarıdan aşağıya sıralanır (eşit uygunluk değerlerinde yeni bireyler üste yazılır). Oluşturulan bu sıralamanın belirli bir yüzdelik kısmı alınır ve yeni popülasyonun ilk kısmı oluşturulur. Kalan kısım çaprazlama ve mutasyon sonucu oluşmuş diğer bireylerin arasından rassal olarak seçilir (mutasyon sonucu oluşan tüm bireyler yeni popülasyona eklenir).

Koruma kombinasyonlarının uygunluk değerleri hesaplanırken her bir koruma kombinasyonu için olası tüm yasaklama kombinasyonları belirlenir ve Bölüm 5.4'de detayları verildiği gibi seviye 3'e gönderilerek, yasaklama kombinasyonunun uygunluk

değeri belirlenir. Geliştirilen bu sezgisel algoritmalarda yasaklama kombinasyonlarının uygunluk değerleri program tarafından kayıt altına alınmaktadır ve eğer bir yasaklama kombinasyonu daha önce kayıt edildi ise tekrar seviye 3'e gönderilmemektedir. Bu da programın performansına katkı sağlamaktadır.

6.3. r_GA ve k_GA Algoritmalarının Parametrelerinin Deney Tasarımı ile Belirlenmesi

Bu bölümde GA temelli r_GA ve k_GA Algoritmalarının parametrelerin en iyi değerlerini belirlemek üzere Taguchi yöntemi ile bir deney tasarımı yapılmış ve Minitab paket programı ile sonuçları analiz edilmiştir. r_GA algoritması 5 parametresinin belirlenmesi için 5 faktör 3'er düzey ve k_GA algoritmasının 6 parametresinin belirlenmesi için 6 faktör 3'er düzeyden oluşan iki deney tasarımı çalışması yapılmıştır. Bu deney tasarımı çalışmasında Taguchi ortogonal dizinlerinden (L27) kullanılarak deneyler tasarlanarak analiz edilmiş ve faktörlerin uygun düzeyleri belirlenmiştir. Her test 4'er kez tekrarlanmış ve her test 2 dakika süre ile çalıştırılmıştır.

Taguchi yöntemi Genichi Taguchi tarafından geliştirilmiş bir istatistiksel metottur. Deney sayısını enküçüklemeyi amaçlayan bu yöntem, sistemin tasarlanması, parametrelerin tasarlanması ve toleransların tasarlanması şeklinde üç aşamadan oluşmaktadır (Taguchi vd., 2000).

r_GA Sezgisel Algoritması Parametreleri için Deney Tasarımı

r_GA'nın en iyi performansı göstereceği parametre değerlerini belirlemek üzere deney şu şekilde tasarlanmıştır. 5 faktör 3'er düzeyli olmak üzere Çizelge 6.1'de görüldüğü gibi tasarlanmıştır. Minitab 19.2020.1 programında, Taguchi yöntemi ile (L27) Çizelge 6.2'deki 27 test üretilmiştir ve her test 4 'er kez tekrarlanmıştır. Her test için elde edilen sonuçların ortalaması Çizelge 6.2'de sonuçlar sütununda verilmiştir.

Çizelge 6.1. r_GA Algoritması Deney Tasarımına Esas Teşkil Eden Parametreler ve Düzeyler

Faktör No	Faktör	Seviyeler
A1	Başlangıç Popülasyonu Büyüklüğü	20, 30, 50
B1	Çaprazlama Oranı (%)	30, 50, 60
C1	Çaprazlama Nokta sayısı	1, 2, 3
D1	Mutasyon Oranı (%)	(0,2), 1, (1,5)
E1	En İyi Birey Oranı (%)	0, 30, 60

Çizelge 6.2. r_GA Algoritması Taguchi (L27) Ortogonal Dizin Test Sonuçları

	Faktörler ve Düzeyleri					
Test No	**A1**	**B1**	**C1**	**D1**	**E1**	**Sonuçlar (z*)**
1	20	30	1	0.2	0	0,42625
2	20	30	1	0.2	30	0,33875
3	20	30	1	0.2	60	0,19625
4	20	50	2	1	0	0,27375
5	20	50	2	1	30	0,33875
6	20	50	2	1	60	0,2725
7	20	60	3	1.5	0	0,30125
8	20	60	3	1.5	30	0,16375
9	20	60	3	1.5	60	0,23
10	30	30	2	1.5	0	0,3825
11	30	30	2	1.5	30	0,16375
12	30	30	2	1.5	60	0,256
13	30	50	3	0.2	0	0,58625
14	30	50	3	0.2	30	0,1525
15	30	50	3	0.2	60	0,46
16	30	60	1	1	0	0,30625
17	30	60	1	1	30	0,13125
18	30	60	1	1	60	0,3825
19	50	30	3	1	0	0,44025
20	50	30	3	1	30	0,5635
21	50	30	3	1	60	0,2725
22	50	50	1	1.5	0	0,53525
23	50	50	1	1.5	30	0,465
24	50	50	1	1.5	60	0,45275
25	50	60	2	0.2	0	0,816667
26	50	60	2	0.2	30	0,6575
27	50	60	2	0.2	60	0,65625
* r_GA'nın amaç fonksiyonu değeri						

Minitab programına, Taguchi Yöntemi ile deneylerin belirlenmesi için program arayüzünden Stat-DOE-Taguchi-Create Taguchi Design seçilerek deneye ait bilgiler ve hangi analizlerin yapılacağı arayüz yardımı ile girilmiştir. Taguchi, tercih edilen kalite karakteristiği olarak sinyal-gürültü (S/N) oranını kullanmıştır. S/N oranı, standart sapma yerine ölçülebilir bir değer olarak kullanılır. Analiz sürecinde S/N oranı seçiminde "daha küçük daha iyi" (smaller is better) seçilerek girilmiştir. Bu parametre 6.1'deki gibi tanımlanmaktadır:

$$\text{S/G Oranı} = -10xLog10(\sum(Y^2)/n) \qquad (6.1)$$

Program her bir test için sinyal gürültü oranlarını (SN) vermiştir. Ayrıca EK-B'de ilgili diğer istatistikler verilmiştir. Sinyal/gürültü oranları için sonuçlar çizelge Çizelge 6.3'de sunulmuştur.

Çizelge 6.3. Sinyal Gürültü Oranı Sonuçları

Daha Küçük – Daha İyi

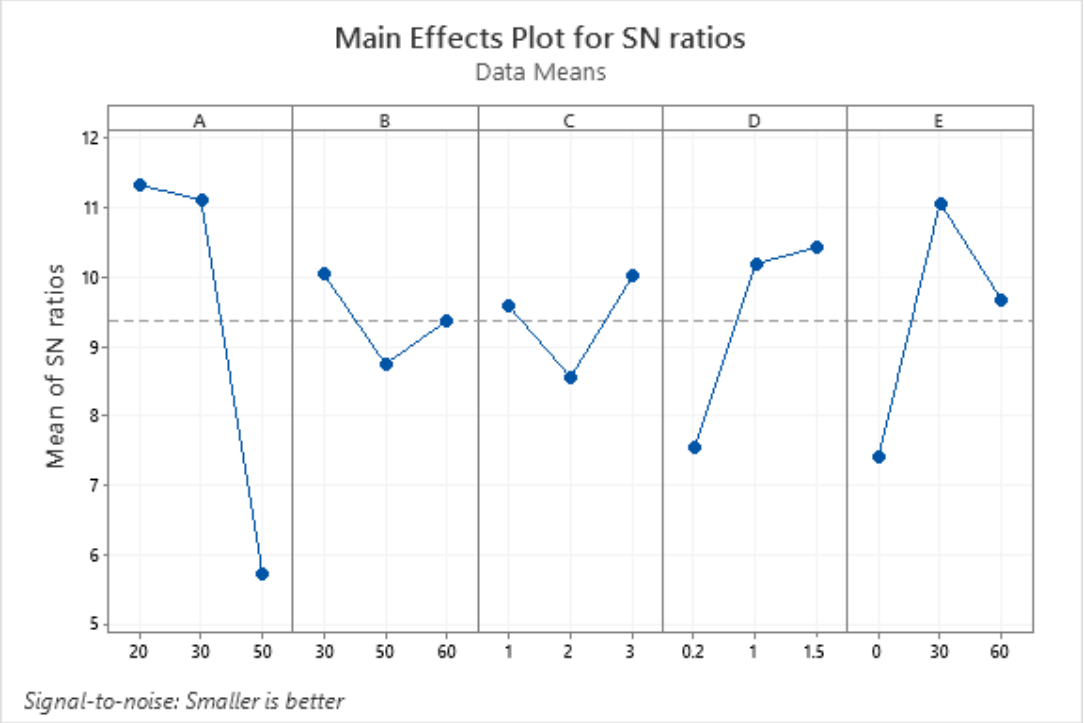
Level	**A**	**B**	**C**	**D**	**E**
1	**11,312***	**10,028***	9,575	7,526	7,407
2	11,098	8,736	8,545	10,179	**11,054***
3	5,714	9,359	**10,002***	**10,418***	9,663
Delta	5,598	1,292	1,457	2,893	3,647
Rank	1	5	4	3	2

Sinyal gürültü oranı sonuçlarını incelediğimizde r_GA algoritması için sonucu en çok etkileyen parametrelerin önem sırasına göre Popülasyon Büyüklüğü (A), En İyi Birey Oranı (E), Mutasyon Oranı (D), Çaprazlama Nokta Sayısı(C) ve Çaprazlama Oranı (B) olduğu görülmektedir. Analiz sonucuna göre ilgili parametreler için belirlenen en iyi düzeyler Çizelge 6.4'de görüldüğü gibidir:

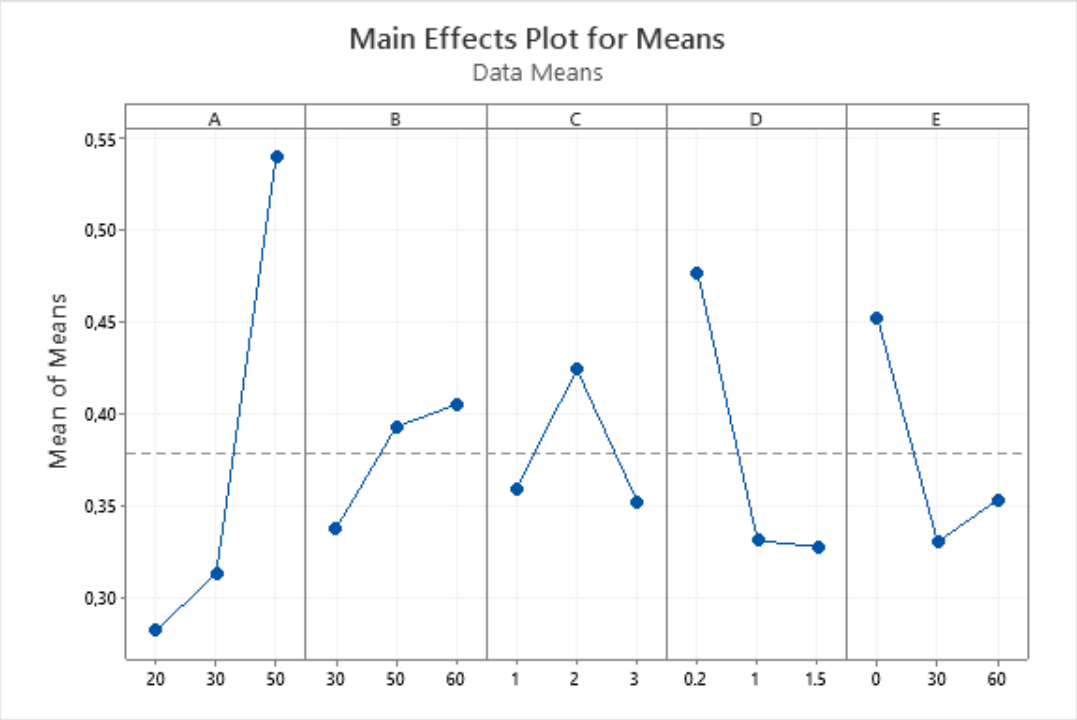
Çizelge 6.4. r_GA algoritması için Taguchi Deney Tasarımı ile Belirlenen Parametre Değerleri

Faktör No	Faktör	En İyi Seviyeler
A1	Başlangıç Popülasyonu Büyüklüğü	20
B1	Çaprazlama Oranı (%)	30
C1	Çaprazlama Nokta sayısı	3
D1	Mutasyon Oranı (%)	1,5
E1	En İyi Birey Oranı (%)	30

Bu sonuçlara ilişkin olarak "Sinyal gürültü oranları için sonuçlar grafiği" ve "Ortalamalar için sonuçlar grafiği" sırasıyla Şekil 6.11 ve Şekil 6.12'de sunulmuştur. SN grafiğinde her bir parametre için en büyük ortalamaya sahip düzey en iyi değer iken ortalamaların ortalamaları grafiğinde ise en küçük değere sahip düzey eniyi değerdir. Her iki grafikte de parametreler için en iyi değerin aynı olduğu görülmektedir.



Şekil 6.11. r_GA Algoritması Sinyal/Gürültü Oranı için Sonuçlar



Şekil 6.12. r_GA Algoritması Ortalamalar için Sonuçlar

k_GA Sezgisel Algoritması Parametreleri için Deney Tasarımı

k_GA'nın en iyi performansı göstereceği parametre değerlerini belirlemek üzere deney şu şekilde tasarlanmıştır. 6 faktör 3'er düzeyli olmak üzere Çizelge 6.5'de görüldüğü gibi tasarlanmıştır. Minitab programında, Taguchi yöntemi ile (L27) Çizelge 6.6'daki 27 test üretilmiştir ve her test 4 'er kez tekrarlanmıştır. Her test için elde edilen sonuçların, ortalaması Çizelge 6.6'da sonuçlar sütununda verilmiştir.

Çizelge 6.5. k_GA algoritması Deney Tasarımına Esas Teşkil Eden Parametreler ve Düzeyler

Faktör No	Faktör	Seviyeler
A2	Başlangıç Popülasyonu Büyüklüğü	20, 30, 50
B2	Çaprazlama Oranı (%)	30, 50, 60
C2	Çaprazlama Nokta sayısı	1, 2, 3
D2	Mutasyon Oranı (%)	(0,2), 1 , (1,5)
E2	En İyi Birey Oranı (%)	0, 30, 60
F2	Kural tabanlı birey oranı (%)	10, 40, 60

Çizelge 6.6. k_GA Algoritması Taguchi (L27) Ortogonal Dizin Test Sonuçları

	Faktörler ve Düzeyleri						
Test No	**A1**	**B1**	**C1**	**D1**	**E2**	**F2**	**Sonuçlar (z*)**
1	20	30	1	0.2	0	10	0,54875
2	20	30	1	0.2	30	40	0,16375
3	20	30	1	0.2	60	60	0,08750
4	20	50	2	1	0	10	0,29875
5	20	50	2	1	30	40	0,24000
6	20	50	2	1	60	60	0,41800
7	20	60	3	1.5	0	10	0,27575
8	20	60	3	1.5	30	40	0,20750
9	20	60	3	1.5	60	60	0,22775
10	30	30	2	1.5	0	40	0,47000
11	30	30	2	1.5	30	60	0,51375
12	30	30	2	1.5	60	10	0,60725
13	30	50	3	0.2	0	40	0,80500
14	30	50	3	0.2	30	60	0,47000
15	30	50	3	0.2	60	10	0,25675
16	30	60	1	1	0	40	0,49250
17	30	60	1	1	30	60	0,53125
18	30	60	1	1	60	10	0,45875
19	50	30	3	1	0	60	0,51875
20	50	30	3	1	30	10	0,64875
21	50	30	3	1	60	40	0,40375
22	50	50	1	1.5	0	60	0,39250
23	50	50	1	1.5	30	10	0,54075
24	50	50	1	1.5	60	40	0,78375
25	50	60	2	0.2	0	60	0,49025
26	50	60	2	0.2	30	10	0,45300
27	50	60	2	0.2	60	40	0,47000
* k_GA'nın amaç fonksiyonu değeri							

Minitab programına, Taguchi Yöntemi ile deneylerin belirlenmesi için program arayüzünden Stat-DOE-Taguchi-Create Taguchi Design seçilerek deneye ait bilgiler ve hangi analizlerin yapılacağı arayüz yardımı ile girilmiştir. Ele aldığımız problem bir en

küçükleme problemidir ve sonuç değerlerinden en küçüğünün en iyi değer olduğu programa, analiz opsiyonlarından "daha küçük daha iyi" (smaller is better) seçilerek girilmiştir.

Program her bir test için sinyal gürültü oranlarını (SN) vermiştir. Ayrıca EK-B'de ilgili diğer istatistikler verilmiştir. Sinyal/gürültü oranları için sonuçlar Çizelge 6.7'de sunulmuştur.

Çizelge 6.7. Sinyal Gürültü Oranı Sonuçları

Daha Küçük – Daha İyi

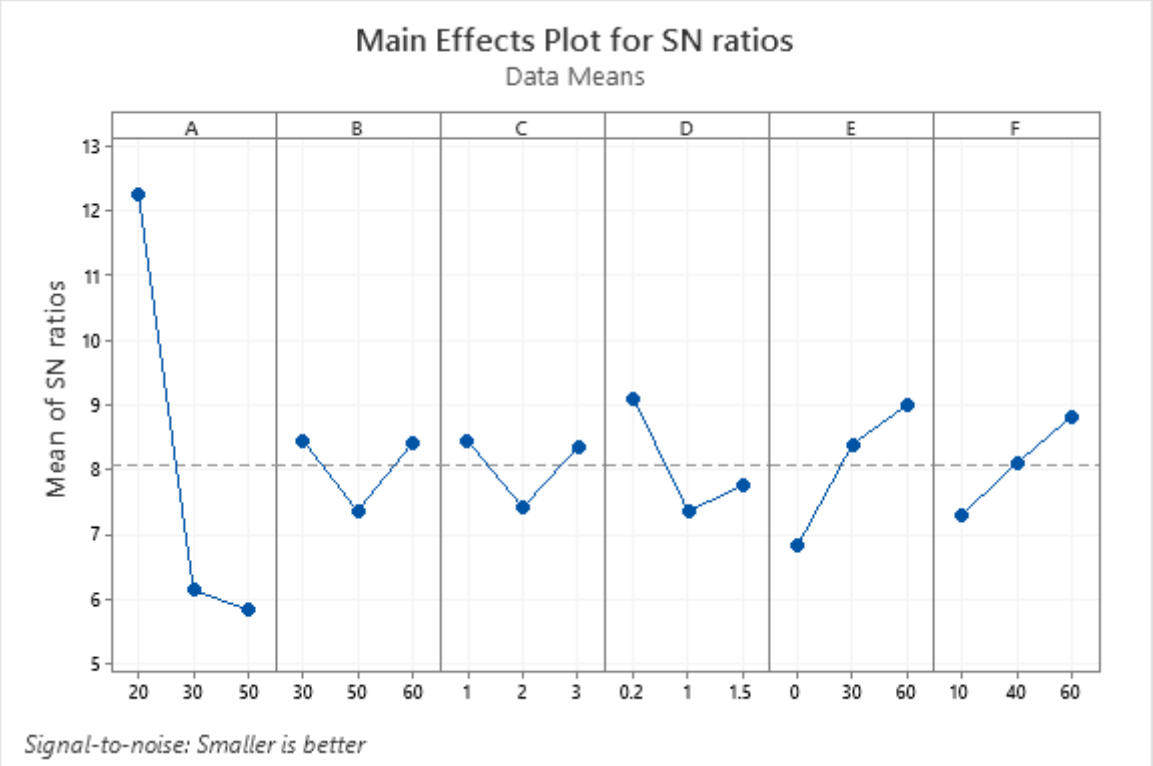
Level	**A**	**B**	**C**	**D**	**E**	**F**
1	**12,251***	**8,456***	**8,454***	**9,108***	6,834	7,309
2	6,149	7,366	7,419	7,358	8,398	8,102
3	5,838	8,416	8,365	7,773	**9,006***	**8,827***
Delta	6,412	1,089	1,035	1,750	2,172	1,517
Rank	1	5	6	3	2	4

Sinyal gürültü oranı sonuçlarını incelediğimizde k_GA algoritması için sonucu en çok etkileyen parametrelerin önem sırasına göre Popülasyon Büyüklüğü (A), En İyi Birey Oranı (E), Mutasyon Oranı (D), Kural Tabanlı Birey Oranı (F), Çaprazlama Oranı (B) ve Çaprazlama Nokta Sayısı (C) olduğu görülmektedir. Analiz sonucuna göre ilgili parametreler için belirlenen değerler Çizelge 6.8'de görüldüğü gibidir:

Çizelge 6.8. k_GA algoritması için Taguchi Deney Tasarımı ile Belirlenen Parametre Değerleri

Faktör No	Faktör	En İyi Seviyeler
A2	Başlangıç Popülasyonu Büyüklüğü	20
B2	Çaprazlama Oranı (%)	30
C2	Çaprazlama Nokta sayısı	1
D2	Mutasyon Oranı (%)	0,2
E2	En İyi Birey Oranı (%)	60
F2	Kural tabanlı birey oranı (%)	60

Bu sonuçlara ilişkin olarak “Sinyal gürültü oranları için sonuçlar grafiği” ve “Ortalamalar için sonuçlar grafiği” Şekil 6.13 ve Şekil 6.14’de sunulmuştur. SN grafiğinde her bir parametre için en büyük ortalamaya sahip düzey en iyi değer iken ortalamaların ortalamaları grafiğinde ise en küçük değere sahip düzey eniyi değerdir. Her iki grafik sonucunda da parametreler için en iyi değerin B ve C faktörleri dışında aynı olduğu görülmektedir. B ve C faktörlerinde sonucu etkileme oranın en düşük olan parametredir. Bu nedenle SN oranları göz önüne alınarak parametre düzeyleri belirlenmiştir.



Şekil 6.13. k_GA Algoritması Sinyal/Gürültü Oranı için Sonuçlar

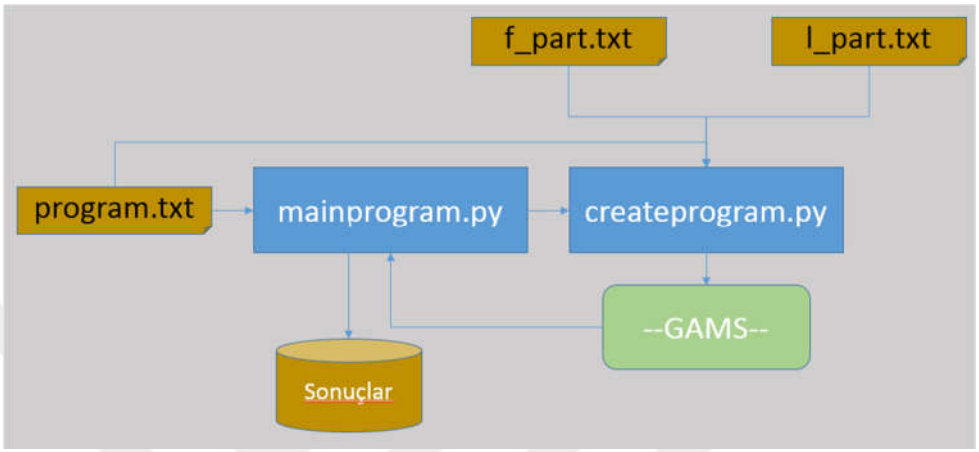
Şekil 6.14. k_GA Algoritması Ortalamalar için Sonuçlar

6.4. Çözüm Yaklaşımının Kodlanması, Kullanılan Yazılım Dili ve Çözücüler

"Üç Seviyeli Çoklu Saldırı Tipli Elektrik Şebekesi Koruma Modeli" Çözüm Algoritması ve büyük problemler için geliştirilen Genetik Algoritma Tabanlı Sezgisel Yaklaşım**,** PyCharm Professional 2018.3 derleyicisi kullanılarak Python programlama dilinin 2.6.1 versiyonu ile kodlanmıştır. Kodlanan program GAMS 24.0.2 programı ile etkileşimli olarak çalışmaktadır. Modelin üst seviye ve orta seviyesine karşılık gelen ve kombinasyonların türetildiği kısımlar Python'da ana program (mainprogram.py) aracılığı ile üretilmektedir.

Problemin kendine özgü parametreleri (program.txt) ve GAMS 24.0.2'ye üzerinde çalışacak olan kodları (ilk parça (f_part.txt) ve son parça (l_part.txt)) programa başlangıçta verilmektedir. Ana program türetilen kombinasyonları çözüm algoritmasına uygun olarak türetmektedir. Türetilen her bir yasaklama kombinasyonu için kombinasyon bilgisi ve diğer parametre bilgileri Pyton'da hazırlanmış olan createprogram.py bölümünde işlenerek hazırlanan bu dosya ana program tarafından GAMS 24.0.2'ye gönderilir ve sonuç bilgisi

Ana program tarafından kayıt altına alınır. Ana program bu işlemi her bir koruma ve yasaklama kombinasyonu için tekrarlayarak eniyi koruma kombinasyonunu ve amaç fonksiyonu değerini belirler. Genel hatları ile çözüm için tasarlanan sistem akış diyagramı ve temel bileşenler Şekil 6.15'de verilmiştir.



program.txt	Parametrelerin ve set büyüklerinin tanımlandığı kısım
mainprogram.py	Kombinasyonları türeten verileri gönderen ve alan program
createprogram.py	GAMS koduna dönüştüren program
f_part.txt	GAMS kodu (Değişkenlerin tanımlandığı kısım)
l_part.txt	GAMS kodu (Denklem ve çıktı operatörlerinin tanımlandığı kısım

Şekil 6.15. Önerilen Çözüm Yaklaşımında Yer Alan Temel Bileşenler

Büyük boyutlu problemlerin çözümü için geliştirilen Genetik Algoritma Tabanlı Sezgisel Yaklaşım Şekil 6.10'da verilen yapıyı koruyarak hazırlanmıştır. r_GA ve k_GA sezgisel algoritmaları tüm kombinasyonların taranması yerine GA'nın operatörleri yardımı ile belirlenen kombinasyonlar taranarak eniyi çözümü bulmayı amaçlamaktadırlar. Bu amaçla Şekil 6.15'daki yapıya uygun olarak Python ile önceden geliştirilmiş yaklaşım, GA'ya ait sözü edilen işlemlerin entegre edilmesi ile geliştirilmiştir. Kombinasyonların üretilmesi ve seçimi ve GA'nın operatörleri (çaprazlama, mutasyon vd.) ne karşı gelen işlemler tekrar kodlanmıştır. Yine başlangıç popülasyonunun kural tabanlı olarak belirlenmesine yönelik önerilen sezgisel algoritma için de programda belirlenen kurallara uyan kombinasyonların seçildiği bir kısım kodlanarak programa eklenmiştir. Çözüm yaklaşımlarına ait Python kodları Ek-C'de verilmiştir. Bölüm 7.2'de gerçekleştirilen

deneysel çalışmalarda GA ile yapılan çözüm ile kural tabanlı sezgisel algoritmanın çözüm performansları karşılaştırılmıştır.

