T.C.
ERCİYES ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
BİLGİSAYAR MÜHENDSİLİĞİ ANABİLİM DALI

YAPAY ARI KOLONİSİ (ABC) ALGORİTMASI İLE ROBOTİK YOL PLANLAMA

Hazırlayan
Ousainou NYASSİ

Danışman
Dr. Öğr. Üyesi Beyza GÖRKEMLİ

Yüksek Lisans Tezi

TEMMUZ 2019
KAYSERİ

T.C.
ERCİYES ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
BİLGİSAYAR MÜHENDSİLİĞİ ANABİLİM DALI

YAPAY ARI KOLONİSİ (ABC) ALGORİTMASI İLE ROBOTİK YOL PLANLAMA

Hazırlayan
Ousainou NYASSİ

Danışman
Dr. Öğr. Üyesi Beyza GÖRKEMLİ

Bu çalışma, Erciyes Üniversitesi Bilimsel Araştırma Projeleri Birimi tarafından FYL-2018-7941 kodlu proje ile desteklenmiştir.

TEMMUZ 2019
KAYSERİ

BİLİMSEL ETİĞE UYGUNLUK

Bu çalışmadaki tüm bilgilerin, akademik ve etik kurallara uygun bir şekilde elde edildiğini beyan ederim. Aynı zamanda bu kural ve davranışların gerektirdiği gibi, bu çalışmanın özünde olmayan tüm materyal ve sonuçları tam olarak aktardığımı ve referans gösterdiğimi belirtirim.



Ousainou Nyassi

İmza

YÖNERGEYE UYGUNLUK

"YAPAY ARI KOLONİSİ (ABC) ALGORİTMASI İLE ROBOTİK YOL PLANLAMA" adlı Yüksek Lisans tezi, Erciyes Üniversitesi Lisansüstü Tez Önerisi ve Tez Yazma Yönergesi' ne uygun olarak hazırlanmıştır.

Hazırlayan

Ousainou NYASSİ

İmza

Danışman

Dr. Öğr. Üyesi Beyza GÖRKEMLİ

İmza

Bilgisayar Mühendisliği ABD Başkanı

Prof. Dr. Veysel ASLANTAŞ

İmza

KABUL VE ONAY

Dr. Öğr. Üyesi Beyza GÖRKEMLİ danışmanlığında **Ousainou NYASSİ** tarafından hazırlanan **"YAPAY ARI KOLONİSİ (ABC) ALGORİTMASI İLE ROBOTİK YOL PLANLAMA"** adlı bu çalışma jürimiz tarafından Erciyes Üniversitesi Fen Bilimleri Enstitüsü **Bilgisayar Mühendisliği** Anabilim Dalında **Yüksek Lisans** tezi olarak kabul edilmiştir.

26/07/2019
Gün / Ay / Yıl

JÜRİ:

Danışman : Dr. Öğr. Üyesi Beyza GÖRKEMLİ .....

Üye : Doç. Dr. Barış KOÇER .....

Üye : Doç. Dr. Celal ÖZTÜRK .....

ONAY:

Bu tezin kabulü Enstitü Yönetim Kurulunun 06/08/2019 tarih ve 2019/57-09 sayılı kararı ile onaylanmıştır.

06 / 08 / 2019

Prof. Dr. Mehmet AKKURT

Enstitü Müdürü

TEŞEKKÜR

Bu çalışmaların başarılı bir şekilde tamamlanması için gösterdiği özveri, öneri, içgörüsel yardım ve sayısız destek için akademik danışmanım Dr. Öğr. Üyesi Beyza GÖRKEMLİ'ye teşekkür ediyorum. Arkadaşlarım ve meslektaşlarım Pa Amat, Malang, Fateh, Malick, Ibrahim, Ahmed ve Youssof'a sayısız destekleri için teşekkür ediyorum. Bu hayırlı yolculuk boyunca bana inanıp destek verdikleri için mütevazi ebeveynlerime ve aileme teşekkür etmek isterim.

Ousainou NYASSİ

07 2019, KAYSERİ

YAPAY ARI KOLONİSİ (ABC) ALGORİTMASI İLE ROBOTİK YOL PLANLAMA

Ousainou NYASSİ

Erciyes Üniversitesi, Fen Bilimleri Enstitüsü
Yüksek Lisans Tezi, TEMMUZ 2019
Danışman: Dr. Öğr. Üyesi Beyza GÖRKEMLİ

ÖZET

Mobil robotların farklı alanlarda kullanımı her geçen gün artmaktadır. Özellikle, robotların dışarıdan bir müdahale olmadan ve engellere çarpmaksızın hedefe ulaşmasını sağlayan sistemler birçok uygulama alanında maliyetin düşürülmesi, karın artırılması, insanlar için tehlikeli durumların azaltılması, vb. noktalarda kritik bir öneme sahiptir. Yol planlamasını tüm haritanın bilindiği küresel yol planlama (çevrim dışı-offline planlama) ya da robotun tüm alan bilgisine sahip olmadığı, sadece sensörleri ile algılayabildiği kısıtlı bir alana dair bilgiye sahip olduğu bölgesel (çevrim içi-online) yol planlaması olarak ele almak mümkündür. Bu çalışmada, 2 boyutlu uzayda bir başlangıç noktası, durağan bir hedef ve durağan elips şeklinde engellerin bulunduğu ortamlar dikkate alınarak bir hareketli robot için bölgesel yol planlama temelli yaklaşımlar ile robotun engellere çarpmadan başlangıçtan hedef noktasına ulaşması sağlanmaya çalışılmıştır. Bunu yaparken popüler ve başarılı bir optimizasyon algoritması olan yapay arı kolonisi (artificial bee clolony-ABC) algoritmasına dayalı yaklaşımlar geliştirilmiştir. Bu yaklaşımlardan biri standart ABC'yi kullanmakta iken diğer yaklaşım hızlı ABC (quick ABC-qABC)'yi kullanmaktadır. Önerilen yaklaşımlar farklı senaryolarda test edilerek performansları karşılaştırılmıştır. Simülasyon sonuçları bu yöntemlerin dikkate alınan senaryolar için başarılı sonuçlar ürettiğini göstermektedir.

Anahtar Kelimeler: robotik yol planlama, 2 boyutlu uzay, bölgesel (çevrim içi-online) yol planlama, durağan engel, tek robot, yapay arı kolonisi (ABC), hızlı yapay arı kolonisi (qABC)

ROBOTIC PATH PLANNING USING ARTIFICIAL BEE COLONY(ABC) ALGORITHM

Ousainou NYASSI

Erciyes University, Graduate School of Natural and Applied Sciences
Master Thesis, July 2019
Supervisor: Dr. Beyza GÖRKEMLİ

ABSTRACT

The use of mobile robots in different areas is increasing day by day. In particular, systems that allow robots to reach the target without external interference and without hitting obstacles can reduce costs, increase profits, reduce hazardous situations for humans on critical conditions and so on. It is possible to consider path planning as global path planning (offline planning) where the entire map is known, or local (online) path planning where the robot does not have all the field information, but only a limited area that it can detect with sensors. In this study, considering the environment is in a 2-dimensional space where there is a starting point, a static target point and static ellipse-shaped obstacles, this technique tries to provide the robot to reach the target point from the beginning without hitting the obstacles by using local path planning based approaches for a mobile robot. In doing so, approaches based on the artificial bee colony(ABC) algorithm, which is a popular and successful optimization algorithm, have been developed. One approach uses standard ABC, while the other approach uses quick ABC(qABC). The proposed approaches were tested in different scenarios and their performances were compared. Simulation results shows that these methods produce successful results for the scenarios considered.

Keywords: robotic path planning, 2-dimensional space, local (online) path planning, static obstacles, single robot, artificial bee colony (ABC), quick artificial bee colony (qABC)

İÇİNDEKİLER

YAPAY ARI KOLONİSİ (ABC) ALGORİTMASI İLE ROBOTİK YOL PLANLAMA

BİLİMSEL ETİĞE UYGUNLUK ii

YÖNERGEYE UYGUNLUK iii

KABUL VE ONAY iv

ÖNSÖZ v

ÖZET vi

ABSTRACT vii

İÇİNDEKİLER viii

KISALTMALAR x

TABLOLAR LİSTESİ xi

ŞEKİLLER LİSTESİ xii

GİRİŞ 1

1. BÖLÜM

GENEL BİLGİLER ve LİTERATÜR ÇALIŞMASI

1.1. Mobil Robotik Yol Planlama Problemi **4**

1.2. Robotik Yol Planlama Algoritmaları **5**

1.3. Mobil Robot Yolu Planlama Kategorileri **6**

1.3.1 Küresel Yol Planlaması (Çevrimdışı) **6**

1.3.2 Bölgesel Yol Planlama (Çevrimiçi) **6**

1.4. Literatür Çalışması **10**

2. BÖLÜM

YÖNTEM

2.1. ABC 14
2.2. qABC 18
2.3. Önerilen Yaklaşımlar 19
2.3.1 Önerilen Algoritma Yapısı ve Yöntemleri 21
2.3.2 Engeller ile Çarpışmayı Önleme 23
2.3.3 Amaç Fonksiyonu 24

3. BÖLÜM

DENEYSEL ÇALIŞMALAR VE TARTIŞMA

3.1. Deneysel Çalışmalar 26
3.2. ABC ve qABC Algoritmalarının Kontrol Parametre Değerleri 27
3.3. Ele Alınan Senaryolar ve Simülasyon Sonuçları..... 27
3.4. Tartışma 30

4. BÖLÜM

SONUÇ ve ÖNERİLER

4.1. Sonuç ve Öneriler..... 43

KAYNAKÇA..... 45

ÖZGEÇMİŞ 48

KISALTMALAR

ABC: Yapay Arı Kolonisi (Artificial Bee Colony)

MRYP: Mobil Robot Yolu Planlama

BYP: Bölgesel Yol Planlama

GPS: Global Konumlandırma Sistemi (Global Positioning System)

qABC: Hızlı Yapay Arı Kolonisi (quick Artificial Bee Colony)

GA: Genetik Algoritma (Genetic Algorithm)

ACO: Karınca Koloni Optimizasyonu (Ant Colony Optimization)

PSO: Parçacık Sürü Optimizasyonu (Particle Swarm Optimization)

DE: Diferansiyel Gelişim (Differential Evolution)

EA: Evrimsel Algoritma

TABLOLAR LİSTESİ

Tablo 3.1. ABC ve qABC algoritmalarının kontrol parametre değerleri..... 27
Tablo 3.2. Senaryo 1 için parametre değerleri 27
Tablo 3.3. 30 Çalıştırmada Gezilen MRYP Optimal Mesafeleri (Senaryo 1) 30
Tablo 3.4. Senaryo 2 için parametre değerleri 30
Tablo 3.5. 30 Çalıştırmada Gezilen MRYP Optimal Mesafeleri (Senaryo 2) 33
Tablo 3.6. Senaryo 3 için parametre değerleri 33
Tablo 3.7. 30 Çalıştırmada Gezilen MRYP Optimal Mesafeleri (Senaryo 3) 36
Tablo 3.8. Senaryo 4 için parametre değerleri 36
Tablo 3.9. 30 Çalıştırmada Gezilen MRYP Optimal Mesafeleri (Senaryo 4) 39
Tablo 3.10. Senaryo 5 için parametre değerleri 39
Tablo 3.11. 30 Çalıştırmada Gezilen MRYP Optimal Mesafeleri (Senaryo 5) 42

ŞEKİLLER LİSTESİ

Şekil 1.1. Çokgenler Olarak Engel 8
Şekil 1.2. Keyfi Biçim Olarak Engel 8
Şekil 1.3. Koordinat noktaları olarak engel 9
Şekil 2.1. Harita temsili 20
Şekil 2.2. Engeller ile harita 20
Şekil 2.3. Önerilen yaklaşımlar için akış şeması 22
Şekil 2.4. Bölgesel yol planlama hareketliliği sistemi 23
Şekil 3.1. Robot yolu (ABC Senaryo 1) 28
Şekil 3.2. Robotun adımları boyunca elde edilen amaç fonksiyonu değerleri (ABC Senaryo 1) 28
Şekil 3.3. Robot yolu (qABC Senaryo 1) 29
Şekil 3.4. Robotun adımları boyunca elde edilen amaç fonksiyonu değerleri (qABC Senaryo 1) 29
Şekil 3.5. Robot yolu (ABC Senaryo 2) 31
Şekil 3.6. Robotun adımları boyunca elde edilen amaç fonksiyonu değerleri (ABC Senaryo 2) 31
Şekil 3.7. Robot yolu (qABC Senaryo 2) 32
Şekil 3.8. Robotun adımları boyunca elde edilen amaç fonksiyonu değerleri (qABC Senaryo 2) 32
Şekil 3.9. Robot yolu (ABC Senaryo 3) 34
Şekil 3.10. Robotun adımları boyunca elde edilen amaç fonksiyonu değerleri (ABC Senaryo 3) 34
Şekil 3.11. Robot yolu (qABC Senaryo 3) 35
Şekil 3.12. Robotun adımları boyunca elde edilen amaç fonksiyonu değerleri (qABC Senaryo 3) 35
Şekil 3.13. Robot yolu (ABC Senaryo 4) 37
Şekil 3.14. Robotun adımları boyunca elde edilen amaç fonksiyonu değerleri (ABC Senaryo 4) 37
Şekil 3.15. Robot yolu (qABC Senaryo 4) 38
Şekil 3.16. Robotun adımları boyunca elde edilen amaç fonksiyonu değerleri (qABC

Senaryo 4) 38
Şekil 3.17. Robot yolu (ABC Senaryo 5)..... 40
Şekil 3.18. Robotun adımları boyunca elde edilen amaç fonksiyonu değerleri (ABC Senaryo 5) 40
Şekil 3.19. Robot yolu (qABC Senaryo 5)..... 41
Şekil 3.20. Robotun adımları boyunca elde edilen amaç fonksiyonu değerleri (qABC Senaryo 5) 41

GİRİŞ

Robot sistemleri, 1954'te George Devol'ün ilk dijital programlanabilir robotunun icat edilmesinden bu yana hesaplamada büyük bir başarı elde etmiştir. Bu tür robotların verimli kullanımı konusunda birçok çalışma yapılmıştır [11,15,27]. Özellikle hareket edebilen mobil robotlar ve robotun insan kontrolü olmadan gezinmesini ele alan yol planlaması problemi, günümüzde yoğun olarak çalışılan araştırma alanlarındandır [5,10,11,12,15,20].

Bahçe işleri, temizlik, süpürme gibi günlük işlerde veya hastane, depo ve endüstri gibi ortamlarda malzemeleri istenen yerlere taşımak için robotların kullanılması insanlara kolaylık sağlamaktadır. Bunların yanında özellikle insanlar için tehlikeli olan veya bir insanın çalışmasının mümkün olmadığı alanlarda ve işlerde robotların kullanımı çok önemlidir. Örneğin, insanların tehlike altında olduğu veya arazi kısıtlamaları nedeniyle belirli hedeflere ulaşamadıkları durumlarda (askeri alanlar / savunma alanları, doğal afet bölgeleri, su altı araştırmaları, uzay araştırmaları vb.) mobil robotlar çalıştırılabilir. Bu tür hareketli robotların insan kontrolüne ihtiyaç duymadan ilerleyebilmesi için yol planlaması önemlidir [19]. Bu planlanan yolda robotun engellere çarpmaması gerekirken, yolun varış noktası ile başlangıç noktası arasındaki en kısa yol olması, minimum enerji tüketimi ile ve minimum sürede hedefe ulaşma gibi çeşitli amaçlar dikkate alınarak en uygun yol bulunmaya çalışılmalıdır.

Literatürde, karınca koloni optimizasyonu [12,18,21], genetik algoritma [10,14], parçacık sürü optimizasyonu [15,16], diferansiyel gelişim [26] gibi evrimsel hesaplamaya dayalı optimizasyon algoritmaları temelli yöntemler ile bu problemin çözümü için farklı yaklaşımlar önerilmektedir [12, 18, 21, 10, 14, 15, 16, 26]. Bu tür yöntemler de optimumu garanti etmediği için daha iyilerinin araştırılması süreci devam etmektedir.

Mobil robot yolu planlama küresel (çevrim dışı-offline) yol planlama ve bölgesel (çevrim

içi-online) yol planlama olmak üzere iki temel başlık altında incelenebilir [19]. Küresel yol planlaması mobil robotun ortamı hakkında tam bir bilgiye sahip olmasını gerektirir [19]. Ancak bölgesel yol planlamasında mobil robot sensörlerini kullanarak algılayabildiği bilgilere sahiptir ve çevresel değişikliklere cevap vererek yolunu planlar [19, 14]. Bu çalışmada, bir başlangıç noktasından durağan bir hedef noktaya ulaşmaya çalışan tek bir mobil robot için bölgesel yol planlaması yapılmıştır ve robotun gezindiği ortamda elips şeklinde durağan engeller bulunmaktadır.

