

## Bayes Ağı ile Küçük Uyduların Yönelim Kestirim ve Kontrol Algoritmalarının Hata Tespitinin Yapılması, Ayıklanması ve Düzeltilmesi

Cansu Yıldırım\*  
Orta Doğu Teknik  
Üniversitesi, Ankara

Halil Ersin Söken†  
Orta Doğu Teknik  
Üniversitesi, Ankara

### ÖZET

*Bu çalışmanın amacı küçük uyduların Yönelim Belirleme ve Kontrol Sisteminde (YBKS) kullanılan Çarpımsal Genişletilmiş Kalman Filtresi'ndeki (ÇGKF) arıza/hata tespitinin yapılması, ayıklanması ve düzeltilmesidir (HTAD). Arıza/Hata tespitinin yapılması ve ayıklanması için gözlemlenebilirlik algoritmalarından ve Bayes Ağından yararlanılmıştır. Daha sonrasında, arıza/hata düzeltmeleri için ise birbirlerinin alternatifleri olan dört farklı filtre tasarımına gidilmiş ve arıza/hata nedenine göre filtreler arasında geçiş sağlanmıştır.*

### GİRİŞ

Küçük uyduların birçok avantajı vardır. Ancak bu avantajlar bazı dezavantajları da beraberinde getirmektedir. Uydunun küçük boyutu nedeniyle yedek sensör kullanımı sınırlıdır. Ayrıca daha önce uzay görevlerinde kullanılmayan düşük kaliteli sensörler de maliyet düşürmek amacıyla kullanılmaktadır. Bunun nedeni daha önce uzay görevinde başarılı olmuş bir sensörün fiyatının artmasıdır. Bu nedenle birçok hata ve başarısız görevle karşılaşmaktadır. Birçok alt sistemde oluşabilecek hataya rağmen belki de en kritik hata Yönelim Belirleme ve Kontrol Sistemi'ndeki (YBKS) hatalardır. Bu hatalar kolaylıkla uydunun arızalanmasına neden olabilir. Bu nedenle, bu çalışmada hibrit YBKS kullanılmış ve farklı algoritmalar ve sensörler ile yönelim belirleme hatası veya arızası en aza indirilmeye çalışılmıştır.

Literatürdeki YBKS Hata Tespit Ayıklama ve Düzeltme (HTAD) algoritmalarına baktığımızda, eşik değerlerinin kullanıldığı [Lee et al., 2018], durum bazlı önerilerde bulunduğu [Nakasuka et al., 2018] ve katmanlı YBKS algoritmalarının kullanıldığı görülmektedir [Shunichiro Nomura, 2017]. Ancak, bu algoritmalar hatanın nedenini anlamada yetersizdir. Ayrıca, birçok ön bilgi gereksinimi vardır [Nakasuka et al., 2018]. Bu durum algoritmaların yeniden kullanımlarının yani; yeni görevler ve

\*Öğrenci, Havacılık ve Uzay Müh. Böl., E-posta: yildirim.cansu@metu.edu.tr

†Doç. Dr., Havacılık ve Uzay Müh. Böl., E-posta: esoken@metu.edu.tr

farklı uydu tasarımları için, uygun olmadığı anlamına gelmektedir. Ayrıca Bayes Ağı literatürde uydu YBKS'si için bir çalışmada kullanılmıştır; ancak, bu ağ Naive Bayes modeli ile oluşturulmuş ve rastgele değişkenler arasındaki birçok ilişki göz ardı edilmiştir [Yanning et al., 2016].

Bu çalışmanın temel amacı, Bayes Ağı vasıtasıyla, uydunun yönelimini belirlemek amacıyla kullanılan Çarpımsal Genişletilmiş Kalman Filtresi'ndeki (ÇGKF) arıza/hatanın kaynağını anlamaktır. Arıza/hata kaynağı belirlendikten sonra uydu yönelim kontrolü, arıza/hata tipine bağlı olarak sağlanacaktır. Ek olarak, önerilen Bayes Ağındaki birçok düğüm gözlemlenebilir niteliktedir. Düğüm gözlemlenebilirliğini sağlamak amacıyla karşılaştırma ve gözlem algoritmaları kullanılmıştır. Uydu üzerinde varolan yönelim sensörlerinin farklı birleşimleri neticesinde dört farklı ÇGKF algoritması elde edilmiştir. Çalışan ÇGKF'deki hatanın kaynağına göre filtre değiştirilerek hatanın giderilmesi hedeflenmiştir.

## YÖNTEM

Kullanılan yöntemler dört ana başlık altında incelenebilir: Bayes Ağı, Kestirim Algoritmaları, Gözlemlenebilirlik Algoritmaları ve Ana Method. Ancak, gözlemlenebilirlik algoritmaları içerik bakımından uzun olduğu ve çalışmada anlatılmaya çalışılan sonuçtan bağımsız olarak incelenebileceği için bu çalışmada method kısmında yer almayacaktır.

### Bayes Ağı

Bayes Ağı bir çeşit olasılıksal grafiksel modeldir. Birbirlerine tek yön ile bağlı rastgele değişkenlerin ilişkilerini göstermektedir. HTAD tasarlanılırken Bayes Ağı kullanılmasının temel nedeni arıza/hatanın gerçekleşme olasılığını anlamak için Bayes Ağının gelişmiş bir araç olmasıdır. Burada gelişmişten kasıt, rastgele değişkenlerin ortak olasılık dağılımlarının bulunmasının ve depolanmasının zor olmasıdır. Ancak, Bayes Ağı bu sorunu çözmek için iyi bir araçtır. Değişkenlerin ilişkileri gösterilirken, birbirlerini görece olarak az etkileyen rastgele değişkenlerin arasında bağımsızlık varsayımı kolayca yapılabilmektedir. Bu özellik direkt olarak, ortak olasılık dağılımının boyutunu düşürmektedir. Buna ek olarak; arıza/hata nedenleri arasındaki neden/sonuç ilişkisinin anlaşılmasını kolaylaştırmaktadır. Son olarak, algoritma ve model bilgisi birbirinden ayrıldığı için farklı uzay görevleri ve uzay araçları için yeniden kullanılabilir. Daha detaylı açıklamak gerekirse, yeni bir uzay aracı tasarımı için sadece olasılık dağılımları değiştirilebilir. Rastgele değişkenlerin birbirleri arasındaki ilişki aynı kalacaktır. Bu konuda detaylı bilgi verilen referanstan elde edilebilir [Daphne Koller, 2009]. Bütün bu nedenlerle, ÇGKF'deki görülebilecek hatalar rastgele değişkenler olarak gösterilmiş olup, dört farklı ÇGKF tasarımına gidilmiştir. Bu dört farklı ÇGKF tasarımının sonucunda dört farklı Bayes Ağı modeli elde edilmiştir (Şekil 1). Ana method kısmında, farklı filtrelerin elde edilmesi ile ilgili açıklama verilmiştir.

