Conformal Map Transformations for Meteorological Modelers

Albion D. Taylor
National Oceanic and Atmospheric Administration
E mail: adtaylor@arlrisc.ssmc.noaa.gov

February, 1997

Abstract

A system of utility software has been written which meteorological computer modelers can incorporate in their programs to provide the mathematical and physical properties of Conformal maps that their models may need. In addition to coordinate transformations, routines supply projection-dependent terms of the governing equations, wind component conversions, and rotation axis orientation components. The routines seamlessly handle the transitions from Polar Stereographic through Lambert Conformal to Mercator Projections. Initialization routines allow concurrent handling of multiple projections, and allow simple means of defining computational model grids to the software.

Key Words: Conformal Projection, Meteorological Computer Models, Meteorological Equations of Motion, CMAPF, FORTRAN, C.

1 Introduction

Meteorological Computer Models are typically constructed as finite difference approximations to differential equations. These finite difference equations are generally based on a square mesh grid imposed on a flat map of the Earth in some selected projection, usually conformal.

To relate model results to the weather on the Earth, the modeler requires subroutines to translate the x-y coordinates of the grid to the latitude-longitude coordinates of the Earth. Several software packages provide this capability, among them certain of the W3LIB routines, Stackpole, (1988) maintained by the National Weather Service and certain routines (EZMAP) of the *ncargraphics* package (Middleton-Link, 1993, pp 63-169).

However, the modeler needs more than coordinate transformation. The model winds are represented by vector components aligned with the grid; conversion to meteorological standards requires rotation by an amount dependent on position and projection. The governing equations include terms which are determined by the mathematical properties of the map projection, and which would not be present if the Earth were flat.

These capabilities are directly related to the same map projection that defines the task of coordinate transformation, and ideally would be provided by the same software package. This paper desscribes an ensemble of subroutines that provide these services for the most common conformal projections (polar centered Lambert Conformal, Mercator, and Stereographic projections). Source code for these Conformal MAP Function (CMAPF) routines is available at URL http://www.arl.noaa.gov/ss/models/cmapf.html, or by anonymous FTP from the server IAMG.ORG.

Our routines are more user-friendly than the earlier cited software. Initialization routines are provided to allow the user to describe the map projection and associated grid in easily understood terms. The user's information is compiled in one or more parameter blocks which can then be passed to the other routines. In this way, the user can work with more than one grid at a time, and exchange data between grids.

By contrast, the W3LIB routines require each separate parameter to be passed individually in every call, which can make code harder to maintain and requires the user to define the grid in the terms dictated by the routine. Further, the ability to orient the grid at an angle to North on the map is restricted in the Lambert Conformal situation and prevented in the Mercator situation. The *ncargraphics* EZMAP routines provide only the forward (Lat-Lon to x-y) transformation, while restricting rotation in the Lambert Conformal situation, as well as the gridsize in the x-y plane. Initialization sets internal parameters, precluding working with more than one grid at a time.

Traditionally, Meteorological Models have been presented on maps created from an assumed spherical Earth, rather than the more exact flattened ellipsoid, and our routines are not an exception in this regard. Accounting for the flattening according to the Ellipsoid of the International Union of Geodesy and Geophysics (Richards & Adler, 1972, pp 1-23) would adjust the scale of our projections by $\pm 0.5\%$ and the aspect ratio (ratio of scale in North-South direction to that in East-West direction) by $\pm 0.7\%$, depending on latitude.

Section 2 presents the standard modeling equations and indicates the terms which will be provided on demand by the subroutines. The next two sections present instructions on how to initialize (Section 3) and use (Section 4) our software. Section 5 covers the differences between the FORTRAN and C versions. Appendix A presents the mathematical basis for the subroutines, while Appendix B outlines special considerations at and around the North and South poles.

Mention of a commercial company or product is done only for illustrative purposes and does not constitute an endorsement by the author, the National Oceanographic and Atmospheric Administration, or any agency of NOAA.

2 Conformal Projections and the Equations of Motion

The system of routines described in this paper support a conformal coordinate system, i.e. an orthogonal coordinate system with distance scales isotropic in the horizontal. Combined with a vertical dimension, they allow for a special case of orthogonal curvilinear coordinates. The form of the Meteorological equations of motion derived below relies only on the intrinsic distance relations of such coordinates, and the gradients of these distances. This feature of the equations means that the information provided by the CGSZ.. functions and the CCRV.. routines is sufficient to construct meteorological models on this grid, as well as suggests how to extend such systems to more general map transformations.

In a general system of orthogonal curvilinear coordinates, we write $x_1 = x$, $x_2 = y$, $x_3 = z$, and suppose the distances given by small changes in coordinates are given by $ds^2 = h_1^2 dx_1^2 + h_2^2 dx_2^2 + h_3^2 dx_3^2$ (c.f., (Haltiner & Williams, 1980, p441)). We designate velocity components either in grid-units per time unit (denoted by u_i) or in physical units of meters per second (denoted as convenient by U_i or by u, v, and w). Then $U_1 = u = h_1\, u_1$, $U_2 = v = h_2\, u_2$ and $U_3 = w = h_3\, u_3$. In such coordinates, the vector operators for gradient, divergence and curl are

$$\nabla p = \left(\frac{\partial p}{\partial x}\frac{\mathbf{i}}{h_1} + \frac{\partial p}{\partial y}\frac{\mathbf{j}}{h_2} + \frac{\partial p}{\partial z}\frac{\mathbf{k}}{h_3} \right)$$

where $\mathbf{i}$, $\mathbf{j}$, and $\mathbf{k}$ are unit vectors in the direction of increasing x, y, and z, respectively,

$$\nabla \cdot \mathbf{u} = \frac{1}{H} \sum_i \frac{\partial}{\partial x_i} (H u_i)$$

where $H = h_1\, h_2\, h_3$, and

$$\begin{aligned} \nabla \times \mathbf{u} \;=\; & \frac{\mathbf{i}}{h_2 h_3}\left(\frac{\partial h_3 w}{\partial y} - \frac{\partial h_2 v}{\partial z} \right) + \frac{\mathbf{j}}{h_3 h_1}\left(\frac{\partial h_1 u}{\partial z} - \frac{\partial h_3 w}{\partial x} \right) \\ & + \frac{\mathbf{k}}{h_1 h_2}\left(\frac{\partial h_2 v}{\partial x} - \frac{\partial h_1 u}{\partial y} \right) \end{aligned}$$

If an orthogonal coordinate system is conformal, we have $h_3 = 1$ while $h_1 = h_2 = \sigma\,(x,y,z) = \sigma_0(x,y)\; r\; /\; a$, where σ_0 represents the gridsize at surface level (the length in km of a unit distance in the x-y coordinate system), a represents the radius of the Earth, and the factor $r/a = 1 + z/a$ is due to the fact that the length of an arc increases proportional to distance from the Earth's center. Then gradient, divergence and curl become

$$\nabla p = \frac{1}{\sigma}\frac{\partial p}{\partial x}\mathbf{i} + \frac{1}{\sigma}\frac{\partial p}{\partial y}\mathbf{j} + \frac{\partial p}{\partial z}\mathbf{k} \quad , \tag{1}$$

$$\begin{aligned}\nabla\cdot\mathbf{u} &= \frac{1}{\sigma^2}\frac{\partial\sigma u}{\partial x}+\frac{1}{\sigma^2}\frac{\partial\sigma v}{\partial y}+\frac{1}{r^2}\frac{\partial}{\partial z}\left(r^2 w\right) \qquad (2)\\ &= \frac{1}{\sigma}\frac{\partial u}{\partial x}+\frac{1}{\sigma}\frac{\partial v}{\partial y}+\frac{\partial w}{\partial z}+\left(uG_x+vG_y+\frac{2w}{a}\right)\frac{a}{r} \quad ,\end{aligned}$$

and

$$\begin{aligned}\nabla\times\mathbf{u} &= \mathbf{i}\left(\frac{1}{\sigma}\frac{\partial w}{\partial y}-\frac{\partial v}{\partial z}-\frac{v}{r}\right)+\mathbf{j}\left(\frac{\partial u}{\partial z}+\frac{u}{r}-\frac{1}{\sigma}\frac{\partial w}{\partial x}\right)\\ &\quad +\mathbf{k}\left(\frac{1}{\sigma}\left(\frac{\partial v}{\partial x}-\frac{\partial u}{\partial y}\right)+\frac{r}{a}\left(vG_x-uG_y\right)\right) \quad , \qquad (3)\end{aligned}$$

where $G_x = \sigma_0^{-2}\partial\sigma_0/\partial x$ and $G_y = \sigma_0^{-2}\partial\sigma_0/\partial y$ define the components of the intrinsic curvature vector of the map projection at the point (x,y). This vector, projected normal to any apparently straight line in the x, y plane, yields the magnitude of the actual curvature of that line on the Earth.

