Mitigating Jamming and Meaconing Attacks Using Direct GPS Positioning

Yuting Ng, Student Member, IEEE and Grace Xingxin Gao, Senior Member, IEEE

Abstract—Direct GPS Positioning (DP) is a robust method that estimates the GPS navigation solution directly from the received GPS signal. In contrast, traditional methods, such as scalar tracking and vector tracking, compute intermediate channel range and range residual measurements independently, then use these as inputs into the navigation filter to estimate the navigation solution. However, a brute force implementation of DP is not computationally efficient. Our research contribution is to evaluate the robustness of DP using our more computationally efficient DP implementation.

Our novel and effective DP receiver architecture initializes the navigation solution guesses as two groups: guesses varying in position and clock bias, guesses varying in velocity and clock drift. We then perform vectorized calculations to get the expected delay and doppler between the receiver and each satellite in view. Following that, we perform batch calculations using Fast Fourier Transforms (FFTs) to obtain the vector correlation and vector spectrum. The navigation solution residual derived from the vector correlation and vector spectrum is then used as input into the navigation filter.

We implemented our receiver architecture using our research platform - PyGNSS. We then evaluated the robustness of our DP receiver architecture by subjecting it to simulated jamming and meaconing attacks. We demonstrate through our experiments the robustness of our DP receiver architecture.

I. INTRODUCTION

Direct GPS Positioning (DP) directly estimates and tracks the navigation solution from the received raw GPS signal [1], [2]. This is unlike traditional scalar tracking which estimates channel range and range rate measurements independently, then solves for the navigation solution via trilateration [3]–[7]. Similarly, traditional vector tracking also produces the channel range and range rate residuals independently [8]–[11]. Channel measurements estimated using traditional methods are degraded under signal multipath and obstruction [12], [13], leading to errors in the navigation solution. On the other hand, the shape of the vector correlation in DP is robust to signal multipath and obstruction, leading to a more robust navigation solution [14], [15].

In our prior work, we have implemented a different DP receiver architecture from the DP receiver architecture presented in this paper. That architecture uses coherent vector correlator measurements with the Unscented Kalman Filter (UKF). Scalar tracking provided the carrier phase measurements to enable the coherent vector correlator. Live experiments were conducted and the results were analyzed for both static and dynamic scenarios. In addition, we have provided a theoretical proof that direct acquisition is more robust than scalar acquisition under meaconing scenarios. We also showed that direct positioning is more robust to jamming than scalar positioning using simulated noise [1].

Work by other researchers can be grouped into two categories: direct acquisition and tracking. Direct acquisition, also known as collective detection can be further separated into coherent and noncoherent direct acquisition [2], [14]–[18]. Coherent direct acquisition is too costly [2]. Noncoherent direct acquisition has lower gain than coherent direct acquisition. However, the noncoherent method involved less computation and was shown to have superior detection to scalar acquisition [14], [15]. MATLAB implementations of direct acquisition was provided in [16], [17]. Simulations were used to compare the performance of direct acquisition against scalar acquisition by [19], [20]. A mathematical treatment was provided by [21]–[23]. With regards to tracking, a tracking loop for the vector correlator was devised by [24], based on prior work [25]. Also basing their work on a tracking loop was [26], [27].

Many of the above implementations were computationally intensive, did not estimate and track the full navigation parameters and did not provide a dedicated evaluation of DP’s robustness. Therefore, using our computationally efficient implementation of DP, we provide a dedicated evaluation of DP’s robustness through simulated jamming and meaconing attacks.

Biographies

Yuting Ng is a graduate student in the Aerospace Engineering Department at the University of Illinois at Urbana-Champaign. She received her B.S. degree in electrical engineering, graduating with university honors, from the same university. Her research interests are in advanced signal tracking algorithms, navigation, control, RADAR and UAVs.

Grace Xingxin Gao received her B.S. degree in mechanical engineering and her M.S. degree in electrical engineering from Tsinghua University, Beijing, China in 2001 and 2003. She received her PhD degree in electrical engineering from Stanford University in 2008. From 2008 to 2012, she was a research associate at Stanford University. Since 2012, she has been an assistant professor in the Aerospace Engineering Department at University of Illinois at Urbana-Champaign. Her research interests are systems, signals, control, and robotics. She is a senior member of IEEE and a member of ION.

Yuting Ng and Grace Xingxin Gao are with the Department of Aerospace Engineering at the University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA. http://gracegao.ee.illinois.edu/
The rest of the paper is organized as follows. Section II of the paper describes our computationally efficient DP receiver architecture used in this paper, with focus on obtaining the vector correlator and vector spectrum measurements, followed by subsequent tracking using a navigation filter. Section III of the paper discusses our implementation and experiments based on field data. We subjected our DP receiver to simulated jamming and meaconing through post-processing using our research platform - PyGNSS. We then compare the robustness of DP against scalar tracking, in Section IV. Finally, Section V summarizes the paper.

