Satellite Selection Algorithm to Optimize a Solution for Autonomous Driving Applications

Tiago Gonçalves
Bosch Car Multimedia
Braga, Portugal
DGAOT, Faculty of Sciences,
University of Porto
Porto, Portugal
Email:
Tiago.Goncalves4@pt.bosch.com

Gianmarco Fedeli
Bosch Car Multimedia
Braga, Portugal
Email:
Gianmarco.Fedeli@pt.bosch.com

Clara Lázaro
DGAOT, Faculty of Sciences,
University of Porto, Porto, Portugal
Interdisciplinary Centre of Marine
and Environmental Research
(CIIMAR), Matosinhos, Portugal
Email: clazaro@fc.up.pt

Hélder Silva
Algoritmi Research Centre
University of Minho
Guimarães, Portugal
Email: hdsilva@dei.uminho.pt

Telmo Vieira
DGAOT, Faculty of Sciences,
University of Porto, Porto, Portugal
Interdisciplinary Centre of Marine and Environmental
Research (CIIMAR), Matosinhos, Portugal
Email: telmo.vieira@fc.up.pt

***Abstract*— In recent years, there has been a significant exponential growth in the number of satellites from Global Navigation Satellite Systems (GNSSs). The proliferation of satellites could lead to noteworthy effects on different industries, such as aviation and autonomous driving, by substantially improving positioning accuracy and optimizing service efficiency. Nevertheless, deploying a large number of satellites comes with certain implications, such as an unavoidable increase in computational demands and higher power consumption for the receiver. A satellite selection algorithm can address these challenges by selecting a smaller subset of the total visible satellites, with comparable or even better positional accuracy. This paper introduces an algorithm for GNSS satellite selection in autonomous driving applications, which incorporates various factors including satellite elevation and signal strength. The algorithm identifies the optimal subset of satellites by applying a Sequential Updating Method (SUM) to generate multiple subsets and subsequently compare their Weighted Position Dilution of Precision (WPDOP). The subset with the lowest WPDOP is ultimately selected for use in the positioning process. The algorithm's performance is assessed in a dynamic scenario under challenging conditions, typical of autonomous driving context, and compared with other algorithms from the literature. Results show that the proposed algorithm is suitable for the target application, due to its ability to achieve higher positioning accuracy and reduce computational time compared to other methods in the literature.**

Keywords - Satellite Selection; GNSS; WPDOP; Computational Effort; Autonomous Driving.

I. INTRODUCTION

Global Navigation Satellite Systems (GNSSs) are a vital component in today's positioning and navigation landscape and stand as a key technology to society's future. A high level of maturity has been achieved with the latest generation of satellites, providing new and improved positioning signals, contributing to the achievement of accuracies that highly surpass those originally planned.

There are several error sources within a GNSS link that degrade the positioning accuracy of the system. Errors related to the atmosphere, clocks of both the satellite and receiver, the local environment, the geometry of the satellite constellations, their orbits and even intentional errors can cause discrepancies in the position of the user. To mitigate part of these effects, a multitude of correction methods have been developed, that seek to provide the best performance while offering the best price and versatility [1].

In open sky conditions, optimizing the Geometric Dilution Of Precision (GDOP) is sufficient to guarantee high accuracy solutions. When considering challenging conditions, such as a vehicle crossing an urban environment, optimizing GDOP is not enough, since different error magnitudes will affect each measurement due to multipath and Non-Line-Of-Sight (NLOS) conditions from the local environment.

With the increase in satellite number, a good satellite selection algorithm is vital to decrease the signal-processing burden of the receiver, while providing good accuracy.

A satellite selection algorithm could also benefit autonomous driving applications, contributing to achieve automation level 4, which only requires the driver to take control in very specific situations [2]. To reach automation level 5, where the vehicle is expected to operate under every condition and in every environment, GNSS could be a key component alongside other sensors, though a lot of features are still to be investigated [3]. The GNSS system can help overcome automation challenges like lane-level maneuvers, the oversight of vision systems, safety through independence, or even unlock interoperability through consistent timing and reference frames for vehicle to everything cooperation [4].

