2024 Tenth Indian Control Conference (ICC)
9 - 11 December, 2024
IISER Bhopal, India

Self-calibrating Offline Trajectory Estimation Technique for Sporadically Observable Systems

Somnath Buriuly, *IIT Bombay* Leena Vachhani, *IIT Bombay*

***Abstract*— Industrial problems, like pipeline inspection and railway track geometry inspection inside tunnels, require accurate offline estimation of the state trajectories, to determine the infrastructure parameters, say pipeline GPS coordinates, rail-track's attitude and placement, etc. These problems are cursed with loss of observability due to the unavailability of localization data, like GPS, inside pipelines and tunnels. In addition, the states at the start and end of the recorded data are observable but not accurately or entirely known. Furthermore, a deviation is observed in the initial calibration of the Inertial Measurement Unit (IMU) sensors, and hence the recorded data are approximated sensor readings. Since the accuracy of these estimated outcomes impacts the maintenance cost of the infrastructure, it is essential to obtain the best possible estimate from the offline available sensor data.**

In this work, we propose an optimization-based solution to estimate the entire trajectory and self-calibrate the sensor readings, simultaneously. In particular, we formulate an offline state trajectory estimation problem with self-calibration features for the sensor readings, compute the gradients for the decisions, and propose a Lagrangian-based gradient descent algorithm. We compare the results with the state-of-the-art Extended Kalman Filter and show improvement in the mean square error values of the estimates.

***Index Terms*—Trajectory estimation, Extended Kalman Filter, Sporadically Observable, Trajectory Optimization, Strapdown Inertial Navigation Systems, Infrastructure inspection**

I. Introduction

Estimation is a core problem of control theory. It has huge practical significance in research and industry alike. The development of estimation techniques is essential because, in many practical problems, direct measurement of the system's state variables is either too expensive or impossible. Real-time state estimation techniques are useful in implementing control algorithms on most practical robotics applications. However, in the pipeline and railway infrastructure inspection problems, there is no element of control, making offline trajectory estimation a viable choice. In addition, the GPS signal is unreachable inside metal or underground structures like pipelines. Tunnels are also shielded by the earth's mass, resulting in loss of GPS signal. In particular, the major chunk of the dataset is only partially observable, while attaining full observability sporadically. In a nutshell, this work investigates the offline trajectory estimation problem with sporadic localization data.

Numerous research articles [1], [2], [3], [4], [5] and patents [6], [7], [8], [9], [10], [11] have been communicated for the estimation of the state trajectory of intelligent Pipeline Inspection Gauge (iPIG) devices. In these in-line inspection problems, Inertial Measurement Unit (IMU), odometers, and sporadic/ occasionally available ground marker (GPS) data are collected by the iPIG device moving through the pipeline [1]. An estimation tool is then used to associate the GPS coordinates to the entire dataset. This problem is readily studied in the pipeline industry, and it is still being investigated to improve accuracy, reduce instrumentation and operation costs, and explore innovative solutions. Similarly, track geometry inspection is essentially an online estimation problem to determine the track parameters like inclination, track width, pitch, curvature, etc. [12], [13]. The estimation problem is identical to the pipeline surveying using iPIG, especially in tunnels, due to the loss of localization data. In brief, the trajectory estimation problem essentially builds the baseline map of the system states which is useful in determining the infrastructure parameters like GPS coordinates of pipeline bends, geometry of tracks, etc. The offline approach for building a better map of the infrastructure requires estimation of the full state trajectory i.e. offline localization. Pipeline localization and track geometry problems are essentially offline-SLAM (**S**imultaneous **L**ocalization **A**nd **M**apping) where localization and mapping is the same problem of tagging the data with GPS coordinates [14]. In this work, we develop an optimization framework for offline trajectory estimation, whose application extends beyond the pipeline and railway industry.

Offline trajectory estimation problem with sporadic observability is cursed by many practical challenges. The localization information is only available at a few select time instances. In addition, the algorithm must be flexible to account for self-calibration because practical systems have known dynamical structures with uncertain parameters, e.g. sensor gains, actuator gains, etc. Last but not least, in offline methods, the algorithm must utilize the past and future trajectory to provide the best estimate of the current state. The proposed optimization framework addresses the above challenges through an iterative framework that simultaneously achieves self-calibration.

Kalman Filter (KF) is a very well-established tool for estimation of the unmeasured and the measured-but-noisy states. It provides an online solution to estimate the current state of any observable system. A common approach for trajectory estimation is to directly use variants of KF for estimating the state trajectories. This is observed in the pipeline inspection literature survey and review articles [1], [14]. In particular, Extended Kalman Filter (EKF) variants [2], [3], [4] are developed for the Strapdown Inertial Navigation System (SINS) [15] and 3D-Reduced Inertial Sensor System (3D-RISS) dynamical

Somnath Buriuly is a post-doctoral fellow with the Center of Excellence in Oil Gas and Energy, IIT Bombay, India. He received his doctoral degree from IITB-Monash Research Academy; e-mail: somnath.buriuly@monash.edu.