II. DIRECT GPS POSITIONING

DP estimates and tracks the underlying signal and navigation state parameter set of receiver position, clock bias, velocity and clock drift directly from the received raw GPS signal. In other words, we are searching for the underlying velocity and clock drift directly from the received raw GPS signal. We then compare the robustness of DP against scalar tracking, in Section IV. Finally, Section V summarizes the paper.

A. Efficient Search for Best Guess Navigation Solution

To speed up the search process, we first split the underlying parameter set into two subsets:

\[ X_{xyz,c} \] : 3D position and clock bias parameters
\[ = [x, y, z, c] \]
\[ X_{\dot{x},\dot{y},\dot{z},c} \] : 3D position and clock bias parameters
\[ = [\dot{x}, \dot{y}, \dot{z}, c] \]
\[ c \] : speed of light, 299792458 \((m/s)\)

Let \( Y \) be the received GPS signal, which is a superposition of signals received from different satellites. To search for the best estimate of the receiver state \( X \) that most likely generated the observed GPS signal \( Y \), we perform the following operations:

\[ Y(t) = \sum_i D_i(t) G_i(s(t_i) + \phi_{i,\text{code}}) e^{j2\pi f(t_i) + \phi_{i,carr}} \]

where \( Y(t) \) is the received signal, \( D_i(t) \) is the databit sequence of the \( i \)th satellite, \( G_i(t) \) is the L1 C/A code sequence of the \( i \)th satellite, and \( f_i \) is the carrier frequency of the \( i \)th satellite.

After performing carrier and code wipeoff, the received GPS signal is expressed as:

\[ Y_{\text{code wipeoff}}^i = Y_{\text{code}}^i \approx G_i^i f_{\text{code}}^i t + \phi_{\text{code}}^i \]

\[ Y_{\text{carr wipeoff}}^i = Y_{\text{carr}}^i \approx e^{j2\pi f_{\text{carr}}^i t + \phi_{\text{carr}}^i} \]

Estimation of the \( X_{xyz,c} \) parameter subset was achieved through performing one correlation per channel. The code phase of the signal replica, which the received signal was correlated against, was first shifted by an amount determined by the current 3D position and clock bias parameters. See equation (4) for the relationship between the code phase and the 3D position and clock bias parameters. This shift allows us to directly estimate the 3D position and clock bias residual \( \Delta X_{xyz,c} \).

Estimation of the \( X_{\dot{x},\dot{y},\dot{z},c} \) parameter subset was achieved through performing one fourier transform per channel. The carrier phase of the received signal was first shifted
by an amount determined by the current 3D velocity and clock drift parameters. See equation (6) for the relationship between the carrier frequency and the 3D velocity and clock drift parameters. This shift allows us to directly estimate the velocity and clock drift residual X˙x X˙y X˙z,ct.

For each channel, the normalized correlation magnitude and frequency spectrum is shown in Fig.1. The triangular correlation peak and sinc spectrum peak would be shifted proportional to Xxyz,ct and X˙x X˙y X˙z,ct residuals. The width of the triangular correlation is related to the code chip duration while the width of the sinc mainlobe is related to the coherent processing interval. The mathematical expressions describing the correlation and spectrum peak is given as follows:

\[
\mathcal{R}_{\text{code only}}(\Phi) = \begin{cases} 
(1 + (\Phi - \Delta \Phi)) & , -1 < (\Phi - \Delta \Phi) \leq 0 \\
(1 - (\Phi - \Delta \Phi)) & , 0 < (\Phi - \Delta \Phi) \leq 1 \\
0 & , \text{otherwise}
\end{cases} 
\]

\[
\mathcal{F}_{\text{curr only}} = \text{sinc}(\pi(f - \Delta f)T_{coh}) \approx \begin{cases} 
\sin(\pi(f - \Delta f)T_{coh}) & , -1 < (f - \Delta f) \leq 0 \\
\pi(f - \Delta f) & , 0 < (f - \Delta f) \leq 1 \\
0 & , \text{otherwise}
\end{cases} 
\]

\[
\Delta \Phi : \text{residual code phase shift due to } X_{xyz,ct} \\
\Delta f : \text{residual carrier frequency shift due to } X_{xyz,ct} \\
T_{coh} : \text{coherent integration period, 0.020s}
\]

