Timing Evaluation of Energy-Based Packet Detection Algorithms on SDRs for Spectrum Sensing Applications

1st Raquel Marina Noguera Oishi
Department of Electrical Engineering (ESAT)
KU Leuven
Leuven, Belgium
raquelmarina.nogueraoishi@kuleuven.be

2nd Franco Minucci
Department of Electrical Engineering (ESAT)
KU Leuven
Leuven, Belgium
franco.minucci@kuleuven.be

3rd Yago Lizarribar Carrillo
Cyber Defense Campus
Armasuisse
Thun, Switzerland
yago.lizarribarcarrillo@armasuisse.ch

4th Gérôme Bovet
Cyber Defense Campus
Armasuisse
Thun, Switzerland
gerome.bovet@armasuisse.ch

5th Sofie Pollin
Department of Electrical Engineering (ESAT)
KU Leuven
Leuven, Belgium
sofie.pollin@kuleuven.be

***Abstract*—With the increasing interest in non-terrestrial networks, spectrum sensing at various altitudes has become an active area of research, particularly in understanding how spectrum occupancy differs from ground level. High-altitude balloon platforms offer a cost-effective and promising solution for collecting wide-area spectrum data at different elevations. However, their limited processing power and storage impose strict constraints on real-time signal detection and timestamping, crucial for applications such as signal localisation, interference mapping, or dynamic spectrum sharing.**

This paper presents an initial study towards developing a compact, low-power spectrum sensing probe for high-altitude balloons. We focus on one of the key technical challenges: accurately detecting radio transmissions and assigning precise timestamps using energy-based methods. Specifically, we evaluate and compare the performance of two energy-based packet detection algorithms (the single sliding window and the double sliding window) in terms of their timing accuracy and computational complexity using the USRP E320 platform. To validate signal-agnosticism, we assess detection performance across three diverse signal types (OFDM, ADS-B, and GFSK). We isolate and characterise the intrinsic error introduced by different timing error sources, independently of other system delays, offering a clear benchmark of achievable accuracy. Our analysis reveals the trade-offs between detection reliability, timestamping precision, and implementation cost, providing a baseline for the design of future high-altitude probes for spectrum sensing.

***Index Terms*—wireless, energy detection, timestamp, software-defined radio**

I. Introduction

The rapid growth of non-terrestrial networks, including satellite systems and high-altitude platforms (HAPs), highlights the need for spectrum sensing and signal localisation capabilities that operate reliably across a wide range of altitudes. These capabilities are especially valuable to regulatory bodies, telecom operators, and research organisations that seek to understand radio usage patterns beyond ground level, for example, during emergency deployments, shared-spectrum policy enforcement, or for monitoring and surveillance [1]. Compared to UAVs (e.g., drones), HAPs offer significantly longer flight durations, wider coverage areas, and greater endurance, making them suitable for passive, long-term RF monitoring. Among HAPs, helium or hydrogen balloons offer a cost-effective, lightweight, and energy-efficient solution for large-scale radio environment monitoring [2], making them attractive for applications such as interference mapping, spectrum occupancy surveys, and passive signal geolocation [1].

However, operating at high altitudes introduces several challenges, including dynamic channel conditions, Doppler shifts, fluctuating signal strengths, and limited hardware capabilities due to power, size, and weight constraints [3], [4]. In particular, the limited onboard storage and write speeds of balloon-borne RF probes make it infeasible to continuously record raw spectrum data. This necessitates onboard real-time detection and timestamping of meaningful transmissions, so only relevant signal events are stored and analysed. To achieve this, a robust and flexible packet detection algorithm is required.

Existing packet detection systems often rely on correlation-based detection, which is highly accurate but requires prior knowledge of the signal's structure, an unrealistic assumption in dynamic, shared-spectrum environments (e.g., ISM bands or uncoordinated deployments). Instead, energy-based detection offers a signal-agnostic approach that can handle diverse modulation schemes and unknown packet formats, generally at the cost of reduced accuracy and greater sensitivity to noise and threshold settings, especially in low-SNR scenarios [5].

To address these challenges, we adopt a sliding window

approach, single and double sliding window algorithms, that perform the decision based on the cumulative energy through one or two windows of samples. The sliding window makes the energy detection more robust to noise and simplifies the computation compared to non-sliding algorithms [6]. Their performance depends on parameters such as window size and thresholds, which influence not only detection rate but also the accuracy of packet timing, a critical factor for time-based applications.

