

## ÇARPIŞMASI MUHTEMEL CİSİMLERİN MANEVRALARININ HESAPLANMASI VE YÖRÜNGELERİNİN GÖRSELLEŞTİRİLMESİ

Uzay Tuğcular<sup>1</sup>, Berkay Daşer<sup>2</sup> ve Alper Akgül<sup>3</sup>  
Türk Havacılık Uzay Sanayii, Ankara

### ÖZET

*Bu çalışmada bir uydu ve bir cismin çarpışma olasılıklarının en yüksek olduğu tarihteki yörüngelerini CDM yardımıyla VTS programı üzerinde görselleştirebilen ve kullanıcının seçtiği manevrayı hesaplayıp uygulayarak tekrar görselleştirebilen ÇÖMAS isimli bir bilgisayar programı geliştirilmiştir. Yazılan uygulama Java tabanlı olup, çalışmada OREKIT 10.0 kütüphanesi kullanılmıştır. Çalışmanın temel hedefi çarpışma manevrasının tipini kolayca belirleyebilmek, manevra işlemleri için gerekli olan süreyi azaltmak ve planlanan manevranın etkisini görselleştirmektir. Bu bağlamda öncelikle çarpışma ikaz mesaj yapısı tanıtılmış ve simülasyon algoritması matematiksel olarak açıklanmıştır. VTS programı lisans kısıtlamaları gereği geliştirilen bilgisayar programı sadece kar amacı olmayan uzay görevlerinde kullanılabilecektir.*

### GİRİŞ

Her yıl giderek yaygınlaşan uzay görevleri, takım uydular ve bunlara bağlı olarak sayısı artan uzay çöpleri, hâlihazırda aktif olarak çalışan sistemleri tehdit etmekte ve çeşitli kazalara neden olmaktadır. 2009 yılında, 1996'da ömrünü doldurmuş olan Strela-2M ve aktif olarak görevini sürdüren Iridium-33 uydusu çarpışarak büyük bir enkaz yığını geride bırakmışlardır. 2009'dan bu yana en az 10 çarpışma daha yaşanmış, günümüzde bu tür kazaların yaşanmaması ve uzay çöpu artışının ivmelenmemesi adına, her uydu operatörünün kendi uydusu için gerekli önlemleri alması bir zorunluluk haline gelmiştir. Günümüzde çeşitli otoriteler tarafından Dünya'nın belirli yerlerine konuşlandırılan yer istasyonları ile belirli bir büyüklüğün üzerinde olan cisimler takip edilerek yörüngeleri yüksek tutarlılıkta belirlenmektedir. Bu otoritelerden biri olan CSpOC, her gün yapılan hesaplamalarla potansiyel çarpışmaları önceden tahmin etmekte ve bu çarpışmaları önlemek amacıyla CDMleri uydu operatörleriyle paylaşmaktadır. CDM içerisinde iki cismin çarpışma olasılığını, TCA ve iska mesafesini, TCA'daki yörünge elemanlarını ve olasılık bulutu matrisini barındırmaktadır. Uydu operatörleri tarafından bu mesajlar kullanılarak çarpışmaların önlenmesi adına çeşitli manevralar icra edilmektedir. Normal şartlar altında muhtemel çarpışma ihtimali zamanından 72 saat önce gönderilen CDMlerin 10-12 saat önce gönderilmesi durumunda uydu operatörleri zamanla yarışmaktadır. CDM, muhtemel çarpışma anına kadar her 6 saatte bir gönderilmeye devam etmektedir, ardışık olarak alınan CDM içinde çarpışma karakteristiğinin değişmesi durumunda manevra hesaplamalarına baştan başlanması gerekebilmekte bu da çok değerli olan hesaplama ve uyduya komut yükleme zamanının azalmasına sebep olmaktadır. Bu gibi durumlarda manevra hesabı için gerekli olan süreyi en aza indirmek büyük önem taşımaktadır. Bu çalışmada özetlenen görselleştirme aracının diğer bir amacı da her 6 saatte bir gelen CDM mesajlarının otomatik olarak okunarak çok hızlı bir şekilde daha önceki CDM mesajları ile karşılaştırılmasının sağlanmasıdır.

Bu kapsamda yapılan çalışmalardan bir tanesi GISTDA tarafından geliştirilen çarpışma risk analiz ve görselleştirme aracı ZIRCON'dur [Channumsin ve diğerleri, 2019]. ZIRCON, THEOS uydusu için yakın irtifadaki uyduların yörüngelerini belirledikten sonra çarpışma risk analizi yapıp, yaptığı analiz sonucunda elde ettiği olasılık bulutu matrislerini, TCA'daki uydu konumlarını ve çarpışmadan kaçınma manevrası yapılması durumunda uyduların yeni konumlarını görselleştiren bir araçtır. Bu çalışmada ZIRCON uygulamasına benzer bir yaklaşım yapılmıştır. Temel olarak CDM girdi olarak kullanılır ve TCA'daki uydu pozisyonları incelenerek manevra yapıp yapılmayacağına karar verilir. Karar verildikten sonra hedefleme algoritması ile manevra hesaplanır. Bu hedefleme algoritması iki

