



Aiming of Optimum Energy Efficiency with Rapidly-Exploring Random Tree and Artificial Bee Colony Path Planning Algorithm for Autonomous Robots

 Yunis TORUN^{1,*} , Züleyha ERGÜL², Ahmet AKSÖZ³ 
^{1,2}Sivas Cumhuriyet University, Electrical and Electronics Engineering Department, 58340, Merkez/SIVAS

³Sivas Cumhuriyet University, Department of Energy Science and Technology, Merkez/SIVAS

Graphical/Tabular Abstract

It is of high importance for autonomous robots that no collision occurs during operation and the target position is reached by using the softest and shortest route. While using the defined route, maximum efficiency is achieved by minimizing energy consumption. Thus, route calculation and optimization are one of the most important criteria's when it comes to continuous improvement.

Article Info:

Research article

Received: 20/08/2019

Revision: 29/09/2019

Accepted: 29/10/2019

Highlights

- ABC algorithm.
- Optimum path planning.
- Matlab-Simulink

Keywords

Rapidly-Exploring
Random Tree,
Artificial Bee Colony
algorithm,
Optimum path planning,
Energy efficiency,
Autonomous robots

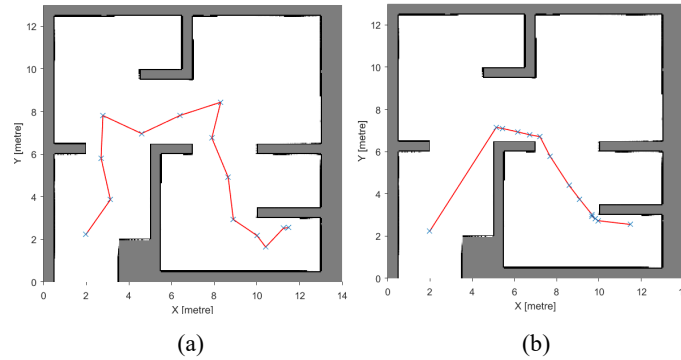


Figure A a. Start point (B) and final point (S) with RRT
b. Start point (B) and final point (S) with hybrid RRT-ABC

Purpose: In this study a hybrid of heuristic and classic route planning approaches have been used to combine their advantages while minimizing the disadvantages of each method.

Theory and Methods: Initially the optimum route for a defined starting and destination point was determined by using Rapidly-exploring Random Tree-RRT method of the classic approach and artificial bee colony-ABC method of the heuristic approach separately. Afterwards a hybrid approach was used and compared by using the Robotic System Toolbox of MATLAB. The results of the study show that energy consumption of a two wheeled mobile robot is higher when using either pure heuristic approach in comparison to a RRT-ABC hybrid approach.

Results: Since the RRT algorithm operates much faster than the hybrid algorithm and the hybrid algorithm follows a shorter path than the RRT, the spent energy values are very close to each other. As a result of these two parameters balancing each other, it was observed that the energy consumed in the hybrid algorithm approached the energy consumed in the RRT algorithm compared to the energy consumed in the ABC algorithm. It was observed that the RRT-ABC hybrid algorithm consumed less energy than ABC alone, taking advantage of the applicability of real time problems due to ABC being an heuristic algorithm in Section 3.

Conclusion: The RRT algorithm tries to find the shortest path, avoiding obstacles as it moves from the starting position to the target position in a previously discovered environment. But since it is a classical method, the path it finds is not optimal. Since ABC is a heuristic algorithm, it aims to find the universal optimum value. The RRT-ABC algorithm has an optimization process for road length, road smoothness and road safety. The success of the RRT-ABC algorithm stems from the advantage that the RRT calculation time is low and the universal minimum can be found by the ABC algorithm. Our proposed algorithm is faster, smoother, and safer than the ABC algorithm, as seen in comparisons in experimental studies.



Optimum Enerji Verimliliğini Hedefleyen Rastgele Ağaçlar ve Yapay Arı Kolonisi Yöntemi ile Otonom Robotlarda Yol Planlama Algoritması

Yunis TORUN^{1,*}, Züleyha ERGÜL², Ahmet AKSÖZ³

^{1,2}Sivas Cumhuriyet Üniversitesi, Mühendislik Fakültesi, Elektrik-Elektronik Mühendisliği Bölümü, 58340, Merkez/SIVAS

³ Sivas Cumhuriyet Üniversitesi, Enerji Bilimi ve Teknolojisi Bölümü, Merkez/SIVAS

Öz

Operatörsüz hareket eden robotların (otonom robotlar) hareket sırasında engellere çarpmadan, en kısa yol ve en yumuşak yolu seçerek hedef konumuna ulaşması büyük önem taşımaktadır. Bu çalışma, yol planlama eyleminde sezgisel ve klasik yöntemlerin avantajlarını birleştirmek, dezavantajlarını ise minimize etmek için, iki yöntemin melez kullanımı ile gerçekleştirildi. Öncesinde klasik yöntemlerden, Rastgele Ağaçlar yöntemi (Rapidly-exploring Random Tree-RRT) ve sezgisel yöntemlerden de Yapay Arı Kolonisi yöntemi (Artificial bee colony-ABC) ayrı ayrı harita üzerinde değerlendirildi. Akabinde bu iki yöntemin melez kullanılmasıyla oluşturulan yaklaşım değerlendirildi. Söz konusu yapılan bu melez değerlendirme, önceden keşfedilmiş, başlangıç ve hedef noktası belli bir haritada yapıldı. Yol kriterlerini optimize ederek, MATLAB Robotik Sistem Araç Kutusu (Robotic System Toolbox) üzerinden benzetimi gerçekleştirildi. Sunulan melez algoritmada, takip edilen yol hesaplanırken, enerji verimliliği ile birlikte yol güvenliği de dikkate alındı. İki tekerli mobil robotun enerji tüketimini RRT, ABC ve melez RRT-ABC yöntemlerinin kullanılması ile elde edilen yollarda hesaplandı ve karşılaştırıldı. Yapılan karşılaştırmalar sonucunda melez algoritmanın daha verimli çalıştığı gözlemlendi.

Makale Bilgisi

Araştırma makalesi
Başvuru: 20/08/2019
Düzeltilme: 29/09/2019
Kabul: 29/10/2019

Anahtar Kelimeler

Rastgele Ağaçlar yöntemi,
Yapay Arı Kolonisi algoritması,
Optimum rota planlama,
Enerji verimliliği,
Otonom robot.

