

# Telsiz Duyurga Ağları için Enerji Etkin Dağıtık Öz Kararlı Maksimal Bağımsız Küme Algoritmaları

**Özkan Arapoğlu, Orhan Dağdeviren**

Ege Üniversitesi, Uluslararası Bilgisayar Enstitüsü, İzmir

[ozkanarapoglu@hotmail.com](mailto:ozkanarapoglu@hotmail.com), [orhan.dagdeviren@ege.edu.tr](mailto:orhan.dagdeviren@ege.edu.tr)

**Özet:** Telsiz duyurga ağları (TDA'lar), enerji kısıtlı duyurga düğümlerinden oluşur. Bu yüzden ağın yaşam ömrünü uzatmak için enerjiyi dikkatlice yönetmek gerekir. Kümeleme, TDA'larda yaşam süresini ve ölçeklenebilirliğini artıran etkin bir yöntemdir. Birbiriyle komşu olmayan düğümlerin oluşturduğu ve yeni bir eleman eklenip genişletilemeyen kümeye maksimal bağımsız küme (MBK) denir. Dağıtık MBK algoritması tasarlamak, TDA'larda küme liderlerini seçmek için önemlidir. Bilgilerimiz dâhilinde literatürdeki dağıtık öz kararlı MBK algoritmaları küme elemanlarını seçerken sadece düğüm kimliklerini değerlendirirler. Dağıtık bir sistem, başlangıçta herhangi bir durumdan başlayıp sınırlı zaman içinde kararlı hale geliyorsa ve dışsal bir müdahale olmadığı sürece kararlı yapıda kalabiliyorsa öz kararlıdır. Bu makalede, ağın yaşam süresini uzatmak amacıyla ağın küme liderlerini içeren MBK'yı enerji etkin seçmek için sadece kimlik numarasına değil düğümlerin derecesine, düğümlerin kalan enerjisine ve her ikisine de önem veren üç yeni algoritma tasarlandı. Önerilen algoritmaların teorik analizi yapıldı ve performansları TOSSIM benzetim ortamında değerlendirilmiş ve etkinliği gösterilmiştir.

**Anahtar Sözcükler:** Telsiz Duyurga Ağları, Öz Kararlı Algoritmalar, Maksimal Bağımsız Küme Algoritmaları, Dağıtık Algoritmalar, Enerji Etkin Küme Lideri Seçimi

**Abstract:** Wireless sensor networks (WSNs), consists of sensor nodes with limited energy power. So, It is necessary to carefully manage this energy to prolong the network lifetime. Clustering is an effective method for increasing the scalability and lifetime of WSNs. A maximal independent set (MIS) is a set of non-neighbor nodes which cannot be extended by adding a new node. A distributed system is self-stabilizing if it can start at any state and regain a legal state in a finite time without any external intervention. Design of a distributed self-stabilizing MIS algorithm is important for selection of cluster heads in WSNs. To the best of our knowledge, existing distributed self-stabilizing MIS algorithms only consider node id during the selection of nodes. In this paper, three new algorithms are designed which not consider only node id but also degree of the nodes, residual energy of the nodes and both of them to choose energy efficient MIS that contains cluster heads of the network to prolong the network lifetime. Theoretical analysis of the proposed algorithms is provided, their performances are evaluated in TOSSIM simulation environment and their efficiencies are shown.

## 1. Giriş

Telsiz Duyurga Ağı (TDA), fiziksel ortam ile dijital dünya arasında iletişim sağlayabilen ve fiziksel ortamdan topladığı verileri işleyebilen duyurga düğümlerinin (İng. Sensor Node) sabit bir altyapı olmadan bir araya gelmesiyle oluşturdukları ağıdır. Duyurga ağlarının, ortamdaki sıcaklık, basınç, nem,

hareket, ses ve ışık gibi değerleri ölçebilen duyurga düğümlerine sahip olması farklı uygulama alanlarında kullanılabilmelerini sağlar [12]. TDA içindeki bir duyurga düğümü; mikroişlemci, telsiz (İng. Transceiver), bellek, güç kaynağı ve duyurga bileşenlerinden oluşur. Kısıtlı bellek alanına, işlem gücüne ve enerjiye sahip olan bir duyurga düğümü üzerinde çalışan

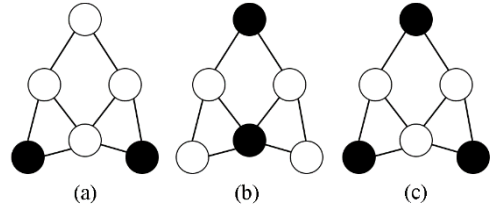
algoritmanın yüksek enerji verimliliği ve hata toleransı sağlaması, ağır yaşam süresini uzatır.