<sup>1</sup> Yer Sistemleri Tasarım Mühendisi, uzay.tugcular@tai.com.tr

<sup>2</sup> Yer Sistemleri Tasarım Mühendisi, berkay.daser@tai.com.tr

<sup>3</sup> Yer Sistemleri Tasarım Mühendisi, alper.akgul@tai.com.tr

noktalı sınır değer problemi çözümlerine dayanmaktadır [Chapra ve Canale,2015, s. 151-157] [Chapra ve Canale,2015, s. 782-789]. Hesaplanan manevra ve CDM ile gelen yörüngeler belirli süreler için ilerletilirler. Bu yörünge hesaplamaları için OREKIT kütüphanesi kullanılmıştır [Fernandez-Martin ve diğerleri, 2010][Maisonobe ve diğerleri, 2010]. İlerletilen bu yörüngeler için bir gökgünlüğü oluşturulur. Bu gökgünlüğü formatı OEM olarak adlandırılır ve yazılan uygulama ile TCP-IP protokolleri kullanılarak VTS programında yeni bir senaryo oluşturulur. Oluşturulan bu senaryoda OEM dosyası kullanılır ve geliştirilen uygulama tarafından çalıştırılarak görselleştirme işlemi gerçekleştirilir.

## YÖNTEM

Bu çalışma yörüngedeki aktif bir uydunun ikincil bir cisimle olası çarpışma durumunu görselleştirmek, gerektiği durumda çarpışmadan kaçınma manevrası için itki vektörünün belirleme sürecini kolaylaştırmak ve karar verilen manevra tipinin hedefleme algoritması ile hesaplanan itki süresi boyunca uygulanması durumunda manevranın sonucunu görselleştirmek için yapılmıştır. Bu bölümde yapılan görselleştirme aracı için izlenen süreçten bahsedilecektir.

JAVA ile yazılan bu görselleştirme aracı, kullanıcıdan CDM'i girdi olarak istemektedir. CCSDS standartlarına göre oluşturulan çarpışma ikaz mesajları; hesaplanan çarpışma anının en yüksek olduğu zamanda çarpışma ihtimali bulunan iki cisim için de gerekli verileri içermektedir.

CDM, görselleştirme aracı tarafından okunur ve işlenir. CDM, TCA anını, bu andaki iska mesafesini, görelî hızları, çarpışma olasılığını, iki uydu için ITRF referans koordinat sisteminde tanımlanmış olan pozisyon ve hız vektörlerini bulundurmaktadır. Bu verilere ek olarak uydulara özgün ve yörünge belirleme işleminden sonra elde edilen kovaryans ve yapılan ölçüm sayısı gibi çıktılar da bu mesaj içinde bulunmaktadır.

CDM oluşturulan görselleştirme aracına girdi olarak verildikten sonra uygun formatta olup olmadığı kontrol edilir. Bu kontrol CCSDS standartlarında belirtilen ayırt edici satırların kontrolüne dayanır. Her CDM dosyası birebir aynı değildir. Değişkenlik gösterebilen satırlar bu dosyaların görselleştirme aracı tarafından işlenmesine engel değildir, bu durumlar göz önünde bulundurularak bu dosyaları okuyup işlemesi için gerekli kod parçası yazılmış ve test edilmiştir.

İşlenen CDM ile ITRF koordinat sistemine göre verilen konum ve hız vektörleri elde edilir. Bu hız ve konum vektörleri ile oluşturulan yörüngeler, PrinceDormand8(53) nümerik çözüm yöntemi kullanılarak, kullanıcının belirlediği zaman aralığında ileri ve geri yönlerde ilerletilir. İleri ve geri yönde ilerletilen bu yörüngelerin zaman ve konum verileri OEM dosya formatında yazılarak VTS uygulamasına gönderilir. VTS uygulaması, gönderilen bu dosya içinde bulunan zaman etiketli pozisyon verilerini alarak Celestia uygulamasına gönderir ve görselleştirmeyi gerçekleştirir.

Çarpışma olasılığının değerine göre uydu operatörünün çarpışmadan kaçınma manevrası yapması gerekmektedir. Bu durumda uygulanması gereken manevra tipine karar verilmeli ve manevra sonunda ulaşılmak istenen iska mesafesi belirlenmelidir. Çarpışma önleme manevraları ikiye ayrılabilir. Bunlar alçaltma ve yükseltme manevralarıdır. Iska mesafesi için ise bir eşik değeri seçilebilir. Manevranın sonucuna etki edecek değişkenlerden iki tanesi uygulanacak itkinin süresi ve itkinin başlayacağı zamandır. Kullanıcıdan itkinin başlayacağı zaman bilgisi, itki kuvveti, spesifik itki katsayısı ve uydu ağırlığı gibi bilgiler alınarak hedefleme algoritması ile itki süresi hesaplanır. İtki süresi, itki sisteminin ne kadar açık kalacağını saniye cinsinden ifade eden bir parametre olup manevra zamanı ise TCA anından kaç yörünge önce yapılması istendiğidir.

Bu sayede kullanıcı, karar verdiği manevra tipini belirlenen süre boyunca ve belirlediği yörünge sayısı kadar önce uygulayıp sonuçlarını görselleştirme aracı vasıtası ile görebilir. Manevra yapılan yörünge manevranın uygulandığı zamandan TCA zamanına kadar ilerletilir. Kullanıcı ara yüzündeki seçenek ile hem manevra yapılmamış yörüngeleri hem de manevra yapılmış yörüngeleri görebilir.

