FACTOR GRAPHS FOR MOBILE POSITION LOCATION

Jung-Chieh Chen, Ching-Shyang Maa, and Jiunn-Tsair Chen

Mobile Communications Lab
Institute of Communications Engr.
National Tsing Hua University, Hsinchu, Taiwan 30055, R.O.C.

ABSTRACT

Making use of the network side time-of-arrival (TOA) measurements, we propose a low-complexity high-accuracy algorithm to estimate the location of a target mobile station (MS). Under a factor graph framework, the proposed algorithm efficiently exchange soft information among local processing units in the mobile switching center (MSC) to iteratively purify the estimate of the MS location. Numerical results show that the proposed algorithm not only enjoys advantages of low complexity, suitable for integrated-circuit implementation, but it is also able to achieve performance very close to the optimum achievable bound, the maximum likelihood (ML) bound.

1. INTRODUCTION

Wireless geolocation technology is considered a key enabling technology [1], because it has great potential to promote many other related applications that could significantly expand the wireless user base. Third generation (3G) wireless communication systems [1] have been among the first to adopt location strategies in their standards. Position location (PL) methods for cellular systems can be divided into two categories: handset-based and network-based methods. Wireless geolocation techniques to locate a mobile station (MS) in a wireless communication system make use of measurements on time of arrival (TOA), angle of arrival (AOA), or their combinations [2]. This paper concentrates on the TOA-based PL techniques applied to the network side.

In the TOA-based PL techniques, each TOA measurement made at a base station (BS) produces a circle, centered at that BS. The target MS is supposed to be located on the circle. Making use of at least three circles produced by TOA measurements at three different BSs, one can identify the two-dimensional (2-D) location of the MS as the intersection of these circles. A traditional way to solve the problem is to calculate the least square (LS) solution after applying the Taylor series linearization (TS-LS) [3] on the lines of position (LOP). The TS-LS algorithm can provide reasonably accurate location estimates, but it requires an initial location guess and may suffer from the convergence problem if the initial guess is not good enough. Alternatively, Caffery [4] proposed a simple geometrical approach in which linear LOPs, rather than circular LOPs, are used to determine the location of the MS. However, the accuracy of the geometrical algorithm is relatively low.

In this paper, we propose a low-complexity high-accuracy novel iterative algorithm, which takes into consideration the stochastic properties of the measurement errors. To effectively utilize the available information, based on a factor graph [5] framework, soft information is exchanged among local processing units in the proposed algorithm to obtain the location estimates of the target MS. Numerical results show that the proposed algorithm not only enjoys advantages of low complexity, easy for implementation, but it is also able to achieve performance very close to the optimum achievable bound, the maximum likelihood (ML) bound.

This paper is organized as follows. The proposed PL algorithm is developed in Section 2-4. Simulation results that compare the performance bounds and the estimation errors of various location algorithms, including the proposed one, are then given in Section 5.

2. FACTOR GRAPH

Factor graph is a bipartite graph that expresses how a complicated *global* function, or global task, of many variables factors into the product of several simple *local* functions, or simple local tasks. Factor graph consists of two kinds of nodes, variable nodes and agent (function) nodes. An edge connects a variable node x and an agent node g_i, named after its associated local function $g_i(\cdot)$, if and only if x is an argument of $g_i(\cdot)$. Fig. 1 shows a simple factor graph representation of $g(x_1, x_2, x_3, x_4) = g_1(x_1, x_2) \cdot g_2(x_2) \cdot g_3(x_2, x_3, x_4)$ with agent nodes g_1, g_2, g_3 and variable nodes x_1, x_2, x_3, x_4. The solution of the variables to the problem described by the factor graph is usually obtained through an iterative soft information-passing procedure based on the sum-product algorithm, to be briefly described next.