Accurate timestamping of detected packets is essential for a wide range of applications, especially localisation using time-of-arrival (TOA) and time-difference-of-arrival (TDOA) techniques [7], [8]. However, timing precision is also crucial for reliable multi-sensor fusion, spectral occupancy tracking, and distributed sensing coordination [9]. While timestamping precision is widely recognised as critical, the specific role of packet detection accuracy in determining timestamp fidelity, and thereby localisation performance, has received limited direct attention in the literature [10], [11].

In our work, we evaluate energy-based detection algorithms on the USRP E320 platform, with a specific focus on their timing accuracy and implementation cost, particularly for resource-constrained balloon-based sensing devices. While these packet detection algorithms have been studied in the literature for specific technologies [6], prior studies have not systematically analysed how detection behaviour affects timestamp precision across heterogeneous signals. Our contribution addresses this gap through a comprehensive timing analysis that isolates and quantifies the different sources of error in timestamp-based systems. In particular:

- We disentangle the intrinsic timing error introduced by imperfect energy-based packet detection from other sources of delay (e.g., software or hardware delays), providing a clear understanding of the best achievable timing accuracy purely from the detection algorithm. This allows us to establish a baseline performance limit against which other timing inaccuracies can be compared or calibrated.
- We demonstrate how detection and other processing delays, often overlooked in detection systems and considered time-of-flight, can lead to significant timestamp deviations, potentially resulting in substantial errors in time-dependent applications, such as localisation errors based on propagation delay measurements. We provide a detailed characterisation of this system-induced timing offset, using both experimental measurement and calibration.
- We show that while double sliding window detection improves robustness and timing precision across diverse signal types, it also doubles the computational load, a key trade-off for embedded systems.
- Finally, our experiments span three different signal types (OFDM-like, ADSB-B-like, and GFSK-modulated), revealing how detector performance generalises to heterogeneous environments, typical in real-world shared spectrum scenarios.

The remainder of the paper is organised as follows. Section II describes the packet detection algorithms analysed in this paper. Section III explains timestamping challenges in embedded SDRs, focusing on the USRP E320 platform. Section IV presents the experimental results of our study and timing analysis. Finally, Section V gives our conclusions and future directions.

II. Energy-based packet detection

We consider two energy-based packet detection approaches: the single sliding window and the double sliding window mechanisms, as described in [6]. Both methods are based on a decision variable m_n compared against a predefined threshold Th to formulate a hypothesis for packet presence:

$$H_0 : m_n < Th \Rightarrow \text{Packet is absent}$$

$$H_1 : m_n \geq Th \Rightarrow \text{Packet is present}$$

These algorithms detect packets by monitoring signal energy, which is low in the presence of noise and increases when a packet arrives.

A. Single sliding window packet detection

The single sliding window algorithm detects the presence of packets by monitoring variations in the received signal energy over an observation window of length L. This window slides sample-by-sample over the received signal to reduce the number of operations, as shown in 1. At each instant n, one new sample enters the window while one old value is discarded. The decision variable m_n is computed as follows:

$$m_n = \sum_{k=0}^{L-1} |r_{n-k}|^2 \tag{1}$$

$$m_{n+1} = m_n + |r_{n+1}|^2 - |r_{n-L+1}|^2 \tag{2}$$

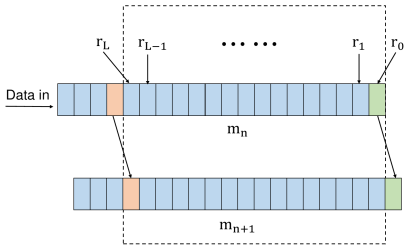

Fig. 1. Single sliding window algorithm

Figure 2 shows the simulation results of the single sliding window over a random signal of 200 samples with an SNR of approximately 5 dB. The figure compares the accumulated energy values for various window sizes. Around index 600, where the signal is injected, the value of the accumulated energy increases. Both the window size and the threshold influence the performance of this method. Larger windows tend to smooth out the noise fluctuations but may also reduce responsiveness to short packets. Moreover, as the decision variable is directly derived from the received signal energy, the method is sensitive to noise, which can lead to false detections under low SNR conditions.

