

T.C.
KARA HARP OKULU
SAVUNMA BİLİMLERİ ENSTİTÜSÜ
HAREKAT ARAŞTIRMASI ANA BİLİM DALI

AĞLARIN HAREKETLİ YOL KESİCİ TARAFINDAN
EN KISA GÜZERGAH KULLANILARAK KESİLMESİ

DOKTORA TEZİ

Hazırlayan
Ahmet KABARCIK

Tez Danışmanları
Yrd. Doç. Dr. Haluk AYGÜNEŞ
Prof. Dr. Levent KANDİLLER

ANKARA - 2013

TEZ TANITIM FORMU

TEZİN TARİHİ : 11.07.2013

TEZİN TİPİ : Doktora Tezi

TEZİN BAŞLIĞI : Ağların Hareketli Yol Kesici Tarafından En Kısa Güzergâh Kullanılarak Kesilmesi

TEZİN YAPILDIĞI BİRİM : Kara Harp Okulu Savunma Bilimleri Enstitüsü Harekat Araştırması Ana Bilim Dalı

SPONSOR KURULUŞ : -

DAĞITIM LİSTESİ : Kara Harp Okulu Savunma Bilimleri Enstitüsü Tez Hazırlama, Onay, Dağıtım ve Muhafaza Esasları Kılavuzunda Belirtilen Yerlere

TEZİN ÖZETİ : Bu çalışmada arkların hareketli yol kesici tarafından devre dışı bırakıldığı bir ağ kesme problemi ele alınmıştır. İç içe geçmiş iki ağdan biri ağ kullanıcı tarafından, diğeri ise yol kesici tarafından kullanılmaktadır. Yol kesici ağı üzerindeki düğümler ağ kullanıcı ağındaki arkların ya da düğümlerin üzerinde konuşlanmıştır. Yol kesici ağı üzerindeki düğümler imha edilmeye aday noktalardır. Bu çalışmada ağ kullanıcının başlangıç ve hedef düğümleri arasındaki tüm güzergâhlarını imha etmek için yol kesicinin kullanacağı en kısa güzergâh bulunmaya çalışılmaktadır. Problemin çözümü için bir matematiksel model oluşturulmuş ve dal sınır yöntemi kullanılarak bir algoritma geliştirilmiştir.

ANAHTAR KELİMELER : Ağ Kesme, Ağ Kullanıcı, Dal Sınır Algoritması, En Kısa Güzergâh, Hareketli Yol Kesici, Maksimum Akış / Minimum Kesme

SAYFA SAYISI : 101

GİZLİLİK DERECESESİ : Tasnif Dışı

T.C.
KARA HARP OKULU
SAVUNMA BİLİMLERİ ENSTİTÜSÜ
HAREKAT ARAŞTIRMASI ANA BİLİM DALI

AĞLARIN HAREKETLİ YOL KESİCİ TARAFINDAN
EN KISA GÜZERGAH KULLANILARAK KESİLMESİ

DOKTORA TEZİ

Hazırlayan
Ahmet KABARCIK

Tez Danışmanları
Yrd. Doç. Dr. Haluk AYGÜNEŞ
Prof. Dr. Levent KANDİLLER

ANKARA - 2013

KARA HARP OKULU

SAVUNMA BİLİMLERİ ENSTİTÜSÜ MÜDÜRLÜĞÜNE

Ahmet KABARCIK'ın, Ağların Hareketli Yol Kesici Tarafından En Kısa Güzergâh Kullanılarak Kesilmesi konulu tez çalışması, jürimiz tarafından HAREKÂT ARAŞTIRMASI Ana Bilim Dalında DOKTORA tezi olarak kabul edilmiştir.

Başkan

Prof. Dr. Cevriye GENCER

Üye

Doç. Dr. İbrahim AKGÜN

Üye

Yard. Doç. Dr. Altan ÖZKIL

Üye

Dr. Hünkar TOYOĞLU

Üye

Yard. Doç. Dr. Haluk AYGÜNEŞ (Danışman)
Prof. Dr. Levent KANDİLLER (Eş Danışman)

ONAY

Yukarıdaki imzaların, adı geçen öğretim üyelerine ait olduğunu onaylarım.

11/07/2013

(İmza ve Mühür)
Önder Haluk TEKBAŞ
Doç. Dr. Müh. Alb.
Enstitü Müdürü

T.C.
KARA HARP OKULU
SAVUNMA BİLİMLERİ ENSTİTÜSÜ
HAREKAT ARAŞTIRMASI ANA BİLİM DALI
ANKARA 2013

AĞLARIN HAREKETLİ YOL KESİCİ TARAFINDAN
EN KISA GÜZERGAH KULLANILARAK KESİLMESİ

DOKTORA TEZİ

Ahmet KABARCIK

ÖZET

Her geçen gün ilerleyen teknoloji yaşamın her alanında daha çok iletişim ihtiyacı doğurmaktadır. İletimi sağlanan şey tren, akaryakıt ya da veri gibi birbirinden tamamen farklı nesneler olabilse de, iletim yapısının ortak bir tanımı vardır. Bu tanıma göre iletim düğümler arasındaki arklar üzerinden sağlanır. Bu yapının adı ağdır.

Ağdaki iletişimin kesilmek istendiği durumlarda ağ üzerinde yeni bir oyuncu tanımlanır. Bu oyuncunun adı yol kesicidir. Ağ üzerinde taşıma gerçekleştiren oyuncuya ise ağ kullanıcı denir. Bu çalışmada ağın düğümleri ve arkları oyuncular tarafından paylaşılmıştır. Ağdaki her düğüm ya da ark sadece oyunculardan biri tarafından kullanılabilir. Bu şekilde ikiye ayrılan ağ sanki içiçe girmiş iki ağ gibidir.

Ağ üzerindeki iki oyuncunun amaçları birbirleriyle çelişmektedir. Birinin amacını gerçekleştirmesi demek diğerinin amacını gerçekleştirme şansının kalmaması demektir. Ağ kullanıcı başlangıç düğümünden hedef düğümüne en kısa güzergâh üzerinden ulaşmayı amaçlamaktadır. Yol kesici ise ağ

kullanıcının başlangıç ve hedef düğümleri arasındaki bağlantısını kendi ark ve düğümleri üzerindeki en kısa güzergâhı kullanarak kesmeye çalışmaktadır. Problemin çözümü için kullanılan dal sınır algoritmasında yol kesici kestiği her güzergâhtan sonra ağ kullanıcının bir sonraki en kısa güzergâhına yönelmektedir.

Anahtar Kelimeler : Ağ Kesme, Ağ Kullanıcı, Dal Sınır Algoritması, En Kısa Güzergâh, Hareketli Yol Kesici, Maksimum Akış / Minimum Kesme

Tez Yöneticileri : Yrd. Doç. Dr. Haluk AYGÜNEŞ,
Prof. Dr. Levent KANDİLLER

Sayfa Sayısı : 101

T.R.
TURKISH MILITARY ACADEMY
DEFENSE SCIENCE INSTITUTE
DEPARTMENT OF OPERATIONS RESEARCH
ANKARA 2013

NETWORK INTERDICTION BY DYNAMIC INTERDICTOR
USING SHORTEST PATH
DISSERTATION

Ahmet KABARCIK

ABSTRACT

Advances in technology yields to the need for more communication in all aspects of every day life. Although the transmitted objects may be completely different from each other like train, petroleum or data, there is a common structure. On which transmission is obtained by using arcs connecting the nodes. This common structure is called as network.

When the transmission on the network is intended to be interdicted, a new player is defined on the network. The name of this player is the network interdictor. The player, who performs transmission on the network, is called the network user. The nodes and arcs of the network are shared by the players. Each node or arc on the network can be used by only one of the players. The network shared by the players seems to be two interconnected networks.

The objectives of two players are contradicting with each other. Realization of the objective of one player means loosing chance of realizing the objective of other. The network user aims to travel between origin and destination nodes by using the shortest path. The network interdictor aims to cut the communication between the origin and the destination nodes of the network

user by using his shortest path along his own nodes and arcs. In the branch and bound method, which is used to solve the problem, network interdicator is heading to the following shortest path of network user after cutting a path.

Keywords : Branch and Bound Algorithm, Dynamic Network Interdictor, Network Interdiction, Network User, Shortest Path, Maksimum Flow / Minimum Cut.

Advisors : Assist. Prof. Haluk AYGÜNEŞ,
Prof. Levent KANDİLLER

Number of Pages : 101

İÇİNDEKİLER

İÇİNDEKİLER.....	v
TABLolar.....	ix
ŞEKİLLER.....	x
KISALTMALAR LİSTESİ.....	xiii
ÖNSÖZ.....	xv
GİRİŞ.....	1

BİRİNCİ BÖLÜM

PROBLEM TANIMI

1. PROBLEM TAKSONOMİSİ.....	3
2. PROBLEMİN AYRINTILARI.....	7

İKİNCİ BÖLÜM

LİTERATÜR ARAŞTIRMASI

1. LİTERATÜRDE AĞ KESME PROBLEMLERİ.....	12
a. Düğüm, Ark ve Güzergâha Bağlı Ağ Kesme Problemleri.....	14
(1) Dügüme Bağlı Ağ Kesme Problemleri.....	14
(2) Arka Bağlı Ağ Kesme Problemleri.....	15
(3) Dügüm ve Arka Bağlı Ağ Kesme Problemleri.....	16
b. Oyuncuların Amaçlarına Göre Ağ Kesme Problemleri.....	17
(1) En Kısa Güzergâhın Kesilmesi.....	17

(2) Maksimum Akışın Kesilmesi.....	18
(3) Minimum Maliyetle Ağ Kesme.....	20
(4) Çok Amaçlı Ağ Kesme.....	21
c. İstihbarata Dayalı Asimetrik Ağ Kesme.....	23
ç. Deterministik ve Stokastik Ağ Kesme Problemleri.....	24
d. Ağ Kesme Problemlerinin Oyun Teorisi ile İncelenmesi.....	26
e. Tek Ürünlü, Çok Ürünlü Ağ Kesme Modelleri.....	27
2. LİTERATÜR ÇALIŞMALARININ PROBLEM TANIMLAMA ÇİZELGESİ İLE AÇIKLANMASI.....	29
3. BU TEZ ÇALIŞMASININ LİTERATÜRE KATKILARI.....	36

ÜÇÜNCÜ BÖLÜM

MATEMATİKSEL MODELLER

1. PROBLEMİN ÇÖZÜMÜ İÇİN OLUŞTURULAN MATEMATİKSEL MODELLER.....	37
--	----

DÖRDÜNCÜ BÖLÜM

DAL SINIR ALGORİTMASI

1. DAL SINIR ALGORİTMASI İLE PROBLEMİN ÇÖZÜMÜ.....	44
a. Dal Sınır Algoritmasında Üst Sınır Değeri.....	44
b. Düğümün Alt Sınır Değeri.....	44
c. Dal Sınır Algoritmasında Dallanma.....	45

ç.	Dal Sınır Algoritmasının Probleme Uyarlanması.....	45
d.	Dal Sınır Algoritmasının Uygulanması.....	47
2.	ÜST SINIR DEĞERİNİN BELİRLENMESİ.....	52
a.	Ağlarda Kesit Oluşturma.....	52
(1)	Düğümlerini Temel Alarak Kesit Oluşturma.....	52
(2)	Maksimum Akış/Minimum Kesme Yöntemiyle Kesit Oluşturma.....	54
(3)	En Uygun Kesit Sayısının Aranması.....	56
b.	Kesitler Üzerinde Güzergâh Arama.....	57
(1)	Düz Çizgi Yöntemi.....	58
(2)	Düz Çizgi 2 Yöntemi.....	59
(3)	En Yakın Komşu Yöntemi.....	60
(4)	En Ucuz Ekleme Yöntemleri.....	60

BEŞİNCİ BÖLÜM

KOŞUM SONUÇLARI

1.	DENEME AĞLARININ OLUŞTURULMASI.....	63
a.	Ağ Kullanıcı Ağının Oluşturulması.....	63
(1)	Ağ Kullanıcı Ağında Öbek Oluşturulması.....	64
(2)	Ağ Kullanıcı Düğümlerinin Oluşturulması.....	64

(3) Ağ Kullanıcı Arklarının Oluşturulması.....	64
b. Yol Kesici Ağının Oluşturulması.....	65
c. Örnek Ağ Oluşturma Uygulaması.....	66
2. DAL SINIR ALGORİTMASININ KOŞUM SONUÇLARI.....	70
a. Kesit Verildiğinde En Uygun Yöntemin Araştırılması.....	70
b. En Uygun Kesit Sayısının Belirlenmesi.....	77
3. MATEMATİKSEL MODELİN KOŞUM SONUÇLARI VE KARŞILAŞTIRMA.....	82
4. AĞIN BOYUTU VE OPERASYON SAYISINA BAĞLI OLARAK PROBLEMİN KARMAŞIKLIK ANALİZİ.....	86

ALTINCI BÖLÜM

SONUÇLAR VE GELECEK ÇALIŞMALAR

1. SONUÇLAR.....	89
2. GELECEK ÇALIŞMALAR.....	91
KAYNAKÇA.....	96
EKLER.....	101

TABLÖLER

Sayfa

Tablo 1.	Problem Tanımlama Çizelgesi.....	4
Tablo 2.	Literatür Çalışmalarının Problem Tanımlama Çizelgesi Kullanılarak Gösterilmesi.....	30
Tablo 3.	En Ucuz Ekleme 3 Yönteminin İlk Adımı.....	62
Tablo 4.	En Ucuz Ekleme 3 Yönteminin İkinci Adımı.....	62
Tablo 5.	Öbekteki Düğüm Sayısının Alt Sınırı ve Alt Sınıra Eklenerek Maksimum Düğüm Sayısının Belirlenme Kuralları.....	64
Tablo 6.	100 Düğümlü 4 Farklı Türdeki 10'ar Adet AK Ağında Tüm Kesitler Kullanılarak 6 Farklı Yöntemle Çözümün Analizi.....	73
Tablo 7.	500 Düğümlü 4 Farklı Türdeki 10'ar Adet AK Ağında Tüm Kesitler Kullanılarak 6 Farklı Yöntemle Çözümün Analizi.....	74
Tablo 8.	AK'nın 10 Düğümlü Olduğu Ağlarda Modelin Çözüm Sonuçları.....	83
Tablo 9.	AK'nın 20 Düğümlü Olduğu Ağlarda Modelin Çözüm Sonuçları.....	84
Tablo 10.	AK'nın 40 Düğümlü Olduğu Ağlarda Modelin Çözüm Sonuçları.....	85
Tablo 11.	Matematiksel Model ve Dal Sınır Algoritmasının Koşum Sonuçlarının Kıyaslanması.....	86
Tablo 12.	AK'nın Düğüm Sayısına Göre Ağlarda Dal Sınır Algoritmasının Optimal Çözüm Değerleri ve Operasyon Sayıları.....	88

ŞEKİLLER

Sayfa

Şekil 1.	İncelenecek Problemin Tanımlanması.....	7
Şekil 2.	AK ve HYK'nın Birleşik Ağı.....	11
Şekil 3.	Literatürdeki Ağ Kesme Problemlerinin Sınıflandırılması.....	13
Şekil 4.	Asimetrik Bilgi Ağları. (a) Gerçek Ağ, (b) Ağ Kullanıcının Ağ Hakkında Sahip Olduğu Bilgiler.....	24
Şekil 5.	İlk EKG'nin Bulunması ve EKG Üzerindeki HYK Düğümünün Belirlenmesi.....	47
Şekil 6.	2-5 Arkının Kesilmesi ve İkinci EKG'nin Belirlenmesi.....	48
Şekil 7.	6-7 Arkının Kesilmesi ve ÜS Değeri Değişmeden Uygun Bir Sonucun Bulunması.....	49
Şekil 8.	5-7 Arkının Kesilmesi ve Yeni EKG'nin Belirlenmesi.....	50
Şekil 9.	6-7 Arkının Kesilmesi ve ÜS Değerinin Güncellenmesi.....	51
Şekil 10.	Tüm Dallanmalar ve Budanmaların Tamamlandığı Dal Sınır Ağacı.....	51
Şekil 11.	1 Numaralı Düğüm Temel Alınarak Oluşturulan Kesit.....	53
Şekil 12.	2 Numaralı Düğüm Temel Alınarak Oluşturulan Kesit.....	53
Şekil 13.	3 Numaralı Düğüm Temel Alınarak Oluşturulan Kesit.....	53
Şekil 14.	4 Numaralı Düğüm Temel Alınarak Oluşturulan Kesit.....	54
Şekil 15.	5 Numaralı Düğüm Temel Alınarak Oluşturulan Kesit.....	54

Şekil 16.	6 Numaralı Düğüm Temel Alınarak Oluşturulan Kesit.....	54
Şekil 17.	HYK'nın Başlangıç Düğümünün (A) Diğer Tüm HYK Düğümlerine Olan Uzaklığı.....	55
Şekil 18.	AK Arklarının Uzunluklarının, HYK'nın A Düğümüne Olan Uzaklıkları ile Değiştirilmesi.....	55
Şekil 19.	Güncellenen AK Ağı Üzerinde Minimum Kesme.....	56
Şekil 20.	2. Öbeğe Atanan Düğümler.....	67
Şekil 21.	İlk 2 Öbeğin Bağlarının Tamamlanması ve Yeni Öbeğin Tanımlanması.....	67
Şekil 22.	Kalan Son Düğümün Son Öbeğe Atanması.....	68
Şekil 23.	AK Ağının Tamamlanması ve YK Düğümlerinin Oluşumu....	68
Şekil 24.	İlk Derinlikteki YK Bağlarının Oluşturulması.....	69
Şekil 25.	İkinci Derinlikteki YK Bağlarının Oluşturulması.....	69
Şekil 26.	Üçüncü Derinlikteki YK Bağlarının Oluşturulması ve Ağın Tamamlanması.....	69
Şekil 27.	AK'nın 100 Düğümlü Olduğu 10 Adet Düşük Derinlik Düşük Yoğunluklu Ağda Yöntemlerin Analizi.....	75
Şekil 28.	AK'nın 500 Düğümlü Olduğu 10 Adet Düşük Derinlik Düşük Yoğunluklu Ağda Yöntemlerin Analizi.....	76
Şekil 29.	500 Düğümlü Düşük Derinlikli Düşük Yoğunluklu AK Ağında Düz Çizgi Yöntemi Kullanılarak En Uygun Kesit Sayısının İncelenmesi.....	79

Şekil 30.	500 Döğümlü Düşük Derinlikli Düşük Yoğunluklu AK Ağında Düz Çizgi 2 Yöntemi Kullanılarak En Uygun Kesit Sayısının İncelenmesi.....	80
Şekil 31.	500 Döğümlü Düşük Derinlikli Düşük Yoğunluklu AK Ağında En Ucuz Ekleme 2 Yöntemi ile En Uygun Kesit Sayısının İncelenmesi.....	81
Şekil 32.	HYK'nın Kısıtlı Kaynak ile Ağ Kesmesi.....	92
Şekil 33.	HYK'nın AK'nın Konumunu Bilerek Ağ Kesmesi.....	92
Şekil 34.	AK ve HYK'nın Birbirlerinin Konumundan ve Amacından Haberdar Olduğu Ağ Kesme Problemi.....	93
Şekil 35.	AK'nın da HYK gibi Ağ Kesebildiği Ağ Problemi.....	93
Şekil 36.	İki Tane HYK ile En Kısa Güzergâhın Kesilmesi.....	94
Şekil 37.	HYK'nın Çok Ürünlü Taşıma Yapılan Ağda Kesme İşlemi Yapması.....	94
Şekil 38.	HYK'nın Bir Öncelik Sırasına Göre Ağ kesmesi.....	95

KISALTMALAR LİSTESİ

A1,...,An: Ağ kullanıcının sahip olabileceği amaçların herbiri.

AK: Ağ kullanıcı

AS: Alt sınır

B: Belirgin (kesin bilgiye sahip olunması).

b-h: Ağ kullanıcın ağındaki başlangıç düğümü – hedef düğümü

BY: Bilgi Yok.

Ç: Çoklu (Birden fazla kaynak olması ya da birden fazla depo olması durumlarında)

ÇAK: Çoklu Ağ-Kullanıcı (Ağ üzerinde birden fazla Ağ kullanıcı olması durumu)

ÇÜ: Çok Ürünlü (Örneğin bir düğümün bir taraftan gıda talep ederken başka tarafa silah arz etmesi durumu).

ÇYK: Çoklu Yol Kesici (Birden fazla yol kesici olması durumu)

DÖA: Derinlik öncelikli arama

eds: Öbekteki düğüm sayısının alt sınırının üzerine eklenebilecek maksimum düğüm sayısı

EKG: En kısa güzergâh

g: Güzergâh

GK: Güzergahlâr kümesi

HYK: Hareketli Yol Kesici

LG: Lagrangian Gevşemesi

S(+): Ağ kullanıcı savunma yapabilir.

S(-): Ağ kullanıcı savunma yapamaz.

SRL: Yol kesicinin yolları bir öncelik sırasına göre kesmesi. (Örneğin öncelikli olarak en kısa güzergâh üzerindeki bir arkın/düğümün kesilmesi durumu)

SRZ: Yol kesicinin yolları bir öncelik sırası olmadan kesmesi.

O: Olasılıklı (kesin bilgiye sahip olunmaması)

oas: Öbekteki düğüm sayısının alt sınırı

SYK: Sabit Yol-Kesici

T: Tekli (bir kaynak ya da bir depo olması durumlarında)

T(-): Ağ kullanıcı kesilen yolu tamir edemez.

TAK: Tek Ağ-Kullanıcı olması

Tk: Ağ kullanıcı kesilen yolu kaynak kısıtına bağlı olarak tamir edebilir.

Tkz: Ağ kullanıcı kesilen yolu kaynak ve zaman kısıtına bağlı olarak tamir edebilir.

TÜ: Ağ kullanıcı ağı üzerinde tek tip ürün sevkiyatı yapmaktadır.

TYK: Tek Yol Kesici olması durumu.

Tz: Ağ kullanıcının kesilen yolu zaman kısıtına bağlı olarak tamir edebilir.

Y1,...,Ym: Yol kesicinin sahip olabileceği amaçların herbiri.

YK: Yol kesici

ÖNSÖZ

Çalışmada ağ üzerindeki belirli bir başlangıç ve hedef düğümü arasında en kısa güzergâh üzerinden iletişim kurulmaya çalışıldığı varsayılmaktadır. Bu iki düğüm arasındaki iletişimi kesecek hareketli bir yol kesici için başlangıçta üzerinde bulunduğu düğümden başlayan en kısa güzergâh aranmaktadır. Ağ kullanıcı ve yol kesici ağ üzerinde birbirinden farklı ark ve düğümleri kullanabilmektedir.

Farklı alanlardaki birçok gerçek hayat problemi ele alındığında, ağda istenilmeyen iletişime engel olmak için hareketli bir yol kesiciye ihtiyaç olduğu görülmüştür. Buna rağmen literatürdeki çalışmalarda bu kavramın kullanılmadığı görülerek bu kavramın önemli bir yer işgal ettiği bu çalışma gerçekleştirilmiştir.

Çalışmada kullanılan farklı boyut ve yapılarıdaki ağların oluşturulması için ve çözümde kullanılan yöntemlerin uygulanması için programlar yazılmıştır. Bu programlar tezin teoride kalmasına engel olarak gerçek hayatta uygulanmasını mümkün kılmaktadır.

Bu çalışmanın gerçekleşmesine olanak sağlayan Kara Harp Okulu Komutanlığı'na ve engin tecrübeleri ile bana sağladıkları sınırsız destek için danışmanlarım Prof.Dr. Levent KANDİLLER ve Yrd.Doç.Dr. Haluk AYGÜNEŞ'e, çok kıymetli yardım ve önerileri için Doç.Dr. İbrahim AKGÜN ve Yrd.Doç.Dr. Altan ÖZKİL'e, tez savunması aşamasındaki değerli görüşleri için Prof.Dr. Cevriye GENCER ve Dr. Hünkar TOYOĞLU'na teşekkürü bir borç bilirim.

Annem E. Fatma ve babam Mustafa KABARCIK ve abilerim Zübeyir, Haşim ve Fahri KABARCIK'a hayatımın her döneminde olduğu gibi bu tez çalışması sırasında da bana verdikleri destek için sonsuz teşekkürlerimi sunarım.

Benimle beraber tez çalışmalarının tüm zorluklarına sabırla göğüs geren, benim için hayatındaki herşeyi ikinci plana atarak desteğini hep hissettiren sevgili eşim Arzu KABARCIK'a minnettarlığımı sunarım.

GİRİŞ

Yaşamı etkileyen bir çok faaliyet ve olay ağ yapısındadır. Telefon hatları, karayolları, demiryolları, kanalizasyonlar, su kanalları, petrol boru hatları gibi ağlar insanlar tarafından oluşturulmuştur. İnsanlar arasındaki sosyal ilişkiler, hastalıkların virüsler yoluyla yayılması, orman yangınları, sel felaketleri gibi durumlarda oluşan ağlar ise insanlar tarafından tasarlanmayan sanal ağlardır. Bu ağlar dinamiktir ve somut bağlantı yolları içermezler.

Ağlar, düğümler ve bu düğümleri birbirine bağlayan arklardan oluşur. Düğümler arasındaki iletişim arklar üzerinden sağlanır. Genelde ağlar bu iletişimi sağlamak amacıyla oluşturulur. Ses iletimi olmadan telefon hatlarından, tren kullanılmadan demiryollarından ya da akaryakıt taşınmadan petrol boru hatlarından bir ağ olarak bahsetmek eksiklik olacaktır. Ancak ağ üzerindeki iletişim her zaman olması istenilen bir durum değildir. Örneğin askeri operasyonlarda düşman birlikleri arasındaki her türlü iletişimin kesilmesine çalışılır. Orman yangınlarında ateşin yayılmasını engellemek için itfaiye araçları kritik düğümlere gönderilir. Sel felaketlerinde suyu kontrol edebilmek için suyun akış yollarına setler kurulur. İletişimi kesmek için ark ya da düğümlerin kullanılamaz hale getirilmesi gerekir.

Ağ kesme problemlerinde iki adet oyuncudan bahsedilebilir. İlk oyuncu kesilmek istenen ağ üzerindeki iletişimi sürdürmeye çalışır. Bu çalışma kapsamında bu oyuncuya ağ kullanıcı (AK) ismi verilmiştir. Diğer oyuncu ise ağı keserek AK'nın iletişimine engel olmaya çalışır. Bu oyuncuya ise yol kesici (YK) ismi verilmiştir.

İletişimin kesilmesi için sabit ve hareketli olmak üzere iki tip YK kullanılabilir. Örneğin askeri faaliyetlerde yola mayın döşenmesi, mobese kamera ile bir bölgenin gözetlenmesi, hava sahasının radar yardımıyla takip edilmesi gibi durumlarda YK sabittir. Sabit yol kesici (SYK) yol üzerinde bulunduğu süre içerisinde yolu kesmektedir.

YK devre dışı bırakılan yol üzerinde sabit kal(a)mayabilir. Mesela uçak sortileriyle saldırılan bir ağda, uçağın imha edilen yol üzerinde sabit beklemesi düşünülemez. Uçak bir yolu devre dışı bıraktıktan sonra imha

edebileceği diğer yol seçenekleri arasından seçim yapmak ve ona göre güzergâhını belirlemek durumundadır. Buradaki hareketli yol kesici (HYK) kendine ait bir ağa sahiptir.

Literatürde ele alınan ağ kesme problemlerinde genellikle YK'nın sabit AK'nın ise hareketli olduğu durumların ele alındığı gözlemlenmiştir. Bu tez çalışmasında yolların HYK tarafından devre dışı bırakıldığı bir ağ kesme problemi ele alınmaktadır. Problemde ele alınan ağ, içiçe geçmiş iki ağdan oluşmaktadır. Ağlardan biri AK tarafından, diğeri ise HYK tarafından kullanılmaktadır. HYK ağı üzerindeki düğümler AK ağındaki arkların ya da düğümlerin üzerinde konuşlanmıştır. Bu düğümler imha edilmeye aday noktalar. Bu çalışmada AK'nın başlangıç (b) ve hedef (h) düğümleri arasındaki tüm güzergâhlarını imha etmek için HYK'nın kullanacağı en kısa güzergâh bulunmaya çalışılmaktadır.

AK'nın başarılı olması demek; $b-h$ düğümleri arasındaki yolculuğunu tamamlayabilmesi demektir. Bu çalışmada AK'nın hedefine ulaşmak için her zaman en kısa güzergâhı (EKG) kullanacağı varsayılmaktadır. HYK'nın da AK'ya engel olmak için kesme işlemini en kısa sürede gerçekleştirmesi yani ağın kendi kullanabildiği bölümdeki EKG'yi kullanması gerekecektir.

Çalışmanın I. bölümünde problemin tanımı yapılmaktadır. Ayrıca hareketli yol kesici kavramı ayrıntısıyla anlatılmaktadır. II. bölümde literatürde konuyla ilgili yapılan çalışmalardan bahsedilmektedir. III. bölümde problemin çözümü için oluşturulan matematiksel modeller anlatılmaktadır. IV. bölüm ise dal sınır yönteminin problemin çözümüne nasıl uyarlandığını ve dal sınır yöntemi üzerinde yapılan iyileştirme çalışmalarını içermektedir. V. bölümde problemin çözümü için oluşturulan yöntemlerin deneneceği her tip ve boyuttaki ağların oluşturulması için yazılan programın algoritması anlatılmakta ve ardından dal sınır algoritması ve matematiksel model için oluşturulan programların koşum sonuçlarından bahsedilmektedir. Son olarak bu bölümde problemin karmaşıklığı analiz edilmektedir. VI. bölümde ise çalışmada elde edilen sonuçlar ve gelecek çalışmalardan bahsedilmektedir.

BİRİNCİ BÖLÜM

PROBLEM TANIMI

1. PROBLEM TAKSONOMİSİ

Ağ kesme problemlerinin incelenmesinde bir çok etmen göz önüne alınmaktadır. Problemlerin daha kolay sınıflandırılabilmesi için bu çalışma kapsamında Tablo 1’de görülen taksonomi oluşturulmuştur. Taksonomi için oluşturulan tabloda 22 adet etmen belirlenmiştir. İlk 11 etmen YK için, 12-19 arasındaki etmenler AK için, 20-22 arasındaki etmenler ise arkaların durumu için tanımlanmıştır.

Tablo 1’de her bir etmen numarasının altında iki adet dikdörtgen kutu görülmektedir. İlk kutuda etmenin ne olduğu tanımlanmaktadır. İkinci kutu ise etmenin alabileceği değerleri göstermektedir. Örneğin 1 numaralı etmenin altındaki ilk kutuda bulunan “Tipi” değişkeni 1. etmenin YK’nın tipini belirttiğini göstermektedir. Aynı etmenin altındaki ikinci kutuda bulunan SYK ve HYK değerleri ise YK’nın tipinin neler olabileceğini göstermektedir. İkinci sırada bulunan kutulardaki değerler kısaltmalar ile gösterilmektedir. Bu kısaltmaların açılımı tablonun altında belirtilmiştir. Ancak 5, 6, 8 ve 16 numaralı etmenlerin altlarındaki ikinci kutularda birden fazla kısaltma değeri bulunmaktadır. 5 numaralı etmenin altındaki kutuları ele alalım. Etmenin altındaki ilk kutuda “AK’nın Başlangıç Düğümünü Biliyor mu?” sorusu sorulmaktadır. Bu etmen YK’nın AK’nın başlangıç düğümü ile ilgili neler bildiğini sorgulamaktadır ve altındaki ikinci kutuda da görüldüğü gibi iki adet bilgi içermektedir. Bu bilgiler iki sütun halinde oluşturulmuştur. İlk sütunda O, B, BY, YB (Olasılıklı, Belirgin, Bilgi Yok, Yanlış Bilgi) değerleri, ikinci sütunda ise T, Ç, BY, YB (Tek Kaynaklı, Çok Kaynaklı, Bilgi Yok, Yanlış Bilgi) değerleri bulunmaktadır. İlk sütun AK’nın başlangıç düğümü ile ilgili eldeki bilgilerin ne kadar net olduğunu, ikinci sütun ise AK’nın kaç adet başlangıç düğümü olduğunu göstermektedir.

Tablo 1. Problem Tanımlama Çizelgesi

YOL KESİCİ (YK)											AĞ KULLANICI (AK)								YOLLARIN DURUMU		
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
Tipi	Amacı	Çoklu Tekli	Kaynak Kısıtı var mı?	AK'nın Başlangıç Düğümünü Biliyor mu?	AK'nın Hedef Düğümünü Biliyor mu?	AK'nın Amacını Biliyor mu?	AK'nın Konumunu ve Sayısını Biliyor mu?	Kesilmenin Olasılığı	Kesime Sıralı mı?	Yol Bilgileri Net mi?	Amacı	Çoklu Tekli	Çok Ürünlü Tek Ürünlü (Taşıma)	YK'nın Amacını Biliyor mu?	YK'nın Konumunu ve Sayısını Biliyor mu?	Tamir Yapabiliyor mu?	Savunma Yapabiliyor mu?	Yol Bilgileri Net mi?	Kapasiteleri Biliyor mu?	Uzunlukları Biliyor mu?	Kesme / Oluşturma Maliyetleri Biliyor mu?
SYK HYK	Y1 Y2 ... Ym	ÇYK TYK	K(-) K(+)	O B BY YB T Ç BY YB	O B BY YB T Ç BY YB	O B BY YB	O B BY YB T Ç BY YB	O B BY YB	SRL SRZ	O B BY YB	A1 A2 ... An	ÇAK TAK	ÇÜ TÜ	O B BY YB	O B BY YB T Ç BY YB	T(-) Tk Tz Tkz	S(-) S(+)	O B BY YB	O B BY YB	O B BY YB	O B BY YB

KISALTMALAR: (Alfabetik sırayla)

A1,...,An : AK'nın amaçlarının herbiri. (Etmen no:12) B : Belirgin (kesin bilgiye sahip olunması). (Etmen no : 5, 6, 7, 8, 9, 11,15, 16, 19, 20, 21, 22) BY: Bilgi Yok. (Etmen no : 5, 6, 7, 8, 9, 11,15, 16, 19, 20, 21, 22) Ç : Çoklu (Örnek: YK 5 numaralı etmende AK'nın birden fazla başlangıç düğümüne sahip olduğu, 6 numaralı etmende birden fazla hedef düğümüne sahip olduğunu bilmektedir) (Etmen no : 5, 6, 8, 16) ÇAK : Çoklu Ağ Kullanıcı (Ağ üzerinde birden fazla AK olması durumu) (Etmen no : 13) ÇÜ : Çok Ürünlü (Örnek: bir düğümün bir taraftan gıda talep ederken başka tarafa silah arz etmesi) (Etmen no : 14) ÇYK : Çoklu Yol Kesici (Birden fazla YK olması durumu) (Etmen no : 3) HYK : Hareketli Yol Kesici (Etmen no : 1)	K(-) : YK'nın kaynak kısıtı yok. (Etmen no: 4) K(+) : YK'nın sınırlı kaynağı var. (Etmen no : 4) O : Olasılıklı sonuç (kesin bilgiye sahip olunmaması) (Etmen no : 5, 6, 7, 8, 9, 11,15, 16, 19, 20, 21, 22) S(-) : AK savunma yapamaz. (Etmen no : 18) S(+) : AK savunma yapabilir. (Etmen no : 19) SRL : YK'nın yolları bir öncelik sırasına göre kesmesi. (Örnek: öncelikli olarak EKG'deki bir arkın kesilmesi) (Etmen no : 10) SRZ : YK'nın yolları bir öncelik sırası olmadan kesmesi. (Etmen no : 10) SYK : Sabit Yol Kesici (Etmen no : 1) T : Tekli (Örnek: YK 5 numaralı etmende AK'nın tek başlangıç düğümüne sahip olduğu, 6 numaralı etmende tek hedef düğümüne sahip olduğunu bilmektedir) (Etmen no : 5, 6, 8, 16)	T(-) : AK kesilen yolu tamir edemez. (Etmen no : 17) TAK : Tek Ağ Kullanıcı olması (Etmen no : 13) Tk : AK kesilen yolu kaynak kısıtına bağlı olarak tamir edebilir. (Etmen no : 17) Tkz : AK kesilen yolu kaynak ve zaman kısıtına bağlı olarak tamir edebilir. (Etmen no : 17) TÜ : AK ağı üzerinde tek tip ürün sevkiyatı yapılmaktadır. (Etmen no : 14) TYK : Tek Yol Kesici olması durumu. (Etmen no : 3) Tz : AK kesilen yolu zaman kısıtına bağlı olarak tamir edebilir. (Etmen no : 17) YB: Yanlış bilgiye sahip olma durumu. (Etmen no : 5, 6, 7, 8, 9, 11,15, 16, 19, 20, 21, 22) Y1,...,Ym : YK'nın amaçlarının herbiri. (Etmen no : 2)
--	---	--

Tablodaki 2 ve 12 numaralı etmenler sırasıyla YK ve AK'nın amaçlarıdır. Bu etmenlerin çok fazla değer alabiliyor olması nedeniyle tablonun altındaki kısaltmalar bölümünde gösterilmesi zordur. O nedenle aşağıda bu etmenlerin alabileceği değerler ve anlamları listelenmiştir.

2 numaralı etmende bulunan YK'nın amaç değerleri ve anlamları:

- Y1: En kısa sürede (en kısa güzergâh üzerinden) ağın kesilmesi.
- Y2: Minimum maliyet ile ağın kesilmesi.
- Y3: AK'nın maksimum akışını minimize etmek.
- Y4: AK'nın en kısa güzergâhını maksimum yapmak.
- Y5: En kısa sürede maksimum sayıda AK güzergâhı kesmek.
- Y6: En az kaynak ile en kısa sürede kesme işlemi gerçekleştirmek
- Y7: En kısa sürede maksimum akışı minimize etmek.
- Y8: Minimum kesme ile ağın kesilmesi
- Y9: k adet ark keserek maksimum akışı minimize etmek.
- Y10: Ağdaki düğümleri keserek bağlantıyı minimize etmek.
- Y11: Ağ kullanıcının tipine göre ağ kullanıcıyı her düğüme ulaştırmaya çalışmak ya da tüm bağlantılarını kesmek.
- Y12: Ağ üzerinde minimum farkedilecek düğüm ve arkları kullanarak ağa saldırmak.
- Y13: En güvenli güzergâhın güvenliğinin minimizasyonu.
- Y14: Tamir süresini maksimize edecek şekilde ağı kesmek.
- Y15: Kesim sonucunda en çok ark ile bağlantı sağlayan düğümün ark sayısını minimize etmek

- Y16: Tamir maliyetini maksimize edecek şekilde ağı kesmek.
- Y17: En az kaynak kullanarak maksimum sayıda AK yakalamak.
- Y18: Kestiği arklar ile AK'nın algılanmasını maksimize etmek.
- Y19: AK'nın taşıma sonrası elde edeceği kârı minimize etmek.

12 numaralı etmende bulunan AK'nın amaç değerleri ve anlamları:

- A1: $b-h$ düğümleri arasını en kısa sürede katetmek.
- A2: $b-h$ düğümleri arasında maksimum taşıma gerçekleştirmek.
- A3: $b-h$ düğümleri arasında en kısa sürede maksimum taşıma gerçekleştirmek.
- A4: Birden fazla başlagıç düğümü ve birden fazla hedef düğümü arasında maksimum taşıma gerçekleştirmek.
- A5: Birden fazla başlagıç düğümü ve birden fazla hedef düğümü arasında en hızlı taşımayı gerçekleştirmek.
- A6: YK'ya yakalanma ihtimalinin en düşük olduğu ark ve düğümleri kullanarak hedefe ulaşmak.
- A7: Ağdaki tüm düğümlere ulaşmak
- A8: Ark ve düğüme göre YK'yı maksimum oranda farkedecek şekilde ağı tasarlamak.
- A9: Stokastik bir güzergâh üzerinden $b-h$ düğümleri arasını katetmek.
- A10: Ağ üzerindeki taşıma sonucu maksimum kârı elde etmek.
- A11: Ağda ilerlerken YK gibi kesme işlemi yapmak.

2. PROBLEMİN AYRINTILARI

Tablo 1'e göre problemleri tanımlarken sadece etmen numaraları ve bu numaraların altındaki ikinci kutu içerisindeki değerlerden biri kullanılır. Bu çalışma kapsamında incelenecek olan problemin özellikleri Şekil 1'de gösterilmektedir. Problemden dikkate alınmayan etmenlerin alacağı değerin kutusu boş bırakılmıştır.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
HYK	Y1	TYK	K(-)	B-T	B-T	B		B	SRL	B	A1	TAK	TÜ			T(-)	S(-)	B		B	

Şekil 1. İncelenecek Problemin Tanımlanması

Bu tez çalışması kapsamında ele alınacak problemin Şekil 1'de verilen özellikleri etmen numaralarına göre aşağıda anlatılmaktadır.

- **1 numaralı etmen:** YK'nın tipini içermektedir. Çalışmada YK hareketli olduğu için "HYK" değerini almıştır.
- **2 numaralı etmen:** HYK'nın amacını içermektedir. HYK en kısa sürede AK'nın *b-h* düğümleri arasındaki bağına koparmayı amaçladığı için "Y1" değerini almıştır.
- **3 numaralı etmen:** Bir tane mi yoksa birden çok mu HYK olduğu bilgisini içermektedir. Çalışmada tek HYK olduğu için "TYK" değerini almıştır.
- **4 numaralı etmen:** HYK'nın kaynak kısıtı olup olmadığını sorgulamaktadır. HYK'nın sonsuz kaynağa sahip olduğu kabul edildiği için "K(-)" değerini almıştır.
- **5 numaralı etmen:** AK'nın başlangıç düğümünün konum ve sayısının HYK tarafından bilinip bilinmediğini sorgulamaktadır. HYK başlangıç düğümünün konumunu bildiği için ve bir tane başlangıç düğümü olduğu için etmen "B-T" değerini almıştır.
- **6 numaralı etmen:** AK'nın hedef düğümünün konum ve sayısının HYK tarafından bilinip bilinmediğini sorgulamaktadır. HYK hedef

düğümünün konumunu bildiği için ve bir hedef düğümü olduğu için etmen "B-T" değerini almıştır.

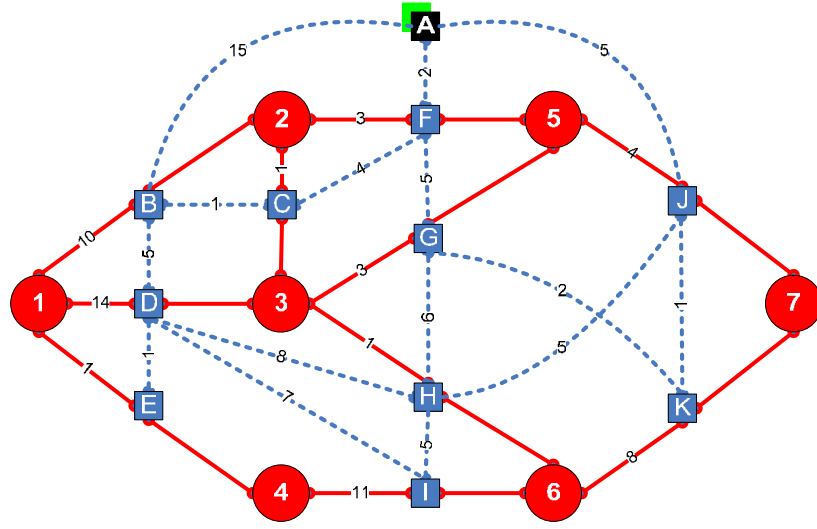
- **7 numaralı etmen:** AK'nın amacının HYK tarafından bilinip bilinmediğinin bilgisini içermektedir. Bu çalışmada HYK, AK'nın hedefine en kısa güzergâh üzerinden ulaşmak istediğini bildiği için kendisi de en kısa güzergâh kullanarak AK'nın önünü kesmek istemektedir. Dolayısıyla AK'nın amacı HYK tarafından bilindiği için etmenin değeri "B" olmaktadır.
- **8 numaralı etmen:** AK'nın konumunun ve sayısının HYK tarafından bilinip bilinmediğini sorgular. Burada HYK *b-h* düğümleri arasındaki iletişimi en kısa güzergâh üzerinden kesmeye odaklanmıştır. Dolayısıyla bu etmenin içerebileceği bilgilerle ilgilenmemektedir. O nedenle bu etmen boş bırakılmıştır.
- **9 numaralı etmen:** HYK'nın kesme işlemi sonunda kısmen kesilme durumu için olasılıklı anlamındaki "O" değerini, tamamen kesilme için belirgin anlamındaki "B" değerini, kesilme işleminin ardından sonucun bilinmediği durumlarda bilgi yok anlamındaki "BY" değerini ve yanlış bilgiye sahip olunan durumlar için "YB" değerini içerir. Çalışmada kesme işleminin sonucunda kesilen nokta net bir şekilde devre dışı kaldığı için etmen değeri "B" olmuştur.
- **10 numaralı etmen:** HYK kesme işleminde bazı önceliklere sahip mi sorusunun cevabını verir. Bu çalışmada problemin çözümü için kullanılan dal sınır algoritmasında öncelikli olarak AK'nın en kısa güzergâhı üzerinde kesme işlemi yapıldığı için etmenin aldığı değer "SRL" olmuştur.
- **11 numaralı etmen:** HYK'nın ağ üzerindeki arklar ve düğümler hakkındaki bilgilerinin kesinliğini içerir. Bu çalışmada HYK arkların uzunluklarıyla ilgilenmektedir ve uzunluk bilgilerini net olarak bilmektedir. O nedenle etmenin aldığı değer "B" olmuştur.

- **12 numaralı etmen:** Bu etmen AK'nın amacını içermektedir. AK en kısa güzergâh üzerinden b düğümünden h düğümüne gitmeyi amaçladığı için bu etmenin aldığı değer "A1" olmuştur.
- **13 numaralı etmen:** AK'nın tek mi yoksa birden fazla mı olduğu bilgisini içerir. Bu çalışmada tek AK vardır ve bu nedenle etmen "T" değerini almıştır.
- **14 numaralı etmen:** AK'nın bir düğümünün başka düğümlere gıda gönderirken, kendisinin de başka düğümlerden silah talep etmesi ağda çok ürünlü taşımaya bir örnektir. Ancak ağdaki taşıma bir düğümün başka bir düğüme gıda göndermesinden ibaretse ağda tek ürünlü taşıma gerçekleşiyor demektir. Bu çalışmada tek ürünlü taşıma olduğu için etmen "TÜ" değerini almıştır.
- **15 numaralı etmen:** HYK'nın amacının AK tarafından bilinip bilinmediği bilgisini içerir. Bu çalışmada bu durumla ilgilenilmediği için etmen değeri boş bırakılmıştır.
- **16 numaralı etmen:** HYK'nın konumunun ve sayısının AK tarafından bilinip bilinmediğini sorgular. Bu çalışmada bu durumla ilgilenilmediği için etmen değeri boş bırakılmıştır.
- **17 numaralı etmen:** AK'nın kesilen yolları tamir edip edemeyeceği bilgisini içerir. Bu çalışmada AK'nın tamir yeteneği olmadığı için etmen "T(-)" değerini almıştır.
- **18 numaralı etmen:** HYK'nın kesmek istediği noktalarda AK'nın savunma yapıp yapamayacağı bilgisini içerir. Bu çalışmada AK'nın savunma yeteneği olmadığı için etmen "S(-)" değerini almıştır.
- **19 numaralı etmen:** Ağ üzerindeki arklar ve düğümler hakkında AK'nın bilgilerinin kesinliği ile ilgili bilgiyi içerir. Bu çalışmada AK'nın doğru ark bilgilerine sahip olduğu kabul edildiği için etmen "B" değerini almıştır.

