



Robot Sistemlerinde Kullanılan Algoritmalar

Gürkan GÜRGÜZE*¹, İbrahim TÜRKOĞLU²

¹Fırat Üniversitesi, Yazılım Mühendisliği Anabilim Dalı, Elazığ, Türkiye

²Fırat Üniversitesi Yazılım Mühendisliği Bölümü Elazığ, Türkiye

Gürkan GÜRGÜZE ORCID No: 0000-0002-2831-498X

İbrahim TÜRKOĞLU ORCID No: 0000-0003-4938-4167

*Sorumlu yazar: gurkangurgoze@gmail.com

(Alınış: 11.12.2018, Kabul: 02.05.2019, Online Yayınlanma: 30.06.2019)

Anahtar Kelimeler

Robot algoritmaları, Robot algoritma başarımları, Parçacık sürüsü, Olasılıksal, İstatistiksel, Analitik

Özet: Robot sistemlerindeki donanımsal ilerlemeler birçok alanda farklı kullanım fikirlerinin ortaya çıkmasına neden olmuştur. Bu düşüncelerin başarılı bir şekilde hayata geçebilmesi donanımsal gelişmelerin paralelinde hızlı, pratik ve fazla güç tüketimi sarf ettirmeyen yazılımsal çözümlerini zorunlu hale getirmiştir. Bu anlamda etkin yazılımların oluşması ve optimum çözümlerin sunulması için robot algoritmaları geliştirilmiştir. Günümüzde olasılıksal, istatistiksel, sezgisel, analitik ve farklı türlerde yapay zekâya dayalı birçok algoritma ve yöntem mevcuttur. Her algoritmanın problemlere yaklaşımları ve çözümleri farklı olduğundan özelliklerinin iyi bilinmesi doğru yerde ve şekilde kullanılmasını sağlayacaktır. Bu amaçla, robot çalışmalarında hangi algoritmaların kullanıldığı, kullanım amaçları, alanları, etkinlikleri, eksiklikleri, başarımları ve hangi alanda daha etkin oldukları incelenen literatür ışığında karşılaştırmalı olarak irdelenmiştir.

Algorithms Used In Robot Systems

Keywords

Robot algorithms, Robot algorithm performances, Particle swarm, Probabilistic, Statistical, Analytical

Abstract: Hardware developments in robot systems reveal different usage ideas in many areas. The success of these ideas has made it necessary for software solutions that fast, practical and do not require high power consumption in parallel with hardware developments. In this sense, robot algorithms have been developed to produce effective software and to provide optimum solutions. Today, there are many algorithms and methods based on probabilistic, statistical, intuitive, analytical and artificial intelligence in different genres. Each algorithm has different approaches and solutions to the problems. For this reason, a good knowledge of the properties of the algorithms will ensure that the algorithms are used correctly. For this purpose, this article describes the algorithms used in robot studies, their usage purposes, their fields, their activities, their deficiencies, their performances and in which fields they are more effective.

1. GİRİŞ

Robot dünyasındaki gelişmeler her geçen gün hızla devam etmektedir. Bununla birlikte ebatları küçülen, maliyetleri azalan, temin edilebilirliği artan, verimliliği ve etkinliği yüksek donanımsal yapıların ortaya çıkması robot çalışma sahasını genişletmiştir. Tabii yaygınlaşan robot uygulamaları çözülmesi gereken sorunların artmasına da neden olmuştur. Özellikle donanımsal gelişmelere paralel olarak yazılımsal çözümlerin geliştirilmesi önemli bir konu haline gelmiştir. Bu nedenle birçok algoritma ve yöntem geliştirilmiş ve kullanılmıştır [1-3].

Önceleri robotların endüstriyel ortamlarda kendini tekrarlayan işlemleri yapmaları, buna yönelik yazılımlar ve denetleyicilerin kullanımını yeterli kılmıştır[1]. Ancak daha sonraları robotların mobil özellik kazanması, sensörlerin insan duyusu gibi hassasiyetle çalışması ve bilgi iletişim özelliklerinin gelişmesi, robotların çalışması için tasarlanan ortam ve insanla etkileşimli olmasına olanak tanımıştır [4,5]. Bu durum sistemlerin karmaşıklaşmasına neden olmuştur. Karmaşıklaşan sistemlerin bütüncül bir yapı içerisinde ortam algılama, görev tanımlama, hareket etme, enerji sağlama vb. işleri aynı anda yapması artık beklenen bir konu haline almıştır. Bu amaçla birçok algoritma geliştirilmiştir [2,6].

Son yıllarda robotlarla ilgili çözüm aranan problemlere baktığımızda bilinmeyen bir ortamı algılayıp hareket etmesi, engellere çarpmaması, belirtilen hedefe en uygun yolla gitmesi, ortam bilgisini alabilmesi, insan etkileşimli olması, insan hislerini algılayabilmesi, çoklu ve sürü robotların birlikte hareketi gibi birçok konu karşımıza çıkmaktadır [7–11]. Bu konuların çözümünde istatistiksel, olasılıksal ve yapay zekâya dayalı algoritmalar önemli bir yer tutmaktadır. Özellikle getirdiği çözümlerin başarılı olması bilim adamlarını doğadaki canlıların davranışlarını incelemeye itmiştir ve sürü zekâsı algoritmaları etkin çözümler sunmuştur [12, 13].

Mesela, robot yol planlamalarında özellikle karınca, arı, ateşböceği, kurt, balık gibi canlıların yiyecek arama ve sürüyle hareket etme özelliklerine yönelik sürü algoritmaları mevcuttur. Robotların bulunduğu konumu bilmesi, ortamı algılaması ve ona göre bir sonraki hareketini ve davranışını gerçekleştirmesi için olasılıksal ve istatistiksel olan kalman filrelesi, markov zinciri, monte carlo yaklaşımı, yapay potansiyel alanlar yöntemi, levy uçuşu, hızlı rastgele ağaç keşfi gibi algoritmalar kullanılarak tahmine dayalı işlemler yapılmaya çalışılmıştır. Karar mekanizmalarını kuvvetlendirmek ve gelen bilgilere göre insan gibi düşünerek daha gerçekçi kararlar verilebilmesi amacıyla YSA, bulanık mantık, genetik algoritma, A* gibi birçok sezgisel algoritmalar kullanılmıştır. Görev planlama, özerklik yaklaşım, çarpışma önleme, sürülerin koordineli çalışmasını sağlayan algoritmalar gibi farklı sorunlara yönelik birçok algoritma bulunmaktadır [14,15].

Bu makalede, incelenen makaleler ışığında son yıllarda robot sistemlerde kullanılan algoritmalar incelenmiş ve bilgiler verilmiştir. Aynı zamanda algoritmaların yalnız ve hibrit kullanımları, uygulama alanları ve başarımları hakkında karşılaştırma yapılmıştır.

2. ROBOTLARDA KULLANILAN ALGORİTMALAR

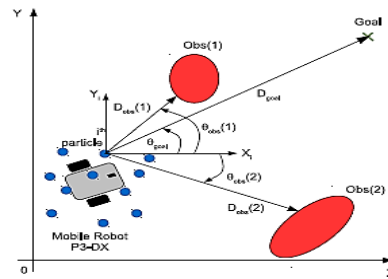
Robotlarla ilgili makaleler incelendiğinde genel olarak parçacık sürü algoritması, kalman filtre çeşitleri, markov zinciri, bulanık mantık, genetik algoritma, karınca kolonisi, monte carlo yöntemi, levy uçuşu, potansiyel alan yöntemi, hızlı rastgele ağaç keşfi, A* gibi algoritmaların ayrı ayrı veya birlikte çözümler getirdiği gözlenmektedir.

2.1. Parçacık Sürü Optimizasyonu

James Kennedy adlı bir sosyolog ile elektrik mühendisi olan Russel Eberhart tarafından 1995 yılında kuş ve balıkların popülasyon halinde birlikte sergiledikleri sosyal davranışları örnek alarak doğrusal olmayan durumlara çözümler getirmek amacıyla geliştirilmiş sezgisel bir algoritmadır. Görüntü işlemeden, robot navigasyonuna, bulanık mantıktan, yapay sinir ağlarına kadar birçok alanda optimuma yakın çözümler üretmede yaygın bir şekilde kullanılmaktadır [16,17].

Parçacık sürüsü optimizasyonunda özellikle bireylerin birbirleriyle haberleşmesi, bilgi paylaşımı ve durumlarını birlikte ortak harekete göre güncellemesi en etkili özelliğidir. Genetik algoritma gibi evrimsel özelliği olmasına rağmen güncel çoklu bilgi alış verişi işlem yükünü azaltmıştır. Son yıllarda mobil robotlarda bilinmeyen ve dinamik ortamların haritalanmasında, engellerden kaçmada, istenilen hedefe en hızlı, en kısa ve en uygun pozisyonda gitmede önemli bir algoritma haline gelmiştir. Özellikle, çok fazla karmaşık ortam bilgisine ihtiyaç duymadan, parçacıkların sezgisel durumlarına ve verdikleri bilgilere göre hareket etmesi uygulanmasını kolaylaştırmıştır [7,18,19].

PSO da parçacıkların rastgele çözümleriyle işleme başlanır. İstenilen hedef için öncelikle parçacıkların probleme göre uygunluk fonksiyonu elde edilir, daha sonra tüm parçacıkların hedef için en iyi değerleri (pbest) ve bu en iyi değerlere göre sürüdeki tüm parçacıkların hedef için en iyi çözüm değerleri (gbest) elde edilir. Bu elde edilen bilgilere göre parçacıkların hız ve konumları yeniden güncellenerek bir sonraki adıma geçilir. Bu durum istenilen hedefe ulaşana kadar devam eder [20–22].



Şekil 1. Parçacık Sürü Optimizasyonu (PSO) [16]

Bu çalışmada, deney materyali olarak yaklaşık olarak eşit büyüklükteki *A. cepa* L. tohumları (ortalama çap: 2.2 cm) kullanılmıştır. Tohumlar bir (1) kontrol ve üç (3) uygulama olarak toplam dört (4) gruba ayrılmış, 85x100 çapında steril beherlerde 24 °C'de 72 saat süresince çimlendirilmiştir. Grup I (kontrol) çeşme suyu, Grup II 50 ppm Tartrazin, Grup III 100 ppm Tartrazin ve Grup IV ise 200 ppm Tartrazin ile muamele edilmiştir. Çimlenme sırasında tohumların kurumaması amacıyla, su ve Tartrazin seviyeleri düzenli olarak kontrol edilerek, gerekli ilaveler yapılmıştır. Uygulama periyodu sonunda, kök uçları dH₂O ile yıkanmış ve standart preparasyon teknikleri kullanılarak sitogenetik analizler için hazır hale getirilmiştir [13].

