

**Mobil Kablosuz Ağlarda Üç Boyutta Konum Tespiti
Algoritmalarının Geliştirilmesi ve Analizi**

Program Kodu: 3501

Proje No: 112E099

Proje Yürütücüsü:
Doç. Dr. Hüseyin AKCAN

Araştırmacı:
Doç. Dr. Cem EVRENDİLEK

Bursiyerler:
Onur ÇAĞIRICI
Ali Teoman UNAY

EKİM 2014
İZMİR

ÖNSÖZ

Bu proje, İzmir Ekonomi Üniversitesi bünyesinde, Doç. Dr. Hüseyin AKCAN yürütücülüğünde, Doç. Dr. Cem EVRENDİLEK'in araştırmacı olarak, Onur ÇAĞIRICI ve Ali Teoman UNAY'ın bursiyer olarak katkılarıyla gerçekleştirilmiştir. Proje 3501 TÜBİTAK Kariyer projesi kapsamında TÜBİTAK tarafından desteklenmiştir. Her iki bursiyerimiz de projenin başından itibaren 24 ay süreyle projeye dahil olmuşlar, yüksek lisans tezlerini bu projede üzerlerine düşen konularda yapmışlardır. Tezlerini oluşturacak kısımlar bu rapor ile birlikte verilmiştir, tezlerinin yazımı ise son aşamasındadır ve bulunduğumuz dönem sonunda bitirmeleri beklenmektedir.

Proje süresince üç boyutta konum tespiti üzerine çalışmalar yapılmıştır. İki boyutta ölçüm hataları olduğu durumlarda karmaşık olan bu problemin üç boyuttaki karakteristiği incelenmiş ve problemi karmaşıklığından kurtaracak şekilde ek kısıtlar eklenerek çözümü üzerine yoğunlaşmıştır. Bu ek kısıtlar üç boyutlu ortamların iki boyutlu düzlemlerden oluştuğu ortamlar (ör. bina katları) ve pozisyonu belirlenmek istenen elemanların birinin veya birden fazlasının hareket edebilir olduğu ortamlar olarak belirlenmiştir. Yapılan çalışmalar sonucunda her iki kısıtın da problemin çözümüne önemli katkıları olduğu tespit edilmiş ve raporlanmıştır. Proje sonucunda hazırlanan bir bildiri uluslararası SIGMOD konferansı bünyesinde onikincisi düzenlenen "ACM International Workshop on Data Engineering for Wireless and Mobile Access (MobiDE)" çalıştayında Haziran 2013'de yürütücü tarafından sunulmuştur. Proje ek olarak iki yüksek lisans öğrencisi için tez çalışması teşkil etmiştir. Her iki tezden de önümüzdeki zamanda yayın beklentisi vardır.

Projede elde edilen sonuçlar ışığında, üç boyutta konumlama için iki boyutlu alt uzayların kullanımının iyi sonuçlar verdiği gözlemlenmiş, bu ve benzeri kısıtlar eklenerek üç boyutta konumlama problemi üzerine daha çok çalışma yapılması kararlaştırılmıştır.

İÇİNDEKİLER

Özet	vi
Abstract	vii
1. Giriş	1
2. Literatür Özeti	2
3. Gereç ve Yöntem	4
3.1 İki boyutta konumlama hatasının azaltılması	7
3.2 Aynı düzlemdeki noktaların tespiti ve düzlemler arası konumlama	7
3.3 Mobil elemanların olduğu durumlarda konumlama	7
4. Bulgular	9
4.1 Bildiri özeti - İki boyutta konumlama hatasının azaltılması	9
4.2 Aynı düzlemdeki noktaların tespiti ve düzlemler arası konumlama	16
4.3 Mobil elemanların olduğu durumlarda konumlama	60
5. Sonuç	75
6. Kaynaklar	76

TABLO VE ŞEKİL LİSTESİ

TABLolar

Tablo 4.2.1 Düzlemsellik faktörünün düzlem sayısı ve düzlem başına düşen düğüm	34
Tablo 4.2.2 Kritik noktalar, en fazla konumlama farkları ve buluşma noktalarının	47

ŞEKİLLER

Şekil 4.1.1 Üçlü pozisyonlama	10
Şekil 4.1.2 Hatalı uzaklık ölçümleri ile üçlü pozisyonlama	10
Şekil 4.1.3 Güvenli çapa noktaları	13
Şekil 4.1.4 Güvenli üçgen testleri	14
Şekil 4.1.5 Üçgen iç açıları testleri	15
Şekil 4.1.6 Güvenli çapa testleri	15
Şekil 4.1.7 Tüm parametre testleri	15
Şekil 4.2.1 Birinci dereceden rijit bir graf.	20
Şekil 4.2.2 Şekil 4.2.1’de görülen birinci dereceden rijit grafın düğümlerinin	20
Şekil 4.2.3 Konumsal sapma.	21
Şekil 4.2.4 3 boyutta bir düğümün bulunabileceği yedi farklı bölge.	22
Şekil 4.2.5 Bir KODun(TBL) yansıması.	22
Şekil 4.2.6 A düğümünün iki uzaklık ölçümü yapılarak konumlanması	24
Şekil 4.2.7 A düğümünün üç uzaklık ölçümü yapılarak konumlanması	24
Şekil 4.2.8 Kesişen iki küre ve kesişim çemberi.	26
Şekil 4.2.9 Kesişen üç küre ve iki kesişim noktası.	26
Şekil 4.2.10 $100 \times 100 \times 100$ bir ortam. $\mu = 0.750$	30
Şekil 4.2.11 $100 \times 100 \times 100$ bir ortam. $\mu = 0.800$	31
Şekil 4.2.12 $100 \times 100 \times 100$ bir ortam. $\mu = 0.900$	31
Şekil 4.2.13 $100 \times 100 \times 100$ bir ortam. $\mu = 0.933$	32
Şekil 4.2.14 $100 \times 100 \times 100$ bir ortam. $\mu = 0.958$	32
Şekil 4.2.15 $100 \times 100 \times 100$ bir ortam. $\mu = 0.992$	33
Şekil 4.2.16 Kesişen düzlemler üzerindeki iki eşdüzlemsel düğüm grubu.	37
Şekil 4.2.17 Her iki algoritmanın uzaklık ölçüm hatası olmadan konumlama	39
Şekil 4.2.18 Her iki algoritmanın uzaklık ölçüm hatası olmadan konumlama	40
Şekil 4.2.19 Her iki algoritmanın uzaklık ölçüm hatası olmadan konumlama	40
Şekil 4.2.20 Her iki algoritmanın uzaklık ölçüm hatası olmadan konumlama	41
Şekil 4.2.21 Her iki algoritmanın uzaklık ölçüm hatası olmadan konumlama	41
Şekil 4.2.22 Her iki algoritmanın uzaklık ölçüm hatası olmadan konumlama	42
Şekil 4.2.23 Her iki algoritmanın uzaklık ölçüm hatası olmadan konumlama	43

Şekil 4.2.24 Her iki algoritmanın uzaklık ölçüm hatası olmadan konumlama	43
Şekil 4.2.25 Her iki algoritmanın uzaklık ölçüm hatası olmadan konumlama	44
Şekil 4.2.26 Her iki algoritmanın uzaklık ölçüm hatası olmadan konumlama	45
Şekil 4.2.27 Her iki algoritmanın uzaklık ölçüm hatası olmadan konumlama	46
Şekil 4.2.28 Quadrilateration için μ ve menzile bağlı olarak gözlenen tasnif edilmemiş	46
Şekil 4.2.29 Grup bilgisi kullanılmadığında, düzlemsellik katsayısı ve hata oranına	48
Şekil 4.2.30 Grup bilgisi kullanıldığında, düzlemsellik katsayısı ve hata oranına	49
Şekil 4.2.31 Grup bilgisi kullanılmadığında, düzlemsellik katsayısı ve hata oranına	49
Şekil 4.2.32 Grup bilgisi kullanıldığında, düzlemsellik katsayısı ve hata oranına	50
Şekil 4.2.33 Grup bilgisi kullanılmadığında, düzlemsellik katsayısı ve hata oranına	51
Şekil 4.2.34 Grup bilgisi kullanıldığında, düzlemsellik katsayısı ve hata oranına	52
Şekil 4.2.35 Şekil 4.2.34'ün yakınlştırılmış hali.	52
Şekil 4.2.36 Grup bilgisi kullanılmadığında, düzlemsellik katsayısı ve hata oranına	53
Şekil 4.2.37 Grup bilgisi kullanıldığında, düzlemsellik katsayısı ve hata oranına	54
Şekil 4.2.38 Grup bilgisi kullanıldığında, düzlemsellik katsayısı ve hata oranına	54
Şekil 4.2.39 Grup bilgisi kullanıldığında, düzlemsellik katsayısı ve hata oranına	55
Şekil 4.2.40 Grup bilgisi kullanılmadığında, düzlemsellik katsayısı ve hata oranına	56
Şekil 4.2.41 Grup bilgisi kullanılmadığında, düzlemsellik katsayısı ve hata oranına	56
Şekil 4.2.42 Şekil 4.2.40'ın sadece orta μ değerleri için yakınlştırılmış hali.	57
Şekil 4.2.43 Şekil 4.2.41'in tüm μ değerleri için yakınlştırılmış hali.	57
Şekil 4.2.44 Grup bilgisi kullanılmadığında konumlama yüzdesinin hatalı gruptama	58
Şekil 4.2.45 Grup bilgisi kullanıldığında konumlama yüzdesinin hatalı gruptama	59
Şekil 4.3.1 Flip Belirsizliği	62
Şekil 4.3.2 Hex çember	63
Şekil 4.3.3 Hexer birim, hex çemberi dahilindeki altı olası noktaya ulaşabilir	65
Şekil 4.3.4 "Pisagor Teoremi" baz alınarak, iki eşkenar heksagonun	67
Şekil 4.2.5 Hexer birimin, trilaterasyon yapabilmesi için gri bölgeye	68
Şekil 4.3.6 "Puanlı Hareket" "Rasgele Hareket"e göre daha iyi sonuç veriyor	71
Şekil 4.3.7: Hexer menzilini arttırmak, adım sayısını azaltmak için iyi	72
Şekil 4.3.8: Kat yüksekliği azaldıkça, hatalı gruptama oranı artıyor	73
Şekil 4.3.9: Hexer birimin menzilini, kat yüksekliğinden daha kısa	73
Şekil 4.3.10 Hexer birimin menzilini arttırmak	74

ÖZET

Mobil kablosuz ağların veya kablosuz algılayıcı ağların kullanıldığı alanlarda, ağ elemanlarının konumlarının tespiti önemli bir araştırma alanıdır. Günümüzde bu ağlar üzerinde çalışan pekçok kritik uygulama hassas konum tespiti yapabilen araçlara gerek duymaktadır. Fakat hassas konum tespiti alanındaki tüm zorluklar henüz aşılamamıştır. Genellikle yaygın olarak kullanılan GPS veya benzeri uydudan konumlama tabanlı sistemler ya yeterli hassasiyeti verememekte veya kapalı alanlarda uydularla doğrudan bağlantı kuramadığı için çalışması mümkün olmamaktadır. Bu proje kapsamında önerdiğimiz konumlama algoritmaları uydudan konumlamanın uygun olmadığı uygulamalar için geliştirilecek konumlama algoritmalarıdır. Bu uygulamalarda uydu yerine elemanların komşu ağ elemanlarıyla bağlantıya geçerek tasarsız biçimde (ad-hoc) konumlama yapılabilmesi ve tüm işlemlerin yerel (local) bilgi kullanılarak herhangi bir altyapıya ihtiyaç duymadan yapılması önemlidir. Bu uygulamalara örnek olarak bir yangına müdahale etmek için yanan bina içerisine giren itfaiye takımının anlık konum bilgilerinin tespiti gösterilebilir. İtfaiye erlerinin üzerinde konumlamayı sağlayacak radyo donanımları olduğu durumda erlerin hareketlerinin takibi harici bir operatör veya şef tarafından yapılabilir. Erlerin bina içerisindeki birbirlerine ve binaya göre konumları anlık belirlenir, izledikleri yollar, kontrol edilen odalar, koridorlar kayıt altına alınabilir. Herhangi bir yere yardım sevk etmek gerektiğinde pozisyonu en uygun kişi buraya yönlendirilebilir. Bu tarz bir uygulamanın pratikte çalışması için herhangi bir altyapıya ihtiyaç duymayacak şekilde her ortamda kolay kurulması ve hassas konum tespiti yapması gerekir.

Bu projenin amacı uydudan konumlamanın yapılamadığı bina içi gibi ortamlarda mobil elemanların uzaklık ölçümüne dayalı algoritmalar kullanarak üç boyutlu yön bilgisinde içeren konumlamanın yapılmasıdır. Konumlama yapılırken elemanların bir kısmının sabit bir kısmının hareketli veya hepsinin hareketli olduğu durumlar göz önüne alınmıştır. Proje yürütücüsünün ağ elemanların hepsinin mobil olduğu durumlarda iki boyutlu yönsel konumlama yapabilen literatüre geçmiş algoritmaları Akcan vd. (2006, 2010) mevcuttur. Buna ek olarak yine proje ekibi tarafından yakın zamanda ağ elemanlarının hepsinin sabit olduğu durumlarda iki boyutlu yönsel konumlama yapabilen algoritmalar Akcan vd. (2011) geliştirilmiştir. Elemanların hepsinin veya bir kısmının mobil olduğu durumlarda uzaklık ölçümüne dayalı üç boyutlu yönsel konumlama yöntemleri üzerine literatürde henüz bir çalışma yoktur. Proje kapsamında üç boyutlu yönsel konumlama yöntemleri üzerine araştırmalara yoğunlaşmıştır.

Anahtar Kelimeler: yönsel konum tespiti, mobil kablosuz ağlar, kablosuz algılayıcı ağlar, algoritmalar

ABSTRACT

Node localization is an important research area in mobile wireless networks or mobile sensor networks. Recently, many safety critical applications designed for these networks require precise localization algorithms to operate properly. However, accurate localization is a non-trivial problem. GPS or similar satellite based localization methods either do not work indoors and visually obstructed areas, or they cannot provide the level of accuracy required for many applications. The localization algorithms we propose in this project are GPS-free localization algorithms designed for applications that do not require GPS signals. For these applications the local neighborhood graph and the inter-distances of nodes in an ad-hoc network are used without the need for an infrastructure. Realtime position information of a firefighter team inside a building can be given as an example application of GPS-free localization. In case the firefighters have a radio to perform localization, the realtime positions of the firefighters can be monitored by an external operator or chief. The relative positions of the firefighters inside the building can be gathered, the paths, rooms or corridors examined by the crew can be recorded. In case of an emergency, the nearest crew member can be dispatched to the target area within the building. Such an application requires ad-hoc deployment without any infrastructure setup time and accurate localization to work properly.

The aim of this project is to perform directional localization of mobile nodes using ranging for scenarios where the GPS localization is not possible. Localization is done for networks with all nodes mobile or for networks that have both mobile and static nodes. The principal investigator of this project already has directional localization algorithms in 2D for mobile networks Akcan vd. (2006, 2010). The project team also has directional localization algorithms for static networks Akcan vd. (2011). To the best of our knowledge, there is no work in literature that investigates the 3D directional localization case for mobile wireless networks. Therefore, the main focus of this project is 3D directional localization.

Keywords: directional localization, mobile wireless networks, mobile sensor networks, algorithms

1. GİRİŞ

Bu projede üç boyutta konum tespiti problemi üzerine arařtırmalar yapılmıřtır. İki boyutta karmařık (NP-Complete) olan problemin üç boyut karakteriřtiđi arařtırılmıř ve karmařıklıđı azaltacak řekilde kısıtlar eklenmesi ile bu kısıtların etkileri üzerine arařtırmalar yapılmıřtır. Yapılan çalıřmalar üç ana bařlıkta özetlenebilir. Bunlardan birincisi iki boyutta karmařıklıđa sebep olan düzlemsel evrimlerin oluřma sebeplerinin tespiti, bu sebepleri azaltacak veya ortadan kaldıracak řekilde konumlama yapılması sonucu karmařıklıđın azaltılması. İkinci ana bařlık olarak üç boyutta konumlama probleminin iki boyutlu alt düzlemlere ayrılması ve öncelikle bu düzlemler ierisinde, sonra ise düzlemler arası konumlama yaparak sonuca ulařma olarak verilebilir. Bu řekilde alt boyutlarda yapılacak çalıřmaların konumlama bařarımı üzerine etkisi arařtırılmıř ve sonuçlar raporlanmıřtır. Son ana bařlık olarak da konumlanmak istenen elemanlardan en az birinin hareketli olması durumunda ve bina ileri gibi üç boyutlu ortamın üst üste duran iki boyutlu düzlemlerle modellenebileceđi kısıtlı durumlarda bu iki parametrenin konumlama bařarımı üzerine etkisi arařtırılmıř ve modellenmiřtir.

Bu raporun geri kalan kısmında Bölüm 2'de literatür özeti verilmiřtir. Bölüm 3'de gere ve yöntemler belirtilmiř ve Bölüm 4'de bulgular sunulmuřtur. Yukarıda bahsedilen üç ana bařlık bulgular bölümünde ayrıntısıyla Bölüm 4.1, 4.2 ve 4.3'de tartiřılmıř ve sonuçlar raporlanmıřtır. Bölüm 5'de ise sonuç kısmı verilerek rapor sonlandırılmıřtır.

2. LİTERATÜR ÖZETİ

Günümüzde mobil ağlar ve bu ağlar üzerinde çalışan uygulamaların artmasıyla mobil elemanların mutlak veya görece konumlarının tespiti üzerine araştırmalar hız kazanmıştır. Konumlama konusunda, eğer sadece mobil elemanların aralarındaki uzaklıklar biliniyorsa bu problem çizge tümleştirmesi (graph embedding) problemine indirgenebilmektedir. Yaylarında (edge) ağırlık olan çizgelerin (graph) tümleştirilebilirliğinin (embeddability) testinin NP-hard olduğu Saxe (1999) tarafından gösterilmiştir. Burada sonuç genel çizgeler (general graph) için rapor edilmiştir fakat mobil ağlarda veya kablosuz algılama ağlarında genel olarak ağ yapıları “unit disk graph” Clark vd. (1990) olarak modellenmektedir. Konumlama probleminin “unit disk graph” tipi çizge yapılarında da NP-hard olduğu Aspnes vd. (2004, 2006) tarafından ispatlanmıştır. Konumlama konusunda araştırılan bir başka alan ise çizge yapısının katı (rigid) çizge olduğu durumdur. Rigidity teorisine göre çizge yapısının “global rigid graph” olduğu durumlarda bile konumlama problemi NP-hard olarak kalmaktadır Eren vd. (2004). Diğer bir yandan çizge yapısının “global rigid” olduğu durumlarda konumlama probleminin tek bir çözümü olduğu da bilinmektedir. Eren vd. (2004) ayrıca üçlü pozisyonlama tekniği (trilateration) kullanılarak, uzaklık ölçümlerinde hata olmadığı durumlarda konumlama probleminin kısıtlı bir hali olan üçlü pozisyonlamanın polinom zamanlı çözümü olduğunu göstermiştir.

Hightower vd. (2001) çeşitli konumlama sistemlerinin ve kullandıkları tekniklerin birbirleriyle kıyaslamasını vermektedir. Konum tespitinde en çok kullanılan yöntemlerden biri uydudan konumlama tekniğidir. GPS buna bir örnek olarak gösterilebilir. GPS kullanarak konum tespiti yapabilmek için en az 4 uydu ile iletişim kurulup konum bilgisi alınması gerekmektedir. GPS uyduları ile belirli şartlarda doğrudan bağlantı kurulamamaktadır, bu şartlara örnek olarak bina içleri gibi kapalı alanlar gösterilebilir. Bu durumlarda konumlama yapılamamaktadır. Bir diğer sorun ise birbirine yakın elemanların konumlamasında GPS sistemlerinin yeterli hassas konumlama bilgisi sağlayamamasıdır. Bu durum araştırmacıları uydu sistemleri dışında konumlama teknikleri araştırmaya yönlendirmiştir. Bu alanda yapılan pekçok çalışma (Bulusu vd., 2000; Doherty vd., 2001; Liu vd., 2006; Nagpal vd., 2003; Savarese vd., 2002; Savvides vd., 2001) sabit kablosuz algılama ağlarında bilinen referans noktalara, anchor veya seed olarak da adlandırılan, göre pozisyonlamayı kapsamaktadır. Anchor ve seed kullanıldığı durumlarda bu elemanların pozisyonları genelde GPS tarafından belirlenmekte veya önceden ayarlanmaktadır. Bu durum pratikte ihtiyaç duyulan tasarsız kullanılabilir konumlama yapısına uygun düşmemektedir. Pekçok uygulamada (Akcan vd., 2006; Akcan vd., 2010; Akcan vd., 2011; Capkun vd., 2002; Iyengar vd., 2003; Moore vd., 2004; Park vd., 2008; Priyantha vd., 2003) elemanların dünya üzerindeki mutlak konumları

yerine yerel koordinat sistemi içerisindeki görece konumları yeterli olmaktadır, dolayısıyla uydu sistemleri veya bir altyapı kullanmadan konumlama yapılabilinmektedir. Bu durumda konumlama için sadece komşuluk bilgileri ve elemanlar arası uzaklıklar gibi yerel bilgiler yeterli olmaktadır. Elemanlar arası uzaklık ölçümlerinde gerçek hayatta her zaman hata olması beklenmektedir. Bu durum araştırmacıların uzaklık bilgilerini kullanmadan konumlama yapabilecek algoritmalar (Bulusu vd., 2000; He vd., 2005; Nagpal vd., 2003; Niculescu vd., 2003) önermelerine yol açmıştır. İstatistik yöntemleri kullanarak konumlama yapan (Hu vd., 2004; Rudafshani vd., 2007) ve veri analizi teknikleri kullanarak konumlama yapan (Giorgetti vd., 2007; Shang vd., 2003; Stathopoulos vd., 2007) algoritmalar da bulunmaktadır.

Yakın zamanda tek bir kablosuz algılayıcı eleman üzerinde çift radyonun kullanıldığı algoritmalar önerilmiştir (Lymberopoulos vd., 2008; Pering vd., 2008). Ağırlıklı olarak yön tespiti (routing) amaçlı kullanılan bu algoritmalarda genelde radyolardan biri düşük enerji tüketimli kısa mesafe iletişim kurarken diğeri daha yüksek enerji gerektiren uzak mesafe iletişim radyolarıdır.

Konumlama sabit ağlarda olduğu gibi sadece belirli elemanların mobil olduğu (Li vd., 2011; Priyantha vd., 2005) veya tüm elemanların mobil olduğu ortamlarda da araştırılmıştır (Akcan vd., 2006; Akcan vd., 2010; Park vd., 2008; Zhang vd., 2008). Buna ek olarak konumlamanın araştırıldığı bir diğer disiplin de araçlar arası ağlardır (vehicular ad-hoc networks) (Benslimane vd., 2005; Boukerche vd., 2008; Kukshya vd., 2005; Parker vd., 2006).

Uzaklık ölçümüne dayalı konumlama algoritmaları hassas uzaklık ölçümü yapan sistemlere ihtiyaç duymaktadır. Literatürde değişik uzaklık ölçüm teknikleri (Caffery vd., 2008; Gustafsson vd., 2005; Patwari vd., 2005; Porretta vd., 2008) mevcuttur. Günümüzde bu konu en hassas bir şekilde Ultra-wideband radyoların standandında gelen uzaklık ölçüm opsiyonu ile yapılabilmektedir (Gezici vd., 2005; Şahinoğlu vd., 2008).

Yönsel konumlama tanımı Akcan vd. (2006, 2010) komşu elemanların her elemanın kendi koordinat sisteminde yönsel ve uzaklık olarak konumlamasının yapılmasıdır. Yönsel konumlama konumlamanın mobil elemanlar için hareket koordinasyonu yapmalarının da önünü açması bakımından önemlidir.

Gerçek hayatta ölçüm hataları kaçınılmazdır. Bu hatalar konumlama algoritmalarının da başarımını etkilemektedir. Kablosuz algılama ağlarında sıklıkla oluşan hatalar Ni vd. (2009) da özetlenmiştir. Hatalı durumların konumlamaya etkisi (Anderson vd., 2010; He vd., 2005; Liu vd., 2006; Moore vd., 2004; Savarase vd., 2002) de araştırılmıştır.

3. GEREÇ VE YÖNTEM

Mobil kablosuz ağlarda konum tespiti, literatürde üzerinde pekçok araştırmanın yapıldığı bir alandır. Literatürde konum tespiti için çeşitli yöntemler önerilmiştir. Bunlardan en bilineni GPS benzeri uydu sistemleri kullanarak yapılan konumlamadır. Bu tarz konumlamada ağ elemanın dünya üzerindeki mutlak konum bilgisi belirli bir hata çerçevesinde elde edilebilir. Uydu kullanarak yapılan konumlamanın birkaç temel problemi vardır. Bunlardan en önemlisi uydularla doğrudan temas kurulmadığı durumlarda bu sistemlerin çalışmamasıdır. Bu durumlara örnek olarak bina içleri veya yüksek binalarla çevrili sokaklar gösterilebilir. Diğer bir önemli problem ise yakın mesafedeki elemanların konumlarının tespitini gerektiren uygulamalarda yeterli hassasiyetin sağlanamamasıdır. Tipik bir GPS sistemi 30 metre civarı konumlama hatası yapar, bu da elemanların arasındaki mesafenin 30 metreden az olduğu uygulamalar için GPS konumlamasını kullanılamaz hale getirmektedir. Bu sebeplerden dolayı literatürde GPS sistemlerine alternatif konumlama algoritmaları geliştirilmiştir.

Uydu konumlamaya alternatif geliştirilen algoritmaların genel uygulama alanları kablosuz ağlarda konumlama yapmaktır. Bu yöntemler özetle altyapıya ihtiyaç duyan sistemler (anchor based, seed based), olasılık tabanlı sistemler, ve uzaklık ölçümü kullanmadan konumlama yapan (range-free localization) sistemler olarak verilebilir. Altyapıya ihtiyaç duyan sistemler bizim uygulama alanlarımızda kurulum maliyeti getirmekte ve tasarsız yapıya uymadıkları için uygulamamızın ihtiyaçlarını karşılamamaktadır. Olasılık tabanlı ve uzaklık ölçümü kullanmadan konum tespiti yapan algoritmalar ise yeterli hassasiyeti sağlayamamaktadırlar. Bu projenin amaçladığı konumlama algoritmalarının kullanımına uygun bir örnek olarak acil durumlarda arama kurtarma yapacak ekiplerin koordinasyonu ve takibi verilebilir. Literatürde Avrupa Birliği projesi kapsamında benzer çözümler önerilmiştir Lo vd. (2008). Bu çözümlerin bizim tasarladığımız konumlama sisteminden en önemli farkı tasarsız yapıyı desteklememeleri ve konumlama için pozisyonlarını uydudan alan baz istasyonu tabanlı alt yapıya ihtiyaç duymalarıdır. Fakat acil durum olan her ortamda bu altyapının olması beklenemeyeceği gibi, yeni bir altyapı kurulması da zaman ve maliyet gerektirdiği için acil arama kurtarma gibi anında müdahale gerektiren uygulamalarda pratik olmayacaktır. Bu tarz arama kurtarma faaliyetlerine bu sebeplerden dolayı en uygun yapı tasarsız kablosuz ağlar kullanılarak yapılan konumlamadır.

Projenin amaçladığı konum tespiti yöntemini kullanabilecek bir diğer uygulama da insansız hava araçlarının yakın mesafe uçuşlarında veya yine insansız kara araçlarının hareket koordinasyonunda mutlak koordinatlar yerine birbirlerine görece yönsel pozisyonlarının hassas bir şekilde bulunmasıdır. Bu tarz uygulamalarda konumlamanın hareketi

destekleyebilmesi için yön bilgisini de içermesi gerekmektedir. Literatürde uzaklık ölçümü tabanlı konumlama algoritmalarında Capkun vd. (2002) sadece uzaklık ölçümlerine dayalı pozisyonlama yapılmış fakat yön bilgisi saklanmamıştır. Üçlü pozisyonlama tekniği tabanlı bu sistemlerde üç referans noktanın pozisyonunun bilinmediği durumlarda en yakın iki elemanın pozisyonları elde olan uzaklıklarla tutarlı olacak şekilde koordinat ekseninde pozitif koordinatlarda varsayılmakta ve geri kalan tüm elemanların pozisyonları da bu sanal koordinat sistemine göre hesaplanmaktadır. Sonuçta elde edilen pozisyon bilgileri üçlü pozisyonlamaya uygun “global rigid” bir çizge yapısı oluştursa da başta varsayılan pozisyonlar sebebiyle yön bilgisini yitirmektedir. Genel olarak yön bilgisi kullanmadan yapılan konumlama yol saptama (routing) uygulamaları için düşünülmüş olup mobil araçların hareket koordinasyonu için uygun değildir. Bu tarz uygulamaları desteklemek için daha önceki çalışmalarımızda Akcan vd. (2006, 2010) yönsel pozisyonlama kavramı ortaya atılmıştır. Yönsel pozisyonlamanın sağladığı avantaj konumlamada uzaklık bilgisinin yanında yerel koordinat sistemine göre yön bilgisini de içermesidir. Böylece iki araç çarpışma durumu farkedip birbirlerinden uzaklaşmak gerektiğinde bu yön bilgisini kullanabileceklerdir.

Sadece uzaklık ölçümüne dayalı konum tespiti probleminin NP-hard olduğu bilinmektedir. Diğer taraftan ağ yapısının “global rigid” olduğu durumlarda üç referans noktadan başlayarak, her seferinde konumu bilinen üç referans noktaya göre yer tespiti yapma yöntemi üçlü pozisyonlama (trilateration) tekniğidir. Bu yöntem literatürde iki boyutta konumlama için kullanılmıştır ve polinom zamanda çalışan bir algoritmadır. Üç referans noktanın pozisyonlarının önceden bilinmediği durumlarda ise yine seçilen bir noktayı referans alarak diğer noktaların bu noktaya göre konumları üçlü pozisyonlama tekniği uygulanarak bulunabilir. Bu durumda literatürde genel olarak pozisyonu bilinmeyen diğer iki noktanın pozisyonları için uzaklıklarla tutarlı olacak şekilde pozisyonlar varsayılır. Bu yaklaşımın bizim açımızdan sakıncası varsayımsal pozisyonlar kullanıldığından dolayı yön bilgisinin kaybedilmesidir. Mobil elemanların belirli bir formasyonda yakın plan hareketlerinin sağlanması için görece pozisyonlarının yanısıra yön bilgisinin de bilinmesi gerekmektedir. Buna çözüm olarak daha önceki çalışmalarımızda iki boyutta elemanların hareket miktarları ve hareket yön bilgilerini kullanarak pozisyonlama Akcan vd. (2006, 2010) veya her eleman üzerinde iki radyo olduğu varsayımıyla pozisyonlama Akcan vd. (2011) üzerine araştırmalar yapılmıştır. Buradaki temel varsayım konumlamamanın yapılabilmesi için pozisyonu öğrenilmek istenen nokta dışında, iki boyutta, üç farklı noktanın pozisyonlarının daha bilinmesi gerekliliğidir. Çözüm olarak hareket bilgisi kullanıldığında iki elemanın hareketinden aranan elemanın pozisyonu dışında üç farklı nokta, veya iki radyo kullanıldığında hareket olmasa bile üç farklı nokta elde edilebilmekte ve bu pozisyonlar kullanılarak konumlama yapılabilmektedir. Anlatılan bu yöntemlerin hepsi iki boyutta yönsel konumlama için

geçerlidir. Üç boyutta konumlama için tekrar rigidity teorisine dönersek aynı düzlem üzerinde olmayan 4 farklı referans noktaya olan uzaklığa ihtiyacımız vardır. Bu durumda iki elemanın hareketi veya sabit elemanlar için her elemanda iki radyo olması yeterli sayıda pozisyonu bilinen referans noktayı sağlamamaktadır, dolayısıyla konumlamayı sağlamak için problemin bize sunduğu hareketten faydalanmak gerekmektedir. Bu da hareketin doğru bir şekilde takip edilebildiği durumlar için birbirine yakın üç elemanın hareketinden pozisyonu bilinmek istenen eleman dışında en fazla beş farklı nokta, veya iki radyolu iki elemanın hareketi ile yedi farklı nokta elde edilmesi anlamına gelmektedir. Burada rigidity teorisi bize tüm referans noktaların aynı düzlem üzerinde olmadığı durumlarda teorik olarak konum bilgisi elde edebileceğimizi söylemektedir.

Projede temel alınan noktalardan biri de kablosuz ağ elemanları arasında uzaklık ölçümlerinin yapılabildiği varsayımdır. Bu konuda uzaklık ölçümleri ne kadar hassas yapılırsa bizim konumlama algoritmamızın da o kadar hassas konumlama yapacağı açıktır. Literatürde uzaklık ölçümleri için pekçok teknik kullanılsa da, biz proje kapsamında ultra-wideband (UWB) radyolarda uzaklık ölçümü tekniğini kullandığı varsayılmıştır. UWB radyolar IEEE 802.15.4a standardına göre seçenekli olarak hassas uzaklık ölçümü yapabilmektedir. Bu standarda göre UWB radyolar LOS durumları için 3.5cm ve NLOS durumları için 6.6cm civarı hatalarla uzaklık ölçümü yapabilmektedir*. Bu da bizim üzerinde çalıştığımız pekçok uygulama için yeterli olmaktadır.

(*) Time Domain P400 RCM Data Sheet. <http://www.timedomain.com/datasheets/320-0289BP400RCMDataSheetFinal.pdf>

Proje kapsamında yukarıda belirtilen problemleri çözmek için yöntem olarak üç temel konu üzerinde araştırmalar yoğunlaştırılmıştır. Bu alanlar sırasıyla:

1. İki boyutta uzaklık hatalarının olduğu durumlarda konumlamayı zorlaştıran düzlemsel evrimlere sebep olan durumların tespiti, konumlama sırasında düzlemsel evrim sayısının azaltılması ve bu durumun konumlama başarımına etkisinin incelenmesidir. Bu kısım ile ilgili detaylar Bölüm 3.1 de verilmiştir.
2. Üç boyutta konumlama problemini kısıtlamak amacıyla öncelikle aynı düzlemde olan elemanların konumlanması, sonrasında düzlemler arası konumlama algoritmalarının geliştirilmesi. Bu kısım ile ilgili detaylar Bölüm 3.2 de verilmiştir.
3. Hareketli kablosuz elemanlarımızın olduğu senaryolarda hareket bilgisini kullanarak, yine düzlemlere ayrılmış ortamlarda konumlama yapılması. Bu kısım ile ilgili detaylar Bölüm 3.3 de verilmiştir.

3.1 İki boyutta konumlama hatasının azaltılması

İki boyutta uzaklık ölçümlerinde hata olduğu durumlarda konumlama probleminin karmaşık olduğu bilinmektedir. Problemin karmaşık olmasındaki temel sebep düzlemsel evrim olarak adlandırdığımız konumlama hatalarının oluşmasıdır. Bu hatalar hem konumlama hatasını yüksek oranda arttırmakta, hem de problemin karmaşıklığına sebep olmaktadır. Bu kısımda problemin karmaşıklığını azaltmak için düzlemsel evrim durumlarına sebep olan konumlama halleri listelenmiş, sebepler incelenmiş ve düzlemsel evrim durumlarını sona erdirecek şekilde metodlar geliştirilmiştir. Düzlemsel evrim sayılarının sıfırlanması sonucunda da konumlama başarımının arttığı deneyler sonucunda gözlemlenmiştir. Elde edilen bu bulgular uluslararası bir bildiri olarak *“Reducing the number of flips in trilateration with noisy range measurements”* başlığı ile sunulmuştur, Akcan vd (2013). Bu bildiri ile ilgili detaylı özet ve kullanılan yöntemlerin detayları Bölüm 4.1’de verilmiştir.

