Parametrik Adams Modeli ve Python Kodu Kullanarak Süspansiyon Analizi

Cahit Bartu YAZICI

İçerik

- Adams Modeli ve Metodun Çalışması
- Modelin Otomatize Edilmesi
- Modelin Manipülasyonu
- Modelin Çıktıları
- Sonuç

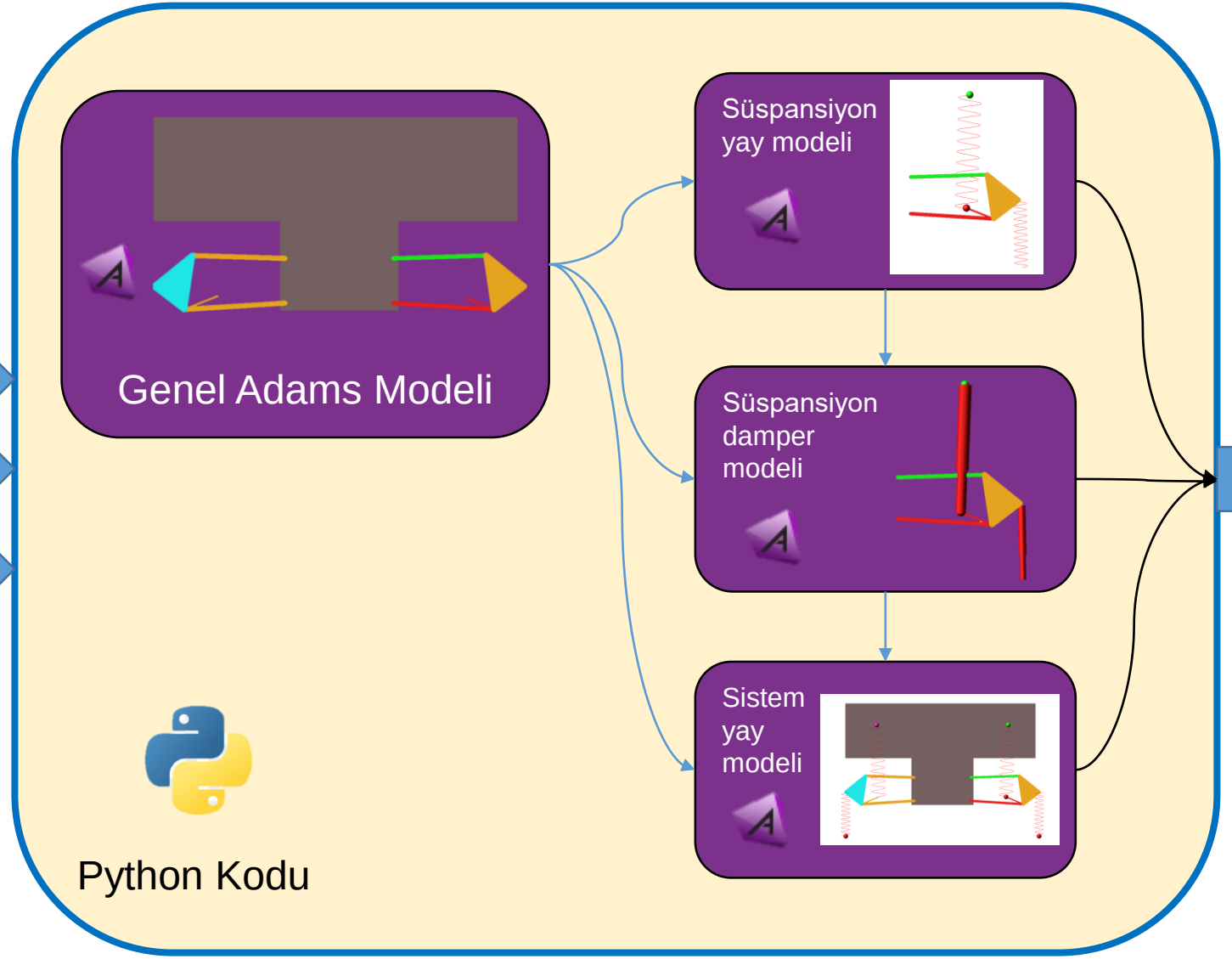
Çalışmanın Amacı

Çalışmanın Amacı

- Konsept aşamasında kinematik, yay davranışı yada damper davranışında meydana gelecek değişikliklerin etkilerini hızlı bir şekilde incelemek adına parametrik bir Adams modeli ve Python kodu oluşturulmuştur.
- 2 boyutlu süspansiyon geometrisi kullanılarak aracın; tekere indirgenmiş yay, damper davranışları ve doğal frekanslarını bulmak amaçlanmaktadır.
- Elde edilen bu bilgiler aynı araçta farklı süspansiyon konfigürasyonlarını yada farklı araçları karşılaştırmak için kullanılabilir.
- Aynı zamanda elde edilen tekere indirgenmiş lineer olmayan yay ve damper davranışları, yüksek hassasiyet ile sürüş konfor simülasyonları yapmak üzere yarım araç modellerinde kullanılabilir.

Çalışmanın Amacı

- Süspansiyon noktaları
- Yay damper dataları
- İstenilen simülasyonlar



Sunuç .txt dosyası

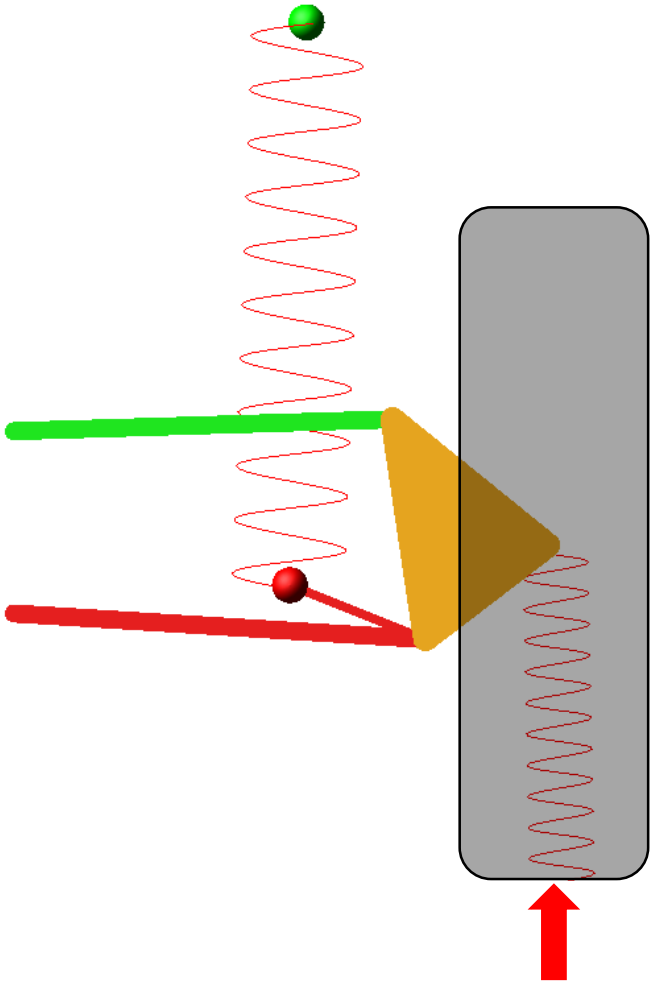
Adams Modeli

Adams Modeli

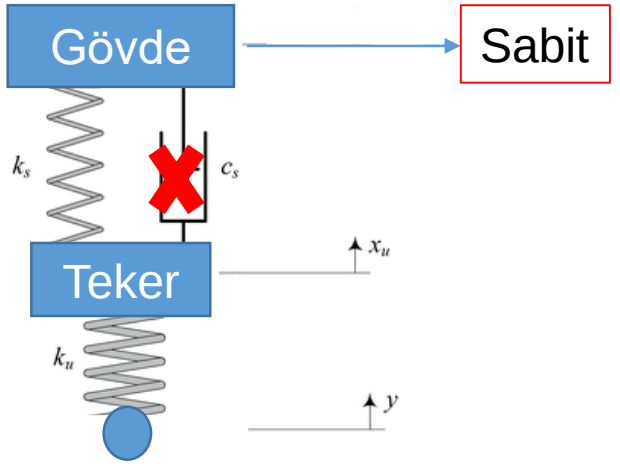
- Adams View programında çift salıncak kollu iki süspansiyon ve gövde modelleri 2 boyutlu olarak kurulmuştur.
- Modelde değişiklikler yapılarak istenilen 3 konfigürasyon elde edilir.

Adams Modeli

1. Konfigürasyon

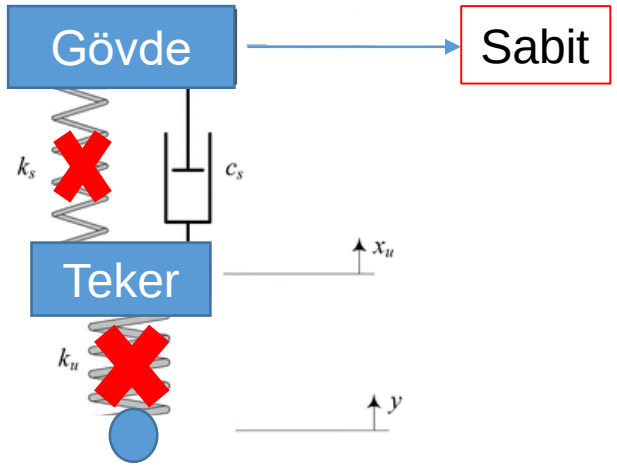


- Süspansiyon sistemine yaylar eklenip teker gezintisi simüle edilerek tekere indirgenmiş yay davranışı ve aktarım oranı bulunabilir.
- Aktarım oranı teker hareketine oranla yayın ne kadar hareket ettiğidir.
- Teker hareketi teker yol kesişme noktasında bulunan bir motion ile sağlanmaktadır.