The continuity equation is written, using the above expressions, as

$$\begin{aligned}\frac{\partial\rho}{\partial t}+\nabla\cdot(\rho\mathbf{u}) &= \qquad (4)\\ \frac{\partial\rho}{\partial t}+\frac{1}{\sigma^2}\left(\frac{\partial\rho\sigma u}{\partial x}+\frac{\partial\rho\sigma v}{\partial y}\right)+\frac{\partial\rho w}{\partial z}+\frac{2\rho w}{r} &= 0\end{aligned}$$

where ρ denotes the atmospheric density. This is essentially the same as (Haltiner and Williams, 1980, p 13), who write m instead of $1/\sigma$, and drop the z-dependence of σ and hence the last term as being negligibly small.

The momentum equation is written as

$$\frac{D\mathbf{u}}{Dt}+\frac{1}{\rho}\nabla p+2\Omega\mathbf{e}_N\times\mathbf{u}+g\mathbf{k}=0 \qquad (5)$$

where p is the pressure and $\mathbf{e}_N = N_x\,\mathbf{i} + N_y\,\mathbf{j} + N_z\,\mathbf{k}$ is a unit vector aligned with the Earth's axis in the North direction. Ω is the rotation rate of the Earth, and $f = 2\Omega N_z$ represents the Coriolis parameter, while $2\Omega N_x$ and $2\Omega N_y$ represent coefficients of coriolis force exchanges between the horizontal and vertical. Du/Dt is the acceleration, expressed in terms of a "substantive derivative" operator D/Dt which yields the rate of change of a quantity following a wind trajectory, and if the quantity is a vector, allowing for the effects of parallel transport on its components.

The acceleration term Du/Dt has components Du_i/Dt given by

$$\frac{Du_i}{Dt}=\frac{\partial u_i}{\partial t}+\sum_{j=1}^{3}u_j\frac{\partial u_i}{\partial x_j}+\sum_{j,k=1}^{3}\left\{\begin{matrix} i\\ jk\end{matrix}\right\}u_j u_k$$

where the first two terms of the right hand side represent the total rate of change in the components of the velocity, and the final term represents the rate of change of the direction of the basis vectors with motion along an air parcel trajectory. The symbol in braces ({}), also written Γ^i_{jk}, is a Christoffel symbol of the second kind, defined in (McConnell, 1957), and for orthogonal curvilinear coordinates, evaluated on page 156 of that reference. Substituting that evaluation, while converting to physical units, we find

$$\frac{DU_i}{Dt} = \frac{\partial U_i}{\partial t} + \sum_{j=1}^{3} \frac{U_j}{h_j}\frac{\partial u_i}{\partial x_j} + \sum_{j \neq i} U_j \left(\frac{U_j}{h_j}\frac{\partial \ln (h_i)}{\partial x_j} - \frac{U_j}{h_i}\frac{\partial \ln (h_j)}{\partial x_{ij}} \right)$$

For the conformal case, substituting for the h_i yields

$$\begin{aligned} \frac{Du}{Dt} &= \frac{du}{dt} + \frac{a}{r}\left(v\left(uG_y - vG_x\right) + \frac{wu}{a} \right) \\ \frac{Dv}{Dt} &= \frac{dv}{dt} + \frac{a}{r}\left(u\left(vG_x - uG_y\right) + \frac{wv}{a} \right) \\ \frac{Dw}{Dt} &= \frac{dw}{dt} - \frac{a}{r}\left(\frac{u^2+v^2}{a} \right) \end{aligned} \tag{6}$$

where

$$\frac{d}{dt} = \frac{\partial}{\partial t} + \frac{u}{\sigma}\frac{\partial}{\partial x} + \frac{v}{\sigma}\frac{\partial}{\partial y} + w\frac{\partial}{\partial z} \quad . \tag{7}$$

Using the above expressions for the wind acceleration, the momentum equation becomes

$$\begin{aligned} \frac{du}{dt} - v\frac{a}{r}\left(f - \left(vG_x - uG_y\right)\right) + \frac{1}{\rho\sigma}\frac{\partial p}{\partial x} &= -w\left(\frac{u}{r} + 2\Omega N_y\right) \\ \frac{dv}{dt} + u\frac{a}{r}\left(f - \left(vG_x - uG_y\right)\right) + \frac{1}{\rho\sigma}\frac{\partial p}{\partial y} &= -w\left(\frac{v}{r} - 2\Omega N_x\right) \\ \frac{dw}{dt} + \frac{1}{\rho}\frac{\partial p}{\partial z} &= \frac{a}{r}\left(\frac{u^2+v^2}{a}\right) + 2\Omega\left(N_y - vN_x\right) \quad . \end{aligned} \tag{8}$$

where $f = 2\Omega N_z$ denotes the Coriolis parameter, as before.

(Haltiner and Williams, 1980, pp 11-13) present momentum equations for the cases of Polar Stereographic projection (origin at North Pole, gridsize= 1 at the equator) and for the Mercator projection (gridsize= 1 at the equator, with the origin at the intersection of the equator with the prime meridian). Comparing Equation (8) with theirs, we find that they are equivalent, noting

$$N_x = -\frac{x}{a}\frac{1+\sin(\phi)}{2\cos(\phi)}, \;\; N_y = -\frac{y}{a}\frac{1+\sin(\phi)}{2\cos(\phi)} \quad ,$$

$$G_x = -\frac{x}{2a^2}, \;\; G_y = -\frac{y}{2a^2}$$

in the Polar Stereographic situation, and

$$N_x = 0, \;\; N_y = 1, \;\; G_x = 0, \;\; G_y = -\frac{\tan(\phi)}{a}$$

in the Mercator situation. In the above equations, ϕ represents the latitude. throughout. We also assume, with (Haltiner and Williams, 1980), that z is negligible in comparison with a, and so replace r with a throughout.

These equations are readily generalized for the Lambert Conformal projection which our subroutines support. They can also be generalized to other conformal projections such as Transverse Mercator and Oblique Stereographic.

The differences between these equations and those that would apply if the Earth were flat and could be assumed uniform, are multiples of the terms $G_x = \sigma_0^{-2}\partial\sigma_0/\partial x$ and $G_y = \sigma_0^{-2}\partial\sigma_0/\partial y$. These terms, whose units are radians per km, define the intrinsic curvature of the projection, i.e. the actual curvature on the Earth of a line which appears straight on the grid. A projection can always be specified so this curvature vanishes at any given point. The effects will be minuscule near that point but gradually become more significant away from that point and can be overwhelming in the vicinity of the pole (except for the case of the Polar Stereographic projection).

