Correcting Course at Sea — Solution

Due Thursday, April 28

Step 1 — Assemble Observation Data

The assumed position:

```
In[∘]:= ϕAP = 37.50;
        λAP = -122.75;
```

The observation time, April 25, 8pm, which is April 26, 3am UT:

```
In[∘]:= timeGMT = 3;
```

The observations: Sirius, *which is a little west of south,* and Mercury, *which is a little north of west,* have these elevations:

```
In[∘]:= hOSirius = 26.42;
        hOMercury = 18.33;
```

The italicized information is needed to disambiguate inverse trig functions.

Step 2 — Get Almanac Information for Astronomical Triangle

Essentially, we need the right ascension and declination of both objects, but rather than simply needing the right ascension we need to translate those to what is known as the local hour angle. Let’s grab the declinations first. Those are used directly:

```
In[∘]:= δSirius = -16.75;
        δMercury = 21.73;
```

The Greenwich hour angle of Sirius is not tabulated. Instead, the Greenwich hour angle of the First Point of Aries is tabulated. At 3am on April 26, it is:

```
In[∘]:= tGreenwichAries = 259.1;
```

My convention (which I believe is the standard convention) is that hour angles get more and more positive as an object sets.

From this we subtract the right ascension of Sirius to get the Greenwich hour angle of Sirius:

```
In[•]:= αSirius = 101.53;
        tGreenwichSirius = tGreenwichAries - αSirius
Out[•]= 157.57
```

Finally, we add the longitude of the assumed position (with the convention that west longitudes are negative), to the Greenwich hour angle to get the local hour angle:

```
In[•]:= tSirius = tGreenwichSirius + λAP
Out[•]= 34.82
```

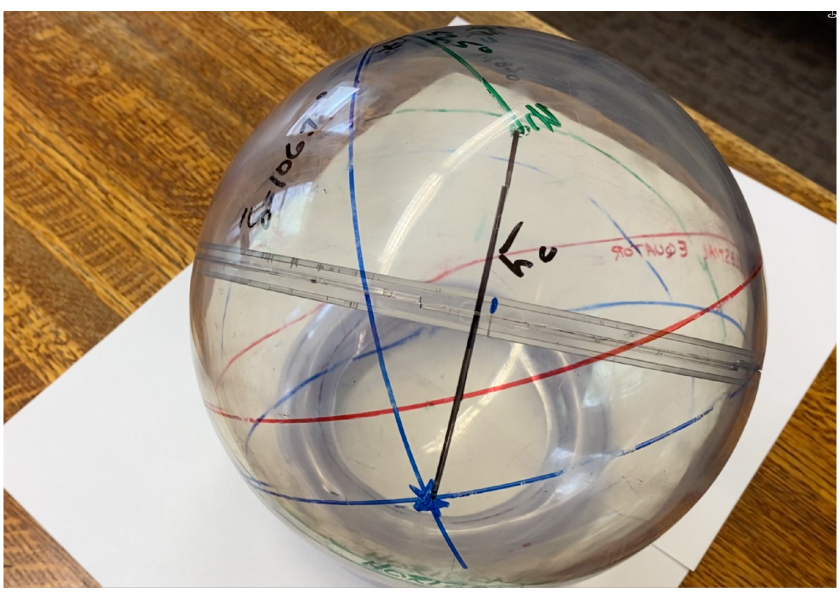
For Mercury, the job is slightly easier because its Greenwich hour angle is tabulated:

```
In[•]:= tGreenwichMercury = -153.82;
        tMercury = tGreenwichMercury + λAP + 360
Out[•]= 83.43
```

I had to add 360° to get the local hour angle of Mercury into the standard range.

Step 3 — Calculate Altitudes from the Astronomical Triangle

The astronomical triangle involves the celestial north pole, the zenith and the astronomical object. It is usually a pretty big triangle because the astronomical object is close to the horizon which makes it far from the zenith, and if it is in the south, even farther from the celestial north pole. Its size is what makes it hard to visualize from any two-dimensional representation. A video of the astronomical triangle is here: https://brianhill.github.io/heavenly-mathematics/resources/celestial-navigation/index.html. This still from the video is inadequate:



In the video you get to see the same astronomical triangle from more angles.

