Polynomial Coefficient Calculation

Map Projection Transformation

Fortran H. Program

Documentation

File: polyn.fort

I. Introduction

The objective of this program is to produce coefficients for polynomials which permit higher-speed transformations of data between maps. The program was prepared by John P. Snyder. The use of polynomials rather than analytical equations is normally justified only for relatively small regions, such as a portion of the United States, and for a large number of data transformations. The coefficients are calculated in this program using least squares and the forward projection formulas. Rectangular coordinates are computed from geodetic coordinates for a matrix of points equally spaced in latitude and longitude (but not necessarily the same in each direction) and falling within a spherical rectangle chosen as the limits of the region or transformation desired. Further discussion and the derivations of formulas are given in USGS Bulletin 1629, "Computer-Assisted Map Projection Research."

The next computations are to determine the various derivatives for iterating the parameters, as described in USGS Bulletin 1629 under Section 5b (6). First the terms used in summations are initialized at zero. Depending on whether the cone constant is 0, between 0 and 1, or 1, computation proceeds to formulas for the Oblique Mercator (including the Transverse Mercator), the oblique conformal conic, or the oblique Stereographic, respectively.

For the oblique conformal conic, for each of the M points, equations (5-41), (5-43), and (5-72) through (5-76), with ellipsoidal modifications in (5-147) and (5-150), are used to determine various fundamental functions, where the following symbols apply, adding primes to ρ, k, Θ, x, and y in these equations to make them consistent with those in the equations used subsequently in this program:

CZ = cos z

SZ = sin z

RHO = ρ'

SKPR = k'

AZ = λ'

THP = Θ'

XP = x'

YP = y'

Differentials are then computed from equations (5-50), (5-51), and (5-119) through (5-127), replacing k with k', with the following symbols:

II. Computations in the Program

Computations take place in the MAIN program and in several subroutines. The inversion of matrices occurs in subroutines MATINV (real) and MTCINV (complex). These two subroutines are identical except for applying complex dimensions to the elements involved in the latter. There is also a separate subroutine for calculation of rectangular coordinates from geodetic coordinates for each projection.

The user select several parameters:

(1) the mode:

0 - forward (real): geodetic to rectangular coordinates

1 - inverse (real): rectangular to geodetic coordinations

2 - rectangular to rectangular coordinates (real)

3 - forward (complex)

4 - inverse (complex)

5 - rectangular to rectangular coordinates (complex);

(2) the limiting meridians, west and east, in degrees;

(3) the limiting parallels, north and south, in degrees;

(4) the number of divisions in the longitude matrix (4 to 9)

(5) the number of divisions in the latitude matrix (4 to 9)

Note: The matrix computed in each direction is one more than the number of divisions.

(6) the projection (if modes 0, 1, 3, or 4), or the input and output projections (if modes 2 or 5), with parameters as required;

The parameters described above are printed after the computer reads them from data cards. The map projection parameters are the same as those used in the General Cartographic Transformation Package (GCTP). At present only the following projections are included (spherical and ellipsoidal):

	Proj. No.	1	2	3	4	5	6	7	8
Albers Equal-Area Conic	3	a	e^2	ϕ_1	ϕ_2	λ_o	ϕ_o	x_o	y_o
Lambert Conformal Conic*	4	a	e^2	ϕ_1	ϕ_1	λ_o	ϕ_o	x_o	y_o
Mercator*	5	a	e^2	--	--	λ_o	ϕ_1	x_o	y_o
Polyconic	7	a	e^2	--	--	λ_o	ϕ_o	x_o	y_o
Equidistant Conic	8	a	e^2	ϕ_1	ϕ_2	λ_o	ϕ_o	x_o	y_o
Transverse Mercator*	9	a	e^2	k_o	--	λ_o	ϕ_o	x_o	y_o
Modified Polyconic (IMW)	21	a	e^2	--	--	λ_o	λ_1	x_o	y_o

Parameters (prefix 1 if 1st proj.: 11, 12, 13, ...; 2 if 2nd proj.: 21, 22, 23, ...)