Thus, for a 10 m sec^{-1} East wind on a Mercator projection, the vorticity due to the term uG_y at latitude 45° ($2.23 \cdot 10^{-6}\,\mathrm{sec}^{-1}$) is 2% of the Coriolis parameter ($1.03 \cdot 10^{-4} sec^{-1}$). At 70° the ratio is 10%, ($1.27\ 10^{-5} sec^{-1}$ vs $1.37\ 10^{-4} sec^{-1}$), and at 85° is 140% ($2.07\ 10^{-4} sec^{-1}$ vs $1.45\ 10^{-4} sec^{-1}$).

To summarize, the form of the equations of motion shows that the following terms depend on the map projection, and may conveniently be provided by calls to any system of routines responsible for handling the projection: the gridsize $\sigma_0\,(x, y)$ as a function of x and y, its gradient $(G_x, G_y) = \sigma_0^{-2}\,(\partial\sigma_0/\partial x, \partial\sigma_0/\partial y)$, and the direction cosines of the Polar axis $(N_x,\ N_y,\ N_z)$. These terms are returned by some of the subroutines we have provided.

3 Initializing the CMAPF functions

The Conformal MAP Function (CMAPF) functions and subroutines perform coordinate transformations between points and vectors on the Earth's surface (given in degrees of latitude and longitude, with North and East as positive), and their equivalents on a class of $X - Y$ grids overlaying Conformal maps.

Each $X-Y$ grid and its corresponding map is defined to the program through a PARMAP array of 9 floating point variables. It is the user's responsibility to provide this array in the calling routines:

`REAL PARMAP(9).`

More than one such grid may be defined, allowing coordinate transformations between two or more grids.

Before coordinate transformations can be performed, the PARMAP arrays must be initialized. This is carried out in a two stage process: first the map projection is specified by a call to STLMBR; subsequently the grid is laid out on the projection and defined by a call either to STCM1P or to STCM2P.

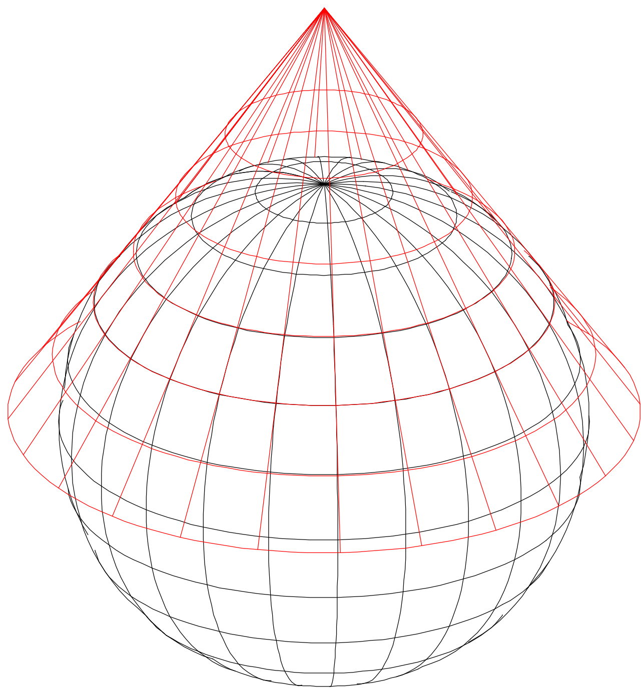

Figure 1: Conic projection

3.1 Initialization Stage 1: Specifying the Projection

In the class of maps supported by the software - Lambert Conformal, including Polar Stereographic and Mercator - the latitudes and longitudes of the Earth are projected onto a cone circumscribed on the earth at a specific latitude, as in Figure 1. A mathematical function maps a point of a given latitude ϕ and longitude λ to a point at a distance $R(\phi)$ from the apex of the cone, along the line from the apex tangent to the sphere at longitude λ. The special instances of the Polar Stereographic and the Mercator projection occur when the cone flattens to a plane tangent at the pole or extends to a cylinder tangent at the equator, respectively. In the special instance of the Polar Stereographic projection, there is a focal point (the opposite pole) from which rays may be conveniently drawn through points on the globe to corresponding points on the map.

After projection, each meridian becomes a line of the cone, drawn through the apex and tangent to the longitude, while circles of latitude map into cir-

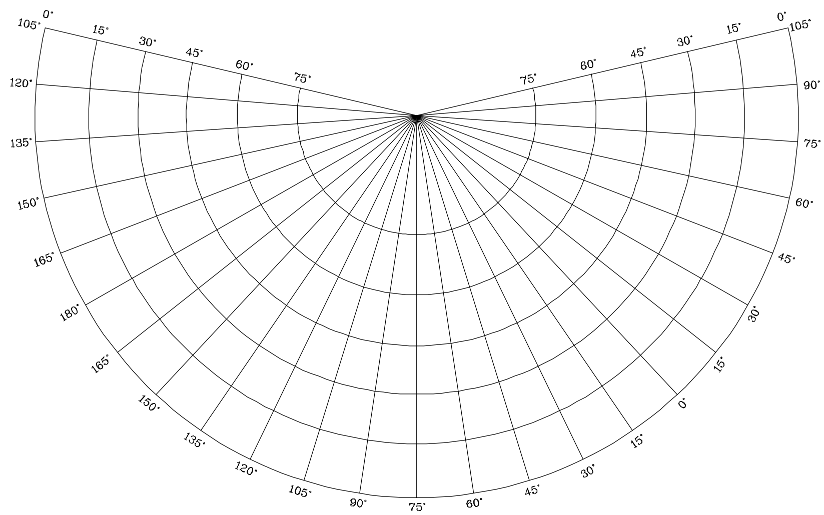

Figure 2: Conic projection as unrolled

cles on the cone whose spacing is determined by the conformal condition that the scale in the North-South direction be the same as that in the East-West direction.

Unrolled and flattened, the projection appears as in Figure 2. Longitudes appear as rays from a common center, and circles of latitude appear as arcs. The arcs do not fill complete circles, but a fraction γ of a circle where γ is the sine of the latitude at which the circumscribed cone is tangent to the Earth. Necessarily, the rolled out projection is "cut" at some specific longitude. Two parameters distinguish each projection from all others:

1. The latitude, TNGLAT, at which the superimposed cone is tangent to the Earth

2. A "reference longitude", REFLON, halfway around the world from where the projection is cut (vertical in Figure 2).

The first step in initializing the PARMAP arrays is to provide this information, using the STLMBR routine:

```
        CALL STLMBR(PARMAP,TNGLAT,REFLON)
e.g.    CALL STLMBR(PARMAP, 35., -75.)   .
```

For Polar Stereographic maps, TNGLAT will be 90.0 if centered about the North Pole, −90.0 if about the South Pole. For Mercator maps, TNGLAT will

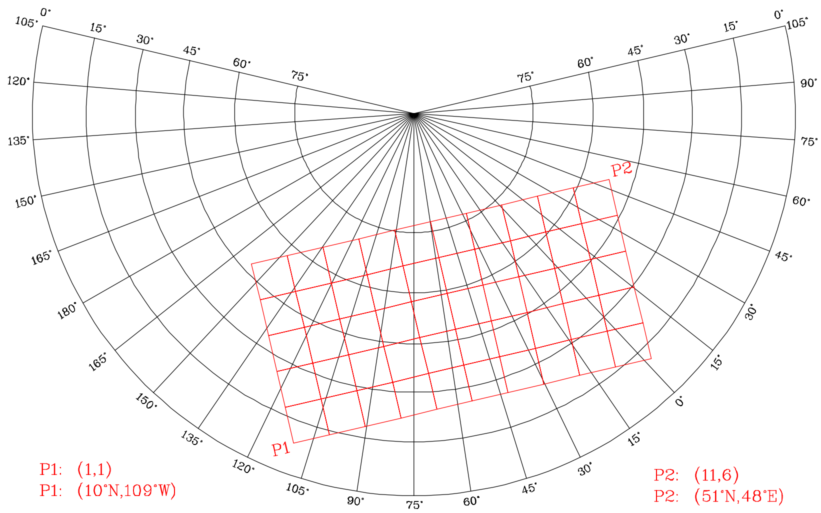

Figure 3: Illustration of two-point grid specification

be 0.0. All other values of TNGLAT, from $-90°$ to $+90°$, refer to Lambert Conformal maps.