### Kestirim Algoritmaları

Bu başlıkta Durağan/Statik Yönelim Belirleme Algoritmalarına ve ÇKFG'ye yer verilecektir. Durağan/Statik Yönelim Belirleme Algoritması ve Yıldız izler kullanılarak kuaternion ölçümleri elde edilmiş olup daha sonrasında Kalman filtresi yardımı ile yönelim kestirimleri elde edilmiştir.

### Durağan/Statik Yönelim Belirleme Algoritması

Durağan/Statik yönelim belirleme algoritması olarak QUEST (Quaternion Estimator Algorithm) algoritması kullanılmıştır. QUEST güneş algılayıcı ve manyetometre bilgisi kullanılarak çalıştırılmış olup kestirilen kuaternion değerleri buradan alınmıştır. Daha sonrasında bu değerler ölçüm değeri olarak kullanılmıştır. Algoritma hakkında detaylı bilgi için belirtilen kaynaktan yararlanılabilir [Markley and Crassidis, 2014].

## ÇGKF

Bu başlıkta iki tür kestirim algoritmasından bahsedilecektir. Esas olarak iki algoritma da ÇGKF'dir. Ancak; birisinde açısal hız ölçümü direkt olarak dönüölçerden alınmaktadır ve durum vektöründe dönüölçer kayması kullanılmaktadır. Bu algoritmaya dönüölçere dayalı ÇGKF denilmektedir. İkinci ÇGKF modelinde ise, açısal hız matematiksel model yardımı ile bulunmaktadır. Burada matematiksel modelden kasıt Newton-Euler denklemleridir. Bu nedenle durum vektöründe direkt olarak açısal hız bulunmaktadır. Bu algoritma matematiksel modele dayalı ÇGKF olarak adlandırılabilir.

### Dönüölçere Dayalı ÇGKF

ÇKFG bir çeşit genişletilmiş kalman filtresi olup özel olarak yönelim durumu kuaternion olarak verildiğinde kullanılmaktadır. Bunun ana nedeni, filtrede kuaternion uzunluğu sabit kalmalıdır; ancak, bu şart ölçüm güncellemesi yapılırken sağlanamamaktadır. ÇGKF bu sorunu çözmek için özel tasarlanmış bir filtre olup kuaternionu direkt durum olarak kullanmayıp Euler açılarındaki hatayı kullanmaktadır. Bu durumlar sınıflandırıldığında “yerel” ve “küresel” olmak üzere iki durum tipi ortaya çıkmaktadır [Markley and Crassidis, 2014]. Yerel durum vektörü formülde (1) gösterilmiştir.  $\delta\theta$  yönelim hatasının yerel gösterimidir.  $\Delta\xi$  ise dönüölçer kaymasının yerel gösterimidir. Global durum vektörü ise kuaternion ( $q$ ) ve dönüölçer kayması ( $\xi$ ) olarak alınmıştır. Ekler kısmında verilen algoritmada (Algoritma 1) ölçüm güncellenmesi, sıfırlama ve öteleme aşamaları sözde kod olarak gösterilmiştir.

