אוניברסיטת בן-גוריון בנגב
Ben-Gurion University of the Negev
הפקולטה למדעי ההנדסה
המחלקה להנדסת חשמל ומחשבים
Faculty of Engineering Science
Dept. of Electrical and Computer Engineering

Fourth Year Engineering Project
Final Report

Cyber Security for next generation LEO Satellite Network Using Phased Array Antenna

Project Number:	**p-2023-009**
Student #1:	Name: Hoffman Yaara
	ID: 206201022
Student #2:	Name: Itzhak Ofir
	ID: 205796238
Supervisors:	Prof. Arnon Shlomi
	Kumar Rajnish
Submitting Date:	29.07.2023

Table of Contents

1. Abstract

1.1. English Abstract

Cyber Security for next generation LEO Satellite Network

Students Names: Hoffman Yaara, Itzhak Ofir
yaarahof@post.bgu.ac.il
Adviser name: Prof. Arnon Shlomi

Today, Low-Earth-Orbit satellite constellations can allow fast and reliable global internet.

However, they are highly vulnerable to physical Denial of Service cyber-attacks, which we aim to defend against.

We propose an innovative approach, using a Phased Array Antenna in Sub-THz frequencies. Phased arrays can cancel interferences by beamforming the antenna (Adaptive Nulling). Sub-THz frequencies allow higher data rates and narrower beams.

We simulated our defensive algorithm using MATLAB and achieved excellent results.

Yet in reality, conditions aren't perfect. Sub-THz beamwidth is narrower, therefore beamforming angular noise ($\Delta\theta°>0$) may impair nullification. Our examination showed a sharp decrease in SNR for every deviation. For $\Delta\theta=0.1°$, our SNR was lower by -1.08 dB from the maximum 10dB.

Our conclusion holds that Phased Array Antennas offer an excellent defense against satellite cyber-attacks. With that being said, more testing must be done in a real environment. We predict that under the described angular deviations our defensive proposal will greatly contribute to the satellite's defense.

Keywords: LEO satellite, Global Internet, DoS physical Cyber Attack, Jamming, Phased Array Antenna, Sub-THz, Beamforming, Beamwidth, Adaptive Nulling, Directional deviation, SNR.

1.2. Hebrew Abstract

אבטחת סייבר לדור הבא של רשתות לווייני נמוכי מסלול בעזרת אנטנת מערך מופע

שמות הסטודנטים: הופמן יערה, יצחק אופיר
yaarahof@post.bgu.ac.il
שם המנחה: פרופ' ארנון שלומי

כיום לוויינים נמוכי מסלול יכולים לאפשר תקשורת אינטרנטית גלובלית מהירה ואמינה.

אולם, לוויינים אלו פגיעים ביותר להתקפות פיזיות מסוג מניעת שירות (Denial of Service - DoS), נגדן אנו מגנים.

אנו מציעים גישה חדשנית על ידי שימוש באנטנת מערך מופע (Phased Array) בתדרי Sub-THz. אנטנת מערך מופע מאפשרת לנטרל הפרעות מכיוונים ספציפיים, בתהליך שנקרא Adaptive Nulling. תדרי Sub-THz מאפשרים תעבורת מידע גדולה ואלומת אות צרה יותר.

את האלגוריתם ההגנתי שלנו בדקנו בתוכנת MATLAB, שהציג תוצאות מצוינות.

אולם, במציאות התנאים אינם מושלמים. בתדרי Sub-THz אלומות האותות צרות, לכן כל רעש זוויתי בגודל $\Delta\theta°>0$ יפגע בנטרול ההפרעה. הפרויקט שלנו העלה כי ישנה ירידה חדה בערכי ה-SNR (יחס אות לרעש) לכל סטייה זוויתית של הנטרול מכיוון ההפרעה. עבור $\Delta\theta°=0.1°$ יחס האות לרעש ירד ב: 1.08 dB- מערכו המקסימלי של 10 dB.

לסיכום, אנו מסיקים שאנטנות מערך מופע מהוות פתרון מצוין להגנה נגד מתקפות סייבר לווייניות. עם זאת, בדיקות נוספות נדרשות להיעשות בסביבת אמת, כדי לוודא מסקנות אלה. אנו צופים, כי תחת הרעשים הזוויתיים שתוארו קודם, המערכת ההגנתית שלנו תתרום רבות ותספק מעטפת הגנתית יעילה ללווייניים.

מילות מפתח: לווייניים נמוכי מסלול, תקשורת אינטרנטית, התקפת מניעת שירות פיזית, שיבוש אות, אנטנת מערך מופע, תדרי Sub-THz, רוחב אלומה, נטרול זוויתי, רעש זוויתי, יחס אות לרעש.

2. Introduction

2.1. Motivation

Low Earth Orbit (LEO) satellites that orbit relatively close to Earth are increasingly popular as an option for providing worldwide internet access. LEO satellites constellations allow faster and more reliable internet connectivity with lower latency compared to terrestrial connections and allows worldwide access even in remote areas or warzones.
Starlink by SpaceX is an operator that provides such internet access, enabling internet connection for the people and military of Ukraine while it defends itself from the Russian invasion [1]. However, this link is now endangered with Russia testing weapons to target Starlink's operation [2].
LEO constellations are highly vulnerable to physical Denial of Service cyber-attacks. As their use is a new technology, ongoing research is conducted examining their defensive techniques, which is the motivation for our project. A satellite under a cyber-attack is even more vulnerable than a normal internet connection as it is far out of reach from any operator, and in harsh space conditions. We suggest using Phased Array Antennas on satellites to defend against such attacks, in sub-THz frequencies.

