

KAYZER ROKET TAKIMI  
TEKNOFEST-2022 ROKET YARIŞMASI  
ÖTR AŞAMASI  
UÇUŞ BENZETİMİ RAPORU



## İçindekiler Tablosu

Sorular ve Cevaplar .....	3
Kinematik Denklemler .....	4
Benzetim Yapısı .....	5
Benzetimin Doğrulanması .....	15
Referanslar .....	15

## Sorular ve Cevaplar

**Soru 1:** Kinematik ve dinamik hareket denklemleri nedir, aralarındaki farklar nelerdir?

Kinematik konum, hız, ivme gibi hareketlerin incelenmesidir. Dinamik ise kuvvetler ve malzeme özellikleri arasındaki etkileşimde zamanla değişen olguların tam olarak dikkate alınmasıdır.

**Soru 2:** İki serbestlik dereceli kinematik benzetimin, roket dinamik denklemlerinin (motor itki kuvveti ve aerodinamik sürüklenme kuvveti) de katılarak roket uçuşuna uyarlanması ile elde edilecek uçuş benzetimi, roket tasarımında ne amaçlarla kullanılabilir, faydaları nelerdir?

Simülasyon programları zaman ve maliyet konularında tasarruf sağlayan teknolojilerdir. Roketin uçuş benzetimi ile uçuş sırasında oluşabilecek sorunlar ve istenilen verilerin karşılanıp karşılanmadığı önceden fark edilebilir. Böylece üretim aşamasına gelmeden tasarıma müdahale edilebilir.

**Soru 3:** İki serbestlik dereceli dinamik uçuş benzetimine Y eksenı etrafında açısal hareket eklenerek elde edilecek 3 serbestlik dereceli benzetimin getireceğı faydalar nelerdir? Bu benzetimin kullanılması için roketin ek olarak hangi bilgilerinin bilinmesi ve kullanılması gerekir?

Roket X, Y, Z eksenlerinde yaptıkları hareketlerin yanında yuvarlanma, yunuslama, dönme hareketleri yaparlar. Bu hareketlerden ikisinin aynı anda incelendiğı benzetim modellerine 2-Serbestlik Dereceli Uçuş benzetimi denirken üçünün aynı anda incelendiğı modellerlere 3-Serbestlik Dereceli Uçuş benzetimi denir. 3-Serbestlik Dereceli Uçuş Benzetimi ile yapılan bir modelleme daha detaylı olacağından gerçeğe daha yakın olur. Bu benzetim için roketin Y ekseninde yaptığı hareketin incelenmesi gerekir.

## Kinematik Denklemler

### Hız denklemleri

$$zaman_{uçuş} = \frac{2 \times v_0 \times \sin(a)}{g}$$

$$a = 70^\circ$$

$$g = 9.801 \text{ m/s}^2$$

$$v_0 = 100 \text{ m/s}$$

$$hız_{yatay} = v_0 \times \cos(a) - 0 \times zaman_{uçuş}$$

$$hız_{dikey} = v_0 \times \sin(a) - g \times zaman_{uçuş}$$

$$hız_{bileşke} = \sqrt{hız_{yatay}^2 + hız_{dikey}^2}$$

### Konum denklemleri

$$menil = hız_{yatay} \times zaman_{uçuş}$$

$$yükseklik = v_0 \times \sin(a) \times zaman_{uçuş} - \frac{1}{2} \times g \times zaman_{uçuş}^2$$

