

Digital Map Concept

Track Geometry for ETCS Onboard Map

Andreas Wenz^{1*}

Swiss Federal Railways, Switzerland

andreas.wenz@sbb.ch

November 17, 2021

Contents

1	Introduction, Background and Scope	1
2	Coordinate Systems	2
3	Map Architecture	3
4	Geometric Objects Calculations	6
4.1	Position Layer	6
4.1.1	Line	6
4.1.2	Circle section	7
4.1.3	Track transition curves	8
4.2	Gradient Layer	9
4.2.1	Lines	9
4.2.2	Circle Section	10
4.3	Cant Layer	11
4.3.1	Line	11
5	Analysis of Data Volume	11
6	Conclusion	12
	Appendix A: Numerical Calculation of Clothoids	12

1 Introduction, Background and Scope

This document shows an approach to model the track geometry so that it can be used as an on-board digital map for train localisation and compare it against a point-based modelling approach. The goal of this documents is to introduce designers of both on-board map system and the on-board localisation system to the two approaches and highlight their advantages and disadvantages.

An onboard digital map is needed for train localisation in order to convert three dimensional position measurements into one-dimensional positions on the track. In addition, track features such as azimuth, cant and gradient can be used to validate position measurements [1].The map needs to be a safe source of information in order to be useful for these applications.

The goal is to define a model that is vector based so that the track features can be captured as accurately as possible while keeping the data volumes to be transferred to the train on-board systems low. The vector model is based on the fact that railway tracks are not constructed with arbitrary geometries, but are constructed using well defined geometric primitives such as lines, circles and clothoids. We can exploit that and formulate a model based on a track segmentation at the transition of these primitives. A track can have different segment borders in the horizontal, vertical and longitudinal projections, therefore three different layers are created to reflect that. More details on the segmentation are given in Section 3.

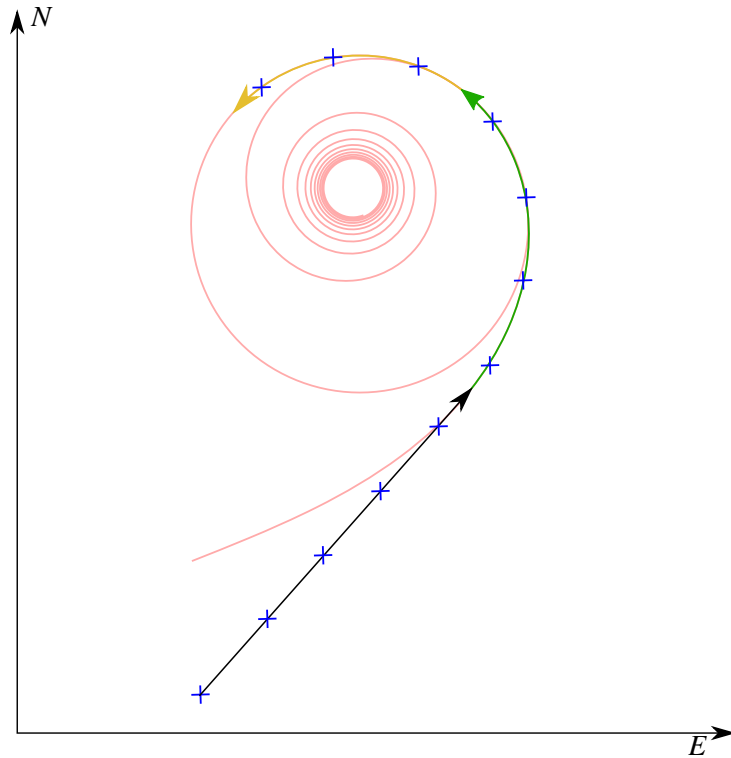


Figure 1: Vector and point based track geometry

Colours indicate different geometric objects, black is a line segment, green a track transition curve and yellow a circle section, shown in light red are the underlying geometric primitives, the blue marks indicate the point locations.

An illustration highlighting the differences between the vector and point based approaches is shown in Figure 1.

The vector-based approach has several advantages over a point based approach: The accuracy of the map is not limited by the discretisation and an interpolation between the track points is not necessary. In addition, the feature trajectories of different segments have to be continuous at the boundaries of the segments which can be exploited in map validation. The main disadvantages of the vector over a point based approach is an expected higher complexity to process map vectors for localisation functions compared to point-based models, which can be challenging when processing at runtime. It should be noted that ultimately the choice of the data model needs to be done according to the needs of the localisation algorithm.

2 Coordinate Systems

The measurement of the parameters of a geometric track object is based on geodetic measurements taken. These measurements, are typically taken in a locally flat coordinate system. Also the attitude of a vehicle is typically expressed, as rotation angles between the vehicle's body frame and the flat coordinate system. Therefore, it is useful to describe the map in a locally flat coordinate system. The transformations between the different coordinate systems of use have to be well defined and implemented.

Figure 2 shows the definitions of three different coordinate systems:

1. The earth centered earth fixed frame (**ECEF**), where the x axis is pointing to the meridian, the z axis is pointing to the north pole and the y axis is constructed using the right hand rule. The ECEF is typically used by global navigation satellite systems.
2. A local flat coordinate system called the **navigation** or **NED frame**, where the x axis is pointing north, the y axis is pointing east and the z axis is pointing towards the earth's centre.

