Kablosuz Algılayıcı Ağlarda Devingen Kapsama Sorunu için Evrimsel Algoritma

M. Aykut YİĞİTEL, Tolga TOLGAY, Cem ERSOY
Boğaziçi Üniversitesi, Bilgisayar Mühendisliği Bölümü, İstanbul
aykut.yigitel@boun.edu.tr, tolga.tolgay@boun.edu.tr, ersoy@boun.edu.tr

Özet: Kablosuz Algılayıcı Ağlarda enerjiyi etkili kullanarak algılanacak bölgeyi mümkün olduğunca kapsamak oldukça önemli bir problemdir. Amacımız, algılanacak bölgeyi kapsarken, bu işi mümkün olduğunca az enerji sarfederek yapmaktır. Bildirimizde bu soruna devingen olarak cevap bulan bir evrimsel algoritma önerdik. Aynı problemi Tamsayı Programlama olarak da formüle ettik ve çözdük. Elde ettiğimiz sonuçları ve karşılaştırmaları sunduk.

Anahtar Sözcükler: Kablosuz Agılayıcı Ağlar, Devingen Kapsama Sorunu, Evrimsel Algoritma

An Evolutionary Algorithm For Dynamic Coverage Problem in Wireless Sensor Networks Design

Abstract: Designing an energy efficient topology that extends the network lifetime is a challenging issue in Wireless Sensor Networks (WSN). Our aim is covering the monitoring area as much as we can in an energy efficient manner. In this work, we proposed an evolutionary algorithm for the Dynamic Coverage Problem. We also give an Integer Linear Programming formulation and solved it. We report the computational results and comparisons between solutions.

Keywords: Wireless Sensor Networks, Dynamic Coverage Problem, Evolutionary Algorithms.

1. Giriş

Algılayıcı teknolojisi ve kablosuz iletişimdeki son gelişmeler, Kablosuz Algılayıcı Ağları (KAA) cazibe merkezi haline getirmiştir. Bir KAA düşük fiyatlı ve az enerjili birçok küçük algılayıcıdan oluşur. Algılayıcılar çevrelerindeki fiziksel bilgileri toplayarak daha güçlü ve yetenekli olduğunu varsaydığımız veri toplama merkezine iletirler. Tüm bigiler toplanıp değerlendirildiğinde, gözlenen alanının genel bir resmi oluşturulabilir.

KAA'ların birçok uygulama alanı vardır. Kapalı alanlar için örnek olarak fabrikalardaki karmaşık ekipmanlar veya üretim hatları örnek verilebilir. Hava kirlilik ölçümü, hayvanların gözlemlenmesi, hudut güvenliği, keşif, hedefleme sistemleri, istihbarat, afet bölgelerinin ve engelli hastaların gözlemlenmesi de bu alanlar arasında sayılabilir.

Bilinen kablosuz ağlara nazaran KAA'lar birçok farklılık arzederler ve bu farklılıklar aşılması gereken sorunları da beraberinde getirir. Bu sebeple varolan kablosuz ağ çözümleri KAA'lar için genişletilmeli ya da ona uyarlanmalıdır. Bu farklılıklardan en önemlileri KAA'ların uygulamaya bağımlı olması, enerji kısıtı, yüksek algılayıcı yoğunluğu, kısıtlı bant genişliği ve devingen bir topolojiye sahip olmasıdır.

KAA'ların bu farklı özelliklere sahip olması, enerji etkili ağ protokolleri ve az enerji ile çalışabilen donanımlar geliştirilmesi için pek çok araştırma yapılmasına neden olmuştur. Bu çalışmalarda pek çok kıstas mevcuttur.

(ör: kapsama, enerji tüketimi, güvenilirlik, ağ ömrü, gecikme) Fakat en önemli kısıt enerjidir. Çünkü çoğu zaman algılayıcıların pillerini değiştirmek ya da doldurmak mümkün olmamaktadır. Doğal olarak güneş enerjisiyle veya kinetik enerji ile yeniden doldurabilen algılayılar mevcut fakat henüz kullanım alanları yeterince genişlemiş değildir. Bu nedenle, KAA'ların ömrünü uzatan protokol ve tasarımların geliştirilmesi kaçınılmaz hale gelmiştir.

Asıl amacımız ağın enerji tüketimini en aza indirerek mümkün olan en fazla alanı kapsamaktır. Bu ikisi birbiri ile tezat ölçütlerdir. Çünkü daha çok alanı kapsamak için daha fazla algılayıcıyı aktif hale getirmek gerekir ki, bu da daha çok enerji tüketimine sebep olur.

