Robust Intersections under Floating Point

Yuting Wang, Chenfanfu Jiang, Craig Schroeder, and Joseph Teran

1 Overview

Our strategy for handling intersections robustly in floating point is to register degeneracies with tolerances in a hierarchical manner before registering the top level intersection primitives in order from most degenerate (vertex-vertex) to least, followed by the top level primitives (edge-edge and triangle-vertex in 2D, triangle-edge and tetrahedron-vertex in 3D). Intersection pairs are stored in hashtables for efficient lookup. When processing each intersection pair, we first check these hashtables to see if a degenerate version of pair has already been registered (e.g., when checking an edge-edge pair, a vertex of one of these edges may already be registered as intersecting the other edge or one if its endpoints). If such a degeneracy is found, processing on the pair terminates and no intersection is registered. Otherwise, the pair is checked to see if it actually intersects. Handling degeneracies in this way allows us to assume that no degeneracy exists, which in turn makes it possible to guarantee that the current pair can be checked robustly. This is the key to guaranteeing robustness.

2 Floating point error

Let $\mathcal{F}[f(x)]$ indicate that the computation of $f(x)$ is being carried out under floating point computation. In general, we will be deriving bounds for floating point error $\mathcal{E}[f(x)] = |\mathcal{F}[f(x)] - f(x)|$ or relative error $\mathcal{R}[f(x)]$, where $\mathcal{E}[f(x)] = |f(x)|\mathcal{R}[f(x)]$ for various operations. In the case of vector quantities, $\mathcal{R}[u] = \max(\mathcal{R}[u_x], \mathcal{R}[u_y])$ for 2D and $\mathcal{R}[u] = \max(\mathcal{R}[u_x], \mathcal{R}[u_y], \mathcal{R}[u_z])$ for 3D. All coordinates are taken to be exact. Let $\epsilon < 10^{-5}$ be the machine epsilon.

For the purposes of computing floating point error we store for each quantity

- s: A string representing how the quantity is computed
- c, p: The maximum magnitude the quantity could have, in the form cL^p, where L is the maximum edge length of the bounding box surrounding each primitive pair.
- n: A flag indicating that the quantity is guaranteed to be nonnegative
- a, b: relative floating point error, of the form $1 + a\epsilon + b\epsilon^2$.
- r: A flag indicating that the floating point error is relative to the quantity being computed. Otherwise, it is relative to cL^p
- v: A flag indicating that the floating point error estimate is valid. This is used in cases where nothing useful can be said about the floating point error, such as when dividing by a quantity that cannot be bounded from zero.

The state is updated using a set of rules for each operation that is used. A result is invalid if any of its inputs are, and s is updated. If all inputs are valid and a rule from below applies, the result is valid. If more than one rule applies, the more restricted rule is used. The rules are applied automatically by instrumenting the code. Note that no upper bound can, in general, be derived for the case of division since no lower bound for the denominator is available. Since we only use division for normalizing vectors, the choice $c = 1$ will always be appropriate. This special case is noted in the table with a "*".

We also assume there is no underflow or overflow in all operations. Underflow can be avoided by shifting the meshes when computing the intersections so their coordinates are all sufficiently bigger than zero. No overflow is a reasonable assumption for simulations that do not blow up.

ID	Rule	Restrictions	n	r
A	$x = -x_1$	none	0	r_1
B	$x = x_1 + x_2$	$p_1 = p_2$	n_1 and n_2	0
C	$x = x_1 + x_2$	n_1 and n_2 and r_1 and r_2 and $p_1 = p_2$	1	1
D	$x = x_1 - x_2$	$p_1 = p_2$	0	0
E	$x = x_1 - x_2$	$p_1 = p_2$ and $a_1 = b_1 = a_2 = b_2 = 0$	0	1
F	$x = x_1 x_2$	none	$s_1 = s_2$ or (n_1 and n_2)	r_1 and r_2
G	$x = x_1/x_2$	r_1 and r_2	n_1 and n_2	1
H	$x = \sqrt{x_1}$	n_1 and r_1	1	r_1
I	$x = \lvert x_1 \rvert$	none	1	r_1
J	$x = \#$	none	$\# \geq 0$	1
K	$x =$ coord	none	0	1

ID	Rule	c	p	a	b
A	$x = -x_1$	c_1	p_1	a_1	b_1
B	$x = x_1 + x_2$	$c_1 + c_2$	p_1	$1 + \frac{c_1 a_1 + c_2 a_2}{c_1 + c_2}$	$\frac{1 + c_1(a_1 + b_1) + c_2(a_2 + b_2)}{c_1 + c_2}$
C	$x = x_1 + x_2$	$c_1 + c_2$	p_1	$1 + \max(a_1, a_2)$	$1 + \max(a_1 + b_1, a_2 + b_2)$
D	$x = x_1 - x_2$	$c_1 + c_2$	p_1	$1 + \frac{c_1 a_1 + c_2 a_2}{c_1 + c_2}$	$\frac{1 + c_1(a_1 + b_1) + c_2(a_2 + b_2)}{c_1 + c_2}$
E	$x = x_1 - x_2$	$c_1 + c_2$	p_1	1	0
F	$x = x_1 x_2$	$c_1 c_2$	$p_1 + p_2$	$1 + a_1 + a_2$	$(1 + a_1)(1 + a_2) + b_1 + b_2$
G	$x = x_1/x_2$	1^*	$p_1 - p_2$	$1 + a_1 + a_2$	$(1 + a_1)(1 + a_2) + b_1 + b_2 + a_2^2$
H	$x = \sqrt{x_1}$	$\sqrt{c_1}$	$p_1/2$	$1 + a_1/2$	$1 + b_1/2 + \lvert a_1/2 - a_1^2/8 \rvert$
I	$x = \lvert x_1 \rvert$	c_1	p_1	a_1	b_1
J	$x = \#$	$\lvert \# \rvert$	0	0	0
K	$x =$ coord	$1/2$	1	0	0

3 Tolerance model

Our goal with this intersection framework is to make robustness guarantees while operating entirely within the limitations of floating point arithmetic. To do this, we use a carefully chosen set of tolerances. When we perform tolerance checks, we do so with corresponding exact bounds on either side. The tolerance checks that we use take the general form $g(\tau) - f(x) \geq 0$, where x represent the input coordinates and f and g are functions, with g strictly monotonically increasing over a suitable range surrounding τ. Since this check will be performed in code under floating point arithmetic, we need a way of determining what we can guarantee based on the outcome of this check. We do this by introducing two more tolerances, τ_- and τ_+ such that $\tau_- < \tau < \tau_+$ and

$$g(\tau_-) - f(x) \geq 0 \implies \mathcal{F}[g(\tau) - f(x)] \geq 0 \implies g(\tau_+) - f(x) \geq 0. \tag{1}$$

Note that the tolerance comparisons against τ_- and τ_+ are in exact arithmetic. These comparisons do not occur in code but rather are used for proving correctness. Only the middle comparison occurs in floating point. This model allows us to isolate the uncertainties of floating point from other aspects of the analysis. For each such tolerance τ, the bounding tolerances τ_- and τ_+ must be chosen so that (1) holds.

Note that if $\mathcal{E}[g(\tau) - f(x)] < |g(\tau) - f(x)|$, then the sign of $\mathcal{F}[g(\tau) - f(x)]$ can be determined unambiguously and agrees with the corresponding exact comparison. In this case, (1) holds trivially. Consider instead the choices x for which $\mathcal{E}[g(\tau) - f(x)] \geq |g(\tau) - f(x)|$. For these x, we require the stricter condition

$$g(\tau_-) - f(x) \leq \mathcal{F}[g(\tau) - f(x)] \leq g(\tau_+) - f(x), \tag{2}$$

Name	T_-	T	T_+	$\mathcal{R}[T]$	description
σ	$6.0\sqrt{\epsilon}L$	$6.5\sqrt{\epsilon}L$	$7.0\sqrt{\epsilon}L$	4ϵ	vertex-vertex
τ	$4.0\sqrt{\epsilon}L$	$4.5\sqrt{\epsilon}L$	$5.0\sqrt{\epsilon}L$	4ϵ	edge-vertex
$\hat{\sigma}$	$5.0\sqrt{\epsilon}L$	$5.5\sqrt{\epsilon}L$	$6.0\sqrt{\epsilon}L$	4ϵ	vertex-vertex degeneracy rejection
κ	$10\epsilon L^2$	$21\epsilon L^2$	$32\epsilon L^2$	4ϵ	edge-edge area bound

Figure 1: Tolerances for 2D.

from which τ_- and τ_+ can be computed given a bound on $\mathcal{E}[g(\tau) - f(x)]$. Note that (2) implies (1).

The other type of comparison that we require is a simple comparison against zero, $f(x) > 0$. To have any chance at robustness, we must guarantee that we always get these comparisons right. The way we do that depends on the extent to which we can bound $f(x)$ from zero and the way in which $f(x) > 0$ is used. The possibilities that arise are

- $|f(x)| > \tau_-$ is guaranteed. We ensure $\mathcal{E}[f(x)] < \tau_-$ and perform the test with $\mathcal{F}[f(x)] > 0$.
- If an intersection exists, then $|f(x)| > \tau_+$. In this case we ensure $\mathcal{E}[f(x)] < \tau_-$ and perform the tests $\mathcal{F}[\tau - |f(x)|] \geq 0$ and $\mathcal{F}[f(x)] > 0$. The test $\mathcal{F}[\tau - |f(x)|] \geq 0$ implies no intersection, since $\mathcal{F}[\tau - |f(x)|] \geq 0 \implies |f(x)| \leq \tau_+$. On the other hand, $\mathcal{F}[\tau - |f(x)|] < 0 \implies |f(x)| > \tau_-$, so that $f(x) > 0$ is computed correctly.