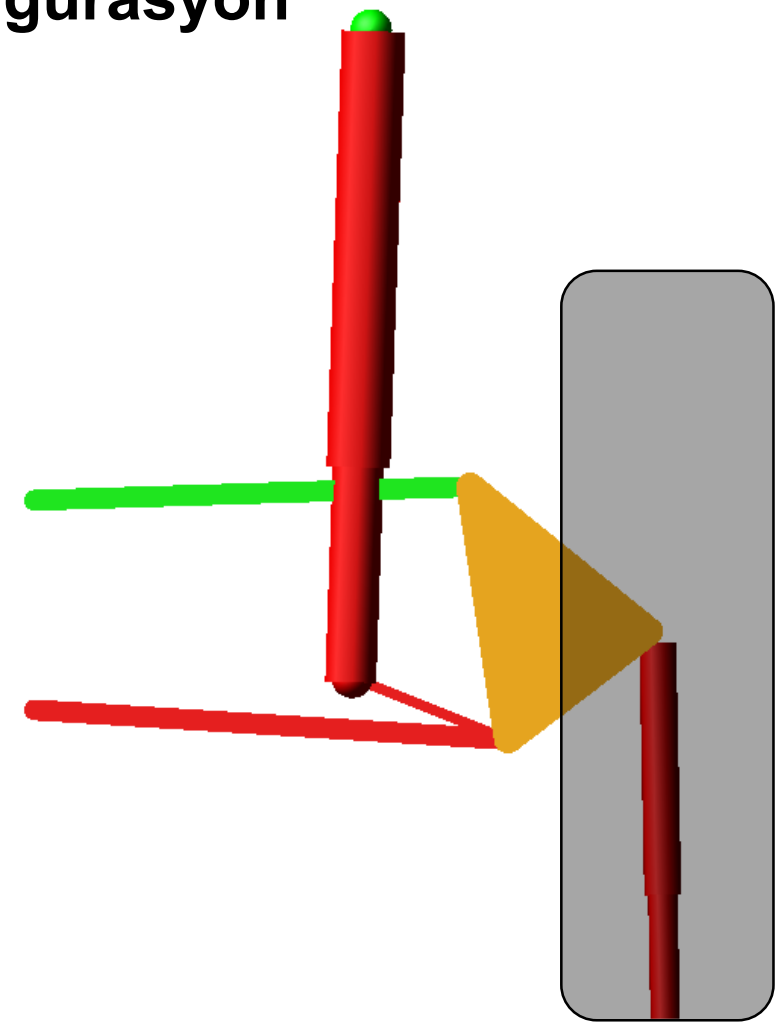


Adams Modeli

- Tekere indirgenmiş damper davranışını bulmak için süspansiyon sistemine damper eklenir.
- Tekere indirgenmiş damper davranışı teker hızı ve süspansiyonun pozisyonuna bağlı bir yüzey oluşturmaktadır.



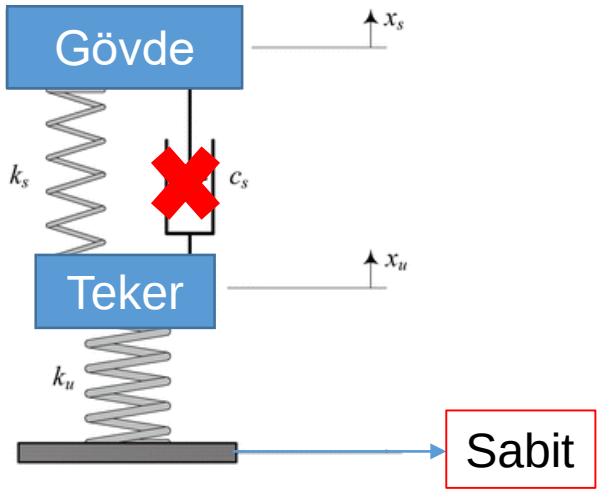
2. Konfigürasyon

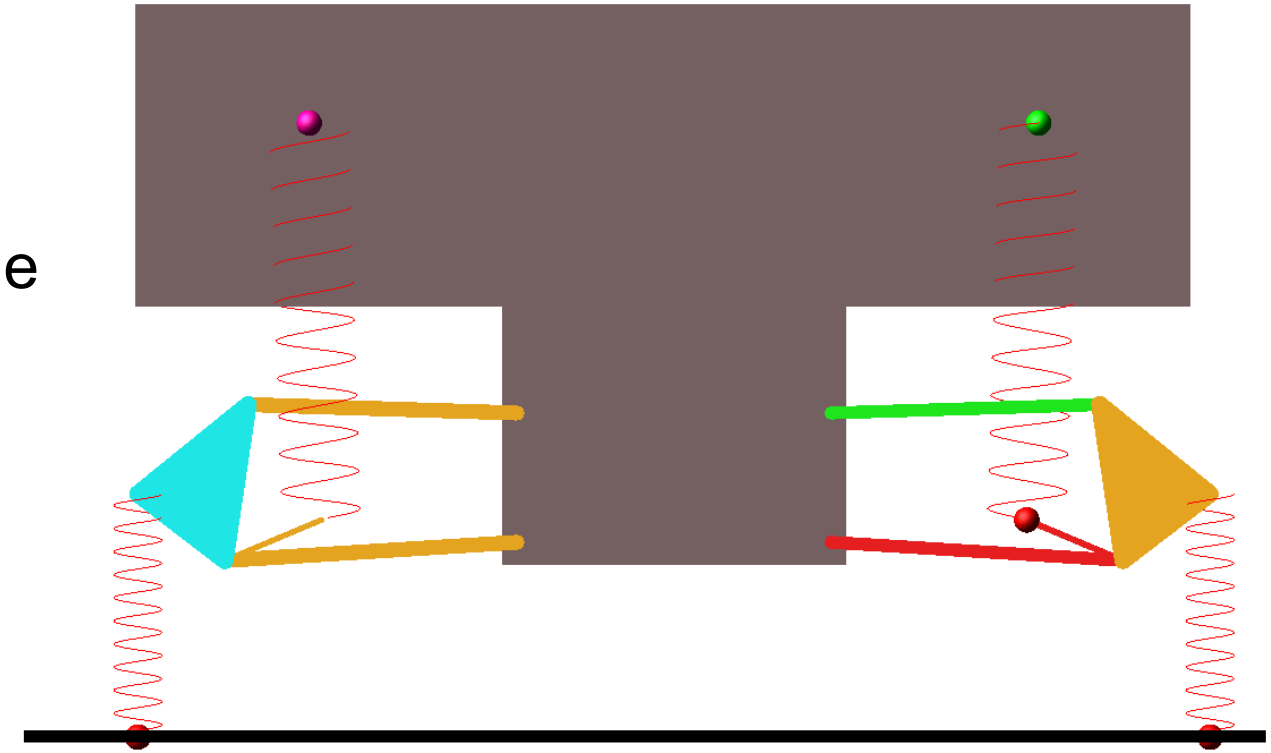


Adams Modeli

3. Konfigürasyon

- Doğal frekans analizi için hareket edebilen bir gövde ve süspansiyon ve teker yayları kullanılmaktadır.
- Teker-yer bağlantıları y düzleminde hareket edebilmekte ancak z düzleminde hareket edememektedirler.