3.2 Aynı düzlemdeki noktaların tespiti ve düzlemler arası konumlama

Genel olarak konumlama problemleri incelendiğinde, gerçek hayatta her ne kadar yaşadığımız ortam üç boyutlu da olsa, konumlanmak istenen cisimlerin ve insanların iki boyutlu şekiller üzerinde durduğu gözlemlenmiştir. Bu duruma açık havada arazide duran cisimler de dahildir. Bina gibi üç boyutlu ortamlar, veya dağ gibi ortamlar ise üç boyutlu olmalarına rağmen iki boyutlu kesitlerin birleşimi olarak da modellenebilirler. Bu durum bizim üç boyutlu modelleme problemimizi iki boyutlu pekçok düzlem üzerinde duran cisimlerin modellenmesi problemine çevirir. Konumlama probleminin bu şekilde iki boyutlu düzelemlere indirgenebildiği durumlarda, konumlama öncelikle aynı düzlemde olan cisimlerin belirlenmesi, bu düzelemler içerisinde yerel konumlamanın iki boyutta yapılması ve son olarak da düzlemler arası konumlamanın yapılarak tüm ortamın üç boyutlu konumlamasının çıkarılması olarak yeniden modellenebilir. Bu durumda düzlemler kullanılarak yapılan konumlama ile doğrudan üç boyutlu ortam üzerinde düzlemleri gözönüne almadan yapılan konumlama algoritmaları karşılaştırılmış, ve düzlemler kullanarak yapılan konumlamanın belirli durumlarda ve düzlemlerin durumuna göre hız ve doğruluk bakımından avantajlar sağladığı deneyler sonucunda gözlemlenmiştir. Bu kısımla ilgili çalışmalar bursiyerlerimizden Onur Çağırıcı tarafından yapılmış ve detaylı bilgiler Bölüm 4.2’de verilmiştir.

3.3 Mobil elemanların olduğu durumlarda konumlama

Projenin bu bölümünde yine Bölüm 3.2’de olduğu gibi düzlemlere ayırmanın konumlama problemi üzerindeki başarımı incelenmiş, fakat bu bölümde özellikle bina içi konumlama senaryolarına odaklanılmıştır. Konumlamanın üç boyutlu bir bina içerisinde olduğu varsayılmış, bu durumda konumlanmak istenen elemanların bina katlarına denk gelecek

şekilde iki boyutlu düzlemlerde bulunduğu durumlarda birbirine paralel olarak uzanan düzlemlerin konumlama işlemlerini ne kadar kolaylaştırdığı incelenmiştir. Bu bölümde ek olarak mobil bir elemanın bina içerisinde dolaştığı ve bu eleman vasıtasıyla aynı kattaki (düzlemdeki) elemanların belirlendiği ve düzlemler arası konumlanmanın gerçekleştirdiği varsayılmıştır. Bu mobil eleman için en az hareket ile en iyi konumlamayı yapacak şekilde algoritmalar geliştirilmiştir. Bu kısımla ilgili çalışmalar bursiyerlerimizden Ali Teoman Unay tarafından yapılmış ve detaylı bilgiler Bölüm 4.3'de verilmiştir.

4. BULGULAR

Bu bölümde proje süresince araştırma yapılan üç temel konu üzerine yapılan çalışmalar detaylı olarak verilmiştir. Bölüm 4.1'e konu olan bildiri bu rapora ek olarak sunulmuştur. Bölüm 4.2 ve 4.3 'de elde edilen sonuçların yayına yönelik olarak çalışmaları devam etmekte, bu iki konudan da yayın beklenmektedir. Bölüm 4.2 ve 4.3' deki konular yine bursiyerlerimizin yüksek lisans tezlerini oluşturmaktadır, tez çalışmalarında son aşamaya gelinmiştir.

4.1 Bildiri özeti - İki boyutta konumlama hatasının azaltılması

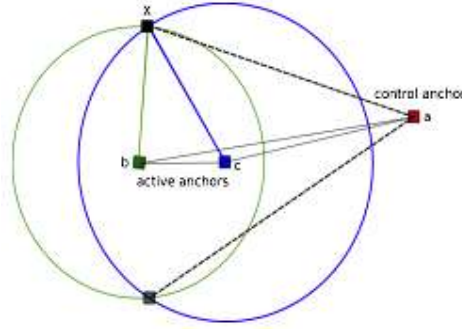
Proje planı kapsamında üçlü pozisyonlamanın üç boyut için genellenmesi üzerine çalışmalar yapılmıştır. Bu çalışmalar kapsamında öncelikle hata durumlarında üçlü pozisyonlamanın gürbüzlüğünü arttıracak şekilde kriterler belirlenmiş ve bu kriterler uygulandığında yapılan benzetim çalışmaları sonucunda üçlü pozisyonlamanın kalitesinde önemli oranda bir artış sağlanmıştır. Üçlü pozisyonlama ile ilgili yapılan bu çalışma derlenip alanında bilinen bir uluslararası çalışmaya bildiri şeklinde gönderilmiştir ve Haziran 2013'de sunumu yapılmıştır. "*Reducing the number of flips in trilateration with noisy range measurements*" isimli bildirinin bir kopyası ekte sunulmuştur. Bildirinin genişletilmiş özeti aşağıda verilmiştir.

Başlık: Üçlü pozisyonlamada uzaklık ölçümlerinin hatalı olduğu durumlarda düzlemsel evrim sayısını azaltmak

Kablosuz ağlar üzerinde çalışan pekçok uygulama doğruluk derecesi yüksek konum belirleme servislerine ihtiyaç duyar. Üçlü pozisyonlama yaygın olarak kullanılan uzaklık ölçümü tabanlı bir konumlama algoritmasıdır. Üçlü pozisyonlama uzaklık ölçümlerinin hatasız olduğu durumlarda polinom zamanda çalışmaktadır, fakat gerçek hayat koşullarında ölçüm hataları her zaman mevcuttur ve hataların olduğu durumlarda oluşan düzlemsel evrimlerden kaynaklanan belirsizlikler üçlü pozisyonlama ile konum tespitini karmaşık hale getirmektedir. Bu çalışmamızda biz düzlemsel evrimlerden kaynaklanan belirsizliklerin nedenlerini analiz edip, pozisyonlama sırasında oluşacak düzlemsel evrimleri, büyük miktarda ölçüm hatalarının olduğu durumlarda bile, en aza indiren yeni bir yöntem öneriyoruz. Önerdiğimiz yöntemin detaylı anlatımı ekte verilmiş olup, geniş bir özetini ise burada aktarıyoruz.

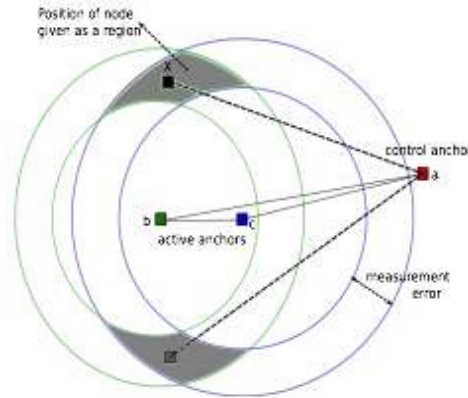
Düzlemsel evrim sayısını azaltmak için geliştirdiğimiz yöntemi anlatmadan önce üçlü pozisyonlama algoritmasının nasıl çalıştığını tekrar hatırlamakta fayda vardır. Üçlü pozisyonlamada bir noktanın pozisyonu konumu daha önceden bilinen diğer üç noktaya (çapa noktalar) olan uzaklığı kullanılarak hesaplanır. Temel olarak burada her uzaklık

ölçümü pozisyonu bilinen nokta etrafında yarıçapı bu uzaklık olan bir çember tanımlar ve oluşan bu üç çemberin kesiştiği koordinatlar ise aranan noktanın iki boyuttaki pozisyonunu verir. Aşağıda verilen şekilden ölçüm hatalarının olmadığı durumda üçlü pozisyonlamanın anlatıldığı şekilde çalışması görülebilir.



Şekil 4.1.1 Üçlü pozisyonlama

Şekil 4.1.1'den de görüldüğü üzere önce pozisyonu bilinen iki nokta üzerinden çizilen çemberlerin kesişmesinden iki alternatif pozisyon hesaplanır. Sonradan bu pozisyonun üçüncü pozisyonu bilinen noktaya olan uzaklığı ile ölçülen uzaklık karşılaştırılarak bu alternatiflerden biri elenir ve aranan noktanın gerçek pozisyonu bulunur.



Şekil 4.1.2 Hatalı uzaklık ölçümleri ile üçlü pozisyonlama

Uzaklık ölçümlerinin hatalı olduğu durumda üçlü pozisyonlama algoritmasının çalışma şekli ise Şekil 4.1.2'de verilmiştir. Bu durumda uzaklık ölçümleri kesin bir çember vermek yerine simit şeklinde bir aralık verecek ve bu aralıkların kesişiminden oluşan alan ise aranan noktanın olası pozisyonunu belirleyecektir. Burada üzerinde durulması gereken önemli konu ise bu aralıkların geniş olduğu durumlarda uzaklık ölçümlerindeki hatadan dolayı üçüncü

pozisyonu bilinen nokta ile alternatif pozisyonlardan birini elemek için uzaklık ölçümü yapıldığında yanlış alternatifin seçilme olasılığıdır. Bu durum kısaca düzlemsel evrimlerden kaynaklanan belirsizlik olarak adlandırılmıştır. Bu durumda oluşan konumlama hatasını genel olarak Şekil 4.2.2'den de görüldüğü gibi büyük olmaktadır.

Düzlemsel evrimlerden kaynaklanan belirsizliklerin diğer bir ciddi sonucu ise üçlü pozisyonlamanın karmaşık hale gelmesine sebep olmasıdır. Bu konudaki karmaşıklık analizi ve ispatı tarafımızdan daha önceden yapılmıştır Evrendilek vd. (2011). Bu ispat incelendiğinde karmaşıklığın düzlemsel evrimlerden kaynaklandığı anlaşılmaktadır. Bu gözlemden yola çıkarak bu çalışmada düzlemsel evrim sayılarını azaltacak sezgisel algoritmalar araştırılmıştır.

Sezgisel algoritmalara geçmeden önce düzlemsel evrim durumunun formal olarak tanımlanmasında fayda vardır.

Tanım 4.1.1: Pozisyonu belirlenmek istenen nokta i ve pozisyonları önceden bilinen noktalar n_1 , n_2 ve n_3 verildiğinde i noktasının gerçek pozisyonunu P_i ve üçlü pozisyonlama sonrasında hesaplanan pozisyonunu T_i olarak tanımlayalım. Eğer P_i ve T_i noktaları pozisyonu bilinen bu üç noktanın ikişer ikişer oluşturduğu üç doğru parçasının her zaman aynı tarafında değilse i noktasının pozisyonu düzlemsel olarak evrilmiş olarak tanımlanır.

Düzlemsel evrimlere sebep olan durumlar araştırıldığında iyi seçilmiş noktalarla yapılan üçlü pozisyonlamalarda düzlemsel evrim sayısının azalabileceği gözlemlenmiştir. Bu gözlemden yola çıkarak düzlemsel evrim sayısını azaltmak için noktanın pozisyonu belirlenirken kullanılacak diğer üç noktanın seçiminde aşağıda detayları verilen kriterler getirilmiştir. Kablosuz algılayıcı ağlarda, algılayıcı elemanların yoğun bir şekilde kullanıldığı uygulamalarda her bir elemanın pozisyonlama için kullanabileceği pekçok alternatif komşusu olacağından, bu olasılıklar arasından düzlemsel evrim sayısını en aza indirecek olan üç noktayı seçmek yöntemiyle elemanların pozisyonunda oluşabilecek düzlemsel evrim hataları olasılığı en aza indirilebilecektir.

Bu durumda kullanılacak güvenli üçgen kriteri şu şekilde tanımlanabilir:

Tanım 4.1.2: Güvenli üçgen ST, noktalarından herbirinin diğer iki noktanın oluşturduğu doğruya olan uzaklığı en az daha önceden tanımlı bir sabit γ olan üç noktanın oluşturduğu üçgen olarak tanımlanabilir. Bu tanımlamada kullanılan γ değeri ise bu üçgenin güvenlik aralığı olarak adlandırılır.

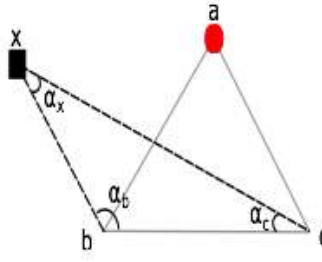
Güvenli üçgen oluşturulması sırasında kullanılan güvenlik aralığı ortamdaki gürültü ve hata miktarına bağlı olarak seçilmelidir. Deneylerimiz sırasında bu parametreyi iki farklı şekilde kullandık. Bunlardan ilkinde bu değeri sabit bir değer olarak seçerken diğer yöntemde ise üçlü pozisyonlamanın ilerleyen evrelerinde oluşan hataların birikerek arttığını gözönüne alarak bu değeri üçlü pozisyonlama sırasına bağlı olarak arttırdık. Burada kullanılan üçlü pozisyonlama sırası şu şekilde tanımlanabilir.

Tanım 4.1.3: Üçlü pozisyonlama sırası üçlü pozisyonlama grafında herhangi bir elemanın üçlü pozisyonlamayı başlatan ilk üç pozisyonu bilinen elemandan en yakın olanına olan sekme sayısı cinsinden uzaklık olarak tanımlanır.

Bu çalışmamızda pozisyon hesaplarırken güvenli üçgen kriterine uymayan üçlü elemanları kullanmamayı öngördük. Bunun doğal sonucu olarak da, özellikle ölçüm hatalarının çok olduğu ortamlarda, pozisyon hesabı sırasında üçlü pozisyonlama yapabileceğimiz üçlü nokta sayısında önemli bir azalma yaşandı ve sonuç olarak da belirli noktaların yerlerini hesaplayamadık. Algoritmamızı bu şekilde tasarlamamızın amacı düzlemsel evrim yapma olasılığı yüksek güvensiz üçgenler kullanıp oldukça hatalı sonuçlar bulmaktansa, o noktaların yerlerini hiç hesaplamayıp, böylece üçlü pozisyonlamanın ileri safalarında da diğer hesaplara bu hataların yayılmasını önlemeyi tercih etmemizdendir. Böylece bu çalışma sonucunda bütün noktaların yerlerini yüksek hatalı olarak bulmaktansa, daha az sayıda noktanın yerini yüksek doğrulukla bulmak pekçok uygulama için daha uygun olduğu sonucuna varılmıştır.

Güvenli üçgen her ne kadar düzlemsel evrimlerden kaynaklanan belirsizlikleri azaltsa da, yeri tespit edilmek istenen noktanın aktif çapa noktalarla neredeyse aynı doğru üzerinde olduğu durumlarda tam olarak yeterli sonucu vermemektedir. Bu tarz durumları ele almak için ise ikinci bir kriter olarak güvenli çapa noktaları testini öneriyoruz. Bu test aşağıdaki gibi tanımlanabilir:

Tanım 4.1.4: Güvenli çapa noktaları a , b , ve c 'den oluşan çapa noktalarının kümesi olarak tanımlanabilir, öyleki pozisyonu hesaplanmak istenen x noktasının bu çapa noktaların ikişer ikişer üzerinden geçen her üç doğruya olan uzaklığı güvenlik aralığından daha fazla olmalıdır.



Şekil 4.1.3 Güvenli çapa noktaları

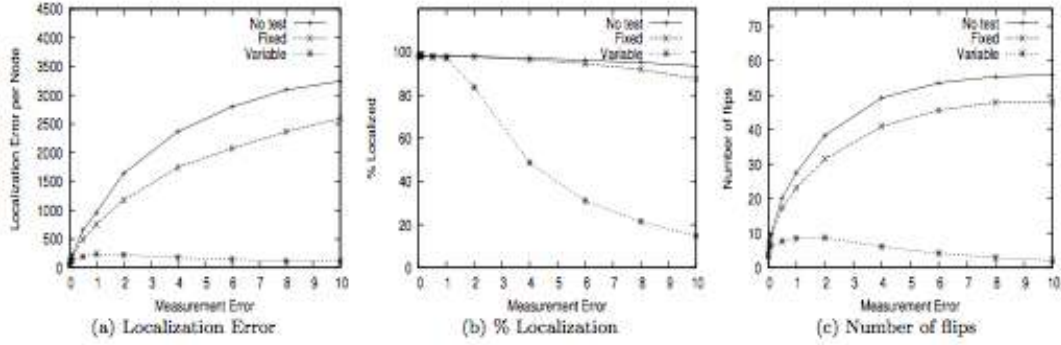
Şekil 4.1.3 üzerinden de görüldüğü gibi eğer x noktasının b ve c çapa noktaları üzerinden geçen doğruya olan uzaklığı güvenlik aralığından az olursa x noktasının pozisyonunun b ve c noktalarından geçen doğru etrafında düzlemsel evrim yaratma ihtimali vardır. Bu ihtimali ortadan kaldırmak için güvenli çapa noktalarının seçimi önemlidir. Buradaki temel sorun, halihazırda x noktasının yerini bilmediğimizden, güvenli çapa noktalarını hesaplamak için x noktasının pozisyonunu kullanma şansımızın olmamasıdır. Bu durumda Tanım 4.1.4'ü değiştirerek uzaklıklar yerine x , b , ve c noktalarının oluşturduğu üçgenin iç açılarını kullanarak benzer bir işlem tanımlayabiliriz. Bu durumda tanımımız, b ve c noktalarının güvenli çapa noktaları olabilmesi için b , c , ve x noktalarının oluşturduğu üçgenin iç açılarının her birinin α olarak daha önceden tanımlanan bir değerden daha büyük olması olarak yeniden verilebilir. Bu durumda üçgenin tüm kenarlarının uzunlukları uzaklık ölçümleri vasıtasıyla bilindiğinden, bu kenar uzunluklarından iç açılar kolaylıkla hesaplanabilmektedir. Deney çalışmalarımız sonucunda güvenli çapa testlerinin düzlemsel evrimlerden kaynaklanan belirsizliklerin azaltılmasında önemli etkileri olduğunu ortaya çıkmıştır.

Güvenli üçgen ve güvenli çapa testlerine ek olarak üçgen iç açıları testi olarak adlandırdığımız bir test daha tanımladık. Her ne kadar güvenli üçgen testi bu testin özel bir hali olsa da, deneylerde bu testin de belirli durumlar için düzlemsel evrim sayısını önemli oranda azalttığını tespit ettik. Üçgen iç açıları testi temel olarak oluşan güvenli üçgenin iç açılarının belirli bir değerinden yüksek olma zorunluluğu olarak tanımlanabilir. Bu testin sonucu olarak daha dengeli güvenli üçgenlerin oluştuğu deneyler sonucunda belirlenmiştir.

Deneyler

Çalışmamızın deneysel kısmında öncelikli olarak güvenli üçgenlerin düzlemsel evrim sayısını azaltmaktaki başarısı test edilmiştir. Temel olarak yapılan testlerde güvenli üçgen seçimi sırasında güvenlik aralığı kullanılmadığında, sabit bir güvenlik aralığı ve zamanla artan bir güvenlik aralığı kullanıldığında bu durumun konum tespit algoritmasının toplam konumlama

hatasına ve nihai olarak oluşan düzlemsel evrim sayısına olan etkileri araştırılmıştır. Benzetim ortamında 100 birime 100 birimlik bir alan üzerine 100 adet kablosuz eleman rastgele olarak atılmış ve her birinin kablosuz erişim uzaklığı 40 birim olarak ayarlanmıştır. Her bir deney 1000 kere tekrarlanıp ortalama değerler raporlanmıştır.

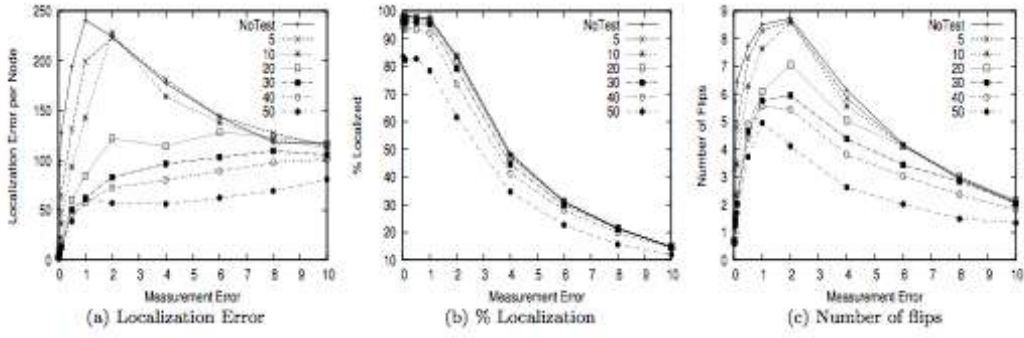


Şekil 4.1.4 Güvenli üçgen testleri

Şekil 4.1.4'den de görüleceği gibi değişken güvenlik aralığı kullanıldığında oluşan konumlama hatası ve düzlemsel evrim sayısının oldukça azaldığı gözlenmiştir. Burada dikkati çeken bir konu hataların çok arttığı durumlarda değişken güvenlik aralığı güvenli üçgen bulmakta zorlandığından konumlama yapılan eleman sayısında oluşan azalmadır. Burada daha az sayıda eleman konumlanmasına rağmen oluşan konumlama hatasının azlığından bu elemanların oldukça doğru bir şekilde konumlandığı anlaşılmaktadır. Şekil 4.1.4'den yapılabilecek diğer bir gözlem ise ölçüm hatalarının 2 cm (birim) olduğu durumlarda değişken güvenlik aralığı kullanıldığında hem düzlemsel evrim sayısı ve hataların oldukça azaldığı hem de elemanların %85'den fazlasının yine de konumlandığı görülmektedir. Ultrawideband radyolar kullanıldığında oluşan ölçüm hatalarının yaklaşık 2 cm civarında olduğu düşünüldüğünde, bu hata aralıklarında değişken güvenlik aralığının başarılı bir sonuç verdiği anlaşılabilir.

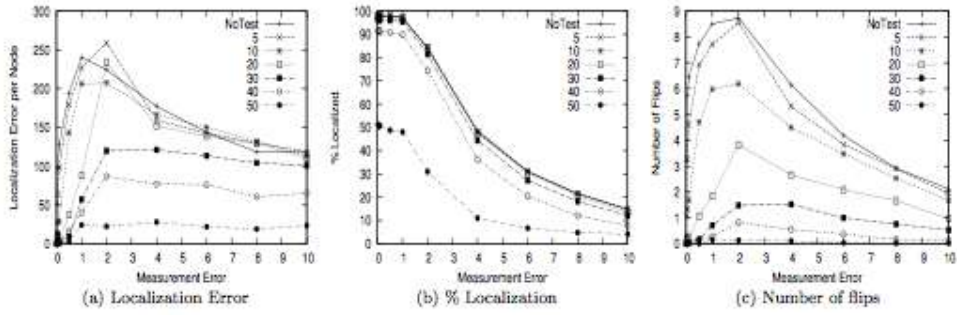
Güvenli çapa ve üçgen iç açıları testleri

Güvenli çapa ve üçgen iç açıları test deneylerini üç aşamalı olarak gerçekleştirdik. Öncelikle üçgen iç açıları testi için değişik değerleri kullanarak deneyler tasarladık. Bu deneylerin sonuçları Şekil-5'de görülebilir. Bu sonuçlardan yola çıkarak değerini 30 derece olarak belirledik.

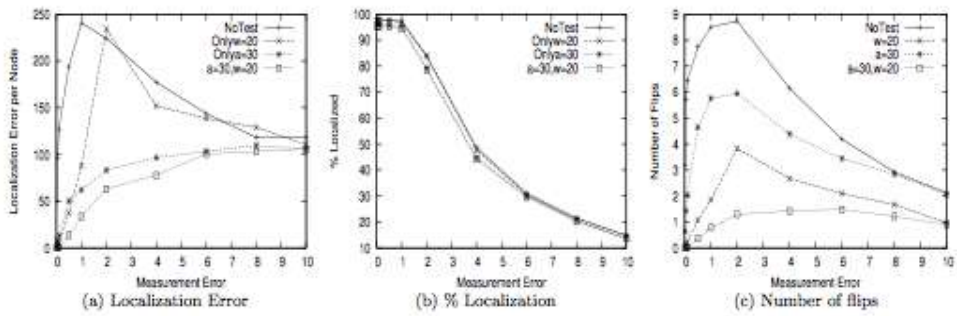


Şekil 4.1.5 Üçgen iç açıları testleri

Güvenli çapa testleri için de yine benzer bir şekilde farklı α değerleri için testler tasarladık ve Şekil 4.1.6'da sonuçları görünen bu testler sonucunda α değerini 20 derece olarak belirledik.



Şekil 4.1.6 Güvenli çapa testleri



Şekil 4.1.7 Tüm parametre testleri

Son olarak bütün geliştirdiğimiz güvenli üçgen, güvenli çapa ve üçgen iç açıları testleri beraber kullanıldığında sonuca olan etkilerini gözlemlemek için testler gerçekleştirdik. Bu testlerin sonuçları ise Şekil 4.1.7'de görülebilir. Bu sonuçlara bakarak özetlersek değişken güvenlik aralığı olan güvenli üçgen kullanımının hem konumlama hatasını hem de düzlemsel

evrim sayısını önemli ölçüde azalttığı gözlemlenmiştir. Bu teste ek olarak yapılan üçgen iç açıları testi konumlama hatasını daha fazla oranda azaltmaya yardımcı olurken, güvenli çapa testlerinin ise oluşan düzlemsel evrim sayısını daha da fazla oranda azaltmakta önemli katkıları olduğu gözlemlenmiştir. Bu durumda geliştirilen bu testlerin tamamının kullanımı kaliteli bir konumlama yapılmasında fayda sağlayacaktır. Yine Şekil 4.1.7' den görülebilen diğer bir sonuç ise, pratikte Ultrawideband radyolar tarafından raporlanan 2 cm civarı ölçüm hatalarının olduğu durumlarda bu üç testin beraber kullanılması sonucu algoritmalarımız sifıra yakın düzlemsel evrim yapmaktadırlar. Bu da geliştirdiğimiz testlerin pratikte fayda sağladığını göstermektedir.

4.2 Aynı düzlemdeki noktaların tespiti ve düzlemler arası konumlama

Bu proje çalışması, 3B'de kablosuz sensör ağları (KSA) konumlama problemini, eşdüzlemselliğe sebep olan ortamlarda inceliyor. Quadrilateration olarak bilinen, dört tane eşdüzlemsel olmayan noktayla bir beşinciye konumlama yönteminin kendi içinde kısıtları mevcut. Hali hazırda konumlanmış birçok dördümlü düğüm grupları, eşdüzlemsel oldukları için etkisiz hale geliyorlar. Ancak, bu grup bilgisini elde etmek bile kendi başına NP-Hard bir problem. Biz ise, bu bilginin geldiğini varsayarak, bilginin elde edilmesinin hangi durumlarda işe yarayacağını inceliyoruz. Bu gözlemi yapabilmek için, 3B'de uygulanabilir bir algoritma da geliştirdik. Algoritmamız, Coplanarity Based Localization (CBL), Türkçe adıyla Eşdüzlemsel Tabanlı Konumlama (ETK), eğer varsa, eşdüzlemsel düğüm grupları arasında rijit bir yapı oluşturmayı hedefliyor. Böylece, tüm sensör düğümlerini konumlamak yerine, problemi, kolay olduğu öngörülen bir safhaya, düğümlerin üzerinde buldukları düzlemleri konumlamaya dönüştürüyor. DTK, grup bilgisi olduğu zaman, deneylerimizde güçlü bir performans sergilemiştir.

4.2.1 Giriş

Kablosuz sensör ağlarında uzaklık tabanlı konumlama, sensör düğümlerinin göreceli pozisyonlarını, bazı sensör çiftlerinin arasındaki uzaklıklar yardımıyla bulma işlemine verilen isimdir. Özellikle GPS türü cihazların kullanımının kısıtlı olduğu kapalı mekanlarda önemi artar. GPS türü cihazlar kapalı mekanlarda iyi sonuç vermediği için, kapalı mekanlarda kullanılacak çeşitli konumlama teknikleri geliştirilmiştir (Akcan vd. 2006; ; Aspnes vd. 2010; Mao vd. 2007; Houry vd. 2009). En çok bilinen yaklaşım olan kablosuz sensör ağlarında konumsal konumlamadır(position-localzation) ve NP-Hard olduğu kanıtlanmıştır. Ayrıca, sensör düğümlerinin sayısı arttıkça, enerji tüketiminin en aza indirilmesi de aşılması gereken önemli bir sorundur. Bu yüzden, sensör düğümlerini sadece konumlamak değil, bunun yanında daha az enerji kullanmak da gerekir. Heinzelman vd. (2000), toplam enerji

tüketimini yarıya indirecek bir protokol sunmuşlardır. Eren vd. (2004), göreceli konumlamanın, trilateration kullanılarak polinom zamanda yapılacağını göstermişlerdir. Trilateration, gürültülü ölçümler ile intractable olmasına rağmen Evrendilek vd. (2011), kapalı mekanlarda göreceli konumlama için en çok tercih edilen yöntemdir. Akcan vd. (2012,2013) yaptıkları çalışmada, yönelimli konumlama(directional-localization) problemine yeni bir yaklaşım getirmişlerdir. Biz ise, bu çalışma kapsamında üç boyutlu kapalı ortamlardaki sabit sensörler arasındaki konumlama konusunu irdeliyoruz.

Kapalı ortamların birçoğu, eşdüzlemselliğe sebep olur. Bu tip ortamlara *düzlemsel ortam* ismini veriyoruz. Nasıl ki 2B konumlama algoritmaları doğrudan noktaları kullanamıyorsa, 3B konumlama algoritmaları da eşdüzlemsel noktaları kullanamaz. İki boyutlu ortamlarda doğrudanlık çok sık gözlemlenen bir durum değildir. Ancak, üç boyutlu ortamlarda eşdüzlemsellik çok sık gözlenir. 2B konumlama için kullanılan teknikler, 3B’de kullanıma açık olsa dahi, düzlemsel ortamlar bu tekniklerin etkin biçimde kullanılmasını engeller. Bu yüzden, 3B’de çalışan bir algoritma hem doğrudanlığı, hem de eşdüzlemselliği kontrol etmek durumundadır ve uzaklık tabanlı konumlamanın zorluğu artar. Bu çalışmada, “grup bilgisini elde etmek yararlı olur mu?” sorusunu cevaplıyoruz. Bu tip ortamlardaki sensör düğümleri, sayıca daha az olan ve düzlemler üzerinde konuşlanmış sensör grupları olarak temsil edilebilir.

Bu gruplara *eşdüzlemsel grup* ismini veriyoruz. Quadrilateration, dört tane eşdüzlemsel olmayan noktayı kullanarak bir beşinci konumlar. Bundan dolayı, bir eşdüzlemsel grup Quadrilateration aracılığıyla etkin olarak işlenemez Mao vd. (2007) ve Quadrilateration kullanabileceği düğüm dörtlülerini ararken hatırı sayılır bir zaman harcar. Biz, eşdüzlemsel sensör gruplarını birer öge olarak kabul ediyoruz ve problemin boyutunu sensör sayısı olan n değerinden, düzlem sayısı olan k değerine indiriyoruz.

Başlangıçta sensör düğümlerinin konuşlandıkları noktaların izlediği desenin sonuç getireceğini bilsek dahi, sensör grafını eşdüzlemsel düğümlere ayırmak da konumlamanın içinde bulunan ve çok önemli bir aşamasıdır. Ancak, bu çalışmanın ana odağı, bu grup bilgisinin elde edilmesinin, konumlama kalitesini arttırdığını deneysel olarak kanıtlamaktır.

Düğümleri tek başına konulamak yerine, Düzlemsel Tabanlı Konumlama, girdinin bir parçası olarak kabul edilen düzlemler arasında rijit bir yapı oluşturmayı hedefliyor.

Bu proje ile söz konusu çalışma alanına yaptığımız ana katkılar şu şekildedir:

- Düzlemsel ortamlarda grup bilgisinin etkisini inceledik.
- Grup bilgisinin var olduğu durumlar için bir algoritma geliştirdik.

- Deneysel olarak, bu algoritmanın 3B'de flip sayısını azalttığını doğruladık.
- Değişik özellikteki ortamlar göz önünde bulundurulduğunda, grup bilgisini elde etmenin gerekli ve gereksiz olduğu durumları inceledik.
- *Saptanmamış Alt Boyuttaki Yapılar ve Alt Boyuttaki Yapıları Konumlama* isimlerini verdiğimiz iki problem tanımladık.

4.2.2 Kablosuz Sensör Ağları (KSA) Konumlama

Konumlama, 2 veya 3 boyuttaki bir nokta kümesinin yerlerini bulma işlemine verilen isimdir. Bu iş için kullanılan en yaygın yöntem Küresel Konumlama Sistemi (GPS) ismi verilen yöntemdir. Ancak, GPS cihazları her zaman düzgün biçimde kullanılamaz. Bu sebeple, sensör düğümlerini konumlamak GPS yardımı olmadan konumlamak için çeşitli yöntemler geliştirilmiştir. Bu yöntemler, bazı sensör düğüm çiftleri arasındaki uzaklıkları baz alarak çalışır ve *uzaklık tabanlı konumlama* ismiyle anılırlar.

Bir kablosuz sensör ağı (KSA), bir düğüm kümesi $N = \{1, 2, 3, \dots, n\}$ ve bu küme içinde bazı çiftler arasında verilen uzaklıklardan oluşur. Aynı zamanda, N kümesinin bir alt kümesi olan $S = \{s_1, s_2, s_3, \dots, s_m\}$ kümesindeki her elemanın yeri bilinir. N kümesine *sensörler kümesi*, S kümesine ise *tohum sensörler kümesi* adı verilir. S kümesi içindeki her elemana *tohum düğüm* denir.

Tanım 4.2.1 (Tohum düğüm) *Tohum düğüm, bir kablosuz sensör ağında, koordinatları girdinin bir parçası olarak verilen düğümdür.*

Konumlamaya, tohum düğümleri kullanarak başlarız. Konumladığımız her düğüm, ağ içerisindeki konumlanmamış bir başka düğümü konumlamak için kullanılabilir. Birtakım hesaplamalardan sonra koordinatları bilinen düğümlere *işaretçi düğüm* denir.

Tanım 4.2.2 (İşaretçi düğüm) *Koordinatları girdinin bir parçası olarak verilmeyip sonradan hesaplanan ve henüz konumlanmamış bir başka düğümün konumlanmasında kullanılan düğüme işaretçi düğüm adı verilir.*

Tohum düğümler veya işaretçi düğümler kullanılarak o anda konumlanmaya çalışılan düğüme *to be localized sensor* (TBL), Türkçe adıyla *konumlanmakta olan düğüm* (KOD) adı verilir.