2.2. Project Goal

Our goal is to derive a defensive algorithm for satellites that will harness the Phased Array Antenna capability of enemy signal nullification and test both its performance and limitations. This method in which we nullify enemy signals is called "Adaptive-Nulling". It allows us to beamform our reception in such a way that we can cancel signals from specific directions [3]. This is a powerful attribute which we have seen huge potential in, and know it is already being used in military communication. This attribute will be examined alongside transmission in sub-THz frequencies, which can improve data rate in the future 6G era [4].

2.3. Previous Work

Previous works [5] tried solving jamming signals by randomizing the network's routing algorithms. This is a good solution to avoid a single line being attacked, but in a multiple attacks scenario its efficiency is decreasing. In addition, total protection is not always guaranteed but rather randomized.

Another approach we found was frequency diversity. This method changes the transmission frequency when attacked. By doing so, the jammer's signal cannot interrupt the satellite anymore. This method results in constant frequency changes, which requires synchronization between many ISL's (inter satellite links) and constant spectrum management. Both make communication very complex and therefore rule out this method [6].

2.4. Support for Our approach

Our approach was chosen since it is based on a long-proven use of Phased Array Antennas in military communication, and it offers a real-time, mostly guaranteed answer to any jamming attack. Moreover, phased array antennas can be steered without any need for mechanical movement. This is very beneficial to the space environment, where satellites operate. Mechanical issues are very hard to deal with remotely in space and therefore are further evaded. We must note that these antennas require an extra cost and size allocation upon the satellite, but we believe the benefits overcome these drawbacks. In the future fork section, we will further discuss our options to minimize space allocation using MLC (Microwave Liquid Crystal) phase shifters. Another limitation is that the number of elements in the array will dictate the number of constraints (hence amplification or nullification) the satellite can apply. We discuss this issue ahead, but arrays usually include a large number of elements which are hard to overcome. In addition, sub-THz frequencies require high precision, which may also be solved by MLC and is also discussed ahead. Overall, we believe our approach delivers an excellent solution to the problem.

3. Solution and Engineering Design

Our chosen solution to the problem is using Phased Array Antennas on satellites to defend against Denial-of-Service attacks in sub-THz frequencies. This solution allows us to nullify and cancel enemy interferences while maintaining high data rates with friendly connections.
We assume the direction and power of an attack are provided via a detection system.
The Satellite’s main program sends this data into our defensive algorithm, which in turn derives the required steering vector. The satellite’s main program sends both friendly and enemy directions, which are amplified or nullified respectively. The steering vector is later received by the phased array controller, which converts it to actual phase shifts applied in each element. These phase shifts beamform our antenna as required. The same attacking signal is instantly received again in the antenna, but its power is reduced to up to complete cancellation. The antenna is constantly beamformed to the optimal satellite’s needs in real-time.

3.1. System Block Diagram

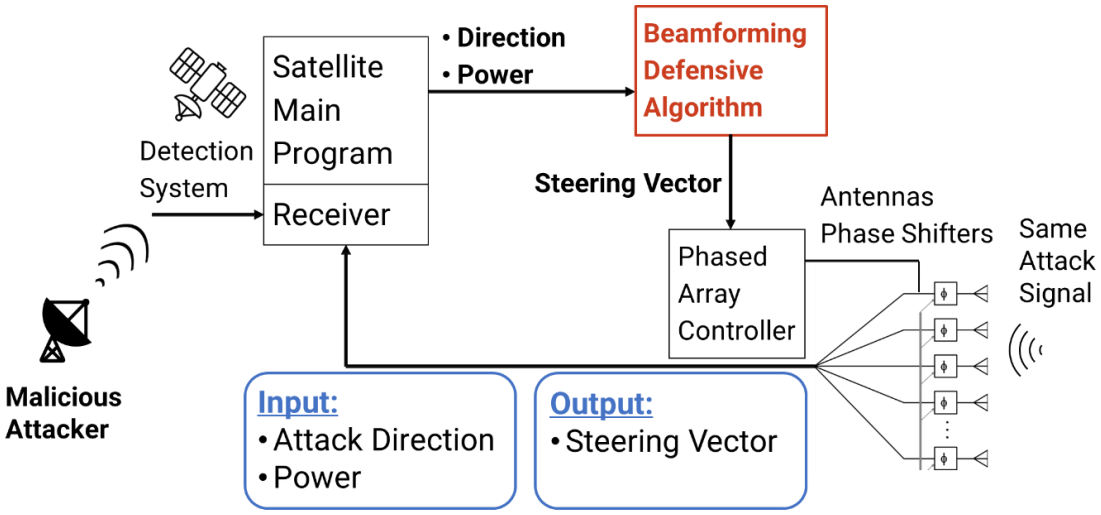

Figure 1. Block diagram of the SUB-THz phased-array defense system

3.1.1. Detection System (DS): A Detection System will provide the direction and power of an ongoing attack in real-time.

3.1.2. Satellite's Main Program: The main program is run by the satellite operating system.

3.1.3. Receiver: Receives the signal caught by the array's antennas.

3.1.4. Beamforming Defensive Algorithm: Our defensive algorithm. Receives direction and power of attack as input, and outputs the steering vector.