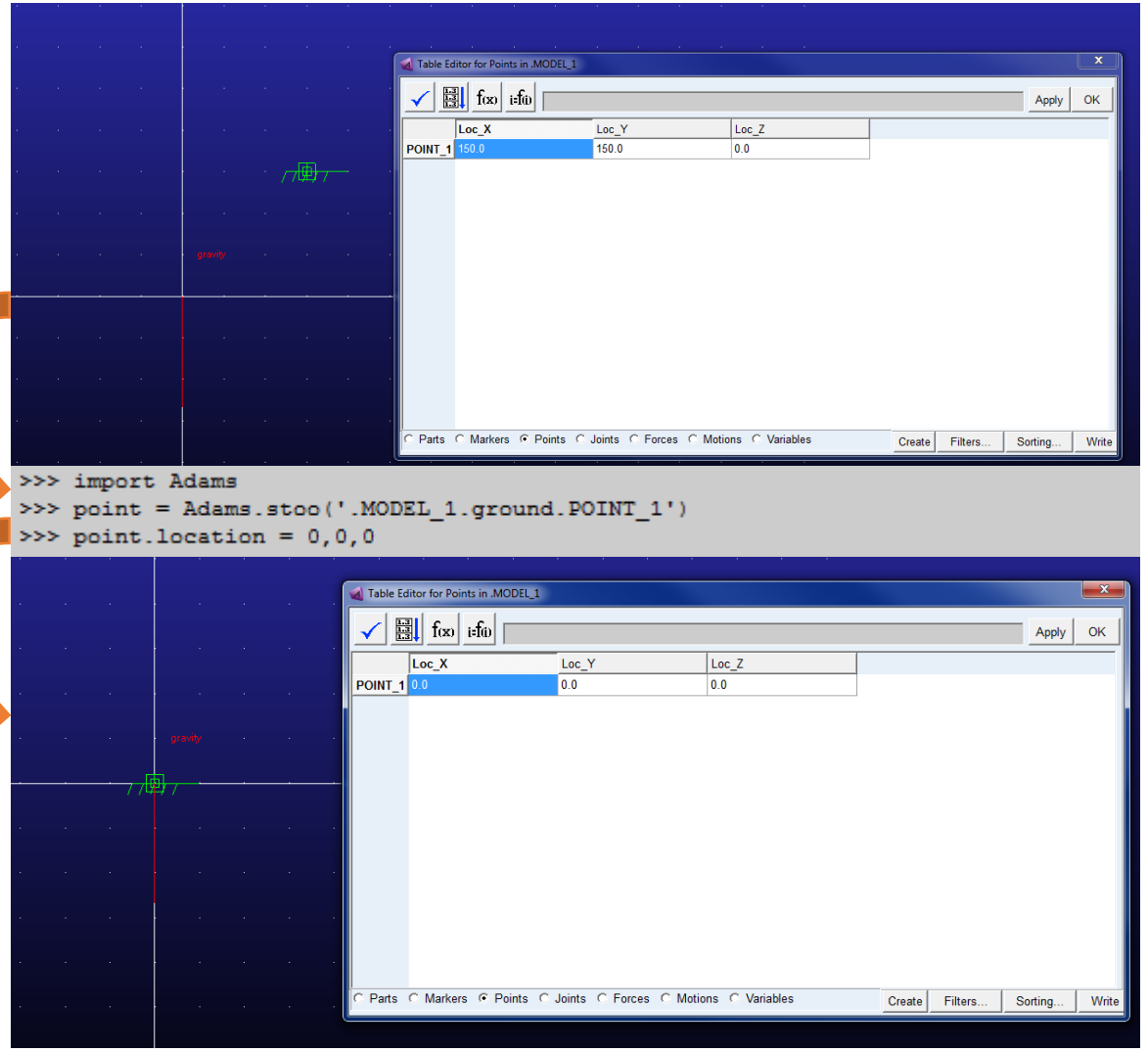


Modelin Otomatize Edilmesi

Modelin Otomatize Edilmesi

- Adams View programında modeller design veriable kullanmak dışında Python objeleri oluşturularak da parametize edilebilir.

- Pythonda oluşturulan objeler, Python kodu kullanılarak manipüle edilebilir



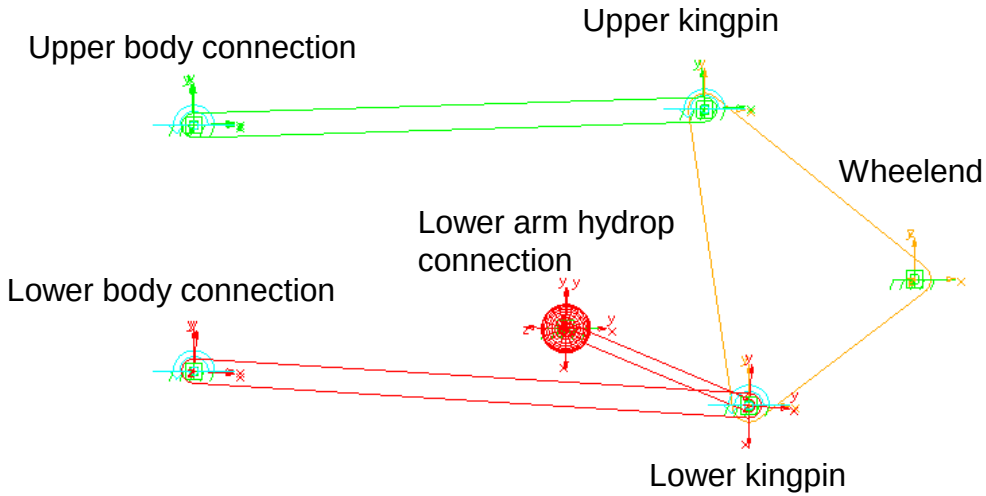
Modelin Otomatize Edilmesi

- Modelin hardpointleri Python kodunda parametize edilmiştir.

```
Upper_Body_Connection = Adams.stoo('.Quarter_Car.ground.Upper_Body_Connection')
Upper_Kingpin = Adams.stoo('.Quarter_Car.ground.Upper_Kingpin')
Lower_Body_Connection = Adams.stoo('.Quarter_Car.ground.Lower_Body_Connection')
Lower_Kingpin = Adams.stoo('.Quarter_Car.ground.Lower_Kingpin')
Lower_Arm_Hydrop_Connection = Adams.stoo('.Quarter_Car.ground.Lower_Arm_Hydrop_Connection')
Upper_Body_Connection_Replica = Adams.stoo('.Quarter_Car.ground.Upper_Body_Connection_Replica')
Upper_Kingpin_Replica = Adams.stoo('.Quarter_Car.ground.Upper_Kingpin_Replica')
Lower_Body_Connection_Replica = Adams.stoo('.Quarter_Car.ground.Lower_Body_Connection_Replica')
Lower_Kingpin_Replica = Adams.stoo('.Quarter_Car.ground.Lower_Kingpin_Replica')
Lower_Arm_Hydrop_Connection_Replica = Adams.stoo('.Quarter_Car.ground.Lower_Arm_Hydrop_Connection_Replica')
Body_CoG = Adams.stoo('.Quarter_Car.ground.Body_CoG')
Body = Adams.stoo('.Quarter_Car.Body')

Wheel_Center = Adams.stoo('.Quarter_Car.ground.Wheel_Center')
Wheel_Center_Replica = Adams.stoo('.Quarter_Car.ground.Wheel_Center_Replica')
Hull_Connection = Adams.stoo('.Quarter_Car.ground.Hull_Connection')
Hull_Connection_Replica = Adams.stoo('.Quarter_Car.ground.Hull_Connection_Replica')
Wheel_End = Adams.stoo('.Quarter_Car.wheel_end')
Wheel_End_Replica = Adams.stoo('.Quarter_Car.wheel_end_Replica')
Tire_Motion = Adams.stoo('.Quarter_Car.Tire_Motion')
```

Hull connection



- Modeldeki bütün Markerlar konum bilgilerini Hardpointlerden almaktadır. Bu sebeple sadece hardpointler hareket ettirilerek model geometrisi modifiye edilmebilir.

Modelin Otomatize Edilmesi

- Modelin geometrisi parametize edildikten sonra yay ve damperler modele eklenmektedir.
- Adams command language komutları kullanılarak yay ve damperler oluşturulur.
- Nonlinear yay ve damper dataları oluşturulan .cmd dosyaları ile spline şeklinde Adamsa aktarılır.

Yay oluşturmak için;

```
marker create marker=.Quarter_Car.Hull_Connection.Marker_1 &
   adams_id=100 &
   location=(LOC_RELATIVE_TO({0,0,0}, ground.Hull_Connection)) &
   orientation=0,0,0
marker create marker=.Quarter_Car.lower_control_arm. Marker_2 &
   adams_id=101 &
   location=(LOC_RELATIVE_TO({0,0,0}, lower_arm.Spring_upper)) &
   orientation=0,0,0
force create direct single_component_force &
    single_component_force_name=.Quarter_Car.Spring_Force &
    adams_id=3 &
    type_of_freedom=translational &
    action_only = off &
    i_marker_name=.Quarter_Car.Hull_Connection. Marker_1 &
    j_marker_name=.Quarter_Car.lower_control_arm. Marker_2 &
    function = "AKISPL(.Quarter_Car.Hydrop_Length,0,SPLINE_1, 0)" &
    comments=""
mdi graphic_force object=.Quarter_Car.Spring_Force type=1
group modify group=SELECT_LIST object=.Quarter_Car.Spring_Force
undo end
```

Yayın üst bağlantı noktası markerı

Yayın alt bağlantı noktası markerı

Spline oluşturmak için;

```
file testdata read  splines &
          use_file_column_labels = yes   &
          independent_column_index = 1  &
          file_name = "Örnek_Spline.txt" &
          model_name = Quarter_Car  &
            &
```

