

**T.C.
SAKARYA ÜNİVERSİTESİ
FEN EDEBİYAT FAKÜLTESİ**

**BİLGİSAYAR AĞ TOPOLOJİLERİYLE ŞEHİR
PLANLAMA YÖNTEMLERİNİN OPTİMİZASYONU**

BİTİRME ÇALIŞMASI

Volkan DİLSİZ

1102.00052

Fakülte Bölümü : MATEMATİK

Bölüm Dersi : TOPOLOJİ

Danışman : Doç. Dr. Soley ERSOY

MAYIS 2017

T.C.
SAKARYA ÜNİVERSİTESİ
FEN EDEBİYAT FAKÜLTESİ

BİLGİSAYAR AĞ TOPOLOJİLERİYLE ŞEHİR PLANLAMA YÖNTEMLERİNİN OPTİMİZASYONU

BİTİRME ÇALIŞMASI

Volkan DİLSİZ

1102.00052

Fakülte Bölümü : MATEMATİK

Bölüm Dersi : TOPOLOJİ

Danışman : Doç. Dr. Soley ERSOY

Bu tez .././.... tarihinde aşağıdaki jüri tarafından oybirliği ile kabul edilmiştir.

.....
Başkan

.....
Üye

.....
Üye

BEYAN

Tez içindeki tüm verilerin akademik kurallar çerçevesinde tarafımdan elde edildiğini, görsel ve yazılı tüm bilgi ve sonuçların akademik ve etik kurallara uygun şekilde sunulduğunu, kullanılan verilerde herhangi bir tahrifat yapılmadığını, başkalarının eserlerinden yararlanılması durumunda bilimsel normlara uygun olarak atıfta bulunulduğunu beyan ederim.

Volkan DİLSİZ

TEŞEKKÜR

Tez danışmanlığımı üstlenip, bilgi ve tecrübesiyle destek veren, çalışmamın her safhasında yardımını esirgemeyen sayın hocam Doç. Dr. Soley ERSOY'a şükran ve saygılarımı sunarım.

Hayatımın her aşamasında yanımda olan, her türlü desteği kayıtsız şartsız önüme seren, en büyük hazinem, biricik ailem; annem Ayşe DİLSİZ, babam Oktay DİLSİZ, kardeşim Orhan Can DİLSİZ'e teşekkürü bir borç bilirim. Lisans hayatım boyunca her türlü yardımından faydalandığım dostum Samet GÜNEŞ'e teşekkür ederim. Kariyerimi planlama aşamalarımda her zaman yanımda olup bana koşulsuz destek olan sevgili arkadaşım Ezgi BİRDAL'a teşekkür ederim.

İÇİNDEKİLER

BEYAN	i
TEŞEKKÜR.....	ii
İÇİNDEKİLER	iii
ÖZET.....	iv
SUMMARY	v
BÖLÜM 1.	
GİRİŞ	1
BÖLÜM 2.	
BİLGİSAYAR AĞLARININ TOPOLOJİLERİ.....	5
BÖLÜM 3.	
ALGORİTMALAR VE OPTİMİZASYON YÖNTEMLERİ	17
BÖLÜM 4.	
ŞEHİR YAPILANMALARININ TOPOLOJİK ANALİZİ	48
BÖLÜM 5.	
SONUÇLAR VE ÖNERİLER	59
KAYNAKLAR	61
ÖZGEÇMİŞ	67

ÖZET

21. yüzyılın en büyük sorunlarından biri olan çarpık kentleşme ve nüfus kalabalığının getirdiği her türden trafik sıkışıklığının insan hayatı üzerindeki olumsuz etkilerinin, uygun bir şehir yapılanması ile çözülebileceğine dair ortaya konmuş matematiksel önerileri derleyen bu çalışmada öncelikle bilgisayar ağ topolojilerinin temel özelliklerine dair detaylı bilgilendirmeler yer almaktadır. Sonrasında bu yapıların iyileştirilmesi için öngörülen modellemeler ve yöntemler sıralanmaktadır. Son olarak bir şehir yapılanmasının topolojik olarak da incelenebileceği ve bu yapıların uygun yol ve yöntemlerle optimize edilebileceği ifade edilmektedir. Sonuç olarak bu çalışmada matematik kullanılarak daha yaşanılır şehirler kurulabilir mi sorusuna bir cevap aranmaktadır.

SUMMARY

One of the biggest problems is crooked urbanization of 21st century. And overpopulated cities bring us every kind of traffic jam. This study presents some mathematical suggestions to solve these kinds of problems in people's life. This study contains deeply information about basic concepts of computer network topologies. Then it contains some methods and models for optimizations of these constructions. And finally it says city planning is a topological subject. Therefore it suggests any city planning method can be optimized like a computer network with proper methods and models. As a consequence in this study the answers are looked for the question: "Is it possible to build more livable cities using Mathematics?"

1. GİRİŞ

Bilim tarihinin başından bu yana üretilen ve geliştirilen her teknik ve teknoloji, bilgisayarın icadı ile birlikte erişilen yüksek hızda büyük boyutlu işlem kapasitesi sayesinde hiç olmadığı kadar yüksek bir üretim ve gelişim hızına ulaştı. Her geçen gün bir öncekine göre daha büyük ivme ile artan üretim ve gelişim hızı; bilgisayar ağları ve internetin icadı ile küresel boyutta binlerce bilim insanının aynı anda gerçek zamanlı çalışarak ürettiği ve geliştirdiği bir bilim ortamı yaratmayı başardı.

Ulaşılan bu geniş çaplı işlem yapma ve veri işleme imkânı ne kadar büyük bir alanda sağlanıyorsa değişkenlerin daha kesin sonuçlarla belirlenmesi ve yapılan işlemlerle elde edilen sonuçların hassaslığı o kadar fazla önem kazanır. Yay marifetiyle fırlatılan bir okun çıkış noktasında yapılacak bir santimetrelik değişikliğin okun düştüğü yerde on metrelerce farklılık yaratacağı gibi pek çok alanda bilgisayarlar ve bunları birbirine bağlayan bilgisayar ağları marifetiyle yapılan işlemlerin hassasiyeti ağların en başta kurulması aşamasındaki planlamalar ve hesaplamalara bağlıdır.

Nispeten yeni bir bilim dalı olan Optimizasyon ise tam olarak bu konuyla ilgilenir. Bir hesaplamada yapılacak olan herhangi bir değişikliğin sonucunun iyi veya kötü, büyük veya küçük, yüksek veya alçak olması durumunu inceleyerek en iyi çıktıyı alabilmek için işlem girdilerinin nasıl olması gerektiği konusunda fikir verir. Bütün bunların ışığında Optimizasyon, bilgisayar bilimlerinin ortaya çıkmasıyla erişilen bütün bu teknolojik imkânların en iyi, en uygun, en doğru, en hızlı ve en düşük maliyet çıkartacak şekilde kullanılmasını sağlayan bir kılavuздur denebilir.

Optimizasyon, doğanın en temel kanunlarından insanoğlunun üretebildiği son teknolojiye kadar her alanda kullanıma sahip olan bir bilim dalıdır. Matematiksel olarak hesaplanan en büyük ve en küçük değerleri bu alanlardaki değişkenlere uygulayarak en iyi sonuçları almayı hedefler.

Bu sonuçlar en yüksek verim, en yüksek hız, en yüksek güvenlik, en kısa zaman, en düşük işgücü, en düşük maliyet olabilir. Sonuç olarak devasa projelerde değişkenlerin optimize edilmesi ve en uygun sonucun alınmaya çalışılması hayati bir öneme sahiptir. Birbirine bağlı bilgisayarların yukarıda bahsedilen değişkenler göz önüne alındığında en uygun ortamda çalışmalarını sağlamak için Optimizasyon biliminden faydalanılması gerekir. Ancak söz konusu bir ağ tasarımı olduğu için kabaca topolojik uzayı tanımlamak için inşa edilen ve belli koşulları sağlayan kümeler ailesi için kullanılan Topoloji'den faydalanılmaması mümkün değildir. Zira her türlü ağın kurulumu bir uzay şeklinde incelenebileceğinden en uygun sonuçları alabilmek adına ağları bir bütün olarak incelersek; Topoloji bize bu uzayın içindeki elemanları nasıl sıralamamız ve yerleştirmemiz gerektiği konusunda en iyi fikirleri verecektir. Bilgisayar ağ elemanlarının en uygun çalışma ortamını yaratacak şekilde yerleştirilmesi konusunu inceleyen ve bunun için Topoloji biliminden faydalanan yüzlerce akademik çalışma mevcuttur.

Bu çalışmaların tamamına yakınında Topoloji bilimi kullanılarak, doğadaki örnekleriyle modellenen gerçek hayatta sıkça karşılaşılan problemlerin çözülmesi ve optimize edilmesi konusu işlenmiştir. Topolojinin; çalışılması güç olan bir ortamdaki bazı belirgin şartların, çalışılması nispeten daha kolay olan bir ortama taşınabilmesi imkânından faydalanan araştırmacılar, gerçek hayatta karmaşık olan bazı işleyişleri doğada örnekleri olan daha basit olaylara uyarlayıp sorunların çözümlerini orada aramışlardır. Daha temel alanlarda çözülen problemler, sonrasında günlük hayata ve karmaşık teknolojik çalışmalara uyarlanarak sonuçlar optimize edilmeye çalışılmış. Bu da bilim adamları tarafından ortaya konan bazı problemlerin çözülmesi konusunda yeni yorumlar getirilmesini sağlayarak teknolojik gelişmelere itici bir güç sağlamıştır.

Daha özgül şekilde bahsetmek gerekirse; “Gezgin Satıcı Problemi” olarak adlandırılan bir matematiksel problemin çözümü için bahsi geçen topolojik optimizasyon yöntemleri yoğun olarak kullanılır. Detayları bu çalışmanın ilerleyen kısımlarında incelenecek olan bu konu; bir çöp arabasının şehrin çeşitli yerlerindeki çöpleri toplamak için izleyeceği yolu en kısa ve en hızlı şekilde belirlemesi,

müşterilerinden topladığı geribildirim verilerini kullanarak en kısa sürede ve en geniş kitleye erişim sağlamak isteyen bir çağrı merkezinin müşteri memnuniyetini sağlaması, piyasa eğilimlerini inceleyerek yeni yatırım fırsatları arayan bir borsa komisyoncusunun portföyünün yönetim şeklini belirlemesi gibi problemlerin çözümü için matematiksel formüller üretebilmek için kullanılır.

Bu problemlerin çözümü için kullanılan yöntemler, ilhamını doğadan alan ve matematiksel olarak doğruluğu formüle edilmiş belli modeller ve algoritmalar sayesinde üretilebilmiştir. Genetik Algoritma, Tabu Arama, Tavlama Benzetimi, Yapay Sinir Ağları, Karınca Kolonisi Algoritması şeklinde adlandırılan bu modeller içlerinde matematiksel hesaplamalar barındıran birer çözüm yöntemi olarak tasarlanmışlardır. Sonrasında bu matematiksel hesaplamaların ortaya koyduğu formüller daha karmaşık problemlerin çözümü için kullanılmıştır.

1.1 Bitirme Çalışmasının Konusu

Bu bitirme çalışmasında, yukarıda bahsi geçen modellerin hangi türden sorunların çözümüne nasıl yollar sağladığı incelenecektir. Bu yollar izlenirken hangi türden Topolojik yapıların özelliklerinden faydalandığı, hangi matematiksel formüllerin kullanıldığı detaylı bir şekilde ortaya konacaktır.

Sonuç olarak bilgisayar ağlarının topolojik yapısını optimize eden yöntemleri konu edinen yüzlerce akademik çalışmadan faydalanarak, üzerinde Türkçe pek çalışma yapılmamış ve özgün denebilecek bir konu olan şehir planlaması yöntemlerinin de; bilgisayar – ev, kablo – yol uyarlamasıyla topolojik optimizasyon hesaplamaları marifetiyle kurgulanabileceği öneriler bazı sonuçlarla verilecektir.

1.2 Bitirme Çalışması Düzeni

İkinci bölümde, bilgisayar ağlarının tasarımı konusunda Topoloji biliminin ortaya koyduğu yol ve yöntemler, teknolojik bir dille tanımlamaları ve örneklendirmeleri yapılarak detaylı şekilde ele alınarak ve açıklanacaktır.

Küçük ölçekte bilgisayarların birbirlerine bağlanmaları durumu, büyük ölçekte ise elektrik, su, doğalgaz, telefon, internet gibi altyapı hizmetlerinin şebeke sağlayıcıları tarafından abonelerine ulaştırılması durumu gibi konularda izlenen yollar ve yöntemler topolojik optimizasyon açısından incelenecektir.

Üçüncü bölümde, oluşturulan topolojik yapının optimizasyonu için kullanılan modeller ve algoritmalarla alakalı tanımlamalar ve detaylı açıklamalar yapılacaktır. Her bir model kendi içinde çözüm ürettiği problemle alakalı örneklendirilerek matematiksel formüller ortaya konacaktır. Bu problemlerin günlük hayatta ve bilimsel çalışmalarda hangi konularda uygulamalarının olduğu örneklerle açıklanacaktır.

Dördüncü bölümde, şehir planlaması yöntemlerinin geliştirilmesi açısından Topoloji biliminin önemine değinilip hâlihazırda kurulmuş bazı şehirlerin topolojik incelemesi örneklendirmeler ve grafiklerle incelenecektir. Ütopik olarak sıfırdan kurulabilecek bir şehrin hangi matematiksel formüllere ve topolojik yapılara riayet edilerek kurulabileceği optimizasyon bağlamında araştırılacaktır.

Beşinci bölümde, son olarak bilgisayar ağ topolojilerinin optimizasyon yöntemleri; bilgisayar – ev, kablo – yol, ağ – şehir uyarlaması ile şehir planlaması yöntemlerine uyarlanabilir önerisi detaylandırılacaktır. Bilgisayar ağlarının hızını, maliyetini, güvenilirliğini optimize eden yol, yöntem, modelleme ve algoritmaların, şehir planlama yöntemlerine uyarlanarak şehirlerin çarpık kentleşme, trafik ve kalabalık sorunlarına teorik bir çözüm getirilip getirilemeyeceği araştırılacaktır. “Matematik kullanılarak daha yaşanılır şehirler kurulabilir mi?” sorusuna detaylı bir cevap aranacaktır.

2. BİLGİSAYAR AĞLARININ TOPOLOJİLERİ

Topoloji bilimi, bir ağın fiziksel ve mantıksal yapısını ifade etmekte yoğun olarak kullanılır. Ağı oluşturan bileşenlerin birbirlerine bağlanış şekilleri, kullanılacak aygıtlar, kablolama standartları, iletişim protokolünün seçimi ve bu protokollerin ağ yapısına uygulanabilirliği; topolojik kavramlar ve yapılar kullanılarak incelenir, araştırılır ve optimize edilir.

Bilgisayarların her geçen gün daha hızlı geliştiği günümüzde, bilgisayarlar arası bağlantılar ve veri alışverişi git gide daha çok önem kazanmaktadır. Bilgisayarlar arasında hızlı ve güvenilir bağlantıların kurulabilmesi, küresel dünyada büyüyen bilgisayar ağ tasarımları için önemli problemlerden bir tanesi durumundadır. Bilgisayar ağlarının en önemli görevleri farklı işlem gücüne sahip birden çok bilgisayar arasında veri ve dosya paylaşımının sağlanması; ses, görüntü, yazı aktarımı yapılması, ortak kullanılan programlar arasında veri akışının sağlanması ve paralel işlem yapılmasına olanak sağlayabilmesidir.

Bunun için uygun altyapıyı sağlayabilmek adına geniş çaplı bilgisayar ağları kurulmadan önce matematiksel hesaplamalara dayanan bağlantı yöntemleri planlanır. Topoloji biliminden faydalanarak oluşturulan bu yöntemler; veri iletiminin sağlanacağı ortamın, kısıtlayıcı şartlara en uygun ve en verimli şekilde çalışmasını sağlar [1-7].

2.1. Topoloji Türleri

Bilgisayarlar arası bağlantılar genel olarak ikiye ayrılmaktadır. Birinci tür “Yerel Erişimli Ağlar” (Local Area Networks) olarak adlandırılır. Bu ağlar genelde birbirine yakın bilgisayarları bağlamak için kullanılmaktadır. Örnek olarak, ev, okul, hastane, kütüphane gibi aynı ortamda bulunan bilgisayarlar arasında bağlantı ve veri transferi sağlamak amacıyla kullanılmaktadır.

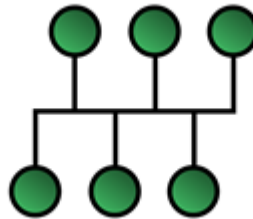
İkinci olarak ise “Geniş Erişimli Ağlar”dan (Wide Area Network) bahsedebiliriz. Bu tip ağlar fiziksel olarak birbirine uzak bilgisayarları veya yerel erişimli bilgisayar ağlarını birbirine bağlamak için kullanılmaktadır. Geniş erişimli ağların en bilinen ve en büyük örneği internet ağıdır.

Çeşitli “Yerel Erişimli Ağları” birbirine bağlamak için ise “Ana Omurga Ağı” (Backbone Network) kullanılmaktadır. Ana omurga ağları yüksek kapasiteli hatlardan oluşan ve yüklü miktarda veriyi taşıyabilen ağlardır. Bilgisayar ağlarında kullanılan her bir bilgisayar veya terminal bir topolojik düğüm olarak nitelendirilmektedir. Bu düğüm noktalarına bağlanmış bir veya birden çok bağlantı olabilir. Bu bağlantılar, bilgisayar ağının topolojisini belirler.

Ağı oluşturan çevre birimlerinin birbirine bağlanırken kullandıkları topolojiler ise, fiziksel bağlantı metotlarını oluşturur. Ağın yapısında kullanılacak kablolu türü ve kullanılacak cihazlar da yine bu topolojide belirlenir. Temel ağ topolojileri yıldız, halka ve yol topolojileridir. Geniş anlamda incelenecek olursa fiziksel topoloji türleri 5 çeşittir. Bunlar; yol topolojisi (bus topology), halka topolojisi (ring topology), yıldız topolojisi (star topology), ağaç topolojisi (hierarchical tree topology) ve karmaşık ağ topolojisi (mesh topology)’dir.

2.1.1. Yol Topolojisi (Bus Topology)

Yol topolojisi bir kablo boyunca tüm terminallerin (sunucular, iş istasyonları ve diğer çevre birimlerinin) doğrusal bir kablo segmentine bağlanması sonucu oluşur. Bu segmente “trunk” adı verilir. Tipik olarak bu “trunk” yapısını ise koaksiyel kablo oluşturur.



Şekil 1 – Yol Topolojisinde Terminallerin Yerleşimi

Yol topolojisinde, sinyal tüm istasyonları dolaşır. Her bir istasyon sinyalin adresini kontrol eder ve bu sinyalin yol üzerinde geçtiği tüm istasyonlar bu adresin kendileri ile ilgili olup olmamaları üzerine sinyali işlerler veya pasif bir şekilde sinyali bırakırlar. Sinyal, istasyonların birbirlerine iletmesi şeklinde değil, kendi başına dolaşarak yol alır.

