



TBL-MAC: Doğrusal Kablosuz Algılayıcı Ağlar için Yeni Bir Ortam Erişim Kontrol Protokolü

TBL-MAC: A New Media Access Control Protocol Design for Linear Wireless Sensor Networks

Musa Çıbuk^{*1} , Davut Arı¹ , Fikri Ağgün² , Ümit Budak³ 

¹Bitlis Eren Üniversitesi Mühendislik ve Mimarlık Fakültesi Bilgisayar Mühendisliği, 13100 Bitlis, TÜRKİYE

²Bitlis Eren Üniversitesi Adilcevas Meslek Yüksekokulu Bilgisayar Teknolojileri Bölümü, 13100 Bitlis, TÜRKİYE

³Bitlis Eren Üniversitesi Mühendislik ve Mimarlık Fakültesi Elektrik - Elektronik Mühendisliği, 13100 Bitlis, TÜRKİYE

Başvuru/Received: 24/01/2022

Kabul / Accepted: 23/03/2022

Çevrimiçi Basım / Published Online: 31/07/2022

Son Versiyon/Final Version: 31/07/2022

Öz

Kablosuz Algılayıcı Ağların özel bir türü olarak literatürde yerini almış olan Doğrusal Kablosuz Algılayıcı Ağlar (DKAA)'da ağ yapısında düğümler sıralı olarak dizilmekte ve iletişim ortamı doğrusal düzende oluşturulmaktadır. Bu tür uygulamalar için geliştirilmiş Ortam Erişim Kontrol (OEK) protokollerinin doğrusal dizilim karakteristiğine uygun olarak tasarlanması önem arz etmektedir. Çünkü ağdaki düğüm sayısı arttıkça paketlerin uçtan uca gecikmesi ve koordinatör düğüme yakın olan düğümlerin veri trafiği çok yüksek seviyelere ulaşmaktadır. Ayrıca artan düğüm sayısı, çarpışma ve sıkışıklık ihtimalini de artırmaktadır. Tüm bu sebeplerden dolayı doğrusal topolojiler için geliştirilen protokollerin gecikme duyarlı, sorunsuz bağlanabilen ve hatasız veri iletimi yapan karakteristiklere sahip olması beklenmektedir.

Bu çalışmada, yukarıda bahsedilen DKAA problemlerini minimuma indirmek üzere etkili ve zor fiziksel şartlarda ağın bağlantı sürekliliğinin üstesinden gelen çevik bir OEK protokolü olan TBL-MAC geliştirilmiştir. Bu çalışma ile Doğrusal Ağ haberleşmesine katkı sunacak birçok yeniliğin literatüre kazandırılması hedeflenmiş, protokolün benzetimi yapılarak önerilerin geçerliliği test edilmiştir. Yapılan benzetim sonuçlarına göre TBL-MAC protokolünün içerdiği yöntemlerin kabul göreceği ve çalışmanın araştırmacılara yeni bakış açıları kazandıracığı düşünülmektedir.

Anahtar Kelimeler

KAA, DKAA, OEK Protokolü, Doğrusal Topoloji, Negatif Adresleme

Abstract

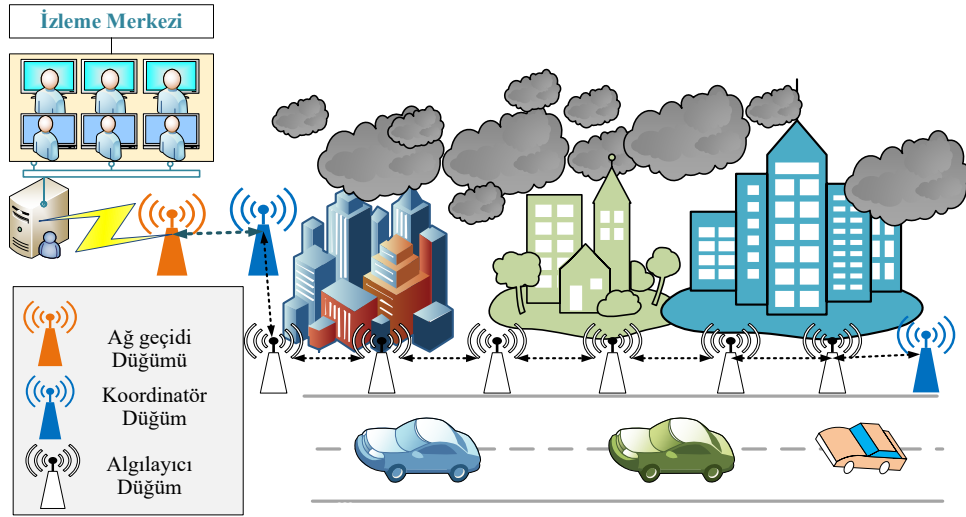
In Linear Wireless Sensor Networks (LWSNs), which has taken its place in the literature as a special type of Wireless Sensor Networks, the nodes are arranged in a sequential manner in the network structure and the communication environment is created in a linear order. It is important that the Media Access Control (MAC) protocols to be developed for such applications are designed in accordance with the linear sequence characteristics. Because as the number of nodes in the network increases, the end-to-end delay of the packets and the data traffic of the nodes close to the coordinator node reach very high levels. In addition, increasing the number of nodes increases the probability of collision and congestion. For all these reasons, protocols developed for linear topologies are expected to have characteristics that are delay sensitive, can be connected seamlessly, and transmit data to the sink without error. In this study, TBL-MAC, an agile MAC protocol that effectively overcomes the connection continuity of the network in difficult physical conditions, has been developed in order to minimize the above-mentioned LWSN problems. With this study, it was aimed to bring many innovations that will contribute to Linear Network communication to the literature, and the validity of the suggestions was tested by simulating the protocol. According to the simulation results, it is thought that the methods included in the TBL-MAC protocol will be accepted and the study will bring new perspectives to the researchers.

Key Words

WSN, LWSN, MAC Protocol, Linear Topology, Negative Addressing

1. Giriş

Günümüzde elektronik teknolojilerindeki gelişmeler ucuz maliyetli, işlem kapasitesi yüksek, düşük boyutlu algılayıcı düğümler (AD) üretilmesini mümkün kılmaktadır. Bu önemli gelişmeler neticesinde çok fazla algılayıcı düğümden oluşabilen Kablosuz Algılayıcı Ağ (KAA) uygulamalarının kolaylıkla gerçekleştirilmesine olanak tanıyan imkânlar ortaya çıkmıştır. Bu süreçte KAA'ların uygulama alanları artmış, uygulama biçimine özgü geliştirilmiş birçok KAA türü ortaya çıkmıştır. Doğrusal KAA (DKAA) Şekil 1'de görüldüğü üzere doğrusal topoloji karakteristiği gerektiren KAA uygulamaları için özelleşmiş bir türdür. Doğrusal dizilim özelliği sayesinde kurulum, bakım, yönlendirme gibi işlemler çok kolaydır. Fakat bu tip ağlarda uçtan uca gecikme, uç düğümlerdeki aşırı veri trafiği, ağ güvenilirliği, düğüm başarısızlığı ve bağlantı hatası(Chen, Tse, & Feng, 2009) gibi çözülmesi gereken zorluklar mevcuttur. Bu tür ağların doğrusal yapısı; güvenilir, verimli, enerji tasarrufu daha yüksek ve ağ ömrü daha uzun özel protokoller tasarlamak için önemli bir motivasyon kaynağı olabilmektedir.(Jawhar & Mohamed, 2009; Varshney, Kumar, & Swaroop, 2015).



Şekil 1. Örnek bir DKAA yapısı

Günümüzdeki teknolojik gelişmeler sayesinde AD'ler, kendi kendini organize edebilir yeteneğe sahip olmakla birlikte sınırlı veri işleme yeteneğine sahiptir. AD'ler sahadaki diğer düğümler ile koordine edildiklerinde, diğer düğümlerden gelen bilgiler ile belirli bir fiziksel ortamı çok ayrıntılı olarak ölçme ve gözleme yeteneğine sahip olabilirler. Bu nedenle KAA, belirli bir eylemi gerçekleştirmek için koordine olan algılayıcı düğümler topluluğu olarak tanımlanabilir. Geleneksel ağlardan farklı olarak, algılayıcı ağlar, yoğun dağıtım ve koordinasyon üzerine çalışırlar. Bu benzersiz özellikleri, KAA'ları geleneksel ağlara göre avantajlı kılar. Algılayıcı ağ uygulamaları öncelikle askeri uygulamalar için kullanılmış olup, savaş, terörle mücadele, düşman faaliyetlerinin tespiti ve gözetimi durumlarında geleneksel ağlara göre birçok avantaj barındırdığı görülmüştür(Jawhar & Mohamed, 2009).

DKAA'lar, boru hattı (Lai, Chen, Li, Huang, & Chu, 2012; Saeed, Ali, Rashid, Qaisar, & Felemban, 2014) maden tünelleri izleme (Jiang, Wu, Chen, Chen, & Leung, 2009), trafik izleme ve yol güvenliği (Low & Talampas, 2017) vb. birçok uygulamada kullanılmıştır (Jawhar & Mohamed, 2009). DKAA'larda, AD'ler doğrusal olarak bir hat boyunca dizilim göstermektedir. Bu tür ağlarda kalite ve güvenilirlik ön plandadır (Jawhar & Mohamed, 2009; Martin & Paterson, 2009).

Ağ kurulumu, yönlendirme işlemleri gibi çoğu KAA türü için zor süreçler, DKAA'larda çok kolaydır. Araştırmacılar köprüler, otoyollar, boru hatları vb. yapıların durumlarını izlerken diğer ağ problemlerine daha çok zaman ayırarak sayıları hızla artan çözümler sunmaktadırlar(Fang, Liu, & Qian, 2011; Karveli, Voulgaris, Ghavami, & Aghvami, 2009). Geleceğin akıllı yollarının ve şehirlerinin inşasında bu habitat içerisindeki unsurların birbirleri ile etkili ve verimli haberleşmesi büyük önem arz etmektedir. Bu da bizi bu unsurlar arasındaki haberleşme protokollerinin ne kadar verimli ve etkili olacağını sorgulamaya yönlendirmektedir. Akıllı sistemlerdeki bu her bir unsuru bir AD olarak düşünürsek bunlara arasındaki etkili ve verimli bilgi akışının sağlanması aslında resmin bütününe sistemin başarımı anlamına gelecektir. Tüm bu gelişmeler değerlendirildiğinde önerilen çalışmanın literatür açısından önemli bir yere sahip olacağı düşünülmektedir.

2. İlgili Çalışmalar

Karveli ve ark. Directional Scheduled MAC (DiS-MAC) (Karveli et al., 2009) olarak adlandırdıkları çalışmalarında; doğrusal bir hat üzerinde dağıtılmış ve birbirleriyle senkronize çalışan AD'ler arasında çarpışmasız iletişimi garanti eden bir protokol önermişlerdir. Yönlü antenlerin kullanımı sayesinde; daha yüksek kazançlar elde edilmiş, düğümler arası daha uzak mesafelere sinyal iletimi sağlanmış ve radyo dalgalarının belirli bir doğrultuda iletimi sonucu gürültü ve çarpışmalar ortadan kaldırılmıştır. DiS-MAC'taki, her bir düğüm bir alıcı-vericiye, yüksek kazanç değerine sahip bir yön antenine ve ters yönde daha düşük bir geri dönüş lobuna sahiptir. DiS-MAC'te veri iletimi ve diğer işlemler için tek bir kanal kullanılmaktadır.

Çekişmeye dayalı protokollerle karşılaştırıldığında, DiSMAC, RTS/CTS paketlerini kullanmaz ve kontrol paketinin yükünü de ortadan kaldırır. Ayrıca yönlü antenlerin kullanımı sayesinde çarpışmalar ve gizli terminal sorunları giderilmiş olur. Protokolün ana odağı, gecikmeden ziyade verimdir. Üç farklı senaryo altında benzetim çalışmaları gerçekleştirilmiştir: Sonuçlar, protokolün, düğümler arasında sabit ve güvenilir bağlantılar sağladığını göstermektedir. Ancak, daha büyük yükler daha fazla hataya ve sistem performansının düşmesine neden olmaktadır. Yazarlar böyle durumlarda kanal kodlama ve veri parçalama tekniklerinin DiS-MAC ile birlikte kullanılabilirliğini önermektedir. Öte yandan önerilen protokolle düğümler arasındaki eş zamanlamanın bozulması durumunda tüm ağın çökerek veri iletişiminin durması kaçınılmaz bir durumdur.

LC-MAC (Long-chain MAC) (Fang et al., 2011) protokolünde Fang ve ark., enerji verimliliğinden ödün vermeden uzun zincirli bir algılayıcı ağda uçtan uca teslimat gecikmesini azaltmak için röle düğümleri için önceden rezervasyon yapan ve bunu bir patlama şeklinde ileten bir mekanizmayı önermişlerdir. Bu mekanizma üç aşamadan oluşmaktadır. İlk aşamada başlatma(initialization) olarak nitelendirilebilecek olan konum algılama aşamasıdır. Bu aşamada röle düğümleri çevrelerindeki komşu röle düğümlerini algılar. Bu kuraldan hareketle; yalnızca bir komşu röle düğümüne sahip olan röle düğümü kendisini zincirin uç noktası olarak ayarlayacaktır.