The rules for computing soft information are described as follows: 1) A piece of soft information passed from a variable node to an agent node is simply the direct product of all the soft information coming to the variable node from all the other neighboring agent nodes. For example, as shown in Fig. 1, the soft information passed from x_2 to g_3 is given by

$$\mathrm{SI}\left(x_2, g_3\right) = \mathrm{SI}\left(g_1, x_2\right) \cdot \mathrm{SI}\left(g_2, x_2\right), \tag{1}$$

where SI(a, b) denotes the soft information passed from node a to node b. 2) A piece of soft information passed from an agent node to a variable node is the product of a local function associated with the agent node and all the soft information coming to the agent node from all the other neighboring variable nodes, summarized over the variable associated with the variable node receiving soft information. In Fig. 1, the soft information passed from g_3 to x_2 can be expressed as

$$\mathrm{SI}\left(g_3, x_2\right) = \int_{x_3}\int_{x_4} g_3(x_2, x_3, x_4) \cdot \mathrm{SI}\left(x_3, g_3\right) \cdot \mathrm{SI}\left(x_4, g_3\right) dx_3\, dx_4, \tag{2}$$

where $g_3(x_2, x_3, x_4)$ is the associated local function, or the constraint rule of the agent node g_3 to define the mutual relation among x_2, x_3 and x_4.

With these two rules, soft information are iteratively passed around among neighboring nodes. Let the overall soft information of a variable node be defined as the product of all incoming soft information, e.g., the soft information of x_2 can be written as

$$\mathrm{SI}\left(x_2\right) = \mathrm{SI}\left(g_1, x_2\right) \cdot \mathrm{SI}\left(g_2, x_2\right) \cdot \mathrm{SI}\left(g_3, x_2\right). \tag{3}$$

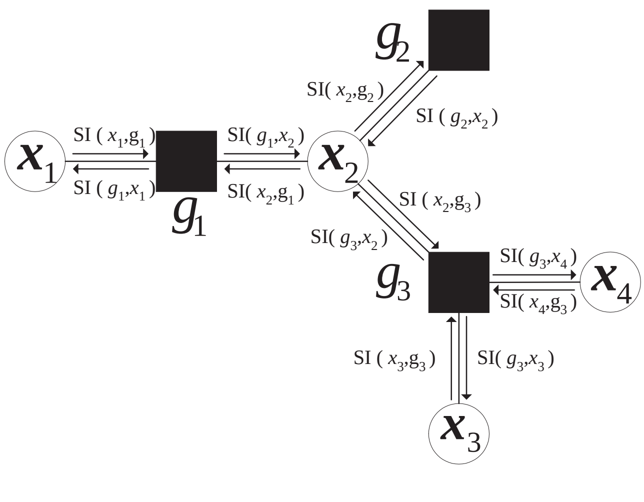

Fig. 1. An example factor graph.

3. SYSTEM MODEL

Let us first construct an appropriate factor graph model for the PL system. In order to reduce the complexity of the 2-D MS location problem, based on the geometrical relationship, we divide the TOA-base PL system into two main groups, $x-$coordinate group and $y-$coordinate group, illustrated in Fig. 2. Each main group contains N subgroups, where N is the total number of involved BSs. Each subgroup describes both the direction and the distance of the target MS as related to the associated BS in the referred coordinate. The two main groups are then connected through agent nodes $\{C_i\}$.

For the purpose of staying low complexity, as will be made clear in the next section, we assume Gaussian statistics for all the variables in Fig. 2. Under this assumption, $\triangle x_i$ and $\triangle y_i$, the signed relative distance estimate from the MS to the i^{th} BS in the referred coordinate can be described as

$$\begin{cases} \triangle x_i = X_i - x + n_{\triangle x_i} \\ \triangle y_i = Y_i - y + n_{\triangle y_i} \end{cases} ; i = 1, 2, \cdots, N, \tag{4}$$