- **20 numaralı etmen:** Çalışmada ark kapasitelerinin bilinip bilinmediği bilgisini içerir. Bu çalışmada ark kapasiteleriyle ilgilenilmediği için bu etmenin değeri boş bırakılmıştır.
- **21 numaralı etmen:** Arkların uzunluklarının bilinip bilinmediğini sorgular. Bu çalışmada ark uzunlukları biliniyor olduğu için etmen "B" değerini almıştır.
- **22 numaralı etmen:** Arkların kesilme maliyetlerinin bilinip bilinmediği bilgisini içerir. Bu çalışmada HYK'nın sonsuz kaynağı olduğu ve arkların kesilme maliyetlerinin eşit olduğu kabul edildiği için arkların kesilme maliyetleriyle ilgilenilmemiştir. O nedenle bu etmenin alacağı değer boş bırakılmıştır.

Bu çalışmada ayrıca belirtilmesi gereken önemli bir durum ağ üzerinde AK ve HYK'nın kullanabileceği düğüm ve arkların farklı olmasıdır. Ağ iki oyuncu arasında paylaştırıldığı için ağın yapısı içiçe girmiş iki ağ gibi gözükmemektedir. Şekil 2'de görülen ağ bu duruma bir örnektir. Şekildeki kare düğümler HYK'nın, yuvarlak düğümler ise AK'nın kullanabileceği düğümlerdir. HYK kesikli çizgiler ile gösterilen arkları, AK ise düz çizgiler ile gösterilen arkları kullanabilmektedir. Her bir arkın uzunluğu üzerinde gösterilmektedir.

Şekil 2'deki örneğe göre, AK, başlangıç (1) düğümünden yola çıkarak, düz çizgiler ile gösterilen ağ üzerinden, hedef (7) düğümüne en kısa güzergâhı kullanarak gitmektedir. HYK, başlangıçta, merkez (A) düğümünde bulunmaktadır ve kesikli çizgiler ile gösterilen ağ üzerinden, imha edeceği noktalara (kare düğümlere) gitmektedir. HYK'nın amacı en kısa güzergâhını kullanarak AK'nın başlangıç ve hedef düğümleri arasındaki bağlantıyı kesmektir. Bu çalışmada HYK ve AK'nın en kısa güzergâhı denildiğinde her iki oyuncunun da kendi ağları üzerindeki en kısa güzergâh kastedilmektedir.



Şekil 2. AK ve HYK'nın Birleşik Ağı

İKİNCİ BÖLÜM

LİTERATÜR ARAŞTIRMASI

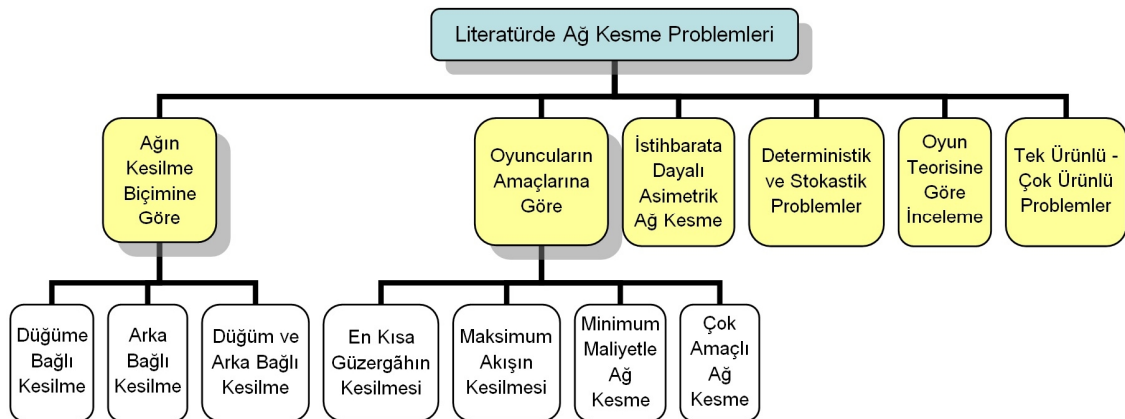
1. LİTERATÜRDE AĞ KESME PROBLEMLERİ

Ağ kesme problemlerinin literatürde görüldüğü ilk çalışmalardan biri Harris ve Ross'un (1955) çalışmasıdır. Bu çalışmada Sovyetler Birliği ile Doğu Avrupa ülkeleri (Polonya, Çekoslovakya, Avusturya ve Doğu Almanya) arasında oluşturulan demiryolu ağı kapasite kısıtlarına göre analiz edilmiştir. Amaç, minimum kesme ile ağı başlangıç (b)-hedef (h) düğümleri arasındaki bağı koparmaktır. Daha sonra Ford ve Fulkerson (1956) maksimum akış / minimum kesme teoremini ortaya atmıştır. Bu teoreme göre b - h düğümleri arasındaki maksimum akış, bağı koparmak için kesilen minimum kapasitedeki arkların toplamına eşittir. Bu teorem ile Harris ve Ross'un çalışmasının başka bir boyutu ortaya çıkmıştır.

Ağ kesme problemleriyle ilgili ilklerden biri sayılabilecek diğer bir çalışma ise Wollmer (1964) tarafından gerçekleştirilmiştir. Çalışmanın amacı; b - h düğümleri arasındaki maksimum akışta en büyük azalmayı sağlayacak k adet arkın bulunmasıdır. McMasters ve Mustin (1970) bu çalışmanın devamı niteliğinde bir çalışma ortaya koymuşlardır. Çalışmalarında kesilecek ark sayısının YK'nın elindeki kaynağa bağlı olduğunu belirtmişlerdir. Amaç ağıdaki akışı minimize etmek için kaynak kısıtına bağlı olarak kesilecek en uygun ark kümesini belirlemektir. Çözüm için arkların kesilme maliyetlerini ve pozitif değerli minimum kapasitelerini göz önüne alan bir algoritma oluşturulmuştur. Her iki çalışmanın da uygulanması küçük boyutlu ağlarda mümkün olmaktadır. Daha sonra Boyle (1988), McMasters ve Mustin'in çalışmasının dualini geliştirerek düzlemsel ağlarda b - h düğümleri arasındaki güzergahları numaralandırmıştır. Fakat bu çalışmada oluşturulan yöntem de yine küçük boyutlu ağlarda uygulanabilmektedir. Bu üç çalışma ağ kesme problemlerindeki bir çok çalışmaya yön vermesi nedeniyle önemli çalışmalardır.

Ağ kesme problemleri bir çok uygulama alanında geniş çaplı yer bulmaktadır. Kennedy (2009) doktora tez çalışmasında ağ modellerinin çok farklı alanların analizinde temel oluşturduğunu vurgulamaktadır. Çizgelerin, tedarik zincirlerinin, ulusal altyapı tesislerinin, sosyal ilişki ve organizasyonların çalışılması ve analizi farklı alanlara birkaç örnek olarak verilmiştir. Uygulamaya konulan ağ kesme problemlerinin en yaygın bilinenlerinden biri Assimakopoulos'un (1987) çalışmasıdır. Amerika'da bir yıl içerisinde hastanede enfeksiyon kaparak ölen insan sayısı AIDS, göğüs kanseri ve araba kazaları sonucu ölen insan sayısının toplamından fazladır. Assimakopoulos çalışmasında hastanelerdeki enfeksiyon kapma olaylarını önlemek için bir karar destek sistemi geliştirmiştir. Enfeksiyon ağındaki yolların kesilmesi esasına dayanan sistem Atina'da bazı hastanelerde uygulanmaktadır. Bu sistem ağ kesme problemlerinin uygulanabileceği farklı alanlara bir örnek olarak önemli bir çalışmadır.

Literatürde incelenen çalışmalarda ağ kesme işleminin düğüme göre, arka göre ya da hem düğüm hem de arka göre yapıldığı çalışmalara rastlanmıştır. Ayrıca problemler oyuncuların amaçları, bilgilerin olasılıklı olması, tek ürünlü veya çok ürünlü olma, sürekli veya kesikli olma gibi yönlerden de ele alınmıştır. Bu bölümde çalışmalar sınıflandırılarak anlatılacak ve böylece problemin farklı yönlerden nasıl ele alındığı gösterilmeye çalışılacaktır. Ağ kesme problemleri ile ilgili literatürdeki çalışmalar Şekil 3'te görüldüğü gibi sınıflandırılarak analiz edilmiştir.



Şekil 3. Literatürdeki Ağ Kesme Problemlerinin Sınıflandırılması

a. Düzüm, Ark ve Güzergâha Bağlı Ağ Kesme Problemleri

Ağı parçalara ayırma işlemi ağ üzerindeki düğümleri ve/veya arkları imha ederek gerçekleştirilir. Sosyal ağlarda ya da bulaşıcı hastalıkların yayılma ağlarında düğümler arasında iletişimi sağlayan somut arklardan bahsedilemez. Bu durumda ağ kesme işlemleri için genellikle ark yerine düğümlere yönelinmektedir. Somut arklar içeren ağlarda ise hem düğüm hem de ark ağ kesme işlemlerinde kullanılabilmektedir.

(1) Düzüme Bağlı Ağ Kesme Problemleri

Ağ üzerindeki kritik düğümlerin belirlenmesi ağların yapısal karakteristiğini ve bağlantı özelliklerini anlamak açısından önemli bir yer tutmaktadır. Aruselman ve diğerlerinin (2009) çalışmasında bahsedildiği gibi sosyal ağlardaki ilişkilerin koparılmasında, terör ağlarındaki iletişimi minimize etmek için devre dışı bırakılacak kişilerin seçilmesinde, telekomünikasyon ağlarında virüsün yayılmasına engel olmak için bazı düğümlerin çevrim dışı bırakılmasında, kablolu telekomünikasyon ağlarında iletişimi kesmek için bazı düğümlerin kapatılmasında, acil durumlarda malzeme ve insan tahliyesi için kullanılacak düğümlerin belirlenmesinde, bir felaket sonrasında alternatif trafik güzergâhları oluşturulmasında kritik düğüm problemi uygulanabilmektedir. Aruselman ve diğerleri düğümler arasındaki bağlantı sayısını minimize etmek için kesilecek kritik düğümleri belirlemeyi amaçlamaktadırlar. Kritik düğüm probleminin NP-tam olduğunu göstermiş ve çözüm için bir tamsayı doğrusal programlama modeli geliştirmişlerdir.

Summa ve diğerleri (2011) kritik düğüm problemini içeren çalışmalarını hiyerarşik bir ağaç şeklindeki çizgeler üzerinde gerçekleştirmektedirler. Çizgedeki her düğüm kendi üstündeki ana düğüm ile ve altındaki çocuk düğüm/düğümler ile bağlantılıdır. Her düğümün bir ağırlık değeri bulunmaktadır. Düğümler arasındaki her bağlantıya da negatif olmayan bir değer verilmiştir. Çalışmadaki amaç kritik düğümler kesildikten sonra geriye kalan düğüm çiftleri arasındaki toplam bağlantı değerini minimize etmektir. Kesilen düğümlerin boyutu problemin bir kısıtıdır. Ayrıca çalışmada düğüm ağırlıkları ve bağlantı değerleri göz önünde bulundurularak problemin karmaşıklığı analiz edilmektedir.

Borgatti'nin (2006) çalışması da ağdaki kritik düğümlerin kullanılmasına ilginç bir örnektir. Çalışmada sosyal ağ üzerinde iki adet anahtar oyuncu tanımlanmıştır. İlk oyuncuya negatif anahtar oyuncu denilmiştir. Bu oyuncuya örnek olarak hastalıkların yayılmaması için karantinaya alınan ya da suç işlenmesine engel olmak için tutuklanan kişiler (toplumdaki düğümler) örnek verilebilir. İkinci oyuncu ise pozitif anahtar oyuncu olarak adlandırılmıştır. Bu oyuncuya örnek olarak sağlık hizmetleri sunabilen ya da bir organizasyonda liderlik özelliği olan insanlar verilebilir. Anlaşılabacağı üzere ilk oyuncunun ağdaki bağlantılarının minimize edilmesi, ikinci oyuncunun bağlantılarının ise maksimize edilmesi amaçlanmaktadır.

Birbiriyle bağlantılı düğüm sayısını minimize etmeyi amaçlayan ve dinamik programlama kullanılan Addis ve diğerlerinin (2011) çalışması ve kritik düğüm probleminin çözümü için dal ve kesme yaklaşımını kullanan Di Summa ve diğerlerinin (2012) çalışması düğümlerin imhası üzerine kurgulanmış diğer ağ kesme çalışmaları olarak gösterilebilir.

(2) Arka Bağlı Ağ Kesme Problemleri

Bu bölümde ağlar üzerindeki iletişimi koparmak ya da zayıflatmak için düğümleri birbirine bağlayan arkları imha etmeyi temel alan bazı çalışmaların analizi yapılacaktır.

Aneja ve diğerlerinin (2001) çalışması tek ürün taşınan yönlü bir ağ üzerinde iki adet problemi içermektedir. İlk problemdeki amaç bir arkın kesilmesinden sonra geriye kalan ağ üzerindeki $b-h$ düğümleri arasındaki akışı maksimize edecek güzergâhın belirlenmesidir. Bir ark kesildiğinde ark üzerinden geçen akış bitirilmiş olmaktadır. Çözüm için "maksimum geri kalan akış problemi" adı verilen bir algoritma oluşturulmuştur. Algoritma ile polinom zamanda çözüm elde edilebilmektedir. İkinci probleme ise "en hayati ark problemi" adı verilmiştir. Bu problemdeki amaç kesilmesiyle ağdaki maksimum akışı minimize edecek arkı belirlemektir.

Du ve Chandrasekaran (2007) maksimum geri kalan akış probleminde bir yerine iki ark kesildiği durumda problemin NP-zor bir problem olduğunu göstermektedirler.

Ball ve diğerlerinin (1989) çalışmasında ele alınan ağlardaki arkların negatif olmayan uzunluk değerleri ve kesilme maliyetleri bulunmaktadır. Kesme işlemi için kullanılacak kaynak sınırlıdır. Çalışmada $b-h$ düğümleri arasındaki en kısa güzergâhın uzunluğunu maksimize etmek için kesilecek en hayati arkların kümesi belirlenmeye çalışılmaktadır. Ayrıca çalışmada problemin NP-zor olduğu gösterilmiştir.

Ford ve Fulkerson (1956), Wollmer (1964), McMasters ve Mustin (1970), Wood (1993), Israeli ve Wood (2002), Myung ve Kim (2004), Lim ve Smith (2007), Royset ve Wood (2007) çalışmalarını ağ üzerindeki arkların kesilmesi temelinde oluşturmuşlardır.

(3) Düğüm ve Arka Bağlı Ağ Kesme Problemleri

Bu bölümde ağ kesme işlemi için hem düğüm hem de ark kullanan çalışmaların analizi yapılacaktır.

Dinh ve diğerleri (2010) doğal afetler ya da kötü niyetli saldırılar gibi yıkıcı olaylar olması durumunda ağda oluşabilecek kırılganlığı ele alarak ağ planlaması ve risk yönetimi üzerinde çalışmışlardır. En yıkıcı senaryolara karşı bile güvenlik önlemi geliştirmeyi amaçlamaktadırlar. Güvenlik açığı ile ilgili yapılan mevcut çalışmaların çoğunda $G=(V,E)$ şeklinde tanımlanan çizgeler üzerindeki düğümler kümesi (V) ve bağlantıyı sağlayan arklar kümesi (E) üzerine odaklanıldığı belirtilmiştir. Fakat çizgelerin homojen olmayan yapıları nedeniyle bazı ark ve düğümlerin kesilmesinin diğerlerine göre iletişimi daha fazla azaltacağı dile getirilmiştir. Güvenlik açığı problemi için "ikili bağlantı" adı verilen bir ölçüm kullanılmış ve problemin formülü oluşturulmuştur.

Granata ve diğerleri (2013) ağı birbiriyle bağlantısı olmayan alt ağlara bölmeye çalışmışlardır. Çalışmanın amacı alt ağlardan en büyüğünün boyutunu yani sahip olduğu düğüm sayısını minimize etmektir. Ağı bölme işlemi için $b-h$ düğümleri arasındaki alt tur içermeyen kritik kesme güzergâhı belirlenmeye çalışılmaktadır. Bölme işlemi bu güzergâhtaki düğüm ve arkların imha edilmesi sonucunda gerçekleşmektedir. Problemin çözümü için gezgin satıcı problemini temel alan bir karışık tamsayılı doğrusal

programlama modeli oluşturulmuştur. Başka bir çözüm seçeneği olarak probleme uyarlanan dal fiyat algoritmasının modelden daha hızlı çözüm sağladığı görülmüştür.

b. Oyuncuların Amaçlarına Göre Ağ Kesme Problemleri

Ağ kullanıcıların (AK) ağ üzerindeki amaçları birbirlerinden farklılıklar gösterebilmek-tedir. Amaç bazen en kısa zamanda hedefe ulaşmak olurken bazen de hedefe olabilecek maksimum miktarda ürün taşımak ya da hedefe minimum maliyetle ulaşmak olabilmektedir. Hatta bazen bahsi geçen amaçlardan birden fazlası aynı anda gözetilebilmektedir. AK'ların amaçları bu kadar çeşitlilik gösterebilirken, onları durdurmaya çalışan yol kesicilerin (YK) tek bir stratejiye sahip olması düşünülemez. AK'nın amacını gerçekleştiremediği durumda mücadelenin kazananı YK olmaktadır. Dolayısıyla YK stratejisini AK'nın amacını baltalamak üzerine kurgulayacaktır. YK'nın amacı $b-h$ düğümleri arasındaki en kısa güzergâhın uzunluğunu maksimize etmek, ağda taşınabilecek maksimum veri/malzeme akışını minimize etmek ya da minimum kaynak ile ağın kullanımını maksimum miktarda engellemek olabilmektedir. Her amaç literatürde farklı çalışmalarda kendine yer bulmaktadır. Bu bölümde her amaç türü için yapılan bazı çalışmalardan bahsedilecektir.

(1) En Kısa Güzergâhın Kesilmesi

Bu bölümde AK'nın başlangıç düğümünden hedef düğümüne en kısa güzergâh üzerinden gitmeyi amaçladığı durum ele alınmaktadır. Güzergâhın uzunluğu arkların uzunlukları ya da katedilme süreleri kullanılarak bulunmaktadır. YK'nın amacı en kısa güzergâhı maksimize etmektir. Ahuja ve diğerlerinin (1993, s.94) çalışması herhangi iki düğüm arasındaki en kısa güzergâhı bulmak için bir doğrusal programlama formülasyonu içermektedir.

Israeli ve Wood (2002) $b-h$ düğümleri arasındaki en kısa güzergâh uzunluğunu maksimize etmek için kesilecek en uygun arkları belirlemeye çalışmaktadırlar. Bir arka saldırıyı arka tamamen kullanılamaz hale getirmek ya da arkin etkin uzunluğunu arttırmak olarak tanımlamışlardır. Çalışmada YK'nın kısıtlı kaynağa sahip olduğu kabul edilmektedir. Bu iki

aşamalı max-min (minimum uzunluğun maksimize edilmesi) tipindeki problemin çözümü için karışık tamsayılı programlama formülasyonu, Benders ayrıştırması ve küme örtme algoritmaları kullanmışlardır.

Yates ve Sanjeevi (2013) en kısa güzergâh problemini ülke güvenliği için kritik öneme sahip ağlara yapılan saldırılar içerisinde değerlendirmişlerdir. Probleme AK'nın gözünden bir çözüm aranmaktadır. Problem sabit Stackelberg oyunu olarak ele alınmış ve iki aşamalı karışık tamsayılı program olarak formüllendirilmiştir. Saldırgan AK'nın tedarik zinciri için kullandığı minimum uzunluktaki güzergâhı maksimize etmeye çalışmaktadır. Problemin amaç fonksiyonu max-min şeklindedir. Otoyollar üzerindeki bölümler arklar olarak kabul edilmiştir. Saldırgan $b-h$ düğümleri arasında farkedilememe olasılığının maksimum olduğu güzergâhları kullanmak istemektedir. Savunma amacıyla ağ üzerine bu maksimum değeri minimize edecek şekilde kısıtlı sayıdaki savunma sensörleri yerleştirilir. Ark üzerindeki etkileri birbirinden farklı olan statik ve dinamik sensörler bulunmaktadır. Ağ hakkındaki bilgileri toplamak, saklamak ve işlemek için bir coğrafi bilgi sistemi kullanılmıştır. Sensörlerin etkileri hakkındaki bilgiler ilişkisel veritabanlarında saklanmıştır. Problem formülasyonunun sonuçları Kaliforniya otoyolunda analiz edilerek sensör parametreleri, bütçe seviyeleri ve hedef kümeleri üzerinde denemeler yapılmıştır.

(2) Maksimum Akışın Kesilmesi

AK kapasiteli ağlarda ark kapasitelerini aşmadan $b-h$ düğümleri arasında veri/malzeme akışının maksimum olduğu güzergahı kullanmaya çalışırken, YK maksimum akışı minimize etmeye çalışır. Ahuja ve diğerlerinin (1993, s.167) çalışmasında da belirtildiği gibi maksimum akışı bulmak için genel olarak kullanılan iki çeşit algoritma vardır. İlk algoritma çeşidi $b-h$ düğümleri arasındaki güzergâhlar boyunca akışın aşamalı olarak arttırıldığı artan güzergâh algoritmalarıdır. İkinci algoritma çeşidi ise akışın tüm ağa yayıldığı ve kapasite fazlası akışın düğümlerden geriye yani başlangıç düğümüne ya da ileriye yani hedef düğümüne doğru gönderilerek rahat bir akışın olmasının sağlandığı önakış - itme algoritmasıdır.

Ghare ve diğerlerinin (1971) çalışmasında N adet savaş alanı üzerindeki düşman ile mücadele eden bir komutan bulunmaktadır. Çalışmada düşman birlikleri arasındaki lojistik sistem üzerinde komutanın nasıl bir kesme politikası izleyeceği belirlenmek istenilmektedir. Amaç birlikler arasında birim zamanda taşınan savaş malzemesini minimize etmektir. Komutanın elinde bulunan kaynak problemin kısıtlarından biridir. Problemin çözümü için dal sınır yöntemi kullanılmıştır.

Bu alanda öne çıkan çalışmalardan biri Wood (1993) tarafından gerçekleştirilmiştir. Çalışmada ABD ile Güney Amerika ülkeleri arasındaki kapasiteli ağ üzerindeki uyuşturucu trafiği ele alınmaktadır. Elde bulunan sınırlı kaynağın imkan verdiği ölçüde ağın arkaları kesilerek akış minimize edilmeye çalışılmaktadır. Problemin çözümü için bir tamsayılı doğrusal programlama modeli oluşturulmuştur. Bu model daha sonra yapılan birçok çalışmaya yön vermiştir. Örneğin Royset ve diğerlerinin (2007) iki amaçlı ağ kesme problemleri ile ilgili çalışmasında, Bingöl (2001) ve Uygun'un (2002) yüksek lisans tez çalışmalarında bu model kullanılmıştır. Altner ve diğerleri (2010) ise bu model üzerinde iyileştirme çalışmaları gerçekleştirmiştir.

Rocco ve diğerlerinin (2009a) çalışmasında ağdaki ark kapasitelerinin ve kesilme maliyetlerinin birbirinden farklı olabileceği göz önüne alınmıştır. Problemin çözüm temelini 3 adımda oluşturmuşlardır. Bunlar; i) Muhtemel ağ kesme stratejileri üretmek için Monte Carlo simülasyonu, ii) $b-h$ arasındaki maksimum akışı analiz etmek için Ford-Fulkerson'un maksimum akış yöntemi, iii) Bir arkın nihai ağ kesme stratejisinde devre dışı kalıp kalmayacağını olasılığını belirleyen optimizasyon tekniğinin tanımı.

Zenklusen (2010) maksimum akışı minimize etmek için kısıtlı kaynak ile ark ve düğüm kesilmesi konusunda çalışmıştır. Çalışmada ağdaki akışın kesilme probleminin kuvvetli NP-tam olarak tanımlandığından bahsedilmiştir. Problemin düzlemsel ağlarda çözümü için psödo polinom bir algoritma oluşturulmuştur. Algoritma birden fazla başlangıç düğümü ve birden fazla hedef düğümü için arz ve talebin eşit olduğu durumlarda çalışmaktadır. Yönteme eklenen bir uzantıyla algoritma güncellenmiştir. Yeni algoritmaya

göre arz ve talep ile ilgili kısıta gerek kalmadan tek başlangıç düğümü ve tek hedef düğümü ile de çözüme ulaşılabilir.

Wood (2011) çalışmasında iki aşamalı ağ kesme problemlerinin iki kişili sıfır toplamli sıralı bir oyun (Stackelberg) olduğunu belirtmektedir. İlk adımda YK sınırlı kaynak ile düşmanın ağındaki düğüm ve/veya arklara saldırarak maksimum akışı minimize etmeye çalışmaktadır. İkinci adımda düşman saldırının sebep olduğu zararı tespit ederek sonucu ve maliyeti optimize etmeye çalışır. Problemin çözümü için ikili değişkenler ve basit kısıtlar ile oluşturulan bir karışık tamsayılı program ve ayrıştırma algoritmaları kullanılmaktadır.

(3) Minimum Maliyetle Ağ Kesme

Bu bölümde incelenen çalışmalarda YK'nın minimum kaynak tüketerek AK'ya engel olmaya çalıştığı durum analiz edilmektedir. YK'nın amacına ulaşması için ağdaki tüm yolları imha etmesi gerekmeyebilir. Zaten bunun için elinde yeterli kaynak da olmayabilir. O nedenle minimum kaynak tüketimiyle AK'nın faaliyetlerini en çok kısıtlayabilecek arkların imha edilmesi hedeflenmelidir. Burada imha edilecek arkların doğru seçilmesi önem kazanmaktadır. Kesilecek arkların seçiminde arkların imha edilme maliyetlerinin farklı olabilmesi YK'nın kaynak tahsisinde önemli bir etken olmaktadır.

Golden'in (1978) çalışmasında ağdaki ark uzunlukları ve arkları kesmek için (uzunluklarını arttırmak için) doğrusal maliyet fonksiyonu verilmektedir. Çalışmada ağdaki $b-h$ düğümleri arasındaki en kısa güzergâhı en az T birim uzatacak en düşük maliyetli kesme stratejisinin belirlenmesi amaçlanmaktadır. Çalışma minimum maliyetli kesme problemidir. Çalışmada uygulamalar ve genelleştirmeler tartışılmış ve birden fazla hedef düğümü olması durumu da irdelenmiştir.

Wood'un (1993) kapasiteli ağ üzerinde düşmanın maksimize etmeye çalıştığı akışın, YK tarafından kısıtlı kaynak ile minimize edilmeye çalışıldığı durumu ele aldığından bahsedilmiştir. Çalışmada her arkın kesimi için bir birim kaynak harcandığında bile problemin NP-tam olduğunu vurgulamıştır.

(4) Çok Amaçlı Ağ Kesme

Rakip ile mücadele ederken şu ana kadar bahsi geçen amaçlardan iki ya da daha fazlası aynı anda göz önüne alınabilir. Mesela, karar verici bir taraftan elindeki kaynağı olabildiğince az kullanmak isterken, diğer taraftan da ağda devre dışı bırakacağı ark sayısını olabildiğince artırmak isteyebilir. Bu durumda kesilecek arklar aynı anda iki amaç gözetilerek seçilir. Birbiriyle çelişen bu iki amaç için bir denge noktası belirlenerek tatminkar bir sonuç elde edebilmesi her zaman kolay olmayabilir. En az kaynak kullanarak maksimum akışı minimize etmek ve bunun yanında en kısa güzergâhı maksimize etmek hedeflendiğinde ise kesilecek yollar aynı anda üç amaç gözetilerek seçilir. Ağ kesme problemlerinde yolların birden fazla amaca göre kesilmeleri gerçek yaşam problemleri ile daha çok örtüşmektedir.

Royset ve diğerlerinin (2007) çalışmasında maliyete ve ağdaki akışa göre iki amaçlı deterministik ağ kesme problemlerinin analizi yapılmıştır. Ağ kesme problemi belirlenen bir dizi amaca göre tek amaçlı olarak ayrı ayrı çözülmüştür. Belirlenen dal sınır algoritması ile çözülen tek amaçlı problemlerde Lagrangian Gevşetmesi (LG) kullanılmıştır. LG'de problemin çözümünü zorlaştıran kısıtlar Lagrangian çarpanı olarak adlandırılan ceza katsayılarıyla amaç fonksiyonuna eklenmektedir. Mesela maksimum akışa göre ağ kesme problemlerinde maliyet kısıtı I olarak adlandırılan ceza katsayısıyla amaç fonksiyonuna eklenmiştir. Sonrasında bu tek amaçlı kesilme problemleri üzerinde Pareto optimalite yaklaşımıyla bir strateji belirlenmeye çalışılmıştır.

Rocco ve diğerlerinin (2009b) çalışmasında birden fazla amacı olan ve bu amaçların tek amaçlı optimizasyon problemleri olarak ayrı ayrı çözümüne ihtiyaç duymayan “çok amaçlı deterministik ağ kesme problemleri” için pareto optimalite temelli bir algoritma oluşturulmuştur. Çalışma Pareto sınırlarını belirlemek için şu yaklaşımlardan faydalanmaktadır; i) Muhtemel ağ kesme stratejileri geliştirmek için Monte Carlo simülasyon teknikleri, ii) Stratejilerin $b-h$ düğümleri arasındaki maksimum akışlarını analiz etmek için çizge teorisi, iii) Bir bağlantının optimal Pareto kümesinde olma olasılığının tespiti için evrimsel bir arama algoritması. Bu evrimsel algoritma “olasılıklı çözümün

keşif algoritması” olarak adlandırılır. Çalışmada çok amaçlı problemlerde çözüm için belirlenen alternatif stratejilerin, amaç fonksiyonu değerlerine göre Pareto optimalitelerinin analiz edilmesi temeline çözüm aranmaktadır.

Literatürde kısıtlı kaynaklarla en kısa güzergâha göre ağ kesme problemleri için çözümler geliştirilmiştir. Rocco ve Ramirez-Marquez'in (2010a) çalışması en kısa güzergâh problemlerini içeren iki amaçlı problemler için ağ kesme stratejilerinin optimal Pareto kümesini kullanan yeni bir algoritma içermektedir. Çalışmada iki amaç optimize edilmeye çalışılmaktadır. İlk amaç en kısa güzergâh uzunluğunun maksimize edilmesidir. Ayrıca bu amaç en güvenli güzergâhın güvenilirliğinin minimizasyonu şeklinde değiştirilerek de çalışılmıştır. İkinci amaç ise ağ kesme stratejisinin maliyetini minimize etmektir. Bu çok amaçlı problemi çözmek için geliştirilen algorithmada ağ kesme stratejileri oluşturabilmek için Monte Carlo simülasyonu temel alınmıştır. Ayrıca en kısa ya da en güvenilir güzergâhı bulmak için kullanılacak stratejinin analizi için çizge teorisinden yararlanılmıştır. Algorithmada bir bağlantının optimal Pareto kümesinde olmasının olasılığı içinde aramalar yapılmaktadır.

Rocco ve diğerlerine (2010b) ait diğer bir çalışmada ise çok amaçlı deterministik ağ kesme problemlerini çözmek için olasılıklı çözümün keşif algoritması oluşturulmuştur. Algoritma yazarların daha önceki çalışmalarında olduğu gibi Monte Carlo simülasyonunu, çizge teorisini ve optimal Pareto kümesini kullanmaktadır. Çalışma iki ve üç amaçlı problemlerde amaçları ayrı ayrı çözmeden tek bir amaç gibi tek seferde çözecek bir yaklaşım ortaya koymaktadır.

Shen ve diğerleri (2012) çalışmalarını yönsüz çizgeler için gerçekleştirmişlerdir. Amaç çizge üzerindeki düğümler arasındaki bağlantısızlığı maksimize etmektir. Amacı gerçekleştirmek için kesilecek en uygun düğüm grubu belirlenmeye çalışılmaktadır. Kesme işlemleri için üç adet ölçüt belirlemişlerdir. Bu ölçütlerden ilki kesme işlemi sonucunda ana ağdan ayrılan bağlantılı bileşenlerin sayısını maksimize etmektir. Böylece her bağlantılı bileşendeki düğüm sayısı minimize edilmiş olacaktır. İkinci ölçüt ise olabilecek en büyük boyutlu bileşenin boyutunu minimize etmektir. Burada

boyut kavramı düğüme doğrudan bağlantılı olan ark sayısı olarak tanımlanmaktadır. Bu iki ölçüt için dinamik programlama temelli bir model oluşturarak çözüm aranmıştır. Üçüncü ölçüt ise kesilen bağlantıların yeniden oluşturulmasının minimum maliyetini maksimize etmektir. Tanımlanan ölçütlere bağlı bir sonuç elde etmek için karışık tamsayılı doğrusal programlama modeli oluşturulmuştur.

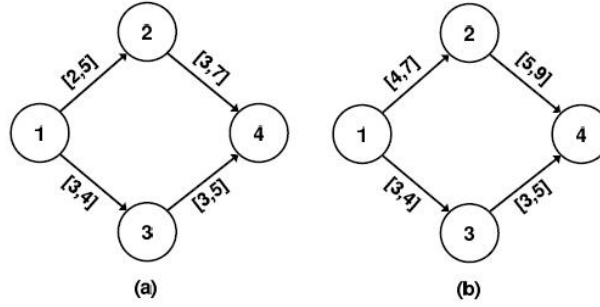
c. İstihbarata Dayalı Asimetrik Ağ Kesme

Asimetrik ağ doğru istihbarata sahip olunmadığı ya da bilginin karşı taraftan başarıyla gizlendiği durumlarda ortaya çıkar. Yanlış istihbari bilgi ağın olduğundan farklı algılanmasına yol açar. Böylece AK'nın seçeceği güzergah veya YK'nın kesmek için belirleyeceği yollar, olduğundan farklı algılanan bir ağa göre oluşur. İstihbari bilgilerin doğruluğu amaç fonksiyonunun sonucunu önemli ölçüde etkilemektedir.

Şu ana kadar anlatılanlar, ağ üzerinde mücadele eden tarafların, ağ hakkında doğru bilgilere sahip oldukları varsayımı üzerineydi. Arkların uzunluğunun ve kapasitesinin her iki karar verici tarafından kesilme öncesinde ve sonrasında doğru olarak bilindiği varsayılmaktaydı. Gerçekte ise oyuncular her zaman doğru bilgiye sahip olamayabilmektedir.

Bayrak ve Bailey (2008) oyuncuların yanlış istihbari bilgiler nedeniyle ağ hakkında gerçek dışı bilgilere sahip olduğu durumlardaki ağ kesme problemlerini analiz etmektedirler. Çalışmalarından alınan Şekil 4'teki ilk ağ doğru bilgiler içermektedir. AK elde ettiği yanlış istihbarat nedeniyle ağ ile ilgili bilgilerin ikinci ağdaki gibi olduğunu düşünmektedir. Parantez içindeki ilk sayılar arkın uzunluğunu, ikinci sayılar ise arkın kesilmesi durumunda yeni uzunluğunun ne kadar olacağını gösterebilir. Bu örnekte AK, (1,2) ve (2,4) arklarının gerçek uzunluğunu ve kesilme sonrası uzunluğunu olduğundan fazla bilmektedir. YK'nın bütçe kısıtı nedeniyle sadece bir arkı kesebildiği farzedilsin. AK, ikinci ağın üzerinde olduğunu düşündüğü için uzunluğu 6 olan 1-3-4 güzergahını seçecektir. YK ise en kısa güzergâh üzerinde en fazla uzamaya sebep olacak (2,4) arkını kesmek için üzerine sensör yerleştirecektir. Fakat AK ikinci ağa göre hareket ettiği için YK'nın uygun hamlesi (3,4) arkının kesilmesi olmalıydı. AK'nın kesilen arklardan haberdar

olduğu kabul edilmektedir. Problem iki aşamada ele alınmaktadır. İlk aşamada YK elindeki kaynaklara bağlı olarak arkları keserek en kısa güzergâhı maksimize etmeye çalışmaktadır. İkinci aşamada ise AK sahip olduğu istihbari bilgilere göre en kısa olan güzergâh üzerinden $b-h$ arasında yol almaktadır. Bayrak ve Bailey problemi doğrusal olmayan karışık tamsayılı programlama şeklinde formüle etmişlerdir. Ayrıca çözüm için dal sınır yöntemi kullanılmıştır.



Şekil 4. Asimetrik Bilgi Ağları. (a) Gerçek Ağ, (b) Ağ Kullanıcının Ağ Hakkında Sahip Olduğu Bilgiler

Asimetrik ağ kesme problemleri gerçek yaşamda sık karşılaşılabilecek bir durum olmasına rağmen bu konuda literatürde kapsamlı bir çalışma bulunamamıştır.

ç. Deterministik ve Stokastik Ağ Kesme Problemleri

Deterministik ağ kesme problemlerinde ağ ile ilgili bilgiler belirlidir ve ağ kesme problemi bu belirlilik içerisinde gerçekleşir. Genellikle maksimum akışın minimizasyonu ya da kaynakların kesilecek arklara optimum tahsisi çalışmalarında öne çıkar. Yukarıda bahsi geçen Wood'un (1993) çalışmasında ve Akgün'ün (2000) çalışmasında deterministik ağ kesme yapısı bulunmaktadır.

Arkları imha etme işlemlerinin başarısı belirsiz olabilir ve/veya iki düğüm arasındaki arkın kapasitesi bir alt sınır ve üst sınır arasında rastgele bir değer olabilir. Bu bölümde bu tür stokastik problemlerin analizi yapılacaktır.

Arkın kesildikten sonra kapasitesinin bir bölümünün kullanılmaya devam edilebildiği problemlere sürekli kesme denir. Kesme işleminden sonra arkın tamamen devre dışı kaldığı problemlere ise kesikli kesme denir.

Cormican ve diğerleri (1998) maksimum akışın kesilmesi problemlerinin stokastik uzantılarını çalışmışlardır. Problemlerinde kesilme başarıları iki değerli olarak ele alınmıştır. Yani başarılı bir kesilme işleminde ağın kapasitesi sabit bir oranda azalırken başarısızlık durumunda ağ kullanılmaya devam edilmektedir. Ark kapasitesinin net bilinemediği durumlarda olasılıklar verilmesi, belirlenen bir alt sınır ile üst sınır arasında olması gibi durumlar ele alınmıştır. Çalışmada problemler bir dizi algoritma ve stokastik programlama teknikleriyle çözülmüştür.

Pan ve diğerleri (2003) nükleer maddeleri algılayabilen detektörlerin ağ üzerine konuşlandırılacakları en uygun yerlerin belirlenmesi için stokastik ağ kesme modelleri tanımlamışlardır. Nükleer madde kaçakçıları ağ üzerinde yakalanmama olasılığının maksimum olduğu güzergâhlar üzerinden taşıma yapmaktadırlar. YK bu maksimum değeri minimize etmeye çalışmaktadır. Bu problem iki aşamalı stokastik karışık tamsayılı program olarak modellenmiştir. Detektörlerin yerleştirilme aşamasında kaçakçının başlangıç ve hedef düğümlerinin bilinmemesi nedeniyle program stokastiktir. Çalışmada problemin kuvvetli NP-zor olduğu ispat edilmiştir.

Janjarassuk ve Linderöth'un (2008) çalışması kapasiteli ağ üzerinde düşmanın maksimum akışını minimize etmek için ağın bir bölümünü kesme konusunu içermektedir. Ağda kesilecek kısmın miktarı eldeki kaynak kısıtı ile sınırlandırılmıştır. Burada ağ kesme problemlerinin stokastik versiyonu çalışılmaktadır. Ağdaki bir arkın başarılı bir şekilde kesilmesi Bernoulli rastgele değişkenidir. Dualite ve doğrusallaştırma teknikleri kullanılarak deterministik karışık tamsayılı program formüle edilmiştir. Formülasyonun yapısı çözüm için ayrıştırma tekniklerinin kullanımına müsade etmektedir.

Zheng ve Castanon'un (2012) çalışmasında ağ kesme probleminin iki kişi arasında oynanan bir oyun olduğu vurgulanmıştır. Çalışmada YK'nın hamlelerinin sonucu belirsizdir. Ele alınan stokastik ağ kesme problemlerinde oyuncuların oyun sırasında bilgi topladığı varsayılmaktadır. YK ağa bir ya da

iki aşamada saldırmaktadır ve YK bir önceki saldırıların bilgisini saklayabilmektedir. YK'nın tek aşamada saldırdığı problemler için gitgide artan hassas alt sınırların kullanıldığı dal sınır algoritması uygulanmıştır. Bu uygulamanın sonucu literatürdeki diğer yaklaşımlardan açıkça daha hızlı sonuç vermektedir. Tek aşamalı problem genişletilerek iki aşamalı problemlerde de kullanılır hale getirilmiştir.

Johnson ve diğerleri (2013) ağ üzerinde istenmeyen geçişlere engel olmak için seçilen yerlere sensör konulması konusunda bir çalışma gerçekleştirmişlerdir. Kaçakçılar başlangıç düğümünden hedef düğümüne gitmek için belirli bir güzergâh (deterministik) ya da Markov zinciriyle belirlenen bir güzergâh (stokastik) kullanabilmektedirler. Çalışmada iki amaç tanımlanmıştır. İlk amaç kaynağın kısıtlı olduğu durumda yakalanan kaçak sayısını maksimize etmektir. İkinci amaç ise tüm kaçakları olabilecek en ucuz maliyetle yakalamaktır. Problemin çözümüne yönelik bazı özel çizge ayarları kullanılmıştır.

d. Ağ Kesme Problemlerinin Oyun Teorisi ile İncelenmesi

Ağ kesme problemleri iki ya da daha fazla kişinin bir ağ üzerinde oynadığı oyundur. Her oyuncu rakibine karşı uyguladığı stratejilerle kârını maksimize etmeye çalışır. Aslında oyuncuların diğer oyuncuyu gözleyerek sıralı adımlar attığı ağ kesme problemleri bir çeşit Stackelberg oyunudur. Doğal olarak literatürdeki çalışmaların çoğu problemlerin bu yönünü vurgulamamaktadırlar. Bu bölümde bu vurgulamayı yapan bazı çalışmalardan söz edilecektir.

Washburn ve Wood (1995) ABD ile Latin Amerika ve Karayipler arasındaki uyuşturucu trafiğini önlemeye yönelik modeller önermişlerdir. Birden fazla başlangıç ve hedef düğümü olması, birden fazla AK ve YK olması ve ağın yönlü olup olmaması durumlarını da ayrıca göz önünde bulundurmışlardır. Çalışmada AK'nın gözetlenen bir arkı kullanması halinde yakalanması ve yakalanmadan geçmesi durumlarının olasılıklarını içeren bir model oluşturulmuştur. Model iki kişili sıfır toplamı bir oyun modelidir. Yani yakalanma olasılığı a ve yakalanmama olasılığı b ise $a + b = 1$ olmakta ve a ve b 'den herhangi birinde olan bir değer değişimi diğerinin değerine

eklenmekte ya da çıkarılmaktadır. Bir arkın kapasitesi ise algılanma olasılığı ile ters orantılı olarak ele alınmaktadır. Yani k arkından geçen kaçakçının algılanma olasılığı p_k ise, arkın kapasitesi $u=1/p_k$ olmaktadır. YK kestiği ark üzerinde düşmanı algılama olasılığını maksimize etmeye amaçlamaktadır. AK ise başlangıç ve hedef düğümünleri arasındaki yolculuğu için seçeceği güzergâhta kesilen bir ark olma olasılığını minimize etmeye çalışır.

Brown ve diğerleri (2006) çalışmalarında ABD için kritik öneme sahip sistemlerin terör saldırıları karşısında daha esnek bir yapıda olabilmeleri için iki seviyeli ve üç seviyeli optimizasyon modelleri oluşturmuşlardır. Ele alınan ilk model İki seviyeli terörist - ağ kullanıcı modelidir. Bu modelde terörist ağa saldırarak ağ kullanıcının minimize etmeye çalıştığı işletme maliyetlerini maksimize etmeyi amaçlamaktadır. Burada amaç fonksiyonu max-min yapısındadır. Diğer bir iki seviyeli model ise ağ kullanıcı - terörist modelidir. Bu modelde ise ağ kullanıcı teröristin maksimize etmeye çalıştığı tespit edilmeme olasılığını minimize etmeyi amaçlamaktadır. Burada amaç fonksiyonu min-max yapısındadır. Oluşturulan diğer bir model ise çözümü zor olan üç seviyeli ağ kullanıcı - terörist - ağ kullanıcı modelidir. Bu modelde terörist ağa saldırarak ağ kullanıcının minimize etmeye çalıştığı işletme giderlerini maksimize etmeye çalışır. Ağ kullanıcı ise ağın savunmasını iyileştirerek teröristin amacını minimize etmeye çalışır. Modelin amaç fonksiyonu min-max-min şeklindedir. Ele alınan problemin Stackelberg oyunu olduğu çalışmada vurgulanarak anlatılmıştır.

e. Tek Ürünlü, Çok Ürünlü Ağ Kesme Modelleri

Şu ana kadar bahsi geçen literatür çalışmalarında ağ üzerinde aksi belirtilmediği sürece tek çeşit ürün taşınmaktadır. Bu bölümde ise ağ üzerinde birden fazla ürün çeşidinin taşındığı problemleri ele alan çalışmalardan bahsedilecektir.

Akgün (2000) yüksek lisans tez çalışmasında AK'nın kapasiteli ve yönsüz $G=(N,A)$ ağı içerisinde K tane birbirinden ayrı önceden tanımlanmış $N'_1, \dots, N'_K$ düğüm grupları arasında maksimum akışla taşıma yaptığını varsaymaktadır. Bu çok ürünlü taşıma işlemi ortak kapasite kısıtıyla bağlanmış K adet tek ürünlü taşıma işlemi olarak modellenmiştir. (Bu aslında

çok ürünli maksimum akış modelidir.) Her düğüm grubunda ayrı bir kaynak ve depo bulunmaktadır. AK'nın amacı doğru düğüm grupları arasında, doğru miktarda taşıma yaparak maksimum akışı sağlamaktır. Taşıma için gruplara atanmış düğümlerin aynı zamanda hem başlangıç ve hem de hedef olabilecekleri varsayılmaktadır.

