# A variation on the Chamberlin Trimetric map projection

Author: author@email.com, 55 XXXXXX St, XXXXXXXX, XX, 01234

**ARTICLE HISTORY**
Compiled December 15, 2020

**ABSTRACT**
A variation of the Chamberlin Trimetric map projection is presented, termed the Matrix Trimetric projection. The Matrix Trimetric projection amounts to a linear transformation of the squares of the distances from a given point to three control points. The analytical form of the forward projection is simpler than the Chamberlin projection, and the inverse projection requires numerical iteration of only one parameter. Comparisons between the two projections are made using a representative list of control points. The Chamberlin projection outperforms the Matrix projection on measures of angle deformation and scale deformation, and the reverse is true for a measure of distance deformation. However, the difference between the projections is small, and the Matrix Trimetric projection is a viable alternative for the use cases of the Chamberlin Trimetric projection.

*Keywords: Chamberlin trimetric, Matrix trimetric, projection, cartography, distortion, distance,*

## 1. Introduction

Any map projection must introduce some form of distortion. In particular, no map projection can be both conformal (angle-preserving) and equal-area. Projections that eliminate one form of distortion often introduce a great deal of another form of distortion. For example, the Mercator projection is a conformal projection that drastically enlarges areas near the poles. (Snyder, 1987) A compromise map projection is one that seeks to balance different kinds of distortion; allowing small amounts of distortion across multiple measures can produce more aesthetically pleasing maps.

The Chamberlin Trimetric projection is such a compromise map projection. It is named for Wellman Chamberlin, a chief cartographer for the National Geographic Society, which has published wall and atlas maps using his projection. The Chamberlin projection is appropriate for mapping whole continents and large portions of continents. (Christensen, 1992) The Chamberlin projection has a simple geometric construction, shown in Figure 1. Three control points on the globe are specified, and a triangle in the plane is constructed having the same distances between its vertices as the true distances between the points on the globe. True distances from those control points to a given point on the globe are measured, and arcs are drawn at those distances from their respective points in the plane. The arcs form a small triangle, and a point in that triangle is chosen as the projection of the original point. In the time period when manual plotters were used the exact definition of this point was less important, but Christensen (1992) and most modern implementations (e.g. PROJ contributors, 2019) use the centroid of the small triangle. Thus, the Chamberlin projection is a form of triangulation. (Snyder & Voxland, 1989)

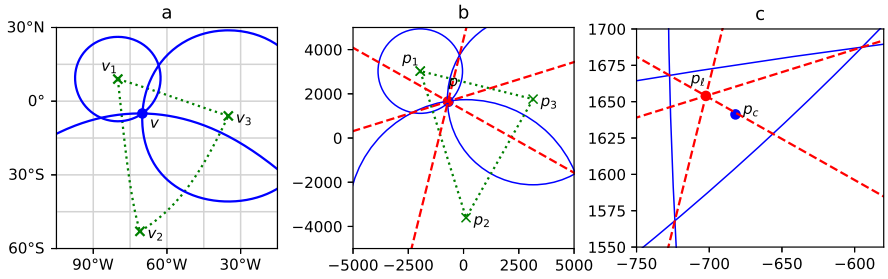


**Figure 1.** Construction of the Chamberlin and Matrix Trimetric projections. Plot 1a is in equirectangular projection. Plot 1b is the projection to the plane. Plot 1c is the same as plot 1b, zoomed in to the small triangle created by the three circles. $\mathbf{p}_c$ is the point projected by the Chamberlin projection and $\mathbf{p}_m$ is the point projected by the Matrix Trimetric projection. Green solid lines indicate the control triangles, the circles are blue dots, and the perpendiculars are red dashes.

This geometric construction is simple, but an analytic formula is preferred for modern computerized GIS, and the algorithmic form for the Chamberlin Trimetric projection is somewhat involved. This is presented in Christensen (1992), and can be seen in the source code to its implementation in PROJ contributors (2019); the algorithmic form has special cases to handle when the given point lies at a control point, or on a geodesic between control points. The geodesic case also requires calculating azimuths between the control points and the projected point.

This text presents a new map projection called the Matrix Trimetric projection as an alternative to the Chamberlin Trimetric projection. A geometric construction related to the Chamberlin Trimetric projection results in a map projection that is similar but has a simpler analytical formula, as well as other advantages.

## 2. Definitions and assumptions

We use the spherical approximation to the Earth. For this text, we assume a sphere with a radius of 6,371 km. The Chamberlin and Matrix projections are compromise projections, and do not perfectly preserve angle, area, or distance. For most practical ellipsoids, distortions due to the spherical approximation are negligible compared to distortions due to the projections themselves.

$\mathbf{v}$ and subscripted versions shall denote points on the sphere, and $\mathbf{p}$ and subscripted versions shall denote points in the Euclidean plane. Representing the points on the sphere as a unit vector allows one to apply the tools of linear algebra. Refer to a text on linear algebra, such as Strang (1980), if needed. Let latitude be $\varphi$ and longitude be $\lambda$, then latitude and longitude can be converted to a unit vector as so: (Lapaine, 2017)

$$\mathbf{v} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} \cos(\varphi)\cos(\lambda) \\ \cos(\varphi)\sin(\lambda) \\ \sin(\varphi) \end{bmatrix}. \tag{1}$$

The conversion from a unit vector to latitude and longitude is:

$$\begin{aligned}\varphi &= \arctan\left(z, \sqrt{y^2 + x^2}\right), \\ \lambda &= \arctan\left(y, x\right),\end{aligned} \tag{2}$$

where the 2-variable form of arctan is used, commonly called `arctan2` or `atan2` in numeric software libraries.