Öz kararlılık kavramı ilk olarak Dijkstra [4] tarafından tanımlanmıştır. Dağıtık bir sistem, harici bir müdahale olmaksızın herhangi bir başlangıç durumundan başlayıp, sınırlı bir zaman içinde kararlı duruma geçiyorsa öz kararlıdır. Öz kararlı algoritmalar bir ağdaki hata toleransını artırmayı hedefler [5]. Dağıtık öz kararlı (İng. self stabilizing) bir sistemde her düğüm genellikle aynı programı çalıştırır ve eş zamanlı olarak düğümler durumunu değiştirebilir. Her düğüm, sahip olduğu kurallarının ön şartlarını kontrol eder. Eğer herhangi bir düğümün kurallarından bir tanesinin ön şartı sağlanıyorsa, bu düğüm öncelikli (İng. Privileged) düğüm olarak adlandırılır. Eğer bir öncelikli düğüm durumunu değiştirirse, bu durum bir hareket (İng. Move) olarak adlandırılır. Kuralların atomik olarak çalıştırıldığı varsayılır. Düğüm hareket yaptığında komşu düğümlere durumunu mesajla bildirir. Telsiz sistemlerde enerji tüketimindeki baskın faktör iletişim olduğundan dolayı öz kararlı algoritmaların hareket sayısının düşürülmesi önemlidir.

Öz kararlı algoritmalar takvimleyici ile (İng. Scheduler) modellenmektedir. Takvimleyici merkezi ve dağıtık olmak üzere iki çeşittir. Merkezi takvimleyici aynı anda öncelikli düğümlerden sadece birisini seçer ve onun hareket edebilmesine izin verir. Dağıtık takvimleyici ise aynı anda birden fazla öncelikli düğümün seçilebilmesini ve hareket edebilmesini sağlar. Dağıtık takvimleyicilerde kendi içinde iki alt gruba ayrılır: Senkron dağıtık takvimleyiciler, aynı anda tüm öncelikli düğümleri seçer ve hareket etmelerini sağlar; adil olmayan (İng. Unfair) dağıtık takvimleyici ise öncelikli düğümlerin boş küme olmayan alt kümelerinden herhangi birini seçer ve bu düğümlerin aynı anda hareket etmelerine izin verir. Adil olmayan dağıtık takvimleyici ise daha az gereksinimi olduğu ve dağıtık asenkron tasarımıyla uyumlu olduğu için

dağıtık sistemlerde kullanılmaya daha uygun bir takvimleyici türüdür. Bu nedenle tasarladığımız algoritmalarda adil olmayan takvimleyici kullanılmıştır.

Çevreden toplanan verilerin enerji verimliliğini sağlayacak şekilde çıkış düğümüne gönderilmesini sağlayacak etkin bir ağ topolojisi kurmak TDA'daki en önemli problemlerden biridir. Etkin bir ağ topolojisi oluşturmanın önemli yöntemlerinden birisi de kümeleme işlemidir [1-2]. Kümeleme, bir ağı bağlı alt kümelere bölme ve bölünen her alt kümeyle bir küme lideri atama işlemidir. Kümeleme yöntemi ile TDA'da veri birleştirme, yönlendirme, yük dengesinin sağlanması, zaman senkronizasyonu, topoloji kontrolü, hata toleransı, güvenli iletişimin sağlanması ve ağır yaşam süresinin uzatılması işlemleri daha etkin ve kolay şekilde yapılır.



Şekil 1: a) Bağımsız küme b) Maksimal bağımsız küme c) Maksimum bağımsız küme

$V$  düğümlerine ve  $E$  kenarlarına sahip olan bir  $G$  çizgesinin, herhangi iki düğümü arasında hiçbir bağlantı kenarı bulunmayan  $V'$  alt kümesine bağımsız küme (İng. independent set) denir. Eğer bir bağımsız kümeyle yeni bir eleman eklemek mümkün değilse bu kümeyle maksimal bağımsız kümenin (MBK'nın) büyüklüğü (İng. Size) içerdiği düğüm sayısı kadardır ve  $\alpha(G)$  simgesi ile gösterilir.  $G$  çizgesinin büyüklüğü en fazla olan MBK'ya maksimum bağımsız küme denir. Maksimum bağımsız kümeyi bulmak NP-Zor problemdir. Şekil 1.a'da örnek bir bağımsız kümeyi, Şekil 1.b'de örnek bir MBK'yı, Şekil 1.c'de ise örnek bir maksimum bağımsız kümeyi görebiliriz. Siyah renkli düğümler bağımsız küme içerisindeki düğümleri göstermektedir. MBK telsiz duyurga ağlarında kümeleme

problemini çözmek, yeni ağ yapıları inşa etmek, gezgin (İng. Mobile) ağlarda değişen ağ yapısına göre sistemin yeniden kurulmasını sağlamak, enerji verimliliğini ve hata toleransını artırarak ağın yaşam ömrünü uzatmak gibi birçok problemin çözümünde kullanılır [6].

Bu çalışmada, kablosuz ağlarda ağın yaşam süresini uzatmak için enerji etkin dağıtık öz kararlı MBK algoritması tasarlanmıştır. Bu algoritma “Enerji Etkin MBK ( $E_{MBK}$ )” olarak adlandırılmıştır. Algoritmanın üç versiyonu vardır. Enerji öncelikli  $E_{E-MBK}$  algoritmasında, küme lideri seçmek için öncelikli olarak düğümlerin kalan enerjileri dikkate alınmıştır. Daha sonra sırasıyla derece ve kimlik dikkate alınmıştır. Derece öncelikli  $E_{D-MBK}$  algoritmasında, derecesi büyük olan düğüm ilk olarak tercih edilmiştir. Daha sonra sırası ile kalan enerji miktarının büyüklüğü ve kimlik dikkate alınmıştır. Oran öncelikli  $E_{R-MBK}$  algoritmasında ise ilk olarak kalan enerji miktarının derece sayısına bölünmesi ile elde edilen oranın ( $R=Kalan Enerji / Derece$ ) büyüklüğüne daha sonra kimliğe önem verilmiştir.

