

# İGÜ Uçak Mühendisliği - UCK 302 AERODİNAMİK II

2021-2022 Bahar Dönemi

## ÖDEV 1

### HAZIRLAYANLAR

190414151

200414359

200414361

210414983

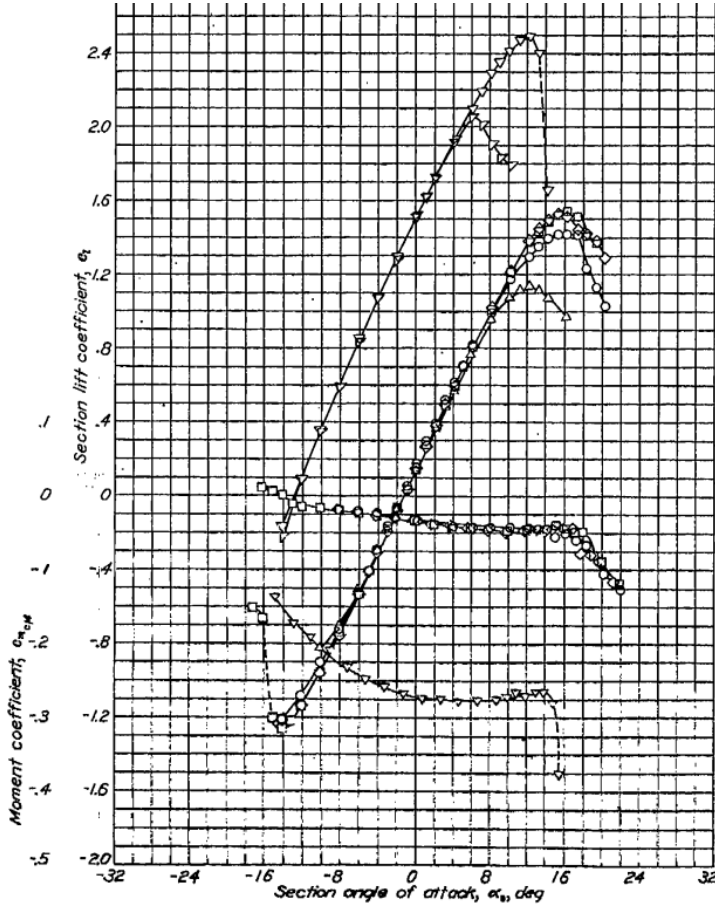
1) NACA 652-215 kamburluklu kanat profilinin NACA TR 824 de yer alan deneysel sonuçlarından sıfır taşıma açısı ve taşıma eğrisi eğimi belirlendi. Taşıma eğrisi eğimini bulmak için Excel kullanıldı. NACA raporunda belirtilmiş olan değer tablosunda 0 değerine en yakın olan taşıma taşıma katsayısı değerine bakılarak sıfır taşıma açısı bulundu. Excel kullanılarak, belirli değerlerdeki hücum açıları ile eğim hesaplandı ve böylece taşıma eğrisi eğimi bulundu.

