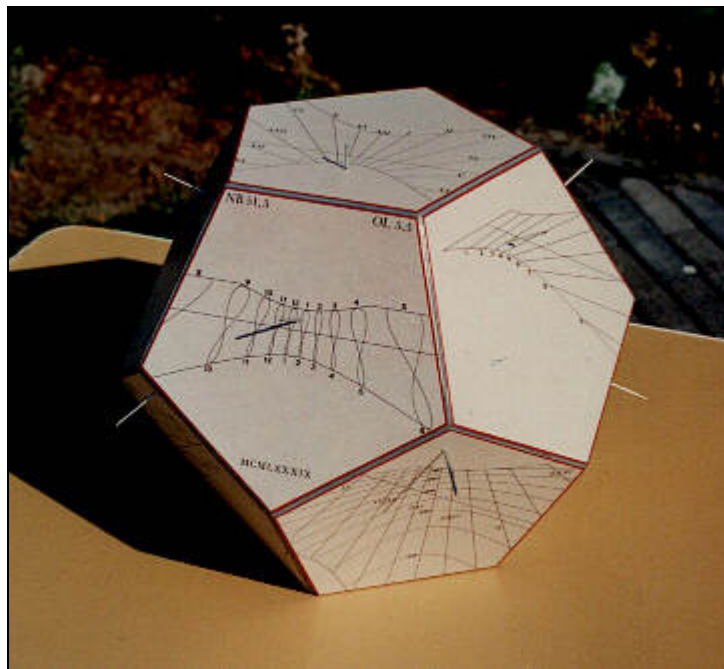# A UNIFORM METHOD TO COMPUTE FLAT SUNDIALS

usable all over the world

by fer j. de vries, netherlands

example of sundials on a dodecahedron

date: january 1997, revised october 2002

preface

In these pages you find the basic procedures to compute all kinds of lines for a flat sundial.
The procedures form a suitable basis for computer programs.
The definitions have been formulated so that the method is uniform in all situations, for both
the northern and southern hemispheres and for any plane, no matter what its orientation.

The principle of a sundial is to convert a certain sun's position into coordinates for a shadow
point on the dial's face. An uniform main procedure does this job.
Secondary procedures are used to find a number of points for any line you want to calculate.

All the drawings and pictures of sundials on these pages are computed with these procedures.

vertical south-east facing sundial at southern latitude 40°.
(phi = - 40°, i = 90°, d = - 45°)

Last Modified 02/05/2004 09:16:26

definitions

---

phi latitude of the place of the dial. -90° <= phi <= 90°, northern latitude positive, southern latitude negative.

g length of (pin) gnomon perpendicular to the dials plane. The tip of the gnomon is the shadow casting point. g > 0.

i inclination of the plane: the zenith distance of the endpoint of the (pin) gnomon. 0° <= i < 180°. (horizontal i = 0 vertical i = 90)

d declination of the plane: azimuth of the gnomon: south = 0°, positive to west, negative to east. -180° <= d <= 180°.

SM standard meridian of time zone. Greenwich = 0°, positive to west, negative to east. -180° <= SM <= 180°.

LM local meridian. (*longitude of the place of the sundial.*) Greenwich = 0°, positive to west, negative to east. -180° <= LM <= 180°.

LC longitude correction LC = SM - LM.

E equation of time in degrees. In november E is positive.
Conversion minutes of time into degrees: 1 minute = 0.25 degrees.

v height of a style relative to the dial's plane, parallel to the earth axis. -90° <= v <= 90°.

b angle of substyle, measured from y-axis. pos. anti clockwise. -180° <= b <= 180°.

ts hourangle of substyle. -180° <= ts <= 180°.

decl declination of the sun, pos. if sun in northern hemisphere, neg. in southern hemisphere. -23.°5 <= decl <= 23.°5

t hourangle of the sun: noon = 0°, positive. to west, negative to east. -180° <= t <= 180°.

x,y coordinates of shadow point. Footpoint gnomon = 0,0. x to right, y upwards. For horizontal dial x east, y north.

dn daynumber. January first at 00:00:00 is daynumber 1.0 , at 12:00:00 is daynumber 1.5.

... diverse other variables are used in the procedures as x0,y0,z0, x1, y1, z1, day, mth and so on. See the procedures.

While these definitions are, of course, arbitrary, many of them are common in gnomonics and the others are constrained by the requirement that the method should be universal all over the world.
If you want to change the definitions , you may also have to change some of the formulae, or introduce a conversion sub-routine.