Modelin Otomatize Edilmesi

Python ile simülasyon başlatılması

Bu örnekte

- Model 1. konfigürasyona getir
- Simülasyon başlatılır
- Sonuçlar dışarı aktarılır

```
if stiffness_calculation == 1:
        Tire_Motion.function = 'IF( time-1 :  '+str(Rebound_travel)+' , 0 , '+str((Bump_travel-Rebound_travel)/4)+' )'
        Adams.execute_cmd('simulation single scripted sim_script_name = .Quarter_Car.Spring_Simulation')
        Adams.read_command_file(wheel_travel)
        Adams.read_command_file(tire_force)
        Adams.read_command_file(spring_force)

        results_wheel_travel = open('Wheel_Travel.txt','r').readlines()
        results_wheel_force = open('Wheel_Force.txt','r').readlines()
        results_spring_force = open('Spring_Force.txt','r').readlines()

        wheel_travel = [ ]
        wheel_force = [ ]
        Spring_Force = [ ]


        for x in range(108,508):
                wheel_travel.append(float(results_wheel_travel[x].split()[1]))
                wheel_force.append(float(results_wheel_force[x].split()[2]))
                Spring_Force.append(float(results_spring_force[x].split()[2]))
                tire_stifness = open('Dosya\\'+stiffness_output_file+'.txt','a')
                tire_stifness.write(str(wheel_travel[x-108]-564.2)+'    '+str(wheel_force[x-108])+'\n')
                tire_stifness.close()
                transmission_ratio = open('Dosya\\'+stiffness_output_file+'_transmission_ratio.txt','a')
                transmission_ratio.write(str(wheel_travel[x-108]-564.2)+'       '+str(wheel_force[x-108]/Spring_Force[x-108])+'\n')
                transmission_ratio.close()


        Adams.execute_cmd('simulation single reset')
```

1. Saniyeye kadar en alta gel sonra en üste çık

Simülasyon başlatma

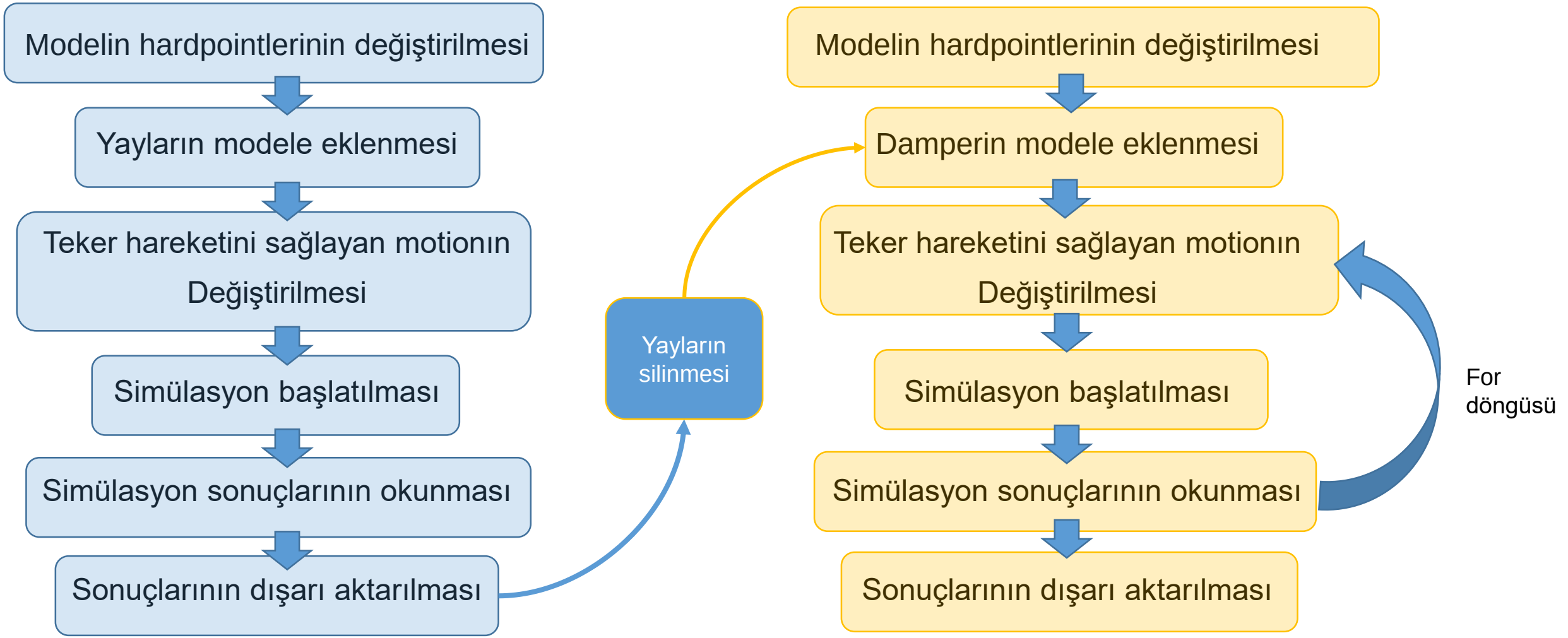
Simülasyon sonuçlarının okunması

Sonuçların işlenmesi

Modelin Otomatize Edilmesi

Yay davranışı simülasyonları:

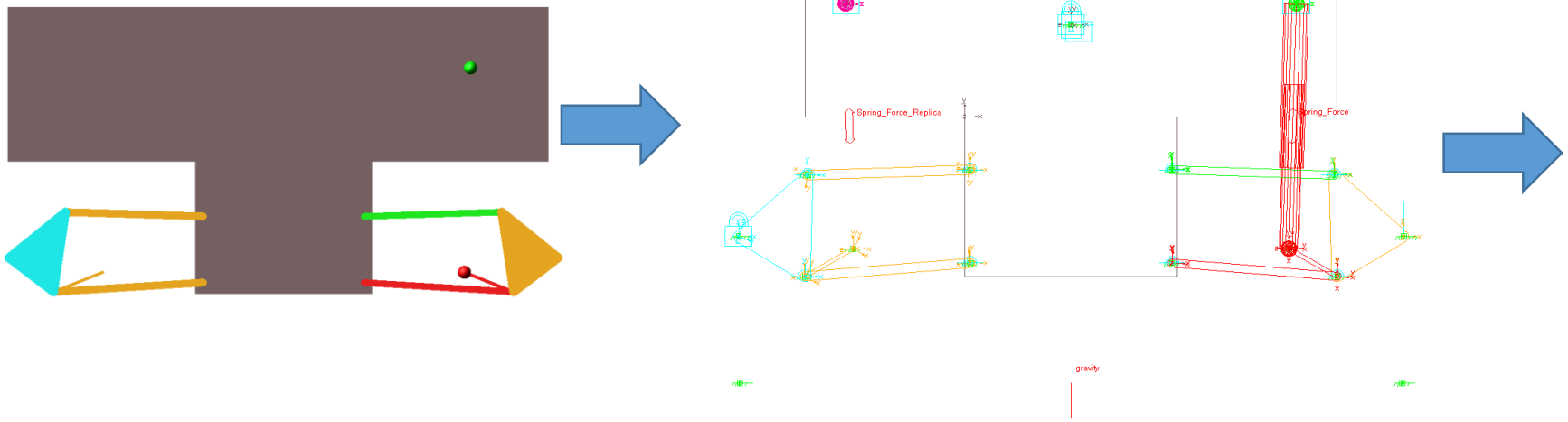
Damper davranışı simülasyonları:

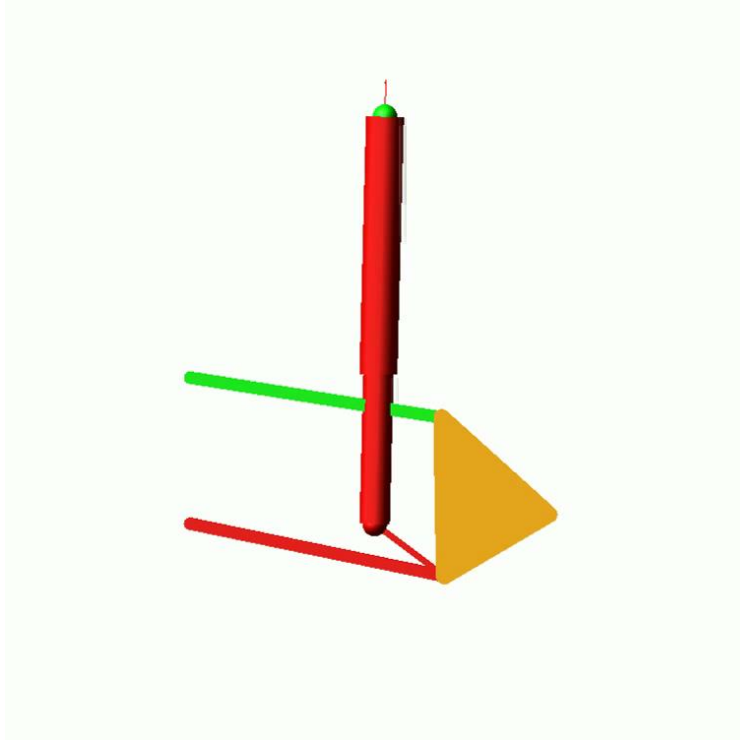


Modelin Manipülasyonu

Adams Modeli

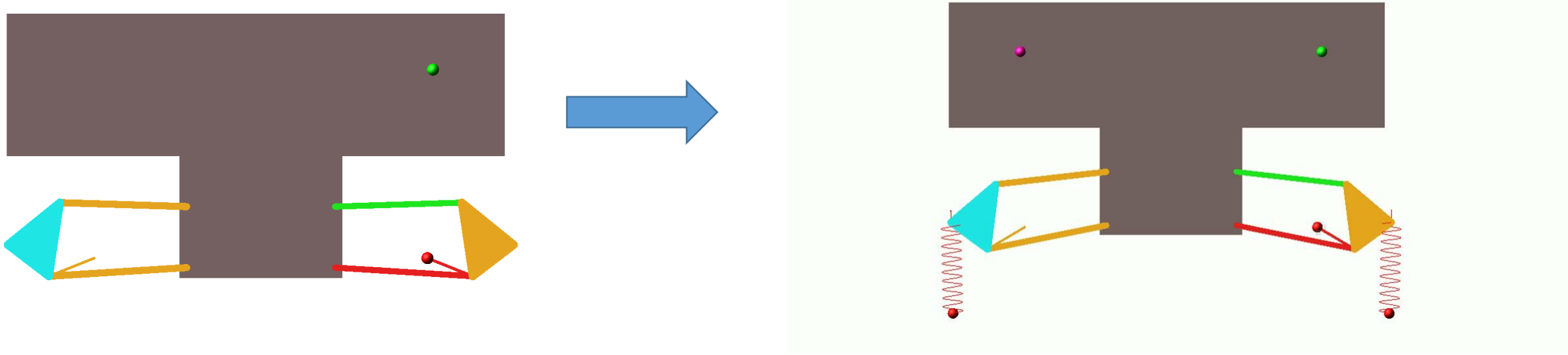
- Genel modelin yapılan simülasyonlar için uygun hale getirilmesi için bazı değişiklikler yapılmaktadır.
- Tekere indirgenmiş yay ve damper davranışı bulunması simülasyonları için gövde yere sabitlenmektedir.





