

**İSTANBUL TEKNİK ÜNİVERSİTESİ ★ BİLİŞİM ENSTİTÜSÜ**

**DUYARGA AĞLARI İÇİN KARINCA KOLONİSİ  
TABANLI BİR YÖNLENDİRME ALGORİTMASI  
ÇÖZÜMLEMESİ VE TASARIMI**

**YÜKSEK LİSANS TEZİ  
Müh. Deniz DEMİRAY**

**Anabilim Dalı : İLERİ TEKNOLOJİLER**

**Programı : BİLGİSAYAR BİLİMLERİ**

**OCAK 2008**

**DUYARGA AĞLARI İÇİN KARINCA KOLONİSİ  
TABANLI BİR YÖNLENDİRME ALGORİTMASI  
ÇÖZÜMLEMESİ VE TASARIMI**

**YÜKSEK LİSANS TEZİ  
Müh. Deniz DEMİRAY  
704051002**

**Tezin Enstitüye Verildiği Tarih : 10 Ocak 2008  
Tezin Savunulduğu Tarih : 29 Ocak 2008**

**Tez Danışmanı : Yrd.Doç.Dr. Deniz Turgay ALTILAR**  
**Diğer Jüri Üyeleri : Prof.Dr. İbrahim EKSİN (İ.T.Ü.)**  
**Prof.Dr. Sema OKTUĞ (İ.T.Ü.)**

*D. Altılar*  
*İbrahim Eksin*  
*Sema Oktug*

**OCAK 2008**

## ÖNSÖZ

Sosyal böcek gruplarının ve özellikle karıncaların gösterdikleri davranış modelleri, karmaşık problemlerin çözümünü sağlayan birçok yöntemin esin kaynağı olmuştur. Basit davranışlar sergileyen çok sayıda bireyden oluşan gruplar, herhangi bir yönetici birey olmaksızın karmaşık davranışlar sergilemektedir. Karıncaların yiyecekleri yuvalarına taşırlarken kullandıkları yöntem, yiyecek kaynağı ile yuvanın arasında optimal yolu bulmanın yanı sıra, herhangi bir şekilde yol kapandığında yeni bir yolun bulunması ve bireyleri bu yola yönlendirilmesi, yeni ya da daha iyi bir yiyecek kaynağı bulunduğunda bireyleri bu yola sevk edilmesi gibi özellikleri barındırmaktadır. Bu özelliklere bakıldığında, karıncaların yiyecek arama yönteminin duyurga ağlarında yönlendirme algoritması olarak kullanılabilmesi düşünüldü ve ANCOR algoritması tasarlandı.

Tez yazımı aşamasında yardımlarını esirgemeyen Ayda DOĞRUL'a, ve yüksek lisans öğrenimimi BİDEB Yurt İçi Yüksek Lisans Bursu ile destekleyen TÜBİTAK'a teşekkürü bir borç bilirim.

Aralık 2007

Deniz Demiray

## İÇİNDEKİLER

<b>KISALTMALAR</b>	<b>vi</b>
<b>TABLO LİSTESİ</b>	<b>vii</b>
<b>ŞEKİL LİSTESİ</b>	<b>viii</b>
<b>SEMBOL LİSTESİ</b>	<b>x</b>
<b>ÖZET</b>	<b>xi</b>
<b>SUMMARY</b>	<b>xii</b>
<b>1. GİRİŞ</b>	<b>1</b>
<b>2. BİYOLOJİK TEMELLER</b>	<b>3</b>
2.1 Temel İletişim Mekanizmaları	3
2.1.1 Dolaylı Toplumsal İletişim Yöntemleri	3
2.1.2 Olumlu Geribesleme	5
2.1.3 Olumsuz Geribesleme	8
2.1.4 Salınımların Kuvvetlendirilmesi	8
2.1.5 Çoklu Etkileşim	9
2.1.6 Feromon Bombası	9
2.2 Sosyal Böceklerde Davranış Modelleri	9
2.2.1 Kendi Kendine Toplanma ve Organize Olma	10
2.2.2 Toplu Karar Alma	11
2.2.3 İş Bölümleme ve Takım Çalışması	13
<b>3. TOPLULUK ZEKASI, KARINCA ALGORİTMALARI VE KARINCA KOLONİSİ OPTİMİZASYONU</b>	<b>15</b>
<b>4. DUYARGA AĞLARINA GENEL BAKIŞ VE KARINCA ALGORİTMALARININ KULLANIMI</b>	<b>20</b>
4.1 Giriş	20
4.2 Duyarga Düğümünün Tanımı	20
4.3 Duyarga Ağlarının Genel Özellikleri	22
4.4 TinyOs	23
4.5 Duyarga Ağları ve Tasarsız Ağlar	24
4.6 Kablosuz Duyarga Ağlarının Kullanım Alanları	25
4.7 Duyarga Ağlarında Kullanılan Yönlendirme Protokolleri	26
4.7.1 Genel Özellikler	26
4.7.2 Sel Baskını Yönlendirme Algoritması	27

4.7.3 Dedikodu Yönlendirme Algoritması	28
4.7.4 Bilgi İçin Müzakereleli Duyarga Protokolleri (SPIN)	28
4.7.5 Veri-Merkezcil Algoritmalar, İlgili Yayılımı ve Özellik Tabanlı İsimlendirme	29
4.7.6 Yönlendirmeli Yayılım Algoritması	31
4.8 Duyarga Ağları İçin Karınca Algoritmaları Tabanlı Yönlendirme Algoritmaları	32
4.8.1 Duyarga GÜdümlü ve Maliyetten Haberdar Karınca Yönlendirmesi Algoritması	33
4.8.2 Sel Baskını Karınca Yönlendirmesi Algoritması	34
4.8.3 Sırtında Taşınabilir Sel Baskını Karınca Yönlendirmesi Algoritması	34
<b>5. ANCOR</b>	<b>35</b>
5.1 Giriş	35
5.2 Algoritmanın Genel Özellikleri	36
5.2.1 ANCOR ve Diğer Algoritmalar	36
5.2.2 Eşleştirme	37
5.3 Feromon	38
5.3.1 Koku Dağılım Modeli	38
5.3.1.1 Temel Koku Dağılım Modeli	39
5.3.1.2 Doğrusal Koku Dağılım Modeli	39
5.3.1.3 Üstel Koku Dağılım Modeli	40
5.3.2 Sıfır (Nötr, Null) Feromon	41
5.3.3 Olumlu (Çekici) Feromon	42
5.3.4 Olumsuz (İtici) Feromon	42
5.3.4.1 Anti Feromon ve Olumsuz Feromon	42
5.3.5 Buharlaştırma	42
5.3.6 Feromon Bombası	43
5.4 Yönlendirme ve Güncelleme Mekanizmaları	44
5.5 Kullanılan Paketler	44
5.5.1 Karınca Paketi	44
5.5.2 Feromon Paketi	46
<b>6. ANCOR'UN SAFHALARI</b>	<b>47</b>
6.1 Başlangıç Safhası	48
6.2 Pekiştirme Safhası	49
6.3 Yönlendirme Safhası	52
<b>7. TEST</b>	<b>54</b>
7.1 Test Platformu	54

7.1.1 ANCORS++	54
7.1.2 ANCORS	55
7.1.3 PANCORS	55
7.2 Test Sonuçları	56
7.2.1 Parametre Tahmini	57
<b>8. TARTIŞMA VE GELECEK ÇALIŞMALAR</b>	<b>75</b>
<b>KAYNAKLAR</b>	<b>76</b>
<b>EKLER</b>	<b>80</b>
<b>ÖZGEÇMİŞ</b>	<b>82</b>

## **KISALTMALAR**

<b>ANCOR</b>	: Ant Colony Routing
<b>TSP</b>	: Traveling Salesman Problem
<b>ACO</b>	: Ant Colony Optimization
<b>GPS</b>	: Global Positioning System
<b>ID</b>	: Identifier
<b>PA</b>	: Power Available
<b>ME</b>	: Maximum Energy
<b>MH</b>	: Minimum Hop
<b>SPIN</b>	: Sensor protocols for information via negotiation
<b>REQ</b>	: Request
<b>ABC</b>	: Ant Based Control

## TABLO LİSTESİ

	<u>Sayfa No</u>
<b>Tablo 4.1</b> Duyurga düğümü temel elemanları .....	21
<b>Tablo 5.1</b> Doğadaki ve ANCOR'daki kavramların eşleştirilmesi .....	37
<b>Tablo 5.2</b> ANCOR karıncasının veri yapısı .....	45
<b>Tablo 5.3</b> ANCOR feromonu veri yapısı .....	46
<b>Tablo 7.1</b> Kare ızgara ağ örnekleri (radyo menzili = 1) .....	59
<b>Tablo 7.2</b> Kare ızgara ağ örnekleri (radyo menzili = 1,8) .....	70



## ŞEKİL LİSTESİ

	<u>Sayfa No</u>
Şekil 2.1 : Karıncaların kullanacakları ikili alternatif yolun gösterimi [7] ..	6
Şekil 2.2 : Zamanla geçen karınca sayısı [7] .....	6
Şekil 2.3 : Yiyecek kaynağı ile yuva arasında en kısa yolun bulunması [7] .	7
Şekil 3.1 : Karınca kolonisi optimizasyonu ileri sezgiseli genel akışı .....	17
Şekil 4.1 : Duyarga düğümü [36] .....	21
Şekil 4.2 : Duyarga düğümü elemanları arası ilişki [37] .....	22
Şekil 4.3 : Duyarga ağı [37] .....	23
Şekil 4.4 : Bir yol örneği [37] .....	26
Şekil 4.5 : SPIN algoritması örneği [37] .....	29
Şekil 4.6 : Bir ağ örneği [37] .....	30
Şekil 4.7 : Yönlendirmeli yayılım genel yapısı [42] .....	31
Şekil 5.1 : Doğrusal koku dağılım modeli örneği .....	40
Şekil 5.2 : Üstel koku dağılım modeli örneği .....	41
Şekil 5.3 : Buharlaştırma örneği .....	43
Şekil 6.1 : ANCOR algoritması başlangıç safhası akış diagramı .....	49
Şekil 6.2 : ANCOR algoritması pekiştirme safhası akış diagramı .....	50
Şekil 6.3 : ANCOR algoritmasında Çinli Şapkası yapısı örneği .....	51
Şekil 6.4 : ANCOR algoritmasında yönlendirme safhası akış diagramı .....	53
Şekil 7.1 : PANCORS akış diagramı .....	56
Şekil 7.2 : 5x5'lik ızgara ağ örneği (radyo menzili = 1) .....	58
Şekil 7.3 : 5x5'lik ızgara ağ örneği (radyo menzili = 1 karınca sayısı = 1) için hedefi bulma ortalaması .....	59
Şekil 7.4 : 5x5'lik ızgara ağ örneği (radyo menzili = 1 karınca sayısı = 1) için hedefi bulma varyansı .....	60
Şekil 7.5 : 5x5'lik ızgara ağ örneği (radyo menzili = 1 karınca sayısı = 5) için hedefi bulma ortalaması .....	61
Şekil 7.6 : 5x5'lik ızgara ağ örneği (radyo menzili = 1 karınca sayısı = 5) için hedefi bulma varyansı .....	61
Şekil 7.7 : 5x5'lik ızgara ağ örneği (radyo menzili = 1 karınca sayısı = 10) için hedefi bulma ortalaması .....	62
Şekil 7.8 : 5x5'lik ızgara ağ örneği (radyo menzili = 1 karınca sayısı = 10) için hedefi bulma varyansı .....	62
Şekil 7.9 : 5x5'lik ızgara ağ örneği (radyo menzili = 1 karınca sayısı = 15) için hedefi bulma ortalaması .....	63
Şekil 7.10 : 5x5'lik ızgara ağ örneği (radyo menzili = 1 karınca sayısı = 15) için hedefi bulma varyansı .....	63
Şekil 7.11 : 10x10'luk ızgara ağ örneği (radyo menzili= 1 karınca sayısı= 1) için hedefi bulma ortalaması .....	64
Şekil 7.12 : 10x10'luk ızgara ağ örneği (radyo menzili= 1 karınca sayısı= 1) için hedefi bulma varyansı .....	64

<b>Şekil 7.13</b>	: 10x10'luk ızgara ağ örneği (radyo menzili= 1 karınca sayısı= 5) için hedefi bulma ortalaması .....	65
<b>Şekil 7.14</b>	: 10x10'luk ızgara ağ örneği (radyo menzili= 1 karınca sayısı= 5) için hedefi bulma varyansı .....	65
<b>Şekil 7.15</b>	: 10x10'luk ızgara ağ örneği (radyo menzili=1 karınca sayısı= 10) için hedefi bulma ortalaması .....	66
<b>Şekil 7.16</b>	: 10x10'luk ızgara ağ örneği (radyo menzili=1 karınca sayısı= 10) için hedefi bulma varyansı .....	66
<b>Şekil 7.17</b>	: 10x10'luk ızgara ağ örneği (radyo menzili=1 karınca sayısı= 15) için hedefi bulma ortalaması .....	67
<b>Şekil 7.18</b>	: 10x10'luk ızgara ağ örneği (radyo menzili=1 karınca sayısı= 15) için hedefi bulma varyansı .....	67
<b>Şekil 7.19</b>	: 25x25'lik ızgara ağ örneği (radyo menzili=1 karınca sayısı= 10) için hedefi bulma ortalaması .....	68
<b>Şekil 7.20</b>	: 25x25'lik ızgara ağ örneği (radyo menzili=1 karınca sayısı= 10) için hedefi bulma varyansı .....	68
<b>Şekil 7.21</b>	: 25x25'lik ızgara ağ örneği (radyo menzili=1 karınca sayısı= 15) için hedefi bulma ortalaması .....	69
<b>Şekil 7.22</b>	: 25x25'lik ızgara ağ örneği (radyo menzili=1 karınca sayısı= 15) için hedefi bulma varyansı .....	69
<b>Şekil 7.23</b>	: 5x5'lik kare ızgara ağ örneği (radyo menzili = 1,8) .....	71
<b>Şekil 7.24</b>	: $\delta = 0.01$ ve 2500 düğümlü ağ için kaynak düğümü bulma ve hedef düğüme dönme ortalamaları .....	71
<b>Şekil 7.25</b>	: $\delta = 0.001$ ve 2500 düğümlü ağ için kaynak düğümü bulma ve hedef düğüme dönme ortalamaları .....	72
<b>Şekil 7.26</b>	: $\delta = 0.01$ ve 2500 düğümlü ağ için kaynak düğümü bulma ve hedef düğüme dönme varyansları .....	73
<b>Şekil 7.27</b>	: $\delta = 0.001$ ve 2500 düğümlü ağ için kaynak düğümü bulma ve hedef düğüme dönme ortalamaları .....	73
<b>Şekil 7.28</b>	: $\delta = 0.01$ ve 2500 düğümlü ağ için hedef düğüme dönme varyansı .....	74
<b>Şekil 7.29</b>	: $\delta = 0.001$ ve 2500 düğümlü ağ için hedef düğüme dönme varyansı .....	74

## SEMBOL LİSTESİ

$c_i$	: TSP probleminde şehirler
$d_{ij}$	: TSP probleminde i ve j şehirleri arası uzaklık
$V$	: Graf üzerindeki şehirler
$E$	: Graf üzerindeki şehirler arasındaki bağlantılar
$\tau_{ij}(t)$	: Graf üzerindeki i ve j şehirleri arasındaki feromon miktarı
$\Omega$	: Kısıt koşulları
$W$	: Graf üzerindeki elemanlara atanmış ağırlıklar
$\theta$	: $\Omega$ kısıt koşullarının zamana göre değişimi
$\alpha$	: ACO'da feromonun etkinliği
$\mu_{ij}$	: ACO'da i j bağlantısının çekiciliği
$\beta$	: $\mu_{ij}$ nin çekiciliği
$p_{ij}$	: ACO'da i düğümündeki karıncanın j düğümüne hareketinin olasılığı
$\rho$	: Buharlaşma
$\Delta\tau_{ij}$	: Karıncanın bıraktığı feromon miktarı
$L_k$	: Karıncanın k. turunun maliyeti
$a$	: Duyarga ağlarında paketin iletilmesi için gerekli olan enerji
$x$	: ANCOR'da feromon miktarı
$y$	: ANCOR'da algılanan feromon miktarı
$maxMesafe$	: ANCOR'da düğüme bırakılan feromonun algılanabileceği uzaklık
$mesafe$	: ANCOR'da feromonun algılandığı uzaklık
$\partial$	: ANCOR'da üstel koku dağılım modeli ince ayar parametresi
$\delta$	: ANCOR'da buharlaşma katsayısı
$\beta$	: ANCOR'da karıncaların sık kullanılan düğümlere gitme eğilimi
$y$	: ANCOR'da algılanan feromon miktarı
$komşu_olasılığı$	: ANCOR'da karıncanın i komşusuna gitme olasılığı
$komşu_feromonu$	: ANCOR'da düğümün i komşusunun feromon miktarı
$maxAtlama$	: ANCOR'da karıncanın gidebileceği en fazla mesafe
$atlama$	: ANCOR'da karıncanın gittiği mesafe
$\Delta$	: ANCOR'da feromon güncelleme miktarı
$\rho$	: ANCOR'da feromon güncellemenin etkinliği
$A$	: Kare şeklindeki ızgara ağ yapısının bir kenarındaki düğüm sayısı
$I$	: Optimal yol uzunluğu

# DUYARGA AĞLARI İÇİN KARINCA KOLONİSİ TABANLI YÖNLENDİRME ALGORİTMASI

## ÖZET

Bu yüksek lisans tezinde, duyurga ağlarında çalışmak üzere tasarlanan yeni bir yönlendirme algoritması (ANCOR) anlatılacaktır. ANCOR algoritması, doğadaki karınca kolonilerinin yeni yiyecek kaynağı bulup yiyeceklerini yuvalarına taşırlarken, ya da koloni için yeni bir yuva mekanı ararlarken gösterdikleri davranışlardan esinlenilerek hazırlanmıştır.

Karınca kolonilerinde, çok sayıda birey tek bir amaç uğruna bir arada çalışırlar. Bu çalışma esnasında herhangi bir yönetici birey mevcut değildir, ayrıca bütün karıncalar birbirlerinin hareketlerinden direkt biçimde haberdar da değildirler, buna rağmen karıncalar organize bir biçimde çalışırlar. Tekil birey olarak ele alındığında son derece basit davranışlar gösteren karıncalar, koloni düzeyinde bakıldığında karmaşık davranış modelleri sergilemektedirler. Sosyal böcek gruplarında sıkça görülen bu olguya *kolektik zeka* veya *topluluk zekası* adı verilmektedir.

Karınca kolonilerindeki her birey birtakım basit kurallara göre hareket ederler ve dolaylı yollardan birbirleri ile iletişim içindedirler. Karıncaların, yaşadıkları fiziksel çevre üzerinde yaptıkları değişimler, ileride aynı çevrede bulunacak olan karıncaların davranışlarını etkiler, bu olgu özellikle birden çok seçenek karşısında karıncanın seçim yapması sırasında etkisini gösterir, söz gelimi karıncanın takip edebileceği birden çok yol varsa, önceden takip edilmiş olan yolun ya da daha fazla karınca tarafından tercih edilmiş yolun kullanılma olasılığı daha fazla olacaktır. Yiyecek ya da yuva arayışı sırasında karıncaların kendi kendilerine organize olmalarını sağlayan bu mekanizmanın, duyurga ağlarında çalışacak, kendi kendine organize olabilen ve etkin bir yönlendirme algoritması olarak kullanılabilmesi düşünüldü ve ANCOR geliştirildi.

# **ANT COLONY BASED ROUTING ALGORITHM FOR SENSOR NETWORKS**

## **SUMMARY**

A new routing algorithm for sensor networks (ANCOR) is presented in this master thesis. The essential idea behind the algorithm is imitating the real acts of ants for finding new food resources and bringing them to their nests or searching for a new or better place for their nests.

In colonies of ants, there is a large number of individuals working together to achieve a single goal. Ants work together in an organized manner, but in fact, they are not communicating directly, and there is no dedicated colony manager. As a single individual, each ant follows some simple rules, but they establish complex behavioral models at colony level. This is a common phenomenon at social insect colonies which we call swarm intelligence.

Each ant in the colony follows some simple rules and is capable of communicating in a simple indirect method. Changes made by ant on the physical environment, affects forthcoming ants behavior. This phenomenon takes effect especially when an ant has to make a decision. For example when an ant has more than one choice of path to follow, the most used path has a higher chance for selection. This mechanism which ensures self organization of ants while foraging or searching for a new nest site, can be tough as a new routing algorithm which inspires ANCOR.

## 1. GİRİŞ

Bu yüksek lisans tezinde, duyurga ağıları için yeni bir yönlendirme algoritması (ANCOR) tasarlanmıştır. ANCOR ismi, ingilizcede karınca kolonisi yönlendirmesi anlamına gelen *Ant Colony Routing* kelimelerinin kısaltılması ile oluşturulmuştur.

Algoritmanın temelindeki ana fikir, doğadaki karıncaların yeni yiyecek kaynakları bulup bu yiyecekleri yuvalarına taşıırken gösterdikleri davranışları taklit etmektir. Yapılan çalışmanın özgün yanı, doğal bir karınca kolonisindeki davranışların olabildiğince gerçeğine yakın taklit edilmiş olmasıdır. Bunun için konuyla ilgili, sosyal böceklerle dair birçok biyolojik yayın incelenmiş, çeşitli sosyal böceklerin davranış modelleri araştırılmıştır.

Karınca kolonilerinde, birden çok birey, tek bir amaca ulaşmak için birlikte çalışırlar. Amaç ne olursa olsun organize bir şekilde çalışırlar, ama gerçekte ne yapmaları gerektiğini ya da nasıl davranmaları gerektiğini söyleyen bir yönetici birey yoktur. Her karınca bulunduğu ortamdan haberdardır, diğer karıncalarla dolaylı yoldan iletişim kurabilir ve birtakım basit kuralları izler. Bu bireylerin basit davranışlarına geniş bir bakış açısından yaklaşıldığında, aslında her bir hareketin büyük ve karmaşık bir mekanizmanın küçük birer parçaları oldukları görülür. Bu yapı, bilişim kültüründe kolektik zeka veya topluluk zekası olarak bilinir ve karınca kolonisi optimizasyonunun temelini oluşturur.

ANCOR algoritmasını geliştirmedeki amacımız, karıncaların ve diğer sosyal böceklerin, kolektik zeka adı altında incelenen davranışlarının bilgisayar ağlarında yeni bir yönlendirme algoritması olarak kullanılabileceğini düşünmemizden kaynaklanmaktadır. Bu konuda daha önceden yapılan bir takım çalışmalar olduğu tespit edilmiş (örneğin *AntNet*) ve bu çalışmalar incelenmiştir.

Doğal ortamda binlerce birey içeren karınca kolonilerindeki her birey, feromon adı verilen kokulu ve havada dağılan bir kimyasal madde salgılayarak geçtikleri yollar üzerinde iz bırakırlar. Takip eden karıncalar da feromon bırakarak, bu izlerin daha kalıcı olmasını sağlarlar. Eğer takipçi karınca sayısı yeterli değilse, buharlaşma

yoluyla bu feromon izleri silinir. Bu yapı, karınca kolonilerinin dinamik olarak yiyecek kaynaklarına doğru en kısa yolu bulabilmesine olanak tanır. Görece daha küçük bireylere sahip karınca kolonilerinde feromon bırakmak yerine zincirleme koşu adı verilen bir yöntem kullanılır. ANCOR algoritmasının tasarlanması sırasında, kalabalık karınca kolonilerinde kullanılan feromon bırakma yöntemi esas alındı. ANCOR'un özellikle duyurga ağlarında kullanılabileceğinin düşünülmesinin nedeni, kablosuz ağlar içerisinde duyurga ağlarındaki düğüm sayısının, örneğin tasarsız ağlara kıyasla, tıpkı doğadan örnek aldığımız karınca kolonilerindeki gibi görece fazla düğüm içermesinden kaynaklanmaktadır.

Bu çalışmanın ilerleyen bölümlerinde öncelikle ANCOR algoritmasının fikrîsel olarak ortaya çıkmasına zemin hazırlayan biyolojik makalelerde anlatılan temel olgular anlatılacak, doğadaki dolaylı toplumsal iletişim yöntemlerinden bahsedilecek, sosyal böceklerin kendi kendilerine organize olmaları ve toplu karar almaları sırasında kullandıkları yöntemler izah edilecek, takım çalışmaları, iş bölümlenme ve görev paylaşımından bahsedilecektir. Daha sonra topluluk zekası ve karınca algoritmalarına değinilecek, ardından duyurga ağları genel olarak anlatılacaktır. Karınca kolonisi algoritmalarının yönlendirme algoritması olarak kullanılmalarına değin yapılmış daha önceki çalışmalardan bahsedilip ANCOR algoritmasına geçilecektir. Tezin geri kalan kısmında ANCOR algoritması detaylı olarak anlatılacaktır.

Bu yüksek lisans tezinde, birbirinden farklı birçok konunun incelenmesinin nedeni, ANCOR algoritmasının disiplinler arası bir araştırmanın ürünü olmasındandır. Bu araştırma kapsamında biyoloji, yapay zeka, davranış bilimleri ve kablosuz ağlar bulunmaktadır.

ANCOR algoritması üzerine yazılan bir makale yayınlanmıştır [1].

## **2. BİYOLOJİK TEMELLER**

ANCOR'un teorik temelleri, karıncaların ve diğer sosyal böceklerin davranışlarından esinlenilerek oluşturulmuştur. Bu nedenle sosyal böceklerin biyolojik tepkileri, genel davranış modelleri ve bu modelleri oluşturan temel mekanizmalar incelenmiştir

Bu bölümde genel olarak sosyal böceklerden ve davranışlarından örnekler verilerek, bu davranışların altında yatan mekanizmalar açıklanmıştır. Bu örnekler, basit bireylere sahip böcek topluluklarının, topluluk olarak ele alındığında ne kadar karmaşık görevlerin üstesinden gelebildiklerini göstermektedir. Bu davranışlarının anlaşılması, son 30 yıl içerisinde sosyal böcek kolonilerinin bir çok bilim adamına yeni sezgisel algoritmalar geliştirmede ilham kaynağı olmuştur.

### **2.1 Temel İletişim Mekanizmaları**

Sosyal böcek kolonilerinde görülen davranış modellerinin temelinde yatan iletişim mekanizmaları türlere göre farklılık göstermektedir. Kendi kendine organize olma, toplu karar alma, iş bölümlenme gibi karmaşık olguların, bireysel olarak ele alındıklarında basit davranışlar sergileyen böcek gruplarında ortaya çıkabilmesini mümkün kılan bu mekanizmalar, topluluk zekası kavramının da ortaya çıkabilmesi için temel oluşturmaktadır.

#### **2.1.1 Dolaylı Toplumsal İletişim Yöntemleri**

Sosyal böcek kolonisindeki dolaylı iletişim kavramı ilk olarak 1959 da Fransız böcek bilimci Pierre-Paul Grassé tarafından, termit yuvalarının inşa yöntemleri üzerine yaptığı araştırmalar sırasında ortaya atılmıştır [2]. Grassé, inşaatın yönetiminin ve düzenlenmesinin termitler arası doğrudan iletişimle değil, yuvanın yapısı ile sağlandığını göstermiştir. Buna göre yuvadaki uyarıcı bir yapılanış, termitin farklı bir tepkisini tetikler, bu davranış sonucunda oluşan yapı ise ileride başka davranışların tetiklenmesinde rol oynar [3]. Termitler planlı ve karmaşık



yuvalar oluşturan böceklerdir. Yuvalarının yapısı belli bir organizasyona tabiidir; dışı koruma amaçlı kalın duvarlarla çevrilidir ve içinde, en merkezi ve korunaklı yerinde kraliçe termitin odası bulunur. Bu odanın yakınlarında yumurtaların konulduğu doğumhane odaları, ve yiyecek depoları vardır. Termit yuvaları ayrıca çok iyi havalandırma sistemlerine sahiptirler.

Grassé, bu karmaşık yapıları inşa edebilmeleri için, termitlerin bir şekilde birbirlerinin yaptıkları işlerden haberdar olmaları gerektiğini, yani bir şekilde iletişim kurmaları gerektiğini ortaya koymuştur. Buna göre iletişim doğrudan olarak kurulabildiği gibi (örneğin antenler aracılığıyla, yiyecek ya da sıvı transferi sırasında, görsel olarak ya da yakındaki termitin kokusu olarak), dolaylı olarak da kurulabilir.

Grassé, *Bellicositermes Natalensis* ve *Cubitermes* termitlerinin yuvalarını nasıl oluşturduklarını detaylı bir biçimde araştırmıştır. Termitler toprak parçacıklarına feromon bırakırlar ve bu parçacıkları topraklar haline getirirler. Termitler hazırladıkları bu toprakları rastgele yerlere üst üste yığarlar. Bu rastgele yığma işlemi, yığıntılardan bir tanesinin kritik bir boyuta ulaşmasına kadar devam eder. Yığmada biriken feromonlu topraklar, çevredeki termitleri tetikleyerek kendine çeker. Bundan sonra eğer yeteri kadar termit varsa, eş güdümlü bir biçimde sütunlar oluşturulur. Daha fazla feromonlu toprak, oluşturulan sütundan daha fazla feromon yayılmasına neden olacak ve böylece daha fazla termit tetiklenecektir (çığ etkisi). Eğer çevrede yeterli miktarda termit mevcut değilse, yığmalardaki feromon çığ etkisi oluşturacak termitleri tetikleyemeyecek ve buharlaşmanın etkisi ile zamanla yok olacaktır [4].

Bu gözlemleri yaptıktan sonra Grassé, *Stigmergy* terimini, “Performansa bağlı olarak işçilerin tetiklenmesi” olarak ortaya koymuştur [4]. En genel anlamı ile *Stigmergy* terimi, hayvanlar arası etkileşime aracılık eden genel mekanizmalar için kullanılmaktadır [3]. Daha özel anlamıyla, kendi kendine organize olup gelişen sistemlerdeki bireylerin, buldukları fiziksel çevreyi değiştirmek suretiyle birbirleri ile dolaylı olarak iletişim kurmaları anlamına gelmektedir. Bu yapıda, bir birey bir davranışta bulunurken, daha önceki bireylerin davranışlarını da göz önünde bulundurur, bir başka deyişle bir bireyin fiziksel çevreyi değiştirmesinden sonraki bir zamanda, başka bir bireyin bu değişim doğrultusunda bir davranış sergilemesi anlamına gelir. Bu yapıya termit yuvaları dışında örnek olarak, yabanası

yuvalarındaki özel bazı odaların, diğer odaların yapımını tetiklemesi gösterilebilir [3, 5]. Bal arılarındaki altıgen peteklerin yapımı da, dolaylı iletişime örnek olarak verilmektedir [2].

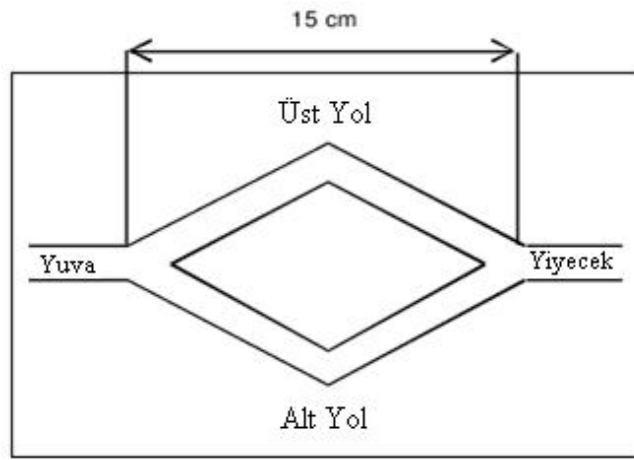
### 2.1.2 Olumlu Geri Besleme

Olumlu geri besleme, sosyal böcek gruplarının kendi kendilerine organize olabilmelerini sağlayan basit davranış kuralları olarak açıklanabilir. Olumlu geri beslemedeki amaç, yapılan bir hareketin sonuçlarının dolaylı yoldan ileriki bireylere aktarılmasıdır. Bir yemek kaynağına doğru bireylerin toplanması ve yemek kaynağına giden yolun pekiştirilmesi örnek olarak verilebilir. Bir yemek kaynağına doğru toplanma, birçok karınca kolonisinde iz bırakma ve iz takip etmeye, arılarda ise dans etmeye dayalı bir olumlu geri besleme mekanizmasıdır [3].