$$\Delta x = \begin{bmatrix} \delta\theta \\ \Delta\xi \end{bmatrix} \quad (1)$$

$\delta\theta$  = yönelim hatasının yerel gösterimi

$\Delta\xi$  = dönüölçer kaymasının yerel gösterimi

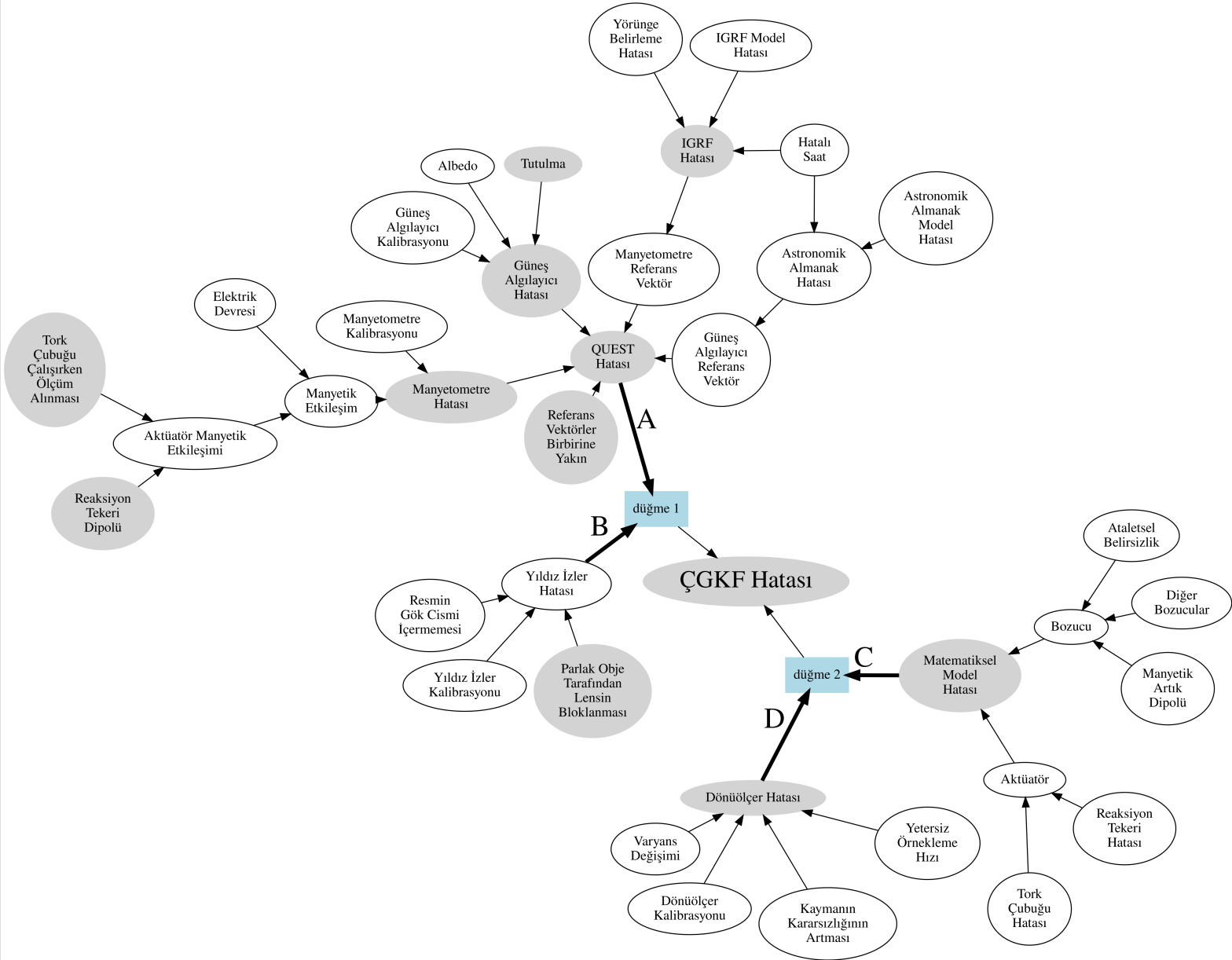
### Matematiksel Modele Dayalı ÇGKF

Daha önce vurgulandığı üzere, matematiksel modele dayalı ÇGKF'de açısal hız matematiksel modelden alınmaktadır. Bu nedenle yerel durum vektörü dönüölçere dayalı ÇGKF'den farklıdır. Formül (2) durum vektörlerini göstermektedir. Dönüölçere dayalı ÇGKF'de olduğu gibi  $\delta\theta$  yönelim hatasının yerel gösterimidir.  $\Delta\omega$  ise açısal hız hatasının yerel gösterimidir. Ekler kısmında verilen algoritmada (Algoritma 2) hesaplama detayları sözde kod olarak verilmiştir.

$$\Delta x = \begin{bmatrix} \delta\theta \\ \Delta\omega \end{bmatrix} \quad (2)$$

$\delta\theta$  = yönelim hatasının yerel gösterimi

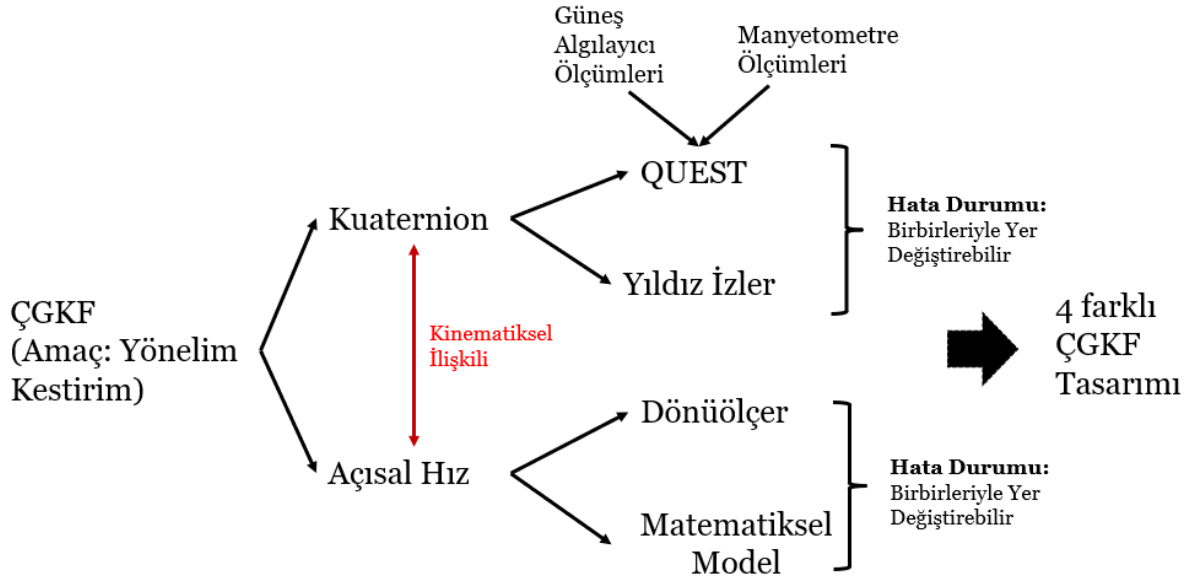
$\Delta\omega$  = açısal hız hatasının yerel gösterimi



Şekil 1: Bayes Ağı, gözlemlenebilir nitelikteki düğümler gölgeli olarak gösterilmiştir.

## Ana Method

ÇGKF, bütün filtreler gibi matematiksel model ve ölçümden oluşur. Filtreyi kuaternion ( $q$ ) ölçümüyle beslemek için iki kaynak vardır: QUEST Algoritması ve Yıldız İzler. Ayrıca,  $q$  ve açısal hız ( $\omega$ ) kinematik olarak ilişkili olduğundan, kuaternion ötelemesi için  $\omega$  da kullanılmalıdır. Aslında  $\omega$  doğrudan dönüölçer ölçümlerinden alınabilir veya Newton-Euler Denklemleri ile elde edilebilir. Öyleyse; toplamda dört farklı filtre ve grafik bulunmaktadır. Bunlar, Şekil 2'de şematik olarak gösterilmiştir.



