

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

**MOBİL ROBOTLARDA
MARKOV KONUMLAMA TABANLI
NAVİGASYON**

YÜKSEK LİSANS TEZİ

Mustafa TANIŞ

Mekatronik Mühendisliği Anabilim Dalı

Mekatronik Mühendisliği Programı

EKİM 2015

**MOBİL ROBOTLARDA
MARKOV KONUMLAMA TABANLI
NAVİGASYON**

YÜKSEK LİSANS TEZİ

**Mustafa TANIŞ
(518121035)**

Mekatronik Mühendisliği Anabilim Dalı

Mekatronik Mühendisliği Programı

Tez Danışmanı: Prof. Dr. Hakan TEMELTAŞ

EKİM 2015

İTÜ, Fen Bilimleri Enstitüsü'nün 518121035 numaralı Yüksek Lisans Öğrencisi **Mustafa TANIŞ**, ilgili yönetmeliklerin belirlediği gerekli tüm şartları yerine getirdikten sonra hazırladığı “**MOBİL ROBOTLARDA MARKOV KONUMLAMA TABANLI NAVİGASYON**” başlıklı tezini aşağıdaki imzaları olan jüri önünde başarı ile sunmuştur.

Tez Danışmanı : **Prof. Dr. Hakan TEMELTAŞ**
İstanbul Teknik Üniversitesi

Jüri Üyeleri : **Yrd.Doç.Dr. Sıddık Murat Yeşiloğlu**
İstanbul Teknik Üniversitesi

Yrd.Doç.Dr. Özgür Turay Kaymakçı
Yıldız Teknik Üniversitesi

.....

Teslim Tarihi : **11 Eylül 2015**
Savunma Tarihi : **7 Ekim 2015**

Eşime ve çocuklarıma,

ÖNSÖZ

Bu tez çalışmasında danışmanlığımı yapan, edindiği akademik ve hayat tecrübelerini benimle paylaşan değerli hocam Prof. Dr. Hakan TEMELTAŞ'a teşekkür ederim. Bu zorlu ve stresli süreçte yanımda olduğu ve bana katlandığı için eşime teşekkür ederim. Son olarak beni günlere getiren değerli anne ve babama çok teşekkür ediyorum.

Ekim 2015

Mustafa TANIŞ
(Araştırma Görevlisi)

İÇİNDEKİLER

Sayfa

ÖNSÖZ	vii
İÇİNDEKİLER	ix
KISALTMALAR.....	xi
ÇİZELGE LİSTESİ.....	xiii
ŞEKİL LİSTESİ.....	xv
ÖZET	xvii
SUMMARY	xix
1. GİRİŞ	1
2. MOBİL ROBOT KONUMLAMA PROBLEMİ.....	5
2.1 Konumlama Problem Çeşitleri	6
2.1.1 Pozisyon izleme.....	6
2.1.2 Global konumlama	6
2.1.3 Kaçırılmış robot problemi	7
2.1.4 Harita edinme	7
2.2 Harita Türleri	7
2.2.1 Topolojik harita.....	7
2.2.2 Izgara modeli harita.....	8
2.2.3 Metrik harita	9
2.3 Göreceli Konumlama.....	9
2.3.1 Odometri.....	9
2.4 Mutlak Konumlama.....	10
3. MARKOV KONUMLAMA YÖNTEMİ.....	11
3.1 Markov Konumlamanın Temeli.....	11
3.2 Matematiksel Yaklaşım	13
3.3 Markov Konumlama Algoritması.....	15
4. OLASILIKSAL MODELLEMELER.....	17
4.1 Aracın Olasılıksal Modellemesi	17
4.1.1 Diferansiyel tekerlek modeli	17
4.2 Sensörün Olasılıksal Modellemesi	19
4.2.1 Lazer sensör yapısı	19
4.2.2 Gözlem modeli	22
5. YAPILAN ÇALIŞMALAR	25
5.1 Navigasyon	25
5.2 Aracın Olasılıksal Modelinin Elde Edilmesi	26
5.3 LMS 200 Sensörün Olasılıksal Modeli.....	30

5.3.1 Gözlem modeli hesaplanması.....	32
5.4 Temel Yaklaşım	34
5.5 Uygulama Verileri	35
5.6 Markov Konumlama Yönteminin Sonuçları	38
6. SONUÇ VE ÖNERİLER	45
KAYNAKLAR.....	47
ÖZGEÇMİŞ	49

KISALTMALAR

LMS	: Laser Measurement Systems
MATLAB	: Matrix Laboratory
MAP	: Maksimum Soncul
GPS	: Genel Pozisyonlama Sistemi
ROS	: Robot Operating System
MRPT	: Mobil Robot Programming Toolkit

ÇİZELGE LİSTESİ

	<u>Sayfa</u>
Çizelge 5.1: Uygulanan yöntemlerin kıyaslanması	43

ŞEKİL LİSTESİ

	<u>Sayfa</u>
Şekil 1.1 : Mobil otonom araç çalışma şeması.	2
Şekil 2.1 : Mobil robot konumlandırmanın grafiksel modeli [1].	5
Şekil 2.2 : Topolojik harita.	8
Şekil 2.3 : Izgara modeli harita.	8
Şekil 2.4 : Metrik harita [2].	9
Şekil 3.1 : Markov konumlandırmanın temeli.	12
Şekil 4.1 : Önde ve arkada birer kastor tekerlekli diferansiyel mobil robotun yapısı.	17
Şekil 4.2 : Önde ve arkada çift kastor tekerlekli diferansiyel mobil robotun yapısı.	18
Şekil 4.3 : Diferansiyel robotu öteleme işlemi.	18
Şekil 4.4 : Sick LMS 200.	20
Şekil 4.5 : Lazer sensörün robot üzerindeki konumu.	20
Şekil 4.6 : LMS 200 çalışma prensibi.	21
Şekil 4.7 : LMS 200 iletişim şeması [3].	21
Şekil 4.8 : LMS 200 ölçüm modeli.	22
Şekil 4.9 : LMS200 lazer sensörlü diferansiyel robot için gözlem modeli.	22
Şekil 4.10 : Çift taraflı LMS200 lazer sensörlü diferansiyel robot için gözlem modeli.	23
Şekil 5.1 : Mobil robotun navigasyonu.	26
Şekil 5.2 : Mobil robotun hareket planı.	27
Şekil 5.3 : Mobil robotun hareketi.	28
Şekil 5.4 : İki boyutlu gaussian değişkenin dağılım grafiği.	32
Şekil 5.5 : Mobil robotun lazerle gözlem yapması.	33
Şekil 5.6 : MATLAB uygulaması.	34
Şekil 5.7 : Mobil robotun konumu ve belirlenen hedef noktası.	35
Şekil 5.8 : Mobil robotun konumu ve belirlenen hedefe ulaşması.	36
Şekil 5.9 : X-Y düzleminde mobil robot konumlanması.	36
Şekil 5.10 : Robotun yönelimi.	37
Şekil 5.11 : Mobil robotun hareketi.	37
Şekil 5.12 : X yönündeki hata oranı.	38
Şekil 5.13 : Y yönündeki hata oranı.	39
Şekil 5.14 : θ açısındaki hata oranı.	39
Şekil 5.15 : X yönündeki hata oranı.	40
Şekil 5.16 : Y yönündeki hata oranı.	40
Şekil 5.17 : θ açısındaki hata oranı.	41

Şekil 5.18	: X yönündeki hata oranı.....	41
Şekil 5.19	: Y yönündeki hata oranı.....	42
Şekil 5.20	: θ açısındaki hata oranı.....	42

MOBİL ROBOTLARDA MARKOV KONUMLAMA TABANLI NAVİGASYON

ÖZET

Çağımızda mobil robotların kullanımı çok geniş bir alana yayılmış durumdadır. Mobil robotlar, engellerden sakınarak hedefe ulaşma, belirli bir yol planını takip etme gibi görevleri yerine getirdiği alanlarda kullanılmaktadır. Mobil robotların kendilerine yüklenen bu görevleri gerçekleştirebilmeleri için nerede olduklarını bilmeleri gerekmektedir. Bu nedenle konum hesaplamaları ile bir mobil robotun sürekli olarak ‘Ben neredeyim?’ sorusuna cevap aranmaktadır. Bu nedenlerden dolayı bir mobil robotun etkili bir şekilde kullanılabilmesi için konumlama verileri önemli bir yere sahip olmaktadır.

Mobil robotların nerede konumlandığı bilgisini elde etmek için bir çok yöntem önerilmiştir. Bu yöntemler hem uygulanan metod hem de kullanılan veriler ve haritalar açısından farklılıklar gösterebilmektedir. Bunlar göreceli konumlama ve mutlak konumlama olarak bilinir. Konumlama işleminin doğruluğunu arttırmak için bu iki yöntem beraber uygulanır. Bunlardan en bilineni odometri yöntemi ile konum belirleme metodudur. Fakat odometri yönteminde tekerleklerden alınan verilerin işleme alınması ile yapılan bu konumlama işleminde hatalar mevcut olabilir. Bunun nedeni ise, tekerleklerin zemin üzerinde patinaj yapması veya kaymasıdır. Bu nedenler ile konumlama işlemi doğru bir şekilde gerçekleşemez.

Mobil robotları konumlandırma işleminde zamanla olasılıksal yaklaşımlar devreye girmiştir. Olasılıksal konumlama, robotun çevreden aldığı bilgileri işleyerek nerede olma ihtimalini arttırıp azaltarak konumunu tespit etmesidir. Olasılıksal konumlama da çeşitli durumlar söz konusudur. Tez de bu çeşitli durumlar anlatılmış ve incelenmiştir. Robotun ilk pozisyonun bilindiği durağan bir ortamda olasılıksal konumlama yöntemlerinden biri olan markov konumlama yöntemi tercih edilmiştir.

Markov konumlama yönteminde robotun sahip olduğu bir inanış vardır. Bu aslında robotun o konumdaki olasılıksal dağılım fonksiyonudur. Mobil robot başlangıç inancına sahip olarak başlar. Bu tez çalışmasında robotun başlangıç inancı bulunmaktadır. Ayrıca markov konumlama yönteminin bir gereği olarak robotun konumun belirlenmesinde odometrik veriler ve dış sensörlerden alınan veriler beraber işlenerek konumlama gerçekleştirilmiştir.

Markov konumlama yöntemi iki ayrı modelleme işleminin uygulanmasıdır. Araçtan odometrik verilerin alınması için bir hareket modellemesi ve kullanılan sensörden alınan verilerin kullanılması için bir algısal modelleme yapmak gerekir.

Bu tez çalışmasında kullanılan araç diferansiyel tekerelek modeline sahiptir. Diferansiyel tekerlek sistemi incelenmiş ve kullanılan aracın olasılıksal modellemesi

yapılmıştır. Ayrıca dış sensör olarak LMS 200 tipi lazer sensör kullanılmıştır. Sensör hakkında incelemeler yapılmış ve modellenerek çalışmada kullanılmıştır.

Yapılan bu tez çalışmasında amacımız odometrik verilerin işlenmesi ile yapılan konumlamadaki hataları gidermek için olasılıksal bir konumlama yöntemi olan markov tabanlı navigasyonu gerçekleştirmektir. Markov yönteminin temeli, bulunduğu çevrede olasılık yoğunluk fonksiyonunu kullanarak bir tahminleme işlemi gerçekleştirmesidir.

Yapılan çalışmalarda, markov konumlama tekniğinin ızgara tabanlı harita üzerinde yapılan işlemlerinde işleme merkezi olarak kullanılan bilgisayara çok fazla bir yük bindiği görülmüştür. Bunun nedeni çok fazla olasılıksal değerin hesaplanması ve işlem yükünün çokluğudur. Bu konuda iyileştirmeler yapılmış ve gerekli sonuçlar elde edilmiştir.

NAVIGATION IN MOBILE ROBOTS BASED ON MARKOV LOCALIZATION

SUMMARY

In our age, the use of mobile robots is spread over a very wide area. Mobile robots avoiding obstacles in achieving the goal plan are fulfilled tasks such as tracking a particular way that spaces are used. Loaded mobile robots in order to perform these tasks themselves, must know where they are. Therefore, a mobile robot with localization calculations "Where am I?" sought to answer the question. For these reasons, positioning data has an important place for using mobile robots effectively.

A robot does not possess natural senses like human beings have. Human beings get information about their surroundings through vision and other sensing powers. A robot cannot explore an unknown environment unless it is provided with some sensing sources to get information about the environment. Different kinds of sensors such as sonars, odometers, laser range finders, inertial measurement units (IMU), global positioning system (GPS) and cameras are used to make a robot capable of sensing a wide range of environments. The map of the environment is a basic need of a robot to perform indoor services like moving room to room, gripping and picking an object from one place and taking it to another place. To perform such type of services, the robot should not only know about the environment but while it is moving it should also be aware of its own location in that environment.

Mobile robot localization can be seen as a problem of coordinate transformation. Maps are described in a global coordinate system, which is independent of robot's position. Localization is the process of establishing correspondence between the map coordinate system and the robot's local coordinate system.

Localization, that is the estimation of a robot's location from sensor data, is a fundamental problem in mobile robotics. The aim of localization is to estimate the position of a robot in its environment, given a map of the environment and sensor data. Most successful mobile robot systems to date utilize localization, as knowledge of the robot's position is essential for a broad range of mobile robot tasks.

A lot of method have been proposed for mobile robots where are located to obtain information. These methods may vary in terms of both applied methods that are used for both data and maps. These are known as relative localization, and absolute localization. Relative localization, evaluating the position and orientation using information provided by various on board sensors (e.g. encoders, gyroscopes, accelerometers etc). Absolute localization, obtaining the absolute position using beacons, landmarks or satellite based signals. To increase the accuracy of the positioning process, these two methods are applied together. One of the most known is odometry which a method of determining the position. The classical technique for

a wheeled robot to calculate its position is to track its location through a series of measurements of the rotations of the robots' wheels, a method often termed odometry. Odometry requires a method for accurately counting the rotation of the robot wheels. A standard method for doing this is to instrument the wheels with optical shaft encoders. But odometry method, the processing of the data from the wheels made with this process, errors in positioning may be present. The reason for this is to skid on the ground or the wheels is slipping. For these reasons, the positioning operation cannot be carried out correctly.

Probabilistic approaches in mobile robot positioning process over time came into effect. Probabilistic localization the robot by processing the information it receives from the environment by increasing or decreasing the likelihood of the location being where to identify. There are various cases in probabilistic positioning. This thesis has been investigated and described several situations. Markov localization method which is preferred in this thesis with a static environment where the initial of robot's position is known.

Markov localization addresses the problem of state estimation from sensor data. Instead of maintaining a single hypothesis as to where in the world a robot might be, Markov localization maintains a probability distribution over the space of all such hypothesis. Markov localization method, there is a belief that the robot has. This is actually the probability distribution function of the robot at that location. The mobile robot starts with an initial belief. In this thesis, the robot has an initial belief. Determination of the robot position we use odometric data from wheels and external sensor data together. That is the requirement of Markov localization.

The Markov localization is the implementation of two separate modelling process. The first one is act modelling, the represents the estimation of actual position of the mobile robot based on the actual information from odometry method. The second one is observe modelling, the represents the improved estimation of the position based on the actual measurements of other sensors.

The vehicle which is used in this thesis has a differential drive system. A differential wheeled robot is a mobile robot whose movement is based on two separately driven wheels placed on either side of the robot body. It can thus change its direction by varying the relative rate of rotation of its wheels and hence does not require an additional steering motion. Differential drive system studied and probabilistic model were applied.

