

Nesnelerin İnterneti İçin Bir Protokol : IETF 6TiSCH

A protocol for Internet of Things : IETF 6TiSCH

Sedat Görmüş*, Ahmet Faruk Yavuz**

*Bilgisayar Mühendisliği Bölümü, Karadeniz Teknik Üniversitesi, Trabzon, Türkiye

**Mavi Alp Bilgi Teknolojileri, Trabzon Teknokent, Trabzon, Türkiye

*{sedatgormus}@ktu.edu.tr, **{ahmetyavuz}@mavialp.com

Özetçe —Günümüde, Kablosuz Duyurga Ağları (KDA) endüstriyel internet, evde sağlık hizmetleri (m-Health) ve Akıllı Şebekeler gibi uygulamaların ortaya çıkmasıyla önem kazanmıştır. KDAlar birbirinden bağımsız olarak çalışan ve duyargalar üzerinden topladıkları uygulama verilerini kablosuz olarak merkezi bir sisteme raporlarlayan düğümlerden oluşur. Endüstriyel uygulamalar dahil olmak üzere bir çok uygulama düşük ve kestirilebilir bir ağ içi gecikmeye ihtiyaç duyarlar. KDAlarda kestirilebilir ağ gecikmesine sahip olmak geleneksel ortam erişim ve yönlendirme protokolleri kullanıldığında en iyi senaryoda bile oldukça zordur. Düşük güç kullanımı, yüksek kararlılık ve kestirilebilir gecikmeye sahip KDAlar gerçekleştirmek adına IETF 6TiSCH adı verilen bir protokol ortaya koydu. Bu protokol zaman paylaşımlı ve kanal atlamalı (TSCH) ortam erişim protokolüyle IPv6 adreslemeyi kullanır ve gerekli kaynakları dinamik olarak ağdaki düğümlere uygulama gereksinimine göre atar.

Bu bildiriye, tarafımızdan Contiki işletimi için gerçekleştirilen 6TiSCH protokolü ve bu protokol için geliştirilen bir dağıtık kaynak ayrımı mekanizması tanıtılmıştır. Protokol performansı farklı ağ senaryoları için Cooja simülöründe test edilip, performans sonuçları slotted Aloha protokolüyle karşılaştırılmıştır¹.

Anahtar Kelimeler—IoT, M2M, 6Tisch, Kablosuz Duyurga Ağları.

Abstract—In recent years, there is an increased interest toward Wireless Sensor Networks (WSNs) with the introduction of application areas such as industrial internet, m-Health and Smart Grid. WSNs consist of sensors with wireless capability operating autonomously and reporting application specific parameters to a central entity. Many applications including industrial applications require the WSN to have a low and deterministic delay. Having a deterministic network delay in a WSN is challenging in the best case when traditional Medium Access Control (MAC) protocols are employed. To enable low power, high reliability and deterministic WSNs, IETF recently proposed 6TiSCH protocol which uses time slotted channel hopping (TSCH) MAC with IPv6 addressing. This protocol dynamically assigns bandwidth resources to the nodes in the network according to the application requirements.

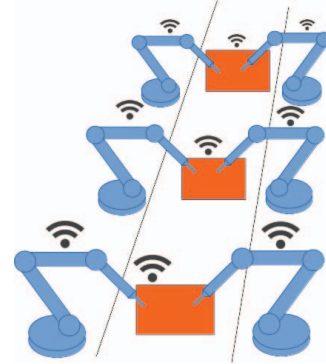
This paper describes an implementation of 6TiSCH protocol in Contiki OS and a distributed scheduling mechanism for this protocol. We evaluate the protocol performance using Cooja simulator for different networking scenarios and compare it to that of a slotted Aloha type access mechanism.

Keywords—IoT, M2M, 6Tisch, Wireless Sensor Networks.

¹Bu çalışma TÜBİRAK 2232 tersine beyin göçü kapsamında 115c116 numaralı proje üzerinden desteklenmiştir.