Algoritmalarımızın performansını ölçmek için TOSSIM benzetim ortamı kullanılmıştır. Performans analizi için TOSSIM benzetim ortamı üzerinde değişen düğüm sayılarına bağlı düğümlerin toplam hareket sayılarına ve MBK büyüklüğüne bakılmıştır. Bu çalışmanın kalan bölümleri şu şekilde düzenlenmiştir. Bölüm 2’de önceki çalışmalar açıklanmıştır. Bölüm 3’de önerilen algoritma  $E_{MBK}$  anlatılmıştır. Bölüm 4’te teorik analiz, Bölüm 5’te performans değerlendirmesi verilmiştir. Bölüm 6’te ise sonuç bölümü yer almıştır.

### 3. Önceki Çalışmalar

Gradinariu ve arkadaşı [8], dağıtık sistemde tekil (İng. Unique) süreç kimlik değerlerine sahip olan ağlar için geliştirdiği düğüm renklendirme (İng. Vertex coloring) algoritması ile MBK’nın elde edilebileceğini göstermiştir. MBK’nın bulunabilmesi için

karşılıklı dışlama (İng. Mutual exclusion) tekniğini [3] ve rasgelelik (İng. Randomization) tekniğini [14] kullanan algoritmalar geliştirilmiştir. Wu ve arkadaşları [16], MBK’yı birim disk çizgeleri üzerinde çalışmıştır. Goddard ve arkadaşları [7], senkron MBK algoritması tasarlamıştır. Guellati ve arkadaşı [9], yaptıkları araştırmada öz kararlı bağımsız küme algoritmalarını incelemişlerdir ve algoritmalar çalıştırıldığı ortam (dağıtık – merkezi), kullanılan topoloji, karmaşıklık ve anonimlik yönünden karşılaştırılmıştır. Hedetniemi ve arkadaşlarının algoritması [10]’da verilmiştir.

Telsiz duyarğa ağlarında kümeleme probleminin çözümü için dağıtık MBK algoritması tasarlamak çok önemlidir [2]. Merkezi sistemde çalışan ilk öz kararlı MBK algoritması Shukla tarafından tasarlanmıştır [13]. 2001 yılında Ikeda,  $n$  düğüm sayısı iken  $n>4$  olan sistemi en fazla  $(n+1)(n+2)/4$  hareket ile öz kararlı hale getiren ilk dağıtık öz kararlı MBK algoritmasını ( $I_{MBK}$ ) tasarlamıştır [11]. 2007 yılında Turau, sistemi en fazla  $(3n-5)$  hareketle öz kararlı hale getiren dağıtık öz kararlı MBK algoritmasını ( $T_{MBK}$ ) tasarlamıştır [15]. Her iki dağıtık algoritma da rasgele topoloji kullanmaktadır [9]. Kablosuz sistemlerde iletişim en fazla enerjiyi harcar ve cihazların enerjisi sınırlıdır. Öz kararlı algoritmaların zaman karmaşıklığı tur sayısı veya hareket sayısı ile ölçülür [15]. Sınırlı enerjiye sahip olan kablosuz sistemlerde hareket sayısını azaltmak tur sayısı kadar önemlidir. Bunun temel nedeni, her düğüm, her hareketten sonra yeni durumunu komşularına bildirmek için mesaj göndermesidir ve sistemdeki hareket sayısını azaltmak ağın yaşam süresini uzatır.

### 4. Önerilen Algoritma

Önerilen algoritmalarımız  $T_{MBK}$  algoritmasının genişletilmiş versiyonudur. Her düğümün yerel durumu ( $s_i$ ), üç değişkenden birisi ile ifade edilir: OUT, IN ve WAIT. Bir düğümün  $s_i=OUT$  değerine

sahip olması MBK'ya dâhil olmadığı;  $s_i=IN$  değerine sahip olması MBK kümesine dahil olduğu;  $s_i=WAIT$  durumunda olması ise MBK kümesine dahil olup olmadığına henüz karar vermemiş olduğu anlamına gelir. Bir dağıtık sistem öz kararlı hale geldiğinde, sistemde sadece IN ve OUT düğümler bulunur.