Leena Vachhani is a Professor in Systems and Control Engineering, IIT Bombay, India; e-mail: leena.vachhani@iitb.ac.in.

Manuscript submitted on June 30, 2024.

979-8-3315-1721-2/24/$31.00 ©2024 IEEE

models [16] of the iPIG. We observed that EKF being an online recursive approach, the update step is unable to utilize future data to correct the current and past states. Rauch-Tung-Stribel-Smoothing (RTSS) approach provides one solution to update the states backward in time, however, that only corrects the trajectory once based on the temporal neighbors [5]. In brief, the KF variants are well suited for real-time state estimation solutions but they suffer from re-evaluating the past states. In addition, although the KF methods balance the motion dynamics and measurement data based on noise co-variances, they are not designed to adjust the motion model to improve performance.

Trajectory optimization (TO) offers an iterative solution to adjust the system and/or the trajectory. An observability-aware TO framework has been investigated in the literature to select the waypoints that improve the observability of the system for self-calibration, which in turn reduces the estimation error in EKF by 50% [17]. Observability-aware framework for appropriately selecting the control inputs for self-calibration of system parameters to improve the estimation quality has also been investigated [18], [19]. To the best of our knowledge, the TO framework is not utilized for offline calibration and trajectory estimation i.e. in problems where the guidance decisions and control inputs are not adjustable.

In this work, we propose an optimization framework for the offline trajectory estimation problem with sporadic observability and develop a Lagrangian-based gradient descent algorithm. Aside from the iterative estimation of the states, the framework allows self-calibration or correction of a few of the system parameters (e.g. sensor gain, etc.) to improve the modeling errors. The approach provides a novel solution to estimate the state trajectories while accounting for model errors, and parameter calibration.

The main contributions of this work are:

- An online optimization framework for offline state trajectory estimation and simultaneous self-calibration/ parameter update.
- Gradient computations for updating the setup and system parameters. A gradient-based state-co-state algorithm to solve the trajectory estimation problem.

The structure of the work is as follows. A couple of dynamical systems are reviewed and reformulated, in a comprehensive manner, in Section II, for brevity. The proposed approach is presented in Section III, alongside an algorithm to solve the offline trajectory estimation problem. The proposed approach is compared with the EKF approach in Section IV. Lastly, the work is concluded with a glimpse at the scope and the future prospects.

II. Preliminaries

In this work, we perform a simulation study on data collected from the kinematic model of a unicycle robot with un-calibrated wheel velocities. We also demonstrate the estimation approach on the Strapdown Inertial Navigation Systems (SINS) model from literature. For brevity, we revisit the unicycle model and the SINS model in this section.

A. Unicycle kinematic model with wheel velocity input

The states of a unicycle kinematic model are the xy-positions and orientation in the xy-plane. In theory, it is controlled by the forward and angular velocities. However, in most ground robot applications, the actuation is a differential drive that achieves angular velocity by the unbalanced left u_l and right u_y motor velocities. The usual sensors mounted on these robots are IMU for accelerometer and gyroscope measurements, GPS for localization data, and wheeled encoders for odometer distance/velocity reading. P1 presents the motion model with the wheel velocity data as the control input.

$$\begin{bmatrix}\dot{p}_x\\ \dot{p}_y\\ \dot{\phi}\end{bmatrix} = \begin{bmatrix}\cos\phi & 0\\ \sin\phi & 0\\ 0 & 1\end{bmatrix}\begin{bmatrix}0.5 & 0.5\\ \frac{1}{l} & -\frac{1}{l}\end{bmatrix}\begin{bmatrix}u_r\\ u_l\end{bmatrix} \tag{P1-a}$$

$$\begin{bmatrix}\bar{y}_1\\ \bar{y}_2\end{bmatrix} = \begin{bmatrix}p_x\\ 0\end{bmatrix},\ \ or\ \ \begin{bmatrix}\bar{y}_1\\ \bar{y}_2\end{bmatrix} = \begin{bmatrix}p_x\\ p_y\end{bmatrix} \tag{P1-b}$$

In Equation (P1-a), the parameter l is the separation between the wheels. The value of l is assumed to be 0.15 in this work. For validation purposes, we assume that the measurements are observed occasionally, while for the rest of the time instances, we have partially observable measurements, as shown in Equation (P1-b). The measurement $\bar{y}_2$ is considered zero whenever unavailable, to achieve dimensional consistency of the cost weights, measurement Jacobian matrix, etc. Also, we have put a bar notation on variables to indicate that these are measurements/recorded data.