MFT-MAC (Lee, Jwa, & Kim, 2013) adlı çalışmalarında Lee ve ark. senkronize bir yaklaşım kullanarak çekişmeye dayalı görev döngüsü olan bir MAC protokolü önermişlerdir. Tek bir döngüde SYNC, DATA ve SLEEP olmak üzere üç periyot kullanır. Enerji verimliliğini arttırmak ve uçtan uca gecikmeyi azaltmak için bir sonraki düğüme iletilecek DATA çerçevelerinin sayısını dikkate alan PION adında bir kontrol çerçevesi kullanır. MFT-MAC, uçtan uca gecikmeden ödün vermeden enerji tüketimini azaltmak için birden fazla veri çerçevesini tekli görev döngüsünde iletir. PION çerçevesi kontrol amaçlıdır ve RTS/CTS mekanizmasına benzer. Bu yolla gizli terminal problemi çözülmeye çalışılır. PION, tüm RTS/CTS alanlarını ve nihai hedef adresini, atlama sayısını ve çoklu karelerin sayısını içerir. SLEEP periyodunda, düğümler çoklu veri çerçevelerini almak için belirli bir zaman diliminde uyanırlar ve daha sonra kendi veri çerçevelerini bir önceki düğümden aldıkları çerçevelerine ekleyerek tekrar uykuya dönerler. Yazarlar, MFT-MAC'ı adaptif bir görev döngüsü kullanan DW-MAC (Y. Sun, Du, Gurewitz, & Johnson, 2008) ve RMAC (Du, Saha, & Johnson, 2007) protokolleri ile karşılaştırmıştır. Sonuçlar, MFT-MAC'ın ortalama güç tüketimi, ortalama uçtan-uca gecikme ve verim bakımından DW-MAC ve RMAC'a üstün olduğunu göstermektedir.

Sun ve ark. CMAC-T (T. Sun, Yan, & Yan, 2013) adını verdikleri çalışmalarında; belirteçlere(tokens) dayanan bir zincir (chain) tipi ortam erişim kontrol protokolü geliştirmişlerdir. Mahsulün büyüyen ortamını izlemek için tasarlanmış ve uygulanmıştır. İşaret (beacon) ve veri çerçevesi olmak üzere iki tip çerçeve kullanılmaktadır. İşaret çerçevesi, komşu düğümler arasındaki senkronizasyon ve kanal izinin atanmasında kullanılır. CMAC-T'de senkronizasyon işlemi için sink düğümü periyodik olarak belirli sayıda işaret çerçevesi gönderir. Ağdaki bir düğüm uyku modundan çıktıktan sonra sink düğümün gönderdiği işaret çerçevelerinden birini almaya çalışır. Eğer alırsa, senkronizasyonu bitirir ve daha sonra komşu düğüme verisini iletir. Aksi halde, düğüm uyku modunda kalmaya devam eder. Bu sayede ağ, veri aktarımında veri çarpışmasından kaçınmakla kalmaz, aynı zamanda güç tüketimini de azaltır. Yazarlar CMAC-T'nin bir serada uygulandığını ve yüksek güvenilirlik ve düşük güç tüketimi gereksinimlerini karşıladığını belirtmektedirler.

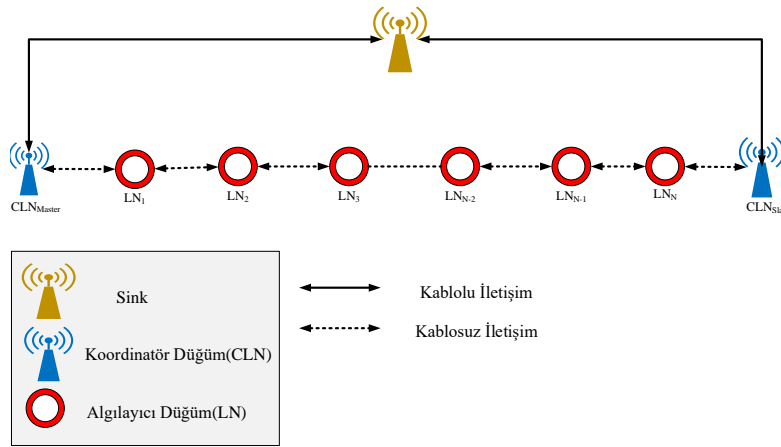
WiWi (Caneva, 2010) adlı çalışmada Caneva ve Montessoro; bağımsız düğümler zinciri boyunca senkronize çoklu sekmeli iletme dayanan çekişmesiz bir MAC protokolü önermişlerdir. Düğümler doğrusal veya eğrisel bir şerit oluşturacak şekilde dizilirler. WiWi temelde bazı özellikler bakımından DiS-MAC (Karveli et al., 2009)'e benzemekle beraber DiS-MAC'ın gelişmiş bir halidir denilebilir. Özellikle her iki protokol, komşu düğümler arasında sıralamaya dayalı alternatif iletim kullanarak eşzamanlı iletimler arasındaki girişimleri önlemektedir. Bir paketin alınması ve iletilmesi arasında, çarpışma olmaması için düğümün 4 zaman dilimi beklemesi gerekmektedir.

DKAA'lar her geçen gün popülerliğini artırarak devam etmektedir. Düşük özellikteki sınırlı kaynaklara sahip olan algılayıcı düğümlerin kaynakları verimli kullanmasına etki eden en önemli faktörler arasında iyi tasarlanmış OEK protokolleri önemli bir yere sahiptir. Katmanlı bir KAA mimarisinde OEK katmanı; gecikmelerin azaltılmaya çalışıldığı, paket kayıplarını azaltmak için olası çarpışmaların engellendiği ve minimum paket trafiği ile enerji verimliliğinin artırılmaya çalışıldığı katmandır. Çünkü düğümlerin haberleşmeyi sağlayan radyo ünitesi bu katman tarafından kontrol edilmektedir. Algılayıcı düğümlerde radyo arabirimi enerji tüketiminin en fazla olduğu birimdir. Sınırlı enerji kaynağına sahip olan AD'ler için enerji tasarrufu ağ ömrü açısından çok önemlidir. Tüm bu sebeplerle OEK katmanı için iyi tasarlanan protokoller KAA uygulamalarının enerji performansı ve diğer ağ performansları açısından önemli bir yere sahiptir.

Bu çalışma kapsamında geliştirilen TBL-MAC protokolü; özellikle şehir içi ve şehirlerarası karayollarındaki yol kenarı aydınlatmalarında kontrol ve enerji tasarrufu konuları dikkate alınarak tasarlanmıştır. Etkin ve verimli bir OEK tasarımı için literatürdeki benzerlerinden farklı olarak paket tekrarları ve özellikle de yankılanma sorunlarına karşı yeni ve etkin bir yöntem olan Negatif Adresleme yaklaşımı geliştirmiş olup bu yaklaşımın etkinliği benzetim sonuçlarında da açıkça gözlemlenmiştir. Ayrıca enerji verimliliği ve ağdaki kötü durum senaryolarının üstesinden gelebilmek için farklı çalışma modları (ToSlave, ToMaster ve 2Way) ortaya konulmuştur. Geliştirilen protokolde LC-MAC (Fang et al., 2011)'teki LDP'ye benzer bir yaklaşımla topoloji ve düğüm konumlandırma yapılmaktadır. Ayrıca paket iletiminde DiS-MAC (Karveli et al., 2009) ve WiWi (Caneva, 2010)'deki çalışmalara benzer olarak sıralamaya dayalı bir iletim mantığını benimsemiştir. Dolayısı ile önerilen OEK protokolünde çekişmesiz bir yaklaşım oluşturulmuştur. Ayrıca Negatif Adresleme yaklaşımı sayesinde gönderilen paketlerin hedefine iletilmesi garanti edilmekte olup bunun için ekstra bir onay veya paket kullanılmamaktadır. Bu da enerji verimliliği ve ağdaki trafiğin azaltılması konularında önemli bir rol üstlenmektedir. Bu yönleriyle, geliştirilen OEK protokolümüzün literatüre etkin bir katkı sağlayacağı düşünülmektedir.

3. TBL-MAC Protokolü İşlem Döngüsü

Geliştirilen TBL-MAC OEK protokolü için önerilen topolojik yapı Şekil 2'de detaylı olarak gösterilmektedir. Temel olarak yol kenarı aydınlatma birimlerinin üzerinde çalışan düğümler ve bunların kendi aralarında haberleşerek bilgi aktarımı yaptıkları bir ortam düşünülmüştür.



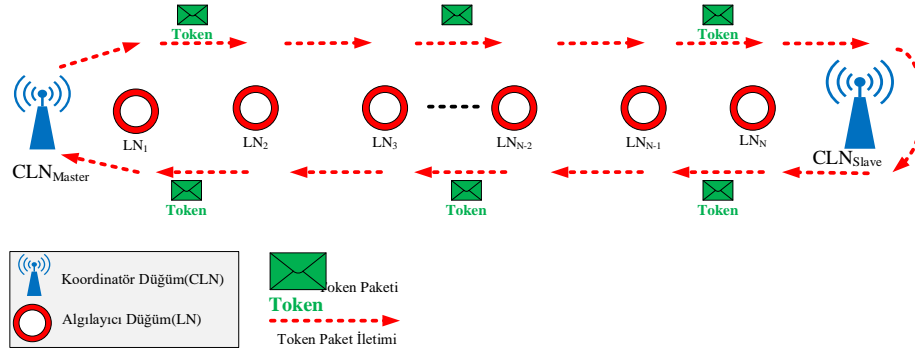
Şekil 2. Protokol örnek topolojik yapısı

Bu yapıda Algılayıcı Düğüm (Lightning Node - LN), Koordinator Düğüm (Coordinator Lightning Node - CLN) ve SINK Düğüm olmak üzere 3 ağ elemanı mevcuttur. Çalışmada ağda bulunan LN'lerin bir sıra halinde uygulama alanına dizilmesi ve bunların CLN'ler ile haberleşerek SINK düğüme verilerini aktarabilecekleri bir protokol tasarlanmıştır.

CLN'nin donanım ve yazılım kabiliyeti diğer tüm düğümlere göre daha gelişmiştir. Algılama görevinin yanı sıra, **token** adı verilen konum algılama paketleri üretmek ağda iletişim hattı kurulumu ve veri paketlerinin toplama işleminin başlatılmasından sorumludur. Benzer çalışmalardan farklı olarak topolojik yapı gereği ağda 2 adet CLN düğüm olmak zorundadır. Bu düğümler token ve veri paketlerini karşılıklı geri yansıtma veya SINK düğüme göndermekle görevlidirler. CLN'lerden biri Master diğeri ise Slave durumunda çalışır. Her zaman ağı harekete geçiren ve lineer yapının başı konumunda olan düğüm Master CLN'dir. Slave CLN daha çok gelen paketlerin yön değişikliğini veya SINK'e iletilmesini gerçekleştirir ve lineer yapının son düğümü konumundadır.

LN'ler algılama görevleri yaparak sahadan veri toplar ve ayrıca veri iletişimi için röle görevi de icra ederler. Bir LN temel olarak herhangi bir paketin başlangıç noktası değildir. LN sadece kendisine gelen token veya veri paketlerine kendi bilgilerini ekleyerek bir sonraki düğüme iletmekle sorumludur.

SINK düğümü tüm verilerin toplandığı uç düğüm ağ elemanı olarak çalışmaktadır. Genel olarak ağda CLN veya LN'ler gibi aktif bir rol oynamazlar.



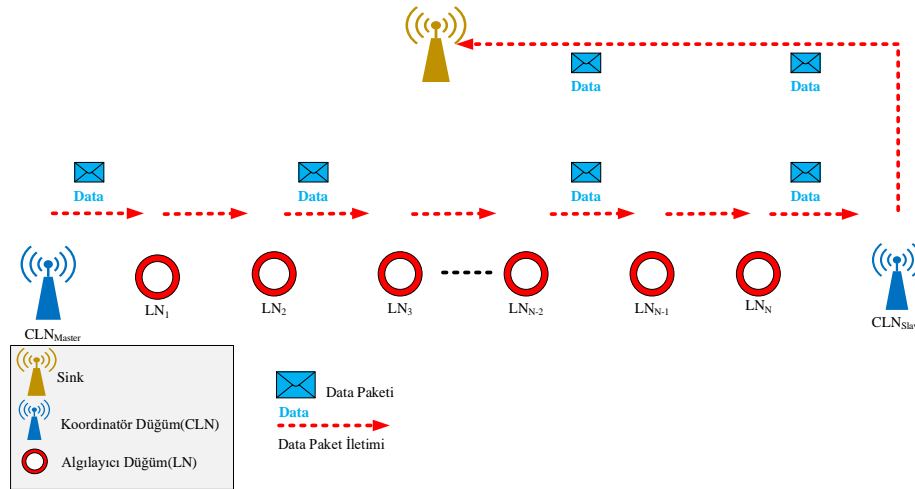
Şekil 3. Token paketi döngüsü

TBL-MAC Ağ Başlatma Mekanizması şöyledir: Ağ ilk çalıştığında master CLN boş bir token paketi oluşturarak ileri yönlü olarak gönderir. İlk gönderen (master) CLN ağı başlatan CLN olur ve gönderdiği token paketinin kendisine geri dönmesini beklemeye başlar. Token paketi hatasız bir şekilde master CLN'ye ulaştığında iletişim hattı kurulmuş ve bu hattın veri gönderimi için uygun olduğu kesinleşmiş olur. Ayrıca gelen token içerisinde ağdaki diğer düğümlerin bilgileri olduğu için Master CLN ağın topolojisini de elde etmiş olur. Bir CLN'nin Master ya da Slave olma durumu TBL-MAC protokolünün çalışma moduna göre dinamik olarak değişebilir. TBL-MAC protokolünde token paketi iletimi sırasında paketi alan her LN kendi kimlik (id), tip ve atlama bilgisini ekleyip sonraki düğüme iletir. Ağın kurulumu token paketinin ağda dolaşımını bitirmesi ile tamamlanmaktadır.

