POSITIONING ALGORITHMS FOR CELLULAR NETWORKS USING TDOA

Christian Mensing and Simon Plass

German Aerospace Center (DLR)
Institute of Communications and Navigation
Oberpfaffenhofen, 82234 Wessling, Germany
E-mail: {christian.mensing, simon.plass}@dlr.de

ABSTRACT

In this paper, we investigate the performance of positioning algorithms in wireless cellular networks based on time difference of arrival (TDoA) measurements provided by the base stations. The localization process of the mobile station results in a non-linear least squares estimation problem which cannot be solved analytically. Therefore, we use iterative algorithms to determine an estimate of the mobile station position. The well-known Gauss-Newton method fails to converge for certain geometric constellations, and thus, it is not suitable for a general solution in cellular networks. Another algorithm is the steepest descent method which has a slow convergence in the final iteration steps. Hence, we apply the Levenberg-Marquardt algorithm as a new approach in the cellular network localization framework. We show that this method meets the best trade-off between accuracy and computational complexity.

1. INTRODUCTION

Positioning in wireless networks became very important in recent years and services and applications based on accurate knowledge of the location of the *Mobile Station* (MS) will play a fundamental role in future wireless systems [1, 2, 3]. In addition to vehicle navigation, fraud detection, resource management, automated billing, and further promising applications, it is stated by the *Federal Communications Commission* (FCC) that all wireless service providers have to deliver the location of all *Enhanced 911* (E911) callers with specified accuracy [2].

MS localization using *Global Navigation Satellite Systems* (GNSSs) such as the *Global Positioning System* (GPS) or the future European *Galileo* system deliver very accurate position information for good environmental conditions. These systems may be a solution for future mass market applications when the problem of high power consumption is resolved and economic aspects are reduced. But nevertheless, the performance loss in urban or indoor areas can be dramatical.

Therefore, in this paper we concentrate on determination of the MS location by exploiting the already available resources of cellular networks. Generally, this localization process is based on measurements in terms of *Time of Arrival* (ToA), *Time Difference of Arrival* (TDoA), *Angle of Arrival* (AoA), and/or *Received Signal Strength* (RSS) [2], provided by the *Base Stations* (BSs). We will focus on processing TDoA measurements which is also part of the 3GPP standard where it is denoted as *Observed TDoA* (OTDoA) [4].

To cope with the resulting non-linear estimation problem where no analytical solution is possible, we describe the iterative *Gauss-Newton* (GN) and the *Steepest Descent* (SD) [1, 5] method as standard solutions for the occurring non-linear least squares problems in the next sections. Finally, we propose the iterative *Levenberg-Marquardt* (LM) algorithm [6, 7] as a new method in the context of positioning in cellular networks. Simulation results show the potential of LM with the best trade-off between accuracy and complexity. Furthermore, a performance comparison with the non-iterative *Chan-Ho* (CH) method [8] is given.

Throughout this paper, vectors and matrices are denoted by lower and upper case bold letters, and random variables are written using *sans serif* font. The matrix $\mathbf{I}_n$ is the $n \times n$ identity matrix, the operation '$\otimes$'denotes the Kronecker product, $\mathrm{E}\{\cdot\}$ expectation, $(\cdot)^{\mathrm{T}}$ transpose, and $\|\cdot\|_2$ the Euclidean norm.

2. SYSTEM MODEL

The time synchronized BSs are organized in a cellular network with cell radius R according to Figure 1. The MS is located at $\boldsymbol{x} = [x, y]^{\mathrm{T}}$ and only the N_{BS} nearest BSs at $\boldsymbol{x}_\nu, \nu \in \{1, 2, \ldots, N_{\mathrm{BS}}\}$ are used for positioning. The distance between the BSs and the MS is given by

$$r_\nu(\boldsymbol{x}) = \|\boldsymbol{x}_\nu - \boldsymbol{x}\|_2 = \sqrt{(x_\nu - x)^2 + (y_\nu - y)^2}. \quad (1)$$