B. Strapdown Inertial Navigation System (planar)

The Strapdown Inertial Navigation System models the dynamics of an Inertial Measurement Unit (IMU) in a rotating Earth-fixed reference frame, where the inertial measurements are made with respect to the Earth-centered frame [15]. Hence, the SINS model is used for predicting the states of rigid bodies moving in space using acceleration and angular velocity data measured by the IMU. The system of equations in P2 presents a simplified 2D motion model for a fixed reference frame i.e. not considering the earth's rotation and geometry.

$$\dot{s} = v_f \tag{P2-a}$$

$$\dot{p} = R_z(\phi_z)\, v_f \tag{P2-b}$$

$$\dot{\phi}_z = \kappa^g \omega_z + b^g \tag{P2-c}$$

$$\dot{v}_f = \kappa^a a_x + b^a \tag{P2-d}$$

The states of the SINS model are distance s, 2D position vector p, z-direction orientation as ϕ_z, and forward velocity v_f. The inputs to the motion model are forward acceleration along the x-direction obtained from accelerometer measurements, and z-direction angular velocity obtained from gyroscope measurements, denoted as a_x and ω_z respectively. The parameters κ^a and κ^g are accelerometer and gyroscope gains, whereas the b^a and b^g are respective biases. The function $R(\phi_z)$ is the rotation matrix to transform the body frame velocity measurements into the fixed frame. The distance state s is recorded by the odometers at every sampling time instant, while the localization states p are only available sporadically.

III. METHODOLOGY

This section begins with a formal introduction to the trajectory estimation problem. Next, an optimization framework is proposed, highlighting the cost function and constraints. Lastly, an algorithm is presented to solve the proposed optimization framework.

A. The trajectory estimation problem statement

In real-world offline trajectory estimation problems, the sensor readings are provided from the actual run of the system - e.g. iPIG through a pipeline, instrumented wagon on railway tracks, etc. These might include actuation signals, IMU data, GPS data (if available), etc. This problem assumes the following information is known: A motion model, denoted as $f(x_k, \bar{u}_k; \Theta)$ is available from first principles. The states x_k are unknown, however the function input $\bar{u}_k$ and the measurements $\bar{y}_k$ are available as a time-stamped dataset. The measurement function $h_k(x_k)$ is partially observable except at a few time instants, especially during the initial and the final measurements ($h_0(x_0)$ and $h_N(x_N)$), it is fully observable. Additionally, a few parameters of the motion model, denoted by vector Θ, are not calibrated i.e. not known accurately. The objective is to estimate the state trajectories from the provided data while mitigating the parametric uncertainties ingrained into the problem i.e. achieve self-calibration.

B. The optimization framework

The proposed optimization framework is presented in P3. The cost function penalizes the mismatch in the measurements, while the constraints enforce the dynamics.

$$J(\{\bar{u}_k\}_{k=0}^{N}, \{\bar{y}_k\}_{k=0}^{N}) :=$$

$$\min_{\Theta, \{x_k\}_0^N} \ \frac{1}{2}\sum_{k=0}^{N}(h(x_k)-\bar{y}_k)^\top R(h(x_k)-\bar{y}_k) \qquad \text{(P3-a)}$$

$$\textit{s.t.} \quad x_{k+1} = f(x_k, \bar{u}_k; \Theta) \qquad \text{(P3-b)}$$

$$\Theta \in \mathbb{R}^\gamma, \{x_k\}_0^N \in \mathbb{R}^n \qquad \text{(P3-c)}$$

The R matrix in the cost function, see Expression P3-a, is the measurement noise covariance matrix. The vector Θ denotes γ non-calibrated parameters of the motion model, e.g. sensor gain. The optimization framework in P3 manipulates the parameters Θ, and the initial state x_0, to produce an estimate that best matches the measured data. In practice, a decent initial guess of the parameters and initial state is generally available. Also, note that both $\bar{u}_k$ and $\bar{y}_k$ are recorded data, hence bar notation.

C. Algorithm

The proposed optimization problem is constrained by the dynamics of the motion model. One option is to solve the problem using a standard Non-linear Programming (NLP) solver. In this work, we compute the gradients and utilize a gradient-descent algorithm. First, we formulate the problem as an unconstrained min-max problem using the Lagrangian approach. Next, we compute the gradients for the uncalibrated parameters and initial state. These gradients are incorporated into a suitable algorithm.