This work provides two key contributions: a satellite selection algorithm targeted for autonomous driving applications, and a performance comparison with well-known solutions from the literature using real dynamic data with a

 ISBN: 978-1-68558-089-6

highly accurate reference ground truth. Section 2 provides background in satellite selection algorithms and Section 3 introduces the proposed solution, while Section 4 presents a performance comparison. In Section 5, a conclusion is given.

II. BACKGROUND

A. Geometric Dilution of Precision

The GDOP is a metric related to the geometry of the solution, i.e., where satellites are in space, relative to the receiver. It is given by [5]:

$$GDOP = \sqrt{trace((G^T G)^{-1})} \quad (1)$$

where G is the design matrix for the position estimation solution (e.g., using a least-squares approach). For dual constellation case, the G matrix used for position and clock estimation is given by [5]:

$$G = \begin{bmatrix} e_x^{i,1} & e_y^{i,1} & e_z^{i,1} & 1 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ e_x^{i,m} & e_y^{i,m} & e_z^{i,m} & 1 & 0 \\ e_x^{j,1} & e_y^{j,1} & e_z^{j,1} & 0 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ e_x^{j,n} & e_y^{j,n} & e_z^{j,n} & 0 & 1 \end{bmatrix} \quad (2)$$

e represents the line-of-sight vector with x, y, and z components for *m* satellites of the *i-th* constellation, and *n* satellites of the *j-th* constellation. The last two columns represent the clock biases common for each measurement within constellations.

B. Optimal Method

One of the objectives of a satellite selection algorithm is to minimize the GDOP for a subset of *k* satellites from a set of *s* visible satellites. An optimal (also known as brute force) approach is to compute the GDOP value for every subset of *k* satellites from the *s* satellites available (with $k<s$).

The total number of computations of the GDOP function is given by the combinations' equation [6]:

$$C_k^s = \frac{s!}{k!\,(s-k)!} \quad (3)$$

For real-time processing, this method is only possible for a low number of satellites, due to the exponential increase in number of combinations (e.g., selecting 10 out of 15 satellites results in 3003 GDOP computations). The brute force method is considered optimal because it always produces the best possible GDOP, but at the expense of high computational cost.

Other algorithms are classified as sub-optimal or quasi-optimal, depending on how close they are to the GDOP value given by the optimal method, but with less computational effort. The Ultra-Rapid satellite selection proposed in [7] is one example, where it utilizes a constrained downdate method. This approach starts by computing the weight coefficient matrix for the all-in-view solution, and in order to avoid matrix inversions, uses the inversion lemma to find the individual contribution of each satellite. The satellite that contributes the least for the increase of GDOP is removed, and the process repeats itself until the desired subset size is achieved. A method focused on dynamic scenarios is presented in [8]. Even though being smartphone based, it is one of the few examples in literature that use dynamic data in the performance analysis, and therefore will be taken in consideration in the performance comparison.

C. Sequential Updating Method

In order to compare the highest number of satellite subsets while reducing the computation effort to a minimum, a Sequential Updating Method (SUM) is introduced in [9]. Figure 1 depicts a flowchart of the method. The process starts by choosing 4 satellites from the constellation with the most satellites to act as an initial subset. It proceeds to add each of the remaining *n*-4 satellites available to create *n*-4 subsets of size 5, with *n* being the number of visible satellites. After the creation of this group of subsets, every satellite that comes after the last satellite added in each subset is included in the subset and the subsets that had a satellite added in common will compare the trace of the inverse matrix with each other, and the one with minimal value will go to the next iteration. The process is repeated until *m* subsets with the desired size *k* remain and the one with the lowest GDOP is chosen to be used for positioning.

