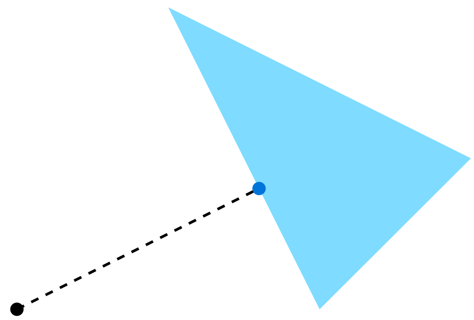
A General Solution to the Minimum Norm Point in a Simplex

Michael Greer
michael@kinematiclaboratories.com



1 Introduction

The Minimum-Norm-on-Simplex problem is concerned with determining which point contained within a simplex is closest to the origin of the space within which the simplex is embedded. The primary application of the solution to this problem has historically been the popular GJK algorithm [1], which is used to determine the distance between two convex polytopes.

The GJK algorithm originally relied on the "Distance Sub-Algorithm" to compute the minimum norm on a simplex. This method suffered from substantial numerical problems, required special treatment of edge cases, and did not eliminate sub-simplex computation when possible. The "Signed Volumes" method [2] improved upon this by addressing the numerical deficiency of the prior work and avoiding unnecessary computation of sub-simplices. Both of these algorithms present solutions to the Minimum-Norm-on-Simplex problem for simplices embedded in 3D space, since applications in rigid body simulations are of primary concern for the authors. Additionally, different cases for tetrahedrons, triangles, and lines are computed with unique algorithms for each.

In this paper, I present a general approach to this problem that computes a solution to any degree simplex embedded in any dimensional Euclidean space. This approach eliminates sub-simplex computation when possible, leading to an efficient computational structure. Furthermore, the algorithm may be formulated in a coordinate-free manner, leading to useful invariances of the solution to rotation and translation.

2 Algorithm Derivation

Consider a simplex of degree k embedded in a Euclidean space of dimension n. Note, that for cases where $k > n$ the simplex is necessarily degenerate, as the simplex necessarily has zero k-dimensional generalized volume. A degree k simplex has $k + 1$ vertices. Simplices are trivially convex, so any point within a simplex may be represented as a convex combination of the vertices:

$$
\begin{aligned}
p &= \sum_{i=1}^{k+1} \alpha_i v_i \\
\sum_{i=1}^{k+1} \alpha_1 &= 1 \\
\alpha_i &\in [0, 1]
\end{aligned}
\qquad (1)
$$

The vertices can be arranged into a matrix and the α elements into a vector to yield a more compact expression:

$$\begin{gathered} V = [v_1, v_2, ..., v_{k+1}] \\ \alpha = [\alpha_1, \alpha_2, ..., \alpha_{k+1}]^T \\ p = V \cdot \alpha \end{gathered} \tag{2}$$

The minimum-norm point in this set is denoted p^*, and the corresponding set of convex combination parameters are denoted α^*. These elements may be expressed using the canonical optimization representation:

$$\begin{aligned} p^* = \arg\min_p \quad & p^T \cdot p \\ \text{subject to: } & p = V \cdot \alpha \\ & 1_{k+1}^T \cdot \alpha = 1 \\ & \alpha_i \geq 0 \end{aligned} \tag{3}$$

Substituting the expression for p gives:

$$\begin{aligned} \alpha^* = \arg\min_\alpha \quad & \alpha^T V^T V \alpha \\ \text{subject to: } & 1_{k+1}^T \cdot \alpha = 1 \\ & \alpha_i \geq 0 \end{aligned} \tag{4}$$

Where 1_{k+1} is a vector of length $(k+1)$ where each element is 1. Observe that if $Q_V = V^T V$, the objective function takes on a familiar form:

$$\alpha^T Q_V \alpha \tag{5}$$

Q_V is trivially positive semi-definite and symmetric. Consider that the value of eq. (5) represents the Euclidean distance of the convex combination from the origin. The distance can only be greater than or equal to zero; therefore, the result of eq. (5) must be greater than or equal to zero. eq. (5) is also symmetric since $\left(V^T V\right)^T = V^T V$. Thus, this optimization problem is a quadratic program with linear equality and inequality constraints.

