# A Fine Way to Kiss the Hyperellipsoid

Kurt A. Motekew

November 28, 2022

**Abstract**

A closed form solution determining the intersection of a pointing vector from a reference point to an arbitrarily oriented hyperellipsoid is presented. The tangent to the hyperellipsoid from a line originating at the reference point, in the direction of the pointing vector, is also derived. The solutions are obtained by transforming the hyperellipsoid into a unit $n$-sphere, followed by a second transformation converting the geometry to a unit circle.

Representation of a hyperellipsoid as a unit $n$-sphere subject to an affine transformation is first illustrated. Next, the process of mapping the three dimensional intersection and tangent point problems to two dimensions is described. This method is then extended to n-dimensional space through the generation of an orthonormal transformation via the tensor calculus definition of the cross product. The two dimensional solutions are derived, concluding with justifying why the intersection and tangent points remain valid through the application of an affine transformation.

## 1 Introduction

The quadratic form

$$\mathbf{z}^{\mathrm{T}}\mathrm{M}^{-1}\mathbf{z} = 1$$

represents an $n$-dimensional hyperellipsoid centered at the origin of a Cartesian reference frame when M is a real symmetric positive definite matrix and $\mathbf{z}$ is a vector indicating a position on the corresponding surface. M has $n$ real, not necessarily distinct, nonzero eigenvalues $\lambda_i$ and $n$ corresponding distinct orthogonal eigenvectors $\mathbf{v}_i$ such that

$$\mathrm{M} = \mathrm{V}\Lambda\mathrm{V}^{\mathrm{T}}$$

where

$$\Lambda = \begin{bmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \ddots & \\ & & & \lambda_n \end{bmatrix}$$

and

$$\mathrm{V} = \begin{bmatrix} \mathbf{v}_1 & \mathbf{v}_2 & \cdots & \mathbf{v}_n \end{bmatrix}$$

Normalizing the eigenvectors

$$\hat{\mathbf{u}}_i = \frac{\mathbf{v}_i}{||\mathbf{v}_i||}$$

and denoting the corresponding orthonormal transformation matrix as

$$\mathrm{U} = \begin{bmatrix} \hat{\mathbf{u}}_1 & \hat{\mathbf{u}}_2 & \cdots & \hat{\mathbf{u}}_n \end{bmatrix}$$

the diagonal matrix $\Lambda$ can be viewed as a hyperellipsoid with principal axes aligned with the Cartesian reference frame subject to a rigid rotation (vs. a reference frame transformation) resulting in M:

$$\mathrm{M} = \mathrm{U}\Lambda\mathrm{U}^{\mathrm{T}}$$

Additionally, the hyperellipsoid $\Lambda$ is the result of applying an affine transformation to the identity matrix

$$\Lambda = \Lambda^{\frac{1}{2}}\mathrm{I}\Lambda^{\frac{1}{2}}$$

Consequently, an arbitrarily oriented hyperellipsoid can be represented as a unit $n$-sphere subject to a diagonal scaling matrix followed by an orthonormal transformation:

$$\begin{aligned} \mathrm{T} &= \mathrm{U}\Lambda^{\frac{1}{2}} \\ \mathrm{M} &= \mathrm{T}\mathrm{I}\mathrm{T}^{\mathrm{T}} \end{aligned}$$

Lanczos [5, Ch. 2] elegantly develops the relationship between hyperellipsoids and symmetric matrices, deriving eigenvalues as the squares of semiaxes lengths and eigenvectors as the principal axes. The concept of the hyperellipsoid as a linear transformation of the unit $n$-sphere is thoroughly explored. Gura and Gersten [4] derive analytic expressions related to the probability of containment of a random variable through the use of affine transformations of an n-dimensional normal density function.

## 2 Ellipsoid to Unit Sphere

A transformation is applied to Cartesian coordinates, mapping the ellipsoid to a unit sphere. In matrix form, it can be constructed by the combination of a reference frame transformation (orthonormal matrix) aligning the ellipsoid principal axes with that of a new Cartesian reference frame, and a diagonal matrix scaling the ellipsoid to a unit sphere[1]. If the ellipsoid semiaxes, as vectors $\mathbf{a}_i$ with lengths $a_i$ and unit principal axes $\hat{\mathbf{a}}_i$,

$$\begin{aligned} \mathbf{a}_1 &= a_1\hat{\mathbf{a}}_1 \\ \mathbf{a}_2 &= a_2\hat{\mathbf{a}}_2 \\ \mathbf{a}_3 &= a_3\hat{\mathbf{a}}_3 \end{aligned}$$

[1]If the ellipsoid is not located at the origin of the reference frame, then a simple translation would first be applied.

are known from the geometry of the problem, the transformation matrix resulting in a unit sphere is

$$\mathrm{T}_e^s = \begin{bmatrix} \frac{1}{a_1} & 0 & 0 \\ 0 & \frac{1}{a_2} & 0 \\ 0 & 0 & \frac{1}{a_3} \end{bmatrix} \begin{bmatrix} \hat{\mathbf{a}}_1^{\mathrm{T}} \\ \hat{\mathbf{a}}_2^{\mathrm{T}} \\ \hat{\mathbf{a}}_3^{\mathrm{T}} \end{bmatrix} \tag{1}$$

If the ellipsoid is defined as a symmetric positive definite matrix, eigendecomposition will yield the eigenvalues $\lambda_i$ and corresponding eigenvectors (in unit vector form) $\hat{\mathbf{a}}_i$ such that

$$\mathrm{T}_e^s = \begin{bmatrix} \frac{1}{\sqrt{\lambda_1}} & 0 & 0 \\ 0 & \frac{1}{\sqrt{\lambda_2}} & 0 \\ 0 & 0 & \frac{1}{\sqrt{\lambda_3}} \end{bmatrix} \begin{bmatrix} \hat{\mathbf{a}}_1^{\mathrm{T}} \\ \hat{\mathbf{a}}_2^{\mathrm{T}} \\ \hat{\mathbf{a}}_3^{\mathrm{T}} \end{bmatrix} \tag{2}$$

given the semiaxes lengths are

$$\begin{aligned} a_1 &= \sqrt{\lambda_1} \\ a_2 &= \sqrt{\lambda_2} \\ a_3 &= \sqrt{\lambda_3} \end{aligned}$$

# 3 Unit Sphere to Circle

The location of the reference point (R) relative to the origin (O), w.r.t. the Cartesian reference frame in which the centered ellipsoid is originally represented ($e$), is $\mathbf{z}^e_{\mathrm{R/O}}$. The pointing vector originating from the reference point is $\mathbf{z}^e_{\mathrm{P/R}}$. These vectors can be transformed to the reference frame in which the ellipsoid is a unit sphere using the result of §2:

$$\begin{aligned} \mathbf{z}^s_{\mathrm{R/O}} &= \mathrm{T}_e^s \mathbf{z}^e_{\mathrm{R/O}} \\ \mathbf{z}^s_{\mathrm{P/R}} &= \mathrm{T}_e^s \mathbf{z}^e_{\mathrm{P/R}} \end{aligned}$$