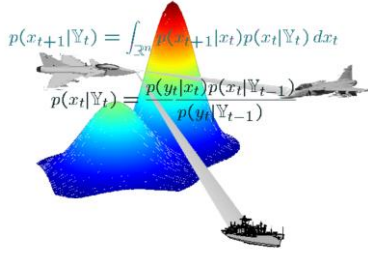
2.2. Kalman Filtre

1960 yılında Rudolf Kalman tarafından ortaya çıkarılan gürültülü doğrusal sistemlerin davranışını tahmin eden gerçek zamanlı dijital bir filtredir. Filtre aslında istatistiksel bir modelledir. Önceki bilgilere göre yeni bilgileri istatistiksel bir yaklaşımla tahmin etme üzerine çalışır. Rekürsif bir yapıdadır. En küçük kareler yöntemine göre geliştirilmiştir. Navigasyon sistemlerinde önemli bir yer bulmuştur. Görüntü

işlemeden, sinyal işlemeye, araç takibine kadar birçok alanda kullanılmaktadır [23].

Kalman filtresi doğrusal sistemlere uygulandığından doğrusal olmayan gerçek dünya problemlerini çözümünde sistemin doğrusallaştırılması gerekir ki bu yeni modele genişletilmiş kalman filtresi denir [24, 25]. Kalman filtresinin matematiksel gösterimi:

$$\hat{X}_k = K_k \cdot Z_k + (1 - K_k) \cdot \hat{X} \quad (1)$$



Şekil 2. Kalman filtresi takip gösterimi [26]

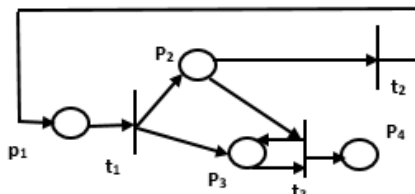
2.3. Petri Ağları

Petri ağları, birçok alanda her geçen gün karmaşıklaşan sistemlerin incelenmesi, anlaşılır olması ve kontrolünün yapılabilmesi amacıyla sistemin şekilsel yapılarla matematiksel olarak modellemesini sağlayan bir araçtır. Araç, belirli işlem tanımlarını daireler, geçişleri çubuklar ve bağlantıları da yönlendirilmiş oklarla yapmamızı sağlayan kolay bir yapıya sahiptir. Bu yapıda nokta (.) işareti o anda gelinen yerde işlem yapıldığını gösterir [27].

Modellemenin matematiksel gösterimi $N = \{P, T, F, Post, M_0\}$ dir.

Kısaca, petri ağı işaretleme vektörleriyle sistemin o anki durumunu gösterir. Bu kolay yapısı ve anlaşılabilirliği birçok mühendislik uygulamalarında kullanılmasını sağlamıştır [28].

Robot sistemlerde otonom çoklu robotların organize çalışmalarını takip etmek ve yönetmek için kullanılır. Çalışmalarda oluşan sorunlardan dolayı ekipteki elemanların çalışmasını durdurmak sistemin durmasına neden olabilir. Bu durumda işlemin yürütülmesi için tekrar başlangıç planına geçilmesi gerekir. Bu başlangıç durumuna geçmek işlem yükünü artırır, işlerliğini azaltır ve sistemi karmaşıklaştırır. Petri ağları bu konuda ekipteki elemanların kesme durumlarını sistemi etkilemeden yaparak sistemin yürütme işlerliğinin devamlılığını sağlar [29,30].



Şekil 3. Örnek petri ağı gösterimi [30]

2.4. Bulanık Mantık

Bulanık Mantık 1965 yılında Azerbaycanlı bir bilim adamı olan Kaliforniya Üniversitesi'nden Lutfi Zadeh tarafından geliştirilmiştir. Bulanık küme teorisine dayalı olan bu yaklaşım, insan düşüncesine özdeş işlemlerin gerçekleşmesini sağlamak amacıyla her duruma karşılık olan bir üyelik derecesi atayarak çalışır ve bu atama keskin ayrımların yerine daha yumuşak ve gerçekçi geçişlerin modellemesini sağlar [31].

Bulanık mantık teorisi üyelere [0,1] aralığında değerler verir. Bu değerler üyelik oranını temsil eder ve bu orana göre bir küme oluşturur. Böylece dereceli bir modelleme sağlanmış olacaktır. Bu da birçok sistemde alınan verilerin gerçek dünya algısına uygun ölçülerde tanımlanmasını sağlamıştır. Günümüzde nükleer reaktörlerden, robotlara, çamaşır makinelerinden buhar tribünlerine kadar birçok alanda yaygın bir şekilde kullanım alanı bulmuştur [31, 32].

2.5. Markov Zinciri ve Gizli Markov Zinciri Modelleri

Markov Analizi herhangi bir yapının şu anki verilerine göre bir sonraki durumdaki davranışlarını tahmin etmede kullanılan bir olasılık modelidir [33]. Gelecekteki durumlarının önceden ne olduğu anlaşılabilen skostatik süreçlerin analizinde kullanılır. Gerçek dünyadaki dinamik birçok durumun bu yapıya sahip olması kullanım alanını genişletmiştir. Matematikten fen bilimlerine, elektronikten robotlara, ekonomiden sosyal bilimlere, sağlıktan pazarlamaya kadar birçok alanda kullanılmıştır [34].

Markov analizinde durumlar arasındaki geçişler, zamanla gerçekleşen rastgele değişimlerine göre analiz edilir. Markov özelliklerine baktığımız zaman en önemli özelliği gelecekteki durumun şu anki duruma bağlı olup daha önceki durumlara bağlı olmamasıdır. Mesela, bir öğrencinin bir sonraki gün için durum tahmininin daha önceki durumlara göre değil en son bu günkü durumuna göre yapılmasıdır.

Bir skostatik süreç markov özelliği taşıyorsa bu sürece markov süreci denilir. Markov süreci eşit, kesik zamanlı ve geçiş olasılığı zamanla değişmeyen bir yapıda ise markov zinciri olarak ifade edilir. Tüm analiz bu markov zinciri ile gerçekleştirilir. Markov zincirindeki geçiş olasılığı, bir durumdan bir sonraki duruma geçme olasılığı olarak tanımlanır.

Gizli Markov modelinde (HMM) bilindiği gibi sistemin t anındaki tüm durumları bilinir ve tahminlerde buna göre yapılır. Ancak bazı sistemlerin belirtilen zamanlardaki durumları bilinmez bu durumda markov modeli istenmeyen olasılıksal sonuçlar üretir. Bu durumu çözmek için 1960-1970 li yıllar arasında gizli markov modeli geliştirilmiştir. Bu sistemde durumlar bilinmediğinden gizli özelliği verilmiştir. Geliştirilen bu yeni modelle sistemin bulunduğu durum bilinmemesine rağmen tetiklediği bir durumun gözlemi yapılarak gözlem matrisi oluşturulur ve var olan diğer durumlara

ek olarak bu gözlem matrisiyle çözümlenmeler yapılır [35].

Ancak çözümlenmelerde birçok gözlem olasılığı ortaya çıkar. Bu durumu çözmek için "Gizli Markov Modelinin Üç Temel Problemi" olarak adlandırılan işlemler yapılarak istenen sonuçlar elde edilebilir [36,37].

Bu üç temel problem:

- 1- Gözlemlenenlerin sistem tarafından gerçekleşme olasılığının hesaplanması ki buna ileri algoritma denir.
- 2- Bu gözlem dizilerinin, yeniden gerçekleşme olasılığı yüksek gözlem dizilerine indirgenmesi işlemi yapılır. Böylece gerçek duruma yakın gözlem değerlerinin çıkarılması sağlanır.
- 3- Elde edilen durumlar başlangıç durumlarına göre yeniden değerlendirilip, gözlemlerin bir daha oluşma olasılıkları hesaplanarak asıl istenene doğru bir nevi eğitim yapılarak en iyi sonucun elde edilmesi sağlanır.

Gizli Markov Modeli ilk olarak sinyal işlemede kullanılmıştır. El yazılarının tahmininden ses sinyal çözümlenmelerine, görüntü işlemeden karmaşık finansal işlemlerin tahminine kadar kullanılmıştır [38].

Mobil robotlarda otonom olarak bilinmeyen ortamlarda hareket etme ve istenilen işi yapmak zor bir durumdur. Bu tür ortamlarda en muhtemel yörüngeyi seçme, seçilen yörüngede hareket etme, engellerden kaçma, tahmini görev dağılımı, ortam durumuna göre hız kontrolü, ortam görüntü analizi gibi tahmini olarak yapılacak işlemlerin en mükemmel şekilde yapılması için yaygın bir şekilde kullanılmıştır.

Bunu yanında EEG sinyalleriyle olan kontrol durumlarında da sinyallerin yapılacak işe göre en uygun şekilde sınıflandırılmasında etkin olarak kullanılmıştır.

2.6. Monte Carlo Yöntemi

Monte Carlo yöntemi (MCL), kalman ve markov modelleri gibi olasılıksal çözümler sunan bir modeldir. Rastgele değerlerle, belirsizliğin yüksek olduğu deterministik durumların olasılıksal çözümünde kullanılmaktadır [39]. Tahmini olarak sistemlerin çözümünü gerçek durumuna yakın bir şekilde çözebilme özelliği matematik, mühendislik, biyoloji, ekonomi, istatistik gibi birçok alanda yaygın olarak kullanılmasını sağlamıştır. Monte Carlo yöntemi ününü geliştirilmesinden yıllar sonra 2. Dünya Savaşında atom bombası çalışmalarında kullanılmasıyla kazanmıştır. Modelin çözüm temelinde kullanılan rastgele sayılar elle veya oluşturulan tablolarla olmaktadır.

Bilgisayarın gelişmesi ve bu rastgele sayı üretiminin kolaylaşması, gerçek dünya olaylarının modellenmesinde tahmini durum ile gerçek durum benzeşmelerini sağlayabilmesi çeşitli benzetim süreçlerinde kullanılmasını sağlamıştır [40].

Monte Carlo Yönteminde rastgele sayılardan üretilen bir küme oluşturulur. $X = \{X_1, X_1, X_1, \dots, X_n\}$. Oluşturulan bu küme ne kadar büyük olursa çözüme o kadar yaklaşılr.

