

# Telsiz Duyurga Ağlarında Kümeleme ve Yönlendirme İşlemi için Gezgin Etmen Kullanımı

Orhan Dağdeviren<sup>1</sup>, Fatih Tekbacak<sup>1</sup>, Kayhan Erciyeş<sup>2</sup>

<sup>1</sup> İzmir Yüksek Teknoloji Enstitüsü, Bilgisayar Mühendisliği Bölümü

<sup>2</sup> Ege Üniversitesi, Uluslararası Bilgisayar Enstitüsü

orhandagdeviren@iyte.edu.tr, fatih.tekbacak@iyte.edu.tr, kayhan.erciyes@ege.edu.tr

**Özet:** Telsiz Duyurga Ağları (TDA) çok sayıda duyurga düğümünün kendini yönetebileceği şekilde tasarsız bir yapıda kurulur. Kümeleme işlemi sayesinde TDA daha küçük parçalara bölünür ve yönlendirme işlemi kolaylaştırılmış olur[1-4]. Gezgin etmenler, çeşitli alanlarda uygulanmıştır ve TDA üzerinde kümeleme ve yönlendirme işlemleri için kullanılmıştır [5-10]. Bu çalışmada güncel araştırma konuları olan TDA ve gezgin etmen teknolojisinin birlikte nasıl kullanıldığını sunacağız. TDA üzerinde gezgin etmenleri kullanarak kümeleme ve yönlendirme yapan algoritmaları açıklayacak, değerlendirmeleri yapacak ve örnek bir gezgin etmen platformu olarak Agilla[11]'yi göstereceğiz.

**Abstract:** Wireless Sensor Networks (WSN) are constructed by large number of nodes in a self-organizing ad hoc manner[1-4]. WSN is divided into small partitions and routing operation is simplified by clustering operation. Mobile Agents are applied in various fields and are used in clustering and routing operations for WSN [5-10]. In this study, we present how to use WSN and mobile agent technology together which are hot research topics. We explain the clustering and routing algorithms using mobile agents, make evaluations and show the Agilla[11] as an example mobile agent platform.

**Anahtar Kelimeler:** Telsiz Duyurga Ağları, Kümeleme, Yönlendirme, Gezgin Etmen.

## 1. Giriş

Telsiz Duyurga Ağları (TDA) çok sayıda duyurga düğümünün kendini yönetebileceği şekilde tasarsız bir yapıda kurulur. TDAnın düğümlerinin herhangi birinin çevreden topladığı veri, diğer düğümler üzerinden çok sıçramalı bir şekilde çıkış düğümüne ulaşır. Çıkış düğümü, diğer düğümlerden gelen verilerin toplandığı ve kullanıcıya yönlendirilmek üzere diğer ağ yapısına iletiildiği ağ geçididir. Çıkış düğümü Internet üzerinden verilerini gönderebileceği gibi uydu haberleşmesini de kullanabilir. TDAların günümüzde birçok uygulama alanı vardır [10]. En önemli kullanım alanı çevre gözlemleme uygulamaları[11-13] olup afet tespiti[14] ve sağlık alanında[15] da uygulamaları bulunmaktadır.

TDAlar üzerindeki en önemli problemlerden biri de uygulamalar için çevreden toplanan verilerin enerji etkin bir şekilde çıkış düğümüne gönderilmek üzere etkili bir ağ topolojisinin oluşturulmasıdır. Bu şekilde ağ topolojisini oluşturmanın en önemli yöntemlerinden biri kümelemedir. Kümeleme en basit tanımıyla ağ parçalara bölüp her parçaya bir lider atama işlemidir. Kümeleme işlemi sayesinde TDA daha küçük parçalara bölünür ve problemimiz parçalanmış olur [1-4]. TDA kümelenecek yönlendirme işlemi kolaylaştırılır. Küme üyesi düğümler çevreden aldıkları verileri küme lideri düğümüne gönderebilirler. Küme lideri düğümde toplanan verilerin çıkış düğümüne ulaşması için de bir yol tanımlanması gereklidir. Bu yolun adına omurga denir.