Protokolde veri haberleşmesi ve veri paket iletim yönüne göre üç farklı iletişim metodu gerçekleştirilmiştir. Tüm senaryolarda ağın kurulduktan sonra veri haberleşmesi aşamasına geçilmektedir. Veri ve token paketlerinin iletilme biçimleri kullanılan iletim metoduna göre yer yer değişiklik göstermektedir. Farklı metotların geliştirilmesindeki amaç olası farklı kötü durum senaryolarına ve verimlilik açısından ortama en uygun yaklaşımın kullanılmasıdır. Bu çalışmada geliştirilen paket iletim metotları aşağıdaki gibidir;

3.1. "ToSlave" Paket iletim metodu

Bu iletim metodu sadece Slave CLN'nin SINK düğüme bağlantısının olduğu durumlar için geliştirilmiştir. Şekil 4'ten de görüleceği üzere veri paketi master CLN'den slave CLN'ye doğru LN'ler üzerinden aktarılarak iletilir.

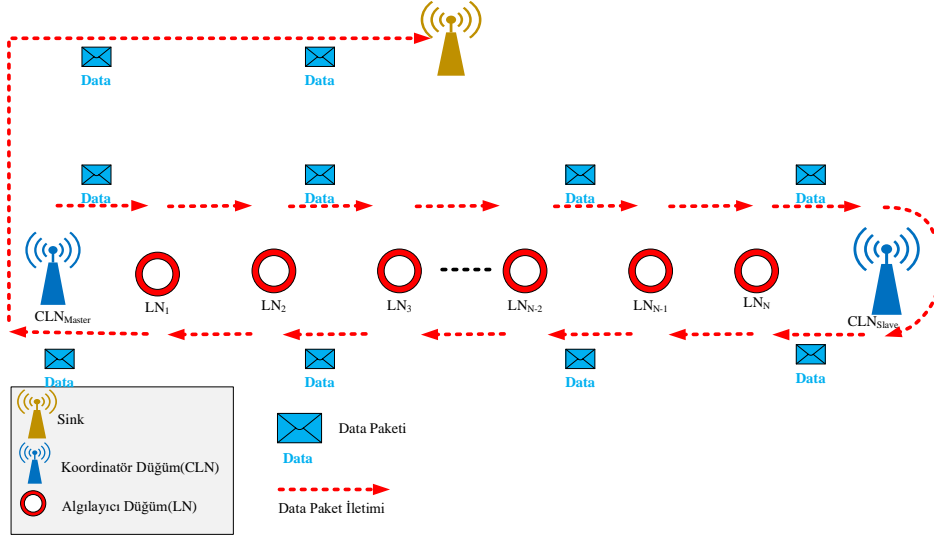


Şekil 4. ToSlave iletişim metodu

Slave CLN kendisine gelen veri paketini varsa kendi verisini de ekleyerek direkt olarak SINK düğüme iletir ve bu şekilde veri iletim süreci tamamlanır. ToSlave metodunda veri aktarımı daima tek yönlüdür. ToSlave metodu deterministik bir yaklaşımı benimsemektedir. Verinin tek yönlü akması geri dönüş gerektirmemesi sayesinde daha az paket trafiği oluşmaktadır. Öte yandan SINK düğüme doğru ilerledikçe veri paketlerinin büyümesi dolayısıyla SINK düğüme yakın düğümlerde daha çok enerji harcanması kaçınılmazdır. Enerji harcaması deneysel sonuçlarına ilerleyen bölümlerde değinilmiştir.

3.2. “ToMaster” Paket iletim metodu

Bu iletim metodu sadece master CLN'nin SINK düğümüne bağlantısının olduğu durumlar için geliştirilmiştir. Şekil 5'te ToMaster metodu için veri akışının nasıl olduğu gösterilmiştir.

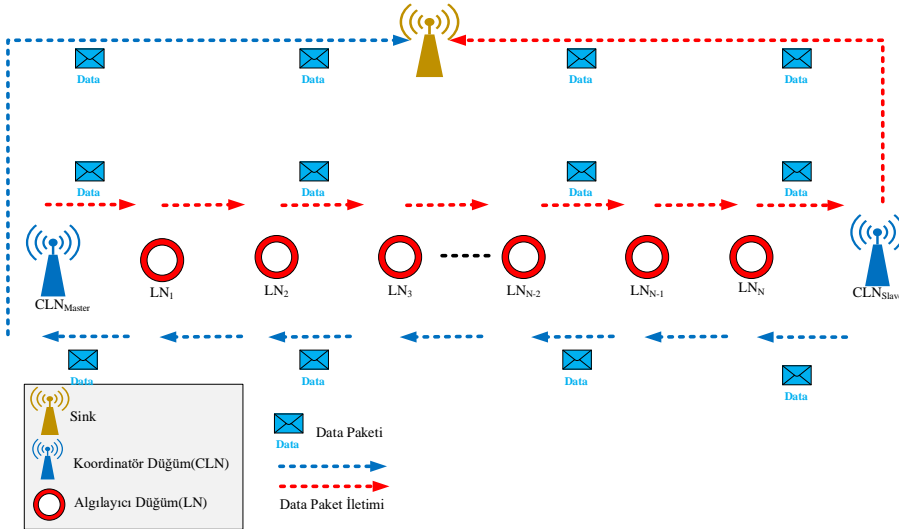


Şekil 5. ToMaster iletişim metodu

Görülebileceği üzere veri paketi master CLN'den slave CLN'ye doğru LN'ler üzerinden aktarılarak iletilir ve nihai durumda paketi alan slave CLN paketin yön bilgisini değiştirerek ve varsa kendi verisini pakete ekleyerek, paketi kendisine gönderen LN düğümüne geri gönderir. Paketi alan LN de ters yönlü olarak varsa kendi güncel verisini de ekleyerek bir sonraki LN'ye aktarır. Süreç, paket tekrar master CLN'ye ulaşmaya kadar devam eder ve master CLN kendisine ulaşan veri paketine varsa kendi verisini de tekrar ekleyerek direkt olarak SINK düğümüne iletir ve bu şekilde veri iletim süreci tamamlanır. ToMaster metodunda veri aktarımı çift yönlüdür. Burada slave CLN yön değiştirme (yansıtma) işleminde rol alır. Burada paket içerisinde yön bilgisi önemli rol oynar. Verinin çift yönlü akması sayesinde ToSlave metodundan farklı olarak düğümler arası enerji harcama durumu daha dengeli olmaktadır. Ancak veri paketleri için hat uzunluğu teknik olarak 2 kat uzayacağından veri paketi boyutları ToSlave metoduna göre daha büyük olacaktır. Tüm senaryoların enerji harcaması deneysel sonuçları ileride incelenmektedir.

3.3. “2Way” Paket iletim metodu

Bu iletişim metodunda veri aktarımı çift yönlü olarak gerçekleştirilir. Şekil 6'da 2Way metodu için veri akışının nasıl olduğu gösterilmiştir.



Şekil 6. 2Way iletişim metodu

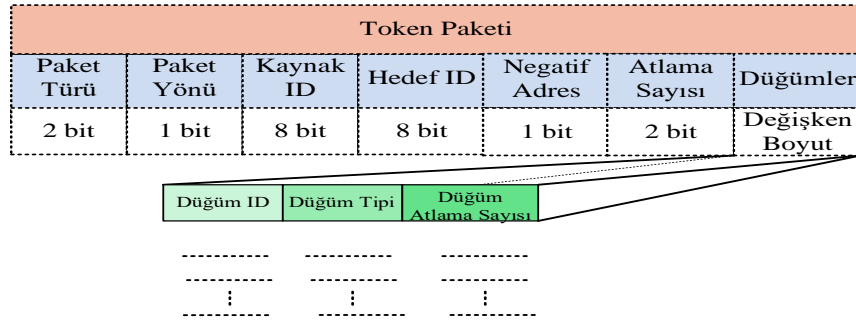
Diğer metotlardan farklı olarak Master ve Slave CLN düğümlerinin ikisi de veri paketi oluşturarak zıt yönlü iletilir. İletişim yine diğer yöntemlerde olduğu gibi master CLN tarafından başlatılır ve veri paketini alan slaveCLN bunu direkt olarak SINK düğümüne gönderir.

Akabinde slave CLN yeni bir veri paketi oluşturarak masterCLN'ye doğru iletir ve paket Master CLN'ye geldiğinde gerekli işlemler yapıldıktan sonra SINK düğümüne gönderir. Diğer iki yöntemden farklı olarak bu yöntemde her iki CLN düğümünün SINK ile bağlantısı vardır.

2Way metodunda diğer yöntemlerin avantajlı tarafları birleştirilmiştir. Hat uzunluğu arttıkça paket boyutları artmakta ve hat sonundaki uç düğümlerin paket anahtarlama maliyetleri artmaktadır. 2Way metodunda ise paketler çift yönlü aktarılmaktadır. Master CLN çıkışı paket akışları ToSlave metodundaki gibi çalışır ancak sonrasında Slave CLN'nin veri akışı başlatması ile işlem ters yönde tekrarlanır. Böylece düğümler arası paket anahtarlama maliyetleri tüm düğümler üzerinde dengelenmiş olur. Ayrıca 2Way metodunda her iki CLN de SINK düğümüne bağlı olduğu için yedekli bir çalışma senaryosuna müsait bir altyapı da sağlanmış olur.

4. Paket Yapıları

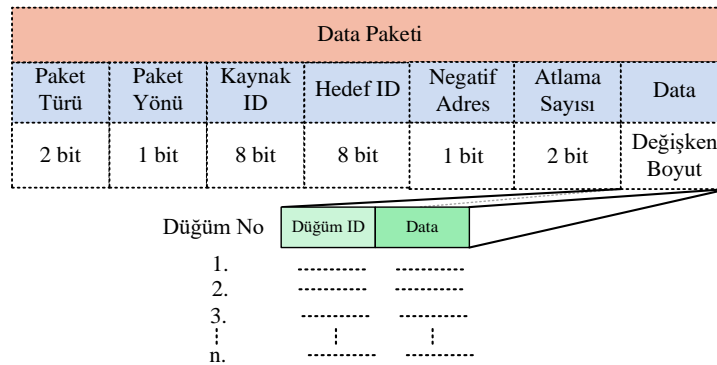
TBL-MAC protokolünde iki tip paket yapısı kullanılmıştır. Paket tipi (türü) bilgisi her pakette ilk alan olup *token için 0* ve *veri için 1* değeri atanmıştır. Teknik olarak her iki paket türünün başlık (header) kısımları aynı olup sadece son alanda birbirlerinden ayrılmaktadırlar. Token paketlerinin yapısı Şekil 7'de gösterilmiştir.



Şekil 7. Token paket yapısı

Token paket yapısında üzerinde durulması gereken iki önemli alan vardır. Bu alanlardan birincisi Negatif Adres alanıdır. Geliştirilen TBL-MAC protokolünün yeni ve en etkin yanlarından biridir. Paket yansımaları sonucu oluşan kısır döngüleri ve paketlerin gittiği yönde ilerlemesini garanti etmek için kullanılmaktadır. Bu alan 1 bitlik bir alan olup set edilmesi durumunda negatif adres mantığına göre hareket edileceği anlamına gelir. Diğer alan olan Atlama Sayısı alanında ilgili paketin kaç atlama mesafesinden iletildiği bilgisi tutulmaktadır. Bu alan özellikle herhangi bir düğümün (kapanma, arızalanma vb. gibi) ağdan kopması durumunda ağın devamını sağlamak üzere kullanılır. Düğümler alanı sıralı bir liste kaydı olup her kayıta (node_item) bir düğümün id, düğüm türü, düğüm atlama sayısı bilgisi bir yapı (struct) şeklinde tutulmaktadır.