The distance function between two unit vectors is

$$d\left(\mathbf{v}_i, \mathbf{v}_j\right) = R \arccos\left(\mathbf{v}_i \cdot \mathbf{v}_j\right) = R \arctan\left(\|\mathbf{v}_i \times \mathbf{v}_j\|, \mathbf{v}_i \cdot \mathbf{v}_j\right), \tag{3}$$

where R is the radius of the Earth. The latter form, using the 2-variable form of arctan, is the most numerically stable. Euclidean distances are found using the usual Euclidean norm: $\|\mathbf{p}_a - \mathbf{p}_b\|$.

Let $\mathbf{v}_1$, $\mathbf{v}_2$, $\mathbf{v}_3$ be control points on the sphere, and let $\mathbf{V}$ be the matrix having $\mathbf{v}_i$ as its $i$th column. Let $\mathbf{p}_1$ etc. be the control points on the plane. The triangles with vertices at $\mathbf{v}_i$ or $\mathbf{p}_i$ are called the control triangles: spherical or planar control triangle, respectively, if the distinction is important. For this paper, the control triangle is not allowed to have very small or zero area. We also exclude the case where all three points of the spherical control triangle lie or almost lie on the same great circle. None of these cases are typical use cases for the Chamberlin projection. Note that the planar control triangle is **not** the image of the spherical control triangle under either projection! The image of the spherical control triangle is slightly larger and has curved edges.

$\mathbf{p}_i$ must be arranged such that $d\left(\mathbf{v}_i, \mathbf{v}_j\right) = \|\mathbf{p}_i - \mathbf{p}_j\|$ for all $i$ and $j$ in $\{1, 2, 3\}$: the spherical length of the edges of the spherical control triangle are equal to the Euclidean lengths of the planar control triangles. Without loss of generality, also assume that $\|\mathbf{p}_i\|$ is the same for all $i$, such that the circumcenter of the control triangle lies at the origin. This just removes a translation in the plane in order to simplify the formulas; false northing and easting can be added later. Given $\mathbf{v}_i$, $\mathbf{p}_i$ can be constructed as follows. Let $i, j, k$ be a cyclic permutation of $\{1, 2, 3\}$, and let $s_k = d\left(\mathbf{v}_i, \mathbf{v}_j\right)$. The circumradius of the Euclidean triangle with sides of length $s_k$ is (Isaacs, 2009)

$$C = \frac{\prod_i s_i}{\sqrt{\sum_i s_i^2 s_{i+1}^2 - s_i^4}}. \tag{4}$$

From the (Euclidean) law of cosines, the interior angle $\phi_i$ at each vertex is

$$\phi_i = \arccos\left(\frac{s_k^2 + s_j^2 - s_i^2}{2 s_k s_j}\right). \tag{5}$$

Let $\mathbf{P}$ be the matrix whose $i$th column is the vector $\mathbf{p}_i$. Then a set of points satisfying the requirements are given as so:

$$\mathbf{P} = \begin{bmatrix} C\cos\left(2\phi_2\right) & C\cos\left(2\phi_1\right) & C \\ C\sin\left(2\phi_2\right) & -C\sin\left(2\phi_1\right) & 0 \end{bmatrix}. \tag{6}$$

These points can be rotated about the origin as desired. Note that some intermediate values in Equation 4 may be large. This should not cause a problem on modern

computers using common geographic units of distance, but some special applications may need to scale these values to avoid overflow.

To measure distortion of area and angle, we use the areal scale factor $s$ and maximum angular deformation $\omega$ as defined in equations 12–15, 27, and 28 in section 4 of Snyder (1987). Derivatives are estimated numerically using second-order central differences on a 1° grid via the `gradient` function in Numpy. (Harris et al., 2020) There is no standard measurement of distance distortion for map projections, but for the projections in this text it makes sense to measure distances from the control points. Let $r_i$ be the distance from $\mathbf{v}_i$ to $\mathbf{v}$ on the sphere, as earlier, and let $\ell_i$ be the distance from $\mathbf{p}_i$ to $\mathbf{p}$ in the plane. The measure, the total distance deviation or $D$, is the sum of differences in the distances measured on projected and unprojected distances:

$$D = \sum_i |r_i - \ell_i| \,. \tag{7}$$

## 3. Forward projection

$r_i$ is also the radius of a circle on the sphere surface such that $\mathbf{v}_i$ is its center and $\mathbf{v}$ lies on its boundary. The Chamberlin projection draws a circle of radius $r_i$ around each point $\mathbf{p}_i$. Each pair of circles intersects in at most two points. If one draws a line through the two points of intersection of each pair of circles, that line is perpendicular to the triangle edge. These are the dashed red lines in Figure 1. Once the lines are drawn for each pair of circles, the three lines appear to meet at the same point, a fact that can be proven with a simple triangle theorem sometimes attributed to Carnot. (Posamentier & Salkind, 1996; Wohlgemuth, 2010) That point is not necessarily within the small triangle, although it is for most points within the control triangle. (If $\mathbf{v}$ lies on a control triangle edge, one pair of circles has a single point of intersection. One can draw a perpendicular through that point of intersection and continue as in the general case.)

