10th World Congress on Structural and Multidisciplinary Optimization
May 19 -24, 2013, Orlando, Florida, USA

A 3D Location Estimation Method using the Levenberg-Marquardt Method for Real-Time Location System

Kazuya Shoda[1], Masao Arakawa[2]
Masashi Morikawa[1], Takeshi Hisano[1], Katsuhiko Matsumura[1]

[1] Daifuku Co., Ltd., 1225 Nakazaiji, Hino-cho, Gamo-gun, Shiga, Japan, kazuya_shoda@ha.daifuku.co.jp
[2] Kagawa University, 2217-20, Hayashi-cho, Takamatsu, Kagawa, Japan, arakawa@eng.kagawa-u.ac.jp

1. Abstract

Recently, real-time location systems (RTLS) are being used for inventory management and safety support in factories and distribution centers. It is necessary in RTLS to estimate the location of moving objects such as forklifts and workers, in addition to stationary bodies. To estimate the location of moving objects, a highly accurate and fast algorithm is required. Therefore, in this study, we applied the Levenberg-Marquardt method (LMM), [1] which is popular as a quickly converging algorithm, to 3D location estimation problems and carried out two experiments. We compared the results of estimating 3D location by LMM and the steepest descent method (SDM) at multiple points and found no significant difference in terms of location accuracy. For the computation time, LMM is several hundred times faster than SDM, being able to consistently compute a location in 0.03 ms or less.

2. Keywords: Material Handling System, Real-Time Location System, Ultra Wideband, Levenberg-Marquardt Algorithm, Optimization

3. Introduction

RTLS are systems for detecting the location information of tags attached to a person or object in real time. RTLS are popular as car navigation systems and for mobile phone map applications. Uses in factories and distribution centers include inventory management and safety support. Location detection devices currently in use include those based on GPS, Wi-Fi, and ultra-wideband (UWB). Figure 1 shows the relationship between the accuracy and the installation interval of sensors for each of these devices. For this study, we selected a UWB device in consideration of the following two points because UWB devices offer a better balance of accuracy and cost than other devices in use at factories and distribution centers.

(a) To detect pallets, cases, workers, and so on, a location accuracy of 20 to 30 cm is needed.
(b) The number of sensors per unit area needs to be kept to a minimum to control cost.

UWB is a radio technique using ultra-wide bandwidth (defined as greater than 450 MHz in Japan), which is used for high-speed data transmission and obstacle detection by radar, in addition to location detection. [2] Measuring methods include those for time of arrival (ToA), time difference of arrival (TDoA), angle of arrival (AoA), and received signal strength (RSS). The following procedure shows a ToA method. Note that the ToA method cannot measure distances in parallel, but its cost is lower than the TDoA measure, which uses multiple sensors in a system at the same time.

Procedure of ToA method

[STEP1] Sensors repeat the following three steps for each tag.

- [STEP1-1] The sensor sends a trigger to a tag. When a tag receives it, the tag returns a response.
- [STEP1-2] The sensor calculates the distance from the arrival time of the received radio-wave.
- [STEP1-3] The sensor sends the distance to location server computer.

[STEP2] The location server computer estimates the location of the tag from the location of sensors and the measured distances using LMM or SDM.

There are several methods for 3D location estimation. (E.g., least squares method (LS), maximum likelihood estimation (ML), maximum a posteriori (MAP)) For this study, we selected LMM as the solver of LS. To estimate the location of moving object such as forklifts and workers, a highly accurate and fast algorithm is required. Therefore, in this study, we applied the LMM, which is popular as a quickly converging algorithm, to 3D location estimation problems and carried out several experiments to demonstrate its effectiveness.

