TEST Engineering & Management

March - April 2020
ISSN: 0193-4120 Page No. 1196 - 1201

Effect of Curve Algorithms on Standard SLAM algorithm for Prediction of Path

Rohit Mittal
Research Scholar
Poornima University
Jaipur
Rohit18mittal@yahoo.com

Vibhakar Pathak
Professor
Arya College of Engineering & I.T, Jaipur
vibhakarp@rediffmail.com

Geeta Chhabra Gandhi
Associate Professor
Poornima University
Jaipur
geetachhabragandhi@gmail.com

Nidhi Mishra
Associate Professor
Presidency University
Bangalore
mishra.nidhi7384@gmail.com

Article Info
Volume 83
Page Number: 1196 - 1201
Publication Issue:
March - April 2020

Article History
Article Received*: 24 July 2019*
Revised*: 12 September 2019*
Accepted*: 15 February 2020*
Publication*: 14 March 2020*

Abstract-
Simultaneous Localization and Mapping (SLAM) is one of the enabling technologies for autonomous navigation of robot. In this paper is to estimate-actual, predicted and optimized pose of robot which is equipped with IMU and non visual sensors using curved features as landmarks. Extended Kalman Filter based SLAM algorithm is applied with Bezier/ B Spline curve parametric algorithm using blending functions to make computations more stable. After conducting 160 runs it has been seen that the deviation is in decrease trends from Extended kalman Filter to Extended kalman Filter with Bezier curve and Extended kalman Filter B Spline along with mathematical modeling

Keywords: *SLAM, EKF, Bezier, B Spline, Curve SLAM*

Introduction

Simultaneous Localization and Mapping (SLAM) is one of the enabling technologies for autonomous navigation of robot. The implementation of Simultaneous Localization and Mapping is accomplished through various types of sensors. These sensors have been argument in the given map for purpose of navigation of the robot in previously known map environment. Authors in this paper discuss about path prediction approach using Simultaneous Localization and Mapping based on Extended Kalman Filter algorithm. Here, EKF model is used as non linear transition of robot. There are two model used in EKF viz: State model and Measurement model.

State model represent as:

$$x_{t+1} = f(x_t, u_t + w_t)$$

Where, x is state model use for prediction of robot, x_{t+1} is next state model, u_t is control data and w_t is noise.

Measurement model represent as:

$$z_t = h(x_t + v_t)$$

Where, z_t is measurement model which includes data from various sensors, $v_t$is noise in measurement model.

TEST Engineering & Management

March - April 2020
ISSN: 0193-4120 Page No. 1196 - 1201

Various experiments had been conducted on non linear path (consists of various curves) and to optimized the path some optimizer functions like blending function is used. In this paper, we presents Bezier and B Spline approach to Simultaneous Localization and Mapping algorithm for better prediction of path in curve route. A few algorithms on interpolation, splitting and matching of aforementioned curve path in coming sections will discuss through a set of control points (4) for a segment of path.

Approach validation will also discuss in upcoming section with the help of experimental hardware used in previously known mapped environment in conjunction with comparison among various curve SLAM techniques. Next section describes about related work in Curve Slam EKF along with mathematical modeling of robot with results set and conclusion.

Related work

Various Simultaneous Localization and Mapping algorithms have incorporated into maps of robots in indoor environment. Place recognition SLAM algorithms have been developed to track feature points. FullSLAM algorithm has full trajectory of path that traversed by robot which is based on previous state data [4].

Various works on Multi Level Relaxation Simultaneous Localization and Mapping algorithm had done by researchers which optimize the map at multiple levels of resolution. In past most of the researchers uses orthogonal planes (windows, doors) to represent structural landmarks [1][2].

Some of them represent line segments for map the environments using vision based sensors. Various works had done in field of vision based sensors. Some projects make use of depth camera sensors and graph optimizations. Kinect based optical sensors uses framework which is based on OpenNI that implements gesture recognitions and motion tracking, for which Haris Corner methods are used [3] [6].

In early work, authors had reported work on curve SLAM algorithms (Bezier) using visual sensors which reduce the number of landmarks by various orders of magnitude. It uses the 6DOF pose of robot and provides the known map environment [5].

These authors do not evaluate the performance of non linear transition systems using EKF argument with B Spline curve optimization algorithm.