ABC algoritması, bal arısının yiyecek arama davranışlarından etkilenerek Karaboga tarafından geliştirilmiş olan sürü zekasına dayalı bir optimizasyon algoritmasıdır [1, 2]. Sürekli ve ayrık optimizasyon problemlerini çözmek için yaygın olarak başarılı bir şekilde kullanılmaktadır [3]. Literatürde ABC kullanılarak geliştirilmiş bölgesel yol planlama temelli bazı yaklaşımlar bulunmaktadır. Faridi ve arkadaşları, mobil robotlara uygun bir yol bulmak için ABC ve evrimsel programlamanın kombinasyonu ile bir strateji önermiştir [4]. Çalışmalarında, kümelenme büyüklüğü yüzdesine sahip düzensiz şekilli engellerin bulunduğu bilinmeyen bir ortamda, mobil robot bir engel fark ederse amaç fonksiyonuna bir ceza eklenmekte ve bu şekilde dinamik veya statik engellerden uzaklaşmaktadır. Diğer bir çalışmada, Liang ve arkadaşları, ABC tabanlı yeni bir yöntem (Efficient Artificial Bee Colony – EABC) önerdiler [5]. Bu yöntem birden çok mobil robot için serbest çarpışma stratejisini kullanarak hedef için uygun amaç fonksiyonu seçerek çevrimiçi yol planlaması yapmayı sağlamaktadır. Her iki çalışmada ([4] ve [5]) da değiştirilmiş ABC algoritması robotun uygulanabilir bir yol bulması için attığı her adımda çağrılmaktadır. Cruz ve arkadaşları, bölgesel yol planlaması en iyi yolu bulmak için ABC algoritması kullanarak uygulanabilir bir yol planlayıcısı uyguladılar ve çarpışmadan kaçınma için Evrimsel Programlama (EP) kullanarak küresel yolu optimize önerlerdi [7]. Girgas ve arkadaşları, yol planlamalarını çözmek için ABC algoritmasını kullanan bir çalışma önerdiler. Ortamda engelleri önlemek için bölgesel bir yol planlama yaklaşımı kullandılar. Deneysel çalışmalar yaptılar ve ABC'yi PSO ile karşılaştırdılar. Sonuçları ABC, PSO'dan daha iyi bir performansa gösterdi [24]. Hızlı ABC (quick ABC-qABC) algoritması ise Karaboga ve Gorkemli tarafından ABC'nin gözcü arı aşamasının daha ayrıntılı bir şekilde yeniden modellenmesi ile oluşturulmuş, birçok problemde ABC'ye göre bölgesel optimuma yakınsama hızı daha yüksek olan algoritmadır [9, 8]. Bildiğimiz kadarıyla mobil robotların bölgesel yol planlaması için literatürde qABC

temelli herhangi bir yaklaşım bulunmamaktadır. Bu tez çalışmasında, ABC ve qABC'nin daha önceki çalışmalarda gösterdiği başarılı performans göz önünde bulundurularak bu problemde de iyi sonuçlar elde edebilecekleri düşünülmüş ve mobil robotların bölgesel yol planlaması probleminin yukarıda ayrıntıları belirtilen versiyonunun çözümü için bu algoritmalara dayalı yeni yaklaşımlar önerilmiştir. Önerilen yaklaşımlar farklı sayıda, büyüklükte ve konumda engeller ve farklı başlangıç-hedef noktası konumları ile farklı senaryolar için test edilmiştir. Simülasyon sonuçları analiz edilerek önerilen yaklaşımlar birbiri ile kıyaslanmıştır. Bu şekilde, bu tez çalışması ile söz konusu problem için literatüre yeni yöntemlerin kazandırılmasının yanı sıra ABC ve özellikle ABC'ye göre daha yeni bir optimizasyon algoritması olan qABC'nin performansının bir gerçek hayat probleminde incelemesi ve başarımının araştırılması da amaçlanmıştır.

Bu çalışmanın birinci bölümünde, mobil robot yol planlama problemleri ve robotik yol planlama yöntemleri hakkında genel bir bilgi verilmektedir Literatür çalışması da bu bölümde sunulmaktadır. İkinci bölüm, önerilen yaklaşımları açıklamaktadır. Üçüncü bölüm, önerilen yaklaşımlar için deneysel çalışmaları ve test sonuçlarını göstermektedir. Son olarak dördüncü bölümde ise, sonuç ve gelecekteki çalışmalar için öneriler verilmektedir.

1. BÖLÜM

GENEL BİLGİLER ve LİTERATÜR ÇALIŞMASI

Bu tez çalışmasında, günümüzde birçok araştırmacının önemli bir araştırma alanı olan mobil robotların yol planlama problemini çözmek için yeni ABC tabanlı yaklaşımlar geliştirilmesi amaçlanmaktadır. Bu nedenle, araştırmada ABC'nin önerilen çalışmamızdaki başarısının yanı sıra bu sorun için daha etkili ve başarılı çözümler sunmaktadır. Ancak, ana çalışmamıza girmeden önce, çalışmamızı daha iyi anlayabilmemiz için birkaç önemli robotik kavramın vurgulanması gerekmektedir.

1.1. Mobil Robotik Yol Planlama Problemi

Geçimimizde bize yardımcı olacak birçok farklı türde robotik sistemler geliştirilmiştir ve çeşitli görevlerde etkin performanslar göstermiştir [22]. Bu görevler arasında robotik sistemlerin akıllı ev ortamlarında, havaalanlarında, askeri alanlarda, laboratuvarlarda, imalat sanayilerinde, alışveriş merkezlerinde ve diğer birçok alanda kullanımı bulunmaktadır. Robotik sistem normalde bir istihbarat algoritması ile programlanır ve yerleştirilir. Mobil robot daha sonra verilen amacı başarmak için farklı teknikleri kullanarak çeşitli görevleri çözmek için zorlanır. Bu nedenle, mobil robot optimum bir yörünge elde etmek için çeşitli zorluklarla karşı karşıyadır. Bu zorluklar daha sonra mobil robotik yol planlama problemleri olarak tanımlanacaktır. Bu problemler yerleştirme, haritalama ve hareket kontrolünü içerecektir.

a) Yerleştirme – Yerleştirme tekniği, mobil robota belirli bir ortamdaki konumunu tanımlamada yardımcı olur. Bu teknik, aralık algılamalar, dış ortamlar için GPS cihazlarının veya ultra-yüksek çözünürlüklü kameraların (Nokta Gri kameralar gibi herhangi biri Bumble bee veya Matrox çerçeve kapmaklı Lady böcekleri, Kinetic 360 kameralar, daha küçük robotlar için CMOS görüntü sensörünün kullanılması) içerebilir. Konumlar, ortam ile ilgili farklı sembolik referanslar kullanılarak tanımlanabilir [22]. Bu

sembolik referanslar 3D düzlem yapıları, Çokgen 2D semboller veya Kartezyen nokta koordinatları olarak gösterilebilir.

b) Haritalama – Robot haritalama fikri ortamının yapısı ile ilgilidir. Robotun harita ortamı statik veya dinamik olarak tanımlanabilir. Harita mobil robotun ortamının fotografik hafızasına sahip olmasına yardımcı olur. Harita, grafik gösterimi veya matris gösterimi kullanılarak tanımlanabilir. Bu teknik, optimum yörüngeyi göstermeye ve ortamda bulunan engellerin grafiksel gösterimini tanımlamaya yardımcı olur.

c) Hareket kontrol – Mobil robotu hedef noktasına doğru en iyi yol boyunca hareket eder. Bu nedenle, en iyi en iyi yolu bulmak için hedef noktanın konumu mobil robot tarafından bilinir. Bu, ilk yol planlama aşaması olarak kabul edilir. Artık robotta, haritada adres yeri olarak hizmet veren belirli bir hedef noktası belirlenmiştir. Hareket kontrolünde robot birçok tekniği dikkate alır. Bu teknikler menzil algılamaları, mesafe değerlendirmeleri ve engellerden kaçınmayı içerir.

1.2. Robotik Yol Planlama Algoritmaları

Mobil robotta zeki gömülü sistem, bir robotun hedef noktasına ulaşmasının başarısının ardındaki temel motordur. Bu zeki sistemi robot için en uygun yolu bulmak amacıyla tasarlanmış algoritmalar içerir. Oluşturulan her bir algoritma, yapısına bağlı olarak belirli bir ortamda avantajlıdır. Bazı algoritmalar, karmaşık kısıtlı yüksek boyutlu alanlarda daha iyi performans gösterirken, diğerleri düşük kısıtlamalı düşük boyutlu sistemlerde daha iyi performans gösterir. Yol planlama için algoritmalar şunları içerir:

Klasik algoritmalar

- Izgara tabanlı algoritmalar: A-star arama algoritması, D-star arama algoritması
- Aralık tabanlı algoritmalar: Dijkstra algoritması
- Geometrik algoritmalar: Minkowski toplamı, Görünürlük grafiği, Hücre ayrışımı
- Ödül tabanlı algoritmalar: Markov karar süreci (MDP), MDP'nin bir uzantısı olan Bulanık Markov karar süreçleri (FDMP).
- Örnekleme tabanlı algoritmalar: Olasılıksal Yol Haritası (PRM), Hızla Keşfedilen Rastgele Ağaç (RRT)

Modern algoritmalar

- Metasezgisel Optimizasyon Algoritmaları: Genetik algoritma (GA), ABC algoritması, Ant Koloni Optimizasyonu (ACO) algoritması, Parçacık Sürüsü Optimizasyonu (PSO) algoritması.

Klasik ve modern algoritmaları arasındaki temel fark, modern algoritmalar klasik algoritmalardan daha hızlı optimal çözümü elde edebilmesidir [29].

Robot Türleri

Birkaç çeşit robot vardır, ancak robotların ana yapısı sabit ve mobil robotlara ayrılabilir. Bu tip robotlar farklı formları takip eder, bu formlar insansı, evsel, sanayileşmiş, medikal, askeri robotlar ve daha pek çoğunu içerir.

1.3. Mobil Robot Yolu Planlama Kategorileri

Bu çalışmanın kapsamı mobil robotların navigasyonu veya yol planlaması üzerinde durulmaktadır. Bu nedenle, robotların navigasyon şeması açıklanmıştır ve bu 2 farklı tekniğe ayrılmıştır. Bu teknikler şunlardır: Küresel yol planlaması (çevrimdışı), Bölgesel yol planlaması (çevrimiçi), Engeller dahil olmak üzere statik veya dinamik ortamlar, ortam boyutları (2D, 3D vb.).

1.3.1 Küresel Yol Planlaması (Çevrimdışı)

Bu yaklaşım temel olarak robotun tüm bilgi ve ortam bilgisi (engel konumları dahil) verildiği robotik yol planlamasının doğrudan yaklaşımı gibidir. Robot belirli bir hızda hareket eder ve başlangıç noktası ile hedef noktası arasındaki mesafeyi hesaplar. Robotun amacı, Ortam hakkında verilen bilgileri kullanmak ve engellere çarpmadan varış yerine uygun bir yol bulmaktır. Yaklaşımın doğası, çoğunlukla kendisinin bir yol yaratmadığı, ancak tüm engellerle bilinen konumla hedef noktasına uygun bir yol bulmadığı, dolayısıyla verilen yaklaşım teriminin mobil robot yolu planlamasında çevrimdışı yaklaşım olduğu statik bir yaklaşımdır.

1.3.2 Bölgesel Yol Planlama (Çevrimiçi)

Bölgesel bir yol planlama kurulumunda yaklaşım, robot için çok dinamik ve kendi kendini eğitir. Otonom robotun ortamı veya engelleri hakkında hiçbir ipucu yoktur,

dolayısıyla yol boyunca engelleri tespit etmek ve çarpışmayı önlemek için yerleşik bir zekaya sahip olarak kendi yoluna karar verir, böylece verilen yaklaşım terimi çevrimiçi yaklaşım veya gerçek zamanlıdır. Mobil robotik yol planlaması için yaklaşım. Yolun her adımından sonra robotlar tarafından oluşturulan yollar ve hedef noktasına ulaşılana kadar en uygun yol seçilir.

• BYP ile çalışmamız arasındaki ilişki.

Robot, Öklid mesafesini başlangıç noktasından belirlenen varış noktasına kadar hesaplar ve belirli bir yarıçaplı yörüngede hedefine doğru hareket etmeye başlar. Yol boyunca her hareketinden sonra robot ortamını hesaplar ve mevcut ortamı içindeki engelleri hedef noktasına giden yol boyunca engellerden kaçınmak için birleşik algılama tekniğini kullanarak bir değişken ayarlar. Robot, her hareketinden sonra değişmiş başlangıç noktaları nedeniyle varış noktasına kadar Öklid mesafesini yeniden değerlendirir ve optimizasyon algoritmamızın (ABC/qABC) yardımıyla en iyi optimum yolu seçer. Etkili bir mobil yol planlama sistemi geliştirmek için robotun ortamını uygun şekilde tanımlanması gerekir. Bunlar (statik veya dinamik) ortamları ve (sabit veya mobil) engelleri içerir.

a) Sabit Engelli Statik Ortamlar

Özerk mobil robot yolu planlaması için statik ortamların kullanılması fikri, çoğunlukla sanal ortamlarda kullanılan robot davranışlarının yakın bağlamda daha iyi anlaşılabileceği yönündedir [5]. Bu tür bir ortamda, içinde bulunan her engel ve varlık durağan ve hareketsizdir, bu nedenle araştırmacılara robotu yakından inceleme ve robot için uygun ve optimal bir yol bulmak için farklı optimizasyon algoritması testleri yapma fırsatı verir. Bu tür ortamın tasarımı genellikle engellerin çoğunlukla farklı çokgen şekil ve boyutlarda temsil edilebildiği yerlerde esnektir. Bu tür ortamların tipik bir örneği, Pacman gibi erken video oyunlarında bulabiliriz.

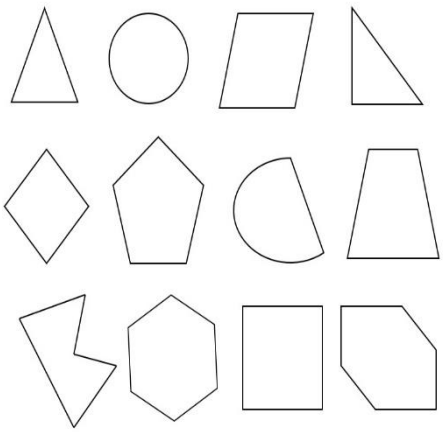
b) Mobil Engellerle Dinamik Ortamlar

Dinamik yaklaşım, aktif bir ortamda gerçek zamanlı hareketli engellerden oluşur. Robot, hedef noktaya doğru ilerlemek ve engellerden kaçınmak için çeşitli sözdizimleri ve yol planlama teknikleriyle çok sağlam bir şekilde donatılmış olmalıdır. Ortamın simüle edici gerçekleri, daha gerçekçi ve dolayısıyla inşa edilmesi çok karmaşık olmasıdır. Yaklaşım

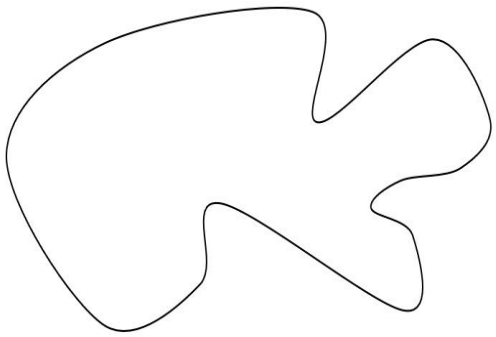
genellikle gerçek bir yaşam durumuyla uğraşmaya çalışan yüksek teknoloji araştırmacıları tarafından benimsenir. Bu, teslimat robotlarını, dronları, askeri insansız robotları (kara mayını dedektörleri), günümüzün kendinden otomatik veya otomatik pilotlu araçları ve uçakları vb. içerir.

c) Engellerin Şekillerine Göre

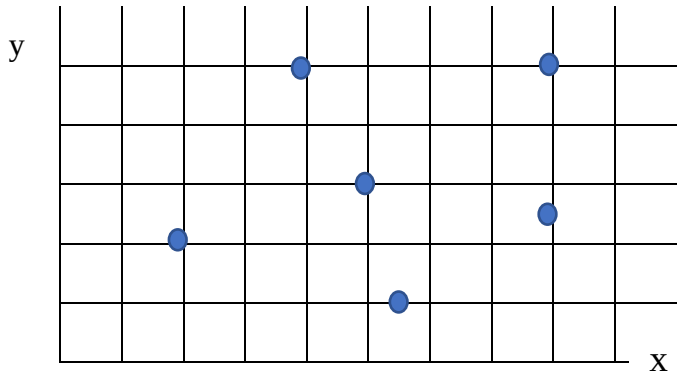
Bölgesel yol planlama tekniği, çeşitli şekil ve boyutlardaki engellerin bulunduğu bir ortamda robot(lar) için güvenilir bir yolun yapılandırılmasında çok yönlü olabilir. Yapılan araştırmaların çoğunda araştırmacıların çoğu, yöntemlerinde veya yaklaşımlarında kısıtlamalarını temsil edecek şekilde çokgen şekilli engelleri (normal veya düzensiz poligonlar) kullanma eğilimindedir. Engelin şekli ve boyutu, bir mobil robot için bir yol planlayıcısının geliştirilmesinde önemli bir rol oynar. Engeller, belirli bir harita veya ortamdaki koordinatlar olarak da gösterilebilir. Robot, engel koordinatlarını, yörüngesindeki engellerle çarpışmamaktan kaçınmalıdır. Diğer şekiller bölgesel noktaları, U şekilleri veya keyfi şekilleri içerir. Mobil robot yolu planlamasında kullanılan engel temsillerinin bazı örnekleri aşağıda Şekil 1.1, Şekil 1.3'e gösterilmiştir:



Şekil 1.1. Çokgenler olarak engel



Şekil 1.2. Keyfi biçim olarak engel



Şekil 1.3. Koordinat noktaları olarak engel

Farklı sayıda robot için BYP

Bölgesel yol planlama teknolojileri, farklı sayıda robot kullanılarak geliştirilebilir. Bu, robotun hedefine ulaşmasında amaç fonksiyonunun değerlendirilmesinde büyük bir ayırıcı faktördür. Yol planlamasında robot sayısını ayırt etmek için kullanılan temel terminolojiler şunlardır: Tek ve Çoklu robotlar.

- Tek robotlu BYP

Bölgesel bir yol planlama yaklaşımı için tek bir robot kullanma fikri genellikle herhangi bir ortamda çalışmak için daha kolay ve daha esnektir. Tek bir robotun amaç fonksiyonunu geliştirirken, kısıtlamalar genellikle birden çok robottan daha az veya daha azdır. Kısıtlamalar genellikle 2'dir; (i) robot başlangıç noktasından hedef noktaya en uygun yolu bulmak ve (ii) belirli bir ortamdaki engellerden kaçınmak. Bu kısıtlamalar, aşağıdaki gibi objektif fonksiyonlar halinde Denklem 1.1 ve 1.2'de gösterilmiştir;

1. Hedef nokta için objektif fonksiyon olarak verilebilir

$$Obj_1(x_i) = \|x_i - Hedef\| \qquad (1.1)$$

x_i, robotun başlangıç konumunu ve $Hedef$ temel olarak hedef noktanın yerini göstermektedir.