Churlzu ve Smith'in (2007) ele aldığı çok ürünli ağ kesme probleminde her arkın bir kesilme maliyeti bulunmaktadır ve YK kesme işlemi için kısıtlı kaynağa sahiptir. K çeşit ürün farklı başlangıç düğümlerinden farklı hedef düğümlerine taşınmaktadır. YK'nın amacı AK'nın çok ürünli taşıma sayesinde elde edeceği maksimum kârı minimize edecek kesme işlemini gerçekleştirmektir. Çalışmada ağ kesme işleminin kesikli ve sürekli durumları olduğu kabul edilmektedir. Kesikli durumda her ark ya yalnız bırakılır ya da tamamen imha edilir. Sürekli durumda ise kesme işleminin ardından arkın kapasitesi kısmen azalmış olabilir. Kesikli durumun çözümü için iki adet karışık tamsayılı model tasarlanmıştır. Sürekli kesme durumu için ise bir bölümleme algoritması oluşturulmuştur.

Garg ve Smith (2008) çok ürünli taşıma yapılacak ağların tasarımı üzerine çalışmışlardır. Çalışmada iki düğüm arasındaki akışın düğümler arasındaki bağlantının kapasite kısıtlarına bağlı olduğu ifade edilmektedir. Ağ üzerinde kullanılamayan düğüm ya da arklar olduğu durumlarda bile çok ürünli taşımanın devam ettirilebilmesi için tasarım aşamasında yeterli kapasite tanımlanması ve ağ üzerinde alternatif birçok güzergâhın oluşturulması gerektiği vurgulanmıştır. Ancak her bir arkı oluşturmanın bir maliyeti olduğu belirtilmiştir. Ağ üzerinde birçok kesilme senaryosu tanımlanmış ve her senaryoda eş zamanlı olarak birden fazla arkın kesildiği durumlar göz önüne alınmıştır. Bu senaryoları da dikkate alarak minimum maliyet ile çok ürünli taşıma yapılabilen ağların optimal tasarımını elde etmek için bazı yöntemler denenmiştir. Problemin çözümü için karışık tamsayılı programlama formülasyonu ve Benders ayrıştırma algoritması kullanılmıştır. Küçük problemlerde (örneğin sekiz düğümlü, 10 adet başlangıç ve hedef düğümü çifti olan ve 10 adet kesilme senaryosu içeren) Benders uygulaması karmaşık tamsayılı programlama formülasyonunun çözüm

süresinin %10'u kadar bir sürede çözüme ulaşabilmektedir. Orta ve büyük boyutlu problemlerde iki saat içerisinde çözüme ulaşılamadığı görülmüştür.

Akgün ve diğerleri (2011) çok terminalli yönsüz ağlarda maksimum akışın kesilmesi konusunda çalışmışlardır. Çalışmada AK ağ üzerindeki uç ya da daha büyük düğüm grubu arasındaki akışı maksimize etmeye çalışmaktadır. AK'nın başlangıç ve hedef düğümleri YK tarafından kesin olarak bilinemeyebilmektedir. Ancak YK başlangıç ve hedef düğümlerini içerebilecek K adet düğüm grubu belirleyebilecek bilgiye sahiptir. YK'nın amacı gerçek ya da olası başlangıç ve hedef düğümleri arasında AK'nın maksimize etmeye çalıştığı akışı minimize etmek için kaynak kısıtına bağlı olarak kesebileceği en uygun ark kümesini belirlemektir. Problemin iki aşamalı min-max tipinde bir problem olarak modellenebileceği belirtilmiş ve modelin kritik altyapıların savunması için analiz edilebileceğinden bahsedilmiştir.

2. LİTERATÜR ÇALIŞMALARININ PROBLEM TANIMLAMA ÇİZELGESİ İLE AÇIKLANMASI

Bu bölüme kadar bahsedilen literatür çalışmaları ayrıntılı olarak anlatılmış olsa bile çalışmaların içeriklerini kıyaslamak zahmetli bir iş olmaktadır. "Problem Tanımlama" bölümünde ağ kesme problemlerinin anlatımının daha kolay olabilmesi için Tablo 1'de görülen "Problem Tanımlama Çizelgesi" oluşturulmuştur. Bu tanımlama çizelgesi ile bu bölümde anlatılan belli başlı çalışmalar yeniden verilerek literatürde ağ kesme problemlerinin neler içerdiği daha net gösterilmeye çalışılacaktır. Bu amaçla Tablo 2 oluşturulmuştur.

Tablo 1'deki çizelge bu tez çalışmasının içeriğini göstermek için daha önce kullanılmıştı. Tablo 2'de bu tez çalışması için oluşturulan çizelge sonucu tekrar verilerek literatürdeki diğer çalışmalarla bu çalışmanın kıyaslanması amaçlanmaktadır. Bir sonraki bölümde ise bu çalışmanın literatüre olan katkısı ayrıntılarıyla anlatılacaktır.

Tablo 2. Literatür Çalışmalarının Problem Tanımlama Çizelgesi Kullanılarak Gösterilmesi

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

SYK

Y8

ÇYK

K(-)

B-Ç

B-Ç

B-Ç

B

SRZ

B

+

ÇAK

ÇÜ

+

B

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

SYK

Y8

ÇYK

B

SRZ

B

+

+

B

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

SYK

Y9

ÇYK

K(+)

B

SRZ

B

+

TÜ

+

B

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

SYK

Y2

ÇYK

K(+)

B

SRZ

B

+

TÜ

+

B

B

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

SYK

Y3

ÇYK

K(-)

O - Ç

O - Ç

O - BY

O

SRZ

O

+

A5

ÇAK

ÇÜ

+

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

Y10

ÇYK

O

O

+

+

Tablo 2'nin devamı

- Summa ve diğerleri (2011)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	
	Y10	ÇYK						O		B	+								+	B	B	B

- Borgatti ve diğerleri (2006)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	
	Y11	TYK									+	A7	ÇAK						+			

- Aneja ve diğerleri (2001)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	
	Y3	TYK									+	A2		TÜ					+	B		

- Du ve Chandrasekaran (2007)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	
	Y9	TYK									+	A2		TÜ					+	B		

- Ball ve diğerlerinin (1989)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	
	Y4	TYK						B	SRZ	B	+	A2		TÜ					+	B		B

- Dinh ve diğerleri (2010)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	
	Y3							BY			+	A2	ÇAK	TÜ					+	B		B

Tablo 2'nin devamı

- Israeli ve Wood (2002)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	
	Y3		K(+)	B - T	B - T			B		B	+		TÜ						+	B		

- Yates ve Sanjeevi (2013)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	
SYK	Y4+Y12	ÇYK		B - T	B - T	B				B	+	A1			BY		T(-)	S(-)	B	+	B	

- Ghare ve diğerleri (1971)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	
	Y3		K(+)	B - Ç	B - Ç	B	B - Ç			B	+		ÇAK	ÇÜ					+	B		

- Wood (1993)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	
	Y3	ÇYK	K(+)	B - T	B - T	B	BY-BY	B	SRZ	B	+	A2	TAK	TÜ			T(-)	S(-)	B	+	B	

- Rocco ve diğerleri (2009a)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	
	Y3	TYK	K(+)	B - T	B - T			O	SRZ	B	+	A2							+	B		B

- Zenklusen (2010)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	
	Y3	TYK	K(+)	B - Ç	B - Ç			B	SRZ	B	+								+	B		B

Tablo 2'nin devamı

- Wood (2011)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	
	Y3	TYK	K(+)	B - Ç	B - Ç	B		B	SRZ	B	+	A2			B		Tk		B	+	B	B

- Golden (1978)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	
	Y2+Y4	TYK	K(+)	B - T	B - Ç			B	SRZ	B	+								+		B	B

- Royset ve diğerleri (2007)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	
SYK	Y2+Y3	TYK	K(+)	B - T	B - T			B	SRZ	B	+								+	B		B

- Rocco ve diğerleri (2009b)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	
SYK	Y2+Y3	TYK	K(+)	B - T	B - T			B	SRZ	B	+								+	B		B

- Rocco ve Ramirez-Marquez (2010a)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	
SYK	(Y2,Y13)+Y4	TYK	K(+)	B - T	B - T			B	SRZ	B	+								+		B	B

- Rocco ve Ramirez-Marquez (2010b)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	
SYK	Y2+Y3+Y14	TYK	K(+)	B - T	B - T			B	SRZ	B	+					Tz			+	B		B

Tablo 2'nin devamı

- Shen ve diğerleri (2012)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
SYK	Y10+Y15+Y16	TYK	K(+)					B	SRZ	B	+					Tk			+		

- Bayrak ve Bailey (2008)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
SYK	Y4	TYK	K(+)	B - T	B - T	B		B	SRZ	B	+	A1	TAK	TÜ			Tk	S(-)	YB	+	B

- Cormican ve diğerleri (1998)

4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	
K(+)	B - T	B - T	B		O	SRZ	B	+								+	O		

- Pan ve diğerleri (2003)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
SYK	Y13	ÇYK	K(+)	BY - BY	BY - BY	B		O	SRZ	O	+	A6				T(-)	S(-)		+		

- Janjarassuk ve Linderöth (2008)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
SYK	Y3		K(+)	B - T	B - T	B		O	SRZ	B	+					T(-)	S(-)		+	B	

- Johnson ve diğerleri (2013)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
SYK	Y2+Y17	ÇYK	K(+)	B - T	B - T	B	BY - BY	B	SRZ	B	+	A9				T(-)	S(-)	B	+		B

Tablo 2'nin devamı

- Washburn ve Wood (1995)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
	Y18	ÇYK		B - Ç	B - Ç			O	SRZ	B	A6	ÇAK							O		

- Churlzu ve Smith (2007)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
SYK	Y19	TYK	K(+)	B - Ç	B - Ç	B		O	SRZ	B	A10	TAK	ÇÜ			T(-)	S(-)	B	B		B

- Akgün ve diğerleri (2011)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
SYK	Y3		K(+)	O - Ç	O - Ç			B	SRZ	B	A2		ÇÜ			T(-)	S(-)		B		B

- Bu tez çalışması

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
HYK	Y1	TYK	K(-)	B - T	B - T	B		B	SRL	B	A1	TAK	TÜ			T(-)	S(-)	B		B	

3. BU TEZ ÇALIŞMASININ LİTERATÜRE KATKILARI

Literatürdeki çalışmalarda ağ üzerinde mücadele eden oyunculardan ağ kullanıcı hareketli, yol kesici ise sabit olarak tanımlanmıştır. Çünkü yol kesici olarak mobese kameraları, mayınlar, radar sistemleri ya da sınırlardaki gümrük noktaları gibi sabit yapılar düşünülmüştür. Bu çalışmada ise yol kesici de ağ kullanıcı gibi ağ üzerinde hareket halindedir. Bu duruma gerçek hayat problemlerinde de sıklıkla rastlanmaktadır. Örneğin askeri operasyonlardaki askerler, orman yangınına müdahale eden itfaiyeciler, sel felaketinde suyun akışını değiştirmeye çalışan insanlar yol kesicidir ve hareket halindedirler.

Literatürdeki sabit yol kesicinin zamana karşı mücadelesi söz konusu değildir. Çünkü yol kesici bir noktaya konuşturıldığı anda kesme işlemine başlar ve ağ üzerindeki başka bölgelerdeki kesilmelerle ilgilenmez. Ancak bu çalışmadaki HYK bir noktayı kestikten sonra diğer noktaları kesmek için ağ üzerinde ilerler. Çünkü amacı AK'nın kullanabileceği tüm güzergâhları en kısa sürede kesmektir. O nedenle zamana karşı bir yarış içerisindedir. Yol kesicinin böylesine bir yarış içerisinde olduğu bir çalışmaya mevcut literatürde rastlanmamıştır.

Literatürdeki çalışmalarda kesilmesi gereken noktaların bir öncelik sırasının olması yaygın bir durum değildir. Ancak bu çalışmada kullanılan dal sınır algoritması ile HYK ağ üzerindeki kesme işlemlerini belirlenmiş bir sıraya göre gerçekleştirir. Algoritmaya göre öncelikle *b-h* düğümleri arasındaki en kısa uzunluktaki AK güzergâhı kesilmelidir. Her güzergâh kesildiğinde bir sonraki en kısa güzergâha yönelinmektedir. Sonuç olarak HYK'nın her adımı AK'nın en kısa güzergâhının uzunluğunu arttırmaya yöneliktir.

Bu çalışmadaki ağın ele alınış şekli de literatürde bir ilktir. Bu çalışmaya göre AK ve HYK'nın aynı anda kullanamayacağı arklar ve düğümler bulunmaktadır. Bir düğüm ya da ark sadece oyunculardan biri tarafından kullanılabilir. Bu paylaşılmış yapı nedeniyle ağ içiçe girmiş iki ağ gibi gözükmektedir. Literatürde ağın böyle bir analizine rastlanmamıştır.

ÜÇÜNCÜ BÖLÜM

MATEMATİKSEL MODELLER

1. PROBLEMİN ÇÖZÜMÜ İÇİN OLUŞTURULAN MATEMATİKSEL MODELLER

Bu bölümde problemin çözümü için oluşturulan iki adet model anlatılmaktadır. Her iki modelde de ağ kullanıcının (AK) her arkı üzerinde bir hareketli yol kesici (HYK) düğümü olduğu varsayılmaktadır. Modeller HYK'nın başlangıçta üzerinde bulunduğu düğümden yola çıkarak AK'nın başlangıç (b) ve hedef (h) düğümleri arasındaki tüm güzergâhları kesen minimum uzunluktaki HYK güzergâhını belirlemektedir.

Oluşturulan modeller HYK ağının tüm çizgesini kullanmaktadır ve HYK'nın takip ettiği güzergâhta alt turlar oluşmasına engel olarak güzergâhta sürekliliği sağlamaktadır. Alt tur oluşumunu engelleyen kısıt kullanılmadan oluşturulan ilk modeller GAMS ile kodlandığında şu şekilde güzergâhların oluşabildiği gözlemlenmiştir: ($A-B, C-D-E-C$). Dikkat edileceği üzere B ve C düğümleri arasındaki ark kullanılmamıştır. $C-D$, $D-E$ ve $E-C$ düğümleri arasındaki arklar $B-C$ düğümleri arasındaki arkten daha kısadır. Model B düğümünü son düğüm gibi kabul etmiş ve kısıtların zorlamasıyla HYK'yı E düğümünden daha önce kesmiş olduğu C düğümüne döndürmüştür. Böylece HYK güzergâhı birbirine bağlı olmayan iki ayrı güzergâhtan oluşmuştur. Dolayısıyla modellerin doğru sonuçlar verebilmesi için alt tur oluşumuna engel olmak birinci derecede önem arz eden durumlardan biri haline gelmiştir.

Oluşturulan ilk modelde HYK ağının tüm çizgesi üzerindeki her düğüme en fazla bir giriş ve bir çıkış olacağı düşüncesi temel alınmıştır. Dolayısıyla çözüm güzergâhında katedilen HYK arkları sayısı HYK düğümleri sayısının bir eksiği olacaktır. Bu mantık çerçevesinde oluşturulan ilk modelin formülasyonu ve ayrıntıları aşağıda anlatılmaktadır.

Kümeler:

$G = (V, Y)$: AK ağı. V düğümleri, Y arkları göstermektedir.

$H = (\hat{A} \cup \{A\}, E)$: HYK ağı. $\hat{A} \cup \{A\}$ ile ağın düğümleri gösterilmektedir. A ile gösterilen düğüm HYK'nın merkez düğümüdür. $\hat{A}$ ise geri kalan HYK düğümleri olup $\hat{A} \cong Y'$ 'dir. E ise HYK'nın arklarını göstermektedir.

$K = (\hat{A} \cup \{A\}, J)$: HYK'nın tüm çizgesi. $\hat{A} \cup \{A\}$ ağın düğümlerini, J ise ağın arklarını göstermektedir.

GK : G ağındaki $b-h$ düğümleri arasındaki tüm güzergâhların kümesi.

$d\{i\}$: K ağı üzerindeki i . düğüme dokunan arklar kümesi.

$P(g)$: AK'nın $b-h$ düğümleri arasındaki herhangi bir g güzergâhı üzerinde bulunan HYK düğümlerinin kümesi.

ST : HYK ağındaki alt turların kümesi.

Parametreler:

d_e : H ağındaki i ve j düğümleri arasındaki en kısa güzergâhın uzunluğu, $e \in J$.

Karar Değişkenleri:

$$x_e = \begin{cases} 1, & K \text{ ağındaki } e \text{ arkı kullanılırsa} \\ 0, & \text{aksi halde} \end{cases}$$

$$y_i = \begin{cases} 1, & \text{Eğer HYK } i \text{ düğümünü çözümde kullanırsa} \\ 0, & \text{aksi halde} \end{cases}$$

Formülasyon:

$$\text{Min} \sum_{e \in J} d_e x_e \quad (1)$$

Öyle ki;

$$\sum_{i \in P(g)} y_i \geq 1 \quad \forall g \in GK \quad (2)$$

$$y_A = 1; \quad \sum_{e \in d(A)} x_e = 1 \quad (3)$$

$$x_e \leq y_i; \quad x_e \leq y_j \quad \forall e = \{i, j\} \in J \quad (4)$$

$$\sum_{e \in d(i)} x_e \leq 2y_i \quad \forall i \in A \quad (5)$$

$$\sum_{e \in J} x_e \geq \sum_{i \in A} y_i - 1 \quad (6)$$

$$\sum_{\substack{e=(i,j) \\ i \in ST, j \notin ST}} x_e \geq 1 \quad (7)$$

$$\begin{aligned} x_e &= 0 \text{ veya } 1 \quad \forall e \in J \\ y_i &= 0 \text{ veya } 1 \quad \forall i \in A \cup \{A\} \end{aligned} \quad (8)$$

Amaç fonksiyonu (1) AK'nın $b-h$ düğümleri arasında kullanabileceği tüm güzergâhları kesen HYK'nın kendi ağı üzerinde katettiği en kısa güzergâhın uzunluğunu vermektedir.

(2) numaralı kısıt ise AK ağındaki (G) $b-h$ düğümleri arasındaki tüm güzergâhların üzerindeki en az bir HYK düğümünün çözüm güzergâhında olmasını garanti altına almaktadır.

(3) numaralı kısıtlar kümesi HYK'nın tüm çizgesinde (K) başlangıç düğümü olan A düğümünden bir kere hareket edileceğini ve HYK güzergâhının bu düğümde sonlanamayacağını ifade etmektedir.

(4) numaralı kısıtlar kümesi HYK'nın K ağı üzerindeki i ve j düğümleri arasındaki güzergâhı katetmesi durumunda bu düğümlerin çözüm güzergâhına eklenmesi gerektiğini ifade etmektedir. Ancak seçilen herhangi iki k ve l düğümü arasında mutlaka bir güzergâh olmasına zorlamamaktadır.

Ayrıca (5) gibi, (4) nolu kısıt kümesi bir düğümün seçilmemesi durumunda o düğümde başlayan veya sonlanan bir arkın HYK güzergâhında olamayacağını ifade etmektedir.

(5) numaralı kısıt ise HYK'nın K ağında A düğümü dışındaki düğümünlerinden birinin seçilen güzergâhın son düğümü olduğu durumda HYK'nın bir patika (en fazla tek giriş ve tek çıkış olmak üzere) güzergâhı halinde yürümesi için yazılmıştır.

(6) numaralı kısıt yol kesicinin güzergâhının ardışık arklardan oluşmasını sağlamaktadır.

(7) numaralı kısıt HYK'nın takip ettiği güzergâhta alt turlar oluşmasını engellemektedir. Bu amaçla alt turların kümesini içeren ST parametresi kullanılmaktadır. ST 'nin içerebileceği iki düğümden oluşan alt turlar $(A-B-A)$, $(B-C-B)$ şeklinde tanımlanırken, üç düğümden oluşan alt turlar $(A-B-C-A)$, $(B-C-D-B)$ şeklinde tanımlanmaktadır. Tek düğümlü alt turların oluşturulmasına gerek görülmemiştir. Çünkü her düğümün kendine dönüş mesafesi sonsuz kabul edilmiştir. Böylece n düğümlü bir HYK ağının tüm çizgesi için ST kümesi $\sum_{i=2}^{n-1} \frac{n!}{(n-i)!}$ adet elemana sahip olacaktır. Alt tur engelleme kısıtları

HYK ağının tüm çizgesine göre oluşturulduğu için HYK'nın güzergâhında geri dönüşler olmasına engel olmazlar.

(8) numaralı kısıt ise tamsayı kısıtlarıdır.

Bu formülasyonda eğer HYK ağının düğümlerinin sayısına $n = |A| + 1$ dersek, alt tur engelleme kısıtı göz ardı edildiğinde $m + n$ adet karar değişkeni ve $2 + 2 \times m + n + |GK|$ kadar kısıt yer almaktadır ($m = |J|$).

Bu model ile optimal çözüm bulunabilmesine rağmen çözüm için harcanan sürenin uzun olduğu görülmüştür. Alt tur oluşumuna engel olma, HYK'nın tüm çizgesini oluşturma ve $b-h$ düğümleri arasındaki her güzergâh üzerindeki HYK düğümlerinin kümelerini belirleme aşamaları çözüm süresinin uzun olmasına sebep olmaktadır.

Problemin çözümü için oluşturulan ikinci modelde de ilk modelde olduğu gibi HYK'nın üzerinden geçtiği her AK arkını kestiği varsayılmaktadır. Bu modelde her HYK düğümüne giriş ve çıkış sayılarının eşit olacağı düşünülmüştür. Çözüm güzergâhındaki son düğüm için bu kural geçerli değildir. Son düğüm ise yalnızca bir düğüm olabilecektir. Bu düğüme sadece bir giriş olacak ve çıkış olmayacaktır. Bu mantık çerçevesinde oluşturulan ikinci modelin formülasyonu ve ayrıntıları aşağıda anlatılmaktadır. İlk modelde tanımlanan küme ve parametrelerin bir kısmı bu modelde de kullanılmaktadır. Bu küme ve parametrelerin tanımları bu modelde yeniden verilmeyecektir.

Kümeler:

$b(j)$: HYK'nın j . düğümünde sonlanan arkların diğer uçlarındaki HYK düğümlerinin kümesi.

Parametreler:

d_{ij} : H ağındaki i ve j düğümleri arasındaki en kısa güzergâhın uzunluğudur. $(i, j) \in J$ 'dir.

Karar Değişkenleri:

$$x_{ij} = \begin{cases} 1, & \text{Eğer HYK'nın } i \text{ ve } j \text{ düğümleri arasındaki ark kullanılırsa} \\ 0, & \text{aksi halde} \end{cases}$$

$$s_j = \begin{cases} 1, & \text{HYK'nın } j \text{ düğümü güzergâh üzerindeki son düğümse} \\ 0, & \text{aksi halde} \end{cases}$$

Formülasyon:

$$\text{Min } \sum_{i \in \hat{A} \cup \{A\}} \sum_{j \in \hat{A}} d_{ij} x_{ij} \quad (9)$$

Öyle ki;

$$\sum_{j \in \hat{A}} x_{Aj} = 1 \quad (10)$$

$$x_{ij} + x_{ji} \leq 1 \quad \forall \{i, j\} \in \hat{A} \cup \{A\} \quad (11)$$

$$\sum_{j \in P(g)} \sum_{i \in b(j)} x_{ij} \geq 1 \quad \forall g \in GK \quad (12)$$

$$s_j = \sum_{i \in \hat{A} \cup \{A\}} x_{ij} - \sum_{i \in \hat{A}} x_{ji} \quad \forall j \in \hat{A} \quad (13)$$

$$\sum_{j \in \hat{A}} s_j = 1 \quad (14)$$

$$\sum_{i \in ST, j \notin ST} x_{ij} \geq 1 \quad (15)$$

$$\begin{aligned} x_{ij} &= 0 \text{ veya } 1 & \forall i \in \hat{A} \cup \{A\}, \forall j \in \hat{A}, i \neq j \\ s_j &= 0 \text{ veya } 1 & \forall j \in \hat{A} \end{aligned} \quad (16)$$

Amaç fonksiyonunda (9) AK'nın $b-h$ düğümleri arasındaki tüm güzergâhlarını kesen HYK'nın kendi ağı üzerinde katettiği güzergâhın uzunluğu minimize edilmeye çalışılmaktadır.

HYK başlangıçta merkez düğümünde yani A düğümündedir. (10) numaralı kısıt HYK'nın K ağındaki A düğümüne bağlı olan arklardan sadece birini katedeceğini göstermektedir. Yani HYK başlangıç düğümünden bir defa yola çıkmak zorundadır.

(11) numaralı kısıt HYK'nın tüm çizgesindeki bir arkın katedildikten sonra ters yönde tekrar katedilemeyeceğini göstermektedir. Yani eğer HYK güzergâhında i düğümünden j düğümüne gidilen ark kullanıldıysa, artık güzergâh j düğümünden i düğümüne gidilen arkı içermeyecektir.

(12) numaralı kısıt $b-h$ düğümleri arasındaki her AK güzergâhının en az bir kez kesileceği güvence altına alır.

(13) ve (14) numaralı kısıtlar HYK ağında belirlenen sonuç güzergâhının A düğümü dışındaki düğümlerden sadece birinde sonlanmasını, kalan düğümlerin ya seçilen güzergâhın dışında kalmasını ya da seçilen güzergâhta ara düğüm olmasını sağlar.

Bir önceki modelde ayrıntılı olarak anlatılan (15) numaralı kısıt HYK'nın takip ettiği güzergâhta alt turlar oluşmasını engellemektedir. Bu amaçla alt turların kümesini içeren ST parametresi kullanılmaktadır.

(16) numaralı kısıt karar değişkenlerinin tamsayı olma kısıtlarıdır. Modeldeki her HYK düğümüne giriş ve çıkış sayısı arasındaki fark sadece 0 ya da 1 olabileceği için s_j ikili değişken olarak tanımlanmıştır.

HYK güzergâhının alt turlar içerecek şekilde parçalı yapıda olmasına engel olmak, modellerin oluşturulma aşamasında en çok çaba harcanan konu olmuştur. İkinci model için eğer HYK'nın tüm çizgesinin (K) düğümlerinin sayısına $n = |A| + 1$ dersek, alt tur engelleme kısıtı göz ardı edildiğinde $m + n$ adet karar değişkeni ve $1 + m + n + |GK|$ kadar kısıt yer almaktadır ($m = |J|$).

İkinci model C ve GAMS yazılımları kullanılarak kodlanmıştır. Kodlama sonuçları Bölüm 5'te verilecektir.

DÖRDÜNCÜ BÖLÜM

DAL SINIR ALGORİTMASI

1. DAL SINIR ALGORİTMASI İLE PROBLEMİN ÇÖZÜMÜ

Bu bölümde problemin çözümü için dal sınır yöntemi kullanılacaktır. Kullanılacak olan dal sınır yöntemi derinlik öncelikli arama (DÖA) yapacaktır. DÖA bir kök değerden başlayarak dallanan ağaç yapısı içerisinde istenilen bir değeri aramada kullanılır. DÖA'nın özelliği öncelikle ağacın dallanmasının derinliğini tamamlamasıdır.

a. Dal Sınır Algoritmasında Üst Sınır Değeri

Dal sınır algoritmasının daha çabuk sonuca ulaşması için optimal sonuç veremeyecek dalların budanması gerekir. Budanacak dalların belirlenmesi için algoritmada üst sınır (ÜS) değeri kullanılır. Optimal çözüm değerinden daha küçük seçilen bir ÜS değeri çözüme ulaşılmasına engel olacaktır. Çok büyük seçilen bir ÜS değeri ise algoritmada fazla iterasyon oluşmasına sebep olabilir. Çalışmanın ileri aşamalarında uygun bir ÜS değerini bulmak için oluşturulan yöntemlerden bahsedilecektir.

b. Düğümün Alt Sınır Değeri

Problemin tanımında AK'nın EKG'yi kullandığı varsayılmıştı. EKG'nin kullanılamaması için üzerindeki arklardan en az birinin kesilmesi gerekir. Dal sınır yönteminin dallanan ağaç yapısı üzerinde HYK'nın EKG üzerindeki her bir düğüme ulaşmak için başlangıç düğümünden itibaren katettiği yolun uzunluğu her bir düğümün alt sınır (AS) değeri olarak kabul edilecektir. AS değeri ÜS değerinden büyük olan düğümün olduğu dal budanacaktır. Budanan dalda optimal güzergâh arama işlemi sona erer. Eğer HYK bir düğüme ulaştığında b-h arasındaki tüm güzergâhlar kesiliyorsa ve bu düğümün AS değeri ÜS değerinden küçük ise AS değeri yeni ÜS değeri olarak atanacaktır.

c. Dal Sınır Algoritmasında Dallanma

Dallanma AK'nın EKG'si üzerindeki daha önce gidilmeyen en küçük AS değerli HYK düğümüne yapılmaktadır. Başlangıç düğümünden dallanma yapılan düğüme gelinceye kadar üzerinden geçilen HYK düğümlerinin bulunduğu AK arkları kesilmiş olarak kabul edilecektir. EKG kesildikten sonra varsa yeni EKG aranacaktır. AK ağındaki yeni EKG aranırken kesilmiş olan ark ve düğümler ağda yokmuş gibi işlem görecektir. Sonrasında yeni EKG üzerindeki HYK düğümlerinin AS değerleri bulunacak ve EKG üzerindeki en küçük AS değerli düğüme doğru dallanmaya devam edilecektir.

Eğer bir düğüme gelindiğinde olurlu bir çözüm bulunuyorsa o düğümden sonra dallanma işlemi duracaktır. Bir düğüme gelen dal budanmışsa o düğümden sonra da dallanma yapılamayacaktır. HYK'nın üzerinde bulunduğu düğümün dallanma şansı kalmamışsa bir üst seviyeye çıkılır ve bu düğüme giden dal budanır. Daha önce gidilmeyen en küçük AS değerli düğümden dallanmaya devam edilir. Sonuç olarak ağacın dallanması AS değerine göre yapılmış olmaktadır.

ç. Dal Sınır Algoritmasının Probleme Uyarlanması

Problemin çözümü için şu ana kadar anlatılanlar doğrultusunda aşağıdaki algoritma oluşturulmuştur. Bu aşamada ÜS değeri sonsuz olarak alınacaktır. En uygun ÜS değerini bulma yöntemleri ileride anlatılacaktır. 1 numaralı düğüm HYK'nın başlangıç düğümü olarak kabul edilmiştir.

ÜS : Üst sınır,

s : EKG üzerindeki her bir HYK düğümü,

AS(s) : HYK'nın EKG üzerindeki her bir s. düğümüne ait alt sınır değeri,

L_{js} : j düğümünden s düğümüne ulaşmak için katedilen EKG'nin uzunluğu,

k : EKG'nin seviyesini gösteren indeks.

Adım 1 : Başlangıç

1.1 $k = 0, AS(1) = 0, \bar{U}S = \infty$.

Adım 2 : Başlangıç ve hedef düğümleri arasındaki bağı kesecek düğümlerin belirlenmesi

2.1 AK ağında $b-h$ düğümleri arasındaki EKG'yi bul. (HYK'nın başlangıç düğümünden dallanacak düğüme gelinceye kadar üzerinden geçtiği AK ağındaki arklar ve düğümler EKG'de olmamalıdır).

2.2 EKG varsa;

- a. $k = k + 1$.
- b. EKG üzerindeki HYK düğümlerini belirle.
- c. $AS(s) = AS(j) + L_{js}$ değerlerini bul.
- ç. Eğer $AS(s) > \bar{U}S$ ise, s . düğümlere gelen dalları buda.
- d. Adım 3'e git.

2.3 EKG yoksa; eğer $\bar{U}S > AS(j)$ ise, $\bar{U}S = AS(j)$ olarak ata. Adım 4'e git.

Adım 3 : Dallanma

- 3.1
- a. Eğer k seviyesinde budanmamış ve gidilmemiş düğüm varsa; en küçük AS değerli düğüme git. Adım 2'ye dön.
 - b. Eğer k seviyesinde gidilebilecek düğüm yoksa; Adım 4'e git.

Adım 4 : Geri izleme

4.1 $k = k - 1$.

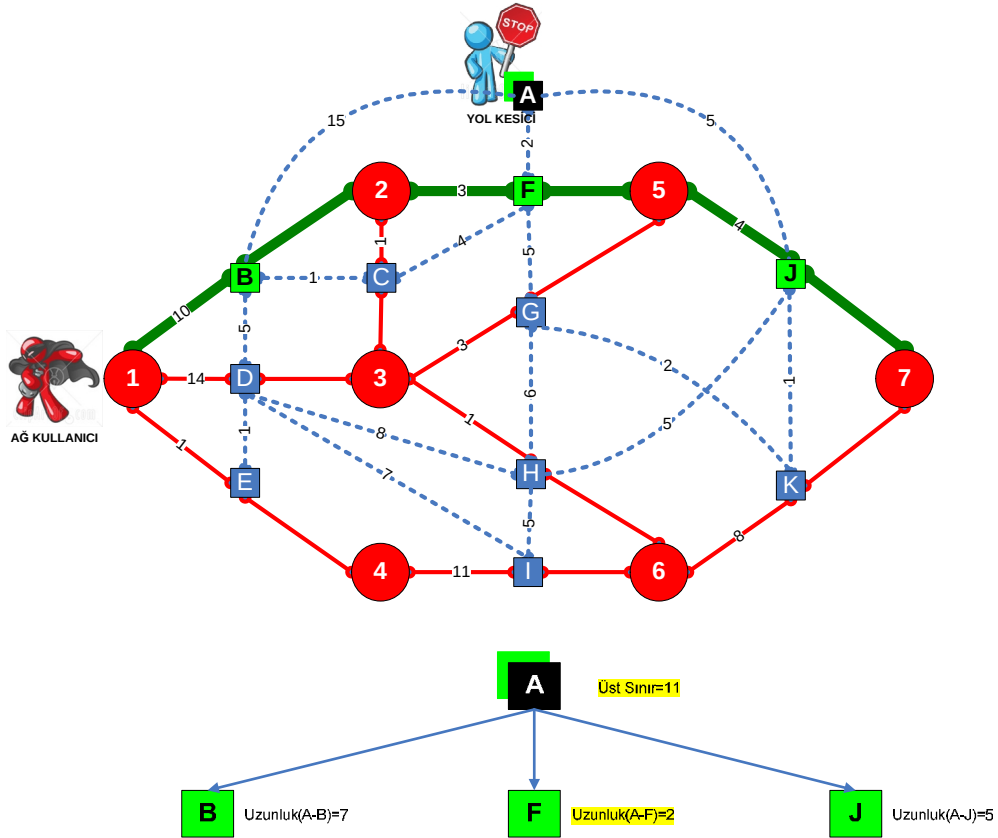
- a. Eğer $k=0$ ise ve gidilecek düğüm kalmadıysa Adım 5'e git.
- b. Aksi durumda, HYK'yı bir üst seviyedeki düğüme taşı. Adım 3'e git.

Adım 5 : Sonlandırma

5.1 Ağacın en son $\bar{U}S$ değerini oluşturan dalı optimal değeri veren güzergâhı içermektedir.

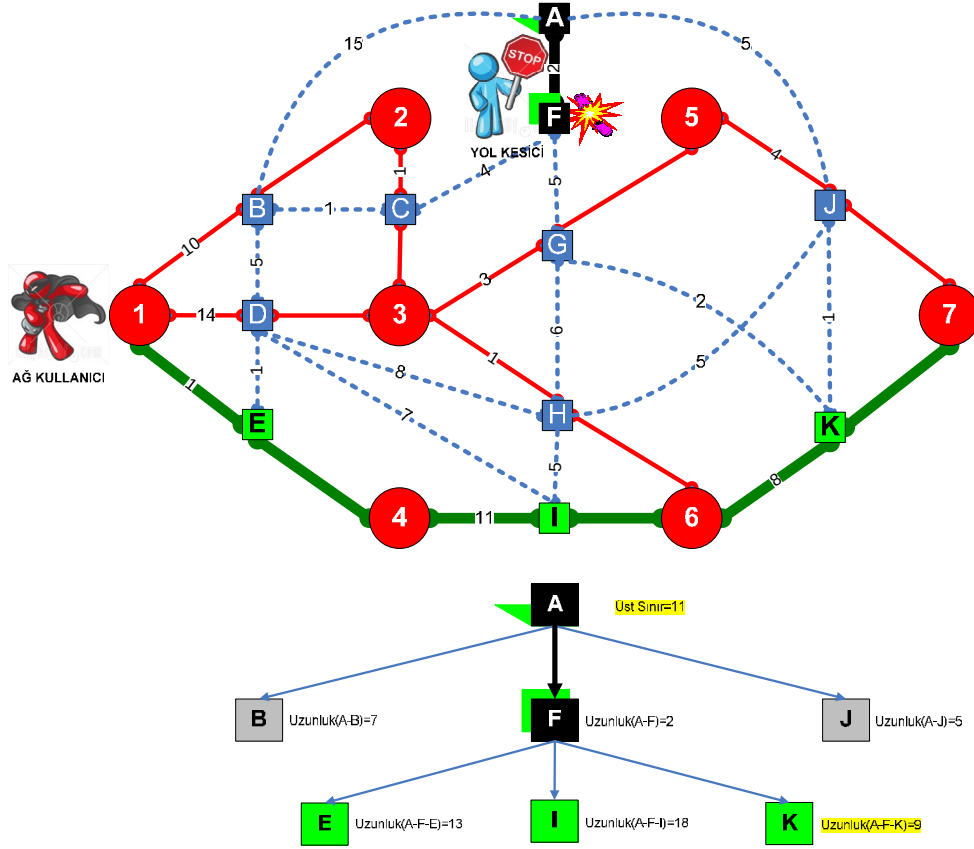
d. Dal Sınır Algoritmasının Uygulanması

Şekil 2'de verilen örnek ağ için yukarıdaki algoritmanın uygulanması bu bölümde şekiller yardımıyla anlatılacaktır. ÜS bu ağ için büyük sayılacak 11 değerini alacaktır.



Şekil 5. İlk EKG'nın Bulunması ve EKG Üzerindeki HYK Düğümlerinin Belirlenmesi

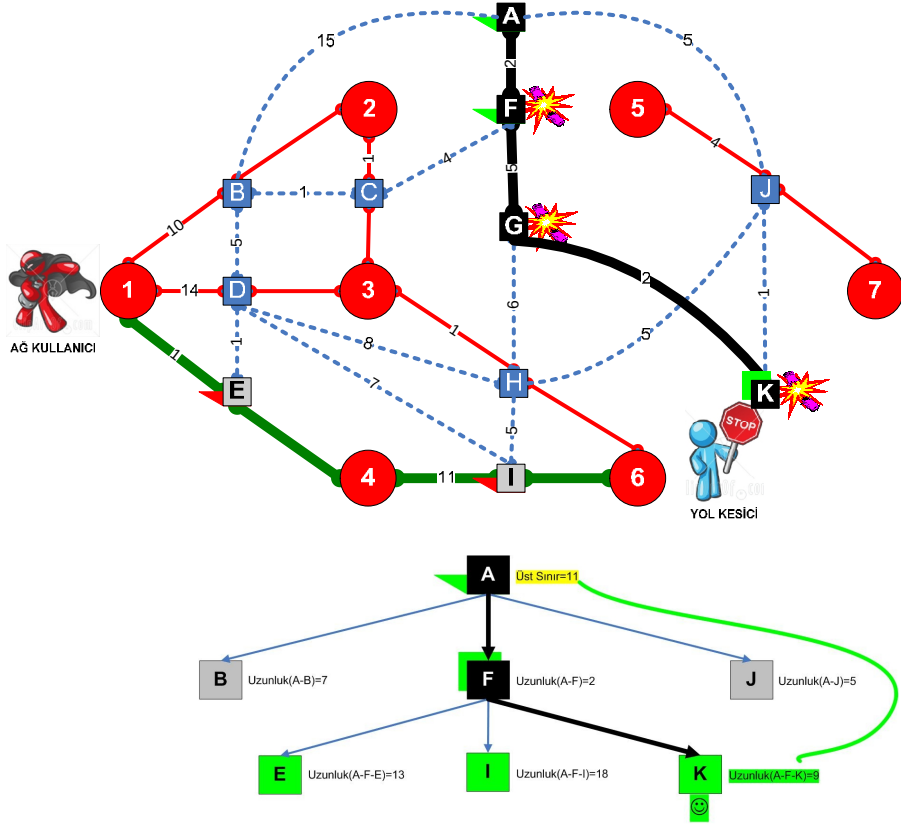
Şekil 5'te başlangıç (1) ve hedef (7) düğümleri arasındaki EKG'nin 1-2-5-7 güzergâhı üzerinde olduğu görülmektedir. Bu güzergâh üzerindeki HYK düğümleri B, F ve J düğümleridir. A düğümüne en yakın olan düğüm F düğümüdür. Dolayısıyla HYK bir sonraki adımında F düğümüne gidecektir. Şekil 5'teki ağın altında dal sınır ağacı gösterilmektedir.



Şekil 6. 2-5 Arkının Kesilmesi ve İkinci EKG'nin Belirlenmesi

Şekil 6'da F düğümüne gelen HYK 2-5 arkını kesmiştir. (2,5) arki kesilmiş haldeki ağda, yeni EKG 1-4-6-7 güzergâhıdır. Bu güzergâh üzerindeki HYK düğümleri E, I ve K düğümleridir. Bir sonraki adımda HYK, F'ye en yakın olan K düğümüne gidecektir.

Şekil 7'de K düğümüne gelen HYK, (6,7) arkını kesmiş ve olurlu bir çözüm bulmuştur. F düğümünden K düğümüne giderken üzerinden geçilen G düğümü de kesilmiştir. K düğümüne gelindiğinde $b-h$ düğümleri arasındaki tüm güzergâhlar kesilmiştir. K düğümünün AS değeri 9'dur. Yani elde edilen olurlu çözümün değeri, 11 olan ÜS değerinden küçüktür. Dolayısıyla ÜS değerinin 9 olarak güncellenmesi gerekmektedir. HYK, F düğümüne geri dönecektir. Fakat F düğümünden gidilebilecek E ve I düğümlerinin AS değerleri ÜS değerinden büyüktür. Dolayısıyla bu iki düğümün olduğu dallar budanacaktır. Daha sonra F düğümünden A düğümüne dönecektir. A düğümünden gidilebilecek bir sonraki en yakın düğüm J düğümüdür. Bu nedenle HYK, J düğümüne gidecektir.

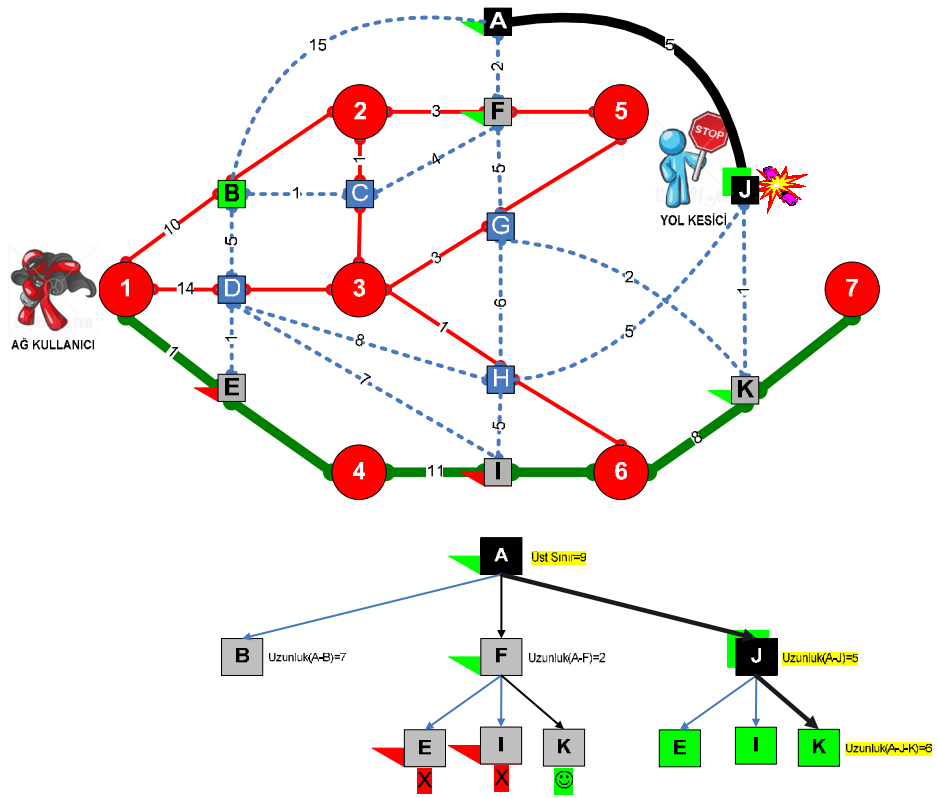


Şekil 7. 6-7 Arkının Kesilmesi ve ÜS Değeri Değişmeden Uygun Bir Sonucun Bulunması

Şekil 8'de J düğümüne gelen HYK (5,7) arkını kesmiştir. (5,7) arkının kesilmiş olduğu durumdaki ağda EKG 1-4-6-7 güzergâhıdır. Bu güzergâh üzerindeki HYK düğümleri E, I ve K düğümleridir. Bir sonraki adımda HYK J'ye en yakın olan K düğümüne gidecektir.

Şekil 9'da K düğümüne gelen HYK (6,7) arkını kesmiş ve olurlu bir çözüm bulmuştur. K düğümünün AS değeri 6'dır. Dolayısıyla 11 olan ÜS değeri 6 ile güncellenecektir. HYK J düğümüne geri dönecektir. Fakat J düğümünden gidebileceği E ve I düğümlerinin AS değerleri yeni ÜS değeri olan 6'dan büyüktür. Bu nedenle bu iki düğümün olduğu dallar budanacaktır. Daha sonra J düğümünden A düğümüne dönelecektir. A düğümünden gidilebilecek bir sonraki en yakın düğüm B düğümüdür. Fakat AS değeri 7 olan B düğümü, ÜS değerinin 6 olması nedeniyle budanacaktır. Dolayısıyla A

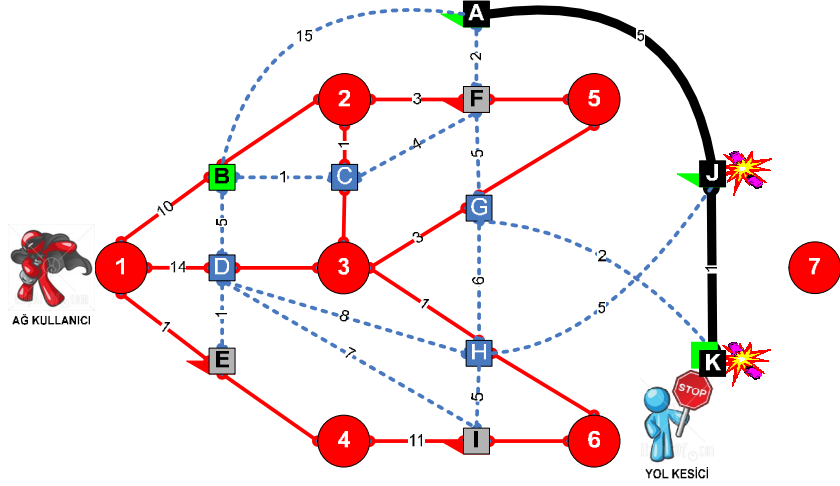
düğümünden gidilebilecek düğüm kalmamıştır. En iyi çözüm A-J-K güzergâhıdır ve en iyi çözüm değeri 6'dır.