Çözümde kullanılan eniyileme aracı GAMS 24.0.2 (Genel Cebirsel Modelleme Sistemi- General Algebraic Modelling System), içinde birçok çözücü seçeneği bulunduran arayüz görevini gören bir yazılım paketidir. GAMS Development Corporation tarafından kurulmuştur. GAMS modelleme ve eniyileme problemlerinin çözümü için kullanılan yüksek seviyeli bir programlama dilidir. Bir dil derleyicisinden ve yüksek performanslı kararlı çözücülerden oluşur. GAMS, karmaşık, büyük ölçekli problemlerin modellenmesi için tasarlanmıştır ve yeni durumlara hızlı bir şekilde adapte edilebilen modeller oluşturmaya olanak tanır. GAMS özellikle doğrusal, doğrusal olmayan ve karışık tamsayı eniyileme problemlerinin modellenmesi için tasarlanmıştır. GAMS ticari çözücüler de dahil olmak üzere 25'ten fazla çözücünün sunmaktadır. Bu çözücülerin kullanılabildiği modeller Çizelge 6.9'da verilmiştir (GAMS, 2019). Bu çalışmada GAMS çözücülerinden DICOPT kullanılmıştır.

Çizelge 6.9. GAMS Çözücüleri ve Çözüm Buldukları Matematiksel Programlama Tipleri

Matematiksel Programlama Tipi	Çözücü
LP / MIP / QCP / MIQCP	CPLEX, GUROBI, MOSEK, XPRESS
NLP:	CONOPT, IPOPTH, KNITRO, MINOS, SNOPT
MINLP	ALFAFEP, ANTİJON, BARON, DICOPT, OQNLP, SBB

Çözüm yaklaşımlarında kullanılan Python yazılım dili, 1990 yılından bu yana geliştirilmekte olan bir programlama dilidir. Python, Perl, Ruby, Scheme veya Java ile karşılaştırılabilir, açık ve güçlü bir nesne yönelimli programlama dilidir (Nesne Yönelimli Programlama (NYP) mantıksal işlemlerden ziyade, nesnelere (object) ve nesneler üzerinde işlemlere odaklanan programlama dili modelidir). Yazılan programları okumayı kolaylaştırmak için anlaşılır bir sözdizimi kullanır, kullanımı kolay bir dildir. Bu, Python'u, prototip geliştirme ve diğer geçici programlama görevleri için ideal hale getirir. Web sunucularına bağlanma, normal ifadelerle metin arama, dosyaları okuma ve değiştirme gibi birçok genel programlama görevini destekleyen büyük bir standart kütüphane ile birlikte

gelir. Ayrıca IDLE adlı paketlenmiş bir geliştirme ortamı da var. C veya C ++ gibi derlenmiş bir dilde uygulanan yeni modüller eklenerek kolayca genişletilebilir. Programlanabilir bir arayüz sağlamak için bir uygulamaya da gömülebilir. Mac OS X, Windows, Linux ve Unix dahil olmak üzere her yerde çalışır. Python'u indirmek ve kullanımı ücretsizdir. Python dili telif hakkıyla korunmakla birlikte açık kaynaklı bir lisans altında da kullanıldığı için serbestçe değiştirilebilir ve yeniden dağıtılabilir (Python, 2019). Python'ın şuan 3.8.3 sürümü aktiftir. Ancak bu çalışmada mevcut GAMS programı ile uyumlu olması amacıyla Python'un 2.6.1 sürümü kullanılmıştır.

7. ÇÖZÜM YAKLAŞIMLARININ DENEYSEL DEĞERLENDİRMESİ

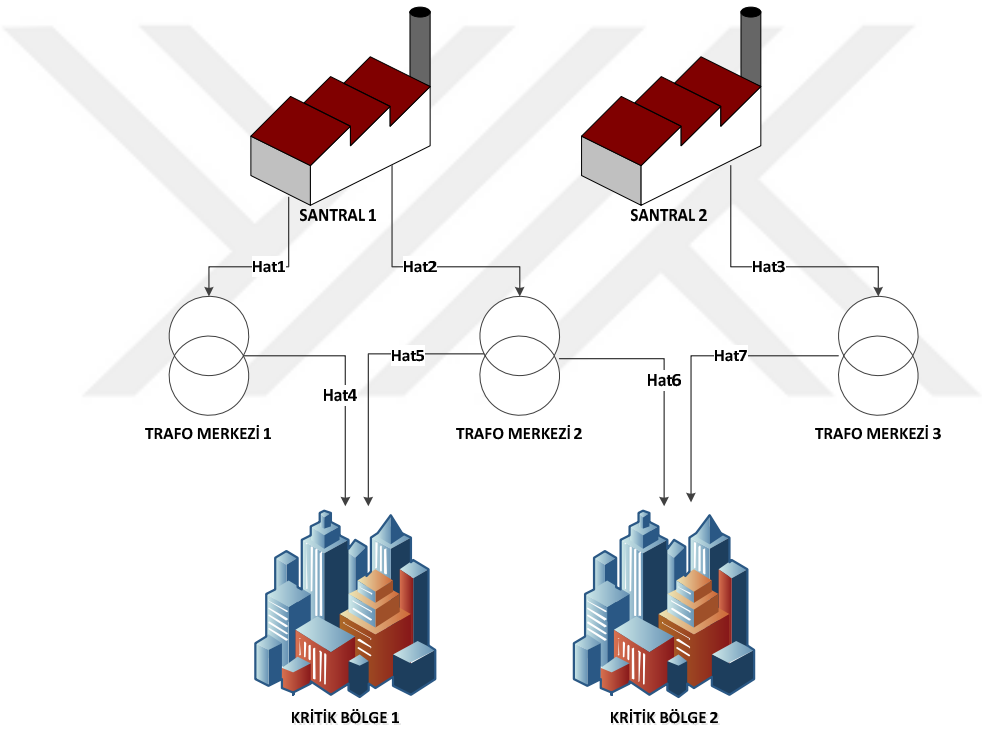
Bu bölümde, ele alınan problem için geliştirilen çözüm yaklaşımlarının deneysel sonuçları incelenmiştir. Bölüm 7.1'de örnek bir problemin matematiksel model ile çözümü için Bölüm 5.4'te önerilen çözüm yaklaşımı kullanılmış ve sonuçları paylaşılmıştır. Ayrıca bu örnek problem için çözüm uzayının büyüklüğünün koruma ve yasaklama kapasitelerinde yapılan değişikler ile nasıl değiştiği gösterilmiştir. Parametrelerde yapılan bu değişiklikler ile birlikte problem boyutunun çok hızlı bir şekilde büyüdüğü görülmektedir. Bölüm 6.2'de, önerilen Genetik Algoritma (GA) tabanlı sezgisel algoritmanın sonuçları ile bölüm 7.1'de elde edilen matematiksel model sonuçları karşılaştırılmıştır.

7.2'de literatürde sıklıkla kullanılan IEEE (The Institute of Electrical and Electronics Engineers)'nin 14 baralı test problemi, geliştirilen matematiksel modele uyarlanmış ve büyük ölçekli problemlerin çözümü için önerilen GA tabanlı sezgisel algoritmaların bu veri seti kullanılarak performansları değerlendirilmiştir. Bu bölümde başlangıç popülasyonunun, rassal olarak üretildiği çözüm yaklaşımı r_GA ile kural tabanlı olarak üretildiği k_GA çözüm yaklaşımının karşılaştırılması, farklı durumlar göz önüne alınarak yapılmıştır. Bu durumlar, koruma kaynaklarının yoğunluğuna göre sıkı-orta ve gevşek olmak üzere üç farklı kategoriye ayrılmıştır. Hangi durumda hangi sezgisel algoritmanın daha iyi sonuç verdiği incelenmiştir.

7.3'de ise koruma kaynaklarının sıkı orta ve gevşek olarak kategorilere ayrılmasının istatistiksel olarak anlamlılığı Kruskal-Wallis non-parametrik testi ile analiz edilmiş ve farklılık olduğu gösterilmiştir. Bölüm 7.2'de k_GA algoritmasının her üç koruma kategorisi içinde daha iyi sonuçlar elde ettiği görülmüştü. 7.3'de bu farkın istatistiksel olarak anlamlı olup olmadığı her üç kategori için ayrı ayrı ele alınmıştır. Bu amaçla Wilcoxon İşaretli Sıralar Testi k_GA ve r_GA'nın performansları analiz edilmiştir. Sıkı ve orta koruma kategorileri için k_GA'nın r_GA'dan daha iyi performans gösterdiği istatistiksel olarak da anlamlı bulunmuştur. Gevşek kategoride ise k_GA'nın test sonuçlarında daha iyi sonuçlar elde etmesine rağmen algoritmalar arasında istatistiksel olarak bir fark görülmemiştir.

7.1. Üç Seviyeli Çoklu Saldırı Tipli Elektrik Şebekesi Koruma Probleminin Matematiksel Model Çözümü

Beşinci bölümde verilen matematiksel model ve önerilen çözüm yönteminin testi için aşağıdaki örnek problem oluşturulmuştur. Bu problemin şematik gösterimi Şekil 7.1'de, problem parametreleri ise Çizelge 7.1'de verilmiştir. Şekil 7.1 de verilen örnekte, 2 santral, 3 trafo merkezi ve 7 hattan oluşan bir sistem, 2 kritik bölgenin enerji tedarikini sağlamaktadır. Problemde santral ve trafo merkezleri 2 farklı saldırı tipi ile iletim hatları ise 1 saldırı tipi ile yasaklanabilir.



Şekil 7.1. Örnek Problemin Şematik Gösterimi

Çizelge 7.1. Örnek Problem için (a) Temel Bileşen Miktarları (b) Koruma Kaynakları Miktarları (c) Saldırı Kaynakları Miktarları

Bileşen	**Miktarı (adet)**
Santral (j)	2
Trafo Merkezi (n)	3
Kritik Bölge (i)	2
İletim hattı (l)	7
Saldırı Tipleri (e)	Santraller için 2, trafo merkezleri için 2, iletim hatları için tek tiptir.

Kaynak	**Koruma Kaynak Miktarı (adet)**
Santral Savunma Tip 1	2
Santral Savunma Tip 2	1
Trafo M. Savunma Tip 1	3
Trafo M. Savunma Tip 2	2
İletim H. Savunma Tip 1	6

Kaynak	**Saldırı Kaynak Miktarı (adet)**
Santral Saldırı Tip 1	1
Santral Saldırı Tip 2	1
Trafo M. Saldırı Tip 1	1
Trafo M. Saldırı Tip 2	1
İletim H. Saldırı Tip 1	1

Yukarıda genel bilgileri verilen problem, Bölüm 5.4'de önerilen yaklaşıma göre çözülmüştür. İlk olarak eldeki koruma kaynakları göz önünde bulundurularak olası tüm koruma kombinasyonları çıkarılmıştır (Seviye 1). Toplam 42 farklı koruma kombinasyonu mevcuttur ve bu kombinasyonlar Çizelge 7.2'de görülmektedir. Çizelge 7.2'de başlık satırı sarı renkle gösterilen sütunlar santrallere, turuncu ile gösterilenler trafo merkezlerine, mavi renkler ise hatlara ait bilgileri göstermektedir. Elde edilen kombinasyonlardan örnek olarak K7 ele alındığında, [(1,1,1,0),(1,1,1,1,1,0),(1,0,1,1,1,1,1)] ,"1" değerleri ilgili bileşenin, ilgili saldırı tipine karşı korunduğu "0" ise korunmadığı anlamına gelmektedir. Buna göre K7 kombinasyonunda ki ilk kısım (1,1,1,0), santral 1'in saldırı tipi 1'e ve 2'ye karşı korunduğu, santral 2'nin ise saldırı 1'e karşı korunduğu, saldırı 2'ye karşı korunmadığı bilgisini taşımaktadır.

Çizelge 7.2. Örnek Problem Koruma Kombinasyonları (Seviye 1)

	P1	P1	P2	P2	T1	T1	T2	T2	T3	T3	H1	H2	H3	H4	H5	H6	H7
K1	1	1	1	0	1	1	1	1	1	0	0	1	1	1	1	1	1
K2	1	0	1	1	1	1	1	1	1	0	0	1	1	1	1	1	1
K3	1	1	1	0	1	1	1	0	1	1	0	1	1	1	1	1	1
K4	1	0	1	1	1	1	1	0	1	1	0	1	1	1	1	1	1
K5	1	1	1	0	1	0	1	1	1	1	0	1	1	1	1	1	1
K6	1	0	1	1	1	0	1	1	1	1	0	1	1	1	1	1	1
K7	1	1	1	0	1	1	1	1	1	0	1	0	1	1	1	1	1
K8	1	0	1	1	1	1	1	1	1	0	1	0	1	1	1	1	1
K9	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	1	1
K10	1	0	1	1	1	1	1	0	1	1	1	0	1	1	1	1	1
K11	1	1	1	0	1	0	1	1	1	1	1	0	1	1	1	1	1
K12	1	0	1	1	1	0	1	1	1	1	1	0	1	1	1	1	1
K13	1	1	1	0	1	1	1	1	1	0	1	1	0	1	1	1	1
K14	1	0	1	1	1	1	1	1	1	0	1	1	0	1	1	1	1
K15	1	1	1	0	1	1	1	0	1	1	1	1	0	1	1	1	1
K16	1	0	1	1	1	1	1	0	1	1	1	1	0	1	1	1	1
K17	1	1	1	0	1	0	1	1	1	1	1	1	0	1	1	1	1
K18	1	0	1	1	1	0	1	1	1	1	1	1	0	1	1	1	1
K19	1	1	1	0	1	1	1	1	1	0	1	1	1	0	1	1	1
K20	1	0	1	1	1	1	1	1	1	0	1	1	1	0	1	1	1
K21	1	1	1	0	1	1	1	0	1	1	1	1	1	0	1	1	1
K22	1	0	1	1	1	1	1	0	1	1	1	1	1	0	1	1	1
K23	1	1	1	0	1	0	1	1	1	1	1	1	1	0	1	1	1
K24	1	0	1	1	1	0	1	1	1	1	1	1	1	0	1	1	1
K25	1	1	1	0	1	1	1	1	1	0	1	1	1	1	0	1	1
K26	1	0	1	1	1	1	1	1	1	0	1	1	1	1	0	1	1
K27	1	1	1	0	1	1	1	0	1	1	1	1	1	1	0	1	1
K28	1	0	1	1	1	1	1	0	1	1	1	1	1	1	0	1	1
K29	1	1	1	0	1	0	1	1	1	1	1	1	1	1	0	1	1
K30	1	0	1	1	1	0	1	1	1	1	1	1	1	1	0	1	1
K31	1	1	1	0	1	1	1	1	1	0	1	1	1	1	1	0	1
K32	1	0	1	1	1	1	1	1	1	0	1	1	1	1	1	0	1
K33	1	1	1	0	1	1	1	0	1	1	1	1	1	1	1	0	1
K34	1	0	1	1	1	1	1	0	1	1	1	1	1	1	1	0	1
K35	1	1	1	0	1	0	1	1	1	1	1	1	1	1	1	0	1
K36	1	0	1	1	1	0	1	1	1	1	1	1	1	1	1	0	1
K37	1	1	1	0	1	1	1	1	1	0	1	1	1	1	1	1	0
K38	1	0	1	1	1	1	1	1	1	0	1	1	1	1	1	1	0
K39	1	1	1	0	1	1	1	0	1	1	1	1	1	1	1	1	0
K40	1	0	1	1	1	1	1	0	1	1	1	1	1	1	1	1	0
K41	1	1	1	0	1	0	1	1	1	1	1	1	1	1	1	1	0
K42	1	0	1	1	1	0	1	1	1	1	1	1	1	1	1	1	0

Seviye 2'de her bir koruma kombinasyonu için yasaklama (saldırı) kombinasyonları belirlenmektedir. Burada örnek problem; anlaşılırlığı arttırmak amacıyla, her bileşen (santral, trafo merkezi ve iletim hattı) için saldırılabilecek tek bir savunmasız nokta bırakılacak şekilde tasarlanmıştır, dolayısıyla her koruma kombinasyonu için alternatif tek bir saldırı planı (kombinasyonu) oluşabilecektir.

Bu kapsamda Çizelge 7.3'de verilen toplam 42 adet saldırı kombinasyonu çıkmıştır. Örnek olarak 7. satırda YK7_1 ile gösterilen 7. yasaklama kombinasyonudur ve ilk dört değer (0,0,0,1), Santral 1'in ve Santral 2'nin sırasıyla Tip1 ve Tip 2 saldırılarıyla yasaklanıp yasaklanmadığını gösterecektir. Buna göre bu kombinasyon Santral 1'in iki tür saldırı ile de yasaklanmadığı, Santral 2'nin ise sadece 2.tür saldırı ile yasaklandığı duruma karşılık gelmektedir. Nitekim Çizelge 7.2'den görüldüğü gibi bu saldırı kombinasyonu (K7) Santral 1'in iki tip saldırıya karşı da korunduğu, Santral 2'nin ise sadece 1. Tip saldırıya karşı korunduğu duruma karşı gelmekteydi, bir başka deyişle bu koruma kombinasyonu için yapılabilecek saldırı kombinasyonu da doğal olarak bu şekilde ortaya çıkmıştır.

Saldırı planlarının amaç fonksiyonu değerleri modelin 3. seviyesinde belirlenmektedir.

Çizelge 7.3. Örnek Problem Yasaklama Kombinasyonları (Seviye 2)

	ST.1	ST.2	ST.1	ST.2	ST.1	ST.2	ST.1	ST.2	ST.1	ST.2	ST.1	ST.1	ST.1	ST.1	ST.1	ST.1	ST.1
	P1	P1	P2	P2	T1	T1	T2	T2	T3	T3	H1	H2	H3	H4	H5	H6	H7
YK1_1	0	0	0	1	0	0	0	0	0	1	1	0	0	0	0	0	0
YK2_1	0	1	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0
YK3_1	0	0	0	1	0	0	0	1	0	0	1	0	0	0	0	0	0
YK4_1	0	1	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0
YK5_1	0	0	0	1	0	1	0	0	0	0	1	0	0	0	0	0	0
YK6_1	0	1	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0
YK7_1	0	0	0	1	0	0	0	0	0	1	0	1	0	0	0	0	0
YK8_1	0	1	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0
YK9_1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	0	0
YK10_1	0	1	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0
YK11_1	0	0	0	1	0	1	0	0	0	0	0	1	0	0	0	0	0
YK12_1	0	1	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0
YK13_1	0	0	0	1	0	0	0	0	0	1	0	0	1	0	0	0	0
YK14_1	0	1	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0
YK15_1	0	0	0	1	0	0	0	1	0	0	0	0	1	0	0	0	0
YK16_1	0	1	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0
YK17_1	0	0	0	1	0	1	0	0	0	0	0	0	1	0	0	0	0
YK18_1	0	1	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0
YK19_1	0	0	0	1	0	0	0	0	0	1	0	0	0	1	0	0	0
YK20_1	0	1	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0
YK21_1	0	0	0	1	0	0	0	1	0	0	0	0	0	1	0	0	0
YK22_1	0	1	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0
YK23_1	0	0	0	1	0	1	0	0	0	0	0	0	0	1	0	0	0
YK24_1	0	1	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0
YK25_1	0	0	0	1	0	0	0	0	0	1	0	0	0	0	1	0	0
YK26_1	0	1	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0
YK27_1	0	0	0	1	0	0	0	1	0	0	0	0	0	0	1	0	0
YK28_1	0	1	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0
YK29_1	0	0	0	1	0	1	0	0	0	0	0	0	0	0	1	0	0
YK30_1	0	1	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0
YK31_1	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	1	0
YK32_1	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0
YK33_1	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	1	0
YK34_1	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0
YK35_1	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	1	0
YK36_1	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0
YK37_1	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	1
YK38_1	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1
YK39-1	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	1
YK40_1	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1
YK41_1	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	1
YK42_1	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1

Çizelge 7.4'de verilen kombinasyonların her biri 3. seviyeye (sistem operatörü seviyesi) gönderilerek amaç fonksiyonu değerleri belirlenmiştir. Hatırlanacak olursa bir koruma kombinasyonuna karşılık birden fazla yasaklama kombinasyonu oluştuğunda bu yasaklama kombinasyonlarına karşılık gelen en kötü değer, o koruma kombinasyonunun değeri olacaktı. Bu örnek problemde önceki bölümde belirtildiği gibi her bir koruma kombinasyonu için tek bir yasaklama kombinasyonu değeri olduğundan, koruma kombinasyonu değeri de söz konusu yasaklama kombinasyonunun değeri olarak oluşacaktır. Ancak bu örneğin aksine, eğer bir koruma kombinasyonu için birden fazla yasaklama kombinasyonu olmuş olsaydı, ilgili koruma kombinasyonuna ait yasaklama kombinasyonlarından elde edilen en büyük (en kötü) değer, yukarıda belirtildiği gibi o koruma kombinasyonunun amaç fonksiyonu değeri olacaktır.

Çizelge 7.4. Örnek Problem Amaç Fonksiyonu Değerleri

	Yasaklama Kombinasyonu	Amaç Fonksiyonu Değeri
1	ll2 : ['0 0 0 1', '0 0 0 0 0 1', '0 0 0 0 0 0 1']	7
2	ll2 : ['0 0 0 1', '0 0 0 0 0 1', '0 0 0 0 0 1 0']	5
3	ll2 : ['0 0 0 1', '0 0 0 0 0 1', '0 0 0 0 1 0 0']	5
4	ll2 : ['0 0 0 1', '0 0 0 0 0 1', '0 0 0 1 0 0 0']	5
5	ll2 : ['0 0 0 1', '0 0 0 0 0 1', '0 0 1 0 0 0 0']	5
6	ll2 : ['0 0 0 1', '0 0 0 0 0 1', '0 1 0 0 0 0 0']	Çözüm Yok
7	ll2 : ['0 0 0 1', '0 0 0 0 0 1', '1 0 0 0 0 0 0']	12
8	ll2 : ['0 0 0 1', '0 0 0 0 1 0', '0 0 0 0 0 0 1']	7
9	ll2 : ['0 0 0 1', '0 0 0 0 1 0', '0 0 0 0 0 1 0']	5
10	ll2 : ['0 0 0 1', '0 0 0 0 1 0', '0 0 0 0 1 0 0']	5
11	ll2 : ['0 0 0 1', '0 0 0 0 1 0', '0 0 0 1 0 0 0']	5
12	ll2 : ['0 0 0 1', '0 0 0 0 1 0', '0 0 1 0 0 0 0']	5
13	ll2 : ['0 0 0 1', '0 0 0 0 1 0', '0 1 0 0 0 0 0']	Çözüm Yok
14	ll2 : ['0 0 0 1', '0 0 0 0 1 0', '1 0 0 0 0 0 0']	12
15	ll2 : ['0 0 0 1', '0 0 0 1 0 0', '0 0 0 0 0 0 1']	7
16	ll2 : ['0 0 0 1', '0 0 0 1 0 0', '0 0 0 0 0 1 0']	5
17	ll2 : ['0 0 0 1', '0 0 0 1 0 0', '0 0 0 0 1 0 0']	5
18	ll2 : ['0 0 0 1', '0 0 0 1 0 0', '0 0 0 1 0 0 0']	5
19	ll2 : ['0 0 0 1', '0 0 0 1 0 0', '0 0 1 0 0 0 0']	5
20	ll2 : ['0 0 0 1', '0 0 0 1 0 0', '0 1 0 0 0 0 0']	Çözüm Yok
21	ll2 : ['0 0 0 1', '0 0 0 1 0 0', '1 0 0 0 0 0 0']	12
22	ll2 : ['0 0 1 0', '0 0 0 0 0 1', '0 0 0 0 0 0 1']	7
23	ll2 : ['0 0 1 0', '0 0 0 0 0 1', '0 0 0 0 0 1 0']	5
24	ll2 : ['0 0 1 0', '0 0 0 0 0 1', '0 0 0 0 1 0 0']	5
25	ll2 : ['0 0 1 0', '0 0 0 0 0 1', '0 0 0 1 0 0 0']	5
26	ll2 : ['0 0 1 0', '0 0 0 0 0 1', '0 0 1 0 0 0 0']	5
27	ll2 : ['0 0 1 0', '0 0 0 0 0 1', '0 1 0 0 0 0 0']	Çözüm Yok
28	ll2 : ['0 0 1 0', '0 0 0 0 0 1', '1 0 0 0 0 0 0']	12
29	ll2 : ['0 0 1 0', '0 0 0 0 1 0', '0 0 0 0 0 0 1']	7
30	ll2 : ['0 0 1 0', '0 0 0 0 1 0', '0 0 0 0 0 1 0']	5

Çizelge 7.5. Örnek Problem Amaç Fonksiyonu Değerleri (devam)

31	ll2 : ['0 0 1 0', '0 0 0 0 1 0', '0 0 0 0 1 0 0']	5
32	ll2 : ['0 0 1 0', '0 0 0 0 1 0', '0 0 0 1 0 0 0']	5
33	ll2 : ['0 0 1 0', '0 0 0 0 1 0', '0 0 1 0 0 0 0']	5
34	ll2 : ['0 0 1 0', '0 0 0 0 1 0', '0 1 0 0 0 0 0']	Çözüm Yok
35	ll2 : ['0 0 1 0', '0 0 0 0 1 0', '1 0 0 0 0 0 0']	12
36	ll2 : ['0 0 1 0', '0 0 0 1 0 0', '0 0 0 0 0 0 1']	7
37	ll2 : ['0 0 1 0', '0 0 0 1 0 0', '0 0 0 0 0 1 0']	5
38	ll2 : ['0 0 1 0', '0 0 0 1 0 0', '0 0 0 0 1 0 0']	5
39	ll2 : ['0 0 1 0', '0 0 0 1 0 0', '0 0 0 1 0 0 0']	5
40	ll2 : ['0 0 1 0', '0 0 0 1 0 0', '0 0 1 0 0 0 0']	5
41	ll2 : ['0 0 1 0', '0 0 0 1 0 0', '0 1 0 0 0 0 0']	Çözüm Yok
42	ll2 : ['0 0 1 0', '0 0 0 1 0 0', '1 0 0 0 0 0 0']	12

Şekil 6.7'de verilen kromozom yapısı hatırlanarak Çizelge 7.4'de verilen program sonuçlarından 2. sırada yer alan ve amaç fonksiyonu değeri 5 olan "ll2 : ['0 0 0 1', '0 0 0 0 0 1 0 0 0 0', '0 0 0 0 0 1 0']" saldırı planını incelersek; 2 no.'lu santralin 2. tip, 3 no.'lu Trafo merkezinin 2. tip ve 6 no.'lu iletim hattının ise 1. tip saldırı ile yasaklandığı görülmektedir. Bu saldırı planına karşılık gelen koruma kombinasyonu ise Çizelge 7.2'de K31'dir. Aşağıda Çizelge 7.5'de tekrar verilen K31 koruma kombinasyonuna, [(1,1,1,0),(1,1,1,1,1,0),(1,1,1,1,1,0,1)] göre; 1 no.'lu santral saldırı tipi 1 ve 2'ye karşı, 2 no.'lu santral ise sadece saldırı tipi 1'e karşı korunmaktadır. Trafo merkezi 1 ve 2, saldırı tipi 1 ve 2'ye karşı korunmaktadır. 3 no.'lu trafo merkezi ise saldırı tipi 2'e karşı korunmaktadır. İletim hatlarından ise 6 numaralı iletim hattı hariç tüm hatların saldırı tipi 1'e karşı korunduğu görülmektedir.

