

KABLOSUZ SENSÖR AĞI İLE KONUM BELİRLEME VE İZLEME

Elektrik-Elektronik Müh. Güngör YILDIRIM

**Yüksek Lisans Tezi
Bilgisayar Mühendisliği Anabilim Dalı
Doç.Dr. Yetkin TATAR
TEMMUZ-2012**

**T.C.
FIRAT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

KABLOSUZ SENSÖR AĞI İLE KONUM BELİRLEME VE İZLEME

YÜKSEK LİSANS TEZİ

Elektrik-Elektronik Müh. G ng r YILDIRIM

(091129103)

Anabilim Dalı: Bilgisayar M hendisliđi

Programı: Donanım

Danışman: Doç.Dr. Yetkin TATAR

Tezin Enstit ye Verildiđi Tarih: 03 Temmuz 2012

TEMMUZ-2012

**T.C.
FIRAT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

KABLOSUZ SENSÖR AĞI İLE KONUM BELİRLEME VE İZLEME

YÜKSEK LİSANS TEZİ

Elektrik-Elektronik Müh. Güngör YILDIRIM

(091129103)

Tezin Enstitüye Verildiği Tarih : 03 Temmuz 2012
Tezin Savunulduğu Tarih : 24 Temmuz 2012

Tez Danışmanı: Doç. Dr. Yetkin TATAR (F. Ü.)

Diğer Jüri Üyeleri: Doç. Dr. Mehmet KAYA (F.Ü.)

Doç. Dr. Engin AVCI (F. Ü.)

TEMMUZ-2012

ÖNSÖZ

Bu tez çalışması, Fırat Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı Yüksek Lisans Programı'nda hazırlanmıştır.

Kablosuz sensör ağları, günümüzde üzerinde çalışmayı hak eden, bir çok farklı alt konu başlığı sunmaktadır. Yapılan çalışmalar sayesinde, yakın zamanda sağlıktan savunma sanayine, trafik denetim sistemlerinden güvenlik sistemlerine kadar pek çok alanda yeniliklerle karşılaşacağımız bir gerçektir.

İlk başlarda savunma sektöründe kullanılan bu teknoloji araştırmacıların ilgisini çekmiş ve bu konu farklı branşlarda başarılı bir şekilde uygulanmıştır. Her araştırma alanında olduğu gibi bu alanda da ortaya çıkan pek çok soru, bilim adamlarına araştırma için yeni kapılar açmıştır ve açmaya da devam etmektedir.

Bu tez çalışmasında, kablosuz sensör ağlarında konum bulma teknikleri baz alınarak, hedef yer saptama ve izleme yapan bir sistem tasarımı sunulmaktadır. Bu çalışmanın yapılmasında yaptığı katkılardan dolayı değerli hocam sayın **Doç.Dr. Yetkin TATAR**'a, sayın **Dr. Mahmut Temel ÖZDEMİR**'e ve sayın **Bilgisayar Yük. Mühendisi Suna YILDIRIM**'a teşekkürü bir borç bilirim.

Ayrıca Fırat Üniversitesi Bilimsel Araştırma Projeleri Yönetim Birimi'ne FÜBAP-MF.11.42no.lu proje dâhilinde yaptığı desteklerden dolayı teşekkür ederim.

Güngör YILDIRIM

Elazığ-2012

İÇİNDEKİLER

	<u>Numara</u>
ÖNSÖZ.....	I
İÇİNDEKİLER.....	II
ÖZET	IV
SUMMARY	V
ŞEKİLLER LİSTESİ	VI
TABLolar LİSTESİ	VIII
SİMGELER LİSTESİ	IX
KISALTMALAR LİSTESİ	X
1. GİRİŞ.....	1
1.1. Genel Bilgi.....	1
1.2. Kablosuz Sensör Ağları İle İlgili Yapılan Çalışmalar	4
1.3. Tezin Amacı	7
1.4. Tezin Yapısı.....	7
2. KABLOSUZ SENSÖR AĞ MİMARİSİ, TOPOLOJİ VE TEKNOLOJİLERİ	9
2.1. Kablosuz Sensör Ağlarının Protokol Çatısı.....	9
2.1.1. Fiziksel Katman	10
2.1.2. Veri Bağı Katmanı	10
2.1.3. Ağ Katmanı	12
2.1.4. Taşıma Katmanı.....	13
2.1.5. Uygulama Katmanı.....	14
2.2. WSN Topolojileri	14
2.3. Kablosuz Sensör Düğümleri	16
2.4. Kablosuz Sensör Düğümlerinde Kullanılan Gömülü İşletim Sistemleri.....	18
2.5. Simülatörler	26
2.5.1. TOSSIM Emülatörü İle Simülasyon Geliştirme Adımları	29
3. KABLOSUZ SENSÖR AĞLARINDA KONUM BULMA.....	34
3.1. Kapsama Alanına Bağlı Konum Bulma	34
3.1.1. Gelen Sinyalin Gücüne Bağlı Mesafe Tahmini (RSSI).....	34
3.1.2. Sinyal Açısına Bağlı Konum Bulma	36
3.1.3. Geliş Süresine Bağlı Konum Bulma.....	36
3.1.4. Geliş Süresi Farkına Bağlı Konum Bulma	37
3.1.5. Parçalanamaz Çoklu Üçgenleme	37
3.1.6. Tekrarlamalı Üçgenleme	37
3.2. Kapsama Alanından Bağımsız Konum Bulma	37
4. SİSTEMİN SİMÜLASYONU VE UYGULAMASI.....	38
4.1. Simülasyon	40
4.1.1. Kenar Maskeleme Yöntemi İle Yönlendirme Simülasyonu.....	41
4.1.2. Konum Bulma ve İzleme Simülasyonu	44
4.1.3. Simülasyonun Gerçeklenmesi	45
4.2. Uygulama	49
4.2.1. Konum Belirleme ve İzleme Uygulaması	49
4.2.1.1. Açık Alanda Yapılan RSSI- Mesafe Kalibrasyon Deneyleri	51
4.2.1.2. Kapalı Alanda Yapılan RSSI – Mesafe Kalibrasyon Deneyleri.....	54
4.2.1.3. Ofsetli Dairesel RSSI-Mesafe Kalibrasyonu	61
4.2.2. Kenar Maskeleme Metoduyla Yönlendirme Uygulaması	72

4.2.3. Kapalı Ortamdaki Bazı Fiziksel Büyüklüklerin Okunması.....	75
5. SONUÇLAR VE ÖNERİLER.....	79
5.1. Sonuçlar	79
5.2. Öneriler.....	80
KAYNAKLAR.....	81
EKLER	87
TELOSB Düğüme Ait Teknik Özellikler	87
ÖZGEÇMİŞ	90

ÖZET

Kablosuz Sensör Ağlar (WSN)'da uygulama geliştirme süreci; düğümlerin ve işletim sisteminin seçilmesi, kümeleri ve küme başlarını oluşturacak ve bilgisayarla haberleşecek düğümlerin istenen işleri başaracak şekilde programlanması, bunların simüle edilmesi ve gerçek sistemin test edilmesi gibi bilgiye dayalı birçok karmaşık aşamadan oluşur.

Bu tez çalışmasında, kapalı mekanlardaki hareketli nesnelerin hem belirli bir hedefe yönlendirilmesi hem de konum tespiti veya izlenebilmesini sağlayacak WSN sisteminin interaktif bir simülasyon ve uygulama platformu şeklinde oluşturulması amaçlanmıştır.

Bunun için “*Kapalı bir ortamda araç yönlendirme ve takip uygulaması*” isimli bir senaryo düşünülmüştür. Senaryonun temel işlevi, içerisinde binlerce çeşit ticari malın bulunduğu kapalı bir depoda, istenen ürünlere bir mobil sensör düğüm aracılığı ile hızlı bir şekilde erişebilmesinin sağlanması ve mobil düğümün konumunun izlenebilmesi şeklinde tanımlanmıştır.

Hareketli nesnenin hedefe yönlendirilmesi için “*Kenar Maskeleye*” olarak adlandırılan bir yöntem önerilmiş ve uygulanmıştır. Konum tespiti için Bounding Box yöntemi kullanılmış, bu yöntemde RSSI-Mesafe ilişkisi için gerekli olan amprik denklem, deneysel çalışmalarla iyileştirilerek olumlu sonuçlar elde edilmiştir.

Gerçekleştirilen çalışmada gömülü anten yapısına sahip MEMSIC TelosB TPR2420 sensor düğümleri, Linux kernel kullanan Ubuntu 11.10 işletim sistemi, TinyOS 2.1.0 gömülü işletim sistemi ve TOSSIM simülatörü kullanılmıştır.

Anahtar Kelimeler: Kablosuz Sensör Ağları, Konum Tespiti, RSSI, TinyOS, TOSSIM

SUMMARY

Position Detection and Tracking in a Wireless Sensor Networks

A process of an application development in Wireless Sensor Networks (WSN) consists of a lot of complex stages based on knowledge, such as choice of nodes and embedded operation system, programming of the nodes, simulation and testing of the system.

In this thesis, a WSN system and an interactive simulation platform to make moving objects in indoor place both guide for desired areas and track them is planned.

For this, a scenario called “Vehicle Routing and Tracking Application” has been developed. The base function of the scenario is to access the desired areas or products in big indoor place which has a thousand of goods quickly via a mobile node and to track the mobile node.

A method named “The Edge Masking Technique” has been proposed for guidance of moving objects. Bounding Box method has been used for position detection. Equations which are necessary for RSSI-Distance relation have been obtained by experimental studies and achieved positive results.

In the study carried out, MEMSIC TelosB TPR2420 sensor nodes, Ubuntu 11.10 operation system, TinyOS 2.1.0, TOSSIM simulator and Tkinter GUI library for visualization of the simulation of the system have been used.

Keywords: WSN, Position Detection, RSSI, TinyOS, TOSSIM

ŞEKİLLER LİSTESİ

	<u>Numara</u>
Şekil 1.1.Kablosuz sensör ağı ve sensör düğümün genel yapısı	2
Şekil 2.1.WSN katman yapısı.....	9
Şekil 2.2. Örnek bir WSN [1].....	12
Şekil 2.3. Örnek bir ağaç topoloji.....	14
Şekil 2.4.Örgü Topoloji.....	15
Şekil 2.5. Sıradüzensel Topoloji.....	15
Şekil 2.6. TelosB düğüm yapısı.....	17
Şekil 2.7.Bileşenler arası genel etkileşim.....	21
Şekil 2.8. Modül genel yapısı [8]	21
Şekil 2.9.Ara yüz yapısı [8]	22
Şekil 2.10. Yapılandırıcı örneği[8].....	23
Şekil 2.11. Yapılandırıcı genel tanımlama	23
Şekil 2.12. Yapılandırıcıların kullanılması.....	24
Şekil 2.13. Derleme işlemleri	24
Şekil 2.14. IEEE 802.15.4 Çerçeve Yapısı.....	25
Şekil 2.15. T-Frame Yapısı.....	26
Şekil 2.16. I-Frame Yapısı.....	26
Şekil 2.17. MspSim ile Blink simülasyonu	28
Şekil 2.18. VisualSense ile alınan sinyal gücü simülasyonu	28
Şekil 2.19. Örnek bir başlık dosyası	29
Şekil 2.20. Bir düğüm dosyasının yapısının genel gösterimi	30
Şekil 2.21. Simülasyon dosyasının akış şeması.....	30
Şekil 2.22. Topoloji dosyasının okunmasına ait örnek bir kod bloğu	31
Şekil 2.23. Gürültü dosyasının simülasyona eklenmesi ait örnek bir kod bloğu.....	31
Şekil 2.24. a)Başlangıç ayarları b) Düğümlere bağlanma	32
Şekil 2.25. Örnek TOSSIM çıktısı.....	32
Şekil 2.26. Simülasyon penceresinin çizilmesine ait örnek bir kod bloğu	33
Şekil 3.1. Bounding Box yöntemi	35
Şekil 4.1. Uygulamanın yapıldığı alan ve düğümlerin yerleşimi	39
Şekil 4.2.İki sensör düğüm ile hedef saptama	41
Şekil 4.3. Mobil düğümün ortamdan veri toplaması ve topladığı bilgileri yayınlaması	42
Şekil 4.4.Yönlendirme Bilgisini Sağlayan Koda ait akış şeması	44
Şekil 4.5. Simülasyonda kullanılan RSSI-Mesafe ilişkisi.....	45
Şekil 4.6. Simülasyon ekranı	46
Şekil 4.7. Araç B21 bölgesinden B26 bölgesine hareket etmesi	47
Şekil 4.8. Aracın B26 bölgesine ulaşması	47
Şekil 4.9. Aracın B26 bölgesinde iken B33 bölgesine gitmesi için merkezden paket alması ve harekete geçmesi.....	48
Şekil 4.10. Aracın B33 bölgesinden B12 bölgesine paket alması ve harekete geçmesi.....	48
Şekil 4.11. Java programının ekranın görüntüsü	50
Şekil 4.12.Arazide ölçüm	51
Şekil 4.13. Açık havada tabure üzerinde yapılan ölçüm grafiği.....	53
Şekil 4.14. Açık havada elde taşınarak yapılan ölçüm grafiği	53
Şekil 4.15.Açık alanda hatalı saptama.....	54
Şekil 4.16.Bina içi yatay ölçüm grafiği	56

Şekil 4.17.Dairesel Kalibrasyon Uygulama Metodu	56
Şekil 4.18. Uygulama ortamı.....	57
Şekil 4.19.Dairesel kalibrasyona göre uzaklık – RSSI grafiği a) 0-180 ⁰ için b) 0-360 ⁰ için	58
Şekil 4.20. a) Çapa1 Yatay kalibrasyon sapma b) Çapa1 Dairesel kalibrasyon sapma	59
Şekil 4.21. a) Çapa2 Yatay kalibrasyon sapma b) Çapa2 Dairesel kalibrasyon sapma	60
Şekil 4.22. a) Çapa3 Yatay kalibrasyon sapma b) Çapa3 Dairesel kalibrasyon sapma	60
Şekil 4.23. a) Çapa4 Yatay kalibrasyon sapma b) Çapa4 Dairesel kalibrasyon sapma	60
Şekil 4.24. Kısa ölçümden dolayı kesişimin sağlanamaması durumu.....	62
Şekil 4.25.Ofset değerin eklenmesi ile kesişimin elde edilmesi.....	62
Şekil 4.26.Yeterli kesişimin olması durumu	63
Şekil 4.27.Ofset değeri çıkarılması durumunda kesişim alanın küçültülmesi.....	63
Şekil 4.28.Çapa1 için yaklaşım grafiği.....	66
Şekil 4.29.Çapa2 için yaklaşım grafiği.....	66
Şekil 4.30.Çapa3 için yaklaşım grafiği.....	67
Şekil 4.31.Çapa4 için yaklaşım grafiği.....	67
Şekil 4.32. Konum 1 için a) Dairesel Kalibrasyon b) Yatay Kalibrasyon	68
Şekil 4.33. Konum 2 için a) Dairesel Kalibrasyon b) Yatay Kalibrasyon	69
Şekil 4.34. Konum 3 için a) Dairesel Kalibrasyon b) Yatay Kalibrasyon	70
Şekil 4.35. a) Konum 5 için a) Dairesel Kalibrasyon b) Yatay Kalibrasyon	71
Şekil 4.36. Maske 15 ‘de B25 bölge kontrolü yapan kod parçası	72
Şekil 4.37. Yatay olarak 2 nolu düşey bölge bildirimi – sadece 2 nolu yeşil led yanıyor.....	73
Şekil 4.38.Üst bölgeye geçme bildirimi - tüm ledler yanıyor	74
Şekil 4.39. Alt bölgeye geçme bildirimi – tüm ledler sönük.....	74
Şekil 4.40. BaseStation üzerinden veri gönderme için yazılan program ekranı.....	75
Şekil 4.41. Deneme ortamından bir görüntü.....	75
Şekil 4.42. Sensör okumaları için a) Yapılandırıcı ayarları b) Okuma kod satırı	76
Şekil 4.43. Sıcaklık bilgisini okunması (x eksen=Zaman, y eksen = ⁰ C)	77
Şekil 4.44. Nem bilgisinin okunması (x eksen=Zaman, y eksen = % nem).....	77
Şekil 4.45. Işık bilgisinin okunması	78
Şekil 4.46. İşlemci bacağındaki voltaj seviyesinin okunması (x eksen=Zaman, y eksen = Voltaj).....	78

TABLÖLAR LİSTESİ

	<u>Numara</u>
Tablo 4.1. Mobil düğümün periyodik olarak yayınladığı payload	42
Tablo 4.2. 9 nolu maske düğümün karşılaştırma tablosu	43
Tablo 4.3. Maske düğümün mobil düğüme gönderdiği veriler	43
Tablo 4.4. Açık alan ölçüm sonuçları	52
Tablo 4.5. Bina içi yatay ölçüm	55
Tablo 4.6. Dairesel Kalibrasyon	57
Tablo 4.7 Yatay ve dairese kalibrasyon sonuçları ile gerçek değerleri gösteren tablo	59
Tablo 4.8. Ofset durumunu belirlemek için yapılan ölçüm sonuçları	64
Tablo 4.9.Genel ölçüm sonuçları	65

SİMGELER LİSTESİ

Bij	: Çalışma alınanındaki bölge kodları
Dmn	: İstenilen bölge kodu
Hu	: Hücre uzunluk
Hg	: Hücre genişlik
Dr	: Düğüm bina içi kapsama alanı
Mt	: Metre
Cm	: santimetre
Dbm	: Desibel
⁰ C	: Santigrat
V	: Volt
W	: Watt
J	: Joule
mW	: Mili Watt
bps	: Saniyedeki bit iletimi
Hz	: Frekans Birimi,Hertz
KHz	: Bin Hertz
MHz	: Milyon Hertz
GHz	: Milyar Hertz
KB	: Kilobayt
MB	: Megabayt
Rx	: Alma modu
Tx	: İletim modu

KISALTMALAR LİSTESİ

WSN	: Kablosuz Sönsör Ağı
RSSI	: Alınan Sinyal Gücü Göstergesi
RF	: Radyo Frekansı
ADC	: Analog Dijital Dönüştürücü
RAM	: Rastgele erişimli hafıza
DPM	: Devingen güç yönetimi
DVS	: Devingen voltaj yönetimi
MAC	: Ortam Erişim Yönetimi
QoS	: Hizmet Kalitesi
SAR	: Ardışık Atama Yönlendirmesi
SMP	: Sensör Yönetim Protokolü
OS	: İşletim Sistemi
CSMA	: Çarpışma Algılayıcıyla Taşıyıcı Dinleyen Çoklu Erişim
TDMA	: Zaman Bölmeli Çoklu Erişim
FDMA	: Frekans Bölmeli Çoklu Erişim
FEC	: İleri Yönlü Hata Kontrolü
ARQ	: Otomatik Tekrar İsteği
UDP	: Kullanıcı Datagram Protokolü
TCP	: İletim Kontrol Protokolü
IP	: İnternet Protokolü
SQLT	: Sensör Sorgulama ve Görevlendirme Protokolü
DFS	: Derin İlk Araştırma
BFS	: Saha İlk Araştırma
FSK	: Frekans Kaydırma Anahtarı
ISM	: Endüstriyel, bilim ve medikal radyo bandı

1. GİRİŞ

1.1. Genel Bilgi

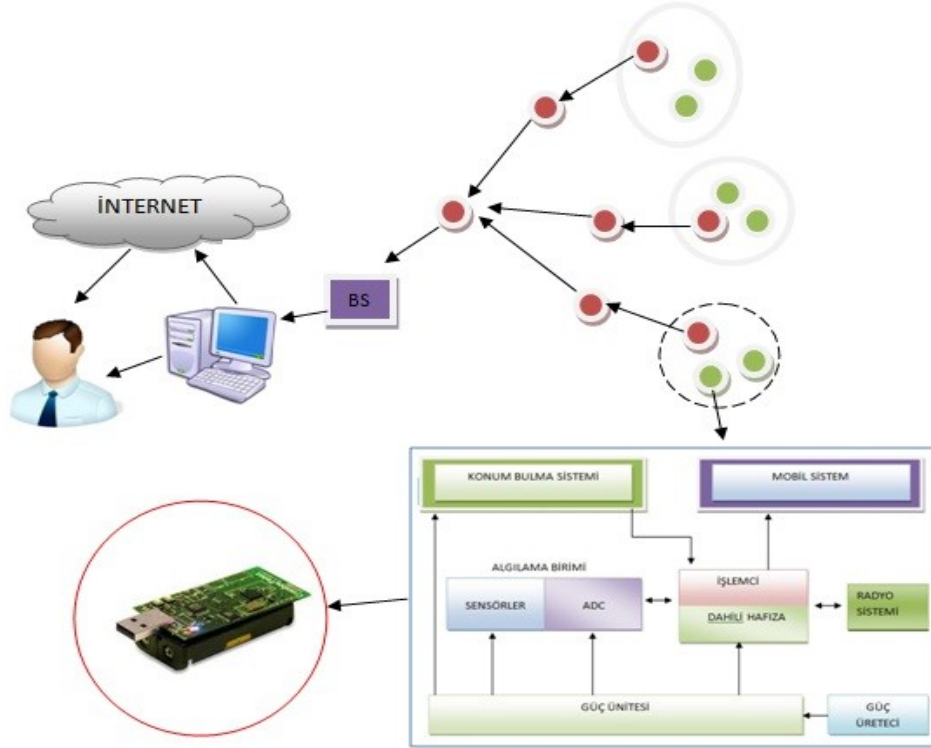
Günümüz teknolojisinin ilgi çeken araştırma konularından biri hiç kuşkusuz kablosuz sensör ağlarıdır. Kablosuz sensör ağları (WSN) birbirleriyle RF (Radyo Frekanslı elektromanyetik dalga) iletişim ortamında 802.15.4 ağ iletişim protokoluna göre haberleşen birçok sensör düğümünden oluşmuş kablosuz bir ağ yapısıdır [1,5,8].

Kullanılan kablosuz sensör düğümleri; mikro sensörler ve ADC'ler, işaretlerin ön işlenmesi ve telsiz iletişimi protokollerinin yürütülmesini sağlayan, üzerinde işletim sistemi barındıran mikroişlemci ile RF ünitesi ve güç biriminden oluşmaktadır [2,15,16]. Sensör düğümleri algıladıkları ortamsal ham verileri, mikroişlemcileri sayesinde ön işlemeyen geçirip kablosuz ağ ortamında bulunan diğer düğümlere iletebilirler.

Sensör düğümlerinin yerleşim şekillerinin önceden hesaplanmasına gerek yoktur. Bu durum aynı zamanda WSN protokol ve algoritmalarının kendi kendini organize etme yeteneğine sahip olmaları anlamında gelmektedir. Bu özellikler WSN'lere, ulaşılabilir bölgelerde ve afet bölgelerinde kullanım esnekliği kazandırır. Kablosuz sensör ağlarının kullanım alanlarının bir kısmı şunlardır [2];

- Habitat izleme ve çevresel gözlem, hava durumu tahmini sistemleri
- Sağlık uygulamaları (hasta, doktor takibi v.b)
- Enerji tedarik ve aktarma sistemleri
- Ev ve ofis uygulamaları
- Uzak yerlerdeki olayların çözülmesi(orman yangını tespiti, vb.)
- Trafik akışının gözlenmesi ve bu gözlemlere dayalı yönlendirme
- Taşıt park yerlerindeki boş ve dolu alanların tespiti
- Güvenlik, bina ve endüstriyel otomasyon
- Otomobil arıza bildirim sistemleri
- Askeri amaçlar ve terörist saldırı tespiti

Bir kablosuz sensör ağı, temel olarak sensör düğümleri, alıcı (sink) düğüm ve alınan bilgileri işleyip değerlendiren bilgisayar yazılımlarından oluşur [1,5,14]. Her bir dağıtık sensör düğümü, veri toplayabilme ve verileri alıcı (sink) düğüme geri yönlendirebilme yeteneğine sahiptir. Geri yönlendirme işlemi çok atlamalı (multihop) olarak yapılır. Sink düğüm, kullanıcı ile internet veya uydu iletişimi yoluyla bağlantı kurabilme yeteneğine sahip olabilir.



Şekil 1.1.Kablosuz sensör ağı ve sensör düğümünün genel yapısı

Bir kablosuz sensör ağı ve düğümünün genel yapısı Şekil 1.1’de gösterildiği gibidir. Bu ağ yapısında sensör düğümler birbirleri ile belirli bir hiyerarşiye göre haberleşip, merkezi birime topladıkları bilgileri iletmektedirler. Kablosuz sensör ağlarında kullanılan düğümler genel olarak 4 ana kısımdan meydana gelir; sensör (algılama) birimi, işlemci birimi, alıcı – gönderici (Telsiz) birimi ve güç birimi. Sensör düğümleri ayrıca görev tanımına bağlı olarak, *konum bulma sistemi*, *güç üretici*, *mobil sistem* de ihtiva edebilir [1,2,15].

Sensörler tarafından elde edilen analog bilgi ADC ile sayısalı çevrilir ve bu bilgi işlem birimine gönderilir. İşlem birimi gerekli veri işleme işlevini yerine getirerek iletilecek veriyi küçük bir hafıza ortamında saklar. Alıcı-verici (transmisyon) birimi düğümü ağına bağlar.

En önemli birimlerden birisi de hiç kuşkusuz güç birimidir. Sensörün yaşam ömrünü ve ağın kararlı çalışmasını birinci derecede etkiler. Güçbiriminin kalitesi kadar, düğümde çalışan gömülü işletim sisteminin de düğümün yaşam ömrüne önemli etkisi vardır.

Yönlendirme ve algılama görevi, yüksek doğruluk oranına sahip konum bilgisine gereksinim duyabilir. Ayrıca bazı sensörlerin hareket edebilme ihtimallerine karşın, taşınabilir olma esnekliği sağlayan Mobil sistem'e ihtiyaç duyabilir.

Enerji bakış açısından yaklaşıldığında, en uygun bellek çeşitleri mikro denetleyici çipi üzerindeki bellek (Dahili Bellek) ve FLASH belleklerdir. Çip dışı RAM'ler seyrek veya hiç kullanılmamaktadır [13-15]. FLASH bellekler, maliyeti ve depolama kapasitesi nedeniyle kullanılmaktadır. Bellek gereksinimleri yüksek oranda uygulama bağımlıdır.

Çok sayıda algılama düğümünden gelen veriler ağın verimini düşürebilir. Bu problemi çözmek için, bazı düğümler küme başları yapılır. Küme başları, veriyi toparlayıp, gerekli hesaplamalar yaparak (ortalama, toplam, en yüksek, vb.) elde ettiği özetleri yayınlar[1,4].

Sensör ağlarının dizaynı yapılırken, etkilendiği bazı faktörleri hesaba katmak gerekir. Bu faktörlerin en önemlileri; hatatoleransı, ölçeklenebilirlik, üretim maliyeti, işletme ortamı, sensör ağ topolojisi, donanım kısıtlamaları, iletim medyası ve güç tüketimidir[1,14].

Hata toleransı, herhangi bir sensörün arızası durumunda ağın fonksiyonlarını yerine getirebilme yeteneğidir. Bazı sensörler güç azlığından dolayı veya çevresel şartlardan arızalanabilir. Bu şekilde hata veren sensörler ağın sağlıklı şekilde çalışmasını engellememelidir[2,5,16].

Sensör sayısı milyonlar ile bile ifade edilebileceğinden, sensör ağlarını belirli bir alandaki yoğunluğu çok fazla olabilir. Sensör ağları, sisteme yeni sensörlerin dahil olmasını veya bazı sensörlerin sistemden çıkmasını algılayabilmeli ve her iki durumda gerekli ölçeklendirmeyi yapabilecek yeteneğe sahip olmalıdır[1,2,5].

Sensör başına düşen üretim maliyetinin düşük olması çok önemlidir. Sayısı binlere hatta milyonlara varabilen sensör düğümlerinin olabileceği düşünüldüğünde, sistemin genel maliyetinin düşük olması için düğüm başına düşen parasal değere dikkat etmek gerekir.

Bu kadar yüksek sensör yoğunluğuna sahip olma ihtimalinden dolayı topoloji çok dikkatli bir biçimde seçilmelidir.İletişim kablosuz medya ile sağlanmalıdır. Bu da radyo sinyalleri, infrared veya optik şeklinde olacaktır. Ancak genel kullanım şekli radyo sinyalleri ile iletişimidir. Radyo sinyalleri genelde ISM bandındadır (2.4 GHz,915MHz gibi) [2,3,15,16].

Güç kaynağı, sensör düğümünün en önemli parçalarının başında gelir ifadesi yanlış olmayacaktır. Çünkü güç kaynağı sensörün yaşam ömrünü belirlediğinden, batarya cinsi önemlidir. Ayrıca çok sıçramalı WSN’de, bazı düğümler yeniden paket yönlendirme, yeniden organize olma gibi özelliklere sahip olacağından bazı sensörlerin güç ihtiyaçları diğerlerinden fazla olabilir [1-3]. Güç tüketiminin bu kadar önemli olduğu bir durumda, güç tasarrufu ve güç yönetimi büyük önem arz eder. Son zamanlarda güç tüketimi farkındalığına sahip protokol ve algoritmalar üzerinde çalışmalar yapılmaktadır. Güç tüketimi, algılama, iletişim ve data işlemi ayrı olarak incelenmektedir. Bir sensör düğümünde komut başına düşen enerji sarfıyatı bir kriterdir. Örneğin, 1 KB veriyi 100 metrelik bir uzaklığa iletmek için gereken enerji, yaklaşık olarak saniyede 100 milyon komut işleyen bir işlemcide 3 milyon komut işlemek için gereken enerjiye eşittir [1].

Kullanılan en önemli iki güç koruma politikası Devingen Güç Yönetimi (Dynamic Power Management-DPM) ve Devingen Voltaj Ölçeklendirme (Dynamic Voltage Scaling-DVS) 'dir [1]. DPM yazılımı, uyku ve askıda modlarında, küçük bir gecikme ile clock bilgisini ve işlemci çekirdek voltajını ayarlar. DVS ise cihaza çalışma anındaki iş yüküne uygun olarak işlemci voltaj ve frekansını ayarlamaya izin verir [1].

1.2. Kablosuz Sensör Ağları İle İlgili Yapılan Çalışmalar

2000’li yılların başlarından itibaren kablosuz sensör ağları üzerine birçok üniversite ve yatırımcı firma ar-ge çalışmaları yapmıştır. Bu çalışmalar iki kısımda gelişmektedir. Birincisi sensör düğümler için mimari, işletim sistemi v.b iyileştirmeler için yapılan araştırmalardır. İkinci kısım ise değişik sektörlerde kablosuz sensör ağ uygulamalarının çoğaltılması şeklindedir. Bunlardan bazıları aşağıda kısaca verilmiştir.

WSN’leri araştırmak amacı ile Güney California Üniversitesi tarafından geliştirilen SCADDS projesinde; radyo iletişim yığıtları, zaman senkronizasyonu ve WSN’ler için kablosuz ağ erişim protokollerinin geliştirilmesine çalışılmıştır. Bu projede özellikle ortama erişim katmanında yapılan iyileştirmeler ile güç tasarrufu sağlanmıştır [4,27].

Sensör ağlarının en önemli amaçlarından biri mümkün olduğu kadar küçük sensörler yapmaktır. Bunun için kullanılacak bataryanın cinsi ve boyutu büyük önem taşımaktadır. Massachusetts Teknoloji Enstitüsü μ AMPS Proje Grubu bunun için sensörün kendi etrafındaki kaynaklardan enerji elde etmesi için çalışmalar üzerine yoğunlaşmıştır [28].

Princeton Üniversitesi, Kenya’da vahşi yaşamdaki zebraların davranışlarını incelemek için, ZebraNET projesini hayata geçirmiştir. Bu projede GPS kullanılarak, büyük bir alanda

zebraların konumlarını saptamıştır. Bu çalışmada IMPALA adında özel bir gömülü işletim sistemi kullanılmıştır [29]. IMPALA, sürekli radyo sinyali dinlemektense, periyodik olarak düğüm keşfi işlemini gerçekleştirerek enerji tasarrufu sağlamaktadır.

Berkley Üniversitesinde çok düşük maliyetli, az enerji ve beraberinde düşük güç tüketimine sahip, kendi kendini konfigüre edebilen kablosuz sensör ağları için SOC (system-on-chip) dizaynlar üzerine çalışma yapılmıştır [4].

Lancaster Üniversitesi, Karlsruhe Üniversitesi, ETH Zürih, VTT Electronics tarafından, ev içi nesnelerin birbirleri ile daha hızlı ve iyi etkileşimini sağlamak amacıyla bir ortak proje geliştirilmiştir. Bu projede kablosuz iletişim ortamı olarak Bluetooth, RF kullanılmıştır. Projede aynı zamanda RFID etiketleri üzerinde araştırmalar gerçekleştirilmiştir [4].

CENS(The Center for Embedded Network Sensing) projesinde ad-hoc olamayan büyük ölçekli sensör sistemleri üzerinde çalışılmıştır. Çalışmalar, habitat, sismik izleme, kirlilik izleme gibi farklı alanları ihtiva etmektedir. Özellikle deprem üzerine yapılan çalışmalar dikkat çekicidir [30].

Rice Üniversitesinde GNOMES (Generalized Network of Miniature Environmental Sensors) adında sensörler geliştirilerek, bu sensörler çevresel ve meteorolojik dataların toplanmasında kullanılmıştır [31].

Birçok uygulama QoS hizmeti isteyebilir. Sensicast System, RF girişimlerinin ve “çok yol” zayıflamasının olduğu zor çevre şartlarında çalışabilecek bir kablosuz sensör ağını geliştirmiştir. Burada QoS metrikleri olarak komşu düğümler arası sinyal kuvvetleri baz alınmıştır [4].

Kablosuz sensör ağ uygulamalarına ait örnek çalışmalar aşağıda verilmiştir.

Tarımsal modernizasyon ve tarımsal çevre korumasını anlamak için, kablosuz sensör ağlarına dayalı bir tarımsal çevre izleme sistemi dizayn edilmiş ve bu şartlara uygun sensör düğümleri için donanım dizaynları geliştirilmiştir [32].

WSN teknolojilerini kullanan bir akıllı ilaç tedavi sistem dizaynı geliştirilmiştir [33]. Tedavi hatırlatmasını içeren sistemin birincil fonksiyonları, hap-dağıtımının yardımı ve kronik rahatsızlığı olan hastalar için tedavi kayıdır. Birçok yaşlı insanın birlikte yaşadığı bakım evinde kolayca uygulanabilen sistem, bir ana panel (MP) ve yedi taşınabilir akıllı hap-kutularından (SPB) oluşur.

Köprüler, tüneller gibi inşaat mühendisliği altyapı konularının izlenmesinde kablosuz sensör ağlarının kullanılmasını öneren çalışmalar gerçekleştirilmiştir [34]. Tünel ve köprü gözlemleri ve alınan verilerin işlenmesi ele alınmıştır.

WSN'lere dayanarak, evde ya da hastanede tedavinin kalitesini artırmak için bir sağlık hizmeti sistemi tasarlanmıştır [35]. Giyilebilir düğümlerinin yapısı, sistemdeki kablosuz multimedya düğümleri üzerine çalışmalar yapılmıştır. Ayrıca, hastalar için lokalizasyon metodunun tipi ve enerji-verimli transmisyon stratejisi de anlatılmıştır.

Yanıcı ve patlayıcı gazların tespiti için kablosuz gaz sensör ağının (WGSN) karakterizasyonunu ve geliştirilmesi üzerine yapılan çalışmada bir yarıiletkenmikro sensör kullanarak erken gaz tespiti elde edilmiştir [36]. Acil durumda, ağ koordinatörü GSM/GPRS veya bir Ethernet ağı aracılığıyla alarm verir ve kablosuz uyarıcı vasıtasıyla gaz emisyonunu bağımsız olarak kontrol eder.

Otlama süresince sürüdeki bir koyunun kafa hareketleri ve bu hareketler ile üretilebilecek enerjinin WSN düğümünün enerji kaynağı olarak kullanılması amaçlanmıştır. Bu problem çözümünde Lagrange-d'Alembert prensibini uygulanmıştır [37].

Kömür madenindeki var olan izleme sistemlerindeki eksikliklere çözüm bulmak için, kablosuz sensör ağları ve diğer teknolojilerden oluşmuş entegre maden ağı dizayn edilmiştir [38].