In the implementation of thesis we use laser range finder sensor which is external sensor. A laser range finder which uses a laser beam to determine the distance to an object. The most common form of laser range finder operates on the time of flight principle by sending a laser pulse in a narrow beam towards the object and measuring the time taken by the pulse to be reflected off the target and returned to the sender. LMS 200 laser range finder used in this implementation. Detailed information about the LMS 200 laser sensor are given in the thesis.

In this thesis our objective is to perform probabilistic Markov based navigation to resolve the error which is coming from processing odometric data. For all possible localization Markov method calculates probability distribution function for belief. Therefore mobile robot has a belief about where it is.

At the end we compare different situations for robot. We create a static environment and get it's map. We apply three different method to localizing the mobile robot. We choose a scenario. Mobile robot avoid from obstacles and arrive the target point. For getting and comparing result give rightful situations. In application we gave the same target point to the mobile robot and the results are compared in terms of error rates.

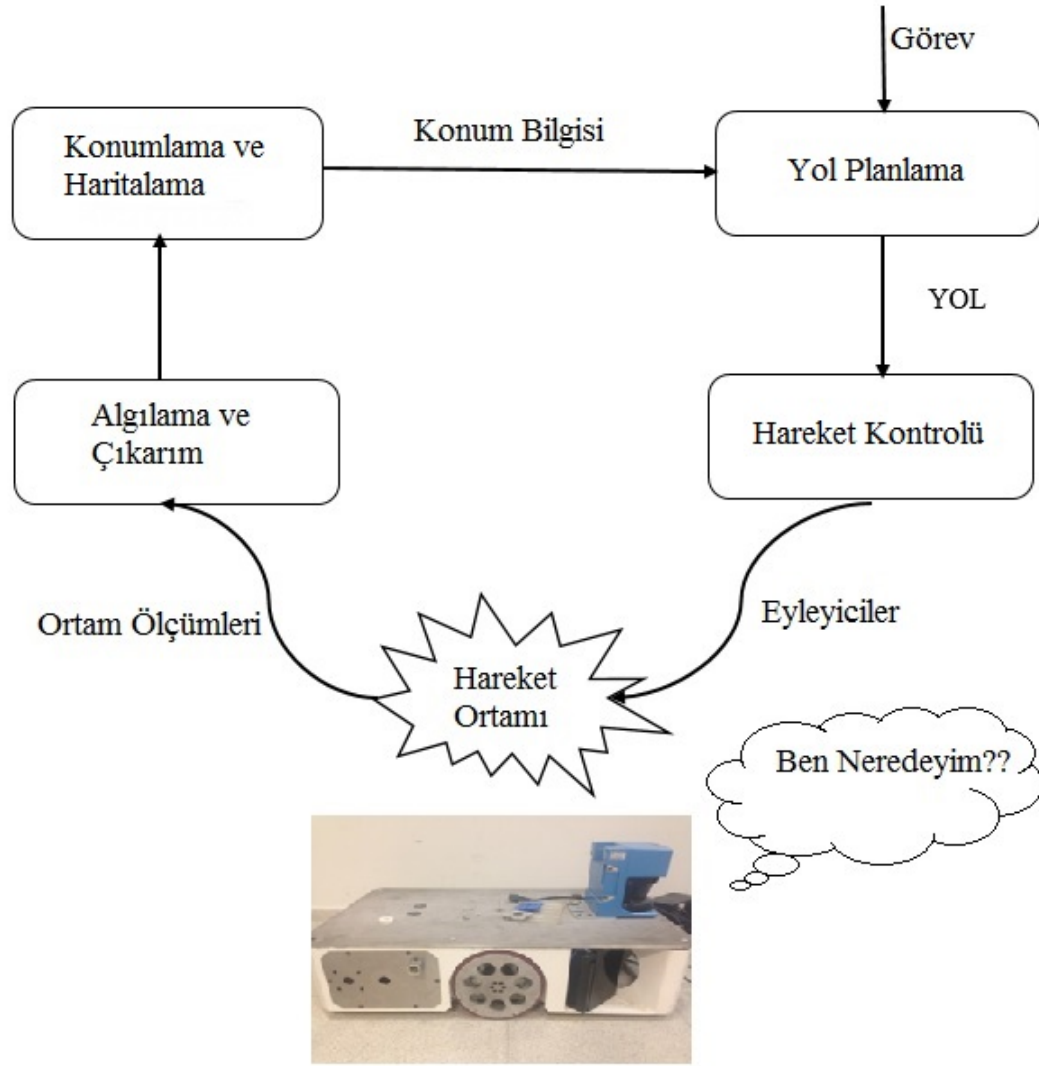
1. GİRİŞ

Günümüzde robotlar birçok alanda kullanılmaktadır. Robotlara duyulan ihtiyacın artmasıyla bir çok farklı alanda farklı robot çeşitleri insanlığa hizmet etmektedir. Bu çeşitliliğin devamı için robotların gelişimi son yarım yüzyılda büyük bir hız kazanmıştır. Teknolojik ilerlemeler robotlara ayrı bir güç ve kabiliyet katmıştır.

İnsan yaşamına verilen değerin artmasıyla, tehlike arz eden ortamlarda robot kullanımı yaygınlaşmıştır. Can kayıpları en aza indirgenmeye çalışılmıştır. Robot üretiminin ilk zamanlarında, robotlar sabit konumlu ve kolları ile verilerin görevleri yerine getiriyorlardı. Daha sonraları robotların gezgin olmaları istenmiş, konumlarının, amaçlarının kullanıcı tarafından bilinmesi ve belirlenmesi mümkün kılınmıştır.

İnsanlar üzerinden yola çıkarak robotların gezgin olma durumunu düşünelim. Düşünün ki, işyerinizde bütün işlerinizi bitirdiniz ve çalışma süreniz doldu. Şimdi eve gitmeye hazırsınız. Peki eve nasıl gideceksiniz? Hiç böyle bir soruyu yönelttiniz mi kendinize? Belki cevabınız, yürüyerek tabi ki de olacaktır. Fakat bu durum öyle sanıldığı kadar kolay değil. Eve doğru giderken veya öncesinde çözmeniz gereken üç temel problem vardır. Birincisi nerede olduğunuzu bilmelisiniz. İkincisi evinizin veya gitmek istediğiniz yerin nerede olduğunu bilmelisiniz. Son olarak evinize ve gideceğiniz yere varmak için bir rota belirlemelisiniz. İnsanoğlunun yetenekleri gereği bu sorunları her gün iç güdüsel olarak her gün çözümlenmekteyiz ve bu yeteneğimiz gözden geçiriyoruz. Robotların da bu eylemleri gerçekleştirmesi için bu yeteneğe sahip olmaları gerekir.

Gezgin robotların görevlerini başarılı şekilde gerçekleştirmesi için bulunduğu ortamı bilmesi veya öğrenmesi, konumunu bilmesi gerekmektedir. Robotlar bu hedeflerini sensörler aracılığıyla gerçekleştirir. Robot, gezinim yapabilmek için şu üç soruya yanıt arar. "Neredeyim?", "Nereye gitmek istiyorum?", "Oraya nasıl ulaşabilirim?". Burada birinci soru, robotun çevrede nerede konumlandığı bilgisini arar. İkinci soru, robotun ulaşmak istediği yere gidebilmek için hedef belirlemesini araştırır. Son soru



Şekil 1.1: Mobil otonom araç çalışma şeması.

ise robotun ulaşmak istediği yere giderken bu işlemi nasıl yapacağını sorar. İlk soru bizim için daha büyük önem teşkil eder. Çünkü ilk soruya yanıt bulmadan diğer soruların yanıtları anlamsız kalır. Bu tez de amacımız genel olarak ilk soruya yanıt bulmak olacaktır.

Gerçekçi bir rota planlaması ve navigasyon için mobil robotun konumunun doğru bir şekilde belirlenmesi ve ortamın doğru bir biçimde algılanmasına bağlıdır. En temel yöntem ise mobil robotu kumanda etmektir. Bu yöntem ile mobil robot kendi rotasını belirleyemez ve ortamın haritasını oluşturmaz.

Mobil robotlar bulundukları ortamı, hareketli veya hareketsiz nesnelerin konumlarını sensörler vasıtasıyla elde edebilir. Ortamın tam olarak modellenmesi çeşitli nedenlere ile bazen güçleşmektedir.

Mobil robotlarda konum belirleme çok temel ve başlıca problemlerdendir. Bu durumda robotun pozisyonu ve açısının belirlenmesi gerekir. İlk olarak odometrik veriler ile konumlama işlemi gerçekleştirilmiştir. Bu konuyla ilgili [4] çözümün gerçekleşmemesi durumunda, robot sadece nesnelerden ve engellerden kaçmış olur. Fakat mobil robot konumunu bildiği sürece sensörler yardımıyla ortamın haritasının çıkarılması, hedef noktaya gidilmesi, engelleri tanınması ve onlara çarpmaması gibi amaçları gerçekleştirebilir.

Robotun konumu, kapalı ve yüzeyün düz olduğu, ortamın iki boyutlu olduğu x, y ve robotun yönelimini belirten θ açısı ile ifade edilir. Konum belirlemede en temel uygulama odometrik veriler kullanılarak yapılan uygulamalardır. Ancak yapılan çalışmalar sadece odometrik veriler kullanılarak yapılan çalışmalarda hatalı sonuçlar elde edildiğini göstermektedir [5]. Odometrik verilerde bu hatanın çeşitli sebepleri olabilir. Bunlar robotun kayması, tekerleklerin patinaj yapması ve hatanın toplanarak artması kaynaklı olabilir.

Odometrik yöntemin hatalarını azaltmak için ek sensörler (laser, sonar, kamera) ile bu hataların en aza indirilmesi hedeflenmiştir [6]. Kullanılan bu sensörler ile odometrik verilerin bir araya getirilip değerlendirilmesi ile daha doğru bir konumlamam işlemi gerçekleştirilmiş olur. Odometrik yöntemin hatalarından kurtulmak için çeşitli yöntemler ve sensörler kullanılmıştır [6–8] . Bu tezde sadece odometrik verilere bağlı kalmadan, olasılıksal yöntemlerden biri olan Markov olasılıksal konumlama yöntemi kullanılması amaçlanmıştır.

Konumlama, bir nesnenin pozisyonunu belirleme yöntemidir. Gezgın robotlarda konumlama ise x, y, θ değerlerini dikkate alarak robotun yerinin belirlenmesi işlemidir.

Bu çalışmada iki boyutlu bir çevrede robotun ilk pozisyonu bilinerek, robotun hareketi ile robotun nerede olacağının tahminlenmesi işlemidir. Olasılıksal robot konumlandırma kullanılan çeşitli yaklaşımlardan Markov olasılıksal konumlama yöntemi tercih edilmiş ve işlemler bu yönde yapılmıştır.

Bu alıřmada, ortamı iki boyutlu tarayabilen LMS 200 kullanılmıřtır. İlk olarak ortam sensör vasıtasıyla taranmıřtır.

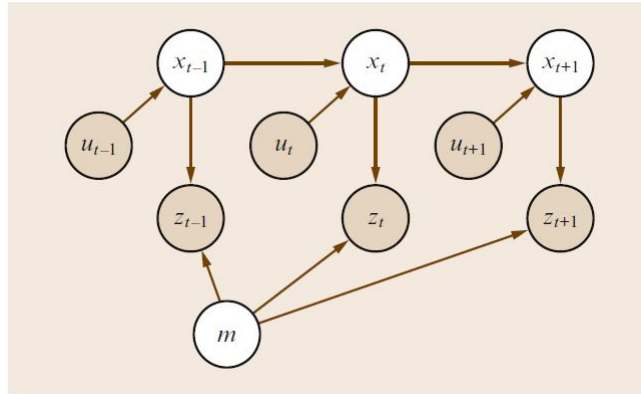
Konumlama iin kullanılan Markov yntemi ile ilgili bir ok alıřma vardır. En temel alıřmalardan biri olan [9] incelenmiřtir. Bu alıřmada dinamik bir evrede konumlandırma yapılmıřtır. Fakat bu tezde ortam statik olarak belirlenmiřtir.

Tezin ikinci blmnde mobil robotlarda konumlama problemi zerinde incelemeler ve alıřmalar yapılmıřtır. Konumlama probleminin kaa ayrıldıėı, ne tr harita eřitlerinin bulunduėunu ieren bilgiler verilmiřtir. nc blmnde olasılıksal konumlandırma yntemi ve bunlardan biri olan Markov olasılıksal konumlama yntemi incelenmiř, temel prensipleri belirtilmiř ve matematiksel yaklařımlar ifade edilmiřtir. Drdnc blmnde kullanılan deney platformu ve kullanılan sensrler tanıtılmıřtır. Bunların olasılıksal modellemeleri yapılmıřtır. Beřinci blmnde yapılan deneysel alıřmalar hakkında detaylı bilgiler verilmiř, yntemler uygulanmıř ve kıyaslanmıřtır. Son olarak sonu blmnde ise alıřmanın eksileri ve artıları tartıřılmıřtır.

2. MOBİL ROBOT KONUMLAMA PROBLEMİ

Mobil robot konumlandırma, verilen haritaya uyarak robotun bulunduğu pozisyonu belirleme problemidir. Genel olarak buna pozisyon tahminleme denir. Mobil robot konumlama, robotların algısal olarak pozisyonu bilmesine yönelik temel bir konudur. Robotikte neredeyse tüm alanlarda konumlama temel bir çalışmayı oluşturur.

Mobil robot konumlama bir koordinat dönüşüm problemi olarak alınabilir. Haritalar, mobil robotun bulunduğu pozisyondan bağımsız olarak global koordinat sistemi ile ifade edilir. Konumlama, robotun bulunduğu koordinat sistemi ile haritanın koordinat sistemi arasında kurulmaya çalışılan bir işlemdir. Bu dönüşümün bilinmesi robotun koordinat düzleminde olan nesnelerin konumunu belirlemesine yardımcı olur. Robotun konumunun belirlenebilmesi, robotun görevini doğru ve etkili bir şekilde tamamlamasını sağlar. Genel olarak robotun konumu üç etkin öge ile tanımlanır. Bunlar (x, y, θ) . Eğer mobil robotun pozisyonu olan $X_t = (x, y, \theta)^T$ başarılı bir şekilde belirlenir ise haritada da konumlandırma rahatça görülebilir.



Şekil 2.1: Mobil robot konumlandırmanın grafiksel modeli [1].

Robot konumlama işleminde bir haritaya ihtiyaç duyulur. Şekil 2.1 de görüldüğü gibi robotun önceden tanımlanmış bir haritası vardır. Mobil robotun bütün amacı, çevreyi algılayarak, önceden belirlenmiş harita yardımıyla konumunu belirleyip istenilen veya hedeflenen noktaya doğru hareket etmektir.

Burada öne çıkan temel sorunlardan biri mobil robotun pozisyonu $X_t = (x, y, \theta)^T$ direkt olarak belirlenememesidir. Bu sorunun çözümü sensörlerden elde edilen verilerden çıkarımda bulunarak yapılan işlemler sonucu elde edilir.

Konum belirleme probleminde istenilen, sensörlerden ve odometrik verilerden yararlanılarak robotun bulunduğu pozisyonun kestirilmesidir. Konumlama problemi olasılıksal tabanlı ifade edilirse, robot pozisyonun belirli bir yerde olma olasılık yoğunluk fonksiyonuna sahiptir. Robotun belirli pozisyonda olma olasılığına robotun inancı da denir. Mobil robot, her yeni veri karşısında bu inancını güncellemektedir. Böylelikle robotun inancı gerçeğe doğru yakınsar. Olasılıksal konum belirleme, inanış veya olasılık yoğunluk fonksiyonun elde edilmesi veya çıkarsamasıdır.

