



Kablosuz Sensör Ağlarında Parçacık Sürü Optimizasyonu, Sürekli Zaman Karınca Koloni Algoritması, Girdap Arama Algoritması, Yabani Ot Optimizasyon Algoritmaları ile Konum Belirleme ve K- Bağlılık İnceleme

Localization with Particle Swarm Optimization, Continuous Time Ant Colony Algorithm, Vortex Search Algorithm, Invasive Weed Optimization Algorithms with Wireless Sensor Networks and K-Connectivity Analysis

Feyza Nur Çakıcı*¹ , Tolga Eren¹ 

¹Kırıkkale Üniversitesi, Mühendislik ve Mimarlık Fakültesi, Elektirik-Elektornik Mühendisliği Bölümü, 71450 Kırıkkale, Türkiye

Başvuru/Received: 06/12/2021

Kabul / Accepted: 27/12/2021

Çevrimiçi Basım / Published Online: 31/12/2021

Son Versiyon/Final Version: 31/12/2021

Öz

Kablosuz sensör ağları (KSA) birçok farklı alanlarda kullanılmaktadır. Uygulanan alanlarda bilginin anlamlı hale gelmesi için alınan bilginin konumu tespit edilmesi önemlidir. KSA'ların konum belirlenmesinde farklı yöntemler kullanılmıştır. KSA'daki düğümlerden herhangi biri devre dışı kalsa bile ağdaki iletişim bozulmamalıdır. Bu durumda k-bağlılık ağ için önemli bir noktadır. Makalede Parçacık Sürü Optimizasyonu (Particle Swarm Optimization-PSO), Sürekli Zaman Karınca Koloni Algoritması (Ant Colony Optimization for Continuous Domains-ACO_R), Girdap Arama Algoritması (Vortex Search Algorithm-VSA) ve Yabani Ot Optimizasyon (Invasive Weed Optimization-IWO) algoritması kullanılarak sensör düğümlerin konumu tespit edilmiştir. İletim aralığına (R) bağlı olarak k-bağlılık hesaplanmıştır. Matlab ortamında algoritmalara k-bağlılık uygulanarak konumlandırma meydana gelen ortalama hata incelenmiş ve simülasyon sonuçları tablolar ve şekiller halinde gösterilmiştir.

Anahtar Kelimeler

“Kablosuz sensör ağlarında konumlandırma, PSO, ACO_R, VSA, IWO, K-bağlılık”

Abstract

Wireless sensor networks (WSN) are used in many different fields. It is important to determine the location of the information received so that the information becomes meaningful in the applied areas. Different methods have been used to determine the location of WSN's. Even if any of the nodes in the WSN are disabled, communication in the network should not be disrupted. In this case, k-connectivity is an important point for the network. In the article, Particle Swarm Optimization (PSO), Ant Colony Optimization for Continuous Domains (ACO_R), Vortex Search Algorithm (VSA), and Invasive Weed Optimization (IWO) the location of the sensor nodes are determined using. The k-connectivity is calculated based on the transmission distance (R). The average error in the localization is examined by applying k-connectivity to the algorithms in the Matlab environment, and the simulation results are shown in tables and figures.

Key Words

“Localization in wireless sensor networks, PSO, ACO_R, VSA, IWO, K-connectivity”

1. Giriş

Kablosuz sensör ağları (KSA) teknolojinin ilerlemesi ile birlikte boyutlar küçülmüş ve enerji tüketimi azalmıştır. Böylelikle birçok farklı alanda kullanılması sağlanmıştır. KSA'lar başlangıçta askeri amaçlar için kullanılmış ve daha sonra eğitim, tarım, sağlık uygulamaları, deprem tespiti ve hayvanların takibi gibi birçok önemli alanlarda uygulanmıştır.

KSA'daki düğümler dört ana bölümden meydana gelmektedir. Bunlar sensör birimi, işlemci birimi, alıcı-gönderici birimi ve güç birimidir. KSA'da yer alan sensör düğümleri coğrafik alanlara rastgele dağıtılmaktadır. Coğrafik alanda tespit edilecek veriler sensörler aracılığı ile algılanmakta ve ayrı ayrı işlenmektedir. Daha sonra işlenen veriler komşu sensörlere iletilmekte ve son olarak baz istasyonuna gönderilmektedir. Rastgele dağıtılan sensör düğümlerinde konumu bilinen ve konumu bilinmeyen düğümler bulunmaktadır. GPS veya manuel yöntemleri aracılığı ile önceden konumu bilinen düğümlere referans (çapa) düğümü olarak adlandırılmaktadır. Konumu bilinmeyen düğümler ise hedef düğüm olarak isimlendirilmektedir. Hedef düğümlerin konumu GPS tekniği ile tespit etmek maliyetli ayrıca enerji tüketimi fazladır. Hareketli yapıya sahip olan sensör düğümlerin manuel ile konumunun tespiti yapmak mümkün değildir. Bu durumda hedef düğümlerin konumunun tespiti için farklı metotlar kullanılmaktadır. Bu metotlardan biri çapa düğümü ile hedef sensörleri arasındaki mesafe hesaplanarak bulunmakta ve bu yöntemle network lozalization (ağ konumu belirleme) denilmektedir (Eren, vd. 2004).

Sensör düğümlerin herhangi bir bakıma gereksinim duymamakta ve oluşabilecek hatalara karşı toleranslı olmaktadır. Bundan dolayı ulaşılması zor alanlarda da kullanılmaktadır. Sensör düğümlerin sahip olduğu avantajları sayesinde birçok alanda kullanıldığı için düğümlerin konumu belirlenmesi önemli bir çalışma alanı oluşturmaktadır. Hedef düğümlerin konumunun tespit edilmesinde literatürde farklı algoritmalar kullanılmakta olup, optimizasyon problemi olarak yer almaktadır. Gerçekleştirilen çalışmaların bazıları açıklamaları ile birlikte yer verilmektedir.

Makalede hedef sensörlerin konum belirlenmesinde PSO, ACO_R, VSA ve IWO algoritmaları kullanılmıştır. Algoritmalarda elde edilen sonuçlar karşılaştırılmıştır. Makalenin 2. Bölümünde KSA'larda konumlandırma problemi ve hata fonksiyonun elde edilmesi hakkında bilgi verilmiştir. 3. Bölümünde konum belirlemede kullanılan algoritmalar hakkında detaylı bir şekilde açıklanmıştır. 4. Bölümde ise KSA'larda k-bağıllık hakkında bilgi verilmiş ve şekiller verilerek konu incelenmiştir. 5. Bölümde ise elde edilen simülasyon sonuçları tablolar ve şekiller halinde gösterilmiş ve son olarak 6. Bölümde algoritmalarda elde edilen sonuçlar grafikte gösterilmiştir.

2. KSA'larda Konumlandırma ve Hata Fonksiyonu

KSA'lar birçok alanda kullanımı artmaktadır. Bu nedenle sensör düğümlerinin konumlarının belirlenmesi önemlidir. Hedef sensör düğümlerin konumu için GPS'den farklı olarak dağıtılmış konumlandırma ve merkezi konumlandırma yöntemleri kullanılmaktadır. Dağıtılmış konumlandırma metodu mesafeden bağımsız ve mesafeye bağlı konumlandırma olmak üzere ikiye ayrılmaktadır. Kullanılan bu iki yöntem çapaya bağlı veya çapadan bağımsız olarak iki farklı şekilde hesaplanmaktadır. Mesafeye bağımlı konum bulma yönteminde, sensör düğümlerin birbirlerine olan uzaklığı hesaplanmaktadır. Mesafeye bağımlı ve mesafeden bağımsız konum belirleme yöntemleri karşılaştırıldığında mesafeden bağımsız konum belirleme yöntemi daha basittir ve maliyeti daha azdır ancak elde edilen hata oranı yüksektir. Mesafeye bağımlı konum belirleme yöntemi, Varış Zamanı (Time of Arrival-ToA), Varış Zamanı Fark (Time Difference of Arrival-TDoA), Alınan Sinyal Gücü Göstergesi (Received Signal Strength Indication-RSSI), Varış Açısı (Angle of Arrival-AoA) gibi yöntemler kullanılarak hesaplanmaktadır. Mesafeden bağımsız konum belirleme yönteminde ise Dv-Hop, Amorphus, Üçgende Yaklaşık Nokta (Approximate Point in Triangle-Apıt) ve Centroid yöntemler kullanılmaktadır. Konumu bilinmeyen sensör düğümlerinin konumunu belirlemek için konumu bilinen referans (çapa) düğümlerinin 2-boyutlu uzayda en az üç tane veya 3-boyutlu uzayda en az 4 tane olması gereklidir.