In most published Lambert Conformal maps, the tangent latitude is not specified. Instead, the map legend will specify the scale at two "Reference Latitudes", e.g. "Scale $1 : 360,000$ at $28°$ and $41°48'$". The tangent latitude needed to specify the projection lies between the two reference latitudes and can be calculated from them. We have included a function, EQVLAT, in our software to perform this calculation. The projection can be specified to the PARMAP array through the call

```
        CALL STLMBR(PARMAP, EQVLAT(REFLT1, REFLT2), REFLON)
e.g.    CALL STLMBR(PARMAP, EQVLAT(28.0, 41.8), -75.).
```

3.2 Initialization Stage 2: Specifying the Grid

3.2.1 Two Point Specification

Once the projection is specified, an $X-Y$ coordinate system is laid down on it. Given the conformal condition that the scale must be isotropic, an orthogonal $x-y$ coordinate system can be completely defined by knowing the geographic location of any two distinct points on it. In 3, the user lays out the grid with corner points P1 (X1=1,Y1=1) and P2 (X2=11,Y2=6) located at latitude $10°N$,

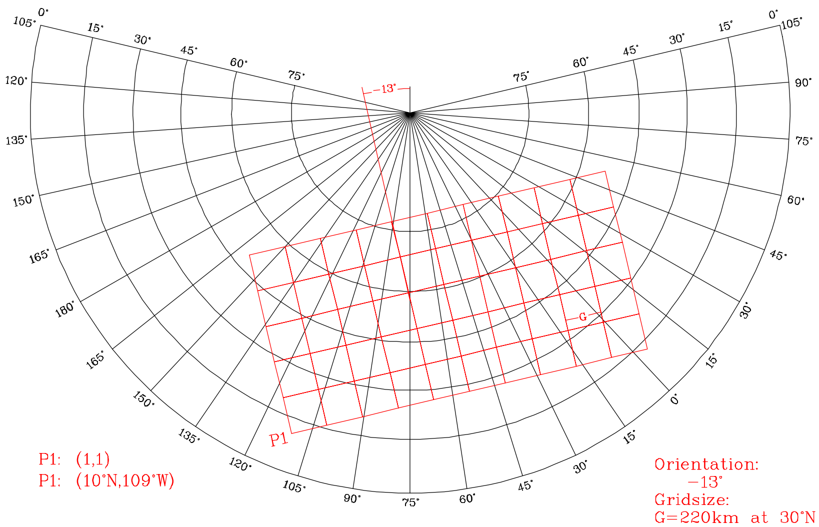

Figure 4: Illustration of one-point grid specification

longitude $109°W$ and $51°N$, $48°E$, respectively. To include this definition in the PARMAP array, the STCM2P two-point grid definition routine is used:

```
      CALL STCM2P(PARMAP,  X1,Y1,  XLAT1,XLON1,  X2,Y2,
    C XLAT2,XLON2)
e.g.    CALL STCM2P(PARMAP, 1.,1., 10.,-109., 11.,6.,
     C  51.,48.)  .
```

In general, it is not necessary that the two points be corner points, or that they have different X or different Y coordinates. It is only necessary that they be different points; i.e. that either the X-coordinates, or the Y-coordinates, or both, differ.

3.2.2 One-Point Specification

Information on grid size and grid orientation may be substituted for one of the points in the two-point specification. As shown in 4, the orientation of the grid coordinate system can be established by the angle at which the Y-coordinate lines cross the reference longitude (in degrees measured clockwise from the North direction) while the scale of the grid can be given by the grid size in km at some latitude. Instead of using STCM2P, this form of specification can be placed in PARMAP by STCM1P:

```
       CALL STCM1P(PARMAP, X1,Y1, XLAT1,XLON1, GLAT,GLONG,

     C GSZ, ORIENT)
e.g.   CALL STCM1P(PARMAP, 1.,1., 10.,-109., 30.,-75.,
     C 220., -13.)  .
```

A facility is provided for allowing the grid to be tilted with respect to the reference longitude since meteorological modelers may well choose to orient their grids along a valley, mountain range, or island chain, which may not be so obliging as to lie in an exactly North-South or East-West direction.

3.3 Initialization Examples using Standard Grids

The grid specification system described above can accommodate many grid definitions in standard usage, several of which are specified in Dey (1996). For example, the National Centers for Environmental Prediction grid 27 (the 65x65 point grid used to display forecast fields from many NCEP models) is defined as Polar Stereographic, oriented along the $80°W$ longitude, grid size 381 km at $60°N$, North Pole at $(33., 33.)$. It can be specified with the code

```
      REAL CEP27(9)
      CALL STLMBR(CEP27, 90., -80.)
      CALL STCM1P(CEP27, 33.,33., 90.,0., 60.,-80.,
     C   381., 0.).
```

NOAA's Automated Weather Interactive Processing System has defined several map grids. AWIPS grid 204 is a Mercator projection covering Hawaii. It has a gridsize of 160 km at latitude $20°N$, oriented with the Y-axis North. Its lower left point (1,1) is located at $29.26°S$, $129.47°E$, and it straddles the International Date Line. It can be specified with the code

```
      REAL AWP204(9)
      CALL STLMBR(AWP204, 0., 180.)
      CALL STCM1P(AWP204, 1.,1., -29.263,129.470,
     C  20.,180., 160., 0.).
```

It could also be specified with the code

```
      REAL AWP204(9)
      CALL STLMBR(AWP204, 0., 180.)
      CALL STCM2P(AWP204, 1.,1., -29.263,129.470,
    C 1.,71., 60.547,129.470).
```

4 Using the CMAPF functions

4.1 Coordinate Transformations

Once the projection and grid specifications have been set in a PARMAP array, the CMAPF functions can be used to transform coordinates and vectors. To

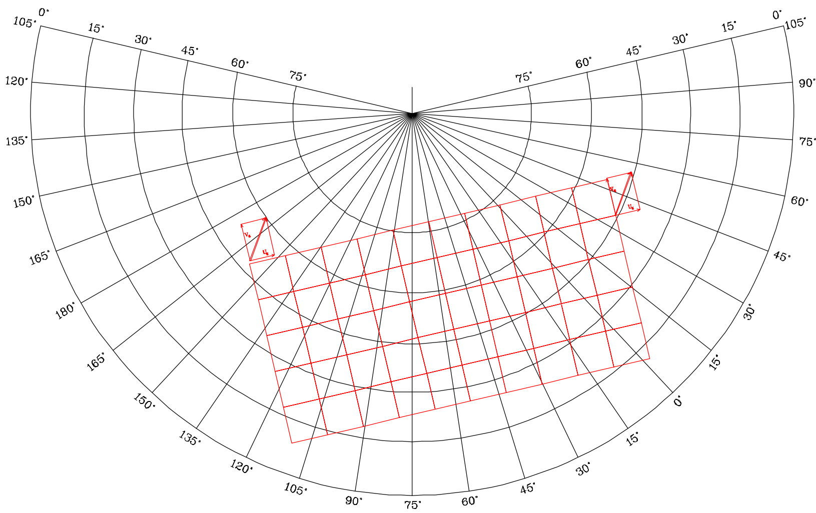

Figure 5: Wind vectors on a grid

transform between geographic coordinates (lat-long coordinates) and the X-Y coordinates of the PARMAP grid, use

```
       CALL CXY2LL(PARMAP, X,Y, XLAT,XLONG)
and    CALL CLL2XY(PARMAP, XLAT,XLONG, X,Y).
```

The first returns, as XLAT and XLONG, the latitude and longitude of the X,Y point on the PARMAP grid, while the second returns the (x,y) coordinates on that grid of the point whose latitude and longitude are XLAT, XLONG.