$N_i$  simgesi  $i$  düğümünün komşularını,  $e_i$  simgesi  $i$  düğümünün enerjisini,  $d_i$  simgesi  $i$  düğümünün derecesini ve  $r_i$  simgesi ise  $i$  düğümünün oranını ifade eder. Aşağıdaki ön tanımlayıcılar  $E_{MBK}$  kurallarını ifade etmek için gereklidir.

$$inNbr \equiv \exists j \in N_i [s_j = IN]$$

$$waitNbrMaxE \equiv \exists j \in N_i [s_j = WAIT \wedge e_j > e_i]$$

$$inNbrMaxE \equiv \exists j \in N_i [s_j = IN \wedge e_j > e_i]$$

$$waitNbrMaxD \equiv \exists j \in N_i [s_j = WAIT \wedge d_j > d_i]$$

$$inNbrMaxD \equiv \exists j \in N_i [s_j = IN \wedge d_j > d_i]$$

$$waitNbrMaxR \equiv \exists j \in N_i [s_j = WAIT \wedge r_j > r_i]$$

$$inNbrMaxR \equiv \exists j \in N_i [s_j = IN \wedge r_j > r_i]$$

Önerilen  $E_{E-MBK}$  algoritması aşağıdaki dört kuralı kullanır.  $E_{D-MBK}$  versiyonunda  $waitNbrMaxE$  ve  $inNbrMaxE$  değerleri sırasıyla  $waitNbrMaxD$  ve  $inNbrMaxD$  değerleri ile değişir.  $E_{R-MBK}$  versiyonunda ise  $waitNbrMaxE$  ve  $inNbrMaxE$  değerleri sırasıyla  $waitNbrMaxR$  ve  $inNbrMaxR$  değerleri ile değişir.

$$K1: s_i = OUT \wedge \neg inNbr \rightarrow s_i := WAIT$$

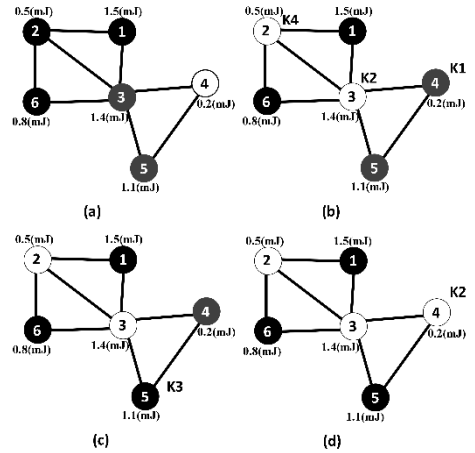
$$K2: s_i = WAIT \wedge inNbr \rightarrow s_i := OUT$$

$$K3: s_i = WAIT \wedge \neg inNbr \wedge \neg waitNbrMaxE \rightarrow s_i := IN$$

$$K4: s_i = IN \wedge inNbrMaxE \rightarrow s_i := OUT$$

Şekil 2'de enerji öncelikli  $E_{E-MBK}$  algoritmasının bir çizge üzerinde çalışma örneği gösterilmiştir. Şekildeki bağlı çizgede 1'den 6'ya kimlik numarasına sahip düğüm vardır. Başlangıçta 1, 2 ve 6 numaralı düğümler IN durumunda, 3 ve 5 numaralı düğümler WAIT durumunda ve 4 numaralı düğüm ise OUT durumundadır. Düğümler komşularından mesaj aldıktan sonra

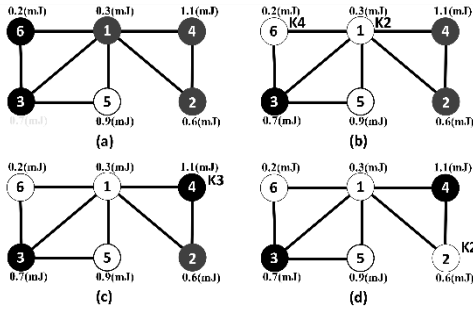
kurallarının ön koşullarını kontrol ederler. Bu işlem sonucunda 2 numaralı düğüm K4'ü çalıştırarak OUT durumuna geçer. Çünkü IN durumundaki komşularından en az bir tanesinin enerjisi daha büyüktür (1 ve 6). 3 numaralı düğüm K2'yi çalıştırarak OUT durumuna geçer. 4 numaralı düğüm ise K1'i çalıştırır ve WAIT durumuna geçer. Birinci turun sonunda sistemin görüntüsü Şekil 2.b'de gösterilmiştir. Bu aşamada sistemde 5 numaralı düğüm öncelikli düğümdür. Bu yüzden 2. turda 5 numaralı düğümün enerjisi 4 numaralı düğümden büyük olduğu için K3'ü çalıştırır ve IN durumuna geçer. Bu turun sonundaki sistem görüntüsü Şekil 2.c'de gösterilmiştir. Son turda ise 4 numaralı düğüm K2'yi çalıştırır ve OUT durumuna geçer. Bu turun sonunda sistem öz kararlı hale gelir ve öz kararlı MBK kümesine sahip sistemin görüntüsü Şekil 2.d'de gösterilmiştir.