**Uydu Dinamik Modeli:** Uydu dinamik modeli aşağıdaki gibi tanımlanmıştır.

$$\vec{a} = \vec{a}_{\oplus} + \vec{a}_{\odot} + \vec{a}_{*} + \vec{a}_T$$

$$\vec{a}_{\oplus} = \text{Kepler ivmesi}$$

$$\vec{a}_{\odot} = \text{Yerçekimi potansiyel ivmeleri}$$

$$\vec{a}_{*} = \text{Üçüncül cisim ivmeleri}$$

$$\vec{a}_T = \text{İtke ivmesi}$$

Yerçekimi potansiyel ivmeleri, 21. harmoniklere kadar kullanılmıştır. Üçüncül cisim ivmeleri, Güneş ve Ay'ın çekim kuvvetleri içermektedir. İtke ivmesi, manevranın uygulanması durumunda hesaba katılmaktadır.

**Manevra Tipleri:** Çarpışmadan kaçınma manevrası için yapılabilecek iki farklı tip manevra vardır. Bu manevralar yükseltme ve alçaltma manevraları olup yörünge düzlemi içinde gerçekleştirilir. Operatör, cisimlerin en yakın geçiş zamanlarındaki pozisyon vektörlerinin boyuna göre manevra tipine karar verebilir ve iki cisim arasındaki mesafe bir eşik değere veya istenen bir değere göre operatör tarafından seçilerek hedefleme algoritmasına gönderilir.

Yükseltme manevrası, birincil uydunun konum vektörü boyunun ikincil uydunun konum vektörü boyundan büyük olduğu durumlarda kullanılabilecek bir yöntemdir. Bu manevrada amaç birincil uydu ile ikincil uydu arasındaki mesafeyi hız vektörü yönünde, hedefleme algoritması ile hesaplanan itki süresi boyunca kuvvet uygulayarak iki cisim arasındaki mesafeyi arttırmaktır.

Alçaltma manevrası ise, birincil uydunun pozisyon vektörü boyunun ikincil uydunun pozisyon vektörü boyundan küçük olduğu durumlarda kullanılabilecek bir yöntemdir. Bu manevrada amaç birincil uydu ile ikincil uydu arasındaki mesafeyi hız vektörünün tersi yönünde, hedefleme algoritması ile hesaplanan itki süresi boyunca kuvvet iki cisim arasındaki mesafeyi arttırmaktır.

**Hedefleme Algoritması:** Hedefleme algoritması, gerçekleştirilecek olan manevranın hedeflenen bir yörüngeyi tutturması için geliştirilmiş bir algoritmadır. Bu algoritma temel olarak "İki Noktalı Sınır Değer" problemlerinin atış yöntemi ile çözülmesine dayanır.

Söz gelimi  $f''(x, f(x), f'(x))$  fonksiyonu lineer olmayan diferansiyel bir denklem olsun ve  $f(x) \in \mathcal{R}$ . Sınır değerleri de  $f(x_0) = f_0$  ve  $f(x_1) = f_1$  olarak tanımlayalım. Bu çalışma için  $f_1$  çarpışma ikazı gelen iki cisim arasındaki pozisyon farktır. Sınır değer problemini şu şekilde başlangıç değer problemi olarak yazabiliriz;

$$f''(x, f(x), f'(x)), \quad f(x_0) = f_0, \quad f'(x_0) = A \quad (1)$$

Bu problemdeki  $A$  değeri itki süresi olarak kabul edilirse, sınır değerlerde  $f_1$  sonucuna ulaşılabilmesi için harici bir fonksiyon şu şekilde tanımlanabilir;

$$G(A) = f(x_1; A) - f(x_1) = 0 \quad (2)$$

$f(x_1; A)$  fonksiyonu başlangıç değer probleminde  $x_1$  sınır koşulunu sağlayan  $A$  değerinin verilerek çözülen  $f''(x, f(x), f'(x))$ dir.  $G(A)$  fonksiyonunda görüldüğü üzere hedefleme problemi bir kök bulma problemine indirgenmiştir. Denklemlerin lineer olmaması sebebi ile nümerik kök bulma yöntemlerinden biri olan Newton-Raphson metodu kullanılabilir. Bu bağlamda iteratif süreç şu şekildedir;