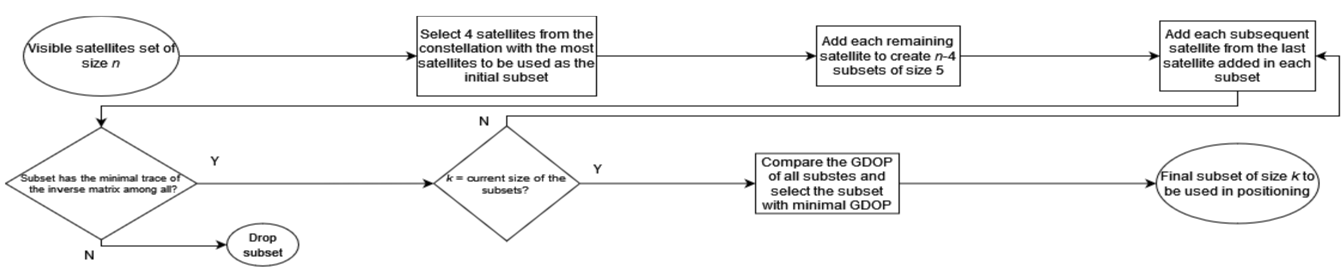

Figure 1. Process of the Sequential Updating Method.

D. Weight Function

For the purpose of creating a holistic satellite selection algorithm, a weight function is designed to provide higher weights to satellites better suited to be used in the positional solution. The proposed weight function has the form:

$$w_i = \sum_{g=1}^{F} w_g p_g \quad (4)$$

where F is the number of factors contributing to the function, w_g is the weight of factor g and p_g is the percentage assigned to the weight w_g, in order to determine the contribution to the total weight w_i. The p_g factors were determined through trial and error, iteratively refined until the optimal combination was identified, yielding the most favorable outcome.

The first two factors of the weight function are presented in [10], with those factors being the elevation and Carrier to Noise Ratio (CNR) of each satellite. The weight of the elevation factor is given as:

$$w_{el_i} = \frac{\theta_i}{\theta_{max}} \quad (5)$$

where θ_i is the elevation angle of the *i-th* satellite, θ_{max} is the maximum elevation angle among all the visible satellites at the current epoch, with all the angles being in degrees. The weight of the CNR is expressed as:

$$w_{CNR_i} = (1 + \alpha_m) \cdot \frac{CNR_i}{CNR_{max}} \quad (6)$$

with α_m being the multipath scaling factor given as [11]:

$$\alpha_m = \frac{R_{coef} - 1}{R_{coef} + 1} \quad (7)$$

where R_{coef} is the reflection coefficient and is expressed as [11]:

$$R_{coef} = \frac{10^{\frac{CNR_{max}}{20}}}{10^{\frac{CNR_i}{20}}} \quad (8)$$

For multipath free signals, the reflection coefficient is 1, while the multipath scaling factor is 0, which will not affect (6). In the case of multipath presence, the multipath scaling factor will be different than 0 and is added to ‘1’ in (6) [12].

Another factor to complement the weight function introduced in [12] is the pseudorange variance. To calculate its value, the RTKLIB software default weighting system is used [13].

$$\sigma_i^2 = \frac{a^2}{sin^2\theta_i} \quad (9)$$

where a was determined empirically in [14] and θ_i is the elevation angle of the *i-th* satellite. Therefore, the weight of this factor is given by normalizing the pseudorange variance:

$$w_{var_i} = \frac{\max(\sigma^2) - \sigma_i^2}{\max(\sigma^2) - \min(\sigma^2)} \quad (10)$$

with the minimum and maximum values of σ^2 being calculated at each epoch.