Orman yangınlarının erken tespiti ve takibi için bir "low-rate wireless personal area network" tabanlı sensör ağı sunulmuş ve aynı zamanda yangının toprak üstü ve toprak altı etkileri izlenmiştir [39].

VDS240 adında kablosuz sensör ağı geliştirilmiş ve bu ağla lokal bir trafik kontrol uygulaması gerçekleştirilmiştir [40]. Rampa tıkanıklıkları, park problemleri, anayol tıkanıklıkları gözlemlenmiş ve California ana yolu üzerinde denenmiştir.

Hastanelerde WSN kullanarak akıllı bir personel ve hasta takibi uygulaması yapılmıştır [41]. Hastaların ilaç zamanları, sağlık ve güvenlik personellerinin gözetimi bu akıllı sistem tarafından gerçekleştirilmiştir.

Sınır güvenliği ve izinsiz içeri sızmaları tespit edebilecek bir WSN sisteminin tasarımı üzerine yapılan çalışma, özellikle askeri alanda önemli bir savunma mekanizması önermektedir [42].

İsmi arıların zig-zaglı karmaşık hareket yapılarından alan "ZigBee" teknolojisi, genelde akıllı ev sistemleri, güvenlik sistemleri gibi günümüzde popüler alanlarda

kullanılan bir kablosuz ağ teknolojisidir. Uluslar arası bir şirket birliği olan “ZigBee Alliance”ın çalışmaları sonucunda geliştirilmiş bir projedir [9,17].

1.3. Tezin Amacı

WSN’lerde uygulama geliştirme süreci; düğümlerin ve işletim sisteminin seçilmesi, kümeleri ve küme başlarını oluşturacak ve bilgisayarla haberleşecek düğümlerin istenen işleri başaracak şekilde programlanması, bunların simüle edilmesi ve gerçek sistemin test edilmesi gibi bilgiye dayalı birçok karmaşık aşamadan oluşur.

Bu tez çalışmasında,

- Kablosuzsensör ağlarının avantajlarından faydalanılarak, GPS’in yetersiz kaldığı büyük kapalı alanlarda (indoor) konum tespiti yapılabilmesi,
- Kapalı mekanda bulunan araç ve çalışanların istenilen hedef bölgelere rahat ulaşmasını sağlayacak bildirimlerin verilebilmesi,
- Gerekligiörülen yerlerde sıcaklık, nem, ışık gibi fiziksel değerlerin okunabilmesi ve çalışma ortamı hakkında istatistiksel verilerin toplanabilmesi,

için hem yönlendirme hem de takip işlemlerini (konum tespiti) kablosuz sensör ağlarını kullanarak yapabilen bir interaktif simülasyon ve uygulama platformunun gerçekleştirilmesi açıklanmıştır. Bunun için “*Kapalı bir ortamda araç yönlendirme ve takip uygulaması*” isimli bir senaryo seçilmiş olup, senaryonun pratikte uygulanabilir olmasına özen gösterilmiştir. Uygulama senaryosunun temel işlevi; içerisinde binlerce çeşit malın bulunduğu büyük kapalı alanları kapsayan depolarda hem mobil araçların aranacak malzemeye hızlı bir şekilde ulaşılabilmesini sağlayacak hemde mobil araçların depo içerisindeki yerlerini merkez ofise bildirebilecek bir WSN yapısı oluşturulması şeklindedir.

1.4. Tezin Yapısı

Bu tez çalışması, 5 ana bölümden oluşmaktadır.

Birinci bölümde, kablosuz sensör ağlarının önemi vurgulanarak genel yapısı ve literatür çalışmaları hakkında bilgi verilmiştir.

İkinci bölüm, kablosuz sensör ağlarında kullanılan protokol yapısı, düğüm teknolojileri, işletim sistemleri ve simülatörleri hakkında temel bilgileri açıklanmıştır.

Üçüncü bölümde, konum bulma teknikleri hakkında bilgi verilmiştir.

Dördüncü bölümde, tezde önerilen sistemin genel açıklaması ile geliştirme adımları detaylı olarak açıklanmıştır.

Beşincibölümde, gerçekleştirilen çalışmalar ve elde edilen sonuçlar değerlendirilerek ileride yapılacak çalışmalar için önerilerde bulunulmuştur.

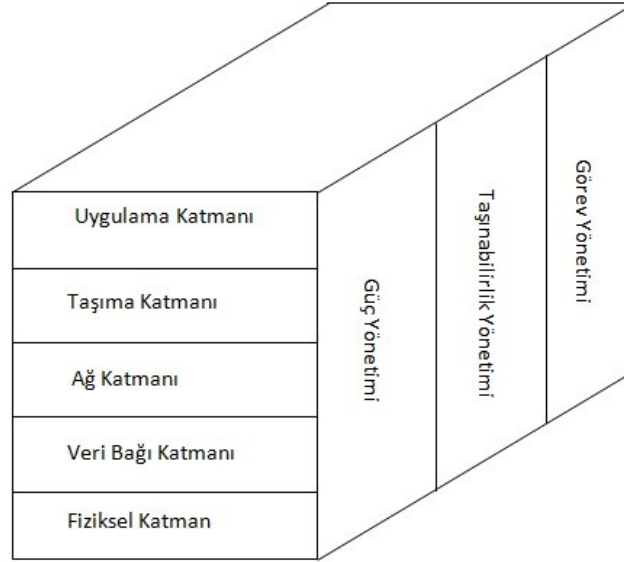
Bu tez çalışması Fırat Üniversitesi Bilimsel Araştırma Proje Birimi tarafından FÜBAP-MF.11.42 no.luproje ile desteklenmiştir.

2. KABLOSUZ SENSÖR AĞ MİMARİSİ, TOPOLOJİ VE TEKNOLOJİLERİ

Bu bölümde, kablosuz sensör ağlarının protokol çatısı, kullanılan topolojiler, düğüm teknolojileri, düğümler üzerinde çalışan gömülü işletim sistemleri ve kullanılan simülatörler açıklanmıştır.

2.1. Kablosuz Sensör Ağlarının Protokol Çatısı

Kablosuz sensör ağları OSI yapısına benzer bir protokol çatısı kullanır. Şekil 2.1’de kablosuz sensör ağlarındaki bu katman yapısını göstermektedir [1].



Şekil 2.1. WSN katman yapısı

Bu katmanların yapıları daha sonra detaylandırılacaktır ancak kısaca özetlemek gerekirse:

- Fiziksel katman; güçlü modülasyon, iletme ve alma tekniklerini barındırır.
- Veri bağı katmanı; ortamdaki gürültü ve düğümlerin taşınabilir olma özelliklerinden dolayı hem güç tüketim farkındalığına sahiptir hem de çarpışmaları minimize eder.
- Ağ katmanı; yönlendirmeden sorumludur .
- Taşıma katmanı; data akışının sürekliliğini sağlar.
- Görev tanımına bağlı olarak uygulama katmanında farklı türlerde yazılımlar kullanılır.

Bunlara ek olarak, güç, taşınabilirlik ve görev yönetim düzlemleri diğer katmanlarla paralel olarak çalışır. Güç yönetim düzlemi, sensörün gücünü nasıl kullandığını kontrol eder. Örneğin boştaçalışma durumunda uyku moduna geçme veya mesaj yinelenmesi olmasın diye kendi mesajını gönderdikten sonra tekrar uyku moduna girmesi gibi. Taşınabilirlik düzlemi, sensörün hareket durumunu algılar ve buna göre yeni komşularını algılayarak yönlendirme rotasını yeniden oluşturur. Görev yönetim düzlemi, verilen belirli bir bölgede algılama yaparak hangi sensörlerin o anda algılama yaptıklarını tespit eder. Beraber çalışma modunu sağlayarak gücün ve kaynakların verimli şekilde kullanılmasını sağlar.

2.1.1. Fiziksel Katman

Frekans seçimi, taşıyıcı frekans üretimi, sinyal algılama, modülasyon ve veri şifreleme ile ilgilenir. “Çok sıçramalı” iletişimde, gölgeleme ve yol kaybı, sensör yoğunluğu ile artar. Bu durumda uygun modülasyon seçimi önem arz eder (M-ary, Binary, PPM kullanan UWB gibi.). En uygun modülasyon türü bu katmanda belirlenir[1,17].

2.1.2. Veri Bağı Katmanı

Veri çerçeve yapısı algılama, veri akışını çoğullama, ortama erişim ve hata kontrol denetimi bu katmanın görev alanı içindedir. Bu katman güvenli bir noktadan-noktaya veya noktadan-çok noktaya iletişimi garanti eder. Bu katman iki önemli alt bölüme sahiptir, MAC ve MEC[1,7,17].

Kendi kendini organize edebilen WSN’lerde MAC protokolü iki önemli amacı elde etmelidir. Birincisi, çok sayıdaki düğümün ağ alt yapısının oluşumu, ikincisi kaynak paylaşımının en verimli şekilde paylaşılmasını sağlamak.

Diğer kablosuz ağlarda kullanılan MAC yapısı, WSN’lerde kullanılamaz. Çünkü hücresel sistemde, temel istasyon kablolu bir omurgadır ve mobil bir düğüm en yakın istasyona sadece bir sıçrama uzaklığındadır. Tabii ki böyle bir istasyona sahip sistemlerde güç tüketimi ikinci planda olacaktır ve de mobil kullanıcılar genelde bataryalarını şarj edebilir durumdadırlar. Oysa kablosuz ağlarda böyle bir merkezi istasyon bulunmamakla birlikte, düğümlerin güç tüketimleri dikkatli bir şekilde kontrol edilmelidir. Güç tüketiminin ağın yaşam süresini doğrudan etkilediği unutulmamalıdır. Bu zorluklar ağ içerisinde senkronizasyon sorunlarının da beraberinde getirir [1-3,15,16]. Örneğin, Bluetooth ve MANET(Mobil ad-hoc) birbirine en yakın kablosuz iletişim sistemleridir. Bluetooth,

kısa mesafeli ve alt yapısız RF sistemidir. Ortada bir *merkez* ve ona bağlı olan *bağımlı* düğümlerden oluşan yıldız topolojiye sahiptir. Bu bir “Piconet” formudur [4]. Piconet, TDMA(time-division) ve frekans atlama tekniklerini kullanır ve 10 metrelik bir çap içerisinde kullanılır. MANET ise bir alt yapıya sahiptir ve bu amaçla mobil sistemlerde yüksek QoS hizmetlerinin sunulmasını sağlar. Kullanıcı bataryaları değiştirilebilir veya şarj edilebilir olduğundan güç tüketimi ikinci planda bir konudur. Bu iki sistemin aksine WSN’ler çok fazla düğüme sahiptir. Topoloji değişimi çok sık olduğundan hata oluşum riski yüksektir. Bütün bu nedenlerden dolayı güç tüketimi farkındalığına sahip bir MAC protokolünün kullanılması gereklidir.

Bunun için önerilen bazı protokoller vardır [1].

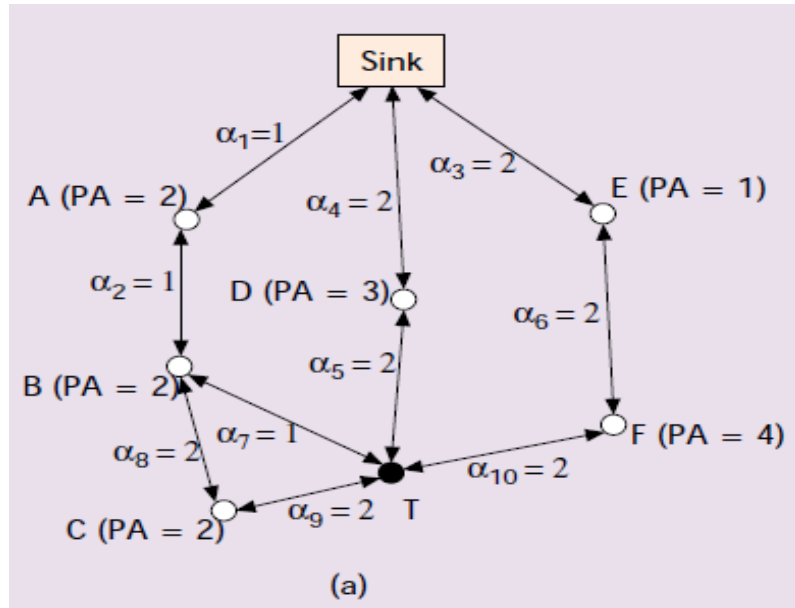
- Kendi kendini organize edebilen sensör ağları için “SMACS” ve “EAR”. Bunlardan SMACS, dağıtık alt yapıli protokoldür ve düğümlerin herhangi bir master düğüme ihtiyaç duymadan komşularını keşfetme ve iletişim gereksinimlerini sağlama imkânı sunar. Komşu keşfi ve kanal atama işlemleri aynı zamanda yapılır. EAR protokolü, kesintisiz bir hizmet sunmayı amaç edinir. Burada her bir düğüm, bağlantının yapılması ve koparılmasına kendisi karar verir. Buğ yapısında her düğümün etrafında belli sayıda düğüm olduğuy varsayılır. En önemli sıkıntısı birbirine komşu olan düğümlerin aslında farklı alt ağların üyeleri olmasıdır.
- Geleneksel CSMA’nın dışında, WSN’de periyodik trafiğın işleyişini takip edebilen ve buna göre dinleme ve yeniden gönderim bileşenlerini kullanabilen kablosuz sensör ağlarına uygun bir CSMA yapısı olmalıdır. Bu yapıda her düğüm eşit haklara sahiptir. Veri gönderim hızı daha yavaş olan düğümler tespit edilerek diğer düğümlere bildirilmelidir.
- *Hybrid TDMA/FDMA*, 10 metre çapından küçük bir alan içerisinde enerji kısıtlılığına sahip sensörlerin kendilerine en yakın ve en yüksek güce sahip olan düğümlerle iletişime geçtikleri varsayılır. Veri gecikmesine bağlı olarak TDMA veya FDMA seçilebilir. Optimum kanal sayısı alıcı ve gönderici düğümlerin güç tüketim oranlarına bağlıdır.

Kısacası hangi MAC tekniğini kullanılırsa kullanılsın güç tüketimi öncelikli kriterdir. En çok kullanılan, göndericinin boştayken uyku moduna geçmesidir. Ancak burada şunu da unutmamak gereklidir ki veri iletimi küçük veri paketleri halinde yapıldığından düğümlerin yeniden çalışmaya başlaması önemli bir problemdir. İşte burada bir “eşik zaman değeri” adı verilen bir kavram işin içine girmektedir. Eğer tasarruf modunda geçen zaman, bu eşik değerin üstünde ise bu güç tasarrufu şekli uygun olacaktır.

Veri iletim katmanının diğer önemli bir fonksiyonu da hata kontrolüdür. Bilinen iki teknik, FEC ve ARQ, kullanılır. Ancak bunlardan ARQ tekrardan veri gönderimini gerektirdiğinden, enerji tüketimi açısından maliyetli olacaktır. WSN’lerde genelde iyi belirlenmiş FEC kullanımı tavsiye edilir. İyi belirlenmiş ifadesindeki kasıt; basit bir kodlama tekniğinin kullanılmasıdır ki bu da kod çözme işlemini daha basitleştirecektir [15,16].

2.1.3. Ağ Katmanı

Normalde klasik ad-hoc yönlendirme teknikleri WSN’ler için uygun değildir[1,8,17]. Bunun yerine, PA ifadesi, düğümün mevcut güç seviyesini, aise iletim esnasında harcanacak enerjiyi temsil etmek üzere, Şekil 2.2 ‘de gösterilen örnek yapıya göre aşağıdaki prensipler uygulanmaktadır.



Şekil 2.2. Örnek bir WSN [1]

*Maksimum PA Oranına Göre:*Yönlendirmede en önemli kriter PA’dır. Yönlendirme rotalarındaki toplam PA’lar ve sıçrama sayıları baz alınır. “Düğüm / PA” oranlarına bakılır.

*Minimum Enerjiye Göre:*Burada sink düğüme kadar olan harcanacak minimum enerji miktarı kriter olarak alınır.

Minimum Sıçramaya Göre: Sink düğüme en az sıçramada yapılacak rota seçilmektedir. Burada düğümler, güç seviyelerini de birbirlerine haber vermek durumundadırlar.

Bir diğerk önemli mesele ise yönlendirmenin data merkezli bir yapıya bağılı olması durumudur. Data merkezli yönlendirme “özellik” bilgisine ihtiyaç duyar. Burada kullanıcının yapacağı “özellik sorgusu” önemlidir [1]. Örneğin; ‘Sıcaklığın 35 derecenin üstünde olduğu bölgeler’ sorgusu, ‘A bölgesinden okunan sıcaklık’ sorgusuna göre daha genel bir sorgudur. Bu örnekte birincisi “Özellik Tabanlı Adlandırma”, ikincisi ise “Konum Tabanlı Adlandırma”dır.

Data merkezli yönlendirmede, ‘Veri Toplama’, oluşabilecek sorunları çözmede kullanılan bir tekniktir [2]. Veri toplama genelde “Data Fusion” olarak bilinir.

Ağ katmanı ağlar arası iletişimi de mümkün kılabilmelidir. Burada sink düğümler, ağları birbirine bağlayan ağ geçitleri olarak kullanılabilir.

Ağ katmanında kullanılan birçok yönlendirme tekniğı bulunmaktadır. Aşağıda bu teknikler kısaca açıklanmıştır [1]:

- **Taşkın** :Eski bir tekniktir. Veri paketini alan her bir düğümün, aldığı paketi her yöne yayımlaması prensibine dayanır.
- **Dedikodu (Gossiping)** :Bir çeşit taşkın tekniğıdir, burada gelen paket her yöne iletilmez, rastgele seçilmiş bir komşuya gönderilir.
- **Dönüş (Spin)** : Data merkezli bir yönlendirme değildir. Burada bilgi ufak bir sorgu ile belirlenen ilgili komşu sensöre yönlendirilir.
- **Ardışık Atama Yönlendirmesi, SAR** : Burada bir sıçrama uzaklığına sahip düğümlerin oluşturduğu ağ yapısı kullanılır.
- **Filtreleme (Leach)** : Kümeleme tabanlı bir protokoldür. Rastgele seçilen küme başı sensörü etrafına toplanan düğümlerden oluşur. Bu protokolün ana amacı güç tüketiminde verimi sağlamaktır.
- **Doğrudan Yayma** : Bu teknikte sink düğüm görev tanımlı bir istek yayınlar. Bu göreve en yatkın sensör, bu bilgiyi aldığı anda gerekli veriyi yerel olarak belirlediğı komşuya göndermek suretiyle sink düğüme sonuç verisini ulaştırır.

2.1.4. Taşıma Katmanı

Taşıma katmanına, özellikle sisteme internet veya harici bir ağdan erişime gerek olduğunda ihtiyaç duyulur. Kullanıcı ile sistem arasında, internet veya uydu üzerinden bağlantılarda UDP ya da TCP kullanılır. Ancak sink ve sensör düğümler arası bir iletişim olacaksa UDP hafıza tüketimi açısından tercih edilir [14].

2.1.5. Uygulama Katmanı

En üst katman olan uygulama katmanında kullanılan bazı protokoller aşağıda kısaca açıklanmıştır [1]:

- *Sensor Management Protocol (SMP)*: Uygulama katmanı programları kullanıcı ile sistem arasında şeffaflık sağlar. SMP, özellik tabanlı veya konum tabanlı adresleme kullanır.

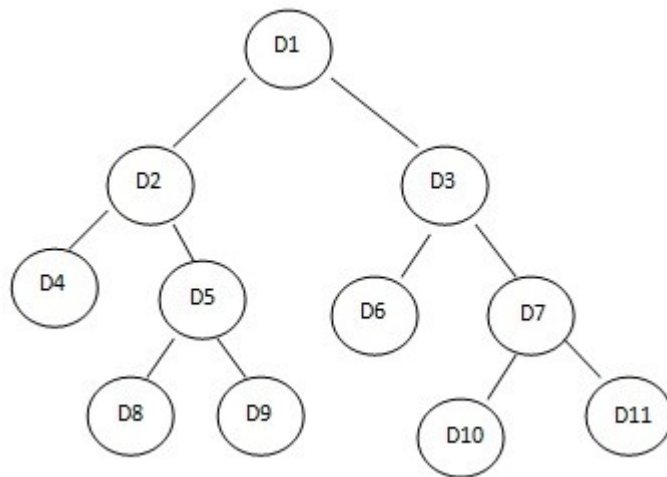
- *Görev Atama ve Veri İlanı Protokolü*: İstenecek ilgili verinin ağa yılmasını sağlar.

- *Sensor Query and Data Dissemination Protocol (SQDDP)*: SQDDP kullanıcıya sorgulama yapma imkânı sunar. Sorgulama, konum veya özellik tabanlı olur. Bu sorgulama için SCTL (Sensor Query and Tasking Language) dili önerilir [1].

2.2. WSN Topolojileri

Ağaç Topoloji

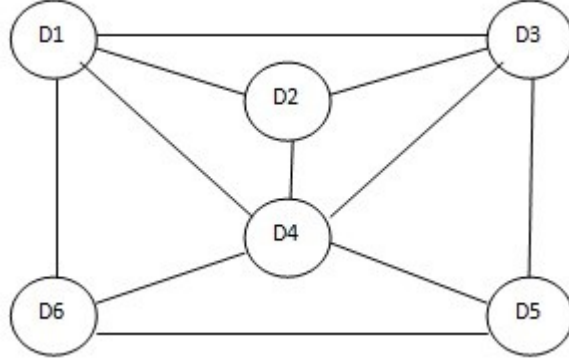
Düğüm, Şekil 2.3'deki gibi kapalı bir döngü oluşturmayacak bir şekilde ağaç yapısında yerleştirilebilir. Bir ağaçta döngü bulunması bilgi yayımını bozar. Kablosuz sensör ağlarında, ağaç yapısı kullanılacaksa, plan aşamasında ağaç yapısını oluşturacak algoritma iyi seçilmelidir. Bir yayılım ağacı kurmada en iyi bilinen yöntem “*deep-first-search (DFS)*” dir. Algoritmanın amacı, kök düğümden itibaren, diğer tüm düğümlere dallanmalar vasıtasıyla mümkün olduğu kadar ulaşmaktır. Kullanılabilecek başka bir algoritma ise “*breadth-first-search (BFS)*” dir [3,7,14].



Şekil 2.3. Örnek bir ağaç topoloji

Örgü(Mesh) Topoloji

Bu topolojide, kablosuz sensör ağında bulunan tüm uçlar birbirine bağlıdır. Mesh topoloji, karmaşık topoloji olarak da bilinir. Bu topolojide bütün uçlar arasında bağlantı olduğundan bir dağıtıcının kullanılmasına gerek yoktur. Bununla beraber yönlendirme bu yapıda önem arz etmektedir. Çünkü bir verinin hangi yolları izleyeceği daha önceden belirlenmiş olmalıdır[7].

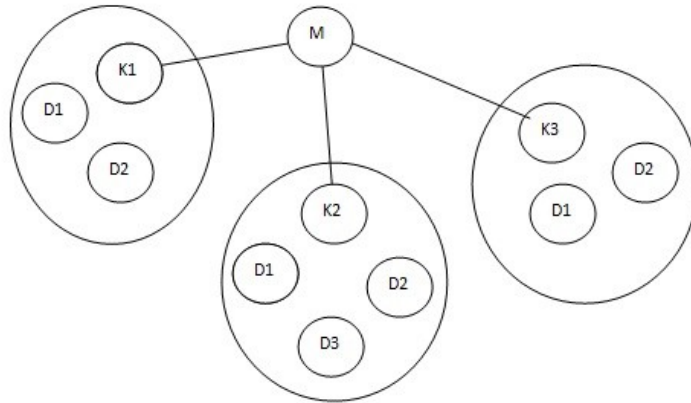


Şekil 2.4.Örgü Topoloji

Şekil 2.4'te görüldüğü gibi, ağaç yapısından farklı olarak mesh topoloji, kapalı döngüler içerebilir. Bu nedenle bu yapı dağıtık sistemlerde kullanım için avantaj oluşturur. Tabi böyle bir dağıtık yapıda veri yayını önceden hesaplanmış olmalıdır.

Sıradüzensel Topoloji

Bu yapıda, mantık olarak hiyerarşik düzen model alınır ve çalışma yapısı olarak kümesel yapı veya askeri yapı işleyişi uygulanır [2,6].



Şekil 2.5. Sıradüzensel Topoloji

Şekil 2.5'te görüldüğü üzere, K1, K2, K3 düğümleri, küme başı veya komutan düğüm olarak adlandırılır. Her bir ağ bölümü bu anadüğümler (parent nodes) tarafından yönetilir. Bu ebeveyn düğümler ana sistemden aldığı komutları ağa, ağdan aldıkları verileri de merkez düğüme (M) iletirler. Her bir ana düğüm, yavru veya asker olarak nitelendirilen (D1, D2 gibi) bir grup düğümden sorumludur [2].

2.3. Kablosuz Sensör Düğümleri

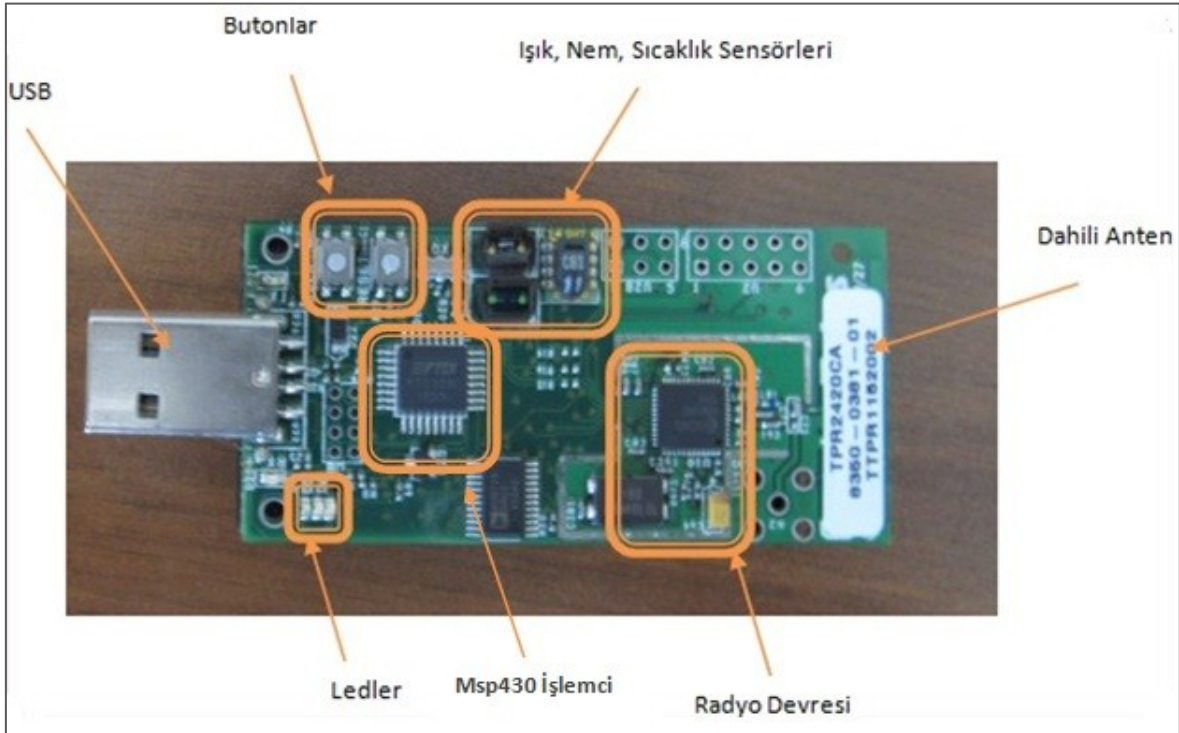
Günümüzde mica2, micaZ, TelosA, TelosB, eMote, IMote2 gibi birçok düğüm modeli bulunmaktadır [11]. Kablosuz sensör ağlarında kullanılan düğüm teknolojileri üzerlerinde barındıkları işlemci, radyo devresi, anten yapısı, ADC yapısı, hafıza miktarı gibi birçok etkene bağlıdır. Enerji farkındalığına sahip düğümlerin kullanılması sistem açısından hayati öneme sahiptir. Düğümlerin diğer ayırt edici özelliklerinden biride anten yapısıdır. Ortalama mesafe 30 metre ile 70 metre civarında olmasına rağmen Mica2 gibi sensör düğümleri 300 metreye kadar ulaşabilir ki bu durum beraberinde fazladan güç tüketimini getirir. Düğümlerde kullanılan mikro denetleyici ve diğer yardımcı birimler (ADC, SPI, UART vb.) açısından bakıldığında ise, kullanılan saat frekansı 4-8 Mhz arasında değişmektedir. Mikro denetleyicilerin komut başına harcadıkları enerji önemli bir değerdir. Sık kullanılan mikrodenetleyici ve radyo devrelerini kısaca özetlemek gerekirse [4,13-16];

- The AT90LS8535 mikrodenetleyici: 15 mW (3.75 nJ/komut) güç tüketir. Uyanma zamanı 1 msn dir.
- ATmega163 mikrodenetleyici: 8 bitlik mikro denetleyici komut başına 1.875 nJ enerji tüketir. Uyanma zamanı 36 µsn gibi oldukça kısa bir zamandır.
- ATmega103L mikrodenetleyici: 4 Mhz'te çalışır. 128 KB program hafızası ve 4 KB RAM vardır.
- ATmega128 mikrodenetleyici: 8 bitlik yaygın kullanılan bir mikro denetleyicidir. 128 KB programlanabilir flash, 8 - 33 mW güç tüketimi, 180 µsn uyanma zamanı vardır.
- The TIMSP430 mikrodenetleyici: Ultra düşük güç tüketimi vardır(3 mW). Ancak program hafızası fazla yüksek değildir.
- Intel PXA271 mikrodenetleyici: Imote2 düğümler tarafından kullanılır. 400 Mhz gibi yüksek frekansa sahiptir. Sunspot düğümünün kullandığı ARM920T ise 180 Mhz'te çalışabilmektedir. ZigBee protokolünü destekler. Genelde görüntü işleme, karbondioksit emilimi gibi uygulamalarda kullanılır. İletişim teknolojisi açısından

bakıldığında yine karşımıza güç tüketimi çıkmaktadır. İletim esnasında harcanan güç, düğümün ömrü açısından oldukça önemlidir.

- CC1000 Radyo Devresi: Düşük güçlü CMOS RF Transceiver'dır.FSK kullanarak 76.8 Kbps'e kadar veri hızı sağlayabilir. Merkezi kontrol birimi (CCU) ile senkronizasyonunu sağlayan dahili bir senkronizör vardır. Rx modda 29 mW ve Tx modunda 42 mW güç tüketimi vardır. ZigBee desteği sunmaz.
- TR1000 Radyo Devresi: Kolay programlama özelliğine sahiptir. ZigBee desteği yoktur.
- CC2420 Radyo Devresi: 2.4 GHz'te çalışır ve 802.15.4/ZigBee desteği sunar. Küçük voltaj gerilimi ile güç tüketimi daha azdır. Geniş bir donanım desteği sunar. MAC şifreleme desteği vardır. Data yetki denetimi, temiz kanal atama, paket zamanlama bilgisi hizmetlerini sunar.
- CC2431 Radyo Devresi: ZigBee desteği sunar. Ayrıca konum motoru denilen özelliği sayesinde düğümün göreceli durumunu tahmin edebilir.

Bu çalışmada kullanılan MEMSIC firmasına ait TelosB düğümünün resmi Şekil 2.6'da görülmektedir [43].



Şekil 2.6. TelosB düğüm yapısı

Kullanılan düğümün özellikleri aşağıdadır.

- IEEE 802.15.4/ZigBee uyumlu RF alıcı,
- 2.3 – 2.4835 GHz, global olarak uyumlu ISM bandı
- 250 kbps data oranı
- Tümlşik onboard anten
- 10 kB RAM’li 8 MHz TI MSP430 mikrokontrolör
- Düşük akım tüketimi
- Data loglama için 1 MB harici flash
- USB aracılığıyla programlama ve data toplama
- Tümlşik ışıık, sıcaklık ve nem sensörünü içeren (TPR2420) opsiyonel sensör

takımı

- TinyOS 1.1.10 veya üst sürümelerini çalıştırma

2.4. Kablosuz Sensör Düğümlerinde Kullanılan Gömülü İşletim Sistemleri

Kablosuz sensör ağlarında kullanılan işletim sistemleri genel olarak iki kısımda incelenir. Olay-tetiklemeli ve çoklu-iş parçacıklı [20]. Olay-tetiklemeli sistemlerde, işletim sisteminin gerçekleştireceği her hareket bir olayla (bir zamanlayıcı, yeni bir sensör okumasını gösteren bir kesme veya gelen bir radyo paketi, vb.) tetiklenmelidir [10-11,20]. Bu tür işletim sistemlerine örnek TinyOS’tur. Çoklu-iş parçacıklı işletim sistemlerinde ise işletim istemi; farklı görevler arasındaki yürütme zamanını çoğullar [10,20]. Bir iş parçacığından diğerine geçerken, o anki içerik kaydedilmeli ve yerine yeni içerik alınmalıdır. Sensör düğümleri için böyle işletim sistemlerine örnek MANTIS’tir. WSN’lerde kullanılan gömülü işletim sistemleri özet olarak aşağıda verilmiştir.

MANTIS

Mantis işletim sistemi geleneksel olarak kullanılan öncelikli bir çoklu iş parçacıklı işletim sistemidir. Mantis bütün işletim sistemlerini ve program hafızasının parçalarını, EEPROM üzerine indirilecek bir program imajıyla tekrar programlamaya imkan veren bir sistemdir. Çoklu iş parçacıklı semantiklerinden dolayı, her Mantis programı sistem yığını tarafından ayrılmış bir yığıt alanına sahip olmalıdır. MANTIS uygulamalarında paket-işleme görevi, algılama görevinden daha yüksek önceliğe sahiptir [44].

CONTIKI

Bütün sistemin tam bir ikili (binary) imajını gerektiren çoğu işletim sistemleri her bir cihaza yüklenmiştir. İkili yapı, işletim sistemini, sistem kütüphanelerini ve sistemin üstünde çalışan tüm uygulamaları içerir. Çoğu durumlarda, özgün bir uygulama bütün sistemin ikilisinden çok küçüktür ve bundan dolayı az enerjiye ihtiyaç duyar. Contiki, eşzamanlı iş parçacıklarının sayısını iki olarak kısıtlamaz [45].

μ C/OS-II

μ C/OS-II, gerçek-zamanlı, öncelikli bir çoklu görev gömülü işletim sistemi çekirdeğidir. Portatif olması, ölçeklenebilir olması ve kullanım kolaylığı bakımından popülerdir. μ C/OS-II gömülü ağ uygulamalarında, çoklu görev, senkronizasyon, zamanlayıcı yönetimi, hafıza yönetimi gibi anahtar özellikleri içerir. μ C/OS-II'nin ağ katmanı kablosuz sensör ağlarında çeşitli yönlendirme algoritmalarını destekler [20].

LIMOS

LIMOS, pratik uygulama çevresine bağlı olarak, sistemin verimini geliştirmek ve kaynak gereksinimlerini azaltmak için olay tetiklemeli veya çoklu iş parçacıklı modda çalışabilen bir hibrit işletim sistemidir. LIMOS akıllı, kaynak-farkındalıklı, düşük-enerjili ve dağıtık gerçek-zamanlı bir mikro-çekirdektir [46].

LiteOS

LiteOS, kablosuz sensör ağları için Unix-benzeri soyutlamaları sağlayan bir çoklu iş parçacıklı işletim sistemidir [47]. Kolay-kullanımlı bir arayüzü hedeflediğinden, LiteOS birkaç orjinal özellik sunar:

- UNIX-benzeri komutları kullanarak kullanıcı etkileşimi için bir kablosuz dış kabuk arayüzü ve hiyerarşik dosya sistemi
- Dinamik yükleme için çekirdek desteği ve çoklu işparçacıklı uygulamaların doğal yürütülmesi
- Çevrimiçi hata ayıklama, dinamik hafıza ve iletişim yığıtlarını destekleyen dosya sistemi.

Nano-Qplus

Nano-Qplus yeni bir çoklu iş parçacıklı, basit ve düşük-güçlü sensör ağı işletim sistemidir. Nano-Qplus, görev kavramını yürütülecek kod parçası için kullanır. [48]. Genel olarak daha az karmaşık kablosuz sensör ağı yapılarında kullanılır.