Arılardaki olumlu geri besleme mekanizmasının çalışma şekli şöyledir: nektar kaynağını bulan bir arı, kovana geri dönüp kovadaki bir arıya nektarını bırakır. Bu noktadan sonra dans ederek diğer arılara nektar kaynağının yönünü ve kovana olan mesafesini gösterebilir, kaynağı diğer arılara göstermeden nektar transferini kendisi üstlenebilir ya da kendi de nektar kaynağına gitmeyebilir. Eğer arı kolonisine eşit mesafelerde ve eşit içerikte iki nektar kaynağı sunulmuşsa, arıların iki kaynağına da simetrik olarak dağıldıkları gözlenmiştir. Eğer sunulan kaynaklardan biri diğerinden daha iyi ise, arıların iyi olan kaynağına yöndikleri görülmüştür. İyi olan kaynak, kötü olandan daha sonra sunulmuş olsa bile, arı kolonisi iyi olan kaynağına geçiş yapabilmıştır [3].

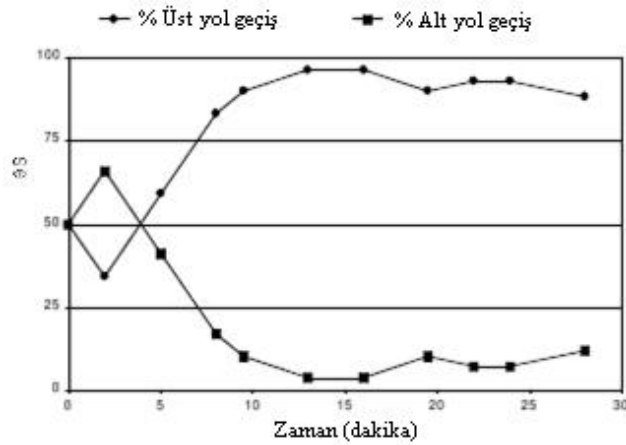
Doğadaki canlıların birçoğu, iletişim kurmak için feromon adı verilen bir kimyasal madde salgılar. Feromonun birçok çeşidi olmasına karşın, bu tez çalışmasında, karıncaların yiyecek ararlarken salgıladıkları feromon çeşidi ele alınmıştır. Bu feromon çeşidi, karıncaları kendine çeken bir etki yapmaktadır. Yiyecek ya da yuvaları için yeni bir mekan ararlarken, karıncalar geçtikleri yolları işaretlemek amacıyla yolun üzerine feromon bırakırlar. Bu yollardan daha çok karınca geçtikçe yol üzerinde daha fazla feromon birikir ve yolun diğer karıncaları çekme gücü artar. Feromon salgıları ile işaretlenen bu yollara *feromon izleri* adı verilir. Yiyecek ve yeni yuva için elverişli mekan ararken iz bırakma – iz takip etme yöntemi karınca kolonileri içerisinde sık görülen bir mekanizmadır [3]. Ancak bu mekanizmada bir mutlak sonucun (totoloji) oluşmasına izin vermeyen rastgelelik de yer alır.

Olumlu geri besleme mekanizmasına basit bir örnekle olarak, yuvadan aynı anda iki karıncanın çıktığı ve yiyecek kaynağına iki farklı yoldan gittiği varsayalım. Karıncalar, yolculukları sırasında salgıladıkları feromonu yola bırakarak yollarını işaretlerler. Yiyecek kaynağına kısa yoldan gidecek olan karınca daha kısa sürede yuvaya döner, böylece kısa yol uzun yola göre iki misli daha fazla feromon barındırır (karınca hem giderken hem dönerken yola feromon bırakır) [6]. Yuvadan çıkan diğer karıncalar çok büyük olasılıkla daha fazla feromon içeren yolu tercih ederler, ve yol üzerine feromon bırakırlar, böylece kısa olan yol daha da çekici hale gelir ve kullanılma olasılığı daha da artar.



**Şekil 2.1:** Karıncaların kullanacakları ikili alternatif yolun gösterimi [7]

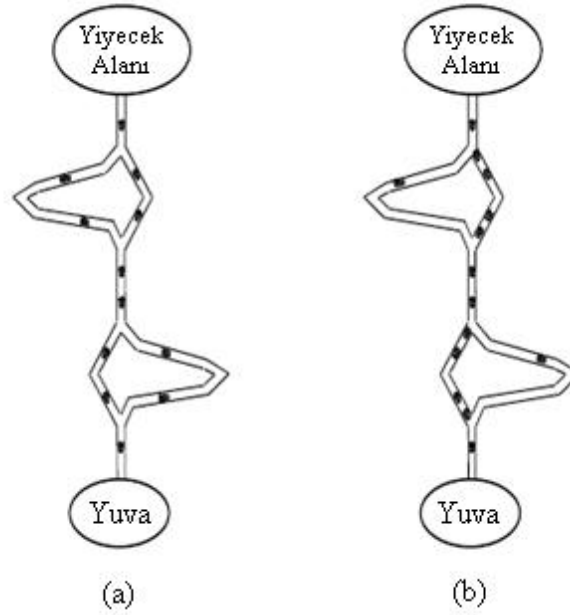
Yola bırakılan feromon miktarının karıncaların yol seçimleri üzerindeki etkilerini araştırmak üzere yapılmış bir deneyde, uzunlukları aynı olan iki adet alternatif yol sonuna yiyecek bırakılmış (Şekil 2.1) ve karıncalar serbest bırakılıp hangi yoldan



kaç karıncanın geçtiğine bakılmıştır [7].

**Şekil 2.2:** Zamanla geçen karınca sayısı [7]

Şekil 2.2’de görüldüğü gibi, yol uzunlukları aynı olmasına karşın, deneyin başlarında üstteki yoldan daha fazla karınca geçtiği için üst yola daha fazla feromon bırakılmıştır. Tesadüfi olarak gerçekleşen bu olay, deneyin ilerleyen safhalarında karıncaların büyük çoğunlukla üstteki yolu tercih etmelerine neden olmuştur.



**Şekil 2.3:** Yiyecek kaynağı ile yuva arasında en kısa yolun bulunması [7]

Şekil 2.3’te, karıncaların yuvaları ile yiyecek kaynakları arasında gidebilecekleri toplam dört adet alternatif yol vardır. Yapılan deneyler, her denemede karıncaların yiyecek kaynağına giden en kısa yolu bulabildiklerini göstermiştir. Şekil 2.3-a, deneyin başlangıç safhasını göstermektedir. Yuva alanından serbest bırakılan karıncaların yarısına yakını uzun yoldan, diğer yarısı ise kısa yoldan gitmişlerdir. Yine, yiyecek kaynağına yakın olan yol ayırımında da karıncalar yaklaşık yarı yarıya iki yola da dağılmışlardır. Birim zamanda, kısa yolu kullanan karınca sayısı daha fazla olacağından, kısa yollardaki feromon miktarı uzun yollara göre daha fazla olacaktır. Bu olgunun sonuçları Şekil 2.3-b’de gösterilmektedir. Karıncalar yuvaları ile yiyecek kaynakları arasındaki alternatif yollardan en kısa olanını seçmişlerdir ve olumlu geri besleme mekanizması ile yol üzerindeki feromon devamlı güncellenmektedir. Ancak Şekil 2.3-b’deki gösterimde de görülebileceği gibi, yol üzerinde ne kadar feromon olursa olsun karıncaların bir kısmı yoldan saparak farklı

seeneklerden gidebilirler. Bu zellik sayesinde, arama alanı geniř tutularak var olan ancak henüz keřfedilmemiř yiyecek kaynakları bulunabilir.

Dađıtık hesaplama ve agzli sezgisel algoritmalar birlikte kullanıldıđında, olumlu geri beslemenin iyi sonuların erken bulunmasında etkin rol oynadıđı gzlenmiřtir [8].

Karınca kolonilerinde olumlu geri besleme farklı amalarla da kullanılabilir. Hart ve Ratnieks, bitki yapraklarını kopartıp tařıyan Atta karıncalarını incelemiřlerdir [9]. Atta karıncaları, yuva giriřinde biriken yaprak paraları yznden yuvaya giriřin zorlařması sorununu zlemek iin, yuva giriřinin dıřında, ve yiyecek tařıma yolu zerinde yaprak kmecikleri oluřturmaktadırlar. Oluřan yaprak kmeciklerinin boyutunun, yaprak tařıyan bir karıncanın yaprađını bırakması olasılıđı zerinde nemli etkisi olduđu saptanmıřtır.

### **2.1.3 Olumsuz Geri Besleme**

Olumsuz geri besleme, olumlu geri beslemenin karřıtı olarak dřnlebilir ve sistemi dengelemeyi sađlar. Doymuluk, tkenme ya da rekabet gibi Őekillerde grlebilir [3]. Yiyecek arama rneđinde olumsuz geri besleme rnekleri, nc karıncaların sayısının kısıtlı olması, doymuluk, yiyecek kaynađının tkenmesi, yiyecek kaynađında sıkıřıklık ya da yiyecek kaynakları arasındaki rekabet olarak sayılabilir.

### **2.1.4 Salınımın Kuvvetlendirilmesi**

Salınımın kuvvetlendirilmesi, karınca kolonilerinin kendi kendilerine organize olabilmelerini sađlayan nemli mekanizmalardan biridir. Rastgele yryřler, yapılan hatalar, yapılan iřin yarıda bırakılması ve bunlar gibi normal iř akıřı dıřında kalan faaliyetleri ierir. Dođal iř akıřı ierisine rastgeleliđi katan bu eylemler aslında koloninin yařamı iin hayati nem tařımaktadır. nk bu normal iř akıřı dıřında kalan faaliyetler sayesinde, yeni yiyecek kaynakları keřfedilebilir ya da mevcut kaynađa giden daha kısa bir yol bulunabilir. Karıncalar yiyeceđe giden yola bırakılmıř feromonu koklayarak yollarını bulurlar [3]. Ancak bazı durumlarda karıncalar yollarını Őařırıp kaybolabilirler. Kaybolan ya da rastgele iř yapan karıncaların oranları, koloni nfusuna gre olduka az sayılabilecek miktardadır. Yoldan sapan bu karıncalar ilk bakıřta iřin verimliliđini dřryor gibi gzkebilir, nk iř gc aısından bir azalma szkonusudur. Ancak yollarından sapan bu

karıncalar yeni yiyecek kaynaklarına rastlayabilirler ve diğer karıncaları bu kaynağa yönlendirebilirler [3].

### **2.1.5 Çoklu Etkileşim**

Karınca kolonilerinin kendi kendilerine organize olabilmelerini mümkün kılan en önemli olgulardan bir tanesi çoklu etkileşimdir. Karınca kolonilerindeki her birey, hem kendi eylemlerinin sonuçlarından, hem de diğer bireylerin eylemlerinin sonuçlarından etkilenecek hareket eder. Örneğin karınca kolonilerinde bireyler yola bırakılan feromonu takip ederlerken, sadece kendi bıraktıkları feromonu değil, diğer tüm bireylerin bıraktıkları feromon sonucu oluşan feromon izlerini takip ederler.

### **2.1.6 Feromon Bombası**

Yiyecek arama ya da taşıma sırasında karıncalar geçtikleri yollara feromon bırakarak feromon izlerinin güncellenmesini sağlarlar. Bu işlem sırasında vücutlarında depolamış oldukları feromonu azar azar kullanarak feromon bırakma işleminin devamlılığını sağlamış olurlar.

Ancak bazı durumlarda, karıncalar vücudunda biriktirdiği bütün feromonu bırakabilirler. Bu mekanizmaya feromon bombası adı verilmektedir. Herhangi bir böceğin ya da başka bir canlının saldırısına uğramış karınca feromon bombası bırakabilir, ve ortaya çıkan kokunun şiddetine bağlı olarak uzak mesafelerdeki karıncaları kendine çeker. Bu bir bakıma yardım çağrısı olarak da düşünülebilir.

Feromon bombası, ANCOR algoritmasında, karıncaların mevcut işleyişini ani bir biçimde değiştirmek amacıyla, ya da bazı özel durumlarda çabuk yakınsama sağlamak amacıyla kullanılmıştır.

## **2.2 Sosyal Böceklerde Davranış Modelleri**

Bu bölümde, doğadaki sosyal böcek gruplarında görülen davranış modelleri anlatılacaktır. İncelemeler sonunda, böcek topluluklarının dahili ve çevresel değişimlere göre iş düzenlerini uyumlu olarak değiştirebilecek çok esnek mekanizmalara sahip oldukları görülmüştür.

Sosyal böcek gruplarında, çalışan işçi böceklerin sayısı değişebildiği gibi, işçi böceklerin yaptıkları işler de oldukça esneklerdir. Örneğin bal arısı kolonilerinde, nektar toplayan arılarla toplanan nektarı alıp depolayan arı sayıları çabuk bir

şekilde değişebilmektedir. İş bölümündeki değişim hemen hemen bütün böcek türlerinde görülebilir; genellikle bir işe yoğunlaşmış böceğin, gerektiğinde başka bir işi de yapabilmesi olarak karşımıza çıkmaktadır. İş organizasyondaki değişimin bir başka türü ise, işin ana yapısının değiştirilmesidir. Yaprak kesici karıncalarda görüldüğü gibi, örneğin bir işçi ile yapılan bir işin, alt işlere bölünerek birkaç işçi tarafından yapılması sağlanabilir.

### **2.2.1 Kendi Kendine Toplanma ve Organize Olma**

Sosyal böcek gruplarında görülen en temel modellerden bir tanesi kendi kendine toplanma ve kendi kendine organize olmadır. En genel anlamda kendi kendine toplanma terimi, sosyal yapıyı oluşturan bireylerin herbirinin, yapının bir parçası olması anlamına gelir. Kendi kendine organize olma ise, bireyler arasındaki basit etkileşimlerin global çerçevede çeşitli örüntülerin ve karmaşık mekanizmaların oluşmasını sağlaması anlamına gelmektedir [10]. Bu iki terim birbiriyle iç içe geçmiştir, kendi kendine toplanma olgusu birçok kez kendi kendine organize olmayı da beraberinde getirmektedir. Her iki terimde de birden çok birey fonksiyonel bir bütünün parçalarını oluşturmak için bir aradadır.

Kendi kendine toplanma davranışını gösteren *Leptothorax* karıncaları, diğer karınca gruplarına göre daha az sayıda birey barındırırlar. Yuvaları diğer karıncalarınkine nazaran daha basit olduğundan, nispeten az sayıda birey içeren koloni kolaylıkla toplanabilir. *Leptothorax* karıncalarının yuvalarının merkezinde kraliçe karınca bulunur. Yumurtlamaktan sorumlu olan kraliçe karınca, diğer karıncalara nazaran yüksek miktarda feromon salgılar. Yuvanın ortasında bulunan kraliçe karıncadan gelen kokunun yoğunluğuna göre, işçi karıncalar yuvadaki yerlerini belirleyebilirler. *Leptothorax* yuvalarında, yaşlı ve yavaş olan işçiler yuvanın iç kesimlerinde çalışırlar ve böylece daha küçük bir alanda hareket ederler. Hızlı işçiler ise yuvanın dış kısımlarında ve yuva dışındaki alanlarda çalışırlar [10].

Kendi kendine organize olmaya verilebilecek tipik örnek, hemen hemen tüm büyük karınca kolonilerinde görülen yiyecek bulma ve yiyeceği yuvaya taşıma davranışıdır. Karınca kolonisi optimizasyonunun da temelini oluşturan bu davranış, koloni üyelerinin izlediği basit kuralların birleşiminden oluşmuştur [6]. Kolonideki her karınca, yiyeceğe giden yolu bulabilmek için en keskin kokuyu takip eder ve geçtiği yola feromon bırakır. Bu basit algoritmanın çalışması şu şekilde

açıklanabilir: yiyeceğe giden iki yol varsa, kısa yoldan giden karınca daha çabuk geri dönecektir. Karıncalar hem giderken hem de gelirken feromon bıraktıklarından, kısa yolun kokusu daha keskin olacaktır. Arkadan gelen karıncalar da en keskin kokuyu takip edeceklerinden, belli bir süre içerisinde yemeğe giden en kısa yol otomatik olarak kullanılıyor olacaktır.

Termit yuvalarının inşası sırasında işçi termitlerin kullandıkları yöntem de, kendi kendine organize olmaya verilebilecek bir diğer güzel örnektir. İnşaat sırasında termitlerin “hava akışını tespit et ve deliği kapat” şeklinde basit bireysel algoritmalar kullanarak büyük ve karmaşık yuvalar inşa ettikleri bilinmektedir [11].

### 2.2.2 Toplu Karar Alma

Sosyal böcek gruplarında toplu karar alma mekanizmaları, topluluğun sahip olduğu birey sayısına göre değişkenlik göstermektedir. Bireyler arası olası yanlış etkileşimlerin stokastik etkilerinin azalmasında birey sayısının fazla olması önemli bir unsurdur [12]. *Leptothorax Albipennis* karıncaları, diğer karınca topluluklarına nazaran oldukça az sayıda birey içeren koloniler kurarlar, bu kolonileri birkaç yüz ya da bir kaç bin birey içerebilir. Bu karıncalar diğer karınca türlerinin aksine feromon salgılamazlar, yani yiyecek aramada ya da yeni bir yuvaya taşınma sırasında, bireylerin izleyecekleri feromon izleri oluşturamazlar. *Leptothorax* karıncaları yeni bir yuva sahasına göç ederlerken, ya karıncalara yol öğretilir ya da teker teker taşınır. Bu yönteme zincirleme koşu adı verilmektedir. Bu yöntemi kullanan bir diğer karınca topluluğu olan *Temnothorax Albipennis* karıncaları deney ortamında incelenmiş ve yemeğe giden yolu diğer bireylere nasıl öğrettikleri araştırılmıştır [13]. *Temnothorax* karıncalarının, yürüyüşün hızını ve rotasını kontrol eden sinyaller kullandığı, bir karıncanın diğerine yemeğe giden yolu öğrettiği anlaşılmıştır. Öğreten karınca, öğrencisine öğretmek için kendi davranışlarını değiştirir. Öğretmek eylemi öğreten karıncaya bir maliyet oluşturur. Öğretme aşamasında öğreten karıncanın yiyeceğe gitme başarımı düşerken, öğrencinin başarımının arttığı görülmüştür. Deneylerde, öğreten karıncayı arkadan takip eden öğrencinin, antenleri ile öndeki karıncanın karnına ya da bacaklarına sıkça vurduğu sürece, öndeki öğreten karıncanın yürümeye devam ettiği görülmüştür. Eğer öğrenci arkasında olmasa, öğretmenin yiyecek taşıma işini yaklaşık dört kat daha hızlı yaptığı tespit edilmiştir. Öğretme sırasında karıncalar arasındaki mesafe açılırsa, öndeki karınca yavaşlar ve arkadaki hızlanır. Bu yöntemle yolu öğrenen karınca



ileride öğretmen olup diğer karıncalara yolu öğretir. Deneylerde ayrıca, öğrenerek yemeğe kadar giden bir karıncanın, geri dönüş yolunda çok daha hızlı hareket ettiği tespit edilmiştir [13].

Kalabalık karınca kolonilerindeki bireyler, feromon salgılayarak toplu karar alma mekanizmasını işletirler. Olumlu geri beribesleme ile güncellenen yollar sayesinde koloninin kararı yine koloni tarafından alınmış olur. Bireyler, gidecekleri yolları seçerlerken yolun uzun ya da kısa olduğuna kendileri bakmadan, koloninin genel kararı ile öğrenip giderler.

*Leptothorax* karıncaları gibi küçük koloniler halinde yaşayan bireylerin üzerine daha ağır bir yük binmektedir. Laboratuvar ortamında yapılan deneylerde [12], kaya oyuklarında yuvalanan *Leptothorax* karıncalarına alternatif yuva seçenekleri sunulduğunda, belirli bir boyutun altındaki yuvalarla ilgilenmedikleri tespit edilmiştir. Yapılan deneylerde, öncelikle küçük bir grup karıncanın keşfe çıktığı gözlemlenmiştir. Yeni bir yuva bulduklarında, karıncalar yuvanın boyutuna bakıp geri dönerek diğer karıncalara yol gösteriliyorlar. Karıncaların davranışı yeni yuvanın kalitesini doğrudan belirlemek yerine göç etme isteğini gösteriyor. Vasat bir yuva için göç çağrısı geciktiriliyor, böylece diğer karıncaların yeni yerleri araması ve olası daha iyi bir yuva bulabilmesi şansı oluyor, ayrıca daha iyi bir yer bulunamasa bile koloni yuvasız kalmamış oluyor. Sonuçta, koloninin boyutu küçüldükçe, toplu karar almanın yanında kişisel kararların da etkili olduğu görülüyor. [14]'de, *Leptothorax* karıncalarının yeni yuva seçimlerinde, koloninin %30'luk kısmından genel kararın çıkmış olduğu gösterilmektedir. Ayrıca, benzer özelliklere sahip yuvalara giden farklı karıncaların, geri dönmeden önce yaptıkları geciktirme işleminin hemen hemen aynı süre tuttuğunu belirtilmektedir. Seçilen yeni yuva için zincirleme koşu başlatılmasının ardından, belirli miktarda karınca yolu öğrenir ve geriye kalan karıncaları üstlerine alarak taşır. Böylece taşınma işlemi büyük ölçüde hızlanır. Üstte taşınan karıncaların yolu öğrenemedikleri tespit edilmiş. Karıncalara iki yuva sunulduğunda, iyi olana talebin daha hızlı olduğu, ancak kötü olan yuvaya da bir miktar karıncanın göç ettiği görülmüştür. Belli bir süre sonra, kötü olan yuvadaki karıncalar ters zincirleme koşu başlatır ve önceki yuvalarına geri dönerler. Bundan sonra genelde zincirleme koşudan ziyade diğer karıncalar tarafından sırtta taşıma yöntemi ile yeni yuvalarına götürülürler [14].

Toplu karar alma mekanizması, bal arılarının yeni yuvalarını kuracakları yeri belirlemelerinde de görülmektedir. Eski yuvalarını terk eden arı kolonisi genelde bir ağaçta bekler ve öncü arılar yeni yuva için elverişli bir ağaç kovuğu ya da benzer bir oyuk arayışına koyulurlar. Potansiyel bir yuva mekanı bulduklarında koloniye geri dönüp koloninin önünde dans ederler. Buldukları yuvanın boyutları, savunmasının ne kadar kolay olduğu gibi özellikler öncü arının dansına yansır. Bu dans ayrıca arılara yuvanın uzaklığını ve yönünü de belirtir. Bu dansların özelliklerine göre koloni bir yuvada karar kılar ve göç eder [15].

### 2.2.3 İş Bölümleme ve Takım Çalışması

Sosyal böcek kolonilerinde görülen bir diğer gelişmiş davranış modeli iş bölümlemedir. Bu davranış, bireyler arasında iş bölümü ve alt işlerin belirlenmesini kapsar.

[16]'da, arılarda nektar (balözü) taşınması işinin nasıl yapıldığı incelenmiştir. İnceleme sırasında bal arılarına (*Apis*), yaban arılarına (*Bombini*) ve iğnesiz arılara (*Meliponini*) bakılmış. Yapılan araştırmalar sonucunda bal arılarının taşıdıkları nektarı kovanın içerisinde bekleyen alıcı arılara teslim ettikleri, buna karşın yaban arılarında nektar transferine rastlanmadığı, iğnesiz arıların ise sadece beş türünde nektar transferi yapıldığı görülmüştür [16].

Bal arılarının nektarı teslim etme işlemi, iş bölümlemeye örnek olarak gösterilebilir. İş bölümleme, koloni genişledikçe daha sık görülen bir olgudur. Bunun nedeni, taşınan ürünü teslim etmek için bir partner bekleme süresinin, kalabalık kolonilerde çok daha düşük olmasındandır. Bir arının taşıdığı nektarı birden çok arıya teslim etmesi olayı da, hem bal arılarında hem de iğnesiz arılarda görülmektedir. Çoklu transfer olayı, taşıyıcı arının dans etmesiyle ilişkilidir.

*Apis mellifera* arı kolonilerinde, yaşa bağlı bir iş bölümüne rastlanılmıştır [17]. Buna göre en genç arılar, kraliçenin ve larvaların bakımından ve doyurulmasından sorumludurlar. Orta yaştaki arıların görevi ise kovanın bakımı ve yiyeceğin depolanmasıdır. En yaşlı arılar ise nektar ve polenin toplanması ve kovanın savunulması işlemlerini gerçekleştirirler. Normal şartlar altında bir arının nektar toplayabilmesi için 21 günlük olması gerekir. Bundan sonra arının 1 ila 3 haftalık ömrü kalmıştır. Ancak olağanüstü durumlarda, örneğin kolonide yeterli miktarda yetişkin arı bulunmaması durumunda, çok daha genç arıların, örneğin 7 günlük

arıların, nektar toplama görevini üstlendikleri görülmüştür. Sonuç olarak arı kolonilerinde iş bölümünün yaş tabanlı olmasının yanı sıra, yiyecek durumu, meteorolojik etkenler, avlanılma gibi çevresel etkenler parametre olarak alındığında oldukça esnek bir yapıya sahip olduğu görülmüştür.

Birçok karınca kolonisinde, işçi karıncalar arasında boyuta göre bir iş bölümü yapılmaktadır [18]. Buna göre küçük karıncalar kuluçkaya yatma görevini üstlenirken, büyük karıncalar genellikle asker olurlar ve yuvanın güvenliğinden sorumludurlar.

*Atta* cinsi yaprak kesen karıncalar, kestikleri yaprakları doğrudan yuvaya taşıyabildikleri gibi, taşıyıcılar arasında doğrudan nakil yoluyla [6], ya da kümeler oluşturma yoluyla [19] iş bölümlere de yapabilirler. Yaprak kesen karıncalar, yiyecek kaynakları ile yuvaları arasında yaprak kümeleri oluştururlar. Bu oluşum genellikle yuva ağzında, yaprak getirme oranı yaprak işleme oranını aştığı zaman, oluşabilecek darboğazı önlemek için meydana getirilir. Amaç, bırakılan bir yaprağın alınma olasılığını yükseltmektir. Taşıma yolu üzerinde de kümeler oluşturulmaktadır. Özellikle yiyecek kaynağına veya yuva ağzına yakın yerlerde olan engeller, küme oluşumuna yol açmaktadır. Ancak herhangi bir karınca kümeden bir yaprak alırken, kendi boyutuna uygun bir yük seçmediğinden, küçük karıncalara büyük yaprak düşmesi durumları görülebilir, ve bu da taşıma işleminin başarımı üzerinde olumsuz etki yapmaktadır [19]. Sonuç olarak, eğer taşıyıcı karıncanın yükünü bırakması gerekiyorsa, yaprakların hedefe ulaştırılması açısından, küme oluşturulması performans açısından en iyi çözümdür.

Sosyal böcek kolonilerinde görülen bir diğer davranış modeli ise takım oluşturmadır. *Hodotermes Mossambicus* termitlerinde [9], bir grup termit çimene tırmanıp çimene üstten keser ve yere atar, diğer bir grup termit ise düşen parçaları yuvaya taşır. Bu davranış, takım çalışmasına verilebilecek tipik bir örnektir [20].

### **3. TOPLULUK ZEKASI, KARINCA ALGORİTMALARI VE KARINCA KOLONİSİ OPTİMİZASYONU**

Topluluk zekası, ya da kolektik zeka kavramı ilk olarak 1989 yılında Gerardo Beni, Susan Hackwood ve Jing Wang tarafından ortaya atılmıştır. Topluluk zekası, optimizasyon problemlerini çözmek için, kendi kendine organize olabilen dağıtık sistemlerdeki ortak davranış modellerinin incelenmesi sonucu ortaya çıkmış bir yapay zeka tekniğidir [21].

Doğadaki sosyal böcekler herhangi bir yönetici birey olmaksızın çalışırlar. Yaptıkları takım çalışması kendi kendine organize olabilir. Bireyler arası eş güdüm yine bireylerin davranışları neticesinde ortaya çıkar. Teker teker bireylerin davranışları çok ilkel düzeyde olsa bile, çok sayıda bireyin bir araya gelmesi ile oluşan kolonilerdeki ortaklaşa çalışma, karmaşık yapıların ortaya çıkmasına zemin hazırlar [6]. Sosyal böcek gruplarının ortaklaşa davranışlarından, ve bu davranışlar sonucu ortaya çıkan karmaşık yapılardan esinlenilerek topluluk zekası kavramı ortaya atılmıştır.

Topluluk zekası ilk olarak basit mekanik ajanların en yakın komşuları ile etkileşimleri ve bu etkileşimlerden doğan kendi kendine organize olan hücresel robotik sistemler üzerine yapılan çalışmalarda ortaya atılmıştır. Daha sonra Marco Dorigo ve arkadaşları topluluk zekasının tanımını “sosyal böcek gruplarının ve diğer hayvan topluluklarının ortaklaşa davranışlarından esinlenilerek hazırlanan algoritmaların ve dağıtık problem çözme metotlarının tümü” olarak genişletmişlerdir [5]. Bu tanım, birçok karınca topluluğunun, bal arılarının, eşek arılarının, termitlerin, hamam böceklerinin ve daha birçok böcek topluluğunun davranışlarını kapsamaktadır. Birçok problemin çözümü, karıncaların yiyecek ararlarken kullandıkları yöntemi temel alan sezgisel algoritmalar tarafından bulunmuştur.

Topluluk zekası kavramının alt başlığı olarak sayılabilecek olan karınca algoritmaları, dolaylı toplumsal iletişim yöntemlerini kullanan gerçek karınca kolonilerinin izlenmesi sonucu ortaya çıkan, çok ajanlı sistemlere verilen genel isimdir. Bu algoritmalarda, ajanlara yapay karıncalar (ya da sadece karıncalar) adı verilir. Karıncalar arasında haberleşme sistemi olarak dolaylı toplumsal iletişim yöntemleri kullanılır. Bu yöntemlerden en çok kullanılanı, karıncaların yiyecek aramada kullandıkları iz bırakma – iz takip etme yöntemidir [4]. Bireyler hareket halindeyken feromon bırakırlar ve daha sonra gelen bireyler belli bir olasılıkla bu feromonlu yolu takip ederler.

Doğadaki karıncaların davranışlarından esinlenilerek hazırlanan ilk karınca algoritması, “karınca sistemi” idi [22, 23]. Ayrık optimizasyon problemlerini çözebilmesi amacıyla tasarlanan karınca sistemi, ilk olarak gezgin satıcı probleminin (TSP) çözümünde kullanılmıştır. TSP çözümünde amaç, verilen bir harita üzerindeki  $n$  adet şehirin hepsini sadece bir kez ziyaret ederek, en kısa yoldan bütün şehirleri dolaşmaktır.  $C_i$  ve  $C_j$  şehirleri arasındaki mesafe  $d_{ij}$  olsun. Bu mesafe ya iki şehir arasındaki direkt öklid mesafesi olabilir ya da bir graf üzerinde  $G = (V, E)$  şeklinde belirtilebilir. Bu grafta şehirler  $V$  ile, şehirler arasındaki bağlantılar da  $E$  ile gösterilmiş olsun. Grafın tam bağlı olmasına gerek olmadığı gibi şehirler arasındaki uzaklıkları belirten uzaklık matrisi de simetrik olmak zorunda değildir. TSP de amaç, verilen bu özellikteki bir graf üzerindeki bütün düğümlerden sadece bir kez geçerek bütün düğümleri en düşük maliyetle ziyaret etmektir. Karınca sistemi, TSP yi çözmek için şöyle bir yöntem kullanır; graf üzerindeki düğümler arasında bulunan bağlantılarda  $(E(i, j))$  bir feromon değişkeni kullanılır  $(\tau_{ij}(t))$  ve graf üzerinde dolaşan karıncalar bu feromon değişkenlerine göre gidecekleri bir sonraki düğümün hangisi olduğunu seçerler. Karınca graf üzerindeki bir turunu tamamladığında, geçtiği bağlantılar üzerindeki feromon değişkenlerini günceller, bu güncelleme işlemi sırasında karıncanın bıraktığı feromon miktarı, karıncanın tamamladığı turun performansına bağlı olarak değişir. Kısa turlar daha performanslı oldukları için, bu yolları oluşturan bileşenler üzerine daha fazla feromon bırakılır.

