



SÜRÜ ROBOTLARI İÇİN İŞ BİRLİĞİNE DAYALI YOL PLANLAMA VE ENGELDEN KAÇINMA ALGORİTMALARININ KARŞILAŞTIRMALI ANALİZİ

Müsemma ALTINDAŞ ÖZMEN^{1*}, Levent GÖKREM¹

¹Tokat Gaziosmanpaşa Üniversitesi, Mühendislik ve Mimarlık Fakültesi, Mekatronik Mühendisliği Bölümü, Tokat, Türkiye

Anahtar Kelimeler	Öz
<i>Engelden Kaçma, Mobil Robotlar, Sürü Robotiği, Yol Planlama, Yol Takip Etme.</i>	Bu çalışmada, sürü robotlarının koordineli hareketi için geliştirilen üç farklı yol planlama ve engelden kaçınma algoritması (VFH-Pure Pursuit, RRT-Pure Pursuit ve PRM-Pure Pursuit) karşılaştırmalı olarak analiz edilmiştir. Önerilen yöntemlerde her bir sürü robotunun bağımsız olarak çevresini algılaması, engellerden kaçınması ve belirlenen hedefe organize şekilde ulaşması hedeflenmiştir. Deneysel çalışmalar, 50x50 boyutlarında tanımlanmış üç farklı ortamda; 3, 5 ve 7 robot ile, ileriye bakma mesafesi (l_d) 0.5 olarak belirlenerek gerçekleştirilmiştir. VFH tabanlı yöntemde, robotlar çevresel koşullara anlık tepki verirken, RRT ve PRM tabanlı algoritmelerde başlangıç ve hedef konumlar arasında önceden planlanmış çarpışmasız yollar kullanılmıştır. Simülasyon sonuçları; robot sayısı arttıkça tamamlanma süresinin uzadığını, ancak PRM algoritmasının daha kısa mesafeli ve optimize yollar sunduğunu göstermiştir. Özellikle PRM-Pure Pursuit yöntemi, en düşük ortalama yol mesafesiyle en verimli performansı sergilemiştir. Elde edilen bulgular, sürü robotları için görev bazlı algoritma seçimlerinin başarımı doğrudan etkilediğini ve planlı yapıdaki algoritmaların karmaşık ortamlarda daha etkili sonuçlar verdiğini ortaya koymaktadır.

A COMPARATIVE ANALYSIS OF COLLABORATIVE PATH PLANNING AND OBSTACLE AVOIDANCE ALGORITHMS FOR SWARM ROBOTS

Keywords	Abstract
<i>Mobile Robots, Obstacle Avoidance, Path Planning, Path Tracking, Swarm Robotics,</i>	In this study, three different path planning and obstacle avoidance algorithms (VFH-Pure Pursuit, RRT-Pure Pursuit, and PRM-Pure Pursuit) developed for the coordinated movement of swarm robots are comparatively analyzed. The proposed methods aim for each swarm robot to independently perceive its environment, avoid obstacles, and reach a predefined target in an organized manner. Experimental studies were conducted in three different environments, each defined as 50x50 units in size, using 3, 5, and 7 robots, with a look-ahead distance (l_d) set to 0.5. In the VFH-based approach, robots react to environmental conditions in real time, while in RRT and PRM-based algorithms, pre-planned collision-free paths are used between start and goal positions. Simulation results demonstrate that as the number of robots increases, the completion time also increases; however, the PRM algorithm offers shorter and more optimized paths. Notably, the PRM-Pure Pursuit method achieved the most efficient performance with the lowest average travel distance. The findings indicate that the choice of task-specific algorithms significantly affects the performance of swarm robots and that structured, pre-planned algorithms yield more effective results in complex environments.

Alıntı / Cite

Altındaş Özmen, M., Gökrem, L., (2025). Sürü Robotları İçin İş Birliğine Dayalı Yol Planlama ve Engelden Kaçınma Algoritmalarının Karşılaştırmalı Analizi, Mühendislik Bilimleri ve Tasarım Dergisi, 13(3), 777-790.

Yazar Kimliği / Author ID (ORCID Number)

M. Altındaş Özmen 0000-0001-7536-1079

L. Gökrem, 0000-0003-2101-5378

Makale Süreci / Article Process

Başvuru Tarihi / Submission Date 09.01.2025

Revizyon Tarihi / Revision Date 26.06.2025

Kabul Tarihi / Accepted Date 11.08.2025

Yayın Tarihi / Published Date 30.09.2025

* İlgili yazar / Corresponding author: musemmaaltindas@gmail.com, +90-542-468-9993

A COMPARATIVE ANALYSIS OF COLLABORATIVE PATH PLANNING AND OBSTACLE AVOIDANCE ALGORITHMS FOR SWARM ROBOTS

Müsemma Altındaş Özmen^{1†}, Levent Gökrem¹

¹Tokat Gaziosmanpaşa Üniversitesi, Mühendislik ve Mimarlık Fakültesi, Mekatronik Mühendisliği Bölümü, Tokat, Türkiye

Highlights

- Using swarm robots in a swarm has several advantages over using a single, complex robot.
- Swarm robot should be able to sense environmental conditions, move appropriately and avoid obstacles.
- RRT, PRM, VFH algorithms were used for the path planning and obstacle avoidance problem.
- The advantages of swarm robots in terms of self-coordination, fault tolerance, flexibility and scalability are examined.

Purpose and Scope

Inspired by living creatures in nature, in this study, swarm robot path planning and obstacle avoidance approaches were investigated. 3 different path planning algorithms, 1 path tracking algorithm and 3 different collaborative algorithms were developed and it was aimed to compare these algorithms.

Design/methodology/approach

RRT (Rapidly Recovered Random Tree Algorithm), PRM (Probabilistic Roadmap Algorithm), VFH (Vector Field Histogram) algorithms are used for the path planning and obstacle avoidance problem and Pure Pursuit algorithms are used for path following. An approach is proposed that determines the behavior of each swarm robot for the collaborative path planning and obstacle avoidance problem.

Collaborative algorithms are as follows:

1. VFH- Pure Pursuit Based Collaborative Obstacle Avoidance and Path Planning for Swarm Robots
2. Pure Pursuit-RRT Collaborative Obstacle Avoidance and Path Planning for Swarm Robots
3. PRM-Pure Pursuit Based Collaborative Obstacle Avoidance and Path Planning for Swarm Robots

In the developed VFH-Pure Pursuit based cooperation algorithm, the swarm robots are guided by the VFH method in the direction where there are no obstacles or robots to avoid collisions with obstacles and robots. When obstacles and robots are not encountered, it follows the predetermined path with Pure Pursuit. In the RRT-Pure Pursuit-based cooperative algorithm and PRM-Pure Pursuit-based cooperative algorithms, an unobstructed path is created between the predetermined start and goal location in the environments. It is aimed to follow the generated path with the Pure Pursuit Algorithm.

Findings

In the developed algorithms, each swarm robot exhibits swarm behavior in an organized manner depending on the viewing distance to the path to be followed in environments 1. The results obtained from systematic experiments show that swarm robots follow a flexible and scalable path to reach the goal. In the systematic experiments, the swarm robots exhibit obstacle avoidance behavior and reach the target location even if the number of robots changes. Depending on the increase in the number of robots, the time it takes for the swarm robots to reach the target location from the initial location varied. In the applied systematic experiments, Pure Pursuit -PRM algorithm for 3 robots reached the target position in the least time (number of iterations). The longest time (number of iterations) was realized in the Pure Pursuit - VFH algorithm with 7 robots. According to the experimental results obtained for each environments and each algorithm for 3, 5 and 7 swarm robots, respectively, the time to reach the target position from the initial position has increased.

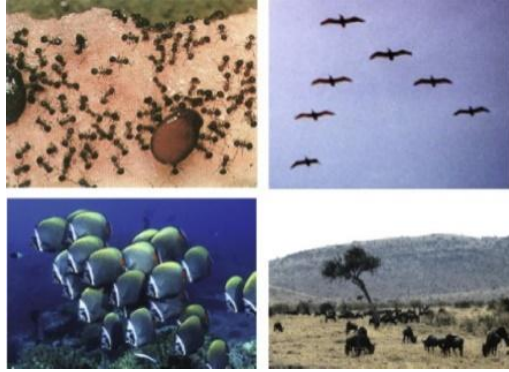
Originality

In this paper, the path planning and obstacle avoidance behaviors of swarm robots are used together. Comparisons are made with 3 different newly developed algorithms, environments with the same environmental conditions and the same number of robots.