I. GİRİŞ

Endüstriyel internet, akıllı şebekeler, akıllı şehir çözümleri gibi uygulamaların popülerlik kazanmasına paralel olarak KDA'lara olan ilgi artmaktadır. Sayılan uygulamalarda duyurga modüller aracılığıyla toplanan veri belli bir merkezde toplanabileceği gibi, çeşitli cihazların uzaktan kontrolü de yapılabilir. Uygulamanın niteliğine göre tolere edilecek ağ gecikmesi, paket kayıp miktarı ve enerji bütçesi değişebilir. Örneğin, Şekil 1'deki endüstriyel kontrol sistemi robotların koordineli olarak üretim bandında çalışmalarını sağlar. Beklenmedik bir durumda sistemin durdurulması yada parametrelerinin kararlı bir şekilde değiştirilmesi gerekebilir. Bu uygulama düşük ve kestirilebilir gecikmeye sahip bir kablosuz haberleşme yapısı gerektirir. Genel olarak kablosuz duyurga ağları rastgele erişim kullanırlar (CSMA/CA) ve bu tür erişim yöntemleri kullanıldığında düğümler kanala erişim için çekişirler. Bu çekişme düğümlerin yüksek miktarda yada belli aralıklarla yüksek miktarda (bursty traffic) veri gönderdikleri durumda yüksek ve kestirilmesi zor gecikmelere sebep olabilir.



Şekil 1: Örnek bir endüstriyel internet çözümü

IEEE kanal erişimini kararlı ve kestirilebilir hale getirmek amacıyla var olan IEEE 802.15.4 protokolüne zaman paylaşımlı ve kanal atlamalı MAC yöntemini eklemiştir [1]. Bu yeni MAC IEEE 802.15.4e TSCH olarak adlandırılır. Bu yöntemle, duyurga düğümleri kanala beklemeden, kendilerine ayrılmış zaman dilimlerinde ve belirlenen frekans kanallarında erişirler. 802.15.4e TSCH protokolü, iki komşu düğümün yüksek kararlılıkla haberleşmesi için yeterlidir. Fakat, bir çok uygulama çoklu sekmeli haberleşme yöntemiyle desteklenen ağları kullanmayı gerektirir. Bu nedenle, 802.15.4e TSCH MAC protokolü ile düşük güçlü duyurga ağları için tasarlanan

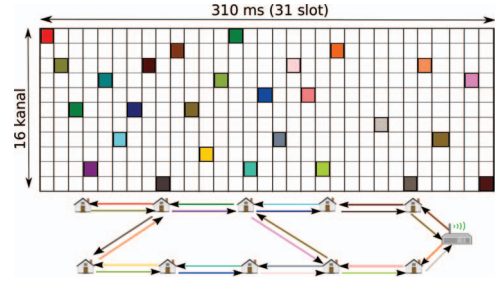
6LoWPAN [2] protokolünü birleştirecek standartlar geliştirilmiştir. IETF 6TiSCH [3] protokolü yüksek kararlılığa sahip kablosuz duyarga ağlarını gerçeklemek adına IETF tarafından geliştirilmektedir. Bu standartlaşma çabasının öncelikli hedefi IPv6 tabanlı bir yönlendirme protokolünün zaman paylaşım ve kanal atlamalı bir MAC protokolü ile uyumlu çalışmasını sağlamaktır.

Uygulama/CoAP/PCE/Tracking	
TCP/UDP/ICMPv6/IPv6	RPL
6LoWPAN	
6Top	
IEEE 802.15.4e TSCH MAC	
IEEE 802.15.4	

Şekil 2: 6TiSCH protokolü yığını öğeleri

Şekil 2 6TiSCH protokolünün katmanlarını göstermektedir. En üst seviyede uygulama katmanı bulunur. Bu katmanda 6TiSCH protokolü bünyesinde kullanılan üç adet protokol verilmiştir. CoAP protokolü [4] kaynak kısıtlı düğümler için özelleştirilmiş bir uygulama katmanı protokolüdür ve temel olarak HTTP protokolünün UDP üzerinden çalışan versiyonu olarak görülebilir. CoAP sayesinde 6TiSCH düğümleri birbirleriyle olan yapılandırma verilerini kolayca paylaşabilirler. Yol hesaplama ögesi (PCE) merkezi kaynak yönetimi gerektiren senaryolarda, merkeze kaynak ayırma isteği göndermek ve merkezden gelen kaynak ayırma onaylarını işlemekle görevlidir. Noktadan noktaya yol oluşturma ögesi (Tracking) ise bu katmanda 6TiSCH için gerçekleştirilen bir başka protokoldür. Bu protokol ağdaki iki nokta arasında bir yol oluşturur. Bu yola ait kaynaklar 6Top tarafından Tracking ögesinden gelen istekler doğrultusunda ayrılır.

