

Chapter 13. Some applications and implications of the cycloid finite motion model

© 1988 – 2012 by Vincent S. Cronin

13.1 Introduction

We learned, in the last chapter, how to find the cycloid trajectory of a point or three on a plate, observed from another plate during finite displacement. Now we are going to see how to model finite displacements along different types of plate boundary. After mastering this material, you should be able to model finite motion along any specific boundary between rigid (or almost rigid) plates. The reason for that lawyerly distinction will become apparent in later chapters, when we look at plate-boundary strain as indicated by GPS data.

Throughout this chapter, we will assume that the angular velocity vectors relative to the external reference frame are constant during the finite time interval used in each model. An equivalent way of expressing that assumption is to say that the geohedron of all relevant angular velocity vectors relevant to our modeling will be rigid. We can use that rigid geohedron as our reference frame in which to track the motions of plates, or we can model the motion of the geohedron relative to a given observer's plate and use it as a basis for tracking the motions of other plates.

We will also use model plate boundaries that are similar to existing plate boundaries, although our goal in this chapter is not to create detailed models of specific plate-boundary interactions. After completing this chapter, you might have a burning urge to model some actual plate boundaries. Peter Bird has digitized boundaries for 56 plates and has made those data available online via http://peterbird.name/publications/2003_PB2002/2003_PB2002.htm (Bird, 2003).

13.2 User-defined functions

We will use the following user-defined function developed in a previous chapter.

```
In[1]:= convert2Cart[lat_, long_] := {Cos[lat Degree] Cos[long Degree],  
    Cos[lat Degree] Sin[long Degree], Sin[lat Degree]};  
  
In[2]:= unitVect3D[vect_] := {(vect[[1]] / Norm[vect]),  
    (vect[[2]] / Norm[vect]), (vect[[3]] / Norm[vect])};
```

```

In[3]:= findGeogCoord[vect_] := Module[{lat, long, a, b, c, d, e, f},
  a = ArcSin[vect[[3]]]; b = {vect[[1]], vect[[2]], 0};
  c = If[(((Abs[vect[[1]]] < (1 × 10-14)) &&
    (Abs[vect[[2]]] < (1 × 10-14))), {1, 1, 0},
    {vect[[1]] / Norm[b], vect[[2]] / Norm[b], 0}];
  d = {1, 0, 0}; e = VectorAngle[c, d];
  f = If[(vect[[2]] < 0), (-e), (e)]; lat = a (180 / π);
  long = If[(((Abs[vect[[1]]] < (1 × 10-14)) && (Abs[vect[[2]]] <
    (1 × 10-14))), 0, (f (180 / π))]; {lat, long}];

In[4]:= makeGreatCircle[normal_] :=
  Module[{a, x1, y1, z1, x2, y2, z2, j1, j2, j3, θ, b},
  a = Table[{Cos[i Degree], Sin[i Degree], 0}, {i, 0, 360, 5}];
  x1 = {1, 0, 0}; y1 = {0, 1, 0}; z1 = {0, 0, 1}; z2 = normal;
  x2 = unitVect3D[Cross[z2, z1]]; y2 = unitVect3D[Cross[z2, x2]];
  j1 = {{x1.x2, y1.x2, z1.x2}, {x1.y2, y1.y2, z1.y2},
    {x1.z2, y1.z2, z1.z2}}; θ = VectorAngle[z1, z2];
  j2 = {{1, 0, 0}, {0, Cos[θ], Sin[θ]}, {0, -Sin[θ], Cos[θ]}};
  j3 = Inverse[j1];
  b = Table[j3.j2.j1.a[[i]], {i, 1, Length[a]}]; b];

In[5]:= cycloid[obsΩ_, movΩ_, refPt_, t_] :=
  Module[{a, b, c, d, e, f, g, h, i, j, k, l, m1, m2, m3, m4, m5,
    n, p, answer}, a = {1, 0, 0}; b = {0, 1, 0}; c = {0, 0, 1};
  d = unitVect3D[obsΩ]; e = Norm[obsΩ]; f = unitVect3D[movΩ];
  g = Norm[movΩ]; h = unitVect3D[d × f]; i = h; j = unitVect3D[h × d];
  k = unitVect3D[i × f]; m1 = {{k[[1]], k[[2]], k[[3]]},
    {i[[1]], i[[2]], i[[3]]}, {f[[1]], f[[2]], f[[3]]}};
  m2 = {{Cos[(-g Degree) t], Sin[(-g Degree) t], 0},
    {-Sin[(-g Degree) t], Cos[(-g Degree) t], 0}, {0, 0, 1}};
  l = VectorAngle[d, k]; If[l < (π / 2), (n = 1), (n = (-1))];
  p = (n) (VectorAngle[f, d]);
  m3 = {{Cos[p], 0, -Sin[p]}, {0, 1, 0}, {Sin[p], 0, Cos[p]}};
  m4 = {{Cos[(e Degree) t], Sin[(e Degree) t], 0},
    {-Sin[(e Degree) t], Cos[(e Degree) t], 0}, {0, 0, 1}};
  m5 = {{j[[1]], h[[1]], d[[1]]}, {j[[2]], h[[2]], d[[2]]},
    {j[[3]], h[[3]], d[[3]]}};
  answer = m5.m4.m3.m2.m1.refPt; answer];

In[6]:= circMotion[x_, angVelVect_, dT_] :=
  Module[{north, rotPole, w, xPole, yPole, m1, m2, m3, answer},
  north = {0, 0, 1}; rotPole = unitVect3D[angVelVect];
  w = Norm[angVelVect]; xPole = unitVect3D[rotPole × north];
  yPole = unitVect3D[rotPole × xPole];
  m1 = {{xPole[[1]], xPole[[2]], xPole[[3]]},
    {yPole[[1]], yPole[[2]], yPole[[3]]},
    {rotPole[[1]], rotPole[[2]], rotPole[[3]]}};
  m2 = {{Cos[(w dT) Degree], -Sin[(w dT) Degree], 0},
    {Sin[(w dT) Degree], Cos[(w dT) Degree], 0}, {0, 0, 1}};
  m3 = Inverse[m1]; answer = m3.m2.m1.x; answer];

```

13.3 Data from NNR-MORVEL56

As in chapter 12, we will use data from the no-net-rotation solution derived by Argus and others (2011) based on the MORVEL velocity model (DeMets and others, 2010) and Peter Bird's descriptions of 56 plates (Bird, 2003). The abbreviations in the first column are an=Antactica, co=Cocos, nz=Nazca, na=North America, nb=Nubia (western Africa), pa=Pacific, and sa=South America. For a given row/record, the second column contains the latitude of the poles around which the corresponding plate moves in a counter-clockwise manner, the third column has the corresponding polar longitude, and the fourth column has the angular speed in degrees per million years.