[†] Corresponding author: musemmaaltindas@gmail.com, +90-542-468-9993

1. Giriş (Introduction)

Robotlar, gelişen teknolojinin hızıyla birlikte hayatımızın her evresinde yer almaktadır. Savunma sanayi, üretim, sağlık sektörü ve arama-kurtarma çalışmalarında robotlar, iş gücünü artıran, insan hayatını kolaylaştıran ve tehlikeli görevleri üstlenen önemli araçlar haline gelmiştir. Fakat geleneksel robotlar, büyük ve karmaşık fiziksel koşulları özellikle dar alanlarda veya zorlu çevre koşullarında etkili olamamaktadır. Bu durum, daha küçük, basit ve daha işlevsel sürü (swarm) robot sistemlerini ön plana çıkarmıştır. Sürü robot sistemlerinde, hedefe ulaşmak için karmaşık yapıları yerine, basit özelliklere sahip çok sayıda robotun görevlendirilmesi amaçlanmaktadır. Bu yaklaşım, böcekler, kuşlar ve arılar gibi canlıların sergilediği sürü davranışlarından ilham almaktadır (Mısır, O. ve Gökrem, L. 2020).



Şekil 1. Doğada bulunan canlı sürüler (Examples of biological swarms found in nature) (Zheng, Y. T. Z. 2013)

Sürü robotlarının uygulama alanları son yıllarda oldukça önem kazanmıştır. Özellikle tehlikeli ya da erişilemeyen alanlar için tercih edilmektedir. Sürü robotlarının birlikte kullanılması, tek ve karmaşık bir robotun kullanımına göre avantajlar sunmaktadır. Bu avantajlar şunlardır:

- Kendi kendini koordine edebilme özelliği
- Hata toleransı
- Sağlamlık
- Esneklik
- Ölçeklenebilirlik

Sürü robotlarının hata toleransı özelliği, sürüdeki bir veya birden fazla robot arızalansa bile görevin gerçekleştirilebilmesini sağlar (Çayırpınar, Ö.). Esneklik, sürünün değişen görev ve çevre şartlarına uyum sağlayabilme yeteneğidir. Sağlamlık ise, sürünün tamamı aktif çalışmasa bile görevin tamamlanabilmesini ifade eder (Mısır, O. ve Gökrem, L. 2021b).

Sürü robotiğinde ele alınan problemler genel olarak iki gruba ayrılmaktadır. İlk grup; toplanma, konumlanma, kendi kendine organize olma ve kapsam alanı gibi davranışları içerirken, ikinci grup; çevresel etkenlere bağlı hedef arama, yiyecek bulma ve arama-kurtarma görevlerine dayanmaktadır (Stormont, D.P. 2005; Zheng, Y. T. Z. 2013).

Sürü robotlarının basit ve etkili şekilde çalışabilmesi için birbirleriyle iletişim kurmaları ve çevre koşullarına göre hareket etmeleri gerekmektedir. Bu etkileşim, robotların haberleşme özellikleri sayesinde sağlanmakta; çevredeki engellerden kaçınmak için ise çeşitli algılama algoritmaları kullanılmaktadır (Coppola vd., 2019).

Mobil robotlar, çevresel koşulları algılayarak bu koşullara uygun hareket etmeli ve engellerden kaçınmalıdır. Bu problemi çözmek amacıyla birçok çalışma yapılmıştır. İlk çalışmalar, 1969 yılında Stanford Üniversitesi tarafından geliştirilen ve hareket planlaması yapabilen "Shakey" isimli mobil robotla başlamıştır (Yaşar, E. 2020). 1979 yılında ise Lozano ve Wesley, mobil robotlara güvenli yol planlayabilmek amacıyla çarpışmadan kaçınma algoritmalarına öncülük etmiştir (Lozano-Pérez, T. ve Wesley, M. A. 1979).

Bugüne kadar çok sayıda algoritma geliştirilmiştir. Bu algoritmalar genel olarak sezgisel (heuristic) ve klasik (conventional) algoritmalar olmak üzere iki gruba ayrılır. Sezgisel algoritmalara genetik algoritmalar, ACO (Karınca Kolonisi Optimizasyonu), PSO (Parçacık Sürü Optimizasyonu) gibi algoritmalar örnek verilebilirken; klasik algoritmalar arasında potansiyel alanlar (Potential Fields), PRM (Probabilistic Roadmap), RRT (Rapidly Exploring Random Tree) ve bug algoritmaları yer almaktadır. Klasik algoritmaların avantajı kesin sonuçlar sunmalarıdır, ancak bu algoritmalar genellikle daha yavaş çalışmaktadır.

Alarabi (2022) ve arkadaşları, PRM tabanlı bir yaklaşım geliştirerek engellerden kaçınmak ve bilinen bir ortamda hedefe ulaşmak amacıyla robotlar için yol planlaması gerçekleştirmiştir. Bu yaklaşım, artan karmaşıklıkta simülasyon ortamlarında test edilmiş ve genetik algoritmalarla karşılaştırılmıştır. Sonuçlar, PRM'nin mesafe ve dönüş bakımından farklı ortamlar için uygun yollar üretebildiğini göstermektedir.

Yijing Li (2021) ise hareketli engellerin bulunduğu bir ortam için RRT tabanlı bir yol planlama algoritması önermiştir. Bu çalışmada hedef veya engel bilgisi önceden tanımlanmamıştır. Geliştirilen gerçek zamanlı algoritma, holonomik olmayan engellere dayalı olarak yeni yol noktaları belirlemede ve yol yeniden planlamasını minimuma indirerek hedefe yönlendirme sağlamaktadır.

Pure Pursuit algoritmasıyla ilgili olarak Huang Tian (2020) ve arkadaşları, yol planlama hatalarını azaltmak amacıyla bir çalışma yapmıştır. Bu çalışmada ileri bakma mesafesi robotun hızı ve yol bilgisine göre hesaplanarak PID denetleyici ile birleştirilmiştir. Böylece daha kararlı yönlendirme açısı elde edilerek yol hatası azaltılmıştır (Huang, Y., Tian, Z., Jiang, Q. ve Xu, J. 2020).

VFH (Vektör Alan Histogramı) algoritması, mobil robotların bulunduğu ortamı temsil etmek ve hareket yönünü belirlemek için kullanılan bir yöntemdir. Her bir yöndeki engel yoğunluğu hesaplanarak, düşük yoğunluklu yönler tercih edilmektedir (Mısır, O., Çelik, M. ve Gökrem, L. 2022). Bu algoritma robotun boyutunu ve dönüş yarıçapını da dikkate alarak, engellerden kaçınıp hedef yönünü takip edebileceği bir yön tayin eder (Borenstein, I. U. J., 1998).

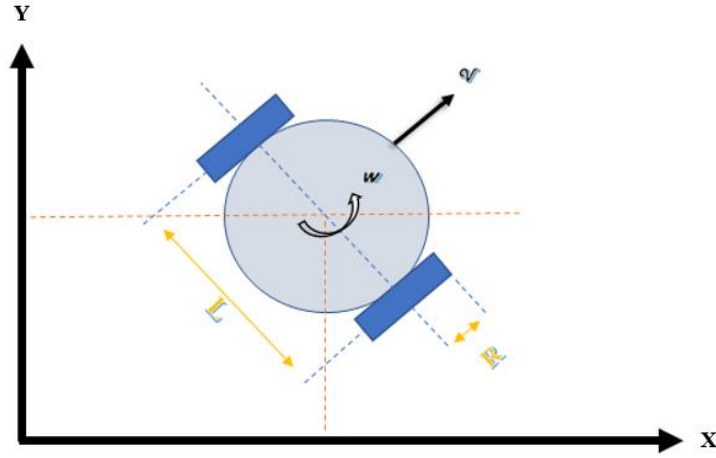
Bu çalışma, sürü robotlarının dinamik ve karmaşık ortamlarda hem yol planlama hem de engelden kaçınma becerilerini bir arada gerçekleştirmesini sağlayacak bütünlük bir kontrol yaklaşımı sunmayı amaçlamaktadır. Seçilen algoritmaların her biri, bu amaca yönelik olarak literatürde önerilen güçlü yönleri nedeniyle tercih edilmiştir. Özellikle Pure Pursuit algoritması, gerçek zamanlı yol izleme uygulamaları için basitliği ve etkinliği ile öne çıkarken; VFH algoritması, yerel çevre bilgisine dayalı esnek engel kaçınma yeteneği sağlamaktadır. RRT ve PRM gibi örnekleme tabanlı küresel yol planlama algoritmaları ise başlangıç ve hedef konumlar arasında geçerli yolların oluşturulmasında yüksek başarı sağlamaktadır. Literatürde bu tür algoritmaların birlikte kullanımı, çoklu robot sistemlerinde hem küresel hem yerel düzeyde güvenli, ölçeklenebilir ve esnek bir hareket stratejisi geliştirmek adına sıklıkla önerilmektedir (LaValle, 2006; Ulrich & Borenstein, 1998; Park & Yoon, 2018a). Bu gerekçeler doğrultusunda çalışmada VFH, Pure Pursuit, RRT ve PRM algoritmalarının birlikte ele alınmasıyla sürü robotları için etkili bir planlama ve kontrol sistemi geliştirilmiştir.