$$A_{i+1} = A_i - \frac{G(A_i)}{G'(A_i)} \quad (3)$$

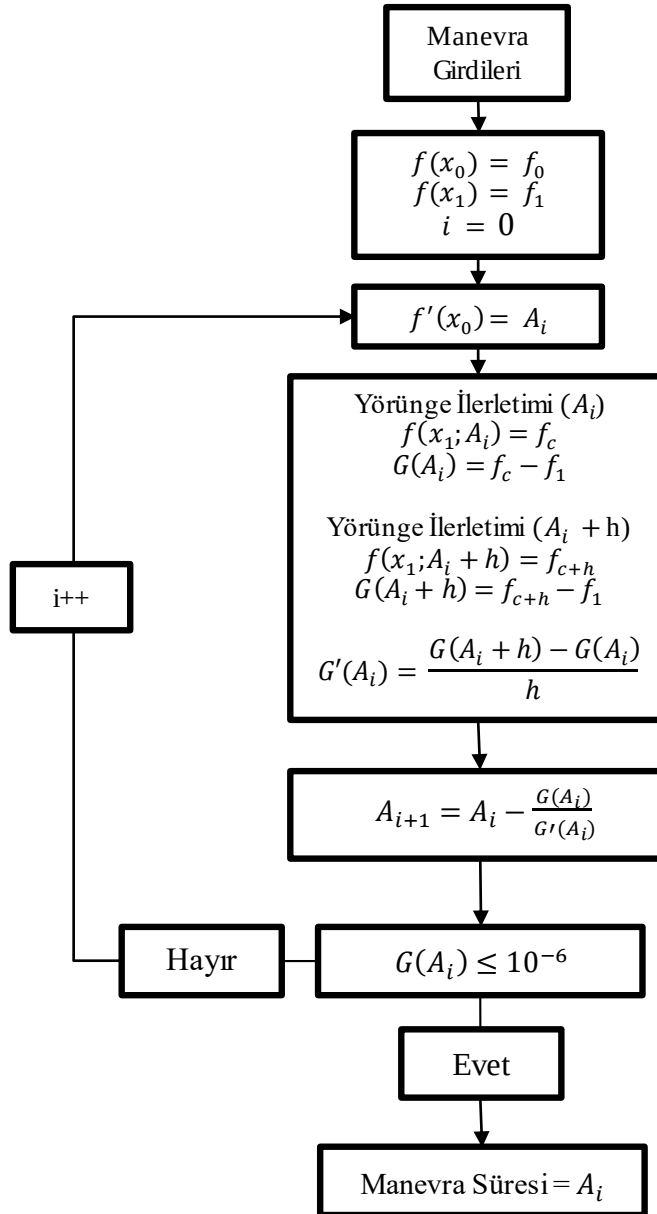
Görüldüğü üzere yöntem gereği  $G'(A)$ 'nın bilinmesi gerekmektedir. Başlangıç değer problemindeki diferansiyel denklem itki süresine bağlı lineer olmayan bir fonksiyondur. Sonuç olarak  $G'(A)$ 'nın nümerik olarak hesaplanması gerekmektedir. Nümerik türev yöntemlerinden sonlu farkların "ileri yönde fark" yaklaşımı kullanılmıştır. Bu bağlamda  $G'(A)$  şu şekilde ifade edilebilir;

$$G'(A) = \frac{G(A+h) - G(A)}{h} \quad (4)$$

Denklem 4'te verilen  $h$  "saptırma adımı" olarak adlandırılmakta ve bu uygulama için  $10^{-6}$  olarak seçilmiştir.

Hedefleme algoritması doğası gereği iteratif bir süreç olduğu için yakınsamadığı durumlar olabilmektedir. Bu gibi durumlar kötü seçilmiş başlangıç tahmininden kaynaklanabilmektedir. Yakınsamayan durumlar için bir hata mesajı oluşturulmuştur. Ayrıca maksimum iterasyon sayısı 50 olarak seçilmiş ve 50 iterasyon sonrasında bir uyarı mesajı ile algoritma durdurularak kullanıcıya sonuncu iterasyondaki itki süresi gösterilmektedir.

Hedefleme algoritmasının yapısı kısaca şu şekilde özetlenebilir;



Şekil 1 Hedefleme Algoritması

Oluşturulan hedefleme algoritması hem GMAT hem de FreeFlyer programlarında bulunan hedefleme algoritması sonuçları ile karşılaştırılmıştır. Karşılaştırma problemi, oluşturulan bir yörünge için "yerberi" geçişi sırasında hız vektörü yönünde ateşleme yapılarak, "yeröte" mesafesini 12000km'ye çıkartmayı hedeflemektedir.

Karşılaştırma için dinamik model sadeleştirilmiş ve sadece kepler ivmesi (iki cisim problemi) kullanılmıştır. Karşılaştırma problemi için aşağıdaki yörünge ve manevra girdileri kullanılmıştır.

Tablo 1 Test Yörüngesi, ECI’de Oluşturulmuş Kartezyen Bileşenleri

|                      | X    | Y    | Z    |
|----------------------|------|------|------|
| <b>Pozisyon (km)</b> | 7100 | 0    | 1300 |
| <b>Hız (km/s)</b>    | 0    | 7.35 | 1    |

Tablo 2 Yereöte Hedeflemesi İçin Manevra Girdileri.

|                | İtke Kuvveti(N) | İtke Katsayısı(s) | Hedef Yeröte(km) | İlk Tahmin(s) |
|----------------|-----------------|-------------------|------------------|---------------|
| <b>Manevra</b> | 1000            | 300               | 12000            | 200           |

GMAT, FreeFlyer ve hedefleme algoritmasında elde edilen sonuçlar şu şekildedir.

Tablo 3 FreeFlyer, GMAT ve Hedefleme Algoritması Sonuçları

|                               | FreeFlyer    | GMAT         | Hedefleme Algoritması |
|-------------------------------|--------------|--------------|-----------------------|
| <b>İtke Süresi(s)</b>         | 1212.0460037 | 1212.046179  | 1212.046180           |
| <b>Elde Edilen Yeröte(km)</b> | 11999.998421 | 11999.999998 | 12000.000000          |
| <b>Uydu Ağırlığı</b>          | 1194.019014  | 1194.018953  | 1194.018953           |

Sonuç olarak hedefleme algoritması GMAT ve FreeFlyer programları ile doğrulanmıştır. Her bir programda tutarlılık 1e-6 olarak seçilmiş, ancak FreeFlyer programı maksimum iterasyon sayısına ulaştığı için, sonuçlar 1e-3 tutarlılığında verilmiştir. Bu bağlamda, hedefleme algoritmasında hesaplanan manevra, istenen yeröte uzaklığını 1e-6 tutarlılığında 12000km’ye çıkartmayı başarmıştır.

