***Space, rocket and aerospace technologies in science and programming.***

**Author, MSc: Adrian Szklarski**: 11.2023

**Subject:** A thick-walled fuel tank is installed in a ballistic missile. What can be the highest internal pressure p[MPa] so that the reduced stresses on the cylindrical part do not exceed $N=100\,MPa$ (for example) . Calculate the change in radius a and b at this pressure $a=67\,cm, b=77\,cm, E=2*10^5\,MPa, n=0.3$

**Temat:** W pocisku balistycznym zainstalowano grubościenny zbiornik na paliwo. Jakie może być największe ciśnienie wewnętrzne p[MPa] aby naprężenia zredukowane na części walcowej nie przekroczyły $N=500\,MPa$ . Obliczyć zmianę promienia *a* i *b* przy tym ciśnieniu $a=20\,cm, b=30\,cm, E=2*10^5\,MPa, n=0.3$

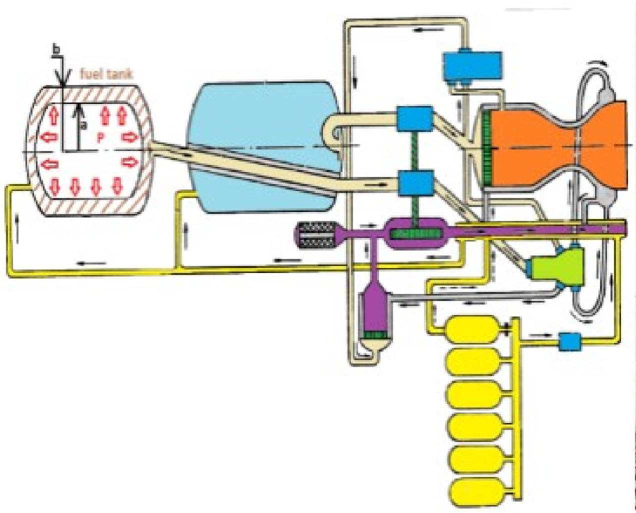


**Fig. 1**: Balisitic Missile R-17 Elbrus [1]

Contact me:

Programming language:
***Python***

No.3, 3/2023

---

**Description of the problem**: A thick-walled fuel tank is installed in a ballistic missile. What can be the highest internal pressure p[MPa] so that the reduced stresses on the cylindrical part do not exceed $N=100\,MPa$ (for example). Calculate the change in radius a and b at this pressure

$$a=67\,cm\,,b=77\,cm\,,E=2*10^5\,MPa\,,n=0.3$$

Tensions: $\sigma_r=\frac{p a^2}{b^2-a^2}(1-\frac{b^2}{r^2})$ ,

$$\sigma_\Theta=\frac{p a^2}{b^2-a^2}(1+\frac{b^2}{r^2}) \quad , \quad \sigma_z=\frac{p a^2}{b^2-a^2}$$

Reduced tension at the radius:

$$\sigma_{zred}=\sqrt{0.5((\sigma_r-\sigma_\Theta)^2+(\sigma_r-\sigma_z)^2+(\sigma_\Theta-\sigma_z)^2)}$$

Condition for maximum reduced tensions:

Deformations: $\varepsilon_\Theta=\frac{1}{E}(\sigma_\Theta-n\sigma_r)-\frac{n}{E}\sigma_z$ ,

$\varepsilon_r=\frac{1}{E}(\sigma_r-n\sigma_\Theta)-\frac{n}{E}\sigma_z$ , so:

$$\varepsilon_\Theta=\frac{u}{r}=C_1+\frac{C_2}{r^2} \quad , \quad \varepsilon_r=\frac{du}{dr}=C_1-\frac{C_2}{r^2} \quad \rightarrow$$

$r=a\,,\sigma_r(a)=-p\,,r=b\,,\sigma_r(b)=0$ so:

$\sigma_r=\frac{E}{1-n^2}[C_1(1+n)-\frac{C_2}{a^2}(1-n)]$ hence we determine $C_1,C_2,u=C_1 r+C_2/r\,,r_a,r_b$

---

**Opis problemu**: W pocisku balistycznym zainstalowano grubościenny zbiornik na paliwo. Jakie może być największe ciśnienie wewnętrzne p[MPa] aby naprężenia zredukowane na części walcowej nie przekroczyły $N=500\,MPa$ . Obliczyć zmianę promienia $a$ i $b$ przy tym ciśnieniu

$$a=20\,cm\,,b=30\,cm\,,E=2*10^5\,MPa\,,n=0.3$$

Naprężenia: $\sigma_r=\frac{p a^2}{b^2-a^2}(1-\frac{b^2}{r^2})$ ,

$$\sigma_\Theta=\frac{p a^2}{b^2-a^2}(1+\frac{b^2}{r^2}) \quad , \quad \sigma_z=\frac{p a^2}{b^2-a^2}$$

Naprężenie zredukowane na promieniu:

$$\sigma_{zred}=\sqrt{0.5((\sigma_r-\sigma_\Theta)^2+(\sigma_r-\sigma_z)^2+(\sigma_\Theta-\sigma_z)^2)}$$

Warunek na maksymalne naprężenia zredukowane: $\sigma_{zred}<N$

Odkształcenia: $\varepsilon_\Theta=\frac{1}{E}(\sigma_\Theta-n\sigma_r)-\frac{n}{E}\sigma_z$ ,

$\varepsilon_r=\frac{1}{E}(\sigma_r-n\sigma_\Theta)-\frac{n}{E}\sigma_z$ , dalej:

$$\varepsilon_\Theta=\frac{u}{r}=C_1+\frac{C_2}{r^2} \quad , \quad \varepsilon_r=\frac{du}{dr}=C_1-\frac{C_2}{r^2} \quad \rightarrow$$