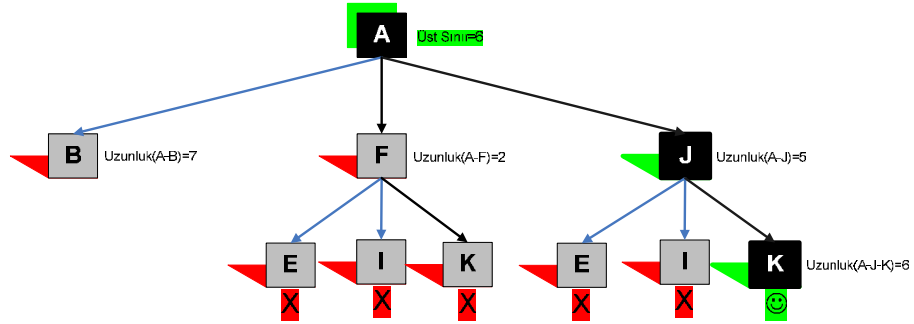
Şekil 8. 5-7 Arkının Kesilmesi ve Yeni EKG'nin Belirlenmesi

Dal sınır ağacının son hali Şekil 10'dedir. Çözümün uygulanması için oluşturulan yukarıdaki şekillerde yol kesici sürekli hareket halinde iken ağ kullanıcı başlangıç düğümünde hep sabit olarak gösterilmektedir. Çünkü bu çalışmadaki problemin tanımında ağ kullanıcının konumunun takip edilmediği belirtilmiştir. Dolayısıyla ağ kullanıcı hep başlangıç düğümünde bekliyor gibi gösterilmiştir.

Çözümün uygulanmasının anlatılması için çizilen ağlarda HYK ve AK'nın insan olarak betimlenmesi çalışmanın uygulanma alanı konusunda yanlış bir izlenim bırakmamalıdır. Çünkü bu uygulama sel suları ve orman yangınlarına engel olma çalışmaları gibi farklı türde hareketli AK'ların olduğu birçok değişik alanda da kendine yer bulabilecektir.



Şekil 9. 6-7 Arkının Kesilmesi ve ÜS Değerinin Güncellenmesi



Şekil 10. Tüm Dallanmalar ve Budanmaların Tamamlandığı Dal Sınır Ağacı

2. ÜST SINIR DEĞERİNİN BELİRLENMESİ

HYK'nın uygun çözüm veren güzergâhlarından birinin uzunluğunu ÜS değeri olarak atamak algorithmada çözümü kolaylaştıracaktır. Ancak uygun

çözümüne sahip birçok güzergâh olabilir. Bu bölümde, bu güzergâhların en kolay şekilde nasıl belirleneceği araştırılmaktadır. Öncelikle güzergâhın nerede aranabileceğini belirlemek için AK ağında kesit oluşturma yöntemleri belirlenmiştir. Daha sonra kesitler üzerindeki HYK düğümlerinden oluşan en uygun güzergâhı belirlemek için çeşitli yöntemler oluşturulmuştur.

a. Ağlarda Kesit Oluşturma

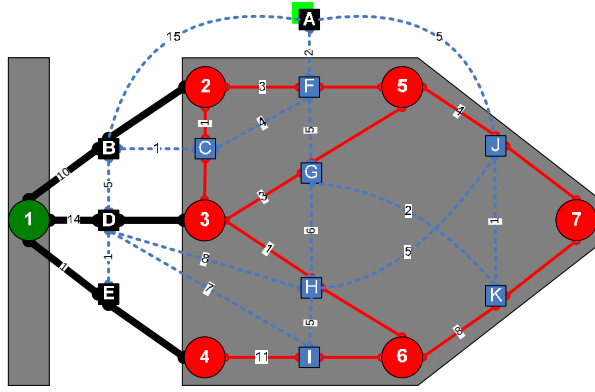
Kesit oluşturmak için iki adet yöntem tanımlanmıştır. Bu yöntemler aracılığıyla, n adet düğüme sahip olan AK ağında b-h düğümlerini birbirinden koparacak n adet kesit belirlemek mümkündür. Bu kesitler üzerinde bulunan HYK düğümleri arasında olabilecek güzergâhlar aranacaktır.

(1) AK Düğümlerini Temel Alarak Kesit Oluşturma

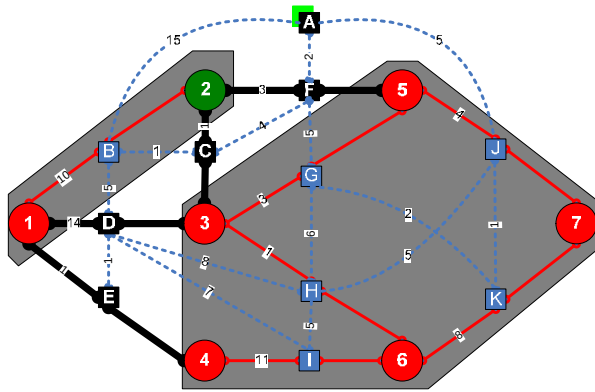
Ağ oluşturma yöntemi anlatılırken AK düğümlerinin ardışık olarak numaralandırıldığı gösterilmişti. Bu aşamada düğüm numaraları kesit oluşturmak için kullanılacaktır.

Bu yöntemde hedef düğümü dışındaki düğümlerin her biri temel alınarak farklı kesitler oluşturulacaktır. Temel alınan düğüm ve bu düğümün numarasından daha küçük numaralı düğümler bir grup olmaktadır. Temel alınan düğümden daha büyük numaralı düğümler ise diğer grubu oluşturmaktadır. İki grubu birbirine bağlayan AK arkları üzerindeki HYK düğümleri kesitin düğümleri olacaktır. Bu yöntemle n adet AK düğümü olan bir ağda $n-1$ adet kesit oluşturulabilmektedir.

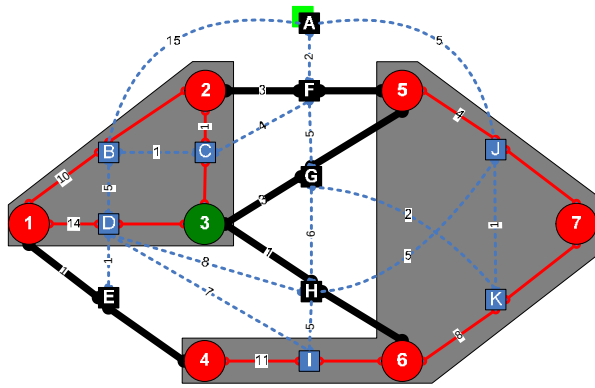
Yöntemin uygulanması Şekil 11-26'te gösterilmektedir. Her bir kesit üzerindeki YK düğümleri arasındaki en kısa uzunluğa sahip güzergâh aranacaktır. Örneğin Şekil 11'deki kesitte A, B, D, E düğümleri bulunmaktadır. Oluşturulan algoritmalar ile bu düğümleri içeren güzergâhlar belirlenecektir. Belirlenen güzergâhların en kısasının uzunluğu ÜS değeri olmaya aday olacaktır. Başka güzergâhlarda daha küçük bir değer oluşmazsa kestteki en kısa güzergâh uzunluğu ÜS değeri olarak atanacaktır.



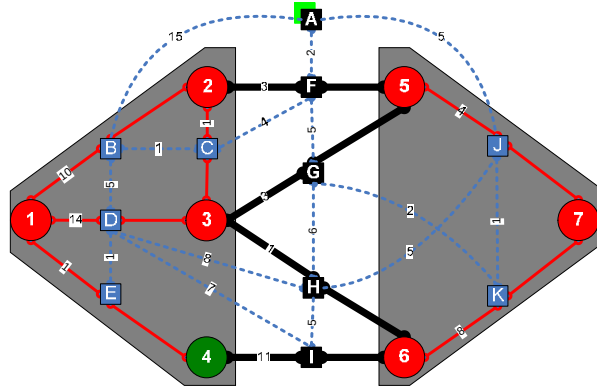
Şekil 11. 1 Numaralı Düğüm Temel Alınarak Oluşturulan Kesit



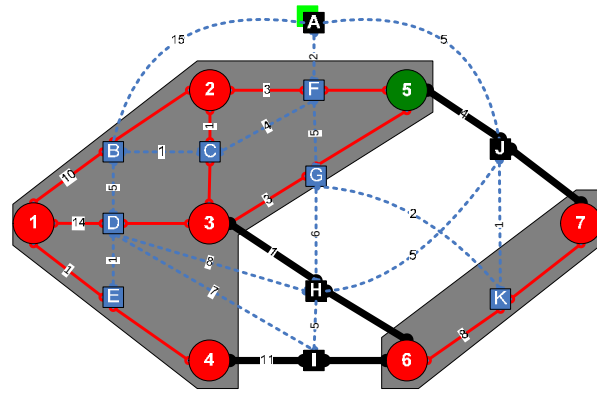
Şekil 12. 2 Numaralı Düğüm Temel Alınarak Oluşturulan Kesit



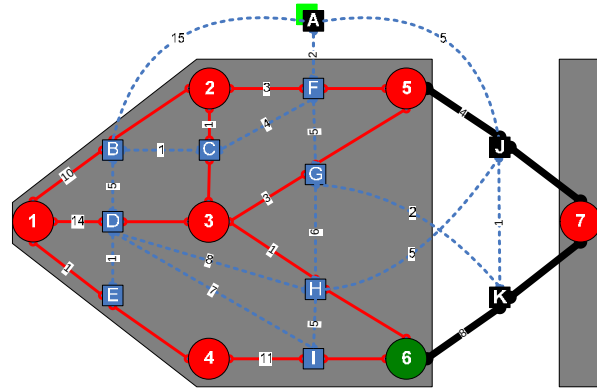
Şekil 13. 3 Numaralı Düğüm Temel Alınarak Oluşturulan Kesit



Şekil 14. 4 Numaralı Düğüm Temel Alınarak Oluşturulan Kesit



Şekil 15. 5 Numaralı Düğüm Temel Alınarak Oluşturulan Kesit



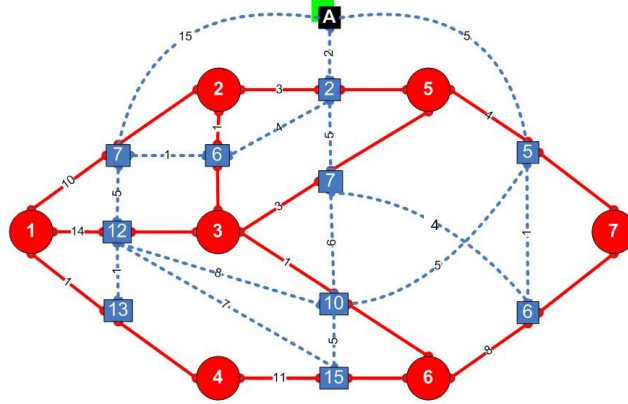
Şekil 16. 6 Numaralı Düğüm Temel Alınarak Oluşturulan Kesit

(2) Maksimum Akış/Minimum Kesme Yöntemiyle Kesit Oluşturma

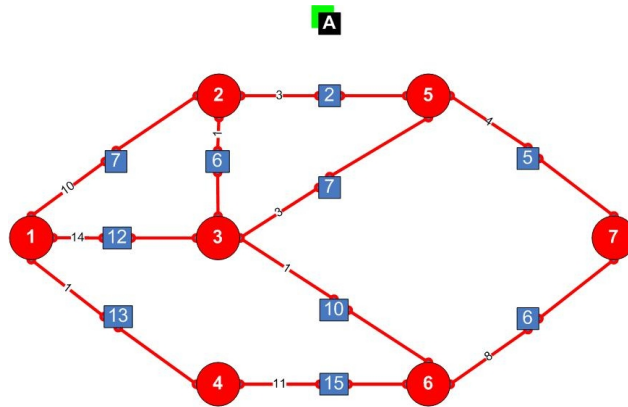
Oluşturulan diğer kesit belirleme yönteminde ise maksimum akış / minimum kesme yöntemi kullanılmaktadır. HYK'nın keseceği düğümlere ulaşması için HYK ağı üzerinde hareket etmesi gerekir. Bu anlamda kesim değeri kesimde bulunan arkların uzunluğundan ziyade HYK'nın başlangıç

düğümünden (A düğümü) bu arklara ulaşmak için katedeceği yol uzunluğu olarak değerlendirilmiştir.

İkinci yöntemin açıklanması aşağıdaki şekiller yardımıyla anlatılacaktır. Şekil 2’de görülen ağ üzerindeki her YK düğümünün merkez düğümü olan A düğümüne uzaklığı belirlenir. Bu durum Şekil 17’de YK düğümlerinin harfleri yerine A düğümüne olan uzaklıkları yazılarak gösterilmiştir. AK ağında 1 ve 2 düğümleri arasındaki arkın uzunluğu 10 birimdir. Fakat HYK’nın A düğümünden bu arka ulaşım değeri 7 birimdir. Bu arkın uzunluk değerinin 10 yerine 7 olduğu farzedilecektir. Diğer tüm AK arklarının uzunlukları da bu şekilde ele alınacaktır. Artık AK ağı Şekil 18’de görüldüğü şekliyle değerlendirilecektir.

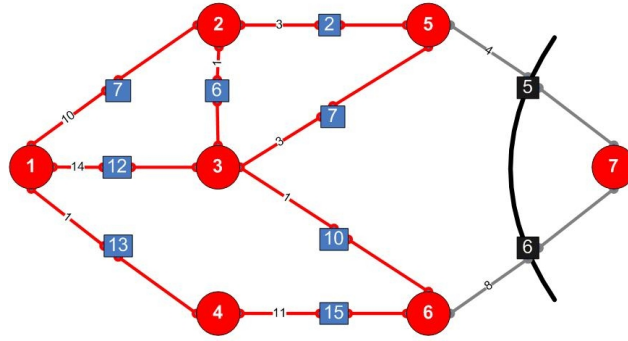


Şekil 17. HYK’nın Başlangıç Düğümünün (A) Diğer Tüm HYK Düğümlerine Olan Uzaklığı



Şekil 18. AK Arklarının Uzunluklarının, HYK’nın A Düğümüne Olan Uzaklıkları ile Değiştirilmesi

Şekil 17’de görülen AK ağı üzerinde Ford-Fulkerson yöntemi yardımıyla maksimum akış / minimum kesme değerleri belirlenir. Minimum kesmeyi sağlayan düğümler yeni kesitin elemanları olarak kabul edilirler. Şekil 2’deki ağ için minimum kesmeyi sağlayan AK arkları (5,7) ve (6,7) arklarıdır. YK’nın başlangıç düğümü olan A düğümü, AK’nın (5,7) arkı üzerindeki J düğümü ve (6,7) arkı üzerindeki K düğümü maksimum akış / minimum kesme kesitinin düğümleridir. Şekil 19’da maksimum akış / minimum kesme kesiti gösterilmektedir.



Şekil 19. Güncellenen AK Ağı Üzerinde Minimum Kesme

(3) En Uygun Kesit Sayısının Aranması

Dal sınır yöntemi ve oluşturulan kesit oluşturma yöntemleri C programlama dili ile kodlanmıştır. 100 ve 500 adet AK düğümüne sahip olan 40’ar ağ üzerinde koşumlar yapılmıştır. Koşum sonuçlarının analizi sonrasında AK ağları için 100 ve 500 adet kesit oluşturulmasına gerçekten ihtiyaç olup olmadığı sorusuna cevap aranmıştır. Bu nedenle ağlar üzerinde farklı miktarlarda kesitler oluşturularak yeniden koşumlar yapılmıştır. Sonuçlar kıyaslanarak çözüm için uygun kesit miktarının ne olabileceği incelenmiştir. Koşum sonuçlarının ayrıntıları ileride anlatılacaktır.

Rastgele sayıda kesit oluşturulurken hangi rassal kesitlerin ele alınacağını bulmak için belirlenen kök değerleri her yerde aynıdır. Yani 100 kesit içerisinde 10 ve 50 tanesi seçiliyorsa 50 kesit bir önce seçilen 10 kesiti de içermektedir. Böylece kıyaslamalar mantıklı olacaktır. Kodlar aşağıda listelenen kesit miktarları ile tek tek çalıştırılmıştır.

- AK ağındaki düğüm sayısının tamamı kadar kesit ile
- AK ağındaki düğüm sayısının rasgele seçilen %50'si kadar kesit ile
- AK ağındaki düğüm sayısının rasgele seçilen %20'si kadar kesit ile
- AK ağındaki düğüm sayısının rasgele seçilen %10'si kadar kesit ile
- AK ağının ilk ve son kesiti ile
- AK ağındaki maksimum akış minimum kesme kesiti ile

b. Kesitler Üzerinde Güzergâh Arama

AK ağının herhangi bir kesitindeki HYK düğümlerinin imha edilmesi durumunda b-h düğümleri arasındaki iletişim koparılmış olacaktır. Ancak burada düğümlerin nasıl bir güzergâh takip edilerek yok edileceği önemlidir. Çünkü amaç en kısa sürede, en kısa uzunluğa sahip güzergâhı bulmaktır. Bu güzergâhın uzunluğu ÜS değeri olarak atanacaktır.

ÜS değerinin belirlenmesi için 6 adet yöntem oluşturulmuştur. Ayrıca tüm HYK düğüm çiftleri arasındaki en kısa güzergâhların ve uzunlukların belirlenerek bu yöntemlere hazır bir şekilde önceden sunulmasının etkileri gözlemlenmek istenmiştir. Güzergâhlar ve uzunluklarının Floyd algoritması ile bulunarak yönteme sunulmasına yöntemin amortize edilmesi denmiştir. Yöntemlerin amortize edilmediği durumlarda ise mesafelerine ihtiyaç duyulan düğüm çiftleri için Dijkstra algoritması ile uzunluklar bulunmaktadır. Amortize edilmiş olan yöntemlerde, ihtiyaç olup olmasına bakılmaksızın tüm düğüm çiftleri arasındaki uzaklıklar baştan hesaplanmaktadır. Amortize edilmemiş yöntemlerde ise sadece ihtiyaç duyulan düğümlerin uzaklık değerleri hesaplanmaktadır. Ancak amortize edilmemiş yöntemlerde her uzaklık arama işleminde ağ baştan sona taranmaktadır. İleride amortize olan ve olmayan durumların kıyaslanması yapılacaktır.

ÜS değerini belirleyecek yöntem ağın belirlenen kesitlerinin her birinde tek tek uygulanmaktadır. Tüm kesitler arasında elde edilen en kısa güzergâh uzunluğu, ÜS değeri olarak kabul edilir. Yöntemin amortize edilmesi durumunda amortize etmek için harcanan süre tüm kesitlere eşit şekilde

bölünerek eklenir. Oluşturulan 6 adet yöntem bir sonraki alt bölümlerde anlatılmaktadır.

(1) Düz Çizgi Yöntemi

HYK'nın başlangıçta üzerinde bulunduğu düğümün diğer tüm HYK düğümlerine olan uzaklığını belirlemek için Dijkstra algoritması uygulanır. Ancak algoritma temel alınan kesit üzerindeki daha önce uğranmamış bir YK düğüme ulaştığında durur. Başlangıç düğümü ile ulaşılan düğüm arasındaki uzaklık algoritmanın durduğu zamanki değeri olarak kabul edilir. Fakat Dijkstra algoritması tüm ağı taramadığı için olabilecek daha iyi uzaklık değerleri gözardı edilmiş olur. Ancak tüm ağın taranmaması işlemin daha kısa sürede tamamlanmasına neden olduğu için ciddi bir zaman kazancı sağlanmaktadır. Hatta düz çizgi yöntemi belirlenen 6 yöntem arasında kesitlerde en çabuk sonuca ulaşan yöntemdir.

Yöntemin uygulanmasının devamında kesit üzerinde ulaşılan düğüm yeni başlangıç düğümü olarak kabul edilir ve tekrar Dijkstra algoritması uygulanır. Kesit üzerindeki tüm düğümlere ulaşıncaya kadar yöntem çalıştırılır. Son düğüme ulaşıldığında elde edilen uzaklık değerlerinin toplamı o kesit için ÜS değeridir. Daha sonra uygulama yapılan tüm kesitlerdeki ÜS değerlerinin en küçüğü dal sınır yöntemine ÜS değeri olarak gönderilir.

Yöntemin uygulanması Şekil 18'deki ağ üzerinde gösterilecektir. HYK başlangıçta A düğümü üzerinde bulunmaktadır. HYK ağı üzerinde Dijkstra algoritması uygulandığında A-B, A-F, A-J düğümleri arasındaki başlangıç uzaklıkları sırasıyla 15, 2, 5 olarak belirlenecektir. A düğümü ile doğrudan bağlantısı olmayan düğümler ile olan mesafeler ise başlangıçta sonsuz kabul edilecektir. Ancak B düğümü ele aldığımız kesit üzerinde olduğu için Dijkstra algoritması durdurulur. HYK'nın artık B üzerinde olduğu kabul edilir. A-B arasındaki en kısa güzergâh A-F-C-B'dir ve bu güzergâhın uzunluğu 7'dir. Fakat Dijkstra algoritması devam ettirilmediği için A-B düğümleri arasındaki mesafe 15 olarak kabul edilecektir. Daha sonra başlangıç düğümü B düğümü olarak kabul edilerek Dijkstra algoritması uygulanmaya başlanacaktır. B-C ve B-D düğümleri arasındaki başlangıç mesafeleri sırasıyla 1 ve 5 olarak

alınacaktır. B düğümü ile doğrudan bağlantısı olmayan diğer düğümler arasındaki başlangıç mesafesi sonsuz olarak alınacaktır. D düğümü kesit üzerinde olduğu için yine Dijkstra algoritması durdurulur ve HYK'nın artık D düğümü üzerinde olduğu kabul edilir. D-E, D-H, D-I arasındaki başlangıç mesafeleri 1,8,7 olarak alınmıştır. D'ye doğrudan bağlantısı olmayan diğer tüm düğümlerin D ile olan başlangıç mesafesi ise sonsuz alınmıştır. E düğümü ele alınan kesit üzerinde olduğu için Dijkstra algoritması durdurulur. Artık HYK'nın E düğümünde olduğu kabul edilir. Kesit üzerindeki tüm düğümlerden geçildiği için yöntem durdurulur. A-B-D-E arasındaki belirlenen mesafe ($15+5+1=20$ birim) bu kesitte bulunan ÜS değeridir.

Bu yöntem oluşturulurken Hamilton sezgisellerinden olan en yakın ekleme yönteminden esinlenilmiştir. Yöntemin amortize edilmeye ihtiyacı yoktur. Çünkü uzaklık değeri yöntemin durduğu andaki değer olarak kabul edilir. Yeniden bir uzaklık bulmaya ihtiyaç duyulmaz. Amortize etmek için harcanan zaman sonuca eklenmediği için yine bir zaman kazancı söz konusu olur.

(2) Düz Çizgi 2 Yöntemi

Düz çizgi 2 yönteminde HYK'nın üzerinde bulunduğu düğüm ile diğer HYK düğümleri arasındaki uzaklıkları belirlemek için Dijkstra algoritması uygulanır. Dijkstra algoritması kesit üzerinde daha önce uğranmamış ilk HYK düğümüne ulaşıldığında bu düğüm bir sonraki başlangıç düğümü olarak belirlenir. Düz çizgi yönteminde algoritma bu noktada durmaktayken düz çizgi 2 yönteminde algoritma devam eder. Amortize edilmemiş yöntemde YK'nın üzerinde bulunduğu düğüm ve ulaşılan YK düğümü arasındaki en kısa uzaklığı bulmak için Dijkstra algoritması tüm ağı tarar. Amortize edilmiş yöntemde ise Dijkstra algoritması ile kesit üzerinde gidilecek bir sonraki düğüm bulunduktan sonra bu iki düğüm arasındaki en kısa uzaklık değeri önceden Floyd ile bulunmuş olan değer olarak atanır. Amortize edilme süresi her kesite eşit olarak paylaştırılarak eklenir.

Şekil 18'deki ağ için düz çizgi yöntemi ile güzergâh A-B-D-E şeklinde bulunmuştu. Düz çizgi 2 yönteminde ise güzergâh A-F-C-B-D-E olacaktır.

Düz çizgi 2 yönteminde düğümler arasındaki en kısa uzaklık değerleri alındığı için güzergâh uzunluğu 13 olurken, düz çizgi yönteminde bu uzunluk 20 olmaktadır. Ancak düz çizgi yönteminin daha hızlı çözüme ulaştığı gözardı edilmemelidir.

(3) En Yakın Komşu Yöntemi

Bu yöntem Hamilton turunu bulmak amacıyla oluşturulan sezgisel yöntemlerden biri olan “en yakın komşu” algoritmasını kullanmaktadır. Yöntem öncelikle HYK’nın başlangıçta bulunduğu düğüm ile kesit üzerindeki tüm HYK düğümleri arasındaki en kısa uzaklıkları bulur. HYK’nın bulunduğu düğüme en yakın olan düğüme gidilir ve bu düğüm bir sonraki başlangıç düğümü olarak atanır. Yeni başlangıç düğümü ile gidilmeyen HYK düğümleri arasında yöntem uygulanmaya devam eder. Kesit üzerindeki tüm HYK düğümlerine ulaşıldığında yöntem durur. HYK’nın bulunduğu düğüm ile kesit üzerindeki düğümler arasındaki uzaklık hem amortize edilmiş hemde amortize edilmemiş durumlar ele alınarak belirlenmiştir.

Yöntemin anlatılması için Şekil 18’deki ağ kullanılacaktır. Ağ üzerindeki kesitte B,D ve E düğümleri bulunmaktadır. HYK’nın başlangıçta üzerinde bulunduğu A düğümü ile kesit üzerindeki düğümler arasındaki en kısa uzaklıklar bulunur. A-B, A-D ve A-E düğümleri arasındaki en kısa uzaklıklar sırasıyla 7,12 ve 13’tür. HYK en yakın uzaklıkta bulunan B düğümüne gider. B düğümü ile kesit üzerindeki gidilmeyen düğümler arasındaki uzaklıklar belirlenir. B-D ve B-E arasındaki en kısa uzaklık değerleri sırasıyla 5 ve 6’dır. YK kendisine daha yakın olan D düğümüne gider. D düğümünden kesit üzerindeki gidilmeyen tek düğüm olan E düğümüne 1 birimlik yol katedilerek geçilir. Katedilen yol 13 birimdir ($7+5+1=13$ birim).

(4) En Ucuz Ekleme Yöntemleri

Diğer 3 çözüm yöntemi Hamilton turunun sezgisel yöntemlerinden biri olan “en ucuz ekleme” algoritmasının türevleridir. En ucuz ekleme algoritmasında başlangıç için 2 adet düğüm seçilmesi gerekmektedir.

Aşağıda bu çalışma kapsamında ilk 2 düğüm olarak hangi düğümlerin seçildiğinin listesi bulunmaktadır. Listedeki her bir seçim için yöntem tek tek ele alınmış ve kodlamalar ona göre yapılmıştır.

- En ucuz ekleme 1 : A düğümü ve kesitteki en küçük numaralı HYK düğümü (Kesitin iki tarafında sınır düğümleri vardır. Bu düğümlerden biri ile A düğümü, ilk iki düğüm olarak alınmıştır.)
- En ucuz ekleme 2 : A düğümü ve kesitteki A'ya en yakın düğüm
- En ucuz ekleme 3 : A düğümü ve kesitteki A'ya en uzak düğüm

Literatürdeki en ucuz ekleme algoritmasında güzergâh başladığı düğümde sonlanmaktadır. Yani bir döngü vardır. Bu çalışmada buna engel olmak için kesitteki her HYK düğümüne ve başlangıç düğümüne güzergâhta bir kez yer verilmektedir. Ayrıca düğümler güzergâhın son düğümünün arkasına da eklenerek kıyaslamalar için alternatif bir güzergâh oluşturulmaktadır. Algoritmanın uygulanma şekli bu farklar dışında literatürdeki ile aynıdır. Belirlenen ilk 2 düğümün arasına ve sonuna kesitteki işleme alınmamış düğümlerin hepsinin yerleştirildiği durumlar tek tek incelenir. Kesitteki hangi HYK düğümü nereye yerleştirilirse en kısa güzergâh oluşur sorusunun cevabına göre yerleştirme yapılır. Artık elimizde 3 düğüm bulunmaktadır. 3 düğümün aralarına ve sonuna kesitteki geriye kalan tüm düğümler tek tek eklenir ve 4 düğümün oluşturduğu güzergâhlardan hangisinin uzunluğu en kısa ise ona göre ekleme yapılır. Bu işlem kesitteki tüm HYK düğümleri güzergâha ekleninceye kadar devam eder.

Bu yöntem kodlanırken yine uzunlukların amortize edilmiş ve amortize edilmemiş durumları ayrı ayrı ele alınmıştır. En ucuz ekleme yöntemlerinin anlatılması için Şekil 18'deki ağ kullanılacaktır. Ağ üzerinde en ucuz ekleme 3 yönteminin nasıl işletildiği gösterilecektir. Ağdaki ele alınan kesit üzerinde B,D ve E düğümleri bulunmaktadır. Bu düğümler arasında A'ya en uzak olan E düğümüdür. Dolayısıyla başlangıç olarak A ve E düğümleri ele alınacaktır.

Tablo 3'e göre A-E arasına B ya da D konulduğunda güzergâh uzamamaktadır. Fakat B ve D düğümleri A-E'nin arkasına eklenirse güzergâh sırasıyla 6 ve 1 birim uzamaktadır. Dolayısıyla A-E arasına bir düğüm

eklenmelidir. A-E'nin arasına B ya da D'nin eklenmesinin farkı olmadığı için rastgele biri seçilecektir (B seçilsin). Artık listemiz A-B-E'dir.

Tablo 3. En Ucuz Ekleme 3 Yönteminin İlk Adımı

i-j	k=B	k=D	min C_k	k^*
A-E	$C_{AB}+C_{BE}-C_{AE}=7+6-13=0$	$C_{AD}+C_{DE}-C_{AE}=12+1-13=0$	0	B,D
E-k	$C_{EB}=6$	$C_{ED}=1$	1	D

Kesit üzerinde eklenmeyen düğümler (D düğümü) A-B ve B-E arasına ve E'den sonraya eklenerek durum incelenecektir. Tablo 4'e göre D düğümü B-E düğümleri arasına eklenecektir. Çünkü bu eklemede güzergâhın uzaması 0 olmaktadır. Artık kesitte eklenmeyen düğüm kalmamış ve kesitteki güzergâhın uzunluğu ve ÜS değeri 13 olmuştur.

Tablo 4. En Ucuz Ekleme 3 Yönteminin İkinci Adımı

i-j	k=D	min C_k	k^*
A-B	$C_{AD}+C_{DB}-C_{AB}=12+5-7=10$	10	D
B-E	$C_{BD}+C_{DE}-C_{BE}=5+1-6=0$	0	D
E-k	$C_{ED}=1$	1	D

BEŞİNCİ BÖLÜM

KOŞUM SONUÇLARI

1. DENEME AĞLARININ OLUŞTURULMASI

Bu tez çalışması kapsamında içiçe girmiş AK ve HYK ağları üzerinde bazı yöntemlerin uygulanmasından bahsedilmektedir. Bu yöntemlerin geçerliliğinin sağlanması için farklı yapılardaki her ağda doğru sonuçlar vermesi gerekmektedir. Bu durum farklı derinlik, yoğunluk ve boyutlarda bir çok ağa ihtiyaç duyulmasına sebep olmaktadır. İhtiyacın karşılanması için C programlama dili ile istenilen yapılarda ağ oluşturan bir kod yazılmıştır.

Deneme ağlarının nasıl oluşturulduğunu anlatmadan önce derinlik ve yoğunluk kavramlarının açıklanması uygun olacaktır. Bu çalışmada bir düğümden başlangıç düğümüne (b) ulaşmak için katedilecek minimum ark sayısı o düğümün derinliği olarak tanımlanmıştır. Yoğunluk ise ağ üzerindeki düğümleri birbirine bağlayan ark sayısının miktarıdır. Yoğunluk kavramı öncelikle AK ağı için tanımlansa da AK ve HYK ağlarının yoğunlukları paralel bir şekilde değişmektedir.

a. Ağ Kullanıcı Ağının Oluşturulması

C programlama dili ile yazılan kod istediğimiz sayıda ve büyüklükte deneme ağı oluşturulmasına olanak tanımıştır. Oluşturulan yöntemlerin denenmesi için 10 ile 1000 arasında AK düğümü içeren bir çok ağ üretilmiştir. Program her koşulda istenilen boyutta 4 farklı tipte 40 adet ağ oluşturmaktadır. Bu tipler şu şekildedir:

- Düşük derinlikli, düşük yoğunluklu (10 adet)
- Yüksek derinlikli, düşük yoğunluklu (10 adet)
- Düşük derinlikli, yüksek yoğunluk (10 adet)
- Yüksek derinlikli, yüksek yoğunluklu (10 adet)

(1) Ağ Kullanıcı Ağında Öbek Oluşturulması

Deneme ağlarını oluştururken öncelikle AK ağı oluşturulmaktadır. Ağ oluşturulurken düğümler öbekler halinde atanmaktadır. 1. öbekte başlangıç düğümü bulunmaktadır. 2. öbeğe başlangıç düğümünden tek bir AK arkı katedilerek ulaşılabilen düğümler konulmaktadır. 2. öbekteki derinlik değeri 1'dir. 3. öbek ise 2. öbekteki en az bir düğümden sadece bir AK arkı katedilerek ulaşılabilen düğümlerden oluşmaktadır. 3. öbekteki derinlik değeri 2'dir. Son öbeğe ulaşınca kadar bu şekilde atama yapılmaya devam edilir. Son öbekte sadece hedef düğümü bulunmaktadır.

(2) Ağ Kullanıcı Düğümlerinin Oluşturulması

Her öbekte olması gereken düğüm sayısının alt sınırı belli bir kurala göre tanımlanmaktadır. Öbekteki alt sınır (oas) belirleme kuralları Tablo 5'te verilmiştir. Alt sınır belirlendikten sonra üzerine ne kadar daha düğüm ekleneceği de (eds) aynı tabloda gösterilmektedir. Tablodaki n değeri ağdaki AK düğüm sayısıdır.

Tablo 5. Öbekteki Düğüm Sayısının Alt Sınırı ve Alt Sınıra Eklenenecek Maksimum Düğüm Sayısının Belirlenme Kuralları

	Öbekteki düğüm sayısının alt sınırı (oas)		oas'a eklenecek max düğüm sayısı (eds)	
	$n < 200$ ise	$n \geq 200$ ise	$n < 200$ ise	$n \geq 200$ ise
Düşük derinlik	$oas = n \times 0.3$	$oas = \lfloor n^2 / 20000 \rfloor$	$eds = n - oas - 1$	$eds = n - oas - 1$
Yüksek derinlik	$oas = 2$	$oas = 2$	$eds = n \times 0.2$	$eds = \lfloor n^2 / 5000 \rfloor$

Her öbekte alt sınıra eklenecek düğüm sayısı rassallık içerir. C programlama dilinde rand() fonksiyonu ile rassal bir sayı (r) üretilir. Bu sayı 0 ile 32767 arasında rassal bir değerdir. Bir öbeğe atanacak düğüm sayısı aşağıdaki formüle göre belirlenmektedir.

$$\text{Öbekteki düğüm sayısı} = oas + r \% eds$$

(3) Ağ Kullanıcı Arklarının Oluşturulması

Her AK arkının uzunluğu 1 ile 20 birim arasında rassal olarak atanmaktadır. 2. öbekten itibaren her öbekteki her düğüm kendisinden önceki öbekteki en az bir düğüm ile mutlaka bağlanmaktadır. Ayrıca son öbek

dışındaki tüm öbeklerdeki her düğüm kendinden sonraki öbeklerdeki en az bir düğüm ile mutlaka bağlanmaktadır.

Düğümler öbeklere atanırken atanma sırasına göre numaralandırılmaktadır. Aynı öbekteki iki düğüm arasında bağlantı olması rassal bir durumdur ve sadece art arda numaralandırılmış düğümler arasında bağlantı olabilmektedir. Sırasıyla art arda düğümler ele alınarak 0 ile 9 arasında rastgele bir sayı üretilmekte ve eğer sayı 4 veya 4'ten küçük bir sayı ise (%50 olasılıkla) düğümler arasına bir ark atanmaktadır.

Ardışık öbeklerdeki düğümler arasında bağlantı kurarken, öncelikle öbeklerdeki düğüm sayıları farkı belirlenmektedir. Daha az düğüme sahip öbekteki düğümlere düşen ark sayısı daha fazla olmaktadır. Arklar daha az düğüme sahip öbekteki düğümler temel alınarak atanmaktadır. Her düğüm zaten bir adet ark ile bağlanmak zorundadır. Az düğüme sahip öbekte fazla bağlantının hangi düğümlere dağılacağını belirlemek için fazlalık sayısına bağlı oluşturulan rassal sayılar kullanılmaktadır. Fazla düğüm içeren öbekte bir düğümün birden fazla ark ile bağlanması olasılığı yine %50 olarak tanımlanmıştır. Böylece oluşan her ağın yoğunluğu rassal olarak belirlenmektedir. Ancak YK ağında yüksek yoğunluk ve düşük yoğunluk kavramları daha net kullanılmaktadır.

Yoğunluğu belirleyen oranlarla ilgili bilinen bir kural olmadığı için değerler belirlenen bir ölçüte göre belirlenmiştir. Ölçüt olarak fazladan eklenecek her arkın oluşma ihtimalinin %50 olmasına karar verilmiştir.

AK düğümleri arasındaki bağlantıyı sağlayan arkların tanımlanmasıyla AK ağının oluşturulması tamamlanmış olmaktadır.

b. Yol Kesici Ağının Oluşturulması

AK ağı üzerindeki tüm arklar üzerinde YK düğümleri olduğu kabul edilmektedir. YK'nın merkez düğümü (A) herhangi bir AK düğümü üzerinde değildir. A ile her derinlikteki en küçük numaralı YK düğümü arasında mutlaka bir bağlantı kurulmaktadır. Derinlikteki bir YK düğümü kendisine komşu olan ve kendisinden büyük numaralandırılmış tüm YK düğümleri ile bağlanmaktadır.

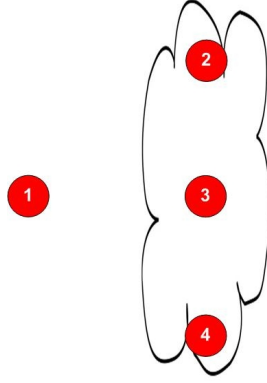
Bir düğümün kendisinden küçük numaralı yani geride kalan düğümler ile bağ kurması rassallık içeren bir durum olarak tanımlanmıştır. Bu durum için 0 ile 9 arasında rastgele bir sayı üretilir. Düşük yoğunluklu ağlarda eğer üretilen sayı 5'ten büyükse düğüm ile geride kalan düğüm arasında bir ark oluşturulur. Ancak oluşturulan ağ yüksek yoğunluklu ise derinlikteki düğüm kendisine komşu olan, kendisinden büyük ve küçük numaralandırılmış tüm YK düğümleri ile bağlanmaktadır. Bu durum aynı AK öbekleri üzerindeki YK düğümleri için geçerli değildir. Çünkü öbekler üzerindeki YK düğümleri öbekler arasındaki YK düğümleri için tanımlanan arklar nedeniyle bağlantısız kalmamaktadır.

YK düğümlerini birbirine bağlayan arkların uzunluğuna 1 ile 20 birim arasında rasgele bir değer atanmaktadır. YK ağının oluşturulması ile içiçe geçmiş AK ve YK ağlarının oluşturulması tamamlanmış olmaktadır. AK ağı düzlemsel bir yapıda oluşturulurken, YK ağı düzlemsel bir yapıda olmak zorunda değildir.

c. Örnek Ağ Oluşturma Uygulaması

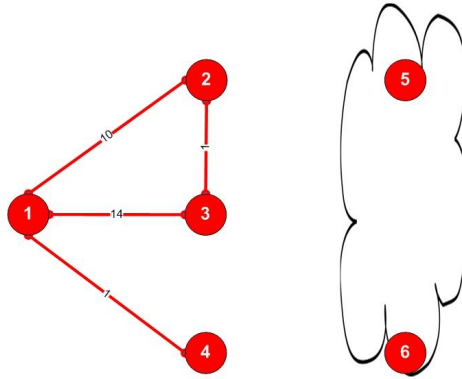
Bu bölümde içiçe girmiş AK ve YK ağlarının oluşturulması işleminin bazı aşamaları şekillere aktararak anlatılmaya çalışılacaktır. AK'nın düğüm sayısının 7 olacağı baştan belirlenmiştir.

Şekil 20'de görüldüğü gibi 1. öbekte sadece 1 numaralı başlangıç düğümü bulunmaktadır. Öbekte olabilecek minimum düğüm sayısı ve bu sayıya eklenecek maksimum düğüm sayısı hesaplanarak 2. öbeğe 3 adet düğüm eklendiği varsayılın. Algoritmaya göre 2. öbeğin derinliği 1'dir. Yani bu öbekteki düğümlerden başlangıç düğümü olan 1 numaralı düğüme en az 1 ark katedilerek ulaşılabilir. Ark atamaları bu mantığa göre yapılacaktır.



Şekil 20. 2. Öbeğe Atanan Düğümler

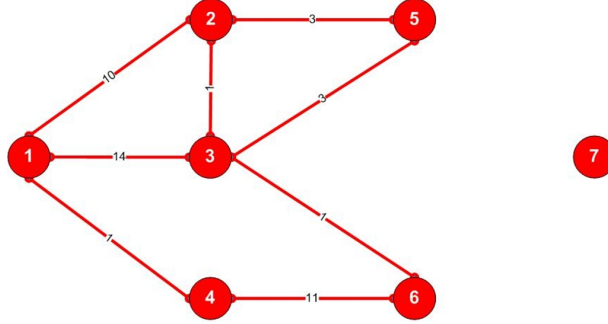
Şekil 21'de 1. ve 2. öbekler arasında bağlar oluşturulmuştur. 1. öbekteki 1 numaralı düğüm ile 2. öbekteki tüm düğümler arasında bağlar kurulmuştur. Ayrıca 2. öbek içerisindeki 2 ve 3 numaralı düğümler arasında belirlenen olasılık dahilinde bir ark oluşturulmuştur. İlk iki öbeğin bağları tamamlandıktan sonra 3. öbek oluşturulmuş ve 2 adet düğüm atanmıştır. 3. öbeğin derinliği 2'dir. Yani 3. öbekteki her düğümden başlangıç düğümü olan 1 numaralı düğüme en az 2 ark katedilerek ulaşılabilir. Yani 3. öbekteki her düğümden başlangıç düğümü olan 1 numaralı düğüme en az 2 ark katedilerek ulaşılabilir.



Şekil 21. İlk 2 Öbeğin Bağlarının Tamamlanması ve Yeni Öbeğin Tanımlanması

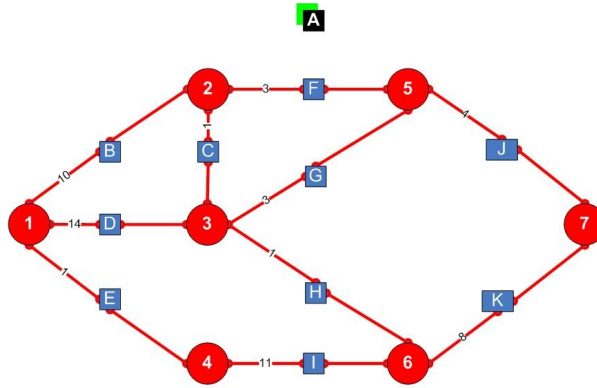
Algoritma gereği 2. ve 3. öbeklerdeki her düğüm diğer öbekteki en az bir düğüm ile bağlanmıştır. İki öbeğin bağlanmış hali Şekil 6'da görülmektedir. Ayrıca Şekil 22'de atanmayan son düğüm olan 7 numaralı düğüm hedef düğüm olarak son öbeğe atanmıştır. Son öbeğin derinlik değeri

3'tür. Yani bu öbekteki düğümden başlangıç düğümü olan 1 numaralı düğüme en az 3 ark katedilerek ulaşılabilir. Şekil 22.



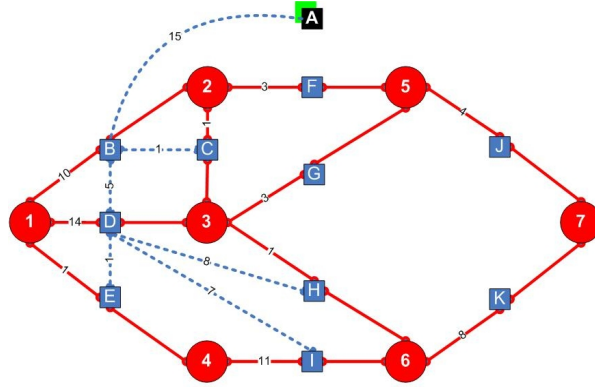
Şekil 22. Kalan Son Düğümün Son Öbeğe Atanması

Şekil 23'te son öbek ile 3. öbek arasındaki bağlar kurulmuş ve AK ağının oluşumu tamamlanmıştır. AK ağının tamamlanması ile birlikte her bir AK arkının üzerinde bir YK düğümü tanımlanmıştır. Bu aşamadan itibaren YK düğümleri arasındaki bağlantılar kurulacaktır.



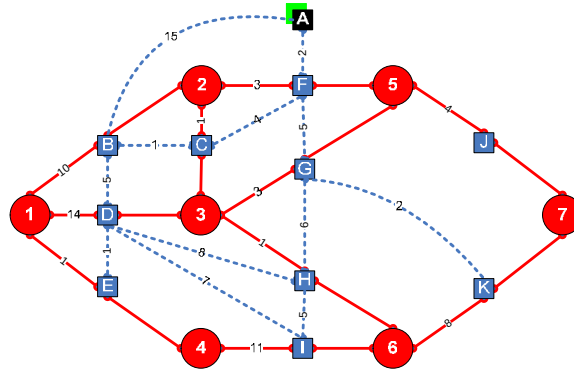
Şekil 23. AK Ağının Tamamlanması ve YK Düğümlerinin Oluşumu

Şekil 24'te YK'nın merkez düğümü olan A düğümü ile ilk derinlikteki YK düğümlerinin en küçük numaralı olan B düğümü arasında bağlantı kurulmuştur. Daha sonra ilk derinlikteki komşu olan tüm YK düğümleri arasına ark atanmıştır. Böylece ilk derinlikteki YK oluşumu tamamlanmıştır.