**VTS Programı Zaman Dönüşümleri:** Görselleştirilmesi istenilen manevrasız veya manevralı yörüngeler VTS uygulamasına OEM dosya formatında verilir. Bu OEM dosya formatı yörünge verilerini (pozisyon ve hız) zaman etiketli olarak bulundurmaktadır.

Pozisyon bilgisi km cinsinden J2000 referans koordinat sistemine göre verilmelidir. Her bir pozisyon bilgisi için satırın başında zaman bilgisi de yer almalıdır. Bu zaman bilgisi JD1950 ve MDJ formatında olabilir. JD1950 referans olarak 1 Ocak 1950 00:00:00.000 tarihini alırken MJD referans olarak 17 Kasım 1858 00:00:00.000 tarihini alır. Bu çalışmada, OEM dosyası oluşturulurken zaman bilgileri MJD olarak ifade edilmiştir.

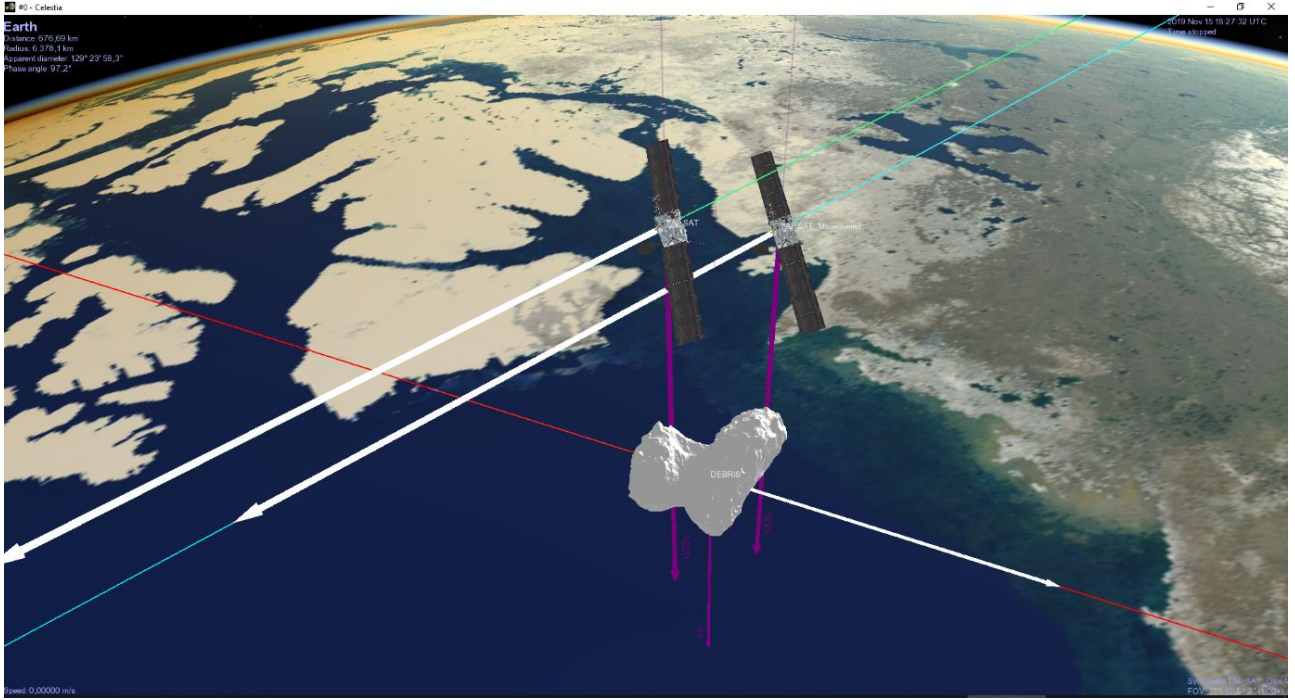
MJD hesaplaması yapılırken pozisyon verisinin üretildiği tarihin JD değeri hesaplanır. Bu değer M.Ö 1 Ocak 4713 12:00:00.000 tarihinden itibaren geçen gün sayısıdır. Pozisyon verilerinin üretildiği MJD değerinin hesaplanması için pozisyon verilerinin üretildiği tarihin JD değerinden 17 Kasım 1858 gece yarısının JD değeri çıkarılır. Sonuç olarak MJD değeri pozisyon verilerinin üretildiği tarihin 17 Kasım 1858 00:00:00.000 tarihinden itibaren geçen gün sayısıdır. Bu sayının tam ve kesir kısmı bulunur. Tam kısım, gün sayısını, kesirli kısım hesaplama yapılan günün saatini ifade eder.

OEM dosyası pozisyon ve zaman bilgisini içerdiği satırlarda zaman bilgisini iki farklı sütunda barındırır. İlk sütunda MJD değerindeki tam kısmı gün olarak ifade ederken ikinci satır MJD değerindeki kesirli kısmın saniyeye dönüştürülmüş halidir.