Keywords

Rapidly-Exploring
Random Tree,
Artificial Bee Colony
algorithm,
Optimum path planning,
Energy efficiency,
Autonomous robots

Aiming of Optimum Energy Efficiency with Rapidly-Exploring Random Tree and Artificial Bee Colony Path Planning Algorithm for Autonomous Robots

Abstract

It is of high importance for autonomous robots that no collision occurs during operation and the target position is reached by using the softest and shortest route. While using the defined route, maximum efficiency is achieved by minimizing energy consumption. Thus, route calculation and optimization are one of the most important criteria's when it comes to continuous improvement. In this study a hybrid of heuristic and classic route planning approaches have been used to combine their advantages while minimizing the disadvantages of each method. Initially the optimum route for a defined starting and destination point was determined by using Rapidly-exploring Random Tree-RRT method of the classic approach and artificial bee colony-ABC method of the heuristic approach separately. Afterwards a hybrid approach was used and compared by using the Robotic System Toolbox of MATLAB. The results of the study show that energy consumption of a two wheeled mobile robot is higher when using either pure heuristic approach in comparison to a RRT-ABC hybrid approach.

1. GİRİŞ (INTRODUCTION)

Gelişen teknolojiyle insansız hava/yer araçlarının farklı uygulama alanları, düşük maliyet, yüksek manevra yetenekleri, can güvenliği gibi birçok avantajlarından dolayı askeri veya sivil alanlarda tercih edilmektedir [1]. Askeri alanda kullanılan kâşif dronlar ve otonom hareket edebilen robotlar örnek olarak sayılabilir [1].

Operatörsüz hareket eden otonom araçlar, temel anlamda insan müdahalesi olmadan bir konumdan başka bir konuma kendi karar verip hareket edebilen araçlardır. Bu işlemi yaparken mümkün olduğunca minimum enerji harcaması verimlilik açısından en önemli konulardandır. İzlenecek rota robot tarafından belirlenirken yapılan hesapta yol planlama ve hareket kontrolü aşamalarını gerçekleştirilmesi gerekmektedir[2]. Yol

planlama aşamasında, başlangıç ve bitiş noktaları arasında hesaplanan rotalardan birinin seçimi yapılır. Ardından hareket kontrolü aşamasında ise robot belirlenen rotanın planlandığı şekilde harekete geçmesi ve güvenli bir şekilde hedef konumuna ulaşması istenir.

Yol planlama aşamasında genel olarak tüm yaklaşımları kapsayan global ve lokal olmak üzere iki temel teknik vardır[3]. Çevrimdışı olarak da adlandırılabilen global yaklaşıma göre ortam önceden keşfedilmiş veya harita önceden bellidir. Verilerini ortamdan bilgi almayarak güncellemediği için dinamik engellerden kaçınmaz. Bulunan yol buna bağlı olarak düşük çözünürlüklüdür. Çevrimiçi yani lokal yaklaşımda ortam hakkında önceden bilgi sahibi olmasa dahi etrafındaki algılayıcılar sayesinde robot çevreyi algılayarak dinamik ve statik engelleri fark edebilir. Robot bu yaklaşım sayesinde, yüksek çözünürlüklü bir yol izleyebilir ama hedefi uzak mesafelerde ve/veya karmaşık ortamlarda verimsiz çalışabilir.

Yol hesaplama algoritmaları klasik ve sezgisel olmak üzere iki ana kategoride sınıflandırılır [3,4]. Klasik yöntemler, lokal optimum çözümü bulurlar ve dinamik ortamlarda doğru çalışmayabilirler [4]. Bu dezavantajlara rağmen klasik yöntemler hızlı çalışma zamanına sahiptirler [4]. Sezgisel yaklaşımlar ise dinamik engellerden kaçınabilir ve global optimum çözümü hesaplayabilirler [4]. Literatürde iki yöntemin dezavantajlarını minimize etmek ve avantajlarını güçlendirmek için melez kullanımı önerilmiştir [4].

Klasik yöntemler; Hücre Ayrışımı yöntemi (Cell Decomposition-CD) [4,5], Potansiyel Alan yöntemi (Potential Field-PFM) [6,7,8], Alt Hedef Ağı yöntemi (Subgoal Method-SG) [9,10,11], Örnekleme Tabanlı yöntemler (Sampling-based methods-SBP) [12] kendi arasında Rastgele Ağaçlar yöntemi (Rapidly Exploring Random Trees-RRT) ve Olasılıksal Yol Haritası yöntemi (Probabilistic Road Map-PRM) olarak karşımıza çıkar. Sezgisel yöntemlerden, birden fazla değişken komşuluk araştırması [13], Yapay Sinir Ağları (Neural Network-NN) yöntemi [14,15,16], Bulanık Mantık (Fuzzy Logic-FL) [17,18], Karar Destek Makinaları [19], Doğadan Esinlenen algoritmalar; Genetik algoritması (Genetic Algorithms-GA) [20,21,22,23], Yarasa algoritması (Bat algorithm) [24], Parçacık Sürüsü optimizasyonu (Particle Swarm Optimization-PSO) [25,26], Karınca Kolonisi optimizasyonu (Ant Colony Optimization-ACO) [27], Yapay Arı Kolonisi Algoritması (Artificial Bee Colony-ABC) [28] ve bunların melez algoritma çalışmaları yapılmıştır.

Otonom araçların, hesapladığı optimum yolu takip ederken minimum enerji tüketmesi istenir. Harcanan enerji; belirlenen yolun uzunluğuna, zeminin yapısına ve dönüş açlarına bağlı olduğu gibi aracın ağırlığına, atalet momentine, aracın yarıçapına, tekerlek yarıçapına ve yerçekimine bağlı olduğu da gözlemlenmiştir. Literatürde iki tekerli bir otonom robotun harcadığı enerji matematiksel olarak tek bir yolla hesaplanmıştır[28]. Çalışmamızda enerji hesaplarken bu modeli kullanarak karşılaştırma yapılmıştır.

Bu çalışmada klasik yöntemlerden Rastgele Ağaçlar yöntemi (RRT) ile sezgisel yöntemlerden Yapay Arı Kolonisi yöntemi (ABC) kullanılarak melez bir yol planlama algoritması geliştirildi ve bulunan rotayı iki tekerli robotun takip etmesi halinde harcayacağı enerji hesaplandı.

RRT algoritmasının hızlı olması ve ABC algoritmasının optimuma daha yakın sonuçlar vermesi, bu iki yöntemin ayrı ayrı kullanılmasına göre melez kullanılmış olması daha umut verici bir yaklaşımdır. Literatürde RRT-Evolutionary Algorithm (RRT-Evrimsel algoritma) [29], PRM-GA [30], ACO-PRM [31], Potential Field-ACO [32], PSO-PRM [33] gibi melez kullanılan algoritmalar dikkat çekmiştir. Rastgele Ağaçlar yöntemi ile Yapay Arı Kolonisi yöntemi melez kullanımı literatürde ilk defa bu çalışmayla ortaya atıldı.

İlerleyen bölümde çalışmada kullanılan yöntemler açıklanmış ve ardından uygulanan deneysel çalışma sunulup karşılaştırmalar yapılmıştır. Üçüncü bölümde ise sonuçlar ve ilerletilebilecek çalışmalar sunulmuştur.