When working with two grids concurrently, it will often be necessary to relate the position of points in one grid relative to another. To find the coordinates on the AWP204 grid of the point whose coordinates on the CEP27 grid are XN,YN, these calls are combined as follows:

```
       CALL CXY2LL(CEP27, XN,YN, XLAT,XLONG)
       CALL CLL2XY(AWP204, XLAT,XLONG, X,Y)
```

The first call returns the latitude and longitude of the point, and the second returns the X,Y coordinates in the AWP204 coordinate system.

4.2 Vector (Wind) Transformations

Winds are normally given with reference to the North-South and East-West directions, but as illustrated in Figure 5, vectors with the same components $(U_g,\ V_g)$ relative to the grid, taken at two different points of the grid, will generally not have the same compass direction. To relate compass wind vectors to the grid wind vectors is equivalent to rotation by an amount depending on the projection and on the location of the wind. This position may be known either in terms of latitude and longitude or of grid coordinates; to avoid confusion, we have supplied a subroutine for each contingency.

To convert wind components from compass-oriented to grid-oriented coordinates or vice-versa, use one of the routines

```
        CALL CC2GXY (PARMAP, X,Y, UE,VN, UG,VG)
        CALL CC2GLL (PARMAP, XLAT,XLONG, UE,VN, UG,VG)
        CALL CG2CXY (PARMAP, X,Y, UG,VG, UE,VN)
        CALL CG2CLL (PARMAP, XLAT,XLONG, UG,VG, UE,VN).
```

The CC2G.. routines accept Compass-oriented wind components (UE,VN) (UE in the East direction, VN in the North Direction), and return Grid oriented components (UG,VG) according to the grid specified in PARMAP (UG in the grid's $X-$direction and VG in the grid's $Y-$direction). The CG2C.. routines reverse the transformation, converting from Grid-oriented to Compass-oriented components. The C...XY routines accept position information in Grid coordinates X, Y, while the C...LL routines accept position information in geographic coordinates XLAT,XLONG. Note: within one degree of each pole, the Compass oriented wind components are oriented according to a "compass" oriented with the "North" direction (the y or v axis) in the direction of the prime meridian (longitude $0°$). See Appendix B.

4.3 Gridsize

The previously described routines accept wind components in km per hour and return them in the same units. The magnitude of the wind does not change.

To determine the distance in grid steps moved by material in a unit of time, or the value of a pressure height gradient between neighboring grid points, one must divide or multiply, as appropriate, by the size of a grid step. This size changes with position. To obtain the gridsize at a given point, use

```
        REAL FUNCTION CGSZXY (PARMAP, X,Y)
  or    REAL FUNCTION CGSZLL (PARMAP, XLAT,XLONG),
```

depending on whether the position of the point is known in $x-y$ or geographic coordinates. Each function returns the gridsize in km at the given location.

4.4 Curvature (Scale Gradient)

The differences between the Modeling Equations of Section 2 and those that would hold if the Earth were flat amount to multiples of the vector $\mathbf{G} = \sigma^{-1}\boldsymbol{\nabla}\sigma$, where σ denotes the gridsize, and the gradient operator is defined by $\boldsymbol{\nabla} = \sigma^{-1}(\partial/\partial X, \partial/\partial Y)$. This vector, whose units are radians per km, represents the difference between a path's apparent curvature on the map and its actual

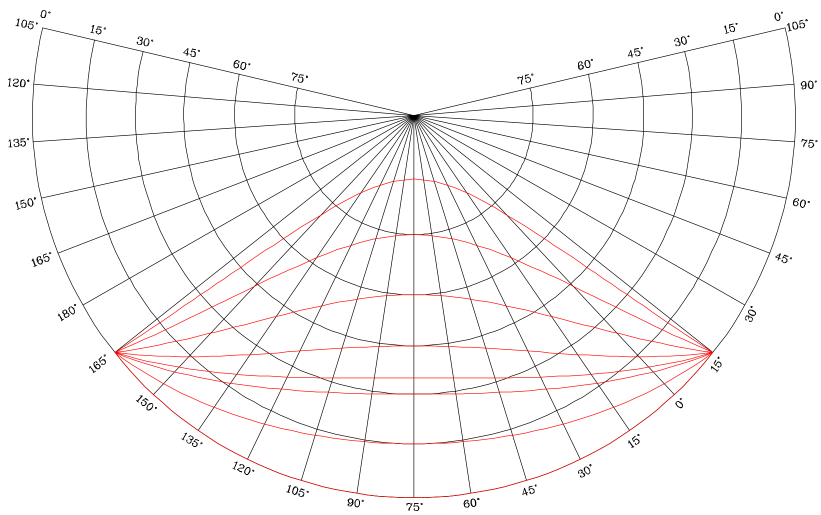

Figure 6: A series of geodesics on a conformal map.

curvature on the Earth. Its appearance in the equations of motion is due to this curvature shift, and the extent of its influence may be estimated by the curvature of geodesics on the map.

Geodesics (Great Circles) are paths without curvature on the Earth's surface, and ideally should be represented by straight lines on the map. This is not possible for a conformal map, and the projection process endows geodesics with a slight apparent curvature on the map. This "induced curvature" of a geodesic at a given point is limited by the magnitude of **G**. The vector is directed toward the tangent latitude circle, and on this circle it is equal to zero.

The induced curvature on a geodesic attains this maximum only where the geodesic is perpendicular to the vector **G**; otherwise the actual curvature vector is the projection of **G** normal to the geodesic, i.e. $\boldsymbol{\kappa} = \mathbf{G} - (\mathbf{t} \cdot \mathbf{G})\,\mathbf{t}$ where $\mathbf{t}$ is the unit vector tangent to the geodesic. Figure 6 shows a sequence of geodesics on a Lambert Conformal projection whose tangent latitude is $35°N$.

The induced curvature is a prime source of distortion of geographic features on the map and may be minimized for a particular area by proper selection of the tangent latitude. Figure 7 shows the induced curvature as a function of latitude for various choices of tangent latitude.

Subroutines are provided to supply the components of the curvature vector **G**. Use

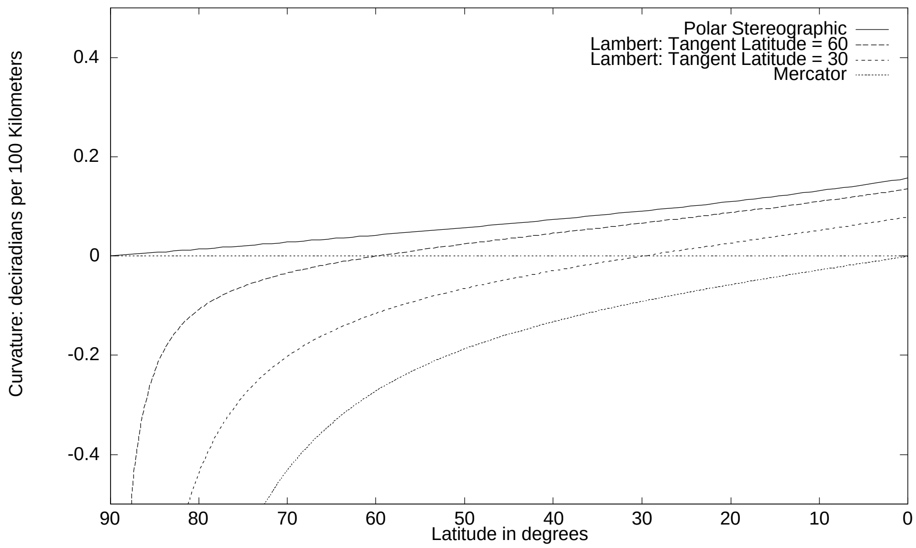

Figure 7: Maximum induced curvature for various tangent latitudes.

```
      CALL CCRVXY (PARMAP, X,Y, GX, GY)
or    CALL CCRVLL (PARMAP, XLAT,XLONG, GX, GY)
```

to obtain the curvature components given grid position (X,Y) or geographic location (XLAT,XLONG), respectively. Units of GX, GY are in radians per km. They are grid-oriented (directed along the X- and Y- axes of the grid) rather than compass oriented (directed in the North or East directions), since these vectors are in fact directed North-South toward the tangent latitude. If desired, the compass components could be obtained using CG2CXY or CG2CLL.