$$\text{In[7]:= nnr} = \begin{pmatrix} \text{co} & 26.93 & -124.31 & 1.198 \\ \text{nz} & 46.23 & -101.06 & 0.696 \\ \text{na} & -4.85 & -80.64 & 0.209 \\ \text{nb} & 47.68 & -68.44 & 0.292 \\ \text{pa} & -63.58 & 114.70 & 0.651 \\ \text{sa} & -22.62 & -112.83 & 0.109 \\ \text{an} & 65.42 & -118.11 & 0.250 \end{pmatrix};$$

We use these data to compute the angular velocity vector for each plate relative to the no-net-rotation external reference frame.

```
In[8]:= eΩco = nnr[[1, 4]] * convert2Cart[nnr[[1, 2]], nnr[[1, 3]]];
```

```
In[9]:= eΩnz = nnr[[2, 4]] * convert2Cart[nnr[[2, 2]], nnr[[2, 3]]];
```

```
In[10]:= eΩna = nnr[[3, 4]] * convert2Cart[nnr[[3, 2]], nnr[[3, 3]]];
```

```
In[11]:= eΩnb = nnr[[4, 4]] * convert2Cart[nnr[[4, 2]], nnr[[4, 3]]];
```

```
In[12]:= eΩpa = nnr[[5, 4]] * convert2Cart[nnr[[5, 2]], nnr[[5, 3]]];
```

```
In[13]:= eΩsa = nnr[[6, 4]] * convert2Cart[nnr[[6, 2]], nnr[[6, 3]]];
```

```
In[14]:= eΩan = nnr[[7, 4]] * convert2Cart[nnr[[7, 2]], nnr[[7, 3]]];
```

13.4 Modeling some typical boundaries

Spreading ridges and rifts

There are two general types of reference frame that we might select for our model: a reference frame external to the lithosphere as might be provided by NNR-MORVEL56, or a reference frame that is fixed to one of the plates. Both will work fine, and will give the same answer although expressed in different coordinate systems. Let's do both and compare. We will refer to the external reference frame as the polar reference frame, because it is effectively fixed to the set of rotational poles whose positions relative to each other are assumed to be constant in the first-order cycloid model.

External (polar) reference frame

We start with a couple of points approximately located along the Cocos-Pacific ridge axis near the Galapagos triple junction (Klein and others, 2005). We will use the simplest of rifts in our model: a segment of a great-circle arc between two points, labeled **northPt** (3.500°N latitude and 101.700°W longitude) and **southPt** (2.617°N latitude and 101.883°W longitude). We then group these two vectors into a list **in1** that describes the initial locations of these points at model time = 0.

```
In[15]:= northPt = convert2Cart[3.500, -101.700];
        southPt = convert2Cart[2.617, -101.883]; in1 = {northPt, southPt};
```

We set an arbitrary model time (**modT**) = 0.5 million years.

```
In[16]:= modT = 0.5;
```

Over the model time, each plate will rotate around its respective axis in accordance with the input angular velocity vector. For example, the Cocos plate will rotate around angular velocity vector ${}_E\Omega_{co}$, or if you prefer around pole ${}_E P_{co}$, in a positive (counter-clockwise) direction by an angle equal to (**modT** * ${}_E\omega_{co}$). We will utilize the user-defined function **circMotion** to do the computational work, moving the vectors in the list **in1** to their displaced positions.

```
In[17]:= outCoco1 = Table[circMotion[in1[[i]], eOmegaCo, modT],
        {i, 1, Length[in1]}]; outPacific1 =
        Table[circMotion[in1[[i]], eOmegaPa, modT], {i, 1, Length[in1]}];
```

Let's take a peek at the results. Each of the following lists contains two rows, each of which represents a unit location vector. List **in1** is the initial location of the model rift, **outCoco1** is that rift line as it would appear on the Cocos plate after 0.5 million years of displacement, and **outPacific1** is that rift line on the Pacific plate after 0.5 Myr.

```
In[18]:= MatrixForm[N[in1]]
```

Out[18]//MatrixForm=

$$\begin{pmatrix} -0.202409 & -0.977396 & 0.0610485 \\ -0.205699 & -0.97755 & 0.0456594 \end{pmatrix}$$

```
In[19]:= MatrixForm[N[outCoco1]]
```

Out[19]//MatrixForm=

$$\begin{pmatrix} -0.198264 & -0.978015 & 0.0646429 \\ -0.201434 & -0.978264 & 0.04923 \end{pmatrix}$$

```
In[20]:= MatrixForm[N[outPacific1]]
```

Out[20]//MatrixForm=

$$\begin{pmatrix} -0.207237 & -0.976289 & 0.0625507 \\ -0.210563 & -0.976442 & 0.0471693 \end{pmatrix}$$

Now we will construct a simple graphic to show us the results.

```
In[21]:= out1 = Graphics3D[{Green, Line[outCoco1],
        Blue, Line[outPacific1], Red, Dashed, Line[in1]},
        AspectRatio -> 1, ColorOutput -> Red, Boxed -> False];
```

```
In[22]:= Show[out1]
```

```
Out[22]=
```

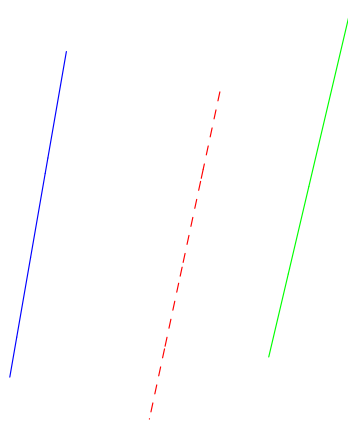


Figure 13-1. Simple model rift axes after 0.5 Myr of model displacement. The blue line is the displaced initial rift on the Pacific plate, the green line is the displaced rift line on the Cocos plate, and the position of the rift at model time = 0 is the red dashed line. All are shown in the polar reference frame.

There is something odd and unexpected about Figure 13-1. Why isn't the dashed red line that represents the initial location of the rift axis located exactly between the blue and the green lines? It is commonly assumed that rifting is a symmetrical process: that an equal amount of new material will be added to the trailing edge of both plates. The dashed red line is not only closer to the green line than the blue; it is lower than either the green or blue lines rather than being, well, half way between the two. The reason for the weirdness is that the dashed line is the position of the rift at **modT** = 0, but not at **modT** = 0.5. Let's find where the ridge axis will be at **modT** = 0.5, assuming symmetrical spreading.

```
In[23]:= newRidge1 = (outCoc1 + outPac1) / 2;
```

```
In[24]:= out2 = Graphics3D[{Red, Line[newRidge1]},  
AspectRatio -> 1, Boxed -> False];
```

```
In[25]:= Show[out1, out2]
```

```
Out[25]=
```

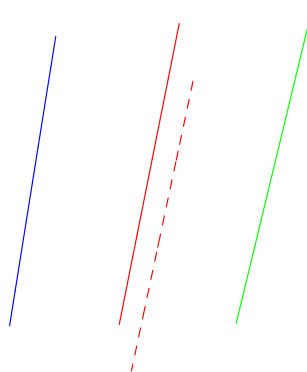


Figure 13-2. Simple model rift axes as in Figure 13-1, but with the rift axis at model time 0.5 Myr added (solid red line). Shown in the polar reference frame.