$r=a\,,\sigma_r(a)=-p\,,r=b\,,\sigma_r(b)=0$ czyli:

$\sigma_r=\frac{E}{1-n^2}[C_1(1+n)-\frac{C_2}{a^2}(1-n)]$ stąd określamy $C_1,C_2,u=C_1 r+C_2/r\,,u_a,u_b$

---

*Starting method: Self-ignition of the starting fuel and oxidizer*
*Fuel supply: Turbopump assembly running from the gas generator*
*Thrust: 13,310-13,380 kg (various sources)*
*Specific Impulse (s.l.): 226 sec*
*Specific Impulse (vac): 258 sec*
*Fuel consumption: 57.83 kg/sec*
*Length: 1,490 mm*
*Maximum diameter: 770 mm*
*The diameter of the combustion chamber: 380 mm*
*Nozzle throat diameter: 124.5 mm*
*Nozzle exit diameter: 400 mm*
*The pressure in the combustion chamber: 69.4 kg /cm2*
*The pressure at the nozzle exit: 0.827 kg /cm2*
*Fuel: Kerosene mixture of TM-185 (B6 OCT-02-43-84).*
*Weight: 822 kg kg (20 degrees C)*
*Polimerdistillate: 56 +/- 1.5%*
*Light Oil Pyrolyse: 40 + /- 1.0% (to increase the density and resistance to oxidation)*
*Trikrezol: 4 +/- 0.5% (prevents crystallization of water)*
*Oxidizer: AK-27I ("Melange" Standard V18112-72)*
*Weight: 2,919 kg (20 degrees C)*
*Concentrated Nitric Acid: 69.8 - 70.2%*
*Nitrogen Tetroxide: 24 - 28%*
*Water: 1.3 - 2%*
*Aluminum salts (not more than 0.01% )*
*Iodine: 0.12 - 0.16% (inhibitor)*
*Density: 1.596 - 1.613*
*Starting fuel: TG-02 "Samin" (GOST V17147-71)*
*Weight: 30-35 kg + 1 liter is filled into the rocket just before launch.*
*Isomeric Xylidines: 50 +/- 2%*
*Triethylamine Technical: 50 +/- 2%*
*Water: 0.4% ( 0.835-0.855 kg)*
*https://www.b14643.de/*

**The result is**: (0.44, 0.42)

```python
"""   Adrian Szklarski 11.2023  """

import math


class Prom:

    def __init__(self, data):
        self.data = data

    def get_stresses(self):
        Simga_r = (math.pow(self.data[0], 2) / (math.pow(self.data[1], 2) - math.pow(self.data[0], 2))) * \
                  (1 - (math.pow(self.data[1], 2) / math.pow(self.data[0], 2)))
        Simga_Theta = (math.pow(self.data[0], 2) / (math.pow(self.data[1], 2) - math.pow(self.data[0], 2))) * \
                  (1 + (math.pow(self.data[1], 2) / math.pow(self.data[0], 2)))
        Simga_z = (math.pow(self.data[0], 2) / (math.pow(self.data[1], 2) - math.pow(self.data[0], 2)))
        Simga_derated = round(math.sqrt(0.5 * self.data[0]), 2)

        if Simga_derated < data[4]:
            p = self.data[4] / Simga_derated
            self.Simga_r = round(Simga_r * p, 2)
            self.Simga_Theta = round(Simga_Theta * p, 2)
            self.Simga_z = round(Simga_z * p, 2)
        else:
            exit(0)
            return f'The condition for reduced stresses is not met'
        return self.Simga_r, self.Simga_Theta, self.Simga_z

    def get_deformations(self):
        self.E_r = round((1 / self.data[2]) * (self.Simga_r - (self.data[3] * self.Simga_Theta)) -
                         (self.data[3] * self.Simga_z / self.data[2]), 5)
        self.E_Theta = round((1 / self.data[2]) * (self.Simga_Theta - (self.data[3] * self.Simga_r)) -
                         (self.data[3] * self.Simga_z / self.data[2]), 5)
        C2 = round((self.Simga_r / self.data[2]) * (1 + self.data[3]) * (
                (math.pow(self.data[0], 2) * math.pow(self.data[1], 2)) / (
                math.pow(self.data[0], 2) - math.pow(self.data[1], 2))) * 100, 2)
        C1 = round((C2 / math.pow(self.data[1] * 10, 2)) * ((1 - self.data[3]) / (1 + self.data[3])), 5)
        return C1, C2

    def get_radius_change(self):
        C1, C2 = self.get_deformations()
        self.delta_ra = round((C1 * self.data[0] * 10) + C2 / (self.data[0] * 10), 2)
        self.delta_rb = round((C1 * self.data[1] * 10) + C2 / (self.data[1] * 10), 2)
        return self.delta_ra, self.delta_rb

    def __str__(self):
        return f'Changing the radius: ra = {self.delta_ra} [mm],  rb = {self.delta_rb} [mm]'


if __name__ == '__main__':
    data = [
        'a=', 67.0, 'cm',
        'b=', 77.0, 'cm',
        'E=', 2E5, 'MPa',
        'n=', 0.3,
        'N=', 100.0, 'MPa'
    ]
    hear = Prom(list(filter(lambda x: type(x) is float, data)))
    hear.get_stresses(), hear.get_deformations(), hear.get_radius_change()
    print(hear)
    print(hear.get_radius_change())
```