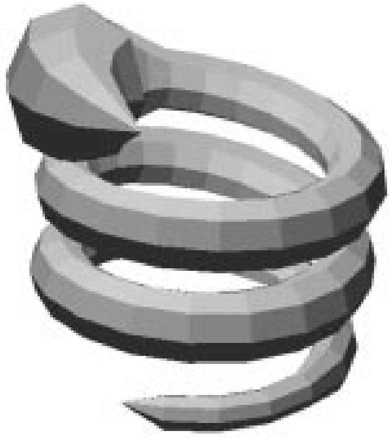

Figure 6.43*. A tube made from successive extrusions of a polygon.

Designing tubes based on 3D curves.
How do we design interesting and useful tubes and snakes? We could choose the individual matrices M_i by hand, but this is awkward at best. It is much easier to think of the tube as wrapped around a curve which we shall call the **spine** of the tube that undulates through space in some organized fashion[5]. We shall represent the curve parametrically as $C(t)$. For example, the helix (recall Section 3.8) shown in Figure 6.44a has the parametric representation

stereo pair of helix

Figure 6.44. a helix - stereo pair.

$$C(t) = (cos(t), sin(t), bt) \tag{6.11}$$

for some constant b.

To form the various waist polygons of the tube we sample $C(t)$ at a set of t-values, $\{t_0, t_1, ...\}$, and build a transformed polygon in the plane perpendicular to the curve at each point $C(t_i)$, as suggested in Figure 6.45. It is convenient to think of erecting a local coordinate system at each chosen point along the spine: the local "*z*-axis" points along the curve, and the local "*x* and *y*-axes" point in directions normal to the *z*-axis (and normal to each other). The waist polygon is set to lie in the local *xy*-plane. All we need is a straightforward way to determine the vertices of each waist polygon.

Figure 6.45. Constructing local coordinate systems along the spine curve.

It is most convenient to let the curve $C(t)$ *itself* determine the local coordinate systems. A method well-known in differential geometry creates the **Frenet frame** at each point along the spine [gray 93]. At each value t_i of interest a vector $\boldsymbol{T}(t_i)$ that is tangent to the curve is computed. Then two vectors, $\mathbf{N}(t_i)$ and $\mathbf{B}(t_i)$, which are perpendicular to $\mathbf{T}(t_i)$ and to each other, are computed. These three vectors constitute the **Frenet frame** at t_i.

Once the Frenet frame is computed it is easy to find the transformation matrix M that transforms the base polygon of the tube to its position and orientation in this frame. It is the transformation that carries the world

[5] The VRML 2.0 modeling language includes an "extrusion" node that works in a similar fashion, allowing the designer to define a "spine" along which the polygons are placed, each with its own transformation.

coordinate system into this new coordinate system. (The reasoning is very similar to that used in Exercise 5.6.1 on transforming the camera coordinate system into the world coordinate system.) The matrix M_i must carry **i**, **j**, and **k** into **N**(t_i), **B**(t_i), **T**(t_i), respectively, and must carry the origin of the world into the spine point $C(t)$. Thus the matrix has columns consisting directly of **N**(t_i), **B**(t_i), **T**(t_i), and $C(t_i)$ expressed in homogeneous coordinates:

$$M_i = \left(\mathbf{N}(t_i)|\mathbf{B}(t_i)|\mathbf{T}(t_i)|C(t_i)\right) \tag{6.12}$$

Forming the Frenet frame.
The Frenet frame at each point along a curve depends on how the curve twists and undulates. It is derived from certain derivatives of $C(t)$, and so it is easy to form if these derivatives can be caclulated.

Specifically, if the formula that we have for $C(t)$ is differentiable, we can take its derivative and form the tangent vector to the curve at each point, $\dot{\mathbf{C}}(t)$. (If $C(t)$ has components $C_x(t)$, $C_y(t)$, and $C_z(t)$ this derivative is simply $\dot{\mathbf{C}}(t) = (\dot{C}_x(t), \dot{C}_y(t), \dot{C}_z(t))$. This vector points in the direction the curve "is headed" at each value of t, that is in the direction of the **tangent** to the curve. We normalize it to unit length to obtain the **unit tangent vector** at t. For example, the helix of Equation 6.11 has the unit tangent vector given by

$$\mathbf{T}(t) = \frac{1}{\sqrt{1+b^2}}(-\sin(t), \cos(t), b) \tag{6.13}$$

This tangent is shown for various values of t in Figure 6.46a.