A second transformation simplifies the geometry to that of a circle. A plane cutting through the unit sphere preserving all the information needed to determine intersection and tangent points is defined by $\mathbf{z}^s_{\mathrm{R/O}}$ and $\mathbf{z}^s_{\mathrm{P/R}}$. The cross product of these two vectors generates the third axis of the basis vectors defining the transformation of the unit sphere into a unit circle,

$$\begin{aligned}
\hat{\mathbf{i}} &= \frac{\mathbf{z}^s_{\mathrm{R/O}}}{\left\|\mathbf{z}^s_{\mathrm{R/O}}\right\|} && (3)\\
\hat{\mathbf{k}} &= \frac{\hat{\mathbf{i}} \times \mathbf{z}^s_{\mathrm{P/R}}}{\left\|\mathbf{z}^s_{\mathrm{P/R}}\right\|} && (4)\\
\hat{\mathbf{j}} &= \hat{\mathbf{k}} \times \hat{\mathbf{i}} && (5)
\end{aligned}$$

such that,

$$\mathrm{T}^c_s = \begin{bmatrix} \hat{\mathbf{i}}^{\mathrm{T}} \\ \hat{\mathbf{j}}^{\mathrm{T}} \\ \hat{\mathbf{k}}^{\mathrm{T}} \end{bmatrix} \tag{6}$$

Take note of the general process—the reference point location forms the first basis vector and is then used with the pointing vector to find the normal to the plane in which the unit circle of interest is defined. That third basis vector is then used with the first to determine the second basis vector. The same process is followed in higher dimensions except $n-2$ mutually orthogonal $n$-dimensional vectors, each normal to the plane containing the unit circle being isolated, must be determined before going back and "squaring up" the system that will be used to form an orthonormal transformation matrix.

Application of $\mathrm{T}^c_s$ brings all relevant information into the first two dimensions.

$$\begin{aligned}
\mathbf{z}^c_{\mathrm{R/O}} &= \mathrm{T}^c_s \mathbf{z}^s_{\mathrm{R/O}}\\
\mathbf{z}^c_{\mathrm{P/R}} &= \mathrm{T}^c_s \mathbf{z}^s_{\mathrm{P/R}}
\end{aligned}$$

The two dimensional reference point and pointing vectors will be denoted as $\mathbf{r}$ and $\mathbf{p}$, respectively. The third component of the three dimensional vectors becomes zero after the transformation.

$$\begin{aligned}
\mathbf{z}^c_{\mathrm{R/O}} &= \left[\mathbf{r}^{\mathrm{T}}\ 0\right]^{\mathrm{T}}\\
\mathbf{z}^c_{\mathrm{P/R}} &= \left[\mathbf{p}^{\mathrm{T}}\ 0\right]^{\mathrm{T}}
\end{aligned}$$

# 4 Extension to Arbitrary Dimension

Transformation of the hyperellipsoid to a unit circle parallels that of the three dimensional case. The conversion to a unit $n$-sphere is identical. Given the semiaxis definition,

$$\mathbf{a}_i = a_i \hat{\mathbf{a}}_i$$

the transformation to an $n$-sphere is an extension of Eqn. (1)

$$\mathrm{T}_e^s = \begin{bmatrix} \frac{1}{a_1} & & & \\ & \frac{1}{a_2} & & \\ & & \ddots & \\ & & & \frac{1}{a_n} \end{bmatrix} \begin{bmatrix} \hat{\mathbf{a}}_1^{\mathrm{T}} \\ \hat{\mathbf{a}}_2^{\mathrm{T}} \\ \vdots \\ \hat{\mathbf{a}}_n^{\mathrm{T}} \end{bmatrix} \tag{7}$$

Likewise, eigendecomposition of an $n$-dimensional covariance naturally extends beyond the three dimensions of Eqn. (2):

$$\mathrm{T}_e^s = \begin{bmatrix} \frac{1}{\sqrt{\lambda_1}} & & & \\ & \frac{1}{\sqrt{\lambda_2}} & & \\ & & \ddots & \\ & & & \frac{1}{\sqrt{\lambda_n}} \end{bmatrix} \begin{bmatrix} \hat{\mathbf{a}}_1^{\mathrm{T}} \\ \hat{\mathbf{a}}_2^{\mathrm{T}} \\ \vdots \\ \hat{\mathbf{a}}_n^{\mathrm{T}} \end{bmatrix} \tag{8}$$

The $n$-dimensional Cartesian reference and pointing vectors can now be transformed to a reference frame in which the hyperellipsoid becomes an $n$-sphere.

$$\begin{aligned} \mathbf{z}^s_{\mathrm{R/O}} &= \mathrm{T}_e^s \mathbf{z}^e_{\mathrm{R/O}} \\ \mathbf{z}^s_{\mathrm{P/R}} &= \mathrm{T}_e^s \mathbf{z}^e_{\mathrm{P/R}} \end{aligned}$$

Isolating the two dimensional unit circle is accomplished through the use of the generalized cross product definition to create the orthonormal transformation $\mathrm{T}_s^c$. This is most easily defined via tensor notation. The three dimensional,

$$W^i = \varepsilon^{ijk} U_j V_k$$

and $n$-dimensional cross product definitions [3, pp. 150–157],

$$V^i = \varepsilon^{ij_1 \ldots j_{n-1}} U_{j_1} \ldots U_{j_{n-1}}$$

are simplified in Cartesian coordinates by the replacement of the Levi-Civita symbol $\varepsilon^{ijk}$ with the permutation symbol $e^{ijk}$ and the equivalence of contravariant ($U^i$) and covariant ($U_i$) vector components. Retaining the Einstein summation notation for convenience,

$$\begin{aligned} W^i &= e^{ijk} U_j V_k \\ V^i &= e^{ij_1 \ldots j_{n-1}} U_{j_1} \ldots U_{j_{n-1}} \end{aligned} \tag{9}$$

The $n$-dimensional cross product takes $(n-1)$ $n$-dimensional vectors, producing an $n^{th}$ vector that is orthogonal to each of the input vectors. This presents a challenge when attempting to construct $\mathrm{T}_s^c$ given only the two arbitrarily oriented vectors $\mathbf{z}^s_{\mathrm{R/O}}$ and $\mathbf{z}^s_{\mathrm{P/R}}$ are provided. These two vectors are used to generate $(n-2)$ mutually orthogonal $n$-dimensional vectors defining the "*group normal*" to the plane they span. Having fixed the geometry, one of the original vectors is selected to be a basis vector in the new reference frame while the other is sacrificed to square up the system.

To begin the process, $(n-3)$ orthogonal vectors must be chosen that are linearly independent of $\mathbf{z}^s_{\mathrm{R/O}}$ and $\mathbf{z}^s_{\mathrm{P/R}}$. A simple method is to choose Cartesian basis vectors $\hat{\mathbf{u}}_i$ for the $n$-dimensional system that are aligned with the axes of the reference frame in use. Begin by selecting the first candidate

$$\hat{\mathbf{u}}_1 = [1\ 0\ 0\ \cdots\ 0]^{\mathrm{T}}$$

If $\hat{\mathbf{u}}_1$ is linearly independent of $\mathbf{z}^s_{\mathrm{R/O}}$ and $\mathbf{z}^s_{\mathrm{P/R}}$, then it becomes $U_{i_3}$ of Eqn. (9). If $\hat{\mathbf{u}}_1$ is within the span of $\mathbf{z}^s_{\mathrm{R/O}}$ and $\mathbf{z}^s_{\mathrm{P/R}}$, then examine the next basis vector,

$$\hat{\mathbf{u}}_2 = [0\ 1\ 0\ \cdots\ 0]^{\mathrm{T}}$$

Continue this process until a linearly independent basis vector is determined. Repeat the process, using the next available Cartesian basis vector $\hat{\mathbf{u}}_k$, until the $(n-1)^{th}$ orthonormal basis vector has been found (Algorithm 1). Given the existence of $n$ Cartesian basis vectors $\hat{\mathbf{u}}_1, \hat{\mathbf{u}}_2, \ldots, \hat{\mathbf{u}}_n$, and two initial vectors,

$$\begin{aligned} U_{i_1} &= \mathbf{z}^s_{\mathrm{R/O}} \\ U_{i_2} &= \mathbf{z}^s_{\mathrm{P/R}} \end{aligned}$$

$(n-3)$ orthonormal vectors $U_{i_3} \cdots U_{i_{n-1}}$ can be determined such that $U_{i_j}$ are linearly independent.