In the following, we treat distances and propagation times as equivalent, and thus, the TDoAs for BS ν w.r.t. BS 1 can be written as

$$d_{\nu,1}(\boldsymbol{x}) = r_\nu(\boldsymbol{x}) - r_1(\boldsymbol{x}), \quad (2)$$

where — without loss of generality — we use BS 1 as the reference BS. The at most $N_{\mathrm{BS}} - 1$ linear independent TDoAs compose the vector

$$\boldsymbol{d}(\boldsymbol{x}) = [d_{2,1}(\boldsymbol{x}), d_{3,1}(\boldsymbol{x}), \ldots, d_{N_{\mathrm{BS}},1}(\boldsymbol{x})]^{\mathrm{T}}, \quad (3)$$

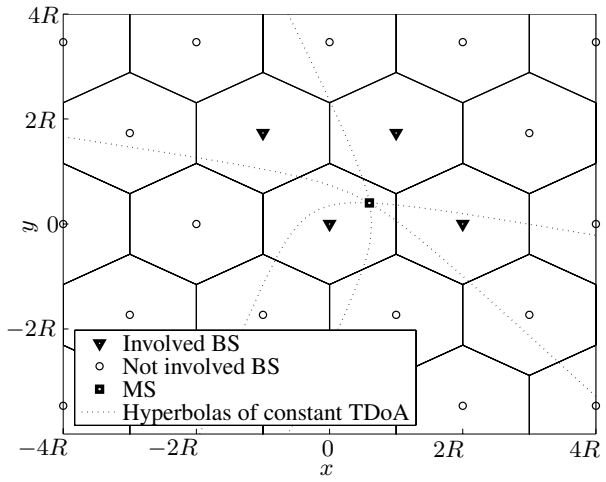

Fig. 1. TDoA positioning in cellular networks, $N_{\mathrm{BS}} = 4$

and the corresponding TDoA measurements are given by

$$\mathbf{d} = \left[\mathsf{d}_{2,1}, \mathsf{d}_{3,1}, \ldots, \mathsf{d}_{N_{\mathrm{BS}},1}\right], \tag{4}$$

based on the measurement model

$$\mathbf{d} = \boldsymbol{d}\left(\boldsymbol{x}\right) + \mathbf{n}, \tag{5}$$

where $\mathbf{n} = \left[\mathsf{n}_{2,1}, \ldots, \mathsf{n}_{N_{\mathrm{BS}},1}\right]^{\mathrm{T}}$ is zero-mean *Additive White Gaussian Noise* (AWGN) with covariance matrix $\boldsymbol{\Sigma}_{\mathbf{n}} = \mathrm{E}\left\{\mathbf{n}\mathbf{n}^{\mathrm{T}}\right\}$.

3. POSITIONING ALGORITHMS

In this section, the considered algorithms for wireless positioning using TDoA are described. With the system model introduced in Section 2 and following the weighted non-linear least squares approach [3, 5] we minimize the cost function

$$\varepsilon\left(\boldsymbol{x}\right) = \left(\mathbf{d} - \boldsymbol{d}\left(\boldsymbol{x}\right)\right)^{\mathrm{T}} \boldsymbol{\Sigma}_{\mathbf{n}}^{-1} \left(\mathbf{d} - \boldsymbol{d}\left(\boldsymbol{x}\right)\right) \tag{6}$$

w.r.t. the unknown MS position $\boldsymbol{x}$ yielding

$$\hat{\boldsymbol{x}} = \underset{\boldsymbol{x}}{\operatorname{argmin}}\, \varepsilon\left(\boldsymbol{x}\right). \tag{7}$$

In the general case, there exists no closed-form solution to the non-linear two-dimensional optimization problem given by (7), and hence, iterative approaches are necessary described in the following.