Nano-RK

Nano-RK, ağı destekli küçük bir gömülü gerçek-zamanlı bir işletim sistemidir [49]. Yapı itibari ile güç farkındalığına önem veren bir sistemdir. Bu da daha az karmaşık kablosuz sensör ağı sistemlerinde tercih edilir.

PAVENET OS

PAVENET OS hibrit çoklu iş parçacıklı bir yapı sağlar: bunlar öncelikli çoklu iş parçacığı ve ortak öoklu iş parçacığıdır . Her iki çoklu iş parçacığı kablosuz sensör ağları üzerinde; gerçek-zamanlı görevler ve en başarılı görevler için optimize edilmiştir. PAVENET OS, TinyOS tarafından gerçekleştirilemeyen zor gerçek-zamanlı görevleri verimli biçimde gerçekleştirebilir. Zor gerçek zamanlı özelliğini fark etmek için, PAVENET OS bir iş parçacığı modeli ve önceliği sağlayarak dizayn edilmiştir. PAVENET OS portatiflikten fedakarlık etmiştir ve büyük bir problemdir [50].

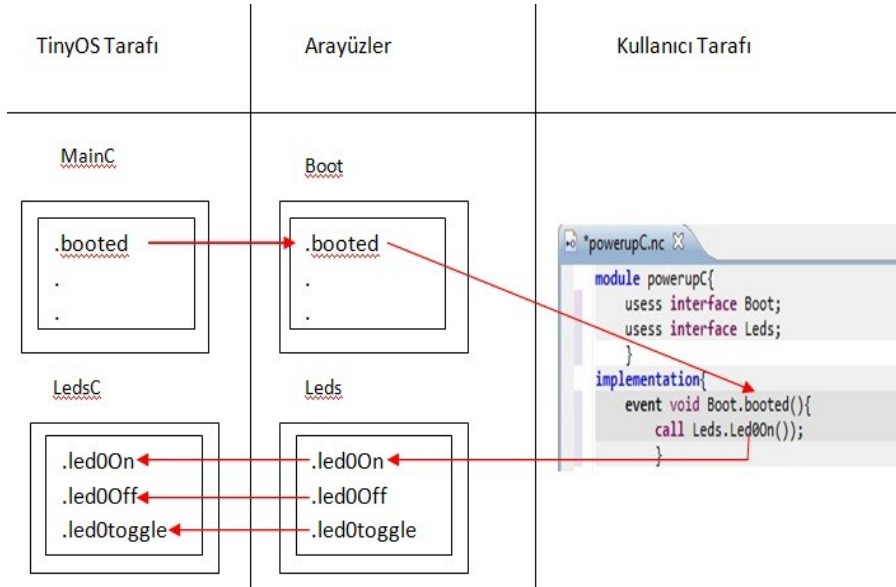
TinyOS

Bu tez çalışmasında da kullanılan TinyOS gömülü işletim sistemi, California Üniversitesinde WSN'ler için geliştirilmiş açık kaynak kod'lu bir işletim sistemidir [11]. TinyOS, düğümün sahip olduğu sınırlı kaynakları dikkate alarak düğümün programlanmasını sağlar. TinyOS 'un uygulama geliştirme için kullandığı dil C tabanlı bir dil olan nesC'dir. Sistemdeki düğümler bu dil ile programlanırlar. Verilerin değerlendirmesini ve gelen verilerin işlemlerini yapacak olan ana sistem ile kablosuz sensör ağı arasındaki bağlantı ise Java, Visual Basic gibi daha yüksek seviyeli bir dil ile yapılabilir [7,10,11].

TinyOS, çevresindeki olayların algılanmasını ve yönetilmesini sağlayacak programlama yapısıyla gömülü sistemler için ideal bir yapı sunar. Kullanılan nesC dili, olaylarla tetiklenen bir mekanizmaya sahiptir ve bileşen tabanlı olduğundan WSN 'ler için oldukça uygundur. NesC dilindeki bileşenler, nesneye dayalı programlamadaki nesnelerle benzetilebilir. Durum bilgisine ve bu durum için gerekli fonksiyonları ihtiva eden

bileşenler kendi aralarında birbirleri ile *arayüzler* aracılığıyla etkileşim kurmaktadır. TinyOS'ta süreçler arasında geçiş mekanizması bulunmaz.

Modül, işin gerçekleştirmesini sağlayan bileşen iken *yapılandırıcı* ise diğer bileşenler arasında bir bağ görevi gören bileşendir. [8,11]. Söz konusu bileşenler arasındaki genel mantık Şekil 2.7'de gösterilmiştir.



Şekil 2.7. Bileşenler arası genel etkileşim

TinyOS'ta bileşenler birbirleri ile etkileşirken tek yönlü bir yapı söz konusu değildir. Bir bileşen herhangi bir bileşenin iç komutlarını çağırabilir ve aynı şekilde komutları çağırılan o bileşen aynı anda karşı bileşendeki olayları tetikleyebilir. [8,10,11]

```

module PowerupC {
  uses interface Boot ;
  uses interface Leds ;
}
implementation {
  event void Boot.booted () {
    call Leds.led0On();
  }
}

```

Şekil 2.8. Modül genel yapısı [8]

Şekil 2.8'de TinyOS'ta kullanılan nesC dilinde yazılan bir modülün genel yapısı ve Şekil 2.9'da ise bir ara yüz yapısı gösterilmektedir.

```

interface Boot {
    event void booted ();
}

interface Leds {
    command void ledOn();
    command void ledOff();
    command void ledToggle();
    ...
}

```

Şekil 2.9.Ara yüz yapısı [8]

Şekil 2.9’da görüleceği üzere, düğüm açıldığı zaman, tetiklemeyi sağlayan bir *booted* olayı vardır. Bu olay *Boot* arayüzünde tanımlıdır. Şekil 2.8’de görüldüğü üzere, *PowerupC* modülü, bu arayüzü kullanmakta ve buradaki *booted* olayından tetiklenmektedir. Dikkat edileceği üzere *PowerupC* modülü, *Leds* adında ikinci bir arayüz daha kullanmaktadır ki bu arayüzde ledleri açıp kapatacak komutlar mevcuttur. *PowerupC* modülü tetiklendiği zaman *Leds* ara yüzünde bulunan bu komutlardan birini çağırılmaktadır.

Buradaki “event” komutu gerekli sinyalleri yaparak gerekli etkileşimleri sağlamıştır. Modüller, görüldüğü üzere, birden çok arayüz ile iletişime geçebilir. Bu örnekte cihazın açılması esnasında led yakma işlemi gerçekleştirilmiştir. Modülün “implementation” kısmı, bu arayüzün ledleri yakacak olan komutları çağırılmaktadır. Komutlar “command” anahtar kelimesi ile belirtilirler ve çağırılmaları “call” anahtar kelimesi ile yapılmaktadır. Dikkat edilmesi gereken nokta, modülün hangi ara yüzler ile etkileşeceğinin ilk başta belirtilmesinin gerekli olduğudur.

Belirtilen arayüzlerin kullanılabilmesi için bir yapılandırıcıya ihtiyaç vardır. Bu yapılandırıcı “*configuration*” anahtar kelimesi ile tanımlanan yapılandırma bileşenlerince gerçekleştirilirler. Bu dosya, modüllerin birbirleri ile etkileşime geçmesini sağlayacak yapıyı tanımlar.

```

configuration LedsC {
  provides interface Leds;
}
implementation {
  ...
}

configuration MainC {
  provides interface Boot;
  ...
}
implementation {
  ...
}

```

Şekil 2.10. Yapılandırıcı örneği[8]

Şekil 2.10’da kullanılan yapılandırıcının kodları ve genel yapısı görülmektedir. *Configuration* anahtar kelimesi ile başlar. “*Provides*” anahtar kelimesi, hangi yapılandırıcının hangi arayüzü kullanacağını tanımlar.

“LedsC” yapılandırıcısı, “Leds” arayüzüne bağlanırken, “MainC” yapılandırıcısı ise “Boot” arayüzüne bağlanmıştır. Şekil 2.11, bu bağlantının nasıl gerçekleştiğini göstermektedir. Görüleceği üzere MainC yapılandırıcısı içinde daha önce belirtilmiş olan Boot arayüzü “ -> ” operatörü ile PowerupC modülündeki kullanıldığı belirtilmiş olan Boot kısmına atanmıştır. Aynı şekilde PowerupC’de kullanılacağı belirtilen Leds arayüzü ise “->” operatörü ile LedsC yapılandırıcısı içinde belirtilen “Leds” arayüzüne atanmıştır. Kısaca Boot arayüzündeki tetiklenmeden haberdar olan MainC yapılandırıcısı, yapılan atama neticesinde PowerupC modülünü tetikleyecektir. Bu tetikleme sonucunda PowerupC’nin, Leds arayüzündeki komutlara ulaşmasını sağlayacak atama işlemi de yine PowerAppC yapılandırıcısında sağlanmıştır.

```

configuration PowerupAppC {

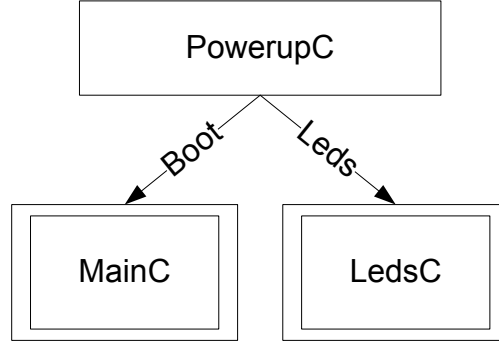
  implementation {
    components MainC , LedsC , PowerupC ;

    MainC.Boot -> PowerupC.Boot ;
    PowerupC.Leds -> LedsC.Leds ;
  }
}

```

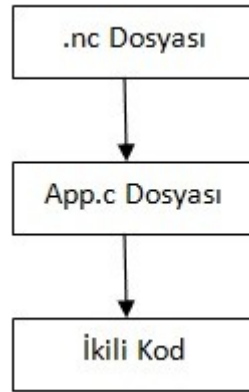
Şekil 2.11. Yapılandırıcı genel tanımlama

Bu atamaların sonucunda Şekil 2.12’de blok olarak gösterilen bağlama işlemleri gerçekleştirilmiş olacaktır [8,10].



Şekil 2.12. Yapılandırıcıların kullanılması

Bu işlemler sonucunda oluşturulan “PowerupC.nc” ve “PowerupAppC.nc” uygulama dosyaları nesC derleyicisi tarafından derlendiğinde “app.c” isimli C dili dosyası oluşmaktadır. Şekil 2.13’de görüleceği üzere derlenmiş olan bu dosya, bir C derleyicisi tarafından bir kez daha derlenerek cihaza yüklenecek ikili kod dosyası elde edilecektir[8,10,11].



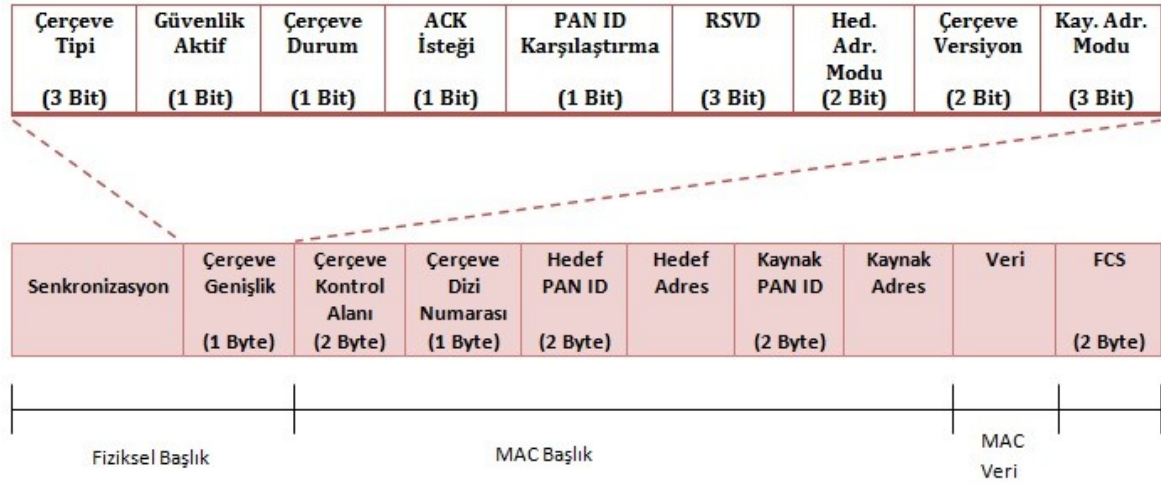
Şekil 2.13. Derleme işlemleri

TinyOS’un bileşen kütüphaneleri; ağ protokollerini, dağıtık servisleri, algılayıcı sürücülerini ve veri toplama araçlarını içermektedir. Veri yayını için ActiveMessageC, AMSenderC and AMReceiverC bileşenleri kullanılabilir[11]. Ağ yapısındaki düzenin kararlı kalması için hem sink hem de diğer düğümlerin, durum bildirim mesajlarını, belirli periyotlarla yayınlamaları gerekir.

Bu konunun daha iyi anlaşılması için IEEE 802.15.4 standardı hakkında bilgi vermek faydalı olacaktır. IEEE 802.15.4 standardı, küçük ölçekli kablosuz kişisel alan ağları için fiziksel ve veri bağı katmanlarını tanımlayan bir standarttır [11,17]. Kablosuz sensör ağları

için temel yapıyı oluşturur. ZigBee, WirelessHART, 6LowWPAN spesifikasyonları IEEE 802.15.4 standardını baz alırlar.

TinyOS platformu, genel olarak IEEE 802.15.4 veri başı ve fiziksel paket formatını kullanır. 802.15.4 birkaç farklı kaynak ve hedef adresleme modlarını destekler. Değişken bir paket başlık yapısına sahiptir. Çerçeve yapısında bulunan 802.15.4 başlık yapısı Şekil 14’de görülmektedir[17].



Şekil 2.14. IEEE 802.15.4 Çerçeve Yapısı

Şekil 2.14’de görüldüğü üzere fiziksel katman başlığı, senkronizasyon ve çerçeve genişliğini belirler. MAC katman başlığında FCF alanı, çerçeve kontrol görevini yerine getirir. Çerçeve tipi, güvenlik, paket alındı talebinin istenip istenmeyeceği, hedef ve kaynak adres modlarının belirlenmesi bu kısımda tanımlanır. DSN alanı, çerçeve sıra numarasını belirlerlerken Hedef PAN ID ve Kaynak PAN ID ağların ID bilgilerini tutar. Hedef Adres ve Kaynak Adres ise hedef ve kaynak düğümlerinin adresleri hakkında bilgileri ihtiva eder. FCS alanı çerçeve yapısının hata kontrolü için kullanılır [17].

TinyOS’ta gönderilen ve alınan tüm mesajlar “ActiveMessages” olarak gerçekleştirilir. ActiveMessages tanımlamaları “tos/types/AM.h” dosyasında tanımlıdır. Her mesaj başlığı, üzerinde “HANDLER ID” bulunur. Bir mesaj alındığında HANDLER ID o işle görevli olan ilgili EVENT’i tetikler ve gerekli icra işlemi gerçekleştirilir. TinyOS’un kullandığı mesaj formatında, hedef adres, varsa grup ID, mesaj tipi, mesaj boyutu ve veri gibi tanımları içermektedir [11,17].

Bir düğüm elde ettiği veriyi göndereceği zaman, kendi programlama yapısına ve ağ topolojisine uygun olarak gerekli hedef adresi ve bulunduğu ağı tanımlayan ağ ID’sini göndereceği mesajda belirtir. Eğer bir küme başı düğümü kullanılıyorsa, bu düğüm ağın

tasarımı aşamasında belirlendiğinden dolayı, kendi kümesindeki düğümlerden gelen bilgileri toplayıp gerekli işlemleri yapabilecek şekilde programlanmıştır. Elde ettiği verileri, bünyesinde toplayıp, kendi yönlendirme tablosuna bakarak, tanımlı mesaj çerçeve yapısına uygun olarak mesajı hedef düğüme geri gönderir. Bu şekilde tüm ağdan gelen veriler sink düğüme ulaştıktan sonra ilgili bilgisayar yazılımına aktarılır[6-8,11].

TinyOS iki çeşit 802.15.4 çerçeve yapısına sahiptir. İlk yapı T-Frame (Şekil 2.15) olarak bilinen paket yapısını kullanan TinyOS ağları, kanallarını diğer kablosuz ağ mimarileri ile paylaşmazlar. Bu paket yapısında, TinyOS'un paketin her bitinin kullanabildiğini ve kullanılan paket yapısını yayınlamaya gerek olmadığını varsayar [17].

802.15.4 Başlık (10 Byte)	Tip (1 Byte)	Veri (115 Byte)(En Fazla)	CRC (2 Byte)
------------------------------	-----------------	------------------------------	-----------------

Şekil 2.15. T-Frame Yapısı

Buradaki “Tip” kısmı 8 bittir ve datanın ihtiva ettiği aktif mesaj tipini gösterir. CRC kısmı paket yapısındaki hata denetimi için kullanılır.

İkinci tip frame yapısı ise I-Frame olarak bilinir. Bu paket yapısı, IPv6 altyapısını kullanan ve sensörlerle iletişime geçme kabiliyeti olan “*6lowpan Ağları (IPv6-based low-power wireless PAN)*” ile paylaşımli kanal kullanımına olanak tanır. Beraber çalışabilirliği sağlamak için paket yapısında ilave olarak “*6lowpan*” alanı bulur. Bu alan 0-63 bit arasındadır (Şekil 2.16).

802.15.4 Başlık (10 Byte)	6lowpan (0-63 Byte)	Tip (1 Byte)	Veri (115 Byte)(En Fazla)	CRC (2 Byte)
------------------------------	------------------------	-----------------	------------------------------	-----------------

Şekil 2.16. I-Frame Yapısı

2.5. Simülâtörler

Bir kablosuz sensör ağının simülasyonu için NS-2, OMNET++, JSIM, OPNET, GloMoSim /QualNet, MATLAB, JiST/SWAN, SSFNet vb. gibi genel amaçlı simülâtörler ve SensorSim, Castalia, VibTOS v.b özel amaçlı simülâtörler mevcuttur .Ayrıca WSN’ler için TOSSIM, ATEMU, EmStar v.b emülâtörler de kullanılmaktadır. Bunlardan TOSSIM, düğümleri programlamak için yazılan TinyOS kodlarında hata ayıklama, test etme, kontrollü ve tekrarlanabilir bir ortamda algoritmaların önceden analiz edilmesini sağlar. Bir protokolü veya sistemi TOSSIM’de simüle edebilmek için kesinlikle onun TinyOS uygulamasının nesC dilinde yazılması gerekir. Bu işlem soyutlanmış bir simülasyondan

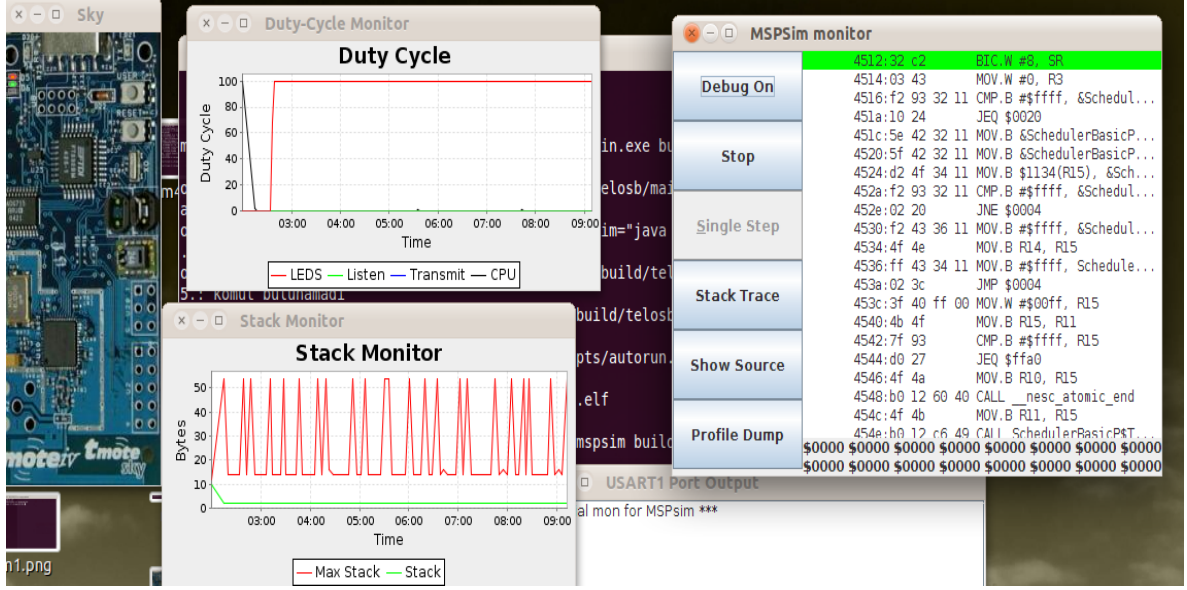
çok daha zordur. Ancak simülasyon için nesC ile yazılmış program gerçek düğümlere de yüklenebildiğinden, aynı zamanda gerçek zamanlı düğüm programlama işlemi de başarılmış olmaktadır [10,11]. TOSSIM, düğüm kodlarını simüle ettiği gibi düğümleri birbirleriyle etkileştirerek, düğümlerin oluşturacağı ağın da modellenip simüle edilmesini sağlayabilir.

WSN’lerde uygulama geliştirme süreci; düğümlerin ve işletim sisteminin seçilmesi, kümeleri ve küme başlarını oluşturacak ve bilgisayarla haberleşecek düğümlerin istenen işleri başaracak şekilde programlanması, bunların simüle edilmesi ve gerçek sistemin test edilmesi gibi bilgiye dayalı birçok karmaşık aşamadan oluşur.

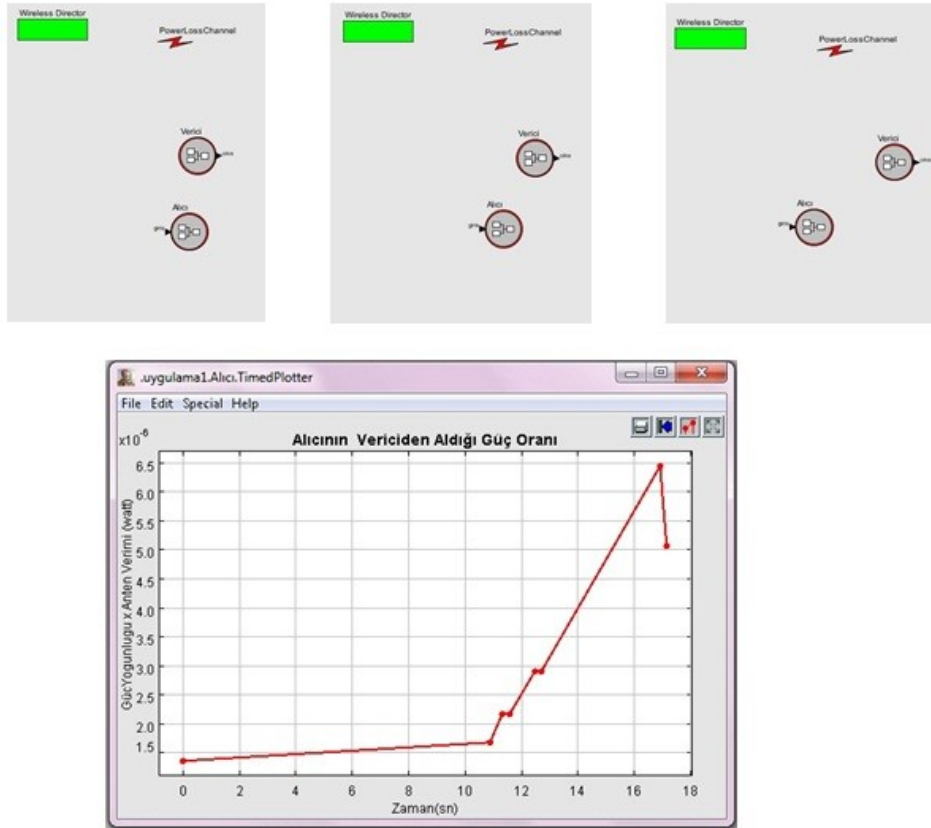
WSN’lerle uygulama geliştirme sürecinin anlaşılması ve öğrenilmesini kolaylaştırmak için; kablosuz sensör düğümlerini ve kablosuz sensör ağını sistematik olarak simüle edip sonuçlarının gözlenmesini sağlayabilecek, kompakt ve görselliği olan interaktif bir sanal ortamın, WSN’lerle ilgilenenler için önemli bir platform oluşturacağı açıktır. Bunun için yapılan çalışmaların bazıları aşağıdadır.

MATLAB’ın TrueTime Toolbox’ının kullanıldığı bir çalışmada, batarya ömrü, iletişim ortamı gibi olguların simülasyona katıldığı, sadece düğümlerin davranışsal özelliklerini baz alan 3 boyutlu bir simülasyon platformu önerilmiştir [24]. Başka bir çalışmada bina içi kablosuz sensör ağlarının simülasyonlarında, açık kaynak kodlu Wonderlan araçlarını kullanan 3 boyutlu bir WSN simülasyonu önerilerek iç mekan uygulamalarında karşılaşılan zorlukların daha iyi analiz edilebildiği savunulmuştur [21]. IDEA1 isimli, sistem seviyesinde bir WSN simülatörünün geliştirilmesinin yapıp, Paket dağıtım oranı, transmisyon gecikmesi ve enerji tüketimi gibi performans değerlendirme kriterlerin simüle edilip değerlendirilmesine olanak sağlanan bir çalışmada ise ayrıca düğüm topolojinin gösterilebildiği bir GUI’de oluşturulmuştur [22]. Öğrencilerin WSN’ler hakkında bilgilendirilmesi için donanımsal bir deney setinin gerçekleştirilmesi çalışması ise bu konuda yapılan çalışmalardan başka bir tanesidir [23].

Kablosuz sensör ağlarında simülasyonun önemi kuşkusuz göz ardı edilemez. Gerçekleştirilecek uygulamanın düğüm kodlarının emüle edilmesi veya mantıksal olarak sistemin analizinin yapılması hem zaman hem de maliyet açısından oldukça kazanım sağlayacaktır. Örneğin Şekil 2.17’de MspSim [51] emülatöründede gerçekleştirilen “Blink” uygulamasının simülasyonu görülürken, Şekil 2.18’de VisualSense [18] simülatöründe gerçekleştirilen alıcı düğümde elde edilen sinyal gücünün simülasyonu görülmektedir.



Şekil 2.17. MspSim ile Blink simülasyonu



Şekil 2.18. VisualSense ile alınan sinyal gücü simülasyonu

Bu tez çalışmasında kullanılan TelosB düğüm kodlarının emülasyonu ve simülasyonu TinyOS'un kendi emülatörü olan TOSSIM kullanılarak yapılacaktır. Bunun için TOSSİM'de uygulama geliştirme adımları aşağıdaki bölümde açıklanmıştır.

2.5.1.TOSSIM Emülatörü İle Simülasyon Geliştirme Adımları

TOSSIM, TinyOS'un kendi emülatörü olup kullanılacak olan TOSSIM ile simülasyon gerçekleştirilirken Python dilinden faydalanılır. Her ne kadar C diline de destek verse de Python dili daha esnek bir yapı sunmaktadır. TOSSIM simülatör programı micaZ düğümlerine göre simülasyon yaptığından simülasyon için yapılan derleme bu düğüm modeline göre yapılmalıdır. TelosB düğümler “mica” düğümlerle benzer donanımsal yapıyı kullandıklarından gerçekleştirilecek simülasyon TelosB düğümler içinde uyum sağlayacaktır [11]. Simülasyon esnasında gürültü modeli, harici olarak bir dosyadan okunup simülasyona eklenebilir. İstenilen düğümler birbirleri ile etkileşime geçirilebilir.

Bir simülasyona başlamadan önce bir başlık dosyasının oluşturulması sonraki adımlar açısından kolaylık sağlayacaktır. Örnek olması açısından bir başlık dosyası yapısı Şekil 2.19'da verilmiştir.

```
#ifndef genel_veriler_h
#define genel_veriler_h

typedef nx_struct mesaj{ // Burada mesaj içinde gönderilecek olan yapı
nx_uint8_t g_dugum;|
nx_uint8_t X;
nx_uint8_t Y;
}mesaj;

enum{ // Bu enum radyo mesajının türünü belirliyor
AM_tipi=6,
dg_sayisi=4, // Kaç çapa düğümden veri alınacak
};

#endif;
```

Şekil 2.19. Örnek bir başlık dosyası

Daha sonra simülasyonda kullanılacak olan düğümlere ait işletim kodları nesC dilinde tek bir dosyaya yazılır. Düğümlere ait kod ayrımları daha önceden belirlenmiş düğüm ID'leri ile birbirinden ayrılabilir. Şekil 2.20, bir düğüm dosyasının genel şeklini temsil etmektedir.

```

#include "Timer.h"
#include "genel_veriler.h"
module dugum @safe(){
    uses{
        interface Leds;
        .
        .
        interface Boot;
    }
    implementation{
        uint8_t i_yer=7;
        .
        .
        message_t paketimiz;

        event void Boot.booted(){

            if(TOS_NODE_ID==5){
                .
                .
                if(TOS_NODE_ID==7){|
                    .
                }

        event void kontrolcu.startDone(error_t hata){
            if (hata==SUCCESS){
                .
                .
                if (TOS_NODE_ID==5){

                    rfr->g_dugum=TOS_NODE_ID;
                    rfr->yonumuz=yon;
                    .
                    .
                    rfr->buradax=burada;
                    .
                    .

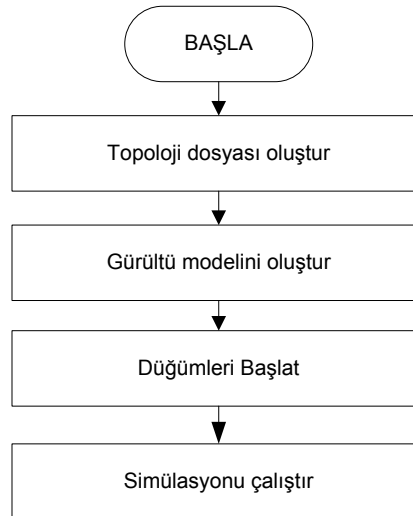
        event void AMSend.sendDone
            (message_t* giden_mesaj,error_t hata){

            if(&paketimiz==giden_mesaj){
                ..
            }
        }
    }
}

```

Şekil 2.20. Bir düğüm dosyasının yapısının genel gösterimi

Düğüm kodları yazıldıktan sonra “simulasyon.py” gibi bir Python dosyası oluşturularak gerekli TOSSIM kütüphaneleri “import” edilmelidir. Daha sonra yapılması gereken adımları temsil eden akış şeması Şekil 2.21’de gösterilmiştir.



Şekil 2.21. Simülasyon dosyasının akış şeması

Topoloji dosyasının okunmasını ve gürültü modelinin simülasyona eklenmesini temsil eden örnek kod kısımları Şekil 2.223 ve 2.24'de verilmiştir.

```
#####  
  
# Okunan topolojiyi ekranda gösterelim  
  
for satir in f:          #Dosya içindeki satırları okuyacak  
  
    dizi=satir.split()   #Tüm stringleri bir diziye aktar  
  
        if dizi:          #Dizi boş değilse  
            print "",dizi[0],"",dizi[1],"",dizi[2]; #satırı ekrana yaz  
            r.add(int(dizi[0]),int(dizi[1]),float(s[2]))#Radyo nesnesine yükle  
  
#####
```

Şekil 2.22. Topoloji dosyasının okunmasına ait örnek bir kod bloğu

```
gurultu=open("gurultu.txt","r")  
  
for satir in gurultu:  
    dizi2=satir.strip() #dikkat strip , split degil.  
    if dizi2:  
        deger=int(dizi2)  
        for i in range(1,10):  
            t.getNode(i).addNoiseTraceReading(deger)  
  
for i in range(1,10):|  
    print "Gurultu modeli olusturuldu. Dugum = ",i;  
    t.getNode(i).createNoiseModel()  
    t.getNode(i).bootAtTime(i*1000001+3500001)
```

Şekil 2.23. Gürültü dosyasının simülasyona eklenmesi ait örnek bir kod bloğu

İstenildiği takdirde arzu edilen düğüme istenilen zamanda bağlanılabılır [11]. Ancak simülasyonun akışının sağlıklı çalışabilmesi için etkileşim bittikten sonra gerekli bağlama işlemi kaldırılmalıdır. Şekil 2.24 a, TOSSIM başlangıç ayarlamalarına örnek kod bloğunu gösterirken, Şekil 2.24 b istenilen düğüme bağlanılma kodlarına nasıl bağlanıldığına ait örnek kod bloğunu temsil etmektedir.

Şekil 2.26’da Tkinter yardımıyla görsel simülasyon penceresinin nasıl hazırlandığına dair küçük bir kod bloğunu göstermektedir.

```
ana_pencere=Tk()
cerceve = Frame()
cerceve.pack(side="bottom",pady=5)
cizim_alani=Canvas(cerceve,bg='white',height=600, width=1100)
yol1=cizim_alani.create_line(350,25,350,150,150,150,150,125,fill="blue")
yol9=cizim_alani.create_line(100,125,100,275,75,275,fill="blue")
yol2=cizim_alani.create_line(75,325,100,325,100,500,175,500,fill="blue")
yol3=cizim_alani.create_line(400,25,400,150,625,150,625,125,fill="blue")
yol4=cizim_alani.create_line(675,125,675,225,fill="blue")
ana_pencere.mainloop()
```

Şekil 2.26. Simülasyon penceresinin çizilmesine ait örnek bir kod bloğu

3. KABLOSUZ SENSÖR AĞLARINDA KONUM BULMA

Kablosuz sensör ağlarında dağıtık bir yapı varsa konum belirleme işlemleri gerekli olabilir. Bu işlemler için birçok yöntem bulunmaktadır. Ancak her yöntem farklı bir ağ yapısı için uygundur. Bundan dolayı ilgili ağın parametreleri iyi bilinmelidir. Temel olarak konum saptama iki şekilde yapılır [6,7,12].

Birincisi, GPS aracılığı ile koordinat hesaplanması ve cihaza bu değerin girilmesidir. Bu yöntemde bütün düğümler sahip oldukları konumu kesin olarak bilirler. Diğer metot ise, konumu belirlenecek düğümün kendi koordinatını kendisinin hesaplamasıdır.

İkinci yöntem uygulanacaksa önceden bazı düğümler belirlenmeli ve bu düğümlerin koordinatları o cihazlara başlangıçta yüklenmelidir. Önceden belirlenmiş bu düğümler, belirli aralıklarla kendi koordinatlarını bildiren mesajlar yayınlayarak diğer algılayıcı düğümlerinin kendi koordinatlarını hesaplamaları için gereken konum bilgilerini bildirir. Referans değerlerini alan hedef düğümler, kendi konumlarını belirli metotlar kullanarak bulurlar.

İkinci yöntem için, hedef düğümlerin konum saptama için kullanacakları metotlar temel olarak ikiye ayrılır [7],

- Kapsama alanı temelli teknikler
- Kapsama alanından bağımsız teknikler

3.1.Kapsama Alanına Bağlı Konum Bulma

Kablosuz sensör ağlarında konum saptama tekniklerinden biri olan Kapsama alanına bağlı olarak konum bulma temelde altıya ayrılır [6,7,12].