2. Engellerden kaçınma için objektif fonksiyon

$$Obj_2(x_i) = \sum_{i=0} \frac{1}{\|x_i - Obs_j\|} \qquad (1.2)$$

Obs_j ortamdaki engellerin yerlerini göstermektedir

- Birden çok robot

Bu yaklaşımda, mobil robotlar, ortamdaki tüm seyrek göstergelerdir ya hepsi ya hep ya da her biri için farklı hedef noktalar için belirlenmiş bir hedef noktasıdır. Bu nedenle, yaklaşımın karmaşıklığına dayanarak, bu faktörlerin amaç bir fonksiyonu bulmak için kısıtlamaların sayısı orantılı olarak artar. Yeni teknikler ve formülasyonlar normal olarak yapıya eklenir ve yalnızca bir hedef noktasına ulaşmaya değil, aynı zamanda diğer robotlarla ve engellerle çarpışmayı önlemek için de yardımcı olurlar. Bu yeni teknikler, robotların birbirleriyle çarpışmasını önlemek için her bir robot için bir hız vektörü veya bir zaman penceresini içeriyor olabilir [5, 8]. Çok robotlu bir yol planlama sistemi için kısıtlamalar, hedefinize bağlı olarak 3 veya daha fazla olabilir. Bu olabilir; (i) robotların başlangıç noktalarından hedef noktaya (lar), (ii) diğer robotlarla çarpışmanın önlenmesi ve (iii) belirli bir ortamda engellerin önlenmesi. Yukarıda belirtilen tüm kısıtlamalar nihai amaç fonksiyonunu formüle etmeden önce dikkate alınmalıdır.

1.4. Literatür Çalışması

Robot yolu planlaması alanında birçok araştırma ve çalışma yapılmıştır. Mobil robot yol planlama çalışması farklı yöntemler ve teknikler içerir. Örneğin, iki boyutlu ve 3 boyutlu alan, dinamik engel ve statik engel, çevrimdışı ve çevrimiçi yol planlama çalışmaları vb. Yol planlaması için yapılandırılmış tek bir robot veya çoklu robotlar dikkate alınmalıdır. Planlama için tercih edilen yöntemler arasında klasik, metasezgisel, makine öğrenme yöntemleri bulunmaktadır [21]. Örneğin Pekiştirmeli Öğrenme (Reinforcement Learning) temeli yöntemler bunlardan biridir [23].

Bu tez çalışmasında ele alına ABC algoritması aynı zamanda problemin farklı senaryoları için metasezgisel bir yöntem olarak da kullanılır. Karaboga [1, 2] tarafından tanıtıldığından beri, algoritma araştırmacılardan çok ilgi görmüştür.

Liang ve arkadaşları, çok mobil robot sistemli yol planlama için serbest çarpışma stratejisini kullanarak, hedef için uygun amaç fonksiyonunu seçerek çoklu mobil robotların çevrimiçi yol planlamasını çözmek için ABC algoritmasına dayalı bir yöntem

önerdi [5]. Çalışma, birden fazla mobil robot için çevrimiçi navigasyonu çözen verimli bir yapay arı kolonisi (EABC) algoritması önermiştir. Her robotun başlangıç pozisyonları önceden tanımlanmış ve hedef pozisyonları da belirlenmiştir. Önerilen çalışmaya göre, mobil robotlar engelli statik bir ortama tanıtıldı. Engellerin çeşitli şekil ve boyutlarda olduğu varsayılmaktadır. Tanımlanmış yarıçapları olan dairesel bir şekilde temsil edilirler. Bu nedenle, engellerden kaçınmak için, birden fazla mobil robot, engellere çarpmadan hedef noktaya yeni yollar oluşturan EABC algoritmasını kullanır. Bu görev çevrimiçi yol planlama tekniği kullanılarak gerçekleştirildi. Bu nedenle, çoklu robotlar her adımı kendi hedef noktalarına ulaşana kadar belirli bir uzunluk veya mesafe ile hesaplar. Önerilen EABC algoritması, objektif fonksiyonların hibridizasyonunu kullanır. Bu amaç fonksiyonlar robot pozisyonları arasında hedef pozisyonları, çoklu robotlar arasındaki mesafeler ve robotların engellere olan mesafeleri [5]. Tüm objektif fonksiyonları, her bir fonksiyonun çalışmanın nihai sonuçları üzerinde belirli bir etkisinin olduğu melez bir fonksiyon oluşturmak üzere dahil edildi. Önerilen çalışma, sistemdeki mobil robotun izleme kontrolünü de içermektedir. İzleme kontrol tekniği mobil robotların yörüngesine bir referans rolü oynar. Bu teknik, tüm mobil robotların çeşitli optimum yollarında hedef noktalara gidebilecekleri şekilde uygun eylemler veya sinyaller ileten izleme denetleyicilerini içerir. Bu teknik ayrıca çevrimiçi yol planlamasından gelen izleme hatalarını da tanımlar.

Önerilen algoritması önerilen çalışmanın performansını test etmek için kıyaslama fonksiyonları ve diğer optimizasyon algoritmaları kullanılarak kullanılmıştır. Bu deneysel testi gerçekleştirmek için EABC algoritması ile geleneksel ABC arasında diğer iki çoklu ajan algoritmasıyla (PSO ve GA) bir karşılaştırma yapmayı seçtiler. EABC algoritmasının diğer algoritmalardan daha iyi bir hassasiyet performansına sahip olduğu görülmüştür. Öte yandan, EABC'nin işlem süresi diğer algoritmalardan daha uzundu, çünkü EABC algoritması yüksek kalite performansını arttırmak için çeşitli stratejiler benimsedi. Deneysel çalışmalar sırasında hibrit objektif fonksiyonlardan (g1, g2, g3) 4 örnek test edilmiştir. İlk örnek, engelsiz tek bir robotun başlangıç noktasından hedef noktasına Öklid mesafesi olan g1 objektif fonksiyonunu temsil ediyordu. İkinci örnek, g1 ve g2 objektif fonksiyonunu temsil ediyordu; Üçüncü örnek, ortamda engel olmadan karşılık gelen hedef noktalarına doğru hareket eden 2 robotun optimum yolunu sunan g1 ve g3'ü temsil etti. Dördüncü ve son örnek, 5 mobil robot için ortamdaki engellerle ortak

bir hedef noktasına en uygun yolu sunan g1, g2 ve g3'ü temsil etmiştir (buna hem engellerden kaçınma hem de robottan kaçınma teknikleri dahil). Önerilen çalışmanın nihai sonuçları, birçok mobil robotun hedeflerine ulaşması için en uygun yolları ve engellerin ve diğer robotların çarpışması olmadan etkili performanslara yol açmaktadır [5].

Ma ve arkadaşları. Mobil robotlar için ABC algoritmasına dayalı bir dinamik yol planlaması önerlerdi [6]. Çalışma, bir yuvarlanma zaman penceresi olarak adlandırılan bir strateji ile birleştirilmiş küresel bir yol planlama yöntemini ortaya koymaktadır. Bu çalışmada, dinamik engellerin aynı anda engellenmesi için uygunluk değeri fonksiyonları tasarlanmış ve önerilen yöntemin performansını görmek için simülasyon testleri yapılmıştır. Cruz ve arkadaşları, ABC algoritması kullanarak uygulanabilir bir yol planlayıcısı uyguladılar ve çarpışmadan kaçınma ile en iyi yolu bulmak için Evrimsel Programlama (EP) kullanarak küresel yolu optimize ettiler [7]. Wang ve arkadaşları, çoklu robot yolu planlaması için çok amaçlı bir ABC algoritması önermişlerdir. Algoritma, yüksek performans için üç ana bileşene dayanan tasarlanmış bir amaç fonksiyonuna sahipti: uzunluk, güvenlik ve yumuşaklık [13]. Lei ve arkadaşları, (Unmanned Aerial Vehicles-UAV) yol planlaması için karmaşık ortamları çözmek için ABC algoritmasını değiştirdi ve yeni bir ABC temelli yaklaşım önerlerdi [22]. Girgas ve arkadaşları, yol planlamalarını çözmek için ABC algoritmasını kullanan bir çalışma önerdiler. Ortamda engelleri önlemek için bölgesel bir yol planlama yaklaşımı kullandılar. Önerilen çalışmada, önce ABC'nin yöntemlerini açıkladılar, sonra karşılaştırma için PSO yöntemlerini tanıttılar. Deneysel çalışmalar yaptılar ve sonuçları ABC, PSO'dan daha iyi bir performansa gösterdi [24].

ABC algoritmasına ek olarak, robotik yol planlama problemlerinin çözümünde pek çok ilgili metasezgisel algoritma da kullanılmıştır. Mohanty ve Parhi, (cuckoo search algorithm) dayanan akıllı bir metot önerlerdi. Bu çalışmada, mobil robotun statik bir ortamda bölgesel yol planlaması için yapı geliştirilmiştir. Algoritma problemi, ilk olarak, robotun başlangıç noktasından en kısa mesafeyi istenen hedefe veya hedefine göre sıralamak için, bir minimizasyon probleminde dönüştürüldüğü için mükemmel bir şekilde çalıştı [17]. Lijun ve arkadaşları ayrıca, robotların yol planlamasının üstesinden gelmek için hızlı bir yakınsama karınca kolonisi algoritması önermişlerdir. Algoritmanın yüksek

bir yakınsama hızını elde etmede, keşif faktörü, feromon buharlaşma katsayısı, feromon artışını iyileştirmek ve karıncaların hedefe odaklanmak yerine körüne ulaşmak için gereksiz yolları ortadan kaldırmayı önerdiler [18].

Ganeshmurthy ve Suresh, dinamik bir ortamda robot yolu planlamasında termal işleme algısına dayanan bir yöntem önermişlerdir [11]. Uriol ve Moran, karmaşık ortamlarda mobil robotların yol planlama problemini çözmek için karınca kolonisi optimizasyonunu kullandılar [12]. Ek olarak, genetik algoritma, robot yolu planlama problemi için kullanılan yapay bir zeka optimizasyon algoritmasıdır [10, 14]. Chen ve arkadaşları, çarpışma içermeyen optimal bir yol arayışında, mobil robotlara PSO tabanlı küçük bir yatay kontrol yaklaşımı getirmişlerdir [30].

2. BÖLÜM

YÖNTEM

Bu bölümde, önerilen yaklaşımlar ve bu yaklaşımların temelinde bulunan ABC ve qABC algoritmaları açıklanmaktadır.

2.1. ABC Algoritması

Yapay Arı Kolonisi (ABC) algoritması, 2005 yılında Karaboga tarafından sunulan, doğadan esinlenilerek geliştirilmiş, bal arılarının yiyecek arama davranışlarını modelleyen bir algoritmadır [1, 2]. Algoritma yapısı üç temel bileşenden oluşur: görevli arılar, görevsiz arılar ve yiyecek kaynağı.

i. Görevli Arılar

Görevli arının görevi temel olarak daha önceden bilinen yiyecek kaynaklarını ziyaret etmek ve o kaynakların etrafında araştırma yaparak daha iyi yiyecek kaynakları bulmaktır. Her görevli arı, ziyaret edilecek belirli bir yiyecek kaynağına sahiptir ve nektar kaynağı hakkındaki yakın ziyaret bilgilerini koloninin geri kalanıyla paylaşır. Getirdikleri bilgiler; nektar kaynağının karlılığı, yuvaya olan mesafesi ve yönü gibi bilgilerdir.

ii. Görevsiz Arılar

Görevsiz arılar iki tür arından oluşur: gözcü arılar ve kaşif arılar. Gözcü arılar temel olarak görevli arıları ziyaret ettikleri yiyecek kaynakları hakkında bilgi almak için beklerler. Görevli arılar, keşfedilen yiyecek kaynakları hakkında bilgi paylaşmak için dans eder. Gözcü arılar bu bilgiyi alırlar ve yiyecek kaynaklarının kalite değerlerine göre belirlenen bir olasılık dahilinde yiyecek kaynakları arasından seçim yaparlar. Öte yandan, kaşif arılar, tükenen kaynaklar terk edildikten sonra yeni yiyecek kaynaklarını araştırırlar.

iii. Yiyecek kaynağı

ABC algoritmasında yiyecek kaynaklarının popülasyonu temel alınmaktadır. Yani bir yiyecek kaynağının pozisyonu olası bir çözümü ifade etmektedir. Yiyecek kaynağının kalitesi ise o çözümün uygunluk değerini göstermektedir.

2.1.1. ABC Algoritmasının İşleyişi

Daha önce belirtildiği gibi, algoritma temel bileşenlere yapılandırılmıştır. Bu bileşenler görevli arılar, gözcü arılar ve kaşif arılar olmak üzere üç çeşit arı içerir. Başlangıçta tüm yiyecek kaynağı konumlarının kaşif arılar tarafından keşfedildiğini varsayıyoruz. Daha sonra, yiyecek kaynakları belirlenen bu kaşif arılar görevli arı haline gelmektedir. Her bir görevli arı mevcut yiyecek kaynağının etrafında komşuluk araştırması yaparak daha iyi bir kaynak bulmaya çalışır ve bulamazsa mevcut kaynağı tüketmeye devam eder. Gözcü arılar ise görevli arıların yiyecek kaynakları hakkında paylaştıkları bilgiye göre kaynaklardan birini seçer ve o kaynağın etrafında gözcü arıya benzer şekilde araştırma yaparlar. Yiyecek kaynağı tükenen görevli arı yeniden kaşif arıya dönüşür ve mevcut kaynağını terk ederek yeni bir kaynakla çalışmaya devam eder. Her bir görevli arısı bir yiyecek kaynağına tahsis edildiği için, görevli arıların sayısı yiyecek kaynaklarının sayısına eşittir. Gözcü arı sayısı da görevli arı sayısına denktir. Algoritmanın genel modeli aşağıda gösterilmiştir:

Başlangıç Aşaması
REPEAT
Görevli Arı Aşaması
Gözcü Arı Aşaması
En iyi yiyecek kaynağını hafızaya al.
Kaşif Arı Aşaması
UNTIL (durdurma kriteri sağlanana kadar)

i. Başlangıç Aşaması

Bu, belirli kaynak parametrelerinden yiyecek kaynaklarının popülasyonunu tanımlayan algoritmanın ilk aşamasıdır. Her bir yiyecek kaynağı, optimizasyon problemine bir çözüm vektörünü temsil ettiğinden, her bir çözüm vektörü, belirli sayıda değişken dizisini de içerir. Bu değişkenler objektif fonksiyonu en aza indirecek şekilde optimize edilecektir [2]. Başlatma Eşitlik 2.1'de aşağıda göstermektedir:

$$x_{k,j} = lb_j + rand(0,1) \cdot (ub_j - lb_j) \quad (2.1)$$

x_k başlangıçtaki yiyecek kaynağının (veya çözümün) j. boyutunda yer alan değeri ifade etmektedir, *rand* (0, 1) arasında bir rastgele sayıyı ifade etmektedir, ub ve lb sırası ile j. boyut yani karar değişkeni için alt ve üst sınırları göstermektedir [1, 2].

ii. Görevli Arı Aşaması

Bu aşamada, Görevli arılar yeni yiyecek kaynakları arayışındadır ve hala popülasyondaki anılarında daha önceki (başlangıçtaki) yiyecek kaynağı ($x_{k,j}$) hakkında bilgi sahibi olurlar. Yeni bir yiyecek kaynağı bulduktan sonra uygunluğunu değerlendirirler (karlılık). Yeni bir komşu yiyecek kaynağının (V_k) kurulması aşağıdaki Eşitlik 2.2'de komşuluk araştırması göstermektedir.

$$v_{k,j} = x_{k,j} + \varphi_{k,j} \cdot (x_{k,j} - x_{m,j}) \quad (2.2)$$

Burada $m \neq k$ için, v_k popülasyonda yeni bir yiyecek kaynağı olduğunda, x_k popülasyondan rastgele seçilmiş bir yiyecek kaynağını, j rastgele seçilmiş bir boyutu ifade etmektedir. φ ise [1,1] aralığında seçilmiş rastgele bir sayıyı göstermektedir. Yeni yiyecek kaynağı v_k yaptıktan sonra hem x_k hem de v_k 'nin uygunluğu hesaplanır ve açgözlü seçim yöntemi uygulanır [1, 2]. En iyi uygunluk değeri sahip yiyecek kaynağı, şu anda en iyi yiyecek kaynağımız olur. Uygunluk değeri değerlendirme teknikleri aşağıdaki denklemlerde gösterilmiştir.

$$fit(x_k) = \begin{cases} \frac{1}{1+f(x_k)}, & eğer,\ f(x_k) \geq 0 \\ 1 + abs(f(x_k)), & eğer,\ f(x_k) < 0 \end{cases} \quad (2.3)$$

$fit(x_k)$ uygunluk değeri, $f(x_k)$ çözümünün amaç fonksiyonu değerini ifade etmektedir

iii. Gözcü Arı Aşaması

Bu aşamadan önce bahsedildiği gibi, görevsiz arıların gözcü arıları ve kaşif arılarından

oluştuğunu anlıyoruz. Gözcü arısı, bu aşamada, önceki aşamada görevli arıları tarafından sağlanan kondisyon değeri bilgisi kullanılarak hesaplanan olasılık değerlerine bağlı olarak bir yiyecek kaynağı seçmek için kullanılır [2]. Olasılık değerleri, uygunluk değerleri elde edildikten sonra hesaplanır. Kondisyon bazlı seçim teknikleri, kondisyonu hesaplamak için kullanılır (örneğin, rulet çarkı seçim yöntemi (Goldberg, 1989)) [2, 3]. Aşağıdaki eşitlik gösterildiği gibi Denklem 2.4 olasılık hesaplanır:

$$p_k = \frac{fit(x_k)}{\sum_{k=1}^{SN} fit(x_k)} \qquad (2.4)$$

p_k seçilen yiyecek kaynakları uygunluğunun olasılığıdır. Yiyecek kaynağı x_k bir Gözcü arısı tarafından olasılıksa olarak seçildikten sonra, yeni bir komşu yiyecek kaynağı v_k denklemi (2) kullanarak belirlenir ve uygunluk değerleri hesaplanır. Görevli arılara benzer şekilde, karşılaştırılan yiyecek kaynakları arasında en iyi zindeliği alan ve onu en iyi küresel yiyecek kaynağı olarak kaydeden açgözlü bir seçim yöntemi uygulanmaktadır. Bu faz tekniği, daha fazla yiyecek maddelerine daha fazla gözcü'nün dahil edilmesini ve olumlu geri besleme davranışlarının elde edilmesini sağlar.

iv. Kaşif Arı Aşaması

Bu aşamadaki kaşif arıları temel olarak, sürü ortamındaki yeni potansiyel yiyecek kaynaklarını araştırmak üzere atanmış olan boşta kalan görevli arılarını temsil etmektedir. ABC algoritmasında her bir yiyecek kaynağı için bir deneme sayacı tutulur. Her bir iterasyonun sonunda deneme sayacı değeri maksimum olan yiyecek kaynağının bu değeri *limit* parametresi ile kıyaslanarak bu yiyecek kaynağının tükenip tükenmediğine karar verilir. Eğer kaynak tükenmişse terk edilir ve ilgili görevli arı bir kaşif arı gibi davranarak Denklem 2.1 ile kendisine yeni bir yiyecek kaynağı belirler.