Biz bu sorunu belirli sayıda periyodlar belirleyerek devingen bir şekilde çözdük. Bu devingenliğin sebebi her algılayıcının enerjilerini orantılı şekilde tüketmesi ve ağ topolojisinde meydana gelebilecek değişikliklere uyarlanabilmesidir. Bu sorun Tamsayı Programlama ile modellenerek çözülebilir ancak işlemler fazla hesaplama gerektirdiğinden uzun sürelerde sonuçlanır. Kabul edilebilir sürede iyi sonuçlar bulmak için evrimsel bir algoritma tasarladık.

Bildiri daha önce yapılan çalışmalar ve temel kavramlar ile devam edecek. Ardından geliştirilen Tamsayı Programlama modeli açıklanacak. 5. kısımda evrimsel algoritma ve genetik işlemler anlatılacak. Hesaplama sonuçları, bildiri sonucu ve geleceğe yönelik yapılacak işlerle makale sonuçlandırılacaktır.

2. İlgili Diğer Çalışmalar

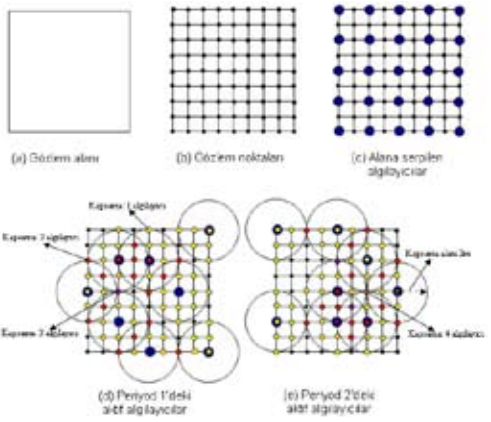
Mümkün olduğunca çok alanı kapsarken, kaynakları etkin kullanarak ağın ömrünü uzatmayı hedefleyen bir çok çalışma yapılmıştır. Vieira ve arkadaşları [1] önerisinde, Voronoi segmantasyon algoritması ile fazladan aktif olan algılayıcıları kapatarak enerji tasarrufu sağlamışlardır. Lin ve Chiu [2] benzetimli tavlama algoritması kullanarak, devingen küme kapsama sorununa çözüm getirmiş ve ağın ömrünü uzatmışlardır.

Nakamura ve arkadaşları [3] KAA'lardaki kapsama ve bağlanırlık sorununa karışık tamsayı programlama ile çözüm aramışlar ve ağın ömrünü uzatmayı başarmışlardır. Quintao ve arkadaşları [4] bu sorun için tamsayı programlama kullanmışlar ve bir evrimsel algoritma geliştirmişlerdir. Ancak evrimsel algoritmanın sonuçlarını her bir algılayıcıya belirli bir ömür biçtiklerinden, yerel arama yöntemiyle uygun hale dönüştürmek zorunda kalmışlardır.

3. Temel Kavramlar

3.1. KAA'larda Kapsama

Algılayıcıların kapsama alanı, etraflarındaki herhangi bir olayı farkedebildikleri bölgedir. Bu bölge R yarıçaplı bir dairedir. Bir KAA'nın kapsama alanı tüm aktif algılayıcıların kapsama alanının birleşimidir.



Şekil 1. Kablosuz Algılayıcı Ağ ve Devingen Kapsama Sorunu Örneği

KAA tarafından gözlemlenecek bölgede, birer metre aralıklarla gözlem noktaları olduğunu varsaydık. Bu varsayım KAA'nın gözlem böl-

gesinde kapsadığı alanı hesaplamak için ortaya atılmıştır. Bu şekilde, gözlem alanı ayrık uzayda değerlendirilebilir. Üsteki şekilde bir örnek mevcuttur.

3.2. KAA'larda Bağlanabilirlik

Wang ve arkadaşları [5] algılama ve iletişim menzili arasındaki ilişkiyi ele almışlardır. Ortaya koydukları sonuçta, algılama menzili iletişim menzilinin yarısından daha az olduğunda; kapsamadan emin olunursa, bağlanabilirlikten de emin olunacağını çıkarmışlardır.

Yapılan bu çalışmanın ışığında, önerimizde bağlanabilirlik değerlendirilmemiştir.

4. Matematiksel Model

Parametreler:

S Algılayıcı ağlar kümesi

D Gözlem noktaları kümesi

T Zaman periyod kümesi

m kapsama duyarlılığı

M_i $i \in S$ algılayıcısının bakım enerjisi

N_j $j \in D$ gözlem noktasının kapsanmama cezası

Model Değişkenleri:

x_{ij}^t eğer i algılayıcısı j gözlem noktasını $t \in T$ periyodunda kapsıyorsa 1, kapsamıyorsa 0.

y_i^t eğer i algılayıcısı $t \in T$ periyodunda aktifse 1, değilse 0.

h_j^t eğer j gözlem noktası $t \in T$ periyodunda kapsanıyorsa 1, kapsanmıyorsa 0.