3.1.1. Gelen Sinyalin Gücüne Bağlı Mesafe Tahmini (RSSI)

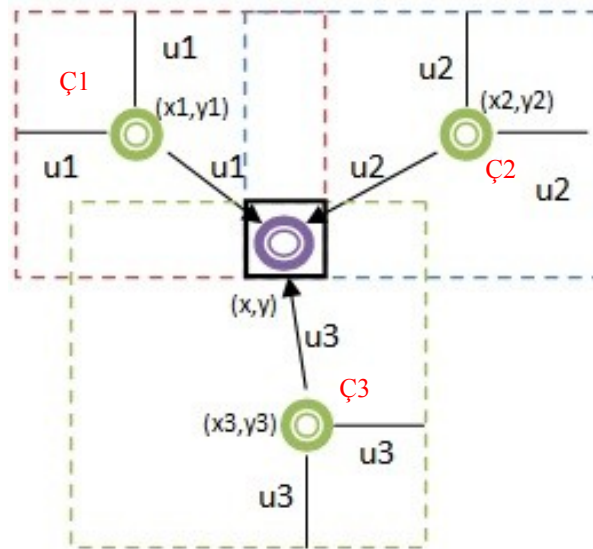
Bu çalışmada da kullanılacak olan RSSI ile mesafe tahmini metodu, gelen mesajın sinyal gücünü temel alır [54]. Gelen sinyalin gücü dB cinsinden ölçülür ve genel matematiksel bağıntısı Denklem 3.1’de gösterildiği şekildedir [25]. Denklemdeki “ K_0 ” 1mt için RSSI değeri iken, “ K_1 ” yol kaybı katsayısıdır. Belli oranlar kullanılarak mesajı yollayan düğüm ile mesajı alan düğümün arasındaki uzaklık hesaplanır. Konum tespiti için konumu önceden bilinen en az üç çapa düğümden değer alınması gereklidir.

$$RSSI = -K_0 - K_1 \log_{10}(\text{mesafe}) \quad (3.1)$$

TelosB düğümü CC2420 radyo devresini kullanır ve düğümden okunan RSSI değeri doğrudan kullanılmaz. Elde edilen değere CC2420 kataloglarında belirtilen -45 dBm’lik bir ofset değerinin eklenmesi gereklidir [54-56]. Bu metodun ekonomik olması önemli bir avantajdır. Bununla beraber çoklu sinyal yayılımı, sinyal gücünün havadaki iletimi esnasında değişmesi ve sinyallerin karışması gibi birçok faktör bu metod için problem teşkil ederler.

Çapa düğümlerden elde edilen uzaklık bilgilerine göre kullanılacak olan bir konum tespit yöntemi ile hedef düğümün bulunabileceği alan değişen bir hata miktarıyla tahmin edilebilir.

Bu çalışmada kullanılacak olan konum tespit yöntemi “**Bounding Box**” metodudur [55]. Bu metodun mantığı oldukça kolaydır. Koordinat bilgileri (x,y) bilinen her bir çapa düğümün hedef düğüme uzaklığı RSSI tekniği ile hesaplandıktan sonra her bir çapa düğümün koordinatlarının x ve y değerlerine bu mesafe eklenerek bir karesel alan oluşturulur. Her bir çapa düğümün oluşturduğu karesel alanların kesişim bölgesi hedef düğümün bulunabileceği alanı gösterecektir. Şekil 3.1 bu yöntemin geometrik olarak gerçekleştirilmesini göstermektedir. Söz konusu şekilde, ortadaki düğüm, yeri tespit edilecek olan hedef düğümü temsil ederken diğer üç düğüm (Ç1,Ç2,Ç3) koordinatları belli olan çapa düğümlerdir.



Şekil 3.1. Bounding Box yöntemi

RSSI ile konum tespiti her zaman güvenilir sonuçlar vermez. Hataya yatkın bir yöntem olduğundan, kullanıldığı ortamın ve kullanılan elemanların parametreleri iyi analiz edilmeli ve gerekli kalibrasyon işlemleri dikkatlice yapılmalıdır. Aksi takdirde istenilen sonuca ulaşmak zor olabilir. Bu uygulama hem bina içi hem açık alanda gerçekleştirilebilir.

Konum tespiti yöntemleri şüphesiz Bounding Box metodu ile sınırlı değildir. Birçok farklı metod konum tespitlerinde kullanılmaktadır. Bunlardan biri, yukarıda anlatılan yöntemde kullanılan karesel çerçeveleme yerine dairesel çevreleme tekniğinin kullanılmasıdır ki burada çember denklemlerinin kullanımıyla yaklaşık bölge tahmini yapılmaktadır [6,7,52].

Üçgen içindeki yaklaşık nokta tespiti yönteminde ise APIT algoritmasından faydalanılır[52]. Bu algoritmada “kesinlik değeri” olarak bilinen bir referans bilgi gereklidir. Çapa düğüm bilgilerine bağlı olarak, belirli üçgenler içerisinde arama yapan sistem, bulduğu değerlerin işlem sonuçlarını kesinlik değeri ile karşılaştırır. Daha sonra elde edilen üçgenin ağırlık merkezi hesaplanarak yaklaşık konum tespiti gerçekleştirilir.

3.1.2. Sinyal Açısına Bağlı Konum Bulma

Düğüm üzerinde, gelen sinyalin açısını hesaplamaya yarayan özel anten yapısına ihtiyaç duyulan bu teknikte, konum, gelen sinyalin açı bilgisi ile hesaplanır. Koordinatları önceden belirlenmiş en az iki düğüm, yayınladığı mesaj bilgisi ile etrafında bulunan komşu düğümlerin konumlarını bulmasını sağlarlar. Düğümler üzerinde bu teknik için özel donanımlara ihtiyaç duyulması hiç şüphesiz bu tekniğin en büyük çıkmazıdır. Kullanılacak düğüm teknolojisinin bu sisteme uygun tasarlanmış olması, kullanılacak anten tiplerine ait teknik özelliklerin iyi olması metodun doğru kullanılmasına doğrudan etki eder [6,7,12,52].

3.1.3. Geliş Süresine Bağlı Konum Bulma

Bilindiği üzere uzaklık ve mesaj gönderim-alım süresi arasında doğru orantı vardır. Hedef düğümün konumunu saptamada kullanılan en eski ve temel yöntemlerden olan geliş süresine bağlı konum bulma metodunda, en az üç düğümün bilgisine ihtiyaç duyulur. Bir düğümden sinyal alındıktan sonra hedef düğüme karşı mesaj yollanır. Geçen süre ile aradaki uzaklık hesaplanır. Bu teknik için en büyük problem Düğümler arasındaki saatlerin eş zamanlı olmamasıdır [7,12,52].

3.1.4. Geliş Süresi Farkına Bağlı Konum Bulma

Bu teknik yukarıda anlatılan geliş süresine bağlı konum bulma tekniğinin bir benzeri olup bir önceki metottan farklı olarak, ışık ve ses gibi iki farklı sinyalinin kullanılmasıyla uygulanır [12,25,52]. Konum bulma hesaplamalarında kullanılacak olan referans bilgisi, söz konusu bu iki farklı sinyalin geliş sürelerine ait bilgiler olacaktır. Bu metodun ek donanımlara ihtiyaç duyması bu tekniğin dezavantajı olarak öne çıkar.

3.1.5. Parçalanamaz Çoklu Üçgenleme

Üç adet düğümden gelen bilgiler ışığında, Pythagoras teoremi kullanılarak gerekli hesaplamalar yapılır [6,7,12,52]. Yapılan konum tespiti belirli hata oranı ile gerçekleştirilir. Söz konusu düğümlerin konumlarına bağlı yapılan analizlerde bu hata oranı bazı özel durumlarda çok daha büyük değerlere ulaşabilir.

3.1.6. Tekrarlamalı Üçgenleme

Koordinatları önceden belirlenmiş en az üç düğümden gelen verileri alan her bir düğüm kendi konumunu hesaplar ve kendini referans göstererek kendi bilgisini komşu düğümlere gönderir [6,7,12,52]. Hedef düğümün konumunun saptanmasında özel algoritmalar kullanılır. Oluşturulan üçgen alanlarında yapılan taramalarda çok sayıda hedef düğüm bulunması durumunda hata miktarıda büyüyecektir.

3.2. Kapsama Alanından Bağımsız Konum Bulma

Bu metod diğer tekniklere göre çok daha büyük hata oranına sahip olmakla beraber oldukça basit yapıya sahiptir. Burada yine koordinatları önceden belli olan referans düğümler kullanılır. Bu düğümler kendilerine ait x ve y bilgilerini belirli aralıklarla yayınlırlar. Hedef düğüm N tane referans düğümden gelen koordinat bilgileri (3.2)'de gösterilen formülle hesaplar [6,7,12,52].

$$X, Y = \frac{x_1+x_2+\dots+x_n}{n}, \frac{y_1+y_2+\dots+y_n}{n} \quad (3.2)$$

Hiç şüphesiz hesaplanan değer in doğruluğu referans düğümlerinin sayısı ile doğru orantılıdır.

4. SİSTEMİN SİMÜLASYONU VE UYGULAMASI

Bilindiği üzere navigasyon ve yönlendirme-takip sistemleri günlük hayatta ve iş hayatında kullanılan oldukça faydalı teknolojik yeniliklerdir. En bilindik navigasyon ve takip yöntemi ise hiç kuşkusuz GPS'dir. Ancak bu sistemin en büyük dezavantajı kapalı mekanlarda ve uyduyu görmediği noktalarda etkisiz olmasıdır.

Bu çalışmada; büyük kapalı mekanlarda GPS'in işlevini yerine getirebilecek, hem yönlendirme hem de takip işlemlerini (konum tespiti) kablosuz sensör ağlarını kullanarak yapabilen bir interaktif simülasyon ve uygulama platformunun gerçekleştirilmesi açıklanmıştır.

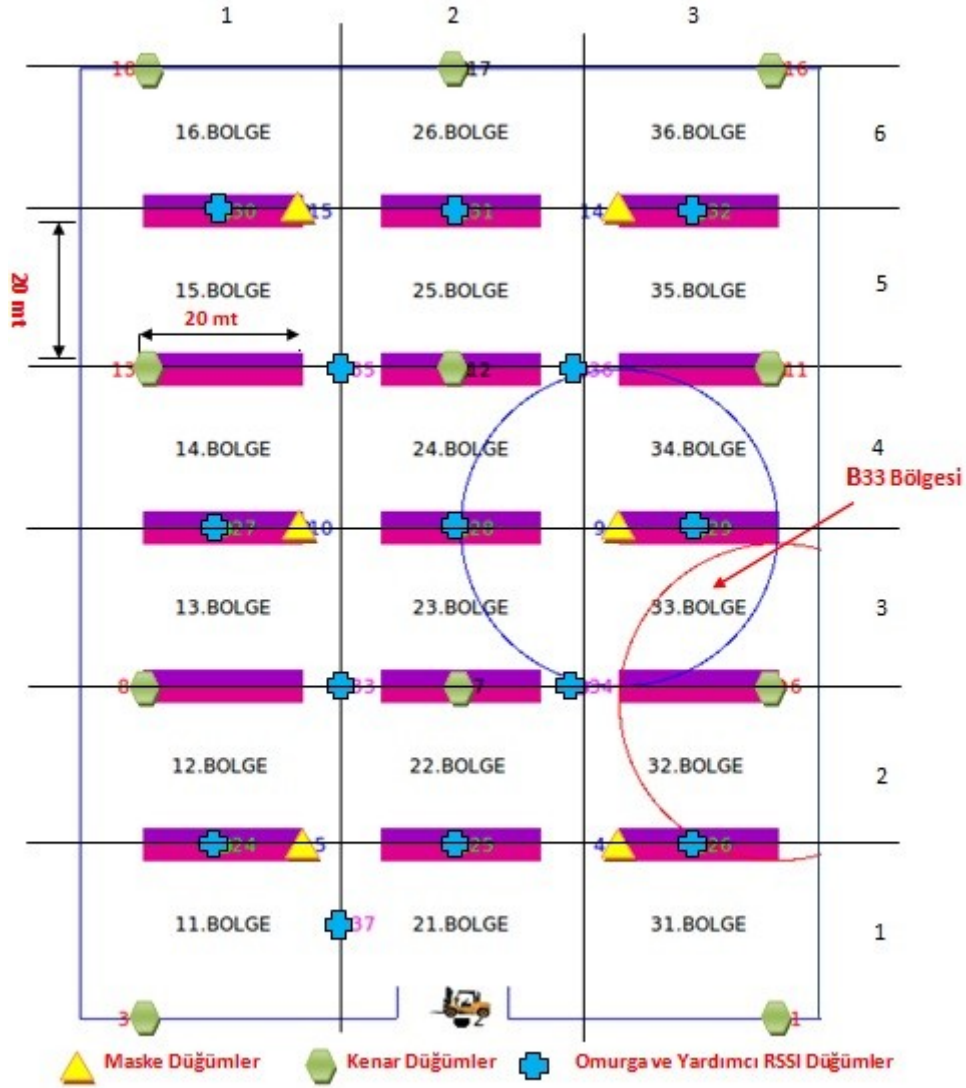
Bunun için “Kapalı bir ortamda araç yönlendirme ve takip uygulaması” isimli bir senaryo seçilmiş olup, senaryonun pratikte uygulanabilir olmasına özen gösterilmiştir.

Uygulama senaryosunun temel işlevi; içerisinde binlerce çeşit malın bulunduğu büyük kapalı alanları kapsayan depolarda mobil araçların, aranacak malzemeye hızlı bir şekilde ulaşılabilmesini sağlayacak bir WSN yapısı oluşturulması şeklindedir. Bu uygulamanın iki amacı vardır; birincisi böyle bir çalışma alanında forklift, kamyon vb. mobil araçların seçilen malların bulunduğu terek bölgelerine hızlı bir şekilde yönlendirilmesi, ikincisi ise mobil araçların alan içindeki konumlarının merkez ofis tarafından saptanabilmesidir.

Gerçekleştirilen sistemde, ayrıca depo merkez ofisinden, birden fazla mobil aracın, depo içerisinde yönlendirilmesi yapılabilmekte, sahadaki mobil araca WSN düğümleri aracılığı ile yeni komutlar verilebilmektedir.

Kapalı bir ortamda araç yönlendirme ve takip uygulaması senaryosu için; 12.000 m² lik (100x120m) kapalı bir hangar depo ortamında sabit ve numaralanmış terek gruplarının olduğu ve her bir terek gurubundaki ürünlerin listelerinin de bir veritabanında tutulduğu varsayılmıştır. Şekil 4.1'de uygulama deposunun krokisi ve sensör düğümlerinin konumları gösterilmektedir. Mobil düğümün depodaki gezen iş araçlarına monte edildiği varsayılmıştır. Mobil aracın istenilen yere gönderilebilmesinin sistematik olarak yapılabilmesi için söz konusu çalışma alanı matrisel olarak haritalanmış olup her bir bölge bir matris elemanı gibi etiketlenmiştir. Böylece binlerce çeşit malın bulunduğu tereklerin hangi matris hücresinde (bölgesinde) olduğu bir veri tabanında tutulması suretiyle, istenilen malzemelerin hangi terekte ve dolayısıyla hangi hücrede olduğu rahatlıkla bilinmektedir. Bu suretle istenilen malzemelere yönlendirilecek mobil düğümlere o

malzemelerin bulunduğu hücre numarası (bölge kodu) ve ürün kodunun bildirilmesi yeterli olacaktır. Böyle bir tasarımın gerçekleşmesinde “Kenar Maskeleme” olarak isimlendirilen bir yöntem geliştirilmiştir. Bu yöntem sayesinde iş aracının bulunduğu konumun yönlendirme düğümü tarafından daha iyi analiz edilmesi sağlanmıştır. Mobil düğümün bulunduğu bölge analizi “maske” düğüm adı verilen düğümler tarafından yapılacak ve gerekli yönlendirme bilgileri mobil düğüme gönderilecektir. Kenar düğümler ise bölgelerin birbirlerinden ayırt edilmesi için kullanılacaktır. Bunların dışında sistemin merkez ofis ile iletişimini sağlamak için omurga vazifesi görecektir sensör düğümleri ile RSSI konum tespitinde kullanılacak yardımcı düğümler Şekil 4.1’deki gibi konumlandırılmıştır.



Şekil 4.1. Uygulamanın yapıldığı alan ve düğümlerin yerleşimi

4.1. Simülasyon

Sistemin simülasyonunda, TinyOS işletim sisteminin kendi simülatörü olan TOSSIM kullanılmıştır. Böylece gerçek uygulamada kullanılacak olan TelosB düğümüne ait kodlarda emüle edilecektir. Şekil 4.1’deki 100 x 120 mt (500x600 piksel)’lik kapalı alanda sensör düğümlerin, tereklerin, mobil düğümlerin konumlarının belirli bir referansa göre bilinmesi gerekir. Bunun için kapalı alan sol üst köşesi (x=0,y=0) koordinatları olmak üzere, 20 x 30 m’lik ızgaralar (bölge) şeklinde modellenmiştir. Her bir ızgara alanı “Bij” olarak etiketlenmiştir. Burada “i” indisi düşey kodu, “j” indisi yatay kodu temsil etmektedir. Kodlama mantığında j değerinin artması bir üst bölgeyi temsil etmektedir. Burada önemli olan hücrelerin boyutlarıdır ki bu boyutlandırma yapılırken kullanılacak olan kablosuz ağ sensörün kapalı alan (indoor range) kapsama alanı dikkate alınmalıdır. Planlamada oluşturulan bölgelerin en ve boy uzunlukları, kullanılacak olan sensörün bina içi kapsama alanından küçük olmamalıdır. Hücre uzunluğu “hu” , hücre genişliği “hg” ve düğüm bina içi kapsama alanı “dr” olmak üzere denklem dd1 bu kuralı ifade etmektedir.

$$\left. \begin{array}{l} hg \geq dr \\ hu \geq dr \end{array} \right\} \quad (4.1)$$

Aksi takdirde farklı bir bölgede de istenmeyen düğümlerin sinyalleri alınabileceğinden tasarım ve programlama güçlükleri meydana gelecektir. Çalışma alanının matrissel haritalaması yapılırken bu değer göz önünde bulundurulmalıdır. Gerekirse planlamadan önce ilgili alanda düğümün kapsama alanı denenmelidir. Bu çalışmanın hedefi, büyük ölçekli kapalı çalışma alanları olduğundan senaryo bu mantığa göre tasarlanmıştır. Bu çalışmada söz konusu kapsama alanı mica, telosb, imote gibi sensör düğümlerin kapalı alan kapsama mesafesi olarak ortalama 20 mt kabul edilmiştir. Varsayılan çalışma alanında tır, kamyon ve diğer büyük ve orta büyüklükteki iş makinelerinin yan yana geçebildiği ve rahatça manevra yapabildiği bölgeler düşünülmüş, haritalama yaparken esas alınan düğüm yarıçapı buna ek olarak hesaba katılmıştır.

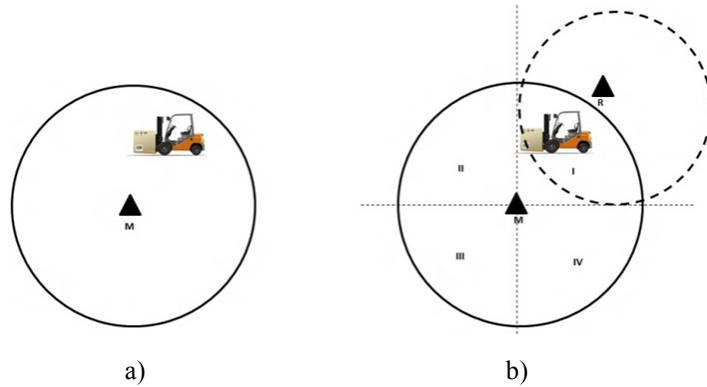
Önerilen sistem, mobil düğümler, maske düğümler, kenar düğümler ve omurga düğümler olmak üzere temelde dört tip düğüm yapısından oluşur. Buradaki mantık; her bir hücrenin WSN içinde etiketlenebilmesi için biri maske düğüm olması şartı ile en az iki farklı “ tip ” düğümünden sinyal almasıdır. Burada anılan “tip” ifadesi teknolojik farklılık değil sistem içerisindeki çalışma farklılığıdır. Düğümler programlanırken mantıksal olarak

işleyiş farklılıkları göz önünde bulundurulur. Örneğin mobil düğüm “tip1” olarak ifade edilirken maske düğümler “tip2 ” olarak ifade edilmektedir. Aynı zamanda düğümler gönderdikleri veri paketlerinin içerisine kendilerine ait bu tip değerini de eklemektedirler. Bu teknik ile, düğümlerin ID’lerine göre değil, görev tiplerine göre analizi yapılacaktır. Bu da düğümlerin programlamasında kolaylık sağlayacaktır.

Maske düğümlerin kapsama alanları birbirlerini kesmemelidir. Düğümlerin yerleştirilmesinde farklı yaklaşımlar ve optimizasyon teknikleri kullanılabilir. Maske düğüm bu sistemde aktif rol oynarken, kenar düğümler ayırt edici özellik sunarlar. Hedef bölgesi belli olan mobil düğümleri algılayıp yönlendirme işlemini yapacak olanlar maske düğümlerdir.

4.1.1. Kenar Maskeleme Yöntemi İle Yönlendirme Simülasyonu

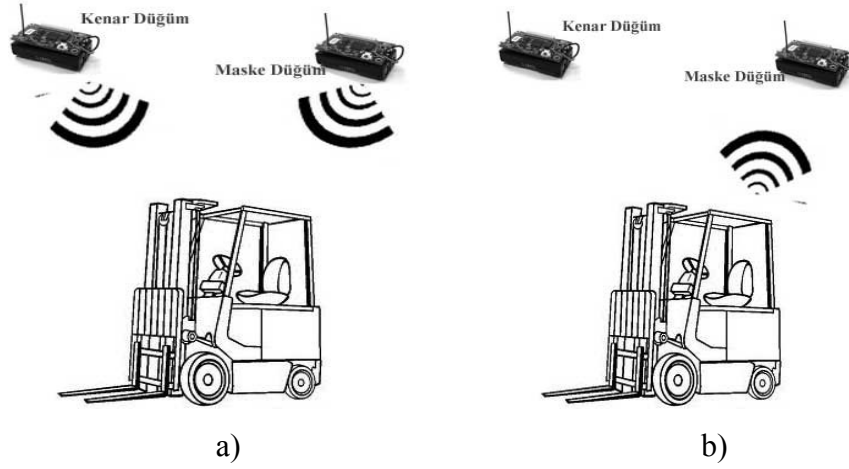
Çalışma alanında forklift, kamyon gibi mobil araçların, seçilen tereklerin bulunduğu bölgeye yönlendirilmesi için mobil düğümün bulunduğu bölgenin net olarak bilinmesi gerekir. Bunu sağlamak için; en az iki farklı çalışma moduna sahip sensör düğümden bilgi alınmasına dayanan “Kenar Maskeleme Yöntemi” geliştirilmiştir. Bu yöntem, birçok sensör düğümünün ortak kapsama alanında bulunan bir mobil düğümün yerinin daha net bir şekilde saptanabilmesini garantiler.



Şekil 4.2.İki sensör düğüm ile hedef saptama

Şekil 4.2 a ‘da görüleceği üzere “M” düğümü, kendi kapsama alanı içine giren mobil düğüm ile haberleşerek söz konusu aracın kendi bölgesinde olduğunu anlayacak fakat bu mobil düğümün kendi kapsama alanının neresinde olduğunu saptayamayacaktır. Ancak ikinci bir “R” düğümünün kapsama alanı, uygun bir şekilde “M” düğümünün kapsama alanını keserse ve söz konusu mobil düğüm, bu kesişim bilgisini, “M” düğümüne iletebilirse o zaman “M” düğümü, gelen kesişme verisinin kendisinde tanımlı olup

olmadığına bakarak, mobil aracın kapsama alanının neresinde olduğunu daha iyi anlayabilecektir.



Şekil 4.3. Mobil düğümün ortamdaki veri toplaması ve topladığı bilgileri yayınlaması

Bu çalışmada mobil düğümün yönlendirilmesi için geliştirilen “kenar maskeleme” yöntemi şu şekilde açıklanabilir; Araçların içindeki mobil düğümler, geçtikleri alanlarda ortamdaki aldığı paketlerin hangi tip düğüme ait olduklarını analiz edip kendi hafızalarında saklar. Eğer maske düğüm tipine sahip bir sinyal alırsa bunu hafızada “maske” değişkeninde, kenar düğüm tipinde bir sinyal alırsa bunu “kenar” değişkeninde saklayacaktır. Ayrıca mobil düğüm konum tespitinde kullanılacak olan RSSI değerlerini de hafızasında tutmalıdır. Mobil düğüm; kendi düğüm ID’si ve Düğüm Tipini, en son sinyal aldığı Maske ID, Kenar ID, gitmek istediği bölgenin düşey (n) ve yatay (m) kodları ve RSSI1 ve RSSI2 değerlerini içeren veri paketini periyodik olarak ortama yayınlamaktadır. Tablo 4.1’de bu veriler gösterilmiştir.

Tablo 4.1. Mobil düğümün periyodik olarak yayınladığı payload

Düğüm ID	Düğüm Tipi	Maske ID	Kenar ID	n	m	RSSI1	RSSI2
----------	------------	----------	----------	---	---	-------	-------

İlgili maske düğüm; Mobil düğümlerden gelen veri paketlerinden, Maske ID, Kenar ID ve hedef bölge kod bilgilerini (n,m) elde ederek Maske ID, Kenar ID’lerinin kendi karşılaştırma tablosunda olup olmadığına bakar. Tablosunda tanımlı Maske ve Kenar ID’lerine göre Mobil düğümün o an bulunduğu bölgeyi belirler. Örneğin Maske ID =9, Kenar ID =7 ise Tablo 4.2’ye göre mobil düğümün B23 bölgesinde olduğu belirlenir.

Tablo 4.2.9 nolu maske düğümün karşılaştırma tablosu

Maske ID	Kenar ID	Mobil düğümün Algılandığı bölge (B _{ij})
9	6	B33
9	7	B23
9	11	B34
9	12	B24

Mobil düğümün bulunduğu bölge belirlendikten sonra, mobil düğümdeki gidilmek istenen hedef bölgenin (D_{nm}) yatay (m) ve düşey(n) indis kontrolü yapılır. Bu kontrol; “ B_{ij} ” mobil düğümün algılandığı bölge ve “ D_{nm} ” ise hedef bölgeyi temsil etmek üzere Denklem 4.2’ye göre yapılır.

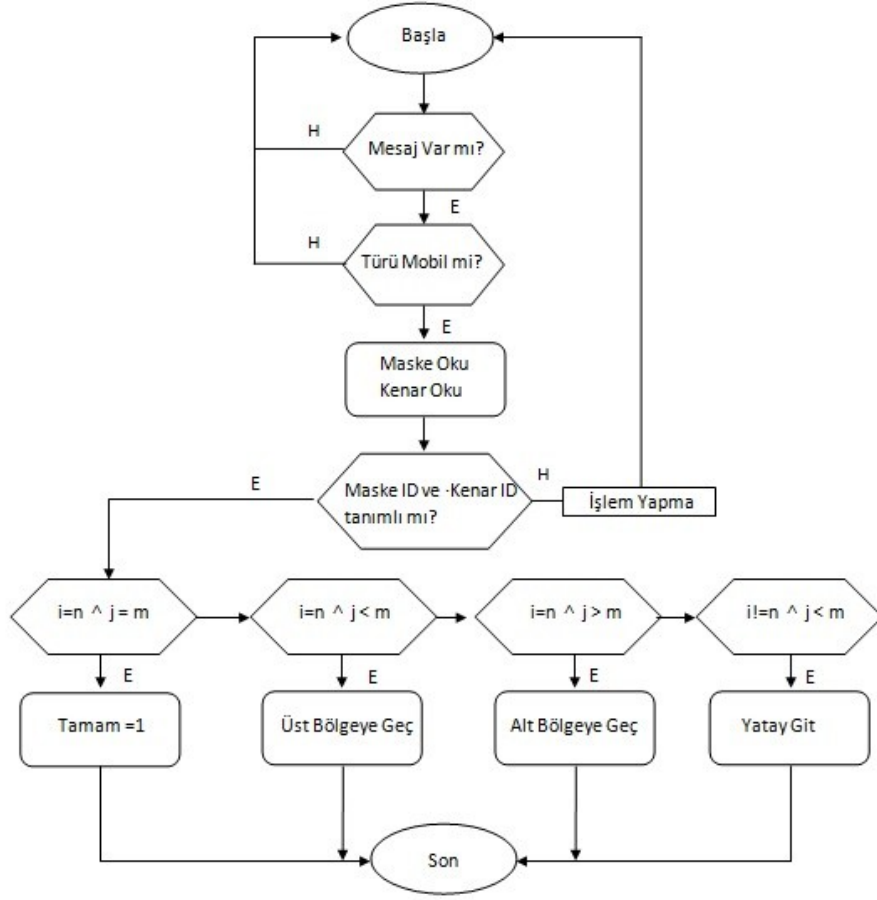
$$\begin{aligned}
 i = n \wedge j = m &\Rightarrow B_{ij} \text{ istenilen bölge} \\
 i = n \wedge j < m &\Rightarrow \text{Hedef bölge için üst bölgeye çık} \\
 i = n \wedge j > m &\Rightarrow \text{Hedef bölge için alt bölgeye in} \\
 i \neq n \wedge j = m &\Rightarrow \text{Hedef bölge için yatay hareket et}
 \end{aligned}
 \tag{4.2}$$

Gerekli kontroller yapıldıktan sonra uygun yönlendirme bilgileri maske düğüm tarafından mobil düğümün alması için gönderilir. Gönderilen veri paketinde Tablo 4.3 ‘deki bilgiler bulunmaktadır. Tablo 4.3 ‘den görüleceği üzere, maske düğüm kendisine ait ID ve tip bilgilerinin yanı sıra, hedef bölgeye ulaşıldığını bildiren “Tamam” bilgisi, yönlendirme bilgileri (üst bölgeye git, alt bölgeye git, yatay git), ilgili hedef bölgeye ait düşey kod bilgisi (Düşey Hedef) mobil düğümüne iletilir.

Tablo 4.3.Maske düğümün mobil düğümüne gönderdiği veriler

Düğüm ID	Düğüm Tip	Tamam	Üst bölgeye git	Alt Bölgeye git	Yatay Git	Düşey Hedef
----------	-----------	-------	-----------------	-----------------	-----------	-------------

Yukarıda açıklanan kenar maskeleme tekniğinde kullanılan yönlendirme işlemini gerçekleştirecek sensör düğümlerin programlanmasına ait akış diyagramı Şekil 4.4’te gösterilmiştir.

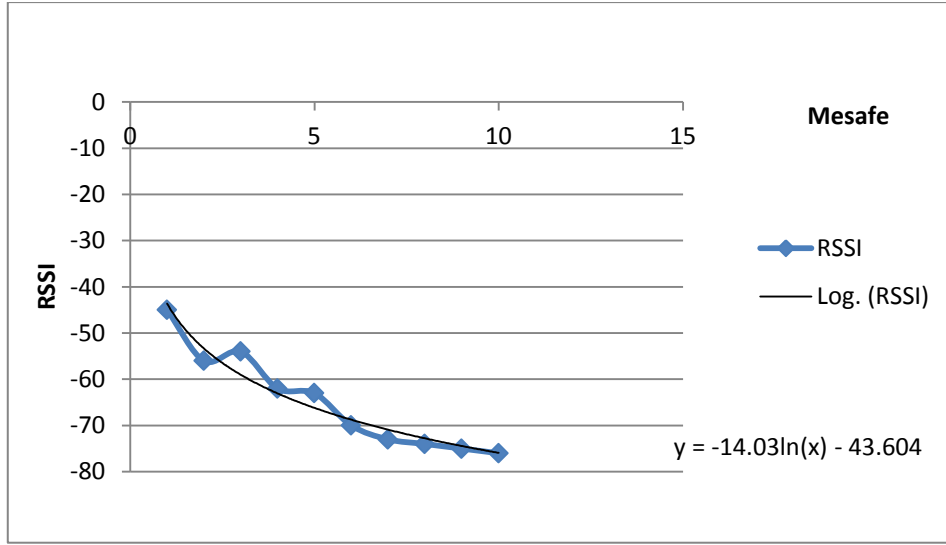


Şekil 4.4.Yönlendirme Bilgisini Sağlayan Koda ait akış şeması

Yönlendirme için “üst bölgeye git”, “alt bölgeye git”, “yatay git”, “tamam” bildirimlerinin yanı sıra hedef düşeyi gösteren 1, 2, 3 nolu ledler kullanıldığı kabul edilmiştir. Burada sistemin ölçeklendirilebilir olması ve daha büyük alanlar için daha fazla düğümün kullanılabilmesinin önünü açmak için gerekli kontrol ve karar verme görevi maske düğümlere verilmiştir. Ancak planlama yanlış yapıldığında veya bir hücreye istenilenden fazla kenar düğüm sinyali geldiğinde, maske düğümdeki kontrol işlemi daha karışık bir hale gelecektir. Her ne kadar kontrol işleminde “tip” özelliğinin yanında düğüm kimliğini bildiren “NodeID” kullanılsa da planlama aşamasında bir hücreye çok fazla kenar düğüm sinyalinin gelmemesine dikkat edilmelidir.

4.1.2. Konum Bulma ve İzleme Simülasyonu

Söz konusu çalışma ortamında araçların konumlarının tespiti için RSSI yönteminin kullanıldığı kabul edilmiştir. Simülasyonda konum belirleme ve izleme için Bölüm 2 ‘de anlatılan “Bounding Box” metodu kullanılmış olup,mesafe-RSSI ilişkisinin hesabında Şekil 4.5 ‘te gösterilen grafik ve denklem kullanılmıştır.