2. Materyal ve Yöntem (Material and Method)

Bu bölümde, sürü robotlarının iş birliği kapsamında yol planlama ve engelden kaçınma algoritmaları incelenmektedir. Yol planlama ve engelden kaçınma problemi için RRT (Hızla Keşfedilen Rastgele Ağaç Algoritması), PRM (Olasılıksal Yol Haritası Algoritması), VFH (Vektör Alan Histogram) algoritmaları ve yol takibi için Pure Pursuit algoritmaları kullanılmıştır. İş birliğine dayalı yol planlama ve engellerden kaçınma problemi için, her bir sürü robotunun davranışını belirleyen bir yaklaşım önerilmektedir. İlk olarak, önerilen algoritma ve yöntemlerin uygulanabileceği mobil robot kinematiği açıklanmaktadır.

2.1. Robot Kinematik Modeli (Robot Kinematic Model)

Bu çalışmada, mobil robot modellerinden holonomik olmayan, iki tekerlekli diferansiyel tahrikli mobil robot referans alınmıştır. Şekil 2'de mobil robotun kinematik modeli gösterilmektedir. Şekil 2'de görüldüğü gibi, $\frac{L}{2}$ yarıçapında dairesel bir yapıya sahiptir. Robotun merkezinde $\frac{L}{2}$ uzağına yerleştirilmiş, R yarıçapında paralel iki tekerlek bulunmaktadır. L ifadesi, sürü robotunun iki tekerlek arasındaki mesafeyi belirtirken, ω açısal hızı ifade etmektedir.



Şekil 2. İki tekerlekli diferansiyel tahrikli mobil robot modeli (Two-wheeled differential drive mobile robot model)

Sürü robotlarının özellikleri Tablo 1’de belirtilmiştir.

Tablo 1. Sürü robotu özellikleri (Swarm robot characteristics)

Özellikleri	Ölçüm Aralıkları
Robot Modeli	DifferentialDrive
Robot Algılama Açısı	360°
Robot Algılama Uzaklığı	10 birim
Engel Algılama Uzaklığı	10 birim
Engel Algılama Açısı	360°

Sürü robotu, Eşitlik 1’de koordinatları ile temsil edilirken, Eşitlik 2’de α ifadesi kullanılarak robot kinematik modeli gösterilmiştir.

$$\alpha = [x, y, \theta]^K \quad (1)$$

$$\dot{\alpha} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos \theta \\ \sin \theta \\ 0 \end{bmatrix} v + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \omega \quad (2)$$

Şekil 2’de görüldüğü gibi, v robotun doğrusal hızı, ω ise açısal hızı olarak tanımlanmaktadır. Sürü robotu kinematik modelinde, iki tekerleğin açısal ve doğrusal hızları arasında bir ilişki vardır. Tekerleklerin açısal hızları sağ (ω_R)-sol (ω_L) tekerlek olarak ayrılır. Bu hızlar aşağıdaki eşitliklerle ifade edilir:

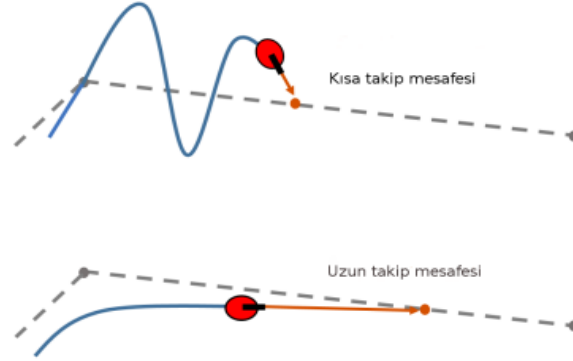
$$v = \frac{R(\omega_L + \omega_R)}{2} \quad \omega = \frac{R(\omega_L - \omega_R)}{L} \quad (3)$$

Sürü robotları, ω_R ve ω_L açısal hızlarına bağlı olarak yönünü belirler:

$$\omega_R = \frac{v + (\frac{L}{2})\omega}{R} \quad \omega_L = \frac{v - (\frac{L}{2})\omega}{R} \quad (4)$$

2.2. Pure Pursuit Algoritması (Pure Pursuit Algorithm)

Pure Pursuit, bir yol takip algoritmasıdır. Bu algoritma, uzakta belirlenen bir noktaya odaklanarak o noktaya hareket eder ve yörüngeyi takip eder (Mısır, O., Çelik, M., ve Gökrem, L. 2022). Pure Pursuit, sabit bir mesafe olan ileri bakma mesafesi (l_d) ve robotun direksiyon açısını kullanarak belirlenen hedef noktaya ilerler. Algoritma, robotun mevcut konumuna bağlı olarak yoldaki ileri bakma noktasını hedef noktasına kadar hareket ettirir. Kısaca, bu algoritma robotun belirlenen yolu takip etmesini sağlar; yani robot sürekli olarak ileriye bakma mesafesini izler (Ahuja, R. K. ve ark., 1990)



Şekil 2. Pure Pursuit ile yol takibi (Path tracking using Pure Pursuit) (Coulter, R 1990)

2.3. Vektör Alan Histogram (VFH) (Vector Field Histogram)

Vektör Alan Histogramı (VFH), mobil robotların bulunduğu ortamı temsil etmek için kullanılır ve bir sonraki hareket yönü, her yönün maliyet fonksiyonuna göre seçilir. Histogram, robotun bakış açısından görülen engel yoğunluğunu gösterir. Mümkün olan her yönde engellerin yoğunluğu sürekli hesaplanır ve robot hareketini engel yoğunluğunun düşük olduğu yöne doğru yapar (Mısır, O., Çelik, M., ve Gökrem, L. 2022). VFH algoritması, robotun engellerden kaçınması ve hedef yönünü takip etmesi için robot boyutu ve dönüş yarıçapını da hesaba katar (Ulrich, I., & Borenstein, J. 1998).

2.4. Olasılıksal Yol Haritası Algoritması (PRM) (Probabilistic Roadmap Algorithm)

PRM algoritması, belirlenen bir haritada engellerden kaçınarak robotun başlangıç konumundan hedef konumuna ulaşması için haritadaki olası yolları ağ grafiği şeklinde oluşturur. PRM yol planlayıcısı iki aşamadan oluşur: öğrenme ve sorgulama. İlk aşamada, harita üzerinde rastgele noktalar oluşturulur ve bu noktaların boş alanda olup olmadığı kontrol edilir. Boş alandaki noktalar ağ grafiğine eklenir. Yeni noktalar, en yakın noktalara bağlanır ve bağlantıların boş alanda olması kontrol edilerek grafiğe eklenir. Bu işlem hedefe ulaşana kadar devam eder. Sorgulama aşamasında, başlangıç ve hedef konumları grafiğe bağlanır ve Dijkstra algoritması kullanılarak en kısa yol bulunur. Rastgele noktaların sayısı arttıkça, PRM algoritmasının yol bulamama olasılığı azalır (Kavraki, 2013).

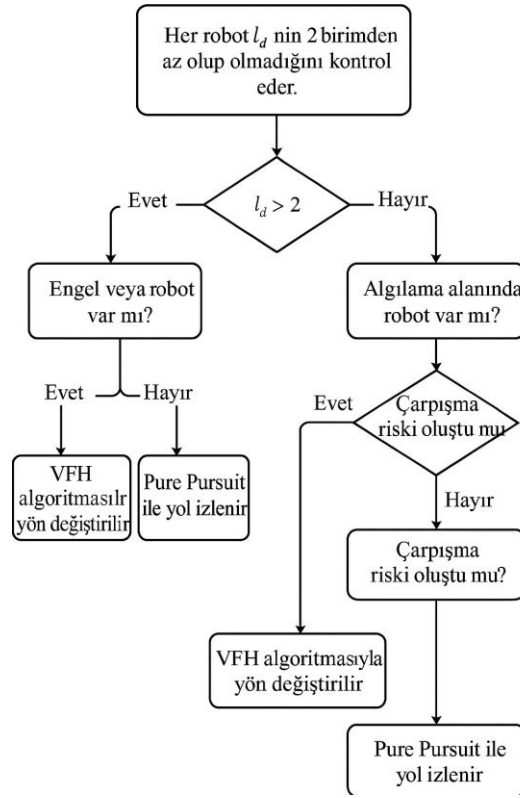
2.5. Hızla Keşfedilen Rastgele Ağaç Algoritması (RRT) (Rapidly Exploring Random Tree Algorithm)

RRT, sürü robotlarının yol planlamasında kullanılan bir diğer algoritmadır. Haritada rastgele örnekler kullanılarak boşluklar verimli şekilde aranır. RRT algoritması, Steven M. LaValle ve James J. Kuffner Jr. tarafından geliştirilmiştir (Nemec, D., Janota, A., Hruboš, M., Gregor, M., ve Pirnik, R. 2017). Algoritma, durum uzayından rastgele örnekler olarak artımlı bir arama ağacı oluşturur. Başlangıç konumu düğüm olarak seçilir ve kendisine en yakın düğümle birleştirilir; böylece iki nokta arasında yol oluşur. Yolun engellerle çarpışmadığı kontrol edilir. Engelsiz ise düğüm ağaca eklenir. Aksi halde yeni düğümler seçilir ve işlem tekrarlanır. Bu süreç, hedefe ulaşılan kadar devam eder (LaValle & Kuffner, 2001).

