2011 50th IEEE Conference on Decision and Control and
European Control Conference (CDC-ECC)
Orlando, FL, USA, December 12-15, 2011

Optimization-based Road Curve Fitting

Sheng Zhao and Jay A. Farrell

Abstract— **Various advanced driver assistance systems (ADAS) are under development that intend to provide improved road safety. These systems require precise road models. In particular, accurate curvature is important for some ADAS applications such as curve over speed and lane departure warning. Existing road models often employ spline functions that are fit by least squares to roadway position data. The curvature calculated for such spline curves may not accurately reflect the curvature of the underlying roadway. This article addresses this problem in an unified framework, using optimization with ℓ_1-norm regularization. In this approach, known roadway characteristics can be enforced optimally with respect to a cost function which finds the best tradeoff between the match to the available data and the number of changes in curvature. Experimental results with show that the proposed method chooses a sparse set of curvature switching points (i.e., piecewise constant curvature) and achieves a high accuracy fit to the roadway dataset.**

I. Introduction

In advanced driver assistance systems (ADAS) or Vehicle Assist and Automation (VAA) [1], a digital map with a precise model of the road is necessary. ADAS combines vehicle state information [2] with trajectory information from the digital map [3] to provide the driver with important information: over-safe-speed warnings when driving on curved roads [4], road/lane departure warnings and so forth. These applications require the map to have lane-level accuracy (pos. error < 0.5 m) and contain a precise model of the road in the sense that attributes of the road, such as the curvature, are accurate [3], [5]. It is easy to see the importance of roadway position accuracy for ADAS applications, while the accuracy requirement for the other attributes is less obvious. Accurate curvature is necessary to compute the maximum safe speed, for different vehicle classes, on curved roads for driver assistance or autonomous cruise control systems. Moreover, curvature can be used to adjust the beam pattern of the vehicle headlamps to minimize the driver's lost visual field caused by road curvature [6]. Inaccuracy of the curvature, in particular, incorrect sign of the curvature dramatically affects the performance of such applications. Therefore, the shape of the roadway, in particular, its curvature must be accurately represented; in particular, straight line connections between nodes is not adequate.

Current digital maps, developed to support routing applications, are not accurate enough to support ADAS applications. Accurate road shape and curvature, which were not high priority specifications in the current generation of maps [4], are vital attributes for future generations of maps. This paper focuses on the reconstruction of the road trajectory to achieve high accuracy position and curvature. The overall roadway trajectory fit procedure involves several steps:

1) **Data Acquisition:** An instrumented car drives along the roadway storing roadway relevant information, e.g., inertial, GNSS, imagery, LIDAR.
2) **Smoothing:** This is a post-processing step that combines the time-stamped roadway information to compute optimal estimates of positions along, for example, each lane edge or centerline. The result is an ordered, closely spaced, sequence of points that represent a curve of interest for the map. Herein, we represent such a data set as
$$\mathcal{D} = \{(t_k, N_k, E_k, D_k)\}_{k=0,\ldots,N}$$
where k counts over the samples, t_k is the measurement time, and (N, E, D) denote the north, east, and down coordinates in a local tangent plane.
3) **Road Curve Fitting:** While $\mathcal{D}$ represents the curve, it is not convenient for storage. Due to the extremely large cumulative number of lane miles to be stored, more efficient data storage representations are desirable. This step converts the data sequence $\mathcal{D}$ into an analytic curve $\gamma(s)$ that is convenient both for storage and for realtime applications.
4) **Storage:** The parameters of the analytic curve fit $\gamma(s)$ are stored in a GIS database along with representations of other lanes, roads, and roadway features.
5) **Real-time Application:** Based on a region of interest, GIS database queries extract curves $\gamma(s)$ necessary for realtime applications. The applications may use the analytic representations in various ways: graphic displays, user warnings, search for nearest trajectory point, etc.

The focus of this paper is on Step 3, the problem of road curve fitting.