KSA'larında konum belirleme probleminde N adet çapa düğümü ve M adet hedef düğümü olduğunu varsayarsak çapa düğümlerin koordinatları (x_j, y_j) hedef düğümlerinin koordinatları (x_i, y_i) şeklinde ifade edilmektedir. Gerçek uzaklık, hedef düğümleri ile çapa düğümleri arasındaki mesafe hesaplanarak bulunaktadır. Gerçek uzaklık denklem (1) kullanılarak hesaplanmaktadır. d_i gerçek uzaklığı ifade etmektedir.

$$d_i = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2} \quad (1)$$

Gerçek uzaklığa, gaussian gürültü (n_i) eklenerek tahmini uzaklık (d_{in}) elde edilmektedir. Tahmini uzaklık denklem (2) kullanılarak hesaplanmaktadır.

$$d_{in} = d_i + n_i \quad (2)$$

KSA'larda konumu bilinmeyen sensör düğümlerinin konumunun belirlenmesi optimizasyon problemi olarak ele alınmıştır. Yapılan çalışmada farklı optimizasyon algoritmaları ile gerçek mesafe ile tahmini mesafe arasındaki farkı en az indirerek hata oranını düşürmektir. Optimizasyon probleminde belirlenen amaç fonksiyonu denklem (3) ile hesaplanmaktadır.

$$f(x, y) = \frac{1}{N} \sum_{i=1}^N (d_i - d_{in})^2 \quad (3)$$

Konumu bilinmeyen hedef düğümlerinin konumları optimizasyon algoritmalar kullanılarak en az hata ile tespit edilmektedir. Hata fonksiyonu denklem (4) kullanılarak elde edilmektedir.

$$E = \frac{1}{M} \sum_{k=1}^M \sqrt{(x_i - X_i)^2 + (y_i - Y_i)^2} \quad (4)$$

Denklem (4)'te (x_i, y_i) gerçek sensör düğümlerinin koordinatlarını, (X_i, Y_i) ise sezgisel optimizasyon algoritması ile tahmin edilen sensör düğümlerinin koordinatlarını ifade etmektedir.

3. Konumlandırmada Kullanılan Optimizasyon Algoritmaları

Konumu bilinmeyen hedef sensör düğümlerinin konumunu belirlemek için PSO, ACO_R, VSA ve IWO algoritmaları kullanılmıştır. Bu bölümde PSO, ACO_R, VSA ve IWO algoritmaları hakkında detaylı bir şekilde bilgi verilmiştir.

3.1. Parçacık Sürü Optimizasyon Algoritması (PSO)

Kennedy ve Eberhart (1995) tarafından geliştirilen algoritma, balık ve kuş sürülerinin yiyecek arama sırasında sergiledikleri davranışlardan esinlenerek tasarlanmıştır. PSO algoritmasında her bir bireye parçacık, parçacıkların oluşturduğu topluluğa da sürü adı verilmektedir. Parçacıklar çözüm uzayına rastgele dağıtılır ve yiyecek arama davranışlar önceki deneyimlerinden ve yakın olan parçacıkların deneyimlerinden faydalanmaktadır. Böylece parçacıkların en iyi konumu bulunmaktadır. Aşağıda verilen denklem (5) kullanılarak parçacıkların başlangıç konumları bulunmaktadır.

$$X_{i,d} = X_{min,d} + r_{i,d} \times (X_{max,d} - X_{min,d}) \quad (5)$$

Burada d-boyutlu uzayda, i tane parçacığın başlangıç konumunu $X_{min,d}$ ve $X_{max,d}$ alt ve üst limitleri, $r_{i,d}$ 0 ile 1 aralığında değer alan rastgele sayıyı ifade etmektedir. Her bir iterasyonda i. parçacığın hızı denklem (6) ile konumu ise denklem (7) ile güncellenmektedir.

$$V_{i,d}^{t+1} = (w^{t+1} \times V_{i,d}^t) + c_1 r_1 \times (Pbest_{i,d}^t - X_{i,d}^t) + c_2 r_2 \times (Gbest_{i,d}^t - X_{i,d}^t) \quad (6)$$

$$X_{i,d}^{t+1} = X_{i,d}^t + V_{i,d}^{t+1} \quad (7)$$

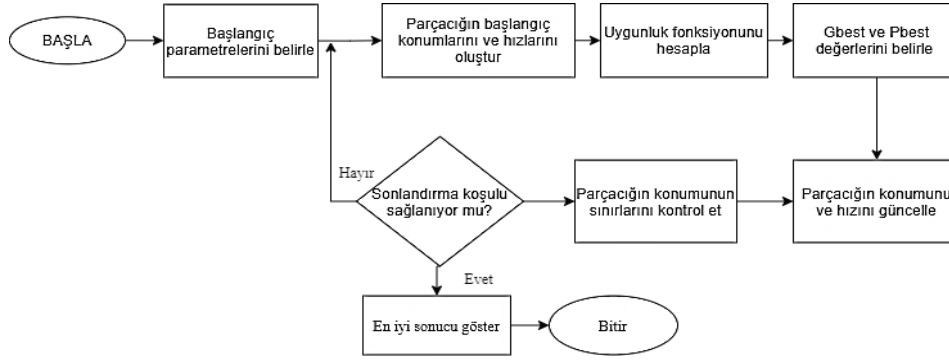
Denklem (6)'da c_1 ve c_2 öğrenme faktörlerini, r_1 ve r_2 0 ile 1 aralığında rastgele değer alan katsayıları ifade etmektedir. $Gbest$, global en iyi konum olarak isimlendirilir ve sürüdeki herhangi bir parçacığın en iyi konumunu göstermektedir. $Pbest$, lokal en iyi konum olarak isimlendirilir ve parçacığın o zamana kadarki en iyi konumunu belirtmektedir (Berber, vd. 2016). Lokal en iyi konum denklem (8) ile güncellenmektedir.

$$Pbest_{i,d}^{t+1} = \begin{cases} Pbest_{i,d}^{t+1} & , f(X_{i,d}^{t+1}) \geq f(Pbest_{i,d}^t) \\ X_{i,d}^{t+1} & , f(X_{i,d}^{t+1}) < f(Pbest_{i,d}^t) \end{cases} \quad (8)$$

Denklem (8)'de amaç fonksiyonu f ile gösterilmektedir. PSO'da hız vektörünün sınırlandırılması denklem (9)'da gösterilmiştir.

$$V_{i,d}^{t+1} = \begin{cases} V_{i,d}^{t+1} & , V_{i,d}^{t+1} < V_{max} \\ V_{max} & , V_{i,d}^{t+1} \geq V_{max} \end{cases} \quad (9)$$

Şekil 1’de PSO’nun akış diyagramı gösterilmektedir.



Şekil 1. PSO Algoritmasının Akış Diyagramı

3.2. Sürekli Zaman Karınca Koloni Optimizasyon Algoritması (ACO_R)

Karınca Koloni Algoritması (ACO) Dorigo ve arkadaşları (1991) tarafından oluşturulmuştur. ACO karıncanın yemek kaynağına ulaşabilecek en kısa yolu bulmasından esinlenerek geliştirilmiştir. Karıncalar vücutlarında feromon adı verilen kimyasal madde salgılamaktadır. Feromon maddenin miktarının daha yoğun olan yönü karıncalar tarafından seçilerek en kısa yolu seçmiş olmaktadır. ACO algoritması ayrık optimizasyon problemleri için Socha ve Dorigo (2008) tarafından ACO algoritması geliştirilerek Sürekli Zaman Karınca Optimizasyon algoritması oluşturulmuştur. ACO_R algoritmasında sürekli bir dağılım olasılık fonksiyonu kullanılmıştır.