2.6. Sürü Robotları için VFH- Pure Pursuit Tabanlı İş Birliğine Dayalı Engelden Kaçınma ve Yol Planlama (VFH- Pure Pursuit Based Collaborative Obstacle Avoidance and Path Planning for Swarm Robots)

Pure Pursuit ve VFH kullanan iş birliğine dayalı yol planlama algoritması, sürü robotlarının başlangıç konumundan hedef konuma organize şekilde hareket etmesini sağlar. Önerilen yaklaşımda, sürü robotlarının sayısı değiştikçe önceden belirlenmiş başlangıç konumundan hedef konuma hareket ederler. Robotlar yollarını düzenli takip

ederken, çevredeki engellere ve koşullara bağlı olarak esnek davranışlar sergilerler. Şekil 4'te Pure Pursuit-VFH iş birliğine dayalı engelden kaçınma ve yol planlama akış diyagramı gösterilmektedir.



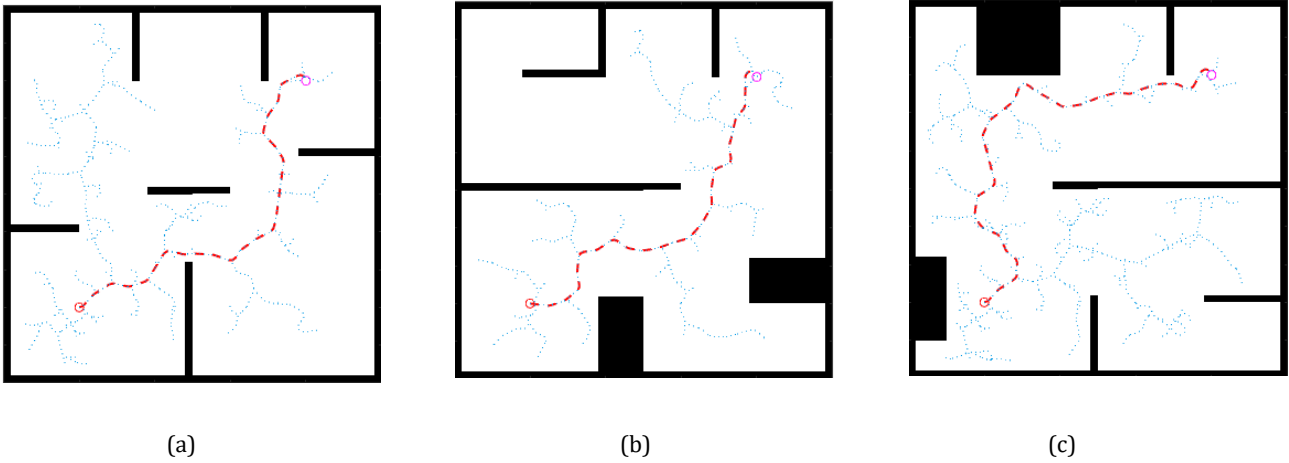
Şekil 3. Pure Pursuit-VFH iş birliğine dayalı engelden kaçınma ve yol planlama akış diyagramı (Flowchart of obstacle avoidance and path planning using Pure Pursuit and VFH)

Geliştirilen algoritmada, sürü robotları l_d değerine bağlı olarak yol izleme kararları alır. Her robot, l_d 'nin 2 birimden az olup olmadığını kontrol eder. Eğer l_d 2 birimden büyükse, çevresinde çarpışma mesafesinde engel veya robot olup olmadığını inceler. Engel veya robotla karşılaşır, engel veya robotun olmadığı yöne VFH yöntemiyle yönlendirilir. Engel veya robot yoksa Pure Pursuit yöntemi ile yolu takip eder. l_d 2 birimden küçükse, algılama alanında robot olup olmadığı kontrol edilir. Robot varsa, bu robotların yönüne yaklaşır. Eğer engelle çarpışma riski oluşursa, VFH ile engelden uzaklaşır; aksi halde Pure Pursuit ile yolu izlemeye devam eder.

2.7. Sürü Robotları İçin Pure Pursuit-RRT İşbirliğine Dayalı Engelden Kaçınma ve Yol Planlama (Pure Pursuit-RRT Collaborative Obstacle Avoidance and Path Planning for Swarm Robots)

Geliştirilen Pure Pursuit-RRT algoritması, başlangıç ve hedef konumları aynı olan üç farklı ortamda test edilmiştir. Algoritmada, başlangıç ve hedef noktaları startPose [10 10 0] ve goalPose [40 40 $-\pi/2$] olarak atanmıştır. Bu noktalar arasında optimal engelsiz yol oluşturulur ve sürü robotları Pure Pursuit yöntemi ile RRT tarafından oluşturulan yolu takip eder.

RRT algoritması, rastgele bir düğüme odaklanır ve kendisine en yakın düğüm ile birleştirir. Oluşan yolun engel ile temas edip etmediği kontrol edilir. Engel yoksa yol ağa eklenir, varsa yeni düğümler seçilerek işlem tekrar edilir. Algoritma, 30000 iterasyon ile yol bulunana kadar devam eder.

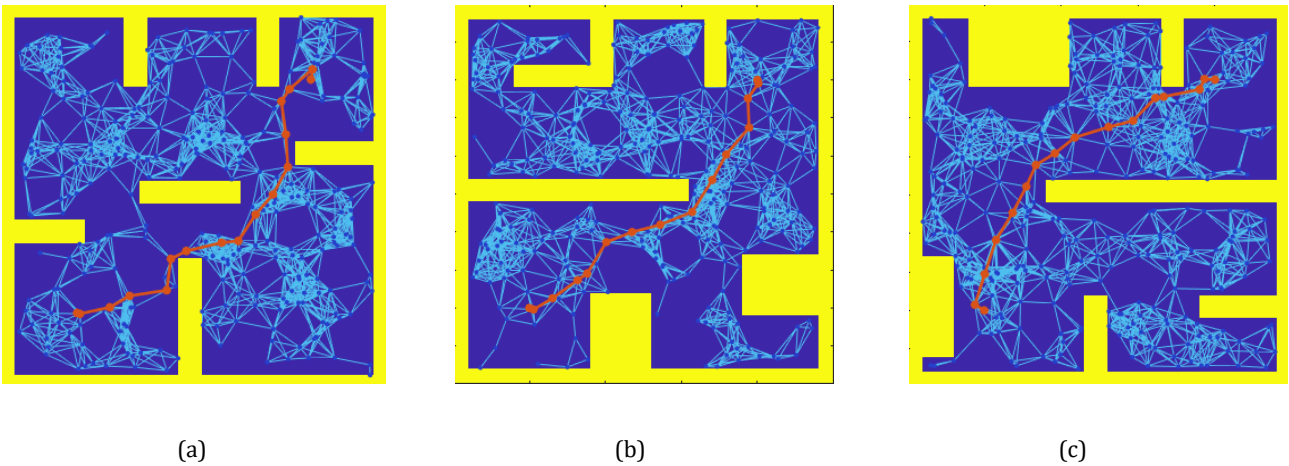


Şekil 4. (a) Ortam 1 , (b) Ortam 2 ve (c) Ortam 3 RRT ile oluşturulan optimal yol sonuçları (Optimal path results generated by RRT in (a) Environment 1, (b) Environment 2, and (c) Environment 3)

2.8. Sürü Robotları için Pure PRM- Pursuit Tabanlı İş Birliğine Dayalı Engelden Kaçınma ve Yol Planlama (PRM-Pure Pursuit Based Collaborative Obstacle Avoidance and Path Planning for Swarm Robots)

Bu çalışmada, Pure Pursuit ve PRM algoritmaları kullanılarak 50x50 boyutlarında üç farklı ortamda yol planlama gerçekleştirilmiştir. Ortam içerisindeki engeller, sürü robotlarının güvenliğini sağlamak amacıyla, robot yarıçapı kadar genişletilmiştir. Genişletilmiş ortamda, rastgele 300 adet düğüm oluşturulmuştur. Her bir düğümün boş alanda yer alıp almadığı kontrol edilerek uygun düğümler bağlantı ağına dahil edilmiştir. Bu düğümler arasında, bağlantı mesafesi olarak 5 birimlik bir "ConnectionDistance" değeri kullanılmıştır. Başlangıç konumu [10 10], hedef konumu ise [40 40] olarak belirlenmiş ve bu iki nokta arasında PRM algoritması ile çarpışmasız bir yol planlanmıştır. Oluşturulan bu yol, sürü robotları tarafından Pure Pursuit algoritması ile takip edilmiştir.