where (X_i, Y_i) is the known exact coordinate vector of the i^{th} BS; (x, y) is the unknown exact location of the target MS; and the estimation errors $n_{\triangle x_i}$ and $n_{\triangle y_i}$ are assumed to be zero-mean Gaussian random variables with variances $\sigma^2_{\triangle x_i}$ and $\sigma^2_{\triangle y_i}$, respectively. Given the MS location estimate $\widehat{x}$ and $\widehat{y}$, (4) can be slightly modified as

$$\begin{cases} \triangle x_i = X_i - \widehat{x} + n_{\triangle x_i} \\ \triangle y_i = Y_i - \widehat{y} + n_{\triangle y_i} \end{cases} ; i = 1, 2, \cdots, N. \tag{5}$$

On the other hand, given $\triangle x_i$ and $\triangle y_i$, the MS location estimate can be written as

$$\begin{cases} \widehat{x} = X_i - \triangle x_i + n_{\triangle x_i} \\ \widehat{y} = Y_i - \triangle y_i + n_{\triangle y_i} \end{cases} ; i = 1, 2, \cdots, N. \tag{6}$$

For convenience, we use the notation $\mathcal{N}\left(x, m, \sigma^2\right)$ to represent a Gaussian pdf, with x being the dummy variable and with m and σ^2 being the mean and variance, respectively.

4. SOFT INFORMATION CALCULATION

We first describe how to obtain the initial pdf guess of the MS location $\widehat{x}$ and $\widehat{y}$ followed by the descriptions in details of the operations conducted by each node in Fig. 2. In the very beginning, a rough MS location $(\widehat{x}(0), \widehat{y}(0))$ can be estimated with an arbitrary 3-BS geometrical algorithm. Based on the rough MS location estimate, we then project the variance of the range measurement error σ^2_i onto the $x-$coordinate axis and the $y-$coordinate axis, respectively, to obtain the variances $\sigma^2_{\triangle x_i}$ and $\sigma^2_{\triangle y_i}$, respectively, in the $x-$direction and in the $y-$direction. Thus, with $\widehat{x}(0)$, $\widehat{y}(0)$, $\sigma^2_{\triangle x_i}$, and $\sigma^2_{\triangle y_i}$ available and with Gaussian pdf assumption of $\widehat{x}$ and $\widehat{y}$, the initial pdf guesses of $\widehat{x}$, $\widehat{y}$ are given as $\mathcal{N}\left(\widehat{x}^0, \widehat{x}(0), \sigma^2_{\triangle x_i}\right)$ and $\mathcal{N}\left(\widehat{y}^0, \widehat{y}(0), \sigma^2_{\triangle y_i}\right)$, respectively.

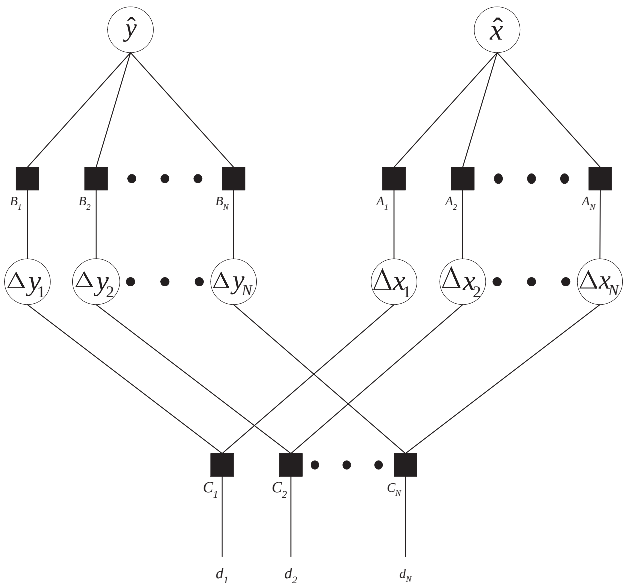

Fig. 2. Factor graph representations of the proposed TOA-based PL system.