Ayrıca yol topolojisi kendi içinde klasik (regular) ve yerel (local) olmak üzere ikiye ayrılır. Klasik yol topolojisinde her bilgisayar “omurga” adı verilen tek yönlü bir hat üzerine bağlanırlar. Yerel yol topolojisinde ise her bilgisayar, omurganın kendisini oluşturan birer noktadır. Genelde uçtan uca bağlantılı ağlar, yerel yol topolojisi şeklinde konfigüre edilirler.

Yol topolojisinin avantajları;

- Bilgisayarların ve diğer çevre birimlerinin ağa kolayca bağlanabilmesi,
- Daha az kablo kullanılması,
- Tasarımı ve genişletilebilirliği kolay olması,
- Geçici amaçlı ve kalıcı olmayan ağların hızlı bir şekilde kurulabilmesi için ideal olması,
- Switch veya hub gibi çevresel bağlantı aygıtlarının kullanılmaması ve böylece ek maliyetlerin ortadan kalkması,
- Bir istasyonun çalışmaması durumunda diğerlerini etkilememesi,
- Büyütülebilirlik açısından en ucuz topoloji olmasıdır.

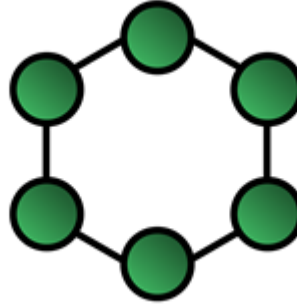
Dezavantajları ise;

- Sorun giderilmesi ve yönetimi zor olması,
- Kısıtlı sayıda istasyon ve kısa mesafe kablo üzerinde olması,
- Ana kabloda oluşan bir kopmanın tüm ağın çalışmasını engellemesi,
- Eklenen her ilave istasyonun toplam ağ performansını kötü anlamda etkilemesi,
- Omurga kablonun her iki ucunda sonlandırıcıların bulunma zorunluluğudur.

Bu topolojide bilinen en yaygın kullanılan kablolama tipi koaksiyel, fiber ve twisted pair kablo ve yaygın kullanılan protokol ise ethernet ve localtalk protokolleridir.

2.1.2. Halka Topolojisi (Ring Topology)

Bu topoloji, bir dairesel (ya da kapalı döngü) uçtan uca bağlantı topolojisidir. Tüm birimler ya doğrudan ya da bir aktarma kablosu ve arayüz ile halkaya bağlıdır. Elektriksel sinyal bir birimden diğer birime tek bir yönde iletilir. Her birim, gelen kabloda alıcı, giden kabloda gönderici işlevi görür. Sinyal her birimde kuvvetlendirildiği veya yeniden oluşturulduğu için zayıflama en alt düzeydedir. Mantıksal olarak halka şeklinde bir yapıya sahip olan bu topoloji aslında fiziksel olarak bir çeşit yıldız topolojisi şeklindedir. İstasyonlar, “Multistation Access Point” (MAU) adı verilen merkezi bir transceiver çevresinde bulunurlar.



Şekil 2 – Halka Topolojisinde Terminallerin Yerleşimi

Token ring yapıda bir paket, halkanın çevresindeki tüm bilgisayarları dolaşarak hedef adrese ulaşır. Token adı verilen ileticiye teslim edilen sinyal, ağ üzerinde hedefe ulaşana kadar halka şeklinde bağlı bulunan istasyonlarca karşılanır ve her istasyon sinyalin kendisine gönderildiğini kontrol eder ve şayet sinyalin üzerindeki adres örtüşmüyorsa sinyali güçlendirerek yeniden halka üzerinde diğer bir istasyona iletir. Sinyal, ilgili istasyona ulaştığında, bu istasyon ilk göndericiye token'ı geri verir ve ağda bir sonraki sinyal taşınımı için yeniden ortam hazırlanır. Bu yüzden halka topolojisi aktif bir topolojidir.

Halka topolojisinin avantajları;

- Ağın büyütülmesi, toplam sistem performansına çok az bir oranda olumlu etki yapar.
- Tüm istasyonlar eşit erişim hakkına sahiptir.

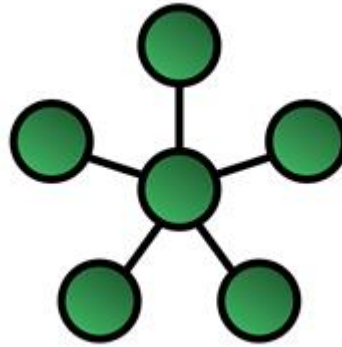
Dezavantajları ise;

- Bilinen en pahalı topolojidir.
- Oldukça karmaşıktır.
- Bir istasyonun arızası durumunda tüm istasyonlar etkilenir.

Bu topolojide yaygın olarak twisted pair ve fiber optik kablolama tipi kullanılır. Uygun protokol ise token ring'dir.

2.1.3. Yıldız Topolojisi (Star Topology)

Yıldız topolojisi, her bir terminalin (sunucular, iş istasyonları ve diğer çevre birimlerinin) switch veya hub adı verilen merkezi konnektörlere direkt olarak bağlanması sonucu oluşur. Veri, hedef adresine gitmek için switch veya hub'dan geçer. Switch veya hub ağın tüm fonksiyonlarını yönetir ve kontrol eder. Ayrıca ağda bir tekrarlayıcı, sinyal güçlendirici (repeater) gibi de çalışırlar.



Şekil 3 – Yıldız Topolojisinde Terminallerin Yerleşimi

Yıldız topolojisinin avantajları;

- Yeni istasyonların eklenmesi kolaydır.
- Yönetimi ve hata tespiti basittir ve kısa zamanda halledilebilir.
- Birbirinden farklı kablolama metotları ile bağdaşabilir.
- Herhangi bir istasyondaki arıza veya yeni bir birimin eklenmesi halinde bundan tüm ağ etkilenmez.

Dezavantajları ise;

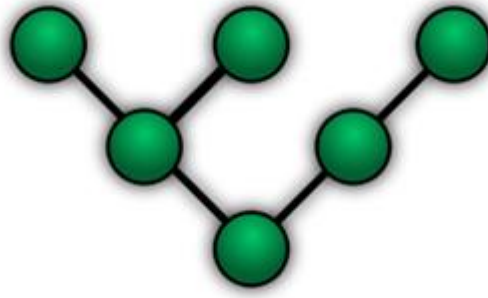
- Diğer topolojilere oranla, çok daha fazla kablo gereksinimi olur.
- Hub veya switch cihazlarında ortaya çıkan sorunlarda tüm ağ etkilenir.
- Bu cihazların kullanılması sonucunda, yol topolojisine göre maliyeti daha yüksektir.

Günümüzde yaygın bir kullanıma sahip olan bu topolojide twisted pair ve fiber optik kablo türleri kullanılır. Ethernet ve localtalk ise yine bu topolojinin yaygın olarak kullanılan protokol tipidir.

2.1.4. Ağaç Topolojisi (Hierarchical Tree Topology)

Temel olarak yol topolojisi ile yıldız topolojisinin karakteristik özelliklerinin kombinasyonu şeklinde ortaya çıkan bir topoloji türüdür. Yıldız şeklinde bağlı istasyonların omurga üzerinde konumlanması sonucu oluşan yol modeli ağaç topolojisini oluşturur. Diğer bir yönden, ağaç topolojisi mantıksal açıdan gelişmiş yıldız topolojisine benzer. Tek farkları ise ağaç topolojisinin herhangi bir merkezi düğüme ihtiyaç duymamasıdır.

İki şekilde ortaya çıkar, omurga ağacı (backbone tree) ve ikili ağaç (binary tree). Omurga ağaç modelinde her düğüm hiyerarşik bir düzen içerisinde alt dallara ayrılır. İkili ağaç yapısında ise her düğüm sadece iki segment halinde bölünerek yapıyı oluşturur. Ağaç topolojisi yapısında sinyalin akış şekli hiyerarşik bir düzende oluşur.



Şekil 4 – Ağaç Topolojisinde Terminallerin Yerleşimi

Bu yapının avantajları;

- Her bir segment için noktadan noktaya bir kablolama yapısı kullanılır, böylece segmentlerde oluşan bir kesinti halinde diğerleri etkilenmez.
- Birbirinden farklı donanım ve yazılım üreticilerinin sağladıkları ürünler uyum içerisinde çalışabilir.

Dezavantajları ise;

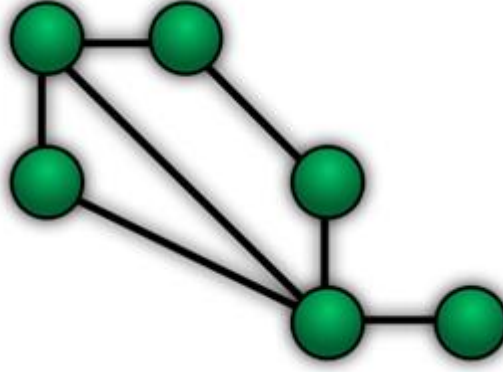
- Kullanılan kablolamanın tipine göre her bir segmentin ortalama uzunluğu belirli bir limiti geçemeyebilir.
- Eğer ana omurga (trunk) yapısında bir kopma olursa tüm ağ işlevini kaybeder.
- Kablolama açısından konfigürasyonu diğer tüm topolojilerden oldukça daha zordur.

Bu yapıda göz önünde bulundurulması gereken bir husus, 5-4-3 ethernet kuralıdır. Bir sinyalin gönderilmesi anında bu sinyal belli bir süre içinde ağın diğer parçalarına ulaşır. Her bir switch, hub veya repeater sinyalin ulaşma süresine nispeten çok küçük bir zaman dilimi daha ekler. Ağdaki iki istasyon arasında maksimum 5 segment olması gerekir ve aynı zamanda fiziksel olarak 4 repeaters, switches veya hub bulunması gerekir. Şayet koaksiyel kablo kullanılmışsa sadece 3 omurga (trunk) olabilir.

Eğer ağ uçtan uca fiber optik kablo ile tesis edilmiş ise veya omurgada fiber kablo ve UTP kablolama ile karma tesis edilmiş ise bu kural 7-6-5 olarak revize edilir. Ethernet protokolünün kullanıldığı ağaç topolojisinde geleneksel olarak kullanılan kablolama türleri ise fiber optik, koaksiyel ve twisted pair kablolardır.

2.1.5. Karmaşık Ağ Topolojisi (Mesh Topology)

Karmaşık ağ topolojisi, ağdaki tüm istasyonların diğer istasyonlar ile uçtan uca kendi aralarında bağlantıları sonucu oluşan topoloji türüdür. Bu yapıda kullanılan kablolamanın çok belirgin avantaj ve dezavantajları vardır.



Şekil 5 – Karmaşık Topolojide Terminallerin Yerleşimi

Bu yapının avantajları;

- Her istasyonun kendi başına diğerleri ile uçtan uca bağlantı kurmasından dolayı, çoklu bağlantı oluşmakta ve böylece herhangi bir bağlantının kopması durumunda, sinyalin hedefine ulaşabilmesi için diğer bağlantıları kullanması en önemli avantajdır.
- Bir istasyondan yayınlanan sinyal farklı hedeflere yöneldiğinde çoklu oluşan bağlantı sayesinde kısa süre içerisinde ağdaki hedeflerine varacaktır, böylece taşınım zamanı kısılacaktır.

Dezavantajları ise;

- Ağ üzerinde az sayıda düğümün bulunduğu durumlarda ve ortam boyutunun küçük olması halinde ortaya çıkan bağlantı miktarının çok fazla gözükmesi ve bu durumda ağ hızının yavaşlamasıdır.

Mantıksal bir perspektiften bakılacak olunursa, bu yapının durumu, performansı, ağdaki merkezi dağıtıcıların ve diğer cihazların sayısı ile doğru orantılıdır. Ayrıca ağdaki her birim diğer tüm birimler için birer bağlantı gerektirdiğinden dolayı genellikle uygulamada pek fazla pratik bulunmayan bir özelliğe sahiptir.

2.2. Ağ Topolojisinin Karmaşıklığı

Bir bilgisayar ağının yapısı $G = (N, L, p)$ fonksiyonu ile gösterilebilir. Burada N bilgisayar sayısını, L bilgisayarlar arasındaki topolojik düğümleri ve p bu bağlantı hatlarının güvenilirliğini göstermektedir. Topolojinin büyüklüğüne bağlı olarak L , aşağıdaki formüle göre hesaplanır.

$$L = \frac{N(N - 1)}{2}$$

Çözüm uzayı ise, L değerine bağlı olarak aşağıdaki gibi üstel bir artış gösterir.

$$\zeta_{uzayl} = 2^L$$

Örnek olarak, 7 bilgisayardan oluşan bir topoloji için;

$L = 21$ ve $\zeta_{uzayl} = 2,1 \cdot 10^6$ olurken, 20 bilgisayarlı bir topoloji için;

$L = 190$ ve $\zeta_{uzayl} = 1,56 \cdot 10^{57}$ değerlerine ulaşılmaktadır.

2.3. Ağ Topolojisinin Güvenilirlik Kriterleri

Yeni bir ağ tasarımı yapılırken maliyet, performans, üretilen veri hacmi, bağlantı gereksinimleri ve güvenilirlik gibi pek çok kriter rol oynamaktadır [8]. Bu kriterlerin içinde en önemli olanı ise, kurulan ağın çalışmasına devam edebilmesi açısından, güvenilirlik kriteridir. Özellikle omurga ağlarda sistemin sürekliliği oldukça önemlidir. Ağ büyüklükleri arttıkça ağlardaki bozulmalar ve problemler de paralel bir artış göstermektedir. Bu hataların bir kısmı yazılımsal bir kısmı ise donanımsal hatalardır. Bu çalışmada, ağlardaki donanımsal bozulmalar altında ağın hayatta kalması ve veri aktarımına devam etmesi incelenmiştir. Ele alınan problemde bilgisayarların bozulmadığı, hatların arızalandığı durum dikkate alınmıştır.

2.3.1. Belirli Kriterler

Ağ güvenilirliğinde belirli ölçüler, ağın çalışamaz hale gelmesi için arızalanması veya devre dışı kalması gereken bileşenler ile ilgilenmektedir. Altıparmak [9] tarafından yapılan çalışmada, bağlılık, birleşme, çap, çevre ve ağın birleşme yüzeyi gibi parametreler aşağıdaki gibi açıklanmıştır.

Bağlılık: Bir ağdaki her bir düğüm noktasının ağın tamamına en az bir hat ile bağlı olması anlamına gelmektedir. Yani ağın içerisindeki tüm düğüm noktalarına en az bir hattın bağlı olması anlamına gelmektedir. Bununla birlikte ağdaki tüm düğümlerin birbirine bağlı olması için ağın en az bir yayılan ağacı kapsaması gerekmektedir. Ağın güvenilirliğinin yüksek olması için bağlılık derecesinin de yüksek olması gerekmektedir. Düğüm çiftleri arasında en az iki hat olması, yani bağlılık, güvenilirliğin yüksek olmasının istendiği durumlarda ve ana şebeke tasarımlarında kullanılmaktadır.

Birleşme: Bir ağdan m düğümlü alt ağlar elde edebilmek için, ağdan çıkartılması gereken hatların ya da düğümlerin en küçük sayısıdır. Yüksek güvenilirliğe sahip bir ağda, elde edilen tüm alt ağlar için birleşme değerinin büyük olması gerekmektedir.

Çap: Ağdaki düğüm çiftleri arasındaki en kısa yolların içindeki en uzun yol bize ağın çapını gösterir. Çap değeri ne kadar düşük olursa, ağın güvenilirliği de o kadar yüksektir.

Çevre: Ağdaki başlangıç ve bitiş düğümleri aynı olan yol olarak tanımlanmaktadır. Çevrim değerinin uzunluğu kapsadığı hatların toplam uzunluğuna eşittir. Bir ağın çevrim değeri ise ağdaki çevrim değerlerinin en küçük uzunluğuna eşittir. Çevrim değerinin büyük olması, ağın güvenilirliğinin yüksek olduğunu göstermektedir.

Ağın Birleşme Düzeyi: Bir ağ kendisinden daha küçük en az iki veya daha fazla alt ağa bölmek için çıkartılması gereken en az düğüm veya hat sayısı, ağın düğüm ya da hat birleşme düzeyi değerini vermektedir. Birleşme düzeyi değerinin yüksek olması ağın güvenilirlik düzeyinin de yüksek olduğunu göstermektedir.

2.3.2. Olasılıklı Kriterler

Olasılıklı ölçüler ile ölçümlenen güvenilirlikte, ağda bulunan parçaların belki olasılıklarla çalışması veya bozulması durumu vardır. Olasılıklı güvenilirliğin hesaplanmasında literatürde üç farklı problem türü vardır. Bunlar iki terminal güvenilirlik problemleri, tüm terminal güvenilirlik problemleri ve k terminal güvenilirlik problemleridir.

İki terminal Güvenilirlik Problemi: Bilgisayar ağındaki bir kaynak düğümünden hedef düğümü arasında en az bir işlevsel yol ile bağlılığın olması olasılığının hesaplandığı problemlerdir. ($T = \{s, t\}$)

Tüm Terminal Güvenilirlik Problemleri: Ağdaki her bir v_1, v_2 düğüm ikililerinin arasında en az bir işlevsel yol ile bağlılığın olması olasılığının hesaplandığı problemlerdir. ($T = V$) Bir başka deyişle ağın en az bir yayılan ağacı kapsaması gerekmektedir.

K Terminal Problemleri: Ağdaki K adet belirlenmiş düğüm noktasının ($2 \leq K \leq n$ olmak üzere) tamamını birbirine bağlayan en az bir işlevsel yol ile bağlılığın olması olasılığının hesaplandığı problemlerdir.

2.3.3. Ağ Güvenilirliğinin Değerlendirilmesi

Ağ güvenilirliğinin değerinin hesaplanması ve değerlendirilebilmesi için literatürde kullanılan üç temel yaklaşım vardır. Bunlar, ağın güvenilirliğinin tam değerinin hesaplanması, ağ güvenilirliğinin sınırlarının bulunması ve güvenilirlik değerinin benzetim veya diğer tahmin yöntemleri ile tahmin edilmesidir.

Tam Değer Hesabı: Tasarlanan bir haberleşme veya bilgisayar ağının güvenilirliğinin tam değerinin hesaplanması için literatürde kullanılan çeşitli yöntemler vardır. Bunlara örnek olarak *Kesme/Yol* kümesinin sayımı metotları ve *Durum Sayma* metotları verilebilir. Ancak tam değer hesaplanabilmesi için gereken işlem sayısı, ağın büyüklüğüne bağlı olarak üstel bir hızla artmaktadır. Ağdaki hat sayısı L olduğuna göre, ağın güvenilirliğinin tam hesaplanması için farklı durumun hesaplanmasına ihtiyaç vardır. Bu dezavantajından dolayı, büyük boyutlu problemlerde tercih edilen bir yöntem değildir.

Sınır Hesabı: Bir haberleşme veya bilgisayar ağında güvenilirlik değerlerinin alt ve üst sınır değerlerinin hesaplanması oldukça tercih edilen bir yöntemdir. Literatürde bu amaçla Jan, R, H. [10] tarafından üst sınır formülü ortaya atılmıştır.