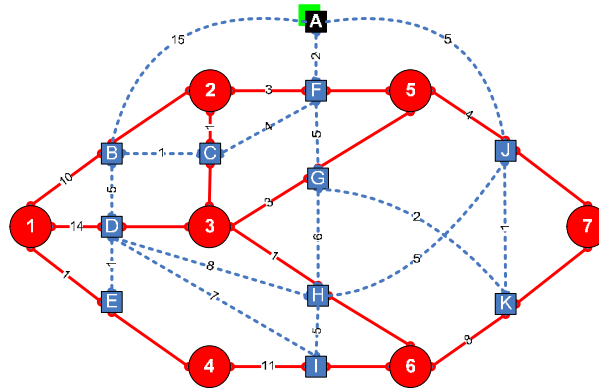
Şekil 24. İlk Derinlikteki YK Bağlarının Oluşturulması

Şekil 25'te YK'nın A düğümü ile AK ağı üzerindeki 2. ve 3. öbek arasındaki YK düğümleri kullanılarak YK ağı oluşturulmaya devam edilmiştir.



Şekil 25. İkinci Derinlikteki YK Bağlarının Oluşturulması

Şekil 26'da ağın oluşumu tamamlanmıştır. Düzlemsel olan AK ağında hiç bir ark birbirini kesmezken, Düzlemsel olmayan YK ağında (H,J) ve (G,K) arkları kesişmektedir. Yani



Şekil 26. Üçüncü Derinlikteki YK Bağlarının Oluşturulması ve Ağın Tamamlanması

2. DAL SINIR ALGORİTMASININ KOŞUM SONUÇLARI

Tanımlanan problemin çözümü için önerilen dal sınır algoritması ve algoritmada iyileştirmeler yapmak için oluşturulan kesit oluşturma yöntemleri ve üst sınır belirleme yöntemleri C programlama dili ile kodlanmıştır. İzleyen bölümde oluşturulan programların koşum sonuçlarının analizi yapılacaktır. Koşumların yapıldığı bilgisayar Intel Core 2, 2.8 GHz işlemci ve 16 GB RAM'e sahiptir.

a. Kesit Verildiğinde En Uygun Yöntemin Araştırılması

Tablo 6 ve Tablo 7 sırasıyla 100 ve 500 adet AK düğümü içeren ağlar için C programlama dili ile elde edilen verilerden oluşmaktadır. 4 farklı ağ türünün her biri için 10 adet ağın analizi ile sonuçlar elde edilmiştir. Tablolar ÜS değerini bulmak için ağlardaki tüm kesitlerin kullanıldığı durumun verilerini içermektedirler.

Tablolarda amortize edilmiş ve amortize edilmemiş yöntemler ile elde edilen çözüm süreleri incelendiğinde her yöntemin amortize edilmiş uygulamasının daha kısa sürelerde çözüme ulaştığı görülmektedir. Amortize edilmemiş yöntemler ise ağlar küçüldükçe daha iyi sonuçlar vermektedir. Fakat burada tüm kesitlerin uygulamaya konulduğunun gözden kaçırılmaması gerekmektedir. Çünkü tüm kesitler içerisinde rasgele %10'u seçildiği durumda yöntemlerin amortize edilmiş ve edilmemiş hallerindeki ortalama çözüm süresi hemen hemen aynı olmaktadır. Kesit sayısı %10'un altına düştüğünde amortize edilmemiş yöntemlerdeki çözümlerin ortalama-sının daha kısa süreler olduğu görülmektedir.

Tablolardaki fark değerleri aşağıdaki formül ile bulunmaktadır. Her yöntem için tüm kesitlerde uygulanan bu formül sonucunda elde edilen sonuçların en büyüğü ve ortalaması tablolarda görülmektedir. Ortalama farkın en düşük değere sahip olduğu yöntem ilgili türdeki ağlarda en iyi sonuca sahiptir.

$$\% \text{ Fark} = \frac{(\text{Yöntemin kesitteki çözüm değeri}) - (\text{Tüm yöntemlerle kesitte bulunan çözümlerin en küçüğü})}{(\text{Tüm yöntemlerle kesitte bulunan çözümlerin en küçüğü})} \times 100$$

Verilerin tablolar şeklinde sunulmasının yöntemleri kıyaslama açısından zor olacağı düşünülerek Şekil 27 ve Şekil 28'de görüldüğü gibi radar çizelgeleri oluşturulmuştur.

Şekil 27 ve Şekil 28'de görülen radar çizelgelerinde merkeze ne kadar yakın değerlere sahip olunursa o kadar iyi sonuçlar elde edildiği düşünülebilir. Çünkü merkeze yaklaştıkça daha kısa çözüm sürelerine, daha iyi/küçük ÜS değerlerine ve diğer yöntemlere göre daha çok kesitte en iyi ÜS değerlerine sahip olunur.

Şekil 27 ve Şekil 28'deki radar çizelgelerine bakıldığında en ucuz ekleme 1 ve en ucuz ekleme 3 algoritmalarının nerdeyse tüm eksenlerde diğer yöntemlere göre daha kötü olduğu ve bu nedenle bu yöntemlerden vazgeçilebileceği söylenebilir. EK A1'den EK A6'ya kadar olan ekler farklı boyut ve yapılarıdaki ağlarda yöntemlerin radar çizelgeleri ile kıyaslanmasını içermektedir. Bu ekler incelendiğinde bu iki yöntemden vazgeçilmesinin gerekliliği daha net ortaya çıkmaktadır. Ayrıca en ucuz ekleme 2 yöntemi için optimal değerden farklara bakıldığında en iyi değerlere sahip olduğu durumlar olduğu görülmektedir. Bu nedenle uygulanmasına devam edilecektir.

Şekil 27 ve Şekil 28'deki radar çizelgelerinin ve EK A1'den EK A6'ya kadar olan eklerin incelenmesinin ardından düz çizgi algoritması her ne kadar tüm kesitlerin çok azında en iyi ÜS değerini bulsa da çözüm süreleri açısından her zaman en iyi olduğu için uygulanmaya değer görülmüştür. (Düz çizgi algoritmasının amortize edilmiş hali olmadığı için çizelgede bu eksenlerde düz çizginin bir değeri bulunmamaktadır.)

Radar çizelgelerinin incelenmesi sonucunda düz çizgi 2 algoritması genelde en iyi/küçük ortalama fark değerine sahiptir. Çözüm süresi açısından da, en iyi değerlere sahip olmasada, gözardı edilemeyecek kadar iyi bir

konumdadır. Bu nedenle bu yöntemde kesit sayısı analizleri yapılınca kadar izlenmeye devam edilecektir.

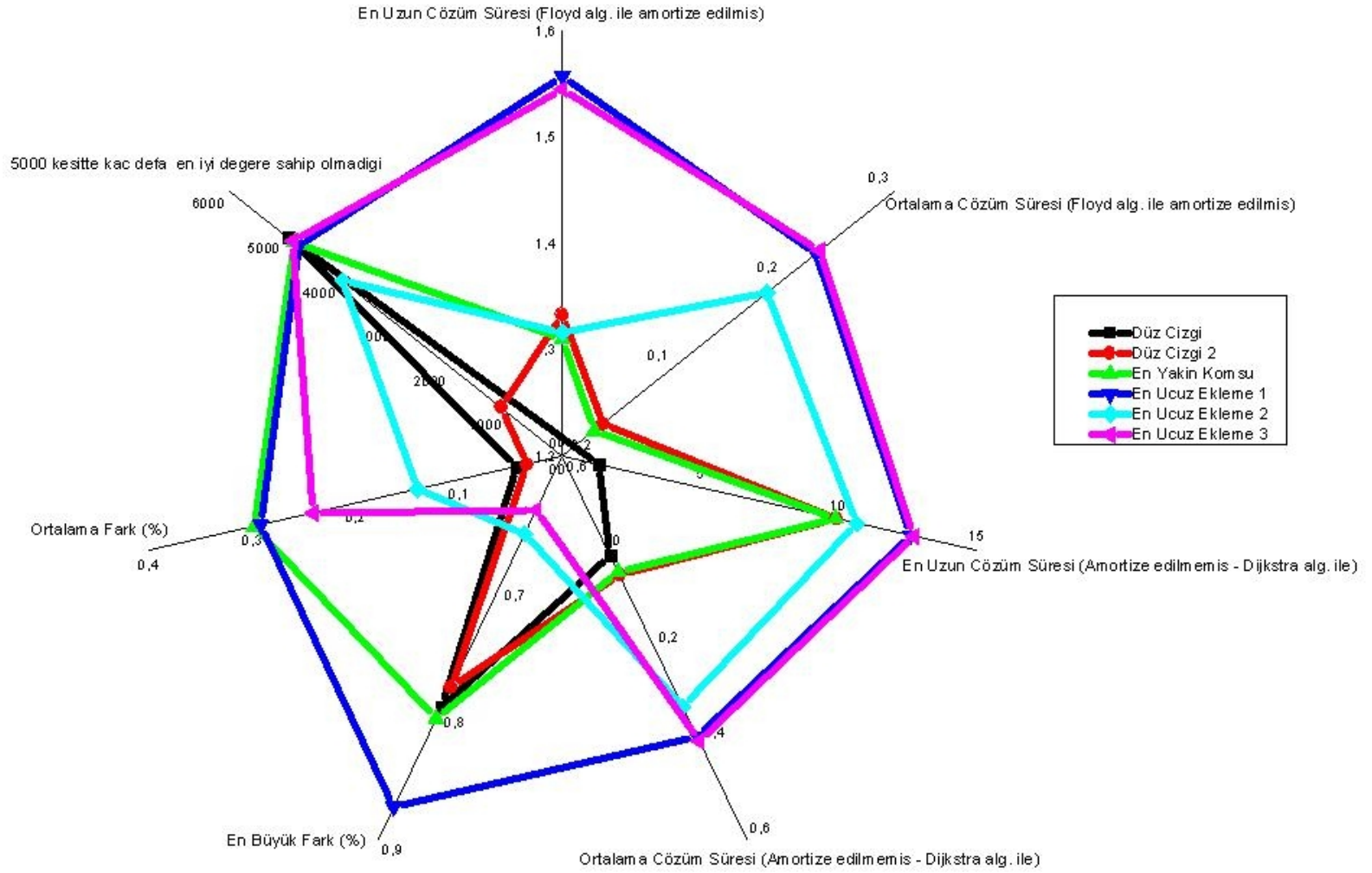
Radar çizelgelerine göre en yakın komşu yöntemi dal sınır algoritması amortize edildiğinde en iyi çözüm sürelerine sahip olan yöntemdir. (Burada amortize edildiği durumda düz çizgi yönteminin hariç tutulduğu unutulmamalıdır.) Fakat bu yöntemin en kötü olduğu eksenler olması nedeniyle bu yöntemden vazgeçilmiştir.

Tablo 6. 100 Düğümlü 4 Farklı Türdeki 10'ar Adet AK Ağında Tüm Kesitler Kullanılarak 6 Farklı Yöntemle Çözümün Analizi

	Algoritma	Amortize edildiğinde (Floyd algoritması ile)		Amortize edilmediğinde (Dijkstra algoritması ile)		En Büyük Fark (%)	Ortalama Fark (%)	1000' e bölünürse kaç defa en iyi değere sahip olmadığı
		Kesitlerin en uzun çözüm sürelisi (saniye)	Kesitlerin ortalama çözüm süresi (saniye)	Kesitlerin en uzun çözüm sürelisi (saniye)	Kesitlerin ortalama çözüm süresi (saniye)			
AK 100 düğümlü (Düşük Derinlik-Düşük Yoğunluk)	Düz Çizgi			0,005320	0,000318	69,08%	6,73%	923
	Düz Çizgi 2	0,005683	0,001404	0,078226	0,001833	69,08%	5,30%	324
	En Yakın Komsu	0,005645	0,001070	0,077937	0,001568	72,34%	17,39%	761
	En Ucuz Ekleme 1	0,008479	0,003042	0,098810	0,004704	76,10%	23,94%	891
	En Ucuz Ekleme 2	0,006247	0,002472	0,084030	0,003994	79,32%	11,39%	601
	En Ucuz Ekleme 3	0,008387	0,003063	0,099743	0,004812	66,88%	22,01%	911
AK 100 düğümlü (Yüksek Derinlik -Düşük Yoğunluk)	Düz Çizgi			0,001513	0,000057	68,22%	3,89%	287
	Düz Çizgi 2	0,001588	0,000584	0,004642	0,000852	68,22%	3,40%	202
	En Yakın Komşu	0,000912	0,000532	0,004656	0,000806	69,30%	6,04%	287
	En Ucuz Ekleme 1	0,001112	0,000651	0,006096	0,001055	75,00%	6,78%	369
	En Ucuz Ekleme 2	0,000962	0,000537	0,005160	0,000893	58,22%	2,23%	190
	En Ucuz Ekleme 3	0,001137	0,000637	0,006092	0,001056	72,00%	4,75%	301
AK 100 düğümlü (Düşük Derinlik-Yüksek Yoğunluk)	Düz Çizgi			0,013950	0,000269	87,68%	0,70%	314
	Düz Çizgi 2	0,015235	0,001980	0,108446	0,002848	87,68%	0,40%	79
	En Yakın Komşu	0,013225	0,001738	0,108262	0,002623	84,84%	5,60%	420
	En Ucuz Ekleme 1	0,020053	0,003860	0,134898	0,005835	81,83%	19,97%	764
	En Ucuz Ekleme 2	0,016349	0,003159	0,114432	0,005007	55,31%	4,37%	345
	En Ucuz Ekleme 3	0,019674	0,003806	0,135144	0,005897	57,91%	16,60%	737
AK 100 düğümlü (Yüksek Derinlik-Yüksek Yoğunluk)	Düz Çizgi			0,001741	0,000043	40,83%	1,19%	206
	Düz Çizgi 2	0,001965	0,000868	0,006137	0,001332	40,83%	0,28%	34
	En Yakın Komşu	0,001457	0,000840	0,006130	0,001325	74,16%	4,70%	196
	En Ucuz Ekleme 1	0,001715	0,001019	0,008017	0,001685	65,79%	9,89%	391
	En Ucuz Ekleme 2	0,001410	0,000839	0,006796	0,001427	62,79%	2,65%	151
	En Ucuz Ekleme 3	0,001665	0,000992	0,008027	0,001686	71,50%	5,19%	326

Tablo 7. 500 Düğümlü 4 Farklı Türdeki 10'ar Adet AK Ağında Tüm Kesitler Kullanılarak 6 Farklı Yöntemle Çözümün Analizi

	Algoritma	Amortize edildiğinde (Floyd algoritması ile)		Amortize edilmediğinde (Dijkstra algoritması ile)		En Büyük Fark (%)	Ortalama Fark (%)	5000' te kaç defa en iyi değere sahip olmadığı
		Kesitlerin en uzun çözüm sürelisi (saniye)	Kesitlerin ortalama çözüm süresi (saniye)	Kesitlerin en uzun çözüm sürelisi (saniye)	Kesitlerin ortalama çözüm süresi (saniye)			
AK 500 düğümlü (Düşük Derinlik-Düşük Yoğunluk)	Düz Çizgi			1,318136	0,007421	79,65%	4,42%	4960
	Düz Çizgi 2	1,333029	0,036741	9,922697	0,048337	78,09%	3,42%	1107
	En Yakın Komsu	1,310240	0,028982	9,903130	0,042343	80,54%	29,79%	4842
	En Ucuz Ekleme 1	1,557495	0,229403	12,622128	0,385724	87,49%	29,15%	4765
	En Ucuz Ekleme 2	1,316399	0,185070	10,667624	0,323301	66,10%	13,99%	3976
	En Ucuz Ekleme 3	1,545349	0,232845	12,738750	0,395434	64,22%	24,11%	4872
AK 500 düğümlü (Yüksek Derinlik -Düşük Yoğunluk)	Düz Çizgi			0,048188	0,000821	74,26%	5,40%	2658
	Düz Çizgi 2	0,053040	0,017788	0,368763	0,026362	67,60%	4,68%	1252
	En Yakın Komşu	0,028319	0,016927	0,368112	0,025779	71,99%	9,10%	2489
	En Ucuz Ekleme 1	0,034805	0,020961	0,451862	0,032568	104,20%	17,22%	3424
	En Ucuz Ekleme 2	0,028356	0,017039	0,409825	0,027749	69,31%	4,19%	1586
	En Ucuz Ekleme 3	0,033686	0,020351	0,485438	0,032820	89,09%	18,24%	3840
AK 500 düğümlü (Düşük Derinlik-Yüksek Yoğunluk)	Düz Çizgi			0,988311	0,005647	67,14%	0,36%	2953
	Düz Çizgi 2	1,014297	0,053229	13,999640	0,075240	66,78%	0,12%	59
	En Yakın Komşu	0,974099	0,047843	13,971425	0,069707	92,88%	7,18%	3307
	En Ucuz Ekleme 1	1,157771	0,252747	17,480322	0,419481	94,41%	23,51%	4808
	En Ucuz Ekleme 2	0,974644	0,206590	14,762921	0,355051	73,50%	9,35%	3459
	En Ucuz Ekleme 3	1,147677	0,248260	17,444189	0,422850	65,78%	19,71%	4915
AK 100 düğümlü (Yüksek Derinlik-Yüksek Yoğunluk)	Düz Çizgi			0,068021	0,000473	50,76%	0,69%	1307
	Düz Çizgi 2	0,078859	0,027704	0,517959	0,043938	50,51%	0,23%	220
	En Yakın Komşu	0,043794	0,027187	0,519340	0,043840	73,40%	5,82%	1752
	En Ucuz Ekleme 1	0,053693	0,033646	0,637861	0,053221	86,44%	16,99%	3297
	En Ucuz Ekleme 2	0,043652	0,027282	0,540569	0,045018	61,84%	3,25%	1196
	En Ucuz Ekleme 3	0,052142	0,032702	0,639078	0,053269	91,76%	17,78%	3984



Şekil 28. AK'nın 500 Düğümlü Olduğu 10 Adet Düşük Derinlik Düşük Yoğunluklu Ağda Yöntemlerin Analizi

b. En Uygun Kesit Sayısının Belirlenmesi

Bir önceki bölümde ÜS değerinin bulunması için uygun yöntemler olarak düz çizgi, düz çizgi 2 ve en ucuz ekleme 2 yöntemleri seçilmişti. Bu yöntemlerle ağılardaki kesitlerin tamamı, rasgele yüzde 50'si, rasgele yüzde 20'si, rasgele yüzde 10'u, ilk ve son kesit ve maksimum akış minimum kesme kesiti kullanılarak ÜS değerleri bulunmuştur. Elde edilen sonuçların analizinin kolay olması için veriler tablolarla değil, doğrudan radar çizelgesi ile verilmiştir.

Radar çizelgesinde verilen fark değerlerinin hesaplanması aşağıdaki formüle göre yapılmıştır.

$$\% \text{ Fark} = \frac{(\text{Yöntemin kesitteki çözüm değeri}) - (\text{YK güzergâhının optimal uzunluk değeri})}{(\text{YK güzergâhının optimal uzunluk değeri})} \times 100$$

Her tür için oluşturulan 10 adet ağın en küçük fark değerleri alınmış ve bunların ortalaması ve maksimum değerlisi radar çizelgesinde gösterilmiştir. Radar çizelgelerinin bir kısmı Şekil 29-30-31'de geri kalanı EK B'de verilmiştir.

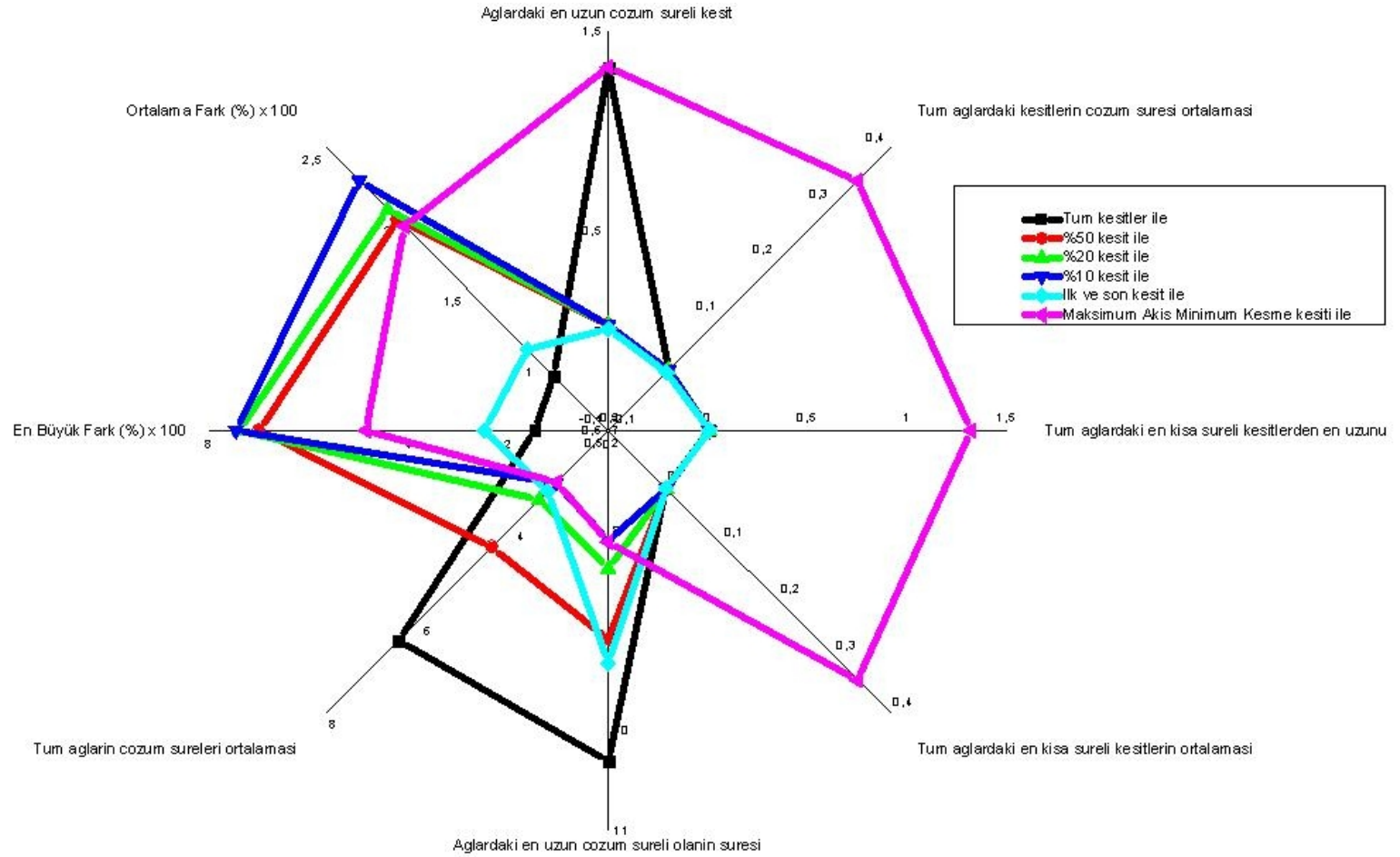
Radar çizelgeleri yine en iyi değerlerin merkezde olduğu bir yapıdadır. Dolayısıyla merkeze yakın değerlere sahip kesit sayısının diğerlerine göre daha üstün olacağı bir yapıda düzenlenmiştir.

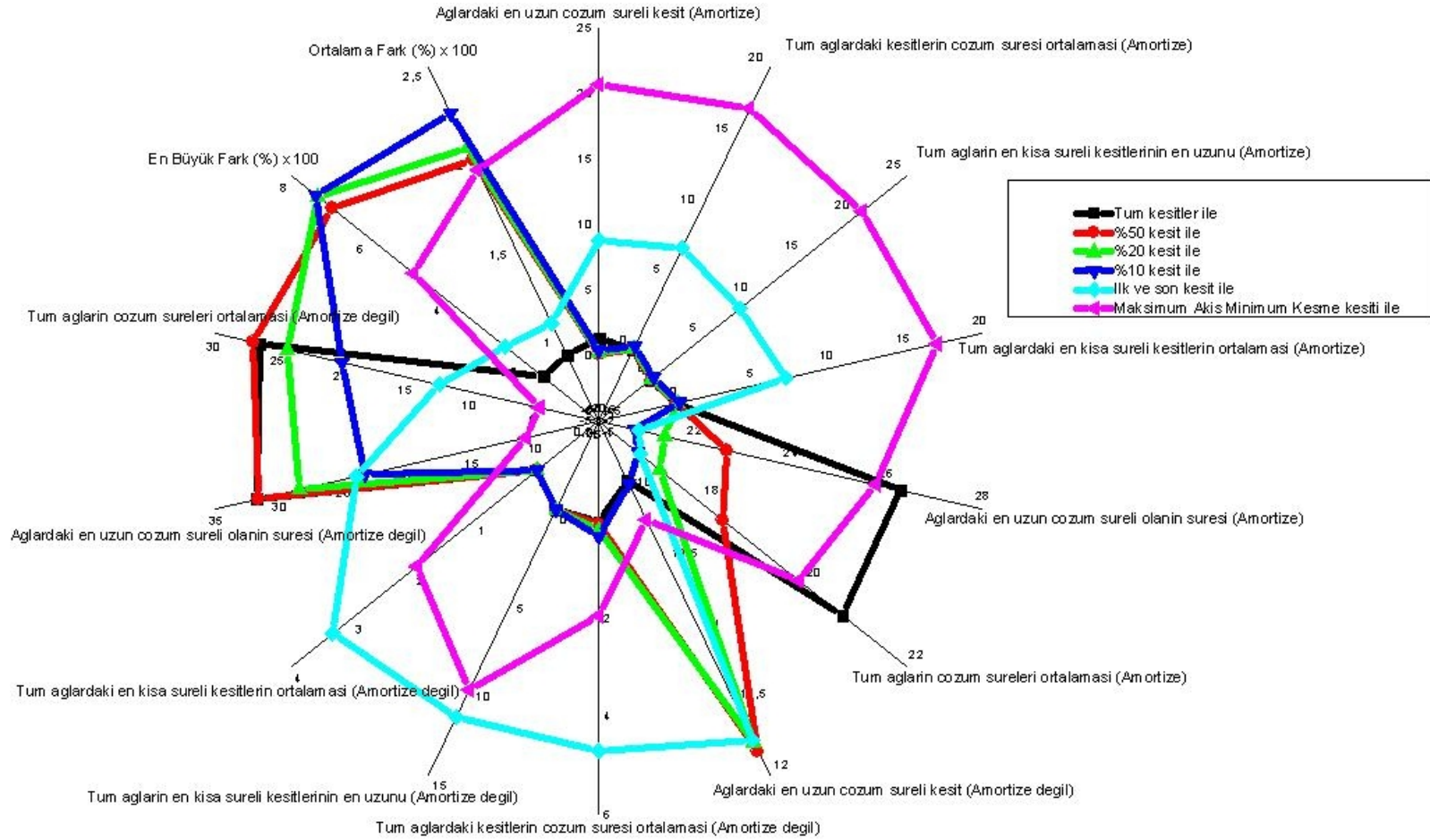
Radar çizelgelerine göre AK düğüm sayısı arttıkça amortize edilmiş yöntemler daha iyi sonuç vermektedir. Kesit sayısının az olduğu maksimum akış minimum kesme kesiti, ilk ve son kesit ve rasgele %10'luk kesit alınması amortize edilmiş yöntemlerde uygun değildir. Az sayıda kesit içeren bu durumlar amortize edilmemiş durumlarda daha iyi sonuçlar vermektedir.

Düz çizgi yöntemi ile genelde en kötü değerlere sahip olan maksimum akış minimum kesme kesitinin bu yöntem ile birlikte kullanımı uygun değildir. Ancak ilk ve son kesitin düz çizgi yönteminde en iyi değerlere sahip olduğu gözlenmiştir. Bu nedenle düz çizgi yöntemiyle kullanımı uygundur.

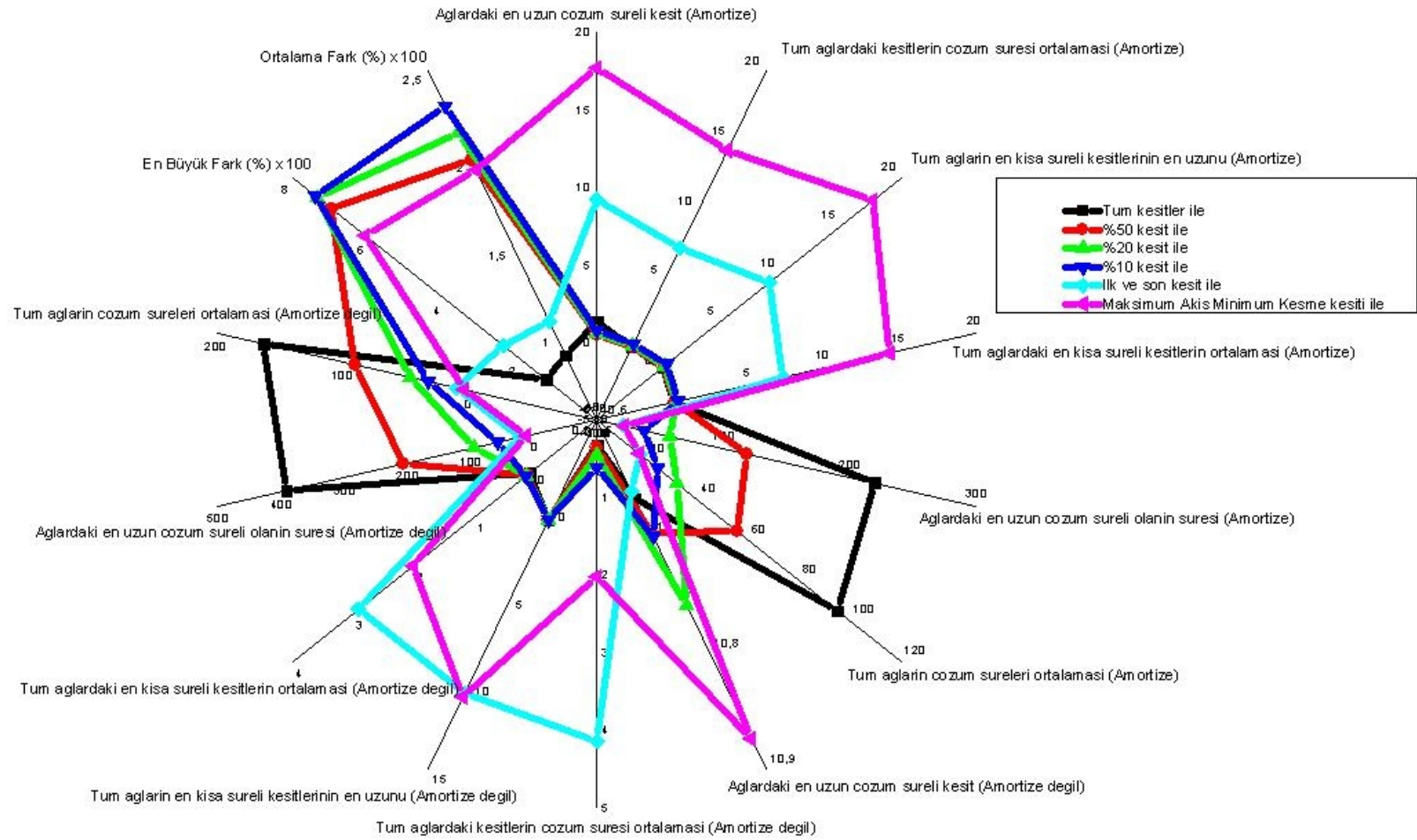
Düz çizgi algoritmasında kısmen, amortize edilmiş yöntemlerde ise açık bir şekilde %50 kesit kullanılması uygun bir sonuç ortaya çıkarmaktadır. Amortize edilmemiş yöntemlerde genel itibarıyla maksimum akış minimum kesme kesiti, ilk ve son kesit daha uygun sonuçlar vermektedir.

Sonuç olarak dal sınır algoritmasının en hızlı çözüme ulaştığı durum düz çizgi yöntemi ile ilk ve son kesitin kullanıldığı durumdur. Bu durum ile başa baş sonuçlar verebileceği söylenecek tek durum ÜS değerinin sonsuz alındığı durumdur. Diğer tüm durumlar daha uzun çözüm sürelerine sahiptir. Ancak ağ büyüdükçe düz çizgi yönteminin etkinliğinin arttığı gözlemlenmiştir.





Şekil 30. 500 Düğümlü Düşük Derinlikli Düşük Yoğunluklu AK Ağında Düz Çizgi 2 Yöntemi Kullanılarak En Uygun Kesit Sayısının İncelenmesi



Şekil 31. 500 Düğümlü Düşük Derinlikli Düşük Yoğunluklu AK Ağında En Ucuz Ekleme 2 Yöntemi ile En Uygun Kesit Sayısının İncelenmesi

3. MATEMATİKSEL MODELİN KOŞUM SONUÇLARI VE KARŞILAŞTIRMA

Problemin çözümü için oluşturulan ikinci model GAMS ile kodlanmış ve çözümleri elde edilmiştir. Modelde AK ağındaki $b-h$ düğümleri arasındaki tüm güzergâhların ve alt turların biliniyor olması gerekmektedir. Bunun için C programlama dili ile bir program yazılmıştır. Program AK ağındaki her güzergâh üzerindeki HYK düğümlerini belirlemektedir. Ayrıca program, üzerinde çalışılan her ağ için doğrudan GAMS kodunu oluşturmaktadır. Alt tur engelleme kısıtları çözüm süresini ciddi bir biçimde arttırdığı için model 2'nin GAMS kodu öncelikle alt tur engelleme kısıtları olmaksızın çalıştırılmıştır. Dal sınır algoritmasının koşum sonuçlarının optimal sonucu verdiği bilinmektedir. Oluşturulan deneme ağlarında GAMS sonuçları ile dal sınır algoritmasının koşum sonuçları kıyaslandığında, ele alınan ağlar için alt tur engelleme kısıtları olmadan da alt tur içermeyen optimal sonuçların bulunduğu görülmüştür. Böylece alt tur kısıtları için harcanacak zaman koşum sonuçlarına dahil edilmeden sonuçlar elde edilebilmiştir. Ancak alt turların oluşabileceği problemlerde optimal sonuca ulaşmak için alt tur engelleme kısıtları ilave edilerek alt tur içermeyen çözümler elde edilebilecektir. Modelin C programı ile hazırlanma aşaması için ve GAMS ile olan koşum zamanı için toplam 3 saatlik zaman limiti konulmuştur. Tablo 6-7-8 sırasıyla AK'nın 10, 20 ve 40 adet düğüme sahip olduğu 40'ar adet ağ için modelin GAMS CPLEX ile koşum sonuçlarını içermektedir. Ayrıca tablolar ağların C ile kodlanan dal sınır algoritmasının koşum sonuçlarını da içermektedir. C kodunda ağlar amortize edilmemiş, kesitlerin %20'si alınmış ve ÜS değerini bulmak için düz çizgi yöntemi kullanılmıştır.

Koşumların yapıldığı bilgisayar Intel Core 2, 2.8 GHz işlemci ve 16 GB RAM'e sahiptir.

Tablo 8'e göre, model 3 saatlik sınır içerisinde 10 adet AK düğümü olan 40 adet ağın hepsinde çözüm sağlayabilmektedir. Tablo 9'a göre, 20 adet AK düğümü olan ağlardan sadece 3 tanesinde zaman sınırları içerisinde çözüme ulaşılamamıştır. Bu 3 ağ 10 adet yüksek derinlikli ve yüksek

yoğunluklu ağ içerisinde bulunmaktadır. Tablo 10'a göre, model 40 adet AK düğümüne sahip olan 40 adet ağın 28 tanesinde zaman sınırı içerisinde çözüme ulaşamamıştır. Bu ağlardan yüksek derinlikli olan 20 adet ağın hiçbirinde çözüm elde edilememiştir. Geriye kalan düşük derinlikli ağlardan düşük yoğunluklu olanlarda 10 adet ağdan 2 tanesinde, yüksek yoğunluklu olanlarda ise 10 adet ağdan 6 tanesinde zaman sınırları içerisinde çözüme ulaşamamıştır.

Tablo 8. AK'nın 10 Düğümlü Olduğu Ağlarda Modelin Çözüm Sonuçları

		AK'nın 10 adet düğümü olduğu 40 farklı ağın model ile çözümünün sonuçları						
		Ağ	Dal-sınır yönteminin sonucu	Dal-sınır yönteminin çözüm süresi (saniye)	Modelin sonucu	Modelin C ile hazırlanma süresi (saniye)	Modelin GAMS ile çözüm süresi (saniye)	Modelin toplam çözüm süresi (saniye)
Düşük Derinlik	Düşük Yoğunluk	1	86	0.000104	86	0.005528	0.104139	0.109667
		2	61	0.000094	61	0.004260	0.308917	0.313177
		3	21	0.000097	21	0.004112	0.273562	0.277674
		4	34	0.000158	34	0.003904	0.161197	0.165101
		5	47	0.000072	47	0.003861	0.218605	0.222466
		6	20	0.000046	20	0.004128	0.158396	0.162524
		7	81	0.000127	81	0.003917	0.754877	0.758794
		8	74	0.000180	74	0.004029	0.221580	0.225609
		9	78	0.000250	78	0.003938	0.232431	0.236369
		10	77	0.000097	77	0.003839	0.103964	0.107803
Yüksek Derinlik	Düşük Yoğunluk	11	46	0.000137	46	0.003812	0.243633	0.247445
		12	15	0.000136	15	0.003773	0.117966	0.121739
		13	31	0.000109	31	0.003892	0.134068	0.137960
		14	27	0.000094	27	0.003689	0.117616	0.121305
		15	18	0.000105	18	0.003892	0.087337	0.091229
		16	27	0.000037	27	0.003686	0.092938	0.096624
		17	28	0.000109	28	0.004288	0.123217	0.127505
		18	13	0.000111	13	0.003910	0.098363	0.102273
		19	6	0.000072	6	0.003820	0.125842	0.129662
		20	11	0.000055	11	0.003831	0.087337	0.091168
Düşük Derinlik	Yüksek Yoğunluk	21	53	0.000142	53	0.004140	0.202327	0.206467
		22	83	0.000178	83	0.004142	0.626410	0.630552
		23	33	0.000182	33	0.004304	0.117616	0.121920
		24	36	0.000091	36	0.004978	0.169598	0.174576
		25	78	0.000086	78	0.004142	0.497767	0.501909
		26	17	0.000054	17	0.004621	0.339021	0.343642
		27	87	0.000118	87	0.004144	0.322569	0.326713
		28	78	0.000161	78	0.004152	1.621944	1.626096
		29	75	0.000104	75	0.004158	0.248709	0.252867
		30	74	0.000113	74	0.004136	0.180449	0.184585
Yüksek Derinlik	Yüksek Yoğunluk	31	33	0.000056	33	0.004324	0.454011	0.458335
		32	11	0.000177	11	0.004301	0.379976	0.384277
		33	21	0.000161	21	0.004735	0.281613	0.286348
		34	13	0.000069	13	0.004320	0.177649	0.181969
		35	25	0.000157	25	0.004344	0.235407	0.239751
		36	22	0.000098	22	0.003959	0.136869	0.140828
		37	18	0.000077	18	0.004295	0.109390	0.113685
		38	36	0.000081	36	0.004159	0.207928	0.212087
		39	22	0.000093	22	0.004138	0.396604	0.400742
		40	14	0.000194	14	0.004072	0.303491	0.307563

Tablo 9. AK'nın 20 Düşümlü Olduğu Ağlarda Modelin Çözüm Sonuçları

		AK'nın 20 adet düğümü olduğu 40 farklı ağın model ile çözümünün sonuçları						
		Ağ	Dal-sınır yönteminin sonucu	Dal-sınır yönteminin çözüm süresi (saniye)	Modelin sonucu	Modelin C ile hazırlanma süresi (saniye)	Modelin GAMS ile çözüm süresi (saniye)	Modelin toplam çözüm süresi (saniye)
Düşük Derinlik	Düşük Yoğunluk	1	146	0.001513	146	0.018968	0.224976	0.243944
		2	77	0.000616	77	0.015567	0.263159	0.278726
		3	136	0.000496	136	0.015942	0.193158	0.209100
		4	89	0.000434	89	0.012155	0.131766	0.143921
		5	72	0.001046	72	0.015155	0.391743	0.406898
		6	76	0.000611	76	0.016973	1.099614	1.116587
		7	166	0.000366	166	0.015994	0.137382	0.153376
		8	129	0.000615	129	0.015915	0.286368	0.302283
		9	81	0.000265	81	0.015635	0.397733	0.413368
		10	85	0.000155	85	0.012399	0.201955	0.214354
Yüksek Derinlik	Düşük Yoğunluk	11	32	0.000220	32	0.021534	1.745906	1.767440
		12	44	0.001034	44	0.064700	6.129762	6.194462
		13	25	0.000221	25	0.025951	2.307598	2.333549
		14	33	0.000310	33	0.041822	1.102608	1.144430
		15	16	0.000321	16	0.034763	2.105643	2.140406
		16	38	0.001115	38	0.032014	1.857084	1.889098
		17	36	0.000261	36	0.067193	2.886322	2.953515
		18	27	0.000182	27	0.028897	1.254402	1.283299
		19	16	0.000675	16	0.024289	0.766267	0.790556
		20	35	0.000301	35	0.073301	5.375287	5.448588
Düşük Derinlik	Yüksek Yoğunluk	21	206	0.001874	206	0.024157	0.707683	0.731840
		22	83	0.001383	83	0.034335	6.173746	6.208081
		23	172	0.000923	172	0.024190	0.649100	0.673290
		24	85	0.000331	85	0.049931	17.897782	17.947713
		25	57	0.000344	57	0.211771	361.325409	361.537180
		26	58	0.000218	58	0.131225	114.315847	114.447072
		27	185	0.000383	185	0.024225	2.301796	2.326021
		28	184	0.000713	184	0.024134	0.713673	0.737807
		29	69	0.000147	69	0.149362	72.478117	72.627479
		30	62	0.000675	62	0.029868	3.737564	3.767432
Yüksek Derinlik	Yüksek Yoğunluk	31	19	0.000197	19	0.105016	12.101366	12.206382
		32	26	0.000134				
		33	23	0.000194	23	0.377176	92.823685	93.200861
		34	11	0.000791	11	9.481522	387.967694	397.449216
		35	21	0.000121	21	0.712826	92.224185	92.937011
		36	35	0.000747	35	0.723275	125.323776	126.047051
		37	67	0.000552	67	0.290391	15.152023	15.442414
		38	32	0.000425				
		39	43	0.001791	43	0.879719	61.259623	62.139342
		40	17	0.000591				

Boş hücreler içeren satırlar 3 saat içerisinde çözilemeyen ağlara aittir.

Tablo 10. AK'nın 40 Düğümlü Olduğu Ağlarda Modelin Çözüm Sonuçları

AK'nın 40 adet düğümü olduğu 40 farklı ağın model ile çözümünün sonuçları							
	Ağ	Dal-sınır yönteminin sonucu	Dal-sınır yönteminin çözüm süresi (saniye)	Modelin sonucu	Modelin C ile hazırlanma süresi (saniye)	Modelin GAMS ile çözüm süresi (saniye)	Modelin toplam çözüm süresi (saniye)
Düşük Derinlik Düşük Yoğunluk	1	292	0.002725	292	2.267936	0.881509	3.149445
	2	133	0.009922	133	1.630591	1.058491	2.689082
	3	302	0.002102	302	2.297660	1.250000	3.547660
	4	157	0.003818	157	1.813829	1.173396	2.987225
	5	102	0.003666				
	6	304	0.003295	304	2.354331	5.291698	7.646029
	7	282	0.003399	282	2.546075	1.869057	4.415132
	8	125	0.004292				
	9	167	0.003472	167	1.630071	2.526415	4.156486
	10	123	0.000766	123	1.990805	4.068302	6.059107
Yüksek Derinlik Düşük Yoğunluk	11	20	0.001652				
	12	25	0.000554				
	13	23	0.00444				
	14	42	0.003772				
	15	39	0.001374				
	16	57	0.002093				
	17	15	0.000711				
	18	16	0.00139				
	19	64	0.003295				
	20	8	0.000354				
Düşük Derinlik Yüksek Yoğunluk	21	351	0.002631	351	1.806413	15.103208	16.909621
	22	116	0.006116	116	0.034335	5.548302	5.582637
	23	387	0.003643	387	1.943699	8.590755	10.534454
	24	104	0.010473				
	25	118	0.002966				
	26	350	0.006201	350	4.843418	14.186226	19.029644
	27	352	0.006335	352	4.981198	11.093774	16.074972
	28	120	0.001629				
	29	127	0.00362				
	30	155	0.002165				
Yüksek Derinlik Yüksek Yoğunluk	31	33	0.000407				
	32	34	0.00237				
	33	25	0.000757				
	34	30	0.001919				
	35	42	0.003788				
	36	75	0.005738				
	37	38	0.003496				
	38	31	0.002295				
	39	67	0.00487				
	40	25	0.000459				

Boş hücreler içeren satırlar 3 saat içerisinde çözülemeyen ağlara aittir.

Tablo 8-9-10'a göre matematiksel modelin ve dal sınır algoritmasının kıyaslanması için Tablo 11 oluşturulmuştur. Tablo 11'e göre dal sınır algoritması tüm ağlarda çözüm sağlarken, matematiksel modelin çözüme ulaşmadığı ağlar bulunmaktadır. Ayrıca dal sınır algoritması düşük yoğunluklu ağlarda binlerce kat daha hızlı çözüme ulaşırken, yüksek yoğunluklu ağlarda onbinlerce kat daha hızlı bir çözüm vermektedir. Yani aynı düğüm sayısındaki ağlarda ark sayısı arttığında model çözüme

ulaşmada dal sınır algoritmasına göre daha fazla zorlanmaktadır. Derinliğin artması yani hedef düğümünün başlangıç düğümüne ulaşabileceği minimum ark sayısının artması her iki yöntemde de çözüm süresini arttırmaktadır.

Tablo 11. Matematiksel Model ve Dal Sınır Algoritmasının Koşum Sonuçlarının Kıyaslanması

		GAMS ile				Dal Sınır Algoritması ile			
		Çözümleme Oranı (%)		Ortalama Çözüm Süresi (saniye)		Çözümleme Oranı (%)		Ortalama Çözüm Süresi (saniye)	
		Düşük Yoğunluk	Yüksek Yoğunluk	Düşük Yoğunluk	Yüksek Yoğunluk	Düşük Yoğunluk	Yüksek Yoğunluk	Düşük Yoğunluk	Yüksek Yoğunluk
AK 10 düğümü	Düşük Derinlik	0.00	0.00	0.257918	0.436933	0.00	0.00	0.000123	0.000123
	Yüksek Derinlik	0.00	0.00	0.126691	0.272558	0.00	0.00	0.000097	0.000116
AK 20 düğümü	Düşük Derinlik	0.00	0.00	0.348256	58.10039145	0.00	0.00	0.000612	0.000699
	Yüksek Derinlik	0.00	30.00	2.594534	114.203183	0.00	0.00	0.000464	0.000554
AK 40 düğümü	Düşük Derinlik	20.00	60.00	5.759874	23.359990	0.00	0.00	0.003746	0.004578
	Yüksek Derinlik	100.00	100.00			0.00	0.00	0.001964	0.002610

4. AĞIN BOYUTU VE OPERASYON SAYISINA BAĞLI OLARAK PROBLEMİN KARMAŞIKLIK ANALİZİ

Problemin karmaşıklığının analizi için AK'nın 200, 300, 400, 500, 750 ve 1000 düğüme sahip olduğu 40'ar adet ağ üretilmiştir. Ağlar ÜS değerinin sonsuz alındığı dal sınır algoritması ile çözülmüştür. Dal sınır algoritmasının C kodunda her ağ için optimal değer aranırken çalışan her komutun sayısı operasyon sayısı olarak toplanmıştır. Optimal değerler ve operasyon sayıları Tablo 12' de verilmiştir.