Overview of Curve SLAMs with EKF

The goal of this paper is to estimate- actual, predicted and optimized pose of robot which is equipped with IMU and non visual sensors using curved features as landmarks. The curves can be determined by number of straight line segments which consist of control points. These points are used in interpolation o approximation techniques, when optimized curve passes through all or some selected control points. For interpolation of control points, blending function is needed to check the position of curve [5].

$$fx(u) = \sum_{i=1}^{n} X_i B_i(u)$$

B spline is linear combination of control points and B Spline basis function. These control points are defined in terms of basis matrix and geometry matrix as:

$$S(t) = T \,.\, M_{bs} . G_{bs}$$

$$= (t^3 \; t^2 \; t \; 1) \begin{pmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 0 & 3 & 0 \\ 1 & 4 & 1 & 0 \end{pmatrix} \begin{pmatrix} P_i \\ P_{i+1} \\ P_{i+2} \\ P_{i+3} \end{pmatrix}$$

The vector that contains basis splines as:

$$B_{bs} = \begin{pmatrix} -t^3 + 3t^2 - 3t + 1 \\ 3t^3 - 6t^2 + 4 \\ -3t^3 + 3t^2 + 3t + 1 \\ t^3 \end{pmatrix}$$

During autonomous navigation of robot on curved surface patches, **Bezier curve** is needed; that is defined by its control points which are located at start and end point of the curve. The proposed curve can be expressed parametrically. Bezier curve is able to distinguish feature points while navigating the robot on previously known structured map in environment. Bezier Curves specifies the control points and which interpolates the end points and approximates the other two remaining points and it can be fitted to any number of control points. Without necessitate tangent vector specification at any of the control points, a set of characteristic polynomial uses approximating functions called Bezier blending functions which blends the control points to produce Bezier curve segment. The degree of Bezier Curve segment is determined by number of control points to be fitted with that curve segment [5].

The Bezier geometry matrix is shown as:

$$G_b = \begin{pmatrix} P_1 \\ P_2 \\ P_3 \\ P_4 \end{pmatrix}$$

The Bezier basis matrix evaluates as:

$$M_b = \begin{pmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}$$

By combining the two notations one can express any curve segment with the help of multiplication of matrices and vectors as:

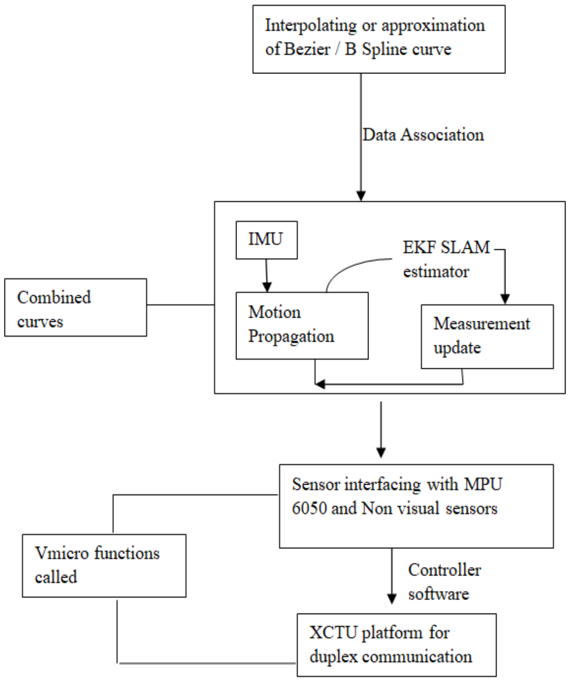
$$S(t) = T\,.M_b.G_b$$

Further, transformations grants a formulation with help of Bezier basis splines

$$B_b = \begin{pmatrix} -t^3 + 3t^2 - 3t + 1 \\ 3t^3 - 6t^2 + 3t \\ -3t^3 + 3t^2 \\ t^3 \end{pmatrix} = \begin{pmatrix} (1-t)^3 \\ 3t(1-t)^2 \\ 3t^2(1-t) \\ t^3 \end{pmatrix}$$

Curve parameter optimization using Blending and EKF technique

Extended Kalman Filter algorithm implemented using floating point arithmetic calculations (that is its prediction and updation steps), to make these computations more stable Bezier/ B Spline curve parameters are used due to its property of convex hull which is discussed in this paper in form of algorithm.