$$irtifa_{max} = \frac{(v_0 \times \sin(a))^2}{2 \times g}$$

### Uçuş yolu açısı hesabı denklemi

$$açı_{uçuş} = \tan^{-1} \left( \frac{hız_{dikey}}{hız_{yatay}} \right)$$

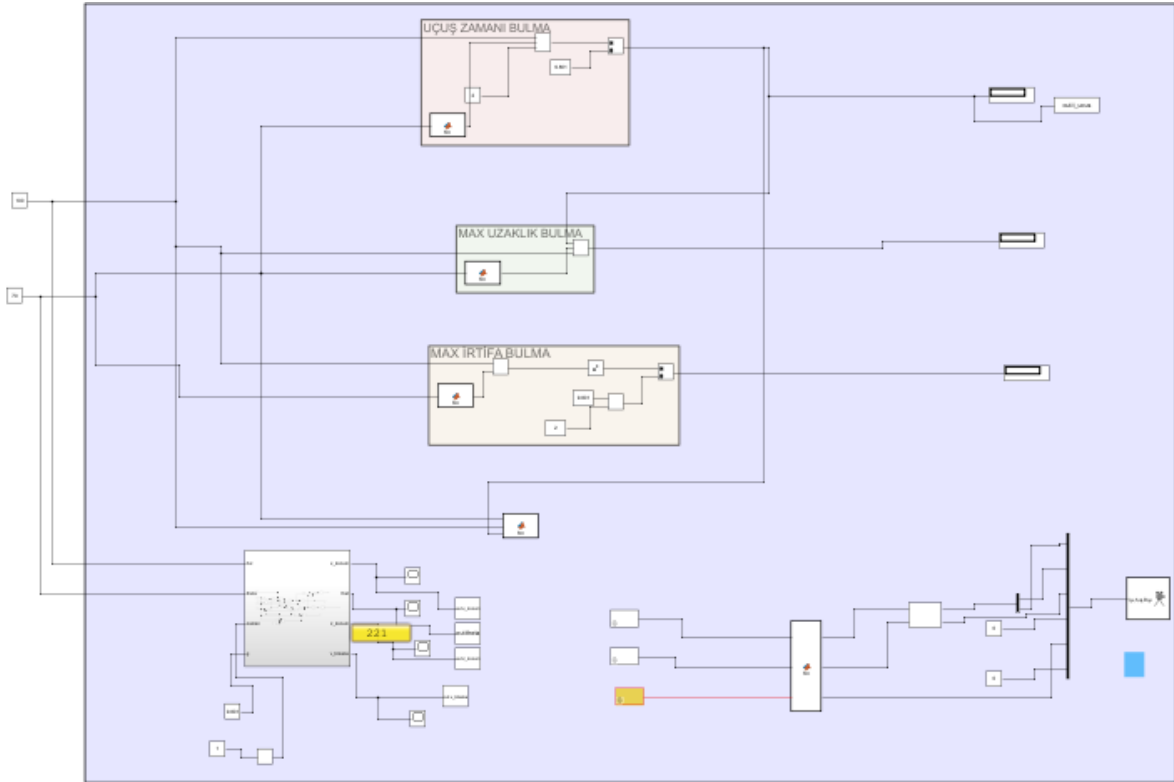
## Benzetim Yapısı

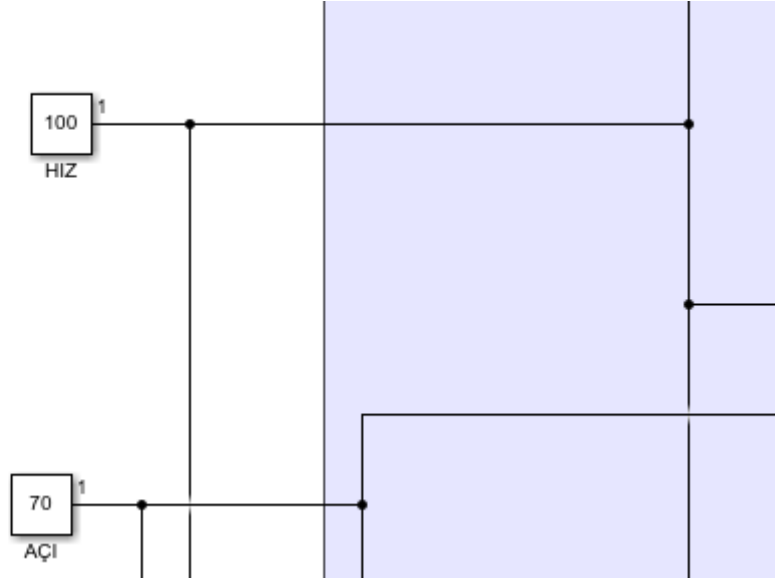
Benzetimin yapısı ve çözüm yöntemi anlatılmalı, kullanılan araçlar/yazılımlar belirtilmelidir. Benzetim kod ile çözülüyorsa kodlar, Simulink ile çözülüyor ise Simulink modelinin ekran görüntüsü ve varsa ek MATLAB kodları paylaşılmalıdır.

Uçuş benzetim 3 farklı türden uygulamayla yapılmıştır. Tüm bu uygulamalar için MATLAB kullanılmıştır. Bu üç uygulama olan MATLAB script, MATLAB Simulink ve Matlab GUI üzerinden uçuş benzetim tamamlanmıştır.

### 1. MATLAB Simulink

Uçuş benzetimi öncelikle Simulink üzerinden yapılmıştır. Simulink ile yapılan sistemin şeması aşağıdadır.



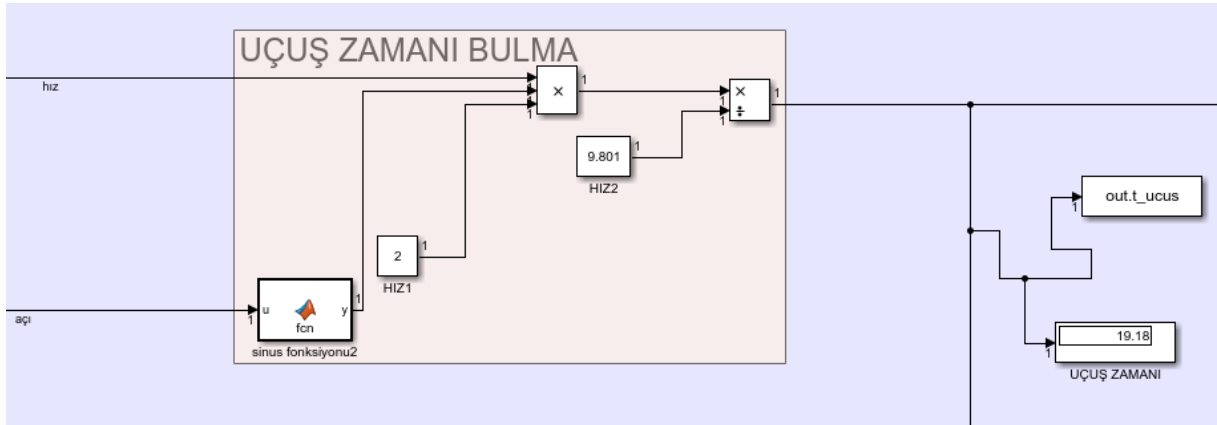


Görüldüğü üzere sisteme girdi olarak ilk hız ve ilk açı değerleri verilmiştir.