2.1 Konumlama Problem Çeşitleri

Konumlandırma problemi kabaca üç parçaya ayrılabilir. Bunlar; konum izleme, global konumlandırma ve harita edinmedir.

2.1.1 Pozisyon izleme

Birçok uygulamada robotun ilk pozisyonu bilinmektedir. Mobil robot uygulamanın gerçekleştiği sırada sensörden alınan ölçümlerle konumunu güncellemelidir. Sadece iç sensörlerden alınan verilerle yapılan bu güncelleme hatalara neden olur. Bu hatalar toplanarak daha da büyüyebilir. Bu nedenle mobil robotun kesin pozisyonunu belirlemek için dışarıdan bir sensöre ihtiyaç vardır. Sensörlerden yapılan ölçümlerle çevrenin haritası arasında yapılan doğru eşleme başarılı bir sonuç verir. Fakat ölçülen ile verilen çevre modelindeki verileri eşleştirmek oldukça güçtür. Verilen çevrenin büyümesi problemi daha da güçleştirecektir.

2.1.2 Global konumlama

Pozisyon izlemede, ilk pozisyon tahmini bilinmektedir. Fakat tamamıyla otonom mobil robotlarda bu pek de mümkün olamamaktadır. İlk pozisyonun hiç bilinmediği veya çok kısıtlı bilgi ile bilindiği konumlandırmaya global konumlandırma adı verilir. Global konumlandırma, pozisyon takibi işlemine göre daha güçtür. Global

konumlandırmanın karmaşıklığı mobil robotun bulunduğu ortamın büyüklüğüne ve simetrisine bağlı olarak farklılıklar gösterir.

2.1.3 Kaçırılmış robot problemi

Kaçırılmış robot problemi, robota bilgi verilmeden robotun pozisyonun aniden değiştirilmesi yani kaçırılması nedeniyle ortaya çıkar. Kaçırılmış robot problemi global konumlandırmanın daha zor olan bir türüdür. Çünkü kaçırılmış robot probleminde robot nerede olduğunu bilmez iken bir inanışa sahip olması gerekir. Global konumlandırma da ise robot nerede olduğunu bilmez.

2.1.4 Harita edinme

Hem pozisyon izleme uygulaması hem de global konumlandırma uygulaması çevrenin bir haritasına ihtiyaç duyar. Bu harita kimi zaman bir binanın planından elde edilirken kimi zamanda robotta bulunan sensörler yardımıyla oluşturulur. Mobil robotta bulunan sensörler aracılığıyla harita elde edebilmek için mobil robotun pozisyonunun bilinmesi gerekir.

2.2 Harita Türleri

Çevre ile ilgili bilgileri ifade etmenin çeşitli yolları vardır. Tez çalışmasında uygulamanın iç ortamda gerçekleştirilmesi hedeflenmiştir. İç ortamlar genelde ev ve ofis ortamlarıdır. Bu ortamları nitelendirmede çeşitli yöntemler kullanılmıştır. Örneğin, 1996 yılında yapılan çalışmada iç ortamları haritalama yöntemi olarak "grid-based" ve "topolojik harita" öne sürülmüştür [10].

2.2.1 Topolojik harita

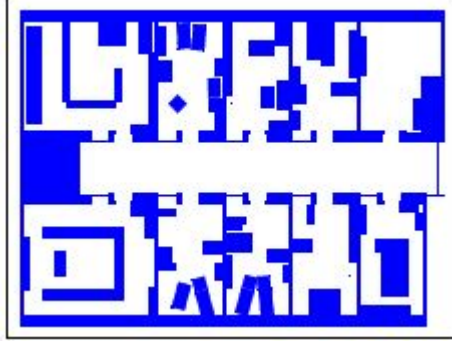
Çevre birbirine bağlı bir grafik ile ifade edilir. Harita üzerinde bulunan düğüm noktaları önemli yerlere denk gelir. Olasılıksal yaklaşımda robotun konumu için bir çok varsayım yapılabilir. Robotun haritada birden fazla yerde olma olasılığı vardır. Bunu engellemek için robotun hareketi ve sensörlerden elde edilen veriler ile çok yapı bir durumdan tek bir konuma yaklaşılr. Örneğin uygulamanın yapıldığı ortamda birden fazla masa olsun. Sensörler yardımıyla robot bir masa önünde olduğunu anlar

A 6x6 grid with alternating gray and white squares. A blue circle is located in the white square at row 3, column 4.

8

2.2.3 Metrik harita

Bu çeşit haritalar genelde ofis, ev gibi ortamları ifade etmek için kullanılır. Çevreyi tanımlamak için farklı renkler ve şekiller kullanılır. Ortama kuş bakışı bir açı ile yaklaşır.



Şekil 2.4: Metrik harita [2].

2.3 Göreceli Konumlama

Konum belirleme işleminde yaygın olarak kullanılan bir işlemdir. Mobil robotların konumlama işleminde uzun süre kullanılmıştır. Göreceli konum belirleme, robotun başlangıç pozisyonundan başlayıp devam eden konum bilgilerinin işlenmesi temeline dayanır. Bu temel işlemi uygulayabilmek için enkoderler, jiroskoplar, ivme ölçer gibi çeşitli sensörler kullanılır. Göreceli konum belirleme, hızı ve yönü belli olan bir robotun belli bir süre sonra nerede olacağını tahmin etmesidir. Bu işlem sırasında hatalar söz konusu olabilir. Bu hatalar giderek büyüyebilir. Çünkü işlem sırasında robotun bulunduğu bir önceki pozisyonundan faydalanılır. Bu konumlamaya en iyi örnek odometri yöntemidir.

2.3.1 Odometri

Odometri, hareket sensörlerinden gelen bilgileri kullanarak robotun pozisyonun tahmin edilmesi işlemidir. Mobil robotun tekerleklerinde bulunan sensör vasıtasıyla tekerleklerin dönüş sayısı, hızı bilinerek robotun nerede olacağı tahminlenir. Odometri, kısa mesafelerde doğruluğu nedeniyle önceleri tercih edilmiştir. Ayrıca kesin konumlama işlemine göre daha ucuz ve kolaydır. Fakat bilgiler tekerlekler üzerinden alındığı için robotun tekerleklerinin kayması, patinaj yapması konum

belirleme için hatalara sebep olur. Odometri, bir önceki konumu kullanarak tahminleme yaptığı için bu hata giderek büyüyüp robotun nerede olduğunu bilmeyip kaybolmasına sebep olur.

2.4 Mutlak Konumlama

Mutlak konumlam işlemi robotun pozisyon ve açısının küresel düzlemde referans olarak konumlandırılmasıdır. Mutlak konumlama işleminde bazı yer işaretlemeleri sayesinde de konumlama yapılmıştır [11]. Bu anlamda en mutlak konumlama işlemi için en popüler olan yöntem GPS kullanılarak yapılan konumlamadır. GPS sayesinde uydu sinyalleri kullanılarak robotun yeryüzündeki hangi enlem, boylam ve yükseklikte olduğu tespit edilebilmektedir. Eğer ölçümler düzgün şekillerde yapılırsa hata oranının çok düşük seviyelere indirilir. Robotun pozisyonu böylelikle başlangıç pozisyonundan ve zamandan bağımsız kılınır. GPS yönteminin de bazı olumsuz yönleri bulunmaktadır. Bunlardan en önemlisi robotun konumunun güncellenmesi için gerekli olan örnekleme zamanının uzunluğu, GPS sinyalinin kapalı ortamlarda kaybolması ve sadece açık alanlarda kullanıma imkan vermesidir. GPS kullanımının mümkün olmadığı iç ortamlarda (mağara, tünel vs) mutlak konumlama robot dışında bulunan kamera, lazer sensör, sonar sensör, yardımıyla yapılan konumlam işlemidir. Bu işlemde sistemin her adımında işlemi bir öncekinden bağımsız gerçekleştirmesi nedeniyle hatalar en aza indirgenmiştir.

Konumlama işleminin başarılı ve düzgün yapılabilmesi için göreceli ve mutlak konumlama bir arada kullanılarak birbirini desteklemesi sağlanmıştır. Her iki yapıdan gelen veriler tek bir sistemden geliyormuş gibi davranılarak işleme alınmıştır.

3. MARKOV KONUMLAMA YÖNTEMİ

İç ortamlarda robot hareketinin devamlılığının sağlanması için nerede olduğunu bilmesi gerekir. Son zamanlarda mobil robotlarının konumlarının ve pozisyonlarının belirlenmesine olan bilimsel ilgi artmıştır. Konumlandırma problemi genel olarak, sisteme tanımlanan harita kullanılarak, robotun üzerindeki sensörler ve kameralar aracılığıyla algılanan veriler kullanılarak bir konumlandırma tahmini yapılır ve bunun doğruluğu test edilir. Genel olarak konum belirleme yöntemi Bayes kuralı uyarınca gerçekleştirilir. Bayes kuralı, bir durumun oluşma olasılığını elde edilen bilgiler ışığında nasıl değişeceğini gösteren bir kuraldır. Bayes teoreminden yararlanılarak yapılan konumlama işlemlerinden biri de Markov konumlama yöntemidir. Burada Bayes teoreminden bahsetmek gerekir. Bayes teoremi, olasılık kuramı içinde incelenen önemli bir konudur. Bu teorem bir rastlantı değişkeni için olasılık dağılımı içinde koşullu olasılıklar ile marjinal olasılıklar arasındaki ilişkiyi gösterir. Bayes teoremi bir stokastik süreç sırasında ortaya çıkan bir rastgele A olayı ile bir rastgele olay B için koşullu olasılıkları ve marjinal olasılıkları arasındaki ilişkidir.

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

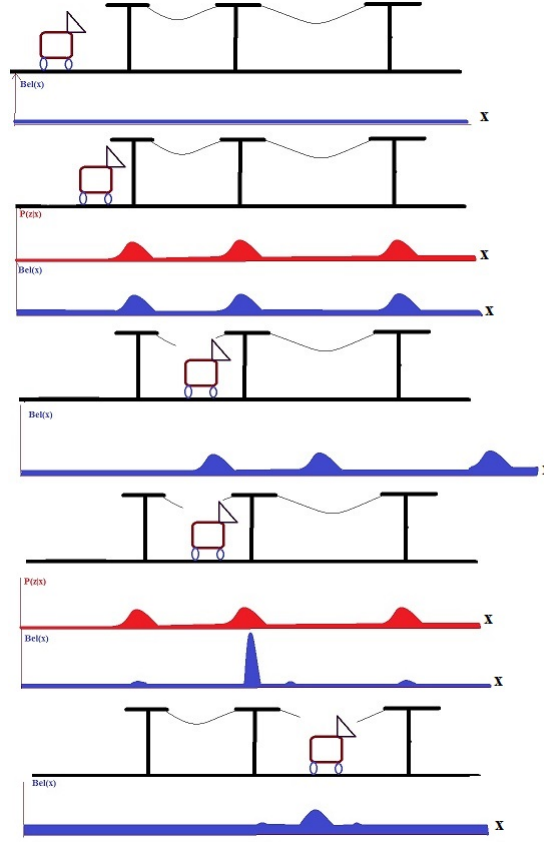
Bayes teoremi eğer B gözlemlenmiş ise, A gözlemi hakkındaki inançların ne şekilde güncelleştirilebileceğini ortaya çıkartır.

Markov yöntemi ilk olarak yapılan bu çalışmalarda [9, 12, 13] karşımıza çıkmaktadır. Markov yöntemi, mobil robotların tam ve kesin olarak pozisyonlarının belirlenemediği bir durumda, olasılıksal bir çalışma tabanına ile pozisyonun tahminlemesi yöntemi ile bulunmasını sağlar.

3.1 Markov Konumlamanın Temeli

Markov yöntemi, sensörlerden alınan veriler ile pozisyon tahminlemedir. Markov konumlandırma yönteminde matematiksel detaylara girmeden önce basit bir örnek ile

açıklayalım. Basit ve anlaşılır olması sebebi ile sadece tek bir düzlemde hareket eden robotun hareketini inceleyelim.



Şekil 3.1: Markov konumlandırmanın temeli.

Şekil 3.1 de gösterilen modeli inceleyelim. Robot ortamda herhangi bir yerde olduğunu varsayalım. Robota nerede olduğu söylenmemiştir. Markov konumlandırma bu belirsizlik durumunu düzgün dağılımlı fonksiyon ile ifade eder. Şimdi robot sensörleri vasıtasıyla nerede olduğunu sorgular ve direğin yanında olduğunu bulur. Markov konumlandırma yöntemi direğin yanında olma olasılığını artırır. Herhangi bir yerde olma olasılığını(inancını) azaltır. Şekil 3.1 deki ikinci diagramda bu durum açıkça görülmektedir. Olasılığın çoklu model olduğu görülmektedir.

Şimdi ise robotun bir metre ileriye gittiğini düşünelim. Şekil 3.1 deki üçüncü diagramda görüldüğü üzere Markov konumlandırma robotun bu ilerleyişi ile olasılık dağılımını kaydırma işlemini gerçekleştirir. Robotun hareketinden kaynaklanan gürültüler(bozucular)hesaba katılarak, yani olasılık bir öncekine göre daha az kesin hale gelmektedir.

Son olarak robot ikinci kez bir direğin yanında olduğunu anlar. Bu algılama ile güncel düzgün olmayan olasılık dağılımı ile çarpılır. Sonuç olarak, Şekil 3.1 deki dördüncü diagramdaki olasılık elde edilir. Şu anda robot kesin bir sonuç ile nerde olduğunu bilmektedir. Robot hareketine devam ettiği son diagramda verilmiştir.

3.2 Matematiksel Yaklaşım

Mobil robotun konumun $l = (x, y, \theta)$ olduğunu belirtmiştik. Burada x, y kartezyen koordinat sistemini ifade ederken θ ise yönelimi belirtir. k_t robotun t anındaki gerçek konumunu belirtsin. K_t rastgele değişkenleri ifade etsin.

Robot kesin olarak yerini bilmemektedir. Onun yerine, nerede olabileceği inancına(olasılığına) sahiptir. $Bel(K_t)$ robotun t anındaki pozisyonun olasılığını ifade etsin. $Bel(K_t = k)$ ise t anındaki k konumun olasılığını ifade eder.

Olasılığı(inancı) iki farklı şekilde güncelleyebiliriz. Bunlardan ilki robot üzerinde bulunan kamera,sonar veya laser sensörlerden alınan veriler ile yada odometrik veriler aracılığıyla yapılabilir. Sensör ölçümlerini l ile odometrik ölçümleri o ile gösterelim. Bunlara karşılık gelen rastgele değişkenleri ise L, O ile ifade edelim.

$$d = d_0, d_1, \dots, d_T$$

d_t ifadesi $0 < t < T$ aralığında sensör yada odometrik verilerden alınan bilgidir.

$$P(K_t = k|d) = P(K_t = k|d_0, \dots, d_T)$$

$$P(d_{t+1}, d_{t+2}, \dots | K_t = k, d_0, \dots, d_t) = P(d_{t+1}, d_{t+2}, \dots | K_t) \forall t$$

$P(K_t = k|d)$ ifadesi hesaplanırken iki durum söz konusudur.