Consider a relaxed version of the problem with the inequality constraint omitted. This simplified problem may be solved directly as a system of linear equations. This version of the problem may be interpreted as computing the closest point to the origin in the space spanned by the edges of the simplex. For example, applying this process to a 2-Simplex (i.e. a triangle) embedded in 3D Euclidean space yields the closest point to the origin on the plane spanned by the triangle, but the point is not necessarily located within the triangle. This relaxed optimization problem is:

$$\begin{aligned} \alpha^* = \arg\min_\alpha \quad & \alpha^T Q_V \alpha \\ \text{subject to: } & 1_{k+1}^T \cdot \alpha = 1 \end{aligned} \tag{6}$$

Because this is a quadratic program with linear equality constraints, it may be solved using the linear system formed by the KKT conditions:

$$\begin{pmatrix} Q_V & 1_{k+1} \\ 1_{k+1}^T & 0 \end{pmatrix} \begin{pmatrix} \alpha^* \\ \lambda^* \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \tag{7}$$

It is trivial to determine if the resulting point is inside the simplex: if all values of α^* are non-negative, the result is a valid convex combination and the point is within the simplex. Otherwise, the point is not within the simplex. If the resulting point is within the simplex, it is the minimum-norm point in the simplex. If it is not within the simplex, the minimum-norm point must lie on one of the facets (boundary simplices) of the original simplex. Conveniently, this means it is sufficient to calculate the minimum-norm point on each of these boundary simplices and select the result with the shortest distance to the origin. Because the above approach generalizes to any simplex of any degree, the same approach may be applied to the boundary. Each boundary simplex is formed by omitting one vertex from the original simplex. The boundary of a simplex of degree k consists of $k + 1$ simplices of degree $k - 1$. This yields a compact, recursive algorithm.

<u>MinimumNormOnSimplex</u>(V):

let $Q_V \leftarrow V^T V$

let $\begin{pmatrix} \alpha^* \\ \lambda^* \end{pmatrix} \leftarrow \begin{pmatrix} Q_V & 1_{k+1} \\ 1_{k+1}^T & 0 \end{pmatrix}^{-1} \begin{pmatrix} 0 \\ 1 \end{pmatrix}$

if $\alpha^* \geq 0$:

  return α^*

else:

  return $\min\left\{ \text{MinimumNormOnSimplex}\left(\frac{V}{v_i}\right) \mid \forall i \in 1..k+1 \right\}$

3 Boundary Elimination

If the origin is not contained within the simplex, the algorithm moves on to the boundary simplices. However, it is not necessary to check every single boundary simplex: only a subset of these need to be considered. Because each boundary simplex can be formed by removing one vertex from the current simplex, these form pairs:

$$\left(\frac{V}{v_i}, v_i \right) \tag{8}$$

Each v_i corresponds to a α_i. Because the origin is not contained in the current simplex, at least one of these α_i^* must be negative. The only boundary simplices the algorithm needs to consider are:

$$\left\{ \frac{V}{v_i} \mid \alpha_i^* < 0 \right\} \tag{9}$$

Admittedly, I do not have the expertise to rigorously prove this; the above lemma has been tested exhaustively and omitting parts of the boundary in this way has yet to yeild an incorrect result in the millions of trial runs.

4 Coordinate-Free Formulation

Consider the elements of matrix Q_V. The element at row i and column j is equal to the dot product $v_i^T v_j$. This product is denoted $v_{i,j}$. So:

$$Q_V = \begin{pmatrix} v_{1,1} & v_{1,2} & \cdots & v_{1,k+1} \\ v_{2,1} & v_{2,2} & \cdots & v_{2,k+1} \\ \vdots & \vdots & \ddots & \vdots \\ v_{k+1,1} & v_{k+1,2} & \cdots & v_{k+1,k+1} \end{pmatrix} \tag{10}$$

I also define $d_{i,j}$ to be the Euclidean distance between v_i and v_j and d_i as the Euclidean distance between v_i and the origin. By the definition of the Euclidean norm:

$$\begin{aligned} d_i^2 &= v_i^T v_i \\ d_{i,j}^2 &= \left(v_i - v_j\right)^T \left(v_i - v_j\right) \end{aligned} \tag{11}$$