II. Technical Problem Statement

The roadway data curve fit is subject to constraints derived from road design rules [7], [8]. One rules is that the road is composed of three basic elements [7]: straight lines, circular curves and transition curves. If we consider straight segments as a special case of circular curves with curvature equal to zero, then the road only consist of circular curves and transition curves. Transition curves are used to connect two circular curves and are often designed to have constant curvature change [9]. Transition curves can either be explicitly modeled as in [7] or approximated as circular curves as in [4]. As argued in [4], the constant curvature approximation

S. Zhao and J. A. Farrell are with the Department of Electrical Engineering, University of California, Riverside, CA 92521 USA, {shzhao,farrell}@ee.ucr.edu

is reasonable because the length of the transition curves is often small. This paper follows the convention, as in [4], to model the road as a curve with segments of piecewise constant curvature. Moreover, while the real road is three-dimensional, the two dimensional flat planar assumption is often used for each local road segment. This paper also assumes that the road is locally two-dimensional.

The problem of road curve fitting can be stated as follows: Given a set of position data $\mathcal{D} = \{N_k, E_k\}_{k=0}^{N}$ representative of a lane centerline, fit a piecewise C^2 curve, with piecewise constant curvature to the data.

Extension to three dimensional data and to lane edges is straightforward.

III. Literature Review

The existing methods to curve fit roadways fall into two main categories: spline based curve fitting [3], [8] and parametric curve fitting [4], [7]. In spline based methods, the user selects a spline basis function, then selects the spline parameters to fit the position data. Road attributes such as the tangent and curvature are computed, from the first and second derivatives of the spline by the application software. There are a few difficulties with the spline approach. First, the spline argument is the arclength s and the arclength in not known part of the data set $\mathcal{D}$. Instead, arclength is estimated from the data. Second, the spline curve fit does not naturally enforce the piecewise constant curvature constraint. Piecewise constant curvature can be attained by selecting splines of the appropriate order. For splines, the points of curvature change are defined by the knot locations; therefore, the knot locations would need to be optimally selected. Therefore, a preprocessing step may be required to estimate arclength and curvature so that the knots can be allocated as a function of arclength at points of curvature discontinuity. Nevertheless, the curvature derived from curve fit splines has high variance that is not related to the actual roadway shape [3], [4]. Third, the spline fitting procedure is not robust to errors in the position data [3], [4]. In contrast to the spline based methods, recent papers [4], [7] report an alternative method based on parametric curve fitting. This method first computes the local pointwise curvature using three consecutive points. Then it segments the road by searching for curvature discontinuities. Finally, each road segment is fit to a parameterized circle function. Neither of the above procedures are optimal. Each has various ad-hoc aspects.

In [10], the authors formulate a related trajectory generation problem as an optimization problem with ℓ_1-norm regularization. More details of the ℓ_1-norm regularization approach can be found in [11]. The approach of [10] is to formulate a commanded vehicle trajectory as the output of a linear system with input u. The constraints are that the trajectory must pass near a set of M points at specified time instants. The optimization is over the magnitude of impulses, defined on a time grid, for the p-th derivative of u. The ℓ_1-norm regularization yields a sparse vector of the p-th derivative of u.

This paper also follows an ℓ_1-norm regularization approach to proposes an optimization formulation of the roadway curve fitting problem in an unified framework that jointly computes the arclength, finds the curvature switching points, and fits the road segments to $\mathcal{D}$. In contrast to [10], the model is nonlinear, the independent variable is arclength instead of time, and the optimization variables are selected to achieve piecewise constant curvature. The basic concept is to develop a road kinematic model with curvature as a state variable, and then regulate the derivative of the curvature using ℓ_1-norm regularization. This approach is capable of rendering a sparse vector for the derivative of curvature.

This article is organized as follow. Section IV presents the vehicle kinematics. Based on the vehicle kinematics and a few assumptions, the road kinematic model is presented in the Section V. With the road kinematic model, the problem is formulated in the optimization framework in the Section VI. The discussion of roadway data acquisition is presented in the Section VII. The experimental results and data storage are discussed in the Sections VIII and IX. The article concludes in the Section X.