The upshot is that applying the Law of Cosines to the astronomical triangle, you can calculate the altitude for Sirius and Mercury:

```
In[∘]:= hCSirius = ArcSin[Sin[δSirius Degree] Sin[ϕAP Degree] +
            Cos[δSirius Degree] Cos[ϕAP Degree] Cos[tSirius Degree]] / Degree
Out[∘]= 26.63
```

```
In[∘]:= hCMercury = ArcSin[Sin[δMercury Degree] Sin[ϕAP Degree] +
            Cos[δMercury Degree] Cos[ϕAP Degree] Cos[tMercury Degree]] / Degree
Out[∘]= 18.0415
```

Step 4 — Calculate Azimuths from the Astronomical Triangle

The same astronomical triangle is now used to calculate the azimuths using the Law of Sines:

```
In[∘]:= zSirius =
         180 - ArcSin[Cos[δSirius Degree] Sin[tSirius Degree] / Cos[hCSirius Degree]] / Degree
Out[∘]= 142.291
```

The arcsin gives the wrong value for Sirius. This is disambiguated using the information in Step 1.

```
In[∘]:= zMercury =
         ArcSin[Cos[δMercury Degree] Sin[tMercury Degree] / Cos[hCMercury Degree]] / Degree
Out[∘]= 76.0624
```

For objects in the west, these angles need to be subtracted from 360° to get compass headings:

```
In[∘]:= aSirius = 360 - zSirius
Out[∘]= 217.709
```

```
In[∘]:= aMercury = 360 - zMercury
Out[∘]= 283.938
```

Step 5 — Calculate Altitude Discrepancies

```
In[∘]:= (hOSirius - hCSirius) * 60
Out[∘]= -12.598
```

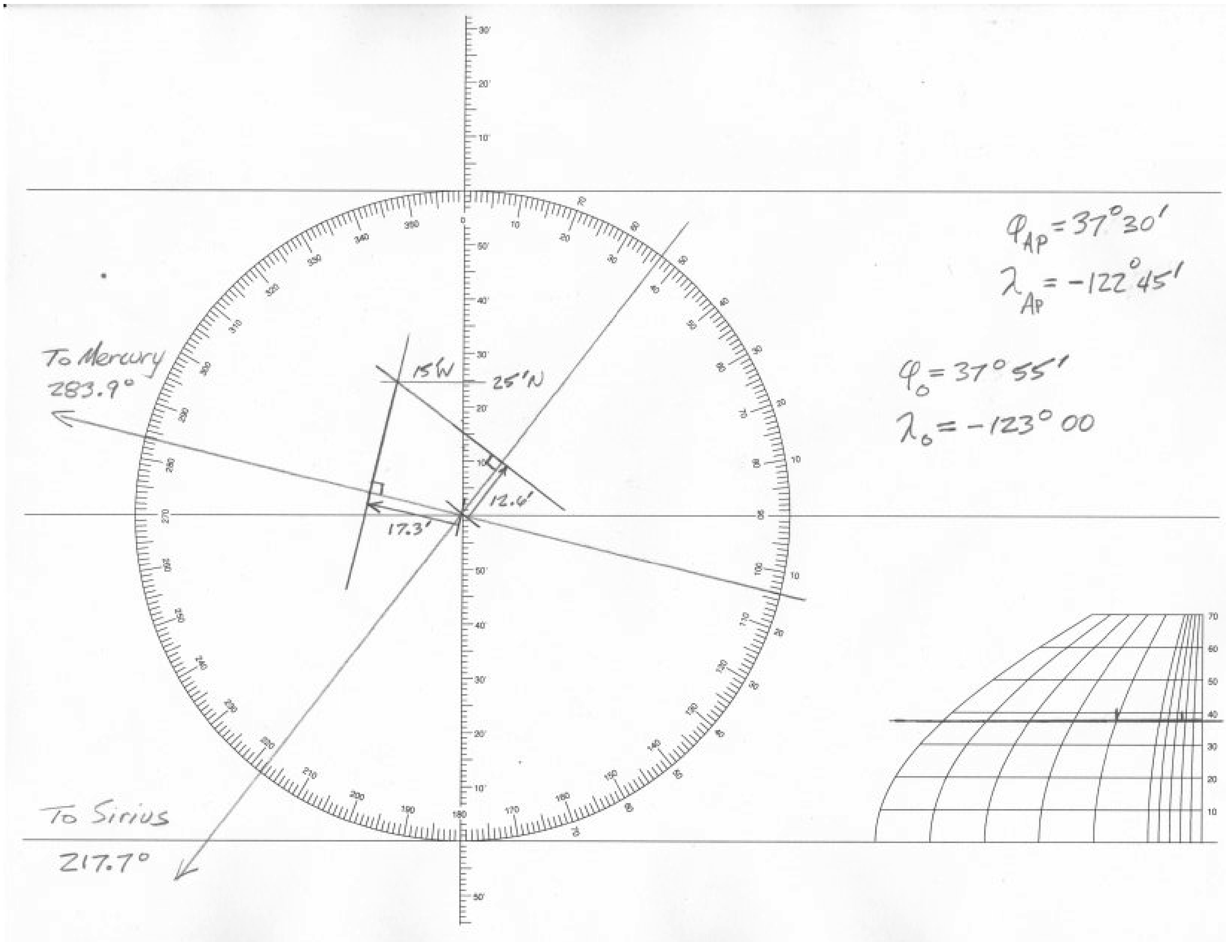
We are 12.6 nautical miles farther from Sirius (along Sirius' line of azimuth) than we thought we were.

```
In[∘]:= (hOMercury - hCMercury) * 60
Out[∘]= 17.3079
```