English-speaking people nowadays use the word **gnomon** to refer to the whole of the line casting the shadow, but in this paper **gnomon** is used in its older and more traditional sense. Here the **gnomon** is a pin, perpendicular to the plane.
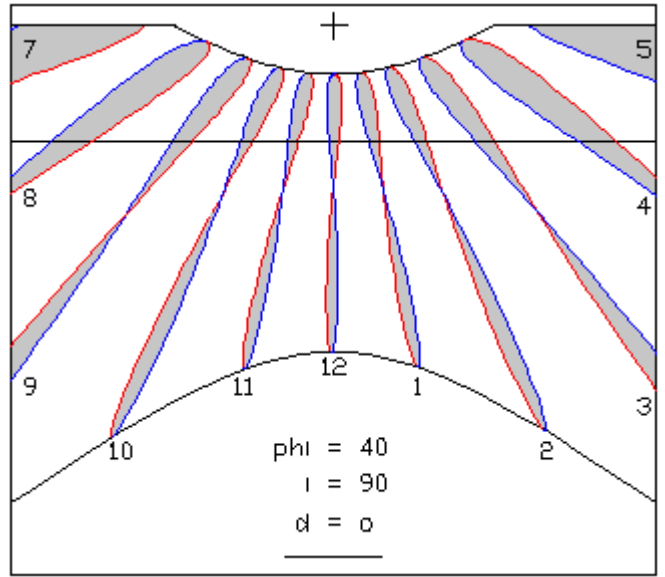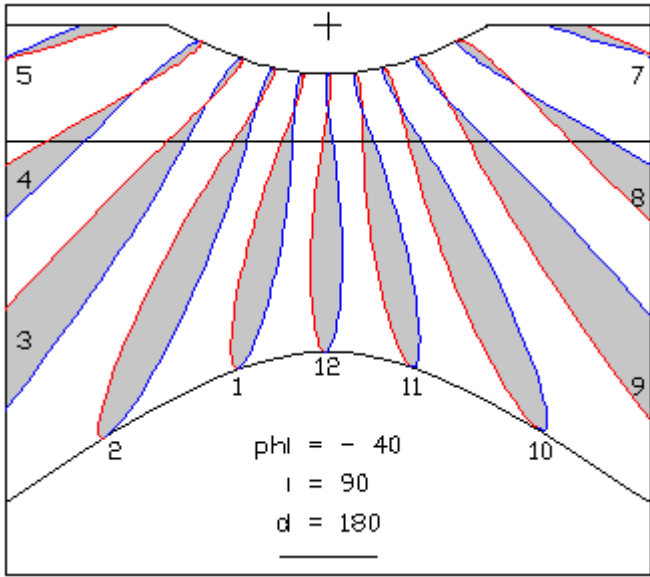
example of vertical dial at northern and southern latitude

4

main procedure

The aim of the main procedure is to convert a given sun's position, defined by the angle of declination **decl** and the sun's hourangle **t** into the coordinates **x, y** of the shadow point of the (pin) gnomon or to output 'point isn't real'.
Just simply calculate the values x0,y0,z0, x1,y1,z1, x2,y2,z2, x3,y3,z3, and finaly x,y as in the 5 routines below.
In these routines you see 2 decision points to find out if a point is or isn't real.

In : phi, g, i, d, decl, t

Out : x, y or 'point isn't real'

$x0 = \sin t \cdot \cos decl$	translation decl, t into x0, y0, z0
$y0 = \cos t \cdot \cos decl$	
$z0 = \sin decl$	

$R = 90 - phi$	translation x0, y0, z0, into x1, y1, z1
$x1 = x0$	by rotation around x-axis through angle 90 - phi
$y1 = y0 \cdot \cos R - z0 \cdot \sin R$	
$z1 = y0 \cdot \sin R + z0 \cdot \cos R$	

if z1 < 0 point isn't real: sun is beneath the horizon.

$R = d$	translation x1, y1, z1, into x2, y2, z2
$x2 = x1 \cdot \cos R - y1 \cdot \sin R$	by rotation around z-axis through angle d
$y2 = x1 \cdot \sin R + y1 \cdot \cos R$	
$z2 = z1$	

$R = i$	translation x2, y2, z2, into x3, y3, z3
$x3 = x2$	by rotation around x-axis through angle i
$y3 = y2 \cdot \cos R - z2 \cdot \sin R$	
$z3 = y2 \cdot \sin R + z2 \cdot \cos R$	

if z3 <= 0 point isn't real: sun isn't above the dial.