Yeni çözüm değişkeni, çözüm uzayından seçilen bir değişkeninin Gaussian örnekleme kullanılarak seçilmektedir. Örnekleme için seçilen değişkenin benzer olmaması için Rulet Çarkı stratejisi kullanılmaktadır. Rulet Çarkı daha geniş çözüm sağlamaktadır (Ojha, Abraham ve Snasel, 2017). Gaussian tabanlı birden fazla olasılık yoğunluk fonksiyonunun toplamından gaussian çekirdek fonksiyonu meydana gelmektedir. Gaussian çekirdek fonksiyonu denklem (10) kullanılarak hesaplanmaktadır.

$$G^i(x) = \sum_{l=1}^k w_l g_l^i(x) = \sum_{l=1}^k w_l \frac{1}{\sigma_l^i \sqrt{2\pi}} e^{-\frac{(x-\mu_l^i)^2}{2\sigma_l^{i2}}} \quad (10)$$

Denklem (10)'da $i=1,2,3...n$ problemin boyutunu, w her bir Gaussian fonksiyonun ağırlık vektörünü, μ_l^i ortalama vektörünü ve σ_l^i standart sapma vektörünü ifade etmektedir. Denklem (11)'de ise μ^i Gaussian çekirdek fonksiyonunun ortalama vektörü gösterilmiştir. S_k^i , aday çözümlerini ifade etmektedir.

$$\mu^i = \{\mu_1^i, \dots, \dots, \mu_k^i\} = \{S_1^i, \dots, \dots, S_k^i\} \quad (11)$$

Ağırlık vektörü w , denklem (12) kullanılarak bulunmaktadır. Ağırlık vektörü amaç fonksiyonunda elde edilen değerlere göre sıralanmaktadır.

$$w_l = \frac{1}{q_k \sqrt{2\pi}} e^{-\frac{(l-1)^2}{2q^2 k^2}} \quad (12)$$

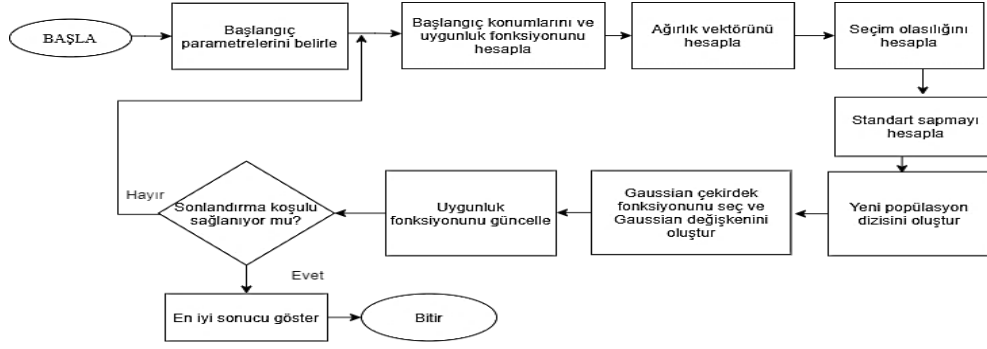
Denklem (12)'de q_k standart sapmayı ifade etmektedir. Ağırlık vektörü, l argümanlı ve ortalama değeri 1 almaktadır. En iyi çözümlerin seçilme olasılıkları q 'nın değeri küçüldükçe artmaktadır. ACO_R'da Gaussian fonksiyonlarının seçilebilme olasılığı formülü denklem (13)'te verilmiştir.

$$P_l = \frac{w_l}{\sum_{r=1}^k w_r} \quad (13)$$

$$\sigma_l^i = \xi \sum_{e=1}^k \frac{|S_e^i - S_l^i|}{k-1} \quad (14)$$

Denklem (14)'te verilen standart sapma σ_l^i kullanılarak seçilen çözümün parametreleri hesaplanmaktadır. Standart sapma ile elde edilen diğer çözümler arasındaki ortalama mesafe elde edilmektedir. ξ parametresinin değeri, en iyi çözüme yakınsama hızını belirtmektedir. ξ değeri arttıkça yakınsama hızı azalmaktadır.

ACO_R algoritmasının akış diyagramı Şekil 2'de gösterilmiştir.

Şekil 2. ACO_R Algoritmasının Akış Diyagramı

3.3. Girdap Arama Algoritması (VSA)

Girdap Arama Algoritması (VSA) girdap modelinden esinlenerek Doğan ve Ölmez (2015) tarafından modellenmiştir. Sürekli optimizasyon problemlerinin çözümü için kullanılır ve oldukça basittir. VSA çemberlerin yarıçapları azaltılarak iç içe geçmiş iki çemberden meydana gelmektedir. Birinci çemberin yani en dıştaki çemberin başlangıç merkezi denklem (15) ile hesaplanmaktadır.

$$\mu_0 = \frac{\text{üstlimit} + \text{altlimit}}{2} \quad (15)$$

Denklem (15)'te üst limit ve alt limit problemin boyutunu belirlemektedir. Başlangıç yarıçapı σ_0 denklem (16) kullanılarak hesaplanmaktadır.

$$\sigma_0 = \frac{\max(\text{üstlimit}) - \min(\text{altlimit})}{2} \quad (16)$$

a şekil parametresini ifade etmektedir. Her iterasyonda güncellenir ve denklem (17) ile bulunmaktadır.

$$a_t = a_0 - \frac{t}{\text{MaxItr}} \quad (17)$$

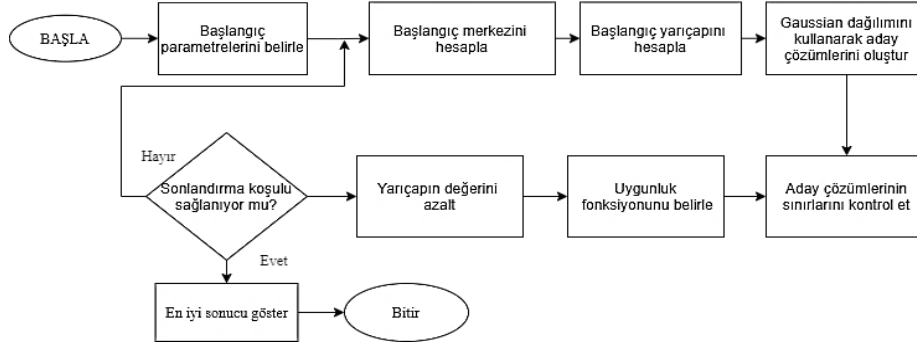
İlk iterasyonda arama alanının tamamının kapsaması için $a_0 = 1$ seçilmektedir. Denklem (17)'de t iterasyon indeksini ve MaxItr ise maksimum iterasyon sayısını temsil etmektedir. Çemberin yarıçapı ise denklem (18) kullanılarak hesaplanmaktadır. Ters tamamlanmamış gama fonksiyonu her iterasyonda yarıçapın değerini azaltmaktadır.

$$r_t = \sigma_0 \times (1/x) \times \Gamma(x, a_t) \quad (18)$$

Denklem (18)'de x 'in değeri 0.1 alınarak daha iyi performans elde edilmektedir. Γ ters gamma fonksiyonunu ifade etmektedir ve Matlab programında `gammaincinv` fonksiyonu kullanılmaktadır. VSA'da gaussian dağılımı kullanılarak aday çözümler elde edilmektedir. Aday çözümler arama sınırları içinde yer almalıdır. Arama sınırları içinde yer almayan aday çözümler içinde yer almalıdır. Arama sınırları içinde yer almayan aday çözümler denklem (19) kullanılarak sınırlara kaydırılmaktadır. Denklem (19)'da $C(s)_k^i$ k. komşu çözümün i. boyutunu ve düzgün dağıtılmış (uniform) rastgele sayıyı ifade etmektedir.

$$C(s)_k^i = \begin{cases} \text{rand.}(\text{üstlimit}^i - \text{altlimit}^i) + \text{altlimit}^i & , C(s)_k^i < \text{altlimit}^i \\ C(s)_k^i & , \text{altlimit}^i \leq C(s)_k^i \leq \text{üstlimit}^i \\ \text{rand.}(\text{üstlimit}^i - \text{altlimit}^i) + \text{altlimit}^i & , C(s)_k^i > \text{üstlimit}^i \end{cases} \quad (19)$$

VSA'nın akış diyagramı Şekil 3'te gösterilmektedir.



Şekil 3. VSA Algoritmasının Akış Diyagramı

3.4. Yabani Ot Optimizasyon Algoritması (IWO)

Yabani ot optimizasyon (IWO) algoritması Mehrabian ve Lucas (2006) tarafından önerilmiştir. Yabani otlar, sağlamdır ve çevreye adapte olan bitkilerdir. Yabani otlar tarlaya rastgele dağılarak istila etmektedir. Tarlada ekili olan bitkilere zarar vermekte ve istenilmeyen alanlarda büyümektedir.