The cross product of $\mathbf{z}^s_{\mathrm{R/O}}$, $\mathbf{z}^s_{\mathrm{P/R}}$, and $U_{j_3} \ldots U_{j_{n-1}}$, in that order, generates an $n^{th}$ vector orthogonal to both $\mathbf{z}^s_{\mathrm{R/O}}$ and $\mathbf{z}^s_{\mathrm{P/R}}$. This vector replaces $U_{j_3}$. The cross product operation is performed again, resulting in a replacement to $U_{j_4}$. This process is repeated until $U_{j_{n-1}}$ replaces itself. The cross product is carried out once more, generating $U_{j_n}$. This completes the group of vectors normal to the plane formed by $\mathbf{z}^s_{\mathrm{R/O}}$ and $\mathbf{z}^s_{\mathrm{P/R}}$.

At this point, $n$ vectors exist. $U_{j_3} \ldots U_{j_n}$ are orthogonal to $\mathbf{z}^s_{\mathrm{R/O}}$ and $\mathbf{z}^s_{\mathrm{P/R}}$. However, $\mathbf{z}^s_{\mathrm{R/O}}$ is most likely not orthogonal to $\mathbf{z}^s_{\mathrm{P/R}}$. The final step of squaring up the system to produce an orthonormal transformation parallels the process of §3. To review, the reference point location was used to form the first basis vector; Eqn. (3). The cross product of this first basis vector and the pointing vector resulted in the third basis vector; Eqn. (4). This basis vector is normal to the plane in which the unit circle of interest is defined—it essentially defines the geometry. The third basis vector was then crossed with the first to produce the second; Eqn. (5).

In $n$-dimensions, instead of generating a single basis vector orthogonal to the plane isolating the unit circle containing $\mathbf{z}^s_{\mathrm{R/O}}$ and $\mathbf{z}^s_{\mathrm{P/R}}$, $(n-2)$ are generated[2]. $U_{j_3} \ldots U_{j_n}$ must be crossed against the first basis vector $\mathbf{z}^s_{\mathrm{R/O}}$ $(U_{j_1})$ to form the

[2] An $n$-dimensional plane has $(n-2)$ linearly independent (noncollinear) normals.

**Algorithm 1** Given two $n$-dimensional vectors, build $(n-3)$ orthonormal vectors that are linearly independent of the inputs, such that the entire set of $(n-1)$ vectors is suitable for the cross product operation.

**procedure** BuildCross($\mathbf{z}_1, \mathbf{z}_2$)<br>
    $n = \texttt{length}\,(\mathbf{z}_i)$<br>
    $\delta_i = 2$<br>
    **for** $i = 3 : (n-1)$ **do**<br>
        $done = false$<br>
        **while** $done \neq true$ **do**<br>
            $\hat{\mathbf{u}}_i\,[1:n] = 0$<br>
            $\hat{\mathbf{u}}_i\,[i-\delta_i] = 1$<br>
            **if** $\texttt{rank}\,(\mathbf{z}_1, \mathbf{z}_2, \hat{\mathbf{u}}_i) == 3$ **then**<br>
                $done = true$<br>
            **else**<br>
                $\delta_i = \delta_i - 1$<br>
            **end if**<br>
        **end while**<br>
    **end for**<br>
    **return** $\hat{\mathbf{u}}_3, \hat{\mathbf{u}}_4, \ldots, \hat{\mathbf{u}}_{n-1}$<br>
**end procedure**

second (Algorithm 2). The $n$ basis vectors can now be normalized and set as row vectors forming,

$$\mathrm{T}_s^c = \begin{bmatrix} \hat{\mathbf{u}}_1^{\mathrm{T}} \\ \hat{\mathbf{u}}_2^{\mathrm{T}} \\ \vdots \\ \hat{\mathbf{u}}_n^{\mathrm{T}} \end{bmatrix} \tag{10}$$

with

$$\begin{aligned} \mathbf{u}_i &= U_{j_i} \\ \hat{\mathbf{u}}_i &= \frac{\mathbf{u}_i}{||\mathbf{u}_i||} \end{aligned}$$

As with the three dimensional case, $\mathrm{T}_s^c$ isolates all relevant information to be in two dimensions.

$$\begin{aligned} \mathbf{z}^c_{\mathrm{R/O}} &= \mathrm{T}_s^c \mathbf{z}^s_{\mathrm{R/O}} \\ \mathbf{z}^c_{\mathrm{P/R}} &= \mathrm{T}_s^c \mathbf{z}^s_{\mathrm{P/R}} \end{aligned}$$

Once again, only the first two components of the reference and pointing vectors are nonzero,

$$\begin{aligned} \mathbf{z}^c_{\mathrm{R/O}} &= [\mathbf{r}^{\mathrm{T}}\ 0\ \cdots\ 0]^{\mathrm{T}} \\ \mathbf{z}^c_{\mathrm{P/R}} &= [\mathbf{p}^{\mathrm{T}}\ 0\ \cdots\ 0]^{\mathrm{T}} \end{aligned}$$

where

$$\mathbf{r} = \begin{bmatrix} r_1 \\ r_2 \end{bmatrix} \tag{11}$$

$$\mathbf{p} = \begin{bmatrix} p_1 \\ p_2 \end{bmatrix} \tag{12}$$

In practice, it is best to normalize each $U_{i_j}$ as they are assigned since the components of higher dimensional cross products can begin to grow quickly in magnitude. Higher dimensional cross products generated through a direct implementation of Eqn. (9) become less numerically stable as the range in magnitudes of the vector components grows. The orthogonality of the resulting cross product begins to suffer. Appendix A describes the use of full QR decomposition as a stable and efficient method of forming the set of basis vectors normal to the span of the position and pointing vectors.

---

**Algorithm 2** Given two $n$-dimensional vectors, determine an orthonormal transformation from the current reference frame to one in which the first basis vector of the new frame is aligned with the first input vector, and the second basis vector is within the span (in the plane defined by) the input vectors. Note, for even dimensions, the transformation will be left handed (not orientation preserving). The first two rows can be swapped to restore right handedness if necessary.