1) Lagrangian approach and the necessary conditions: The formulation P3 is a constrained optimization problem with quadratic cost and non-linear dynamical constraints. Applying the Lagrangian approach on P3 results in an unconstrained min-max problem given in P4; where the Lagrangian $\mathcal{L}$ is shown in Equation (5). The matrix R is symmetric and positive definite, representing the noise covariance.

$$\max_{\{\lambda_k\}_{k=0}^{N-1}} \ \min_{\Theta, \{x_k\}_{k=0}^{N}} \ \mathcal{L}(\{x_k\}_{k=0}^{N}, \{\lambda_k\}_{k=0}^{N-1}; \Theta) \qquad \text{(P4-a)}$$

$$\textit{s..t.} \qquad \Theta \in \mathbb{R}^\gamma, \{x_k\}_{k=0}^{N}, \{\lambda_k\}_{k=0}^{N-1} \in \mathbb{R}^n \qquad \text{(P4-b)}$$

$$\begin{aligned}\mathcal{L}(\{x_k\}_{k=0}^{N}, \{\lambda_k\}_{k=0}^{N-1};\ \Theta) :=& \\ &\frac{1}{2}(h(x_N)-\bar{y}_N)^\top R(h(x_N)-\bar{y}_N) \\ &+ \sum_{k=0}^{N-1}\Big\{\ \frac{1}{2}(h(x_k)-\bar{y}_k)^\top R(h(x_k)-\bar{y}_k) \\ &- \lambda_k^\top\left(x_{k+1} - f(x_k, \bar{u}_k; \Theta)\right)\ \Big\}\end{aligned} \qquad (5)$$

The Lagrangian variables λ_k are unconstrained because the dynamics enforce equality constraints. The optimal solution to the above min-max problem is a saddle point with zero gradients, giving the necessary conditions[1] in NC.

$$\frac{\partial \mathcal{L}}{\partial \lambda_k} = x_{k+1} - f(x_k, \bar{u}_k; \Theta) = 0 \qquad \text{(NC-a)}$$

$$\frac{\partial \mathcal{L}}{\partial x_k} = (h(x_k)-\bar{y}_k)^\top R\frac{\partial h}{\partial x_k} + \lambda_k^\top\frac{\partial f}{\partial x_k} - \lambda_{k-1}^\top = 0 \qquad \text{(NC-b)}$$

$$\frac{\partial \mathcal{L}}{\partial \Theta} = \sum_{k=0}^{N-1}\lambda_k^\top \left.\frac{\partial f}{\partial \Theta}\right|_k = 0 \qquad \text{(NC-c)}$$

The necessary conditions NC are valid for $k = \{1, \ldots, N-1\}$. For $k = N$, we have $\frac{\partial \mathcal{L}}{\partial x_N} = (h(x_N)-\bar{y}_N)^\top R\frac{\partial h}{\partial x_N} - \lambda_{N-1}^\top = 0$, implying $\lambda_{N-1} = \frac{\partial h}{\partial x_N}^\top R(h(x_N)-\bar{y}_N)$. Similarly, for $k = 0$, we have $\frac{\partial \mathcal{L}}{\partial x_0} = (h(x_0)-\bar{y}_0)^\top R\frac{\partial h}{\partial x_0} + \lambda_0^\top\frac{\partial f}{\partial x_0}$.

The forward dynamics propagating from the initial state x_0 ensures that the necessary condition in Equation (NC-a) is met. Similarly, the backward propagating co-state dynamics, shown in Equation (7), achieved from Equation (NC-b) ensures zero gradients at the solution. Lastly, the condition in Equation (NC-c) requires all co-state variables λ_k to be in the null space of the Jacobian with respect to parameters Θ.

$$\lambda_{k-1} = -\frac{\partial f}{\partial x_k}^\top \lambda_k + \frac{\partial h}{\partial x_k}^\top R(h(x_k)-\bar{y}_k) \qquad (7)$$

2) The proposed self-calibration algorithm: In this work, we observe that the optimization problem P4 is a two-point boundary value problem with unknown parameters. Hence, we propose the following algorithm:

S0: Set an initial guess for Θ and x_0.

[1]By default, $\frac{\partial}{\partial x_k}$ and $\frac{\partial}{\partial \lambda_k}$ are evaluated at x_k, $\bar{u_k}$, & λ_k. For $\frac{\partial}{\partial \Theta}$, we denote by $|_k$ that the partial derivative is evaluated at data from k^{th} instant.

S1: **Start loop:** Starting at x_0 evaluate the forward dynamics, given in Equation (P3-b), to achieve $\{h(x_k)\}_{k=0}^{N}$.

S2: Starting at $\lambda_{N-1} = \frac{\partial h}{\partial x_N}^\top R(h(x_N) - \bar{y}_N)$ evaluate all the co-states $\{\lambda_k\}_{k=0}^{N-1}$, backward in time, using the co-state dynamics in Equation (7), and the measurement values $\{h(x_k)\}_{k=0}^{N}$.