3. ALGORİTMALAR VE OPTİMİZASYON YÖNTEMLERİ

Bir problemin çözümü ile ilgili en önemli kısıtlamalar, problemin içinde bulunduğu uzayın şartlarının çözümü zorlaştırdığı hatta imkânsız kıldığı durumlardır. Topoloji bilimi ise bu problemin çözümü ile ilgili en önemli yardımcı etkeni bize sunmaktadır. Çalışılan uzayın şartlarından bağımsız bir şekilde problemi çözme imkânı sağlayan Topoloji, çözümün farklı uzaylara da uygulanması konusunda yardımcı olmaktadır. Yani çalışılan uzayın zorluk oluşturan şartlarından etkilenmemek adına, problem yeni bir uzaya taşınır; orada çözülür, sonrasında tekrar ilk uzaya uyarlanarak genel manada problem çözümü tamamlanmış olur.

Diğer taraftan, doğada çözümü mevcut olan veya suni yollarla üretilmiş ve çözülmüş bazı temel problemlerin çözümleri, daha karmaşık olaylara uyarlanarak bazı modellemeler ve optimizasyon yöntemleri geliştirilmiştir. Bu yöntemler algoritmik şekilde düzenlenerek, araştırmacılar tarafından zor problemlerin çözümü ve optimize edilmesi için sıkça kullanılmıştır. Bu bölümde bu türden çözümler ve optimizasyon yöntemleri ile alakalı örneklendirmeler ve detaylı yöntem analizleri ele alınacaktır. Bunların matematiksel karşılıkları formülleri ile birlikte açıklanmaya çalışılacaktır.

3.1. Literatür İncelemesi

Bahsi geçen problemlerin çözümü ile ilgili birçok bilim insanının araştırdığı ve geliştirdiği birçok modelleme mevcuttur. Bilgisayar ağlarının topolojik yapıları ile ilgili literatürdeki ilk çalışmalardan birisi Boorstyn ve Frank [11] tarafından yapılmıştır. Bu çalışmada ağ tasarımı problemlerinden işlemci yerleşimi problemi, terminal atama problemi, en küçük yayılan ağaç probleminin kısıtlandırılmış bir hali olan terminal yerleşim problemi ve dağınık bilgisayar ağlarının topolojik yerleşimi problemi ve ana ağ düğüm yerleşimi problemleri incelenmiştir.

Aggaewal ve Suresh [12] bilgisayar ve haberleşme ağlarının güvenilirliğini değerlendirmek için yayılan ağaç temelli bir metot geliştirmişlerdir. Ağ güvenilirliği ve $s - t$ terminal güvenilirliği bütün hatların eşit olasılıkta arızalandıkları varsayımı altında incelenmiş ve örneklenmiştir.

Chopra vd. [13], bilgisayar ağları için, bilgisayarların bulunduğu düğümlerin bilindiği varsayımı ile maliyet kısıtını dikkate alarak $s - t$ güvenilirliğini en çoklayan yeni bir algoritma sunmuşlardır.

Güvenilirlik kısıtı altında bilgisayar ağlarının topolojik en iyilenmesi için Venetsanopoulos ve Singh [14] tarafından geliştirilen algoritmada, güvenilirlik kısıtı için yeni bir ölçme sistemi kullanılmış ve buna dayalı olan sezgisel bir algoritma geliştirilmiştir. Geliştirilen algoritmanın çözüm hızını arttırdığı gösterilmiştir.

Kumar vd. [15] var olan bir bilgisayar ağının genişletilmesi problemini ele almıştır. Var olan bilgisayar ağının güvenilirlik kısıtını bozmadan, maliyet kısıtı altında yeni düğüm ve hatlar eklenmesinde Genetik Algoritma'yı kullanmışlardır. Geliştirilen algoritmanın (Genetic Algorithm Based Computer Network Expansion Methodology - GANE) amaç fonksiyonu üzerinde yapılabilecek değişiklikler ile bu problemin farklı türlerinin çözümü için de kullanılabileceği belirtilmiştir. GANE ile elde edilen sonuçlar birerleme metodu ile elde edilen çözümler ile karşılaştırılmış ve algoritmanın hesaplama zamanı açısından daha iyi olmasına rağmen problem için en iyi sonucu garanti etmemektedir.

Kumar, Pathak, Gupta ve Parsaei [16] bilgisayar ağlarının topolojik tasarımı problemine Genetik Algoritma tabanlı genelleştirilmiş bir algoritma sunmuşlardır. Yapılan çalışmada çap, ortalama uzaklık ve bilgisayarların güvenilirliği parametreleri dikkate alınmıştır. Bu yaklaşım aynı zamanda geniş alan ağlarına da uygulanabilmektedir. Elde edilen sonuçlar ayrıntılı araştırma sonuçları ile karşılaştırılmış ve geçerliliği kanıtlanmıştır.

Costamagna vd. [17], tavlama benzetimi yöntemi kullanılarak haberleşme ağlarının topolojik optimizasyonu üzerine geliştirdikleri algoritmayı sunmuşlardır. Çalışmanın sonuçları literatürdeki diğer yöntemler ile bulunan sonuçlarla karşılaştırılmış ve önerilen algoritmanın başarılı olduğu gösterilmiştir.

Deeter ve Smith [18], topolojik yerleşim problemi için, tüm terminal güvenilirlik kısıtı altında Genetik Algoritma kullanılmıştır. Her hat için farklı hat güvenilirlikleri olmak üzere, yaklaşık veya tam sistem güvenilirliği hesaplanmış ve literatürdeki test problemleri üzerinde algoritmanın esnek ve etkin çalıştığı gösterilmiştir.

Hahuja [19] bilgisayar ağlarında kapasite ve performans probleminin çözümü üzerine yoğunlaşmıştır. Problemi Genetik Algoritma kullanılarak, sabit maliyet kısıtını dikkate alarak, en büyük güvenilirlik değeri için çözmüş ve sonuçlar literatürdeki en iyi çözümler ile karşılaştırılmıştır.

Dengiz, Altıparmak ve Smith [20] belirli bir güvenilirlik kısıtı altında, en küçük maliyetli bilgisayar ağı topolojisini oluşturmak için Genetik Algoritma'yı kullanmışlardır. Ağ gösterimini 0-1 şeklinde kodlamışlar ve uygun olmayan çözümler için ceza fonksiyonundan yararlanmışlardır. Bu çalışmada geliştirilen algoritmanın sistem güvenilirliğini daha hızlı hesaplayabilmesi için Monte Carlo benzetimi ile güvenilirlik tahmini yapılmıştır.

Pierre ve Legault [21] dağıtık bilgisayar ağlarının topolojik en iyilenmesi için gecikme ve güvenilirlik kısıtları altında çözüm arayan bir algoritma sunmuşlardır. Genetik Algoritma'ya dayalı bir yöntem geliştirmişler ve bu yöntem ile çözülen orta büyüklükteki problemler için literatürdeki bilinen yöntemlere göre daha iyi sonuçlar elde etmişlerdir.

Pierre ve Elgibaoui [22] tarafından yapılan çalışmada, tabu arama metodu kullanılarak, güvenilirlik kısıtı altında en küçük maliyetli bilgisayar ağı tasarımı problemleri için bir algoritma sunulmuştur. Geliştirilen algoritma her iterasyonda mevcut topolojinin komşuluklarını üretmekte ve üretilen bu çözümlerden güvenilirlik

kısıtını saęlayan en kk maliyetli zm seilmektedir. Bu iřlem aday topolojinin komřuluęu kalmayıncaya kadar devam etmektedir. Bu yaklařım genelleřtirilmiř yerel arama metodu olarak isimlendirilmiřtir. On iki ile otuz dęm arasındaki byklklerden oluřan test problemleri iin elde edilen sonular literatrdeki dięer algoritmalar ile karřılařtırılmıřtır. Karřılařtırma sonularına gre nerilen algoritmanın daha iyi sonular verdięi gsterilmiřtir.

Costamagna vd. [23] Tabu Arama Algoritması'nı kullanarak yaptıkları alıřmada oklayıcı merkezlerinin yerleřimi problemi zerinde durmuřlardır. Oluřturdukları aęda fiber optik kablo kullanmıřlar ve fiber optik kabloların yksek maliyetinden dolayı yayılan aęa yapısını tercih etmiřlerdir. Seilen oklayıcıların durumunu belirtmek zere ikili (0-1) kodlama sistemi kullanmıřlardır. Geliřtirdikleri algoritmanın yapısında dinamik tabu listesi, frekans temelli uzun dnemli hafıza ve durdurma kořulu kullanan yazarlar, elde ettikleri sonuları literatrdeki dięer sonularla karřılařtırmıřlar ve zm kalitesi aısından geliřme saęlamıřlardır.

Gheng [24] ana bilgisayar aęları iin toplam hat maliyetini en aza indirmeyi hedefleyen ve sadece bir hat bozulmasına izin verilen bir yaklařım geliřtirmiřtir. Genetik Algoritma kullanarak geliřtirilen yaklařım ile literatrdeki test problemleri bu alıřmada zlmřtir.

Deeter ve Smith [25] haberleřme ve bilgisayar aęlarının tasarımı problemi iin Genetik Algoritma'yı kullanarak, maliyet ve gvenilirlik kısıtlarını dikkate alan genelleřtirilmiř bir yaklařım ile zmřlerdir. Arařtırma sonuları geliřtirilen algoritmanın daha hızlı ve verimli alıřtıęını gstermektedir.

Altıparmak, Dengiz ve Smith [26] Genetik Algoritma ile maliyet kısıtı altında gvenilirlięin en byklenmesi problemine sezgisel bir yaklařım uygulamıřlardır. Dęm noktaları ve aralarındaki hatları tamsayı olarak kodlamıřlar ve literatrdeki test problemlerini zmřlerdir.

Konak ve Smith [27] bilgisayar omurga ağ tasarımı problemi için melez bir genetik algoritma sunmuşlardır. Hat yoğunluğunun da göz önüne alındığı araştırmanın sonuçları geliştirilen algoritmanın literatürdeki diğer çalışmalara göre oldukça etkin olduğunu göstermişlerdir.

Liu ve Iwamura [28] Genetik Algoritma, bağımlı şans çok amaçlı programlama, bağımlı şans amaç programlama yöntemlerini kullanarak çoklu güvenilirlik amaçlı problemler için yeni bir yaklaşım geliştirmişlerdir. Geliştirilen yaklaşımın etkinliği literatürdeki sayısal örnekler üzerinde gösterilmiştir.

Aboelfotoh ve Al-Sumait [29] tüm terminal güvenilirlik kısıtını dikkate alarak en küçük maliyet için topolojik yerleşimin bulunmasını amaçlamışlardır. Yapay sinir ağları tabanında geliştirilen Opti-Net Algoritması ile yapılan çalışmada literatürdeki test problemleri çözülmüş ve Genetik Algoritma ile karşılaştırılmıştır. Test sonuçlarına göre özellikle büyük boyutlu problemlerde daha iyi sonuçlar elde edilmiştir.

Fard ve Lee [30] var olan bilgisayar ağının güvenilirliğini arttırmak için yeni bir algoritma sunmuşlardır. Düğüm çiftlerinin arasına yeni hat eklenmesi ile oluşan yayılan ağaç sayılarına bakılarak hangi düğüm çifti arasına yeni hat eklenmesi gerektiği hesaplanmaktadır. Sayısal bir güvenilirlik hesaplaması yapılmasına ihtiyaç duyulmayan bu yöntemde ayrıca ağın derece matrisinden de faydalanılmaktadır.

Koide vd. [31] yaptıkları çalışmalarında, Jan ve arkadaşları tarafından, tüm terminal güvenilirlik kısıtı altındaki topolojik ağların en iyilenmesi problemi için önerilen algoritmayı geliştirmişlerdir. Algoritmalarını, farklı hat olasılıklarına sahip topolojik problemleri çözebilmek için uygun hale getirmişlerdir. Ayrıca algoritmaya çeşitli eklemeler yaparak hızlandırmışlar ve daha kaliteli çözümler elde ettiklerini belirtmişlerdir.

Kumar, Parida ve Gupta [32] haberleşme ağlarının topolojik tasarımı için Genetik Algoritma tabanlı ve çok ölçütlü bir model tasarlamışlardır. Birbirlerine en az bir

ölçüte göre baskın gelemeyen çözümlerden faydalanılmış ve Pareto Yakınsak Genetik Algoritma ile çözüm aramışlardır. Elde edilen test sonuçları diğer sezgisel yöntemler ile elde edilen sonuçlarla karşılaştırılmıştır.

Srivaree-ratana, Konak ve Smith [33] tüm terminal ağ güvenilirliğinin tahmini için yapay sinir ağları yöntemini kullanmışlardır. Geliştirilen yöntem için gereken eğitim sürecinde üst limit yöntemi kullanılmıştır. Yapay sinir ağları ile geliştirilen algoritma, literatürdeki diğer yöntemler ile karşılaştırılmış ve işlem zamanı açısından çok büyük kazançlar sağlanmıştır.

Altıparmak, Dengiz ve Smith [34] maliyet kısıtını dikkate alarak ağ güvenilirliğinin en büyüklenmesi problemi için bir sezgisel yöntem geliştirerek literatürdeki önerilen yöntemler ile genel bir karşılaştırma yapmışlardır. Tepe tırmanma, tavlama benzetimi, Genetik Algoritma ve bu algoritmanın karma bir versiyonu olan Memetik Algoritmayı bilinen test problemlerinin çözümünde kullanmışlardır. Elde ettikleri sonuçlara göre Memetik Algoritmanın performansı diğer algoritmalara göre daha iyi olduğunu belirtmişlerdir.

Mandal vd. [35] gerçek hayat kısıtlarına yakın güvenilirlik kısıtları kullanarak omurga ağ tasarımı üzerinde durmuşlardır. Ağdaki herhangi bir hat arızalandığında da ağın çalışmaya devam etmesi kısıtı altında RAS Algoritması'nı kullanarak, toplam maliyeti en küçükmeye çalışmışlardır. Araştırma sonuçlarını Genetik Algoritma ile bulunan sonuçlarla karşılaştırmışlar ve geliştirdikleri algoritma ile daha kaliteli sonuçlar elde etmişlerdir.

Altıparmak, Dengiz ve Smith [36] tüm terminal bilgisayar ağlarının güvenilirliğini tahmin etmek için yapay sinir ağları yöntemini kullanmışlardır. Çalışmada homojen ve heterojen hat güvenilirlikleri de dikkate alınmıştır. Yapay sinir ağlarının eğitim sürecinde rassal ve deneysel tasarım yöntemleri kullanılmış ve deneysel tasarım ile elde edilen eğitim setinin daha iyi sonuçlar verdiği belirtilmiştir.

Altıparmak, Gen, Dengiz ve Smith [37] güvenilirlik kısıtı altında bilgisayar ağlarının topolojik en iyilenmesi için ağ tabanlı ve bulanık mantık kontrollü bir Genetik Algoritma yapısı (flc-NB GA) sunmuşlardır. Araştırmada Prüfer sayı tabanlı bir kodlama yöntemi kullanılmış, iki noktalı çaprazlama yapılmış ve mutasyon işlemi için yerel arama yöntemi kullanılmıştır. Geliştirilen algoritmanın bulduğu sonuçlar, Dal Sınır Algoritması, Genetik Algoritma tabanlı ikilik gösterim ve ağ tabanlı Genetik Algoritma sonuçları ile karşılaştırılmıştır.

Shao vd. [38] maliyet kısıtını dikkate alarak, dağıtımli erişime sahip ağların güvenilirliğini en iyilemeye çalışmışlardır. Öncelikle maliyet kısıtlı güvenilirlik problemi kombinatoriyal ağaç arama işlemi olarak ele alınmıştır. Daha sonra bu yöntemin çok fazla işlem gücü ve zaman gerektirmesi sebebi ile daha hızlı bir yöntem olan Daraltma ve Arama Algoritması uygulanmıştır. Geliştirilen algoritmanın etkinliği simülasyon ve örnek olay incelemeleri ile test edilmiştir.

Xiong ve Gong [39] ağ güvenilirliğinin tahmini için Oransal Yaklaşım Algoritması kullanmıştır. Geliştirilen algoritmanın tüm terminal sistemlere rahatlıkla uygulanabildiği ve güvenilirlik fonksiyonu eğrisini sağladığı gösterilmiştir.

Reichelt, Gmilkowsky ve Linser [40] tekrarlı yerel arama metodunu kullanarak ağ topolojilerinin en iyilenmesi problemini çözmüşlerdir. Burada, Yerel Arama Algoritması'nın geliştirilmiş bir versiyonu olan Tekrarlı Yerel Arama Algoritması'nda daha düşük maliyetli ve güvenilirlik kısıtını bozmayan çözümler aranmaktadır. Araştırma sonuçları Genetik Algoritma yöntemi ile karşılaştırılmış ve önerilen algoritmanın daha iyi olduğu gösterilmiştir.

Marseguerra vd. [41] Genetik Algoritma ve Monte Carlo simülasyonu bir arada kullanılarak, bilgisayar ağlarının topolojik tasarımı için çok amaçlı bir eniyileme modeli geliştirilmiştir. Güvenilirlik tahmini için hem bağlantıların hem de düğüm noktalarının güvenilirlik düzeylerinden faydalanmışlardır. Birbirine en az bir kriter gere göre baskın gelemeyen çözüm kümeleri arasından, karar vericiler yardımı ile

özmlerin risk profillerine gre seim yapılmıřtır. nerilen bu algoritma ile farklı tasarımlara ulařmak mmkn olmuřtur.

Gen, Kumar, Kim [42] bilgisayar aęlarının tasarımı problemi iin yayılan aęa tabanlı melez bir Genetik Algoritma uygulamıřlardır. nerilen yntem derece-kısıtlı en kk yayılan aęa problemleri, yetkilendirilmiř en kk yayılan aęa problemleri, sabit ykl tařıma problemleri ve yerel alan aę tasarımı problemlerine uygulanabilmektedir. Arařtırma sonuları, Genetik Algoritma yaklařımının bu alanda nemli bir potansiyele sahip olduęunu gstermektedir.

Reichelt ve Rothlauf [43] haberleřme aęı tasarımı problemini gvenilirlik kısıtını dikkate alarak en kk maliyet iin zmřtur. Bu alıřmada evrimsel yntemlere dayalı iki yeni algoritma sunulmuřtur. Bulunan uygun olmayan zmler iin dzeltme algoritması uygulanmıřtır. Uygun olmayan zmlere ceza yntemi uygulayan evrimsel algoritmalar ile nerilen algoritmalar karřılařtırılmıř ve nerilen algoritmaların daha iyi sonular verdięini gsterilmiřtir.

Konak ve Bartolacci [44] bilgisayar aęlarının tasarımının en iyilenmesi problemi iin gvenilirlik kısıtı yerine aę esneklięi kısıtını dikkate alarak en kk maliyetli tasarımı bulmayı amalamıřlar ve bunun iin Karma Genetik Algoritma kullanmıřlardır. Aę esneklięinin tahmin edilmesi iin aędaki trafik durumundan faydalanan bir yntem geliřtirmiřlerdir. Kullanılan Karma Genetik Algoritma'da, zelleřtirilmiř yerel arama operatrleri ve basitleřtirilmiř uygun ceza fonksiyonları kullanılmıř, olduka etkin sonular sunmuřlardır.

Cancela ve Petingi [45] bilgisayar aęlarının topolojik tasarımı probleminde gvenilirlik kısıtı yerine dęmler arası direkt uzaklıęı, yani apı dikkate almıřlardır. K-terminal gvenilirlięin yerine tasarladıkları ap kısıtlı gvenilirlik yaklařımını baskınlık durumu ile birlikte deęerlendirmiřtir.