2.1.2. ABC Algoritmasının Parametreleri

Aşağıdakiler ABC algoritmasında parametre olarak listelenmiştir:

i. Koloni büyüklüğü – hem görevli hem de görevsiz arıları temsil eder. Yiyecek kaynağı sayısının iki katıdır.
ii. Kaynak sayısı (popülasyon büyüklüğü) – yiyecek kaynaklarının miktarını gösterir.
iii. Problemin boyutu – çözümlerin parametre sayısını temsil eder.

iv. Boyut aralığı – alt ve üst sınırlar (lb ve ub) olarak temsil edilen yiyecek kaynaklarındaki çözümlerin ölçeği veya sınırlarıdır.

v. Çevrim (iterasyon) sayısı –algoritmanın daha iyi çözümler bulmak için başlatmadan sonra çalıştırması gereken maksimum döngü sayısını tanımlar.

vi. Limit – kaşif arı için yiyecek kaynaklarının tükenme durumunun kontrolünü sağlayan parametredir.

2.2. qABC Algoritması

ABC algoritmasının başarısı, Karaboga ve Gorkemli'nin qABC algoritmasını geliştirmesine ilham verdi [8, 9]. qABC, standart ABC algoritmasının güncellenmiş bir versiyonudur. Bu yeni sürüm, izleyici arıların davranışını daha doğru bir şekilde modellemekte ve bölgesel minimuma yakınsama hızı açısından standart ABC algoritmasının performansını arttırmaktadır [8]. qABC algoritması, standart ABC'den daha hızlı optimum çözüme yaklaşır. Bunun nedeni, qABC algoritmasında, gözcü arıların yiyecek kaynaklarını farklı bir yaklaşımla seçmeleridir. Gözcü arılar, standart ABC'de olduğu gibi görevli arıların danslarına göre daha yiyecek kaynaklarını seçerler. Ancak bu seçilen kaynağı (x_k) tüketmek için seçmezler. Seçtikleri kaynağın merkezinde bulunduğu bir alanda yani bu kaynağın belirli bir komşuluğunda (N_k) bulunan en iyi yiyecek kaynağını tüketecekleri kaynak olarak belirleyip yine standart ABC'de olduğu gibi çalışmaya devam ederler. Bu komşuluğa giren yiyecek kaynaklarını belirlemek için popülasyondaki bütün kaynakların x_k'ye olan mesafesi (md_k)bulunur. Eğer bir yiyecek kaynağının x_k'ye olan mesafesi $md_k \times r$'den küçük veya eşitse bu yiyecek kaynağı söz konusu komşuluktadır. Burada r komşuluk yarıçapını belirlemeye yarayan qABC'nin bir kontrol parametresidir. Daha sonra bu komşuluktaki en iyi yiyecek kaynağı ($x_{N_k}^{best}$) seçilir ve bu kaynaktan aday çözüm üretilerek süreç ABC'de olduğu gibi devam eder. Gözcü arı aşamasında yeni üretilen aday çözümün denklemi aşağıda Denklem 2.5 gösterilmiştir.

$$v_{N_k,j}^{best} = x_{N_k,j}^{best} + \varphi_{k,j}(x_{N_k,j}^{best} - x_{m,j}) \qquad (2.5)$$

Burada $\varphi_{k,j}$ ve $x_{m,j}$ daha önce verilen aday çözüm üretme denklemleri ile aynı anlamada ve şekilde kullanılmıştır.

2.3. Önerilen Yaklaşımlar

Önerilen algoritma, tanımlanan mobil robot için en uygun yörüngeyi ve engel çarpışma önleme tekniğini bulmak olan ana hedefimize uyacak şekilde uyarlanmış bir ABC algoritmasından oluşur. Önerilen bu algoritmanın geliştirilmesi titizlikle aşağıda açıklanmıştır ve detaylandırılmıştır:

1. Önerilen mobil robot yolu planlama sisteminin haritalanması ve yerleştirilmesi
   - Ortam temsili
   - Izgara temsili
   - Engel temsili
2. Önerilen algoritma yapısı ve yöntemleri

A. Ortam temsili

Bu çalışmada, önerilen çalışma için statik bir ızgara tabanlı harita ortamı kullanılmıştır. Çalışma ve gelişmeler hakkında daha yakından ve daha spesifik bir görüşe sahip olmak için sanal bir ortam geliştirilmiştir. Ortam, belirli sayıda koordinat ve engel içeren tanımlanmış bir ölçeklendirilmiş alandan oluşacaktır. Mobil robotun başlangıç noktası ve hedef noktası, ortamda önceden tanımlanacaktır. Her iki koordinat için ortam arazisinin büyüklüğü tanımlanacaktır. Bu nedenle, hem x hem de y koordinatları eşit ölçek oranından (örneğin 3m x 3m) oluşacaktır.

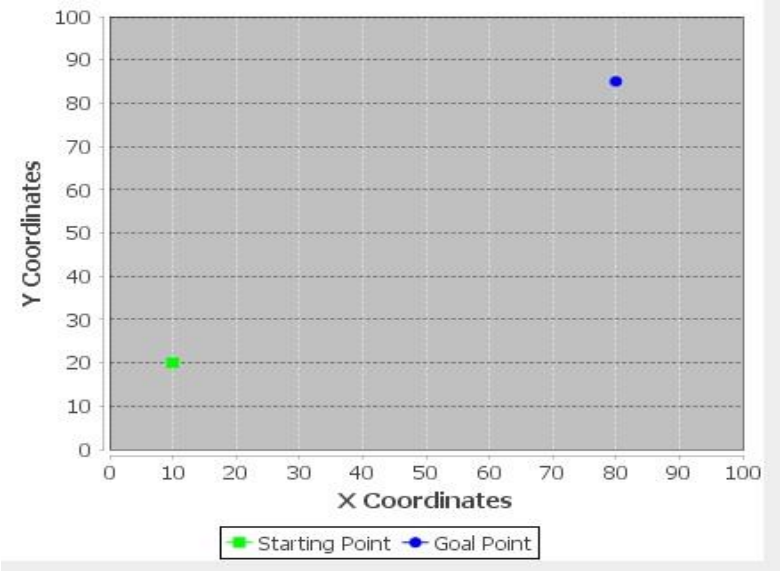
B. Izgara temsili

Önerilen kullanılan ızgara haritası, ortamı 2 Boyutlu bir uzayda temsil eder. Izgara temsili, her ızgara köşeli kenarlara sahip ve olası koordinatları içeren kare biçiminde tasarlanmıştır. Şebeke temsilinin amacı temel olarak ortamın arazisini belirlemek ve mobil robotun yörüngesinin daha net bir yolunu göstermektir. Haritadaki her koordinat, robotun yörüngesi için olası bir konum veya engelleyici bir engel olarak nitelendirilir. Mobil robotun yörüngesi, tanımlanan algoritmanın önerilen algoritma tarafından üretilen sonuçlarının yolunu izler. Şekil 2.1 önerilen ızgara haritasının bir grafik gösterimini vermektedir.

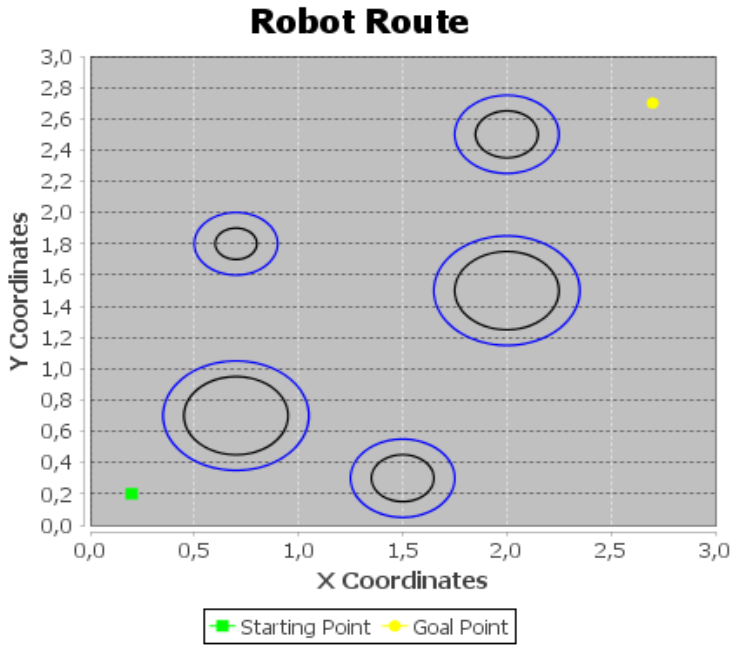
C. Engel temsili

Engellerin şekli veya boyutu farklı ortamlarda değişebilir, ancak bu çalışma için tanımlanmış koordinat merkez noktalarıyla her bir engeli temsil etmek için düzenli elips

şeklinde alanlar kullanılmıştır. Robotun başlangıç ve hedef noktaları / koordinatları dışında tanımlanan haritadaki herhangi bir koordinat noktasına bir engel tahsis edilebilir. Engeller statik ve hareketsizdir.



Şekil 2.1. Harita temsili



Şekil 2.2. Engeller ile harita

Tanımlanan sınırlı engel alanı, herhangi bir şekil veya büyüklükte bir engel teşkil edebilir. Bu nedenle, bir engelin boyutunu tanımlamak için, sınırlaması için belirli bir yarıçap rP tahsis edilir. Engel koordinat merkez noktası (x_i, y_i) ile ifade edilir. Engel koordinatlarının örnek bir grafiksel gösterimi Şekil 2.3.2'de verilmektedir.

2.3.1. Önerilen Algoritma Yapısı ve Yöntemleri

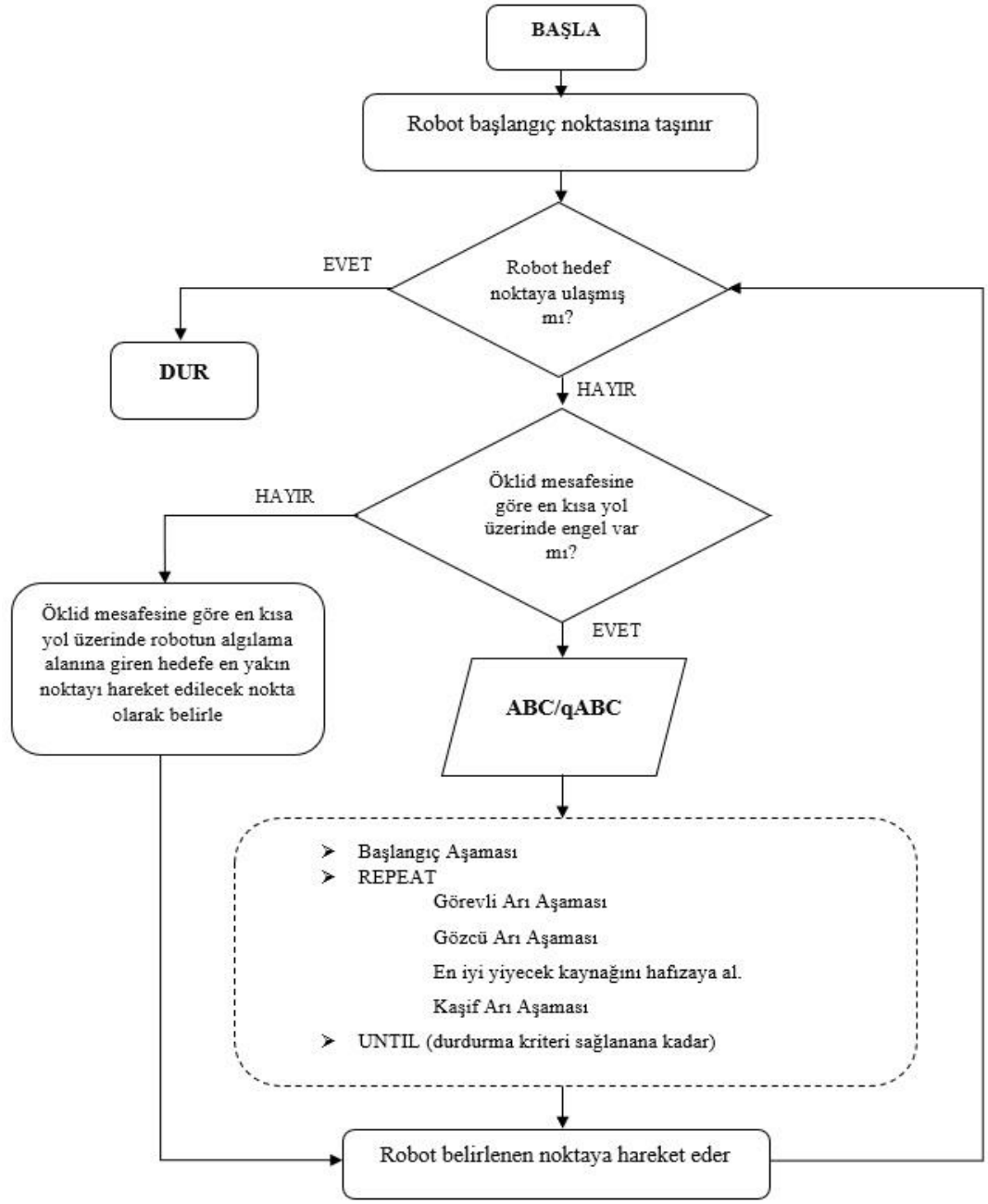
Önerilen çalışmanın temel amacı, tek bir mobil robot için en kısa yolu bulmak ve engeller ile çarpışmayı önlemektir. Bunu yaparken, [17]'de olduğu gibi robotun her bir adımında robot ve hedef arasında Öklid mesafesine göre en kısa yol üzerinde bir engel yoksa bu en kısa yol tercih edilmekte, varsa evrimsel hesaplamaya dayalı bir yöntemle belirlenen noktaya hareket edilmektedir. Robotun algılama alanında engel bulunması halinde [5]'te kullanılan bölgesel arama yöntemine benzer bir şekilde robotun yolu belirlenmekte ve [5]'ten farklı olarak sadece evrimsel hesaplamaya dayalı yöntem olarak standart ABC veya qABC algoritmaları kullanılmaktadır. Önerilen yapının akış diyagramı Şekil 2.3.3'te gösterilmektedir.

Sistemin ayarlanması için dahil olan parametreler ABC veya qABC algoritması parametreleri ve etkili bir otonom robot çevrimiçi bölgesel yol planlaması için kullanılan ek parametrelerdir. Bu ek parametreler şunları içerir:

- Ortam toplam arama alanı (ör. 3x3)
- Hareketli arama alanının yarıçapı
- Engellerin sayısı
- Fonksiyon denetleyicileri (c_1, c_2)
- Robot başlangıç noktası
- Robot hedef noktası

Robotun arama/algılama alanının dairesel olduğu varsayıldığından ABC veya qABC ile bir sonraki konum belirlenirken [5]'te olduğu gibi Denklem 2.6 ile çözümlerin başlangıç değerleri belirlenmiştir.

$$x_p = x_i + \varphi \cdot r \cdot \cos(\theta) \qquad (2.6)$$
$$y_p = y_i + \varphi \cdot r \cdot \sin(\theta)$$



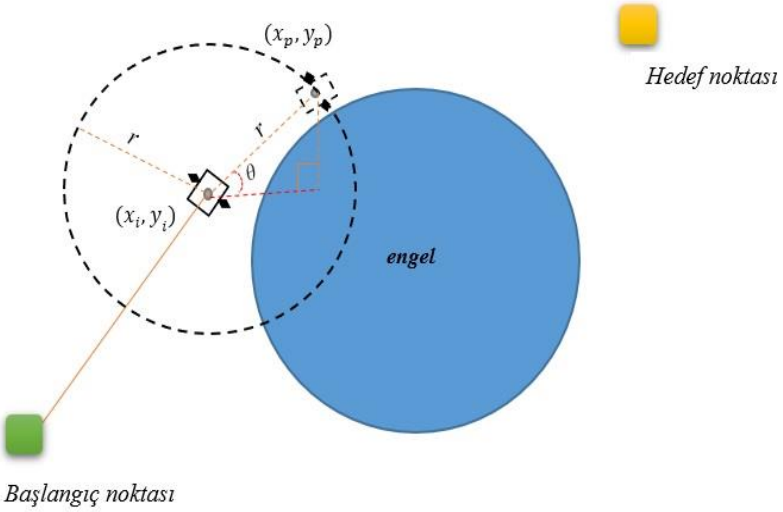
Şekil 2.3. Önerilen yaklaşımlar için akış şeması

Burada, φ [0, 1] arasındaki rastgele bir değeri, r hareketli arama alanının yarıçapını ve θ [0, 2π] arasındaki rastgele bir değeri ifade etmektedir. Not: Yarıçapın r hareketli arama alanı veya robotun n yiyecek boyutu ile orantılı olarak tanımlandığını unutmayın. Robotun n başlatılması şimdi aktif hale gelir ve potansiyel bir sonraki pozisyon veya yiyecek kaynağı potansiyeli olan bir arama alanı popülasyonu oluşturur. Robotu kontrol etmek için n, tanımlanmış hareketli arama alanını aşmamak için bir sınır kontrol tekniği

tanıtılmıştır. Bu tez çalışmasında, ABC ve qABC algoritmaları ile aday çözüm üretirken robotun algılama (arama) alanı dışında bir konum belirlenmesini engellemek için yeni bir teknik önerilmiştir. Bu tekniğe göre eğer üretilen aday çözüm robotun algılama alanı dışındaysa, robotun mevcut konumu (algılama alanının merkezi) ile aday çözümün arasındaki Öklid mesafesine göre en kısa yolu veren doğrultu ile algılama alanının sınırının kesişim noktası belirlenir ve aday çözüm bu sınıra çekilir.