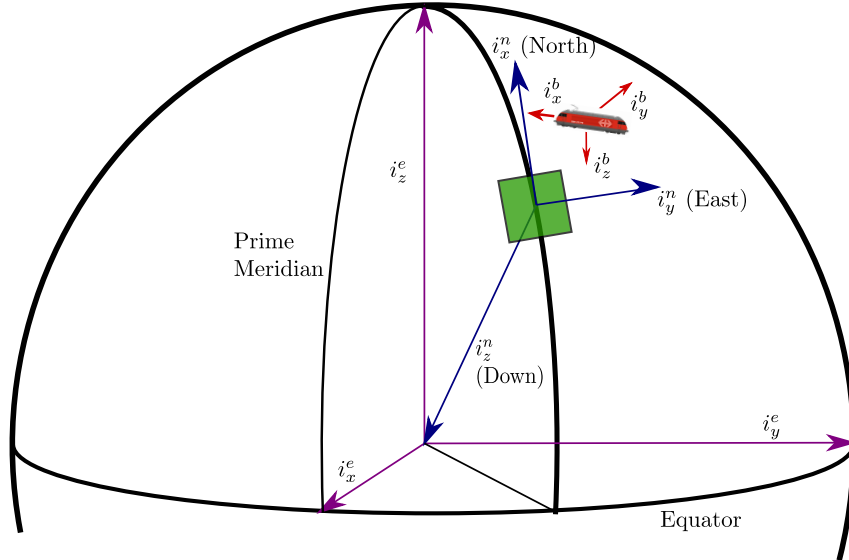


Figure 2: Definition of Coordinate Systems

Earth Centered Earth Fixed (**ECEF frame**) (i^e), **Navigation frame**(i^n) and **Body frame** (i^b)

3. The **body frame**, which is fixed to the body frame of the vehicle. Typically the x axis is pointing in longitudinal direction of the vehicle, the z axis downwards and the y axis is constructed again using the right hand rule.

For the here discussed approach to a digital map based on geometric track objects, mainly the navigation frame is of concern. Track elements are typically recorded by geometers in this frame. Common frames are the universal transverse mercator (UTM) coordinate system or the more accurate military grid reference system (MGRS). The here presented approach is agnostic to the used projection as long as the necessary transformations are known and do not add further inaccuracies (e.g. a problem in Germany with old DHDN and new DB-REF coordinate systems). The coordinate system used should fulfil the following requirements:

1. Global or at least European coverage (outside polar regions)
2. Scale error smaller than 0.1 m (to be validated)
3. If the coordinate frame is split into approximation cells, the transitions need to be defined
4. Fixed to the European tectonic plate

3 Map Architecture

Shown in Figure 4 is a hierarchical view of the proposed data structure. A track is split into nodes and edges. Where nodes can be the entry points of a railway point, buffer stops or system borders. This means that a typical railway point consists of three nodes and two edges. An example for the node-edge at a railway point is shown in Figure 3.

A track edge is then further segmented (see Figure 5) for each of the three different layers defined on the track geometry which together form a complete description of the 7 degrees of freedom (north, east, down coordinates $\mathbf{p} = [p_n \ p_e \ p_d]$, radius r , azimuth ψ , cant c and gradient g) of any track point. In addition we will convert cant and gradient to cant angle ϕ and gradient angle θ for ease of comparison to the attitude of a vehicle on the track. For each of these layers the track is split into segments on which the geometry can be described by a geometric primitive with a set of parameters. Essential for the calculation of the desired output parameter is the distance from the start of the segment to the track point. Since the segmentation is different for each layer this has to be done for each layer separately.

Figure 6 shows the on board map architecture and the functions that have to be performed in order to compute the track feature for a given sensor position measurement $\mathbf{p}^s = [p_n^s \ p_e^s \ p_d^s]$ with driving