Veri paketlerinin yapısı token paketi ile birebir aynı olup sadece Data alanında farklılık göstermektedirler. Bu alanda token paketlerinden farklı olarak "0" yerine "1" değeri vardır ve bu protokol içerisinde PK_TYPE_DATA sabiti ile temsil edilmektedir. Şekil 8'de Veri paketlerinin yapısı gösterilmiştir.



Şekil 8. Veri paket yapısı

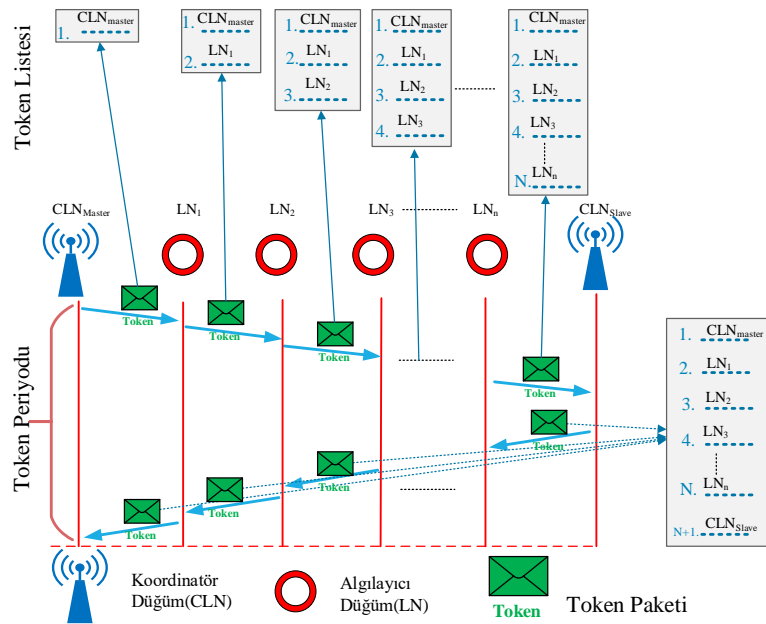
Şekil 8’den de görüleceği üzere Veri paketinin “data” alanı değişken uzunlukta olup düğümlerin adres (id) bilgisi ve ilgili düğümün kendisine ait veriyi barındırmaktadır. Bu alanın içeriği bu çalışma kapsamında olmayıp, bu çalışmada tüm düğümlerin periyodik olarak artan bir sayısal bilgiyi aktardıkları varsayılmış ve veriler buna göre oluşturulmuştur.

TBL-MAC protokolünde öncelikli olarak zaman bölmeli bir yaklaşımla (Ari, Cibuk, & Ağgün, 2018) çoklu kanal kullanımı amaçlanmıştır. Ancak protokol için benimsenen token tabanlı deterministik yaklaşım sayesinde tek bir kanal üzerinden tüm düğümlerin haberleşmesinin daha etkin olduğu görülmüş ve bu doğrultuda “sabit kanal tahsis yaklaşımı” benimsenmiştir. Tek bir kanal kullanıldığından, kanal yönetimi oldukça kolaydır. Çoklu kanal mekanizmalarında görülen kanal anahtarlama (değiştirme) ve girişim kaynaklı problemler yaşanmamaktadır. Tek kanal kullanımı sayesinde geliştirilen algoritmalar ve donanım kaynaklarının daha basit ve uygun maliyetli olması sağlanmıştır.

TBL-MAC protokolünde NRF905 (Nordic Semiconductor, n.d.) alıcı-verici donanımı baz alınarak 902 MHz frekansı 200KHz bant genişliğinde ve 999KHz data rate değerinde bir kanal yapısı baz alınarak çalışmalar yürütülmüştür.

5. Haberleşme Algoritma Yapısı

Düğümlerin ağa katılması token paketinin hat boyunca dolaşması ile olmaktadır. Her düğüm, token paketine bilgisini ekleyip sonraki düğüme iletmesi ile ağa katılma işlemi tamamlanmaktadır. Token paketi ilk token paketini gönderen Master CLN’ye ulaştığında ağdaki tüm düğümler ağa katılma işlemini tamamlamış olmaktadır. Şekil 9’da token paketi için zaman çizelgesi ve token paketinin ilerleyişi gösterilmektedir.

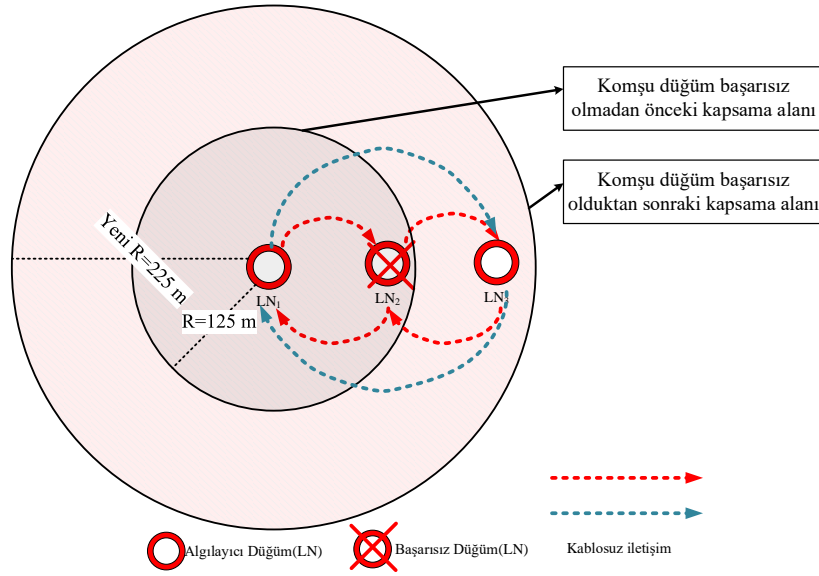


Şekil 9. Token ile ağ kurulumu zaman çizelgesi

Token paketleri belirlenen periyotlarda sürekli olarak gönderilir. Böylece belirli bir zaman diliminde, ağdaki düğümlerin konumu ve canlılığı test edilerek, her token döngüsünde ağ yenilenmiş olur. Bir token döngüsü başladığında veri akışı durdurulur ve token döngüsünün sağlıklı olarak tamamlanması beklenir. Bu şekilde benimsenen yaklaşım sayesinde veri paketleri ağın kurulumundan emin olunduktan sonra gönderileceği için olası veri kayıplarından kaçınılmış olur. Veri paketi haberleşmesi, Token döngüsü bittikten ve iletim hattı kurulduktan sonra başlar. Ağdaki düğümlerin veri toplama (üretme) işlemi senkron ve asenkron olarak iki farklı şekilde ele alınmıştır. Senkron veri üretiminde; tüm düğümler belirli bir zaman diliminde aynı aralıklarda eşit sayıda veri üretirler ve gönderirler. Asenkron veri üretiminde ise düğümler farklı periyotlarda değişik miktarda veri üretir ve iletirler. Geliştirilen TBL-MAC protokolü her iki veri üretme biçimine de uyumlu olarak tasarlanmış olup, yöntem ve çalışmaların sağlıklı kıyaslanabilmesi için bu çalışmada senkron veri üretimi tercih edilmiştir.

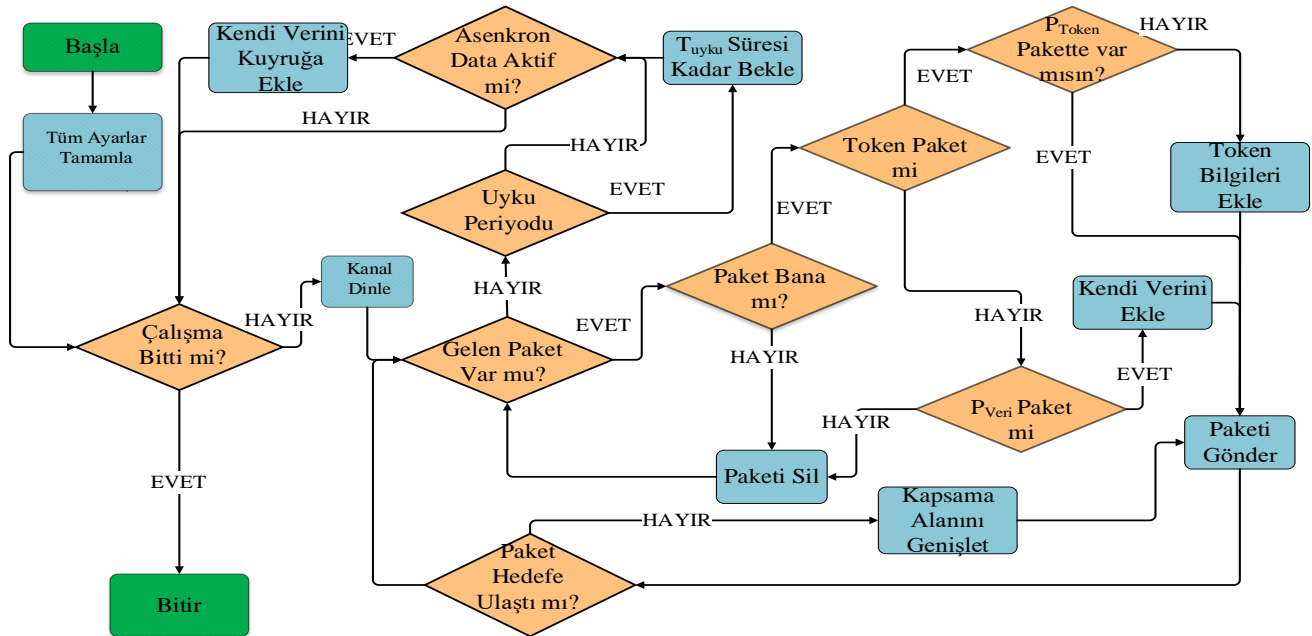
İleri yönde hareket eden paketin kablosuz haberleşmenin yayım (broadcast) özelliği gereği gönderici düğüme de ulaşmaktadır. Bu durum gereksiz işlem maliyetine ve kısır paket döngülerine sebep olduğundan dolayı istenmeyen bir durumdur. TBL-MAC protokolünde bu olumsuz duruma yeni bir yaklaşım olan “Negatif Adresleme” adı verilen yöntem ile çözüm aranmış ve başarılı sonuçlar elde edilmiştir.

TBL-MAC protokolü düğümün ağdan düşmesi, kopması gibi başarısızlıklarına karşı ağ iletişiminin sürmesi için bir mekanizma tanımlanmış olup, bu mekanizma ile düğüm ağdan düştüğünde komşu düğümler paket hareketliliğine bakarak ağdan düştüğünü anlamakta kendi kapsama alanlarını artırarak iletim hattının devamlılığını sağlamaktadırlar(Çıbuk, Arı, Ağgün, & Budak, 2021). Şekil 10'da söz konusu mekanizmanın çalışma prensibi görülmektedir.



Şekil 10. Düğüm başarısızlığı durumunda düğümlerin davranış şeması

Protokol içerisinde kullanılmak üzere tanımlanmış olan LN ve CLN düğümlerinin tasarımı, geliştirilen yöntemlere uygun olarak gerçekleştirilmiştir. TBL-MAC protokolünde LN'lerin alandan veri toplamak ve gelen paketleri bir sonraki düğüme iletmek (röle görevi) olmak üzere iki görevi vardır. LN alandan topladığı verileri kaydeder, kendisine bir veri paketi geldiğinde bu veriyi pakete ekleyerek bir sonraki düğüme iletir. Şekil 11'de LN düğümünün çalışma mekanizması (algoritması) gösterilmiştir.



Şekil 11. LN düğüm algoritmasının akış şeması

Şekil 11'den de görüleceği üzere LN'ler ilk çalıştığında öncelikli olarak (kanal, frekans, başlangıç değerlerinin atanması vb. gibi) temel ayarları yapılıp. Daha sonra LN, yaşam süresi (bu çalışmada benzetim süresi olarak belirlendi) bitene kadar şemada belirtilen görevleri yerine getirir. LN öncelikle kanalda herhangi bir paket olup olmadığına bakar. Eğer paket var ise bunların P_{token} ya da P_{veri}

gereği veriler birikerek uç düğümlere ulaşır. Dolayısıyla uçta bulunan CLN düğümlerin iş yükü çok fazladır. CLN'lerin algoritma akış şeması Şekil 12'de gösterilmektedir.

