

PAPER DETAILS

TITLE: Tek Atlamali Kablosuz Algılayıcı Ağlarda Yeni Bir Hızlı Ağa Katılım Algoritması

AUTHORS: Musa ÇIBUK

PAGES: 72-83

ORIGINAL PDF URL: <https://dergipark.org.tr/tr/download/article-file/498086>

Araştırma Makalesi / Research Article

Tek Atlamalı Kablosuz Algılayıcı Ağlarda Yeni Bir Hızlı Ağ Katılım Algoritması

Musa ÇIBUK^{1*}

¹Bitlis Eren Üniversitesi, Bilgisayar Mühendisliği, Bitlis

Öz

Kablosuz Algılayıcı Ağlar'da, algılayıcı düğümlerin veri iletmeye başlayabilmeleri için öncelikle bir ağa katılmaları gerekir. Ağ katılmada yavaş kalınması, özellikle gerçek zamanlı uygulamalarda sorunlara yol açar. Diğer taraftan, eğer bir düğüm ağa katılmak için çok aceleci davranırsa, çarpışmalara, ağ performansının düşmesine ve toplam enerji tüketiminin artmasına neden olacaktır. Dolayısıyla Kablosuz Algılayıcı Ağlar için algılayıcı düğümün ağa bağlanma sürecinin çabuk ve etkili olması çok önemlidir.

Bu çalışmada, Tek Atlamalı Kablosuz Algılayıcı Ağlar için ortam erişim katmanında, hızlı ve efektif bir ağ katılım algoritması (FNJ) önerilmiştir. Önerilen algoritmanın kavramsal yapısı ve çalışma mantığı açıklanarak başarımleri ortaya konulmuştur. Daha sonra önerilen algoritmanın benzetimi yapılarak elde edilen bulgular, benzer diğer çalışmalar ile kıyaslanmış ve algoritmanın başarımleri test edilmiştir. Sonuç olarak yapılan kıyaslamalarda önerilen algoritmanın kayda değer başarımleri sergilediği görülmüştür.

Anahtar kelimeler: Kablosuz Algılayıcı Ağlar, Ağ Katılım, Algoritma, Benzetim

A New Fast Network Joining Algorithm for Single-Hop Wireless Sensor Networks

Abstract

In Wireless Sensor networks, firstly, sensor nodes must be joined in a network before they can begin transmitting data. Slow network joining, especially in real-time applications, leads to major problems. On the other hand, if a node is too hasty to join the network, it will cause collisions, decrease the performance of the network, and increase total energy consumption. Therefore, it is very important for the Wireless Sensor Networks that the sensor node's network join process is quick and effective.

In this study, a fast and effective network joining algorithm (FNJ) was proposed in the media access layer for single-hop wireless sensor networks. The proposed algorithm's conceptual structure and operational logic were explained and performance criteria were introduced. The results obtained by simulation of the proposed algorithm were compared with other similar studies and the performance of the algorithm was tested. As a result, comparisons show that the proposed algorithm performs remarkably well.

Keywords: Wireless Sensor Networks, Joining Network, Algorithm, Simulation

*Sorumlu yazar: mcibuk@beu.edu.tr

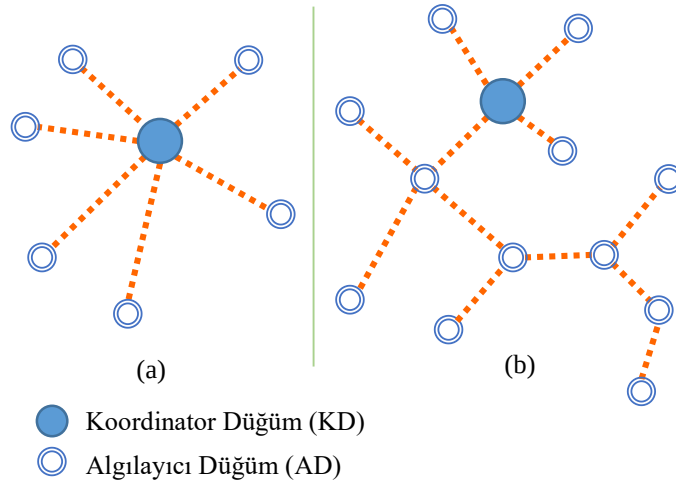
Geliş Tarihi: 03/05/2018 Kabul Tarihi: 29/06/2018

1. Giriş

Teknolojide son yıllarda meydana gelen hızlı gelişmeler sonucunda, fiziksel olay ve nicelikleri gözlemlemek amacıyla, bilgi toplama, işleme ve işlediği bilgiyi başka yerlere iletebilme kabiliyetlerine sahip ve algılayıcı olarak adlandırılan çok küçük boyutlu cihazlar ortaya çıkmıştır. Algılayıcılar ve bunların oluşturdukları ağlar, günümüz bilim dünyasında çok hızlı gelişen ve üzerinde en çok araştırma yapılan konulardan biri haline gelmiştir [1].

Kablosuz Algılayıcı Ağlar (KAA) çok sayıda algılayıcı düğümün (AD) bir araya gelerek oluşturdukları bir veri toplama ekosistemidir. Bu ağlar çevresel [2] ve askeri uygulamalar [3, 4] başta olmak üzere günümüzde pek çok alanda [5–9] kendine yer edinmiş ve hızla yaygınlaşmaktadırlar. Bu yaygınlaşma ile birlikte nitelikli veri toplama işlemleri için düğüm sayılarında ciddi derecede artışlar gözlemlenmektedir [10]. Günümüzde yüzlerce ve hatta binlerce düğümden oluşmuş ağlara rastlamak mümkündür. Düğüm sayılarındaki artış, ağdaki çarpışmaları ve enerji tüketimindeki artışı da beraberinde getirmektedir. Bu ağlardaki en büyük enerji tüketimi safhalarından biri de ağa katılım yani organizasyon safhasıdır. Bu aşamada özellikle düğümler ağdaki koordinatörlere bağlanmak ve kendilerini ağa dâhil etmek için bir çaba içerisine girerler. Ağdan verimli sonuç alınabilmesi için KAA'ları oluşturan düğümlerin hızlı ve efektif bir şekilde ağa katılıp verileri aktarmaya başlaması çok önemlidir.

KAA'larda bağlantı durumuna göre veri aktarımı genellikle Şekil 1'de gösterildiği üzere tek atlamalı veya çok atlamalı olmak üzere iki şekilde olur.