2. MATERYAL VE METOT (MATERIAL AND METHOD)

Otonom robotlarda optimum rota belirlenirken sezgisel ve/veya klasik yöntemler kullanılmaktadır. Bu çalışmada sezgisel algoritmalar ABC, klasik algoritmalar RRT kullanıldı. Ardından önerilen algoritma iki tekerlekli bir robotun hareket ederken harcadığı enerji formülü ile harcanan enerji hesaplandı. Benzetim çalışmaları MATLAB üzerinden gerçekleştirildi.

2.1. Rastgele Ağaçlar Yöntemi (Rapidly-exploring Random Tree-RRT)

RRT başlangıç noktasından hedef noktasına ilerlerken çevresinde rastgele noktalar seçerek ve bu noktalara sürekli dallanmalar gerçekleştirerek rastgele yollar belirler. Bu işleme başlangıç noktasından başlar ve çevresinde belirlenen limit değerinde rastgele noktalar atar. Bir sonraki nokta seçimlerini en yakın olan düğümü başlangıç noktası düşünerek yine rastgele yani önceki düğümleri dikkate almadan herhangi bir hesap yapmadan belirler. Ancak yeni noktalardan önceki belirlenen noktalara en yakın olanlar birbirine bağlanır. Düğümler birbirine bağlanırken engele çarpıyorsa o dallanmadan vaz geçilir. Böylece işlemler bulunan en yakın nokta hedef noktası olana kadar sürekli tekrarlanır. Bağlantılardan belirlenen harita içerisinde çeşitli dallanmalar gerçekleşerek yollar meydana getirir. Elde edilen yollardan başlangıç-hedef noktası arası en kısa olan seçilir. RRT yol planlamada ilk kez LaValle ve arkadaşları tarafından, hem sabit engellerden hem de çok boyutlu alanları hızlıca keşfetmek için verimli bir veri yapısı ve örnekleme şeması olarak tanıtılmıştır [34].

RRT' deki bir problem, rastgele tekniği kullanarak çalışma alanında birçok dal ile bir yol üretmesidir. Kuffner ve arkadaşları daha hızlı, daha optimum ve çok fazla dallanma olmadan bir çözüm elde etmek için RRT algoritmasını hem başlangıç hem de hedef noktasından başlatma fikri sunmuşlardır [35]. İyileştirilmiş RRT de başlangıç ve hedef den gelen rastgele dallanmaların oluşturduğu noktalar arası mesafe belirli bir sayının altında ise noktalar birleştirilir ve yol arama işlemi durdurulacak şekilde iyileştirilmiştir. Sonuç olarak daha az dallanma gerçekleştirerek yol planlama işlemi daha hızlı gerçekleştirilmiştir.

Ardiyanto ve arkadaşları dinamik ve karmaşık ortamlarda bir mobil robot için yeni bir yol planlama yaklaşımı sunmuştur [36]. Sezgisel Varış Zamanlı Rastgele Ağaçlar yöntemi (Heuristic Arrival Time Field-biased-HeAT- Random Tree) olarak adlandırılmış yöntem optimum yolu, güvenliği ve gerçek zamanlı uygulanabilirliği sezgisel varış zamanı alanı ile rastgele ağaçlar yönteminin birleştirilmiş halini sunmaktadır. RRT rastgele noktalar seçmek yerine varış zamanı alanından noktalar seçer ve böylece hedef noktasına doğru daha kararlı bir şekilde yönelir. Varış zamanı alanları açık renkli olanlar hedefe daha yakın olanlar iken koyu renkliler daha uzak olan bölgelerdir. Dolayısıyla seçimler açık renkli alanlardan yapılır ve RRT' nin rastgele seçimlerinin çok fazla yol oluşturmasının önüne geçilmiş olunur. Klasik RRT yöntemi oldukça hızlı bir hareket planı üretebilmesi açısından dikkat çekmektedir. Ancak tam olarak optimum yolu - en yumuşak geçişli veya en verimli yol- bulamaması ve gerçek zamanlı ortamlarda dinamik engellerden kaçamaması açısından tek başına kullanışlı değildir. Martin ve arkadaşları dinamik ortamlarda daha optimum sonuçlar elde etmek için RRT yöntemi sezgisel yöntemlerle melez kullanımını sunmuştur [29]. Çalışmada evrimsel algoritma ile RRT aynı anda karar verecek şekilde melez bir algoritma geliştirilmiş ve elde edilen sonuçlar oldukça verimli olup RRT'nin sezgisel algoritmalar ile melez kullanımının gerçek zamanlı kullanımda daha verimli sonuçlar elde edilebileceği görülmüştür.

2.2. Yapay Arı Kolonisi Algoritması (Artificial bee colony-ABC)

Sezgisel algoritmalar en popüler olanlar doğanın düzenini taklit eden algoritmalar. Bu algoritmalar doğadaki popülasyonların davranış biçimlerinden faydalanılarak geliştirilir. Bunlardan biri de arıların polen bulma eylemini taklit eden ABC algoritmasıdır. ABC, 2006 yılında Karaboğa tarafından sunulan bir arı sürüsü zekâsı algoritmasıdır [37]. Araştırmada arıların yiyecek kaynağı arama işlemlerinin kusursuz bir şekilde gerçekleştirdiği ve arılar arasındaki işbölümünün karışıklığa uğramadan belirlendiği saptanmıştır. Böylece kovandaki işlerde ne bir aksama nede bir dengesizlik meydana gelir. Bu tespitler sunulan algoritmanın hem gerçek zamanlı hem de optimum sonuçlar verebileceğine işaret etmektedir. Karaboğa arıların bu davranış biçiminin matematiksel modellemesini gerçekleştirerek algoritmasını geliştirmiştir. Bir kovandaki arı sürüsünde üç çeşit görevde arı bulunur. Bunlar kaşif arılar, işçi arılar ve gözcü arılar olarak gruplandırılırlar. Kaşif arılar kovandan çıkıp etraftaki en uygun kaynağı, belli kriterlere bağlı bir olasılık fonksiyonu ile belirlerler. Kaşif arılar kaynağı seçip nektar almaya başladığı anda işçi arılara döner. İşçi arılar nektarı alıp kovana döndüklerinde herbirini ayrı bir gözcü arı karşılar. Gözcü arılar işçi arıların danslarını takip ederek kaynaklara ulaşırlar. Kaynaklara ulaşan gözcü arılar nektar toplayıp kovana dönerler. Kaynaklardaki nektar bitene kadar bu iş bölümü devam eder. Ardından işçi arılar tekrardan kaşif arılara dönüşerek yeni kaynaklar aramaya devam ederler [37].