The equations of each perpendicular line, taken together, create a linear system. It is an overdetermined system of 3 equations in 2 variables, but since all 3 lines meet at the same point, it has a solution. Ultimately this system can be solved for $\mathbf{p}_m$ to define a forward map projection as follows.

$$\mathbf{p}_m = \mathbf{M} \begin{bmatrix} r_1^2 & r_2^2 & r_3^2 \end{bmatrix}^\top , \tag{8}$$

$$\mathbf{M} = \frac{1}{2T} \begin{bmatrix} y_3 - y_2 & y_1 - y_3 & y_2 - y_1 \\ x_2 - x_3 & x_3 - x_1 & x_1 - x_2 \end{bmatrix} = \frac{1}{2T} \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \mathbf{P} \begin{bmatrix} 0 & -1 & 1 \\ 1 & 0 & -1 \\ -1 & 1 & 0 \end{bmatrix} , \tag{9}$$

$$T = \begin{vmatrix} x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \\ 1 & 1 & 1 \end{vmatrix} . \tag{10}$$

$T$ is equal to twice the area of the Euclidean control triangle.

The matrix $\mathbf{M}$ has a (right) nullspace spanned by the vector $[1, 1, 1]^\top$. This implies the Matrix Trimetric projection is not one-to-one for all possible values of $r_i$; for

example, for any values of $r_i$ such that $r_1 = r_2 = r_3$, then $\mathbf{p}_m = [0, 0]^\top$. It turns out that the Chamberlin and Matrix Trimetric projections project the entire sphere to a bounded portion of the plane. The boundary of that portion can be termed the boundary of the projection. There is a region of overlap that is mapped into the same area but in reverse orientation. That region includes the antipodes of the control points. In real applications, the overlap region can be excluded.

## 4. Inverse projection

Given $\mathbf{p}_m$, start to invert the projection as so:

$$\begin{bmatrix} k_1 & k_2 & k_3 \end{bmatrix}^\top = \mathbf{M}^+ \mathbf{p}_m \,, \tag{11}$$

$$\mathbf{M}^+ = \frac{2}{3} \begin{bmatrix} 2x_1 - x_2 - x_3 & 2y_1 - y_2 - y_3 \\ -x_1 + 2x_2 - x_3 & -y_1 + 2y_2 - y_3 \\ -x_1 - x_2 + 2x_3 & -y_1 - y_2 + 2y_3 \end{bmatrix} = \frac{2}{3} \begin{bmatrix} -2 & 1 & 1 \\ 1 & -2 & 1 \\ 1 & 1 & -2 \end{bmatrix} \mathbf{P}^\top . \tag{12}$$

$k_i = r_i^2 - h$ for some value of the free parameter $h$. This is a general solution to inverting Equation 8: $\mathbf{M}^+$ is the pseudoinverse of $\mathbf{M}$ and vice versa. Because $\mathbf{M}^+$ has a left nullspace spanned by the vector $[1, 1, 1]$, it follows that $\sum_i k_i = 0$, and also that $h = \frac{1}{3} \sum_i r_i^2$. These equations are not enough to determine $r_i$ and $\mathbf{v}$; information about the sphere needs to be introduced.

To simplify the derivation, use a unit sphere with $R = 1$, so that $r_i$ has units of radians of arc on the surface of the sphere. The circle of points $\mathbf{v}$ at distance $r_0$ from a point $\mathbf{v}_0$ is simply the circle created by a plane intersecting the sphere. This plane may be specified as $\mathbf{v}_0 \cdot \mathbf{v} = \cos(r_0)$. Replacing $\mathbf{v}_0$ with $\mathbf{v}_i$ and $r_0$ with $r_i$ for each $i$ gives a linear system. Thus,

$$\mathbf{v} = \mathbf{V}^{-1} \begin{bmatrix} \cos(r_1) & \cos(r_2) & \cos(r_3) \end{bmatrix}^\top . \tag{13}$$

For the point to lie on the unit sphere, $\|\mathbf{v}\| = 1$. Let $\mathbf{c}$ be a vector with $i$th component $\cos(r_i)$. Then, $\mathbf{c}^\top \left(\mathbf{V}^\top \mathbf{V}\right)^{-1} \mathbf{c} = 1$. Make the substitution

$$r_i = \sqrt{k_i + h} \,. \tag{14}$$

We now have an equation with one unknown, $h$.

Some obvious bounds can be placed on $h$. In units of radians, $0 \le r_i \le \pi$. Since this must hold for every $r_i$, it follows that

$$-\min_i k_i \le h \le \pi^2 - \max_i k_i \,. \tag{15}$$

Within these bounds, there may be at most two solutions for $h$. The solution with smaller $h$ is the desired one, and the one with larger $h$ corresponds to the overlap region.