Sistemdeki alt sistemler sayesinde istenilen veriler ve grafikler ve simülasyonlar elde edilmiştir.

İlk olarak uçuş zamanını bulan alt sistem tasarlanmıştır. Bunun için aşağıdaki formül kullanılmıştır.

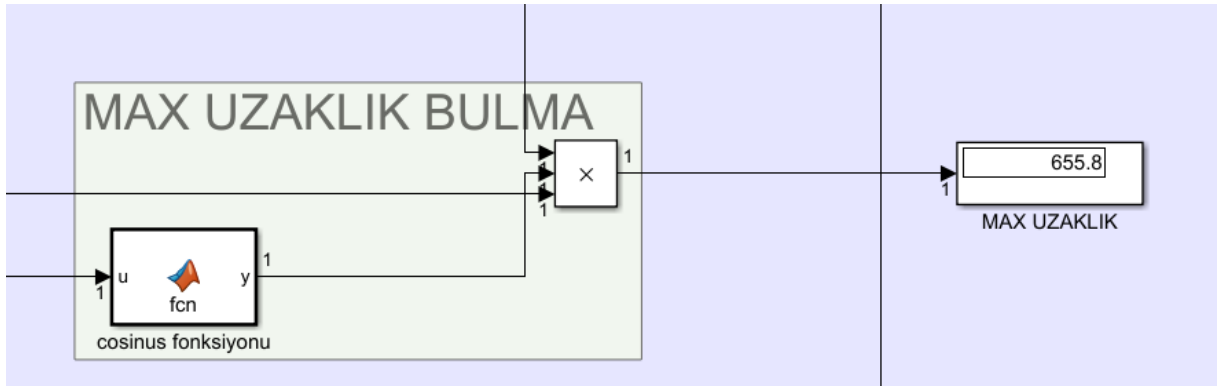
$$zaman_{uçuş} = \frac{2 \times v_0 \times \sin(a)}{g}$$



Görüldüğü üzere alt sistem uçuş zamanını 19.18 saniye bulmuştur ve bulunan bu sonucu MATLAB workspace'e yazdırmıştır.

İkinci alt sistem maximum yatay uzaklığı bulmak için tasarlanmıştır. Bunun için aşağıdaki formül kullanılmıştır.

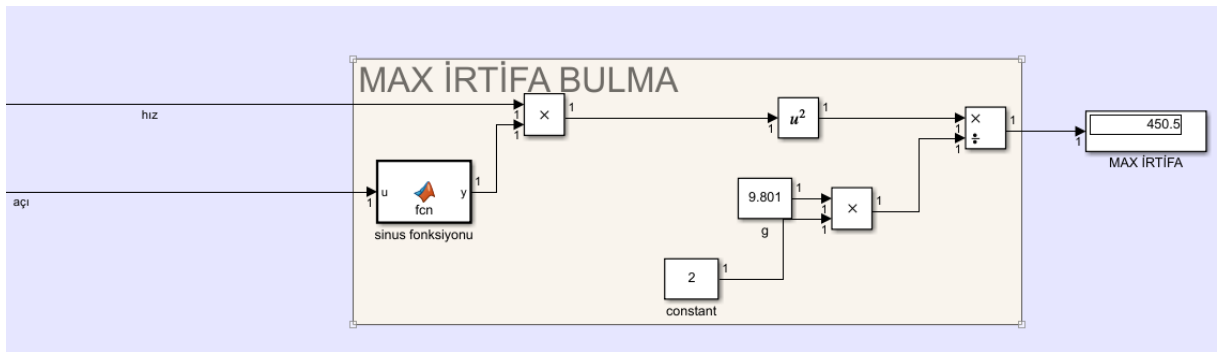
$$uzaklık_{max} = hız_{yatay} \times zaman_{uçuş}$$



Görüldüğü üzere maksimum yatay uzaklık 655,8 metre olarak bulunmuştur.

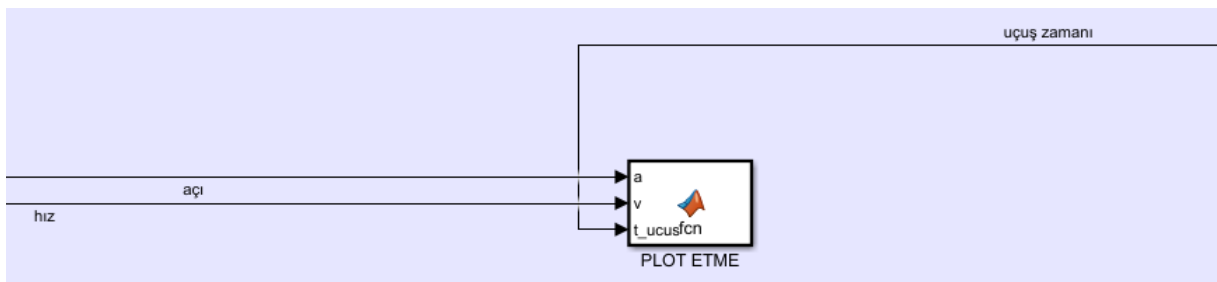
Ardından, maksimum irtifa tespit etmek için bir alt sitem tasarlanmıştır. Kullanılan formül aşağıdadır.

$$irtifa_{max} = \frac{(v_0 \times \sin(a))^2}{2 \times g}$$



Sonuç olarak maksimum irtifa 450,5 metre olarak bulunmuştur.