A_{ij} eğer j gözlem noktası i algılayıcısının kapsama alanındaysa 1, değilse 0.

Amaç Fonksiyon:

$$\min \sum_{i \in S} \sum_{t \in T} M_i \times y_i^t + \sum_{j \in D} \sum_{t \in T} N_j \times h_j^t$$

Kısıtlar:

$$\sum_{i \in S} x_{ij}^t + h_j^t \geq m, \forall j \in D, \forall t \in T \quad (1)$$

$$x_{ij}^t = y_i^t \times A_{ij}, \forall i \in S, \forall j \in D, \forall t \in T \quad (2)$$

$$0 \leq h_j^t \leq m, \forall j \in D, \forall t \in T \quad (3)$$

$$x, y \in \{0,1\} \quad (4)$$

Amaç fonksiyonu, kapsama alanını attırmaya çalışırken enerji tüketimini de azaltmaya çalışır. Kısıt (1)'de eğer bir gözlem noktası en az m tane algılayıcı tarafından kapsanmıyorsa, uygunluk değerine bir kapsamama cezası eklenir. (2)'nci kısıtta; eğer algılayıcı i, gözlem noktası j'yi kapsıyorsa ve periyod t'de aktif ise, x_{ij}^t değeri 1 olmalıdır. Kısıt (3)'de kapsamama cezasının çarpanına en küçük ve en büyük değer aralığı veriliyor. Böylece olası en büyük çarpan kapsama duyarlılığına eşit olur.

5. Evrimsel Algoritma

5.1. Genetik Arama

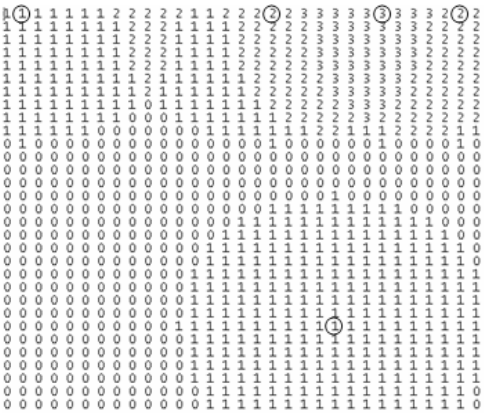
Evrimsel algoritmamızda parametreler ikili kodlanarak kullanılır. Ayrıca tek bir

kromozomda bir çok periyod gösterilebilir. Kromozomlardaki her bir periyod gösterimi, ağdaki algılayıcı sayısı büyüklüğündedir. Genin 1 olması o gene karşılık gelen algılayıcının o periyodda aktif, 0 olması o algılayıcının aynı periyodda pasif olduğu anlamına gelir.

Örneğin 10 algılayıcılı bir ağda, 1,5 ve 8'inci algılayıcıların aktif olması durumunda kromozomun o periyodu için gösterimi şu şekilde olur:

1	0	0	0	1	0	0	1	0	0

Bir periyoddaki aktif algılayıcılar bilindiğinde, bu periyodun kapsama matrisi elde edilebilir. 30x30'luk bir gözlem alanı için algoritmamızın yarattığı kapsama matrisi Şekil-2'de görülebilir.



Şekil 2. Örnek Kapsama Matrisi - 30m x 30m Gözlem Alanı - 5 Aktif Algılayıcı - 10m Algılama Menzili

Verilen örnek algoritma tarafından bulunan en iyi sonucu değil, rasgele seçilmiş bir çözüm adayını göstermektedir. İşaretlenen yerler aktif algılayıcı noktalarını göstermektedir. Gözlem noktalarındaki her bir rakam, o noktayı kapsayan algılayıcı sayısını belirtmektedir.

5.1.1. Başlangıç Popülasyonu

Başlangıç popülasyonu oluşturmak kromozomları ve genlerini yaratmaktır. Her bir gen K olasılıkla birörnek dağılım kullanılarak yaratılır. Kromozomun bir geni yaratılacağı zaman, o algılayıcıyı temsil eden gen K olasılık ile 1 (aktif) olabilir. Bu yaklaşımı kullanarak evrimsel algoritmaya daha enerji etkin kromozomlarla başlayabilir ya da başlangıç popülasyonundaki aktif algılayıcı oranını bu olasılık parametresi ile değiştirebiliriz. Bu parametrenin etkisi hesaplama sonuçları bölümünde gösterilecektir.