Let $\mathbf{A} = \left(\mathbf{V}^\top \mathbf{V}\right)^{-1}$ and

$$f(h) = \mathbf{c}^\top \mathbf{A} \mathbf{c} - 1 \,. \tag{16}$$

The derivative of $f(h)$ is

$$f'(h) = -\mathbf{c}^\top \mathbf{A}\mathbf{b}\,, \tag{17}$$

where $\mathbf{b}$ is a vector with $i$th component $\operatorname{sinc}\left(\sqrt{k_i + h}\right)$. Note that $f'(h)$ and $f(h)$ share many of the same terms, which can be exploited for more efficient calculation.

The lower solution for $\mathbf{p}_m = [0, 0]^\top$ is a global minimum for $h$, and can be calculated as so:

$$h_0 = \arccos^2\left(\frac{1}{\sqrt{\sum \mathbf{A}}}\right), \tag{18}$$

where $\sum \mathbf{A}$ denotes the sum of all entries in the matrix $\mathbf{A}$.

Given all the preceding, Newton's method can be applied to solve for $h$. A suitable initial condition is

$$h_{\min} = \max\left(h_0, -\min_i k_i\right), \tag{19}$$

which appears to consistently result in convergence to the lower root of $f(h)$. One could use $\frac{1}{3}\sum_i l_i^2$, where $l_i = |\mathbf{p}_m - \mathbf{p}_i|$ approximates $r_i$ fairly well, but for some $\mathbf{p}_m$ outside the control triangle the iteration converges to the wrong root.

A good approximation is achieved for points inside the control triangle within a few iterations. Convergence is somewhat slower farther away from the control triangle. At the boundary of the projection, the lower solution and upper solution are the same and $f(h) = f'(h) = 0$ so Newton's method converges at a merely linear rate. (Burden & Faires, 2006)

## 5. Comparison

A software implementation of the Matrix Trimetric projection was created in Python, using the libraries Numpy (Harris et al., 2020), Scipy (Virtanen et al., 2020), Pandas (Wes McKinney, 2010), GeoPandas (Jordahl et al., 2020), and their dependencies. The implementation of the Chamberlin Trimetric projection comes from PROJ (PROJ contributors, 2019). Because the Chamberlin Trimetric and Matrix Trimetric projections are programmed in different languages, one compiled and one interpreted, a comparison of computation time would be unfair and is not included here.

Christensen (1992) gave a list of control triangles for the Chamberlin projection in that paper's Table 1. The control triangles are repeated in this text's Table 1. These serve as a set of test cases for comparing the Chamberlin and Matrix projections. The length of each side and the area of each triangle (with the spherical approximation given earlier) is also given, and the control triangles are sorted by area.

For each control triangle, summary statistics for $\omega$, $D$, and $s$ are shown in Figures 2–4. Note that the control triangles are sorted by area in these figures as well. These statistics are measured within the control triangles, and should not be taken to summarize the entire map; for most of the triangles, the region of interest extends outside the control triangle. Rather, the statistics allow a quantitative comparison of the two projections.

As shown in Figure 2, the Matrix projection consistently has a larger $\omega$ than the

| Region | Point | | | Side length | | | Area |
|---|---|---|---|---|---|---|---|
| | 1 | 2 | 3 | 1 | 2 | 3 | |
| Africa Wall | 19°3'W, 24°25'N | 20°E, 35°S | 59°3'E, 24°25'N | 7,783 | 7,785 | 7,783 | 32.38 |
| North America Wall | 150°W, 55°N | 92°30'W, 10°N | 35°W, 55°N | 7,064 | 6,434 | 7,064 | 23.71 |
| South America Wall | 80°W, 9°N | 71°W, 53°S | 35°W, 6°S | 6,161 | 5,259 | 6,947 | 17.70 |
| Europe Wall | 15°E, 72°N | 8°W, 33°N | 38°E, 33°N | 4,254 | 4,541 | 4,541 | 9.09 |
| E South America | 63°33'W, 8°8'N | 58°33'W, 34°35'S | 35°13'W, 5°47'S | 4,000 | 3,502 | 4,779 | 7.25 |
| S South America | 43°W, 18°S | 72°W, 18°S | 72°W, 56°S | 4,225 | 4,874 | 3,064 | 6.77 |
| Australia | 134°E, 8°S | 110°E, 32°S | 158°E, 32°S | 4,487 | 3,643 | 3,643 | 6.76 |
| NW South America | 69°W, 25°S | 55°W, 10°N | 85°W, 10°N | 3,284 | 4,261 | 4,177 | 6.70 |
| Canada Wall | 150°W, 60°N | 97°30'W, 50°N | 45°W, 60°N | 3,423 | 5,197 | 3,423 | 6.11 |
| Canada Atlas | 98°13'W, 61°39'N | 135°W, 40°N | 55°W, 40°N | 6,560 | 3,761 | 3,449 | 5.28 |

**Table 1.** Table of control triangles and their measurements, from Christensen (1992). Side $n$ is opposite point $n$. Lengths are in km, and areas are in millions of square km. Measurements use spherical approximation.