Etmen, istemcisi adına belirli işleri yapabilen bir programdır [6]. Etmenler yapay zeka, dağıtık sistemler ve ağ haberleşmesi gibi çok farklı alanlarda kullanılırlar. Kelime işlemcilerinde biçimlendirme ve gramer kontrolü için çalışan programlar etmen örnekleridir. Gezgin etmen, heterojen bir ağ ortamında düğümünden düğüme göçebilen, diğer etmenler ile haberleşebilen bir yazılım etmeni veya düğümler arasında gezip işlemler gerçekleştirebilen bir donanım etmenidir [6]. Geleneksel istemci/sunucu modelinden farklı olarak gezgin etmenler kod, veri ve durumlarını taşıyabilirler. Bu taşıma işlemi etmenin gerçekleştiği dile bağlıdır. Gezgin bir etmen çalışmasını durdurabilir, başka bir düğüme geçebilir ve çalışmasına devam edebilir.

Gezgin etmenlerin iki türlü hareketi vardır: Güçlü ve zayıf göç. Güçlü göç kod, veri ve durumun taşınmasıdır. Yüklü ve zaman tüketici bir işlemdir. Zayıf göç işlemi ise kod ve verinin taşınması fakat durumun taşınmamasıdır. TDAlar üzerinde uygulama tasarımı, kısıtlı bant genişliği, gecikme ve özellikle kısıtlı pil gücünden dolayı zordur. Bu açıdan bakıldığında, zayıf göç işlemi destekleyen etmenler, daha az iletişim yaptıkları ve veri deposu tükettikleri için TDA üzerindeki uygulamalarda kullanılmaya daha müsaittirler [6].

Bu çalışmada gezgin etmenler kullanılarak TDA üzerinde kümeleme ve yönlendirme işleminin yapılması gösterilecektir. 2. Bölümde algoritmalar verilecek ve değerlendirmeleri yapılacak, 3. Bölümde TDA üzerinde gezgin etmenlerin gerçekleşmesi için bir uygulama ortamı olan Agilla[11] gösterilecek, 4. Bölümde ise sonuçlar verilecektir.

## **2. Yaklaşımlar**

Bu bölümde Sugar'ın, Denko'nun, Tong'un, Lotfinezhad'ın, Rajagopalan'ın algoritmaları anlatılacak ve değerlendirilmesi yapılacaktır.

### **2.1. Sugar'ın Algoritması**

Sugar ve Imre kablosuz tasarsız ağlarda kü-

meleme işlemi gezgin etmenler ile yapmak için bir algoritma önermiştir [5]. Kümeleme etmeni düğümlerin üyelik değişimi, kümelerin parçalanması ve birleştirilmesini sağlar. Bunun yanında kümeleme etmenleri küme üyeliği değişimleri veya küme bilgilerini toplamak için komşu etmenler ile haberleşirler. Ağın kurulum aşamasında çok sayıda düğüm etmen üretebilir. Algoritmada dört küme durum değişimi vardır: Komşu kümeler arasında düğüm değişimi, küme bölünmesi, küme birleşmesi ve düğüm keşfi. Düğümlerin hareketinden dolayı, ağın şartlarının değişmesinden dolayı veya diğer sebeplerden dolayı etmenler kümeler arasında hareket edebilirler. Transfere karar veren etmen, transfer olacağı kümeyi seçtikten sonra o kümeyle ait etmen ile iletişim kurup, bir istekte bulunur. Eğer cevap olumlu ise, üyelik değişimi yapılır. Cevabı olumsuz alan etmen transfer olmak üzere farklı bir küme veya farklı bir düğüm seçer. Küme içindeki eleman sayısı eğer çok artarsa ve komşu kümeler transferi kabul etmezse küme bölünmesi yapılabilir. Bölünen küme eşit iki parçaya bölünür, küme etmeni yavrular (spawn) ve yeni bir küme lideri seçilir. Küme birleştirilmesi (merging) işlemi küme bölünmesi işleminin tersidir. Küme birleştirilmesi için işlemi yapacak küme etmenlerinin anlaşması gereklidir. Eğer iki küme etmeni de olumlu karar verirse, iki küme birleşip yeni bir küme oluştururlar.