Şekil 1. Bağlantı Durumuna Göre KAA'lar: a) Tek Atlamalı KAA, b) Çok Atlamalı KAA

Tek atlamalı KAA'larda ağdaki tüm düğümler, koordinatör düğüm (KD)'e doğrudan bağlanmaya çalışırlar. Bu tip senaryolarda ağdaki düğüm sayısı arttıkça KD'nin iş yükü ve harcadığı enerji dramatik olarak artmaktadır. Bu da KD'nin yavaşlamasına ve ağa katılmalarda gecikmelere neden olacaktır. KD'deki ağa katılmaları kontrol eden algoritma çok hızlı cevap veren bir yapıya sahip olmalıdır ki ağ hızlı bir şekilde organize olabilsin. Aynı zamanda AD'lerin ağa katılım aşamasında takip edeceği metodun da çok iyi analiz edilmesi gerekmektedir[11].

Çok atlamalı KAA'larda ise AD'ler, doğrudan KD'ye erişemediklerinden, ağa katılma ve veri gönderme işlerini başka düğümler aracılığıyla yaparlar. Bu aşamada genellikle röle mekanizmaları kullanılır [12–15]. Bu tip yapılarda, ağdaki KD'ler daha az enerji harcayarak organizasyonu sağlar. Ancak AD'ler üzerinde fazladan bir iş yükü oluşur.

Tüm bunların ışığında, özellikle çok düğümlü KAA'larda ağa katılım algoritmalarının, ağ performansını doğrudan etkileyen faktörlerden biri olarak karşımıza çıktığı görülmektedir. KAA'lar ile ilgili olarak literatüre bakıldığında, genellikle düğümlerin enerji verimliliği [16–19] ve elde edilen verilerin merkeze uygun yollardan yönlendirilerek iletilmesi [20, 21] konularının daha çok ele alındığı görülmektedir. Yine literatürdeki çalışmaların önemli bir çoğunluğu ağırlıklı olarak “Ağ Ömrünü

Uzatma (Network Lifetime Maximization)” konuları üzerine yoğunlaşmaktadır[22]. Oysaki günümüzde, daha çok KAA’lar üzerinden gerçek zamanlı veri iletim uygulamalarına doğru bir eğilimin [23, 24] olduğu görülmektedir. Gerçek zamanlı veri iletimi gerektiren KAA uygulamalarında; özellikle düğümlerin çok hızlı bir şekilde ağa katılma ve kopmadan ağda veri aktarımını devam ettirme konuları önemlidir. Bu tür özellikler gürbüz bir katılım algoritması sağlanabilir. Böylece KAA’nın başarımı ve esnekliği artırılabilir.

2. Materyal ve Metot

Araştırmacılar geliştirdikleri yöntem ve çalışmalarını test edebilmek için öncelikli olarak daha pratik ve kolay ulaşılabilir olması nedeni ile benzetim araçlarından faydalanırlar. KAA’lar için yapılan geliştirme ve çalışmalarda da pek çok bilim insanı ağ benzetim araçlarını kullanmaktadır [25, 26]. Böylece fiziksel olarak gerçekleştirmesi ve geliştirilmesi uzun ve maliyetli olan süreçler azaltılmış olur. Ağ benzetim araçları, görsel veya gerçek zamanlı izleme özelliklerinden ziyade dağıtılmış protokol veya bazı özel algoritmaların geçerliliğine ya da performansına önem vermektedir. Geliştirilen ağ benzetim araçlarının önemi ve kalitesi, ağ ortamlarında yaygın olarak kullanılan algoritma ve protokolleri desteklemesine bağlıdır [27].

Bu çalışmada KAA’lar için hızlı ve esnek bir ağa katılma algoritması geliştirilmiş ve benzetimi yapılmıştır. Benzetim ortamı olarak Riverbed (Opnet) Modeller [28] programı kullanılmıştır. Opnet, bir ağ ortamı üzerinde tasarım, modelleme, performans analizi ve trafik izleme gibi birçok işlemi çok ayrıntılı olarak yapabilen bir programdır. Program araştırmacıların kendi geliştirdikleri ağ protokollerini test etme olanaklarına da sahiptir.

Makale boyunca kullanılan semboller, teknik kısaltmalar ve anlamları Tablo 1’de sunulmuştur.