4 2D

Before beginning our treatment of the individual cases, we list each tolerances (T) with their corresponding bounds (T_- and T_+) in Figure 1 for reference.

4.1 Tolerance Computation

We compute the maximum bounding box edge length for each primitive in the simulation mesh L_0 and for each primitive on the cutting mesh L_1. We will process any two primitives if their bounding boxes are within distance σ_+ of one another. Note that σ_+ is the maximum of the tolerances. The maximum coordinate difference that can be obtained (in exact arithmetic) is $\hat{L} = L_0 + L_1 + \sigma_+$. Using $\hat{L}$ as the upper bound, $\sigma_+ = 8\sqrt{\epsilon}\hat{L}$. Thus, we can say $\hat{L} = L_0 + L_1 + 8\sqrt{\epsilon}\hat{L}$ or $\hat{L} = \frac{L_0+L_1}{1-8\sqrt{\epsilon}}$. Further, we know that $\mathcal{F}[\hat{L}] > (1-5\epsilon)\hat{L}$. Let $k = 1 + 5\epsilon$. To account for floating point error, we will instead define $L = k\frac{L_0+L_1}{1-8\sqrt{\epsilon}}$ (which corresponds to the definition $\sigma_+ = 8\sqrt{\epsilon}L$ we will eventually use). With this definition, $\mathcal{F}[L] > \hat{L}$. Then, L, even though it is computed under floating point, is an upper bound on the exact difference between coordinates. The relative errors in the tolerances are shown in Figure 1. Pseudocode for computing tolerances is in Algorithm 1.

Algorithm 1 Computing tolerances in 2D

function COMPUTE_TOLERANCES(A, B)
  $L_a \leftarrow$ maximum bound box edge length of mesh A
  $L_b \leftarrow$ maximum bound box edge length of mesh B
  $s \leftarrow \sqrt{\epsilon}$
  $L \leftarrow \frac{1+5\epsilon}{1-7s}(L_a + L_b)$
  $t = sL$
  $\sigma = 6.5t$
  $\tau = 4.5t$
  $\hat{\sigma} = 5.5t$
  $\kappa = 21\epsilon L^2$
end function

4.2 Vertex-Vertex

Let A and B be two vertex locations. If

$$d^2 = \|A - B\|^2 \leq \sigma^2 \tag{3}$$

we register a vertex-vertex intersection. In the cases that follow, we may now assume that $\|A - B\| > \sigma_-$ in exact arithmetic. Pseudocode for this case is given in Algorithm 2.

Algorithm 2 Vertex Vertex 2D

```
function VERTEX_VERTEX(A, B)
    if ||A − B||² ≤ σ² then
        return TRUE
    else
        return FALSE
    end if
end function
```

Floating point: $\mathcal{R}[d^2] \leq 5\epsilon$. $\mathcal{E}[d^2] \leq 9\epsilon L^2$.

Tolerance constraints:

- $\min(\sigma_+ - \sigma, \sigma - \sigma_-) > 8\epsilon\sigma$.

4.3 Edge-Vertex

Let AB and P be an edge and a vertex. The check we need to perform is

$$d = \frac{|(A - B) \times (P - A)|}{\|A - B\|} \leq \tau \qquad 0 < a < 1, \tag{4}$$

where a is the interpolation weight. Let

$$\hat{u} = A - B \quad m = \|\hat{u}\| \quad u = \frac{\hat{u}}{m} \quad w = P - A \quad \hat{a} = u \cdot w \quad \bar{a} = m - \hat{a} \quad a = \frac{\hat{a}}{m} \quad d = |u \times w|. \tag{5}$$

Let us require that $\tau_+ < \frac{\sqrt{3}}{2}\sigma_-$. Treating vertex-vertex degeneracies guarantees $\|P - A\| > \sigma_-$ and $\|P - B\| > \sigma_-$. If in addition $m \leq \hat{\sigma}_+ = \sigma_-$, we conclude $d > \tau_+$ (See Figure 2(a)). Thus, step 7 of Algorithm 3 will not reject a valid intersection. This additional check allows us to assume $m > \hat{\sigma}_-$ in later stages, which in turn allows us to protect the division in step 10. Noting that $\hat{a}$ and $\bar{a}$ are signed distances to A and B along the segment AB, we also have $|\hat{a}| > \sqrt{\sigma_-^2 - \tau_+^2}$ and $|\bar{a}| > \sqrt{\sigma_-^2 - \tau_+^2}$ (See Figure 2(b)). We use these inequalities to protect the comparisons in step 17 of Algorithm 3.

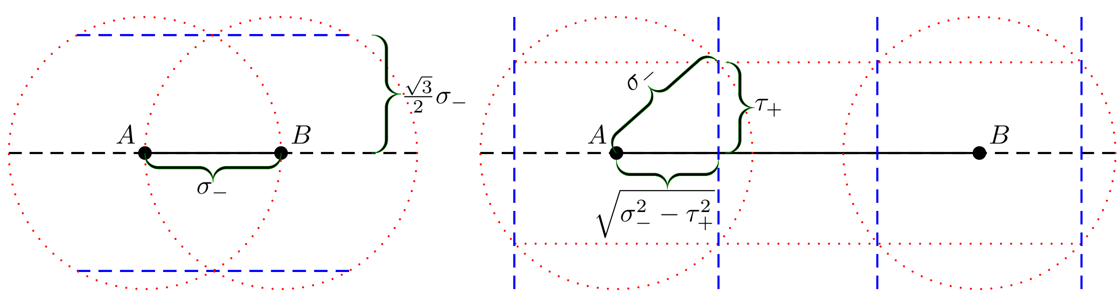
Floating point: $\mathcal{R}[m] \leq 4\epsilon$. $\mathcal{E}[m] \leq 5\epsilon L$. $\mathcal{E}[d] \leq 19\epsilon L$. $\mathcal{E}[\hat{a}] \leq 19\epsilon L$. $\mathcal{E}[\bar{a}] \leq 26\epsilon L$.

Tolerance constraints:

- $m > 0$, enforced by $m \geq \hat{\sigma}$, with $\mathcal{E}[m] \leq 5\epsilon L$, so $\hat{\sigma}_- > 5\epsilon L$.
- $\bar{a} > 0$, enforced by $\bar{a} > \sqrt{\sigma_-^2 - \tau_+^2}$, with $\mathcal{E}[\bar{a}] \leq 26\epsilon L$, so $\sqrt{\sigma_-^2 - \tau_+^2} > 26\epsilon L$.
- $\hat{a} > 0$, enforced by $\hat{a} > \sqrt{\sigma_-^2 - \tau_+^2}$, with $\mathcal{E}[\hat{a}] \leq 19\epsilon L$, so $\sqrt{\sigma_-^2 - \tau_+^2} > 19\epsilon L$.

Algorithm 3 Edge Vertex 2D

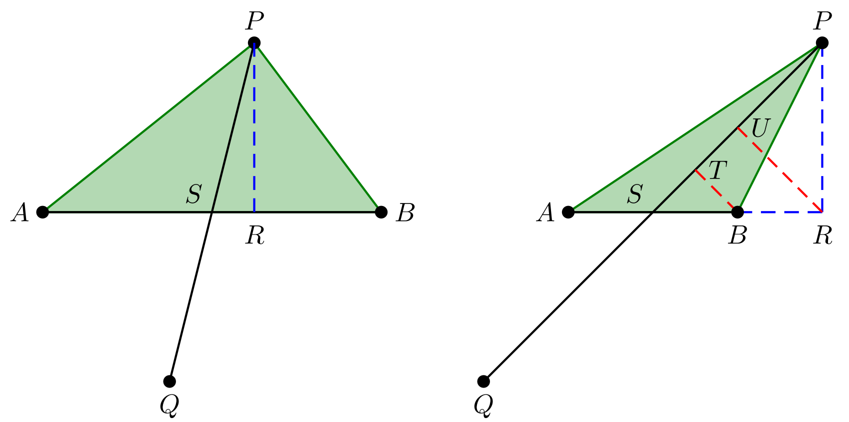
function EDGE_VERTEX(A, B, P)
    if Degenerate intersection **then**
        return (FALSE, 0)
    end if
    $\hat{u} \leftarrow A - B$
    $m \leftarrow \|\hat{u}\|$
    if $m \leq \hat{\sigma}$ **then**
        return (FALSE, 0)
    end if
    $u \leftarrow \frac{\hat{u}}{m}$
    $w \leftarrow P - A$
    if not $|u \times w| \leq \tau$ **then**
        return (FALSE, 0)
    end if
    $\hat{a} \leftarrow u \cdot w$
    $\bar{a} \leftarrow m - \hat{a}$
    if $\hat{a} < 0$ **or** $\bar{a} < 0$ **then**
        return (FALSE, 0)
    end if
    return (TRUE, $\frac{\hat{a}}{m}$)
end function



(a) Bound on m.

(b) Bound on $\hat{a}$ and $\bar{a}$.

Figure 2: Proof illustrations for vertex edge in 2D



(a) Area bound for edge-edge, R in AB. (b) Area bound for edge-edge, R not in AB.

Figure 3: Proof illustrations for edge edge in 2D.

- $\tau_+ < \frac{\sqrt{3}}{2}\sigma_-$.
- $\sigma_- = \hat{\sigma}_+$.
- $\min(\hat{\sigma}_+ - \hat{\sigma}, \hat{\sigma} - \hat{\sigma}_-) > 8\epsilon L$.
- $\min(\tau_+ - \tau, \tau - \tau_-) > 20\epsilon L$.

4.4 Edge-Edge

Let AB and PQ be two edges and let

$$a_A = 2\,\text{area}(APQ) \qquad a_B = 2\,\text{area}(BPQ) \qquad a_P = 2\,\text{area}(ABP) \qquad a_Q = 2\,\text{area}(ABQ). \tag{6}$$

The edges AB and PQ intersect if a_A and a_B differ in sign and a_P and a_Q differ in sign.