Ancak çok fazla seçmekte işlem yükünü arttır. Daha sonra bu elde edilen rastgele sayılarla ilgili örneklem alınır. Alınan örneklem analizi birkaç defa yapılarak istenilen sonuca yaklaşım sağlanır. En sonunda değerler birbirine yaklaşık rastgele sayılar üretecek duruma gelir ki buda sistemin gerçek bir şekilde simüle edilmesi demektir. Bu yöntem çalışıkça optimum sonuçlar üretebilen bir sistemdir [41].

Monte Carlo Yöntemi aynı zamanda robot sistemlerde konum tahmini ve yönelim problemlerinin çözümünde kullanılmaktadır. Robotlarda hareket planlamasının yapılabilmesi için öncelikle bulunduğu ortamın global ve yerel olarak neresinde olduğunu bilmesi gerekir. Global konum tahmini çeşitli sensörler ve haritalama yöntemleriyle belirtilebilirken özellikle yerel konumlarda ortam bilgisinin eksikliği konum belirlemede belirsizliklerin oluşmasına neden olmaktadır. Monte Carlo Yöntemi bu durumu çözmeye kullanılan yöntemlerden biridir. Markov ve Kalman filtre yöntemlerinin özellikle global konumlandırmada etkinliğinin yanında Monte Carlo Yöntemi hem global hem de yerel konum belirlemede etkin çözümler sunmaktadır. Çünkü olasılık hesabını rastgele belirlenmiş değerlerden oluşan parçacık ile gerçekleştirmesi işlemi hızlandırmasına ve bu da gerçek zamanlı çözüm demektir [42]. Ancak işlemi gerçekleştirmede fazla örneklem ihtiyacı olan durumlarda bu yöntem fazla bellek ve süre almasına neden olabilir. Özellikle robotlarda görüntü tabanlı haritalama ve yönelim planlamalarında aldığı bilgilere göre kapalı mekânlarda bulunduğu konumu bilme noktasında etkin sonuçlar elde etmiştir. Müze, kütüphane, hastane gibi ortamlarda kullanılan hizmet robotlarının geliştirilme aşamalarında bu yöntem etkin bir şekilde kullanılmıştır [43].

2.7 Levy Uçuşu

Levy uçuşu doğadaki hayvanların bilmedikleri ortamlarda yiyecek, su gibi ihtiyaçlarını arama stratejilerinden ortaya çıkmış olasılığa dayalı hedef arama algoritmasıdır. 1930'lu yıllarda Fransız bilim adamı P.Levy tarafından Brownian yürüyüşü yönteminden esinlenerek kuşların, bal arılarının, karıncaların ve denizdeki bazı balıkların rastgele yürüyüş ve ihtiyaç arama durumlarından ortaya çıkarılmıştır. Rastgele yürüyüş, olasılıksal dağılıma göre yapılmış olması çok fazla ortam bilgisine ihtiyaç duyulmamasını ve böylece işlemlerin hızlı bir şekilde yapılmasını sağlamaktadır [44].

Var olan diğer hedef bulma algoritmalarına göre yöntem geçmiş olduğu yerlere dönme ihtimalinin en aza indirilmesi açısından ön plana çıkmaktadır. Bu özelliği ile geliştirilen sistemlerin zor ve tehlikeli ortamlarda gereksiz hareket yapmaması ve enerji harcamaması amaçlanmıştır [45].

Günümüzde özellikle otonom olarak çalışan mobil robotlarda en önemli sorunlardan biri bilinmeyen ortamlarda arama kurtarma, uçakların kara kutularını arama, bilgi toplama gibi görevlerde hedeflerin tespiti ve

buna göre hareket planlamasıdır. Başlangıçta hedeflenen görev için ortam bilgisini oluşturmak tek sistemli robotlara verilmişken daha sonraki çalışmalarda daha küçük ve işlevselliği yüksek çok sayıda robotun geniş alanlara dağılımlarının daha etkin tarama yapabildiği görülmüştür. Bu çoklu robotlar ile arama yapmada kullanılan birçok algoritma bulunmakla birlikte son yapılan çalışmalarda Levy uçuş algoritmasının etkin sonuçlar vermesi kullanımını arttırmıştır [46].

Levy olasılık dağılımı formülasyonu;

$$P_{\alpha\gamma}(l) = \frac{1}{\pi} \int_{-\infty}^{\infty} e^{-\gamma q^{\alpha}} \cos(ql) dq \quad (2)$$

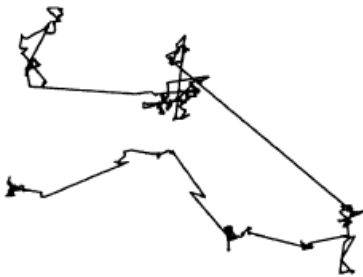
Burada dağılım $l = 0$ değerine göre simetrik olması sağlanır. γ ölçek, α ise dağılımın tipini belirler. Dağılımın şekli 0-2 arasında bir değer alır. α her yeni ortamda farklı olasılıksal çözümler için farklı dağılım tiplerini belirleyerek ortama uygun çözümler üretmemizi sağlar. α değeri 2 sınırına yaklaştığında Levy dağılımı yerini Gauss dağılımına verecek ve γ ölçek değeri 1 haline gelecektir. Bu durumda l değeri en büyük değere;

$$P_{\alpha}(l) \approx l^{-\alpha} \quad (3)$$

formülasyonu ile yaklaştırılacaktır. Daha sonra denklemdeki uçuş parametrelerinin optimizasyonu çeşitli tekniklerle gerçekleştirilir. Bu işlemlerde öncelikle başarılı Levy uçuşlarının sayısı aşağıdaki eşitlik 4 ile hesaplanır.

$$N \approx \left(\frac{\lambda}{r_v}\right)^{\frac{(\alpha-1)}{2}} \quad (4)$$

λ ard arda olan hedef noktaların arasındaki ortalama mesafe, r_v noktalar arasındaki algılama aralığı olarak verilmiştir. Ortaya çıkan dağılımlardaki bilgilere göre $\alpha = 1$ yapılarak arama stratejisi ortak bilgi halinde ayarlanır. Böylece hangi alanda nasıl arama yapıldığı bilgisi elde edilerek hedefe yakın veya hedefin olmadığı tespit edilen alanların dışında aramaların yapılması sağlanmış olur [47- 49].



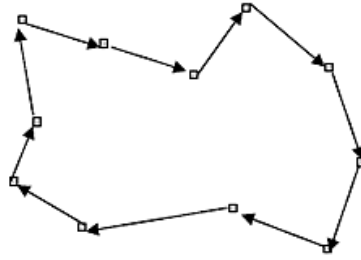
Şekil 4. Levy Uçuş Örneği [49]

2.8. Genetik Algoritma

Genetik algoritma Darwin'in evrim teorisini ele alarak geliştirilen evrimsel tekniklerin 1970 yılında bilgisayar

bilimcisi John Holland tarafından karmaşık, çözülmesi zor ve hesaplaması uzun süreler alacak problemlerin bilgisayar ortamında çözümlenmesi amacıyla ortaya atılan bir arama ve optimizasyon yöntemidir. Genetik Algoritma, her zaman eldeki verilere göre en iyiyi arama stratejisinde olduğundan dolayı imalat sanayisinde, coğrafi sistemlerde, matematikte, fen bilimlerinde ve son zamanlarda özellikle akıllı sistemler ile robotlarda yön bulma, en iyi rota seçimi ve hedef tahmini uygulamalarında etkin bir şekilde kullanılmaktadır.

Genetik algoritmada temel özellik çok geniş problem ağı ve çözüm kümeleri olabilen durumların en iyi ve olası rastsal seçimlere göre ortalama ve gerçek duruma yakın sonuçlar üretebilmesidir. Bu yakınsanmış çözümler; evrimdeki çevreye uyum, elde edilen bilgilere göre en iyi çözümlere doğru mutasyon veya değişim ilkesine dayanarak her defasında popülasyonun en iyiye ulaşması sağlanarak bulunur. Karmaşık sistemdeki büyük popülasyonların sadece rastgele bir bölümünün örneklem şeklinde alınarak kullanılması işlem hızını yükseltmesine rağmen bazen alınan örneklemelerin sistemi temsil edememesinden dolayı sonuç üretilmemesine neden olmaktadır. Ancak hiç çözüm üretilmeyen veya üretilmesi zor sistemlerde bir sonuç vermesi ile bu yönü birçok durumda göz ardı edilmektedir [50].



Şekil 5. Genetik algoritma ile en kısa yol problem çözümü [51]

2.9. Yapay Potansiyel Alanlar Yöntemi

Robot dünyasındaki hızlı gelişmeler mobil robotların bilinen veya bilinmeyen her türlü ortamda otonom olarak verilen görevleri yapabileceği uygulamaların yapılmasını sağlamıştır. İstenilen görevlerin gerçekleştirilmesinde hareket kabiliyeti önemlidir. Bu noktada istenilen hedeflere yönelimde ortamdaki hareketli veya hareketsiz engellere çarpmadan en uygun yolda ilerlemek ve hedefe ulaşmak öne çıkan problemlerden biridir [52].

Mobil robotlar hareketlerini ortam bilgisine göre gerçekleştirirler. Kendi konumu, hedef konumu, hareket yörüngesi, engeller, engellerin statik veya dinamik oluşu hareket mekanizmasını oluşturan önemli unsurlardır. Tüm bu bilgiler bilinerek bir hareket planlaması gerçekleştiriliyorsa genel, anlık verilere dayalı bilinmeyen veya eksik bilgilerle bir hareket planlaması oluşturuluyorsa yerel bir planlamadan bahsedilir.