	A								B	C	D	E	F	G	H
67	6.250	0.7596	0.01332	0.00754	-0.0163	0.4447	0.9191								
68	6.500	0.7431	0.01453	0.00798	-0.0088	0.3085	0.9351								
69	6.750	0.7247	0.01618	0.00888	-0.0020	0.1764	0.9568								
70	7.000	0.7283	0.01837	0.01043	-0.0004	0.0856	0.9769					hücum açısı	Değerler		Eğim
71	7.250	0.7490	0.01980	0.01179	-0.0015	0.0695	1.0000					10	1,0172		0,08711
72	7.500	0.7598	0.02057	0.01258	-0.0002	0.0639	1.0000					0	0,1461		
73	7.750	0.7688	0.02186	0.01380	0.0011	0.0591	1.0000								
74	8.000	0.7855	0.02280	0.01477	0.0017	0.0553	1.0000								
75	8.250	0.8016	0.02383	0.01579	0.0024	0.0523	1.0000								
76	8.500	0.8178	0.02501	0.01692	0.0032	0.0500	1.0000								
77	8.750	0.8400	0.02638	0.01825	0.0036	0.0480	1.0000								
78	9.000	0.8643	0.02732	0.01925	0.0038	0.0464	1.0000								
79	9.250	0.8885	0.02832	0.02029	0.0040	0.0442	1.0000								
80	9.500	0.9152	0.02944	0.02143	0.0038	0.0425	1.0000								
81	9.750	0.9495	0.03092	0.02290	0.0028	0.0412	1.0000								
82	10.000	1.0172	0.03488	0.02696	-0.0032	0.0398	1.0000								
83	10.250	1.0388	0.03665	0.02896	-0.0028	0.0395	1.0000								
84	10.500	1.0539	0.03827	0.03084	-0.0014	0.0390	1.0000								
85	10.750	1.0656	0.04005	0.03289	0.0002	0.0384	1.0000								
86	11.000	1.0791	0.04253	0.03565	0.0015	0.0382	1.0000								
87	11.250	1.0895	0.04553	0.03893	0.0029	0.0384	1.0000								
88	11.500	1.0959	0.04884	0.04252	0.0045	0.0388	1.0000								
89	11.750	1.1008	0.05301	0.04693	0.0059	0.0393	1.0000								
90	12.000	1.1144	0.05570	0.04980	0.0069	0.0405	1.0000								
91	12.250	1.0473	0.06262	0.05760	0.0146	0.0468	1.0000								
92	12.500	0.9380	0.05869	0.05421	0.0206	0.0455	1.0000								
93	12.750	0.9171	0.06435	0.06006	0.0212	0.0467	1.0000								

Sıfır taşıma hücum açısı:  $-1,250^\circ$  olarak bulundu.

Taşıma eğrisi eğimi:  $0,08711^\circ$  olarak bulundu.

Sonsuz bir kanadın taşıma eğrisi eğimi lineer eğri üzerindeki herhangi iki noktadan da elde edilebilir. NACA 65(2)215 profili için grafik üzerinde bulunan eğriye bakılarak iki tane etkin açı değerindeki taşıma katsayıları bulunur ve eğim hesaplanır.



**NACA 652-215**  
[Stations and ordinates given in percent of airfoil chord]

Upper surface		Lower surface	
Station	Ordinate	Station	Ordinate
0	0	0	0
.406	1.170	.594	-1.070
.645	1.422	.855	-1.282
1.132	1.805	1.368	-1.591
2.385	2.506	2.635	-2.134
4.848	3.557	5.152	-2.925
7.342	4.380	7.658	-3.532
9.841	5.080	10.159	-4.035
14.848	6.175	15.182	-4.829
19.868	7.018	20.137	-5.426
24.882	7.668	25.118	-5.868
29.904	8.123	30.096	-6.179
34.927	8.426	35.073	-6.366
39.952	8.569	40.048	-6.427
44.976	8.522	45.024	-6.332
50.000	8.271	50.000	-6.065
55.021	7.815	54.979	-5.625
60.039	7.189	59.961	-5.047
65.053	6.433	64.947	-4.373
70.062	5.572	69.938	-3.628
75.065	4.638	74.935	-2.848
80.063	3.658	79.937	-2.061
85.055	2.649	84.945	-1.308
90.040	1.660	89.960	-0.626
95.020	.744	94.980	-.112
100.000	0	100.000	0