5.1.2. Rasgele Mutasyon (RM)

Bu işlemde, en iyisi hariç popülasyonun tüm kromozomlarına uğranır. m olasılık ile bir kromozom seçilir. Daha sonra bu kromozomun rasgele bir periyodu ve geni seçilerek değeri tersine çevirilir.

5.1.3. Aktif Algılayıcı Azaltma Mutasyonu (AAAM)

Bu işlemde bir önceki gibi yapılır. Fakat bir kromozom seçildiğinde tüm periyodları ve genleri gezilir. Eğer gen 1 ise b olasılık ile 0'a çekilir. Amacımız kromozomdaki aktif algılayıcı sayısını azaltmaktır.

5.1.4. Rasgele Eşleme (RE)

Bu işlem birleştirerek yeni kromozomlar oluşturmak için popülasyondan rasgele iki kromozom seçer.

5.1.5. Ağırlıklı Eşleme (AE)

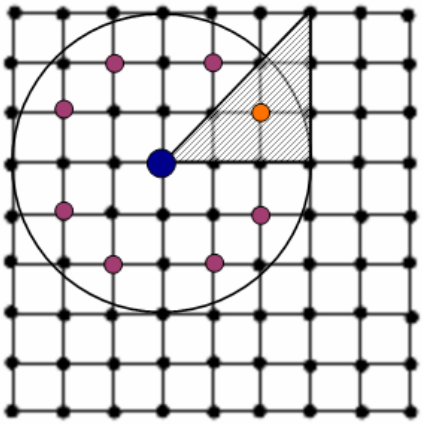
Ağırlıklı Eşleme, amaç fonksiyonda daha iyi değerler elde eden kromozomları eşleyerek daha iyi kromozomlar elde etmek için tasarlanmıştır [6]. Kromozomların ağırlıkları amaç değerlerine göre sıralandıktan sonra elde edilen sıra numaralarına göre verilir. Kullanılabilecek daha detaylı bir diğer yöntem de kromozomları amaç değerlerine göre ağırlıklandırmaktır. Bu yöntemi seçmeme sebebimiz gerçekleştiriminin zor olması ve popülasyon sıralama işleminin çaprazlama için zaten yapılacak olmasıdır.

5.1.6. Çaprazlama
RE ya da AE ile seçilen iki kromozomu şu şekilde çaprazlarız:

- İlk çocuk babanın ilk T/2 periyodunu, geriye kalanını annesinden alır.
- İkinci çocuk annenin ilk T/2 periyodunu, geriye kalanını babasından alır.

Yeni yaratılan kromozomlar popülasyondaki en kötü kromozomların yerine geçer. Eski populasyonun α (kalma oranı) kadarı saklanır geri kalan kısmı (1-α) yenilerle değiştirilir.

5.2. Kapsama Matrisi Oluşturma
Kapsama için Öklit uzaklığını kullandık.



Şekil 3. Kapsama Hesaplama

Örnekte kapsama menzili 3 metre olan bir algılayıcının matrisi oluşturulmak isteniyor. Geleneksel yöntemlerle 49 (7x7) nokta için tek tek kapsama hesabı yapılması gerekiyor. Yoğun ağlarda bu hesaplama algoritmanın karmaşıklığını bir hayli arttırmaktadır. Kısa zamanda çözüme ulaşmak için algoritmamız sadece taralı üçgendeki gözlem noktaları değerlendirilip tüm noktalara yaymaktadır. Böylelikle kapsama hesaplama zamanı yaklaşık 1/5'ine düşürülmüştür.

5.3. Dikey Sıralama
Ortalama ve en kötü olarak O(n log n) karmaşıklığa sahiptir. Algoritmamızda, sıralama yaparken kromozomları bir yerden bir yere kopyalamak yerine kopyalama işlemini oluşturduğumuz dizinde yapıyoruz. Böylece sadece oluşturduğumuz dizini sıralayıp, istediğimiz kromozoma bu dizindeki gösterge ile ulaşıyoruz.