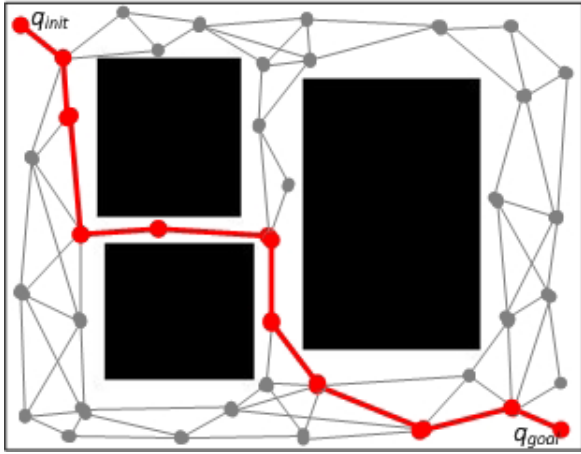
Bu alanda olasılıksal, sezgisel, geometrik, görüntü tabanlı birçok hareket ve yol planlama algoritmaları mevcuttur [53]. Dinamik çevrelerde ve anlık çözümlene yapabilen yaygın metotlardan biri yapay potansiyel alan

2.12. Olasılıksal Yol Haritalama

Olasılıksal yol haritalama metodu özellikle statik engellerin bulunduğu, robotların kısıtlı ve belirli hareketleri yaptığı uygulamalarda olasılıksal çözümlerle ortam bilgisini öğrendiği ve buna göre hareketini gerçekleştirdiği bir yol planlama yöntemidir.

Bu planlama yönteminde öğrenme ve sorgu olarak iki aşama mevcuttur. Öğrenmede, öncelikle ortam bilgisini oluşturmak için rastgele üretilen noktalar ve bu noktaların birleşiminden oluşan bir olasılıksal ortam haritalama gerçekleştirilir. Bu haritalamada seçilen noktaların engellere denk gelmeden oluşturulmasıyla engellerin tespiti de elde edilmiş olur. Oluşturulan harita graf yapısındadır. Noktalar arasında yapılan bağlantılar bu grafın kenarları olarak tanımlanır. İkinci aşama olan sorgu aşamasında ise, elde edilen bu olasılıksal haritaya göre istenilen kaynak ve hedef nokta arasında var olan yollardan bağlantısı yapılabilecek en kısa yollar belirlenir, eğer kaynak ve hedef bu yollar üzerinde değilse en kısa kenar bağlantıları yapılarak bağlantı yapılmış olur [64].

Bu metot ile yapılan uygulamalarda ilk ortam öğrenme aşamasından sonra belirtilen noktalar arasındaki hareketler yeniden ortam bilgisi için işlem yükü getirmeden çok hızlı çalıştığını göstermektedir [65].



Şekil 9. Olasılıksal yol haritalama örneği [64]

2.13. Ateş Böceği Algoritması

Ateş böceği algoritması 2008 yılında Xin-She Yang tarafından ateş böceklerinin yaydıkları ışığın özelliğine göre yaptığı iletişim üzerine geliştirilen sezgisel bir arama algoritmasıdır. Bu algoritma ateş böceklerinin yaydıkları ışığın şiddetine, yanıp sönmeye süresine ve ritmine göre oluşturdukları görselle iletişim kurma becerileri üzerine kurulmuş bir yapıya sahiptir. Bu özellikleri ile eş bulma, avlarının dikkati çekme veya avcılara karşı korunma mekanizması oluşturmayı gerçekleştirebilmektedirler [66].

Bu algoritmayla var olan ortamdaki ateş böceklerinin ışık şiddetlerine göre az olandan daha parlak olan bir yönelim ve hareket oluşturulmaktadır [67].

Ateş böceği algoritması 3 temel kural üzerine gerçekleştirilir.

- 1) Ateş böceklerinin cinsiyetsiz olma özelliği ışık şiddetlerinin düşük olandan yüksek olana bir yönelimin olmasını sağlayacaktır.
- 2) Parlaklık ile çekici olması doğru orantılıdır. Ancak mesafe arttıkça parlak azalmaktadır.
- 3) Parlaklığı istenilen duruma göre belirlenir.

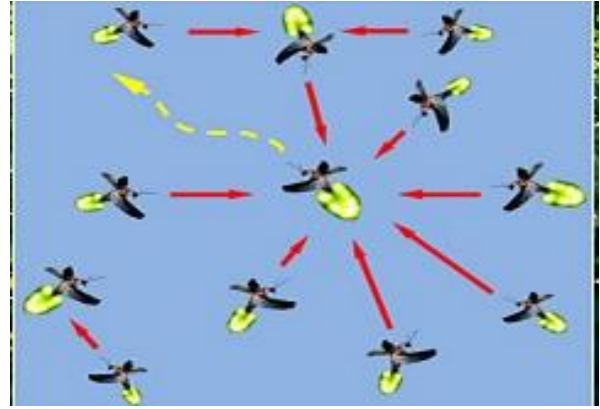
Formülüne baktığımızda elde edilmek istenen ışığın çekicilik şiddeti;

$$\beta = \beta_0 e^{-\gamma r^2} \quad (5)$$

Burada β_0 , r_0 daki çekicilik şiddetidir. r ışığın görünme mesafesidir. Alandaki iki ateş böceğinin birbirine göre çekme durumları ise;

$$x_i^{t+1} = x_i^t + \beta_0 e^{-\gamma r_{ij}^2} (x_j^t - x_i^t) + \alpha \epsilon_i^t \quad (6)$$

formülü ile ortaya konulur. Bu durumlara göre bir yönelim gerçekleştirilmiş olur [19, 68, 69].



Şekil 10. Ateş böceği algoritma örneği [70]

2.14 Gri Kurtların Algoritması

Gri Kurt algoritması doğadaki kurtların organize olarak avlarını bulma, hareket etme, toplanma, haberleşme gibi özelliklerini kullanarak yaptıkları davranışları taklit ederek ortaya konulan bir algoritmadır. 2007 yılında Xing Yang tarafından geliştirilmiştir. Geniş alanlarda yol planlaması sağlanması bu algoritmanın ön plana çıkan özelliğidir [19].

Kurt algoritmasında, kurtların bazıları keşif için kullanılır. Bu kurtlar buldukları avın konumunu uluma ile diğer kurtlara bildirerek arama uzayını [71] da işlem yükünü azaltıp etkin sonuca ulaşılmasını sağlar. Diğer kurtlar ise ava yaklaşır ve avı kuşatırlar. Tüm kurtların hedefe ulaşmasıyla av güçlü kurttan başlanarak zayıf kurtlara doğru paylaşılır. Bu şekilde yapılan paylaşım ile bir sonraki arama uzayındaki kurtların daha güçlü olanlarının kalması için zayıf kurtların açıklıktan ölme stratejisiyle sürüden çıkarılması sağlanır. Böylece daha güçlü arama özelliğine sahip sürü oluşturulmuş olur [72,73].

gerçekleştiği gözlemlendi. Bunu gerçekleştirmek için adaptif gizli Markov modeli (ADHSMM) kullanılmıştır.

Tang ve ark. [15] Çoklu robotların işbirliğine dayalı Monte Carlo yöntemi ile ortam yerleştirme gerçekleştiren bir yaklaşım sunmaktadır. Robotların lokalizasyonu için Genişletilmiş Kalman Filtresi kullanılmıştır. Farklı çoklu robot grupları için deneysel sonuçlara bakıldığında ilk grubun 100 referans noktası için ilk durumun yerleşimi 91,9 cm iken iyileştirme sonucunda 99,9 cm noktasına ulaştığı ve tüm gruplarla 20 kez yapılan çalışmadan sonra ortalama standart sapmanın 5,1cm 'den 0,6 cm'ya kadar gerilediği gözlemlendi.

Lee ve ark. [5] Çoklu robotlarının hareketleri sırasında çarpışmadan kaçınma ve kontrollü oluşumu üzerine geri çekilmiş ufuk parçacığı sürüsü optimizasyonuna (RHPSO) dayanan yeni bir model öngörme kontrolü (MPC) sunulmuştur. Deneysel çalışmada kontrol otoritesindeki bozulmanın %10'nu geçmesi halinde çarpışmaların olduğu ve ortalama hareket yaklaşımının 0.4m olduğu gözlemlendi.

Su ve ark. [7] Çalışmada mobil robotun engellere çarpmadan en kısa yolu bulmasını sağlayacak parçacık sürüsü optimizasyon algoritmasına dayalı yeni bir yol planlama algoritması önerilmiştir. Bu yeni algoritma rastgele kodlama parçacık sürü algoritmasına (RCPSO) dayanmaktadır. Sürü boyutu 60, yineleme sayısı 200, ivme katsayıları 0,2 olarak alınmıştır. Performans için eylemsizlik ağırlığı kullanılmıştır. Buna göre Dijkstra, Karınca kolonisi, temel PSO ile önerdiğimiz algoritma karşılaştırmasına göre 140 iterasyonlu çözümlerde RCPSO'nun daha hızlı istenilen konuma yaklaştığı en yakın ise karınca kolonisinin 40. İterasyonla gerçekleştirdiği görülmüştür.

Gong ve ark. [13] Makalede tehlikeli kaynaklara sahip ortamlarda robotun yolunu bulması ve en uygun yolu planlaması üzerine yeni adaptif mutasyonlu geliştirilmiş çok amaçlı parçacık sürü algoritması (WMPSO) sunulmaktadır. Bu yeni algoritmanın MPSO ile karşılaştırılması yapılmıştır. Deneysel çalışmada popülasyon 50, yineleme sayısı 30, ivme katsayıları 2 olarak alınmıştır. Belirlenen yolla göre MPOS ve WMPOS algoritmaları kıyaslandığında 3. Deneysel çalışmaya göre WMPOS 'un yakınsamasının daha iyi olduğu, ancak dağılımın ise kötü olduğu gözlemlendi.

Ahmadzadeh ve ark. [16] Mobil robotların ev, ofis vb. kapalı alanlarda hareket edebilme ve gezinmesi üzerine bir çalışma sunmaktadır. Bu çalışmada özellikle bilinmeyen ortamlarda çevreyi algılama ve başka bilgilere ihtiyaç duymadan hareket kabiliyetini sağlamasıyla ilgili PSO tekniğiyle yeni bir algoritma sunulmuştur.

Islam ve ark. [17] Çoklu hedeflerle dinamik bir ortamda bir mobil robotun navigasyonuna yönelik akıllı bir yaklaşım önerilmektedir. Hareketli engeller ve çoklu hedef ortamları için geliştirilmiştir PSO yöntemi kullanılmaktadır. 100 iterasyonlu, 0,4, 0,9 ivme

katsayıları kullanılmıştır. Popülasyon boyutu 50 olarak belirlenmiştir. Sonuçlara baktığımızda hedefe ulaşım süreleri maksimum 6,9 sn olarak engellere çarpmadan gerçekleştirilmiştir.