Ardından yükseklik menzil grafiğini elde etmek için simulink-matlab fonksiyonu ile çalışan bir alt sistem tasarlanmıştır.



Fonksiyon için kullanılan kodlar aşağıda verilmiştir.

```
function fcn(a,v,t_ucus)

for t=0:0.1:t_ucus

time=0:0.01:t;

g=9.801; %yerçekimi ivmesi

vx=v*cosd(a);

x_konum=vx.*time;

z_konum=(v*sind(a).*time) - (1/2).*g.*(time.^2);

[maxx,idx] = max(x_konum);

[maxz,idz] = max(z_konum);

plot(x_konum,z_konum,x_konum(idx),z_konum(idx),'pr');

trxt=('Max Uzaklık');

txt=('Maximum İrtifa \uparrow');

text(x_konum(idx),z_konum(idx),trxt,'VerticalAlignment','bottom');

text(x_konum(idz),maxz,txt,'HorizontalAlignment','right');

if t==t_ucus

    break

end

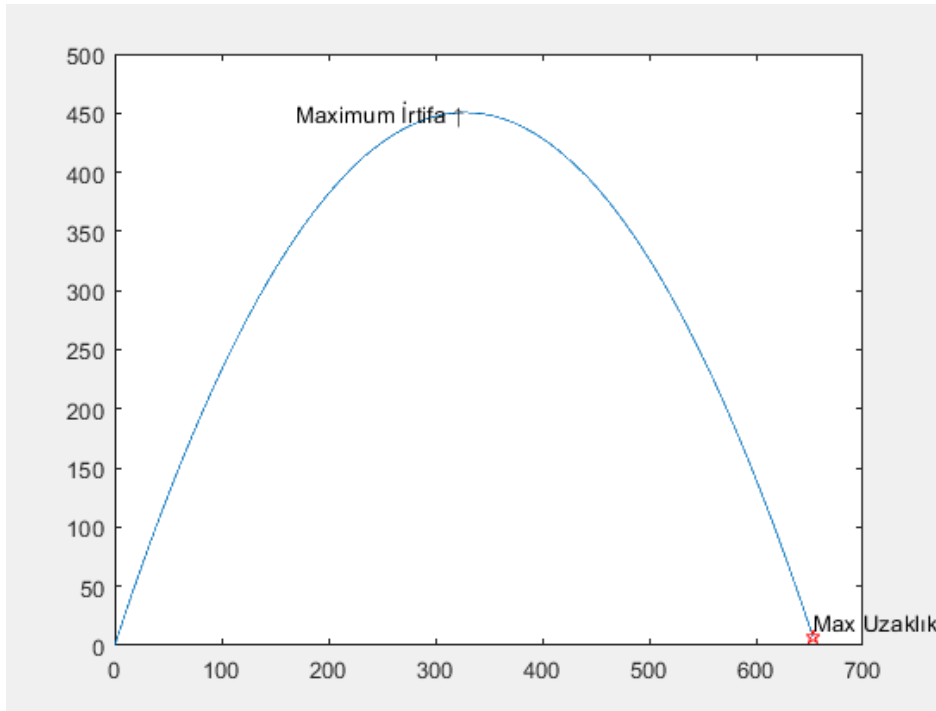
drawnow;

end

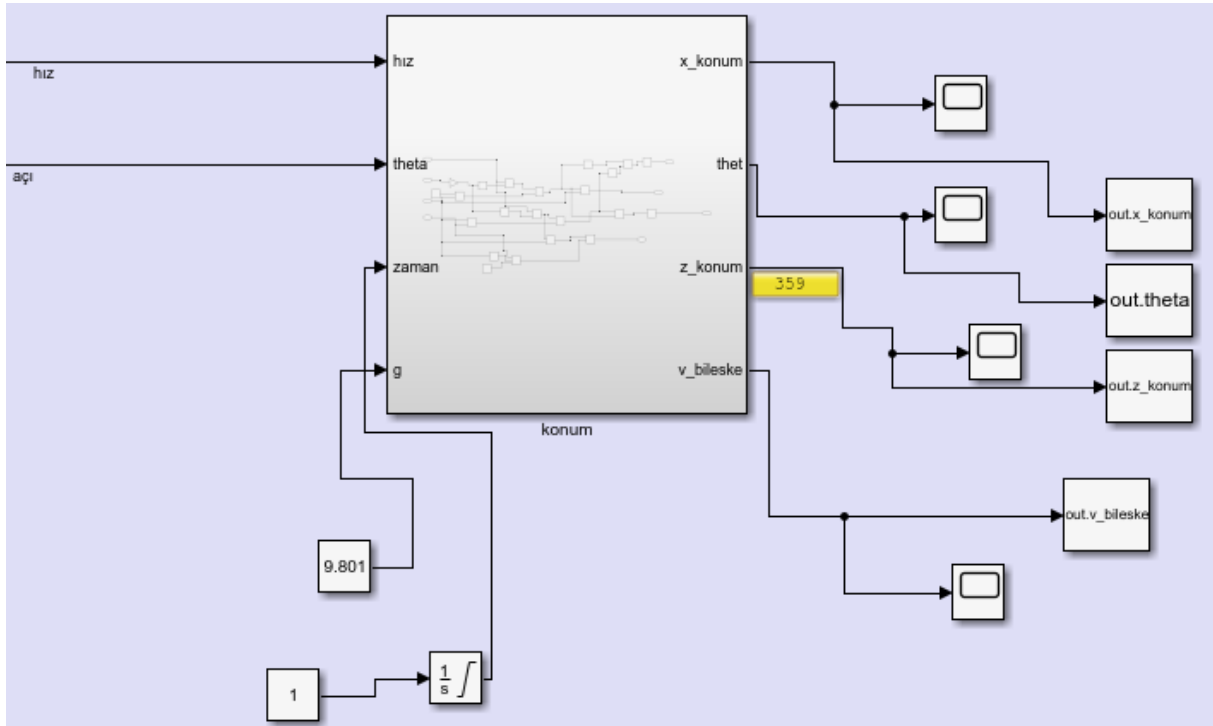
end
```