$x = x3 \cdot g / z3$	translation x3, y3, z3, into x, y
$y = y3 \cdot g / z3$	these are the wanted coordinates of the shadow point.

Remarks:

It is advisable to program each of the 5 routines separately rather than as one combined.

For the shadow points, you should bear in mind the desired maximum value for x and y. Otherwise you may perhaps calculate points that are very far away from your dial.

inclining and declining sundial at southern latitude

secondary procedures

On this page you find the secondary procedures to compute all kinds of lines on a flat sundial.
Make a choice for the kind of lines you want to compute.
Chose a single line or make a loop for the lines in that kind in the range as mentioned.
Then for each line make a loop for a number of points for that line.
Sometimes this will be a loop for decl, another time this will be a loop for t or for something else.
Sometimes it is easier not to make loops directly for decl or t, but to make a loop for daynumbers and then to compute the needed decl and t or get them out of a table.
In the same way you may get the equation of time if needed.
Most of the time I use a loop in daynumbers in the procedures below.
With each decl and t call the mainprocedure to get the wanted shadowpoint if real.

Lines for local suntime.
Input (range of) wanted lines u. 0 <= u <= 24
Calculate hourangle t = (u - 12) * 15
Loop for daynumbers dn from 172 to 355
Calculate decl for daynumber (dn + 0.5) *(that is noon)*
Call mainprocedure
Refinement: If you want to draw these lines to the intersection point of the (pole)style make a loop for decl from -90 to 90 instead of a loop for daynumbers. This only if v isn't 0)

Lines for local suntime with equation of time curve for period dec. 21 to june 21.
Input (range of) wanted lines u. 0 <= u <= 24
Loop for daynumbers dn from 355 to 538
If dn > 366 then dn = dn - 366
Calculate decl for daynumber (dn + 0.5)
Calculate equation of time E for daynumber (dn + 0.5)
Calculate hourangle t = (u - 12) * 15 + E
Call mainprocedure

Lines for local suntime with equation of time curve for period june 21 to dec 21.
Input (range of) wanted lines u. 0 <= u <= 24
Loop for daynumbers dn from 172 to 355
Calculate decl for daynumber (dn + 0.5)
Calculate equation of time E for daynumber (dn + 0.5)
Calculate hourangle t = (u - 12) * 15 + E
Call mainprocedure

Lines for suntime for standard meridian. *(This is time with longitude correction)*
Input (range of) wanted lines u. 0 <= u <= 24
Calculate hourangle t = (u - 12) * 15 + LC
Loop for daynumbers dn from 172 to 355
Calculate decl for daynumber (dn + 0.5)
Call mainprocedure
Refinement: If you want to draw these lines to the intersection point of the (pole)style make a loop for decl from -90 to 90 instead of a loop for daynumbers. This only if v isn't 0)

**Lines for suntime for standard meridian
with equation of time curve for period dec. 21 to june 21.**
Input (range of) wanted lines u. 0 <= u <= 24
Loop for daynumbers dn from 355 to 538
If dn > 366 then dn = dn - 366
Calculate decl for daynumber (dn + 0.5)
Calculate equation of time E for daynumber (dn + 0.5)
Calculate hourangle t = (u - 12) * 15 + E + LC
Call mainprocedure

**Lines for suntime for standard meridian
with equation of time curve for period june 21 to dec 21.**
Input (range of) wanted lines u. 0 <= u <= 24
Loop for daynumbers dn from 172 to 355
Calculate decl for daynumber (dn + 0.5)
Calculate equation of time E for daynumber (dn + 0.5)
Calculate hourangle t = (u - 12) * 15 + E + LC
Call mainprocedure

Lines for the sun's declination.
Input (range of) wanted lines for decl in the range of -23.44 to 23.44
Loop for hourangle t in the range of -180 to 180
Call mainprocedure

Lines for dates (month, day).
Input (range of) months mth
Input (range of) days day
Calculate daynumber dn :
p = int((mth + 9) / 12)
q = int(275 * mth / 9) - 2 * p + day - 30
if leapyear dn = q + p else dn = q
Calculate decl for (dn + 0.5)
Loop for hourangle t in the range of -180 to 180
Call mainprocedure