Ağ katmanında 6TiSCH protokolüyle ilişkili olan tek protokol RPL protokolüdür [5]. Bu protokol ağdaki hiyerarşiyi oluşturarak, düğümlerin 6TiSCH protokolünün gerektirdiği senkronizasyon sırasında en uygun zamanlama kaynağını seçmelerini sağlar. 6LoWPAN katmanı ise büyük IPv6 veri paketlerinin 802.15.4 paket boyuna göre parçalanıp birleştirilmesi ve gerektiğinde paketlerdeki başlıkların sıkıştırılması görevini üstlenir [2]. 6Top katmanı zaman paylaşım sistemleri için gerekli olan dağıtık çizelgeleme ve çizelge koşma gibi fonksiyonları yerine getirir [3]. Bu katmadaki fonksiyonlar sayesinde IEEE 802.15.4e TSCH MAC katmanına ait zaman ve frekans kaynakları dinamik olarak üst katmanların hizmetine sunulur. Kaynak ayırma uygulamanın ihtiyacına göre dinamik bir şekilde yapılır. Örnek bir 6TiSCH çizelgesi Şekil 3'de verilmiştir. Şekilden de anlaşılacağı gibi çizelge çerçevesi (slot frame) 31 zaman dilimine sahiptir ve bu örnekte her zaman dilimi 10 ms olacak şekilde tasarlanmıştır. Çizelge her 310 milisaniyede kendini tekrar eder. Her ardışık çizelge koşulmasında aralarından haberleşen düğümler farklı frekans kanalları kullanır. Bu yaklaşım frekans çoğullaması sayesinde verilerin yüksek kararlılıkla ağ içinde taşınmasına sağlar [6]. IEEE 802.15.4e TSCH katmanı ise zaman paylaşım ve kanal atlamalı MAC protokolünü gerçekler [1]. Bu katman 6Top katmanından gelen kaynak ayırma/serbest bırakma isteklerini gerekli şekilde uygular.

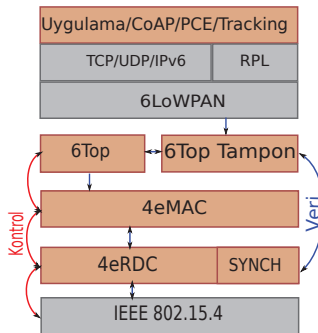


Şekil 3: Örnek 6TiSCH Çizelgesi

Bu bildiriye ilk olarak dağıtık bir çizelgeleme mekanizmasının 6TiSCH protokolünde gerçekleştirilmesi Bölüm II'de anlatılacaktır. Gerçeklenen yığının performansı Contiki işletim sisteminin bir parçası olan Cooja simülasyonu ile Bölüm III'te analiz edilecektir. Son olarak, Bölüm IV bu çalışmadan elde edilen çıkarımlar özetlenecektir.

II. 6TiSCH PROTOKOLÜNDE DAĞITIK ÇİZELGELEME

Contiki işletim sistemi için gerçekleştirilen 6TiSCH protokolü yığına ait genel yapı Şekil 4'te verilmiştir². Ağdaki düğümlerin senkronizasyonunu sağlayan öge SYNCH ögesidir. Bu öge işaretçi (Beacon) adı verilen ve periyodik olarak ağa gönderilen paketleri üretir. İşaretçi paketlerini alan düğümler zamanlama parametrelerini işaretçiden çıkartarak ağa senkronize olurlar. Ağa dahil olan düğümler, yetkisi olan düğümlerden gelen cevap (ACK) paketlerini kullanarak da ağdaki senkronizasyonlarını güncellerler. Düğümler işaretçi paketini aldıklarında senkronizasyon dışında çeşitli ağ yapılandırma parametrelerini de bu paketten elde ederler. Örneğin, çizelge uzunluğu, milisaniye cinsinden zaman dilimi (slot) uzunluğu, bütün ağdaki düğümler tarafından ortak olarak tutulan ve çizelgeyi doğru izlemeyi sağlayan ASN sayıcısı parametreleri işaretçi paketten elde edilir. Böylece, işaretçi paketin çözülmesi ile elde edilen parametrelerle yığın yapılandırılmış olur. İşaretçi paketin yapısı IETF 6TiSCH dokümanında tanımlanmıştır [3].

