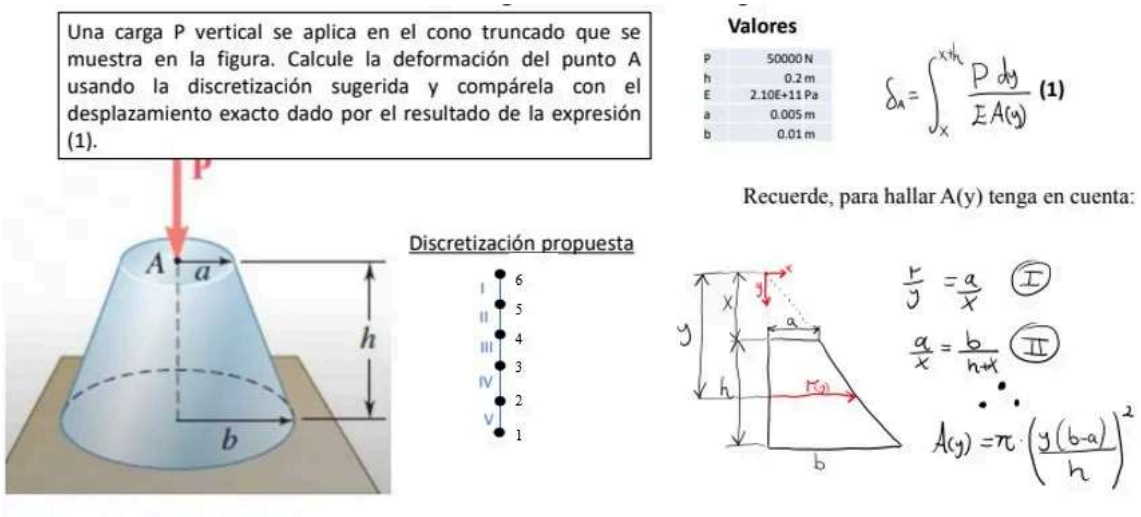
Tarea Cono truncado

Miguel Angel Reales Gaitan Cc. 1007694581



Datos del problema:

- P = 50000 N (carga vertical)
- h = 2.10E+11 Pa (módulo de Young)
- a = 0.005 m (radio superior)
- b = 0.01 m (radio inferior)

```
import numpy as np
from scipy.integrate import quad

# Datos del problema
P = 50000  # N
E = 2.10e11  # Pa
a = 0.005  # m
b = 0.01  # m
h = 0.05  # m

# Discretización con 6 segmentos
n = 6
x = np.linspace(0, h, n+1)
dx = h/n

def A(y):
    """Calcula el área en función de la altura y"""
    return np.pi * ((y*(b-a)/h + a))**2

# Método del trapecio
deformacion_trap = 0
for i in range(n):
    A_promedio = (A(x[i]) + A(x[i+1]))/2
    deformacion_trap += (P * dx)/(E * A_promedio)

# Método de integración numérica
deformacion_quad, _ = quad(lambda y: P/(E*A(y)), 0, h)

print(f"Deformación (trapecio): {deformacion_trap*1000:.6f} mm")
print(f"Deformación (quad): {deformacion_quad*1000:.6f} mm")
```

```
Deformación (trapecio): 0.075183 mm
Deformación (quad): 0.075788 mm
```

Cálculo de errores

```
# Cálculo de errores
error_absoluto = abs(deformacion_quad - deformacion_trap)
error_relativo = (error_absoluto/deformacion_quad) * 100
```

```
print(f"Error absoluto: {error_absoluto*1000:.6f} mm")
print(f"Error relativo: {error_relativo:.4f}%")
```

```
Error absoluto: 0.000605 mm
Error relativo: 0.7980%
```

Deformación punto a

```
import numpy as np
from scipy.integrate import quad

# Datos del problema
P = 50000  # N
E = 2.10e11  # Pa
a = 0.005  # m (radio en punto A)
b = 0.01  # m (radio en base)
h = 0.05  # m (altura total)

def A(y):
    """Calcula el área en función de la altura y"""
    return np.pi * ((y*(b-a)/h + a))**2

# Método del trapecio para punto A
n = 6  # número de segmentos
x = np.linspace(0, h, n+1)  # desde punto A (y=0) hasta base (y=h)
dx = h/n

deformacion_A_trap = 0
for i in range(n):
    x_i = x[i]
    x_i1 = x[i+1]
    A_promedio = (A(x_i) + A(x_i1))/2
    deformacion_A_trap += (P * dx)/(E * A_promedio)

# Método quad para punto A
deformacion_A_quad, _ = quad(lambda y: P/(E*A(y)), 0, h)

# Cálculo de errores
error_absoluto = abs(deformacion_A_quad - deformacion_A_trap)
error_relativo = (error_absoluto/deformacion_A_quad) * 100

print(f"Deformación en punto A (trapecio): {deformacion_A_trap*1000:.6f} mm")
print(f"Deformación en punto A (quad): {deformacion_A_quad*1000:.6f} mm")
print(f"Error absoluto: {error_absoluto*1000:.6f} mm")
print(f"Error relativo: {error_relativo:.4f}%")

# Validación con la fórmula exacta dada en el problema
deformacion_A_exacta = (P*h)/(E*np.pi*a*b)
print(f"\nDeformación exacta según fórmula (1): {deformacion_A_exacta*1000:.6f} mm")
print(f"Error vs fórmula exacta: {abs(deformacion_A_exacta - deformacion_A_quad)*1000:.6f} mm")
```

```
Deformación en punto A (trapecio): 0.075183 mm
Deformación en punto A (quad): 0.075788 mm
Error absoluto: 0.000605 mm
Error relativo: 0.7980%

Deformación exacta según fórmula (1): 0.075788 mm
Error vs fórmula exacta: 0.000000 mm
```

Comparación deformacion total y deformación punto a

```
import numpy as np
from scipy.integrate import quad

# Datos del problema
P = 50000  # N
E = 2.10e11  # Pa
a = 0.005  # m
b = 0.01  # m
h = 0.05  # m

def A(y):
    """Calcula el área en función de la altura y"""
    return np.pi * ((y*(b-a)/h + a))**2

# 1. Deformación usando método del trapecio
n = 6  # número de segmentos
x = np.linspace(0, h, n+1)
```

```
dx = h/n

deformacion_trap = 0
for i in range(n):
    A_promedio = (A(x[i]) + A(x[i+1]))/2
    deformacion_trap += (P * dx)/(E * A_promedio)

# 2. Deformación en punto A usando la fórmula exacta
deformacion_A_exacta = (P*h)/(E*np.pi*a*b)

print("\nCOMPARACIÓN DE RESULTADOS:")
print(f"Deformación por trapecio: {deformacion_trap*1000:.6f} mm")
print(f"Deformación en punto A (exacta): {deformacion_A_exacta*1000:.6f} mm")

# Cálculo del error
error_absoluto = abs(deformacion_A_exacta - deformacion_trap)
error_relativo = (error_absoluto/deformacion_A_exacta) * 100

print(f"\nDiferencia absoluta: {error_absoluto*1000:.6f} mm")
print(f"Error relativo: {error_relativo:.2f}%")
```

```
COMPARACIÓN DE RESULTADOS:
Deformación por trapecio: 0.075183 mm
Deformación en punto A (exacta): 0.075788 mm

Diferencia absoluta: 0.000605 mm
Error relativo: 0.80%
```