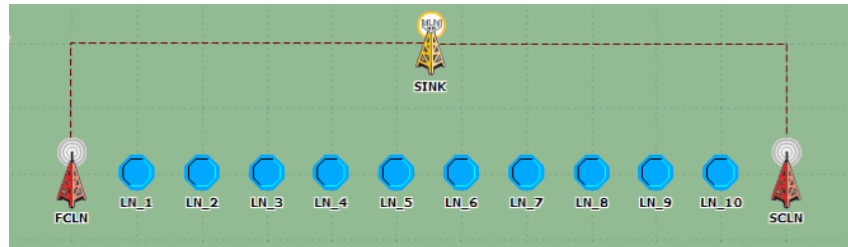
Şekil 12'den de görüleceği üzere CLN'ler ilk çalıştığında öncelikli olarak temel ayarları yapılarak yaşam süresi bitene kadar akış şemasında bertilen görevleri yerine getirir. Tüm CLN'ler sürekli olarak ortak kanalı "herhangi bir paket hareketliliği var mı?" diye kontrol etmektedir. Eğer kanalda paket varsa paketleri alır paket bitene kadar P_{token} ya da P_{veri} paketlerini işler. Ağda Master CLN token paketlerinin oluşturulmasından sorumludur. Master CLN'in oluşturduğu P_{token} tüm ağı dolaşır gelmiş ve ağ kurulumu(veri iletim hattı) tamamlanmış ise bu sefer veri iletim aşamasına geçer ve ilk Veri paketini oluşturarak hattan iletir. İlgili düğüm eğer Master CLN değilse gelen Token veya Veri paketlerine çalışma metotlarına göre davranır. Eğer gelen paket P_{veri} ise, paketi daha önceki bölümde anlatıldığı üzere mevcut üç iletim mekanizmasından (ToSlave, ToMaster, 2Way) uygun olanı seçerek iletişimi gerçekleştirir. Master CLN ağda belirli periyotlarla Token ve Veri paketi üretir. Bu sayede ağda paket karmaşıklığı ve paket çarpışmasının önüne geçilir. LN'lerde olduğu gibi CLN'ler de ağa paket gönderdiğinde paketin hedefe ulaştığını anlamak için paketin negatif adresli geri yansımaları takip eder. Bu şekilde ağda oluşabilecek düğüm ve bağlantı başarısızlığının önüne geçilmeye çalışılır.

6. TBL-MAC Benzetim ve Performans Değerlendirmesi

Geliştirilen TBL-MAC protokolünde performans testleri ile benzetim işlemleri, dış dünya fiziksel özelliklerine çok yakın ağ ortamı sunan eski adıyla Opnet ve şimdiki adıyla Riverbed Modeler (Riverbed Modeler, n.d.) olarak bilinen benzetim ortamında yapılmıştır. Benzetim işlemlerinde (2x10 çekirdekli, 3.4 GHz Intel Xeon işlemci, Nvidia Quadro P5500 GPU kartı, 64GB RAM özelliklerine sahip) HP Z840 iş istasyonu kullanılmıştır.

6.1. Önerilen protokolün ağ model

Çalışılan ağ modeli; DKAAs özelliği taşımaktadır. Önerilen yaklaşımın ağ yapısı Şekil 13'de gösterilmektedir. Bu örnekte 10 adet LN düğüm, Master ve Slave olmak üzere 2 adet CLN ve 1 adet SINK düğüm yer almaktadır. Ayrıca düğümlerin dinamik kapsama alanı yönetimi için *wcoverage* olarak adlandırılan yardımcı bir düğüm modeli kullanılmıştır.



Şekil 13. TBL-MAC protokolüne ait proje modeli

Bu ağ ortamında düğümler arası mesafe 100m ve her bir düğüm 125m yarıçaplı alanda iletişim kurabilmektedir. Tüm ağ elamanları proje kapsamında özel olarak tasarlanmıştır. Her bir düğümün, teorik olarak, tasarlanan yeni TBL-MAC protokolüne uyum sağlayacak donanım alt yapısına sahip olduğu varsayılmıştır.

6.2. Paket yapıları

Token ve Veri(data) paketi olmak üzere iki farklı paket tasarımı gerçekleştirilmiş olup, bu paketlere ait veri yapıları aşağıda gösterilmiştir. Token paketleri ile ağ kurulumu için tüm düğümlerin bağlanabilirliği dolayısıyla iletişim hattının oluşabilirliği test edilmektedir. Şekil 14'te paket Opnet editöründe tasarlanmış olan Token paketi gösterilmektedir.

ptype (2 bits)	direction (1 bit)	src_id (8 bits)	dst_id (8 bits)	is_na (1 bit)	hop_count (2 bits)	nodes (0 bits)
-------------------	----------------------	--------------------	--------------------	------------------	-----------------------	-------------------

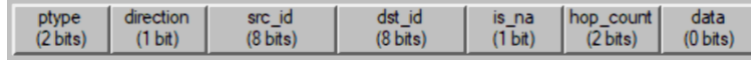
Şekil 14. Token paket yapısı

<i>ptype</i> :	2 bitlik paket tipinin tutulduğu alandır.
<i>direction</i> :	1bitlik paket yönü bilgisini tutan alandır.
<i>src_id</i> :	8bitlik gönderici düğüm bilgisi tutulur.
<i>dst_id</i> :	Alıcı düğümün 8 bitlik kimlik bilgisi tutulur.
<i>is_na</i> :	Negatif adresleme bilgisi tutulan 1bitlik alanıdır.

hop_count: 2bitlik atlama sayısı bilgisinin saklandığı alandır.

nodes: Başlangıçta 0 bit boyutludur. Token hat boyunca ilerledikçe düğümler bilgilerini bu listeye ekler ve uzunluğu artar. Burada saklanan bilgi daha önce de bahsedildiği üzere yapı(struct) türünde bir listedir.

Veri paketleri ise düğümlerin ortamdaki topladıkları verileri iletmek için tasarlanmıştır. Bu paketler iletim modlarına bağlı olarak CLN üzerinden SINK düğüme iletilmektedir. Şekil 15’de Veri paketi gösterilmektedir.



Şekil 15. Veri paketi yapısı

ptype: 2bitlik, paket tipinin tutulduğu alandır.

direction: 1bitlik paket yönü bilgisini tutan alandır.

src_id: 8 bitlik gönderici düğümün id bilgisi tutulur.

dst_id: Alıcı düğümün 8 bitlik kimlik bilgisi tutulur.

is_na: Negatif adresleme bilgisi tutulan 1 bitlik alanıdır.

hop_count: 2 bitlik atlama sayısı bilgisinin saklandığı alandır.

data: Boyut başlangıçta 0’dır. Bu alan düğümlerden toplanan verilerin tutulduğu bir listedir. Hat boyunca düğümler verilerini bu listeye ekler. Böylece paket boyutu artar.

6.3. Başarım analizi ve deneysel sonuçlar

TBL-MAC protokolünün başarım değerlendirilmesi için hazırlanan Ağ Proje ortamının benzetim parametreleri Tablo 1’de gösterildiği şekilde seçilmiştir.

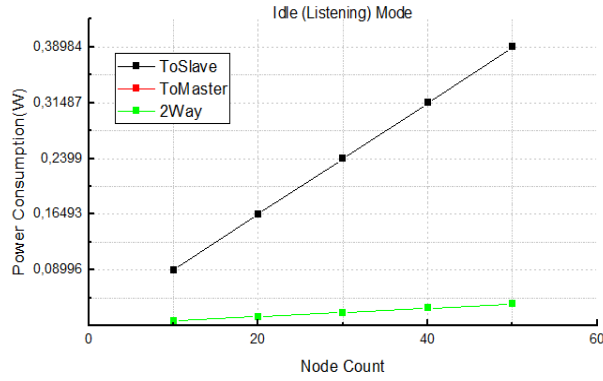
Tablo 1. Benzetim ölçütleri

Radyo yayılım gecikme modeli	dra_propdel
Modülasyon	bpsk
Radyo yayılım aralığı	125m ve 225m
İletim gücü	SR_tx = 0.027w
Alım gücü	Sr_rx = 0.0366w
LN sayısı	10, 20, 30, 40, 50
CLN sayısı	2
Sink sayısı	1
Düğüm yerleşimi	Doğrusal
Kullanılan kanal sayısı	1
Düğümler arası mesafe	100m
Benzetim süresi	25s

6.3.1. Enerji tüketimi değerlendirmesi

Sınırlı kaynaklara sahip algılayıcı düğümlerden oluşan DKAA için etkili güç tüketimi yönetimi en önemli başarım analizlerinden birisidir. Öyle ki petrol boru hattı izleme gibi uzun menzilli karasal uygulamalarda etkili güç tüketimi hayati öneme sahiptir.

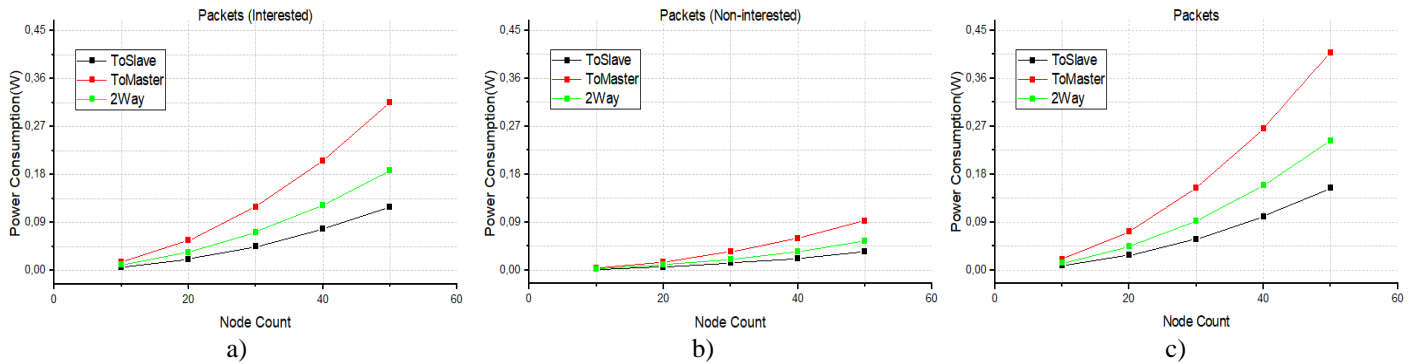
Çalışma kapsamında önerilen TBL-MAC protokolünün tüm güç tüketim maliyetleri analiz edilmiş, güç tüketimine sahip her faaliyet ayrı ayrı test edilmiştir. Şekil 16’da tüm düğümlerin boştaki enerji tüketim değerleri gösterilmiştir.



Şekil 16. Tüm düğümlerin Idle mod güç harcaması

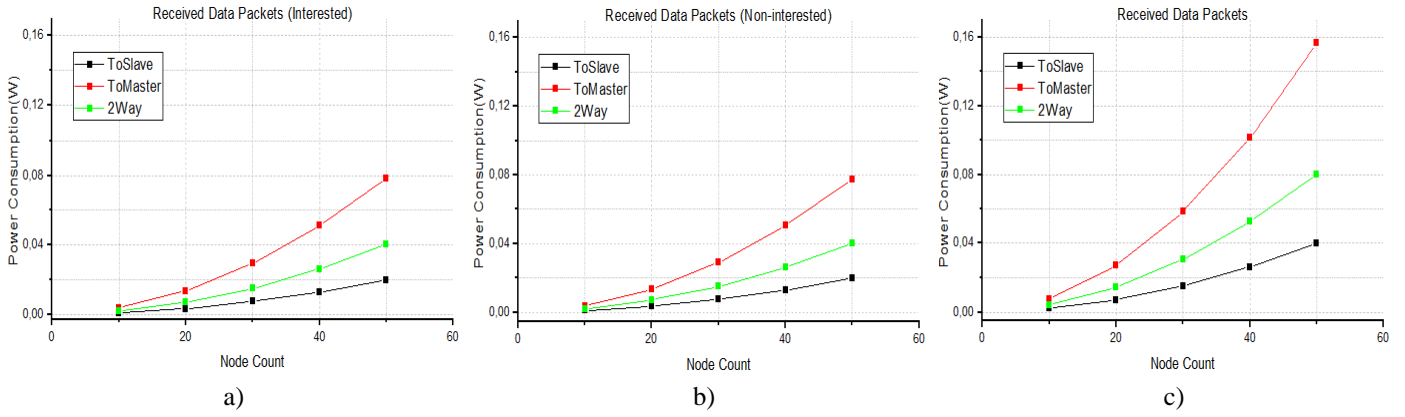
Şekil 16 incelendiğinde geliştirilen OEK protokolünün Dinleme (Idle) durumunda en az güç tüketimine sahip modun 2Way ve ToMaster metodu olduğu görülmekte olup, ToSlave yönteminin Dinleme durumunda daha fazla enerji tüketimi yaptığı gözlemlenmektedir. Aslında bu durum paket hareketliliği ile kolayca açıklanabilir. ToSlave yönteminde her bir periyottaki paket akışı tek yönlü olduğundan daha az sayıda paket iletilmekte ve bu nedenle düğümler daha çok boşa kalmaktadırlar. 2Way ve ToMaster metodunda ise her aktarım periyodu için çift yönlü olarak paket hareketliliği olduğundan düğümler Dinleme durumuna geçmek için daha az fırsat yakalamakta ve işlem yapmak üzere diğer durumlarda daha fazla kalmaktadırlar. Dolayısıyla en az paket hareketliliğine sahip olan ToSlave metodu bu anlamda daha fazla enerji harcar. Ancak ters bir bakış açısıyla bu durumun iyi olduğu söylenebilir. Zira Dinleme modundaki bir düğüm daima aktif moddaki bir düğümden daha az enerji harcayacaktır. Enerji tüketimi ile ilgili diğer grafiklerden de bu durum rahatlıkla görülebilir.