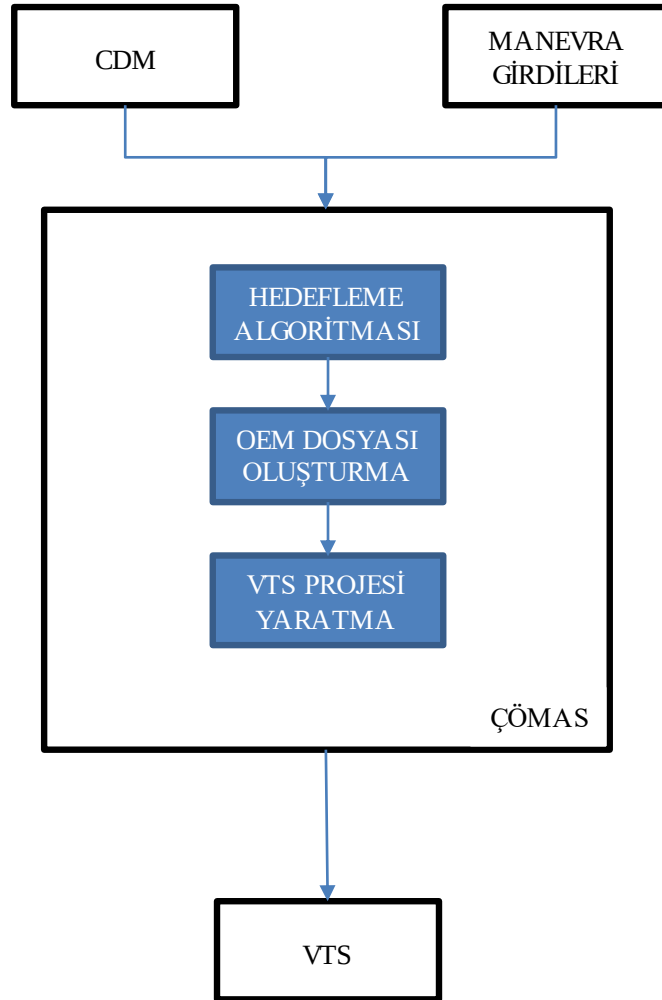
**Program Yapısı:** ÇÖMAS programı temel olarak OREKIT kütüphanesine dayanmakta olup, bu uygulama JAVA ortamında geliştirilmiştir. Oluşturulan bir arayüz sayesinde kullanıcı girdileri alınmakta ve elde edilen çıktılar kullanıcı ile buluşturulmaktadır. Programın amacı çarpışma manevrasının türüne karar verme işlemini hızlandırmak ve ilgili manevrayı hesaplayarak görselleştirmektir. Bu bağlamda sıfırdan bir OpenGL programı geliştirilmesi yerine, hâlihazırda var olan VTS programı ile geliştirilen arayüz konuşturularak, yörüngesel hesaplama işlemlerini JAVA ortamında yapılmasını sağlayıp, elde edilen çıktıların VTS programı tarafından anlaşılacak hale getirilmesi ile görselleştirme işleminin yapılması sağlanmıştır.

Şekil 2 ÇÖMAS Arayüzü

Şekil 2’de oluşturulan arayüz verilmiştir. Bu arayüz aracılığı ile kullanıcı kendisine gelen CDM’i uygulamaya girdi olarak verebilmektedir. Seçilen bu CDM’e ait bilgiler ara yüzün sağ tarafında “CDM Özeti” alanında görüntülenir. Eğer kullanıcı manevra yapmak istiyorsa, operatörün iki uydunun yörüngelerinin birbirlerine göre konumlarından, birinci uyduya uygulayacağı manevra tipine karar vermesi ve “Manevra Hesapla” kutucuğunu işaretlemesi beklenir. Bu durumda kullanıcıdan “Manevra Tipini”, “Uydu Ağırlığını”, “Yanma Süresi İlk Tahminini”, “Manevra Zamanını”, “İtki Kuvvetini”, “Spesifik İtki Katsayısını” ve “Hedef İska Mesafesi” değerlerini girmesi beklenir. ÇÖMAS manevra hesabını hedef algoritması ile hesaplar ve operatörün girdiği ısk mesafesini eşik değerini sağlayacak şekilde ( $1e-6$ ) yakınsamaya çalışır. Bu mesafeyi sağlayan itki süresini hesapladıktan sonra manevrayı birinci uyduya uygular. Eğer kullanıcı direkt olarak TCA anını görmek istiyorsa “En Yakın Geçiş Zamanına İlerlet” butonu sayesinde, çarpışmaya en yakın ana ilerleyebilmektedir. Eğer bir CDM seçildi ise “Simülasyonu Başlat” butonuna basılarak simülasyon başlatılabilir ve bir VTS projesi oluşturulur. Bu proje, kullanılması istenilen OEM dosyaları ile üç boyutlu uydu modellerinin, “.xml” uzantılı proje dosyanın içindeki ilgili başlıkların altındaki bilgilerin düzenlenmesi ile yaratılır. Açık olan simülasyon “Çıkış” butonu ile kapatılabilir ve yeni bir simülasyon “Simülasyonu Başlat” butonuyla tekrar başlatılabilir. Hedefleme algoritmasının yakınsamadığı durumlarda kullanıcıya yanma süresinin bulunamadığını belirten bir hata mesajı verilir. Şekil 3’te başlatılan bir simülasyon örneği, Şekil 4’te ise program yapısı verilmiştir.



Şekil 3 Aktif Simülasyon Görseli. VTS Programında Oluşturulmuş Bir Senaryo



Şekil 4 ÇÖMAS Program Yapısı.