Robot's navigation behavior

The robot's **initial state** is shown as:

$$x = \sum_{i=1}^{8} [x_i \quad y_i \quad \dot{x} \quad \dot{y}]$$

$$\overline{x_n} = [x \quad y \quad z \; \theta \quad \emptyset \quad \varphi \quad \dot{\beta}]$$

Where $x \quad y \quad z$ is pose of robot while navigation, $\theta \quad \emptyset \quad \varphi$ is Infrared, Sonar, Friction and $\dot{\beta}$ is (bias gyroscope measurement). The measurement matrix is shown as;

$$\bar{z} = [\dot{x}_{cp} \quad \dot{y}_{cp} \quad \dot{z}_{cp} \quad \dot{\theta} \quad \dot{\emptyset} \quad \dot{\varphi}]$$

$$h = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

The **state vector** is represented as:

$$x = [x \quad y \quad \dot{x} \quad \dot{y}]^T$$

To measure the orientation of robot in 2D, the process relies on the measurement of friction (φ), acceleration.

Algorithm

1. To set the predicted state of robot values of X_0 and Y_0 are needed for non linear

March - April 2020
ISSN: 0193-4120 Page No. 1196 - 1201

extended kalman filter at any discrete time t.

2. State of robot consists of Inertial Measurement Unit and friction variables.
3. Call robot_predict() by taking θ $\emptyset$ φ variables
4. User routine to set the number of line segments per curve section for B Spline interpolation
   i. Call Blend, FIRST_BLEND, LAST_BLEND
   ii. Array size is 4 by maximum number of lines for blending function
5. As Bezier curve follows convex hull property so to cover the region mid points are needed along with desired degree of subdivision
   i. Find First_level mid points
   ii. Find second_level mid points
6. Call BEZIER-ABS (X,Y)
7. Start B Spline curve using 4 control points by loading XSM, YSM (sample points in X and Y direction) 4 by lines per section containing blending function values
   i. Call Make_Curve(FIRST_BLEND)
   ii. Call Make_Curve(SECOND_BLEND)
8. Measure the gyroscopic orientation of robot with respect to aforementioned variables.

Results

After navigation of robot in experimental area, conducting 160 runs we conclude by averaging of 10 runs to I value we found that:

Deviation from actual position of robot to prediction position in Extended kalman Filter is 36% in X direction and 20% in Y direction which are generated by navigation of robot and deviation is computed using aforementioned porting algorithm. The deviation between distances travelled by robot on actual path to predicted path is 2.5% and with respect to optimized path it is 55%, after that the deviation gets decreases to 20% which is shown in fig 1 and in last the deviation becomes 10%. The results are discussed in table 1 and 2. The deviation is in decrease trends from Extended kalman Filter to Extended kalman Filter with Bezier curve and Extended kalman Filter B Spline, which indicates that if Extended kalman Filter is augment with one curve it produce better results with decrease of 2.2% to 1.8% depends on curve algorithm.

Table 1 Navigation of robot using Bezier Curve

Number of runs	Actual Path (in cm.)		Predicted Path (in cm.)		Optimized Path (in cm.)	
	x-axis	Y-axis	x-axis	Y-axis	x-axis	Y-axis
1	98	53	79	51	88	52
2	165	87	126	89	145	88
3	271	99	299	105	334	103
4	318	116	313	111	315	113
5	402	221	408	188	403	193
6	476	400	491	354	480	373
7	550	523	565	476	555	497
8	710	601	662	561	681	587
9	820	532	791	509	802	504
10	888	404	904	446	891	422
11	912	376	965	361	945	370
12	965	351	987	341	973	345
13	1116	323	1020	312	1086	318
14	1208	272	1098	262	1111	265
15	1267	206	1234	219	1246	211
16	1303	189	1267	205	1292	198