Şekil 2:  $E_{E-MBK}$  çalışma örneği

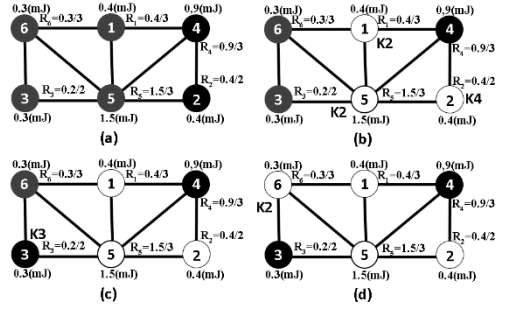
Şekil 3.a'da yer alan çizge üzerinde  $E_{D-MBK}$  algoritması çalıştırılırsa 6 numaralı düğümün derecesi 3 numaralı düğümün derecesinden küçük olduğu için 6 numaralı düğüm K4'ü çalıştırır ve OUT durumuna geçer. 1 numaralı düğümün IN durumunda en az bir komşusu olduğu için K2'yi çalıştırır ve OUT durumuna geçer. Birinci turun sonunda sistemin görüntüsü Şekil 3.b'de yer almaktadır. Bu adımda öncelikli düğüm WAIT durumundaki 4 numaralı düğümdür. 2

ve 4'ün dereceleri aynıdır; fakat dereceleri aynı olan düğümlerde simetriyi kırmak için kullanılan ikinci öncelikli ölçüt düğüm enerjileridir. Bu durumda 4 numaralı düğümün enerjisi 2 numaralı düğümden daha büyük olduğu için bu düğüm K3'ü çalıştırır ve IN durumuna geçer. İkinci turun sonundaki sistemin görüntüsü Şekil 3.c'de gösterilmiştir. Bu aşamada öncelikli düğüm sadece 2'dir. 2 numaralı düğüm K2'yi çalıştırır ve OUT durumuna geçer. Böylece sistem öz kararlı hale gelir ve kararlı sistemin görüntüsü Şekil 3.d'de gösterilmiştir.



Şekil 3:  $E_{D-MBK}$  çalışma örneği

Başlangıç durumu Şekil 4.a'da yer alan çizge üzerinde  $E_{R-MBK}$  algoritması çalıştırılırsa 2 numaralı düğümün oranı 4 numaralı düğümün oranından küçük olduğu için 2 numaralı düğüm K4'ü çalıştırır ve OUT durumuna geçer. 1 ve 5 numaralı düğümler ise K2'yi çalıştırır ve OUT durumuna geçerler. Birinci turun sonundaki sistemin görüntüsü Şekil 4.b'de görülmektedir. Bu aşamada 3 ve 6 numaralı düğümlerin oranları eşittir. Bu durumda simetriyi kırmak için kimlik numarası küçük olan düğümün önceliği vardır. Bu yüzden öncelikli düğüm 3 numaralı düğümdür. Bu düğüm K3'ü çalıştırır ve IN durumuna geçer. İkinci turun sonundaki sistemin görüntüsü Şekil 4.c'de gösterilmiştir. Son turda ise 6 numaralı düğüm K2'yi çalıştırır ve OUT durumuna geçer. Üçüncü turun sonunda sistem öz kararlı hale gelmiştir ve öz kararlı sistemin görüntüsü Şekil 4.d'de gösterilmiştir.



Şekil 4:  $E_{R-MBK}$  çalışma örneği

#### 4. Teorik Analiz

Bu çalışmada tasarlanan algoritma  $E_{MBK}$  başlangıçta kararsız yapıda olan  $n > 4$  düğümlü bir sistemi en kötü durumda  $(3n-5)$  hareket ile öz kararlı hale getirir ve ispat edilmiştir. Bu nedenle algoritmamızın zaman karmaşıklığı  $O(n)$ 'dir. İspat adımları  $T_{MBK}$ 'ya benzer yapılmıştır.

**Ön Teorem 3.2.1:**  $E_{MBK}$ 'da herhangi bir düğümün kuralı aktif olmadığı zaman IN durumundaki düğümlerin oluşturduğu küme MBK'dır.

**İspat:** WAIT durumunda bekleyen bir  $v$  düğümü bulunduğunu varsayalım ve hiçbir kural aktif olmasın. Eğer  $v$  düğümünün IN komşusu varsa K2 aktif olur ve OUT durumuna geçiş yapar. Eğer IN komşusu yoksa ve WAIT durumundaki komşuları içinde en büyük enerjiye ( $E_{D-MBK}$  için dereceye;  $E_{R-MBK}$  için orana) sahipse K3 aktif olur IN durumuna geçiş yapar. Eğer IN komşusu yoksa ve WAIT durumundaki komşuları içinde en büyük enerjiye ( $E_{D-MBK}$  için dereceye;  $E_{R-MBK}$  için orana) sahip değilse WAIT durumundan kurtulmak için K2 veya K3'ün aktif olmasını bekler. K3'ün aktif olmaması için IN komşusu olmaması gereklidir. IN komşusunun olmaması için bu komşuların IN komşularının olması gereklidir. Düğüm  $v$ 'nin komşuları bu durumda K2'yi çalıştıracak ve OUT duruma geçecektir. Düğüm  $v$ 'nin OUT olmayan diğer komşuları WAIT durumundadır. Bu aşamada K3 ya düğüm  $v$  için ya da onun WAIT

durumundaki komşularından biri için aktif olacaktır. Her olasılık için K2 veya K3 aktiftir, teoremdaki varsayımımız ile çelişiyoruz, bir düğümün WAIT durumunda kalması mümkün değildir. K4 aktif olmadığından dolayı IN durumundaki düğümler bağımsızdır. K1 aktif olmadığından dolayı küme genişletilemez.

**Ön Teorem 3.2.2:**  $E_{MBK}$ 'da eğer herhangi bir düğüm K3'ü çalıştırırsa bir daha kural çalıştırmaz. Bu düğümün komşuları en fazla 1 kural çalıştırır ve bu kural, K2'dir.