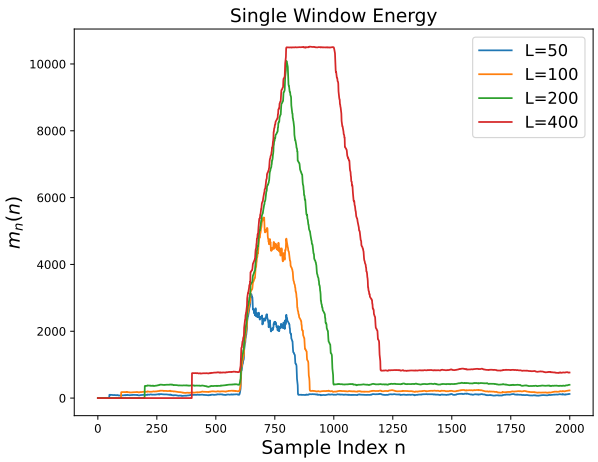

Fig. 2. Simulation of single sliding window energy for different window sizes

B. Double sliding window packet detection

The double sliding window algorithm compares the energy in two consecutive sliding windows A and B of lengths M and L, respectively. The decision variable m_n is computed as the ratio of the energy in the two windows as:

$$a_n = \sum_{k=0}^{M-1} |r_{n-k}|^2 \tag{3}$$

$$b_n = \sum_{k=1}^{L} |r_{n+k}|^2 \tag{4}$$

$$m_n = \frac{a_n}{b_n} \tag{5}$$

Where a_n is the accumulated energy over window A and b_n over window B. Figure 3 shows the expected response of the double window algorithm. When a packet arrives, the energy in window A increases while window B remains low, resulting in the rise of m_n until it reaches a peak. As window B starts to fill, the value of b_n increases, making m_n decrease. Theoretically, the maximum m_n occurs when window A is filled with signal energy and window B still contains noise. In this case, we can obtain the start of the packet by subtracting the window size A from the maximum m_n index.

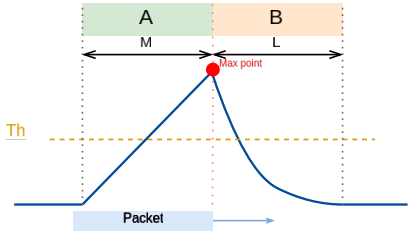

Fig. 3. Double sliding window algorithm

Figure 4 shows the simulation results using different window sizes. The signal is injected from index 1000 to 1200. The results show that smaller window sizes produce sharper peaks in the decision variable, facilitating the detection of the maximum point. Larger windows tend to reduce the influence of the noise but flatten the slope and introduce more ripple near the peak. When the window length exceeds the packet duration (e.g., $M = L = 400$), the decision variable curve becomes flatter, making the detection unreliable. This algorithm is more robust to noise than the single-window algorithm, as the decision is based on a relative energy comparison rather than an absolute value, making it suitable for low SNR conditions at the expense of doubling the required computations.

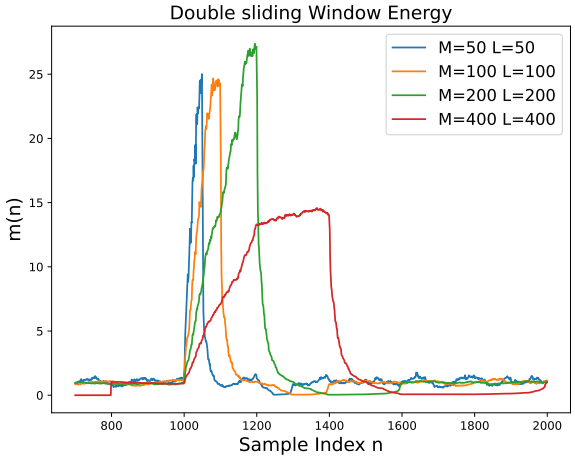

Fig. 4. Simulation of double sliding window energy for different window sizes

III. Aspects of Timestamping Accuracy

Accurate timing is essential for time-based wireless applications. In particular, localisation techniques based on signal propagation time, such as TOA and TDOA, rely on precise reception timestamps to ensure accurate position estimation.

Timestamping assigns a time reference to a detected packet, and its accuracy directly affects the precision of packet detection. There are mainly three types of timestamps: hardware-based, software-based, and firmware-based (or hybrid) [12]. Hardware-based timestamping offers high precision but requires specialised and often costly hardware. Software-based timestamping suffers from significant inaccuracy due to variable processing delays. Firmware-based timestamping provides a practical trade-off by offering precision and cost-effectiveness using programmable platforms.