Birinci durum; Eğer $d_T = l_T$ ise,

$$P(K_t = k|d) = P(K_t = k|d_0, \dots, d_{T-1}, l_T)$$

Bayes formülü uygulanırsa;

$$\frac{P(l_T|d_0, \dots, d_{T-1}, K_T = k)P(K_T = k|d_0, \dots, d_{T-1})}{P(l_T|d_0, \dots, d_{T-1})}$$

Basitleştirirsek;

$$\frac{P(l_T|K_T = k)P(K_T = k|d_0, \dots, d_{T-1})}{P(l_T|d_0, \dots, d_{T-1})}$$

$$P(K_t = k|d) = \alpha_T P(l_T|K_T = k)P(K_T = k|d_0, \dots, d_{T-1})$$

$$Bel(K_T = k) = P(K_T = k|d_0, \dots, d_T)$$

$$Bel(K_T = k) = \alpha_T P(l_T|k)Bel(K_{T-1} = k) \quad (3.1)$$

İkinci durum; $d_T = o_T$ ise,

$$P(K_t = k|d) = \int P(K_T = k|d, K_{T-1} = k')P(K_{T-1} = k'|d)dk'.$$

Şimdi denklemin sağ tarafındaki birinci ifade ile ilgilenelim.

$$\begin{aligned} P(K_T = k|d, K_{T-1} = k') &= P(K_T = k|d_0, \dots, d_{T-1}, o_T, K_{T-1} = k') \\ &= P(K_T = k|o_T, K_{T-1} = k') \end{aligned}$$

Şimdi ise ikinci ifade ile ilgilenelim.

$$P(K_{T-1} = k'|d) = P(K_{T-1} = k'|d_{T-1}, o_T) = P(K_{T-1} = k'|d_{T-1})$$

$$P(K_t = k|d) = \int P(K_T = k|d, K_{T-1} = k') = P(K_{T-1} = k'|d_0, \dots, d_{T-1})dl'$$

$$Bel(K_T = k) = \int P(k|o_T, k')Bel(K_{T-1} = k')dk' \quad (3.2)$$

3.3 Markov Konumlama Algoritması

Burada Markov konumlamanın temeli (3.1) ve (3.2) denklemlerinin güncellenmesidir. İlk olarak eğer varsa ilk konum bilgisi robota verilir. Bu robotun ilk inaniş bilgis olur aynı zamanda ve işlem başlamış olur. Yukarıdaki denklemlerde görüldüğü gibi $P(k|o, k')$ ifadesi robotun hareket modeli, $P(l|k)$ ise algısal modeli temsil ediyor. Bu ifadeler sırasıyla robotun hareketinden ve sensörlerden okunan verilerden etkilenir. İlk bilgi iletimi gerçekleştirildikten sonra robotun sensörlerinden alınan bilgiler işleme alınır. İlk olarak düzeltme sabiti sıfırlanır. Ardından her konum için ilk olarak;

$$Bel^-(K_T = k) = P(l_T|k)Bel(K_{T-1} = k)$$

daha sonra;

$$\alpha_T = \alpha_T + Bel^-(K_T = k)$$

ve son olarak;

$$Bel(K_T = k) = \alpha_T^{-1} Bel^-(K_T = k)$$

işlemi gerçekleştirilir.

Odometrik bilgilerin alınıp güncellenmesi de gerekmektedir. Bu nedenle de;

$$Bel(K_T = k) = \int P(k|o_T, k') Bel(K_{T-1} = k') dk'$$

işlemi gerçekleştirilir.

Bu işlemler robotun her konumu için tekrar edilerek yapılır. Burada sensörler ile algısal modeli tanımladık ve işlemleri gösterdik. Ayrıca hareketli modeli de odometrik bilgilerin işlenmesi ile gerçekleştirmiş olduk.

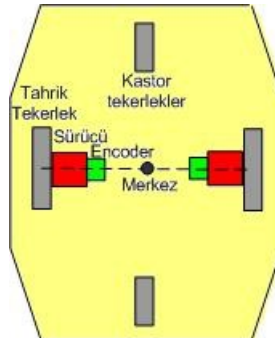
4. OLASILIKSAL MODELLEMELER

Markov konumlama işleminde prensip olarak iki temel konu öne çıkar. Bunlardan ilki robotun inancı olan $Bel(K)$ nasıl belirtileceğidir. İkincisi ise $P(k|o, k')$ ve $P(l|k)$ olasılıklarının nasıl hesaplanacağıdır. Bu olası değerlerinin hesaplanabilmesi için olasılıksal tabanlı bir modelleme gerekir. Kullanılan araç ve sensörün modellemeleri bu kısımda anlatılmıştır.

4.1 Aracın Olasılıksal Modellemesi

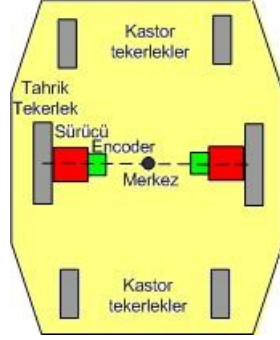
4.1.1 Diferansiyel tekerlek modeli

Diferansiyel tekerlek yapısı bir çok tekerlek yapılarına göre kontrolü karmaşık ve zor olsa da bir çok uygulama da hareket yapısının esnek olması, dönme, geri geri gelme, engellerden hızlı bir şekilde yön değiştirebilme gibi istenilen hareketleri rahatlıkla yapabilme kabiliyetlerinden dolayı tercih edilmektedir. Diferansiyel tekerlek yapısı gerek 3 tekerlekli, gerek 4 tekerlekli gerekse de 6 tekerlekli yapıda olabilmektedir. 3 tekerlekli bir yapıda önde bir kastor arka da ise 2 adet tahrik tekerleği; Şekil 4.1'de görüldüğü gibi 4 tekerlekli yapıda önde ve arkada iki kastor orta da ise 2 adet tahrik tekerleği bulunmaktadır.



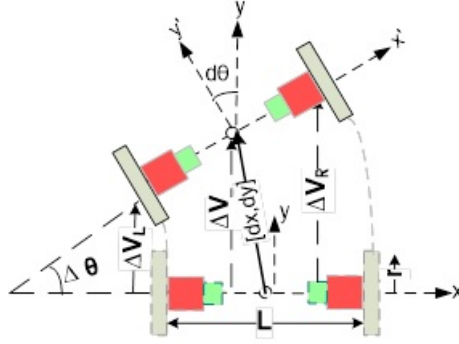
Şekil 4.1: Önde ve arkada birer kastor tekerlekli diferansiyel mobil robotun yapısı.

Şekil 4.2'de gösterilen 6 tekerlekli yapıda ise önde ve arkada ikişer adet toplam 4 kastor ve ortada 2 adet tahrik tekerleği bulunmaktadır.



Şekil 4.2: Önde ve arkada çift kaster tekerlekli diferansiyel mobil robotun yapısı.

Diferansiyel aracı bir yerden bir yere ötelediğimizde aracın pozisyonunun bulunması için gerekli olan kinematik denklemler mevcuttur. Şimdi bu denklemleri inceleyelim.



Şekil 4.3: Diferansiyel robotu öteleme işlemi.

$$C_m = \frac{\Pi D_n}{nC_e}$$

$$\Delta V_l = C_m N_{l,r} = C_m N_r$$

$$\Delta V = \frac{\Delta V_r + V_l}{2}; \Delta \Theta = \frac{\Delta V_r - \Delta V_l}{L}$$

D_n : Nominal tekerlek çapı

C_e : Encoder çözünürlüğü

n : Dişli oranı

ΔU_L : Sol tekerlek hız değişimi

ΔU_R : Sağ tekerlek hız değişimi

$\Delta\theta$: Aracın yönündeki açı değişimi

N_L : Sol tekerlek encoder değeri farkı

N_R : Sağ tekerlek encoder değeri farkı

$$dx = \Delta V \cos(\theta(t+1))$$

$$dy = \Delta V \sin(\theta(t+1))$$

$$d\theta = \Delta\theta$$

$$X_v(t+1) = X_v(t) + \Delta V \cos(\theta(t) + \Delta\theta)$$

$$Y_v(t+1) = Y_v(t) + \Delta V \sin(\theta(t) + \Delta\theta)$$

$$\theta(t+1) = \theta(t) + \Delta\theta$$

$$\Delta V = r(\Delta\omega_R + \Delta\omega_L)/2$$

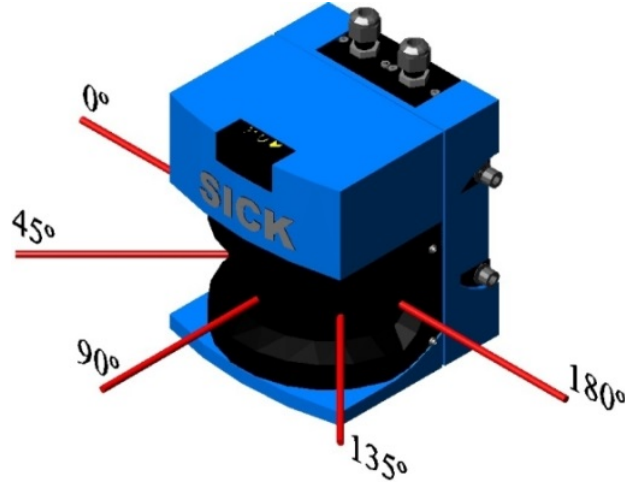
$$\Delta\omega = r(\Delta\omega_R - \Delta\omega_L)/L$$

$$\Delta\theta = \Delta\omega$$

4.2 Sensörün Olasıksal Modellemesi

4.2.1 Lazer sensör yapısı

Nesneye lazer dalga boylu ışık göndererek ölçüm yapmaktadır. Çeşitli konfigürasyon yapılarında olmakla beraber en çok kullanılanı ışını yayan ve alan elemanların tek bir cihaz içinde olanıdır. İnsansız kara araçlarında en çok kullanılan lazer tipi 2D lazer range finder olarak bilinen hem nesnenin lazere olan mesafesini hem de nesnenin lazere göre yönünü açı cinsinden ölçen tip lazerdir. Aşağıdaki şekil ve tabloda SICK firmasının üretmiş olduğu LMS 200 lazer range findera ait bilgiler görülmektedir.



Şekil 4.4: SICK LMS 200.

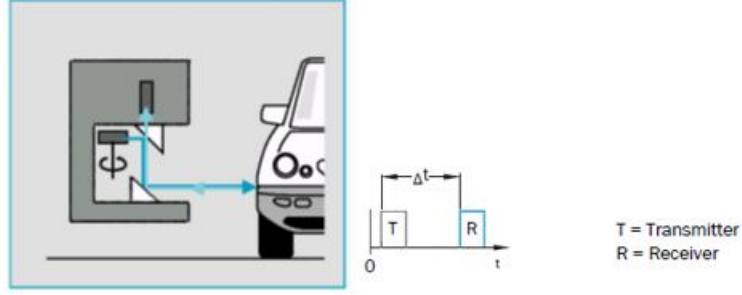


Şekil 4.5: Lazer sensörün robot üzerindeki konumu.

LMS 200 sensörü ışın darbe sistemi ile çalışır. Nesne ile sensör arasındaki mesafe lazer darbesini gönderme ve geri almak için geçen süre ile birlikte hesaplanır. Nesnenin rengi ve yüzey şekli ne olursa olsun bu prensip ile ölçüm gerçekleştirilir.

LMS 200 bir çok farklı alanda kullanılır. Bunlar,

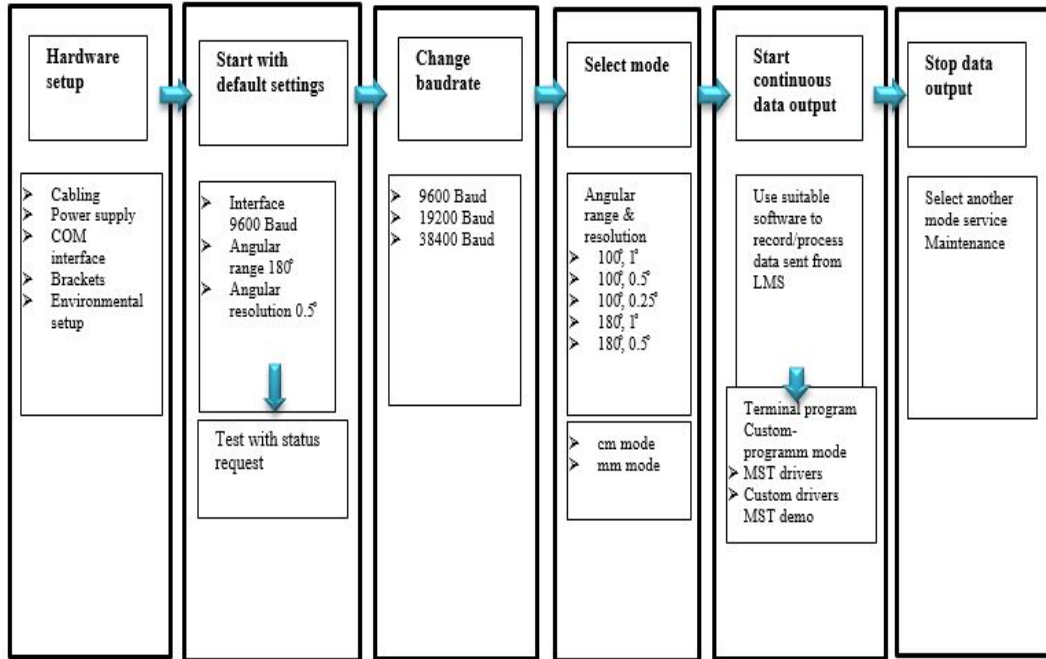
- Nesnelerin hacmini belirtmede
- Nesnelerin pozisyonunu, konumunu ve yerini belirlemede
- Konumlamada (yerleştirme işlemlerinin kontrolünde)
- Nesneleri sınıflandırmada (araç algılayıcı)
- Çevre güvenliği için açık alanları izlemede
- Toplu materyallerin hacmini ve şeklini belirlemede



Şekil 4.6: LMS 200 çalışma prensibi.

- Araç veya vinçler için çarpışmaları önlemede
- Baştan aşağıya kontrol etme ve alan izleme(çok katlı otoparklarda)

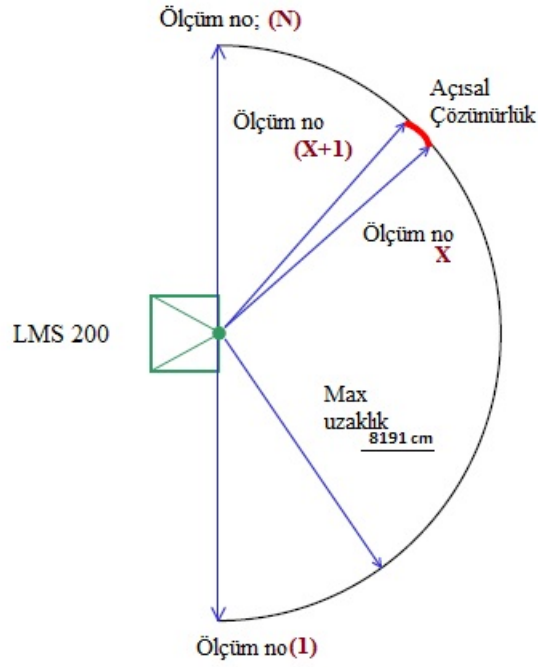
LMS 200 iletişim şeması sensörün bilgi formundan öğrenilmiştir.



Şekil 4.7: LMS 200 iletişim şeması [3].