---

**procedure** NDIMTO2D$(\mathbf{z}_1, \mathbf{z}_2)$
  $n = \texttt{length}\,(\mathbf{z}_i)$
  $\hat{\mathbf{u}}_1 = \frac{\mathbf{z}_1}{||\mathbf{z}_1||}$
  $\hat{\mathbf{z}}_2 = \frac{\mathbf{z}_2}{||\mathbf{z}_2||}$
  $\hat{\mathbf{u}}_3, \hat{\mathbf{u}}_4, \ldots, \hat{\mathbf{u}}_{n-1} = \textsc{BuildCross}\,(\hat{\mathbf{u}}_1, \hat{\mathbf{z}}_2, )$
  **for** $i = 3 : (n-1)$ **do**
    $\mathbf{u}_i = \textsc{Cross}\,(\hat{\mathbf{u}}_1, \hat{\mathbf{z}}_2, \hat{\mathbf{u}}_3, \hat{\mathbf{u}}_4, \ldots, \hat{\mathbf{u}}_{n-1})$
    $\hat{\mathbf{u}}_i = \frac{\mathbf{u}_i}{||\mathbf{u}_i||}$
  **end for**
  $\mathbf{u}_n = \textsc{Cross}\,(\hat{\mathbf{u}}_1, \hat{\mathbf{z}}_2, \hat{\mathbf{u}}_3, \hat{\mathbf{u}}_4, \ldots, \hat{\mathbf{u}}_{n-1})$
  $\hat{\mathbf{u}}_n = \frac{\mathbf{u}_n}{||\mathbf{u}_i||}$
  $\mathbf{u}_2 = \textsc{Cross}\,(\hat{\mathbf{u}}_3, \hat{\mathbf{u}}_4, \ldots, \hat{\mathbf{u}}_n, \hat{\mathbf{u}}_1)$  $\triangleright\ \hat{\mathbf{j}} = \hat{\mathbf{k}} \times \hat{\mathbf{i}}$ (Eqn. 4)
  $\hat{\mathbf{u}}_2 = \frac{\mathbf{u}_2}{||\mathbf{u}_2||}$
  $\mathrm{T} = [\hat{\mathbf{u}}_1, \hat{\mathbf{u}}_2, \ldots, \hat{\mathbf{u}}_n]^{\mathrm{T}}$
  **return** T
**end procedure**

---

# 5 Intersecting the Circle

Given a reference point $\mathbf{r}$ and pointing vector $\mathbf{p}$ from Eqns. (11) and (12), the intersection $\boldsymbol{\rho}$ shown in Figure 1 is determined by finding the roots of a quadratic

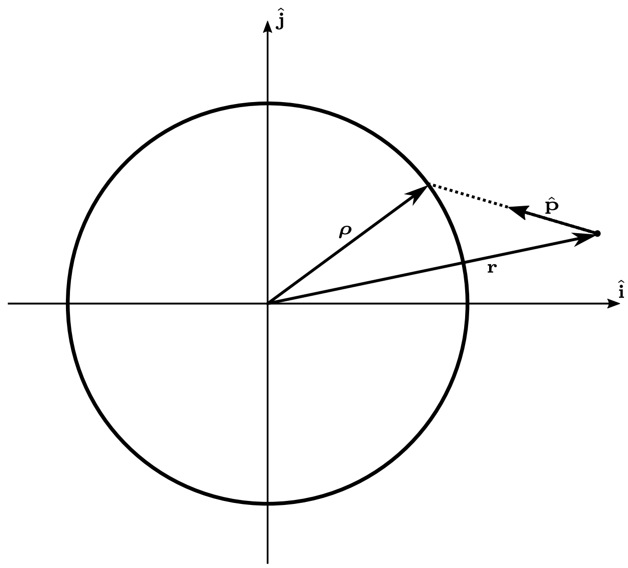


Figure 1: Unit Circle Intersection Geometry

equation. Consider

$$\boldsymbol{\rho} = \mathbf{r} + s\hat{\mathbf{p}}$$

where

$$\hat{\mathbf{p}} = \frac{\mathbf{p}}{||\mathbf{p}||}$$

Given the intersection point $\boldsymbol{\rho}$ exists on the unit circle,

$$||\mathbf{r} + s\hat{\mathbf{p}}|| = 1 \tag{13}$$

Equation (13) consists of a single unknown—the length of the vector from the reference point to the unit circle; $s$. Breaking each vector into components

$$(r_x + s\hat{p}_x)^2 + (r_y + s\hat{p}_y)^2 = 1$$

with

$$\begin{aligned} \mathbf{r} &= \begin{bmatrix} r_x \\ r_y \end{bmatrix} \\ \hat{\mathbf{p}} &= \begin{bmatrix} \hat{p}_x \\ \hat{p}_y \end{bmatrix} \end{aligned}$$

followed by expanding and collecting terms about powers of $s$,

$$\begin{aligned} \left(\hat{p}_x^2 + \hat{p}_y^2\right) s^2 + 2\left(r_x\hat{p}_x + r_y\hat{p}_y\right) s + r_x^2 + r_y^2 - 1 &= 0 \qquad (14) \\ as^2 + bs + c &= 0 \end{aligned}$$

allows for $s$ to be resolved through the use of the quadratic formula.

$$s = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \tag{15}$$

Substituting the coefficients of Eqn. (14) into (15) and simplifying, the roots of (14) become

$$s = \frac{-\beta \pm \sqrt{\beta^2 - \alpha\gamma}}{\alpha} \tag{16}$$

with coefficients:

$$\begin{aligned} \alpha &= \hat{p}_x^2 + \hat{p}_y^2 \\ \beta &= r_x\hat{p}_x + r_y\hat{p}_y \\ \gamma &= r_x^2 + r_y^2 - 1 \end{aligned}$$

Multiple tests must be performed to determine the intersection. First, before solving for $s$, compute the discriminant

$$d = \beta^2 - \alpha\gamma$$

rewriting Eqn. (16) as

$$s = \frac{-\beta \pm \sqrt{d}}{\alpha} \tag{17}$$

If $d < 0$, there is no real solution and the pointing vector misses the unit circle. Under these conditions, computing the tangent point of §6 is often of use.

Three conditions can occur if $d >= 0$. First, the reference point could be within the unit circle ($||\mathbf{r}|| < ||\boldsymbol{\rho}||$). Adding $\sqrt{d}$ to $-\beta$ in the numerator of Eqn. (17) will determine the proper value of $s$. If $||\mathbf{r}|| = ||\boldsymbol{\rho}||$, then the intersection is simply the reference point location ($s = 0$).

If $d > 0$ and $||\mathbf{r}|| > ||\boldsymbol{\rho}||$, then the reference point is outside the unit circle and the line defined by the pointing vector $\mathbf{p}$ intersects it. However, if the pointing vector is pointing away from the circle, then the solution will shoot backwards. This causes the pointing vector, in the direction it points, to miss the circle. If

$$||\hat{\mathbf{r}} + \hat{\mathbf{p}}|| < 1$$

with

$$\hat{\mathbf{r}} = \frac{\mathbf{r}}{||\mathbf{r}||} \tag{18}$$

then the intersection exists (review Figure 1) because the sum of the two unit vectors is within the unit circle—the pointing vector is directed inward. Under these conditions, subtract $\sqrt{d}$ from $-\beta$ in Eqn. (17). Otherwise, there is no intersection and the tangent point can instead be computed if appropriate.