Sistem çıkış olarak istenilen grafiği gerçek zamanlı olarak değişen şekilde vermektedir.



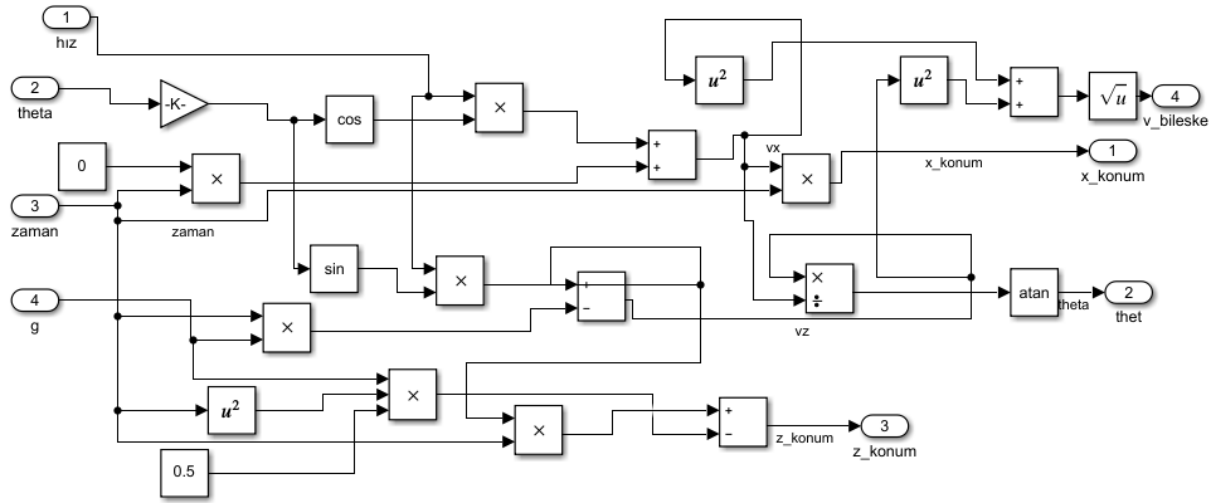


Ardından diğer bir alt sistem sayesinde roketin x ve z eksenindeki konumu, açısı ve hız değerlerini 0,01 saniye zaman adımı ile vektör şeklinde kaydedip grafiklerini scope ile gösteriyor.



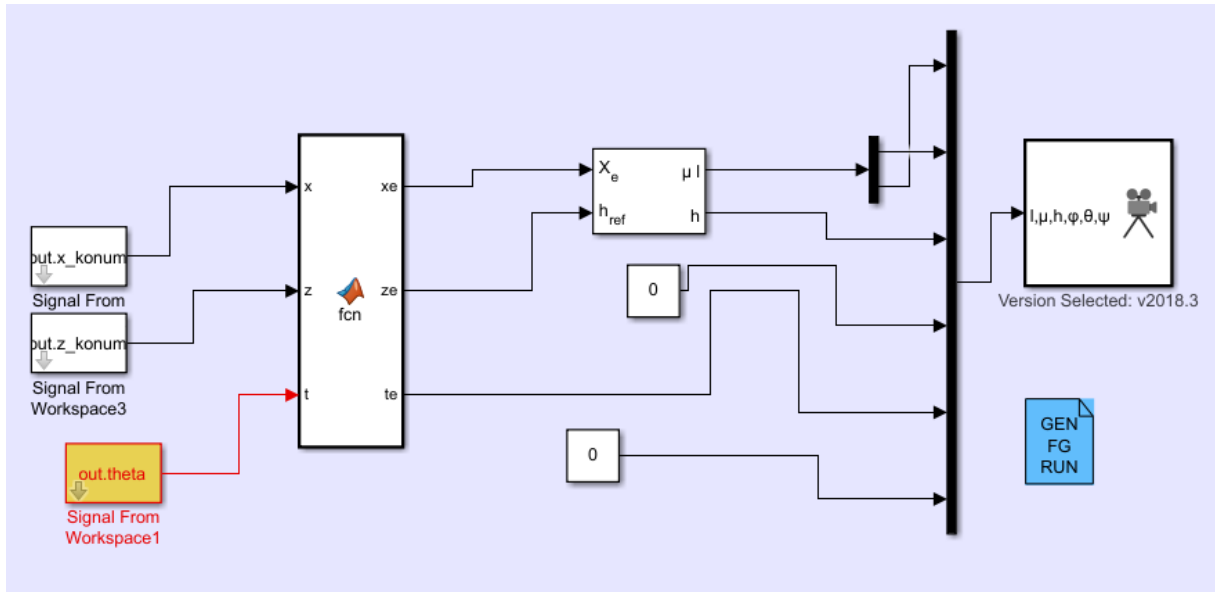
Burada zaman adımı integrator fonksiyonu sayesinde sağlanmıştır.

