POLITECNICO DI TORINO
Repository ISTITUZIONALE

Pseudorange and Doppler-Based Positioning: Enabling Convergence of Least-Squares Estimation from MEO to LEO

Original
Pseudorange and Doppler-Based Positioning: Enabling Convergence of Least-Squares Estimation from MEO to LEO / Morichi, L., Zocca, S., Minetto, A., Nardin, A., Dovis, F.. - In: NAVIGATION. - ISSN 0028-1522. - ELETTRONICO. - 73:1(2026). [10.33012/navi.734]

Availability:
This version is available at: 11583/3007129 since: 2026-01-30T18:21:56Z

Publisher:
Institute of Navigation (ION)

Published
DOI:10.33012/navi.734

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

(Article begins on next page)

19 November 2026

Received: 10 May 2024 | Revised: 22 August 2025 | Accepted: 6 October 2025

DOI: 10.33012/navi.734

REGULAR PAPERS

Pseudorange and Doppler-Based Positioning: Enabling Convergence of Least-Squares Estimation from MEO to LEO

Luca Morichi* | **Simone Zocca** | **Alex Minetto** | **Andrea Nardin** | **Fabio Dovis**

[1] Department of Electronics and Telecommunications (DET)
Politecnico di Torino, Turin, Italy

Correspondence
*Luca Morichi
Department of Electronics and Telecommunications (DET), Politecnico di Torino
Email: luca.morichi@polito.it

Abstract

The growing interest of the space industry in satellite systems within the low Earth orbit (LEO) region has prompted attention to their potential for positioning, navigation, and timing applications. This study addresses the convergence issue highlighted in the literature when the Gauss–Newton (GN) method is applied to least-squares (LS) position estimation algorithms in LEO scenarios, with analyses conducted independently for pseudorange and Doppler shift measurements. To address these limitations, this paper examines two line-search techniques in combination with the GN method. A comprehensive analysis of the LS method is conducted through tests on satellite constellations at various orbital altitudes, from medium Earth orbits to LEOs. The results, evaluated in terms of the number of iterations required to achieve convergence, show that adjusting the GN step using a damping factor, namely the damped GN factor, effectively resolves convergence issues, even in LEO scenarios. In particular, the proposed algorithm consistently converges in the LEO region within an average of seven iterations.

Keywords

damped Gauss–Newton, Doppler positioning, least-squares method, LEO-PNT, line-search methods, pseudorange positioning

1 | INTRODUCTION AND MOTIVATION

In recent years, there has been a notable increase in investment from both the academic research and space industry sectors focused on applications revolving around low Earth orbit (LEO) satellite systems. Currently, such systems provide a diverse range of services, encompassing broadband connectivity, applications for the internet of things, Earth observations, and synthetic aperture radar (Kassas, 2021; Prol et al., 2022; Reid et al., 2021). In addition to these applications, LEO satellite constellations are also appealing for positioning, navigation, and timing (PNT) applications. While PNT applications have traditionally relied on medium Earth orbit (MEO) satellite systems since the advent of the Global Positioning System (GPS), there is increasing interest in the development of LEO-PNT systems. This transition holds promise for mitigating some of the limitations inherent when a global navigation satellite system (GNSS) is operating with satellites at MEO.

Licensed under CC-BY 4.0 © 2026 Institute of Navigation

Using LEO satellites have some advantages, owing to their proximity to the Earth. This proximity translates into reduced signal path loss and potentially improved signal quality. In addition, the higher velocity of LEO satellites compared with their MEO counterparts contributes to greater multipath decorrelation and faster resolution of carrier-phase integer ambiguities (Reid et al., 2021). However, this proximity also presents challenges in terms of coverage. LEO satellites generally have a smaller footprint than MEO satellites, requiring a greater number of satellites to achieve coverage comparable to that of GNSS satellites (Prol et al., 2022; Reid et al., 2021).

Current research in the field of LEO-PNT has focused on GNSS augmentation (Zeng et al., 2019; Zhang et al., 2023), meta-signals (Minetto et al., 2024; Nardin et al., 2021, 2024), and independent LEO-PNT service, among others. In particular, the latter application, also known as stand-alone LEO-PNT, can be achieved either by developing systems designed and optimized for PNT services (Çelikbilek et al., 2022; Prol et al., 2022) or by adopting an opportunistic approach. For this, one must utilize signals of opportunity (SoOs) from satellites not designed for PNT services and extract measurements that are useful for positioning.

Among various measurements, the Doppler shift has attracted increasing attention among the scientific community for LEO-PNT applications. The Doppler shift can be harnessed either through SoOs, as it is relatively easy to extract from various modulated signals (Khalife et al., 2021; Kozhaya & Kassas, 2023; Psiaki & Slosman, 2019), or in stand-alone systems. With the velocity provided by LEO satellites and the possible higher carrier frequencies carrying their transmitted signals, the Doppler positioning paradigm, reminiscent of the Transit system (Kershner & Newton, 1962), is gaining momentum. Moreover, an approach based exclusively on Doppler measurements presents the additional benefit of reduced signal bandwidth, as it does not necessarily require the presence of ranging codes (van Graas, 2023).

Researchers have investigated various receiver architectures to extract and process measurements, leveraging SoOs (Farhangian & Landry, 2020; Khalife & Kassas, 2019; Orabi et al., 2021). Several algorithms have also been proposed for performing receiver state estimation; the main PNT estimators leverage maximum likelihood and maximum *a posteriori* paradigms through least-squares (LS) methods and Bayesian estimators such as Kalman filter (KF) and particle filter (PF) estimators (Prol et al., 2022), respectively. LS methods still constitute the baseline for absolute state estimation and for initializing the aforementioned Bayesian estimation algorithms. Moreover, LS approaches have been shown to be effective in LEO scenarios, as highlighted by Hartnett (2022), Morichi et al. (2024), Psiaki (2021), and Shi et al. (2023).

This paper will focus on an LS method, approaching it from the perspective of convergence and placing special emphasis on its ability to converge in LEO satellite constellation scenarios. The receiver state estimation problem solved through the LS method ultimately reduces to an optimization problem, where the goal is to find a minimum of the objective function. This problem is generally tackled using Newton's method or one of its variants. These methods are iterative, meaning that the minimum of the objective function is estimated by starting from an initial guess, which is then refined at each iteration (Björck, 1996).

The selection of this initial estimate is crucial, as it can significantly impact the final convergence of the algorithms (Yan et al., 2013). In GNSS positioning problems, it is common practice to place this estimate at the center of the Earth, as this represents a neutral point regardless of the receiver's position on Earth. However, as discussed by Shi et al. (2023), when dealing with LEO satellites, the LS algorithm becomes more sensitive to the initial guess, and placing the estimate at the

center of the Earth does not necessarily guarantee convergence. For this reason, in much of the literature addressing these scenarios, the algorithm is initialized within a given radius in the proximity of the receiver. However, it is worth noting that this approach significantly diverges from the traditional methodology utilized in GNSSs for maximum likelihood estimations, as it depends on the availability of prior information on the receiver state. Moreover, it is important to note that convergence issues are not isolated to LS methods based on Doppler shift measurements, but also arise for those based on pseudorange (Van Uytsel et al., 2023). In a previous study (Morichi et al., 2024), we also identified a convergence problem with the LS method based on both measurements. In that paper, we conducted an investigation to determine the maximum distance from the ground truth of the receiver at which the algorithm can be initialized to guarantee convergence.

Unlike previous approaches, in this work, we initialize the estimate at the Earth's center and achieve convergence by directly modifying the algorithm. By applying numerical approximation methods, we adapt the Gauss–Newton (GN) method to ensure convergence with LEO satellites, without relying on *a priori* information about the receiver state.

This paper proposes two algorithms: one based on the damped GN (DGN) method, denoted as the adaptive DGN (ADGN) method, and another method that incorporates heuristics based on Wolfe's conditions, called adaptive Wolfe DGN (AW-DGN) (Dennis Jr & Schnabel, 1996; Gill et al., 2019). The performance of these two algorithms is evaluated based on the number of iterations required to achieve convergence, and their effectiveness is validated by comparing the obtained solution with the receiver's ground truth. As the objective of this study is to enable convergence in receiver state estimation, an analysis of positioning accuracy and precision is considered beyond the scope of this article. The solutions identified by the algorithm are dependent on the objective function being minimized, rather than the approach used to reach the minimum. The presented algorithms offer the same accuracy performance as other LS-based methods, whose performance for LEO satellites has already been discussed in previous studies (Morichi et al., 2024; Psiaki, 2021; Shi et al., 2023).

This article is organized as follows. In Section 2, we provide an overview of the mathematical background of the LS problem, including Newton numerical methods typically employed in the scientific literature to address the LS problem. In Section 3, we introduce the simulation methodology used to investigate various orbital altitudes and present the pseudocode of the ADGN algorithm, along with respective results. Subsequently, we introduce Wolfe's conditions and present the AW-DGN algorithm, along with its convergence analysis. We then conclude the section by providing the pseudocode for the best-performing algorithm. In Section 4, we further test this algorithm across different scenarios, considering various satellite geometries and receiver locations. Finally, in Section 5, we draw conclusions.