Çizelge 7.6. K31 Koruma Kombinasyonu

	ST.1	ST.2	ST.1	ST.2	ST.1	ST.2	ST.1	ST.2	ST.1	ST.2	ST.1	ST.1	ST.1	ST.1	ST.1	ST.1	ST.1
	P1	P1	P2	P2	T1	T1	T2	T2	T3	T3	H1	H2	H3	H4	H5	H6	H7
K31	1	1	1	0	1	1	1	1	1	0	1	1	1	1	1	0	1

Çizelge 7.6'da aynı problem için koruma ve saldırı kaynaklarında yapılan değişikler ile oluşturulmuş 8 farklı örneğe ait amaç fonksiyonu değerleri ve çözüm süreleri analiz edilmiştir. Bu örneklerde koruma ve saldırı kaynaklarında yapılan değişiklikler koruma kombinasyonu sayısı ve saldırı kombinasyonu sayısını değiştirmekte ve aynı oranda çözüm süreleri de bu kombinasyon sayılarına bağlı olarak değişmektedir. Çizelge 7.6'da ilk satırda tüm örneklerde yer alan bileşenlerin sayıları ve kaç tip saldırı ile tehdit edildikleri verilmiştir.

Tüm örneklerde santral sayısı 2, trafo merkezi sayısı 3, santral ve trafo merkezlerine yapılabilecek saldırı tip sayısı ise 2 olarak verilmiştir. İletim hattı sayısı ise tüm örneklerde 7, 1-5 örnek için saldırı tip sayısı 1, 6-8 örnek için saldırı tip sayısı 2 olarak verilmiştir.

Çizelge 7.7. Örnek Problemler için Matematiksel Model Çözüm Süreleri

Tüm Örneklerde, (Santral Sayısı, Santral Saldırı Tip Sayısı)= (2,2), (Trafo Merkezi Sayısı, Trafo Merkezi Saldırı Tip Sayısı) = (3,2)
1-5 örnek, (İletim Hattı Sayısı, İletim Hattı Saldırı Tip Sayısı) = (7,1), 6-8 örnek, (İletim Hattı Sayısı, İletim Hattı Saldırı Tip Sayısı) = (7,2)

Örnek No	Santral, Trafo Merkezi ve İletim Hattı Bileşenleri için Koruma ve Saldırı Kaynak Sayıları			Örnekler için Koruma ve Saldırı Kombinasyonları		Örnekler için Çözüm Süreleri ve Amaç Fonksiyonu Değerleri	
	Santral	Trafo M.	İletim H.	Koruma	Saldırı	Süre	*H(z)*
1	(2,1)*	(2,2)	(5)	378	756	7' 07''	5.0
	(1,1)**	(1,1)	(1)				
2	(1,1)	(2,1)	(4)	1260	7560	1 saat 10' 50''	5.0
	(1,1)	(1,1)	(1)				
3	(2,1)	(3,2)	(5)	126	252	2' 36''	5.0
	(1,1)	(1,1)	(1)				
4	(1,1)	(1,1)	(2)	756	15120	2 saat 45' 49''	7.0
	(0,1)	(1,1)	(1)				
5	(1,1)	(1,1)	(3)	1260	30240	5 saat 53' 43''	5.0
	(0,1)	(1,1)	(2)				
6	(1,1)	(1,1)	(5,5)	15876	127008	25 saat 19' 10''	5.0
	(0,1)	(1,1)	(2,1)				
7	(1,1)	(1,1)	(3,4)	44100	1058400	7 günde tamamlanmadı	-
	(0,1)	(1,1)	(2,3)				
8	(1,1)	(1,1)	(3,3)	44100	6350400	7 günde tamamlanmadı	-
	(0,1)	(1,1)	(2,2)				

(2,1)* Santraller için Tip 1 saldırıya karşı 2, tip 2 saldırıya karşı 1 koruma kaynağı vardır.
(1,1)** Santraller için Tip 1 saldırı için 1, tip 2 saldırıiçin 1 saldırı kaynağı vardır.

Çizelge 7.6'dan örnek 4'ü ele alırsak, santral için koruma kapasitesi (1,1), saldırı kapasitesi (0,1) verilmiştir. Buna göre 2 santralden biri, Tip 1 saldırısına, yine 2 santralden biri, Tip 2 saldırısına karşı korunabilmektedir. Saldırı kaynakları ise tip 1 için 0, tip iki için 1 adettir, Tip 2 saldırısı ile saldırgan, korunmayan santrallerden birini yasaklayabilir. Benzer şekilde trafo merkezleri için (1,1) koruma kaynağı ve (1,1) saldırı kaynağı verilmiştir. İletim hatları için (2) koruma kaynağı ve (1) saldırı kaynağı verilmiştir (bu örnekte tip sayısı birdir). 7 iletim hattının 2'si korunabilir, korunmayan hatlardan 1 ise yasaklanabilir. Verilen koruma ve saldırı kaynaklarına göre koruma kombinasyonu 756 adet ve saldırı kombinasyonu 15.120 adettir. Bu örnek 2 saat 45 dakika 49 saniyede çözülmüştür ve amaç fonksiyonu değeri 7.0'dir.

Koruma kapasiteleri ve saldırı kapasiteleri azaldıkça koruma ve saldırı kombinasyonlarının sayısının ve dolayısıyla çözüm süresinin arttığı görülmektedir. Ayrıca son iki örnekte iletim hattına yapılabilecek saldırı tipinin birden ikiye çıkması ile 7 gün geçmesine rağmen çözüm elde edilememiştir. Bu durum problemin bir sezgisel algoritmaya olan ihtiyacı ortaya koymaktadır. Python'da kodlanmış çözüm algoritmasının örnek bir problem için sonuçları Ek-D'da gösterilmiştir.

Çözüm sürelerini etkileyen faktörlere baktığımızda, öncelikle seviye 1'de oluşan koruma kombinasyonu ile seviye 2 de oluşan saldırı kombinasyonu sayılarının doğrudan çözüm süresini etkilediği görülmektedir. Kombinasyon sayılarını ise bileşen sayıları ve tip sayıları etkilemektedir. Bölüm 5.5'te karmaşıklık analizi ile çözüm uzayının bileşen ve tip sayılarından nasıl etkilendiği detaylı olarak açıklanmıştı. Özelikle tip sayısındaki artış çözüm uzayının üssel olarak büyümesine sebep olmaktadır. Ayrıca seviye 3'ün çözüm süresi de problemin parametrelerindeki değişimden doğrudan etkilenmektedir. Santral, trafo merkezi, iletim hattı ve tip sayılarındaki artış her bir saldırı kombinasyonunun çözümü için gereken süreyi arttırmaktadır. Bu nedenle önerilen sezgisel algoritma ile seviye 1 için kural tabanlı bir sezgisel algoritma, seviye 2 için ise seviye 3'e gönderilen saldırı kombinasyon sayısını azaltmaya yönelik bazı kontrol işlemleri uygulanmıştır. Bu işlemler ile seviye 3'te işlem gören saldırı kombinasyonu sayısını azaltarak, çözüm süresinin azaltılması amaçlanmaktadır.

7.2. Önerilen Genetik Algoritma Tabanlı Sezgisel Yaklaşımların Deneysel Değerlendirmesi

Üç Seviyeli Çoklu Saldırı Tipli Elektrik Şebekesi Koruma Probleminin Matematiksel Model ile Çözümü için önceki bölümde örnek bir ana problem türetilmiş ve bu problemin koruma kapasiteleri, saldırı kapasiteleri ve saldırı tip sayılarında yapılan değişikler ile 8 farklı test örneği üretilmiştir. Bu 8 test örneğinin matematiksel model ile çözüm sonuçları değerlendirilmiştir. Bu örneklerden 7 ve 8'de makul bir çözüm elde edilememiştir.

Bu bölümde ise önerilen genetik algoritma tabanlı sezgisel algoritmanın performansı değerlendirilmiştir. GA tabanlı r_GA sezgisel algoritmasının pseudo kodu aşağıdadır:

Algoritma 7.1: r_GA Sözde (Pseudo) Kod

BP: Başlangıç Popülasyonu, UD: Uygunluk Değeri, YP: Yeni Popülasyon, BS: Popülasyon Birey Sayısı, ÇP: Çaprazlama Oranı, EB: En İyi Birey Oranı

```
BP'nu oluştur
        Koruma kombinasyonlarını türet
        Koruma kombinasyonları içinden rassal olarak BS adet seç
        UD'leri hesapla

Yeni birey türet (BS adet)
        BP içinden rastgele bireyleri seç
                Çaprazlama işlemi
                Yerel Arama algoritması
                Mutasyon işlemi
Tekrarla (6-8)

5'de türetilen yeni bireylerin UD'lerini hesapla
        Koruma kombinasyonları karşılık gelen yasaklama kombinasyonlarını türet
        Yasaklama kombinasyonlarını hafızadaki kombinasyonlar ile karşılaştır,
                Eğer hafızada kayıtlı ise o değeri döndür,
                Kayıtlı değilse UD'ni hesapla
        Yeni yasaklama kombinasyonu ve değerini hafızaya kaydet
```

17: YP'yi oluştur.

18: BP ve yeni bireyleri UD'ne göre büyükten küçüğe sırala.

19: Eğer 18'deki sıralamada bireylerin UD'leri eşitse yeni bireyi üst sıraya yerleştir

20: Sıralamadan ilk EBxBS bireyi seç YP'ye ekle

21: Mutasyon ile 5'de üretilen bireyleri seç YP'ye ekle

22: Çaprazlama ve Yerel Arama algoritması ile 5'de üretilen yeni bireylerden rasgele ÇPxBS birey seç YP'ye ekle

23: YP'yi BP yerine ata (İterasyon sayısı kadar tekrarla)

24: Tüm popülasyonlar içinden eniyi amaç fonksiyonu değerine sahip koruma kombinasyonunu seç

Önerilen r_GA sezgisel algoritması kullanılarak Bölüm 7.1'de verilen problemler, matematiksel model ile çözülemeyenler de dahil olmak üzere çözülebilmiştir algoritması (bu karşılaştırmada örnek problemin gerçek bir problem olmaması nedeni ile r_GA'nın popülasyon büyüklüğü, çaprazlama oranı vd. parametreler tahmini olarak belirlenmiştir, bir analiz ile tespit edilmemiştir). Çizelge 7.7'de matematiksel model ve önerilen GA tabanlı sezgisel yaklaşım çözümleri karşılaştırılmıştır.

Çizelge 7.8. Matematiksel Model Sonuçları ile GA Tabanlı Sezgisel Yaklaşım Sonuçlarının Karşılaştırılması

Tüm Örneklerde, (Santral Sayısı, Santral Saldırı Tip Sayısı)= (2,2), (Trafo Merkezi Sayısı, Trafo Merkezi Saldırı Tip Sayısı) = (3,2)
1-5 örnek, (İletim Hattı Sayısı, İletim Hattı Saldırı Tip Sayısı) = (7,1), 6-8 örnek, (İletim Hattı Sayısı, İletim Hattı Saldırı Tip Sayısı) = (7,2)

Örnek No	Örnekler için Koruma ve Saldırı Kombinasyonları		Örnekler için Çözüm Süreleri ve Amaç Fonksiyonu Değerleri		Örnekler için r_GA Sezgiseli Çözüm süreleri, Başlangıç Populasyonu (BP) Süresi, En İyi değer ve İterasyon Sayıları			
	Koruma	Saldırı	Süre	*H(z)*	Çözüm Süresi	BP Süresi	H(z)	İterasyon
1	378	756	7' 07''	5.0	3' 42''	1' 50''	5.0	3
2	1260	7560	1 saat 10' 50''	5.0	22' 32''	5' 49''	5.0	5
3	126	252	2' 36''	5.0	2'23	2' 02''	5.0	3
4	756	15120	2 saat 45' 49''	7.0	37' 42''	11' 50''	7.0	3
5	1260	30240	5 saat 53' 43''	5.0	51' 15''	17' 10''	5.0	3
6	15876	127008	25 saat 19' 10''	5.0	23' 25''	7' 05''	5.0	3
7	44100	1058400	7 günde tamamlanmadı	-	1 saat 8' 50''	18' 10''	5.0	3
8	44100	6350400	7 günde tamamlanmadı	-	7 saat 35' 40''	1 saat 23' 35''	5.0	3

(2,1)* Santraller için Tip 1 saldırıya karşı 2, tip 2 saldırıya karşı 1 koruma kaynağı vardır.
(1,1)** Santraller için Tip 1 saldırı için 1, tip 2 saldırıiçin 1 saldırı kaynağı vardır.

Çizelge 7.7'de görüldüğü gibi önerilen genetik algoritma tabanlı sezgisel yaklaşım eniyi amaç fonksiyonu değerine tüm örneklerde ulaşmaktadır. Çözüm sonuçları incelendiğinde 3. iterasyonda tüm en iyi değerlerin yeni nesilde biriktiği görülmektedir. Örnek bir problem için elde edilen popülasyonlar ve uygunluk değerleri Ek-E'de paylaşılmıştır. Çözüm süreleri açısından karşılaştırıldığında ise küçük boyutlu problemlerin çözüm süreleri her iki yaklaşımda da birbirine yakın olarak elde edilmektedir. Küçük problemlerde, sezgisel yaklaşım ile üretilen başlangıç popülasyonunda ele alınan koruma kombinasyonlarının, tüm kombinasyonların önemli bir kısmını oluşturması buna neden olmaktadır. Diğer örneklerde ise sezgisel yaklaşım ile çözüm sürelerinin önemli ölçüde kısaldığı görülmektedir. Matematiksel model çözümü ile makul sürelerde çözüm alamadığımız örnek 7 ve 8 için önerilen yaklaşım ile çözüm bulunmuştur. Önerilen GA tabanlı sezgisel algoritmanın örnek problem için iyi bir performans gösterdiği görülmektedir.

Bölüm 6.2'de önerilen GA tabanlı sezgisel algoritmada, başlangıç popülasyonunun oluşturulmasında iki farklı yöntem kullanıldığı açıklanmıştır. Bunlardan biri başlangıç popülasyonunun tüm koruma kombinasyonları arasından rassal olarak seçildiği r_GA sezgisel algoritmadır. Diğer algoritma ise kural tabanlı bakış açısı ile önerilen k_GA sezgisel algoritmasıdır. Bu algoritmada başlangıç popülasyonu oluşturulurken bazı kurallara uyan kromozomların başlangıç popülasyonunda yer alması sağlanmaktadır. Bu durumun amaç fonksiyonu değerine daha kısa sürede ulaşmaya katkı sağlayacağı düşünülmektedir. Bu amaçla, bu iki algoritmayı kıyaslamak ve hangi durumlarda hangi algoritmanın kullanılmasının uygun olacağını belirlemek amacıyla test problemleri ile deneysel çalışmalar yapılmış ve izleyen kısımda sunulmuştur. k_GA'nın çözüm algoritmasının pseudo kodu aşağıdadır:

Algoritma 7.2: k_GA Sözde (Pseudo) Kod

BP: Başlangıç Popülasyonu, UD: Uygunluk Değeri, YP: Yeni Popülasyon, BS: Popülasyon Birey Sayısı, ÇP: Çaprazlama Oranı, EB: En İyi Birey Oranı, KTB: Kural Tabanlı Birey Oranı

```
1:  BP'nu oluştur
2:      Koruma kombinasyonlarını türet
3:      Koruma kombinasyonlarını tanımlı kurallara göre değerlendir
```

4: En yüksek kural değerine sahip koruma kombinasyonları içinden rassal olarak KTBxBS birey seç.
5: Koruma kombinasyonları içinden rassal olarak (1-KTB)xBS birey seç
6: UD'leri hesapla

7: Yeni birey türet (BS adet)
8: BP içinden rastgele bireyleri seç
9: Çaprazlama işlemi
10: Yerel Arama algoritması
11: Mutasyon işlemi
12: Tekrarla (8-10)

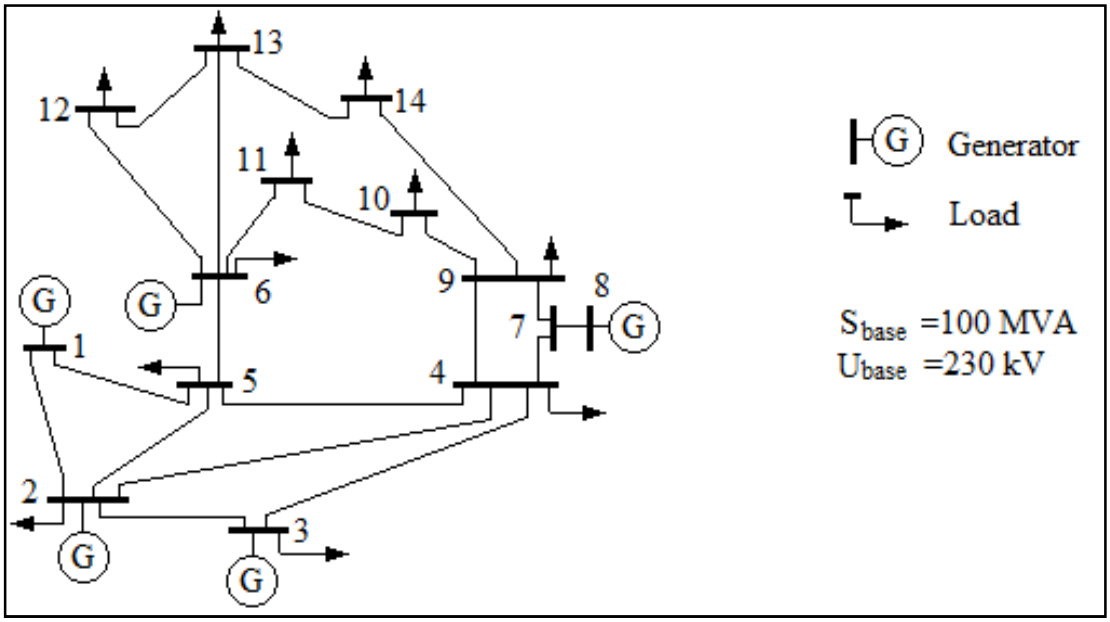
13: 7'de türetilen yeni bireylerin UD'lerini hesapla
14: Koruma kombinasyonları karşılık gelen yasaklama kombinasyonlarını türet
15: Yasaklama kombinasyonlarını hafızadaki kombinasyonlar ile karşılaştır,
16: Eğer hafızada kayıtlı ise o değeri döndür,
17: Kayıtlı değilse UD'ni hesapla
18: Yeni yasaklama kombinasyonu ve değerini hafızaya kaydet

19: YP'yi oluştur.
20: BP ve yeni bireyleri UD'ne göre büyükten küçüğe sırala.
21: Eğer 20'deki sıralamada bireylerin UD'leri eşitse yeni bireyi üst sıraya yerleştir
22: Sıralamadan ilk EBxBS bireyi seç YP'ye ekle
23: Mutasyon ile 7'de üretilen bireyleri seç YP'ye ekle
24: Çaprazlama ve Yerel Arama algoritması ile 7'de üretilen yeni bireylerden rasgele ÇPxBS birey seç YP'ye ekle

25: YP'yi BP yerine ata (İterasyon sayısı kadar tekrarla)
26: Tüm popülasyonlar içinden eniyi amaç fonksiyonu değerine sahip koruma kombinasyonunu seç

r_GA sezgisel algoritması ile k_GA sezgisel algoritması için, literatürde sıklıkla kullanılan IEEE (*The Institute of Electrical and Electronics Engineers*)'nin 14 baralı test problemi verileri için elde edilen çözüm sonuçları performans kıyaslaması ile sunulmuştur. IEEE'nin 14 baralı örnek problem seti literatürde yer alan ve elektrik şebeke sistemleri ile ilgili bilimsel çalışmalarda test verisi olarak yaygın kullanılan bir settir. 1962 Şubat ayı itibariyle Amerikan Elektrik Güç Sisteminin (Orta Batı ABD'de) bir bölümünü temsil etmektedir (University of Washington - Electrical Engineering, 1962). Bu test verileri, gerçekçi bir sistemi ele alarak önerilen algoritmaların performansını görmek açısından önemlidir. Ancak diğer çalışmalarda bu çalışmada olduğu gibi saldırı türleri dikkate alınmadığı için sonuçlar karşılaştırılamamıştır.

Çalışmamızdaki saldırı türünü tek tip varsayarak var olan çalışma sonuçları ile karşılaştırma istendiğinde ise elektrik şebekeleri ile ilgili ve bu test verisini kullanan diğer çalışmaların, farklı amaçları olduğu (kritik bölgelerin taleplerini olabildiğince karşılamak vb.) ve farklı karakteristikler taşıdığı görülmektedir. IEEE 14 Bara test sistemine ait parametreler Çizelge 7.8'de verilmektedir ve Şekil 7.2'de bu test sistemine ait tek hat diyagramı sunulmaktadır.



Şekil 7.2. IEEE 14 Bara Sistemine Ait Tek Hat Diyagramı

Elektrik enerji sisteminde elektriksel niceliklerin büyüklükleri çok farklı seviyelerde olabilir. Örneğin bir üretim biriminin aktif güç üretim değeri megawatt (10^6) seviyelerinde iken bara gerilimleri kilovolt (10^3), hat empedansı miliohm (10^{-3}) seviyelerinde olabilir. Değerlerdeki bu seviye farkları, matematiksel işlemlerin uygulanmasını ve yorumlanmasını zorlaştırmaktadır. Bu nedenle elektrik enerji sistemlerinin analizinde gerçek değerler (*actual values*) yerine *per unit (pu)* değerler kullanılmaktadır. Her hangi bir parametrenin *per unit (pu)* değeri şöyle hesaplanabilir.

$$\text{Per Unit Değer} = \frac{\text{Gerçek Değer}}{\text{Taban Değeri}} \qquad (7.1)$$

Hesaplamalar sonucu elde edilen *pu* değerler, kullanılan taban değerleri (*base values*) ile çarpılarak tekrar gerçek değerlere (*actual values*) dönüştürülebilir. Taban değerlerinin seçimi sistemi analiz edecek kişiye bağlıdır. Güç ve gerilim için taban değerleri seçilmesi yeterli olup diğer tüm elektriksel niceliklere (akım, impedans, admittans, vb.) ait taban değerleri, güç ve gerilim için seçilmiş taban değerleri ile hesaplanabilir.

Bu örnekte de elektrik enerji sistemlerinin analizinde *pu* değerler kullanıldığından, Çizelge 7.8'de parametreler de *pu* cinsinden verilmektedir.

Çizelge 7.9. IEEE 14 bara test sistemi (a) Hat parametreleri (b) Trafo Merkezi (Bara) Gerilim ve Faz Açısı Limitleri (c) Aktif Güç Üretim Sınırları ve Aktif Güç Talepleri

Hat No(l)	*Trafo merkezinden (Baradan) (n)*	*Trafo merkezine (Baraya) (b)*	*Direnç (Resistans)* rs_{nb} *(pu)*	*Reactans* rc_{nb} *(pu)*	S_{bn}^{max} *(pu)*
1	1	2	0,01938	0,05917	1,20
2	1	5	0,05403	0,22304	0,65
3	2	3	0,04699	0,19797	0,36
4	2	4	0,05811	0,17632	0,65
5	2	5	0,05695	0,17388	0,50
6	3	4	0,06701	0,17103	0,65
7	4	5	0,01335	0,04211	0,45
8	4	7	0,00	0,20912	0,55
9	4	9	0,00	0,55618	0,32
10	5	6	0,00	0,25202	0,45
11	6	11	0,09498	0,19890	0,18
12	6	12	0,12291	0,25581	0,32
13	6	13	0,06615	0,13027	0,32
14	7	8	0,00	0,17615	0,32
15	7	9	0,00	0,11001	0,32
16	9	10	0,03181	0,08450	0,32
17	9	14	0,12711	0,27038	0,32
18	10	11	0,08205	0,19207	0,12
19	12	13	0,22092	0,19988	0,12
20	13	14	0,17093	0,34802	0,12

Çizelge 7.10. IEEE 14 bara test sistemi (a) Hat parametreleri (b) Trafo Merkezi (Bara) Gerilim ve Faz Açısı Limitleri (c) Aktif Güç Üretim Sınırları ve Aktif Güç Talepleri (devam)

Trafo Merkezi (Bara) No (*n*)	V_n^{min} (*pu*)	V_n^{max} (*pu*)	θ_n^{min} (*rad*)	θ_n^{max} (*rad*)
1	1,00	1,00	0,00	0,00
2	0,95	1,05	-1,57	1,57
3	0,95	1,05	-1,57	1,57
4	0,95	1,05	-1,57	1,57
5	0,95	1,05	-1,57	1,57
6	0,95	1,05	-1,57	1,57
7	0,95	1,05	-1,57	1,57
8	0,95	1,05	-1,57	1,57
9	0,95	1,05	-1,57	1,57
10	0,95	1,05	-1,57	1,57
11	0,95	1,05	-1,57	1,57
12	0,95	1,05	-1,57	1,57
13	0,95	1,05	-1,57	1,57
14	0,95	1,05	-1,57	1,57

Bara No (*n*)	PG_j^{min} *(pu)*	PG_j^{max} *(pu)*	P_n *(pu)*
1	0	3,35	0
2	0	1,40	0,217
3	0	1,00	0,942
4	0	0	0,478
5	0	0	0,076
6	0	1,00	0,112
7	0	0	0
8	0	1,00	0
9	0	0	0,295
10	0	0	0,09
11	0	0	0,035
12	0	0	0,061
13	0	0	0,135
14	0	0	0,149

r_GA sezgisel algoritması ile k_GA sezgisel algoritmasının karşılaştırılması amacı ile yukarıda tek hat akış diyagramı ve parametreleri verilen IEEE 14 baralı örnek veri setinden yararlanılarak, koruma ve saldırı kaynaklarında değişiklikler yapılarak 3 kategoride, 12 örnek problem seti oluşturulmuştur. Bu kategoriler, koruma kaynağı sayısına göre sıkı koruma, orta koruma ve gevşek koruma olmak üzere 3 kategoriye ayrılmıştır. Sıkı koruma, bileşenlerin büyük çoğunluğunun korunabileceği kaynağa sahip olunan durumları temsil etmektedir. Gevşek koruma kategori ise bileşenlerin çoğunu korumak için yeterli koruma kaynağının olmadığı vakaları temsil etmektedir. Orta koruma ise sıkı koruma ile gevşek koruma arasında yer almaktadır.

Her bir kategori için 4 örnek problem oluşturulmuştur. Bunlardan bir kısmı saldırganın yasaklama kaynaklarının az olduğu, diğerleri ise daha fazla yasaklama kaynağına sahip olduğu örnek problemdir. Oluşturulan problemlere ait koruma kaynakları, saldırı kaynakları ve bunlara bağlı oluşan olası koruma ve yasaklama kombinasyonu sayıları Çizelge 7.9'da görülmektedir. Yasaklama kaynaklarının az olması yasaklama kombinasyonu sayısının dolayısıyla problemin çözüm süresinin artmasına sebep olmaktadır.