3.1.5. Steering Vector: This vector is sent to the Phased Array Antenna Controller and sets the appropriate phase shifts that beamform the antenna. This vector holds all the information the controller needs to conduct amplification of friendly signals, and nullification of enemy signals.

3.1.6. Phased Array Controller: Applies the required phase shifts to the elements of the array.

3.1.7. Antennas Phase Shifter: Achieve required phase shift in each of the array's elements, to complete the required beamforming.

3.2. Design Considerations

Our system design is based on existing phased array antennas. We chose element spacing to be λ_min/2, to ensure that grating lobes do not appear when the beam is scanned to ±60° of any cut [7]. This structure creates a known array factor which usually amplifies the middle bearing of 0° [8]. The number of elements does also affect the array factor. We tried several options for N (number of elements) in our ULA (uniform linear array).

The following graphs (fig. 2) show the normalized power received from every direction in a specific array with N elements:

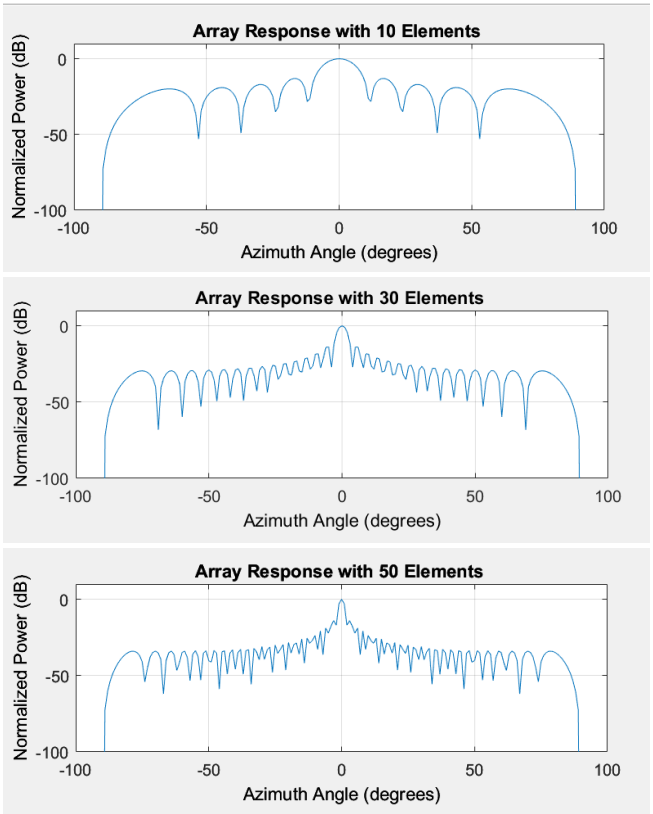

Figure 2. Received power (normalized) of a simple array without beamforming.

We can clearly see that increasing N changes the array response to be more direct.
As mentioned before, to achieve adaptive nulling, we are required to force constraints upon our array. The constraints must be at a smaller count than the total element count in our array, N. This is an important limitation, as an enemy count greater than the number of our elements will prevail in our defense.
Beside this limitation, we must consider that the more elements we use:

✓ We can force more constraints.
✓ Better precision.
☒ Higher costs.
☒ Size increases.

Therefore, we chose a 30 elements array as a reasonable and effective example for our purpose. It allows us to deal with up to 30 constraints of friendly or enemy directions, which makes it a fair tradeoff between cost, efficiency, and area on the satellite.

Area Calculation is: $(\frac{\lambda}{2} \cdot (N-1)) = (\frac{c}{2f} \cdot N) = (\frac{3 \cdot 10^8}{2 \cdot 100 \cdot 10^9} \cdot 29) = 4.35_{cm}$

We have chosen an innovative use of transmission in this frequency in sub-THz band which is 90 - 300 GHz. This use allows a higher data rate [9] but creates a problem of precision which we will discuss ahead.

In our simulations we chose an Enemy/Friendly power ratio of 5, defined as:

$$ratio = \frac{P_{enemy}}{P_{friendly}} = 5$$

Hence the attacker is transmitting at 5 times the power of a friendly transmission.

4. Experiments and Final Tests

To evaluate our defensive system, we've tested several scenarios to challenge it using MATLAB. For the simple scenario of one attacker signaling an interference in direction 40° and one friendly signaling in direction 0° we've received the following results:

4.1. Testing without phase deviation

4.1.1. Friendly signals from [0°] and Enemy signals from [40°]:

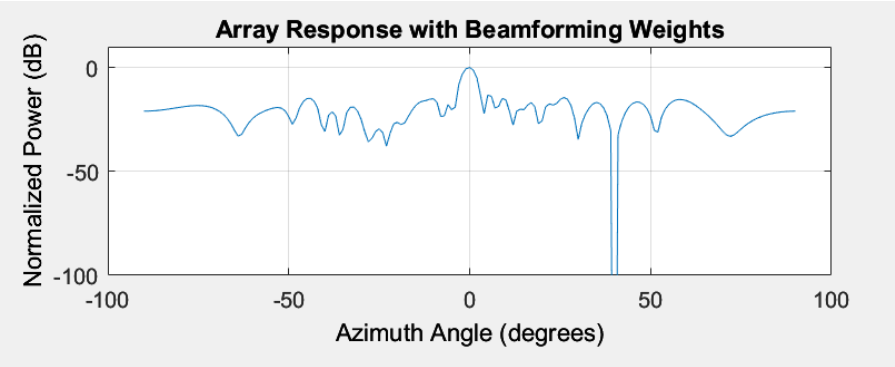

Figure 3. First example array response with amplification at 0° and nulling at 40°.