Tablo 1. Çalışmada kullanılan sembol ve kısaltmalar

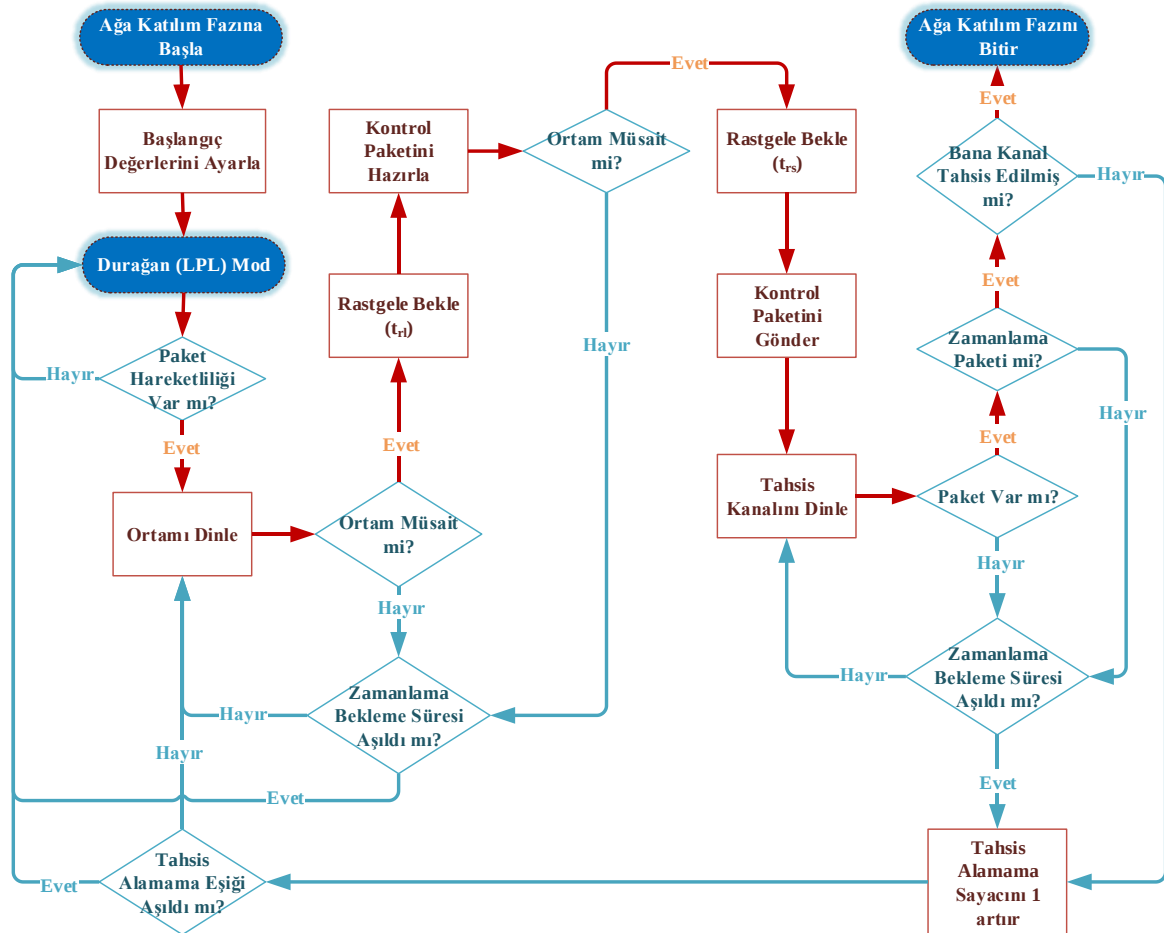
Kısaltma	Anlamı
<i>KAA</i>	Kablosuz Algılayıcı Ağ
<i>KD</i>	Koordinatör Düğüm
<i>AD</i>	Algılayıcı Düğüm
<i>LPL</i>	Düşük Güçte Dinleme
<i>P_{sch}</i>	Zamanlama Paketi
<i>P_{ctrl}</i>	Kontrol Paketi
<i>t_{rl}</i>	Dinleme Rastgele Bekleme Süresi (sn)
<i>t_{rs}</i>	Gönderme Rastgele Bekleme Süresi (sn)
<i>W_t</i>	Bekleme Süresi (sn)
<i>W_{sch}</i>	Zamanlama Paketi Bekleme Süresi (sn)
<i>Ch_{ctrl}</i>	Kontrol Kanalı
<i>Ch_{sch}</i>	Zamanlama Kanalı
<i>BW_{chctrl}</i>	Kontrol Kanalı Bantgenişliği (bps)
<i>NS</i>	Tahsis Alamama Sayacı
<i>NS_{max}</i>	Tahsis Alamama Sayacı Eşik Değeri
<i>NS_{trhs}</i>	Tahsis Alamama Eşik Süresi (sn)
<i>N</i>	Ağa katılmak isteyen Düğüm Sayısı
<i>N_{max}</i>	Maksimum Düğüm Sayısı
<i>C</i>	Çarpışma Sayısı
<i>t_{sctrl}</i>	Kontrol Paketi Gönderme Süresi (sn)
<i>S_{sctrl}</i>	Kontrol Paketi Uzunluğu (bit)
<i>R_{ncc}</i>	Düğüm Başına Düşen Çarpışma Miktarı
<i>t_{join}</i>	Katılım Fazı Süresi (sn)
<i>t_{start}</i>	Katılım Fazı Başlangıç Zamanı (sn)
<i>t_{jlast}</i>	En son düğümün Ağa Katılma Zamanı (sn)
<i>V_{nj}</i>	Düğüm Katılım Hızı (nps)
<i>P_j</i>	Katılım Fazında Harcanan Toplam Güç (W)

Makalenin bundan sonraki kısımlarında önce geliştirilen katılım algoritmasından bahsedilmiştir. Daha sonra geliştirilen algoritma ile ilgili temel kavramsal ifadeler ve başarımlar kriterleri

üzerinde durulmuştur. Sonraki bölümde geliştirilen algoritmanın benzetim çalışmalarından bahsedilerek 3. Bölümde benzetimde elde edilen bulgulara ve bunların irdelemesi yapılmıştır. Son olarak 4. Bölümde sonuç ve önerilere yer verilmiştir.

2.1. Katılım Algoritması

Bu çalışmada tek atlamalı KAA'lar için CSMA-CA[29] tabanlı hızlı ve gürbüz bir ağa katılma algoritması (FNJ) önerilmiştir. Önerilen algoritmanın akış diyagramı Şekil 2'de gösterildiği gibidir.



Şekil 2. Ağa Katılma Algoritması Akış Diyagramı

Bu çalışmadaki yaklaşımda AD'ler ağı katılma isteği olan kontrol paketlerini (P_{ctrl}) çekişme tabanlı bir kanal olan kontrol kanalından (Ch_{ctrl}) gönderirler. KD'nin AD'lerden gelen katılım isteklerini kabul ettiğini gösteren zamanlama paketleri (P_{sch}) ise farklı bir kanal olan zamanlama kanalından (Ch_{sch}) gönderilir. AD'lerin kontrol paketi göndermek için ayrı kanal kullanması, KD'nin zamanlama (P_{sch}) paketleri göndermek için ayrı kanal kullanması algoritmanın önemli ve etkili bir üstünlüğüdür. Ayrıca ağı katılan bir AD, çekişmeli olan kontrol kanalından çıkarak kendisine tahsis edilen kanalı kullanmaktadır. İlgili AD ağına bağlı kaldığı müddetçe bir daha kontrol kanalını kullanmamaktadır. Böylece ağı katılıp kontrol kanalından çıkan her AD, bu kanalı rahatlatarak ağı katılmaya istekli diğer AD'lere daha yüksek katılma şansı sunmaktadırlar. Bu durum diğer AD'lerin ağına daha hızlı katılımını sağlamakta ve bu aşamadaki olası çarpışmaları engellemektedir.

Algoritmada başlangıçta tüm düğümler gerekli başlangıç değerlerini ayarlayarak Durağan (LPL) durumunda beklemeye geçerler. Bu durumda iken ortamda herhangi bir hareketlilik olmadığı müddetçe AD'ler herhangi bir aktivite göstermez ve böylece gereksiz enerji harcamalarından kaçınmış olurlar. AD'ler ortamda herhangi bir paket hareketliliği gözlemler ise ortam dinlemesine başlarlar. Eğer ortam müsait ise algoritma gereği rastgele bir süre (t_{rl}) bekler ve ağa dâhil olma isteğini taşıyacak olan