Mousavi ve ark. [80] Hızlı üretim hatlarında kullanılan otomatikleştirilmiş robotların sayısının ve çalışma kabiliyetlerinin artırılması üzerine genetik algoritma (GA) ve parçacık sürü optimizasyon (PSO) algoritmalarına dayalı hibrit bir model sunulmuştur. Bu modelin yalnız başına GA ve PSO'ya göre verimi incelenmiştir. Geliştirilen model ile güç tüketiminin çalışma zamanına göre azaldığı ve operasyonun büyüklüğüne göre sabit robot yerine gerekli sayıda robotun kullanımının başarılı olduğu gözlemlendi. Performans verimi olarak GA %69,4, PSO %74 ve GA-PSO hibritin %79,8 olduğu belirlendi.

Deepak ve ark. [18] Bir mobil robotun engellere ve bilinmeyen ortama göre en iyi rotayla belirtilen hedefe ulaşması için parçacık sürü optimizasyon tabanlı bir yaklaşım sunulmuştur. 80 popülasyonla ve 20 defa tekrarlanan deneysel bir çalışma sunulmuş. Sonuçta her tekrarda hedefe en kısa uygun yolla ulaştığı belirlenmiştir. Sürü arasındaki en iyi konum uygunluk değerine göre seçilmiştir. Bulanık mantıkla geliştirilen iki duruma göre karşılaştırıldığında daha yakın mesafede ve belirlenen en iyi uygunluk değerine göre PSO'nun daha başarılı sonuçlar ürettiği gözlemlendi.

Adriansyah ve ark. [81] Bir robotun hedefe doğru bir şekilde yönlendirilmesi amacıyla bulanık mantık ve parçacık sürü optimizasyonunun birlikte kullanımıyla geliştirilen yeni bir davranış tabanlı parçacık sürü bulanık denetleyiciden (PSFC) bahsedilmiştir. Özellikle bilinmeyen ortamlarda hedefe doğru yönelim için bulanık denetleyicinin yalnız kullanımında parametrelerin manuel girilmesi gerekirken, yeni modelde etrafın algılanması ile bulanık denetleyicinin parametrelerini PSO ile belirlenmesi sağlanarak öğrenme tabanlı bir yaklaşım sunulmuştur. Magellan Pro mobil robotu ile yapılan deneysel çalışmada genetik algoritma bulanık denetleyici ve önerilen parçacık sürü bulanık denetleyici kıyaslanarak performans değerlendirilmesi yapılmıştır. Sonuçta önerilen yaklaşımın daha erken nesillerde etkili arama davranışı sergilediği gözlemlendi.

Schoreder ve ark. [82] Bir robot sürüsünün etkili alan kaplaması ve etraftan gelebilecek tehditleri algılayıp davranış oluşturmaya üzerine Levy dağılımına dayalı bir kontrol modeli sunulmuştur. Davranışlar karıncaların topluluk halinde yaptıkları davranışlardan esinlenilmiştir. Karıncalardaki feromon kaynaklı hareket stratejisi kullanılmıştır. Aynı zamanda acil durumlara karşı davranışları dikkate alınmıştır. Alan kaplamada ve yönelimde gürültü yoğunluğu, feromonun difüzyon hızı ve buharlaşma hızı önemli parametreler olarak dikkate alınmıştır. Deneysel sonuçta modelin güven seviyesinin %95 olduğu tespit edilmiştir. Yapılan çalışmada gürültü yoğunluğu 0,05 iken, buharlaşma hızı 0,0001 ve difüzyon hızı 0,0001 olarak gerçekleştirilmiştir.

Abbas ve ark. [83] Lider ve takipçileri şeklinde hareket eden çoklu robotlarda iletişimin gürültü veya harici arızalar nedeniyle başarısızlığında dahi takipçilerin liderin konumunu tahmin edip harekete devam edebilmesi üzerine parçacık sürü algoritma tabanlı bir yaklaşım sunulmuştur. 1 lider 2 takipçiyle yapılan deneysel çalışmada hareketin 10 ve 20 saniyeleri arasında oluşan kesilmeye rağmen takipçilerin liderin konumunu tahmin edip izlemeye devam ettikleri gözlemlendi. PSO parametrelerine baktığımızda 100 popülasyonlu, 100 yinelemeli ve 0,8 ağırlıklıdır.

Farinelli ve ark. [27] Takım odaklı çalışan çoklu robotların görevler sırasında karmaşık ortamlarda bir tehlike gibi karşılaştıkları sorunlarda faaliyetlerini kesip, yeni bir planlamayla göreve devam etmesi üzerine petri ağı tabanlı bir program sunulmuştur. Deniz robotlarıyla çarpışma tehlikesi üzerine deneysel çalışma gerçekleştirilmiştir. Çalışmada belirlenen planların kesmeler sonucunda yeniden uyarlanabildiği gözlemlendi.

Pei ve ark. [84] Bu çalışmada özerk mobil robot navigasyon sistemi için geliştirilen dağıtılmış parçacık sürü optimizasyonlu dağıtılmış parçacık filtresi önerilmiştir. Performansını ölçmek için parçacık sürüsü optimize edilmiş dağıtılmış parçacık filtresi SLAM (PSO-DPF-SLAM) algoritması ve atalet parçacığı sürüsü optimize dağıtılmış parçacık filtresi SLAM (IPSO-DPF SLAM) algoritmasının tahmini sonuçları karşılaştırılmıştır. İki ayrı deney yapılmıştır. Sonuçta önerilen algoritmanın daha uzun periyotlarda daha iyi tahminde bulunduğu gözlemlendi. PSO popülasyonu 100, yineleme 10 olarak alınmıştır.

Katada ve ark. [44] Sürü robotlarının hedef tespitinde Levy uçuşuna göre rastgele yürüyüşün değişken adım boyutuna ve robot sayısına etkisi araştırılmıştır. Levy uçuş dağılımının sabit ve rastgele normal yürüyüş adım boyutuna göre daha başarılı olduğu deneylerde gözlemlenmiştir. Sürüde 20 robot kullanılmıştır. Ağ ile iletişim gerçekleştiren robotların A,B,C ve D bölümlerinden oluşan alan hedef deney düzeneğinde A'da bulunan robotların yürüyüş stratejilerine göre D'ye ulaşma durumları rastgele yürüyüş stratejisiyle yüksek orana ulaştığı gözlemlendi. Adım büyüklüğüne göre ise rastgele yürüyüş RN(2) adım ile sadece %10'unu C ve D bölgesine ulaşırken, RN (6) adım boyutunda %35'i ve Levy rastgele yürüyüş LF(6) adımında ise %60'ının C ve D bölgesine ulaştığı ortaya çıkmış. Böylece levy dağılımı rastgele yürüyüşün normal rastgele yürüyüşe oranla daha başarılı olduğu gözlemlenmiştir.

Patle ve ark. [14] Belirsiz ortamlarda bir mobil robotun normal olasılık dağılımı üzerinde ateşböceği algoritması ile statik engellere rağmen hedefe en uygun rotada ulaşması üzerine gerçek zamanlı seyrüsefer yaklaşımı sunulmuştur. Deneysel çalışmada Khepera robotu kullanılmıştır. Belirlenen üç senaryo ile simülasyon ve gerçek zaman arasındaki alınan mesafe ve zaman farkları gözlemlenmiştir. Mesafe olarak, simülasyon ile gerçek zaman arasında 1. senaryo için %4,86, 2. Senaryo için %3,88, 3. Senaryo için %5,78 hata olduğu gözlemlendi. Zaman olarak, 1. Senaryo için %4,52, 2. Senaryo için

%5,84 ve 3. Senaryo için %4,32 hata oluşmuştur. Sonuçta önerilen algoritmanın yol uzunluğunu koruduğu ve zaman sınır içerisinde hedefe ulaştığı belirlendi.

Guruji ve ark. [58] Bir mobil robotun bulunduğu ortamda dinamik engellere çarpmadan kaynak ve hedef arasında en uygun yolu işlem süresini en aza indirerek bulması için geliştirilmiş Astar algoritması sunulmuşlardı. Önerilen algoritmanın standart A star algoritmasından farkı her düğüm için sezgisel hesaplamayı yapmayıp sadece var olan engellerle çarpışma olmadan yaparak gerçekleştirmektedir. Yapılan deneysel çalışmada bu durum sayesinde %95'e kadar bir zaman tasarrufu sağladığı gözlemlenmiştir. Algoritmanın en fazla %3,4 oranında yolu uzattığı, işleme süresini ise en az %65 düşürdüğü elde edilmiştir.

Duchon ve ark. [57] Bu makalede ızgara şeklinde belirlenen bir harita üzerinden geliştirilmiş A star algoritmasına dayalı uygun yol bulma metodu önerilmiştir. Yeni geliştirilmiş Astar algoritma standart Astar ile Astar algoritmasından türeyen teta star ve Phi star algoritmalarıyla karşılaştırılarak performansı ortaya konmuştur. Yeni yaklaşımda amaçlanan kontrol edilen ızgara hücrelerinin sayısını azaltarak işlemleri gerçekleştirmektedir. İşleme alınan hücrenin komşuluk hücrelerinin engele yakın olmaması halinde kırılması üzerinedir. Bu sayede incelenecek hücre sayısı azaltılmış olur. Deneysel sonuçlara göre yolun hızlı bulunmasında önerilen sistem çok başarılı iken, daha uzun yol bulması dezavantaj olarak ortaya çıkmaktadır. Yol uzunluğu açısından ise temel teta algoritmasının daha başarılı olduğu gözlemlendi.

Sudhakara ve ark. [85] Bilinmeyen ve statik engelli bir ortamda mobil robotun kaynaktan hedefe en uygun yolu çarpmadan gerçekleştirmesi üzerine geliştirilmiş A* algoritması sunulmuştur. Karşılaştırma 3 deneysel senaryo ile standart A* algoritmasıyla yapılmış. Izzara harita temelli bir çalışma yapılmıştır. Geliştirilen A*'da yapılan değişiklik, bulunulan noktada hareket için gidilebilecek olan yukarı-aşağı ve sağ-sol kadrantların her biri için hesaplama yapmadan sadece hedefe doğru olan kadrantın belirlenip hesaplanması üzerinedir. Deneysel çalışmada normal A* ile önerilen algoritmaların karşılaştırılmasına baktığımızda sırasıyla; Birinci uygulamada 1,922 iken önerilen metod da 0,505, ikinci uygulamada 3,334 da karşın 1,205, 4,671'e 1,368 şeklinde elde edilmiştir. Görüldüğü gibi önerilen metod daha kısa işlem süresinde hedefe yönelmiştir.