Çizelge 7.11. IEEE 14 Baralı Test Probleminden Türetilen Örnek Problem Setleri

	SET 1		Kaynak Sayıları		**SET 2**		Kaynak Sayıları	
			Koruma	Saldırı			Koruma	Saldırı
Sıkı Koruma	S	Tip 1 Saldırı	5	1	S	Tip 1 Saldırı	4	1
		Tip 2 Saldırı	4	1		Tip 2 Saldırı	4	1
	TM	Tip 1 Saldırı	13	1	TM	Tip 1 Saldırı	12	1
		Tip 2 Saldırı	13	1		Tip 2 Saldırı	12	2
	H	Tip 1 Saldırı	19	1	H	Tip 1 Saldırı	17	1
	Sayısı			19.600	Sayısı			236.008.500
	Saldırı Kombinasyonu Sayısı			19.600	Saldırı Kombinasyonu Sayısı			1.416.051.000
	SET 3		Kaynak Sayıları		**SET 4**		Kaynak Sayıları	
			Koruma	Saldırı			Koruma	Saldırı
	S	Tip 1 Saldırı	4	1	S	Tip 1 Saldırı	4	1
		Tip 2 Saldırı	4	1		Tip 2 Saldırı	4	1
	TM	Tip 1 Saldırı	13	1	TM	Tip 1 Saldırı	12	2
		Tip 2 Saldırı	12	1		Tip 2 Saldırı	12	2
	H	Tip 1 Saldırı	19	1	H	Tip 1 Saldırı	18	1
	Sayısı			637.000	Sayısı			39.334.750
	Saldırı Kombinasyonu Sayısı			1.274.000	Saldırı Kombinasyonu Sayısı			78.669.500

Çizelge 7.9. IEEE 14 Baralı Test Probleminden Türetilen Örnek Problem Setleri (devam)

Orta Koruma	**SET 5**		Kaynak Sayıları		**SET 6**		Kaynak Sayıları	
			Koruma	Saldırı			Koruma	Saldırı
	S	Tip 1 Saldırı	4	1	S	Tip 1 Saldırı	4	1
		Tip 2 Saldırı	3	1		Tip 2 Saldırı	4	1
	TM	Tip 1 Saldırı	11	1	TM	Tip 1 Saldırı	11	2
		Tip 2 Saldırı	12	1		Tip 2 Saldırı	10	2
	H	Tip 1 Saldırı	15	1	H	Tip 1 Saldırı	16	2
	Sayısı			25.677.724.800	Sayısı			44.133.589.500
	Saldırı Kombinasyonu Sayısı			1.027.108.992.000	Saldırı Kombinasyonu Sayısı			4.766.427.666.000
	SET 7		Kaynak Sayıları		**SET 8**		Kaynak Sayıları	
			Koruma	Saldırı			Koruma	Saldırı
	S	Tip 1 Saldırı	4	1	S	Tip 1 Saldırı	4	1
		Tip 2 Saldırı	3	1		Tip 2 Saldırı	4	1
	TM	Tip 1 Saldırı	11	3	TM	Tip 1 Saldırı	11	2
		Tip 2 Saldırı	12	2		Tip 2 Saldırı	10	3
	H	Tip 1 Saldırı	15	4	H	Tip 1 Saldırı	16	3
	Sayısı			1.604.857.800	Sayısı			44.133.589.500
	Saldırı Kombinasyonu Sayısı			16.048.578.000	Saldırı Kombinasyonu Sayısı			1.059.206.148.000

Gevşek Koruma	**SET 9**		Kaynak Sayıları		**SET 10**		Kaynak Sayıları	
			Koruma	Saldırı			Koruma	Saldırı
	S	Santral Tip 1	3	2	S	Santral Tip 1	3	2
		Santral Tip 2	3	2		Santral Tip 2	2	3
	TM	TM Tip 1	10	4	TM	TM Tip 1	9	5
		TM Tip 2	9	4		TM Tip 2	9	4
	H	Hat Tip 1	14	6	H	Hat Tip 1	13	6
	Sayısı			7.767.511.752.000	Sayısı			31.070.047.008.000
	Saldırı Kombinasyonu Sayısı			38.837.558.760.000	Saldırı Kombinasyonu Sayısı			1.087.451.645.280.000
	SET 11		Kaynak Sayıları		**SET 12**		Kaynak Sayıları	
			Koruma	Saldırı			Koruma	Saldırı
	S	Santral Tip 1	3	2	S	Santral Tip 1	2	2
		Santral Tip 2	2	2		Santral Tip 2	2	3
	TM	TM Tip 1	8	6	TM	TM Tip 1	8	6
		TM Tip 2	9	5		TM Tip 2	8	6
	H	Hat Tip 1	12	8	H	Hat Tip 1	12	7
	Sayısı			75.733.239.582.000	Sayısı			113.599.859.373.000
	Saldırı Kombinasyonu Sayısı			227.199.718.746.000	Saldırı Kombinasyonu Sayısı			2.726.396.624.952.000

Örnek problem setleri, r_GA ve k_GA algoritmaları için ayrı ayrı olarak her set için belirlenen süreler kadar çalıştırılmaktadır. Elde edilen sonuçlar ve çalıştırma süreleri toplu şekilde Çizelge 7.10'da özet olarak sunulmuştur.

Çizelge 7.12. Örnek Problemler için k_GA ve r_GA Sezgisel Algoritmalarının Sıkı Orta ve Gevşek Koruma Katagorileri Performans Sonuçları

	Sıkı Koruma							
	SET 1		SET 2		SET 3		SET 4	
	k_GA	r_GA	k_GA	r_GA	k_GA	r_GA	k_GA	r_GA
Başlangıç Popülasyonu Değeri	0,254	0,266	2,36	1,95	1,027	1,205	2,18	3,910
Başlangıç Populasyon Süresi (sn)	5,05	4,85	238	27,29	14,15	7,64	135	11,69
En İyi Değer	0	0,145	0	0,965	0	0,48	0	1,075
En İyi Değer İterasyon No	36	4	39	27	12	7	34	41
Algoritmaların Çalışma Süresi	30 sn		15 dk		1 dk		5 dk	
Çalışma Süresindeki Son İterasyon No	42	6	109	31	25	7	62	41

	Orta Koruma							
	SET 5		SET 6		SET 7		SET 8	
	k_GA	r_GA	k_GA	r_GA	k_GA	r_GA	k_GA	r_GA
Başlangıç Popülasyonu Değeri	2,32	4,700	5,204	5,484	4,950	4,702	11,31	22,6
Başlangıç Populasyon Süresi (sn)	405	236	887	704	184	51	514	261
En İyi Değer	1,51	2,95	4,59	5,445	1,95	4,595	1,595	4,95
En İyi Değer İterasyon No	40	12	13	1	147	3	31	4
Algoritmaların Çalışma Süresi	60 dk		90 dk		30 dk		60 dk	
Çalışma Süresindeki Son İterasyon No	63	12	22	6	148	38	57	12

Çizelge 7.10. Örnek Problemler için k_GA ve r_GA Sezgisel Algoritmalarının Sıkı Orta ve Gevşek Koruma Katagorileri Performans Sonuçları (devam)

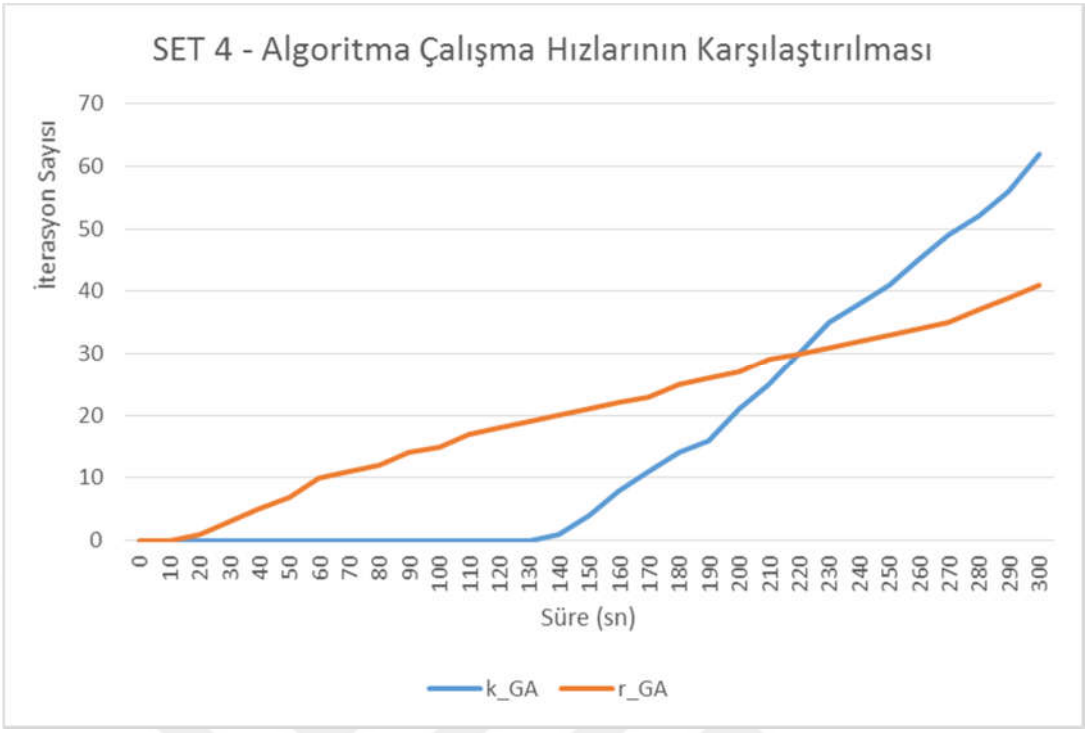
	Gevşek Koruma							
	SET 9		SET 10		SET 11		SET 12	
	k_GA	r_GA	k_GA	r_GA	k_GA	r_GA	k_GA	r_GA
Başlangıç Popülasyonu Değeri	5,75	5,75	13,61	5,75	5,75	10,3	5,75	5,75
Başlangıç Populasyon Süresi (sn)	91	22	121	133	12	12	68	67
En İyi Değer	4,5	5,64	5,27	3,37	5,575	6,87	5,575	5,75
En İyi Değer İterasyon No	286	15	76	7	58	6	232	0
Algoritmaların Çalışma Süresi	90 dk		120 dk		15 dk		90 dk	
Çalışma Süresindeki Son İterasyon No	300	116	232	42	300	80	271	72

Algoritmaların, çalışma süresi sonunda bulunduğu iterasyon sayıları incelendiğinde k_GA algoritmasının daha hızlı şekilde çalıştığı ve daha fazla iterasyon ile arama yaptığı görülmektedir. Bu duruma r_GA ve k_GA'nın parametreleri arasındaki fark sebep olmaktadır. Ayrıca her iki algoritmada da yasaklama kombinasyonu daha önce hesaplanmış ise tekrar aynı yasaklama kombinasyonu üretilirse, tekrar uygunluk değeri hesaplanmamaktadır. k_GA'da kural tabanlı seçilen bireyleri bir birine benzerdir ve bu bireylerden benzer yasaklama kombinasyonları üretilmektedir. Dolayısıyla bu k_GA'nın çalışma hızını arttırmaktadır.

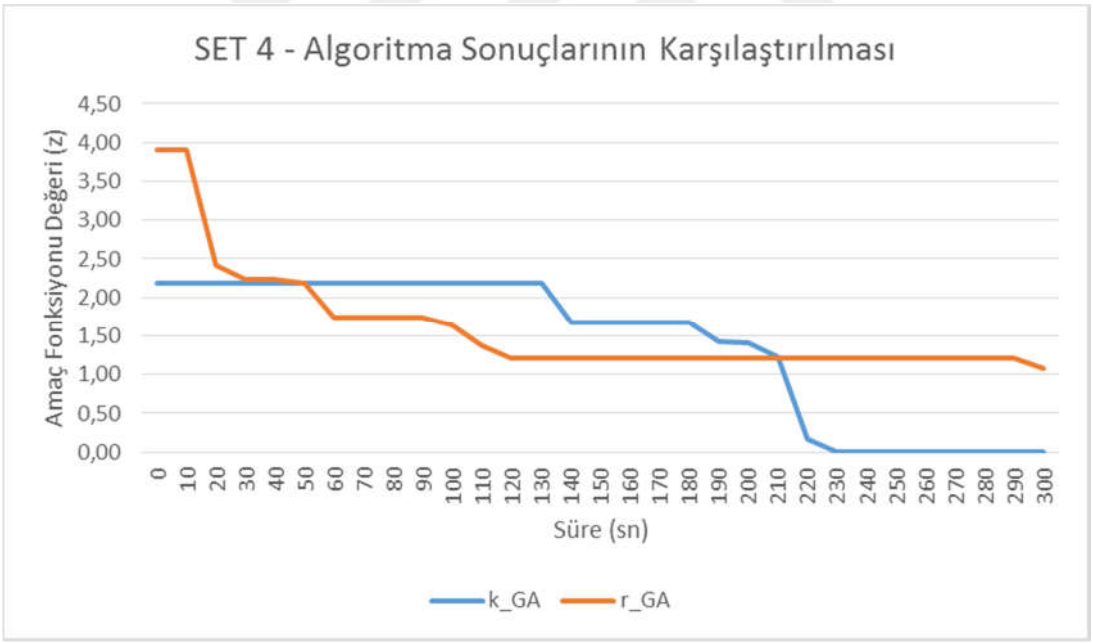
Elde edilen sonuçlar incelendiğinde ise tüm kategorilerde k_GA'nın daha iyi sonuç elde ettiği görülmektedir. Özellikle sıkı ve orta koruma kategorilerindeki örneklerde bu farkın daha belirgin olduğu görülmektedir. Gevşek koruma kategorisindeki örneklerin biri dışında k_GA'nın daha iyi sonuç verdiği görülmektedir. Ancak bu kategoride elde edilen sonuçlar birbirine yakındır. İki algoritma arasındaki bu farklılıkların istatistiksel olarak anlamlılığı ise 7.3 bölümünde detaylı olarak her bir koruma kategorisi için incelenmiştir.

Başlangıç popülasyonu süreleri incelendiğinde ise k_GA'nın başlangıç popülasyonunu üretme süresi daha uzundur. Bu durumda, k_GA algoritmasının kural tabanlı bireyleri belirlemeye yönelik gerçekleştirdiği işlemler etkili olmaktadır. Algoritmalar arasındaki farkı daha detaylı irdelemek için her bir kategoriden bir örneği algoritma çalışma hızı ve elde edilen sonuç değerleri üzerinden inceledik.

Öncelikle her koruma kategorisi için birer örnek seçildi (set 4, set 8 ve set 11). Sıkı koruma kategorisi seçilen set 4'de r_GA ve k_GA algoritmaları 300 sn. çalıştırılmıştır. Bu süre içinde k_GA algoritması 62 iterasyon çalışırken r_GA algoritması 41 iterasyon çalışmıştır. Bu örnekte her bir koruma kombinasyonu için iki yasaklama kombinasyonu üretilmektedir. Buda bir iterasyon için ihtiyaç duyulan sürenin düşük olmasına sebep olmaktadır. k_GA algoritmasında başlangıç popülasyonu üretirken mevcut çalıştırma süresinin önemli bir kısmını kullanmaktadır. Burada Kural tabanlı algoritmanın tanımlanmış kurallara göre "tüm koruma kombinasyonları için değerlendirme yaptığı adım" etkili olmaktadır. Ancak sonrasında k_GA algoritması mevcut süre içinde daha fazla iterasyon yapmıştır. Bu örnek problem için sürenin artması k_GA için iterasyon sayısını arttıracağı için avantaj sağlayacaktır. Elde edilen sonuçlara bakıldığında k_GA'nın daha iyi bir başlangıç değeri ile başladığı görülmektedir. Bu kural tabanlı seçilen bireylerin başlangıç popülasyonunda bulunmasının daha iyi bir popülasyon ile başlamayı sağladığını göstermektedir. Elde edilen sonuçlara baktığımızda k_GA 0 (sıfır) değerinde ulaşarak eniyi değere ulaştığı görülmektedir. r_GA ise elde ettiği 1,075 ile en iyi değerden uzaktır (Şekil 7.3). Ayrıca k_GA algoritması mevcut çalışma süresi içinde en iyi sonucu veren birden fazla koruma kombinasyonu tespit etmiştir.



(a)



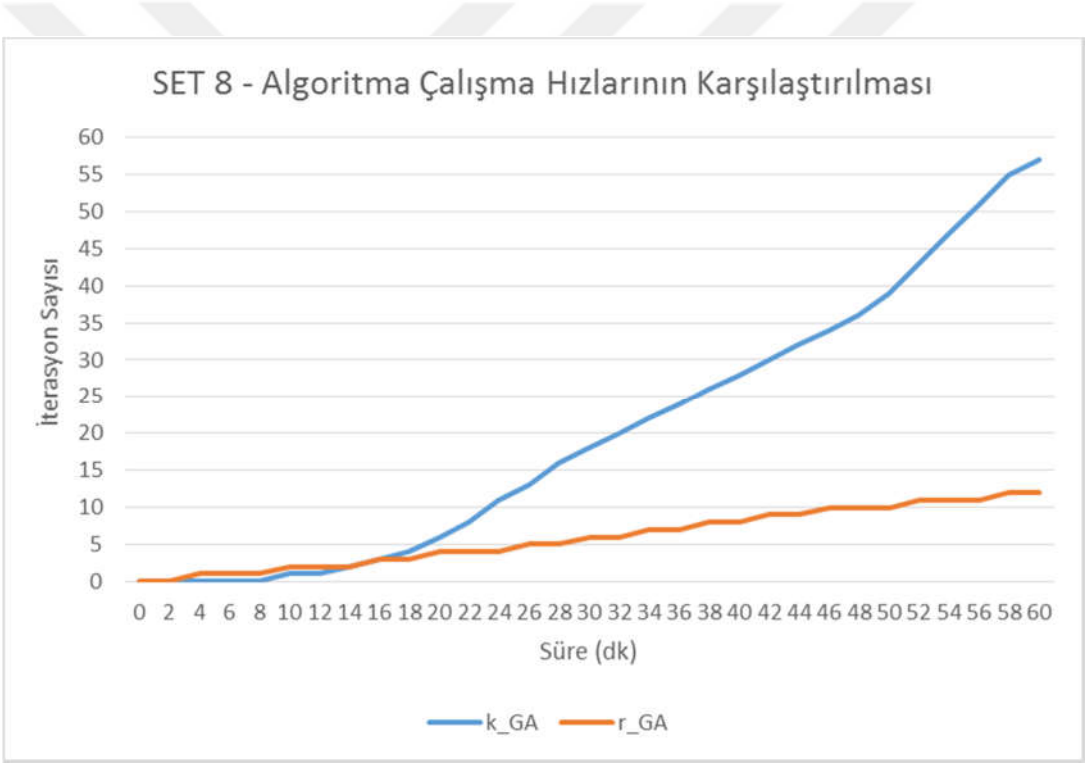
(b)

Şekil 7.3. Örnek 4 için GA Tabanlı Sezgisel Yaklaşımların Karşılaştırılması (a) Algoritma Çalışma Hızlarının Karşılaştırılması (b) Algoritma Sonuçlarının Karşılaştırması

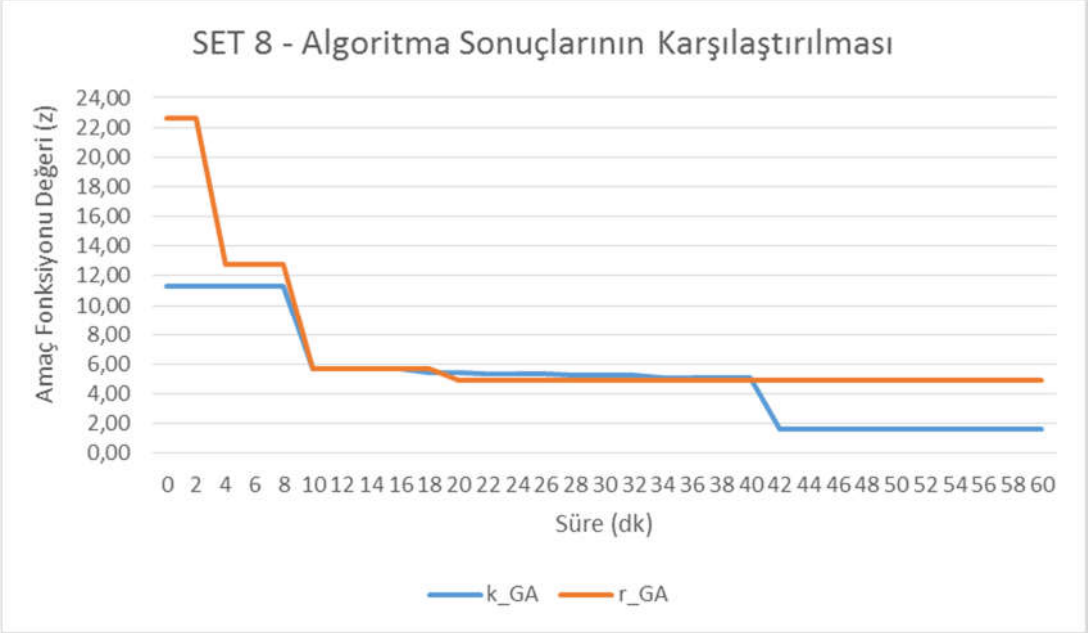
Koruma kaynağı açısından "orta koruma" kategorisinde sunulan Set 8 örneği Şekil 7.4'de incelendiğinde Algoritma çalışma hızı k_GA'da daha hızlıdır. Bu örnek için toplam çalışma süresi 60 dk.'dır ve bu sürede k_GA 57 iterasyon yaparken r_GA 12 iterasyon

yapabilmiştir. Bu örnekte iterasyon süresini etkileyen en önemli faktör koruma kombinasyonu başına gerçekleştirilen yasaklama kombinasyonudur. Örnekte her koruma kombinasyonu için 24 yasaklama kombinasyonu incelenmektedir. Burada yine k_GA r_GA algoritmaları arasında ki farklar ve algoritmaların ürettiği yasaklama kombinasyonlarını ve amaç fonksiyonu değerlerinin bellekte tutulması ve tekrarlanan yasaklama kombinasyonlarının tekrar 3. seviyeye gönderilmemesi etkili olmaktadır.

r_GA ve k_GA sezgisel algoritmalarının sonuçları kıyaslandığında k_GA'nın daha iyi bir sonuç elde etmiştir. r_GA algoritması ise iterasyon sayısının da az olması nedeni ile elde edilen sonucu iyileştirememiştir.



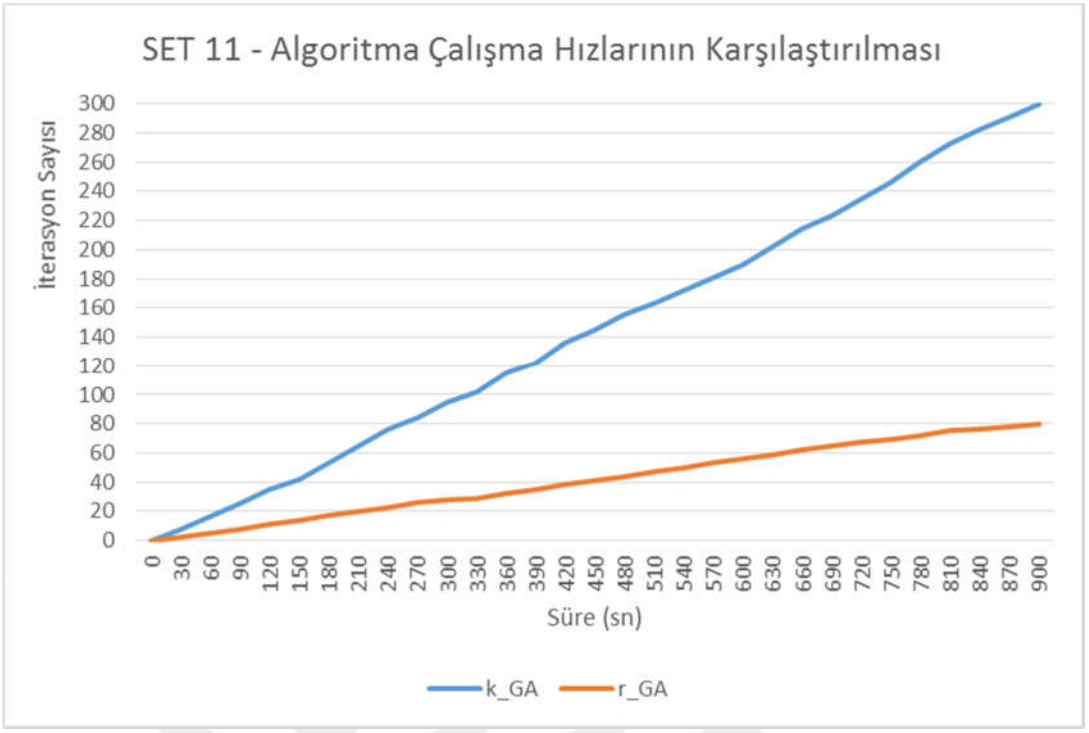
(a)



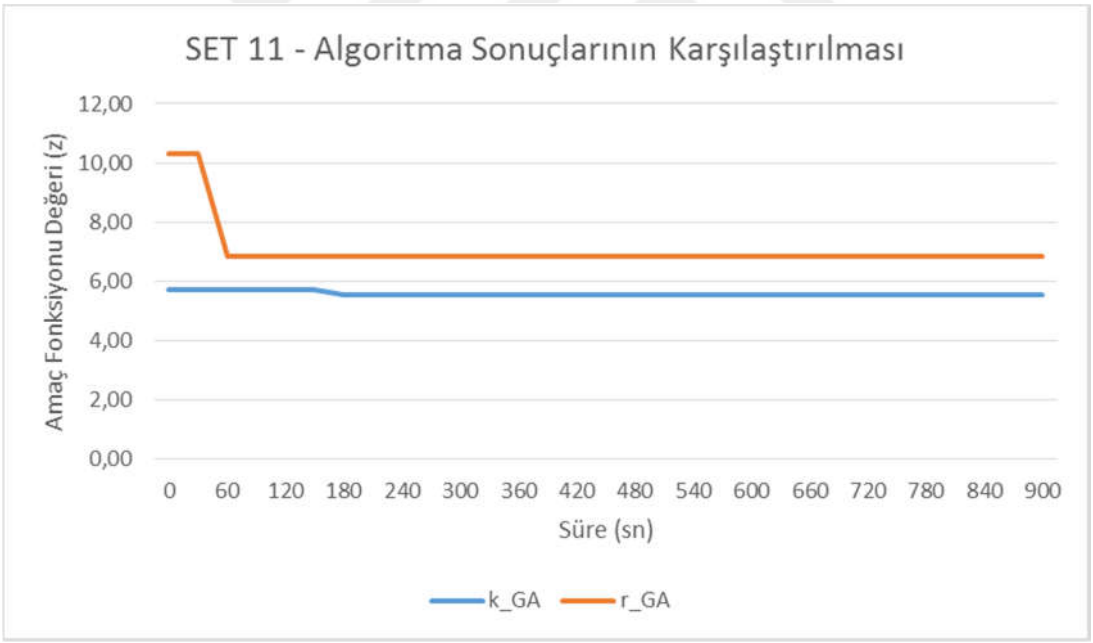
(b)

Şekil 7.4. Örnek 8 için GA Tabanlı Sezgisel Yaklaşımların Karşılaştırılması (a) Algoritma Çalışma Hızlarının Karşılaştırılması (b) Algoritma Sonuçlarının Karşılaştırması

Gevşek koruma kategorisine örnek olarak Şekil 7.5'de, Set 11 için elde edilen sonuçlar paylaşılmıştır. Bu kategoride bileşenlerin korunması kaynak yetersizliğinden dolayı seyrek olarak yapılmaktadır. Bu durum olası koruma kombinasyonu arttırmaktadır. Örnekte her koruma kombinasyonu için 3 yasaklama kombinasyonu incelenmektedir. Koruma kombinasyonu sayısının büyük olması çözüm uzayının büyük olmasına sebep olmaktadır. Geniş bir çözüm uzayının taranması sebebi ile r_GA'nın rassallığı ile daha iyi sonuç vermesi beklenmesine rağmen gevşek kategorideki bir örnek dışında yine k_GA daha iyi sonuçlar vermiştir. Tüm örnekler için başlangıç popülasyonlarının en iyi değerleri incelendiğinde çoğu örnekte k_GA daha iyi kromozomları seçtiği görülmektedir ve k_GA'nın başlangıç popülasyonu oluşturmada etkin olduğunu söylenebilir.