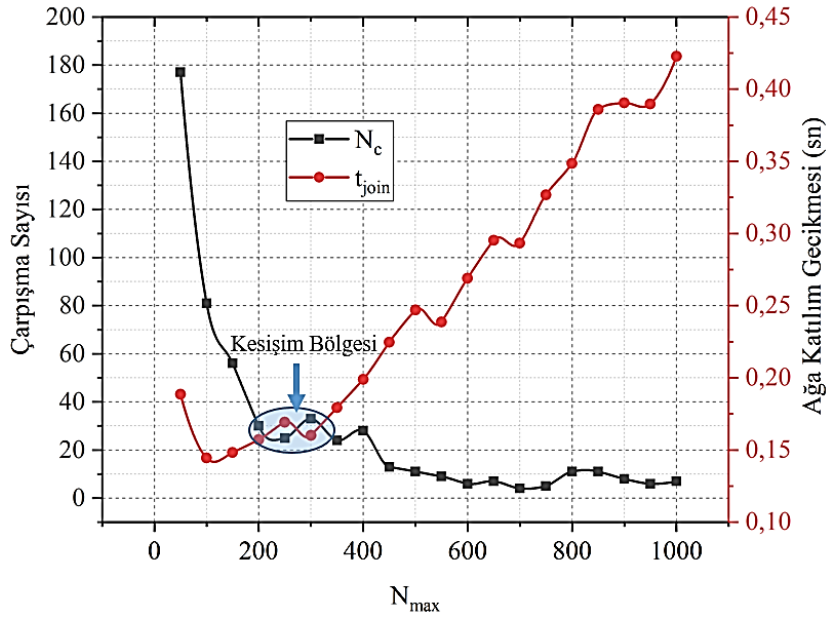
kontrol paketini (P_{ctrl}) hazırlar. Paketi göndermeden önce tekrar bir ortam dinlemesi yaparak, ortamın müsait olup olmadığına bakarlar. Eğer ortam müsait ise yine rastgele bir süre (t_{rs}) bekleyerek istek paketini kontrol kanalından (Ch_{ctrl}) gönderirler. Eğer ortam müsait değil ise AD, bekleme (W_t) süresinin W_{sch} eşik değerini aşıp aşmadığını kontrol eder. Bu süre de aşılmış ise AD LPL durumuna geri döner. Kontrol paketini (P_{ctrl}) gönderen AD artık zamanlama kanalını (Ch_{sch}) dinlemeye başlar ve kendisine bir zamanlama paketi (P_{sch}) gelmesini bekler. Gelen her uygun olmayan zamanlama paketi için tahsis alamama sayacı (NS) bir artırır. NS sayacı maksimum değeri (NS_{max}) aştığında bu sefer tahsis alamama eşiği (NS_{trsh}) kontrol edilir. Bu eşik değer de aşılmış ise LPL durumuna, aşılmamış ise katılım sürecini yeniden başlatmak üzere ortam dinlemesine geçilir. Eğer bir zamanlama paketi gelmiş ve düğüm bu paket içerisinde kendine tahsis edilen kanal bilgisini elde etmiş ise artık ağı katılmış ve ağı katılım fazı başarı ile tamamlanmış olur.

Algoritma CSMA-CA[29] mantığıyla hareket etmektedir ve çarpışmalardan kaçınma temeline dayanmaktadır. Bu bağlamda algoritmadaki t_{rl} ve t_{rs} sürelerinin rastgeleselliği algoritmanın başarımında önemli rol oynamaktadır. Bu süreler saniye cinsinden olup hesaplanmasında aşağıdaki denklem 1 ve 2 de verilen formüller kullanılmıştır.

$$t_{rl} = rnd_{uni}(N_{max}) * t_{sctrl} \quad (1)$$

$$t_{rs} = rnd_{exp}\left(\frac{t_{sctrl}}{100}\right) \quad (2)$$

Denklem 1'deki N_{max} değeri ağdaki olası maksimum düğüm sayısını ifade etmektedir. Bu değer büyüdükçe rastgele üretilen sayıların çakışma ihtimali düşmekle beraber düğümlerin ağı katılmalarındaki bekleme sürelerini uzatmakta bu da düğümlerin ağı hızlı bir şekilde katılmalarını olumsuz etkilemektedir. N_{max} değerinin çok küçük olması ise düğümlerin daha girişken bir şekilde ağı katılma çabalarını artırmaktadır. Bu da erişim çarpışmalarının artmasına ve ağın genel performansının düşmesine neden olmaktadır. Dolayısı ile bu değer, ağın performansı göz önünde bulundurularak uygun bir şekilde belirlenmesi gerekmektedir. Optimal N_{max} değerinin tespiti için değişen N_{max} değerlerine karşılık, ağdaki çarpışma sayısı ve AD'lerin ağı katılma süreleri üst üste bindirilerek kesişim noktaları bulunmalıdır.



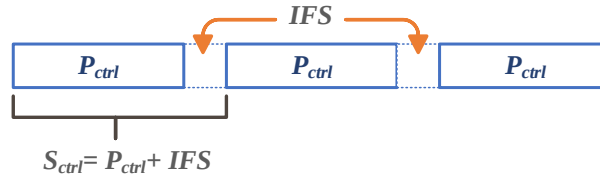
Şekil 3. N_{max} değerinin çarpışmalar ve düğümlerin ağı katılma gecikmeleri üzerine etkisi

Şekil 3'te bu çalışmada önerilen ağa katılma algoritmasını kullanan, $N = 200$ düğümlü bir senaryo için değişen N_{max} değerlerine karşılık, ağa katılım fazında meydana gelen çarpışma ve düğümlerin ağa katılma gecikmesi gösterilmektedir. Buradaki ağa katılma gecikmesi ifadesi tüm AD'lerin ağa katılması için geçen süreyi, diğer bir deyişle katılım fazı süresini (t_{join}) ifade eder. N_{max} için en uygun değerler bölgesi Şekil 3'te gösterildiği üzere eğrilerin kesişim bölgesidir. N_{max} değerinin 215, 266 ve 320 değerleri için eğriler kesişmektedir. Buradan hareketle N_{max} değeri, bu çalışmada, orta kesişme noktası olan $266 = 1.33N$ olarak seçilmiş ve N_{max} bağımlı tüm hesaplamalar buna göre yapılmıştır.

Denklem 1'deki $rnd_{uni}()$ fonksiyonu homojen dağılımlı, denklem 2'deki $rnd_{exp}()$ fonksiyonu ise üstel dağılımlı rastgele sayı üretici fonksiyonlarıdır. t_{sctrl} ifadesi bir kontrol paketi (P_{ctrl}) gönderilmesi için gerekli zamanı saniye cinsinden ifade eder. t_{sctrl} 'nin hesaplanmasında ise aşağıda verilen denklem 3 kullanılır.

$$t_{sctrl} = \frac{S_{ctrl}}{BW_{chctrl}} \quad (3)$$

Denklem 3'teki BW_{chctrl} ifadesi kontrol kanalının (Ch_{ctrl}) bps cinsinden bant genişliğini ve S_{ctrl} ifadesi ise bir kontrol paketinin (P_{ctrl}) gönderilmesi için gerekli süreyi bit cinsinden ifade etmektedir. S_{ctrl} ifadesi Şekil 4'te gösterildiği gibi hesaplanmaktadır.