Güney ve ark. [25] Çalışmada, bir mobil robotun farklı hareket manevralarının takibi için standart, adaptif ve genişletilmiş kalman filtreleri ile doğrusal ve doğrusal olmayan modellerden oluşan çoklu model kalman filtre kestirim algoritması üzerine bir çalışma sunulmuştur. Deneysel çalışmada kalman filtresinin yalnız başına doğrusal hareketlerde en iyi sonucu verdiği görülmüş. Ancak doğrusal olmayan dönüşler gibi, noktalarda hatalı kestirimler yaptığı gözlemlenmiş. Hibrit olarak önerilen modelin ise dönüş manevralarını belirli noktalarda az hata olsa da hareket takibini başarılı bir şekilde yaptığı tespit edilmiş.

Song ve ark. [86] Yapay potansiyel alan algoritmasının dezavantajlarından yerel minimumların net kuvvetlerinin sıfıra eşit olmasıyla oluşan durma veya yanlış hareket için sanal alt hedef ve sanal daire fikri ile yeni yapay potansiyel alan algoritması önerilmiştir. Hizmet robotları için geliştirilmiştir. Bu robotların kullandıkları ortamların sabit olması önerilen yaklaşımı kolaylaştırmıştır. Statik engelleri kullanarak ortamda küresel bir ortam model bilgisi oluşturularak herhangi bir sıkışmada kurtulması amaçlanmıştır. Bu yöntemin hareketi yumuşak ve daha güvenli yaptığı gözlenmiştir.

Simining ve ark. [87] Mobil robot yol planlamasında kullanılan yapay potansiyel alan algoritmasının yerel uç noktada takılı kalması probleminin çözümüne yönelik yeni bir çözüm sunulmaktadır. Yöntemde hedeflenen noktaya yakın bir yerde ek olarak sanal hedef nokta oluşturularak çakılma durumu engellenerek hareketin devamı sağlanmıştır.

Yongshen ve ark. [88] İnsansız yer araçlarının hareket ve oluşum kontrolü için çarpışma, hız ve bir araya gelme özelliklerine dayalı yapay potansiyel alan yöntemi önerilmiştir. Özellikle çarpışmadan kaçınmayı önemli ölçüde başardığı gözlenmiştir.

Patle ve ark. [89] Mobil robot yol planlamasında daha kısa ve uygun yol için iyileştirilmiş ateş böceği algoritması önerilmiştir. Belirsiz ve değişen çevre koşullarına hızlı cevap veren bir yaklaşım amaçlanmıştır. Önerilen algoritma çevre koşullarını verimli araştırıp daha az yinleme ve hesaplamayla yol planlamayı oluşturmuş. Genetik, fuzzy gibi yöntemlerle karşılaştırıldığında %6 oranında daha başarımlı olduğu tespit edilmiştir.

Nazarahari ve ark. [90] Çoklu mobil robotların görevlerde en uygun yolları bulması için geliştirilmiş genetik algoritma yaklaşımı önerilmiştir. Bu yaklaşımda öncelikle birden fazla yolun bulunması için yapay potansiyel alan yöntemi kullanılmış. Daha sonra yolların iyileştirilmesi için yolun uzunluğu, düzgünlüğü, güvenliği ve çarpışma parametreleri göz önüne alınarak genetik algoritma ile iyileştirilmiş yeni yumuşak güvenli yol oluşumu sağlanmıştır. Çarpışmada mesafe kriter olarak kullanılmıştır. A star, PSO, PRM, B-RRT gibi

algoritmalarla karşılaştırılmış çalışma süresi ve yol planlama başarımının daha yüksek olduğu gözlenmiştir.

Dewang ve ark. [91] Mobil robotların yol planlaması için uyarlamalı PSO tabanlı bir yaklaşım önerilmiştir. Bu yaklaşımda robotun hedefe ve engellere olan mesafeleri ölçüt olarak alınarak yeni bir hesaplama fonksiyonu ile optimize edilmiş bir PSO yol planlaması geliştirilmiştir. Klasik yöntemlere göre engellerden kaçınma ve hedefe daha kısa sürede ulaşma başarımının yüksek olduğu simülasyonlarda gözlenmiştir.

Che ve ark. [92] Patlamaya dayanıklı robotların en uygun yolu bulmasına yönelik kurt kolonisi ile optimize edilmiş bir karınca kolonisi yaklaşımı geliştirilmiştir. Çekim kuvveti olarak hedef noktaları, feromon güncellemesini ise kurt kolonisi mantığına göre gerçekleştirmekte. Böylece küresel bilinmeyen bir ortamda istenilene yakın yerel koordinat ve bilgilere yakınsama ile hızlı yolu dinamik olarak elde edebildiği klasik karınca kolonisi ile karşılaştırıldığında görülmüştür.

Rao ve ark. [93] En uygun yol oluşumunda yolların keskinliklerinin azaltılması hedefe ulaşımı hızlandırır ve enerji tasarrufunu artırır. Gri kurt algoritması ile yerel bilgilerden küresel hareket oluşumuna kolay geçilerek yumuşak eğrili bir yolun elde edilmesi sağlanmış. Yumuşak eğri ile hareket olayı azaldığından enerji ve süre tasarrufu ile hızlı ulaşım sağlanmış. Bunu da kurt kolonisinin birlikte hareketi ile iletişim becerileri geliştirmiştir. Olasılık yol haritalama ile karşılaştırıldığında daha hızlı ve yumuşak bir yol elde edilmiştir.

Zu ve ark. [94] Çoklu İHA takımlarının koordineli olarak gerçek zamanlı bilinmeyen dinamik ortamda engellerden kaçınarak, çarpışmadan hedefe ulaşımı için optimize edilmiş RRT yaklaşımı önerilmiştir. RRT ile oluşan yoldaki gereksiz düğümlerin silinmesi ve engellerin hızlı tespiti üzerine odaklanılmıştır. Dinamik engellerin hızlı tespitiyle yeni hızlı yol oluşumu gerçekleştirilmiştir. Çalışmanın başarımının iyi olduğu simülasyonlarla gözlenmiştir.

Tablo1. Kullanım alanlarına göre algoritmalar ve başarımları

Kullanılan Algoritmalar	Ref.	Kullanım Amacı	Başarımları
Gizli Markov Modeli, Adaptif Gizli Markov Modeli	[76]	İnsan niyet tahmini, interaktif etkileşim, insanların davranışlarını değiştirme	Deneyisel olarak %68 oranında başarımlı niyet tahmini yapılmış ve Alzheimer hastalarına yardımcı araç geliştirilebileceği gözlenmiştir.
	[79]	Robotların durum tahmini, insan durum algılaması, durum öğrenme	Reaktif ve proaktif davranışların %100 oranında tanımlanmıştır.
Monte Carlo + G. Kalman Filtresi	[15]	Ortam yerleştirme ve lokalizasyon	Hedefe ulaşmada ortalama standart sapmanın 5,1cm'den 0,6 cm'ya kadar gerilediği gözlenmiştir.
(RHPSO)	[5]	Çoklu robotlarının hareketleri sırasında çarpışmadan kaçınma	Bozulmanın %10'nu geçmesi halinde çarpışmaların gerçekleştiği, ortalama hareket yaklaşımının 0,4m
PSO	[78]	Parçacık sürü algoritmasıyla optimize edilmiş bir SVM (destek vektör makinesi) kullanılarak el hareketlerinin anlaşılması	%77-%81 aralığında başarımın yüksek olduğu, ancak PSO popülasyonunun küçük seçilmesiyle başarımın %65 gibi oranlara gerilediği tespit edildi.
	[91]	Uygun yol planlama	Daha kısa sürede, çarpışmadan kaçınma başarısı yüksek
	[83]	Robot sürülerinde robot liderin konumunu tahmin edip harekete devam edebilmesi	10 ve 20 saniyeleri arasındaki kesilmeye rağmen takipçilerin lideri takibi devam etmiş