2 | LS AND NEWTON'S METHODS

2.1 | The LS Problem

In its most general form, solving an LS problem involves addressing the following minimization problem (Dennis & Schnabel, 1996; Quarteroni et al., 2014):

$$\min_{\mathbf{x}\in\mathbb{R}^n} \Phi(\mathbf{x}) \quad \text{with} \quad \Phi(\mathbf{x}) = \frac{1}{2}\|\mathbf{R}(\mathbf{x})\|_2^2 = \frac{1}{2}\sum_{i=1}^{m} r_i^2(\mathbf{x}) = \frac{1}{2}\sum_{i=1}^{m}\left[y_i - f_i(\mathbf{x})\right]^2 \tag{1}$$

where $\mathbf{x}$ is a decision variable vector in $\mathbb{R}^n$, containing the variables to be optimized to minimize the objective function $\Phi(\mathbf{x})$. The notation $||\mathbf{R}(\mathbf{x})||_2^2$ denotes the squared ℓ^2-norm of the function $\mathbf{R}(\mathbf{x}) = \left(r_1(\mathbf{x}), \cdots, r_m(\mathbf{x})\right)^\top$ that contains the residuals $r_i = y_i - f_i(\mathbf{x})$. These residuals represent the difference between the observed value y_i and the value predicted by the mathematical model $f_i(\mathbf{x})$ serving as a metric for the discrepancy between the estimated parameters and observed values. The objective of the LS method is to minimize the sum of squares of the residuals, thereby identifying the parameters that most accurately fit the observations. If the function $f_i(\mathbf{x})$ exhibits a linear relationship with the variable $\mathbf{x}$, the problem in Equation (1) is referred to as a linear LS problem; conversely, if there exists a nonlinear relation, the problem will be classified as a nonlinear LS problem. It is important to note that Equation (1) relies on the assumption that the observation noise is independent and identically distributed, an assumption that has been adopted, without a loss of generality, in this work.

In radionavigation and radiolocalization problems as well as in the processing of GNSS observables, the LS is commonly employed to estimate the receiver or target state. In this context, the term $\mathbf{x}$ in Equation (1) represents the state to be estimated, which can be written in vector form as follows:

$$\mathbf{x} = \begin{bmatrix} \mathbf{r}_u \\ \mathbf{v}_u \\ \tau_u \end{bmatrix} \tag{2}$$

where $\mathbf{r}_u$ denotes a 3×1 vector indicating the user's position relative to a specific reference frame, $\mathbf{v}_u$ represents a 3×1 vector denoting the user's velocity, and $\tau_u = \left[\delta t_u \delta \dot{t}_u\right]^\top$ is a 2×1 vector containing an unknown local clock offset and drift relative to a specified time scale. Specifically, the clock bias δt_u adjusts for the time offset between the user and the GNSS time scale, whereas the clock drift $\delta \dot{t}_u$ determines whether the local oscillator operates at a slower or faster frequency compared with the GNSS system time.

The residuals depend on both the observations and the mathematical model employed in the estimation problem. In this paper, we focus on pseudorange and Doppler shift observations, using the generic observable notation y_i for both cases to ensure uniformity, although the two types of measurements are treated separately throughout the analysis. In the case of pseudorange observations, $y_i = \rho_i$ denotes pseudorange measurements from the *i*-th satellite, and f_i represents the pseudorange model, expressed as follows:

$$f_i(\mathbf{x}) = ||\mathbf{r}_i - \mathbf{r}_u|| + c\delta t_u \tag{3}$$

where $\mathbf{r}_i$ is a 3×1 vector indicating the position of the *i*-th satellite and c is the speed of light.

In the case of Doppler shift observations, $y_i = \dot{\rho}_i$ represents the Doppler shift measurements associated with the signal arriving from the *i*-th satellite, and f_i denotes the Doppler shift model, mathematically described as follows:

$$f_i(\mathbf{x}) = -\frac{1}{\lambda}\left[\left(\mathbf{v}_i - \mathbf{v}_u\right)\cdot\frac{\mathbf{r}_i - \mathbf{r}_u}{||\mathbf{r}_i - \mathbf{r}_u||} + c\delta \dot{t}_u\right] \tag{4}$$

where $\mathbf{v}_i$ is a 3×1 vector indicating the velocity of the *i*-th satellite and λ is the wavelength of the transmitted signal. From Equations (3) and (4), it is evident that

a nonlinear relationship exists between the observation and the receiver state. Therefore, our focus shifts to mathematical methods tailored to address such nonlinearity. Whereas linear equations allow direct solutions, the nonlinearity of $\Phi(\mathbf{x})$ makes closed-form solutions unfeasible. Therefore, we will investigate iterative techniques, such as Newton's method, to approximate the solutions.

2.2 | Newton Methods to Solve LS Problems

Let us suppose that we want to compute the zeros of a generic function $\Lambda(\mathbf{x})$. Newton's method is a numerical technique that leverages the information provided by the function's derivative to iteratively approximate the zeros. The process begins with an initial guess for the root, which is progressively refined at each iteration through the application of a correction term, $\delta\mathbf{x}$, to improve the approximation. This correction term is calculated as the ratio of the function $\Lambda(\mathbf{x})$ to its gradient $\nabla\Lambda(\mathbf{x})$ and is applied to the current estimate $\mathbf{x}^{(k)}$, guiding the algorithm toward a more accurate solution, as shown in the following equation:

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \delta\mathbf{x}^{(k)} \quad \text{with} \quad \delta\mathbf{x}^{(k)} = \frac{\Lambda\left(\mathbf{x}^{(k)}\right)}{\nabla\Lambda\left(\mathbf{x}^{(k)}\right)} \tag{5}$$

where the superscript k represents the iteration index. To address the LS problem outlined in Equation (1), it is necessary to examine the stationary points of the objective function $\Phi(\mathbf{x})$. This process reduces the problem to solving the equation $\nabla\Phi(\mathbf{x}) = 0$. It is precisely in looking for the zeros of the gradient of the objective function that Newton's method proves to be a viable option. When the model in Equation (5) is fitted to our scenario, it becomes evident that $\Lambda(\mathbf{x}) = \nabla\Phi(\mathbf{x})$ and that $\nabla\Lambda(\mathbf{x}) = \nabla^2\Phi(\mathbf{x})$, corresponding to the Hessian of the objective function $\Phi(\mathbf{x})$. In light of these considerations, it is possible to reformulate Equation (5) as follows:

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \underbrace{\boxed{\frac{\nabla\Phi\left(\mathbf{x}^{(k)}\right)}{\nabla^2\Phi\left(\mathbf{x}^{(k)}\right)}}}_{\delta\mathbf{x}^{(k)}} \tag{6}$$

where $\nabla\Phi\left(\mathbf{x}^{(k)}\right)$ and $\nabla^2\Phi\left(\mathbf{x}^{(k)}\right)$ can be further developed as follows (Gander et al., 2014; Quarteroni et al., 2014):

$$\nabla\Phi\left(\mathbf{x}^{(k)}\right) = -\mathbb{J}_{\mathbf{R}}\left(\mathbf{x}^{(k)}\right)^{\top}\mathbf{R}\left(\mathbf{x}^{(k)}\right) = -\mathbb{J}_{\mathbf{f}}\left(\mathbf{x}^{(k)}\right)^{\top}\mathbf{R}\left(\mathbf{x}^{(k)}\right) \tag{7a}$$

$$\nabla^2\Phi\left(\mathbf{x}^{(k)}\right) = \mathbb{J}_{\mathbf{f}}\left(\mathbf{x}^{(k)}\right)^{\top}\mathbb{J}_{\mathbf{f}}\left(\mathbf{x}^{(k)}\right) + \sum_{i=1}^{m}\frac{\partial\, r_i\left(\mathbf{x}^{(k)}\right)}{\partial\, x_l x_j} r_l\left(\mathbf{x}^{(k)}\right), \quad l, j = 1, \cdots, n \tag{7b}$$

Here, the superscript $\top$ denotes the transpose operator. $\mathbb{J}_f$ represents the Jacobian matrix, where each row contains the gradient vector of the corresponding function f_i with respect to the variables of the vector $\mathbf{x}$. Regarding the Hessian matrix in Equation (7b), it is noteworthy that its expression includes two terms. The first term, which is usually the most significant, depends solely on the Jacobian matrix, whereas the second term consists of second-order derivatives multiplied by the residuals. When the model closely approximates the solution, resulting in small residuals, the influence of this second term decreases compared with the first. In

addition, computing $m \times n^2$ derivatives for the second term can be computationally demanding, especially as the dimension of the state increases. Consequently, in practical applications, it is common practice to approximate or even neglect the second term. In particular, the GN method is a well-known method that neglects the second term of Equation (7b), approximating the entire matrix with only the term related to the Jacobian matrix ($\nabla^2\Phi(\mathbf{x}) \approx \mathbb{J}_{\mathbf{f}}(\mathbf{x})^\top \mathbb{J}_{\mathbf{f}}(\mathbf{x})$). For this variant, the formulation in Equation (5) can be rewritten as follows:

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} \underbrace{\boxed{-\frac{\mathbb{J}_{\mathbf{f}}\left(\mathbf{x}^{(k)}\right)^\top \mathbf{R}\left(\mathbf{x}^{(k)}\right)}{\mathbb{J}_{\mathbf{f}}\left(\mathbf{x}^{(k)}\right)^\top \mathbb{J}_{\mathbf{f}}\left(\mathbf{x}^{(k)}\right)}}}_{\delta\mathbf{x}^{(k)}} \tag{8}$$

The objective is to generate a sequence of estimates $\mathbf{x}^{(k)}$ such that the following holds:

$$\lim_{k\to\infty} \mathbf{x}^{(k)} = \alpha \tag{9}$$

where $\alpha \in \mathbb{R}^n$ and $\nabla\Phi(\alpha) = 0$ (Quarteroni et al., 2010). The sequence $\{\mathbf{x}^{(k)}\}$ is said to converge to α if, for each pair of successive iterations $(\mathbf{x}^{(k)}, \mathbf{x}^{(k+1)})$, the approximation error decreases according to $\left|\mathbf{x}^{(k+1)} - \alpha\right| \leq C\left|\mathbf{x}^{(k)} - \alpha\right|$, where $C \geq 0$. When the difference between two successive estimates falls below a predefined tolerance or threshold, ϵ, the algorithm is considered to have reached convergence, and thus, the iterations are halted (Quarteroni et al., 2014). Mathematically, this condition can be expressed as follows:

$$\left|\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}\right| \leq \epsilon \tag{10}$$

If the threshold is properly chosen, further iterations would not bring significant improvements. The number of iterations required to reach convergence and the actual success of convergence depend on the properties of the objective function, the choice of the initial point $\mathbf{x}^{(0)}$, and the satisfaction of certain regularity conditions (Bierlaire, 2021; Hager, 1988; Nocedal & Wright, 1999; Quarteroni et al., 2014; Yan et al., 2013). These conditions include ensuring that the Jacobian matrix has full rank, that the neglected second-order terms in Equation (7) are either zero or small in magnitude, and that the residuals are small, particularly close to the minimum of the objective function. If any of these conditions are not met, convergence is no longer guaranteed, and the GN method may stagnate, diverge, or converge to a non-stationary point. One commonly employed technique for enhancing the likelihood of convergence is to utilize the GN method in conjunction with a line-search approach. This approach, known as the DGN method (Björck, 1996; Dennis Jr & Schnabel, 1996), involves introducing a multiplicative factor that dampens the Newton step ($\delta\mathbf{x}$). The correction in Equation (8) is thus replaced by the following:

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \mu^{(k)}\delta\mathbf{x}^{(k)} \tag{11}$$

The damping factor μ determines the distance traversed along the direction $\delta\mathbf{x}$. Typically, the value of this factor ranges from 0 to 1. Intuitively, smaller values of μ increase the number of iterations needed for the algorithm to converge, but increase stability, thereby avoiding overshooting the minimum we seek to reach.