L. E. radius: 1.505  
Slope of radius through L. E.: 0.084

2) Üç-boyutlu kanatların Prandtl taşıyıcı çizgi teorisiyle incelenmesine yönelik bir bilgisayar programı yazıldı. Bu kodlar Python programı aracılığıyla yazıldı.

```
import math
import numpy as np
import matplotlib.pyplot as plt

# DEĞERLER
N = 11 # İstasyon Sayısı
S = 20.67 # Kanat Alanı [m^2] - Wing Area
AR = 9.08 # Açıklık Oranı - Aspect Ratio
TR = 0.80 # Taper Ratio
alpha_hucum = 10.0 # Hücüm Açısı [derece] - Twist Angle
alpha_sonsuz = 5.98 # Taşıma Eğrisi Eğimi [derece] - Lift Curve Slope
alpha_zero_lift = -1.250 # Sıfır Taşıma Açısı [derece] - Zero Lift Angle of Attack

# HESAPLAMALAR
```

```

b = math.sqrt(AR * S) # Kanat Açıklığı (m)
print("b değeri: ", b)
MAC = S / b # Ortalama Veter Uzunluğu (m)
kok_chord = (1.5 * (1 + TR) * MAC) / (1 + TR + TR ** 2) # Kök kanat[m] -
root chord

theta = np.linspace((math.pi / (2 * N)), (math.pi / 2), N) # Dizi
oluşturuldu.

k = (b / 2) * np.cos(theta)
c = kok_chord * (1 - (1 - TR) * np.cos(theta))
mu = alpha_sonsuz * c / (4 * b)
mu_hesaplama_2 = mu * (np.array(alpha_hucum) - alpha_zero_lift)
mu_hesaplama_2_rad = np.deg2rad(mu_hesaplama_2)

teta_grafik = []
for i in range(1, 2 * N + 1, 2):
    teta_grafik_iter = np.sin(i * theta) * (1 + (mu * i) /
(np.sin(list(theta))))
    teta_grafik.append(teta_grafik_iter)

denklem = np.asarray(teta_grafik)
denklem_transpoz = np.transpose(denklem)
denklem_is_active = np.linalg.inv(denklem_transpoz)

new_matrix = np.matmul(denklem_is_active, mu_hesaplama_2_rad)
alan_deneme_sayilari = np.divide((4 * b), c)

iter0 = (np.sin((1) * theta)) * new_matrix[0] * alan_deneme_sayilari
iter1 = (np.sin((3) * theta)) * new_matrix[1] * alan_deneme_sayilari
iter2 = (np.sin((5) * theta)) * new_matrix[2] * alan_deneme_sayilari
iter3 = (np.sin((7) * theta)) * new_matrix[3] * alan_deneme_sayilari
iter4 = (np.sin((9) * theta)) * new_matrix[4] * alan_deneme_sayilari
iter5 = (np.sin((11) * theta)) * new_matrix[5] * alan_deneme_sayilari
iter6 = (np.sin((13) * theta)) * new_matrix[6] * alan_deneme_sayilari
iter7 = (np.sin((15) * theta)) * new_matrix[7] * alan_deneme_sayilari
iter8 = (np.sin((17) * theta)) * new_matrix[8] * alan_deneme_sayilari
iter9 = (np.sin((19) * theta)) * new_matrix[9] * alan_deneme_sayilari
iter10 = (np.sin((21) * theta)) * new_matrix[10] * alan_deneme_sayilari

# TAŞIMA KATSAYISI HESABI VE DEĞERLERİN OKUNABİLMESİ İÇİN MATRİSE ATANMASI
CL_grafik = iter0 + iter1 + iter2 + iter3 + iter4 + iter5 + iter6 + iter7 +
iter8 + iter9 + iter10
CL_hesap = (math.pi * AR * new_matrix[0])

rho = (2 * (new_matrix[1] / new_matrix[0])) + (3 * (new_matrix[2] /
new_matrix[1])) + (4 * (new_matrix[3] / new_matrix[2])) + (5 * (new_matrix[4]
/ new_matrix[3])) + (6 * (new_matrix[5] / new_matrix[4])) + (7 *
(new_matrix[6] / new_matrix[5])) + (8 * (new_matrix[7] / new_matrix[6])) +