Şekil 4. Kontrol Paketleri Gönderim Süresi

2.2. Başarım Kriterleri

Bu çalışmada önerilen algoritmanın başarımının analiz edilebilmesi ve diğer çalışmalar ile kıyaslanabilmesi için aşağıdaki kriterler ortaya konmuş ve bu kriterlere göre analizler yapılmıştır;

Katılım Fazı Süresi (t_{join}) :

Bu kriter, saniye cinsinden, senaryodaki ilk düğümün ağa katılma anından en son düğüm ağa katılana kadar geçen süredir. Özellikle gürbüz ve esnek bir KAA'da bu sürenin kısa olması ağın hızlı bir şekilde veri aktarmaya başlamasında belirleyici bir bilgidir. Kıyaslamalarda bu sürenin düşük olması başarımın bu kriter açısından daha iyi olduğu anlamına gelir. Bu değer t_{start} katılım fazının başlama anı ve t_{jlast} en son düğümün ağa katıldığı anı göstermek üzere denklem 4'te gösterildiği gibi belirlenir.

$$t_{join} = t_{jlast} - t_{start} \quad (4)$$

Çarpışma Sayısı (C) :

Bu kriter, katılım fazı süresi (t_{join}) boyunca ağda meydana gelen tüm çarpışmaların toplamı olarak ifade edilir. Çarpışmaların minimum hatta hiç olmaması istenir. Dolayısı ile bu değer küçük olması kıyaslamalarda başarım açısından daha iyi olarak değerlendirilir.

Düğüm Katılım Hızı (V_{nj}) :

Bir KAA'da katılım fazında saniyede kaç düğümün ağa dâhil edilebildiğini gösteren değer olup birimi nps (node per second) olarak ifade edilir. Bu değer yüksek olması demek ağa katılım algoritmasının kısa sürede çok daha fazla düğümü ağa katabileceği anlamına gelir. Bu değer ağdaki toplam düğüm sayısı N katılım fazı süresi t_{join} olmak üzere denklem 5'te gösterildiği gibi hesaplanır.

$$V_{nj} = \frac{t_{join}}{N} \quad (5)$$

Katılım Gücü (P_j) :

KAA'lardaki en önemli başarımlar kriterlerinden biri de harcanan güçtür. Çünkü doğaları gereği KAA'lardaki KD ve AD'lerin enerjisi sınırlıdır. Önerilen algoritma için harcanan güç miktarları bu anlamda belirleyici bir başarımlar kriteri olacaktır. Katılım gücü, katılım fazı süresi boyunca her bir düğüm tarafından ağa katılım için harcanan güçlerin toplamı olarak ifade edilir ve denklem 6'da gösterildiği gibi hesaplanır.

$$P_j = \sum_{i=1}^N P_j(i) \quad (6)$$

2.3. Benzetim Çalışması

Bu çalışmada önerilen ağa katılma algoritmasının Riverbed (Opnet) Modeler ortamında benzetimi yapılmıştır. AD sayıları daha önceki çalışmalarla karşılaştırılması ve düğüm sayısındaki artışa karşın ağı benzetim ortamında göstereceği performansın ölçülebilir olması açısından 25, 50, 100, 200, 250, 400, 500, 600, 750, 800, 1000, 1250 ve 1500 olarak seçilmiştir. Düğümler her senaryo için alana rastgele dağıtılmıştır. Tüm senaryolarda sadece bir koordinatör düğüm (KD) kullanılmış olup tüm senaryolarda aynı noktaya yerleştirilmiştir. Benzetim çalışması, bu makalede önerilen ağa katılma algoritması dikkate alınarak, özellikle düğümlerin ağa katılma fazına yoğunlaşmıştır. Çalışmadaki temel amaç düğümlerin hızlı bir şekilde ağa dâhil olup bir daha kopmamak üzere veri aktarımına başlaması olduğundan dolayı benzetim süresi tüm düğümlerin ağa dâhil olduğunun gösterilebileceği bir süre ile sınırlandırılmıştır. Düğüm sayıları dışında tüm senaryolar özdeş olup benzetim çalışmasında kullanılan ortam parametreleri Tablo 2 gösterildiği gibidir.

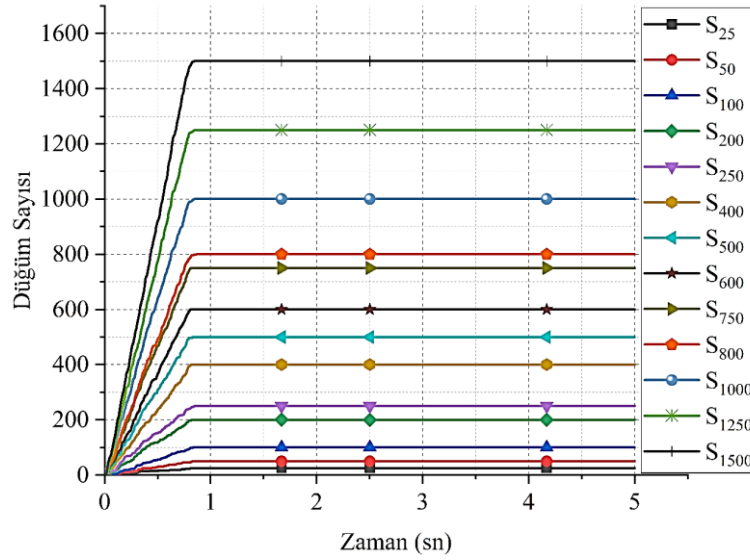
Tablo 2. Önerilen algoritma için benzetim ortamı parametreleri

Parametre	Parametre Değeri
Ortam Modeli	Free Space
Ortam Veri Tipi	USGS DEM
Ortam Ölçüleri	(100m x 100m) Kare
Düğüm Yerleşimi	Rastgele
Kontrol Paketi Boyutu (S_{ctrl})	32+7 = 39 bit
Radyo Bantgenişliği (BW_{ctrl})	99 kHz
Sinyal Kapsama Alanı (r)	150 m
Radyo Gücü (P_{tx})	27 mW
Modülasyon Türü	qam64_ber_snr
Benzetim Süresi	5 sn
Max. Düğüm Sayısı (N_{max})	2000
NS_{max}	2
NS_{trsh}	0.033 sn

3. Bulgular ve Tartışma

Bu çalışmada önerilen FNJ algoritmasının farklı düğüm sayılarındaki ağ senaryoları (S_{25} , S_{50} , S_{100} , S_{200} , S_{250} , S_{400} , S_{500} , S_{600} , S_{750} , S_{800} , S_{1000} , S_{1250} , S_{1500}) için ayrı ayrı benzetimleri yapılmıştır. Yapılan benzetimler sonucunda elde edilen değerler MDF algoritması [11] ile karşılaştırılarak önerilen FNJ algoritmasının performans gelişimi ortaya konmuştur.