We can clearly see the nulling response at 40°. Any signal arriving at our antenna from this direction is completely canceled. On the other hand, any signal arriving from our friendly direction of 0° is received completely unharmed. Any other direction wasn't constrained and therefore is a degree of freedom.

We test our performance using SNR (Signal to Noise Ratio), compared to the noised channel since loss space and noise cannot be avoided. In the discussed case:

$SNR_{Noised\ Channel} = 9.97\ dB$

$SNR_{With\ Nulling} = 9.97\ dB$

$SNR_{Without\ Nulling} = -11.23\ dB$

$$\Delta SNR_{With\ Nulling} = SNR_{With\ Nulling} - SNR_{Noised\ Channel} = 0\ dB$$
$$\Delta SNR_{Without\ Nulling} = SNR_{Without\ Nulling} - SNR_{Noised\ Channel} = -21.2\ dB$$

This is an outstanding result. We succeeded in maintaining the maximum SNR possible with $\Delta SNR = 0$ even when an attack occurred. This means our antenna had successfully canceled all the attacker's interference. We can see the nulling notch at exactly 40°, causing the any received signal from this specific direction to be completely ignored.
Same notch can be seen also in the azimuth cut:

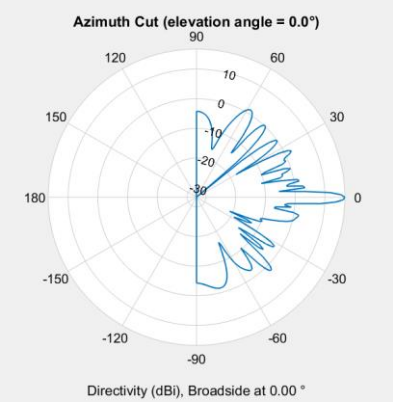

Figure 4. First example azimuth cut with amplification at 0° and nulling at 40°.

The following figure shows the transmitted **original signal** alongside:

- The noised channel signal.
- The received signal **without** nulling.
- The received signal **with** nulling.

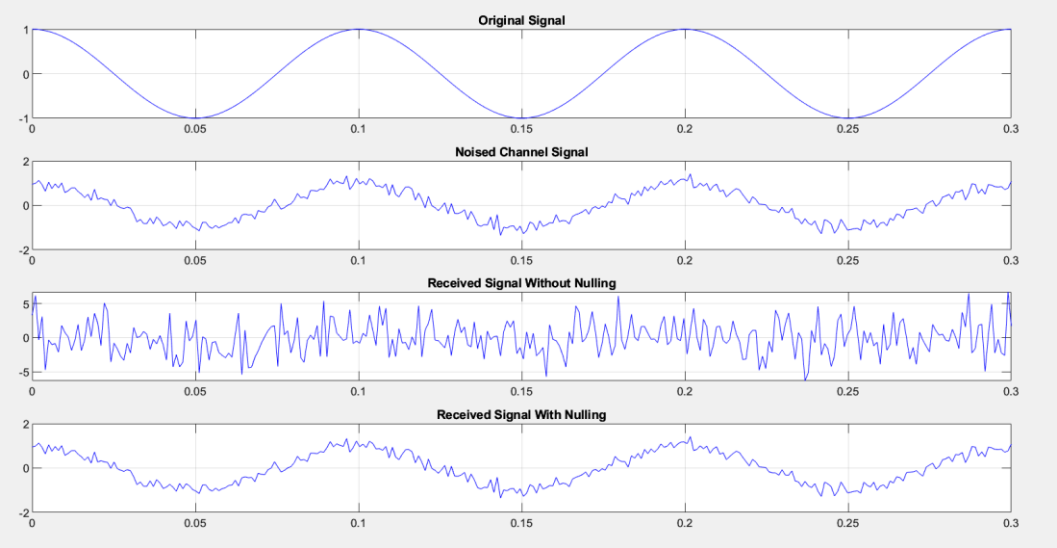

Figure 5. Signal with and without nulling. The nulling process clears the signal.

We can see the pure cosine signal transmitted, and the noised signal passed over the channel. The received signal without nulling is completely jammed, supporting our previous result of $\Delta SNR_{Without\ Nulling} = -21.2\ dB$ for this case. On the contrary, by using the adaptive nulling defensive algorithm we can see the received signal is identical to the noised channel one. a difference of $\Delta SNR_{With\ Nulling} = 0\ dB$ supports this view.
A similar attack with power ratio = 500 was also successfully canceled.