4.5 Polar Axis

To evaluate the effects of the Coriolis Force in Meteorological models, it is useful to know the components of a unit vector along the rotational axis of the Earth. The vertical component of this vector, multiplied by twice the angular velocity of the Earth's rotation, yields the Coriolis parameter f which governs the rotation of momentum in the horizontal plane. If the full Coriolis force is included (Coriolis exchanges between the vertical and horizontal velocities), the appropriate terms involve the same multiple of the other two components of this vector. To evaluate these components, use

```
      CALL CPOLXY (PARMAP, X,Y, ENX, ENY,ENZ)
or    CALL CPOLLL (PARMAP, XLAT,XLONG, ENX, ENY,ENZ).
```

Either subroutine returns, as ENX, ENY and ENZ, the components of the unit North Polar vector in the x-, y- and z- directions, respectively.

5 Using the C code version

C source code is also available. The syntax is virtually identical to the FORTRAN version, with the following language-related variations:

Prototypes and type definitions are contained in the cmapf.h header file, which should be #included in any program:

```
#include ''cmapf.h''
```

Storage for the map parameter block must be reserved by a declaration statement using typedef maparam (defined in cmapf.h):

```
maparam parmap;
```

The function and subroutine names and parameters of the FORTRAN version are replicated identically in type and meaning, except that the map parameter blocks, and any parameters which are to be returned, are passed by reference (&) rather than by value. For example, in

```
cxy2ll(& parmap, x,y, & latit, & longit);
```

x and y are passed by value, but parmap, latit and longit are passed by reference.

The C counterpart of a FORTRAN SUBROUTINE is a function returning void, but the C counterpart of a FORTRAN function is a function of the same type.

For reasons of coding convenience, all parameters and non-void functions in the C version of CMAPF are of type "double" as specified in the prototypes in the cmapf.h header file. There is little practical impact in this choice: single precision would be quite adequate for meteorological modeling since granularity of single precision latitude and longitude coordinates amounts on most machines to 2-5 meters in ground position.

6 References

Dey, C.H. 1996, Office Note 388 (GRIB) The WMO Format for the Storage of Weather Product Information and the Exchange of Weather Product Messages in Gridded Binary Form: National Centers for Environmental Prediction, Suitland, MD, 82 p.

Haltiner, G.J. and Williams, R.T., 1980, Numerical Prediction and Dynamic Meteorology, 2nd ed.: Wiley, New York, 477 p.

McConnell, A.J., 1957, Applications of Tensor Analysis: Dover, New York, 318 p.

Middleton-Link, A., 1993, NCAR Graphics Contouring and Mapping Tutorial: National Center for Atmospheric Research, Boulder, Colorado, 429 p.

Richardus, P. and Adler, R.K., 1972, Map Projections For Geodesists, Cartographers and Geographers: North-Holland American Elsevier, New York, 174 p.

Stackpole, J.D., 1988, W3FB06, W3FB07, W3FB08, W3FB09, W3FB11, W3FB12, subroutines available by anonymous ftp from URL ftp://nic.fb4.noaa.gov/pub/nws/nmc/codes/ijll/.

U.S. Naval Weather Service Command, 1989, Meteorological Station Report, Description and Instrumentation, USAP South Pole Station, Annual Report 1/18/89: Fleet Numerical Meteorology and Oceanography Detachment, Asheville, North Carolina, 2p.

World Meteorological Organization, 1988, Publication No. 306, Manual on Codes, Vol. 1, Part A: Secretariat of the WMO, Geneva, Switzerland, 598 p.

A Equations of the Conformal Transform

We present here a brief overview of the mathematical basis of the CMAPF subroutines. A more complete and detailed discussion of the mathematics of map transformations, including the effects of the non-spherical Earth, non-polar map orientations, and non-conformal projections, can be found, e.g. in Richardus & Adler (1972).

A conformal transform is a mapping from points on the globe to points on a plane in which the scale at any point is isotropic. As shown in Figures 1 and 2, a circle of latitude at latitude ϕ, with a circumference of $2\pi a \cos(\phi)$ (where a is the radius of the Earth), is mapped into an arc of radius $\rho(\phi)$, say, with a length of $2\pi\gamma\rho$. Then the scale in the East-West direction is

$$\mu_E = \frac{\gamma\rho}{a\cos(\phi)} \quad .$$

In the North-South direction, the scale is

$$\mu_N = \frac{1}{a}\frac{\partial\rho}{\partial\phi} \quad .$$

Equating the two scales, we see that ρ, as a function of latitude ϕ, is given by the solution of

$$\frac{1}{\gamma\rho}\frac{\partial\rho}{\partial\phi} = \frac{1}{\cos(\phi)}$$

If we define the Mercator coordinate y_m as

$$y_m(\phi) = \int \frac{d\phi}{\cos(\phi)} = \frac{1}{2}\ln\left(\frac{1+\sin(\phi)}{1-\sin(\phi)}\right) \quad , \tag{9}$$

we have

$$\rho(\phi) = \frac{\exp(-\gamma y_m)}{\gamma} \tag{10}$$

choosing the arbitrary constant so that the scale at the equator will always be $1/a$.

The scale μ , equal to both μ_N and μ_E, is given by

$$\mu(\phi) = \frac{\exp(-\gamma y_m)}{a\cos(\phi)} \quad , \tag{11}$$

and the gridsize, as we have defined it, is just $\sigma = 1/\mu$.

When a Lambert Conformal projection is specified with two reference latitudes ϕ_1 and ϕ_2, we must choose γ as the solution of $\mu(\phi_1) = \mu(\phi_2)$, or

$$\begin{aligned} \gamma &= \frac{\ln(\cos(\phi_1)) - \ln(\cos(\phi_2))}{y_m(\phi_2) - y_m(\phi_1)} \\ &= \frac{\ln\left(\frac{1-\sin(\phi_1)}{1-\sin(\phi_2)}\right) + \ln\left(\frac{1+\sin(\phi_1)}{1+\sin(\phi_2)}\right)}{\ln\left(\frac{1-\sin(\phi_1)}{1-\sin(\phi_2)}\right) - \ln\left(\frac{1+\sin(\phi_1)}{1+\sin(\phi_2)}\right)} \end{aligned} \tag{12}$$

The logarithmic gradient of the gridsize, i.e. the curvature induced on geodesics by the map transform, is given by

$$-\frac{1}{a\mu}\frac{d\mu}{d\phi} = \frac{\gamma}{a}\frac{dy_m}{d\phi} = \frac{\gamma - \sin(\phi)}{a\cos(\phi)} \tag{13}$$

and is zero when $\sin(\phi) = \gamma$. This latitude is also the latitude at which the cone of the projection, fitted over the globe, would be tangent to the globe (cf. Figure 1.)