Adams Modeli

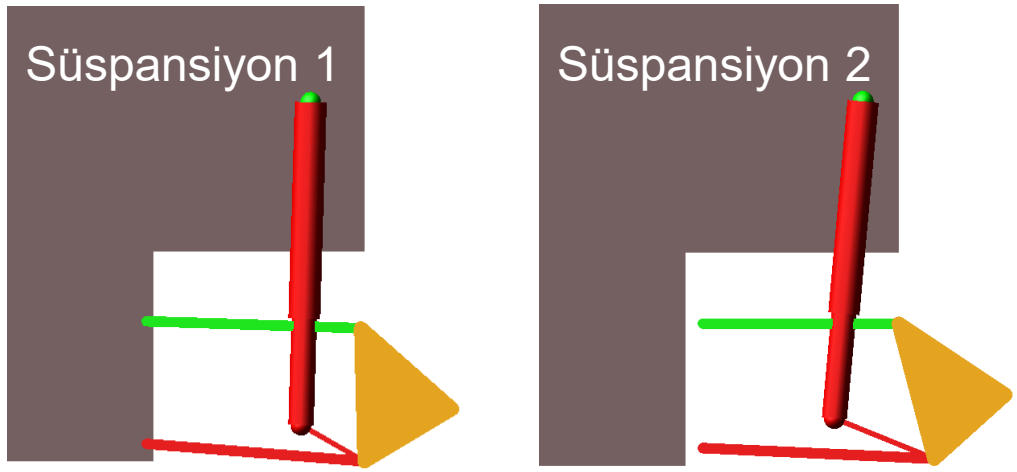
- Sistemin doğal frekansını bulmak için gövde serbest bırakılmış ve sisteme yaylar eklenmiştir.
- Sönümleme oranını bulmak için damperler modele eklenmiştir.

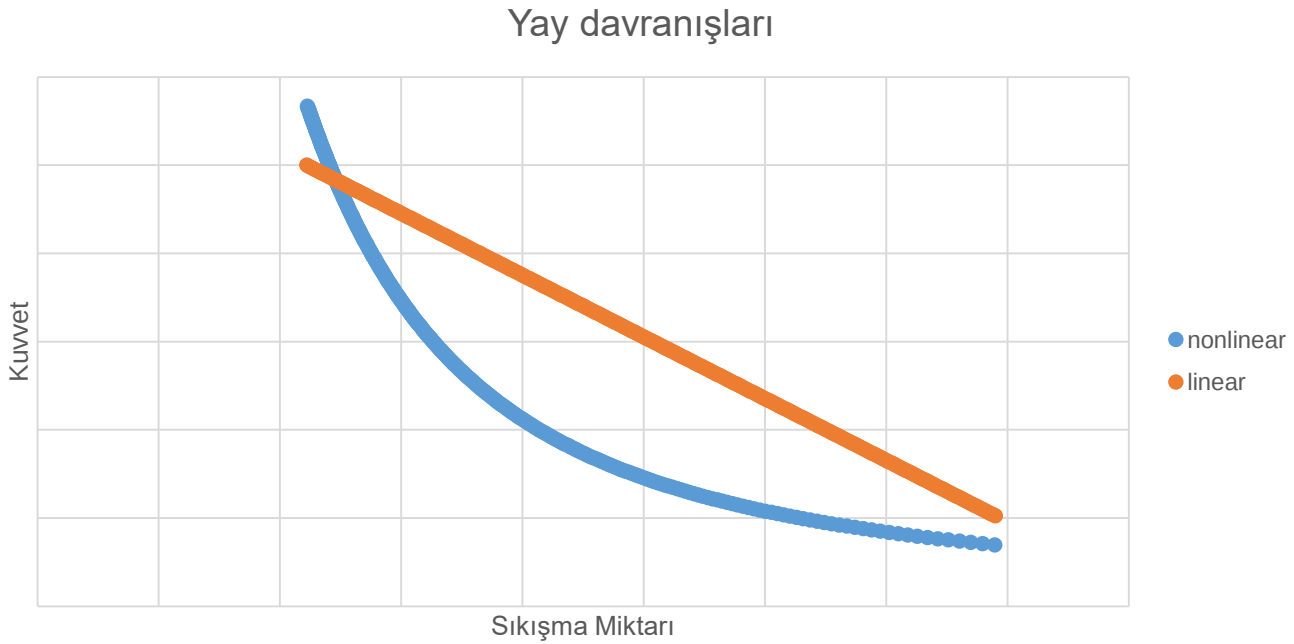


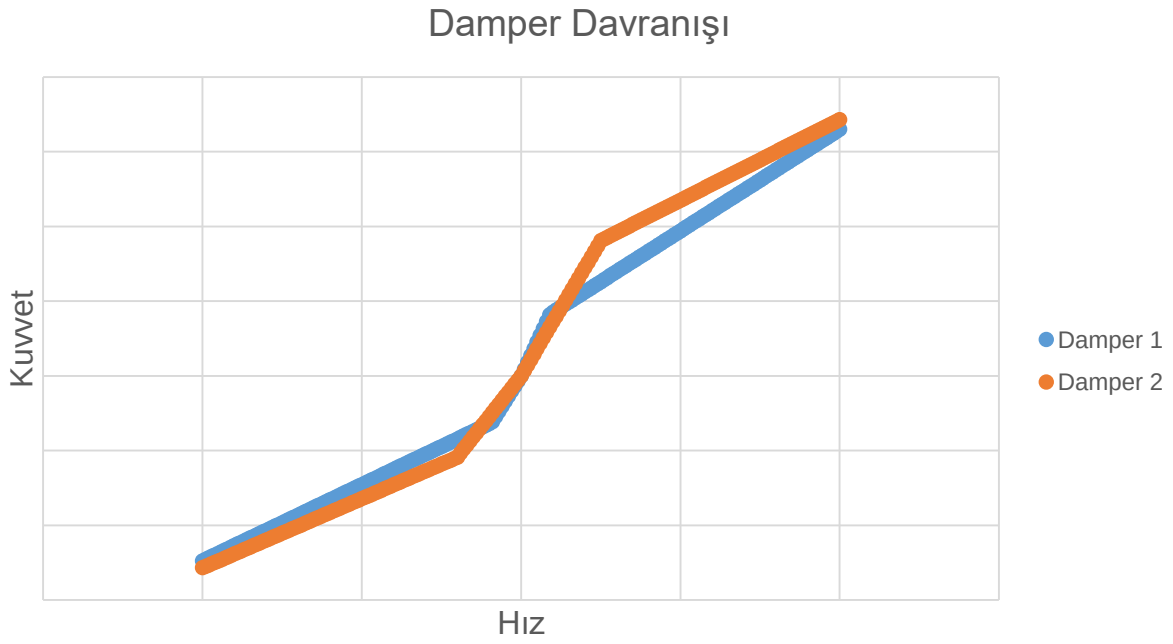
Modelin Çıktıları

Modelin Çıktıları

Modelin kullanım alanlarını göstermek adına iki farklı süspansiyon kinematiği, iki farklı yay eğrisi ve iki farklı sönümleme elemanı eğrisi kullanılmıştır.