Sistemin iç kısmı aşağıda gösterilmiştir.

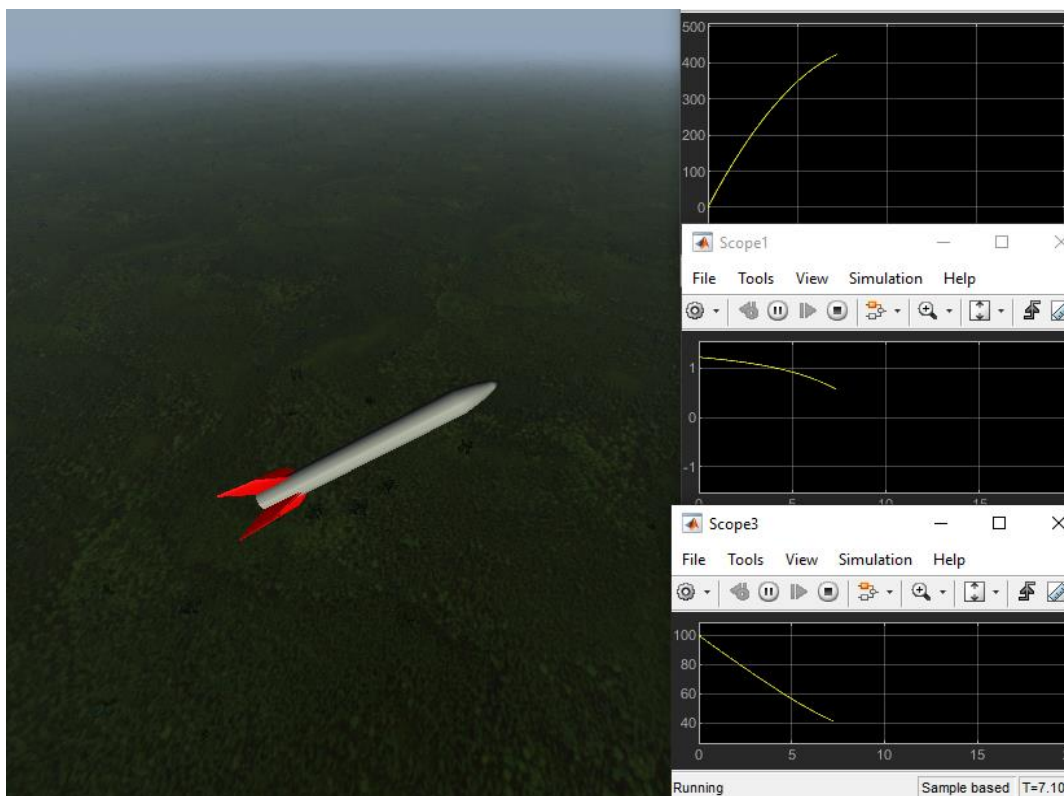


Buradaki işlemler, denklemler kısmında verilen denklemler kullanılarak tasarlanmıştır.

Ardından roketin gerçeğe çok benzer simülasyonu için elde edilen ve yukarıdaki sistem sayesinde kaydedilen verileri kullanan bir sistem geliştirildi.