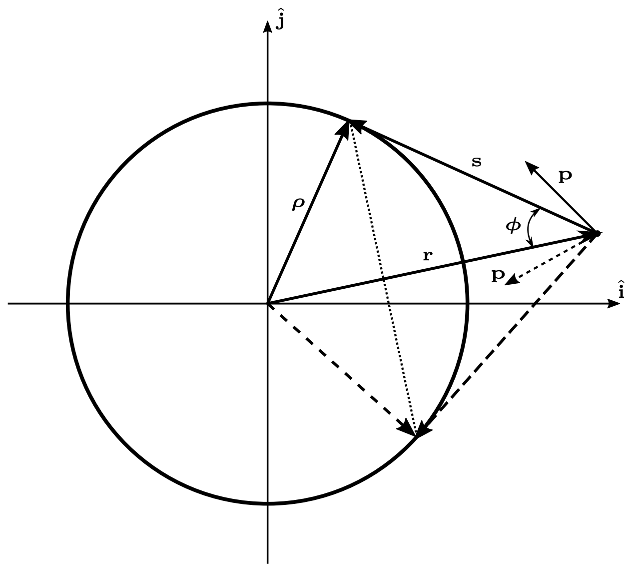


Figure 2: Unit Circle Tangent Geometry

# 6 Tangent to the Unit Circle

Deriving the tangent points $\boldsymbol{\rho}$ to a unit circle given an external reference point $\mathbf{r}$, as illustrated in Figure 2, is trivial compared to that of an ellipse. A pointing vector $\mathbf{p}$ is required only to determine which tangent point should be chosen, as there will always be two unless $||\mathbf{r}|| \leq ||\boldsymbol{\rho}||$. As with §5, the reference and pointing vectors are the same $\mathbf{r}$ and $\mathbf{p}$ from Eqns. (11) and (12).

The magnitude of the tangent point vector $\boldsymbol{\rho}$ is the radius of the unit circle. Therefore, taking advantage of the orthogonality between $\boldsymbol{\rho}$ and $\mathbf{s}$, the magnitude of $\mathbf{s}$ can be computed

$$\begin{aligned} \rho &= 1 \\ r^2 &= s^2 + \rho^2 \\ s &= \sqrt{r^2 - 1} \end{aligned}$$

denoting

$$\begin{aligned} r &= ||\mathbf{r}|| \\ \rho &= ||\boldsymbol{\rho}|| \end{aligned}$$

The tangent point can be constructed as a linear combination of the reference point $\mathbf{r}$ and the normal to this vector $\mathbf{r}_\perp$. Convenient unit basis vectors are $\hat{\mathbf{r}}$, Eqn. (18), and

$$\hat{\mathbf{r}}_\perp = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \hat{\mathbf{r}}$$

where the permutation matrix is simply a 90° vector rotation (direct transformation) matrix in the counterclockwise direction. The tangent point $\boldsymbol{\rho}$ illustrated by the solid line in Figure 2 is then

$$\boldsymbol{\rho} = \hat{\mathbf{r}}\,(r - s\cos\phi) + \hat{\mathbf{r}}_{\perp} s \sin\phi$$

Substituting,

$$\begin{aligned}\cos\phi &= \frac{s}{r}\\ \sin\phi &= \frac{1}{r}\end{aligned}$$

the two tangent points are

$$\boldsymbol{\rho} = \left(r - \frac{s^2}{r}\right)\hat{\mathbf{r}} \pm \frac{s}{r}\hat{\mathbf{r}}_{\perp} \tag{19}$$

Finally, if the pointing vector $\mathbf{p}$ and $\hat{\mathbf{r}}_{\perp}$ are in the same direction, the second term of Eqn. (19) is added. In other words, if $\mathbf{p}^{\mathrm{T}}\hat{\mathbf{r}}_{\perp} > 0$, add the second term; otherwise, subtract it.

If $\mathbf{p}$ and $\mathbf{r}$ are collinear ($\mathbf{p}^{\mathrm{T}}\hat{\mathbf{r}}_{\perp} = 0$), then either tangent point can be chosen. If the reference point is on the unit circle ($r = \rho$), then $\boldsymbol{\rho} = \mathbf{r}$. If the reference point is within the unit circle ($r < \rho$), there is no tangent point. Depending on the application, projecting the reference point to the circle such that $\boldsymbol{\rho} = \hat{\mathbf{r}}$ may be a useful result.

# 7 Return to Higher Dimensions

Once the two dimensional intersection or tangent point vector $\boldsymbol{\rho}$ of §5 or §6 is computed, it needs to be transformed back to $n$-dimensional Cartesian space. First, the transpose (equal to the inverse) of the orthonormal transformation from the unit $n$-sphere to unit circle system (Eqns. (6) or (10)) must be applied. Extending $\boldsymbol{\rho}$ to $n$-dimensions through padding with zeros,

$$\mathbf{z}^{c}_{\rho/\mathrm{O}} = [\boldsymbol{\rho}^{\mathrm{T}}\ 0\ \cdots\ 0]^{\mathrm{T}}$$

allows for the $n$-sphere solution

$$\begin{aligned}\mathrm{T}^{s}_{c} &= (\mathrm{T}^{c}_{s})^{\mathrm{T}}\\ \mathbf{z}^{s}_{\rho/\mathrm{O}} &= \mathrm{T}^{s}_{c}\mathbf{z}^{c}_{\rho/\mathrm{O}}\end{aligned}$$

Finally, the inverse (not transpose) of the Cartesian to unit $n$-sphere transformation (Eqns. (1), (2), (7), or (8)) is applied to convert the vector to the reference frame in which the hyperellipsoid is defined:

$$\begin{aligned}\mathrm{T}^{e}_{s} &= (\mathrm{T}^{s}_{e})^{-1}\\ \mathbf{z}^{e}_{\rho/\mathrm{O}} &= \mathrm{T}^{e}_{s}\mathbf{z}^{s}_{\rho/\mathrm{O}}\end{aligned}$$

## 8 Limitations

The reference and pointing vectors are used to build $(n-2)$ vectors that are orthogonal to to their span, allowing for the problem to be isolated within a two dimensional plane. This is not possible when these vectors are collinear. Under such conditions, the tangent point is completely undefined as there is no way to determine what direction should be taken. The intersection, however, becomes trivial. Once the geometry is transformed to the unit $n$-sphere system, create a unit vector from the position vector. The conversion of that unit vector back to Cartesian space is the intersection.

$$\begin{aligned}
\mathbf{z}^s_{\mathrm{R/O}} &= \mathrm{T}^s_e \mathbf{z}^e_{\mathrm{R/O}} \\
\mathbf{z}^e_{\rho/\mathrm{O}} &= \mathrm{T}^e_s \frac{\mathbf{z}^s_{\mathrm{R/O}}}{\left|\left|\mathbf{z}^s_{\mathrm{R/O}}\right|\right|}
\end{aligned}$$

The primary challenge surrounds handling the intersection problem where, from a numerical stability perspective, the position and pointing vectors are very *nearly* collinear. For this case, an $n$-dimensional version of Eqn. (14) should be used. After transformation of the hyperellipsoid to a unit $n$-sphere via $\mathrm{T}^s_e$, solve for the roots of Eqn. (16) as before where the coefficients are now accumulated over all $n$ dimensions:

$$\begin{aligned}
\alpha &= \sum_{i=1}^{n} \hat{p}_i^2 \\
\beta &= \sum_{i=1}^{n} r_i \hat{p}_i \\
\gamma &= \sum_{i=1}^{n} r_i^2 - 1
\end{aligned}$$

## 9 Challenges of Validation Beyond 3D

Validation of implemented algorithms is trivial in three dimensions. An ellipsoidal coordinate system can be chosen from which points on the surface of the ellipsoid are naturally be defined. After conversion to Cartesian coordinates, reference points can be chosen with the difference used to create pointing vectors. The intersection algorithm is then applied and the results validated against the original truth values.