Parameters:

a = semimajor axis of ellipsoid, meters or other desired units

e^2 = square of eccentricity of ellipsoid (use 0 for sphere)

(ϕ_1, ϕ_2) = standard parallels, degrees, except on Mercator: ϕ_1 = latitude of true scale

λ_o = central meridian, degrees

k_o = central scale factor along central meridian

ϕ_o = latitude of origin of rectangular coordinates along λ_o, degrees

x_o = false eastings, in same units as a.

y_o = false northings, in same units as a.

* = conformal projection. Complex coefficients may be determined if no non-starred projection is involved and if same ellipsoid or sphere is used if there are two projections.

If a complex computation is prescribed, but one of the projections involved is not conformal, or the transformation is from an ellipsoidal to a spherical projection, the computation is changed to the corresponding real transformation. The modified Polyconic is not in the GCTP.

The (x,y) coordinates for each latitude/longitude (ϕ,λ) of the prescribed matrix are calculated for the input projection and for the output projection (if required), using the forward equations for the projection. (These equations are listed in USGS Bulletin 1532, except for those of the modified Polyconic, which are listed in Cartographica, 1982, v. 19, nos. 3 and 4, (1982) pp. 31-43).

The coordinates are then printed and stored in input (XP,YP) and output (XX,YY) matrices, as required for the requested transformation. Latitude and longitude are stored in radians, but the latitude is converted to isometric latitude for complex conversion if modes 3 or 4 are prescribed.

The input coordinates are averaged, and the average is subtracted from each value. Starting with a fit to a second-order polynomial, the matrix [A] (BA in the program) of equation (2-14) of Bulletin 1629 is then developed. Underflow is overcome by checking to see whether the sum of the logarithms of numbers to be multiplied is less than that of the computer limit; if so, the product is made zero. Matrix [D] (DD in the program) is developed by transposition, multiplication, and inversion (equation 2-15)). The coefficients C_n (CFA(I)) and C_n' (CFB(I)) are found as in equations (2-16) and (2-17). If coefficients are so small that their effect does not exceed 10^{-4} in units of ($\underline{x}$,$\underline{y}$) of 10^{-10} radians, they are made zero.

The coefficients are printed in a format placing those of a different order on a different line. They are then used to calculate output values (XT,YT) which are subtracted from values (XX,YY), and the root mean square of the residuals is reported.

The program then repeats the calculation for a third-order polynomial, printing coefficients and RMSE, and likewise for the fourth and fifth order. After this, it returns to the start of the program for a new transformation request. If there is none, the computation is ended.

Complex computation, if required, follows a similar pattern, but using equations (2-18) through (2-20) (A_c is called GBA, and D_c is called GDD in the program) to develop coefficients ($K_n + iK_n'$), stored as GCF(I).

The final coefficients of various orders are used in equations (2-1), (2-2) or (2-4), or in their inverse or rectangular-to-rectangular modifications as described just after these formulas. In equation (2-1), C_1 is the zero-order coefficient, C_2 and C_3 are the first-order coefficients in the order printed in the program, etc.

Operating Instructions

The first card of the input data deck must consist of various constants with the following specifications:

Card	Column	Format	Data
1	1	I1	Code for transformation*
	2	I1	Code for continuing (0 = new calculations; nonzero = stop)
	11-20	F10.4	Westernmost meridian, degrees (east is positive)
	21-30	F10.4	Easternmost meridian, degrees (east is positive)
	31-40	F10.4	Northernmost parallel, degrees (north is positive)
	41-50	F10.4	Southernmost parallel, degrees (north is positive)
	51-52	I2	Number of divisions in longitude matrix (4-9)
	53-54	I2	Number of divisions in latitude matrix (4-9)

*0 = geodetic to rectangular (real)

1 = rectangular to geodetic (real)

2 = rectangular to rectangular (2 projections) (real)

3 = same as 0, but complex (conformal projections only)

4 = same as 1, but complex (conformal projections only)

5 = same as 2, but complex (conformal projections only)

Cards 2 to 4 provide information for the input projection for transformation codes 2 or 5, or the only projection for codes 0, 1, 3, and 4:

2	1-2	I2	Number of map projection (see above listing by projection)
3	1-12	F12.2	Parameter 1 (see above listing by projection)
	13-24	F12.9	Parameter 2 (see above listing by projection)
	25-36	F12.6	Parameter 3 (see above listing by projection)
	37-48	F12.6	Parameter 4 (see above listing by projection)
	49-60	F12.6	Parameter 5 (see above listing by projection)
	61-72	F12.6	Parameter 6 (see above listing by projection)
4	1-12	F12.2	Parameter 7 (see above listing by projection)
	13-24	F12.2	Parameter 8 (see above listing by projection)