Karınca sisteminin ardından ortaya atılan bir başka yöntem ise, yine karıncaların yiyecek ararlarken kullandıkları yöntemden esinlenilerek ortaya atılmış karınca kolonisi optimizasyonudur (ACO). Graflar üzerinde uygun yollar bulma problemine dönüştürülebilir hesaplamalı problemleri çözen olasılıksal bir teknik olarak ilk defa

Marco Dorigo'nun doktora tezinde bahsedilmiştir [24, 25]. Bu yöntem, temel karınca algoritmalarının incelenmesi sonucunda elde edilmiş bir ileri sezgiseldir ve karınca sistemi dahil tüm karınca algoritması tabanlı yöntemleri içinde barındırır. İleri sezgisel, çeşitli hesaplamalı problem sınıflarına ve optimizasyon problemlerine çözüm sunan genel bir algoritma, ya da diğer sezgiselleri uygun sonuçlar bulmaları yolunda yönlendiren bir üst seviye stratejisi olarak açıklanmaktadır. Karınca kolonisi optimizasyonunun dışında evrimsel algoritmalar, tabu arama ve benzetimli tavlama çok bilinen ileri sezgisellerdir. Özellikle dinamik olarak değişen graflar üzerinde, karınca kolonisi optimizasyonunun diğer ileri sezgisellere göre avantajları vardır; karınca kolonisi optimizasyonu devamlı olarak çalışabilir ve kendini gerçek zamanlı olarak değişimlere uydurabilir.

Karınca kolonisi optimizasyonunun graflar üzerinde minimum maliyetli uygun yollar bulur. Yolun uygunluğu denilince problemin içerisine bir takım kısıt koşulları konulmuş olur.  $G = (C, L, W)$  ve  $\Omega$  ile belirtilen bir graf üzerinde  $C = \{c_1, c_2, \dots, c_n\}$  problemin elemanları olsun.  $L = \{l_{ij}(c_i, c_j) \mid c_i, c_j \in C\}$  ile elemanlar arasındaki bağlantılar belirtilmiş olsun.  $W, C$  elemanlarına ya da  $L$  bağlantılarına, ya da her ikisine de atanmış ağırlıklar kümesini belirtsin ve  $\Omega(C, L, \theta)$   $C$  ve  $L$  üzerine belirtilmiş kısıt koşulları olsun. Burada  $\theta$  kısıt koşullarının zaman içerisinde değişebileceğini belirtir. Bu bağlamda, az önce ele alınan TSP problemine dönülürse,  $C$  ile belirtilen elemanlar TSP deki şehirler,  $L$  ile belirtilen şehirlerin arasındaki bağlantılar,  $W$  ile belirtilen ise bu bağlantıların uzunluğudur.  $\Omega$  ile belirtilen kısıtlamalar ise uygun çözümlerin graf üzerindeki bütün şehirleri yalnızca bir kez içermesi gerektiğini belirtir.

```

procedure Karınca Kolonisi Optimizasyonu İleri Buluşalı
  while sonlandırma koşulu yerine gelmedi do
    karınca_aktiviteleri()
    feromon_buharlamasi()
    diğer_aktiviteler()
  end while
end Karınca Kolonisi Optimizasyonu İleri Buluşalı

```

**Şekil 3.1** Karınca kolonisi optimizasyonu ileri sezgiseli genel akışı

Şekil 3.1'de görüldüğü gibi, karınca kolonisi optimizasyonu ileri sezgiseli, bir sonlandırma koşulunun doğru olduğu sürece dönen bir döngü içerisinde tanımlı bir

takım fonksiyonlardan ibarettir. *karınca\_aktiviteleri()* fonksiyonu, graf üzerindeki karıncaların uygun bir sonuç bulmaları ve elemanlar ya da bağlantılar üzerindeki feromon miktarlarını güncellemeleri işlemlerini içermektedir. *feromon\_buharlamasi()* fonksiyonu, grafda tanımlı olan elemanlar ya da bağlantılar üzerindeki feromon miktarlarının zaman içerisinde azalması işlemini yürütür. *diger\_aktiviteler()* ile belirtilen fonksiyon isteğe bağlı olarak kullanılır ve salt karıncaların yaptığı işlemler ile kotarılamayan birtakım işlerin yürütülmesi amacı ile kullanılır.

Karınca kolonisi optimizasyonu ileri buluşsalında kullanılan genel formüller aşağıda sırası ile verilmiştir. Formül 3.1, i düğümündeki bir karıncanın j düğümüne hareket etme olasılığını göstermektedir.

$$p_{i,j} = \frac{[\tau_{i,j}]^\alpha [\mu_{i,j}]^\beta}{\sum [\tau_{i,j}]^\alpha [\mu_{i,j}]^\beta} \quad (3.1)$$

Formül 3.1 de  $\tau_{i,j}$  ile gösterilen, i ile j elemanları arasındaki bağlantı üzerindeki feromon miktarı,  $\alpha$  ile gösterilen ise  $\tau_{i,j}$  nin etkisini kontrol eden bir parametredir.

$\mu_{i,j}$  ile gösterilen, ij bağlantısının çekiciliğidir (genellikle  $1/d_{i,j}$  olarak alınır), ve  $\beta$  parametresi  $\mu_{i,j}$  nin etkisini kontrol etmeye yarar. Karıncalar graf üzerinde hareket ederlerken geçtikleri elemanların ya da bağlantıların sahip olduğu feromon değişkenini güncellerler. Karınca kolonisi optimizasyonu ileri sezgiselinde bu güncelleme işlemi Formül 3.2'deki şekilde genelleştirilmiştir.

$$\tau_{i,j} = \rho \tau_{i,j} + \Delta \tau_{i,j} \quad (3.2)$$

Formül 3.2 de  $\tau_{i,j}$  ile gösterilen, i ile j elemanları arasındaki bağlantı üzerindeki feromon miktarıdır.  $\rho$  parametresi buharlaşma mekanizmasını kontrol eder ve  $\Delta \tau_{i,j}$  parametresi ise karıncanın bıraktığı feromon miktarını belirtir. Karıncanın bıraktığı feromonun miktarı Formül 3.3'de gösterilmektedir.

$$\Delta \tau_{i,j}^k = \begin{cases} 1/L_k & k \text{ karıncası } i,j \text{ bağlantısı üzerinde ise} \\ 0 & \text{diğer durumlar} \end{cases} \quad (3.3)$$

Formül 3.3 de  $L_k$  ile gösterilen, karıncanın k. turunun maliyetidir, ve bu maliyet genellikle yolun uzunluğu olarak alınır.

1991'den günümüze birçok optimizasyon probleminin çözümü için karınca kolonisi optimizasyonu ileri sezgiseli tabanlı algoritmalar geliştirilmiştir. Bu problemlerin en bilinenleri TSP [8, 26], karesel atama [27], zamanlama problemleri [28], araç yönlendirmesi [29], bağlantı yönelimli ağ yönlendirmesi [30], graf renklendirme [31], frekans ataması [32], genelleştirilmiş atama [33], çoklu knapsack [34], optik ağlarda yönlendirme problemleridir [35].



## **4. DUYARGA AĞLARINA GENEL BAKIŞ VE KARINCA ALGORİTMALARININ KULLANIMI**

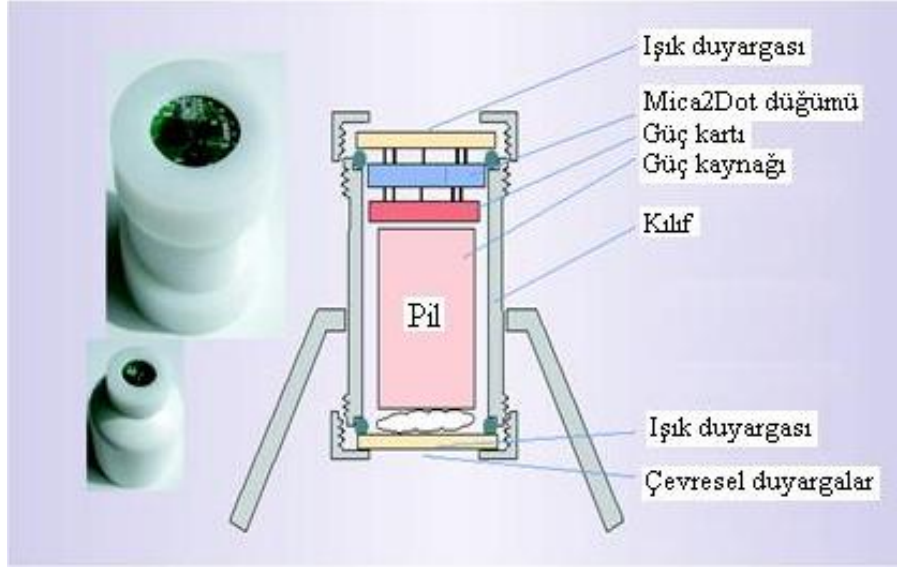
### **4.1 Giriş**

Bu bölümde kısaca duyarga ağları ve genel özellikleri anlatılacaktır. Her geçen gün gelişen ve fiyatları daha da ucuzlayan mikro çipler, çevredeki olayları algılayan ve temel bilgi işleme yeteneklerine sahip olan duyarga düğümlerinin ortaya çıkmasına olanak sağlamıştır. Duyarga ağlarında, çok sayıda duyarga düğümleri birarada çalışarak buldukları ortama dair bilgileri bize ulaştırırlar. Hem sivil hem askeri uygulama alanları bulan bu teknolojinin örnek kullanım alanları bu bölümde anlatılacaktır.

Bu bölümde öncelikle bir duyarga düğümünün tanımı yapılacak, duyarga ağlarının genel özelliklerinden bahsedilecek, ardından duyarga ağlarının kullanım alanlarına örnekler verilecek ve bu ağlarda kullanılan yönlendirme algoritmalarından bahsedilecektir.

### **4.2 Duyarga Düğümünün Tanımı**

En basit anlatımla bir duyarga düğümü, fiziksel dünyadaki kuvvetleri ve alanları algılayabilen, entegre alıcı-vericileri ile haberleşebilen küçük elektromekanik yapılardır (Şekil 4.1). Bu küçük aygıtlar bir fiziksel alana yayılarak gerçek fiziksel olayı yansıtacak yüklü miktarda algıyı aralarında koordine bir biçimde işler ve iletirler. Birçok uygulamada ağ içerisindeki düğümler kablosuzdur ve ağ uzun müddet çalışmak zorundadır.



**Şekil 4.1:** Duyarga düğümü [36]

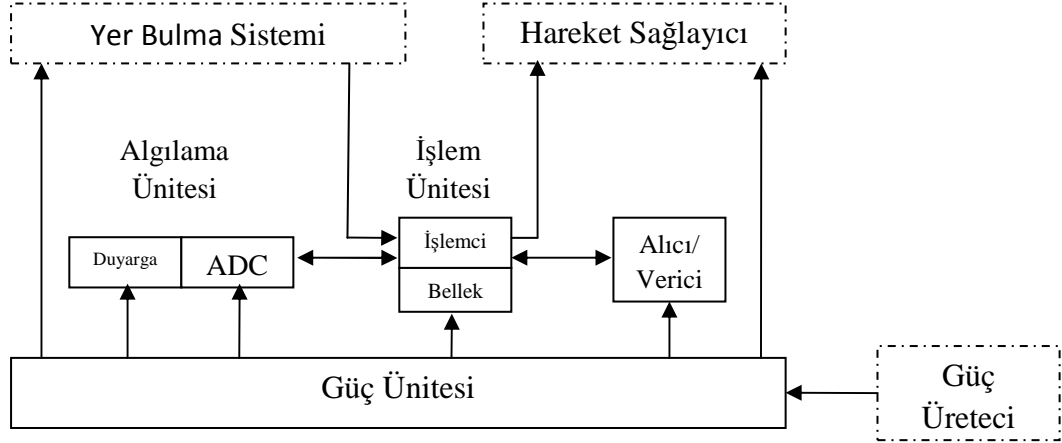
Şekil 4.1’de bir duyarğa düğümünün genel yapısı görülmektedir. Arazi şartlarına uygun hazırlanmış bu örneği, koruyucu bir kılıfın altına yerleştirilmiş, algılama yeteneklerine sahip bir mikrobilgisayar gibi düşünülebilir. Şekil 4.1’de görüldüğü gibi çeşitli duyarğalar da bu kılıfın içerisine yerleştirilmiştir. Bu duyarğalar düğümün fiziksel dünya ile olan bağlantısını sağladıklarından, düğümün işlevini doğrudan etkilerler. Duyarğalar çok çeşitlidir, titreşim algılayan bir duyarğa ile düğümün yakın çevresindeki canlıların hareketleri algılanabilir, eğer ısıyı algılayan bir duyarğa mevcutsa, düğüm yangın riskine karşı kullanılabilir, bunlardan başka çeşitli amaçlara yönelik birçok çeşit duyarğa mevcuttur.

**Tablo 4.1:** Duyarga düğümü temel elemanları

algılama ünitesi
işlem ünitesi
alıcı-verici ünitesi
güç ünitesi

Tablo 4.1’de görüldüğü üzere, bir duyarğa düğümü 4 temel elemandan oluşur. Bu elemanlardan ilki olan algılama ünitesi, duyarğalar ve duyarğaların algılarını işlem ünitesinin anlayabileceği sayısal işaretlere dönüştüren analog-dijital çeviricilerden oluşur. İşlem birimi, genelde küçük bir depolama birimiyle birlikte bulunur,

algılayıcılardan gelen verileri işler, duyarga düğümünün işlevini yapabilmesi için gerekli olan işlemleri yapar. Alıcı-verici ünitesi, düğümün ağa bağlanabilmesini sağlayan arayüzdür, bir radyo cihazı, ya da kızılötesi gibi optik bir cihaz olabilir. Güç ünitesi, duyarga birimine enerji sağlayan bir pildir, verimliliğin artırılabilmesi açısından güneş hücreleri gibi fazladan donanımlar içerebilir.



**Şekil 4.2:** Duyarga düğümü elemanları arası ilişki [37]

Bu temel birimlerin dışında, duyarga düğümünde fazladan bazı üniteler de eklenebilir. Amaca yönelik eklenen bu cihazlara örnek olarak Şekil 4.2’de gösterildiği gibi, yer bilgisinin bilinmesi gereken uygulamalarda kullanılan yer belirleme cihazı (GPS) verilebilir. Bazı uygulamalarda ise, düğümümüze hareket edebilme özelliği kazandırmak isteyebiliriz, bunun için de uygun ekipmanlar düğümümüze eklenebilir.

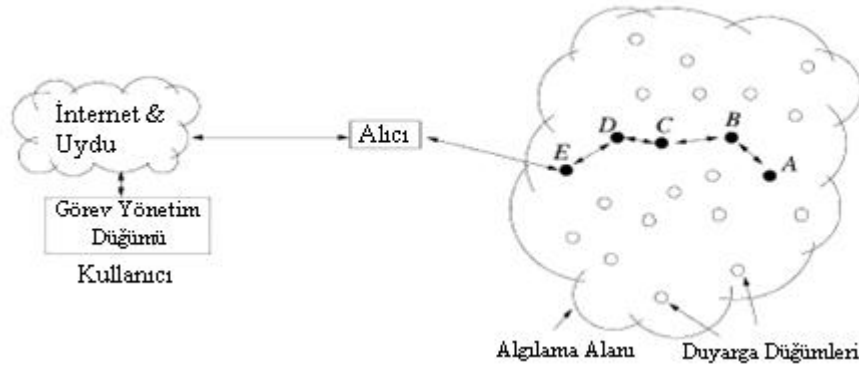
Unutulmaması gereken nokta, bütün bu ünitelerin kibrit kutusuna yakın bir hacimdeki büyüklüğe sığdırılmak zorunda olduğudur.

### 4.3 Duyarga Ağlarının Genel Özellikleri

Kablosuz duyarga ağlarındaki her bir cihaz yapısı itibariyle bazı kısıtlamalara haizdir. Pil gücü, işlem kapasitesi, depolama kapasitesi ve iletişim yetenekleri kısıtlıdır. Duyarga ağı uzun süre çalışmak zorunda olduğundan, pil ömrü ağın yaşam süresi açısından birinci derecede öneme sahiptir. Enerji gereksinimini azaltmak için, radyo gibi donanımların birçoğu genelde kapalı tutulur ve gerektiğinde çalıştırılır. Özellikle açık arazi uygulamalarında, alana yayılan düğümler değişken fiziksel çevreyle fazlaca iç içe olacaklarından, ağı oluşturan düğümler zorlu çevresel

koşullara maruz kalırlar. Geniş bir alana yayılan sık düğümler arasında sıkı bir iletişim olacaktır. Açık alan uygulamalarındaki değişken çevre sonucu düğümler arası bağlantılar da değişimlere uğrayacaktır. Birçok kez, gönderilen paketler yerine ulaşamayacak, pili biten ya da bozulan düğümler ağdan ayrılacaktır. Bütün bu faktörler, iletişimde kullanılan protokolleri daha da karmaşık hale getirmektedir.

Özellikle açık arazi uygulamalarında, duyarga düğümlerinin enerji kaynakları değiştirilemez. Küçük boyutlarından dolayı kısıtlı enerji kaynaklarına sahip düğümler için en önemli kısıtlama, düşük enerji tüketimi gereksinimidir. Bu sebepten dolayı geleneksel ağlar yüksek servis kalitesine erişmeye çalışırken, duyarga ağları için tasarlanan protokoller enerji korunumuna odaklanmışlardır. Açık araziye yayılan düğümler, alıcı düğüm adı verilen düğümden yapılan sorgular çerçevesinde duyargaları aracılığıyla veri toplamaya, ve bu verileri alıcı düğüme iletmeye başlarlar. Ağ içerisinde kullanılan mimari çok atlamalı ve iç yapısızdır (Şekil 4.3).



**Şekil 4.3:** Duyarga ağı [37]

Algılama alanına yayılmış çok miktardaki duyarga düğümlerinden toplanan veri kaynak düğüme kadar duyarga ağı içerisinde ulaştırılır. Bu noktadan sonra internet veya uydu aracılığı ile kullanıcı ile iletişime geçilir.

#### 4.4 TinyOs

TinyOs, kablosuz duyarga düğümlerinde çalışmak üzere tasarlanmış bir işletim sistemidir. Uygulamaya özel yazılımların duyarga düğümleri üzerinde

çalışabilmeleri için bir çatı oluşturur. TinyOs, geliştiriciler için büyük bir yazılım kütüphanesi sunar. Bu kütüphanenin içerisinde ağ protokolleri, dağıtık servisler, duyarga sürücüler ve veri toplama araçları mevcuttur. Sunulan olay güdümlü fonksiyonlar kodlamada kolaylık sağlamaktadır.

Unix gibi klasik işletim sistemleri 50 100 MHz de çalışan 32 bitlik bir mikroişlemciyle birkaç megabyte bellek ve birkaç gigabyte kapasiteli bir sisteme ihtiyaç duyarlar. Günümüzde böyle bir sistem, dahili pili ile saatlerce çalışabilen bir el bilgisayarında bile bulunabilir. Buna karşılık kablosuz duyarga ağlarında bu kaynaklardan çok daha kısıtlı kaynaklar mevcuttur ve bazı uygulamalarda düğümün küçük bir pil ile aylarca çalışması istenebilir. Açık kaynak kodlu TinyOs işletim sistemi geliştirilirken bu kısıtlamalar gözönüne alınmıştır.

Farklı birçok duyarga sistemi için TinyOs işletim sistemi uyarlanmıştır ve böylece geniş bir kullanım alanı bulmuştur.

#### **4.5 Duyarga Ağları ve Tasarsız Ağlar**

Bu bölümde duyarga ağları ile tasarsız ağlar arasındaki farklardan bahsetme gereği duyulmasının sebebi, doğadaki karınca kolonilerinin incelenmesi sırasında, farklı miktarda birey içeren koloniler arasında, yiyecek kaynakları ile yuvaları arasındaki yolları bulma ve diğer karıncalara gösterme metotları açısından farklılıklar olduğunun bulunmasıdır. Feromon ile geçilen yolların işaretlenmesi, ve takipçi karıncalar tarafından yine feromon bırakılmak suretiyle olumlu geri besleme mekanizmasının işletilmesi metodu, yüksek sayıda birey içeren karınca kolonileri arasında kullanılmaktadır. Bununla birlikte düşük sayıda birey içeren kolonilerde, zincirleme koşu metodunun kullanıldığı görülmüştür. Eğer karınca kolonilerini kablosu ağlar olarak ele alırsak, kuşkusuz kalabalık karınca kolonileri olarak duyarga ağlarını incelememiz gerekir. Bu noktada, düşük miktarda birey içeren kolonilere karşılık olarak tasarsız ağların getirilebileceğine inanılmaktadır. Bu sebeple bu bölümde duyarga ağları ile tasarsız ağları karşılaştırma gereği duyulmuştur.

Duyarga ağları ile tasarsız ağlar arasındaki temel farklardan en önemlisi, duyarga ağlarındaki düğüm sayısının çok daha fazla olmasıdır. Duyarga ağlarında düğümler yoğun olarak yerleştirilirler. Değişen fiziksel çevreye mutakiben duyarga ağlarında ağ topolojisi sıkça değişebilir. Bu değişimden dolayı ağda hatalar oluşma olasılığı

fazladır. Yani duyurga düğümleri hataya daha eğilimlidirler. Bir çok tasarsız ağ noktadan noktaya haberleşme tabanlıdır, buna karşılık duyurga ağlarında yayınlama yöntemi ile haberleşme sağlanır. Duyurga düğümlerin işlem gücü, hafıza kapasitesi ve pil ömrü kısıtlıdır, buna karşılık tasarsız ağlarda daha güçlü cihazlar kullanılır. Duyurga ağlarında düğümlerin genel bir tanımlayıcı numaraları (ID) yoktur çünkü ağda çok fazla düğüm bulunur, bunun yerine bölgesel tekil bir tanımlayıcı kullanılır. Tasarsız ağlarda ise her düğümün kendi tanımlayıcısı mevcuttur. Enerji kısıtlamalarından dolayı duyurga ağları için hazırlanan algoritmalar ve protokoller enerji tüketimini azaltmayı ve böylece duyurga ağının genel yaşam süresini uzatmayı hedeflerken, tasarsız ağlarda enerji sorunu genellikle geri planda tutularak başka servis kalitesi parametlerine de aynı önem gösterilebilir.

#### **4.6 Kablosuz Duyurga Ağlarının Kullanım Alanları**

Kablosuz duyurga ağlarının kullanım alanları askeri uygulamalardan doğal hayatı gözlemlemeye, hatta biyomedikal uygulamalara kadar uzanan geniş bir yelpazeye sahiptir.

Kablosuz duyurga ağlarının askeri amaçlı kullanılması, düşman hatlarının ardından kritik verilerin güvenli bir biçimde alınabilmesi açısından çok önemlidir. Uçaktan saçılarak ya da başka bir alet tarafından düşman hattının arkasına saçılan duyurgalar, askerlerin ulaşamadığı alanlar hakkında veri toplamaya başlarlar.

Olası bir uygulama olarak titreşim algılayıcılarla donatılmış duyurgalardan oluşan bir ağ, savaş sırasında düşmanların tanklarının ve diğer birimlerinin yerleri konusunda bilgi verebilir. Algıladığı titreşimleri hafızasındaki titreşim modeli ile karşılaştıran düğüm, belirli bir hata yüzdesi ile o coğrafi bölgede hangi düşman araçlarının faaliyet gösterdiği bilgisini iletebilir. Bu tarz bir ağda yapılabilecek bir sorgu, verilen x-y koordinatlarındaki hareketlilik hakkında bilgi almak olabileceği gibi, herhangi bir sorgu olmadan algıladıkları herhangi bir olayı ileten olay güdümlü algoritmalar da oluşturulabilir.

Olası bir başka uygulamada ise, ısı duyurgaları ile donatılmış düğümler ormana saçılabilir. Olay güdümlü tasarlanmış böylesi bir ağ, orman yangınlarına erken müdahale edilebilmesi açısından yangının başlangıç safhasında alıcı düğüme yangın mahalinin yerini bildirerek görevlilere etkin müdahale olanağı sunabilir.

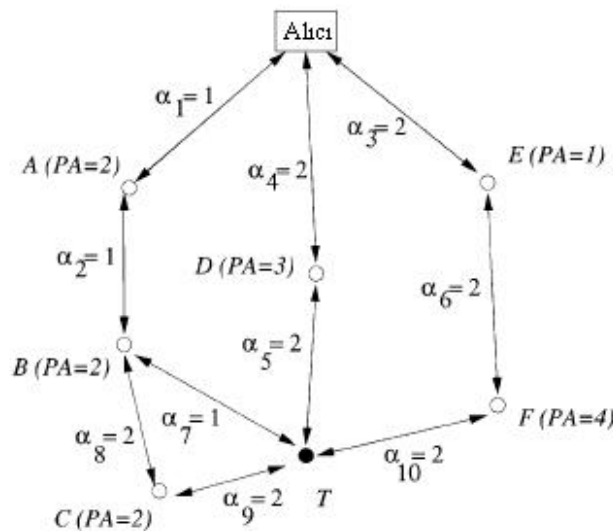
## 4.7 Duyarga Ağlarında Kullanılan Yönlendirme Protokolleri

### 4.7.1 Genel Özellikler

Duyarga düğümleri, bir olayın içine veya yakınına yoğun olarak dağıtılmışlardır. Bir duyarga düğümü ile alıcı düğüm arasında özel çok atlamalı kablosuz yönlendirme protokollerine ihtiyaç duyulur. Hali hazırda bulunan tasarsız ağ yönlendirme protokolleri, duyarga ağlarının ihtiyaçlarını tam olarak karşılayamamaktadır. Duyarga ağlarının ağ katmanı aşağıdaki prensiplere göre geliştirilir:

- Etkin güç kullanımı her zaman en önemli etkidir.
- Duyarga ağları büyük çoğunlukla veri merkezlidir.
- Veri toplama, duyarga düğümlerinin çalışma bütünlüğünü engellemediği sürece kullanışlıdır.
- İdeal bir duyarga ağı özellik tabanlı ve buldukları yerden haberdar olmalıdır.

Duyarga düğümlerinin küçük boyutları, yeterli enerji kaynaklarına sahip olmalarını imkansız hale getirirken, duyarga ağları bu kısıtlı enerji kaynakları ile uzun süre çalışmak zorundadır. Enerjinin etkin kullanımı için aşağıdaki yaklaşımlardan biri ele alınabilir. Bu yaklaşımları açıklamak için T düğümünün ortamı algılayan hedef düğüm olarak kullanıldığı Şekil 4.4 kullanılacaktır.



Şekil 4.4: Bir yol örneği [37]

Alıcı düğümle haberleşmek için aşağıdaki 4 yol olasılığı vardır.

Yol 1: Alıcı – A – B – T, toplam PA = 4, toplam  $\alpha = 3$ ,

Yol 2: Alıcı – A – B – C – T, toplam PA = 6, toplam  $\alpha = 6$ ,

Yol 3: Alıcı – D – T, toplam PA = 3, toplam  $\alpha = 4$ ,

Yol 4: Alıcı – E – F – T, toplam PA=5, toplam  $\alpha = 6$ ,

Şekil 4.4'te PA mevcut gücü ve  $\alpha$  ise ilgili yoldan paketin iletilmesi için gerekli olan enerjiyi ifade eder. Bu bağlamda ortaya atılabilecek çeşitli yaklaşımların hangi yolu seçeceklerine bakılacaktır.

- **En Fazla Gücü (PA) Olan Yol Yaklaşımı**

Bu yaklaşımda, toplamda en fazla gücü olan yol tercih edilir. Toplam PA yol boyunca bütün PA ların toplamına eşittir. Bu yaklaşımla Şekil 4.4'te yol 2 seçilir. Fakat yol 2, yol 1 i içinde içinde barındırdığı gibi fazladan bir düğümü de içerir. Bundan dolayı toplamda en güçlü yol olmasına rağmen gücü en etkin kullanan yol değildir. Sonuç olarak, kaynak düğümü alıcı düğümüne bağlayan yolu uzatarak alternatif olarak sunmak önemli değildir. İkinci yolu eleyerek PA metodu sonunda yol 4 ü elde etmiş oluruz.

- **Minimum Enerji Yolu (ME) Yaklaşımı**

Kaynak düğüm ile alıcı düğüm arasında veri paketlerinin taşınması için en az enerji gerektiren yol seçilir. Şekil 4.4'te yol 1 en az enerji gerektiren yoldur.

- **Minimum Atlamalı Yol (MH)**

Kaynak düğümünden alıcı düğümüne veri paketlerinin taşınması için, bu iki düğüm arasında kalan en az atlamalı yol tercih edilir. Bu yaklaşıma göre Şekil 4.4'te en etkin yol 3. yoldur.  $\alpha$  değerleri aynı olduğunda ME ile MH nin aynı yolu seçtikleri görülür. Bundan dolayı, düğümler aynı güç seviyesinde herhangi bir güç kontrolü yapmadan yayın yaptıklarında MH ile ME aynı olur.

#### **4.7.2 Sel Baskını Yönlendirme Algoritması**

Sel baskını, duyurga ağlarında yönlendirme algoritması olarak kullanılan eski bir algoritmadır [38, 39]. Burada bir veriyi alan her düğüm yayınlama ile veriyi iletir. Bu işlem, veri paketi için belirlenen bir maksimum hop sayısına ulaşana veya verinin gönderileceği düğümüne ulaşılan kadar işlemeye devam eder. Sel baskını



algoritması, masraflı topoloji bakımlarını ve karmaşık yol bulma algoritmalarını gerektirmez, ancak aşağıdaki gibi birkaç eksikliği vardır;

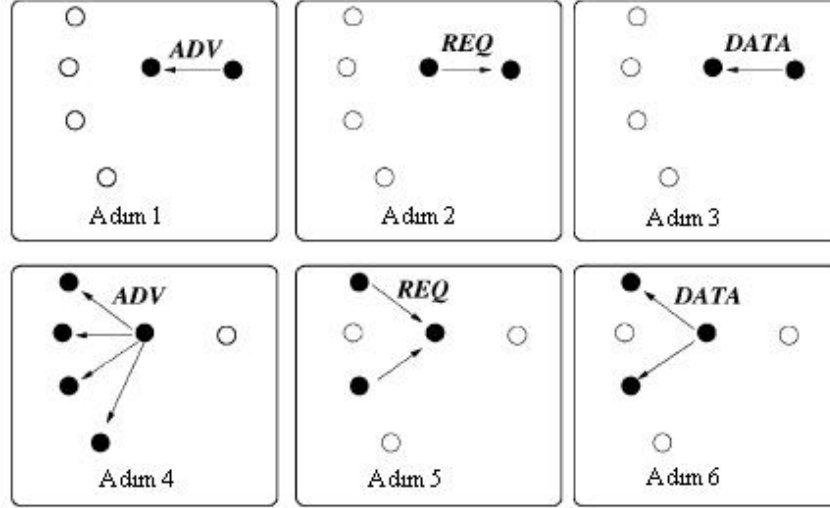
- Göçme: Aynı düğüme bir mesajın birden çok kopyasının gönderilmesine göçme denilir. Eğer A duyarga düğümünün B'nin de komşusu olan N tane komşusu var ise, B düğümü A duyarga düğümü tarafından yollanan N kopya mesajı alır.
- Çakışma: Eğer iki düğüm aynı bölgeyi izliyorsa, aynı olayı aynı anda algılayabilir. Bunun sonucunda da komşu düğümler aynı mesajın birden çok kopyasını alabilir.
- Kaynak Körlüğü: Sel baskını algoritması, mevcut enerji kaynaklarını hesaba katmaz. Enerjiden haberdar olan bir algoritma, her an mevcut enerji miktarından haberdar olmalıdır.

#### **4.7.3 Dedikodu Yönlendirme Algoritması**

Sel baskınının bir türevi olan dedikoduda [40, 41], düğümler kendilerine gelen paketleri yayınlamazlar, rasgele seçtikleri bir komşularına gönderirler. Komşu düğüm veriyi aldığıında, rasgele başka bir düğümü seçer. Bu yaklaşım mesajın her düğümde bir kopyasının bulunmasını sağlayarak çökme problemini önler fakat mesajın bütün düğümleri yayılması uzun sürer.