The pointing vectors can then be used to generate tangent points. Validity is verified by first forming the vector from the reference point $\mathbf{z}_r$ to the suspected tangent point $\mathbf{z}_\rho$:

$$\mathbf{z}_s = \mathbf{z}_\rho - \mathbf{z}_r$$

First, $\mathbf{z}_s$ must be a linear combination of the reference $\mathbf{z}_r$ and pointing $\mathbf{z}_{\hat{p}}$ vectors (the unit pointing vector notation $\hat{p}$ is used for the subscript only to more easily distinguish it from the ellipsoid point denoted via $\rho$). Second, it must be normal to the surface normal at the tangent point, which is easily computed for an ellipsoid as the cross product of the basis vectors in the tangent plane to the ellipsoid—at the tangent point...

Validation is slightly more challenging in higher dimensions. First, it is always helpful to ensure the tangent and intersection points are actually on the surface of the hyperellipsoid. This is easily accomplished through the matrix definition of a quadratic surface [5, p. 85],

$$\mathbf{z}_\rho^{\mathrm{T}} \mathrm{M}^{-1} \mathbf{z}_\rho = 1 \tag{20}$$

where $\mathbf{z}_\rho$ is the $n$-dimensional vector representing the point on the hyperellipsoid and M is once again the symmetric positive definite matrix of the hyperellipsoid. Once it is verified that the point is on the surface of the hyperellipsoid, form the vector from the reference point to the intersection, $\mathbf{z}_s = \mathbf{z}_\rho - \mathbf{z}_r$, as with the three dimensional case. To confirm an intersection, this vector and the pointing vector must be collinear (once again, to within some tolerance).

$$\frac{\mathbf{z}_s^{\mathrm{T}} \mathbf{z}_{\hat{p}}}{||\mathbf{z}_s||\,||\mathbf{z}_{\hat{p}}||} = 1$$

However, it is necessary to validate which intersection point has been computed. In three dimensions, this can be accomplished through visual inspection of a surface plot. In higher dimensions, the quadratic surface definition Eqn. (20) is once again employed. First, increase the length of $\mathbf{z}_s$ by a differential amount ($\mathbf{z}_{s+}$) and compute a displaced intersection point

$$\mathbf{z}_{\rho+} = \mathbf{z}_r + \mathbf{z}_{s+}$$

This vector should be within the hyperellipsoid since $\mathbf{z}_{s+}$ penetrates the surface:

$$\mathbf{z}_{\rho+}^{\mathrm{T}} \mathrm{M}^{-1} \mathbf{z}_{\rho+} < 1$$

Next, decrease the length of $\mathbf{z}_s$ by a differential amount, and repeat the test, verifying

$$\mathbf{z}_{\rho-}^{\mathrm{T}} \mathrm{M}^{-1} \mathbf{z}_{\rho-} > 1$$

because $\mathbf{z}_{s-}$ falls short of the surface. If both tests pass, then $\mathbf{z}_\rho$ is indeed the first intersection, and not the point on the "backside" of the surface.

Verifying a tangent point once again begins by ensuring it is on the surface of the hyperellipsoid. Next, as with the three dimensional problem, confirm $\mathbf{z}_s$ is a linear combination of $\mathbf{z}_r$ and $\mathbf{z}_{\hat{p}}$. Finally, recall that a vector tangent to a convex surface will have a single intersection. Perform the same tests against Eqn. (20) as with the intersection validation, except now both values are expected to be external to the surface:

$$\begin{aligned} \mathbf{z}_{\rho+}^{\mathrm{T}} \mathrm{M}^{-1} \mathbf{z}_{\rho+} &> 1 \\ \mathbf{z}_{\rho-}^{\mathrm{T}} \mathrm{M}^{-1} \mathbf{z}_{\rho-} &> 1 \end{aligned}$$

Neither the lengthened nor shortened forms of $\mathbf{z}_s$ should fall within the surface as that would indicate more than one point of intersection.

## 10 Justification

Three affine transformations, and their inverses, are applied to solve the hyperellipsoid intersection and tangent point problems. Two of the affine transformations are orthonormal—nothing more than a change of Cartesian basis vectors. The transformation converting the hyperellipsoid to a unit $n$-sphere scales along the principal axes, motivating review of key affine transformation properties:

- An affine transformation is invertible. Without the ability to transform back to the original Cartesian system, this method would be of little use.
- The affine transformation is linear—lacking curvature, the metric tensor of the resulting system is constant across all space. A line maps to another line.
- Application of an affine transformation A to a hyperellipsoid is another hyperellipsoid. A transformation $\mathrm{AMA}^{\mathrm{T}}$ to a symmetric positive definite matrix M remains a symmetric positive definite matrix with $n$ eigenvalues as semiaxis lengths and $n$ corresponding distinct and orthogonal eigenvectors as principal axes.
- Collinearity is preserved through an affine transformation. Two lines that are collinear, or parallel, will remain so after the transformation is applied [2, p. 260].
- If $f : C \to A$ is a function representing the application of an affine transformation, then $f$ is onto, one-to-one, and the metric spaces $(C, d_C)$ and $(A, d_A)$ preserve distance. The Cartesian and affine spaces, with their associated metric tensors, are metrically equivalent [6, pp. 60–62]. The distance between points before and after transformation remains invariant.

Consider the quadratic surface constraint, Eqn. (20), subject to an affine transformation A:

$$\begin{aligned}
(\mathrm{A}\mathbf{z}_\rho)^{\mathrm{T}} \left(\mathrm{AMA}^{\mathrm{T}}\right)^{-1} \mathrm{A}\mathbf{z}_\rho &= \\
\mathbf{z}_\rho^{\mathrm{T}} \mathrm{A}^{\mathrm{T}} \left(\mathrm{A}^{\mathrm{T}}\right)^{-1} \mathrm{M}^{-1} \mathrm{A}^{-1} \mathrm{A}\mathbf{z}_\rho &= \\
\mathbf{z}_\rho^{\mathrm{T}} \mathrm{M}^{-1} \mathbf{z}_\rho &= 1
\end{aligned}$$

An intersection or tangent point resolved in one reference frame remains on the surface of the hyperellipsoid after an affine transformation.

The constant nature of the transformation across space results in a tangent point, realized by a line passing through the reference point and having a *single* common point with the hyperellipsoid surface, remaining a tangent point. Further consider two lines parallel to a line tangent to a surface, offset by a differential distance in opposite directions, not in the plane tangent to the surface. The differential displacement results in one line that does not intersect the surface, and the other intersecting at two points—one line misses the surface while the other passes through it. An affine transformation results in all three lines, offset by infinitesimal amounts, remaining parallel. The two offset lines continue to miss and double intersect the surface. The line between them, the transition from two to zero intersections, must then continue to have a single intersection with the surface.

