Roadway Geometry and Coordinate Transform Simulink Blocks

Paul Griffiths, 11-17-2004

For the final project, you use the bicycle model, which describes the motion of a simulated vehicle in \( X, Y \) coordinates. Your simulated vehicle rolls around on a two-dimensional plane without any need for a roadway, however since you are interested in driving your vehicle on a road. A roadway is defined simply by a centerline, a path lying on the plane and defined in \( X, Y \) coordinates. To understand your vehicle’s lateral (side-to-side) and longitudinal (fore-aft) motion with respect to the roadway, a second coordinate system relative to the road is more convenient. We define a coordinate \( S \) to be the distance traveled along the road, and another coordinate \( N \) to be the distance away from the centerline of the road, where the centerline has an \( N \) coordinate value of zero and the right side of the road has a positive value. See Fig. 1. For a road with smooth curves and that does not intersect itself, and when the vehicle is “close” to the centerline, the mapping from \( X, Y \) to \( S, N \) coordinates is one-to-one and on-to. Despite the fact that the transformation between the coordinate systems is well defined, it is not easily determined.

![Figure 1](image_url)

**Figure 1.** Defined above are coordinate systems \( X, Y \) and \( S, N \), where the \( S \) measures the distance along the road centerline and \( N \) measures the distance to the right of the road.

The simulated vehicle motion is most easily expressed in \( X, Y \) coordinates, but the vehicles position relative to the road is more conveniently expressed in \( S, N \) coordinates.

We have provided several Simulink blocks that allow you to access the roadway geometry and transform back and forth between \( X, Y \) and \( S, N \) coordinates. There are three basic blocks for accessing roadway geometry,

- **Road Geometry Centerpoint P:** determines the \( X, Y \) coordinates \((x, y)\) of a point on the road centerline, a distance \( s \) down the road from the origin.
- **Road Geometry Right Vector**: determines the \(X, Y\) components \((rx, ry)\) of a unit vector pointing to the right (perpendicular to the centerline) of a point on the road centerline, a distance \(s\) down the road from the origin.

- **Road Geometry Curvature**: determines the curvature \(\kappa\) (1/radius of curvature) of the road centerline a distance \(s\) down the road from the origin.

A fourth roadway geometry block, **Road Geometry Forward Vector**, is the same as **Road Geometry Right Vector** except that it determines a unit vector tangent to the road centerline and pointing in the direction of increasing \(S\) coordinate values. This block simply rotates the vector from **Road Geometry Right Vector** by 90 degrees using \((fx, fy) = (-ry, rx)\).

There are two coordinate transformation blocks; one converts \(S,N\) to \(X,Y\) coordinates and the other block performs the inverse transformation.

- **Roadway Coordinate Transform \((s,n) \rightarrow (x,y)\)**: given \(s\) and \(n\), this block determines \(x\) and \(y\). **Road Geometry Centerpoint P** finds \(X,Y\) coordinates \((px,py)\) for at the center of the road, a distance \(s\) down the road. **Road Geometry Right Vector** returns a unit vector, which when scaled by \(n\) and added to \((px,py)\), equals \((x,y)\).

- **Roadway Coordinate Transform \((x,y) \rightarrow (s,n)\)**: given \(x\) and \(y\), this block converges to the correct \(s\) and \(n\). This block is a masked subsystem, so double-click the block to see the implementation. A feedback stabilized closest-point algorithm is used to determine the closest point on the road centerline to the input point \((x,y)\). This determines the \(S\) coordinate and then the \(N\) coordinate is simply the distance from the closest-point on the road center line to the point \((x,y)\).