Şekil 2: ÇGKF’de kullanılan Ölçüm Kaynakları/Algoritmalar/Denklemeler

Bu dört farklı filtre birbirinden farklı arıza/hata kaynaklarına sahiptir. Şekilde (Şekil 1) dört farklı filtre yapısı arasındaki geçişler düğmelerle yapılmaktadır (Düğme 1 ve 2).



Yukarıda belirtildiği gibi, dört farklı filtre varsa, bu dört farklı grafik olduğu anlamına gelir. Bu filtreler soldaki şekilde gösterilmektedir. Şekilde, Yİ, Yıldız İzler anlamına gelir. Ayrıca, filtre yerine grafik ismi kullanılmıştır ve buna göre numaralandırılma yapılmıştır.

Şeklin anlamı daha derinlemesine anlaşılmalı çalışılırsa, aşağıdaki gözlemler yapılabilir:

- \* QUEST, Dönüölçer, Yıldız İzler (Yİ) ve Matematiksel Model (MatModel) ÇGKF içindeki *kontrol noktalarıdır*. Algoritmada sorulan sorular bu noktalarla alakalı olmalıdır.
- \* QUEST ve Yıldız İzler filtrede kullanım amacına göre birbirinin alternatifidir.

- \* QUEST ve Yıldız İzler için geçerli olan alternatiflik durumu, Matematiksel Model ve Dönüölçer için de geçerlidir.

ÇGKF’deki arıza/hatalar birbirinin alternatifi olan algoritma ve ölçümleri kullanarak kontrol edilebilir. Ana strateji bunu dayanmaktadır.

Aşağıdaki örnek algoritmanın mantığını daha net anlatabilmek için oluşturulmuştur.

### Ana Algoritmanın Çalışmasını Gösteren Örnek

1. ÇGKF’de hata gözlemlendi (Ortalama Kare Hatasına bakılarak anlaşılabilir)
2. Hangi filtrenin yani grafiğin aktif olduğunun anlaşılması gerekmektedir (Grafik#1 veya Grafik#2 veya Grafik#3 veya Grafik#4)

**Varsayım:** Grafik#1 kullanıldı.

3. Grafik#1'deki bütün gözlemlenebilir düğümler gözlemlenir.
4. Olasılık hesaplama algoritması koşturulur.
5. Aşağıdaki olasılıklar hesaplanır ve birbirleriyle karşılaştırılır:

$$P(\text{Quest Hatası} \mid \text{ÇGKF Hatası}, \text{Diğer Gözlemlenebilir Düğümler})$$

>

=

<

$$P(\text{Dönüölcer Hatası} \mid \text{ÇGKF Hatası}, \text{Diğer Gözlemlenebilir Düğümler})$$

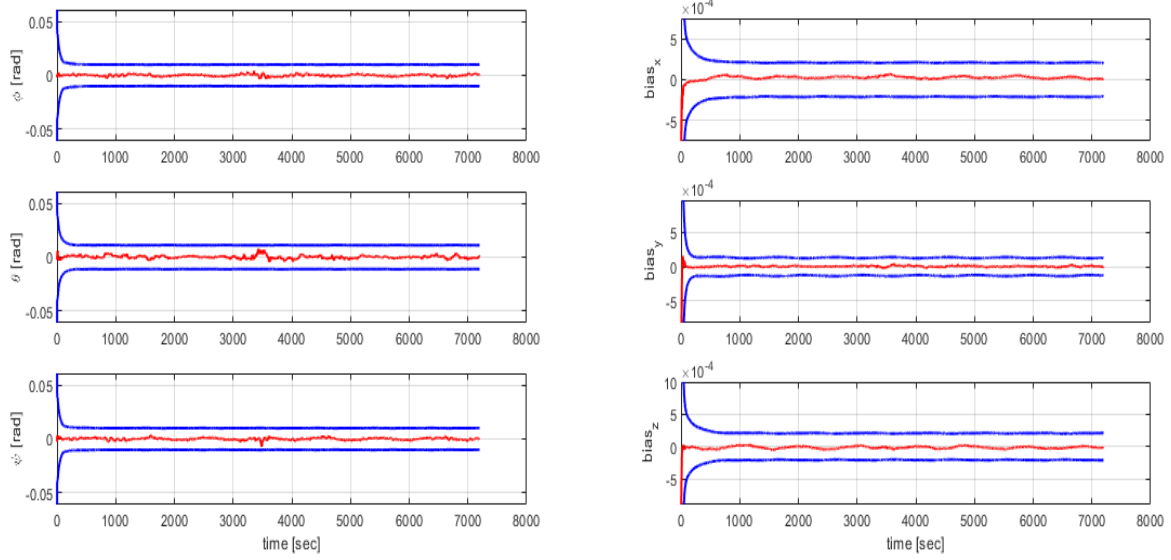
6. Eğer >  $\Rightarrow$  Grafik(Filtre)#1'i Grafik(Filtre)#2 ile değiştir
- Eğer <  $\Rightarrow$  Grafik(Filtre)#1'i Grafik(Filtre)#3 ile değiştir
- Eğer =  $\Rightarrow$  Grafik(Filtre)#1'i Grafik(Filtre)#2 ile değiştir ve Grafik(Filtre)#1'i Grafik(Filtre)#3 ile değiştir, sonra; yeniden gözden geçir.

## UYGULAMALAR VE DEĞERLENDİRME

Bayes Ağındaki şartlı olasılık dağılımlarını bulmak için uzman bilgisinden yararlanılmıştır. 1 durumu hatanın olma olasılığını göstermektedir ve bütün değişkenler ikili yapıdadır. Buna ek olarak; hesaplama algoritması olarak kesin hesaplama-değişken eleme (*Ing. exact inference-variable elimination*) yöntemi kullanılmıştır. Bu yöntem için Python pgmpy kütaphanesinden yararlanılmıştır. Fakat, diğer kodlar MATLAB'da yazılmıştır. Bu yüzden, Windows Komut Satırı MATLAB ve Python'un ortak çalışması için kullanılmıştır.