### UYGULAMALAR ve DEĞERLENDİRME

Bu bölümde ÇÖMAS uygulamasının oluşturulan iki farklı CDM ile test edildiği senaryolar anlatılmış ve test edilen senaryoların girdileri ile elde edilen manevra çıktıları tablo halinde ifade edilmiştir. İki senaryoda da öncelikli olarak CDM dosyası uygulama aracılığı ile okutulmuş ve görselleştirme aracılığı ile manevra tipine karar verilmiştir. Karar verilen manevra tipinin uygulanması için arayüz vasıtası ile manevra girdileri de aşağıdaki tablolarda verildiği gibi ÇÖMAS programına iletilmiştir.

Birinci oluşturulan CDM'e göre iki uydu arası mesafe seçilen TCA'da 481 metre ıska mesafesi olacak şekilde ayarlanmıştır ve yörünge bilgileri Tablo 4'te verilmiştir. Bu senaryoya göre TUSAŞSat uydusunun ikincil uyduya göre daha aşağıda olmasından kaynaklı düzlem içi alçaltma manevrası uygulanması gerektiğine karar verilmiştir ve manevra girdileri; itki kuvveti 4N, itki katsayısı 220s, hedef ıska mesafesi 4000m, uydu ağırlığı 900kg, manevra zamanı TCA'dan 7 yörünge önce ve yanma süresi ilk tahmini 22s olarak Tablo 5'da gösterildiği gibi girilmiştir. Hedefleme algoritması istenilen manevrayı 45 iterasyonda tamamlamıştır. Elde edilen deltaV değeri 0.031372 m/s olup itki süresi 7.058807s olarak hesaplanmıştır. Hedeflenen ıska mesafesi 1e-6 tutarlılıkta 4000.0000002 metre olarak elde edilmiştir.

İkinci oluşturulan CDM'de ise ıska mesafesi 570 metre olarak ayarlanmıştır. Yörünge bilgileri Tablo 7'de verilmiştir. Bu senaryoya göre TUSAŞSat uydusunun ikincil uyduya göre daha yukarıda olmasından dolayı düzlem içi yükseltme manevrası uygulanması gerekmektedir ve manevra girdileri; itki kuvveti 2N, itki katsayısı 190s, hedef ıska mesafesi 11300m, uydu ağırlığı 1000kg, manevra zamanı TCA'dan 7 yörünge önce ve yanma süresi ilk tahmini 30s olarak Tablo 8'da gösterildiği gibi girilmiştir. Bu manevrada, hedefleme algoritması istenilen manevrayı 9 iterasyonda tamamlamıştır. DeltaV değeri 0.092749 m/s, itki süresi 46.373439s olarak hesaplanmıştır. Hedeflenen ıska mesafesi 1e-6 tutarlılıkta 11299.999999 metre olarak elde edilmiştir.

Tablo 4 Birinci Test Senaryosu İçin Oluşturulan CDM Parametreleri

|                          | Pozisyon (km)           |            |             | Hız (km/s) |          |          |
|--------------------------|-------------------------|------------|-------------|------------|----------|----------|
|                          | X                       | Y          | Z           | X          | Y        | Z        |
| <b>TUSAŞSat</b>          | 2190.41261              | 4735.23004 | -4770.35030 | 3.73318    | 3.74662  | 5.43801  |
| <b>Debris</b>            | 2189.99138              | 4735.46105 | -4770.38051 | -0.37908   | -5.34383 | -5.39398 |
| <b>TCA</b>               | 2018-03-14T07:03:45.215 |            |             |            |          |          |
| <b>İska Mesafesi (m)</b> | 481.0                   |            |             |            |          |          |

Tablo 5 Birinci Test Senaryosu İçin Karar Verilen Manevra Tipi ve Girdileri

|                     | İtki Kuvveti (N)    | İtki Katsayısı (s) | Hedef İska Mesafesi (m) | Uydu Ağırlığı (kg) | Manevra Zamanı (n) | İlk Tahmin (s) |
|---------------------|---------------------|--------------------|-------------------------|--------------------|--------------------|----------------|
| <b>Manevra</b>      | 4                   | 220                | 4000                    | 900                | 7                  | 22             |
| <b>Manevra Tipi</b> | Düzlem İçi Alçaltma |                    |                         |                    |                    |                |



Tablo 6 Birinci Test Senaryosu Hedefleme Algoritması Sonuçları

|                                        | İterasyon Sayısı | DeltaV (m/s) | İtki Süresi (s) | Iska Mesafesi (m) |
|----------------------------------------|------------------|--------------|-----------------|-------------------|
| <b>Hedefleme Algoritması Sonuçları</b> | 45               | 0.031372     | 7.058807        | 4000.0000002      |

Tablo 7 İkinci Test Senaryosu İçin Oluşturulan CDM Parametreleri

|                          | Pozisyon (km)           |            |              | Hız (km/s) |          |           |
|--------------------------|-------------------------|------------|--------------|------------|----------|-----------|
|                          | X                       | Y          | Z            | X          | Y        | Z         |
| <b>TUSAŞSat</b>          | -2883.554638            | 729.588484 | -6414.410974 | 6.882501   | 1.084982 | -2.971866 |
| <b>Debris</b>            | -2883.585163            | 729.061786 | -6414.195099 | -5.523947  | 4.242237 | 2.837630  |
| <b>TCA</b>               | 2018-05-27T11:16:25.971 |            |              |            |          |           |
| <b>Iska Mesafesi (m)</b> | 570.0                   |            |              |            |          |           |