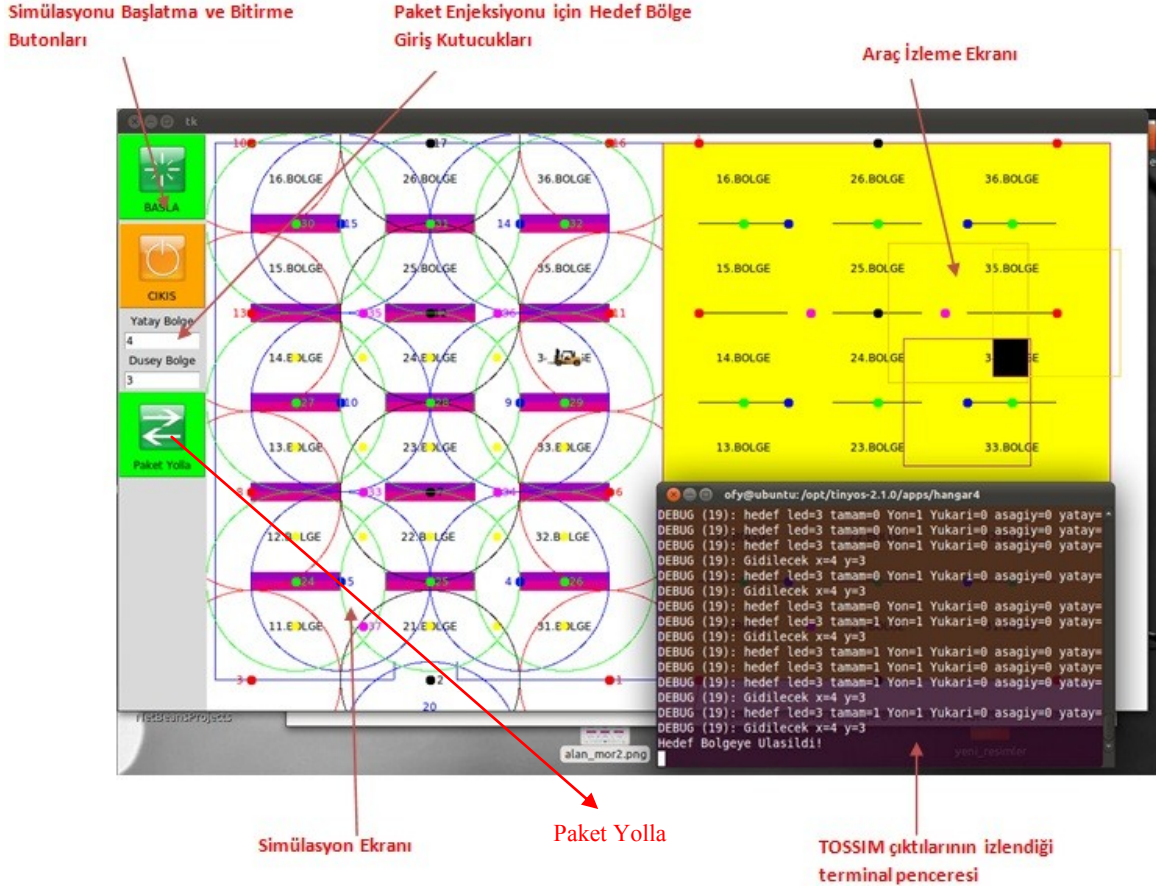


Şekil 4.5. Simülasyonda kullanılan RSSI-Mesafe ilişkisi

4.1.3. Simülasyonun Gerçeklenmesi

Söz konusu uygulamanın simülasyonu için Bölüm 4.1.1 ‘de açıklanan “kenar maskeleme tekniği”, Bölüm 4.1.2 ‘de açıklanan “Konum bulma ve izleme yöntemi” ve Bölüm 4.1 ‘deki kapalı alan modeli birlikte kullanılmıştır. Mobil aracın konum simülasyonu, kapalı alanın matematiksel modeli kullanılarak yapılmıştır. Böylece mobil aracın hangi matris hücresinde olduğu, simülasyon tarafından bilinmektedir. Simülasyonda kapalı alan üzerine ölçekli olarak yerleştirilen maske, kenar, RSSI yardımcı düğümleri ile mobil aracın üzerindeki düğüm, yazılan TOSSIM simülasyon kodları tarafından etiketlenir. Bu etkileşim sonucunda her düğüme ait önceden yazılan düğüm görev kodları çalıştırılır. Simülasyonun çalışma mantığı kısaca bu şekildedir. Simülasyonda kullanılan düğüm kodları nesC dili ile yazılmış olup aynı zamanda bu kodlar gerçek düğümlerin programlanması içinde kullanılacaktır. Tkinter grafik kütüphanesi yardımıyla düğümlere ait kodlar TOSSIM simülatöründe görsel hale getirilmiştir. Simülasyon esnasında belirli bir noktada bulunan bir araca veri gönderilme işlemi TOSSIM’de bulunan “paket enjeksiyonu” yöntemiyle yapılmıştır [11]. Söz konusu simülasyona ait ana pencere görüntüsü Şekil 4.6’da verilmiştir. Pencerenin sol tarafında simülasyonu başlatıp kapatan butonlar, paket enjeksiyonunda yeni yatay ve düşey konumların girileceği alanlar ve paket enjeksiyonunu başlatacak olan “Paket Yolla” butonu bulunmaktadır. Ekranın orta penceresi simülasyon alanının görüntüsünü gösterirken, sağ taraftaki sarı alan, merkez

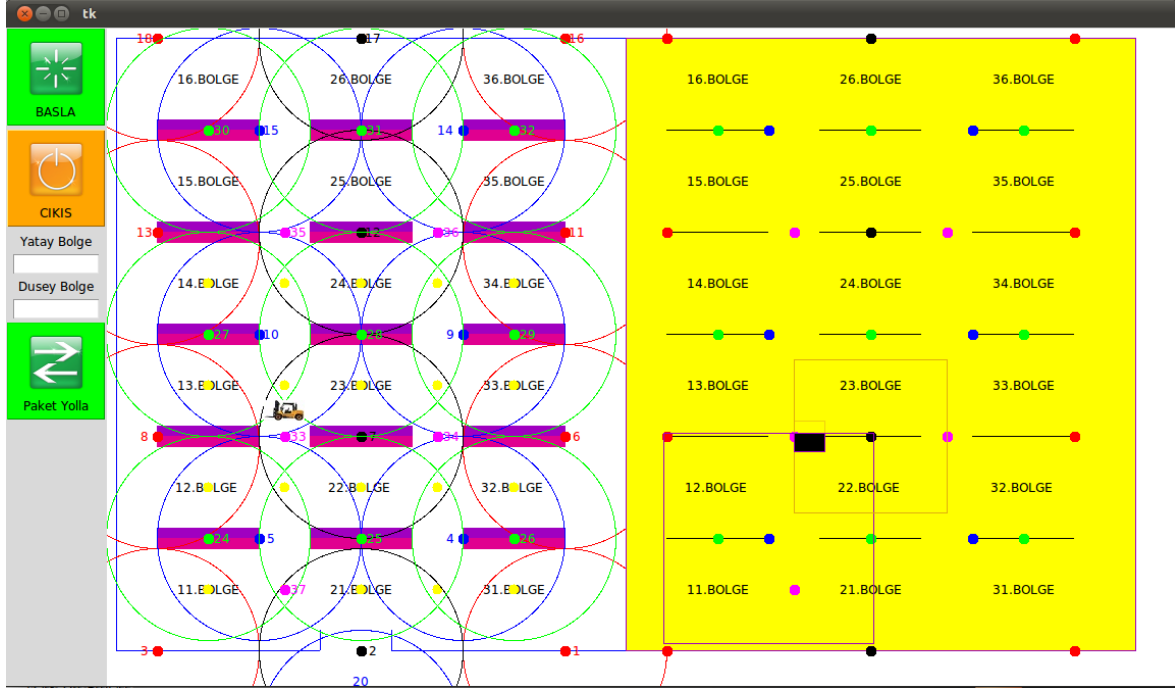
birimdeki RSSI yöntemi ile araçların hareket ve konumlarının tespitinde kullanılan izleme ekranını temsil etmektedir.



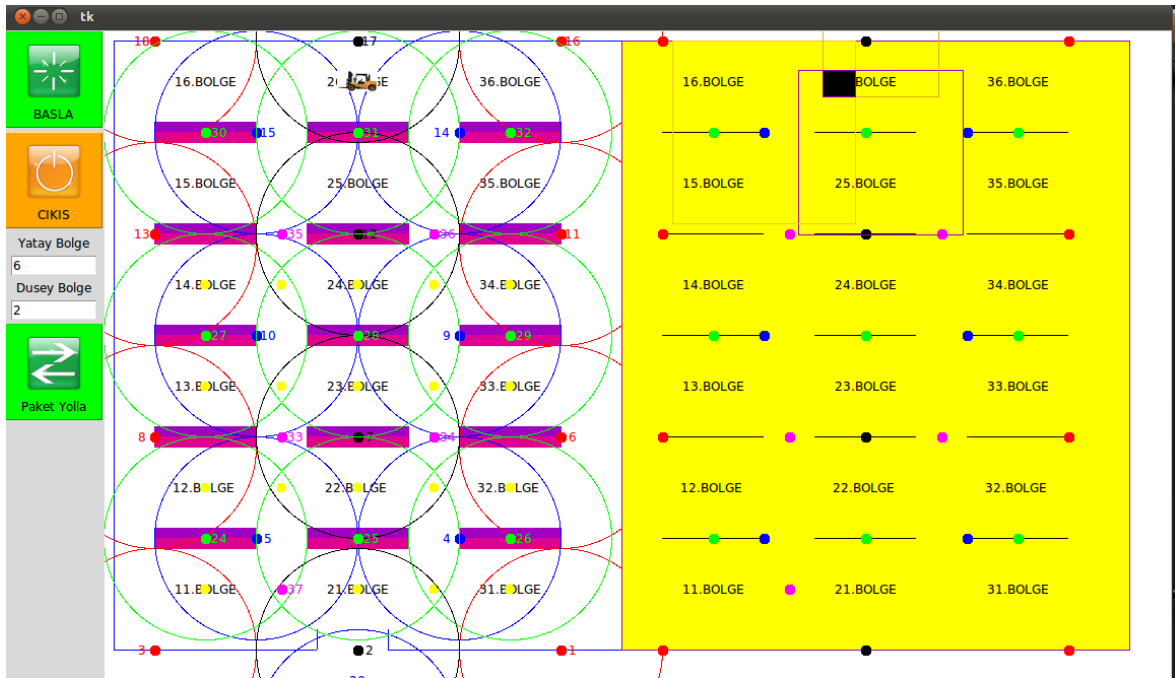
Şekil 4.6. Simülasyon ekranı

Gideceği hedef bölge bilgisi yüklenen mobil düğümün başlangıçta tüm yönlendirme değişkenleri sıfırdır. Maske ve kenar düğümlerden aldığı bilgileri kaydederek tetiklenme süresi dolduğunda bu verileri yayınlamaktadır. En yakın maske düğüm mobil düğümden gelen verileri okuyacak ve kendisinde tanımlı olup olmadığını kontrol edecektir. Eğer tanımlı maske ve kenar düğümler ise mobil düğümün gitmek istediği hedef bölgenin düşey ve yatay indislerini kendi tanımlı olduğu bölgenin yatay ve düşey indisleri ile karşılaştıracaktır. Karşılaştırma sonucunda, yönlendirme değişkenlerine gerekli bilgileri yükledikten sonra veri paketini, tetikleme süresi dolunca, mobil düğümün alması için ortama bırakılacaktır. Yönlendirme verisini alan mobil düğüm gerekli bildirimleri led veya ekran vasıtası ile sürücüye bildirecektir. Şekil 4.7, 4.8, 4.9 ve 4.10’da simülasyondan alınan ekran çıktılarından göstermektedir. Örnek olarak Şekil 4.7’de içinde mobil düğümün bulunduğu iş aracı, B26 (26. Bölge)bölgesine gitmesi bilgisini almıştır. Araç hareket halinde

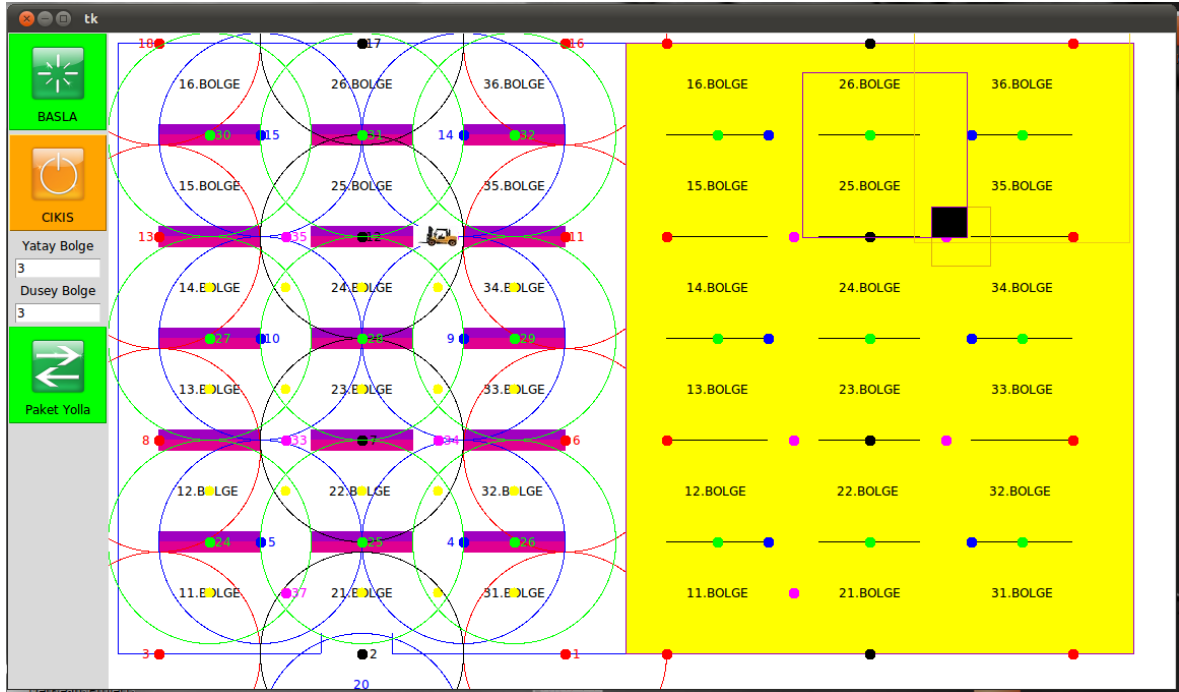
iken, merkezi izleme bölgesinde aracın bulunduğu bölgeye ait tahmini bölge siyah alan olarak belirtilmektedir. Mobil düğümün hangi bölgede hangi düğümlerle etkileşeceği, piksel bilgileri baz alınmak kaydıyla, simülasyon kodlarının yazımı esnasında belirlenmiştir.



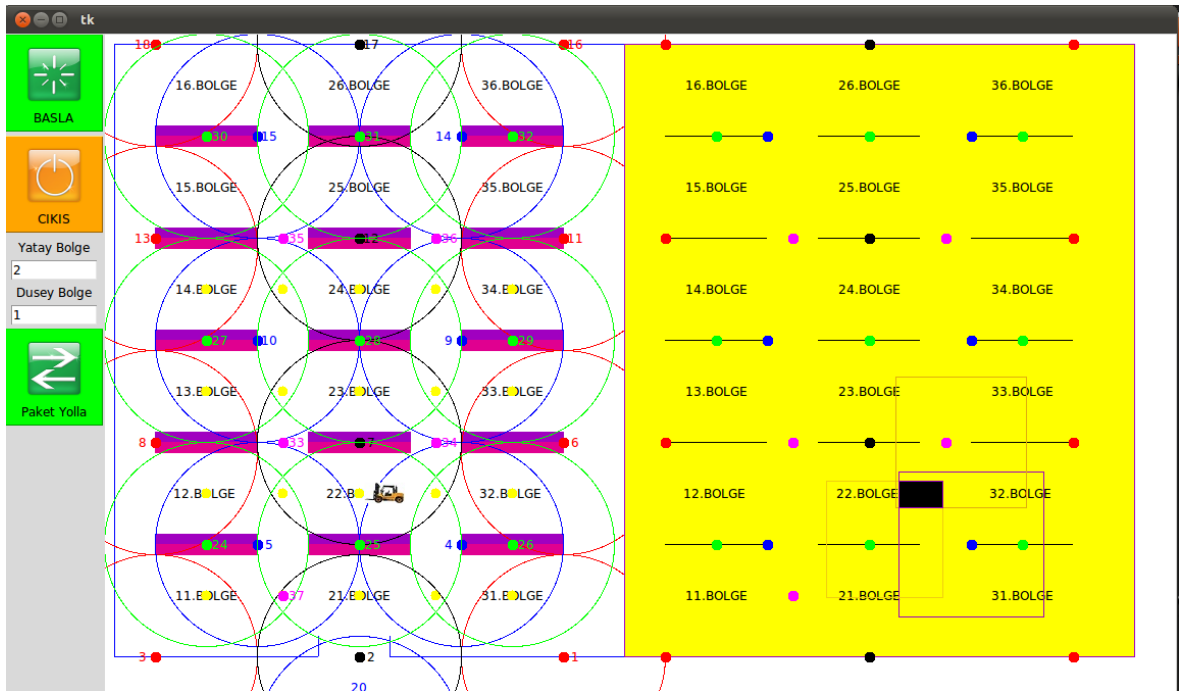
Şekil 4.7. Araç B21 bölgesinden B26 bölgesine hareket etmesi



Şekil 4.8. Aracın B26 bölgesine ulaşması



Şekil 4.9. Aracın B26 bölgesinde iken B33 bölgesine gitmesi için merkezden paket alması ve harekete geçmesi



Şekil 4.10. Aracın B33 bölgesinden B12 bölgesine paket alması ve harekete geçmesi

Önerilen uygulama projesi için geliştirilen simülasyon başarıyla çalışmıştır. Bundan sonraki aşama simülasyonda kullanılan düğüm kodlarının TelosB düğümleri kullanılarak denenmesi aşamasıdır.

4.2. Uygulama

Söz konusu WSN sisteminin gerçekleştirilmesi için, Bölüm 4.1.1 'de açıklanan “Kenar Maskeleme Tekniği” ve Bölüm 4.1.2 'de açıklanan “Konum Bulma ve İzleme Yöntemi” gerçekleştirme uygulamaları bu bölümde açıklanacaktır.

4.2.1. Konum Belirleme ve İzleme Uygulaması

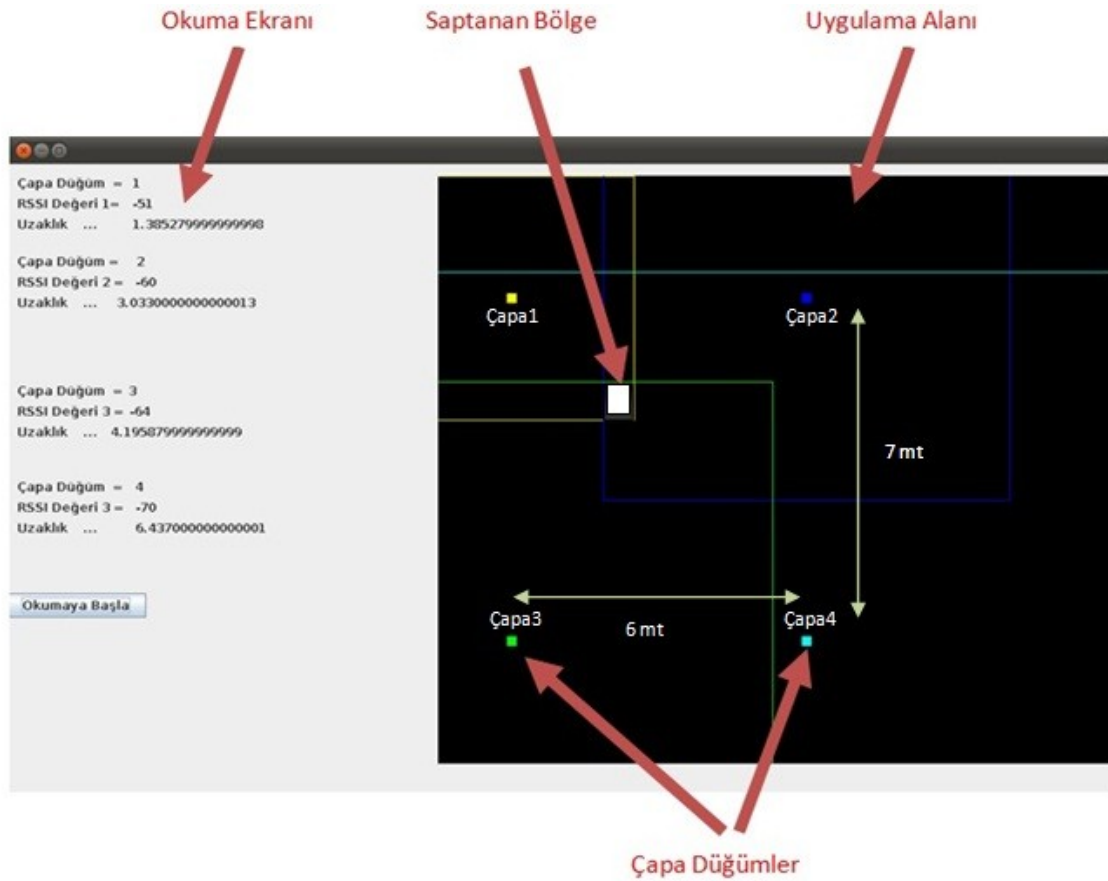
Gerçekleştirilecek olan ilk uygulama, bu tez çalışmasında da önerilen, kablosuz sensör ağ sistemindeki konum saptama yöntemlerinden olan RSSI değeriyle mesafe tahmini ve “Bounding Box” yöntemi ile konum tespiti çalışması olacaktır. Çalışma mantığı her noktada aynı olduğundan söz konusu pratik gerçekleştirme, çalışma alanının bir bölümü için yapılmıştır.

Öncelikle bilinmesi gereken nokta, bu tekniğin her zaman bir hata miktarına sahip olduğudur. Çünkü radyo sinyalinin gücü çevresel etkenler, anten tipi, sıcaklık gibi bir çok faktörden etkilenir. Ayrıca; çok yollu girişim, yansıma ve sönmüleme etkileri de özellikle kapalı alanlarda, sinyal bazlı lokalizasyon sistemlerinde önemli hataların sebebidir [57]. Bu nedenledir ki yapılan uygulamada bu hata miktarını en aza indirmek için öncelikle kullanılacak düğümün RSSI-Mesafe kalibrasyonu yapılmıştır. Uygulamalarda kullanılan TelosB düğümler F tipi gömülü anten yapısına sahiptir [43]. Aynı zamanda yapılan ön deneylerde, birbirine yakın mesafelerde, farklı sinyal güç seviyeleri ölçülmüştür. Bu da radyo sinyal güç dağılımının ortamda homojen olmadığını göstermektedir. Düğümün bulunduğu konum, duruş pozisyonu, yerden yüksekliği doğrudan alınan RSSI değerini etkilemektedir [57]. Böyle bir durumda hata oranının azaltılması için çalışılan ortama göre kullanılacak olan düğümlerin kalibrasyonun yapılıp, RSSI değeri ile uzaklık arasındaki ilişkiyi verecek uygun matematiksel ilişkinin çıkarılması çok önemlidir.

Gerçekleştirilen RSSI ile mesafe tahmini uygulamasında, kullanılan düğümlerin kalibrasyonu için üç farklı alanda iki farklı kalibrasyon tekniği uygulanmıştır [54-57]. Bunlardan ilki, açık alanda yatay bir doğru üzerinde belirli mesafelerde yapılan ölçümlerle uzaklık – RSSI arasındaki matematiksel ifadenin çıkarılması, ikincisi, bina içerisinde yatay bir doğru üzerinde belirli mesafelerde alınan RSSI değerleri ile matematiksel ifadenin elde edilmesi, üçüncüsü ise bina içindeki çalışma ortamında belirli açılarda ve mesafelerde alınan RSSI değerlerinin aritmetik ortalamalarına bağlı olarak ilgili matematiksel ifadenin elde edilmesidir.

RSSI-Mesafe kalibrasyon denemelerinde, belirli zaman aralığı ile sinyal gönderen bir çapa düğüm, çapa düğümünden aldığı sinyalin RSSI değerini ölçen bir hedef (kör) düğüm, USB portuna bağlı toplayıcı düğüm bulunan dizüstü bilgisayar kullanılmıştır. Bu bilgisayarda aynı zamanda toplayıcı düğümünden gelen verileri işleyen ve gerekli hesapları yaparak görsellik sağlayan bir java yazılımı geliştirilmiştir. Bu yazılım ile hem kalibrasyon uygulamaları esnasında gelen RSSI okumaları alınmış hem de elde edilen denklemlere göre, yapılan deneylerin görselleştirilmesi sağlanmıştır.

Konum tespiti / izleme ve görselleştirme işlemi için Bounding Box yönteminden faydalanılmıştır. Şekil 4.11’de geliştirilen java programının ekran görüntüsünü göstermektedir.



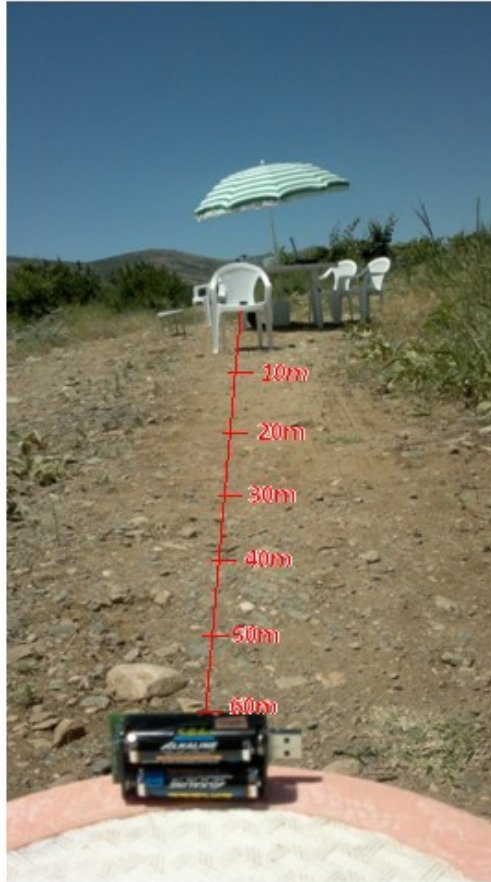
Şekil 4.11. Java programının ekran görüntüsü

Ekranın sol tarafında, her bir çapa düğümünden ölçülen RSSI değerleri ve bu değerlere göre hesaplanmış uzaklık mesafeleri gösterilir. Ekranın sağ tarafında ise hesaplanan değerlere göre elde edilen ekran görüntüsünü göstermektedir. Bu ekran görüntüsünde sarı, mavi, yeşil, açık mavi renkli küçük kareler sırasıyla çapa1, çapa2, çapa3 ve çapa4’ü temsil

etmektedir. Çalışma alanının boyutu 6 x 7 mt yani 600 x 700 cm'dir. Bu ekranda ise çapa düğümler arasında yatayda 300 piksel düşeyde 350 piksel mesafe bulunmaktadır. Diğer bir ifadeyle 1/2 ölçek ile çizim yapılmaktadır. Her bir çapa düğümden okunan mesafe cm'ye dönüştürölüp gerekli yuvarlama işlemi yapıldıktan sonra ölçek uygulanarak çapa düğümün rengiyle aynı olmak koşulu ile "Bounding Box" yönteminde anlatıldığı gibi kapsadığı alan çizilmektedir. Hedef düğümün yerinin saptanabilmesi için en az 3 çapa düğümün kapsayıcı alanlarının kesişmesi gerekmektedir. Daha net sonuç alınması için kapsama çizgilerinin tamamı dışarıda kalan çapa düğümlerin değerleri ihmal edilebilir. Eğer söz konusu kesişme olmazsa hedef nokta doğru şekilde saptanamamış demektir.

4.2.1.1. Açık Alanda Yapılan RSSI- Mesafe Kalibrasyon Deneyleri

RSSI- Mesafe ilişkisini belirlemek için, açık alandaki ölçümler belirli bir yatay ekseninde yapılmıştır. Şekil 4.12'de görüldüğü gibi belirli bir noktada sabit duran çapa düğüm, RSSI değerleri okunan hedef düğüm ve bilgisayara bağlı toplayıcı düğüm kullanılmıştır.



Şekil 4.12. Arazide ölçüm

Karşılaştırma maksadı ile hedef düğüm, hem tabure üzerinde hem de elde taşınmak suretiyle okumalar yapılmıştır. Elde edilen veriler, yazılan bir Java uygulaması ile okunarak kaydedilmiştir. Mesafelere göre okunan RSSI değerleri Tablo 4.4’te verilmiştir.

Tablo 4.4. Açık alan ölçüm sonuçları

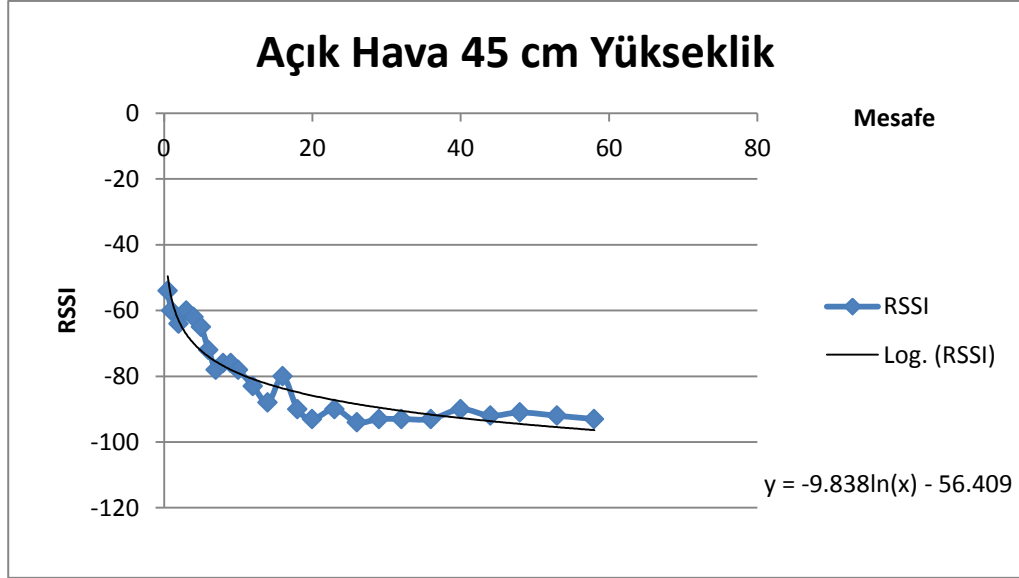
Mesafe (mt)	RSSI (dBm) (45 cm yüksekte)	RSSI (dBm) (Gezgin)
0.5	-54	-54
1	-60	-59
2	-64	-62
3	-60	-59
4	-62	-64
5	-65	-63
6	-72	-60
7	-78	-60
8	-76	-68
9	-76	-66
10	-78	-76
12	-83	-72
14	-88	-76
16	-80	-84
18	-90	-80
20	-93	-89
23	-90	-84
26	-94	-89
29	-93	-90
32	-93	-86
36	-93	-92
40	-90	-93
44	-92	-93
48	-91	-90
53	-92	-93
58	-93	-93

Bu tablodaki veriler ışığında, yerden 45 cm yükseklikte yapılan deneye ait matematiksel ifade, MATLAB, Datafit veya Excel gibi bir bilgisayar programında eğri uydurma yöntemi kullanılarak Denklem 4.3 elde edilmiştir. Ölçüm sonuçları ve uydurulan eğri grafiği ise Şekil 4.13’de verilmiştir.

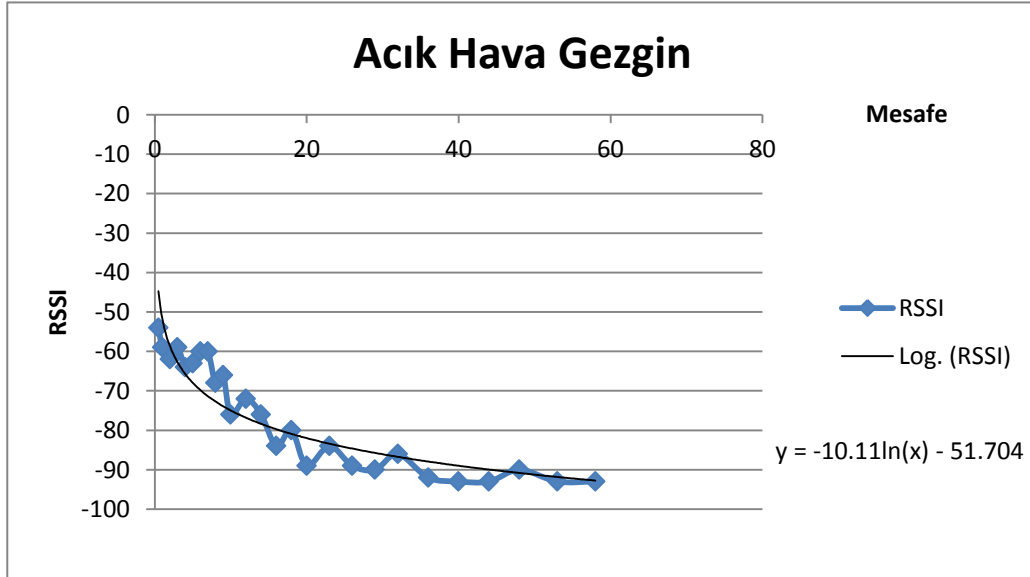
Hedef düğümün elde taşınarak oluşturulan ölçüm sonuçlarına göre, eğri uydurma programından elde edilen matematiksel gösterim Denklem 4.4’de verilmiştir. Burada “D” mesafe, RSSI ise ölçülen sinyal güç seviyesidir. Ölçüm sonuçları ve eğri grafiği Şekil 4.14’te gösterilmiştir.

$$D = e^{-(RSSI+56.409)/9.838} \quad (4.3)$$

$$D = e^{-(RSSI+51.704)/10.11} \quad (4.4)$$



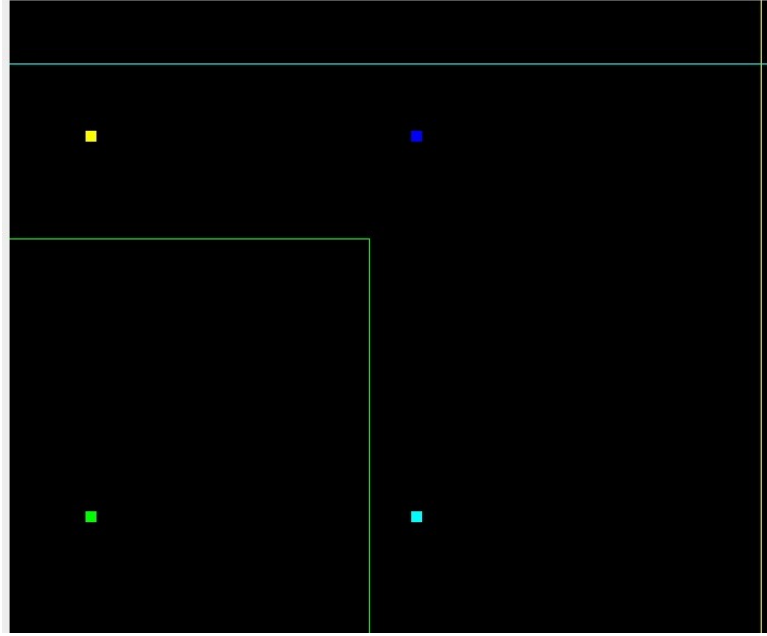
Şekil 4.13. Açık havada tabure üzerinde yapılan ölçüm grafiği



Şekil 4.14. Açık havada elde taşınarak yapılan ölçüm grafiği

Açık havada alınan değerlerden sonra konum tespiti için 7 x 6 m’lik bir alanın köşelerine yerleştirilen çapa düğümlerle konum tespiti deneyleri yapılmıştır. Yapılan çalışma sırasında konum ya büyük bir hatayla bulunmuş ya da hiç tespit edilememiştir. Örneğin Şekil 4.15 javada hazırlanan programın ekran görüntüsünde köşelerde bulunan

sarı, mavi, yeşil ve açık mavi renkli kareler sırasıyla çapa1, çapa2, çapa3 ve çapa4 düğümlerini göstermektedir. Şekilden de görüleceği üzere herhangi bir kesişimin elde edilmemesinden dolayı, çalışma ortamında, merkez noktaya yerleştirilen hedef düğümün konumu tespit edilememiştir.



Şekil 4.15.Açık alanda hatalı saptama

4.2.1.2. Kapalı Alanda Yapılan RSSI – Mesafe Kalibrasyon Deneyleri

Yatay Kalibrasyon

İkinci olarak benzer bir ölçüm ise bina içinde (indoor) belirli bir yatay doğrultuda yapılmıştır. Çapa düğüm belirli bir noktaya yerleştirilerek ve konumu belirlenecek olan kör düğüm ile belirli mesafelere taşınarak RSSI değerleri alınmıştır. Yapılan ölçüm neticesinde elde edilen sonuçlar Tablo 4.5’de gösterilmektedir.

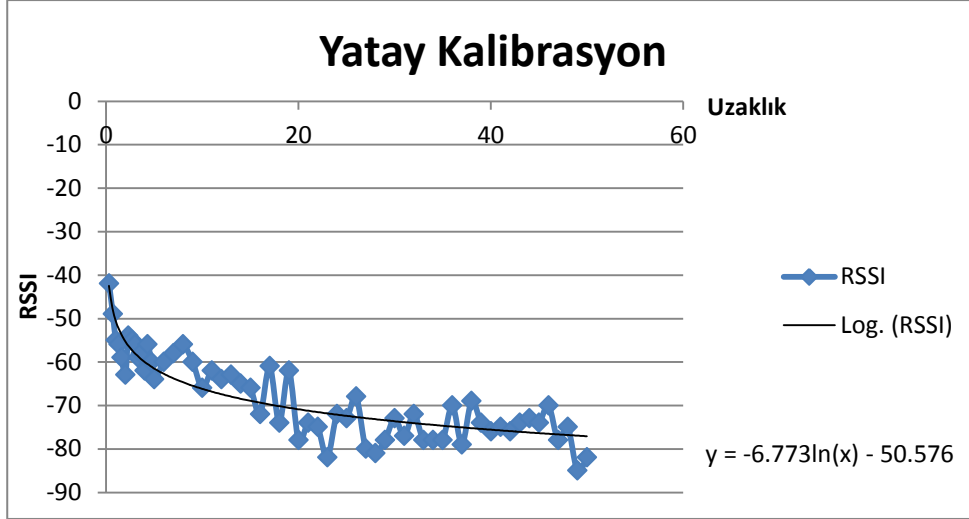
Tablo 4.5. Bina içi yatay ölçüm

Mesafe (mt)	RSSI (dBm)	Mesafe (mt)	RSSI (dBm)	Mesafe (mt)	RSSI (dBm)
0.3	-42	11	-62	31	-77
0.7	-49	12	-64	32	-72
1	-55	13	-63	33	-78
1.3	-56	14	-65	34	-78
1.6	-59	15	-66	35	-78
2	-63	16	-72	36	-70
2.3	-54	17	-61	37	-79
2.7	-55	18	-74	38	-69
3	-56	19	-62	39	-74
3.3	-59	20	-78	40	-76
3.6	-58	21	-74	41	-75
4	-62	22	-75	42	-76
4.3	-56	23	-82	43	-74
4.7	-60	24	-72	44	-73
5	-64	25	-73	45	-74
6	-60	26	-68	46	-70
7	-58	27	-80	47	-78
8	-56	28	-81	48	-75
9	-60	29	-78	49	-85
10	-66	30	-73	50	-82

Elde edilen sonuçlar bilgisayar yardımıyla analiz edilerek Denklem 4.5 ve Denklem 4.6'daki hem üssel hem de ikinci derece denklemler elde edilmiştir. Alınan değerlere göre elde edilen grafik Şekil 4.16'da gösterilmektedir.

$$D = e^{-(RSSI+50.576)/6.773} \quad (4.5)$$

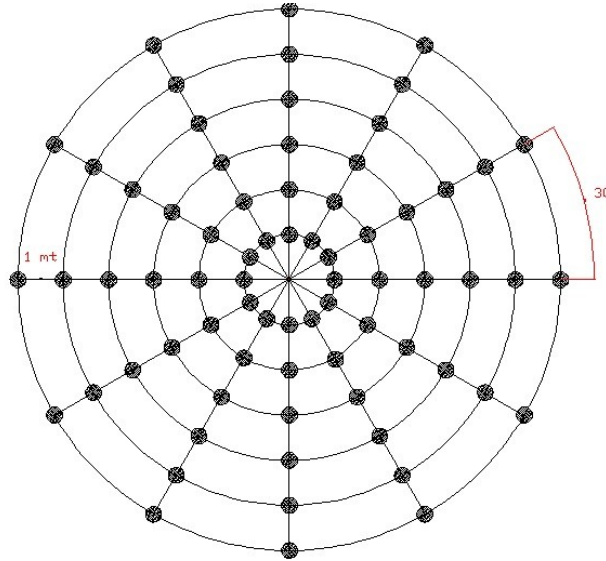
$$D=0,0095*RSSI^2 - 0,1411*RSSI - 32 \quad (4.6)$$



Şekil 4.16.Bina içi yatay ölçüm grafiği

Dairesel Kalibrasyon

Bina içinde yapılan son kalibrasyon tipi ise diğer yönleme göre daha tutarlı sonuçları olan, belirli açı ve mesafelerde okunan RSSI değerleri ile elde edilen matematiksel denkleme bağlı kalibrasyon yöntemidir[54-57]. Bu yöntemde Şekil 4.17’de gösterildiği gibi merkezde bir çapa düğüm bulunmaktadır ve hedef düğüm 30 derecelik açı aralıklarıyla 1’er metrelik mesafeler ile bu çapa düğüm etrafında döndürülerek her bir noktadaki RSSI değeri okunmaktadır. Bu kalibrasyon yapılırken hedef düğümün duruş pozisyonu hep sabit tutulmuştur.