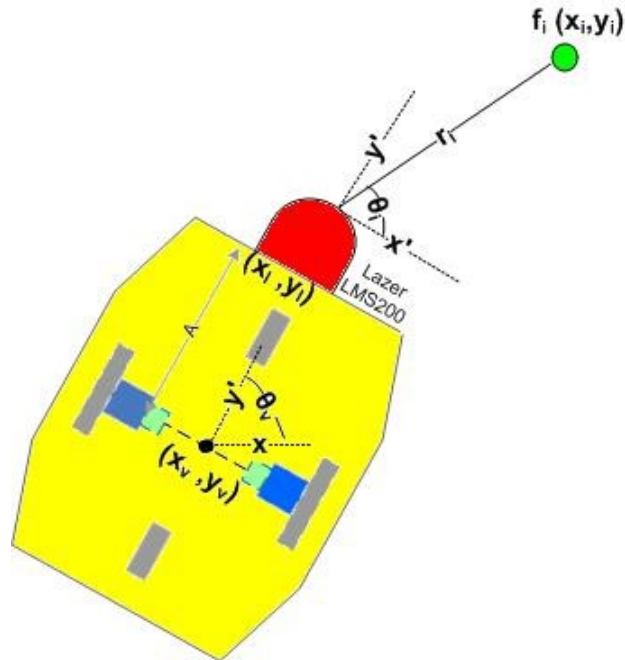
LMS 200 ün lazer taraması istenilen ölçümlere ayarlanabilmektedir. LMS 200 açısal olarak kaç derece aralıklarla ölçüm yapması gerektiğini kullanıcı belirler. Bu da çıkarılmak istenen harita da ayrıntıları ayarlamamıza yardımcı olur. Örneğin 180 derecelik alanı 1 derecelik açılar ile tarama yapabileceğiniz gibi, 180 derecelik alanı 10 derecelik bir tarama da gerçekleştirebilirsiniz. Burada LMS 200 sensöründen alınan verilerin sayısı hemen hemen on kat artmış olacaktır. Bu verilerin artışı sistemde

güncellenmesi gereken alan fazlalığı ortaya çıkarır. İşlem yükünün fazlalığı ise sistemin gerçek zamanlı çalışmasına mani olmaktadır.



Şekil 4.8: LMS 200 ölçüm modeli.

4.2.2 Gözlem modeli



Şekil 4.9: LMS200 lazer sensörlü diferansiyel robot için gözlem modeli.

$$(X_1, Y_1) = (X_v, Y_v + A)$$

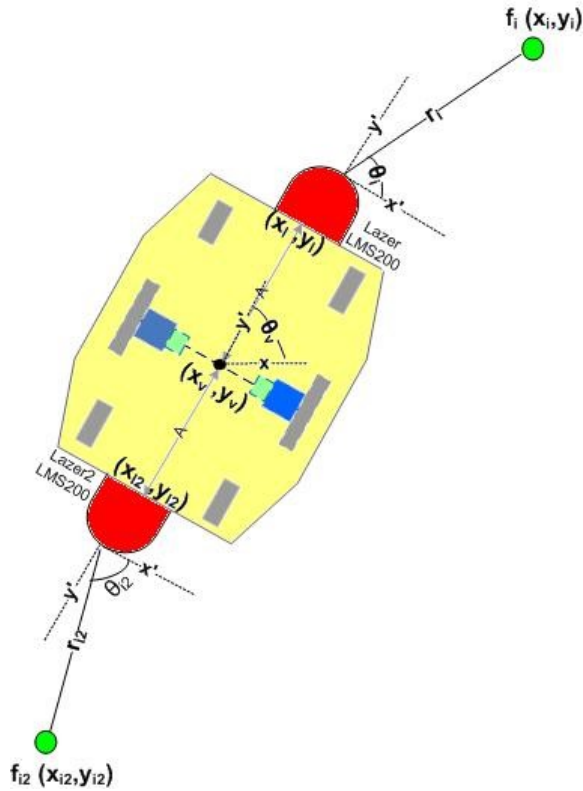
$$Z(t) = \begin{bmatrix} r_i \\ \theta_i \end{bmatrix}$$

$$Z(t) = h(x(t), w(t))$$

$$Z(t) = \begin{bmatrix} \sqrt{(X_i - X_1(t))^2 + (Y_i - Y_1(t))^2} \\ a \tan 2(Y_i - Y_1(t)/X_i - X_1(t)) - \theta_v \end{bmatrix}$$

Burada elde edeceğimiz değişkenler (X_i, Y_i) 'dir. Bunun için yukarıdaki denklemi çözmeye gerek yoktur. Matlab ortamı kullanılarak (X_i, Y_i) koordinatlarını elde edilecektir.

Çift taraflı lazer sensör olan aracın gözlem modeli nasıl çıkarılır. Tek taraflıya benzer bir şekilde gözlem modelini oluştururuz. Fakat bu iki gözlem modelinden gelen verilerin veri ilişkilendirme algoritmaları veya öngörü ve güncelleme adımları farklı olabilir.



Şekil 4.10: Çift taraflı LMS200 lazer sensörlü diferansiyel robot için gözlem modeli.

$$(X_{12}, Y_{12}) = (X_v, Y_v + A)$$

$$Z_2(t) = \begin{bmatrix} r_{i2} \\ \theta_{i2} \end{bmatrix}$$

$$Z_2(t) = h(x(t), w(t))$$

Gözlem modeli aşağıdaki gibi çıkarılır.

$$Z_2(t) = \begin{bmatrix} \sqrt{(X_{i2} - X_{12}(t))^2 + (Y_{i2} - Y_{12}(t))^2} \\ a \tan 2(Y_{i2} - Y_{12}(t) / X_{i2} - X_{12}(t)) - \theta_v \end{bmatrix}$$

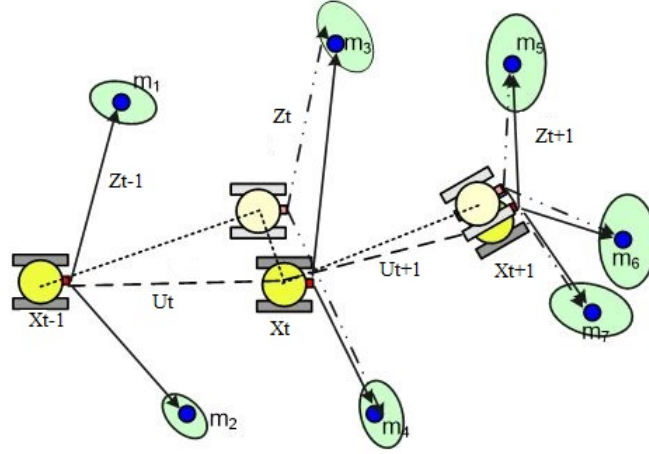
Burada elde edeceğimiz değişkenler (X_{i2}, Y_{i2}) 'dir. Bunun için yukarıdaki denklemi çözmeye gerek yoktur. Matlab programı kullanılarak (X_{i2}, Y_{i2}) koordinatlarını elde edilecektir. Bu koordinatları elde ettikten sonra önemli olan bu koordinatların lazer 1'in daha önce gördüğü bir işaretçi nesneye mi ait olduğu yoksa yeni bir nesneye mi ait olduğu bilgisidir. Bu bilgi ancak iki lazeride kapsayan özel bir veri ilişkilendirme algoritması ile elde edilebilir. Eğer lazer1 ve lazer2 iki ayrı harita oluşturup sonra bu haritalar birleştirecekse özel bir veri ilişkilendirme algoritmasına gerek kalmaz. Fakat böyle bir durumda da haritaların birleştirilmesi için özel hesaplamalar ve baz alınması gereken ortak noktaların bulunması gerekecek. Bu ortak nokta araca GPS konularak elde edilebilir. Fakat uygulama alanı olarak seçilen ortam kapalı bir çevre olduğu için GPS sistemine uygun değildir. Her iki teknikte denenerek doğru haritaya ne kadar yaklaşıldığı incelenebilir. İkinci tekniğin işlemciye daha çok yük bindireceği ve yazılımsal bir yavaşlamaya neden olacağı düşünülmüştür.

5. YAPILAN ÇALIŞMALAR

5.1 Navigasyon

Navigasyon, mobil robotun belirlenen varış noktasına en kısa, en optimum yol güzergahını tespit edip planlayarak, hedefe doğru kontrollü bir şekilde ulaşmasını sağlayan sistemdir. Bu işlem GPS gibi uydu tabanlı sensörler aracılığıyla olabileceği gibi, pusula gibi cihazlarla aracılığıyla da gerçekleştirilmektedir. Navigasyon çok genel bir tabirdir. Kara araçlarının otoyollarda gideceği güzergahının belirlenmesi, gemilerin denizde ve uçakların havada veya uzay araçlarının uzayda rotalarının belirlenmesi de birer navigasyondur. Hatta daha ilkel tabiriyle dağdaki çobanın yıldızlara ve aya bakarak yönünü bulması da bir navigasyondur. Mobil robotlar da benzer şekilde bu teknikleri kullanarak navigasyon yapabilmektedirler. Otonom mobil robotların navigasyonunda en çok kullanılan teknik mesafe sayacından lokalizasyon hesabı dediğimiz tekniktir. Mesafe sayacından lokalizasyon hesabı tek başına bir anlam ifade etmez bunun için aracın bir haritaya sahip olması gerekiyor. Mesafe sayacından lokalizasyon hesabı aracın tekerleklerinden alınan enkoder bilgilerinden aracın bulunduğu pozisyonunun (x ve y eksenindeki değerleri) ve referans alınan düzleme göre aracın yönünü ve açısının hesaplanması işlemidir. Aracın yönü gyroscope dediğimiz sensörlerle bulunabileceği gibi diferansiyel araçlarda olduğu gibi tekerleklerin farklı dönme hızlarından da hesaplanabilmektedir. Biz bu işleme lokalizasyon yani konum belirleme diyoruz. Mobil robotun kendi pozisyonunu hesaplaması da tek başına navigasyon açısından bir anlam ifade etmez. Bunun için mobil robotun nereden başladığını, nerede olduğunu ve nereye gittiğini de bilmesi gerekiyor. Bunun için mobil robota bir harita verilmesi ve başladığı noktanın bu haritada gösterilmesi gerekiyor.

Şekil 5.1’de mobil robotun navigasyonu gösterilmiştir. x_{t-1} aracın ilk pozisyonu, z_{t-1} robotun $(t - 1)$ anında lazer ile algıladığı nesnelere ait ölçümler, u_t ise robotu t



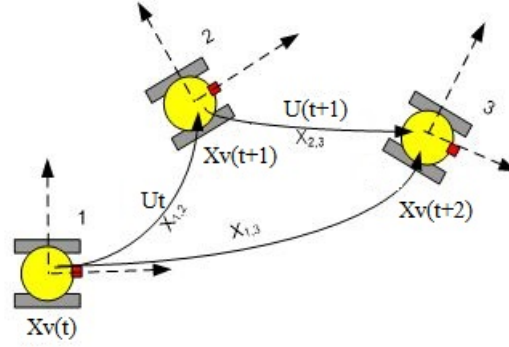
Şekil 5.1: Mobil robotun navigasyonu.

anındaki pozisyonuna götürmek için uygulanan kontrol işareti, içi dolu küçük daireler işaretçi nesneler, elipsler ise ölçüm belirsizliğini ifade etmektedir. Mobil robot $t-1$ anından t anına daha sonra da $t+1$ anına öteleniyor. Mobil robotun izlenmesi istenilen güzergah içi dolu robot sembolü ile ifade edilmiştir. Mobil robota verilen haritada m ile ifade edilen işaretçi nesnelerin pozisyonları bilinmektedir. Mobil robot u_t kontrol sinyali ile ötelendikten sonra olması gereken pozisyondan saptığı ve içi boş robot sembolünün olduğu pozisyona gittiği görülmektedir. Mobil robot o anda yaptığı z_t ölçüm sonuçlarına göre kendi pozisyonunu düzeltmekte yani güncellemektedir. Mobil robottaki bu sapma mesafe sayıcı ve robottaki diğer ölçüm belirsizliklerinden kaynaklanmakta ve robot içi boş sembolle belirtildiği yerde olmasına rağmen içi dolu robot sembolüyle belirtilen yerde olduğunu sanmaktadır. İşte lazerden alınan ölçümler aracın kendi pozisyonunu düzeltmesine ve güncellemesine yardımcı olmaktadır. Tabi bu güncelleme mükemmel olamaz. Çünkü lazer ölçümünün de bir gürültüsü ve belirsizliği bulunmaktadır. Bu belirsizlik elips ile ifade edilmiştir. Yapılan ölçüme göre nesne bu elipsin içindeki herhangi bir yere düşmektedir. Robot elipsin merkezine göre kendi konumundaki sapmayı buluyor ve ona göre haritada ki konumunu (x) güncelliyor.

5.2 Aracın Olasıksal Modelinin Elde Edilmesi

Mobil robotun Şekil 5.2’de 1. pozisyondan 2. pozisyona gidebilmesi oradan da 3. pozisyona gidebilmesi ve bu pozisyonlarda aracın konumunun kestirilmesi için

koordinat eksenlerinin transformasyonundan faydalanılacaktır. Burada iki tane $\oplus$ ve $\odot$ operatörlerini tanımlanacaktır.



Şekil 5.2: Mobil robotun hareket planı.

Yukarıdaki Şekil 5.2 gözönüne alındığında $X_{1,3} = X_{1,2} \oplus X_{2,3}$ olduğu görülür. $X_1 = (X_1, Y_1, \theta_1)$ ve $X_2 = (X_2, Y_2, \theta_2)$ için;

$$X_1 \oplus X_2 = \begin{bmatrix} X_1 + X_2 \cos \theta_1 - Y_2 \sin \theta_1 \\ Y_1 + X_2 \sin \theta_1 + Y_2 \cos \theta_1 \\ \theta_1 + \theta_2 \end{bmatrix}$$

$$\odot X_1 = \begin{bmatrix} -X_1 \cos \theta_1 - Y_1 \sin \theta_1 \\ X_1 \sin \theta_1 - Y_1 \cos \theta_1 \\ -\theta_1 \end{bmatrix}$$

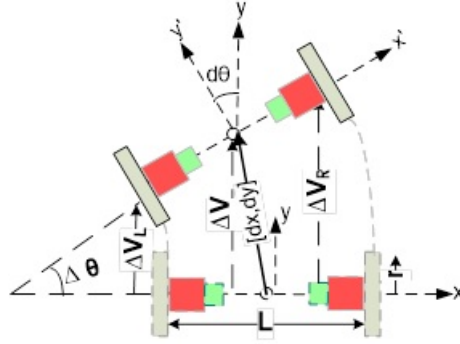
Bu denklemler bir eksenlerdeki pozisyonları diğer bir eksenlerdeki karşılıklarını ifade etmektedir. Mobil robotun önceki mesafe sayacından lokalizasyon hesabıyla elde edilen pozisyonundan $X_v(t)$, yeni bir hareketle $U(t)$ yeni mesafe sayacından lokalizasyon hesabıyla elde ettiği pozisyonuna $X_v(t+1)$ gittiğinde yeni mesafe sayacından lokalizasyon hesabıyla elde edeceği pozisyonunu aşağıdaki denklem ile hesaplanır.

$$X_v(t+1) = X_v(t) \oplus U(t)$$

Buradaki $U(t)$ yeni hareket yani mobil robotu 1 nolu eksenden 2 nolu eksene taşıyan kontrol işaretinin bir sonucu ve verileri mesafe sayacından lokalizasyon hesabından elde edilir.

$$U(t) = \odot X_v(t) \oplus X_v(t+1)$$

Bu ifadeler araç modeli halinde üretilecek olursa simülasyonlarda kullanılabilecek şekilde denklemler ayırık hale dönüştürülmüş bir non-lineer model haline getirilmiş olacaktır.