Distributing and rearranging the second line:

$$\begin{aligned} d_{i,j}^2 &= \left(v_i^T - v_j^T\right)\left(v_i - v_j\right) \\ &= v_i^T v_i - v_i^T v_j - v_j^T v_i + v_j^T v_j \\ &= d_i^2 - 2v_{i,j} + d_j^2 \\ 2v_{ij} &= d_i^2 + d_j^2 - d_{i,j}^2 \\ v_{ij} &= \frac{1}{2}\left(d_i^2 + d_j^2 - d_{i,j}^2\right) \end{aligned} \tag{12}$$

Substituting these results back into eq. (10):

$$Q_V = \begin{pmatrix} d_1^2 & \frac{1}{2}\left(d_1^2 + d_2^2 - d_{1,2}^2\right) & \dots & \frac{1}{2}\left(d_1^2 + d_{k+1}^2 - d_{1,k+1}^2\right) \\ \frac{1}{2}\left(d_2^2 + d_1^2 - d_{2,1}^2\right) & d_2^2 & \dots & \frac{1}{2}\left(d_2^2 + d_{k+1}^2 - d_{2,k+1}^2\right) \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{2}\left(d_{k+1}^2 + d_1^2 - d_{k+1,1}^2\right) & \frac{1}{2}\left(d_{k+1}^2 + d_2^2 - d_{k+1,2}^2\right) & \dots & d_{k+1}^2 \end{pmatrix} \tag{13}$$

Thus, the optimization problem may be expressed entirely in terms of the Euclidean distances between the vertices and the origin. Elements of this matrix are invariant to the location of the vertices of the simplex in space. If the simplex is transformed relative to the origin, the only values that change are the distances of each vertex with respect to the origin, not the distances between vertices of the simplex.

In future work, it is worth exploring if this distance-only formulation could be generalized to non-Euclidean spaces of constant curvature.

5 Numerical Consideration

The primary numerical consideration that must be addressed in this algorithm is the occurrence of degenerate simplices, or simplices of zero generalized volume. While a truly zero-volume simplex may be rare in a continuous context, algorithms working with floating-point numbers suffer innacuracies even in near-degenerate cases.

One approach to dealing with this numerical problem is to calculate the enclosed volume of the simplex directly. Computing the Cayley-Menger determinate [3] offers a generalized approach to calculating the volume of a simplex of any degree, embedded in space of any dimension. This formulation also relies entirely on the distances between vertices and not their actual coordinate-frame locations. Thus, the calculated distances may be re-used to form the matrix Q_V, avoiding extra computation.

When a simplex is degenerate, V drops rank, and so too does the corresponding Q_V matrix. Many approaches for solving systems of linear equations offer built-in methods to check for low-rank matrices. For example, computing the matrix inversion via the singular value decomposition (SVD) allows the algorithm to verify that the minimum singular value is above some machine precision value.

If the simplex has zero volume, it necessarily cannot contain the origin and the minimum-norm point must lie on its boundary. If the simplex is degenerate, the necessary information to inform the aforementioned boundary simplex elimination outlined above cannot be computed. In this case, the minimum-norm point on every boundary simplex must be computed. In the most degenerate case (a simplex consisting of all collocated vertices), every boundary simplex will be computed.

6 Algorithm

Integrating all of the proposed enhancements gives:

$\underline{\text{MinimumNormOnSimplex}}(V)$:

let $Q_V \leftarrow V^T V$

let $U \Sigma V^T \leftarrow \text{SVD}\left(\begin{pmatrix} Q_V & 1_{k+1} \\ 1_{k+1}^T & 0 \end{pmatrix}\right)$

let $\begin{pmatrix} \alpha^* \\ \lambda^* \end{pmatrix} \leftarrow V \Sigma^{-1} U^T \begin{pmatrix} 0 \\ 1 \end{pmatrix}$

if $\alpha^* \geq 0$:

  return α^*

else if $\min(\Sigma) < \varepsilon$:

  return $\min\left\{ \text{MinimumNormOnSimplex}\left(\frac{V}{v_i}\right) \mid \forall i \in 1..k+1 \right\}$

else:

  return $\min\left\{ \text{MinimumNormOnSimplex}\left(\frac{V}{v_i}\right) \mid \alpha_i^* < 0 \right\}$

7 Results

I implemented the presented algorithm using the Rust programming language, and the code is freely available and open source here.