PRM algoritması iki ana aşamadan oluşmaktadır: öğrenme ve sorgulama aşamaları. Öğrenme aşamasında, ortam robot yarıçapı kadar genişletilerek bir güvenlik alanı oluşturulur. Bu genişletilmiş ortamda rastgele 300 düğüm üretilir ve yalnızca boş alanda kalan düğümler grafik ağına dahil edilir. Her düğüm, en yakın komşularıyla 5 birim uzaklığa kadar bağlantı kurar. Bağlantıların da boş alandan geçip geçmediği kontrol edilir ve yalnızca çarpışmasız yollar bağlantı grafiğine eklenir. Sorgulama aşamasında ise başlangıç ve hedef konumları grafiğe bağlanarak, bu iki nokta arasındaki en kısa çarpışmasız yol Dijkstra algoritması kullanılarak belirlenir. Düğüm sayısının artması, yol bulunamama ihtimalini büyük ölçüde azaltmaktadır. Bu yöntemle, sürü robotları hedefe güvenli ve organize bir şekilde ulaşabilmektedir (Park & Yoon, 2018b)



Şekil 5. (a) Ortam 1 , (b) Ortam 2 ve (c) Ortam 3 PRM ile oluşturulan optimal yol sonuçları (Optimal path results generated by PRM in (a) Environment 1, (b) Environment 2, and (c) Environment 3)

3. Sonuçlar ve Tartışmalar (Results and Discussions)

Bu bölümde, geliştirilen üç farklı yol planlama ve engelden kaçınma algoritmasının karşılaştırılabilmesi amacıyla yalnızca Ortam 1'de gerçekleştirilen deneyler dikkate alınmıştır. Simülasyonlar, 3, 5 ve 7 sürü robotu ile ve 0.5 birim ileriye bakma mesafesi (l_d) değeri temel alınarak yürütülmüştür. Deney sonuçları, robot sayısına göre ayrı ayrı sunulmakta ve ileriye bakma mesafesi değişiminin etkileri tartışılmaktadır.

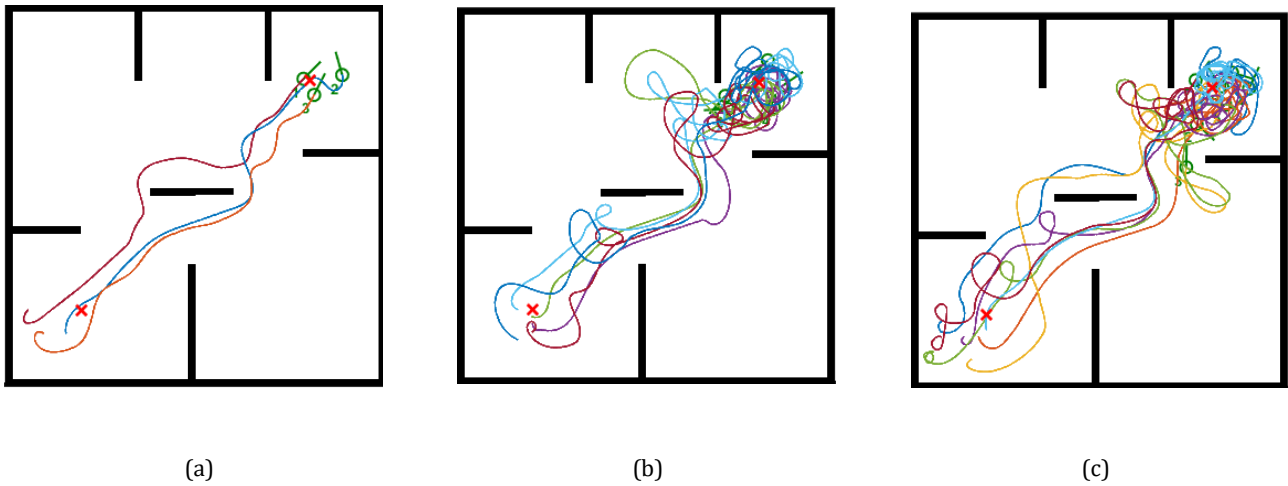
Yol planlama ve engelden kaçınma yaklaşımlarının performansını değerlendirmek amacıyla simülasyon süresi ölçütü kullanılmıştır. Simülasyon süresi, her robot grubunun başlangıç konumundan hedef konuma ulaşması için gereken toplam simülasyon adımı sayısı olarak tanımlanmıştır. Elde edilen sonuçlar; sürü robotlarının ölçeklenebilirliği, esnekliği ve çarpışmasız yol planlaması ile Pure Pursuit algoritmasına dayalı yol takip performansı açısından değerlendirilmiştir.

3.1. VFH- Pure Pursuit Tabanlı İş Birliğine Dayalı Engelden Kaçınma ve Yol Planlama Algoritması İçin Deney Sonuçları (Experiment Results for VFH- Pure Pursuit Based Collaborative Obstacle Avoidance and Path Planning Algorithm)

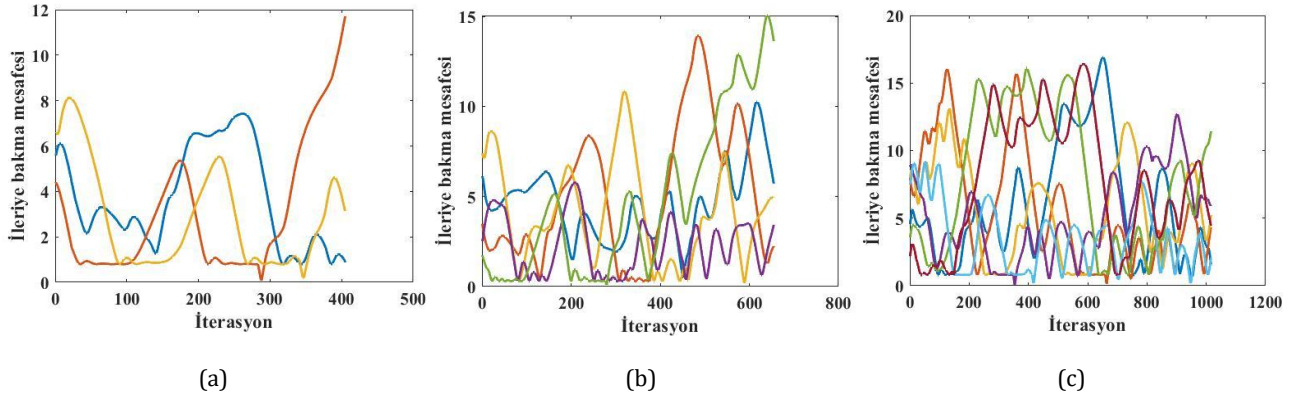
Şekil 7'de, 50x50 boyutlarındaki Ortam 1'de gerçekleştirilen simülasyon deneylerinde 3, 5 ve 7 sürü robotu ile uygulanan VFH-Pure Pursuit tabanlı yol planlama algoritmasının sonuçları sunulmaktadır. Her bir sürü robotu, başlangıç konumundan hedef konuma çarpışmadan ulaşmayı başarmıştır.

Simülasyon sırasında robotların yol takibi sırasında engellerle veya keskin dönüşlerle karşılaşması durumunda, belirlenen yola olan mesafelerinde artış gözlenmiştir. Bu durum, tamamlanma süresini uzatmıştır. En uzun simülasyon süresi, 7 robot ile yapılan ve ileriye bakma mesafesi 0.5 olan deneyde kaydedilmiştir. En kısa sürede tamamlanan deney ise 3 robotlu ve yine 0.5 ileriye bakma mesafesine sahip simülasyondur.

Bu bulgular, robot sayısındaki artışın simülasyon süresini doğrudan artırdığını göstermektedir. Yani, sistemdeki robot sayısı arttıkça çarpışmadan kaçınma ve koordinasyon süreci daha karmaşık hale gelmekte, bu da tamamlanma süresine yansımaktadır.



Şekil 7. VFH-Pure Pursuit algoritması ile Ortam 1'de gerçekleştirilen simülasyonlar:
 (a) 3 sürü robotu, (b) 5 sürü robotu, (c) 7 sürü robotu (Simulations in Environment 1 using the VFH-Pure Pursuit algorithm:
 (a) 3 swarm robots, (b) 5 swarm robots, (c) 7 swarm robots)



Şekil 8. 0.5 birim ileriye bakma mesafesi ile VFH–Pure Pursuit tabanlı iş birliğine dayalı engelden kaçınma ve yol planlama algoritması için simülasyon sonuçları: (a) 3 robot, (b) 5 robot, (c) 7 robot (Simulation results of the cooperative obstacle avoidance and path planning algorithm based on VFH–Pure Pursuit with a 0.5-unit look-ahead distance: (a) 3 robots, (b) 5 robots, (c) 7 robots)

3.2. RRT- Pure Pursuit Tabanlı İş Birliğine Dayalı Engelden Kaçınma ve Yol Planlama Algoritması İçin Deneysel Sonuçları (Experiment Results for RRT- Pure Pursuit Based Collaborative Obstacle Avoidance and Path Planning Algorithm)

Tablo 2’de, RRT-Pure Pursuit tabanlı iş birliğine dayalı engelden kaçınma ve yol planlama simülasyonunun kurulumu gösterilmiştir.