Assuming no degeneracy has been registered and an edge-edge intersection exists, $\min(|a_A|, |a_B|, |a_P|, |a_Q|) \geq \kappa_+$, where $\kappa_+ = 2\tau_-^2$. Since these are symmetrical, it suffices to consider a_P. Let $\ell(AB)$ represent the length of segment AB. Figure 3(a) and Figure 3(a) show the major cases. In Figure 3(a), the point R on line AB closest to P lies between A and B, so that the edge-vertex degeneracy assumption implies $\ell(PS) \geq \ell(PR) > \tau_-$. Figure 3(b) shows a case where R does not lie between A and B. In this case, $\ell(PR) \geq \ell(RU) \geq \ell(BT) > \tau_-$. As before, $\ell(PS) \geq \ell(PR) > \tau_-$. Repeating the logic from points A and B gives the equivalent bounds $\ell(AS) > \tau_-$ and $\ell(BS) > \tau_-$. Finally, $|a_P| = \ell(AB)\ell(PR) = (\ell(AS) + \ell(BS))\ell(PR) > 2\tau_-^2$.

Assuming the signs differ as required above, the interpolation weights are

$$\alpha_{PQ} = \frac{a_P}{a_P - a_Q} \qquad \alpha_{AB} = \frac{a_A}{a_A - a_B}, \tag{7}$$

which can always be computed robustly in floating point. The algorithm is shown in Algorithm 4.

Floating point: $\mathcal{E}[a_P] \leq 9\epsilon L^2$. $\mathcal{E}[a_Q] \leq 9\epsilon L^2$. $\mathcal{E}[a_A] \leq 9\epsilon L^2$. $\mathcal{E}[a_B] \leq 9\epsilon L^2$.

Tolerance constraints:

- $\kappa_+ = 2\tau_-^2$.
- $|a_P| > 0$, enforced by guaranteeing $|a_P| > \kappa_-$ and requiring $\kappa_- > \mathcal{E}[a_P] \geq 9\epsilon L^2$.
- $\min(\kappa_+ - \kappa, \kappa - \kappa_-) > 10\epsilon L^2$.

Algorithm 4 Edge Edge 2D

function EDGE_EDGE(A, B, P)
  if Degenerate intersection **then**
    return (FALSE, $0, 0$)
  end if
  $a_P \leftarrow (P - A) \times (B - A)$
  $a_Q \leftarrow (Q - A) \times (B - A)$
  if $|a_P| \leq \kappa$ **or** $|a_Q| \leq \kappa$ **or** $\operatorname{sgn}(a_P) = \operatorname{sgn}(a_Q)$ **then**
    return (FALSE, $0, 0$)
  end if
  $a_A \leftarrow (A - P) \times (Q - P)$
  $a_B \leftarrow (B - P) \times (Q - P)$
  if $|a_A| \leq \kappa$ **or** $|a_B| \leq \kappa$ **or** $\operatorname{sgn}(a_A) = \operatorname{sgn}(a_B)$ **then**
    return (FALSE, $0, 0$)
  end if
  return (TRUE, $\frac{a_A}{a_A - a_B}, \frac{a_P}{a_P - a_Q}$)
end function

4.5 Triangle-Vertex

Let ABC and P be a triangle and a vertex. Let

$$a_A = 2\,\text{area}(PBC) \qquad a_B = 2\,\text{area}(APC) \qquad a_C = 2\,\text{area}(ABP). \tag{8}$$

An intersection occurs if $\operatorname{sgn}(a_A) = \operatorname{sgn}(a_B) = \operatorname{sgn}(a_C)$.

Assuming no degeneracy has been registered and an triangle-vertex intersection exists, $\min(|a_A|, |a_B|, |a_C|) \geq \kappa_+$, where $\kappa_+ = 2\tau_-^2$. Since these are symmetrical, it suffices to consider a_C. Since $|a_P| = \frac{1}{2}\ell(AB)\ell(PR)$, we need to bound $\ell(AB)$ and $\ell(PR)$. If R is on segment AB, then $\ell(PR) > \tau_-$ by edge-vertex degeneracy as shown in Figure 4(a). Otherwise, we have the case shown in Figure 4(a), in which case $\ell(PR) \geq \ell(PW) \geq \ell(PS) \geq \tau_-$. Similar bounds $\ell(PT) > \tau_-$ and $\ell(PS) > \tau_-$ are obtained when considering edges AC and BC. In figure Figure 4(a) we see that $\ell(AB) \geq \ell(VU) \geq \ell(PT) + \ell(PS) \geq 2\tau_-$. Finally, $|a_P| = \frac{1}{2}\ell(AB)\ell(PR) > 2\tau_-^2$.

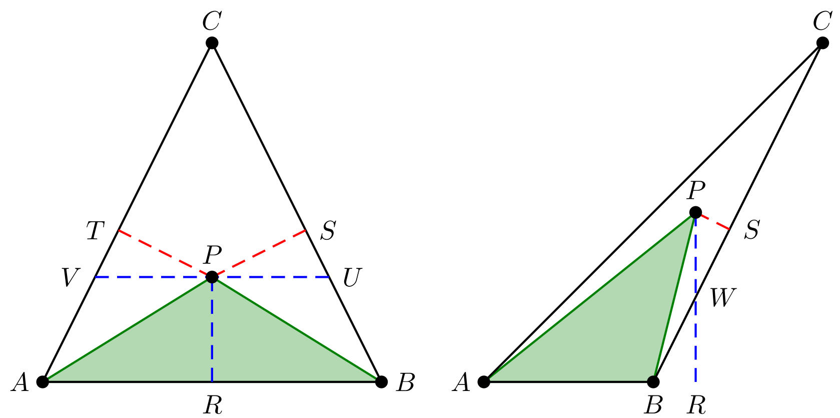
Since these areas all have the same sign, the barycentric weights are robustly computed in floating point as

$$\gamma_A = \frac{a_A}{a_A + a_B + a_C} \qquad \gamma_B = \frac{a_B}{a_A + a_B + a_C} \qquad \gamma_C = \frac{a_C}{a_A + a_B + a_C}. \tag{9}$$

The final algorithm is shown in Algorithm 5.

Algorithm 5 Triangle Vertex 2D

function TRIANGLE_VERTEX(A, B, P)
  if Degenerate intersection **then**
    return (FALSE, $0, 0, 0$)
  end if
  $a_A \leftarrow (B - P) \times (C - P)$
  $a_B \leftarrow (P - A) \times (C - A)$
  $a_C \leftarrow (B - A) \times (P - A)$
  if $|a_A| \leq \kappa$ **or** $|a_B| \leq \kappa$ **or** $|a_C| \leq \kappa$ **or** $\operatorname{sgn}(a_A) \neq \operatorname{sgn}(a_B)$ **or** $\operatorname{sgn}(a_B) \neq \operatorname{sgn}(a_C)$ **then**
    return (FALSE, $0, 0, 0$)
  end if
  return (TRUE, $\frac{a_A}{a_A + a_B + a_C}, \frac{a_B}{a_A + a_B + a_C}, \frac{a_C}{a_A + a_B + a_C}$)
end function



(a) Area bound for triangle-vertex. (b) P does not project onto AB.

Figure 4: Proof illustrations for triangle vertex in 2D.

Floating point: $\mathcal{E}[a_A] \leq 9\epsilon L^2$. $\mathcal{E}[a_B] \leq 9\epsilon L^2$. $\mathcal{E}[a_C] \leq 9\epsilon L^2$.

Tolerance constraints:

- $\kappa_+ = 2\tau_-^2$.
- $a_A > 0$, enforced by guaranteeing $|a_A| > \kappa_-$ and requiring $\kappa_- > \mathcal{E}[a_A] \geq 9\epsilon L^2$.
- $\min(\kappa_+ - \kappa, \kappa - \kappa_-) > 10\epsilon L^2$.

5 3D

5.1 Tolerance Computation

The situation with L in 3D is similar to 2D, except that tolerances will now be of the form $7\epsilon^{\frac{1}{4}}\hat{L}$, so $\hat{L} = \frac{L_0+L_1}{1-7\epsilon^{\frac{1}{4}}}$. Further, we know that $\mathcal{F}[\hat{L}] > (1-5\epsilon)\hat{L}$. Let $k = 1+5\epsilon$. To account for floating point error, we will instead define $L = k\frac{L_0+L_1}{1-7\epsilon^{\frac{1}{4}}}$. With this definition, $\mathcal{F}[L] > \hat{L}$. Then, L, even though it is computed under floating point, is an upper bound on the difference that could be computed between two coordinates. Since each tolerance ζ is computed as $\zeta = c_\zeta\sqrt{\sqrt{\epsilon}}L$ for some c_ζ, $\mathcal{R}[\zeta] < 4\epsilon$. Also, $\mathcal{R}[\zeta^2] < 9\epsilon$ and $\mathcal{R}[\sqrt{\zeta^2-\kappa^2}] < 5\epsilon$. Pseudocode for computing tolerances is in Algorithm 6.

5.2 Vertex-Vertex

Let A and B be two vertex locations. If

$$d^2 = \|A-B\|^2 \leq \sigma^2 \tag{10}$$

we register a vertex-vertex intersection. In the cases that follow, we may now assume that $\|A-B\| > \sigma_-$ in exact arithmetic. The algorithm is shown in Algorithm 7.

Floating point: $\mathcal{R}[d^2] \leq 6\epsilon$. $\mathcal{E}[d^2] \leq 16\epsilon L^2$.

Tolerance constraints:

- $\min(\sigma_+ - \sigma, \sigma - \sigma_-) > 8\epsilon\sigma$.