Tanım 4.2.3 (Konumlanmakta olan düğüm (KOD)) *Konumlanmakta olan düğüm, tüm ağın konumlanma sürecindeki herhangi bir anda, bir grup tohum düğüm veya işaretçi düğüm aracılığıyla koordinatları hesaplanmakta olan düğüme verilen isimdir.*

Tanım 4.2.4 (KSA konumlama) *Bir sensörler kümesi ve bir tohum sensörler kümesi verildiğinde, konumu bilinmeyen düğümleri, konumu bilinen düğümlere ve ağ içindeki bazı sensör çiftleri arasındaki uzaklıklara göre konumlama işlemine KSA konumlama denir.*

KSA'nın konumlanabilirliği

2B KSA'nın tümüyle konumlanabilmesi için gerek ve yeter şartlar Aspnes vd. (2010). tarafından belirlenmiştir. Ancak, bu şartlar daha yüksek boyutlara uygulanamaz. İki boyutta bir KSA'nın konumlanabilirliğini tespit etmek için kullanılan iki önemli konsept vardır. İsmen, *nokta dizilişi* ve *rijidite*.

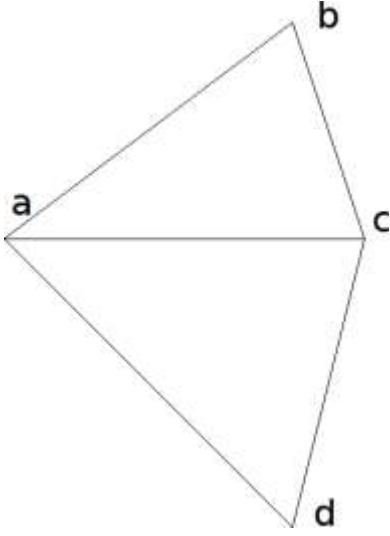
Nokta Dizilişi

\mathbb{F}_p , *nokta dizilişi*ni simgeler ve $\mathbb{F}_p = (\{p_1, p_2, \dots, p_n\}, \mathcal{L})$ şeklinde gösterilir. Her nokta dizilişi, $\mathbb{G}_{\mathbb{F}_p} = \{V, \mathcal{L}\}$ şeklinde gösterilen özgün bir graf belirler. Bu grafta V , düğüm kümesi(vertex set), \mathcal{L} ise kenar kümesi(edge set)dir. Graf, aynı zamanda $\delta : \mathcal{L} \rightarrow \mathbb{R}$ şeklinde ifade edilen bir fonksiyon içerir. δ fonksiyonu, p_i ve p_j isimli iki nokta arasındaki uzaklığı belirtir ve bu uzaklık (i, j) olarak gösterilir.

Rijidite

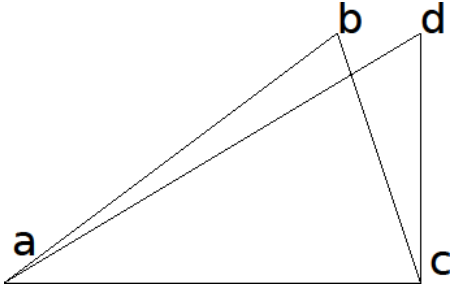
\mathcal{L} kümesindeki her (i, j) kenarı için, $(p_i - p_j)^T (p_i - p_j) = \delta(i, j)^2$ uzaklık denklemi, doğrusal bir denklem üretir. Eğer $\mathbb{G}_{\mathbb{F}_p}$ grafı, δ fonksiyonuna göre tüm olası dizilişlerde aynı kalıyorsa, kenar uzunlukları da değişmez. Bu da, $\mathbb{G}_{\mathbb{F}_p}$ grafının rijit olduğunu gösterir.

Şekil 4.2.1'de, birinci dereceden rijit bir graf görüyoruz.



Şekil 4.2.1 Birinci dereceden rijit bir graf.

Şekil 4.2.2 ise, aynı grafın bir başka düzenleşimi(configuration).



Şekil 4.2.2 Şekil 4.2.1'de görülen birinci dereceden rijit grafın düğümlerinin bir başka dizilimi.

Uzaklık Ölçüm Hataları

Eğer bir sensör düğümü, kendisiyle bir başka düğüm arasındaki uzaklığı yanlış hesaplırsa *uzaklık ölçüm hatası* yapmış olur. Uzaklık ölçüm hataları, çevresel gürültü neticesinde meydana gelir. Her sensör ağı için bir hata marjı vardır ve konumlama esnasında hesaba katılmak durumundadır. Söz konusu marj, *hata oranı* şeklinde adlandırılır.

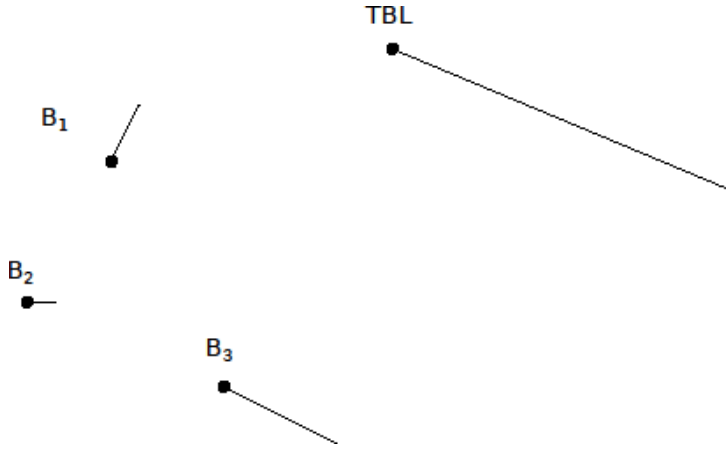
Tanım 4.2.5 (Hata oranı) *Bir KSA içindeki herhangi bir sensörün, kendisiyle bir başka sensör arasındaki ölçtüğü uzaklığın gerçek uzaklıktan olası sapma değerine hata oranı denir.*

Örnek Diyelim ki, i ve j isimli iki sensörün arasında 1 birim uzaklık var ve hata payı %5. Eğer bu sensörlerden herhangi birinden, diğerine olan uzaklığı ölçmesini istersek, bize vereceği cevap hata oranı yüzünden 0.95 ve 1.05 arasında bir değer olacaktır.

Bir KSA'yı gürültülü uzaklık ölçümleriyle konumlamak mümkün olsa bile, NP-Hard olduğu Evrendilek vd. (2011) tarafından kanıtlanmıştır. Hatalı ölçümlerin, konumlamaya iki büyük etkisi vardır. Bunlardan ilki konumsal sapma, bir diğeryse yansıma(flip)dır.

Konumsal Sapmalar

Konumsal sapma, bir düğümün asıl konumuyla hesaplanmış konumu arasındaki farka verilen isimdir. Gürültülü uzaklık ölçümlerinde, çok ufak hatalar birikir ve kayda değer konumsal sapmalara yol açar. Böylece konumlanmanın doğruluğu azalır. Hatalar biriktiğinde, ufacak bir hata KODların pozisyonlarında çok büyük değişikliklere sebep olur. Şekil 4.2.3'te, B_1, B_2, B_3 adı verilen üç işaretçi bulunmakta. Siyah noktalar, düğümlerin hesaplanmış konumlarını, noktaların ardındaki siyah çizgilerin diğer ucuysa düğümlerin asıl konumlarını göstermekte. İşaretçi düğümlerin pozisyonlarındaki nispeten ufak değişikliklerin, konumlanmakta olan düğümün (TBL) konumundaki fazlasıyla büyük değişimi görüyoruz.



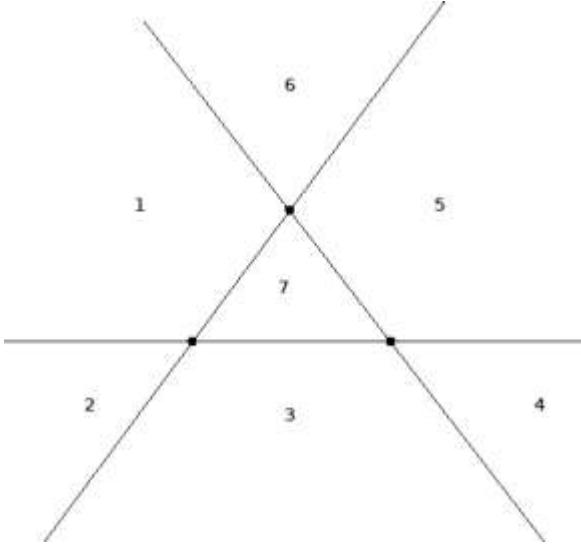
Şekil 4.2.3 Konumsal sapma.

Sapmalar, hatalı ölçümlerin tek sonucu değildir. Bunun yanında, düğümler yansıyabilir de.

Yansımalar

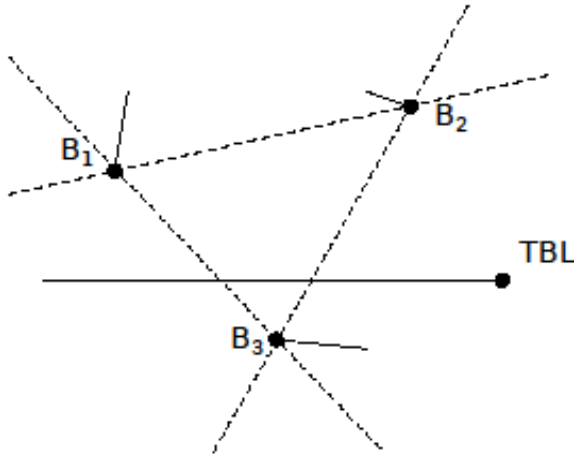
d boyuttaki bir düğümü konumlayabilmek için, konumu bilinen $d + 1$ tane düğüme ihtiyaç duyarız. $B = \{b_1, b_2, b_3, \dots, b_{d+1}\}$, KODu konumlamak için kullandığımız işaretçi düğüm kümesi olsun. T_o , KODun asıl konumu ve T_c de KODun hesaplanmış konumu olsun. Bir yansımayı saptayabilmek için, öncelikle, $d + 1$ tane işaretçi düğümün $d + 1$ 'li kombinasyonlarını kullanarak birer alt boyuttaki yapı oluştururuz. (Alt boyuttaki yapı, bulunduğu boyutun bir eksiği boyutu temsil eden yapıdır. Örneğin, bir küp içindeki düzlem veya bir düzlem üzerindeki doğru.) Bu yapılar ortamı 2^{d+1} bölgeye ayırır. T_o , bu bölgelerin birinin içindedir. Eğer T_o , r_o bölgesinde ve T_c de r_c bölgesinde dersek, ancak ve ancak $r_o \neq r_c$ olduğu durumlarda düğüm yansımış diyebiliriz.

Şekil 4.2.4, iki boyutta bir bölge ayırmasını gösteriyor.



Şekil 4.2.4 İki boyutta bir düğümün bulunabileceği yedi farklı bölge.

Şekil 4.2.5, yansımış bir KODu gösteriyor.



Şekil 4.2.5 Bir KODun(TBL) yansıması.

İki çeşit yansıma vardır. İsmen, *tasnif edilmiş yansıma* ve *tasnif edilmemiş yansıma*.

Tanım 4.2.6 (Tasnif edilmiş yansıma) Konumlama esnasında herhangi bir KODun koordinatları için, işaretçi düğümlere göre birden fazla çözüm söz konusuysa, bu duruma *tasnif edilmiş yansıma* ismini veriyoruz.

Tasnif edilmiş yansımaları, basitçe, yansıyan düğümü konumlanmadı olarak işaretleyip kaldığımız yerden devam ederek engelleyebiliriz.

Tanım 4.2.7 (Tasnif edilmemiş yansıma) *Konumlama esnasında herhangi bir KODun koordinatları için, işaretçi düğümlere göre sadece bir çözüm varsa ve bu çözüm bir yansımayla sonuçlanıyorsa, bu duruma tasnif edilmemiş yansıma ismini veriyoruz.*

Normalde, tek çözüm olduğundan tasnif edilmemiş yansımalar saptanamaz. Ancak deneylerimizde, her KODu konumladıktan sonra kontrol ettik ve tasnif edilmemiş yansımaları da böylece engelledik.

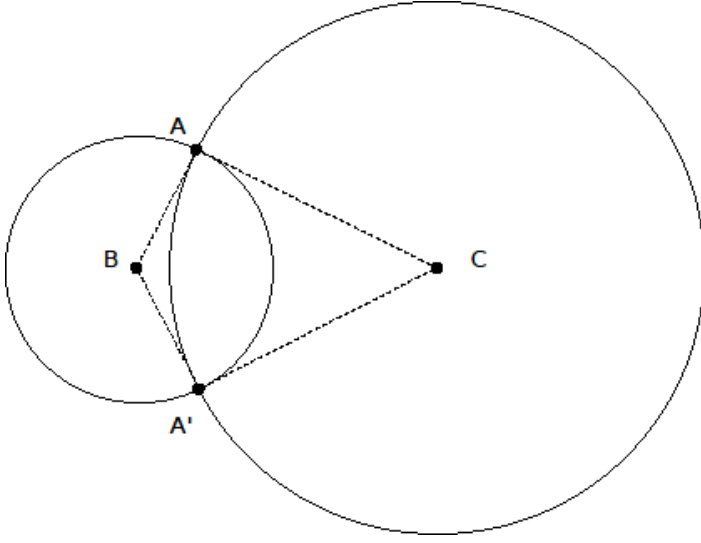
4.2.3 2B KSA Konumlama

2B KSA konumlama problemi, düğümlerin iki boyuttaki konumlarının ikili uzaklık ölçümleri yardımıyla bulunmasıdır. İki sensör arasındaki uzaklık ölçümü, noktaların bir serbestlik derecesini azaltır. 2B'deki bir düğümün üç serbestlik derecesi mevcuttur. Bu sebepten, bir sensör düğümünü konumlamak için, doğrudan olmayan ve konumları bilinen üç sensör düğümünden ayrı ayrı uzaklık ölçümlerine ihtiyaç duyarız. Bu problem, metin boyunca 2-KSAK olarak kullanılacaktır.

Trilateration

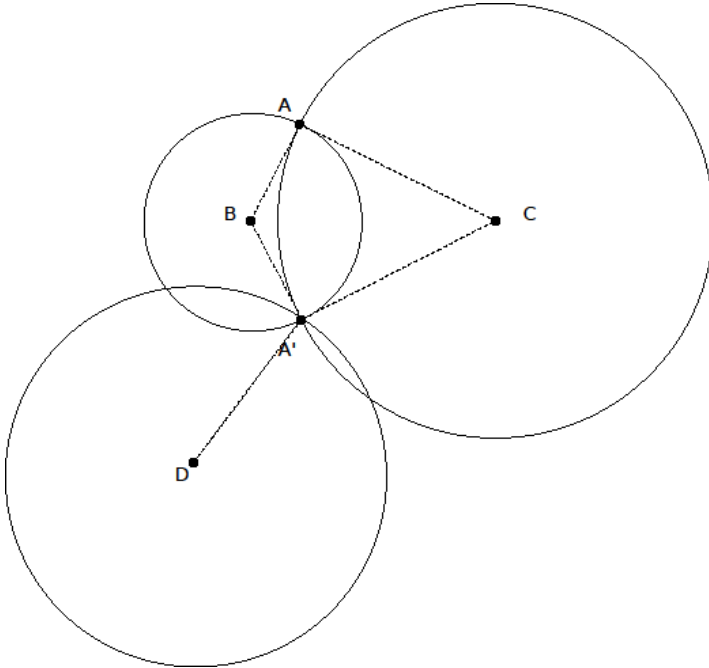
Trilateration, üç tane eşdoğrusal olmayan tohum veya işaretçi düğüm kullanılarak dördüncü bir düğümün konumunun elde edilmesi işlemidir. Trilateration esnasında, herhangi bir düğümü konumlamak için, eşdoğrusal olmayan ve konumları bilinen üç farklı düğüme olan uzaklık ölçümlerine ihtiyaç duyulur. Bu üç uzaklık ölçümünü kullanarak, üç tane çember oluştururuz. Çemberlerin yarıçapları, KODdan diğer üç düğüme olan uzaklıklar ve merkezleri ise işaretçi düğümlerin konumları olarak tanımlanır. İlk olarak, iki çemberin kesişim noktalarını hesaplarız. Bu iki nokta, KODun koordinatları için iki ayrı çözüm anlamına gelir. Çözümleri bire indirgemek için, üçüncü çembere ihtiyaç duyulur. Üçüncü çemberi sağlayan çözüm KODun koordinatları olarak kabul edilir.

Bu yöntem, uzaklık ölçüm hatası olmadığı zaman 100% doğru olarak çalışır. Ancak, Bölüm 4.2.2'de incelediği üzere, gürültülü uzaklık ölçümleri konumlama sürecindeki kaliteyi düşürür. Şekil 4.2.6'da, *B* ve *C* işaretçi düğümlerine göre, *A* düğümü için iki olası iki çözüm gösteriliyor. Bu durumda *A* noktasını mı yoksa *A'* noktasını mı seçeceğimizi bilemeyiz.



Şekil 4.2.6 *A* düğümünün iki uzaklık ölçümü yapılarak konumlanması

Şekil 4.2.7'de, *D* düğümünü de kullanıyoruz ve böylece olası çözümlerden birini seçebilecek konuma geliyoruz.



Şekil 4.2.7 *A* düğümünün üç uzaklık ölçümü yapılarak konumlanması

Bir KSA grafının tamamen Trilateration ile konumlanabilir olması için, gu grafın *trilateration sırası* olması gerekir.

2B KSA Konuqlamada Yansımalar

İki çember kesiştiğinde, ortaya iki tane kesişim noktası çıkar. KODun koordinatlarını belirleyebilmek için bu kesişim noktalarından birini seçmemiz gerekir. Seçimi yapmak için üçüncü çemberi kullanırız. Ancak bu seçim Bölüm 4.2.2'de tasvir edildiği üzere, bir yansımaya sebep olabilir.

Akcan vd. (2013) Trilateration kullanırken yansımaların sayısını azaltmak için bir yöntem önerdi. Ancak, bu yöntemin uygulanabilirliği sadece 2B ortamlar için deneylerle kanıtlandı. 3B ortamlar için herhangi bir etkisi olup olmadığı bilinmiyor.

4.2.4 3B KSA Konuqlama

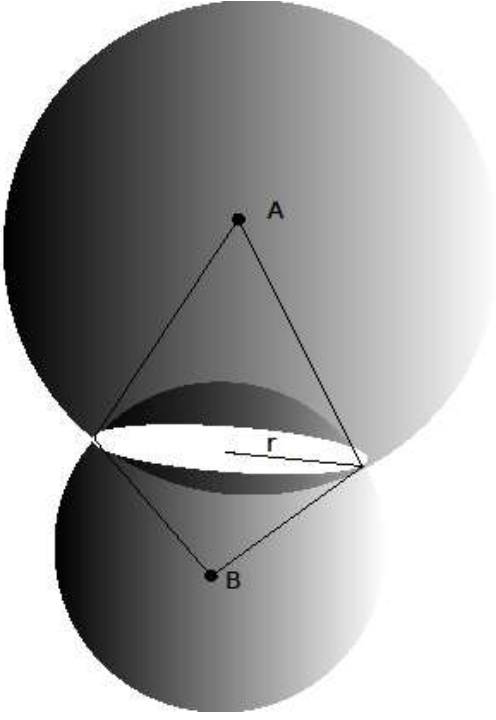
Girdi olarak verilen ikili uzaklıkları kullanarak üç boyutta yapılan KSA konuqlamasına 3B KSA Konuqlama adı verilir. 2B'den farklı olarak, fazladan bir koordinat — z koordinatı bulunur. z koordinatı serbestlik derecesini bir artırarak dörde çıkarır ve bu yüzden ki üç boyutta bir düğümü konuqlamak için dört tane işaretçi düğüme ihtiyaç duyarız.

Quadrilateration

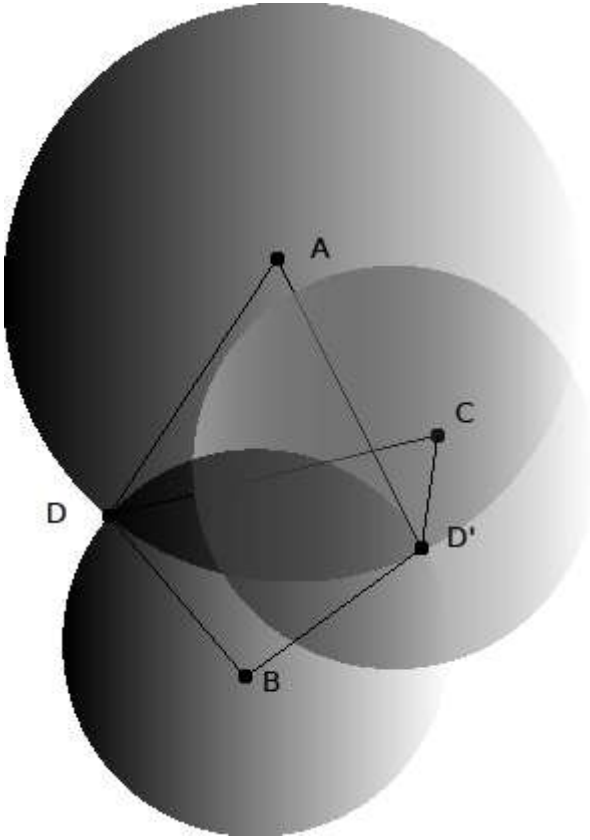
Quadrilateration, Trilateration'un üç boyuttaki eşleniğidir. Dört tane eşdüzlemsel olmayan nokta kullanarak bir beşincisi konuqlanır. Algoritma, genelde her dörtlüyü deneyerek yapabileceği en yüksek konuqlama yüzdesine ulaşmaya çalışır. Ancak biz deneylerimizde sadece tek adımda, en yüksek dereceli dörtlü düğüm gurubunu seçerek işlemimizi gerçekleştiriyoruz ve diğer dörtlüleri denemiyoruz.

Şekil 4.2.8'de, kesişen iki küre görüyoruz. *A* ve *B* işaretçi düğümler ve *D* düğümü KOD. Kürelerin yarıçapları, *A* ve *B* düğümlerinin *D* düğüme olan uzaklıkları, merkezleri ise *A* ve *B* düğümlerinin koordinatları. İki kürenin kesişimi üç boyutta bir çemberi tanımlıyor ve sonsuz sayıda olası çözüm elde ediyoruz.

Çözüm sayısını azaltmak için, bir başka küre kullanıyoruz. Şekil 4.9'da görüldüğü üzere, olası çözümler ikiye düşüyor. Ancak bu hala yeterli değil. KODun koordinatları için tek bir nokta seçebilmek ancak dördüncü kürenin devreye girmesiyle mümkün.



Şekil 4.2.8 Kesişen iki küre ve kesişim çemberi.



Şekil 4.2.9 Kesişen üç küre ve iki kesişim noktası.

3B KSA Konumlamada Yansımalar

2B'dekinden farklı olarak, 3B'deki ortam, toplam 15 farklı bölgeye ayrılır. Bununla beraber gelen fazladan bir serbestlik derecesi, 3B ortamlarda yansıma olasılığını arttırmakta.

4.2.5 2B KSA Konumlama Tekniklerinin 3B'ye Uygulanabilirliği

Gerek ve yeter koşulların uygulanamadığı gibi, 2B KSA konumlama teknikleri de tamamen 3B'ye uygulanamaz. 2B KSA probleminin 3B KSA probleminden farklı bir doğası vardır. En önemli farklılık, graflardaki kenarların yarattığı etkidir. 2B KSA grafindaki bir kenar, düğümün serbestlik derecesini ikiye indirirken, 3B KSA grafindaki bir kenar, serbestlik derecesini üçe indirir. Serbestlik derecesini sıfıra indirmek için dört tane düğüm kullandığımızdaysa, ölçüm hatalarının birikiminin artmasına sebep oluruz. Hataların birikmesi ise, daha fazla yansıma ve konumsal sapmaya yol açar. 100% doğru ölçümlerde dahi tasnif edilmemiş yansımalarla karşılaşabiliyoruz.

4.2.6 İlişkili Problemler

Bu bölümde, KSA konumlama problemiyle ilişkili problemleri tartışıyoruz.

Saptanmamış Alt Boyuttaki Yapılar Problemi

d boyutlu bir ortamda, Cayley-Menger determinantı sıfırdan farklı ve büyüklüğü d olan herhangi bir nokta kümesi, $d - 1$ boyutlu bir yapı oluşturur. d boyutta bir p_0 noktasını konumlamak için iki şart sağlanmalıdır. İlk şart, büyüklüğü d olan bir nokta kümesi $\{p_1, p_2, \dots, p_d\}$ ve bu kümenin her elemanından p_0 noktasına uzunluk ölçümlerinin $(p_0, p_i); i \in \{1, 2, \dots, n - 1\}$ olması ve ikinci şart ise, $\{p_1, p_2, \dots, p_d\}$ kümesinin Cayley-Menger determinantının sıfırdan farklı olması. Klasik yöntemi kullanırken, bu koşulları sağlayan bir nokta kümesi bulamazsak, tohum noktalar bulamamış oluruz ve konumlama işlemi başarısız olur. Ancak, p_0 noktasını d boyut yerine $d - 1$ boyutta konumlamaya çalışsaydık, bunu başarabilecektik. İşbu probleme Saptanmamış Alt Boyuttaki Yapılar problemi (SABY problemi) ismi verilir. 3 boyutta SABY problemini çözmek amacıyla öncelikle eşdüzlemsel noktaları buluyoruz ve daha sonra bu düzlemleri birbirlerine göre konumluyoruz. Akcan vd. (2011) aynı probleme iki boyutta da bir yaklaşım geliştirmiştir. İkili-radyo adı verdikleri yapıları kullanarak, eşdüzlemsel noktaları elle oluşturarak konumlama yüzdesinde hatırı sayılır bir artış sağlamışlardır. Connely (1991), Laman (2002) ve Eren vd. (2004) 2 boyutta bir sensör ağının konumlanabilmesi için gerek ve yeter şartları araştırdılar ve bir sonuç elde ettiler. Ancak, 3-B için, gerek koşulları bilmemize rağmen yeter koşullar hakkında hala bir bilginiz bulunmamaktadır.

Lemma 4.2.1 d boyuttaki bir konumlama problemini, SABY problemini tekrar tekrar çözerek $2B$ 'ye indirgeyebiliriz.

Kanıt P , Π_0 düzlemi üzerinde bulunan bir nokta kümesi olsun. Π_0 düzlemini, Σ_0 ismini verdiğimiz $3B$ bir ortamın içine yerleştirelim ve bu $3B$ ortama Π_1, Π_2, \dots şeklinde isimlendirilen başka düzlemleri de ekleyelim. Böylece, üzerindeki noktaların konumlarını bildiğimiz düzlemler elde etmiş oluyoruz. Bu sayede, sadece düzlemlerin göreceli konumlarını bularak Σ_0 içindeki tüm noktaların göreceli konumlarını bulabiliriz. Benzer olarak, içindeki tüm noktaların konumlarını bildiğimiz Σ_0 $3B$ yapısını, kendisi gibi $3B$ olan $\Sigma_1, \Sigma_2, \dots$ yapılarıyla beraber $4B$ 'deki başka bir ortama koyabiliriz. Bu işlem, d boyuttaki bir yapıya kadar sürebilir. Tam tersten düşünecek olursak, d boyuttaki bir konumlama problemini, her $2 < e < d$ boyutu için SABY problemini çözerek $2B$ 'ye indirgeyebiliriz.

Alt Boyuttaki Yapıları Konumlama Problemi

Daha önce de belirttiğimiz gibi, SABY problemini çözmek, noktaların konumlanmasını sağlamaz. Elde edilen alt boyutlu yapıların da birbirlerine göre konumlanmaları gerekir. $d - 1$ boyutlu yapıların d boyutta konumlanması problemine Alt Boyuttaki Yapıları Konumlama (ABYK) problemi denir.

Lemma 4.2.2 ABYK problemi NP-Hard bir problemdir.

Kanıt Bir doğru üzerindeki noktaları konumlama probleminin NP-Hard olduğunu biliyoruz. Nokta, boyutsuz (0 boyutlu) bir yapıdır. Doğru ise 1 boyutlu bir yapıdır. Bir doğru üzerindeki n tane nokta ve ikili uzaklıklar verildiğinde, 2^n tane olası çözüm vardır. Bu durum, bir noktanın (0B), bir doğru üzerinde (1B) iki serbestlik derecesi olmasından kaynaklanır. Aynı durum bir düzlem üzerindeki doğrular ve bir uzay içindeki düzlemler için geçerlidir. Genelleyecek olursak, $d - 1$ boyutlu bir yapı, d boyutlu bir ortamda 2 serbestlik derecesine sahiptir. Bir doğru üzerindeki noktaları konumlama problemi, birincil ABYK problemidir ve d boyuta genellenebilir.

4.2.7 Düzlemsel Ortamlarda Konumlama

Düzlemselliğe sebep olan ortamlara *düzlemsel ortamlar* adı verilir. Problemimiz, düzlemsel bir ortamda k tane düzleme düzenli olarak dağılmış olan n tane sensör düğümü ve düğümler arasındaki girdi olarak verilen bazı ikili uzaklıkları kullanarak konumlama yapmak. Geliştirdiğimiz algoritma, Eşdüzlemsel Tabanlı Konumlama (ETK), söz konusu düzlemleri $3B$ 'de konumlamaya yarar. Düzlemleri konumlamak, birbirlerine göre denklemlerini bulmak

anlamına gelir. Böylesi bir yaklaşım problemin boyutunu n düğümü konumlamaktan, çok daha küçük olan k düzlemi konumlamaya indirir.

Aynı düzlemde bulunan düğümlere *eşdüzlemsel düğüm grupları* adı verilir. İki farklı düzlem arasındaki bağlantıya *düzlemlerarası bağlantı*, diğer bağlantılara *düzlemiçi bağlantı* adı verilir.

Eşdüzlemsel grupların verildiğini varsayarsak, düzlemsel bir düzen, k tane düzlemsel oluşumların konumlanması n tane düğümün konumlanmasından daha kolay bir problemdir ve bu safha tamamlandığında eşdüzlemsel gruplar 2B konumlama teknikleri kullanılarak konumlanabilir.

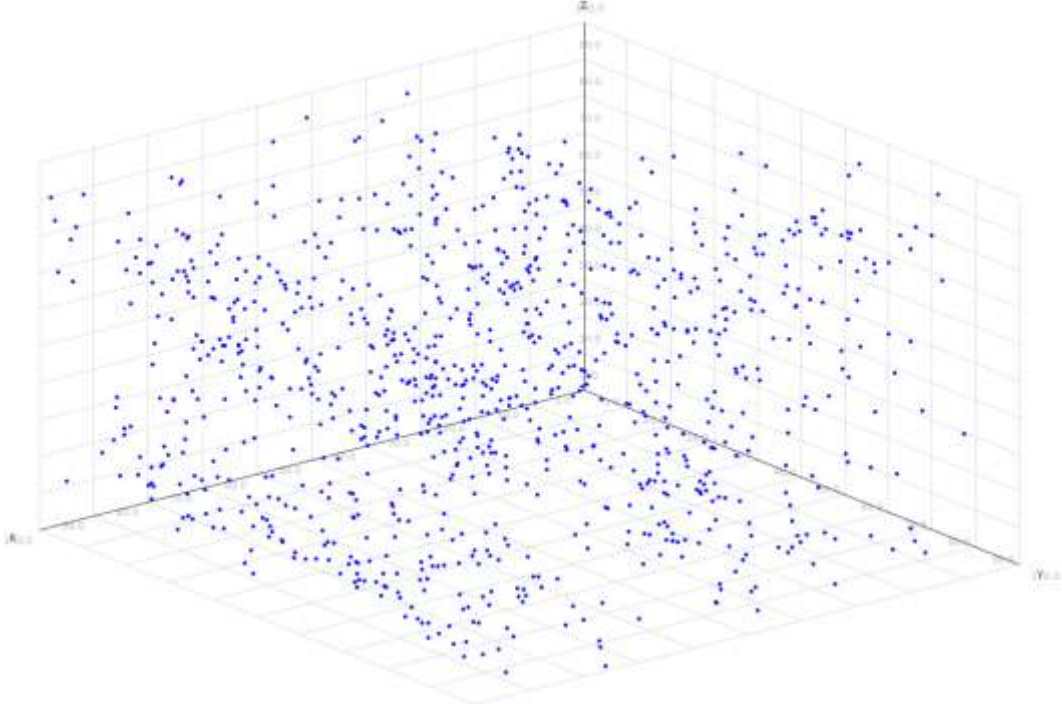
Gruplar verilmediğinde, yeni bir problemle karşı karşıya kalırız. Düğümleri doğru gruplara yerleştirmek olarak ifade edilebilen bu probleme *düzlemeleme* problemi adı verilir. Düzlemeleme problemini çözenin en direkt yolu Cayley-Menger determinantını kullanmaktır Blumenthal (2004). ETK ise, düzlemelemeden hemen sonra devreye girer.

Düzlemsel bir ortamda, düzlemlerin sayısı 1 ile $\frac{n}{3}$ arasında değişebilir ve her ortam farklı karakteristik özellik gösterir. Düzlemselliği modellemek adına, *düzlemsellik çarpanı* adı verdiğimiz ve $1 - \frac{k}{n}$ şeklinde hesaplanan bir katsayı tanımlıyoruz ve μ ile gösteriyoruz.

Düzlemsellik Katsayısı

Bu bölümde, *düzlemsellik katsayısı* adını verdiğimiz yeni bir terim tanımlıyoruz. Düzlemsellik katsayısı, bir ortamdaki düğümlerin ne derece düzlemsel olduğunu belirler. Düzlem sayısı ve her düzlem üzerindeki düğüm sayısına bağlı olarak değişir. Ancak, bu iki parametre düzlemselliği modellemek için yeterli değildir. Aynı zamanda ortamın hacmini de hesaba katmamız gerekir. Bizim durumumuzda, ortamın hacmi sabit ve 10^6 birim küp olarak belirlenmiştir.

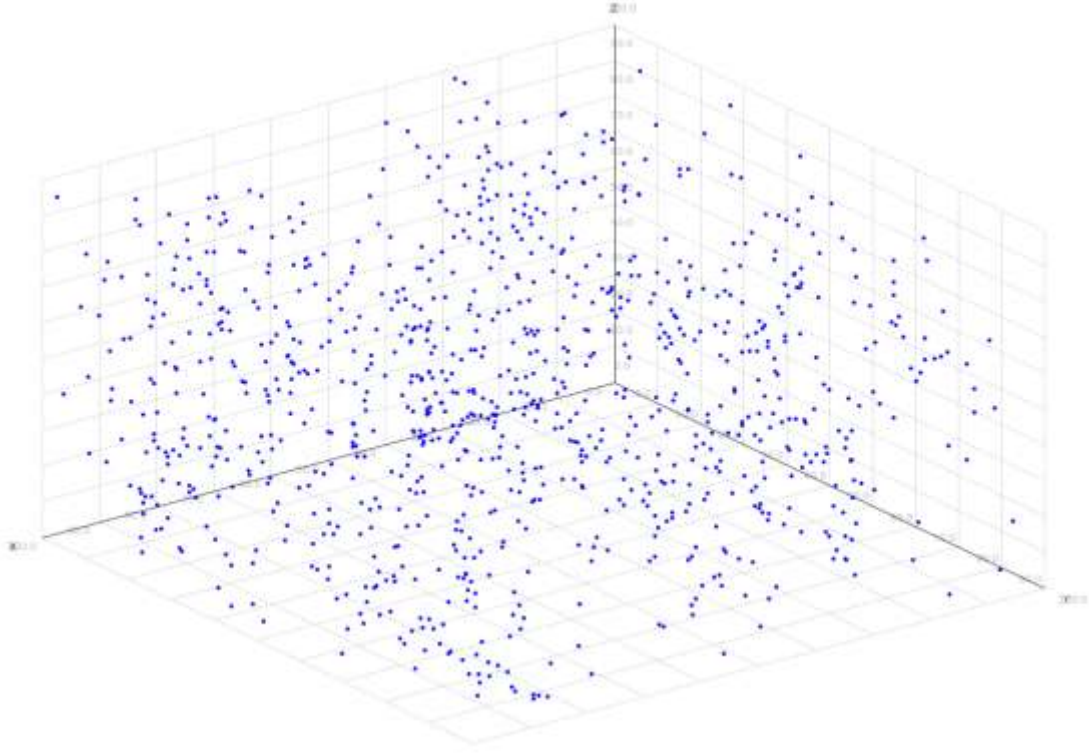
Şekil 4.2.10'da, her düzlemin üzerinde dört düğüm bulunuyor. Yani, ortam neredeyse düzlemsel değil. Gözle görülür düzlemsel yapılar mevcut değil.