Lines for Babylonian hours. *Restriction -66.56 <= phi <= 66.56*
Input (range of) wanted lines u. 0 <= u <= 24
Loop for daynumbers dn from 172 to 355
Calculate decl for daynumber (dn + 0.5)
Calulate half daylength T = arccos(- tan phi * tan decl)
Calculate hourangle t = u * 15 - T
Call mainprocedure

Lines for Italian hours. *Restriction -66.56 <= phi <= 66.56*
Input (range of) wanted lines u. 0 <= u <= 24
Loop for daynumbers dn from 172 to 355
Calculate decl for daynumber (dn + 0.5)
Calulate half daylength T = arccos(- tan phi * tan decl)
Calculate hourangle t = u * 15 + T
Call mainprocedure

Lines for antique hours or unequal hours. *Restriction -66.56 <= phi <= 66.56*
Input (range of) wanted lines u. 0 <= u <= 12
Loop for daynumbers dn from 172 to 355
Calculate decl for daynumber (dn + 0.5)
Calulate half daylength T = arccos(- tan phi * tan decl)
Calculate hourangle t = (u - 6) * T / 6
Call mainprocedure

Lines for the sun's azimut.
Input (range of) wanted lines a. -180 <= a <= 180
Loop for the sun's height h. 0 <= h <= 90
Calculate:
x1 = sin a * cos h
y1 = cos a * cos h
z1 = sin h
R = -(90 - phi)
x0 = x1
y0 = y1 * cos R - z1 * sin R
z0 = y1 * sin R + z1 * cos R
decl = arcsin(z0)
if decl is in range from -23.44 to 23.44 then call mainroutine half away with the values
x1,y1,z1
(Start calculating x2, y2, z2 and continue the mainprocedure)
else point isn't real

Lines for the sun's height.
Input (range of) wanted lines h. 0 <= h <= 90
Loop for the sun's azimut a. -180 <= a <= 180
Calculate:
x1 = sin a * cos h
y1 = cos a * cos h
z1 = sin h
R = -(90 - phi)
x0 = x1
y0 = y1 * cos R - z1 * sin R
z0 = y1 * sin R + z1 * cos R
decl = arcsin(z0)
if decl is in range from -23.44 to 23.44 then call mainroutine half away with values x1, y1, z1
(Start calculating x2, y2, z2 and continue the mainprocedure)
else point isn't real

Lines for sidereal time for period dec. 21 to june 21.
Input (range of) wanted lines u. 0 <= u <= 24
Loop for daynumbers dn from 355 to 538
If dn > 366 then dn = dn - 366
Calculate decl for daynumber (dn + 0.5)
Right ascension RA = arcsin(tan decl / tan 23.44)
Hourangle t = u * 15 - RA
Call mainprocedure

Lines for sidereal time for period june 21 to dec 21.
Input (range of) wanted lines u. 0 <= u <= 24
Loop for daynumbers dn from 172 to 355
Calculate decl for daynumber (dn + 0.5)
Right ascension RA = 180 - arcsin(tan decl / tan 23.44)
Hourangle t = u * 15 - RA
Call mainprocedure

Lines for planetary time for period dec. 21 to june 21. *Restriction -66.56 <= phi <= 66.56*
(Remember that planetary time is different from antique or unequal time. I refer to what is
written by Joseph Drecker in his book 'Die Theorie der Sonnenuhren', 1925, page 73.)
Input (range of) wanted lines u. 0 <= u <= 12
Loop for daynumbers dn from 355 to 538
If dn > 366 then dn = dn - 366
Calculate decl for daynumber (dn + 0.5)
Right ascension RA = arcsin(tan decl / tan 23.44)
Longitude of sun LS = arcsin(sin decl / sin 23.44)
Longitude of rising point LE = LS + u * 15
Declination of rising point DE = arcsin (sin LE * sin 23.44)
Right ascension of rising point RE = arcsin (tan DE / tan 23.44)
Half day length of rising point T = arccos(- tan phi * tan DE)
Hourangle t = - T - RA + RE
Call mainprocedure