The final factor added to this function is the CNR variation from epoch to epoch. This factor can be seen as an indicator of multipath and including its contribution in the weight function allows multipath affected measurements to have lower weight [15]. The CNR factor is expressed as:

$$\sigma_{CNR_j} = \sqrt{\frac{t-1}{t} \cdot \sigma_{CNR_{j-1}}^2 + \frac{1}{t} \cdot (CNR_i - \mu_j)^2} \quad (11)$$

$$\mu_j = \frac{t-1}{t} \cdot \mu_{j-1} + \frac{1}{t} * CNR_i \quad (12)$$

where t is the number of consecutive epochs in which the measurement was present, j represents the current epoch, σ_{CNR} and μ are the standard deviation and the mean, respectively, of the carrier to noise ratio among all satellites and CNR_i is the carrier to noise ratio of the *i-th* satellite. Hence, the weight is defined as:

$$w_{var_i} = \frac{\max(\sigma_{CNR}) - \sigma_{CNR_i}}{\max(\sigma_{CNR}) - \min(\sigma_{CNR})} \quad (13)$$

with σ_{CNR_i} being the CNR standard deviation of the *i-th* satellite, the minimum σ_{CNR} is calculated in each epoch and the maximum σ_{CNR} is a fixed value obtained by calculating the maximum σ_{CNR} of all the epochs.

III. THE PROPOSED ALGORITHM

Following the definition of the weight function, the same can be incorporated within the SUM. Before going through the SUM process, the weight of each satellite is calculated in order to create the W matrix, containing all the satellite weights. Afterwards, the weight matrix is used in the calculation of the Weighted Position Dilution of Precision (WPDOP) [16].

$$WPDOP = \sqrt{trace((G^T W G)^{-1})} \quad (14)$$

 ISBN: 978-1-68558-089-6

$$G = \begin{bmatrix} e_x^{i,1} & e_y^{i,1} & e_z^{i,1} \\ \vdots & \vdots & \vdots \\ e_x^{i,m} & e_y^{i,m} & e_z^{i,m} \\ e_x^{j,1} & e_y^{j,1} & e_z^{j,1} \\ \vdots & \vdots & \vdots \\ e_x^{j,n} & e_y^{j,n} & e_z^{j,n} \end{bmatrix} \quad (15)$$

$$W = \begin{bmatrix} w_1 & 0 & 0 & 0 & 0 \\ 0 & \ddots & \dots & \dots & 0 \\ 0 & \vdots & \ddots & \vdots & 0 \\ 0 & \vdots & \dots & \ddots & 0 \\ 0 & 0 & 0 & 0 & w_n \end{bmatrix} \quad (16)$$

By using the WPDOP for subsets comparisons in place of the GDOP, the positional accuracy is emphasized instead of the geometry of the satellites in the selected subset. Furthermore, the application of weights enhances the versatility and robustness since it takes in consideration multiple factors, hence making it a holistic algorithm. The proposed approach is labeled as Weighted Sequential Updating Method (WSUM) and takes the following steps:

1. Calculate the weight for each visible satellite.
2. Select the 5 satellites with the highest weight to be used as the initial subset. Making the initial subset bigger will reduce the number of iterations, therefore reducing the computational effort.
3. Go through the process of the Sequential Updating Method to find *m* subsets with *k* satellites.
4. Use the subset with minimal WPDOP for positioning.

IV. RESULTS AND ANALYSIS

GNSS data retrieved by the Vehicle Motion and Position Sensor (VMPS) designed by Bosch [17] installed in a car was used to evaluate the performance of the proposed algorithm. MATLAB was the software used to process the data and analyze results. The PC hardware is comprised of a 11th Generation Intel Core i7-11850H @ 2.50GHz, a NVIDIA RTX A3000 GPU and 32 GB of RAM. The location of the dataset collection was in Braga (Portugal) and the car goes through the urban environment of the city of Braga as well as some highway like roads. The GPS, GLONASS and Galileo constellations were considered in the experiments and Single Point Positioning (SPP) applying Kalman Filter was used to calculate the position. This dataset presents different types of conditions for the proposed algorithm to be tested, in order to verify its versatility and robustness. The optimal method, SUM, Ultra-Rapid and a smartphone-based satellite selection were also tested for comparison purposes.