Finally, the relationship between the intersection or tangent point $\mathbf{z}_\rho$ to the linear combination of the reference point location $\mathbf{z}_r$ and a scaled unit pointing vector $\hat{\mathbf{z}}_p$, with the application of an affine transformation is

$$\begin{aligned} \mathbf{z}_\rho &= \mathbf{z}_r + s\hat{\mathbf{z}}_p \\ \mathrm{A}\mathbf{z}_\rho &= \mathrm{A}\left(\mathbf{z}_r + s\hat{\mathbf{z}}_p\right) \end{aligned}$$

Therefore, the distance from the reference point also remains invariant (when measured with each reference frame's associated metric tensor):

$$\begin{aligned} \mathbf{z}_\rho - \mathbf{z}_r &= s\hat{\mathbf{z}}_p \\ \mathrm{A}\left(\mathbf{z}_\rho - \mathbf{z}_r\right) &= \mathrm{A}s\hat{\mathbf{z}}_p \end{aligned}$$

# A QR Decomposition

The $n$-dimensional cross product was utilized to generate an $[n{\times}n]$ orthonormal transformation essentially triangularizing an $[n{\times}2]$ matrix composed of the position and pointing vectors. Although this process is analogous to that of the 3-dimensional scenario, full QR decomposition through Householder transformations is more suitable for this purpose. Only two reflections are required to form an $[n{\times}n]$ orthonormal matrix isolating the position and pointing vectors in the first two dimensions of an $n$-dimensional Cartesian reference frame. This method is significantly more computationally efficient than direct implementation of successive cross product operations. Bierman [1, pp. 59–63] introduces the Householder transformation as a method of matrix triangularization. Trefethen [7, pp. 69–73] further explores applications and numerical properties of such transformations.

The full QR decomposition of an $[n{\times}m]$ matrix A,

$$\begin{aligned} \mathrm{Q}_1\mathrm{Q}_2\ldots\mathrm{Q}_n\mathrm{R} &= \mathrm{A} \\ \mathrm{QR} &= \mathrm{A} \end{aligned}$$

results in an $[n{\times}n]$ orthonormal matrix Q and $[m{\times}m]$ upper triangular matrix R, such that

$$\mathrm{R} = \mathrm{Q}^{\mathrm{T}}\mathrm{A}$$

Each $\mathrm{Q}_i$ is an $[n{\times}n]$ orthonormal matrix composed of an $[(i-1){\times}(i-1)]$ identity matrix in the upper diagonal block, and an $[(n-i-1){\times}(n-i-1)]$ Householder transformation in the lower diagonal block. Householder matrices are full rank ($|\mathrm{H}| \neq 0$), symmetric ($\mathrm{H} = \mathrm{H}^{\mathrm{T}}$), idempotent ($\mathrm{H}^2 = \mathrm{I}$), and therefore orthonormal transformations that reflect a point relative to a plane. Defining the Householder transformation by $\mathbf{v}$, a vector normal to the reflecting plane,

$$\mathrm{H} = \mathrm{I} - 2\frac{\mathbf{v}\mathbf{v}^{\mathrm{T}}}{\mathbf{v}^{\mathrm{T}}\mathbf{v}}$$

The position vector $\mathbf{z}^s_{\mathrm{R/O}}$ of §4 is used to define the plane reflecting itself into the first Cartesian basis vector $\hat{\mathbf{i}}$.

$$\begin{aligned} \mathbf{z} &= \mathbf{z}^s_{\mathrm{R/O}} \\ \mathbf{v}_1 &= \mathbf{z} + \mathtt{sign}\,(z_1)\,||\mathbf{z}||\,\hat{\mathbf{i}} \end{aligned}$$

Figure 3 illustrates the geometry. The plane normal to $\mathbf{v}_1$ is depicted by the centerline $\zeta$. For $\mathbf{z}$ located in the first quadrant, $z\hat{\mathbf{i}}$ is added to $\mathbf{z}$ to maximize numerical stability by choosing a reflector that moves $\mathbf{z}$ as far away as possible [7, pp. 72–73] to $\mathbf{z}_r$. Subtracting $z\hat{\mathbf{i}}$ to locate $\mathbf{v}_1$ would still generate a valid reflector. For this case, $\zeta$ would be $\mathbf{v}_1$ and vice versa. The first transformation, $\mathrm{Q}_1$, is equal to the Householder reflection:

$$\begin{aligned} \mathrm{Q}_1 &= \mathrm{H}_1 \\ &= \mathrm{I} - 2\frac{\mathbf{v}_1\mathbf{v}_1^{\mathrm{T}}}{\mathbf{v}_1^{\mathrm{T}}\mathbf{v}_1} \end{aligned}$$

Generating $\mathrm{Q}_2$ begins with the $2^{nd}$ through $n^{th}$ elements of the transformed pointing vector.

$$\begin{aligned} \mathbf{z} &= \left(\mathrm{Q}_1 \mathbf{z}^{s}_{\mathrm{P/R}}\right)[2:n] \\ \mathbf{v}_2 &= \mathbf{z} + \mathtt{sign}\left(z_1\right)||\mathbf{z}||\,\hat{\mathbf{i}} \end{aligned}$$

Note $\mathbf{z}$ is reflected to $\hat{\mathbf{i}}$ (now of dimension $n-1$) as was done with the position vector. Now that the position vector is isolated, the $n-1$ dimensional transformation being formed only affects the pointing vector. The second transformation is composed of the identity matrix (in this case, the scalar value 1), and an $[(n-1)\times(n-1)]$ reflection:

$$\mathrm{Q}_2 = \begin{bmatrix} 1 & \\ & \mathrm{H}_2 \end{bmatrix}$$

The orthonormal transformation isolating the intersection and tangent point geometry can now be formed.

$$\begin{aligned} \mathrm{T}^{c}_{s} &= \mathrm{Q}^{\mathrm{T}} \\ &= \left(\mathrm{Q}_1 \mathrm{Q}_2\right)^{\mathrm{T}} \end{aligned}$$

Algorithm 3 is included to illustrate full QR decomposition.

**Algorithm 3** Using Householder transformations, decompose an $[m\times n]$ matrix into the product of an orthonormal $[m\times m]$ Q and upper triangular $[m\times n]$ matrix R such that A = QR. Loop over each column of A. The $i^{th}$ iteration modifies rows $i$ through $m$ and columns $i$ through $n$, allowing matrix A to be modified in place.