Kümeleme ile ilgili kararları verirken birkaç ağ parametresi önemlidir. Eğer bu işlem yeteri kadar basit olursa kural tabanlı reaktif etmen ile çözülebilir. Parametre sayısı çok ise hedefe yönelik müzakereci (deliberative) etmenler kullanılabilir. Bu durumda daha zorlu koşullara tepkiler verilebilir. Şu ağ parametreleri kümeleme yapılırken önemli olabilir: düğüm bağlantısı, düğüm hareketleri, düğümün kabiliyetleri, bağ durumu ve bant genişliği.

Etmenler önceki deneyimlerine ve önceki kararlarının etkilerine göre uyum (adaptation) sağlayabilirler. Örneğin bir etmenin önceki

bölünme işlemleri başarılı olmuşsa, gelecek-  
teki bölünme işlemlerine olumlu karar verme  
olasılığı yükselecektir.

## **2.2. Denko'nun Algoritması**

Denko'nun [6] gezgin tasarsız ağlarda küme-  
leme ve yönlendirme için önerdiği yaklaşımda  
bütün gezgin düğümler iki etmen kullanmak-  
tadırlar. Yönlendirici Gezin Etmen (YGE) ve  
Kümeleyici Statik Etmen (KSE) adındaki bu  
etmenler, gezgin araçlara uygun olarak geliştiri-  
lmiş ve Java sanal makinesi üzerinde çalışan  
(örneğin Kilo sanal makinesi) bir etmen plat-  
formunda işletilirler.

KSE, kümeleme bilgisini bir kümeleme tablo-  
sunda güncel olarak tutar. Kümeleme tablosu,  
komşuların bilgisini, düğümün rolünü (küme  
lideri, küme elemanı, küme geçidi), gezginlik  
bilgisini ve derecesini tutar. KSE, küme ile il-  
gili bilgileri topladığında bu tabloyu günceller.  
Küme lideri olan düğüm diğerlerinden özel-  
leşmiş olarak küme içi yönlendirme işleminde  
hak sahibi olan ayrıcalıklı bir düğümdür ve bir  
düğümünden bir paket aldığı anda tercih edilen  
küme geçidi yardımıyla komşu kümeye aktarı-  
r. Bölünme ve birleşme işlemleri ağ üzerin-  
deki tüm kümelerin dengeli olmasını amaçlar.  
Enerjisi azalan, dar boğaz olan veya işlevini  
yitiren küme lideri, yerine yenisinin getirilmesi  
için çalışır.

KSE, kümeleme bilgilerini toplamak ve gün-  
cellemekle yükümlüdür. YGE ise ağ üzerinde  
hareket ederken yönlendirme tablolarını top-  
lama ve değişimlere göre tabloları güncelleme  
görevini üstlenmektedir. Bunun yanında YGE,  
melez bir yönlendirme algoritmasını çalıştırma  
yetisine de sahiptir. Yönlendirme algoritması  
basit olarak şu şekilde çalışır:

Eğer YGE rota bilgisine sahip değil ise

1. Küme liderine göçüp rota bilgisini alma-  
ya çalışıyor.

2. Eğer küme lideri bilgiye sahip ise kendi  
rota bilgisini düzenliyor ve geri göçüyor.

3. Küme lideri bilgiye sahip değil ise diğer  
kümelerin liderlerine veya küme geçidine  
göçüp bilgiyi bulmaya çalışıyor.

Küme içi yönlendirme, küme liderleri tarafın-  
dan veya her düğümün kendine ait yönlendir-  
me etmenleri tarafından yapılmaktadır. Küme  
dışı yönlendirme ise küme liderleri veya geçit-  
ler tarafından yapılır.

Kümeleme mimarisi, küme boyutu, küme lide-  
ri değişimleri, küme üyeliği değişimleri, küme-  
lerin sayısı, küme bölünme ve birleşmesi gibi  
metriklerle değerlendirilir. Belirtilen yaklaşımda  
yönlendirme için performans metrikleri  
ise paket taşınma oranı, yönlendirme yükü ve  
uçtan uca gecikmedir.

Bu doğrultuda bakıldığında normalde iki düğü-  
mün iki adet işlem yapabilmek için toplam dört  
kez mesajlaşması gerekirken beraber etmenle-  
rin iş listesini üzerine aldıkları durumlarda tek  
bir göç ile bu operasyonun gerçekleştirildiği  
görülmektedir.