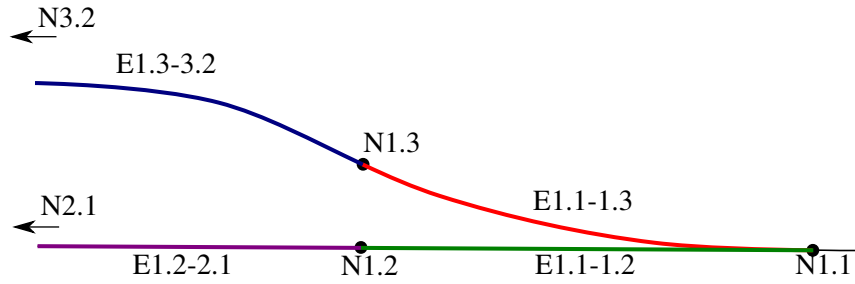


Figure 3: Example for the node-edge model at a switch point

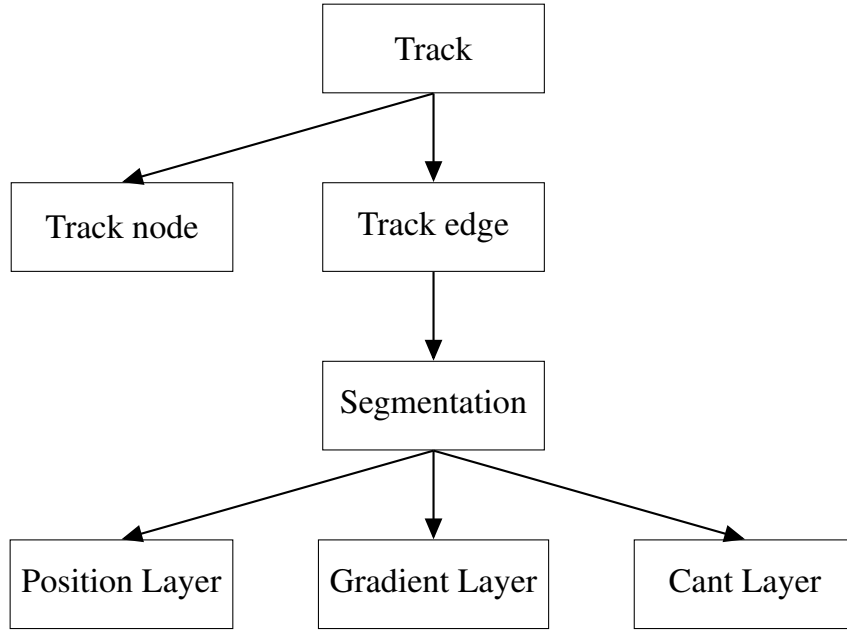


Figure 4: Data model

direction δ_s . In the function *Get Track Edge* the sensor position the closest track edge is found and the sensor position is projected onto that edge. The resulting information is the track edge ID ID_t , the distance along the edge d_t and the orientation of the track edge in comparison to the movement direction δ_t . Note that depending on the estimated position by the sensor and its uncertainty several projections could be possible. In this case a separate functions solving the track selection problem might be needed.

The function *Get Segments* uses this information, to together with the segmentation information, to compute the segment IDs ID_p, ID_g, ID_c of each Layer and the distances to the start of the respective segment d_p, d_g, d_c . The different distance definitions are shown in 5.

We than use this to identify the type of geometric objects, which will be defined in Section 4, for each layer and get the needed parameters \mathbf{P} . With these parameters and the distance along the segment the outputs can then be calculated.

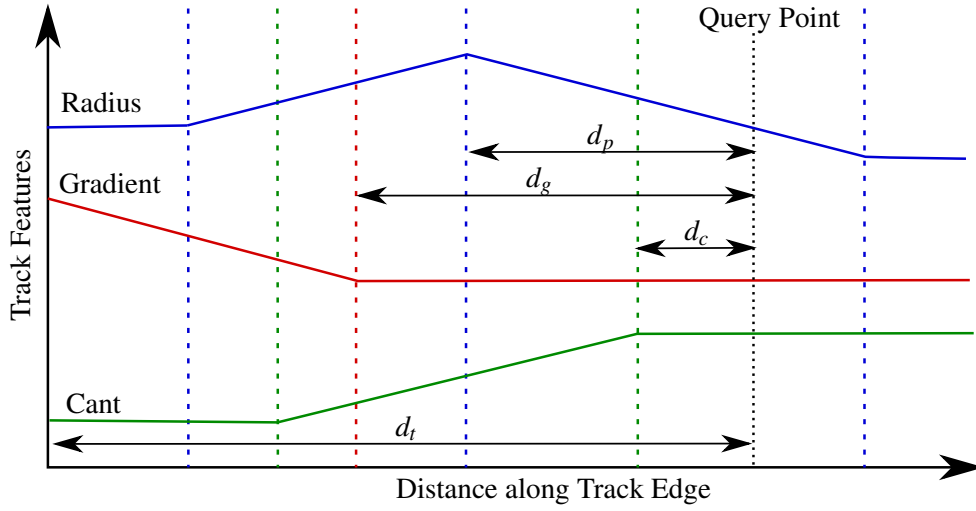


Figure 5: Segmentation of a track edge in different layers

blue: position layer, red: height layer, green: cant layer, dashed lines indicate segment borders