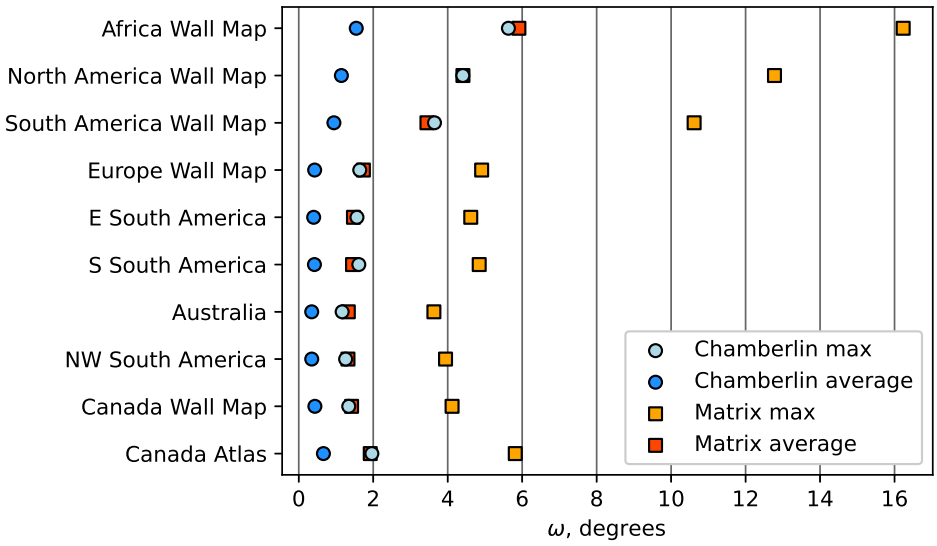


**Figure 2.** Comparison of maximum angular deformation $\omega$.

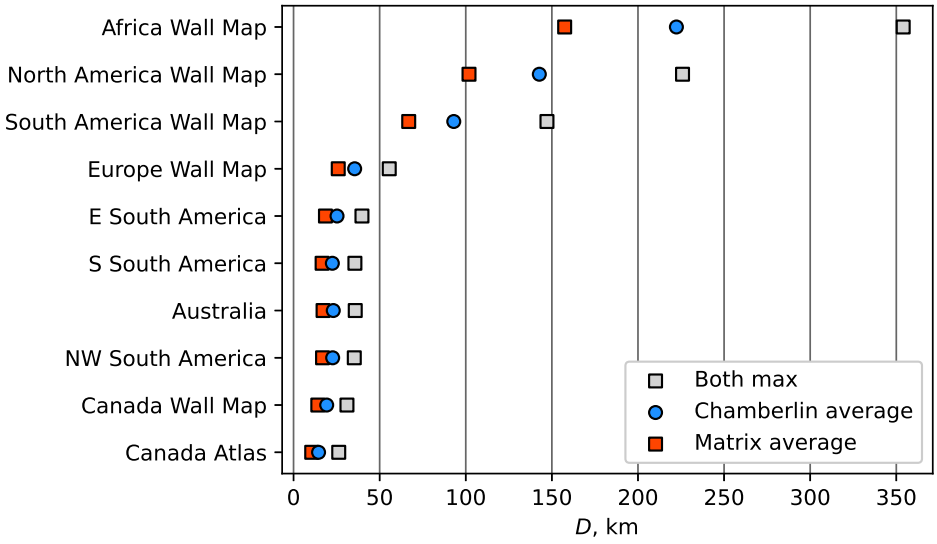


**Figure 3.** Comparison of total distance deviation $D$.

Chamberlin projection; in fact, the maximum $\omega$ for Chamberlin is near the average $\omega$ for Matrix. Maximum and average $\omega$ trend upwards with control triangle area, although asymmetry of the control triangle has some influence too, as in the Canada triangles. The maximum $\omega$ for Matrix is about 3 times that for Chamberlin, and the average $\omega$ is about 3 to 4 times. However, for moderately-sized triangles, the maximum $\omega$ values are small for both projections, not exceeding 6°. Even for the large triangles – Africa, North America, and South America Wall Maps – the maximum distortion is tolerable, and the average does not exceed 6°.

Total distance deviation $D$ in Figure 3 shows an even clearer trend, monotonically increasing with triangle area. Both projections have the same maximum $D$ values. The Matrix projection has consistently lower average $D$ values than the Chamberlin; among these triangles, the maximum $D$ for Chamberlin is about 1.35 to 1.4 times the maximum $D$ for Matrix. The worst distortion, Africa Wall Map's maximum $D$, is less than 5% of its edge lengths, and smaller control triangles have a maximum $D$ that is an even smaller percent.

As true 1:1 scale maps are rare, absolute values of $s$ are unimportant; the quantity $\sigma = \frac{\max s}{\min s} - 1$ is plotted in Figure 4 instead. The trend of $\sigma$ is less clear. A combination of control triangle area and asymmetry influences this value. In general, this value is higher for Matrix than for Chamberlin, but not by any consistent factor, and for the large and symmetric Africa Wall Map triangle the Matrix projection is actually lower than Chamberlin. The amount of scale distortion is small for both; except for the very obtuse Canada Atlas triangle, all values are below 10%.

A comparison of the two projections using the South America Wall Map control points is in Figure 5, using land mass shape files from Patterson and Kelso (2020). The South America Wall Map control triangle is fairly representative, being somewhat asymmetric but not too obtuse. The small part of Central America in the upper left is somewhat shifted between the two maps, as is the western area near Ecuador and Peru, but no features on either map are conspicuously distorted compared to the other.