## **2.3. Tong'un Mimarisi**

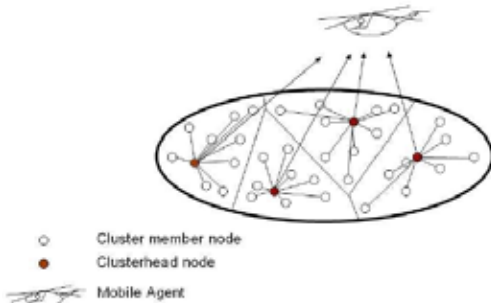
Tong ve arkadaşları [7], geniş ölçekli ve düşük  
enerji gereksinimleri ile çalışan duyarga ağla-  
rı için bir mimari önermişlerdir. Bu mimari-  
de fiziksel katman ve MAC katmanı tasarımı  
bulunmaktadır. Gezin etmenlerin mimariye  
eklenmesi, yoğun şekilde hesaplama ihtiyacı  
hissedilen görevlerin basit yapıdaki duyarga-  
lardan daha güçlü gezgin etmenlere aktarılma-  
sını sağlamıştır.

Gezin Etmenler ile Duyarga Ağları (GEDA)  
yaklaşımına göre, duyargalar, sınırlı işlem ve  
iletişim kapasitesi olan düşük güç ve düşük  
maliyetli düğümlerdir. Çok dikkatli bir ağ dü-  
zenlemesine gerek duymadan çok geniş bir  
alanda dağıtılabilmektedirler. GEDAdaki gez-

gin etmenler ise, hem iletişim hem de çalışma yetenekleri açısından güçlü ve duyurga ağına gezinme yeteneğine sahip donanım birimleridir. Bu etmenler, yüksek veri erişim hızları ile uydularla bağlantı kurabilirler. Gezgin etmenler sürekli olarak duyurgalarla beraber operasyonlar gerçekleştirmeye ihtiyaç duymamaktadırlar, sadece veri toplama ve ağ bakımına ihtiyaç duyulan durumlarda devreye girmektedirler.

#### 2.4. Lotfinezhad'ın Mimarisi

Lotfinezhad ve arkadaşlarının [8] önerisine göre, veriyi merkeze gönderme (reachback) işlemini gerçekleştiren duyurgalar kendi veri paketlerini gezgin etmene iletebilmek için var olan kanalı kullanabilmek amacıyla yarışır. Bu tip veri iletimi enerji verimi ve gecikme süresinin azaltılması amacıyla tasarlanır. Belirtilen amacı gerçekleştirmek üzere sunulan yaklaşım, verinin merkeze gönderildiği gezgin etmenler ile kullanılacak TDAların (K-GEDA) bir kümeleme düzenini önermektedir. K-GEDA, duyurgaları kümeler şeklinde gruplar ve böylece düğümler sadece en yakın küme lideri ile haberleşir, Şekil 1'de küme mimarisi görülmektedir [8]. Küme lideri, veri toplama ve gezgin etmenle haberleşme görevlerini üstlenir. Belirtilen yaklaşımda, küme liderleri kanalı ele geçirebilmek için ALOHA'ya benzer bir düşük yüklü MAC mekanizması kullanılır. Duyurgaların küme liderleri ile tek sıçramada haberleştikleri varsayılır.



Şekil 1. K-GEDA'daki tipik bir ağ mimarisi

K-GEDA'da küme formasyonu, gezgin etmen tarafından tetiklenir. Bir küme liderinin yükünden emin olabilmek amacıyla tüm düğümler boyunca dolanılır ve küme formasyonu her veri toplama aşamasındaki dolaşımarda uygulanır. Kümeleri oluşturabilmek amacıyla her düğüm kendisini sabit bir  $p_k$  olasılığında küme lideri olarak seçer. Eğer bir düğüm küme lideri olursa, diğer küme üyeleri ile veri iletişimi amacıyla statüsünü belirtmek için bir duyuru paketi (ADV) yayınlar. Alınan ADV paketlerinin sinyal gücüne bağlı olarak her düğüm kendisi etrafındaki küme liderlerinin uzaklığını kestirmeye çalışır ve en yakın küme liderinin bulunduğu kümeye katılır. Ağı dolaşan gezgin etmen, iletişimi, ağ bakımını ve talep edildiğinde küme formasyonunu sağlamak için bir *beacon* mesajı yayınlar. Gezgin etmeden yayınlanan *beacon* mesajının alınması ile kümedeki iletişim başlar. Senkronizasyon açısından, düğümler gezgin etmen tarafından yayınlanan *beacon* mesajını kullanırlar.