4.1.2. Friendly signals from [0°] and Enemy signals from [40°,60°,75°]:

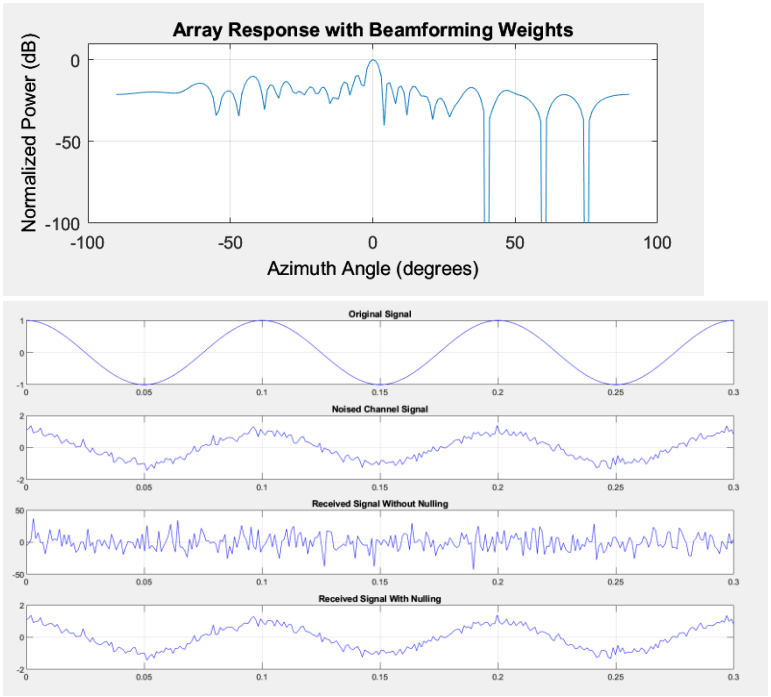

Figure 6. Results with multiple attackers and precise nulling.

We can again see the adaptive nulling notches at all enemy directions [40°,60°,75°].
The received signal is again clear of any interferences and includes solely the unavoidable channel noise. In this scenario there were 3 active attacker and therefore that signal is completely jammed with $\Delta SNR_{Without\ Nulling} = -35.05\ dB$
Which completely disrupts the received signal.

$$SNR_{Noised\ Channel} = 10.02\ dB$$
$$SNR_{With\ Nulling} = 10.02\ dB$$
$$SNR_{Without\ Nulling} = -25.03\ dB$$

$$\Delta SNR_{With\ Nulling} = SNR_{With\ Nulling} - SNR_{Noised\ Channel} = 0\ dB$$
$$\Delta SNR_{Without\ Nulling} = SNR_{Without\ Nulling} - SNR_{Noised\ Channel} = -35.05\ dB$$

4.1.3. Friendly signals from [-50°, -20°,0°] and enemy signals from [40°,60°,75°]:

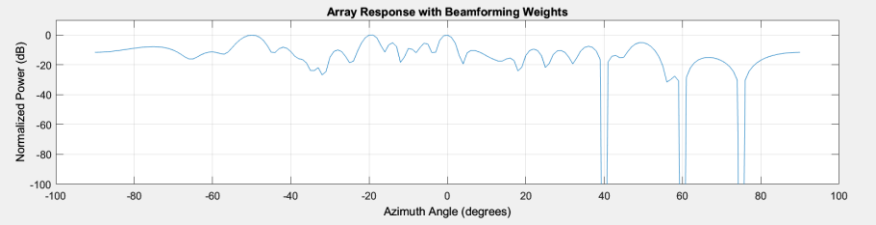

Figure 7. Results with multiple nulls and multiple amplifications.

The major differences in this scenario are the added friendly constraints in directions -50° and -20°. We can see the added peaks at the array response at the corresponding directions. This means that any signal arriving to the antenna from these directions will be received unharmed.

$$SNR_{Noised\ Channel} = 9.61\ dB$$
$$SNR_{With\ Nulling} = 9.61\ dB$$
$$SNR_{Without\ Nulling} = -24.52\ dB$$

$$\Delta SNR_{With\ Nulling} = SNR_{With\ Nulling} - SNR_{Noised\ Channel} = 0\ dB$$
$$\Delta SNR_{Without\ Nulling} = SNR_{Without\ Nulling} - SNR_{Noised\ Channel} = -34.13\ dB$$

We wanted to test a more complex scenario where constraints of amplification and nullification are much more adjacent.

