

# VANET Ağlarda Kümeleme Teknikleri

**Emre Kağan Akkaya**  
**Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı**  
**Hacettepe Üniversitesi**  
**emrekaganakkaya@gmail.com**

**Öz - Kablosuz ağlarda son yıllardaki ilerlemeler araçsal Ad-Hoc ağların (VANET) ortaya çıkmasına ve gittikçe hız kazanarak güvenli, etkin ve güvenilir iletişim ağlarından biri olma yolunda ilerlemesine neden olmuştur. Sürücü güvenliği, yakıt tasarrufu, acil durum uyarı sistemleri, trafik sıkışıklığına karşı tedbirler gibi birçok fayda sağlayabilecek VANET ağlar yüksek mobilite içeren ve sürekli değişen yapıda olduğu için iletişim yükünün dağıtılması ve stabil, ölçeklenebilir hale gelmesi büyük önem taşır. Bu kapsamda önerilen çeşitli kümeleme yöntemleri bu çalışmada incelenmiş ve nitelikleri, zayıf ve güçlü yanları anlatılmaya çalışılmıştır.**

## **I. Giriş**

VANET ağlar; birbirleriyle kablosuz yada kablosuz olarak haberleşen birden fazla araç/mobil düğümden oluşan dinamik ağlardır. Araçlar ve bazı durumlarda kullanılan yol kenarı birimler (RSU) arası iletişimle toplanan veriler, sürücülere daha güvenli, hızlı veya yakıt tasarruflu bir ulaşım sağlamak için kullanılabilir. Dolayısıyla son yıllarda bu alandaki çalışmaların hız kazandığı ve belirli bir standarda oturtulmaya çalışıldığı gözlemlenebilir.

VANET ağdaki araçların yüksek hızda seyredebilmeleri, topolojinin sürekli değişim içinde olması, düğüm yoğunluğunun değişken olması gibi nedenlerden dolayı ağda meydana gelen yükün önüne geçmek, daha iyi yönetilmesini sağlamak gibi amaçlarla ağda kümeleme yöntemlerine ihtiyaç duyulmuştur. Kümelemede ağdaki araçlar olabildiğince büyük ve az sayıda sanal gruplara ayrılarak, her bir araç, lider düğüm (CH), üye düğüm (CM) yada

geçit düğüm (gateway) olarak üç ayrı rolden birine bürünür. Lider düğümün görevi, üstlendiği kümedeki diğer düğümler arası koordinatör görevi üstlenerek, küme için mesaj iletiminin yönetilmesi, veri paylaşımı gibi görevlerden sorumludur. Geçit düğüm ise (ki bu bazı algoritmalarda lider düğüm ile aynıdır) diğer kümelerle iletişimden yani kümeler arası haberleşmeden sorumludur. Üye düğüm ise herhangi bir özel rol üstlenmemiş, kümedeki sıradan düğümlere karşılık gelir. Kümeleme yönlendirme (routing), güvenlik, hizmet kalitesi (QoS) gibi farklı amaçlara hizmet eder ve hedeflenen uygulama alanına göre farklı özellikler gösterebilir.

Bu incelemede de daha önceki inceleme makalelerinden yola çıkılarak [1-3] bir kategorizasyon oluşturulmaya çalışılmış ve incelenen makaleler güçlü ve zayıf yanlarıyla ortaya konulmuştur. Yazının devamında sırasıyla; II. kısımda kümelemenin genel özellikleri, III. kısımda incelenen algoritmalar, IV. kısımda kullanılan kümeleme metrikleri, V. kısımdaysa sonuç ve incelemenin devamında yapılacaklar anlatılmıştır.

## **II. Genel Özellikler ve Nitelikler**

Kümeleme algoritmalarının hedefi ağdaki küme sayısının azaltılması, kümelerin olabildiğince büyük ölçekli ve uzun ömürlü olmalarının sağlanmasıdır ve bu sayede mesajlaşma yükünün azaltılması, iletişim maliyetlerinin düşürülmesi olarak nitelendirilebilir.

İncelenen kümeleme tekniklerinde de görülmüştür ki, komşuluk keşfi, lider düğüm seçimi, üyelik, ve küme bakımı gibi belli başlı adımlar tüm algoritmalarda ortaktır.