Khan ve Engelbrecht [46] yerel bilgisayar aęlarının topolojik tasarımı iin birbirleriyle ters orantı ierisinde olan maliyet, gvenilirlik, aę gecikmesi ve kaynak

ile hedef arasındaki üst üste binen yollar gibi amaçları bir araya getiren yeni bir bulanık operatörü geliştirmişlerdir. Tasarladıkları operatörü literatürde bilinen diğer bir bulanık operatörü ile karşılaştırmışlar ve önerdikleri operatörün performansının daha iyi olduğunu göstermişlerdir.

Lucio vd. [47] bilgisayar ve haberleşme ağlarının topolojik tasarımı problemini, ağdaki yoğunluğu temel alarak yeni bir yöntem ile çözmüşlerdir. Tepe tırmanma metodunu geliştirerek hızlı yerel arama (FLS) ve yerel en küçük değerlerden kurtulabilen yönlendirilmiş yerel arama (GLS) yöntemlerini geliştirmişlerdir. Ayrıca bu iki yöntemi birleştirmişlerdir. Oluşturdukları yeni algoritmayı Genetik Algoritma ve Tavlama Benzetimi Algoritması ile karşılaştırmışlar, önerdikleri algoritmanın daha az parametre ile daha iyi sonuçlar verdiğini göstermişlerdir.

Hui [48] büyük boyutlu ağ en iyilenmesi problemleri için kullanılan güvenilirlik tahmini yöntemlerinin çok fazla işlem zamanı gerektirmesinden dolayı tüm ağın güvenilirliğine ihtiyaç duymadan güvenilirlik derecesini tahmin eden bir yöntem geliştirmiştir. Yazar tarafından önerilen algoritma, diğer yöntemlere göre 30.000 kat daha hızlı sonuçlar verebilmektedir.

Marqueza ve Rocco [49] tüm terminal güvenilirlik kısıtı altında en küçük maliyetli amaç fonksiyonuna sahip bilgisayar ağlarının topolojik en iyilenmesi problemi için evrimsel melez bir algoritma kullanmıştır. Olasılıklı çözümler ve Monte Carlo simülasyonundan faydalanmışlar ve geliştirdikleri algoritmayı literatürdeki test problemleri ile test etmişlerdir.

Sem ve Malhotra [50] yirmi düğüme sahip ya da daha küçük boyutlu topolojik ağ tasarımı problemlerini Genetik Algoritma tabanlı yeni bir yöntem ile çözmüştür. Tüm terminal güvenilirlik kısıtı altında en küçük maliyetli ağ tasarımı problemini dikkate almışlar ve önerdikleri algoritmayı literatürdeki diğer Genetik Algoritma tabanlı yaklaşımların işlem zamanları ile karşılaştırmışlar ve daha verimli sonuçlar elde etmişlerdir.

3.2. Modellemeler

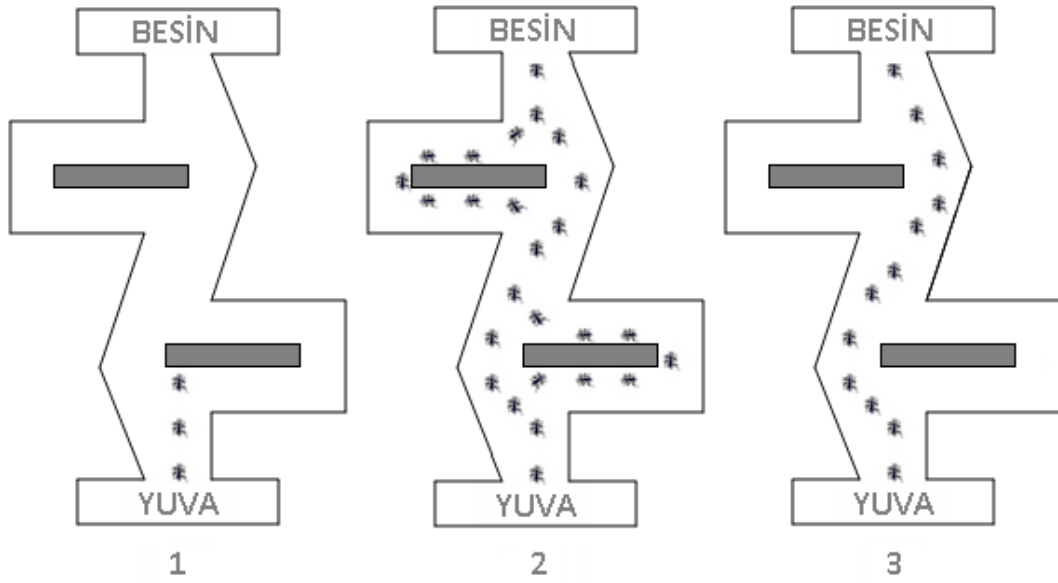
Bu başlık altında, yukarıda listelenen çalışmaların temelinde kullanılan hesaplama yöntemleri ve yapılan matematiksel işlemlerin oluşturduğu formüller, literatürde geçen isimleriyle örneklendirilerek detaylı şekilde açıklanacak ve incelenecektir.

3.2.1. Karınca Kolonisi Algoritması

Bilim adamları, böcek davranışlarını inceleyerek başarılı optimizasyon algoritmaları geliştirmişlerdir. Bu teknikler birçok bilimsel alanda ve mühendislik problemlerinde başarıyla uygulanmıştır.

Teknikler, statik problemlerdeki yüksek performanslarına ilaveten, dinamik özellik gösteren problemlerde de yüksek derecede esnekliğe sahiptirler. Karınca algoritmaları ilk olarak Dorigo [51] ve meslektaşları tarafından; gezgin satıcı problemi (GSP) ve kuadratik atama (QAP) gibi zor optimizasyon problemlerinin çözümü için geliştirilmiştir. Optimizasyon problemlerinin çözümü amacıyla karınca algoritmaları üzerine birçok çalışma devam etmektedir.

Karıncalar, yiyecek kaynaklarından yuvalarına en kısa yolu görme duyularını kullanmadan bulma yeteneğine sahiptirler. Aynı zamanda, çevredeki değişime adapte olma yetenekleri vardır. Dış etkenler sonucu takip ettikleri mevcut yol artık en kısa yol değilse, yeni en kısa yolu bulabilmektedirler.



Şekilde de görüldüğü gibi karıncalar, başlangıçta rastlantısal bir yolu takip etmekte ve bu esnada feromon olarak adlandırılan bir hormonu salgılayıp yol güzergâhına bırakarak kendilerinden sonra gelen karıncaların daha kısa olan yoldan besin kaynağına gitmeleri konusunda yol göstermektedirler. Bu algoritma aşağıdaki özelliklerle tanımlanabilir:

- Önlerine bir engel konulduğunda feromonları takip edemediklerinden, karıncalar gidebilecekleri iki yoldan birini öncelikle rastsal biçimde seçerler.
- Kısa olan yoldan birim zamandaki geçiş daha fazla olacağından bırakılan feromon miktarı daha fazla olur.
- Buna bağlı olarak, zaman içerisinde kısa olan yolu tercih eden karıncaların sayısında artış olur.
- Başta rastsal hareket eden karıncaların izleri kontrol ederek yüksek olasılıkla izlerin yoğun olduğu yönü takip etmesi otokatalitik bir davranış şeklidir ve karıncaların karşılıklı etkileşimde bir sinerjik etki vardır.
- Algoritma, karıncaların hareketlerinden esinlenerek geliştirildiğinden sistem Karınca Sistemi (KS), algoritma ise Karınca Kolonisi Algoritması (KKA) olarak adlandırılır.
- Karınca Kolonisi Algoritması Optimizasyon problemlerinde kullanılır.

Karınca Kolonisi Algoritması'nın döngüsü ise aşağıdaki gibidir:

ADIM 1: Başlangıç feromon değerleri belirlenir.

ADIM 2: Karıncalar her topolojik düğüme rastsal olarak yerleştirilir.

ADIM 3: Her karınca olasılığa bağlı olarak seçim yapma suretiyle turunu tamamlar.

ADIM 4: Her karınca tarafından kat edilen yolların uzunluğu hesaplanır.

ADIM 5: En iyi çözüm hesaplanır ve global feromon yenilemesinde kullanılır.

ADIM 6: Maksimum iterasyon sayısı sağlanana kadar ADIM 2'ye gidilir.

Geçiş kurallarının matematiksel hesaplaması ise aşağıdaki gibidir:

$$j = \max_{u \in J_k(i)} \{ [\tau(i, u)]^\alpha \cdot [\mu(i, u)]^\beta \} \quad \text{eğer } q \leq q_0 \text{ ise}$$

Burada $\tau(i, u)$, (i, u) hattındaki feromon izidir. $\mu(i, u) = 1/\delta(i, u)$ i noktasından u noktasına uzaklığın tersidir. $J_k(i)$ i noktasındaki k karıncası tarafından henüz gidilmemiş yolları temsil etmektedir. β ($\beta > 0$) feromonun güncellenmesinde, uzaklığın göreceli önemliliğini belirleyen parametredir. q_0 ($0 < q_0 < 1$) ise çözüm uzayını araştırmanın göreceli önemliliğini gösteren parametredir.

Gidilebilecek yolların seçilebilme olasılıkları ise aşağıdaki formülle hesaplanır:

$$P_k(i, j) = \begin{cases} \frac{[\tau(i, u)]^\alpha \cdot [\mu(i, u)]^\beta}{\sum_{u \in J_k(i)} [\tau(i, u)]^\alpha \cdot [\mu(i, u)]^\beta}, & \text{eğer } j \in J_k(i) \\ 0, & \text{diğer durumlar} \end{cases}$$

t iterasyonuna kadar biriken feromon düzeyine bağlı olarak karıncaların toplam tur uzunlukları da aşağıdaki formülle bulunabilir:

$$\Delta\tau_{ij}^k(t+1) = \begin{cases} \frac{1}{L^k(t+1)}, & (i, j) \text{ yolu kullanılmışsa} \\ 0, & \text{diğer durumlar} \end{cases}$$

$L^k(t + 1)$ karıncaların toplam tur uzunluğudur. Lokal feromon güncellemesi, turları dinamik olarak değiştirerek geçiş yapılan yolları cazip hale getirir. Karıncalar değişen feromon miktarlarına bağlı olarak her iterasyonda turlarını da değiştirmektedirler. Böylelikle sürekli olarak kısa turları bulmak amaçlanmaktadır.

KKA'da global feromon güncellemesi geçerli adımdaki en iyi sonuca sahip karıncanın izlediği yolun feromon düzeyinin arttırılmasından oluşur ve iterasyonlarda bulunan en iyi sonuçların belli bir oranda ileriki iterasyonlara aktarılmasını sağlar. Global feromon güncellemesi lokale benzer. Aşağıdaki denkleme göre yapılır:

$$\tau_{ij}(t + 1) = (1 - \rho) \cdot \tau_{ij}(t) + \Delta\tau_{ij}^k(t + 1)$$

$$\Delta\tau_{ij}(t + 1) = \begin{cases} \frac{1}{L_{en\ iyi}(t + 1)}, & (i, j) \text{ en iyi tura ait ise} \\ 0, & \text{diğer durumlar} \end{cases}$$

Burada $L_{en\ iyi}(t + 1)$ geçerli iterasyonda bulunan en iyi turun uzunluğudur.

3.2.2. Genetik Algoritma

Bilim adamlarının evrimsel yaklaşımı bilgisayar teknolojilerinde kullanmaya çalışması oldukça eskiye dayanmaktadır. Bilgisayar teknolojileri ve evrim yaklaşımının birleştirilmesi için ilk denemeler 1950'lerin sonunda ve 1960'ların başında yapılmıştır. Ancak, ilk denemeler o günlerdeki biyolojik evrim yaklaşımına uygun olarak mutasyon işlemine çok fazla bağımlı olmaları nedeniyle çok fazla başarılı olmamıştır.

1960'lı yılların ortalarında John Holland mutasyon işlemine eşleşme ve çaprazlama işlemlerini ekleyerek zor problemlerin doğal seçim prensibi doğrultusunda çözülebilmesine imkân sağlayacağına inandığı bir programlama yöntemi

geliştirmiştir. Bu çalışmasıyla Holland genetik algoritmalar alanının kurucusu sayılmıştır. Daha sonra dünyada bu alanda çalışan pek çok araştırmacı Holland'ın orijinal yöntemine dayanan pek çok değişik yöntem geliştirmişler ve pek çok farklı alana uygulamışlardır [4],[22],[30],[31],[36],[41],[42],[47].

Genetik Algoritma'nın döngüsü ise aşağıdaki gibidir:

ADIM 1: Gösterim (kodlama) yönteminin belirlenmesi.

ADIM 2: Başlangıç popülasyonunun (ilk nesil) oluşturulması.

ADIM 3: Başlangıç popülasyonundaki her bireyin amaç fonksiyonun hesaplanması.

ADIM 4: Yeni neslin oluşturulmasında kullanılacak bireylerin seçilmesi.

ADIM 5: Seçilmiş bireylere genetik işlemlerin uygulanarak yeni nesil elde edilmesi.

ADIM 6: Yeni neslin bireylerinin uygunluk fonksiyonlarının hesaplanması.

ADIM 7: Bitiş koşulu sağlanmamışsa 4. adıma dönülmesi.

ADIM 8: Bitiş koşulu sağlanmışsa en iyi bireyin sonuç olarak dönülmesi.

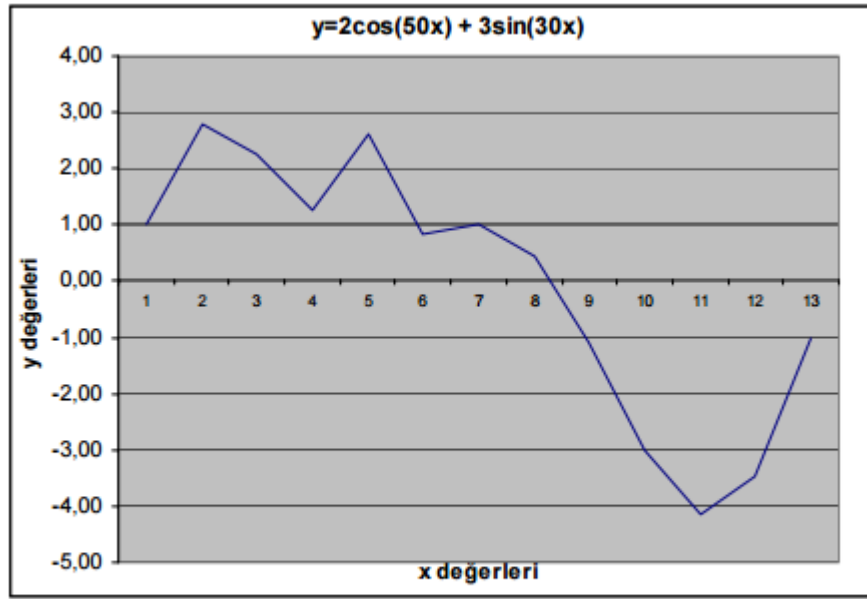
Genetik Algoritma'nın avantajları ise aşağıda maddelendiği gibidir:

- Çözüm uzayının tamamını değil, yalnızca bir kısmını taraması,
- Sürekli ve ayrık parametreleri optimize etmesi,
- Türevsel bilgiler gerektirmemesi,
- Amaç fonksiyonunu geniş bir spektrumda araştırması,
- Çok sayıda parametrelerle çalışma imkânı olması,
- Paralel PC'ler kullanılarak çalıştırılabilmesi,
- Karmaşık amaç fonksiyonu parametrelerini, lokal minimum veya maksimumlara takılmadan optimize edebilmesi,
- Sadece tek çözüm değil, birden fazla parametrelerin optimum çözümlerini elde edebilmesi olarak sıralanabilir.

Bunun bir uygulamasının örneğini ise tek değişkenli bir fonksiyonun optimizasyonu ile verebiliriz. Örnek fonksiyon;

$$y = 2 \cos(50x) + 3 \sin(30x) \quad 0 \leq x \leq 12$$

olarak grafiği aşağıdaki gibi olan bir trigonometrik fonksiyon olarak verilsin.



$F(x)=2\cos(50x) + 3\sin(30x)$ Grafiği

Amaç $[0,12]$ aralığında fonksiyonu minimize eden x^* değerini bulmaktır. Çözümü zor olmayan bu problem çeşitli yöntemlerle ve elle çözülebilir. Bu yöntemlerden biri ile çözüldüğünde, $x^* = 10,2$ 'ye karşılık gelen $f(x^*)$ değeri $-4,15$ olarak bulunmaktadır. Elde edilen değer, burada sadece Genetik Algoritma'yla yapılan çözümün sonucu ile karşılaştırma amacıyla hesaplanmıştır. Genetik Algoritma'yla fonksiyonun optimum değerini bulmak için ise Turbo Pascal'da kodlanmış Goldberg'in basit Genetik Algoritma kodu örneğe uyarlanarak kullanılmıştır. Problemin çözümü için ikili kodlama kullanılmış ve kromozomlar 30 bitlik dizilerden oluşturulmuştur. Kromozom uzunluğu genellikle gerçek uygulamalarda uzun alınmaktadır.

İlk popülasyon oluşturulmuş, minimum değer olarak -4,9039264020 elde edilmiştir. Kullanılan Genetik Algoritma’da, çaprazlama ve mutasyon olmak üzere iki temel genetik operatör kullanılmıştır. Parametreler ise şöyle seçilmiştir:

Popülasyon büyüklüğü	: 100
Çaprazlama olasılığı	: 0,70
Mutasyon olasılığı	: 0,01
Durdurma Kriteri	: 15 kuşak
Seçim stratejisi	: Kuşaksal Strateji

Deney sonuçları: İlk popülasyon yaratıldıktan sonra program, pentium 120 mHz bir PC’de çalıştırılmıştır. Hesaplama zamanı 13 saniye olarak ölçülmüştür. Aşağıdaki tabloda iyileşmenin görüldüğü kuşaklar ve minimum değerler özetlenmiştir. [53]

Kuşak Sayısı	Minimum Uygunluk Değeri
0	-4,9039264020
1	-4,6593255932
2	-4,5596444645
9	-4,1082673943
12	-4,1131100874

Genetik Algoritma’nın çalıştırılması sonucunda 12. kuşakta minimum değer bulunmuştur. Bu değer, optimal çözüme çok yakın olup kısa bir sürede elde edilmiştir. Kullanılan parametreler ise, yapılan çeşitli deneysel çalışmalara ve literatürde karşılaşılan benzer örneklerle dayanılarak belirlenmiştir.

3.2.3. Tabu Arama Algoritması

Tabu Arama Algoritması, optimizasyon problemlerinin çözümü için F. Glover tarafından geliştirilmiş iteratif bir araştırma algoritmasıdır. Temel yaklaşım, son çözüme götüren adımın dairesel hareketler yapmasını önlemek için bir sonraki döngüde tekrarın yasaklanması veya cezalandırılmasıdır. Böylece yeni çözümlerin incelenmesiyle Tabu Arama Algoritması, bölgesel en iyi çözümün daha ilerisinde bulunan çözümlerin araştırılabilmesi için bölgesel-sezgisel araştırmaya kılavuzluk etmektedir [5],[37],[38].