Her ağ grubu için operasyon sayılarının ortalaması alınmıştır. Ortalama operasyon sayısı ve AK düğümlerinin sayısı arasındaki ilişki aşağıdaki listede gösterildiği şekildedir. (n; AK'nın düğüm sayısı olarak kullanılmıştır. i; ortalama operasyon sayısını ifade etmektedir.)

- $n = 200$ için; $i = n^4 + 114917468 \times n + 21$
- $n = 300$ için; $i = n^4 + 27397770 \times n + 17$
- $n = 400$ için; $i = n^4 + 505322 \times n + 15$
- $n = 500$ için; $i = n^3 + 2520771 \times n + 316$
- $n = 750$ için; $i = n^3 + 7618314 \times n + 191$
- $n = 1000$ için; $i = n^3 + 23118965 \times n + 59$

Operasyon sayısı ve AK düğümleri arasındaki ilişkiye bakıldığında küçük ağlarda operasyon sayılarının en kötü n^4 'e yakınsadığı gözlemlenmiştir. Ancak n arttıkça ortalama operasyon sayısının n^3 düzeyinde olduğu gözlemlenmiştir. Dolayısıyla problemlerin dal sınır yöntemiyle ortalamada $O(n^3)$ asimtotik karmaşıklığında çözülebileceği kabul edilmiştir.

Tablo 12. AK'nın Düğüm Sayısına Göre Ağlarda Dal Sınır Algoritmasının Optimal Çözüm Değerleri ve Operasyon Sayıları

	Ağ	AK 200 düğümlü			AK 300 düğümlü			AK 400 düğümlü			AK 500 düğümlü			AK 750 düğümlü			AK 1000 düğümlü		
		Optimal Değer	Operasyon Sayısı	Çözüm Süresi	Optimal Değer	Operasyon Sayısı	Çözüm Süresi	Optimal Değer	Operasyon Sayısı	Çözüm Süresi	Optimal Değer	Operasyon Sayısı	Çözüm Süresi	Optimal Değer	Operasyon Sayısı	Çözüm Süresi	Optimal Değer	Operasyon Sayısı	Çözüm Süresi
Düşük Derinlik Düşük Yoğunluk	1	1098	2814407855	6.36	1098	1559000611	3.43	947	8649657430	19.78	507	99042066	0.23	1088	220784715	0.51	1050	275573203	0.64
	2	870	2495573269	5.65	870	221337150	0.49	682	1634942234	3.80	231	944097002	2.13	788	6084688115	13.95	620	1367766229	3.21
	3	1058	618482401	1.43	1058	2351831828	5.17	1135	1832668729	4.17	488	1453176356	3.25	792	129581180	0.30	527	8111051987	18.35
	4	937	639653824	1.49	937	36751976494	81.14	2531	407617841	0.93	527	79365582	0.18	289	911771172	2.09	683	9205407221	20.87
	5	2068	1303099309	2.99	2068	1.13999E+11	249.62	778	4518396812	10.16	220	542936288	1.24	603	804374981	1.84	1248	466842064	1.07
	6	523	1293564778	2.93	523	265781165	0.61	1026	15151721226	33.78	574	1338844010	3.07	1084	834286594	1.90	2433	6770499893	15.43
	7	1264	378648837	0.88	1264	1220379644	2.70	393	162002175	0.37	1056	289985266	0.66	1012	878135314	2.01	530	1161444386	2.72
	8	843	837660186	1.92	843	616907101	1.38	1602	36304782857	81.11	562	2179481011	4.89	445	114363203	0.27	871	826170114	1.92
	9	1328	2373519121	5.31	1328	2474795828	5.42	1152	1557999846	3.54	4064	156937450	0.36	710	3885710554	8.66	718	398785354	0.93
	10	899	619585432	1.42	899	802662069	1.82	592	377155699	0.87	680	582839069	1.32	673	299278368	0.70	1212	7726103029	17.61
Yüksek Derinlik Düşük Yoğunluk	11	111	5.0385E+11	1143.01	111	29673314918	62.47	71	2.47365E+11	545.38	30	491380065	1.11	60	5013649409	10.89	71	31705439908	68.23
	12	42	9237238070	19.90	42	1464362821	3.14	34	4463132713	9.75	31	1348420058	2.98	33	436724886	1.00	62	16339620874	35.07
	13	54	333375117	0.76	54	1774688770	3.81	25	2720981027	6.02	35	401045892	0.90	39	1202666745	2.68	73	27882092718	60.00
	14	97	72438828531	156.72	97	2.03426E+11	437.07	49	2455188797	5.47	44	213235885	0.49	33	1103289307	2.47	41	17431711620	37.54
	15	49	74159366	0.18	49	10113905423	21.50	45	6496884175	14.11	57	1366848175	0.31	68	1895745241	4.23	94	13198022790	28.53
	16	20	1924871610	4.20	20	184207430	0.42	20	1842226531	4.08	47	1138625783	2.49	72	3949770526	8.64	116	80485424753	175.21
	17	27	35389692796	76.15	27	347615007	0.77	90	4.41986E+11	998.95	50	4143625205	9.11	54	18126507561	39.32	124	8619007790	18.64
	18	82	5973198660	13.13	82	733718194	1.61	49	621781356	1.40	24	247000556	0.56	31	25678679037	55.97	55	725004400	1.64
	19	92	1.01032E+11	219.01	92	1693321680	3.68	89	91846659763	200.06	16	7613785	0.02	70	9458461524	20.74	112	7584888577	16.48
	20	84	50274751710	109.00	84	35195341194	73.94	45	445752787	1.01	90	3137256694	6.93	46	2802215738	6.10	114	76754441382	165.85
Düşük Derinlik Yüksek Yoğunluk	21	1215	5081129111	11.36	1215	2504192288	5.48	944	2788340425	6.28	493	2494178291	5.63	969	728061513	1.64	1097	381405510	0.88
	22	788	3290356726	7.40	788	5270522129	11.50	556	851144702	1.94	236	134127522	0.30	864	390296773	0.90	506	1797273149	4.15
	23	987	1406744335	3.21	987	547998744	1.24	1102	1080191980	2.44	577	956759278	2.14	744	2187869208	4.87	481	2.21288E+11	483.80
	24	954	19347760177	43.10	954	3841120965	8.50	2658	49824439057	109.99	621	176556396	0.40	304	3242111704	7.27	734	971497479	2.21
	25	2044	2135313846	4.90	2044	2381516054	5.23	660	1855679236	4.16	260	359695274	0.82	591	1496413598	3.38	1147	1712133943	3.84
	26	639	927182418	2.12	639	3149495863	7.06	1203	5556141050	12.39	658	181928314	0.42	1033	476780589	1.09	2540	21025195127	47.10
	27	1277	427461833	1.00	1277	2495263742	5.51	336	5910223624	13.42	1091	210756482	0.48	1023	655021324	1.49	585	3519109786	8.10
	28	740	682725724	1.55	740	603506029	1.37	1704	8781185864	19.37	648	410633882	0.92	523	1012493309	2.32	824	1617131934	3.70
	29	1261	25309385110	55.50	1261	1358984722	3.01	905	1741267791	3.92	4804	68482240	0.16	622	1609064733	3.59	750	389375003	0.90
	30	938	2971347059	6.65	938	3002391840	6.71	607	490230972	1.11	632	200240078	0.45	703	2308377702	5.23	1400	2263513813	5.07
Yüksek Derinlik Yüksek Yoğunluk	31	44	60423926443	129.72	44	16119933661	33.71	94	14991421850	32.25	50	1081423751	2.41	46	76614872652	169.68	62	1.02213E+11	220.93
	32	70	7626861536	16.49	70	2357239226	5.08	6	7264249629	15.70	33	794463974	1.79	41	2468280745	5.48	78	75540032418	162.69
	33	44	2058010032	4.48	44	2152588574	4.64	20	1319833306	2.96	12	484939608	1.08	41	1833942568	4.08	29	2106953599	4.68
	34	107	4513439662	9.80	107	17338444446	36.45	61	8200172061	17.78	34	607577769	1.36	38	2540490557	5.57	80	6031691221	13.27
	35	73	126646728	0.30	73	17763930094	37.45	84	2844027127	6.38	60	1658710553	3.70	52	760222267	1.71	67	76995677296	165.78
	36	47	656198784	1.50	47	47481128607	99.12	23	3750940138	8.30	42	227802979	0.52	38	3338172060	7.27	25	7762618284	16.77
	37	26	25012788635	53.83	26	521863350	1.18	63	39597172411	85.03	52	12226848650	26.99	76	2884553251	6.35	110	3538780377	7.75
	38	85	7488601400	16.36	85	5724859212	12.25	39	309057410	0.71	32	2143034473	4.70	56	41199194789	8.65	54	7359835028	15.76
	39	88	2358535289	5.28	88	2426534538	5.28	87	2794546311	6.24	22	31159266	0.07	70	9898932644	21.50	102	1.02561E+11	221.36
	40	87	17593347915	38.12	87	70862174371	148.64	55	1293786603	2.90	102	12034557698	26.67	36	5948591523	12.91	77	8649643274	18.70
		Ortalama Operasyon Sayısı:		Ortalama Çözüm Süresi:	Ortalama Operasyon Sayısı:		Ortalama Çözüm Süresi:	Ortalama Operasyon Sayısı:		Ortalama Çözüm Süresi:	Ortalama Operasyon Sayısı:		Ortalama Çözüm Süresi:	Ortalama Operasyon Sayısı:		Ortalama Çözüm Süresi:	Ortalama Operasyon Sayısı:		Ortalama Çözüm Süresi:
		24583493621		54.63	16319331017		34.99	25802128816		57.45	1385385817		3.08	6135610692		11.48	24118965591		52.43

ALTINCI BÖLÜM

SONUÇLAR VE GELECEK ÇALIŞMALAR

1. SONUÇLAR

Ağ kesme problemlerinin ele alındığı bu çalışmada, literatürde yer almayan bir ağ kesme problemi tanımlanarak formüle edilmiş ve çözümü için bir dal sınır metodu geliştirilmiştir.

Ağ üzerinde iki adet oyuncu tanımlanmıştır. Oyunculardan biri "hareketli yol kesici (HYK)", diğeri ise "ağ kullanıcı (AK)" olarak adlandırılmıştır. Bir başlangıç düğümünden hedef düğümüne en kısa güzergâhı kullanarak gitmeyi amaçlayan AK ağı esas alınarak oyuncuların ağları içiçe geçmiş iki ağ şeklinde tanımlanmıştır. Her oyuncu ağın sadece kendisi için tanımlanan bölümlerini kullanabilmektedir. HYK kavramı gerçek hayat problemlerinde yaygın olarak görülmesine rağmen literatürde kendisine yer bulamamıştır. Literatürdeki çalışmalarda yol kesicinin hareketli değil sabit olduğu durumlar ele alınmıştır. Bu çalışmada HYK ve sabit yol kesici arasındaki farklar ayrıntısıyla anlatılmıştır.

Oyuncular ağ üzerinde birbiriyle rekabet halindedir ve birinin amacının gerçekleşmesi diğlerinin amacına ulaşamaması anlamına gelmektedir. AK'nın amacı bir düğümden diğlerine en kısa güzergâhı kullanarak ulaşmaktır. HYK, AK'nın başlangıç (b) ve hedef (h) düğümlerinden haberdardır. Ağın kendi kullanabileceği bölümlerindeki en kısa güzergâh üzerinden AK'nın tüm güzergâhlarını kesmeyi amaçlamaktadır. Problemin çözümü için bir matematiksel model oluşturulmuştur ve C programlama dili ve GAMS optimizasyon yazılımı kullanılarak kodlanmıştır. GAMS ile modelin optimal çözümü elde edilmeye çalışıldığında küçük ağlarda bile uzun sayılabilecek sürelerde çözüm verdiği görülmüştür. Bu nedenle problem için bir dal sınır algoritması geliştirilerek hızlı çözüm veren bir çözüm yöntemi oluşturulmuştur. Dal sınır algoritması çok daha büyük ağlarda bile çok daha

hızlı sonuçlar vererek problem için kabul edilebilir sürelerde çözüm ortaya koymaktadır.

Çözüm için oluşturulan yöntemlerin denenebileceği çok sayıda, farklı boyut ve tiplerde ağa gereksinim duyulmuştur. Bu gereksinimi karşılamak için C programlama diliyle yazılan program istenilen boyut ve sayıda ağ oluşturabilmektedir. Ayrıca problem için tanımlanan ağ yapısının daha önce literatürde olmaması nedeniyle oluşturulan ağ oluşturma algoritması da literatürde bir ilk olma özelliğini taşımaktadır.

Matematiksel model ve dal sınır algoritması aynı ağlar üzerinde çalıştırılarak çözüm performansları incelenmiştir. C ve GAMS programlama dilleri kullanılarak kodlanan matematiksel model beklenildiği gibi çözüm süresi açısından daha kötü bir performans ortaya koymuştur. Matematiksel modelin koşumu için 3 saatlik bir zaman kısıtı konulmuştur. Ancak modelin bu süre içerisinde 20 adet AK düğümünün olduğu ağlarda bile çözüme ulaşamadığı olmuştur. Fakat C programlama dili ile kodlanan dal sınır algoritmasının bu süre içerisinde 1000 adet ağ kullanıcı düğümüne sahip ağlarda bile çok rahat bir şekilde çözüme ulaşabildiği görülmüştür.

Çalışmada dal sınır algoritması üzerinde iyileştirme çalışmaları yapılarak daha hızlı sonuç elde edilmeye çalışılmıştır. Algoritmada kullanılan üst sınır değeri ne kadar iyi seçilirse, çözümün de o kadar hızlı elde edileceği düşünülmüştür. Bu nedenle en uygun üst sınır değerini belirlemek için ağ üzerinde kesitler oluşturulmuş ve bu kesitler üzerindeki HYK düğümlerinden oluşan güzergâhlar bulunmaya çalışılmıştır. Bulunan güzergâhlardan en kıtasının uzunluğu üst sınır değeri olarak atanmıştır. Güzergâhları belirlemek için literatürdeki en yakın ekleme, en yakın komşu ve en ucuz ekleme algoritmalarından birini temel alan 6 adet yöntem geliştirilmiştir. Ayrıca yöntemlerin uygulanabileceği 6 farklı miktarda kesit seçilmiştir. Bu kesitlerin oluşturulması için iki çeşit yöntem oluşturulmuştur. En uygun üst sınır değerini bulmak için b düğümünü ağdan ayıran ve h düğümünü ağdan ayıran ilk ve son kesitler üzerinde en yakın ekleme algoritması temelli düz çizgi yönteminin kullanılmasının uygun olduğu görülmüştür.

Problemin karmaşıklığını belirlemek için ağ kullanıcının düğüm sayısı (n) ve dal sınır algoritmasındaki operasyon sayısı arasında ilişki kurulmuştur. Bu ilişki sonrasında problemin $O(n^3)$ operasyon karmaşıklığında çözülebileceğine karar verilmiştir.

2. GELECEK ÇALIŞMALAR

Çalışmada tanımlanan HYK kavramı gerçek yaşamda birçok alanda görülmektedir. Buna rağmen literatürde bu kavramın henüz keşfedilmediğini söylemek abartı olmayacaktır. Ayrıca çalışmada oluşturulan farklı ağ yapısı değişik boyutlarıyla analiz edilmeye açıktır. HYK kavramı ve ağ yapısının farklı yönlerden inceleneceği başka çalışmalar gerçekleştirmek mümkündür. Aşağıda bu çalışmalara örnekler verilmektedir.

i) HYK'nın sınırlı kaynağı olması

Bu tez çalışmasında HYK sınırsız kaynağa sahip olduğu için $b-h$ düğümleri arasındaki tüm AK güzergâhlarını kesene kadar yoluna devam etmekteydi. Kaynağın sınırlı olduğu durumlarda HYK tüm güzergâhları kesemeyebilir. Dolayısıyla daha zor koşullarda bile olsa AK hedefine ulaşabilir. Bu durumda HYK kesebileceği k adet ark ya da düğüm ile AK'nın ulaşımını hangi yönlerden zorlaştıracığına karar vermek zorundadır. HYK'nın kendi ağı üzerindeki en kısa güzergâh üzerinden k adet kesme işlemi ile maksimum sayıda AK güzergâhını kesmesi, AK'nın en kısa güzergâhını maksimize etmesi ya da maksimum akışı minimize etmesi HYK'nın amaçları olarak bu senaryo dahilinde çalışılabilecek alanlardır.

HYK'nın kesebileceği tüm ark ve düğümler için eşit miktarda kaynak harcanmıyorsa, çalışmalarda dikkate alınacak önemli bir unsur daha ortaya çıkmış demektir. Çünkü bir taraftan ark uzunlukları ya da kapasiteleri değerlendirilirken diğer taraftan kesme maliyeti de göz önüne alınmak zorunda kalınacaktır. Önerilen çalışmalar problem tanımlama çizelgesine göre Şekil 32'de gösterilmektedir.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
HYK	Y4,Y5,Y9	TYK	K(+)	B-T	B-T	B		B		B	A1,A2	TAK	TÜ			T(-)	S(-)	B	B, ,	, B,	B

Şekil 32. HYK'nın Kısıtlı Kaynak ile Ağ Kesmesi

ii) HYK'nın AK'yı izleyerek yönünü belirlemesi

AK bir arkın üzerinden geçtiğinde o ark ile işi bitmiştir. Çünkü hedef düğümüne ulaşması için o arka olan ihtiyacı kalmamıştır. Bu tez çalışmasında HYK güzergâhını belirlerken AK'nın konumu ile ilgili bilgiye sahip değildir. O nedenle gereksiz bir şekilde AK'nın üzerinden geçtiği arki kesmeye yönelebilmektedir. Çünkü HYK sadece b ve h düğümlerini bilmektedir ve bu iki düğüm arasındaki bağlantıyı kesmeye odaklanmıştır. Eğer AK'nın ulaştığı her düğümde konum bilgisi HYK tarafından biliniyor olsaydı, durum çok farklı olacaktı. Böyle bir durumda HYK sürekli olarak AK'nın bulunduğu düğümü başlangıç düğümü olarak güncelleyecekti. Bir yandan kendi güzergâhında ilerlerken bir yandan da her başlangıç düğümü için yeniden bir kesme planı oluşturacaktı. Bu dinamik yapı çalışılmaya değer başka bir alan olarak önerilmektedir. Önerilen çalışma problem tanımlama çizelgesine göre Şekil 33'te gösterilmektedir.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
HYK	Y4	TYK	K(-)	B-T	B-T	B	B-T	B		B	A1	TAK	TÜ			T(-)	S(-)	B		B	

Şekil 33. HYK'nın AK'nın Konumunu Bilerek Ağı Kesmesi

iii) HYK ve AK'nın birbirlerini izleyerek yönlerini belirlemesi

Bu tez çalışmasında probleme hep HYK'nın gözünden bakılmaktadır. Ancak AK'nın zeki bir karar verici olduğu ve HYK'yı izleyerek yönünü belirlediği durumlar olabilir. Her iki oyuncunun da birbirini izlediği ve geldikleri her düğümün diğer oyuncu tarafından bilindiği durumlar çalışılmaya değer alanlar olarak görülmektedir. Burada probleme yine HYK tarafından bakılır. Ancak bu sefer HYK'nın kendisini izleyen zeki rakibi AK nedeniyle işi daha zordur. Problemden AK'nın geldiği her düğümde HYK yeni bir kesme planı oluşturur. AK da HYK'nın en kısa sürede tüm güzergâhlarını kesmeyi planladığını bilerek geldiği her düğümde hedefine ulaşma planında gerekli

değişiklikleri yapmak zorunda kalır. Önerilen çalışma problem tanımlama çizelgesine göre Şekil 34'te gösterilmektedir.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
HYK	Y4	TYK	K(-)	B-T	B-T	B	B-T	B		B	A1	TAK	TÜ	B	B-T	T(-)	S(-)	B		B	

Şekil 34. AK ve HYK'nın Birbirlerinin Konumundan ve Amacından Haberdar Olduğu Ağ Kesme Problemi

iv) AK'nın birden fazla başlangıç ve hedef düğümü olması

Bu çalışmada HYK'nın tek yapması gereken AK'nın h düğümüne ulaşmasına engel olmak için en kısa güzergâhı üzerinden yol almaktır. Kendisi için bir tehdit bulunmamaktadır. AK ise hedef düğümüne en kısa sürede ulaşmaya çalışmaktadır. AK kesilen arklar üzerinden geçememektedir.

Her iki oyuncu da diğerinin hangi düğümde olduğu bilgisine sahip olsa ve geçtikleri arkları ve düğümleri diğer oyuncu için kullanılamaz hale getirirse idi problem bambaşka bir yapıya dönüşürdü. Bu yapı askeri faaliyetlerde yaygın olarak oluşan bir durumdur. Örneğin AK olarak kabul edilebilecek teröristler geçtikleri yollara mayın döşeyerek ağ kesme işlemi gerçekleştirmektedirler. Burada yine HYK'nın gözünden probleme bakılarak AK'nın durdurulması konusunda çalışılabilir. Önerilen çalışma problem tanımlama çizelgesine göre Şekil 35'te gösterilmektedir.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
HYK	Y4	TYK	K(-)	B-T	B-T	B	B-T	B		B	A1+A11	TAK	TÜ	B	B-T	T(-)	S(-)	B		B	

Şekil 35. AK'nın da HYK gibi Ağ Kesebildiği Ağ Problemi

v) Farklı düğümlere konuşlanmış birden fazla HYK olması

Bu çalışma kapsamında AK'nın b ve h düğümleri arasındaki bağı kesmeye çalışan bir tane HYK bulunmaktadır. Ancak HYK ağında farklı düğümlerde konuşlanmış birbiriyle eş güdümlü olarak çalışan iki adet HYK olduğu durumda AK'nın güzergâhları daha hızlı bir şekilde kesilecektir. Yapılacak böyle bir çalışmada HYK'ların eş güdümlü atacakları adımların

doğru planlanarak aynı düğüm ya da güzergâhların her iki HYK tarafından kesilmesine engel olmak önem kazanmaktadır. Önerilen çalışma problem tanımlama çizelgesine göre Şekil 36'da gösterilmektedir.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
HYK	Y4	ÇYK	K(-)	B-T	B-T	B		B		B	A1	TAK	TÜ			T(-)	S(-)	B		B	

Şekil 36. İki Tane HYK ile En Kısa Güzergâhın Kesilmesi

vi) İkidenden fazla AK düğümünün hem *b* hem de *h* düğümü olması

Bu çalışmada AK'nın HYK tarafından da bilinen birer adet başlangıç ve hedef düğümü bulunmaktadır. Ancak bir düğüm hem başlangıç hem de hedef düğümü olabilir. Örneğin düğüm başka bir düğüme gıda gönderirken, kendisi de başka bir düğümden silah alabilir. AK'nın bu şekilde çok ürünlü taşıma gerçekleştirdiği ikiden fazla düğümü olsun. Probleme yine HYK'nın gözünden bakarak bu düğümleri birbirinden koparacak en kısa HYK güzergâhını belirleyecek çözümler oluşturulmasına çalışılabilir. Önerilen çalışma problem tanımlama çizelgesine göre Şekil 37'de gösterilmektedir.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
HYK	Y4	ÇYK	K(-)	B-Ç	B-Ç	B		B		B	A1	ÇAK	TÜ			T(-)	S(-)	B		B	

Şekil 37. HYK'nın Çok Ürünlü Taşıma Yapılan Ağda Kesme İşlemi Yapması

vii) AK ağının HYK'ya karşı en iyi savunacak şekilde tasarlanması

İsrail, İran'ın nükleer çalışmalarına en başından beri karşı çıkmakta ve sık sık nükleer tesislere saldırmakla tehdit etmektedir. Ancak bu olası saldırı pek kolay değildir (Sabah Gazetesi [web], 2012). Nükleer tesisler ülkedeki farklı düğüm noktalarına dağılmış durumdadır. İsrailin tek seferde tüm tesislere saldırabileceği bir uçak gücü olmadığı bilinmektedir. Saldırı hava savunmasını aşarak tek bir tesis ile sınırlandırılmak istense bile mesafe nedeniyle ciddi sorunlar ortaya çıkacaktır. Çünkü bu denli büyük bir saldırıyı gerçekleştirebilecek tonlarca ağırlıktaki bomba ile yakıt ikmali yapmadan binlerce kilometre uzağa gidip dönmek mümkün gözükmemektedir. Bu problemler çözülerek tesislere sırayla saldırılacak güç oluşturulsa bile yine

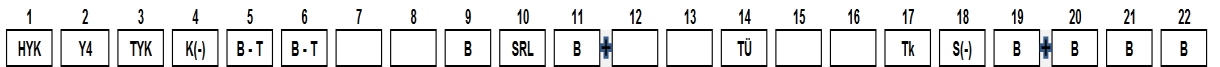
operasyonun başarısı için ciddi engeller bulunmaktadır. Örneğin ilk tesise yapılan habersiz saldırının ardından İran diğer kritik tesislerinde en üst düzeyde savunmaya geçecektir. İsrail'in başarısız olma ihtimalinin temel sebebi tesislerin İran ağındaki yerleşim şeklidir.

İran ve İsrail örneğinden yola çıkarak AK ağının tasarımında kritik tesislerin HYK saldırılarına karşı en iyi savunabilecek şekilde yerleşimi çalışılabilecek diğer bir çalışma alanıdır. Burada diğer önerilen çalışmaların aksine problem AK'nın gözünden çözülmeye çalışılmaktadır. Her iki tarafta birbirinin sahip olduğu olasılıklı güçten haberdardır. Çalışmada AK'nın savunma yeteneği olması dikkat edilmesi gereken bir noktadır.

viii) Kesme işleminde arkların eşit oranda sonucu etkilememesi

Aynı kapasite ve uzunluğa sahip iki farklı arkın kesilmelerinin ardından AK'nın durdurulmasına olan etkileri aynı olmayabilir. Örneğin büyük bir nehir üzerine kurulan bir köprünün imha edilmesiyle, bir otoyolun imha edilmesinin etkileri aynı değildir. Çünkü otoyolun tamir edilmesi daha kolaydır. Ayrıca otoyol imha edilse bile etrafındaki toprak alan kullanılarak daha az kapasite ile de olsa ağın kullanımına devam edilebilir. Ancak aynı durum nehir üzerinde kurulan köprünün imha edilmesi durumunda geçerli değildir. Dolayısıyla kesilecek arklar için uzunluk, kapasite ve kesme maliyetine göre bir öncelik belirlenmesi gerekecektir. Kesilme işlemi bu öncelik sırasına göre gerçekleştiğinde AK'nın durdurulmasına olan etki daha uzun süreli olacaktır.

Bu bölümde HYK'nın en kısa güzergâhı üzerinden en az maliyet ile öncelikli olarak kritik düğümleri kestiği bir ağ kesme probleminin çalışılması önerilmektedir. Burada HYK'nın birden fazla amaç gözetmesi dikkat çekmektedir. HYK bu tez çalışmasında olduğu gibi AK'nın konumu ile ilgilenmemektedir. Önerilen çalışma ayrıntıları Şekil 38'de gösterilmektedir.



Şekil 38. HYK'nın Bir Öncelik Sırasına Göre Ağ kesmesi

KAYNAKÇA

ADDIS, B., M. Di SUMMA, A. GROSSO. "Removing Critical Nodes From a Graph: Complexity, Results, and Polynomial Algorithms for the Case of Bounded Treewidth", Submitted to Optimization Online, 2011.

AKGÜN, İbrahim. The K-Group Maximum-Flow Network-Interdiction Problem, Yüksek Lisans Tez Çalışması, Naval Postgraduate School, 2000.

AKGÜN, İ., B.Ç. TANSEL, R.K. WOOD. "The Multi-Terminal Maximum-Flow Network-Interdiction Problem", European Journal of Operational Research, 2011, 241-255.

ALTNER, D.S., Ö. ERGUN, N. A. UHAN. "The Maximum Flow Network Interdiction Problem: Valid inequalities, integrality gaps, and approximability", Elsevier - Operations Research Letters 38, 2010, 33-38.

ANEJA, Y.P., R. CHANDRASEKARAN, and K.P.K. NAIR. "Maximizing Residual Flow Under an Arc Destruction", Networks Vol.38(4), 2001, 194–198.

ARUSELVAN, A., C.W. COMMANDER, L. ELEFTERIADOUL, P.M. PARDALOS. "Detecting Critical Nodes in Sparse Graphs", Elsevier - Computer & Operations Research 36, 2009, 2193-2200

ASSIMAKOPOULOS, Nikitas. "A network interdiction model for hospital infection control", Computers in Biology and Medicine, Volume 17, Issue 6, 1987, 413-422.

BALL M. O., B. L. GOLDEN, R.V. VOHRA. "Finding the Most Vital Arcs in a Network", Operations Research Letters Vol.8, 1989, 73–76.

BAYRAK, H. ve M.D. BAILEY. "Shortest Path Network Interdiction with Asymmetric Information", Networks, Vol 52(3), 2008, 133-140.

BINGOL, Levent. A Lagrangian Heuristic for Solving a Network Interdiction Problem, Yüksek Lisans Tez Çalışması, Naval Postgraduate School, 2001.

BORGATTI Stephen P. "Identifying sets of key players in a network", Computational and Mathematical Organization Theory 12, 2006, 21–34.

BOYLE M. Partial-Enumeration for Planar Network Interdiction Problems, Master of Science dissertation, Naval Postgraduate School, 1988.

BROWN G.G., M.W. CARLYLE, J. SALMERÓN, R.K. WOOD. "Defending Critical Infrastructure", Interfaces 36, 2006, 530-544.

CHURCH, R.L. ve M.P. SCAPARRA. "Protecting Critical Assets: The r-interdiction median problem with fortification", Kent Business School, Working Paper No.79, February 2005.

CHURLZU, L. ve J.C. SMITH. "Algorithms for Discrete and Continuous Multicommodity Flow Network Interdiction Problems", IIE Transactions 39, 2007, 15-26.

CORMICAN, K.J., D.P. MORTON, R.K. WOOD. "Stochastic Network Interdiction", Operations Research, Vol. 46, No.2, 1998.

DINH,T.N., Y., XUAN, M.T. THAI, E. K. PARK, T. ZNATI. "On Approximation of New Optimization Methods for Assessing Network Vulnerability", In:Proceedings of the 29th conference on information communications, INFOCOM'10, Piscat- away, NJ,USA:IEEE Press, 2010, 2678–2686.

DI SUMMA, M., A. GROSSO. M. LOCATELLI. "Branch and cut Algorithms for Detecting Critical Nodes in Undirected Graphs", Computational Optimization and Applications, 2012,1–32.

DU, Donglei, R. CHANDRASEKARAN. "The Maximum Residual Flow Problem: NP-hardness with Two-Arc Destruction", Networks Vol.50(3), 2007, 181-182.

FORD, L.R., FULKERSON, D.R. "Maximal Flow Through a Network", Canadian Journal of Mathematics Vol. 8, 1956, 399-404.

GARG, M. ve J. C. SMITH. "Models and Algorithms for the Design of Survivable Multicommodity Flow Networks with General Failure Scenarios", Elsevier - Omega Vol. 36(6), December 2008, 1057–1071.

GHARE, P.M., D.C MONTGOMERY, W.C. TURNER. "Optimal Interdiction Policy For A Flow Network", Naval Research Logistics Quarterly Vol.18, 1971, 37–45.

GOLDEN Bruce. "A Problem in Network Interdiction", Naval Research Logistics Quarterly, Vol. 25, Issue 4, 1978, 711–713.

GRANATA, D., G. STEEGER ve S. REBENNACK. "Network interdiction via a Critical Disruption Path: Branch-and-Price algorithms", Elsevier – Computers and Operations Research, Volume 40, Issue 11, 2013, 2689-2702.

HARRIS, T.E. ve F.S. ROSS. "Fundamentals of a Method for Evaluating Rail Net Capacities", 1955, I. Hudson (Ed.), The Maximum Flow Problem, Denison University, 1 Mayıs 2004.

ISRAELI, E. ve R.K. WOOD. "Shortest-Path Network Interdiction", Networks, Vol. 40(2), 2002, 97-111.

JANJARASSUK, Udom ve Jeff LINDEROTH. "Reformulation and Sampling to Solve a Stochastic Network Interdiction Problem", Networks Vol. 52(3), 2008, 120-132.

JOHNSON, M.P., A. GUTFRAIND ve K. AHMADIZADEH. "Evader Interdiction: Algorithms, Complexity And Collateral Damage", Annals of Operations Research, 2013, 1-19.

KENNEDY, Kevin T. Synthesis, Interdiction, And Protection of Layered Networks, Doktora Tezi, Air Force Institute Of Technology, 2009.

LIM C., J.C. SMITH. "Algorithms for Discrete and Continuous Multicommodity flow Network Interdiction Problems", IIE Transactions 39, 2007,15–26.

MCMASTERS, A.W. ve T.M. MUSTIN. "Optimal interdiction of a supply network", Naval Research Logistics, Volume 17, Issue 3, September 1970, 261–268.

MYUNG Y.S., H.J. KIM. "A Cutting Plane Algorithm for Computing k-edge Survivability of a Network", European Journal of Operational Research 156, 2004, 579–589.

PAN, F., W. CHARLTON, ve D.P. MORTON. "Interdicting Smuggled Nuclear Material", Network Interdiction and Stochastic Integer Programming - Operations Research/Computer Science Interfaces Series Vol.22, 2003,1-19.

AHUJA, R.K., T.L. MAGNANTI, and J.B. ORLIN. Network Flows: Theory, Algorithms, and Applications, Prentice Hall, 1993.

ROCCO S., C.M. ve J.E. RAMIREZ-MARQUEZ. "Deterministic Network Interdiction via an Evolutionary Approach", Reliability Engineering and system Safety, 94, 2009a, 568-576.

ROCCO S., C.M. ve J.E. RAMIREZ-MARQUEZ. "Stochastic network interdiction optimization via capacitated reliability modeling and probabilistic solution discovery", Reliability Engineering and system Safety, 94, 2009b, 913-921.

ROCCO S., C.M. ve J.E. RAMIREZ-MARQUEZ. "A Bi-Objective Approach for Shortest-Path Network Interdiction", Computers & Industrial Engineering 59, 2010a, 232-240.

ROCCO S., C.M. ve J.E. RAMIREZ-MARQUEZ, D.E. SALAZAR A. "Bi and Tri-Objective Optimization in the Deterministic Network Interdiction Problem", Reliability Engineering and System Safety 95, 2010b, 887-896.

ROYSET, J.O. ve R.K. WOOD. "Solving the Bi-Objective Maximum-Flow Network Interdiction Problem", Informs, Vol.19, No.2, 2007, 175-184.

Sabah Gazetesi Web Sayfası. 27 Şubat 2012.

<<http://www.sabah.com.tr/NewYorkTimes/2012/02/27/israilin-iranin-nukleer-tesislerine-yapacagi-olasi-saldiri-pek-kolay-degil>>.

SHEN, S., J.C. SMITH ve R. GOLI. "Exact interdiction models and algorithms for disconnecting networks via node deletions", Elsevier – Discrete Optimization 9, 2012, 172-188.

SUMMA M.D., A. GROSSO, ve M. LOCATELLI. "Complexity of the Critical Node Problem Over Trees", Computers & Operations Research Vol.38, 2011, 1766-1774.

UYGUN Adnan. Network Interdiction by Lagrangian Relaxation and Branch and Bound, Yüksek Lisans Tez Çalışması, Naval Postgraduate School, 2002.

WASHBURN, A. ve K. WOOD. "Two Person Zero-Sum Games For Network Interdiction", Operations Research, Vol.43, No.2, 1995, 243-251.

WOLLMER, Richard. "Removing Arcs from a Network", 1964, M.O. BALL, B.L. GOLDEN, R.V. VOHRA (Ed.), "Finding the most vital arcs in a network", Elsevier - Operations Research Letters, Volume 8, Issue 2, April 1989, 73–76.

WOOD, R.Kevin. "Deterministic Network Interdiction", Mathematical and Computer Modeling, Vol. 17, No.2, 1993, 1-18.

WOOD, R.K. "Bilevel Network Interdiction Models: Formulations and Solutions", in Wiley Encyclopedia of Operations Research and Management Science. DOI: 10.1002/9780470400531.eorms0932, 2011.

YATES J. ve S. SANJEEVI. "A Length-Based, Multiple-Resource Formulation for Shortest Path Network Interdiction Problems in the Transportation Sector". international journal of critical infrastructure protection 6, 2013, 107–119.

ZENKLUSEN, R. "Network Flow Interdiction on Planar Graphs", Optimization and Applications Seminar, Swiss Federal Institute of Technology, Zurich, Volume 158, Issue 13, 6 July 2010, 1441–1455.

ZHENG, J. ve D.A. CASTANON. Stochastic Dynamic Network Interdiction Games, American Control Conference, 27-29 June 2012, 1838-1844.

EKLER

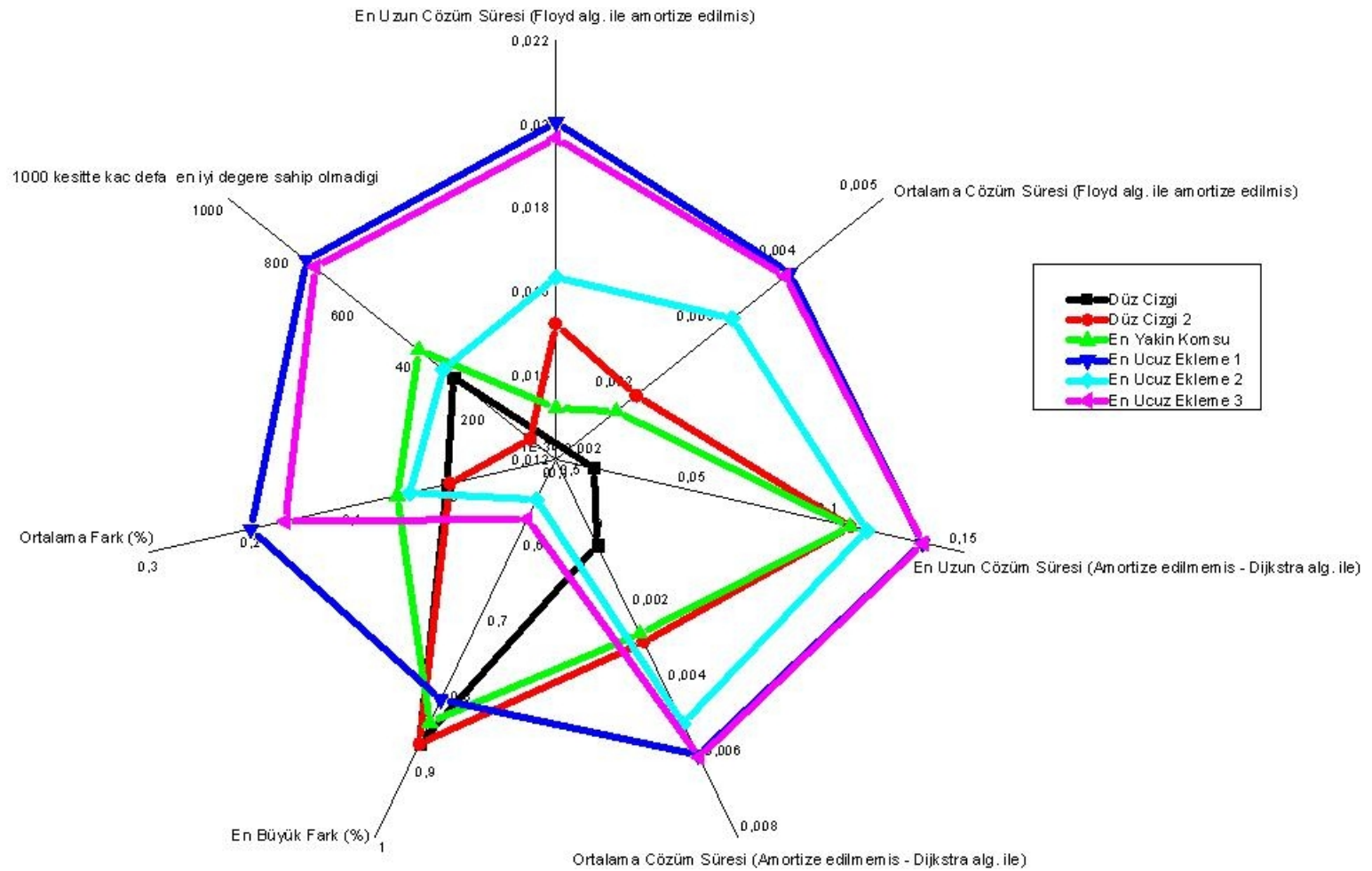
EK-A: Üst sınır değerini bulmak için kullanılan yöntemlerin farklı türlerdeki ağlarda kıyaslanması

EK-B: Üst sınır değerini bulmak için kullanılan kesit sayılarına göre çözümün analizi

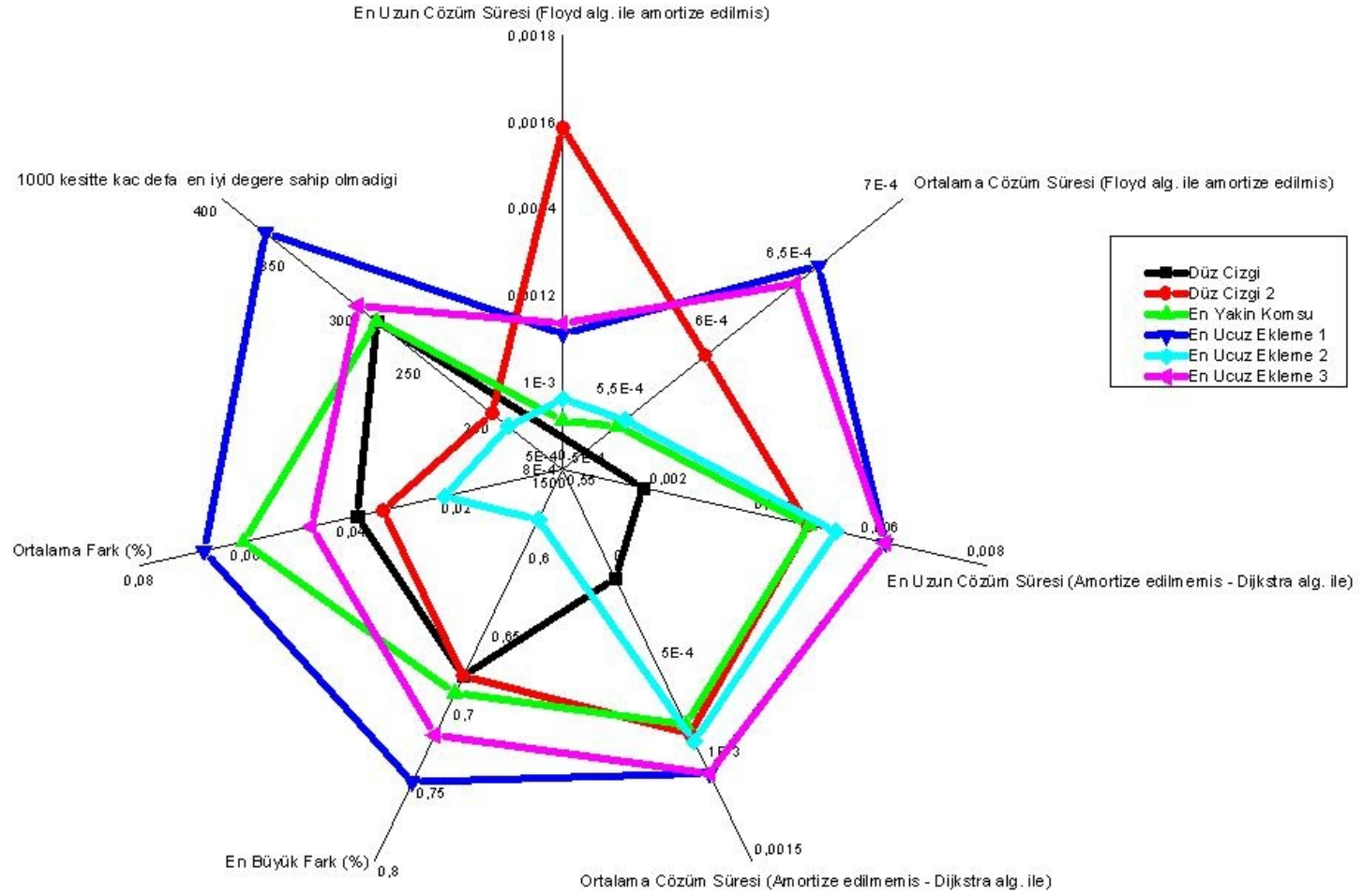
EK-C: Çalışma içerisinde kullanılan algoritmaların C kodları CD'de verilmiştir.

EK-D: Çalışmada kullanılan koşum sonuçları CD'de verilmiştir.