Tablo 2: RRT simülasyon özellikleri (RRT simulation parameters)

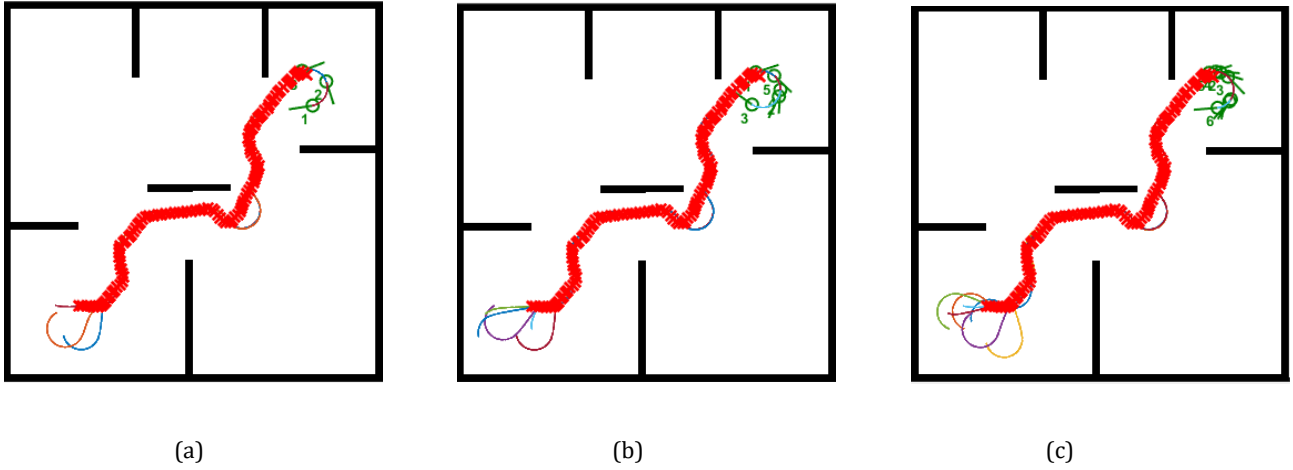
Ortam boyutu	{50x50}
l_d Değeri	{0.5}
Robot sayısı	{3, 5, 7}
Bağlantı Mesafesi	2,5
Minimum Dönüş Yarıçapı	0,75

Tablo 2’de belirtilen robot sayıları (3, 5 ve 7), Pure Pursuit yöntemi için l_d değeri (0.5) ve ortam boyutları sabit tutulmuş; ancak her bir simülasyon farklı ortamlarda gerçekleştirilmiştir. Bağlantı mesafesi 2.5 birim ve minimum dönüş yarıçapı 0.75 birim olarak belirlenmiş ve bu parametrelere göre engelsiz yollar planlanmıştır. VFH-Pure Pursuit algoritmasından farklı olarak, bu çalışmada yol, başlangıç ve hedef konumları arasında RRT algoritması ile önceden planlanarak oluşturulmuştur.

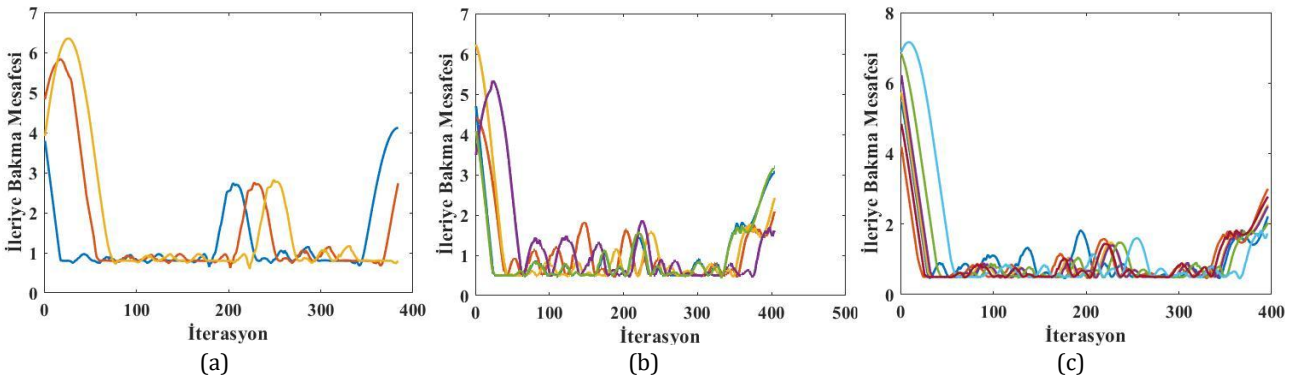
Simülasyonlarda robotlar, 50x50 boyutundaki Ortam 1’e belirli bir başlangıç aralığında rastgele dağıtılmıştır. Robotların üst üste binmesini önlemek için rastgele dağılım işlemi tekrar edilmiştir. Bu sayede her bir sürü robotu, ortam içinde farklı başlangıç konumlarında yer almaktadır. Pure Pursuit-RRT algoritmasında kullanılan robotlar, VFH tabanlı algoritmada kullanılanlarla aynıdır. Şekil 9’da görüldüğü üzere, simülasyon ortamı dört tarafı ve içi engellerle çevrili kare yapıda tasarlanmıştır.

Önerilen yol planlama ve engelden kaçınma algoritması, Ortam 1’de farklı sayıda robot (3, 5 ve 7) ve sabit 0.5 birim l_d değeri ile test edilmiştir. Her bir robot sayısı için elde edilen simülasyon sonuçları ayrı ayrı analiz edilmiştir. İleriye bakma mesafesine göre elde edilen grafikler Şekil 10’da sunulmuştur.

Şekil 9’da gösterilen simülasyonlarda, sürü robotları başlangıç konumlarından ayrılarak RRT algoritmasıyla oluşturulan engelsiz yolu takip etmiş ve hedef konuma başarıyla ulaşmıştır. Şekil 10’da ise farklı robot sayıları için (3, 5 ve 7) elde edilen yineleme (iterasyon) sayıları karşılaştırmalı olarak sunulmuştur.



Şekil 9. RRT-Pure Pursuit algoritması ile Ortam 1’de gerçekleştirilen simülasyonlar: (a) 3 sürü robotu, (b) 5 sürü robotu, (c) 7 sürü robotu (Simulations in Environment 1 using the RRT-Pure Pursuit algorithm: (a) 3 swarm robots, (b) 5 swarm robots, (c) 7 swarm robots)



Şekil 10. 0.5 birim ileriye bakma mesafesi ile RRT-Pure Pursuit tabanlı iş birliğine dayalı engelden kaçınma ve yol planlama algoritması için simülasyon sonuçları: (a) 3 robot, (b) 5 robot, (c) 7 robot (Simulation results of the cooperative obstacle avoidance and path planning algorithm based on RRT-Pure Pursuit with a 0.5-unit look-ahead distance: (a) 3 robots, (b) 5 robots, (c) 7 robots)

Elde edilen bulgulara göre robot sayısı arttıkça, simülasyonun tamamlanma süresi de artmaktadır. Ancak VFH tabanlı algoritmadan farklı olarak, RRT tabanlı yaklaşımda deney süreleri daha kısadır. Bunun temel nedeni, sürü robotlarının harekete geçmeden önce takip edecekleri yolun belirlenmiş ve engelsiz olmasıdır.

3.3. Sürü Robotları için PRM- Pure Pursuit Tabanlı İş Birliğine Dayalı Engelden Kaçınma ve Yol Planlama (Experiment Results for PRM- Pure Pursuit Based Collaborative Obstacle Avoidance and Path Planning Algorithm)

Geliştirilen PRM-Pure Pursuit tabanlı iş birliğine dayalı engelden kaçınma ve yol planlama algoritması için sistematik simülasyon deneyleri gerçekleştirilmiştir. Bu deneyler, Tablo 3’te belirtilen parametrelere göre, 50x50 boyutlarında Ortam 1’de, l_d değeri 0.5 olacak şekilde ve 3, 5 ve 7 robot kullanılarak uygulanmıştır. Amaç; farklı sayıda sürü robotunun, ölçeklenebilir şekilde yol planlama ve engelden kaçınma davranışlarını göstererek, başlangıç konumundan hedefe en kısa yoldan ulaşmasını sağlamaktır.