If transformation codes 2 or 5 are shown on card 1, cards 5-7 must repeat the data of cards 2-4, but for the output projection. The next card, either card 5 for codes 0, 1, 3, and 4, or card 8 for codes 2 and 5, begins a repetition of cards 1-4 for a second transformation requested, or may call for an end of calculations with 1 in column 2. An indefinite number of different transformations may be included when running one program.

Minimum-Error Projection

Selection

Fortran H Program

Documentation

File: confn.fort

I. Introduction

The objective of this program is to determine parameters for minimum-error conformal map projections when applied to the mapping of a particular region. The least-squares principle is used, and the program permits finding the minimum-error Transverse Mercator, Oblique Mercator, oblique conformal conic, or oblique Stereographic Projections, as well as complex conformal transformations of any of these. There are apparent limitations in the universality of the computations; that is, the optimum transformed pole cannot be found at the same time that several complex coefficients are being optimized, but many combinations can be computed. All these projections may be applied to the ellipsoid, for which conformal latitudes are computed. (The use of conformal latitudes for the oblique aspect is generally satisfactory, but the Transverse Mercator so obtained does not have a central meridian true to scale. The optimum central meridian and reduced scale factor for the Transverse Mercator may be determined with this program, however, and then applied to standard ellipsoidal equations (not in this program) for a practical solution.)

For derivations and formulas, see USGS Bulletin 1629, "Computer- Assisted Map Projection Research." While a less versatile program was originally used for calculating the 50-State map projection, using somewhat different equations, this program will produce the same results. This program was prepared by John P. Snyder.

II. Computation in the Program

Except for subroutines NRME and SMLE, which are used to solve three or more simultaneous equations, all computations take place in the MAIN program. After the number M of points for the least-squares fit is read, together with the eccentricity of the reference ellipsoid (0 if a sphere) and the arbitrary origin of rectangular coordinates, the latitude/longitude and relative weight of each point is entered. Next a data line or card is read with the number of complex coefficients desired, the initial estimates of the latitude/longitude of the transformed pole of the projection and of the cone constant, and certain codes. These codes indicate whether certain parameters are fixed or are to be allowed to vary. This will establish the type of base projection. For example, if the cone constant is held at 0, but the pole of projection can be moved, the optimum Oblique Mercator projection will be computed as the base or final projection, depending on whether a second- or higher-order complex transformation is included.

The next data line or card read establishes the range of latitudes and longitudes, as well as the increment in each, for a printout of x and y coordinates of points based on the projection determined by the program from parameters given.

The first computations based on parameters are those to establish the rectangular coordinates of the arbitrary origin relative to the projection frame of reference. These coordinates of the origin, which should be placed somewhere near the center of the final map, will later be subtracted from the other rectangular coordinates of the base projection for conversion to polar coordinates used in computing coefficients for complex transformation. For maps of North America, the origin was made latitude 50° N. and longitude 100° W. for the oblique conformal conic and Oblique Mercator, but it is made the center of projection of the oblique Stereographic. For the Stereographic, the coordinates of this point remain (x=0, y=0), even though its latitude and longitude may be changed upon iteration. For the other projections, the rectangular coordinates of this point vary if the projection parameters are changed with iteration.

DTDP $= \partial\theta'/\partial\phi_p$

DRDP $= \partial\rho'/\partial\phi_p$

DKPDP $= \partial k'/\partial\phi_p$

DTDL $= \partial\theta'/\partial\lambda_p$

DRDL $= \partial\rho'/\partial\lambda_p$

DKPDL $= \partial k'/\partial\lambda_p$

DTDN $= \partial\theta'/\partial n$

DRDN $= \partial\rho'/\partial n$

DKPDN $= \partial k'/\partial n$

CC $= \partial x'/\partial\phi_p$

DD $= \partial y'/\partial\phi_p$

EE $= \partial x'/\partial\lambda_p$

GG $= \partial y'/\partial\lambda_p$

HH $= \partial x'/\partial n$

HJ $= \partial y'/\partial n$

For the Oblique Mercator and oblique Stereographic, the alterations to these formulas are described in equations (5-139) through (5-146) and (5-150a) through (5-150f).