```

```

(9* (new_matrix[8] / new_matrix[7])) + (10 * (new_matrix[9] / new_matrix[8]))
+ (11 * (new_matrix[10] / new_matrix[9]))

CL_induklenmis_grafik = CL_grafik ** 2 / (math.pi * AR) * rho
CL_induklenmis_hesap = CL_hesap ** 2 / (math.pi * AR) * rho

CL1_dizi = np.append(0, CL_grafik)
CL_induklenmis_dizi = np.append(0, CL_induklenmis_grafik)
y_eksen = [b / 2, k[0], k[1], k[2], k[3], k[4], k[5], k[6], k[7], k[8], k[9],
k[10]]

print("Tasima Katsayisi: ", CL_hesap)
print("Induklenmis Surukleme Katsayisi: ", CL_induklenmis_hesap)

fig = plt.figure("AERODİNAMİK ÖDEVİ [12.GRUP] CL")
plt.plot(y_eksen, CL1_dizi, marker="o")
plt.title("NACA 65(2)-215")
plt.xlabel("Kanat Yarı-Açıklık Konumu (m)")
plt.ylabel("Taşıma Katsayısı")
plt.grid()
plt.show()

fig2 = plt.figure("AERODİNAMİK ÖDEVİ [12.GRUP] CL_induklenmis")
plt.plot(y_eksen, CL_induklenmis_dizi, marker="o")
plt.title("NACA 65(2)-215")
plt.xlabel("Kanat Yarı-Açıklık Konumu (m)")
plt.ylabel("İndüklenmiş Sürüklenme Katsayısı")
plt.grid()
plt.show()

```

3) NACA 652 – 215 profilinde dikdörtgensel üstgörünümlü bir kanat için uygulanmış herhangi bir açıklık oranında taşıma katsayısı ve indüklenmiş sürüklenme katsayısı:

**AR=9,08**

**b = 13.69976642136646 m, S = 20,67 m<sup>2</sup> ve c = 1.4725215484915752 m**

İstasyon sayısı (N)= 11 olursa istasyonların açı aralığı şu şekilde olur:

$$\Delta\theta = \frac{\pi}{2N} = \frac{\pi}{22}$$

olarak bulunur.

10 derece hücum açısı için;

İstasyon theta degerleri: [0.14279967 0.28559933 0.428399 0.57119866 0.71399833 0.856798 0.99959766 1.14239733 1.28519699 1.42799666 1.57079633]

$$\mu = \frac{a_{\infty} \times c}{4 \times b} = 0.16069031013269844 \text{ olarak}$$

bulundu.

$$\sum_{n=1}^N \left( 1 + \frac{n\mu_m}{\sin \theta_m} \right) A_n \sin n\theta_m = \mu_m (\alpha_m - \alpha_{0m})$$

formülü kullanılarak kod yazıldı.

```
mu = alpha_sonsuz * c / (4 * b)
mu_hesaplama_2 = mu * (np.array(alpha_hucum) - alpha_zero_lift)
mu_hesaplama_2_rad = np.deg2rad(mu_hesaplama_2)

teta_grafik = []
for i in range(1, 2 * N + 1, 2):
    teta_grafik_iter = np.sin(i * theta) * (1 + (mu * i) / (np.sin(list(theta))))
    teta_grafik.append(teta_grafik_iter)

denklem = np.asarray(teta_grafik)
denklem_transpoz = np.transpose(denklem)
denklem_is_active = np.linalg.inv(denklem_transpoz)

new_matrix = np.matmul(denklem_is_active, mu_hesaplama_2_rad)
alan_deneme_sayilari = np.divide((4 * b), c)

iter0 = (np.sin((1) * theta)) * new_matrix[0] * alan_deneme_sayilari
iter1 = (np.sin((3) * theta)) * new_matrix[1] * alan_deneme_sayilari
iter2 = (np.sin((5) * theta)) * new_matrix[2] * alan_deneme_sayilari
iter3 = (np.sin((7) * theta)) * new_matrix[3] * alan_deneme_sayilari
iter4 = (np.sin((9) * theta)) * new_matrix[4] * alan_deneme_sayilari
iter5 = (np.sin((11) * theta)) * new_matrix[5] * alan_deneme_sayilari
iter6 = (np.sin((13) * theta)) * new_matrix[6] * alan_deneme_sayilari
iter7 = (np.sin((15) * theta)) * new_matrix[7] * alan_deneme_sayilari
iter8 = (np.sin((17) * theta)) * new_matrix[8] * alan_deneme_sayilari
iter9 = (np.sin((19) * theta)) * new_matrix[9] * alan_deneme_sayilari
iter10 = (np.sin((21) * theta)) * new_matrix[10] * alan_deneme_sayilari
```