Tabu Arama Algoritması'nın bölgesel optimalliği aşmak amacıyla kullandığı temel prensip, değerlendirme fonksiyonu tarafından her iterasyonda en yüksek değerlendirme değerine sahip hareketin bir sonraki çözümü oluşturmak amacıyla seçilmesine dayanmaktadır. Bunu sağlamak amacıyla bir tabu listesi oluşturulur, tabu listesinin orijinal amacı önceden yapılmış bir hareketin tekrarından çok tersine dönmesini önlemektir.

Tabu listesi kronolojik bir yapıya sahiptir ve esnek bir hafıza yapısı kullanır. Tabu Arama Algoritması her ne kadar istenmeyen noktaların işaretlenmesi olarak açıklanmış olsa da daha cazip noktaların işaretlenmesi olarak da kullanılır.

Bu algoritma ile ilgili matematiksel bir uygulama ortaya konacak olursa, aşağıdaki fonksiyonun genel minimum noktasının; rastgele bir noktadan aramaya başlanarak bulunması örnek olarak verilebilir:

$$f(x) = \begin{cases} x^2, & x \leq 1 \\ (x - 3)^2 - 3, & \text{değilse} \end{cases}$$

Söz konusu fonksiyonun en küçük değeri $f(3) = -3$ olur. Rastlantısal bir noktadan aramaya başlanacaktır. Aramaya $f(-7,39) = 54,61$ noktasından başlanmaktadır. Dört farklı komşu seçme fonksiyonu vardır. ($\mu \in [0,1]$ rastlantı bir sayı) Örnek TA Algoritması'nın temel işleyişini canlandırmak için verildiğinden yalnız yakınlığa

bağlı bellek yapısı kullanılmıştır. Ters hareketlerin yapılması tabu belirlenmiştir. Basit bir örnek olduğundan tabu süresi 1 olarak alınmıştır. Her bir hareketin tabu listesi h_x ile temsil edilmiştir.

Denklemden $f(x)$ 'in küçük olduğu noktayı arayan bir TA Algoritması:

t	i	$i = \varphi_x(i)$	$f(i)$	$f(i)$	μ	x	h_1	h_2	h_3	h_4
0	-7,39	-7,75	54,61	60,20	0,18	3	0	0	0	0
1	-7,39	-5,71	54,61	32,55	0,42	2	2	0	0	0
2	-5,71	-6,13	32,55	37,60	0,21	3	2	0	0	3
3	-5,71	-2,36	32,55	5,56	0,84	2	4	0	0	3
4	-2,36	-3,09	5,56	9,56	0,37	3	4	0	0	5
5	-2,36	-3,97	5,56	15,73	0,80	3	6	0	0	5
6	-2,36	-3,98	5,56	15,84	0,81	3	7	0	0	5
7	-2,36	-4,44	5,56	19,74	0,52	4	8	0	0	5
8	-2,36	-6,19	5,56	38,37	0,96	4	8	0	0	5
9	-2,36	-3,82	5,56	14,63	0,73	3	8	9	0	5
10	-2,36	-0,12	5,56	0,02	0,56	2	11	10	0	5
11	-0,12	-0,37	0,02	0,14	0,12	3	11	10	0	12
12	-0,12	2,41	0,02	-2,65	0,63	2	13	10	0	12
13	2,41	1,39	-2,65	-0,41	0,51	3	13	10	0	14
14	2,41	2,23	-2,65	-2,40	0,09	3	15	10	0	14
15	2,41	2,48	-2,65	-2,73	0,02	2	16	10	0	14
16	2,48	2,12	-2,73	-2,22	0,18	3	16	10	0	17
17	2,48	2,26	-2,73	-2,45	0,06	4	18	10	0	17

Fonksiyonda $f(3) = -3$ noktası en küçük değerli noktadır. Görüldüğü gibi TA Algoritması 18 çevrimde bu noktaya yaklaşmıştır. TA yerel en küçük olan $f(0) = 0$ değerine 11. çevrimde gelmiş 12. çevrimde $x = 2$ hareketiyle yerel en küçük noktasından kurtulmuştur.

3.2.4. Tavlama Benzetimi Algoritması

Tavlama Benzetimi (TB), büyük boyutlu kombinatoriyal optimizasyon problemleri için iyi çözümler veren olasılıklı bir arama yöntemidir. 1980'lerin başlarında Kirkpatrick ve diğerleri (1983) ve Cerny (1985) birbirlerinden bağımsız olarak tavlama kavramlarını kombinatoriyal optimizasyon problemlerinde kullanmaya başlamışlardır [6],[32].

Tavlama Benzetimi ismi, katıların fiziksel tavlama süreci ile olan benzerlikten ileri gelmektedir. TB'yi belli bir kombinatoriyal optimizasyon problemine uyarlamak için verilmesi gereken bir takım kararlar vardır. Bunlar; probleme özgü kararlar, tavlama planına özgü kararlardır.

TB Algoritması için probleme özgü kararlar, problemin çözüm kümesinin, bir çözümünden diğerine geçerken kullanılacak hareket mekanizmasının ve en iyilenecek amaç fonksiyonunun belirlenmesini kapsamaktadır. Çözüm uzayı, parça seçim ve makina yükleme problemi için en kritik kararlardan biri hangi takımların hangi makinalara atanacağını belirlemesidir. Makinaların takım magazini yükleri, makinalar tarafından gerçekleştirilebilecek işlemleri, dolayısıyla da işlemlerin atanabileceği makinaları sınırlar.

Bu çalışmada önceliğe sahip olan problemimiz parça seçimi olduğundan ve her parçanın sadece bir takım seti ile işlenebildiğini varsaydığımızdan, makinaların mümkün olduğunca farklı takımlarla donatılması istediğimiz sonucu almamıza yardımcı olacaktır. Bu nedenle problemin çözümü, ilgili aletin hangi makinaya atandığını gösteren t boyutlu bir vektörle tanımlanmıştır. Bu çözüm, her takımın sadece bir makinaya atanmasını gerektirmektedir. Böylece, her işlemin atanabileceği bir makina olacaktır ve takım magazini yükleri ile makinaların iş yükleri doğrudan belirlenebilecektir.

Amaç fonksiyonu, modelin sıralı olarak en iyilenmesi gereken iki amacı vardır. Problemin çok amaçlı yapısını temsil etmek için amaçlar ilk önce oransal hale

getirilmiş, sonra da bütünleşik bir fonksiyon olarak ifade edilmiştir. Ayrıca, uygun olmayan çözümleri de arama prosedürüne dâhil etmek için mevcut kısıttaki uygunsuzluk miktarını

$$\left(\sum_{m=1}^M \left(\sum_{t=1}^T S_t \cdot R_{t,m} - C_m \right) \right)$$

ile dikkate alan bir ceza fonksiyonu tanımlanmıştır. Bütünleşik amaç fonksiyonu ise aşağıdaki gibi belirlenmiştir:

$$\max Z = \gamma \cdot \frac{(\sum_{i=1}^I W_i \cdot Y_i) - \alpha \cdot (\text{toplam uygunsuzluk})}{\sum_{i=1}^I W_i} - \frac{\sum_{m=1}^M (U_m + O_m)}{\sum_{i=1}^I T_i \cdot Y_i}$$

Bu formülün bir örneğinin uygulaması için izlenilen algoritma döngüsü ise aşağıdaki gibi sıralanmaktadır:

ADIM 1: Başlangıç değerlerinin verilmesi. ($T_b, T_f, r, M(t), \max it$)

ADIM 2: Başlangıç çözümü x 'i üret, mevcut ve en iyi çözüm olarak kaydet.

ADIM 3:

3.1: Sıcaklık değişkenini başlangıç sıcaklığına eşitle. ($Temp = T_b$)

3.2: Gerçekleşen hareket ve toplam hareket sayaçlarını sıfırla.

3.3: Eğer $Temp < T_f$ ise ADIM 4'e git.

3.4: x 'e komşu x^* çözümünü rassal olarak üret ve $f(x)$ 'i hesapla.

3.5: Eğer $\exp[f(x^*) - f(x)/Temp] > U(0,1)$ ise komşu çözümü kaydet.

Aksi halde 3.6'ya git. Eğer $f(x) > best_f$ ise en iyi çözümü kaydet.

3.6: $iterasyon = iterasyon + 1$; eğer $iterasyon > max_{it}$ ise 3.3'e git.

ADIM 4: En iyi çözümü yazdır.

ADIM 5: DUR

3.2.5. Yapay Sinir Ağları Algoritması

Yapay sinir ağları (YSA), insan beyninin özelliklerinden olan öğrenme yolu ile yeni bilgiler türetebilme, yeni bilgiler oluşturabilme ve keşfedebilme gibi yetenekleri, herhangi bir yardım almadan otomatik olarak gerçekleştirebilmek amacı ile geliştirilen bilgisayar sistemleridir.

Yapay sinir ağları; insan beyninden esinlenerek, öğrenme sürecinin matematiksel olarak modellenmesi uğraşı sonucu ortaya çıkmıştır. Bu nedenledir ki, bu konu üzerindeki çalışmalar ilk olarak beyni oluşturan biyolojik üniteler olan nöronların modellenmesi ve bilgisayar sistemlerinde uygulanması ile başlamış, daha sonraları bilgisayar sistemlerinin gelişimine de paralel olarak birçok alanda kullanılır hale gelmiştir [52].

İnsan beyninin çalışma prensibini taklit ederek çalışan bu sistemler, her ne kadar bilgisayar teknolojisi hızlı bir gelişim göstermiş, işlem hızları nano saniyeler mertebesine inmiş olsa da, bırakalım insan beynini, ilkel bir canlı beyninin fonksiyonları dahi baz alındığında, böyle bir organizmanın yanında çok ilkel kalmaktadır. Nanosaniyeler bazındaki işlem hızları ile YSA'lar, milisaniyeler mertebesindeki işlem hızları ile işlem yapan insan beyninin işlevselliğinin henüz çok uzaktadır. YSA'nın avantajları aşağıdaki gibi sıralanabilir:

- YSA'lar birçok hücreden meydana gelir ve bu hücreler eş zamanlı olarak çalışarak karmaşık işlevleri yerine getirir. Diğer bir deyişle karmaşık işlevler birçok hücrenin eş zamanlı çalışması ile meydana getirilir. Süreç içerisinde bu hücrelerden herhangi biri işlevini yitirse dahi sistem güvenli bir şekilde çalışmasına devam edebilir.
- Eğitim esnasında kullanılan nümerik bilgilerden, problemin genel özellikleri elde etmesi ve böylelikle eğitim sırasında kullanılmayan girdiler için de, anlamlı yanıtlar üretebilmesidir.
- Yapı üzerinde dağılmış belli tipteki non-lineer alt birimler, non-lineer problemlerin de çözümünü mümkün kılmaktadır.

- YSA'lar makina öğrenmesi gerçekleştirebilirler. Yapay sinir ağlarının temel işlevi zaten bilgisayarın öğrenmesini sağlamaktır. Olayları öğrenerek benzer olaylar karşısında mantıklı kararlar verebilirler.
- Bilgi işleme yöntemleri geleneksel programlamadan farklıdır. Bu nedenle geleneksel programlamanın getirdiği birçok olumsuzluk ortadan kaldırılabilir.
- Bilgiler ağın tamamında saklanır. Geleneksel programlamada olduğu gibi bilgiler veri tabanları ya da dosyalarda belli bir düzende tutulmaz, ağın tamamına yayılarak değerler ile ölçülen ağ bağlantılarında saklanmaktadır. Hücrelerden bazılarının işlevini yitirmesi, anlamlı bilginin kaybolmasına neden olmaz.
- Dağıtık belleğe sahiptirler. YSA'larda bilgi ağa dağılmış bir şekilde tutulur. Hücrelerin bağlantı ve ağırlık dereceleri, ağın bilgisini gösterir. Bu nedenle tek bir bağlantının kendi başına anlamı yoktur.
- Örnekleri kullanarak öğrenirler. YSA'nın öğrenebilmesi için örneklerin belirlenmesi, bu örneklerin ağa gösterilerek istenen çıktılara göre ağın eğitilmesi gerekmektedir. Ağın başarısı, seçilen örnekler ile doğru orantılıdır, ağa olay bütün yönleri ile gösterilemezse ağ yanlış çıktılar üretebilir.
- Daha önce görülmemiş örnekler hakkında bilgi üretebilirler. YSA'lar eğitimleri sırasında kendilerine verilen örneklerden genellemeler çıkarırlar ve bu genellemeler ile yeni örnekler hakkında bilgi üretebilirler.
- Algılamaya yönelik olaylarda kullanılabilirler. YSA'ların en başarılı oldukları alanlar, algılamaya yönelik uygulama alanlarıdır. Bu alanlarda başarıları kanıtlanmıştır.
- Örüntü (pattern) ilişkilendirme ve sınıflandırma yapabilirler. YSA'lar kendilerine örnekler halinde verilen örüntüleri kendisi veya diğerleri ile ilişkilendirebilir. Ayrıca kendisine verilen örneklerin kümelenmesi ile, bir sonraki verinin hangi kümeye dâhil olacağının karar verilmesi konusunda kullanılabilirler.
- Örüntü tamamlama yapabilirler. Ağa eksik bilgileri içeren örüntüler verildiğinde eksik bilgilerin tamamlanması konusunda başarılıdırlar.
- Kendi kendine öğrenebilme ve organize etme yetenekleri vardır. YSA'lar çevrimiçi olarak öğrenebilirler ve kendi kendilerini eğitebilirler.

Matematiksel bir örnek vermek gerekirse, ‘‘Yapay Sinir Ağları Algoritması’’nın uygulandığı aktivasyon fonksiyonları çeşitlerinden biri olan Sigmoid Fonksiyonu, unipolar bir eğri için;

$$\varphi(v) = \frac{1}{1 + e^{-\beta v}} = \frac{1}{2} \cdot (\tanh\left(\frac{\beta v}{2}\right) + 1)$$

bipolar bir eğri için ise;

$$\varphi(v) = \tanh(\beta v)$$

şeklinde yazılıp burada β parametresi fonksiyonun eğimini kontrol eder. Sigmoid fonksiyonları, aktivasyon fonksiyonu seçimi için en popüler fonksiyonlardır.

3.3. Topolojik Optimizasyon Uygulaması

Bilgisayar ağları, n tane düğüm içeren V düğümler kümesi ile tane hat içeren E hatlar kümesinden oluşan bir ağıdır ve $G = (V, E)$ fonksiyonu ile gösterilebilir. Düğümler ağıdaki bilgisayarların bulundukları noktaları ve hatlar ise bu bilgisayarlar arasındaki iki ya da tek yönlü hatları ifade etmektedir. Bu çalışmada, iki yönlü hatların kullanıldığı bir bilgisayar ağının güvenilirlik kısıtı altında en küçük maliyetli topolojik tasarımı ile ilgilenilmektedir. Problemin matematiksel modeli aşağıdaki gibidir:

$$f(x) = \sum_{i=1}^{n-1} \sum_{j=i+1}^n c_{ij} \cdot x_{ij} \quad Rel_0(x) \geq Rel_0$$

Ayrıca bu modeldeki her çözümün en azından 2-bağlı bir ağ olması dikkate alınan bir ön kısıttır. 2-bağlı ağlarda, her bir düğüm ağı en az iki farklı yol ile bağlanmaktadır. Böylece bilgisayar ağındaki her bir düğüm noktasından en az iki adet bağlantı hattı geçmektedir.

Burada;

$$x(i,j) = \begin{cases} 1, & (i,j) \text{ düğüm çifti arasında hat varsa} \\ 0, & \text{diğer durumlar} \end{cases}$$

x : x_{ij} hatlarından oluşan aday ağı

$f(x)$: x aday ağının toplam maliyeti

c_{ij} : (i,j) düğüm çifti arasındaki hattın toplam maliyeti

$Rel(x)$: x aday ağının güvenilirlik değeri

Rel_0 : istenilen güvenilirlik kısıtı

şeklindedir. Bu problemde dikkate alınan varsayımlar ise şunlardır;

- Ağdaki tüm düğümlerin yeri bilinmektedir ve düğüm arızaları dikkate alınmamaktadır.
- Ağdaki tüm c_{ij} değerleri bilinmektedir.
- Ağdaki bir hattın çalışma olasılığı p 'dir ve her hat eşit çalışma olasılığına sahiptir.
- Her hat iki yönlüdür.
- Hat arızalanmaları birbirinden istatistiksel olarak bağımsızdır.
- Tüm düğüm çiftleri arasında birbirlerine doğrudan bağlantı yapılabilir.
- Tüm terminal güvenilirliği dikkate alınmıştır.

3.3.1. Uygulamanın Algoritması

Bu bölümde bilgisayar ağlarının tasarımı için önerilen “Melez Karınca Kolonisi Algoritması” incelenmektedir. Geliştirilen algoritma temel olarak beş adım halinde özetlenebilir. Bu adımlar; başlangıç değerleri, karınca sistemi ile bir GSP başlangıç çözümünün oluşturulması bulunan en iyi GSP çözümünün seçilmesi, topolojide istenilen güvenilirlik düzeyinin sağlanmasıdır. [53] Algoritma yapısı şu şekildedir:

ADIM 1: Başla;

Feromon ve sezgisel matrislerine başlangıç değerlerini ata;

$$\mu_{ij} = 1/d_{ij}, \tau_{ij} = \tau_0 = m/C^n ;$$

$$T = 0;$$

ADIM 2: Karınca Sistemi ile başlangıç GSP çözümün oluştur;

$$T \leftarrow 1;$$

tekrar;

$$s \leftarrow 1;$$

tekrar;

$$l \leftarrow 1;$$

tekrar;

p_{ij}^l olasılık değerini hesapla;

Rulet Çemberi Algoritması'nı kullanarak karıncanın seçeceği bir sonraki düğümü bul;

$$l = l + 1 ;$$

kadar ($l = m$) ;

Feromon buharlaşmasını sağla;

Karıncaların sisteme feromon eklemesini sağla;

$$s = s + 1 ;$$

kadar ($s = N$) ;

C^l değerini hesapla;

$$T = T + 1 ;$$

kadar ($T = T^k$) veya ($C^l = GSP \text{ en iyi değer}$) ;

ADIM 3: Bulunan en iyi çözüme (G_{aday}) adım 4'ü uygula;

ADIM 4: Topolojide istenilen sistem güvenilirlik seviyesini (R_0) sağla;

$$T \leftarrow 1 ;$$

tekrar;

$$l \leftarrow 1;$$

tekrar;

Aday ağ topolojisinin güvenilirliğini [10] üst sınır yaklaşımına göre hesapla ($R(G_{aday}^l)$);

eğer ($R(G_{aday}^l) \geq R_0$) ise;

tekrar;

Turnuva seçimi mekanizmasıyla topolojide olmayan μ_{ij} değerine göre en iyi hattı ekle;

Yeni aday çözümlerin güvenilirliğini [10] üst sınırına yaklaşımına göre hesapla ((G_{aday}^l));

kadar ($R(G_{aday}^l) \geq R_0$);

Eğer bulunan l . çözümün amaç fonksiyonu değeri ($f(G_{aday}^l)$) daha önce bulunan $f(G_{aday}^l)$ değerlerinden daha küçük ise MC simülasyonu ile ağın güvenilirliğini ($R(G_{aday}^l)_{MC}$) tahmin et;

eğer ($R(G_{aday}^l) < R(G_{aday}^l)_{MC}$) ise;

tekrar;

Turnuva seçimi mekanizmasıyla topolojide olmayan μ_{ij} değerine göre en iyi hattı ekle;

Yeni aday çözümün güvenilirliğini MC ile tahmin et ($R(G_{aday}^l)_{MC}$);

kadar $R(G_{aday}^l)_{MC} \geq R_0$;

Eğer aday topolojinin amaç fonksiyonu değeri ($f(G_{aday}^l)$) daha önce bulunan değerlerden küçük ise;

$$G_x = G_{aday}^l ;$$

$$l = l + 1 ;$$

kadar ($l = m$);

$$T = T + 1 ;$$

kadar ($T = T^l$) veya ($G_x = en\ iyi$);

ADIM 5: Dur.