Bunlardan komşuluk keşfi; yeni bir ağa dahil olan bir düğümün kendini tanıtmak için bilgilerini içeren mesaj yayımlaması ve k-hop komşuluğundaki diğer düğümleri keşfetmek için de ya gelen mesajları pasif bir şekilde dinlemesi yada etkin bir şekilde komşulardan bilgi talep etmesi sürecidir. Buradaki söz konusu bilgi alışverişi çoğunlukla düğümün konum ve hız bilgisini içeren bir mesajdır.

Lider düğüm seçimi; komşuları hakkında bilgi toplamış bir düğümün bir kümeye dahil olmak (yada yeni küme oluşturmak amacıyla) belirli kıstaslara göre komşularının lider düğüm (CH) uygunluğunu kontrol eder. Eğer komşuluk tablosunda böyle bir düğüm varsa, ona katılmak için mesaj gönderir yada böyle bir düğüm yoksa ve küme arayan düğümün kendisi liderliğe uygunsa kendisini lider düğüm (CH) ilan eder.

Üyelik adımıdaysa, küme arayan düğüm, lider olarak belirlediği düğüme katılmak için mesaj gönderir. Burada bazı algoritmalar, lider düğümün zaten var olması gerektiği koşulunu koyarken, diğerlerinde normal bir düğüm de bu süreç ile birlikte lider haline gelebilir. Devamındaysa (eğer varsa maks. küme elemanı sayısı vb. gibi koşullar da göz önünde bulundurularak) lider düğüm, küme arayan düğümü kabul eder. Bazı yaklaşımlarda düğümün üyeliğinin kabulü yada reddine dair cevap mesaj liderden küme arayan düğüme gönderilerek bir el sıkışma (handshaking) protokolü de uygulanır.

Küme bakımındaysa, lider düğüm periyodik olarak üye düğümlerin durumunu sorgular. Üyeye ulaşamama durumunda söz konusu düğüm üyelik tablosundan çıkarılır. Bazı algoritmalar da buna ek olarak üye düğümlerin lider düğüm olma uygunluğu da periyodik olarak kontrol edilerek, gerektiğinde lider düğüm rolünü devredebilir. Bunun dışında lider düğüm, kendinden daha büyük (veya algoritmaya göre liderliğe daha uygun) bir başka küme lideriyle karşılaştığında, kendi rolünden vazgeçerek kümeler birleştirilebilir. Üye düğümler içinse, bakım süreci, lider ile olan bağın ayakta olduğunun mesaj göndererek (yada poll) kontrolüyle sağlanır.

**Tablo I: CH seçimi için ağırlıklı metrik kullanan algoritmalar**

Algoritma	Metrik	Komşu Keşfi	El Sıkışması
AMACAD	Uzaklık, görelî hız, hedef konum	Hello	Var
APROVE	Uzaklık, görelî hız	Hello	Var
DMCNF	Gecikme oranı, takip eden araç sayısı	Hello	Var
GDMAC	ID, düğüm derecesi	Hello	Yok
PPC	ID, seyahat süresi, ort. görelî hız	Hello	Yok
Sp-CI	Uzaklık, görelî hız	Hello	Var
TBC	Uzaklık, görelî hız	Inquiry	Yok
TC-MAC	Uzaklık, görelî hız, dönüş yönü	Hello	Yok

### III. Kümeleme Algoritmaları

AMACAD [4]; GPS/NS sistemiyle donatılmış araçlardan oluşan ve trafik sıkışıklığı, kaza vb. durumlarda araçların alternatif yollara yönlendirilmesi gibi sorumluluklara sahip bölgesel baz istasyonlarını da içeren varsayımsal bir sistemde, araçların birer sokak/blok vb. alanları temsil eden *bölgelere* girmesiyle; görelî hedef bilgisini güncellediği bir senaryo öne sürer. Buna göre Hello mesajlarıyla

birbiri hakkında bilgi toplayan düğümler; son hedef bilgisi, görelî hedef bilgisi, sürat ve konumu içeren bir metrik kümesi kullanarak birbirlerine ait ağırlıklı fonksiyon değerlerini elde eder, tüm araçlar hesaplandıktan sonra öncelik sırasına göre her düğüm görelî en düşük değeri düğümü lider (CH) seçer. Bu sayede benzer rotaları hedefleyen araçların bir küme içerisinde yer alınması hedeflenmektedir. Küme bakımı ve kümeye katılım sırasında işletilen el sıkışma (handshaking) protokolü sırasında, lider düğüm (CH) gelen üye mesajına göre, eğer üye daha iyi bir lider adayıysa, liderliği ona devredebilir veya kümenin üye sınırı vb. gibi kısıtlara takıldıysa üyeliği reddedebilir.