11 bilinmeyen ve 11 denklem oluşmuştur.

```
A degerleri: [ 3.30381191e-02  3.93241728e-03  1.50584408e-03  3.65059767e-04  
2.19686948e-04  5.17747319e-05  5.76024961e-05  6.28605144e-06  
2.18072142e-05 -4.29536950e-06  9.79380043e-06]
```

Taşıma Katsayısı= 0.9424341941935923

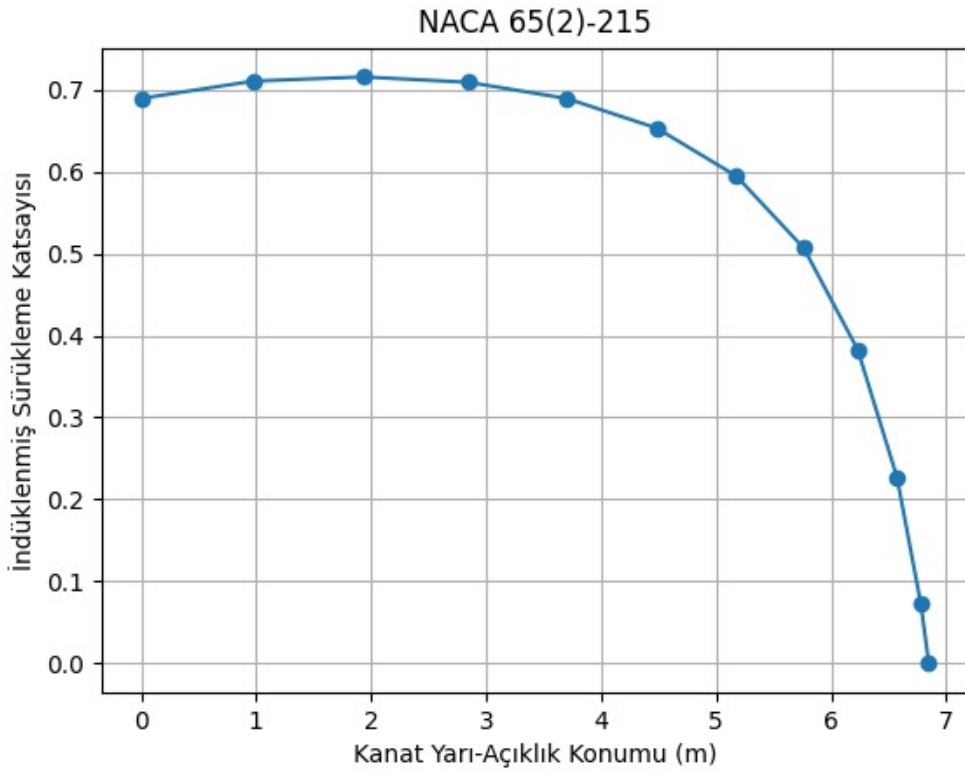
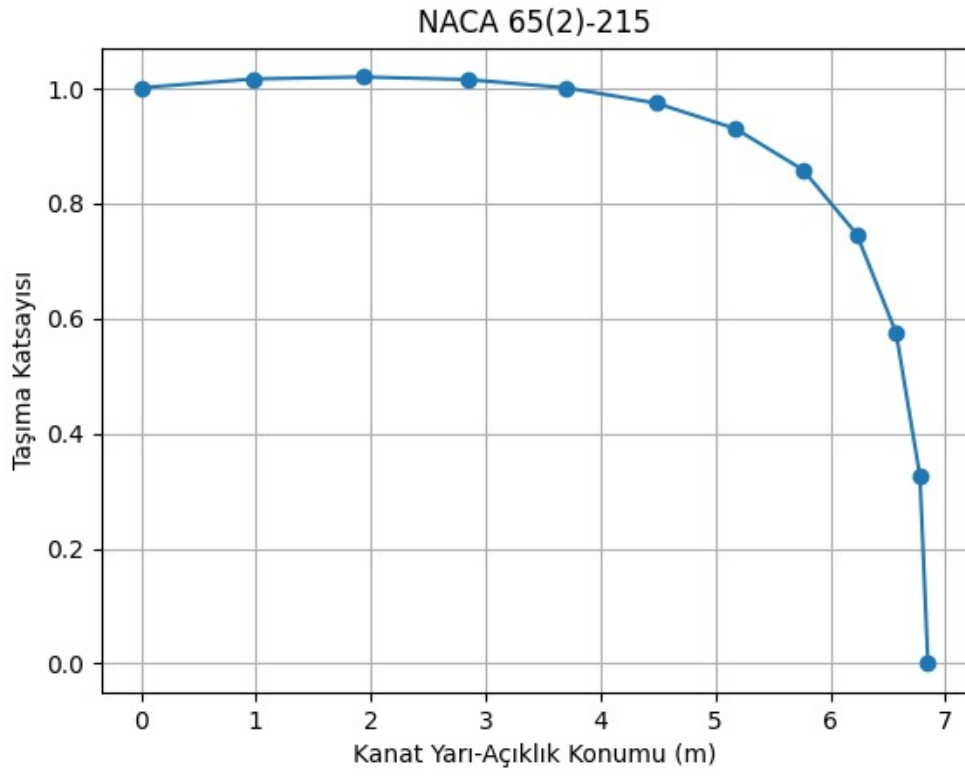
İndüklenmiş sürüklenme katsayısı= 0.6106487646972549 olarak bulundu.