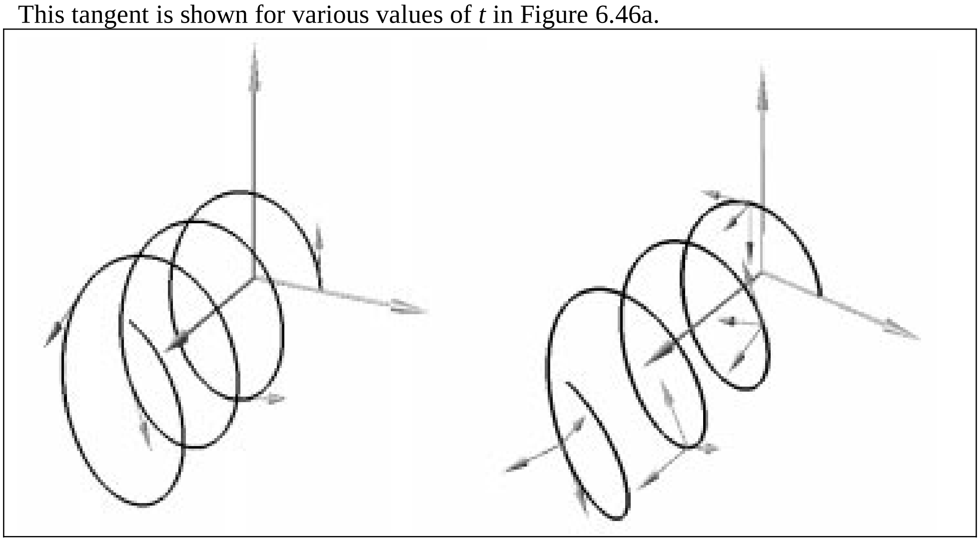

Figure 6.46*. a). Tangents to the helix. B). Frenet frame at various values of t, for the helix.

If we form the cross product of this with any non-collinear vector we must obtain a vector perpendicular to **T**(t) and therefore perpendicular to the spine of the curve. (Why?) A particularly good choice is the **acceleration**, based on the second derivative, $\ddot{C}(t)$. So we form $\dot{\mathbf{C}}(t) \times \ddot{\mathbf{C}}(t)$, and since it will be used for an axis of the coordinate system, we normalize it, to obtain the "unit **binormal**" vector as:

$$\mathbf{B}(t) = \frac{\dot{\mathbf{C}}(t) \times \ddot{\mathbf{C}}(t)}{\left|\dot{\mathbf{C}}(t) \times \ddot{\mathbf{C}}(t)\right|} \tag{6.14}$$

We then obtain a vector perpendicular to both **T**(t) and **B**(t) by using the cross product again:

$$\mathbf{N}(t) = \mathbf{B}(t) \times \mathbf{T}(t) \tag{6.15}$$

Convince yourself that these three vectors are mutually perpendicular and have unit length, and thus constitute a local coordinate system at ***C***(t) (known as a **Frenet frame**). For the helix example these vectors are given by:

$$\mathbf{B}(t) = \frac{1}{\sqrt{1+b^2}}(b\sin(t), -b\cos(t), 1)$$
$$\mathbf{N}(t) = (-\cos(t), -\sin(t), 0) \tag{6.16}$$

Figure 6.46b shows the Frenet frame at various values of t along the helix.

Aside: Finding the Frenet frame Numerically.
If the formula for ***C***(t) is complicated it may be awkward to form its successive derivatives in closed form, such that formulas for **T**(t), **B**(t), and **N**(t) can be hard-wired into a program. As an alternative, it is possible to approximate the derivatives numerically using:

$$\dot{\mathbf{C}}(t) \doteq \frac{C(t+\varepsilon) - C(t-\varepsilon)}{2\varepsilon}$$
$$\ddot{\mathbf{C}}(t) \doteq \frac{C(t-\varepsilon) - 2C(t) + C(t+\varepsilon)}{\varepsilon^2} \tag{6.17}$$

This computation will usually produce acceptable directions for **T**(t), **B**(t), and **N**(t), although the user should beware that numerical differentiation is an inherently unstable process [burden85].

Figure 4.47 shows the result of wrapping a decagon about the helix in this way. The helix was sampled at 30 points, a Frenet frame was constructed at each point, and the decagon was erected in the new frame.