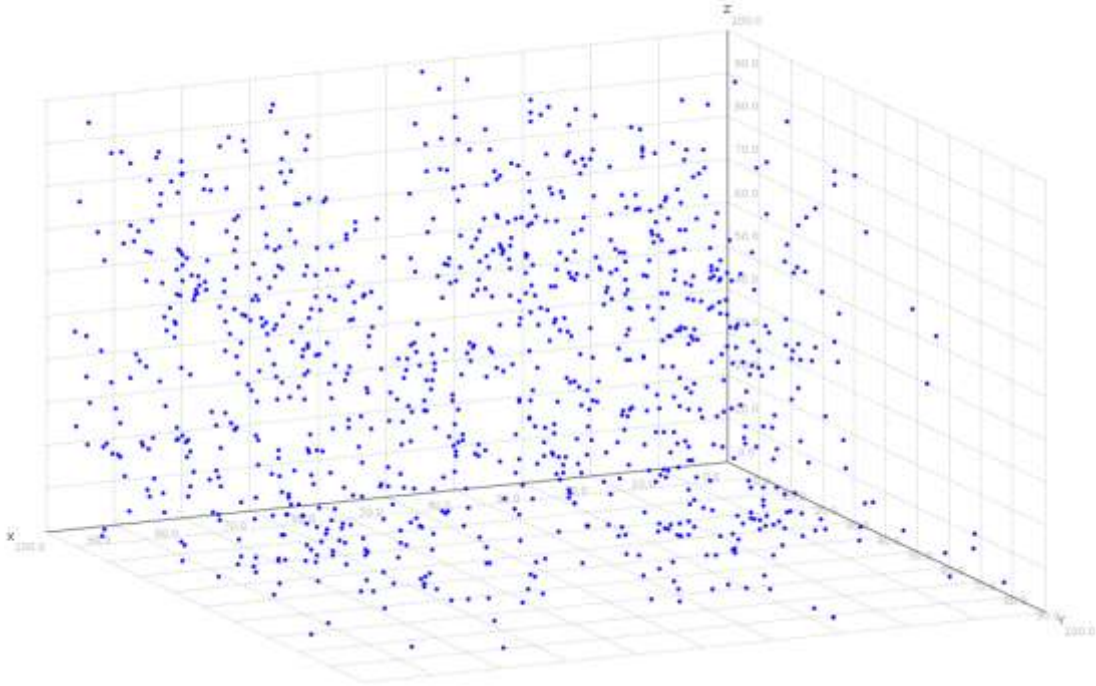
Şekil 4.2.10 $100 \times 100 \times 100$ bir ortam. $\mu = 0.750$

Şekil 4.2.11'de 192 tane düzlem ve her birinde beşer tane düğüm bulunmakta.

Şekil 4.3.12, orta düzlemsellik faktörünün alt eşiği olan bir örnek görüyoruz. Düzlemsel yapılar çok net olmasa da, 96 düzlem var ve düzlem başına 10 düğüm düşüyor. Bu da ortamı oldukça düzlemsel yapar.

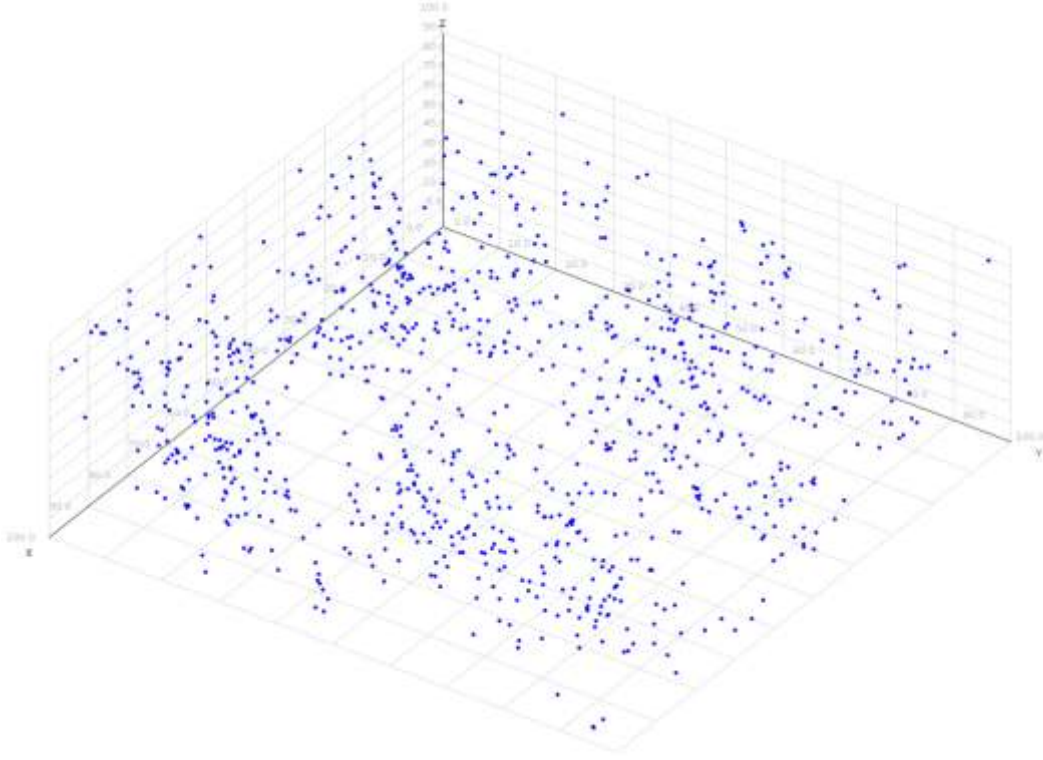


Şekil 4.2.11 100 × 100 × 100 bir ortam. $\mu = 0.800$



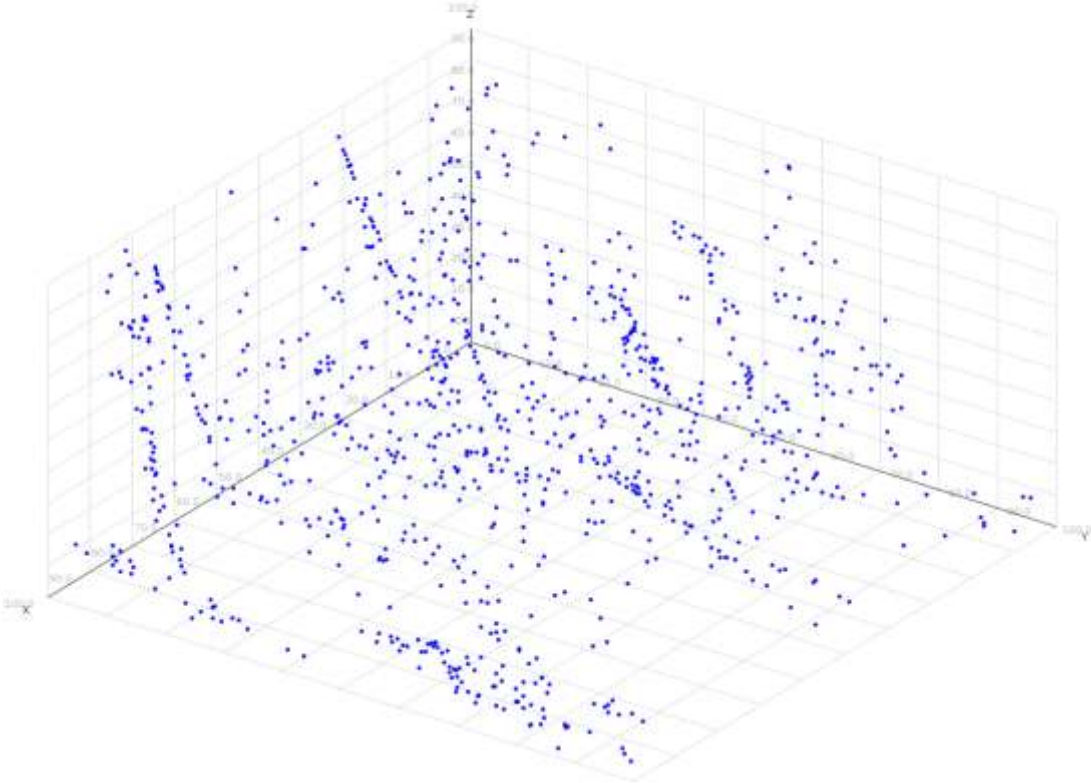
Şekil 4.2.12 100 × 100 × 100 bir ortam. $\mu = 0.900$

Şekil 4.2.13, yine orta düzlemsellik katsayısına örnek bir dizilişi gösteriyor. Ancak bu sefer, önceki şekillerden farklı olarak, düzlemsel yapılar daha belirgin.



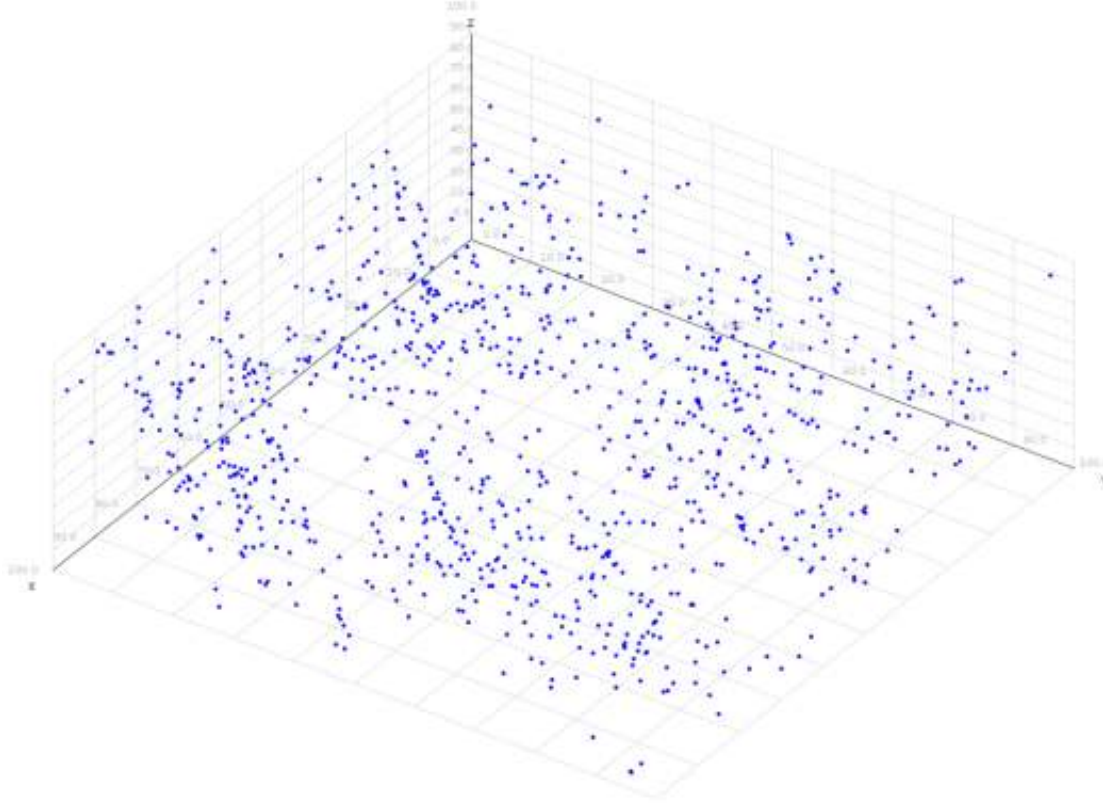
Şekil 4.2.13 100 × 100 × 100 bir ortam. $\mu = 0.933$

Şekil 4.2.14, ortamın ortasında bir düzlemsellik katsayısını örnekliyor. 40 düzlem ve her düzlem üzerinde 24 sensör düğümü mevcut.



Şekil 4.2.14 100 × 100 × 100 bir ortam. $\mu = 0.958$

Şekil 4.2.15'te yüksek μ gözlüyoruz. Deneylerimizde kullandığımız en düzlemsel ortam bu şekilde örneği verilen ortam. 8 tane düzlem ve her düzlemde 120 tane sensör düğümü var. μ bu denli yüksekken, konumlanmanın kalitesi düzlemlerin dizilişine bağlıdır. Sadece 8 düzlem olduğundan, bu düzlemlerin birbirine çok yakın olması çok başarılı bir konumlamaya, çok ayırık olması da göreceli olarak düşük başarılı bir konumlamaya sebebiyet verir.



Şekil 4.2.15 $100 \times 100 \times 100$ bir ortam. $\mu = 0.992$

Tablo 4.2.1, düzlemsellik katsayısının düzlem ve düzlem başına düşen düğüm sayısına göre değişimini gösteriyor. P kolonu, $100 \times 100 \times 100$ ortamdaki düzlem sayısını, NpP , düzlem başına düşen düğüm sayısını ve μ de düzlemsellik katsayısını belirtiyor. Son kolonsa söz konusu düzlemsellik faktörünün hangi aralığa düştüğünü gösteriyor.

Tablo 4.2.1 Düzlemsellik faktörünün düzlem sayısı ve düzlem başına düşen düğüm sayısına göre değişimi ve düştüğü aralık.

P	NpP	μ	Aralık
8	120	0.992	Yüksek
16	60	0.983	
24	40	0.975	Orta
32	30	0.967	
40	24	0.958	
48	20	0.950	
64	15	0.933	
80	12	0.917	
96	10	0.900	
120	8	0.875	
160	6	0.833	Düşük
192	5	0.800	
240	4	0.750	

Eşdüzlemsellik Tabanlı Konumlama

ETK, düzlemsel ortamlarda düğümlerin konuşlandığı düzlemleri konumlamaya yarayan bir algoritmadır. İlk olarak, tohum düzlem adını verdiğimiz, denklemlerinin bilindiği kabul edilen iki düzlem seçiyoruz. Bu düzlemleri, düzlemlerarası bağlantısı en fazla olan düzlemlerden iki tanesi olarak alıyoruz. Bir eşdüzlemsel düğüm grubunu konumlamak, bu düğüm grubunun üzerinde bulunduğu düzlemi konumlamak demektir. Bu düzlemi konumlamak ise, sözkonusu düğüm grubundan üç tane düğümün koordinatlarını bulmak anlamına gelir. Düzlemin denklemini bulmamıza yardımcı olan düğümlere *destek düğümleri* adını veriyoruz.

Algoritmamız, tohum düzlemler üzerindeki tüm düğümlerin konumlarının doğru olarak bilindiğini varsayarak başlıyor. Daha sonra kalan düğümleri konumluyoruz. Her adımda, bir sensör düğümü seçip konumlanabilir olup olmadığına bakıyoruz. Ancak, düğümler arasında rastgele seçim yapmak yerine, bir düzlemdeki düğüme baktıktan sonra aynı düzlemdeki diğer düğümleri dolaşyoruz. Eğer eşdüzlemsel noktalardan üçünü konumlayabilirsek, aynı düzlem üzerindeki kalan noktaları da bu üç destek düğümü sayesinde konumlayabiliriz.

Eğer P ile gösterilen bir eşdüzlemsel düğüm grubu içinde $\{p_i, p_j, p_k\}$ destek düğüm altkütmesi bulunuyorsa, M adı verdiğimiz bir dönüştürme matrisi tanımlarız. Bu matrisi tanımlarken destek düğümlerinin sanal konumlarını, $\{p_i^v, p_j^v, p_k^v\}$ ve gerçek konumlarını $\{p_i^r, p_j^r, p_k^r\}$

kullanıyoruz. $O(0,0,0)$ orijin noktası olmak üzere, M matrisi $\{Op_i^v, Op_j^v, Op_k^v\}$ vektörlerini $\{Op_i^r, Op_j^r, Op_k^r\}$ vektörlerine dönüştürür. Daha sonra bu dönüştürümü $p_u \in P \setminus \{p_i, p_j, p_k\}$ düğümlerine uygulayarak kalan düğümlerin koordinatlarını buluruz.

Algoritma 7.1'de ETK'yi gösteriyoruz.

Algorithm 7.1: CBL	
	input: SENSOR GRAPH G
1	if <i>there is no localized node</i> then call FINDSEED();
2	$N \leftarrow$ <i>localized planes</i> ;
3	$P \leftarrow \emptyset$;
4	foreach $n \in N$ do $P \leftarrow P \cup$ <i>unlocalized neighbors of</i> n ;
5	while N <i>is not empty</i> do
6	$N \leftarrow \emptyset$;
7	foreach $p \in P$ do
8	if ISLOCALIZABLE(p) then mark p as localized;
9	end
10	foreach $n \in N$ do
11	$P \leftarrow P \cup$ <i>unlocalized neighbors of</i> n ;
12	end
13	end

İlk satırda, hali hazırda konumlanmış düğüm yoksa, en yüksek dereceli iki düzlemi tohum düzlem olarak seçiyoruz. Bu işlem Algoritma 7.2'de görülebilir. İki ila dördüncü satırlar arasında, konumlanmış düzlemlerin konumlanmamış komşul düzlemlerini potansiyel konumlanabilir düzlemler olarak bir kümeye ekliyoruz. Satır 5'ten satır 10'a kadar, bu düzlemleri bir döngü içerisinde, konumlanacak düzlem kalmayana kadar konumlamaya çalışıyoruz. Sekizinci satır bir düzlemin konumlanabilir olup olmadığını kontrol ediyor. Kontrol esnasında Algoritma 7.3'te gösterilen algoritmayı kullanıyor.

Algorithm 7.2: FINDSEED	
1	sort coplanar sensor groups descending <i>w.r.t</i> their interplanar connectivity;
2	return first two groups;

Algoritma 7.3'te, üçüncü satır, Algoritma 7.4'ün çıktısına bağlı olarak sözkonusu düğüm üçlüsünü işaretler.

Algorithm 7.3: ISLOCALIZABLEPLANE**input:** COPLANAR NODE GROUP P

- 1 $M \leftarrow$ unlocalized node set;
- 2 **foreach** non-collinear node triplet $\{i, j, k\} \in P \cap M$ **do**
- 3 | **if** ARELOCALIZABLESENSORS($\{i, j, k\}$) **then return true;**
- 4 **end**

Algoritma 7.4'te bir düğüm konumlanabilir mi diye kontrol ediyoruz.

Algorithm 7.4: ISLOCALIZABLESENSOR**input:** NODE n

- 1 $L \leftarrow$ localized neighbors of n ;
- 2 **if** there exists a non-coplanar node quadruplet $\{i, j, k, l\} \in L$ **then**
- 3 | **call** LOCALIZE($n, \{i, j, k, l\}$);
- 4 | **return true;**
- 5 **end**

Algorithm 7.5: ARELOCALIZABLESENSORS**input:** NODE SET N

- 1 **if** $\forall n \in N, \text{ISLOCALIZABLESENSOR}(n)$ **then return true;**

Algorithm 7.6: LOCALIZE**input:** NODE TO BE LOCALIZED n , BEACON NODES $\{i, j, k, l\}$

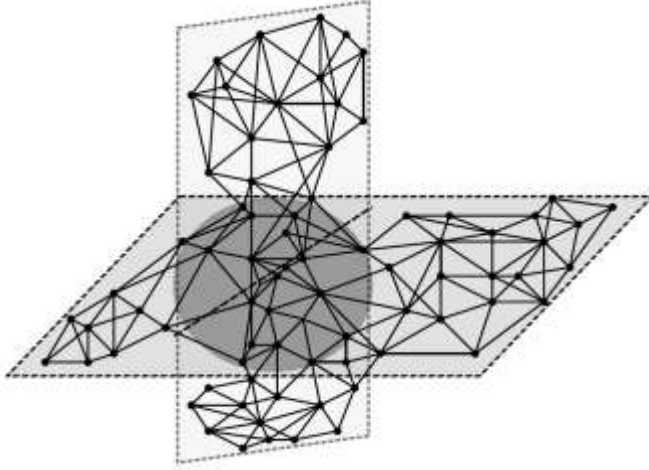
- 1 create four spheres using the beacon nodes and their Euclidean distances to n ;
- 2 find the intersection points of any three spheres;
- 3 **if** any of the intersection points is on the remaining sphere **then return** that point;

Grup bilgisini kullanmayan, naif konumlama algoritmasını Quadrilateration şeklinde isimlendireceğiz. Orijinal Quadrilateration'dan farklı olarak, bu algoritma, her düğüm dörtlüsünü denemek yerine sadece en yüksek dereceli düğüm dörtlüsünü deniyor ve başka dörtlüye geçmiyor.

Quadrilateration algoritmasının, ne kadar düğüm dörtlüsü denerse denerse, ETK kadar başarılı olamayacağı senaryolar mevcut. Örneğin, iki kesişen düzlem üzerine dağılmış sensörleri düşünelim. Bu durumda Quadrilateration ile ancak kesişime yakın olan, yani eşdüzlemsel olmayan düğümleri konumlayabiliriz. Ancak ETK kullanarak, her iki düzlem

üzerinde geriye kalan düğümleri de konumlamamız mümkün. Şekil 4.2.16'da anlatılan senaryoyu görüyoruz.

Şekil 4.2.16'da, koyu kısım eşdüzlemsel olmayan noktaların kapsadığı menzil.



Şekil 4.2.16 Kesikşen düzlemler üzerindeki iki eşdüzlemsel düğüm grubu.

4.2.8 Deneysel Sonuçlar

Deneyselerimizi yapmak için Java programlama dilini kullandık. Tüm deneyselerimiz $100 \times 100 \times 100$ kübik bir uzayda ve 960 düğümlerle gerçekleşti.

Düzlemsel ortamları modellemek için, bahsedilen kübik uzayı 8 eşit kübe bölüyoruz ve her birine eşit sayıda düzlem koyuyoruz. Rastgele oluşturduğumuz sensör düğümleri, bu düzlemlerin üzerine konuşlanıyor. Her düzleme eşit sayıda sensör düğümü koyuyoruz.

Deneyselerimizi üç ana başlık altında toplayabiliriz:

- **Hatasız mesafe ölçümleriyle yapılan deneyseler:** Bu grupta, mesafe ölçümlerinde hata bulunmamakta. 13 farklı düzlemsellik katsayısıyla çalışıyoruz. Her bir düzlemsellik katsayısı için, grup bilgisini kullandığımız ve kullanmadığımız durumları inceliyoruz. Bu katsayılar şu şekilde: 0.750, 0.800, 0.833, 0.875, 0.900, 0.917, 0.933, 0.950, 0.958, 0.967, 0.975, 0.983 ve 0.992.
- **Hatalı mesafe ölçümleriyle yapılan deneyseler:** Bu deney grubu, hatalı ölçümlerin konumlama üzerindeki etkisini gösterir nitelikte. Ölçüm hatasını, gerçek uzaklıklara $\pm \%e$ kadar bir rastgele sayı ekleyerek modelliyoruz. $e \in [1,20]$. Sensörlerin menzilleri ilk grup deneyin sonuçlarına bağlı olarak değişkenlik gösteriyor. Bu grup deneyseler için tüm düzlemsellik katsayılarını kullanmak yerine, aynı karakteristiği gösteren

düzlemsellik katsayılarını atlıyoruz ve 0.900, 0.917, 0.933, 0.958, 0.967, 0.992 katsayılarını kullanarak deneylerimizi gerçekleştiriyoruz.

- **Hatalı grup bilgisiyle yapılan deneyler:** Bu grupta ise, grup bilgisinin hatalı geldiğini varsayarak deneyler yapıyoruz. Test verisini oluşturduktan sonra, rastgele $\%m$ sensör düğümünü seçerek gruplarından ayırıyoruz. Deneylerimiz esnasında m sayısı 1 ila 50 arasında değişiyor, hata payı $\pm\%1$ ve sensör menzili 18 birim.

Terminoloji

Deney sonuçları olarak sunduğumuz çizgelerde aşağıdaki terminolojiyi kullanıyoruz:

Quadrilateration: Grup bilgisini kullanmadığımız durumlar.

CBL: Grup bilgisini kullandığımız durumlar, ETK.

Recall percentage: Konumlanmış olarak işaretlenen sensör yüzdesi. CBL için $\%C$, Quadrilateration için $\%Q$.

Average offset: Düğüm başına düşen konumsal sapma.

Unclassified flips: Tüm işlemdeki toplam tasnif edilmemiş yansıma sayısı. Her düğüm sadece bir kez sayılıyor.

Recall difference: Grup bilgisinin kullanıldığı ve kullanılmadığı durumlar arasındaki konumlama yüzdesi farkı. $\%C - \%Q$ olarak hesaplanır ve R_{dif} olarak gösterilir.

Maximum difference: $|R_{dif}|$ değerinin en yüksek olduğu nokta.

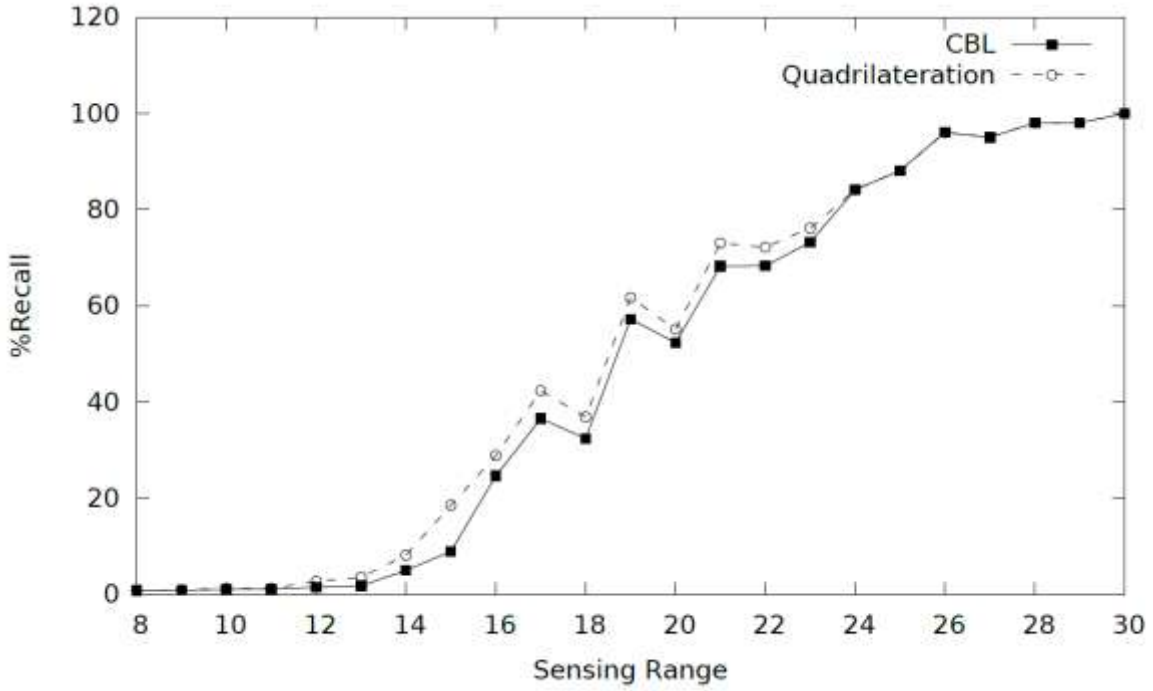
Critical point: y eksenindeki farkın en büyük olduğu noktanın x eksenindeki karşılığı.

Meet point: x ekseninde, $|R_{dif}|$ değerinin yüzde beşinde altında olduğu ve öyle kaldığı ilk nokta.

Hatasız Mesafe Ölçümleriyle Yapılan Deneyler

13 farklı düzlemsellik katsayısıyla gerçekleştirdiğimiz ve hatasız ölçümlerle yaptığımız deneyleri bu bölümde sunuyoruz.

Şekil 4.2.17'de, en düşük düzlemsellik katsayısıyla yapılan deneyi görüyoruz. Bu durumda, grup bilgisini kullanmak vakit kaybından çok, konumlamanın kalitesini de düşürmekte. R_{dif} , sıfırın altına düşüyor. Buluşma noktası(meet point) 25 birim menzil.

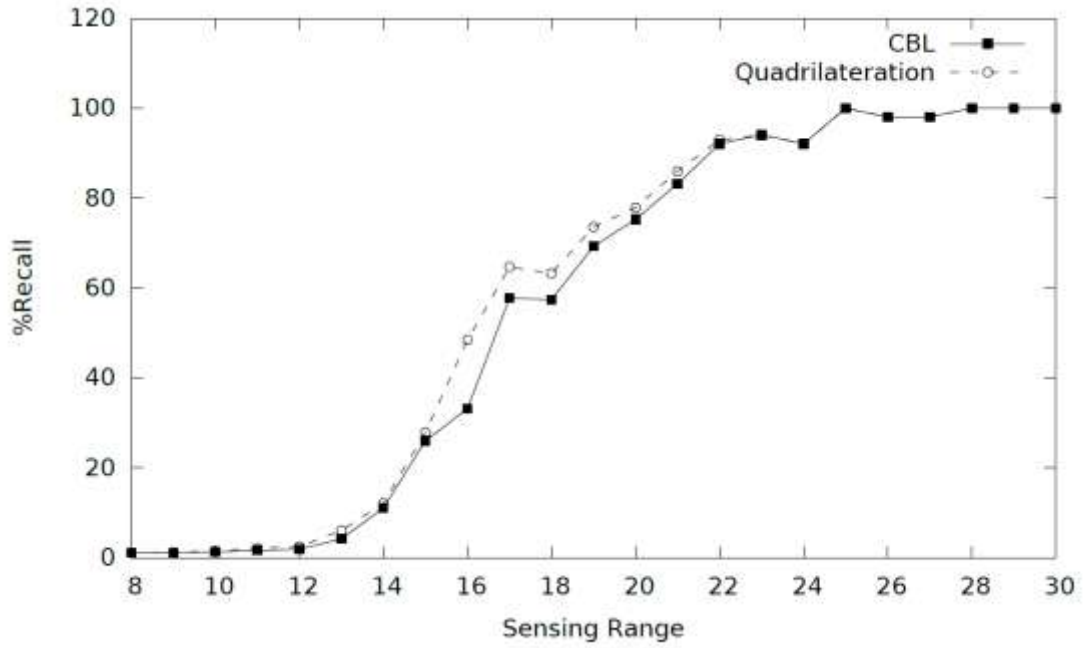


Şekil 4.2.17 Her iki algoritmanın uzaklık ölçüm hatası olmadan konumlama yüzdeleri.
 $\mu = 0.750$

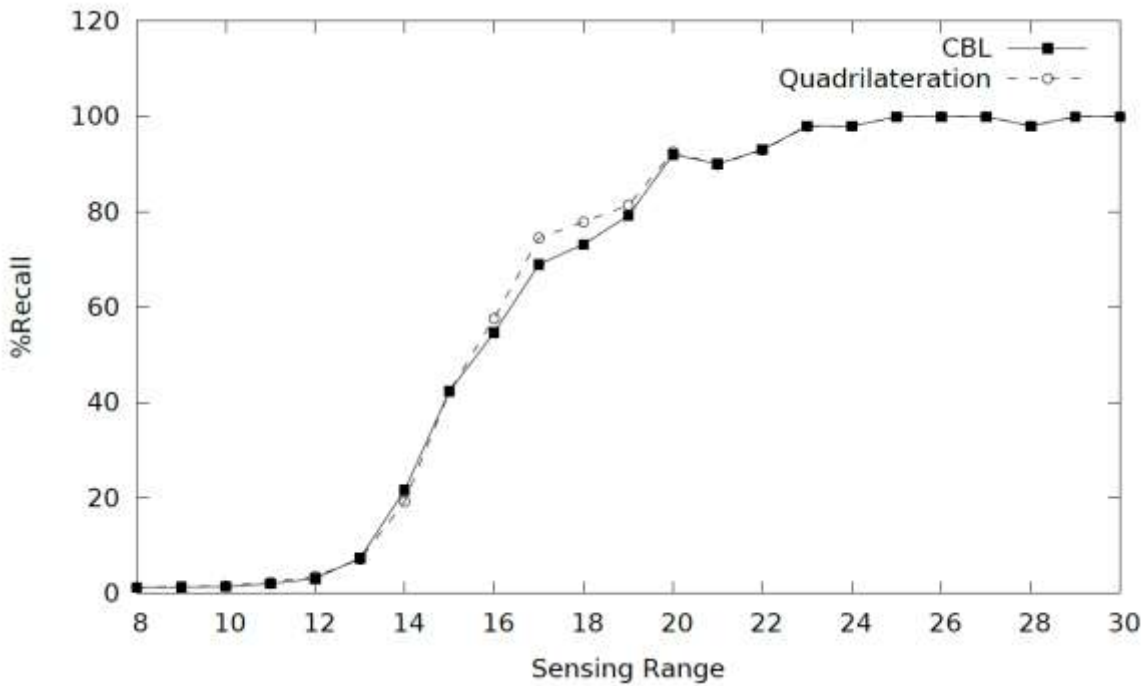
Şekil 4.2.18'de, genel olarak performans farkı Şekil 4.2.17'de olduğundan daha düşük. Ancak yine de, ortam yeterince düzlemsel olmadığı için grup bilgisini kullanmak kötü sonuçlar elde etmeye yol açıyor. Düzlemsellik katsayısının 0.800 olduğu bu durumda, buluşma noktası 23.

Şekil 4.2.19'da gösterilen grafikte düzlemsellik katsayısı düşüğün üst sınırına yakın. Burada da görüyoruz ki, grup bilgisini kullanmak, kullanmamaktan daha kötü sonuç veriyor. Şekil 4.2.20, düşük düzlemsellik faktörünün üst sınırıyla yapılan deneylerin sonuçlarını gösteriyor. 13 ila 16 arası algılama menziline grup bilgisini kullanmak konumlama sürecinde gelişme görmemizi sağlıyor. μ 0.857 olduğunda, kritik nokta (critical point), 14 birim menzil.