Benzetim sonuçları sırayla incelenecek olursa, ilk olarak düğüm sayılarına göre ağa katılım gecikmesi grafiği sonuçları Şekil 5'te gösterildiği gibidir.



Şekil 5. Düğüm sayılarına göre ağa katılım gecikmesi

Şekil 5'te gösterildiği üzere, tüm senaryolar için düğümler 1 saniyenin altında bir sürede ağa katılmaktadırlar. Ağa dâhil olan AD'ler benzetim boyunca ağdan kopmadan işlemlerine devam etmektedirler. Tüm senaryolarda ağa katılım sabit bir ivme ile gerçekleşmiştir. Bu da önerilen algoritmanın düğüm sayısından bağımsız olarak dengeli bir ivme ile çalıştığını göstermektedir. Düğümlerin senaryolara göre ağa katılım durumları ve çarpışma sayıları Tablo 3'de gösterilmiştir.

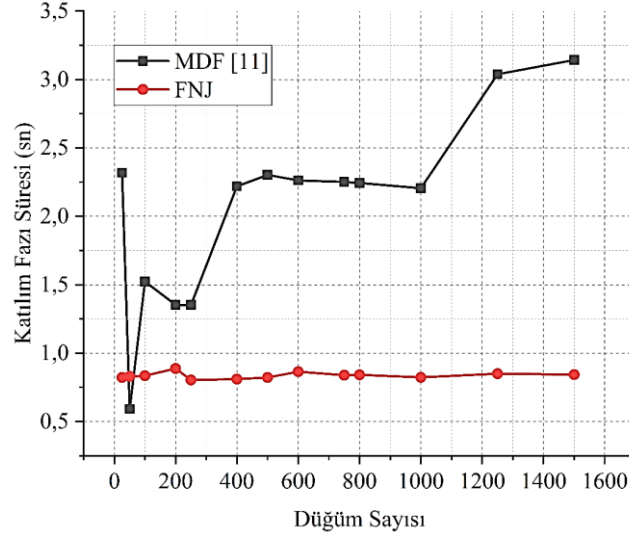
Tablo 3. Düğümlerin senaryolara göre ağa katılım durumları ve çarpışma sayıları

Senaryo	Düğüm Sayısı (N)	Katılım Fazı Süresi (t_{join})	Çarpışma Sayısı (C)
S_{25}	25	0.822	0
S_{50}	50	0.829	0
S_{100}	100	0.835	2
S_{200}	200	0.87	5
S_{250}	250	0.804	13
S_{400}	400	0.81	20
S_{500}	500	0.822	52
S_{600}	600	0.864	31
S_{750}	750	0.839	53
S_{800}	800	0.841	73
S_{1000}	1000	0.823	82
S_{1250}	1250	0.85	151
S_{1500}	1500	0.843	249

Tablo 3 incelendiğinde, tüm senaryolarda katılım fazı sürelerinin birbirine çok yakın değerler olduğu görülmektedir. Bu durum önerilen algoritmadaki stabil davranış ile açıklanabilir. Ancak düğüm sayısı arttıkça ağdaki çarpışmalarında arttığı gözlemlenmektedir. En yoğun çarpışmanın olduğu S_{1500} senaryosunda dahi düğüm başına ortalama 0.166 gibi bir çarpışma değeri düşmektedir. 1500 düğümlü

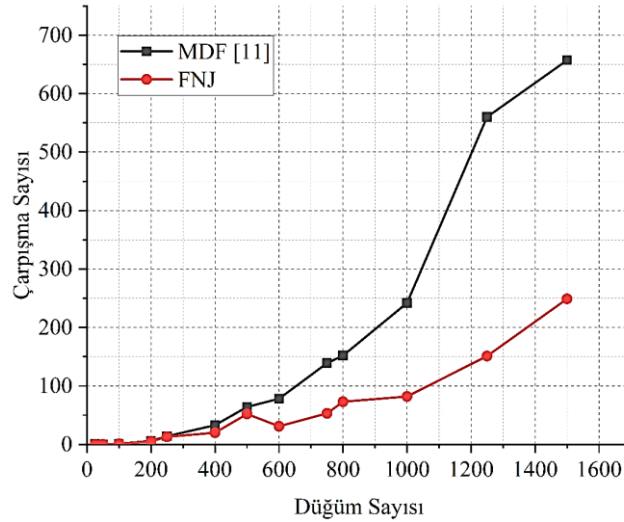
bir ağ düşünüldüğünde bu oranın gayet iyi bir değer olduğu söylenebilir. Tüm senaryolar dikkate alındığında ortalama katılım fazı süresi 0.8348 sn olmaktadır.

Düğüm sayısına bağlı olarak katılım fazı sürelerindeki değişimlerin diğer çalışmalarla karşılaştırması Şekil 6'da gösterildiği gibidir.



Şekil 6. Düğüm sayısına bağlı olarak Katılım Fazı Sürelerindeki değişimlerin karşılaştırması

Şekil 6'da görüldüğü üzere önerilen algoritmanın katılım fazı süreleri MDF'ye göre çok daha düşüktür. Bu da önerilen FNJ algoritmasının öncekine göre tüm düğümleri çok daha kısa sürede ağa kattığını göstermektedir. MDF çalışmasında bir düğümün ortalama ağa katılma süresi 0.012182 sn iken önerilen FNJ algoritması ile bu süre ortalama 0,005748 sn olarak gerçekleşmiştir. Böylece bu anlamda önerilen çalışma ile %52.82 daha iyi bir başarımla elde edilmiştir.

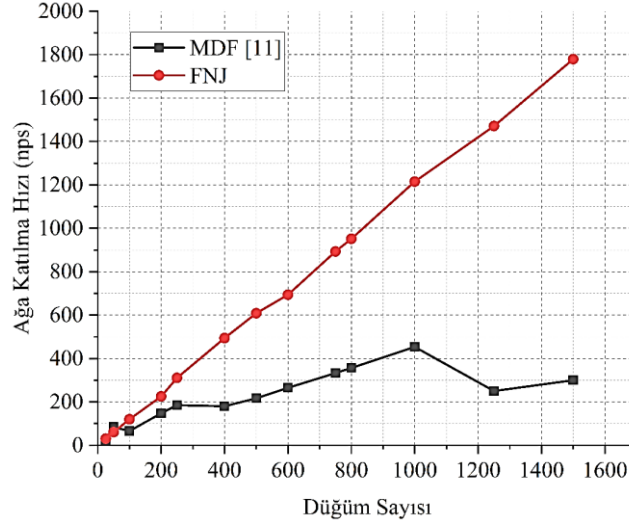


Şekil 7. Katılım fazında meydana gelen toplam çarpışmaların karşılaştırması

Katılım fazında meydana gelen toplam çarpışmaların karşılaştırması ise Şekil 7'de sunulmuştur. Şekil 7'de gösterildiği üzere önerilen FNJ algoritması ile çarpışma sayılarında, MDF'ye kıyasla önemli düşüşler elde edilmiştir. 500 düğümlü senaryoya kadar her iki çalışma da benzer performans sergilerken 500 düğümden sonra MDF'deki çarpışma miktarlarında ciddi artışlar söz konusudur. Öte yandan önerilen FNJ algoritmasının çarpışma sayılarında çok daha düşük ivmeli bir artış olmuştur. Düğüm sayıları arttıkça çalışmalar arasındaki farkın gittikçe açıldığı bariz bir şekilde görülmektedir. MDF çalışmasında tüm senaryolar toplamında düğüm başına düşen çarpışma miktarı 0.262222 iken önerilen