2.3.2 Engeller ile Çarpışmayı Önleme

Robotun engellerle çarpışmasını önlemek için [5]'tekine benzer bir mekanizma kullanılır. Öklid mesafesine göre en uygun yol boyunca bir engel olup olmadığını belirlemek için, merkez noktasından sınırlandırılmış engel yarıçapı hesaplanır. Daha sonra, engelden kesinlikle kaçınmak için güvenlik sigortası olarak yarıçapa bir güvenlik değeri SV eklenir. Engel Obs_k'nin merkez koordinat noktaları hesaplanır ve her engelin güvenlik değeri ile birlikte toplam bir engel yarıçapı vardır. Toplam engel yarıçapı $= r_{obs} =$ $engelin\ fiziksel\ yarıçapı + \mathrm{SV}$. Robot hedefe doğru ilerlerken Öklid mesafesine göre en kısa yol üzerinde engel olup olmadığını anlamak için bu nokta ile engelin merkez noktası arasındaki mesafeyi hesaplar. Eğer bu mesafe r_{obs}'den daha büyükse engel ve bir çarpışma tehlikesi yok demektir, ancak r_{obs}'den küçük veya ona eşitse bu durumda o noktada engelle çarpışma durumu vardır. Robotun hareketliliği şemasındaki resim aşağıdaki Şekil 2.4'de gösterilmiştir.



Şekil 2.4. Bölgesel yol planlama hareketliliği sistemi

2.3.3. Amaç Fonksiyonu

Önerilen sistemin daha önce bahsedildiği gibi asıl amacı robot için en uygun yolu bulmak ve engellerin çarpışmasını önlemektir. Bu nedenle, amaç fonksiyonu [5]'teki gibi oluşturulmuş ve kullanılmıştır. Bu yöntemde, ana amaç fonksiyonu $Obj(n_p)$ hedef noktaya olan mesafeyi minimize etmeyi sağlayan $Obj_1(n_p)$ ve engel ile çarpışmayı önlemeyi sağlayan $Obj_2(n_p)$ alt amaç fonksiyonlarının birlikte kullanılması ile elde edilir. Burada $n_p = (x_i, y_i)$, yani robotun olası konumudur. Alt amaç fonksiyonlarının ikisi de belirli bir fonksiyon denetleyici ile çarpılır. Bu fonksiyon denetleyicileri (c_1, c_2), alt amaç fonksiyonlarının etkisini ayarlamaya yarayan pozitif ağırlık değerleridir. Fonksiyon denetleyicilerinin değerlerinin toplam değeri 1'e eşit olmalıdır. Bu nedenle, $\sum_{p=1}^{2} c_p = 1$'dir. Böylece robot için nihai amaç veya maliyet fonksiyonu daha önce belirtilen iki alt amaç fonksiyonunun birlikte kullanımı ile elde edilir [5]. Bu amaç fonksiyonu Denklem 2.7'de verilmiştir.

$$Obj(n_p) = c_1 * Obj_1(n_p) + c_2 * Obj_2(n_p) \tag{2.7}$$

Hedef noktası için amaç fonksiyonu Denklem 2.8'de verilmiştir.

$$Obj_1(n_p) = \|n_p - Hedef\| \tag{2.8}$$

Hedef tanımlanmış hedef noktasının (x_g, y_g) konumunu belirtir. Burada, $Obj_1(n_p)$'nin minimizasyonu ile mobil robot ve hedef arasındaki mesafe en aza indirilmeye çalışılır. Engel ile çarpışmayı engellemek için engellerden uzaklaşmayı sağlayan amaç fonksiyonu Denklem 2.9 ve 2.10'de gösterilmiştir.

$$Obj_{2,k}(n_p) = \begin{cases} 0, & \|n_p - Obs_k\| > r_{obs_k}, k = 1,2 \ldots . h \\ \dfrac{1}{\|n_p - Obs_k\|} - \dfrac{1}{r_{obs_k}}, & \|n_p - Obs_k\| \leq r_{obs_k}, k = 1,2 \ldots . h \end{cases} \tag{2.9}$$

$$Obj_2(n_p) = \sum_{k=1}^{h} Obj_{2,k}(n_p) \tag{2.10}$$

Burada, Obs_k ortamdaki k'ninci engelin merkez noktasını ifade etmektedir, h toplam engel sayısını, n_p robotun mevcut pozisyonunu göstermektedir. r_{obs_k} k'ninci engelin toplam yarıçapıdır.

3. BÖLÜM

DENEYSEL ÇALIŞMALAR VE TARTIŞMA

Bu bölümde önerilen yöntemler farklı senaryolarda test edilmiş ve performansları kıyaslanmıştır.

3.1. Deneysel Çalışmalar

Deneysel çalışmalar statik engeller ile statik bir ortamda gerçekleştirilmiştir. Her engel eliptik bir şekilde temsil edilmektedir ve belirli bir yarıçapa sahiptir. Bir adet mobil robot mevcuttur ve belirtilen bir başlangıç noktasında durağan bir hedefe doğru yol planlamasını çevrim içi (online) bir şekilde yaparak ilerlemektedir.

Bu deneysel çalışmada farklı ortamlarda 5 farklı senaryo dikkate alınmıştır. Senaryolar önerilen algoritmaların başarılarını farklı durumlarda inceleyebilmek için farklı sayıda ve pozisyonlarda engeller kullanılarak, farklı başlangıç ve bitiş noktalarıyla yapılandırılmıştır. Bazı senaryolarda, mobil robotu en iyi yolu bulma konusunda zorlamak için engeller bilinçli bir şekilde Öklid yolu boyunca kümelenmiştir. İlk senaryo, farklı boyutlarda 3 statik engelden oluşmaktadır. Engeller sırasıyla 0.15, 0.25, 0.25 yarıçapına sahiptir. İkinci senaryo, daha yoğun kümelenmelerle farklı boyutlarda 5 statik engelden oluşmaktadır. Bu senaryoda, engeller sırasıyla 0.1, 0.15, 0.15, 0.25, 0.25 yarıçapına sahiptir. Üçüncü senaryoda 5 büyük statik engel ve daha yoğun kümeler bulunmaktadır. Engeller sırasıyla 0.25, 0.35, 0.35, 0.35 yarıçapına sahiptir. Dördüncü ve beşinci senaryolarda daha yüksek başlangıç noktaları ve kümelenmiş engeller mevcuttur. Dördüncü senaryoda, engeller sırasıyla 0.15, 0.25, 0.35, 0.35 yarıçapına sahiptir. Beşinci senaryoda, engeller sırasıyla 0.15, 0.25, 0.25, 0.25, 0.35 yarıçapına sahiptir. Ortam haritası boyutu tüm senaryolarda $3m \times 3m$ olarak değişmeden kalır. Bütün senaryolarda $c_1 = c_2 = 0.5$'tir ve robot için algılama yarıçapı (hareketli arama alanının yarıçapı) 0.146'dır. Senaryo 1-5 için robotun başlangıç noktadan hedef noktaya Öklid mesafelerin

değerleri, 2.2847m, 4.2426m, 3.7537m, 2.9069m, 3.1906m sırasıyla hesaplanmıştır.

3.2. ABC ve qABC Algoritmalarının Kontrol Parametre Değerleri

Deneysel çalışmada kullanılan algoritmalar için kontrol parametreleri Tablo 3.1'de gösterilmiştir.

Tablo 3.1. ABC ve qABC algoritmalarının kontrol parametre değerleri

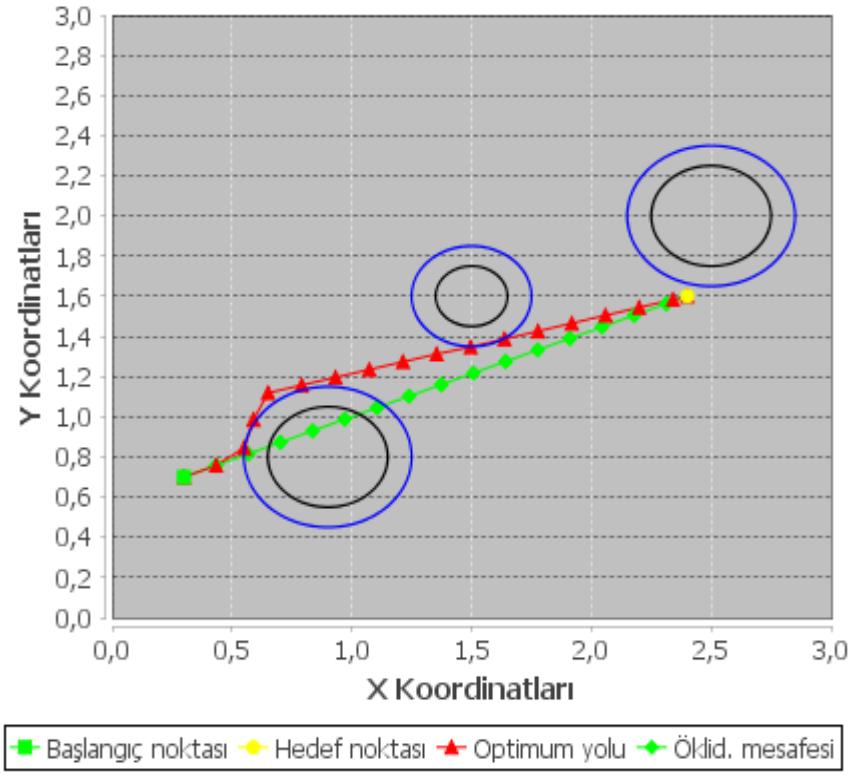
Parametre	Değer	
	ABC	qABC
Yiyecek kaynağı sayısı	10	10
Problemin boyutu	2	2
Değerlendirme sayısı	200, 2000	200, 2000
Limit	20	20
Komşuluk yarıçapı (r)	–	1

3.3 Ele Alınan Senaryolar ve Simülasyon Sonuçları

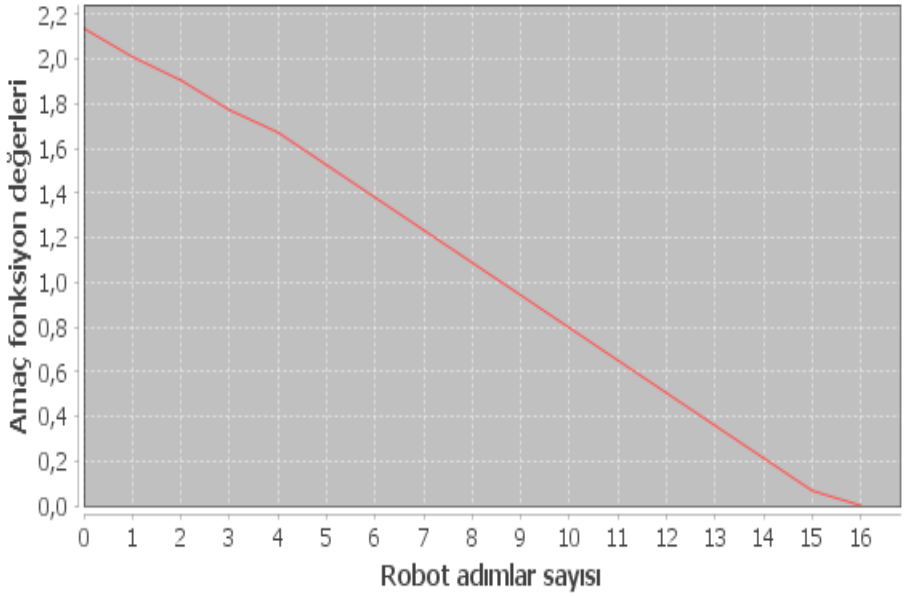
Önerilen yaklaşımların performansı ve etkinliği, otonom mobil robot için en iyi yolu geliştirmek için test edilmiştir. Senaryo 1-5 için robotun başlangıç konumu, hedef konumu ve engel sayısı Tablo 3.2, Tablo 3.4, Tablo 3.6, Tablo 3.8, Tablo 3.10 de sırasıyla belirtilmektedir. Tüm senaryolar her bir parametre seti ile 30'ar kez çalıştırılmış ve bu 30 çalıştırma için elde edilen ortalama mesafe, bu mesafelerin standart sapması, en iyi mesafe ve en kötü mesafe değerleri Tablo 3.2, Tablo 3.4, Tablo 3.6, Tablo 3.8, Tablo 3.10.'da sırasıyla senaryo 1-5 için verilmiştir. Bu tablolarda daha iyi değerler koyu font ile vurgulanmıştır. Senaryo 1-5 için 200 değerlendirme ile 30 koşma arasından en iyi sonuç için robotun izlediği yol Şekil 3.1, Şekil 3.5, Şekil 3.9, Şekil 3.13, Şekil 3.17'te ABC için, Şekil 3.3, Şekil 3.7, Şekil 3.11, Şekil 3.15, Şekil 3.19'te qABC için sırasıyla gösterilmiştir. Benzer şekilde senaryo 1-5 için robotun adımları boyunca elde edilen amaç fonksiyonu değerleri Şekil 3.2, Şekil 3.6, Şekil 3.10, Şekil 3.14, Şekil 3.18'da ABC için, Şekil 3.4, Şekil 3.8, Şekil 3.12, Şekil 3.16 Şekil 3.20'da qABC için sırasıyla verilmiştir.

Tablo 3.2. Senaryo 1 için parametre değerleri

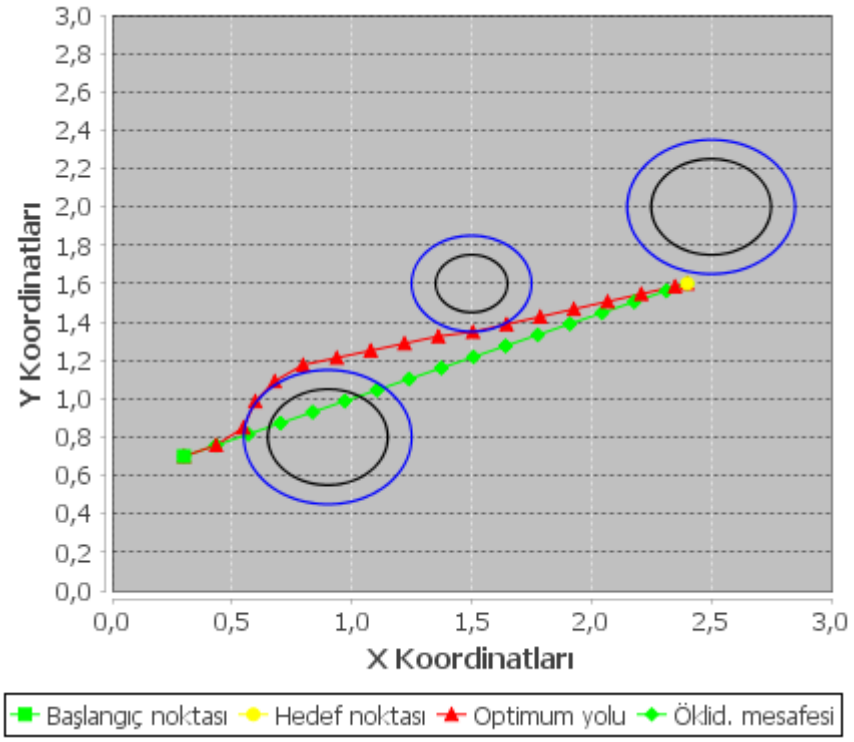
Parametre	Değer
Robot başlangıç noktası (x, y)	(0.3, 0.7)
Robot hedef noktası (x, y)	(2.4, 1.6)
Engel Sayısı	3



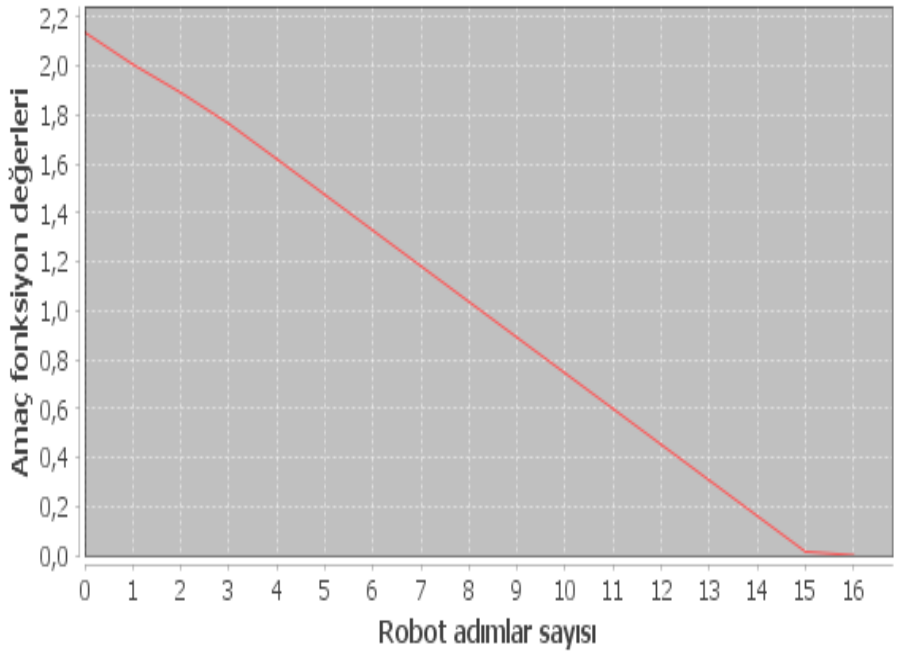
Şekil 3.1. Robot yolu (ABC Senaryo 1)