3.1. Gauss-Newton

The GN algorithm [3, 9] linearizes the system model in (5) about some initial value $\boldsymbol{x}^{(0)}$ yielding

$$\boldsymbol{d}\left(\boldsymbol{x}\right) \approx \boldsymbol{d}\left(\boldsymbol{x}^{(0)}\right) + \boldsymbol{\Phi}\left(\boldsymbol{x}\right)\Big|_{\boldsymbol{x}=\boldsymbol{x}^{(0)}} \left(\boldsymbol{x} - \boldsymbol{x}^{(0)}\right), \tag{8}$$

with the elements of the $(N_{\mathrm{BS}} - 1) \times 2$ Jacobian matrix

$$\boldsymbol{\Phi}\left(\boldsymbol{x}\right) = \nabla_{\boldsymbol{x}}^{\mathrm{T}} \otimes \boldsymbol{d}\left(\boldsymbol{x}\right) = \begin{bmatrix} \frac{x-x_2}{r_2} - \frac{x-x_1}{r_1} & \frac{y-y_2}{r_2} - \frac{y-y_1}{r_1} \\ \frac{x-x_3}{r_3} - \frac{x-x_1}{r_1} & \frac{y-y_3}{r_3} - \frac{y-y_1}{r_1} \\ \vdots & \vdots \\ \frac{x-x_{N_{\mathrm{BS}}}}{r_{N_{\mathrm{BS}}}} - \frac{x-x_1}{r_1} & \frac{y-y_{N_{\mathrm{BS}}}}{r_{N_{\mathrm{BS}}}} - \frac{y-y_1}{r_1} \end{bmatrix}, \tag{9}$$

where $\nabla_{\boldsymbol{x}} = \left[\frac{\partial}{\partial x}, \frac{\partial}{\partial y}\right]^{\mathrm{T}}$. Afterwards, using (8) and (6), the linear least squares procedure is applied resulting in the iterated solution

$$\begin{aligned} \boldsymbol{x}^{(k+1)} &= \boldsymbol{x}^{(k)} + \left(\boldsymbol{\Phi}^{\mathrm{T}}\left(\boldsymbol{x}^{(k)}\right) \boldsymbol{\Sigma}_{\mathbf{n}}^{-1} \boldsymbol{\Phi}\left(\boldsymbol{x}^{(k)}\right)\right)^{-1} \\ &\quad \cdot \boldsymbol{\Phi}^{\mathrm{T}}\left(\boldsymbol{x}^{(k)}\right) \boldsymbol{\Sigma}_{\mathbf{n}}^{-1} \left(\mathbf{d} - \boldsymbol{d}\left(\boldsymbol{x}^{(k)}\right)\right) \\ &= \boldsymbol{x}^{(k)} + \boldsymbol{A}^{(k),-1} \boldsymbol{g}^{(k)}. \end{aligned} \tag{10}$$

The GN algorithm provides very fast convergence and accurate estimates for good initial values. For poor initial values and bad geometric conditions the algorithm results in a rank-deficient, and thus, non-invertible matrix $\boldsymbol{A}^{(k)}$ for certain constellations of MS and BSs. In this case the algorithm diverges.

3.2. Steepest Descent

Contrary to GN, the SD algorithm [3] is a gradient based procedure with search direction $\nabla_{\boldsymbol{x}} \varepsilon\left(\boldsymbol{x}\right)$ and step size μ yielding

$$\begin{aligned} \boldsymbol{x}^{(k+1)} &= \boldsymbol{x}^{(k)} - \mu^{(k)} \boldsymbol{\Phi}^{\mathrm{T}}\left(\boldsymbol{x}^{(k)}\right) \boldsymbol{\Sigma}_{\mathbf{n}}^{-1} \left(\mathbf{d} - \boldsymbol{d}\left(\boldsymbol{x}^{(k)}\right)\right) \\ &= \boldsymbol{x}^{(k)} - \mu^{(k)} \boldsymbol{g}^{(k)}. \end{aligned} \tag{11}$$

The easiest but suboptimum way to find a step size is to choose a constant $\mu^{(k)} = \mu$ for all iteration steps. The optimum step size for each iteration step k can be determined using an optimum line search procedure which is given by the non-linear one-dimensional optimization problem

$$\mu^{(k)} = \underset{\mu}{\operatorname{argmin}}\; \varepsilon\left(\boldsymbol{x}^{(k)} - \mu \boldsymbol{g}^{(k)}\right). \tag{12}$$

We refer to this method as *SD with optimum Line Search* (SDLS). Due to the high computational effort for evaluating (12), we consider this procedure only for performance comparison with the SD method based on a constant step size. Main drawbacks of SD methods are the possibility for running in local minima and slow convergence in the final iteration steps.