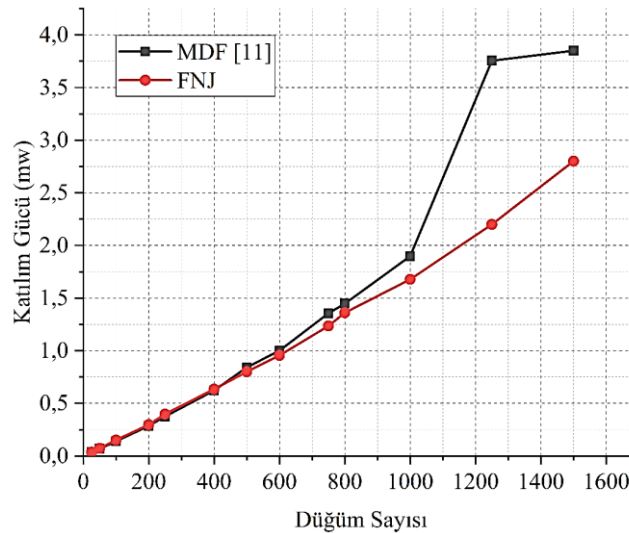
algoritma ile bu değer 0.098451 olarak gerçekleşmiştir. Böylece önerilen FNJ algoritması ile MDF çalışmasına göre çarpışmalar %62.46 oranında azaltarak daha iyi bir başarımla elde edilmiştir.

Ağa katılan düğümlerin sayısına bağlı olarak düğümlerin ağa katılma hızları, MDF ve önerilen FNJ algoritması için Şekil 8’de gösterildiği gibidir.



Şekil 8. Ağ katılma hızlarının karşılaştırılması

Şekil 8’de gösterildiği üzere önerilen algoritma, MDF’ye göre daha iyi bir başarımla sergilemiştir. Düğüm sayısı arttıkça MDF’nin başarımla sınırlı kalmakla beraber en büyük başarımla 1000 düğümlü senaryoda göstermiştir. MDF’nin başarımla sınırlı kalması Şekil 7’deki çarpışma grafiği dikkate alındığında anlamlı olmaktadır. Öte yandan bu çalışmada önerilen FNJ algoritması, düğüm sayısının artışına paralel bir başarımla sergilemektedir. Ağ katılma hızı kararlı bir artış göstermektedir. MDF çalışmasında tüm senaryolar toplamında ortalama hız 245.27 nps iken bu değer önerilen FNJ algoritması ile 681.39 nps olarak gerçekleşmiştir. Böylece önerilen FNJ algoritması ile MDF çalışmasına göre ağ katılma hızı %277.81 oranında artırılarak daha iyi bir başarımla elde edilmiştir.



Şekil 9. Ağ katılma fazında düğümlerce harcanan toplam güç miktarları

Son olarak KD ve AD’lerin ağa katılma fazında harcadıkları toplam güçlerin MDF ve FNJ algoritmaları için karşılaştırması Şekil 9’da gösterilmiştir. Her iki çalışmada da başlangıçta harcanan güç miktarları birbirine çok yakın iken, 500 düğümlü senaryodan itibaren MDF’nin güç harcama değerlerinde FNJ’ye göre artışlar olduğu görülmektedir. Özellikle 1000 düğümlü senaryodan itibaren

eğriler arasındaki fark ciddi oranda artmaktadır. Enerji harcama konusunun KAA'ların ömrü açısından çok önemli bir parametre olduğu düşünülürse FNJ algoritmasının bu konuda MDF'ye göre daha başarılı olduğu görülmektedir. Tüm senaryolar toplamında düğüm başına düşen ortalama harcanan güç miktarı $2.11 \mu W$ iken bu değer FNJ algoritması için $1.7 \mu W$ 'dir. Sonuç olarak bu çalışmada önerilen FNJ algoritması ile MDF'ye göre %19.5'lik bir başarımla elde edilmiştir.

4. Sonuç ve Öneriler

Bu çalışmada, tek atlamalı KAA'lar için AD'lerin hızlı ve etkili bir şekilde ağa katılmalarını sağlamak üzere, yeni bir ağa katılma algoritması (FNJ) geliştirilmiştir. Geliştirilen algoritma ile düğümlerin hızlı bir şekilde ağa katılarak ağda kalmaları, kopma durumlarında ise yeniden hızlı bir şekilde ağa bağlanmaları amaçlanmıştır. Geliştirilen algoritmanın Opnet Modeler ortamında farklı düğüm yoğunluklarında benzetim çalışması yapılmış ve elde edilen bulgular MDF ağa katılım algoritması ile kıyaslanmıştır.

Elde edilen bulgular ışığında, bu çalışmada önerilen FNJ algoritmasının MDF algoritmasına kıyasla katılım fazı sürelerinde %52.82, çarpışmaları azaltma açısından %62.46, düğümlerin ağa katılma hızında %277.81 ve son olarak KD ve AD'lerin ağa katılma fazında harcadıkları toplam güç değerlerinde de %19.5 daha başarılı olduğu görülmüştür.

Geliştirilen algoritmanın özellikle çok yüksek düğüm sayılarında bile stabil davranması ve 1500 düğümü 1 saniyenin altında ağa dâhil edebilmesi yönüyle dikkate değerdir. Tüm bunların yanında harcanan güç miktarları ve çarpışma değerlerinde de olumlu yönde iyileştirmeler söz konusudur. Sadece tek atlamalı KAA'lar için uygun olması ve N_{max} parametresine sıkı bağımlılığı, algoritmanın dezavantajları olarak görülebilir.

Gelecek çalışmalarda, geliştirilen FNJ algoritmasının Mobil KAA'lar için optimize edilmesi ve performans iyileştirmeleri hedeflenmektedir. Ayrıca bu algoritmanın KAA'ların özel bir durumu olan Doğrusal KAA'lara uygulanması düşünülmektedir.