The path of the car is shown in Figure 2, where the blue points coordinates were collected by the device iMar iTraceRT-MVT 600, that allows to obtain errors from the positioning system under test, to be used as a reference [19]. Throughout the experiment, there are between 0 and 23 visible satellites, as shown in Figure 3. The car went through some tunnels; therefore, no measurements were collected at some instances during the test.

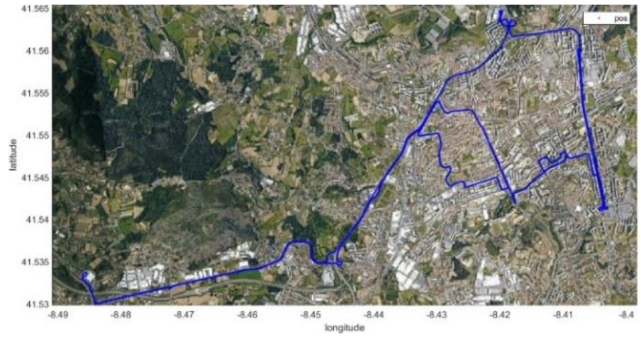

Figure 2. Path that the car went through in the city of Braga, Portugal (Satellite Image by Google Maps ©).

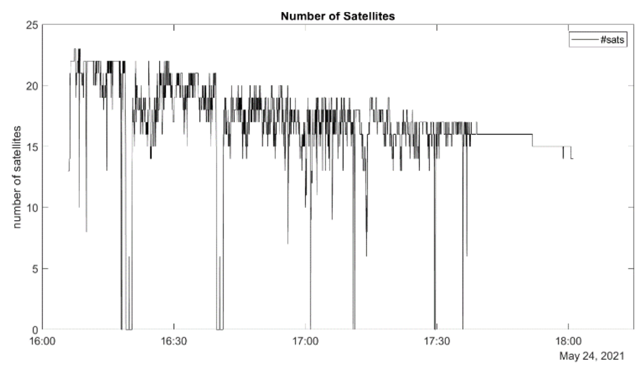

Figure 3. Number of visible satellites along the experiment.

Figures 4 and 5 show the empirical CDF of horizontal and vertical positioning errors obtained with the all-in-view (no satellite selection) solution and the four methods introduced. The WSUM exhibits an overall better horizontal error distribution compared to the other methods, while in terms of vertical error, the proposed method provides an overall better distribution until the 90th percentile.

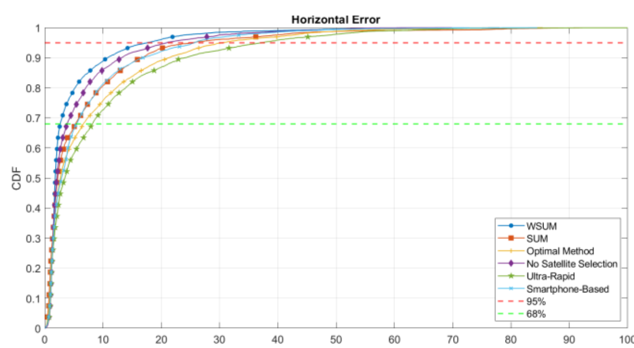

Figure 4. Empirical Cumulative Distribution Function (CDF) of horizontal positioning error for various satellite selection algorithms when selecting 9 satellites and no satellite selection.

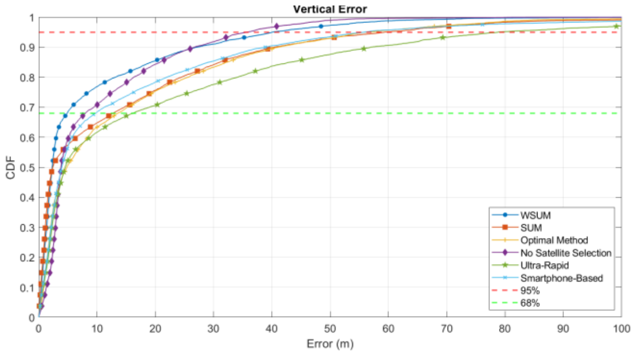

Figure 5. Empirical CDF of vertical positioning error for various satellite selection algorithms when selecting 9 satellites and no satellite selection.