7a Sampling Random Simplices on Hyperspheres

In 1992, the 53rd Putnam Competition presented examinees with the following question: *"Four points are chosen independently and at random on the surface of a sphere (using the uniform distribution). What is the probability that the center of the sphere lies inside the resulting tetrahedron?"* [4] The answer to this question is $\frac{1}{8}$. In general, if $k+1$ points are sampled on the surface of a $k-1$-sphere, they form a simplex of degree k. The probability that this sampled simplex contains the center of the hypersphere is:

$$\frac{1}{2^{k+1}} \tag{14}$$

We can verify this result empirically by randomly sampling points on a hypersphere centered at the origin and computing the minimum norm point on that simplex. If the minimum-norm point is exactly the origin, the simplex contains the origin and thus the center of the circle.

The probability of any one sample containing the origin may be estimated by performing a Monte-Carlo simulation. The probability that any one simplex contains the origin is the number of samples containing the origin divided by the total number of samples. The table below presents the estimated probability for multiple dimensions k.

Dimension	Analytical Result	Empirical Result	Error Rate
2	0.25	0.2488	0.0012
3	0.125	0.12593	0.00093
4	0.0625	0.06317	0.00067
5	0.03125	0.03182	0.0057
6	0.015625	0.01608	0.000455

7b Computing Distances Between Hyperspheres

This generalized simplex solver may be used as part of the GJK algorithm to compute the closest points between two convex shapes. One particularly challenging scenario occurs when shapes are very nearly touching. The table below lists the accuracy of this algorithm for two 3-spheres as they are brought closer and closer together.

Analytical Distance	Calculated Distance	Error	Percent Error	Iterations
1.0	1.0000000155117723	1.5511772311072036e-8	1.5511772311072036e-6	7
0.1	0.10000021515200824	2.1515200815391822e-7	2.1515200815391822e-4	14
0.01	0.01000012428887733	1.2428887732028604e-7	1.2428887732028604e-3	22
0.001	0.001000435898786521	4.358987866310768e-7	4.358987866310768e-2	22
0.0001	0.00010971798612974	9.717986129533827e-6	9.717986129533827e0	18

The absolute distance error remains similar across different distances, but the error, relative to the analytical distance, increases rapidly. This is a relatively well-known issue with the GJK algorithm when applied to continuous shapes like spheres and capsules. When GJK is applied to discrete polytopes, this problem is mitigated significantly. Users should consider employing analytical approaches to compute distances between continuous shapes and GJK when computing distances in pairs involving discrete polytopes.

8 Conclusion

In this paper, I presented a novel solution to the Minimum-Norm-on-Simplex problem for any degree simplex embedded in any dimension that is coordinate-free and robust to degenerate simplices. I also provided a software implementation and several examples demonstrating the algorithm.

In the future, the coordinate-free formulation may lend itself to extension to non-Euclidean spaces of constant curvature. Additionally, I plan to explore hardware acceleration strategies for this algorithm, since the branch-free formulation may lend itself to SIMD hardware.

Bibliography

[1] E. G. Gilbert, D. W. Johnson, and S. S. Keerthi, “A fast procedure for computing the distance between complex objects in three-dimensional space,” *IEEE Journal on Robotics and Automation*, vol. 4, no. 2, pp. 193–203, 2002.

[2] M. Montanari, N. Petrinic, and E. Barbieri, “Improving the GJK algorithm for faster and more reliable distance queries between convex objects,” *ACM Transactions on Graphics (TOG)*, vol. 36, no. 3, pp. 1–17, 2017.

[3] T. F. Havel, “Some examples of the use of distances as coordinates for Euclidean geometry,” *Journal of Symbolic Computation*, vol. 11, no. 5–6, pp. 579–593, 1991.

[4] J. Scholes, “53rd Putnam 1992.” [Online]. Available: https://prase.cz/kalva/putnam/psoln/psol926.html