4.1.4. Friendly signals from [0°,15°] and enemy signals from [-10°, 10°, 40°]:

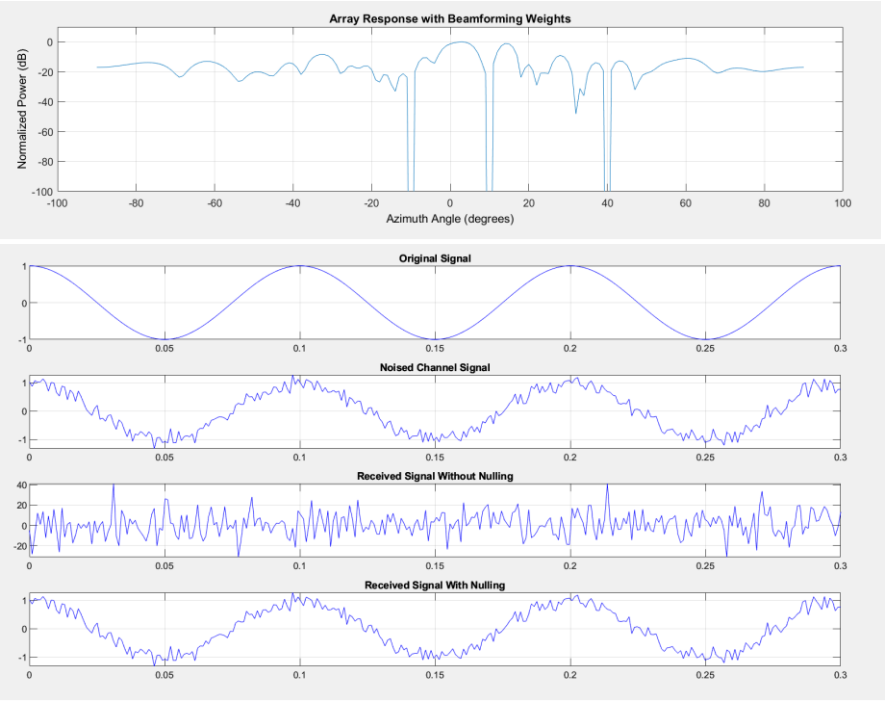

Figure 8. Signal with and without nulling, with multiple attackers and multiple amplified.

Even in such scenario, where a friendly signal from direction 0° was surrounded by attacks from -10° and 10°, the defensive system accomplished its purpose successfully.

$$SNR_{Noised\ Channel} = 10\ dB$$
$$SNR_{With\ Nulling} = 10\ dB$$
$$SNR_{Without\ Nulling} = -24.47\ dB$$

$$\Delta SNR_{With\ Nulling} = SNR_{With\ Nulling} - SNR_{Noised\ Channel} = 0\ dB$$
$$\Delta SNR_{Without\ Nulling} = SNR_{Without\ Nulling} - SNR_{Noised\ Channel} = -34.47\ dB$$

4.1.5. Friendly signals [0°,5°,15°], enemy signals [-70°, -50°, -20°, 60°, 75°]:

We pushed our system to an even more complex scenario holding a total of 9 constraints:

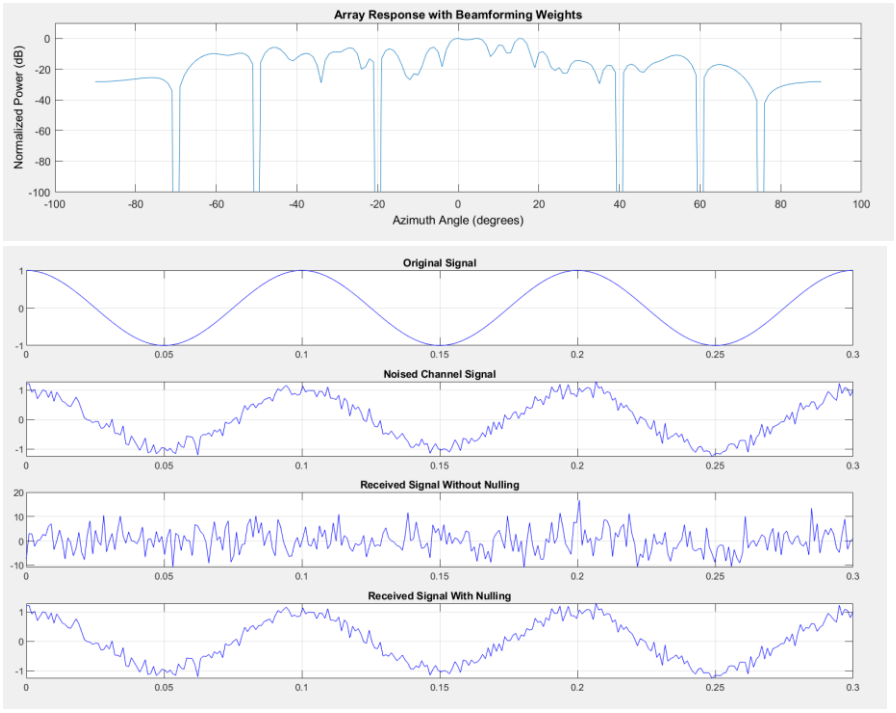

Figure 9. Results with many friendly and malicious signals, and precise nulling.

$$SNR_{Noised\ Channel} = 10.10\ dB$$
$$SNR_{With\ Nulling} = 10.10\ dB$$
$$SNR_{Without\ Nulling} = -16.52\ dB$$

$$\Delta SNR_{With\ Nulling} = SNR_{With\ Nulling} - SNR_{Noised\ Channel} = 0\ dB$$
$$\Delta SNR_{Without\ Nulling} = SNR_{Without\ Nulling} - SNR_{Noised\ Channel} = -26.62\ dB$$

This case was again excellently delt with.

4.2. Adaptive Nulling Error due to sub-THz beamwidth:

So far, our defensive system had shown excellent and lossless results. Yet in real-life we expect directional noise to occur, especially when we use sub-THz frequencies as we suggest which inflict much narrower beamwidth and need for precision.

We will examine the case in which our attacker is deviated $\Delta\theta°$ from the actual nulling direction. We will then analyze the SNR achieved in each case. All the experiments will be made upon the following case of friendly signal from [0°] and nullification at [40°,70°], with a power ratio of 5.
In the following figure the attackers were at [42°,70°], therefore we acquire $\Delta\theta=2°$. the results are in fig. 10.