3.3.2. Algoritmanın Bileşenleri

Melez Karınca Kolonisi Algoritması'nda, oluşturulan başlangıç değerleri ve kullanılan diğer yardımcı yöntem ve metotlar aşağıdaki başlıklar altında toplanabilir.

3.3.2.1. Algoritmanın Başlangıç Değerleri

Melez Karınca Kolonisi Algoritması'nın ilk bölümünde kullanılan Karınca Sistemi Algoritması için başlangıç değerleri, Dorigo'nun önerdiği algoritmaya ve değerlere sadık kalınarak alınmıştır.

Karınca Sistemi meta sezgiselinde, diğer sezgisellerde olduğu gibi tekrar sayısının belirlenmesi gerekmektedir (T^k). Tekrar sayısı attıkça algoritmanın bulduğu sonuçlarda da bir iyileşme gözlenmektedir. Bu çalışmada, kolay karşılaştırma yapılabilmesi ve algoritmanın iyi sonuçlara ulaşabilmesi için, algoritmanın her iki aşaması için de 100 tekrar yapılmaktadır.

3.3.2.2. Turnuva Seçim Mekanizması

Turnuva seçim mekanizması, özellikle Genetik Algoritma içerisinde uygunluk fonksiyonu yüksek olan bireylerin seçilerek bir sonraki nesle aktarılmasında kullanılmaktadır. Bu çalışmada ise, oluşturulan başlangıç ağ tasarımının istenilen güvenilirlik düzeyine çıkartılabilmesi için, eklenecek hatların seçilmesi işleminde kullanılmaktadır. Turnuva seçim mekanizmasının algoritması aşağıda verilmiştir.

ADIM 1: Başla;

ADIM 2: Ağ tasarımına eklenmiş hatlardan n tane seç;

ADIM 3: Seçilen hatlardan en küçük maliyetli bir hattı tasarıma ekle;

ADIM 4: Dur.

3.3.2.3. Güvenilirlik Hesabında Üst Sınır Yaklaşımı

Oluşturulan bilgisayar ağının güvenilirliğinin belirlenmesi için aşağıda verilen Jan'ın üst sınır yaklaşımı [10] tercih edilmiştir. Bunun sebebi, ağın gerçek tüm terminal güvenilirliğinin hesaplanmasının NP-zor bir problem olması ve problem büyüklüğüne göre çözüm zamanının üstel olarak artmasıdır. Ayrıca üst limit yaklaşımı işlem zamanı açısından hızlı çalışan bir yapıya sahip olduğu için, büyük boyutlu problemlerde algoritmanın genelini yavaşlatmamaktadır.

$$R(G) \leq 1 - \left\{ \sum_{i=1}^n q^{d_i} \cdot \prod_{k=1}^{m_i} (1 - q^{d_k-1}) \cdot \prod_{k=m_i+1}^{i-1} (1 - q^{d_k}) \right\}$$

Burada;

$R(G)$: Ağın güvenilirliği,

n : Düğüm sayısı,

m_i : En küçük ($d_i, i - 1$), tüm i değerleri için,

q : Ağdaki hatların bozulma olasılığı;

d_i : i . düğümün bağlılığının derecesidir.

3.3.2.4. Monte Carlo Benzetimi

Oluşturulan bilgisayar ağının güvenilirliğinin tahmin edilmesinde kullanılan bir diğer metot da Yeh vd. [54] tarafından önerilen Monte Carlo simülasyonudur. Bu simülasyon yönteminin kullanılmasındaki temel sebep, üst limit ile tahmin edilen ağ güvenilirliği için daha kesin bir sonuç elde edebilmektir. Monte Carlo simülasyonu üst limit yaklaşımına göre işlem zamanı açısından daha yavaş çalışan bir yapıya sahiptir.

3.3.2.5. Algoritmanın Temel Yapısı

Bilgisayar ağları için geliştirilen Melez Karınca Kolonisi Algoritması iki aşamadan oluşmaktadır. Birinci aşama başlangıç çözümünün oluşturulduğu Karınca Sistemi Algoritması'dır. İkinci aşama ise, oluşturulan başlangıç çözümünün istenilen güvenilirlik düzeyine çıkartılması ve güvenilirliğin kontrol edilmesidir. Birinci aşamada Dorigo'nun önermiş olduğu Karınca Sistemi Algoritması üzerinde herhangi bir değişiklik yapılmadan kullanılmıştır. [55]

Birinci aşamada Karınca Sistemi'nin kullanılmasının sebebi, başlangıç çözümü olarak elde edilmek istenilen GSP çözümünü, büyük boyutlu ve matematiksel modelleme yöntemi ile çözümü bulunamayan veya çok uzun zaman alan problemler için, hızlı ve etkin bir şekilde bulabilmesidir. İkinci aşamada ise, ilk aşamada bulunan en iyi GSP çözümü seçilir.

Seçilen topolojiye, istenilen güvenilirlik düzeyine ulaşmaya kadar hat ekleyerek sonuç ağ topolojisi elde edilmektedir. Bu çalışmada oluşturulan Melez Karınca Kolonisi Algoritması'nda elde edilen ağın 2-bağlılığı için herhangi bir ek kontrol mekanizmasına ihtiyaç duyulmamıştır. Bunun sebebi, oluşturulan başlangıç GSP çözümünün 2-bağlılık koşulunu sağlamasıdır.

Algoritmanın içerisinde, elde edilen topolojiden hat çıkarılmadığı için de oluşturulan başlangıç çözümündeki 2-bağlılık kısıtı bozulmamaktadır. Algoritmayı adım adım inceleyecek olursak, birinci adımda algoritmanın başlangıç değerleri belirlenmektedir. Başlangıç değerleri ile ilgili bilgi 3.3.2.1'de anlatılmaktadır. İkinci adımda, Karınca Sistemi kullanılarak başlangıç çözümü üretilmektedir.

Algoritmanın nasıl çalıştığı çalışmanın 3.2.1 bölümünde detaylı olarak anlatılmaktadır. Burada elde edilmek istenilen GSP en iyi çözümdür. Algoritmanın ikinci adımının sonlanması için iki koşuldan sadece birisini sağlaması yeterlidir. Bu koşullar; tekrar sayısına ulaşılması veya GSP en iyi çözümün bulunmasıdır.

Büyük boyutlu problemlerde GSP en iyi çözümü bilinmediğinden dolayı algoritmanın istenilen tekrar sayısına ulaşması yeterli olacaktır. İkinci adımda her bir iterasyonda karınca sayısı kadar GSP çözümü elde edilmektedir. Bulunan çözümlerin her birisinin amaç fonksiyonu değeri incelenmekte ve iterasyonun en iyi amaç fonksiyonu değerine sahip çözümü daha önce bulunan en iyi amaç fonksiyonu değerleriyle karşılaştırılır. Eğer daha önce bulunan çözümden daha iyi amaç fonksiyonu değerine sahip bir GSP çözümü elde edildi ise bulunan bu çözüm bulunabilen en iyi GSP çözümü olarak saklanmaktadır.

Üçüncü adımda, ikinci adımda bulunan en iyi çözüm seçilmektedir. Melez Karınca Algoritması'nın ilerleyen aşamasında seçilen bu çözüm başlangıç çözümü olarak kullanılacaktır. İlerleyen adımlar sadece tek bir başlangıç çözümü üzerinden sonuca ulaşmaya çalışmaktadır.

Dördüncü adımdaki amaç başlangıç çözümünde elde edilen ağın istenilen güvenilirlik düzeyine ulaşacak şekilde hatların eklenmesidir. Bunun için Genetik Algoritma'da kullanılan Turnuva Seçim Mekanizması ile başlangıç çözümün hat ekleme yöntemi kullanılmaktadır. Bu adım da sezgisel bir yöntem içerdiği için, tekrarlı olarak uygulanmaktadır. Adımın başlangıcında tekrar sayısı olan (T) değeri sıfıra eşitlenmektedir. Bu adımda durma koşulu istenilen tekrar sayısına ulaşılması veya ağ için ulaşılabilecek en iyi değerin bulunmasıdır.

Karınca kolonisinde kullanılan karınca sayısı dördüncü adımda da kullanılmaktadır. Amaç yarışmacı bir yaklaşım izleyerek problemin çözümüne daha hızlı ulaşabilmektir. Her iterasyonda karınca sayısı kadar aday çözüm oluşturulmaktadır.

Eğer hesaplanan yaklaşık sistem güvenilirliği üst sınırı, istenilen sistem güvenilirliğinden daha düşük ise ağa Genetik Algoritma'da kullanılan bir seçim mekanizması olan Turnuva Seçimi ile yeni bir hat eklenmektedir. Yeni hattın eklenmesinde hatların sezgisel matrisindeki değerlerine bakılmaktadır. Daha önce seçilmemiş hatların arasından rasgele hatlar seçilmekte ve bu hatlardan en küçük değere sahip olan hat ağa eklenmektedir. Daha sonra ağın tekrar güvenilirliği

hesaplanmaktadır. Bu işlem Jan'ın [10] üst limiti ile hesaplanan güvenilirlik düzeyi istediğimiz güvenilirlik düzeyine eşit veya daha fazla oluncaya kadar tekrarlanmaktadır.

Karıncalar tarafından temsil edilen tüm aday çözümlerin Jan'ın [10] üst limit yaklaşımı ile güvenilirliği hesaplandıktan sonra, amaç fonksiyonu değeri en küçük olan aday çözüm seçilmektedir. Seçilen bu çözümün amaç fonksiyonu değeri, daha önceki çevrimlerde bulunan ağ topolojisinin değerinden daha düşük ise, aday ağ tasarımının Monte Carlo benzetimi ile güvenilirlik tahmini yapılmaktadır.

Aday ağın Monte Carlo benzetimi sonucunda elde edilen güvenilirlik düzeyi istenilen güvenilirlik düzeyinden düşük ise yine turnuva seçim mekanizması yardımı ile ağa yine sezgisel matris değerleri kullanılarak yeni bir hat eklenir ve Monte Carlo benzetimi tekrarlanır. Bu işlem aday ağın güvenilirlik düzeyi, istenilen güvenilirlik düzeyine eşit veya daha yüksek oluncaya kadar tekrarlanır.

Beşinci adımda ise algoritma bulabildiği en iyi çözümü üretmiştir ve algoritma sonlanmaktadır. [53]

4. ŞEHİR YAPILANMALARININ TOPOLOJİK ANALİZİ

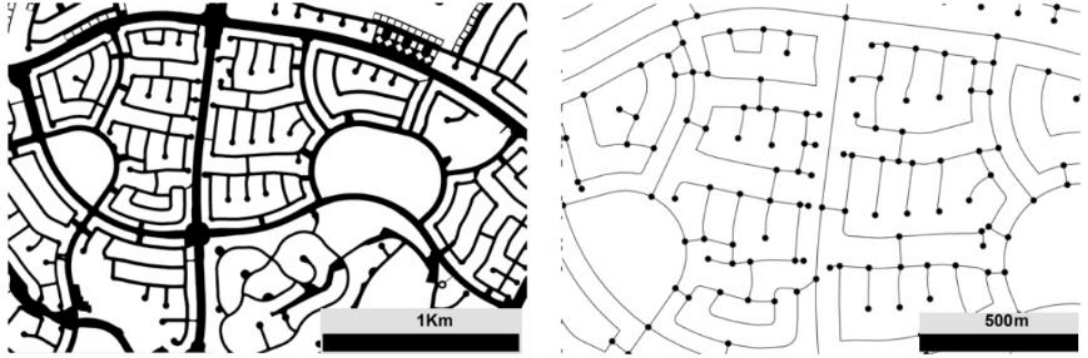
İdeal bir şehir yapılanması şekli var mıdır? Bu soru, şehirler küresel çapta genişlemeye devam ederken sürdürülebilirlik ve ekonomik açıdan uygunluk bağlamında milyarlarca insanı ilgilendiren temel bir konunun çözümüne dair bir sorgulamadır.

Yabancı araştırmacılar bu sorunun cevabını, matematiksel analizler ve teoremler bütünüyle yanıtlamaya dair çalışmalarda bulunmuştur. Bu çalışmaların içinde en belirgin şekilde matematiksel açıklamaları teoremlerle destekleyerek ortaya koyanlardan biri olan Christa Brelsford, Taylor Martin, Joe Hand and Luis M. A. Bettencourt [56] bu soruya geometrik olmasa da topolojik kanıtlarla cevap vermişlerdir. Çalışmanın bu bölümünde bu çalışmayla alakalı detaylı açıklamalar ve Türkçesi mevcut olmayan öğretilerin çevirileri yer alacaktır. Mevzubahis çalışmanın tek cümlelik özet kısmında yazan “Tüm şehirler topolojik olarak aynı özellikleri paylaşırlar öyle ki iki şeritli bir yolun; kademeli olarak gelişen bir mahalleye dönüşümü gibi.” önermesini anlamaya ve anlatmaya çalışılacaktır.

4.1. Ulaşım Sistemlerinin Topolojisi

Tüm yerleşim alanlarının analizi iki temel başlık altında incelenebilir. Birincisi tüm yolları ve patikaları içeren ulaşım sistemleri, ikincisi ise tüm binaları ve ortak kullanım alanlarını içeren mekânlar. Bu başlıkta biz ulaşım sistemlerini inceleyeceğiz. Bu mekânları temsil eden uzaylar arasındaki ilişkileri ve bunların birbirine topolojik olarak özdeşliklerinin ispatlarını ilerleyen başlıklarda vereceğiz.

Kentsel ulaşım sistemi, herhangi bir şehrin patika ve yollarının fizikî yapısının toplam hacmidir. Bunun şematik bir örneği, aşağıda Las Vegas’ın ufak bir parçasının kentsel ulaşım sistemi çalışılırken kullanılan kroki ile verilmiştir.

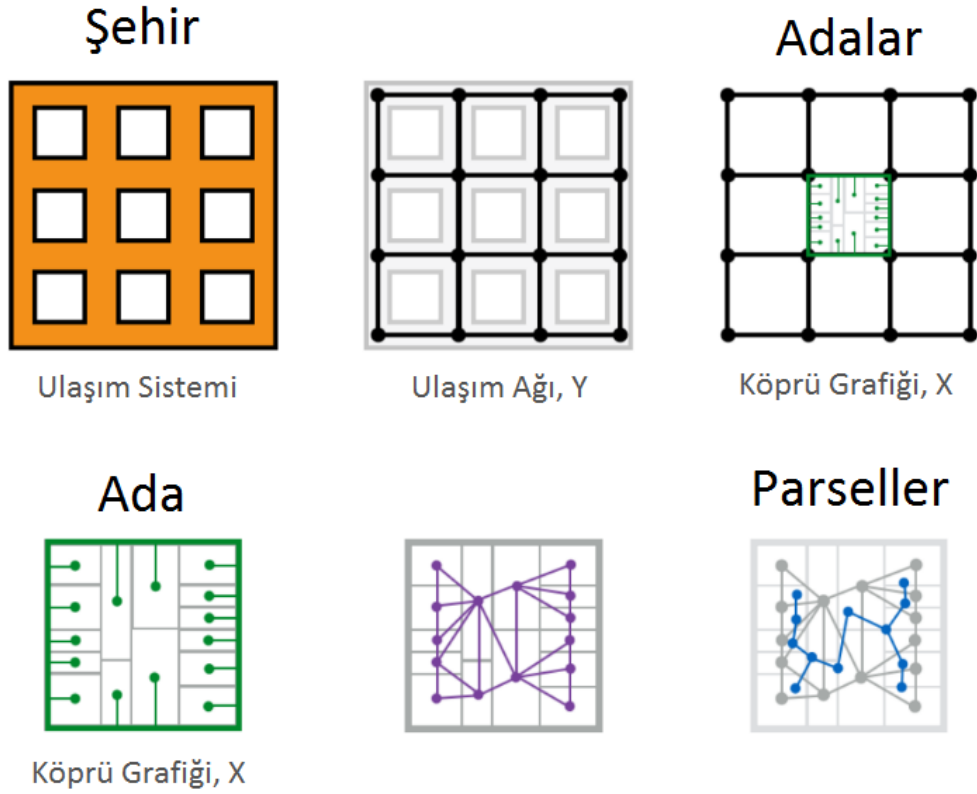


Her şehrin ulaşım sistemi her zaman kompakt, bağlantılı, yönlendirilebilir 2-boyutlu yüzeylerdir. Bağlantılı kelimesiyle kastedilen bu yüzeyde seçilen herhangi bir noktaya, yüzeydeki başka bir noktadan yüzey boyunca hareket ederek ulaşım sağlanabileceğidir. Yönlendirilebilir kelimesiyle anlatılmaya çalışılan genel manada yön tanımlamasının (örneğin; aşağıdan yukarıya) yapılabilmesidir. Bu demek oluyor ki bu türden yüzeyler, yönlendirilemeyen “Möbius Şeridi” gibi yüzeylerle herhangi bir benzerlik göstermezler. Kompaktlık ya da “Tıkızlık” ise daha çok yüzeylerin topolojik sınırlılığına atıfta bulunan soyut bir tanımlamadır. [57]

Uygulamada, bir yüzey ancak ve ancak sonlu sayıda üçgen kullanılarak üçgenleştirilebiliyorsa kompakttır. [58] Bu pratikte demek oluyor ki ulaşım sistemlerinin oluşturduğu bir yüzeyi belli sayıda üçgene bölebiliriz. Bunu daha büyük ya da daha küçük üçgenler kullanarak yapabiliriz. Fakat her durumda üçgen sayısı sonlu olmalıdır. Sonuç olarak ulaşım sisteminin 2-boyutlu karakteristiği bize açık olarak gösteriyor ki yolları ve patikaları kullanarak yüzey üzerinde ilerleyebiliriz ancak bu yüzeyin altta kalan yüzünü kullanamayız.

Ayrıca, kentsel ulaşım sistemleri sınırlara sahip bir 2-boyutlu yüzeydir. Küreler ve yumrular gibi bilindik kapalı kompakt yüzeylerin aksine iç ve dış sınırların ikisine de sahiptirler. Dışardan şehirlerin bitiş çizgileri ile sınırlı iken, içerden ise mekânların

başladığı ve bittiği yerlerde sınırlıdır. Aşağıda bu sınırların örneklendirildiği bir grafik verilmiştir.