Algorithm 6 Computing tolerances in 3D

function COMPUTE_TOLERANCES(A, B)
  $L_a \leftarrow$ maximum bound box edge length of mesh A
  $L_b \leftarrow$ maximum bound box edge length of mesh B
  $s \leftarrow \sqrt{\epsilon}$
  $a \leftarrow \sqrt{s}$
  $b \leftarrow sa$
  $L \leftarrow \frac{1+5\epsilon}{1-7a}(L_a + L_b)$
  $c = aL$
  $d = L^2$
  $e = bLd$
  $f = \epsilon d^2$
  $\sigma = 6.5c$
  $\tau = 4.5c$
  $\delta = 2.25c$
  $\gamma = 2.25c$
  $\hat{\sigma} = 5.5c$
  $\mu = 24e$
  $\rho = 56e$
  $\xi = 56e$
  $\lambda = 1215f$
  $\phi = 470f$
  $\nu = 6844.5f$
  $\zeta = 1317f$
end function

Name	T_-	T	T_+	$\mathcal{R}[T]$	description
σ	$6.0\epsilon^{\frac{1}{4}}L$	$6.5\epsilon^{\frac{1}{4}}L$	$7.0\epsilon^{\frac{1}{4}}L$	4ϵ	vertex-vertex
τ	$4.0\epsilon^{\frac{1}{4}}L$	$4.5\epsilon^{\frac{1}{4}}L$	$5.0\epsilon^{\frac{1}{4}}L$	4ϵ	edge-vertex
δ	$2.0\epsilon^{\frac{1}{4}}L$	$2.25\epsilon^{\frac{1}{4}}L$	$2.5\epsilon^{\frac{1}{4}}L$	4ϵ	face-vertex
γ	$2.0\epsilon^{\frac{1}{4}}L$	$2.25\epsilon^{\frac{1}{4}}L$	$2.5\epsilon^{\frac{1}{4}}L$	4ϵ	edge-edge
$\hat{\sigma}$	$5.0\epsilon^{\frac{1}{4}}L$	$5.5\epsilon^{\frac{1}{4}}L$	$6.0\epsilon^{\frac{1}{4}}L$	4ϵ	vertex-vertex degeneracy rejection
λ	$910\epsilon L^4$	$1215\epsilon L^4$	$1521\epsilon L^4$	4ϵ	edge-edge area bound
ϕ	$150\epsilon L^4$	$470\epsilon L^4$	$760.5\epsilon L^4$	4ϵ	edge-edge volume bound
ν	$3422.25\epsilon L^4$	$6844.5\epsilon L^4$	$10266.75\epsilon L^4$	4ϵ	triangle-vertex area bound
ζ	$658\epsilon L^4$	$1317\epsilon L^4$	$\approx 1976\epsilon L^4$	4ϵ	triangle-vertex volume bound
μ	$16\epsilon^{\frac{3}{4}}L^3$	$24\epsilon^{\frac{3}{4}}L^3$	$32\epsilon^{\frac{3}{4}}L^3$	4ϵ	triangle-edge volume bound
ξ	$28\epsilon^{\frac{3}{4}}L^3$	$56\epsilon^{\frac{3}{4}}L^3$	$\approx 83\epsilon^{\frac{3}{4}}L^3$	4ϵ	triangle-edge volume bound
ρ	$28\epsilon^{\frac{3}{4}}L^3$	$56\epsilon^{\frac{3}{4}}L^3$	$\approx 83\epsilon^{\frac{3}{4}}L^3$	4ϵ	tetrahedron-vertex volume bound

Figure 5: Tolerances for 3D.

Algorithm 7 Vertex Vertex 3D

function VERTEX_VERTEX(A, B)
  if $\|A - B\|^2 \leq \sigma^2$ **then**
    return TRUE
  else
    return FALSE
  end if
end function

5.3 Edge-Vertex

Let AB and P be an edge and a vertex. The check we need to perform is

$$d = \frac{\|(A-B)\times(P-A)\|}{\|A-B\|} \leq \tau \qquad 0 < a < 1, \tag{11}$$

where a is the interpolation weight. Let

$$\hat{u} = A - B \quad m = \|\hat{u}\| \quad u = \frac{\hat{u}}{m} \quad w = P - A \quad \hat{a} = u \cdot w \quad \bar{a} = m - \hat{a} \quad a = \frac{\hat{a}}{m} \quad d^2 = \|u \times w\|^2. \tag{12}$$

Let us require that $\tau_+ < \frac{\sqrt{3}}{2}\sigma_-$. The details of 3D are identical to 2D, except that d now involves a square root rather than an absolute values. Since our crude approach to bounding roundoff error will not work for $\mathcal{E}[d]$, we instead check d^2 against τ^2. The bounds $|\hat{a}| > \sqrt{\sigma_-^2 - \tau_+^2}$ and $|\bar{a}| > \sqrt{\sigma_-^2 - \tau_+^2}$ and the correctness of the algorithm (see Algorithm 8) are obtained in the same way as in 2D.

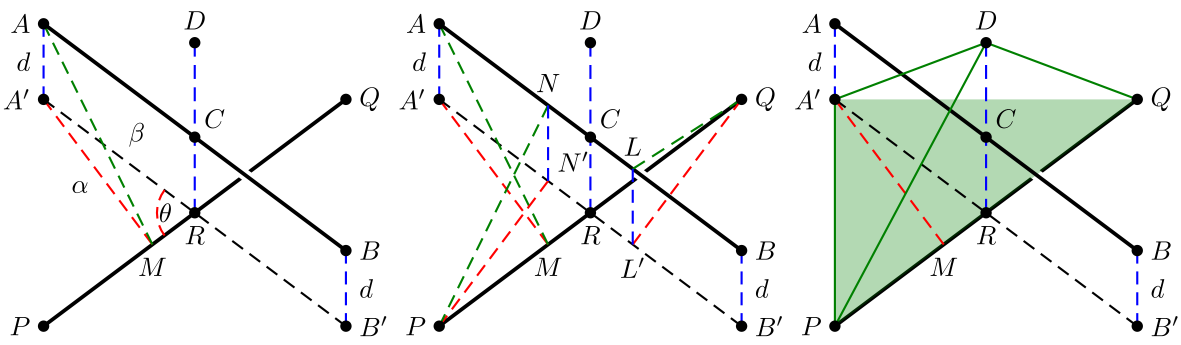
Algorithm 8 Edge Vertex 3D

function EDGE_VERTEX(A, B, P)
  if Degenerate intersection **then**
    return (FALSE, 0)
  end if
  $\hat{u} \leftarrow A - B$
  $m \leftarrow \|\hat{u}\|$
  if $m \leq \hat{\sigma}$ **then**
    return (FALSE, 0)
  end if
  $u \leftarrow \frac{\hat{u}}{m}$
  $w \leftarrow P - A$
  if not $\|u \times w\|^2 \leq \tau^2$ **then**
    return (FALSE, 0)
  end if
  $\hat{a} \leftarrow u \cdot w$
  $\bar{a} \leftarrow m - \hat{a}$
  if $\hat{a} < 0$ **or** $\bar{a} < 0$ **then**
    return (FALSE, 0)
  end if
  return (TRUE, $\frac{\hat{a}}{m}$)
end function

Floating point: $\mathcal{R}[m] \leq 4\epsilon$. $\mathcal{E}[m] \leq 7\epsilon L$. $\mathcal{E}[d^2] \leq 261\epsilon L^2$. $\mathcal{E}[\hat{a}] \leq 31\epsilon L$. $\mathcal{E}[\bar{a}] \leq 42\epsilon L$.

Tolerance constraints:

- $m > 0$, enforced by $m \geq \hat{\sigma}$, with $\mathcal{E}[m] \leq 7\epsilon L$, so $\hat{\sigma}_- > 7\epsilon L$.
- $\bar{a} > 0$, enforced by $\bar{a} > \sqrt{\sigma_-^2 - \tau_+^2}$, with $\mathcal{E}[\bar{a}] \leq 42\epsilon L$, so $\sqrt{\sigma_-^2 - \tau_+^2} > 42\epsilon L$.
- $\hat{a} > 0$, enforced by $\hat{a} > \sqrt{\sigma_-^2 - \tau_+^2}$, with $\mathcal{E}[\hat{a}] \leq 31\epsilon L$, so $\sqrt{\sigma_-^2 - \tau_+^2} > 31\epsilon L$.
- $\tau_+ < \frac{\sqrt{3}}{2}\sigma_-$.



(a) A (b) B (c) C

Figure 6: Proof illustrations for edge edge in 3D.

- $\sigma_- = \hat{\sigma}_+$.
- $\min(\hat{\sigma}_+ - \hat{\sigma}, \hat{\sigma} - \hat{\sigma}_-) > 9\epsilon L$.
- $\min(\tau_+ - \tau, \tau - \tau_-) > \frac{131\epsilon L^2}{\tau}$.

5.4 Edge-Edge

Let AB and PQ be two edges. Let

$$u = B - A \quad v = Q - P \quad w = P - A \quad r = u \times v \quad m^2 = \|r\|^2 \quad n = r \times w \tag{13}$$

The distance d and interpolation weights a (from A to B) and b (from P to Q) are

$$\hat{d} = r \cdot w \quad \hat{a} = n \cdot v \quad \hat{b} = n \cdot u \quad d = \frac{|\hat{d}|}{m} \quad a = \frac{\hat{a}}{m^2} \quad b = \frac{\hat{b}}{m^2} \quad \bar{a} = m^2 - \hat{a} \quad \bar{a} = m^2 - \hat{a} \tag{14}$$

and the intersection should be registered if $\hat{a} > 0$, $\bar{a} > 0$, $\hat{b} > 0$, $\bar{b} > 0$, and $\hat{d}^2 < \gamma^2 m^2$. In the case that an intersection has occurred but no degenerate intersection has been registered, we can bound both m, $\hat{a}$, $\bar{a}$, $\hat{b}$, and $\bar{b}$ from zero.