Şekil 4.2.21'den itibaren orta düzlemsellik katsayısıyla yaptığımız deneyleri gösteriyoruz. Şekil 4.2.21'de düzlemsellik katsayısı 0.900. Grup bilgisini kullanmak, menzil 16 birimin altındayken fazlasıyla fayda sağlıyor. Bu durumda kritik nokta 14 birim menzil. Ancak, sensör menzili 16 ila 18 birim arasındayken,



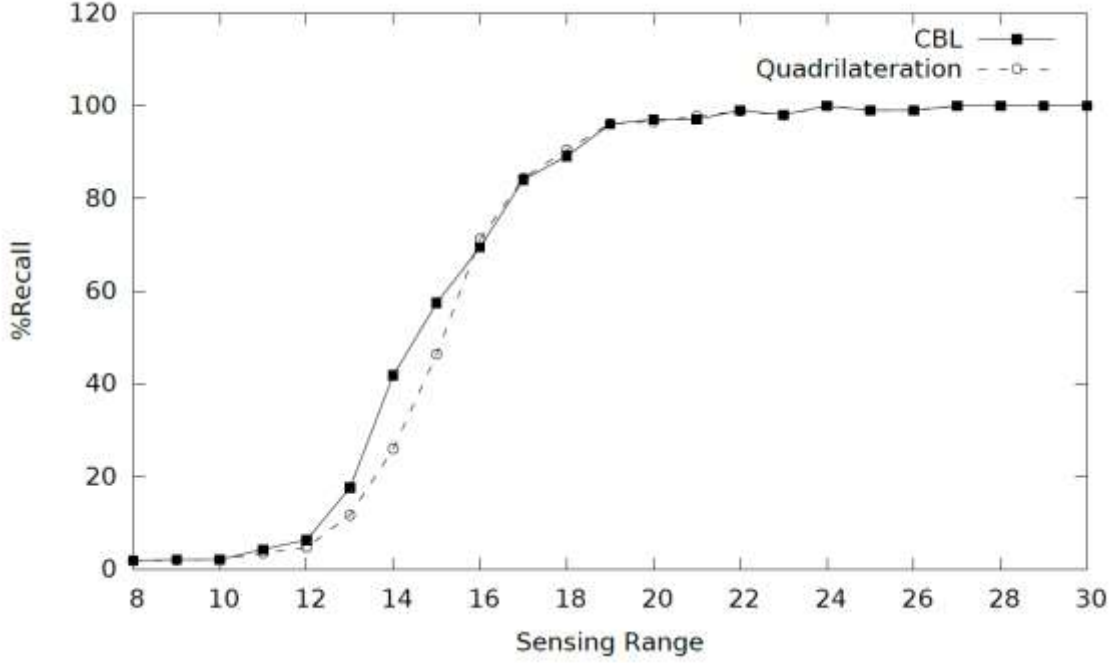
Şekil 4.2.18 Her iki algoritmanın uzaklık ölçüm hatası olmadan konumlama yüzdeleri.
 $\mu = 0.800$



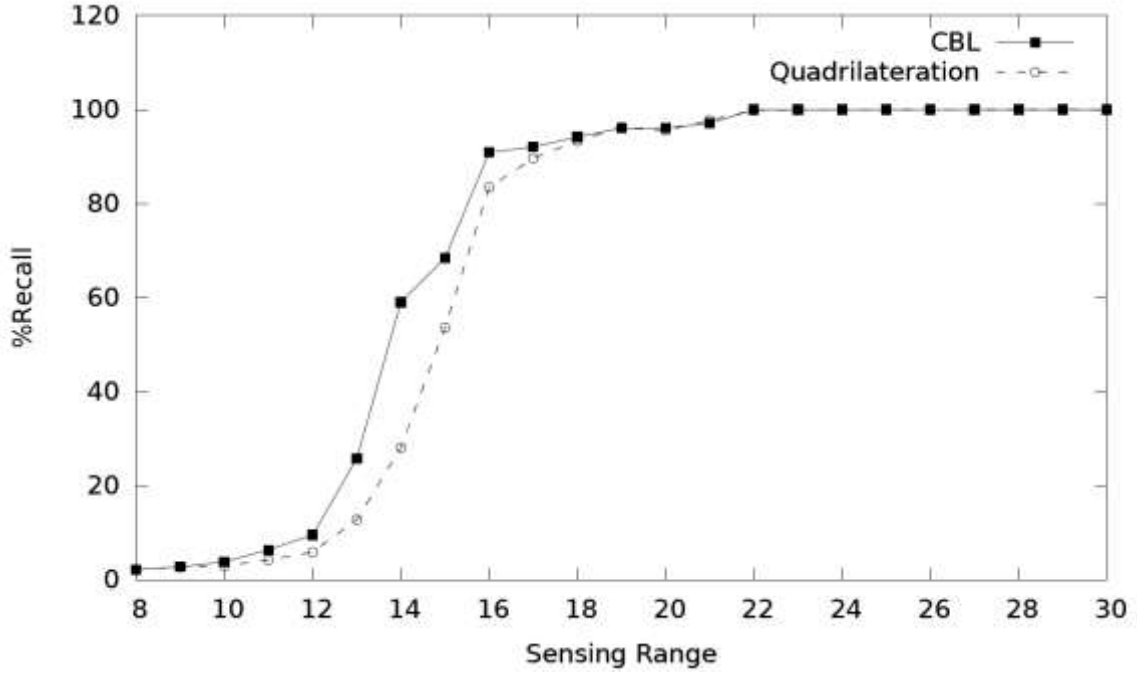
Şekil 4.2.19 Her iki algoritmanın uzaklık ölçüm hatası olmadan konumlama yüzdeleri.
 $\mu = 0.833$

ETK'nin performansında bir artış gözlemlenmiyor. Bu durum, düzlemlerarası bağlantıların azlığından doğan bir sonuç. Düzlemlerarası yeterli bağlantı olmaması, ETK'nin bir

düzlemden diğerine devam etmesini engelliyor. Düzlemsellik faktörü 0.900 iken buluşma noktası 17 birim menzil.

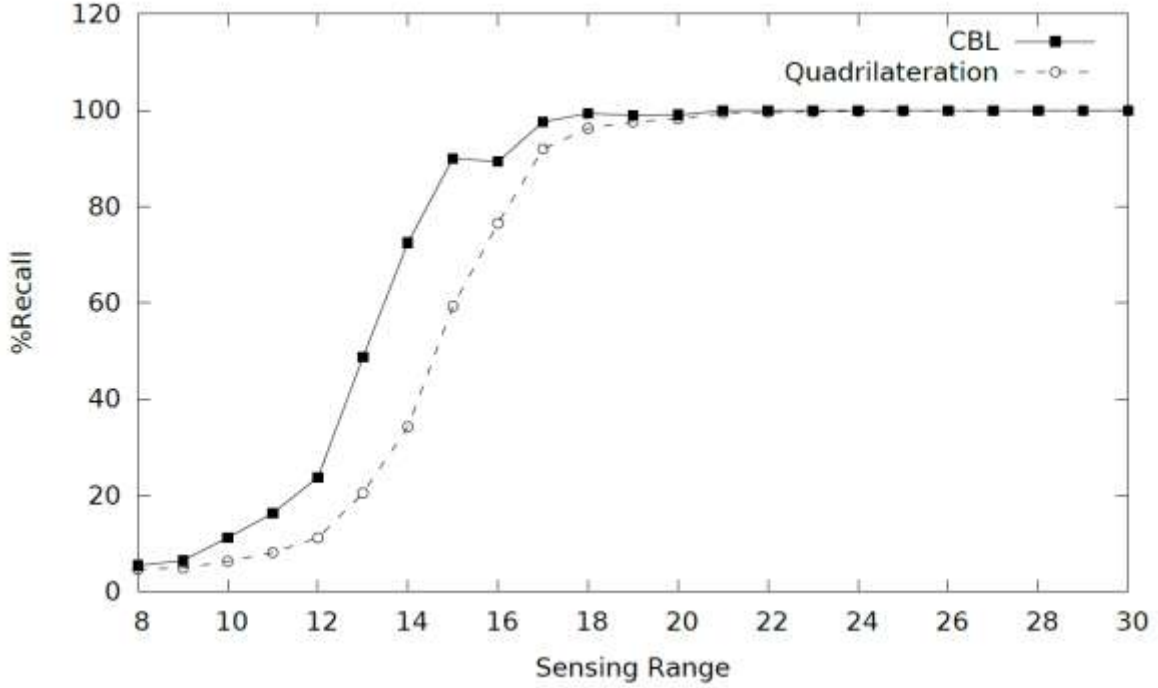


Şekil 4.2.20 Her iki algoritmanın uzaklık ölçüm hatası olmadan konumlama yüzdeleri.
 $\mu = 0.875$



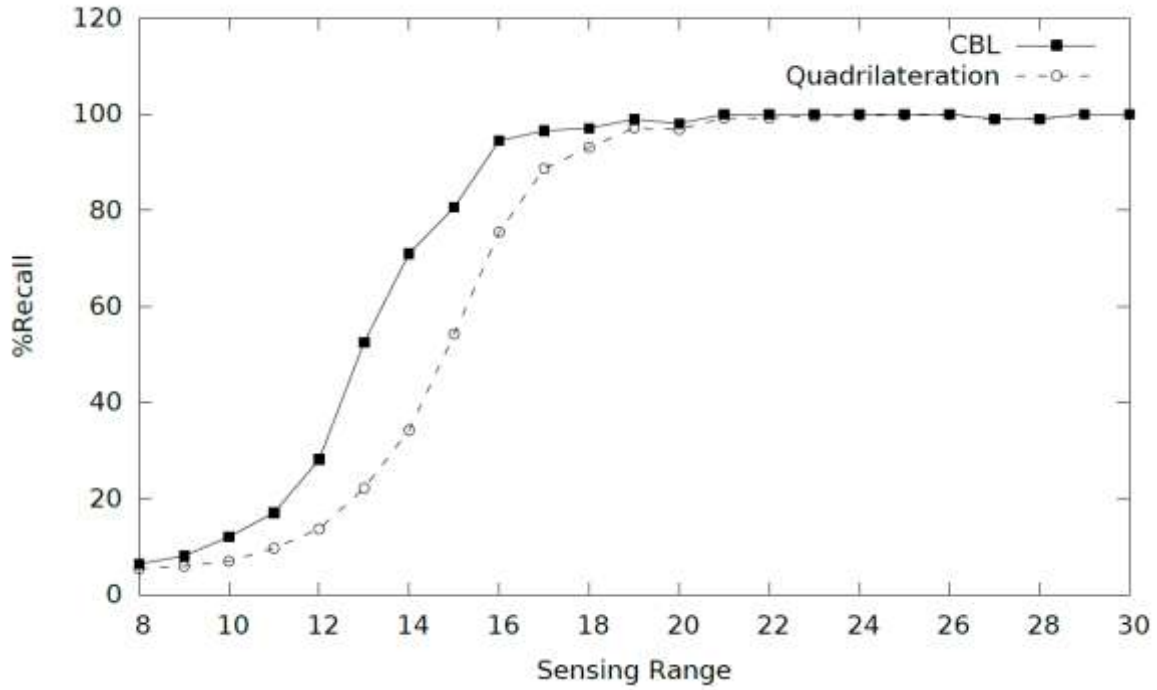
Şekil 4.2.21 Her iki algoritmanın uzaklık ölçüm hatası olmadan konumlama yüzdeleri.
 $\mu = 0.900$

Şekil 4.2.22, 8.23, 8.24 ve 8.25, grup bilgisinin elde edilmesinin konumlama sürecine çok şey katacağı durumları gösteriyor. 10 birim menzilden başlayarak, grup bilgisi her zaman daha iyi sonuç veriyor. Ta ki düğüm bağlantılarının her düğümü konumlayabilmemizi sağlayacak kadar fazla olduğu duruma kadar. Şekil 4.2.22’de, ETK’nin düşük menzillerde Quadrilateration’a kıyasla ne kadar yüksek yüzdelerle konumlama yaptığını görebiliyoruz.



Şekil 4.2.22 Her iki algoritmanın uzaklık ölçüm hatası olmadan konumlama yüzdeleri.
 $\mu = 0.950$

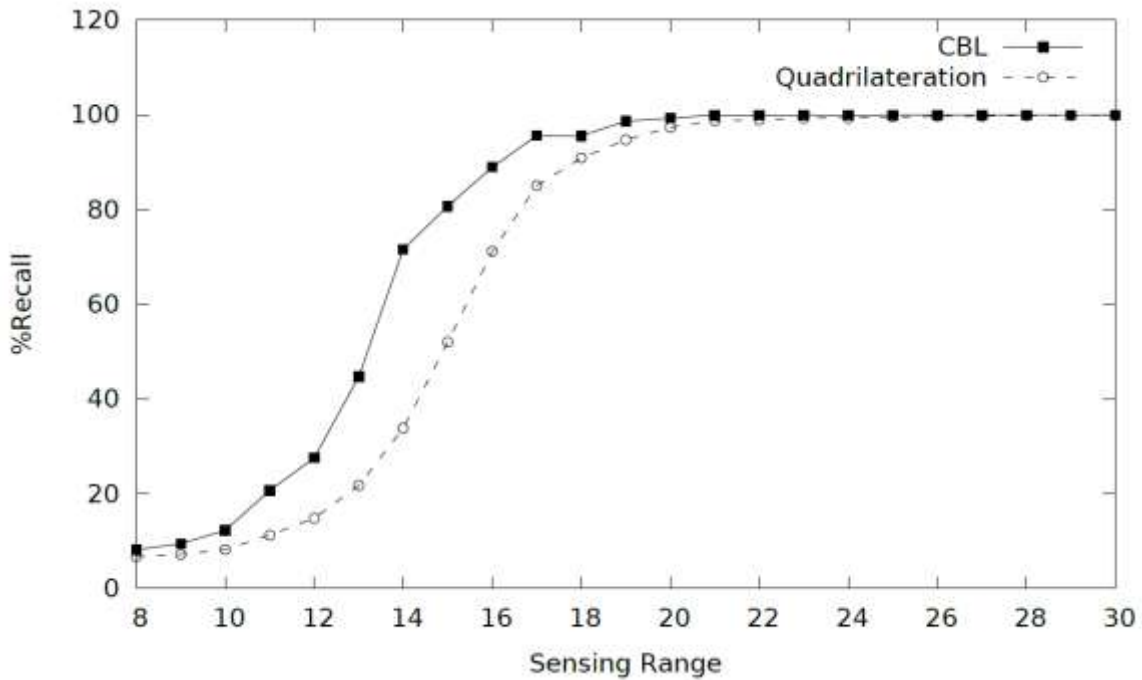
Şekil 4.2.23’te, bir başka orta μ ile yaptığımız deneyleri görüyoruz. Kritik nokta 14 birim menzil ve $R_{dif} = 36$. Buluşma noktası 21 birimlik menzilde gerçekleşmekte.



Şekil 4.2.23 Her iki algoritmanın uzaklık ölçüm hatası olmadan konumlama yüzdeleri.

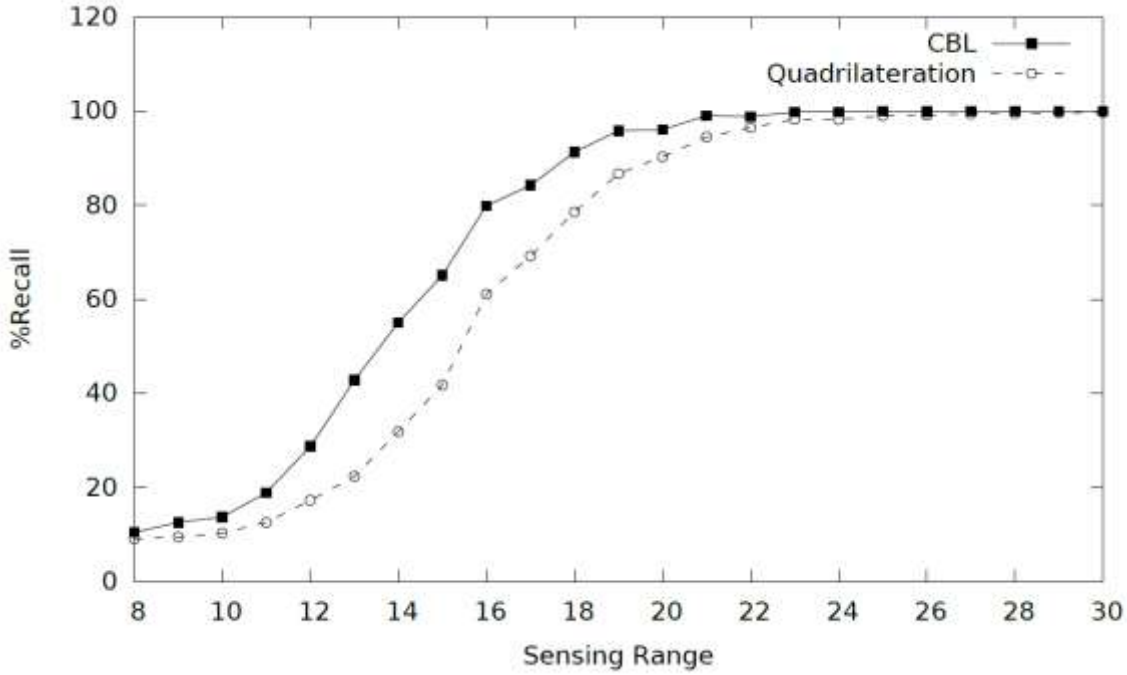
$$\mu = 0.958$$

Şekil 4.2.24'teki grafik, Şekil 4.2.23 grafik ile aynı karakteristiğe sahip. Düşük menzillerde fazlasıyla iyi performans gösteren ETK, menzil yükseldikçe, düğümler arası bağlantılardan dolayı Quadrilateration ile aynı yüzdede konumlama yapıyor. Ancak μ değeri 0.967 olduğu zaman, kritik nokta 14'te kalırken, gözlemlenen en yüksek R_{dif} değerine ulaşıyoruz. R_{dif} , bu durumda %39.



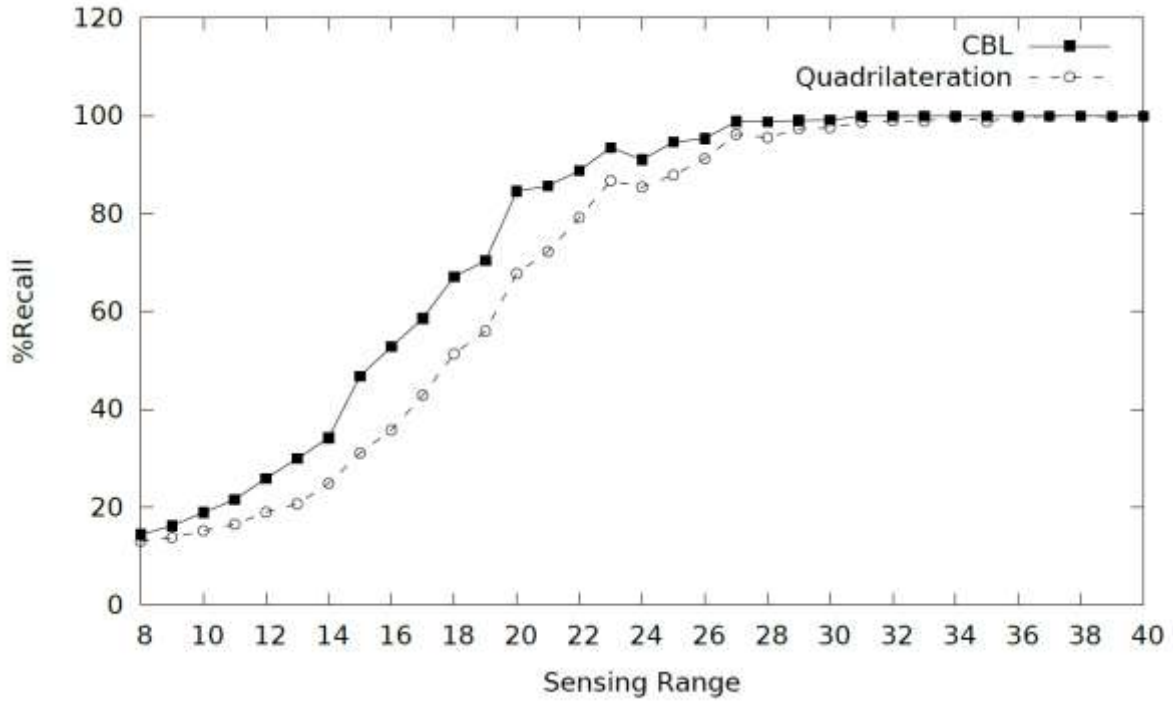
Şekil 4.2.24 Her iki algoritmanın uzaklık ölçüm hatası olmadan konumlama yüzdeleri.
 $\mu = 0.967$

Şekil 4.2.25, performans çizgileri arasındaki boşluğun git gide daralmaya başladığını gösteren bir grafik sunuyor. Ancak, grup bilgisi hala gayet kullanışlı vaziyette. Quadrilateration uygun dörtlülere bulamadığından konulamaya devam edemezken, ETK, grup bilgisini kullanarak devam edebiliyor.



Şekil 4.2.25 Her iki algoritmanın uzaklık ölçüm hatası olmadan konumlama yüzdeleri.
 $\mu = 0.975$

Şekil 4.2.26, yüksek düzlemsellik katsayılarıyla deney yaptığımız ilk grafiği gösteriyor. Grafikte, önceki şekillerden farklı olarak, grup bilgisinin işe yaradığını daha net görebilmek amacıyla, algılama menzilinün üst sınırını artırıyoruz. Diğer şekillerde 30 birimlik menzil buluşma noktasını görmemize yetiyorken, $\mu = 0.983$ olduğu zaman 40 birime kadar çıkması gerekiyor. Buluşma noktası 32. Bu da, bu denli yüksek bir düzlemsellik katsayısı söz konusu olunca, menzilin çok işe yaramadığını gösteriyor.



Şekil 4.2.26 Her iki algoritmanın uzaklık ölçüm hatası olmadan konumlama yüzdeleri.

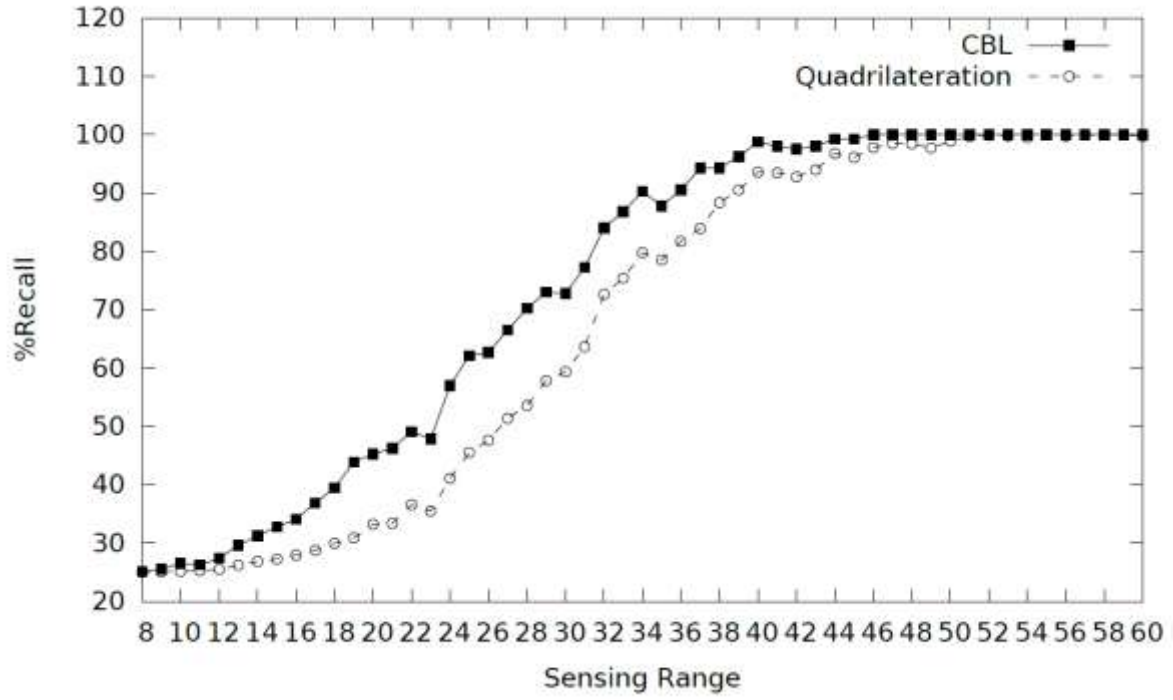
$\mu = 0.983$

Şekil 4.2.27'de, düzlemsellik katsayısı aşırı fazla. Toplam 8 düzlem ve her birinin üzerinde 120 tane düğüm mevcut. Doğal olarak, bir önceki deneydeki durumun benzeri burada da yaşanıyor. Sensörlerin menzilin limitini 60 olarak tanımlıyoruz. Böylece, buluşma noktasının 51 olduğunu gözlemleyebiliyoruz.

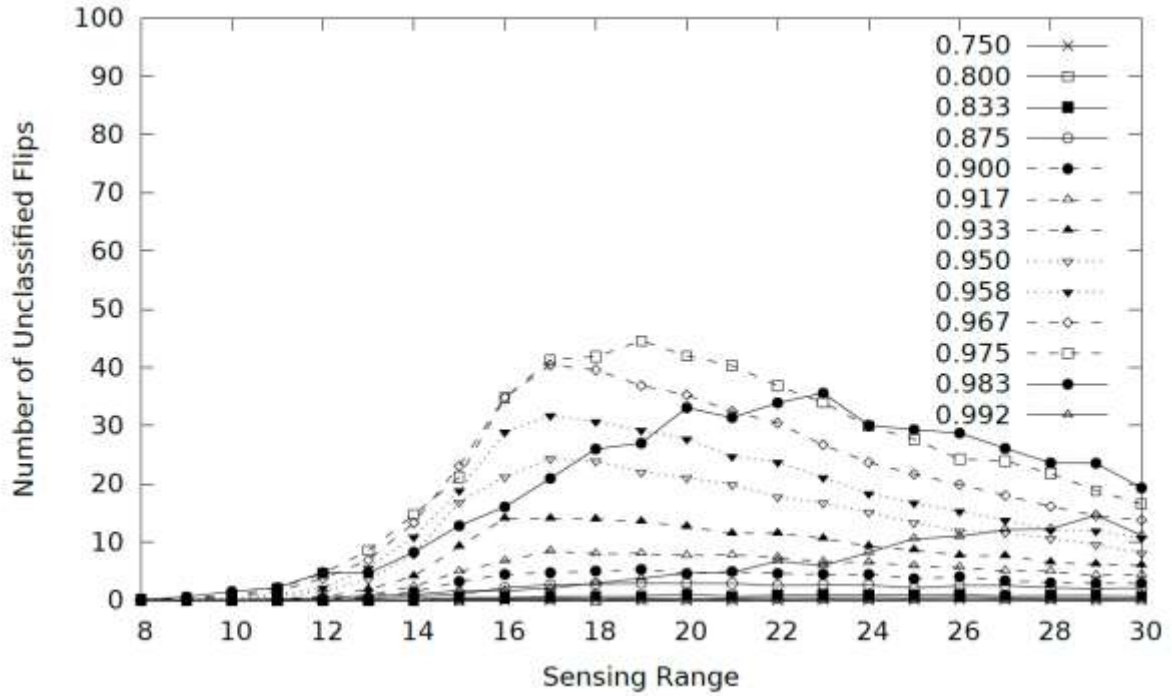
Şimdiye kadarki deney sonuçlarından, grup bilgisinin en fazla işe yaradığı koşulların hatasız ölçümler olduğunda ve sensör algılama menzili 14 birim olduğunda gözlemlendiğini söyleyebiliriz.

Hatalı ölçümlerle yaptığımız deneylere geçmeden önce, grup bilgisini kullanmanın önemli bir etkisini daha göstermek istiyoruz.

Grup bilgisinin işlenmesi, ölçüm hatası olmayan her durumda tasnif edilmemiş yansımaları engelliyor. Şekil 4.2.28, Quadrilateration'un çeşitli μ değerleri ve sensör menzillerine göre kaydettiği tasnif edilmemiş yansıma sayısını göstermekte. Ancak, gösterilen grafikte sensör menzili 30 birime kadar gidiyor.



Şekil 4.2.27 Her iki algoritmanın uzaklık ölçüm hatası olmadan konumlama yüzdeleri.
 $\mu = 0.992$



Şekil 4.2.28 Quadrilateration için μ ve menzile bağlı olarak gözlenen tasnif edilmemiş yansımalar.

Tablo 2’de, bu bölümde yapılan deneylerin kritik noktaları, konumlama yüzde farkları ve buluşma noktalarının düzlemsellik katsayısına bağlı değişimi görülebilir. **CP** kolonu kritik noktaları, **MP** kolonuysa buluşma noktalarını temsil ediyor.

Tablo 4.2.2 Kritik noktalar, en fazla konumlama farkları ve buluşma noktalarının düzlemsellik katsayısıyla değişimi

μ	CP	R _{dif}	MP
0.750	15	-9	24
0.800	16	-15	23
0.833	17	-5	21
0.875	16	-2	20
0.900	14	31	17
0.917	14	29	17
0.933	14	35	17
0.950	14	39	18
0.958	14	36	18
0.967	14	37	18
0.975	14	23	21
0.983	20	18	26
0.992	28	17	41

Hatalı Mesafe Ölçüleriyle Yapılan Deneyler

Bu bölümde, sensör düğümlerinin mesafe ölçümlerine hata payı katıyoruz ve yedi farklı düzlemsellik katsayısıyla deneylerimizi yapıyoruz. Hata payı %1’den başlayıp %20’ye kadar çıkıyor. Deneylerimiz, grup bilgisi olan ve olmayan durumlarda kritik noktalar ve buluşma noktalarında yapılan testleri kapsamaktadır.

Konumlamanın kalitesini üç metrik üzerinden değerlendirmeye alıyoruz:

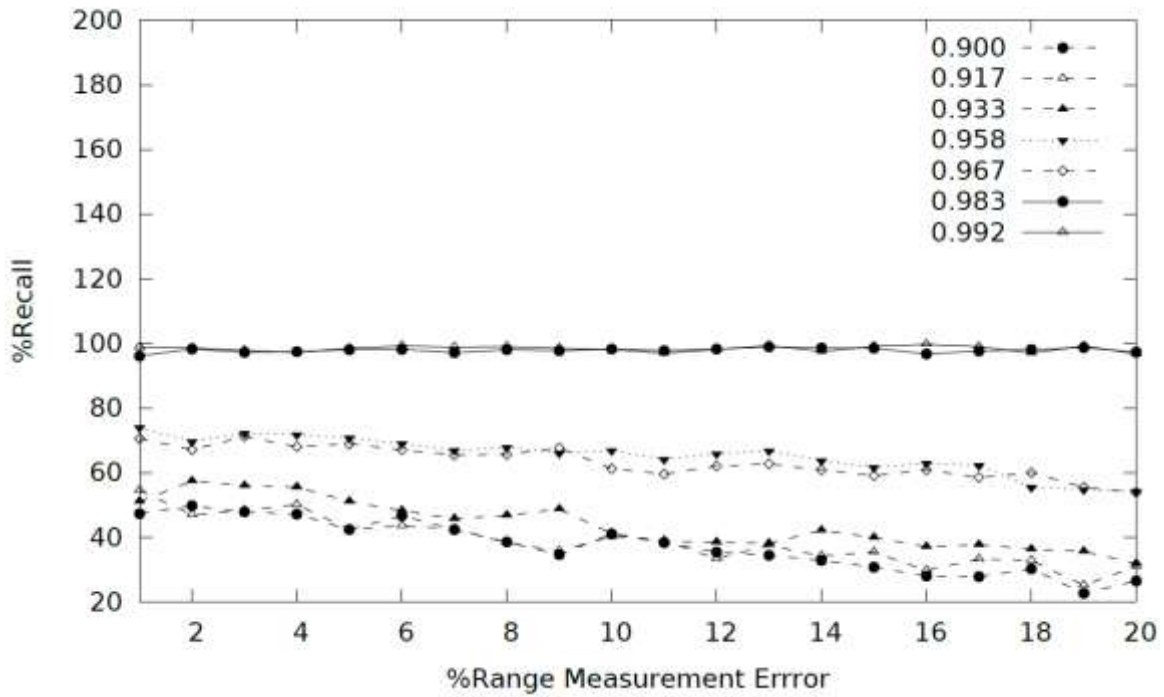
- Konumlama yüzdesi (recall percentage)
- Ortalama sapma (average offset)
- Tasnif edilmemiş yansımaların sayısı (number of unclassified flips)

Buluşma Noktalarıyla Yapılan Deneyler

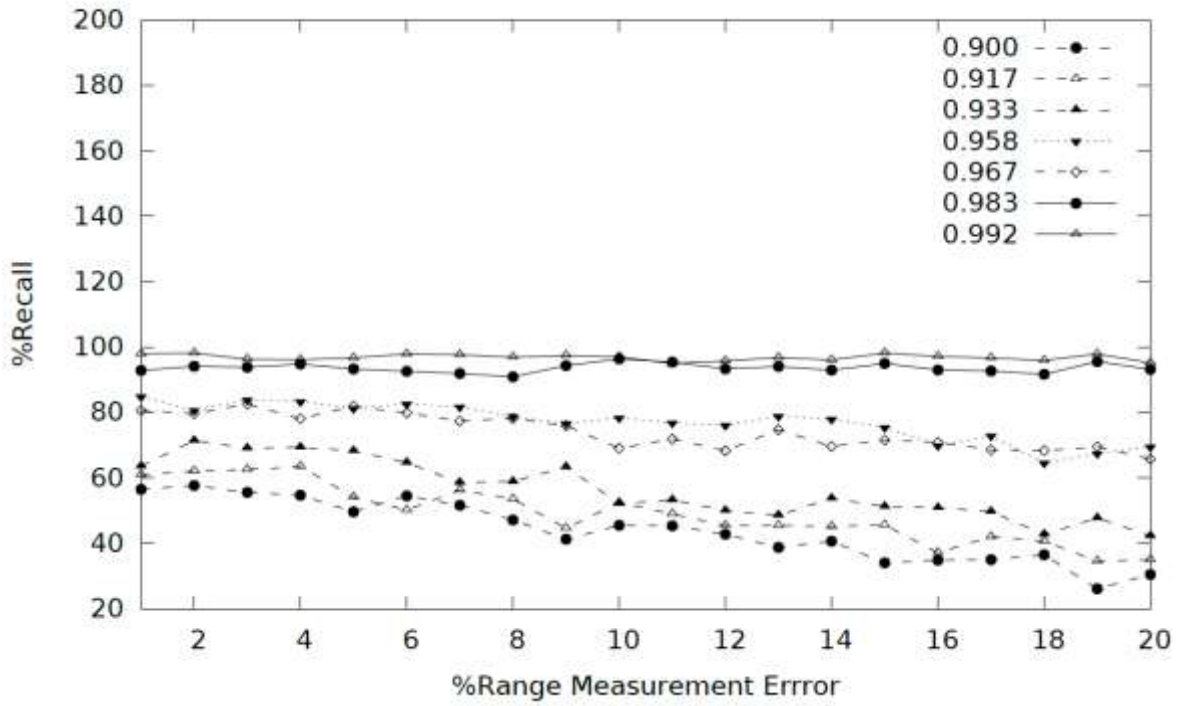
Bu bölümdeki deneylerimiz, ölçüm hatası olmadığında grup bilgisini kullanmak veya kullanmamak arasında bir fark olmadığı durumlarda, buluşma noktalarında, hata payı devreye girince neler olduğunu gözlemlemek için yapılmıştır.

Şekil 4.2.29'da, grup bilgisi kullanmadığımızda ortaya çıkan konumlanma yüzdeleri görüyoruz. Yüksek μ değerlerinde, menzil de çok yüksek olduğundan, çok bir değişim gözlemlemiyoruz. Ancak düzlemsellik katsayısının orta olduğu durumlarda konumlama yüzdesi de hatasız ölçümlerin olduğu durumlara kıyasla düşüş göstermekte.

Şekil 4.2.30, Şekil 4.2.29'da yapılan deneylerin grup bilgisi kullanılarak yapılmış halini gösteriyor. Görmekteyiz ki, hatalı ölçümler ETK'yi de etkilemekte. Ancak ETK'ye yaptığı etki Quadrilateration'a yaptığı etki kadar büyük değil.