Figures 6 and 7 display the distribution of the horizontal and vertical errors, respectively, where the y-axis is the percentage of trials each error range is obtained. WSUM demonstrates a higher percentage of errors between 0 and 2 meters in the horizontal category it slightly falls short of the performance exhibited by the SUM algorithm.

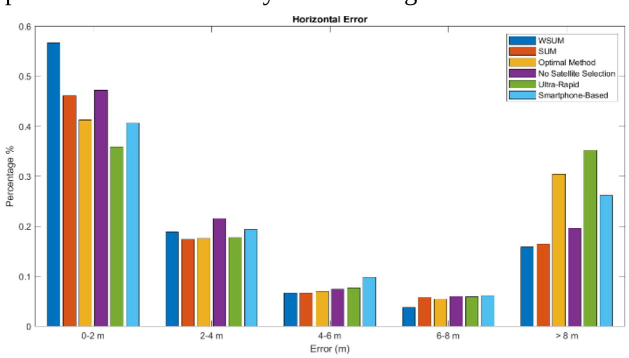

Figure 6. Percentage of trials each horizontal positional error range is obtained when selecting 9 satellites and no satellite selection.

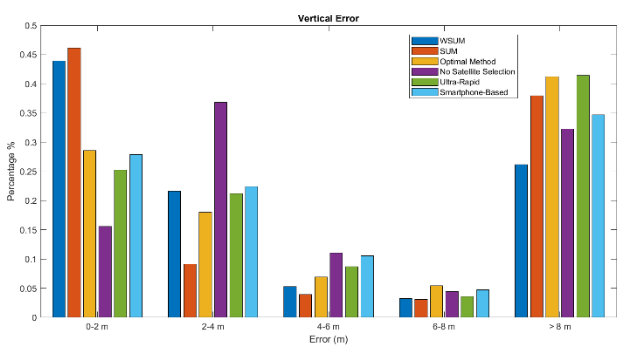

Figure 7. Percentage of trials each vertical positional error range is obtained when selecting 9 satellites and no satellite selection.

The WSUM presents 4.33 m and 8.52 m in terms of mean horizontal and vertical errors, while without satellite selection mean horizontal and vertical errors of 5.26 m and 9.52 m were obtained, respectively. The SUM, Optimal method, Ultra-Rapid and Smartphone-Based algorithms provide 6.56 m, 7.65 m, 8.93 m, and 6.25 m values of mean horizontal error, respectively, and 13.69 m, 14.30 m, 18.88 m, and 13.35 m values of mean vertical error, respectively.

Stability and computational effort are also important performance indicators of a satellite selection algorithm. Computational time is measured through the 'tic-toc' MATLAB functions and several runs were made in order to obtain the average computational time, in seconds, of each algorithm. The stability metric is measured as the ratio between number of satellites changed/removed due to algorithm decision from epoch *n* to epoch *n+1* (ΔN), and number of satellites used in epoch *n* ($Nsat_n$):

$$Stability = \left(1 - \frac{\Delta N}{Nsat_n}\right) * 100 \qquad (17)$$

Table I shows that the increase in the size of the initial subset from 4 to 5 slightly decreases the computational time over the SUM algorithm, therefore making it faster than the original method. Stability is also displayed and while the proposed algorithm presents a lower value than SUM, it can still be labeled as highly stable.