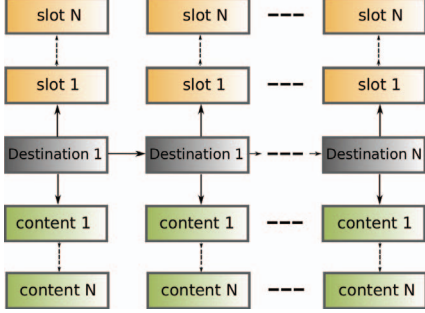


Şekil 4: Gerçeklenen 6TiSCH protokolü yığını öğeleri

Uygulamadan gelen veri paketleri bir sonraki hedefe gönderilmeden önce 6Top tamponuna yerleştirirler. 6Top tamponu

²Yığına ait alt parçaların bir çoğunun detayları yer kısıtlamasından dolayı bu bildiriye anlatılmamıştır

üç boyutlu bağlı bir liste olarak tasarlanmıştır ve her bir hedef adres için ayrı bir veri tamponu ve kaynak tamponu tutulur. 6Top tamponuna ait veri yapısı Şekil 5'te verilmiştir. Tampona yerleştirilen paketler öncelik sırasına göre bir sonraki düğüme gönderilecekleri ana kadar bekletilirler. An itibarıyla gerçekleşen yığın tamponda yüksek öncelikli paket olduğu sürece, düşük öncelikli paketleri işlememektedir.



Şekil 5: Gerçeklenen 6Top Tampon Yapısı

Tapmonda tutulan *Destination* veri yapıları, *Slot* veri yapıları ve *Content* veri yapıları çeşitli istatistik ve kontrol parametrelerini içerir. Yeni bir adrese ait bir paket tampona eklendiğinde, ilk olarak yeni bir *Destination* yapısı oluşturulur ve bu yapıya bir *Content* veri yapısı eklenir. *Destination* veri yapısı hedef düğüme ait tanımlanmış zaman dilimlerinin sayısını (*Slot*), geçerli zaman dilimini gösteren bir sayısal değeri ³, hedef düğüme yapılan gönderimleri tutan bir sayıcıyı ve hedef MAC adresini içerir. *Content* veri yapısı ise her mesajın boyutu, tamponda tutulan paketin türü ve önceliğini içerir. *Slot* veri yapısı ise bir kimliklendirme numarası, ve zaman diliminin tipini tutar. Zaman dilimleri Gönderim (TRANSMIT), Alım (Receive) ve Ortak (SHARED) tiplerinde biriyle işaretlenebilir.

A. Dağıtık Çizelgeleme

Dağıtık çizelgeleme algoritması Şekil 2'de verilen 6Top katmanında gerçekleştirilmiştir. Çizelgeleme algoritması periyodik olarak komşu düğümlerde ayrılmış olan kaynakları sorgular. Bu şekilde kendi yerel çizelgeleme tablosunu oluşturur. Düğüm yeni bir kaynağa ihtiyaç duyduğunda gerekli zaman dilimlerini ve frekans kanallarını bir liste olarak yerel çizelgeleme tablosuna bakarak oluşturur. Bu listeyi hedef düğüme erişmesi için tanımlanmış yönlendirici düğüme gönderir ve düğümden listede belirtilmiş kaynaklardan belli sayıda kaynağı veri alış verişi için ayırmasını ister. İstek gönderilen düğüm listeden kendisi için en uygun olan kaynakları yerel çizelgeleme tablosuna alım (RECEIVE) olarak işaretler ve başarılı kaynak ayırmasını içeren mesajı istek yapan düğüme gönderir. Kaynak ayırımı başarısız olursa isteği yapan düğüme bir hata mesajı döndürülür. Çizelgeleme için gerekli kontrol mesajları ortak olarak tanımlanmış zaman dilimlerinde gönderilir.