Farklı düğüm sayıları ve 3 farklı veri toplama metodu kullanımında senaryoların paket iletimi için harcadığı enerji durumları Şekil 17'de gösterilmiştir. Paket iletiminde enerji tüketimi en kötü olan durum, 50 düğümlü ToMaster metodunda gözlemlenmekte olup, harcadığı toplam enerji miktarı 0,42W seviyesindedir. Bu harcanan enerjinin oldukça ekonomik bir düzeyde olduğu gözlemlenmektedir. Bu durum düğüm ve işlem süreçlerinde enerji duyarlı tasarım yapılmış olması efektif TBL-MAC algoritmasıyla ilişkilidir.



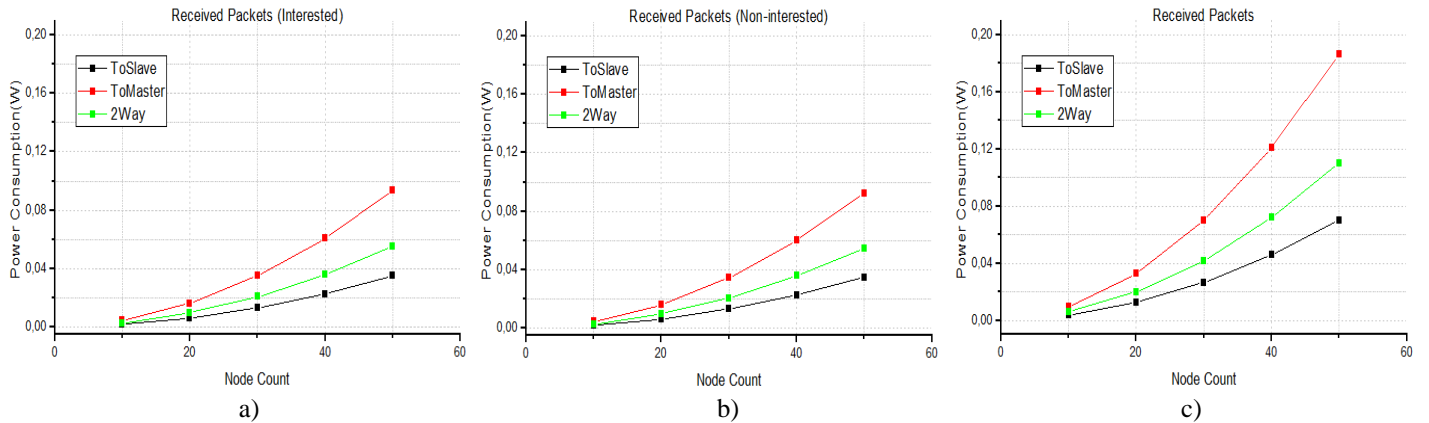
Şekil 17. a) Alakalı paketlerin güç harcaması b) Alakasız paketlerin güç harcaması c) Toplam paket iletimi güç harcaması

Şekil 17 a)'daki sadece düğümleri ilgilendiren paketlerin yaptığı güç harcaması incelendiğinde; ToMaster metodu en çok enerji tüketen metot olurken, ToSlave en az enerji harcayan metot olmuştur. Bu şekilde paket iletişimi için harcanan yalın enerji gözlemlenmiştir. Gereksiz paketler için yapılan güç harcaması Şekil 17 b)'de gösterilmiştir. Alakalı paket iletişiminde olduğu gibi ToMaster metodu en çok enerji tüketen metot olurken, ToSlave en az enerji harcayan metot olmuştur. Grafiklerden de görüldüğü üzere alakalı ve alakasız paketlerin enerji harcaması toplamı Şekil 17 c)'de gösterilen toplam paket iletişim güç harcamasına eşit olmaktadır. Tüm senaryolarda veri paketi alımı için harcadığı enerji durumları Şekil 18'teki grafiklerde gösterilmiştir.



Şekil 18. a) Alınan alakalı veri paketlerinin harcadığı güç b) Alınan alakasız veri paketlerinin harcadığı güç c) Alınan tüm veri paketlerinin harcadığı güç

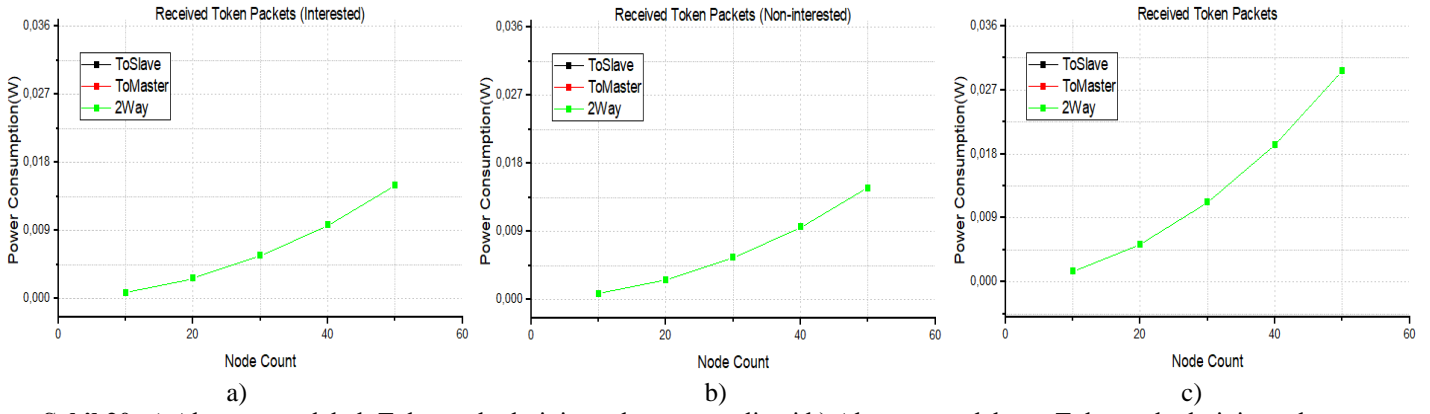
Düğümün sadece kendini alakadar eden veri paketleri alırken harcadığı enerji değerleri toplamı Şekil 18 a)'da gösterilmiştir. Önceki değerlerde olduğu gibi ToMaster en çok enerji harcayan veri toplama metodu olduğu görülmektedir. Şekil 18 b)'de alınan alakasız veri paketlerinin harcadığı güç incelendiğinde, aşağıdaki grafiklerden de görüleceği üzere TBL-MAC protokolü çalışma modlarına göre veri iletiminde de harcadığı enerjiyi mümkün olduğunca minimize etmekte pil ömrüne olumlu etki yapmaktadır. Şekil 18 c)'de tüm veri paketlerinin harcadığı toplam enerji miktarları görülmektedir. Bu değerler her bir düğümün aldığı tüm Veri paketlerine harcadığı enerji toplamından elde edilmiştir. Tüm düğüm sayılarında ToSlave metodunda en az enerji harcama performansı gösterirken, ToMaster metodunda en çok enerji harcama değerleri elde edilmiştir. Tüm senaryolarda paket türü dikkate alınmaksızın tüm paketler için harcanan enerji durumları Şekil 19'da grafiklerde gösterilmiştir.



Şekil 19. a) Alınan tüm alakalı paketlerin toplam güç maliyeti b) Alınan tüm alakasız paketlerin toplam güç maliyeti c) Alınan tüm paketlerin toplam güç maliyeti

Düğümün alınan tüm alakalı paketler için harcadığı toplam enerji değerleri Şekil 19 a)'da gösterilmiştir. Diğer tüm enerji harcamalarında olduğu gibi alakalı paket alımı durumu içinde ToMaster metodu en çok enerji harcayan, ToSlave en az enerji harcayan metot olduğu görülebilmektedir. Şekil 19 b)'de alakasız paketlerin harcadıkları güç değerleri verilmiştir. Bu değerler düğümleri alakadar etmeyen ve komşu düğümlerin yayın(broadcast) yaptığı paketlerden gelmektedir. Düğümler negatif adresleme sayesinde bu paketlerin kendisini alakadar etmediğini tespit ederek daha fazla işlem yapmamaktadır. Düğümlerin tüm aldığı paketler için harcadığı toplam enerji değerleri Şekil 19 c)'de gösterilmiştir. Diğer tüm enerji harcamalarında olduğu gibi paket alımı durumu içinde ToMaster en çok enerji harcayan, ToSlave en az enerji harcayan metot olmuştur.

Token paketleri ile tüm senaryolarda eşit paket sayıları kullanılarak ağın kurulması sağlanmıştır. Önerilen protokol için Token paketleri alımı enerji harcama grafikleri Şekil 20'de verilmiştir.

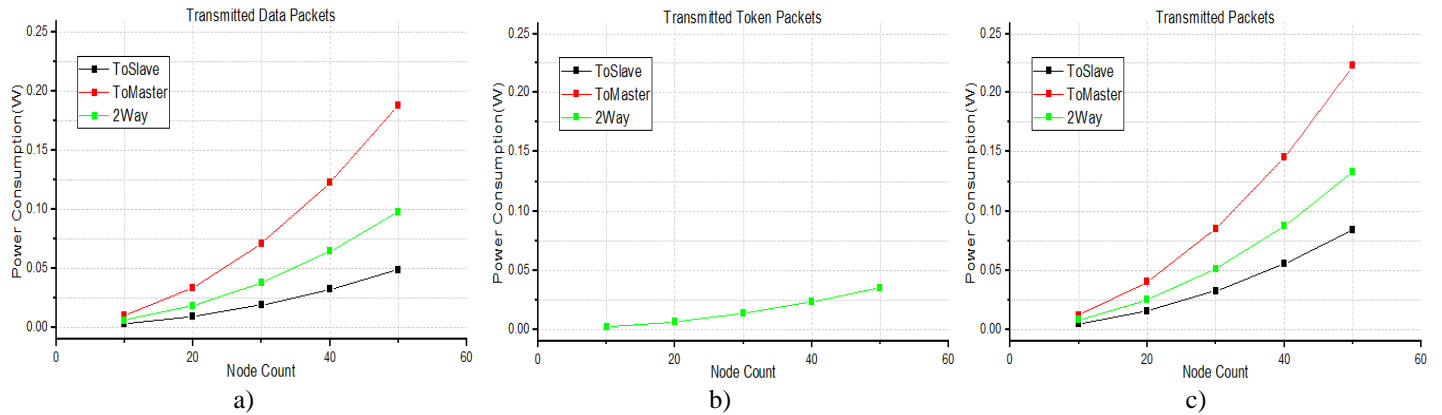


Şekil 20. a) Alınan tüm alakalı Token paketlerinin toplam güç maliyeti b) Alınan tüm alakasız Token paketlerinin toplam güç maliyeti c) Alınan tüm Token paketlerinin toplam güç maliyeti

Protokolde kullanılan ve iletişimin düzgün yapılmasını sağlayan Token paketlerinin her modda aynı büyüklüğe sahip olduğu, dolayısı ile harcadıkları toplam güç miktarının hem çok küçük hem de eşit olduğu Şekil 20'deki grafiklerden görülmektedir. Düğüm sayısı arttıkça Token boyutunun artması daha fazla enerji harcanmasına neden olmaktadır. Grafikteki düğüm sayısına bağlı olarak enerjide görülen artış Token paket sayısı ile değil ilgili paketin boyutunun artması ile alakalıdır. Şekil 21'deki paket sayıları grafiklerine bakıldığı zaman değerlerin tutarlı olduğu görülebilmektedir. Tüm senaryolar için veri paketi gönderimi için harcanan enerji durumları Şekil 21 a)'da gösterilmiştir. Bu değerler her bir düğümün gönderdiği tüm veri paketlerine harcadığı enerji toplamından elde edilmiştir.

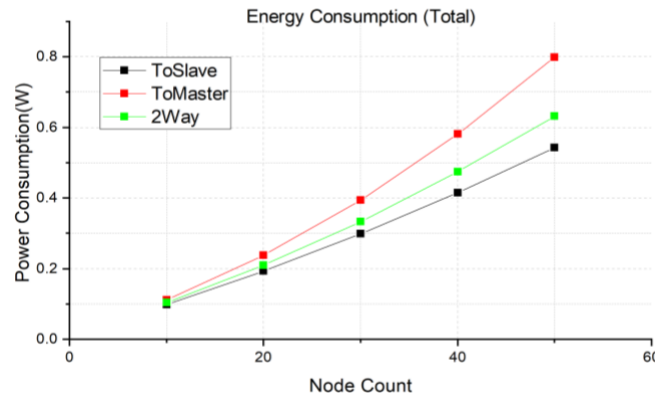
Veri paketi alınırken olduğu gibi tüm düğüm sayılarında veri paketi gönderiminde de ToSlave metodunda en az enerji harcama performansı gösterirken, ToMaster metodunda en çok enerji harcama değerleri elde edilmiştir. Token paketi gönderimi için düğümlerin harcadığı toplam enerji değerleri Şekil 21 b)'de gösterilmiştir.