IV. Vehicle Kinematics

The kinematic model of a vehicle traveling without slip on a planar surface is

$$\dot{p}_x(t) = V(t)\cos\big(\psi(t)\big) \quad (1)$$

$$\dot{p}_y(t) = V(t)\sin\big(\psi(t)\big) \quad (2)$$

$$\dot{\psi}(t) = \omega(t) \quad (3)$$

where $[p_x, p_y]$ denotes position, ψ is the yaw angle of the vehicle relative to the x-axis, and the inputs to the kinematic model are the speed V and the angular rate ω.

V. Road Model

Given that the roadway is locally planar and intended to be traveled by vehicles, the kinematic model of eqns. (1-3) is the foundation of the roadway model. Since the road shape is not evolving versus time but as a function of arclength s, some modifications are necessary to derive the road kinematic model.

Noting that $\dot{p}_x(t) = dp_x(t)/dt$, $\dot{p}_y(t) = dp_y(t)/dt$, $\dot{\psi}(t) = d\psi(t)/dt$, and that $ds = V(t)dt$, eqns. (1-3) can be rewritten as

$$dp_x(t) = \cos(\psi(t))\ ds \quad (4)$$

$$dp_y(t) = \sin(\psi(t))\ ds \quad (5)$$

$$d\psi(t) = \kappa(t)\ ds \quad (6)$$

$$d\kappa(t) = \eta(t)\ ds \quad (7)$$

where $\kappa = \omega(t)/V(t)$ represents curvature and η is the derivative of the curvature κ with respect to arc-length s.

The equivalent discrete-time model by Euler integration is

$$p_x(k+1) = p_x(k) + \Delta s(k)\cos\psi(k) \quad (8)$$
$$p_y(k+1) = p_y(k) + \Delta s(k)\sin\psi(k) \quad (9)$$
$$\psi(k+1) = \psi(k) + \kappa(k)\Delta s(k) \quad (10)$$
$$s(k+1) = s(k) + \Delta s(k) \quad (11)$$
$$\kappa(k+1) = \kappa(k) + \beta(k) \quad (12)$$

where $s(k)$, $\kappa(k)$ are the arclength and the curvature of the k-th point on the curve, $\beta(k)$ is the increment of $\kappa(k)$ at point k and $\beta(k) = \eta\Delta s(k)$, $\Delta s(k) = V(k)T_s$ where T_s is the integration time step.

Eqns. (8-12) define the road kinematic model with the state vector:

$$x(k) = [p_x, p_y, \psi, s, \kappa]_{(k)}. \quad (13)$$

The model inputs are $\Delta s(k)$ and $\beta(k)$. An important point to notice here is that the curvature κ is exclusively controlled by β which allows the ℓ_1-norm regularization approach to yield trajectories with piecewise constant curvature. Note that this road kinematic model is accurate only when the change of $\psi(k)$, equivalently $\kappa(k)\Delta s(k)$, is sufficiently small. Physically, the separation between data points should be small relative to the roadway radius of curvature ($\Delta s(k) \ll \frac{1}{\kappa(k)}$).

VI. Problem Formulation

The road curve fitting problem can be formulated as a nonlinear constrained optimization with ℓ_1-norm regularization in the following way:

$$\mathbf{P}_0 : \min_{x(0),\beta(k),\Delta s(k)} \sum \|y(k) - W(k)\|_2^2 + \lambda \|\mathbf{b}\|_1 \quad (14)$$

subject to the road dynamic model constraint defined in eqns. (8-12) and

$$-\Delta s(k) \le 0,\ k = 0, ..., N-1 \quad (15)$$

where $y(k) = [p_x(k), p_y(k)]$ is the curve fit trajectory location at the k-th arc length according to eqns. (8-12), $\mathbf{b} = [\beta(0), ..., \beta(N-1)]$, $W(k) \in \mathcal{D} = \{N_k, E_k\}_{k=0}^{N}$ is the measured position along the road, and λ is a parameter that weights the tradeoff between trajectory smoothness and accuracy of the data fit. The inequality constraint (15) ensures that the increments of arc length are non-negative.