After initialization, we next introduce the operation carried out by each variable node and each agent node in Fig. 2 as follows.

1. Variable nodes $\widehat{x}$ and $\widehat{y}$: Based on (1), the soft information passed from variable node $\widehat{x}$ to agent node A_i in the k^{th} iteration is a Guassian pdf of $\widehat{x}^k$ and can be expressed as

$$\mathrm{SI}\left(\widehat{x}^k, A_i^k\right) = \prod_{j\neq i} \mathrm{SI}\left(A_j^k, \widehat{x}^k\right), \tag{7}$$

where $\mathrm{SI}\left(A_j^k, \widehat{x}^k\right)$ is a Gaussian pdf of $\widehat{x}^k$ with its initial value being $\mathcal{N}\left(\widehat{x}^0, \widehat{x}(0), \sigma^2_{\triangle x_j}\right)$ and the superscript k denotes the iteration index. Note that the product of any J Gaussian pdf's is also Gussian and can be simply derived as [5]

$$\prod_{j=1}^{J} \mathcal{N}\big(x, m_j, \sigma_j^2\big) \propto \mathcal{N}\big(x, m_\Lambda, \sigma_\Lambda^2\big) \tag{8}$$

where

$$\frac{1}{\sigma_\Lambda^2} = \sum_{j=1}^{J} \frac{1}{\sigma_j^2};\ m_\Lambda = \sigma_\Lambda^2 \sum_{j=1}^{J} \frac{m_j}{\sigma_j^2}. \tag{9}$$

From (8) and (9), the calculation of (7) is just simple arithmetic. A similar procedure, as that described above for variable node $\widehat{x}$, can be applied to variable node $\widehat{y}$.

2. Variable nodes $\triangle x_i$ and $\triangle y_i$: According to (1) again, variable node $\triangle x_i$ directly passes Gaussian pdf's of $\triangle x_i$,

$$\mathrm{SI}\left(\triangle x_i^k, A_i^k\right) = \mathrm{SI}\left(C_i^k, \triangle x_i^k\right) \tag{10}$$

and

$$\mathrm{SI}\left(\triangle x_i^k, C_i^k\right) = \mathrm{SI}\left(A_i^k, \triangle x_i^k\right), \tag{11}$$

as the soft information between agent nodes A_i and C_i. It is the same for variable node $\triangle y_i$ with respect to agent nodes B_i and C_i.

3. Agent nodes A_i and B_i: Based on (2), the soft information passed from agent node A_i to variable node $\triangle x_i$ in the k^{th} iteration is a Gaussian pdf of $\triangle x_i^k$ and can be expressed as

$$\mathrm{SI}\left(A_i^k, \triangle x_i^k\right) = \int_{\widehat{x}^k} g_{A_i}(\triangle x_i^k \mid \widehat{x}^k) \cdot \mathrm{SI}\left(\widehat{x}^k, A_i^k\right) d\widehat{x}^k. \tag{12}$$

On the other hand, the soft information passed from variable node A_i to agent node $\widehat{x}$ in the k^{th} iteration is again a Gussian pdf of $\widehat{x}_i^k$ and can be expressed as

$$\mathrm{SI}\left(A_i^k, \widehat{x}^k\right) = \int_{\triangle x_i^k} g_{A_i}(\widehat{x}^k \mid \triangle x_i^k) \cdot \mathrm{SI}\left(\triangle x_i^k, A_i^k\right) d\triangle x_i^k. \tag{13}$$

Note that the left hand sides of (12) and (13) are still Gaussian, because the integration of the product of any two Gaussian pdf's is Gaussian and can be calculated as [5]

$$\int_{-\infty}^{\infty} \mathcal{N}\big(y, -x+\beta, \sigma_a^2\big) \cdot \mathcal{N}\big(x, m_b, \sigma_b^2\big)\, dx \propto \mathcal{N}\big(y, -m_b+\beta, \sigma_a^2+\sigma_b^2\big). \tag{14}$$

Therefore, (12) and (13) can be easily computed via (14). Again, a similar procedure, as that described above for agent node A_i, can be applied to agent node B_i.