5.4. Algoritma

1. Algılayıcı yeri ilklendirme
2. Popülasyon ilklendirme
3. WHILE en çok yineleme DO
   - 3.1.Çaprazlama için kromozom seç
   - 3.2.Çaprazla,yenileri eskilerle değiştir
   - 3.3.Mutasyon
   - 3.4.WHILE popülasyon sayısı DO
     - 3.4.1.Kapsama matrisi yarat
     - 3.4.2.Kapsanmayan noktaları hesapla
     - 3.4.3.Aktif algılayıcı sayısı hesapla
     - 3.4.4.Amaç değer hesapla
   - 3.5.END WHILE
   - 3.6.Popülasyonu sırala
4. END WHILE

6. Hesaplama Sonuçları

Testlerimizde her biri farklı bir durumu gösteren altı farklı örneği ele aldık. Bu altı örneği iki grupta topladık. Birinci gruptaki örneklerin periyodu iki iken ikinci gruptakilerin periyodunu dört olarak kullandık. Bulduğumuz sonuçları ticari olmayan bir tamsayı programlama aracı olan LP_SOLVE sonuçları ile karşılaştırdık. Testlerimizde kullandığımız parametreleri Tablo 1'de bulabilirsiniz.

Her örnek iki ya da üç tane sınama oyunu içeriyor. Her sınama oyunu için bazı parametreleri değiştirerek bu parametrelerin etkilerini gözlemledik. (Tablo 2) Ayrıca her sınama oyununda genetik operasyonlarımızı da çeşitlendirerek sonuçlar üzerinde büyük etkileri olduğunu gözlemledik.

Parametre	Değeri	Tanımı
T	2 or 4	Periyod Sayısı
m	1	Kapsama Duyarlılığı
ME	10000	Bakım Enerjisi
NC	10000	Kapsamama Cezası
K	20%, 40% or 50%	Başlangıç Popülasyonu Birörnek Dağılım Olasılığı
Π	200	Popülasyon Büyüklüğü
α	50%	Kalma Oranı
μ	10%	Rasgele Mutasyon Olasılığı
γ	15%	Aktif Algılayıcı Azaltma Mutasyonu Olasılığı
#Öteleme	150	Evrimsel Algoritmanın Tekrarlanma Sayısı
#Koşma	25	Her Sınama Oyununun Koşma Sayısı

Tablo 1. Algoritma ve Örnek Senaryo Parametreleri

6.1. Grup 1

Bu grupta farklı konfigurasyonlarda 4 tane örneğimiz var.

6.1.1. Örnek 1

Bu örnek 15 metre kapsama alanına sahip 16 algılayıcı, 60m x 60m'lik bir gözlem alanına sahiptir. Başlangıç Popülasyonu Birörnek Dağılım Olasılığı (K) %20'dir. Birinci sınama oyununda AE – RM kullanılırken ikinci sınama oyununda AE – AAAM kullanılmaktadır.

Bütün algılayıcılar tüm periyodlarda aktif olsa bile gözlem alanının %91.35'i ancak kapsanabiliyor. Bu da bize ağımızda doğal bir kapsama sorununun var olduğunu gösteriyor.

Birinci sınama oyununun sonuçlarını değerlendirdiğimizde, kapsama alanının gayet başarılı olduğunu görüyoruz. İşlem zamanı olarak da eniyileyiciden yaklaşık olarak 21 kat daha hızlı olduğunu görüyoruz.

Öte yandan, ikinci sınama oyunun sonuçları hem birinciye hem de eniyileyiciye göre daha kötüdür. Ancak gerçekleştirimindeki hedef doğrultusunda olabildiğince az sayıda algılayıcı aktif hale getirilmiştir.

Grup	Periyod Sayısı	Örnekler	Algılayıcı Sayısı	Algılama Alanı	Kapsama Menzili	Sınama Oyunu (SO)	Genetik Operasyonlar	K
1	2	1	16	60x60	15	**1**	AE and RM	20%
			16	60x60	15	**2**	AE and AAAM	20%
		2	16	50x50	15	**3**	RE and RM	20%
			16	50x50	15	**4**	AE and RM	20%
		3	16	60x60	20	**5**	RE and RM	20%
			16	60x60	20	**6**	AE and AAAM	20%
			16	60x60	20	**7**	RE and RM with K = 40%	40%
		4	32	60x60	15	**8**	RE and RM	20%
			32	60x60	15	**9**	RE and AAAM	20%
2	4	5	16	60x60	15	**10**	RE and RM	20%
			16	60x60	15	**11**	AE and RM	20%
		6	16	60x60	15	**12**	RE and RM with K = 50%	50%
			16	60x60	15	**13**	AE and RM with K = 50%	50%

Tablo 2. Sınama Oyunları

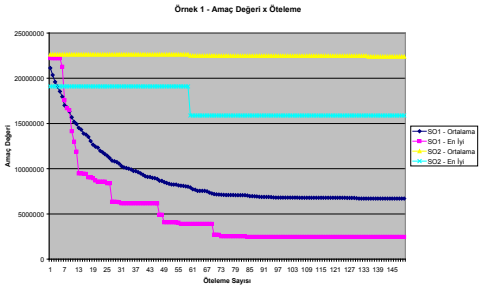
	K.N	% Kapsama	AKT	İşleme Zamanı	Amaç Değeri
LP_S	666,96	91,03	22,4	332916,28	6893600
SO1	645,8	91,32	23,32	15636,44	6691200
SO2	2227,72	70,06	12,48	11393,92	22402000