APROVE [5]; veri madenciliğinde yaygın olarak kullanılan ve K-means gibi diğer kümeleme tekniklerinden daha kısa sürede çalışan ve daha az hata üreten Affinity Propagation (AP) algoritmasını, lider düğüm (CH) seçimi için uyarlamıştır. Bu bağlamda, birbirleriyle mesajlaşan düğümler, *sorumluluk ve uygunluk* fonksiyonlarına (bu değerler periyodik olarak değiş tokuş edilir) göre en yüksek değere sahip lider düğümü belirlemeye çalışırlar. Söz konusu fonksiyonlar düğümler arası o anki Öklid mesafesinden ve belli bir süre sonraki Öklid mesafesinden hesaplanmaktadır. Algoritmaya göre periyodik olarak gönderilen bu mesajlar saklandığı ve hesaplamada kullanılmaya devam edildiği için, daha önceden lider düğüm (CH) seçilen bir düğümün yeniden lider seçilmesi daha olasıdır ve böylelikle yeniden kümelenme (re-clustering) işlemlerinin azaltılması öngörülmüştür. Algoritmanın her düğümde asenkron olarak çalıştırılabilir olması, değiş tokuş edilen RESP ve AVAIL mesajlarının en azından bir iterasyon eski olması anlamına da gelir, dolayısıyla mesaj kaybının sıklıkla yaşanabileceği bir senaryoda eldeki eski bilgiler doğrultusunda hareket edilecek ve algoritmanın başarımı düşecektir.

**Tablo II: CH seçimi için öncelik sırası kullanan algoritmalar**

Algoritma	Metrik	Komşu Keşfi	El Sıkışma
RMAC	Uzaklık, hız, büyüklük	Inquiry	Var
VMaSC	Görelî hız	Hello	Var

RMAC [6] algoritmasında, bir veya daha fazla CM'den oluşan bir kümedeki herhangi bir düğüm CH, CM yada her ikisinin birden rolünü üstlenebilmektedir. Buna göre bir kümenin lideri (CH) konumunda olan bir düğüm başka bir kümenin de üyesi olabilir (CM). Dolayısıyla bir düğüm birden fazla kümede yer alabilmekte, kümeler üst üste binebilmekte ve bu sayede iletişimdeki kopmaların önüne geçilmesi hedeflenmektedir. Kümeler için bağlantılar lider düğümün (CH) periyodik olarak üye düğümleri sorgulamasıyla (polling) sağlanmaktadır. Kümedeki her düğüm komşularına ait bilgileri (araç ID, konum, sürat, yön, eğer CH ise bağlı olduğu küme büyüklüğü, bağlı olduğu küme sayısı ve zaman damgası) bir tabloda tutmakta, ancak komşular sadece kümedeki 1-hop uzaklıktaki düğümlerden değil, aynı zamanda maks. 2-hop (ZOI) uzaklığındaki diğer düğümleri de kapsamaktadır. Bu çözümle kümeden ayrılan bir düğümün, yeni bir küme aramasının kolaylaştırılması hedeflenmiştir. Dolayısıyla kümeye bağlı olmayan bir düğüm öncelikle tablosunu kontrol edecek, eğer burada 1-hop uzaklıkta komşu düğüm bulamıyorsa ağda sorgulama paketleri (INQ) yayınlayarak komşuları hakkında bilgi toplamaya çalışacaktır. RMAC maks. küme sayısını sınırlamak ve bir düğümün katılabileceği küme sayısını sınırlamak adına, max-cluster-size ve max-cluster-membership kısıtlarını kullanmaktadır. Bu kısıtları geçmemiş komşular arasından yakınlık, küme büyüklüğü ve en uzun

süre CH olarak kalma hesaplamalarına göre en öncelikli olacak şekilde CH seçimi, katılma isteği (JOIN\_REQ) paketi gönderilerek yapılır. RMAC bu adımda öncesindeki birçok algoritmadan farklı olarak el sıkışma protokolü kullanmaktadır, bu bağlamda lider düğümden (CH) üye isteğinde bulunan düğüme kabul yada ret durumunda cevap paketi gönderilmektedir. Sürat, konum ve yön bilgilerinden oluşan bir grup metrik kullanılmaktadır. Bu bilgilerin araçta yer alan GPS biriminden sağlandığı varsayılmaktadır