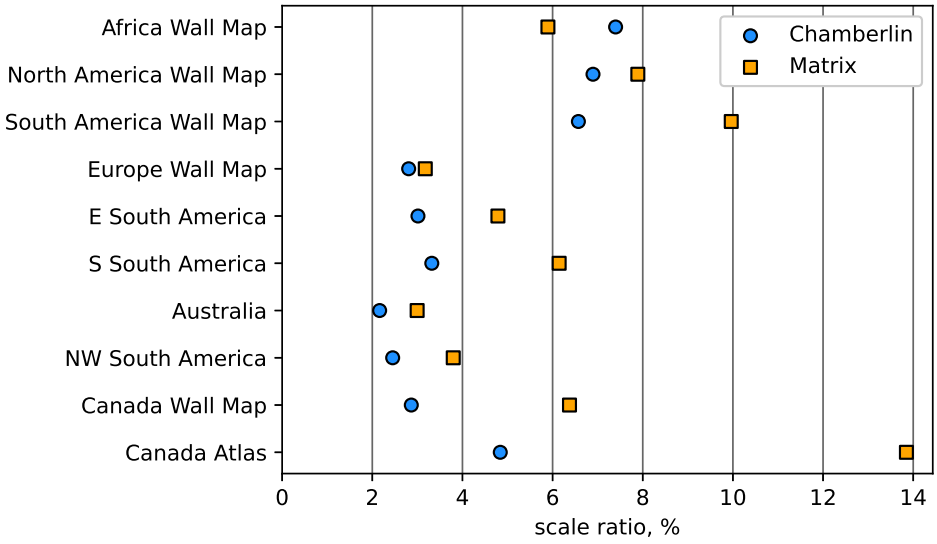


**Figure 4.** Comparison of areal scale ratio $\sigma$.

Figure 6 shows ellipses of distortion with centers on a grid at steps of 15° latitude and longitude. Distortion is somewhat more visible in this figure; one can see that the Matrix projection introduces slightly more non-conformal distortion near the control points.

Contour lines of $\omega$ are shown in Figure 7. Those in the Chamberlin projection are ovals, while those for the Matrix Trimetric projection extend outward through each edge of the control triangle. Both reach a minimum $\omega$ of 0 at a point inside the control triangle. Highly obtuse control triangles, like Canada Atlas, may have two local minima for $\omega$ within the control triangle. $\omega$ is higher for Matrix than Chamberlin, except for a small region in the control triangle where both are near 0 and some regions near the boundary of the map.

Figure 8 depicts contour lines of $D$ for these two projections. Both projections have a local minimum of 0 at the control points, which follows from the geometric construction. Both have a local maximum inside the control triangle. $D$ for the Matrix projection is low along the control triangle edges, while for the Chamberlin projection it is larger. $D$ is lower for Matrix than Chamberlin in a region including the control triangle and nearly all of South America, except for a small region near the shared maximum. It is also lower in some regions extending outward from the control points.

Contour lines of $s$ are shown in Figure 9. The contour lines have the same elliptical structure in both projections, centered on a point where $s$ reaches a local minimum. This point is different for each projection, which influences the differences in aggregate area distortion in Figure 4. That minimum value of $s$ is different for each projection.

In most practical applications the area pictured in these figures will be sufficient, but we can state trends as both projections extend outward from the control triangle, outside the area shown in these figures. $\omega$ increases to 180° at the projection boundary, representing the inversion of the overlap region. $D$ increases in general, but for the Matrix Trimetric projection $D$ remains low on the great circles containing the control triangle edges. $s$ increases to a maximum and then decreases to 0 at the projection

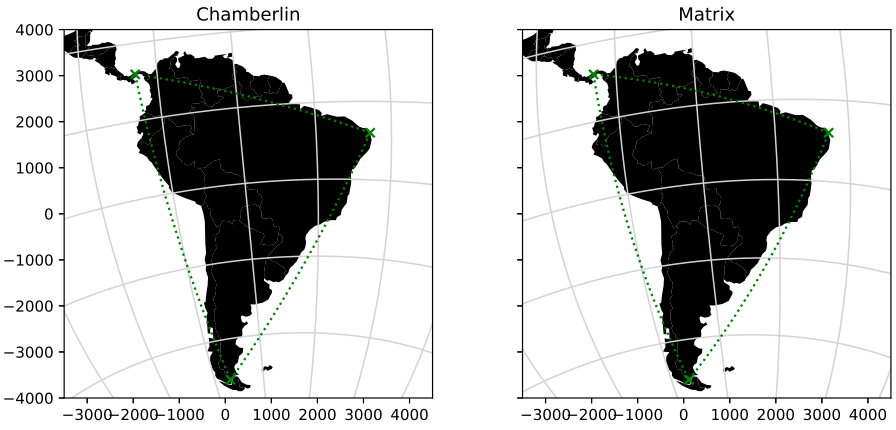


**Figure 5.** Projection of South America and surroundings. Green lines indicate the control triangle and control points in this and the following figures.