Conversely, larger values of μ can speed up convergence but may introduce instability that can lead to divergence. When $\mu = 1$, the method reverts to the GN approach in Equation (8).

Theoretically, an optimal $\mu^{(k)}$ value can be found by an exact line-search technique, which involves solving the following minimization problem:

$$\mu^{(k)} = \underset{\mu \in \mathbb{R}_0^+}{\arg\min}\, \Phi\left(\mathbf{x}^{(k)} + \mu\, \delta \mathbf{x}^{(k)}\right) \tag{12}$$

In the presence of multiple minima, it is common practice to select the first minimum; that is, $\mu^{(k)}$ is the smallest element of $\underset{\mu \in \mathbb{R}_0^+}{\arg\min}\, \Phi\left(\mathbf{x}^{(k)} + \mu\, \delta \mathbf{x}^{(k)}\right)$ (Bierlaire, 2021).

In practical applications, however, it can be cumbersome to invest extensive effort in solving an additional minimization problem. Instead, an inexact line search based on a trial-and-error approach is often preferred, where different damping factors are tested and the first suitable factor is selected. In the following sections, we will introduce algorithms that implement different methods for selecting the damping factor. Our goal is to demonstrate their potential to enable convergence to the global minimum of the objective function when applied to the position, velocity, and time algorithm based on the LS method. This assessment is particularly important under challenging convergence conditions, such as those encountered in LEO scenarios.

3 | METHODOLOGY AND CONVERGENCE PERFORMANCE

3.1 | Simulation Framework

To assess the convergence of the LS method using pseudorange or Doppler shift measurements, we use a method of investigation called "orbit scale down," which we introduced in a previous work (Morichi et al., 2024). This theoretical method allows the algorithm to be evaluated by simulating variations in the orbital radius of the satellites used for the correction. In the simulations, the satellites were progressively lowered along the line-of-sight (LOS) vector connecting the receiver to the satellite, ensuring that the direction of the receiver–satellite steering vector remains fixed, as shown in Figure 1.

A total of 100 orbital altitudes were considered in the simulations, with 60 distributed in the LEO region between 160 and 2000 km and 40 in the MEO region up to the orbital altitude of the GPS satellites (20200 km). This approach allowed a trend to be outlined for convergence in the two regions. With each change in the orbital radius of the satellites, both their position and velocity are consistently adjusted. Specifically, by assuming circular orbits, the satellites' velocity can be adapted based on the orbital period, derived from Kepler's third law, which is given by the following:

$$T_s \approx 2\pi \sqrt{\frac{R_j^3}{GM}} \tag{13}$$

where R_j is the orbital radius of the satellite, M is the mass of the Earth, and G is the universal gravitational constant. From Equation (13), the satellite's velocity for a given orbital radius can be derived. Furthermore, knowing both the initial orbital

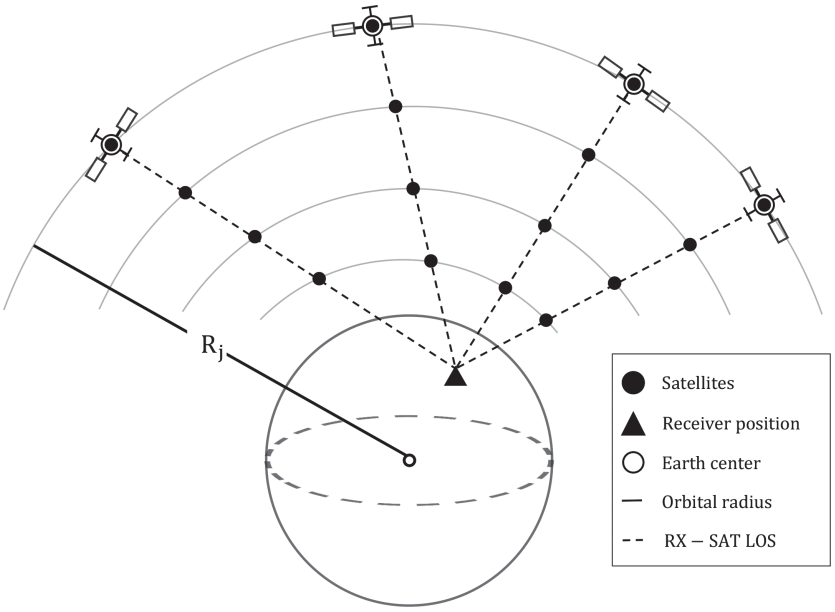


FIGURE 1 Pictorial view of the orbit scale-down methodology (Morichi et al., 2024); RX: receiver

radius and velocity of the satellite allows us to compute the satellite velocity at a new orbital radius as follows:

$$\begin{cases} v_{s_1} = \sqrt{\dfrac{GM}{R_{s_1}}} \\ v_{s_2} = \sqrt{\dfrac{GM}{R_{s_2}}} \end{cases} \Rightarrow \quad v_{s_2} = v_{s_1}\sqrt{\frac{R_{s_1}}{R_{s_2}}} \tag{14}$$

Given that we are working with synthetic satellite positions and velocities, the measurements used by the LS algorithm must also be adapted. Specifically, both the pseudorange and Doppler shift measurements, originally derived from real observables, must be adjusted accordingly.

To achieve these adjustments, synthetic, ideal measurements were constructed based on the receiver's ground truth. To make these measurements more realistic, zero-mean Gaussian noise, independent for each satellite, was added. The variance of this noise corresponds to that of the original measurements obtained when the satellites were at their true positions. In the subsequent analyses, we did not model any other types of errors. The focus of this study is to develop an algorithm capable of guiding the estimation of the receiver state toward the global minimum of the objective function, even in scenarios where the traditional GN-based positioning algorithm fails. Provided that the regularity conditions of the GN method are satisfied, the presence of errors and distortions in the measurements may affect the rate of convergence but does not hinder overall convergence. The minimum is still reached, although it may be shifted from its nominal location. As a result, omitting these errors does not compromise the generality of our findings, as any convergence issues observed can be attributed solely to the algorithm and the geometry of the analyzed scenario.

The effectiveness of the proposed algorithm was initially tested using a static receiver located in Turin, Italy (45.0653° N, 7.6589° E, 321.11 m). We started with real data collected through an antenna and front-end (i.e., Ettus Research

USRP N210), which were then processed with a software receiver implemented in MATLAB. The real scenario involved the use of five GPS satellites for the correction. Based on the position and velocity information of these five satellites, obtained via post-processing of the collected data, we applied the orbit scale-down framework presented above. This approach allowed us to calculate the position and velocity of the same five satellites at different orbital altitudes. Unlike the approach described by Morichi et al. (2024), where the algorithm's convergence relied on gradually adjusting the initialization points of the GN method as the satellite's orbital altitude decreased, in this case, no prior knowledge about the receiver position is assumed. Consequently, the initial estimate of the receiver state is set to all zeros, as in conventional approaches, meaning that the position state is placed at the center of the Earth and all other components related to the bias (or drift) of the receiver clock are assumed to be zero.

3.2 | ADGN Algorithm

To enhance the resilience of the GN-based positioning algorithm to convergence issues, we propose utilizing the DGN method. This section presents the algorithm that we have developed, which is based on the theoretical framework outlined in Section 2. The pseudocode for the ADGN method, which is our adaptation of the GN method, is shown in Algorithm 1. This method is adaptive in that it introduces a damping factor, μ, that is dynamically adjusted: the damping factor increases when the algorithm converges and decreases when the algorithm does not converge.

To determine whether the estimates are converging between iterations, it is not possible to use the error estimate as presented in Section 2.2, because the true value of the minimum is not *a priori* known. However, taking advantage of the fact that we are looking for the minimum of the objective function, it is possible to check the descent condition between two successive iterations. Specifically, the goal is to find a value for the damping factor μ that satisfies the following:

$$\Phi\Big(\underbrace{\mathbf{x}^{(k)}+\mu^{(k)}\delta\mathbf{x}^{(k)}}_{\mathbf{x}^{(k+1)}}\Big) \leq \Phi\left(\mathbf{x}^{(k)}\right) \;\Rightarrow\; ||\,\mathbf{R}\Big(\underbrace{\mathbf{x}^{(k)}+\mu^{(k)}\delta\mathbf{x}^{(k)}}_{\mathbf{x}^{(k+1)}}\Big)|| \leq ||\,\mathbf{R}\left(\mathbf{x}^{(k)}\right)|| \tag{15}$$

This formula indicates whether the objective function, computed using the updated state estimate, $\mathbf{x}^{(k+1)}$, decreases. The demand for exact equality may be overly stringent, leading to unnecessary iterations when the difference between two consecutive iterations of the objective functions is negligible. Specifically, we calculate the tolerance as $\left|\mathbf{R}\left(\mathbf{x}^{(k+1)}\right)-\mathbf{R}\left(\mathbf{x}^{(k)}\right)\right|<\tau$, where τ is an arbitrary level of tolerance allowing the descent condition to be validated even when the residuals of two successive iterations are not perfectly equal. If Equation (15) is satisfied, the algorithm is considered to be converging, and in this case, the damping factor μ is increased for the next iteration to accelerate the descent to the minimum. Conversely, if Equation (15) is not met, an inexact line-search method is employed. The damping factor μ is iteratively halved until a value that satisfies the condition in Equation (15) is found. The results presented in the following sections employ a tolerance of $\tau = 10^{-8}$.