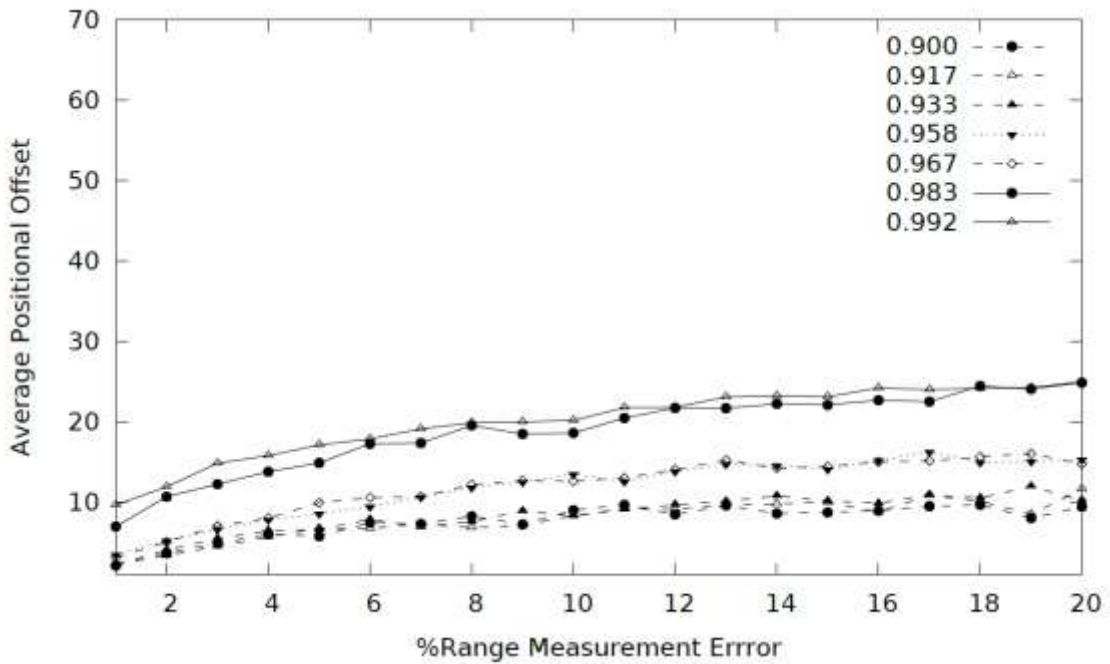


Şekil 4.2.29 Grup bilgisi kullanılmadığında, düzlemsellik katsayısı ve hata oranına bağlı olarak ortaya çıkan konumlama yüzdesi. Menziller buluşma noktası olarak seçilmiştir.



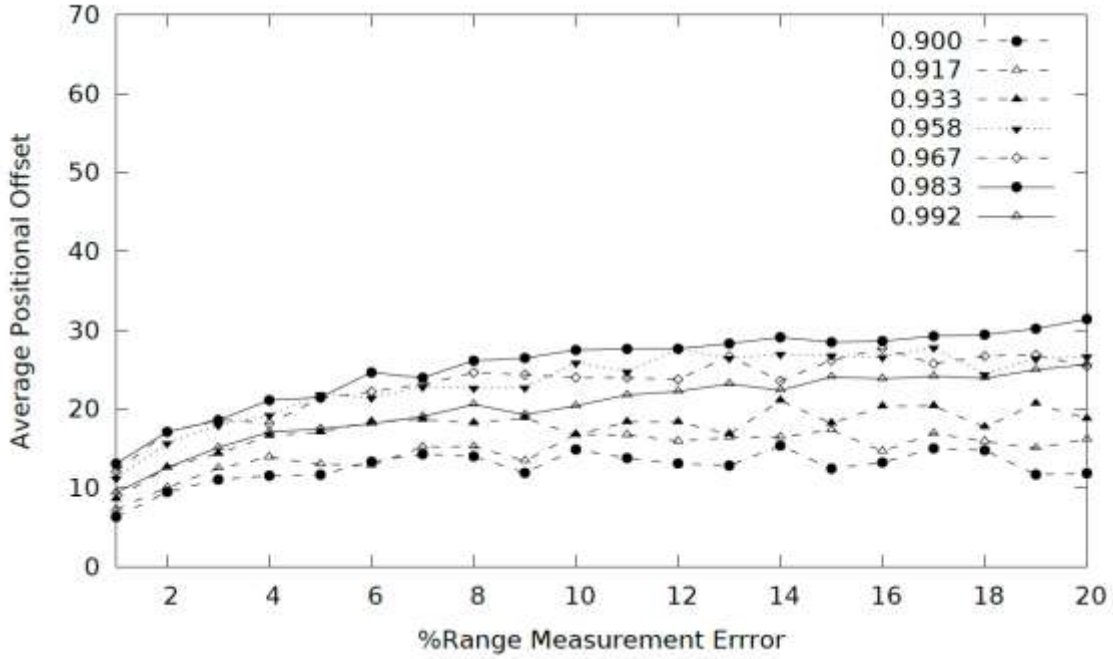
Şekil 4.2.30 Grup bilgisi kullanıldığında, düzlemsellik katsayısı ve hata oranına bağlı olarak ortaya çıkan konumlama yüzdesi. Menziller buluşma noktası olarak seçilmiştir.

Şekil 4.2.31, çeşitli düzlemsellik katsayılarına göre Quadrilateration tarafından kaydedilen ortalama konumsal sapmaların grafiğini gösteriyor.



Şekil 4.2.31 Grup bilgisi kullanılmadığında, düzlemsellik katsayısı ve hata oranına bağlı olarak ortaya çıkan ortalama konumsal sapma. Menziller buluşma noktası olarak seçilmiştir.

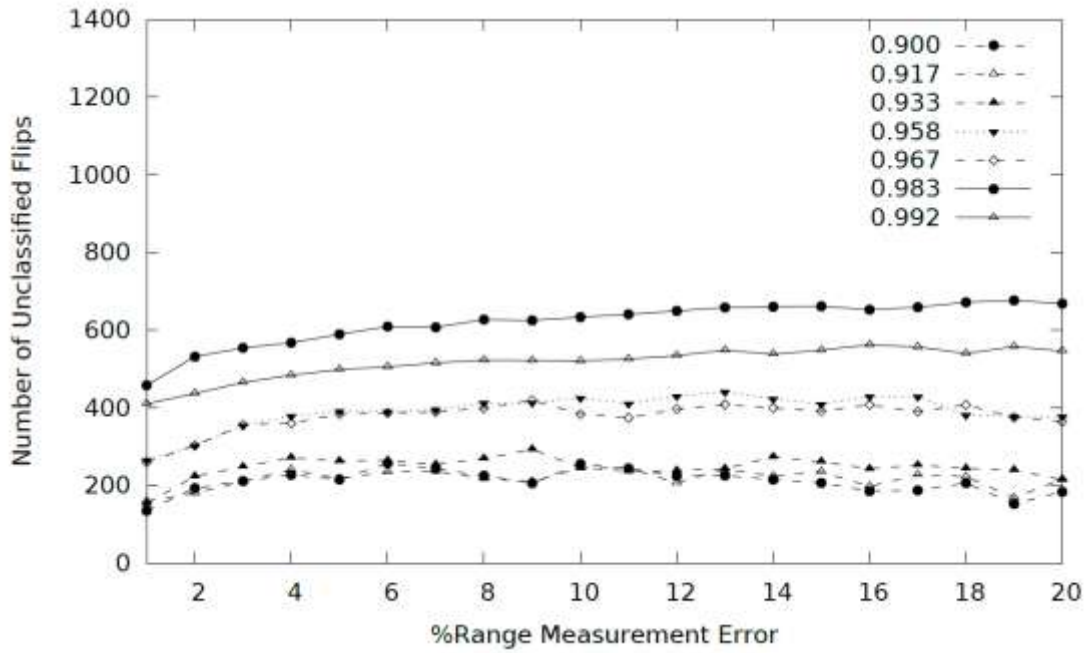
Şekil 4.2.32, grup bilgisi kullanıldığında, ortalama konumsal sapmaların hata oranına göre değişimini gösteriyor. Şekil 4.2.31'de kullandığımız yedi farklı μ değerlerinin aynılarını kullanıyoruz.



Şekil 4.2.32 Grup bilgisi kullanıldığında, düzlemsellik katsayısı ve hata oranına bağlı olarak ortaya çıkan ortalama konumsal sapma. Menziller buluşma noktası olarak seçilmiştir.

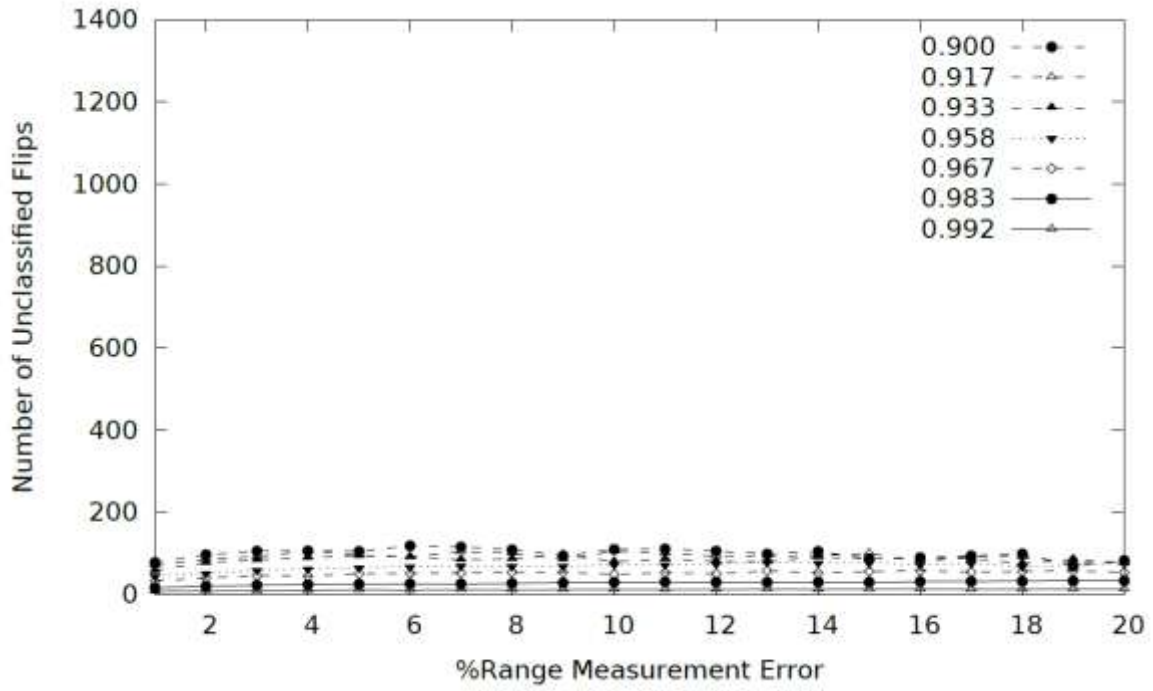
Grup bilgisini kullanmak, daha fazla konumlama yüzdesi elde etmemizi sağlıyor. Öteki yandan, ortalama konumsal kaymalar da daha fazla gözlemleniyor. Sadece bu iki metriğe bakarak, grup bilgisini kullanmanın konumlama kalitesini düşürdüğünü söyleyebiliriz. Ancak, grup bilgisini kullanmamak, tasnif edilmemiş yansımaların sayısında aşırı bir artışa sebep oluyor.

Şekil 4.2.33'te, grup bilgisi kullanılmadığı zaman gözlemlenen tasnif edilmemiş yansıma sayısının grafiğini görüyoruz.

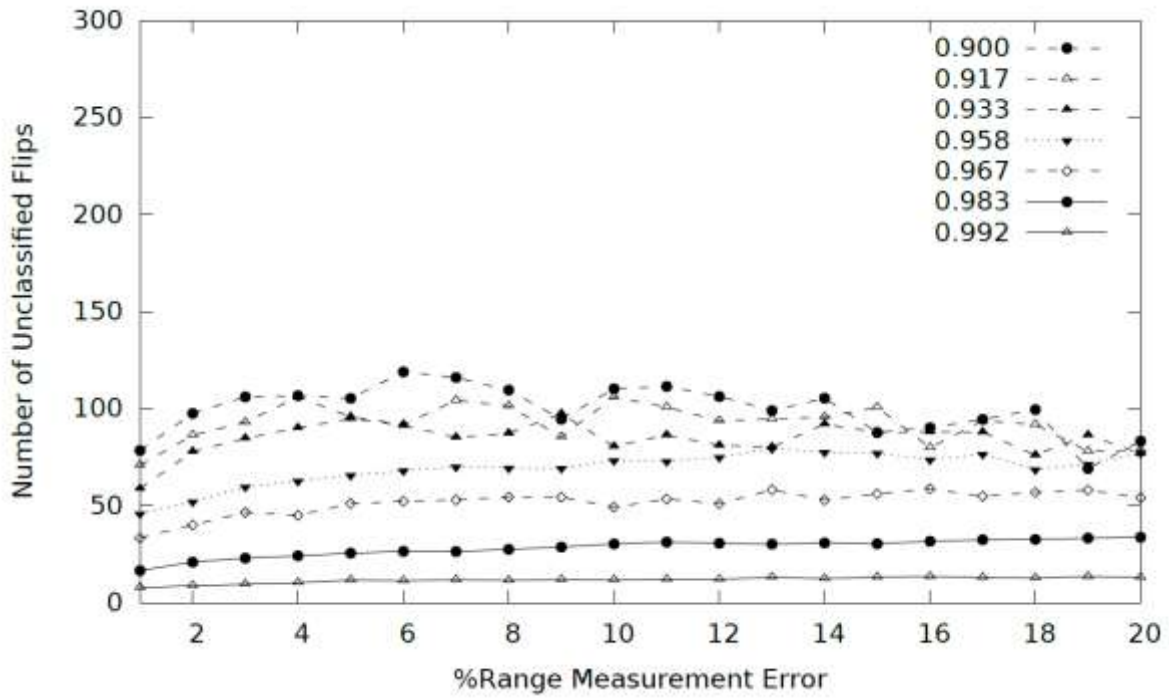


Şekil 4.2.33 Grup bilgisi kullanılmadığında, düzlemsellik katsayısı ve hata oranına bağlı olarak ortaya çıkan tasnif edilmemiş yansıma sayısı. Menziller buluşma noktası olarak seçilmiştir.

Aynı ölçeği kullanarak, grup bilgisini de kullandığımızda, Şekil 4.2.34'ten de görülebileceği üzere, tasnif edilmemiş yansıma sayısı aşırı derecede azalmıştır. Şekil 4.2.34'ü daha iyi yorumlayabilmek için, daha yakınlaştırılmış halini Şekil 4.2.35'te gösteriyoruz. Şekle baktığımızda, ortamın düzlemselliği, tahmin edileceği gibi, tasnif edilmemiş yansımaların sayısına ters orantılı olduğunu anlayabiliriz.



Şekil 4.2.34 Grup bilgisi kullanıldığında, düzlemsellik katsayısı ve hata oranına bağlı olarak ortaya çıkan tasnif edilmemiş yansıma sayısı. Menziller buluşma noktası olarak seçilmiştir.

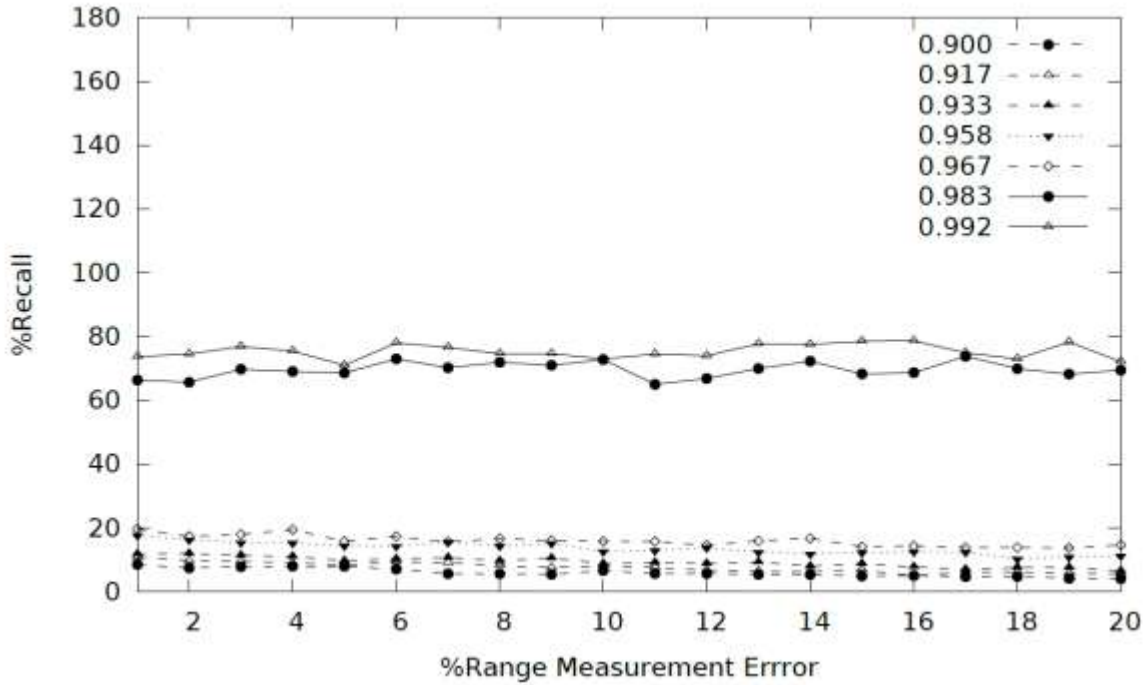


Şekil 4.2.35 Şekil 4.2.34'ün yakınlaştırılmış hali.

Kritik Noktalarla Yapılan Deneyler

Bir düzlemsellik katsayısı için kritik nokta , grup bilgisinin kullanılmasının en faydalı olduğu menzildir. Bu bölümdeki deneylerimiz orta ve yüksek μ değerlerini kapsıyor. Çünkü, düşük μ değerlerinde grup bilgisini kullanmanın faydalı olmadığını hali hazırda hatasız ölçümlerle yaptığımız deneylerde gördük.

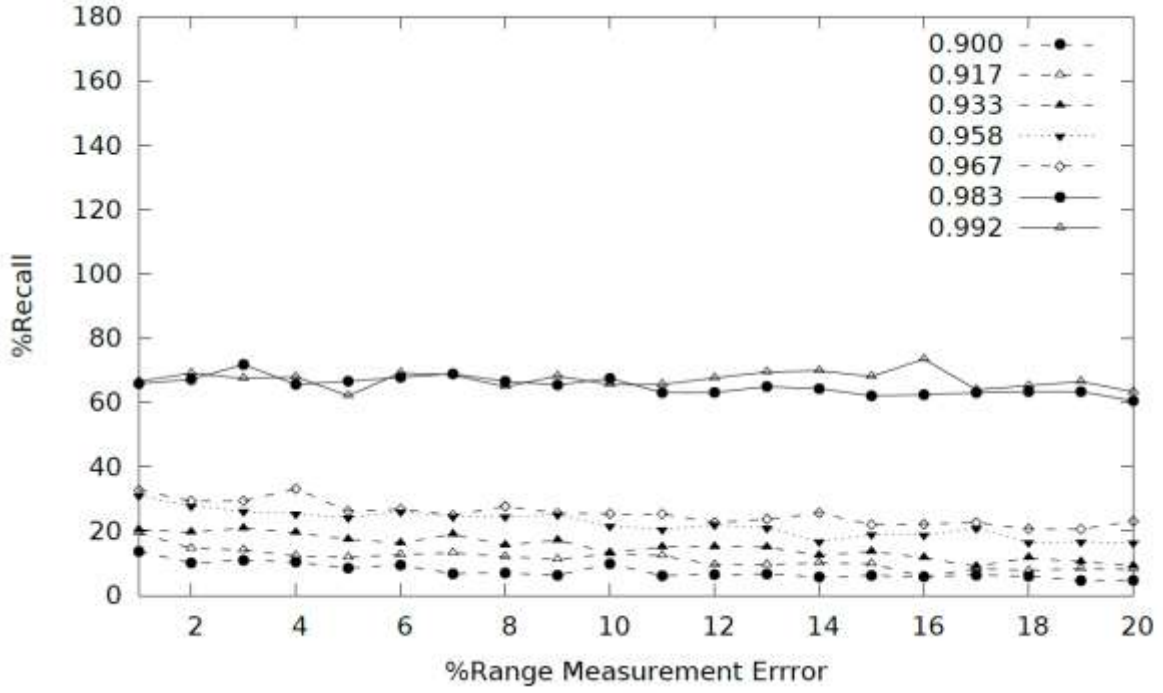
Şekil 4.2.36'da, Quadrilateration'un, hatalı ölçümler olduğu zaman yaptığı konumlama yüzdesini görüyoruz. Hata olmadığı zaman, %20 - %40 bandında olan Quadrilateration performansı, ölçüm değerlerine hata katıldığında %20'nin altında seyrediyor. Önceki deneylerden de bildiğimiz üzere, yüksek düzlemsellik katsayısı, menzil de fazla olduğu için, benzer etki yaratmıyor.



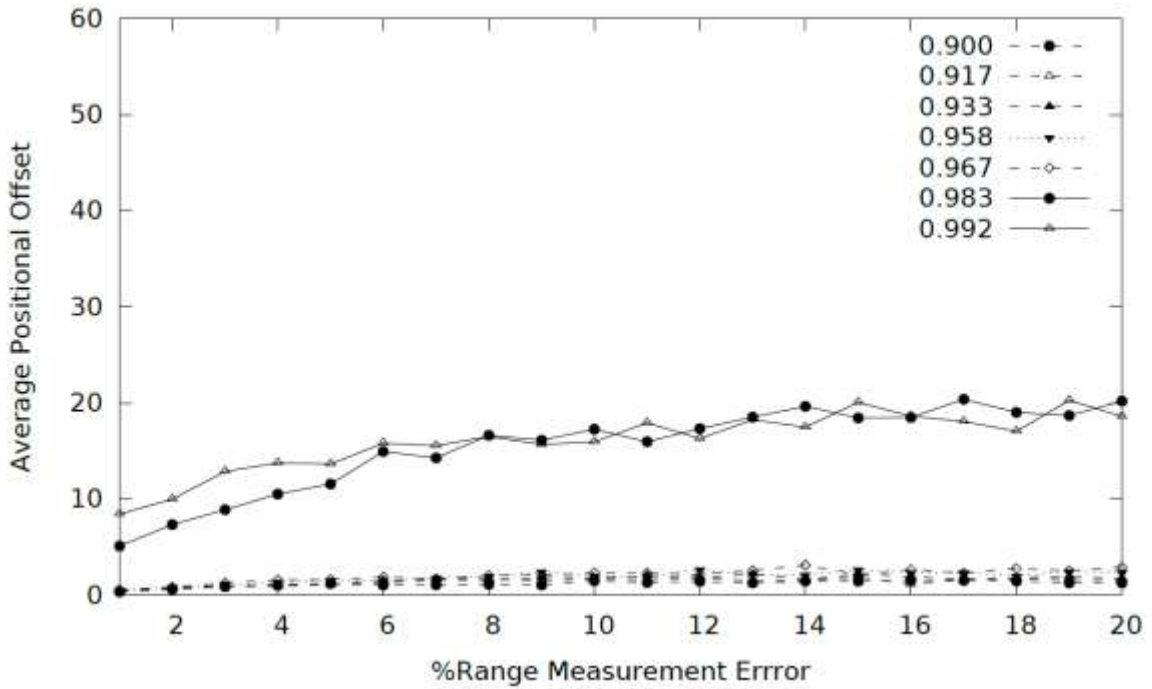
Şekil 4.2.36 Grup bilgisi kullanılmadığında, düzlemsellik katsayısı ve hata oranına bağlı olarak ortaya çıkan ortalama konumlama yüzdesi. Menziller kritik nokta olarak seçilmiştir.

Şekil 4.2.37'de, grup bilgisi kullanılıyor ve ölçüm hataları mevcut. Grafikte Şekil 4.2.36'ya benzer bir durum gözlemliyoruz. Konumlama yüzdesi bir bant alta iniyor ama yüksek μ değerlerinde değişkenlik göstermiyor.

Sıradaki deney grubumuz, hatalı mesafe ölçümleri varken kritik noktalarda grup bilgisi kullanmanın nasıl bir etki yaptığını gösteriyor. Şekil 4.2.38'de, grup bilgisi kullanmadığımız durumlardaki ortamala konumsal sapmaları görüyoruz. Bölüm 8.3.1'deki gibi, daha fazla konumlanan sensör, daha fazla konumsal kayma anlamına geliyor.

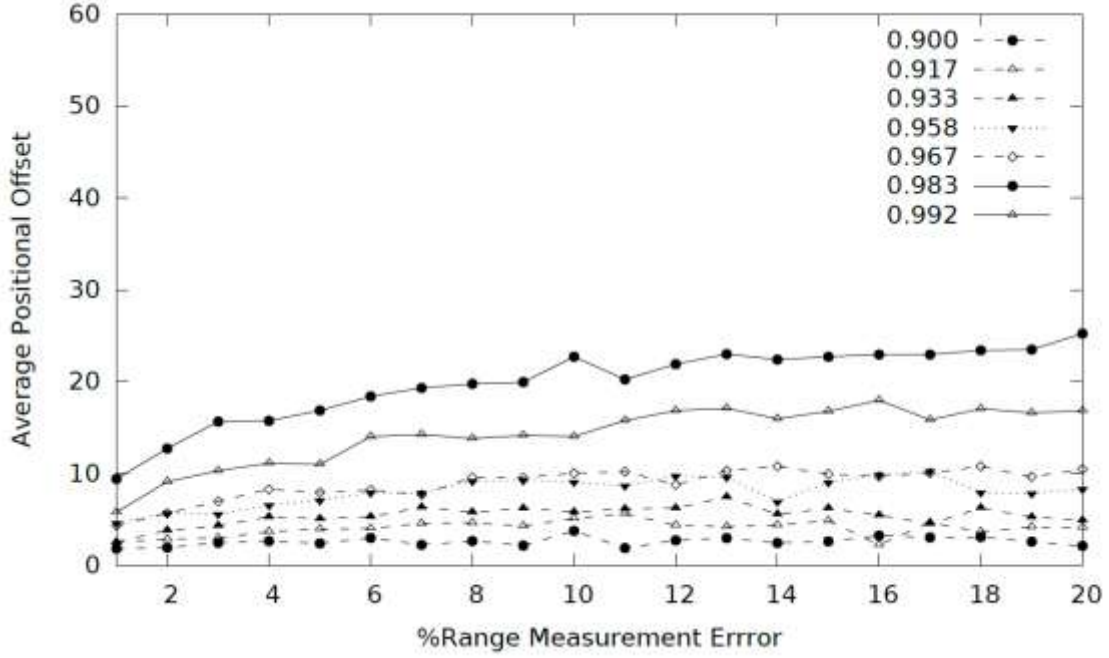


Şekil 4.2.37 Grup bilgisi kullanıldığında, düzlemsellik katsayısı ve hata oranına bağlı olarak ortaya çıkan ortalama konumlama yüzdesi. Menziller kritik nokta olarak seçilmiştir.



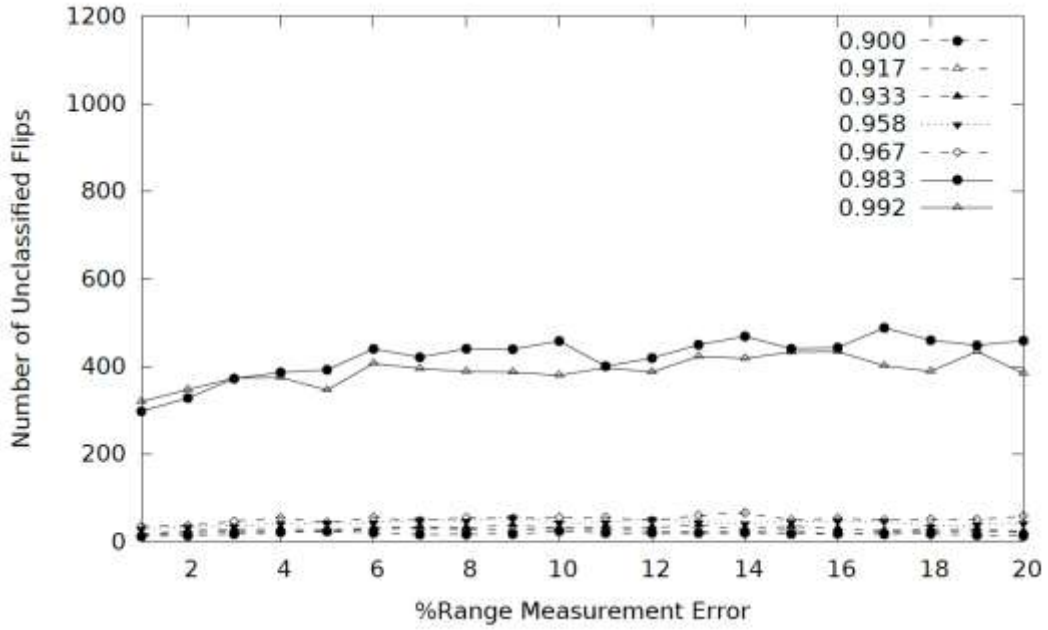
Şekil 4.2.38 Grup bilgisi kullanıldığında, düzlemsellik katsayısı ve hata oranına bağlı olarak ortaya çıkan ortalama konumsal sapma. Menziller kritik nokta olarak seçilmiştir.

Şekil 4.2.39, bize grup bilgisini kullanmanın daha fazla konumsal sapmaya yol açtığını gösteriyor. Ancak, konumlanmanın kalitesi hakkında yargıda bulunmak için sıradaki şekilleri incelemek gerekiyor.

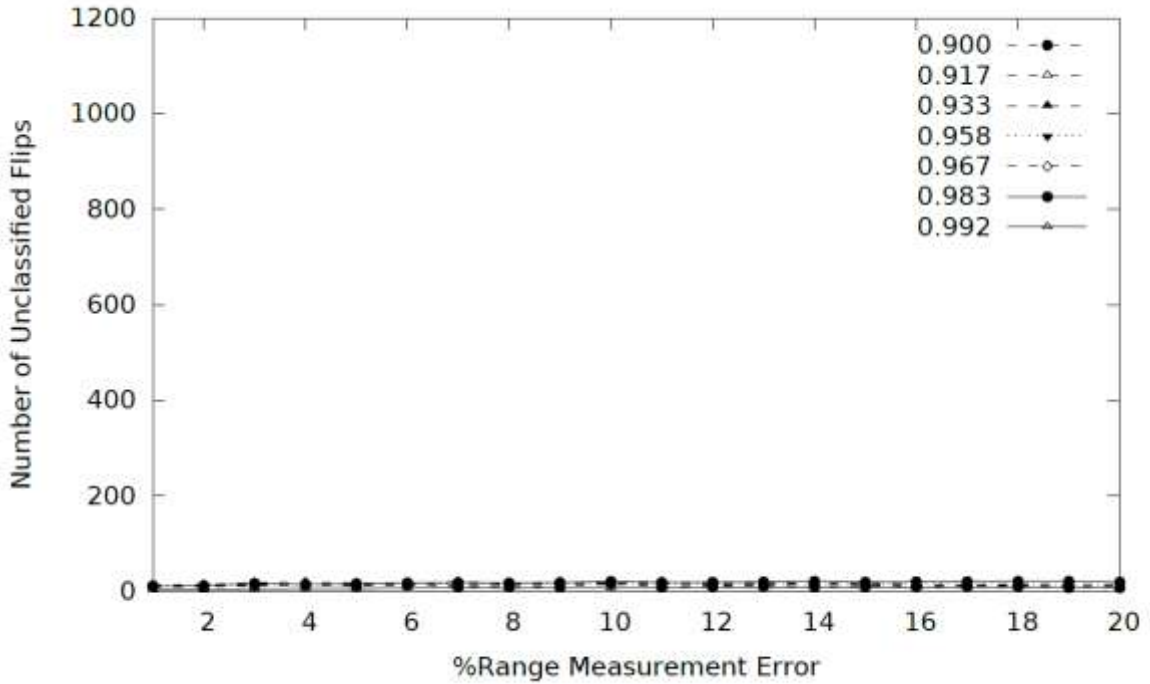


Şekil 4.2.39 Grup bilgisi kullanıldığında, düzlemsellik katsayısı ve hata oranına bağlı olarak ortaya çıkan ortalama konumsal sapma. Menziller kritik nokta olarak seçilmiştir.

Şekil 4.2.40 ile Şekil 4.2.41'i karşılaştırdığımızda görüyoruz ki tasnif edilmemiş yansıma sayısı, grup bilgisi kullanıldığında çok daha az. Özellikle μ değerinin yüksek olduğu durumlarda, düzlemsellik tasnif edilmemiş yansımaları fazlasıyla engelliyor.



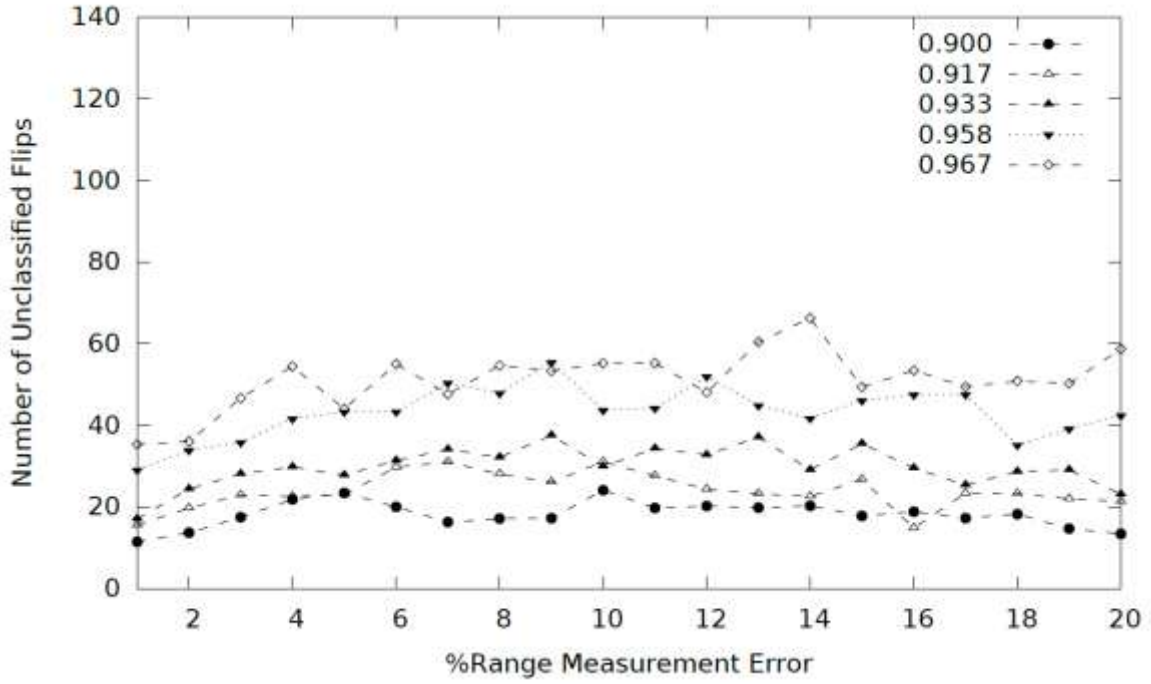
Şekil 4.2.40 Grup bilgisi kullanılmadığında, düzlemsellik katsayısı ve hata oranına bağlı olarak ortaya çıkan tasnif edilmemiş yansıma sayısı. Menziller kritik nokta olarak seçilmiştir.