İki düğüm arasında önceden ayrılmış bir zaman dilimi kaynağı belli bir süre boyunca kullanılmamışsa yada ayrılmış kaynak miktarı düğümün tamponuna gelen veri yoğunluğundan

³Örneğin, 31 adet slot içeren bir çerçeve için şu anki zaman dilimini gösteren sayı 0 ile 30 arasında olabilir.

Algoritma 1 Kaynak ihtiyacı hesaplama algoritması

```

1: function CALCRESOURCEREQ(c_period, in_items,
   out_items, buf_occupancy, num_tx_slots)
2:    $r_{avg\_in} \leftarrow r_{avg\_in} * \alpha + in\_items * (1 - \alpha)$ 
3:    $r_{avg\_out} \leftarrow r_{avg\_out} * \alpha + out\_items * (1 - \alpha)$ 
4:    $service\_rate \leftarrow r_{avg\_in} / r_{avg\_out}$ 
5:    $slot\_req \leftarrow service\_rate * num\_tx\_slots$  ▷ Kaynak
   ihtiyacı tampon servis hızına göre hesaplandı.
6:    $req\_slot \leftarrow 0$ 
7:   if  $buf\_occupancy > avg\_buf\_occupancy$  then ▷ Şu
   anki tampon seviyesi geçmişe ait ortalamasının üzerinde
8:      $req\_slots \leftarrow round(slot\_req - num\_tx\_slots)$ 
9:   end if
10:  return  $req\_slots$ 
11: end function

```

daha fazlasını karşılayabilecek durumdaysa, gerekli hesaplamalar yapılarak ihtiyaç fazlası kaynakları silmek için çizelgeleme ögesi bir silme isteği oluşturur. Silme isteği, zaman dilimi kaynağında alıcı olarak işaretlenmiş düğüme gönderilir ve olumlu bir silme işlemi için bu düğümden onay beklenir. Eğer istek gönderilen düğüm belli sayıda silme isteği denemesine cevap vermezse, kaynak silinmiş ve kullanılmıyor kabul edilerek isteği yapan düğümün çizelgesinden çıkarılır. Silme isteğinin ne zaman tetikleneceği seçilen algoritmaya bağlıdır. Bu bildiride gerçekleştirilen kaynak ihtiyacı hesaplama fonksiyonu Algoritma 1'de verilmiştir. Bu algoritma 6Top tamponunda belli bir hedef adrese ait paketlerin servis edilme hızını dikkate alarak kaynak silme ve ekleme gerekliliğini hedef düğüme iletir. Algoritmanın çıktısı negatif olduğunda bu bir silme işlemine, pozitif olduğunda ise bu bir kaynak ekleme işlemine karşılık gelir.

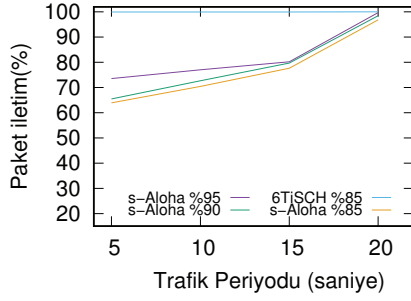
III. 6TISCH PERFORMANS ANALIZI

Bu bölümde, en son protokol ayrıntılarını ve dağıtık ölçeklenebilir bir çizelgeleme algoritması içeren IETF 6TISCH protokol yığınının performansı Cooja emulasyonu yardımıyla analiz edilmiştir. Protokol performansını 50 düğümlük bir ağda farklı paket gönderim sıklıkları ve farklı kanal başarımları için test edilmiş ve performans sonuçları slotted ALOHA [7] protokolüne ait performans sonuçlarıyla karşılaştırılmıştır. Toplayıcı düğüm test edilen ağın merkezine konumlandırılmıştır, 50 adet duyarga düğüm toplayıcı düğüm etrafına eşdağılımlı (uniform) olarak yerleştirilmiştir. Ağdaki duyarga düğümler aynı anda periyodik olarak kök düğüme mesaj gönderirler. Bu da her periyot esnasında bir trafik patlaması oluşturur (bursty traffic). Tablo I ağ yapılandırılmaları için kullanılan parametreler hakkında bilgi verir.