S3: Update the initial guess: $\Theta \leftarrow \Theta - \alpha \frac{\partial \mathcal{L}}{\partial \Theta}$ and $x_0 \leftarrow x_0 - \alpha \frac{\partial \mathcal{L}}{\partial x_0}$; where, α is the learning rate, $\frac{\partial \mathcal{L}}{\partial \Theta} = \sum_{k=0}^{N-1} \lambda_k^\top \frac{\partial f}{\partial \Theta}\big|_k$ is gradient from Equation (NC-c), and $\frac{\partial \mathcal{L}}{\partial x_0} = (h(x_0) - \bar{y}_0)^\top R \frac{\partial h}{\partial x_0} - \lambda_0^\top \frac{\partial f}{\partial x_0}$ from Equation (NC-b).

S4: Check for breaking condition. If not satisfied, go to step **S1**.

The algorithm is motivated by the indirect approach for solving a trajectory optimization problem [20], [21] and primal-dual column generation algorithm [22], with added computations for gradient-based parameter calibration.

IV. Results and discussion

In this work, we compare the state trajectory plots over various systems. The input readings are generated by the cumulative sum of $N \times |u|$ random data sampled from a uniform distribution in the range $[-1, 1]$, while the output readings are generated by forward propagating/evolving the error-free system dynamics. We also compare the results by varying the error in the guessed value for the initial state and the sensor gain (parameter). The improvement in the guess value highlights the effectiveness of the proposed approach. The following subsections showcase results for a linear double integrator system and a non-linear unicycle system; the Mean Square Error (MSE) values, computed using the state trajectories, are summarized in Table I. All trajectory plots are shown in Figure 1. We also show the results for the Strapdown Inertial Navigation System (SINS), due to its application in pipeline and railway-track trajectory estimation problems. In all the simulation results presented in this section, the temporal size of the dataset is 31, sampled at 10 Hz, where the measurements are fully observable at instances $0, 15, \& 30$ i.e. at the start, middle, and end.

A. Linear double integrator system

For the linear double integrator system P5, we compare the EKF with the proposed approach, with and without self-calibration of parameters k and b. In P5, position p and velocity v are the states, while u is the acceleration input. In this study, we assume that only the initial states, gain, and bias parameters are not known correctly. The data is collected with values of $x_0 = [2, 1]^\top$, $k = 1.5$ and $b = -0.1$, while the estimators assume with guess values as $x_{0,guess} = [2, -1]^\top$, $k_{guess} = 2.5$ and $b_{guess} = 0$. Observe that, for the initial guess, we assume a disparity in the velocity value of the initial guess.

$$\begin{bmatrix} \dot{p} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0.1 \end{bmatrix} \begin{bmatrix} p \\ v \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} (ku + b) \tag{P5-a}$$

$$\begin{bmatrix} \bar{y}_1 \\ \bar{y}_2 \end{bmatrix} = \begin{bmatrix} v \\ 0 \end{bmatrix}, \quad or \quad \begin{bmatrix} \bar{y}_1 \\ \bar{y}_2 \end{bmatrix} = \begin{bmatrix} v \\ p \end{bmatrix} \tag{P5-b}$$

B. Non-linear unicycle kinematic model

For the non-linear unicycle kinematic model P1, we again compare the EKF with the proposed approaches. In Equation 9, the wheel velocity control inputs u_r and u_l are achieved from the encoder-based velocity measurement data v_r and v_l. Again, in this study, we assume that only the initial states, gain, and bias parameters are not known correctly. The data is collected with the initial condition as $x_0 = [2, 1, 0.1]^\top$. The true value of parameters are $k^r = 1.5$, $k^l = 1.2$, $b^r = -0.1$ and $b^l = -0.05$ in Equation (9). The guess values for the estimations are $x_{0,guess} = [2, 1, 0.5]^\top$, $k^r_{guess} = 1.3$, $k^l_{guess} = 1.1$, $b^r_{guess} = 0$, and $b^l_{guess} = 0$. Here, we assume that the position data of the initial guess is correctly known. The initial heading data, however, is erroneous.

$$\begin{bmatrix} u_r \\ u_l \end{bmatrix} = \begin{bmatrix} k^r & 0 \\ 0 & k^l \end{bmatrix} \begin{bmatrix} v_r \\ v_l \end{bmatrix} + \begin{bmatrix} b^r \\ b^l \end{bmatrix} \tag{9}$$

C. Strapdown Inertial Navigation System

The comparison results are also evaluated for the 2D-SINS motion model. Table I and Figures 1c, 1f, 1i, and 1l demonstrate the reduction in the MSE values for the state trajectories in the proposed method. In this study, we assume that only the initial heading direction is unknown, while the initial distance and initial position states are known. The data is collected with the initial state as $x_0 = [0, 2, 1, -0.1, 0]^\top$. The true value of parameters are $k^a = 1.5, k^g = 1.2, b^a = -0.1$ and $b^g = -0.05$ in the Formulation (P2). The guess values for the estimations are $x_{0,guess} = [0, 2, 1, 0.05, 0]^\top$, $k^a_{guess} = 1.4$, $k^g_{guess} = 1.3$, $b^a_{guess} = -0.15$, and $b^g_{guess} = -0.1$. Note that, since the heading is not measured, there is a disparity in the initial guess state with respect to the actual initial value.