(a)



(b)

Şekil 7.5. Örnek 11 için GA Tabanlı Sezgisel Yaklaşımların Karşılaştırılması (a) Algoritma Çalışma Hızlarının Karşılaştırılması (b) Algoritma Sonuçlarının Karşılaştırması

Örnek problemler için deneysel sonuçlar incelendiğinde koruma kaynaklarının yoğun olduğu ve sıkı koruma olarak isimlendirdiğimiz kategori ile orta koruma kategorilerinin tamamındaki örneklerde, k_GA algoritması r_GA algoritmasına göre daha

iyi sonuçlar elde etmiştir. Koruma kaynaklarının daha seyrek olduğu ve dolayısıyla koruma kombinasyonu sayısının daha büyük olduğu örneklerde ise bir örnek hariç yine en iyi değerlere k_GA algoritması ulaşmıştır.

Elektrik sistemleri gibi kritik sistemler, yüksek koruma önlemlerinin uygulandığı sistemlerdir ve bu tarz sistemlerde k_GA sezgisel algoritmasının kullanımı daha etkin olacağı görülmektedir. Bu açıdan özellikle elektrik sistemleri, doğalgaz üretim ve iletim sistemleri gibi yüksek koruma önlemlerine sahip hizmet sunan kritik alt yapılar için bu yorum genelleştirilebilir.

Deneysel olarak elde edilen bu sonuçların istatistiksel olarak da anlamlı olup olmadığı izleyen bölümde gerçekleştirilen non-parametrik testlerle incelenmiştir.

7.3. GA Temelli k_GA ve r_GA Algoritmalarının Non-Parametrik Testler ile Analizi

Çalışmanın bu bölümünde öncelikle oluşturulan sıkı ortak ve gevşek koruma örnek grupları için gruplar arası bir farklılık olup olmadığını belirlemek üzere k-GA ve r_GA sezgisel algoritmaları için elde edilen sonuçlar, ayrı ayrı değerlendirilerek örnek grupları Kuruskal-Wallis testi ile analiz edilmiştir.

Kuruskal-Wallis testi sonrası koruma kategorileri arasında anlamlı bir farklılık olduğu görülmüştür. Daha sonra her bir koruma kategorisi için algoritmaların performansları arasında istatistiksel bir fark olup olmadığı Wilcoxon İşaretli Sıralar Testi ile analiz edilmiştir. Bu test işlemleri Minitab 19.2020.1 paket programı ile gerçekleştirilmiştir. Aşağıda işlemler ayrıntıları ile açıklanmaktadır.

k_GA algoritması çözüm sonuçlarına göre sıkı orta ve gevşek koruma kategorileri arası farklılığın Kuruskal-Wallis testi ile değerlendirilmesi

Bu analiz ile örnek setlerini, sıkı orta ve gevşek koruma kategorilerine ayırmanın, anlamlı bir farklılık yaratıp yaratmadığı test edilmiştir. Kuruskal-Wallis testi 3 yada daha fazla bağımsız grup arasında anakütle medyanlarının eşitlik durumunu sınamak için kullanılan non-parametrik bir istatistiksel test yöntemidir (Kuruskal ve Willas, 1952). Bu

işlemler Minitab 19.2020.1 programında Kruskal Wallis testi ile gerçekleştirilmiştir. Çizelge 7.11'de k_GA algoritmasına ait çözüm değerlerini içeren test verileri, bu test verilerinin Kruskal Wallis testi sonunda elde edilen tamamlayıcı istatistikler verilmiştir.

Çizelge 7.13. (a) Koruma Kategorilerine Karşı Gelen Sonuçların Farklılığının Analizi (k_GA), (b) Kuruskal-Wallis Testi Tamamlayıcı İstatistikler

SET No	Gruplar	k_GA Algoritma Sonuçları
1	Sıkı	0,000
2	Sıkı	0,000
3	Sıkı	0,000
4	Sıkı	0,000
5	Orta	1,510
6	Orta	5,590
7	Orta	1,950
8	Orta	1,595
9	Gevşek	4,500
10	Gevşek	5,270
11	Gevşek	5,575
12	Gevşek	5,575

Tamamlayıcı İstatistikler				
Kategori	N	Medyan	Ortalama Sıra	Z-Değeri
Sıkı	4	0,0000	2,5	-2,72
Orta	4	1,7725	7,5	0,68
Gevşek	4	5,4225	9,5	2,04
	12		6,5	

H_0: Gruplar arasında fark yoktur.

H_1: En az bir grup farklıdır.

Test sonucu elde edilen P-değeri 0,018'dir, $p<0.05$ anlamlılık düzeyinde, H0 hipotezi reddedilmiş ve H1 hipotezi kabul edilmiştir. Gruplar arasında fark vardır. Analize ait ayrıntılar ve Minitab çıktıları Ek-F'de verilmiştir.

r_GA algoritması sonuçlarına göre sıkı orta ve gevşek koruma kategorileri arası farklılığın Kuruskal-Wallis testi ile değerlendirilmesi

Bu analiz r_GA algoritması ile elde edilen sonuçlar içinde gerçekleştirilmiştir. Çizelge 12'de r_GA algoritmasına ait çözüm değerlerini içeren test verileri ve tamamlayıcı istatistikler verilmiştir.

Çizelge 7.14. (a) Koruma Kategorilerine Karşı Gelen Sonuçların Farklılığının Analizi (r_GA), (b) Kuruskal-Wallis Testi Tamamlayıcı İstatistikler

SET No	Gruplar	r_GA
1	Sıkı	0,145
2	Sıkı	0,965
3	Sıkı	0,480
4	Sıkı	1,075
5	Orta	2,950
6	Orta	5,545
7	Orta	4,595
8	Orta	4,950
9	Gevşek	5,640
10	Gevşek	3,370
11	Gevşek	6,870
12	Gevşek	5,750

Tamamlayıcı İstatistikler				
Kategori	N	Medyan	Ortalama Sıra	Z-Değeri
Sıkı	4	0,7225	2,5	-2,72
Orta	4	4,7725	7,3	0,51
Gevşek	4	5,6950	9,8	2,21
	12		6,5	

H_0: Gruplar arasında fark yoktur.

H_1: En az bir grup farklıdır.

Test sonucu elde edilen P-değeri 0,015'dir, $p<0.05$ anlamlılık düzeyinde, H0 hipotezi reddedilmiş ve H1 hipotezi kabul edilmiştir. Gruplar arasında fark vardır. Analize ait ayrıntılar ve Minitab çıktıları Ek-F'de verilmiştir.

Sıkı orta ve gevşek koruma kategorileri için k_GA ve r_GA algoritmalarının performanslarının istatistiksel olarak karşılaştırılması

Grupların farklılıkları belirlendikten sonra her bir grup için önerilen k_GA ve r_GA sezgisel algoritmalarının performansları arasında ki farklılıklar Wilcoxon non-parametrik testi ile analiz edilmiştir. Wilcoxon testi bağımlı örneklerin non-parametrik olarak farklarının olup olmadığını analiz etmek için kullanılan bir test yöntemidir. Wilcoxon İşaretli Sıralar Testi olarak da isimlendirilir (Wilcoxon, 1945). Aynı örneğe ait farklı sonuçların farkının anlamlı olup olmadığının analizidir. Algoritmaların çözüm sonuçlarına bakıldığında 12 test problemi için biri hariç tamamında k_GA algoritmasının daha iyi değerler elde ettiği görülmüştür. Bu farkın istatistiksel olarak bir anlam ifade edip etmediği ise Wilcoxon testi ile her kategori için k_GA ve r_GA algoritmalarının test sonuçları karşılaştırılarak belirlenmiştir. Wilcoxon testindeki örnek sayısı arttırılarak daha anlamlı sonuçlar elde edilmesi amacıyla üretilen 12 test problemi aynı koşullarda tekrar çözülerek her kategori için örnek sayısı iki katına çıkarılmıştır. Çizelge 13'de Wilcoxon non parametrik testi için kullanılan çözüm sonuçları ve teste girecek olan fark değerleri test düzenine uygun olarak verilmiştir.

Çizelge 7.15. Sıkı Orta ve Gevşek Koruma Kategorileri için k_GA ve r_GA Algoritmalarının 1- sample Wilcoxon Test Veri Düzeni

Sıkı Koruma			
Set No	k_GA	r_GA	Fark
1	0	0,145	-0,145
2	0	0,965	-0,965
3	0	0,48	-0,48
4	0	1,075	-1,075
1	0	0,175	-0,175
2	0	0,305	-0,305
3	0	0,745	-0,745
4	0	0,745	-0,745

Orta Koruma			
Set No	k_GA	r_GA	Fark
5	1,51	2,95	-1,44
6	4,59	5,445	-0,855
7	1,95	4,595	-2,645
8	1,595	4,95	-3,355
5	1,43	2,65	-1,22
6	3,49	5,75	-2,26
7	1,755	3,86	-2,105
8	1,15	4,75	-3,6

Gevşek Koruma			
Set No	k_GA	r_GA	Fark
9	4,5	5,64	-1,14
10	5,27	3,37	1,9
11	5,575	6,87	-1,295
12	5,575	5,75	-0,175
9	4,15	5,64	-1,49
10	5,27	5,65	-0,38
11	5,575	5,575	0
12	4,475	5,75	-1,275

<u>Sıkı Koruma kategorisinde</u> k_GA ve r_GA algoritmalarının Wilcoxon testi ile değerlendirilmesi;

η: sıkı koruma kategorisi için k_GA ve r_GA algoritmaları farkının medyanı olmak üzere, Çizelge 7.14'de tamamlayıcı istatistikler görülmektedir.

Çizelge 7.16. Sıkı Koruma Kategorisinde, Algoritmaların Farkı için Tamamlayıcı İstatistikler

Algoritmaların Farkı	N	Medyan
k_GA – r_GA	8	-0,59

Hipotezler:

H_0: η = 0 Algoritmaların sıkı koruma kategorisi için performansları arası fark yoktur.

H_1: η ≠ 0 Algoritmaların sıkı koruma kategorisi için performansları arası fark vardır.

Analiz sonucu elde edilen P-değeri 0,014'dür, $p<0.05$ anlamlılık düzeyinde, H0 hipotezi reddedilmiştir. Algoritmalar arasında sıkı koruma kategorisi için performans farkı vardır. k_GA algoritması r_GA algoritmasına göre sıkı koruma kategorisi için daha iyi performans göstermektedir.

<u>Orta Koruma kategorisinde</u> k_GA ve r_GA algoritmalarının Wilcoxon testi ile değerlendirilmesi;

η: orta koruma kategorisi için k_GA ve r_GA algoritmaları farkının medyanı olmak üzere, Çizelge 7.15'de tamamlayıcı istatistikler görülmektedir.

Çizelge 7.17. Sıkı Koruma Kategorisinde, Algoritmaların Farkı için Tamamlayıcı İstatistikler

Algoritmaların Farkı	N	Medyan
k_GA – r_GA	8	-2,205

Hipotezler:

H_0: η = 0 Algoritmaların orta koruma kategorisi için performansları arası fark yoktur.

H_1: η ≠ 0 Algoritmaların orta koruma kategorisi için performansları arası fark vardır.

Analiz sonucu elde edilen P-değeri 0,014'dür, p<0.05 anlamlılık düzeyinde**,** H0 hipotezi reddedilmiştir. Algoritmalar arasında orta koruma kategorisi için performans farkı vardır. k_GA algoritması r_GA algoritmasına göre orta koruma kategorisi için daha iyi performans göstermektedir.

<u>Gevşek Koruma kategorisinde</u> k_GA ve r_GA algoritmalarının Wilcoxon testi ile değerlendirilmesi;

η: Gevşek koruma kategorisi için k_GA ve r_GA algoritmaları farkının medyanı olmak üzere, Çizelge 7.16'da tamamlayıcı istatistikler görülmektedir.

Çizelge 7.18. Gevşek Koruma Kategorisinde, Algoritmaların Farkı için Tamamlayıcı İstatistikler

Algoritmaların Farkı	N	Medyan
k_GA – r_GA	8	-0,69125

Hipotezler:

H_0: η = 0 Algoritmaların gevşek koruma kategorisi için performansları arası fark yoktur.

H_1: η ≠ 0 Algoritmaların gevşek koruma kategorisi için performansları arası fark vardır.

Analiz sonucu elde edilen P-değeri 0,272'dür, p<0.05 anlamlılık düzeyinde**,** H0 hipotezi kabul edilmiş. Algoritmalar arasında gevşek koruma kategorisi için performans farkı istatistiksel olarak yoktur. k_GA algoritması r_GA algoritmasına göre gevşek koruma kategorisi için daha iyi performans gösterdiği söylenemez.

8. BULGULAR VE TARTIŞMA

Bu bölümde tez kapsamında elde edilen en temel bulgular özetlenmekte olup ayrıntılı sonuçlar izleyen bölümde açıklanmaktadır. Bölüm 4.1 ve 4.2'de sunulan modeller askeri ve kritik tesisler göz önüne alınarak tasarlanmıştır. Konunun hassasiyeti ve güvenlik konularında gerçek veriye ulaşma güçlüğü, test verisi türetmeyi gerektirmiştir. Örnek problem çözüldüğünde R_eIMF modelinin eniyi koruma planlarını türettiği görülmüştür. Aynı şekilde $SD - R_e$IMF modeli de öncelikle arz-talep dengesinin bozulmasına neden olacak yasaklama tiplerine karşı koruma planlarını belirlemiş ve bunların içinden en düşük maliyetli koruma planlarını ve tesis – talep merkezi eşleştirmelerini belirlemiştir.

Söz konusu tehditlerin sadece terörist eylemler, savaş vb. istenmeyen durumlarda ortaya çıkabilecek eylem planları ile ilişkili düşünülmemesi gerektiği, bir elektrik şebekesinde rassal olarak karşılaşılabilecek bir arızanın da geliştirilen modellerde ki saldırı olarak kullanılabileceği düşünülürse bu yaklaşımların kullanımının yaygın etkisi olabilecektir.

Bölüm 5.3'de sunulan Üç seviyeli çoklu saldırı tipli elektrik şebekesi koruma modeli için olası tüm koruma ve bu koruma kombinasyonlarına karşı gelen saldırı kombinasyoları türetilerek bu durumlara karşılık gelen en iyi amaç fonksiyonu değerini bulmak üzere bir çözüm yaklaşımı geliştirilmiştir. Bu yaklaşım ile problem boyutu büyüdüğünde çözüm elde edilememiştir. Bu amaçla geliştirilen GA temelli r_GA ve k_GA sezgisel yaklaşımları önerilmiştir. Bu yaklaşımların performansları bölüm 7.1'de karşılaştırılmış ve örnek problemler için en iyi amaç fonksiyonu değerlerine ulaşıldığı görülmüştür.

r_GA ve k_GA sezgisel algoritmaları ise kendi aralarında IEEE'nin 14 baralı gerçek veri seti üzerinden türetilmiş 3 ayrı kategorideki 12 örnek set ile kıyaslanmıştır. Yapılan bu deneysel çalışmanın sonuçları incelendiğinde ise sıkı koruma olarak isimlendirdiğimiz koruma kaynaklarının yoğun olduğu vakalar ve orta koruma kategorileri için k_GA algoritmasısını r_GA algoritmasına göre daha iyi performans verdiği görülmektedir. Elektrik sistemleri gibi kritik sistemler, yüksek koruma önlemlerinin uygulandığı sistemlerdir. Elektrik sistemleri ve benzer yüksek koruma önlemlerine sahip hizmet sunan

sistemlerde başlangıç popülasyonun kural tabanlı olarak oluşturulduğu k_GA sezgisel algoritmasının kullanımının daha etkin olacağı söylenebilir

9. SONUÇ VE ÖNERİLER

Kritik tesisler, toplum için önemli, herhangi bir kesintiye uğradığında toplumu önemli derecede etkileyebilecek hizmetler sunmaktadır. Bu tesisler, bir ülke için yüksek önem düzeyine sahip, herhangi bir sebeple, savaş, terörist saldırı, doğal afetler vb. gibi kendilerine verilecek zarar neticesinde hizmet sunmaları engellendiğinde, ülkeyi önemli ölçüde etkileyebilecek tesislerdir. Ülkemizin bulunduğu coğrafya ve günümüz çıkar çatışmaları göz önüne alındığında çok çeşitli tehditler söz konusudur ve bu tehditlere karşı kritik tesislerin savunulması önemli konuların başında gelmektedir.

Bu çalışmada kritik tesislerin korunması problemi ele alınmıştır. Kritik tesislere yapılacak olası saldırılara karşı, koruma planlamasının belirlenmesine yönelik, karar vericiler için yeni modeller ve bu modellerin çözüm yaklaşımları geliştirilmiştir.

Problem literatürde yasaklama (interdiction) problemleri olarak çalışılan ve Stackelberg'in Oyun Teorisi'ne dayanan türdedir. Yasaklama problemleri ağ yasaklama ve tesis yasaklama problemleri olarak ikiye ayrılmaktadır. Tesis yasaklama bir ağ üzerinde ki bağlantı noktalarının yasaklanması iken ağ yasaklama ise bu bağlantı noktaları arasındaki arkların yasaklanmasını ele almaktadır.

Çalışmada, ilk katkı olarak, daha önce göz ardı edilmiş olan farklı saldırı tipleri ve karşı gelen savunma tipleri göz önüne alınmıştır. Kritik tesislere saldırı düzenleyenlerin saldırı yapabilecekleri farklı alternatifler mevcuttur. Aynı şekilde kritik tesislerin korunması amacı ile saldırı tiplerine yönelik farklı önlemler de alınabilir. Kritik tesislerin korunması problemi için hangi tesislerin hangi tip saldırılara karşı korunması gerektiğini göz önüne alan (yeni model 1) ve sınırsız kaynağın olmadığı durumlarda, saldırganın arz talep dengesi bozma öncelikli yapacağı saldırıları göz önüne alan (yeni model 2) iki yeni matematiksel model geliştirilmiştir. Önerilen bu modeller genel olarak kritik tesislerin korunmasında kullanılabilecek modellerdir. Bu modellerin saldırı/koruma tipleri bakış açısı göz önüne alınarak daha özel bir alt alanda, elektrik şebekelerinin korunması problemi ele alınmıştır.

Elektrik şebekeleri, zarar görmesi durumunda pek çok diğer tesisin çalışmasını etkilemesi nedeni ile kritik tesisler içinde en önemlilerinden biri olarak görülmüştür. Elektrik sistemlerinde oluşabilecek bir kesinti, hem ekonomik hem de toplumda oluşturacağı etki açısından önemlidir. Bu tesislerin olası saldırılara karşı etkin bir şekilde savunulması gerekmektedir. Bu kapsamda elektrik sistemlerine özgü kısıt ve parametreler göz önüne alınarak saldırı/koruma tipi temelinde yeni bir "Üç Seviyeli Çoklu Saldırı Tipli Elektrik Şebekesi Koruma modeli" (yeni model 3) önerilmiştir. Ayrıca modelin amaç fonksiyonu kritiklik seviyesi daha düşük olan noktalardan yük atma gerçekleştirmek üzere düzenlenmiştir.

Önerilen matematiksel modelin çözümü için önce tüm koruma kombinasyonlarının tarandığı bir çözüm algoritması Phyton ile geliştirilmiştir. Gerçekleştirilen deneysel çalışmalar ve yapılan karmaşıklık analizi ile ele alınan modellerin NP- zor olduğu görülmüştür. Bileşen sayısındaki artış problemin polinom olarak büyümesine sebep olurken, tip sayısındaki artış problemin üssel olarak büyümesine sebep olmaktadır.

NP-zor olan bu probleme çözüm bulmak amacıyla GA tabanlı iki sezgisel yaklaşım geliştirilmiştir. Bu sezgisel yaklaşımlar, başlangıç popülasyonun, rassal olarak oluşturulduğu r_GA ve kural tabanlı bir yapı ile oluşturulduğu k_GA algoritmalarıdır. Geliştirilen algoritmaların en iyi parametre değerlerini belirlemek üzere Taguchi Yöntemi ile deney tasarımı gerçekleştirilmiştir. Bu analiz ile algoritmaların, tüm parametreleri için eniyi performansı veren düzeyleri ayrı ayrı belirlenmiştir.

Oluşturulan elektrik şebekesi modeli ve önerilen algoritmalar, IEEE'nin 14 baralı gerçek veri seti çözülerek incelenmiştir. Bu veri setinden, koruma kaynağı ve saldırı kaynaklarında yapılan değişikler ile 12 set örnek problem türetilmiştir. Bu örnekler koruma kaynağı yoğunluğuna göre sıkı-orta ve gevşek olmak üzere üç farklı kategoriye ayrılmıştır. k_GA ve r_GA algoritmalarının performansları, örnek setler için elde ettikleri sonuçlar ile karşılaştırılmıştır. Tüm kategorilerde k_GA algoritması daha iyi sonuç değerleri elde etmiştir. Bu farklılığın anlamlılığı gerçekleştirilen non-parametrik testler ile analiz edilmiştir ve k_GA algoritması sıkı ve orta koruma kategorileri için üstünlüğü istatistiksel olarak anlamlı bulunmuştur. Özellikle sıkı ve orta koruma kategorisine giren elektrik sistemleri için k_GA algoritması daha etkindir. Koruma kaynağı yoğunluğuna göre oluşturulan

kategorilerin anlamlılığı yine k_GA ve r_GA algoritması test sonuçları üzerinden değerlendirilmiştir ve kategoriler arası anlamlı bir farklılık vardır.

Çalışmanın hedef problemi aslında spesifik bir tesis ve bu tesise yapılabilecek sonlu sayıda saldırı türü ve bağlı olarak sonlu sayıda koruma türü çerçevesindedir. Bu tür kritik tesislere yapılabilecek saldırılar diğer NP-zor problemlerle karşılaştırıldığında göreceli olarak özellikle pahalı teknolojiler söz konusu ise oldukça az sayıdadır, problemi NP zor yapan faktör ağırlıklı olarak kombinasyon sayılarındaki büyüklüktür. Bu nedenle tanımlanan problemin ve karşı gelen çözüm yaklaşımlarının çok sayıda tesis ve bu tesislere yapılabilecek çok sayıda saldırı türü ve koruma türü kapsamında olmadığı varsayılmakta olup gerçekçi olanın da bu sınırlama olduğu düşünülmektedir.

İlerleyen çalışmalarda, önerilen modeller ve algoritmalar şu şekilde geliştirilebilir:

- Çalışmada bilinçli olarak yapılan (kasıtlı) saldırılar göz önüne alınarak modeller geliştirilmiştir. Doğal yollar ya da arıza gibi sebepler ile gerçekleşecek kesintiler için modeller geliştirilmeye devam edebilir.
- Ele alınan elektrik şebekesi modeli için savunanın ve saldıranın ilgili bölge için kritiklik algılarının aynı olduğu kabul edilmiştir. Ancak saldıranın amacı, kendi önceliklerine göre değişebileceğinden saldırı yaparken belirlediği kritiklik değeri ile savunma planlayıcının kritiklik değeri farklılık gösterebilir. Modellerde bu durum göz önüne alınabilir.
- Çalışmada bir saldırı çeşidine karşı korunma kavramı ile aslında o saldırı türüne "uygun" ve o türde saldırıyı bertaraf edebilecek korunma tipinin kullanıldığı varsayılmıştır, ancak bazı durumlarda bir savunma (korunma) çeşidi birden fazla tipte saldırıyı engelleyebilir. Bu durumları içerecek şekilde modeller geliştirilebilir.
- Çalışmada kritik tesislerin yerlerinin belli olduğu varsayılmaktadır. Bu tesislerin yerlerinin kesin bilinmediği durumlar için modeller geliştirilebilir.
- Saldırgan, farklı tiplerde ve belirli sayılarda saldırı yapabilme yeteneği ve donanımına sahiptir ve bu bilinmektedir. Bunun bilinmediği durumlar için olasılığa dayalı çalışmalar geliştirilebilir.
- Bir koruma kaynağı bir tesise atanmaktadır. Bir koruma kaynağı birden fazla tesisi aynı anda koruyabilir, bu durum göz önüne alınabilir. Koruma kaynaklarının,

konuşlanacağı yere göre bir alanı korumasının söz konusu olduğu modeller geliştirilebilir.

- Bu çalışmada k_GA sezgisel algoritmasının uzman görüşleri ile oluşturulan kural seti, algoritmanın elde ettiği sonuçlar ile kendini yetiştirerek bu kuralları kendi belirleyeceği bir şekilde geliştirilebilir.
- Literatürde elektrik şebekesi problemlerinde, saldırı tiplerinin göz önüne alınmamış olması nedeni ile önerilen k_GA ve r_GA algoritmaları kendi aralarında karşılaştırılmıştır. İlerleyen çalışmalarda, bu algoritmalar IEEE'nin 14 baralı gerçek veri setinden türetilen test problemleri kullanılarak farklı sezgisel algoritmalar ile karşılaştırılabilir.

KAYNAKLAR DİZİNİ

Agudelo, L., Lezama, J. M., Munoz-Galeano, N., 2015, Vulnerability assessment of power systems to intentional attacks using a specialized genetic algorithm", DYNA 82 78–84.

Aksen, D., Aras, D., 2011, A bilevel fixed charge location model for facilities under imminent attack, Computers & Operations Research 39, 1364–1381.

Aksen, D., Akca, S.Ş., Aras, N., 2014, A bilevel partial interdiction problem with capacitated facilities and demand outsourcing, Computers & Operations Research 41, 346–358

Aksen, D., Piyade, N., Aras, N., 2010, The budget constrained r-interdiction median problem with capacity expansion, CEJOR 18,269–291

Aliakbarian, N., 2015, A bi-level programming model for protection of hierarchical facilities under imminent attacks, Computers & Operations Research 64, 210–224.

Arroyo, J.M., Galiana, F.D., 2005, On the Solution of the Bilevel Programming Formulation of the Terrorist Threat Problem, IEEE Transactions On Power Systems 20

Assimakopoulos, N., 1987, A network interdiction model for hospital infection control, Computers in Biology and Medicine 17 (6), 413–422.