#### **4.7.4 Bilgi İçin Müzakereli Duyarga Protokolleri (SPIN)**

Bir uyarlamalı protokol ailesi olan SPIN [37, 41] klasik sel baskını algoritmasının eksikliklerini, kaynak uyarlaması ve müzakere yöntemiyle çözer. SPIN protokol ailesi, 2 temel fikri temel alır: duyarga düğümleri daha etkin çalışmalı, verinin tamamını göndermektense veri tanımını göndererek enerji korunmalı ve duyarga düğümlerinin enerji seviyeleri gözlenmelidir.



**Şekil 4.5:** SPIN algoritması örneği [37]

SPIN’de 3 tip mesaj vardır: ADV, REQ, DATA. DATA mesajı gönderilmeden önce, duyarga düğümü DATA’ya dair bir tanımlayıcı içeren ADV mesajını yayınlar (Şekil 4.5’te Adım 1 ile gösterilmiştir). Eğer bir komşu veriyle ilgilenirse, DATA için bir REQ mesajı gönderir ve DATA bu komşu duyarga düğümüne yollanır (Şekil 4.5’te Adım 2 ve Adım 3). Ardından, komşu duyarga düğümü bu işlemi sürdürür (Şekil 4.5 Adım 4, 5 ve 6). Sonuçta, duyarga ağı içerisindeki DATA ile ilgilenen bütün duyarga düğümleri, bir kopya almış olur. SPIN’in veri merkezli yönlendirme tabanlı olduğunu, ve burada duyarga düğümlerin mevcut verilerini ilan edip ilgili alıcı düğümlerden talep beklediklerine dikkat etmek gerekir.

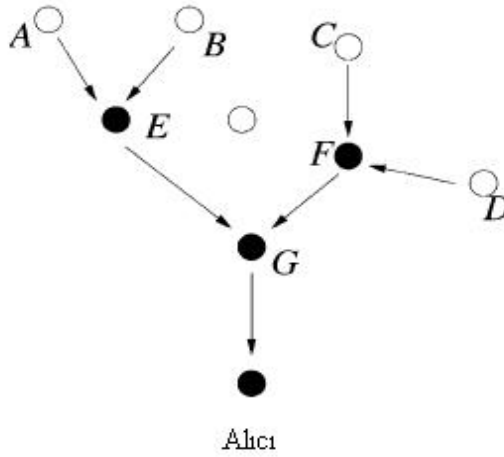
#### **4.7.5 Veri-Merkezcil Algoritmalar, İlgilili Yayılımı ve Özellik Tabanlı İsimlendirme**

Kısıtlılı enerji kaynaklarının dışında bir diğeri önemli konu ise, kablosuz duyarga ağlarında kullanılan yönlendirme algoritmalarının veri merkezli olabileceğidir. Veri merkezli yönlendirmede, ilgi yayılımı, duyarga düğümlerine algılama görevleri atar. İlgilili yayılımına dair iki yaklaşım vardır; birinci yaklaşımda, ilgilenilen veri alıcı düğüm tarafından yayınlanır. İkinci yaklaşımda ise, duyarga düğümleri ellerindeki veriyi yayınlamak sureti ile ilan ederler, ardından ilgilenilen alıcı düğümlerden talep beklerler.

Veri merkezli yönlendirme, özellik tabanlı isimlendirme gerektirir. Özellik tabanlı isimleme, olaya dair özellikleri kullanarak sorgulama yapmaktır. Özellik tabanlı

isimlendirme için kullanıcılar tek bir düğümü sorgulamaktansa olaya dair bir özelliği sorgular. Örnek olarak verilebilecek sorgulardan ilki “70 °C üzeri sıcaklıktaki alanlar” sorgusu, ikincisi ise “x düğümün okuduğu sıcaklık” sorgusu olsun. Bu iki sorgudan ilki bir özellik (alan) ve bu özelliğe dair bir değer (70 °C) belirtir ve ikinci sorguya göre daha fazla kullanılan bir sorgu örneğidir [37].

Veri toplama, veri merkezci yönlendirmelerdeki göçme ve üst üste binme problemlerini çözmeye yönelik bir tekniktir. Bu teknikte, duyarga ağı, ters bir çoğa gönderim ağacı olarak görülür. Burada alıcı düğüm, duyarga düğümlerden olayın durumunu raporlamalarını isteyebilir. Alıcı düğüme doğru yönelmiş ortak bir yolda birleştikleri zaman, bir çok duyarga düğümünden gelen veriler olayın aynı özelliği olarak toplanır.



**Şekil 4.6:** Bir ağ örneği [37]

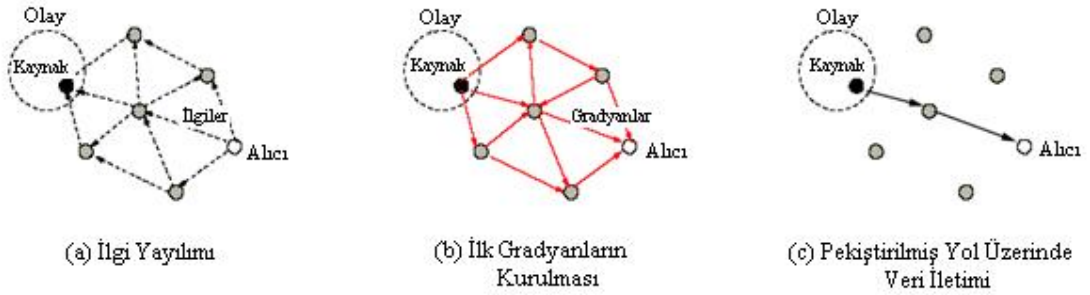
Şekil 4.6’da E düğümü A ile B düğümünden gelen verileri; F düğümü ise C ile D düğümünden gelen verileri toplamaktadır. Veri toplama, birçok düğümden bir anlamlı bilgiler kümesine gelen verilerin, otomatik metod kümelerinde toplanması işlemidir. Veri yollayan duyarga düğümlerinin yerleri gibi veri özellikleri, veri toplarken dikkat edilmesi gereken özelliklerdendir. Bu özellikler bazı uygulamalar için gerekli olabilir [37].

Ağ katmanının bir başka önemli fonksiyonu ise, diğer duyarga ağları, yönetim ve kontrol sistemleri ve internet gibi harici ağlarla birlikte çalışabilme olanağını sağlamaktır. Alıcı düğümün diğer ağlara ağ geçidi olarak kullanıldığı durumlar olduğu gibi, alıcı düğümlerin birlikte omurga oluşturup, bu omurganın bir ağ geçidi aracılığıyla diğer ağlara eriştiği bir durum da olabilir.

#### 4.7.6 Yönlendirmeli Yayılım Algoritması

Directed diffusion, duyurga ağlarının en büyük gereksinimlerinden olan sağlamlık, ölçeklenebilirlik ve etkin enerji kullanımı göz önünde bulundurularak tasarlanmış bir yönlendirme algoritmasıdır. Yönlendirmeli yayılım, veri merkezci bir algoritmadır. Duyurga düğümleri tarafından oluşturulan veri, özellik-değer çifti ile isimlendirilir. Bir düğüm veri isterken, isimlendirilmiş veri için ilgi mesajları gönderir. İlgi mesajına uyan veri, ağ içerisinde bu düğümüne doğru çekilir. Ara düğümler veriyi saklayabilir, değiştirebilir ya da belleklerdeki verilere göre ilgi mesajlarını yönlendirebilirler.

Yönlendirmeli yayılım temel 4 elemandan oluşmuştur: ilgi mesajları, veri mesajları, gradyanlar ve pekiştirmeler.



Şekil 4.7: Yönlendirmeli yayılım genel yapısı [42]

İlgi mesajları, kullanıcının isteğine karşı gelen bir soru veya sorgudur. Her ilgi mesajı, veri elde etmek için duyurga ağı tarafından yapılan algılama işleminin tanımını içerir. Genellikle, duyurga ağlarındaki veri, fiziksel bir olaya dair toplanmış veya işlenmiş veridir. Bu tarz veri, algılanan olayın kısa bir açıklaması gibi düşünülebilir.

Yönlendirmeli yayılımda veri, özellik-değer çifti ile isimlendirilir. Algılama işlemi (ya da alt işlemleri) ağ içerisinde, isimlendirilmiş veriye dair ilgi mesajları olarak yayılır (Şekil 4.7-a). Bu yayılım, ilgi mesajına uyan verilerin kullanıcıya ulaşabilmesi için gradyanlar kurar (Şekil 4.7-b). Gradyan, ilgi mesajını alan her düğüm içerisinde yaratılan bir yön durumudur. Gradyanlar, ilgi mesajının alındığı komşu düğümüne doğru yönlendirilir. Olaylar, ilgi mesajlarını yaratan düğümlere doğru birçok gradyan ve yol üzerinden akmaya başlar. Duyurga ağı, bu yollardan bir

veya (az sayıda olmak kaydıyla) birkaçını pekiştirir ve bu pekişmiş yollar üzerinden veri akışını gerçekleştirir (Şekil 4.7-c).

#### **4.8 Duyarga Ağları İçin Karınca Algoritmaları Tabanlı Yönlendirme Algoritmaları**

Bu bölümde, karınca algoritmalarından esinlenilerek hazırlanmış yönlendirme algoritmalarından bazıları anlatılacaktır. Bu yönlendirme algoritmalarından ilki, temel karınca yönlendirme algoritması olarak tabir edilen ve birçok karınca algoritması tabanlı yönlendirme algoritmasına ilham kaynağı olmuş *AntNet* algoritmasıdır [43]. *AntNet* algoritmasının tasarlanmasından daha önce ortaya atılmış, devre anahtarlama ağılarda çalışmak üzere tasarlanmış karınca kolonisi tabanlı bir yönlendirme algoritması olan karınca tabanlı kontrol (ABC) mevcuttur [44, 45], ancak bu çalışmada sadece paket anahtarlama ağılar üzerinde çalışan algoritmalar inceleneceğinden, devre anahtarlama ağılarda çalışmak üzere tasarlanmış algoritmaların detaylarına yer verilmeyecektir.

*AntNet* algoritmasında kullanılan temel yapıda, kaynak düğümden hedef düğüme doğru yola çıkan ve yol boyunca bilgi toplayan bir karınca ile, hedefe varıldığında yaratılan, hedefe kadar önceki karıncanın geçtiği ara düğümlerden geri giderek kaynak düğüme ulaşan bir ikinci karınca söz konusudur. Hedefte yaratılan ikinci karıncanın görevi, ilk karıncanın yolculuk performansına göre, geçtiği yolun feromon miktarlarını ayarlamaktır.

*AntNet* algoritmasının başlangıcında, kaynak düğümde belli zaman aralıkları ile karıncalar yaratılır ve bu karıncalar hedef düğümü bulmak üzere ağda dolaşmaya başlar. Hedefi bulmak amacı ile ağ üzerinde hareket eden karıncalar, gidecekleri bir sonraki düğümü seçmek için, gidebilecekleri olası düğümler ile buldukları düğüm arasındaki bağlantıların gidilebilme olasılığının dağılımına bakarlar. Algoritmanın başlangıcında, düğümler arasındaki bütün bağlantıların gidilebilme olasılıkları eşittir. Hedefe doğru ilerleyen bütün karıncalar, geçtikleri düğümlerin bir listesini tutarlar. Böylece aynı düğümden bir kez daha geçmezler ve gereksiz döngülere girmezler. Hedefi arayan karıncalardan biri hedefi bulduğunda, hedef ile kaynak düğüm arasında karıncanın geçtiği bağlantıların feromon miktarlarını ayarlamak üzere yeni bir karınca yaratılır ve bu karınca kaynak düğüme doğru yola çıkar. Kaynak düğüme doğru geri dönen karınca, geçtiği düğümlerin hedef düğüme olan

uzaklıklarına göre yolların feromon miktarını günceller. Geri dönen bu karınca kaynak düğüme vardığında, döndüğü yolun başarımına göre karınca yaratma zamanının aralığı ayarlanır.

Kablosuz duyurga ağlarının temel özelliklerinden biri olan asimetrik bağlantılar, *AntNet* algoritmasının kablosuz duyurga ağlarında etkin kullanımını zorlaştırır. Asimetrik bağlantıların temel nedeni, kablosuz duyurga ağlarındaki bütün düğümlerin yayın özelliklerinin aynı olmayabileceğidir. Bunun sonucunda, herhangi bir duyurga düğümü, kendi yayın alanı içerisinde kalan komşularından herhangi birinin yayın alanı içerisinde olmayabilir. Bir düğümün gönderdiği paketi alan komşusu, cevap niteliğinde bir paket gönderdiğinde önceki düğüme ulaşmayabilir. *AntNet* algoritmasında, hedefe ulaşıldığında yaratılan yeni karınca, hedefi bulan karıncanın geçtiği ara düğümlerden geçerek bu düğümlerin feromon miktarlarını ayarlamaktan sorumludur. Ancak bu yöntem, kablosuz duyurga ağlarının asimetrik bağlantı özelliği göz önünde alındığında, gidilen yoldan geri dönülemeyeceği için çalışmayabilir.

*AntNet* algoritmasının duyurga ağları üzerinde çalışabilmesi amacıyla, algoritma üzerinde bazı değişiklikler yapılmış, ve *AntNet* algoritması tabanlı yeni yönlendirme algoritmaları ortaya çıkmıştır.

#### **4.8.1 Duyurga GÜdümlü ve Maliyetten Haberdar Karınca Yönlendirmesi Algoritması**

*AntNet* algoritmasının duyurga ağlarında kullanılması sonucunda karşılaşılan zorluklardan bir tanesi, algoritmanın başlangıç safhalarında karıncaların hedef düğümü bulmak için çok zaman harcamalarıdır. Bu sorunu gidermek için, karıncaların hedefin kokusunu alabildikleri varsayımından yola çıkılarak “Duyurga GÜdümlü ve Maliyetten Haberdar Karınca Yönlendirmesi Algoritması” ortaya çıkmıştır [46]. Algoritmada koşturulan karıncaların hedefin kokusunu alabilmeleri kavramının uygulanabilirliği, duyurga ağları için oluşturulan yönlendirme algoritmalarının özellik-tabanlı olmalarından dolayı mümkün olabilmektedir. Ağ üzerinde aranılan bilgiye dair bir özellik, ya da duyurgaların coğrafi konumu, karıncaların algıladıkları kokunun modellenmesi için kullanılabilir. Pratikte, her düğümün komşusuna olan uzaklığının tutulması, kokunun gerçekleşmesi için uygun olabilir.

Algoritmanın maliyetten haberdar olması kavramı, tatbiki sırasında en kısa yol dışında diğer yönlendirme metriklerinin de kullanılabileceğini belirtmek için eklenmiştir. Duyarga ağlarının özellikleri ve sorunları göz önüne alındığında, eklenebilecek metriklerden en önemlisi yönlendirme için harcanan enerjidir.

#### **4.8.2 Sel Baskını Karınca Yönlendirmesi Algoritması**

*AntNet* algoritması üzerinde yapılan bir diğer çalışmada, algoritmanın başlangıç safhasında hedefin kısa sürede bulunabilmesi için sel baskını algoritmasından yararlanılmıştır. Algoritma başladığında yayınlanan karıncayı alan düğümler tekrardan karınca yayınlamaya, sorguyu ağ üzerinde yayarlar. Hedef düğüm bulunduğu, tıpkı *AntNet* algoritmasında olduğu gibi hedef düğümde yaratılan bir karınca kaynak düğüme doğru geri gelir ve geçtiği düğümlerin feromon miktarlarını günceller [46].

Duyarga güdümlü ve maliyetten haberdar karınca yönlendirmesi algoritmasında bahsedilen karıncaların hedefin kokusunu alabilmeleri özelliği, bu algorithmada da kullanılabilir. Böylece algoritmanın başında sel baskını sırasında, sadece hedef düğüme yakın olan düğümler sel baskınına katılır ve böylece haberleşme maliyeti azaltılmış olur.

#### **4.8.3 Sırtında Taşınmalı Sel Baskını Karınca Yönlendirmesi Algoritması**

Bu algorithmada, hedef düğümü arayan ve hedef düğümden kaynak düğüme geri gelip feromon miktarlarını güncelleyen karıncalar birleştirilmeye çalışılmıştır. Sel baskını yöntemi ile hedefi arayan karıncaları, geçtikleri düğümlerin bir listesini de beraberlerinde taşırlar. Hedef düğüm bulunduğu yaratılan karınca bu listede belirtilen düğümlerden kaynak düğüme geri döner ve geçtiği düğümlerin feromon miktarlarını günceller [46].

Hedef düğümü arayan karıncaların taşıdıkları liste, sel baskını algoritmasının zaten yüksek olan haberleşme maliyetini daha da arttırmakla birlikte, algoritmanın başarımının artmasına önemli katkıları olmuştur.

## 5. ANCOR

### 5.1 Giriş

ANCOR, kablosuz duyurga ağlarında kullanılmak üzere geliştirilmiş yeni bir yönlendirme algoritmasıdır. Ancor ismi, ingilizcede *Ant Colony Routing* kelimelerinin kısaltılması ile oluşturulmuştur. Algoritmanın temelinde yatan fikir, doğadaki karıncaların yiyecek bulmada ya da yuvaları için yeni (veya daha iyi) bir mekan aramada sergiledikleri davranışları mümkün olduğunca taklit etmektir.

Karınca kolonilerinde, çok sayıda birey ortak bir amaç için birlikte çalışırlar. Amaç ne olursa olsun birlikte düzenli bir biçimde çalışırlar ancak gerçekte bireylere ne yapmaları ve nasıl davranmaları gerektiğini anlatan global bir yönetici birey yoktur. Her karınca çevreyi takip ederek kazandığı algılar doğrultusunda bir takım basit kuralları izleyerek hareket eder, ve salgıladığı feromon maddesi sayesinde diğer karıncalar ile dolaylı yoldan iletişim kurar. Bu davranış modelinin detayları ve dolaylı iletişim yolları daha önceki bölümlerde detaylı olarak anlatılmıştır.

Tasarsız ağlardan farklı olarak duyurga ağlarında daha fazla sayıda düğüm bulunur. Kalabalık karınca kolonilerinde görülen davranışların, duyurga ağlarında yönlendirme algoritması olarak kullanılabileceği düşüncesi de, kalabalık karınca kolonilerindeki birey sayısının da tıpkı duyurga ağlarındaki gibi fazla sayıda olmasından ileri gelmektedir. Doğada ayrıca görece az sayıda birey içeren karınca kolonileri de bulunmaktadır. Bu koloniler yiyecek ve barınak aramada zincirleme koşu adı verilen başka bir yöntem kullanırlar. Bu yöntemin görece az sayıda düğüm içeren tasarsız ağlarda yönlendirme algoritması olarak kullanılması düşünülebilir.

Bu bölümde öncelikle ANCOR algoritmasının genel özelliklerinden, ANCOR'da kullanılan feromonun özelliklerinden ve koku dağılım modellerinden bahsedilecektir.



## 5.2 Algoritmanın Genel Özellikleri

### 5.2.1 ANCOR ve Diğer Yönlendirme Algoritmaları

Karınca algoritmalarının, yönlendirme algoritması olarak ve özellikle duyurga ağlarında yönlendirme algoritması olarak kullanıldığı bazı örnek çalışmalardan önceki konularda bahsedilmiştir. Temel yönlendirme algoritması Di Caro ve Dorigo tarafından ortaya atılan *AntNet* algoritmasıdır [43]. Zhang ve arkadaşları [46], mevcut karınca kolonisi tabanlı yönlendirme algoritmalarının duyurga ağlarında iyi sonuç vermediğini ortaya koymuşlardır. Bunun nedeni, mevcut algoritmaların ağ içerisinde hedef düğümü bulmada çok vakit kaybetmeleridir. Ayrıca duyurga ağlarında var olabilecek asimetrik bağlantılar algoritmaları uygulanamaz kılmaktadır.

Mevcut karınca kolonisi tabanlı yönlendirme algoritmalarının hepsinde, karıncalar düğümlerden geçerken, ziyaret ettikleri düğümleri içeren bir liste taşırlar, ve yolculuklarının sonunda geçtikleri yolu oluşturan düğümlerin feromon miktarlarını güncellerler. ANCOR algoritmasındaki karıncalar, önceki yönlendirme algoritmalarının aksine ziyaret ettikleri düğümlerin bilgilerini içeren bir liste taşımazlar. Bunun yerine sadece bulunduğu bir önceki düğümün bilgisini tutarlar, ve her adımda ziyaret ettikleri düğümlerin feromon miktarlarını güncellerler. Bu özellikler, ANCOR algoritmasının doğayı daha gerçekçi modellemesini sağlamakla kalmaz, ayrıca haberleşmede kullanılan veri paketlerinin diğer algoritmalara göre daha küçük boyutlu olmasını da sağlar. Kullanılan paketlerinin küçük boyutta olmaları, pratikte duyurga ağının paket yayınlama süresinin kısılmasını sağlar. Böylece, duyurga ağlarında karşılaşılan en büyük sorunlardan biri olan, güç tüketiminin azaltılması konusunda kolaylık sağlanır. Bunun neticesinde, duyurga ağının net yaşam süresinin uzamasına katkıda bulunulacağı öngörülmektedir.

Ağ üzerinde hareket eden her karınca, sadece geçmiş olduğu düğüm sayısını, hedefi ilk bulan karınca olup olmadığını gösteren bir bayrağı, bir önceki ve bir sonraki düğüm bilgisini ve ağ üzerinde aradığı ya da taşıdığı bilgiyi içerir. Karıncanın aradığı bilgi, bizim yayınladığımız sorgudan ibarettir. Bu bilgi, karıncanın geçtiği düğümlerin aradığımız hedef düğüm olup olmadığını anlaşılması için kullanılmaktadır. Ağ üzerinde ilerleyen karıncalar, önceki karıncaların bıraktıkları feromon miktarlarını mukayese ederek gidecekleri bir sonraki düğümü tespit

ederler. Eđer algoritma yeni alıřtırılmıřsa ve henüz ađ zerinde ilerlemiř herhangi bir karınca olmamıřsa, dđmlerin herbirinin feromon yođunluđu sıfır (ya da ntr) kabul edilir. Bu durumda ađ zerinde yeni dolařmaya bařlamıř bir karınca iin btn komřu dđmlerin gidilebilme olasılıkları eřit olur.

### 5.2.2 Eřleřtirme

Bu blmde ANCOR'da kullanılan mekanizmalar ile dođadaki ve dođal karınca kolonilerindeki mekanizmalar eřleřtirilecek, hangi nesnenin dođadaki neye karřı geldiđi daha detaylı anlatılacaktır. Bahsedilecek olan kavramlar ve kavramların dođadaki karıřlılıklarının listesi Tablo 5.1'de verilmiřtir.

**Tablo 5.1:** Dođadaki ve ANCOR'daki kavramların eřleřtirilmesi

Dođadaki Kavramlar	ANCOR'daki Kavramlar
Karınca	Veri Paketi (ya da Sorgu)
Yuva	Alıcı Dđm
Yiyecek Kaynađı	Kaynak Dđm
Yol (Patika)	Patika

Alıcı dđmden kaynak dđme dođru ele alınan bir senaryoda, alıcı dđmde yaratılan bir sorgu ađ zerinde yayılır. Dđmlerden elde edilen veriler ıkıř dđmnde toplanırlar. Buna gre duyurga ađlarındaki alıcı dđm dođadaki karınca yuvasına benzetebiliriz nk dođada da karıncalar yuvalarından ıkıp yiyecek aramaya bařlarlar ve buldukları yiyecekleri tekrar yuvalarına tařırlar. Karıncaların aradıkları yiyecek kaynakları, duyurga ađlarında aranan bilgiyi barındıran dđmlere, yani kaynak dđmlere karřı gelmektedir.

ANCOR'da her bir karınca, ađ zerinde yol alan bir sorguya ya da veri paketlerine karřı gelmektedir. Ađ zerindeki her bir duyurga dđm, dođada karıncaların hareket ettikleri yolun bir parası olarak ele alınabilir, buradan yola ıkarak duyurga dđmleri feromon bilgisini tařımak zorundadırlar ki dođadaki feromon izleri sanal olarak duyurga ađında oluřturulabilsin. Dođadaki karıncaların takip ettikleri yol, ANCOR'da birka duyurda dđm tarafından oluřturulmuř patikalar olarak ifade edilmektedir.

Doğadaki feromon kavramı, ANCOR'daki yönlendirme kavramının temelini oluşturmaktadır. Ağdaki her düğümün komşu düğümlerinin feromon miktarlarını tutan bir listesi vardır, böylece bu düğüme gelen bir karınca, yolculuğunun bir sonraki adımında gideceği düğümü bulabilmek için bu listeden yararlanır. Bu kavram, doğadaki karıncaların feromonu koklayarak gideceği yeri tespit etmesi ile eşleştirilebilir. Bir düğümün feromon miktarı, o düğümün ağda dolaşan karıncalar tarafından ziyaret edilebilme olasılığını temsil eden bir parametredir. Doğadaki kimyasal feromonun taşıdığı özellikler (yoğunluk, buharlaşma), ANCOR'daki feromonda da aynı şekilde muhafaza edilmiştir.

### **5.3 Feromon**

Doğadaki feromon, çeşitli hayvanlar tarafından salgılanan kokulu bir kimyasaldır. Karınca kolonilerinde feromon, bireyler arasında dolaylı olarak iletişim kurmak amacıyla salgılanır. Yiyecek kaynağını arayan karınca, feromonu takip ederek yiyecek kaynağına ulaşır, ve aynı kokuyu takip ederek yiyeceği yuvasına taşır. Bu işlem sırasında ardından gelen karıncalar da yolu takip edebilsin diye feromon bırakırlar.

Bu bölümde ANCOR algoritmasında kullanılan feromon için çeşitli dağılım modelleri anlatılacak, daha sonra nötr feromon ve olumsuz feromon kavramlarından bahsedilecektir.

#### **5.3.1 Koku Dağılım Modeli**

Bu bölümde ağda dolaşan karıncaların düğümlere bıraktıkları feromonun ağda dağılımını temsil eden modeller anlatılacaktır. Karınca bir düğüm üzerine geldiğinde o düğüme belli bir miktar feromon bırakır ve düğümün feromon miktarı değiştirilmiş olur. Düğüm bu yeni feromon miktarını yayınlamak için komşularının haberdar olmasından sorumludur. Düğümlerdeki feromon miktarları buharlaşma yöntemi ile de değişmektedir ancak ağa aşırı yayın yükü getireceğinden buharlaşma sonucu değişen feromon miktarları komşulara bildirilmez. Ağdaki her düğüm, komşularının feromon yoğunluklarını tuttuğu listedeki feromon değerlerini düzenli olarak buharlaştırmaktan sorumludur.

### 5.3.1.1 Temel Koku Dağılım Modeli

Bu modelde, bir düğümden yayılan koku sadece o düğümün yayın mesafesi içerisindeki komşu düğümler tarafından algılanır ve komşu düğümlerin komşularına iletilmez. İki düğüm arasındaki mesafe ne olursa olsun, komşu düğümden gelen feromon miktarı olduğu gibi alınır, yani komşulardan algılanan feromon üzerinde mesafeye bağlı bir azalma mekanizması işletilmez. Bu modelin uygulanması çok kolaydır çünkü düğümlerin birbirlerine olan uzaklıkları hesaba katılmaz. Ancak doğadaki feromonun yayılım modelini gerçekçi olarak temsil eden bir model olmaktan uzaktır.

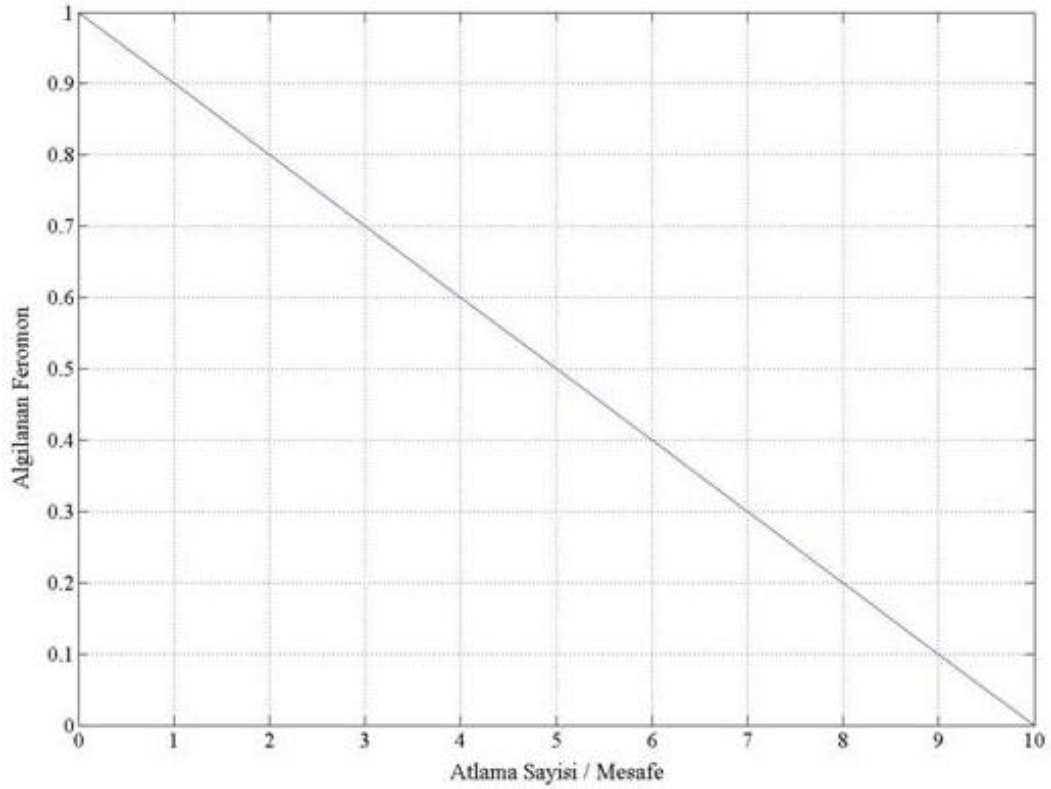
### 5.3.1.2 Doğrusal Koku Dağılım Modeli

Bu model, feromonun düğümler arası yayılımını, temel koku dağılım modelinden daha iyi temsil etmesine karşın, düğümlerin birbirlerine olan uzaklıklarının bilinmesini gerektirir (Formül 5.1, Şekil 5.1). Bunun için duyarga düğümlerinin üzerinde dahili yer belirleme sistemlerinin (GPS) bulunması gerekmektedir. Duyarga düğümleri üzerinde herhangi bir yer belirleme sistemi bulunmaması durumunda, ağ içerisinde yayınlanan feromon paketlerinin içeriği değiştirilerek, aslen bırakıldığı düğümden olan uzaklığı kesitirebilir. Bunun için feromon paketi içerisine geçtiği düğümlerin sayısını belirten bir değişken konulabilir.

Formül 5.1’de  $x$  değişkeni, feromonun bırakılmış olduğu düğüm üzerindeki miktarını belirtir.  $maxMesafe$  değişkeni, herhangi bir düğüme bırakılmış olan feromonun ne kadar uzak mesafeden algılanabileceğini ayarlayan değişkendir.  $mesafe$  ile belirtilen ise feromonun algılandığı uzaklığı gösterir. Fonksiyonun sonucunu belirten  $y$  değişkeni ise algılanan feromon miktarını belirtir.

$$y = x \frac{maxMesafe - mesafe}{maxMesafe}$$

(5.1)



**Şekil 5.1:** Doğrusal koku dağılım modeli örneği

Bu model için GPS kullanımının dışında bir başka gerçekleştirim önerisi ise, feromon paketi içerisine geçtiği düğümlerin listesinin konulması olabilir. Bu listenin uzaklık bilgisini elde etmek açısından daha kesin veriler içereceği öngörülse bile, paket boyutları büyüyeceğinden, haberleşme üzerine fazladan yük bindireceği kesindir.

Şekil 5.1’de görüldüğü üzere bu modelde algılanan feromon miktarı, feromonun salgılanmış olduğu düğümden uzaklaşıldıkça doğrusal olarak azalmaktadır. Şekil 5.1’de  $maxMesafe = 10$  ve  $x = 1$  olarak alınmıştır.