Algoritmanın çalışma mantığı aşağıdaki gibidir [38];

Kaynak sayısı kadar başlangıç popülasyonu belirle $x_{i,j}$

while ($i <$ maksimum iterasyon sayısı)

{ İşçi arılar için yeni $v_{i,j}$ çözümlerini $x_{i,j}$ komşuluğunda üret

$$v_{i,j} = x_{i,j} + \Phi_{ij}(x_{i,j} - x_{k,j})$$

x_i ve v_i arasında yeni ve daha iyiye eğilimli keşif operasyonunu kullan ve yeni kaynaklar bul,

for (bulunan kaynaklar için)

$$\{ \text{Olasıksal değerini hesapla } P_i = \frac{fit_i}{\sum_{i=1}^{SN} fit_i},$$

Uygunluk değerini hesaplamak için,

$$fit_i = \left\{ \begin{array}{l} \frac{1}{1+|f_i|} \text{ if } f_i \geq 0 \\ 1 + |(f_i)| \text{ if } f_i < 0 \end{array} \right\}$$

end for

Yeni çözümler üret $v_{i,j}$

Yeni ve daha iyiye eğilimli keşif operasyonunu kullan

if (Yeni bulunan $v_{i,j}$ daha iyi ise)

{ Eski $v_{i,j}$ kaynağını terk et

else (Yeni bulunan $v_{i,j}$ daha kötü ise)

Değişiklik yapma}

end if

if (Terk edilen kaynak varsa)

{Terk edilen kaynağı tanımla ve yeni rastgele belirlenen $x_{i,j}$ kaynağı ile değiştir

else (Terk edilen kaynak yoksa)

*İzci arı tarafından $x_{ij} = \min_j + \text{rand}(0,1) * (\max_j - \min_j)$ denklemini kullanarak yeni kaynak bul,*

Bulunan en iyi çözümü hafızaya kaydet }

end if }

end while (Maximum çevrim sayısı bitene kadar devam et)

Literatürde ABC algoritması bazı durumlarda Genetik Algoritma (GA), Parçacık Sürü Algoritması (PSO), Diferansiyel Gelişim algoritması (DE) ve Parçacık Sürü Evrimsel Algoritma (PS-EA) gibi diğer doğadan ilham alan algoritmalarından daha iyi çalıştığı gözlemlenmiştir[28,39]. ABC algoritması gerçek zamanlı ve karmaşık ortamlarda optimum sonuca ulaşabilir. Fakat yüksek hesaplama zamanına sahiptirler. Bunun önüne geçebilmek için bu çalışmada, ABC-RRT melez algoritması fikri iki yaklaşımın olumlu yanlarını birleştirmek amaçlı sunulmuştur.

2.3. Melez RRT ve ABC Algoritması (Heuristic RRT and ABC Algorithm)

Yapılan çalışmada yol planlama algoritmasını geliştirirken ABC algoritmasının gerçek zamanlı uygulanabilirliği ile RRT algoritmasının hızı birleştirilerek yeni melez bir ABC-RRT algoritması geliştirildi. Algoritmanın çalışma mantığına göre önceden belirlenmiş bir haritaya ve belirlenmiş başlangıç-sonuç noktalarına ihtiyaç duyar. Bu uzayda öncelikle RRT algoritması çalıştırılıp bir sonuca ulaştırılır.

Ardından belirlenen bu rota ABC algoritmasında yol uzunluğunu, güvenliğini, yumuşaklığını (akıcılığını) dikkate alarak yeni bir çözüm üretir. RRT algoritmasının belirlediği rota sayesinde daralan uzay ABC algoritmasının daha hızlı çalışmasına olanak sağlıyor.

Elde edilen sonuç yol uzunluğu, güvenliği ve yumuşaklık (akıcılık) değerlerini içeren bir maliyet fonksiyonu sayesinde optimuma yakın enerji tüketimine olanak sağlar.

$$\text{maliyet fonksiyonu} = W_p * Pl + W_a * Sm + W_s * Sa \quad 1$$

Maliyet fonksiyonu, sadece en kısa yolu hesaplamaya odaklanmaz aynı zamanda enerji tüketimini minimize etmeyi hedefler. Bunu en uygun açıyla dönmeyi ve engellerden kaçınırken optimum mesafeyi korumayı da hesaba katarak gerçekleştirir.

Denklem (1)'de Pl değişkeni metre cinsinden minimum yol uzunluğunu ifade edilirken W_p sabiti ile de kontrol edilmektedir. Yolun her bir düğümüne bağlı kenarlar arasındaki açıların ortalamasını Sm ile belirtirken bu açının akıcılık şiddeti W_a ile ayarlandı. Buradaki amaç robotun keskin virajları daha yumuşak geçişlerle tamamlamasını sağlamak, böylece enerji verimliliğini arttırmak ve aynı zamanda ömrü uzatabilmektir. Robotların engellerden kaçınırken en uygun mesafede kalabilmesi için aralarındaki en güvenli minimum mesafede kalması gerekmektedir. Bu değer Sa ile formüle dahil edildi. Ws robot boyutuna göre belirlenen, engeller ile robotun arasındaki minimum mesafenin ayarlanmasında kullanıldı. Maliyet fonksiyonunda tüm bu kısıtlar dikkate alınarak fonksiyon en küçük değeri araması sağlanır. Bu da optimale yani en az enerji harcayan en verimli yola yaklaştığı anlamına gelir.

İlk olarak RRT aşağıdaki adımları izler;

```

Haritayı çağır
Başlangıç ve bitiş noktalarını belirle
while (i < düğüm sayısı veya hedefe ulaşına kadar)
  { Harita üzerinden rastgele  $q_{rand}$  düğümlerini belirle
  for (Başlangıç noktasına en yakın  $q_{near}$  seç)
    { Başlangıç noktasından  $q_{near}$  düğümüne dal oluştur,
     $q_{near}$  düğümünden  $q_{rand}$  düğümü yönüne yeni dallanmalar için yönlendir
    if (Düğüm çok uzaksa)
      { Enterpolasyonla düğümü yakınlaştır  $q_{new}$  düğümüne ulaş
      Engele çarpmadığını kontrol et,
       $q_{near}$  düğümü ile  $q_{new}$  düğümü arası  $C_{min}$  maliyet fonksiyonunu hesapla,
      Düğüm listesine  $q_{new}$  düğümünü ekle }
    end if }
  end for }
end while
Belirlenen minimum yolu belirleyen tüm düğümlerinin x ve y değerlerini hafızaya at

```

RRT ile gerçekleştirilen yolun takip ettiği düğümler $path_{x,y} = \sum_{i=2}^{n-1} N_i$ $\{i = 1,2,3,\dots,n\}$ matris şeklinde hafızaya atılıp ABC algoritması tarafından çağırılır. Formülde n belirlenen yolu çizen düğüm sayısıdır.