The algorithm continues iterating until convergence is achieved. Convergence is determined by monitoring the norm of the applied Newton step, $||\,\mu^{(k)}\delta\mathbf{x}^{(k)}||$, which indicates the magnitude of the correction applied to the current state estimate. Once this norm becomes sufficiently small, indicating that further iterations

will produce negligible changes, the algorithm terminates. In our simulations, the threshold for this check was set using the parameter $\epsilon = 10^{-5}$ for both the pseudorange- and Doppler-based algorithms.

To mitigate problems associated with the two iterative loops of the algorithm, it is necessary to introduce two exit conditions. The first condition halts the convergence iteration loop if the number of iterations exceeds a predefined limit, set to $\zeta = 500$ in our simulations. The second condition concludes the routine for determining the appropriate damping factor, μ, and is triggered when μ falls below a specified threshold, η, which is set to 10^{-6} in our simulations. If the algorithm exits based on either of these conditions, the simulation is considered non-converging. This outcome suggests that no value of μ leads to a decrease in the objective function, indicating that the algorithm may be following an incorrect direction.

3.2.1 | *ADGN Algorithm Results*

The ADGN algorithm was tested with both pseudorange and Doppler shift measurements, with the same experiments performed separately for each measurement type. The theoretical "orbit scale-down" framework was applied to both positioning algorithms, enabling an evaluation of the performance with satellites at various orbital altitudes. For each orbital radius, the algorithm was initialized with eight different damping factors ($\mu \in \{0.01, 0.02, 0.04, 0.08, 0.16, 0.32, 0.64, 1\}$), and the average number of iterations required for convergence across epochs was monitored. Additionally, we tested a different adaptive method for the damping factor, referred to as *1B* in the following figures, which employs a backtracking strategy by setting the damping factor to 1 in line 6 of Algorithm 1. This approach ensures that the damping factor does not retain memory of the previous iteration's state; the damping factor always starts from 1 and only decreases when Equation (15) is not met. To ensure that the algorithm converged on the desired minimum, the position found was compared with the receiver position ground truth.

Figure 2 presents the baseline reference results obtained from the LS method solved by the GN method. The heatmap shows the orbital altitude of the satellites on the x-axis and versions of the algorithm based on pseudorange and Doppler shift on the y-axis. The color gradient represents the average number of iterations across the epochs for each scenario. The MEO and LEO regions are graphically marked by a vertical split on the heatmaps. The black region highlights scenarios in which the algorithm failed to converge. This failure occurred predominantly in the LEO region. The presented results establish a benchmark for interpreting and gaining a better understanding of the advantages of employing the proposed technique.

Figure 3 presents the results of the ADGN algorithm. Each heatmap depicts satellite orbital altitudes on the x-axis, while the y-axis displays various initializations of the damping factor μ, with colors indicating the average iteration count. Figures 3(a) and 3(b) show the convergence results for the ADGN algorithm, applied to pseudorange and Doppler shift measurements, respectively. As in Figure 2, scenarios in which the algorithm fails to converge are shown in black.

Figures 3(a) and 3(b) both show a clear trend. Low values for the initial damping factor $\mu^{(0)}$ ensure convergence for each satellite altitude considered. This result arises because the correct descent path can be followed from the first iteration, with the damping factor progressively doubled until convergence is achieved. A higher damping factor in the initial iterations does not always guarantee convergence, as it can cause the algorithm to overshoot the minimum from the very start. Additionally, the heatmaps in Figure 3 indicate that higher damping factor values lead to faster convergence for satellites in higher orbits, while the number of

ALGORITHM 1
ADGN Algorithm

begin
  /* Initialize the algorithm by setting the damping factor (μ), the initial estimate of the receiver state (x), and the parameters to control the algorithm. */
  Set $\mu \epsilon \eta$ and ζ
  $\mathbf{x}^{(k=0)} = \mathbf{0}$
  repeat
    /* Calculate the residuals and the related Jacobian matrix for the selected receiver state estimate and generate the new estimate and related residuals to be tested. */
    $\mathbf{R}(\mathbf{x}^{(k)}) = \mathbf{y} - \mathbf{f}(\mathbf{x}^{(k)})$ // Residual
    $\delta\mathbf{x}^{(k)} = -(\mathbb{J}_{\mathbf{f}}(\mathbf{x}^{(k)})^{\top}\mathbb{J}_{\mathbf{f}}(\mathbf{x}^{(k)}))^{-1}\mathbb{J}_{\mathbf{f}}(\mathbf{x}^{(k)})^{\top}(\mathbf{y} - \mathbf{f}(\mathbf{x}^{(k)}))$
    // GN step
    /* In case of backtracking: $\mu^{(k)} = 1$ */
    $\mathbf{x}^{\mathrm{UT}} = \mathbf{x}^{(k)} + \mu^{(k)}\,\delta\mathbf{x}^{(k)}$ // Receiver state estimate under test (UT)
    $\mathbf{R}(\mathbf{x}^{\mathrm{UT}}) = \mathbf{y} - \mathbf{f}(\mathbf{x}^{\mathrm{UT}})$ // Residual UT
    /* Control on descent condition to modify damping factor */
    if $||\mathbf{R}(\mathbf{x}^{UT})|| \leq ||\mathbf{R}(\mathbf{x}^{(k)})||$ **then**
      /* When the descent condition is met, the damping factor is increased to accelerate convergence. The damping factor is increased to a maximum of 1. */
      $\mu^{(k)} = \min(2 \times \mu^{(k)}, 1)$
    else
      /* If the descent condition is not met, we decrease the damping factor until the condition is satisfied. */
      repeat
        $\mu^{(k)} = \mu^{(k)}/2$
        $\mathbf{x}^{\mathrm{UT}} = \mathbf{x}^{(k)} + \mu^{(k)}\,\delta\mathbf{x}^{(k)}$
        $\mathbf{R}(\mathbf{x}^{\mathrm{UT}}) = \mathbf{y} - \mathbf{f}(\mathbf{x}^{\mathrm{UT}})$
      until $||\mathbf{R}(\mathbf{x}^{UT})|| \leq ||\mathbf{R}(\mathbf{x}^{(k)})|| \vee \mu^{(k)} > \eta$
    $\mathbf{x}^{(k)} = \mathbf{x}^{\mathrm{UT}}$ // Validates the selected receiver estimate
    $k = k + 1$
  until $||\mu^{(k)}\,\delta\mathbf{x}^{(k)}|| \leq \epsilon \vee k \leq \zeta$