Düğüm Sayısı	s-Aloha Yapılandırması	Başarımlar	Mesaj Periyodu
50	5(7)	95, 90, 85	5,10,20

Tablo I: Ağ parametreleri

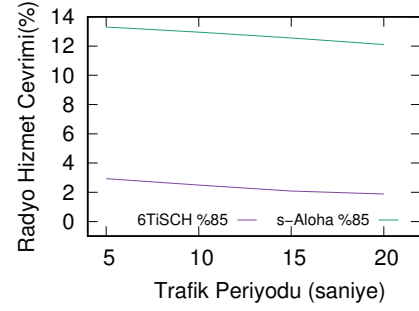
Tablo I'de verilen *s-Aloha yapılandırması* parametresi slotted Aloha protokolü için ağdaki ortak kullanılan zaman dilimi sayısını (5) ve çerçeve uzunluğunu verir(7). Buradan da anlaşılacağı gibi protokol yapılandırması ağ kaynaklarını %71 oranında kullanır. Ayrıca simülasyonlar sırasında kullanılan zaman dilimi boyutu 15 milisaniyedir.



Şekil 6: s-Aloha ve 6TiSCH paket iletim oranları

Şekil 6 ağıın paket iletim performansını vermektedir. Performans sonuçları farklı mesaj gönderim periyotları ve farklı kanal iletim oranları için verilmiştir. Şekilde 6TiSCH için sadece %85 kanal iletim oranı verilmiştir, zira bütün yapılan testlerde 6TiSCH %99'dan yüksek oranda bir paket iletim performansı ortaya koymuştur. Kanal başarımları serbest uzay yol (Free Space Path Loss) kaybına göre hesaplanmaktadır. Bu durumda iki düğüm haberleşebilecekleri en uzak mesafedeyken gönderdikleri mesajların başarımlarıyla verilen miktarı iletebilmektedir⁴ [8]. Mesajların gönderim sıklığı 5 saniye olduğunda slotted Aloha protokolünün paket iletim performansı oldukça düşük seviyelere inmektedir. Burada ortak zaman dilimlerinden aynı anda ağdaki 50 düğümün mesaj göndermeye çalışması çakışmalara neden olmakta ve performansı olumsuz yönde etkilemektedir. Kanalda herhangi bir çakışma algılandığında slotted Aloha protokolü belli bir erteleme (Back-off) mekanizmasıyla mesajı tekrar iletmeye çalışmaktadır. Bu erteleme mesaj periyodunun düşük olduğu durumlarda ağıın performansında ciddi bir iyileştirme sağlayamamaktadır. Bunun nedeni birçok düğümün çakışmaya maruz kalması ve erteleme sonucunda da çakışmaların belli ölçüde devam etmesidir. 6TiSCH protokolü ise birbirleriyle çakışmayan kaynakları düğümler arasında dağıttığından, ani bir trafik yoğunluğunda çakışmalar minimum düzeyde kalmaktadır. Bu da 6TiSCH performansının slotted Aloha'dan daha üstün olması sonucunu doğurmaktadır. Slotted Aloha performansı şekilden de görüldüğü gibi 20 saniye mesaj periyodu için 6TiSCH performansına yaklaşmaktadır. Ayrıca, slotted Aloha protokolünün performansı kanal başarımları arttıkça artmaktadır. Bu durum özellikle mesaj gönderim frekansının yüksek olduğu 5 saniye için geçerlidir. Bunun nedeni ise mesajların kötü kanal başarımlarından dolayı kaybedilmesi sonucu düğümlerin daha fazla gönderime ihtiyaç duymalarıdır.