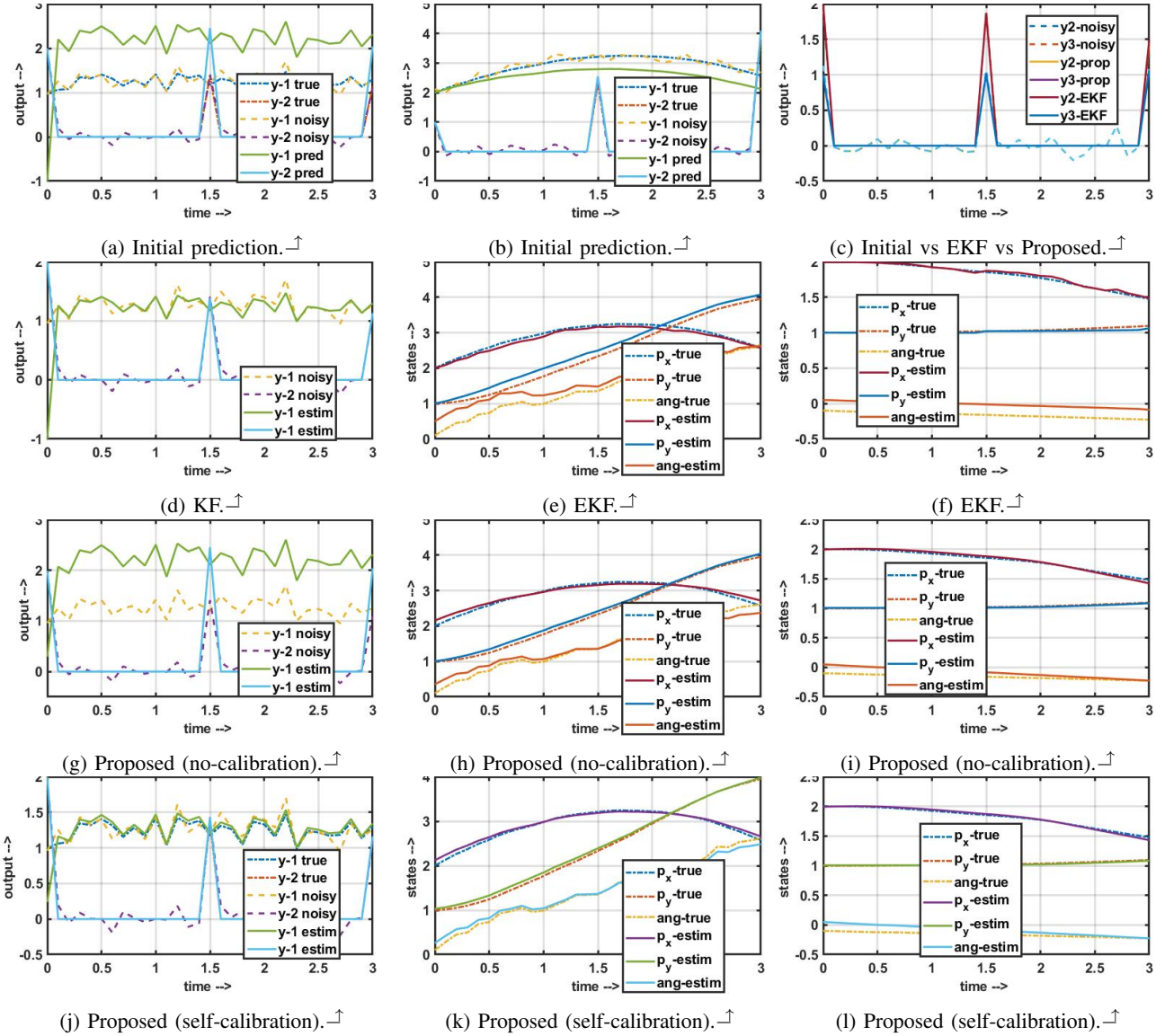
Discussion: The first two sub-figures of the first row are open loop predictions with initial state and parameter error for double integrator systems and unicycle model respectively, see Figures 1a and 1b. For the double integrator system, the initial velocity guess ($\bar{y}_1(0)$) and the sensor parameters are erroneous, therefore a large deviation is observed in the predicted trajectory. For the unicycle model, only the initial heading state and the sensor parameters are incorrect, hence the deviation is observed eventually. For the 2D-SINS in Figure 1c, we overlay the position output for initial vs proposed vs EKF, so that we can show state plots in other sub-figures. Note that some of the measurements are sporadic i.e. only available at specific time instances, observed as spikes in the trajectory. This change in the measurement model results in sporadic observability.