Çeşitli AR değerleri için indüklenmiş sürüklenme katsayıları hesaplandı.

$$C_{D_i} = \frac{C_L^2}{\pi AR} (1 + \delta)$$

Formülü kullanılarak indüklenmiş sürüklenme katsayısı bulundu.

Açıklık oranı arttıkça taşıma katsayısının arttığı ve açıklık oranının fazla artışı ile de indüklenmiş sürüklenme katsayısının da arttığı görüldü.





#### 4) RAPOR

Bu ödev kapsamında, takım çalışmasının ne denli önemli olduğu ve takımdaki bireylerin iş paylaşımındaki tutumunun ödeve olan katkısı anlaşılmıştır. Prandtl Taşıyıcı Çizgi Teorisi hakkında daha detaylı bilgiler edinildi. Uygulamadaki kullanımı hakkında deneyim kazanıldı. Çeşitli oranların ve değerlerin taşıma katsayısı üzerindeki etkisi gözlemlendi. Rho değerinin formüle dahil edilmediği zaman indüklenmiş sürüklenme katsayısının daha az olduğu hesaplanmıştır.

Aynı yüzey alanına sahip bir kanatta açıklık oranı artınca indüklenmiş sürüklenme azalacağı için taşıma artar. Deney sonuçlarına bakılarak da bu çok açık bir şekilde görülebilir. Bunun nedeni kanat ucu vortexlerinin, açıklık oranı artınca iki boyutlu kanatta olduğu gibi taşımaya etkisi azalır.

Planörlerin açıklık oranının bu denli fazla olmasının nedeni de budur.

Yüzey alanı ve açıklık oranı sabit kalmak şartıyla en az indüklenmiş sürüklenme katsayısı eliptik kanat geometrisinde gözlenir. Bunun dışındaki geometrilere indüklenmiş sürüklenme artacaktır.