It is well known that ℓ_1-norm regularization is able to render sparse results [10], [11], in the sense that most of the elements of the vector $\mathbf{b}$ are zero. Because the non-zero elements in the sparse vector $\mathbf{b}$ represent the discontinuities in curvature $\kappa(k)$, the constraint of piecewise constant curvature can be enforced in the above formulation. Optimization problem $\mathbf{P}_0$ finds the best tradeoff between the road curve data fit and the number of changes in the curvature.

In order to solve the optimization problem $\mathbf{P}_0$, there is a standard trick to convert the ℓ_1-norm into a vector dot product by introducing additional inequality constraints. Thus, by defining the positive vector $\mathbf{z} = [z(0), ..., z(N-1)]^T$, the optimization problem (14) can be rephrased as

$$\min_{x(0),\beta(k),\Delta s(k)} \sum \|y(k) - W(k)\|_2^2 + \lambda \mathbf{1}^T \mathbf{z} \quad (16)$$

subject to the constraints in eqns. (8-12), (15) and

$$-z(k) \le \beta(k) \le z(k), k = 0, ..., N-1 \quad (17)$$

If a stacked vector is defined as:

$$\mathbf{X} = [x(0), ..., x(N), \mathbf{b}, \mathbf{z}, \mathbf{\Delta S}]$$

where $\mathbf{\Delta S} = [\Delta s(0), ..., \Delta s(N-1)]$, then the optimization problem can be further written into a more compact form:

$$\mathbf{P}_1 : \min_{\mathbf{X}} f(\mathbf{X})$$
$$s.t.\ q(\mathbf{X}) = 0$$
$$c(\mathbf{X}) \le 0$$

where $f(\mathbf{X})$ is the cost function in (16), $q(\mathbf{X})$ is the equality constraints in eqns. (8-12), and $c(\mathbf{X})$ is the inequality constraints in (15) and (17).

Now the optimization problem is in a standard form and thus can be solved by classical interior-point algorithms (such as "fmincon" in MATLAB®). To get a better result, we can solve the optimization problem iteratively [10]:

- Solve the optimization problem $\mathbf{P}_1$ using $\lambda = \lambda_1$ to get a sparse vector $\mathbf{b}$ ($\mathbf{b} = [\beta(0), ..., \beta(N-1)]$)
- Add additional constraints: $\beta(k) = 0$ for $k \in \mathcal{K} = \{k \,|\, |\beta(k)| \le \beta_0\}$ and choose a different $\lambda = \lambda_2$.
- Then solve the new optimization problem again.

Typically, the problem converges in two iterations. In this paper, β_0 is chosen to be 10^{-4}. Other more refined methods, such as weighted ℓ_1 minimization [12], can also be applied here to solve $\mathbf{P}_0$.

VII. Roadway Data

The following section demonstrates the algorithm. To attain the lane-level accuracy, a set of accurate roadway data is required. At the time the paper was written, a smoothing algorithm had not yet been implemented; therefore, the data that we use is the result of real-time vehicle state estimation. A test vehicle was instrumented with a Carrier Phase Differential GPS aided Inertial Navigation System (INS) [13]–[17] that is capable of achieving vehicle positioning accuracies better than 0.1m while the vehicle is maneuvering at highway speeds. The INS state vector is saved at a 30 Hz rate. Only the INS position estimates are used in the approach that follows. To acquire data for the examples portion of this paper, we drove the test vehicle along a roadway that was of interest.

The vehicle drove at a maximum speed of 60 miles/h and the sampling rate is 30Hz. Therefore, $\Delta s(k)$ is less that 0.81 meter which is small relative to the (minimum) radius of curvature ($1/\kappa$) experienced in the data set. Therefore, the road kinematic model of eqns. (8-12) is applicable. The data is obtained for a trajectory merging onto Highway 215 at Fair Isle Dr. in Riverside CA. This dataset contains 800 points (376 m in arc length).

This data acquisition methodology results in two sources of roadway position estimation error. The first source is the error in the INS estimate of the vehicle position. The second source is the inability of the driver to exactly follow the

roadway centerline. While the INS vehicle position error is small ($< 0.1m$) and uncorrelated for lags greater than a few seconds. The driver error is bounded only by one half the lane width, but may be biased and correlated over short periods of time. Nonetheless, the curve fit results show excellent performance and will only be further enhanced when the input data $\mathcal{D}$ is the result of a smoothed data fusion process.