Timestamping should also occur as close as possible to the medium to avoid non-deterministic delays introduced by higher layers [13]. The timing accuracy of the timestamp is affected by different factors:

- Time base resolution: Defined by the internal clock period. It determines the smallest time interval that can be represented.
- Timestamp resolution: This refers to the granularity with which timestamps can be generated.
- Clock imperfections: Variability in clock behaviour, including skew and jitter, can introduce timestamping deviations.

In our experimental setup, timestamping is performed using the timed commands in the UHD driver. This timestamp mechanism is firmware-based, performing the actual timestamping in the programmable logic of the USRP. On the transmitter side, the assigned timestamp refers to the moment in the future when the packet exits the digital transmit chain. On the receiver side, however, only the first sample in each received buffer is timestamped. To obtain the timestamp corresponding to any other sample in the buffer, we use the relation:

$$T_i = T_{buffer} + Ts \cdot i \tag{6}$$

, where T_i is the time of the target sample, T_{buffer} is the timestamp of the first sample in the buffer, T_s is the sampling period, and i is the index of the sample within the buffer. Any imperfections in the sampling clock will also affect the accuracy of T_i.

Several delays must be considered to determine the actual time of packet reception start, as illustrated in Figure 5. The FPGA delay refers to the internal latency between the timestamp generation and the actual moment the packet is transmitted or received, typically due to pipeline or queuing delays within the digital logic. Analogue components in the transmission and reception chains introduce the delay in the RF frontend. Lastly, on the receiver side, the packet detection delay is the latency introduced by the detection algorithm itself.

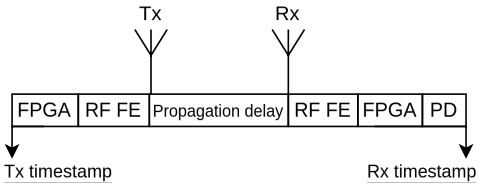

Fig. 5. Delays from Tx to Rx

To characterise the delays in our USRP device, we use a loopback configuration of the Tx and Rx channels via coaxial cable. A square wave signal was transmitted to ensure a sharp rising edge when the signal is received. To detect the signal with high time accuracy, we selected a very low threshold slightly above the noise variance level as shown in Figure 6, so that the first sample exceeding the noise level could be detected.

The measured delays are shown in Figure 7. These delays represent the processing and frontend delays in both Tx and Rx, as well as the propagation delay through the coaxial cable. The master clock frequency used in this case was $F_{CLK} = 50MHz$, so the minimum time step that can be measured is $T_{CLK} = 20ns$. The estimated mean delay is $\bar{\tau} = 3.35\mu s$ with a standard deviation of $\sigma = 17.3ns$. Other measurements we conducted at different clock frequencies revealed that the observed delay is not solely affected by time resolution but is also significantly influenced by the clock frequency itself. This indicates that the FPGA-related delay is frequency-dependent; therefore, delay calibration must be performed at the specific frequency used in the actual measurements. The delay in the coaxial cable was also characterised using a VNA, obtaining an average value of $35.62ns$.

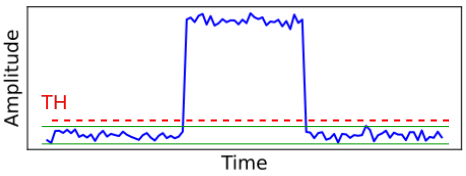

Fig. 6. Threshold selection rule

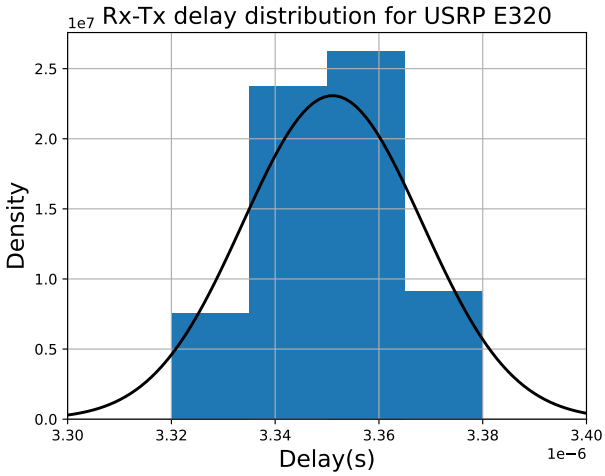

Fig. 7. Characterisation of delay between Tx and Rx in USRP E320 measured at 50MHz.

IV. Experimental results

We evaluated the packet detection algorithms using a USRP E320 in a loopback setup, ensuring the same master clock between Tx and Rx, and avoiding synchronisation issues. Packet transmissions were scheduled periodically using UHD timed commands, and the timestamps were logged. On the receiver side, the incoming signal buffers and their associated timestamps were stored.