Şekil 21 b)'de de görüleceği üzere ToMaster, ToSlave ve 2Way metodlarında harcanan toplam güç eşit olarak görülmektedir. Çünkü bu yöntemler veri toplama yöntemidir ve oysa benzetim boyunca aynı sayıda Token kullanılarak iletişim hattı kurulmaktadır. Öte yandan düğümlerin veri paketleri dâhil tüm gönderdiği paketler için harcadığı toplam enerji değerleri Şekil 21 c)'de gösterilmiştir.



Şekil 21. a) Gönderilen Veri paketlerin toplam güç, b) Gönderilen Token paketlerin için harcama toplam güç, c) Gönderilen tüm paketlerin toplam güç maliyeti

Paket gönderiminde, gönderilen her paket alakalı paket olarak değerlendirilir. Dolayısı ile paket alınırken olduğu gibi alakalı-alakasız paket ayrımı söz konusu değildir. Diğer tüm enerji harcamalarında olduğu gibi paket iletimi için de ToMaster metodu en çok enerji harcayan metot olmuştur. ToSlave en az enerji harcayan metot olarak görülebilmektedir. Son olarak tüm paket işlemleri ve dinleme durumu dâhil toplam enerji harcamaları grafiği Şekil 22'de gösterilmiştir.



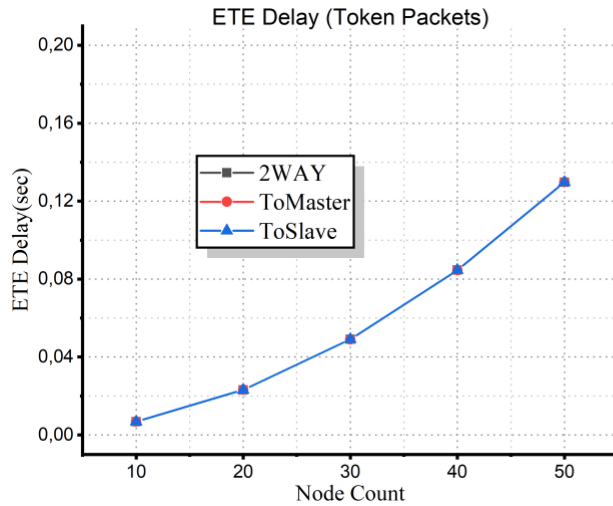
Şekil 22. Toplam güç tüketim maliyeti

Diğer tüm sonuçlarda olduğu gibi toplam enerji harcamalarında da en çok güç tüketen metot ToMaster olur iken, ToSlave metodu en az güç tüketen metot olmuştur.

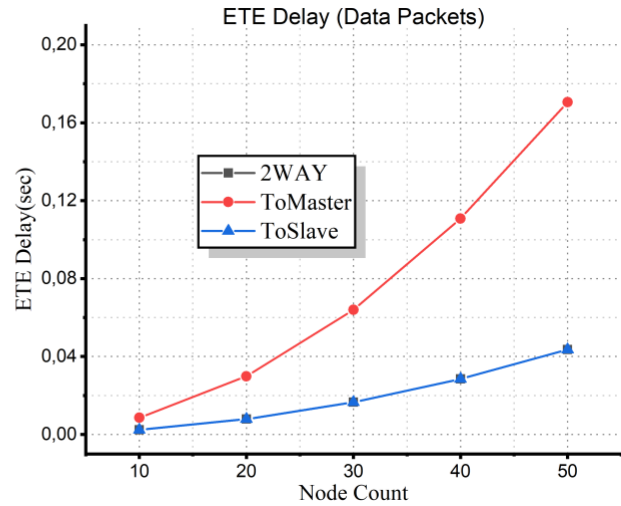
6.3.2. Uçtan uca gecikmenin değerlendirilmesi

Şekil 23'te token paketleri için uçtan uca gecikme grafiği ve gösterilmektedir. TBL-MAC protokolünün Token paketlerinin uçtan uca gecikme metriği; CLN_{Master} 'ın Token paketi göndermesinden itibaren paketin tüm ağı dolaşıp tekrar CLN_{Master} düğüme dönmeye kadar geçen zaman olarak hesaplanmıştır. Veri paketleri için uçtan uca gecikme metriği ise; CLN_{Master} veri paketi göndermesinden başlayıp paketin tüm ağı dolaşıp SINK düğüme ulaşma zamanı olarak hesaplanmıştır. ToSlave ve 2Way metotlarında veri paketi ağı 1 kere dolaşıp SINK'e ulaşmaktadır. ToMaster metodunda ise veri paketi ağda gidiş-dönüş yapıp SINK'e ulaşmaktadır. Bundan dolayı göre daha fazla uçtan uca gecikmeye sahiptir.

Şekil 24'te ise veri paketlerinin uçtan uca gecikme performansları Şekil 24'te görüldüğü üzere ToMaster metodu diğer iki metoda



Şekil 23. Token paketleri için uçtan uca gecikme

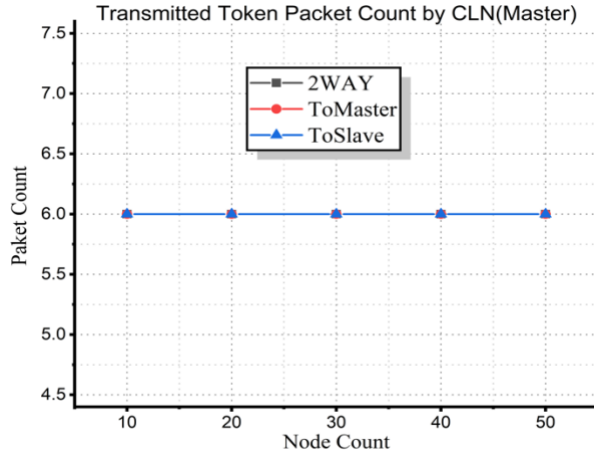


Şekil 24. Veri paketleri için uçtan uca gecikme

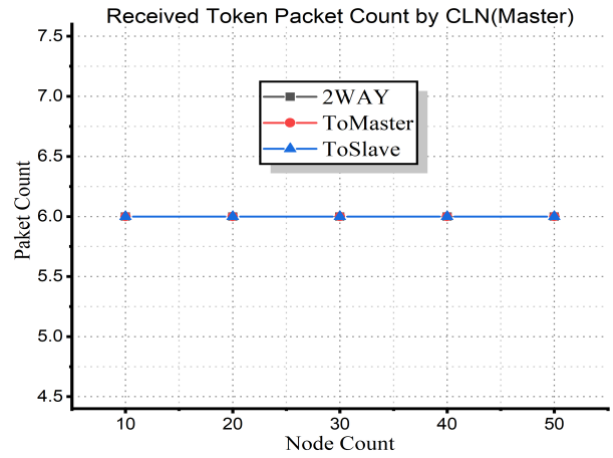
ToSlave, ToMaster ve 2Way veri toplama metodudur. Dolayısıyla TBL-MAC protokolü algoritması gereği Token paketler etkilenmemesi gerekir. Şekil 23'te de görüldüğü üzere, tüm senaryolarda eşit gecikme olduğundan algoritmanın doğru çalıştığı söylenebilir.

6.3.3. Paket iletim oranının değerlendirilmesi

Şekil 25'de CLN_{Master} tarafından gönderilen toplam Token paket sayıları, Şekil 26'de de alınan Token paketi sayıları gösterilmiştir. Tüm düğüm senaryolarında 6 adet Token paketi gönderilmiş ve aynı sayıda Token paketi geri alınmıştır.

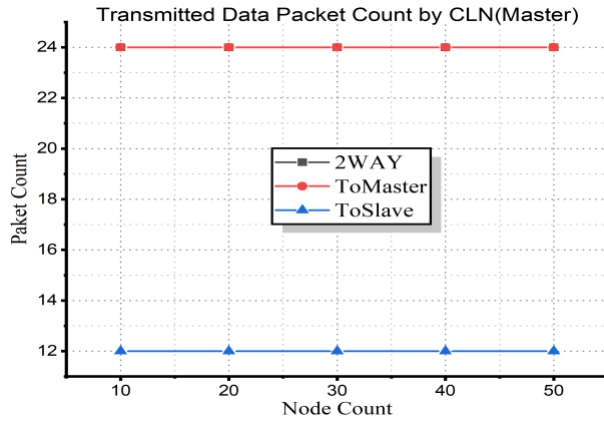


Şekil 25. CLN_{Master} tarafından gönderilen Token paket sayıları

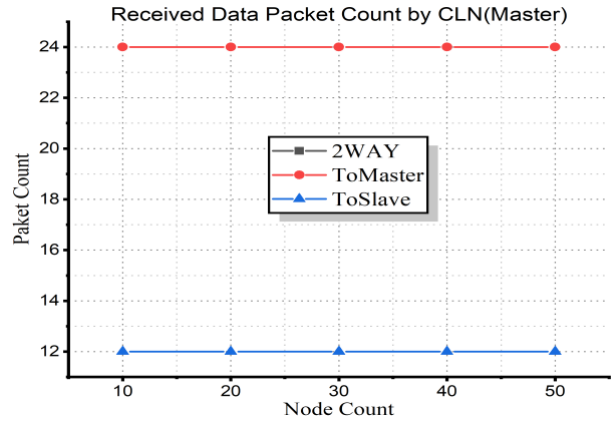


Şekil 26. CLN_{Master} tarafından alınan Token paket sayıları

Tüm bu sonuçlar Token paket iletiminde önerilen TBL-MAC protokolünün hatasız olarak çalıştığı göstermektedir. Dolayısıyla token paketleri için veri iletim oranı benzetim boyunca %100 olmuştur. Öte yandan Şekil 27 ve Şekil 28’de tüm senaryolar için gönderilen ve alınan toplam veri paket sayıları görülmektedir.



Şekil 27. CLN_{Master} tarafından gönderilen Veri paket sayıları

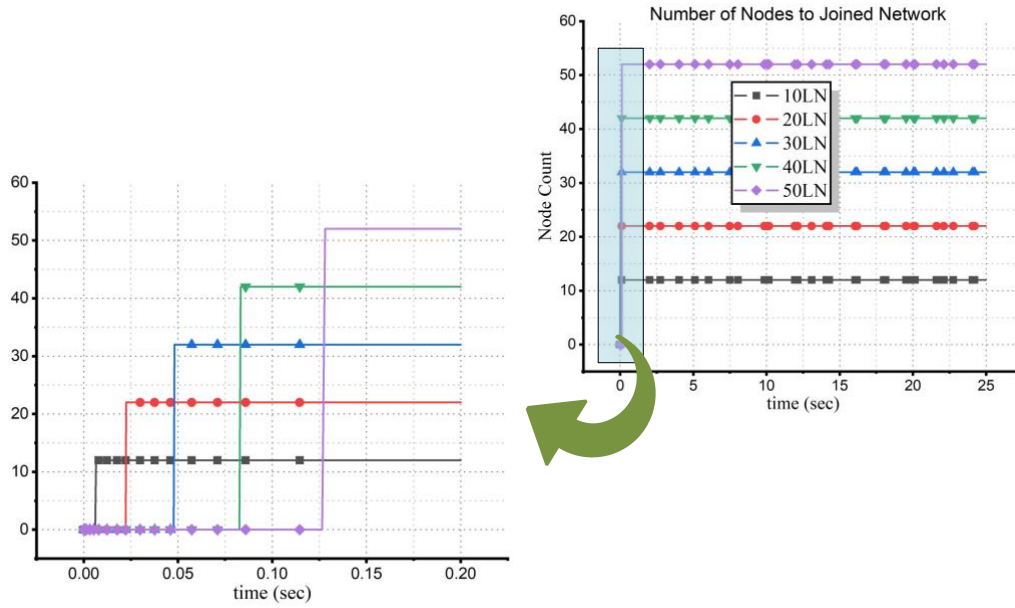


Şekil 28. CLN_{Master} tarafından alınan Veri paket sayıları

Tüm düğüm senaryolarında benzetim süresince 2Way ve ToMaster için 24 adet, ToSlave metodu için 12 adet veri paketi gönderilmiştir. Gönderilen ve alınan paket sayıları kıyaslandığında veri paketi iletiminde TBL-MAC protokolünün hatasız olarak çalıştığı görülmektedir.

6.3.4. Düğümlerin bağlantı sürekliliği analizi

DKAA düğümlerin ağa bağlanıp ağdan düşmeden görevlerini yerine getirmesi ağ performansı açısından büyük öneme sahiptir. Özellikle DKAA’ların lineer topolojik dizilimlerinden dolayı bu durum çok daha hayati bir hal alır. Bir düğümün başarısız olmasının tüm ağı etkilemesi söz konusudur. Şekil 29’da proje kapsamında geliştirilen protokolün ağ bağlantı performansları verilmiştir.