Testler Güneş senkron bir nanouydu için yapılmıştır. Nanouydu Alçak Dünya yörüngesindedir ve yörünge yaklaşık olarak dairesel olup irtifası 680 [km]'dir. Hesaplamalarda kullanılan sensör parametreleri bu doğrultuda seçilmiş olup, ilgili kaynaktan yararlanılmıştır [Hajiyev and Soken, 2020].

Algoritmanın çalıştığını anlamak için dört farklı test hazırlanmıştır. Ancak hatalı filtre sonuçlarını göstermeden önce hatasız filtre sonuçlarına bakılması gerekmektedir. Şekilde (Şekil 3(a)) kuaternion hatasının karekök ortalaması hatası ve güncellenmiş varyans ( $\pm 3\sigma$ ) değerleri gösterilmiştir. Benzer bir durum Şekil 3(b) için de geçerlidir. Burada ise dönüölcer kayma hatasının karekök ortalaması hatası ve güncellenmiş varyans ( $\pm 3\sigma$ ) değerleri gösterilmiştir. Her iki durumda da karekök ortalama hatası, beklenen varyans aralığında ( $\pm 3\sigma$ ) kalmaktadır. Bu da filtrenin doğru şekilde çalıştığını göstermektedir.



(a) Grafik (Filtre)#1 kuarternion hatasının karekök ortalama hatası (Kırmızı) ve güncellenmiş varyans  $\pm 3\sigma$  değeri (Mavi)

(b) Grafik (Filtre)#1 dönüölçer kayma hatasının karekök ortalama hatası (Kırmızı) ve güncellenmiş varyans  $\pm 3\sigma$  değeri (Mavi)

Şekil 3: Hata Olmadığı Durumda ÇGKF Sonuçları

#### □ Test #1

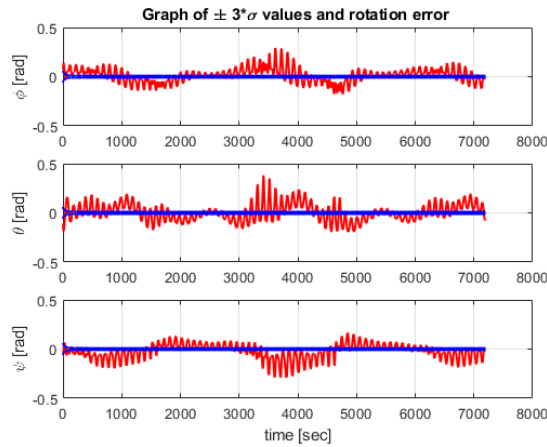
**Amaç:** Bu testteki amaç QUEST algoritmasının bir alt dalı olan “Tork Çubuğu Çalışırken Ölçüm Alınması” düğümündeki hatayı simüle etmektir. “Tork Çubuğu Çalışırken Ölçüm Alınması” düğümünün etkisi manyetometrede kayma olarak gözlemlenebilir. Bu nedenle, manyetometreye kayma eklenmiştir.

**Hata Tipi:** Manyetometre Kayma Hatası - 5000 [nT]

**Gözlemlenebilir Düğümler:** “Tork Çubuğu Çalışırken Ölçüm Alınması” ve “ÇGKF Hatası” düğümleri

**Aktif Grafik (Filtre):** Grafik#1

**Beklenen Sonuç:** Arıza/Hata nedeni QUEST



Şekil 4: Test #1, ÇGKF Hatalı Sonuçları

$$P(QUEST Hatası = 1 | \text{ÇGKF Hatası} = 1, \text{Ölçüm Tork Rod Aktif} = 1) = 0.9985$$

$$P(\text{Dönüölcer Hatası} = 1 | \text{ÇGKF Hatası} = 1, \text{Ölçüm Tork Rod Aktif} = 1) = 0.0832$$

Sonuç = Arıza/Hata nedeni QUEST. Grafik (Filtre)#2'yi kullan (Yıldız İzler + Dönüölcer)

Şekilde (Şekil 4) görüldüğü üzere kuaternion hatasının karekök ortalama hatası, güncellenmiş varyans aralığında ( $\pm 3\sigma$ ) kalmamıştır. Bu durum filtrede bir veya birden fazla hata/arıza olduğunu göstermektedir. Buna ek olarak; Bayes Ağından alınan olasılık sonuçlarında hata/arıza nedenin QUEST olma olasılığı daha yüksektir. Sonuç olarak doğru tahmin yapılmış olup, filtrede QUEST algoritmasının alternatifi olan Yıldız İzlere geçilmiştir.

#### □ Test #2

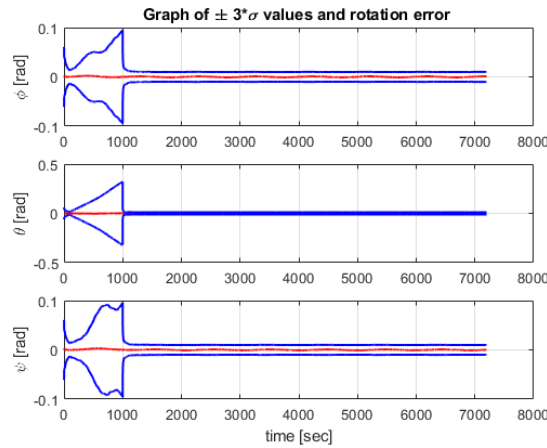
**Amaç:** Bu testteki amaç Yıldız İzlerin alt dalı olan “Parlak Objeye Tarafından Lensin Bloklanması” düğümündeki hatayı simüle etmektir. “Parlak Objeye Tarafından Lensin Bloklanması” düğümünün etkisi Yıldız İzlerin belirli saniyeler arasında ölçüm vermemesi olarak gözlemlenebilir. Bu nedenle, Yıldız İzlerin [100,1000] [s] aralığında ölçüm vermediği düşünülmüştür ve simülasyon buna göre çalıştırılmıştır.