The ridge axis has moved from the red dashed line, its initial position, to the solid red line after 0.5 million

years of symmetrical spreading. It has moved perpendicular to its trend because the tangential velocity of the Pacific plate along the ridge is greater than that of the Cocos plate, and so the ridge migrates in the direction the Pacific plate is moving. There is also motion parallel to the ridge axis, illustrated by the fact that the solid red line is higher than the dashed red line.

Plate-based reference frame

We will now run the same experiment, but our reference frame will be fixed to the Pacific plate. We employ the user-defined function **cycloid** to do the computational work of defining the new position of the initial rift axis as it appears on the Cocos plate after 0.5 million years of displacement.

```
In[26]:= outCoco2 = Table[
           cycloid[eΩpa, eΩco, in1[[i]], modT], {i, 1, Length[in1]};
In[27]:= initialRidge = Table[
           circMotion[in1[[i]], -eΩpa, modT], {i, 1, Length[in1]};
In[28]:= out3 = Graphics3D[{Green, Line[outCoco2], Blue, Line[in1], Red,
           Dashed, Line[initialRidge]}, AspectRatio → 1, Boxed → False];
In[29]:= Show[out3]
```

Out[29]=

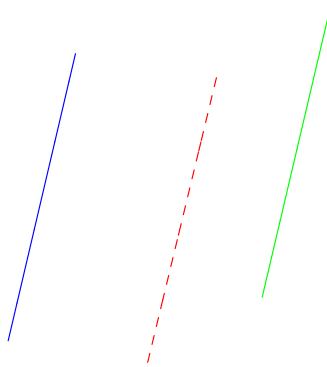


Figure 13-3. Simple model rift axes after 0.5 Myr of model displacement. The blue line is the displaced initial rift on the Pacific plate, the green line is the displaced rift line on the Cocos plate, and the position of the rift at model time = 0 is the red dashed line. All are shown relative to the Pacific plate.

Figure 13-3 looks a lot like figure 13-1. Now let's add the new ridge axis.

```
In[30]:= newRidge2 = (outCoco2 + in1) / 2;
In[31]:= out4 = Graphics3D[{Red, Line[newRidge2]},
           AspectRatio → 1, Boxed → False];
```

```
In[32]:= Show[out3, out4]
```

```
Out[32]=
```

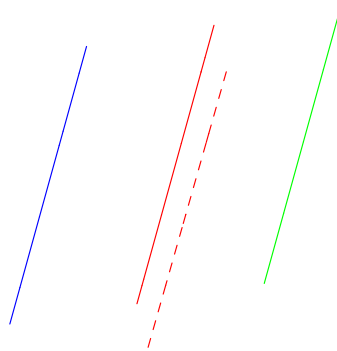


Figure 13-4. Simple model rift axes as in Figure 13-3, but with the rift axis at model time 0.5 Myr added (solid red line). Shown in a reference frame fixed to the Pacific plate.

It can be shown that the results shown in Figures 13-1 and 13-2 are identical to those shown in Figures 13-3 and 13-4, except for the change in reference frame. We will depict all of the boundaries in the polar coordinate system throughout the rest of this chapter.

Subduction

Subduction is the tectonic equivalent of sweeping something under the rug or, more precisely, of moving the rug over the floor (or the floor under the rug). We will model a small part of the Nazca plate as it subducts under the South American plate near the Chilean states of Atacama and Coquimbo over the next 1 million years.

```
In[33]:= modT = 1;
```

To help us see the motion of the subducting plate, we choose to place a triangle on the model Nazca plate. The coordinates of the triangle at initial model time 0 are contained in the list **triangle0**, and the coordinates at model time 1 Myr is called **triangle1**.

```
In[34]:= tri1 = convert2Cart[-27, -73]; tri2 = convert2Cart[-27.5, -73.25];
tri3 = convert2Cart[-27.5, -72.75];
```

```
In[35]:= triangle0 = {tri1, tri2, tri3, tri1};
```

```
In[36]:= triangle1 = Table[circMotion[triangle0[[i]], eΩnz, modT],
{i, 1, Length[triangle0]}];
```

The leading edge of the over-riding plate is represented by a straight line (small-circle arc) between two points along the boundary, and the list containing those coordinates is called **trench0**. The position of the leading edge at model time 1 Myr is called **trench1**.

```
In[37]:= northPt = convert2Cart[-26.511, -71.670];
southPt = convert2Cart[-29.932, -72.493];
trench0 = {northPt, southPt};
```

```
In[38]:= trench1 = Table[circMotion[trench0[[i]], eΩsa, modT],  

{i, 1, Length[trench0]}];
```

The graphic is constructed in the polar reference frame. The triangle on the Nazca plate is red, and the leading edge of the South American plate is blue.

```
In[39]:= out5 = Graphics3D[{Red, Line[triangle1], Blue, Line[trench1],  

Red, Dashed, Line[triangle0], Blue, Dashed, Line[trench0]},  

AspectRatio -> 1, Boxed -> False];
```

```
In[40]:= Show[out5]
```

```
Out[40]=
```

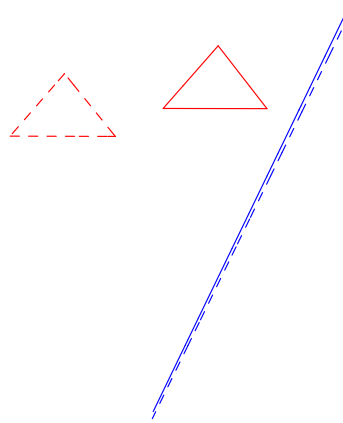


Figure 13-5. Initial position of an arbitrary triangle on the Nazca plate (dashed red triangle) and the leading edge of the South American plate (dashed blue line), and subsequent positions (solid lines) after 1 Myr of displacement viewed in the polar reference frame. The base of the dashed triangle is along latitude 27.5 S, and is approximately 50 km long.

The triangle is rotated around the Nazca pole, and the trench is rotated around the South American pole. If an observer on the South American plate tracked one of the vertices of the triangle on the Nazca plate, it would move along a cycloidal trajectory during the model displacement. The oblique motion of the Nazca plate relative to the trench results in a right-lateral component of shear parallel to the trench. Strain is partitioned along the leading edge of the South American plate, leading to right-lateral displacement along the Atacama fault, which is located east-southeast of the trench oriented approximately parallel to the trench.

Transform faults

Transform faults are strike-slip plate-boundary faults. In first-generation models of plate kinematics, the trace of a transform fault between any two plates A and B was thought to have the shape of a circle that is concentric with the pole of relative motion between the two plates, ${}_A P_B$. In that early view of kinematics, ${}_A P_B$ is fixed to both plates, and trajectories of points along the fault on one plate are all concentric circles, as viewed from the other plate. Consequently, transform faults were thought to be surfaces along which there is neither convergence nor divergence, but only strike slip.