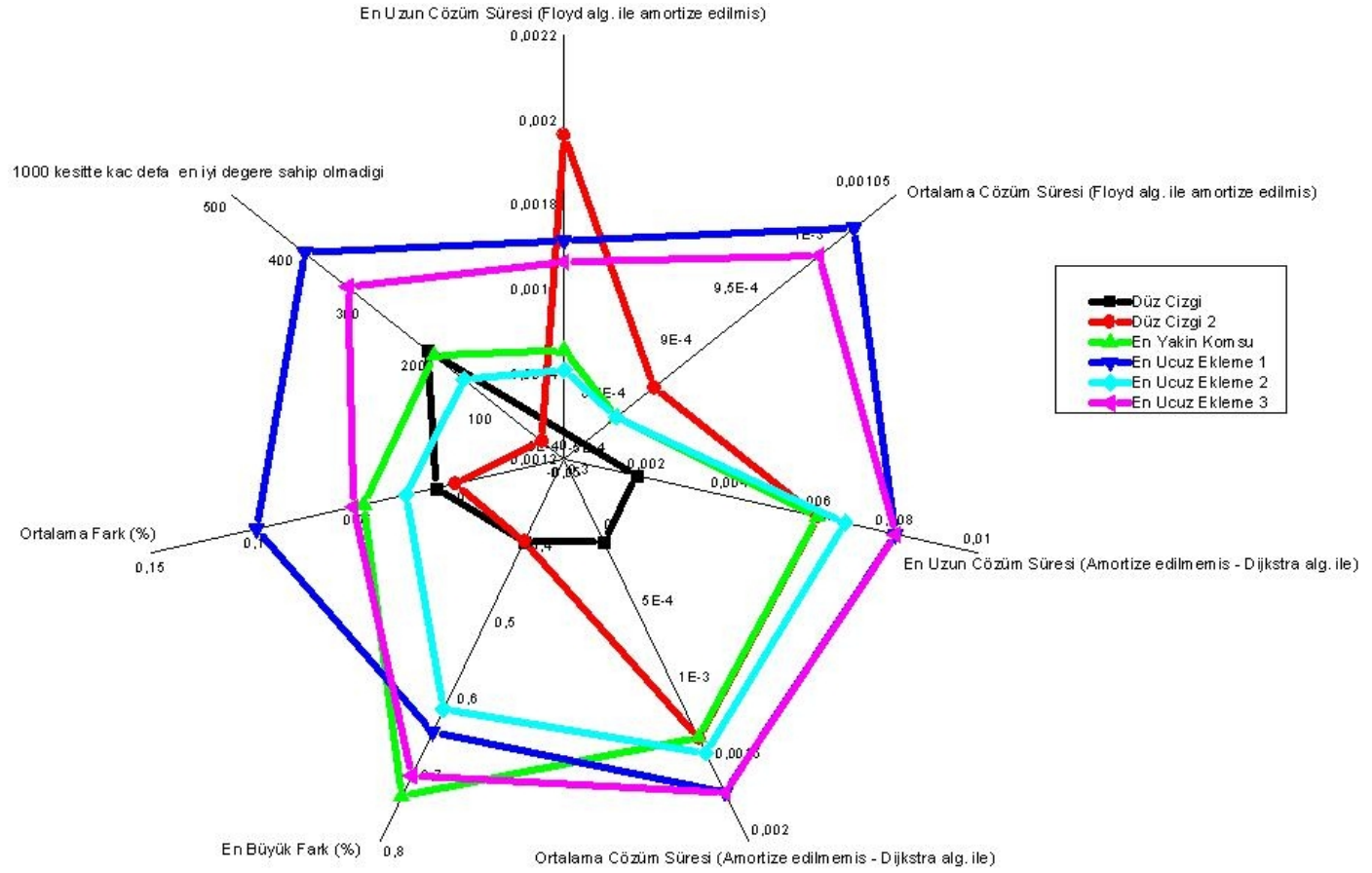
EK A1 : AK'nın 100 Döğümlü Olduđu 10 Adet Düşük Derinlik Yüksek Yoğunluklu Ağda Yöntemlerin Radar Çizelgesi ile Kıyaslanması



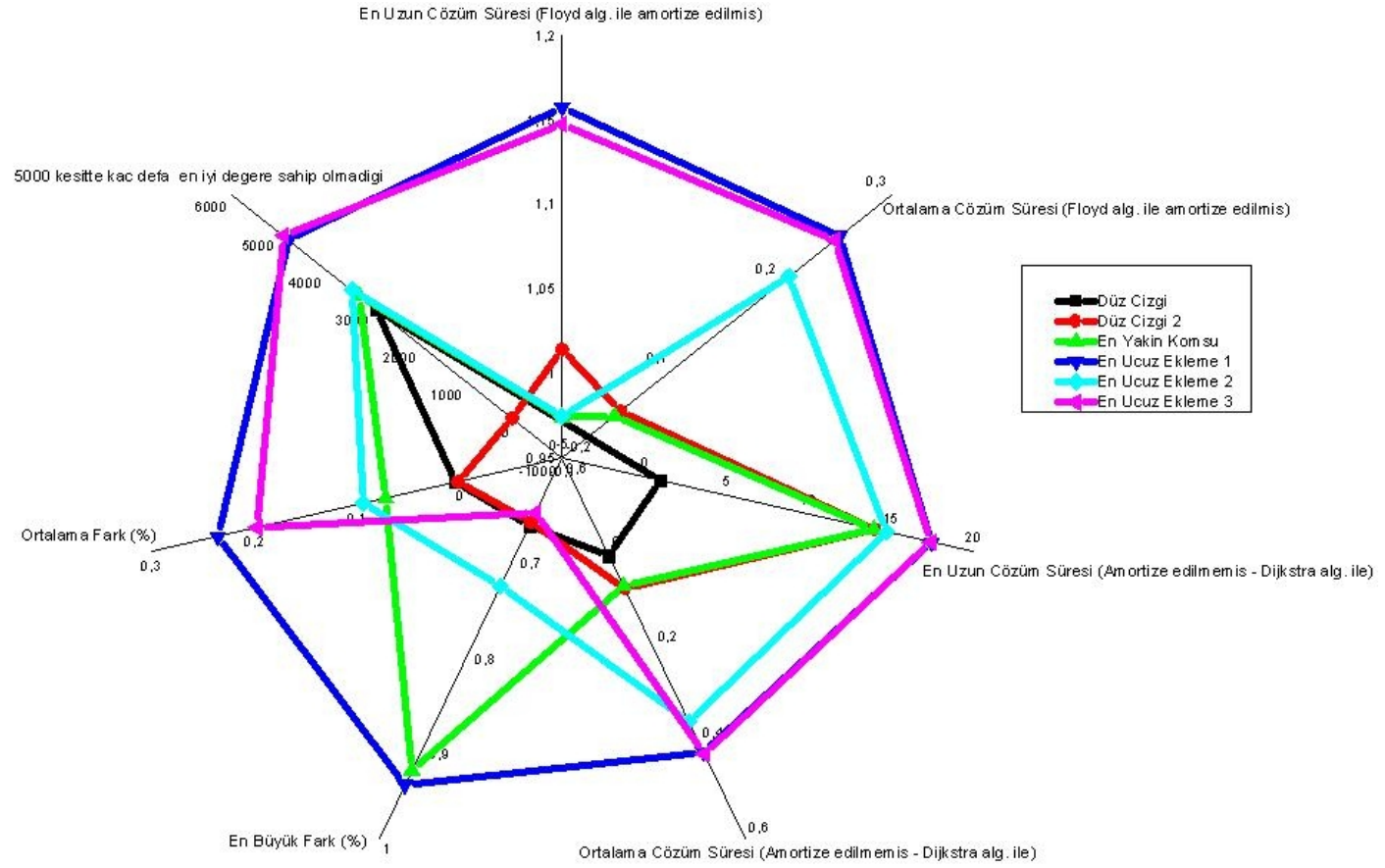
EK A2 : AK'nın 100 D ğ ml  Oldu u 10 Adet Y ksek Derinlik D   k Yo unluklu A da Y ntemlerin Radar  izelgesi ile Kiyaslanmas 



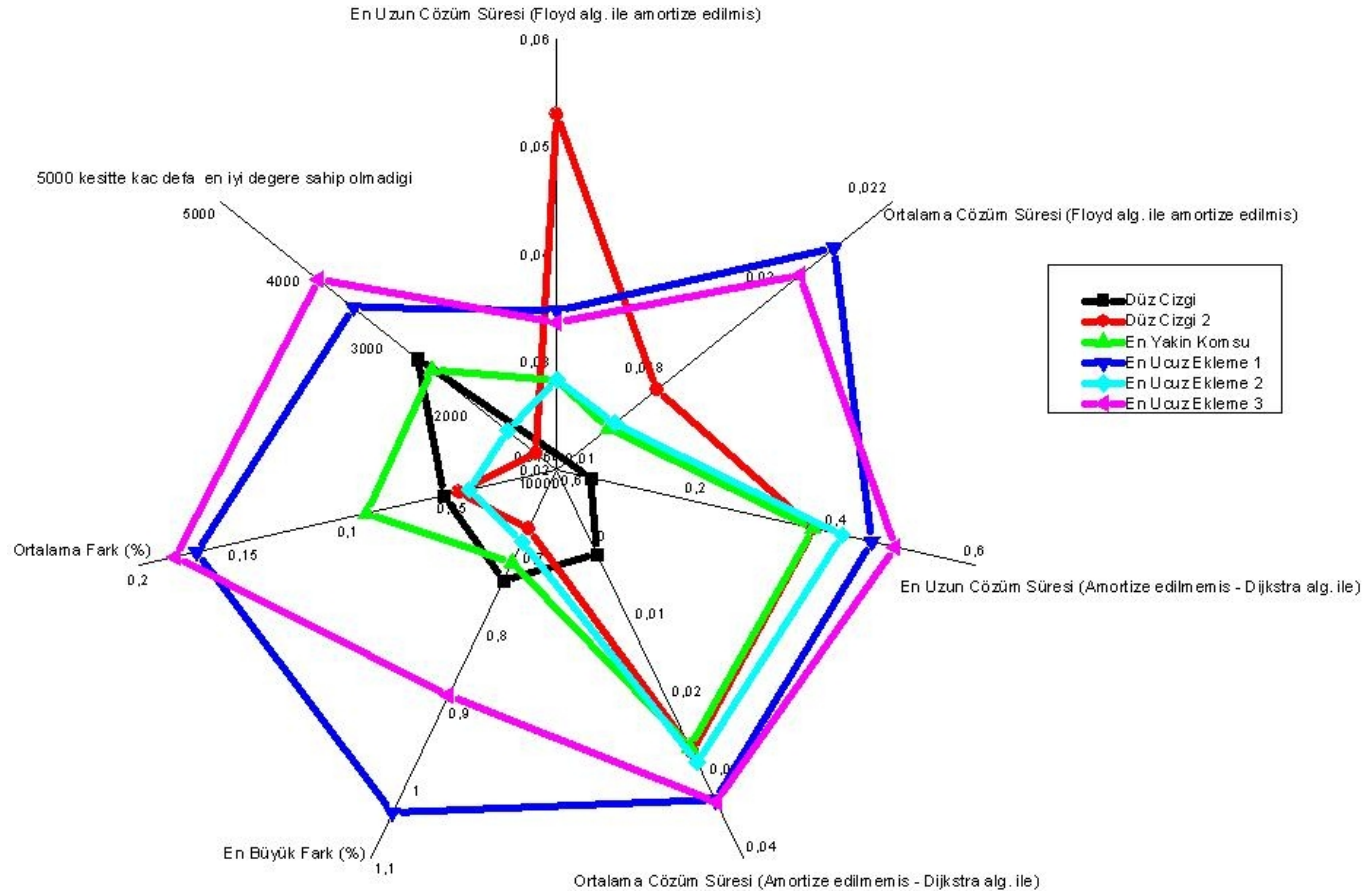
EK A3 : AK'nın 100 Düğümlü Olduğu 10 Adet Yüksek Derinlik Yüksek Yoğunluklu Ağda Yöntemlerin Radar Çizelgesi ile Kıyaslanması



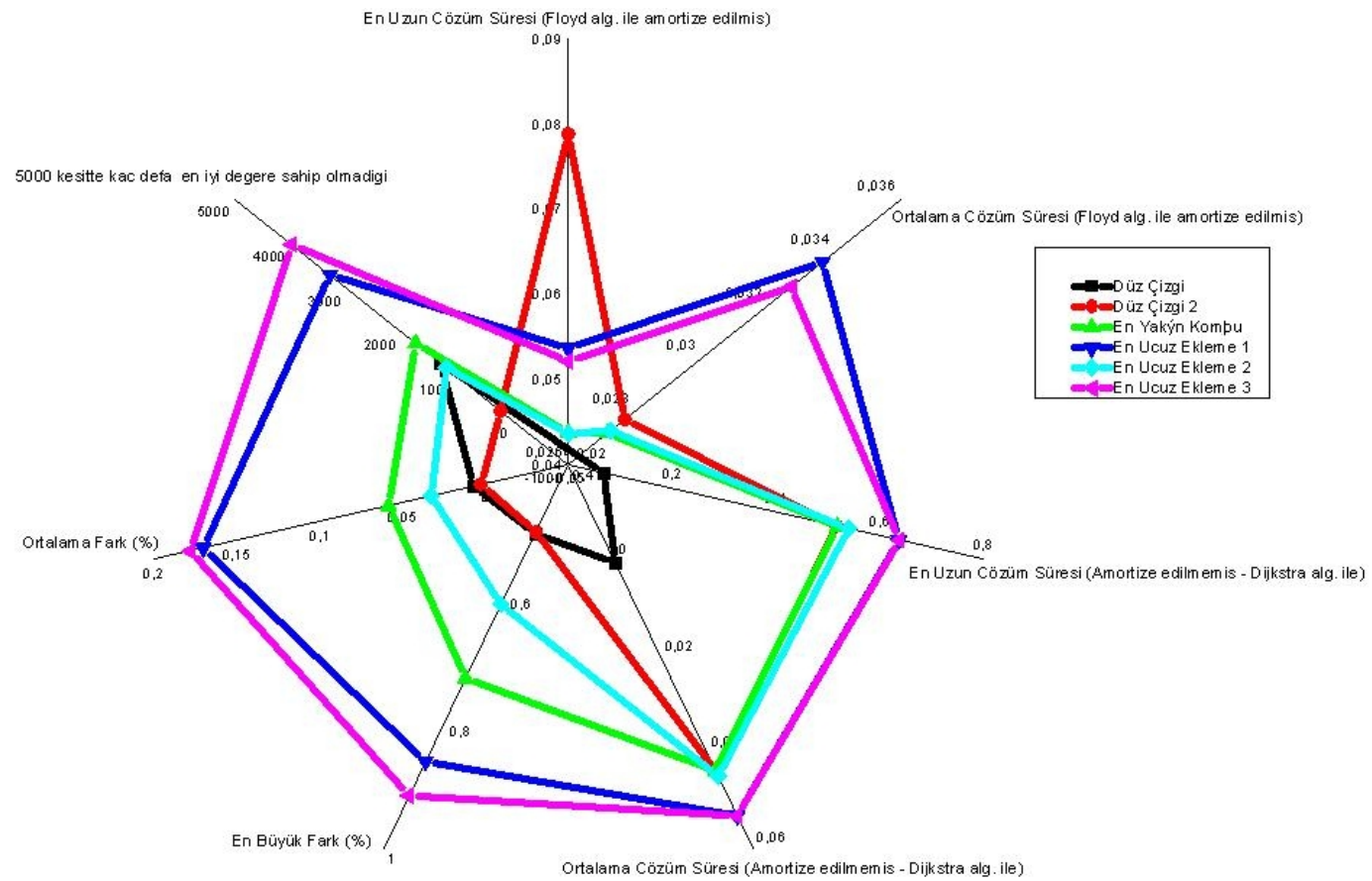
EK A4 : AK'nın 500 D    ml   Oldu   10 Adet D    k Derinlik Y  ksek Yo  unluklu A  da Y  ntemlerin Radar   izelgesi ile Kiyaslanmas  



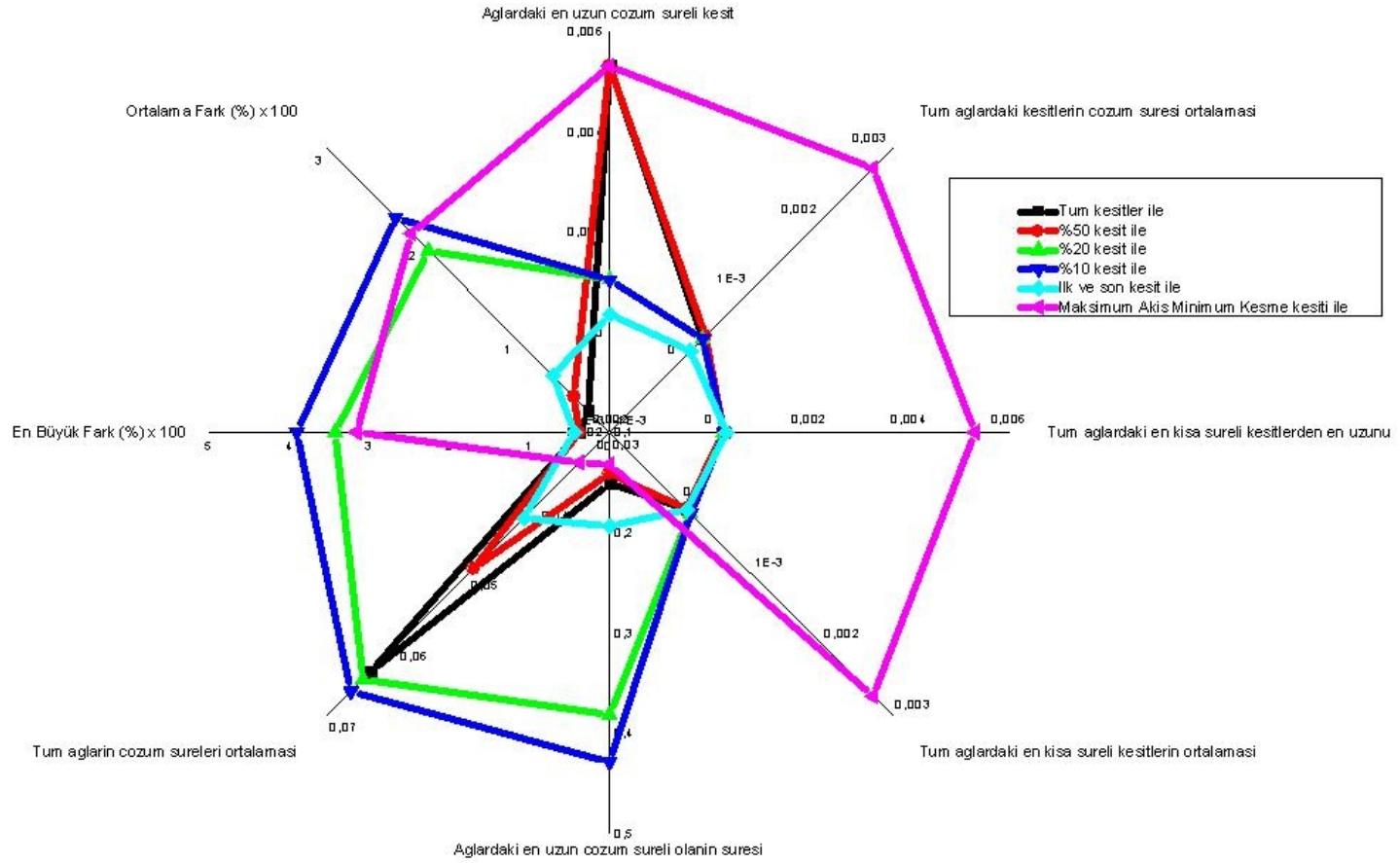
EK A5 : AK'nın 500 D ğ ml  Oldu u 10 Adet Y ksek Derinlik D   k Yo unluklu A da Y ntemlerin Radar  izelgesi ile Kiyaslanması



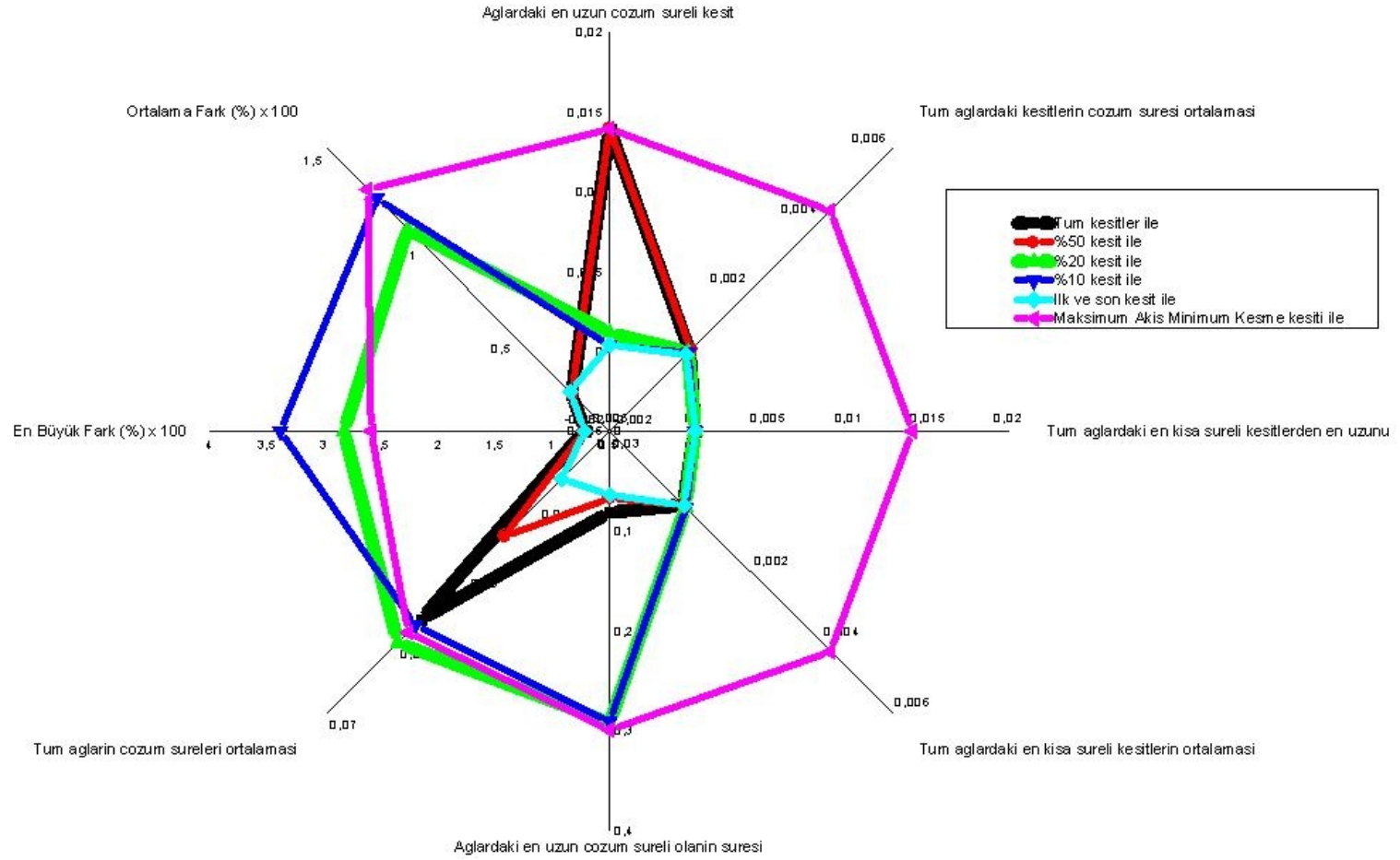
EK A6 : AK'nın 500 Dügümlü Olduğu 10 Adet Yüksek Derinlik Yüksek Yoğunluklu Ağda Yöntemlerin Radar Çizelgesi ile Kıyaslanması



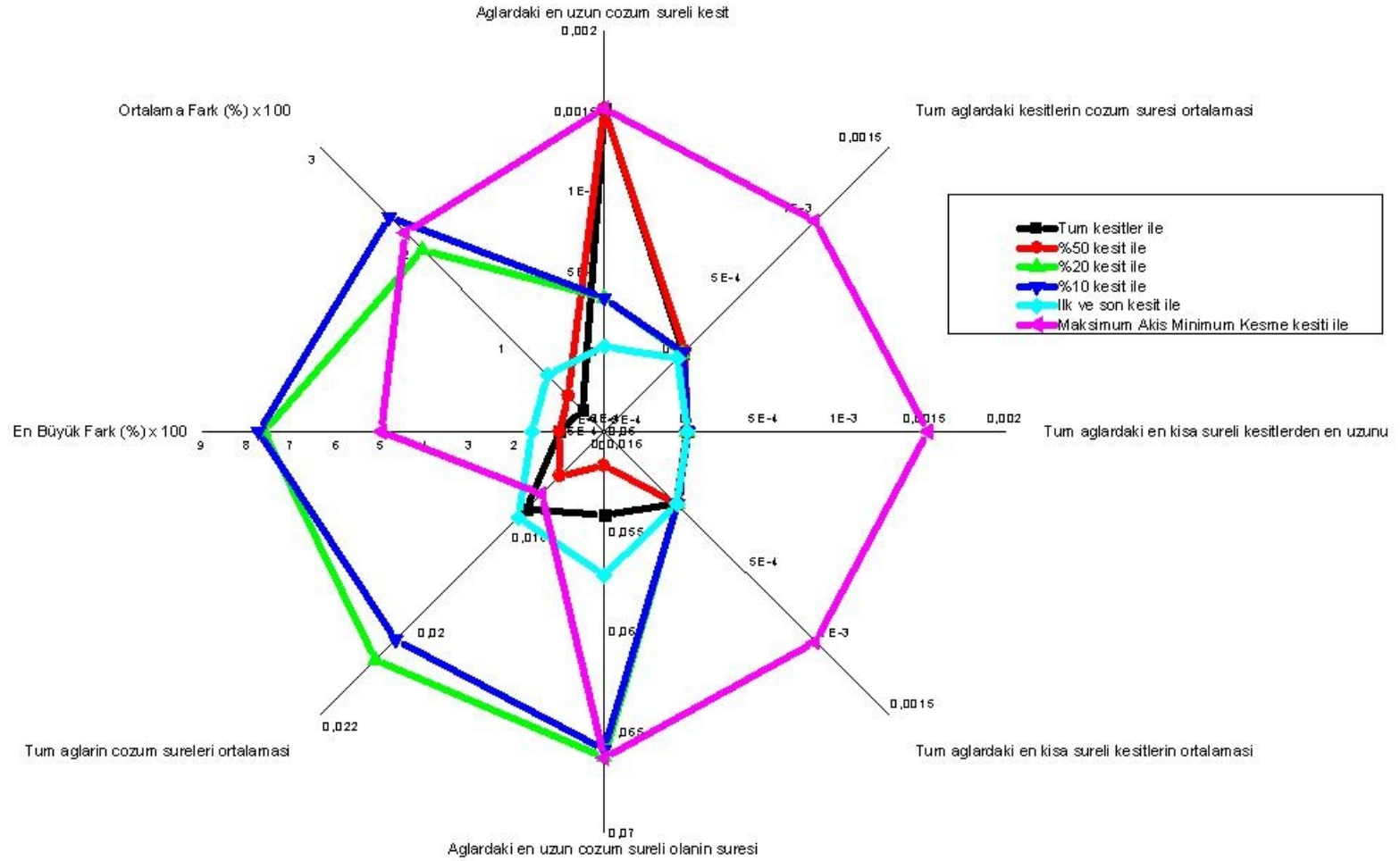
EK B1: 100 düğümlü düşük derinlikli düşük yoğunluklu AK ağında düz çizgi yöntemi ile en uygun kesit sayısının incelenmesi



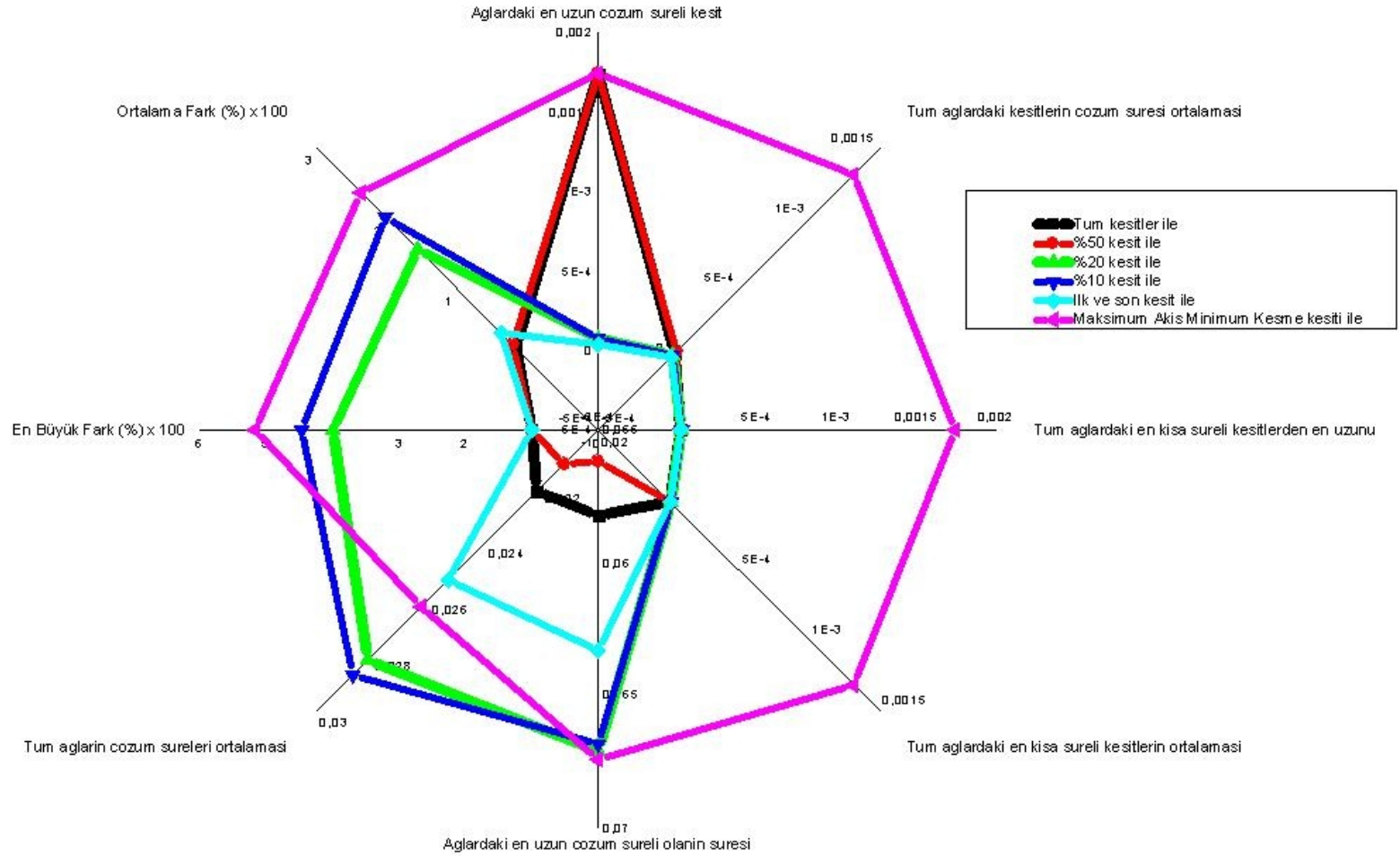
EK B2: 100 Düğümlü Düşük Derinlikli Yüksek Yoğunluklu AK Ağında Düz Çizgi Yöntemi ile En Uygun Kesit Sayısının İncelenmesi



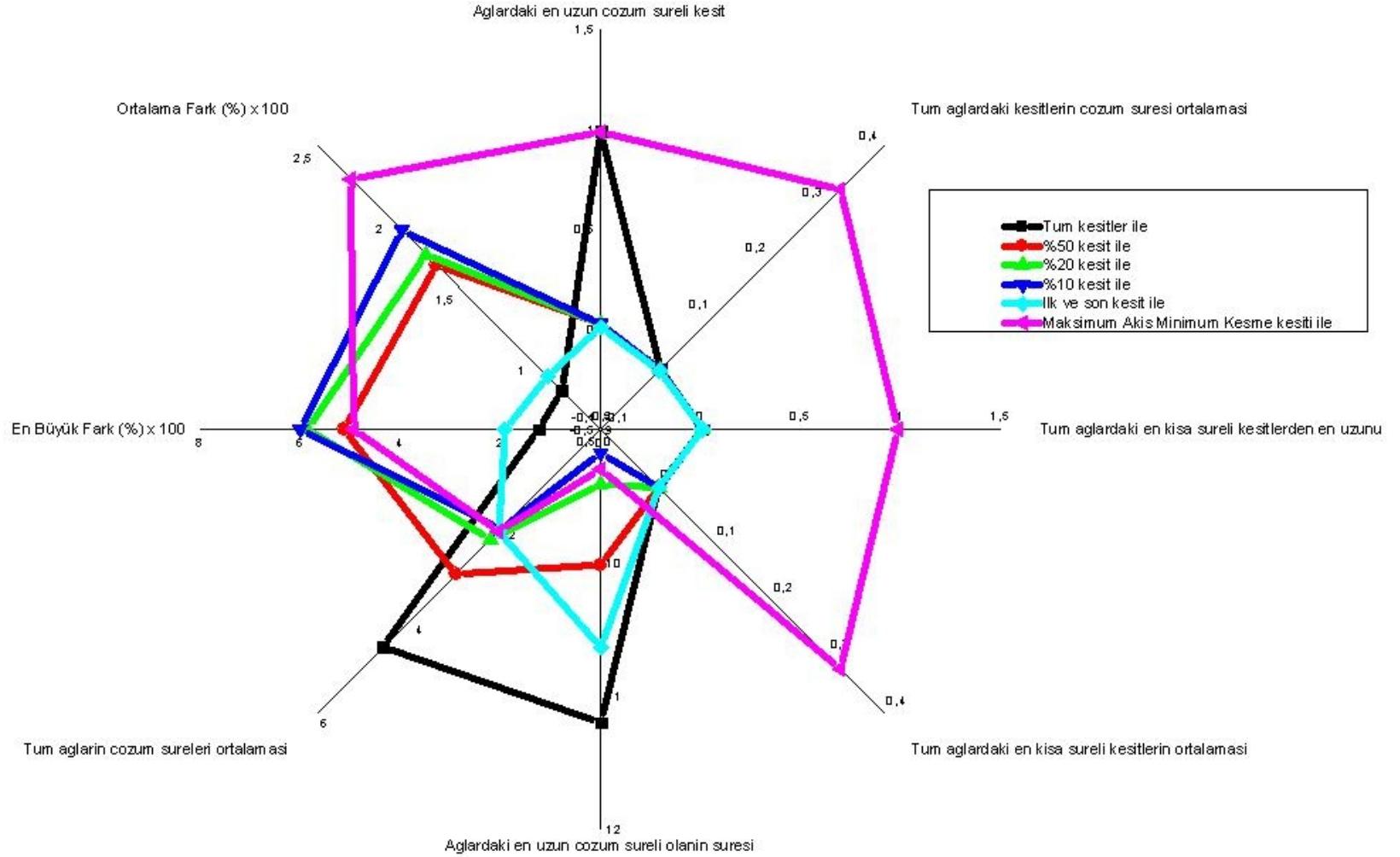
EK B3: 100 Düğümlü Yüksek Derinlikli Düşük Yoğunluklu AK Ağında Düz Çizgi Yöntemi ile En Uygun Kesit Sayısının İncelenmesi



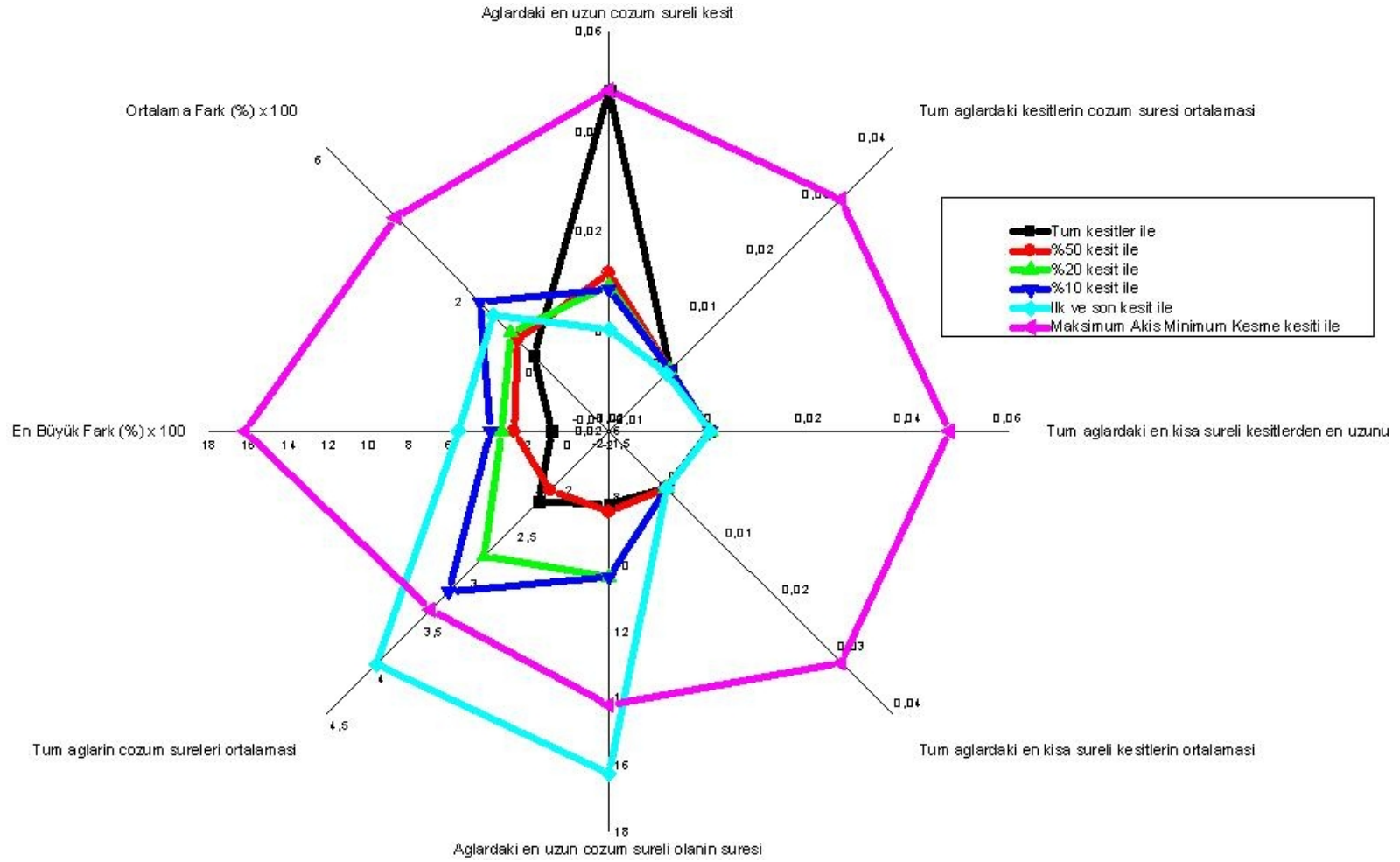
EK B4: 100 Düğümlü Yüksek Derinlikli Yüksek Yoğunluklu AK Ağında Düz Çizgi Yöntemi ile En Uygun Kesit Sayısının İncelenmesi



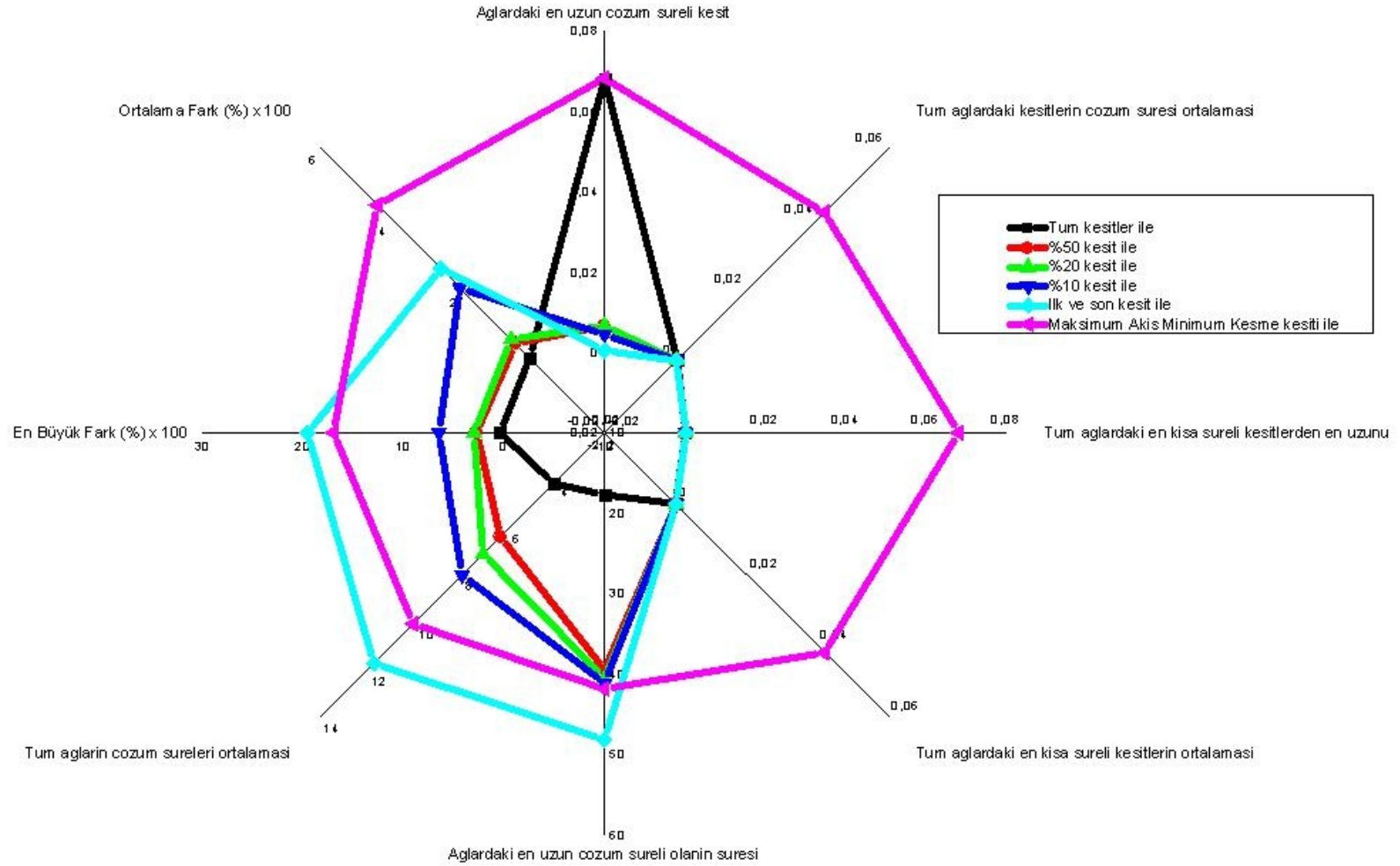
EK B5: 500 Düğümlü Düşük Derinlikli Yüksek Yoğunluklu AK Ağında Düz Çizgi Yöntemi ile En Uygun Kesit Sayısının İncelenmesi



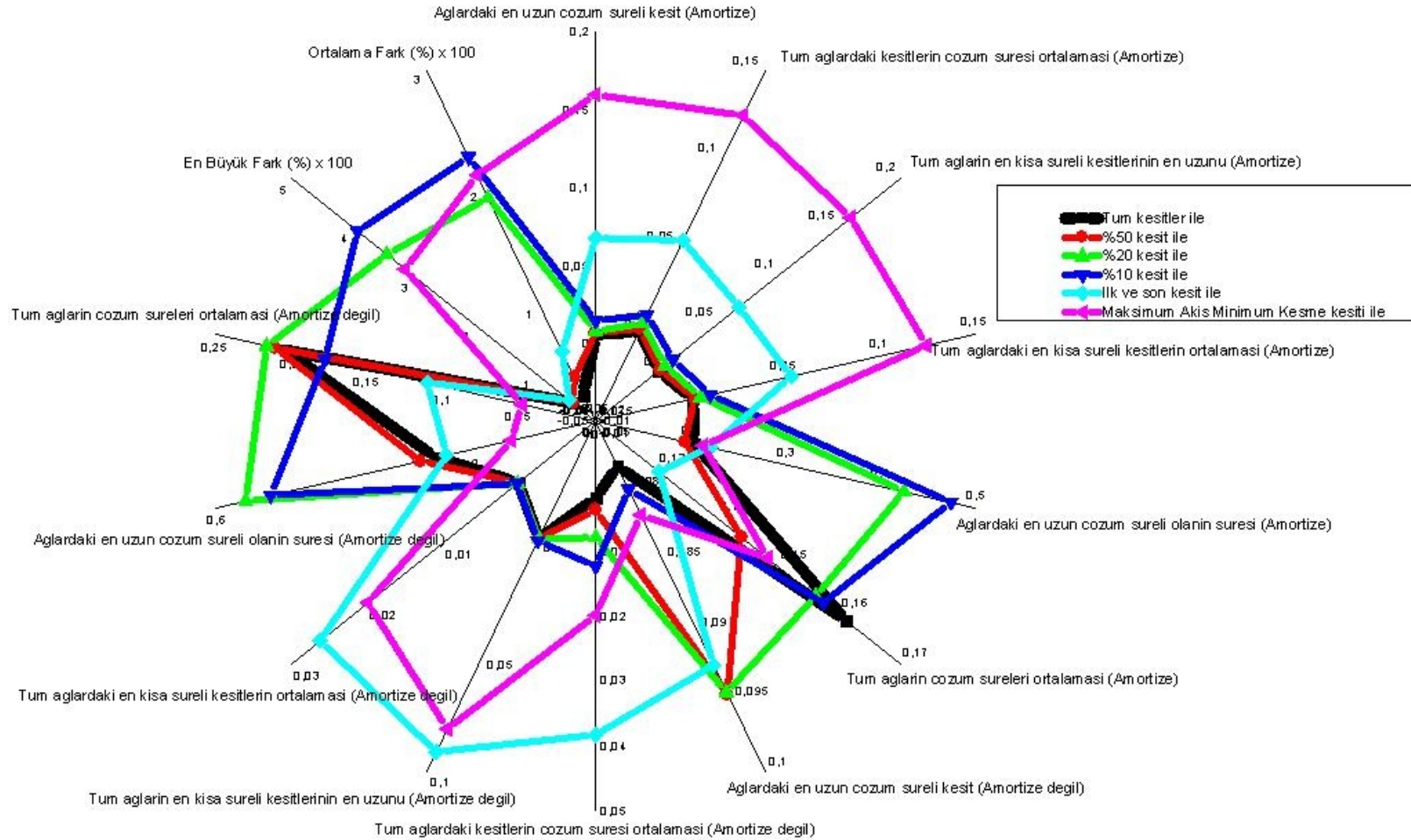
EK B6: 500 Düğümlü Yüksek Derinlikli Düşük Yoğunluklu AK Ağında Düz Çizgi Yöntemi ile En Uygun Kesit Sayısının İncelenmesi