**procedure** QR($\mathrm{A}_{m\times n}$)
  $\mathrm{E} = \mathrm{I}_{m\times m}$
  $\mathrm{Q} = \mathrm{I}_{m\times m}$
  $\mathrm{R} = \mathrm{A}$
  **for** $i = 1 : n$ **do** ▷ Note dynamic sizing of $\mathbf{x}$, $\mathbf{v}$, $\hat{\mathbf{u}}$, H
    $\mathbf{x} = \mathrm{A}\,[i:m]\,[i]$
    $\mathbf{v} = \mathbf{x} + \mathtt{sign}\left(\mathbf{x}\,[1]\right)||\mathbf{x}||\,\mathrm{E}\,[i:m]\,[i]$
    $\hat{\mathbf{u}} = \mathbf{v}/\,||\mathbf{v}||$
    $\mathrm{H} = \mathrm{E}\,[i:m]\,[i:m] - 2\hat{\mathbf{u}}\hat{\mathbf{u}}^{\mathrm{T}}$
    $\mathrm{R}\,[i:m]\,[i:n] = \mathrm{HR}\,[i:m]\,[i:n]$
    $\mathrm{Q} = \mathrm{Q}\begin{bmatrix} \mathrm{I}_{(i-1)\times(i-1)} & \\ & \mathrm{H} \end{bmatrix}$
  **end for**
  **return** Q, R
**end procedure**

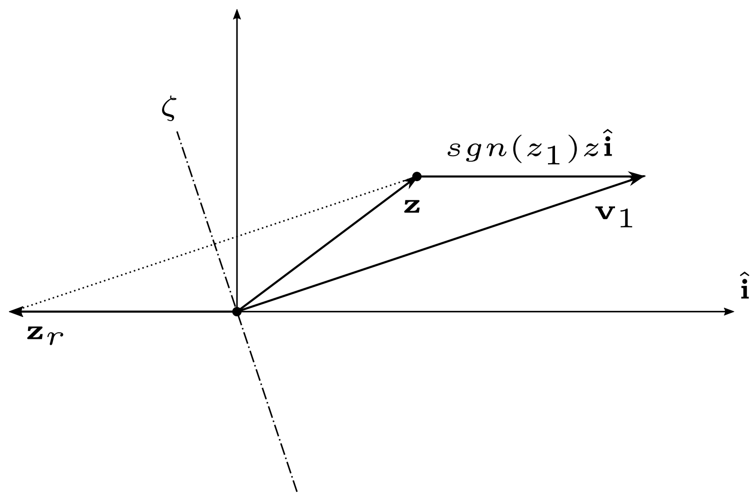


Figure 3: Householder Reflection

# B 3D Tangent Point Example

Figure 4 illustrates the geometry of a three dimensional tangent point example. The plane formed by the position vector (blue) and the pointing vector (black) defines an ellipse intersecting the ellipsoid. Following an affine transformation converting the ellipsoid to a unit sphere (Fig. 5), the intersecting ellipse becomes a unit circle. The final transformation (Fig. 6) isolates the position, pointing vector, and unit circle, within a two dimensional reference frame.

# C Hyperellipsoid Visualization in 3D

Depicting ellipsoids of greater than three dimensions within only two proved to be beyond reach of the author. However, $n-2$ unique three dimensional slices of an $n$-dimensional geometry can be plotted in 2D through row permutations of $\mathrm{T}_s^c$ (§4). Figure 7 illustrates both the tangent and intersection (red) to a nine dimensional hyperellipsoid. All 7 ellipsoids share the same two dimensional ellipse (highlighted in purple). The intersection and tangent points, along with the position and pointing vectors, all retain the same coordinates and components within the $x$-$y$ (Fig. 8) plane that has been isolated as the dimensions of $\mathrm{T}_s^c$ are permuted.

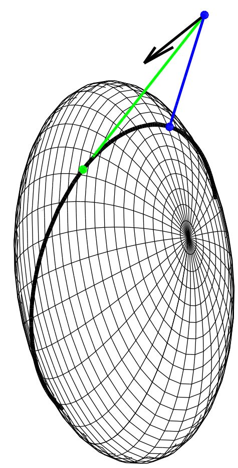

Figure 4: Ellipse Tangent Point Geometry in Cartesian Space

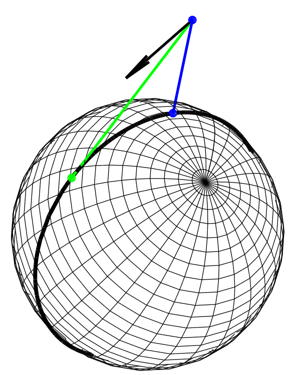

Figure 5: Unit Sphere Tangent Geometry Following an Affine Transformation

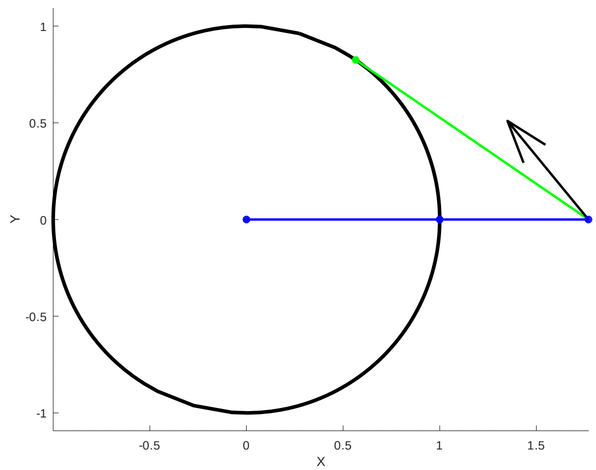


Figure 6: Unit Circle Tangent Geometry

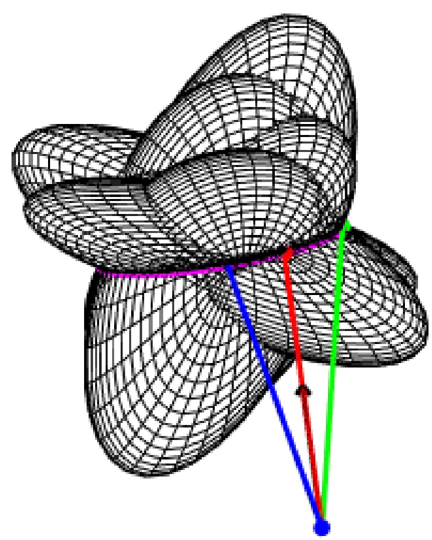

Figure 7: 9D Tangent and Intersection, 3D Slices

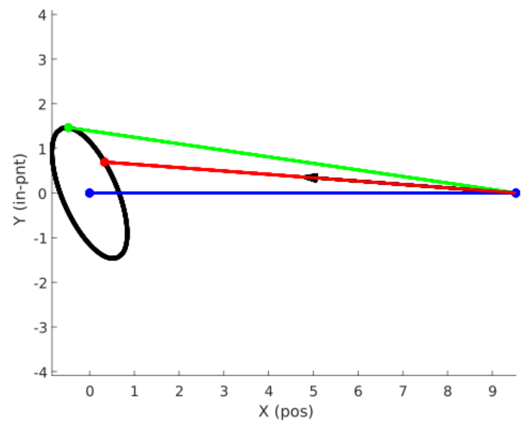


Figure 8: 9D Tangent and Intersection, 2D Slice

# References

[1] G. J. Bierman, *Factorization Methods for Discrete Sequential Estimation*, Dover Publications, Inc., Mineola, NY, 1977.

[2] O. Byer, F. Lazebnik, and D. L. Smeltzer, *Methods for Euclidean Geometry*, Mathematical Association of America, Inc., 2010.

[3] P. Grinfeld, *Introduction to Tensor Analysis and the Calculus of Moving Surfaces*, Springer, New York, NY, 2013.

[4] I. Gura and R. Gersten, *On analysis of n-dimensional normal probabilities*, Tech. Rep. TR-0066(5129-01)-2, Aerospace Corporation, 1970.

[5] C. Lanczos, *Applied Analysis*, Dover Publications, Inc., New York, NY, 1988.

[6] B. Mendelson, *Introduction to Topology: Third Edition*, Dover Books on Mathematics, Dover Publications, New York, NY, 2012.

[7] L. N. Trefethen and D. Bau III, *Numerical Linear Algebra*, SIAM, Philadelphia, PA, 1997.