Belirlenen probleme uyarladığımız ABC algoritması aşağıdaki adımları izler;

```

Repeat:
for (Engel yoksa ve  $k < M$  iken)
  { Kaynak oluştur  $F_{x,y} = \sum_{k=1}^M f_{x,y}$   $\{k=1,2,3\dots M\}$ ,  $M$  kaynak sayısıdır }
end for
for (i < n)
  { RRT'de belirlenen düğüm noktalarını (ABC'de kaynak)  $N_i = N_i + \Phi_i(N_i - F_{k,j})$ 
formülü ile

```

yenileri ile değiştir,
 Yeni düğümler için $obj = W_p * Pl + W_a Sm * + W_s Sa$ (maliyet fonksiyonu)
 fonksiyonunu hesapla
if (Yeni belirlenen düğümün maliyet fonksiyonu eskisinden daha iyi ise)
 { Hafızadaki eskinin yerini yeni düğüm ile değiştir}
end if
 Düğümlerin olasılıksal değerini hesapla
 Olasılık değerine göre rastgele düğüm oluştur
 En iyi olan yolu hafızaya kayıt et }
end for
until (İterasyon sayısına ulaşına kadar tekrar et)

RRT ve ABC melez algoritmasının geliştirilmesi ve uygulanması MATLAB Robotik Sistem Araç Kutusu (Robotic System Toolbox) içerisindeki önceden belirlenmiş Occupancy Grid haritası üzerinde gerçekleştirildi ve ardından iki tekerlekli robotun bu yolu izlemesi halinde harcadığı enerji hesaplandı.

2.4. İki Tekerli Mobil Robotlarda Enerji Modeli (Energy Model in Two Wheeled Mobile Robots)

Mobil robotlar yol alırken motorlarda tüketilen enerjinin minimize edilmesi hem enerji verimliliği açısından hem görevin mevcut sınırlı kaynaklar ile tamamlanabilmesi açısından önemlidir[29]. Dolayısıyla optimum rota belirlenirken harcanan enerjinin minimum olması istenir. Bu harcanan enerjinin optimize edilmesini sağlamak yolun en kısa mesafede en güvenilir rotada ve en uygun yumuşaklıkta kat edilmesi halinde sağlanacağı ön görülmektedir. Elde edilen optimum rotadaki enerji harcama değerini belirlemek için aracın hızına, dönüş açısına ve açısal hızına bağlı bir denklem çözümlenmelidir[29].

Liu ve Sun [29] iki tekerli mobil robotun optimum yolu hesaplarırken kullandığı maliyet fonksiyonuna enerji minimizasyon kriterini de katarak enerji harcamasını en aza indirmeyi hedeflemişlerdir. Çalışmada kinetik enerji, sürtünme enerjisi ve diğer kayıpları hesaba katarak enerji tüketimi formülleri türetilmiştir. Kinetik enerji aracın açısal ve lineer hızına bağlı olarak değişirken sürtünme kayıpları zeminin cinsine ve kat edilen yolun uzunluğuna bağlı olarak değişmektedir.

İki tekerli bir aracın sağ teker lineer hızı v_{right} sol teker lineer hızı v_{left} olarak gösterilerek denklem (2) ile hesaplanır;

$$\begin{bmatrix} v_{left} \\ v_{right} \end{bmatrix} = \begin{bmatrix} 1 & -b \\ 1 & b \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \quad 2$$

Verilen denklemde v mobil robotun lineer hızı, w açısal hızı ve b robotun iki teker arası mesafenin yarısıdır. Robotun kinetik enerji denklemi;

$$\begin{aligned} E_k(t) &= \frac{1}{2} m v(t)^2 + \frac{1}{2} I w(t)^2 \\ &= \int_t \left(d \left(\frac{1}{2} m v(t)^2 \right) + d \left(\frac{1}{2} I \omega(t)^2 \right) \right) dt \\ &= \int_t (mv(t)a(t) + I\omega(t)\beta(t)) dt \end{aligned} \quad 3$$

şeklinde yazılabilir. Formülde m robotun kütlesi ve I robotun atalet momentini ifade eden sabitler iken α ve β sırasıyla lineer ve açısal ivmelerdir.

Robotun yavaşlaması durumunda ivme negatif olur. Bu durumda formül harcanan bir miktar kinetik enerjinin elektrik enerjisine geri dönüşüymüş gibi bir sonuç verir. Ancak böyle bir durum söz konusu olmadığından kinetik enerji değerini sıfır ile pozitif değerler arasında aşağıdaki gibi sınırlanmıştır.

$$E_k(t) = \int_t (m \max\{v(t)a(t), 0\} + I \max\{\omega(t)\beta(t), 0\}) dt \quad 4$$

Yol alma sırasında sağ ve sol tekerlerle yüzey arasında bir enerji kaybı oluşur. Bu enerjiye sürtünme enerjisi denir. Sürtünme kayıpları denklem (5) ile hesaplanır.

$$E_f = 2\mu mgL \quad 5$$

Sürtünme kayıpları μ yüzeye bağlı sürtünme sabiti, m robotun kilogram cinsinden ağırlığı, g yerçekimi ivmesi ve L alınan yolun metre cinsinden değerine bağlı olarak değişmektedir. İki tekerlekli robot kullandığımızdan formülü iki ile çarpabiliriz. Denklem (5)'i iki tekerde harcanan sürtünme enerjisini hesaba katmak için iki ile çarpılmıştır.

Sürtünme enerjisi iki teker için harcanan güç cinsinden;

$$E_f(t) = \int_t (P_{left} + P_{right}) dt \quad 6$$

Şeklinde yazılabilir. Sağ ve sol tekerin harcanan gücü denklem (1)' deki hız formüllerini kullanarak aşağıdaki gibi ifade edilebilir.

$$\begin{bmatrix} P_{left} \\ P_{right} \end{bmatrix} = \mu mg \begin{bmatrix} |v(t) - b\omega(t)| \\ |v(t) + b\omega(t)| \end{bmatrix} \quad 7$$

P_{left} ve P_{right} değerleri denklem (6)' da yerine yazılırsa;

$$E_f(t) = 2\mu mg \int_t \max\{|v(t)|, |b\omega(t)|\} dt \quad 8$$

olarak ifade edilir.

Sensör den kaynaklı kayıplar, elektriksel kayıplar gibi mekanik olmayan diğer kayıplar zamanla değişmez ve dolayısıyla sabittir. Bu kayıp gücü P_s ile ifade edecek olursak denklem (9)'daki gibi kayıp enerjisi ifade etmiş oluruz.

$$E_D(t) = P_s t \quad 9$$

Sonuç olarak iki tekerli mobil robotun toplam kayıp gücü;

$$\begin{aligned} E_{total} &= E_k + E_f + E_D \\ &= \int_t (m \max\{v(t)a(t), 0\} + I \max\{\omega(t)\beta(t), 0\} + 2\mu mg \max\{|v(t)|, |b\omega(t)|\}) + P_s t \end{aligned} \quad 10$$

şeklinde ifade edilir.

Liu ve Sun [2014] çalışmalarında yukarıda verilen formüllerle elde ettiği enerji modelini kullanarak bir algoritma oluşturulmuştur. Algoritma sayesinde bulunan optimal yolu takip eden iki tekerli bir aracın rotayı izlerken harcadığı toplam enerji hesaplanabilmektedir.