Tüm veriler bir küme lideri tarafından alındığında, küme lideri veri bütünlemesini sağlar ve veriyi gezgin etmene aktarabilmek amacıyla merkeze gönderme kanalını ele geçirir. Veri birleştirme (aggregation) aşamasında isteğe bağlı paket sayıları tek bir pakete (toplanan bilgilerin minimum, maksimum veya ortalamasını bulmak gibi) indirgenir.

K-GEDA'da bulunan bir küme lideri veri toplamasını bitirdikten sonra topladığı veriyi, bir zaman aralığında (time slot) kanalı ele geçirmeye çalışan küme liderlerinin sayısına bağlı bir olasılıkla iletir. Eğer bu paket doğru bir şekilde ulaştırılmazsa küme lideri diğer zaman aralığında yeni bir olasılıkla paketi tekrar iletmeye çalışır.

#### 2.5. Değerlendirme

Bu bölümde algoritmaları, kullandıkları etmen tipi, enerji etkinliği, ağ yükü ve maliyet açısından değerlendirip karşılaştıracacağız.

**1. Etmen Tipleri:** Denko'nun ve Sugar'ın algoritmalarındaki gezgin etmenler, kodu, durumu ve veriyi taşıyabilen yazılım etmenleridir. Tong'un ve Lotfinezhad'ın yaklaşımlarındaki etmenler, gezgin, güçlü donanım birimleridir.

**2. Enerji etkinliği:** Denko'nun ve Sugar'ın algoritmalarında yazılım etmeni kullanıldığı için düğümlerin üzerindeki kaynaklar kullanılır. Bu durum pil kısıtlı düğümler için bir dezavantajdır. Tong'un ve Lotfinezhad'ın yaklaşımlarında ise gezgin etmenler düğümlerden bağımsız donanımlardır. Tong'un GEDA yaklaşımında, enerji verimi diğer MAC yaklaşımlarına göre üstün yönler içermektedir. Birçok duyarganın olduğu bir ağda iletilen bir komşu duyargadan gelen iletimi dinleyen bir duyarganın enerji tüketimi, özellikle paket alımı sırasında iletim anından 2-3 kat fazla seviyelere çıkabilir. Bunun yanında GEDA yaklaşımında ise duyargaların veri toplamada görev almadığı, sadece gezgin etmenlerin görevli alıcı terminaller oldukları göze çarpmaktadır.

**3. Ağ Yükü:** Sugar'ın ve Denko'nun algoritmaları yazılım etmenlerini kullandıkları için protokol yükü açısından düşünüldüğünde, tansarsız bir ağda her ağ katmanına başlıklar eklenmesi, çerçeveleme ve hata kontrolü özellikleriyle bağlantı kurulması ve devam ettirilmesi yükünün ağ trafiğinin neredeyse %99'unu kapsadığı iddia edilir [7]. Bunun yanında duyarga ağlarındaki düğümlerin komşularının profili hakkında bilgi ihtiyacı da ek bir yük getirmektedir ve komşu profilini güncellemek amacıyla duyargalar sürekli sorgu paketleri gönderirler. GEDA ve K-GEDA, duyarga düğümleri ve gezgin etmenler arasında tek sıçramalı iletim prensibiyle çalışır. Doğrudan gezgin etmenlere iletim, MAC, yönlendirme ve daha üst seviye fonksiyonların getirdiği yükün ciddi seviyede azalmasını sağlar.

**4. Maliyet:** Yazılım etmenlerinin kullanılması etmen olarak kullanılan donanım birimlerine göre daha az maliyetlidir. Çünkü donanım

birimlerinin, gelecek ve bilgileri toplayacak, işleyebilecek ve hatta gerektiğinde uyduyla haberleşecek özelleşmiş cihazlar olması gerekir. İnsanlı ve insansız hava aygıtları bu etmenlere örnektir [7]. Bunun yanında yazılım etmenleri için sadece bir yazılım etmen platformunun kurulması yeterlidir. Sonuç olarak Sugar ve Denko'nun algoritmaları, Tong'un ve Lotfinezhad'ın yaklaşımlarına göre çok daha az maliyetlidir.