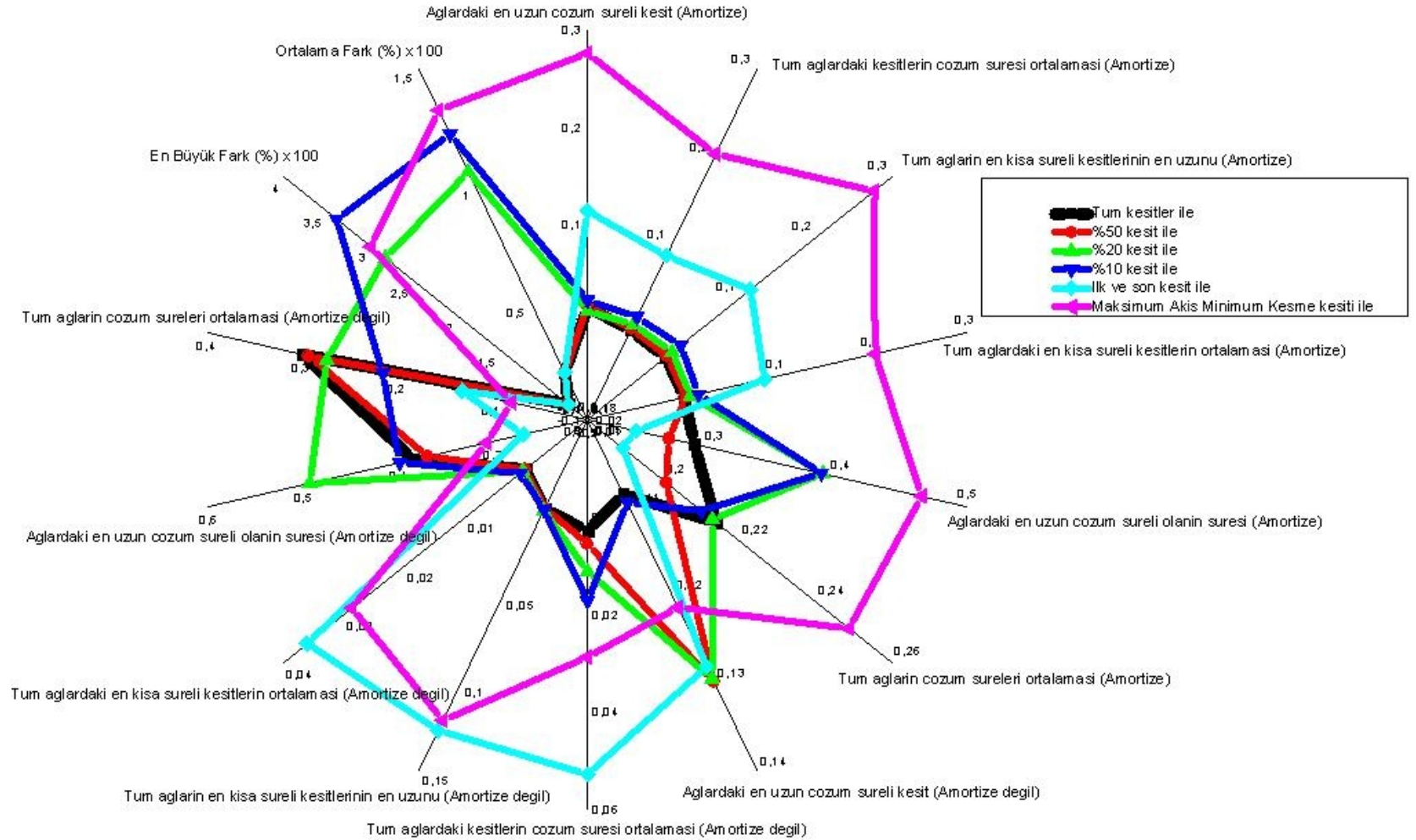
EK B7: 500 Düğümlü Yüksek Derinlikli Yüksek Yoğunluklu AK Ağında Düz Çizgi Yöntemi ile En Uygun Kesit Sayısının İncelenmesi



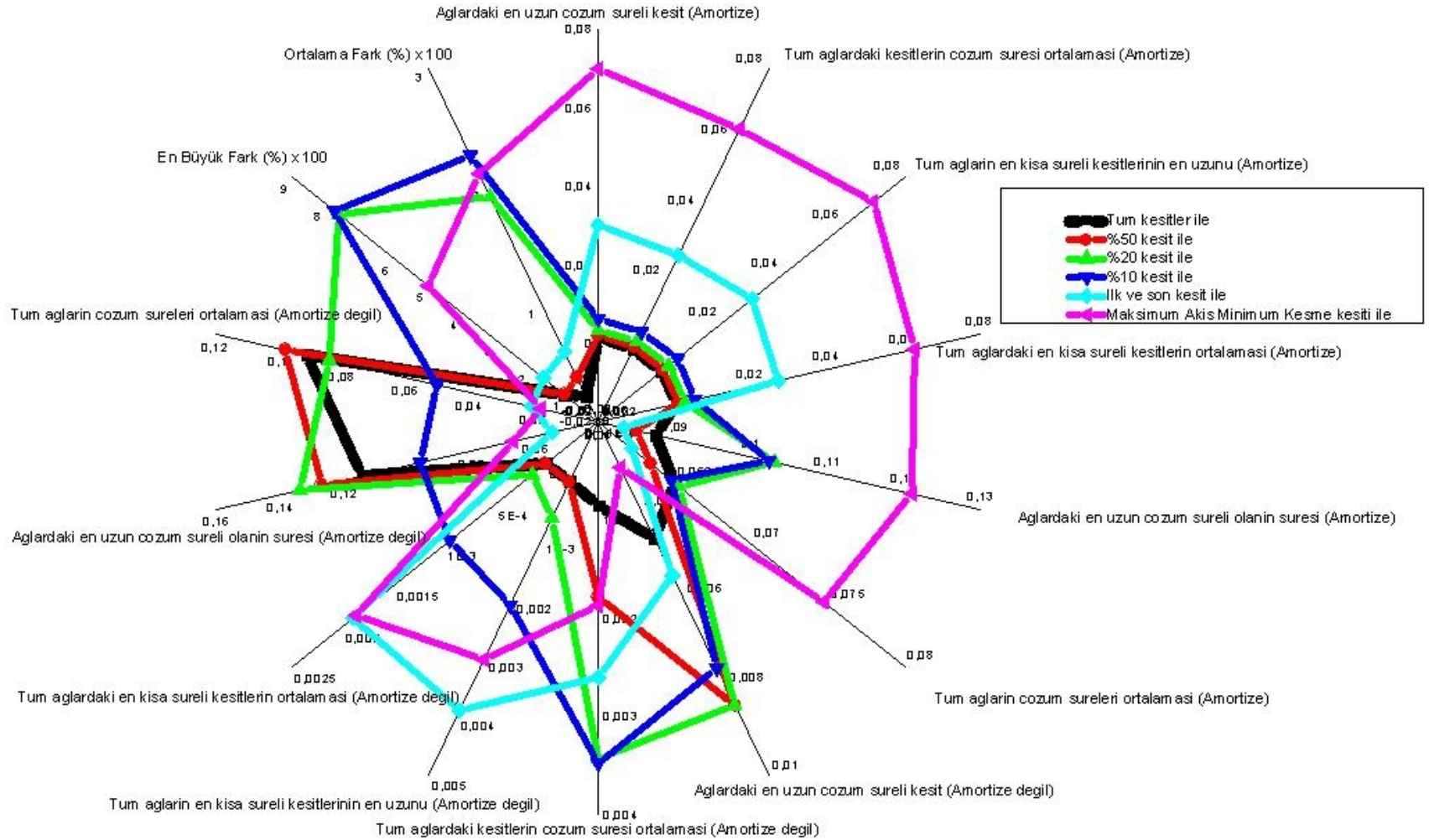
EK B8: 100 Düğümlü Düşük Derinlikli Düşük Yoğunluklu AK Ağında Düz Çizgi 2 Yöntemi ile En Uygun Kesit Sayısının İncelenmesi



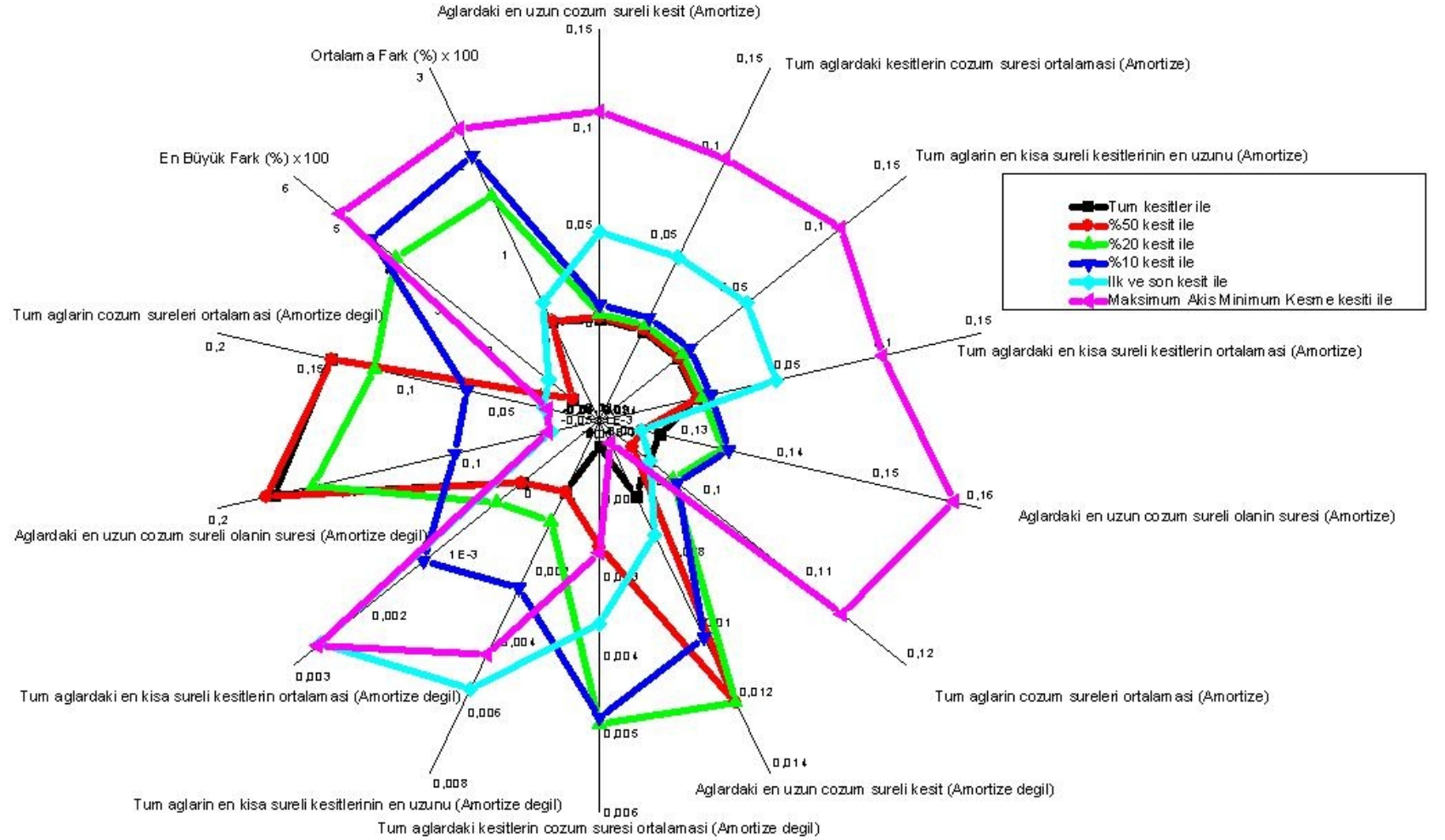
EK B9: 100 Düğümlü Düşük Derinlikli Yüksek Yoğunluklu AK Ağında Düz Çizgi 2 Yöntemi ile En Uygun Kesit Sayısının İncelenmesi



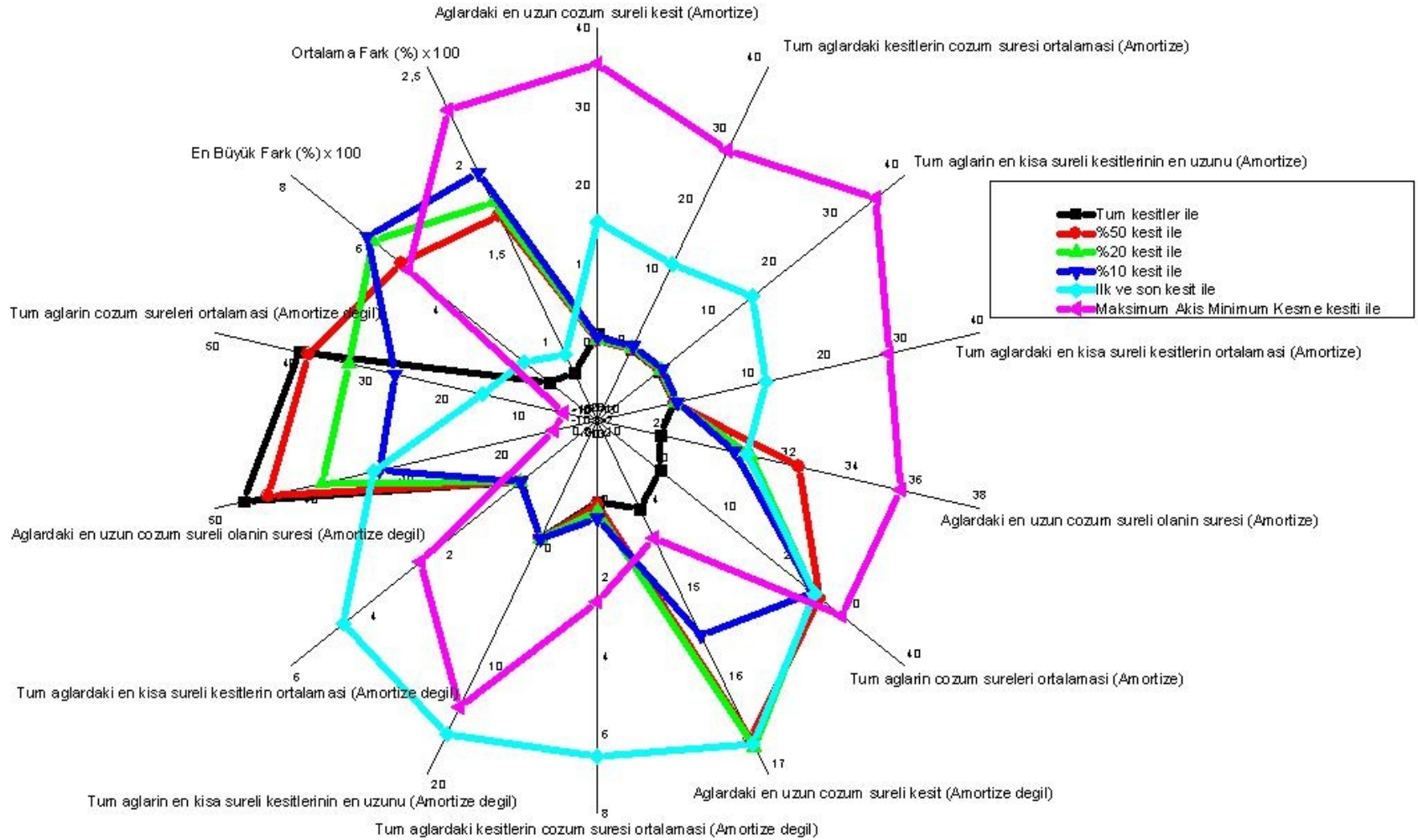
EK B10: 100 Düğümlü Yüksek Derinlikli Düşük Yoğunluklu AK Ağında Düz Çizgi 2 Yöntemi ile En Uygun Kesit Sayısının İncelenmesi



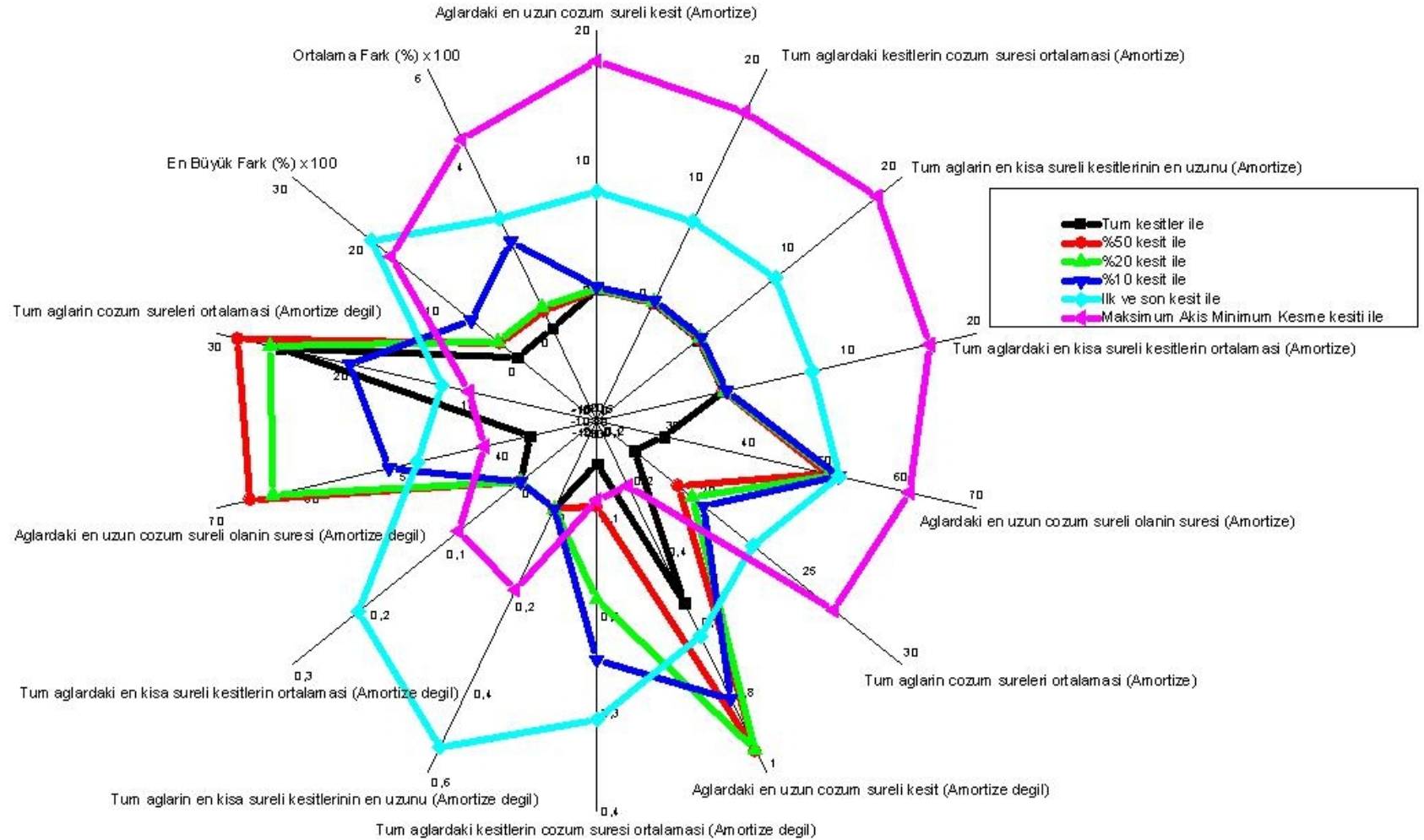
EK B11: 100 Düğümlü Yüksek Derinlikli Yüksek Yoğunluklu AK Ağında Düz Çizgi 2 Yöntemi ile En Uygun Kesit Sayısının İncelenmesi



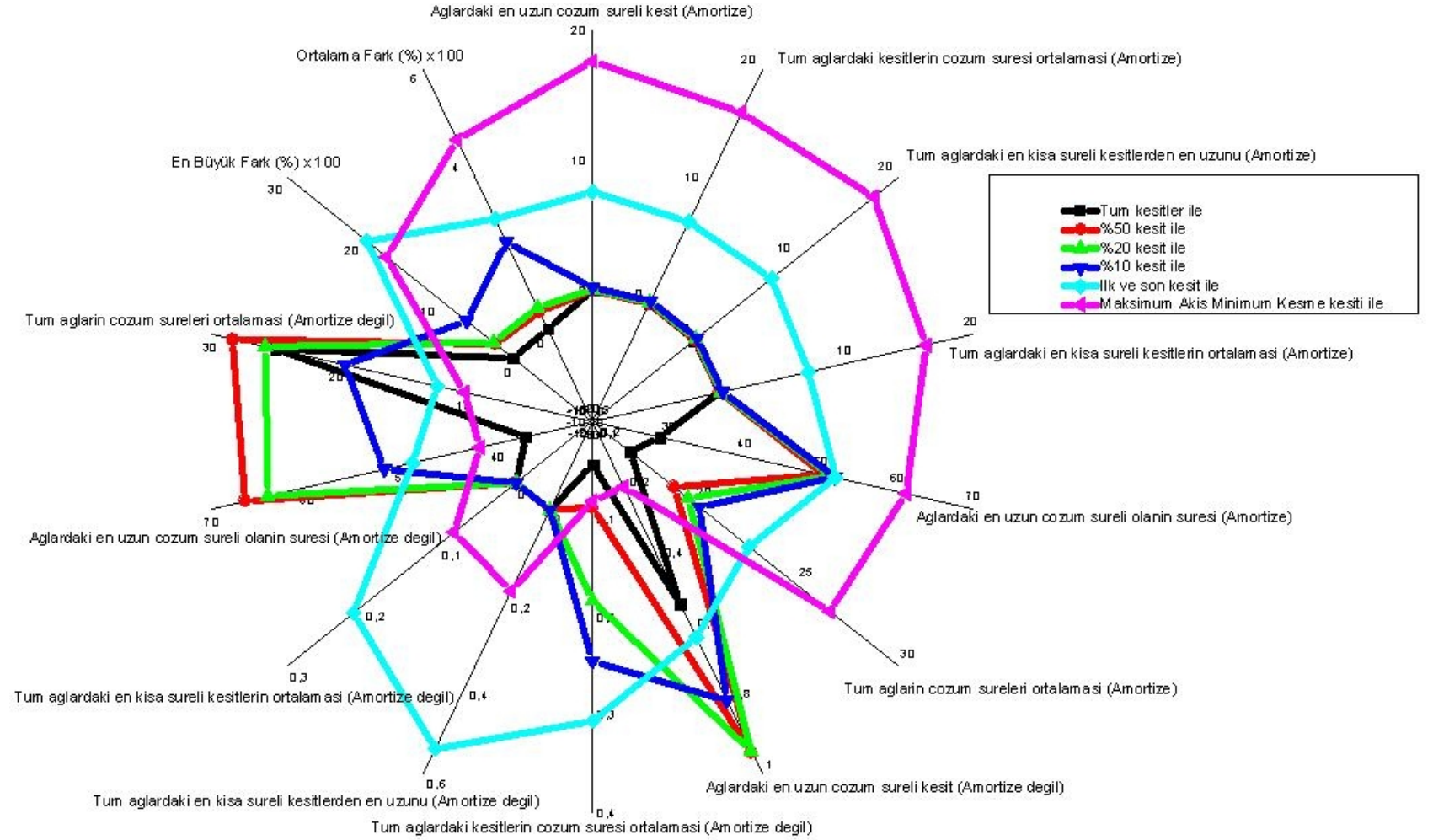
EK B12: 500 Düğümlü Düşük Derinlikli Yüksek Yoğunluklu AK Ağında Düz Çizgi 2 Yöntemi ile En Uygun Kesit Sayısının İncelenmesi



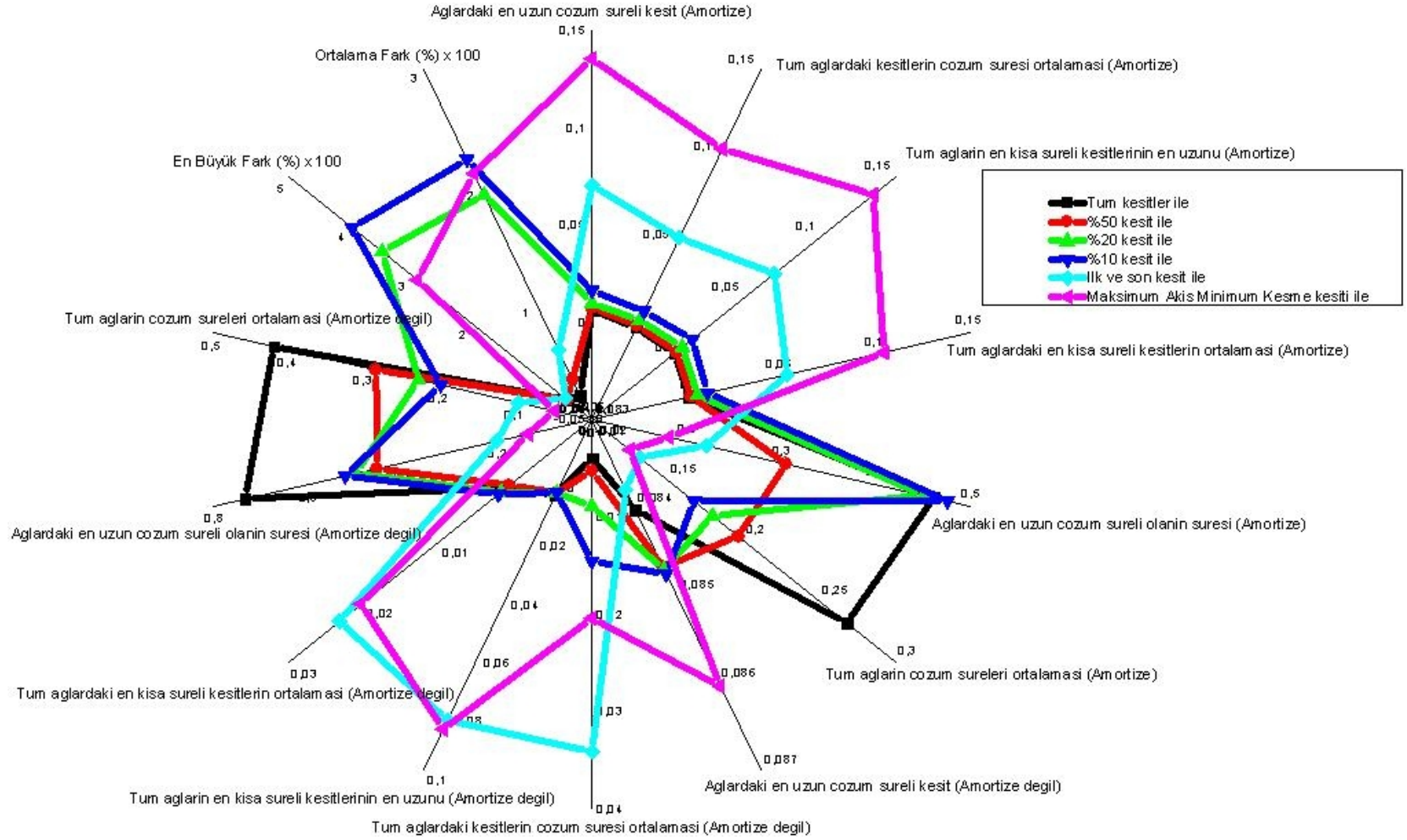
EK B13: 500 Düğümlü Yüksek Derinlikli Düşük Yoğunluklu AK Ağında Düz Çizgi 2 Yöntemi ile En Uygun Kesit Sayısının İncelenmesi



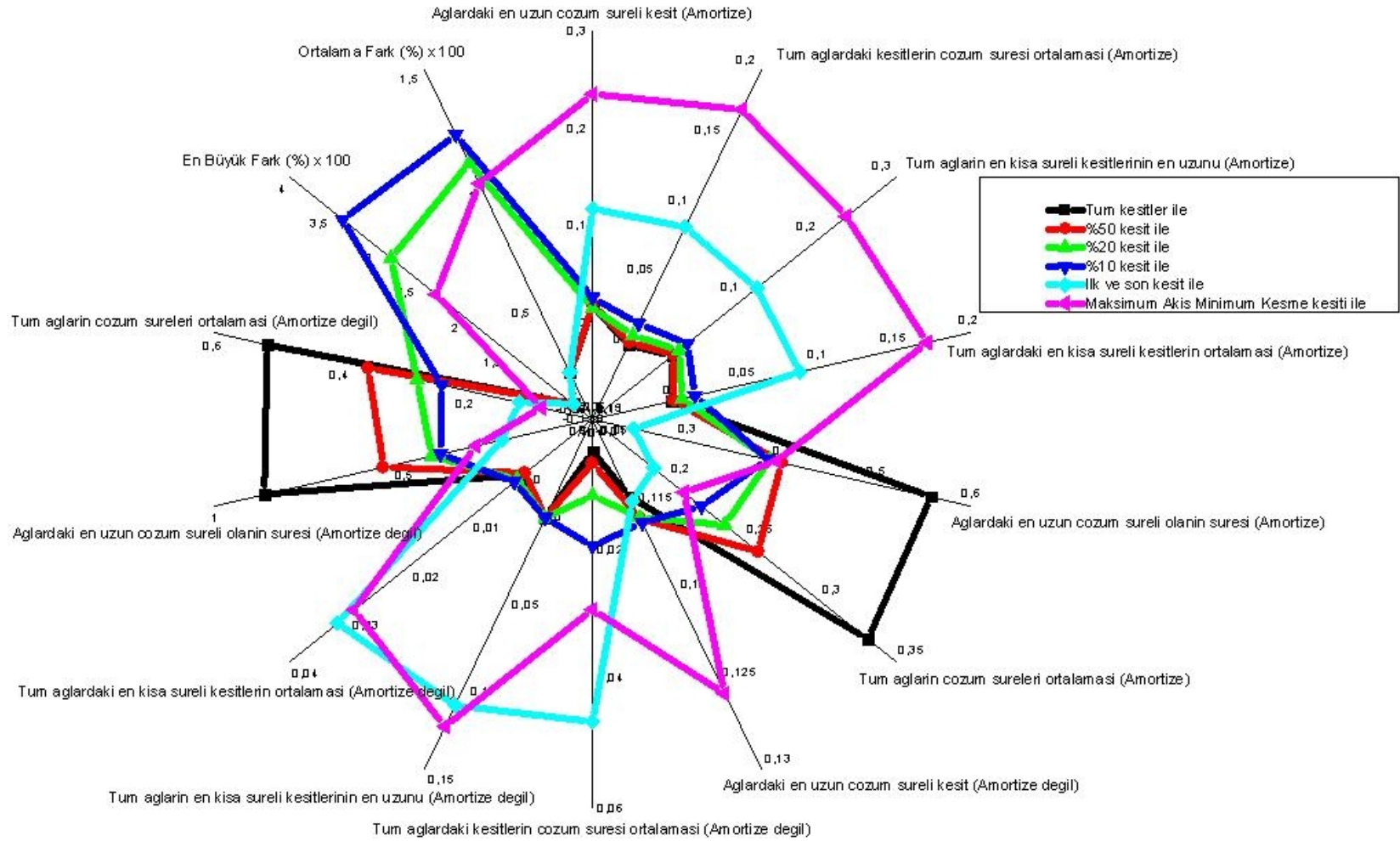
EK B14: 500 Düğümlü Yüksek Derinlikli Yüksek Yoğunluklu AK Ağında Düz Çizgi 2 Yöntemi ile En Uygun Kesit Sayısının İncelenmesi



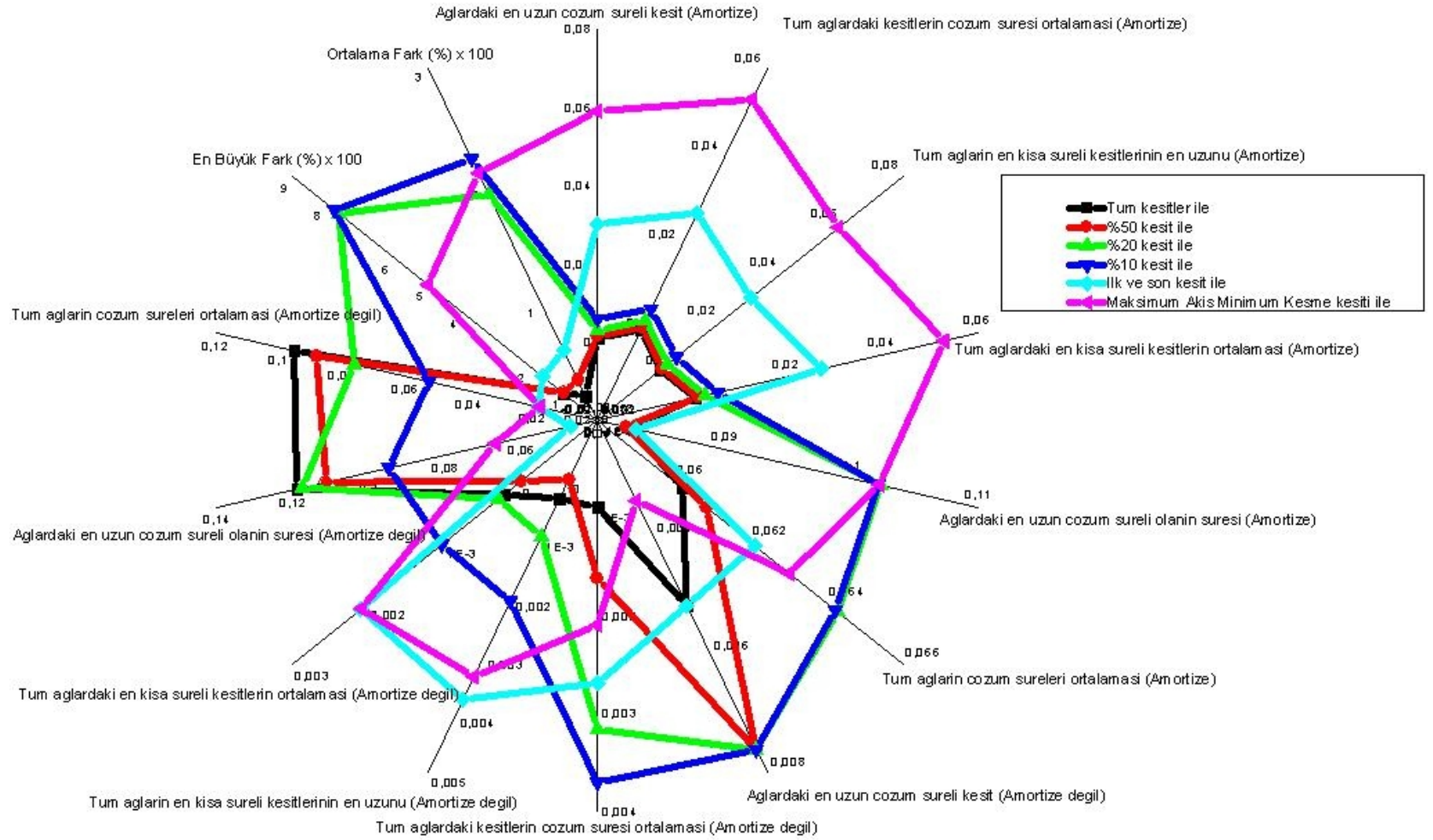
EK B15: 100 Düğümlü Düşük Derinlikli Düşük Yoğunluklu AK Ağında En Ucuz Ekleme 2 Yöntemiyle En Uygun Kesit Sayısının İncelenmesi



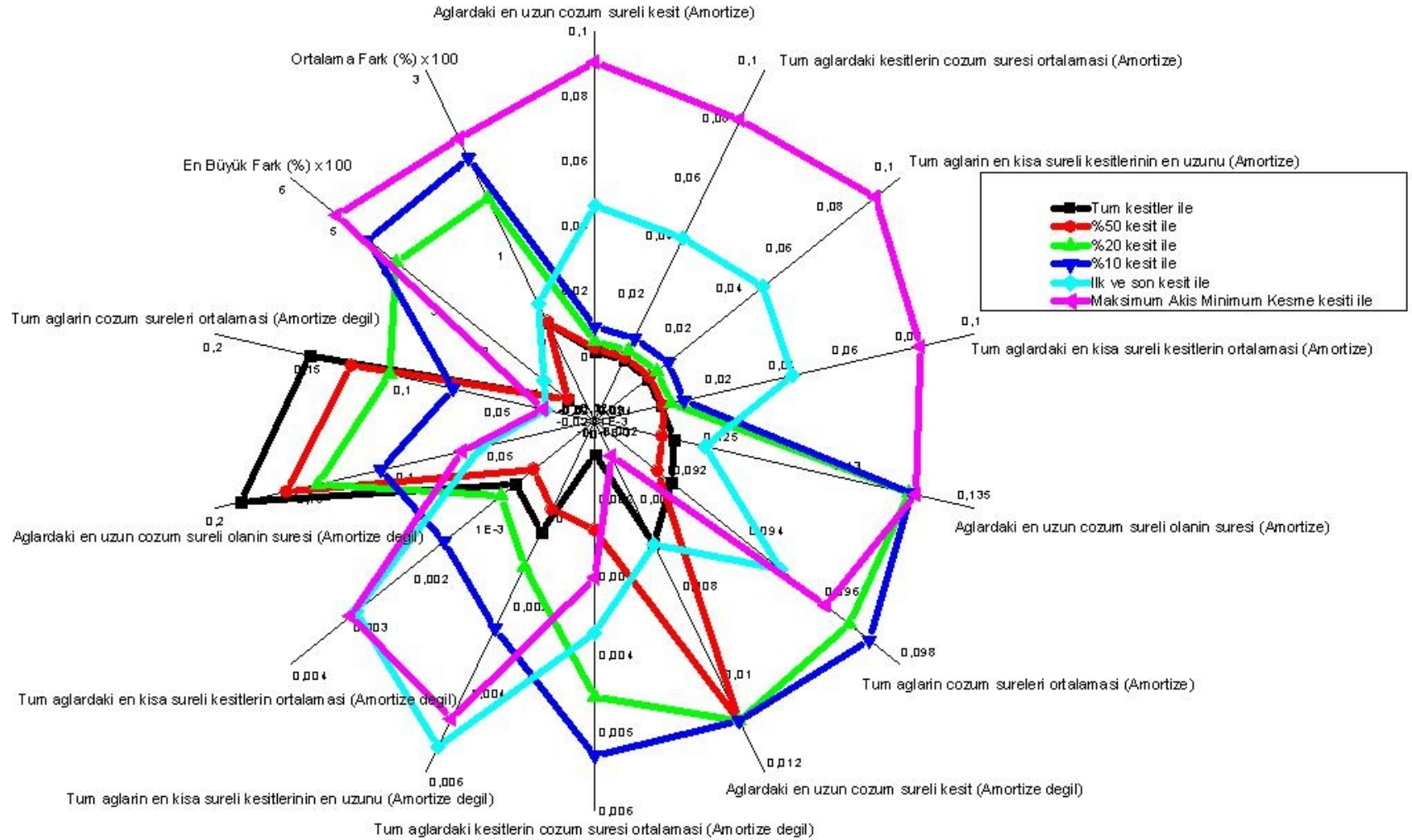
EK B16: 100 Düğümlü Düşük Derinlikli Yüksek Yoğunluklu AK Ağında En Ucuz Ekleme 2 Yöntemiyle En Uygun Kesit Sayısının İncelenmesi



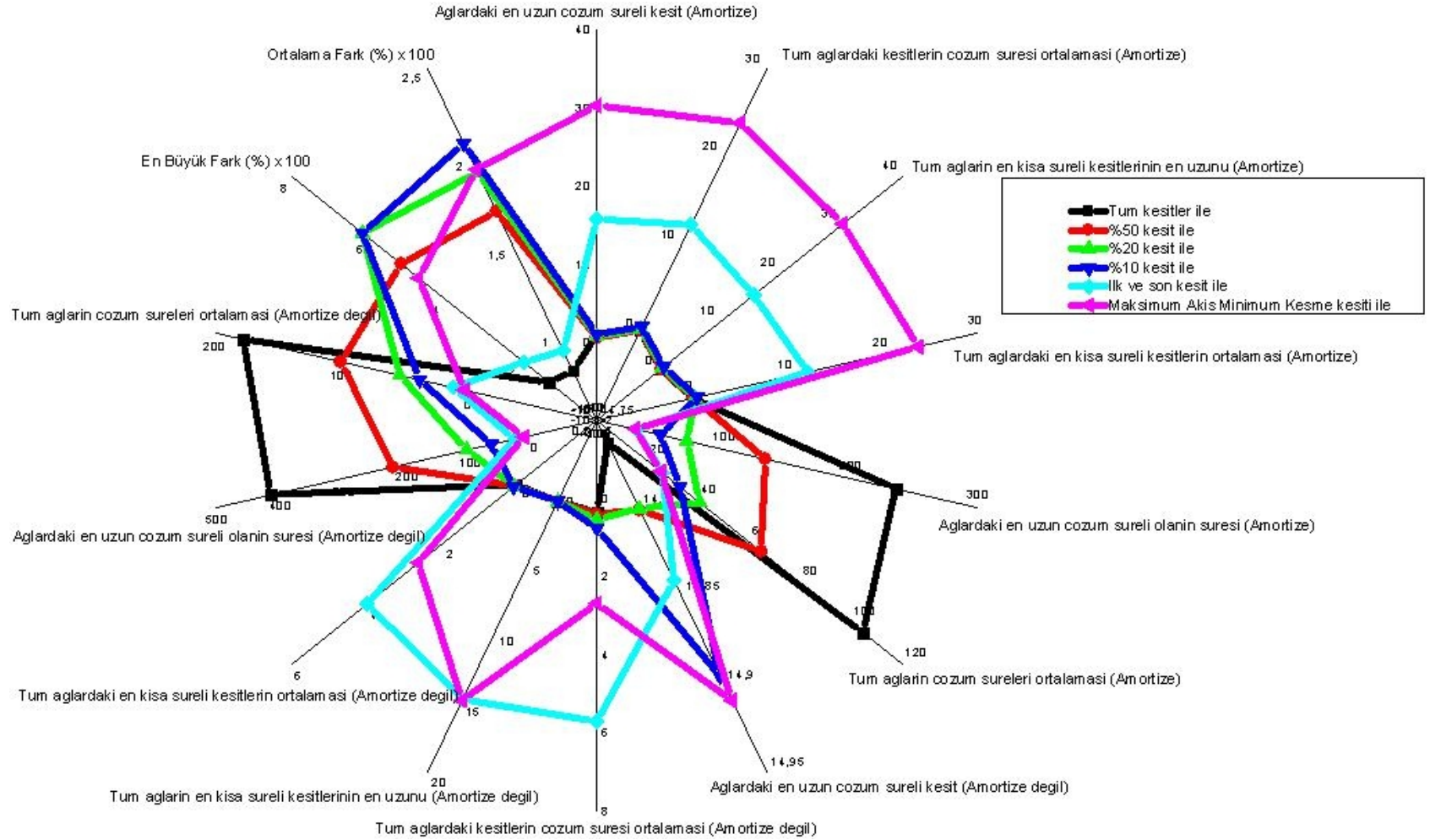
EK B17: 100 Düğümlü Yüksek Derinlikli Düşük Yoğunluklu AK Ağında En Ucuz Ekleme 2 Yöntemiyle En Uygun Kesit Sayısının İncelenmesi



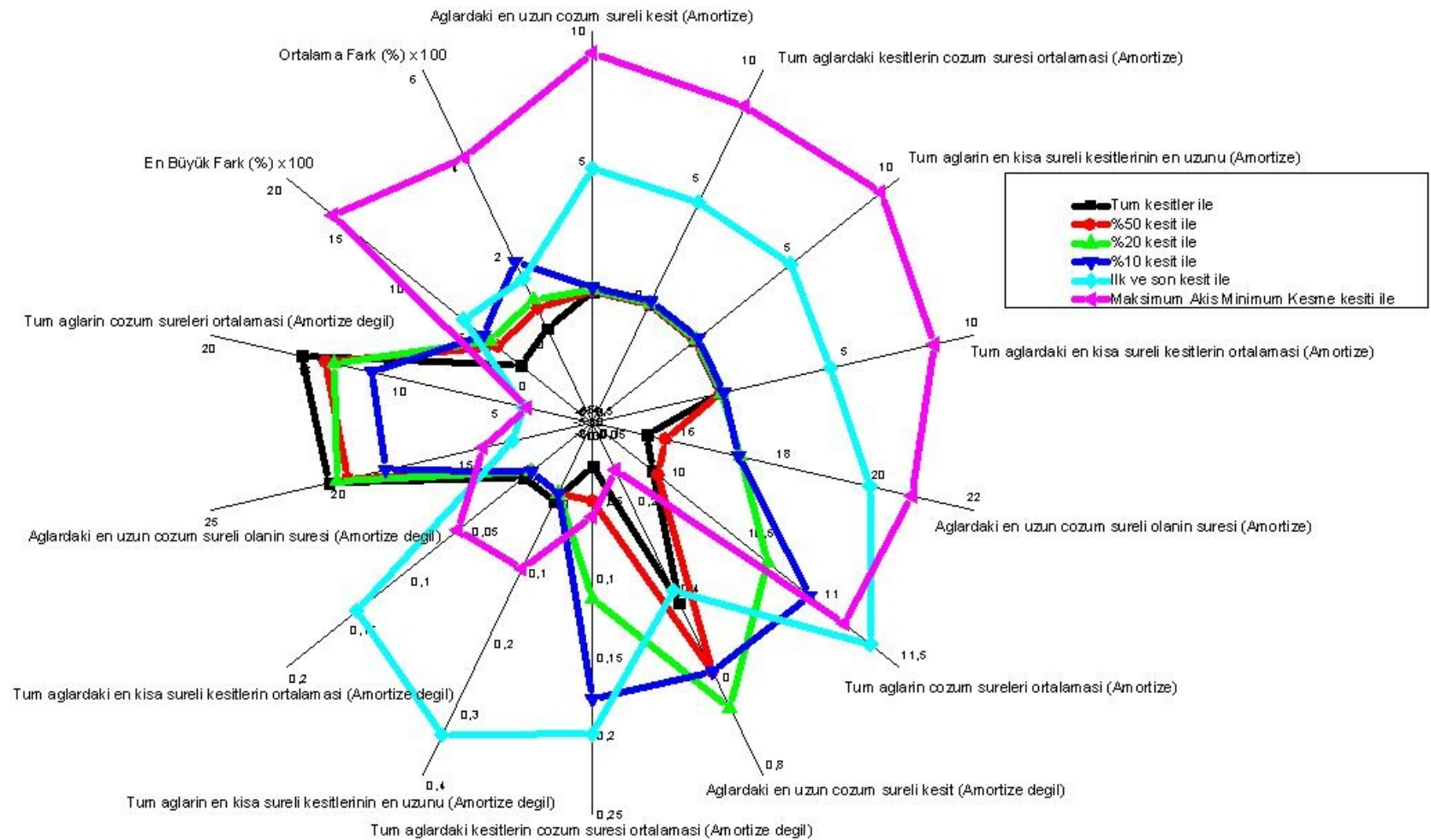
EK B18: 100 Düğümlü Yüksek Derinlikli Yüksek Yoğunluklu AK Ağında En Ucuz Ekleme 2 Yöntemiyle En Uygun Kesit Sayısının İncelenmesi



EK B19: 500 Düğümlü Düşük Derinlikli Yüksek Yoğunluklu AK Ağında En Ucuz Ekleme 2 Yöntemiyle En Uygun Kesit Sayısının İncelenmesi



EK B20: 500 Dügümlü Yüksek Derinlikli Düşük Yoğunluklu AK Ağında En Ucuz Ekleme 2 Yöntemiyle En Uygun Kesit Sayısının İncelenmesi



EK B21: 500 Düğümlü Yüksek Derinlikli Yüksek Yoğunluklu AK Ağında En Ucuz Ekleme 2 Yöntemiyle En Uygun Kesit Sayısının İncelenmesi