The axis of a mid-ocean ridge or continental rift zone is not necessarily perpendicular to the direction of spreading. The trench axis of a subduction zone is not necessarily perpendicular to the direction of conver-

gence. But the trace of a transform fault will be very close to parallel to the direction of relative motion across the fault. Consequently, transform faults are very important kinematic indicators.

The Kane transform fault is part of the boundary between the North American and Nubian (African) plates. Map images of the Kane fault and other features of the ocean basins are accessible via GeoMapAp (<http://www.geomapap.org>) and Virtual Ocean (<http://www.virtualocean.org>) -- two free applications developed by folks at the Lamont-Doherty Earth Observatory of Columbia University under the Marine Geoscience Data System.

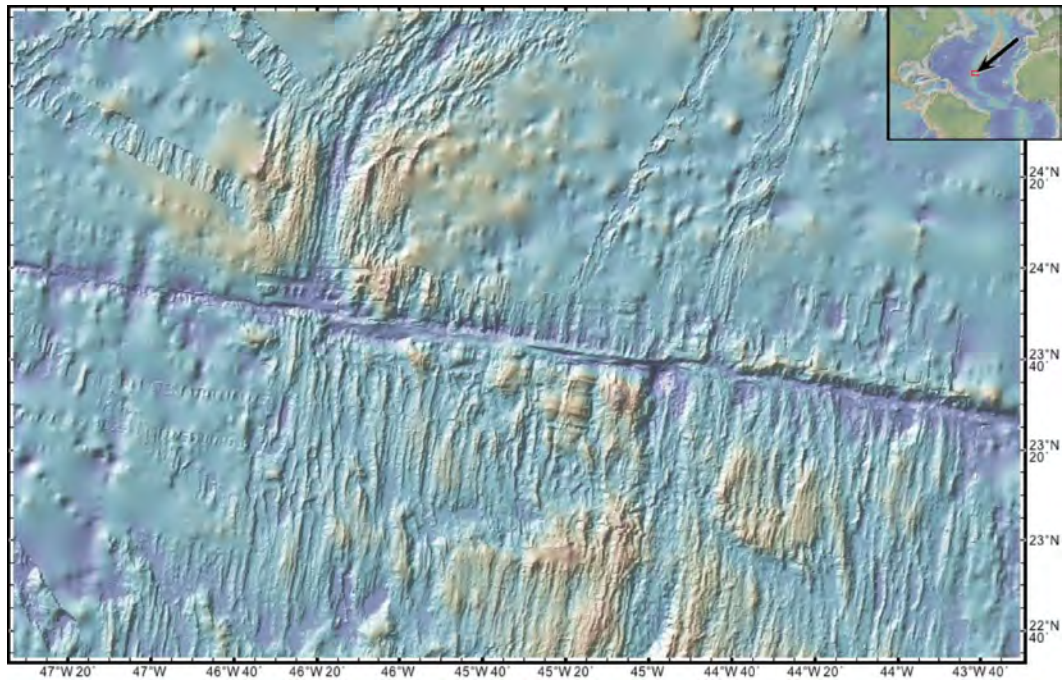


Figure 13-6. Bathymetric hillshade map in the area of the Kane fault (between longitudes $\sim 45^\circ\text{W}$ and $\sim 46^\circ 20'\text{W}$) in the North Atlantic Ocean Basin (see inset map) derived from multibeam seismic surveys and satellite data. The North American plate is to the west (left) and the Nubian plate is to the east (right). From GeoMapAp.

We begin with the approximate location of the two ends of the Kane fault (Tucholke and Schouten, 1988; Vogt, 1986; Schouten, Cande and Klitgord, 1985), and then create a synthetic Kane fault that is the same size as the observed Kane fault, has the supposed optimal shape of a small circle around the ${}_{na}P_{nb}$ pole, and whose west end is the same as the west end of the observed Kane fault (**wKaneObs**).

```
In[41]:= wKaneObs = convert2Cart [23.850, -46.362] ;  
eKaneObs = convert2Cart [23.640, -44.908] ;
```

We use the angular velocity vectors of the two plates relative to the external frame of reference (**e Ω nb** and **e Ω na**) to compute the angular velocity vector of the Nubian plate relative to the North American plate (**na Ω nb**).

```
In[42]:= na $\Omega$ nb = e $\Omega$ nb - e $\Omega$ na ;
```

The length of the angular velocity vector is the angular speed (**na ω nb**).

```
In[43]:= na $\omega$ nb = Norm [na $\Omega$ nb] ;
```

```
In[44]:= a = VectorAngle[Cross[wKaneObs, naΩnb],
      Cross[eKaneObs, naΩnb]] / Degree;
```

```
In[45]:= fullT = a / naωnb;
```

Find the location of eleven points along a small circle through the observed western end of the Kane fault (**modKane**) that is concentric with the angular velocity vector **naΩnb**.

```
In[46]:= modKane = Table[
      circMotion[wKaneObs, naΩnb, i], {i, 0, fullT, 0.1 * fullT}];
```

Set a model time (**modT**) that is a significant percentage (say, 85%) of the time it would take for displacement of one fault length along the Kane fault (**fullT**).

```
In[47]:= modT = 0.85 * fullT;
```

Rotate the North American side of the synthetic Kane fault for (**modT**) million years around its axis (**eΩna**) to its new position (**outNAmerica4**). In a similar manner, rotate the Nubian side of the Kane fault around its axis (**eΩnb**) to its new position (**outNubia4**).

```
In[48]:= outNAmerica4 = Table[circMotion[modKane[[i]], eΩna, modT],
      {i, 1, Length[modKane]}];
      outNubia4 = Table[circMotion[modKane[[i]], eΩnb, modT],
      {i, 1, Length[modKane]}];
```

Prepare and show the output, using different colors and line styles to differentiate.

```
In[49]:= out6 = Graphics3D[
      {Green, Line[outNubia4], Blue, Line[outNAmerica4], Black,
      Dashed, Line[modKane]}, AspectRatio -> 1, Boxed -> False];
```

```
In[50]:= Show[out6]
```

Out[50]=

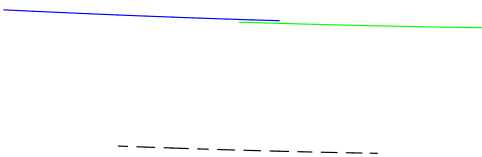


Figure 13-7. North American (blue) and Nubian (green) sides of the synthetic Kane fault after displacement of 85% of the fault's total length, as observed in the polar reference frame. The dashed line is the initial position of the synthetic Kane fault before displacement.

Two features of the results are immediately apparent. First, the Kane fault has moves in the frame of reference that is fixed to the poles. It moves radially (towards or away from) the pole of instantaneous relative motion between the North American and Nubian plate. Even though we created the synthetic Kane

fault with the supposed optimal shape of a circle concentric with the North American-Nubian pole, finite motion of that fault radial to the pole would require a change in shape of the fault if there was to be no convergence or divergence along the fault. Of course, the change in shape would be accomplished through plate-boundary strain involving oblique convergence or divergence (that is, transpression or transtension), so the model fault cannot be a pure strike-slip fault.