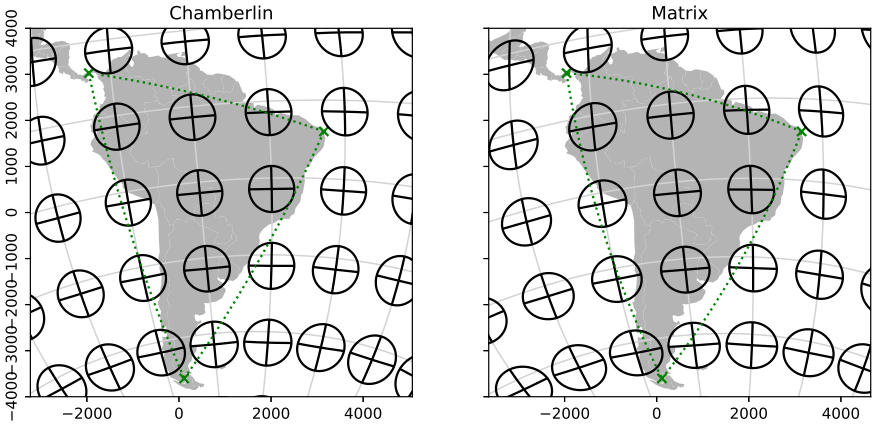


**Figure 6.** Ellipses of distortion, on a 15° grid.

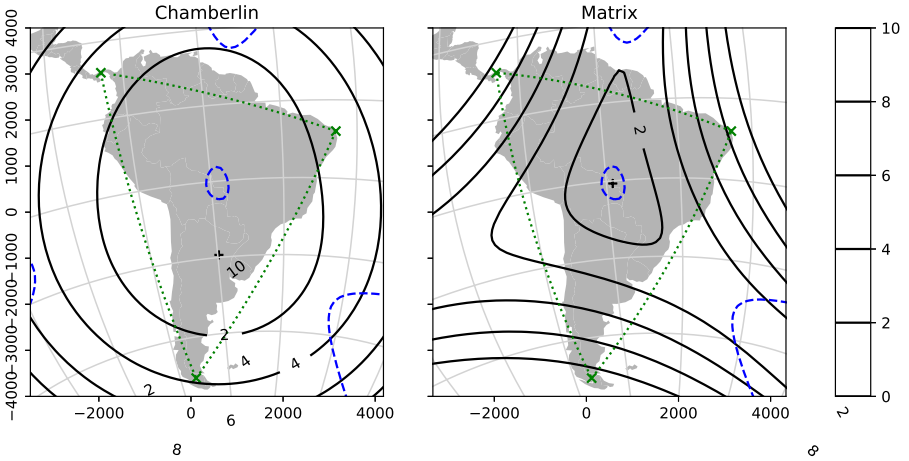


**Figure 7.** Maximum angular deformation $\omega$, in degrees. Blue dashed lines indicate where the two projections have equal $\omega$ values.

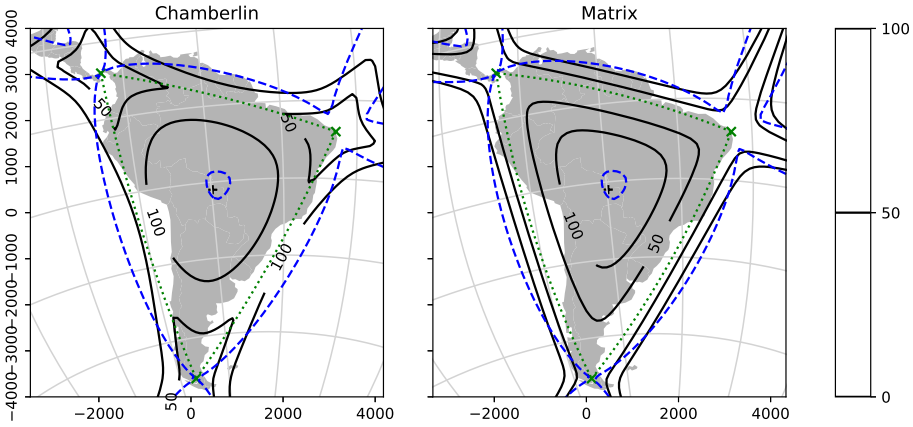


**Figure 8.** Total distance deviation $D$. Blue dashed lines indicate where the two projections have equal $D$ values.

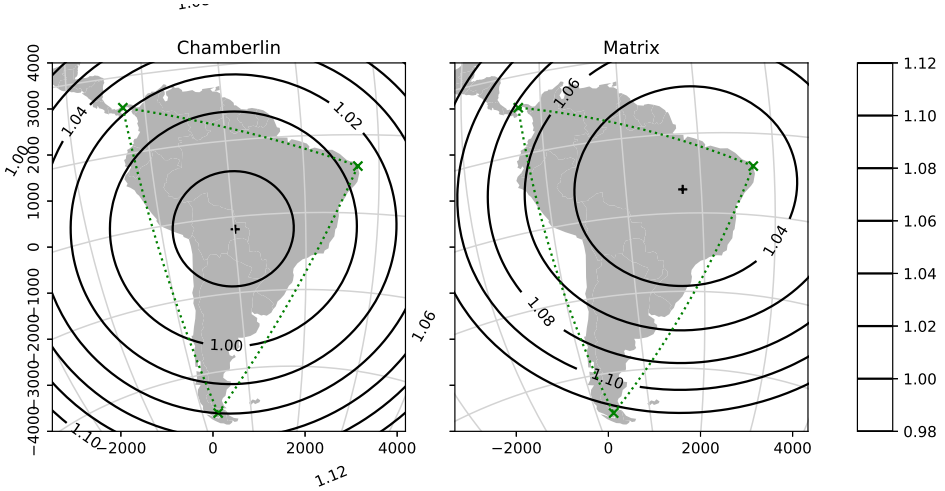


**Figure 9.** Areal scale factor $s$.

boundary.