**İspat:** Düğüm  $v$ 'nin K3'ü çalıştırdığını farz edelim. Bu olay gerçekleşecek ise düğüm  $v$ 'nin komşularının OUT olması veya enerjisi ( $E_{D-MBK}$  için derecesi;  $E_{R-MBK}$  için oranı) küçük olup WAIT durumunda olması gerekmektedir. Bundan dolayı düğüm  $v$ 'nin komşularının K3 veya K4'ü çalıştırması mümkün değildir; çünkü kuralların ön şartları sağlanmamaktadır. Bu yüzden,  $v$ 'nin komşuları aynı tur içinde sadece K1 veya K2'yi çalıştırabilir. Bu turdan sonra  $v$  düğümü IN durumuna geçer ve komşuları OUT veya WAIT durumunda olur.  $V$  düğümünün bu aşamada çalıştırabileceği kural K4'tür; fakat bu durumun gerçekleşmesi için komşularından en az birinin IN durumuna geçmesi gerekir. Fakat  $v$  düğümü IN durumunda olduğu sürece bu olayın gerçekleşmesi imkânsızdır. Bu yüzden  $v$  düğümü dışsal müdahale olmadığı sürece asla başka kural çalıştıramaz. Dahası,  $v$  düğümünün sadece WAIT durumundaki düğümleri bir kural çalıştırabilir. Bu kural, K2'dir. Bu olaydan sonra  $v$  düğümünün komşuları OUT durumuna geçer ve bir daha başka kural çalıştıramaz.

**Ön Teorem 3.2.3:** Adil olmayan takvimleyici kullanan  $E_{MBK}$  algoritması çalışırken her düğüm için aşağıdaki dört durum sıralamaları ve onların son ekleri mümkündür:

WAIT OUT WAIT OUT

WAIT OUT WAIT IN

IN OUT WAIT OUT

IN OUT WAIT IN

IN OUT

**İspat:** (a) Başlangıçta  $v$  düğümünün WAIT durumunda olduğunu varsayalım. Bu durumda  $v$  sadece K2 veya K3'ü çalıştırabilir.  $v$  düğümü K3'ü çalıştırdığı zaman, Ön Teorem 3.2.2'ye göre  $v$  düğümü IN durumuna geçer ve sonsuza kadar bu durumda kalır. Bu durum WAIT IN sıralamasını oluşturur. Diğer durumda  $v$  düğümü K2'yi çalıştırır ve OUT durumuna geçer. Bu aşamadan sonra  $v$  düğümünün IN komşusunun bulunmadığını ve tekrar K1'i çalıştırdığını varsayalım. Şimdi  $v$  düğümü WAIT durumundadır. Eğer  $v$  düğümünün enerjisi ( $E_{D-MBK}$  için derecesi;  $E_{R-MBK}$  için oranı) komşularından daha büyük ise  $v$ 'nin hiçbir komşusu IN durumuna geçemez. Ön Teorem 3.2.2'ye göre düğüm tekrar kural çalıştıramaz. Bu durum WAIT OUT WAIT IN sıralamasını oluşturur. Eğer  $v$ 'nin herhangi bir WAIT komşusunun enerjisi ( $E_{D-MBK}$  için derecesi;  $E_{R-MBK}$  için oranı)  $v$  düğümünden fazla ise K3'ü çalıştırır IN durumuna geçer. Bu durumda  $v$  düğümü sadece bir kural çalıştırır. Bu kural K2'dir ve WAIT OUT WAIT OUT sıralaması gerçekleşir.

(b) Başlangıçta  $v$  düğümünün OUT durumunda olduğunu varsayalım. Bu durumda  $v$  düğümü sadece K1'i çalıştırır ve WAIT durumuna geçer. Bir sonraki adımda IN durumunda komşusu yoksa K3'ü çalıştırır ve başka kural çalıştıramaz. Bu durumda OUT WAIT IN sıralaması gerçekleşir. Eğer komşularından herhangi biri,  $v$  düğümünden önce IN durumuna geçerse  $v$  sadece K2'yi çalıştırır OUT durumuna geçer ve başka kural çalıştıramaz. Bu hareket OUT WAIT OUT sıralamasını oluşturur.

(c) Başlangıçta  $v$  düğümünün IN durumunda olduğunu varsayalım. Bu durumda sadece K4'ü çalıştırabilir ve OUT durumuna geçebilir. Eğer tekrar IN komşusu bulunmazsa  $v$  düğümü tekrar K1'i çalıştırır

WAIT durumuna geçer. Bu aşamadan sonra K2 veya K3'ü çalıştırabilir. Bu durum IN OUT WAIT IN veya IN OUT WAIT OUT sıralamasını oluşturur.

**Teorem 3.2.1:**  $E_{MBK}$  algoritması adil olmayan takvimleyici altında öz kararlıdır ve  $n$  düğüm sayısını göstermek üzere MBK ile en fazla  $3n-5$  harekette kararlı olur ( $n>4$  için).