To reiterate, the correlations help search through the first parameter set, \((x, y, z, c, t)\), while the frequency spectrums help search through the second parameter set, \((\dot{x}, \dot{y}, \dot{z}, \dot{c}, \dot{t})\). As such, carrier wipeoff is performed before the correlations and code wipeoff is performed before the Fourier transform. Our DP implementation in this paper utilized a coherent integration period of 20ms. The second step is generating the candidate receiver state error vectors, with two sets, fixed velocity and fixed position. The third step is assigning correlation amplitudes and spectrum magnitudes from the correlations and spectrums to the respective candidate receiver state error vectors. The noncoherent sum of the correlation amplitudes and spectrum magnitudes across each channel forms the vector correlation and vector spectrum. The fourth step estimates the error input vector \(e\) as the weighted mean of the candidate receiver state error vectors. The weights are determined from the vector correlation and vector spectrum. In this effective and efficient manner, the input to the joint tracking and navigation filter is generated.

B. Estimation and Filtering

The joint tracking and navigation filter used in our effective and computationally efficient implementation of DP is a Kalman Filter (KF) [28], [29]. Since we are obtaining direct measurements of the state parameters, the geometry matrix is an identity matrix and the equations can be drastically simplified.
The KF measurement update at epoch \( k \):

\[
e = \text{error input vector} \quad (14)
\]

\[
e = e(\Delta X_{x,y,z,c}, \Delta X_{\tilde{x},\tilde{y},\tilde{z},\tilde{c}}) = [\Delta x, \Delta y, \Delta z, \Delta \phi_{t}, \Delta \dot{x}, \Delta \dot{y}, \Delta \dot{z}, \Delta \phi_{t}]^T
\]

\[
W = \text{error noise covariance input matrix} \quad (15)
\]

\[
W = W(\Delta X_{x,y,z,c}, \Delta X_{\tilde{x},\tilde{y},\tilde{z},\tilde{c}})
\]

\[
K = \text{Kalman gain matrix} \quad (16)
\]

\[
K = \Sigma_k H^T (H \Sigma_k H^T + W)^{-1}
\]

\[
\Delta X = \text{state error vector} \quad (17)
\]

\[
\hat{X}_k = X_k + \Delta X
\]

\[
\Sigma_k = \text{corrected state error covariance matrix} \quad (20)
\]

\[
\Sigma_k = (I - KH) \Sigma_k
\]

The KF update interval \( \Delta T \) is set to be the same as the coherent integration interval \( T_{coh} \). Having \( \Delta T = T_{coh} \) removes the need for measurement interpolation. Using our flexible research platform - PyGNSS, we are not limited to a \( T_{coh} \) of 20 milliseconds, as shown in our work on joint GPS and vision direct positioning [30]. A \( T_{coh} \) of 20 milliseconds was selected for this paper as it allows us to conveniently compare our results against literature. A \( T_{coh} \) of 20 milliseconds was also used in our work on Direct Time Estimation [31]. We performed navigation bit wipe-off such that the KF update interval and coherent integration period is synchronous across channels and can straddle navigation bit boundaries.

The KF time update equations at epoch \( k + 1 \):

\[
\hat{X}_{k+1} = \text{predicted state vector} \quad (21)
\]

\[
\hat{X}_{k+1} = FX_k
\]

\[
\hat{\Sigma}_{k+1} = \text{predicted state error covariance matrix} \quad (22)
\]

\[
\hat{\Sigma}_{k+1} = F\Sigma_k F^T + Q_k
\]

\[
F = \text{state propagation matrix} \quad (23)
\]

\[
F = F(\Delta T)
\]

\[
= \begin{bmatrix}
1 & 0 & 0 & 0 & \Delta T & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & \Delta T & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & \Delta T & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & \Delta T \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}
\]

\[
\Delta T = \text{update interval, 0.020 (s)} \quad (24)
\]

\[
Q_k = \text{state process noise covariance matrix} \quad (25)
\]

\[
Q = F \begin{bmatrix}
0 & 0 & 0 \\
0 & 0 & Q_{v,k}
\end{bmatrix} F^T
\]

\[
Q_{v,k} = \text{velocity and clock drift component of } Q_k \quad (26)
\]

\[
= \begin{bmatrix}
f(v_k) & 0 & 0 & 0 \\
0 & f(v_k) & 0 & 0 \\
0 & 0 & f(v_k) & 0 \\
0 & 0 & 0 & (c \times \sigma_{\tau})^2
\end{bmatrix}
\]

\[
\sigma_{\tau} = \text{allan deviation of the frontend oscillator, (s)}
\]

\[
f(v_k) = \text{saturation function for velocity } v_k \quad (27)
\]

\[
= 1 + 250/(\min(max(v_k^2, 5^2), 25^2))
\]