Eventually it is expected [18], [19] that large volumes of vehicle position data will become available when volunteer drivers allow their GPS enabled cell phones to be used as probes. While this data source will introduce new challenges (e.g., detection of lane changes, detection of erroneous data, estimation of phone position relative to the vehicle center), it also highlights the fact that in the near future large volumes of lane data will become available and reliable methods for automatically extracting high-accuracy maps will be required. Existing and emerging roadway geometry data collection methods are discussed in [5].

VIII. Experimental Results

Trial 1 used $\lambda_1 = 150$ in the first iteration of the optimization and $\lambda_2 = 180$ in the second iteration of the optimization. The results are shown in Fig. 1. The upper left subplot shows the trajectory drawn on the tangent plane. The coordinates of the road have been shifted to be centered at zero. The red asterisks indicate the points where discrete changes in curvature occur (i.e., nodes, see Section IX). The travel direction is from left to right. The top right subplot shows the trajectory yaw angle which is continuous and piecewise linear. The blue curve in the right subplot in the second row shows the curvature κ, which clearly exhibits the piecewise constant property. For the comparison purpose, we also plot the curvature computed from the spline curve fit method [8] using cubic Hermite splines. For spline method 1 (green), we choose the spline knots evenly every 50 meters. For spline method 2 (red), we choose the same knots at the points of curvature discontinuity identified by the ℓ_1 optimization. It is clear that in both cases, the spline curvature is not piecewise constant. In addition the variation of the spline curvature is significantly larger than is necessary to fit the $\mathcal{D}$. Finally, the spline curvature has the incorrect sign at some locations.

The bottom right subplot shows the change in curvature, i.e. β, which is an impulse train as desired. The left subplot in the second row shows the curve fit error that is computed pointwise as

$$e(k) = \vec{\mathbf{N}}(k) \cdot (W(k) - y(k))$$

where $\vec{\mathbf{N}}(k) = [\sin\psi(k), -\cos\psi(k)]^T$ is the (right) normal vector to the fitted curve at the k-th point. Therefore, $e(k)$ is positive if $W(k)$ is on the right hand side of the fitted curve and negative if $W(k)$ is on the left hand side of the fitted curve. It is clear that the magnitude of the curve fit error is less than 5 cm at most locations and never larger than 10 cm. [1] In addition, it is clear that the road has two portions. The first portion of the road has high curvature while the second portion has lower curvature. The dividing line is marked in the subplot by the red dash vertical line that corresponds to the fourth curvature change. The mean square error (MSE) for each portion of the road is shown in row one of Table I. Table I also shows summary results of a second trial using $\lambda_1 = 20$ and $\lambda_2 = 80$. For both trials, the MSE is greater in the first portion of the road than the second portion. This is reasonable given that the driver has greater difficulty keeping the vehicle on the centerline of the road when the road has significant curvature. Table I also shows that the different choice of weights λ did not drastically change the accuracy nor the number of required nodes. The bottom left subplot displays the unbiased error auto-correlation function showing that the spatial correlation of the error drops below 4×10^{-4} in a few meters. This illustrates that the fit error, as mentioned in Section VII, is only correlated over a short time periods.

TABLE I
Optimization Performance

Test	MSE ($\times 10^{-3}$) $s \in [0, 115]$m	$s \in [115, 375]$m	$s \in [0, 375]$m	#Nodes
1:	1.42	0.31	0.76	11
2:	0.49	0.27	0.34	15

The algorithm optimally selects the number and the locations of curvature change points jointly to minimize the cost function defined in (16). Therefore, this algorithm eliminates the need of any pre-processing to search for curvature change. Moreover, there are only two parameters to tune (λ and β_0). Regarding computational complexity, the optimization algorithm is very efficient (done in less than 2 minutes) for dataset presented herein. However, due to the nonlinearity in eqns. (8-12), the problem is not a convex optimization problem. Hence, sometimes the optimization algorithm converges slowly. But by choosing a different λ the problem can usually be solved quickly.