Alguacil, N., Delgadillo, A., Arroyo, J. M., 2014, A trilevel programming approach for electric grid defense planning, Computers & Operations Research 41, 282–290.

Bier, V. M., Gratz, E. R., Haphuriwa,t N. J., Magua, W., Wierzbicki, K. R., 2007, Methodology for identifying near-optimal interdiction strategies for a power transmission system, Reliability Engineering and System Safety 92, 1155–1161.

Bergen, A. R., 1986, Power system analysis, Prentice Hall, p.191

Bachmann, P, 1894, Analytische zahlen theorie (Analytic Number Theory) (in German). 2. Leipzig: Teubner.

Bard, J.F., 1991, Some Properties of the bi-level programing problem, Journal of Optimization Theory And Applications, 68(2), 371-378

Cappanera, P., Scaparra, M. P., 2011, Optimal allocation of protective resources in shortest-path networks, Transportation Science 45 (1), 64–80.

Church, RL, Scaparra MP, Middleton RS, 2004, "Identifying critical infrastructure: the median and covering facility interdiction problems." Ann Assoc Am Geogr 94(3):491–502

KAYNAKLAR DİZİNİ (devam)

Cormican, K. J., Morton, D. P., Wood, R. K., 1998, Stochastic network interdiction, Operations Research, 46 (2), 184–197.

Çakır, H., 1989, Enerji iletimi, 3-19, Birsen Yayınevi, İstanbul.

Delgadillo, A., Arroyo, J.M., 2010, Analysis of Electric Grid Interdiction With Line Switching, IEEE Transactions On Power Systems, 25.

Fard, A.M.F., Hajiahhaei-Keshteli, M., 2018, A bi-objective partial interdiction problem considering different defensive systems with capacity expansion of facilities under imminent attacks, Applied Soft Computing, 68, 343–359

Farley, J. D., 2003, Breaking al qaeda cells: A mathematical analysis of counterterrorism operations (a guide for risk assessment and decision making), Studies in Conflict & Terrorism 26 (6), 399–411.

GAMS, 2019, An introduction to GAMS, https://www.gams.com/products/introduction/, erişim tarihi: 14.02.2019

Gen, M., Cheng, R.,1997, Genetic algorithms and engineering design, John Wiley & Sons Inc., Hoboken.

Ghaffarinasaba, N., Atayi R., 2018, An implicit enumeration algorithm for the hub interdiction median problem with fortification, European Journal of Operational Research 267, 23–39

Goldberg, D. Deb, K. 1991, A comparative analysis of selection schemes used in genetic algorithms. In Foundations of Genetic Algorithms, ed. G. Rawlins, Morgan Kaufmann, San Mateo, p. 69-93.

Israeli, E., Wood, R. K., 2002, Shortest-path network interdiction, Networks 40 (2), 97–111.

Jian, G.J., Liu, X., Sun, L., Yin, J., 2015, Optimal allocation of protective resources in urban rail transit networks against intentional attacks, Transportation Research Part E 84,73–87.

Jiang, J., Liu, X., 2018, Multi-objective Stackelberg game model for water supply networks against interdictions with incomplete information, European Journal of Operational Research, 266, 920-933.

Hamilton, W.D., Ridley, M., 2005, Narrow roads of gene land, The Collected Papers Of W.D. Hamilton, Oxford University Press, p.142.

KAYNAKLAR DİZİNİ (devam)

Hagelstam, A. , Narinen, K., 2018, Cooperating to counter hybrid threats, https://www.nato.int/docu/review/2018/Also-in-2018/cooperating-to-counter-hybrid-threats/EN/index.htm, erişim tarihi: 20.04.2019.

Holand, J., 1975, Adaptation In Natural and Artificial Systems, University of Michigan Press, Ann Arbor.

Holmgren, A.J, Jenelius, E., Westin, J., 2007, Evaluating strategies for defending electric power networks against antagonistic attacks, IEEE Transactions On Power Systems, 22, No. 1.

Keçici, S., Aras, N., Verter, V., 2012 Facility network design under the threat of terrorist attacks, Springer Verlag, Optim Lett, 6, 1101–1121 DOI 10.1007/s11590-011-0412-1.

Kruskal W. H., Wallis W. A., 1952, Use of ranks in one-criterion variance analysis, Journal of the American Statistical Association Vol. 47 No.26, 583–621,

Landau, E., 1909, "Handbuch der Lehre von der Verteilung der Primzahlen (Handbook on the theory of the distribution of the primes) (in German), Leipzig: B. G. Teubner. p. 883.

Lezama, J.M.L., Gomez, J.C., Galeano, N. M., 2017, Assessment of the electric grid interdiction problem using a nonlinear modeling approach, Electric Power Systems Research 144, 243–254.

Liberatore, F.,Scaparra, M.P., Daskin, M.S., 2011, Analysis of facility protection strategies against an uncertain number of attacks: The stochastic R-interdiction median problem with fortification, Computers & Operations Research, 357–366.

Losada, C., Scaparra, M.P., Church, R.L., 2010, On a bi-level formulation to protect uncapacitated p-median systems with facility recovery time and frequent disruptions, Electronic Notes in Discrete Mathematics, 36, 591–598.

Losada, C., Scaparra, M.P., Church, R. L., Daskin, M. S., 2012-b, The stochastic interdiction median problem with disruption intensity levels, Ann Operations Research, 201, 345–365, DOI 10.1007/s10479-012-1170-x.

Losada, C., Scaparra, M.P., O'Hanley, J.R., 2012-a, Optimizing system resilience: A facility protection model with recovery time, European Journal of Operational Research, 217, 519–530.

KAYNAKLAR DİZİNİ (devam)

Mahmoodjanloo, M., Parvasi, S.P., Ramezanian, R., 2016, A tri-level covering fortification model for facility protection against disturbance in r-interdiction median problem, Computers & Industrial Engineering, 102, 219–232.

McMasters, A. W., Mustin, T. M., 1970, Optimal interdiction of a supply network, Naval Research Logistics Quarterly, 17, 261–268.

Mitchell, M., 1999, An Introduction to Genetic Algorithms, A Bradford Book The MIT Press, p.102.

Morton, D. P., Pan F., Saeger, K. J., 2007, Models for nuclear smuggling interdiction, IIE Transactions, 39 (1), 3–14.

Moore, J. T., Bard, J. F., 1990, The mixed integer linear bilevel programming problem, Operations Research, 38(5), 911–921.

Motto, A.L., Arroyo, J.M., Galiana, F.D., 2005, A mixed-integer lp procedure for the analysis of electric grid security under disruptive threat", IEEE Transactions On Power Systems, Vol. 20, No. 3.

Obitko, N., 1998, Introduction to genetic algorithms, Gzech Technical University.

Pasia, J., M., Harmosilla, A.,Y., Ombao, H., 2005, A useful tool for statistical estimation: genetic algorithm, Journal of Statistical Computation and Simulation, 75, 237-251

PYTHON, 2019, BeginnersGuide/Overview, https://wiki.python.org/moin/BeginnersGuide/Overview, erişim tarihi: 14.02.2019

Ramamoorthy, P., Jayaswal, S., Sinha, A., Vidyarthi, N., 2016, Hub interdiction & hub protection problems: model formulations & exact solution methods, Indian Institute Of Management Ahmedabad, 380 015, India.

Ramamoorthy, P., Jayaswal, S., Sinha, A., Vidyarthi, N., 2018, Multiple allocation hub interdiction and protection problems: Model formulations and solution approaches", European Journal of Operational Research 1–16.

Reeves, C., Rowe, J.E., 2002, Genetic algorithms: principles and perspectives, Operations Research/Computer Science Interfaces Series, p.77.

Salmeron, J., Wood, K., Baldick, R., 2004, Analysis of electric grid security under terrorist threat, IEEE Transactions On Power Systems, Vol. 19, NO. 2.

KAYNAKLAR DİZİNİ (devam)

Salmeron, J., Wood, K., 2015, The value of recovery transformers in protecting an electric transmission grid against attack, IEEE Transactions On Power Systems, Vol. 30, NO. 5.

Santoso, S., Hapsari, D., Susiloatmadja, R., 2019, Complexity Analysis and Performance of the Madenda Filter Algorithm, Fourth International Conference on Informatics and Computing (ICIC).

Scaparra, M.P., Church, R.L., 2006, A bilevel mixed-integer program for critical infrastructure protection planning.

Scaparra, M.P., Church, R.L., 2008, An exact solution approach for the interdiction median problem with fortification, European Journal of Operational Research 189, 76–92

Sivanandam, S. N., Deepa, S. N., 2008, Introduction to genetic algorithms, Springer-Verlag Berlin Heidelberg, p.23.

Shimizu, K., Ishizuka, Y., Bard, J. F., 2012, Nondifferentiable and two-level mathematical programming, Springer Science & Business Media.

Taguchi G, Chowdhury S., Taguchi S., 2000, Robust engineering, New York : McGraw-Hill.

TEİAŞ, 2019, Türkiye Elektrik İletim A.Ş. 2019-2023 Stratejik Planı, https://www.teias.gov.tr/tr-TR/stratejik-plan, erişim tarihi: 12.02.2020.

University of Washington - Electrical Engineering, 1962, Power system test case archive, http://labs.ece.uw.edu/pstca/pf14/pg_tca14bus.htm, erişim tarihi: 20.12.2019

Ünver, M., 2010, Kritik Altyapıların Korunması, BTK Raporu

Üstünel, M., 2012, I. sınıf elektrik tesisatçılığı, enerji üretimi iletimi ve dağıtımı, MEB yayınları, Bölüm 10, s.236

Washburn, A., Wood, K., 1995, Two-person zero-sum games for network interdiction, Operations Research 43 (2), 243–251.

Wilcoxon, F., 1945, Individual comparisons by ranking methods, Biometrics, 1, 80-3.

Wollmer, R., 1964, Removing Arcs from a Network, Operations Research, 12(6), 934-940.

Wood, R. K., 1993, Deterministic network interdiction, Mathematical and Computer Modelling 17 (2), 1–18.

KAYNAKLAR DİZİNİ (devam)

Wu, X., Conejo, A.J., 2017, An efficient tri-level optimization model for electric grid defense planning, IEEE Transactions On Power Systems, Vol. 32, No. 4.

Yao, Y., Edmunds, T., Papageorgiou, D., Alvarez, R., 2007, trilevel optimization in power network defense, IEEE Transactions On Systems, Man, And Cybernetics—Part C: Applications and Reviews, Vol. 37, No. 4.

Zhang, X., Zheng, Z., Zhu, Y., Cai, K, 2014, Protection issues for supply system involving random attacks, Computers & Operations Research, 43, 137–156.

Zhu, Y., Zheng, Z., Zhang, X., Cai, K., 2013, The r-interdiction median problem with probabilistic protection and its solution algorithm, Computers & Operations Research, 40, 451–462 .

EK AÇIKLAMALAR

Ek Açıklama-A: Yeni Model 1 ve Yeni Model 2'nin Matematiksel Model Çözümleri Uygulama Örneği Parametreleri

Ek Açıklama-B: r_GA ve k_GA Algoritmaları Deney Tasarımı Minitab Sonuçlar

Ek Açıklama-C: Genetik algoritma tabanlı sezgisel algoritma yazılımı Python kodları

Ek Açıklama-D: Python ile kodlanmış matematiksel model çözüm algoritmasının çözüm sonuçları (Örnek Problem için)

Ek Açıklama-E: GA tabanlı sezgisel ile çözülen örnek problemin parametreleri ve çözüm popülasyonları

Ek Açıklama-F: r_GA ve k_GA Algoritmalarının Non_Parametrik Testler ile Performans Karşılaştırması Analiz Sonuçları

Ek Açıklama-A: Yeni Model 1 ve Yeni Model 2'nin Matematiksel Model Çözümleri Uygulama Örneği Parametreleri

Yeni Model 1 ve Yeni Model 2'nin testi için hizmet sunan dört kritik tesis (j), hizmet alan dört talep merkezi (i) ve iki farklı yasaklama (e) ve karşılık gelen iki farklı koruma tipinin (f) olduğu türetilen bir test problemi kullanılmıştır. Koruma kaynağı beş, saldırı kaynağı ise iki olarak alınmıştır.

Probleme ilişkin parametreler:

Çizelge A.1. Kritik Tesisler (j) ile Talep Merkezleri (i) Arası Mesafeler ve Talep Merkezleri Talep Miktarları

i \ j	1	2	3	4	Talep (a_i)
1	12	11	11	10	10
2	16	17	19	18	12
3	20	22	16	21	15
4	10	21	10	15	20

Kritik tesis (j)'nin yasaklanması sonrası hizmet alınabilecek, kendisinden daha uzak mesafeli kritik tesisleri (k) gösteren L_{ijk} katsayısı seti Çizelge 2'de verilmiştir. Bu katsayı değerleri, (i) talep noktasının (k) tesisine olan mesafesi, (j) tesisine olan mesafesine eşit yada daha uzak mesafede ise 1, diğer durumda 0 değerini alır.

Çizelge A.2. L_{ijk} Katsayılar Matrisi

i, j \ k	1	2	3	4
1.1	0	0	0	0
1.2	1	0	0	0
1.3	1	0	0	0
1.4	1	1	1	0
2.1	0	1	1	1
2.2	0	0	1	1
2.3	0	0	0	0
2.4	0	0	1	0
3.1	0	1	0	1
3.2	0	0	0	0
3.3	1	1	0	1
3.4	0	1	0	0
4.1	0	1	0	1
4.2	0	0	0	0
4.3	0	1	0	1
4.4	0	1	0	0

Ek Açıklama-B: r_GA ve k_GA Algoritmaları Deney Tasarımı Minitab Sonuçları

r_GA parametreleri için Taguchi Deney Tasarımı ve Analiz Sonuçları

Test No	A1	B1	C1	D1	E1	r_GA test Sonuçlar	SNRA1
1	20	30	1	0,2	0	0,42625	7,406712
2	20	30	1	0,2	30	0,33875	9,402414
3	20	30	1	0,2	60	0,19625	14,14381
4	20	50	2	1	0	0,27375	11,25292
5	20	50	2	1	30	0,33875	9,402414
6	20	50	2	1	60	0,2725	11,29267
7	20	60	3	1,5	0	0,30125	10,42146
8	20	60	3	1,5	30	0,16375	15,71637
9	20	60	3	1,5	60	0,23	12,76544
10	30	30	2	1,5	0	0,3825	8,347371
11	30	30	2	1,5	30	0,16375	15,71637
12	30	30	2	1,5	60	0,256	11,8352
13	30	50	3	0,2	0	0,58625	4,638343
14	30	50	3	0,2	30	0,1525	16,3346
15	30	50	3	0,2	60	0,46	6,744843
16	30	60	1	1	0	0,30625	10,27848
17	30	60	1	1	30	0,13125	17,63801
18	30	60	1	1	60	0,3825	8,347371
19	50	30	3	1	0	0,44025	7,126013
20	50	30	3	1	30	0,5635	4,982122
21	50	30	3	1	60	0,2725	11,29267
22	50	50	1	1,5	0	0,53525	5,428866
23	50	50	1	1,5	30	0,465	6,650941
24	50	50	1	1,5	60	0,45275	6,882831
25	50	60	2	0,2	0	0,816667	1,759103
26	50	60	2	0,2	30	0,6575	3,642085
27	50	60	2	0,2	60	0,65625	3,658614

Linear Model Analysis: SN ratios versus A; B; C; D; E

Estimated Model Coefficients for SN ratios

Term	Coef	SE Coef	T	P
Constant	9,3744	0,6020	15,572	0,000
A 20	1,9372	0,8514	2,275	0,037
A 30	1,7235	0,8514	2,024	0,060
B 30	0,6537	0,8514	0,768	0,454
B 50	-0,6379	0,8514	-0,749	0,465
C 1	0,2011	0,8514	0,236	0,816
C 2	-0,8292	0,8514	-0,974	0,345
D 0.2	-1,8488	0,8514	-2,172	0,045
D 1	0,8048	0,8514	0,945	0,359
E 0	-1,9678	0,8514	-2,311	0,034
E 30	1,6796	0,8514	1,973	0,066

Model Summary

S	R-Sq	R-Sq(adj)
3,1281	66,16%	45,01%

Analysis of Variance for SN ratios

Source	DF	Seq SS	Adj SS	Adj MS	F	P
A	2	181,113	181,113	90,557	9,25	0,002
B	2	7,510	7,510	3,755	0,38	0,687
C	2	10,102	10,102	5,051	0,52	0,606
D	2	46,399	46,399	23,200	2,37	0,125
E	2	60,986	60,986	30,493	3,12	0,072
Residual Error	16	156,557	156,557	9,785		
Total	26	462,667				

Unusual Observations for SN ratios

Observation	SN ratios	Fit	SE Fit	Residual	St Resid	
14	16,335	10,919	1,997	5,416	2,25	R

R denotes an observation with a large standardized residual.

Response Table for Signal to Noise Ratios

Smaller is better

Level	A	B	C	D	E
1	11,312	10,028	9,575	7,526	7,407
2	11,098	8,736	8,545	10,179	11,054
3	5,714	9,359	10,002	10,418	9,663
Delta	5,598	1,292	1,457	2,893	3,647
Rank	1	5	4	3	2

Settings

A	B	C	D	E
20	30	3	1.5	30

Prediction

S/N Ratio	Mean
15,3168	0,116309

k_GA parametreleri için Taguchi Deney Tasarımı ve Analiz Sonuçları

Test No	A2	B2	C2	D2	E2	F2	k_GA Test Sonucları	SNRA1
1	20	30	1	0.2	0	10	0,54875	5,212509
2	20	30	1	0.2	30	40	0,16375	15,71637
3	20	30	1	0.2	60	60	0,0875	21,15984
4	20	50	2	1	0	10	0,29875	10,49384
5	20	50	2	1	30	40	0,24	12,39578
6	20	50	2	1	60	60	0,418	7,576474
7	20	60	3	1.5	0	10	0,27575	11,18969
8	20	60	3	1.5	30	40	0,2075	13,65964
9	20	60	3	1.5	60	60	0,22775	12,85083
10	30	30	2	1.5	0	40	0,47	6,558043
11	30	30	2	1.5	30	60	0,51375	5,784963
12	30	30	2	1.5	60	10	0,60725	4,33265
13	30	50	3	0.2	0	40	0,805	1,884082

14	30	50	3	0.2	30	60	0,47	6,558043
15	30	50	3	0.2	60	10	0,25675	11,80979
16	30	60	1	1	0	40	0,4925	6,151875
17	30	60	1	1	30	60	0,53125	5,494021
18	30	60	1	1	60	10	0,45875	6,768478
19	50	30	3	1	0	60	0,51875	5,700838
20	50	30	3	1	30	10	0,64875	3,758453
21	50	30	3	1	60	40	0,40375	7,877749
22	50	50	1	1.5	0	60	0,3925	8,123207
23	50	50	1	1.5	30	10	0,54075	5,340069
24	50	50	1	1.5	60	40	0,78375	2,116449
25	50	60	2	0.2	0	60	0,49025	6,191648
26	50	60	2	0.2	30	10	0,453	6,878036
27	50	60	2	0.2	60	40	0,47	6,558043

Linear Model Analysis: SN ratios versus A; B; C; D; E; F

Estimated Model Coefficients for SN ratios

Term	Coef	SE Coef	T	P
Constant	8,07931	0,7144	11,310	0,000
A 20	4,17124	1,0103	4,129	0,001
A 30	-1,93021	1,0103	-1,911	0,077
B 30	0,37640	1,0103	0,373	0,715
B 50	-0,71290	1,0103	-0,706	0,492
C 1	0,37434	1,0103	0,371	0,717
C 2	-0,66048	1,0103	-0,654	0,524
D 0.2	1,02828	1,0103	1,018	0,326
D 1	-0,72181	1,0103	-0,714	0,487
E 0	-1,24534	1,0103	-1,233	0,238
E 30	0,31906	1,0103	0,316	0,757
F 10	-0,77003	1,0103	-0,762	0,459
F 40	0,02269	1,0103	0,022	0,982

Model Summary

S	R-Sq	R-Sq(adj)
3,7120	60,55%	26,74%

Analysis of Variance for SN ratios

Source	DF	Seq SS	Adj SS	Adj MS	F	P
A	2	235,325	235,325	117,662	8,54	0,004
B	2	6,868	6,868	3,434	0,25	0,783
C	2	5,924	5,924	2,962	0,21	0,809
D	2	15,051	15,051	7,525	0,55	0,591
E	2	22,596	22,596	11,298	0,82	0,461
F	2	10,368	10,368	5,184	0,38	0,693
Residual Error	14	192,902	192,902	13,779		
Total	26	489,034				

Unusual Observations for SN ratios

Observation	SN ratios	Fit	SE Fit	Residual	St Resid	
1	5,213	12,014	2,576	-6,802	-2,54	R
3	21,160	15,703	2,576	5,457	2,04	R

R denotes an observation with a large standardized residual.

Response Table for Signal to Noise Ratios

Smaller is better

Level	A	B	C	D	E	F
1	12,251	8,456	8,454	9,108	6,834	7,309
2	6,149	7,366	7,419	7,358	8,398	8,102
3	5,838	8,416	8,365	7,773	9,006	8,827
Delta	6,412	1,089	1,035	1,750	2,172	1,517
Rank	1	5	6	3	2	4