Şekil 4.2.41 Grup bilgisi kullanılmadığında, düzlemsellik katsayısı ve hata oranına bağlı olarak ortaya çıkan tasnif edilmemiş yansıma sayısı. Menziller kritik nokta olarak seçilmiştir.

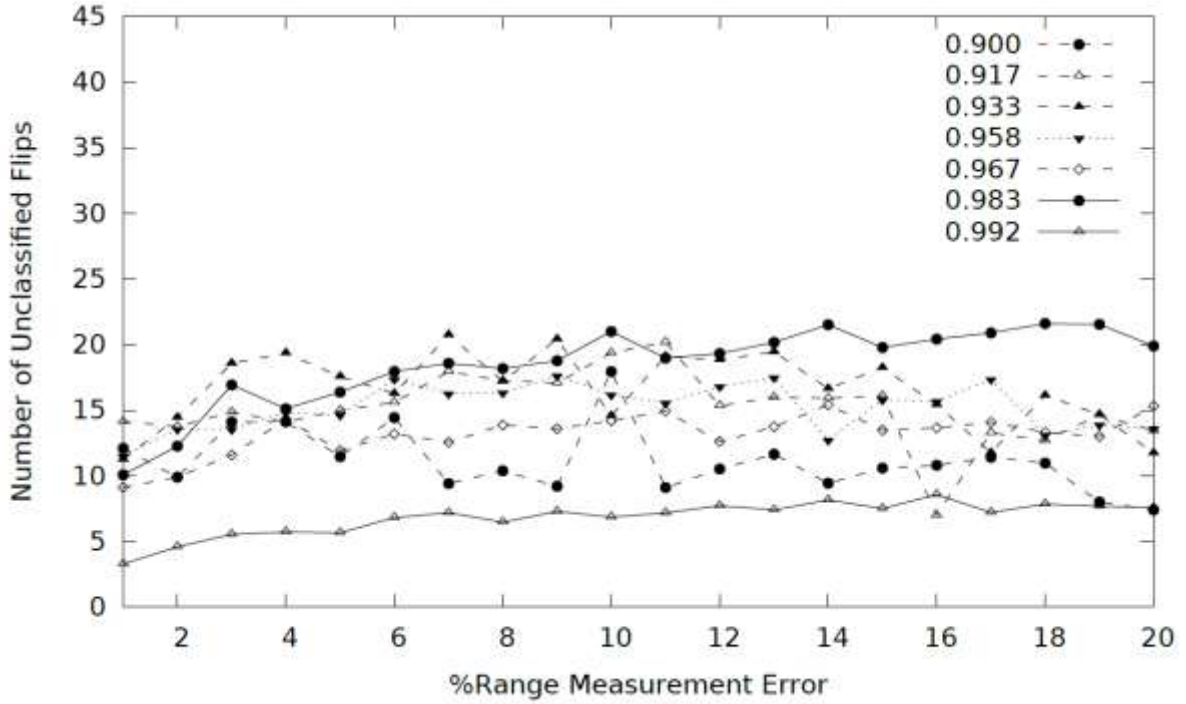
Düzlemsellik katsayısının orta olduğu durumlarda, Quadrilateration'un yaptığı tasnif edilmemiş yansımalar 60'ın altında seyrederken, bu değer ETK için her durumda 23'ü

aşmamaktadır. Bu durumu daha iyi inceleyebilmek için Şekil 4.2.42'de ve Şekil 4.2.43'te, önceki iki grafiğin yakınlaştırılmış hallerini gösteriyoruz.



Şekil 4.2.42 Şekil 4.2.40'in sadece orta μ değerleri için yakınlaştırılmış hali.

Şekil 4.2.43, grup bilgisi kullanılmadığında ortaya çıkan tasnif edilmemiş yansımaların sayısını gösteriyor.



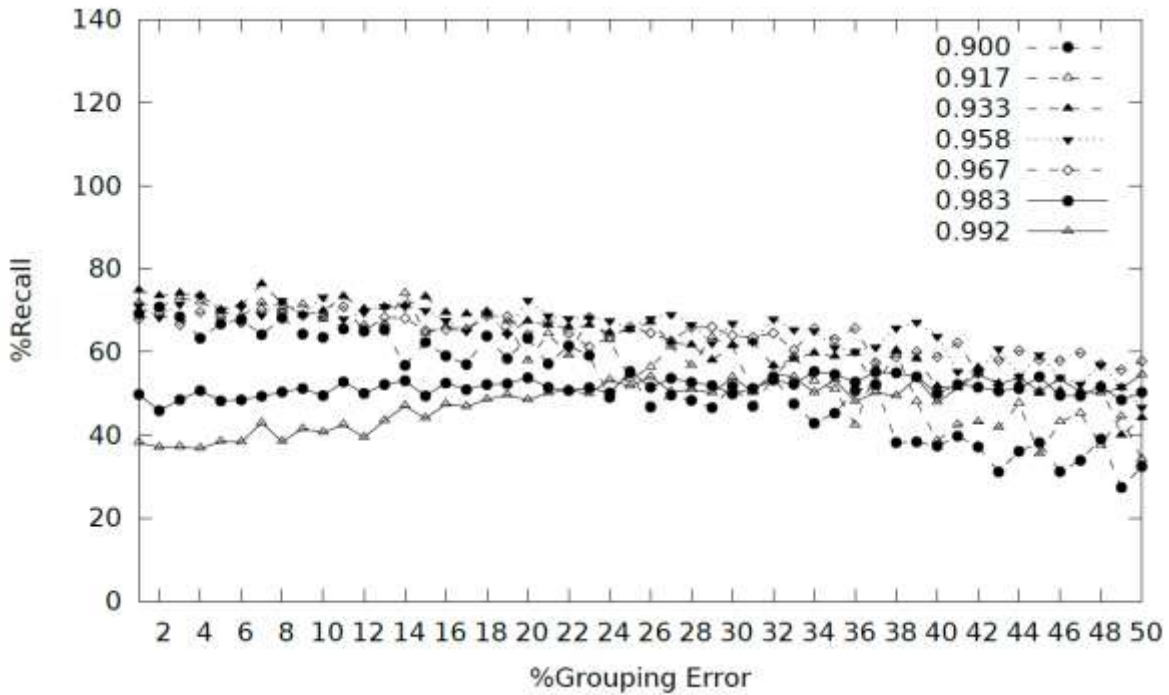
Şekil 4.2.43 Şekil 4.2.41'in tüm μ değerleri için yakınlaştırılmış hali.

Hatalı Grup Bilgileriyle Yapılan Deneyler

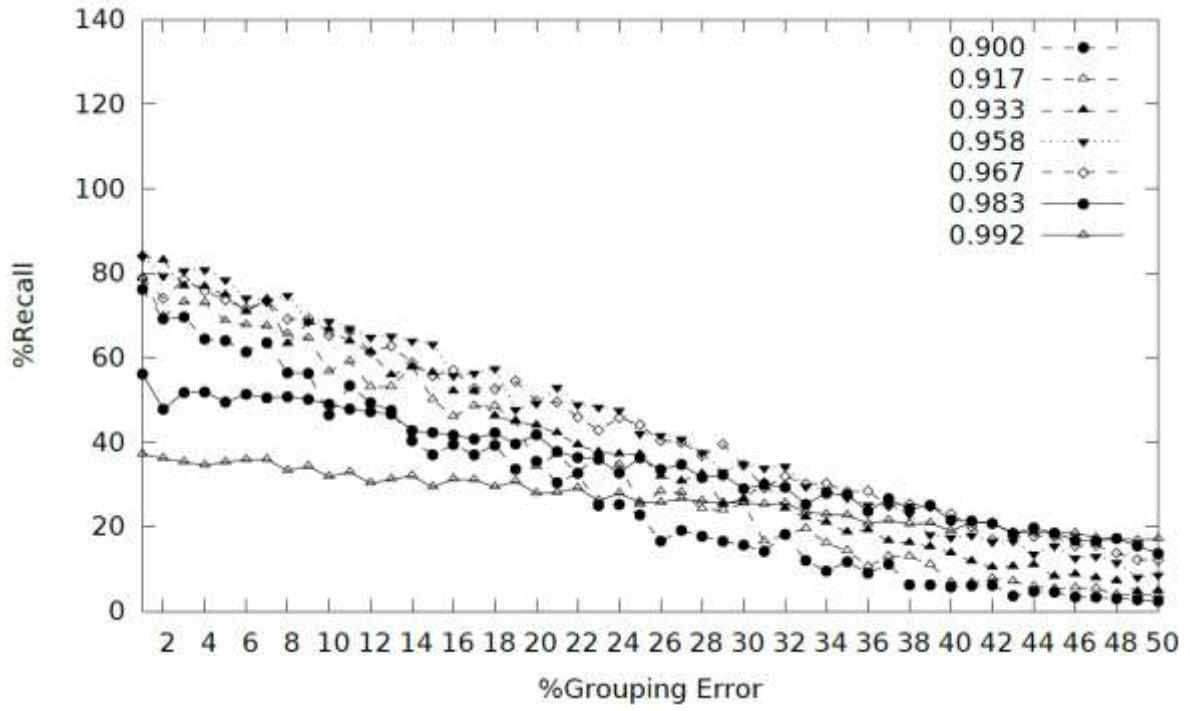
Bu bölümde, EDK'nin grup bilgileri hatalı olduğu zaman nasıl davrandığını inceliyoruz.

Şekil 4.2.28'e dayanarak, algılama menzilini 18 birim seçiyoruz. Hatalı ölçümler söz konusu olduğunda ETK'nin konumlama yüzdesinin bir parça düşmesine rağmen (bkz. Şekil 4.2.37), 18 birimin özelliği, her düzlemsellik katsayısı için en yüksek sayıda tasnif edilmemiş yansımayı vermesidir.

Akcan vd. (2011) yaptığı gibi, sensörlerin hata payını $\pm\%1$ olarak seçtik. Şekil 4.2.44'te, grup bilgisinin hatalı olmasının Quadriateration üzerindeki etkisini görüyoruz. Şekil 4.2.45'te, grup bilgisinin hatalı olmasının ETK üzerindeki etkisini görüyoruz. Beklendiği gibi, düzlemdışı noktaların sayısının artışı, grup bilgisini gitgide kullanışsız hale getiriyor. Grup bilgisini kullanmamızın gereksiz olduğu nokta %16 grupta hata. Bu noktada, hem Quadriateration hem de ETK %60 civarında konumlama yapabiliyor. Bu noktadan sonrasında Quadriateration 'un performansı çok değişmezken, ETK'nin performansında büyük ölçüde düşüş gözlemlenmekte.



Şekil 4.2.44 Grup bilgisi kullanılmadığında konumlama yüzdesinin hatalı grupta ve çeşitli düzlemsellik katsayılarına bağlı olarak değişimi. Sensör menzili 18 birim ve hata oranı $\pm\%1$ olarak alınmıştır.



Şekil 4.2.45 Grup bilgisi kullanıldığında konumlama yüzdesinin hatalı gruplama ve çeşitli düzlemsellik katsayılarına bağlı olarak değişimi. Sensör menzili 18 birim ve hata oranı $\pm\%1$ olarak alınmıştır.

4.3 Mobil elemanların olduđu durumlarda konumlama

4.3.1 Giriş

Konum tespiti, n boyutlu düzlemlerdeki kablosuz algılayıcı birimlerin, kablosuz algılayıcı ağlardaki (KAA) tam pozisyonlarını saptama işlemi için kullanılan genel bir terimdir. Konum tespiti işlemi için en yaygın seçim GPS olsa da, bu tüneller ve kapalı alanlar gibi spesifik senaryolar için mümkün olmayabilir. Ayrıca, hassas yer bilgisine ihtiyaç duyan senaryolar için GPS'in sunduđu veri yeterli gelmeyebilir. Dış mekanlarda geçen senaryolar için bile GPS, sistemdeki ekstra bir maliyettir. Biz bir çok GPS'siz konum tespiti tekniđi arasından *uzaklık tabanlı konum tespitini* tercih ettik, Hightower vd. (2001).

Uzaklık tabanlı konum tespiti, bize düğümlerinin çeşitli mesafe tayini algoritmalarıyla komşuları arasındaki mesafeleri ölçülen algılayıcı birimleri temsil ettiđi bir çizge verir. Lakin, sadece mesafe tayini metodları kullanmak, durumu çizge algılama problemine çevirir ki bu da NP-complete'tir Yemini (1979).

Bu çalışmada, hareketli ve sabit birimler, çok katlı binalarda kullanılacaktır. Çok katlı binalar probleme bir kısıt olarak sunulmuştur. Bu kısıtın sebebi, üç boyutlu bir ortamı, iki boyutlu düzlemlere bödererek konum tespitini daha kolay hale getirmektir. Bir mobil birim, ortamda gezerek algılayıcı ağı saptayacak ve gerektiđi yerlerde müdahaleler yaparak rijid bir ağ yapısı

elde etmeye çalışacaktır. Mobil birim, bu algılayıcı ağlardaki her birimi düzlemlerine göre gruplayacaktır. Mobil birimin, ortam dahilindeki belli sınırları geçmeyeceđi varsayılmıştır.

KAA'da konum tespiti gerektiren gerçek senaryolar; mevcut duruma, çevreye ve donanıma göre deđişiklik gösterir. Bu durumlar için verilebilecek örneklerden bazıları şu şekildedir:

- Yangın fırtına deprem gibi felaket durumlarında; itfaiyecilerin, kurtarma ekibinin veya binada mahsur kalmış sivillerin kesin pozisyonlarının konum tespiti.
- Büyük, çok katlı depolardaki belirli malların konum tespiti.
- İnsanlar için uygun olmayan alanlarda, algılayıcı birimler vasıtasıyla gerekli ölçüm deđerlerini elde etme.

Bu gibi senaryolarda eđer KAA'daki bütün birimler sabit ise, her zaman konum tespiti yapmak mümkün deđildir. Algılayıcı birimler her zaman 100% hassaslıkta ölçüm yapamaz ve bu durum, tutarsızlıklara sebebiyet verir. Bu sebepten, konum tespitine yardımcı olması için hareketli birim kullanmak, makul bir seçim olacaktır.

Bu çalışmanın literatüre temel katkısı, uzaklık tabanlı konum tespiti problemini daha kolay bir hale getirmek ve KAA'daki her sabit birimi, tek bir mobil birimin yardımıyla düzlemlerine göre gruplamaktır. Daha spesifik olarak, sunduğumuz algoritma:

- üç boyutlu bir ağı, birimlerinin buldukları düzlemlere göre iki boyutlu ağlara bödecek,
- elde edilen iki boyutlu ağların rijid olmalarını sağlayacak,
- rijiditeyi bozan birimlere müdahale edecek,
- her boyuttan algılayıcı ağda çalışacak,
- hatalı ölçümler ile çalışabilecek,
- ekstra altyapı gerektirmeyecek.

Çalışmamızı kendi tasarladığımız simülasyonda yaptığımız deneyler ile destekledik. Analizlerimiz temel olarak, hatalı ölçümlerle; en kötü şartlardaki hareket miktarı, düzlemlerin dağılımı ve rijidite bozulmalarının onarılmasının başarı oranı üzerinedir.

4.3.2 İlgili Çalışmalar

Kablosuz algılayıcı ağlar üzerine yapılmış önceki çalışmalar genel olarak sabit algılayıcı ağlar üzerinde duruyor (Akcan vd. 2011, Olson vd. 2006). Yine de, tamamen ya da kısmen hareketli birimlerden oluşan ağlarla ilgili çalışmalar da mevcut (Park vd. 2008; Akcan vd. 2010; Priyantha vd. 2005; Djughash vd. 2006; Dantu vd. 2005).

Başlangıçtaki ağ yapısını, konum tespiti açısından güçlendirmenin ve/veya başlangıçta konum tespitini engelleyen faktörleri onarmanın önemi, çeşitli çalışmalarda belirtilmiştir (Priyantha vd. 2005; Dantu vd. 2005), GPS'siz konum tespiti üzerine, sadece uzaklık ölçülerini kullanan ve mobil birimler ve sanal birimlerin yardımıyla, ağ yapısını daha rijid bir hale getirmeye çalışan bir çalışma. Djughash vd. 2006, da sanal birimlerinden yardım alıyor olsa da, daha ziyade haritalama ve SLAM algoritması duruyor. Bu çalışma ayrıca fiziksel engelleri saptamak için bir sonik cihazı kullanıyor, Akcan vd. 2010, her algılayıcı birimin bir pusulasının olduğu ve her birimin bağımsız olarak hareket edebildiği bir ortamda çalışan GDL algoritmasının sunumunu yapıyor. Aynı çalışmadaki GCDL algoritması, GDL algoritmasının pusulasız versiyonu fakat onda da işlemin uygulanabilmesi için ekstra hareketlere ihtiyaç duyuluyor.

4.3.3 HexMove Yaklaşımı

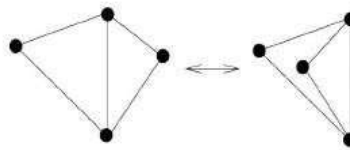
HexMove'un ana fikri, eldeki üç boyutlu algılayıcı ağdaki birimleri saptamak, ağ yapısındaki rijiditeyi bozan birimleri onarmak ve bütün birimleri düzlemlerine göre gruplayarak, bu üç boyutlu ağ yapısını iki boyutlu parçalara bölmektir. Bu işlemler, hareketli bir birimin yardımıyla yapılacaktır. Bu hareketli birim, kendisini her zaman eşkenar heksagonlara bönmüş iki boyutlu bir ortamda varsayar.

Bu bölümde, HexMove yaklaşımının detayları ve gerekli ön bilgileri açıklanacaktır.

Ön Bilgiler

Bu bölümde, algoritmamız tarafından kullanılan bazı konseptler kısaca açıklanacaktır.

Ağ Konum Tespiti ve Rijidite: *Ağ konum tespiti*, bir ağdaki bütün birimlerin birbirlerine olan konumlarının tespit edilmesi operasyonuna verilen isimdir. Bütün algılayıcı birimler, belirli bir radyo haberleşme mesafesi limitine sahiptir ki bu limite de *birim menzili* denir. Algılayıcı birimler sadece kendi menzillerindeki diğer birimleri saptayabilir. GPS'siz konum tespitinde, en az üç adet yerleri bilinen ve ko-lineer olmayan birime komşu olan bir algılayıcı birim, diğer birimlere göre kendi yerini konumlandırabilir. Eğer birbirlerini gören birimleri kağıt üzerinde birleştirirsek bir çizge yapısı elde ederiz.



Şekil 4.3.1 Flip Belirsizliği

Eğer bir çizge yapısı, *jenerik global rijid* ise, konum tespiti açısından çözülebilir olduğu kanıtlanmıştır (Connely 2005; Aspnes vd. 2006). Lakin, sabit birimler her zaman rijid bir yapı oluşturmazlar. İki boyutlu konum tespitinde, eğer çizge yapısı jenerik global rijid değil ise, bazı birimler kendilerini iki farklı noktada konumlandırabilirler. Bu belirsiz duruma, *flip belirsizliği* diyoruz (Şekil 4.3.1) Goldenberg vd. (2005).

Cayley-Menger Determinant: Komşu birimlerin ko-lineerlik kontrolü için, Cayley-Menger determinant kullanıyoruz:

A, B ve C kablosuz algılayıcı ağdaki sabit birimler olsun. $d(AB)$, A ve B arasındaki uzaklık diyelim.

Herhangi bir A, B ve C'nin ko-lineer olduğunu söyleyebilmemiz için ancak ve ancak:

$$\det \begin{bmatrix} 0 & d(AB)^2 & d(AC)^2 & 1 \\ d(AB)^2 & 0 & d(BC)^2 & 1 \\ d(AC)^2 & d(BC)^2 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix} = 0$$

Cayley-Menger determinantı hangi sabit birimin hangi düzleme ait olduğunu tespit etmek için de kullanıyoruz. Bir sabit birimin belli bir düzleme ait olup olmadığını tespit edebilmek için, o düzlemin üzerinde olan en az üç adet komşu sabit birime daha ihtiyaç vardır. Bu sabit birimler, birbirlerine olan uzaklıklarını da biliyor olmalıdırlar.

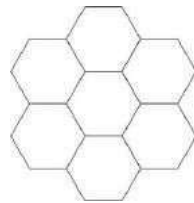
A, B, C ve D kablosuz algılayıcı ağdaki birimler olsun. $d(AB)$ A ve B arasındaki uzaklık diyelim.

A, B, C ve D, düzlemde olduğunu söyleyebilmemiz için ancak ve ancak:

$$\det \begin{bmatrix} 0 & d(AB)^2 & d(AC)^2 & d(AD)^2 & 1 \\ d(AB)^2 & 0 & d(BC)^2 & d(BD)^2 & 1 \\ d(AC)^2 & d(BC)^2 & 0 & d(CD)^2 & 1 \\ d(AD)^2 & d(BD)^2 & d(CD)^2 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 \end{bmatrix} = 0$$

Konsept

Bu alt bölümde, algoritmamızın temel konseptleri detaylandırılacaktır.



Şekil 4.3.2 Hex Çember

Terminoloji: Bu çalışmada, hareketli birim, kendisini eşkenar altıgenlere bölünmüş bir haritada varsayar. Bu haritada, çember şeklinde birbirine komşu olan heksagonların oluşturduğu yapıya, *Hex Çember* adını verdik (Şekil 4.3.2).

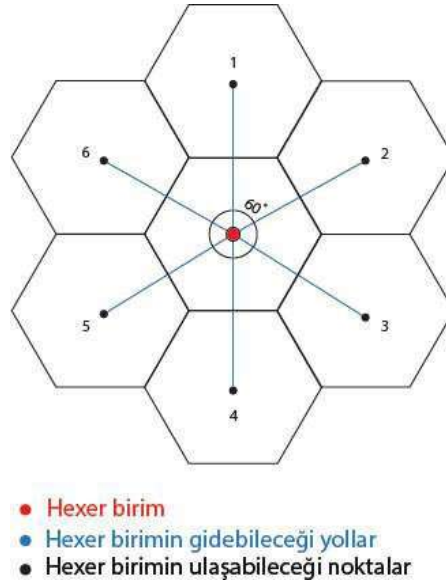
Hexer birim, her zaman bir heksagonun merkezinden diğerine hareket eder. Mobil birime, bu heksagon tabanlı hareket yapısından ötürü *hexer birim* adını verdik. Bu merkezden merkeze hareketi sağlamak için, hexer birimin iki heksagon arasında olan uzaklığı bilmesi gerekir ki bu uzaklığa da *hexadım* diyoruz. Hexadım, hexer birimin radyo menziline göre hesaplanan sabit ve bilinen bir değerdir.

Eğer bir birimin düzlemi hexer birim tarafından saptanmışsa, o birim *gruplanmış birim* olarak işaretlenir ve eğer bir birim flip belirsizliğine yol açıyorsa, *flip birim* olarak işaretlenir. Birimlerin konumları bilinmediğinden, hexer birim, sabit bir birimi gruplamak ya da flip bir birimi onarmak için uzaklık ölçümlerine ihtiyaç duyar. Cayley-Menger determinantını kullanabilmek için en az üç noktanın birbirleri arasındaki uzaklık ölçümlerinin karelerine ihtiyaç vardır. Bu sebepten, hexer birim, sabit bir birim üzerinde işlem yapabilmek için farklı heksagon merkez noktalarından da ölçüm alması gerekir. Bu ekstra noktalar, o anki işlem bitene kadar hexer birimin hafızasında sanki sabit bir birimlermiş gibi kayıtlı durur ve *sanal birim* adını alır.

Hexer Birim Özellikleri:

Hexer birimin hareket kabiliyeti, bu çalışmanın önemli bir parçasıdır. Hexer birimin hız/saniye değerini bildiğimizi varsayarsak, istenilen bir noktaya gitmesi için, o noktanın doğrultusunda kaç saniye boyunca ilerlenmesi gerektiğini hesaplayabiliriz. Hexer birim, sadece bir heksagonun merkezinden bir başka heksagonun merkezine hareket edebilir. Bu sebepten, gidebileceği toplamda altı yön vardır. Yani, hexer birim, her sıralı dönüş hareketinde, arzu edilen yönde 60° kadar dönüş hareketi gerçekleştirir.

$D = \{ d_1, d_2, d_3, d_4, d_5, d_6 \}$, hexer birimin yönelebileceği altı yön kümesi olsun. $r_c = a$ ve $r_t = b$ olsun. r_c hexer birimin şu anki yönü, r_t ise hexer birimin hedeflenen yönü olsun ve $\{a, b\} \subset D$ olsun. Eğer $(b-a) > 0$ ise hexer birim dönüş hareketini saat yönünde yapacaktır. Eğer $(b-a) < 0$ ise hexer birim dönüş hareketini saat yönünün tersi yönünde yapacaktır. $\alpha = (|b-a| \times 60)$ olsun α , hexer birimin döneceği toplam açı miktarıdır (Şekil 4.3.3).



Şekil 4.3.3 Hexer birim, hex çemberi dahilindeki altı olası noktaya ulaşabilir

Maalesef, algılayıcı birimler her zaman net doğrulukta ölçüm yapamayabilirler. Bu durum, sabit birimler için belirsizliklere yol açabilir. Hareketli birim de bu durumdan zarar görür ve bu durum sadece uzaklık ölçümleri için değildir, aynı zamanda birim/zaman ve/veya dönüş derecesi hesaplamaları da hatalı olabilir. Bu yanlış ölçümler, tek seferlik durumlar için küçük ve ihmal edilebilir hatalardır lakin her adımda, bu hatalar birbirleri üzerine eklenir ve artımsal hatalara dönüşür. Artımsal hatalar sebebiyle, hareketli birimin uzun vadede bulunduğu koordinatlar değersizdir. Bu sebepten, koordinat bilgilerini sadece kısa süreliğine ve geçici olarak tutarlar. Artımsal hataları önlemek için, hexer birim sadece kendisine göre belirlediği lokal koordinat bilgisini kullanır ki, bu bilgi flip onarma ya da gruplama gibi en fazla altı heksadım sürecek işlemler için kullanılır. Söz konusu işlem tamamlandıktan sonra da, lokal koordinat bilgisi ortadan kaldırılır.

Eğer hexer birim, gruplanmamış bir sabit birim ya da bir fliper birim saptarsa, her zaman o sabit birimin etrafında bir hex çemberi çizerek kolineer olmayan sanal birimler bırakır ve o birimin düzlemini saptamaya çalışır. Hexer birim sadece bir heksagon merkezinden diğerine hareket ettiği için, sanal birimleri de sadece heksagon merkezlerine bırakır. Bu yüzden, gruplanmamış ya da fliper bir birimi bir heksagonun herhangi bir noktasına bırakırsak, hexer birimin o noktayı görecek ve sanal birim bırakacak en az üç pozisyona daha gidebiliyor olması gerekir.

Hexer birimin menzilinı efektif bir Őekilde ayarlamak iŐin, sanal birim bırakacak  geŐerli nokta bulma gerekliliŐini gz nnde bulundurmamız gerekir.

Bu gereksinimler iŐıŐında;

1) Hexer birimin menzil alanı (r yi hexer birimin maksimum algı menzili olduĐunu varsayarsak r yarıŐaplı ember) en az bir altıgenden byk olmalıdır ki, o altıgen dahilindeki en az bir statik birimi tespit edebilsin.

2) Bir hex emberinde, yeri tespit edilecek birimin ortadaki altıgende olduĐunu varsayarsak, hexer birimin ortadaki altıgene komŐu olan altıgenlerin en az ikisinden daha sz konusu birimi grebiliyor olması lazım ki gerekli sanalr birimleri yerleŐtirebilsin. Lakin ikiden fazlasına da ihtiya  yok zira iki boyutlu trilaterasyon, yer tespiti iŐin sadece  noktaya ihtiya  duyar.

Bu sebepten, hexer birimin algı menzili, en az iki heksagonun alanından daha byk olmalıdır:

k , eŐkenar bir altıgenin bir kenar uzunluĐu olsun ve r hexer birimin algı menzilinın oluŐturduĐu emberin yarıŐapı olsun;

$$\begin{aligned} \frac{3\sqrt{3}k^2}{2} &\Rightarrow \text{bir altıgenin alanı} \\ \pi r^2 &\Rightarrow \text{bir emberin alanı} \end{aligned} \quad (1)$$

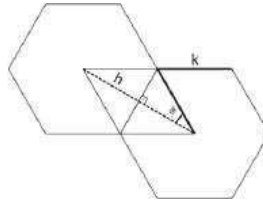
Daha nceden de belirtildiĐi gibi, hexer birimin algı menzili en az iki altıgeni kaplayacak kadar byk olmalı:

$$2 \frac{3\sqrt{3}k^2}{2} \leq \pi r^2$$

(2)

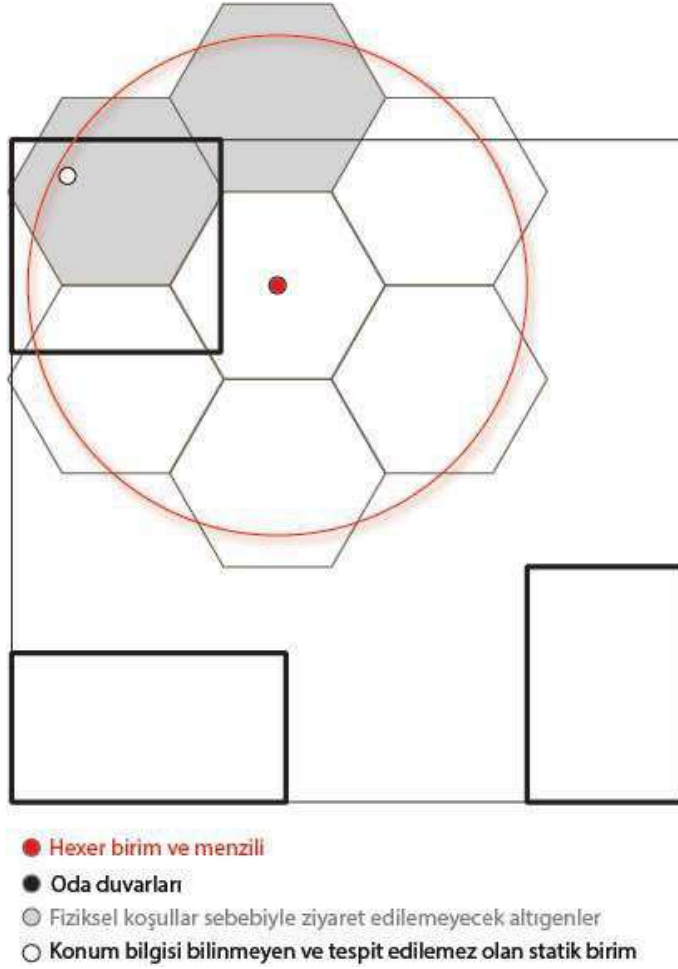
$$\Rightarrow k \leq \frac{\sqrt{\sqrt{3}\pi}}{3} r$$

Denklem (1) e göre, bir heksagonun kenar uzunluğu bilinmektedir.



Şekil 4.3.4 “Pisagor Teoremi” baz alınarak, iki eşkenar heksagonun merkezi arasındaki uzaklık kolaylıkla bulunabilir.

Daha önce de bahsedildiği gibi, kullandığımız altyapının sınırları mevcut. Hexer birimin, kendisi ve bu sınırlar arasındaki mesafeyi ölçebildiğini varsayıyoruz. Bu sayede, hexer birim, gideceği bir sonraki heksagonun uygun olup olmadığını kontrol edebiliyor. Eğer heksagon sınırların dışında ise, hexer birim, sonraki adımının tamamlanabileceğine emin olana kadar hexadım uzunluğunu yarıya indiriyor. Bu işlemin ardındaki amaç, hexer birimin gezebileceği alanı arttırmaktır. Aksi takdirde, özellikle hexer birimin hexadım uzunluğunun yüksek olduğu senaryolarda, hexer birim haddinden fazla sınırlanacak ve hareket kabiliyeti kısıtlanacaktır.



Şekil 4.2.5 Hexer birimin, trilaterasyon yapabilmesi için gri bölgeye ulaşır, sanal birimler bırakması lazım.

Şekil 4.2.5'te görüldüğü üzere, hexer birim gruplanmamış bir birim saptamış. Bu birimi gruplamak bulmak için, hexer birimin gri alanlardan en az birine bir sanal birim bırakabilmesi gerekmektedir ve bu fiziksel engellerden dolayı mümkün olmadığından, bu statik birim "gruplanamaz" olarak işaretlenir.

Eğer bir sabit birim halihazırda gruplanmışsa ve fliper değilse, hexer birim, o birimin varlığını yoksayar.

Rasgele Hareket ve Puanlı Hareket: Rasgele Hareket ve Puanlı Hareket, HexMove algoritmasının, hexer birimin bir sonraki hexadında hangi heksagona gideceğini hesaplayan hareket alt algoritmalarıdır. "Rasgele Hareket" yalnızca 1 ve 6 arası rasgele bir numara yaratır ve ona göre sonraki heksagonu seçer. "Puanlı

Hareket”in de rasgele bir altyapısı olmasına rağmen, daha deterministik bir yaklaşım sunar.

“Puanlı Hareket”te (Algoritma 1), hexer birim, her adımda, gidebileceği tüm komşu heksagonlara 1 ile 4 arası rasgele bir puan dağıtır. Daha sonra, en büyük puanın hangi heksagonda olduğunu kontrol eder ve ona doğru ilerler. Eğer gideceği yön engelli ise, hexadım uzunluğuyla beraber, o yönün puanını da yarıya indirir. Eğer hexer birim gruplanmamış bir sabit birim saptarsa, diğer birimlerin de o birime yakın gruplandığını düşünerek, gittiği yönün puanını 6 artırır. Her hexadımda, 0’ın üzerindeki puanlar 1 azaltılır.

Algoritma 1 Puanlı Hareket

```
1: siradakiYon ← en yüksek puana sahip yönü
seç 2: if (seçili heksagon sınır dışındaysa) then
3:    $yonPuanı[siradakiYon] ← yonPuanı[siradakiYon]/2$  4: end if
5: if (sabit birim saptanmışsa) then
6:    $yonPuanı[siradakiYon] ← yonPuanı[siradakiYon] + 6$  7: end if
8: for (puana sahip her yön > 0) do
9:    $yonPuanı[yon] ← yonPuanı[yon] - 1$  10: end for
11:  $rsg ← rasgeleSayı[0-5]$ 
12:  $yonPuanı[rsg] ← yonPuanı[rsg] + rasgeleSayı[1-4]$ 
```

Algoritma

HexMove algoritmasında, (Algoritma 2) ilk olarak hexer birim, bir hex çember yaratır ve kendisini o hex çemberin ortasında varsayar. Gerçek koordinatlar bilinmediğinden, hexer birim kendisini baz alarak bir koordinat sistemi oluşturur ve hesaplamaları ona göre yapar. Bunu yapmasının sebebi, artımsal hatalardan kaçınmaktır.

Her hexadımda, bir sonraki hexadımdan önce kontrol edilmesi gereken üç unsur vardır:

- Herhangi bir gruplanmamış sabit birim saptanmış mı.
- Herhangi bir fliper birim saptanmış mı.
- Seçilmiş hareket algoritmasına göre (“Rasgele Hareket” veya “Puanlı Hareket”), hareket edilecek bir sonraki heksagon seçilir..