IX. Data Storage and Realtime Usage

At the completion of the optimization, we obtain the optimal curve as a sequence of points $\mathcal{T} = \{x(k), \beta(k)\}_{k=0}^{N}$ (see eqn. (13)). Only the nodes ($\mathcal{N} = \{x(k+1) \mid \beta(k) \neq 0\}$) need to be stored as they are sufficient to exactly represent $\mathcal{T}$. To clarify the presentation, we use n to count over the nodes in $\mathcal{N}$. For node n, we denote its position as $P_n = \left[\ p_x(n), p_y(n)\ \right]^\top$.

Applications require a parameterized function $\gamma(s)$ that computes the trajectory $\mathcal{T}$ as a function of arclength $s \in [0, s_N]$. Using only the nodes $\mathcal{N}$, the procedure is as follows:

1) Given $s \in [0, s_N)$, find n such that $s \in [s_n, s_{n+1})$.

[1] The discrete changes in the curve fit error are caused by the realtime Kalman filter corrections at GPS measurement times GPS. These would be absent if smoothed data were available.

2) If $\kappa_n = 0$, then this segment is a line; therefore,

$$\gamma(s) = P_n + \mu T_n$$

where $\mu = \frac{s-s_n}{s_{n+1}-s_n}$ and $T_n = P_{n+1} - P_n$. The curvature is zero and the direction is ψ_n which should match the four quadrant arctangent of T.

3) If $\kappa \neq 0$, then the segment is a circle. The center of the circle can be computed by either

$$\begin{aligned} P_c &= P_n + R_n \vec{\mathbf{N}}_n \\ P_c &= P_{n+1} + R_n \vec{\mathbf{N}}_{n+1} \end{aligned}$$

where $R_n = \frac{1}{|\kappa_n|}$ is the radius of curvature and $\vec{\mathbf{N}}_n = -sign(\kappa_n)[\sin\psi_n, -\cos\psi_n]^T$ is the inward pointing unit normal. Both equations should yield exactly the same center location, which is useful as a check. The direction of the tangent to the circle at s is

$$\psi(s) = \psi_n + \kappa_n(s - s_n).$$

The location on the circle at arclength s is

$$\gamma(s) = P_c - R_n \vec{\mathbf{N}}(s)$$

where $\vec{\mathbf{N}}(s) = -sign(\kappa_n)[\sin\psi(s), -\cos\psi(s)]^T$.

The above computation procedure is straightforward and exactly represents $\mathcal{T}$. The resulting $\gamma(s)$ is available for realtime applications.

In comparison with traditional roadway representations that store only $[s_n, P_n]$ and use linear connections, the approach above requires storage of $[s_n, P_n, \psi_n, \kappa_n]$ which is two extra parameters. The tradeoff is that a more accurate roadway representation is obtained. In addition, it may turn out that with the improved representation, the nodes can be more widely spaced, allowing for the overall memory requirements per lane to stay constant.

X. Conclusions and Future Research

This article has discussed the problem of generating roadway trajectory curve fits that accurately match available data while enforcing roadway characteristics known from the design principles (i.e., piecewise constant curvature). The article includes experimental results and discussion of the straightforward analytic trajectory representation in terms of nodes.

This work could be extended in several different directions. First, ultimately, the curve fit procedure must work in three dimensions. A challenge in that extension is that the vertical and lateral curvature changes are independent. Second, the INS or smoothing process maintains an estimate of the position error covariance matrix at each time instant. This error covariance matrix could be incorporated as a time varying weight into the trajectory optimization procedure. This article has only used the measured position data in the cost function. The INS or smoothed trajectory also has other information (e.g., velocity, attitude, angular rate) that might be useful.

XI. Acknowledgments

This material is based upon work supported by the California Department of Transportation under Contract No. 65A0311 and the DOT Federal Highway Administration (FHA) Agency Award No: DTFH61-09-C-00018. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the sponsors.