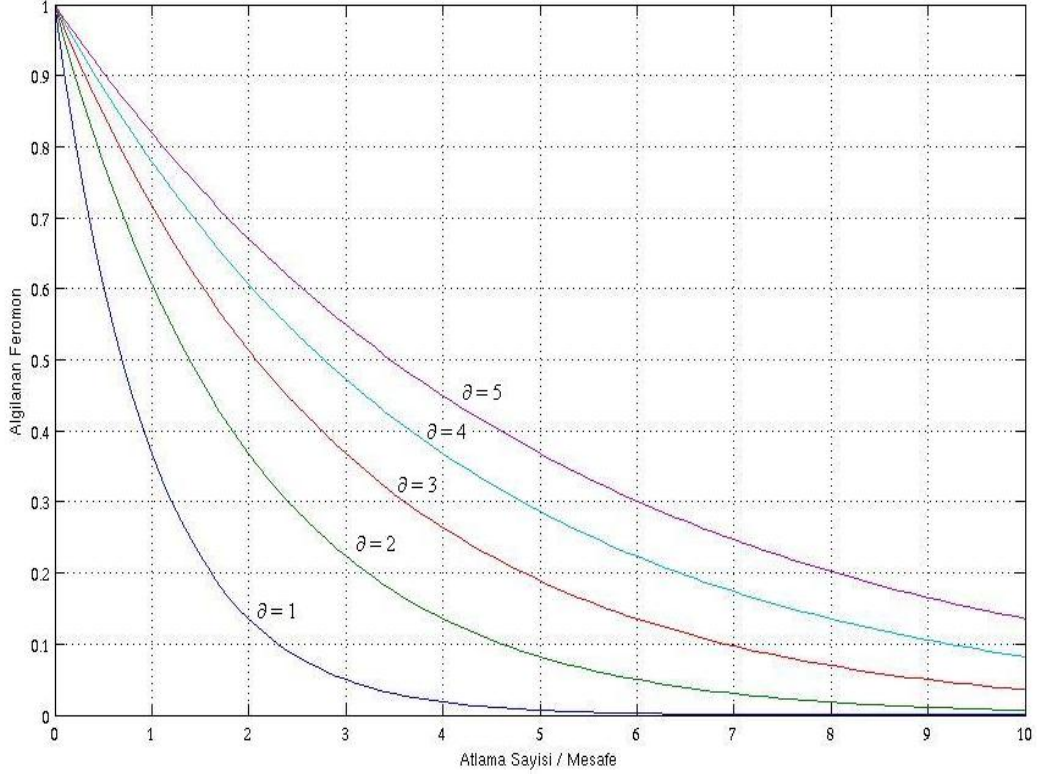
### 5.3.1.3 Üstel Koku Dağılım Modeli

Bu modelde, kokunun düğümler arası yayılım fonksiyonu, üstel bir fonksiyon olarak ele alınmıştır (Formül 5.2, Şekil 5.2).

$$y = x e^{-\frac{mesafe}{d}} \quad (5.2)$$

Formül 5.2’de  $x$  değişkeni feromonun miktarını gösterir.  $mesafe$  ile gösterilen değer iki düğüm arasındaki mesafeyi belirtirken,  $d$  değişkeni bir pozitif doğal

sayıdır ve uygulamaya yönelik ince ayar yapmak için kullanılmaktadır.  $\theta$  değişkeninin değeri büyüdükçe, artan mesafelerde algılanan feromon miktarı daha fazla olacaktır. Fonksiyonun sonucunu belirten  $\gamma$  değişkeni ise uzaktaki düğüme ulaşan toplam feromon miktarını, yani algılanan feromonu belirtir.



**Şekil 5.2:** Üstel koku dağılım modeli örneği

Tıpkı doğrusal koku dağılım modelinde belirtildiği gibi, üstel koku dağılım modelinin gerçek duyarga ağlarında uygulanabilmesi için donanımsal bir yer belirleme cihazına, ya da paket ve algoritma içeriğinde bazı değişiklikler yapılmasına ihtiyaç vardır. Bu değişiklikler, paket içerisine geçilen düğümlerin listesinin eklenmesi, ya da geçilen düğüm sayısının eklenmesi olabilir.

Şekil 5.2’de görüldüğü gibi, bu modelde algılanan feromon miktarı, feromonun bırakıldığı düğümden uzaklaştıkça üstel olarak azalmaktadır. Şekil 5.2’de  $\theta = 1$  ve  $\gamma = 1$  olarak alınmıştır.

### 5.3.2 Sıfır (Nötr, Null) Feromon

Sıfır ya da nötr feromon, aslında düğümden feromon olmadığını belirten bir ifadedir. Algoritmanın başlangıç safhası haricinde bu duruma ulaşılabilmesi için ya duyarga

düğümünün feromon yoğunluğunun herhangi bir karınca tarafından değiştirilmemiş olması, ya da düğümün feromon yoğunluğunun buharlaşma yardımı ile sıfırlanması gerekmektedir. Ağa yeni katılan düğümlerin başlangıç feromon miktarları sıfır feromon olarak belirtilir.

### **5.3.3 Olumlu (Çekici) Feromon**

Olumlu feromon olgusu, doğadaki karıncaların kullandıkları feromondan esinlenilerek çekici bir etkiye sahip olarak ANCOR'da kullanılmaktadır.

### **5.3.4 Olumsuz (İtici) Feromon**

ANCOR algoritmasının doğal karınca kolonilerinin davranışlarından ayrıldığı en önemli nokta olumsuz feromon olgusudur. Doğadaki karıncaların salgıladıkları feromon, çekici bir etki yaparak takip eden karıncaları kendine çeker. Böylece karıncalar aynı yolu izleyerek yuvalarına ya da yiyecek kaynaklarına ulaşırlar. ANCOR algoritmasında ortaya atılan olumsuz feromon ise, normal feromonun aksine itici bir etkiye sahiptir. Böyle bir olgunun ortaya atılma amacı, özellikle algoritmanın başlangıç safhasında, karıncalar ağ üzerinde hedef düğümü aralarken, mümkün olduğu kadar etkin bir arama yapabilmektir. Buna göre ANCOR algoritmasının başında, başlangıç düğümünde yaratılan karıncalar ağ üzerinde dolaşmaya başlarken olumsuz feromon salgırlar. Böylece i. karıncanın gittiği düğüme i+1. karıncanın gitme olasılığı azaltılmış, ve karıncalar ağ üzerine daha etkili bir biçimde yayılmış olur.

#### **5.3.4.1 Anti Feromon ve Olumsuz Feromon**

Aramayı daha etkin kılmak için olumsuz feromon benzeri bir yapı Montgomery ve arkadaşları tarafından *Anti-Pheromone* olarak ortaya atılmıştır [47], ancak ANCOR'da kullanılan feromon yapısı, hem çekici hem itici hem de nötr etkiyi bünyesinde barındırdığı için diğer feromon gerçekleştirmelerinden farklıdır.

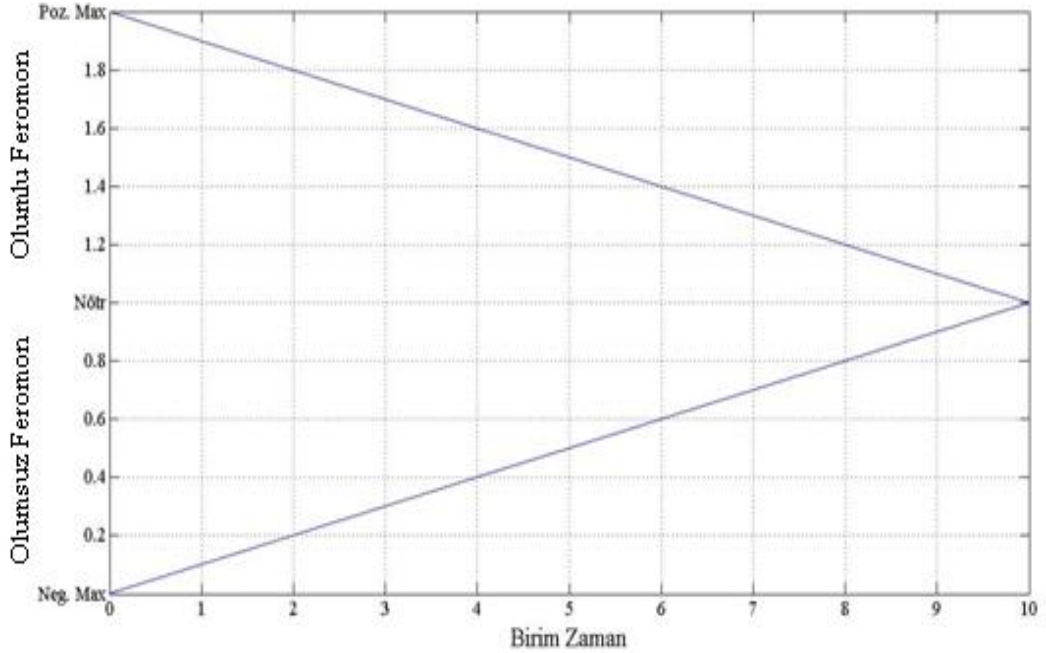
### **5.3.5 Buharlaşma**

Karınca kolonisi optimizasyonunda olduğu gibi, ANCOR algoritmasının da en önemli bileşenlerinden bir tanesi buharlaşmadır. Olumlu feromonun yanı sıra, algoritmanın dinamiklik özelliğini sağlayan bir diğer temel mekanizma olan buharlaşma, doğal hayatta da karıncaların yiyecek kaynaklarına giden kestirme yolları daima bulmaları için kullandıkları bir mekanizmadır.

$$x_{t=i} = \begin{cases} x_{t=i} + (\delta \times i) & x > \text{nötr feromon} \\ x_{t=i} - (\delta \times i) & x < \text{nötr feromon} \end{cases} \quad (5.3)$$

Formül 5.3'de görüldüğü gibi,  $f = 0$  anında bir düğüme bırakılan feromon,  $i$  birim zaman süresince  $\delta$  ile gösterilmiş olan buharlaşma katsayısı tarafından doğrusal bir biçimde azaltılır. Bu azalma işlemi, düğümün feromon yoğunluğunun nötr feromondan farklı olduğu sürece ( $x \neq \text{nötr feromon}$ ) devam etmektedir.

Şekil 5.3'de verilen buharlaşma örneğinde görüldüğü gibi, düğüm üzerindeki feromon olumlu feromon da olsa olumsuz feromon da olsa, güncellenmediği sürece nötr feromona doğru yaklaşmaktadır. Şekil 5.3'de  $\delta = 0.1$  olarak alınmıştır.



Şekil 5.3: Buharlaşma örneği

### 5.3.6 Feromon Bombası

Önceki bölümlerde de bahsedildiği gibi, karıncalar bazı olağan dışı durumlarda vücutlarındaki feromonun tümünü bırakırlar. Bu işleme feromon bombası denilir ve doğadaki uygulaması bir çeşit yardım çağrısı niteliğindedir.

ANCOR'da da feromon bombası kullanılmaktadır ancak, ANCOR'da iki çeşit feromon olduğu için iki çeşit feromon bombası vardır. Olumlu feromon bombası, bir düğümün feromon miktarını olabilecek en yüksek seviyedeki olumlu feromona eşitlerken, olumsuz feromon bombası ise, düğümün feromon miktarını olabilecek en yüksek seviyedeki olumsuz feromona eşitlemektedir.



## 5.4 Yönlendirme ve Güncelleme Mekanizmaları

$$komşu_i \text{ olasılığı} = (\beta \times komşu_i \text{ feromonu}) + (1 - \beta) \text{rastgele}_{sayı} \quad \beta \in [0..1] \quad (5.1)$$

ANCOR algoritmasında, karıncalar gidecekleri bir sonraki düğümü seçmek için komşu düğümlerin feromon miktarlarına bakar (Formül 5.1). Karıncalar her ne kadar olumlu feromona gitme eğilimi ve olumsuz feromondan kaçınma eğilimi gösterebilirler de, belli bir miktar karınca rastgele düğümler seçerek yollarına devam ederler. Formül 5.1’de,  $\beta$  parametresi karıncaların en sık kullanılan düğümlere gitme eğilimini ayarlayan bir parametredir. Buna göre  $\beta = 1$  için karıncalar devamlı en fazla feromona sahip olan düğüme, yani en sık kullanılan yola yöneleceklerdir.  $\beta = 0$  için karıncalar rastgele düğümlere gideceklerdir ve feromon bilgisini kullanmayacaklardır. Listedeki her komşu düğüm için Formül 5.1 işletilir ve sonuçta gidilme olasılığı en fazla olan komşu düğüm bulunur.

$$\pm \Delta = \frac{\max \text{Atlama} - \text{atlama}}{\max \text{Atlama}} \rho \quad \rho \in [0..1] \quad (5.2)$$

Formül 5.1’e göre belirlemiş olduğu düğüme gelen karınca, Formül 5.2’ye göre geldiği düğümün feromon miktarını günceller. Eğer düğüm olumsuz feromona sahipse, güncelleme miktarını belirten  $\Delta$  değişkeni eksi olarak alınır. Formül 5.2’de gösterilen  $\rho$  değişkeni,  $\Delta$  değişkeninin etkinliğini ayarlar, yani düğümün karıncaların bıraktığı feromondan ne kadar etkilendiğini belirtir.

İlerleyen alt başlıklarda, ANCOR algoritmasını oluşturan bu üç aşama daha detaylı olarak anlatılacaktır. Anlatımın tek alıcı tek kaynak senaryosuna göre tasarlandığı hatırlatılır.

## 5.5 Kullanılan Paketler

### 5.5.1 Karınca Paketi

ANCOR algoritması çerçevesinde, bir duyurga ağına yapılan sorgu paketlerine ve sorgunun cevabını barındıran hedef düğümden cevabın geri gelmesini sağlayan paketlere *karınca* adı verilmektedir. Duyurga ağına bir sorgu yapıldığında, tıpkı gerçek dünyadaki karıncaların yuvalarından çıktıkları gibi, ANCOR’daki karıncalarda başlangıç düğümünde yaratılırlar ve ağ üzerinde hedef düğümü

aramaya başlarlar. ANCOR'da bu arama işi için kullanılan karınca paketlerinin özellikleri Tablo 5.2'de gösterilmektedir.

**Tablo 5.2:** Ancor karıncasının veri yapısı

<b>Ancor Karıncası</b>
önceki düğüm id
sonraki düğüm id
geçtiği düğüm sayısı
sorgu / veri
hedefi ilk bulan karınca mı?

Tablo 5.2'de görüldüğü gibi, karıncalar geçtikleri düğümlerin tümünü saklayacak bir veri yapısına sahip değillerdir. Bunun yerine sadece buldukları bir önceki düğümün tanımlayıcısına (ID) sahiplerdir. Bu bilginin tutulmasının nedeni, karıncanın iki düğüm arasında gidip gelmek suretiyle kısıt döngüye girmesini engellemektir. Oluşması muhtemel daha büyük döngülere karşı, karıncaların geçtikleri düğüm sayısı saklanmaktadır. Karınca yeni bir düğüme geldiğinde bu sayı artırılır. Uygulama sırasında, karıncaların geçebileceği en fazla düğüm sayısı belirlenir, ve bu sayıdan daha fazla düğüm dolaşmış karıncalar ağdan silinir. Karıncaların geçebileceği en fazla düğüm sayısı uygulamaya göre değişkenlik göstermesine karşın, bu sayıyı belirlemede kullanılacak en önemli parametre ağdaki toplam düğüm sayısıdır.

Karınca paketi içerisinde, sonraki düğüm belirleyicisi, karıncanın gideceği bir sonraki düğümünün hangisi olduğunu belirtir. Karıncalar bir düğüme geldiklerinde, düğümün komşularının feromon miktarlarını incelenerek karıncanın gideceği bir sonraki düğüm bulunur ve bu düğümün belirleyicisi pakete eklenir. Kablosuz duyarga ağlarında düğümler yayınlama yöntemi ile haberleşikleri için, yayını yapan düğümün kapsama alanı içerisindeki komşular (eğer alıcı-verici üniteleri açık ise) paketi alacaklardır. Bu noktada paketi alan düğüm, içerisindeki hedef düğüm bilgisiyle kendi belirleyicisini karşılaştırır ve hedef düğümün kendisi olup

olmadığını kontrol eder, eğer hedef kendisi ise gerekli işlemleri yaparak karıncanın yoluna devam etmesini sağlar, aksi halde paketi yok eder.

Ancor algoritması içerisinde, gönderilen paketlerin alıcısına ulaştığının garanti edecek bir mekanizma eklenmemiştir, gönderilen paketin hedefe ulaştığı varsayılmaktadır. Ancak bu tip bir mekanizma, paket gönderimi ardından paketin alındığına dair bir cevap gönderilerek oluşturulabilir.

### 5.5.2 Feromon Paketi

Ancor algoritmasında kullanılan ikinci paket ise feromon paketidir. Duyarga düğüme gelen bir karınca tarafından düğümün feromon miktarı değiştirilirse, düğüm yeni feromon bilgisini, feromon paketi yayınlamak komşularına bildirir.

**Tablo 5.3:** Ancor feromonu veri yapısı

Ancor Feromonu
feromonun salgılandığı düğüm id
feromonun yoğunluğu

Yayımlanan feromon paketini alan düğümler, paket içeriğindeki feromonun bırakıldığı düğümün belirleyicisine (ID) bakarlar, ve bu belirleyiciye sahip düğümün, komşu düğümler listesinde var olup olmadığı kontrol ederler. Eğer düğüm listede yoksa, ya duyarga ağı yeni işlemeye başlamıştır ve düğümler komşuluk listelerini daha tamamlamamışlardır, ya da ağa yeni katılan bir düğüm vardır ve bu düğüm komşuları ile ilk kez iletişime geçmektedir. Komşu düğümler listesi kontrol edilir, eğer düğüm listede yoksa feromon miktarı bilgisiyle listeye eklenir, eğer listede mevcutsa, sadece feromon miktarı bilgisi güncellenir.

Düğümlerin feromon miktarları, buharlaşma mekanizması ile zamana bağlı olarak değişmektedir. Her değişimden sonra düğüm yeni feromon bilgisini yayınlamaz, bunun yerine her düğüm, komşu düğümlerinin feromon miktarlarını tuttuğu liste üzerinde buharlaştırma işlemini yürütür.

## 6. ANCOR'UN SAFHALARI

ANCOR algoritması, doğadaki bir karınca kolonisinin gözlenen, ikinci ve üçüncü bölümde anlatılan mekanizmalarının temelinde oluşturulmuştur. Bu bölümde yapılan açıklamalarda, Tablo 5.1’de belirtilen eşleştirme uyarınca, doğadaki olaylardan esinlenmeyi pekiştirmek için, eşdeğer sözcükler değişmeli olarak kullanılmıştır. Ağdaki her bir düğüm komşu düğümlerin feromon miktarlarını içeren bir liste barındırır. Böylece düğüme gelen bir karınca “kokladığı” feromonun miktarına göre gideceği bir sonraki düğümü seçer. Karınca yeni bir düğüme geldiğinde, düğümün feromon miktarını günceller ve düğüm bu yeni feromon bilgisini yayınlayıp komşularına bildirir. Bilgiyi alan komşu düğümler kendi listelerini günceller. Her düğüm kendi listesi üzerinde buharlaştırma mekanizmasını yürütmekten sorumludur. Böylece buharlaşma ile değişen feromon bilgisinin yayınlanması gerekmez.

ANCOR algoritması, üç safhalı olarak tasarlanmıştır: başlangıç, pekiştirme ve yönlendirme.

Başlangıç safhasındaki temel amaç, alıcı düğüme oluşturulan sorgunun, kaynak düğümü aramalarını daha etkin kılmaktır. Bunun için, sorguların ağ üzerine mümkün olduğunca yayılmaları hedeflenir. Bu mekanizma doğada karıncaların yuvalarından çıkmasına karşı düşer. Ağ üzerinde dolaşan karınca yeni bir yiyecek kaynağı bulduğunda, yani ağ üzerinde yapılan sorguya karşı düşen bilgileri içeren kaynak düğüme ulaştığında, pekiştirme safhası başlar. ANCOR algoritmasında kaynak düğüme ilk ulaşan karınca, kaynak düğüm ile alıcı düğüm arasındaki yolu

pekiştirmekle görevlidir. Alıcı ile kaynak düğümler arasında pekiştirilmiş bir yol oluşturulduğunda da, yönlendirme safhası başlar.

ANCOR içindeki bu safhaların içerikleri ve sıraları, sabit, tek alıcı tek kaynak probleminin çözümüne yönelik olarak gerçekleştirilmiştir. Örneğin hareketli hedefi takip etmek gibi bir senaryoda, hedef düğümün yeri sürekli değişeceğinden, yönlendirme safhası içerisinde de ayrıca pekiştirme safhaları olacaktır. Benzer şekilde birden fazla alıcının ya da birden fazla kaynağın sözkonusu olduğu senaryolarda, algoritmanın safhalarının ve bu safhaların içeriklerinin uygulamaya uygun bir biçimde değiştirilmesi gerekecektir.

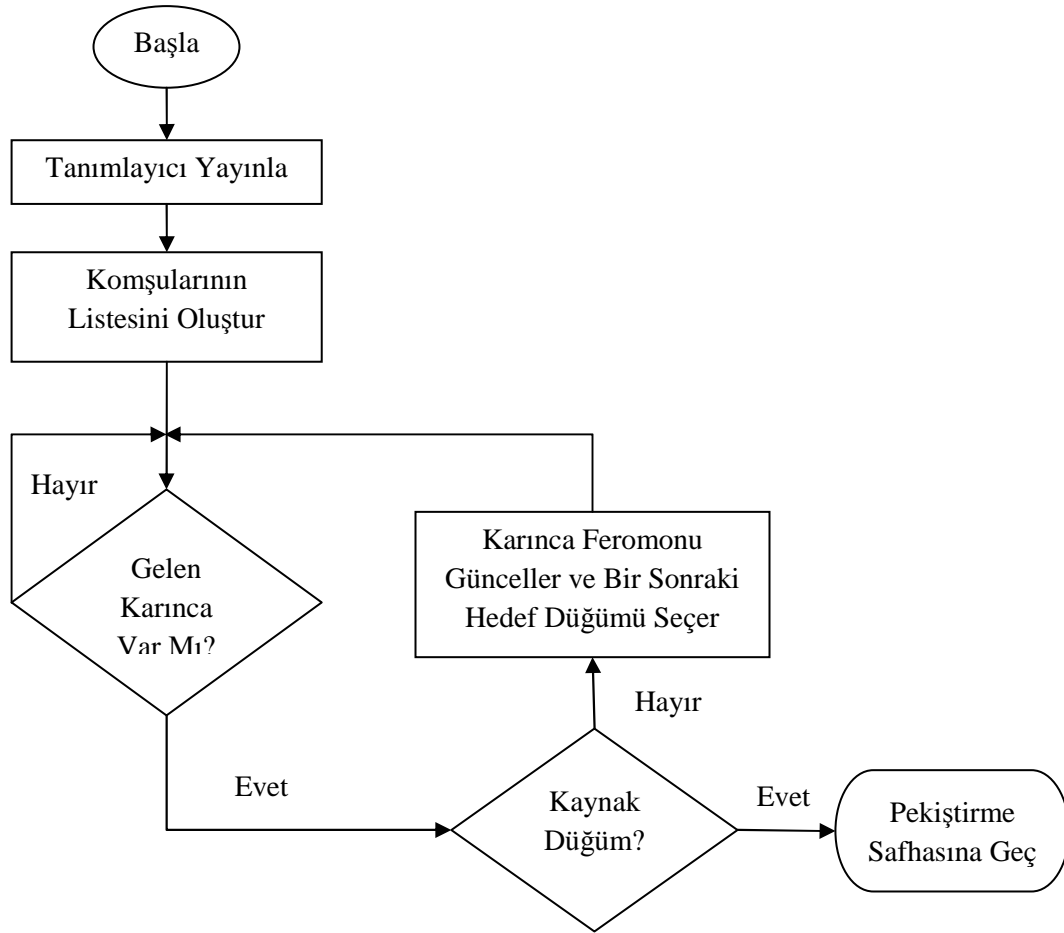
### **6.1 Başlangıç Safhası**

Dağılmayı sağlayan bu safhanın başlangıç koşulu olarak, ağ üzerinde hiçbir karıncanın bulunmadığı, bu nedenle bütün düğümlerdeki feromon miktarının nötr feromona eşit olduğu kabul edilmiştir. Bütün düğümlerin nötr feromona sahip olması, karıncaların dağılımında tüm düğümlerin eşit başlangıç koşullarına sahip olmasını sağlar.

ANCOR algoritmasındaki düğümlerin komşularının feromon miktarlarını tutukları listeleri ilk olarak başlangıç safhasında oluşturulur (Şekil 6.1). Bunun için her yeni çalışmaya başlayan düğüm, tanımlayıcısını (ID) içeren bir paket yayınlara, ve bu paketi alan düğümler kendi listelerini güncellerler. Tanımlayıcıların çakışmaması, katılan her düğümün özgün (biricik) bir tanımlayıcıya sahip olması kabulü nedeniyle değerlendirilmemiştir.

Bu safhadaki karıncalar, ağdaki düğümlerin hepsi sıfır (nötr) feromona sahip olduğundan, bir sonra gidecekleri düğümleri rastgele olarak seçme eğilimindedirler. Tanımlanan mekanizmada, karınca gideceği bir sonraki düğümü seçmek üzere komşu düğümlerin feromon yoğunluklarının tutulduğu listeyi inceler. Listedeki değerlerin aynı olduğunu, nötr feromona eşit olduğunu görüldüğünde hedef düğüm seçimi rastgele yapılmış olur. Olay alıcı düğüm (sink) açısından değerlendirildiğinde, sadece ilk karınca tamamen rastgele bir seçim yapar. Çünkü takip eden bir sonraki karınca seçimini yaparken, ilk karınca nedeniyle ilgili komşu düğümün feromon miktarı değiştirilmiş olacaktır. Özellikle büyük ağlarda az miktarda karınca ile yürütülen senaryolarda, karıncalar ilerledikçe rastgele seçimler yapacakları öngörülür.

Karıncanın yeni bir düğüme geldiğinde, öncelikli olarak geldiği düğümün kaynak düğüm olup olmadığına bakılır. Eğer gelen düğüm kaynak düğüm değilse, karıncanın düğümün feromon miktarını günceller ve gideceği bir sonraki düğümü belirler. Feromon miktarı değiştirilen düğüm bu yeni değeri yayınlar ve komşuları listelerini günceller.



**Şekil 6.1:** ANCOR algoritması başlangıç safhası akış diagramı

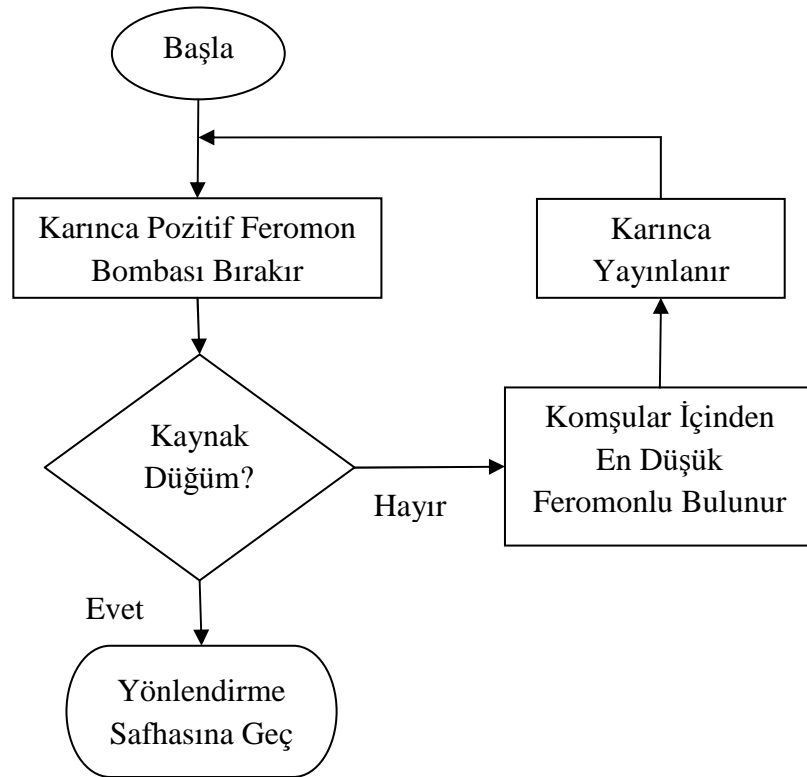
Başlangıç safhasında feromon güncelleme işlemi büyük önem taşır, çünkü karıncaları ağ üzerinde etkin bir biçimde yayabilmek için kullanılacak yegane unsur düğümlerin feromon miktarlarıdır.  $i$ . düğümü ziyaret eden bir  $k$ . karınca var ise,  $(k+1)$ . karıncanın  $i$ . düğümü ziyaret etmemesini sağlamanın tek yolu,  $i$ . düğümün feromon özelliğini olumlu değil olumsuz kılmaktır. ANCOR algoritmasında bunu gerçekleştiren mekanizma olumsuz feromondur.

Başlangıç safhası kaynak düğüm bulununcaya dek devam eder.

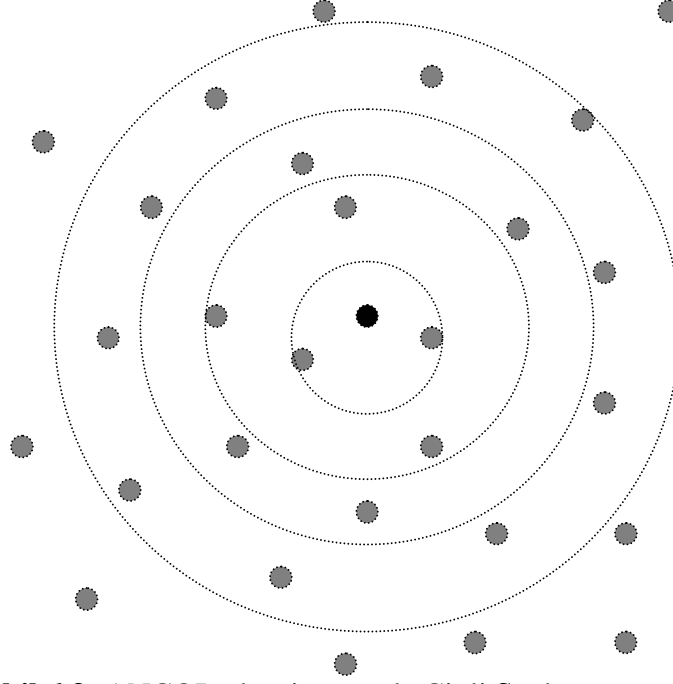
## 6.2 Pekiştirme Safhası

Başlangıç safhası kaynak düğüm bulunduğunda sona erer. Kaynak bulunduktan sonra (doğadaki karıncalar yiyecek kaynağını keşfettikten sonra), kaynak düğüm ile alıcı düğüm arasında veri iletişimine zemin hazırlayacak bir yolun tahsis edilmesine gereksinim vardır.

ANCOR algoritmasının pekiştirme safhasında ana hedef, alıcı düğüm ile kaynak düğüm arasında bir yol kurmaktır. Ağ üzerinde hareket eden karıncaları kurulan bu yola çekebilmek için, yol üzerinde bulunan duyarga düğümlerin feromon miktarları mümkün olduğunca yüksek tutulur. Bunun için kaynağı ilk bulan karınca geri dönerken, bırakabileceği en fazla miktarda feromonu düğümlere bırakır. Bu mekanizmaya feromon bombası denilmektedir ve daha önceki bölümlerde açıklanmıştır.



Şekil 6.2: ANCOR algoritması pekiştirme safhası akış diğramı



**Şekil 6.3:** ANCOR algoritmasında Çinli Şapkası yapısı örneği

Alıcı ile kaynak düğümler arasında yol kurma prosedürü, ANCOR algoritmasının doğadaki karınca kolonilerindeki yapılardan ayrıldığı bir noktayı teşkil eder. Karıncalar yiyecek kaynaklarını bulduktan sonra geri dönüp yuvalarını bulurlar. Bu safhada koku duyularının yanında görme duyularını ve kısıtlı hafızalarını da kullanırlar ama yine de bu olayın tüm detayları ile ilgili bilgi elimizde yoktur. ANCOR karıncası geçtiği düğümlerin listesini barındırmadığı için, hedef düğümü bulduğunda, bu hedefe dek geçmiş olduğu düğümleri bilemez. Bunun yerine geri dönüş güzergahını bulabilmek için düğümlerin koku miktarlarını kullanmak zorundadır. Buna göre, her adımda komşu düğümlerden gelen en olumsuz koku takip edilerek alıcı düğüme ulaşılmaya çalışılır. Pekiştirme safhası, olumsuz feromonun izlendiği yegane safhadır.