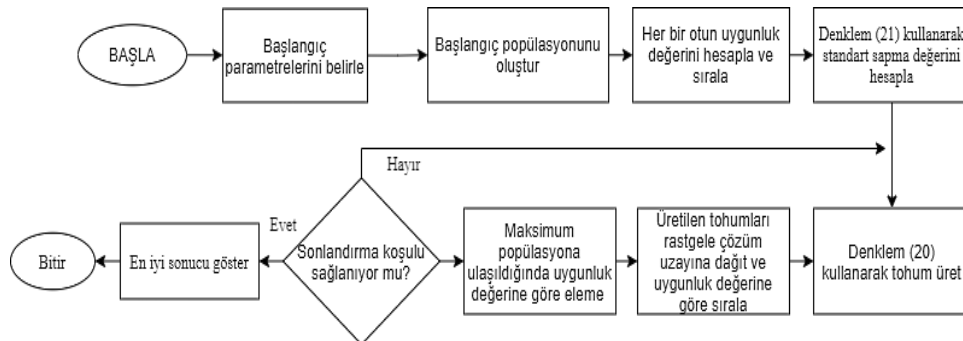
IWO algoritmasında popülasyon d-boyutlu arama uzayına rastgele dağıtmakta, her bireye tohum denilmekte ve optimizasyon değişkenini temsil etmektedir. Popülasyonda her üyenin kendisinin ve kolonideki diğer bireylerin sahip olduğu en düşük ve en yüksek amaç fonksiyonunun aldığı değere göre tohum üretilmektedir. Yani yeni tohumlar üretilmeden önce uygunluk değerine bağlı olarak yabani otlar sıralanır ve kolonideki sıralamasına göre tohumlar üretilmesine izin verilmektedir. Popülasyondaki otların ürettiği tohum sayısı, mümkün olan en az tohum üretiminden maksimum tohum üretimine doğru artmaktadır. Böylece tüm otların üretimine izin verilmektedir.

$$S = \left[S_{min} + (S_{max} - S_{min}) \left(\frac{U_i - U_{en\ kötü}}{U_{en\ iyi} - U_{en\ kötü}} \right) \right] \quad (20)$$

Denklem (20)'de S_{max} ve S_{min} maksimum sayıda ve minimum sayıda üretilecek tohum sayısını, $U_{en\ kötü}$ amaç fonksiyonunun aldığı en kötü değeri, $U_{en\ iyi}$ amaç fonksiyonunun aldığı en iyi değeri, U_i popülasyondaki bireyin sahip olduğu uygunluk değerini ifade etmektedir. Ayrıca ana bitkiden üretilen tohumlar d-boyutlu arama uzayına rastgele dağıtılmaktadır. Denklem (21) kullanılarak standart sapma hesaplanmaktadır.

$$\sigma_{iter} = \frac{(iter_{max} - iter)^n}{(iter_{max})^n} (\sigma_{ilk} - \sigma_{son}) + \sigma_{son} \quad (21)$$

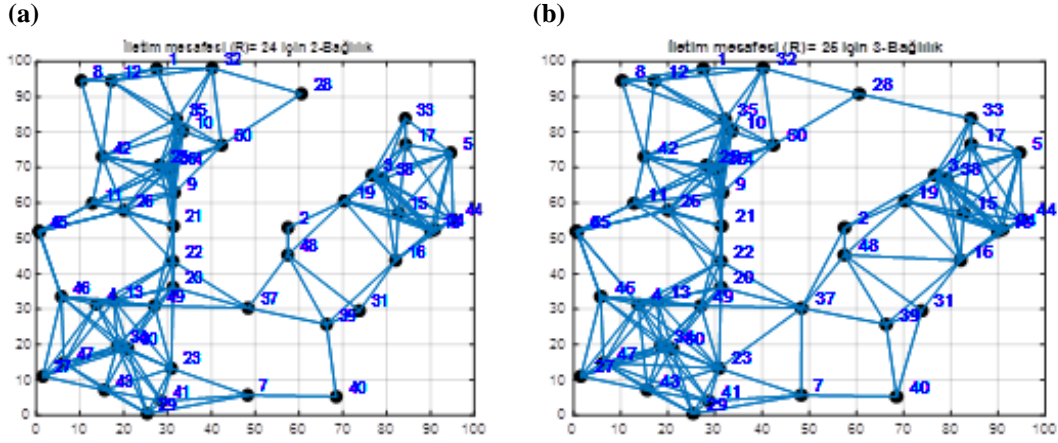
Denklem (21)'de σ_{iter} iterasyona ait standart sapmayı, $iter_{max}$ maksimum iterasyon sayısını, σ_{ilk} ilk standart sapma değerini, n doğrusal olmayan modülasyon indeksini belirtmektedir. Sonraki aşamada kolonideki maksimum bitki sayısını sınırlamak için bitkiler arasında rekabetçi eleme yapılmaktadır. Tüm tohumlar arama alanında konumlarını belirledikten sonra ebeveynleri ile sahip oldukları uygunluk değerine göre sıralanır. Daha sonra maksimum popülasyona erişmek için kolonideki sıralamalarında en düşük uygunluk değerine sahip olan otlar elenmektedir. Böylece elenmeyen otlar yeni tohumlar üretebilmektedir. Bu şekilde en iyi uygunluk değerine sahip olanlar hayatta kalmakta ve üremelerine izin verilmektedir. Şekil 4'te IWO'nun akış diyagramı gösterilmektedir.



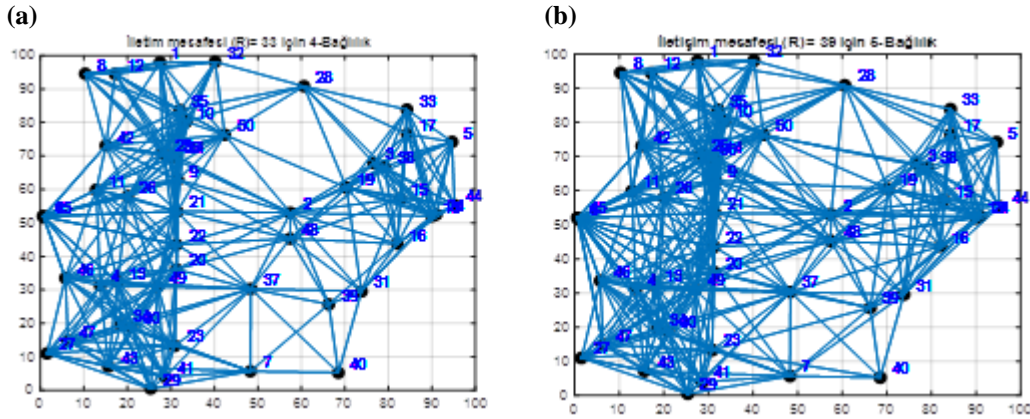
Şekil 4. IWO Algoritmasının Akış Diyagramı

4. KSA'larda K-Bağlılık Özelliği

KSA'da izlenilecek ve takip edilecek ortamdaki veriler toplandıktan sonra sensörler aracılığı ile baz istasyonuna gönderilmektedir. Verinin baz istasyonuna gönderilirken ortamda bulunan herhangi bir sensör düğümünden farklı yollarla iletilmesi gerekmektedir. Bu nedenle KSA'daki tüm sensör düğümlerinin iletişimi kesilmemesi önemlidir. KSA'lar tam bağlantılı olması durumunda gönderilen veri baz istasyonuna her zaman ulaşmaktadır.



Şekil 5. (a) R=24'te 2-Bağlılık; (b) R=25'te 3-Bağlılık



Şekil 6. (a) R=33'te 4-Bağlılık; (b) R=39'da 5-Bağlılık

KSA'da herhangi bir sensör düğümünde meydana gelebilecek arızada, bazı sensör düğümlerinin ağa ile iletişimi kesilebilmekte veya KSA'daki yapıyı ikiye bölerek ağın bozulmasına sebep olmaktadır. Bu durumda KSA'da her bir sensör düğümü arasında birden fazla bağlantı olursa herhangi bir sensör düğümünün devre dışı kalsa bile alternatif bağlantılar kullanılarak sensör düğümü arasındaki iletişim kesilmemektedir. KSA'da herhangi iki sensör düğümü arasında en az k adet bağlantı mevcut ise KSA k -bağlılık (k -connectivity) denilmektedir. k -bağlı KSA'da, $k-1$ sensör düğümünün çalışmaması durumunda ağda bulunan diğer sensör düğümlerinin iletişimini etkilememektedir.

Şekil 5(a)'da iletim aralığı $R=24$ 'te 50 düğümden oluşan 2-bağlı KSA örneği gösterilmiştir. KSA'da 37 ve 7 numaralı düğümlerin devre dışı kalması ile KSA'daki 2, 3, 7, 15, 16, 17, 18, 24, 31, 33, 38, 39, 40, 44 ve 48 numaralı düğümlerin ağdaki diğer düğümler arasındaki bağlantı kesilmektedir. Şekil 5(b)'de $R=25$ 'te 3-bağlı, Şekil 6(a)'da $R=33$ 'te 4-bağlı ve Şekil 6(b)'de $R=39$ 'da 5-bağlı KSA örnekleri gösterilmiştir.

5. Simülasyonlar

KSA'larda konumu bilinmeyen hedef sensörlerin konumlarını belirlemek için kullanılan PSO, ACO_R, VSA ve IWO algoritmalarının simülasyon sonuçları Şekil 7, 8, 9, 10, 11, 12, 13 ve 14'te gösterilmektedir. Algoritmaların k -bağlılığa bağlı olarak hesaplanan ortalama hata değeri Şekil 15'te gösterilmektedir.