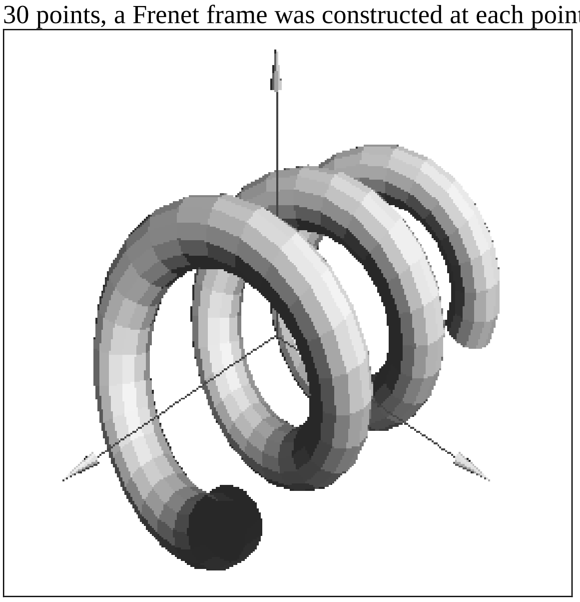

Figure 6.47. A tube wrapped along a helix.

Figure 4.48 shows other interesting examples, based on the toroidal spiral (which we first saw in Section 3.8.) [gray93, p.212]. (The edges of the individual faces are drawn to clarify how the tube twists as it proceeds. Drawing the edges of a mesh is considered in Case Study 6.7.) A toroidal spiral is formed when a spiral is wrapped about a torus (try to envision the underlying invisible torus here), and it is given by

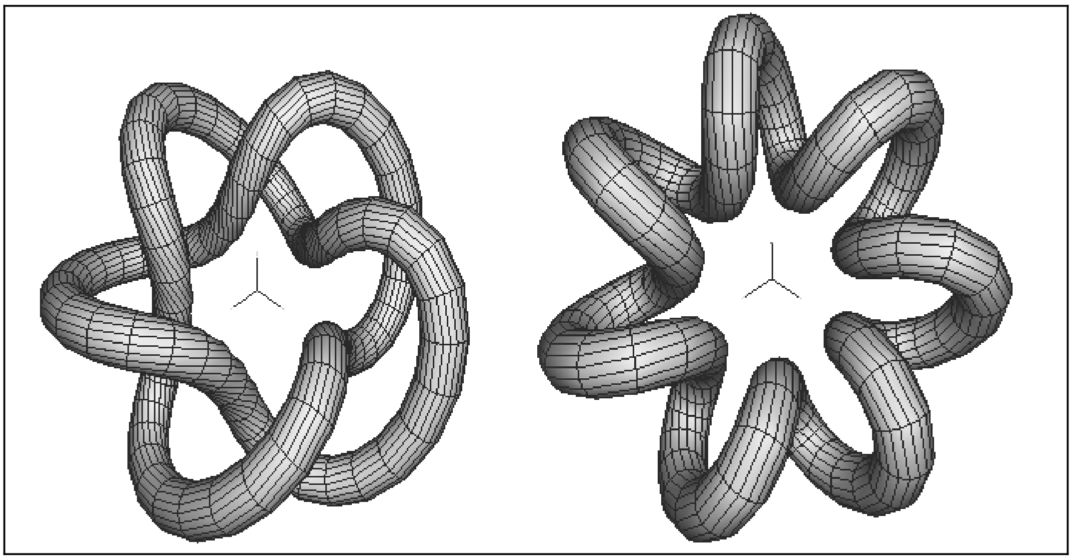

Figure 6.48. Tubes based on toroidal spirals. (file: torusKnot.bmp, file: torusKnot7.bmp)

$$C(t) = ((a + b\cos(qt))\cos(pt), (a + b\cos(qt))\sin(pt), c\sin(qt)) \quad (6.18)$$

for some choice of constants *a*, *b*, *p*, and *q*. For part a the parameters *p* and *q* were chosen as 2 and 5, and for part b they are chosen to be 1 and 7.

Figure 6.49 shows a "sea shell", formed by wrapping a tube with a growing radius about a helix. To accomplish this, the matrix of Equation 6.12 was multiplied by a scaling matrix, where the scale factors also depend on *t*:

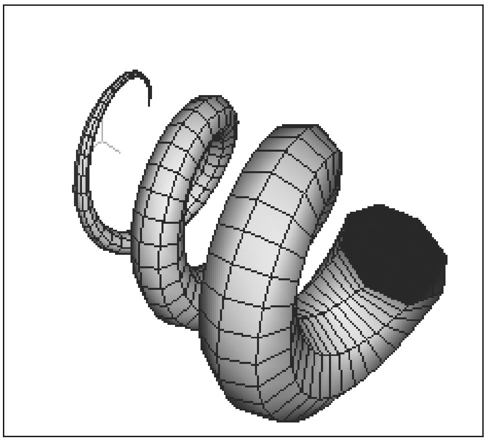

Figure 6.49. A "sea shell".

$$M' = M\begin{pmatrix} g(t) & 0 & 0 & 0 \\ 0 & g(t) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Here $g(t) = t$. It is also possible to add a rotation to the matrix, so that the tube appears to twist more vigorously as one looks along the spine.