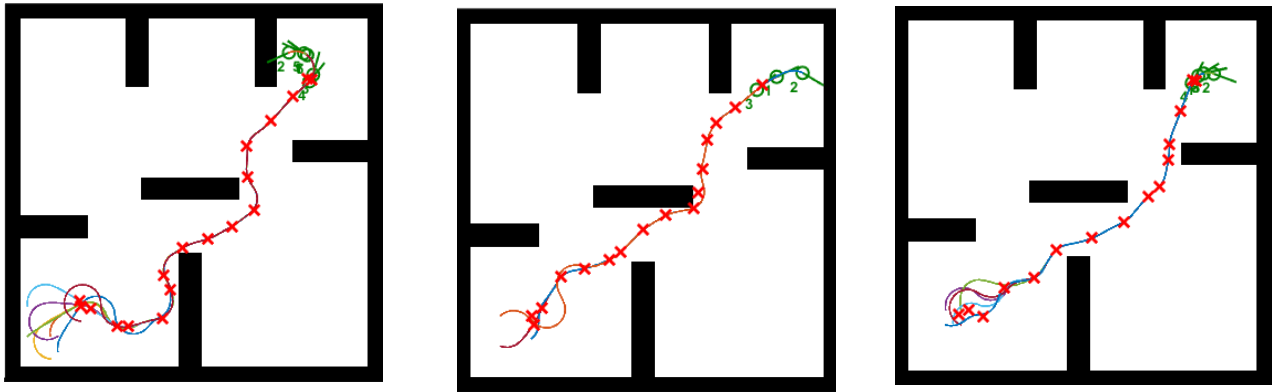
PRM algoritması, ortam içinde rastgele düğümler oluşturarak ve bu düğümler arasında bağlantılar kurarak, başlangıç ve hedef konum arasında engelsiz bir yol planlamaktadır. Robotun bu yolu takip etmesi için Pure Pursuit algoritması kullanılmıştır. Dijkstra algoritması ise çarpışmasız en kısa yolun bulunmasında görev almıştır.

Tablo 3. PRM Simülasyon özellikleri (PRM simulation parameters)

Ortam boyutu	{50x50}
l_d Değeri	{0.5}
Robot sayısı	{3, 5, 7}
Düğüm sayısı	300
Bağlantı mesafesi	5

Tablo 3'te yer alan parametreler doğrultusunda, 50x50 boyutundaki Ortam 1'de 300 rastgele düğüm oluşturulmuş ve bu düğümler arasındaki bağlantılar 5 birim mesafe kuralına göre yapılmıştır. Bu bağlantılar sayesinde, çarpışmasız ve engelsiz yollar planlanmıştır. Simülasyon süresi, başlangıç konumundan hedef konuma ulaşana kadar geçen adım sayısı olarak tanımlanmıştır ve sistemin performansını ölçmede temel kriter olarak kullanılmıştır. Elde edilen sonuçlar, sürü robotlarının ölçeklenebilirliği ve esnekliği açısından değerlendirilmiştir.

VFH-Pure Pursuit algoritmasından farklı olarak, PRM algoritması RRT algoritmasına benzer şekilde yol planlamayı hareket öncesinde gerçekleştirmektedir. Rastgele oluşturulan düğümlerle belirlenen yol, robotlar harekete geçmeden önce tanımlanmakta ve robotlar bu yolu Pure Pursuit algoritmasıyla takip etmektedir.



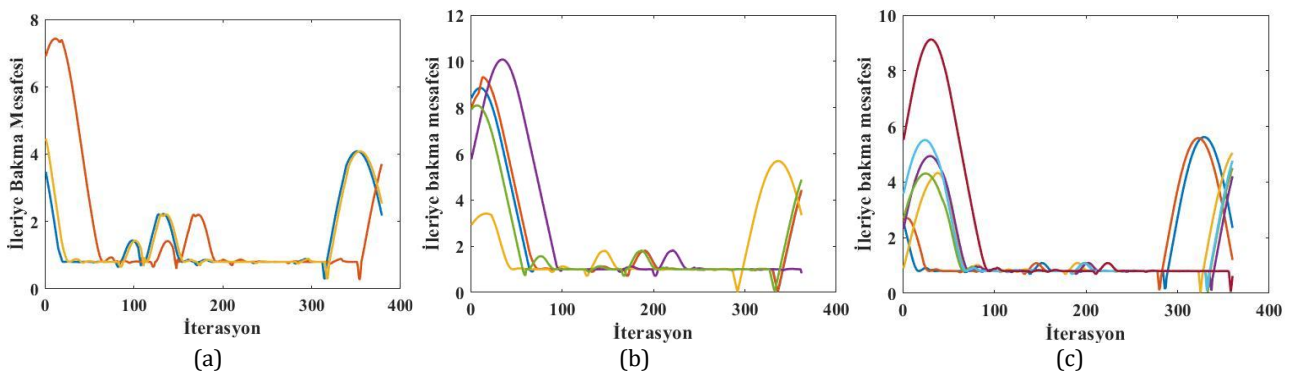
(a)

(b)

(c)

Şekil 11. PRM-Pure Pursuit algoritması ile Ortam 1'de gerçekleştirilen simülasyonlar:

(a) 3 sürü robotu, (b) 5 sürü robotu, (c) 7 sürü robotu (Simulations in Environment 1 using the PRM-Pure Pursuit algorithm: (a) 3 swarm robots, (b) 5 swarm robots, (c) 7 swarm robots)



(a)

(b)

(c)

Şekil 12. 0.5 birim ileriye bakma mesafesi ile PRM-Pure Pursuit algoritması için simülasyon sonuçları:

(a) 3 robot, (b) 5 robot, (c) 7 robot (Simulation results of the PRM-Pure Pursuit algorithm with a 0.5-unit look-ahead distance: (a) 3 robots, (b) 5 robots, (c) 7 robots)

Şekil 11’de, Ortam 1’de rastgele oluşturulan 300 noktaya dayalı olarak planlanan engelsiz yollar ve bu yolları takip eden sürü robotlarının simülasyon sonuçları gösterilmektedir. Yol planlaması, Pure Pursuit–RRT algoritmasındaki gibi, robotların hareket etmeden önce takip edecekleri yolun belirlenmesi esasına dayanmaktadır. Sürü robotları, belirlenen bu yolu l_d 0.5 değerine göre organize bir şekilde takip ederek hedef konuma ulaşmaktadır. Şekil 12’de, 3, 5 ve 7 robotla yapılan deneylerdeki simülasyon süreleri (yineleme sayıları) gösterilmiştir. Grafikte de görüldüğü üzere, robot sayısı arttıkça tamamlanma süresi artış göstermektedir. Bu durum, robotlar arası çakışmaların ve koordinasyon ihtiyacının artmasından kaynaklanmaktadır.

Tablo 4. RRT-PRM Algoritmaları için Katedilen Mesafe (Travel distances of RRT and PRM algorithms)

Ortam 1 için Katedilen Mesafe	
Pure Pursuit-RRT İş birliğine Dayalı Algoritma için Katedilen Mesafe	62 cm
Pure Pursuit-PRM İş birliğine Dayalı Algoritma için Kat edilen Mesafe	57 cm

Tablo 4’te, PRM ve RRT algoritmaları için sürü robotlarının başlangıç konumdan hedef konuma kadar kat ettiği ortalama mesafeler karşılaştırılmıştır. Sonuçlara göre, en kısa mesafe Pure Pursuit–PRM algoritmasında elde edilirken, en uzun mesafe ise Pure Pursuit–RRT algoritmasıyla oluşmuştur. Bu durum, PRM algoritmasının daha kompakt ve optimize yollar üretme eğilimini yansıtmaktadır.

VFH algoritmasında kat edilen mesafe bilgisinin verilmemesi, algoritmanın yapısından kaynaklanmaktadır. VFH, çevredeki engellere göre dinamik yön belirleme yapan reaktif bir yöntemdir. Bu nedenle, VFH algoritmasının uygulandığı senaryolarda robotlar önceden planlanmış sabit bir yol izlemez, engellere göre sürekli yön değiştirir. Bu durum, her simülasyonda farklı rotaların oluşmasına ve tekrarlanabilir mesafe ölçümlerinin standardize edilememesine neden olur. Bu nedenle, kat edilen mesafe bilgisi VFH için doğrudan karşılaştırılabilir veya anlamlı bir metrik oluşturmamaktadır. Dolayısıyla çalışmada, daha anlamlı ve karşılaştırılabilir sonuçların elde edilebildiği algoritmalar olan RRT ve PRM için mesafe analizi yapılmıştır. VFH’nin performansı ise simülasyon süresi (tamamlanma süresi) üzerinden değerlendirilmiştir.