3.3. Levenberg-Marquardt

To cope with the problems of GN and SD (robustness and slow convergence, respectively), we present a method introduced by Levenberg [6] and Marquardt [7] for minimizing the cost function (6). We newly propose this algorithm in the

context of positioning problems. It is based on a damped GN procedure given by

$$\begin{aligned}\boldsymbol{x}^{(k+1)} &= \boldsymbol{x}^{(k)} + \left(\boldsymbol{\Phi}^{\mathrm{T}}\left(\boldsymbol{x}^{(k)}\right)\boldsymbol{\Sigma}_{\mathsf{n}}^{-1}\boldsymbol{\Phi}\left(\boldsymbol{x}^{(k)}\right) + \lambda^{(k)}\mathbf{I}_2\right)^{-1} \\ &\quad \cdot \boldsymbol{\Phi}^{\mathrm{T}}\left(\boldsymbol{x}^{(k)}\right)\boldsymbol{\Sigma}_{\mathsf{n}}^{-1}\left(\mathbf{d} - \boldsymbol{d}\left(\boldsymbol{x}^{(k)}\right)\right) \\ &= \boldsymbol{x}^{(k)} + \left(\boldsymbol{A}^{(k)} + \lambda^{(k)}\mathbf{I}_2\right)^{-1}\boldsymbol{g}^{(k)}.\end{aligned} \tag{13}$$

The damping parameter $\lambda^{(k)}$ makes sure that the belonging matrix — in comparison to GN — can always be inverted yielding a much more robust implementation. The damping parameter and the MS location estimate can be calculated using the computational efficient Algorithm 1. It is based on suboptimum line search provided by the additional parameter $\rho^{(k)}$ processing actual and previous estimates. The non-critical initialization parameter τ influences the step size in the first iteration and is set to $\tau = 5$ for the considered scenarios. Note, that for $\lambda^{(k)} = 0$ we obtain the GN solution whereas for $\left|\lambda^{(k)}\right| \gg 1$ we have a SD behavior. LM provides fast convergence and is very robust against inaccurate initial values.

Algorithm 1 Levenberg-Marquardt

$k \leftarrow 0$
$\nu^{(k)} \leftarrow 2$
$\boldsymbol{A}^{(k)} \leftarrow \boldsymbol{\Phi}^{\mathrm{T}}\left(\boldsymbol{x}^{(k)}\right)\boldsymbol{\Sigma}_{\mathsf{n}}^{-1}\boldsymbol{\Phi}\left(\boldsymbol{x}^{(k)}\right)$
$\boldsymbol{g}^{(k)} \leftarrow \boldsymbol{\Phi}^{\mathrm{T}}\left(\boldsymbol{x}^{(k)}\right)\boldsymbol{\Sigma}_{\mathsf{n}}^{-1}\left(\mathbf{d} - \boldsymbol{d}\left(\boldsymbol{x}^{(k)}\right)\right)$
$\lambda^{(k)} \leftarrow \tau \max\left\{\left[\boldsymbol{A}^{(k)}\right]_{i,i}\right\}$
repeat
  $\boldsymbol{h}^{(k)} \leftarrow \left(\boldsymbol{A}^{(k)} + \lambda^{(k)}\mathbf{I}_2\right)^{-1}\boldsymbol{g}^{(k)}$
  $\boldsymbol{x}^{(k+1)} \leftarrow \boldsymbol{x}^{(k)} + \boldsymbol{h}^{(k)}$
  $\rho^{(k)} \leftarrow \dfrac{\varepsilon\left(\boldsymbol{x}^{(k)}\right) - \varepsilon\left(\boldsymbol{x}^{(k+1)}\right)}{\boldsymbol{h}^{(k),\mathbf{T}}\left(\lambda^{(k)}\boldsymbol{h}^{(k)} + \boldsymbol{g}^{(k)}\right)}$
  if $\rho^{(k)} > 0$ **then**
    $\boldsymbol{A}^{(k+1)} \leftarrow \boldsymbol{\Phi}^{\mathrm{T}}\left(\boldsymbol{x}^{(k+1)}\right)\boldsymbol{\Sigma}_{\mathsf{n}}^{-1}\boldsymbol{\Phi}\left(\boldsymbol{x}^{(k+1)}\right)$
    $\boldsymbol{g}^{(k+1)} \leftarrow \boldsymbol{\Phi}^{\mathrm{T}}\left(\boldsymbol{x}^{(k+1)}\right)\boldsymbol{\Sigma}_{\mathsf{n}}^{-1}\left(\mathbf{d} - \boldsymbol{d}\left(\boldsymbol{x}^{(k+1)}\right)\right)$
    $\lambda^{(k+1)} \leftarrow \lambda^{(k)}\max\left\{\frac{1}{3}, 1 - \left(2\rho^{(k)} - 1\right)^3\right\}$
    $\nu^{(k+1)} \leftarrow 2$
  else
    $\boldsymbol{A}^{(k+1)} \leftarrow \boldsymbol{A}^{(k)}$
    $\boldsymbol{g}^{(k+1)} \leftarrow \boldsymbol{g}^{(k)}$
    $\lambda^{(k+1)} \leftarrow \lambda^{(k)}\nu^{(k)}$
    $\nu^{(k+1)} \leftarrow 2\nu^{(k)}$
  end if
  $k \leftarrow k + 1$
until convergence