Lines for planetary time for period june 21 to dec 21. *Restriction -66.56 <= phi <= 66.56*
(Remember that planetary time is different from antique or unequal time. I refer to what is
written by Joseph Drecker in his book 'Die Theorie der Sonnenuhren', 1925, page 73.)
Input (range of) wanted lines u. 0 <= u <= 12
Loop for daynumbers dn from 172 to 355
Calculate decl for daynumber (dn + 0.5)
Right ascension of sun RA = 180 - arcsin(tan decl / tan 23.44)
Longitude of sun LS = 180 - arcsin(sin decl / sin 23.44)
Longitude of rising point LE = LS + u * 15
Declination of rising point DE = arcsin (sin LE * sin 23.44)
Right ascension of rising point RE = arcsin (tan DE / tan 23.44)
Half day length of rising point T = arccos(- tan phi * tan DE)
Hourangle t = - T - RA + RE
Call mainprocedure
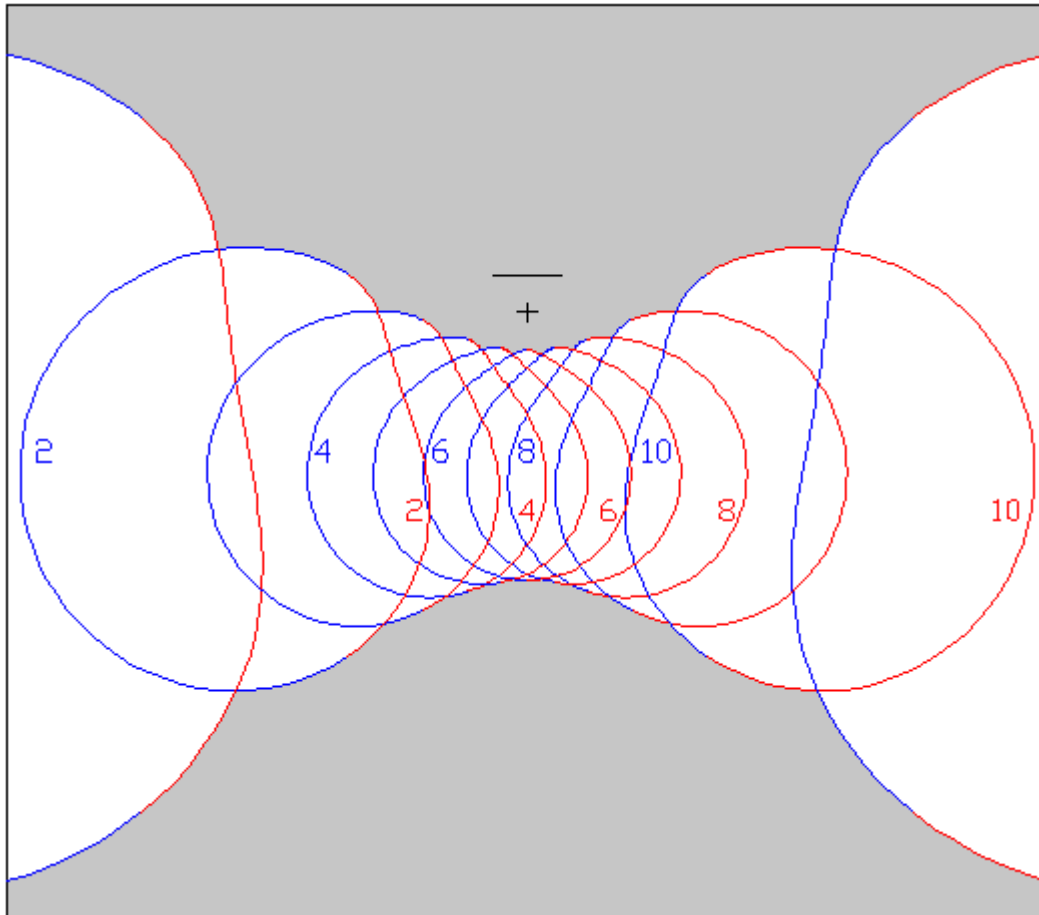
example of horizontal sundial with planetary hours for southern latitude 52.

Remarks:

The loops for daynumber are made in that way that you may choose an incremental step of 1 or 3 days, just how many points you want to calculate.

If you know a line is a straight line start with the loop for each line at one side till you find the first point and than start the loop in reverse at the other side to find the second point.

If the line isn't a real one, in both cases you will find no real point.

(Refinement: if you use the formulae to compute the sun's declination and the equation of time as given in the paragraph 'end notes' you will be a little more accurate if you calculate the values with dn + 0.5 + LM / 360.
In that way you get the values at your local noon.)

end notes

Not only shadow points have to be computed but also the height of the (pole) style and the angle for the substyle.
This also can be done with the same mainprocedure.