Başlangıç safhasının ardından, duyurga ağındaki düğümlerin feromon yoğunluklarına bakıldığında, *Çinli Şapkası* benzeri bir yapı ortaya çıktığı görülür (Şekil 6.3). Karıncalar alıcı düğüme yaratılıp ağa yayıldıklarından, başlangıç safhasında en fazla olumsuz feromon alıcı düğüme bırakılmış olur, çünkü yaratılan bütün karıncalar alıcı düğüme olumsuz feromon bırakırlar. Bu bağlamda, alıcı düğümden uzaklaştıkça, düğümlerdeki olumsuz feromon miktarı azalacaktır. Bunun nedeni, uzak düğümlerden karınca geçme olasılığının daha az olmasıdır. Alıcı düğümün ağın ortasında olduğu ve diğer düğümlerin alıcı düğümün etrafına



saçıldığı bir ağ varsayımında, ağdaki feromon yayılımı şu şekilde olacaktır: en fazla olumsuz feromon ağın orta kesimlerinde olacak ve kenarlara doğru olumsuz feromon yoğunluğu azalacaktır. Bu yapı tıpkı ters tutulmuş yayvan bir koniye benzeyen çimli şapkalarına benzediği için *Çimli Şapkası* ismini almıştır. Şekil 6.3'de gösterilen örnek ağ üzerinde ortadaki siyah nokta alıcı düğümü belirtmektedir. Belirtilen alıcı düğümden uzaklaştıkça, halkalarla belirtilen alanlar içerisindeki ortalama olumsuz feromon yoğunluğu azalmaktadır. Yani en içteki halkanın olumsuz feromon yoğunluğu en fazla, en dıştaki halkanın ortalama olumsuz feromon yoğunluğu ise en azdır. Başlangıç safhasında kaynak düğümü bulan karıncanın pekiştirme safhasında alıcı düğümü geri döneceğini garantileyen yapı işte bu kademeli dağılan olumsuz feromon yapısı, yani *Çimli Şapkasıdır*. Ağın en dış noktasında bile olsa kaynak düğüm bulunduğunda, ortalama olumsuz feromon miktarları gitgide yükselen halkaları takip eden karınca sonunda en içteki halkaya ulaşacaktır ve en yoğun olumsuz feromona sahip düğüm olan alıcı düğümü bulacaktır. Bu işlem sırasında geçtiği düğümlere olumlu feromon bombası bırakıp kaynağa giden düğümü pekiştirecek böylece diğer karıncaların yoldan haberdar olması sağlanacaktır.

### **6.3 Yönlendirme Safhası**

Alıcı düğüm ile kaynak düğüm arasındaki yolu meydana getiren düğümlere mümkün olan en yüksek miktarda olumlu feromon bırakmak suretiyle karıncaların takip edecekleri güzergah belirlenmiş olur. Bu aşamadan sonra kaynak düğümeden alıcı düğümü doğru veri akışı gerçekleşmeye başlar ve kullanıcı iptal edene kadar ya da herhangi bir sonlandırma koşulu yerine getirilene kadar sürer. ANCOR algoritmasının bu safhasına yönlendirme safhası adı verilir.



**Şekil 6.4:** ANCOR algoritması yönlendirme safhası akış diagramı

ANCOR karıncaları, yukarıdaki üç adımı sonsuz bir döngü içerisinde takip ederler. Algoritma bu yönü ile doğal karınca kolonilerinde uygulanan sistem ile çok büyük benzerlik gösterir. Algoritmanın genelinde, bu üç aşamayı takip etmeyen tek karınca, başlangıç safhasında kaynak düğümü bulduktan sonra pekiştirme safhasında en olumsuz feromonu takip ederek alıcı düğümüne dönen karıncadır.

## 7. TEST

Bu bölümde, ANCOR algoritmasını denemek üzere hazırlanmış yazılımlardan ve bu yazılımlardan elde edilen test sonuçlarından bahsedilecektir.

### 7.1 Test Platformu

Benzetim programı olarak üç adet yazılım hazırlanması planlanmıştır. Bunlardan ilki nesneye dayalı hazırlanmış ve C++ dili ile kodlanmış, diğeri ise C programlama dili ile kodlanmıştır. Son benzetim programı ise çok yüksek miktarda düğüm içeren ağlarda koşturulacak olan senaryoların daha verimli çalışabilmesi açısından tasarlanan paralelleştirilmiş bir benzetim programıdır. C ile hazırlanan yazılıma verilen ANCORS ismi, *ANCOR Simulator* kelimelerinin kısaltılması ile elde edilmiştir. ANCORS++ yazılımı C++ ile yazılmıştır. Anlatılacak en son yazılım olan PANCORS'un ismi ise *Parallel ANCOR Simulator* kelimelerinin kısaltılması ile elde edilmiştir ve ANCORS yazılımının paralelleştirilmiş versiyonu olduğunu belirtir.

#### 7.1.1 ANCORS++

ANCOR algoritmasını test etmek üzere hazırlanmış ilk benzetim yazılımı, Microsoft firmasının hazırlamış olduğu Visual C++ 6.0 yazılımı ile hazırlanmıştır. Nesneye dayalı olarak tasarlanan yazılım kodlandıktan sonra, yayınlanan makalede [1] kullanılmak üzere bir takım sonuçlar alınmıştır. Ancak programdaki her unsurun nesne olarak tasarlanması ve dinamik yapılardan ziyade sabit dizilerin kullanılmış olması, özellikle büyük ağları temsil eden senaryolar yürütüldüğünde yeterli randımanın alınamamasına neden olmuştur. Yaklaşık olarak 1000 düğüm içeren senaryoları problemsiz koşturan yazılım, daha yüklü miktarda düğüm içeren senaryolarda oldukça yavaşlamaktadır. Yayınlanan makalede de [1] test kısmında en büyük ağ olarak 625 düğüm kullanılmıştır.

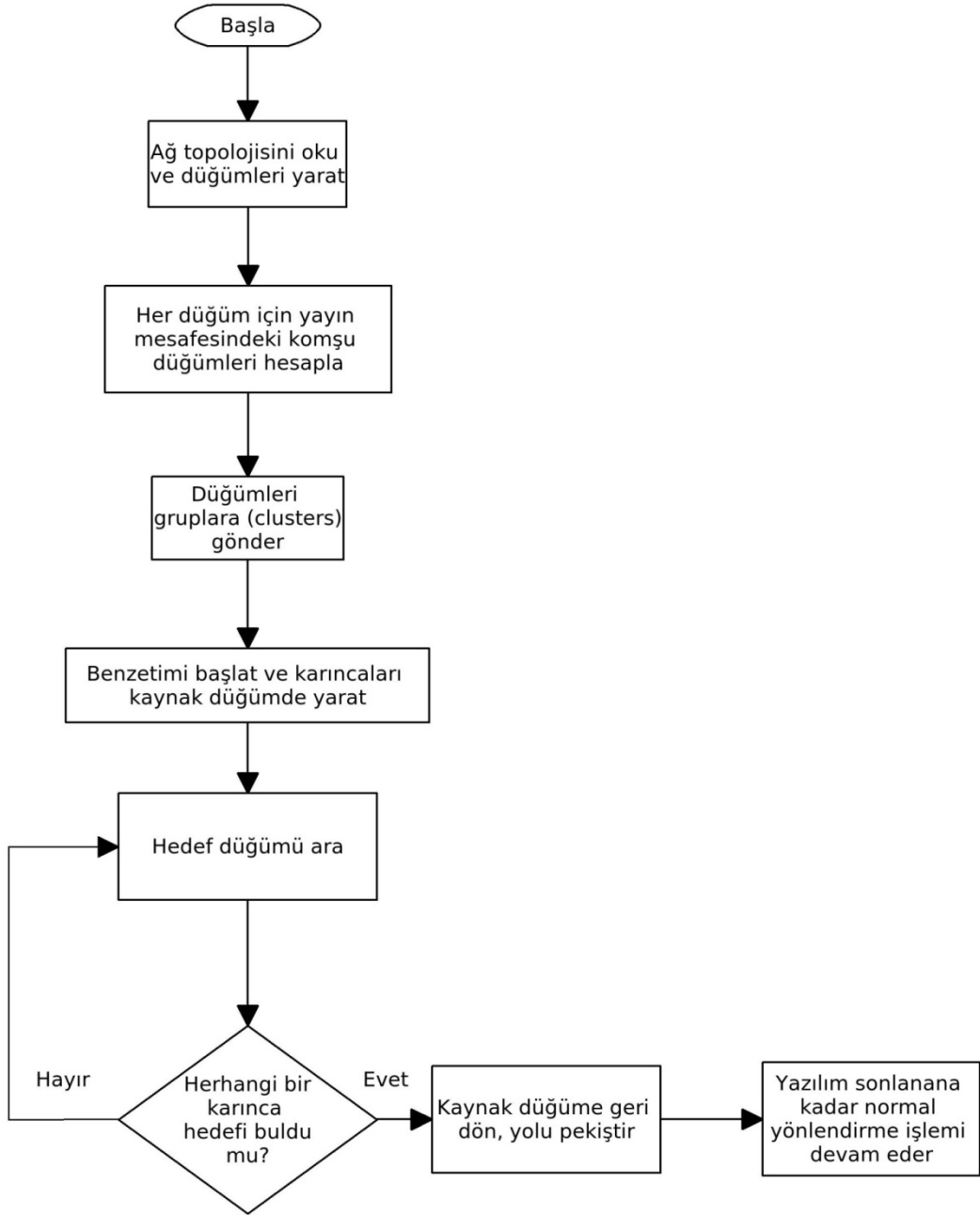
### **7.1.2 ANCORS**

1000 düğümden daha fazla miktarda düğüm içeren senaryoları koşturmak üzere başka bir yazılım geliştirilmiştir. Bunun için iki alternatiften biri, yazılımı daha dikkatli bir biçimde C programlama dili ile hazırlamak ve kodları mümkün olduğunca optimize etmektir. Bunun için öncelikle ANCORS++ yazılımının içinde kullanılan sabit diziler, dinamik veri yapılarına çevirilmiştir. GNU C derleyicisi ile derlenen program, problemsiz bir şekilde 10000 düğüme kadar olan senaryoları koşturabilmektedir.

Hem ANCORS hem de ANCORS++ tek görevli yazılımlar olarak tasarlanmıştır. Performans artışı sağlamak amacıyla bu yazılımların çok görevli versiyonları tasarlanabilir.

### **7.1.3 PANCORS**

Ancors programının koşturamadığı daha büyük ağlar içeren senaryolar için, paralelleştirilmiş kütüphaneler kullanan yeni bir yazılımın hazırlıkları sürmektedir. Bunun için kullanılacak kütüphaneler araştırıldığında, OpenMP, Global Arrays ve çeşitli Mpi gerçekleştirmelerini içeren kütüphaneler bulunmuştur. En yaygın olarak kullanılan OpenMPI kütüphanesi (versiyon 1.2.3) seçilmiş, ancak yazılım henüz bitirilememiştir. Yazılım bitirilince alınacak sonuçların bir konferansta sunulması öngörülmektedir. Şekil 7.1’de PANCORS yazılımının akış diagramı gösterilmektedir.



Şekil 7.1: PANCORS akış diagramı

## 7.2 Test Sonuçları

ANCOR algoritmasını test edebilmek için hazırlanan benzetim programlarından ANCORS kullanılarak çeşitli testler yapılmıştır. Buna göre:

- Programın harcadığı zaman ölçülmüştür.
- Hedef düğüm ile kaynak düğüm arasında tahsis edilen yolun optimal yoldan ne kadar saptığı incelenmiştir (sapma miktarı fazladan kullanılan düğümlerin sayısı olarak ortaya konulmuştur).

- Çeşitli ağ büyüklükleri için çeşitli miktarda karıncalar ve değişik parametrelerle algoritma koşturulmuş ve hedefi bulmadaki performans gözlenmiştir.

Test ortamı olarak kullanılan işletim sistemi 2.6.22-14-generic linux çekirdeğine sahip ubuntu gnu/linux işletim sistemidir. Donanımlar ise;

1. 2.66 Ghz intel pentium 4 işlemci ve 1 Gbyte bellek
2. Amd Athlon X2 3800+ işlemci ve 1 Gbyte bellek
3. Amd Athlon X2 4200+ işlemci ve 2 Gbyte bellek

olarak üç farklı sistemden oluşmaktadır.

### 7.2.1 Parametre Tahmini

ANCOR algoritmasının içerdiği parametrelerin optimum değerlerini kestirebilmek amacıyla, Şekil 7.2’de görülen kare ızgara yapılı bir ağ üzerinde çeşitli testler yapıldı. Şekil 7.2’de siyah renkli düğüm alıcı düğümü belirtirken, gri renkteki düğüm ise kaynak düğümü belirtmektedir. Yapılan testler sırasında aşağıdaki parametreler sabit alınmış ve karıncaların kaynak düğümü bulup geri gelme sürelerine (geçilen düğüm cinsinden) bakılmıştır;

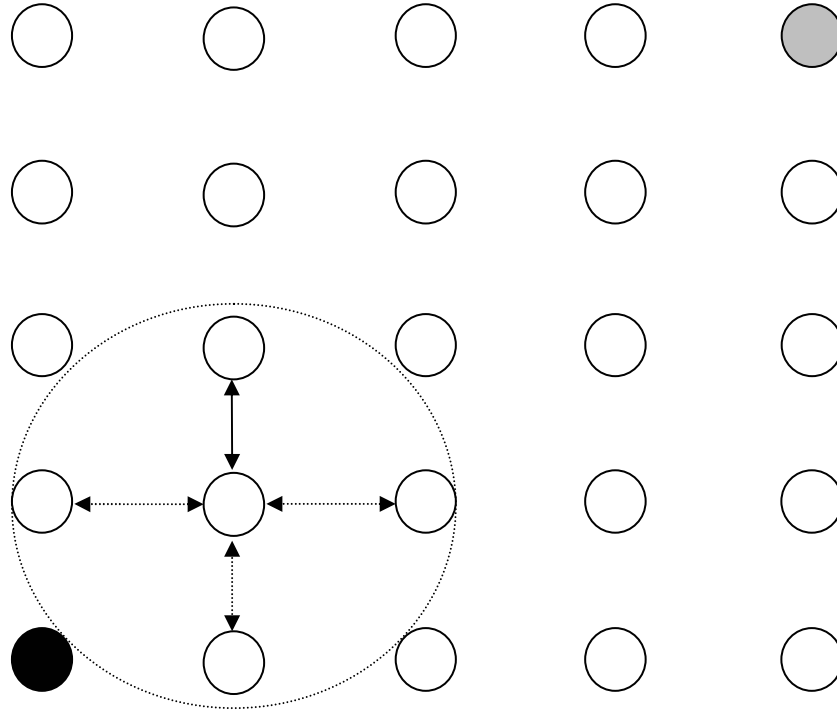
- Düğümlerin radyo menzili = 1
- Buharlaşma katsayısı  $\delta = 0$

Bu testler sırasında buharlaşma mekanizmasının devre dışı bırakılmasının nedeni, düğümler üzerindeki feromon bilgisinin kaybolmamasını sağlamaktır. Radyo menzilinin 1’e eşit olması, Şekil 7.2’de gösterildiği gibi kare ızgara yapılı bir ağda, düğümlerin sadece sağ, sol, yukarı ve aşağılarındaki komşuları ile haberleşebilmelerini sağlamaktadır. Karıncaların ağdaki yaşam süreleri (*maxAtlama*) ise yeterli miktarda büyük tutularak karıncaların hedefi bulmaları garanti altına alınmıştır.

Sabit parametrelerin dışında değişken olarak alınan parametreler ise;

- Karıncanın gideceği bir sonraki düğümün seçilmesi sırasında en sık kullanılan yola gitme eğilimini ayarlayan  $\beta$  ( $\beta \in [0..1]$ )
- Karıncanın bıraktığı feromonun etkinliğini ayarlayan  $\rho$  ( $\rho \in [0..1]$ )

- Karınca sayısı 1, 5, 10 ve 15



**Şekil 7.2:** 5x5'lik kare ızgara ağ örneği (radyo menzili = 1)

Şekil 7.2'de gösterilen 5x5 boyutlarındaki kare ızgara ağ örneği 25 düğüm içermektedir. Radyo menzili 1 olan düğümlerden kurulmuş böyle bir ağ yapısı için, Şekil 7.2'de belirtildiği gibi yerleştirilen alıcı ve kaynak düğümler arasındaki optimal yolun uzunluğu, Formül 7.1'de verilmiştir. Ancak bu şekilde tasarlanmış bir ağda birden çok sayıda optimal yol mevcut olduğundan, hedefi bulma başarımına bakılacak ve parametreler üzerinde bu başarıyı optimize etmeye yönelik ayarlamalar yapılacaktır.

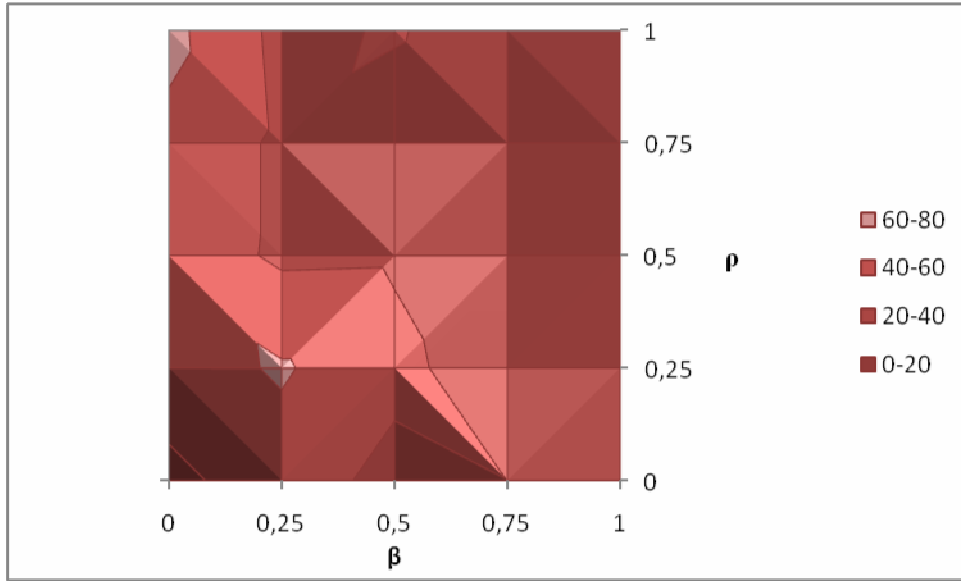
$$I = 2(A - 1) \quad (7.1)$$

Formül 7.1'de  $A$  ile gösterilen, kare şeklindeki ızgara ağ yapısının bir kenarındaki düğüm sayısı,  $I$  ile gösterilen ise alıcı ile kaynak düğümler arası optimal yol uzunluğudur. Tablo 7.1'de, testlerde kullanılan ağlardaki düğüm sayıları ve optimal yol uzunlukları verilmiştir.

Boyut	Düğüm Sayısı	İdeal Yol Uzunluğu
5x5	25	8
10x10	100	18
25x25	625	48

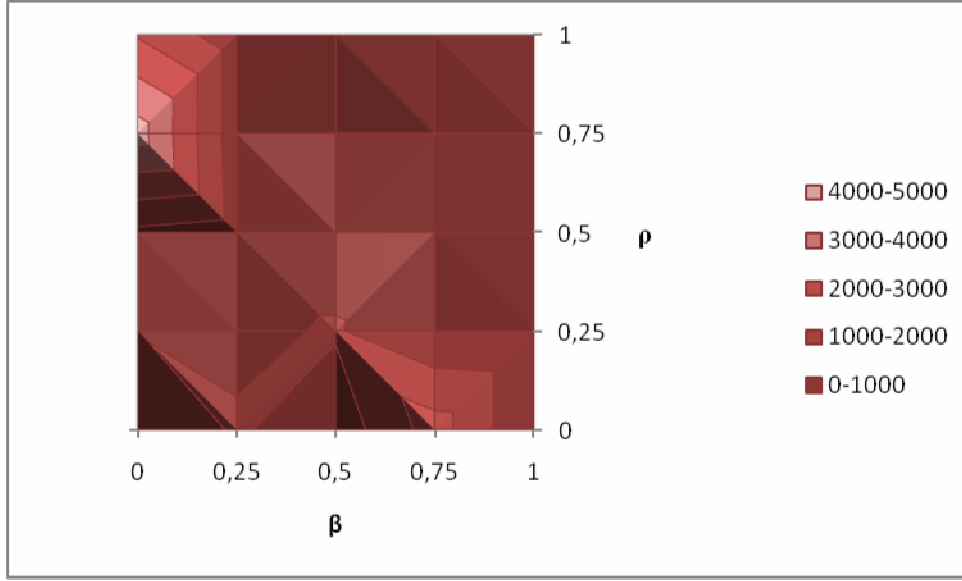
**Tablo 7.1:** Kare ızgara ağ örnekleri (radyo menzili = 1)

$\beta$  ve  $\rho$  nun değişiminin çeşitli büyüklükteki ağlarda hedefi bulma süreleri üzerindeki etkisine bakılmıştır. Karıncaların buldukları düğümden bir sonraki düğüme gitmeleri için gerekli olan süre, her karınca için eşit alınmıştır, buna göre program testi sırasında bahsedilen süre, karıncanın hedef düğüme kadar geçtiği düğümlerin sayısı ile doğru orantılıdır. Her testte  $\beta$  ve  $\rho$  değerleri 0, 0.25, 0.5, 0.75 ve 1.0 olarak alınmış, bu değerler için ANCORS programı 10 tur çalıştırılmış, bunun sonucunda elde edilen değerlerin ortalaması ve varyansı hesaplanmıştır. Hesaplanan ortalama ve varyans değerleri 3 boyutlu grafiklerle gösterilmiştir.



**Şekil 7.3:** 5x5'lik kare ızgara ağ (radyo menzili = 1 karınca sayısı = 1) için hedefi bulma ortalaması

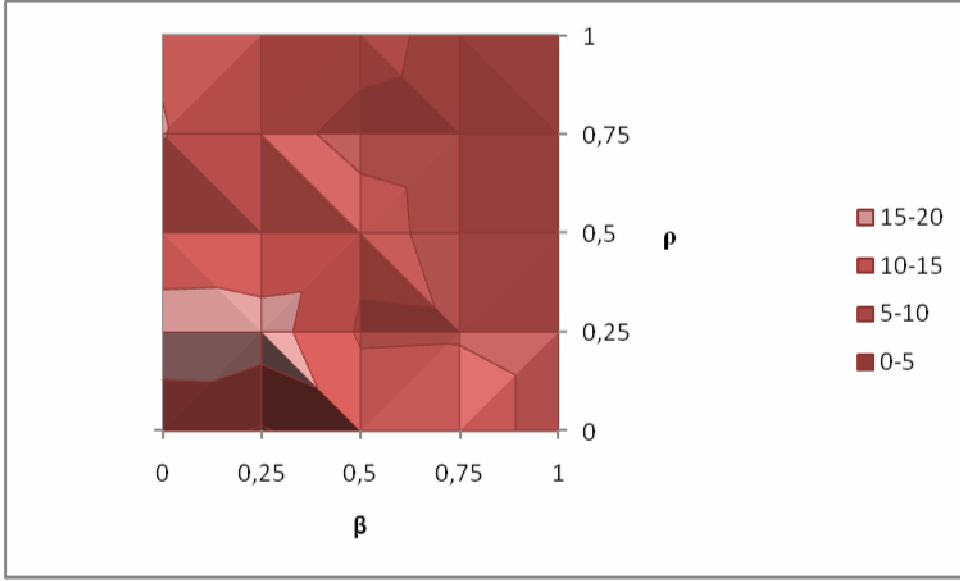




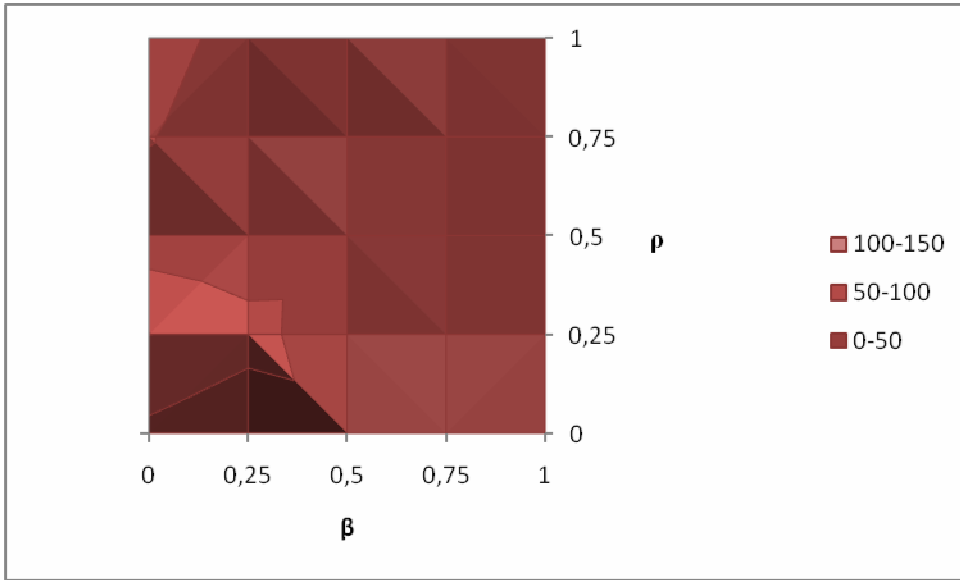
**Şekil 7.4:** 5x5'lik kare ızgara ağ (radyo menzili = 1 karınca sayısı = 1) için hedefi bulma varyansı

Şekil 7.3'te, 25 düğüm içeren bir ağda, bir karınca ile yürütülen algoritmanın ortalama hedefi bulma süresi gösterilmiştir. Şekil 7.4'te ise, bu değerlere denk düşen varyanslar gösterilmiştir. ANCOR'un çok sayıda karınca içeren kolonileri temel olarak tasarlandığı göz önüne alındığında, sadece 1 karınca kullanılması zaten mantıklı değildir. Yapılan gözlemler sonucunda daha fazla karınca kullanımının hem performans artışı sağladığı hemde varyansın azalmasını sağladığı görülmüştür.

Şekil 7.5 incelendiğinde, 1 yerine 5 adet karınca kullanılmasının daha iyi performans gösterdiği görülmektedir. Ayrıca Şekil 7.6'da varyansın da düştüğü görülebilir.  $\beta$  parametresinin 0..0,5 arası değerleri için varyans yüksek çıkmaktadır. Bunun yanında  $\rho$  parametresinin 0,5'ten büyük değerleri için daha iyi performans elde edildiği gözlenebilir. Benzer şekilde 10 adet karınca kullanılması da genel performansda artış sağlamıştır.

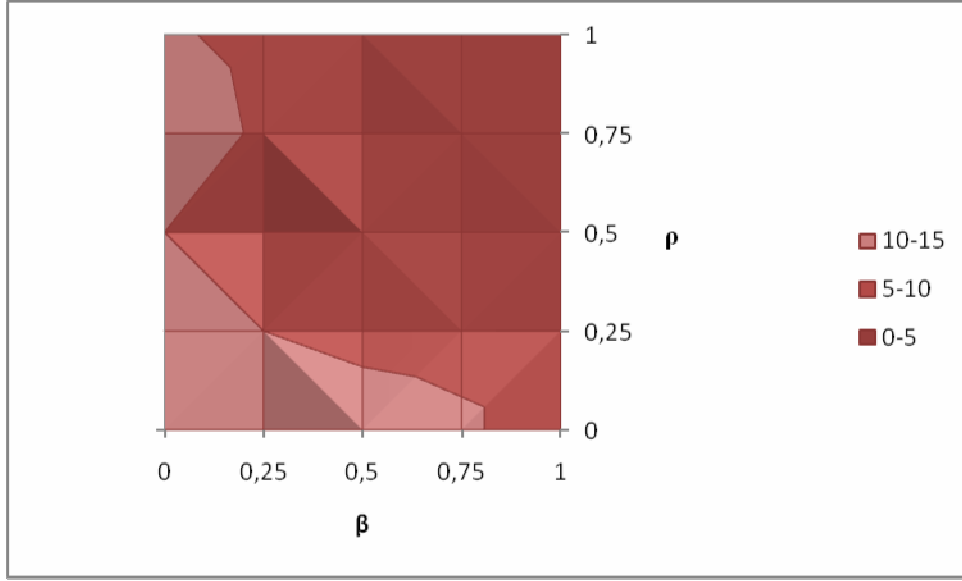


**Şekil 7.5:** 5x5'lik kare ızgara ağ (radyo menzili = 1 karınca sayısı = 5) için hedefi bulma ortalaması

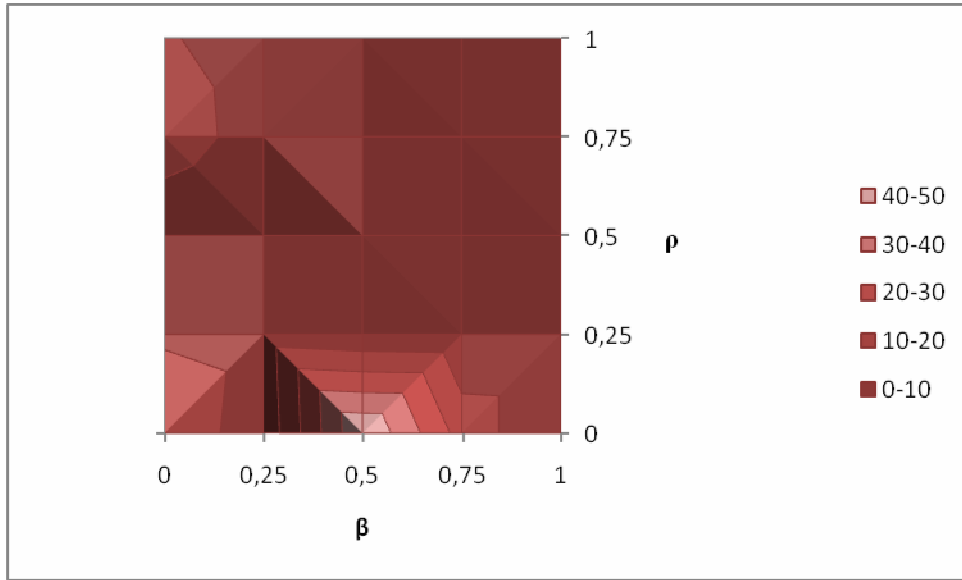


**Şekil 7.6:** 5x5'lik kare ızgara ağ (radyo menzili = 1 karınca sayısı = 5) için hedefi bulma varyansı

Şekil 7.7 ve Şekil 7.8 incelendiğinde hem ortalama değerlerin hem de varyans değerlerinin genelinde bir azalma gözlenebilir.  $\beta$  ve  $\rho$  parametrelerinin değer aralıklarının, 5 karıncalı senaryodakinden fazla farklı olmadığı görülebilir.