İki tekerlekli mobil robotlar için Enerji Algoritması çalışmamızda MATLAB üzerinden, denklem (10)'da yerine yazılarak hesaplanmıştır. Formülde denklem (9)'da belirtilen toplam kayıp gücü ihmal edilmiştir.

3. DENEYSEL ÇALIŞMA (EXPERIMENTAL STUDY)

Bu çalışma, MATLAB Robotik Sistem Araç Kutusu (Robotic System Toolbox) kullanılarak simülasyon ortamında test edildi. İntel i7 7500U, 8 GB ram ve Ubuntu İşletim Sistemine sahip bir bilgisayar kullanıldı.

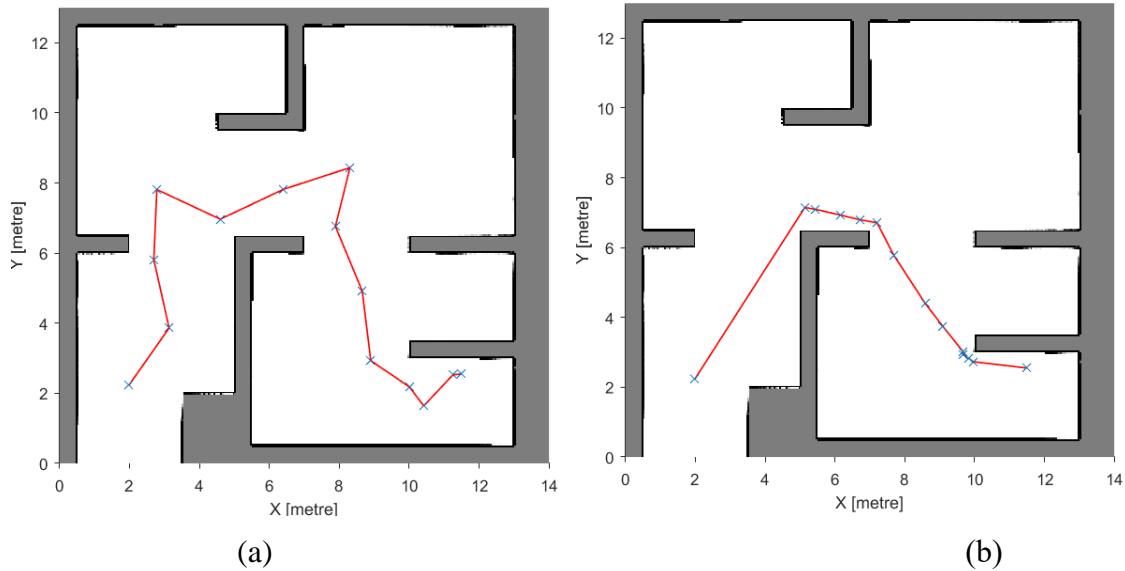
Algoritma ayrıca Robot İşletim Sistemi (ROS) ile de uyumludur, dolayısıyla gerçek zamanlı uygulamalara uygulanabilir. Bu çalışmada Gazebo 3D Robot simülatörlerine uygulanan algoritma da elde edilen veriler ile robotun harcadığı enerji hesaplandı.

MATLAB Robotik Sistem Araç Kutusu (Robotic System Toolbox) içerisindeki önceden belirlenmiş Occupancy Grid haritası iki boyutlu doluluk ızgara haritası oluşturur. Her bir hücrenin, o hücrenin doluluk olasılığını temsil eden bir değeri vardır. 1'e yakın değerler, hücrenin bir engel içerdiği yüksek bir kesinlik gösterir. 0'a yakın değerler, hücrenin engel bulundurmadığını kesin olarak belirtir. Haritanın özellikleri Tablo 1'de gösterilmiştir.

Tablo 1: Kullanılan haritanın özellikleri

Tanım	Değer
İşgal Edilen Alan Eşik Değeri	0.6500
Serbest Alan Eşik Değeri	0.2000
Olasılık Sınır Doygunluğu	[0.0010 0.9990]
Izgara Boyutu	[260 280]
Çözünürlüğü	20
X Ekseni Limit Değeri	[0 14]
Y Ekseni Limit Değeri	[0 13]
Uzaydaki Konumu	[0 0]

RRT ve RRT-ABC algoritmalarının haritada bulduğu yollar Şekil 1.a ve Şekil 1.b'de sunulmuştur.



Şekil 1. a. Başlangıç (B) ve sonuç (S) noktası belli haritada çalıştırılmış RRT
b. Başlangıç (B) ve sonuç (S) noktası belli haritada çalıştırılmış melez RRT-ABC

Kullandığımız maliyet fonksiyonunu sadece en kısa yolu hesaplamaz en ideal eğim faktörü ve en güvenli yol faktörünü de belirler. Belirlediğimiz problemde elde edilen değerler Tablo 2.'de karşılaştırıldı. En kısa yolun RRT-ABC melez algoritmasında elde edildiği ve bununla birlikte ABC algoritmasının tek başına problemi çözmesinden daha hızlı bir sonuç elde edildiği belirlendi. Robot hareket ederken keskin dönüşler yapmaması ve yumuşak geçişlerle yolunu tamamlaması, yumuşaklık değerinin (S_m) en küçük olması ile sağlanır. Melez algoritmanın RRT ve ABC ye göre daha akılcı bir yol bulduğu Tablo 2.'deki sonuçlardan açıkça görülmektedir. Güvenlik değerinin (S_a), ABC de büyük olması robotun gerekenden fazla uzaktan seyrettiğini, RRT de küçük olması güvenlik mesafesinin aşıldığını gösterir. Melez algoritmada daha uygun

bir mesafenin ayarlanması istediğimiz bir sonuçtur. Tablo 2.'de, benzetim sonuçları karşılaştırıldığında maliyet fonksiyonu en küçük olan melez algoritmanın daha verimli bir sonuç elde ettiği gösterildi. Önerilen melez algoritmanın daha yumuşak geçişlere sahip olması, minimum yolu seyretmesi daha az enerji sarfiyatı yaparak yolun tamamlanması anlamına gelmektedir. Bununla birlikte belirlediği yolun güvenli bir şekilde izlenmesi de dikkate alındı.