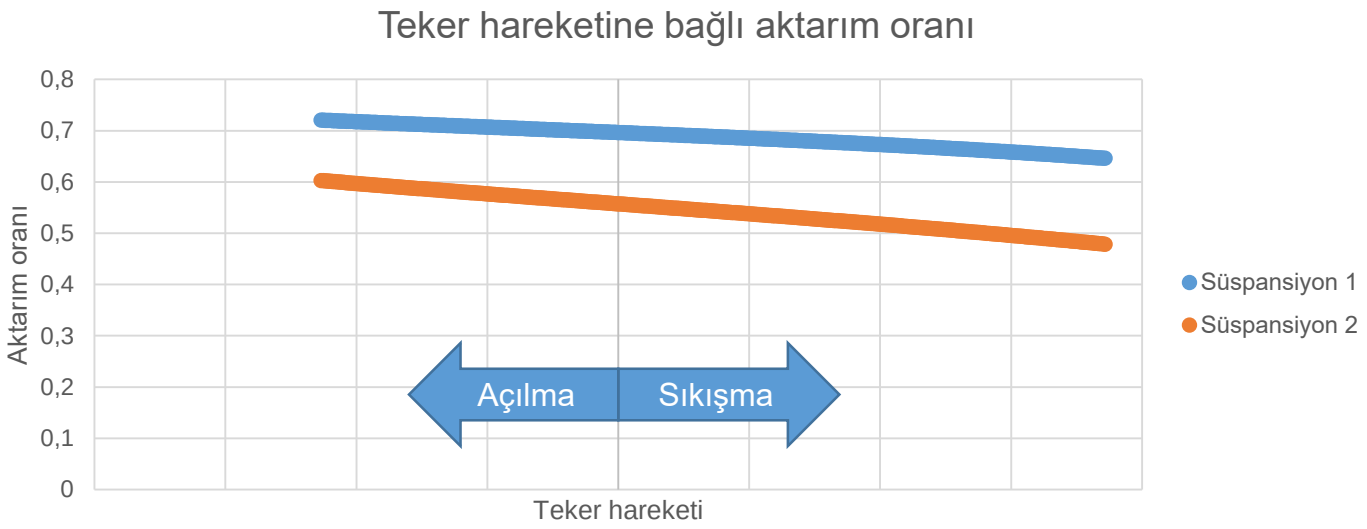


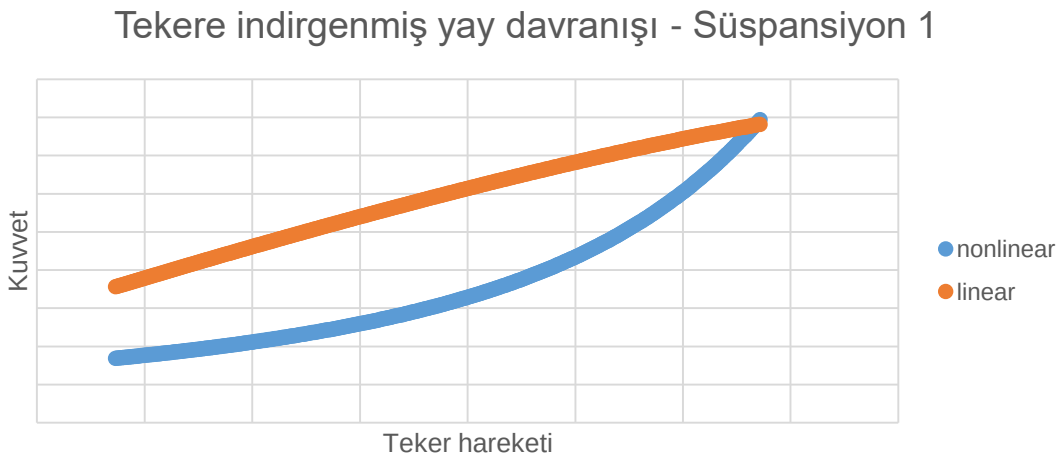


Modelin Çıktıları

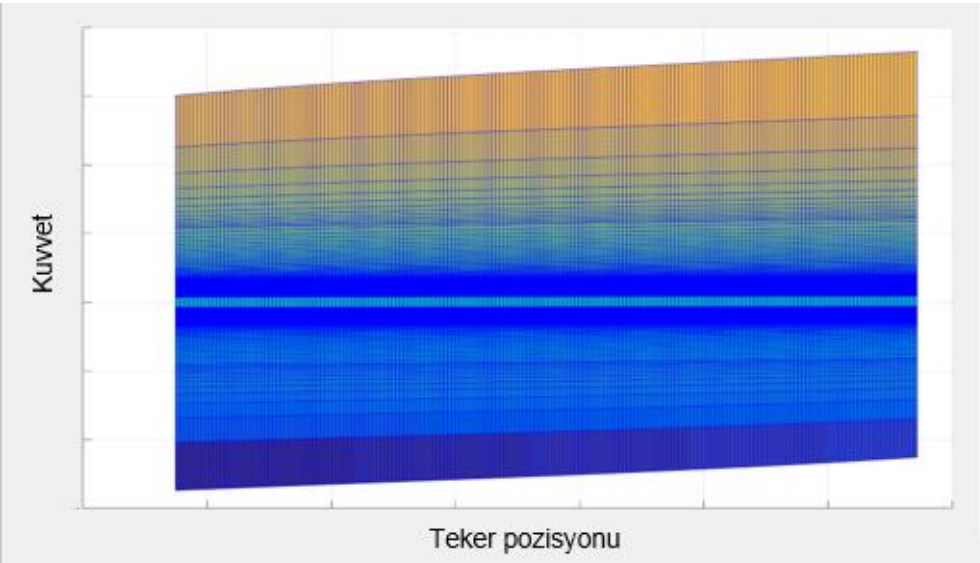
Simülasyon 1

- Süspansiyon sisteminin teker hareketine göre kuvvet oranı ve tekere indirgenmiş yay davranışı hesaplanmaktadır.
- Teker hareketine göre kuvvet oranı, süspansiyon kinematiğinin etkisini görselleştirmek için kullanılabilir.
- İki farklı süspansiyon sistemi karşılaştırmak için kullanılabilir.
- Kinematiğin yay davranışına etkisi incelenebilir.



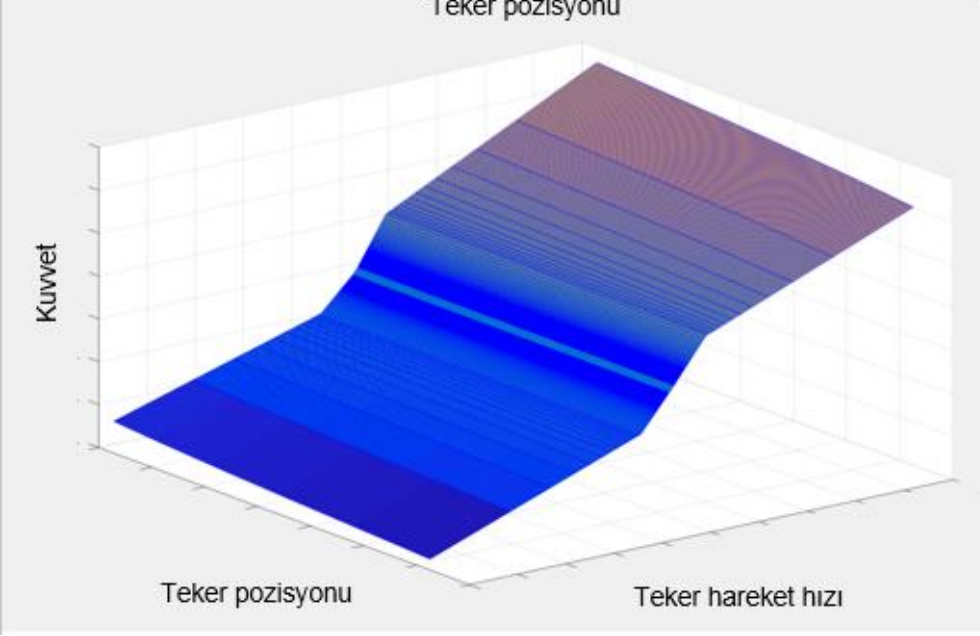


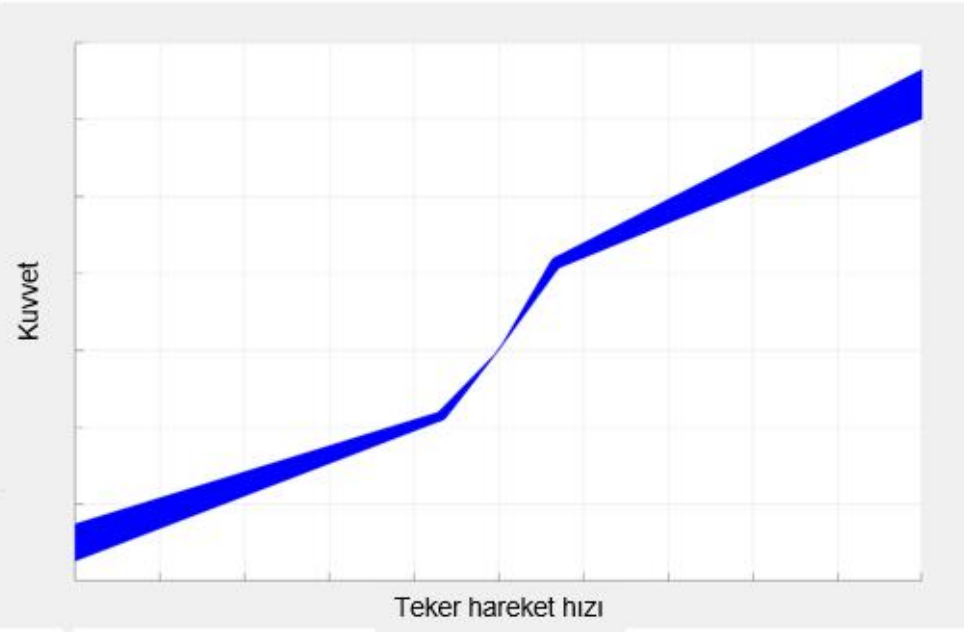
Modelin Çıktıları



Simülasyon 2

- Tekere indirgenmiş damper yüzeyi sayesinde linear olmayan damper davranışının teker hızı ve teker haraketine göre değişimi incelenebilir.
- İki farklı süspansiyonu yada iki farklı damperi incelemek için kullanılabilir.



