Example of Resolving Acceleration into Tangential and Normal Components

Consider the path \( X(t) = (t^2, t) \).

\[
X := \begin{bmatrix} t^2 \\ t \end{bmatrix}
\]

\[
\text{plot([X[1],X[2], t=-3..3], scaling=constrained]);
\]

\[
V := \begin{bmatrix} 2t \\ 1 \end{bmatrix}
\]

\[
A := \begin{bmatrix} 2 \\ 0 \end{bmatrix}
\]

\[
\]

\[
\text{Speed} := \sqrt{4t^2 + 1}
\]

\[
\text{LinearAcceleration} := \frac{4t}{\sqrt{4t^2 + 1}}
\]

\[
T := \left[ \begin{array}{c} 2t \\ \sqrt{4t^2 + 1} \\ 1 \\ \sqrt{4t^2 + 1} \end{array} \right]
\]
> dTdt := diff(T,t);

\[ dTdt := \frac{2}{\sqrt{4t^2 + 1}} - \frac{8t^2}{(4t^2 + 1)^{3/2}} - \frac{4t}{(4t^2 + 1)^{3/2}} \]

> dTdt := simplify(dTdt);

\[ dTdt := \frac{2}{(4t^2 + 1)^{3/2}} - \frac{4t}{(4t^2 + 1)^2} \]

> dTds := [dTdt[1]/Speed, dTdt[2]/Speed];

\[ dTds := \left[ \frac{2}{(4t^2 + 1)^{3/2}} - \frac{4t}{(4t^2 + 1)^2} \right] \]

> kappa := sqrt(dTds[1]^2 + dTds[2]^2);

\[ \kappa := 2 \sqrt{\frac{1}{(4t^2 + 1)^3} + \frac{4t^2}{(4t^2 + 1)^4}} \]

> kappa := simplify(kappa);

\[ \kappa := 2 \sqrt{\frac{1}{(4t^2 + 1)^3}} - \frac{2t}{\sqrt{4t^2 + 1}} \]

> DivideBy := sqrt(2^2 + (4*t)^2);

\[ DivideBy := 2 \sqrt{4t^2 + 1} \]

> N := [2/DivideBy, -4*t/DivideBy];

\[ N := \left[ \frac{1}{\sqrt{4t^2 + 1}} - \frac{2t}{\sqrt{4t^2 + 1}} \right] \]

The unit normal vector N is, by definition, what we get when we normalize dT/ds. Since dT/ds and dT/dt are parallel (one is just a scalar multiple of the other, either multiply or divide by Speed), we can find N by normalizing whichever of dT/ds or dT/dt looks simpler. In this example, they are about the same. What is clear from both dT/ds and dT/dt is that they are both scalar multiples of the vector [2, -4t]. So we may as well normalize that.

> DivideBy := sqrt(2^2 + (4*t)^2);

\[ DivideBy := 2 \sqrt{4t^2 + 1} \]

> N := [2/DivideBy, -4*t/DivideBy];

\[ N := \left[ \frac{1}{\sqrt{4t^2 + 1}} - \frac{2t}{\sqrt{4t^2 + 1}} \right] \]

> DivideBy := sqrt(2^2 + (4*t)^2);

\[ DivideBy := 2 \sqrt{4t^2 + 1} \]

> N := [2/DivideBy, -4*t/DivideBy];

\[ N := \left[ \frac{1}{\sqrt{4t^2 + 1}} - \frac{2t}{\sqrt{4t^2 + 1}} \right] \]

> DivideBy := sqrt(2^2 + (4*t)^2);

\[ DivideBy := 2 \sqrt{4t^2 + 1} \]

> N := [2/DivideBy, -4*t/DivideBy];

\[ N := \left[ \frac{1}{\sqrt{4t^2 + 1}} - \frac{2t}{\sqrt{4t^2 + 1}} \right] \]

> DivideBy := sqrt(2^2 + (4*t)^2);

\[ DivideBy := 2 \sqrt{4t^2 + 1} \]

> N := [2/DivideBy, -4*t/DivideBy];

\[ N := \left[ \frac{1}{\sqrt{4t^2 + 1}} - \frac{2t}{\sqrt{4t^2 + 1}} \right] \]

> DivideBy := sqrt(2^2 + (4*t)^2);

\[ DivideBy := 2 \sqrt{4t^2 + 1} \]

> N := [2/DivideBy, -4*t/DivideBy];

\[ N := \left[ \frac{1}{\sqrt{4t^2 + 1}} - \frac{2t}{\sqrt{4t^2 + 1}} \right] \]