Ulaşım sistemleri hakkında bildiğimiz bu 5 karakteristik özellik (kompakt, bağlantılı, yönlendirilebilir, 2-boyut ve sınırlar) ile şehirleri iyi bilinen matematiksel sonuçlarla karakterize edebiliriz. Şimdi adım adım izleyeceğimiz yöntemleri şekillendirelim. Bu bölümdeki temel amacımız herhangi bir şehri karakterize eden topolojik bir değişmez türetilmesidir. Yani farklı şehirleri birbirine bağlayan veya bir şehrin içini birbirine bağlayan ulaşım sistemlerinin topolojik eşdeğerliklerini ortaya koymaktır. İlk olarak şu teoremle başlayalım.

4.1.1. Teorem: B sınırları ile sınırlandırılmış herhangi bir şehrin ulaşım sistemleri, $B = b + 1$ daireleri çıkartılmış küre ile topolojik olarak aynıdır.

İspat: Kinsey [58] teorem 4.17’de 2-boyutlu ve yönlendirilebilir yüzeylerin belirli durumlarında ulaşım sistemlerinin “kolları” olmadığını açıklar.

Sonuç: b ’nin herhangi bir değeri için b sınırlarına sahip iki farklı şehrin ulaşım sistemleri topolojik olarak aynıdır.

İspat: Teorem 4.1.1’de görülüyor ki, iki şehrin ulaşım sistemlerinin topolojik olarak aynı olması, bunların ayrı ayrı $b + 1$ daireleri çıkartılarak oluşturulan küre ile topolojik olarak aynı olmasındandır.

Sonuç: b_1 sınırları ile sınırlandırılmış bir şehrin herhangi bir alt parçasının ulaşım sistemleri, b_2 sınırları ile sınırlandırılmış başka bir şehrin bir alt parçasının ulaşım sistemleri ile topolojik olarak aynıdır ancak ve ancak $b_1 = b_2$ ise.

İspat: b_1 sınırları ile sınırlandırılmış bir şehir alt parçası tanımlayalım öyle ki dış sınırları ile bir çembere izomorf olacak şekilde. Sonrasında Teorem 4.1.1. bize garanti ediyor ki bu alt parça $b + 1$ daireleri çıkartılmış bir küre ile topolojik olarak aynıdır. Yani bu iki alt parça birbiriyle topolojik olarak aynıdır ancak ve ancak iki parça da ayrı ayrı, aynı sayıda dairelerin çıkarıldığı kürelerle topolojik olarak aynı ise. Dolayısıyla bu iki alt parça birbiri ile aynıdır ancak ve ancak $b_1 = b_2$ ise.

Sonuç: b sınırları ile sınırlandırılmış bir şehrin tamamı, başka bir şehrin aynı sayıdaki sınıra sahip herhangi bir alt parçası ile topolojik olarak aynıdır.

İspat: Bir şehrin alt parçasının, bir başka şehrin tamamıyla aynı tipte bir yüzeye sahip olması, aynı sınırlara sahip iken topolojik olarak aynı olması sayesinde gerçekleşir.

Tüm bunlardan hareketle, kalan kısımda ulaşım sistemlerini topolojinin evrensel değişmezlerinin terimleriyle karakterize etmeye çalışacağız. Bir topolojik değişmezi α olarak tanımlarsak, eğer X ve Y cisimleri topolojik olarak aynı ise $\alpha(X) = \alpha(Y)$ olacağını hatırlayalım.

Kentsel ulaşım sistemlerinin yüzeylerini bilindik Euler karakteristiği olan χ 'nin tanımları ile karakterize edebiliriz. Euler karakteristiği χ , herhangi bir boyuttaki nesneye uygulanabilir. [57]

Bir grafik için χ , $\chi = v - e$ şeklinde tanımlansın. Burada v , köşelerin sayısı; e ise kenarların sayısı olsun. Ağaç şeklinde olan (tree graph theory) herhangi bir grafikte bilindik T için sonuç $\chi(T) = 1$ olacaktır.

Yüzeyler için, düzlemsel grafikler de dâhil olmak üzere χ , $\chi = v - e + f$ şeklinde tanımlanacak olup (burada f yüz sayısıdır) herhangi bir düzlemsel grafik için Y , $\chi(Y) = 2$ şeklindeki bir başka bilindik sonuç olarak bulunacaktır. [59]

Sonuç olarak yapmak istediğimiz şey, herhangi bir kentsel ulaşım sistemi için Euler karakteristiğini hesaplayabilmek. Bunun için gerekli son bileşen, sınırları olan bir yüzey için χ 'nin nasıl hesaplanacağını bilmesidir. Bunu yapmak için B sınırlarına sahip genel bir S yüzeyi ele alalım. Sonra bununla ilişkili yardımcı bir S^* yüzeyi tanımlayalım. S^* yüzeyi, S ile tüm B sınırlı dairelerin yamalar şeklinde S 'nin sınırlarına dikilmesi sonucunda elde edilmiş olsun. Şimdi bu iki yüzey arasındaki ilişkiyi Euler karakteristiği ile sonuçlandıracak olursak;

$$\chi(S^*) = \chi(S) + B$$

şeklinde olacaktır çünkü S ne olursa olsun Euler karakteristiği; S^* ile B ayrık dairelerinin farkıdır. Böylece sınırlandırılmış herhangi bir yüzey için onlara karşılık gelen S^* yüzeylerinin sınırlarını çıkartarak Euler karakteristiklerini hesaplayabiliriz. Bu bizim, b sınırlarına sahip bir kentsel ulaşım sisteminin (bir şehir veya şehir alt parçasının) Euler karakteristiğini hesaplamamıza olanak sağlar.

$$\chi(b \text{ sınırlarına sahip ulaşım sistemi}) = 1 - b$$

Burada S^* 'ın küre olduğunu ve böylece $\chi(S^*) = 2$ olduğunu kullandık. Bu sonucun doğru olduğu Kinsey [58] Teorem 5.17 ile görülebilir.

4.1.2. Teorem: S_1 ve S_2 kompakt, bağlantılı ve sınırlı iki yüzey olsun. S_1 ile S_2 topolojik olarak aynıdır ancak ve ancak aynı sayıda sınır bileşenine sahiptir, ikisi de yönlendirilebilirdir (veya ikisi de yönlendirilemezdir) ve aynı Euler karakteristiğine sahipler ise.

Sonuç: Kentsel ulaşım sistemleri topolojik olarak aynıdır ancak ve ancak aynı sayıda sınıra sahip iseler.

İspat: Bu sonuç, aynı sınır sayısına sahip iki kentsel ulaşım sisteminin aynı Euler karakteristiğine sahip olmasından gelmektedir. Bize göstermektedir ki genel manada herhangi bir şehrin herhangi bir alt parçasının ulaşım sistemi bir başka sistemle topolojik olarak aynıdır ta ki aynı sayıda sınıra sahip oldukları sürece. Her biri diğerine sürekli olarak deforme edilebilir. Yani Paris, New York, Las Vegas ya da Sakarya'nın şehir yapısı geometrik olarak radikal farklılıklar gösterse de topolojik olarak aynıdır. Sonuç olarak ulaşım sistemleri yüzeylerindeki topolojik işaretleri korurlar. Bunu çok bariz örneklendirmeler ile grafiksel bir sunum hâlinde vereceğiz.

4.2. Mekânların Topolojisi

Şimdi ise daha farklı ve daha zorlayıcı topolojik bir problemi düşünelim. Şehir sınırlarının iç yapılanmalarının düzenini. Genel bir terim olarak “parsel” kelimesini, şehir yapılanmasındaki boş yerleri ve binaları kalan kısımlarla ayrışmasını ifade etmek için kullanacağız.

Şehir yapılanmasını topolojik olarak analiz etme işlemi 2 temel adımdan oluşur. İlk adım parsellerin ulaşım sistemleri ile aralarındaki ilişkileri analiz etmektir. Eğer bir parsel, herhangi bir ulaşım sisteminin bir parçası ile komşu ise (bitişik – yan yana) bu parsel “erişilebilir” diyeceğiz. Eğer böyle bir komşuluk mümkün değilse pratikte bu parsel, başka parsellerin içinden erişim sağlar diye bahsedilecek.

Bir bloktaki tüm parseller erişilebilir ise de bu bloğa evrensel olarak erişilebilir denir. Bölüm 4.3'te Bu blokların daha önce bahsettiğimiz ulaşım sistemleri ile topolojik olarak aynılığını inceleyeceğiz. Gelişmiş şehirlerde tüm bloklar evrensel olarak erişilebilir şekildedir. Bu nedenle bizim öneri olarak ortaya koyduğumuz sonuç, tüm şehirlerin gelişmeye devam ettikçe; tüm bloklarını evrensel olarak erişilebilir hâle getirmeye çalışmasıdır. Böylece şehirlerin gelişimi, ulaşım sistemleri ile mekânlar arasındaki topolojik yapının incelenmesi ile analiz edilebilir.

Yapılanmayı topolojik olarak incelemenin 2. adımı ise evrensel olarak erişilebilir durumda olmayan blokların incelenmesinden geçer. Bu tür bloklar, inşası sırasında herhangi bir planlamaya konu edilmeden gelişigüzel ve düzensiz şekilde komşuluklar sağlanarak, yoğun olarak üretilen binalar ve yerleşim yerleri sonucunda oluşur. Bu durum günümüzün gecekondü kültürünü ve geçmişteki antik şehirlerin yapılarının karakteristik özelliklerini ortaya koyar. Aşağıdaki kroki, şuan gelişmiş durumda olan pek çok şehrin daha önceden evrensel olarak erişilebilir olmayan çok sayıda blok içerdiğini bariz bir örnekle ortaya koymaktadır.



Burada San Antolin, San Marcos, Toledo İspanya bölgesindeki yerleşimin tarihi evrimi gösterilmektedir. [60] Solda daha önce gelişigüzel sağlanan yerleşim, gelişim

sağlandıkça ulaşım sistemlerinin konumuna göre parsellere bölünerek evrensel erişime sahip bloklarla sağlanır duruma gelmiş. Bu türden diğer detaylı örnekler, Kostof and Tobias [61] çalışmasıyla yerleşim yerlerinin tarihsel evrimi bağlamında incelenebilir.

İleriki bölümlerde vereceğimiz Teorem 1 ve Teorem 2 ile minimum sayıda parselin, evrensel olarak erişim sağlanabilen komşuluklarla kesişmesi gerektiğini ortaya koyacağız. Sonrasında verilecek Teorem 3 ise minimum uzunluktaki ulaşım sisteminin geometrik ve topolojik olarak nasıl evrensel erişilebilir bir komşuluk üretebileceğini ortaya koyacaktır.

Bu sonuçlarla, ulaşım sistemlerinin topolojisi ile sadece mevcut gelişmiş şehirlerin topolojik analizinin yapılması değil ayrıca çarpık kentleşmenin olduğu geri kalmış şehirlerin matematiksel olarak ne yönde nasıl gelişim göstermeleri gerektiğine dair fikir edinilebileceğini kanıtlayacağız. Bu da bize ideal bir şehir planlamasının matematiksel algoritmik bir optimizasyon problemi şeklinde ele alınabileceğini gösterir.

4.3. Ulaşım Sistemleri ile Mekânların Topolojik Olarak Aynılığı

Şimdi ulaşım sistemleri (yollar) ile mekânlar (binalar) arasındaki topolojik ilişkiyi inceleyeceğiz. Etrafında yollar ve patikalar mevcut ve hâlen gelişmekte olan bir şehir tanımlayalım. Bunu şehrin ulaşımının grafiksel olarak sunumunun düzenlenmesi yani mekânlar ve yolların krokilerle haritalanmasıyla sağlayalım. Bu harita ortaya koyar ki parseller ve yolların ilişkisi ile krokilerdeki çizimler homotopik olarak aynıdır. [62] Homotopik olarak aynı olma topolojik olarak aynı olmanın özel bir durumudur.

4.3.1. Tanım: (Köprü Grafiği) 1-Kompleks bir köprü grafiği tanımlayalım. Buna X diyelim. Y ise parseller ve ulaşım sistemleri arasındaki ilişkiyi temsil etsin. Köprü grafiği yolları ve patikaları sembolize eden kenarları içerecektir. Düğümler ise bu yollar arasındaki bağlantılar olacaktır. x 'i X 'teki herhangi bir bileşen (kenar ya da düğüm) olarak tanımlayalım. Y 'deki tüm bileşenlere ek olarak, X ayrıca şehirdeki her

parselin merkezini temsil eden düğümleri de içermektedir. Bu başlangıçta bağlantısız olan N düğümlerinin bir serisini oluşturur, öyle ki N 'de yer alan her n_i düğümü şehirdeki belirli bir parseli temsil eder. Sonrasında biz bir e_i kenarı eklersek, bu ya bir n_i düğümünü kenar ile birleştirecektir ya da bu parsel zaten doğal olarak erişilebilir durumdadır. Eğer e_i ile n_i Y 'nin elemanı olan x 'in içindeki mevcut bir n_i^* düğümü ile birleşiyorsa değişikliğe ihtiyaç yoktur. Ancak eğer e_i ile n_i Y 'nin elemanı olan x 'in içindeki bir e_γ kenarı ile birleşiyorsa; bu kenar e_{γ_1} ve e_{γ_2} şeklinde parçalara ayrılıp kenar iyileştirme denen işlem uygulanmalıdır. n_i^* düğümü de bu parseli ulaşım sistemine bağlayan kavşak olarak adlandırılmalıdır. Sonrasında e_{γ_1} ise n_i^* ile n_γ 'yi birleştiren kenar olarak adlandırılacaktır.

Bu yolla X , Y uzayını yollar ve bu yolları birbirine bağlan kavşakları temsil eden kenarlar ve köşelerle örtecektir. Ayrıca X , farklı parsellere geçiş yapmayı sağlayan, kavşaklarla insanların yürüyüş yollarının arasındaki düğümleri de içermektedir. Bu tanımla birlikte Y 'nin, X 'in bir alt uzayı olduğu açıkça görülmektedir. Buna rağmen X 'teki bazı düğümler Y 'nin kenarlarını ve X 'in Y ile örtüşen bazı kenarlarını içermektedir.

X 'in Y ile topolojik olarak aynı olduğunu göstermek, topolojik olarak aynılığın bir türü olan “homotopik aynılık” tanımını kullanmamızı sağlayacaktır. [62] Buradan hareketle X ve Y 'nin homotopik olarak aynılığı mevcutsa X topolojik uzayının Y 'ye sürekli olarak deforme olduğu söylenebilir. Burada ise X 'in Y 'ye sürekli deformasyonunun tanımına ihtiyaç duyarız. Cebirsel Topoloji'de mevcut “Güçlü Deformasyon Geri Çekilmesi” (strong deformation retraction) tanımının temel konseptini aşağıda verelim.

4.3.2. Tanım: $H(x, t)$ şeklinde verilen sürekli bir harita, X uzayının bir Y alt uzayına güçlü deformasyon geri çekilmesidir eğer aşağıdaki 3 şart sağlanıyorsa.

- i. $H(x, 0) = x$
- ii. $H(x, 1) \in Y$
- iii. $H(y, t) = y$

Burada $x \in X$, $y \in Y$ ve $t = [0,1]$ şeklindedir. Bu tanımla birlikte bu bölümün ana sonucunu aşağıdaki gibi verebiliriz.

4.3.1. Teorem: Herhangi bir şehirde mekânların uzayı ile ulaşım sistemlerinin uzayı evrensel erişilebilir bloklar için homotopik olarak aynıdır.

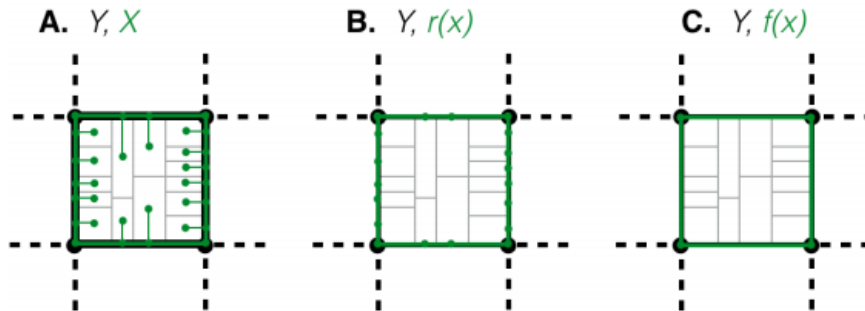
İspat: X köprü grafiğinden Y ulaşım ağına güçlü deformasyon geri çekilmesi mevcutsa, bu iki uzay homotopik olarak aynıdır. [62] Şimdi de X 'ten Y 'ye bir güçlü deformasyon geri çekilmesi inşa edelim. Öncelikle $r(x): X \rightarrow X$ şeklinde bir harita tanımlayalım.

$$r(x) = \begin{cases} x, & x \in Y \\ n'_i, & x \notin Y \end{cases}$$

Burada $\{r(x)\}$ 'in n düğümleri ve e kenarları ile, Y 'nin ise Y_n düğümleri ve Y_e kenarları ile oluşturulduğu bilgisiyle ikinci bir harita tanımlayalım:

$$f(x) = \begin{cases} n \in Y_n ; r(x) \in Y_n \\ e \in Y_e ; r(x) \in Y_e \\ n_Y ; r(x) = n \notin Y_n \end{cases}$$

Harita f , X köprü grafiğinden Y ulaşım sistemine bir “geri çekilme”dir. r haritası ise parselleri temsil eden tüm düğümleri, en yakın kavşaklara ulaşım sistemi aracılığı ile “geri çeker”. Ayrıca f haritası özel ve genel kavşakları ulaşım sistemindeki yol ve patikalarla sadece genel kavşaklara “geri çeker”. Geri çekilmenin köprü grafiği aşağıdaki gibi illüstre edilmiştir.



Şimdi de $H(x, t)$ şeklinde bir harita tanımlayalım.

$$H(x, t) = (1 - t)x + f(x)$$

Bu $H(x, t)$ haritasının aşağıdaki güçlü deformasyon geri çekilmesi için 3 koşulun tamamını sağladığını görelim:

Koşul 1:

$$H(x, 0) = (1 - 0)x + 0 \cdot f(x) = x$$

Koşul 2: $H(x, 1) \in Y$

$r(x)$ ve $f(x)$ 'in tanımlarına dayanarak söyleyebilir ki Y 'nin elemanı olmayacak şekildeki tüm x 'ler n'_i 'ne geri çekilir. Eğer $n'_i \notin Y_n$ ise n_Y 'ye geri çekilecektir.

Koşul 3: $y \in Y$ için $H(y, t) = y$

Sonrasında $r(y) = y$ ve $f(y) = y$ ve ayrıca $H(y, t) = y$ eşitliklerinin her t için sağlandığı tanımlardan rahatlıkla görülebilir. Ayrıca şunu not edelim ki burada, H 'de “geriye dönük” dönüşüm uyguladık öyle ki t 1'den 0'a değişirken. Böylece Y 'den X 'e bir yeniden yapılandırma gerçekleştirmiş olduk.

Böylelikle köprü grafiği olan X 'ten ulaşım sistemi olan Y 'ye sürekli bir harita inşa etmiş olduk öyle ki herhangi bir parselin, ulaşım sistemine ve ulaşım sisteminin bir parçasına erişim imkânı olduğu. Bu gösteriyor ki her bir evrensel olarak erişilebilir şehir bloklarının içindeki bağlantılı parsellerin topolojisi ait olduğu ulaşım ağı ile aynıdır.