DMCNF [7] çözümünde; daha önce bahsi geçen 1-hop uzaklığındaki düğümlerden kümeleme oluşturan çoğu yaklaşımın aksine, n-hop uzaklıktaki komşular arasından küme oluşturulması ve bu sayede daha fazla aracı içeren daha büyük ölçekli ve daha az sayıda küme yaratılması öngörülür. İkinci dikkat çekilen noktaysa, diğer algoritmaların çoğunun GPS sistemlerine dayandığı ve bunun doğruluk oranının her zaman iyi olamayacağı ve GPS'in erişilemediği durumlarda da VANET ağların stabil olarak işlerliğinin devam ettirilebilmesi. Bu amaçla, komşuluk takibi stratejisi adı verilen yöntemle, her bir düğüm 1-hop uzaklıktaki başka bir düğümü takip eder. Eğer takip edilen düğüm, lider (CH) ise doğrudan ilişki, değilse dolaylı ilişki olarak kümeye dahil olunur. Bu yöntemle, dinamik olarak eğer bir düğüm takip ettiği lider düğümden daha fazla sayıda takipçiye sahipse ve bu takipçilere göreli ortalama hızı daha düşükse, kendisi lider düğüm (CH) konumunu üstlenir. Söz konusu n-hop kümeleme tekniğinin genel olarak dezavantajı ise, farklı kümeler arası araçların haberleşebilmesi için daha karmaşık routing protokollerinin geliştirilmesi zorunluluğu olarak gösterilebilir.

TBC [8] algoritması, özellikle araçların hızlı seyahat ettiği otoyolların kapsam olarak seçildiği, kümeleme sürecinin komşuları arasında en az hıza sahip düğüm tarafından başlatıldığı bir yöntem önerir. Komşu araçlar sürat ve yön bilgilerine göre stabil yada stabil olmayan olarak ikiye ayrılır ve sadece stabil olanlarla küme kurulabilir. Bununla birlikte, r uzaklığına göre araçlar r-stabil yada 2r-stabil olarak nitelendirilir. Bu bağlamda, 2r-stabil komşular en yavaş düğümden gelen mesaja

cevap vererek geçici olarak o düğümü lider (CH) seçerler ve kendi uygunluk değerini hesaplarlar ki bu değer stabil komşuların ortalama pozisyon ve hız değerlerinin düğümün kendi değerlerine karşılaştırılması sonucu bulunur. Aynı zamanda değer sonucu bekleme süresi elde edilir ve bu bekleme süresi sonunda bir başka düğüm CH olmadıysa kendilerini lider düğüm ilan ederler, eğer başka düğüm olduysa ona üye düğüm olarak katılırlar. Birbirinin kapsamına giren iki küme içinse, eğer hız deviasyonu belirli bir eşiğin altındaysa birleşme işlemi gerçekleşir ve daha az sayıda üyesi olan lider düğüm rolünden vazgeçer ve üye halini alır. Dağılan kümedeki diğer üyeler ise kapsamdalarsa yeni kümeye katılırlar yada değilse en uzun kümede kalma süresine göre diğer kümeleri değerlendirirler.

**Tablo III: CH seçiminde zamanlayıcı kullanan algoritmalar**

Algoritma	Metrik	Komşu Keşfi	Eİ Sıkışma
DBA-MAC	Bağlantı sona erme süresi	Hello	Var

DBA-MAC [9] algoritması ise otoyolda yüksek hızda seyreden araçların, bir tehlike (kaza, yol durumu vb.) anında arkasındaki araçları uyarması için kullanılabilecek bir yöntem önermektedir. Buna göre, klasik kümelerde uyarı mesajı yayınlanarak (broadcast) gönderilirken, kümenin risk alanındaki (risk zone) tüm araçları uyaramayacak kadar dar kapsamlı olmasından yola çıkılarak *omurga* (backbone) adını verdikleri bir model ile tek sıra halinde önden arkaya mesaj dağıtan bir yapı sunulmaktadır. Bunun avantajı olarak ise, unicast olarak gönderilen mesajların ACK cevabıyla iletiminin kontrol edilebilmesi ve multi-hop mesaj dağıtırken (tüm düğümlerin aldıkları mesajı tekrar yayınlamalarına nazaran) daha etkin bir şekilde çalışabilmesi gösterilmektedir. Yaklaşım araçları, omurga üyesi (BM) ve normal araç (NV)

olarak ikiye ayırmakta ve bir araç kendisini omurga üyesi olarak tanımlamadan önce belli bir süre beklemesini öngörür. Kendini omurga üyesi ilan eden düğüm diğer düğümlere kendi metriklerinden oluşan mesajları yayınlar ve kendi bekleme süreleri bitmeden bunu duyan başka düğüm varsa ilk mesajı yayınlayan düğümün ne kadar süreyle kapsamında olacaklarını hesaplar. Eğer hesaplanan süre belli bir eşiği geçiyorsa, düğüm sonunda CANDIDATURE mesajı göndereceği bir sürece başlar ve bu mesajı ilk gönderen düğüm, omurga üyesi (BM) için bir sonraki omurga üye düğümü halini alır. Benzer şekilde süreç tekrarlanarak omurga oluşturulur.