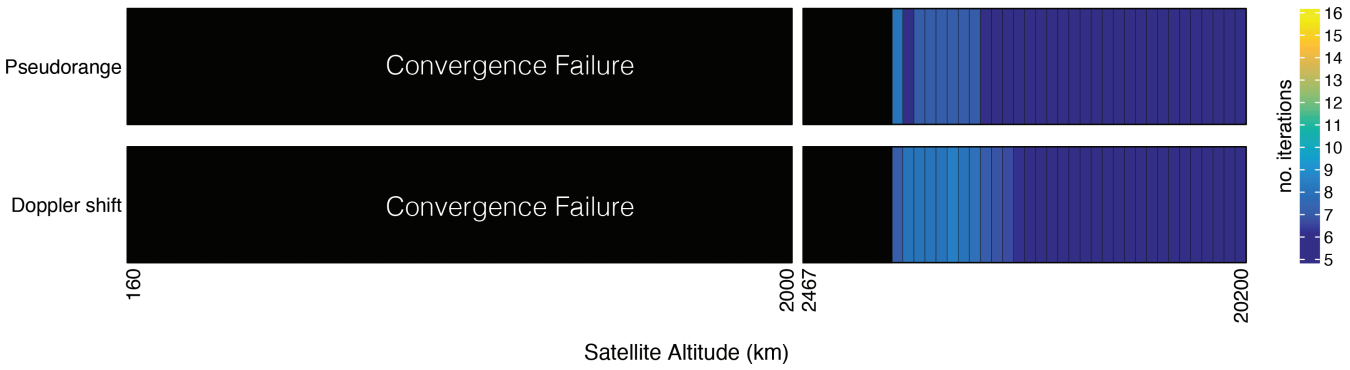


FIGURE 2 Average number of iterations for the LS method solved via the GN method

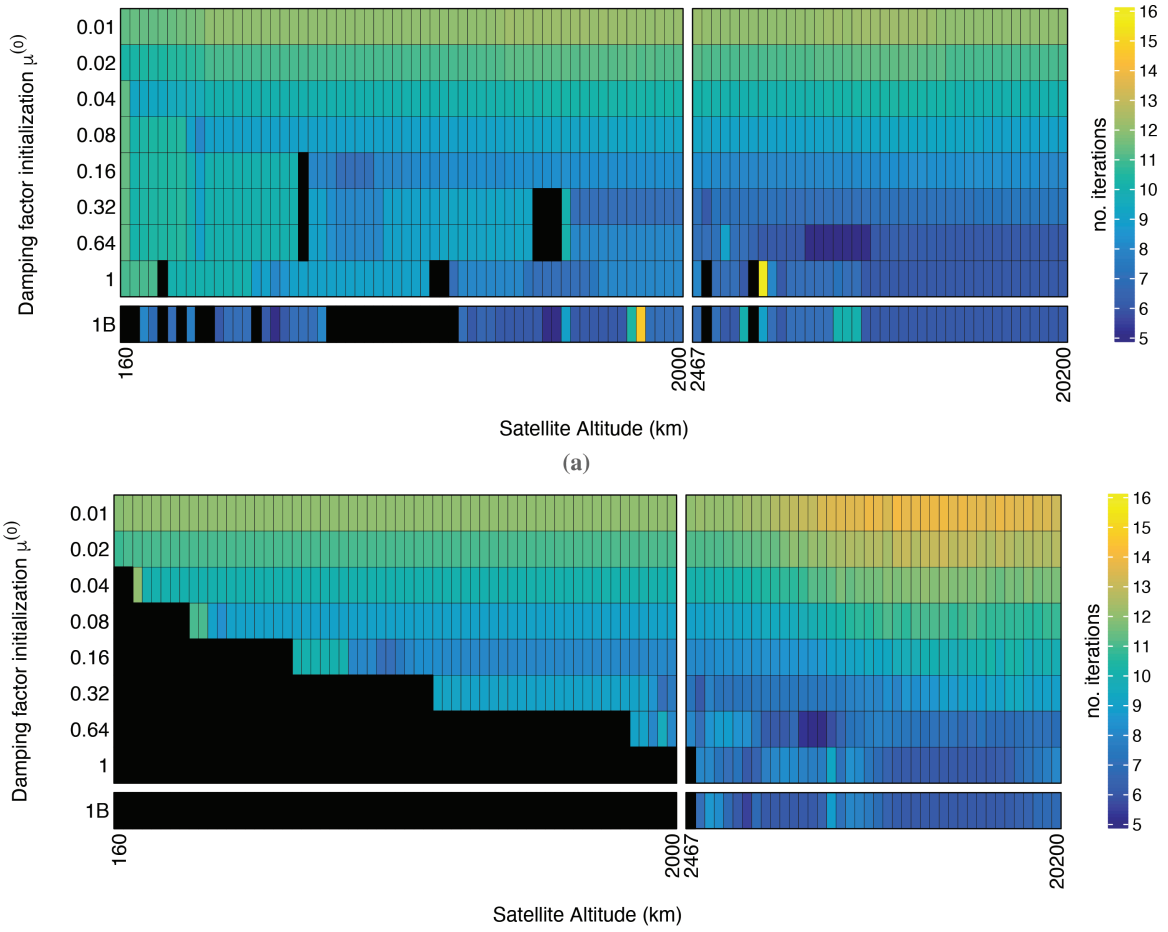


FIGURE 3 Average number of iterations for the LS method solved using the ADGN algorithm presented in Algorithm 1 (a) Convergence heatmap (pseudorange) (b) Convergence heatmap (Doppler shift)

iterations required increases as satellite altitude decreases. This trend is observed because, to satisfy Equation (15), the damping factor must first decrease, necessitating more iterations to return to 1. To address this issue, we also tested an alternative adaptive method, where the damping factor is reset to 1 at each new iteration. However, although this approach records fewer iterations in some successful cases, returning to 1 at every iteration can introduce some risk, resulting in a significantly higher number of divergence cases compared with other initializations. By averaging the number of iterations needed for convergence across all satellite altitudes, we observed that the damping factor yielding the fewest average iterations across all satellite altitudes is 0.08 for pseudorange, with an average iteration count of 9.17. Conversely, for the Doppler shift, a damping factor value of 0.02 is found, with an average iteration count of 11.5.

The upcoming subsections propose modifications to the ADGN algorithm to enhance its performance and guarantee convergence for all orbital radii and damping factor initializations.

3.3 | AW-DGN Algorithm

The descent condition presented in Equation (15) is not always sufficient. In fact, it is well known in the mathematical literature that the mere verification of

Equation (15) is not sufficient to confirm the appropriateness of the damping factor (Bierlaire, 2021; Quarteroni et al., 2014). When choosing an adequate damping factor, it is important to remember that the damping factor cannot be too large or too close to zero (Bierlaire, 2021). Whereas higher values of the damping factor may reduce the number of iterations required for convergence, excessively large values risk overshooting the desired minimum, resulting in non-convergence or convergence to a different minimum. Conversely, a damping factor too close to zero can completely halt progress, which is equally detrimental.

Therefore, criteria that are stricter than Equation (15) are often employed, such as Wolfe conditions (Dennis Jr & Schnabel, 1996; Gill et al., 2019). These conditions establish an upper and lower bound for the dimension of the damping factor, guiding the selection of an appropriate size, and can be expressed as follows:

$$\Phi\left(\mathbf{x}^{(k)}+\mu\,\delta\mathbf{x}^{(k)}\right)<\Phi\left(\mathbf{x}^{(k)}\right)+\mu\beta_1\nabla\Phi\left(\mathbf{x}^{(k)}\right)^{\top}\delta\mathbf{x}^{(k)} \tag{16a}$$

$$\nabla\Phi\left(\mathbf{x}^{(k)}+\mu\,\delta\mathbf{x}^{(k)}\right)^{\top}\delta\mathbf{x}^{(k)}\geq\beta_2\nabla\Phi\left(\mathbf{x}^{(k)}\right)^{\top}\delta\mathbf{x}^{(k)} \tag{16b}$$

where β_1 and β_2 are two parameters, usually chosen such that $0<\beta_1<\beta_2<1$. Once these parameters are determined, the objective is to find a damping factor μ that satisfies both conditions.

The first Wolfe condition (Equation (16a)), also known as Armijo's rule (Dennis Jr & Schnabel, 1996; Quarteroni et al., 2014), is based on the concept of a sufficient decrease in the objective function between two successive iterations. The use of this condition aims to prevent the selection of damping factors that are large but fail to result in a significant decrease in the objective function. The decrease in objective function should be proportional to the slope of the function in $\mathbf{x}^{(k)}$ along the direction $\delta\mathbf{x}^{(k)}$.

In contrast, the second Wolfe condition (Equation (16b)), also known as the curvature condition, prevents the damping factor from being too small by checking the slope at $\mathbf{x}^{(k)}+\mu\,\delta\mathbf{x}^{(k)}$ and at $\mathbf{x}^{(k)}$.

Figure 4 provides a two-dimensional pictorial representation, inspired by Quarteroni et al. (2014), that helps clarify the meaning of the Wolfe conditions. For the purpose of this example, we consider a univariate objective function $\Phi(\mu)=\Phi(x^{(k)}+\mu\,\delta x^{(k)})$. The goal is to find a minimum of this function, ideally the global minimum. In Figure 4, $\Phi(\mu)$ is shown as a solid black curve. At iteration k, the function can be approximated using a first-order Taylor expansion around $x^{(k)}$, i.e., $\Phi(x^{(k)})+\mu\,\nabla\Phi(x^{(k)})\,\delta x^{(k)}$, which is shown as a dashed blue line in Figure 4(a). The first Wolfe condition requires that, after a step of size μ has been taken, the objective function value $\Phi(\mu)$ lie below this linear approximation. The values of μ that satisfy this condition are highlighted in blue in Figure 4(a). The second Wolfe condition concerns the slope of $\Phi(\mu)$. Specifically, this condition ensures that the derivative at the chosen μ is not smaller than the slope at $x^{(k)}$, thereby preventing the selection of excessively small damping factors. In Figure 4 b, the dashed blue line represents $\nabla\Phi(x^{(k)})\,\delta x^{(k)}$, and the shaded regions indicate the values of μ for which the second Wolfe condition is satisfied. Finally, in Figure 4(c), the points along the curve $\Phi(\mu)$ highlighted in blue represent the values of μ that simultaneously satisfy both Wolfe conditions.

In scenarios with noise-affected measurements, such as the scenario under consideration, applying Wolfe conditions to guide the choice of a damping factor can be challenging. Additionally, these conditions depend on two parameters, β_1 and β_2, which must be set according to the specific scenario. While the mathematical literature suggests values for these parameters (Bierlaire, 2021; Quarteroni et al., 2014)

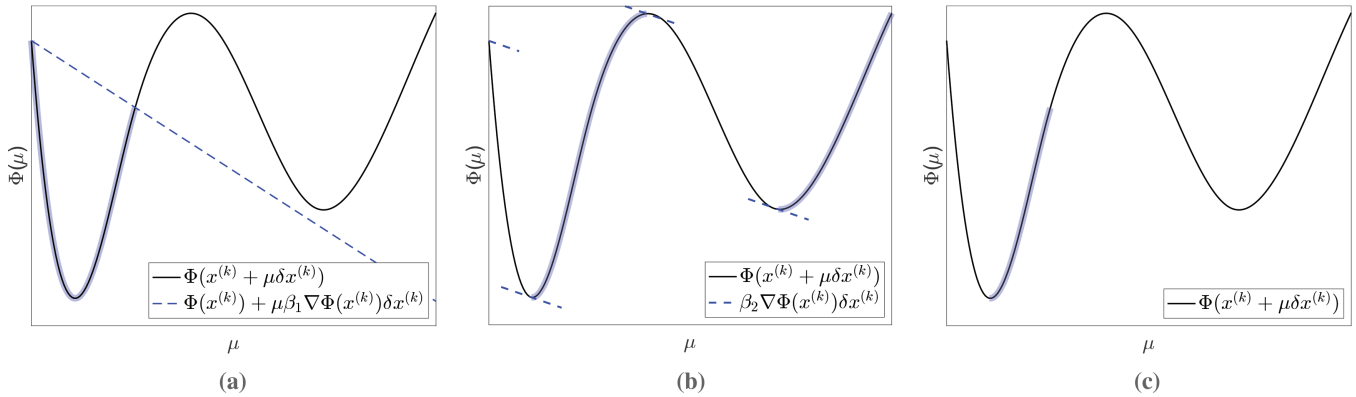


FIGURE 4 Pictorial representation of the Wolfe conditions for a univariate objective function $\Phi(\mu)$ (a) First Wolfe condition (b) Second Wolfe condition (c) Combined view of both Wolfe conditions
The points highlighted in blue correspond to points that satisfy the Wolfe conditions.

(typically $\beta_1 = 10^{-4}$ and $\beta_2 = 0.99$), experiments conducted by the authors have shown that these values are often ineffective. Generalizing parameters that work across various scenarios is not straightforward. Although an adaptive approach for adjusting β_1 and β_2 during iterations may be preferable, it too often proves to be ineffective. Furthermore, computing Equation (16b) requires information about the prime derivatives at the point UT, adding computational burden to the problem. For these reasons, we aim to leverage the principles behind Wolfe conditions and propose a heuristic solution that is more computationally efficient in terms of parameter settings and cost.