The second obvious feature of the model is that the blue and green curves are divergent after displacement of 85% of the total length of the synthetic Kane fault. This indicates that there must be some component of convergence or divergence along the fault during finite displacement.

Ridge-transform-ridge system

Let us now combine our modeling of ridge and transform boundaries to examine how the synthetic Kane fault might evolve during future displacements. We will find one point on the ridge that extends north of the west end of the Kane fault (**nRidge**) and connect it to the west end of the Kane using the built-in *Mathematica* function **Prepend**, to represent one ridge

```
In[51]:= nRidge = convert2Cart [24.459, -46.239];
```

```
In[52]:= temp = Prepend[modKane, nRidge];
```

and find a point on the ridge that extends south of the east end of the Kane fault (**sRidge**) and connect it to the east end of the Kane using the built-in *Mathematica* function **Append**, to represent the other ridge.

```
In[53]:= sRidge = N[convert2Cart [22.768, -45.022] ];
```

```
In[54]:= rTRmodel = Append[temp, sRidge];
```

Then we will displace the two plates by 85% of the length of the synthetic Kane fault, as before.

```
In[55]:= outNAmerica5 = Table[circMotion[rTRmodel[[i]], eΩna, modT],
  {i, 1, Length[rTRmodel]};
  outNubia5 = Table[circMotion[rTRmodel[[i]], eΩnb, modT],
  {i, 1, Length[rTRmodel]};
```

We find the new locations of the north and south ridge axes (**northRidge** and **southRidge**).

```
In[56]:= northRidge = {((outNAmerica5[[1]] + outNubia5[[1]]) / 2),
  ((outNAmerica5[[2]] + outNubia5[[2]]) / 2)};
```

```
In[57]:= southRidge = {((outNAmerica5[[Length[rTRmodel] - 1]] +
  outNubia5[[Length[rTRmodel] - 1]]) / 2),
  ((outNAmerica5[[Length[rTRmodel]]] +
  outNubia5[[Length[rTRmodel]]]) / 2)};
```

Finally, we prepare and show the output.

```
In[58]:= out7 =
  Graphics3D[{Green, Line[outNubia5], Blue, Line[outNAmerica5],
  Red, Line[northRidge], Red, Line[southRidge], Black,
  Dashed, Line[rTRmodel]}, AspectRatio → 1, Boxed → False];
```

In[59]:= Show[out7]

Out[59]=

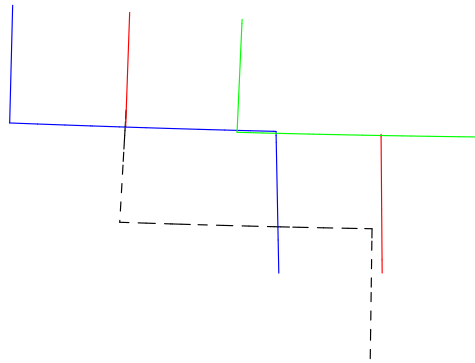


Figure 13-8. North American (blue) and Nubian (green) sides of the synthetic Kane fault and adjoining ridges after displacement of 85% of the fault's total length, as observed in the polar reference frame. The red lines are the locations of the ridge axes at the end of the modeled displacement. The dashed line is the initial position of the ridge-transform-ridge system before displacement.

Now we can more clearly see that the model Kane fault undergoes transpression during displacement, because the North American plate overlaps the Nubian plate in Figure 13-8. We might infer that the fault shape would have to evolve toward a circle with a smaller radius, because it is moving closer to the pole of North American-Nubian relative motion. We also observe that the modeled ridge axes each have a component of motion perpendicular to the axis, and that the northern ridge moved west and the southern ridge moved east relative to their respective initial positions, as viewed in the polar reference frame.

13.5 Modeling a triple junction

The triple junctions imagined in first-generation plate tectonics were simple geometric forms. Many, most or all actual triple junctions are interestingly complex. We will look at real triple junctions in a future chapter. Here, we simply indicate how we can model a couple of triple junctions.

Ridge-ridge-ridge triple junction

The first model triple junction is a simplified version of the Galapagos triple junction in which the tiny Galapagos plate is omitted. The model triple junction is between the Pacific, Cocos and Nazca plate.



Figure 13-9. Bathymetric hillshade map in the area of the Galapagos triple junction in the east-central Pacific Ocean Basin (see inset map) derived from multibeam seismic surveys and satellite data. The Pacific plate is to the west (left), the Cocos plate is to the northeast (upper right) and the Nazca plate is to the southeast (lower right). From GeoMapAp.

The first thing we do is to define a 2-segment polyline on each of the three plates: each segment lies along a ridge axis, and the segments are joined at the triple junction.

```
In[60]:= inCocos6 = {convert2Cart [4.303, -102.305],
                convert2Cart [2.304, -102.094], convert2Cart [2.109, -100.354]};
inPacific6 = {convert2Cart [4.303, -102.305], convert2Cart [
                2.304, -102.094], convert2Cart [-0.070, -102.305]};
inNazca6 = {convert2Cart [-0.070, -102.305],
                convert2Cart [2.304, -102.094], convert2Cart [2.109, -100.354]};
```

We create a list of location vectors along the triple junction prior to displacement (**initTJ6**).

```
In[61]:= initTJ6 =
  {convert2Cart[4.303, -102.305], convert2Cart[2.304, -102.094],
   convert2Cart[2.109, -100.354], convert2Cart[2.304, -102.094],
   convert2Cart[-0.070, -102.305]};
```

We choose to examine the triple junction after a model time (**modT**) of 3 million years of displacement.

```
In[62]:= modT = 3;
```

Rotate the Cocos boundary polyline for (**modT**) million years around its axis (**eΩco**) to its new position (**outCocos6**). In a similar manner, rotate the Pacific boundary polyline around its axis (**eΩpa**) to its new position (**outPacific6**), and rotate the Nazca boundary polyline around its axis (**eΩnz**) to its new position (**outNazca6**).

```
In[63]:= outCocos6 = Table[circMotion[inCocos6[[i]], eΩco, modT],
  {i, 1, Length[inCocos6]}];
outPacific6 = Table[circMotion[inPacific6[[i]], eΩpa, modT],
  {i, 1, Length[inPacific6]}];
outNazca6 = Table[circMotion[inNazca6[[i]], eΩnz, modT],
  {i, 1, Length[inNazca6]}];
```

Constructing the new ridge-ridge-ridge triple junction takes some work, so we created a function called **tjCorner** to help. The problem that we confront with a RRR triple junction that we have not confronted before is that the ridge segments must propagate or grow during displacement. There are a couple of ways of handling this issue; here, we will maintain the ridge axes as great-circle arcs and create new 2-segment polylines to represent the new boundaries of each of the three plates.