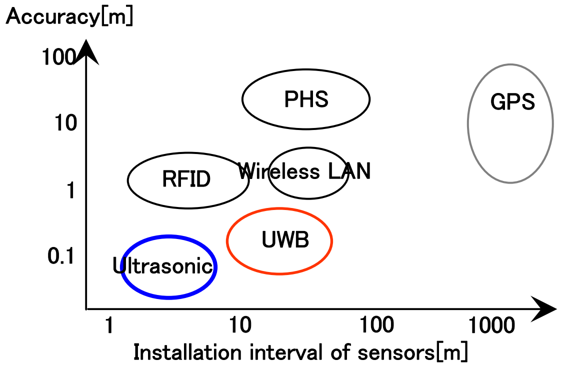

Figure 1: Type of localization device

4. Levenberg-Marquardt Method (LMM)

In outline, the 3D location estimation problem is defined as in Eq. (1).

$$\min_{x,y,z} \sum_{i=1}^{M} \{d_i^2 - (x-x_i)^2 - (y-y_i)^2 - (z-z_i)^2\}^2 \quad (1)$$

(x_i, y_i, z_i) : location of sensor

d_i : distance between sensor and tag

(x, y, z) : probable location of tag as calculated by the algorithm

M : number of sensors ($M \geq 3$)

It is a quadratic and unconstrained optimization problem, which is to solve for (x, y, z) that minimizes the error sum of squares of d_i and (x_i, y_i, z_i) . As the solver, several good methods have been proposed, but LMM is popular as a quickly converging algorithm and LMM is effective when there are many unknown parameters.

$$F(x) = \{f_1(x), f_2(x), \ldots, f_m(x)\}^T \quad (2)$$

When vector function $F(x)$ is defined as in Eq. (2), we consider the minimization of the sum of squares of the elements of this vector function.

$$\min F(x)^T F(x) = \min \sum_{i=1}^{m} f_i(x)^2 \quad (3)$$

Accordingly, Eq. (3) suggests solution by the least-squares method. LMM considers the first component of the Taylor expansion as given in Eq. (4).

$$F(x) = F(x^i + s) = F(x^i) + \nabla F(x^i)^T s \quad (4)$$

So we can consider this problem as that of solving for the incremental vector s . Substituting Eq. (4) into Eq. (3) gives Eq. (5).

$$F(x^i + s)^T F(x^i + s) = F(x^i)^T F(x^i) + 2\nabla F(x^i)^T F(x^i)s + s\nabla F(x^i)^T \nabla F(x^i)s \quad (5)$$

Partially differentiating the left-hand side of Eq. (5) by the incremental vector s and setting the result equal to zero, we can immediately obtain a solution.

$$2\nabla F(x^i)^T F(x^i) + 2\nabla F(x^i)^T \nabla F(x^i)s = 0 \quad (6)$$

$$s = -(\nabla F(x^i)^T \nabla F(x^i))^{-1} \nabla F(x^i)^T F(x^i) \quad (7)$$

However, oversensitivity to errors may cause s to become large. Therefore, matrix D is added to Eq. (7) as a control term.

$$s = -(\nabla F(x^i)^T \nabla F(x^i) + \lambda D)^{-1} \nabla F(x^i)^T F(x^i) \quad (8)$$

Matrix D is an ordinary unit matrix. If matrix D is the diagonal element of $\nabla F(x^i)^T \nabla F(x^i)$, this is expected to speed up the update of Eq. (9).

$$x^{i+1} = x^i + s \quad (9)$$

Additionally, LMM generally uses a linear search of λ to utilize the obtained Jacobian because LMM takes the greatest amount of its time in solving for the Jacobian. However in the current case, the Jacobian has already been calculated in formulating the problem, and so we repeatedly apply a given rule and do not use linear search. In this case, λ is one thousandth of Eq. (1).

5. Verification experiment

Handling 3D location estimation problems involves solving non-linear least-squares problems to minimize the error sum of squares of distance as measured by sensors and the probable distance as calculated by the algorithm.

One general method for solving this problem is the SDM. SDM is applied to problems that have multiple continuous design variables, a single objective function, and no constraint conditions. SDM minimizes the objective function along its gradient. Here, we use linear search of a coefficient to improve the convergence of SDM.

In this study, we compared SDM to LMM as a 3D location estimation method. As shown in Figure 2, we set up one sensor each at the four corners of a 12×12 m square and placed nine tags within the square. Locations of the sensors and tags were measured using a total station system with an accuracy of ±2 mm. These locations are shown in Tables 1 and 2 for the sensors and tags, respectively. We ran two experiments; the results for SDM and LMM are listed in Tables 3 through 6.