The first step in the transform from geographic coordinates to general grid coordinates is to map into Cartesian coordinates of a standard, or canonical type (Figure 8). Let ξ and η be Cartesian Coordinates. Let λ_0 be the reference longitude defined above. We define our standard representation in the following way: The point on the equator at longitude λ_0 maps into the origin of the ξ, η plane, and the λ_0 meridian lies along the η axis, with ϕ increasing in the positive η direction. This implies the following coordinate transformation:

$$\begin{aligned} \eta &= \frac{1 - \exp(-\gamma y_m)\cos\gamma(\lambda - \lambda_0)}{\gamma} \\ \xi &= \exp(-\gamma y_m)\frac{\sin\gamma(\lambda - \lambda_0)}{\gamma} \end{aligned} \tag{14}$$

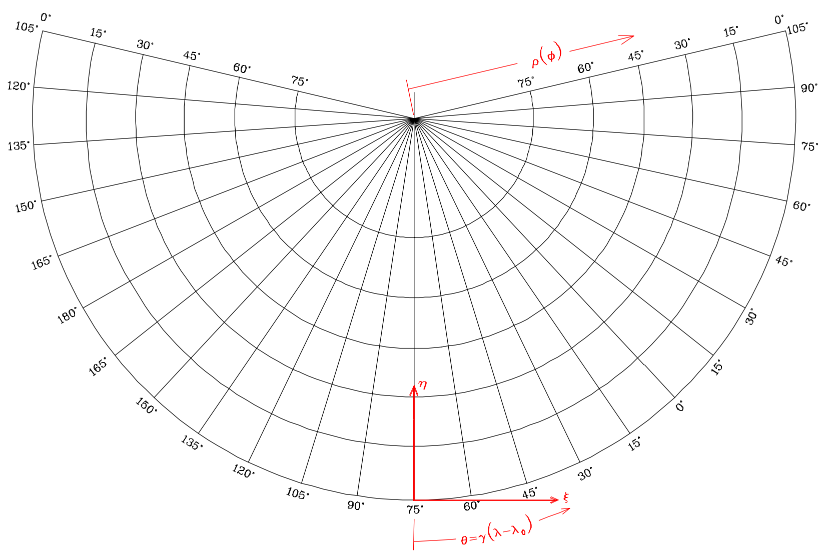

Figure 8: Canonical Coordinate system for Conformal Maps

and the inverse transformation:

$$\begin{aligned} y_m &= -\frac{1}{2\gamma} \ln \left[(1-\gamma\eta)^2 + (\gamma\xi)^2 \right] \\ \lambda &= \lambda_0 + \frac{1}{\gamma} \tan^{-1} \left(\frac{\gamma\xi}{1-\gamma\eta} \right) \quad . \end{aligned} \tag{15}$$

Letting y_m go to $\pm\infty$ shows that one of the poles is located at the point $(\xi, \eta) = (0, 1/\gamma)$. When $\gamma > 0$, this point is the North Pole, but when $\gamma < 0$, it is the South Pole. The other Pole does not map into a finite point in the (ξ, η) plane. Letting γ tend to zero (the tangent latitude tends to the equator, resulting in a Mercator projection), the (ξ, η) coordinates tend to $(\lambda - \lambda_0, y_m)$, i.e. the η coordinate is the Mercator coordinate defined above, while the ξ coordinate is longitude in radians, and neither Pole maps into a finite point in the (ξ, η) plane.

To complete the transformation to the grid, we make a general orthogonal linear transformation from (ξ, η) to (x, y). The general form for such a transformation is

$$\begin{aligned} x &= x_0 + \frac{a}{G_0} (c_1\xi + c_2\eta) \\ y &= y_0 + \frac{a}{G_0} (c_1\eta - c_2\xi) \end{aligned} \tag{16}$$

where G_0 is the gridsize at the equator, x_0 and y_0 are the grid coordinates of the origin of the (ξ, η) system, and $(c1,\ c2)$ are the coordinates in the (ξ, η) plane of the unit vector in the x- direction of the grid.

The inverse transformation is

$$\begin{aligned} \xi &= \frac{G_0}{a} \left[c_1 (x - x_0) - c_2 (y - y_0) \right] \\ \eta &= \frac{G_0}{a} \left[c_1 (y - y_0) + c_2 (x - x_0) \right] \end{aligned} \tag{17}$$

and the gridsize of the combined transformation is given by $G_0 \ / \ (a\mu(\phi))$.

To describe completely a given grid, we need to know the seven terms γ, λ_0, G_0, x_0, y_0, c_1, and c_2. The first two are provided by the initial call to STLMBR.

To find the remaining terms from the information provided to STCM2P, let (ξ_a, η_a) and (ξ_b, η_b) be the ξ, η coordinates of two distinct points, while (x_a, y_a) and (x_b, y_b) are the $x,\ y$ coordinates of the same two points. Then we make the following calculations in the order listed:

$$\begin{aligned} d_x &= \sqrt{(x_a - x_b)^2 + (y_a - y_b)^2} \\ d_\xi &= \sqrt{(\xi_a - \xi_b)^2 + (\eta_a - \eta_b)^2} \\ G_0 &= \frac{a d_x}{d_\xi} \end{aligned}$$

$$
\begin{aligned}
c_1 &= \frac{(x_a - x_b)(\xi_a - \xi_b) + (y_a - y_b)(\eta_a - \eta_b)}{d_x d_\xi} \\
c_2 &= \frac{(x_a - x_b)(\eta_a - \eta_b) - (y_a - y_b)(\xi_a - \xi_b)}{d_x d_\xi} \\
x_0 &= x_a - \frac{(c_1 \xi_a + c_2 \eta_a) d_x}{d_\xi} \\
y_0 &= y_a - \frac{(c_1 \eta_a - c_2 \xi_a) d_x}{d_\xi}
\end{aligned} \tag{18}
$$

Using instead the information provided to STCM1P, let ϕ_G be the latitude at which the grid should have size G, and let ϑ be the (clockwise, or compass) angle the y- axis makes with the reference meridian λ_0. Then we make the following calculations:

$$
\begin{aligned}
G_0 &= G a \mu(\phi_0) \\
c_1 &= \cos(\vartheta) \\
c_2 &= \sin(\vartheta) \\
x_0 &= x_a - \frac{a}{G_0}(c_1 \xi_a + c_2 \eta_a) \\
y_0 &= y_a - \frac{a}{G_0}(c_1 \eta_a - c_2 \xi_a)
\end{aligned} \tag{19}
$$

and obtain all information needed to define this particular modeler's grid.

To convert wind components from Meteorological (North- and East-) to Grid-based ($x-$ and $y-$ direction), all that is needed is the Grid-based components (N_x, N_y) of the unit vector in the North- direction. Then, if (u_g, v_g) are the grid based components, while (u_E, v_N) are the Meteorological components, we have

$$
\begin{aligned}
u_E &= N_y u_g - N_x v_g \\
v_N &= N_x u_g + N_y v_g
\end{aligned}
$$

and the inverse relationship

$$
\begin{aligned}
u_g &= N_y u_E + N_x v_N \\
v_g &= -N_x u_E + N_y v_N
\end{aligned}
$$

To obtain the North vector components, at a location given by grid coordinates (x, y), we note that the Pole is located at the point $(\xi, \eta) = (0, 1/\gamma)$. Converting to (x-, y-) coordinates, performing a vector subtraction and multiplying by , we find

$$
\begin{aligned}
N_x &= c_2 - \gamma \frac{G_0}{a}(x - x_0) \\
N_y &= c_1 - \gamma \frac{G_0}{a}(y - y_0)
\end{aligned} \tag{20}
$$

is proportional to the desired vector; to normalize, we divide by $\left(N_x^2 + N_y^2\right)^{1/2}$. Note that, for $\gamma < 0$, the multiplication by γ reverses the direction of the vector, directing it from pole to point instead of point to pole. Since the pole is now the South pole, the resultant vector is still directed North.