	[17]	Çoklu robot ve hareketli hedefli ortamda bir mobil robotun hareketi	Hedefe ulaşım süreleri maksimum 6,9 sn olarak engellere çarpmadan gerçekleştirilmiştir.
	[16]	Kapalı ve bilinmeyen ortamda hareket	Daha az ortam bilgisiyle hareket başarımı göstermiştir.
Rastgele (PSO)	[78]	Parçacık sürü algoritmasıyla optimize edilmiş bir SVM (destek vektör makinesi) kullanılarak el hareketlerinin anlaşılması	%77-%81 aralığında başarımın yüksek olduğu, ancak PSO popülasyonunun küçük seçilmesiyle başarımın %65 gibi oranlara gerilediği tespit edildi.
PSO ve WMPSO	[13]	En uygun yol planlaması	WMPOS 'un yakınsamasının PSO ve MPSO'ya oranla daha iyi olduğu, ancak dağılımın ise kötü olduğu gözlenmiş
Davranış tabanlı PSO ve Bulanık Mantık	[81]	Robotun hedefe doğru bir şekilde yönlendirilmesi	Daha erken nesillerde etkili arama davranışı sergilediği ve öğrenme başarımının %100 olduğu gözlenmiş
	[18]	Engellere ve bilinmeyen ortama göre en iyi rotayla belirtilen hedefe ulaşması	Kısa mesafede ve başarılı uygunluk değeriyle hedefe en kısa uygun yolla ulaştığı belirlenmiştir
Levy Uçuşu	[44]	Hedef tespiti, doğru konumda en kısa adımda hedefe ulaşması	Levy uçuş dağılımının sabit ve rastgele normal yürüyüş adım boyutuna göre daha başarılı olduğu gözlenmiştir
Levy Uçuşu dağılımı ve Karınca kolonisi algoritması	[82]	Robot sürüsünün etkili alan kaplaması ve tehdit algılaması üzerine karınca feromon kaynaklı hareket stratejisi oluşturmak	DeneySEL sonuçta modelin güven seviyesinin %95 olduğu tespit edilmiştir.
Petri Ağı	[27]	Sürü robotlarda görev planlarının kesilmesi durumunda, plan tekrarı	Planlamaya göre yapılan işlemlerde robotun kesmeleri kontrol edebildiği gözlenmiştir.
SLAM (IPSO-DPF SLAM)	[84]	Normal PSO'ya oranla önerilen algoritmanın daha uygun navigasyon sağlayıp sağlayamayacağı hedeflenmiştir	Sonuçta önerilen algoritmanın daha uzun periyotlarda daha iyi tahminde bulunduğu gözlenmiş
	[14]	Statik engellere rağmen hedefe en uygun yolla gitmek	Sonuçta önerilen algoritmanın yol uzunluğunu koruduğu ve zaman sınırı içerisinde hedefe ulaştığı belirlendi.
	[89]	Belirsiz ve değişen çevre koşullarına hızlı cevap veren bir yaklaşım amaçlanmıştır	Genetik, fuzzy gibi yöntemlerle karşılaştırıldığında %6 oranında daha başarımlı olduğu tespit edilmiştir.
A*(star)	[58]	Dinamik engellere çarpmadan hedefe ulaşmak	Her düğüm için hesaplama yapmayı sadece çarpışma olmadan engellere göre hesaplama yaparak işlem hızını arttırmıştır. %95 zaman tasarrufu sağlamıştır.
	[57]	Izgara hücre yaklaşımında hesaplanan izgara hücre sayısını azaltarak işlem hızını arttırmak	Gereksiz yönelmeler kırılabilir, işlem süresi az ve hızlı yakınsama ancak işlem süresini azaltmak için uzun yol oluşturabilir.
	[75]	Hedefe en uygun ve en az maliyetle gitmesi	İşlem süresi kısa, hızlı yakınsama ancak yaklaşık sonuç üretir
	[85]	Bilinmeyen ve statik engelli ortamda mobil robotun kaynaktan hedefe en uygun yolu çarpmadan gerçekleştirmesi	Daha hızlı ve kısa işlem süresine sahip bir algoritma geliştirilmiştir
Genetik Algoritma (GA)	[90]	Çoklu mobil robotlarda uygun yol bulmak için geliştirilmiş GA yaklaşımı önerilmiştir	A star, PSO, PRM, B-RRT gibi algoritmalarla karşılaştırılmış çalışma süresi ve yol planlama başarımının daha yüksek olduğu gözlenmiştir
	[80]	Endüstriyel hatlarda robotların başarımlarını arttırmak için GA ve PSO ile hibrit bir yaklaşım sunulmuş	Performans verimi olarak GA %69,4, PSO %74 ve GA-PSO hibritin %79,8 olduğu belirlendi
Kalman Filtresi	[25]	Doğrusal ve doğrusal olmayan alanlarda doğru yol kestirimi ve takibi sağlamak	Adaptif ve genişletilmiş kalmanın hibrit kullanımının daha az hatayla hareket takibi yaptığı gözlenmiş
	[77]	3D kamera ve IMU kullanarak insan eliyle kontrol edilen robot sistemler için insan elinin konumunu en az hatayla bulunması	Kalman ve parçacık filtresi ile hibrit yapının her biri ile kıyaslandığında daha başarılı ve hassas olduğu gözlemlendi.
Yapay Potansiyel Alan (YPA)	[88]	Uygun yol bulma ve çarpışmadan kaçınma	Yolu kısaltma ve çarpışmadan kaçınmada klasik YPA yöntemine göre daha başarılı.
	[87]	YPA algoritmasındaki takılma sorununa sanal hedef ile çözüm	Sanal hedefler ile hareketin devamlılığı ve çakılması engellenmiştir
	[86]	YPA dezavantajı olan çakılma ve durmayı sanal daire ve hedef ile engelleme	Sıkışma ve çakılı kalma durumunu çözmüştür.
Karınca Kolonisi	[92]	Dinamik ve bilinmeyen çevrelerde en uygun yolu bulma	Feromon güncellemenin kurt kolonisi optimizasyonu hızlı dinamik hareket sağlamıştır.
Gri Kurt Kolonisi	[93]	Yumuşak yol eğrisi ile daha tasarruflu ve hızlı yol planlama oluşumu	Olasılık yol haritalama ile kıyaslandığında toplam hareket süresinin daha kısa olduğu, robotun daha az manevra yaptığı saptanmıştır.
RRT	[94]	Dinamik bilinmeyen ortamda hızlı engel tespiti ve gereksiz manevra noktalarının azaltılması	Simülasyonda başarımının tatmin edici olduğu gözlemlendi.

5. SONUÇ

Bu çalışmada, robotlarla ilgili yapılan uygulamalarda ortaya çıkan problemlerin çözümlerinde kullanılan algoritmalar incelenmiştir. Sürü parçacığı algoritması, genetik algoritma, karınca kolonisi, yol haritalama yöntemi, hızlı rastgele ağaç keşfi, potansiyel alan yöntemi, kalman filtre kullanımı, monte carlo yöntemi, levny uçuşu, petri ağları, ayarlanabilir özerklik yaklaşımı, bulanık mantık gibi algoritma, metotlar ve sınıflandırıcıların yaygın olarak kullanıldığı gözlenmiştir. Belirlenen algoritmaların hangi alanlarda hangi amaçlarla kullanıldığı ve aynı zamanda problemlere yönelik başarımları ile hangi konuda daha etkin çözümler sunduğu tespit edilmiştir. Genellikle çalışmalarda hibrit yaklaşımların daha çok kullanıldığı ve başarımları yükselttiği tespit edildi.

KAYNAKLAR

- [1] Yapıcı KO. 14 Serberstlik dereceli iki ayaklı bir robotun dinamik yürüme hareketinin kontrolü. İstanbul: İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü;2008.
- [2] Camcı M. Ortam tarama için robotlarla duyurga ağı konumlandırma. İstanbul: İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü;2009.
- [3] Wang T, Dang Q, Pan P. A multi-robot system based on a hybrid communication approach. 2013;1(1):91-100.
- [4] Kara M. Oğul robotları yön bulma problemi. Ankara: Ankara Üniversitesi, Fen Bilimleri Enstitüsü; 2016.
- [5] Lee S, Myung H. Receding horizon particle swarm optimisation-based formation control with collision avoidance for non-holonomic mobile robots. 2015;9,2075-2083.
- [6] Turanlı M. Multi robot coverage methods. İstanbul: İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü; 2011.
- [7] Su K, Wang Y. Robot path planning based on random coding particle swarm optimization. 2015;6(4):58-64.
- [8] Abukhalil T, Patil M, Patel S, Sobh T. Coordinating a heterogeneous robot swarm using robot utility-based task assignment (RUTA). In 2016 IEEE 14th International Workshop on Advanced Motion Control (AMC). IEEE, 2016;(57-62)
- [9] Ferrús RM, Somonte MD. Design in robotics based in the voice of the customer of household robots. Rob. Auton. Syst., 2016;79,99-107.
- [10] Diprose J, MacDonald B, Hosking J, Plimmer B. Designing an API at an appropriate abstraction level for programming social robot applications. J. Vis. Lang. Comput., 2017;39,22-40.
- [11] Das B, Couceiro MS, Vargas PA. MRoCS : A new multi-robot communication system based on passive action recognition, Rob. Auton. Syst., 2016;82,46-60.
- [12] Canal G, Escalera S, Angulo C. A real-time human-robot interaction system based on gestures for assistive scenarios. 2016;149,65-77.
- [13] Gong D, Zhang J, Zhang Y. Multi-objective particle swarm optimization for robot path planning in environment with danger sources. 2011;6(8):1554-1561.
- [14] Patle B. K., Parhi D., Jagadeesh A., Sahu O. P. Real time navigation approach for mobile robot. 2017;12(2):135-142.
- [15] Tang Q., Eberhard P. Relative observation for multi-robot collaborative localisation based on multi-source signals. 2014;26(4):571-591.
- [16] Ahmadzadeh S., Ghanavati M., Branch A., Branch M. Navigation of mobiler robot using the PSO particle swarm. 2012;2(1):32-38.
- [17] Islam, M. R., Tajmiruzzaman, M., Muftee, M. M. H., & Hossain, M. S. Autonomous robot path planning using particle swarm optimization in dynamic environment with mobile obstacles & multiple target. In International conference on mechanical, industrial and energy engineering, 2014;1-6.
- [18] Deepak BBVL, Parhi DR, Raju BMVA. Advance particle swarm optimization-based navigational controller for mobile robot. Arabian Journal for Science and Engineering, 2014;39(8),6477-6487.
- [19] Akyol S, Alataş B. Güncel sürü zekası optimizasyon algoritmaları . Nevşehir Üniversitesi Fen Bilimleri Enstitüsü Dergisi, 2012;1(1).
- [20] Ortakci Y, Göloğlu C. Parçacık sürü optimizasyonu ile küme sayısının belirlenmesi, 2012;1-6.
- [21] Dewang HS, Mohanty PK, Kundu S. A robust path planning for mobile robot using smart particle swarm optimization. Procedia computer science, 2018;133,290-297.
- [22] Pekdemir E, Güneş M. Otonom bir mobil robotun parçacık sürü algoritması ile optimum yörünge kontrolü, 2016;19(3):157-164.
- [23] Suliman C, Crucearu C, Moldoveanu F. Mobile robot position estimation using the Kalman filter. 2009;6,75.
- [24] Sharing TI, Kalman filtresi ve bir navigasyon uygulaması. 2012;1-5.
- [25] Güney S, Bilen M. The comparison of estimation algorithms for mobile robot navigation. 24th (SIU) IEEE. 2016;797-800.
- [26] Özcan F. Available from: <https://ozcanfatih.wordpress.com/2014/05/10/kalman-filtresi/> [cited 2014 May 10].
- [27] Farinelli A, Raeissi MM, Marchi N, Brooks N, Scerri P. Interacting with team oriented plans in multi-robot systems, Auton. Agent. Multi. Agent. Syst., 2017;31(2):332-361.
- [28] Gündoğar E, İpek M. Petri ağları ile modelleme esasları.Sakarya University Journal of Science, 2016;1(1):23-28.
- [29] Durmuş MS, Söylemez MT. Petri ağları ile demiryolu anlaşılan ve sinyalizasyon tasarımı. Elektrik Elektronik Bilgisayar Mühendisliği Sempozyumu ve Fuarı. 2008.
- [30] Atanak MM, Özkan HA, Hocaoglu FO. Petri ağlarında en kısa yol probleminin pekiştirilmeli öğrenme metodu ile çözülmesi. 2005
- [31] Akpolat ZH, Application of Fuzzy-sliding mode control and electronic load emulation to the robust