Şekil 3.2. Robotun adımları boyunca elde edilen amaç fonksiyonu değerleri (ABC Senaryo 1)



Şekil 3.3. Robot yolu (qABC Senaryo 1)



Şekil 3.4. Robotun adımları boyunca elde edilen amaç fonksiyonu değerleri (qABC Senaryo 1)

Tablo 3.3. 30 Çalıştırmada Gezilen MRYP Optimal Mesafeleri (Senaryo 1)

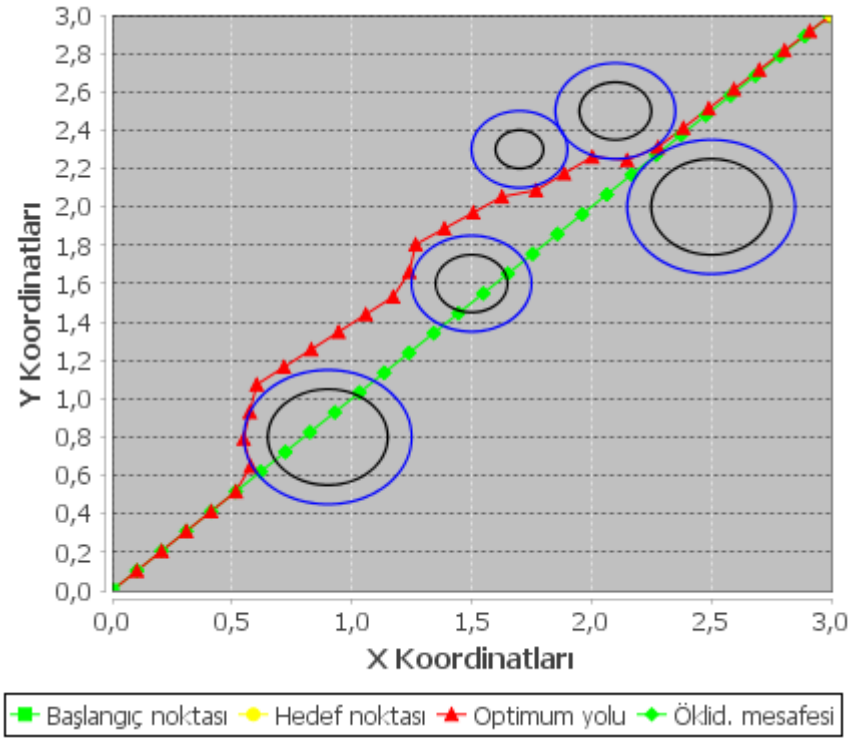
Değerlendirme Sayısı		ABC	qABC
200	Ortalama mesafe	2.3827	***2.3789***
	Std. Sapma	0.0176	0.0154
	En iyi mesafe	2.3566	***2.3538***
	En kötü mesafe	2.4239	***2.4235***
	Robot adımlarının ortalaması	17.03	***17.0***
	30/30'da hedef noktasına ulaşıldı	EVET	EVET
2000	Ortalama mesafe	2.3623	***2.3622***
	Std. Sapma	0.0021	0.0020
	En iyi mesafe	***2.3572***	2.3590
	En kötü mesafe	2.3690	***2.3664***
	Robot adımlarının ortalaması	17.0	17.0
	30/30'da hedef noktasına ulaşıldı	EVET	EVET

Tablo 3.4. Senaryo 2 için parametre değerleri

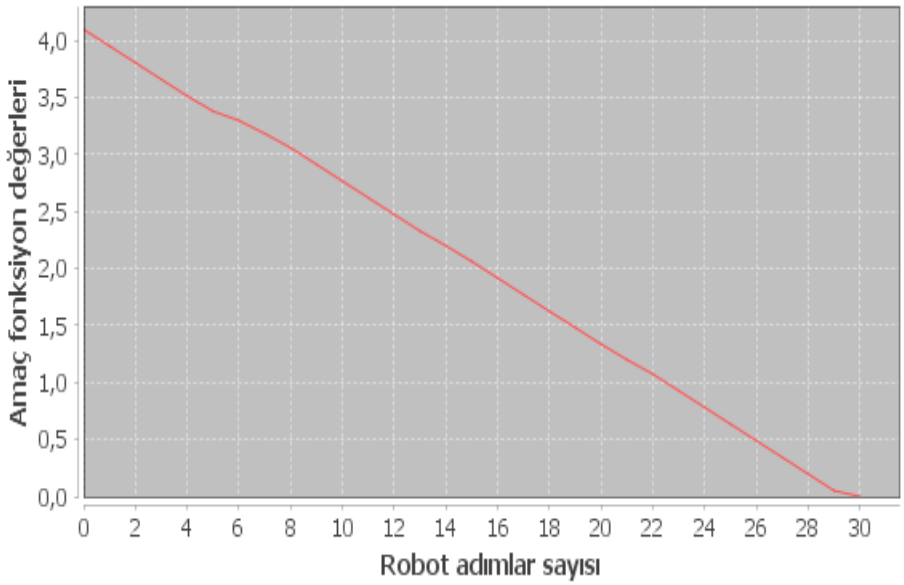
Parametre	Değer
Robot başlangıç noktası (x, y)	(0, 0)
Robot hedef noktası (x, y)	(3, 3)
Engel Sayısı	5

3.4. Tartışma

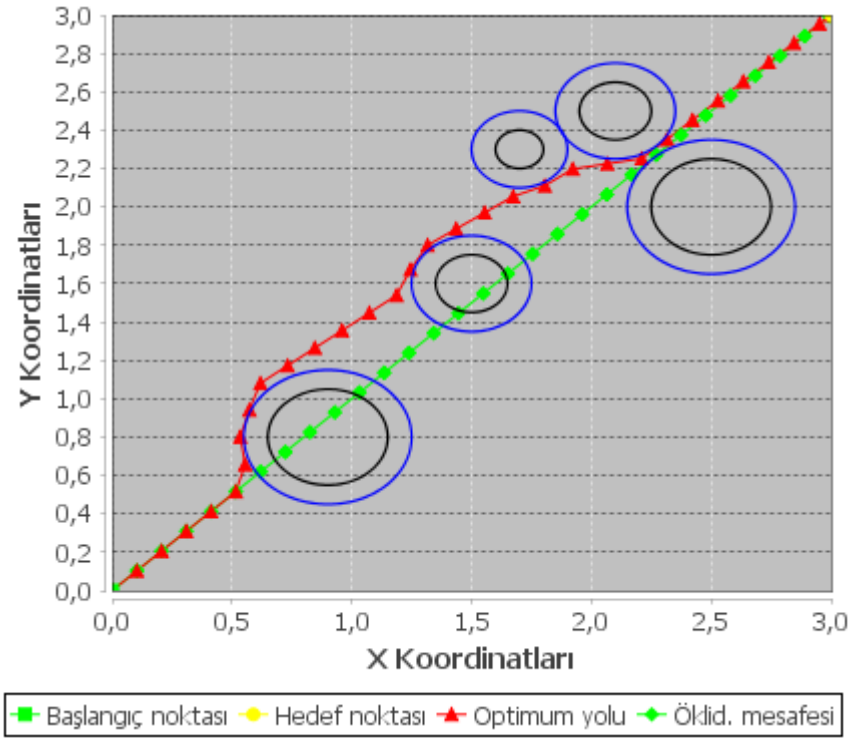
Deneysel çalışmanın 5 farklı senaryosu vardı. Tüm senaryolar statik engellerin bulunduğu statik bir ortamda test edildi. Çalışmamızın 2. bölümünde belirtildiği gibi, mobil robot bir engel bulunmadıkça Öklid yolunda hareket etmeye devam eder, daha sonra eğer mobil robotu bir engel keşfederse, uygun bir yol bulmak ve hedef noktaya ulaşmak için algoritmayı (ABC veya qABC) çağırır. Her program çalışmasında, mobil robotun toplam optimum mesafesi ve robotun hedef noktaya ulaşmak için attığı toplam adım sayısı hesaplanır ve hafızada alınır. 30 çalışmanın sonunda, mobil robotun koşu başına toplam optimal mesafenin ortalaması, koşu başına toplam optimal mesafenin standart sapması, en iyi optimal mesafe, en kötü mesafe, koşu başına robot adımlarının ortalaması hesaplanır. Bu değişkenler, iki algoritma (ABC ve qABC) arasındaki performansları karşılaştırmak için kullanılır.



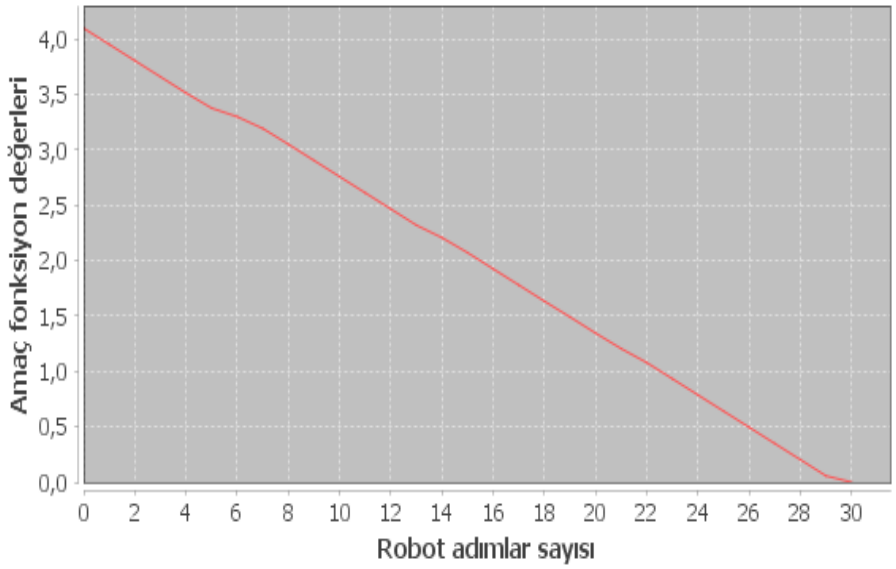
Şekil 3.5. Robot yolu (ABC Senaryo 2)



Şekil 3.6. Robotun adımları boyunca elde edilen amaç fonksiyonu değerleri (ABC Senaryo 2)



Şekil 3.7. Robot yolu (qABC Senaryo 2)



Şekil 3.8. Robotun adımları boyunca elde edilen amaç fonksiyonu değerleri (qABC Senaryo 2)

Birinci senaryoda, her iki algoritma da statik bir ortamda 3 statik engelle test edildi. Mobil

robot başlangıç noktası ve hedef noktası, (0.3, 0.7) ve (2.4, 1.6)'dir. 30 çalışmadan sonra, Tablo 3.2, 200 ve 2000 değerlendirmelerinde hem ABC hem de qABC performansının sonuçlarını göstermektedir. Şekil 3.1'den Şekil 3.4'e kadar, en iyi çalışmanın, mobil robot yolunu ve robot adımları boyunca amaç fonksiyonu değerlerini veren grafiği gösterir (hem ABC hem de qABC için). Her iki algoritma da başarılı sonuçlara ulaştı, hedef noktaya ulaştı ve engellerden kaçındı. Ancak, qABC'nin daha açgözlü davranış nedeniyle, ABC'den daha iyi bir sonuç elde etti.

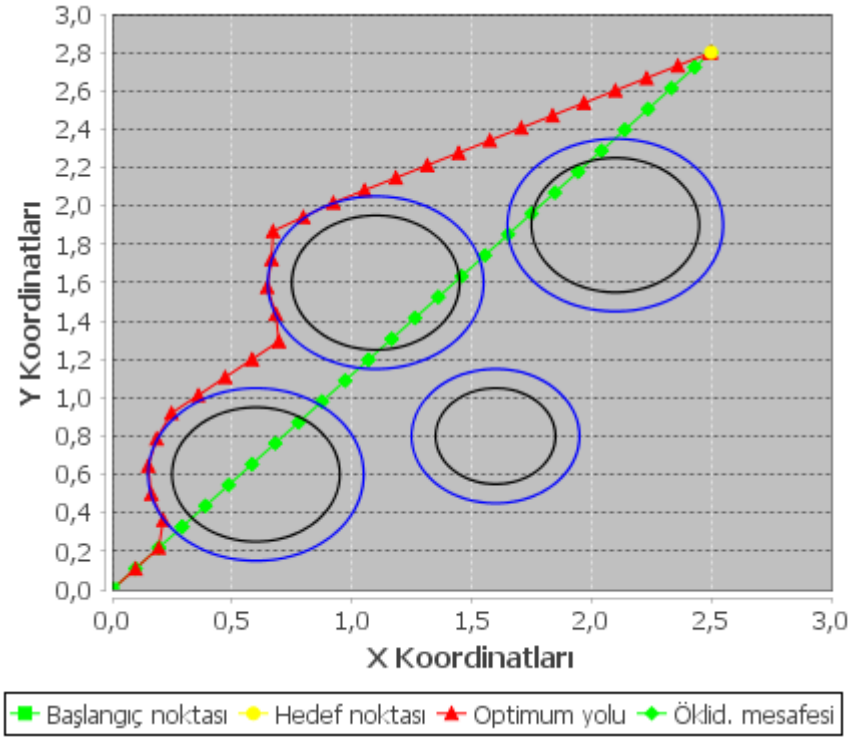
Tablo 3.5. 30 Çalıştırmada gezilen MRYP Optimal mesafeleri (Senaryo 2)

Değerlendirme Sayısı		ABC	qABC
200	Ortalama mesafe	4.4672	***4.4630***
	Std. Sapma	0.0250	0.0182
	En iyi mesafe	***4.4277***	4.4311
	En kötü mesafe	4.5753	***4.5033***
	30/30'da hedef noktasına ulaşıldı	31.34	***31.33***
	Optimal çözüme ulaşıldı	EVET	EVET
2000	Ortalama mesafe	4.4347	***4.4344***
	Std. Sapma	0.0019	0.0015
	En iyi mesafe	***4.4279***	4.4307
	En kötü mesafe	4.4378	***4.4365***
	Robot adımlarının ortalaması	31.0	31.0
	30/30'da hedef noktasına ulaşıldı	EVET	EVET

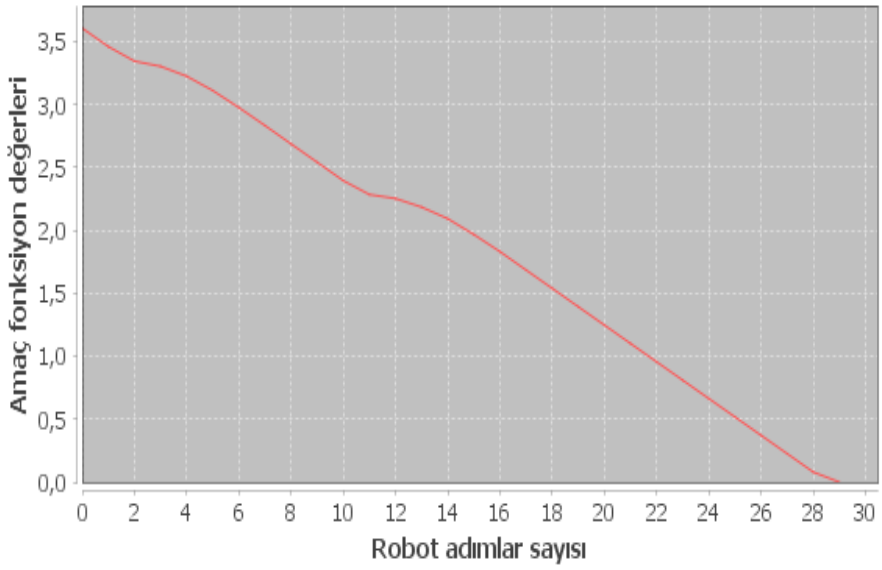
Tablo 3.6. Senaryo 3 için parametre değerleri

Parametre	Değer
Robot başlangıç noktası (x, y)	(0, 0)
Robot hedef noktası (x, y)	(2.5, 2.8)
Engel Sayısı	4

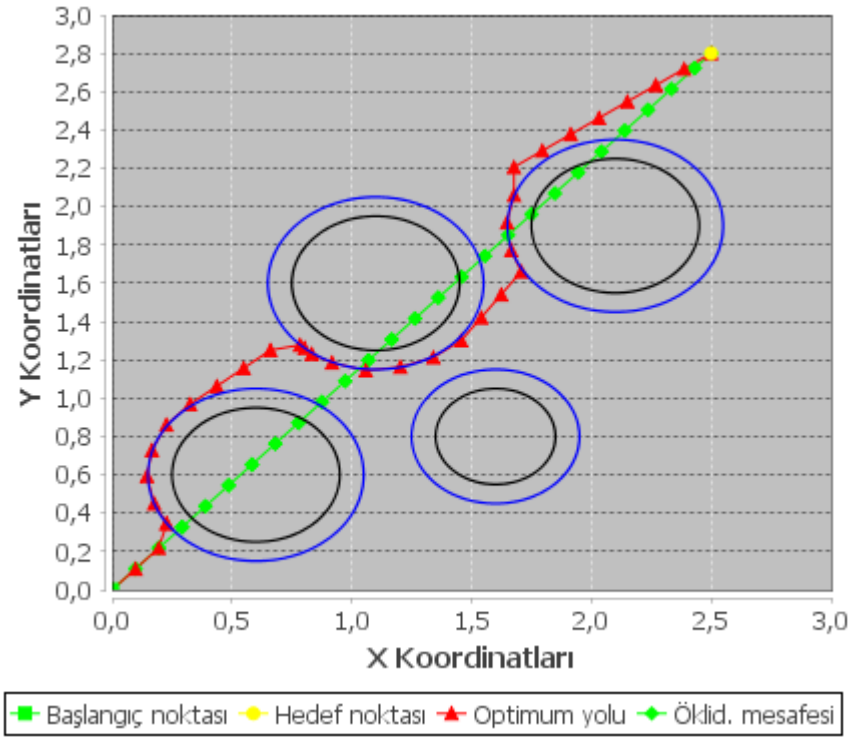
İkinci senaryoda, her iki algoritma da robotu daha uygun bir yol bulmaya zorlamak için 5 daha karmaşık statik engel içeren statik bir ortamda test edildi. Mobil robot başlangıç noktası ve hedef noktası, (0.0, 0.0) ve (3.0, 3.0)'dir. 30 çalışmadan sonra, Tablo 3.4, 200 ve 2000 değerlendirmelerinde hem ABC hem de qABC performansının sonuçlarını göstermektedir. Şekil 3.5'den Şekil 3.8'e kadar, en iyi çalışmanın, mobil robot yolunu ve robot adımları boyunca amaç fonksiyonu değerlerini veren grafiği gösterir (hem ABC hem de qABC için). Her iki algoritma da başarılı sonuçlara ulaştı, hedef noktaya ulaştı ve engellerden kaçındı. Ancak, qABC tekrar ABC'den biraz daha iyi sonuç gösterdi.