Tablo 3. Örnek 1'in Sonuçları

	K.N	% Kapsama	AKT	İşleme Zamanı	Amaç Değeri
LP_S	234,48	95,4	17,6	252560,32	2520800
SO3	291,72	94,39	20	11343,32	3117200
SO4	240,84	95,3	17,6	12678,24	2584400

Tablo 4. Örnek 2'nin Sonuçları

Şekil 4'de görüldüğü gibi ikinci sınama oyunumuzun en iyi sonucu yerel minimuma takılmıştır. Bunun sebebi ağımızın seyrek olması ve bu sınamada AAAM kullanmış olmamızdır.



Şekil 4. AE – RM ve AE – AAAM Karşılaştırması

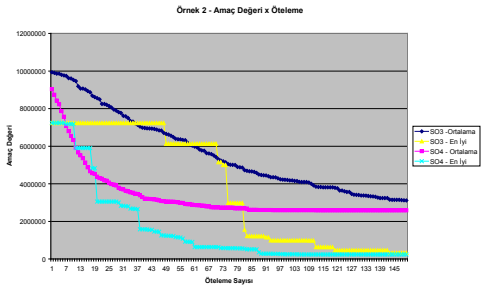
6.1.2. Örnek 2

Bu örnek 15 metre kapsama alanına sahip 16 algılayıcı, 50m x 50m'lik bir gözlem alanına sahiptir. Başlangıç Popülasyonu Birörnek Dağılım Olasılığı (K) %20'dir. Üçüncü sınama oyununda RE – RM kullanılırken dördüncü sınama oyununda AE – RM kullanılmaktadır.

Bu örneğimizde tüm algılayıcılar aktif olduğunda %95.4'lük bir kapsama elde ediyoruz. SO4 neredeyse tüm aktif kapsama ile aynı sonuca ulaşmıştır. Bunun yanı sıra LP_SOLVE ile aynı sayıda algılayıcı aktif ederek, ondan sadece %2 daha kötü bir kapsama sağlamasına rağmen çözümü çok daha hızlı üretmiştir.

SO3 ve SO4'ü karşılaştırdığımızda SO3'ün daha hızlı çalıştığını ancak SO4'e göre daha kötü kapsadığını görüyoruz. Bunun sebebi SO4'ün daha iyi kromozomları seçen AE kullanmasıdır.

Şekil 5'de görüldüğü gibi SO4 SO3'e nazaran daha hızlı düşüşe geçiyor. Ayrıca her ikisinin de en iyi sonuçları hemen hemen aynı değerlere sahiptir. Bu nedenle eğer yeterli öteleme yapılırsa ortalama sonuçlarının da aynı değere doğru gideceği söylenebilir.



Şekil 5. RE – RM ve AE – RM Karşılaştırması

6.1.3. Örnek 3

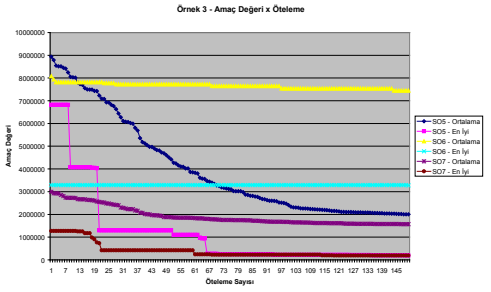
Bu örnek 20 metre kapsama alanına sahip 16 algılayıcı, 60m x 60m'lik bir gözlem alanına sahiptir. Başlangıç Popülasyonu Birörnek Dağılım Olasılığı (K) %20'dir. Beşinci sınama oyununda RE – RM kullanırken altıncı sınama oyununda AE – AAAM kullanılmaktadır. Yedinci sınama oyunu ise RE – RM kullanırken K değerini de %40 olarak alır.

	K.N	% Kapsama	AKT	İşleme Zamanı	Amaç Değeri
LP_S	170,56	97,70	14,96	699523,92	1855200
SO5	180,72	97,57	19,36	19809	2000800
SO6	733,2	90,14	11,8	14991	7450000
SO7	138,56	98,13	18,76	21203	1573200

Tablo 5. Örnek 3'ün Sonuçları

Bu sonuçlar da bize RM'ye nazaran AAAM'nin daha az algılayıcı aktif ettiğini ve böylece daha kötü kapsama sağladığını göstermiştir. Bu örnekteki en önemli sonuçlardan bir tanesi SO7'nin LP_SOLVE'a göre çok daha iyi bir kapsama sağlamış olmasıdır.