Ek Açıklama-C: Genetik Algoritma Tabanlı Sezgisel Algoritma Yazılımı Python Kodları

```
from gams import *
import os
import sys
import createprogramyeni
import time
import itertools
import random
import copy


def get_model_text(args):
    ret = createprogramyeni.convertVariableTables(args)
    return ret


def readVariableFromProgramTxt():
    lines = createprogramyeni.readFile('programyeni.txt')
    values = dict()
    for l in lines:
        data = l.split()
        if len(data) < 3:
            continue
        if data[0] == 'variable':
            values[data[1]] = int(data[2])
    return values


varVals = readVariableFromProgramTxt()
i_program = varVals['i']
ne_program = varVals['ne']


def readNumbersFromProgramTxt():
    global i_program, ne_program, varVals
    lines = createprogramyeni.readFile('programyeni.txt')
    values = dict()
    for l in lines:
        data = l.split()
        if len(data) < 3:
            continue

        if data[0] == 'variable':
            if data[1] == 'j':
                values['santral_sayisi'] = int(data[2])
            elif data[1] == 'n':
                values['trafo_merkezi_sayisi'] = int(data[2])
elif data[1] == 'l':
                values['hat_sayisi'] = int(data[2])
            elif data[1] == 'je':
                values['santral_tip_sayisi'] = int(data[2])
            elif data[1] == 'ne':
                values['trafo_tip_sayisi'] = int(data[2])
            elif data[1] == 'le':
                values['hat_tip_sayisi'] = int(data[2])
            elif data[1] == 'i':
```

```
                i_program = int(data[2])
            elif data[1] == 'ne':
                ne_program = int(data[2])

    if len(values) != 6:
        print("!!!!!!!!!!!!!!!!!!!!! readNumbersFromProgramyeniTxt
eksik sayida deger okundu  : " + str(len(values)))
    return values


def getProtectionAttackNumbers():
    pass


def convert_list_to_str(t):
    return (' '.join([str(elem) for elem in list(map(convert_list_to_str,
t))])  if isinstance(t, (list, tuple)) else str(t)).strip()

evaluated_results= dict()
def getKalkanDeger(line,variableArguments,ws):
    lnn= convert_list_to_str(line)
    if lnn in evaluated_results.keys():
        return evaluated_results[lnn]
    removeJunk()
    startt=time.time()

    result=0
    start=True
    for args in variableArguments:

        with open('prog5yeni.txt','w') as ff:
            ff.write(get_model_text(args))

        t1 = ws.add_job_from_string(get_model_text(args))
        t1.run()

        returnCode=t1.out_db["ms"][0].value

        for rec in t1.out_db["Amac"]:
            if start:
                result=rec.level
                start=False
            else:
                if rec.level>result:
                    result=rec.level

        res = "z = " + str(rec) + '     args : ' + str(args)
    endd=time.time()
    evaluated_results[lnn]=result
    return result


def getNumberOfProtectionFromUser(values, typee, savunmaSaldiriType):
    obj_type = 0
    name = ''

    savunmaSaldiriTypeName = ''
    if savunmaSaldiriType == 1:
        savunmaSaldiriTypeName = 'savunma '
    else:
```

```
        savunmaSaldiriTypeName = 'saldiri '

    if typee == 1:
        name = 'santral'
        obj_type = values['santral_tip_sayisi']
    elif typee == 2:
        name = 'trafo'
        obj_type = values['trafo_tip_sayisi']
    elif typee == 3:
        name = 'hat'
        obj_type = values['hat_tip_sayisi']
    else:
        print "bir hata var !!!!!!!!!!!!!!!!!!!
getNumberOfProtectionFromUser"
        return

    userInput = raw_input(
        name + ' icin aralarinde bir bosluk birakarak ' +
savunmaSaldiriTypeName + ' degerlerini toplamda ' + str(
            obj_type) + ' deger olacak sekilde giriniz.  :  \n')
    data = userInput.split()
    if len(data) != obj_type:
        print "bir hata var !!!!!!!!!!!!!!!!!!!
getNumberOfProtectionFromUser " + str(
            obj_type) + ' deger olacak sekilde girilmesi gerekiyordu
!!!!!!!!.'
        return

    intData = [int(i) for i in data]
    return intData


def getNumbersFromUser(values):
    santralSavunma = getNumberOfProtectionFromUser(values, 1, 1)
    santralSaldiri = getNumberOfProtectionFromUser(values, 1, 2)

    trafoSavunma = getNumberOfProtectionFromUser(values, 2, 1)
    trafoSaldiri = getNumberOfProtectionFromUser(values, 2, 2)

    hatSavunma = getNumberOfProtectionFromUser(values, 3, 1)
    hatSaldiri = getNumberOfProtectionFromUser(values, 3, 2)

    return [santralSavunma, santralSaldiri, trafoSavunma, trafoSaldiri,
hatSavunma, hatSaldiri]


def getValue(arr):
    res = 0
    for i in range(len(arr)):
        res += (2 ** i) * arr[i]
    return res


def getPermutaionsWithoutRepeat(slots, ones, vall=2):
    numbers = dict()
    result = []
    arr = [vall for i in range(ones)] + [0 for i in range(slots - ones)]
        allperms = itertools.islice(itertools.permutations(arr,
slots),1000)
```

```
    for prob in allperms:
        val = getValue(prob)
        if val in numbers:
            continue
        else:
            numbers[val] = 1
            result.append(prob)
    return result


def createProtection(numberOfObjects, numberOfTypes, data):
    valuesForTypePerSantral = []
    for i in range(numberOfTypes):

valuesForTypePerSantral.append(getPermutaionsWithoutRepeat(numberOfObject
s, data[i]))

    return valuesForTypePerSantral


def createProtectionListForOneObject(numberOfObjects, numberOfTypes,
data):
    liste = createProtection(numberOfObjects, numberOfTypes, data)
    numberOfvals = []
    result = list(itertools.product(*liste))
    return result


def createProtectionList(values, savunmaData):
    santralkoruma =
createProtectionListForOneObject(values['santral_sayisi'],
values['santral_tip_sayisi'], savunmaData[0])

    trafokoruma =
createProtectionListForOneObject(values['trafo_merkezi_sayisi'],
values['trafo_tip_sayisi'], savunmaData[2])

    hatkoruma = createProtectionListForOneObject(values['hat_sayisi'],
values['hat_tip_sayisi'], savunmaData[4])

    alldat = [santralkoruma, trafokoruma, hatkoruma]

    result = list(itertools.product(*alldat))
    return result


def getZeros(santralData):
    num = 0
    for i in santralData:
        if i == 0:
            num += 1
    return num

def combineTwo(org2, created2):
    created = list(created2)
    org = list(org2)
    created.reverse()
    for i in range(len(org)):
        if org[i] == 0:
            org[i] = created.pop()
```

```
    for i in range(len(org)):
        if org[i] == 2:
            org[i] = 0

    return org


def santralTipSaldir(santralTipData, santralTipSaldiri):
    slots = getZeros(santralTipData)
    santralWithAttack = getPermutaionsWithoutRepeat(slots,
santralTipSaldiri, 1)
    santralWithAttack2 = []
    for i in santralWithAttack:
        santralWithAttack2.append(combineTwo(santralTipData, i))
    return santralWithAttack2


def santralSaldir(santralData, santralSaldiri):
    santrals = []
    for i in range(len(santralData)):
        santralTipdat = santralTipSaldir(santralData[i],
santralSaldiri[i])
        santrals.append(santralTipdat)
    return list(itertools.product(*santrals))


def saldir(prob, savunmaSaldiriData):
    res = []
    for i in range(len(prob)):
        santralTrafoveyaHat = santralSaldir(prob[i], savunmaSaldiriData[i
* 2 + 1])

    return list(itertools.product(*res))


def combineTipsBySantral(res):
    for i in range(len(res)):
        rs = []
        for j in res[i]:
            p = zip(*j)
            rs.append(p)
    return rs


def combineAllLinesBySantral(lines):
    res = []
    for l in lines:
        res.append(combineTipsBySantral(l))
    return res


def convertListToStr(liste):
    allstrs = []
    for ll in liste:
        strlist = [str(i) for i in ll]
        res = ' '.join(strlist)
        allstrs.append(res)

    res = ' '.join(allstrs)
    return res
```

```
def convertListToStr_new(liste):
    New_list = remove_type2(liste)
    allstrs = []
    strlist = [str(i) for i in New_list]
    res = ' '.join(strlist)
    allstrs.append(res)
    res = ' '.join(allstrs)
    return res

def convertOneLineOfIntLists(line):
    res = []
    for l in line:
        res.append(convertListToStr_new(l))
    return res


def convertOneProbOfAttacks(lines):
    res = []
    for line in lines:
        res.append(convertOneLineOfIntLists(line))
        return res


def addZerosToTheEndOfMiddle(lines):
    global ne_program, i_program
    addStr = ' ' + ('0 ' * (ne_program * i_program)).strip()
    # print "*** ",addStr
    res = []
    for i in range(len(lines)):
        ll = [lines[i][0], lines[i][1] + addStr, lines[i][2]]
        res.append(ll)
    return res


def removeJunk():
    lst = os.listdir('.')
    for i in lst:
        if i.startswith('_'):
            o.remove(i)


def remove_type(lines):
    for lines1 in lines:
        res1 = []
        for lines2 in lines1:
            res = []
            res1.append('')
            for l in lines2:
                if k > 0:
                    k1 = [1]
                    res = res+ k1
                else:
                    k2 = [0]
                    res = res + k2
            res1.extend(res)
            res1.append("'")
        return res
    return res1
```

```
def remove_type2(lines):
    res1 = []
    res = []
    for l in lines:
        k = int(sum(l))
        if k > 0:
            k1 = [1]
            res = res + k1
        else:
            k2 = [0]
            res = res + k2
        res1.extend(res)
    return res


def convertSaldir(lines):
    res = []
    for l in lines:
        rs = []
        for ll in l:
            rs.append(zip(*ll))
        res.append(rs)
    return res

def print_lines(res):
    for l in res:
        print l


def listit(t):
    return list(map(listit, t)) if isinstance(t, (list, tuple)) else t


def calculate_forrules(res):
    k=0
    newList=[]
    for l in res:
        k +=1

        kombinasyon=l
        a=0
        kritik_list=[KritikS,KritikTM,KritikH]
        zz = 0
        #print kritik_list
        total=0
        for ll in kombinasyon:

            parca=ll
            aaaa=int(kritik_list[zz])-1
            zz += 1
            parca_sonuc= 1

            for lll in parca:
                oge= lll

                sonuc = int(lll[aaaa])
                parca_sonuc = parca_sonuc * sonuc

            parcam= parca_sonuc/int(type[zz-1])
            total+=parcam
```

```
        newList.append(total)

    return newList
    enbindex= newList.index(max(newList))
    print enbindex

def rules_based_choose(res):
    newList = calculate_forrules(res)
    b = [i for i, j in enumerate(newList) if j == max(newList    c =
random.sample(b,12)
print "Kural Tabanli Secilen kombinasyonlar=", c
    baslangicList=[]
    for i in c:
        baslangicList.append(res[i])
    return (baslangicList)


def create_initial_set(savunmaData):                        res =
listit(createProtectionList(values, savunmaData))
    Listofrules = rules_based_choose(res)
    Popsize=20
    A=Popsize-len(Listofrules)
    res=random.sample(res,A) + Listofrules
    return res


def evaluate_set(res,savunmaData):
    removeJunk()
    if len(sys.argv) > 1:
        ws = GamsWorkspace(system_directory=sys.argv[1])
    else:
        ws = GamsWorkspace(working_directory='.')
    returnDict=[]

    minimumRes = 1000000000000
    iii = 0
    for l in res:
        iii += 1
        lines = saldir(l, savunmaData)
        lines = convertSaldir(lines)
        lines3 = convertOneProbOfAttacks(lines)
result = getKalkanDeger(l,lines3, ws)
                if result<minimumRes:
            minimumRes=result
                returnDict.append((l, result))
    print "minumum result : " + str(minimumRes)
    return returnDict


def add__and_sort_sets(set1,set2,ratio):
    liste1 = [(k,v,2) for  (k,v) in set1]
    liste2 = [(k,v,1) for  (k,v) in set2]
    liste3=liste1+liste2
    liste3.sort(key=lambda tup: (tup[1],tup[2]))
    liste3= liste3[:int(len(liste3)*(ratio/2))]
    liste3= [(k,v) for (k,v,n) in liste3]
    return liste3


def crossover(person1,person2):
```

```
    person3 = copy.deepcopy(person1)
    A= 1
    if A==1:
        l1=random.randrange(0,3)
        cpoint = random.randrange(0, (len(person3[l1])))
        person3[l1][cpoint] = person2[l1][cpoint]

    if A==2:
        l1=random.randrange(0,3)
        l2=random.randrange(0,3)
        cpoint = random.randrange(0, (len(person3[l1])))
        person3[l1][cpoint] = person2[l1][cpoint]
        cpoint = random.randrange(0, (len(person3[l2])))
        person3[l2][cpoint] = person2[l2][cpoint]
        #print "**", person3
    if A==3:
        for i in range(3):
            cpoint=random.randrange(0,(len(person3[i])))
            person3[i][cpoint]= person2[i][cpoint]
    return person3

def hamming_distance(string1, string2):
string1=string1[0]+string1[1]+string1[2]
    string1=string1[0]+string1[1]+string1[2]+string1[3]+string1[4]
    string2 = string2[0] + string2[1] + string2[2]
    string2=string2[0]+string2[1]+string2[2]+string2[3]+string2[4]
    distance = 0
    L = len(string1)
    for i in range(L):
        if string1[i] != string2[i]:
            distance += 1
        return distance

def newcrossover(prevgen,CL):
    population_size = 6
    Popi = population_size + 1
    positionvalue=[]
    new_parent_childs = []
    for i in range(len(prevgen)):
         positionvalue.append(hamming_distance(CL,prevgen[i]))
        for j in range(1,20):
        for i in range(0,20):            #DOE1
            if positionvalue[i] == j:
                person2=[]
                person2.append(prevgen[i])
                Z=crossover(CL,person2[0])
                new_parent_childs.append(Z)
        return(new_parent_childs[0:5])


def mutation(person):
    for i in range(len(person)):
        j=random.randrange(0, len(person[i]))
        if 0 in person[i][j] and 2 in person[i][j]:
            ind0= random.choice([index for index, value in
enumerate(person[i][j]) if value == 0])
            ind2= random.choice( [index for index, value in
enumerate(person[i][j]) if value == 2])
            person[i][j][ind0]= 2
            person[i][j][ind2] = 0
```

```
def crete_new_set_from_old(prev_gen
    new_set = []
    for j in range(20):
        new_set.append(crossover(prev_gen[random.randrange(0,
len(prev_gen))], prev_gen[random.randrange(0, len(prev_gen))]))
    random.shuffle(new_set)
    return new_set


def CL_bulma(set):
    set1=[]
    for i in range(len(set)):
        set1.append((set[i])[1])
    A=min(set1)
    AA=set1.index(A)
    C=set[AA]
    return (C[0])

def create_new_generation(prev_gen_evaluated,new_gen ,savunmaData):
    new_gen_evaluated= evaluate_set(new_gen,savunmaData)
    CL= CL_bulma(new_gen_evaluated)
    Komsular=newcrossover(new_gen,CL)
    Komsular_evaluated=evaluate_set(Komsular,savunmaData)
    new_gen_evaluated += Komsular_evaluated
    best_of_two_evaluated=
add__and_sort_sets(prev_gen_evaluated,new_gen_evaluated,0.53)
    remaining_evaluated = [item for item in new_gen_evaluated if item not
in best_of_two_evaluated]
    mutation_list = [k for (k,v) in remaining_evaluated[0:1]]    #DOE4
       for i in range(len(mutation_list)):
        mutation(mutation_list[i])
    mutation_list_evaluated = evaluate_set(mutation_list,savunmaData)
    XRC= len(prev_gen_evaluated)-len(best_of_two_evaluated)-
len(mutation_list_evaluated)-XXXRC
    if XRC<0:
        XRC=0
    bulk_of_second_evaluated=
random.sample(remaining_evaluated,XXXRC)+random.sample(prev_gen_evaluated
,XRC)
    eval_list=
best_of_two_evaluated+bulk_of_second_evaluated+mutation_list_evaluated
    list_only = [k for (k, ) in eval_list]
    return list_only,eval_list


def write_population_to_list(f,population_evaluated,info):
    f.write(info+"\n")
    for (k,v) in population_evaluated:
        f.write(str(v)+"  "+repr(k)+"\n")
    f.write("\n\n")


if __name__ == "__main__":
    values = readNumbersFromProgramTxt()
    print "values : ", values
    savunmaData = getNumbersFromUser(values)
    print "savunmaData : ", savunmaData
    f = open('generation_results.txt', 'w')
```

```
    startto = time.time()
    set_n_minus_1=create_initial_set(savunmaData)
    set_n_minus_1_evaluated = evaluate_set(set_n_minus_1,savunmaData)
    write_population_to_list(f,set_n_minus_1_evaluated,"initial set")
    enddo=time.time()
    print "(rule based) initial set complate time =" + str(enddo-startto)
    print "+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+"

    iteration_number = 600
    for i in range(int(iteration_number)):
        startt = time.time()
        print "itreasyon  *** ",i+1
        new_set= crete_new_set_from_old(set_n_minus_1)
        set_n_minus_1,
set_n_minus_1_evaluated=create_new_generation(set_n_minus_1_evaluated,
new_set, savunmaData)
        write_population_to_list(f, set_n_minus_1_evaluated, "iteration
"+str(i+1))
        result_of_iter=min(set_n_minus_1_evaluated, key=lambda t: t[1])
        print "RESULT OF ITERATION : ",result_of_iter
        endd = time.time()
        print "Iteration time: " + str(endd - startt)
        print "-------------------------------------"
    f.close()
```

Ek Açıklama-D: Python ile Kodlanmış Matematiksel Model Çözüm Algoritmasının Çözüm Sonuçları (Örnek Problem için)

C:\Python26\python.exe C:/Users/ob/Documents/workspace/python/gams_api_deneme/bizim.py

C:\Users\ob\Documents\workspace\python\gams_api_deneme

values : {'santral_tip_sayisi': 2, 'hat_tip_sayisi': 1, 'trafo_tip_sayisi': 2, 'santral_sayisi': 2, 'trafo_merkezi_sayisi': 3, 'hat_sayisi': 7}

santral icin aralarinde bir bosluk birakarak savunma degerlerini toplamda 2 deger olacak sekilde giriniz. :

2 1

santral icin aralarinde bir bosluk birakarak saldiri degerlerini toplamda 2 deger olacak sekilde giriniz. :

1 1

trafo icin aralarinde bir bosluk birakarak savunma degerlerini toplamda 2 deger olacak sekilde giriniz. :

3 2

trafo icin aralarinde bir bosluk birakarak saldiri degerlerini toplamda 2 deger olacak sekilde giriniz. :

1 1

hat icin aralarinde bir bosluk birakarak savunma degerlerini toplamda 1 deger olacak sekilde giriniz. :

6

hat icin aralarinde bir bosluk birakarak saldiri degerlerini toplamda 1 deger olacak sekilde giriniz. :

1

savunmaData : [[2, 1], [1, 1], [3, 2], [1, 1], [6], [1]]

C:\Users\ob\Documents\workspace\python\gams_api_deneme

z = Amac: level=7.0 args : ['0 0 0 1', '0 0 0 0 0 1 0 0 0 0', '0 0 0 0 0 0 1'] -----> 1.0

time : 0.360000133514

1 -) result : 7.0

orginal: l (((2, 2), (2, 0)), ((2, 2, 2), (2, 2, 0)), ((2, 2, 2, 2, 2, 2, 0),))

greatest result so far : 7.0

C:\Users\ob\Documents\workspace\python\gams_api_deneme

z = Amac: level=5.0 args : ['0 0 0 1', '0 0 0 0 0 1 0 0 0 0', '0 0 0 0 0 1 0'] -----> 1.0

time : 0.327999830246

```
2  -) result : 5.0

orginal: l (((2, 2), (2, 0)), ((2, 2, 2), (2, 2, 0)), ((2, 2, 2, 2, 2, 0, 2),))

greatest result so far : 5.0

--------------------------

C:\Users\ob\Documents\workspace\python\gams_api_deneme

z = Amac:  level=5.0     args : ['0 0 0 1', '0 0 0 0 0 1 0 0 0 0', '0 0 0 0 1 0 0'] -----> 1.0

time : 0.453000068665

3  -) result : 5.0

orginal: l (((2, 2), (2, 0)), ((2, 2, 2), (2, 2, 0)), ((2, 2, 2, 2, 0, 2, 2),))

greatest result so far : 5.0

--------------------------

C:\Users\ob\Documents\workspace\python\gams_api_deneme

z = Amac:  level=5.0     args : ['0 0 0 1', '0 0 0 0 0 1 0 0 0 0', '0 0 0 1 0 0 0'] -----> 1.0

time : 0.469000101089

4  -) result : 5.0

orginal: l (((2, 2), (2, 0)), ((2, 2, 2), (2, 2, 0)), ((2, 2, 2, 0, 2, 2, 2),))

greatest result so far : 5.0

--------------------------

C:\Users\ob\Documents\workspace\python\gams_api_deneme

z = Amac:  level=5.0     args : ['0 0 0 1', '0 0 0 0 0 1 0 0 0 0', '0 0 1 0 0 0 0'] -----> 1.0

time : 0.546999931335

5  -) result : 5.0

orginal: l (((2, 2), (2, 0)), ((2, 2, 2), (2, 2, 0)), ((2, 2, 0, 2, 2, 2, 2),))

greatest result so far : 5.0

--------------------------

C:\Users\ob\Documents\workspace\python\gams_api_deneme
```

z = Amac: level=1000000000.0   args : ['0 0 0 1', '0 0 0 0 0 1 0 0 0 0', '0 1 0 0 0 0 0'] -----> 10.0

time : 0.43799996376

6 -) result : 1000000000.0

orginal: l (((2, 2), (2, 0)), ((2, 2, 2), (2, 2, 0)), ((2, 0, 2, 2, 2, 2, 2),))

greatest result so far : 5.0

C:\Users\ob\Documents\workspace\python\gams_api_deneme

z = Amac: level=12.0   args : ['0 0 0 1', '0 0 0 0 0 1 0 0 0 0', '1 0 0 0 0 0 0'] -----> 1.0

time : 0.748999834061

7 -) result : 12.0

orginal: l (((2, 2), (2, 0)), ((2, 2, 2), (2, 2, 0)), ((0, 2, 2, 2, 2, 2, 2),))

greatest result so far : 5.0

C:\Users\ob\Documents\workspace\python\gams_api_deneme

z = Amac: level=7.0   args : ['0 0 0 1', '0 0 0 1 0 0 0 0 0 0', '0 0 0 0 0 0 1'] -----> 1.0

time : 0.594000101089

8 -) result : 7.0

orginal: l (((2, 2), (2, 0)), ((2, 2, 2), (2, 0, 2)), ((2, 2, 2, 2, 2, 2, 0),))

greatest result so far : 5.0

C:\Users\ob\Documents\workspace\python\gams_api_deneme

z = Amac: level=5.0   args : ['0 0 0 1', '0 0 0 1 0 0 0 0 0 0', '0 0 0 0 0 1 0'] -----> 1.0

time : 0.59299993515

9 -) result : 5.0

orginal: l (((2, 2), (2, 0)), ((2, 2, 2), (2, 0, 2)), ((2, 2, 2, 2, 2, 0, 2),))

greatest result so far : 5.0

```
--------------------------

C:\Users\ob\Documents\workspace\python\gams_api_deneme

z = Amac:  level=5.0     args : ['0 0 0 1', '0 0 0 1 0 0 0 0 0 0', '0 0 0 0 1 0 0'] -----> 1.0

time : 0.59500002861

10  -) result : 5.0

orginal: l (((2, 2), (2, 0)), ((2, 2, 2), (2, 0, 2)), ((2, 2, 2, 2, 0, 2, 2),))

greatest result so far : 5.0

--------------------------

C:\Users\ob\Documents\workspace\python\gams_api_deneme

z = Amac:  level=5.0     args : ['0 0 0 1', '0 0 0 1 0 0 0 0 0 0', '0 0 0 1 0 0 0'] -----> 1.0

time : 0.496000051498

11  -) result : 5.0

orginal: l (((2, 2), (2, 0)), ((2, 2, 2), (2, 0, 2)), ((2, 2, 2, 0, 2, 2, 2),))

greatest result so far : 5.0

--------------------------

C:\Users\ob\Documents\workspace\python\gams_api_deneme

z = Amac:  level=5.0     args : ['0 0 0 1', '0 0 0 1 0 0 0 0 0 0', '0 0 1 0 0 0 0'] -----> 1.0

time : 0.720999956131

12  -) result : 5.0

orginal: l (((2, 2), (2, 0)), ((2, 2, 2), (2, 0, 2)), ((2, 2, 0, 2, 2, 2, 2),))

greatest result so far : 5.0

--------------------------

C:\Users\ob\Documents\workspace\python\gams_api_deneme

z = Amac:  level=1000000000.0     args : ['0 0 0 1', '0 0 0 1 0 0 0 0 0 0', '0 1 0 0 0 0 0'] -----> 10.0

time : 0.81200003624

13  -) result : 1000000000.0
```

```
orginal: l (((2, 2), (2, 0)), ((2, 2, 2), (2, 0, 2)), ((2, 0, 2, 2, 2, 2, 2),))

greatest result so far : 5.0

---------------------------

C:\Users\ob\Documents\workspace\python\gams_api_deneme

z = Amac:  level=12.0     args : ['0 0 0 1', '0 0 0 1 0 0 0 0 0 0', '1 0 0 0 0 0 0'] -----> 1.0

time : 0.50200009346

14  -) result : 12.0

orginal: l (((2, 2), (2, 0)), ((2, 2, 2), (2, 0, 2)), ((0, 2, 2, 2, 2, 2, 2),))

greatest result so far : 5.0

---------------------------

C:\Users\ob\Documents\workspace\python\gams_api_deneme

z = Amac:  level=7.0     args : ['0 0 0 1', '0 1 0 0 0 0 0 0 0 0', '0 0 0 0 0 0 1'] -----> 1.0

time : 0.419000148773

15  -) result : 7.0

orginal: l (((2, 2), (2, 0)), ((2, 2, 2), (0, 2, 2)), ((2, 2, 2, 2, 2, 2, 0),))

greatest result so far : 5.0

---------------------------

C:\Users\ob\Documents\workspace\python\gams_api_deneme

z = Amac:  level=5.0     args : ['0 0 0 1', '0 1 0 0 0 0 0 0 0 0', '0 0 0 0 0 1 0'] -----> 1.0

time : 0.406000137329

16  -) result : 5.0

orginal: l (((2, 2), (2, 0)), ((2, 2, 2), (0, 2, 2)), ((2, 2, 2, 2, 2, 0, 2),))

greatest result so far : 5.0

---------------------------

C:\Users\ob\Documents\workspace\python\gams_api_deneme

z = Amac:  level=5.0     args : ['0 0 0 1', '0 1 0 0 0 0 0 0 0 0', '0 0 0 0 1 0 0'] -----> 1.0
```

time : 0.608999967575

17 -) result : 5.0

orginal: l (((2, 2), (2, 0)), ((2, 2, 2), (0, 2, 2)), ((2, 2, 2, 2, 0, 2, 2),))

greatest result so far : 5.0

C:\Users\ob\Documents\workspace\python\gams_api_deneme

z = Amac: level=5.0 args : ['0 0 0 1', '0 1 0 0 0 0 0 0 0 0', '0 0 0 1 0 0 0'] -----> 1.0

time : 0.516000032425

18 -) result : 5.0

orginal: l (((2, 2), (2, 0)), ((2, 2, 2), (0, 2, 2)), ((2, 2, 2, 0, 2, 2, 2),))

greatest result so far : 5.0

C:\Users\ob\Documents\workspace\python\gams_api_deneme

z = Amac: level=5.0 args : ['0 0 0 1', '0 1 0 0 0 0 0 0 0 0', '0 0 1 0 0 0 0'] -----> 1.0

time : 0.641999959946

19 -) result : 5.0

orginal: l (((2, 2), (2, 0)), ((2, 2, 2), (0, 2, 2)), ((2, 2, 0, 2, 2, 2, 2),))

greatest result so far : 5.0

C:\Users\ob\Documents\workspace\python\gams_api_deneme

z = Amac: level=1000000000.0 args : ['0 0 0 1', '0 1 0 0 0 0 0 0 0 0', '0 1 0 0 0 0 0'] -----> 10.0

time : 0.561000108719

20 -) result : 1000000000.0

orginal: l (((2, 2), (2, 0)), ((2, 2, 2), (0, 2, 2)), ((2, 0, 2, 2, 2, 2, 2),))

greatest result so far : 5.

Ek Açıklama-E: GA Tabanlı Sezgisel ile Çözülen Örnek Problemin Parametreleri ve Çözüm Popülasyonları

GA tabanlı sezgisel algoritmanın çalıştırıldığı 8 örnek problemden aşağıda parametreleri, çözüm sonucu, başlangıç popülasyonu ve 5 iterasyonda ki koruma kombinasyonlarına kromozomlar verilmiştir.