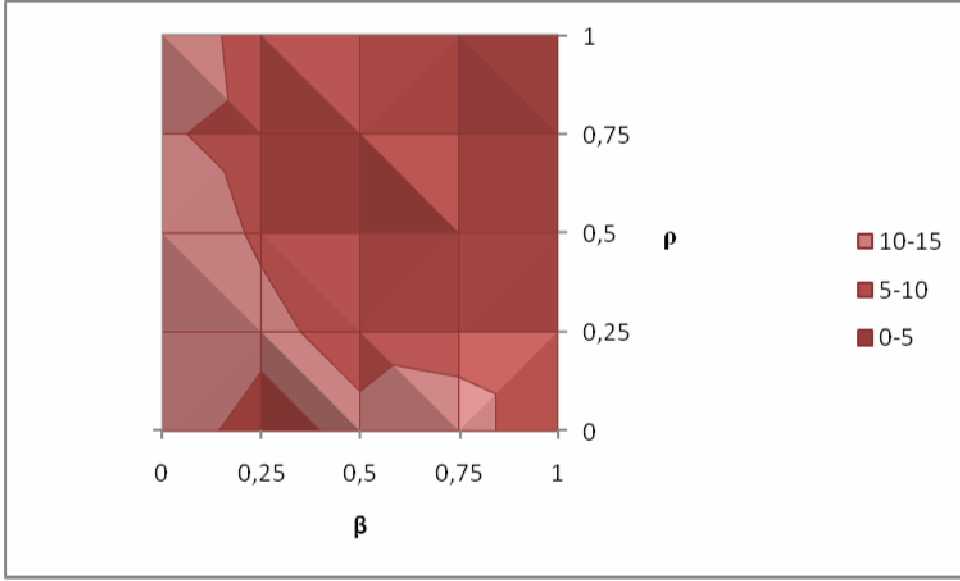


**Şekil 7.7:** 5x5'lik kare ızgara ağ (radyo menzili = 1 karınca sayısı = 10) için hedefi bulma ortalaması

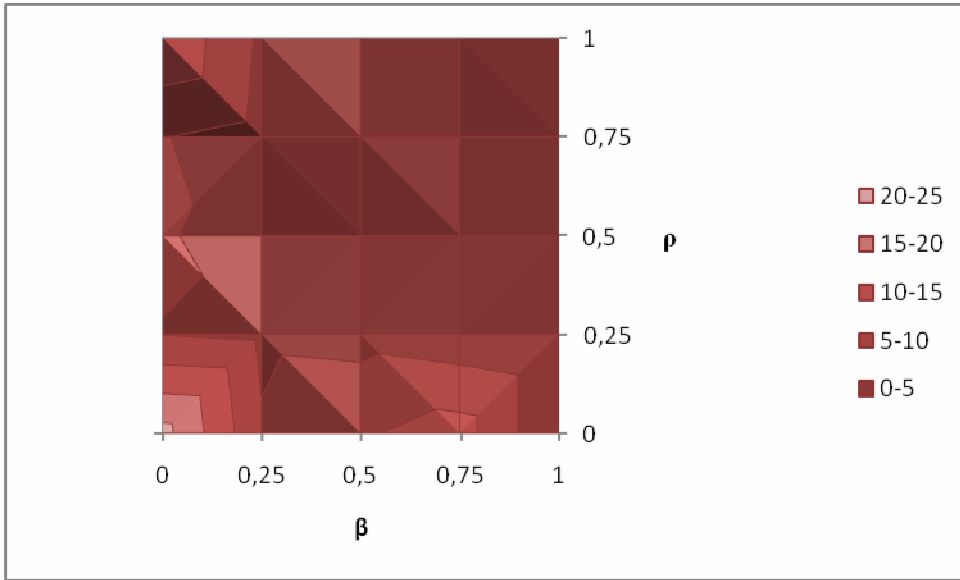


**Şekil 7.8:** 5x5'lik kare ızgara ağ (radyo menzili = 1 karınca sayısı = 10) için hedefi bulma varyansı

Şekil 7.9 ve Şekil 7.10'da gösterilen veriler, 15 karıncanın kullanılmasının, hedefi bulma ortalamasını değiştirmese bile, varyansın azalmasını sağladığı görülmektedir. 0,5 ten büyük  $\beta$  değerleri için daha yüksek performans elde edildiği gözlemlenmiştir.

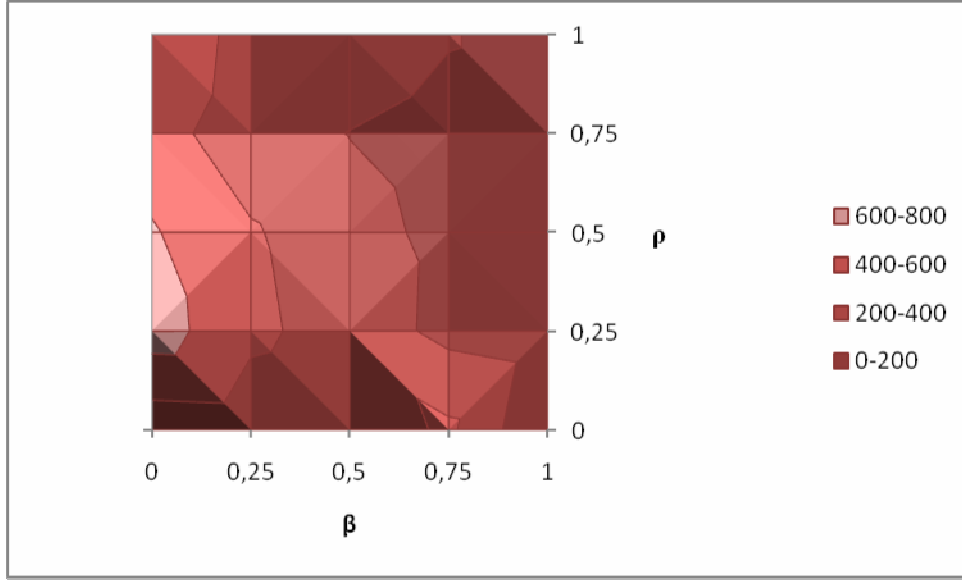


**Şekil 7.9:** 5x5'lik kare ızgara ağ (radyo menzili = 1 karınca sayısı = 15) için hedefi bulma ortalaması

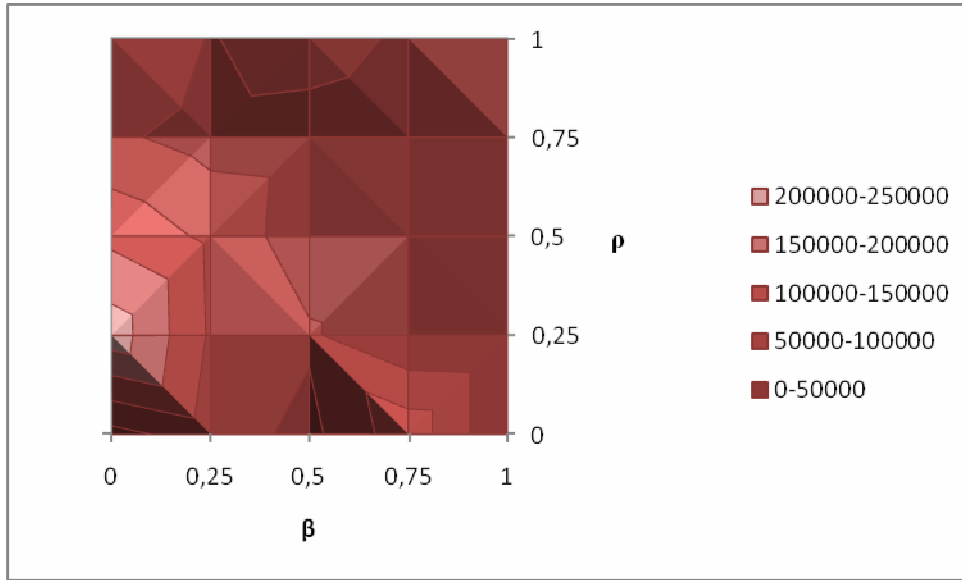


**Şekil 7.10:** 5x5'lik kare ızgara ağ (radyo menzili = 1 karınca sayısı = 15) için hedefi bulma varyansı

Şekil 7.3'ten Şekil 7.10'a kadar olan verileri elde etmek için ANCORS programı, bilgisayar donanımlarından birincisinde koşturulmuş, ve her çalıştırmada ortalama 0,04 saniye zaman harcamıştır.



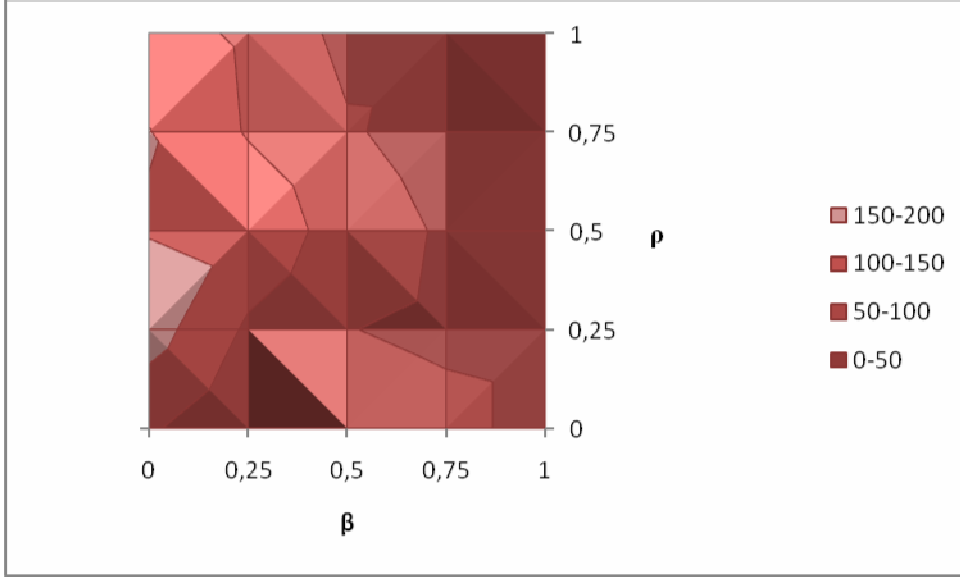
**Şekil 7.11:** 10x10'luk kare ızgara ağ (radyo menzili = 1 karınca sayısı = 1) için hedefi bulma ortalaması



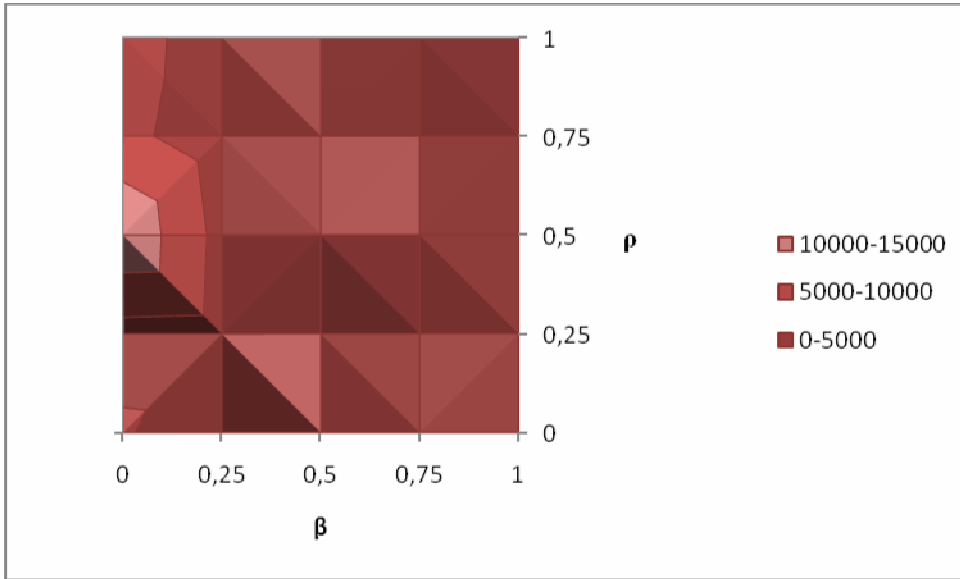
**Şekil 7.12:** 10x10'luk kare ızgara ağ (radyo menzili = 1 karınca sayısı = 1) için hedefi bulma varyansı

100 düğüm içeren bir ağda, 1 karınca kullanılan senaryo sonucunda Şekil 7.11 ve Şekil 7.12'deki sonuçlar elde edilmiştir. Tıpkı 25 düğümlü ağda olduğu gibi, 1 karınca kullanılan senaryoda varyans çok büyük çıkmıştır. Şekil 7.13 ve Şekil 7.14'de görüldüğü üzere, 5 karınca kullanılması da ortalama ve varyans değerini, 25 düğümlük ağlarda olduğu gibi makul sayılabilecek ölçülere indirememiştir.

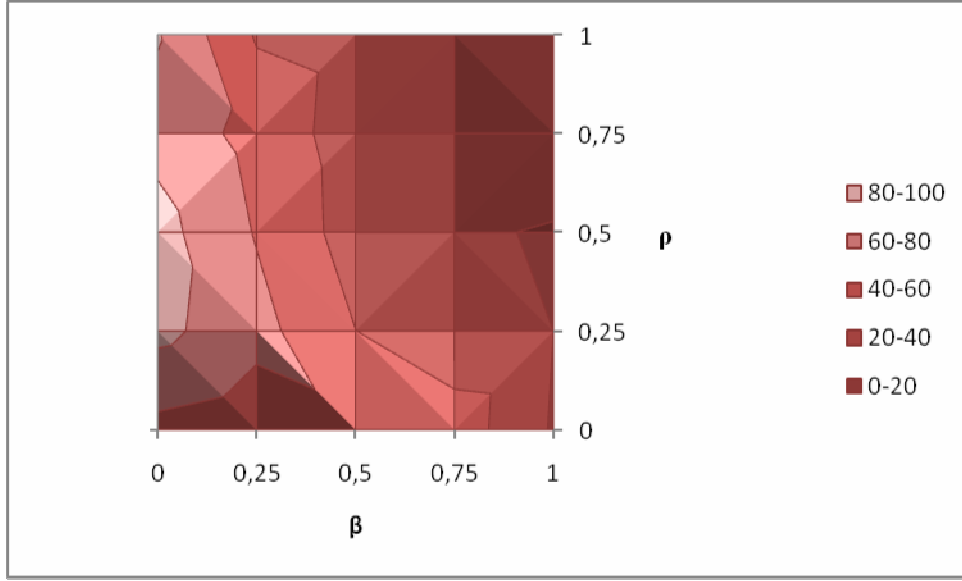
Deneyde kullanılan ağların boyutları büyüdükçe daha fazla karınca kullanılması ihtiyacı doğmaktadır.



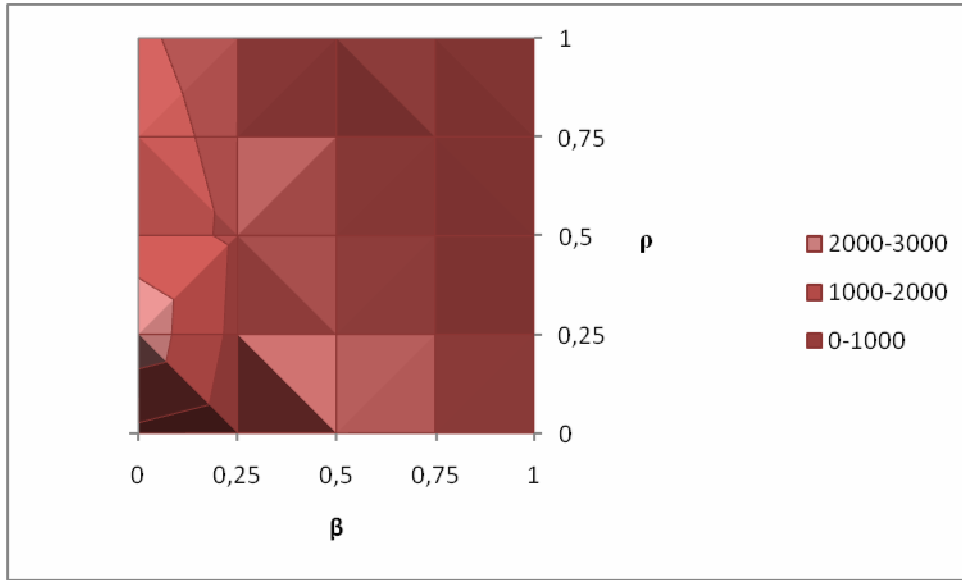
**Şekil 7.13:** 10x10'luk kare ızgara ağ (radyo menzili = 1 karınca sayısı = 5) için hedefi bulma ortalaması



**Şekil 7.14:** 10x10'luk kare ızgara ağ (radyo menzili = 1 karınca sayısı = 5) için hedefi bulma varyansı

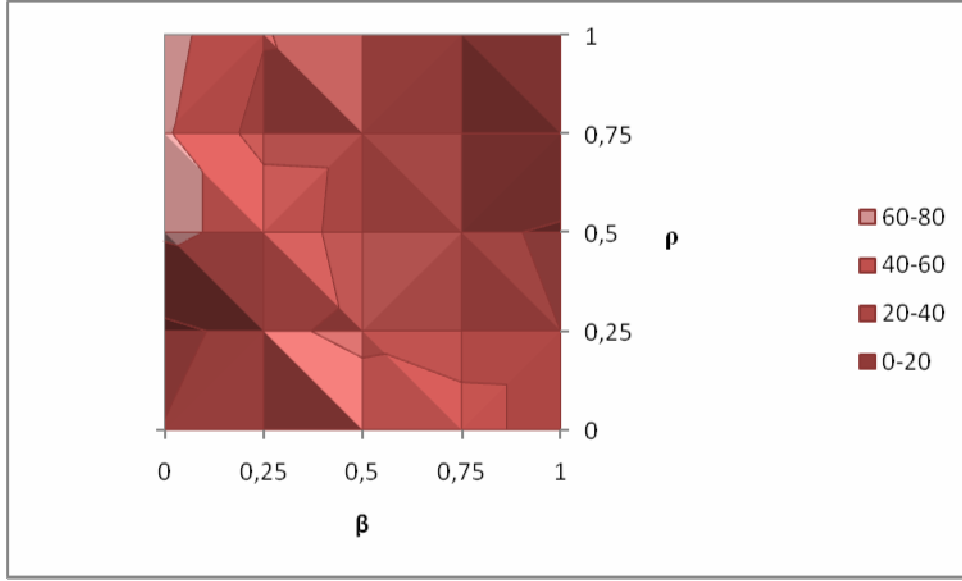


**Şekil 7.15:** 10x10'luk kare ızgara ağ (radyo menzili = 1 karınca sayısı = 10) için hedefi bulma ortalaması

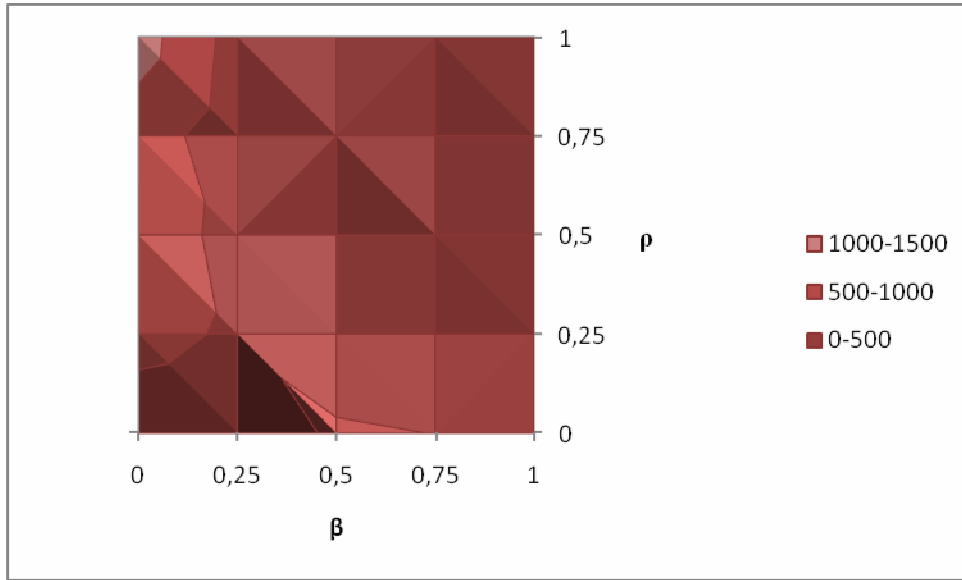


**Şekil 7.16:** 10x10'luk kare ızgara ağ (radyo menzili = 1 karınca sayısı = 10) için hedefi bulma varyansı

Ağ boyutunun büyümesi ile birlikte genel performansın artışında en önemli etken karınca sayısı olmaktadır. Şekil 7.15 ve Şekil 7.17'ye bakıldığında, hedefe varış ortalama değerlerinin, 10 ve 15 karınca kullanıldığında sırasıyla azalmaya devam ettiği görülmektedir. Benzer şekilde Şekil 7.16 ve Şekil 7.18'de gösterilen varyans değerlerinin de, Şekil 7.12 ve Şekil 7.14'e göre belirgin şekilde azaldığı görülebilir.



**Şekil 7.17:** 10x10'luk kare ızgara ağ (radyo menzili = 1 karınca sayısı = 15) için hedefi bulma ortalaması

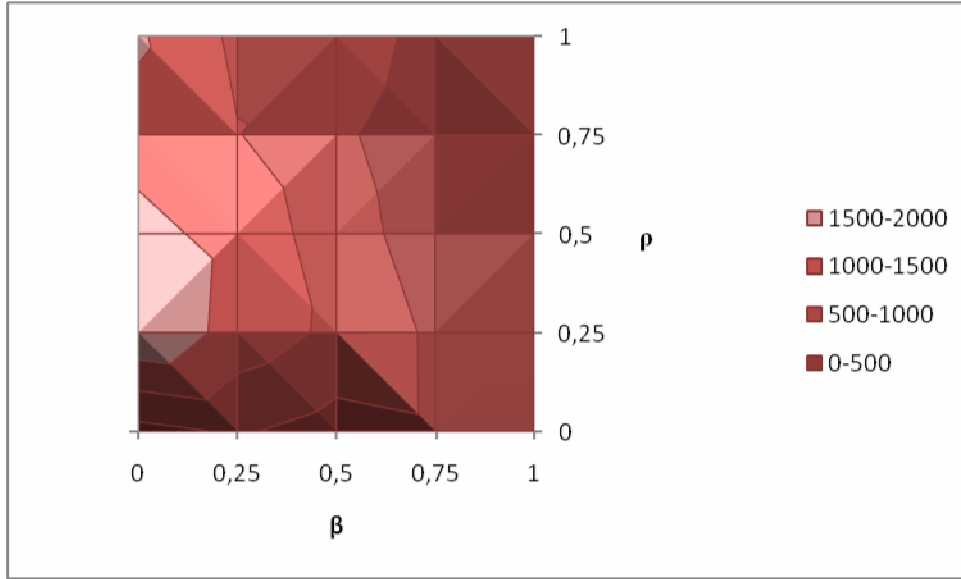


**Şekil 7.18:** 10x10'luk kare ızgara ağ (radyo menzili = 1 karınca sayısı = 15) için hedefi bulma varyansı

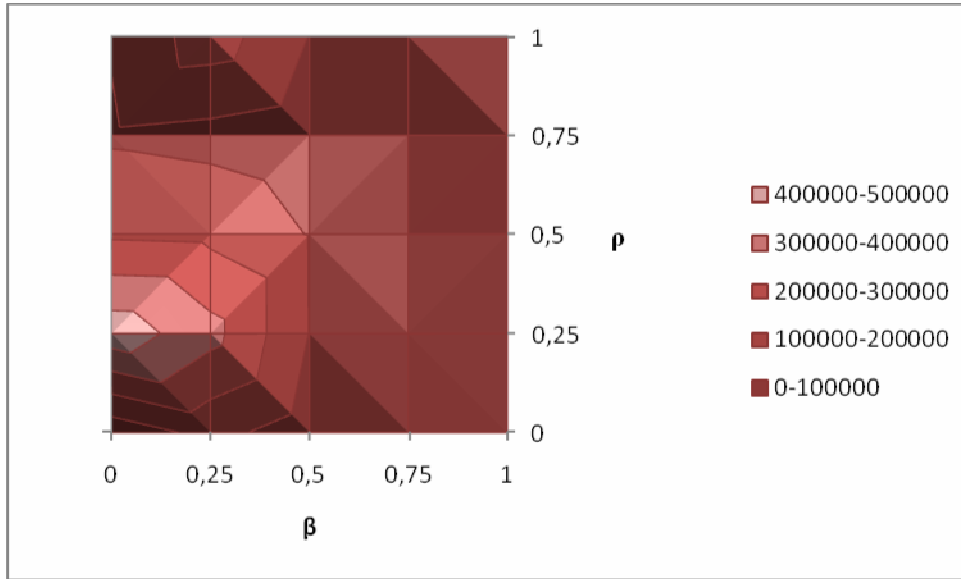
100 düğüm içeren senaryolar ANCORS programı ile, bilgisayar donanımlarından birincisinde koşturulmuş, ve her çalıştırmada ortalama 0,06 saniye zaman harcamıştır. Şimdiye kadar incelenen örnekler sonucunda, ANCOR algoritmasında az sayıda karınca kullanmanın performansı olumsuz yönde etkilediği görülmüştür. 100 düğüm içeren ağlarda  $\beta$  ve  $\rho$  parametreleri incelendiğinde, ANCOR'un  $\beta$ 'nin 0,5ten büyük değerleri için hem ortalama değerlerde hemde varyanslarda daha iyi



sonuçlar alındığı, benzer şekilde  $\rho$  parametresinin 0,5 ten büyük değerleri için de daha iyi sonuçlar alındığı gözlenmiştir.



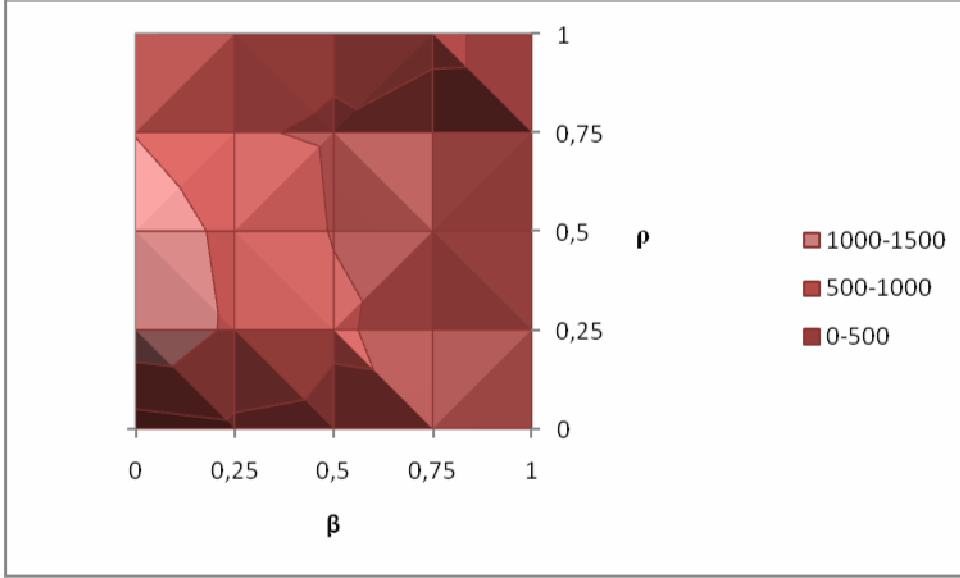
**Şekil 7.19:** 25x25'lik kare ızgara ağ (radyo menzili = 1 karınca sayısı = 10) için hedefi bulma ortalaması



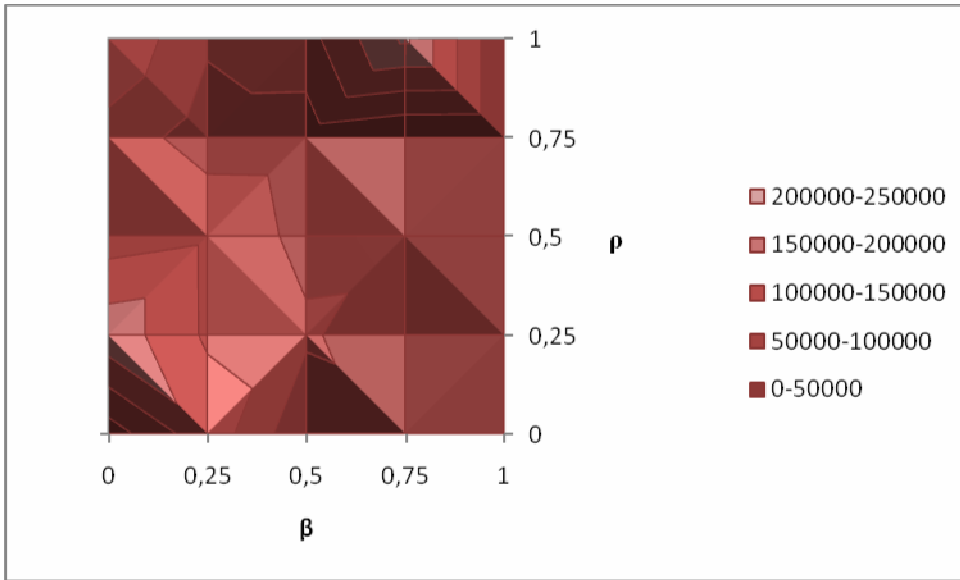
**Şekil 7.20:** 25x25'lik kare ızgara ağ (radyo menzili = 1 karınca sayısı = 10) için hedefi bulma varyansı

625 düğüm içeren ağlar için, 1 ve 5 karınca ile yapılmış testlerin sonuçları, daha önceki testlerde bu miktarda karınca kullanımının ortalama ve varyans değerlerini çok yüksek kıldığından dolayı çalışma dışında bırakılmıştır. 25x25 boyutlu bir ağ için, 10 karıncanın da yeterli olmadığı Şekil 7.19 ve Şekil 7.20'deki değerler

incelendiğinde görülebilir. Şekil 7.21'e bakıldığında, 15 karınca kullanımının 10 karınca kullanımına nazaran ortalama süreler üzerinde büyük bir faydası olmadığı görülse bile, Şekil 7.20 ile Şekil 7.22 karşılaştırıldığında, 10 yerine 15 karınca kullanımının, varyans değerlerini ortalama yarı yarıya azalttığı görülebilir.



**Şekil 7.21:** 25x25'lik kare ızgara ağ (radyo menzili = 1 karınca sayısı = 15) için hedefi bulma ortalaması



**Şekil 7.22:** 25x25'lik kare ızgara ağ (radyo menzili = 1 karınca sayısı = 15) için hedefi bulma varyansı

$\beta$  parametresi açısından yapılan gözlemler, önceki gözlemlerle benzerlik göstermektedir, 0,5'ten büyük değerler kullanıldığında, algoritmada performans

artışı gözlenmektedir.  $\rho$  parametresinin 0,75 değeri hem ortalama süre, hemde sürenin varyansının optimizasyonunda uygun olduğu gözlenmiştir.

625 düğüm içeren ağlarda yapılan denemelerde, 1 numaralı donanım kullanılmıştır ve ANCORS programı çalışmak için ortalama 0,5 saniyeye ihtiyaç duymuştur.

Buraya kadar Şekil 7.2’de örneği verilmiş, 25, 100 ve 625 düğüm içeren ağlar için  $\beta$ ,  $\rho$  ve karınca sayılarına bakılmıştır. Karınca sayılarındaki artışın ANCOR algoritmasının başarımı üzerinde etkili olduğu gözlemlenmiştir. Ayrıca  $\beta > 0,5$  ve  $\rho$  parametresinin 0,75 ve 1 arasındaki değerlerinin yüksek başarımlar gösterdiği saptanmıştır.