4. Sonuçlar (Results)

Bu çalışmada, sürü robotlarının iş birliğine dayalı olarak engellerden kaçınma ve yol planlama performanslarını artırmak amacıyla üç farklı algoritma (VFH–Pure Pursuit, RRT–Pure Pursuit ve PRM–Pure Pursuit) önerilmiş ve karşılaştırmalı simülasyonlar gerçekleştirilmiştir. Her algoritma, 50x50 boyutlarında tanımlanan sabit bir ortamda; 3, 5 ve 7 sürü robotu kullanılarak ve ileriye bakma mesafesi l_d 0.5 olacak şekilde test edilmiştir. Simülasyonlar, her algoritmanın ölçeklenebilirlik ve esneklik açısından sunduğu performans farklarını ortaya koymuştur.

VFH–Pure Pursuit algoritmasında, robotlar çevredeki engellere dinamik olarak tepki verirken, en uzun simülasyon süresi 7 robotla yapılan deneyde, en kısa süre ise 3 robotla yapılan deneyde elde edilmiştir. Bu durum, robot sayısının artmasıyla birlikte çakışma olasılığının ve buna bağlı olarak yol değişikliklerinin arttığını göstermektedir.

RRT–Pure Pursuit algoritmasında ise, simülasyon başlamadan önce rastgele oluşturulan düğümler aracılığıyla planlanan yollar sayesinde, robotlar önceden belirlenmiş rotaları takip etmiş ve bu nedenle tamamlanma süresi, VFH algoritmasına kıyasla daha kısa olmuştur. Bu yapı, özellikle karmaşık ortamlarda güvenli yol planlamasına olanak sağlamaktadır.

PRM–Pure Pursuit algoritması ile gerçekleştirilen simülasyonlarda ise 300 rastgele düğüm kullanılmış ve robotlar, oluşturulan bağlantı ağı üzerinden Dijkstra algoritması ile belirlenen en kısa yolu izlemiştir. Sonuçlar, PRM tabanlı algoritmanın daha kısa mesafeli yollar oluşturduğunu ve bu nedenle daha az simülasyon adımında hedefe ulaştığını göstermektedir.

Simülasyon verileri karşılaştırıldığında, en kısa ortalama mesafenin 57 cm ile PRM–Pure Pursuit algoritmasına, en uzun ortalama mesafenin ise 62 cm ile RRT–Pure Pursuit algoritmasına ait olduğu belirlenmiştir. Bu fark, PRM algoritmasının düğüm bağlantı yapısının daha optimal yollar üretmesinden kaynaklanmaktadır.

Sonuç olarak, geliştirilen üç algoritma da sürü robotlarının iş birliğine dayalı şekilde hedefe ulaşmasını başarılı bir şekilde sağlamış; ancak farklı algoritmalar, ortamın yapısı ve robot sayısına göre farklı avantajlar sunmuştur. VFH algoritması çevresel esnekliğiyle öne çıkarken, RRT ve PRM algoritmaları daha planlı ve optimize yollar sunarak simülasyon süresini kısaltmıştır. Elde edilen sonuçlar, sürü robotlarının farklı görev senaryolarına göre uygun algoritma tercihinin yapılmasının önemini göstermektedir.

Çıkar Çatışması (Conflict of Interest)

Yazar tarafından herhangi bir çıkar çatışması beyan edilmemiştir. No conflict of interest was declared by the author.

Kaynakça (References)

- Ahuja, R. K., Mehlhorn, K., Orlin, J., & Tarjan, R. E. (1990). Faster algorithms for the shortest path problem. *Journal of the ACM*, 37(2), 213-223. <https://doi.org/10.1145/77600.776158>
- Alarabi, S., Luo, C., & Santora, M. (2022). A PRM Approach to Path Planning with Obstacle Avoidance of an Autonomous Robot. 2022 8th International Conference on Automation, Robotics and Applications (ICARA), 76-80. <https://doi.org/10.1109/ICARA5094.2022.9738559>
- Borenstein, İ. U. J. (1998). The vector field histogram - fast obstacle avoidance for mobile robots. *İeee İnternational Conference On Robotics And Automation*, 7(3), 278 - 288. <https://doi.org/10.1109/umagd.70.88137>
- Coppola, M., Guo, J., Gill, E., & de Croon, G. C. H. E. (2019). Provable self-organizing pattern formation by a swarm of robots with limited knowledge. *Swarm Intelligence*, 13(1), 59-94. <https://doi.org/10.1007/s11721-019-00163-0>
- Coulter, R. Implementation of the Pure Pursuit Path Tracking Algorithm. Carnegie Mellon University, Pittsburgh, Pennsylvania, Jan 1990.
- Çayırpunar, Ö. (ty.). Çoklu Robot Sistemlerinde Robotlar Arası Haberleşme Ve İş Birliği Kullanılarak Arama Verimliliğinin Artırılması.
- Dr. Lydia E. Kavraki: A Woman Making Robots Work. (2013, Ekim 14). *Mental Floss*. <https://www.mentalfloss.com/article/53176/dr-lydia-e-kavraki-woman-making-robotswork>
- Huang, Y., Tian, Z., Jiang, Q., & Xu, J. (2020). Path Tracking Based on Improved Pure Pursuit Model and PID. 2020 IEEE 2nd International Conference on Civil Aviation Safety and Information Technology (ICCASIT), 359-364. <https://doi.org/10.1109/ICCASIT50869.2020.9368694>
- Kavraki, L. E. (2013, October 14). A woman making robots work. *Mental Floss*. <https://www.mentalfloss.com/article/53176/dr-lydia-e-kavraki-woman-making-robotswork>
- LaValle, S. M., & Kuffner, J. J. (2001). Randomized kinodynamic planning. *The International Journal of Robotics Research*, 20(5), 378-400. <https://doi.org/10.1177/02783640122067453>
- LaValle, S. M. (2006). *Planning Algorithms*. Cambridge University Press.
- Li, Y. (2021). An RRT-Based Path Planning Strategy in a Dynamic Environment. 2021 7th International Conference on Automation, Robotics and Applications (ICARA), 1-5. <https://doi.org/10.1109/ICARA51699.2021.9376472>
- Lozano-Pérez, T., & Wesley, M. A. (1979). An algorithm for planning collision-free paths among polyhedral obstacles. *Communications of the ACM*, 22(10), 560-570. <https://doi.org/10.1145/359156.359164>
- Mısır, O., & Gökrem, L. (2020). Sürü Robotları için Esnek ve Ölçeklenebilir Toplanma Davranışı Metodu. *Avrupa Bilim Ve Teknoloji Dergisi*100-109. <https://doi.org/10.31590/ejosat.779162>
- Mısır, O., Çelik, M., & Gökrem, L. (2022). Waypoint-Based Path Tracking Approach For Self-Organized Swarm Robots. *Uluslararası Muhendislik Arastirma ve Gelistirme Dergisi*, 14, 799-815. <https://doi.org/10.29137/umagd.1118039>
- Misir, O., & Gökrem, L. (2021b). Flocking-Based Self-Organized Aggregation Behavior Method for Swarm Robotics. *Iranian Journal of Science and Technology, Transactions of Electrical Engineering*, 45(4), 1427-1444. <https://doi.org/10.1007/s40998-021-00442-9>
- Nemec, D., Janota, A., Hruboš, M., Gregor, M., & Pirnik, R. (2017). Mutual acoustic identification in the swarm of e-puck robots. *International Journal of Advanced Robotic Systems*, 14, 172988141771079. <https://doi.org/10.1177/1729881417710794>
- Park, J.-H., & Yoon, T.-W. (2018a). Efficient path planning using a modified PRM algorithm for a mobile robot. *Journal of Institute of Control, Robotics and Systems*, 24(5), 406-412.
- Park, J.-H., & Yoon, T.-W. (2018b). Maximizing the Coverage of Roadmap Graph for Optimal Motion Planning. *Complexity*, 2018, e9104720. <https://doi.org/10.1155/2018/9104720>
- Stormont, D.P. (2005). Autonomous rescue robot swarms for first responders. CIHSPS 2005. Proceedings of the 2005 IEEE International Conference on Computational Intelligence for Homeland Security and Personal Safety, 2005, 151-157.
- Ulrich, I., & Borenstein, J. (1998). VFH+: reliable obstacle avoidance for fast mobile robots. Proceedings. 1998 IEEE International Conference on Robotics and Automation (Cat. No.98CH36146), 2, 1572-1577 vol.2.
- Ulrich, I., & Borenstein, J. (1998). VFH+: Reliable Obstacle Avoidance for Fast Mobile Robots. Proceedings of the 1998 IEEE International Conference on Robotics and Automation, 1572-1577.
- Yaşar, E. (2020). Sürü Robotların Hareket Planlamada Kullanılması. *Avrupa Bilim Ve Teknoloji Dergisi*(20), 24-29. <https://doi.org/10.31590/ejosat.763444>
- Zheng, Y. T. Z. (2013). Research advance in swarm robotics. *Defence Technology*, 9(1), 18-39. <https://doi.org/10.1016/j.dt.2013.03.001>