> DivideBy := sqrt(2^2 + (4*t)^2);

\[ DivideBy := 2 \sqrt{4t^2 + 1} \]

> N := [2/DivideBy, -4*t/DivideBy];

\[ N := \left[ \frac{1}{\sqrt{4t^2 + 1}} - \frac{2t}{\sqrt{4t^2 + 1}} \right] \]

> DivideBy := sqrt(2^2 + (4*t)^2);

\[ DivideBy := 2 \sqrt{4t^2 + 1} \]

> N := [2/DivideBy, -4*t/DivideBy];

\[ N := \left[ \frac{1}{\sqrt{4t^2 + 1}} - \frac{2t}{\sqrt{4t^2 + 1}} \right] \]

> DivideBy := sqrt(2^2 + (4*t)^2);

\[ DivideBy := 2 \sqrt{4t^2 + 1} \]

> N := [2/DivideBy, -4*t/DivideBy];

\[ N := \left[ \frac{1}{\sqrt{4t^2 + 1}} - \frac{2t}{\sqrt{4t^2 + 1}} \right] \]

> DivideBy := sqrt(2^2 + (4*t)^2);

\[ DivideBy := 2 \sqrt{4t^2 + 1} \]

> N := [2/DivideBy, -4*t/DivideBy];

\[ N := \left[ \frac{1}{\sqrt{4t^2 + 1}} - \frac{2t}{\sqrt{4t^2 + 1}} \right] \]

> DivideBy := sqrt(2^2 + (4*t)^2);

\[ DivideBy := 2 \sqrt{4t^2 + 1} \]

> N := [2/DivideBy, -4*t/DivideBy];

\[ N := \left[ \frac{1}{\sqrt{4t^2 + 1}} - \frac{2t}{\sqrt{4t^2 + 1}} \right] \]

We now have all the "bits" needed to write acceleration in terms of tangential and normal components.
\[ A = \frac{d^2s}{dt^2} T + \left(\frac{ds}{dt}\right)^2 \kappa N \]

Some extra comments....

Just to check our work...

\[
\text{HopeItEqualsA} := \text{LinearAcceleration} \cdot T + \text{Speed}^2 \kappa \text{N};
\]

\[
\text{HopeItEqualsA} := 4t \left[ \frac{2t}{\sqrt{4t^2 + 1}} - \frac{1}{\sqrt{4t^2 + 1}} \right] + 2 \left(4t^2 + 1\right)^{\frac{1}{3}} \left[ \frac{1}{\sqrt{4t^2 + 1}} - \frac{2t}{\sqrt{4t^2 + 1}} \right]
\]

Maple is having trouble combining terms, so let's look at the components separately.

\[
\text{HopeItEqualsA1} := \text{LinearAcceleration} \cdot T[1] + \text{Speed}^2 \kappa \text{N}[1];
\]

\[
\text{HopeItEqualsA1} := \frac{8t^2}{4t^2 + 1} + 2 \sqrt{4t^2 + 1} \left[ \frac{1}{\left(4t^2 + 1\right)^{\frac{3}{2}}} \right]
\]

\[
\text{HopeItEqualsA2} := \text{LinearAcceleration} \cdot T[2] + \text{Speed}^2 \kappa \text{N}[2];
\]

\[
\text{HopeItEqualsA2} := \frac{4t^2}{4t^2 + 1} - 4 \sqrt{4t^2 + 1} \left[ \frac{1}{\left(4t^2 + 1\right)^{\frac{3}{2}}} \right]
\]

Maple is still having trouble, perhaps because it is afraid that \( t \) might be a complex number. Let's tell Maple to assume \( t \) is real.

\[
\text{assume}(t, \text{real});
\]

\[
\text{simplify(\text{HopeItEqualsA1});}
\]

\[
\text{simplify(\text{HopeItEqualsA2});}
\]
A nice interpretation of "curvature".

At any point of a smooth curve, we can draw a tangent circle. In fact, we can draw a tangent circle of any given radius. Of all these circles, one fits the curve most closely (a "second order fit"). That circle is the one whose radius is \(1/\kappa\). Conversely, you can check that if \(C\) is a circle of radius \(r\), then the curvature \(\kappa(C)\) is exactly \(1/r\).