Şekil 4.17.Dairesel Kalibrasyon Uygulama Metodu

Aynı mesafedeki her açı dilimine denk gelen RSSI değerlerinin aritmetik ortalamaları alınarak gerekli veriler elde edilmiştir. İlk 5'er metrelik dilimler dairesel olarak 180 derece 360 derece için alınmış, 6-10 metrelik dilimler ise ölçüm alanı yetersiz geldiğinden 1 metrelik yatay eksen üzerinde alınmıştır. Yapılan ölçüm sonuçları ve elde edilen veriler Tablo 4.6'te verilmiştir.

Tablo 4.6. Dairesel Kalibrasyon

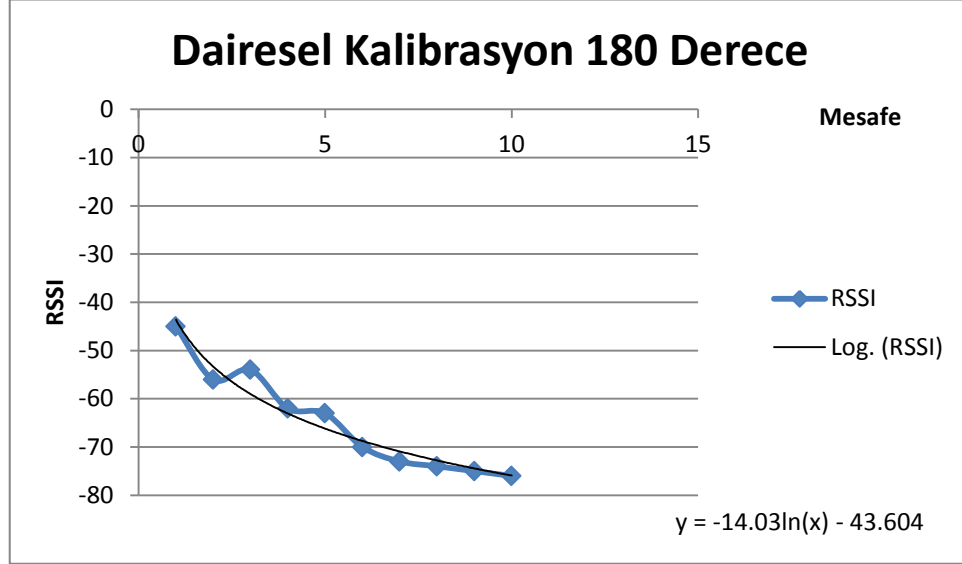
Mesafe	30	60	90	120	150	180	210	240	270	300	330	360	0-180 Drc	0-360 Drc
1	-39	-53	-37	-48	-44	-42	-49	-49	-55	-43	-51	-51	-44.8571	-46.75
2	-59	-43	-60	-58	-58	-59	-55	-51	-54	-58	-57	-53	-55.7143	-55.4167
3	-68	-54	-60	-74	-55	-60	-66	-60	-70	-60	-58	-64	-62.1429	-62.4167
4	-64	-62	-62	-60	-60	-62	-64	-58	-66	-63	-64	-54	-60.5714	-61.5833
5	-60	-66	-60	-62	-63	-65	-62	-68	-65	-66	-66	-64	-62.8571	-63.9167
6	-70	-70	-70	-70	-70	-70	-70	-70	-70	-70	-70	-70	-70	-70
7	-73	-73	-73	-73	-73	-73	-73	-73	-73	-73	-73	-73	-73	-73
8	-74	-74	-74	-74	-74	-74	-74	-74	-74	-74	-74	-74	-74	-74
9	-75	-75	-75	-75	-75	-75	-75	-75	-75	-75	-75	-75	-75	-75
10	-76	-76	-76	-76	-76	-76	-76	-76	-76	-76	-76	-76	-76	-76

Şekil 4.18'de, dairesel kalibrasyonun ve bina içi RSSI uygulamasının yapıldığı ortamı göstermektedir.

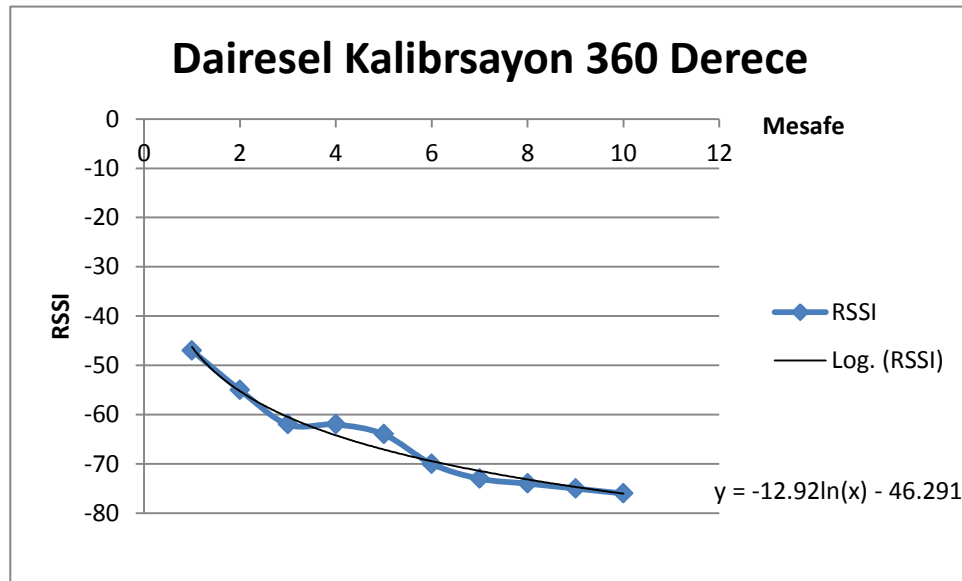


Şekil 4.18. Uygulama ortamı

Daha sonra bu değerlere bağlı olarak matematiksel denklemler çıkarılmıştır. Bu sonuçlara bağlı grafik Şekil 4.19’da, denklemi ise Denklem 4.7 ve 4.8’de gösterilmektedir.



a)



b)

Şekil 4.19.Dairesel kalibrasyona göre uzaklık – RSSI grafiği a) 0-180° için b) 0-360° için

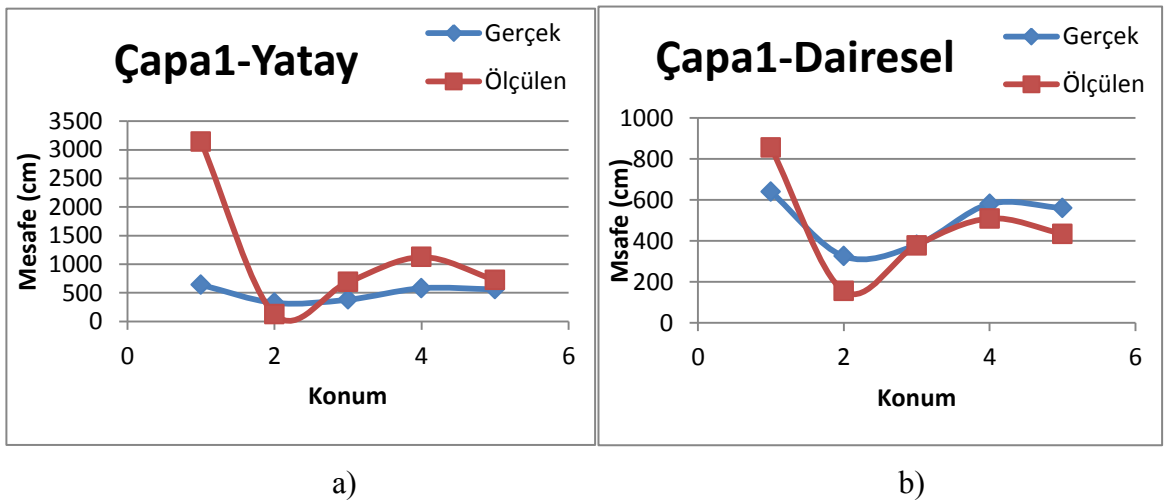
$$D = e^{-(RSSI+43.604)/14.03} \quad (4.7)$$

$$D = e^{-(RSSI+46.291)/12.92} \quad (4.8)$$

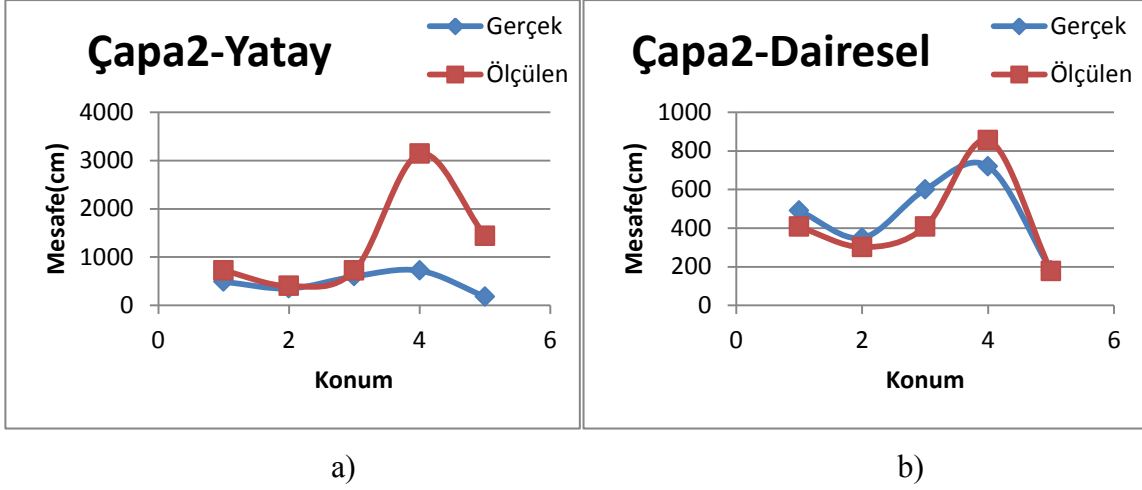
Bina içi yapılan hem yatay kalibrasyon hem de dairesel kalibrasyona ait çalışma denklemleri elde edildikten sonra uygulama safhasına geçilmiştir. Daha öncede belirtildiği üzere, çalışma ortamı, 6 x 7' mt lik dikdörtgensel bir alandır. Alanın her bir köşesine çapa düğümler yerleştirilmiş ve hedef düğüm farklı noktalara yerleştirilerek “Bounding Box” yöntemine göre konum tespiti yapılmıştır. Bu işlem yapılırken, hem yatay hem de dairesel kalibrasyonlar sonucunda elde edilen denklemler kullanılmış ve çıkan sonuçlar karşılaştırılmıştır. Gerçek uzaklıklar ölçülerek, elde edilen sonuçlarla karşılaştırılmıştır. Dairesel ve yatay kalibrasyon yapılmış denklemlere göre elde edilen ölçüm değerleri Tablo 4.7’de verilmiştir. Tabloda belirtilen uzunluklar cm cinsindendir ve her konumda okunan RSSI değerleri, konumlara ait gerçek değerleri ve ölçülen değerleri gösterilmektedir. Gerçek değerlere göre her çapa düğüm için yaklaşım grafikleri Şekil 4.20, Şekil 4.21, Şekil 4.22 ve Şekil 4.23’de gösterilmiştir.

Tablo 4.7 Yatay ve dairesel kalibrasyon sonuçları ile gerçek değerleri gösteren tablo

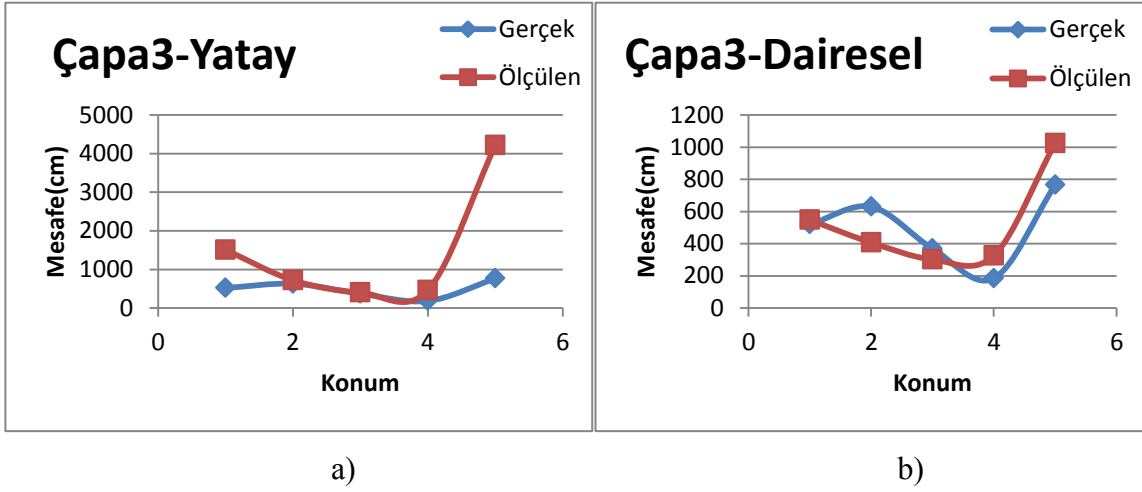
Konum	Çapa1						Çapa2						Çapa3						Çapa4					
	Yatay			Dairesel			Yatay			Dairesel			Yatay			Dairesel			Yatay			Dairesel		
	grç	ölç	rsi	grç	ölç	rsi	grç	ölç	rsi	grç	ölç	rsi	grç	ölç	rsi	grç	ölç	rsi	grç	ölç	rsi	grç	ölç	rsi
1	640	3140	-74	640	856	-74	490	723	-64	490	407	-64	520	1507	-69	520	548	-69	280	402	-60	280	302	-60
2	325	124	-51	325	155	-51	350	402	-60	350	302	-60	630	720	-64	630	407	-64	680	1745	-70	680	636	-70
3	380	688	-63	380	378	-63	600	723	-64	600	407	-64	370	402	-60	370	302	-60	600	1780	-71	600	636	-71
4	580	1124	-67	580	509	-67	720	3140	-74	720	856	-74	185	465	-61	185	326	-61	470	723	-64	470	407	-64
5	560	723	-64	560	434	-64	180	1439	-53	180	177	-53	765	4221	-78	765	1025	-78	530	1745	-70	530	651	-70



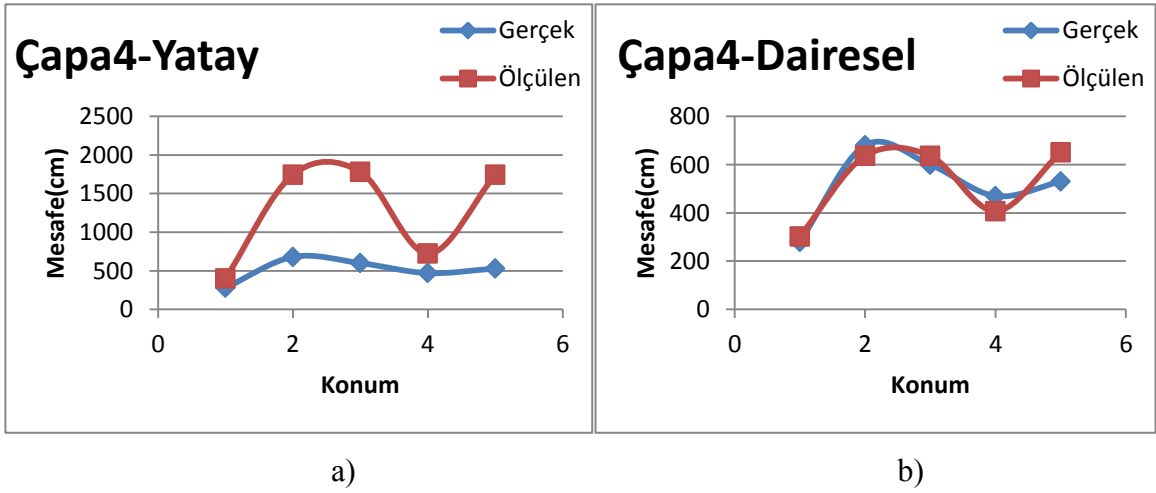
Şekil 4.20. a) Çapa1 Yatay kalibrasyon sapma b) Çapa1 Dairesel kalibrasyon sapma



Şekil 4.21. a) Çapa2 Yatay kalibrasyon sapma b) Çapa2 Dairesel kalibrasyon sapma



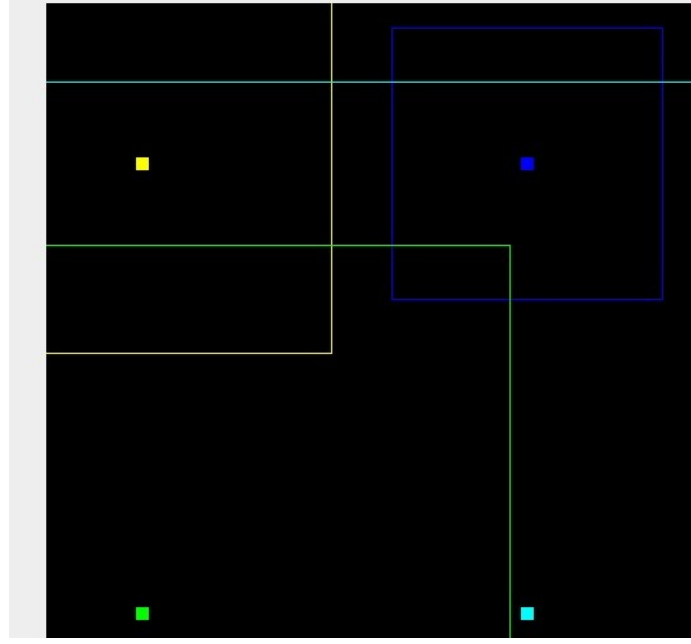
Şekil 4.22. a) Çapa3 Yatay kalibrasyon sapma b) Çapa3 Dairesel kalibrasyon sapma



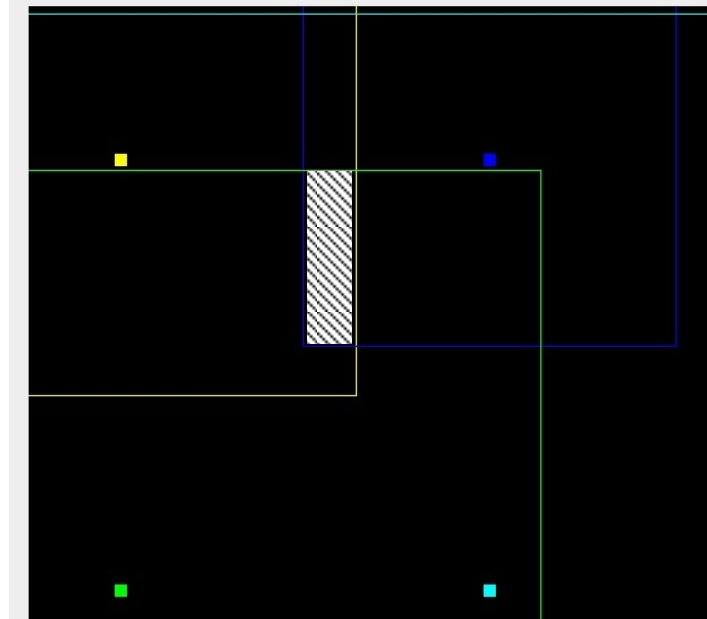
Şekil 4.23. a) Çapa4 Yatay kalibrasyon sapma b) Çapa4 Dairesel kalibrasyon sapma

4.2.1.3. Ofsetli Dairesel RSSI-Mesafe Kalibrasyonu

Yukarıdaki grafiksel gösterimlerden de anlaşılacağı üzere dairesel kalibrasyon konum tespitinde daha başarılı olmuştur. Ancak her ne kadar alınan değerler gerçek değerlere yakın olsa da birkaç konumda, dairesel kalibrasyondan alınan uzaklık bilgileri ile çizilen kapsama çizgileri, konum saptamayı kesin olarak belirleyecek kadar kesişmemiştir. Söz konusu problemin üstesinden gelebilmek amacıyla bu tez çalışmasında “ofset değer” adı verilen bir değeri belirli ölçümler sonucunda bulunan bir tolerans değerinin kullanılması önerilmiştir. Bu durumda dairesel kalibrasyona ait sonuçları tekrar analiz etmek için ikinci bir deney gerçekleştirilmiştir. Bu deney sonucunda gerçek değerler ile ölçülen değerler arasındaki farklar incelenmiştir. Eksik gelen ölçüm değerleri eksi (-), fazla gelen ölçüm değerleri artı (+) işaretlenerek yapılan toplama sonucunun ortalamasının sıfırdan büyük veya küçük olup olmama durumuna bakılmıştır. Eğer bu toplam ortalaması eksi (-) çıkarsa, bu yapılan ölçümlerin genel olarak gerçek değerlerden kısa olduğu, artı (+) çıkarsa yapılan ölçümlerin gerçek ölçümlerden daha fazla olduğu ifade edecektir. Sonucun eksi (-) çıkması, bazı konumlarda yeterli kesişmeye engel olacağı gibi, artı (+) çıkması bazı konumlarda saptanan bölgenin çok daha geniş bir alan olacağını anlamına gelecektir. Elde edilen toplamın durumuna göre seçilen bir “*ofset*” değeri ölçülen değerlere eklenmek veya çıkarılmak suretiyle söz konusu kapsama çizgilerinin kesişimi arttırılıp azaltılabilir. Yapılan bu işlem, tamamen opsiyonel olup ekranda elde edilecek kesişim alanının ayarlanması için kullanılacaktır. Bu ofset değeri yazılımsal olarak programa ilave edilerek çalışma zamanında aktif veya pasif yapılabilir. Örnek olarak Şekil 4.24’te her ne kadar ölçüm değerleri tutarlı olsa da kısa ölçüm neticesinde kapsama çizgileri kesişmemiştir. Tercih edilen ofset değerinin eklenmesiyle Şekil 4.25’deki ekran görüntüsü elde edilecektir ki buda hedef düğümün konumunun bulunabilmesine yardımcı olacak kesişim alanını verecektir. Kesişimin daha iyi fark edilmesi için söz konusu alan taralı olarak gösterilmiştir. Kapsama çizgilerinden her hangi birsinin tamamı çalışma bölgesinin dışına çıkarsa bu çizgi ihmal edilir.

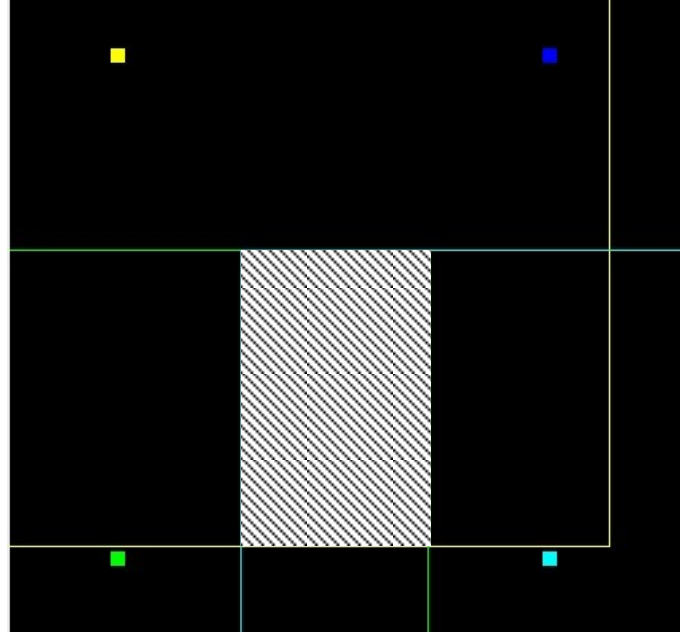


Şekil 4.24. Kısa ölçümden dolayı kesişimin sağlanamaması durumu

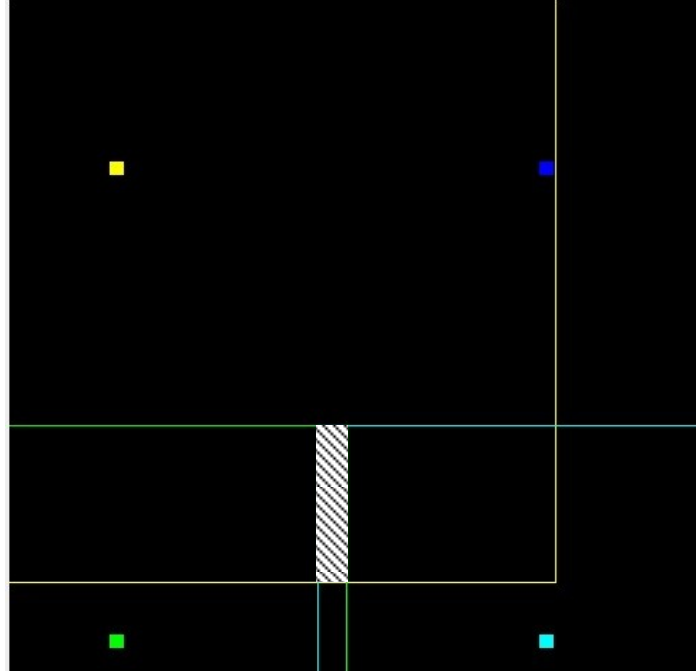


Şekil 4.25. Ofset değerin eklenmesi ile kesişimin elde edilmesi

Her durumda ofset değerinin kullanılmasına gerek olmayabilir. Örneğin Şekil 4.26 ofset değeri kullanılarak elde edilen ekran görüntüsünde kesişim alanı iyi olduğundan Şekil 4.27’de olduğu gibi ofset değeri sıfır alınarak kesişim alanı küçültülebilir. Buda daha iyi bir konum tespiti sağlar.



Şekil 4.26.Yeterli kesişimin olması durumu



Şekil 4.27.Ofset değeri çıkarılması durumunda kesişim alanının küçültülmesi

Belirlenecek ofset değerinin makul bir değer olması önemlidir. Bu tez çalışmasında ofset değerinin seçilmesi için ayrı bir değer okuma testi yapılmıştır. Ofset değerinin seçilmesi için yapılan 8 ayrı ölçüm test değerleri ve fark tablosu Tablo 4.8’de gösterilmiştir.

Tablo 4.8. Ofset durumunu belirlemek için yapılan ölçüm sonuçları

	Çaps1		Çaps2		Çaps3		Çaps4		Fark1	Fark2	Fark3	Fark4	Top. Fark
	Ölçülen	Gerçek	Ölçülen	Gerçek	Ölçülen	Gerçek	Ölçülen	Gerçek					
Ölçüm1	439	570	407	720	351	190	439	480	-131	-313	161	-41	-324
Ölçüm2	636	460	685	613	261	312	636	490	176	72	-51	146	343
Ölçüm3	378	470	193	470	795	450	548	440	-92	-277	345	108	84
Ölçüm4	472	640	378	477	472	520	225	270	-168	-99	-48	-45	-360
Ölçüm5	509	670	144	340	472	686	352	350	-161	-196	-214	2	-569
Ölçüm6	225	330	193	340	440	620	685	630	-105	-147	-180	55	-377
Ölçüm7	133	210	509	490	1200	700	548	550	-77	19	500	-2	440
Ölçüm8	378	700	993	720	407	300	194	300	-322	273	107	-106	-48
												Ortalama Fark	-101.375

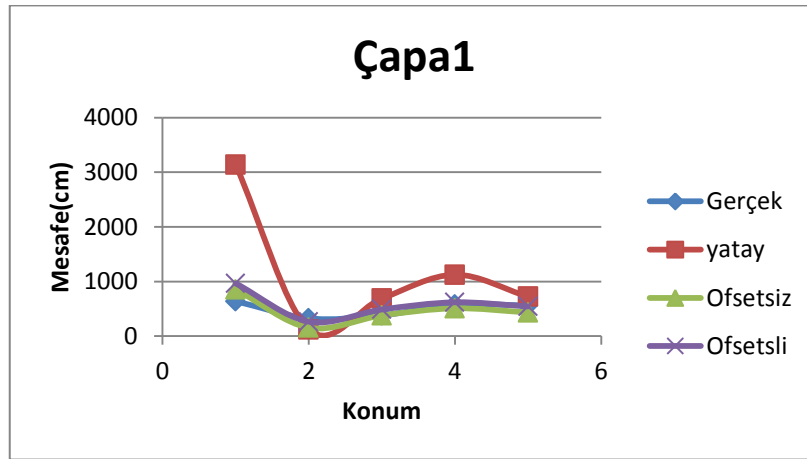
Tablo 4.9.Genel ölçüm sonuçları

-	Ç1						Ç2						Ç3						Ç4																	
Konum	Yaay	Dairesel ofsetsiz			Dairesel ofsetli			Yaay	Dairesel ofsetsiz			Dairesel ofsetli			Yaay	Dairesel ofsetsiz			Dairesel ofsetli			Yaay	Dairesel ofsetsiz			Dairesel ofsetli										
	ölç	rssı	grç	ölç	rssı	grç	ölç	rssı	grç	ölç	rssı	grç	ölç	rssı	grç	ölç	rssı	grç	ölç	rssı	grç	ölç	rssı	grç	ölç	rssı	grç	ölç	rssı	grç						
1	640	3140	-74	640	856	-74	640	966	-74	490	723	-64	490	407	-64	490	517	-64	520	1507	-69	520	548	-69	520	658	-69	280	402	-60	280	302	-60	280	412	-60
2	325	124	-51	325	155	-51	325	265	-51	350	402	-60	350	302	-60	350	412	-60	630	720	-64	630	407	-64	630	517	-64	680	1745	-70	680	656	-70	680	746	-70
3	380	688	-63	380	378	-63	380	488	-63	600	723	-64	600	407	-64	600	517	-64	370	402	-60	370	302	-60	370	412	-60	600	1780	-71	600	656	-71	600	746	-71
4	580	1124	-67	580	509	-67	580	619	-67	720	3140	-74	720	856	-74	720	966	-74	185	465	-61	185	326	-61	185	456	-61	470	723	-64	470	407	-64	470	517	-64
5	560	723	-64	560	434	-64	560	544	-64	180	1439	-53	180	177	-53	180	287	-53	765	4221	-78	765	1025	-78	765	1135	-78	530	1745	-70	530	651	-70	530	761	-70

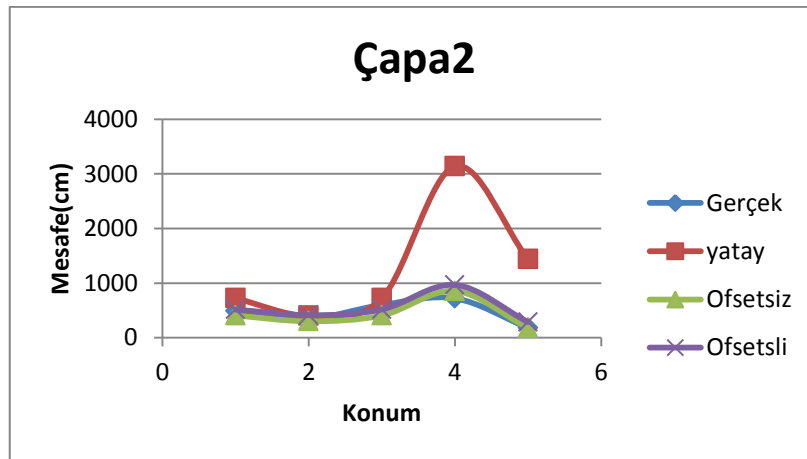
Tablodan da görüleceği üzere yapılan ölçümler ile gerçek değerler arasındaki farkların toplamının ortalaması sıfırdan küçüktür. Bu değerler genel olarak ölçüm değerlerinin gerçek değerlere oranla daha kısa olduğu anlamına gelmektedir. Böylece seçilen ofset değerinin ölçüm sonuçlarına eklenerek çizim yapılması gerektiği ortaya çıkmaktadır. Ofset

değerinin seçimi tercihe bağlı olmakla beraber, burada ortalama fark değerini de içine alacak şekilde 1.1 mt yani 110 cm seçilmiştir.

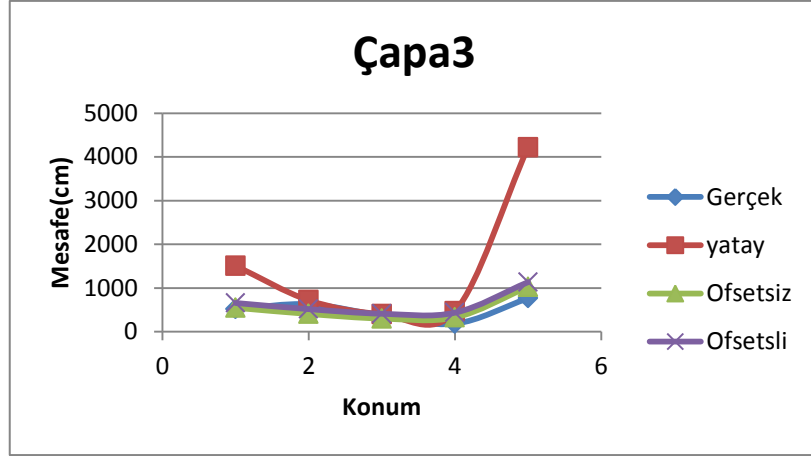
Ofset değeri seçildikten sonra, ölçümler tekrar yapılarak, yatay kalibrasyonlu, dairesel ofsetsiz kalibrasyonlu ve dairesel ofset kalibrasyonlu değerler alınmıştır. Ölçümler sonucu elde edilen değerler tablosu Tablo 4.9 'da verilmiştir. Elde edilen sonuçlara göre üç durum için yaklaşım grafikleri Şekil 4.28, 4.29, 4.30 ve 4.31 'de verilmiştir.