SO7 ve SO5'in arasındaki tek fark K değerinin %40 olmasıdır. K değerinin yarattığı etki Tablo 5'de açıkça görülebilir. SO7, SO5'e nazaran %3 daha az algılayıcı aktif ederken %23 daha iyi kapsamaya ve %21 daha iyi uygunluk değerine sahiptir. SO7'nin tek sorunu %7 daha yavaş olmasıdır. Bunun sebebi ise başlangıç aktif algılayıcı sayısının SO5'e göre daha fazla olmasıdır.



Şekil 6. RE – RM ve AE – AAAM Karşılaştırması, K'nın Etkisi

6.1.4. Örnek 4

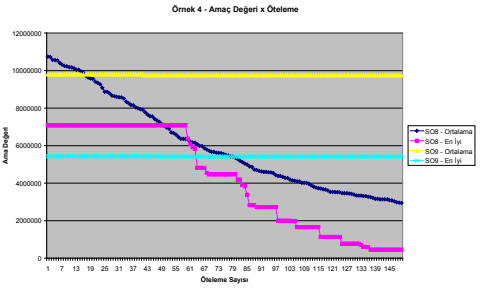
Bu örnek 15 metre kapsama alanına sahip 32 algılayıcı, 60m x 60m'lik bir gözlem alanına sahiptir. Başlangıç Popülasyonu Birörnek Dağılım Olasılığı (K) %20'dir. Sekizinci sınama oyununda RE – RM kullanırken dokuzuncu sınama oyununda RE – AAAM kullanılmaktadır.

	K.N	% Kapsama	AKT	İşleme Zamanı	Amaç Değeri
LP_S	183,24	97,53	23,16	982813,8	2064000
SO8	262,88	96,46	31,64	16718	2945200
SO9	953,72	87,18	21,04	12143	9747600

Tablo 6. Örnek 4'ün Sonuçları

Sonuçlarımı LP_SOLVE ile karşılaştırdığımızda algoritmamızın daha kötü sonuçlar verdiğini gördük. Sekizinci sınama oyununun kapsama oranında LP_SOLVE'a ile arasında %8'lik bir fark vardır. SO9'da yine AAAM'den beklenildiği şekilde aktif algılayıcı sayısı aşağılara çekilmiştir.

Bu örneğin birinci örnekten tek farkı 32 algılayıcı bulunmasıdır. Sonuçlarımıza göre RM seyrek ağlarda daha başarılı sonuçlar vermektedir. Aynı şekilde AAAM'ye baktığımızda ağlar yoğunlaştıkca sonuçları iyileşmiştir.



Şekil 7. RE – RM ve RE – AAAM Karşılaştırması

6.2. Grup 2

Bu grupta 2 tane benzer örneğimiz var. Bu iki örneğin tek farkı birinin K değerini %20, diğerinin ise %50 olarak kullanmasıdır.

Her iki örnekte de 15 metre kapsama alanına sahip 16 algılayıcı, 60m x 60m'lik bir gözlem alanına serpilmiştir.

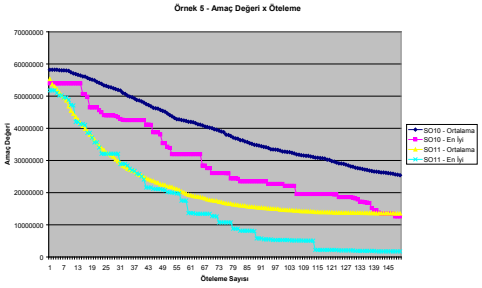
6.2.1. Örnek 5

Onuncu sınama oyununda RE – RM kullanırken dokuzuncu sınama oyununda AE – RM kullanılmaktadır.

	K.N	% Kapsama	AKT	İşleme Zamanı	Amaç Değeri
LP_S	1501,6	89,91	44,32	1280252,4	15459200
SO10	2506,24	83,16	38,52	24827	25447600
SO11	1307,44	91,21	45,52	30474	13529600

Tablo 7. Örnek 5'in Sonuçları

AE kullanan SO11, LP_SOLVE ve SO10'a göre çok daha başarılı sonuçlar vermiştir. Ayrıca SO11'de LP_SOLVE ile aynı sayıda algılayıcı aktif edilmesine rağmen kapsama alanı çok daha iyidir.