The initial predicted state error covariance matrix \( \Sigma_0 \) is initialized using 20 past state vectors and should be similar in order of magnitude to the initial state process noise covariance matrix \( Q_0 \). In addition, since we are using a constant acceleration model, for dynamic scenarios such as that on a road vehicle, vehicle accelerations are addressed through a time-varying state process noise covariance matrix, \( Q_0 \) [32], [33], set using a saturation function \( f(v_k) \) on a 20 sample running average amplitude of the receiver’s velocity \( v = ||\dot{x}, \dot{y}, \dot{z}|| \). The coefficients of the saturation function were determined from a least squares fit to acceleration time data of a generic vehicle by [33]. The allan deviation of the frontend oscillator can be obtained through the manufacturer’s datasheet [34].

### III. IMPLEMENTATION AND EXPERIMENT SETUP

This paper evaluates the robustness of DP under stressful conditions. We do so using simulated jamming and meaconing input signals and our research platform - PyGNSS.

We generated the input signals in the following manner:

1) A “clean” signal is first collected under an ideal, open-sky environment, jamming and meaconing is absent.

2) The “clean” signal is then processed to simulate two significant forms of attacks, jamming and meaconing.

The methods used to simulate these attacks are discussed below. In addition, in order to provide a deeper analysis, the “severity” of the attacks is adjusted by changing the amplitude of the attacking signal.

3) The attacked signal is then fed into both DP and scalar tracking receiver architectures. A comparison of the tracking results is then provided.

We simulated the following two attacks on the clean signal:

- **Jamming**: A random complex voltage \( V = A \cos \Phi + jA \sin \Phi \) is added to each sample, where \( A \sim N(0, \sigma_A) \) and \( \Phi \sim U(0, 2\pi) \). In other words, a white noise signal with Gaussian distributed amplitude and Uniform distributed phase is superposed onto the original clean signal to simulate the jamming effect. In addition, by increasing the standard deviation of the amplitude \( \sigma_A \), we are effectively simulating a stronger jamming source.

- **Mecaconing**: Another signal collected at a separate location and/or a separate time, known as the meaconing signal is superposed on the clean signal. The meaconing signal attempts to deceive the receiver into an incorrect navigation solution by “covering” the original received

\[
\begin{bmatrix}
1 & 0 & 0 & 0 & \Delta T & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & \Delta T & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & \Delta T & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & \Delta T \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}
\]
signal. In our simulation, the meaconing signal is collected from a receiver location 1.25 miles away from the original signal.

IV. RESULTS AND ANALYSIS

During a regular tracking scenario where there is neither meaconing nor jamming, both DP and scalar tracking receiver architectures easily track both code and carrier frequencies, $f_c$ and $f_i$, successfully. We then evaluate how jamming and meaconing attacks impact the performance of DP as well as scalar tracking.

1) Jamming: As the noise floor is raised by increasing the jamming amplitude through the parameter $\sigma_A$, DP remains robust whereas the tracking performance of scalar tracking is degraded. This is shown in Fig.3 where the results from different jamming scenarios are color-coded and superposed.

2) Meaconing: A meaconing signal from 1.25 miles away is unable to significantly deceive the DP receiver that is already tracking onto the authentic signal. In order to deceive the DP receiver, the meaconing signal has to first fall within the main lobe then have similar process parameters. As such, it is expected for the DP receiver to remain robust to the above mentioned meaconing signal. The meaconing signal then acts like a jamming signal. Results are shown in Fig.4.

V. CONCLUSION

Direct Positioning (DP) has the potential to outperform traditional methods, such as scalar tracking and vector tracking, in terms of robustness. This is because it utilizes information from the entire raw received signal to generate direct measurements of the navigation solution. However, it is not widely implemented as previous implementations of DP are not computationally efficient. To shake up the status-quo, we proposed and implemented a novel, effective and computationally efficient DP receiver architecture and went on to provide a dedicated evaluation of its robustness.

We subjected our DP receiver architecture to simulated jamming and meaconing attacks. We did so by collecting GPS signals in benign environments then post-processing them into attacked signals using our research platform - PyGNSS. We input the attacked signals into our DP receiver architecture, benchmarking the tracking results against scalar tracking. We demonstrated with our DP receiver architecture, DP's robustness, as compared to scalar tracking, with respect to jamming and meaconing attacks.

ACKNOWLEDGMENT

Special thanks to Ganshun Lim for his continuous support, Arthur Hsi-Ping Chu for running the jamming and meaconing simulations, Sri Ramya Bhamidipati for kindly proofreading the paper.

REFERENCES