5.4.1 Bound on m

Let C be the point on AB closest to PQ, and R is the point on PQ closest to AB. Assume WLOG $\beta = \ell(AC) = \min(\ell(AC), \ell(BC), \ell(PR), \ell(QR))$. Let A' and B' be the points A and B projected down to the plane Z containing PQ and parallel to AB as shown in Figure 6(a). Assume also that the angle $\theta = A'RP$ is the smaller of the two angles made between $A'B'$ and PQ. Let M be the point on PQ closest to A'. Let $\alpha = \ell(A'M)$ and note that $\beta = \ell(AC) = \ell(A'R)$. $\sin\theta = \frac{\alpha}{\beta}$. Note that the definition of β implies that M will lie on segment PQ. Since $\ell(AM)$ is an edge-vertex pair, $\ell(AM) > \tau_-$. $\ell(AA')$ is the distance from AB to the plane of $A'B'$, so $\ell(AA') = d \leq \gamma_+$. $\beta \geq \alpha = \ell(A'M) = \sqrt{\ell(AM)^2 - \ell(AA')^2} > \sqrt{\tau_-^2 - \gamma_+^2}$. Using the definition of β, $\|u\| = \ell(AB) = \ell(AC) + \ell(BC) \geq 2\beta$ and $\|v\| = \ell(PQ) = \ell(PR) + \ell(QR) \geq 2\beta$. Finally, $m = \|r\| = \|u \times v\| = \|u\|\|v\| \sin\theta \geq (2\beta)(2\beta)\frac{\alpha}{\beta} = 4\alpha\beta > 4(\tau_-^2 - \gamma_+^2)$.

5.4.2 Bound on $\hat{a}$

In Figure 6(b), A' and A project onto line PQ at M, B' and B project onto line PQ at K (not shown), P projects onto lines AB and $A'B'$ at N and N', and Q projects onto lines AB and $A'B'$ at L and

L'. Triangles $RA'M$, $RB'K$, RPN', and RQL' are similar triangles. Since $\beta = \ell(AC) = \ell(A'R) = \min(\ell(AC), \ell(BC), \ell(PR), \ell(QR)) = \min(\ell(A'R), \ell(B'R), \ell(PR), \ell(QR))$, we also have $\sqrt{\tau_-^2 - \gamma_+^2} \le \alpha = \ell(A'M) = \min(\ell(A'M), \ell(B'K), \ell(PN'), \ell(QL'))$.

Let D be a point on line RC such that $|RD| = m > 0$. $|\hat{a}| = |(r \times w) \cdot v| = |((D-R) \times (P-A)) \cdot (Q-P)| = |((D-R) \times (P-A')) \cdot (Q-P)| = |((D-P) \times (A'-P)) \cdot (Q-P)| = 6\,|\text{vol}(A'DPQ)| = 2m\,\text{area}(A'PQ) = m\ell(A'M)(\ell(PR) + \ell(QR)) > 2m(\tau_-^2 - \gamma_+^2) > 8(\tau_-^2 - \gamma_+^2)^2$. The same bounds apply to the symmetrical cases $|\bar{a}|$, $|\hat{b}|$, and $|\bar{b}|$.

5.4.3 Algorithm

Pseudocode for the edge edge case is shown in Figure 9. The main difficulty in proving correctness is the form of the comparison on line 14. We must be able to bound $\mathcal{E}[m] \ll m$ for reasonable bounds to be found for γ_- and γ_+. There are two difficulties with doing this. The first is that no bounds on $\mathcal{E}[m]$ can be derived using the simple automated framework in Section 2, and the second complication is that $\mathcal{E}[m^2]$ can be derived but not $\mathcal{R}[m^2]$. Using the test $\hat{d}^2 \le \gamma^2 m^2$ rather than $|\hat{d}| \le \gamma m$ avoids the need to compute m entirely.

Deriving $\mathcal{R}[m^2]$ from $\mathcal{E}[m^2]$ can be achieved by bounding m^2 from below. Unfortunately, the bound on m^2 only holds provided the intersection exists, but the test on line 14 must be reliable even when no intersection exists since it might otherwise allow an edge edge pair that is too distant to be counted as an intersection. The solution to this is to bound m^2 first using the test on line 9, where $\lambda_+ = 16(\tau_-^2 - \gamma_+^2)^2$. If the test on line 9 causes the algorithm to terminate, then $\mathcal{F}[\lambda - m^2] \ge 0$, in which case $m^2 \le \lambda_+ = 16(\tau_-^2 - \gamma_+^2)^2$ implies no intersection. If the algorithm continues to line 12, then $m^2 \ge \lambda_-$. Correctness of line 14 depends on suitably choosing γ_- and γ_+.

What remains is to check the signs of $\hat{a}$, $\bar{a}$, $\hat{b}$, and $\bar{b}$. Following the established logic, we would do two checks for each, namely $|\hat{a}| \le \phi$ and $\hat{a} < 0$, where $\phi_+ = 8(\tau_-^2 - \gamma_+^2)^2$. If either test succeeds, there is no intersection. Both tests together are equivalent (even under floating point) to the test $\hat{a} \le \phi$. The logic is the same for the other three sign checks, so line 22 performs the remaining checks correctly. Note that the way $\bar{a}$ was computed and checked for sign implies that $\frac{\hat{a}}{m^2}$ will be computed between 0 and 1 in floating point. The case of $\bar{b}$ is similar, so that my returned results are robust (there may be some accuracy loss, however).

5.4.4 Addressing $\gamma^2 m^2 - \hat{d}^2$

We want $\mathcal{F}[\gamma^2 m^2 - \hat{d}^2] \ge 0$ to imply $\gamma_+^2 m^2 - \hat{d}^2 \ge 0$, and $\mathcal{F}[\gamma^2 m^2 - \hat{d}^2] < 0$ to imply $\gamma_-^2 m^2 - \hat{d}^2 < 0$, and we prove that the tolerances given in Figure 5 will guarantee this.

Since $m^2 > \lambda_- = 910\epsilon L^4$ and $\mathcal{E}[m^2] \le 129\epsilon L^4$, $\mathcal{R}[m^2] < \frac{129}{910}$. Using $\mathcal{R}[\gamma^2] < 10\epsilon$ we have $\mathcal{R}[\gamma^2 m^2] < (1 + 12\epsilon)\mathcal{R}[m^2] + 12\epsilon < \frac{1}{7}$ and then $\mathcal{E}[\gamma^2 m^2] < \frac{1}{7}\gamma^2 m^2$.

Note that if $\hat{d}^2 - \mathcal{E}[\hat{d}^2] > \gamma^2 m^2 + \mathcal{E}[\gamma^2 m^2]$, we will have $\mathcal{F}[\gamma^2 m^2 - \hat{d}^2] < 0$, so that the sign can be computed unambiguously. Thus, we can restrict ourselves to the case $\hat{d}^2 - \mathcal{E}[\hat{d}^2] \le \gamma^2 m^2 + \mathcal{E}[\gamma^2 m^2]$ or $\hat{d}^2 \le \frac{8}{7}\gamma^2 m^2 + \mathcal{E}[\hat{d}^2]$. We also have $\mathcal{F}[\hat{d}^2] \le \left(\hat{d} + \mathcal{E}[\hat{d}]\right)^2 (1+\epsilon) = \hat{d}^2 + \hat{d}^2\epsilon + 2\hat{d}\mathcal{E}[\hat{d}](1+\epsilon) + \mathcal{E}[\hat{d}]^2(1+\epsilon)$ and $\mathcal{F}[\hat{d}^2] \ge \left(\hat{d} - \mathcal{E}[\hat{d}]\right)^2 (1-\epsilon) = \hat{d}^2 - \hat{d}^2\epsilon - 2\hat{d}\mathcal{E}[\hat{d}](1-\epsilon) + \mathcal{E}[\hat{d}]^2(1-\epsilon)$, so $\mathcal{E}[\hat{d}^2] \le \hat{d}^2\epsilon + 2\hat{d}\mathcal{E}[\hat{d}](1+\epsilon) + \mathcal{E}[\hat{d}]^2(1+\epsilon) \le (\frac{8}{7}\gamma^2 m^2 + \mathcal{E}[\hat{d}^2])\epsilon + 2\sqrt{\frac{8}{7}\gamma^2 m^2 + \mathcal{E}[\hat{d}^2]}\,\mathcal{E}[\hat{d}](1+\epsilon) + \mathcal{E}[\hat{d}]^2(1+\epsilon)$.

