3.3.15

Shivam Sawarkar
AI25BTECH11031

September 30, 2025

Question

Find the equation of the set of points P the sum of whose distances from $A(4, 0, 0)$ and $B(-4, 0, 0)$ is equal to 10.

Solution

We want the locus of points $\mathbf{p} \in \mathbb{R}^3$ such that

$$\|\mathbf{p} - \mathbf{A}\| + \|\mathbf{p} - \mathbf{B}\| = 10, \tag{1}$$

where

$$\mathbf{A} = \begin{pmatrix} 4 \\ 0 \\ 0 \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} -4 \\ 0 \\ 0 \end{pmatrix}. \tag{2}$$

Step 1 - Decomposition of p:
Define the unit vector along the foci axis:

$$\mathbf{e} = \frac{\mathbf{A} - \mathbf{B}}{\|\mathbf{A} - \mathbf{B}\|} = \frac{\begin{pmatrix} 8 \\ 0 \\ 0 \end{pmatrix}}{8} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}. \tag{3}$$

Solution

Decompose $\mathbf{p}$ into parallel and perpendicular components:

$$\mathbf{p} = (\mathbf{e}^\top \mathbf{p})\,\mathbf{e} + (I - \mathbf{e}\mathbf{e}^\top)\mathbf{p}. \tag{4}$$

Let:

$$\alpha := \mathbf{e}^\top \mathbf{p}, \quad R := I - \mathbf{e}\mathbf{e}^\top. \tag{5}$$

Then the perpendicular squared component is:

$$s := \|R\mathbf{p}\|^2. \tag{6}$$

Step 2 - Distances to the foci:

$$\|\mathbf{p} - \mathbf{A}\| = \sqrt{(\alpha - 4)^2 + s}, \quad \|\mathbf{p} - \mathbf{B}\| = \sqrt{(\alpha + 4)^2 + s}. \tag{7}$$

The condition becomes:

$$\sqrt{(\alpha - 4)^2 + s} + \sqrt{(\alpha + 4)^2 + s} = 10. \tag{8}$$

Solution

Step 3 - Eliminate square roots:
Square both sides and rearrange to eliminate the radicals. After algebraic manipulation, we obtain:

$$-36\,\alpha^2 - 100\,s + 900 = 0. \tag{9}$$

The equation becomes:

$$-36\,\mathbf{p}^\top(\mathbf{e}\mathbf{e}^\top)\mathbf{p} - 100\,\mathbf{p}^\top R\mathbf{p} + 900 = 0. \tag{10}$$

$$\mathbf{p}^\top\left(-36\,\mathbf{e}\mathbf{e}^\top - 100\,R\right)\mathbf{p} + 900 = 0. \tag{11}$$

Step 4 - Simplify using $R = I - \mathbf{e}\mathbf{e}^\top$:

$$-36\,\mathbf{e}\mathbf{e}^\top - 100(I - \mathbf{e}\mathbf{e}^\top) = -100I + 64\,\mathbf{e}\mathbf{e}^\top. \tag{12}$$

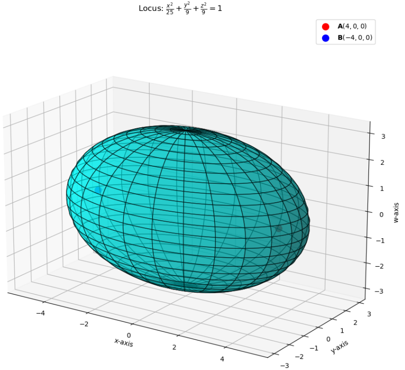
Thus:

$$\mathbf{p}^\top \left(I - \frac{64}{100}\mathbf{e}\mathbf{e}^\top \right) \mathbf{p} = 9. \tag{13}$$

$$\boxed{\mathbf{p}^\top \begin{pmatrix} \frac{1}{25} & 0 & 0 \\ 0 & \frac{1}{9} & 0 \\ 0 & 0 & \frac{1}{9} \end{pmatrix} \mathbf{p} = 1,} \tag{14}$$

which is the equation of a prolate spheroid with semi-axes $5, 3, 3$.