4. Agent node C_i: The agent node C_i plays an important role in converting the soft information from the $x-$coordinate to the $y-$coordinate, and vice versa. According to the Pythagorean law, the constraint rule between variable node $\triangle x_i$ and variable node $\triangle y_i$, can be described by a circle,

$$d_i^2 = \left(\triangle x_i^k\right)^2 + \left(\triangle y_i^k\right)^2. \tag{15}$$

In order to keep the soft information passed from agent node C_i to variable node $\triangle y_i$ Gaussian, some approximation has to be made: Let us draw a tangent on the circle (15) with the $x-$coordinate q_x^k of the contact point (q_x^k, q_y^k) being the MS relative location estimate, i.e., the mean value of $\triangle x_i^k$ with a Gaussian pdf from (11). We thus obtain a linear constraint between $\triangle x_i^k$ and $\triangle y_i^k$ to locally approximate (15). The linear equation of the tangent can be written as

$$\triangle y_i^k = \xi \cdot \triangle x_i^k + \left(q_y^k - \xi \cdot q_x^k\right), \tag{16}$$

where $q_y^k = \pm\sqrt{d_i^2 - q_x^k}$ and $\xi = -q_y^k/q_x^k$. Note that, by observing the distribution of $\mathrm{SI}\big(\triangle x_i^k, C_i^k\big)$, we can easily decide whether q_y^k is a positive number or a negative one. Since $\mathrm{SI}\big(\triangle x_i^k, C_i^k\big)$ is Gaussian and (16) is a linear mapping, the soft information passed from agent node C_i to variable node $\triangle y_i$ is also Gaussian and can be expressed as

$$\mathrm{SI}\left(C_i^k, \triangle y_i^{k+1}\right) = \mathcal{N}\big(\triangle y_i^k, q_x^k, \xi^2 \eta_x^k\big), \tag{17}$$

where η_x^k is the variance of the Gaussian soft information SI $\left(\triangle x_i^k, C_i^k\right)$. Similarly, the soft information SI $\left(C_i^k, \triangle x_i^k\right)$ passed from agent node C_i to variable node $\triangle x_i$ can be calculated in a way similar to the procedure above.

In each round of iteration, according to (3), the MS location estimate can be updated by the product of the soft information coming from all the edges connected to the variable nodes $\widehat{x}$ and $\widehat{y}$, SI $\left(\widehat{x}^k\right) = \prod_{j=1}^{N}$ SI $\left(A_j^k, \widehat{x}^k\right)$ and SI $\left(\widehat{y}^k\right) = \prod_{j=1}^{N}$ SI $\left(B_j^k, \widehat{y}^k\right)$. After the iteration procedure converges, the decision on the MS location can be made as the mean value of the Gaussian distributions, SI($\widehat{x}^k$) and SI($\widehat{y}^k$). To highlight the low-complexity characteristics of the proposed system, the required operations for each node in the factor graph are summarized in Table 1. It is clearly shown in Table 1 that only the means and the variances of the incoming Gaussian pdf's need to be processed to produce the mean and variance of the outgoing Gaussian pdf, and only very little calculation is needed in each node. The distributed nature of factor graph also makes the proposed system highly suitable for integrated-circuit implementation.

5. SIMULATION RESULTS

In this section, we demonstrate the performance improvement of the PL system introduced by the proposed algorithm through simulations. For comparison, we also test two existing algorithms, the geometrical algorithm and the geometrical-algorithm-aided TS-LS algorithm (G-TS-LS). Note that the G-TS-LS algorithm uses the solution obtained from the geometrical algorithm, instead of a random coordinate vector of the MS location, as its initial guess in the TS-LS algorithm. On the other hand, we also numerically evaluate two performance bounds, the LS bound and the ML bound derived in [6], to clarify the performance differences among these algorithms. In our simulations, the scenario of cellular system is a standard hexagonal cellular system and the radius of each cell is assumed to be 5 km.