Kaynaklar

1. Ata O., Uçar E., Balık H.H. 2011. Kablosuz Algılayıcı Ağlarda Kullanılan Teknoloji ve Protokoller Üzerine Bir inceleme, İstanbul Aydın Üniversitesi Derg, 3(12):51-68.
2. Mao X., Miao X., He Y., Li X-Y., Liu Y. 2012. CitySee: Urban CO2 monitoring with sensors, 2012 Proceedings IEEE INFOCOM (IEEE), pp 1611-1619.
3. Winkler M., Tuchs K., Hughes K., Barclay G. 2008. Theoretical and practical aspects of military wireless sensor networks, J Telecommun Inf Technol, 2:37-45.
4. Hussain M.A., Khan P., Kwak kyung Sup. 2009. WSN Research Activities for Military Application, Adv Commun Technol 2009 ICACT 2009 11th Int Conf, 01:271-274.
5. Khan P., Hussain M.A., Kwak K.S. 2009. Medical Applications of Wireless Body Area Networks, Int J Digit Content Technol its Appl, 3(3):185-193.
6. Aktaş F., Çeken C., Erkan K., Yıldırım M. 2011. Kablosuz Algılayıcı Ağlar Kullanılarak Birinci Dereceden Ölü Zamanlı Bir Sistemin Denetimi, 6th International Advanced Technologies Symposium (IATS'11), pp 251-256, Elazığ.
7. Cibuk M., Balık H.H. 2011. A novel solution approach and protocol design for bio-telemetry applications, Adv Eng Softw, 42(7):513-528.
8. Dilay Y., Soy H., Bayrak M. 2012. Hassas Tarımda Kablosuz Algılayıcı Ağların Kullanımı ve Uygulama Alanlarının İncelenmesi, Iğdır Üniversitesi Fen Bilim Enstitüsü Derg, 2(2):21-26.
9. Türker G.F., Tarımer İ. 2011. Türkiye'de Kablosuz Algılayıcı Ağlar ile Yapılan Teknolojik Uygulamalar Üzerine Bir İnceleme, Akademik Bilişim'11 - XIII. Akademik Bilişim Konferansı, pp 75-81, Malatya.
10. Ökdem S., Karaboğa D. 2007. Kablosuz Algılayıcı Ağlarında Yönlendirme Teknikleri, Akademik Bilişim'07 - IX. Akademik Bilişim Konferansı, pp 409-415.

11. Çibuk M., Arı D., Çınar H. 2016. A New Multi-Channel Algorithm Of Join The Network For Single Hop Multi-Node Sensor Networks, International Engineering, Science and Education Conference - INESEC 2016, pp 1–3, Diyarbakır.
12. Çibuk M., Arı D., Ağgün F. 2017. Network Joining Algorithm with a New Relay Mechanism for Vehicle Networks, International Advanced Technologies Symposium (IATS17), pp 3451–3458, Elazığ.
13. Arı D., Çibuk M., Ağgün F. 2017. Effect of Relay-Priority Mechanism on Multi-hop Wireless Sensor Networks, Bitlis Eren Univ J Sci Technol, 7(2):145–153.
14. Souissi M., Meddeb A. 2017. Modelling of clustering with relay nodes in wireless sensor networks. 2017 IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC) (IEEE), pp 1–6.
15. Djenouri D., Bagaa M. 2017. Energy-Aware Constrained Relay Node Deployment for Sustainable Wireless Sensor Networks, IEEE Trans Sustain Comput, 2(1):30–42.
16. Anchora L., Capone A., Mighali V., Patrono L., Simone F. 2014. A novel MAC scheduler to minimize the energy consumption in a Wireless Sensor Network, Ad Hoc Networks, 16:88–104.
17. Jha M.K., Pandey A.K., Pal D., Mohan A. 2011. An energy-efficient multi-layer MAC (ML-MAC) protocol for wireless sensor networks, AEU - Int J Electron Commun, 65(3):209–216.
18. Vullers R., Schaijk R., Visser H., Penders J., Hoof C. 2010. Energy Harvesting for Autonomous Wireless Sensor Networks, IEEE Solid-State Circuits Mag, 2(2):29–38.
19. Xie L., et al. 2015. Multi-Node Wireless Energy Charging in Sensor Networks, IEEE/ACM Trans Netw, 23(2):437–450.
20. Huang X., Zhai H., Fang Y. 2008. Robust Cooperative Routing Protocol in Mobile Wireless Sensor Networks, IEEE Transactions on Wireless Communications, 5278–5285.
21. Tang H., Cao J., Sun C., Lu K. 2013. REA-MAC: A low latency routing-enhanced asynchronous duty-cycle MAC protocol for wireless sensor networks, J Cent South Univ, 20(3):678–687.
22. Yetgin H., Cheung K.T.K., El-Hajjar M., Hanzo L. 2017. A Survey of Network Lifetime Maximization Techniques in Wireless Sensor Networks, IEEE Commun Surv Tutorals, 19(2):828–854.
23. Shukeri N.M., Rahim M.A.A., Wan T-C. 2016. Empirical testing of prototype real-time multi-hop MAC for Wireless Sensor Networks. 2016 6th IEEE International Conference on Control System, Computing and Engineering (ICCSCE) (IEEE), pp 10–15.
24. Ngo-Quynh T., Tran-Quang V., Nguyen-Trung Q. 2016. A low-latency communication protocol for target tracking in wireless sensor networks, EURASIP J Wirel Commun Netw, 2016 (1): 33.
25. Li X., Peng M., Cai J., Yi C., Zhang H. 2016. OPNET-based modeling and simulation of mobile Zigbee sensor networks. Peer-to-Peer Netw Appl, 9 (2): 414–423.
26. Biswas S. 2017. Simulation Model of Beacon Enabled 802.15.4 Networks with OPNET Modeler, 2017 International Electrical Engineering Congress (IEECON) (IEEE), pp 1–4.
27. Gürtürk U., Daş R. 2017. A Review of Network Simulation and Modeling Tools, Afyon Kocatepe Univ J Sci Eng, 17 (2): 516–536.
28. Riverbed. 2018. OPNET Technologies – Network Simulator. Available at: <https://www.riverbed.com/gb/products/steelcentral/opnet.html> (Erişim Tarihi: 02.05.2018).
29. Wikipedia.org. 2018. Carrier-sense multiple access with collision avoidance (CSMA/CA). Available at: https://en.wikipedia.org/wiki/Carrier-sense_multiple_access_with_collision_avoidance (Erişim Tarihi: 02.05.2018).