Şekil 5.3: Mobil robotun hareketi.

D_n : Nominal tekerlek çapı

C_e : Encoder çözünürlüğü

n : Dişli oranı

ΔU_L : Sol tekerlek hız değişimi

ΔU_R : Sağ tekerlek hız değişimi

$\Delta\theta$: Aracın yönündeki açı değişimi

N_L : Sol tekerlek encoder değeri farkı

N_R : Sağ tekerlek encoder değeri farkı

$$C_m = \frac{\Pi D_n}{nC_e}$$

$$X_v(t+1) = X_v(t) + \cos(\theta(t) + \Delta\theta)$$

$$Y_v(t+1) = Y_v(t) + \sin(\theta(t) + \Delta\theta)$$

$$\theta(t+1) = \theta(t) + \Delta\theta$$

$$\Delta V = r(\Delta\omega_R + \Delta\omega_L)/2$$

$$\Delta\omega = r(\Delta\omega_R - \Delta\omega_L)/L$$

$$\Delta\theta = \Delta\omega$$

Aşağıdaki birinci denklemde gösterilen hareket vektörü mobil robotu 1.eksenden 2.eksene taşıyan kontrol işaretinin ürettiği mesafe sayacından lokalizasyon hesabı ile hesaplanan vektördür. Mobil robotun hareketleri artımlı hareket olarak düşünülebilir. Yani robota uygulanan kontrol işaretinin etkisiyle mobil robotun x yönündeki pozisyonu d_x , y yönündeki pozisyonu d_y ve aracın yönü ise d kadar değişir. Bu artımlı hareketle robotun yeni pozisyonu önceki pozisyonu ile artımlı hareketin transformasyonel toplanması ile elde edilir. Bu transformasyonel ifade aşağıdaki ikinci denklem ile gösterilmiştir. Aracın dinamik modeli bize aracın eski pozisyonu ve kontrol işareti ile yeni pozisyonunun bilgisini vermektedir. Bu bağlamda ikinci denklem ile gösterilen formül aracın modelinin bir ifadesi olmaktadır. Aracın önceki pozisyonu ile uygulanan kontrol işaretinin etkisiyle elde edilen x,y ve θ 'daki değişimin transformasyonel toplanması aracın yeni pozisyonunu belirleyen bir model olmaktadır.

$$U(t) = \begin{bmatrix} d_x \\ d_y \\ d\theta \end{bmatrix}$$

$$X_v(t+1) = f(X_v(t), U(t))$$

$$X_v(t+1) = X_v(t) \oplus U(t)$$

Son denklem ile verilen robot modeline ait Jacobianlar aşağıdaki denklemler ile hesaplanır.

$$JF_x(X_v t, U_t) = \frac{\sigma(X_v t \oplus U_t)}{\sigma X_v t} = \begin{bmatrix} 1 & 0 & -dx \sin \theta_1 - dy \cos \theta_1 \\ 0 & 1 & dx \cos \theta_1 - dy \sin \theta_1 \\ 0 & 0 & 1 \end{bmatrix}$$

$$JF_u(X_v t, U) = \frac{\sigma(X_v t \oplus U_t)}{\sigma U_t} = \begin{bmatrix} \cos \theta_1 & -\sin \theta_1 & 0 \\ \sin \theta_1 & \cos \theta_1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Mobil robotun x, y ve θ 'daki yer deęiřtirmeleri mesafe sayıcından alınan bilgiyle yapılmaktadır. Bu da belli bir gürültüyü beraberinde getirmektedir. Bu gürültü sistem gürültü kovaryans matrisi Q ile ifade edilecektir. Q x, y ve θ 'daki deęişimlerin hangi standart sapma ile gerçekleşeceğini bilgisini taşımaktadır. Burada iki varsayım yapılmıştır. Bu varsayımlardan birincisi x, y ve θ 'daki gürültülerin birbirinden bağımsız oldukları, ikincisi ise bu gürültülerin gaussian olduklarıdır. Q sistem gürültü kovaryans matrisi son denklemde verilmiştir. $d - r$ ifadesi mesafe sayıcından lokalizasyon hesabının kısaltmasıdır.

$$Q = U_{d-r} = \begin{bmatrix} \sigma_{d-rx}^2 & 0 & 0 \\ 0 & \sigma_{d-ry}^2 & 0 \\ 0 & 0 & \sigma_{d-r\theta}^2 \end{bmatrix}$$

5.3 LMS 200 Sensörün Olasıksal Modeli

Ardışıl bayes teoremi harita kestiriminden esinlenilerek ortaya atılmıştır. Harita kestiriminde öncül bilgiler ve o anki ölçümler birleştirilerek deęerleri kestirilmeye çalışılmaktadır. Yeni bir gözlemde ise önceki kestirilen ve artık öncül hale gelen deęeri üzerinden yeni deęerleri kestirilmeye çalışılacaktır. Bu yaklaşım robot uygulamalarında kullanılmaktadır.

Sensör Z gözlem deęerlerini elde edecektir. Her t . adımda o ana kadar verilen bütün gözlemlerden $[Z_k = (Z_1, Z_2, \dots, Z_t) Z_t$: o anlık ölçüm deęerleri] kestirim yapmaya çalışılacaktır. Şimdi Bayes kuralını tam olarak uygulayalım.

$$P(x, Z^t) = P(x|Z^t)P(Z^t)$$

dięer bir ifadeyle

$$P(x, Z^t) = P(Z^t|x)P(x)$$

Bu iki ifadeyi eşitleyecek olursak,

$$P(x|Z^t)P(Z^t) = P(Z^t|x)P(x)$$

Koşullu gözlemlerin birbirinden bağımsız olduklarını düşünürsek;

$$P(Z^t|x) = P(Z^{t-1}|x)P(z_t|x)$$

Z_t :t anındaki gözlemler Yukarıdaki denklemleri birleştirecek olursak

$$P(x|Z^t)P(Z^t) = P(Z^{t-1}|x)P(z_t|x)P(x)$$

$P(Z^{t-1}|x)$ için Bayes teoremini uygulayalım.

$$P(Z^{t-1}|x) = \frac{P(x|Z^{t-1})P(Z^{t-1})}{P(x)}$$

Denklemler birleştirildiğinde,

$$P(x|Z^t)P(Z^t) = P(z_t|x)P(x|Z^{t-1})P(Z^{t-1})$$

$$P(x|Z^t) = \frac{P(z_t|x)P(x|Z^{t-1})P(Z^{t-1})}{P(Z^t)}$$

$$P(z_t|Z^{t-1}) = \frac{P(z_t, Z^{t-1})}{P(Z^{t-1})}$$

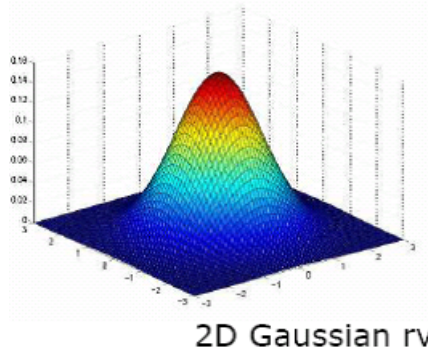
$$P(z_t|Z^{t-1}) = \frac{P(Z^t)}{P(Z^{t-1})}$$

Sonuç olarak istediğimiz sonucu elde ederiz.

$$P(x|Z^t) = \frac{P(z_t|x)P(x|Z^{t-1})}{P(z_t, Z^{t-1})}$$

Bu sonuç neyi ifade ediyor. Amacımız t adımı dahil bütün adımlardaki gözlemlere bağımlı x'in koşullu dağılımı $P(x|Z^t)$ 'yi bulmak. $P(x|Z^t)(t-1)$ adımdaki gözlemlerle koşullu en son en iyi kestirilen x'in öncül dağılımını ifade ediyor. Bayes teoremi, yeni bilgilerin o anki olasılık fonksiyonu ile öncül x'in koşullu dağılımıyla çarpılarak basit bir şekilde eklenebildiği için çok güçlü bir kestirim mekanizması olarak bilinir.

Eğer x çok değişkenli rastgele değişkense;



Şekil 5.4: İki boyutlu gaussian değişkenin dağılım grafiği.

Kovaryans matrisi P ;

$$P = \begin{bmatrix} P_{11} & P_{12} & \cdot & \cdot & P_{1n} \\ P_{21} & P_{22} & \cdot & \cdot & P_{2n} \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ P_{n1} & P_{n2} & \cdot & \cdot & P_{nn} \end{bmatrix}$$

5.3.1 Gözlem modeli hesaplanması

Mobil robottaki lazerle yapılan ölçümler lazer ve işaretçi nesne arasındaki mesafe r_i ve lazerle işaretçi nesne arasındaki x eksenine göre olan açı θ_i 'yi vermektedir. Ölçüm vektörü r_i ve θ_i bilgilerini veren iki elemanlı bir vektördür.

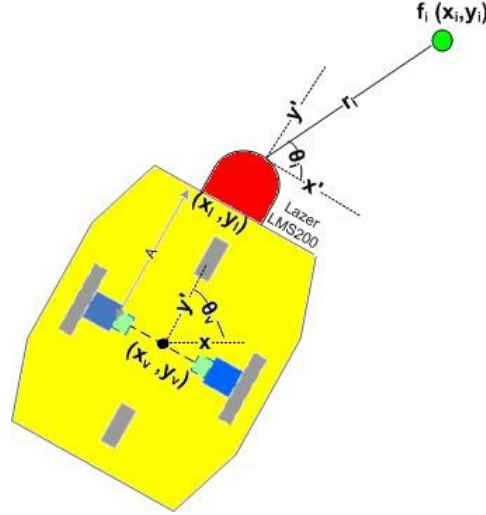
$$(X_1, Y_1) = (X_v, Y_v + A)$$

$$Z(t) = \begin{bmatrix} r_i \\ \theta_i \end{bmatrix}$$

$$Z(t) = h(x(t), w(t))$$

Son denklemde verilen gözlem modeli işaretçi nesne ve lazer arasındaki geometrik ilişkinin bir ifadesidir. Gözlem modeli elde edilen r_i ve θ_i ölçümlerinden işaretçi nesnenin x ve y eksenindeki değerleri X_i ve Y_i 'yi vermektedir. X_i ve Y_i değerleri ise aşağıdaki denklem ile hesaplanmaktadır. Bu sebeple, aşağıdaki denklem sistemin gözlem modelidir.

$$Z(t) = \begin{bmatrix} \sqrt{(X_i - X_1(t))^2 + (Y_i - Y_1(t))^2} \\ a \tan 2(Y_i - Y_1(t)/X_i - X_1(t)) - \theta_v \end{bmatrix}$$



Şekil 5.5: Mobil robotun lazerle gözlem yapması.

Yukarıda verilen denklemi cebirsel olarak çözüp X_i ve Y_i değerlerinin matematiksel olarak ifadesini bulmak gereksizdir. İki nokta arasındaki mesafe ve açı değerlerinden x ve y eksenlerindeki değişim bu sayede bulunabilmektedir.

$$\nabla H_{x_1} = \frac{\sigma h}{\sigma_x} = \begin{bmatrix} \frac{\sigma r}{\sigma_{x_1}} & \frac{\sigma r}{\sigma_{y_1}} & \frac{\sigma r}{\sigma_{\theta_1}} \\ \frac{\sigma \theta}{\sigma_{x_1}} & \frac{\sigma \theta}{\sigma_{y_1}} & \frac{\sigma \theta}{\sigma_{\theta_1}} \end{bmatrix}$$

$$\nabla H_{x_1} = \frac{\sigma h}{\sigma_x} = \begin{bmatrix} \frac{X_i - X_1(t)}{Y_i - Y_1(t)} & \frac{Y_i - Y_1(t)}{X_i - X_1(t)} & 0 \\ \frac{r}{r^2} & -\frac{r}{r^2} & -1 \end{bmatrix}$$

$$\nabla H_{x_f} = \frac{\sigma h}{\sigma_{x_f}} = \begin{bmatrix} \frac{\sigma r}{\sigma_{x_i}} & \frac{\sigma r}{\sigma_{y_i}} \\ \frac{\sigma \theta}{\sigma_{x_i}} & \frac{\sigma \theta}{\sigma_{y_i}} \end{bmatrix}$$

$$\nabla H_{x_f} = \frac{\sigma h}{\sigma_{x_f}} = \begin{bmatrix} \frac{X_i - X_1(t)}{Y_i - Y_1(t)} & \frac{Y_i - Y_1(t)}{X_i - X_1(t)} \\ -\frac{r}{r^2} & \frac{r}{r^2} \end{bmatrix}$$

$$\nabla H_x = \begin{bmatrix} \nabla H_{x_1} & 0 \\ 0 & \nabla H_{x_f} \end{bmatrix}$$

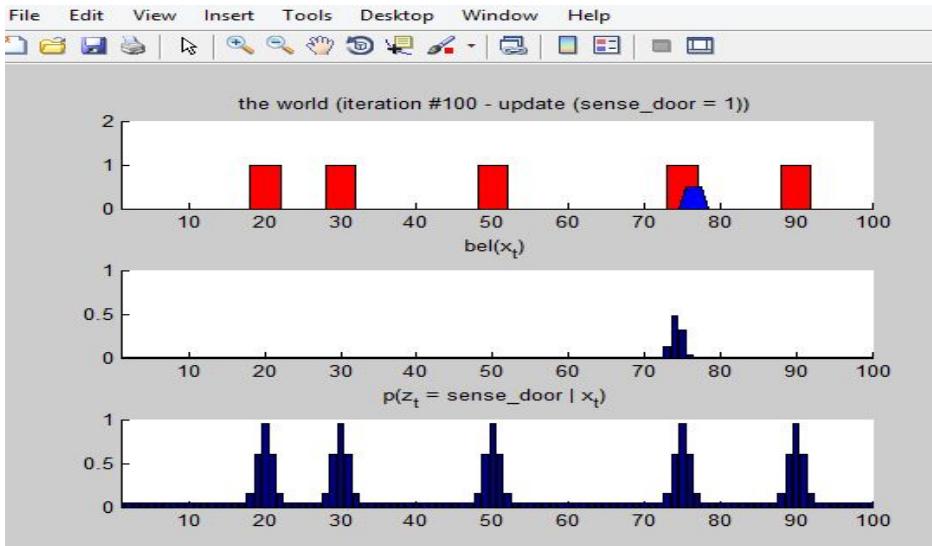
Gözlem modelinin gürültü kovaryans matrisi denklem oluşur. Burada lazer mesafe gürültüsü ve açı gürültüsü arasında herhangi bir korelasyon bulunmadığı ve hatalarında gaussian olduğu varsayılmıştır. R matrisinin doğru bulunabilmesi için lazerin doğru modellenmesi gerekmektedir. Simulasyonda bu değerler rastgele seçilmektedir. Gerçek zamanlı uygulamalarda lazerle mesafe ve açı ölçümleri yapılarak gerçek

değerlerinden ne kadar saptıkları belirlenmelidir. Bu hatalardan elde edilen dağılımın standart sapmaları hesaplanarak R kovaryans matrisi oluşturulacaktır.