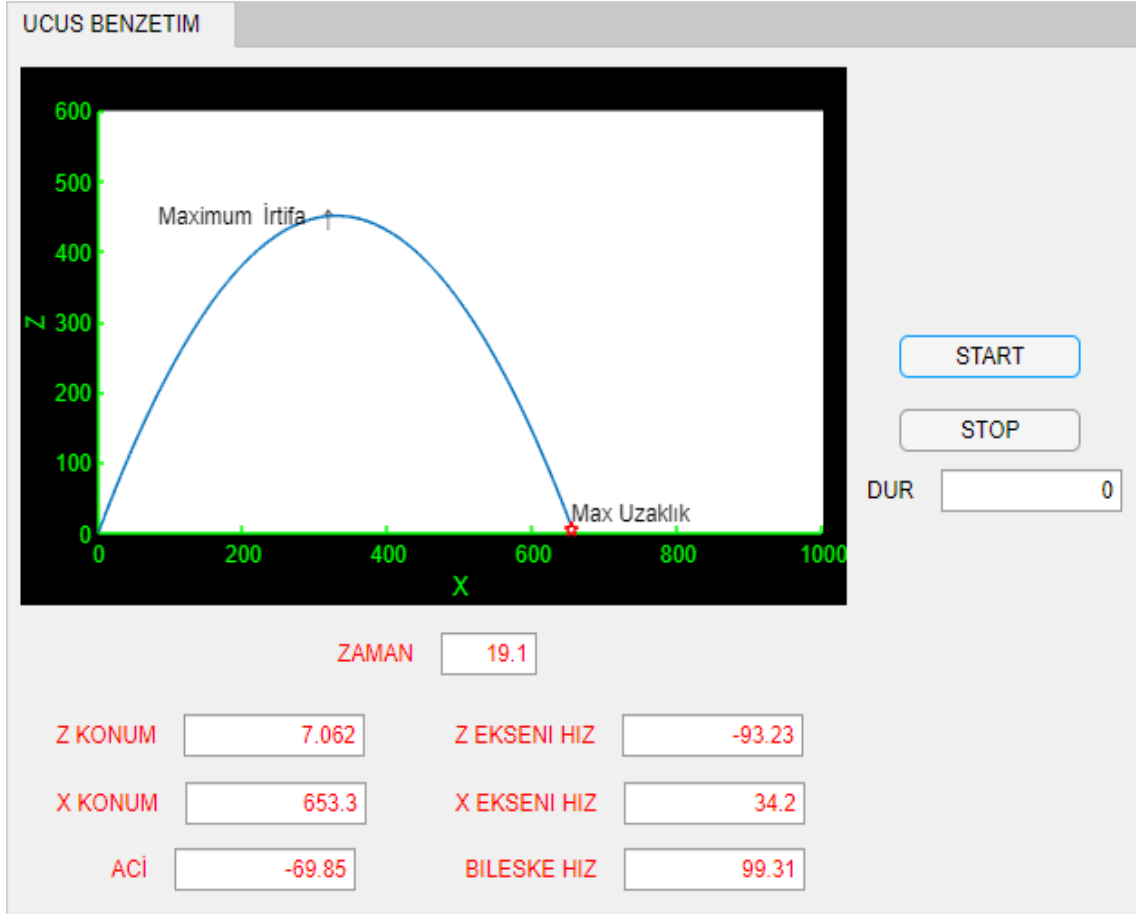


Gerçeğe yakın roket uçuş benzetim simülasyonu için FlightGear programı kullanıldı. Simülasyonun zaman adımı 0.01 saniye olarak ayarlandı. Simülasyondan elde edilen ekran görüntüleri aşağıdadır. Ayrıca simülasyonun [videosu](#) link olarak raporun en alt kısmına eklenmiştir.



## 2. MATLAB GUI

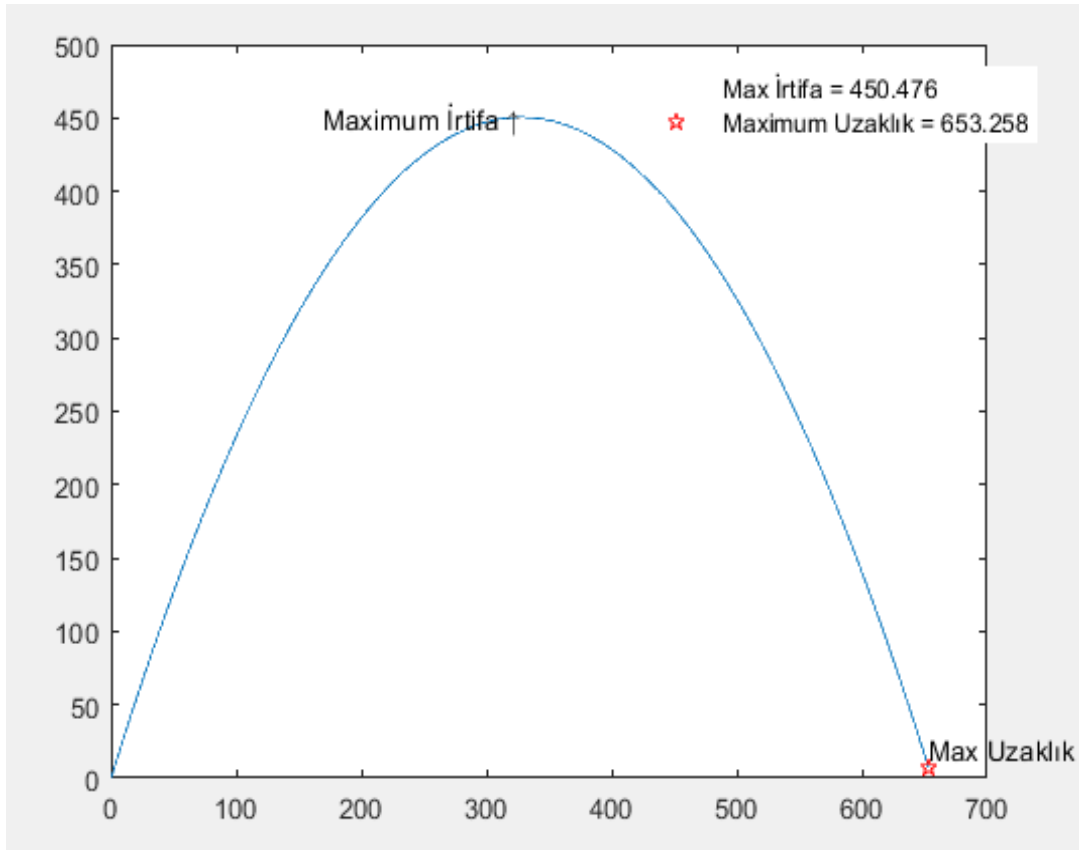
Uçuş benzetim için MATLAB GUI kullanılarak kullanıcı ara yüzü bir uygulama tasarlanmıştır. Bu uygulama sayesinde gerçek zamanlı olarak konum değerleri, hız değerleri ve açı değerleri gözlemlenebilir.



Uygulamanın çalışırken çekilen [videosu](#) raporun sonunda paylaşılan linkte mevcuttur.