Let $g(x) = (\frac{8}{7}\gamma^2 m^2 + x)\epsilon + 2\sqrt{\frac{8}{7}\gamma^2 m^2 + x}\mathcal{E}[\hat{d}](1+\epsilon) + \mathcal{E}[\hat{d}]^2(1+\epsilon) - x$. Note that

$$\begin{aligned}
g\left(\frac{2}{35}\gamma^2 m^2\right) &= \left(\frac{8}{7}\gamma^2 m^2 + \frac{2}{35}\gamma^2 m^2\right)\epsilon + 2\sqrt{\frac{8}{7}\gamma^2 m^2 + \frac{2}{35}\gamma^2 m^2}\mathcal{E}[\hat{d}](1+\epsilon) + \mathcal{E}[\hat{d}]^2(1+\epsilon) - \frac{2}{35}\gamma^2 m^2 \\
&= \gamma^2 m^2\left(\frac{6}{5}\epsilon - \frac{2}{35}\right) + 2\sqrt{\frac{6}{5}}\gamma m \mathcal{E}[\hat{d}](1+\epsilon) + \mathcal{E}[\hat{d}]^2(1+\epsilon) \\
&\leq \gamma^2 m^2\left(\frac{6}{5}\epsilon - \frac{2}{35}\right) + 2\sqrt{\frac{6}{5}}\gamma m (47\epsilon L^3)(1+\epsilon) + (47\epsilon L^3)^2(1+\epsilon) \\
&= \left(\gamma m\left(\frac{6}{5}\epsilon - \frac{2}{35}\right) + 94\sqrt{\frac{6}{5}}\epsilon L^3(1+\epsilon)\right)\gamma m + (47\epsilon L^3)^2(1+\epsilon) \\
&\leq \left((2.25\epsilon^{\frac{1}{4}}L)(4(\tau_-^2 - \gamma_+^2))\left(\frac{6}{5}\epsilon - \frac{2}{35}\right) + 94\sqrt{\frac{6}{5}}\epsilon L^3(1+\epsilon)\right)\gamma m + (47\epsilon L^3)^2(1+\epsilon) \\
&= \left(\frac{351}{4}\epsilon^{\frac{3}{4}}L^3\left(\frac{6}{5}\epsilon - \frac{2}{35}\right) + 94\sqrt{\frac{6}{5}}\epsilon L^3(1+\epsilon)\right)\gamma m + (47\epsilon L^3)^2(1+\epsilon) \\
&= \left(\frac{351}{4}\left(\frac{6}{5}\epsilon - \frac{2}{35}\right) + 94\sqrt{\frac{6}{5}}\epsilon^{\frac{1}{4}}(1+\epsilon)\right)\epsilon^{\frac{3}{4}}L^3\gamma m + (47\epsilon L^3)^2(1+\epsilon) \\
&\leq -5\epsilon^{\frac{3}{4}}L^3\gamma m + (47\epsilon L^3)^2(1+\epsilon) \\
&\leq -5\epsilon^{\frac{3}{4}}L^3(2.25\epsilon^{\frac{1}{4}}L)(4(\tau_-^2 - \gamma_+^2)) + (47\epsilon L^3)^2(1+\epsilon) \\
&\leq -\frac{1755}{4}\epsilon^{\frac{3}{2}}L^6 + (47\epsilon L^3)^2(1+\epsilon) \\
&= \left(-\frac{1755}{4} + 47^2\epsilon^{\frac{1}{2}}(1+\epsilon)\right)\epsilon^{\frac{3}{2}}L^6 \\
&\leq -400\epsilon^{\frac{3}{2}}L^6 \\
&< 0
\end{aligned}$$

Further, if $x \geq \frac{2}{35}\gamma^2 m^2$ then

$$\begin{aligned}
g'(x) &= \epsilon + \frac{\mathcal{E}[\hat{d}](1+\epsilon)}{\sqrt{\frac{8}{7}\gamma^2 m^2 + x}} - 1 \\
&\leq \epsilon + \frac{\mathcal{E}[\hat{d}](1+\epsilon)}{\sqrt{\gamma^2 m^2}} - 1 \\
&= \epsilon + \frac{\mathcal{E}[\hat{d}](1+\epsilon)}{\gamma m} - 1 \\
&\leq \epsilon + \frac{47\epsilon L^3(1+\epsilon)}{(2.25\epsilon^{\frac{1}{4}}L)(4(\tau_-^2 - \gamma_+^2))} - 1 \\
&= \epsilon + \frac{188\epsilon^{\frac{1}{4}}(1+\epsilon)}{351} - 1 \\
&\leq \epsilon + \frac{188\epsilon^{\frac{1}{4}}(1+\epsilon)}{351} - 1 \\
&< -0.9
\end{aligned}$$

Thus, I conclude that $g(x) < 0$ for all $x \geq \frac{2}{35}\gamma^2 m^2$. Since $g(\mathcal{E}[\hat{d}^2]) \geq 0$, we must have $\mathcal{E}[\hat{d}^2] < \frac{2}{35}\gamma^2 m^2$.

We know $\mathcal{F}[\gamma^2 m^2 - \hat{d}^2] < 0$ implies $\mathcal{F}[\gamma^2 m^2] - \mathcal{F}[\hat{d}^2] < 0$, which leads to $0 > \mathcal{F}[\gamma^2 m^2] - \mathcal{F}[\hat{d}^2] > \gamma^2 m^2 - \mathcal{E}[\gamma^2 m^2] - \hat{d}^2 - \mathcal{E}[\hat{d}^2] > \gamma^2 m^2 - \frac{1}{7}\gamma^2 m^2 - \hat{d}^2 - \frac{2}{35}\gamma^2 m^2 \geq \frac{4}{5}\gamma^2 m^2 - \hat{d}^2 > \gamma_-^2 m^2 - \hat{d}^2$. Similarly $\mathcal{F}[\gamma^2 m^2 - \hat{d}^2] \geq 0$ implies $\mathcal{F}[\gamma^2 m^2] - \mathcal{F}[\hat{d}^2] \geq 0$, which leads to $0 \leq \mathcal{F}[\gamma^2 m^2] - \mathcal{F}[\hat{d}^2] \leq \gamma^2 m^2 + \mathcal{E}[\gamma^2 m^2] - \hat{d}^2 + \mathcal{E}[\hat{d}^2] \leq \gamma^2 m^2 + \frac{1}{7}\gamma^2 m^2 - \hat{d}^2 + \frac{2}{35}\gamma^2 m^2 = \frac{6}{5}\gamma^2 m^2 - \hat{d}^2 \leq \gamma_+^2 m^2 - \hat{d}^2$.

Algorithm 9 Edge Edge 3D

function EDGE_EDGE(A, B, P, Q)
    if Degenerate intersection **then**
        return (FALSE, $0, 0$)
    end if
    $u \leftarrow B - A$
    $v \leftarrow Q - P$
    $r \leftarrow u \times v$
    $m^2 \leftarrow \|r\|^2$
    if $m^2 \leq \lambda$ **then**
        return (FALSE, $0, 0$)
    end if
    $w \leftarrow P - A$
    $\hat{d} \leftarrow r \cdot w$
    if not $\hat{d}^2 \leq \gamma^2 m^2$ **then**
        return (FALSE, $0, 0$)
    end if
    $n \leftarrow r \times w$
    $\hat{a} = n \cdot v$
    $\hat{b} = n \cdot u$
    $\bar{a} = m^2 - \hat{a}$
    $\bar{b} = m^2 - \hat{b}$
    if $\hat{a} \leq \phi$ **or** $\hat{b} \leq \phi$ **or** $\bar{a} \leq \phi$ **or** $\bar{b} \leq \phi$ **then**
        return (FALSE, $0, 0$)
    end if
    return (TRUE, $\frac{\hat{a}}{m^2}, \frac{\hat{b}}{m^2}$)
end function

Floating point: $\mathcal{E}[m^2] \leq 129\epsilon L^4$. $\mathcal{E}[\hat{d}] \leq 47\epsilon L^3$. $\mathcal{E}[\hat{d}^2] \leq 589\epsilon L^6$. $\mathcal{E}[\hat{a}] \leq 129\epsilon L^4$. $\mathcal{E}[\hat{b}] \leq 129\epsilon L^4$. $\mathcal{E}[\bar{a}] \leq 281\epsilon L^4$. $\mathcal{E}[\bar{b}] \leq 281\epsilon L^4$. $m^2 \leq 12L^4$.

Tolerance constraints:

- $\lambda_+ = 16(\tau_-^2 - \gamma_+^2)^2$
- $\phi_+ = 8(\tau_-^2 - \gamma_+^2)^2$
- $m^2 \leq \lambda$ with $\mathcal{E}[m^2] \leq 129\epsilon L^4$ leads to $\min(\lambda_+ - \lambda, \lambda - \lambda_-) > 130\epsilon L^4$
- $\hat{a} \leq \phi$ with $\mathcal{E}[\hat{a}] \leq 129\epsilon L^4$ leads to $\min(\phi_+ - \phi, \phi - \phi_-) > 130\epsilon L^4$.
- $\bar{a} \leq \phi$ with $\mathcal{E}[\bar{a}] \leq 281\epsilon L^4$ leads to $\min(\phi_+ - \phi, \phi - \phi_-) > 282\epsilon L^4$.
- $\hat{d}^2 \leq \gamma^2 m^2$, addressed in Section 5.4.4.

5.5 Triangle-Vertex

Let ABC and P be a triangle and a vertex. Then, Let

$$u = B - A \quad v = C - A \quad w = P - A \quad r = u \times v \quad m^2 = \|r\|^2 \quad n = r \times w \tag{15}$$

Now, the signed distance d and barycentric weights a, b, c can be computed from

$$\hat{d} = r \cdot w \quad \hat{c} = -n \cdot u \quad \hat{b} = n \cdot v \quad \hat{a} = m^2 - \hat{b} - \hat{c} \quad d = \frac{\hat{d}}{m} \quad c = \frac{\hat{c}}{m^2} \quad b = \frac{\hat{b}}{m^2} \quad a = \frac{\hat{a}}{m^2}. \tag{16}$$

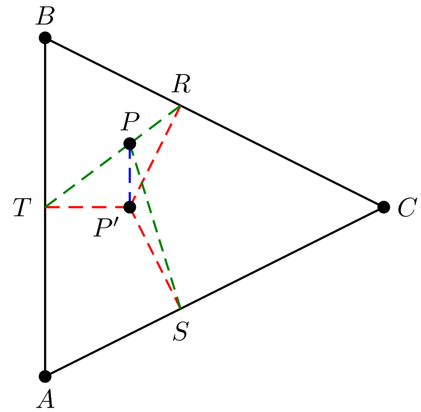
The intersection criterion is now $\hat{d}^2 \leq \delta^2 m^2$, $\hat{a} > 0$, $\hat{b} > 0$, and $\hat{c} > 0$. If satisfied and no degenerate intersection was registered, then we can bound m, $\hat{a}$, $\hat{b}$, and $\hat{c}$ from zero.