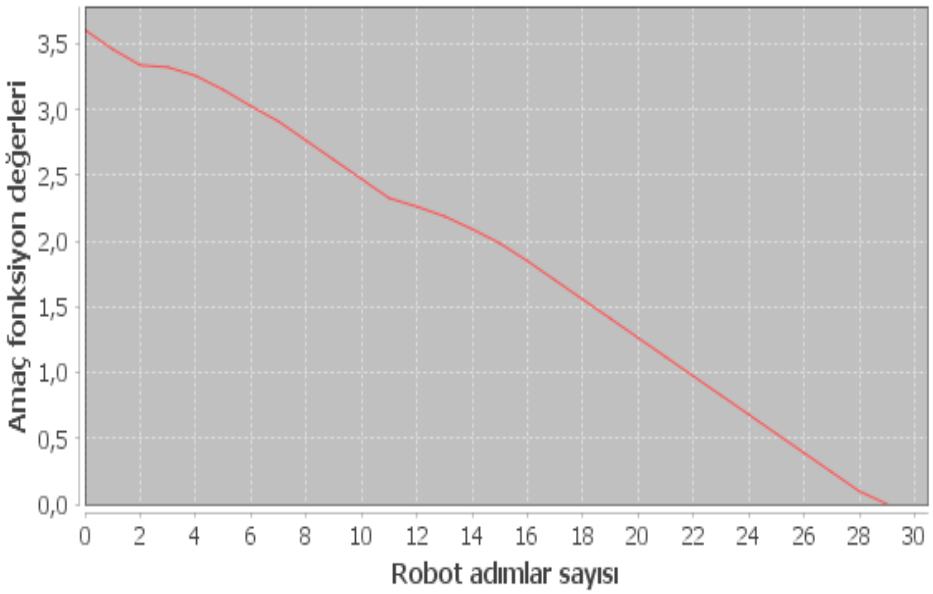
Şekil 3.9. Robot yolu (ABC Senaryo 3)



Şekil 3.10. Robotun adımları boyunca elde edilen amaç fonksiyonu değerleri (ABC Senaryo 3)



Şekil 3.11. Robot yolu (qABC Senaryo 3)



Şekil 3.12. Robotun adımları boyunca elde edilen amaç fonksiyonu değerleri (qABC Senaryo 3)

Tablo 3.7. 30 Çalıştırmada gezilen MRYP optimal mesafeleri (Senaryo 3)

Değerlendirme Sayısı		ABC	qABC
200	Ortalama mesafe	***4.2622***	4.2637
	Std. Sapma	0.0691	0.0742
	En iyi mesafe	***4.1857***	4.2055
	En kötü mesafe	4.5109	***4.4788***
	Robot adımlarının ortalaması	***30.48***	30.75
	30/30'da hedef noktasına ulaşıldı	EVET	EVET
2000	Ortalama mesafe	4.2227	***4.2128***
	Std. Sapma	0.0331	0.0040
	En iyi mesafe	4.2015	***4.1989***
	En kötü mesafe	4.3471	***4.2163***
	Robot adımlarının ortalaması	29.06	29.06
	30/30'da hedef noktasına ulaşıldı	EVET	EVET

Tablo 3.8. Senaryo 4 için parametre değerleri

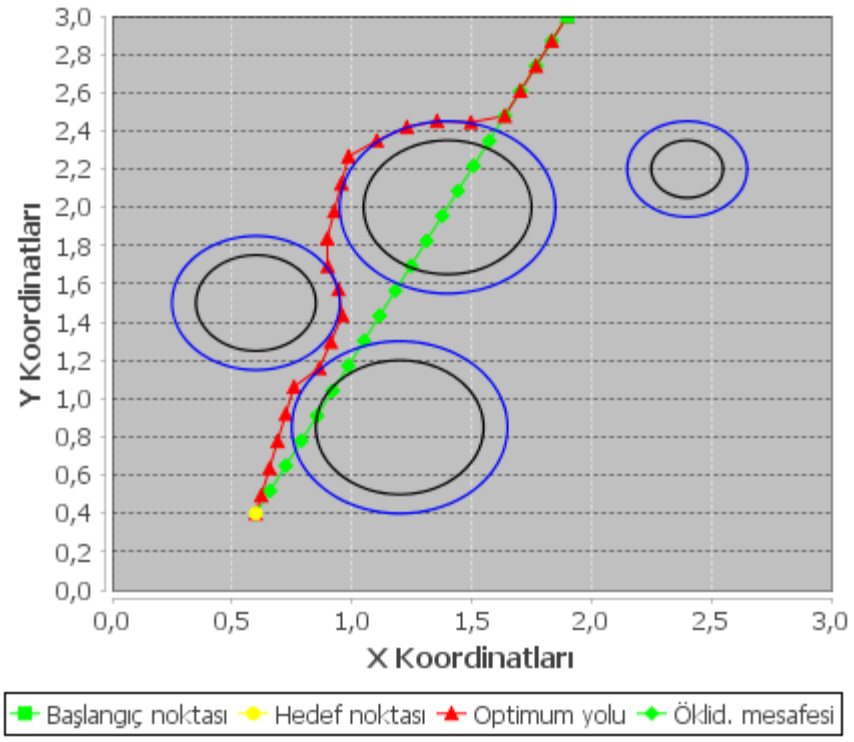
Parametre	Değer
Robot başlangıç noktası (x, y)	(1.9, 3.0)
Robot hedef noktası (x, y)	(0.6, 0.4)
Engel sayısı	4

Üçüncü senaryoda, ABC ve qABC daha fazla çalışmak için statik bir ortamda 4 daha büyük statik engel ile test edildi. Mobil robot başlangıç noktası ve hedef noktası, (0.0, 0.0) ve (2.5, 2.8)'dir. 30 çalışmadan sonra, tablo 3.6, 200 ve 2000 değerlendirmelerinde hem ABC hem de qABC performansının sonuçlarını göstermektedir. Şekil 3.9'den Şekil 3.12'e kadar, en iyi çalışmanın, mobil robot yolunu ve robot adımları boyunca amaç fonksiyonu değerlerini veren grafiği gösterir (hem ABC hem de qABC için). Her iki algoritma da başarılı sonuçlara ulaştı, hedef noktaya ulaştı ve engellerden kaçındı. Ancak, bu senaryoda, ABC 200 değerlendirmede daha iyi sonuçlara ulaştı ve qABC 2000 değerlendirmesinde daha iyi sonuçlara ulaştı.

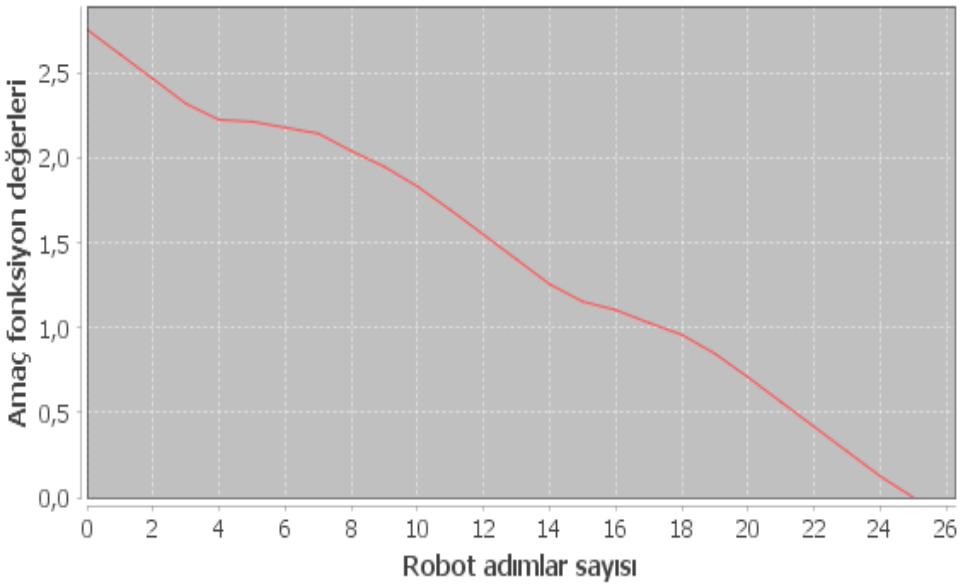
Dördüncü senaryoda, her iki algoritma da 4 statik engel içeren statik bir ortamda test edildi. Mobil robot başlangıç noktası ve hedef noktası, (1.9, 3.0) ve (0.6, 0.4)'tür. 30 çalışmadan sonra, Tablo 3.8, 200 ve 2000 değerlendirmelerinde hem ABC hem de qABC performansının sonuçlarını göstermektedir. Şekil 3.13'den Şekil 3.16'e kadar, en iyi çalışmanın, mobil robot yolunu ve robot adımları boyunca amaç fonksiyonu değerlerini veren grafiği gösterir (hem ABC hem de qABC için). Her iki algoritma da başarılı

sonuçlara ulaştı, hedef noktaya ulaştı ve engellerden kaçındı. Ancak, ABC 200 değerlendirmede tekrar daha iyi sonuçlara ulaştı ve qABC 2000 değerlendirmesinde daha iyi sonuçlara ulaştı.

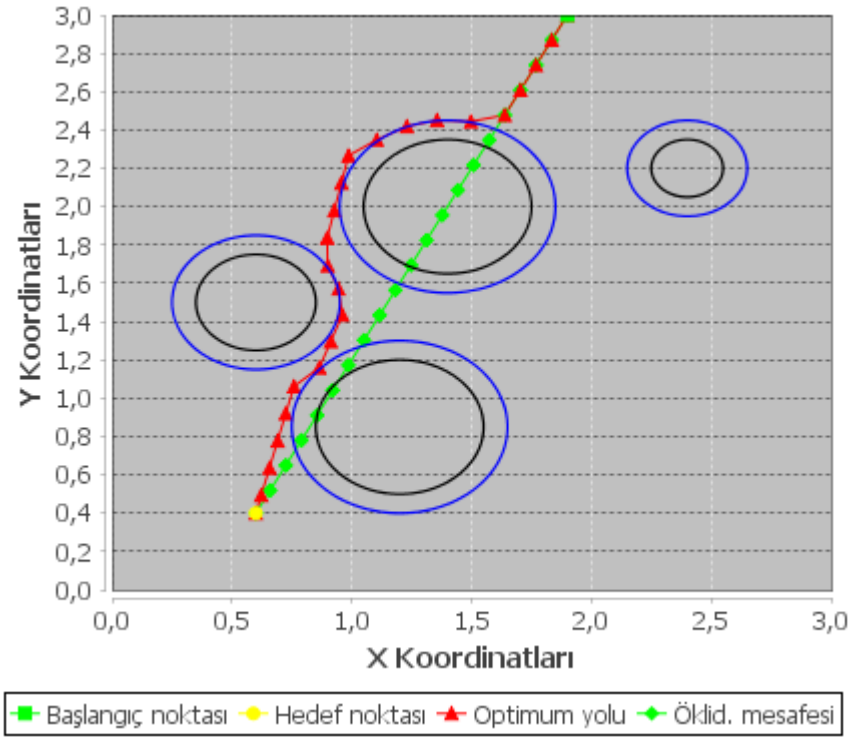
Beşinci senaryoda, her iki algoritma da 5 statik engel içeren statik bir ortamda test edildi. Burada, programın uygulanabilir bir yol bulunup bulunmadığını görmek için 3 engel daha kümelenmiştir. Mobil robot başlangıç noktası ve hedef noktası, (2.9, 2.2) ve (0.2, 0.5)'tir. 30 çalışmadan sonra, Tablo 3.10, 200 ve 2000 değerlendirmelerinde hem ABC hem de qABC performansının sonuçlarını göstermektedir. Şekil 3.17'den Şekil 3.20'ye kadar, en iyi çalışmanın, mobil robot yolunu ve robot adımları boyunca amaç fonksiyonu değerlerini veren grafiği gösterir (hem ABC hem de qABC için). Her iki algoritma da başarılı sonuçlara ulaştı, hedef noktaya ulaştı ve engellerden kaçındı. Bu senaryoda, qABC ABC'den daha iyi bir performansa gösterdi.



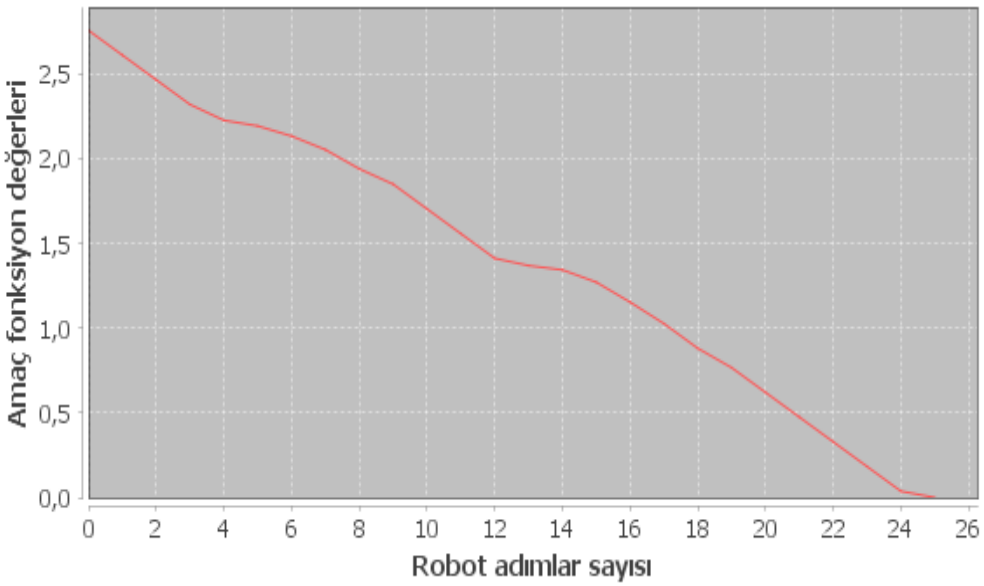
Şekil 3.13. Robot yolu (ABC Senaryo 4)



Şekil 3.14. Robotun adımları boyunca elde edilen amaç fonksiyonu değerleri (ABC Senaryo 4)



Şekil 3.15. Robot yolu (qABC Senaryo 4)



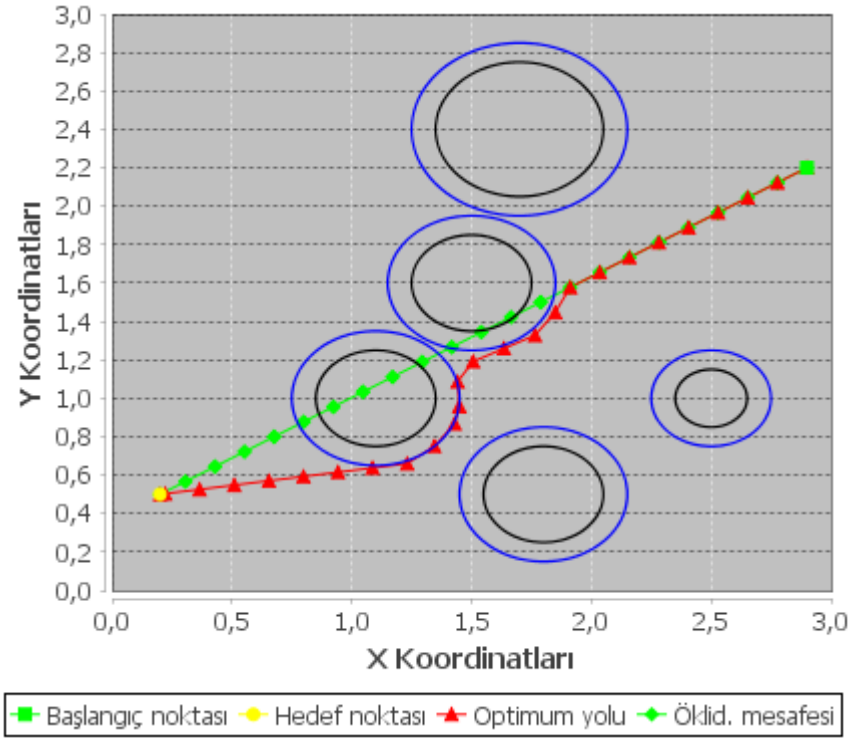
Şekil 3.16. Robotun adımları boyunca elde edilen amaç fonksiyonu değerleri (qABC Senaryo 4)

Tablo 3.9. 30 Çalıştırmada gezilen MRYP optimal mesafeleri (Senaryo 4)