4. SIMULATION RESULTS

We test the proposed algorithms in a cellular network with cell radius $R = 3\,\mathrm{km}$ and assume constant noise power for all involved links from the BSs to the MS, i.e., $\boldsymbol{\Sigma}_{\mathsf{n}} = \sigma_{\mathsf{n}}^2\mathbf{I}_{N_{\mathrm{BS}}-1}$.

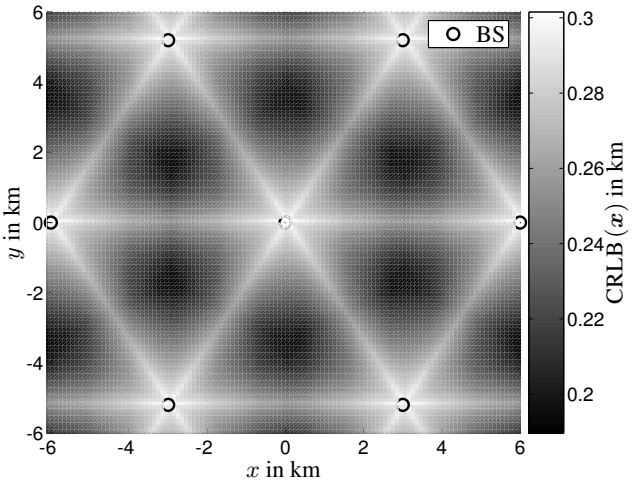

Fig. 2. CRLB $(\boldsymbol{x})$ for $\sigma_{\mathsf{n}} = 0.2\,\mathrm{km}$, $R = 3\,\mathrm{km}$, $N_{\mathrm{BS}} = 3$

The performance bound is given by the *Cramer-Rao Lower Bound* (CRLB) [5] defined as

$$\mathrm{CRLB}\left(\boldsymbol{x}\right) = \sqrt{\mathrm{trace}\left(\boldsymbol{\Phi}^{\mathrm{T}}\left(\boldsymbol{x}\right)\boldsymbol{\Sigma}_{\mathsf{n}}^{-1}\boldsymbol{\Phi}\left(\boldsymbol{x}\right)\right)^{-1}} \tag{14}$$

for each MS position. Figure 2 shows CRLB $(\boldsymbol{x})$ using $N_{\mathrm{BS}} = 3$ for positioning. It can be seen, that near the BSs and on the direct links of neighboring BSs the positioning performance is restricted due to geometric constellation. Inside the polygon spanned by the involved BSs we can expect more accurate results. Clearly, for increasing N_{BS} the geometric conditions will change.