GDMAC [10], daha önceden MANET ağlar için geliştirilmiş DMAC [11] algoritmasının VANET mobilitesine yönelik geliştirilmiş halidir. DMAC algoritmasında, her bir düğüm (kalan enerji, hız vb. Metrikler üzerinden) kendi ağırlığını hesaplar (ağırlık değeri zaman içinde değişeceğinden periyodik olarak hesaplanır ve komşularla sürekli bilgi alışverişi yapılır) ve komşularıyla ağırlık karşılaştırması yapar. Eğer en 'ağır' düğüm kendisiyse, kendini lider düğüm (CH) ilan eder. Ağırlığı daha fazla bir lider düğüm ile karşılaştığında kendini fes eder ve ona katılır. Bu şekilde basit ve katı kurallar içeren DMAC algoritmasında, kümelerde sıklıkla meydana gelen değişiklikler yüzünden topoloji sürekli değişir ve düğümler arası mesaj yükü oldukça artar.

GDMAC algoritmasıysa, bu olumsuzlukların önüne geçmek adına, sıkı kuralların esnetildiği ve daha ağır bir lider düğümle karşılaştığında dahil üye düğümlerin aynı lidere bağlı kalmasını sağlayacak bazı tanımlar getirir. Bu tanımlardan ilki  $H$  parametresi; bir küme üyesinin (CM) yeni bir lider düğümün kapsamına girmesi halinde ancak ve ancak yeni lider düğümün ağırlığı varolan lider düğümün ağırlığından en az  $H$  kadar daha iyiye üye düğümün yeni lidere geçmesini öngörür. Bu sayede  $H$  değeri ne kadar büyükse, üye düğümlerin küme değiştirmelerinin o kadar önüne geçilecektir. Diğer parametresyse,  $K$ , lider düğüm yoğunluğunu belirler. Buna göre  $K=0$  olduğunda iki lider düğüm (CH) yan yana bulunamaz ve bir

rolünü bırakarak diğerine katılırken,  $K>0$  olduğu durumlarda birden fazla lider düğüm yan yana bulunarak gereksiz yeniden kümeleme işlemlerinden kaçınılır.

PPC [12] algoritması ise <düğüm ID, CH ID, konum, lidere giderken ara düğümün ID'si, öncelik> değerlerinden oluşan bilgi mesajlarının periyodik olarak düğümlerden yayınlanmasıyla ve önceden belirlenmiş  $L$  küme çapı içerisinde yer alan düğümlerin varsa varolan bir lidere katılması yoksa yeni bir küme oluşturmasını öngörür. Yan yana gelen iki lider düğüm arası mesafe önceden belirlenmiş  $D$  eşiğinden küçükse, daha az sayıda olan lider rolünü bırakarak diğerine katılır. Mesaj alışverişi sırasında elde edilen bilgiler ışığında her bir düğüm kendi *öncelik* (priority) değerini hesaplar ve eğer bir düğüm 1-hop uzaklığındaki komşuları arasında en öncelikliyse ve bu komşulardan en az birinin 1-hop komşuları arasında da en yüksek önceliğe sahipse lider düğüm halini alır. Söz konusu öncelik değeri hesaplanırsa, maksimal seyahat süresi ve diğerlerine nazaran sürat deviasyonu göz önüne alınır. Bu sayede daha uzun süre yol alacak araçlara ve hız değişimi az olan araçlara daha yüksek öncelik verilerek daha stabil kümeler hedeflenir.