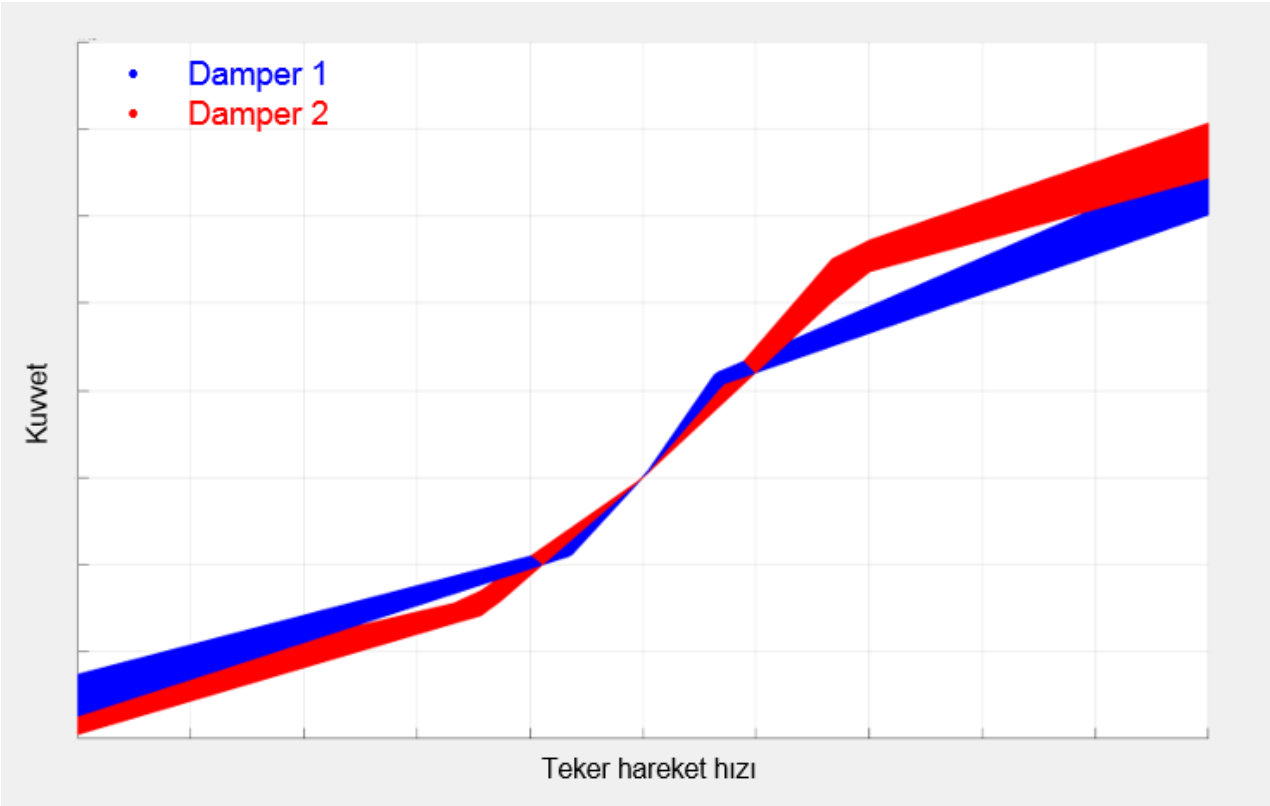


Modelin Çıktıları

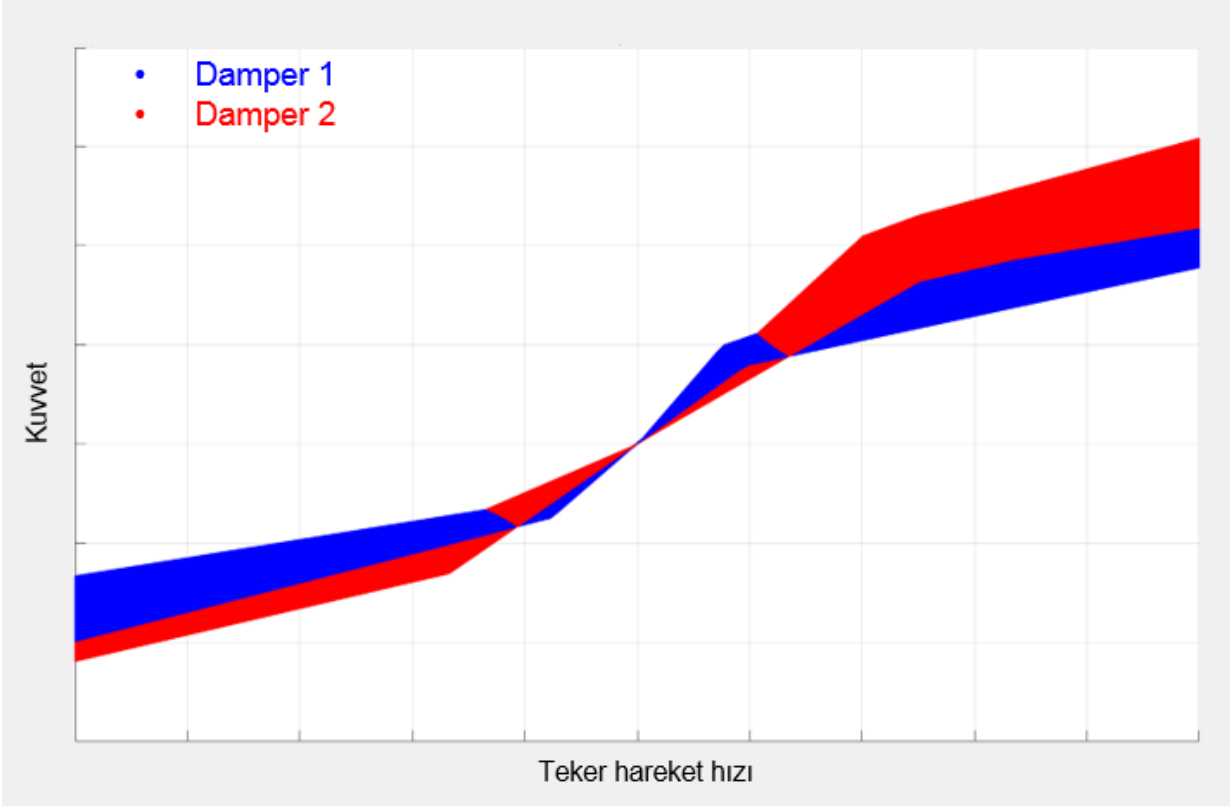
Simülasyon 2

- Aynı süspansiyonda farklı damperlerin etkilerinin görselleştirilmesi;