Algorithm 10 Triangle Vertex 3D

function TRIANGLE_VERTEX(A, B, C, P)
  if Degenerate intersection **then**
    return (FALSE, $0, 0, 0$)
  end if
  $u \leftarrow B - A$
  $v \leftarrow C - A$
  $r \leftarrow u \times v$
  $m^2 \leftarrow \|r\|^2$
  if $m^2 \leq \nu$ **then**
    return (FALSE, $0, 0, 0$)
  end if
  $w \leftarrow P - A$
  $\hat{d} \leftarrow r \cdot w$
  if not $\hat{d}^2 \leq \delta^2 m^2$ **then**
    return (FALSE, $0, 0, 0$)
  end if
  $n \leftarrow r \times w$
  $\hat{b} = n \cdot v$
  $\hat{c} = -n \cdot u$
  $\hat{a} = m^2 - \hat{b} - \hat{c}$
  if $\hat{a} \leq \zeta$ **or** $\hat{b} \leq \zeta$ **or** $\hat{c} \leq \zeta$ **then**
    return (FALSE, $0, 0, 0$)
  end if
  return (TRUE, $\frac{\hat{a}}{m^2}, \frac{\hat{b}}{m^2}, \frac{\hat{c}}{m^2}$)
end function

5.5.1 Bound on m

Let P' be P projected into the plane ABC as shown in Figure 7(a). P projects to lines AB, AC, and BC at T, S, and R respectively. If R lies between B and C, then $\ell(PR) > \tau_-$. Since $\ell(P'P) = d \leq \delta_+$, we have $\ell(P'R) > \sqrt{\tau_-^2 - \delta_+^2}$, with similar bounds for $\ell(P'S)$ and $\ell(P'T)$. If R projects beyond B and C, the same bound can be obtained from $\ell(P'S)$ or $\ell(P'T)$ using the same logic as in the 2D case. Note that the inradius r of ABC must satisfy $r > \sqrt{\tau_-^2 - \delta_+^2}$. Noting also that the ratio of the area of the incircle to the area of the triangle must satisfy $\frac{\pi r^2}{\text{area}(ABC)} \leq \frac{\pi}{3\sqrt{3}}$, with equality for an equilateral triangle, we have $m = 2\,\text{area}(ABC) \geq 6\sqrt{3}r^2 > 6\sqrt{3}(\tau_-^2 - \delta_+^2)$.



(a) Area bound for triangle-vertex.

Figure 7: Proof illustrations for triangle vertex in 3D.

5.5.2 Bound on $\hat{a}$, $\hat{b}$, and $\hat{c}$

Let $s = P' - P$ and $V = |\text{vol}(ABP'P)| = \frac{1}{6}|((P' - A) \times (P - A)) \cdot (B - A)| = \frac{1}{6}|((P' - P) \times (P - A)) \cdot (B - A)| = \frac{1}{6}|(s \times w) \cdot u|$. Since $s = \pm\frac{d}{m}r$, $|\hat{c}| = |-n \cdot u| = |(r \times w) \cdot u| = \frac{m}{d}|(s \times w) \cdot u| = \frac{6m}{d}V = \frac{2m}{d}\ell(PP')\text{area}(ABP') = 2m\,\text{area}(ABP')$. The bound $\text{area}(ABP') > \tau_-^2 - \delta_+^2$ is obtained from $\min(\ell(P'R), \ell(P'S), \ell(P'T)) > \sqrt{\tau_-^2 - \delta_+^2}$ as in the 2D case. Finally, $|\hat{c}| > 2m(\tau_-^2 - \delta_+^2) > 12\sqrt{3}(\tau_-^2 - \delta_+^2)^2$.

5.5.3 Algorithm

Pseudocode for the face-vertex case is shown in Figure 10. This algorithm and its correctness follows the edge-edge case rather closely. Line 9 bounds m^2 from zero using the fact that $m^2 > \nu_+ = (6\sqrt{3}(\tau_-^2 - \delta_+^2))^2 = 108(\tau_-^2 - \delta_+^2)^2$ whenever an intersection should be registered. In particular, if the comparison on line 9 succeeds and the algorithm terminates, then we know that no intersection should have been registered. This test allows the test on line 14 to be reliable. The correctness of the test on line 14 will follow from choosing δ_- and δ_+ properly.

What remains is the check on the signs of $\hat{a}$, $\hat{b}$, and $\hat{c}$ on line 21. Following the logic of the edge-edge case, we use tests of the form $\hat{a} \leq \zeta$, where $\zeta_+ = 12\sqrt{3}(\tau_-^2 - \delta_+^2)^2$, since this covers both the sign and magnitude checks on $\hat{a}$ that would otherwise be required. The roundoff error analysis for $\hat{d}^2 \leq \delta^2 m^2$ is identical to the edge-edge case and is omitted here.

Floating point: $\mathcal{E}[m^2] \leq 129\epsilon L^4$. $\mathcal{E}[\hat{d}] \leq 47\epsilon L^3$. $\mathcal{E}[\hat{a}] \leq 445\epsilon L^4$. $\mathcal{E}[\hat{b}] \leq 129\epsilon L^4$. $\mathcal{E}[\hat{c}] \leq 129\epsilon L^4$.

Tolerance constraints:

- $\nu_+ = 108(\tau_-^2 - \delta_+^2)^2$
- $\zeta_+ = 12\sqrt{3}(\tau_-^2 - \delta_+^2)^2$
- $m^2 \leq \nu$ with $\mathcal{E}[m^2] \leq 129\epsilon L^4$ leads to $\min(\nu_+ - \nu, \nu - \nu_-) > 130\epsilon L^4$
- $\hat{a} \leq \zeta$ with $\mathcal{E}[\hat{a}] \leq 445\epsilon L^4$ leads to $\min(\zeta_+ - \zeta, \zeta - \zeta_-) > 446\epsilon L^4$.
- $\hat{b} \leq \zeta$ with $\mathcal{E}[\hat{b}] \leq 129\epsilon L^4$ leads to $\min(\zeta_+ - \zeta, \zeta - \zeta_-) > 130\epsilon L^4$.
- $\hat{c} \leq \zeta$ with $\mathcal{E}[\hat{c}] \leq 129\epsilon L^4$ leads to $\min(\zeta_+ - \zeta, \zeta - \zeta_-) > 130\epsilon L^4$.

- $\hat{d}^2 \leq \delta^2 m^2$, identical to the case addressed in Section 5.4.4, since $\gamma_- = \delta_-$, $\gamma = \delta$, $\gamma_+ = \delta_+$, and the values $\mathcal{E}[m^2]$ and $\mathcal{E}[\hat{d}]$ are the same.

5.6 Triangle-Edge

Let ABC and PQ be a triangle and an edge and

$$v_A = 6\,\mathrm{vol}(BCPQ) \qquad v_B = 6\,\mathrm{vol}(CAPQ) \qquad v_C = 6\,\mathrm{vol}(ABPQ) \tag{17}$$

$$v_P = 6\,\mathrm{vol}(ABCP) \qquad v_Q = 6\,\mathrm{vol}(ABCQ). \tag{18}$$

The triangle intersection barycentric coordinates are given by

$$\gamma_A = \frac{v_A}{v_A + v_B + v_C} \quad \gamma_B = \frac{v_B}{v_A + v_B + v_C} \quad \gamma_C = \frac{v_C}{v_A + v_B + v_C} \quad \alpha_P = \frac{v_P}{v_P - v_Q} \quad \alpha_Q = \frac{v_Q}{v_P - v_Q}. \tag{19}$$

The intersection criteria are that α_P and α_Q differ in sign but that γ_A, γ_B, and γ_C agree in sign. If these conditions are met, we show the above computations to be robust.

Be begin with a bound on $|v_P|$ and $|v_Q|$. Consider the case shown in Figure 8(a), where P projects to plane ABC at a point P in triangle ABC and Q projects to a point Q' outside triangle ABC. Then, $\ell(PP') > \delta_-$ and $\ell(QQ') \geq \ell(SR) > \gamma_-$. Thus, we see that, in general, $\ell(PP') > \min(\gamma_-, \delta_-)$. $|v_P| = 6\,|\mathrm{vol}(ABCP)| = 2\,\mathrm{area}(ABC)\ell(PP')$. Note that $\min(\ell(DT), \ell(DU), \ell(DV)) > \gamma_-$, so that the case of bounding $\mathrm{area}(ABC)$ is analogous to that considered in Section 5.5.1 and leads to $2\mathrm{area}(ABC) > 6\sqrt{3}\gamma_-^2$. Finally, $|v_P| = 2\,\mathrm{area}(ABC)\ell(PP') > 6\sqrt{3}\gamma_-^2 \min(\gamma_-, \delta_-) = \xi_+$.

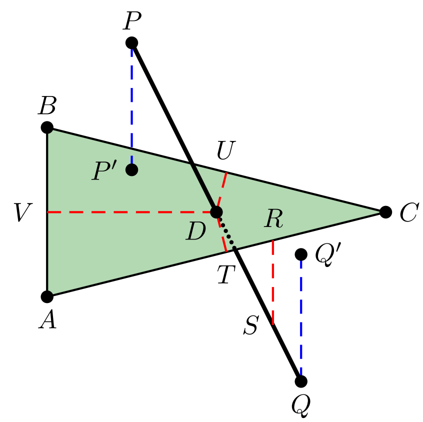
Next, we consider a bound on $|v_A|$, $|v_B|$, and $|v_C|$. Here, $|v_A| = 6\,|\mathrm{vol}(BCPQ)| = 6\,|\mathrm{vol}(BCPD)| + 6\,|\mathrm{vol}(BCDQ)| = 2\mathrm{area}(DBC)\ell(PP') + 2\mathrm{area}(DBC)\ell(QQ')$. Bounding $\mathrm{area}(DBC)$ is analogous to the proof in Section 4.5 using $\min(\ell(DT), \ell(DU), \ell(DV)) > \gamma_-$, leading to $2\,\mathrm{area}(DBC) > 2\gamma_-^2$. Finally, $|v_A| = 2\mathrm{area}(DBC)(\ell(PP') + \ell(QQ')) > 4\gamma_-^2 \min(\gamma_-, \delta_-) = \mu_+$.