TABLE I. STABILITY AND COMPUTATIONAL TIME OF ALL THE ALGORITHMS WHEN SELECTING 9 SATELLITES

Algorithm	Computational Time (s)	Stability
Optimal Method	**0.72261**	**94.78%**
SUM	**0.00604**	**96.63%**
Ultra-Rapid	**0.00036**	**95.51%**
Smartphone-Based	**0.00017**	**90.44%**
WSUM (Proposed)	**0.00447**	**93.55%**

When it comes to the empirical CDF of the GDOP and PDOP values, Figures 8 and 9 indicate that WSUM provides the highest values. This result is justified, given that the proposed algorithm uses the WPDOP in the comparison, thus the final chosen subset does not necessarily have the lowest values in terms of PDOP and GDOP. Furthermore, the high accuracy presented by the proposed method emphasizes the fact that a good satellite geometry does not directly translate to a good positional accuracy, and this factor by itself is not sufficient to select the best possible subset to be used in positioning.

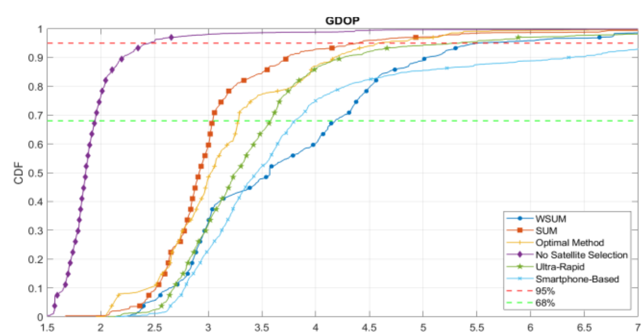

Figure 8. Empirical CDF of the GDOP for various satellite selection algorithms when selecting 9 satellites and no satellite selection.

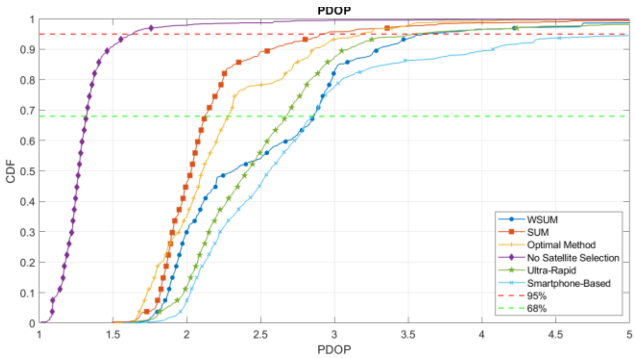

Figure 9. Empirical CDF of the PDOP for various satellite selection algorithms when selecting 9 satellites and no satellite selection.

V. CONCLUSION

In this paper, we propose a satellite selection algorithm capable of being used in dynamic scenarios in order to optimize a solution for autonomous driving. The proposed algorithm excels in providing better accuracy compared to other algorithms from the literature, as well as using all visible satellites. The computational effort was greatly minimized from the optimal method and the increase of the size of the initial subset of satellites provided a slight decrease in computational time compared to the original SUM method. The addition of the weight function did not increase the computational complexity of the proposed algorithm in any substantial way. Furthermore, it enhances the performance and versatility by making it a holistic satellite selection algorithm. Results show that the WSUM provides an improvement of 0.93 m and 1 m in terms of average horizontal and vertical error, respectively, over the use of all the visible satellites available at each epoch. Further optimizations to the proposed algorithm can be made in terms of computational effort and the implementation of Precise Point Positioning (PPP) will be made alongside the use of correction services in order to obtain higher levels of accuracy. Finally, the algorithm discussed in this paper presents a novel method that can improve positioning accuracy in challenging environments, therefore making it a promising solution to be incorporated in Highly Automated Driving vehicles.

ACKNOWLEDGMENT

This work is supported by the European Structural and Investment Funds in the FEDER component, through the Operational Competitiveness and Internationalization Programme (COMPETE 2020) [Project nº 179491; Funding Reference: SIFN-01-9999-FN-179491].

REFERENCES

[1] M. Karaim, M. Elsheikh, and A. Noureldin, GNSS Error Sources. Multifunctional Operation and Application of GPS. InTech 2018. doi: 10.5772/intechopen.75493.