The input parameters for the function **tjCorner** include...

```
In[64]:= tjCorner[seglenda_, seglendb_, seglmida_, seglmidb_,
  seg2enda_, seg2endb_, seg2mida_, seg2midb_, tj0_] :=
Module[{a, b, c, d, e, f, g, h, i, answer},
  a = unitVect3D[(seglenda + seglendb) / 2];
  b = unitVect3D[(seglmida + seglmidb) / 2];
  c = unitVect3D[(seg2mida + seg2midb) / 2];
  d = unitVect3D[(seg2enda + seg2endb) / 2]; e = a × b;
  f = c × d; g = unitVect3D[e × f]; h = VectorAngle[g, tj0];
  i = If[(h < (π / 2)), g, -g]; answer = {a, i, d}; answer];
```

We use the function **tjCorner** to define the new position of the triple junction boundary for each plate (**outTJCocos**, **outTJNazca** and **outTJPacific**).

```
In[65]:= outTJCocos =
  tjCorner[outPacific6[[1]], outCocos6[[1]], outPacific6[[2]],
  outCocos6[[2]], outCocos6[[3]], outNazca6[[3]],
  outCocos6[[2]], outNazca6[[2]], outCocos6[[2]]];
outTJNazca = tjCorner[outPacific6[[3]],
  outNazca6[[1]], outPacific6[[2]], outNazca6[[2]],
  outCocos6[[3]], outNazca6[[3]], outCocos6[[2]],
  outNazca6[[2]], outCocos6[[2]]]; outTJPacific =
  tjCorner[outPacific6[[1]], outCocos6[[1]], outPacific6[[2]],
  outCocos6[[2]], outPacific6[[3]], outNazca6[[1]],
  outPacific6[[2]], outNazca6[[2]], outCocos6[[2]]];
```

We display the mid-point of each of the three polylines as a green dot

```
In[66]:= markers = {outTJCocos[[2]], outTJNazca[[2]], outTJPacific[[2]]};
```

```
In[67]:= out9 = ListPointPlot3D[markers, AspectRatio -> 1, Boxed -> False,
    PlotStyle -> Directive[Green, PointSize[Medium]]];
```

and prepare the various polylines for graphic display.

```
In[68]:= out8 =
    Graphics3D[{Brown, Line[outCocos6], Green, Line[outPacific6],
    Blue, Line[outNazca6], Red, Line[outTJCocos],
    Line[outTJNazca], Line[outTJPacific], Black, Dashed,
    Line[initTJ6]}, AspectRatio -> 1, Boxed -> False];
```

```
In[69]:= Show[out8, out9]
```

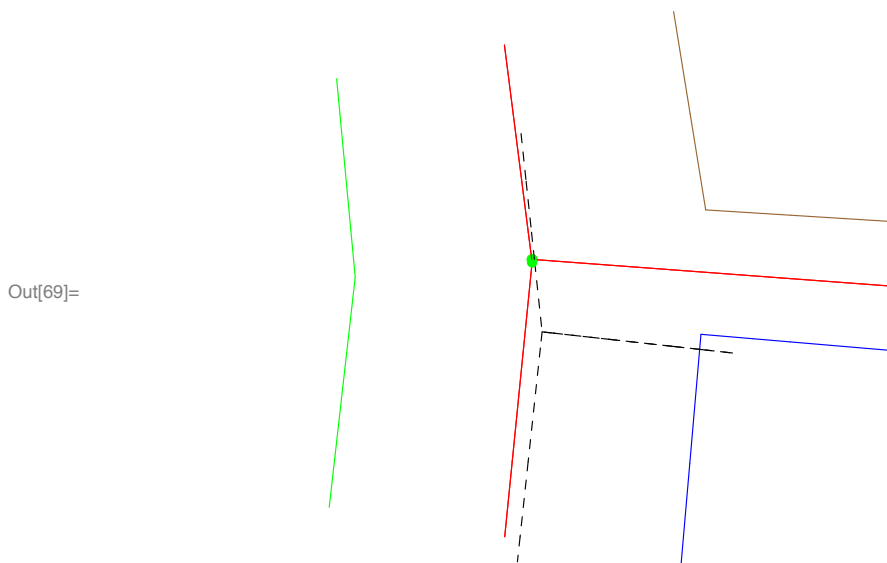


Figure 13-10. Synthetic ridge-ridge-ridge triple junction between the Pacific (green), Cocos (brown) and Nazca (blue) plates after model displacement of 3 million years. The initial position of the triple junction is indicated by the dashed lines. Green dots indicate the apices of the 2-segment polylines that represent the final position of the boundary for each plate. Depicted in the polar reference frame.

The triple junction migrates from its initial position (dashed lines in Figure 13-10) to a different position (red lines) during 3 million years of modeled displacement, as observed from the polar reference frame. Perhaps that was to be expected, given our previous results. The interesting and new feature we see in the output is that the mid-points of the three polylines that represent the new boundaries of each of the three plates are not coincident. The three green dots in Figure 13-10 are not in the same spot. From this we can infer that there is a boundary mismatch that develops at the triple junction under the conditions imposed by our model. Let's compute the distance, in kilometers, between these three points.

```
In[70]:= VectorAngle[outTJCocos[[2]], outTJNazca[[2]]] * 6371
```

```
Out[70]= 0.694379
```

```
In[71]:= VectorAngle[outTJCocos[[2]], outTJPacific[[2]]] * 6371
```

```
Out[71]= 3.37897
```

```
In[72]:= VectorAngle[outTJNazca[[2]], outTJPacific[[2]]] * 6371
```

```
Out[72]= 3.31445
```

Perhaps this mismatch is part of an explanation for the observed development of a tiny plate in the middle of this triple junction: the Galapagos plate.

Ridge-trench-trench triple junction

We showed in chapter 11 that there are 25 different types of triple junction. Let's look at the interesting triple junction between the Nazca, Antarctic and South American plates located just offshore from the Taitao Peninsula in Chile.