5.4 Temel Yaklaşım

Bu tez çalışmasında konu ile ilgili çalışmalar etap etap yapılmıştır. İlk olarak Markov konumlama yönteminin temelleri esas alınarak Matlab ortamında yazılım oluşturulmuş ve daha sonra bu işlemin bir simülasyonu gerçekleştirilmiştir. Daha sonra odometrik veriler kullanılarak mobil robotun belirlenen rota üzerinde hareket etmesi istenmiş ve bununla ilgili çalışmalar yapılmıştır. Sonra Markov konumlama yöntemi, odometrik veriler ve sensörden alınan bilgiler ışığında farklı filtreleme yöntemlerinin uygulanması neticesinde konumlama işlemi gerçekleştirilmiştir.

Markov konumlama yönteminin daha iyi bir şekilde anlaşılması için bölüm üçte anlatılan çalışma gerçekleştirilmeye çalışılmıştır. Burada robotun tek yönlü ilerlemesi sağlanmıştır. Önceden olasılıksal dağılım fonksiyonları ile yeri belirlenmiş kapıları robotun hareketi yönünde fark edince inanışında meydana gelen değişimler gözlenmek istenmiştir. Bu yönde önceden dağılım fonksiyonu ile belirlenmiş yapıyı fark edince robotun inanışında nasıl bir değişim olduğu görülmüştür. Bunun için MATLAB programından yararlanılarak bir simülasyon gerçekleştirilmiştir.



Şekil 5.6: MATLAB uygulaması.

Şekil 5.6'da görüldüğü üzere tanınan cisimler 20,30,50,75,90 konumları ile tanımlanmıştır. İstenirse burada tanımlanan cisimlerin konumları değiştirilebilir. Belirli sayıda

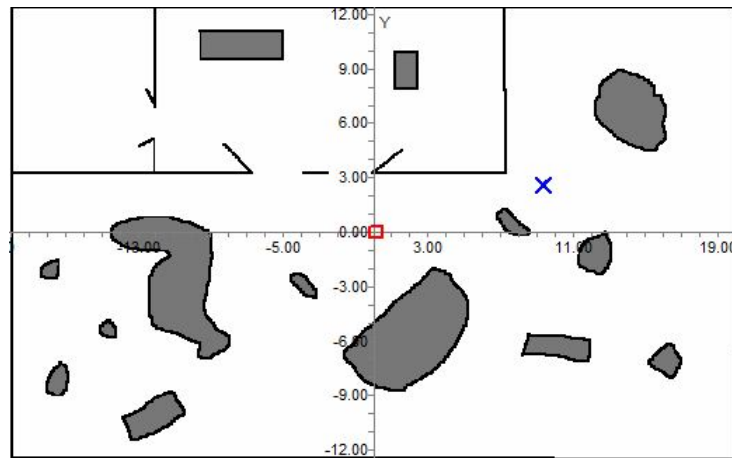
bir döngüde program yürütülmüştür. Belirlenen sayıda iterasyon sağlanmış ve markov konumlama işleminin nasıl gerçekleştiği kavranmıştır.

Markov konumlama işleminin gerçekleştirmek için MATLAB, ROS ve Mobil robot programlama aracından yararlanılmıştır. ROS, içerisinde barındırdığı kütüphaneler ve araçlarla robot uygulamaları geliştirmeyi sağlayan bir yazılım sistemidir. ROS, açık kaynak kodlu bir yazılımdır. ROS'u kullanabilmeniz için Linux tabanlı bir işletim sistemine ihtiyacınız olacaktır. ROS, yayıncı/alıcı paradigması üzerine kuruludur. Hazırlanan robot davranışları node adı verilen program parçaları üzerinden simülasyon programına gönderilir (yayın), hareketi için gerekli verileri çevreden alınır (alıcı). Mobil robot programlama aracı ise ROS ile etkileşimli çalışabilecek çeşitli dillerde yazılmış kütüphanelerin yanı sıra çeşitli sensörlerin kullanılmasını destekleyen bir yazılımdır.

5.5 Uygulama Verileri

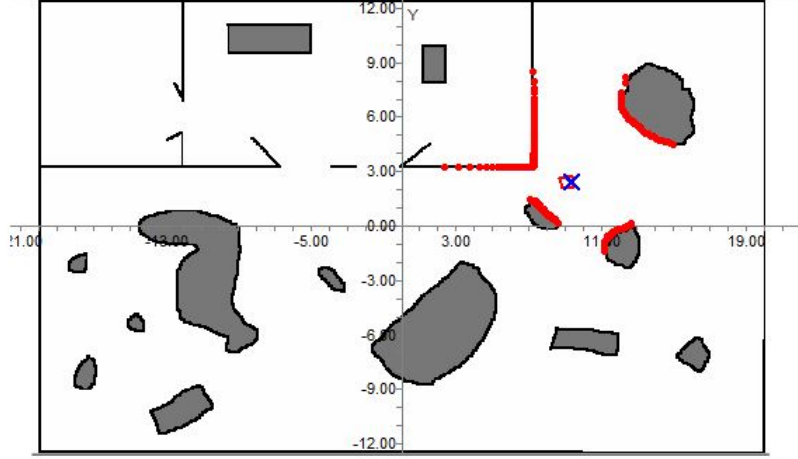
Markov konumlama yönteminin uygulanması için bir senaryo oluşturulmuştur. Burada mobil robotumuz iç ortamlarda engellerden sakınarak belirlenen hedefe en iyi şekilde ulaşması amaçlanan bir senaryo oluşturuldu. Belirlenen senaryo gereği robota gitmesi gereken bir hedef noktası verilmiştir. Noktaya ilerlerken oluşan değişim gözlenmiştir.

Şekil 5.7'de görüldüğü üzere robot (0,0) noktası durmaktadır. Mobil robotun hareketi için (0,0) noktası başlangıç noktası olarak kabul edilmiştir. Mobil robota ulaşması gereken hedef nokta bilgisi verilir ve robotun bu noktaya ulaşması sağlanır.



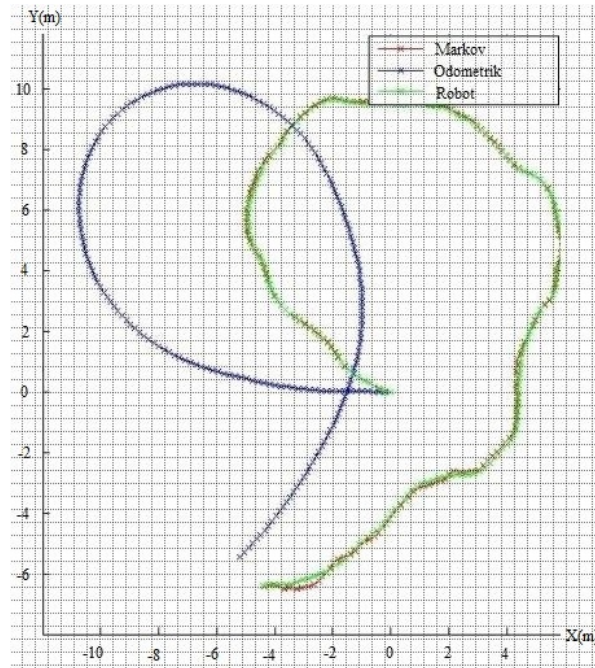
Şekil 5.7: Mobil robotun konumu ve belirlenen hedef noktası.

Şekil 5.8’de görüldüğü üzere mobil robotumuz istenilen noktaya yönelim sağlamış ve verilen hedef koordinatlarına ulaşmayı başarmıştır. Burada mobil robotumuzun belirlenen hedef noktaya ulaşması için odometrik yöntem kullanılmıştır. Bunun neticesinde elde edilen x ve y yönündeki parametreler değerlendirilmiştir.



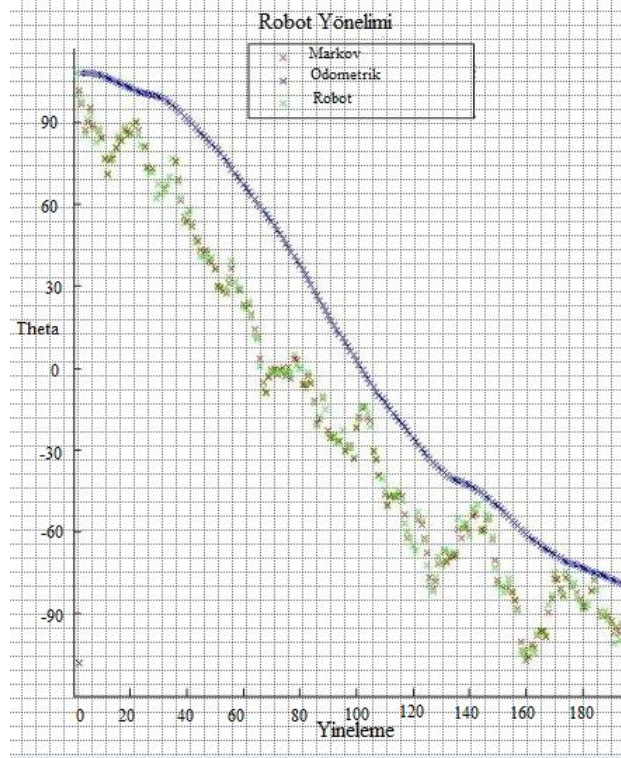
Şekil 5.8: Mobil robotun konumu ve belirlenen hedefe ulaşması.

Oluşturulan senaryo temel olarak uygulandıktan sonra, mobil robot markov yöntemi ve odometrik verilerin analizi neticesinde nasıl bir davranış sergilediği gözlemlenmiştir. Odometrik veriler ile markov konumlama yönteminin analizleri yapılarak bu iki yöntem grafiksel olarak kıyaslanmıştır.



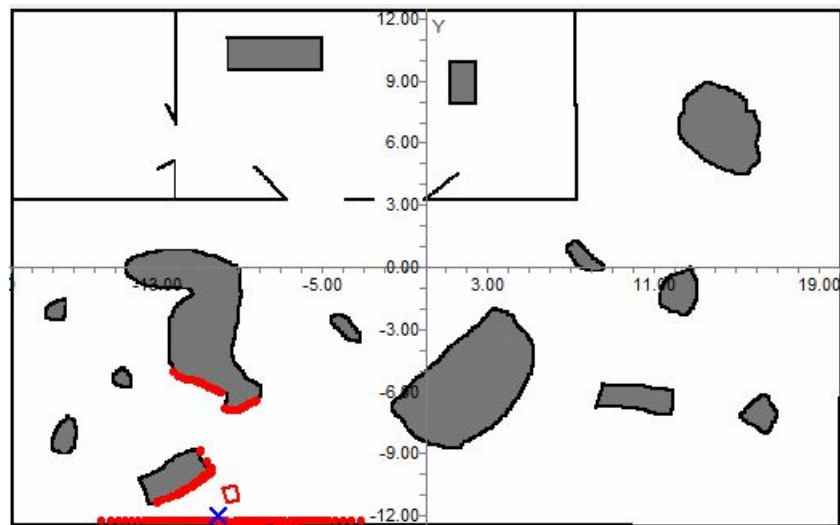
Şekil 5.9: X-Y düzleminde mobil robot konumlanması.

X-Y düzleminde robotun hareketi sonucu analizler yapılmıştır. Bu analizler mobil robotun x ve y konum değerlerindeki değişimler şekil 5.9’da görüldüğü şekilde elde edilmiştir. Ayrıca yönelim açısı olan θ açısı da incelenmiştir.



Şekil 5.10: Robotun yönelimi.

Şekil 5.10’da robotun θ açısının x ve y konumlarındaki parametrelere bağlı olarak değişimi gösterilmiştir.

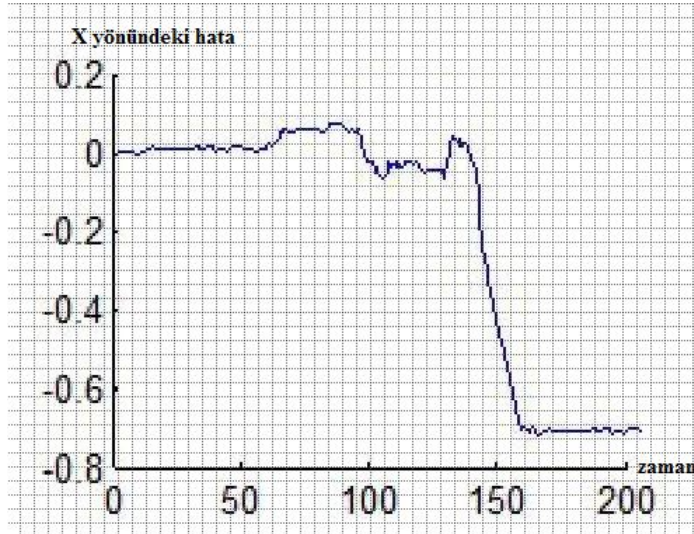


Şekil 5.11: Mobil robotun hareketi.

5.6 Markov Konumlama Yönteminin Sonuçları

Bu bölümde uygulanan senaryo çalışması neticesinde elde edilen veriler grafiksel olarak sunulmuştur. Burada birbirinden farklı üç yönelim uygulanmıştır. İlk olarak odometrik yöntem kullanılarak konumlama işlemi gerçekleştirilmiş ve yapılan çalışma sonucunda elde edilen sonuçlar gösterilmiştir. Daha sonra klasik markov konumlama işlemi gerçekleştirilmiştir. Bu işlem neticesinde elde edilen veriler analiz edilmiştir. Son olarak gaussian filtreleme kullanılarak markov konumlama yaklaşımı yapılmış ve mobil robotun bu yöntem neticesinde konumlamanın gerçekleştirilmesi incelenmiştir.

Şekil 5.12’de x düzlemindeki hataların analizi sonucunda elde edilen veri paylaşılmıştır. Burada dikey sütun x düzlemindeki hata oranını metre cinsinden tespit etmektedir. Yatay sütun ise hedeflenen noktaya varıncaya kadar geçen süreyi saniye cinsinden ifade etmektedir. Görüldüğü üzere mobil robot hareketin 100. saniyelerinde 0.1 civarında olan hata oranı hareketin sonlarına doğru artarak -0.7 ye yükselmiştir. Buradaki eksi ifadesi hareketin pozitif kabul edilen yönün tersine olduğunu ifade eder.



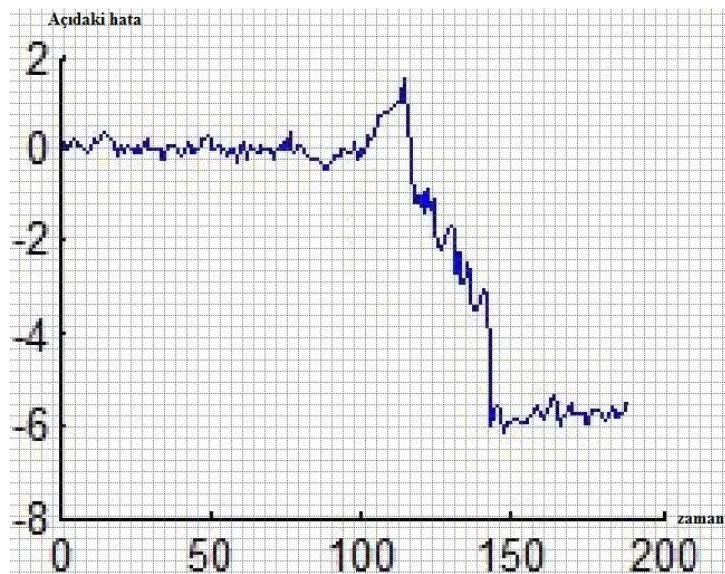
Şekil 5.12: X yönündeki hata oranı.