**Hata Tipi:** Yıldız İzler [100,1000] [s] arasında ölçüm vermediğinde

**Gözlemlenebilir Düğümler:** “Parlak Objeye Tarafından Lensin Bloklanması” ve “ÇGKF Hatası” düğümleri

**Aktif Grafik (Filtre):** Grafik#2

**Beklenen Sonuç:** Arıza/Hata nedeni Yıldız İzler



Şekil 5: Test #2, ÇGKF Hatalı Sonuçları

$$P(\text{Dönüölcer Hatası} = 1 | \text{ÇGKF Hatası} = 1, \text{Parlak Objeye Ta. Lens Kapatıldı} = 1) = 0.0897$$

$$P(\text{Yİ Hatası} = 1 | \text{ÇGKF Hatası} = 1, \text{Parlak Objeye Ta. Lens Kapatıldı} = 1) = 0.9820$$

Sonuç = Arıza/Hata nedeni Yıldız İzler. Grafik (Filtre)#1'i kullan (QUEST + Dönüölcer)

Şekilde (Şekil 5) görüldüğü üzere güncellenmiş varyans değerleri ( $\pm 3\sigma$ ) [100,1000] [s] aralığında artma eğilimindedir. Bu durum, bize güncellenmiş varyans değerlerinin ıraksama



eğiliminde olduğunu göstermektedir ve filtrenin doğru çalışmadığını söyleyebiliriz. Buna ek olarak; Bayes Ağından alınan olasılık sonuçlarında hata/arıza nedenin Yıldız İzler olma olasılığı daha yüksektir. Sonuç olarak; doğru tahmin yapılmış olup, filtrede Yıldız İzlerin alternatifi olan QUEST algoritmasına geçiş yapılmıştır.

### □ Test #3

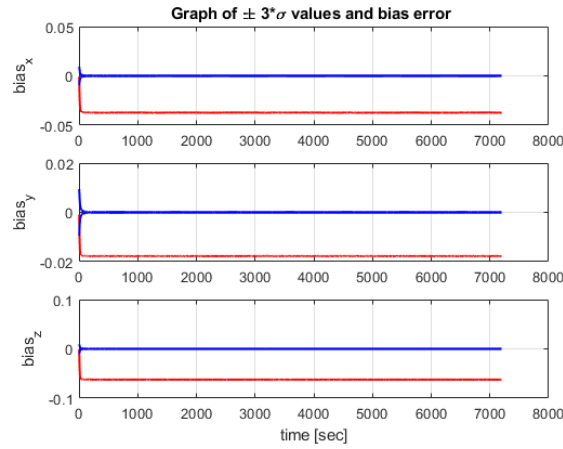
**Amaç:** Bu testteki amaç dönüölçerin bir alt dalı olan “Varyans Değişimi” düğümündeki hatayı simüle etmektir. “Varyans Değişimi” düğümünün etkisi kullanılan Dönüölçerin gürültüsünün artırılmasıyla gözlemlenebilir.

**Hata Tipi:** Dönüölçer Gürültüsünün Artırılması -  $1000\sigma_g$

**Gözlemlenebilir Düğümler:** “Varyans Değişimi” ve “ÇGKF Hatası” düğümleri

**Aktif Grafik (Filtre):** Grafik#1

**Beklenen Sonuç:** Arıza/Hata nedeni dönüölçer



Şekil 6: Test #3, ÇGKF Hatalı Sonuçları

$$P(\text{QUEST Hatası} = 1 \mid \text{ÇGKF Hatası} = 1, \text{Varyans Değişimi} = 1) = 0.9975$$

$$P(\text{Dönüölçer Hatası} = 1 \mid \text{ÇGKF Hatası} = 1, \text{Varyans Değişimi} = 1) = 0.9089$$

Sonuç = Arıza/Hata nedeni QUEST. Sonuç doğru değil. Olasılık tablosunun güncellenmesi!

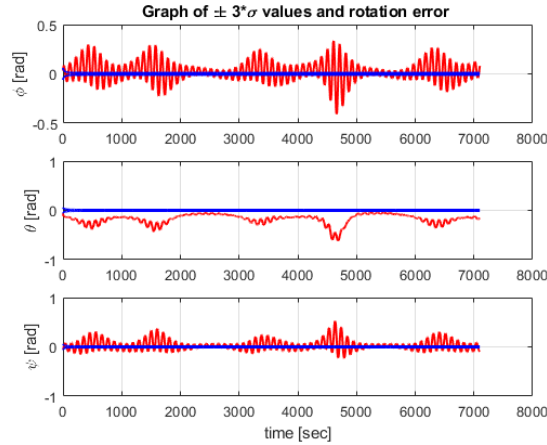
Şekilde (Şekil 6) görüldüğü üzere kuaternion hatasının karakök ortalama hatası, güncellenmiş varyans aralığında ( $\pm 3\sigma$ ) aralığında kalmamaktadır. Bu durum, açıkça filtrenin yanlış çalıştığını göstermektedir. Buna ek olarak; Bayes Ağından alınan olasılık sonuçlarında hata/arıza nedeni dönüölçer olarak beklenmektedir. Ancak; yukarıda verilen olasılık değerlerine baktığımızda, Bayes Ağı yüksek olasılıklı hata/arıza nedenini QUEST olarak göstermektedir. Bu nedenle, ileriki çalışmalarda olasılık tablosu güncellenmesi söz konusudur. Bunun haricinde; yine ileriki çalışmalarda, düğümler arası hassaslık analizleri yapılabilir.

### □ Test #4

**Amaç:** Bu testteki amaç QUEST algoritmasının bir alt dalı olan “Yörünge Belirleme Hatası” düğümündeki hatayı simüle etmektir. “Yörünge Belirleme Hatası” düğümünün etkisi IGRF sonuçlarınının 100 [s] kaydırılması ile gözlemlenebilir.