5.1. Parametreler ve Değerleri

PSO, ACO_R, VSA ve IWO algoritmalarında kullanılan parametre değerleri Tablo 1, 2, 3 ve 4'te gösterilmektedir. Kullanılan algoritmalarda problemin boyutu 2, popülasyon sayısı 20, iterasyon sayısı 100, alt limit 0 ve üst limit 100 alınmaktadır.

Tablo 1. PSO Algoritmasında Kullanılan Parametreler ve Değerleri

PARAMETRELER	DEĞERLERİ
Ağırlık Katsayısı (w)	1
Ağırlık Katsayısı Sönümlenme Oranı (w_{damp})	0.99
Bilişsel Öğrenme Faktörü (c_1)	1.5
Sosyal Öğrenme Faktörü (c_2)	2.0

Tablo 2. ACO_R Algoritmasında Kullanılan Parametreler ve Değerleri

PARAMETRELER	DEĞERLERİ
Standart Sapma (q)	0.5
Sapma-Mesafe Oranı (ξ)	1

Tablo 3. VSA Algoritmasında Kullanılan Parametreler ve Değerleri

PARAMETRELER	DEĞERLERİ
x	0.1
Şekil parametresi (a_0)	1

Tablo 4. IWO Algoritmasında Kullanılan Parametreler ve Değerleri

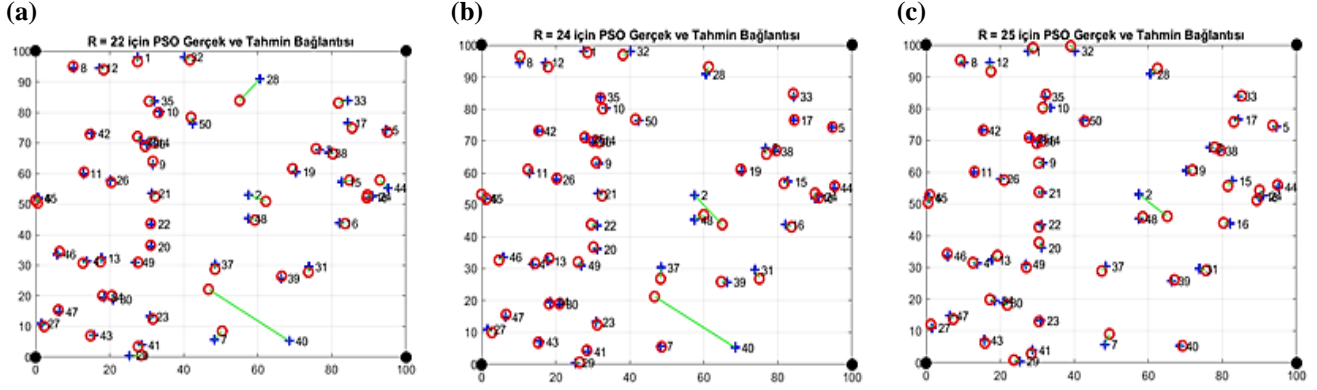
PARAMETRELER	DEĞERLERİ
Başlangıç popülasyon boyutu	20
Maksimum popülasyon boyutu	20
Minimum tohum sayısı (S_{min})	1
Maksimum tohum sayısı (S_{max})	5
Varyans azaltma üssü	3
Standart sapmanın başlangıç değeri	20
Standart sapmanın son değeri	0.2

5.2. Bulgular

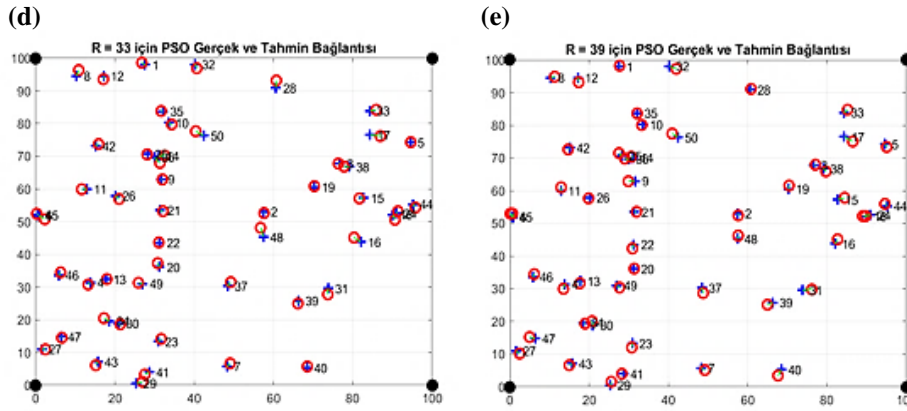
Matlab programında 4 adet referans düğümü (çapa) 100x100' lük bir alanda köşelere yerleştirilmiştir. 50 adet hedef sensör düğümü rastgele dağıtılmıştır. 50 adet hedef sensör düğümünün konum tespiti için PSO, ACO_R, VSA ve IWO kullanılmıştır. Her bir sensörün iletim aralığı R=22, R=24(2-Bağlılık), R=25(3-Bağlılık), R=33(4-Bağlılık) ve R=39(5-Bağlılık) birim alınarak aynı düğüm koordinatları üzerinde PSO, ACO_R, VSA ve IWO algoritmaları için ayrı ayrı hesaplanmıştır. Tüm algoritmaların iletim aralığına bağlı 2-bağlılık, 3-bağlılık, 4-bağlılık ve 5-bağlılık elde edilmiş ve Şekil 7, 8, 9, 10, 11, 12, 13 ve 14'te gösterilmiştir. Şekillerde yer alan mavi renkli + işaretler düğümün gerçek konumunu, kırmızı renkli yuvarlak işaretler algoritmalar kullanılarak elde edilen tahmini konumu ve siyah renkli yuvarlak işaretler GPS ile konumu tespit edilen referans düğümü ifade etmektedir. Ayrıca yeşil renkli çizgiler sensör düğümlerin gerçek konumu ile algoritmalar kullanılarak bulunan tahmini konum arasındaki mesafe ifade edilmiştir. Gerçek konum ile tahmin edilen konum arasındaki mesafeler PSO için Şekil 7(a), 7(b), 7(c), 8(d) ve 8(e), ACO_R için 9(a), 9(b), 9(c), 10(d) ve 10(e), VSA için 11(a), 11(b), 11(c), 12(d) ve 12(e) ve IWO için 13(a), 13(b), 13(c), 14(d) ve 14(e)'te simülasyon sonuçları gösterilmiştir. Tablo 5'te iletim aralığı ile k-bağlılık arasındaki ilişki gösterilmiştir. Tablo 6'da ise PSO, ACO_R, VSA ve IWO algoritmalarının simülasyon sonuçları 30 defa tekrarlanarak elde edilen ortalama hata değerleri gösterilmiş ve Şekil 15'te ayrıntılı bir şekilde incelenmiştir.

Tablo 5. İletim Aralığı ve K-Bağlılık

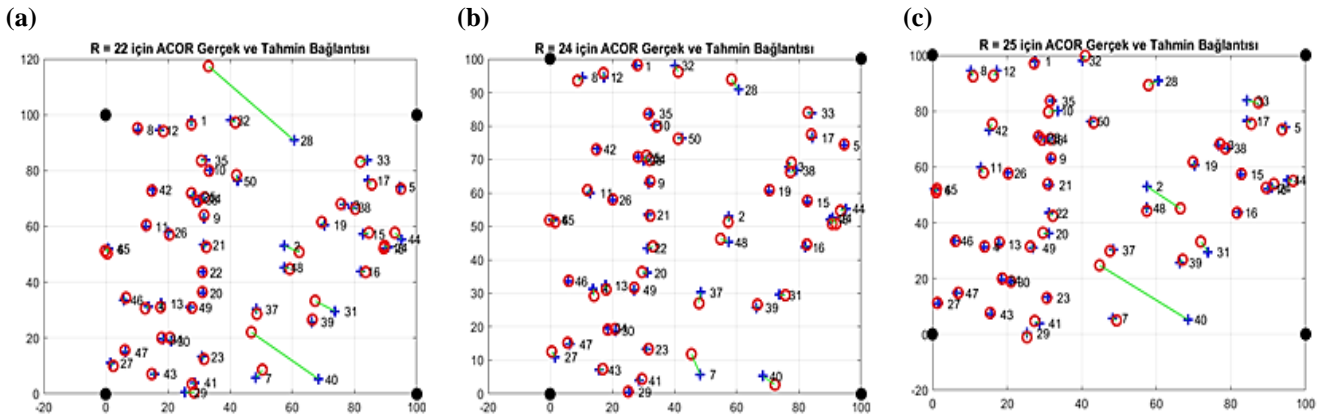
İletim Aralığı (R)	K-bağlılık
R=24	2-Bağlılık
R=25	3-Bağlılık
R=33	4-Bağlılık
R=39	5-Bağlılık