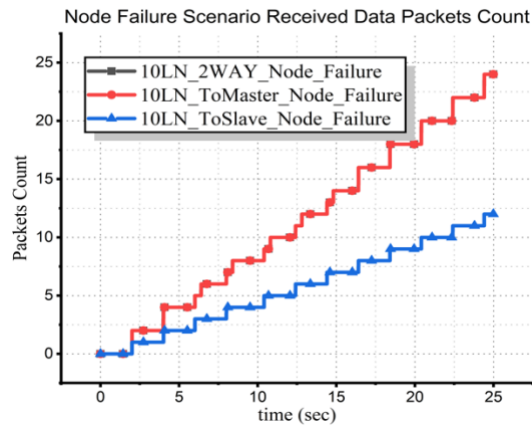
Şekil 29. Düğümlerin ağda bağlantı sürekliliği

Şekil 29'da da görüleceği üzere; senaryo adları LN düğüm sayılarına göre verilmiştir ve her senaryoda iki adet CLN de olduğundan grafikte düğüm sayıları her bir senaryoda +2 adet olarak bulunmaktadır. Ağın oluşması; 10 düğümlü senaryoda 0.00652742042 saniyede, 20 düğümlü senaryoda 0.022510019 saniyede, 30 düğümlü senaryoda 0.048102335 saniyede, 40 düğümlü senaryoda 0.083304177 saniyede ve 50 düğümlü senaryoda ise 0.128115648 saniyede gerçekleşmiştir.

Buna göre düğümler tüm senaryolarda kısa sürede ağa bağlanıp hattan kopmadan benzetimi başarı ile tamamlamıştır. Tüm sonuçlar göstermiştir ki TBL-MAC, ağ performansını olumsuz etkileyen en önemli faktörlerden olan bağlantı hatası konusunda önemli derecede başarılı olmuştur ve benzetim süresince düğümler ağdan kopmadan işlemlerine devam etmişlerdir.

6.3.5. Başarısız düğüm durum davranışı

Bu başlık altında herhangi bir nedenden dolayı ağdan düşen (kopan) bir düğümün belirli bir süre sonra tekrar ayağa kalkması ile geliştirilen TBL-MAC protokolünün bu durum karşısındaki tepkiselliği (Çıbuk, Arı, Ağgün, & Budak, 2021) ve davranış biçimi irdelenmiştir. Bu kapsamda hat üzerinde tek düğümlü düşüşler ele alınmıştır. Ardışık çok sayıda düğüm hatası (2 ve daha fazla veya hata patlaması) durumları bu çalışma kapsamında ele alınmamıştır. Yapılan testlerde ağdan kopan düğüm olmasına karşın veri paketi iletiliminden herhangi bir sorun oluşmadığı Şekil 30'da görülebilmektedir. Ağdan düğüm kopması ile komşu düğümler kapsama alanını iki katına çıkararak iletişimi sürdürmüş ve ağda bağlantı sürekliliği sağlanmıştır. Sonuçlar TBL-MAC protokolünün düğüm başarısızlığı testinden başarı ile geçtiğini göstermiştir.



Şekil 30. Düğüm başarısızlığı testi (Veri Paketi Sayısı)

7. Sonuç ve Tartışma

Bu çalışma kapsamında DKAA ağlar için TBL-MAC adında yeni bir OEK protokolü geliştirilmiştir. Ayrıca yapılan benzetim çalışmalarıyla gerçekleştirilen protokolün başarımları ölçülmüştür.

Geliştirilen (TBL-MAC) OEK protokolü; özellikle şehir içi ve şehirlerarası karayollarındaki yol kenarı aydınlatmalarında kontrol ve enerji tasarrufu konuları dikkate alınarak önerilmiştir. Etkin ve verimli bir OEK tasarımı için literatürdeki benzerlerinden farklı olarak paket tekrarları ve özelliklede yankılanma sorunlarına karşı yeni ve etkin bir yöntem olan “Negatif Adresleme” kullanılmış olup bu yaklaşımın etkinliği deneysel analizlerde de açıkça test edilip gözlemlenmiştir. Negatif Adresleme yaklaşımı sayesinde gönderilen paketlerin hedefine iletilmesi garanti edilmekte olup bunun için ekstra bir onay veya paket kullanılmamaktadır. Ayrıca enerji verimliliği ve ağdaki kötü durum senaryolarının üstesinden gelebilmek için farklı çalışma modları (ToSlave, ToMaster ve 2Way) geliştirilerek veri toplama işlemleri gerçekleştirilmiştir. Bu veri toplama metotları arasında ToSlave metodu uçtan uca gecikme ve enerji tüketimin en düşük olduğu metod olarak gözlemlenmiştir. Özellikle yoğun veri trafiğine ihtiyaç duyulmayan ve enerji kaynağı sınırlı olan DKAA’lar için çok uygun bir yaklaşım olarak karşımıza çıkmaktadır. ToMaster metodu uçtan uca gecikmenin yüksek olduğu görülmekle beraber bir veri toplama periyodunda düğümlerden iki defa veri alması nedeni ile daha veri yoğun uygulamalar için kullanışlı olduğu görülmektedir. Öte yandan 2Way metodunda ise çift yönlü veri aktarımı açısından özellikle veri yoğun ve hatta gerçek zamanlı olabilecek iletişime daha uygun olmakla beraber uçtan uca gecikme değerleri de düşüktür. Bu metodun tek dezavantajı her iki CLN’nin de SINK düğüme bağlı olması olarak görülebilir belki ama yedeklilik açısından bakıldığında bu durumun aslında kötü durum senaryolarında etkin bir çözüm yolu sunduğu da görülebilir.

Bu çalışmada TBL-MAC protokolünde, benzetim çalışmalarında 10, 20, 30, 40 ve 50 tüm düğüm sayılarında sadece 6 adet Token paketi kullanılarak iletişim hattının kurulması sağlanmıştır. Token yöntemi düğüm sayısından yada veri toplama metotlarından bağımsız olarak çalışmaktadır. Böylece az sayıda token paketi kullanılarak iletişim hattının kurulması sağlanmıştır.

DKAA’lar için tasarlanan OEK protokollerinin en önemli özelliklerinden biriside düğüm başarısızlığı durumlarına duyarlı olması gerekmektedir. TBL-MAC protokolünde ise herhangi bir düğüm ağdan düştüğünde, komşu düğümler kapsama alanını artırmak suretiyle iletişimin devamlılığını sağlamaktadır. Bu mekanizmanın TBL-MAC için, deneysel sonuçlar ile başarılı bir şekilde çalıştığı görülmüştür.

Yine aynı şekilde ağı hızlı organize olup iletişimin başlaması için TBL-MAC etkili ağa katılım mekanizmasına sahiptir. Bu mekanizma Token paketleri ile sağlanmaktadır. Sonuçlarda da görüldüğü üzere TBL-MAC protokolünde ağın oluşması tüm senaryolarda çok kısa sürelerde gerçekleşmiştir. Benzetim boyunca ağdan kopma olmadan iletişim devam etmiştir.

Sonuç olarak tüm bu yönleriyle, TBL-MAC OEK protokolünün literatüre etkin bir katkı sağlayacağı düşünülmektedir. Öte yandan bu çalışmanın odağı kapsamında olmayıp gelecek çalışmalar için; geliştirilen protokolün ToMaster, ToSlave ve 2Way modları arasında ihtiyaç durumuna göre dinamik geçişler yapabilecek bir yapıya kavuşturulması ve çoklu düğüm hataları (hata patlamaları) karşısında protokolün davranışının geliştirilmesi düşünülmektedir.

Teşekkür

Bu çalışma Bitlis Eren Üniversitesi Bilimsel Araştırma Projeleri Birimi tarafından BEBAP2019.006 proje numarası ile desteklenmiştir. Bitlis Eren Üniversitesi Bilimsel Araştırma Projeleri Birimi’ne katkı ve desteklerinden dolayı teşekkür ederiz.

Referanslar

Ari, D., Cibuk, M., & Ağgün, F. (2018). A New Time Division Method for Linear Wireless Sensor Networks. In *1st INTERNATIONAL ENGINEERING AND TECHNOLOGY SYMPOSIUM* (pp. 1166–1170). Batman, Turkey.

Caneva, D. De. (2010). A Synchronous and Deterministic MAC Protocol for Wireless Communications on Linear Topologies. *Int’l J. of Communications, Network and System Sciences*, 03(12), 925–933. <https://doi.org/10.4236/ijcns.2010.312126>

Chen, H., Tse, C. K., & Feng, J. (2009). Impact of topology on performance and energy efficiency in wireless sensor networks for source extraction. *IEEE Transactions on Parallel and Distributed Systems*, 20(6), 886–897. <https://doi.org/10.1109/TPDS.2009.14>

Çibuk, M., Ari, D., Ağgün, F., & Budak, Ü. (2021). Investigation of failed node method to support healthy communication for linear wireless sensor networks. In *1st Advanced Engineering Days (AED)* (pp. 41–46). Mersin, Turkey. Retrieved from <http://aed.mersin.edu.tr/wp-content/uploads/2021/12/AED1.pdf>

- Du, S., Saha, A. K., & Johnson, D. B. (2007). RMAC: A Routing-Enhanced Duty-Cycle MAC Protocol for Wireless Sensor Networks. In *IEEE INFOCOM 2007 - 26th IEEE International Conference on Computer Communications* (pp. 1478–1486). IEEE. <https://doi.org/10.1109/INFCOM.2007.174>
- Fang, C., Liu, H., & Qian, L. (2011). LC-MAC: An efficient MAC protocol for the long-chain wireless sensor networks. *Proceedings - 2011 3rd International Conference on Communications and Mobile Computing, CMC 2011*, 495–500. <https://doi.org/10.1109/CMC.2011.65>
- Jawhar, I., & Mohamed, N. (2009). A hierarchical and topological classification of linear sensor networks. In *2009 Wireless Telecommunications Symposium, WTS 2009*. IEEE. <https://doi.org/10.1109/WTS.2009.5068941>
- Jiang, H., Wu, J., Chen, L., Chen, S., & Leung, H. (2009). A Reliable and High-Bandwidth Multihop Wireless Sensor Network for Mine Tunnel Monitoring. *IEEE Sensors Journal*, 9(11), 1511–1517. <https://doi.org/10.1109/JSEN.2009.2022878>
- Karveli, T., Voulgaris, K., Ghavami, M., & Aghvami, A. H. (2009). DiS-MAC: A MAC protocol for sensor networks used for roadside and highway monitoring. *2009 International Conference on Ultra Modern Telecommunications and Workshops*. <https://doi.org/10.1109/ICUMT.2009.5345439>
- Lai, T. T. Te, Chen, W. J., Li, K. H., Huang, P., & Chu, H. H. (2012). TriopusNet: Automating wireless sensor network deployment and replacement in pipeline monitoring. *IPSN'12 - Proceedings of the 11th International Conference on Information Processing in Sensor Networks*, 61–71. <https://doi.org/10.1145/2185677.2185686>
- Lee, E., Jwa, J. W., & Kim, H. (2013). MFT-MAC: A duty-cycle MAC protocol using multiframe transmission for wireless sensor networks. *International Journal of Distributed Sensor Networks*, 2013. <https://doi.org/10.1155/2013/858765>
- Low, K. S., & Talampas, M. C. R. (2017). Wireless sensor networks for intelligent transportation applications: A survey. *Industrial Wireless Sensor Networks: Applications, Protocols, and Standards*, 47–78. <https://doi.org/10.1201/b14072>
- Martin, K. M., & Paterson, M. B. (2009). Ultra-lightweight key predistribution in wireless sensor networks for monitoring linear infrastructure. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 5746 LNCS(May 2014), 143–152. https://doi.org/10.1007/978-3-642-03944-7_11
- Nordic Semiconductor. (n.d.). Nordic Semiconductor. Retrieved October 28, 2021, from <https://www.nordicsemi.com/-/media/Publications/WQ-Product-guide/Wireless-Q---Q1---Product-Guide.pdf?la=en&hash=55195302B5331ECA0F7D841B58D7F97DCC945F26>
- Riverbed Modeler. (n.d.). Riverbed Modeler. Retrieved April 9, 2021, from <https://www.riverbed.com/gb/products/steelcentral/steelcentral-riverbed-modeler.html>
- Saeed, H., Ali, S., Rashid, S., Qaisar, S., & Felemban, E. (2014). Reliable monitoring of oil and gas pipelines using wireless sensor network (WSN) - REMONG. *Proceedings of the 9th International Conference on System of Systems Engineering: The Socio-Technical Perspective, SoSE 2014*, 230–235. <https://doi.org/10.1109/SYSOSE.2014.6892493>
- Sun, T., Yan, X., & Yan, Y. (2013). A chain-type wireless sensor network in greenhouse agriculture. *Journal of Computers (Finland)*, 8(9), 2366–2373. <https://doi.org/10.4304/jcp.8.9.2366-2373>
- Sun, Y., Du, S., Gurewitz, O., & Johnson, D. B. (2008). DW-MAC: a low latency, energy efficient demand-wakeup MAC protocol for wireless sensor networks. *Proceedings of the 9th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, 53–62. <https://doi.org/10.1145/1374618.1374627>
- Varshney, S., Kumar, C., & Swaroop, A. (2015). Linear sensor networks: Applications, issues and major research trends. *International Conference on Computing, Communication and Automation, ICCCA 2015*, 446–451. <https://doi.org/10.1109/CCAA.2015.7148418>