**İspat:**  $n$  adet düğümün artan kimlik numarası ile bir çizgi üzerinde her düğümün komşularına bağlı olacak şekilde sıralandığını varsayalım. Eğer başlangıçta bütün düğümler IN durumunda ise sistemdeki en büyük enerjiye ( $E_{D-MBK}$  için en büyük dereceye;  $E_{R-MBK}$  için en büyük orana) sahip olan düğüm hariç en fazla  $n-1$  düğüm OUT durumuna geçer. Daha sonra OUT durumundaki  $n-2$  düğüm WAIT durumuna geçer. Bu aşamadan sonra her turda en az bir düğüm K3 veya K2'yi çalıştırarak öz kararlı hale gelir. Sistemin öz kararlı hale gelene kadar yapacağı hareket sayısı en fazla  $3n-5$  ( $n>4$ ) olur.

## 5. Performans Değerlendirmesi

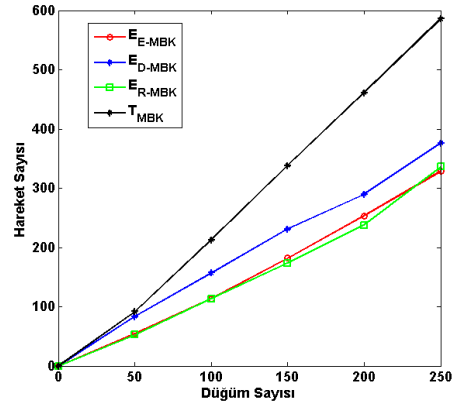
$E_{MBK}$  ve  $T_{MBK}$  algoritmalarının TOSSIM'de bağlı yönsüz birim disk çizgeleri üzerinde çalıştırılarak performans değerlendirilmesi yapıldı. Her ölçüm için 50, 100, 150, 200 ve 250 düğümden oluşan rasgele oluşturulmuş birim disk çizgeleri kullanıldı. Başlangıçta düğümlerin yarısı IN, yarısı ise OUT durumunda olacak şekilde sistem oluşturuldu. Düğümlerin durumları rasgele seçildi. TOSSIM'de çalıştırılan her algoritma için herhangi bir durumdan başlayıp sistem öz kararlı hale gelene kadar geçen sürede düğümlerin toplam hareket sayıları ve MBK büyüklüğü ölçüldü. Benzetim sonuçlarına göre algoritmalar çizelge ve grafikler kullanılarak karşılaştırıldı.

Benzetim sonuçlarına göre Çizelge 1'de ve Şekil 5'deki grafikte rasgele oluşturulmuş bağlı topolojide çalıştırılan algoritmaların değişen düğüm sayılarına bağlı sistemdeki

hareket sayıları gösterilmiştir. Sonuçlara göre en az hareket sayısı ile sistemin öz kararlı hale gelmesini sağlayan algoritma  $E_{R-MBK}$  algoritması olmuştur. Daha sonra sırası ile  $E_{E-MBK}$ ,  $E_{D-MBK}$  ve  $T_{MBK}$  algoritmaları gelmektedir.  $E_{MBK}$  algoritmasının her üç versiyonu da sadece kimlik numarasına göre karar veren  $T_{MBK}$  algoritmasından daha iyi performans göstermiştir.

Çizelge 1: Değişen düğüm sayısına bağlı hareket sayısı çizelgesi

Düğüm Sayısı	Hareket Sayısı			
	$T_{MBK}$	$E_{E-MBK}$	$E_{D-MBK}$	$E_{R-MBK}$
50	91,60	54,40	83,50	52,50
100	213,00	113,10	157,00	113,70
150	338,00	182,20	230,83	173,60
200	461,00	253,00	289,50	237,40
250	586,00	328,40	376,20	336,10

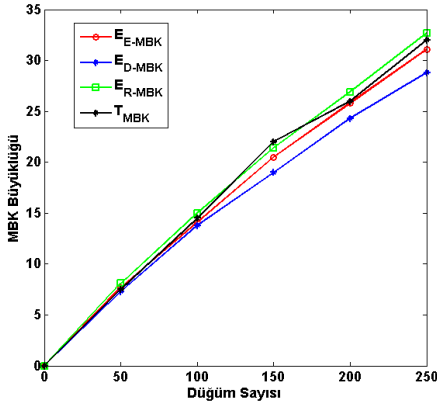


Şekil 5: Değişen düğüm sayısına bağlı hareket sayısı grafiği

Benzetim sonuçlarına göre Çizelge 2'de ve Şekil 6'daki grafikte rasgele oluşturulmuş bağlı topolojide çalıştırılan algoritmaların değişen düğüm sayılarına bağlı sistemdeki MBK büyüklüğü gösterilmiştir. Sonuçlara göre en az MBK büyüklüğü ile sistemin öz kararlı hale gelmesini sağlayan algoritma  $E_{D-MBK}$  algoritması olmuştur. Daha sonra sırası ile  $E_{E-MBK}$ ,  $T_{MBK}$  ve  $E_{R-MBK}$  algoritmaları gelmektedir.