We first focus on the first Wolfe condition, which aims to ensure a sufficient decrement in the objective function, proportional to the damping factor. However, because a numerical method is employed, the specific context of the problem to which the condition applies is not considered, but the context is relevant and can be exploited. In our scenario of interest, we are dealing with a positioning problem, where the primary objective is to obtain an accurate estimate of the receiver states.

The knowledge that the receiver is situated on Earth can be exploited to our advantage and forms the basis of our heuristic approach. Our idea is to prevent the iterative approximation point from deviating too far from the Earth's surface. This strategy helps us avoid local minimum points located off the Earth's surface, which are unlikely to correspond to the receiver's actual position. Therefore, in addition to the descent condition in Equation (15), we introduce an additional criterion that triggers a decrease in damping factor even when the estimated receiver position extends beyond the Earth's surface.

In terms of pseudocode, the modifications are minimal and involve adjusting the verification of the descent condition (line 9 of Algorithm 1) as follows:

$$||\mathbf{R}(\mathbf{x}^{\mathrm{UT}})|| < \left\|\mathbf{R}(\mathbf{x})\right\| \wedge ||\{\mathbf{x}\}_{i=1}^{3}|| < \psi + \theta \tag{17}$$

where $\{\mathbf{x}\}_{i=1}^{3}$ represents the components of the receiver state $\mathbf{x}$ related to its position, ψ identifies the mean radius of the Earth ($\psi = 6.371$ km), and θ is a tolerance parameter. In our simulations, θ has been set to 50 km, but this term can be tailored to specific applications. The ADGN algorithm with this condition added will be referred to as AW-DGN from here on.

This condition is beneficial not only for implementation purposes, as it does not require any additional calculations beyond what is already available, but also for its operational efficiency. This condition is particularly relevant during the initial

iterations, when there is a risk of the position estimate drifting too far away, which could cause the algorithm to diverge or converge to other local minima. After properly adjusting the initial steps, the algorithm progresses smoothly towards the minimum point, and the second condition no longer applies. This approach ensures that the algorithm is not overly restrictive and allows it to converge effectively.

The second Wolfe condition (Equation (16b)), which prevents choosing a damping factor too close to zero, is not explicitly verified. This is because the proposed algorithm already imposes a constraint on the damping factor, ensuring that it cannot be smaller than η, which is set to 10^{-6} in our simulations. Considering that the initialization of the damping factor used is at least four orders of magnitude larger than η and that the algorithm gradually reduces the damping factor whenever the descent condition is not satisfied, the chosen damping factor is inherently chosen not to be too small, making our implementation significantly simpler compared with enforcing the second Wolfe condition (Equation (16b)).

3.3.1 | *AW-DGN Algorithm Results*

Figure 5 shows two heatmaps for the LS method based on pseudorange and Doppler shift measurements. The color scale indicates the average number of iterations across epochs required by the algorithm to reach convergence. For each satellite orbital altitude, nine different damping factor initializations were tested,

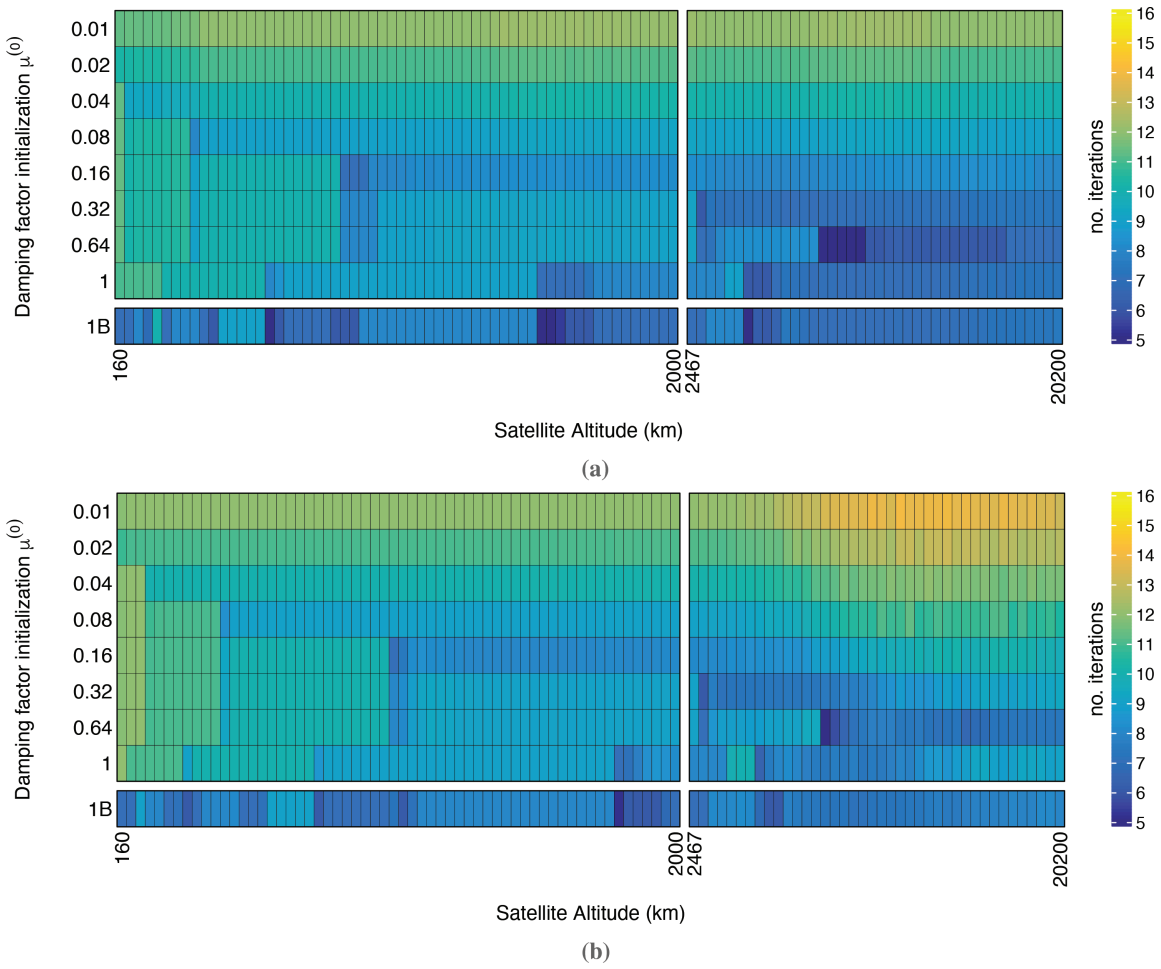


FIGURE 5 Average number of iterations for the LS method solved using the AW-DGN algorithm (a) Convergence heatmap (pseudorange) (b) Convergence heatmap (Doppler shift)

ALGORITHM 2
AW-DGN-BT

begin
  Set ϵ η and ζ
  $\mathbf{x}^{(k=0)} = \mathbf{0}$
  repeat
    $\mathbf{R}(\mathbf{x}^{(k)}) = \mathbf{y} - \mathbf{f}(\mathbf{x}^{(k)})$ `// Residual`
    $\boldsymbol{\delta}\mathbf{x}^{(k)} = -(\mathbb{J}_{\mathbf{f}}(\mathbf{x}^{(k)})^{\top}\mathbb{J}_{\mathbf{f}}(\mathbf{x}^{(k)}))^{-1}\mathbb{J}_{\mathbf{f}}(\mathbf{x}^{(k)})^{\top}(\mathbf{y} - \mathbf{f}(\mathbf{x}^{(k)}))$ `// GN step`
    $\mu^{(k)} = 1$
    $\mathbf{x}^{\mathrm{UT}} = \mathbf{x}^{(k)} + \mu^{(k)}\,\boldsymbol{\delta}\mathbf{x}^{(k)}$ `// Receiver state estimate UT`
    $\mathbf{R}(\mathbf{x}^{\mathrm{UT}}) = \mathbf{y} - \mathbf{f}(\mathbf{x}^{\mathrm{UT}})$ `// Residual UT`
    if $\neg(||\mathbf{R}(\mathbf{x}^{UT})|| \leq ||\mathbf{R}(\mathbf{x})|| \wedge ||\{\mathbf{x}\}_{i=1}^{3}|| < \psi + \theta)$ **then**
      repeat
        $\mu^{(k)} = \mu^{(k)}/2$
        $\mathbf{x}^{\mathrm{UT}} = \mathbf{x}^{(k)} + \mu^{(k)}\,\boldsymbol{\delta}\mathbf{x}^{(k)}$
        $\mathbf{R}(\mathbf{x}^{\mathrm{UT}}) = \mathbf{y} - \mathbf{f}(\mathbf{x}^{\mathrm{UT}})$
      until $||\mathbf{R}(\mathbf{x}^{UT})|| \leq ||\mathbf{R}(\mathbf{x}^{(k)})|| \vee \mu^{(k)} > \eta \wedge ||\{\mathbf{x}^{(k)}\}_{i=1}^{3}|| < \psi + \theta) \vee \mu^{(k)} > \eta$
    $\mathbf{x}^{(k)} = \mathbf{x}^{\mathrm{UT}}$ `// Validates the selected receiver estimate`
    $k = k + 1$
  until $(||\mu^{(k)}\,\boldsymbol{\delta}\mathbf{x}^{(k)}|| \leq \epsilon) \vee k \leq \zeta$

including the backtracking approach labeled as "1B" in the heatmap. A notable difference between Figure 5 and Figure 3 is the absence of black areas that indicate convergence failure in the algorithm. This result demonstrates that the modifications applied to the algorithm have been effective, resulting in convergence for every scenario analyzed.

Furthermore, upon closer examination, it is evident that for a lower damping factor, the number of iterations remains unchanged compared with the previous simulation, as the applied modification does not directly affect the number of iterations. However, for higher damping factors, where the modification of the descent condition comes into play, differences can be observed. Nevertheless, the general trend observed previously is confirmed, wherein for higher initialization damping factor values, the number of iterations increases as the orbital altitude of the satellites decreases.