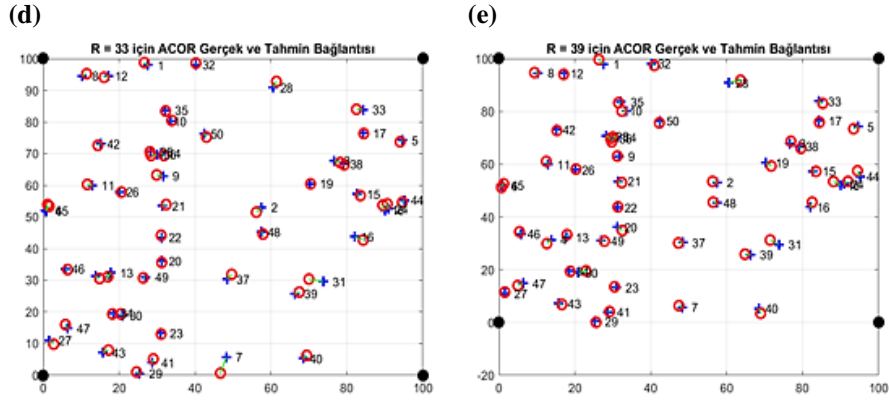


Şekil 7. (a) PSO'nun R=22'de gerçek ve tahmin konumu arasındaki mesafe; (b) PSO'nun R=24'te gerçek ve tahmin konumu arasındaki mesafe; (c) PSO'nun R=25'te gerçek ve tahmin konumu arasındaki mesafe

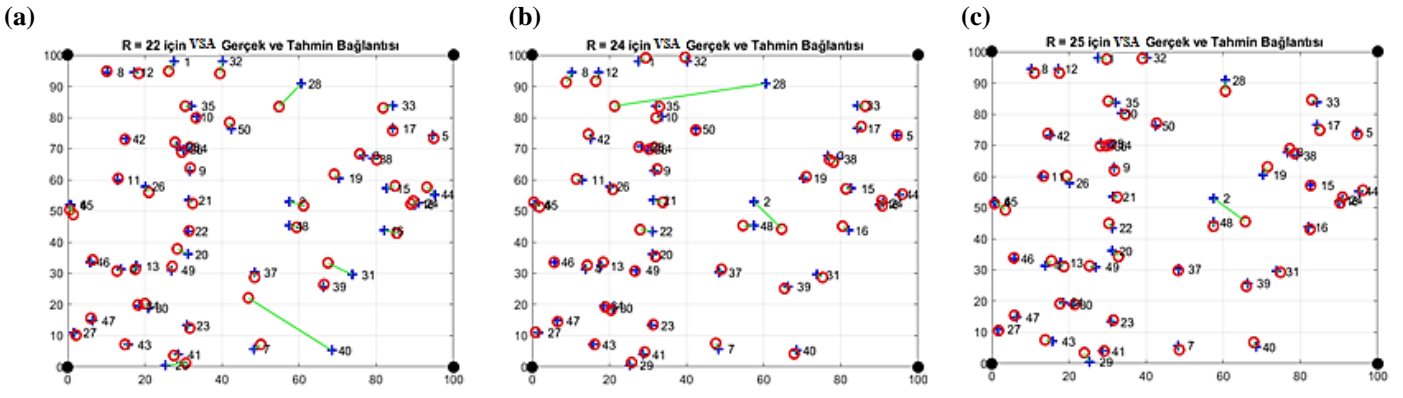


Şekil 8. (d) PSO'nun R=33'te gerçek ve tahmin konumu arasındaki mesafe; (e) PSO'nun R=39'da gerçek ve tahmin konumu arasındaki mesafe

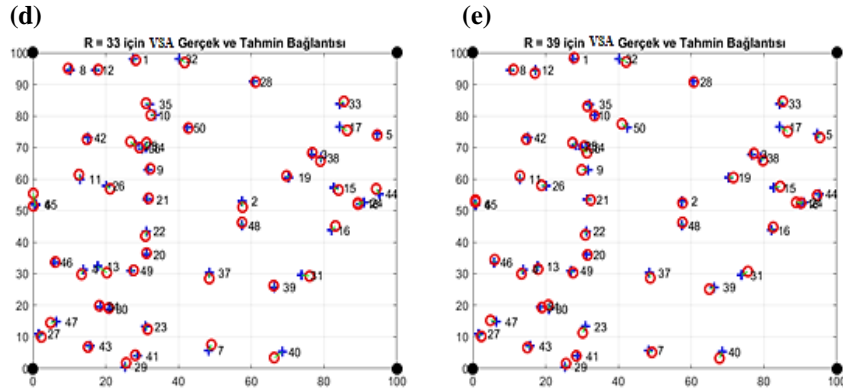
Şekil 9. (a) ACO_R'nin R=22'de gerçek ve tahmin konumu arasındaki mesafe; (b) ACO_R'nin R=24'te gerçek ve tahmin konumu arasındaki mesafe; (c) ACO_R'nin R=25'te gerçek ve tahmin konumu arasındaki mesafe



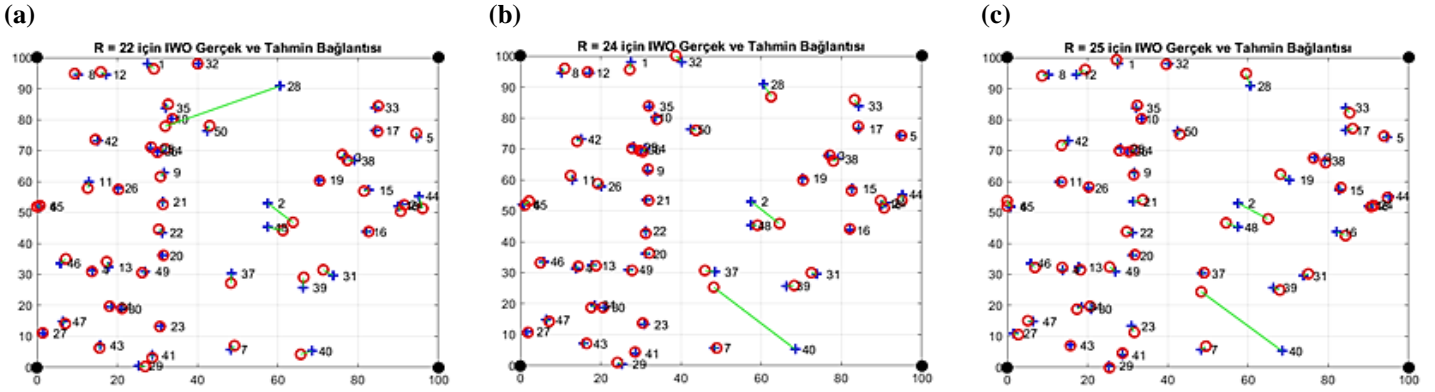
Şekil 10. (d) $ACOR_R$ 'nin $R=33$ 'te gerçek ve tahmin konumu arasındaki mesafe; (e) $ACOR_R$ 'nin $R=39$ 'da gerçek ve tahmin konumu arasındaki mesafe



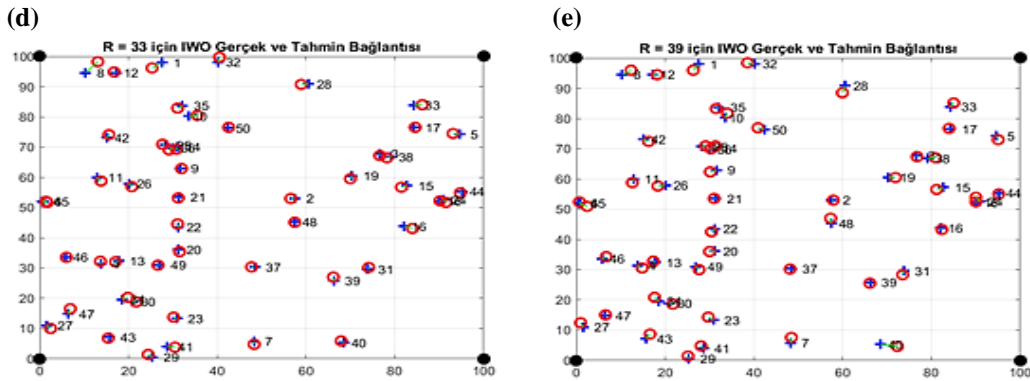
Şekil 11. (a) VSA 'nın $R=22$ 'de gerçek ve tahmin konumu arasındaki mesafe; (b) VSA 'nın $R=24$ 'te gerçek ve tahmin konumu arasındaki mesafe; (c) VSA 'nın $R=25$ 'te gerçek ve tahmin konumu arasındaki mesafe