## 6. Conclusion

We have demonstrated a new compromise map projection, the Matrix Trimetric projection. The forward formula is given in Equations 8–10. Equation 8 is a simple product of a matrix and a vector. The vector depends on the point being transformed. The matrix calculated by the latter two equations depends only on the control points, so can be calculated once and reused. The forward formula does not have the special cases of the Chamberlin projection, making it more efficient to calculate and easier to program. The Chamberlin projection also requires calculation of azimuths, which the Matrix projection does not. Both projections share the same distance calculations, which limits the degree to which they can differ. That said, when programmed in the same language, the Matrix Trimetric projection is likely to be somewhat faster than the Chamberlin Trimetric projection.

The inverse formula is calculated by Equations 11–14 and a single-variable Newton's method iteration using Equations 16 and 17 with the lower bound of Equation 19 as an initial condition. The matrices in Equations 12 and 13, as well as matrix **A** in Equations 16 and 17, also depend only on the control points and can be reused. The Chamberlin projection, in contrast, has no known inverse aside from brute force two-variable inversion.

Figures 2–4 plot aggregate measures of distortion within each of the listed control triangles. The Chamberlin projection outperforms the Matrix projection in terms of angular distortion and (except in one case) scale distortion, but the reverse is true for distance distortion. For both projections, distortion tends to be larger for a larger control triangle. Figures 5–8 show the projections applied to South America, demonstrating the shape of distortions across the mapped area. Differences between the two projections are difficult to see with the eye.

The same control points were used for both projections in this work, but the optimal placement of control points may be somewhat different for the application of different projections to the same geographic feature. The angular distortion of the Matrix

Trimetric projection is lower in a region extending through the middle of the control triangle edges, so one may wish to orient the triangle to take advantage of that.

Overall, the Matrix Trimetric projection is a viable alternative for the use cases of the Chamberlin Trimetric projection. The Matrix Trimetric has the advantage if an inverse is needed, if one wishes to reduce distance distortion, or if processing time is a concern. In general, the differences between the results of the Chamberlin and Matrix projections are small enough that either one is adequate for their common use cases.

The Matrix Trimetric projection is closely related to the Chamberlin projection. Both are related to the azimuthal equidistant projection and the two-point equidistant projection, which depend on measurements from a single point and from a pair of points, respectively. (Snyder, 1987) This suggests an "$n$-metric" family of projections based on distances from some number of points. The form of the forward formula suggests a possible family of projections that are simple functions of such distances, such as a polynomial function. Additionally, there may exist similar projections using measurements other than distance, such as area.

A Python implementation of the Matrix Trimetric projection, as well as code used to produce the calculations and figures in this text, is available on the author's Github site. (Author, 2020)

## References


Author. (2020). *xxxxxx.* https://github.com/xxxxxxxx/xxxxxxx.

Burden, R., & Faires, J. (2006). Solutions of equations in one variable. In *Numerical analysis* (8.1 ed., pp. 67–85). Cengage Learning.

Christensen, A. H. (1992). The chamberlin trimetric projection. *Cartography and Geographic Information Systems*, *19*(2), 88-100.

Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., ... Oliphant, T. E. (2020, September). Array programming with NumPy. *Nature*, *585*(7825), 357–362.

Isaacs, I. (2009). Triangles. In *Geometry for college students* (pp. 50–93). American Mathematical Society.

Jordahl, K., den Bossche, J. V., Fleischmann, M., Wasserman, J., McBride, J., Gerard, J., ... Leblanc, F. (2020). *geopandas/geopandas: v0.8.1.* https://zenodo.org/record/3946761. Zenodo.

Lapaine, M. (2017). Modelling the world. In A. Kent & P. Vujakovic (Eds.), *The Routledge handbook of mapping and cartography* (pp. 187–201). Taylor & Francis.

Patterson, T., & Kelso, N. V. (2020). *Natural earth.* https://www.naturalearthdata.com/. (Accessed: 2020-01-17)

Posamentier, A., & Salkind, C. (1996). The Pythagorean theorem. In *Challenging problems in geometry* (pp. 85–86). Dover Publications.

PROJ contributors. (2019). PROJ coordinate transformation software library [Computer software manual]. Retrieved from `https://proj.org/`

Snyder, J. P. (1987). *Map projections–a working manual* (No. 1395). US Government Printing Office.

Snyder, J. P., & Voxland, P. M. (1989). *An album of map projections* (No. 1453). US Government Printing Office.

Strang, G. (1980). *Linear algebra and its applications.* Academic Press.

Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau,

D., ... SciPy 1.0 Contributors (2020). SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, *17*, 261–272.

Wes McKinney. (2010). Data Structures for Statistical Computing in Python. In Stéfan van der Walt & Jarrod Millman (Eds.), *Proceedings of the 9th Python in Science Conference* (p. 56 - 61).

Wohlgemuth, M. (2010). Ein Satz von Carnot. In *Mathematisch für fortgeschrittene Anfänger: Weitere beliebte Beiträge von Matroids Matheplanet* (pp. 273–276). Spektrum Akademischer Verlag.