Bu bize, mekânların uzayını ve aralarındaki etkileşimlerini ve ayrıca şehirlerin ulaşım sistemlerinin birbirleri ile topolojik olarak aynılığını sonuçlandırma imkânı veriyor. Böylelikle herhangi bir şehrin tüm topolojisini tanımlayabiliriz.

5. SONUÇLAR VE ÖNERİLER

Çalışmanın tamamında verilen bilgiler ışığında, bir şehir planlaması yapılırken; Topoloji, Bilgisayar, Elektronik, Optimizasyon, Olasılık ve İstatistik, Cebir, genel manada Matematik Bilimleri kullanımının kritik bir öneme sahip olduğu sonucuna varılabilir. Yani mevcut bir şehrin gelişiminin planlanması veya sıfırdan bir şehir inşası söz konusu olduğunda, Matematik Bilimleri sayesinde daha iyi ve en iyi sonuçlara ulaşılabileceği söylenebilir.

Bir iş yapılırken adımların gelişigüzel takip edilmesi yerine önceden matematiksel olarak hesaplanmış yöntemlerin uygun sıra ile izlenmesi gerekliliği su götürmez bir gerçektir. Bu çalışmada, gerek gelişmiş şehirlerimiz gerekse gelişmekte olan şehirlerimiz ölçeğinde de gözlemlenebilecek plansız ve matematiksel modellemelere dayanmayan işlemler bütünüünün yarattığı sıkıntıları gidermek adına matematiksel şehir planlama yöntemleri önerilmektedir.

Çalışmanın giriş kısmında Topoloji Biliminin, bilgisayar ağlarının yerleşimini nasıl incelediği teknolojik yönleri ile detaylı şekilde irdelenmiştir. Topolojinin yoğun ve etkin bir biçimde kullanıldığı bu alan, bilgisayar ağlarının yapılarını; avantajlı ve dezavantajlı durumlarını incelemek için oluşturulmuş kullanışlı bir konu başlığıdır. Bu yapının her durumunu en ince detayına kadar inceleyebilme imkânı sağlayan bu başlık, aslen yapıyı iyileştirmek için kullanılan yöntemler bütününe de altyapı sağlamaktadır.

Çalışmanın 2. bölümünde detaylandırılan "Bilgisayar Ağlarının Topolojileri" konusu, yukarıda bahsedilmeye çalışılan yöntemlerin bilgisayar ağlarının optimize edilmesi konusunda nasıl etken olarak kullanıldığını ortaya koymaktadır. Öyle ki bu konuda yapılmış yerli ve yabancı yüzlerce güncel çalışmanın ne kadar detaylı olduğu 5. ve 12. sayfalar arasındaki uzun konu başlıklarına yansısıyla gözlemlenebilir.

3. bölümde mevcut bir ağın daha iyi bir duruma getirilirken kullanılması gereken yöntem ve modellemelerin bir kesitini sunmaktadır. Sadece belli başlı yöntemlerden bahsedilmesine rağmen gayet detaylı ve uzun açıklamaları barındıran bu bölüm ele alınan konu üzerine yerli ve yabancı çok sayıda araştırmacının çalıştığını ortaya koymaktadır. "Literatür İncelemesi" kısmında bu durum açık bir şekilde görülmektedir. Öyle ki matematikten faydalanarak kurdukları öğretileri modelleyip algoritmalar hâline getiren mühendisler, çok önemli projelere kılavuzluk etmiş, yüksek maddi ederi olan katkılarda bulunmuşlardır.

Ancak son bölüme gelindiğinde irdelenen konunun, çok az sayıda bilim insanı tarafından araştırıldığı özellikle ulusal çapta pek ele alınmadığı ifade edilebilir. Şehir planlama konusunu topolojik açıdan değerlendirme durumunun alışılmış ve yaygın bir bakış açısı olmaması, bu konuda uluslararası düzeyde az sayıda çalışma olması nedeniyle söylenebilir. 4. bölümün tamamını oluşturan [56] çalışmasının bir bölümünün çevirisi olan kısım bize gösteriyor ki şehirleri uzaylar olarak inceleyebilir; onların üzerlerinde topolojik işlemler yapabiliriz.

Dolayısıyla, şehirlerin topolojik uzaylar olarak incelenmesi durumu, bilgisayar ağlarının topolojik olarak incelenmesi neticesinde üretilen optimizasyon yöntemlerinin şehir planlaması yöntemlerine de uygulanmasını sağlayabilir. Bir bilgisayar ağı kurulurken kullanılan bilgisayar, kablo ve bağlantı elemanları gibi; evler, yollar ve bu yolları birleştirmeye yarayan kavşaklar da bölüm 3'de bahsedilen modellemeler ve algoritmalar ile uygun şekilde konumlandırılabilir.

Sonuç olarak mevcut bir şehrin ulaşımının hızı, maliyeti, güvenilirliği gibi kıstaslar hesaplanabilir ve bu kısıtlamalara göre iyileştirmeler yapılabilir. Yani mevcut şehirlerin uygun gelişmişlik düzeyine çıkması için ne türden değişikliklerin yapılması gerektiği matematiksel olarak hesaplanabilir. Matematik kullanılarak daha yaşanılır şehirler kurulabilir...

KAYNAKLAR

- [1] AGGARWAL K.K., CHOPRA Y.C., BAJAWA J.S., Topological Layout of Links for Optimizing The S-T Reliability in Computer Communication System, Microelectronics & Reliability, vol.22, no.3, s341-345, 1982.
- [2] JAN, R., H., HWANG, F., J., CHENG, S., T., Topological Optimization of a Communication Network Subject to Reliability Constraint, IEEE Transaction on Reliability, vol.42-1, s.63-70, 1993.
- [3] DENGİZ, B., ALTIPARMAK, F., SMITH, A., E., Efficient Optimization of All Terminal Reliable Networks, IEEE Transactions on Reliability, vol.46, no.1, s.18-26, 1997a.
- [4] DENGİZ, B., ALTIPARMAK, F., SMITH, A., E., Local Search Genetic Algorithm for Optimal Design of Reliable Networks, IEEE Transactions on Evolutionary Computation, vol.1, no.3, s.179-188, 1997b.
- [5] DENGİZ, B., ALABAS, C., A Tabu Search Algorithm for Design of Computer Networks, 4th World Multi-Conference on Circuits, Systems, Communications and Computers, s.363-368, 2000.
- [6] DENGİZ, B., ALABAS, C., A Simulated Annealing Algorithm for Design of Computer Communication Networks, 5th Multi-Conference on Systemics, Cybernetics, and Informatics, vol.5, Part I. s.188 -193, 2001.
- [7] HOSAM, M., F., ABOELFOTOH AND LOULWA S. AL-SUMAIT, A Neural Approach to Topological Optimization of Communication Networks, With Reliability Constraints, IEEE Transactions On Reliability, vol.50, no.4, 2001.
- [8] KONAK, A., SMITH, A., E., Handbook of Optimization in Telecommunications, Springer Science + Business Media, s.735-761, 2006.
- [9] ALTIPARMAK, F., Genetik Algoritma ile Haberleşme Şebekelerinin Topolojik Optimizasyonu, Doktora Tezi, Gazi Üniversitesi Fen Bilimleri Enstitüsü, Ankara, 1996.
- [10] JAN, R, H., Design of Reliable Networks, Computers and Operations Research, vol.20-1, s.25-34, 1993.
- [11] BOORSTYN, Robert R. and FRANK Howard, Large Scale Network

- Topological Optimization, IEEE, Transactions on Communications, vol.25, no.1, s.29-37, 1977.
- [12] AGGARWAL, K. K. and SURESH R., Reliability Evaluation in Computer Communication Networks, IEEE Transactions on Reliability. Vol. R-30, no. 1, s. 32-35, 1981.
 - [13] CHOPRA, Y., C., SOHI, B., S., TIWARI, R., K., Network Topology For Maximizing The Terminal Reliability In A Computer Communication Network, Microelectronics and Reliability, vol.24, s.911-913, 1984.
 - [14] VENETSANOPOULOS, A. N. and SINGH, I., Topological Optimization Of Communication Networks Subject To Reliability Constraints, Prob. Control Info. Theory. Vol.15, no.1, s.63-78, 1986.
 - [15] KUMAR, Anup, PATHAK, Rakesh, M., GUPTA, Yash, P., Genetic Algorithm - Based Reliability Optimization for Computer Network Expansion, IEEE Transactions On Reliability, vol. 44, no.1, s.63-72, 1995.
 - [16] KUMAR, Anup, PATHAK, Rakesh, M., GUPTA, Yash, P., PARSAEI, Hamid, R., A Genetic Algorithm for Distributed System Topology Design, Computers and Industrial Engineering, vol.28, no.3, s.596-670, 1995.
 - [17] COSTAMAGNA, Eugenio, FANNI, Alessandra, GIACINTO, Giorgio, Simulated Annealing Algorithm for The Optimization Of Communication Networks, International Symposium on Signals Systems and Electronics, San Francisco, s.405-408, 1995.
 - [18] DEETER, Darren, L. and SMITH, Alice, E., Heuristic Optimization of Network Design Considering All - Terminal Reliability, Annual Reliability and Maintainability Symposium, Philadelphia, s.194-199, 1997.
 - [19] AHUJA, Sanjay, P., Performance Based Reliability Optimization for Computer Networks, Southeastcon '97, s.121-125, 1997.
 - [20] DENGİZ, Berna, ALTIPARMAK, Fulya, SMITH, Alice, E., Efficient Optimization of All - Terminal Reliable Networks, Using an Evolutionary Approach, IEEE Transactions of Evolutionary Reliability, vol.46 no.1, s.18-26, 1997.
 - [21] PIERRE, Samuel and LEGAULT, Gisele, A Genetic Algorithm for Designing Distributed Computer Network Topologies, IEEE Transactions On Systems, Man, And Cybernetics - Part B: Cybernetics, vol.28, no.2, s.249-258, 1998.
 - [22] PIERRE, Samuel and ELGIBAOUI, Ali, A Tabu-Search Approach for Designing Computer-Network Topologies with Unreliable Components, IEEE Transactions On Reliability, vol. 46, no. 3, 1997.
 - [23] COSTAMAGNA, Eugenio, FANNI, Alessandra, GIACINTO, Giorgio, A Tabu Search Algorithm for the Optimization of Telecommunication

- Networks, European Journal of Operational Research, vol.106, s.357-372 , 1998.
- [24] CHENG, Sheng, Tzong, Topological Optimization of a Reliable Communication Network, IEEE Transactions On Reliability, vol.47, no.3, 1998.
 - [25] DEETER, Darren, L., and SMITH, Alice, E., Economic Design of Reliable Networks, IIE Transactions, vol.30, s.1161-1174, 1998.
 - [26] ALTIPARMAK, Fulya, DENGIZ, Berna, SMITH, Alice, E., Reliability Optimization of Computer Communication Networks Using Genetic Algorithms, International Conference on Systems, Man and Cybernetics, vol.5, s.4676-4681, 1998.
 - [27] KONAK, Abdullah and SMITH, Alice, E., A Hybrid Genetic Algorithm Approach for Backbone Design of Communication Networks, Proceedings of the 1999 Congress on Evolutionary Computation, Washington, D.C. (USA), s. 1817–1823, 1999.
 - [28] LIU, Baoding, IWAMURA, K., Topological Optimization Models for Communication Network with Multiple Reliability Goals, Computers and Mathematics with Applications, vol.39, s.59-69, 2000.
 - [29] ABOELFOTOH, Hosam, M., F. and AL-SUMAIT, Loulwa, S., A Neural Approach to Topological Optimization of Communication Networks With Reliability Constraints, IEEE Transactions on Reliability, vol. 50, no. 4, 2001.
 - [30] FARD, N., LEE, Tea, Han, Spanning Tree Approach in All-Terminal Network Reliability Expansion, Computer Communications, vol.24, s.1348-1353, 2001
 - [31] KOIDE, T., SHINMORI, S., ISHII, H., Topological Optimization With a Network Reliability Constraint, Discrete Applied Mathematics, vol.115, s.125-149, 2001.
 - [32] KUMAR, Rajeev, PARIDA, Pranja, P., GUPTA, Mohit, Topological Design of Communication Networks using Multi Objective Genetic Optimization, Proceedings of the 2002 Congress on Evolutionary Computation, vol.1, s.425-430, 2002.
 - [33] SRIVAREE-RATANA, Chat, KONAK, Abdullah, SMITH, Alice E., Estimation Of All-Terminal Network Reliability Using An Artificial Neural Network, Computers & Operations Research, vol.29, s.849-868, 2002.
 - [34] ALTIPARMAK, Fulya, and DENGIZ, Berna, Optimal Design of Reliable Computer Networks: A Comparison of Metaheuristics, Journal of Heuristics, vol.9, s.471-487, 2003.

- [35] MANDALL, Swamp, SAHA, Debashis, MUKHEJEE, Rajarshi, ROY, Anandarup, An Efficient Algorithm for Designing Optimal Backbone Topology for a Communication Networks, International Conference on Communication Technology, Beijing, China, vol.1, s.103-106, 2003.
- [36] ALTIPARMAK, Fulya, DENGIZ, Berna, SMITH, Alice E., Reliability Estimation of Computer Communication Networks: ANN Models, Eighth IEEE International Symposium on Computers and Communication (ISCC'03), Kemer, Antalya, vol.2, s.1-6, 2003.
- [37] ALTIPARMAK F., GEN, M., DENGIZ, B., SMITH, A., E., A Genetic Algorithm with Fuzzy Logic Controller for Design of Communication Networks, Transactions on Electronics, Information and Systems, vol.24, no.10, s.1979-1985, 2004.
- [38] SHAO, Fang-Ming, SHEN, Xuemin, and HO, Pin-Han, Reliability Optimization of Distributed Access Networks With Constrained Total Cost, IEEE Transactions On Reliability, vol.54, no.3, s.421-430, 2005.
- [39] XIONG Jintao, GONG Weibo, A Novel Algorithm on Network Reliability Estimation, Mathematical And Computer Modeling, vol.41, no.2-3, s.119-133, 2005.
- [40] REICHELT, Dirk, GMILKOWSKY, Peter, LINSER, Sebastian, A Study of an Iterated Local Search on the Reliable Communication Networks Design Problem, EvoWorkshop 2005, Lausanne, s.156-165, 2005.
- [41] MARSEGUERRA Marzio, ZIO Enrico, PODOFILLINI Luca, and COIT David W., Optimal Design of Reliable Network Systems in Presence of Uncertainty, IEEE Transactions On Reliability, vol.54, no.2, s.243-253, 2005.
- [42] GEN, Mitsuo, KUMAR, Anup, KIM, Jong, Ryul, Recent Network Design Techniques Using Evolutionary Algorithms, International Journal of Production Economics, s.251–261, 2005.
- [43] REICHELT, Dirk, ROTHLAUF, Franz, Reliable Communication Network Design with Evolutionary Algorithms, International Journal of Computational Intelligence and Applications, vol.5, No.2, s.251–266, 2005.
- [44] KONAK, Abdullah, BARTOLACCI, Michael, R., Designing Survivable Resilient Networks: A Stochastic Hybrid Genetic Algorithm Approach, The International Journal of Management Science, vol.35, s.645-658, 2007.
- [45] CANCELA, Hector, PETINGI, Louis, Properties of a Generalized Source to All Terminal Network Reliability Model with Diameter Constraints, The International Journal of Management Science, vol.35, s.659-670, 2007.

- [46] KHAN, Salman, A., ENGELBRECHT, Andries, P., A New Fuzzy Operator and Its Application to Topology Design of Distributed Local Area Networks, *Information Sciences*, vol.177, s.2692-2711, 2007.
- [47] LUCIO, Gilberto, Flores, REED, Martin, J., HENNING, Ian, D., Guided Local Search as a Network Planning Algorithm That Incorporates Uncertain Traffic Demands, *Computer Networks*, vol.51, s.3172-3196, 2007.
- [48] HUI, Kin-Ping, Monte Carlo Network Reliability Ranking Estimation, *IEEE Transactions on Reliability*, vol.56, no.1, 2007.
- [49] MARQUEZA, Jose, Emmanuel, Ramirez, ROCCO, Claudio, M., All-Terminal Network Reliability Optimization via Probabilistic Solution Discovery, *Reliability Engineering and System Safety* 93, s.1689–1697, 2008.
- [50] SEM, K., MALHOTRA, S., Multi Criteria Network Design Using Genetic Algorithm, *Wireless, Mobile and Multimedia Networks, IET International Conference*, s.56-60, 2008.
- [51] DORIGO, M., MANIEZZO, V., COLORNI, A., Positive Feedback as a Search Strategy, Technical Report 91-016, Dipartimento di Elettronica, Politecnico di Milano, Milan, 1991.
- [52] KUAN, C.M., WHITE, H., Artificial Neural Networks: An Econometric Perspective, Vol. 13, No. 1, pp. 1-91, *Econometric Reviews*, 1994.
- [53] ÖZDEMİR, Y.S., Topological Optimization of Computer Networks Using Ant Colony Algorithm, Başkent University, Institute of Science, 2008
- [54] YEH, M., LIN, J., YEH, W., A New Monte Carlo Method for Estimating Network Reliability, *IEEE Transactions on Reliability*, vol.29-1, s.27-32, 1994
- [55] COLORNI, A., DORIGO, M. and MANIEZZO, V., Distributed Optimization by Ant Colonies, *Proceedings of the First European Conference on Artificial Life*, s.134-142, 1992.
- [56] BRELSFORD, C., MARTIN, T., HAND J., and BETTENCOURT, L., The Topology of Cities, Santa Fe Institute, s.18-39, 2015.
- [57] LEE, J. M., Introduction to Topological Manifolds, Springer, New York, ed. 1, 2000.
- [58] KINSEY, L. C., Undergraduate Texts in Mathematics, Topology of Surfaces, s81, Springer Science NY., 1993.
- [59] WEST, D. B., Introduction to Graph Theory, Prentice Hall, Upper Saddle River, NJ, ed. 2nd, 2000

- [60] PASSINI J., MOLENAT J.P., Colegio Oficial de Arquitectos de Castilla-La Mancha, Delegación de Toledo, Toledo a finales de la Edad Media Colegio Oficial de Arquitectos de Castilla-La Mancha, Delegación de Toledo, Toledo, 1997
- [61] KOSTOF S., TOBIAS R., The City Shaped: Urban Patterns and Meanings Through History; Little, Brown and Co., Boston, 1999
- [62] LEE J.M., Introduction to Topological Manifolds; Graduate texts in mathematics; Springer, New York, ed. 1, 2000

ÖZGEÇMİŞ

Volkan DİLSİZ, 9 Kasım 1993 tarihinde İstanbul’da doğdu. İlk ve orta öğrenimini İstanbul’da Göktürk İlköğretim Okulu ve Süleyman Nazif Anadolu Lisesi’nde tamamladı. 2011 yılında Sakarya Üniversitesi, Matematik Bölümünde lisans eğitimine başladı. Sakarya Üniversitesi Geleneksel Türk Okçuluğu Topluluğu onursal üyesi ve Sakarya Okçuluk Spor Kulübü Yönetim Kurulu üyesidir. Hâlen 2012 yılında kurduğu İstanbulMarka Kurumsal adlı şirketin kurucu yönetici pozisyonunda çalışmaktadır.