References

[1] S. Mortensen, "USDOT's demonstration and deployment programs on vehicle assist and automation for bus rapid transit," in *Intelligent Transportation Systems, 2009. Proceedings*, (St. Louis, MO), 2009.

[2] I. Skog and P. Händel, "In-car positioning and navigation technologies - a survey," *IEEE Transaction on Intelligent Transportation Systems*, vol. 10, pp. 4–21, March 2009.

[3] CAMP, "Enhanced digital mapping project final report," tech. rep., United States Department of Transportation, Washington, DC, 2004.

[4] K. Li, H.-S. Tan, J. A. Misener, and J. K. Hedrick, "Digital map as a virtual sensor-dynamic road curve reconstruction for a curve speed assistant," *Vehicle System Dynamics*, vol. 46, pp. 1141–1158, 2008.

[5] B. Spear, A. Vandervalk, and D. Snyder, "Roadway geometry and inventory trade study for intellidrivesm applications applications," tech. rep., FHWA, 2010.

[6] K. Kim, D. Yum, D. Byeon, D. Kim, and D. Lee, "Improving driver's visual field using estimation of curvature," in *2010 International Conference on Control Automation and Systems (ICCAS)*, pp. 728–731, IEEE, 2010.

[7] F. Jiménez, F. Aparicio, and G. Estrada, "Measurement uncertainty determination and curve-fitting algorithms for development of accurate digital maps for advanced driver assistance systems," *Transportation Research Part C: Emerging Technologies*, vol. 17, no. 3, pp. 225–239, 2009.

[8] A. Chen, A. Ramanandan, and J. Farrell, "High-precision lane-level road map building for vehicle navigation," in *Position Location and Navigation Symposium (PLANS), 2010 IEEE/ION*, pp. 1035–1042, IEEE, 2010.

[9] K. G. Baass, "The use of clothoid templates in highway design," *Transportation Forum 1*, pp. 47–52, 1984.

[10] H. Ohlsson, F. Gustafsson, L. Ljung, and S. Boyd, "Trajectory Generation Using Sum-of-Norms Regularization," *49th IEEE Conference on Decision and Control*, 2010.

[11] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge Univ Pr, 2004.

[12] E. Candes, M. Wakin, and S. Boyd, "Enhancing sparsity by reweighted ℓ_1 minimization," *Journal of Fourier Analysis and Applications*, vol. 14, no. 5, pp. 877–905, 2008.

[13] J. A. Farrell, T. Givargis, and M. Barth, "Real-time differential carrier phase GPS-aided INS," *IEEE Transactions on Control Systems Technology*, vol. 8, no. 4, pp. 709–721, 2000.

[14] J. A. Farrell, H. Tan, and Y. Yang, "Carrier phase GPS-aided INS based vehicle lateral control," *ASME Journal of Dynamics Systems, Measurement, & Control*, vol. 125, no. 3, pp. 339–353, 2003.

[15] Y. Yang and J. A. Farrell, "Magnetometer and differential carrier phase GPS aided INS for advanced vehicle control," *IEEE Trans. Robotics and Automation*, vol. 19, no. 2, pp. 269–282, 2003.

[16] Y. Yang and J. A. Farrell, "Two antenna GPS aided INS for attitude determination," *IEEE Trans. on Control Systems Technology*, vol. 11, no. 6, pp. 905–918, 2003.

[17] J. A. Farrell, *Aided Navigation Systems: GPS and High Rate Sensors*. New York: McGraw-Hill, 2008.

[18] S. Schroedl, K. Wagstaff, S. Rogers, P. Langley, and C. Wilson, "Mining GPS traces for map refinement," *Data Mining and Knowledge Discovery*, vol. 4, pp. 127–139, June 2004.

[19] G. Zhang and C. Wilson, "An integrated DGPS/DR/map system for vehicle safety applications," in *Proceedings of the 2000 National Technical Meeting of the Institute of Navigation*, pp. 253–257, January 2000.