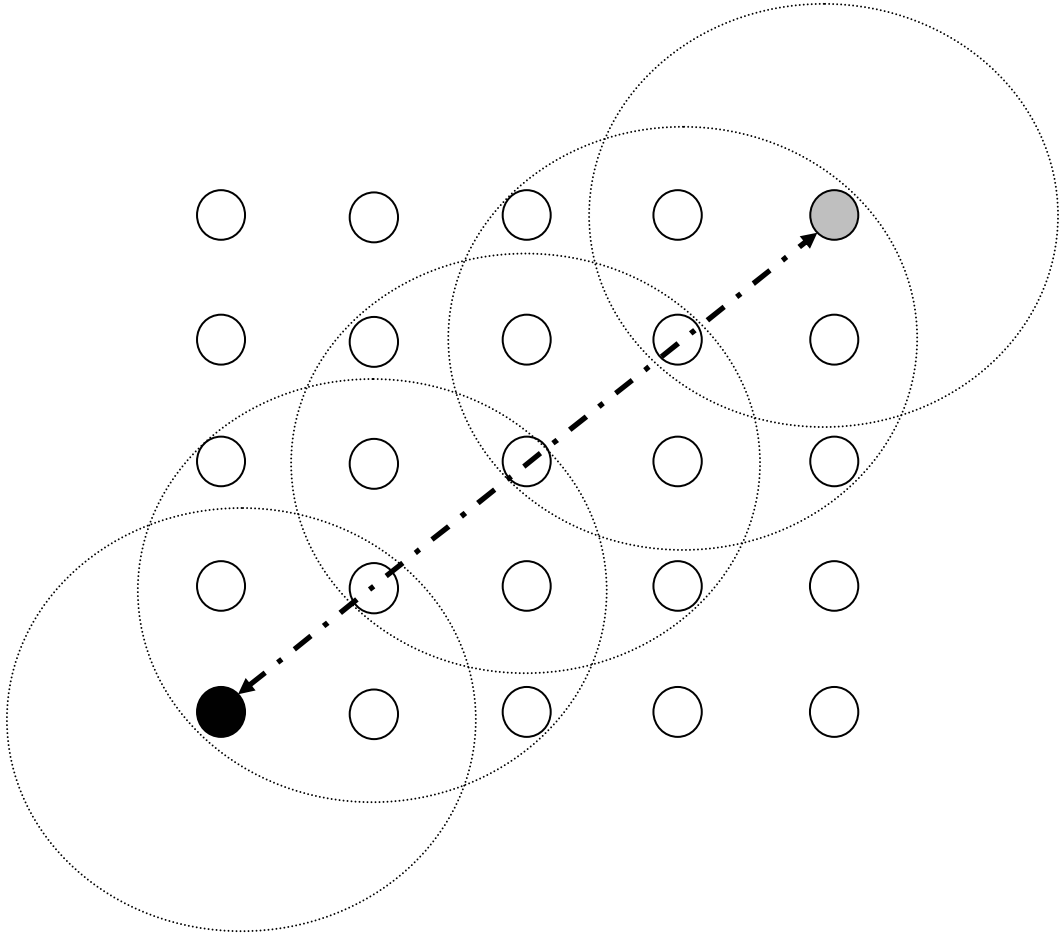
**Tablo 7.2:** Kare ızgara ağ örneği (radyo menzili = 1,8)

Boyut	Düğüm Sayısı	İdeal Yol Uzunluğu
50x50	2500	49

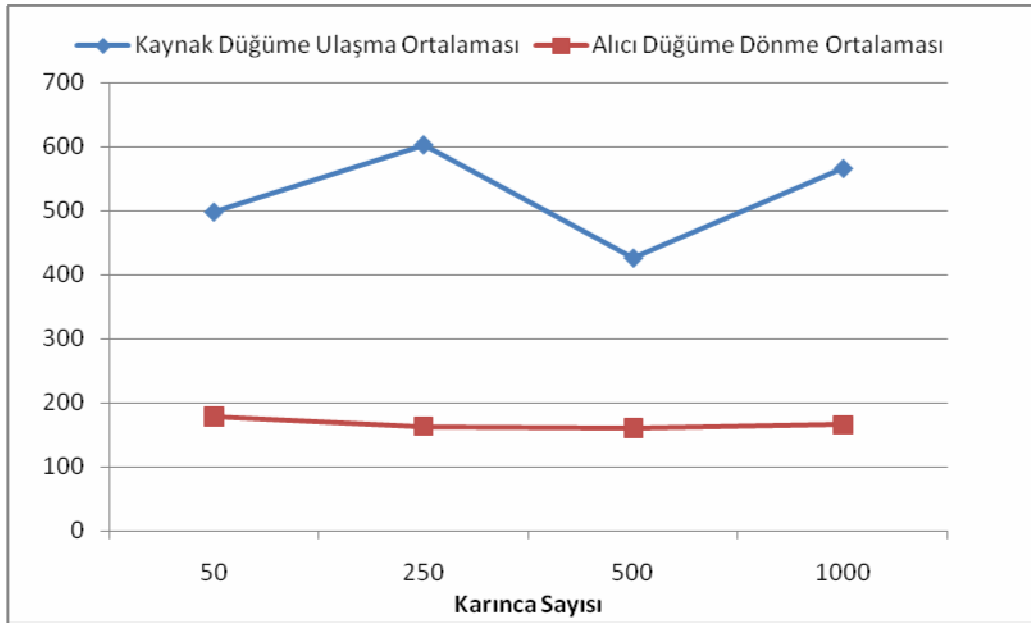
Şekil 7.23’te gösterilen ağ örneğinde, düğümler sağ, sol, yukarı ve aşağılarındaki komşularının yanı sıra çaprazlarındaki komşuları ile de haberleşebilmektedirler. Şekil 7.2’de gösterilen ağ örneğinde, kaynak düğüm ile hedef düğüm arasında çok sayıda optimal yol olmasına karşın, Şekil 7.23’te gösterilen ağ örneğinde sadece bir adet optimal yol bulunmaktadır. Bu yol ağı diagonal olarak geçer. Kare şeklindeki ve bir kenarının uzunluğu  $A$  birim olan bir yapının diagonalinin uzunluğu  $A\sqrt{2}$  olmasına karşın, düğümler arası mesafe ne olursa olsun iletim süresinin eşit olduğu varsayılmış ve hedefe varana dek geçilen düğüm sayısının mesafe olarak alınmasına karar verilmiştir. Bu varsayımlar ve böyle bir ağ yapısı için optimal yol uzunluğu Formül 7.2’de verilmiştir. Buna göre kullanılacak örnek ağ yapısı ve optimal yol uzunluğu Tablo 7.2’de gösterilmiştir.

$$I = A - 1$$

(7.2)

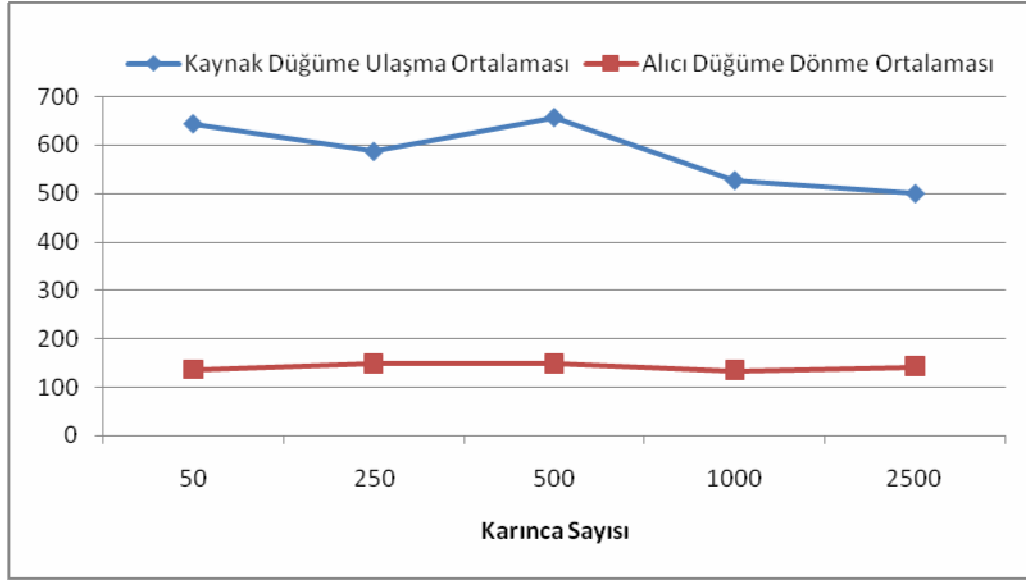


Şekil 7.23: 5x5'lik kare ızgara ağ örneği (radyo menzili = 1,8)



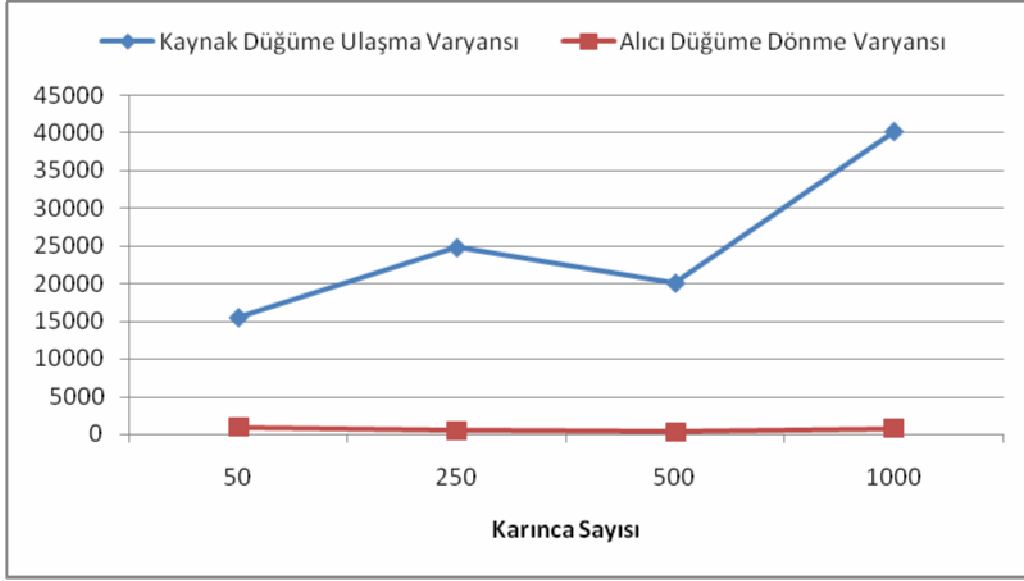
Şekil 7.24:  $\delta = 0.01$  ve 2500 düğümlü ağ için kaynak düğümü bulma ve hedef düğüme dönme ortalamaları

2500 düğümlü ağda koşturulan senaryoda buharlaşma mekanizması da işletilmiştir ve buharlaşma katsayısı  $\delta = 0.01$  ve  $\delta = 0.001$  olarak alınmıştır. Buharlaşma mekanizması zamanla feromonun miktarını azaltır. Miktarı azalan feromonun etkinliği de azalır. Bu nedenle, testlerde buharlaşma kullanıldığı durumlarda  $\rho = 1$  alınmasının daha iyi sonuç verdiği görülmüştür. Bunun nedeni  $\rho < 1$  değerlerinin feromon üzerinde, feromon bırakıldığı anda belli bir süre buharlaşma işletilmiş etkisini yapmasındandır.

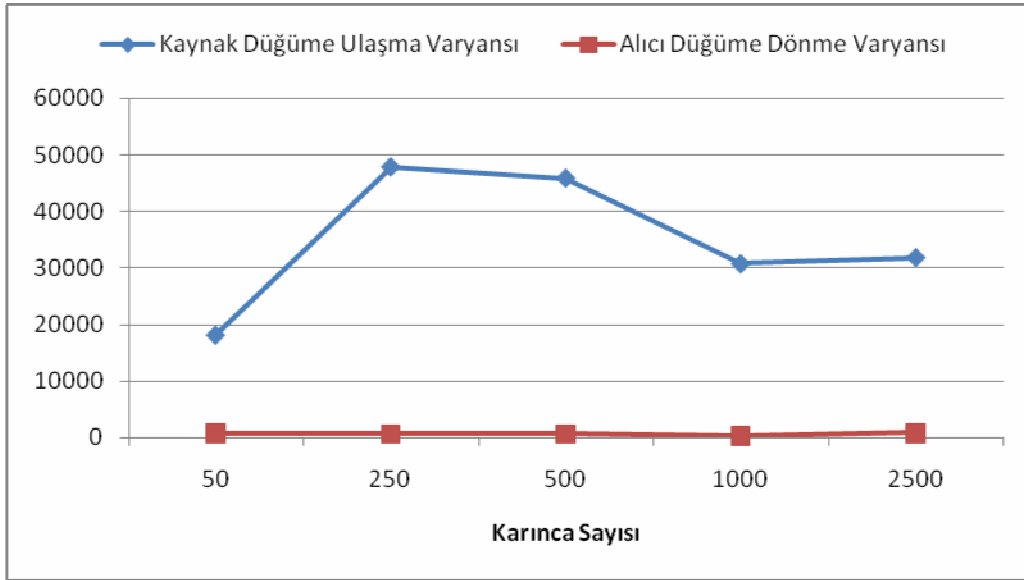


**Şekil 7.25:**  $\delta = 0.001$  ve 2500 düğümlü ağ için kaynak düğümü bulma ve hedef düğüme dönme ortalamaları

Hem Şekil 7.24'te, hem de Şekil 7.25'te görüldüğü gibi, ANCOR algoritmasının başlangıç safhasında kaynak düğümü arama işlemi, kaynak bulunduğunda alıcı düğüme dönülmesi işleminden daha maliyetlidir. Bunun nedeni, başlangıç safhasında, ağ üzerinde karıncaların yönünü tayin edecek feromon bilgilerinin olmamasındandır. Ağ üzerindeki tüm düğümlerin nötr feromona sahip olması, karıncaların ağ üzerinde rastgele arama yapmalarına neden olmaktadır. Kaynak düğüm bulunduğunda ise, kaynak düğüme gelene kadar karıncaların bıraktıkları feromonlar *Çinli Şapkası* modelini oluşturur. Pekiştirme safhasında, kaynak düğümden alıcı düğüme dönen karınca, *Çinli Şapkası* modelini kullanarak alıcı düğüme doğru yönelir.

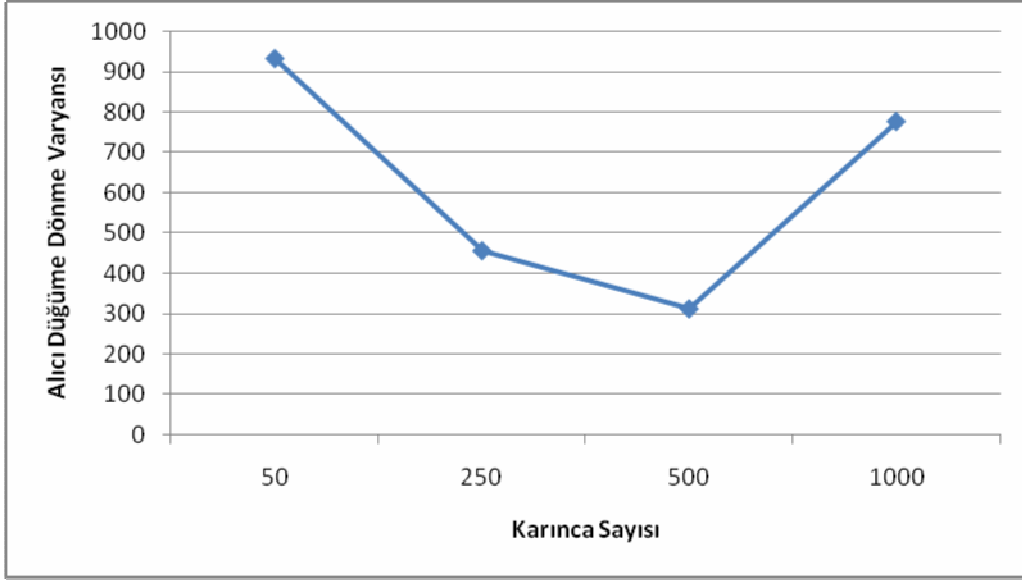


**Şekil 7.26:**  $\delta = 0.01$  ve 2500 düğümlü ağ için kaynak düğümü bulma ve hedef düğüme dönme varyansları

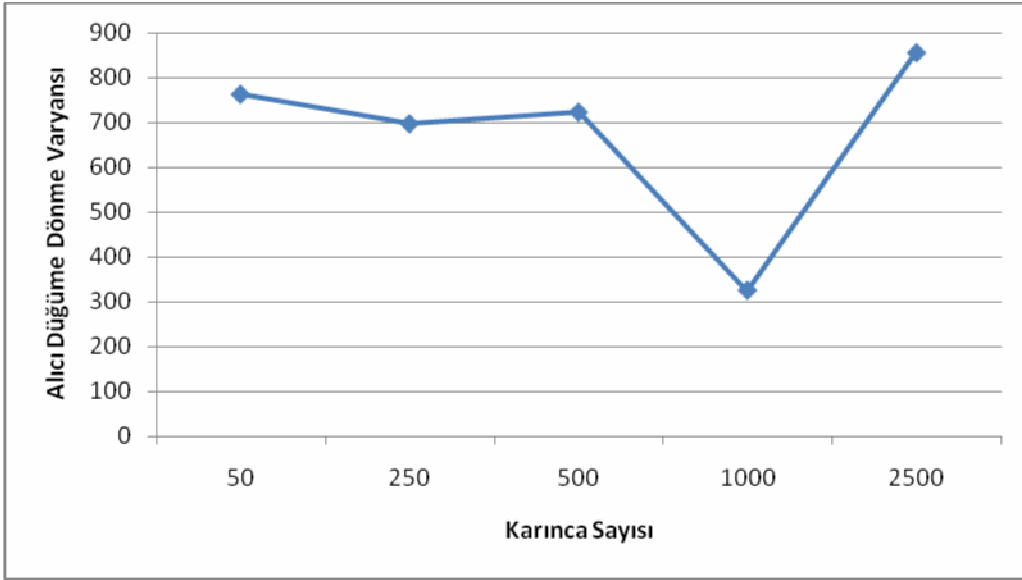


**Şekil 7.27:**  $\delta = 0.001$  ve 2500 düğümlü ağ için kaynak düğümü bulma ve hedef düğüme dönme varyansları

Şekil 7.26 ve Şekil 7.27’de, karıncaların hedef düğümü bulmaları ve alıcı düğüme dönmelerinin varyansları gösterilmektedir. *Çinli Şapkası* modelinin başarımı, varyans değerleri üzerinde de etkilidir. Karıncaların kaynak düğüme ulaşma varyansları, alıcı düğüme dönme varyanslarından çok daha yüksektir. Alıcı düğüme dönme varyansları Şekil 7.28 ve Şekil 7.29’da detaylı olarak verilmektedir.



**Şekil 7.28:**  $\delta = 0.01$  ve 2500 düğümlü ağ için hedef düğüme dönme varyansı



**Şekil 7.29:**  $\delta = 0.001$  ve 2500 düğümlü ağ için hedef düğüme dönme varyansı

Ağda daha fazla karınca kullanımı, alıcı düğüme geri dönme başarımını belli bir yere kadar arttırmaktadır. Aşırı karınca kullanımı, fazla miktarda karıncanın ağda feromon bırakmasına yol açtığı için *Çinli Şapkası* modelinin başarımını olumsuz etkilemektedir (Şekil 7.28 ve Şekil 7.29).

## 8. TARTIŞMA VE GELECEK ÇALIŞMALAR

ANCOR'da kullanılan paketlerin küçük boyutlu olması, iletişim sırasında bantgenişliğinin daha etkin kullanımını sağlamaktadır. ANCOR algoritmasının karmaşıklığı incelendiğinde, ağ üzerindeki her bir düğümün komşu düğümlerinin sayısına bağlı olarak  $O(n)$  işlem yaptığı bulunmuştur. Buna göre ANCOR algoritması ölçeklenebilir bir algoritmadır ve koşturulduğu ağ ne kadar büyük olursa olsun, düğümlerdeki işlem sayısı komşu düğümlerin sayısı ile orantılıdır.

Bu yüksek lisans çalışmasında ANCOR algoritması, tek alıcı düğüm ve tek kaynak düğüm senaryosuna göre koşturuldu. Test aşamasında çeşitli parametrelerin kaynak düğümü bulma ve alıcı düğüme dönme başarımı üzerindeki etkilerine bakıldı. Testler sırasında doğrusal koku dağılım modeli kullanıldı.

Gelecek çalışmalarda, üstel koku dağılım modeli kullanılarak bu modelin ANCOR üzerindeki başarımı test edilecektir. Bunun için yer bilgisinin sağlandığı bir GPS cihazının kullanıldığı varsayılacak ya da yön bilgisi algoritmada kullanılan paketler ile sağlanacaktır.

Yönlendirme safhasında, pekiştirme safhasında kurulan yoldan daha optimal bir yolun kurulması başarımı araştırılacaktır. Bu başarımın artmasında, özellikle üstel koku dağılım modelinin kullanılmasının etkin olacağı öngörülmektedir.

Çok alıcı düğümün ve çok kaynak düğümün bulunduğu senaryolar için ANCOR algoritması geliştirilecek ve koşturulacaktır. Bunun yanında olay güdümlü senaryolarda da ANCOR algoritması koşturulup test edilecektir.

Bant genişliğini daha etkin kullanmak adına, düğümlerin güncellenen feromon miktarlarını yayınlamak için feromon paketleri yerine, güncellenmiş feromon bilgisinin gömüldüğü karınca paketleri kullanılacaktır.

PANCORS yazılımı halen geliştirilme aşamasındadır. Paralleştirilmiş bir gerçek zaman benzetim programı olan PANCORS'un, tek işlemlile halefi ANCORS'a göre başarımı araştırılacaktır.



## KAYNAKLAR

- [1] **Demiray, D. and Altılar, D.T.**, 2007. ANCOR: A Novel Ant Colony Routing Approach for Sensor Networks, *IT&IC*, Vol.1, No.4.
- [2] **Bonabeau, E.**, 1999. A Brief History of Stigmergy, *Artificial Life*, Vol. 5, No. 2, 97-116.
- [3] **Theraulaz, G., Bonabeau, E., Dorigo, M. and Theraulaz, G.**, 1999. Swarm Intelligence From Natural to Artificial Systems, Oxford University Press, London.
- [4] **Dorigo, M., Bonabeau, E. and Theraulaz, G.**, 2000. Ant algorithms and stigmergy, *Future Generation Computer Systems*, **16**, 851-871.
- [5] **Tarasewich, P. and McMullen, P.R.**, 2002. Swarm Intelligence: Power In Numbers, *Communications of The Acm*, Vol. 45, No. 8
- [6] **Bonabeau, E. and Meyer, C.**, 2001. Swarm Intelligence: A Whole New Way To Think About Business, *Harvard Business Review*, May 2001
- [7] **Dorigo, M., Di Caro, G. and Gambardella, L.M.**, 1999. Ant Algorithms for Discrete Optimization, *Artificial Life*, Vol. 5, No.3, 137-172
- [8] **Dorigo, M., Maniezzo, V. and Colorni A.**, 1991. Positive Feedback as a Search Strategy, Politecnico di Milano Department of Electronics Technical Report, Milano, Italy.
- [9] **Hart, A.G. and Ratnieks, F.L.W.**, 2000. Leaf Caching in Atta Leafcutting Ants: Discrete Cache Formation Through Positive Feedback, *Animal Behaviour*, **59**, 587-591
- [10] **Sendova-Franks, A.B. and Franks, N.R.**, 1999. Self-assembly, self organization and division of labour, *Phil. Trans.*, **354**, 1395-1405.
- [11] **Bonabeau, E., Theraulaz, G., Deneubourg, J.L., Franks, N.R., Rafelsberger, O., Joly, J.L. and Blanco, S.**, 1998. A model for the emergence of pillars, walls and royal chambers in termite nests, *Royal Society*, **353**.
- [12] **Mallon, E.B., Pratt, S.C. and Franks, N.R.**, 2001. Individual and collective decision-making during nest site selection by the ant *leptothorax albipennis*, *Behav. Ecol. Sociobiol.*, **50**, 352-359.
- [13] **Franks, N.R., and Richardson, T.**, 2006. Teaching in tandem-running ants, *Nature*, **439**.

- [14] **Pratt, S.C., Mallon, E.B., Sumpter, D.J.T. and Franks, N.R.**, 2003. Collective Decision making in a small society: how the ant *leptothorax albipennis* chooses a nest site, [www.cs.ualberta.ca/~parker/papers/](http://www.cs.ualberta.ca/~parker/papers/)
- [15] **Britton, N.F., Franks, N.R., Pratt, S.C. and Seeley, T.D.**, 2002. Deciding on a new home: how do honeybees agree?, *Proc. Royal Society*, **269**, 1383-1388.
- [16] **Hart, A.G and Ratnieks, F.L.W**, 2002. Task-partitioned nectar transfer in stingless bees: work organization in a phylogenetic context, *Ecological Entomology*, **27**.
- [17] **Robinson, G.E.**, 1999. Integrative animal behaviour and sociogenomics, *Elsevier Science TREE*, Vol., 14, No. 5.
- [18] **Robinson, G.E**, 2002. Sociogenomics takes flight, *SCIENCE*, **297**.
- [19] **Hart, A.G. and Ratnieks F.L.W.**, 2001. Leaf caching in the leafcutting ant *Atta colombica* an organizational shift, task partitioning and making the best of a bad job, *ANIMAL BEHAVIOUR*, **62**.
- [20] **Anderson, C. and Franks, N.R.**, 2001. Teams in animal societies, *International society for behavioral ecology*, Vol.12, No.5.
- [21] **Nedjah, N., and Mourelle, L.M.**, 2006. Swarm Intelligent Systems, *Studies in Computational Intelligence*, **26**.
- [22] **Dorigo, M., Maniezzo, V. and Colorni, A.**, 1991. Ant System: An Autocatalytic Optimizing Process, 1991. Politecnico di Milano Department of Electronics Technical Report, Milano, Italy.
- [23] **Dorigo, M., Maniezzo, V. and Colorni, A.**, 1996. Ant System: Optimization by a Colony of Cooperating Agents, *IEEE Transactions on Systems, Man and Cybernetics*, Vol. 26, No. 1.
- [24] **Dorigo, M. and Socha, K.**, 2006. An Introduction to Ant Colony Optimization, IRIDIA Technical Report No. TR/IRIDIA/2006-010, Brussels, Belgium.
- [25] **Dorigo M.**, 1992. Optimization, Learning and Natural Algorithms, PhD thesis, Politecnico di Milano, Milano, Italy.
- [26] **Dorigo, M. and Gambardella, L.M.**, 1997. Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem, *IEEE Transactions on Evolutionary Computation*, Vol.1 No.1.

- [27] **Stützle, T., and Dorigo, M.,** 1999. ACO Algorithms for the Quadratic Assignment Problem, *New Ideas In Optimization*, McGraw-Hill, New York.
- [28] **Colorni, A., Dorigo, M., Maniezzo, V. and Trubian, M.,** 1994. Ant System for job-shop scheduling, *Belgian J. Oper. Res. Statist. Comput. Sci.*, **34**, 39-53.
- [29] **Bullnheimer, B., Hartl, R.F. and Strauss, C.,** 1999. An improved Ant System algorithm for the vehicle routing problem, University of Vienna Institute of Management Science Technical Report Ann. Oper. Res. 89, Vienna Austria.
- [30] **Schoonderwoerd, R., Holland, O., Bruten, J. and Rothkrantz, L.,** 1996. Ant-based load balancing in telecommunications networks, *Adaptive Behav.*, **5**, 169–207.
- [31] **Costa, D. and Hertz, A.,** 1997. Ants can colour graphs, *J. Oper. Res. Soc.*, **48**, 295–305
- [32] **Maniezzo, V. and Carbonaro, A.,** 2000. An ANTS heuristic for the frequency assignment problem, *Future Generation Comput. Systems*, **16**.
- [33] **Ramalhinho, L.H. and Serra, D.,** 1998. Adaptive approach heuristics for the generalized assignment problem, Universitat Pompeu Fabra Department of Economics and Management Technical Report EWP Series No.304, Barcelona, Spain.
- [34] **Leguizamón, G. and Michalewicz, Z.,** 1999. A new version of Ant System for subset problems, *Proceedings of the 1999 Congress on Evolutionary Computation*, IEEE Press, Piscataway, NJ, 1999, pp. 1459–1464
- [35] **Navarro, V.G. and Sinclair, M.C.,** 1999. Ant colony optimization for virtual-wavelength-path routing and wavelength allocation, *Proceedings of the 1999 Congress on Evolutionary Computation*, IEEE Press, Piscataway, NJ, 1999, pp. 1809–1816
- [36] **Culler, D., Estrin, D. and Srivastava, M.,** 2004. Overview of sensor networks, *IEEE Computer Magazine*, Vol. 37, No.8, 41-49.
- [37] **Akyildiz, I.F., Su, W., Sankarasubramaniam, Y. and Cayirci, E.,** 2001. Wireless sensor networks, Broadband and Wireless Networking Laboratory, School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA

- [38] **Stojmenovic, I.**, 2005. Handbook of Sensor Networks Algorithms and Architectures, Wiley Series on Parallel and Distributed Computing, Wiley, West Sussex, UK.
- [39] **Santini, P.**, 2005. Topology Control in Wireless Ad Hoc and Sensor Networks, Wiley, West Sussex, UK.
- [40] **Shorey, R., Ananda, A., Chan, M.C. and Ooi W.T.**, 2006. Mobile, Wireless and Sensor Networks, Wiley, West Sussex, UK.
- [41] **Karl, H. and Willig, A.**, 2005. Protocols and Architectures for Wireless Sensor Networks, Wiley, West Sussex, UK.
- [42] **Intanagonwiwat, C., Govidan, R., Estrin, D., Heidemann, J. and Silva, F.**, 2003. Directed Diffusion for Wireless Sensor Networking, *IEEE/ACM Transactions On Networking*, Vol.11, No.1.
- [43] **Di Caro, G. and Dorigo, M.**, 1997. AntNet: A Mobile Agents Approach to Adaptive Routing, IRIDIA Technical Report No.97-12, Brussels, Belgium.
- [44] **Schoonderwoerd, R., Holland, O.E., Bruten, J.L. and Rothkrantz, L.J.M.**, 1997. Ant-Based Load Balancing in Telecommunications Networks, *Adaptive Behavior*, Vol. 5, No. 2, 169-207.
- [45] **Leith, C. and Takahara, G.**, 2003. A Control Framework for Ant-Based Routing Algorithms, *Proceedings of the IEEE Congress on Evolutionary Computation*, Vol. 3, pp.1788-1795.
- [46] **Zhang, Y., Kuhn, L.D. and Fromherz, M.P.J.**, 2004. Improvements on Ant Routing for Sensor Networks, *ANTS 2004*, pp. 154-165, Springer-Verlag, Heidelberg.
- [47] **Montgomery, J. and Randall, M.**, 2002. Anti-pheromone as a Tool for Better Exploration of Search Space, *ANTS 2002*, pp. 100-110, Springer-Verlag, Heidelberg.

**SÖZLÜK**

Açgözlü	: Greedy.
Ağ geçidi	: Gateway.
Alıcı düğüm	: Sink node.
Benzetimli tavlama	: Simulated annealing.
Bilgi için müzakereli duyurga protokolleri	: Sensor protocols for information via negotiation.
Çatı	: Framework.
Çoğa gönderim	: Multicast.
Çok atlamalı	: Multihop.
Çoklu etkileşim	: Multiple interaction.
Çoklu transfer	: Multiple transfer.
Dedikodu	: Gossiping.
Dolaylı toplumsal iletişim yöntemleri	: Stigmergy.
Duyurga ağları	: Sensor networks.
Düğüm	: Node.
Eş güdüm	: Coordination.
Evrimsel algoritmalar	: Evolutionary algorithms.
Gezgin satıcı problemi	: Traveling salesman problem.
Göçme	: Implosion.
Hücresele robotik sistemler	: Cellular robotic systems.
Izgara	: Grid.
İleri sezgisel	: Metaheuristic.
İlgi mesajı	: Interest message.
İş bölümlenme	: Task partitioning.
Karesel atama	: Quadratic assignment.
Karınca kolonisi optimizasyonu	: Ant colony optimization.
Karınca kolonisi yönlendirmesi	: Ant colony routing.
Karınca sistemi	: Ant system.
Karınca tabanlı kontrol	: Ant based control.
Kaynak düğüm	: Source node.
Kendi kendine organize olma	: Self organization.
Kendi kendine toplanma	: Self assembly.
Kolektik zeka	: Swarm intelligence.
Olay güdümlü	: Event driven.
Olay	: Event.
Olumlu geri besleme	: Positive feedback.
Olumsuz geri besleme	: Negative feedback.
Pekiştirme	: Reinforcement.
Salınımların kuvvetlendirilmesi	: Amplification of fluctuations.
Sel baskını	: Flooding.
Sezgisel	: Heuristic.
Sırtında taşıma	: Piggybacking.
Tabu arama	: Tabu search.
Tasarsız ağlar	: Ad hoc networks.

Ters zincirleme kořu	: Reverse tandem running.
Toplanma	: Recruitment.
Toplu karar alma	: Collective decision making.
Topluluk zekası	: Swarm intelligence.
Uyarlamalı	: Adaptive.
Üst üste binme	: Overlap.
Veri merkezci	: Data centric.
Veri toplama	: Data aggregation.
Yönlendirme algoritması	: Routing algorithm.
Yönlendirmeli yayılım	: Directed diffusion.
Zamanlama	: Scheduling.
Zincirleme kořu	: Tandem running.

## **ÖZGEÇMİŞ**

Deniz Demiray 1982 yılında İstanbul'da doğdu. İlkokulu Kalamış İlkokulu'nda, ortaokulu ve liseyi Galatasaray Lisesi'nde okudu. 2005 yılında Kocaeli Üniversitesi Bilgisayar Mühendisliği Bölümü'nden fakülte birincisi olarak mezun oldu ve aynı yıl İstanbul Teknik Üniversitesi Bilişim Enstitüsü Bilgisayar Bilimleri'nde yüksek lisansa başladı.