Simulation results in Fig. 3 show that the proposed algorithm not only always outperforms the geometrical algorithm, the G-TS-LS algorithm, and the LS bound, but it also approaches the optimal ML bound. We can also find that if more BSs are considered, the proposed algorithm approaches the ML bound even more closely. This is obviously because more intertwined iterative exchange of soft information helps reduce the suffering from the large measurement errors produced by some BSs.

Table 1. The operations required for the proposed algorithm.

Node	(mean,variance)	
	input	output
A$_i$	(m_i,σ_i^2)	$(-m_i + X_i, \sigma_i^2 + \sigma_{\triangle x_i}^2)$
B$_i$	(m_i,σ_i^2)	$(-m_i + Y_i, \sigma_i^2 + \sigma_{\triangle y_i}^2)$
C$_i$	(m_i,σ_i^2,d_i)	$\left(\pm\sqrt{d_i^2 - m_i^2}, \frac{m_i^2\sigma_i^2}{d_i^2 - m_i^2}\right)$
$\triangle x_i$,$\triangle y_i$	(m_i,σ_i^2)	(m_i,σ_i^2)
$\widehat{x}$,$\widehat{y}$	(m_j,σ_j^2)	$\left(\sigma_i^2 \sum_{j\neq i} \frac{m_j}{\sigma_j^2}, \sigma_i^2 = \frac{1}{\sum_{j\neq i}\frac{1}{\sigma_j^2}}\right)$

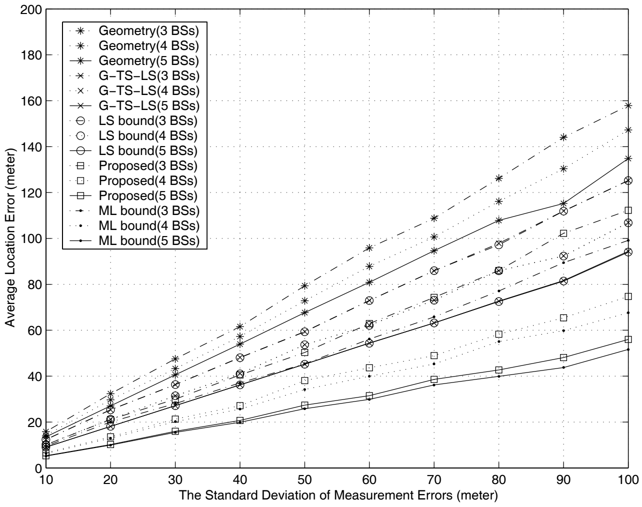

Fig. 3. Performance comparison among various PL algorithms and PL bounds.

6. REFERENCES

[1] Q. Bi, G.L. Zysman, and H. Menkes, "Wireless mobile communications at the start of the 21st century," *IEEE Communications Magazine*, vol. 39, pp. 110-116, Jan. 2001.

[2] J. Caffery, Jr. and G. Stuber, "Subscriber location in CDMA cellular networks," *IEEE Transactions on Vehicluar Technology*, vol. 47, pp. 406-416, May 1998.

[3] W. H. Foy, "Position location solutions by Taylor-Series estimation," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 12, no. 2, pp. 187-193, Mar. 1976.

[4] J. Caffery, "A new approach to the geometry of TOA location," in *IEEE Vehicular Technology Conference*, vol. 4, pp. 1943-1949, Sept. 2000.

[5] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Transactions on Information Theory*, vol. 47, pp. 498-519, Feb. 2001.

[6] J. C. Chen, C. S. Maa, and J. T. Chen, "Network-Based Mobile Position Location Using Factor Graphs," submitted to *IEEE Transactions on Signal Processing*, Oct. 2002.