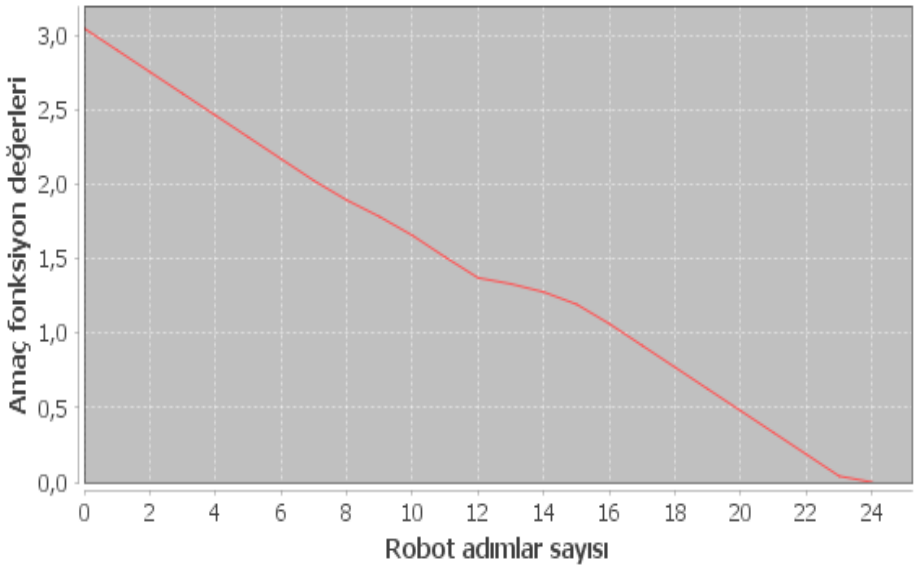
Değerlendirme Sayısı		ABC	qABC
200	Ortalama mesafe	***3.2932***	3.3006
	Std. Sapma	0.0468	0.0477
	En iyi mesafe	3.2438	***3.2301***
	En kötü mesafe	***3.4078***	3.4244
	Robot adımlarının ortalaması	***24.46***	24.57
	30/30'da hedef noktasına ulaşıldı	EVET	EVET
2000	Ortalama mesafe	3.2295	***3.2189***
	Std. Sapma	0.0250	0.0099
	En iyi mesafe	3.2077	***3.2049***
	En kötü mesafe	3.3538	***3.2530***
	Robot adımlarının ortalaması	23.03	***22.96***
	30/30'da hedef noktasına ulaşıldı	EVET	EVET

Tablo 3.10. Senaryo 5 için parametre değerleri

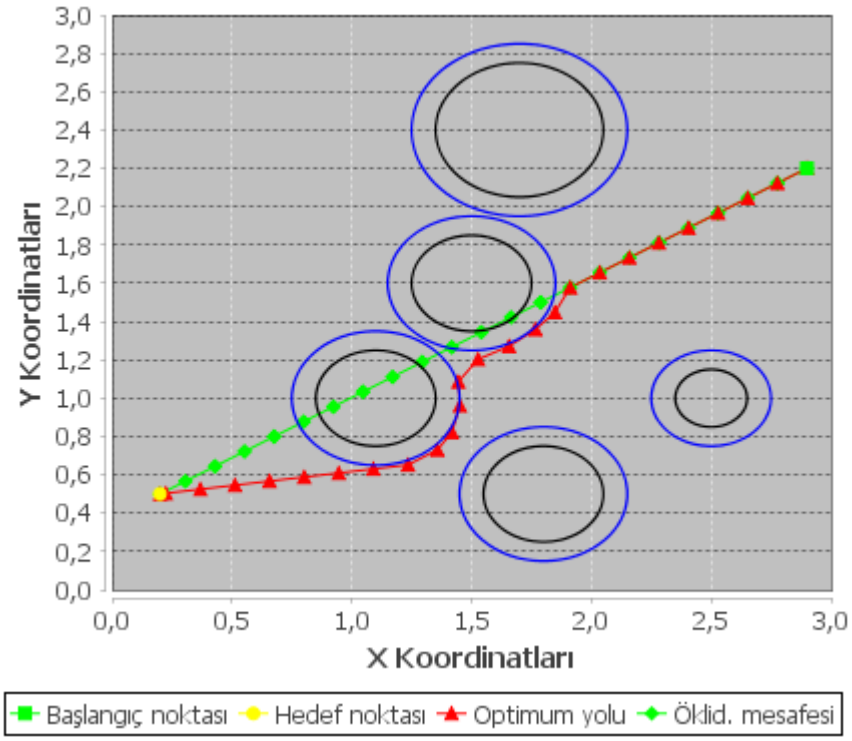
Parametre	Değer
Robot başlangıç noktası (x, y)	(2.9, 2.2)
Robot hedef noktası (x, y)	(0.2, 0.5)
Engel sayısı	5



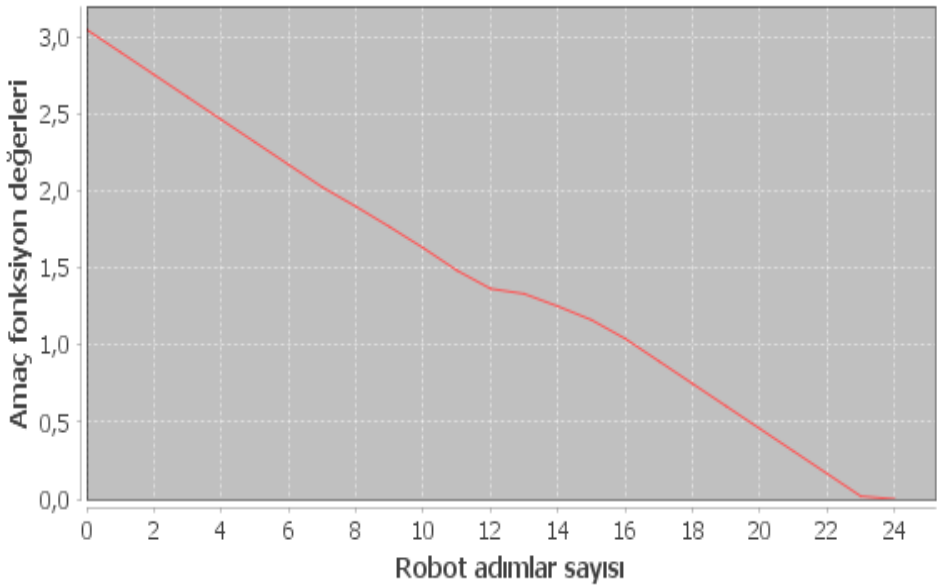
Şekil 3.17. Robot yolu (ABC Senaryo 5)



Şekil 3.18. Robotun adımları boyunca elde edilen amaç fonksiyonu değerleri (ABC Senaryo 5)



Şekil 3.19. Robot yolu (qABC Senaryo 5)



Şekil 3.20. Robotun adımları boyunca elde edilen amaç fonksiyonu değerleri (qABC Senaryo 5)

Tüm deney senaryolarında, her iki algoritma da mobil robot için en uygun yolları bulmuş,

hedef noktaya ulaşmış ve engellerden kaçınmıştır. Ancak, her bir senaryo için alınan 30 koşma sonucunda elde edilen mesafe değerlerinin ortalamalarına göre 200 değerlendirme dikkate alındığında qABC beş senaryonun üçünde, 2000 değerlendirme için ise bütün senaryolarda ABC'den daha başarılı olmuştur.

Tablo 3.11. 30 Çalıştırmada gezilen MRYP optimal mesafeleri (Senaryo 5)

Değerlendirme Sayısı		ABC	qABC
200	Ortalama mesafe	3.4644	***3.4601***
	Std. Sapma	0.0223	0.0293
	En iyi mesafe	3.4240	***3.4019***
	En kötü mesafe	***3.5095***	3.5393
	Robot adımlarının ortalaması	24.88	***24.86***
	30/30'da hedef noktasına ulaşıldı	EVET	EVET
2000	Ortalama mesafe	3.4225	***3.4157***
	Std. Sapma	0.0100	0.0087
	En iyi mesafe	3.4050	***3.3967***
	En kötü mesafe	3.4404	***3.4316***
	Robot adımlarının ortalaması	24.0	24.0
	30/30'da hedef noktasına ulaşıldı	EVET	EVET

4. BÖLÜM

SONUÇ ve ÖNERİLER

Bu tez çalışmasında, mobil robotlar için insan kontrolü olmadan, durağan engellerin ve durağan bir hedefin bulunduğu bir ortamda bölgesel yol planlaması için ABC tabanlı iki yöntem önerilmiştir. Bunlardan biri standart ABC diğeri ise qABC algoritmasını kullanmaktadır. Bu yöntemlerde, eğer robot ile hedef arasındaki en kısa mesafenin olduğu doğrultuda bir engel yoksa, robot bu doğrultuda hareket etmektedir. Eğer bu doğrultuda bir engel varsa robot gideceği noktayı bulmak için ABC veya qABC algoritmalarını çalıştırmaktadır. Burada kullanılan amaç fonksiyonu engellerden uzaklaşmayı ve hedefe yaklaşmayı sağlayan yapılar içermektedir.

Deneysel çalışmada hem ABC hem de qABC çok iyi performanslara sahiptir. Her iki algoritma da mobil robot için en uygun yolları bulmuş ve hedef noktaya ulaşmıştır. Senaryolarda 1,2 ve 5 qABC hem 200 hem de 2000 değerlendirmelerinde ABC'ye göre daha iyi performans göstermiştir. Bununla birlikte, 3. ve 4. senaryolarda, ABC 200 değerlendirmede daha iyi performans göstermiştir. Sonuç olarak qABC ABC'den daha iyi bir performans göstermekle birlikte, simülasyon sonuçları her iki algoritma ile önerilen yaklaşımların mobil robotların bölgesel yol planlamasında kullanılabileceğini göstermiştir.

Gelecekteki çalışmalarda, geliştirilen sistemi daha sonraki gelişmelerde kullanılabilir ve geliştirilebilir. Birden fazla hedef noktasıyla uyarlanabilen dinamik ortamlarda birden fazla robot için yol planlaması yapılabilir. Ortamın boyutsal yapısı 2D'den farklı olabilir. Nesnelerin interneti (Industrial Internet of Things- IIoT) teknolojilerinin, siber güvenlik istihbaratının, askeri savunma sistemlerinin, otomasyon mimarilerinin, ticari dağıtım sistemlerinin mevcut kullanımı da önerilen sisteme dahil edilebilir. Önerilen yaklaşımlar daha farklı senaryolarda literatürde yer alan diğer yöntemlerle kıyaslanabilir. Kontrol

parametrelerinin değerlerini belirlemek için deney tasarımı yapılabilir.

KAYNAKÇA

[1] Karaboga, D., 2005. An idea based on honey bee swarm for numerical optimization, Technical Report TR06, Erciyes University, Engineering Faculty, Computer Engineering Department.

[2] Karaboga, D., 2010. Artificial bee colony algorithm, **Scholarpedia, 5**(3): 6915.

[3] Karaboga, D., Basturk, B., 2008. On the performance of artificial bee colony (ABC) algorithm, **Applied Soft Computing, 8**: 687–697

[4] Faridi, A., Sharma, S., Shukla, A., Tiwari, R., Dhar, J., 2018. Multi-robot multi-target dynamic path planning using artificial bee colony and evolutionary programming in unknown environment, **Intelligent Service Robotics, 11**: 171-186

[5] Liang, J., Lee, C., 2015: Efficient collision-free path-planning of multiple mobile robots system using efficient artificial bee colony algorithm, **Advances in Engineering software, 79**: 47-56

[6] Ma, Q., Lei, X., 2010, Dynamic Path Planning of Mobile Robots Based on ABC Algorithm, **International Conference (AICI), 6320**: 267–274.

[7] Cruz, M., Ramirez, V., Belmonte, U., 2015. Mobile robot path planning using artificial bee colony and evolutionary programming, **Applied Soft Computing, Volume, 30**: 319-328

[8] Karaboga, D., Gorkemli, B., 2014. A quick artificial bee colony (qABC) algorithm and its performance on optimization problems, **Applied Soft Computing, 23:** 227–238

[9] Karaboga, D., Gorkemli, B., 2012. A quick artificial bee colony -qABC- algorithm for optimization problems, pp. 2-4. *2012 International Symposium on Innovations in Intelligent Systems and Applications (INISTA)*

[10] Alsouly, H., Bennaceur, H., 2016. Enhanced Genetic Algorithm for Mobile Robot Path Planning in Static and Dynamic Environment, **8th International Joint Conference on Computational Intelligence, 1**: 121-131

[11] Ganeshmurthy, M., Suresh, G., 2015. Path Planning Algorithm for Autonomous

Mobile Robot in Dynamic Environment, pp. 376-381. *3rd International Conference on Signal Processing, Communication and Networking (ICSCN)*

[12] Uriol, R., Moran, A., 2017. Mobile Robot Path Planning in Complex Environments Using Ant Colony Optimization Algorithm, **3rd International Conference on Control, Automation and Robotics, 6088** (7): 15-20

[13] Wang, Z., Li, M., 2015. A Novel Multi-Objective Artificial Bee Colony Algorithm for Multi-robot Path Planning, pp. 481-486. *IEEE International Conference on Information and Automation*

[14] Sedighi, K., Ashenayi, K., Manikas, T., Wainwright, R., Tai, H., 2004. Autonomous Local Path-Planning for a Mobile Robot Using a Genetic Algorithm, **Congress on Evolutionary Computation (IEEE), 2**: 1338-1345

[15] Li, G., Chou, W., 2017. Path planning for mobile robot using self-adaptive learning particle swarm optimization", **Science China Information Sciences, 61**(5): 1

[16] Panda, M., Priyadarshini, R., Pradhan, S., 2016. Autonomous mobile robot path planning using hybridization of particle swarm optimization and Tabu search, pp. 1-7. *IEEE International Conference on Computational Intelligence and Computing Research (ICCIC)*

[17] Mohanty, P., Parhi, D., 2016. Optimal path planning for a mobile robot using cuckoo search algorithm, **Journal of Experimental & Theoretical Artificial Intelligence, 28**(1):1-18

[18] Lijun, Y., Zhihong, W., Hui, W., Ying, D., Zhengan, W., 2017. Path planning for mobile robot based on fast convergence ant colony algorithm, pp. 1493-1497. *IEEE International Conference on Mechatronics and Automation (ICMA)*

[19] Pandey, A., Pandey, S., Parhi, DR., 2017. Mobile Robot Navigation and Obstacle Avoidance Techniques: A Review, **International Robotics & Automation Journal, 2**(3): 96-105

[20] Saicharan, B., Tiwari, R., Roberts, N., 2016. Multi Objective Optimization based Path Planning in Robotics using Nature Inspired Algorithms: A survey, pp. 1-6. *1st IEEE International Conference on Power Electronics, Intelligent Control and Energy Systems*

[21] Garcia, M., Montiel, O., Castillo, O., Sepúlveda, R., Melin, P., 2009. Path planning for autonomous mobile robot navigation with ant colony optimization and fuzzy cost function evaluation, **Applied Soft Computing, 9**(3): 1102-1110

[22] Lei, L., Shiru, Q., 2012. Path planning for Unmanned Air Vehicles using an improved artificial bee colony algorithm, pp. 2486 – 2491. *Proceedings of the 31st Chinese Control Conference*

[23] Svaco, M., Jerbic, B., Polančec, M., Šuligoj, F., 2018. A Reinforcement Learning Based Algorithm for Robot Action Planning, **Advances in Service and Industrial Robotics, 6**: 493-503

[24] Gigras, Y., Jora, N., Dhull, A., 2016. Comparison between Different Meta-Heuristic Algorithms for Path Planning in Robotics, **International Journal of Computer Applications, 142**(3):6-10

[25] Janis, A., Bade, A., 2016. Path Planning Algorithm in Complex Environment: A Survey, **Transactions on Science and Technology, 3** (1): 31-40.

[26] Zhang, J., Zhang, Y., Zhou, Y., 2018. Path Planning of Mobile Robot Based on Hybrid Multi-Objective Bare Bones Particle Swarm Optimization With Differential Evolution**, IEEE Access, 6**: 44542-44555

[27] Panda, R., Choudhury, B., 2015. An Effective Path Planning of Mobile Robot Using Genetic Algorithm, pp. 287-291. *IEEE International Conference on Computational Intelligence and Communication Technology*

[28] Wang, X., Zhang, G., Zhao, J., Rong, H., Ipate, F., Lefticaru, R., 2015. A Modified Membrane-Inspired Algorithm Based on Particle Swarm Optimization for Mobile Robot Path Planning, **International Journal of Computers, Communications & Control (IJCCC), 10**(6): 732-745

[29] Hussein, A., Mostafa, H., Badrel-din, M., Sultan, O., Khamis, A., 2012. Metaheuristic Optimization Approach to Mobile Robot Path Planning, pp. 1-6. *International Conference on Engineering and Technology (ICET)*

[30] Chen, Y., Lu, Q., Yin, K., Zhang, B., Zhong, C., 2017. PSO-Based Receding Horizon Control of Mobile Robots for Local Path Planning, pp. 5587-5592. *43rd Annual*

Conference of the IEEE Industrial Electronics Society

[31] Malone, B., 2011, George Devol: A Life Devoted to Invention, and Robots, pp. 1-3. *IEEE Spectrum*

ÖZGEÇMİŞ

KİŞİSEL BİLGİLER

Adı Soyadı:	Ousainou Nyassi
Uyruğu:	Gambiya
Doğum Tarihi ve Yeri:	01.02.1992 – Banjul
Medeni Durum:	Bekar
e-mail:	nyassious@gmail.com
Yazışma Adresi:	Seyirtepe 50, Mevlana Mah,Talas/Kayseri/Turkey

EĞİTİM

Derece	Kurum	Mezuniyet Tarihi
Lisans	University of The Gambia, Bilgisayar M.	2014
Lise	Gambia Methodist Academy, Bakau	2010

İŞ DENEYİMLERİ

Yıl	Kurum	Görev
2014-2015	Guaranty Trust Bank Limited Gambia, Banjul (Gambia)	2015

YABANCI DİL

İngilizce, Türkçe