Şekil 5.13’te y düzlemindeki hataların analizi sonucunda elde edilen veri paylaşılmıştır. Burada dikey sütun y düzlemindeki hata oranını metre cinsinden tespit etmektedir. Yatay sütun ise hedeflenen noktaya varıncaya kadar geçen süreyi saniye cinsinden ifade etmektedir. Görüldüğü üzere x düzlemindeki hata oranına paralel olarak y düzlemindeki hata oranı da 100. saniyeye kadar olağan giderken, ilerki zamanda hata oranının giderek arttığı gözlenmiştir.



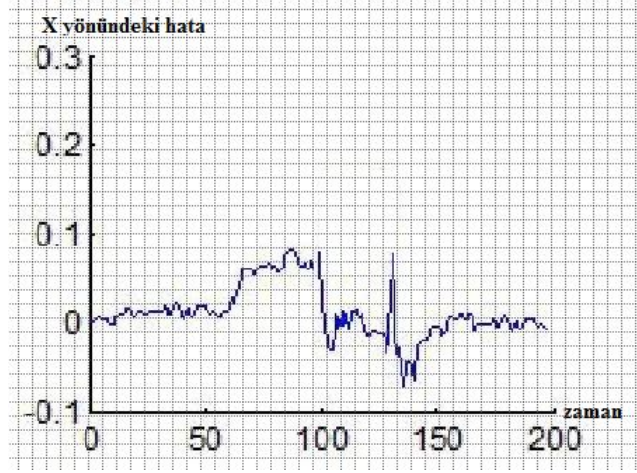
Şekil 5.13: Y yönündeki hata oranı.

Şekil 5.14’de θ açısındaki hataların analizi sonucunda elde edilen veri paylaşılmıştır. Burada dikey sütun θ açısındaki hata oranını derece cinsinden tespit etmektedir. Yatay sütun ise hedeflenen noktaya varıncaya kadar geçen süreyi saniye cinsinden ifade etmektedir. Burada da x ve y parametreleri için elde edilen sonuçlar ile bir paralellik söz konusudur. Hareketin belirli bir aşamasından sonra hata oranı giderek yükselmiştir. θ açısındaki hata -2 derecelerden -6 derecelere kadar büyümüştür. Eğer hareketin devamlılığı sağlansa hatada artacağında robot istenilen konumdan oldukça sapacak ve hedefe ulaştığı düşünülen mobil robot hedefin çok uzağında kalacaktır. Bu sonuçlar neticesinde hedeflenen noktaya ulaşmak pek mümkün olmamıştır.



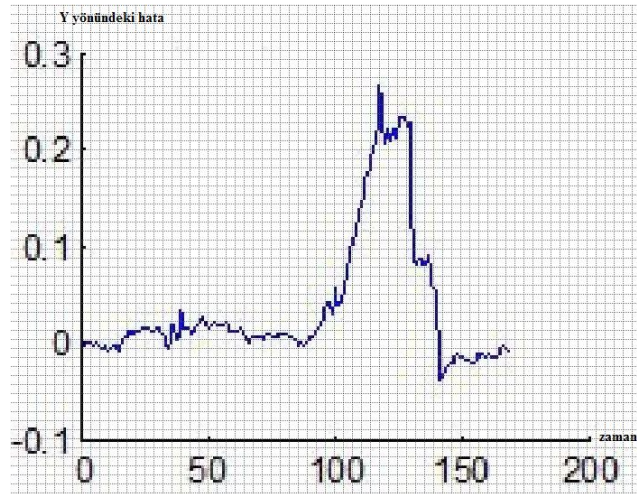
Şekil 5.14: θ açısındaki hata oranı.

Grafiklerden de anlaşılacağı gibi hareketin ilk anlarında mobil robot hataları minimum seviyede iken son anlara yaklaşıldığında hatalar giderek büyümektedir. İlk yaklaşımdan sonra klasik markov konumlama yöntemi kullanılarak, aynı hedef koordinat noktası sisteme girilmiştir. Sürekli olarak aynı hedef nokta girilerek değişken parametrelerin azaltılması öngörülmüştür. Şekil 5.15'te dikey ve yatay sütun ifadeleri bir önceki yaklaşım ile aynıdır. Şekil 5.15'te görüldüğü üzere x yönündeki hata oranı belirli bir süre minimum seviyede seyretmiş. 100. saniyeden sonra hata arttırsa bile hareketin ilerleyen zamanlarında hata oranı sıfıra yakınsamaya çalışır.



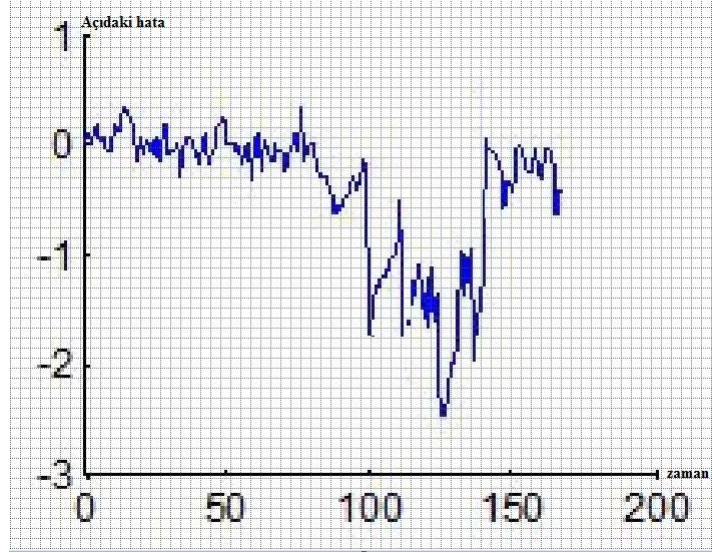
Şekil 5.15: X yönündeki hata oranı.

Şekil 5.16'da dikey sütun ve yatay sütun parametreleri aynıdır. Şekil 5.16'da 100. saniyedeki değişim etkili olmuştur. Burada da belirtilen zamandan sonra hata oranı artmış olsa dahi hareketin ilerleyen aşamasında hata oranı sıfıra yakınsar.



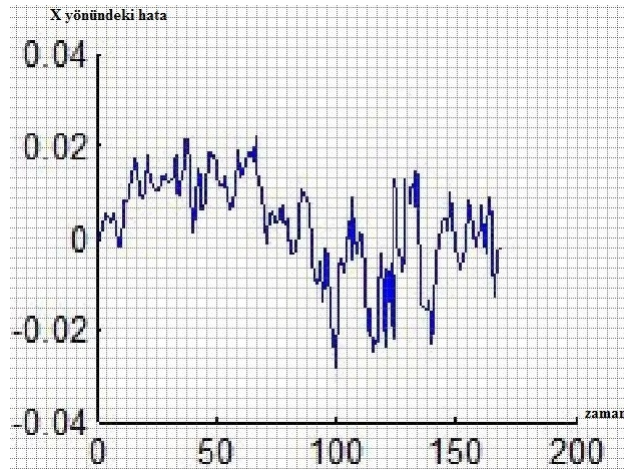
Şekil 5.16: Y yönündeki hata oranı.

Şekil 5.17’de dikey ve yatay sütun parametreleri değişmemektedir. 100.saniye de meydana gelen hata artma refleksi burada da görülmektedir. Mobil robotun hareketi ilerletmesi ile 140. saniyelerden sonra hata hızla azalmaktadır.



Şekil 5.17: θ açısındaki hata oranı.

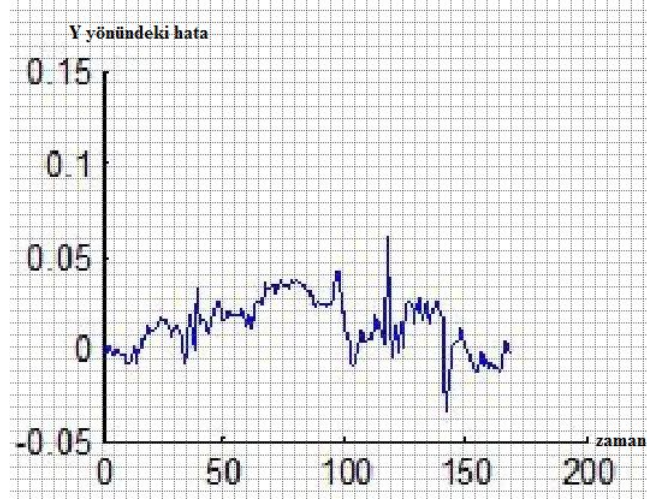
Mobil robotun yapısı ve uygulanan alan koşulları göz önüne alındığında sistemde gerekli iyileştirmeler yapıp daha düzenli bir metod geliştirilmiştir. Markov yöntemi bu inanış ve olasılıksal dağılımları gerçek zamanlı olarak değiştirip mobil robotun konumlama sistemini kuvvetlendirmiştir. Yöntemin geliştirilmiş filtre işlemleri ile daha da iyileştirilmesi mümkün olabileceği düşünülmüştür. Son olarak da odometrik veriler ve markov yönteminin beraber kullanıldığı ve filtrelemenin farklılaştırıldığı konumlama işleminde elde edilen sonuçlara yer verilmiştir.



Şekil 5.18: X yönündeki hata oranı.

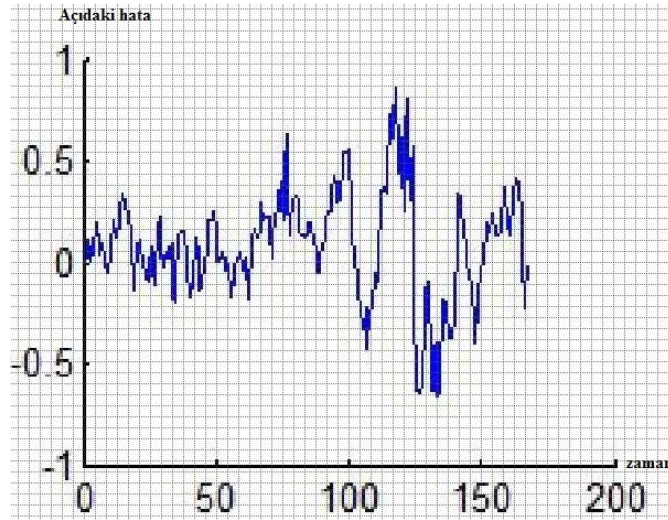
Şekil 5.18’de dikey ve yatay sütun parametreleri değişmemektedir. X yönündeki hata oranı her ne kadar dalgalı olsa da sürekli olarak sıfır değerine yakın seyretmiştir. Bu sistemde yapılan iyileştirmelerin sonuç verdiğini göstermektedir.

Şekil 5.19’da dikey ve yatay sütun parametreleri aynıdır. Burada x yönündeki hata oranı ile paralellik söz konusudur. Hata oranının sıfıra yakın değerlerde tutulabilmesi konumlama işleminin gerçeğe yakın yapıldığını göstermektedir.



Şekil 5.19: Y yönündeki hata oranı.

Son olarak Şekil 5.20’de aynı parametreler kullanılarak sonuç elde edilmiştir. 100. saniyede meydana gelen hatanın artma refleksi burada da gözükse de sistem gayet iyi bir şekilde mobil robotu yönlendirmiştir. Yapılan iyileştirme ve filtrelemede ki farklılık son iki yaklaşım arasında gayet net görülmektedir.



Şekil 5.20: θ açısındaki hata oranı.

En son olarak uygulanan üç yöntemin hata, yakınsama ve hız açısından kıyaslanması Çizelge 5.1’de gösterilmiştir.

Çizelge 5.1: Uygulanan yöntemlerin kıyaslanması

Yöntem	Hata	Yakınsama	Hız
Yöntem1	Çok	Kötü	Çok
Yöntem2	Normal	İyi	Normal
Yöntem3	Az	İyi	Az

6. SONUÇ VE ÖNERİLER

Bu tez çalışmasında mobil robotlar için genellikle önerilen odometrik konumlama yöntemi yanı sıra olasılıksal yaklaşımlardan biri olan Markov konumlama yöntemi kullanılarak iç ortamlarda robotun gezinim yapılması sağlanmıştır. Mobil robot gezinim yaparken engellerden kurtulmayı ve hedeflenen yolu izleyerek görevi tamamlamıştır. Kullanılan bu yöntemi ilerleeyen aşamalarda robotun ilk konumunu bilmeden ve ani konum değişikliklerini robota bildirmeden aktif konumlama işleminin başarı ile yapılması hedeflenmiştir. Ayrıca Markov olasılıksal konumlama yöntemi kullanılarak birden fazla robotun iletişim halinde olduğu ve birbirine çarpmadan olasılıksal konumlama yapabilmeleri hedeflenebilir. Kullanılan odometrik yöntem haricinde dışarda kullanılan aktif sensörler başka tür sensörler ile mukayese edilebilir.

KAYNAKLAR

- [1] **Siciliano, B. ve Khatib, O.** (2008). *Springer handbook of robotics*, Springer Science & Business Media.
- [2] **Thrun, S., Burgard, W. ve Fox, D.** (2005). *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*, The MIT Press.
- [3] LMS200/211/221/291 Laser Measurement Systems, <http://sicktoolbox.sourceforge.net/docs/sick-lms-technical-description.pdf>, alındığı tarih: 20.04.2015.
- [4] **Borenstein, J., Everett, H., Feng, L. ve diğerleri** (1996). Where am I? Sensors and methods for mobile robot positioning, *University of Michigan*, **119**(120), 27.
- [5] **Negenborn, R.** (2003). Robot localization and kalman filters, *Doktora Tezi*, CiteSeer.
- [6] **Duchoň, F., Babinec, A., Rodina, J., Fico, T. ve Hubinský, P.** (2014). Probabilistic Approach to Mobile Robot Localization Based on Gaussian Models of Sensors, *Applied Mechanics and Materials*, cilt607, Trans Tech Publ, s.803–810.
- [7] **Gopalakrishnan, A., Greene, S. ve Sekmen, A.** (2005). Vision-based mobile robot learning and navigation, *Robot and Human Interactive Communication, 2005. ROMAN 2005. IEEE International Workshop on*, IEEE, s.48–53.
- [8] **Ivanjko, E. ve Petrovic, I.** (2004). Extended Kalman filter based mobile robot pose tracking using occupancy grid maps, *Electrotechnical Conference, 2004. MELECON 2004. Proceedings of the 12th IEEE Mediterranean*, cilt 1, IEEE, s.311–314.
- [9] **Fox, D., Burgard, W. ve Thrun, S.** (1999). Markov localization for mobile robots in dynamic environments, *Journal of Artificial Intelligence Research*, 391–427.
- [10] **Thrun, S. ve Bücken, A.** (1996). Integrating grid-based and topological maps for mobile robot navigation, *Proceedings of the National Conference on Artificial Intelligence*, s.944–951.

- [11] **Leonard, J.J. ve Durrant-Whyte, H.F.** (1991). Mobile robot localization by tracking geometric beacons, *Robotics and Automation, IEEE Transactions on*, 7(3), 376–382.
- [12] **Nourbakhsh, I., Powers, R. ve Birchfield, S.** (1995). DERVISH an office-navigating robot, *AI magazine*, 16(2), 53.
- [13] **Simmons, R. ve Koenig, S.** (1995). Probabilistic robot navigation in partially observable environments, *IJCAI*, cilt 95, s.1080–1087.

ÖZGEÇMİŞ

Ad Soyad:Mustafa Tanış

Doğum Yeri ve Tarihi:Silvan-28.08.1988

Adres:

E-Posta:mtanis@itu.edu.tr

Lisans:İnönü Üniversitesi Elektrik-Elektronik Mühendisliği