Nevertheless, we are interested in the positioning accuracy for all possible MS locations in the cellular network. Thus, we introduce $\overline{\mathrm{CRLB}} = \mathrm{E}_{\boldsymbol{x}}\left\{\mathrm{CRLB}\left(\boldsymbol{x}\right)\right\}$ as mean value of the bound for the whole network. We compare this bound with the achievable *Root Mean Square Error* (RMSE) for the algorithms defined as

$$\mathrm{RMSE} = \sqrt{\mathrm{E}_{\boldsymbol{x}}\left\{\left\|\boldsymbol{x} - \hat{\boldsymbol{x}}\right\|_2^2\right\}} \geq \overline{\mathrm{CRLB}} \tag{15}$$

and averaged over several MS positions and noise realizations, where $\hat{\boldsymbol{x}} = \boldsymbol{x}^{(K)}$ is the estimate provided by the algorithms after K iteration steps.

In Figure 3 the performance of the proposed algorithms is analyzed for $N_{\mathrm{BS}} = 3$ and $\sigma_{\mathsf{n}} = 0.2\,\mathrm{km}$. Initial value for the iterative algorithms is the mean value of the positions of all involved BSs, i.e.

$$\boldsymbol{x}^{(0)} = \frac{1}{N_{\mathrm{BS}}}\sum_{\nu=1}^{N_{\mathrm{BS}}}\boldsymbol{x}_\nu. \tag{16}$$

GN diverges for bad geometric conditions (cf. Figure 2) and poor initial values. Therefore, in these cases the resulting estimate is set to $\hat{\boldsymbol{x}} = \boldsymbol{x}^{(0)}$ to show the loss w.r.t. $\overline{\mathrm{CRLB}}$. Anyway, for perfect conditions GN provides the fastest convergence. The LM algorithm converges after $K_{\mathrm{LM}} = 5$ iteration steps and reaches nearly $\overline{\mathrm{CRLB}}$. SD with $\mu = 0.48$

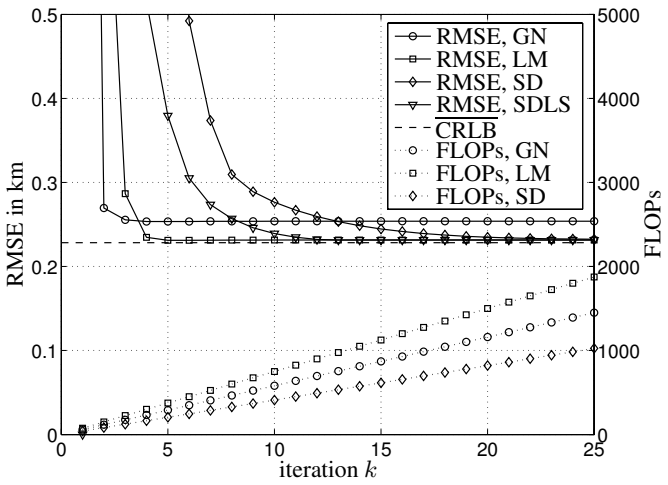

Fig. 3. RMSE and FLOPs vs. iterations for $\sigma_{\mathrm{n}} = 0.2\,\mathrm{km}$, $R = 3\,\mathrm{km}$, $N_{\mathrm{BS}} = 3$