To calculate these constants of the dial use these 5 routines with input decl = 90° and t = 0°, but don't use the 2 decision points.
Then v = arcsin(z3). (no test of quadrant is necessary)

If v = 0 then the style is parallel to the dial's face and there isn't an intersectionpoint of the style with the dial
else: the coordinates of the intersectionpoint are x,y.
The substyle can be drawn from the footpoint of the gnomon (point 0, 0) to this intersectionpoint.

The value of v equals with the latitude where the sundial's plane becomes a horizontal plane.
Also remember that if v > 0 the endpoint of the style points to the north pole and the hourlines run clockwise but if v < 0 the endpoint of the style points to the south pole and the hourlines run anticlockwise.

However not necessary to construct your sundial I also give a formula to compute the hourangle of the substyle ts.
tan ts = sin i * sin d / (cos phi * cos i + sin phi * sin i * cos d)
Controle ts for the right quadrant.

Fomulae to compute the equation of time and the sun's declination out of a daynumber:

```
L             = DAGNR*360/365.2422 - 80.412001          DEGREES

EQUATION      =   - 109.2587*SIN( L) - 428.0240*COS( L)
                  + 595.9691*SIN(2L) -   2.1295*COS(2L)
                  +   4.5072*SIN(3L) +  19.2449*COS(3L)  SECONDS (of time)

LAMBDA        = L +  0.4365*SIN( L) +   1.8636*COS( L)
                  -  0.0179*SIN(2L) +   0.0089*COS(2L)   DEGREES

EPSILON       = 23.43746                                 DEGREES

DECLINATION   = ARCSIN(SIN LAMBDA * SIN EPSILON)         DEGREES
```

Strictly these formulae are for 2014 *(a year between two leap years)* and for 12.00 UT, but for sundials you can use them during a long time.
See also 'Cousins', page 236, for such a fomula for the year 1931.
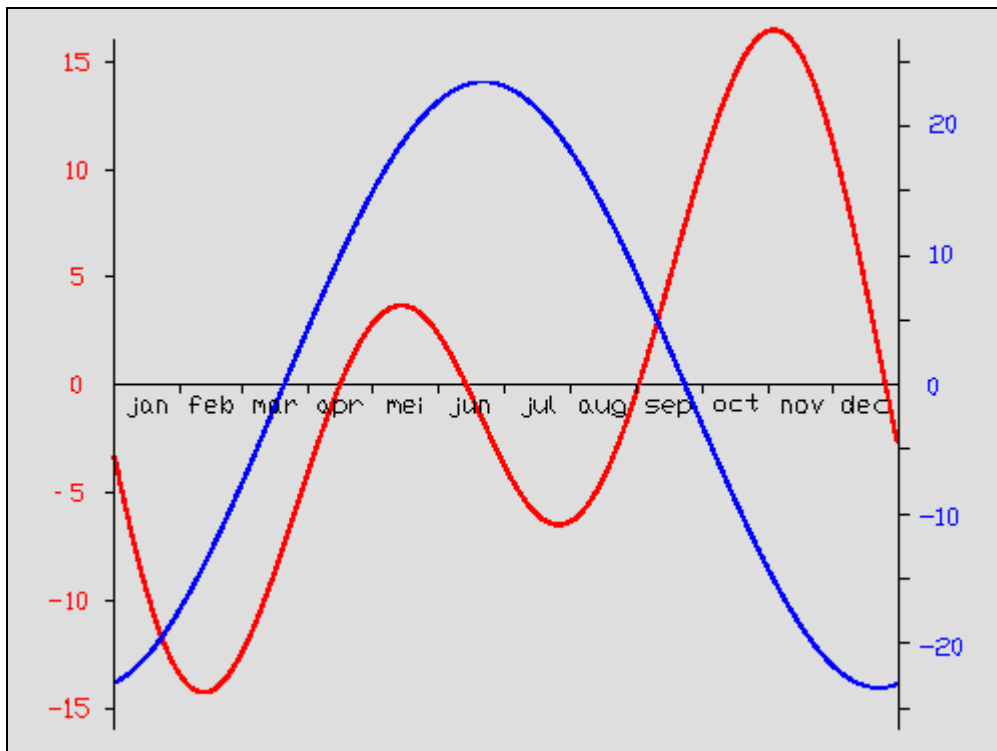
graph for the equation of time and the sun's declination