Algoritma yaklaşımı diğerlerinden daha ilginç olarak nitelendirilebilecek yöntemlerden biriye Sp-CI [13]; Coulomb yasasına dayanır. Buna göre Coulomb sabiti birbirine doğru yada birbirleriyle paralel hareket eden araçlar için pozitif, birbirinden uzaklaşan araçlar içinse negatif olacak şekilde hesaplanır. Elektromanyetikte kullanılan söz konusu formüldeki yükler ise her bir aracın özelliğine göre (örneğin çoğu zaman sabit yol ve sabit hız izleyeceği varsayılan otobüs, kamyon gibi araçların yükü daha fazladır yada yol ayrımlarında dönüşü olmayan şeritlerde hareket eden araçların ağırlığı daha fazladır vb.) belirlenir. Bu sayede fiziksel bir ortamdaki yüklü noktalar olarak varsayılan araçların birbirlerine uzaklıklarının karesinin tersi kadar birbirlerine kuvvet uygulayacakları ve komşularına göreli en yüksek kuvvete sahip düğümün lider rolüne bürüneceği açıklanır. Kullanılan bu metriklerle, birbirlerinden uzaklaşan araçların

kümelenmesinin önüne geçilerek, birlikte hareket eden araçlardan küme oluşturma eğilimi görülmektedir.

Bir diğer kümeleme algoritması TC-MAC [14] ise küme içi iletişimin lider düğüm tarafından yönetildiği bir TDMA yaklaşımı ortaya koyar. Araştırılan makalede CH seçimi için kullanılan algoritmanın detaylarına girilmemiş, daha önceki makale referans [15] verilerek, aracın bulunduğu şeride göre CH seçimi yapılabileceği varsayılmıştır. Küme içi güvenlik mesajı olmayan mesajların dağıtımı için lider düğüm kendi (bkz. TDMA) slotunu kullanarak normal bir düğüm gibi davranırken, güvenlik gerekçeli mesajlarda diğer verilerin iletişimi kesilerek tüm uygun slotlardan mesaj yayınlanır. Küme içi bire-bir düğüm iletişimindeyse CH kullanılmadan, iletişimi başlatan düğüm kendi slotu üzerinden hedef düğümle haberleşip, el sıkışma cevabı sonrası herhangi birinin slotu kullanılarak iletişim sağlanır. Küme içi bir düğümün birden fazla düğümüne mesaj yayınlaması istendiğindeyse, hedef düğüm sayısı az ise az önceki yöntem, değilse hedef düğümleri de içeren bir vektörün lider düğümüne iletilip iletişim talep edilmesiyle haberleşme sağlanır.

VMaSC [16] ise <yön, mevcut sürat, hop sayısı, ortalama göreceli sürat, bağlı olunan lider düğüm (CH) ID değeri> gibi metriklerin başlangıçta ve devamında periyodik olarak düğümler arasında paylaşıldığı, lider seçiminde ise düğümün kendine en yakın göreceli hıza sahip lider düğümüne katılma isteği gönderdiği ve el sıkışma mesajıyla birlikte kümeye dahil olduğu birden fazla hop için uygulanabilecek bir algoritma önerir. Eğer kümeleme 1-hopluk ise, düğüm yeni lider düğümlerin oluşumunu engellemek için, varolan bir kümedeki bir üyeye iletişime girip, *misafir üye* (guest node) rolünü üstlenebilir. Yeni küme oluşumunda, bir düğüm komşularına göreceli en az mobiliteye sahipse kendisini lider düğüm ilan eder. Bakım sürecinde ise üye düğümler de lider düğümde önceden belirli zaman dilimleriyle bir diğerinden paket bekler, paketin eşik sürede gelmediği durumda iletişimin kaybedildiği anlamına gelir ve lider, üyeyi küme listesinden çıkarırken, benzer durumda üye de yeni bir lider ve küme arayışı

içine girer. Algoritmada kullanılan maksimum hop değeri, hem birden fazla hop ile oluşturulan kümelerde mesajın ara düğümlerce ne kadar uzağa yeniden yayınlanacağını belirler hem de yılan benzeri tek sıra halinde kümelerin oluşmasını önler. Genel olarak birden fazla hopu destekleyen kümeleme algoritmalarında görüldüğü üzere yoğun düğüme sahip senaryolarda gönderilen mesaj yükü fazlasıyla artar ve bu da güvenilirliği veya bilgi dağıtım hızını etkileyebilir.

#### IV. Kümeleme Metrikleri

MANET ağlarda genellikle odak noktası olan enerji tasarrufu ve hesaplama gücü söz konusu alandaki algoritmaların değerlendirilmesindeki en önemli metrikleri oluştursa da, senaryo VANET ağlarda tamamen farklıdır. Öyle ki, kümelerin stabil ve uzun soluklu olması, bununla birlikte bakım maliyetinin ve bant genişliği kullanımının en azda tutulması önem kazanmıştır.