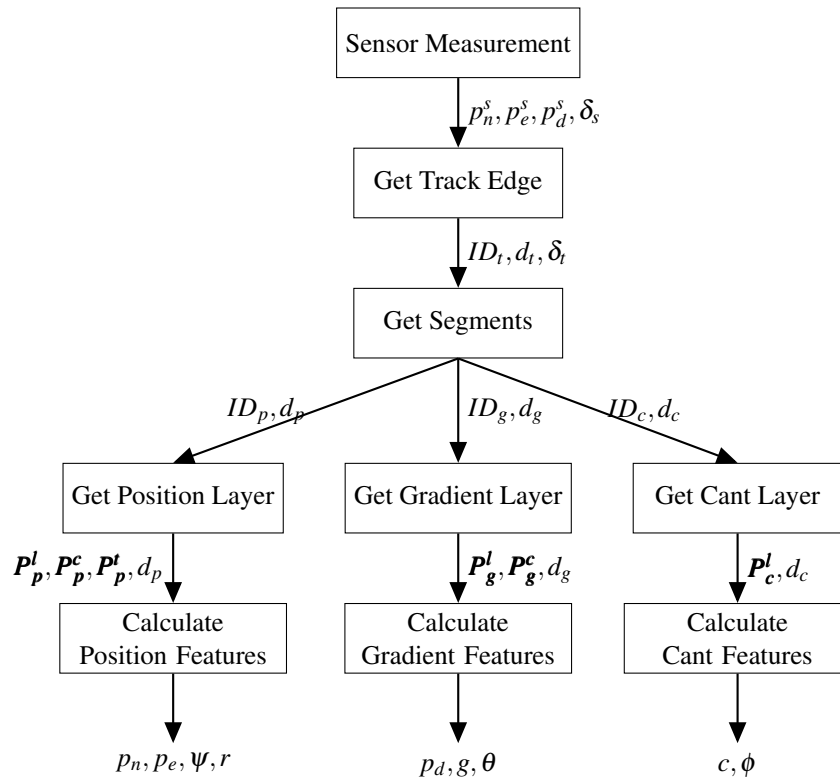


Figure 6: Processing of vector-based map information within the localisation algorithm

4 Geometric Objects Calculations

This section describes the mathematical formulas needed to calculate the track features for each layer.

4.1 Position Layer

The position layer is a projection of the track geometry on the north-east plane. It includes information that allows the calculation of the north (p_n) and east coordinates (p_e) as well as the radius(r) and azimuth(ψ) at any given query point on a given track segment having a distance d_p from the start of the position segment. On the position segments the following geometric primitives are used:

- Lines,
- Circle sections, for curves
- Track transition curves, defined by Euler spirals (clothoids)

4.1.1 Line

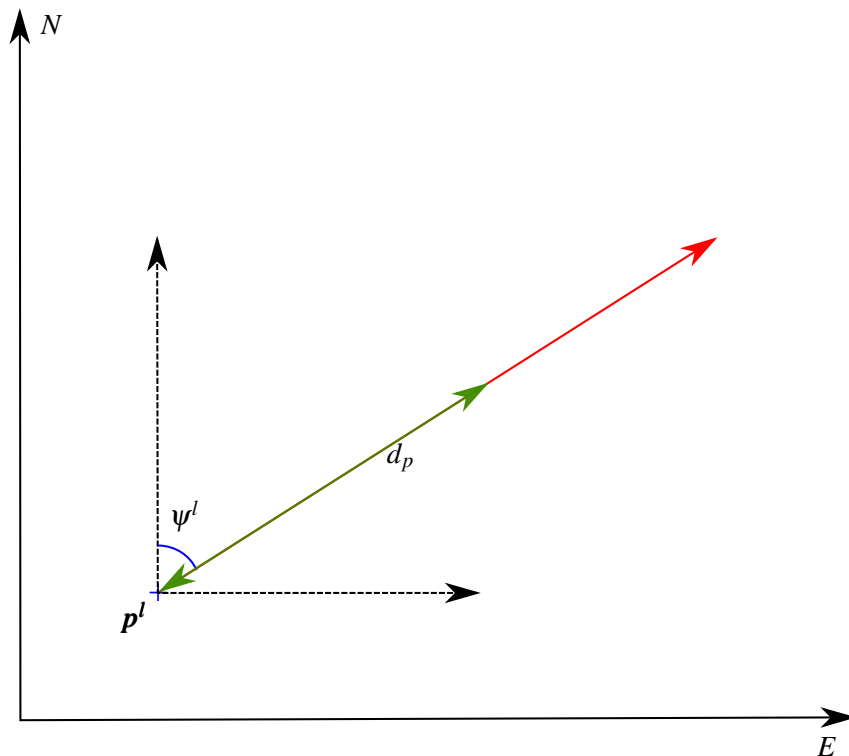


Figure 7: Line in position layer

The green arrow shows the distance between the starting point and the query point

A line has the following parameters, shown in Figure 7:

- Starting Point ($\mathbf{p}^s = [p_n^l, p_e^l]$): North, East coordinate
- Azimuth (ψ^l): Angle between the north direction and the line
- Distance (d_p): Distance along line from the beginning of the segment

The desired track features can then be calculated as follows:

$$p_n = p_n^l + d_p \cos(\psi) \quad (1)$$

$$p_e = p_e^l + d_p \sin(\psi) \quad (2)$$

$$\psi = \psi^l \quad (3)$$

$$r = \text{sign}(\psi^l) \infty \quad (4)$$

4.1.2 Circle section

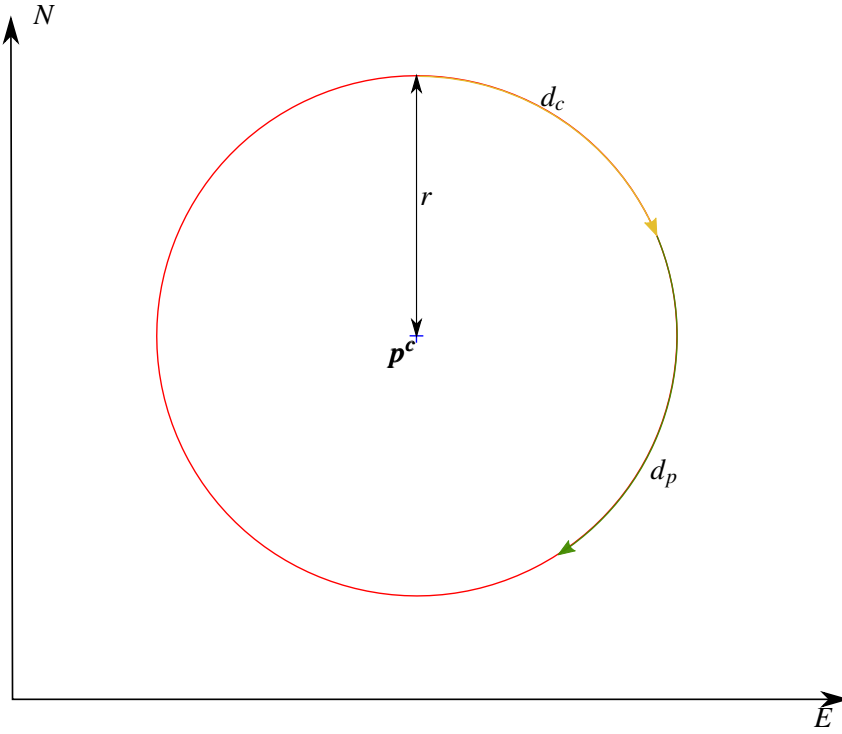


Figure 8: Circle section

The green arrow shows the distance between the starting point and the query point

A circle section has the following parameters, shown in Figure 8:

- Centre Point ($\mathbf{p}^c = [p_n^c, p_e^c]$): North, East coordinate of the circle centre
- Radius (r^c): Radius of the curve with sign.¹
- Initial arc length (d_c): Distance from the north pole of the circle to starting point
- Distance (d_p): Distance along circle section from the starting point to the query point

The desired track features can then be calculated as follows:

$$p_n = p_n^c + \|r\| \cos\left(\frac{d_c + d_p \text{sign}(r)}{\|r\|}\right) \quad (5)$$

$$p_e = p_e^c + \|r\| \sin\left(\frac{d_c + d_p \text{sign}(r)}{\|r\|}\right) \quad (6)$$

$$\psi = \text{sign}(r) \frac{\pi}{2} + \frac{d_c + d_p \text{sign}(r)}{\|r\|} \quad (7)$$

$$r = r^c \quad (8)$$

¹A circle section having its centre point to the right of the track has a positive radius, whereas a circle section having its centre point to the left of the track has a negative radius. In this case the track direction is defined in azimuth direction.

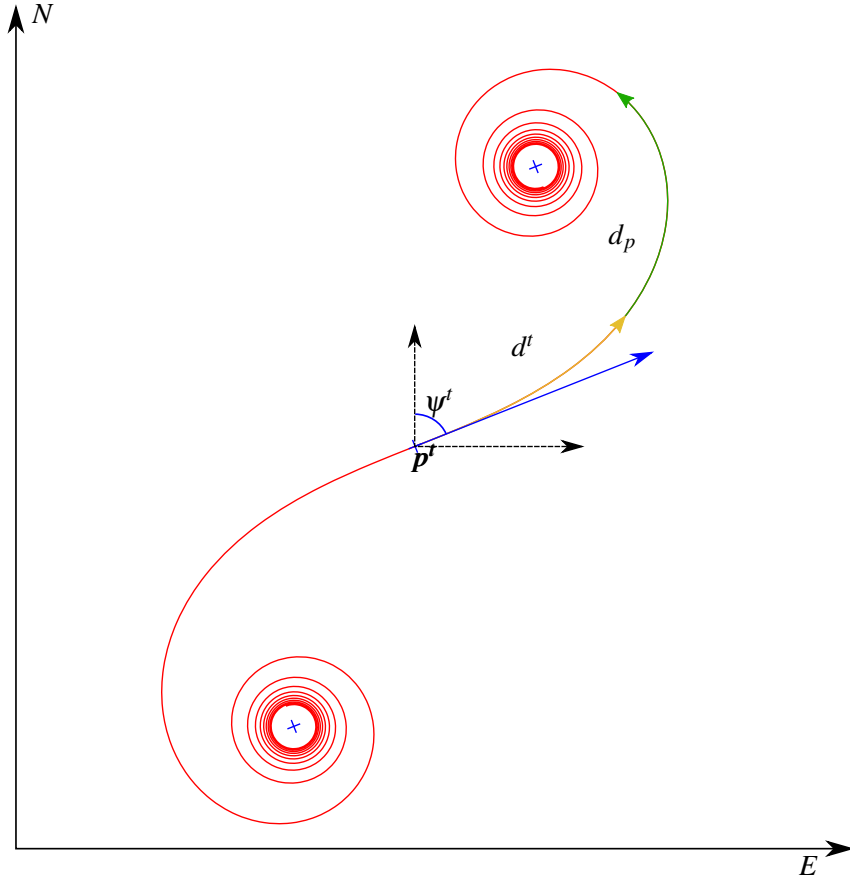


Figure 9: Track transition curve for clothoid constant $k^t > 0$

4.1.3 Track transition curves

A track transition curve section, is part of a clothoid and is used to transition between straight line segments and circle section segments. The radius is depending on the distance to the centre point of the clothoid following the equation:

$$r(d) = \frac{(k^t)^2}{d} \quad (9)$$

The track transition curve has the following parameters, shown in Figure 9:

- Centre Point ($\mathbf{p}^t = [p_n^t, p_e^t]$): North, East coordinate of the clothoid centre
- Clothoid constant (k^t): Parameter
- Initial arc length (d^t): Distance from the centre point of the clothoid to starting point
- Initial Azimuth (ψ^t): Angle between the north direction and the orientation of the clothoid centre point
- Distance (d_p): Distance along the clothoid from the starting point to the query point

The desired track features are given by the following equations:

$$\begin{bmatrix} p_n \\ p_e \end{bmatrix} = \mathbf{p}^t + \begin{bmatrix} \cos \psi^t & -\sin \psi^t \\ \sin \psi^t & \cos \psi^t \end{bmatrix} \int_0^{d^t+d_p} \begin{bmatrix} \cos \left(\frac{s^2}{2(k^t)^2} \right) \\ \sin \left(\frac{s^2}{2(k^t)^2} \right) \end{bmatrix} ds \quad (10)$$

$$\psi = \psi^t + \frac{(d^t + d_p)^2}{2(k^t)^2} \quad (11)$$

$$r = \frac{(k^t)^2}{(d^t + d_p)} \quad (12)$$

Note that (11) can only be solved approximately. Several methods exist for approximating the integrals, one possibility is shown in the Appendix 6.

4.2 Gradient Layer

The gradient layer is a projection of the track geometry on the along track - down plane. It includes information that allows calculation of the track gradient (pitch) angle θ and the height of the track above sea level (h). This can be calculated at any query point on the track having a distance d_g from the beginning of the gradient layer segments. The geometric primitives used are:

- Lines: Used for ramps or flat track sections
- Circle sections: Used for transitions between different line sections

4.2.1 Lines

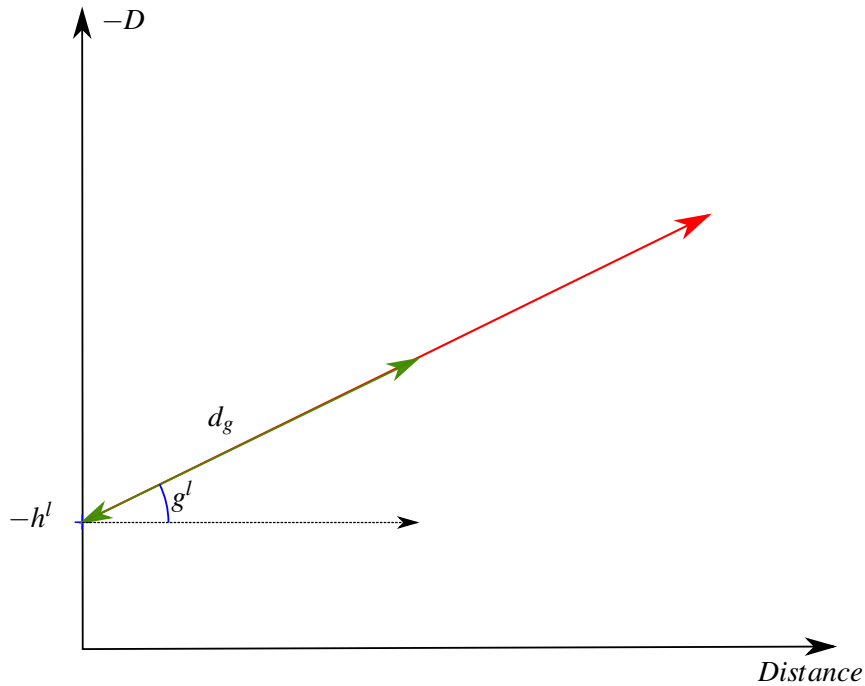


Figure 10: Line in gradient layer

A line has the following parameters, illustrated in Figure 10:

- Starting altitude (h^l): Altitude at starting point
- Gradient (g^l): Inclination of the track in permille
- Distance (d_g): Distance along line from the beginning of the segment