We are 17.3 nautical miles closer to Mercury (along Mercury's line of azimuth) than we thought we were.

Step 6 — Plot on Universal Plotting Sheet

In[∘]:=



From the plot, we see that we are 25’ north and 15’ west of our assumed position. So we adjust the assumed position and get the observed position:

In[∘]:=
```
ϕO = ϕAP + 25 / 60
```
Out[∘]=
```
37.9167
```

In[∘]:=
```
λO = λAP - 15 / 60
```
Out[∘]=
```
-123.
```

NB: the scale in the lower right of the universal plotting sheet had to be used to determine that we are 15’ west (because minutes of longitude get closer together as you get farther from the equator).

Ideally the two astronomical objects are at right angles to each other. The more the two azimuth lines are parallel to each other, the more that errors in the position of either line affect the intersection point.

We still have one last computation to do. We have to tell the helm what heading to steer.

Step 7 — Compute Distance and Direction to San Francisco Bay

Fortunately, this is a calculation we have gotten good at: given the coordinates of two points on the globe, compute the distance and direction to go from one to the other.

In[∘]:=
```
ϕSF = 37.8;
λSF = -122.5;
```

From Law of Cosines, the distance to San Francisco is:

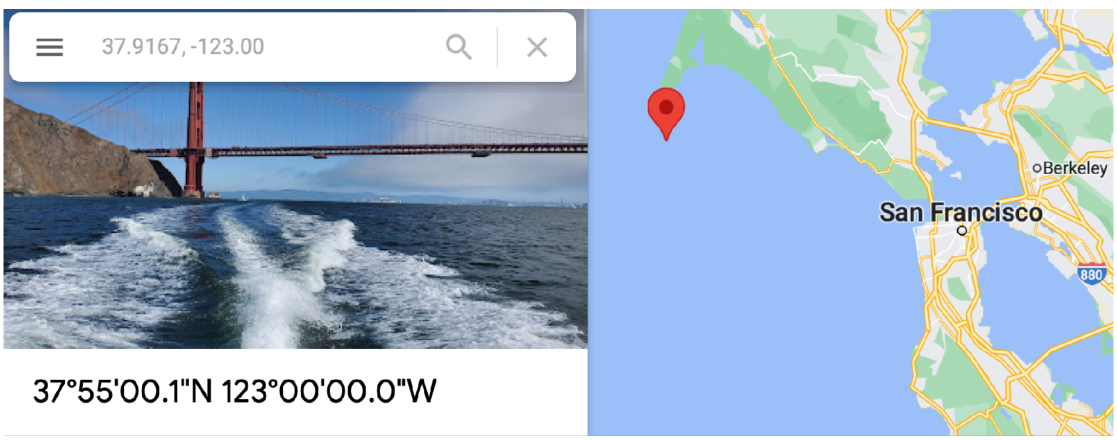
In[∘]:=
```
d = ArcCos[Sin[ϕ0 Degree] Sin[ϕSF Degree] +
      Cos[ϕ0 Degree] Cos[ϕSF Degree] Cos[λSF Degree - λAP Degree]] / Degree;
d * 60
```

Out[∘]=
```
13.757
```

We are about 14 nautical miles away. From Law of Sines, our heading should be:

In[∘]:=
```
heading = ArcSin[Sin[λSF Degree - λAP Degree] Cos[ϕSF Degree] / Sin[d Degree]] / Degree
```

Out[∘]=
```
59.4908
```



Acck! Looking at the chart, 59° is clearly not the right heading; San Francisco is southeast of us. Once again arcsin has failed us. It's the other value of arcsin that we want. Steer the ship on heading:

In[∘]:=
```
180 - heading
```

Out[∘]=
```
120.509
```