The damping factor that resulted in the fewest average iterations across all sets of orbital radii was the 1B initialization for both the pseudorange and Doppler shift, with average iteration counts of 7.23 and 7.5, respectively. This result suggests that by using Equation (17), which prevents algorithm divergence, the damping factor with backtracking to 1 at each iteration offers the best performance in terms of convergence iterations compared with all other damping factor initializations, especially in the LEO region. Thus, this approach is the most effective among all tested implementations. Algorithm 2 presents the pseudocode with all modifications needed to implement the AW-DGN with backtracking (AW-DGN-BT).

4 | TESTING AW-DGN-BT

In this section, we conduct a stress test of the AW-DGN-BT algorithm, evaluating its performance based on the number of iterations required for convergence

across different satellite geometries, in order to assess its overall effectiveness. AW-DGN-BT algorithm results were evaluated against both the GN method and the ADGN algorithm. In the latter case, the damping factor is updated using the same backtracking strategy adopted in the AW-DGN-BT method, in order to enable a fair comparison between the two approaches. The version of the ADGN algorithm that employs the backtracking update rule will be referred to hereafter as ADGN-BT.

4.1 | Simulation Framework

We simulated a Walker Delta constellation similar to the Galileo system, consisting of 24 MEO satellites. Using MATLAB's built-in orbit propagator, we computed the satellite positions and velocities over a 10-h period, with sampling every 10 s, which resulted in 600 epochs. The receiver position was simulated at three locations: Cape Town, Longyearbyen, and Shanghai. A visibility analysis was then conducted to determine which satellites were usable for each location. The large time window provides a significant advantage in terms of satellite geometry, as the satellite positions change not only between different scenarios but also within the same scenario over time. This framework incorporates the orbit scale-down approach described in Section 3.1. For each simulated location, both satellite positions and velocities were adjusted, and the positioning algorithm (Algorithm 2) was tested to evaluate its robustness in terms of convergence. Table 1 summarizes the three scenarios analyzed.

4.2 | Simulation Results

Figure 6 compares the average number of iterations across epochs required for convergence among the GN-based positioning algorithm, ADGN-BT, and AW-DGN-BT, across the different scenarios summarized in Table 1. Each column in the heatmaps corresponds to a different satellite altitude, whereas rows distinguish between the three algorithms being compared. For each scenario, both pseudorange and Doppler measurements were tested. Black areas in the heatmaps indicate cases where the algorithm did not converge, which occur frequently for both the GN and ADGN-BT methods as the satellite orbital radius approaches the LEO region. In contrast, the AW-DGN-BT algorithm is highly effective, consistently achieving convergence in all scenarios. These trends, visually illustrated in Figure 6, are quantitatively summarized in Tables 2 and 3. In particular, Table 2 reports the percentage of successful convergence for each algorithm and measurement type across the three representative scenarios. The AW-DGN-BT algorithm achieves a consistent 100% success rate in all cases, unlike GN and ADGN-BT,

TABLE 1
Number of Visible Satellites in the Scenarios UT

Scenario	Receiver position	Mean no. of satellites in view	Min no. of satellites in view	Max no. of satellites in view
S1a	Cape Town	6.74	5	9
S1b	Longyearbyen	8.31	7	9
S1c	Shanghai	7.03	5	9

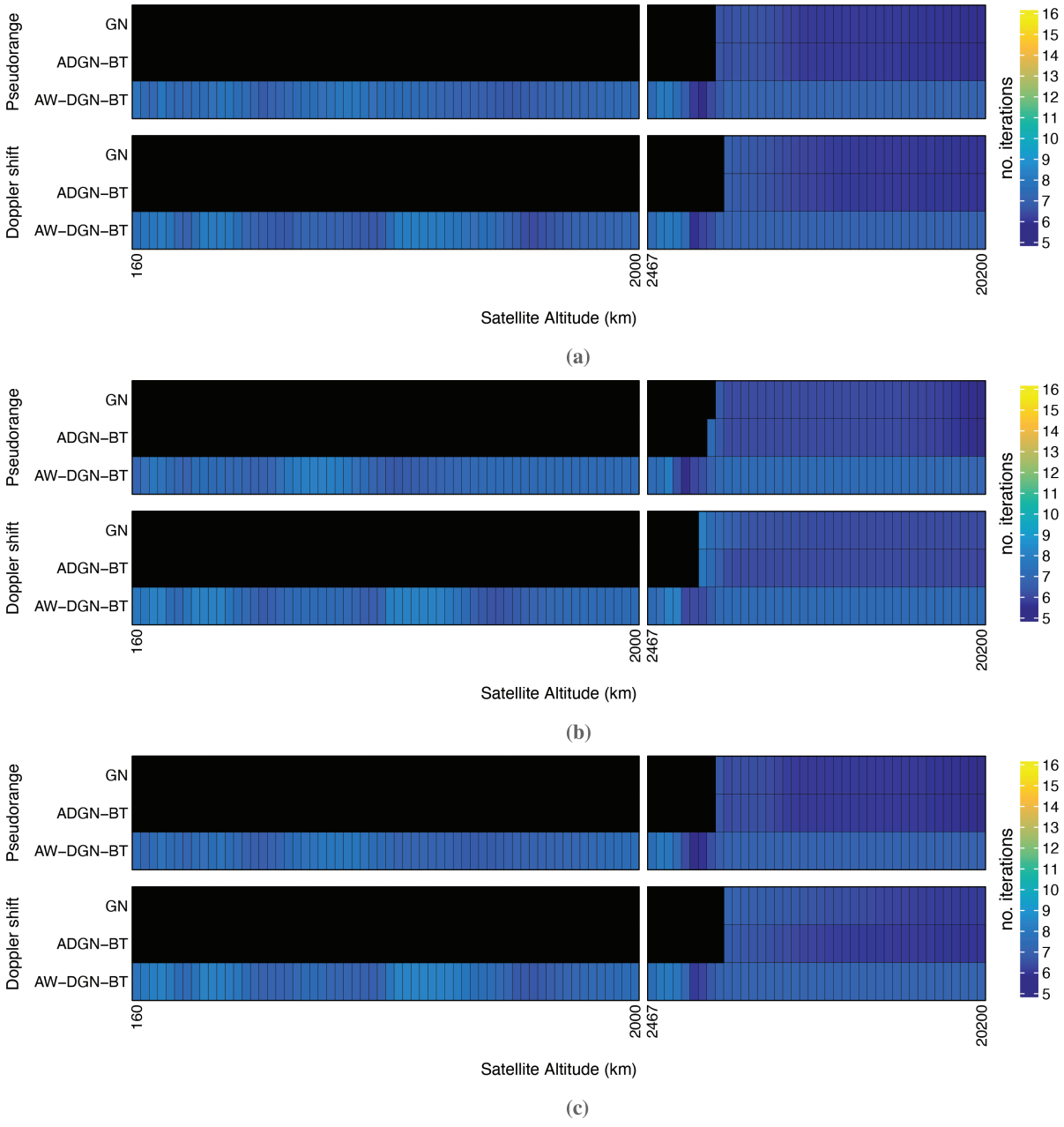


FIGURE 6 Average number of iterations for the LS method solved using both the GN and AW-DGN-BT algorithms; LLA: latitude, longitude, altitude (a) S1a: Cape Town (South Africa), LLA = [−33.92899°, 18.41740°, 0 m] (b) S1b: Longyearbyen (Norway), LLA = [78.22894°, 15.39493°, 0 m] (c) S1c: Shanghai (China), LLA = [31.14274°, 121.80411°, 5 m]

TABLE 2
Percentage Availability for Each Scenario When Using the GN, ADGN-BT, and AW-DGN-BT Algorithms with Both Pseudorange and Doppler Shift Measurements

Scenario	GN		ADGN-BT		AW-DGN-BT	
	Pseudorange	Doppler shift	Pseudorange	Doppler shift	Pseudorange	Doppler shift
S1a	32%	31%	32%	31%	100%	100%
S1b	32%	34%	33%	34%	100%	100%
S1c	32%	31%	32%	31%	100%	100%

TABLE 3
Average Number of Iterations and Execution Time When Using the GN, ADGN-BT, and AW-DGN-BT Algorithms for Different Types of Measurements

Measurement	GN		ADGN-BT		AW-DGN-BT	
	Mean no. of iterations	Execution time (ms)	Mean no. of iterations	Execution time (ms)	Mean no. of iterations	Execution time (ms)
Pseudorange	6.02	0.59	6.03	0.57	7.08	0.73
Doppler shift	6.22	0.79	6.11	0.81	7.18	1.21

which show significantly lower success percentages (below 40% in virtually all scenarios). Table 3 further reports the average number of iterations and execution time (in milliseconds) across all considered satellite altitudes and scenarios (Table 1) for each algorithm and measurement type. These averages are calculated only over the satellite altitudes where the algorithm successfully converged. Because AW-DGN-BT converges in all cases, its averages are computed over all 100 satellite altitudes, whereas those for GN and ADGN-BT include only the subset of altitudes that resulted in successful convergence. Although AW-DGN-BT requires slightly more iterations (around 7) and marginally higher computation time (around 1 ms), this trade-off is offset by its superior convergence reliability. Note that the execution time is reported here as a representative metric, although it may vary depending on implementation details, hardware specifications, and the programming environment used.

5 | CONCLUSION

LS-based methods resolved using the GN method may exhibit convergence issues, particularly for LEO satellite constellations. These issues necessitate the availability of prior information regarding the receiver's position and the positioning of the algorithm's initial state estimate in close proximity to the receiver's ground truth. To address this concern, this study proposes utilizing a variation of the GN method, which is known in mathematical literature as DGN. The introduction of an adaptive damping factor, ranging from 0 to 1, into the traditional GN step was intended to improve convergence. The damping factor is adaptive in that it can decrease in the case of non-convergence and increase otherwise. Experimentation revealed that initiating the damping factor with smaller values (e.g., 0.01, 0.02) enabled convergence in both MEO and LEO regions. Nevertheless, larger damping factor initializations still failed to ensure convergence in certain scenarios. To address such scenarios, we devised a heuristic based on Wolfe conditions tailored to GNSS terrestrial positioning. This heuristic seamlessly integrates with our proposed algorithm, effectively resolving convergence issues not addressed by previous algorithms while retaining ease of implementation. Our findings suggest that by modifying the LS algorithm, its convergence can be enhanced, enabling its application with LEO satellites using both pseudorange and Doppler shift measurements.