To assess the generalizability across different communication protocols, we generated datasets based on three signal types:

- OFDM-like packets: these packets exhibit high peak-to-average power ratios and frequency-domain sparsity. Each packet contains 200 samples.

- ADS-B-like packets: these signals are characterised by sharp transitions between symbols and a return-to-zero structure. Each packet contains 256 samples.
- GFSK-modulated packets: these signals have smooth phase transitions and relatively constant envelope properties. Each packet contains 248 samples.

Each dataset includes approximately 115 transmissions distributed over a time span of 30 seconds.

Single sliding window

This algorithm estimates the packet start at the first index exceeding a predefined threshold, making detection accuracy highly dependent on the threshold level. In this study, the threshold (T_h) was empirically selected as a fixed fraction of the power within the window, allowing for the controlled evaluation of its influence on timing accuracy. We selected threshold levels at 10% and 50% to represent low and high sensitivity detection regimes. The window size also affects the accuracy, as larger windows increase uncertainty in pinpointing the exact start index.

In a practical deployment, the threshold should adapt dynamically to the actual signal and noise conditions to maintain reliable detection across varying SNR levels. The implementation of adaptive thresholding mechanisms is therefore considered as future work.

Double sliding window

In this method, we estimate the packet detection time by locating the maximum point of the decision variable within the interval defined by a threshold. The packet start index is then:

$$i_{packet} = i_{max} - M \tag{7}$$

The corresponding packet start time is derived from the index i_{packet} timestamp. Here, the threshold only defines the search interval and does not impact timing accuracy. To assess this, we tested different thresholds, and the detected peak remained consistent. This indicates that the algorithm yields a clear peak, and the point is to determine the accuracy of this peak within the window.

A. Detection time accuracy

Figure 8 shows the detection time errors for each method. 'SW' and 'DW' refer to the single window and double window algorithms, 'Th' indicates the threshold level (as a percentage of signal power), and 'w' denotes the window size. The detection error for each of the $K = 115$ transmissions is computed as:

$$\Delta T_{PDk} = T_{PDk} - T_{GTk}, \tag{8}$$

where T_{PDk} is the time estimated by the packet detector for the k^{th} packet, and T_{GTk} is the ground truth time of the packet k, obtained as the first sample above the noise as shown in Figure 6. Note that these detections are discrete and inherently subject to quantisation error due to the sampling period.

This error represents only the packet detection error, without considering other delays. While absolute values are plotted, the errors of the double window algorithm can also be negative. This occurs when the peak is detected before window A is filled, causing an underestimation of the true start index after subtracting M.

The results show that the detection accuracy is usually consistent across signal types, making both algorithms suitable for heterogeneous signal environments. As expected, the threshold level and window size significantly impact the accuracy of the single-window algorithm, with time errors reaching up to $3\mu s$. Lower thresholds detect earlier but are more prone to false detections, while higher thresholds yield more stable detection at the cost of increased latency. This highlights the need for careful threshold calibration.

In contrast, the double sliding window method demonstrates better performance, with errors typically in the range of $100 - 200$ ns, and it is also more robust to changes in window size. In general, larger window sizes do not necessarily improve accuracy; in fact, they may blur the packet start by introducing ripple around the peak. However, at lower SNRs, larger windows can provide better noise averaging, potentially improving reliability at the cost of timing precision.

We also evaluated the timing offset between the detected packet and the scheduled transmission time. Since packet detector timestamps include hardware-induced delays (Figure 5), we compensate by subtracting the mean hardware delay measured in our setup (Figure 7), improving the approximation of the true arrival time. However, since the hardware delay exhibits some degree of non-determinism, using a fixed mean also introduces some error. Table I summarises the average detection times relative to the ground truth, the transmission time, and the transmission time after compensation using the mean delay ($\bar{\tau} = 3.35\mu s$). The results show that using $T_{PD} - T_x$ without compensation yields large errors (in μs) due to the fixed hardware delay, while compensation significantly reduces them, bringing values closer to the ground truth. Note that the double window often underestimates packet start time (having negative $T_{PD} - T_{GT}$), so subtracting $\bar{\tau}$ can increase its error. As a result, the single window may appear more accurate post-compensation, despite the double window being inherently more precise. This highlights the need for precise calibration, and potential correction of the systematic underestimation in the double window method may further improve the accuracy.