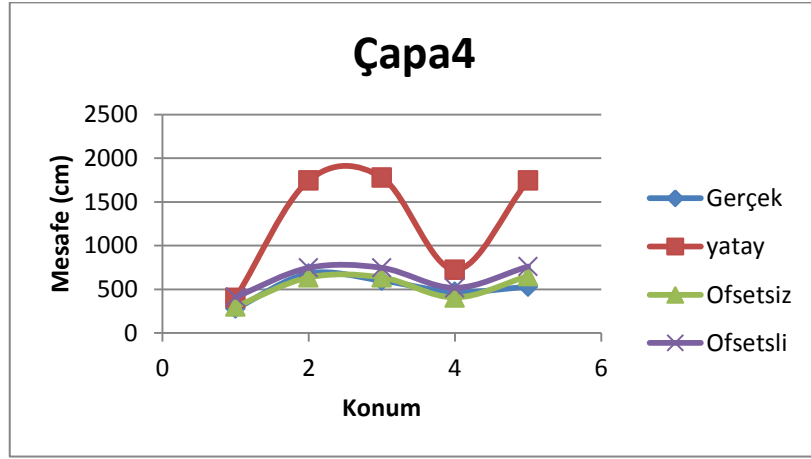
Şekil 4.28.Çapa1 için yaklaşım grafiği



Şekil 4.29.Çapa2 için yaklaşım grafiği

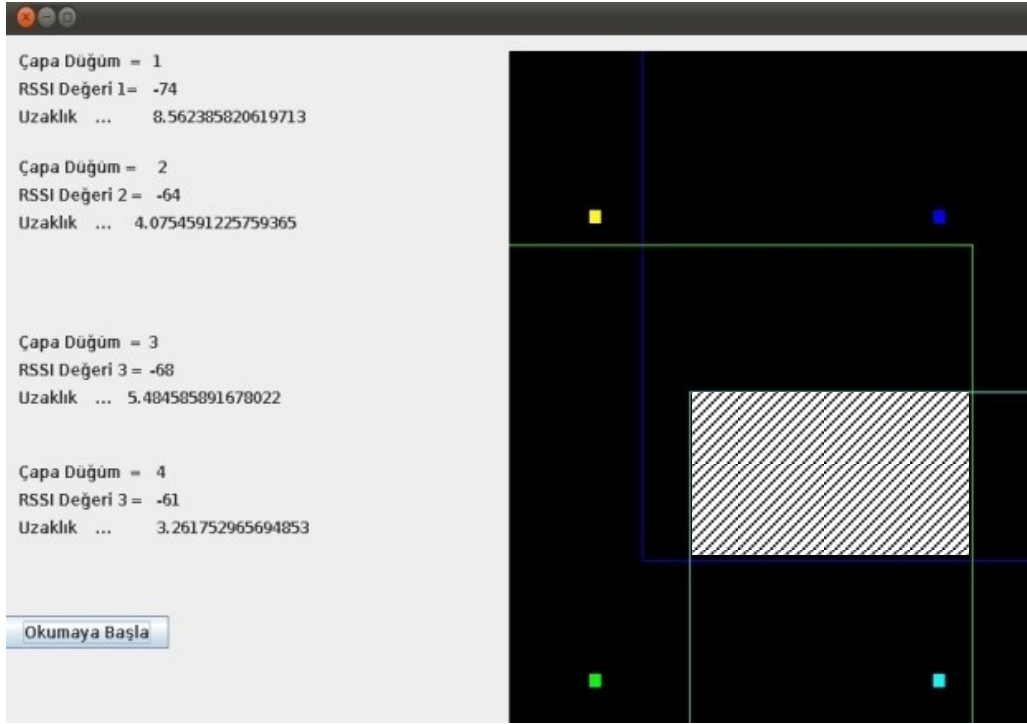


Şekil 4.30.Çapa3 için yaklaşım grafiği

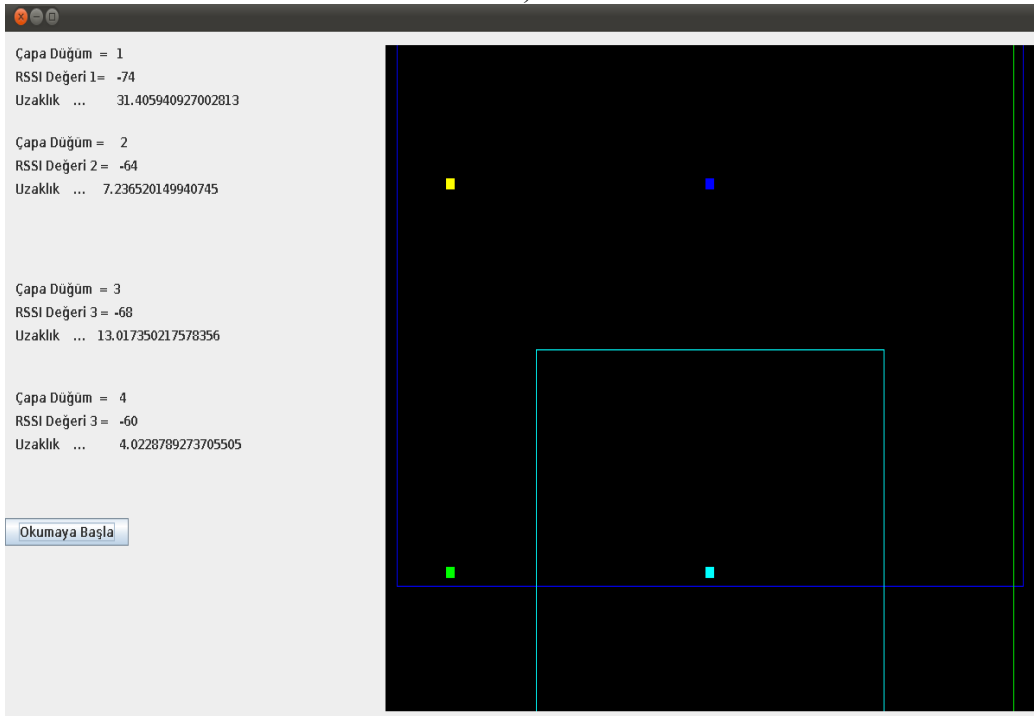


Şekil 4.31.Çapa4 için yaklaşım grafiği

Yapılan denemeler sonucunda ofsetli olsun yada olmasın dairesel kalibrasyon tekniğinin yatay kalibrasyon tekniğine göre çok daha iyi sonuç verdiği tespit edilmiştir. Genel olarak ofset değeri kullanılarak yapılan konum analizleri gerçek değerleri içine aldığı için kesişim bölgeleri daha net görülmüştür. Dairesel kalibrasyon kullanılarak elde edilen uzaklık – RSSI ilişkisini temsil eden denkleme göre yapılan ölçümlerin ekran görüntüleri ile yatay kalibrasyonla elde edilen denkleme göre yapılan ölçümlerin ekran görüntüleri Şekil 4.32a-b, Şekil 4.33 a-b,Şekil 4.34 a -b,Şekil 4.35 a -b’de gösterilmiştir. Görüntülerden de anlaşılacağı üzere yatay kalibrasyon tekniğinde tatmin edici sonuçlar alınamazken dairesel kalibrasyonda belirli bir hata çerçevesinde yaklaşık konum tespiti yapılabilmiştir.

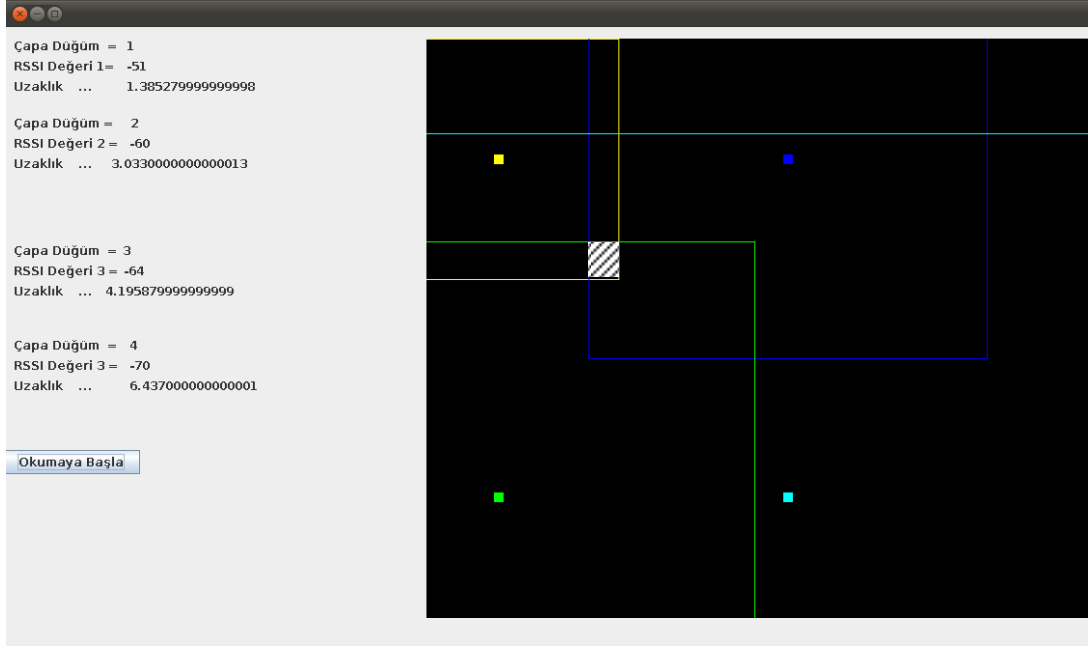


a)

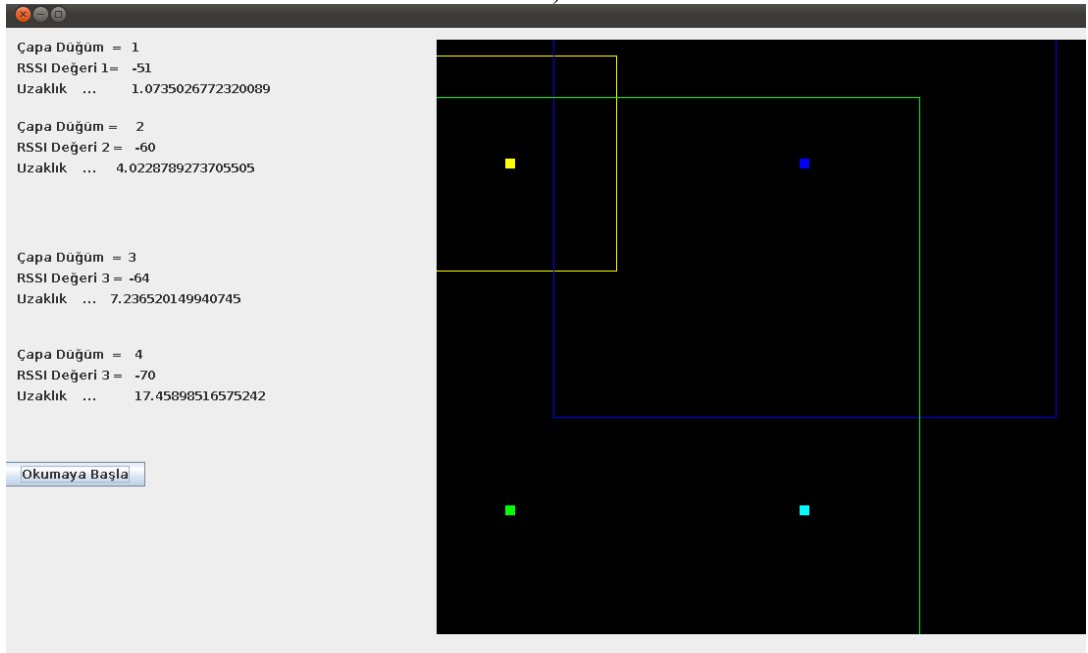


b)

Şekil 4.32. Konum 1 için a) Dairesel Kalibrasyon b) Yatay Kalibrasyon

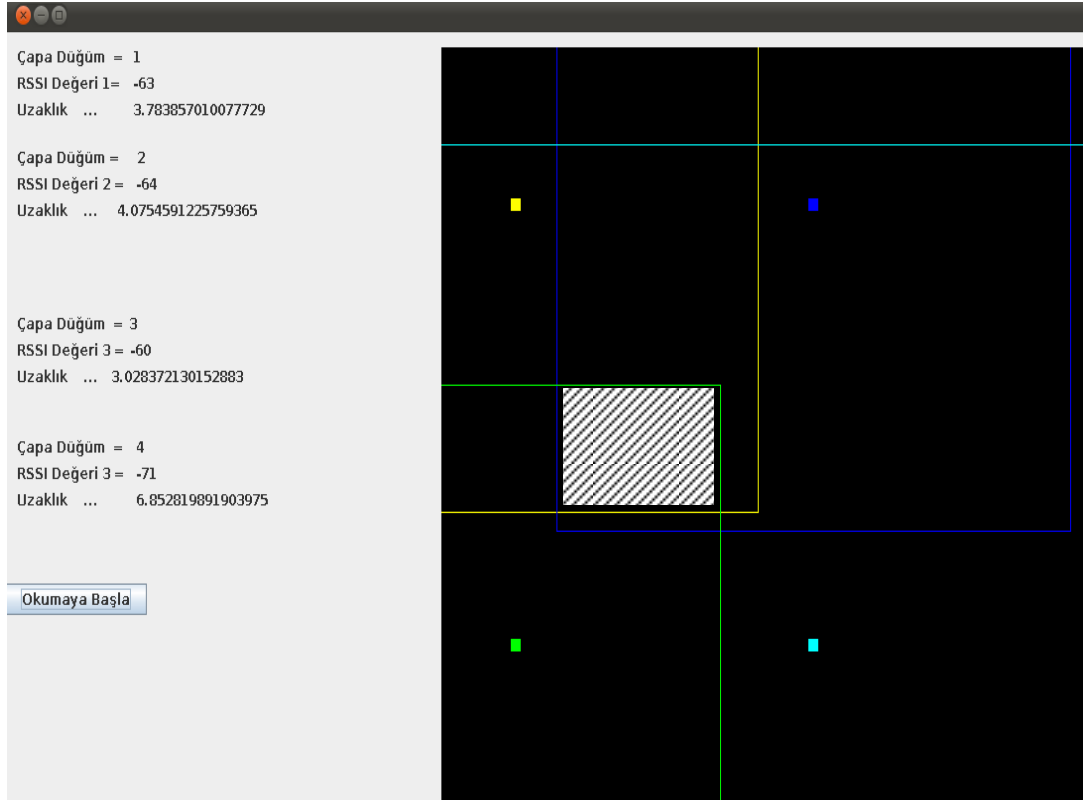


a)

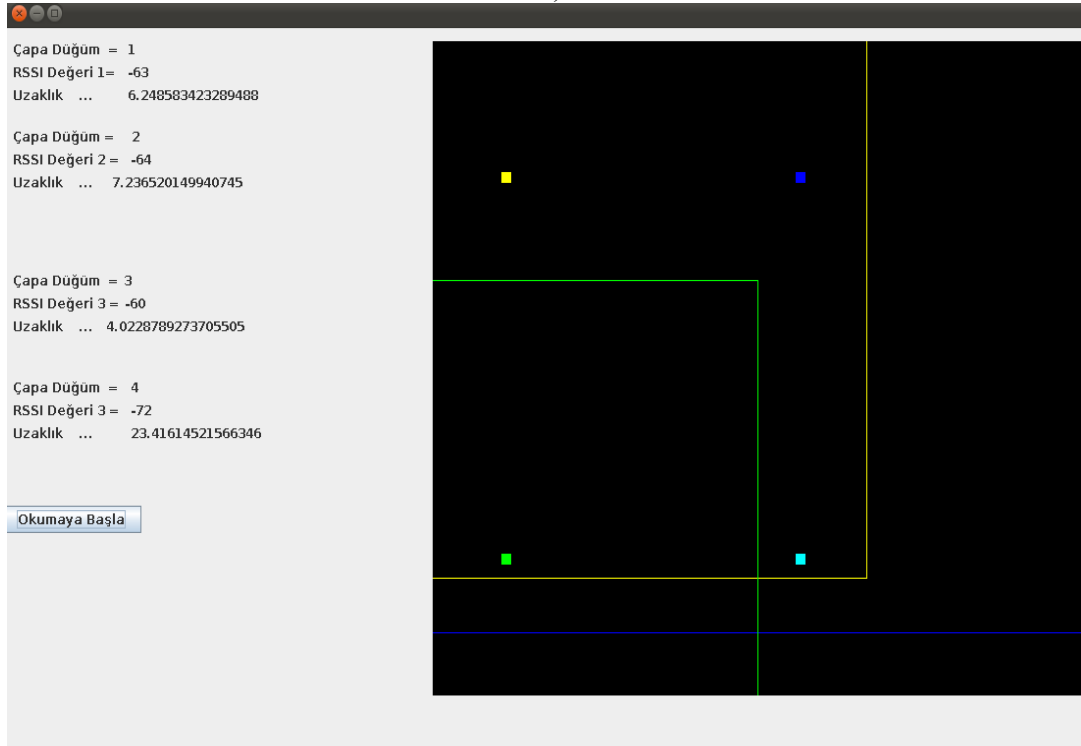


b)

Şekil 4.33. Konum 2 için a) Dairesel Kalibrasyon b) Yatay Kalibrasyon

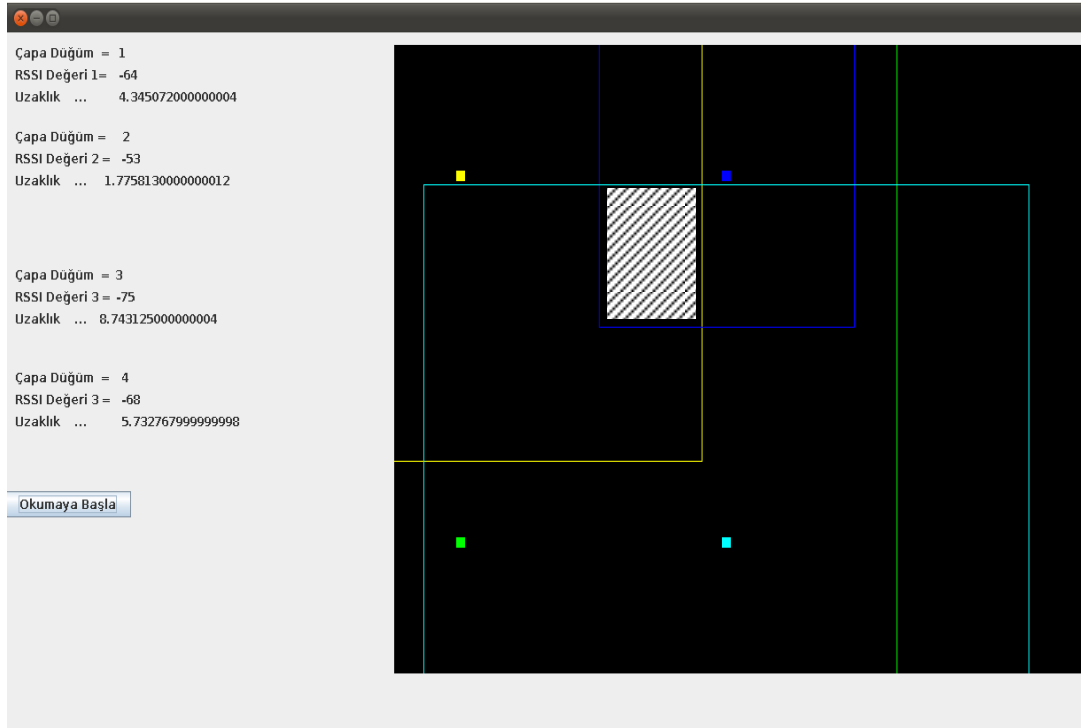


a)

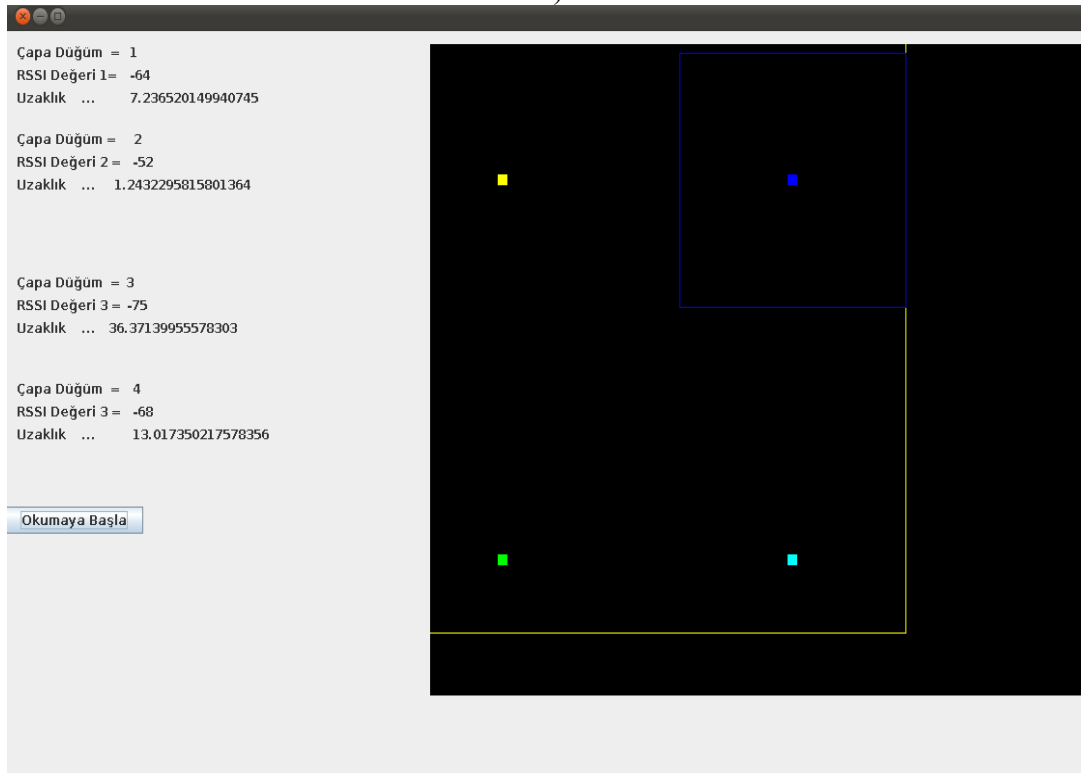


b)

Şekil 4.34. Konum 3 için a) Dairesel Kalibrasyon b) Yatay Kalibrasyon



a)



b)

Şekil 4.35. a) Konum 5 için a) Dairesel Kalibrasyon b) Yatay Kalibrasyon

Dairesel kalibrasyon kullanılarak yapılan yaklaşık konum tespiti denemelerinde genellikle tutarlı sonuç alınmasına rağmen özellikle çalışma alanının kenarları üzerinde ve

kenarlara çok yakın noktalarda istenilen konum tespitleri net olarak yapılamamıştır. Ancak yardımcı çapa düğümler kullanılarak bu sorunun üstesinden gelinebilir.

4.2.2. Kenar Maskeleme Metoduyla Yönlendirme Uygulaması

Bu tez çalışmasında geliştirilen WSN sisteminde, mobil aracın yönlendirilmesi için Bölüm 4.1.1 'de açıklanan “Kenar Maskeleme Yöntemi” kullanılmıştır. Bunun için simülasyonda kullanılan maske, kenar, mobil düğüm kodları gerçek TelosB TPR2420 düğümlere doğrudan yüklenerek bu işlem başarılmıştır. Söz konusu çalışma alanında bulunan düğümlerin çalışma mantıkları birbirleri ile tam bir benzerlik göstermesinden dolayı, istenilen herhangi bir bölge için bu gerçeklemenin yapılması yeterli olacaktır.

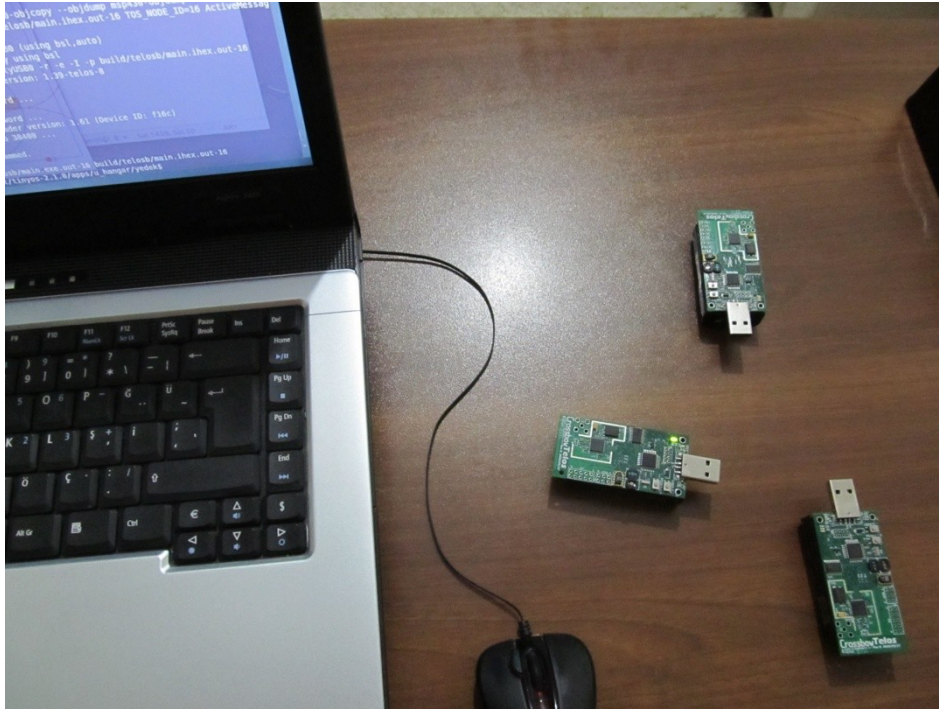
Daha önceden söylendiği gibi karar mekanizması olarak maske düğümler kullanılmaktadır. Örnek olarak, Şekil 4.36'daki kod parçasığı, 15 nolu maske düğümün B25 bölgesi için yaptığı kontrolü göstermektedir.

```
if(maske==15 && kenar==12){//25
    if(h_yatay==5 && h_dusey==2){ |
        tamam=1;
        yukari_cik=0;
        asagi_in=0;
        yatay_git=0;
        yon=0;
        hedef_led=h_dusey;
    }
    if(h_yatay<5){
        yukari_cik=0;
        tamam=0;
        asagi_in=1;
        yatay_git=0;
        hedef_led=h_dusey;
        d=2;
    }
}
```

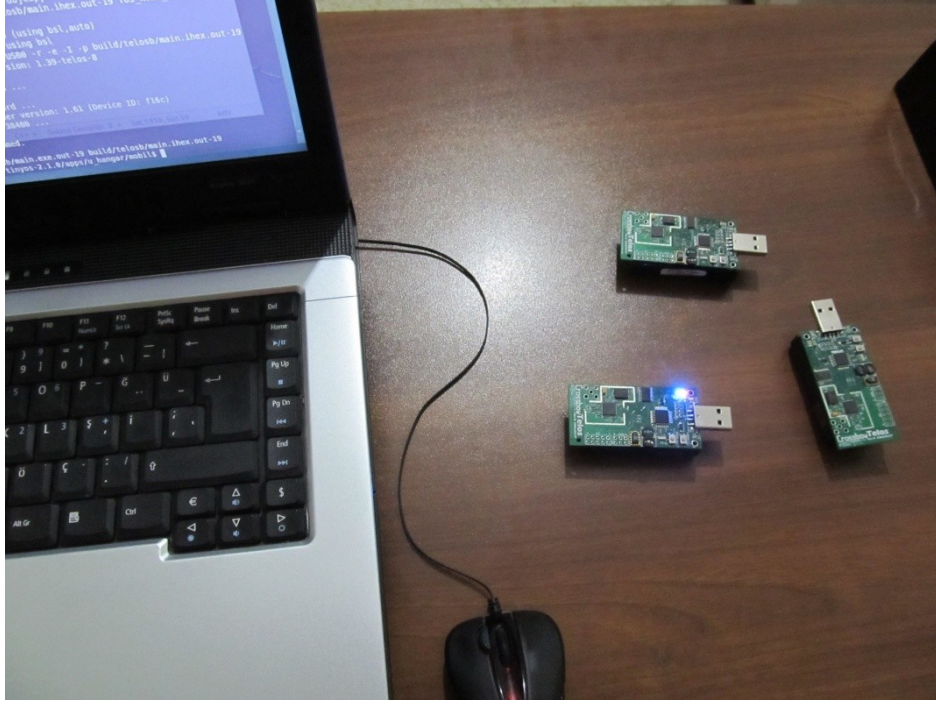
Şekil 4.36. Maske 15 'de B25 bölge kontrolü yapan kod parçası

Düğüm kodlarının çalışıp çalışmadığının anlaşılmasında TelosB düğümler üzerindeki ledler kullanılmıştır. Eğer mobil düğüm “Alt Bölüme İn” uyarısı alırsa tüm ledler sönecek (Şekil 4.39), “Üst Bölüme Çık” bildirimi alırsa tüm ledler yanacak (Şekil 4.38), eğer yatay bölüm seviyesi yakalanmış ise sadece düşey bölge kodunu verecek olan led yanacaktır (Şekil 4.37). Mobil düğüme, merkezden yani “BaseStation” üzerinden veri göndermek için bir java uygulaması yazılmış ve gerçek zamanlı olarak denenmiştir. Uygulamanın

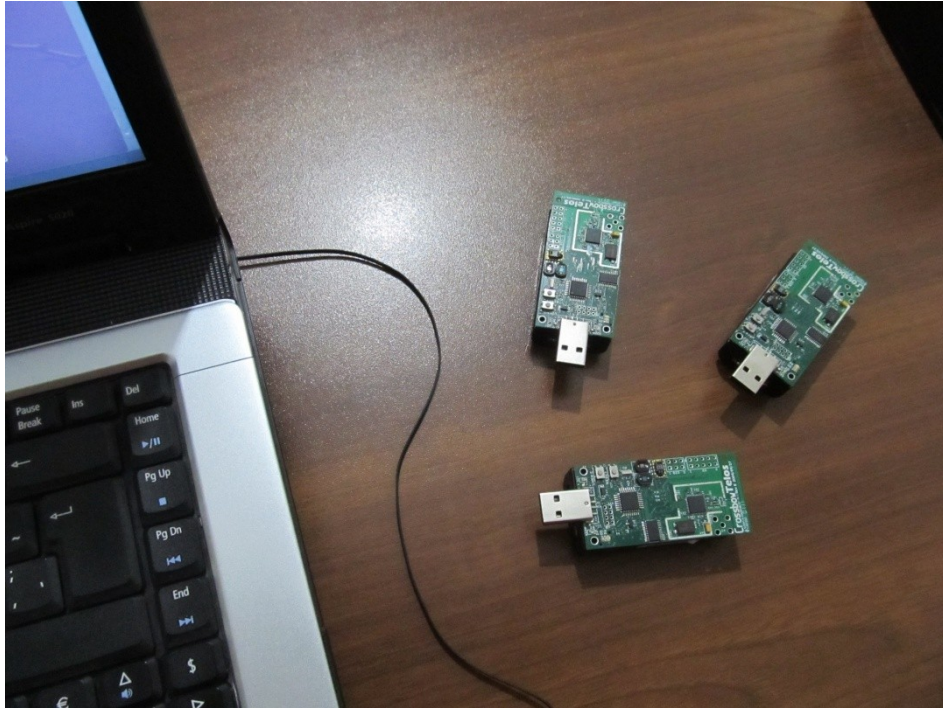
gerçekleştirme bölümünde simülasyonu yapılan kodlar TelosB düğümlere yüklenmiş ve sistemin sorunsuz çalıştığı görülmüştür. Şekil 4.37’de mobil düğümün, 9 nolu maske düğüm ve 11 nolu kenar düğümle etkileştiği B34 bölgesinde bulunma durumu denenmiştir. Bu durumda mobil düğümün hedef bilgisi ise B24 bölgesidir. Yönlendirme bilgisini alan mobil düğüm yatay olarak gidilmesi gerektiğini bildirmek için 2 nolu (orta led) ledi yakmıştır. Benzer şekilde Şekil 4.38’de ki örnek durumda, merkezden B15 bölgesine gitme komutu geldiğinde bulunduğu yerin maske düğümüne hedef yerini bildirmiş ve üst bölgeye geçmesi bilgisini almıştır. Bu durumu ise tüm ledlerini yakarak bildirmiştir. Son olarak merkezden B11 noktasına gitmesi komutu gelmesi üzerine yönlendirme bilgisini ilgili maske düğümle güncelleyen mobil düğüm, alt bölgeye gidilmesi gerektiğini tüm ledlerini söndürerek göstermiştir. Bu duruma ait deneme görüntüsü ise Şekil 4.39’da gösterilmiştir. Merkez düğümden (BaseStation) veri yayınlamak için yazılan Java programının ekran görüntüsü Şekil 4.40ve gerçekleştirme esnasındaki resmi Şekil 4.41’de verilmiştir.



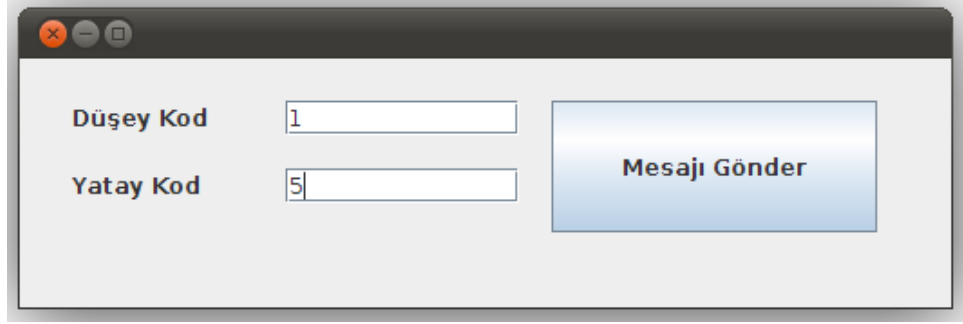
Şekil 4.37. Yatay olarak 2 nolu düşey bölge bildirimi – sadece 2 nolu yeşil led yanıyor



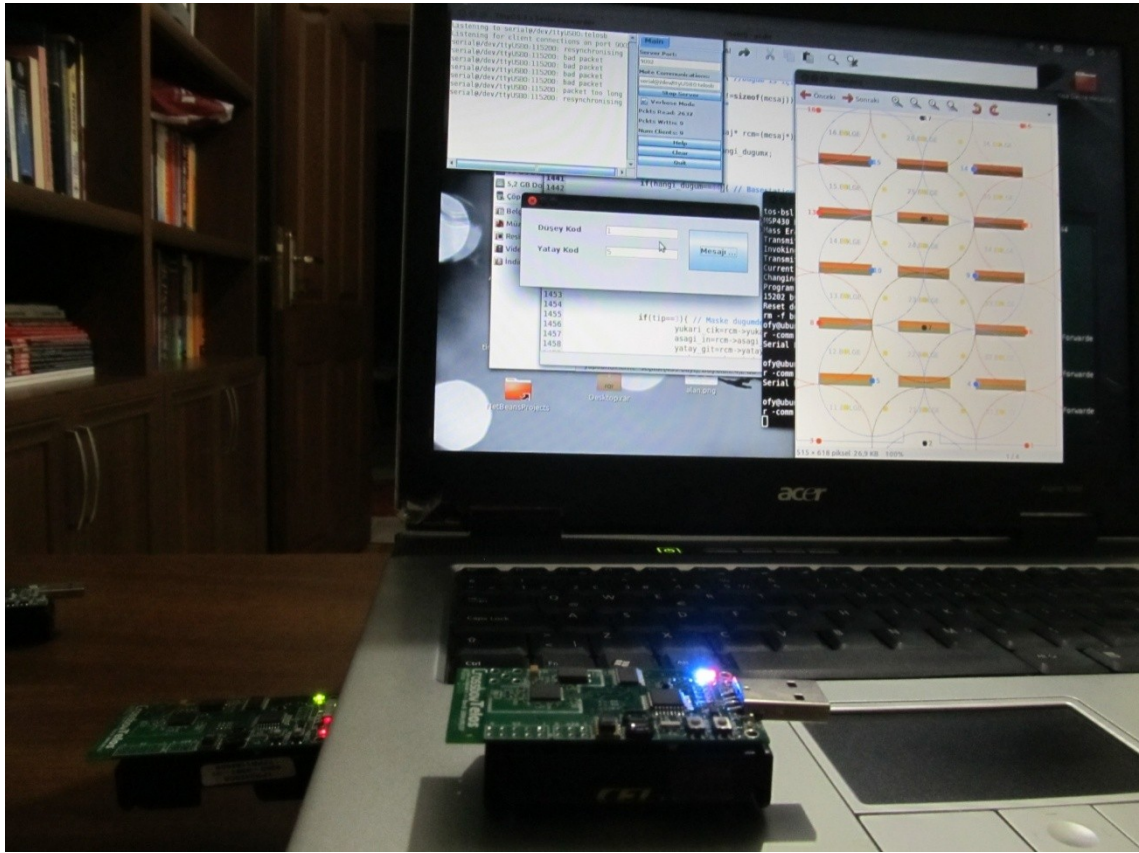
Şekil 4.38.Üst bölgeye geçme bildirimi - tüm ledler yanıyor



Şekil 4.39. Alt bölgeye geçme bildirimi – tüm ledler sönmük



Şekil 4.40. BaseStation üzerinden veri gönderme için yazılan program ekranı



Şekil 4.41. Deneme ortamından bir görüntü

4.2.3. Kapalı Ortamdaki Bazı Fiziksel Büyüklüklerin Okunması

Çalışma alanında bazı bölgelerde ürünlerin dayanım kontrolü için ürünlerin bölgelerin sıcaklık, nem, ışık değerleri veya düğümün voltaj seviye durumu gibi fiziksel özellikler hakkında veri toplanması istenebilir. Bu durumda sensör okumalarının yapılması gerekecektir. Kullanılan TelosB düğümü üzerinde sıcaklık, nem ve ışık sensörleri barındırmaktadır. Düğüm programında bu sensör okumalarını yapmak için yapılandırıcı dosyalarına uygun komponentlerin bağlanması gerekmektedir. Bu komponentlerin bir kısmı “*tos/platforms/telos/chips/*” klasörü altındaki “*s10871*” ve “*sht11*” alt klasörlerinde

bulunduğundan bunların “telosb” klasörüne kopyalanması gerekebilir. Şekil 4.42 a yapılandırıcı ayarlarını gösterirken Şekil 4.42 b sensörün okuma yapma kodunu göstermektedir.

```
configuration OscilloscopeAppC { }
implementation
{
    components OscilloscopeC, MainC, ActiveMessageC, LedsC,
        new TimerMilliC(), new SensirionSht11C() as Sensor,
        new AMSenderC(AM_OSCILLOSCOPE), new AMReceiverC(AM_OSCILLOSCOPE);

    OscilloscopeC.Boot -> MainC;
    OscilloscopeC.RadioControl -> ActiveMessageC;
    OscilloscopeC.AMSend -> AMSenderC;
    OscilloscopeC.Receive -> AMReceiverC;
    OscilloscopeC.Timer -> TimerMilliC;
    OscilloscopeC.Read -> Sensor.Humidity;
    OscilloscopeC.Leds -> LedsC;
}
```

a)

```
event void Read.readDone(error_t result, uint16_t data) {
    if (result != SUCCESS)
    {
        data = 0xffff;
        report_problem();
    }

    local.readings[reading++] = (-0.0000028*data*data)+0.0405*data-4;
}
```

b)

Şekil 4.42. Sensör okumaları için a) Yapılandırıcı ayarları b) Okuma kod satırı

TelosB düğümü üzerindeki sensörlerden yapılan okumalar ham veriler olup bu okumaların sensör kataloglarında verilen formüllerle anlamlı hale getirilmesi gerekmektedir. Denklem 4.9 sıcaklık okuması, Denklem 4.10 lineer nem okuması, Denklem 4.11 sıcaklığa bağlı nem okuması [19] için kullanılan formüllerdir.