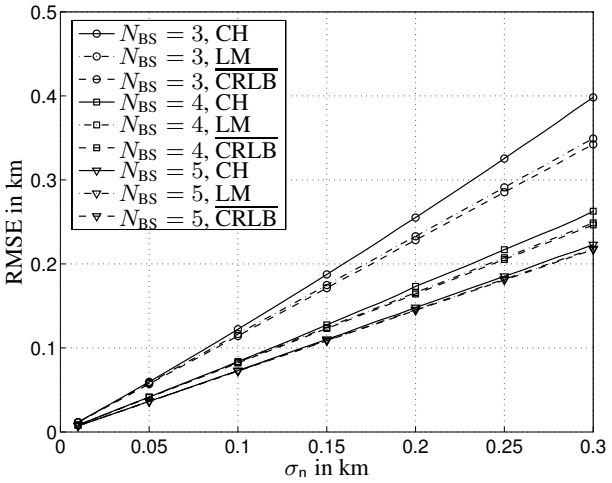

Fig. 4. RMSE vs. σ_{n} for LM and CH algorithm, $R = 3\,\mathrm{km}$

and even SDLS have a much slower convergence compared to LM and no improvement can be achieved after $K_{\mathrm{SD}} = 22$ and $K_{\mathrm{SDLS}} = 12$ iterations. Additionally, in Figure 3 the needed *FLoating point OPerations* (FLOPs) as measure for computational complexity are depicted for the algorithms. Obviously, LM offers the best trade-off between performance and complexity. Although the number of FLOPs for each iteration step using LM is higher than for using SD, the FLOPs for delivering one MS estimate are much smaller due to the reduced number of required iterations until convergence. For this scenario, we achieve about 60% reduction of complexity. Note, that the FLOPs for LM are based on a worst case scenario with $\rho^{(k)} > 0$ (cf. Algorithm 1) for all iteration steps.

Finally, Figure 4 shows a performance comparison of the LM algorithm for various numbers of involved BSs. It can be seen, that the deviation from $\overline{\mathrm{CRLB}}$ is very small, even for $N_{\mathrm{BS}} = 3$ and high noise power. The results are compared with the non-iterative algorithm invented by Chan and Ho [8]. This is a three-step procedure achieving $\overline{\mathrm{CRLB}}$ for low noise power, but with restricted accuracy for higher noise power. The number of required FLOPs is slightly smaller than the FLOPs for LM with $K_{\mathrm{LM}} = 5$, but we can observe that the performance of the *Chan-Ho* (CH) method — especially in the most interesting case of $N_{\mathrm{BS}} = 3$ — is considerably worse.

5. CONCLUSIONS

In this paper, we investigated the performance of mobile station positioning in cellular networks using TDoA measurements. The standard Gauss-Newton algorithm diverges for inaccurate initial values and the steepest descent method has a poor convergence behavior in the final iteration steps. To avoid these drawbacks, we propose to use the Levenberg-Marquardt algorithm as a new approach in the positioning framework. Simulation results show that this method is suitable to estimate the mobile station location with high accuracy and low complexity. In the case of 3 involved base stations for positioning Levenberg-Marquardt is very close to the Cramer-Rao lower bound and a complexity reduction of about 60% can be achieved.

6. REFERENCES

[1] J. J. Caffery, *Wireless Location in CDMA Cellular Radio Systems*, Kluwer Academic Publishers, 2000.

[2] A. H. Sayed, A. Tarighat, and N. Khajehnouri, "Network-Based Wireless Location," *IEEE Signal Processing Magazine*, vol. 22, no. 4, pp. 24–40, July 2005.

[3] F. Gustafsson and F. Gunnarsson, "Mobile Positioning Using Wireless Networks," *IEEE Signal Processing Magazine*, vol. 22, no. 4, pp. 41–53, July 2005.

[4] Y. Zhao, "Standardization of Mobile Phone Positioning for 3G Systems," *IEEE Communications Magazine*, vol. 40, no. 7, pp. 108–116, July 2002.

[5] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*, Prentice Hall, 1993.

[6] K. Levenberg, "A Method for the Solution of Certain Problems in Least Squares," *Quarterly of Applied Mathematics*, vol. 2, pp. 164–168, 1944.

[7] D. Marquardt, "An Algorithm for Least Squares Estimation on Nonlinear Parameters," *SIAM Journal on Applied Mathematics*, vol. 11, pp. 431–441, 1963.

[8] Y. T. Chan and K. C. Ho, "A Simple and Efficient Estimator for Hyperbolic Location," *IEEE Transactions on Signal Processing*, pp. 1905–1915, August 1994.

[9] W. Foy, "Position-Location Solutions by Taylor-Series Estimation," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 12, pp. 187–193, March 1976.