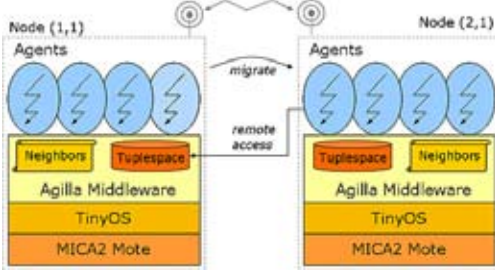
### 3. Agilla

Agilla, TDAda gezgin etmen uygulamalarının üzerinde yazılabileceği bir ara katman yazılımıdır (middleware) [11]. Agilla uygulamaları kod ve durumlarıyla ağ üzerinde göç edebilirler. Gezgin etmenler ağa yayılıp uygulamalar için hassas kontrol sağlayabilir. Gezgin etmenler verinin ağ üzerinde gönderilmesi yerine veri üzerinde işlem yapmayı tercih ettikleri için enerji etkin olduğu söylenmiştir [11]. Agilla etmenleri tüm ağa yayılmak yerine uygulamaya bağlı olarak belli bölgelere yayılabilir.

Gezgin etmenlerin TDA üzerinde birçok uygulaması olabilir. İşgalci tespiti, yangın izlenmesi, kargo izlenmesi gibi uygulamalarda gezgin etmenler kullanılabilir [16-18]. Örneğin bir işgalci, güvenli bir alanı işgal ettiğinde, etmenler ihlalin olduğu yerin yakınlarındaki düğümlerde kendilerini klonlayıp, işgalci çevresinde bir alan oluşturmaya çalışırlar. Sadece bir grup etmen sınırlı sayıda düğüm üzerinde çalıştığı için enerji etkili çözümler oluşturulur.

Agilla mimarisi Şekil 2'de gösterilmiştir. Agilla, TinyOS işletim sistemi [19] üzerinde çalışır ve birden fazla etmenin bir düğüm üzerinde çalışmasına izin verir. Etmen sayısı değişkendir ve öncelikle boş bellek alanıyla belirlenir. Her etmen otonomdur fakat ara katman yazılım kaynaklarını diğer etmenler ile paylaşır. Agilla her düğüme 2 önemli kaynak sağlar: komşuluk listesi ve çokuzlu (tuple) uzayı. Bu kaynaklar etmenlerin nereye hareket edeceğine veya ne-

reye klonlanacağına karar vermek açısından gereklidir. Çokuzlu uzayı etmenlerin iletişimi için gereklidir.



Şekil 2. Agilla Mimarisi

Agilla etmen ara katman yazılımı, TDAlar üzerinde gezgin etmen uygulamalarının gerçekleştirilebileceği örnek bir platformdur. Bu konudaki teorik çalışmalara, uygulama açısından destek olması bakımından çok önemlidir. Algoritma tasarımcıları, benzetim ortamlarında algoritmalarını denedikten sonra Agilla platformu üzerinde gerçek uygulamalarını yapabilirler.

#### 4. Sonuçlar

Bu çalışmada teorik ve uygulama boyutunda gezgin etmenler kullanılarak TDA üzerinde kümeleme ve yönlendirme işlemi gösterilmiştir. Teorik olarak algoritmalar anlatılmış ve değerlendirilmeleri yapılmıştır. Uygulama boyutunda, enerji, bant genişliği ve bellek kısıtları olan TDA düğümleri üzerinde gezgin etmen algoritmalarının gerçekleştirilebileceği Agilla ara katman yazılımı anlatılmış ve gösterilmiştir. Bu çalışmanın güncel iki teknolojinin bir arada kullanımını anlatması açısından gelecekteki çalışmalara faydalı olacağını düşünmekteyiz. Değerlendirme bölümünü performans testi olarak genişletmek ve bir kümeleme protokolü tasarlayıp Agilla platformu üzerinde denemek gelecekteki hedeflerimiz arasındadır.