Bu kapsamda, küme stabilitesi genellikle lider düğüm (CH) ömrü, üye düğüm (CM) ömrü, lider düğüm (CH) değişim sayısı ve küme sayısı gibi metrikler üzerinden ölçülür.

Lider düğüm (CH) ömrü, bir düğümün lider seçilmesinden bu rolü bıraktığı ana kadar geçen süre, benzer şekilde, üye düğüm (CM) ömrü bir düğümün bir düğümüne katılmasından ayrıldığı ana kadar geçen süreye işaret eder. Bu sürelerin ne kadar uzun olduğu, kümenin ne kadar stabil olduğunu ve yeni küme oluşturma maliyetinin düşürüldüğünün göstergesidir.

Lider düğüm (CH) değişim sayısı ise, lider düğüm rolünden diğer rollere geçen düğüm sayısını ifade etmekte, yüksek değişim sayısı bakım maliyetinin yüksek olduğunu göstermektedir.

Küme sayısı ise VANET ağda ne kadar etkili bir routing yaklaşımının yer edinebileceğini işaret eder ve az sayıdaki ve büyük ölçekli kümelerin daha faydalı olduğu anlamına gelir.

#### V. Sonuç ve Gelecek Çalışma

Daha önceki incelemeler [1-3] de dikkate alınarak, VANET ağlarda kümeleme teknikleri zayıf ve güçlü yanlarıyla olabildiğince

incelenmeye çalışılmış ve farklı tür yaklaşımlarla karşılaşıldıkça basit bir kategorizasyon sunulmuştur. Bu inceleme çalışmasının devamında, söz konusu kategorizasyona daha fazla makale incelenerek eklemeler yapılacaktır. Bu bağlamda kapsam olarak özellikle şu ana kadar incelenmemiş iki konu; (1) güvenlik (2) sürücü davranışı gibi nedenlerle araçlardaki yön ve hızdaki değişimlerin tahmin edilebilmesi yönünde yapılan çalışmalar da incelenecektir.

Şu ana dek incelenen makalelerde, ilk araştırmaların MANET ağlardaki kümeleme tekniklerinin VANET ağlara uyarlama çalışması olduğu ve araçların yüksek mobilitesi ve sürekli değişen ağ topolojisini gözetemedikleri için başarısız oldukları söylenebilir. Özellikle son yıllarda VANET ağlara yönelik çözümlerin MANET ağlardan ayrılmaya başladığı ve olması gerektiği gibi yüksek mobilite kısıtlarını gözetir yaklaşımların sunulduğu görülmüştür. Bununla birlikte çoğu yaklaşımın dayandığı varsayımların gerçek dışı olduğu ve bu nedenle şu haliyle simülasyonlardan gerçek hayata uyarlanamaz olduğu da tartışılabilir.

Örneğin birçoğunun dayandığı GPS verisi isabetsiz çalışabilir veya aracın o anda GPS'e hiç erişimi olmayabilir, küme oluşturma yada bakımı için gönderilen paketler - çakışma, darboğaz gibi - iletişim aksaklıkları nedeniyle hedefe erişemeyebilir yada tamamen bu teknik değişenlerden bağımsız olarak sürücü davranışına bağlı araç hızı ve yönünde ani değişimler meydana gelebilir. Bu ve benzeri daha bir çok neden kümeleme tekniklerinde araçlara özgü daha fazla niteliğin dikkate alınması gerektiğini ortaya koyar.

Bununla birlikte bir kısmında simülasyonda kullanılan modellerden bahsedilmemiş, geri kalanların çoğundaysa ns-2 gibi bir ağ simülatörü VANET ağlardaki mobiliteyi gerçek anlamda karşılamayan modeller ile kullanılmış. VANET ağlarda araçların hareketi rastgelelikten uzak, belirli bir alan içerisinde çoğu zaman doğrusal ilerleyen ancak sürücü davranışına bağlı olarak yön değiştiren bir model ile ifade edilebilir. İnceleme yazısının devamında, kullanılan simülatörler ve farklı mobilite modelleri de incelenecektir.