Şekil 8. K %20 iken Algoritma Sonuçları

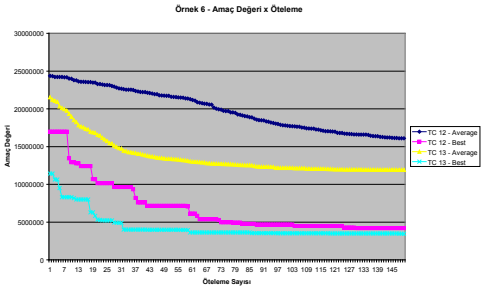
6.2.2. Örnek 6

Onikinci sınama oyununda RE – RM kullanırken onüçüncü sınama oyununda AE – RM kullanılmaktadır.

	K.N	% Kapsama	AKT	İşleme Zamanı	Amaç Değeri
LP_S	1501,6	89,91	44,32	1280252	15459200
SO12	1561,24	89,51	47,64	30474	16088800
SO13	1143,76	92,31	47,92	32158	11916800

Tablo 8. Örnek 6'nın Sonuçları

K olasılığı büyüdüğünde hem RE – RM hem de AE – RM daha iyi sonuçlar vermektedir. SO12 LP_SOLVE ile yaklaşık aynı değerlere ulaşmıştır. SO13'de ise %8 fazla algılayıcı aktif ederek %23 daha iyi uygunluk değerine ulaşılmıştır.



Şekil 9. K %50 iken Algoritma Sonuçları

K değerini %20'den %50'ye çektiğimiz zaman SO10 için %37,7, SO11 için %12,51 daha iyi kapsama elde ediyoruz. *K* değerinin %50 olduğu sınama oyunlarında başlangıç uygunluk değerlerinin çok daha iyi olduğunu böylece daha iyi sonuçlar elde edildiğini gözlemliyoruz.

7. Sonuçlar

KAA'lardaki devingen kapsama sorununu ele aldık. Problemi sunduk ve bunun için bir matematiksel model önerdik. Bu matematiksel modeli ticari olmayan bir tamsayı programlama aracı ile çözdük.

Aynı sorunu çözmek için evrimsel bir algoritma geliştirip aldığımız sonuçları tamsayı programlama aracı sonuçları ile karşılaştırdık.

Önerdiğimiz algoritmanın diğer çözümlere nazaran çok daha hızlı çalıştığını gördük. Problemimiz devingen kapsama olduğundan sonuçları olabildiğince hızlı almak durumundayız. Bizim algoritmamız sadece hızlı olmakla kalmayıp aynı zamanda yeterince iyi sonuçlar vermektedir.

Geleceğe yönelik olarak daha yoğun ağlarda evrimsel algoritma parametrelerini değiştirerek sınamalar yapmak istiyoruz. Bu şekilde parametrelerin yoğun ağlardaki etkisini gözlemleyebiliriz. Ayrıca bu işi bir KAA yönetim çatısına uyarlamak ilginç bir deneyim olabilir.

Kaynaklar

[1] Vieira, M.A.M., Vieira, L.F.M., Ruiz, L.B., Loureiro, A.A.F., Fernandes, A.O., Nogueira, J.M.S. "Scheduling Nodes in Wireless Sensor Networks: A Voronoi Approach." LCN '03: Proceedings of the 28th Annual IEEE International Conference on Local Computer Networks, IEEE (2003) 423

[2] Lin, F.Y.S. Chiu, P.L. "A simulated annealing algorithm for energy-efficient sensor network design", Third International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks, 2005. WIOPT 2005. page(s): 183- 189

[3] Nakamura, F.G. Quintao F.P., Menezes, G.C., Mateus, G.R. "An Optimal Node Scheduling for flat Wireless Sensor Networks" ICN 2005 - International Conference on Networking, IEEE 2005

[4] Quintao, F. P. ; Nakamura, F. G. ; Mateus, G. R. . "Evolutionary Algorithm for the Dynamic Coverage Problem Applied to Wireless Sensor Networks Design". IEEE Congress on Evolutionary Computation (CEC 2005), Edimburgo. Proceeding of the IEEE Congress on Evolutionary Computation, 2005.

[5] Wang, X., Xing, G., Zhang, Y, Lu, C., Pless, R., Gill, C. "Integrated Coverage and Connectivity Configuration in Wireless Sensor Networks." SenSys '03: Proceedings of the 1st International Conference on Embedded Networked Sensor Systems, ACM Press, 2003 28-39

[6] Haupt, R.L., Haupt, S.E. Practical Genetic Algorithms, Second Edition. John Wiley & Sons, Inc. (2004)

[7] LP_SOLVE 5.5.0.10, Non-commercial Mixed Integer Linear Programming Solver. http://lpsolve.sourceforge.net/5.5/