Süspansiyon 1



Süspansiyon 2

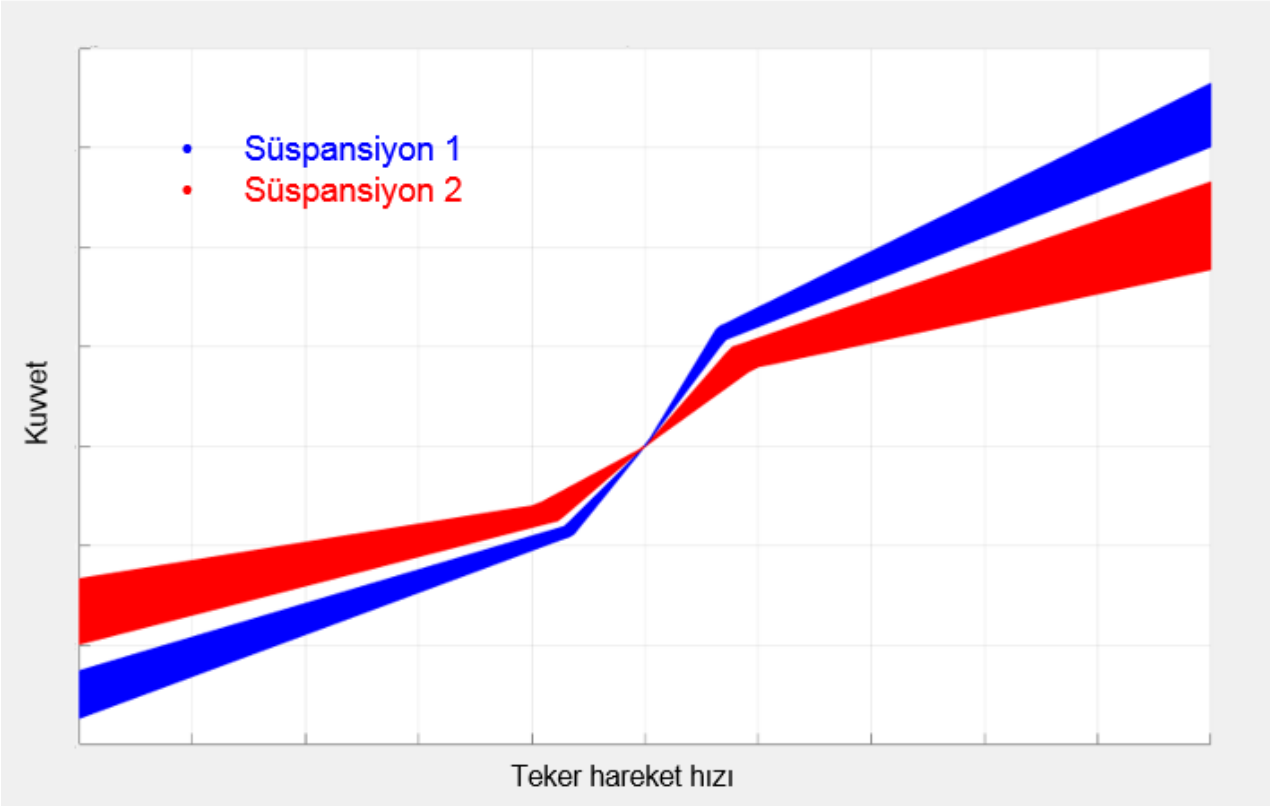


Modelin Çıktıları

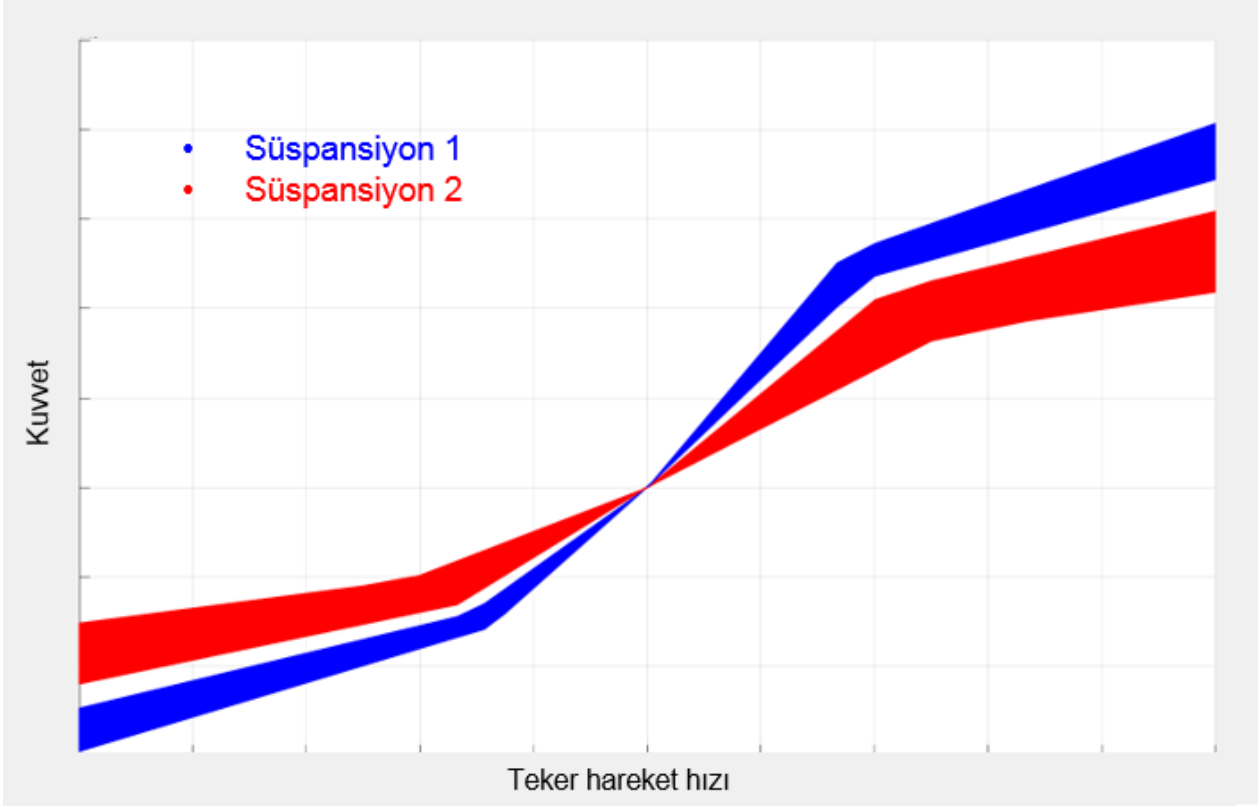
Simülasyon 2

- Aynı damperin farklı süspansiyonlarda görselleştirilmesi;

Damper 1



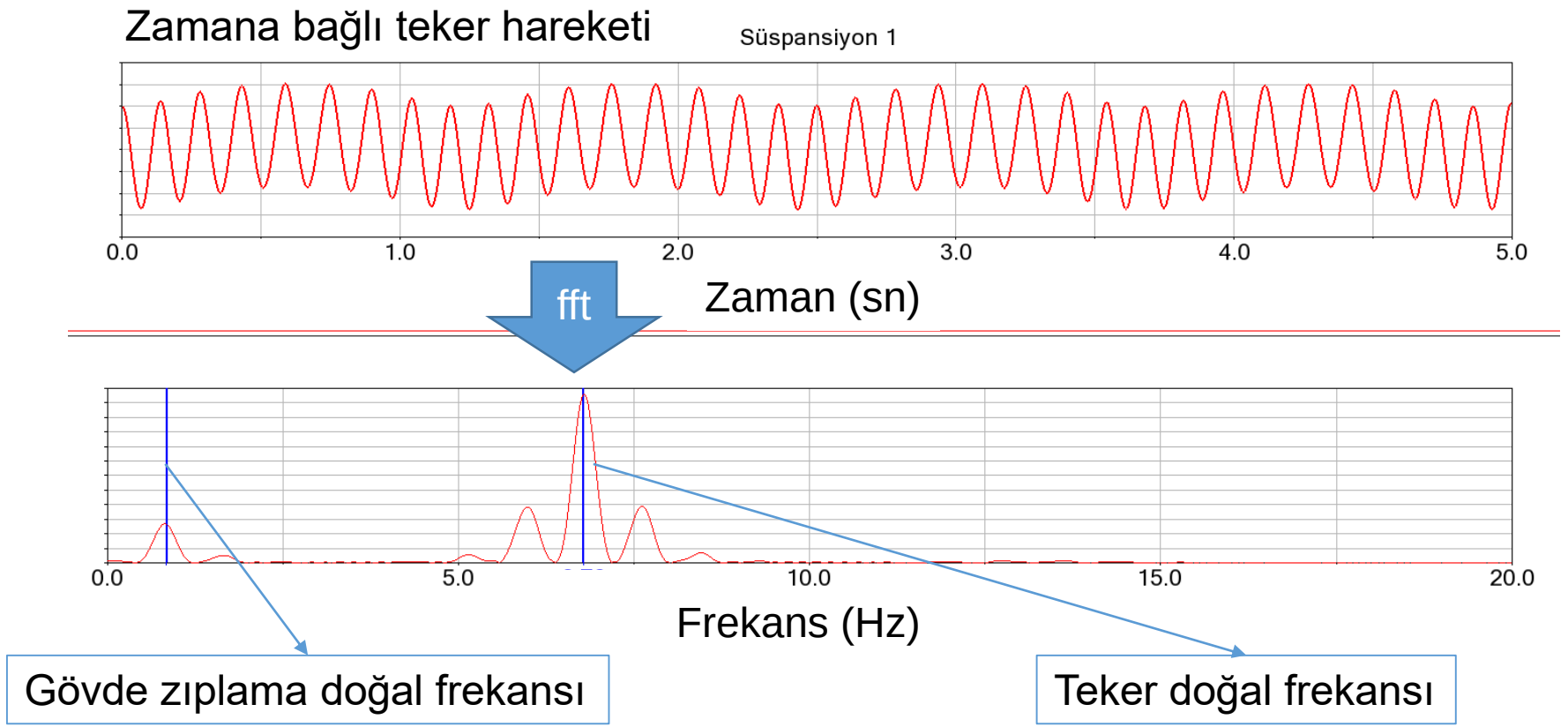
Damper 2



Modelin Çıktıları

Simülasyon 3

- Doğal frekans hesabı simülasyon sonuçları hem araç gövdesi doğal frekansını hem de teker doğal frekanslarını hesaplamaktadır.
- Tekerlerin zamana bağlı düşey yer değiştirme datalarının FFTsi alınarak istenilen değerler bulunabilir.



Sonuç

Sonuç

- Kurulan genel süspansiyon modeli ile tekere indirgenmiş yay ve damper davranışları, teker hareketine bağlı hareket oranı ve doğal frekans hesapları yapılabilir.
- Süspansiyon parametreleri Python koduna girildikten sonra Python kodu, modelde gereken değişiklikleri otomatik olarak yapmakta ve gereken simülasyonları koşturup istenilen sonuçları yazdırmaktadır.
- Önerilen metot yay ve damper davranışlarındaki değişimlerin etkisini incelemek ve farklı süspansiyon sistemlerini karşılaştırmak için kullanılabilir.

Olası gelişme alanları;

- Python koduna optimizasyon algoritmaları eklenerek bir veya daha çok parametrenin belirlenen önceliklere göre optimize edilmesi sağlanabilir.
- Sistemin sönümleme oranı, aracın yatma eğilimi gibi incelenmek istenen başka alanlar modele eklenebilir.

Dinlediğiniz için teşekkürler.

www.fnss.com.tr
www.fnsssocial.com