Tablo 8 İkinci Test Senaryosu İçin Karar Verilen Manevra Tipi ve Girdileri

|                     | İtki Kuvveti (N)     | İtki Katsayısı (s) | Hedef Iska Mesafesi (m) | Uydu Ağırlığı (kg) | Manevra Zamanı (n) | İlk Tahmin (s) |
|---------------------|----------------------|--------------------|-------------------------|--------------------|--------------------|----------------|
| <b>Manevra</b>      | 2                    | 190                | 10000                   | 11300              | 7                  | 30             |
| <b>Manevra Tipi</b> | Düzlem İçi Yükseltme |                    |                         |                    |                    |                |

Tablo 9 İkinci Test Senaryosu Hedefleme Algoritması Sonuçları

|                                        | İterasyon Sayısı | DeltaV (m/s) | İtki Süresi (s) | Iska Mesafesi (m) |
|----------------------------------------|------------------|--------------|-----------------|-------------------|
| <b>Hedefleme Algoritması Sonuçları</b> | 9                | 0.092749     | 46.373439       | 11299.999999      |

## SONUÇ

Son yıllarda uydu projelerinin artması ile Dünya etrafında aktif ve aktif olmayan yapay uydu sayıları sürekli artmaktadır. Bu sebeple, Dünya yörüngesinde bulunan uydular gün geçtikçe daha fazla çarpışma uyarısı almaktadır. Çarpışma ihtimalinin olduğu durumlarında CSpOc ve ESA gibi otoriteler tarafından uydu operatörlerine çarpışma ikaz mesajları gönderilir. Bu mesaj içinde bulunan çarpışma olasılığına göre uydu operatörü manevra yapıp yapmayacağına karar verir. CDM'ler 2-3 gün önce uydu operatörüne iletebileceği gibi bu mesaj en yüksek çarpışma olasılığının hesaplandığı zamandan 10-12 saat önce de uydu operatörüne iletebilir.

Sonuç olarak yazılan JAVA uygulaması ile kullanıcılar TCA zamanındaki uydu konumlarını analiz edebilmekte ve bu analiz sonucunda yapılması gereken manevraya karar verebilmektedirler.

Hedefleme algoritması ile hesaplanan bu manevralar VTS programına gönderilerek manevralı görselleştirme işlemi gerçekleştirilir. Yapılmak istenen manevralar, temel olarak ısk mesafesini eşik bir değerde tutmayı hedeflemektedir. Kullanıcıdan girdi olarak alınan hedef ısk mesafesi metre cinsindendir. Hedefleme algoritmasının tutarlılığı  $1e-6$  olmakla birlikte, hedeflenen ısk mesafesini mikrometre mertebesinde tutturmayı sağlamaktadır. Uygulama bölümünde verilen sonuçlardaki iterasyon sayılarının azaltılması, ki bu doğrudan hesaplama süresini değiştirmektedir, hedefleme algoritmasının tutarlılığını  $1e-3$  mertebesine düşürerek yapılabilir. Bu mertebeye, milimetre cinsinden ısk mesafesini belirlemektedir. İleriki çalışmalarda bunun kararı kullanıcıya arayüz vasıtası ile bırakılacaktır.

### Kaynaklar

- Channumsin, S., Sreesawer, S., Saroj, T., Saingyen, P., ve Diğerleri, 2019. *Collision Avoidance Strategies and Conjunction Risk Assessment Analysis Tool At GISTDA*, First International Orbital Debris Conference, Texas, 9-12 Aralık.
- Chapra, S. C., Canale, R. P., 2015. *Numerical Methodos for Engineers*, Mc Graw Hill Education, 7. Baskı, s.151-157.
- Chapra, S. C., Canale, R. P., 2015. *Numerical Methodos for Engineers*, Mc Graw Hill Education, 7. Baskı, s.782-789.
- Fernandez-Martin, C., Maisonobe, L., 2010. Sharing the Knowledge: An Open-Source Vision for Flight Dynamics, SpaceOps, Alabama, 25-30 Nisan.
- Maisonobe, L., Pommier, V., Parraud, P., 2010. *OREKIT: An Open Source Library for Operational Flight Dynamics Applications*, ICATT, Madrid, 29-30 Nisan.

### EK-1: KISALTMALAR

| Kısaltmalar | İngilizce Açılımı                                       | Türkçe Karşılığı                |
|-------------|---------------------------------------------------------|---------------------------------|
| CCSDS       | Consultative Committee for Space Data System            | -                               |
| CDM         | Conjunction Data Message                                | Çarpışma İkaz Mesajı            |
| CSpOC       | Combined Space Operation Center                         | -                               |
| ÇÖMAS       | -                                                       | Çarpışma Önleme Manevra Sistemi |
| ECI         | Earth Centered Inertial                                 | -                               |
| ESA         | European Space Agency                                   | Avrupa Uzay Ajansı              |
| GISTDA      | Geo-Informatics and Space Technology Development Agency | -                               |
| GMAT        | General Mission Analysis Tool                           | -                               |

|        |                                                   |                       |
|--------|---------------------------------------------------|-----------------------|
| ITRF   | International Terrestrial Reference Frame         | -                     |
| JD1950 | Julian Date 1950                                  | -                     |
| MDJ    | Modified Julian Date                              | -                     |
| VTs    | Visualization Tool for Space                      | -                     |
| OEM    | Orbit Ephemeris Message                           | -                     |
| TCA    | Time of Closest Approach                          | En Yakın Geçiş Zamanı |
| TCP-IP | Transmission Control Protocol / Internet Protocol | -                     |
| THEOS  | Thailand Earth Observation Satellite              | -                     |