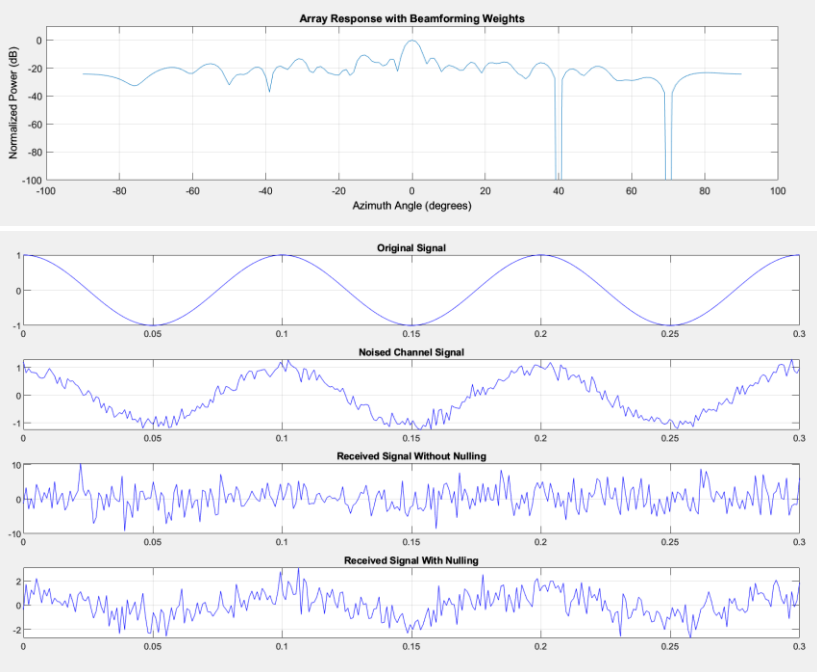

Figure 10. Results with many friendly and malicious signals, and adaptive nulling.

We can see the remnant of the cosine wave in the received signal with nulling. Even so, the nulling was applied to a direction with deviation of $\Delta\theta=2°$. This deviation caused the received signal to be very noisy, and greatly impaired.

We scanned the SNR of the received signal as function of the directional deviation Δθ°, in jumps of 0.1°. The attacker direction changed while nulling remained fixed at 70°. The results are very interesting.
We present the results is a chart in appendix B, and here is a visual representation:

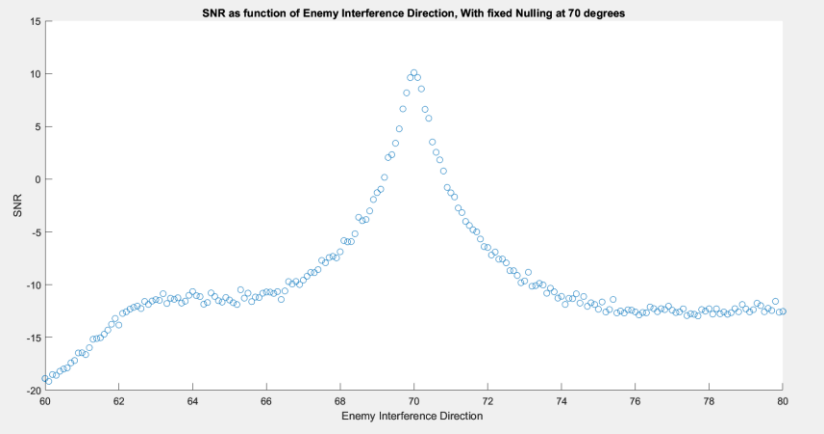

Figure 11. SNR as a function of interference direction, nulling fixed at 70°, growing deviation.

To explain why the graph is asymmetric, we must look at the array response from Figure 10:
Zooming in between 60° to 80° we can see a risen amplification towards 60°, compared to the right-most side. This explains the sharper SNR drop in the left-most side of the graph.

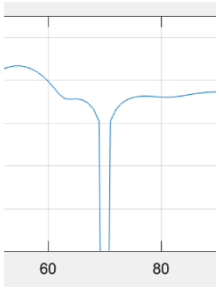

Figure 12. Zoom in array response between 60° to 80°.

Clearly, the most effective deviation is the trivial case of Δθ=0°. Interesting to see it the exponential drop for every Δθ°, positive or negative in an already logarithmic scale. This points to a high need of precision in our system, but according to our findings there is a tolerable deviation of Δθ=0.2° which will cause a loss of up to ~1.939 dB.

Attack Direction	Deviation (from 70°)	SNR
69.8°	0.2°	8.58790035979558
69.9°	0.1°	9.34137043995761
70°	0°	10.1365372464506
70.1°	0.1°	9.13811740049446
70.2°	0.2°	8.19763181831083

In conclusion, at a deviation of Δθ=0.2° will keep a fair SNR of about 8 dB.

5. Problems and Solutions

5.1. Planning Phase

- At first, finding the exact subject took quite some time. Extensive research was made to find the right problem needed solving which fits the project's scale. The solution was investing a lot of time.
- After we found the problem of solving DoS attacks in LSNs, we had to decide how to try and mitigate it. Among many options, we decided to focus on the antenna's defense mechanism.