The desired track features can then be calculated as follows:

$$p_d = -h^l - d_g \frac{g^l}{1000} \quad (13)$$

$$g = g^l \quad (14)$$

$$\theta = \arctan \frac{g^l}{1000} \quad (15)$$

$$(16)$$

4.2.2 Circle Section

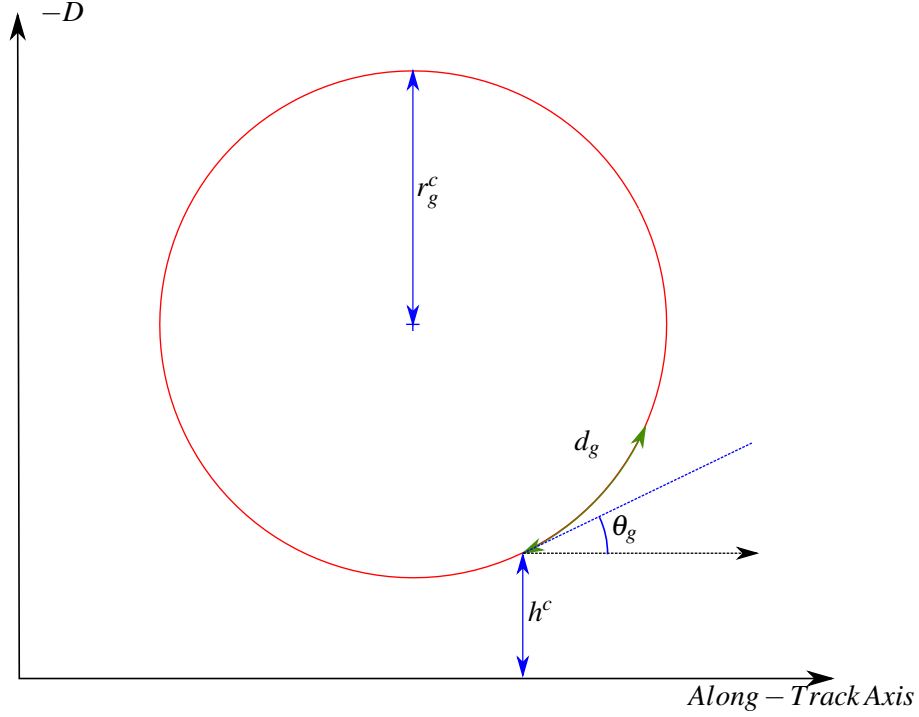


Figure 11: Circle in gradient layer

A circle section has the following parameters, shown in Figure 11:

- Radius (r_g^c): Radius of circle ²
- Initial height (h^c): Height at starting point
- Initial gradient angle (θ^c): Angle between the north-east plane and the track at starting point
- Distance (d_g): Distance along circle section from the starting point to the query point

The desired track features can then be calculated as follows:

$$p_d = -h^c - \|r_g^c\| \sin \left(\theta^c - \text{sign}(r_g^c) \frac{\pi}{2} \right) - \text{sign}(r_g^c) \sqrt{(r_g^c)^2 - (d_g - L)^2} \quad (17)$$

$$L = \|r_g^c\| \cos \left(\theta^c - \text{sign}(r_g^c) \frac{\pi}{2} \right) \quad (18)$$

$$\theta = \arctan \left(\frac{\text{sign}(r_g^c) \sqrt{(r_g^c)^2 - (d_g - L)^2}}{d_g - L} \right) - \text{sign}(r_g^c) \frac{\pi}{2} \quad (19)$$

$$g = 1000 \cdot \tan(\theta) \quad (20)$$

²A circle section of the gradient layer having its centre point above the track has a positive radius, whereas a circle section having its centre point below the track has a negative radius. In this case the track direction is defined in azimuth direction.

4.3 Cant Layer

The cant layer is the projection of the track geometry onto the cross-track - down plane. It includes information that allows calculation of the cant angle (roll angle) ϕ at any query point with distance d_c . The cant angle only has one geometric primitive:

- Line: Used for constant cant segments and transitions between constant cant segments

4.3.1 Line

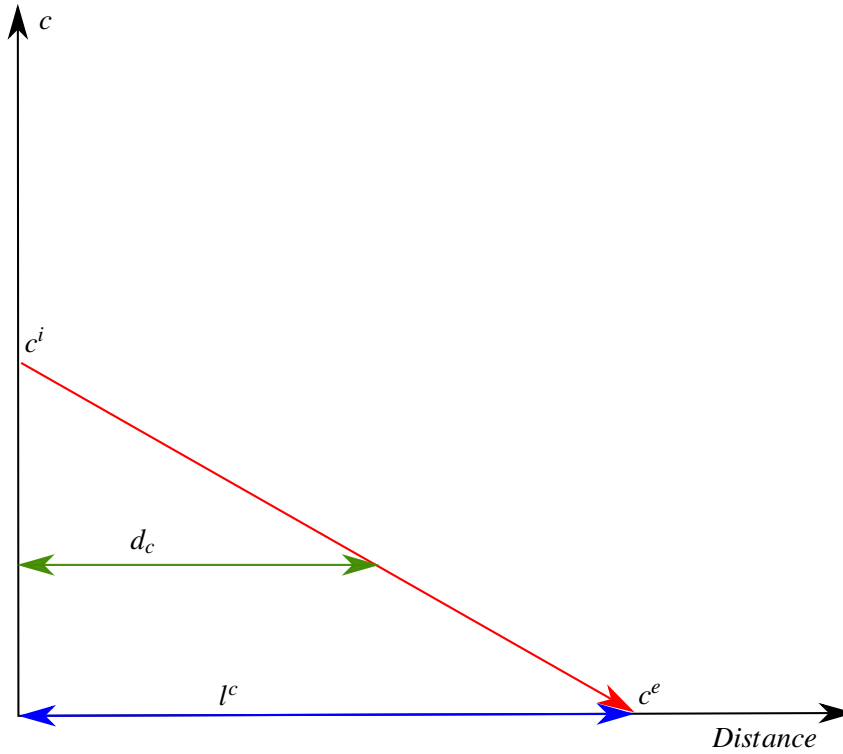


Figure 12: Line in cant layer

A line in the cant layer has the following parameters (Figure 12):