5.1. Experiment 1

We calculated distances between each sensor and each tag from the data in Tables 1 and 2, and then used LMM or SDM to estimate the location of each tag from these distances. This experiment is a comparison of the two algorithms under ideal conditions. For each tag, the initial location estimate was $(x, y, z) = (0,0,0)$. The location for each tag was calculated 10,000 times; the results using SDM and LMM are summarized in Tables 3 and 4, respectively accuracy than SDM for the error sum of squares of Eq. (1). LMM also produces smaller location errors than SDM. For the computation time, LMM is several hundred times faster than SDM. Therefore, LMM is revealed to be superior to SDM under ideal conditions. Note that the computation time of SDM is large because of the linear search of the coefficient for updating.

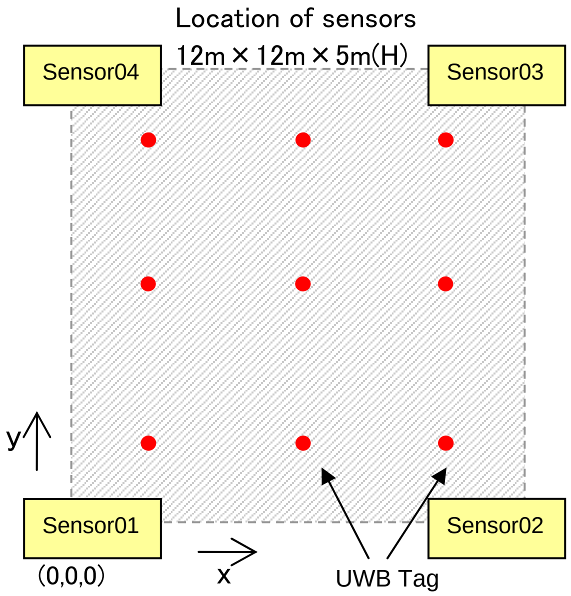

Figure 2: Arrangement of sensors and tags

Table 1: Locations of UWB sensors

	X [mm]	Y [mm]	Z [mm]
Sensor01	-19	43	5002
Sensor02	12040	-25	5039
Sensor03	12034	11968	5046
Sensor04	-1	12030	5044

Table 2: Locations of UWB tags

	X [mm]	Y [mm]	Z [mm]
Tag01	1995	1996	3024
Tag02	2007	6002	3027
Tag03	1997	9998	3020
Tag04	6003	1996	3019
Tag05	5999	6007	3013
Tag06	5994	9996	3014
Tag07	9991	1993	3018
Tag08	9999	6007	3022
Tag09	10000	10004	3018

5.2. Experiment 2

We detected the location of each tag 10,000 times using the actual equipment (i.e., the sensors). Similarly to in experiment 1, the initial estimated location for each tag was $(x, y, z) = (0,0,0)$. Tables 5 and 6 are the results using SDM and LMM, respectively, of the 10,000 trials. The detection accuracy of the two methods is about the same, although the location obtained using SDM is near the true location in some cases. Similarly to in experiment 1, LMM is several hundred times faster than SDM.

6. Conclusions

In this study, we applied LMM as a 3D location estimation method. We found that LMM was faster by a factor of several hundred to several thousand than SDM, which is the most basic method for solving non-linear least-squares problems. In addition, the LMM consistently converged to the global optimum solution at the all trials (2 experiments × 10,000 trials × 9 tags). We could thus verify the effectiveness of LMM through the experiments.

In the experiments of this study, stationary bodies were measured in an environment with no obstructions. As a result, we could understand the basic characteristic of RTLS. In future studies, we would like to make experiments involving moving objects in a factory or distribution center under actual operating conditions.

7. References

[1] DW Marquardt, 1963, "An Algorithm for Least-Squares Estimation of Nonlinear Parameters", Journal of the Society for Industrial and Applied Mathematics,1963, Vol. 11, No. 2 : pp. 431-441

[2] MBOA-SIG White Paper, 2004,"Ultrawide-band: High-speed, short-range technology with far-reaching effects"