Plot



Locus: $\frac{x^2}{25}+\frac{y^2}{9}+\frac{z^2}{9}=1$
$\mathbf{A}(4,0,0)$
$\mathbf{B}(-4,0,0)$
x-axis
y-axis
w-axis

Figure:

C Code

```
#ifndef LOCUS_H
#define LOCUS_H

#include <stdio.h>
#include <math.h>

void standard_form(double A[3], double B[3], double constant_s
                   double *a2, double *b2, double *c2) {
    double dist_AB = sqrt(pow(A[0]-B[0],2) + pow(A[1]-B[1],2)
    double a = constant_sum / 2.0;
    double c = dist_AB / 2.0;
    double b = sqrt(a*a - c*c);
    *a2 = a*a;
    *b2 = b*b;
    *c2 = b*b; // spheroid symmetry
}
#endif
```

C Code

```
#include <stdio.h>
#include <math.h>
#include "solution.h"

int main() {
    double A[3], B[3];
    double constant_sum;
    double a2, b2, c2;

    printf("Enter coordinates of point A (Ax Ay Az): ");
    scanf("%lf %lf %lf", &A[0], &A[1], &A[2]);

    printf("Enter coordinates of point B (Bx By Bz): ");
    scanf("%lf %lf %lf", &B[0], &B[1], &B[2]);
```

C Code

```
    printf("Enter the constant sum of distances: ");
    scanf("%lf", &constant_sum);

    standard_form(A, B, constant_sum, &a2, &b2, &c2);

    printf("\nStandard form of the locus equation:\n");
    printf("x^2/(%.6lf) + y^2/(%.6lf) + z^2/(%.6lf) = 1\n", a2

    printf("\nWhere:\n");
    printf("a = %.6lf, b = %.6lf, c = %.6lf\n", sqrt(a2), sqrt

    return 0;
}
```

Python Code

```
import numpy as np

def standard_form(A, B, constant_sum):
    # Distance between foci
    dist_AB = np.linalg.norm(np.array(A) - np.array(B))

    # Semi-major axis
    a = constant_sum / 2.0
    # Focal length
    c = dist_AB / 2.0
    # Semi-minor axis
    b = np.sqrt(a*a - c*c)

    return a*a, b*b, b*b  # a², b², c²
```

```
if __name__ == "__main__":
    A = list(map(float, input("Enter coordinates of point A (A
    B = list(map(float, input("Enter coordinates of point B (B
    constant_sum = float(input("Enter the constant sum of dist

    a2, b2, c2 = standard_form(A, B, constant_sum)

    print("\nStandard form of the locus equation:")
    print(f"x^2/({a2:.6f}) + y^2/({b2:.6f}) + z^2/({c2:.6f}) =
    print("\nWhere:")
    print(f"a = {np.sqrt(a2):.6f}, b = {np.sqrt(b2):.6f}, c =
```

Python + C Code

```
import ctypes
import numpy as np

# Load the shared library
locus = ctypes.CDLL("./solution.so")

# Define argument types
locus.standard_form.argtypes = [
    ctypes.POINTER(ctypes.c_double), ctypes.POINTER(ctypes.c_
    double),
    ctypes.c_double,
    ctypes.POINTER(ctypes.c_double), ctypes.POINTER(ctypes.c_
    double), ctypes.POINTER(ctypes.c_double)
]
```

Python + C Code

```
def standard_form(A, B, constant_sum):
    A_arr = (ctypes.c_double * 3)(*A)
    B_arr = (ctypes.c_double * 3)(*B)
    a2 = ctypes.c_double()
    b2 = ctypes.c_double()
    c2 = ctypes.c_double()
    locus.standard_form(A_arr, B_arr, constant_sum,
    ctypes.byref(a2), ctypes.byref(b2), ctypes.byref(c2))
    return a2.value, b2.value, c2.value

if __name__ == "__main__":
    A = list(map(float, input("Enter coordinates of point
    A (Ax Ay Az): ").split()))
    B = list(map(float, input("Enter coordinates of point
    B (Bx By Bz): ").split()))
    constant_sum = float(input("Enter the constant sum of
    distances: "))
```

```
a2, b2, c2 = standard_form(A, B, constant_sum)
print("\nStandard form of the locus equation:")
print(f"x^2/({a2:.6f}) + y^2/({b2:.6f}) + z^2/({c2:.6f})
= 1")
print("\nWhere:")
print(f"a = {np.sqrt(a2):.6f}, b = {np.sqrt(b2):.6f},
c = {np.sqrt(c2):.6f}")
```