- Initial cant (c^i): Cant in meters at start of segment
- End cant (c^e): Cant in meters at end of segment
- Segment length (l^c): Length of the segment in meters
- Track width(t^c): The distance between the centre of the running surface on both rails
- Distance (d_c): Distance along line between starting point and query point

The desired track features can then be calculated as follows:

$$c = c_i + \frac{c^e - c^i}{l^c} d_c \quad (21)$$

$$\phi = \arctan\left(\frac{c}{t^c}\right) \quad (22)$$

5 Analysis of Data Volume

We have analysed the data volumes needed for covering the Swiss normal gauge network with the geometric object approach shown here and a point based approach with 10m spacing. Note that in practice

the distance between points should be aligned with accuracy requirements and actual change of radius, gradient and cant. The point based approach is based on a node-edge model, where each track point has a reference to a track edge.

	No. Entries	Size
Track Segmentation	426461	35.796 MB
Position Layer	196434	37.589 MB
Gradient Layer	191660	24.313 MB
Cant Layer	117254	12.506 MB
Total	12101 km	115.83 MB
Size per km		0.010 MB
Track Edges	32589	1.308 MB
Track Point	1001478	84.570 MB
Total	12628 km	85.878 MB
Size per km		0.007 MB

Table 1: Data amounts for the geometry and point based modelling approaches for the Swiss normal gauge network

In table 1 we show the data needs for the two approaches. We see that the point based approach has somewhat lower data needs for the whole track network than the vector based approach. If we calculate the data amount per kilometre track length, the difference is about 3kB per kilometre.

In addition, we have calculated the data requirements if the map data is to be sent with the ETCS movement authority (MA). For this case we have analysed a dataset consisting of MAs from the Swiss ETCS Level 2 lines. An MA consists of an array of balise group IDs with the distances between the balise groups. We call one segment between two balise groups a MA-element. We have identified 3887 unique MA-elements in our dataset. One MA can have up to 31 of these elements. The mean data volume which would have to be added to one MA-element for the point based approach is 5.8 kB whereas the here presented approach has a mean added data volume of is 4.21 kB. Note that the ETCS Level 2 lines in Switzerland are mainly high-speed lines with mainly straight line segments or long curves, reducing the number of geometry objects needed per kilometre.

6 Conclusion

An approach to modelling the track geometry of railway lines for onboard localisation using a map vectors has been presented. This vector-based approach has been compared to a point-based approach. The differences in the data amounts required to be transferred to the on-board system a minor, however the accuracy gains by the vector-based approach could be significant and need to be analysed further.

The final selection of a modelling approach should be based on the accuracy and processing requirements of the localisation algorithm of the Vehicle Locator.

Appendix A: Numerical Calculation of Clothoids

The Fresnel Integrals can be rewritten to power series:

$$\int_0^{d^t+d_p} \cos\left(\frac{s^2}{2(k^t)^2}\right) \approx \sum_{i=0}^N \frac{(-1)^i \tau^{4i+1}}{(2i)! 4i+1} \quad (23)$$

$$\int_0^{d^t+d_p} \sin\left(\frac{s^2}{2(k^t)^2}\right) \approx \sum_{i=0}^N \frac{(-1)^i \tau^{4i+3}}{(2i+1)! 4i+3} \quad (24)$$

$$\tau = \frac{d^t + d^p}{\sqrt{2k^t}} \quad (25)$$

We than iterate the power series until the coordinates have reached conversion ($\Delta p < 0.001m$) or a maximum number of iterations has been reached. In the later case an error code is outputted.

References

- [1] A. Wenz, “Integrity Measures and Verification for Train Localisation,” Swiss Federal Railways, Tech. Rep., 2020. [Online]. Available: https://www.smartrail40.ch/service/download.asp?mem=0&path=%5Cdownload%5Cdownloads%5CIntegrity_v1.2.pdf

List of Figures

Figure 1	Vector and point based track geometry	2
Figure 2	Definition of Coordinate Systems	3
Figure 3	Example for the node-edge model at a switch point	4
Figure 4	Data model	4
Figure 5	Segmentation of a track edge in different layers	5
Figure 6	Processing of vector-based map information within the localisation algorithm	5
Figure 7	Line in position layer	6
Figure 8	Circle section	7
Figure 9	Track transition curve for clothoid constant $k^t > 0$	8
Figure 10	Line in gradient layer	9
Figure 11	Circle in gradient layer	10
Figure 12	Line in cant layer	11

List of Tables

Table 1	Data amounts for the geometry and point based modelling approaches for the Swiss normal gauge network	12
---------	---	----