For our parabola above, when \(t=0\), \(\kappa = \)

\[\text{subs}(t=0, \kappa);\]
\[2\]

\[\text{with(plots): with(plottools): C:=circle([1/2,0],1/2, color=blue):}\]
\[\text{Warning, the names arrow and changecoords have been redefined}\]
\[\text{Warning, the name arrow has been redefined}\]

\[\text{TheCurve:=plot([X[1],X[2], t=-3..3], scaling=constrained):}\]
\[\\text{display(\{TheCurve,C\});}\]
It is not so easy to find this special circle, called the "kissing circle" (actually, "osculating circle", but go look up "osculate") at other points, because it is a little hard to locate the center. We have to go from a point on the curve out along the normal vector \( N \), but go a distance = \( 1/\kappa \). In other words, take the vector \( \frac{dT}{ds} \) and divide by the square of its length (its length=\( \kappa \)).

\begin{verbatim}
> Point:=subs(t=1,X);
    Point := [1, 1]

> dTdsThere:=subs(t=1,dTds);
    dTdsThere := [2/25, -4/25]

    LengthThereSquared := 4/125

> RadiusThere:=subs(t=1, 1/kappa);
    RadiusThere := 1/2 \sqrt{125}

> CenterOfCircle:=Point+(1/LengthThereSquared)*dTdsThere;
    CenterOfCircle := [7/2, -4]

> CircleThere:=circle([7/2, -4], sqrt(125)/2, color=blue):
> display(CircleThere,TheCurve,C);
\end{verbatim}
The idea of "osculating circle" extends to curves in $\mathbb{R}^3$ (or higher); but it takes some more work to understand how to construct the circle. We still use $1/\kappa$ as the radius, and the recipe

$$\text{CenterOfCircle} := \text{Point} + (1/\text{LengthThereSquared}) \ast d\text{There}$$

for the center. Perhaps it is easier to think of a tangent sphere, rather than a tangent circle.

Here is one final example just for those who are interested in these geometric ideas.

```plaintext
> X := [3*cos(t), 3*sin(t), t];
X := [3 cos(t), 3 sin(t), t]

> V := diff(X, t);
V := [-3 sin(t), 3 cos(t), 1]

> A := diff(V, t);
A := [-3 cos(t), -3 sin(t), 0]

> HelixCurve := tubeplot(X, t=0..9, radius=.2, scaling=constrained, axes=boxed, orientation=[15, -100]): display(HelixCurve);
```
\[
\text{Speed} := \sqrt{9 \sin(t^2) + 9 \cos(t^2) + 1};
\]

\[
\text{Speed} := \text{simplify(Speed)}; \\
\text{Speed} := \sqrt{10};
\]

\[
T := [(1/\text{Speed}) \times V[1], (1/\text{Speed}) \times V[2], (1/\text{Speed}) \times V[3]]; \\
T := \left[ -\frac{3}{10} \sqrt{10} \sin(t), \frac{3}{10} \sqrt{10} \cos(t), \frac{1}{10} \sqrt{10} \right];
\]

\[
dTdt := \text{diff}(T,t); \\
dTdt := \left[ -\frac{3}{10} \sqrt{10} \cos(t), -\frac{3}{10} \sqrt{10} \sin(t), 0 \right];
\]

\[
dTds := (1/\text{Speed}) \times dTdt; dTds := \text{expand}(dTds); \\
dTds := \left[ -\frac{3}{10} \sqrt{10} \cos(t), -\frac{3}{10} \sqrt{10} \sin(t), 0 \right];
\]

\[
\kappa := \sqrt{dTds[1]^2 + dTds[2]^2 + dTds[3]^2}; \kappa := \text{simplify}(\kappa); \\
\kappa := \frac{3}{10} \sqrt{\sin(t^2) + \cos(t^2)};
\]
> Point:=subs(t=1,X);
   \( \text{Point} := [3 \cos(1), 3 \sin(1), 1] \)

> dTds:=subs(t=1,dTds);
   \[
   dTds := \begin{bmatrix}
   -\frac{3}{10} \cos(1), & -\frac{3}{10} \sin(1), & 0
   \end{bmatrix}
   \]

   \[
   \text{LengthThereSquared} := \frac{9}{100} \cos^2(1) + \frac{9}{100} \sin^2(1)
   \]

> RadiusThere:=subs(t=1, 1/kappa);
   \( \text{RadiusThere} := \frac{10}{3} \)

> CenterOfCircle:=Point+(1/LengthThereSquared)*dTds;
   \[
   \text{CenterOfCircle} := \begin{bmatrix}
   -\frac{1}{3} \cos(1), & -\frac{1}{3} \sin(1), & 1
   \end{bmatrix}
   \]

> SphereThere:=sphere([CenterOfCircle[1], CenterOfCircle[2],
                      CenterOfCircle[3]], RadiusThere, style=HIDDEN);

> display({HelixCurve, SphereThere}, orientation=[-80, -120]);