To better understand the practical implications of the packet detection timing errors, we translated them into the equivalent localisation errors, assuming propagation time measurement-based methods. The error in the estimation of time-of-flight is translated to distance by

$$\Delta d = c \cdot \Delta t \tag{9}$$

, where Δd is the distance error, Δt is the timing error, and $c \approx 3 \times 10^8 m/s$ is the speed of light. For TOA systems, this error directly affects the position estimate at each receiver. For instance, the single window with window 10 has typical packet detection time errors around 300-500 ns, which corresponds to a localisation error of 90-150 meters, while the double sliding window has time differences around 150-200

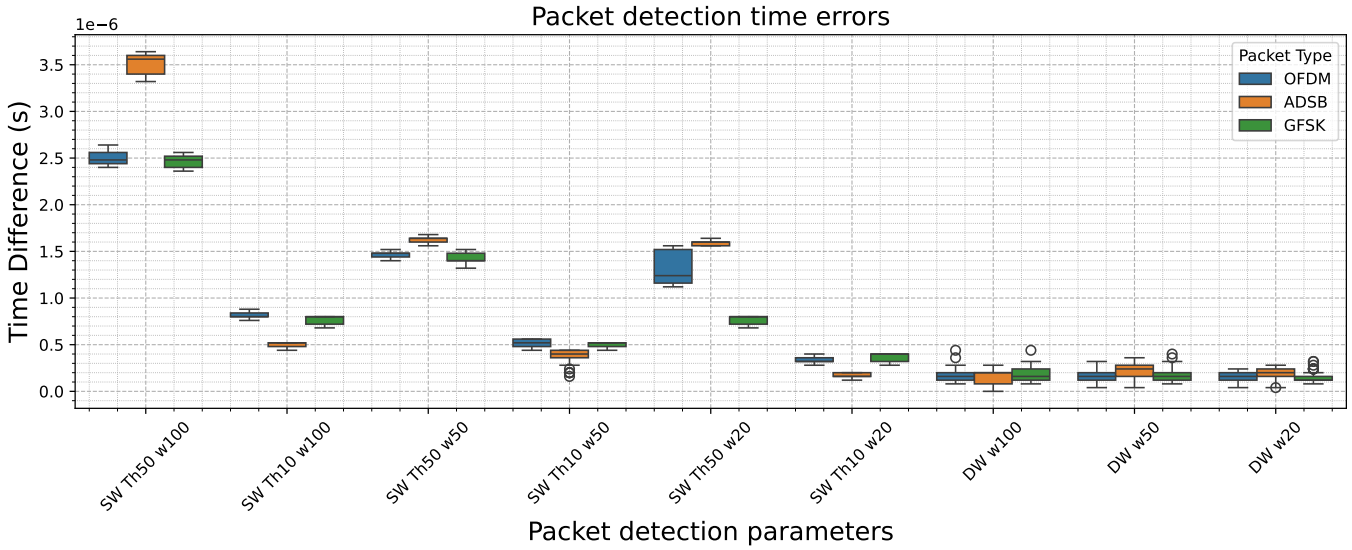

Fig. 8. Comparison of packet detection time errors. 'SW' stands for Single Window and 'DW' for Double Window, the number following 'Th' is the threshold percentage, and the number following 'w' represents the window size

TABLE I
COMPARISON OF TIMING ERRORS. 'O', 'A', AND 'G' DENOTE OFDM, ADS-B, AND GFSK SIGNALS, RESPECTIVELY.

Packet detector	$T_{PD} - T_{GT}$	$T_{PD} - T_x$	$T_{PD} - T_{GT} - \tau$
SW TH50 W50	O: $1.465\mu s$ A:$1.626\mu s$ G:$1.432\mu s$	O: $4.726\mu s$ A:$4.919\mu s$ G:$4.605\mu s$	O: $1.376\mu s$ A: $1.569\mu s$ G: $1.255\mu s$
SW TH10 W50	O: $515.2ns$ A:$387.4ns$ G: $502.5ns$	O: $3.776\mu s$ A: $3.68\mu s$ G: $3.676\mu s$	O: $426.2ns$ A: $330.35ns$ G: $325.8ns$
SW TH10 W20	O: $336.2ns$ A: $162.8ns$ G: $364.2ns$	O: $3.597\mu s$ A: $3.456\mu s$ G: $3.538\mu s$	O: $247.2ns$ A: $105.79ns$ G: $187.5ns$
DW W100	O: $-171ns$ A: $-158.6ns$ G: $-178.6ns$	O: $3.09\mu s$ A: $3.135\mu s$ G: $2.99\mu s$	O: $-260ns$ A:$-215.48ns$ G: $-355.2ns$
DW W50	O: $-156.9ns$ A:$-226.2ns$ G: $-171.9ns$	O: $3.104\mu s$ A:$3.067\mu s$ G: $3.001\mu s$	O: $-245.86ns$ A:$-283.1ns$ G: $-348.6ns$
DW W20	O: $-146.5ns$ A:$-189.7ns$ G: $-146.3ns$	O: $3.114\mu s$ A:$3.103\mu s$ G: $3.027\mu s$	O: $-235.5ns$ A: $-246.6ns$ G: $-323ns$