**Hata Tipi:** IGRF Hatası - 100 [s] ölçümlerin kaydırılması

**Gözlemlenebilir Düğümler:** “Yörünge Belirleme Hatası” ve “ÇGKF Hatası” düğümleri

**Aktif Grafik (Filtre): Grafik#1****Beklenen Sonuç: Arıza/Hata nedeni QUEST**

Şekil 7: Test #4, ÇGKF Hatalı Sonuçları

$$P(QUEST Hatası = 1 \mid \text{ÇGKF Hatası} = 1, IGRF Hatası = 1) = 0.9988$$

$$P(\text{Dönüölçer Hatası} = 1 \mid \text{ÇGKF Hatası} = 1, IGRF Hatası = 1) = 0.0831$$

Sonuç = Arıza/Hata nedeni QUEST. Grafik (Filtre)#2'yi kullan (Yıldız İzler + Dönüölçer)

Şekilde (Şekil 7) görüldüğü üzere kuaternion hatasının karakök ortalama hatası, güncellenmiş varyans aralığında ( $\pm 3\sigma$ ) aralığında kalmamaktadır. Buna ek olarak; Bayes Ağından alınan olasılık sonuçlarında hata/arıza nedeni QUEST algoritması olarak beklenmektedir. Sonuç olarak; doğru tahmin yapılmış olup, filtrede QUEST algoritmasının alternatifi olan Yıldız İzlere geçiş yapılmıştır.

**SONUÇ**

Bu çalışmada küçük uydular için dört farklı ÇGKF tasarımına gidilmiş ve bu farklı tasarımlar uydunun yönelim kontrolü için kullanılmıştır. Küçük uydu üzerinde kullanılan herhangi bir ÇGKF'deki arıza/hata gözlemlenebilirlik algoritmaları yardımıyla anlaşılmıştır. Daha sonrasında arıza/hatanın nedenini anlamak için Bayes Ağı kullanılmıştır. Bu ağ yardımıyla hata/arıza olası nedenleri için olasılıklar hesaplanmıştır; daha sonrasında, gerekli aksiyon alınmıştır. İlerleyen çalışmalarda Bayes Ağındaki koşullu olasılık değerleri uzman bilgisinden ve küçük uydu üzerinden alınan verilerle makine öğrenmesi algoritmaları çalıştırılarak geliştirilebilir. Bu çalışmada sadece uzman bilgisinden yararlanılmıştır.

**Kaynaklar**

Daphne Koller, N. F. 2009 . *Probabilistic Graphical Models: Principles and Techniques*. Adaptive Computation and Machine Learning series. The MIT Press, 1 edition.

Hajiyev, C. and Soken, H. E. 2020 . *Fault Tolerant Attitude Estimation for Small Satellites*. CRC Press, 1st edition.

- Lee, D. Y., Park, H., Romano, M., and Cutler, J. 2018 . Development and experimental validation of a multi-algorithmic hybrid attitude determination and control system for a small satellite. *Aerosp. Sci. Technol.*, 78:494–509.
- Markley, F. L. and Crassidis, J. L. 2014 . *Fundamentals of Spacecraft Attitude Determination and Control*. Springer.
- Nakasuka, S., Miyata, K., Tsuruda, Y., Aoyanagi, Y., and Matsumoto, T. 2018 . Discussions on attitude determination and control system for micro/nano/pico-satellites considering survivability based on Hodoyoshi-3 and 4 experiences. *Acta Astronaut.*, 145:515–527.
- Shunichiro Nomura, Ryohei Takashi, M. I. K. O. T. O. S. I. R. F. 2017 . Initial Design of EQUULEUS Attitude Determination and Control System: How to Design an ADCS with High Reliability for a Deep Space CubeSat. *International Symposium on Space Technology and Science*.
- Yanning, G., Youmin, G., Sheng, G., and Guangfu, M. 2016 . The node selection of satellite attitude control system based on sensitivity function in Bayesian network. In *2016 Chinese Control and Decision Conference (CCDC)*, pages 7047–7051. IEEE.

## Ekler

---

### Algorithm 1 Dönüölçere Dayalı ÇGKF Algoritması

---

$$x_{k+1} = Ax_k + Bu_k + Gw_k$$

$$y_k = Cx_k + Hv_k$$

$$x_0 \sim N(x_0; \bar{x}_0, \Sigma_0)$$

$$w_k \sim N(w_k; 0, Q)$$

$$v_k \sim N(v_k; 0, R)$$

$w_k$  = süreç gürültüsü

$v_k$  = ölçüm gürültüsü

$$C = [I_3 \quad 0_{3 \times n}]$$

$$H = I_3$$

**For**  $k = 0, \dots, N$

% Ölçüm Güncellemesi

$$\hat{y}_{k|k-1} = \delta\hat{\theta}_k^- = 2 \frac{(\hat{q}_{k|k-1} \otimes \hat{q}_{k|k-1}^{-1})_{1:3}}{(\hat{q}_{k|k-1} \otimes \hat{q}_{k|k-1}^{-1})_4} + [I_3 \quad 0_{3 \times n}] \begin{bmatrix} \delta\hat{\theta}_k^- \\ \Delta\hat{\xi}_k^- \end{bmatrix}$$

$$y_k = \delta\theta_k = 2 \frac{(q_{k,measured} \otimes \hat{q}_{k|k-1}^{-1})_{1:3}}{(q_{k,measured} \otimes \hat{q}_{k|k-1}^{-1})_4}$$

$$S_k = C * \Sigma_{k|k-1} * C^T + HRH^T$$

$$K_k = \Sigma_{k|k-1} * C^T * S_k^{-1}$$

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k * (y_k - \hat{y}_{k|k-1}) = \begin{bmatrix} \delta\hat{\theta}_k^+ \\ \Delta\hat{\xi}_k^+ \end{bmatrix} = \begin{bmatrix} \delta\hat{\theta}_{k|k} \\ \Delta\hat{\xi}_{k|k} \end{bmatrix}$$