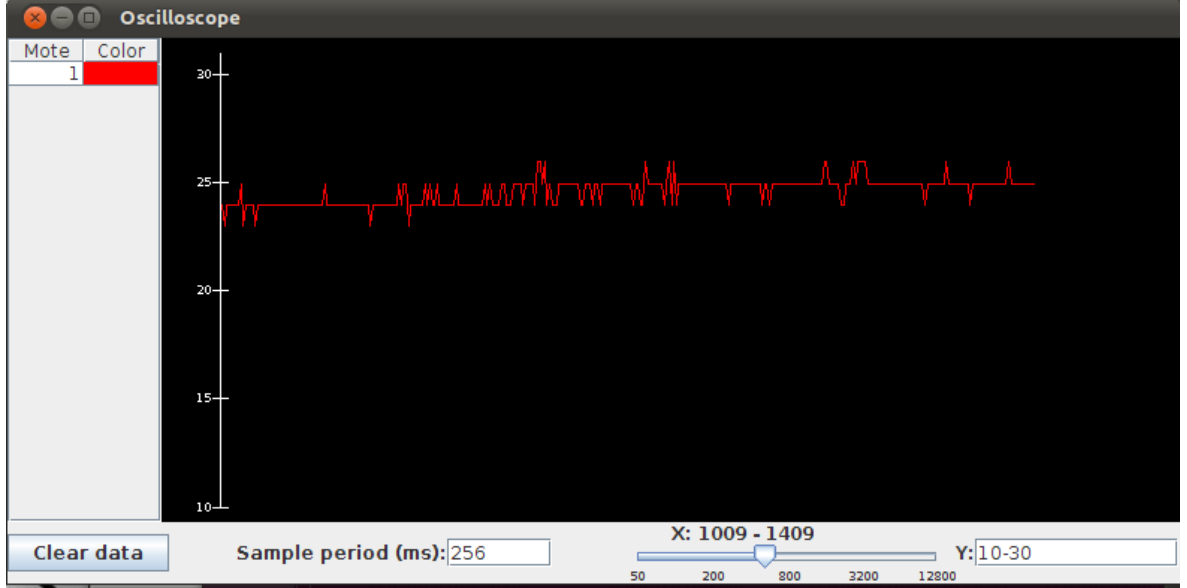
$$Sıcaklık [C] = -38.4 + 0.0098 * veri \quad (4.9)$$

$$Nem [\%] = -0.0000028*veri*veri + 0.0405*veri-4 \quad (4.10)$$

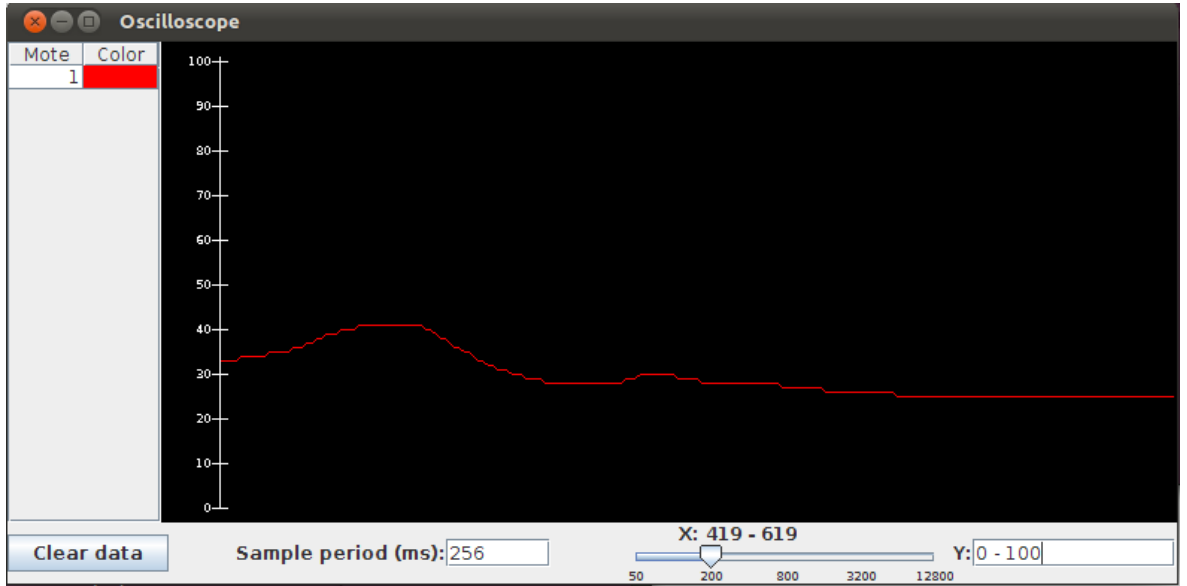
$$Nem(Sıcaklığa bağlı)[\%] = (Sıcaklık-25)*(0.01+0.00008*veri)+Nem \quad (4.11)$$

Yapılan ayarlamal neticesinde TelosB düğümü ile yapılan sıcaklık, nem, ışık ve voltaj seviyelerinin okumalarına ait ekran görüntüleri Şekil 4.43, Şekil 4.44, Şekil 4.45 ve Şekil 4.46’da gösterilmiştir. Sıcaklık okumasında sensör daha sıcak bir ortama taşınarak sıcaklık değişimi, farklı nem değerleri olan ortamlara taşınarak nem değişiminin izlenimi, üzerine

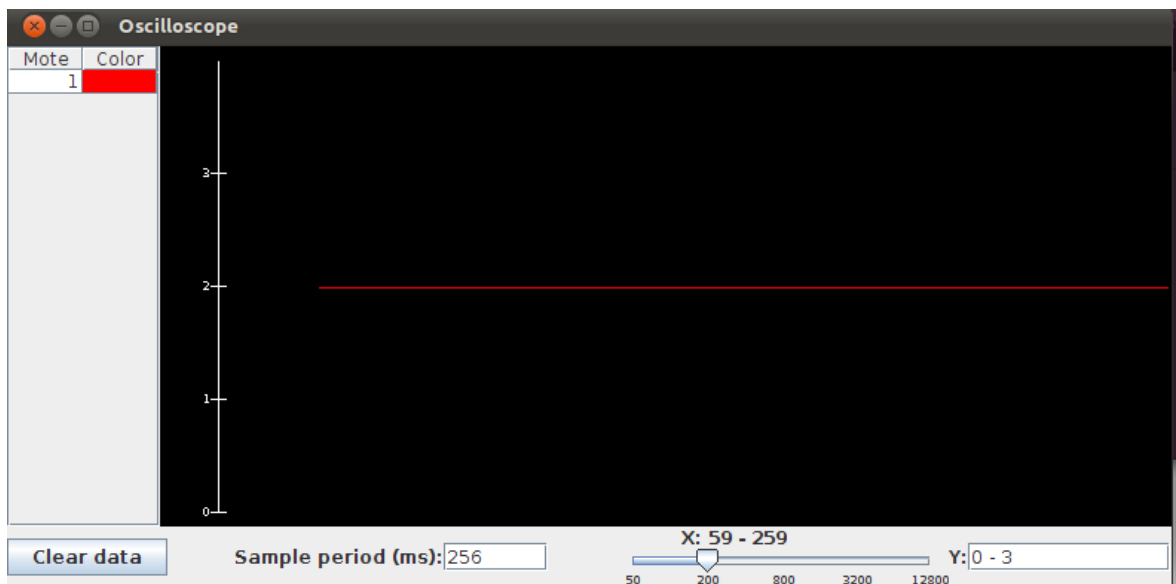
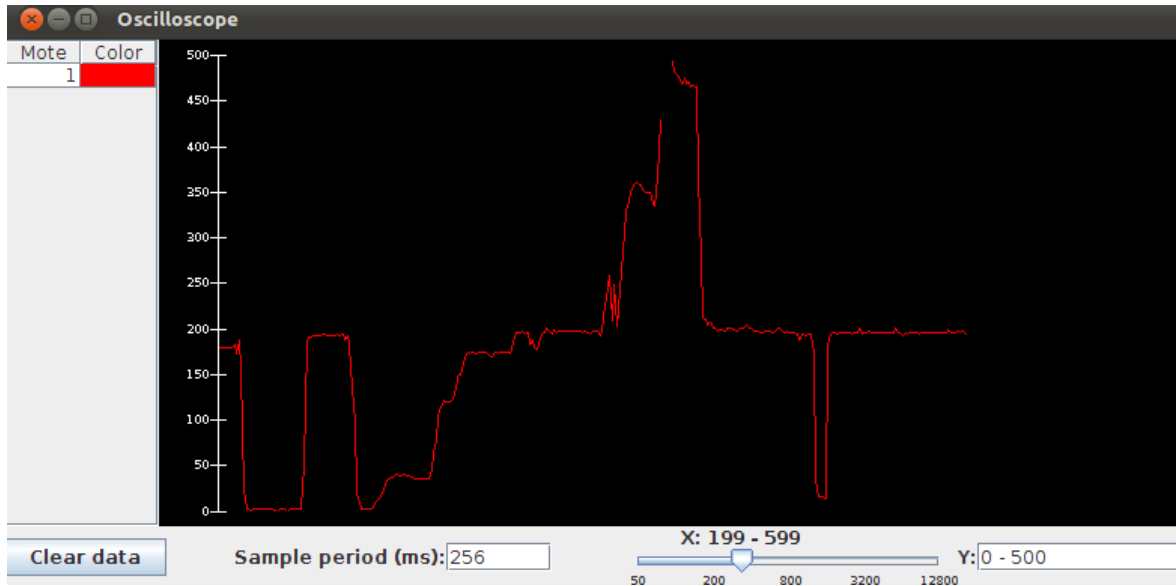
ışık kaynağı tutarak veya el ile ışık sensörünün üzeri tamamen kapatılarak değişimler izlenmiştir.



Şekil 4.43. Sıcaklık bilgisini okunması (x eksen=Zaman, y eksen = $^{\circ}\text{C}$)



Şekil 4.44. Nem bilgisinin okunması (x eksen=Zaman, y eksen = % nem)



Gerçekleştirilen uygulamanın avantajları, kapalı mekanlarda yönlendirme ve izleme yapılabilmesi, araç ve çalışanların en kısa şekilde yeni yerlere gönderilerek işletme kolaylığı sağlaması, araç ve ürün bölgelerine ait istatistiksel verilerin toplanabilmesine olanak sağlaması, insansız iş araçlarının yönlendirilme çalışmalarında denenebilecek olmasıdır. Uygulamanın eksik yönleri ise, sistemin geometriye bağlı olması, küçük alanlarda verimli olmaması, bir hücrede istenilenden çok fazla düğüm sinyali alınması durumunda programlama zorluklarının ortaya çıkmasıdır.

5. SONUÇLAR VE ÖNERİLER

5.1. Sonuçlar

Kablosuz sensör ağlarının yakın gelecekte çok daha popüler olacağı bir gerçektir. Şu an itibari ile bu alanda üzerinde çalışılacak birçok konu bulunmakta ve araştırmacıların yakın ilgisini çekmektedir. Uygulama alanının oldukça geniş olması da bu konuyu dahacazip hale getirilmektedir.

Bu tez çalışmasında gerçekleştirilmek istenen sistemin amacı, kapalı ortamlarda konum tespiti, iş araçlarının yönlendirilebilmesi, işletme şartlarına esneklik ve kolaylık sağlanması, ortam hakkında istatistiksel bilgilerin toplanması gibi özelliklerin kablosuz sensör ağlarıyla gerçekleştirilmesini sağlamaktır. Sistem iki kısımdan meydana gelmektedir; birincisi, iş araçlarının istenilen bölgeye doğru bir şekilde yönlendirilebilir olmalarını sağlamak, ikincisi hem çalışma alanının fiziksel özellikleri hem de araçların durumu hakkında bilgi toplamak ve gerektiğinde yeni yerlere yönlendirilebilmelerini sağlamak için merkezi izleme tasarımı gerçekleştirilmektir. Araçların konum tespiti için RSSI ile mesafe tahmini ve konum saptama için Bounding Box metodu seçilmiştir. RSSI ile mesafe tahmini her ne kadar hata oranına sahip olsa da ortama göre gerçekleştirilecek iyi bir kalibrasyon ile kullanışlı hale getirilebilmektedir. Düğüm kalibrasyonu için hem yatay hem de dairesel kalibrasyon yöntemi denenmiş ve en iyi sonuç dairesel kalibrasyonda alınmıştır. Elde edilen sonuçlar ne kadar tutarlı olsa da bazen hedef düğümün bulunduğu bölge tespiti net olarak elde edilememiştir.

Bunu tolere edebilmek için yapılan deneylerde elde edilen sonuçların gerçek konum değerleri ile aralarındaki farkları incelenerek “ofset değeri” adı verilen bir tolerans mesafesi saptanmıştır. Ofset değerinin sıfırdan büyük veya küçük olması durumu incelenerek ölçüme nasıl katılacağı belirlenmiştir. Bu ofset değeri opsiyonel olduğu için istenildiği zaman aktif ve pasif edilebilmektedir. Konum tespitinin net yapılamadığı durumlarda kullanılması olumlu sonuçlar verirken diğer durumlarda kullanımına ihtiyaç duyulmamıştır.

Ayrıca tezde ofset değerli kalibrasyon sonuçlarının diğer kalibrasyonlarla detaylı karşılaştırılmasına yer verilmiştir. Araçların yönlendirilmesinde ise “Kenar Maskeleme” adı verilen bir yöntem geliştirilmiştir. Bu yöntemde hedef düğüm, biri maske düğüm diğeri kenar düğümler olmak üzere farklı çalışma mantığına sahip en az iki düğümle haberleşerek yönlendirme bilgisini almaktadır.

Önerilen sistemin uygulanabilir olduğunu görmek için öncelikle sistemde kullanılan düğümlere ait simülasyon yapılmıştır. Simülasyon başarılı olmuş ve gerçek sensör düğümleri ile uygulama aşamasına geçilmiştir. Simülasyonu yapılmış düğüm kodları gerçek düğümler üzerinde sorunsuz çalışmış ve sistemin uygulanabilir olduğu görülmüştür. Sistemin bilgisayar tarafında, toplayıcı (sink) düğümden alınan verilerin okunabilmesini, gerektiğinde BaseStation üzerinden ağa veri paketi gönderilmesini sağlayan program Java dilinde geliştirilmiş ve denenmiştir.

Önerilen sistem, GPS'in yetersiz kaldığı kapalı mekanlarda konum tespiti yapma, araçların veya çalışanların hedef bölgeye yönlendirilebilmelerini sağlama, ortam hakkında istatistiksel bilgiler toplama gibi işletme kolaylığı sağlayan avantajlar sunmaktadır. Bunun yanı sıra geometriye bağlı olması, küçük alanlarda büyük kapsama alanlı düğümlerle kullanılamaması, RSSI ile mesafe tahmininde bir hata oranının olması dezavantajlarıdır. Ayrıca bu yapı RFID teknolojisi ile beraber kullanılarak daha esnek, kabiliyetli sistemlerin tasarımında ve kapalı alanlarda insansız araç yönlendirme çalışmalarında da kullanılabilme imkanına sahiptir.

Kablosuz sensör ağları konusunda çalışmak isteyen araştırmacıların, günümüzde kullanılan teknolojilerin, kullanılan algoritmaların avantaj ve dezavantajlarını iyi bilmeleri oldukça önemlidir. Bununla beraber kablosuz sensör ağları, üzerinde çalışmayı hak eden heyecan verici ve geniş bir çalışma alanını da beraberinde getirmektedir.

5.2. Öneriler

Yapılan tez çalışması sürecinde ortaya çıkan ve sonraki çalışmalarda ele alınabilecek çeşitli araştırma ve iyileştirme konuları aşağıda maddeler halinde verilmiştir:

- Düğümlerin RSSI değerleri ile yapılacak kalibrasyon hesaplamalarında yumuşak hesaplama tekniklerin kullanılması,
- RFID teknolojisi ile birlikte kullanılarak insansız araçların kapalı alanlarda yönlendirilebilmesini sağlayacak çalışmaların gerçekleştirilmesi,
- Kapalı alanlarda bölge analizi yapabilecek farklı yaklaşım metodlarının denenmesini sağlamak.

Önerilen bu çalışmaların, kablosuz sensör ağları ile hayata geçirilmesi beraberinde birçok farklı çalışma konusuna beraberinde getirecektir.

KAYNAKLAR

- [1] **Akyildiz, I.F., Weilian Su, Yogesh Sankarasubramaniam, and Erdal Cayirci**,2002, Wireless Sensor Networks: A Survey, Computer Networks, 38(4):393--422, March-Georgia Institute of Technology
- [2] **Tahir Emre Kalaycı, Ege Üniversitesi** ,2009, Kablosuz Sensör Ağlar ve Uygulamaları, Akademik Bilişim 2009(AB'09),Harran Üniversitesi,Şanlıurfa
- [3] **H. Qi, S. S. Iyengar, K.Chakrabarty**, 2001, Distributed sensor networks - a review of recent research Journal of the Franklin Institute, Elsevier Science
- [4] **K. Kosbar, D. Lord, A. Snodgrass, L. Hogrebe, and M. Brown** ,2005, IFT – UMR,Ad Hoc Network Project, Telemetry Learning Center Department of Electrical and Computer Engineering University of Missouri – Rolla
- [5] **Vidyasagar Potdar, Atif Sharif, Elizabeth Chang**,Wireless Sensor Networks: A Survey, 2009, Advanced Information Networking and Applications Workshops, WAINA '09. International Conference on Digital Ecosystems and Business Intelligence Institute, Page 636-641, Curtin University of Technology, Perth, WesternAustralia
- [6] **Ceylan O**,Telsiz Sensör Ağlarında Konum Belirleme, 2007, Kontrol Dergisi, Sayı 1, Sayfa 6-10, İTÜ, İstanbul
- [7] **Döner Ç**, 2010, Kablosuz Algılayıcı Ağları İle Orman Yangınlarının Tespiti Ve Önlenmesi Lisans Tezi, Ege Üniversitesi, İzmir
- [8] **Yıldırım K.S, Kantarcı A**,2010, Kablosuz Algılayıcı Ağları İçin Tinyos İle Uygulama Geliştirme, Akademik Bilişim 2010 (AB'10), Muğla
- [9] **Arslan O**, 2009, ZigBee İle Bina İçi Güvenlik Otomasyon Sistemi Lisans Tezi, İTÜ, İstanbul
- [10] **Ke Liu**, 2005, TinyOS/Motes, nesC Tutorial Presented Dept. Of Computer Science, <http://pdffinder.net/TinyOS/Motes,-nesC-Tutorial.html>, 3 Mart 2012
- [11]**www.tinyos.net**, 2 Ocak 2012

- [12] **Joo-Ho Lee, Member, IEEE, and Hideki Hashimoto**, 2003 Controlling Mobile Robots in Distributed Intelligent Sensor Network, Industrial Electronics, Volume 50, Page 890-902
- [13] **Michel Vasilevski, Hassan Aboushady, Francois P**, 2007, Modeling Wireless Sensor Network Nodes Using SystemC-AMS, International Conf. On Microelectronics, Cairo, Egypt
- [14] **Anna Ha'c**, Wireless Sensor Network Designs, 2003 , University of Hawaii at Manoa, Honolulu, Publisher: Jhon Wiley & Sons, USA
- [15] **Zhi-Yan Cao, Zheng-Zhou Ji, and Ming-Zeng Hu**, 2005, An Image Sensor Node for Wireless Sensor Networks, ITCC '05, School of Computer Science and Technology, Harbin Institute of Technology, Harbin, China
- [16] **Teresa Riesgo Angel De Castro Centro, Jorge Portilla**, 2007, A Reconfigurable Fpga-Based Architecture For Modular Nodes In Wireless Sensor Networks, Programmable Logic, 2007. SPL '07. 2007 3rd Southern Conference on, Page 203-206, Madrid
- [17] <http://freaklabs.org/index.php/Blog/Zigbee/IEEE-802.15.4-in-the-Context-of-Zigbee-Part-2-MAC-Layer.html>, 11 Nisan 2012
- [18] **Philip Baldwin, Sanjeev Kohli**, 2004, Modeling of Sensor Nets in Ptolemy II, <http://ptolemy.eecs.berkeley.edu/visualsense/>, 12 Mart 2012
- [19] <http://www.eecs.iu-bremen.de/wiki/index.php/TinyOS>, 14 Mart 2012
- [20] **Lalit Saraswat, Pankaj Singh Yadav**, 2010, A Comparative Analysis of Wireless Sensor Network Operating Systems, Int J Engg Techsci Vol 1(1) , Page 41-47, Indonesia
- [21] **V., Baljak, S., Honiden**, 2010 "Discovery of Configurations for Indoor Wireless Sensor Networks Through Use of Simulation in Virtual Worlds", 4. International. Conf. On sensor Technologies and applic. pp:323-328, Japan

- [22] **Du, W., Mieleveille,F., Navarro,D.**,2010 "IDEA1: A SystemC-based system-level simulator for wireless sensor networks," in IEEE Inte. Conference WCNIS2010, pp.618-623 IEEE,
- [23] **Aguilar, L., Licea, G., García. M, J. A.**,2011, “An Experimental Wireless Sensor Network Applied in Engineering Courses”, Comp. Appl. Eng. Educ., 19: Page : 777–786.
- [24] **Alberto,C.,Sergio,S., Amiincio,S., et.al**, 2009,"Simulation platform for wireless sensor networks based on the TrueTime toolbox," Proceedings - IECON 2009, 35th Annual Conference of the IEEE, pp.2115-2120.
- [25] **AYNA Mutlu A.,ALKAN Ferhat** ,2010”Telsiz Duyarga Ağlarında Alınan İşaret Gücü Göstergesi (RSSI) İle Konum Belirleme Ve Haritalama”,http://www.emo.org.tr/ekler/203890ee8233a1a_ek.pdf, 20 Mart 2012
- [26] **BEKÇİBAŞIU., TENRUHM.**, 2012, “Kablosuz Algılayıcı Ağlarda Konum Saptama Teknikleri ve Mesafe Bağımlı Tekniklerde Dördüncü Çapa Yaklaşımı”, akademik Bilişim (AB’12), Muğla Üniversitesi, Muğla
- [27] www.scadds.com/, 20 Mayıs 2012
- [28] <http://mtlweb.mit.edu/researchgroups/icsystems/uamps/>, 20 Mayıs 2012
- [29] <http://www.princeton.edu/~mrm/zebranet.html>, 20 Mayıs 2012
- [30]<http://research.cens.ucla.edu/>, 20 Mayıs 2012
- [31] **Erik Welsh, Walt Fish, J. Patrick Frantz**,2003 ,GNOMES: A Testbed For Low Power heterogeneous Wireless Sensor Networks, ISCA3(4), Page 836,839, Rice University
- [32]**Yingli Zhu, Jingjiang Song, Fuzhou Dong**, 2011,Applications of wireless sensor network in the agriculture environment monitoring, Procedia Engineering, Volume 16, Page 608-614

- [33]**Wen-Wei Chang, Tung-Jung Sung, Heng-Wei Huang, Wei-Chih Hsu, Chi-Wei Kuo, Jhe-Jhao Chang, Yi-Ting Hou, Yi-Chung Lan, Wen-Cheng Kuo, Yu-Yen Lin, Yao-Joe Yang**, 2011, A smart medication system using wireless sensor network Technologies, Sensors and Actuators, Volume 172, Page 315-321
- [34]**Frank Stajano, Neil Hault, Ian Wassell, Peter Bennett, Campbell Middleton, Kenichi Soga**, 2010, Smart bridges, smart tunnels: Transforming wireless sensor Networks from research prototypes into robust engineering infrastructure , Ad Hoc Networks, Volume 8, Page 872-888
- [35]**SHA Chao, WANG Ru-chuan, HUANG Hai-ping, SUN Li-juan**, 2010, A type of healthcare system based on intelligent wireless sensor Networks The Journal of China Universities of Posts and Telecommunication, Volume 17, Page 30-39
- [36]**Andrey Somov, Alexander Baranov, Alexey Savkin, Denis Spirjakin, Andrey Spirjakin, Roberto Passerone**, 2011, Development of wireless sensor network for combustible gas monitoring –Sensors and Actuators A: Physical, Volume 171, Page 398-405
- [37]**E.S. Nadimi, V. Blanes-Vidal, R.N. Jorgensen, S. Christensen**, 2011, Energy generation for an ad hoc wireless sensor network-based monitoring system using animal head movement, Computers and Electronics in Agriculture, Volume 75, Page 238-242
- [38]**Yang Yuan, Zhang Shen, Wang Quan-fu, Song Pei**, 2009, Long distance wireless sensor networks applied in coal mine-Procedia Earth and Planetary Science, Volume 1, Page 1461-1467
- [39]**A. Bayo, D. Antolín, N. Medrano, B. Calvo, S. Celma**, 2010, Early Detection and Monitoring of Forest Fire with a Wireless Sensor Network System, Procedia Engineering, Volume 5, Page 248-251
- [40]**Amine Haoui, Robert Kavalier, Pravin Varaiya**, 2007, Wireless magnetic sensors for traffic surveillance, ScienceDirect, Page 13-26

- [41] **Miguel S. Familiar, José F. Martínez, Iván Corredora, Carlos García-Rubio,** 2011, Building service-oriented Smart Infrastructures over Wireless Ad Hoc Sensor Networks: A middleware perspective, Computer Networks The International Journal of Computer and Telecommunication Networking, P1303
- [42] **Zhi Sun, Pu Wang, Mehmet C. Vuran, Mznah A. Al-Rodhaan, Abdullah M. Al-Dhelaan, Ian F. Akyildiz,** 2011, BorderSense: Border patrol through advanced wireless sensor Networks, Ad Hoc Networks, Page 468-478
- [43] <http://www.memsic.org>, TelosB Datasheet, MEMSIC
- [44] **Shah Bhatti, James Carlson, Hui Dai, Jing Deng, Jeff Rose, Anmol Sheth, Brian Shucker, Charles Gruenwald, Adam Torgerson, Richard Han,** 2005, MANTIS OS: An Embedded Multithreaded Operating System for Wireless Micro Sensor Platforms , Mobil Networks & Applications Journal
- [45] https://www.sics.se/contiki/wiki/index.php/Main_Page, 23 Mart 2012
- [46] **Hai-Ying Zhou, Kun-Mean Hou, Chanet, J.-P. , de Vaulx, C., De Sousa, G.,**2006, LIMOS: A Tiny Real-Time Micro-Kernel for Wireless Objects, Wireless Communications, Networking and Mobile Computing
- [47] <http://www.liteos.net/>, 23 Mart 2012
- [48] **Seungmin Park, Jin Won Kim, Kee-Young Shin, and Daeyoung Kim,** 2006, A Nano Operating System for Wireless Sensor Networks, ICACT 2006
- [49] <http://www.nanork.org/>, 23 Mart 2012
- [50] <http://adsabs.harvard.edu/abs/2012JCMSI...5...24S>, 23 Mart 2012
- [51] <http://www.sics.se/project/mspsim/>, 23 Mart 2012
- [52] **Mehmet Pekmezci,** “Kablosuz Sensor Ağlarında Lokalizasyon Problemi”, 2010, <http://wsnlab.maltepe.edu.tr/wsn/docs/localization.pdf>, 23 Mart 2012
- [53] <http://wiki.python.org/moin/TkInter>, 23 Mart 2012

- [54] **Mohit Saxena, Puneet Gupta, Bijendra Nath Jain**, 2008, Experimental Analysis of RSSI-based Location Estimation in Wireless Sensor Networks, Third Int. Conf. On Communication System Software and Middleware, India
- [55] **Nick Verbaendert, David Henderickx, Jeroen Doggen**, 2008, Emote: A Wireless Sensor Network Monitoring Tool With Node Localization Using RSSI, Papers of e-Lab Master Theses, The University Collage of Antwerp, Belgium
- [56] **Marchel Baunach, Clemens Mühlberger, Florian Füller**, 2009, Analysis of Radio Signal Parameters fo Calibrating RSSI Localization System, University of Würzburg, Germany
- [57] **Chuan-Chin Pu and Wan-Young Chung**, 2008, An Integrated Approach for Position Estimation using RSSI in Wireless Sensor Network, Journal Of Ubiquitous C 78 Onvergence Technology, Vol.2, Page.2-12,

EKLER

TELOSB Düğüme Ait Teknik Özellikler

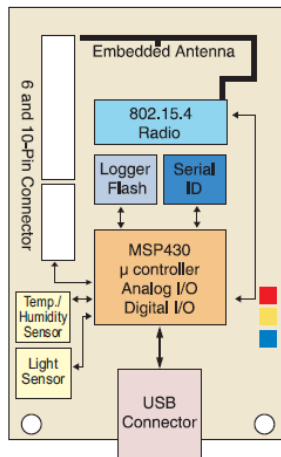
TELOSB

TELOSB MOTE PLATFORM

- IEEE 802.15.4 Compliant
- 250 kbps, High Data Rate Radio
- TI MSP430 Microcontroller with 10kB RAM
- Integrated Onboard Antenna
- Data Collection and Programming via USB Interface
- Open-source Operating System
- Integrated Temperature, Light and Humidity Sensor

Applications

- Platform for Low Power Research Development
- Wireless Sensor Network Experimentation



TPR2420CA Block Diagram



TELOSB

MEMSIC's TelosB Mote TPR2420 is an open-source platform designed to enable cutting-edge experimentation for the research community. The TPR2420 bundles all the essentials for lab studies into a single platform including: USB programming capability, an IEEE 802.15.4 radio with integrated antenna, a low-power MCU with extended memory and an optional sensor suite. TPR2420 offers many features, including:

- IEEE 802.15.4 compliant RF transceiver
- 2.4 to 2.4835 GHz, a globally compatible ISM band
- 250 kbps data rate
- Integrated onboard antenna
- 8 MHz TI MSP430 microcontroller with 10kB RAM
- Low current consumption
- 1MB external flash for data logging
- Programming and data collection via USB
- Sensor suite including integrated light, temperature and humidity sensor
- Runs TinyOS 1.1.11 or higher

The TelosB platform was developed and published to the research community by UC Berkeley. This platform delivers low power consumption allowing for long battery life as well as fast wakeup from sleep state. The TPR2420 is compatible with the open-source TinyOS distribution.

TPR2420 is powered by two AA batteries. If the TPR2420 is plugged into the USB port for programming or communication, power is provided from the host computer. If the TPR2420 is always attached to the USB port no battery pack is needed.

TPR2420 provides users with the capability to interface with additional devices. The two expansion connectors and onboard jumpers may be configured to control analog sensors, digital peripherals and LCD displays.

TinyOS is a small, open-source, energy-efficient software operating system developed by UC Berkeley which supports large scale, self-configuring sensor networks. The source code software development tools are publicly available at: <http://www.tinyos.net>

Specifications	TPR2420CA	Remarks
Module		
Processor Performance	16-bit RISC	
Program Flash Memory	48K bytes	
Measurement Serial Flash	1024K bytes	
RAM	10K bytes	
Configuration EEPROM	16K bytes	
Serial Communications	UART	0-3V transmission levels
Analog to Digital Converter	12 bit ADC	8 channels, 0-3V input
Digital to Analog Converter	12 bit DAC	2 ports
Other Interfaces	Digital I/O,I2C,SPI	
Current Draw	1.8 mA	Active mode
	5.1 μ A	Sleep mode
RF Transceiver		
Frequency band ¹	2400 MHz to 2483.5 MHz	ISM band
Transmit (TX) data rate	250 kbps	
RF power	-24 dBm to 0 dBm	
Receive Sensitivity	-90 dBm (min), -94 dBm (typ)	
Adjacent channel rejection	47 dB	+ 5 MHz channel spacing
	38 dB	- 5 MHz channel spacing
Outdoor Range	75 m to 100 m	Inverted-F antenna
Indoor Range	20 m to 30 m	Inverted-F antenna
Current Draw	23 mA	Receive mode
	21 μ A	Idle mode
	1 μ A	Sleep mode
Sensors		
Visible Light Sensor Range	320 nm to 730 nm	Hamamatsu S1087
Visible to IR Sensor Range	320 nm to 1100nm	Hamamatsu S1087-01
Humidity Sensor Range	0-100% RH	Sensirion SHT11

Resolution	0.03% RH	
Accuracy	± 3.5% RH	Absolute RH
Temperature Sensor Range	-40°C to 123.8°C	Sensirion SHT11
Resolution	0.01°C	
Accuracy	± 0.5°C	@25°C
Electromechanical		
Battery	2X AA batteries	Attached pack
User Interface	USB	v1.1 or higher
Size (in)	2.55 x 1.24 x 0.24	Excluding battery pack
(mm)	65 x 31 x 6	Excluding battery pack
Weight (oz)	0.8	Excluding batteries
(grams)	23	Excluding batteries

ÖZGEÇMİŞ

Güngör YILDIRIM

DSİ 9. BÖLGE MÜDÜRLÜĞÜ

Barajlar Şubesi

Elazığ

0 424 2386911/1713

E-posta: gyildirim@dsi.gov.tr

- 1978 : Elazığ’da doğdu.
- 1995 : Mehmet Akif Ersoy Lisesi’ni bitirdi.
- 2000 : Fırat Üniversitesi Mühendislik Fakültesi Elektrik–Elektronik Bölümünden mezun oldu.
- 2000-2005 : Fırat Üniversitesi Tunceli MYO’da Öğretim Görevlisi olarak çalıştı.
- 2005-.. : DSİ 9. Bölge Müdürlüğü Barajlar Şubesinde Elektrik-Elektronik Müh. olarak çalışmaktadır.
- 2009-... : Fırat Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Ana Bilim Dalı Donanım Bilim Dalında Yüksek Lisansa başladı.