Tablo 2: Benzetim sonuçlarının karşılaştırılması

	RRT	ABC**	RRT-ABC
Yol Uzunluğu (PI), metre cinsinden	18.3386	19.1253	14.4993
Yumuşaklık-Akıcılık Değeri (Sm)	37.3402	52.7178	18.5757
Güvenlik Değeri (Sa)	0.3183	0.4818	0.3616
Maliyet fonksiyonu (obj fonksiyonu)	38.412	66.3731	30.1372
Bilgi işlem zamanı (saniye)	6.54	14.162	11.17

**ABC uygulaması MATLAB üzerinden Onak'ın algoritması kullanılarak gerçekleştirilmiştir [41].

Elde edilen benzetim sonuçlarının minimum enerji sarfiyatı yaptığı, maliyet fonksiyonundaki değişkenlerden tahmin edildi. Ancak kesin bir sonuca varabilmek için harcanan enerji belirlenmek istendi. Bunun için algoritmamızı iki tekerlekli robot simülasyonu olan Gazebo 3D Robot simülöründe çalıştırıp, aldığı yola karşı zaman değerleri kaydedildi. Melez algoritmanın çalıştırıldığı aynı Occupancy Grid haritası üzerinde alınan üç boyutlu uzaydaki x,y değerlerine karşılık, zaman değeri iki tekerlekli robotlarda enerji modelini kullanmaya olanak sağlandı.

Çalışmanın son aşamasında RRT, ABC ve melez RRT-ABC algoritmasının Gazebo 3D Robot simülörlerindeki değerleri kaydedilip iki tekerlekli mobil robotlarda enerji modeli algoritmasında MATLAB'da çalıştırılarak her üç yolu izlerken harcanan enerji Tablo 3.'deki gibi hesaplandı.

Tablo 3: RRT, ABC ve RRT-ABC algoritmalarında harcanan enerji

	RRT	ABC	RRT-ABC
İki tekerlekli mobil robotun harcadığı enerji (Joule)	505.2983	610.3465	519.7831

RRT algoritmasının melez algoritmaya göre çok daha hızlı çalışmasından, melez algoritmanın da RRT ye göre daha kısa bir yolu takip etmesinden dolayı, harcanan enerji değerleri birbirine çok yakın çıkmıştır. Bu iki parametrenin birbirini dengelemesi sonucunda, melez algortmada harcanan enerji, ABC algoritmasında harcanan enerjiye göre RRT algoritmasında harcanan enerjiye yaklaştığı gözlemlendi. RRT-ABC melez algoritmanın, ABC'nin sezgisel algoritma olmasından kaynaklı gerçek zamanlı problemlerde uygulanabilirliği avantajını da kullanarak, ABC'nin tek başına kullanılmasından daha az enerji sarfiyatı yaptığı gözlemlendi.

4. SONUÇ (CONCLUSION)

Bu çalışmada sezgisel ve klasik yöntemlerin avantajlarını birleştirerek, kullanılan RRT ve ABC algoritmalarına göre daha optimum bir yol elde eden yeni bir yöntem geliştirildi. RRT algoritması önceden keşfedilmiş ortamda başlangıç pozisyonundan hedef pozisyonuna ilerlerken engellerden kaçınarak en kısa yolu bulmaya çalışır. Fakat klasik bir yöntem olduğundan bulduğu yol optimum değildir. ABC sezgisel bir algoritma olduğundan evrensel optimum değerini bulmayı amaçlar. RRT-ABC algoritması, yol uzunluğu, yolun düzgünlüğü ve yol güvenliği için optimizasyon sürecine sahiptir. RRT-ABC algoritmasının başarısı, RRT hesaplama zamanının az olması avantajından ve evrensel minimumun ABC algoritması tarafından bulabilme kabiliyetinden kaynaklanmaktadır. Önerilen algoritmamız, deneysel çalışmalardaki karşılaştırmalarda görüldüğü gibi, ABC algoritmasından daha hızlı, daha yumuşak geçişli ve daha güvenlidir.

Ayrıca ABC 'den daha hızlı çalışan ve daha kısa yoldan takip eden RRT algoritması ile melez ABC-RRT algoritmasının enerji sarfiyatı açısından karşılaştırdığımızda melez algoritmanın ABC algoritmasına göre daha az enerji ile yolunu tamamladığı görüldü. Gerçek zamanlı problemlerde RRT-ABC melez algoritmasının kullanımı daha efektif sonuçlar elde etmemize olanak sağlar. Gelecekteki çalışmalarda, dinamik engellere çarpmaktan kaçınabilen daha az enerji ile çalışan algoritmalar geliştirmek bu problemin

iyileştirilmesi açısından yeni dallanmalar ile daha yumuşak geçişli daha optimum yollar bulmak ana hedefler olabilir.

TEŞEKKÜR (ACKNOWLEDGMENTS)

Katkılarından dolayı Sivas Cumhuriyet Üniversitesi ROBOLAB ekibine teşekkürlerimizi sunarız.

KAYNAKLAR (REFERENCES)

- [1] Klancar, G., Zdesar, A., Blazic, S., & Skrjanc, I. Wheeled mobile robotics: from fundamentals towards autonomous systems. Butterworth-Heinemann, 2017.
- [2] Montiel, O., Orozco-Rosas, U., & Sepúlveda, R., Path planning for mobile robots using Bacterial Potential Field for avoiding static and dynamic obstacles. *Expert Systems with Applications*, 42(12), 5177-5191, 2015.
- [3] Mac, T. T., Copot, C., Tran, D. T., & De Keyser, R., Heuristic approaches in robot path planning: A survey. *Robotics and Autonomous Systems*, 86, 13-28, 2016.
- [4] Rosell, J., & Iniguez, P., Path planning using harmonic functions and probabilistic cell decomposition. In *Proceedings of the 2005 IEEE international conference on robotics and automation*. IEEE. pp. 1803-1808. Apr., 2005.
- [5] Šeda, M., Roadmap methods vs. cell decomposition in robot motion planning. In *Proceedings of the 6th WSEAS International Conference on Signal Processing, Robotics and Automation*. World Scientific and Engineering Academy and Society (WSEAS). pp. 127-132. Feb., 2017.
- [6] Cosío, F. A., & Castañeda, M. P., Autonomous robot navigation using adaptive potential fields. *Mathematical and computer modelling*, 40(9), 1141-1156, 2004.
- [7] Yin, L., Yin, Y., & Lin, C. J., A new potential field method for mobile robot path planning in the dynamic environments. *Asian Journal of Control*, 11(2), 214-225, 2009.
- [8] Zhang, Q., Yue, S. G., Yin, Q. J., & Zha, Y. B., Dynamic obstacle-avoiding path planning for robots based on modified potential field method. In *International Conference on Intelligent Computing*. Springer Berlin Heidelberg, pp. 332-342, Jul., 2013.
- [9] Singh, N. N., Chatterjee, A., Chatterjee, A., & Rakshit, A., A two-layered subgoal based mobile robot navigation algorithm with vision system and IR sensors. *Measurement*, 44(4), 620-641, 2011.
- [10] Liu, H., Wan, W., & Zha, H. A dynamic subgoal path planner for unpredictable environments. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*. IEEE. pp. 994-1001. May., 2010.
- [11] Candido, S., Kim, Y. T., & Hutchinson, S., An improved hierarchical motion planner for humanoid robots. In *Humanoids 2008-8th IEEE-RAS International Conference on Humanoid Robots*. IEEE. pp. 654-661. Dec., 2008.
- [12] Lee, J., Kwon, O., Zhang, L., & Yoon, S. E., A selective retraction-based RRT planner for various environments. *IEEE Transactions on Robotics*, 30(4), 1002-1011, 2014.
- [13] Hidalgo-Paniagua, A., Vega-Rodríguez, M. A., & Ferruz, J., Applying the MOVNS (multi-objective variable neighborhood search) algorithm to solve the path planning problem in mobile robotics. *Expert Systems with Applications*, 58, 20-35. 2016.
- [14] Dezfoulian, S. H., Wu, D., & Ahmad, I. S., A generalized neural network approach to mobile robot navigation and obstacle avoidance. In *Intelligent Autonomous Systems 12*. Springer Berlin Heidelberg, pp. 25-42, 2013.
- [15] Singh, M. K., & Parhi, D. R., Path optimisation of a mobile robot using an artificial neural network controller. *International Journal of Systems Science*, 42(1), 107-120, 2011.