The second row of sub-figures, see Figure 1d, 1e, and 1f, are robust KF/EKF responses that attenuate the error eventually, however, they have larger error initially, which is not immediately entirely corrected due to partial observability for most time instances. For the double integrator system, we show the estimated measurement trajectories. For the unicycle model and the 2D-SINS, we show the partially measured position states and the unmeasured heading states to highlight the inability to correct the states due to partial observability. In particular, the system is sporadically observable, hence, recurring algorithms like KF/EKF are unable to update the unobserved states.

The third row of sub-figures, see Figures 1g, 1k, and 1l, are proposed optimization approaches without updates for parameter guess. They attenuate the initial state error, however, they

TABLE I: Comparison results. The smallest values of each row is highlighted.

System	Criterion	Open loop	EKF	Proposed	
				X0-only	Self-calibration
Double integrator system	**Initial state error**	2.00e+0	2.00e+0	**2.52e-1**	2.90e-1
	Parameter error	5.05e-1	5.05e-1	5.05e-1	**1.50e-2**
	MSE	1.15e+0	1.70e-2	9.03e-1	**4.31e-4**
Unicycle kinematic model	**Initial state error**	5.33e-2	5.33e-2	2.96e-2	**1.38e-2**
	Parameter error	1.56e-2	1.56e-2	1.56e-2	**1.53e-2**
	MSE	7.83e-3	1.15e-3	1.52e-4	**1.86e-5**
SINS (2D)	**Initial state error**	**4.50e-3**	4.50e-3	5.16e-3	4.91e-3
	Parameter error	6.25e-3	6.25e-3	6.25e-3	**6.09e-3**
	MSE	5.67e-5	9.65e-5	1.29e-5	**1.21e-5**



(a) Initial prediction. (b) Initial prediction. (c) Initial vs EKF vs Proposed.

(d) KF. (e) EKF. (f) EKF.

(g) Proposed (no-calibration). (h) Proposed (no-calibration). (i) Proposed (no-calibration).

(j) Proposed (self-calibration). (k) Proposed (self-calibration). (l) Proposed (self-calibration).

Fig. 1: Comparison plots for EKF and TO-based proposed approach. The sub-figures of the first column are trajectories obtained for a double integrator system; the sub-figures of the second column are trajectories obtained for a unicycle system; and sub-figures of the third column are trajectories for a SINS model.

have large errors due to parameter mismatch for the double integrator system. For the unicycle model and the 2D-SINS

model, the parameter deviation is small and the estimation is comparable to EKF.

Lastly, the fourth row of sub-figures, see Figures 1j, 1k, and 1l have parameter and initial state correction enabled. The estimated trajectories from this approach have the lowest MSE values among all other methods.

A quantitative overview is observed in Table I, where the overall MSE value is the smallest for the proposed approach. In addition, for two of the systems, the error in the initial state and parameter values are much smaller. For 2D-SINS, the MSE is better even if the parameters and initial states were not accurately identified due to the noise. Furthermore, a comparison of the two variants of the proposed approach shows that parameter update/ self-calibration is useful for the optimization framework to produce acceptable outcomes.

Conclusion

In this work, we investigated a novel approach to address the state trajectory estimation problem with sporadic observability. We formulated the problem in an optimization framework and developed a suitable gradient descent algorithm, that utilizes the co-state dynamics. The state-co-state approach ensures that the number of decision variables in the framework is small. The proposed work was able to reduce the error in the initial part of the trajectory (past) which the state-of-the-art KF/EKF methods don't address. In comparison with the EKF approach, the results demonstrated smaller mean squared error values, especially in the initial part of the trajectory, and demonstrated self-calibration capability.

The proposed optimization framework lays the groundwork for addressing trajectory estimation with real-world GPS data, having uncertainty in the model and parameters. For instance, the framework may be extended to investigate the 3D-SINS and 3D-RISS models available in the literature. Furthermore since the Lagrangian approach is sensitive to acceleration factor in general, especially for large data, hybrid algorithms may also be developed in the future, to improving its accuracy and efficiency, while maintaining the smaller decision/formulation size.

Acknowledgement

The authors thank Mr. Shubham Sharma and Dr. Kannan Chandrasekaran from IOCL's RnD division for extensive discussions on the problem. The authors acknowledge the Center of Excellence for Oil Gas and Energy (CoEOGE), IIT Bombay, for funding the work.

References

[1] L. Guan, Y. Gao, H. Liu, W. An, and A. Noureldin, "A review on small-diameter pipeline inspection gauge localization techniques: Problems, methods and challenges," in *2019 International Conference on Communications, Signal Processing, and their Applications (ICCSPA)*. IEEE, 2019, pp. 1–6.