Şekil 12. (d) VSA 'nın $R=33$ 'te gerçek ve tahmin konumu arasındaki mesafe; (e) VSA 'nın $R=39$ 'da gerçek ve tahmin konumu arasındaki mesafe



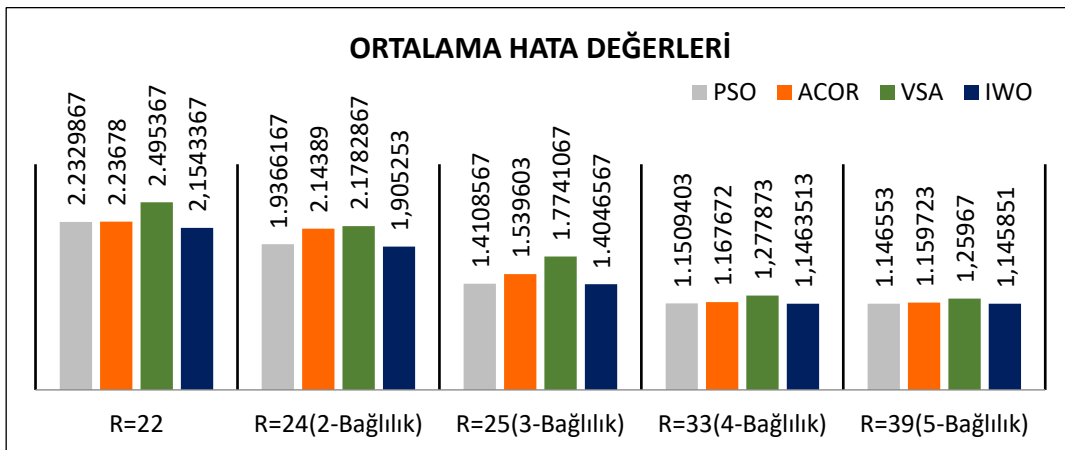
Şekil 13. (a) IWO'nun R=22'de gerçek ve tahmin konumu arasındaki mesafe; (b) IWO'nun R=24'te gerçek ve tahmin konumu arasındaki mesafe; (c) IWO'nun R=25'te gerçek ve tahmin konumu arasındaki mesafe



Şekil 14. (d) IWO'nun R=33'te gerçek ve tahmin konumu arasındaki mesafe; (e) IWO'nun R=39'da gerçek ve tahmin konumu arasındaki mesafe

Tablo 6. 30 İterasyonun Ortalama Hatası

Algoritmalar	R=22	R=24	R=25	R=33	R=39
PSO	2,2329867	1,9366167	1,4108567	1,1509403	1,146553
ACOR	2,23678	2,14389	1,539603	1,167672	1,159723
VSA	2,495367	2,1782867	1,7741067	1,253583	1,214003
IWO	2,1543367	1,905253	1,4046567	1,1463513	1,145851



Şekil 15. Algoritmaların Ortalama Hatası

6. Sonuçlar

Makale rastgele dağıtılan hedef sensör düğümlerin konumlarını tespit etmek için PSO, ACO_R, VSA ve IWO algoritmaları kullanılmıştır. İletim aralığına bağlı olarak k-bağlılık özelliği algoritmalar üzerinden incelenmiş ve her bir algoritmanın 30 iterasyonun ortalama hatası hesaplanmıştır. Tablo 6'daki verilere göre IWO diğer algoritmalara göre R=22, R=24(2-bağlılık), R=25(3-bağlılık), R=33(4-bağlılık) ve R=39(35-bağlılık) için en iyi ortalama hatası elde edilmiştir. ACO_R, IWO ve PSO algoritmaların ortalama hatası birbirine yakın olmasına rağmen VSA algoritması en yüksek değere sahiptir. Şekil 15 incelendiğinde IWO, ACO_R, VSA ve PSO algoritmalarının R=22 birimden R=39 birime kadar hesaplanan ortalama hata önemli derecede düşmüştür. Tüm algoritmalarda R=33(4-bağlılık) birim ile R=39 (5-bağlılık) biriminden elde edilen ortalama hatanın çok az seviyede düştüğü gözlemlenmiştir. Ayrıca iletim aralığı arttıkça k-bağlılık değerinin arttığı gözlemlenmiştir. K-bağlılık arttıkça harcanan enerji miktarı da artmaktadır. Bu sebepten dolayı R=33 kullanılması hem enerji hem de ortalama hata açısından daha verimlidir.

Referanslar

- Akram, V. K., Yiğit, Y. & Dağdeviren, O. (2018). Telsiz Duyurga Ağlarında K-Bağlılık Denetleme Algoritmalarının Performans Değerlendirilmesi. 20.Akademik Bilişim Konferansı, Adana, Turkey, 15 Ağustos.
- Aspnes, J., Eren, T., Goldenberg, D. K., Morse, A. S., Whiteley, W., Yang, R., Anderson, B. D. O. & Belhumeur, P. N. (2006). A theory of network localization. *IEEE Transactions on Mobile Computing*, 5(12), 1663–1677.
- Almasaeid, H. M. & Kamal, A. E. (2009). On the Minimum k-Connectivity Repair in Wireless Sensor Networks. 2009 IEEE International Conference on Communications, 1-5.
- Bacanin, N., Tuba, E., Zivkovic, M., Strumberger, I. & Tuba, M. (2019). Whale Optimization Algorithm with Exploratory Move for Wireless Sensor Networks Localization. *International Conference on Hybrid Intelligent Systems*, 328-338.
- Bekçibaşı, U., Tenruh, M. (2012). Kablosuz Algılayıcı Ağlarda Konum Saptama Teknikleri ve Mesafe Bağımlı Tekniklerde Dördüncü Çapa Yaklaşımı. *Akademik Bilişim'12 - XIV. Akademik Bilişim Konferansı Bildirileri*.
- Cui, H., Liang, Y., Zhou, C. & Cao, N. (2018). Localization of Large-Scale Wireless Sensor Networks Using Niching Particle Swarm Optimization and Reliable Anchor Selection. *Hindawi Wireless Communications and Mobile Computing*, 1-18.
- Çakıcı, F. N. & Eren, T. (2021). Kablosuz Sensör Ağlarında Parçacık Sürü Optimizasyonu, Sürekli Zaman Karınca Koloni Algoritması ve Girdap Arama Algoritması ile Konum Belirleme. *International Journal of Engineering Research and Development*, 13(2), 719-732.
- Dağdeviren, O. & Akram, V. K. (2017). TinyOS tabanlı telsiz duyurga ağları için bir konumlandırma ve k-bağlılık denetleme sistemi. *Bilişim Teknolojileri Dergisi*, 139–139.
- Dalkılıç, G. & Türkmen, F. (2002). Karınca kolonisi optimizasyonu. *YPBS2002–Yüksek Performanslı Bilişim Sempozyumu*, Kocaeli, Ekim.
- Dariane, A.B. & Moradi, A.M. (2009). Reservoir Operating by Ant Colony Optimization for Continuous Domains (ACO_R) Case Study: Dez Reservoir. *International Journal of Mathematical, Physical and Engineering Sciences*, 3(2), 125-129.
- Doğan, B. & Korürek, M. (2010). EKG Vurularının Sürekli Zaman Karınca Koloni Optimizasyonu Yardımıyla Sınıflandırılması ECG Beat Classification Using Ant Colony Optimization for Continuous Domains. *National Conference on Electrical, Electronics and Computer Engineering*, Bursa, 497-501.
- Doğan B. & Ölmez, T. (2015). A new metaheuristic for numerical function optimization: Vortex Search Algorithm. *Information Sciences*, 293, 125-145.
- Doğan, R. (2018). Kablosuz Sensör Ağları İçin RSSI Değeri İle Yapay Sinir Ağı Yaklaşımlı Konum Bulma. Yüksek Lisans Tezi, Fırat Üniversitesi, Elazığ, Türkiye.
- Doğru, A.S., Temel, B. & Eren, T. (2019). Kablosuz Sensör Ağlarında Konum Belirlemede Parçacık Sürü Optimizasyonu ve Yarasa Algoritması Yöntemlerinin Karşılaştırılması. *Uluslararası Mühendislik Araştırma Ve Geliştirme Dergisi*, 11(3), 793-801.
- Dorigo M., Maniezzo, V. & Colorni, A. (1991). Positive feedback as a search strategy. Dipartimento di Elettronica-Politecnico di Milano, Milan, Italy, Technical Report NO. 91-016