ns, corresponding to 45-60 meters. These results highlight that even sub-microsecond timing inaccuracies can significantly degrade localisation performance in non-terrestrial systems, especially when precise positioning is required. In TDOA systems, since the position estimate relies on time differences between receivers, some components of the packet detection timing error, particularly systematic or correlated ones, may partially cancel out. As a result, the impact of detection timing inaccuracies on localisation accuracy is often lower compared to TOA-based approaches. However, uncorrelated or jitter-induced errors can still contribute to localisation uncertainty.

B. Computational complexity

To evaluate the computational complexity of the detection algorithms, we measured the execution time required to process one of the collected datasets for each sliding window algorithm. The dataset consists of 750 million samples, corresponding to 30 seconds of transmission sampled at 25 MSPS. Both the single and double sliding window algorithms were evaluated using the same window sizes to ensure a fair comparison. The execution time was measured using the cProfile tool, and each test was repeated four times. The average values are presented in Table II.

All computations were performed offline on a general-purpose server to isolate the algorithmic cost. The absolute execution times are not directly relevant in this context, as they depend on various parameters, such as the hardware's computational capabilities, programming language, or dataset size. However, the relative comparison provides meaningful insight. Results show that the double window algorithm requires approximately twice the computation time of the single window algorithm, highlighting the key trade-off between detection accuracy and processing cost.

Although the exact computational capabilities of the USRP E320 are not characterised in this study and are left for future work, the observed ratios provide a scalable indication of relative complexity. This information is directly applicable to embedded or FPGA-based implementations, where processing resources and power budget are limited. In such scenarios, the proportional increase in computational demand observed for the double sliding window algorithm would likely translate into a similar rise in power consumption and hardware resource usage.

The impact of different window sizes on computational

TABLE II
COMPUTATION TIME

Packet detection algorithm	Running time avg (s)
Single window	402.25
Double window	812
Ratio DW/SW	≈ 2

time was not evaluated, as both algorithms are optimised with sliding windows. In this approach, only one new sample is added and one old sample is removed at each iteration, avoiding recalculation over the whole window. Therefore, in a streaming scenario, the number of operations per iteration remains constant regardless of the window sizes.

V. CONCLUSIONS AND FUTURE WORK

This work presents a detailed evaluation of two energy-based packet detection algorithms - the single sliding window and double sliding window - focusing on the timestamping accuracy for time-based applications, such as localisation, in signal-agnostic spectrum sensing systems.

The results demonstrate that while the single sliding window approach is more computationally simpler and faster, its detection time accuracy is strongly influenced by the choice of threshold and window size. In contrast, the double sliding window offers improved detection time accuracy and is more robust to noise and parameter variations, but at the cost of increased computational complexity.

We validated both methods using real captured signals on the USRP E320, characterised internal delays, and proposed a timestamping compensation for hardware constraints. Experiments across multiple signal types (OFDM, ADS-B, and GFSK packets) confirmed the algorithm's protocol-agnostic performance of both algorithms and quantified the trade-offs between detection and computational load.

As future work, we plan to explore adaptive thresholding techniques, including machine learning-based approaches to enhance detection robustness under varying conditions. We also aim to evaluate the performance of the algorithms in more realistic wireless environments, particularly under low-SNR, multipath, and mobile scenarios. Additionally, extending the framework to multi-device setups with unsynchronised clocks will allow us to assess synchronisation robustness in distributed systems. Finally, implementing the detection algorithms directly on the FPGA will be considered to reduce latency and improve scalability for real-time scenarios, while also analysing the energy consumption and real-time computational performance of the algorithms on the USRP E320 probe to assess their feasibility for long-duration balloon-based missions.