ACKNOWLEDGMENTS

This study was conducted within Ministerial Decree no. 1062/2021 and received funding from the FSE REACT-EU - Programma Operativo Nazionale (PON) Ricerca e Innovazione 2014–2020. This manuscript reflects only the authors' views and opinions, neither the European Union nor the European Commission can be considered responsible for the views presented herein.

A. Minetto acknowledges funding from research contract no. 32-G-13427-5 DM 1062/2021 funded within the PON Ricerca e Innovazione of the Italian Ministry of University and Research (MUR).

This paper is part of the NODES project, which has received funding from the MUR – M4C2 1.5 of PNRR funded by the European Union - NextGenerationEU (grant agreement no. ECS00000036).

REFERENCES

Bierlaire, M. (2021). *Optimization: Principles and algorithms* (2nd ed.). EPFL Press English Imprint. https://transp-or.epfl.ch/books/optimization/html/OptimizationPrinciplesAlgorithms2018.pdf

Björck, Å. (1996). *Numerical methods for least squares problems*. Society for Industrial and Applied Mathematics (SIAM). https://epubs.siam.org/doi/10.1137/1.9781611971484

Çelikbilek, K., Saleem, Z., Morales Ferre, R., Praks, J., & Lohan, E. S. (2022). Survey on optimization methods for LEO-satellite-based networks with applications in future autonomous transportation. *Sensors*, *22*(4), 1421. https://doi.org/10.3390/s22041421

Dennis, J. E., Jr., & Schnabel, R. B. (1996). *Numerical methods for unconstrained optimization and nonlinear equations*. Society for Industrial and Applied Mathematics (SIAM). https://epubs.siam.org/doi/10.1137/1.9781611971200

Farhangian, F., & Landry, R. (2020). Multi-constellation software-defined receiver for Doppler positioning with LEO satellites. *Sensors*, *20*(20). https://doi.org/10.3390/s20205866

Gander, W., Gander, M. J., & Kwok, F. (2014). *Scientific computing-an introduction using Maple and MATLAB* (Vol. 11). Springer Science & Business.

Gill, P. E., Murray, W., & Wright, M. H. (2019). *Practical optimization*. Society for Industrial and Applied Mathematics (SIAM).

Hager, W. (1988). *Applied numerical linear algebra*. Prentice-Hall International.

Hartnett, M. J. (2022). Performance assessment of navigation using carrier Doppler measurements from multiple LEO constellations. *Theses and Dissertations*, *5456*. https://scholar.afit.edu/etd/5456

Kassas, Z. (2021). Navigation from low Earth orbit, part 2: Models, implementation, and performance. In Y. T. J. Morton, F. van Diggelen, J. J. Spilker, B. W. Parkinson, S. Lo, & G. Gao (Eds.), *Position, navigation, and timing technologies in the 21st century: Integrated satellite navigation, sensor systems, and civil applications*, 1381–1412. Wiley-IEEE Press. https://doi. org/10.1002/9781119458555.ch43b

Kershner, R. B., & Newton, R. R. (1962). The Transit system. *Journal of Navigation*, *15*(2), 129–144. https://doi.org/10.1017/S0373463300035943

Khalife, J., Neinavaie, M., & Kassas, Z. M. (2021). Blind Doppler tracking from OFDM signals transmitted by broadband LEO satellites. In *2021 IEEE 93rd Vehicular Technology Conference (VTC2021-Spring)*, 1–5. https://doi.org/10.1109/VTC2021-Spring51267.2021.9448678

Khalife, J. J., & Kassas, Z. M. (2019). Receiver design for Doppler positioning with LEO satellites. *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 5506–5510. https://doi.org/10.1109/ICASSP.2019.8682554

Kozhaya, S. E., & Kassas, Z. M. (2023). Positioning with Starlink LEO satellites: A blind Doppler spectral approach. In *2023 IEEE 97th Vehicular Technology Conference (VTC2023-Spring)* (pp. 1–5). https://doi.org/10.1109/VTC2023-Spring57618.2023.10199264

Minetto, A., Nardin, A., & Dovis, F. (2024). On the use of meta-signals to counteract ionospheric phase advance effects on wideband LEO PNT signals. *Proc. of the 2024 International Technical Meeting of the Institute of Navigation*, Long Beach, CA, 94–108. https://doi.org/10.33012/2024.19524

Morichi, L., Minetto, A., Nardin, A., Zocca, S., & Dovis, F. (2024). Pseudorange and Doppler-based state estimation from MEO to LEO: A comprehensive analysis of maximum likelihood estimators. *Proc. of the 2024 International Technical Meeting of the Institute of Navigation*, Long Beach, CA, 677–691. https://doi.org/10.33012/2024.19508

Nardin, A., Dovis, F., & Fraire, J. A. (2021). Empowering the tracking performance of LEO-based positioning by means of meta-signals. *IEEE Journal of Radio Frequency Identification*, *5*(3), 244–253. https://doi.org/10.1109/JRFID.2021.3077082

Nardin, A., Dovis, F., Zanier, F., & Melman, F. (2024). Flexible meta-signal tracking: Architecture robustness in challenging PNT scenarios. In *IEEE Transactions on Aerospace and Electronic Systems, 60*(3), 3095–3107. https://doi.org/10.1109/TAES.2024.3359594

Nocedal, J., & Wright, S. J. (1999). *Numerical optimization*. Springer.

Orabi, M., Khalife, J., & Kassas, Z. M. (2021). Opportunistic navigation with Doppler measurements from Iridium Next and Orbcomm LEO satellites. *2021 IEEE Aerospace Conference (50100)*, 1–9. https://doi.org/10.1109/AERO50100.2021.9438454

Prol, F. S., Ferre, R. M., Saleem, Z., Välisuo, P., Pinell, C., Lohan, E. S., Elsanhoury, M., Elmusrati, M., Islam, S., Çelikbilek, K., Selvan, K., Yliaho, J., Rutledge, K., Ojala, A., Ferranti, L., Praks, J., Bhuiyan, M. Z. H., Kaasalainen, S., & Kuusniemi, H. (2022). Position, navigation, and timing (PNT) through low Earth orbit (LEO) satellites: A survey on current status, challenges, and opportunities. *IEEE Access*, *10*, 83971–84002. https://doi.org/10.1109/ACCESS.2022.3194050

Psiaki, M. L. (2021). Navigation using carrier Doppler shift from a LEO constellation: Transit on steroids. *NAVIGATION*, *68*(3), 621–641. https://doi.org/10.1002/navi.438

Psiaki, M. L., & Slosman, B. D. (2019). Tracking digital FM OFDM signals for the determination of navigation observables. *Proc. of the 32nd International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS+ 2019), Miami, Florida*, 2325–2348. https://doi.org/10.33012/2019.17120

Quarteroni, A., Sacco, R., & Saleri, F. (2010). *Numerical mathematics* (Vol. 37). Springer Science & Business Media.

Quarteroni, A., Saleri, F., & Gervasio, P. (2014). *Scientific computing with MATLAB and Octave*. Springer Berlin, Heidelberg.

Reid, T. G., Walter, T., Enge, P. K., Lawrence, D., Cobb, H. S., Gutt, G., O'Conner, M., & Whelan, D. (2021). Navigation from low Earth orbit, Part 1: Concept, currrent capability and future promise. In Y. T. J. Morton, F. van Diggelen, J. J. Spilker, B. W. Parkinson, S. Lo, & G. Gao (Eds.), *Position, navigation, and timing technologies in the 21st century: Integrated satellite navigation, sensor systems, and civil applications*, 1359–1379. Wiley-IEEE Press. https://doi. org/10.1002/9781119458555.ch43a

Shi, C., Zhang, Y., & Li, Z. (2023). Revisiting Doppler positioning performance with LEO satellites. *GPS Solutions*, *27*(3), 126. https://doi.org/10.1007/s10291-023-01466-w

van Graas, F. (2023). Doppler processing for satellite navigation. *Proc. of the 2023 IEEE/ION Position, Location and Navigation Symposium (PLANS)*, Monterey, CA, 365–371. https://doi.org/10.1109/PLANS53410.2023.10140011

Van Uytsel, W., Janssen, T., Halili, R., & Weyn, M. (2023). Exploring positioning through pseudoranges using low Earth orbit satellites. In L. Barolli (Ed.), *Advances on p2p, parallel, grid, cloud and internet computing*, 278–287. Springer International Publishing. https://doi.org/10.1007/978-3-031-19945-5_28

Yan, J., Tiberius, C. C. J. M., Janssen, G. J. M., Teunissen, P. J. G., & Bellusci, G. (2013). Review of range-based positioning algorithms. *IEEE Aerospace and Electronic Systems Magazine*, *28*(8), 2–27. https://doi.org/10.1109/MAES.2013.6575420

Zeng, T., Sui, L., Jia, X., Lv, Z., Ji, G., Dai, Q., & Zhang, Q. (2019). Validation of enhanced orbit determination for GPS satellites with LEO GPS data considering multi ground station networks. *Advances in Space Research*, *63*(9), 2938–2951. https://doi.org/10.1016/j.asr.2018.06.012

Zhang, P., Ding, W., Qu, X., & Yuan, Y. (2023). Simulation analysis of LEO constellation augmented GNSS (LeGNSS) zenith troposphere delay and gradients estimation. *IEEE Transactions on Geoscience and Remote Sensing*, *61*, 1–12. https://doi.org/10.1109/TGRS.2023.3241956

How to cite this article: Morichi, L., Zocca, S., Minetto, A., Nardin, A., & Dovis, F. (2026). Pseudorange and doppler-based positioning: Enabling convergence of least-squares estimation from MEO to LEO. *NAVIGATION, 73*. https://doi.org/10.33012/navi.734