From these derivatives, equations (5-130) through (5-137) are combined with (5-112), (5-117), (5-118) and the corresponding derivatives with respect to the other parameters as described with these formulas, to form matrices shown in equation (5-138). This equation, actually several simultaneous equations, is solved using subroutines NRME and SMLE for the first adjustments of the latitude/longitude of the pole, the cone constant, and the complex coefficients, depending on what is allowed to vary. The changes in parameters are printed. Then the coordinates and derivatives are recalculated, the matrices reassembled, and new adjustments computed. When the sum of the absolute values of all the parameter changes at a given iteration no longer exceeds 10^{-5}, the iteration is considered complete.

The final parameters are printed and the RMSE is reported for the scale error of the M points. Then the rectangular coordinates and scale factor are computed and printed for each intersection on the prescribed graticule, or just the scale factor is reported for each of the M points, depending on the code entered on card M + 2 listed below.

Operating Instructions

The first data card on the input deck must consist of the following information:

Card	Column	Format	Data
1	1-3	I3	M (number of points)
	11-20	F10.8	E2 (e^2, eccentricity of reference ellipsoid; 0 if sphere)
	21-30	F10.8	CHIR* (conformal latitude of arbitrary center of coordinates, degrees, positive or no sign if north, negative if south)
	31-40	F10.8	CLONR* (longitude of arbitrary center of coordinates, degrees, positive or no sign if east, negative if west)

* ignored if oblique Stereographic

The next M cards provide latitude/longitude and relative weight of each point:

Card	Column	Format	Data
2 to M+1	1-4	F4.1	Latitude (degrees, one decimal, north latitude. If some points are south latitude, change format to F5.1 to provide room for minus sign).
	5-10	F6.1	Longitude (degrees, one decimal, positive if east, negative if west).
	11-20	F10.5	Weight. This can be square degrees times cos latitude, or other relative basis.

The next two cards are as follows:

M+2	1-2	I2	N, number of complex coefficients desired for complex transformation. For no complex transformation, enter N=1. An N of 2 means that A_1, A_2, and B_2 will be iterated. In any case B_1 remains 0.
	11-20	F10.6	PHIP, initial estimate of latitude of pole of projection, or of center of projection (if oblique Stereographic), degrees positive or no sign if north, negative if south.
	21-30	F10.6	CLONP, initial estimate of longitude of pole of projection, or of center of projection (if oblique Stereographic), degrees, positive or no sign if east, negative if west.
	31-40	F10.6	EN, initial estimate of cone constant.
	41-42	I2	KODE. Enter 0 if cone constant is to vary. Enter any other integer if cone constant is to remain fixed.
	43-44	I2	KODE1. Enter 0 if location of pole of projection can vary. Enter 1 if only the longitude of the pole can vary (e.g. for the Transverse Mercator). Enter 2 if the pole is fixed.

	45-46	I2	KODE2. Enter 0 if coordinates and scale factor for a graticule are to be printed. Enter any other integer if only the scale factors for the M points are to be printed.
M+3	11-20	F10.3	TLAT, northernmost latitude, degrees, of graticule, if graticule is to be listed with coordinates (positive or no sign if north, negative if south).
	21-30	F10.3	BLAT, southernmost latitude, degrees, of graticule.
	31-40	F10.3	WLONG, westernmost longitude, degrees, of graticule (positive or no sign if east, negative if west).
	41-50	F10.3	ELONG, easternmost longitude, degrees, of graticule.
	51-60	F10.3	DLAT, increment of latitude, degrees, to be used in listing graticule.
	61-70	F10.3	DLONG, increment of latitude, degrees, to be used in listing graticule.

If KODE2 is not zero, card (M+3) must still be inserted, but it may be blank, or shown with parameters for a future graticule.

If only one set of parameters is to be iterated for the M points, the next card is as follows.

M+4 blank or zero.

If additional projections are to be computed in the same program run (for the same M points), card (M+3) is followed by another pair of cards identical in format with (M+2) and (M+3), respectively. This may be continued indefinitely, but finally a blank card must be used to stop the program properly.