$$\Sigma_{k|k} = \Sigma_{k|k-1} - K_k S_k K_k^T$$

% Sıfırlama

$$\hat{q}_{k|k} = \delta q(\delta\hat{\theta}^+) \otimes \hat{q}_{k|k-1} = \frac{1}{\sqrt{1 + \|\delta\hat{\theta}^+/2\|^2}} \begin{bmatrix} \delta\hat{\theta}^+/2 \\ 1 \end{bmatrix} \otimes \hat{q}_{k|k-1}$$

$$\hat{\xi}_{k|k} = \hat{\xi}_{k|k-1} + \Delta\hat{\xi}_{k|k}$$

$$\delta\theta_{k+1|k} = 0_{vec} \quad \& \quad \Delta\xi_{k+1|k} = 0_{vec}$$

% Öteleme

$$\hat{\omega}_k = \omega_{k,measured} - \hat{\xi}_{k|k}$$

$$\hat{\Omega}_k = \begin{bmatrix} 0 & \hat{\omega}_3 & -\hat{\omega}_2 & \hat{\omega}_1 \\ -\hat{\omega}_3 & 0 & \hat{\omega}_1 & \hat{\omega}_2 \\ \hat{\omega}_2 & -\hat{\omega}_1 & 0 & \hat{\omega}_3 \\ -\hat{\omega}_1 & -\hat{\omega}_2 & -\hat{\omega}_3 & 0 \end{bmatrix}$$

$$\hat{q}_{k+1|k} = (I + \hat{\Omega}_k \Delta t / 2) \hat{q}_{k|k}$$

$$\hat{\xi}_{k+1|k} = \hat{\xi}_{k|k}$$

$$A_k = \begin{bmatrix} -\hat{\omega}_k \times & -I_{3 \times n} \\ 0_{n \times 3} & 0_{n \times n} \end{bmatrix} + I_{(n+3) \times (n+3)}$$

$$\Sigma_{k+1|k} = A_k \Sigma_{k|k} A_k^T + GQG^T$$

**End For**

---

**Algorithm 2** Matematiksel Modele Dayalı ÇGKF Algoritması

$$x_{k+1} = Ax_k + Bu_k + Gw_k$$

$$y_k = Cx_k + Hv_k$$

$$x_0 \sim N(x_0; \bar{x}_0, \Sigma_0)$$

$$w_k \sim N(w_k; 0, Q)$$

$$v_k \sim N(v_k; 0, R)$$

$w_k$  = süreç gürültüsü

$v_k$  = ölçüm gürültüsü

$$C = [I_3 \quad 0_{3 \times n}]$$

$$H = I_3$$

**For**  $k = 0, \dots, N$

% Ölçüm Güncellemesi

$$\hat{y}_{k|k-1} = \delta \hat{\theta}_k^- = 2 \frac{(\hat{q}_{k|k-1} \otimes \hat{q}_{k|k-1}^{-1})_{1:3}}{(\hat{q}_{k|k-1} \otimes \hat{q}_{k|k-1}^{-1})_4} + [I_3 \quad 0_{3 \times n}] \begin{bmatrix} \delta \hat{\theta}_k^- \\ \Delta \hat{\xi}_k^- \end{bmatrix}$$

$$y_k = \delta \theta_k = 2 \frac{(q_{k,measured} \otimes \hat{q}_{k|k-1}^{-1})_{1:3}}{(q_{k,measured} \otimes \hat{q}_{k|k-1}^{-1})_4}$$

$$S_k = C * \Sigma_{k|k-1} * C^T + H R H^T$$

$$K_k = \Sigma_{k|k-1} * C^T * S_k^{-1}$$

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k * (y_k - \hat{y}_{k|k-1}) = \begin{bmatrix} \delta \hat{\theta}^+ \\ \Delta \hat{\xi}^+ \end{bmatrix} = \begin{bmatrix} \delta \hat{\theta}_{k|k} \\ \Delta \hat{\xi}_{k|k} \end{bmatrix}$$

$$\Sigma_{k|k} = \Sigma_{k|k-1} - K_k S_k K_k^T$$

% Sıfırlama

$$\hat{q}_{k|k} = \delta q(\delta \hat{\theta}^+) \otimes \hat{q}_{k|k-1} = \frac{1}{\sqrt{1 + \|\delta \hat{\theta}^+ / 2\|^2}} \begin{bmatrix} \delta \hat{\theta}^+ / 2 \\ 1 \end{bmatrix} \otimes \hat{q}_{k|k-1}$$

$$\hat{\xi}_{k|k} = \hat{\xi}_{k|k-1} + \Delta \hat{\xi}_{k|k}$$

$$\delta \theta_{k+1|k} = 0_{vec} \quad \& \quad \Delta \xi_{k+1|k} = 0_{vec}$$

% Öteleme

$$\hat{\Omega}_k = \begin{bmatrix} 0 & \hat{\omega}_3 & -\hat{\omega}_2 & \hat{\omega}_1 \\ -\hat{\omega}_3 & 0 & \hat{\omega}_1 & \hat{\omega}_2 \\ \hat{\omega}_2 & -\hat{\omega}_1 & 0 & \hat{\omega}_3 \\ -\hat{\omega}_1 & -\hat{\omega}_2 & -\hat{\omega}_3 & 0 \end{bmatrix}$$

$$\hat{q}_{k+1|k} = (I + \hat{\Omega}_k \Delta t / 2) \hat{q}_{k|k}$$

$$\dot{\xi} = J^{-1} * (\xi_{k|k} \times (J * \xi_{k|k}))$$

$$\xi_{k+1|k} = \xi_{k|k} + \dot{\xi} * dt$$

$$A_k = \begin{bmatrix} -\hat{\omega}_k \times & -I_{3 \times n} \\ 0_{n \times 3} & J^{-1}(- (J \xi_{k|k}) \times + (\xi_{k|k} J) \times) \end{bmatrix} + I_{(n+3) \times (n+3)}$$

$$\Sigma_{k+1|k} = A_k \Sigma_{k|k} A_k^T + G Q G^T$$

**End For**