Şekil 7 ağdaki düğümlerin radyolarının ortalama açık kalma sürelerinin oransal olarak gösterimini içermektedir. Daha önce de belirtildiği gibi, ağdaki düğümlerin radyoları bulunan zaman diliminin tipine göre kısa bir süre açık olarak kanala gönderilecek mesajlar kanala gönderilmekte ya da kanaldan gelen mesajlar alınmaktadır. Bu sayede ağıın düşük enerjiyle verimli bir şekilde çalışması sağlanabilir. Şekilden de görüldüğü gibi 6TiSCH teknolojisi her mesaj gönderim sıklığı için ortalama olarak slotted Aloha protokolünden daha üstün bir radyo hizmet çevrimi (radyo duty cycling) performansı göstermektedir. Bu fark her mesaj gönderim periyodu için yak-



Şekil 7: Radyo hizmet çevrimi değişimi

laşık 7 kattır. Bilindiği gibi slotted Aloha protokolünde zaman dilimlerinin tamamı ortak (SHARED) olarak kullanılmaktadır, bu da radyoların trafik olmasa bile belli bir süre dinleme amacıyla açılmasına neden olmaktadır. Gereksiz olarak radyonun açılması hem radyo çevrimi oranını arttırmakta ve hemde gereksiz enerji sarfiyatına neden olmaktadır. Beklenildiği gibi hem 6TiSCH ve hem de slotted Aloha protokolleri paket gönderim sıklığının yüksek olduğu durumlarda daha yüksek bir radyo hizmet çevrimi oranı ortaya koymaktadırlar. Bu durum yoğun mesaj trafiğinden dolayı radyoların daha uzun süre açık kalmasıyla ilişkilendirilir. Şekil 7'de verilen sonuçlara farklı kanal başarımları dahil edilmemiştir. Bunun nedeni ise bütün sonuçların benzer bir trend göstermeleridir.

IV. SONUÇ

Bu bildiriye, her geçen gün hayatımıza daha fazla yer bulan küçük cihazların internete dahil olmasını hızlandıracak bir protokol olan 6TiSCH protokolüne ait dağıtık bir çizelgeleme algoritmasının gerçekleştirilmesi Contiki işletim sistemi için anlatılmıştır. Gerçeklenen dağıtık çizelgeleme algoritmasını kullanan 6TiSCH protokolü yığınının performansı Cooja emülatöründe test edilmiştir. 6TiSCH Protokolü performansı slotted Aloha protokolüyle karşılaştırılmıştır. Sonuçlar 6TiSCH protokolünün dağıtık bir çizelgeleme algoritmasıyla düşük güçlü duyurulara ağlarına ölçeklenebilir ve yüksek performanslı bir çözüm sunduğunu ortaya koymuştur.

KAYNAKÇA

- [1] *IEEE 15.4, "IEEE 802.15.4™: WIRELESS PERSONAL AREA NETWORKS (PANs)", IEEE Std.*
- [2] "Internet Engineering Task Force (IETF) IPv6 over Low Power Wireless Personal Area Networks (6lowpan) Working Group," <http://datatracker.ietf.org/wg/6lowpan/>, Dec. 2010.
- [3] IETF, "IPv6 over the tsch mode of ieee 802.15.4e (6tisch)," <https://datatracker.ietf.org/wg/6tisch/charter/>.
- [4] Z. Shelby, "The constrained application protocol (coap)," IETF, Tech. Rep., 2014. [Online]. Available: <https://tools.ietf.org/html/rfc7252>
- [5] T. Winter, P. Thubert, and et al., "RPL: IPv6 Routing Protocol for Low Power and Lossy Networks, RFC 6550," IETF ROLL WG, Tech. Rep., March 2012.
- [6] D. Stanislawski, X. Vilajosana, Q. Wang, T. Watteyne, and K. Pister, "Adaptive synchronization in ieee802.15.4e networks," *Industrial Informatics, IEEE Transactions on*, vol. 10, no. 1, pp. 795–802, Feb 2014.
- [7] H. Takagi and L. Kleinrock, "Optimal transmission ranges for randomly distributed packet radio terminals," *IEEE Transactions on Communications*, vol. 32, no. 3, pp. 246–257, Mar 1984.
- [8] C. A. Balanis, *Antenna Theory: Analysis and Design*, Third ed. John Wiley and Sons Inc., 2005.

⁴Örneğin, %85 başarımlarında, iletilen her yüz mesajdan 85'inin alıcı tarafından alınması beklenmektedir.