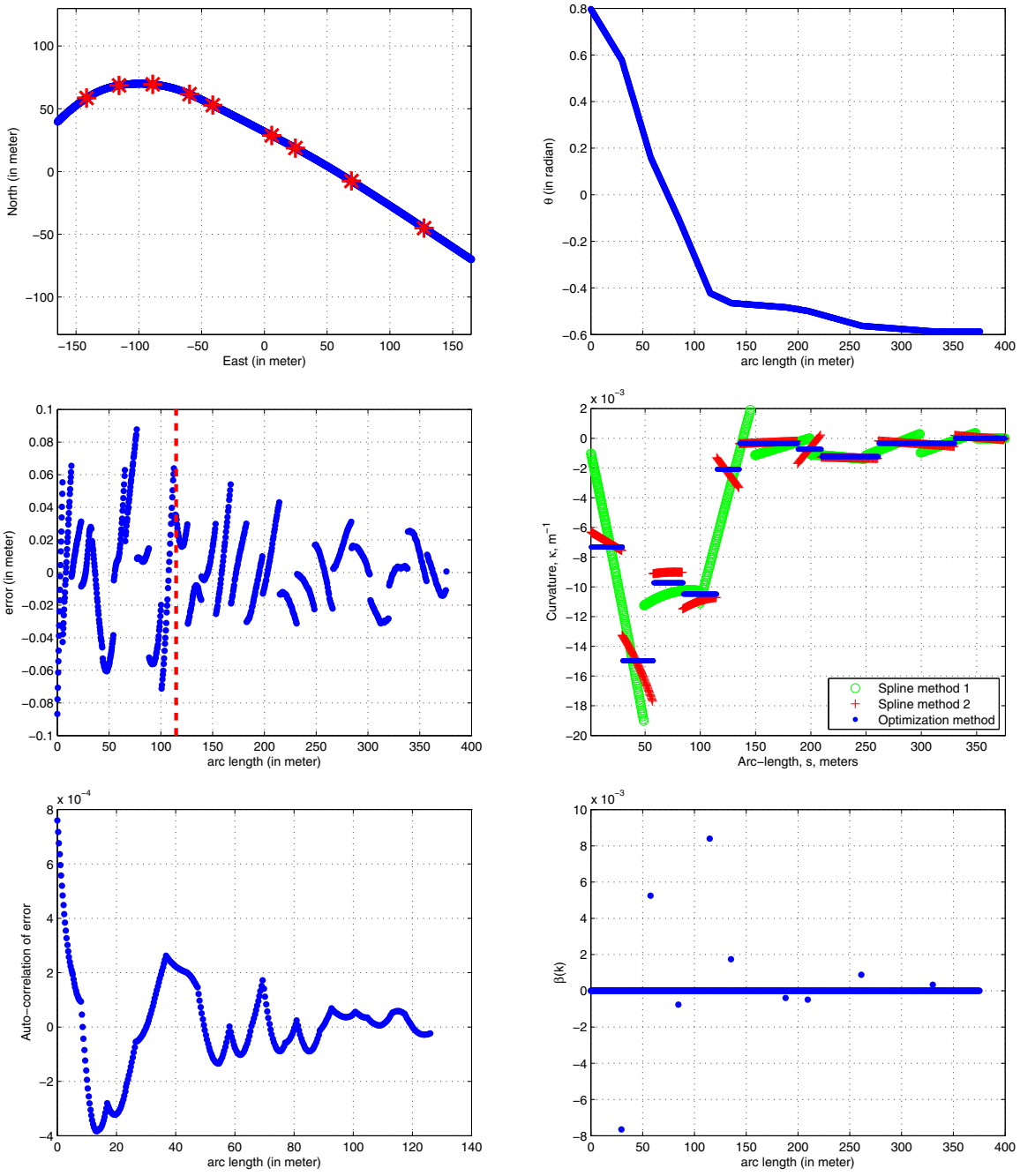

Fig. 1. Test 1 results. Upper left – Tangent plane trajectory position. Second row left – Error vs arclength. Bottom left – Autocorrelation of error vs arclength. Upper right – Trajectory yaw angle vs arclength. Second row right – Curvature κ computed from 3 different methods vs arclength. Bottom right – Change in curvature vs arclength.