- control of motor drives, University of Nottingham, England, 1999.
- [32] Koca GÖ. Kayma kipli kontrol ve bulanık mantık kullanarak elektrikli araçların hız kontrolü. F.Ü. Fen Bilimleri Enstitüsü, 2005.
- [33] Roza L, Calinon S, Caldwell DG, Jim P. Learning physical collaborative robot behaviors from human demonstrations. 2016;32(3):513–527.
- [34] Can T, Öz E, Tuncay CAN, Ersoy ÖZ. Marka tercihlerine ve gizli markov modelinin uygulanması. Eskişehir Osmangazi Üniversitesi Sosyal Bilimler Dergisi, 2009;10(2):167-185.
- [35] Argunşah AÖ, Cetin M. A brain-computer interface algorithm based on Hidden Markov models and dimensionality reduction. In 2010 IEEE 18th Signal Processing and Communications Applications Conference (pp. 93-96). IEEE. 2010
- [36] Haberdar H. Saklı markov model kullanarak görüntüden gerçek zamanlı türk işaret dili tanıma sistemi. İstanbul. Yıldız Teknik Üniversitesi, Fen Bilimleri Enstitüsü, 2005.
- [37] Zhu Q. Hidden Markov Model, 1991;7(3):533–536.
- [38] Yang J, Member S, Xu Y, Chen CS, Member S. Hidden markov model approach to skill learning and Its application to telerobotics. 1994;10(5).
- [39] Öztürk L. Monte carlo simülasyon metodu ve bir işletme uygulaması. 116-122.
- [40] Çavuş TF, Yanıkoğlu E, Yılmaz AS. Paralel sistemlerin monte carlo yöntemi ile güvenilirlik analizi. 36–39.
- [41] Bayaliev U, Brimkulov U, Sultanov R. Seyyar robotlarda kullanılan stokastik konum belirleme algoritmalarının karşılaştırmalı analizi. Manas Mühendislik Dergisi. 2015;3(1):21-34.
- [42] Wen F, Qu Z, Wang C, Hu B. Precise localization of indoor mobile robots in FMS based on distributed vision. 2008;2412–2416.
- [43] Dellaert F, Fox D, Burgard W, Thrun S. Monte carlo localization for mobile robots. In ICRA. 1999;2:1322-1328.
- [44] Katada Y, Nishiguchi A, Moriwaki K, Watakabe R. Swarm robotic network using Lévy flight in target detection problem. Artif. Life Robot., 2016;21(3):295–301.
- [45] Sutantyö DK, Kernbach S, Levi P, Nepomnyashchikh VA. Multi-robot searching algorithm using Lévy flight and artificial potential field, 2011;2–7.
- [46] Randall G. Lecture 12: Levy Flights. ss. 1–8, 2003.
- [47] Sutantyö D, Levi P, Möslinger C, Read M. Collective-adaptive lévy flight for underwater multi-robot exploration. In 2013 IEEE International Conference on Mechatronics and Automation (pp. 456-462). IEEE. 2013.
- [48] Chechkin AV, Metzler R, Klafter J, Gonchar VY. Introduction to theory of Levy flights. sayı 1, ss. 1–41.
- [49] Saldívar O. Levy flight as a robotic search pattern. massachusetts institute of technology, 2012.
- [50] Özçakar N. Genetik algoritmalar. İşletme Fakültesi Dergisi, C. 27, ss 1 1998 s. 69-82
- [51] Öznur İŞÇ, Korukoğlu S. Genetik algoritma yaklaşımı ve yöneylem araştırmasında bir uygulama. Yönetim ve Ekonomi: Celal Bayar Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi, 10(2), 191-208.
- [52] Yıldız Y, Çonkur E. Potansiyel alan metodu ile engelli bir alanda hedefine ulaşabilecek bir mobil robot için yazılan simülasyon programı. 2010.
- [53] Çınar E, Parlaktuna O, Yazıcı A. Robot navigasyonunda potansiyel alan metodlarının karşılaştırılması ve iç ortamlarda uygulanması, 2007;12:603-608.
- [54] Rimon E, Koditschek DE. Exact robot navigation using artificial potential functions. Departmental Papers (ESE), 1992;323.
- [55] Vaščak J. Navigation of mobile robots using potential fields and computational intelligence means. 2007;4(1):63–74.
- [56] Kala R. Code for robot path planning using artificial potential fields, 2014;2–4.
- [57] Babinec FDĚA, Kajan M, Be P, Florek M. Path planning with modified A star algorithm for a mobile robot. 2014;96:59–69.
- [58] Guruji AK, Agarwal H, Parsediya DK. Time-efficient A * algorithm for robot path planning. 2016;23:144–149.
- [59] Mahadevi S, Shylaja KR, Ravinandan ME. Memory based A-star algorithm for path planning of a mobile robot. 2014;3(6):1351–1355.
- [60] Russell SJ, Norvig P. Artificial intelligence: A modern approach, c. 9, sayı 2. 1995.
- [61] Vonásek V, Faigl J, Krajník T, Přeučil L. RRT-path—a guided rapidly exploring random tree. In Robot Motion and Control (pp. 307-316). Springer, London. 2009
- [62] Bry A, Roy N. Rapidly-exploring random belief trees for motion planning under uncertainty. IEEE International Conference on Robotics and Automation (pp. 723-730) 2011.
- [63] Ferguson D, Kalra N, Stentz A. Replanning with rrt*. ICRA. (pp. 1243-1248). IEEE. 2006.
- [64] Koyuncu E. Probabilistic motion planning in complex environments for unmanned aerial vehicles. İstanbul: İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, 2008.
- [65] Kavraki L, Svestka P, Overmars MH. Probabilistic roadmaps for path planning in high-dimensional configuration spaces, 1994;1–34.
- [66] Değertekin SÖ, Lamberti L, Ülker M. Uzay kafes yapılarının ateş böceği algoritması yöntemiyle optimizasyonu. 2015;390–397.
- [67] Belen MA, Alıcı M, Çor A, Güneş F. Ateşböceği algoritması ile mikrodalga transistör performans karakterizasyonu, ss. 27–29, 2014.
- [68] Ali N, Othman MA, Husain MN, Misran MH. A review of firefly algorithm, 2014;9(10):1732–1736.
- [69] Fister I, Yang XS, Fister D. Firefly algorithm: a brief review of the expanding literature. In cuckoo search and firefly algorithm Springer, Cham. 2014;347-360.
- [70] G Sakir. Available from: <https://www.emaze.com/@AQLRIRFW> [cited 2017 Apr 11].
- [71] Wu HS, Zhang FM. Wolf pack algorithm for unconstrained global optimization. Mathematical Problems in Engineering, 2014.

- [72] Hongdan L, Sheng L, Zhuo Y. Application of adaptive wolf colony search algorithm in ship collision avoidance. *International Journal of Simulation: Systems, Science and Technology*, 2015;16(2A):14.
- [73] Changan LIU, Xiaohu YAN, Chunyang LIU, Hua WU. The wolf colony algorithm and Its application., 2011;20(2).
- [74] Mirjalili A. Available from: <http://www.alimirjalili.com/GWO.html> [cited 2015 May 17].
- [75] Erdoğan ŞZ. Kendini klonlayan karınca kolonisi yaklaşımıyla optimal yolun bulunması. Edirne: Trakya Üniversitesi, Fen Bilimleri Enstitüsü, 2008.
- [76] Durdu A, Erkmén I, Erkmén AM. Estimating and reshaping human intention via human-robot interaction, *Turkish J. Electr. Eng. Comput. Sci.*, 2016;24(1):88–104.
- [77] Du G, Zhang P. Robotics and Computer-Integrated Manufacturing A novel human – manipulators interface using hybrid sensors with Kalman filter and particle filter. *Robot. Comput. Integr. Manuf.*, 2016;38:93–101.
- [78] Liu J, Luo Y, Ju Z. An interactive astronaut-robot system with gesture control., 2016.
- [79] Rozo D, Leonel S, Joao C. Learning controllers for reactive and proactive behaviors in human – robot, 2016;3:1–11.
- [80] Mousavi M, Yap HJ, Musa SN, Tahriri F, Dawal SZ. Multi-objective AGV scheduling in an FMS using a hybrid of genetic algorithm and particle swarm optimization. ss. 1–25, 2017.
- [81] Adriansyah A, Gunardi Y, Ihsanto E. Goal-seeking behavior-based mobile robot using particle swarm fuzzy controller. 2015;13(2):528–539.
- [82] Schroeder A, Ramakrishnan S, Kumar M, Trease B. Efficient spatial coverage by a robot swarm based on an ant foraging model and the Lévy distribution, c. 11, sayı 1. Springer US, 2017.
- [83] Abbas R, Wu Q. Communication failure in formation control of multiple robots based on particle swarm optimization algorithm. 2015;5–8.
- [84] Pei F. Distributed SLAM system using particle swarm optimized particle filter for mobile robot navigation. 2016;994–999.
- [85] Sudhakara P, Ganapathy V. Trajectory planning of a mobile robot using enhanced A-star algorithm, 2016;9:1–10.
- [86] Song M, Yang J, Wang Y, Yu C, Zhao D. Path planning algorithm based on an improved artificial potential field for mobile service robots. *IEEE*. 2018;441-445.
- [87] Siming W, Tiantian Z, Weijie L. Mobile robot path planning based on improved artificial potential field method. (IRCE) 2018;29-33.
- [88] Yongshen L, Xuerong Y, Yajun Y, Shengdong P. Formation control of UGVs based on artificial potential field. *IEEE*, 2018;6830-6835.
- [89] Patle BK, Pandey A, Jagadeesh A, Parhi DR. Path planning in uncertain environment by using firefly algorithm. *Defence technology*, 2018;14(6):691-701.
- [90] Nazarahari M, Khanmirza E, Doostie S. Multi-objective multi-robot path planning in continuous environment using an enhanced genetic algorithm. *Expert Systems with Applications*, 2019;115:106-120.
- [91] Dewang HS, Mohanty PK, Kundu S. A robust path planning for mobile robot using smart particle swarm optimization. 2018;133:290-297.
- [92] Che H, Wu Z, Kang R, Yun C. Global path planning for explosion-proof robot based on improved ant colony optimization. (ACIRS) *IEEE*, 2016;36-40.
- [93] Rao AM, Ramji K, Kumar TN. Intelligent navigation of mobile robot using grey wolf colony optimization. *Proceedings*, 2018;5(9):19116-19125.
- [94] Zu W, Fan G, Gao Y, Ma Y, Zhang H, Zeng H. Multi-UAVs Cooperative Path Planning Method based on Improved RRT Algorithm. In 2018 *IEEE International Conference on Mechatronics and Automation (ICMA) IEEE*, 2018;1563-1567.