- [16] Al-Sagban, M., & Dhaouadi, R., Neural-based navigation of a differential-drive mobile robot. In Control Automation Robotics & Vision (ICARCV), 2012 12th International Conference on IEEE. pp. 353-358, Dec., 2012.
- [17] Chang, H., & Jin, T., Command Fusion Based Fuzzy Controller Design for Moving Obstacle Avoidance of Mobile Robot. In Future Information Communication Technology and Applications. Springer Netherlands, pp. 905-913, 2013.
- [18] Abdessemed, F., Faisal, M., Emmadeddine, M., Hedjar, R., Al-Mutib, K., Alsulaiman, M., & Mathkour, H., A hierarchical fuzzy control design for indoor mobile robot. International Journal of Advanced Robotic Systems, 11, 2014.
- [19] Morales, N., Toledo, J., & Acosta, L., Path planning using a Multiclass Support Vector Machine. Applied Soft Computing, 43, 498-50, 2016.
- [20] Roberge, V., Tarbouchi, M., & Labonté, G., Comparison of parallel genetic algorithm and particle swarm optimization for real-time UAV path planning. IEEE Transactions on Industrial Informatics, 9(1), 132-141, 2013.
- [21] Alajlan, M., Koubâa, A., Châari, I., Bennaceur, H., & Ammar, A., Global path planning for mobile robots in large-scale grid environments using genetic algorithms. In Individual and Collective Behaviors in Robotics (ICBR), 2013 International Conference on IEEE, pp. 1-8, Dec., 2013.
- [22] Oleiwi, B. K., Al-Jarrah, R., Roth, H., & Kazem, B. I., Multi Objective Optimization of Trajectory Planning of Non-holonomic Mobile Robot in Dynamic Environment Using Enhanced GA by Fuzzy Motion Control and A. In International Conference on Neural Networks and Artificial Intelligence. Springer International Publishing, pp. 34-49, Jun., 2014.
- [23] Karami, A. H., & Hasanzadeh, M., An adaptive genetic algorithm for robot motion planning in 2D complex environments. Computers & Electrical Engineering, 43, 317-329, 2015.
- [24] Wang, G., Guo, L., Duan, H., Liu, L., & Wang, H., A bat algorithm with mutation for UCAV path planning. The Scientific World Journal, 2012.
- [25] Das, P. K., Behera, H. S., & Panigrahi, B. K., A hybridization of an improved particle swarm optimization and gravitational search algorithm for multi-robot path planning. Swarm and Evolutionary Computation, 28, 14-28, 2016.
- [26] Zhang, Y., Gong, D. W., & Zhang, J. H., Robot path planning in uncertain environment using multi-objective particle swarm optimization. Neurocomputing, 103, 172-185, 2013.
- [27] Chen, X., Kong, Y., Fang, X., & Wu, Q., A fast two-stage ACO algorithm for robotic path planning. Neural Computing and Applications, 22(2), 313-319, 2013.
- [28] Karaboga, D., & Basturk, B., A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. Journal of global optimization, 39(3), 459-471, 2007.
- [29] Liu, S., & Sun, D., Minimizing Energy Consumption of Wheeled Mobile Robots via Optimal Motion Planning. IEEE/ASME Transactions on Mechatronics, 19(2), 401-411. doi:10.1109/tmech.2013.2241777, 2014.
- [30] Martin, S. R., Wright, S. E., & Sheppard, J. W., Offline and online evolutionary bi-directional RRT algorithms for efficient re-planning in dynamic environments. In Automation Science and Engineering, 2007. CASE 2007. IEEE International Conference on .IEEE, pp. 1131-1136, Sept., 2007.
- [31] Zhao, D., & Yi, J., Robot planning with artificial potential field guided ant colony optimization algorithm. In International Conference on Natural Computation .Springer, Berlin, Heidelberg, pp. 222-231, Sept., 2006.
- [32] Santiago, R. M. C., De Ocampo, A. L., Ubando, A. T., Bandala, A. A., & Dadios, E. P. ,Path planning for mobile robots using genetic algorithm and probabilistic roadmap. In Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM), 2017 IEEE 9th International Conference on .IEEE, pp. 1-5, Dec., 2017.

- [33] Chen, Y., Su, F., & Shen, L. C., Improved ant colony algorithm base on PRM for UAV route planning. *Journal of System Simulation*, 21(6), 1658-1666, 2009.
- [34] Masehian, E., & Sedighizadeh, D., Multi-objective PSO-and NPSO-based algorithms for robot path planning. *Advances in electrical and computer engineering*, 10(4), 69-76, 2010.
- [35] LaValle, S. M., Rapidly-exploring random trees: A new tool for path planning. TR 98-11, Computer Science Dept., Iowa State Univ. , Oct., 1998.
- [36] Kuffner, James J., and Steven M. LaValle. "RRT-connect: An efficient approach to single-query path planning." *Robotics and Automation*, 2000. Proceedings. ICRA'00. IEEE International Conference on. Vol. 2. IEEE, 2000.
- [37] Ardiyanto, J. Miura, Real-time navigation using randomized kinodynamic planning with arrival time field, *Robot. Auton. Syst.* 60 (2012) 1579–1591.
- [38] Karaboğa, D.,Yapay Zeka Optimizasyon Algoritmaları, 2014.
- [39] Karaboga, D., & Basturk, B., Artificial bee colony (ABC) optimization algorithm for solving constrained optimization problems. In *International fuzzy systems association world congress* (pp. 789-798). Springer, Berlin, Heidelberg, Jun., 2007.
- [40] UZLU E . Türkiye için gri kurt optimizasyon algoritması ile yapay sinir ağlarını kullanarak enerji tüketiminin tahmini. *Gazi Üniversitesi Fen Bilimleri Dergisi Part C: Tasarım ve Teknoloji*. 2019; 7(2): 262-245.
- [41] MathWorks internet sayfası.<https://www.mathworks.com/matlabcentral/fileexchange/55177-robot-path-planning?focused=5915988&tab=function> (2016).