Given the location in latitude and longitude coordinates, rather than $(x,\ y)$ coordinates, it is not necessary to compute the location in $(x,\ y)$ or (ξ, η) coordinates. The meridian is inclined at an angle $\gamma\left(\lambda - \lambda_0\right)$ counter-clockwise to the axis. The normalized $(N_x,\ N_y)$ coordinates are then

$$\begin{aligned} N_x &= -c_1 \sin\left(\lambda - \lambda_0\right) + c_2 \cos\left(\lambda - \lambda_0\right) \\ N_y &= c_1 \cos\left(\lambda - \lambda_0\right) + c_2 \sin\left(\lambda - \lambda_0\right) \end{aligned} \tag{21}$$

B Behavior at the Pole

As noted in Appendix A, the pole is located, in (ξ, η) coordinates, at $(0,\ 1/\gamma)$, where the North Pole is meant for $\gamma > 0$, the South Pole for $\gamma < 0$. The other pole, and both poles in the Mercator case $(\gamma = 0)$, is not at any finite location. On transformation to model grid coordinates, the pole is still placed at some finite location.

However, except for the Polar Stereographic cases, $\gamma = \pm 1$, it makes little sense to include the pole in the area being modeled. In the first place, the gridsize at the pole is zero except for the Polar Stereographic cases.

Indeed, when ρ, the non-dimensional map distance from the pole, satisfies $\rho << |\gamma|^{-1}\left(\left(1-\gamma\right)\left(1+\gamma\right)\right)^{\gamma/2}$, σ tends to 0 like $2a\left(|\gamma\rho|\right)^{1/|\gamma|-1}$ as $\rho \to 0$. In the same region, the magnitude of the induced curvature vector tends to ∞ like $\left(1/\left(2a\right)\right)\left(1-|\gamma|\right)|\gamma\rho|^{-1/\gamma}$ as $\rho \to 0$.. Thus, as $\gamma \to 1$ (i.e. tends to the stereographic case), the rate of increase of the curvature vector steepens, while the region in which this occurs shrinks to a point.

Except for the Polar Stereographic case, wind directions cannot be defined at the pole. By taking a sequence of geodesics approaching the pole (cf. Figure 6), it is clear that the limiting geodesic through the pole bends through an angle of $2\pi\left(1-\gamma\right)$ radians as it passes through the pole. Thus, unless $\gamma = 1$, the direction of that geodesic at the pole is undefined.

In the exceptional case of the Polar Stereographic projection, it is possible, and often done, to include the Pole in the region being modeled. The gridsize at the Pole is twice that at the equator, and the induced curvature vector is zero. There are no bends in the geodesics, and in the grid coordinate system, vector directions are well defined.

For this projection, the only way in which the pole is treated as a special case is the reporting, in standard meteorological coordinates, of vector quantities such as wind, since compass directions have no meaning for winds there. International standards for reporting wind direction at (or within 1° of) the North Pole are governed by Code Table 878 of the WMO Manual on Codes

(WMO 1988). However, the Manual on Codes is silent on standards for the South Pole.

In essence, the North Pole convention is to orient a compass face so that the 0° (or North) index is aligned with the prime (0°) meridian. Winds are reported according to that compass face, so that a wind blowing from a given meridian is given a direction according to the longitude of that meridian, if a west longitude, or 360° minus that longitude, if east.

One would expect the South Pole to be treated analogously, orienting a compass face with the 0° (North) index along a specific meridian. This meridian could be (i) the prime (0°) meridian, (ii) the ($180^\circ E$) meridian, or (iii) some other meridian. Observations made at the South Pole Station have been made according to (i) (c.f. (Naval Weather Service, 1988, Block D)). Analyses and Forecasts in the form of gridded data, however, as reported in gridded binary (GRIB) form by the National Weather Service, appear to be reported according to (ii) (Dey, 1996, Section 1, pp 12-13). Actually, the latter reference contradicts itself by asserting both that their practice is (i) and that it is (ii), but the implementation described, taking the limit as the pole is approached along the 180° meridian, conforms with (ii). Thus, at the South Pole, gridded data wind reports and observations may be rotated 180° with respect to each other. Users of South Pole vector data must exercise caution.

In our routines, we assume Meteorological data aligned according to (i). However aligned, Meteorological vector components are obtained from a right hand set of $x-y$ axes oriented with the positive y-axis in the "North" direction of the compass face, and vector components are reported according to these axes, i.e. $U_E = -U\sin(d)$, $V_N = -U\cos(d)$, where U is the magnitude of the wind and d is the direction defined above.

C SUBROUTINE Summary

Initialization calls to establish a particular projection form:

```
      SUBROUTINE STLMBR (PARMAP,TNGLAT,REFLON)
      REAL FUNCTION EQVLAT (REFLT1, REFLT2)
```

Initialization calls to establish a particular scale and orientation:

```
      SUBROUTINE STCM2P (PARMAP, X1,Y1, XLAT1,XLON1,
     C    X2,Y2, XLAT2,XLON2)
      SUBROUTINE STCM1P (PARMAP, X1,Y1, XLAT1,XLON1,
     C    GLAT,GLONG, GSZ, ORIENT)
```

Coordinate transformation calls:

```
      SUBROUTINE CXY2LL (PARMAP, X,Y, XLAT,XLONG)
      SUBROUTINE CLL2XY (PARMAP, XLAT,XLONG, X,Y)
```

Vector wind transformation calls:

```
      SUBROUTINE CC2GXY (PARMAP, X,Y, UE,VN, UG,VG)
      SUBROUTINE CC2GLL (PARMAP, XLAT,XLONG, UE,VN, UG,VG)
      SUBROUTINE CG2CXY (PARMAP, X,Y, UG,VG, UE,VN)
      SUBROUTINE CG2CLL (PARMAP, XLAT,XLONG, UG,VG, UE,VN)
```

Gridsize evaluation:

```
      REAL FUNCTION CGSZXY (PARMAP, X,Y)
      REAL FUNCTION CGSZLL (PARMAP, XLAT,XLONG)
```

Curvature vector evaluations:

```
      SUBROUTINE CCRVXY (PARMAP, X,Y, GX, GY)
      SUBROUTINE CCRVLL (PARMAP, XLAT,XLONG, GX, GY)
```

North Polar Vector evaluations:

```
      SUBROUTINE CPOLXY (PARMAP, X,Y, ENX, ENY,ENZ)
      SUBROUTINE CPOLLL (PARMAP, XLAT,XLONG, ENX, ENY,ENZ)
```

C.1 Parameters used:

REAL PARMAP(9) - a 9-word array holding parameters characterizing a particular map. Must be initialized by a call to STLMBR, followed by a call either to STCM2P or STCM1P.

REAL TNGLAT - tangent latitude - the latitude at which the projection cone is tangent to the earth.

REAL REFLON - the longitude furthest from the cut in the map layout.

REAL REFLT1,REFLT2 - reference latitudes as normally given in the legend for Lambert Conformal maps. EQVLAT will return the equivalent tangent latitude.

REAL X1,Y1, X2,Y2 - x,y coordinates of anchor points which establish a particular scale and rotation.

REAL XLAT1,XLON1, XLAT2,XLON2 - latitude,longitude coordinates of anchor points which establish a particular scale and orientation.

REAL GSZ,GLAT - Gridsize in km and the latitude at which the scaled map is to attain that gridsize.

REAL GLONG,ORIENT - longitude of key meridian to orient map, and the angle between it and the y-axis.

REAL X,Y - x,y coordinates of a generic location on the map.

REAL XLAT,XLONG - latitude,longitude coordinates of a generic point on the globe.

REAL UE,VN - wind vector components in the East- and North- directions (Compass components).

REAL UG,VG - wind vector components in the x- and y- directions (Grid components).

REAL GX,GY - the geodesic curvature vector. Used to determine rates of change in direction per unit distance. See text.

REAL ENX,ENY,ENZ - direction cosines of the North Polar axis. ENX,ENY form a convenient locator of True North in grid components, while ENZ multiplied by $2\Omega = 4\pi/23.93h$ yields the local Coriolis factor.