## 3. MATLAB SCRIPT

Matlab Simulink ve Matlab GUI'nin dışında, Matlab script'de yazılan kodlar sayesinde istenilen tüm değerleri ve gerçek zamanlı olarak uçuş benzetim simülasyonu elde edilmiştir. Simülasyonun [video](#) bağlantısı rapor sonunda mevcuttur.



Kullanılan kodlar aşağıdadır.

```
clc;
clear;
clear all;
g=9.801; %yerçekimi ivmesi
v=100;
a=70;% uçuş açısı;
t_ucus=(2*v*sind(a))/g;
time=0:0.01:t_ucus;
vx=v*cosd(a);
vz=v*sind(a)-g*time;
```

```
max_irtifa=(v*sind(a)^2)/(2*g);
max_uzaklik=v*cosd(a).*t_ucus;
x_konum=vx.*time;
z_konum=(v*sind(a).*time) -
(1/2).*g.*(time.^2);
[maxxx,idx] = max(x_konum);
[maxz,idz] = max(z_konum);
```

```

plot(x_konum,z_konum,x_konum(idz),z_konum(idz),'pr');

legend(sprintf('Max İrtifa = %0.3f',maxz),sprintf('Maximum Uzaklık = %0.3f',maxx));

axis([xlim min(x_konum) max(x_konum)+1]);

trxt=('Max Uzaklık');

txt=('Maximum İrtifa \uparrow');

text(x_konum(length(time)),z_konum(length(time)),trxt,'VerticalAlignment','bottom');

text(x_konum(idz),maxz,txt,'HorizontalAlignment','right');

fprintf('Tepe noktası yüksekliği= %.2f\n',maxz);

fprintf('Maksimum yatay menzil= %.2f\n',maxx);

```

```

max_irtifa_vx=v*cosd(a);

max_irtifa_vz=v*sind(a)-
g*time(idz);

max_irtifa_bileske=sqrt(max_irtifa_vx^2+max_irtifa_vz^2);

fprintf('Tepe noktası bileşke hız= %.2f\n',max_irtifa_bileske);

fprintf('Tepe noktası zamanı= %.2f\n',time(idz));

```

```

konum=[x_konum(end),0,z_konum(end)];

fprintf('Son Pozisyon= %.2f\n',konum);

son_hiz_vx=v*cosd(a);

son_hiz_vz=v*sind(a)-g*time(idz);

son_hiz_bileske=sqrt(son_hiz_vx^2+son_hiz_vz^2);

fprintf('Son Hız Bileşke= %.2f\n',son_hiz_bileske);

son_ucus_aci=atand(son_hiz_vz/son_hiz_vx);

fprintf('Son Uçuş Açısı= %.2f\n',son_ucus_aci);

```

```

clc;

clear;

clear all;

g=9.801; %yerçekimi ivmesi

v=100;

a=70;% uçuşu yola açısı;

t_ucus=(2*v*sind(a))/g;

t_cikis=t_ucus/2;

```

```

for t=0:0.1:t_ucus

time=0:0.01:t;

vx=v*cosd(a);

x_konum=vx.*time;

z_konum=(v*sind(a).*time) -
(1/2).*g.*(time.^2);

[maxx,idx] = max(x_konum);

[maxz,idz] = max(z_konum);

```

```

plot(x_konum,z_konum,x_konum(idz),z_konum(idz),'pr');

legend(sprintf('Max İrtifa = %0.3f',maxz),sprintf('Maximum Uzaklık = %0.3f',maxx));

trxt=('Max Uzaklık');

txt=('Maximum İrtifa \uparrow');

text(x_konum(idz),z_konum(idz),trxt,'VerticalAlignment','bottom');

text(x_konum(idz),maxz,txt,'HorizontalAlignment','right');

drawnow;

```

Çıktılar Matlab Command Window’da yazdırılmıştır.

```
Command Window
Tepe noktası yüksekliği= 450.48
Maksimum yatay menzil= 655.65
Tepe noktası bileşke hız= 34.20
Tepe noktası zamanı= 9.59
Son Pozisyon= 655.65
Son Pozisyon= 0.00
Son Pozisyon= 0.51
Son Hız Bileşke= 99.95
Son Uçuş Açısı= -69.99
Son Uçuş Zamanı= 19.17
```

## Benzetimin Doğrulanması

Doğrulama Başlangıç Koşul Değerleri	
	Değer
Pozisyon [m]	[0, 0, 0]
Hız (bileşke) [m/s]	100
Uçuş Yolu Açısı [derece]	70

Benzetim Çıktı Formatı	
	Değer
Tepe Noktası Yüksekliği [m]	450.48
Tepe Noktası Hızı (bileşke) [m/s]	34.20
Tepe Noktası Zamanı [s]	9.59
Son Pozisyon [m]	[655.65, 0, 0]
Son Hız (bileşke) [m/s]	99.95
Son Uçuş Yolu Açısı [derece]	-69.99
Son Uçuş Zamanı [s]	19.17

Test Videoları Linki:

Simulink: <https://youtu.be/3eUa5lz150Q>

Script: <https://youtu.be/BYgvY1qV2zg>

Gui: <https://youtu.be/exUDkkCUdvA>

## Referanslar

- 1) Anekwe, U. O. (2019) Equations of an Oblique Projectile Motion Without Calculus.
- 2) fizik.narkive.info.tr