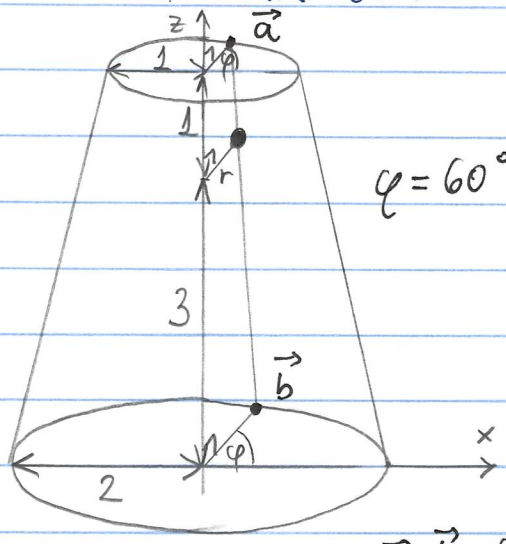
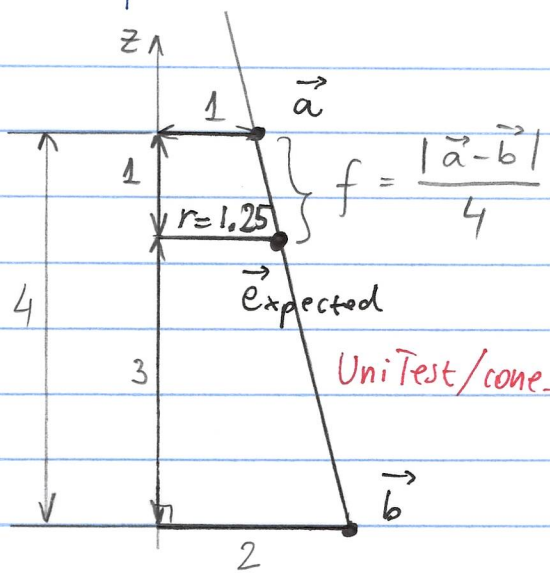
linear_Vector3d-fit

pmh_2019_1031

function and unit test



$\varphi = 60°$



UniTest/cone_endpoints.inc

$\vec{a}, \vec{b}$ from Cone::get_endpoints()

$a_x = 1 \cdot \cos\varphi = \frac{1}{2}$ $\quad b_x = 2 \cdot \cos\varphi = 1$

$a_y = 1 \cdot \sin\varphi = \sqrt{3}/2$ $\quad b_y = 2 \cdot \sin\varphi = \sqrt{3}$

$a_z = 4$ $\quad b_z = 0$

$\vec{e}_x = r \cdot \cos\varphi = 1.25 \times \frac{1}{2} = 0.625$

$\vec{e}_y = r \cdot \sin\varphi = 1.25 \times \sqrt{3}/2 = 0.625 \times \sqrt{3}$

$\vec{e}_z = 3$

Cone::intercept() uses linear_Vector3d_fit() with

f = Cone::MINIMUM_DISTANCE

Cone::contains, Vector3d::is_between from pmh_2015_0508

is amended to allow for non-colinear vectors by requiring that the condition: $(\vec{w} - \vec{A}) \cdot (\vec{w} - \vec{B}) \leq 0$ be satisfied by all individual components, i.e.,

$(w_x - A_x)(w_x - B_x) \leq 0$ and

$(w_y - A_y)(w_y - B_y) \leq 0$ and

$(w_z - A_z)(w_z - B_z) \leq 0$

The developments on page 1 were motivated by the need to obtain a unique raytracing solution in case a Ray is to exit a Cone-defined Zone through a Node shared by two Cone-shaped Faces.