[2] M. S. Chowdhury and M. F. Abdel-Hafez, "Pipeline inspection gauge position estimation using inertial measurement unit, odometer, and a set of reference stations," *ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems Part B: Mechanical Engineering*, vol. 2, no. 2, p. 021001, 2016.

[3] L. Guan, X. Cong, Y. Sun, Y. Gao, U. Iqbal, and A. Noureldin, "Enhanced mems sins aided pipeline surveying system by pipeline junction detection in small diameter pipeline," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 3560–3565, 2017.

[4] W. M. Al-Masri, M. F. Abdel-Hafez, and M. A. Jaradat, "Inertial navigation system of pipeline inspection gauge," *IEEE Transactions on Control Systems Technology*, vol. 28, no. 2, pp. 609–616, 2018.

[5] Q. Chen, Q. Zhang, X. Niu, and Y. Wang, "Positioning accuracy of a pipeline surveying system based on mems imu and odometer: Case study," *IEEE access*, vol. 7, pp. 104 453–104 461, 2019.

[6] J. R. Adams, P. S. Price, and J. W. Smith, "Inertial based pipeline monitoring system," Patent, 1988, uS Patent 4,945,775,A. [Online]. Available: http://www.google.it/patents/US4945775A

[7] M. B. Ignagni, "Apparatus and method for accurate pipeline surveying," Patent, 2003, uS Patent 6,553,322,B1. [Online]. Available: http://www.google.it/patents/US6553322B1

[8] Chinese-authors, "Inertial navigation measurement method for pipeline center line," Patent, 2003, china Petroleum and Natural Gas Co Ltd. [Online]. Available: http://www.google.it/patents/CN103697886A

[9] ——, "Combination navigation method of inertial navigation odometer for pipeline surveying and mapping," Patent, 2013, beijing Automation Control Equipment Institute BACEI. [Online]. Available: http://www.google.it/patents/CN103727938A

[10] ——, "A kind of small pipeline gis-geographic information system measuring device and its measurement method," Patent, 2015, harbin Engineering University. [Online]. Available: http://www.google.it/patents/CN105066917B

[11] ——, "Error compensation method for strapdown inertial navigation and navigation system," Patent, 2020, chongqing Youth Vocational & Technical College. [Online]. Available: http://www.google.it/patents/CN111811537B

[12] A. Hidalgo, J. Ros, and J. Escalona, "A kalman filter-based algorithm for imu signals fusion applied to track geometry estimation," in *Proc. ECCOMAS Thematic Conf. Multibody Dyn. Barcelona*, 2015, pp. 1–13.

[13] P. G. Sessa, V. De Martinis, and F. Corman, "A hybrid dynamic-kinematic ekf for train trajectory estimation," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2018, pp. 2297–2302.

[14] S. Kazeminasab, N. Sadeghi, V. Janfaza, M. Razavi, S. Ziyadidegan, and M. K. Banks, "Localization, mapping, navigation, and inspection methods in in-pipe robots: A review," *IEEE Access*, vol. 9, pp. 162 035–162 058, 2021.

[15] D. Titterton, J. L. Weston, and J. Weston, *Strapdown inertial navigation technology*. IET, 2004, vol. 17.

[16] A. Noureldin, T. B. Karamat, J. Georgy, A. Noureldin, T. B. Karamat, and J. Georgy, "Inertial navigation system," *Fundamentals of inertial navigation, satellite-based positioning and their Integration*, pp. 125–166, 2013.

[17] K. Hausman, J. Preiss, G. S. Sukhatme, and S. Weiss, "Observability-aware trajectory optimization for self-calibration with application to uavs," *IEEE Robotics and Automation Letters*, vol. 2, no. 3, pp. 1770–1777, 2017.

[18] J. A. Preiss, K. Hausman, G. S. Sukhatme, and S. Weiss, "Trajectory optimization for self-calibration and navigation." in *Robotics: Science and Systems*, vol. 13, 2017.

[19] ——, "Simultaneous self-calibration and navigation using trajectory optimization," *The International Journal of Robotics Research*, vol. 37, no. 13-14, pp. 1573–1594, 2018.

[20] R. Tedrake, *Underactuated Robotics, Chapter-10*, 2023. [Online]. Available: https://underactuated.csail.mit.edu/trajopt.html

[21] N. Khaneja, T. Reiss, C. Kehlet, T. Schulte-Herbrüggen, and S. J. Glaser, "Optimal control of coupled spin dynamics: design of nmr pulse sequences by gradient ascent algorithms," *Journal of Magnetic Resonance*, vol. 172, no. 2, pp. 296–305, 2005. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1090780704003696

[22] E. Bartolini, J.-F. Cordeau, and G. Laporte, "Improved lower bounds and exact algorithm for the capacitated arc routing problem," *Mathematical Programming*, vol. 137, no. 1, pp. 409–452, 2013. [Online]. Available: https://doi.org/10.1007/s10107-011-0497-4