REFERENCES

[1] B. E. Y. Belmekki and M.-S. Alouini, "Unleashing the potential of networked tethered flying platforms: Prospects, challenges, and applications," *IEEE Open Journal of Vehicular Technology*, vol. 3, pp. 278–320, 2022. DOI: 10.1109/OJVT.2022.3177946.

[2] M. Schäfer, Y. Lizarribar, G. Bovet, and D. Verbruggen, "Let's take this upstairs: Localizing ground transmitters with high-altitude balloons," in *MILCOM 2024 - 2024 IEEE Military Communications Conference (MILCOM)*, 2024, pp. 475–480. DOI: 10.1109/MILCOM61039.2024.10773949.

[3] B. Reynders, F. Minucci, E. Perenda, *et al.*, "Skysense: Terrestrial and aerial spectrum use analysed using lightweight sensing technology with weather balloons," in *Proceedings of the 18th International Conference on Mobile Systems, Applications, and Services*, ser. MobiSys '20, Toronto, Ontario, Canada: Association for Computing Machinery, 2020, pp. 352–363, ISBN: 9781450379540. DOI: 10.1145/3386901.3389026. [Online]. Available: https://doi.org/10.1145/3386901.3389026.

[4] Z. Wei, L. Wang, Z. Gao, *et al.*, "Spectrum sharing between high altitude platform network and terrestrial network: Modeling and performance analysis," *IEEE Transactions on Communications*, vol. 71, no. 6, pp. 3736–3751, 2023. DOI: 10.1109/TCOMM.2023.3262305.

[5] T. Yucek and H. Arslan, "A survey of spectrum sensing algorithms for cognitive radio applications," *IEEE Communications Surveys & Tutorials*, vol. 11, no. 1, pp. 116–130, 2009. DOI: 10.1109/SURV.2009.090109.

[6] Y. Huang, L. Yuan, and W. Gong, "Research on ieee 802.11 ofdm packet detection algorithms for household wireless sensor communication," *Applied Sciences*, vol. 12, no. 14, 2022, ISSN: 2076-3417. DOI: 10.3390/app12147232. [Online]. Available: https://www.mdpi.com/2076-3417/12/14/7232.

[7] J. Shen, A. F. Molisch, and J. Salmi, "Accurate passive location estimation using toa measurements," *IEEE Transactions on Wireless Communications*, vol. 11, no. 6, pp. 2182–2192, 2012. DOI: 10.1109/TWC.2012.040412.110697.

[8] F. Ricciato, S. Sciancalepore, F. Gringoli, N. Facchi, and G. Boggia, "Position and velocity estimation of a non-cooperative source from asynchronous packet arrival time measurements," *IEEE Transactions on Mobile Computing*, vol. 17, no. 9, pp. 2166–2179, 2018. DOI: 10.1109/TMC.2018.2792443.

[9] M. Maróti, B. Kusy, G. Simon, and Á. Lédeczi, "The flooding time synchronization protocol," in *Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems*, ser. SenSys '04, Baltimore, MD, USA: Association for Computing Machinery, 2004, pp. 39–49, ISBN: 1581138792. DOI: 10.1145/

1031495.1031501. [Online]. Available: https://doi.org/10.1145/1031495.1031501.

[10] I. Daramouskas, I. Perikos, M. Paraskevas, V. Lappas, and V. Kapoulas, "Performance analysis for time difference of arrival localization in long-range networks," *Smart Cities*, vol. 7, no. 5, pp. 2514–2541, 2024, ISSN: 2624-6511. DOI: 10.3390/smartcities7050098. [Online]. Available: https://www.mdpi.com/2624-6511/7/5/98.

[11] B. Xu, G. Sun, R. Yu, and Z. Yang, "High-accuracy tdoa-based localization without time synchronization," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 8, pp. 1567–1576, 2013. DOI: 10.1109/TPDS.2012.248.

[12] A.-C. Orgerie, P. Gonçalves, M. Imbert, J. Ridoux, and D. Veitch, "Survey of network metrology platforms," in *2012 IEEE/IPSJ 12th International Symposium on Applications and the Internet*, 2012, pp. 220–225. DOI: 10.1109/SAINT.2012.41.

[13] R. Exel, "Clock synchronization in ieee 802.11 wireless lans using physical layer timestamps," in *2012 IEEE International Symposium on Precision Clock Synchronization for Measurement, Control and Communication Proceedings*, 2012, pp. 1–6. DOI: 10.1109/ISPCS.2012.6336622.