[2] SAE International, "Taxonomy and definitions for terms related to on-road motor vehicle automated driving systems", 2021.

[3] ISO, "ISO/IEC TR 24027:2021 - Information technology — Artificial intelligence (AI) — Bias in AI systems and AI aided decision making", Retrieved October 11, 2023, [Online]. Available: https://www.iso.org/standard/77607.html.

[4] International Organization for Standardization, Road Vehicles – Functional Safety (ISO/DIS Standard No. 26262-6), 2018, Retrieved from https://www.iso.org/standard/68388.html

[5] E. Kaplan and C. Hegarty, Understanding GPS/GNSS: Principles and Applications, Third Edition, Artech, 2017.

[6] J. J. Spilker and B. W. Parkinson, Global Positioning System: Theory and Applications vol. 1, AAIA, 1996.

[7] C. Chi, X. Zhan, T. Wu, and X. Zhang," Ultra-Rapid Direct Satellite Selection Algorithm for Multi-GNSS", Proceedings of the International Conference on Aerospace System Science and Engineering 2019, ICASSE 2019, pp. 11–25. https://doi.org/10.1007/978-981-15-1773-0_2

[8] M. F. ElKhalea, H. M. Hendy, A. M. Kamel, I. I. Arafa, and A. Abosekeen, "Smartphone Positioning Enhancement Using Several GNSS Satellite Selection Techniques," 2022 5th International Conference on Communications, Signal Processing, and their Applications (ICCSPA), Cairo, Egypt, 2022, pp. 1-6, doi: 10.1109/ICCSPA55860.2022.10019004.

[9] A. Peng, G. Ou, and G. Li, "Fast satellite selection method for multi-constellation Global Navigation Satellite System under obstacle environments," IET Radar, Sonar & Navigation, vol. 8, no. 9, pp. 1051–1058, Dec. 2014, doi: 10.1049/iet-rsn.2013.0387.

[10] B. Wang, S. Wang, L. Miao, and S. Jun, An Improved Satellite Selection Method in Attitude Determination using Global Positioning System, Recent Patent on Space Technology, 2009.

[11] M. S. Braasch, "Multipath Effects, Global Positioning Systems: Theory and Applications", American Institute of Aeronautics and Astronautics, 1996, pp. 547-568.

[12] V. S. Srinivas, A. D. Sarma, and A. S. Reddy, "Weighted Quasi-optimal and Recursive Quasi-optimal Satellite Selection Techniques for GNSS", 2015.

[13] T. Takasu, RTKLIB: Open Source Program Package for RTK-GPS, FOSS4G 2009 Tokyo, Japan, November 2, 2009.

[14] N. Naciri and S. Bisnath, "An uncombined triple-frequency user implementation of the decoupled clock model for PPP-AR", *J Geod 95*, 2021, https://doi.org/10.1007/s00190-021-01510-y

[15] H. Tokura and N. Kubo, "Efficient Satellite Selection Method for Instantaneous RTK-GNSS in Challenging Environments," Trans Jpn Soc Aeronaut Space Sci", vol. 60, no. 4, pp. 221–229, 2017, doi: 10.2322/tjsass.60.221.

[16] E. R. Matera, A. G. Pena, C. Milner, and B. Ekambi, "Smart Exploitation of Pseudorange and Pseudorange-rate Error Characterization to Improve the PVT Solution", *32nd International Technical Meeting of the Satellite Division of The Institute of Navigation*, 2019, doi:10.33012/2019.17109

[17] Vehicle motion and position sensor, Retrieved October 11, 2023, from https://www.bosch-mobility.com/en/solutions/sensors/vehicle-motion-and-position-sensor/

[18] C. Pendão, A. G. Ferreira, A. Moreira, C. Martins, and H. Silva, "Challenges in characterization of GNSS precise positioning systems for automotive", CEUR Workshop Proceedings", 2020.