**Çizelge 2:** Değişen düğüm sayısına bağlı MBK büyüklüğü çizelgesi

Düğüm Sayısı	MBK Büyüklüğü			
	$T_{MBK}$	$E_{E-MBK}$	$E_{D-MBK}$	$E_{R-MBK}$
50	7,50	7,7	7,3	8,10
100	14,50	14,1	13,8	15,00
150	22,00	20,5	19	21,40
200	26,00	25,8	24,3	26,90
250	32,00	31,1	28,8	32,70



**Şekil 6:** Değişen düğüm sayısına bağlı MBK büyüklüğü grafiği

## 6. Sonuç

Bu çalışmada dağıtık öz kararlı MBK algoritmaları tasarımı üzerine çalışılmıştır. Turau tarafından tasarlanan  $T_{MBK}$  algoritması geliştirilmiş ve düğümleri sadece kimlik numaralarına göre değil, aynı zamanda derecelerine ve enerjilerine göre MBK'ya dahil edilmesi sağlanmıştır.

Tasarladığımız  $E_{MBK}$  algoritması teorik analizde en kötü durumda  $T_{MBK}$  algoritması ile eşit  $O(n)$  zaman karmaşıklığına sahiptir.  $E_{MBK}$ , teorik analizde  $T_{MBK}$  ile aynı sonuçlara sahip olsa da performans analizindeki benzetim sonuçları  $E_{MBK}$  algoritmasının üç versiyonunun da genel olarak değişen düğüm sayısına bağlı hareket sayısı ve MBK büyüklüğü bakımından  $T_{MBK}$  daha etkin çalıştığını göstermiştir. Ayrıca, MBK'ya dâhil olan düğümler küme lideri olduğundan

diğer düğümlere göre daha fazla enerji harcarlar. Performans değerlendirmesi hareket sayısı bakımından oran öncelikli  $E_{R-MBK}$ 'nin en etkin çalışan algoritma olduğunu gösterdi. MBK büyüklüğü bakımından en etkin algoritma ise  $E_{D-MBK}$ 'dir.

## Teşekkür

Yazarlar TÜBİTAK ARDEB'e 215E115 numaralı desteğinden dolayı teşekkür eder.

## Kaynaklar

- [1] Abbasi, A.A. and Younis, M., "A survey on clustering algorithms for wireless sensor networks", **Computer Communications**, 30:2826-2841 (2007).
- [2] Alzoubi, O. F. K. And Wan, P.J., "Maximal independent set, weakly-connected dominating set, and induced spanners in wireless ad hoc networks", **International Journal of Foundations of Computer Science**, 14:2 287-303 (2003).
- [3] Beaquier, J., Datta, A.K., Gradinariu, M. and Magniette, F., "Self-stabilizing local mutual exclusion and daemon refinement", **Chicago Journal of Theoretical Computer Science**, (2002).
- [4] Dijkstra EW., "Self-stabilizing systems in spite of distributed control", **Commun ACM**, 17(11):643-644 (1974).
- [5] Dolev S., "Self-stabilization", **MIT Press**, Cambridge, MA,USA, (2000).
- [6] Erciyes, K., 2013, Distributed Graph Algorithms for Computer Networks, **Springer-Verlag**, London, 324 p.
- [7] Goddard, W., Hedetniemi S.T., Jacobs D.P. and Srimani P.K., "Self-stabilizing protocols for maximal matching and maximal independent sets for ad hoc networks", **In Proceedings of the 5th**



- International Parallel and Distributed Processing Symposium (IPDPS)**, (2003)
- [8] Gradinariu, M. and Tixeuil, S., “Self-stabilizing vertex coloring of arbitrary graphs”, **In Proceedings of the 2nd Workshop on Self-Stabilizing Systems (WSS99)**, 48-53 (2000).
- [9] Guellati N. and Kheddouci H. “A survey on self-stabilizing algorithms for independence, domination, coloring, and matching in graphs”, **J. Parallel Distrib. Comput**, 70(4):406-415 (2010).
- [10] Hedetniemi, S.M., Hedetniemi, S.T., Jacobs, D.P. and Srimani, P.K., 2003, Self-stabilizing algorithms for minimal dominating sets and maximal independent sets, *Computer Mathematics and Applications*, 46(5-6):805-811 pp.
- [11] Ikeda M, Kamei S. and Kakugawa H., “A space-optimal self-stabilizing algorithm for the maximal independent set problem”, **In Proceedings of the 3rd International Conference on Parallel and Distributed Computing**, 70-74 (2002).
- [12] Li Y., Thai T. and Wu,W., “Wireless sensor networks and applications”, **Springer Science & Business Media**, New York, 464 (2008).
- [13] Shukla, S.K., Rosenkrantz, D.J. and Ravi, S.S., “Observations on self-stabilizing graph algorithms for anonymous networks”, **In Proceedings of the 2nd Workshop on Self-Stabilizing Systems**, (1995).
- [14] Turau, V. and Weyer, C., “Randomized self-stabilizing algorithms for wireless sensor networks”, **In Proceedings of the International Workshop on Self Stabilizing Systems**, 74-89 (2006).
- [15] Turau V., “Linear self-stabilizing algorithms for the independent and dominating set problems using an unfair distributed scheduler”, **Inf. Process. Lett.**, 103(3):88-93 (2007).
- [16] Wu, Y., Du, H., Jia, X., Li, Y. and Huang, S.C.-H., “Minimum connected dominating sets and maximal independent sets in unit disk graphs”, (2005).