March - April 2020
ISSN: 0193-4120 Page No. 1196 - 1201

Table 2 Navigation of robot using B spline curve

Number of runs	Actual Path (in cm.)		Predicted Path (in cm.)		Optimized Path (in cm.)	
	x-axis	Y-axis	x-axis	Y-axis	x-axis	Y-axis
1	109	59	97	55	103	57
2	178	95	157	91	167	92
3	295	105	299	105	297	105
4	309	124	315	114	312	119
5	413	239	408	207	410	218
6	481	434	479	454	480	444
7	571	554	565	538	572	543
8	698	552	673	539	691	541
9	799	567	788	554	792	561
10	876	444	897	439	888	441
11	923	387	951	367	936	372
12	968	329	978	339	971	333
13	1043	304	1020	312	1030	308
14	1122	254	1131	249	1128	251
15	1205	211	1222	219	1211	215
16	1298	175	1301	184	1304	179

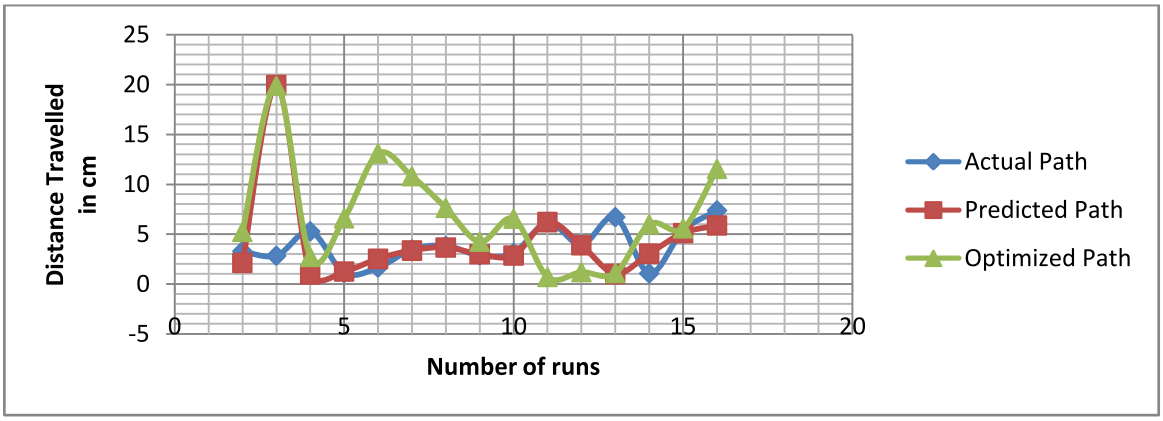


Fig 1 Deviation of robot on different using Curve SLAM algorithms

Conclusion

It has been concluded that after conducting number of runs using ported algorithm the deviation from actual position of robot to prediction position in Extended kalman Filter is 36% in X direction and 20% in Y direction which are generated by navigation of robot and deviation was computed. It has been seen that through Bezier curve optimized path could be achieved and 2.2 % deviation in between actual, predicted and optimized path can be more optimized by 1.8% if B spline blending functions are applied. Finally, Mathematical modeling is done provide profiled map in unknown environment along with optimized algorithm.

References

[1] Rohit Mittal, Vibhakar Pathak et.al "Localization and Impulse Analysis of Experimental Bot using chromos ez430" in Applications of Computing, Automation and Wireless Systems in Electrical Engineering. Lecture Notes in Electrical Engineering, vol 553. Springer, Singapore, June 2019

[2] Mittal, R., Mishra, N., Pathak, V. "Exploration of designed robot in non linear movement using EKF SLAM and its stability on loose concrete

March - April 2020
ISSN: 0193-4120 Page No. 1196 - 1201

surface", 2019, International Journal of Engineering and Advanced Technology
[3] Esa Apriaskar, Yayan P. Nugraha et.al "Simulation of Simultaneous Localization and Mapping Using Hexacopter and RGBD Camera", 2017 2nd International Conference on Automation, Cognitive Science, Optics, Micro Electro-Mechanical System, and Information Technology (ICACOMIT),
[4] Nitin Kumar Dhiman, Dipti Deodhare et.al "Where am I? Creating spatial awareness in unmanned ground robots using SLAM: A survey"
[5] Schmid C, Zisserman A. The geometry and matching of lines and curves over multiple views. Int J Comput Vision. 2000;40(3):199–233.
[6] KevinMeier1 Soon-Jo Chung et.al Visual-inertial curve simultaneous localization and mapping: Creating a sparse structured world without feature points, Journal of Field Robotics 2017