Eğer hexer birim herhangi bir sabit birim saptamazsa, seçilen heksagona doğru hareket eder ve önceki adımları tekrar eder.

Eğer hexer birim bir sabit birim saptarsa, olduğu yere derhal bir sanal birim bırakır. Amaç, üç kolineer olmayan sanal birim bırakıp, Cayley-Menger determinant kullanarak sabit birimin

düzlemini saptamaktır. İlk sanal birim bırakıldıktan sonra, hexer birim, hex çemberinde gezmeye başlar ve sabit birimi görebileceği iki nokta daha arar. Eğer hexer birim hex çemberindeki tüm olası heksagonları gezdiyse ve sabit birimi gören bir heksagon bulamadıysa, sabit birim *başka bir düzlemde* olarak işaretlenir. Bu demektir ki, sözkonusu sabit birim hexer birim ile aynı düzlemde değildir. Eğer bir ya da daha fazla heksagona, çeşitli engeller sebebiyle gidemiyorsa ve gidilebilecek diğer heksagonların hepsi ziyaret edilmesine rağmen üç sanal birim bırakılmadıysa, sözkonusu sabit birim *bu düzlem için gruplanamaz* olarak işaretlenir. Bu, hexer birim başka bir düzleme geçtiğinde bu birimlerin *saptanmamış* olarak işaretleneceği anlamına gelir. Eğer hexer birim halihazırda son düzlemdeyse, sözkonusu sabit birim kalıcı olarak *gruplanamaz* olarak işaretlenir. Daha açık olmak gerekirse, eğer hexer birim sabit birimin etrafındaki hex çemberini tamamlayamazsa, sabit birimin gruplanayacağına ya da sadece bu düzlemde olmadığına net olarak karar veremez.

Eğer hexer birim bir fliper birim saptarsa, derhal o birimin etrafında bir hex çember çizer. Daha sonra, bir sanal birim bırakır. Sanal birimin bırakılmasının ardından fliperlik durumu tekrar kontrol edilir. Eğer flip belirsizliği çözülmüşse, birim, fliper olmaktan çıkar. Fliper birimleri onarmanın asıl amacı, iki boyutlu ağ yapısını jenerik global rijid bir hale sokmak olduğu için, fliper birimin önce gruplanmış olması gerekir. Bu durumda fliper birim gruplanamaz olarak işaretlenmişse, aynı zamanda *onarılamaz* olarak da işaretlenir.

Hexer birim, tüm sabit birimler gruplanmış/onarılmış ya da gruplanamaz/onarılamaz olarak işaretlenene kadar seyahatine devam eder.

Algoritma 2 HexMove

```
1: for (KAA'daki her birim için) do
2:   if (eldeki birimde flip belirsizliği varsa) then
3:     eldeki birimi fliper olarak işaretle
4:   end if
5: end for
6: while (seyahat tamamlanmamışsa) do
7:   if (hexer birim, sabit bir birim saptamışsa) then
8:     if (saptanmış birim gruplanmamışsa) then
9:       saptanmış birimi grupta
10:    end if
11:    if (saptanmış birim gruplanmışsa ve flip belirsizliğine sahipse) then
12:      flip belirsizliğini onar
13:    end if
14:  end if
15:  if (kat tamamlanmışsa) then
16:    üst kata çık
17:  else
18:    ziyaret edecek sonraki heksagona karar ver
19:  end if
20: end while
```

4.3.4 Deney Sonuçları

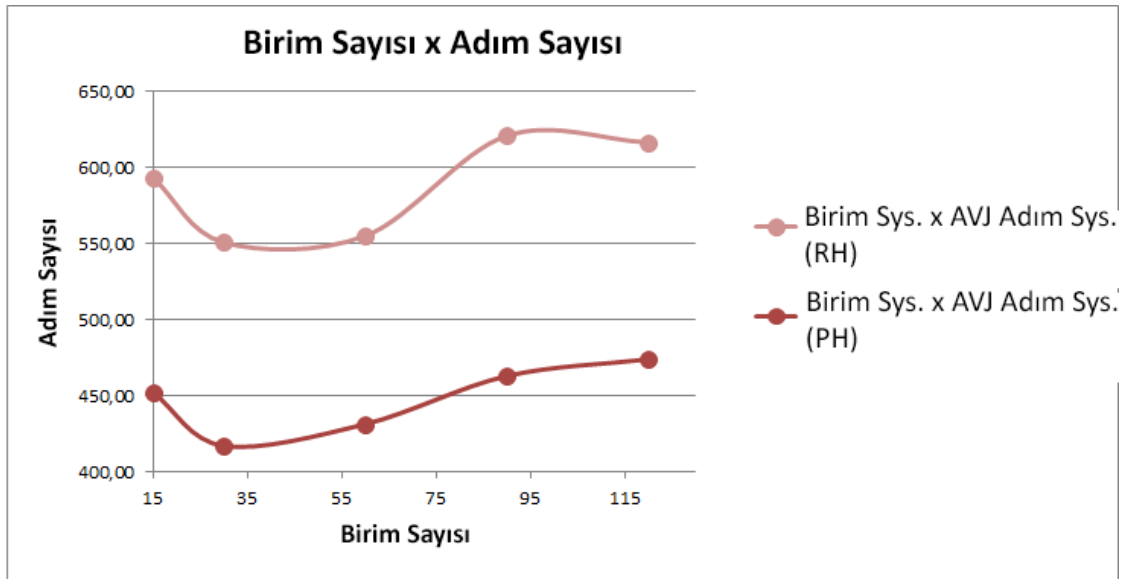
Bu çalışmada, aşağıdaki gibi bir deney düzeni kullandık:

- n static birim rasgele $1000 \times h \times 1000$ lik çok katlı bir binaya dağıtılıyor. Burada n birim sayısını ve h kat yüksekliğini göstermektedir. Bütün katlar birbirine paraleldir.
- Her katta eşit miktarda sabit birim mevcuttur.
- Sabit birimler her zaman bağlı bir çizge yapısı oluşturur.
- Hexer birim ve sabit birimler 10%'a kadar ölçüm hatası yapabilirler.
- Harekete başlanmadan önce, fliper birimler tespit edilir.
- Hexer birim, tüm sabit birimler gruplanmış/onarılmış ya da gruplanamaz/onarılamaz olarak işaretlenene kadar seyahatine devam eder.
- Her deney 100 defa çalıştırılıp ortalama değerleri raporlanmıştır.

Puanlı Hareket - Rasgele Hareket karşılaştırması

PH algoritması ile RH algoritmasını kıyaslamak önemlidir çünkü RH algoritması, adım sayısı açısından karşımıza çıkabilecek en kötü senaryodur. Bu deney yalnızca adım sayısı açısından

önemlidir çünkü HexMove'daki hareket algoritmaları, ağ yapısının bulunması üzerinedir. Diğer işlemler her ikih hareket algoritması için de sabittir.

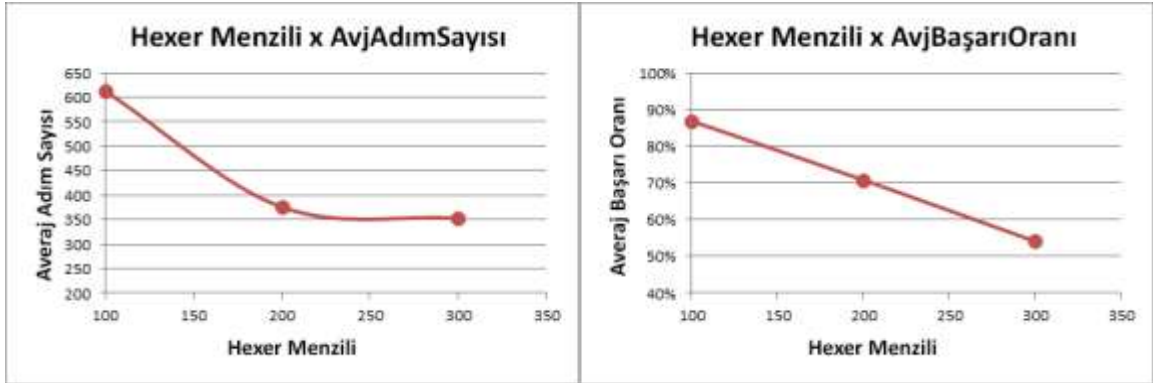


Şekil 4.3.6 “Puanlı Hareket” “Rasgele Hareket”e göre daha iyi sonuç veriyor

Şekil 4.3.6 deney sonuçlarına göre, PH, RH'e kıyasla çok daha kısa sürede işlemini tamamlamaktadır. Ayrıca, açıkça belirtmek adına, *BirimSayısı* = 15–35 aralığında, Cayley-Menger determinantının doğru kullanımı için yeterli sayıda birim mevcut olmadığından, adım sayısında bir azalma gözlemlenmektedir.

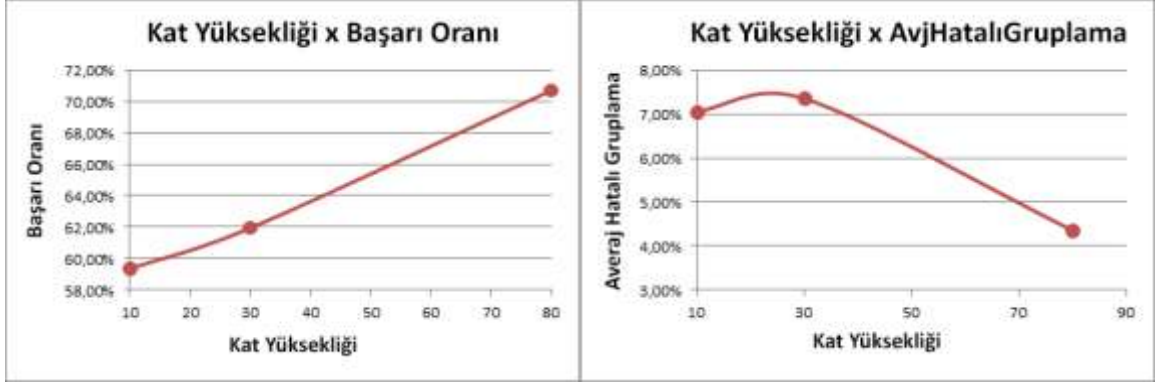
Hexer birimin radyo menzilinın Adım Sayısı ve Gruplama Başarı Oranı'na etkisi

Hexer birimin radyo menzilinı efektif bir şekilde ayarlamak, HexMove yaklaşımının önemli bir kısmını oluşturur. Aşağıdaki sonuçlar hexer birimin radyo menzilinın adım sayısı ve gruplama başarı oranına etkisini göstermektedir.



Şekil 4.3.7: Hexer menzilinı arttırmak, adım sayısını azaltmak için iyi bir yol fakat başarı oranında düşüşe sebebiyet veriyor

Bulgularımıza göre, hexer birimin geniş çaplı algılama menziline sahip olması, adım sayısını azaltıyor (Şekil 4.3.7 sol) lakin, bu durum hexer birimin daha büyük adımlar atmasına sebep oluyor ve çevresel sınırlamalar yüzünden sanal birim koyacak daha az nokta bulabiliyor. Bu yüzden, gruplama başarısında azalma görünüyor (Şekil 4.3.7 sağ). Hexer birimin menzilinı arttırmanın bir diğer etkisi, diğer düzlemlerdeki birimlerin görölme oranının artması oluyor. Bu durum, hatalı gruplama oranında artışa sebebiyet verebiliyor.



Şekil 4.3.8: Kat yüksekliği azaldıkça, hatalı gruplama oranı artıyor ve gruplama başarı oranı düşüyor

Kat yüksekliğinin etkisinin hatalı gruplamaya (Figür 8 sağ) ve gruplama başarı oranına (Şekil 4.3.8 sol) olan etkisini görüyoruz.

Bu sonuçlar gösteriyor ki, diğer katlardaki birimlerin saptanma oranı direkt olarak başarı oranına etki ediyor. Sıradaki deneyde, hexer birimin menzilini 70'e indiriyoruz ve 80 kat yüksekliğini sabit bırakıyoruz. Bu şekilde, hexer birim asla başka bir düzlemdeki sabit birimi saptayamıyor.



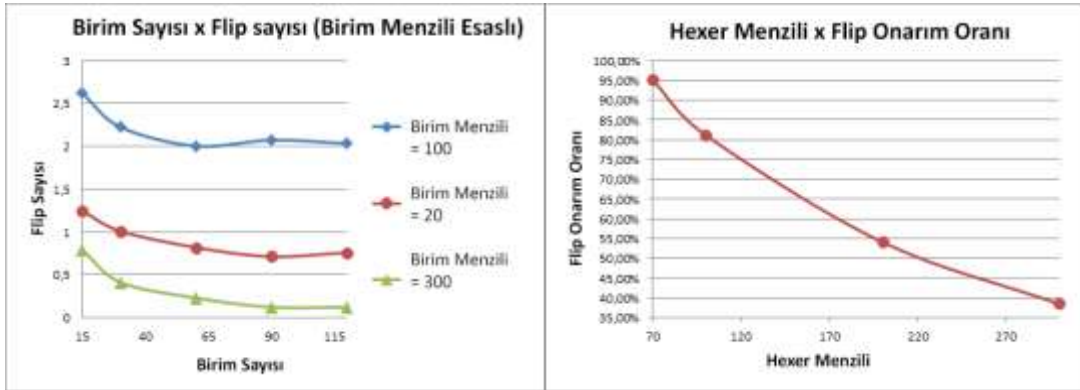
Şekil 4.3.9: Hexer birimin menzilini, kat yüksekliğinden daha kısa yapmak, hatalı saptama (gruplama) oranını neredeyse 0'a indiriyor

Şekil 4.3.9 gösteriyor ki, hatalı saptama gözle görülür bir biçimde azalıyor. Aynı zamanda görüyoruz ki, sabit birim menzili de hatalı saptama oranı üzerine etkili.

Son tahlilde, hexer birimin menzilini azaltmak, başarı oranını arttırmak için iyi bir yol olmasıyla beraber aynı zamanda daha fazla adım sayısı anlamına geliyor.

Flip Onarım Oranı

Bir önceki deneyde, menzilin, başarı oranına olan etkisini gözlemledik. Bir fliper birim gruplanamamışsa, aynı zamanda onarılamaz. Bu durumda gruplama başarı oranı, flip onarım oranını da doğrudan etkileyecektir. Sıradaki deneyler birim sayısının flip sayısına olan etkisini ve hexer menzilin flip onarım oranına olan etkisini göstermektedir.



Şekil 4.3.10: Hexer birimin menzilini arttırmak, gruplama başarı oranını kötü etkiler dolayısıyla flip onarım oranını da kötü etkiliyor

Şekil 4.3.10 sonuçlarına göre, hexer birimin menzilin artışı, flip onarım oranında azalmaya neden oluyor.

5. SONUÇ

Bu projede üç boyutta konumlama probleminin çözümü üzerine arařtırmalar yapılmıřtır. Yapılan arařtırmalar üç boyutta problemin ölçüm hataları olduđu durumlarda karmařık olmasından yola çıkmıř ve bu karmařıklığı çeřitli kısıtlar ekleyerek çözmek üzerine arařtırmalar yoğunlařmıřtır. Proje süresince üç temel konu üzerine arařtırma yapılmıřtır. Bu konulardan birincisi iki boyutta üçlü pozisyonlama probleminde karmařıklığa sebep olan düzlemsel evrimlerin giderilmesidir. Bu kapsamda konumlama sırasında ölçüm hatalarından kaynaklanan düzlemsel evrimlerin sebepleri arařtırılmıř, düzlemsel evrime sebep olan durumlar belirlenmiř ve bu durumların oluřmasını engelleyecek yöntemler geliřtirilmiřtir.

Proje süresince çalıřılan ikinci konuda ise üç boyuttaki problem iki boyutlu düzlemler kullanarak modellenmiř ve bu düzlemler üzerindeki noktaların konumlanması ile ve sonrasında düzlemlerin birbirlerine göre konumlanmasıyla konumlama problemini çözecek algoritmalar geliřtirilmiřtir. Buradaki temel gözlem, üç boyutta gerçek hayatta cisimlerin iki boyutlu düzlemler üzerinde durduđudur (ör. bina katları). Geliřtirilen konumlama algoritmamız bilinen algoritmalarla karşılařtırılmıř ve özellikle noktaların rasgele dađılmadıđı ve belirli düzlemler üzerine dađıldıđı durumlarda başarılı sonuçlar verdiđi gözlenmiřtir.

Proje süresince çalıřılan üçüncü ve son konuda ise konumlanmak istenen noktalardan en az birinin hareketli olduđu, özellikle bina içi konumlama uygulamalarında, bu hareketli nokta vasıtasıyla yine noktaları düzlemlere (katlara) ayırma ve katlar arası konumlamanın en iyi ne řekilde yapılacađı üzerine arařtırmalar yapılmıř, ve buna uygun hareket algoritmaları geliřtirilmiřtir.

Yapılan bu çalıřmalar sonucunda bir adet uluslararası bildiri hazırlanmıř ve sunulmuřtur. İki adet de yüksek lisans tezi tamamlanmak üzeredir. Bu iki teze konu olan kısımlardan da yakın zamanda yayın yapma beklentisi vardır. Proje süresince geliřtirilen üç boyutlu konumlama probleminin iki boyutta düzlemlere ayrılarak alt uzayda çözümlenmesi kendi içinde arařtırma olarak yenilikler barındıran bir fikirdir ve bu konu üzerine daha fazla arařtırma yapılması planlanmaktadır. Yine özellikle bina içi gibi katların birbirine paralel durduđu durumlarda iki boyutlu düzlemler kullanarak üç boyutlu konumlama problemini çözmek karmařıklığı azaltıcı faydalar sağlamaktadır, ve özellikle bina içi konumlama problemlerinin çözümünde bu konu üzerinde çalıřmalar yapılmasının faydalı olduđu düşünölmektedir.

6. KAYNAKLAR

Akcan, H., ve Evrendilek, C. GPS-Free Directional Localization via Dual Wireless Radios, *Computer Communications*, Volume 35, Issue 9, pages 1151-1163, (2011).

Akcan, H., ve Evrendilek, C. Reducing The Number Of Flips In Trilateration With Noisy Range Measurements. In *Proceedings of the 12th ACM International Workshop on Data Engineering for Wireless and Mobile Access (MobiDE'13)*, in conjunction with SIGMOD, pages 20-27, New York, NY, USA, June 2013.

Akcan, H., Kriakov, V., Bronnimann, H., Delis, A. GPS-Free node localization in mobile wireless sensor networks. In *Proceedings of the 5th ACM International Workshop on Data Engineering for Wireless and Mobile Access (MobiDE'06)*, pages 35–42, Chicago, Illinois, USA, June 2006.

Akcan, H., Kriakov, V., Bronnimann, H., Delis, A. Managing cohort movement of mobile sensors via GPS-free and compass-free node localization. *Journal of Parallel and Distributed Computing*, 70(7):743–757, 2010.

Anderson, B. D. O., Shames I., Mao G., Fidan B. Formal theory of noisy sensor network localization. *SIAM J. Discrete Math.*, 24(2):684–698, 2010.

Aspnes, J., Eren, T., Goldenberg, D. K., Morse, A. S., Whiteley, W., Yang, Y. R., Anderson, B. D. O., Belhumeur, P. N. A theory of network localization. *IEEE Transactions on Mobile Computing*, 5(12):1663–1678, 2006.

Aspnes, J., Goldenberg D. K., Yang Y. R. On the computational complexity of sensor network localization. In *First International Workshop on Algorithmic Aspects of Wireless Sensor Networks (ALGOSENSORS'04)*, pages 32–44, Turku, Finland, July 2004.

Benslimane A. Localization in vehicular ad hoc networks. In *Systems Communications*, pages 19–25, Montreal, Quebec, Canada, August 2005.

Blumenthal, L.M. In *Theory and applications of distance geometry* (2nd ed.), page 347. Chelsea Publishing Company, Bronx, New York, 2004.

Boukerche, A., Oliveira, H. A. B. F., Nakamura, E. F., Loureiro, A. A. F. Vehicular ad hoc networks: A new challenge for localization-based systems. *Computer Communications*, 31(12):2838–2849, 2008.

Bulusu, N., Heidemann, J., Estrin, D. Gps-less low cost outdoor localization for very small devices. *IEEE Personal Communications Magazine*, 7(5):28–34, October 2000.

Caffery, J., Stober, G. Overview of radiolocation in cdma cellular systems. *IEEE Communications Mag.*, 36(4):38–45, 1998.

Capkun, S., Hamdi, M., Hubaux. J.-P. GPS-free Positioning in Mobile Ad Hoc Networks. *Cluster Computing*, 5(2):157–167, 2002.

Choset, H., Konukseven, I. and Rizzi, A. (1997). ‘Sensor based planing: A control law for generating the generalized voronoi graph’. *IEEE International Conference in Advanced Robotics*, pp. 333–338.

Clark, B. N., Colbourn, C. J., Johnson, D. S. Unit disk graphs. *Discrete Mathematics*, 86(1-3):165–177, 1990.

Connelly, R. On generic global rigidity. *DIMACS Ser. Discrete Math. Theoret. Comput. Sci.*, 4:147 – 155, 1991.

Dantu, K., Rahimi, M. Shah, H., Babel, S., Dhariwal, A. and Sukhatme, G.S. (2005). ‘Robomote: enabling mobility in sensor networks’. *Proc. Inter. Symp. Information Processing in Sensor Networks (IPSN '05)*, pp. 404–409.

Djugash, J., Singh, S, Kantor, G. and Zhang, W. (2006). ‘Range-Only SLAM for robots operating cooperatively with sensor networks’. *IEEE International Conference on Robotics and Automation*, pp. 2078–2084.

Doherty, L., Pister, K., Ghaoui, L. El. Convex position estimation in wireless sensor networks. In *INFOCOM*, pages 1655–1663, Anchorage, AK, USA, 2001.

Eren, T., Goldenberg, O., Whiteley, W., Yang, Y., Morse, A., Anderson, B., Belhumeur, P. Rigidity, computation, and randomization in network localization. In *INFOCOM*, volume 4, pages 2673–2684, Hong Kong, China, March 2004.

Evrendilek, C., Akcan, H. "On the complexity of trilateration with noisy range measurements" IEEE Communications Letters, Volume 15, Issue 10, pages 1097-1099, October 2011.

Gezici, S., Tian, Z., Giannakis, G., Kobayashi, H., Molisch, A., Poor, H., Sahinoglu, Z. Localization via ultra-wideband radios: a look at positioning aspects for future sensor networks. Signal Processing Magazine, IEEE, 22(4):70–84, July 2005.

Giorgetti, G., Gupta, S. K. S., Manes, G. Wireless localization using self-organizing maps. In Proceedings of the 6th International Conference on Information Processing In Sensor Networks (IPSN'07), pages 293–302, Cambridge, Massachusetts, USA, April 2007.

Goldenburg, D.K., Krishnamurthy, W., Maness, W.C., Yang, Y., Young, A., Morse A.S., Savvides, A. and Anderson, B.D.O. (2005). 'Network localization in partially localizable networks'. IEEE INFOCOM, 1, pp. 313–326.

Gustafsson, F., Gunnarsson, F. Mobile positioning using wireless networks. IEEE Signal Processing Magazine, pages 41–53, 2005.

Han, G., Xu, H., Duong, T.Q., Jiang, J., Hara, T. Localization algorithms of wireless sensor networks: a survey. Telecommunication Systems, 31(4):1 – 18, 2011.

He, T., Huang, C., Blum, B. M., Stankovic, J. A., Abdelzaher, T. F. Range-free localization and its impact on large scale sensor networks. ACM Trans. Embedded Comput. Syst., 4(4):877–906, 2005.

Heinzelman, W.R., Chandrakasan, A., Balakrishnan, H. Energy-efficient communication protocol for wireless microsensor networks. In System Sciences, 2000. Proceedings of the 33rd Annual Hawaii International Conference on, pages 10 pp. vol.2–, 2000.

Hightower, J., Borriello, G. Location systems for ubiquitous computing. IEEE Computer, 34(8):57–66, 2001.

Hu, L. Evans, D. Localization for mobile sensor networks. In Proceedings of the International Conference on Mobile Computing and Networking (MOBICOM'04), pages 45–57, Philadelphia, PA, USA, 2004.

Iyengar, R., Sikdar, B. Scalable and distributed GPS free positioning for sensor networks. In Proceedings of the IEEE International Conference on Communications (ICC'03), pages 338–342, Anchorage, AK, USA, May 2003.

Ke, W.C., Liu, B.H. and Tsai, M.J. (2007). 'Constructing a wireless sensor network to fully cover critical grids by deploying minimum sensors on grid points is NP-complete'. *IEEE Trans. Comput.*, 56(5), pp.710–715.

Khoury, H.M., Kamat, V.R. Evaluation of position tracking technologies for user localization in indoor construction environments. *Automation in Construction*, 18(4):444 – 457, 2009.

Kukshya, V., Krishnan, H., Kellum, C. Design of a system solution for relative positioning of vehicles using vehicle-to-vehicle radio communications during gps outages. In IEEE Vehicular Technology Conference (VTC'05 Fall), pages 1313–1317, Dallas, TX, USA, September 2005.

Laman, G. On graphs and rigidity of plane skeletal structures. *Journal of Engineering Mathematics*, 4(10):331 – 340, 2002.

Li, X., Mitton, N., Simplot-Ryl, I., Simplot-Ryl, D. Mobile beacon assisted sensor localization with dynamic beacon mobility scheduling. In Proceedings of the 8th IEEE International Conference on Mobile Ad-hoc and Sensor Systems (MASS'11), Valencia, Spain, October 2011.

Liu, J., Zhang, Y., Zhao, F. Robust distributed node localization with error management. In Proceedings of the 7th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc'06), pages 250–261, Florence, Italy, May 2006.

Lo, A., Xia, L., Niemegeers, I. G., Baugé, T., Russell, M. , Harmer, D. EUROPCOM - An Ultra-WideBand (UWB)-Based Ad Hoc Network for Emergency Applications. *VTC Spring 2008*: 6-10

Luo, R.C. and Chen, O. (2012). 'Mobile sensor node deployment and asynchronous power management for wireless sensor networks'. *Industrial Electronics, IEEE Transactions*, 59(5), pp. 2377–2385

Lymberopoulos, D., Priyantha, N. B., Goraczko, M., Zhao, F. Towards energy efficient design of multi-radio platforms for wireless sensor networks. In Proceedings of the 7th International Conference on Information Processing In Sensor Networks (IPSN'08), pages 15–26, St. Louis, Missouri, USA, April 2008.

Mao, G., Fidan, B., Anderson, B.D.O. Wireless sensor network localization techniques. *Computer Networks*, 51(10):2529 – 2553, 2007.

Moore, D. Leonard, J., Rus, D., Teller, S. Robust distributed network localization with noisy range measurements. In Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems (SenSys'04), pages 50–61, Baltimore, MD, USA, 2004.

Nagpal, R., Shrobe, H. E., Bachrach, J. Organizing a global coordinate system from local information on an ad hoc sensor network. In Proceedings of the 2nd International Conference on Information Processing In Sensor Networks (IPSN'03), pages 333–348, Palo Alto, California, USA, 2003.

Ni, K., Ramanathan, N., Chehade, M. N. H., Balzano, L., Nair, S., Zahedi, S., Kohler, E., Pottie, G., Hansen, M., Srivastava, M. Sensor network data fault types. *ACM Trans. Sen. Netw.*,5:25:1–25:29, June 2009.

Niculescu, D. Nath, B. Dv based positioning in ad hoc networks. *Telecommunication Systems*, 22(1-4):267–280, 2003.

Olson,E., Leonard,J.J. and Teller,S. (2006). 'Robust range-only beacon localization'. *IEEE J. Oceanic Eng.*, 31(4), pp. 949–958

Park, J., Demaine, E. D., Teller, S. J. Moving-baseline localization. In Proceedings of the 7th International Conference on Information Processing In Sensor Networks (IPSN'08), pages 15–26, St. Louis, Missouri, USA, April 2008.

Parker, R., Valaee, S. Vehical localization in vehicular networks. In IEEE Vehicular Technology Conference (VTC'06 Fall), pages 1–5, Montreal, Quebec, Canada, September 2006.

Patwari, N., Ash, J. N., Kyperountas, S., Hero III, A. O., Moses, R. L., Correal, N. S. Locating the nodes: cooperative localization in wireless sensor networks. *Signal Processing Magazine, IEEE*, 22(4):54–69, July 2005.

Pering, T., Agarwal, Y., Gupta, R. K., Want, R. CoolSpots: reducing the power consumption of wireless mobile devices with multiple radio interfaces. In *Proceedings of the 4th International Conference on Mobile Systems, Applications, and Services (MobiSys'06)*, pages 220–232, Uppsala, Sweden, June 2006.

Porretta, M., Nepa, P., Manara, G., Giannetti, F. Location, location, location. *IEEE Vehicular Technology Magazine*, pages 20–29, 2008.

Priyantha, N. B., Balakrishnan, H., Demaine, E., Teller, S. Anchor-free distributed localization in sensor networks. In *Proceedings of the First International Conference on Embedded Networked Sensor Systems (SenSys'03)*, pages 340–341, Los Angeles, CA, USA, 2003.

Priyantha, N. B., Balakrishnan, H., Demaine, E. D., Teller, S. J. Mobile-assisted localization in wireless sensor networks. In *INFOCOM*, pages 172–183, Miami, FL, USA, March 2005.

Rudafshani, M. Datta, S. Localization in wireless sensor networks. In *Proceedings of the 6th International Conference on Information Processing In Sensor Networks (IPSN'07)*, pages 51–60, Cambridge, Massachusetts, USA, April 2007.

Sahinoglu, Z., Gezici, S., Guven, I. *Ultra-wideband Positioning Systems: Theoretical Limits, Ranging Algorithms, and Protocols*. Cambridge University Press, 2008.

Savarese, C., Rabaey, J. M., Langendoen, K. Robust positioning algorithms for distributed ad-hoc wireless sensor networks. In *USENIX*, pages 317–327, Monterey, CA, USA, 2002.

Savvides, A., Han, C.-C., Strivastava, M. B. Dynamic fine-grained localization in ad-hoc networks of sensors. In *Proceedings of the 7th annual International Conference on Mobile Computing and Networking (MobiCom'01)*, pages 166–179, New York, NY, USA, 2001. ACM Press.

Saxe, J. B. Embeddability of weighted graphs in k-space is strongly np-hard. In *17th Allerton Conf. Commun. Control Comput.*, pages 480–489, 1979.

Shang, Y., Ruml, W., Zhang, Y., Fromherz, M. P. J. Localization from mere connectivity. In Proceedings of the 4th ACM International Symposium on Mobile Ad Hoc Networking & Computing (MobiHoc'03), pages 201–212, Annapolis, Maryland, USA, 2003.

Stathopoulos, T., Lukac, M., McIntire, D., Heidemann, J. S., Estrin, D., Kaiser, W. J. End-to-end routing for dual-radio sensor networks. In INFOCOM, pages 2252–2260, Anchorage, Alaska, USA, May 2007.

Xiao, J., Han, S., Zhang, Y. and Xu, G. (2010). 'Hexagonal grid-based sensor deployment algorithm'. *Proc. Control Decision Conf.*, pp. 4342–4346

Yemini, Y. Some theoretical aspects of position-location problems. In Foundations of Computer Science, 1979., 20th Annual Symposium on, pages 1–8, 1979.

Zhang, P., Martonosi, M. Locale: Collaborative localization estimation for sparse mobile sensor networks. In IPSN '08: Proceedings of the 7th International Conference on Information Processing In Sensor Networks, pages 195–206, St. Louis, Missouri, USA, April 2008.

TÜBİTAK
PROJE ÖZET BİLGİ FORMU

Proje Yürütücüsü:	Doç. Dr. HÜSEYİN AKCAN
Proje No:	112E099
Proje Başlığı:	Mobil Kablosuz Ağlarda Üç Boyutta Konum Tespiti Algoritmalarının Geliştirilmesi Ve Analizi
Proje Türü:	Uluslararası
Proje Süresi:	24
Araştırmacılar:	CEM EVRENDİLEK
Danışmanlar:	
Projenin Yürütüldüğü Kuruluş ve Adresi:	İZMİR EKONOMİ Ü.
Projenin Başlangıç ve Bitiş Tarihleri:	01/10/2012 - 01/10/2014
Onaylanan Bütçe:	143560.0
Harcanan Bütçe:	78357.47
Öz:	<p>Bu proje, İzmir Ekonomi Üniversitesi bünyesinde, Doç. Dr. Hüseyin AKCAN yürütücülüğünde, Doç. Dr. Cem EVRENDİLEK'in araştırmacı olarak, Onur ÇAĞIRICI ve Ali Teoman UNAY'ın bursiyer olarak katkılarıyla gerçekleştirilmiştir. Proje 3501 TÜBİTAK Kariyer projesi kapsamında TÜBİTAK tarafından desteklenmiştir. Her iki bursiyerimiz de projenin başından itibaren 24 ay süreyle projeye dahil olmuşlar, yüksek lisans tezlerini bu projede üzerlerine düşen konularda yapmışlardır. Tezlerini oluşturacak kısımlar bu rapor ile birlikte verilmiştir, tezlerinin yazımı ise son aşamasındadır ve bulunduğumuz dönem sonunda bitirmeleri beklenmektedir.</p> <p>Proje süresince üç boyutta konum tespiti üzerine çalışmalar yapılmıştır. İki boyutta ölçüm hataları olduğu durumlarda karmaşık olan bu problemin üç boyuttaki karakteristiği incelenmiş ve problemi karmaşıklığından kurtaracak şekilde ek kısıtlar eklenerek çözümü üzerine yoğunlaşmıştır. Bu ek kısıtlar üç boyutlu ortamların iki boyutlu düzlemlerden oluştuğu ortamlar (ör. bina katları) ve pozisyonu belirlenmek istenen elemanların birinin veya birden fazlasının hareket edebilir olduğu ortamlar olarak belirlenmiştir. Yapılan çalışmalar sonucunda her iki kısıtın da problemin çözümüne önemli katkıları olduğu tespit edilmiş ve raporlanmıştır. Proje sonucunda hazırlanan bir bildiri uluslararası SIGMOD konferansı bünyesinde onikincisi düzenlenen ACM International Workshop on Data Engineering for Wireless and Mobile Access (MobiDE) çalıştayında Haziran 2013'de yürütücü tarafından sunulmuştur. Proje ek olarak iki yüksek lisans öğrencisi için tez çalışması teşkil etmiştir. Her iki tezden de önümüzdeki zamanda yayın beklentisi vardır.</p> <p>Projede elde edilen sonuçlar ışığında, üç boyutta konumlama için iki boyutlu alt uzayların kullanımının iyi sonuçlar verdiği gözlemlenmiş, bu ve benzeri kısıtlar eklenerek üç boyutta konumlama problemi üzerine daha çok çalışma yapılması kararlaştırılmıştır.</p>
Anahtar Kelimeler:	yönsel konum tespiti, mobil kablosuz ağlar, kablosuz algılayıcı ağlar, algoritmalar
Fikri Ürün Bildirim Formu Sunuldu Mu?:	Hayır
Projeden Yapılan Yayınlar:	1- Reducing The Number Of Flips In Trilateration With Noisy Range Measurements (Bildiri)1- Reducing The Number Of Flips In Trilateration With Noisy Range Measurements (Bildiri - Uluslararası Bildiri - Sözlü Sunum),