## VI. Referanslar

- [1] S. Vodopivec, J. Bester, and A. Kos, "A survey on clustering algorithms for vehicular ad-hoc networks," in Telecommunications and Signal Processing (TSP), 2012 35th International Conference on, July 2012, pp 52-56.
- [2] R. S. Bali, N. Kumar, and J. J. Rodrigues, "Clustering in vehicular ad hoc networks: Taxonomy, challenges and solutions," Vehicular Communications, vol. 1, no. 3, pp. 134-152, 2014.
- [3] Cooper, C., Franklin, D., Ros, M., Safaei, F., and Abolhasan, M, "A Comparative Survey of VANET Clustering Techniques," IEEE Communications Surveys & Tutorials, 2016.
- [4] M. Morales, C. seon Hong, and Y. C. Bang, "An adaptable mobility-aware clustering algorithm in vehicular networks," in Network Operations and Managements Symposium (APNOMS), 2011 13th Asia-Pacific Sept 2011, pp. 1-6.
- [5] C. Shea, B. Hassanabadi, and S. Valace, "Mobility-based clustering in vanets using affinity propagation," in Global Telecommunications Conference, 2009, GLOBECOM 2009, IEEE, Nov 2009, pp 1-6.
- [6] R. Goonewardene, F. Ali, and E. Stipidis, "Robust mobility adaptive clustering scheme with support for geographic routing for vehicular ad hoc networks," IET Intell. Transp. Syst., vol. 3, no. 2, p. 148, 2009.
- [7] Y. Chen, M. Fang, S. Shi, W. Guo, and X. Zheng, "Distributed multi-hop clustering algorithm for vanets based on neighborhood follow," EURASIP Journal on Wireless Communications and Networking, vol. 1, pp. 1-12, April 2015.
- [8] Z. Y. Rawashdeh and S. Mahmud, "A novel algorithm to form stable clusters in vehicular ad hoc networks on highways," EURASIP J. Wireless Comm. and Networking, vol. 1, pp. 1-13, Jan 2012.
- [9] L. Bononi and M. di Felice, "A cross layered MAC and clustering scheme for efficient broadcast in VANETs," in 2007 IEEE

International Conference on Mobile Ahdoc and Sensor Systems, Pisa, Italy, Oct. 2007, pp. 1-8.

[10] R. Ghosh and S. Basagni, "Mitigating the impact of node mobility on ad hoc clustering," *Wirel. Commun. Mob. Comput.*, vol. 8, no. 3, pp. 295-308, Mar. 2008.

[11] Basagni S. Distributed clustering for ad hoc networks. In *Proceedings of the 1999 International Symposium on Parallel Architectures, Algorithms, and Networks (I-SPAN'99)*, Zomaya AY, Hsu DF, Ibarra o, Origuchi S, Nassimi D, Palis M (eds). Perth/Fremantle, Australia, June 23-25 1999. IEEE Computer Society, 310-315.

[12] Z. Wang, L. Liu, M. Zhou, and N. Ansari, "A position-based clustering technique for ad hoc intervehicle communication," *Systems, Man, and Cybernetics, Part C: Applications and Reviews*, IEEE Transactions on, vol. 38, no. 2, pp. 201-208, March 2008.

[13] L., Maglaras and D. Katsaros, "Distributed clustering in vehicular networks," in *Wireless and Mobile Computing, Networking and*

*Communications (WiMob)*, 2012 IEEE 8th International Conference on, Oct 2012, pp. 593-599.

[14] M. Almalag, S. Olariu, M. Weigle, and S. El-Tawab, "Tdma cluster-based mac for vanets (tc-mac)," in *World of Wireless, Mobile and Multimedia Networks (WoWMom)*, 2012 IEEE International Symposium on, June 2012, pp. 1-6.

[15] Mohammad S. Almalag and Michele C. Weigle. "Using Traffic Flow for Cluster Formation in Vehicular Ad-hoc Networks," In *Proceedings of the Workshop On User Mobility and Vehicular Networks (ON-MOVE)*, Denver, CO, October 2010.

[16] S. Ucar, S. Ergen, and O. Ozkasap, "Vmasc: Vehicular multi-hop algorithm for stable clustering in vehicular ad hoc networks," in *Wireless Communications and Networking Conference (WCNC)*, 2013 IEEE, April 2013, pp. 2381-2386.

**Tablo IV: İncelenen algoritmalar ve ilgili makalelerdeki karşılaştırılan algoritmalar**

<b>İncelenen Algoritma</b>	<b>Karşılaştırıldığı Algoritmalar</b>
AMACAD	LID, MOBIC, DGMA, MobHid
APROVE	MOBIC, ALM, PPC
DBA-MAC	Yok
DMCNF	K-hop
GDMAC	DMAC
PPC	LID, HD
RMAC	DMAC
Sp-CI	LID
TBC	CBMAC, PPC
TC-MAC	LID, HD, UF
VMaSC	K-hop, MDMAC