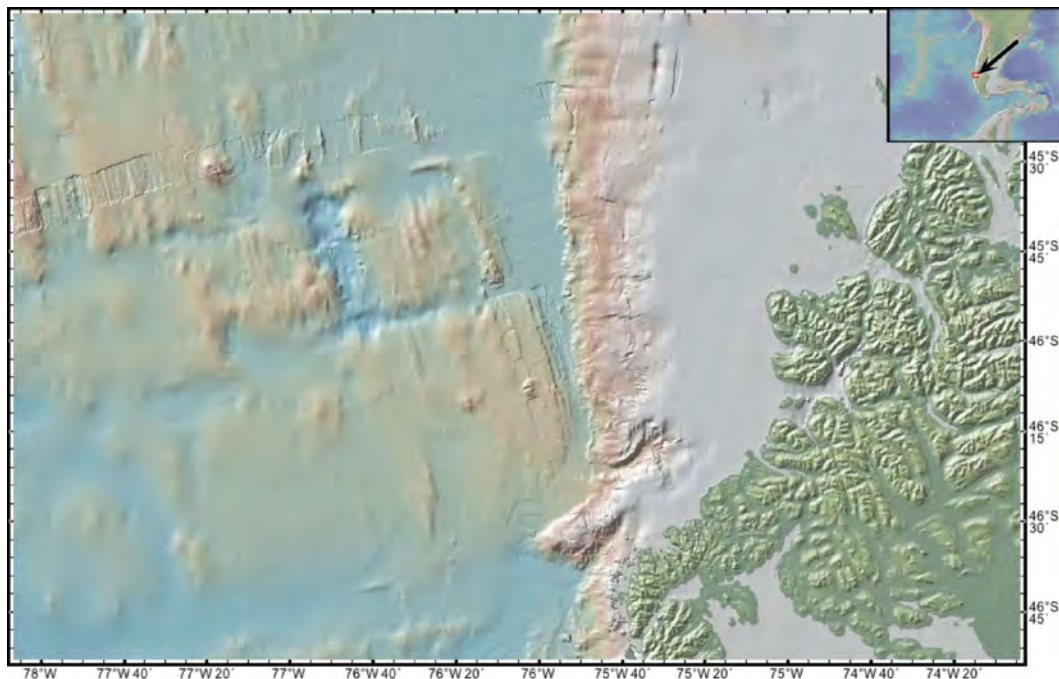


Figure 13-11. Bathymetric hillshade map in the area of the Nazca-Antarctica-South America triple junction in the southeast Pacific Ocean Basin (see inset map) derived from multibeam seismic surveys and satellite data. The Antarctic plate is to the southwest (lower left), the Nazca plate is to the northwest (upper left) and the South American plate is to the east (right), with part of the Taitao Peninsula shown in green. The trench is located between longitudes $\sim 75^{\circ}50'W$ and $75^{\circ}55'W$. From GeoMapAp.

The South American boundary is modeled as a simple trench represented by a polyline that joins three points (**inSAmerica7**). The Antarctic and Nazca plate boundaries are along the Antarctic Ridge. We choose to include two ridge axes and two transform faults in our model of that ridge, so both the Antarctic and Nazca boundaries are represented by a polyline that includes 5 points (**inAnta7** and **inNazca7**).


```

In[73]:= inSAmerica7 = {convert2Cart[-44.603, -75.779],
  convert2Cart[-46.317, -75.832],
  convert2Cart[-47.196, -76.078]};
inAnta7 = {convert2Cart[-45.798, -77.845],
  convert2Cart[-45.614, -76.904],
  convert2Cart[-45.969, -76.755], convert2Cart[
  -45.835, -76.025], convert2Cart[-46.317, -75.832]};
inNazca7 = {convert2Cart[-45.798, -77.845],
  convert2Cart[-45.614, -76.904],
  convert2Cart[-45.969, -76.755], convert2Cart[
  -45.835, -76.025], convert2Cart[-46.317, -75.832]};

```

We create a list of location vectors along the triple junction prior to displacement (**initTJ7a** and **initTJ7b**).

```

In[74]:= initTJ7a = {convert2Cart[-45.798, -77.845],
  convert2Cart[-45.614, -76.904], convert2Cart[-45.969,
  -76.755], convert2Cart[-45.835, -76.025], convert2Cart[
  -46.317, -75.832], convert2Cart[-44.603, -75.779]};

In[75]:= initTJ7b = {convert2Cart[-46.317, -75.832],
  convert2Cart[-47.196, -76.078]};

```

We choose to use a model time of 1 million years in the future.

```
In[76]:= modT = 1;
```

We then rotate the South American boundary polyline for (**modT**) million years around its axis (**eΩsa**) to its new position (**outSAmerica7**). In a similar manner, we rotate the Nazca boundary polyline around its axis (**eΩnz**) to its new position (**outNazca7**), and rotate the Antarctic boundary polyline around its axis (**eΩan**) to its new position (**outAnta7**).

```

In[77]:= outSAmerica7 = Table[circMotion[inSAmerica7[[i]], eΩsa, modT],
  {i, 1, Length[inSAmerica7]};
outNazca7 = Table[circMotion[inNazca7[[i]], eΩnz, modT],
  {i, 1, Length[inNazca7]};
outAnta7 = Table[circMotion[inAnta7[[i]], eΩan, modT],
  {i, 1, Length[inAnta7]};

```

We estimate the position of the ridge axes at **modT**.

```

In[78]:= northRidge = {((outAnta7[[2]] + outNazca7[[2]]) / 2),
  ((outAnta7[[3]] + outNazca7[[3]]) / 2)};
southRidge = {((outAnta7[[4]] + outNazca7[[4]]) / 2),
  ((outAnta7[[5]] + outNazca7[[5]]) / 2)};

In[79]:= out10 =
  Graphics3D[{Brown, Line[outAnta7], Green, Line[outSAmerica7],
  Blue, Line[outNazca7], Red, Line[northRidge],
  Line[southRidge], Black, Dashed, Line[initTJ7a],
  Line[initTJ7b]}, AspectRatio → 1, Boxed → False];

```

In[80]:= Show[out10]

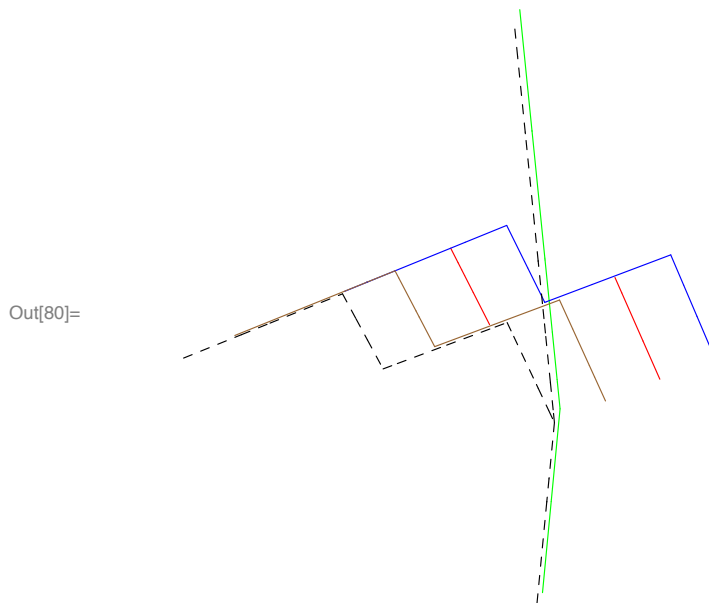


Figure 13-12. Evolution of the triple junction between the Antarctic (brown polyline), Nazca (blue) and South American (green) plates over a model time of 1 million years, as seen in the polar reference frame. Red lines are the ridge axes at 1 Myr, and the dashed lines mark the initial position of the triple junction.

At the starting time, today, the triple junction is a ridge-trench-trench triple junction. Over the next million years as the ridge is subducted, it will become a transform-trench-trench triple junction, and will migrate north along the trench. The transform fault that is closest to the trench appears to be in transtension, while the other transform fault appears to be in transpression. In just one million years, the ridge axis that we see at the trench today will have subducted, placing hot asthenosphere within a few kilometers of the base of the South American crust immediately southeast of the triple junction. Right-lateral displacement along the transform fault at the triple junction is likely to induce some shearing in the over-riding South American plate, and as subduction progresses the topographically higher southwest end of the transform fault will begin to scrape or bulldoze the underside of the South American plate. So the tectonic forecast for the Taitao Peninsula calls for possible increases in igneous activity due to the subducting ridge axis, along with right-lateral shear and differential uplift related to the subducting transform fault.