Çizelge E.3. Örnek 2'ye Ait Problem Parametreleri

(Santral Sayısı, Saldırı/Koruma Tipi Sayısı)	(2,2)
Santral Koruma Kaynağı Kapasiteleri (Tip1,Tip2)	(1,1)
Santral Saldırı Kaynağı Kapasiteleri. (Tip1,Tip2)	(1,1)

(Trafo Merkezi. Sayısı, Saldırı/Koruma Tipi Sayısı)	(3,2)
Trafo Merkezi Koruma Kaynağı Kapasiteleri (Tip1,Tip2)	(2,1)
Trafo Merkezi Saldırı Kaynağı Kapasiteleri. (Tip1,Tip2)	(1,1)

(İletim Hattı Sayısı, Saldırı/Koruma Tipi Sayısı)	(7,1)
İletim Hattı Koruma Kaynağı Kapasiteleri (Tip1)	(4)
İletim Hattı Saldırı Kaynağı Kapasiteleri. (Tip1)	(1)

Çizelge E.4. Örnek 2 GA Çözüm Sonuçları:

Sezgisel Çözüm Süresi	Başlangıç Populasyonu Süresi	İterasyon Sayısı	Sezgisel Çözüm Sonucu
22' 32''	5' 49''	5	5.0

Üretilen Popülasyonlar:

initial set

1000000000.0 [[[0, 2], [0, 2]], [[2, 2, 0], [0, 0, 2]], [[0, 0, 2, 2, 0, 2, 2]]]

7.0 [[[2, 0], [2, 0]], [[0, 2, 2], [0, 2, 0]], [[2, 2, 0, 0, 2, 2, 0]]]

1000000000.0 [[[0, 2], [2, 0]], [[2, 0, 2], [2, 0, 0]], [[0, 2, 2, 2, 2, 0, 0]]]

1000000000.0 [[[2, 0], [2, 0]], [[2, 0, 2], [0, 2, 0]], [[2, 0, 2, 0, 2, 0, 2]]]

1000000000.0 [[[0, 2], [0, 2]], [[2, 0, 2], [0, 0, 2]], [[0, 0, 2, 2, 2, 0, 2]]]

1000000000.0 [[[0, 2], [0, 2]], [[2, 0, 2], [0, 2, 0]], [[0, 2, 0, 2, 2, 2, 0]]]

1000000000.0 [[[2, 0], [2, 0]], [[0, 2, 2], [0, 0, 2]], [[2, 0, 0, 2, 2, 2, 0]]]

1000000000.0 [[[0, 2], [0, 2]], [[2, 0, 2], [0, 0, 2]], [[2, 2, 0, 2, 0, 2, 0]]]

7.0 [[[2, 0], [2, 0]], [[2, 0, 2], [2, 0, 0]], [[2, 2, 0, 2, 0, 2, 0]]]

1000000000.0 [[[0, 2], [0, 2]], [[2, 0, 2], [0, 0, 2]], [[2, 2, 2, 0, 2, 0, 0]]]

7.0 [[[2, 0], [2, 0]], [[2, 2, 0], [0, 2, 0]], [[2, 2, 2, 2, 0, 0, 0]]]

1000000000.0 [[[0, 2], [0, 2]], [[2, 0, 2], [0, 2, 0]], [[0, 2, 2, 2, 2, 0, 0]]]

1000000000.0 [[[2, 0], [0, 2]], [[2, 2, 0], [0, 2, 0]], [[2, 2, 0, 2, 2, 0, 0]]]

12.0 [[[2, 0], [2, 0]], [[2, 2, 0], [0, 0, 2]], [[0, 2, 2, 2, 2, 0, 0]]]

1000000000.0 [[[2, 0], [0, 2]], [[2, 2, 0], [0, 0, 2]], [[0, 0, 2, 2, 2, 0, 2]]]

7.0 [[[2, 0], [2, 0]], [[2, 2, 0], [2, 0, 0]], [[2, 2, 2, 0, 2, 0, 0]]]

12.0 [[[2, 0], [2, 0]], [[0, 2, 2], [0, 2, 0]], [[0, 2, 2, 0, 2, 0, 2]]]

1000000000.0 [[[0, 2], [2, 0]], [[0, 2, 2], [0, 2, 0]], [[0, 2, 2, 2, 2, 0, 0]]]

1000000000.0 [[[0, 2], [2, 0]], [[2, 2, 0], [0, 0, 2]], [[2, 0, 2, 0, 0, 2, 2]]]

7.0 [[[2, 0], [2, 0]], [[2, 2, 0], [0, 0, 2]], [[2, 2, 0, 2, 0, 2, 0]]]

iteration 1

5.0 [[[0, 2], [2, 0]], [[2, 2, 0], [2, 0, 0]], [[2, 2, 0, 0, 0, 2, 2]]]

7.0 [[[2, 0], [2, 0]], [[0, 2, 2], [0, 2, 0]], [[2, 2, 0, 0, 2, 2, 0]]]

7.0 [[[2, 0], [2, 0]], [[0, 2, 2], [0, 2, 0]], [[2, 2, 2, 0, 2, 0, 0]]]

7.0 [[[2, 0], [2, 0]], [[2, 0, 2], [2, 0, 0]], [[2, 2, 0, 2, 0, 2, 0]]]

7.0 [[[2, 0], [2, 0]], [[2, 2, 0], [0, 2, 0]], [[2, 2, 2, 2, 0, 0, 0]]]

7.0 [[[2, 0], [2, 0]], [[2, 2, 0], [2, 0, 0]], [[2, 2, 2, 0, 2, 0, 0]]]

12.0 [[[2, 0], [2, 0]], [[0, 2, 2], [0, 2, 0]], [[0, 2, 2, 0, 2, 0, 2]]]

1000000000.0 [[[0, 2], [2, 0]], [[0, 2, 2], [0, 2, 0]], [[0, 2, 2, 2, 2, 0, 0]]]

12.0 [[[2, 0], [2, 0]], [[0, 2, 2], [0, 2, 0]], [[0, 2, 2, 2, 2, 0, 0]]]

5.0 [[[0, 2], [2, 0]], [[2, 2, 0], [2, 0, 0]], [[2, 2, 0, 0, 0, 2, 2]]]

12.0 [[[2, 0], [2, 0]], [[0, 2, 2], [0, 2, 0]], [[0, 2, 2, 0, 0, 2, 2]]]

1000000000.0 [[[0, 2], [0, 2]], [[2, 2, 0], [0, 2, 0]], [[2, 0, 0, 2, 0, 2, 2]]]

12.0 [[[2, 0], [2, 0]], [[2, 0, 2], [0, 2, 0]], [[0, 2, 2, 0, 2, 0, 2]]]

1000000000.0 [[[2, 0], [0, 2]], [[2, 0, 2], [0, 0, 2]], [[2, 2, 0, 0, 0, 2, 2]]]

1000000000.0 [[[0, 2], [2, 0]], [[2, 0, 2], [2, 0, 0]], [[0, 0, 2, 0, 2, 2, 2]]]

1000000000.0 [[[0, 2], [2, 0]], [[0, 2, 2], [0, 0, 2]], [[0, 0, 2, 2, 2, 0, 2]]]

1000000000.0 [[[0, 2], [0, 2]], [[2, 2, 0], [0, 2, 0]], [[2, 0, 0, 2, 0, 2, 2]]]

1000000000.0 [[[2, 0], [2, 0]], [[2, 2, 0], [2, 0, 0]], [[2, 0, 2, 0, 0, 2, 2]]]

1000000000.0 [[[0, 2], [2, 0]], [[0, 2, 2], [0, 2, 0]], [[0, 2, 2, 2, 2, 0, 0]]]

1000000000.0 [[[0, 2], [0, 2]], [[2, 2, 0], [0, 2, 0]], [[2, 2, 0, 2, 0, 2, 0]]]

iteration 2

5.0 [[[2, 0], [2, 0]], [[2, 2, 0], [2, 0, 0]], [[2, 2, 0, 0, 2, 0, 2]]]

5.0 [[[0, 2], [2, 0]], [[2, 2, 0], [2, 0, 0]], [[2, 2, 0, 0, 0, 2, 2]]]

7.0 [[[2, 0], [2, 0]], [[0, 2, 2], [0, 2, 0]], [[2, 2, 2, 0, 2, 0, 0]]]

7.0 [[[2, 0], [2, 0]], [[2, 0, 2], [2, 0, 0]], [[2, 2, 0, 2, 0, 2, 0]]]

7.0 [[[2, 0], [2, 0]], [[2, 2, 0], [0, 2, 0]], [[2, 2, 2, 2, 0, 0, 0]]]

7.0 [[[2, 0], [2, 0]], [[2, 2, 0], [2, 0, 0]], [[2, 2, 2, 0, 2, 0, 0]]]

1000000000.0 [[[2, 0], [0, 2]], [[2, 2, 0], [2, 0, 0]], [[2, 0, 2, 0, 0, 2, 2]]]

1000000000.0 [[[0, 2], [0, 2]], [[2, 2, 0], [0, 2, 0]], [[0, 2, 0, 0, 2, 2, 2]]]

7.0 [[[2, 0], [2, 0]], [[2, 2, 0], [0, 0, 2]], [[2, 2, 2, 2, 0, 0, 0]]]

12.0 [[[2, 0], [2, 0]], [[2, 2, 0], [0, 0, 2]], [[0, 2, 2, 0, 2, 0, 2]]]

1000000000.0 [[[0, 2], [2, 0]], [[2, 2, 0], [0, 0, 2]], [[0, 2, 2, 0, 2, 0, 2]]]

5.0 [[[2, 0], [2, 0]], [[2, 0, 2], [0, 2, 0]], [[2, 2, 0, 0, 2, 0, 2]]]

7.0 [[[2, 0], [2, 0]], [[0, 2, 2], [0, 2, 0]], [[2, 2, 2, 0, 2, 0, 0]]]

12.0 [[[2, 0], [2, 0]], [[2, 2, 0], [0, 0, 2]], [[0, 2, 2, 2, 2, 0, 0]]]

12.0 [[[2, 0], [2, 0]], [[2, 2, 0], [0, 2, 0]], [[0, 2, 2, 2, 2, 0, 0]]]

1000000000.0 [[[2, 0], [0, 2]], [[0, 2, 2], [2, 0, 0]], [[2, 0, 2, 0, 0, 2, 2]]]

1000000000.0 [[[0, 2], [2, 0]], [[2, 0, 2], [0, 0, 2]], [[2, 0, 2, 0, 0, 2, 2]]]

1000000000.0 [[[2, 0], [0, 2]], [[2, 2, 0], [0, 0, 2]], [[2, 0, 2, 0, 2, 0, 2]]]

12.0 [[[2, 0], [2, 0]], [[2, 0, 2], [0, 2, 0]], [[0, 2, 0, 0, 2, 2, 2]]]

1000000000.0 [[[2, 0], [2, 0]], [[2, 0, 2], [2, 0, 0]], [[0, 0, 2, 2, 2, 2, 0]]]

1000000000.0 [[[0, 2], [2, 0]], [[2, 2, 0], [0, 0, 2]], [[0, 2, 0, 0, 2, 2, 2]]]

5.0 [[[2, 0], [2, 0]], [[2, 0, 2], [0, 2, 0]], [[2, 2, 0, 0, 2, 0, 2]]]

1000000000.0 [[[2, 0], [0, 2]], [[2, 0, 2], [2, 0, 0]], [[2, 2, 2, 0, 0, 2, 0]]]

iteration 3

5.0 [[[2, 0], [2, 0]], [[2, 2, 0], [2, 0, 0]], [[2, 2, 0, 0, 2, 0, 2]]]

5.0 [[[0, 2], [2, 0]], [[2, 2, 0], [2, 0, 0]], [[2, 2, 0, 0, 0, 2, 2]]]

5.0 [[[2, 0], [2, 0]], [[2, 0, 2], [0, 2, 0]], [[2, 2, 0, 0, 2, 0, 2]]]

5.0 [[[2, 0], [2, 0]], [[2, 0, 2], [0, 2, 0]], [[2, 2, 0, 0, 2, 0, 2]]]

7.0 [[[2, 0], [2, 0]], [[0, 2, 2], [0, 2, 0]], [[2, 2, 2, 0, 2, 0, 0]]]

7.0 [[[2, 0], [2, 0]], [[2, 2, 0], [0, 2, 0]], [[2, 2, 2, 2, 0, 0, 0]]]

1000000000.0 [[[2, 0], [0, 2]], [[2, 2, 0], [0, 2, 0]], [[2, 2, 0, 2, 2, 0, 0]]]

1000000000.0 [[[0, 2], [2, 0]], [[2, 2, 0], [0, 0, 2]], [[2, 0, 2, 0, 0, 2, 2]]]

12.0 [[[2, 0], [2, 0]], [[2, 0, 2], [0, 2, 0]], [[0, 2, 2, 0, 2, 0, 2]]]

1000000000.0 [[[0, 2], [2, 0]], [[2, 2, 0], [0, 0, 2]], [[2, 2, 2, 0, 2, 0, 0]]]

12.0 [[[2, 0], [2, 0]], [[2, 2, 0], [0, 0, 2]], [[0, 2, 2, 0, 2, 0, 2]]]

1000000000.0 [[[0, 2], [2, 0]], [[2, 2, 0], [0, 0, 2]], [[0, 2, 2, 0, 2, 0, 2]]]

1000000000.0 [[[0, 2], [2, 0]], [[2, 0, 2], [2, 0, 0]], [[2, 2, 0, 2, 0, 2, 0]]]

1000000000.0 [[[2, 0], [2, 0]], [[2, 0, 2], [0, 2, 0]], [[0, 0, 2, 2, 2, 2, 0]]]

1000000000.0 [[[2, 0], [0, 2]], [[0, 2, 2], [0, 2, 0]], [[2, 2, 2, 0, 2, 0, 0]]]

12.0 [[[2, 0], [2, 0]], [[2, 0, 2], [0, 0, 2]], [[0, 2, 2, 0, 2, 0, 2]]]

1000000000.0 [[[2, 0], [2, 0]], [[2, 0, 2], [0, 0, 2]], [[2, 0, 0, 2, 2, 0, 2]]]

12.0 [[[2, 0], [2, 0]], [[2, 0, 2], [0, 2, 0]], [[0, 2, 2, 2, 2, 0, 0]]]

12.0 [[[2, 0], [2, 0]], [[2, 0, 2], [0, 0, 2]], [[0, 2, 2, 0, 2, 2, 0]]]

1000000000.0 [[[2, 0], [0, 2]], [[2, 2, 0], [2, 0, 0]], [[2, 2, 0, 2, 0, 2, 0]]]

iteration 4

5.0 [[[2, 0], [2, 0]], [[2, 2, 0], [2, 0, 0]], [[2, 2, 0, 0, 2, 0, 2]]]

5.0 [[[0, 2], [2, 0]], [[2, 2, 0], [2, 0, 0]], [[2, 2, 0, 0, 0, 2, 2]]]

5.0 [[[2, 0], [2, 0]], [[2, 0, 2], [0, 2, 0]], [[2, 2, 0, 0, 2, 0, 2]]]

5.0 [[[2, 0], [2, 0]], [[2, 0, 2], [0, 2, 0]], [[2, 2, 0, 0, 2, 0, 2]]]

7.0 [[[2, 0], [2, 0]], [[0, 2, 2], [0, 2, 0]], [[2, 2, 2, 0, 2, 0, 0]]]

7.0 [[[2, 0], [2, 0]], [[2, 2, 0], [0, 2, 0]], [[2, 2, 2, 2, 0, 0, 0]]]

1000000000.0 [[[2, 0], [2, 0]], [[2, 2, 0], [0, 2, 0]], [[2, 0, 0, 2, 0, 2, 2]]]

1000000000.0 [[[2, 0], [0, 2]], [[0, 2, 2], [0, 0, 2]], [[0, 2, 2, 2, 2, 0, 0]]]

1000000000.0 [[[0, 2], [0, 2]], [[2, 0, 2], [0, 2, 0]], [[0, 2, 0, 2, 2, 2, 0]]]

1000000000.0 [[[0, 2], [2, 0]], [[0, 2, 2], [0, 2, 0]], [[2, 2, 0, 0, 2, 2, 0]]]

5.0 [[[0, 2], [2, 0]], [[2, 2, 0], [2, 0, 0]], [[2, 2, 0, 0, 0, 2, 2]]]

1000000000.0 [[[0, 2], [2, 0]], [[2, 2, 0], [0, 0, 2]], [[2, 2, 2, 0, 2, 0, 0]]]

1000000000.0 [[[2, 0], [0, 2]], [[2, 0, 2], [2, 0, 0]], [[0, 0, 2, 2, 2, 0, 2]]]

1000000000.0 [[[0, 2], [2, 0]], [[2, 0, 2], [0, 0, 2]], [[0, 2, 2, 0, 2, 2, 0]]]

1000000000.0 [[[2, 0], [0, 2]], [[2, 2, 0], [0, 2, 0]], [[2, 2, 0, 2, 2, 0, 0]]]

12.0 [[[2, 0], [2, 0]], [[0, 2, 2], [0, 0, 2]], [[0, 2, 2, 0, 2, 0, 2]]]

12.0 [[[2, 0], [2, 0]], [[2, 2, 0], [0, 0, 2]], [[0, 2, 0, 2, 2, 2, 0]]]

5.0 [[[2, 0], [2, 0]], [[2, 0, 2], [0, 2, 0]], [[2, 2, 0, 0, 2, 0, 2]]]

1000000000.0 [[[0, 2], [0, 2]], [[2, 0, 2], [0, 2, 0]], [[2, 2, 2, 0, 0, 0, 2]]]

7.0 [[[2, 0], [2, 0]], [[2, 2, 0], [2, 0, 0]], [[2, 2, 0, 2, 2, 0, 0]]]

iteration 5

5.0 [[[2, 0], [2, 0]], [[2, 2, 0], [2, 0, 0]], [[2, 2, 0, 0, 2, 0, 2]]]

5.0 [[[0, 2], [2, 0]], [[2, 2, 0], [2, 0, 0]], [[2, 2, 0, 0, 0, 2, 2]]]

5.0 [[[2, 0], [2, 0]], [[2, 2, 0], [2, 0, 0]], [[2, 2, 0, 0, 2, 0, 2]]]

5.0 [[[2, 0], [2, 0]], [[2, 0, 2], [0, 2, 0]], [[2, 2, 0, 0, 2, 0, 2]]]

5.0 [[[0, 2], [2, 0]], [[2, 2, 0], [2, 0, 0]], [[2, 2, 0, 0, 0, 2, 2]]]

5.0 [[[2, 0], [2, 0]], [[2, 0, 2], [0, 2, 0]], [[2, 2, 0, 0, 2, 0, 2]]]

1000000000.0 [[[0, 2], [2, 0]], [[0, 2, 2], [2, 0, 0]], [[2, 2, 2, 0, 2, 0, 0]]]

12.0 [[[2, 0], [2, 0]], [[2, 0, 2], [0, 2, 0]], [[0, 2, 2, 0, 2, 0, 2]]]

1000000000.0 [[[0, 2], [2, 0]], [[0, 2, 2], [0, 2, 0]], [[0, 0, 2, 0, 2, 2, 2]]]

1000000000.0 [[[0, 2], [2, 0]], [[0, 2, 2], [0, 2, 0]], [[2, 2, 0, 0, 2, 0, 2]]]

1000000000.0 [[[2, 0], [0, 2]], [[0, 2, 2], [0, 2, 0]], [[2, 2, 0, 2, 0, 2, 0]]]

12.0 [[[2, 0], [2, 0]], [[2, 0, 2], [0, 2, 0]], [[0, 2, 2, 2, 2, 0, 0]]]

1000000000.0 [[[0, 2], [0, 2]], [[2, 0, 2], [0, 0, 2]], [[2, 0, 2, 2, 0, 0, 2]]]

12.0 [[[2, 0], [2, 0]], [[2, 2, 0], [0, 2, 0]], [[0, 2, 2, 0, 2, 2, 0]]]

1000000000.0 [[[0, 2], [0, 2]], [[2, 2, 0], [2, 0, 0]], [[0, 2, 0, 0, 2, 2, 2]]]

1000000000.0 [[[2, 0], [0, 2]], [[2, 0, 2], [2, 0, 0]], [[2, 0, 2, 2, 0, 0, 2]]]

1000000000.0 [[[0, 2], [2, 0]], [[0, 2, 2], [0, 0, 2]], [[2, 0, 0, 2, 0, 2, 2]]]

12.0 [[[2, 0], [2, 0]], [[2, 0, 2], [0, 2, 0]], [[0, 2, 0, 2, 0, 2, 2]]]

1000000000.0 [[[2, 0], [2, 0]], [[2, 2, 0], [2, 0, 0]], [[2, 0, 0, 2, 0, 2, 2]]]

12.0 [[[2, 0], [2, 0]], [[2, 2, 0], [0, 0, 2]], [[0, 2, 0, 2, 2, 2, 0]]]

Ek Açıklama-F: r_GA ve k_GA Algoritmalarının Non Parametrik Testler ile Performans Karşılaştırması Analiz Sonuçları

Kuruskal-Wallis Testi ile sıkı-orta-gevşek koruma kategorilerinin farklılığının analizi minitab test sonuçları (k_GA algoritması test sonuçları ile)

Descriptive Statistics

sıkı orta gev	N	Median	Mean Rank	Z-Value
1	4	0,0000	2,5	-2,72
3	4	1,7725	7,5	0,68
5	4	5,4225	9,5	2,04
Overall	12		6,5	

Test

Null hypothesis H_0: All medians are equal

Alternative hypothesis H_1: At least one median is different

Method	DF	H-Value	P-Value
Not adjusted for ties	2	8,00	0,018
Adjusted for ties	2	8,32	0,016

The chi-square approximation may not be accurate when some sample sizes are less than 5.

Kuruskal-Wallis Testi ile sıkı-orta-gevşek koruma kategorilerinin farklılığının analizi minitab test sonuçları (r_GA algoritması test sonuçları ile)

Descriptive Statistics

sıkıorta gev	N	Median	Mean Rank	Z-Value
2	4	0,7225	2,5	-2,72
4	4	4,7725	7,3	0,51
6	4	5,6950	9,8	2,21
Overall	12		6,5	

Test

Null hypothesis	H_0: All medians are equal
Alternative hypothesis	H_1: At least one median is different

DF	H-Value	P-Value
2	8,35	0,015

The chi-square approximation may not be accurate when some sample sizes are less than 5.

Sıkı Koruma Kategorisi için r_GA ve k_GA algoritmalarının performans farklılıklarının Wilcoxon İşaretli Sıralar Testi ile analizi Minitab test sonuçları

Descriptive Statistics

Sample	N	Median
Fark_sıkı	8	-0,59

Method

η: median of Fark_sıkı

Test

Null hypothesis	H_0: $\eta = 0$
Alternative hypothesis	H_1: $\eta \neq 0$

Sample	N for Test	Wilcoxon Statistic	P-Value
Fark_sıkı	8	0,00	0,014

Orta Koruma Kategorisi için r_GA ve k_GA algoritmalarının performans farklılıklarının Wilcoxon İşaretli Sıralar Testi ile analizi Minitab test sonuçları

Method

η: median of Fark_orta

Descriptive Statistics

Sample	N	Median
Fark_orta	8	-2,205

Test

Null hypothesis $H_0: \eta = 0$
Alternative hypothesis $H_1: \eta \neq 0$

Sample	N for Test	Wilcoxon Statistic	P-Value
Fark_orta	8	0,00	0,014

Gevşek Koruma Kategorisi için r_GA ve k_GA algoritmalarının performans farklılıklarının Wilcoxon İşaretli Sıralar Testi ile analizi Minitab test sonuçları

Method

η: median of Fark_Gev

Descriptive Statistics

Sample	N	Median
Fark_Gev	8	-0,69125

Test

Null hypothesis $H_0: \eta = 0$
Alternative hypothesis $H_1: \eta \neq 0$

Sample	N for Test	Wilcoxon Statistic	P-Value
Fark_Gev	7	7,00	0,272

ÖZGEÇMİŞ

İLETİŞİM BİLGİLERİ

E-MAİL : orkunbaskan@gmail.com, orkunbaskan@anadolu.edu.tr

LINKEDIN: https://tr.linkedin.com/in/orkun-baskan-775553a

TEL : 0532 786 30 93 – 0222 310 12 80

ADRES : Ertuğrulgazi Mah. Kutluk sok. No:18 D:2 ESKİŞEHİR

DOKTORA **Osmangazi Üniversitesi ENDÜSTRİ MÜHENDİSLİĞİ (2013 - devam)**

YÜKSEK LİSANS **Dokuz Eylül Üniversitesi ENDÜSTRİ MÜHENDİSLİĞİ (2005 - 2008)**

LİSANS **Osmangazi Üniversitesi ENDÜSTRİ MÜHENDİSİ (2000 - 2004)**

YABANCI DİL İngilizce (Dokuz Eylül Üniversitesi_Hazırlık)

BİLGİSAYAR MS Word, Excel, PowerPoint; MS Visio, MS Project, GAMS, Lingo, Matlab, Minitab QS gibi paket programları.

Python, Visual Basic gibi yazılım programları

TECRÜBELER :

01.2015 – DEVAM ANADOLU ÜNİV., ESKİŞEHİR

TTO Yönetici Yardımcısı ve Teknoloji Transferi Grup Yöneticisi görevi ile Anadolu Üniversitesi ve ESTÜ TTO'su ARİNKOM Teknoloji Transfer Ofisinde görev yapmaktadır. Bu kapsamda üniversite sektör işbirlikleri, fikri haklar, ticarileştirme, girişimcilik şirketleşme ve sermaye alanlarında yapılan çalışmalara ve çalışma ekiplerine liderlik etmektedir.

Aynı zamanda **RTTP** (Registered Technology Transfer Professional) ünvanına sahiptir.

05.2017 - DEVAM ESKİŞEHİR TEKNİK ÜNİV., ESKİŞEHİR

Endüstri Mühendisliği Bölümünde **Öğretim Görevlisi** kadrosunda yer almaktadır.

02.2012 – 01.2015 ANADOLU ÜNİV., ESKİŞEHİR

Bilgi ve İletişim Yöneticisi görevi ile Anadolu Üniversitesi ARİNKOM Teknoloji Transfer Ofisinde farkındalığın ve tanınırlığının sağlanması veri yönetim sisteminin kurulması ve yönetilmesi amacıyla görev yaptı.

08.2009 – 02.2012 MATASAN A.Ş., ESKİŞEHİR

Kalite Müdürü görevi ile Kalite Sisteminin yönetimi, kendisine bağlı kalite kontrol ekibinin yönetimi, Giriş kalite, Proses kalite ve sevk kaliteden sorumlu olarak çalıştı, kalite projelerini yürüttü.

10.2007 – 12.2008 ELKİMA TRAFO, İZMİR

Endüstri Mühendisi görevi ile verimliliğin arttırılması, strateji geliştirme ve iş geliştirme alanlarında çeşitli çalışmalar gerçekleştirdi.

09.2006 – 12.2008 ESDİL EĞİTİM KURUMLARI, İZMİR

Üniv. öğrencilerine ve mezunlarına Matematik ve İstatistik derslerini verdi.

06.2004 – 10.2004 YÖNTEK DANIŞMANLIK, ESKİŞEHİR

Danışmanlık görevi ile ISO 9001:2000 konusunda özel şirketlere danışmanlık hizmeti verdi.

01.2004 – 06.2004 YÖRÜKOĞLU MAKİNE, ESKİŞEHİR

Dokümantasyon sorumlusu görevi ile işletmenin ISO 9001:2000 çalışmalarında yer aldı.

PROJELER

TechUP Teknoloji Odaklı Hızlandırma Programı (22.11.2016-devam), http://techup.bebka.org.tr/, Program Lideri.

Uluslararası Rekabet İçin Ar-Ge ve Yeniliğin Korunması, Ticarileştirilmesi projesi (TR41/15/TD/0059) (11.2015-12.2015), BEBKA (Bursa Eskişehir Bilecik Kalkınma Ajansı) Teknik Destek Programı, Proje Yürütücüsü.

Anadolu Üniversitesi ARİNKOM Teknoloji Transferi Ofisi Projesi, (01.01.2014- 31.12.2023) TÜBİTAK 1513 - Teknoloji Transfer Ofisleri Destekleme Programı, Proje yazım ekibinde yer alınmıştır, Proje Çalışanı

ANP (Analytic Network Process) and To Estimate Market Share in Transformer Industry with Using ANP. (Eylül 2008, Yüksek Lisans)

Tesis Yerleşimi (Tez Konusu çerçevesinde bir işletme de en iyi tesis yerleşimi tespit edilmiştir.) (Haziran 2004, Bitirme Tezi);

YAYINLAR

Başkan O., Dağ M.H., 2020, "Avrupa'da Teknoloji Transfer Ofisleri", Kitap Bölümü, "Teknoloji Transfer Ofisleri", ARİNKOM TTO Yayınları.

Başkan O., Sevik O., Tigin Ö., Dağ M.H., 2019, "Yükseköğretim Kurumlarında, Patentlerin Lisanslanmasından Elde Edilen Gelirin Paylaşımında Karşılaşılan Zorluklar ve Belirsizlikler", Bildiri, Ulusal Patent Fuarı ve Kongresi.

Başkan O., Sağır M., 2017, "Savunma Sistemlerine Ait Problemlerin Çözümlerinde, Endüstri Mühendisliği ve Yöneylem Araştırması Yaklaşımları", Bildiri, 2. Uluslararası Savunma Sanayi Sempozyumu

Taş R., Başkan O., Tigin Ö., Timurçin B., 2017, "Turkish TTO Good Practices – TTO Operation Models (Anadolu University IPR Policy – Use of TRL Methodogy at Commercialization Process), Kitap Bölümü, "Technology Transfer Book of Knowledge with Turkish TTO Good Practices", TTGV, 273-286.

EĞİTİMLER

Şirket Değerleme, (15-17 Ekim 2019, Deloitte Academy)

Kuluçka Yönetici Programı, (16-22 Aralık 2015, T-Jump, ABD, San Francisco);

UTTP Uygulamalı Teknoloji Ticarileştirme Programı Eğitimi (16 Eylül 2015 – 15 Aralık 2015, TTGV, Cyberpark ve the University of Texas at Austin)

Eczacılık Alanında Patent Süreçleri, (7 Mart 2019, Yalçıner Patent)

TRİZ (Theory of Inventive Problem Solving) Eğtimi, (16-19 Kasım 2016, Kaykayoğlu Inovation Group)

Fikri Mülkiyetin Koruması ve Lisanslama Yoluyla Teknolojinin Ticarileştirilmesi (LES 100), (02 Eylül 2015, TTGV, LES Turkey)

Proje Yönetimi Eğitimi (13-17 Mayıs 2013, FABE Eğitim)

Teknoloji Transferi Eğitmenlerin Eğitimi (24-26 Mayıs 2012 AUTM-EBİLTEM)

Solidworks Katı modelleme (08-19 Ağustos 2011)

KİŞİSEL BİLGİLER

DOĞUM TARİHİ : 05.11.1982 (Antalya)

MEDENİ DURUM : Evli

ASKERLİK : Tamamladı

SÜRÜCÜ BELGESİ : B sınıfı

İLGİ ALANLARI : Yüzme, Kitap Okumak, Bisiklete Binmek