The logic for Algorithm 11 is very similar to that of Algorithm 4 and Algorithm 5, where we perform separate sign and magnitude comparisons.

Algorithm 11 Triangle Edge 3D

function TRIANGLE_EDGE(A, B, C, P, Q)
  if Degenerate intersection **then**
    return (FALSE, $0, 0, 0, 0$)
  end if
  $v_A \leftarrow ((B - Q) \times (C - Q)) \cdot (P - Q)$
  $v_B \leftarrow ((C - Q) \times (A - Q)) \cdot (P - Q)$
  $v_C \leftarrow ((A - Q) \times (B - Q)) \cdot (P - Q)$
  if $|v_A| \leq \mu$ **or** $|v_B| \leq \mu$ **or** $|v_C| \leq \mu$ **or** $\mathrm{sgn}(v_A) \neq \mathrm{sgn}(v_B)$ **or** $\mathrm{sgn}(v_B) \neq \mathrm{sgn}(v_C)$ **then**
    return (FALSE, $0, 0, 0, 0$)
  end if
  $v_P \leftarrow ((A - P) \times (B - P)) \cdot (C - P)$
  $v_Q \leftarrow ((A - Q) \times (B - Q)) \cdot (C - Q)$
  if $|v_P| \leq \xi$ **or** $|v_Q| \leq \xi$ **or** $\mathrm{sgn}(v_P) = \mathrm{sgn}(v_Q)$ **then**
    return (FALSE, $0, 0, 0, 0$)
  end if
  return (TRUE, $\frac{v_A}{v_A+v_B+v_C}, \frac{v_B}{v_A+v_B+v_C}, \frac{v_C}{v_A+v_B+v_C}, \frac{v_P}{v_P-v_Q}$)
end function

Floating point: $\mathcal{E}[v_A] \leq 47\epsilon L^3$. $\mathcal{E}[v_B] \leq 47\epsilon L^3$. $\mathcal{E}[v_C] \leq 47\epsilon L^3$. $\mathcal{E}[v_P] \leq 47\epsilon L^3$. $\mathcal{E}[v_Q] \leq 47\epsilon L^3$.



(a) Area bound for triangle-edge.

Figure 8: Proof illustrations for triangle edge in 3D.

Tolerance constraints:

- $\mu_+ = 4\gamma_-^2 \min(\gamma_-, \delta_-)$
- $\xi_+ = 6\sqrt{3}\gamma_-^2 \min(\gamma_-, \delta_-)$
- $v_A > 0$, enforced by guaranteeing $|v_A| > \mu_-$ and requiring $\mu_- > \mathcal{E}[v_A] \geq 47\epsilon L^3$.
- $v_P > 0$, enforced by guaranteeing $|v_P| > \xi_-$ and requiring $\xi_- > \mathcal{E}[v_P] \geq 47\epsilon L^3$.
- $\min(\mu_+ - \mu, \mu - \mu_-) > 48\epsilon L^3$.
- $\min(\xi_+ - \xi, \xi - \xi_-) > 48\epsilon L^3$.
- Similar bounds for v_B, v_C, and v_Q

5.7 Tetrahedron-Vertex

Let $ABCD$ and P be a tetrahedron and a vertex. Let

$$v_A = 6\,\mathrm{vol}(PBCD) \qquad v_B = 6\,\mathrm{vol}(APCD) \tag{20}$$

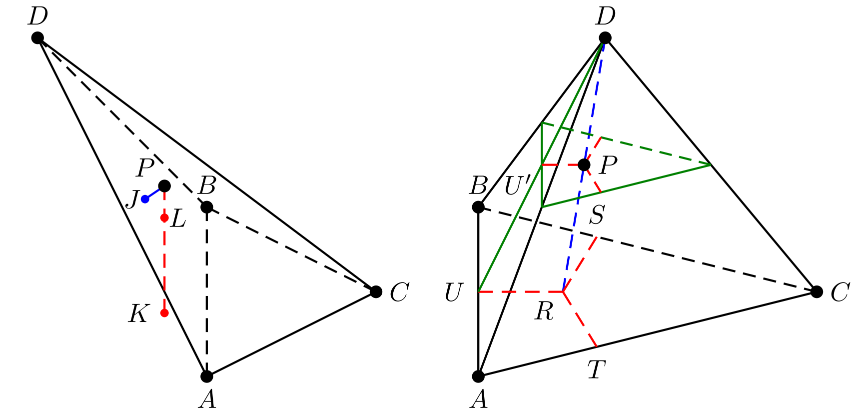
$$v_C = 6\,\mathrm{vol}(ABPD) \qquad v_D = 6\,\mathrm{vol}(ABCP) \tag{21}$$

An intersection occurs if all have the same sign. Since these volumes all have the same sign, the barycentric weights are robustly computed as

$$\gamma_A = \frac{v_A}{v_A + v_B + v_C + v_D} \qquad \gamma_B = \frac{v_B}{v_A + v_B + v_C + v_D} \tag{22}$$

$$\gamma_C = \frac{v_C}{v_A + v_B + v_C + v_D} \qquad \gamma_D = \frac{v_D}{v_A + v_B + v_C + v_D}. \tag{23}$$

If an intersection exists and no degeneracy is registered, then $|v_A|$, $|v_B|$, $|v_C|$, and $|v_D|$ can be bounded. We begin by bounding the distances from P to the face planes of the tetrahedron. In Figure 9(a), point P projects to point J in triangle ABD but a point K on plane ABC outside triangle ABC. The degeneracy assumption immediately gives $\ell(PJ) > \delta_-$. For the more difficult case, PK must intersect one of the other faces of the tetrahedron at L, so that $\ell(PK) \geq \ell(PL) > \delta_-$. Thus, the distance from P to any of the tetrahedron's bounding planes is larger that δ_-.



(a) Area bound for tetrahedron-vertex. (b) Area bound for tetrahedron-vertex.

Figure 9: Proof illustrations for tetrahedron vertex in 3D.

Next consider the setup in Figure 9(b), where DP intersects ABC at R. It can be seen that the distance $\ell(RU)$ from R to an edge of triangle ABC can be bounded as $\ell(RU) \geq \ell(PU') > \delta_-$, since P is bounded away from plane ABD. Similarly, $\ell(RS) > \delta_-$ and $\ell(RT) > \delta_-$. Using the same logic as Section 5.5.1, we conclude $2\,\text{area}(ABC) > 6\sqrt{3}\delta_-^2$. If P projects to K in the plane of ABC, then $|v_D| = 6\,|\text{vol}(ABCP)| = 2\,\ell(PK)\text{area}(ABC) > 6\sqrt{3}\delta_-^3$. The same bound is obtained for the other volumes. Thus, $\min(|v_A|, |v_B|, |v_C|, |v_D|) \geq 6\sqrt{3}\delta_-^3 = \rho_+$.

The logic for Algorithm 12 is very similar to that of Algorithm 11.

Algorithm 12 Tetrahedron Vertex 3D

function TETRAHEDRON_VERTEX(A, B, C, D, P)
  if Degenerate intersection **then**
    return (FALSE, $0, 0, 0, 0$)
  end if
  $v_A \leftarrow ((B - P) \times (C - P)) \cdot (D - P)$
  $v_B \leftarrow ((P - A) \times (C - A)) \cdot (D - A)$
  $v_C \leftarrow ((B - A) \times (P - A)) \cdot (D - A)$
  $v_D \leftarrow ((B - A) \times (C - A)) \cdot (P - A)$
  if $|v_A| \leq \rho$ **or** $|v_B| \leq \rho$ **or** $|v_C| \leq \rho$ **or** $|v_C| \leq \rho$ **then**
    return (FALSE, $0, 0, 0, 0$)
  end if
  if $\text{sgn}(v_A) \neq \text{sgn}(v_B)$ **or** $\text{sgn}(v_B) \neq \text{sgn}(v_C)$ **or** $\text{sgn}(v_C) \neq \text{sgn}(v_D)$ **then**
    return (FALSE, $0, 0, 0, 0$)
  end if
  return (TRUE, $\frac{v_A}{v_A+v_B+v_C+v_D}, \frac{v_B}{v_A+v_B+v_C+v_D}, \frac{v_C}{v_A+v_B+v_C+v_D}, \frac{v_D}{v_A+v_B+v_C+v_D}$)
end function

Floating point: $\mathcal{E}[v_A] \leq 47\epsilon L^3$. $\mathcal{E}[v_B] \leq 47\epsilon L^3$. $\mathcal{E}[v_C] \leq 47\epsilon L^3$. $\mathcal{E}[v_D] \leq 47\epsilon L^3$.

Tolerance constraints:

- $\rho_+ = 6\sqrt{3}\delta_-^3$
- $v_A > 0$, enforced by guaranteeing $|v_A| > \rho_-$ and requiring $\rho_- > \mathcal{E}[v_A] \geq 47\epsilon L^3$.

- $\min(\rho_+ - \rho, \rho - \rho_-) > 48\epsilon L^3$.
- Similar bounds for v_B, v_C, and v_D