13.6 Some implications of cycloidal relative motion

Preface

There is an 800 pound gorilla in the room. At the time of this writing, in early 2012 and a half-century after the initial papers relevant to plate kinematics were published, many geoscientists assume that finite motion is circular, around fixed Euler poles. The model presented thus far is not a circular-motion model.

I accept that I am now going to construct a “straw man” called the circular finite motion model (CFMM). And we all know what ultimately happens to straw men. According to proponents of the CFMM, the finite trajectory of any point on one plate as observed from an adjacent plate over a finite time interval will be a circle that is concentric with the angular velocity vector of that two-plate system. The angular velocity vector does not change its position or magnitude relative to either of the two plates throughout that finite time interval. As a consequence, (1) transform faults will be purely strike-slip faults with no convergence or divergence along them, (2) transform faults and oceanic fracture zones will have the shape of small circles concentric with the angular velocity vector of the two-plate system, (3) the shape of transform faults and oceanic fracture zones will follow flow lines of one plate relative to the other, (4) transform faults will remain the same angular distance from the poles of relative motion between the relevant two plates, (5) the spreading segments of mid-ocean ridges and rifts exhibit finite motion around an angular velocity vector that is coincident with the angular velocity vector of the two plates, but has half the magnitude, and (6) the tangential velocity of any point on one plate as observed from the other plate will be constant in magnitude and remain tangent to the same circular trajectory throughout the finite time interval. Deviations from these predicted behaviors are assumed to be due to some change in the steady-state conditions -- changes in plate motions caused by some geodynamic change.

The problem is that the CFMM does not work in any 3+ plate world in which the angular velocity vectors for the various plate pairs are not colinear with each other. The CFMM does not work on today's Earth, judging from our best estimates of the instantaneous relative motion of plates. So comparing the CFMM with the cycloidal finite motion model or any more complex model that is kinematically admissible is a little bit like comparing the finite motion of a bronze statue of the great racehorse Seabiscuit versus that of a lazy basset hound. It makes sense to talk about the finite (but limited) motion of the basset hound, but it makes no sense to talk about the finite motion of the bronze Seabiscuit. There is no point in comparing the results of the cycloid model with the supposed results of the CFMM, because the CFMM is not valid for continuous finite motion given the observed instantaneous motions between today's plates on Earth.

I have now knocked down the straw man, and the 800-pound gorilla turned out to be a stuffed momento of a county fair from long ago.

Implications

There are several consequences of cycloidal relative motion of plates. For the moment, we will consider the consequences of the first-order cycloid model, in which the angular velocity vectors are constant throughout the finite time interval being modeled. In nerdspeak, the geohedron of angular velocity vectors of plates moving relative to other plates and relative to an external reference frame is assumed rigid or constant during the model time interval. This is the simplest cycloid model. Given the types of constraining observational data that we will begin to consider in a future chapter, higher-order models can be constructed in which the angular velocity vectors vary with time.

Some of the consequences of simple cycloidal motion between plates include the following.

- (1) Transform faults cannot be purely strike-slip faults with no convergence or divergence along them.
- (2) Transform faults and oceanic fracture zones do not have the shape of small circles around the corresponding pole of relative motion.
- (3) The shape of transform faults and oceanic fracture zones do not follow flow lines of one plate relative to the other. An oceanic fracture zone on a given plate traces the flow line of the corresponding ridge-fault intersection.
- (4) Transform faults do not maintain a constant angular distance from the pole of relative motion between the relevant two plates.
- (5) The spreading segments of mid-ocean ridges and rifts are not constrained to exhibit finite motion around the axis of relative motion between the two plates, but has half the angular speed between the two plates.
- (6) The tangential velocity of any point on one plate as observed from the other plate varies in direction and magnitude over time, but is tangent to a small circle around the relative pole at any instant. The effect of this change in speed and direction of relative motion across a plate boundary over time has been called cycloid wobble.

There are a few other consequences that we are not yet in a position to appreciate, based on our work thus far. But there is more fun to come. We will focus on first-order modeling of mid-ocean ridges, triple junctions, transform faults and oceanic fracture zones in future chapters, in which we will compare modeling results with observed features of actual boundary systems.

It turns out that cycloidal plate kinematics is much more interesting than the original ideas of circular relative motion, which do not work for the observed plate system of Earth.

13.7 References and some relevant texts

Argus, D.F., Gordon, R.G., and DeMets, C., 2010, Geologically current motion of 56 plates relative to the no-net-rotation reference frame: *Geochemistry Geophysics Geosystems*, v. 12, Q 1101, 13 p., doi:10.1029/2011GC003751.

Bird, P., 2003, An updated digital model of plate boundaries: *Geochemistry Geophysics Geosystems*, v. 4, no. 3, p. 1027, doi:10.1029/2001GC000252. Available online via http://peterbird.name/publications/2003_PB2002/2003_PB2002.htm

Cronin, V.S., 1987, The contribution of cycloid wobble to plate boundary strain: *Geological Society of America, Abstracts with Programs*, v. 19, p. 149.

Cronin, V.S., 1988, Cycloid tectonics -- A kinematic model of finite relative plate motion: College Station, Texas A&M University, Ph.D. dissertation, 118 p.

Cronin, V.S., 1992, Types and kinematic stability of triple junctions: *Tectonophysics*, v. 207, p. 287-301.

Cronin, V.S., 1994, Instantaneous velocity of mid-ocean ridges: *Tectonophysics*, v. 230, p. 151-159.

Klein, E.M., Smith, D.K., Williams, C.M., and Schouten, H., 2005, Counter-rotating microplates at the Galapagos triple junction: *Nature*, v. 433, 24 February, p. 855-858.

McKenzie, D.P., and Morgan, W.J., 1969, The evolution of triple junctions: *Nature*, v. 224, p. 125-133.

Schouten, H., Cande, S.C., and Klitgord, K.D., 1985, Magnetic anomaly profiles, south (22°N to 28°N), in Rabinowitz, P.D., and Schouten, H., [editors], *Mid-Atlantic ridge between 22° and 38°N*: Woods Hole, Massachusetts, Marine Science International, Ocean Margin Drilling Program, Regional Atlas Series, Atlas 11.

Tucholke, B.E., and Schouten, H., 1988, Kane fracture zone: *Marine Geophysical Researches*, v. 10, p. 1-39, 10 plates.]

Vogt, P.R., 1986, Magnetic anomalies of the North Atlantic Ocean, *in* Vogt, P.R., and Tucholke, B.E., [editors], *The western North Atlantic region*: Boulder, Colorado, Geological Society of America, *The Geology of North America*, volume M, plate 3.]

Some web resources to investigate:]

<http://www.virtualocean.org/>]

<http://www.geomapapp.org/>]

<http://worldwind.arc.nasa.gov/java/>]