Table 3: Results of experiment 1 using SDM

	X [mm]	Y [mm]	Z [mm]	Location Error [mm]	Error sum of squares [m^2]	Funcation call [times]	Computation time [ms]
Tag01	1995.16	1996.16	3022.36	1.654	9.30E-05	91	26.957
Tag02	2007.29	6002.00	3025.63	1.397	8.80E-05	62	18.123
Tag03	1997.28	9997.72	3018.29	1.754	1.14E-04	70	20.774
Tag04	6003.00	1996.15	3017.58	1.424	9.30E-05	61	17.806
Tag05	5998.91	6007.06	3011.85	1.157	9.30E-05	23	6.527
Tag06	5994.00	9996.28	3015.40	1.424	9.20E-05	21	5.995
Tag07	9990.73	1993.28	3016.31	1.735	1.11E-04	72	21.478
Tag08	9999.30	6007.00	3023.27	1.306	8.30E-05	11	3.006
Tag09	9999.75	10003.75	3016.22	1.816	1.15E-04	73	21.802

Table 4: Results of experiment 1 using LMM

	X [mm]	Y [mm]	Z [mm]	Location Error [mm]	Error sum of squares [m^2]	Funcation call [times]	Computation time [ms]
Tag01	1995.00	1996.00	3023.70	0.297	6.00E-06	5	0.026
Tag02	2007.00	6002.00	3026.93	0.065	0	6	0.043
Tag03	1997.00	9998.00	3019.99	0.011	0	7	0.043
Tag04	6003.00	1996.00	3018.94	0.063	0	6	0.024
Tag05	5999.00	6007.00	3012.28	0.716	3.30E-05	6	0.024
Tag06	5994.00	9996.00	3013.96	0.042	0	7	0.026
Tag07	9991.00	1993.00	3017.99	0.011	0	7	0.027
Tag08	9999.00	6007.00	3021.96	0.044	0	7	0.027
Tag09	10000.00	10004.00	3017.32	0.678	3.00E-05	7	0.027

Table 5: Results of experiment 2 using SDM

	Average			Standard deviation						
	X [mm]	Y [mm]	Z [mm]	X [mm]	Y [mm]	Z [mm]	Location Error [mm]	Error sum of squares [m^2]	Funcation call [times]	Computation time [ms]
Tag01	1923.04	1994.89	3109.18	18.23	18.52	31.19	111.51	1.529	104.155	31.224
Tag02	1999.11	6034.75	3016.84	9.52	9.51	27.06	35.19	1.930	65.368	19.702
Tag03	1991.16	10073.44	3137.11	19.13	19.32	28.74	139.43	2.347	77.115	23.206
Tag04	5995.69	1946.07	2901.24	8.27	8.40	18.80	128.11	0.062	59.918	18.033
Tag05	5999.78	6015.16	2886.87	7.17	7.30	20.46	126.39	0.030	8.595	2.852
Tag06	6031.11	10029.42	2887.61	8.62	8.75	22.41	135.89	0.055	46.766	14.191
Tag07	9985.75	1929.53	3048.36	11.14	11.08	27.70	70.56	0.112	71.899	21.769
Tag08	10031.31	6000.47	2928.15	8.63	8.64	20.43	99.47	0.567	43.455	13.339
Tag09	10052.81	10028.52	3045.47	63.84	63.60	69.91	64.38	2.616	70.056	20.859

Table 6: Results of experiment 2 using LMM

	Average			Standard deviation						
	X [mm]	Y [mm]	Z [mm]	X [mm]	Y [mm]	Z [mm]	Location Error [mm]	Error sum of squares [m^2]	Funcation call [times]	Computation time [ms]
Tag01	1921.97	1994.60	3111.37	18.23	18.52	31.26	113.88	1.529	4.999	0.021
Tag02	1999.00	6034.75	3017.66	9.53	9.52	27.14	34.98	1.930	5.002	0.022
Tag03	1990.87	10073.73	3138.99	19.14	19.33	28.80	141.18	2.347	6.000	0.025
Tag04	5995.69	1945.98	2902.01	8.27	8.40	18.81	127.45	0.062	5.001	0.022
Tag05	5999.73	6015.18	2886.96	7.17	7.30	20.51	126.30	0.030	5.999	0.025
Tag06	6031.11	10029.31	2886.84	8.62	8.74	22.40	136.59	0.055	6.003	0.026
Tag07	9986.02	1929.25	3050.05	11.14	11.08	27.73	71.53	0.112	6.000	0.025
Tag08	10031.14	6000.47	2927.42	8.63	8.64	20.44	100.11	0.567	6.000	0.025
Tag09	10053.03	10028.74	3047.21	63.86	63.62	70.20	65.40	2.616	6.993	0.030