#### Kaynaklar

- [1] Erciyeş, K., Ozsoyeller, D., Dagdeviren, O., Distributed Algorithms to Form Cluster based Spanning Trees in Wireless Sensor Networks, in Proc. of ICCS, Springer Verlag LNCS 5101, 2008.
- [2] Erciyeş, K., Dagdeviren, O., Cokuslu, D., Ozsoyeller, D., A Survey of Graph Theoretic Clustering Algorithms in Mobile Ad hoc Networks and Wireless Sensor Networks, Journal of Applied and Computational Mathematics, vol. 2-6, pp. 162-180, 2007.
- [3] Dagdeviren, O., Erciyeş, K., Cokuslu, D., A Merging Clustering Algorithm for Mobile Ad hoc Networks, ICCSA, Springer Verlag LNCS, 2006.
- [4] Abbasi, A., A., Younis, M., F., A survey on clustering algorithms for wireless sensor networks, Computer Communications, vol. 30, pp. 2826-2841, 2007.
- [5] Sugar, R., Imre, S., 'Adaptive Clustering Using Mobile Agents in Wireless Ad-Hoc Networks', in Proc. of the 8th International Workshop on Interactive Distributed Multimedia Systems, pp.199-204, 2001.
- [6] Denko, M., K., The use of mobile agents for clustering in mobile ad hoc networks, in Proc. of the 2003 annual research conference of the South African institute of computer scientists and information technologists on Enablement through technology, pp.241-247, 2003.
- [7] Tong, L., Zhao, Q., Adireddy, S., Sensor Networks with Mobile Agents, in Proc. of the Military Communications Intl Symp., 2003.

- [8] Lotfinezhad, M., Liang, B., Energy efficient clustering in sensor networks with mobile agents, in Proc. of the IEEE WCNC'05, pp. 1872-1877, 2005.
- [9] Rajagopalan, R., Mohan, C., K., Varshney, P., K., Mehrotra, K., Multi-objective mobile agent routing in wireless sensor Networks, in Proc. of the IEEE Congress on Evolutionary Computation, vol. 2, pp.1730-1737, 2005.
- [10] Ning X., A Survey of Sensor Network Applications, EURASIP 2005. vol. 5(5) , pp. 774-788, 2005.
- [11] Fok, C.-L., Roman, G.-C., Lu, C., Agilla: A Mobile Agent Middleware for Sensor Networks, Washington University in St. Louis, WUCSE-2006-16, 2006.
- [12] Mainwaring, A., Polastre, J., Szewczyk, R., Culler, D., Anderson, J., Wireless sensor networks for habitat monitoring, in Proc. of the ACM International Workshop on Wireless Sensor Networks and Applications WSNA'02, 2002.
- [13] Biagioni E., Bridges, K., The application of remote sensor technology to assist the recovery of rare and endangered species, in Special issue on Distributed Sensor Networks for the International Journal of High Performance Computing Applications, vol. 16, no. 3, 2002.
- [14] <http://www.alertsystems.org>.
- [15] Schwiebert, L., Gupta, S. K. S., Weinmann, J., Research challenges in wireless networks of biomedical sensors, in Mobile Computing and Networking, pp. 151-165, 2001.
- [16] Hackmann, G., Fok, C. L., Roman, G.-C., Lu, C., Zuver, C., English, K., Meier, J., Agile Cargo Tracking Using Mobile Agents, in Proc. of the Third Annual Conference on Embedded Networked Sensor Systems (SenSys 2005), 2005.
- [17] Bhattacharya, S., Fok, C.-L., Lu, C., Roman, G.-C., A Hierarchical Location Directory Service Across Sensor and IP Networks, in Proc. of the 4th international Conference on Embedded Networked Sensor Systems (SenSys'06), pp. 365-366, 2006.
- [18] Sun, F., Fok, C.-L., Roman, G.-C., sChat: A Group Communication Service Over Wireless Sensor Networks, in Proc. of the 6th international conference on Information processing in sensor networks (IPSN 2007), pp. 543-544, 2007.
- [19] Levis, P., Madden, S., Gay, D., Polastre, J., Szewczyk, R., Woo, A., Brewer, A., Culler, D., The Emergence of Networking Abstractions and Techniques in TinyOS, in Proc. of the First USENIX/ACM Symposium on Networked Systems Design and Implementation, 2004.