- Eberhart, R. & Kennedy, J. (1995). A New Optimizer Using Particle Swarm Theory. Sixth International Symposium On Micro Machine and Human Science, Nagoya, Japan, 39-43.
- Eren, T. (2016). Graph Invariants for Unique Localizability in Cooperative Localization of Wireless Sensor Networks: Rigidity Index and Redundancy Index, *Ad Hoc Networks*, 44, 32-45, 2016.
- Eren, T., (2017). The effects of random geometric graph structure and clustering on localizability of sensor networks. *International Journal of Distributed Sensor Networks*, 13(12), 1-15.
- Eren, T., (2019). Conditions for Unique Localizability in Cooperative Localization of Wireless ad hoc and Sensor Networks. *Cooperative Localization and Navigation: Theory, Research and Practice*, 31-48, Gao, C.(Ed.), Zhao, G. (Ed.), Fourati, H. (Ed.). Boca Raton: CRC Press (2019).
- Eren, T. Goldenberg, D. & Whiteley, W. (2004). Rigidity, Computation and Randomization in Network Localization. In *Proceedings of the 2004 International Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2004)*, Hong Kong, pp. 2673–2684.
- Freris, N. M., Kowshik, H. & Kumar, P.R. (2010). Fundamentals of Large Sensor Networks: Connectivity, Capacity, Clocks and Computation. *Proceedings of the IEEE*, 98(11), 1828-1846.
- Gangal, V. (2015). Kablosuz Algılayıcı Ağlarda Karınca Koloni Algoritmaları ile Enerji Etkin Rotalamanın İncelenmesi. Yüksek Lisans Tezi, Karadeniz Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Trabzon, Türkiye.
- Goyal, S. & Patterh, M.S. (2015). Flower Pollination Algorithm based Localization of Wireless Sensor Network. 2nd International Conference on Recent Advances in Engineering & Computational Sciences (RAECS), 1-5, doi:[10.1109/RAECS.2015.7453299](https://doi.org/10.1109/RAECS.2015.7453299).
- Güngüneş, R., Ateş, V., Lüy, M. & Eke, İ. (2019). Doğrusal Olmayan Yüklere Sürekli Zaman Karınca Kolonisi Algoritması İle Optimal Değerli Kondansatör Seçimi ve Güç Kontrolü. *Uluslararası Mühendislik Araştırma ve Geliştirme Dergisi*, 11(3), 780-792, doi:[10.291374/umagd.636390](https://doi.org/10.291374/umagd.636390).
- Kannan, A.A., Mao, G. & Vucetic, B. (2006). Simulated annealing based localization in wireless sensor network. *Journal of Computers*, doi:[10.4304/jcp.1.2.15-22](https://doi.org/10.4304/jcp.1.2.15-22).
- Karaboğa, D. (2004). Yapay Zeka Optimizasyon Algoritmaları. Nobel Yayın Dağıtım, 2004.
- Kennedy, J. & Eberhart, R. (1995). Particle Swarm Optimization. *IEEE Int. Conf. Neural Networks*, Perth, Australia, 1942-1948.
- Keskintürk, T. & Söyler, H. (2006). Global Karınca Kolonisi Optimizasyonu. *Gazi Üniversitesi, Mühendislik-Mimarlık Fakültesi Dergisi*, 21(4), 689-698.
- Kızılkaplan, E., Eren, T. & Yalçınkaya, F. (2020). Kablosuz Sensör Ağlarında Konum Belirlemede Sezgisel Algoritmaların Kuantum Davranışları ile Karşılaştırılması. *Uluslararası Mühendislik Araştırma ve Geliştirme Dergisi*, 12(2), 587-602, doi:[10.29137/umagd.746589](https://doi.org/10.29137/umagd.746589).
- Kumar, A., Khosla, A., Saini, J.S. & Singh, S. (2012). Meta-Heuristic Range Based Node Localization Algorithm for Wireless Sensor Networks. *International Conference on Localization and GNSS (ICL-GNSS)*, IEEE, Munich, Germany, 1-7, doi:[10.1109/ICL-GNSS.2012.6253135](https://doi.org/10.1109/ICL-GNSS.2012.6253135).
- Li, J., Andrew, L.L.H., Foh, C.H., Zukerman, M. & Chen, H. H. (2009). Connectivity, coverage and placement in wireless sensor networks. *Sensors*, 9(10), 7664–7693.
- Mácha, T., Stančík, P. & Novotný, V. (2008). Connectivity in a wireless sensor network. *IJCSNS International Journal of Computer Science and Network Security*, 8(12), 382-387.
- Mehrabian, A.R. & Lucas, C. (2006). A novel numerical optimization algorithm inspired from weed colonization. *Ecological Informatics*, 1(4), 355-366.
- Okkan, U., Köse, O., Özsoy, M. & Uysal, H. (2017). Yabani Ot Ve Diferansiyel Evrim Algoritmalarının Aylık Kavramsal Bir Yağış-Akış Modeli Kalibrasyonu Üzerinden Performanslarının İrdelenmesi. *DSİ Teknik Bülteni*, 124, pp. 20-33.

Ojha, V.K., Abraham, A. & Snasel, V. (2017). ACO for Continuous Function Optimization: A Performance Analysis. 14th International Conference on Intelligent Systems Design and Applications, Okinawa, Japan, 28–30 November, 145–150, [doi:10.1109/ISDA.2014.7066253](https://doi.org/10.1109/ISDA.2014.7066253).

Özsağlam, M.Y. & Çunkaş, M. (2008). Optimizasyon Problemlerinin Çözümü için Parçacık Sürü Optimizasyonu Algoritması. Politeknik Dergisi, 11(4), 299-305.

Öztürk, C. (2016). Yapay Arı Koloni Algoritması ile Kablosuz Algılayıcı Ağlarda Lokalizasyon. Erciyes Üniversitesi Bilimsel Araştırma Projeleri Koordinasyon Birimi, Proje No:FBA-12-4029.

Poli, R., Kennedy, J. & Blackwell, T. (2007). Particle swarm optimization. Swarm Intell, 33-57, [doi:10.1007/s11721-007-0002-0](https://doi.org/10.1007/s11721-007-0002-0).

Ramson, S.R.J. & Moni, D.J. (2017). Applications of Wireless Sensor Networks-A Survey. Proceedings of IEEE International Conference on Innovations in Electrical, Electronics, Instrumentation and Media Technology, 325-329, [doi:10.1109/ICIEEIMT.2017.8116858](https://doi.org/10.1109/ICIEEIMT.2017.8116858)

Rashid, H. & Turuk, A.K. (2013). Localization of Wireless Sensor Networks Using a Single Anchor Node. Wireless Personal Communications, 72(2), 975-986, [doi:10.1007/s11277-013-1050-y](https://doi.org/10.1007/s11277-013-1050-y).

Reyes-Sierra, M. & Coello Coello, C.A. (2006). Multi-Objective Particle Swarm Optimizers: A Survey of the State-of-the-Art. International Journal of Computational Intelligence Research, 2(3), 287-308, [doi:10.5019/j.ijcir.2006.68](https://doi.org/10.5019/j.ijcir.2006.68)

Sağ, T. (2020). Çok Merkezli Girdap Arama Algoritması. Düzce Üniversitesi Bilim Ve Teknoloji Dergisi, 8, 1279-1294, [doi:10.29130/dubited.644881](https://doi.org/10.29130/dubited.644881).

Shrivastava, A. & Bharti, P. (2015). Localization Techniques for Wireless Sensor Networks. International Journal of Computer Applications. 16(12), 13-18.

Singh, S.P. & Sharma, S.C. (2015). Range Free Localization Techniques in Wireless Sensor Networks: A Review. Procedia Computer Science, 57, 7-16, [doi:10.1016/j.procs.2015.07.357](https://doi.org/10.1016/j.procs.2015.07.357)

Socha, K. & Blum, C. (2007). An ant colony optimization algorithm for continuous optimization: application to feed-forward neural network training. Neural Computing & Applications, 16, 235-247, [doi:10.1007/s00521-007-0084-z](https://doi.org/10.1007/s00521-007-0084-z).

Socha, K. & Dorigo, M. (2008). Ant colony optimization for continuous domains. European Journal of Operational Research, 185(3), 1155-1173, [doi:10.1016/j.ejor.2006.06.046](https://doi.org/10.1016/j.ejor.2006.06.046).

Tripathi, R.N., Gaurav, K. & Singh, Y.N. (2019). On Partial Coverage and Connectivity Relationship in Deterministic WSN Topologies, 1-4.

Wang, C., Chen, J., Sun, Y. & Shen, X.S. (2009). Wireless Sensor Networks Localization With Isomap. IEEE International Conference On Communications, [doi:10.1109/ICC.2009.5199576](https://doi.org/10.1109/ICC.2009.5199576).

Xiao, J. & Li, L.P. (2011). A hybrid ant colony optimization for continuous domains”, Expert Systems with Applications. 38, 11072-11077.

Yıldırım, G. (2012). Kablosuz Sensör Ağı İle Konum Belirleme ve İzleme. Yüksek Lisans Tezi, Fırat Üniversitesi, Elazığ, Türkiye.