5.2. Execution Phase

- We decided to use MATLAB to execute our idea, as it has a phased array library. However, the built-in library functions did not fit the theoretical knowledge we acquired. We tried two methods, one was to write methods from scratch to fit our knowledge. In the end, after delving into the theory behind MATLAB's functions and learning them, we decided that it was better to use the built-in methods.

5.3. Production Phase

- The MATLAB tests we wanted to run are scarcely used, therefore there was minimal documentation and running the tests took a learning curve. In the end we learned this part as well and got the information described in this document.
- Our results deteriorate when there is a large angle of deviation between the real malicious signal and our nulling direction, therefor real-world usage requires very accurate antennas. A possible solution for this hardware demand is the usage of liquid crystal steering for our antennas, which is discussed in the next chapter.

6. Future Work: Microwave Liquid Crystal Phase Shifters

Besides testing in real-environment communication using real satellites, another important future work may be conducted. As stated before, a strong precision demand on the hardware is required. One possibility to achieve it may be Microwave Liquid Crystal Phase Shifters.

MLC phase shifters are stacks of thin integrated liquid crystal (LC) varactors that can be used to steer antenna signals in the sub-THz band [10]. There are several ways in which MLCs can be beneficial to fulfilling our project in real-world situations:

6.1. Precise Control of Phase Shift: MLC technology applies an electric field to the LC material, which changes the orientation of the LC molecules and, in turn, changes the refractive index of the material.

By controlling the refractive index of the LC material, the phase of the electromagnetic waves passing through the material can be precisely controlled.

This can fix the issue of angle deviation in nullification discussed above.

6.2. Low Power Consumption: MLC-based antennas have a low power consumption, which is crucial in space application since the satellite's power source is limited.

For example, for a 16x16 array with 4 beams the power consumption of MLC shifter is around 19W compared to 65W in a silicon-based shifters [10].

6.3. Small Size: MLC technology enables the manufacturing of compact and flat tunable RF components, which allows for the development of low-cost, robust, and reliable mm-wave hardware solutions in a smaller form factor. This is another important factor to consider in spacecraft antennas.

6.4. Low Frequency Dispersion: This is not unique to MLCs, but a very important characteristic to enable the usage of sub-THz frequencies. MLC components exhibit low frequency dispersion, which means that their electrical properties remain stable over a wide range of frequencies. This important characteristic ensures that the component's performance is consistent across a wide range of frequencies, including the sub-THz range.

7. Conclusions

7.1. With Phased Array Nulling we can evade physical DoS attacks on LEO satellites in the sub-THz band, even with shift inaccuracies. Physical real-world experiments should be done to test our findings, yet we predict good results with high DoS attack resilience.

7.2. Our solution shows good results as an add-on defensive algorithm for future satellites. Moreover, it allows reliable connections that can be steered electrically, an important feat for extra-celestial devices.

7.3. Deviation must be under 0.2° to avoid low SNR. We suggest future experiments with liquid crystal phase shifters to maintain low error.

7.4. In addition, the most common ISLs currently utilized rely on visible-light optical links. This posed light pollution which impacts humans, animals, and astronomers defending Earth from potential asteroid damage. By adopting our solution of sub-THz frequency phased-array-based ISLs, we can mitigate the negative effects and ensure a more sustainable and protected environment.

8. Progress

Due	Task	Status
01/09/2022	Define project objective	Done
23/10/2022	Pre-reading and theoretical knowledge - LEO satellites, antenna basics, DoS attacks	Done
07/11/2022	PDR	Done
20/11/2022	Studying and understanding phased array physics, array factor, adaptive nulling	Done
27/11/2022	Preliminary Report	Done
24/12/2022	First MATLAB Simulation	Done
23/01/2023	Presentation of our findings and research so far with suprevisors Prof. Shlomi Arnon and PHD student Kumar Rajnish	Done
15/03/2023	Improved MATLAB program and simulation	Done
26/03/2023	Progress Report	Done
08/04/2023	Extended simulation for more complex situations	Done
22/04/2023	Testing and comparing our suggestion to existing LEO technology	Done
10/05/2023	Test feasibility of applying our method to existing LEO technology	Done
04/06/2023	Poster	Done
18/06/2023	Presentation	Done
22/06/2023	Project Convention	Done
30/07/2023	Final Report	Done

9. References

[1] K. Duffy, “SpaceX Starlink has 150,000 daily users in Ukraine 5 weeks after being activated, government official says”, INSIDER, May 2022. https://www.businessinsider.com/elon-musk-spacex-starlink-internet-ukraine-users-150000-russia-satellite-2022-5

[2] A. Horton, “Russia tests secretive weapon to target SpaceX’s Starlink in Ukraine”, *Washington Post, Apr. 2023.* https://www.washingtonpost.com/national-security/2023/04/18/discord-leaks-starlink-ukraine/

[3] SU Chengxiao, WANG Jiegui, LIU Kai, “Study of Adaptive Nulling Methods under Different Constraints”, Proceedings of the 2nd International Conference on Computer Science and Electronics Engineering, 2013.

[4] Yoann Corre, Gregory Gougeon, Jean-Baptiste Doré, Simon Bicaïs, Benoit Miscopein, et al. “Sub-THz Spectrum as Enabler for 6G Wireless Communications up to 1 Tbit/s” *6G Wireless Summit, Mar 2019*. hal-01993187

[5] S. Arnon, R. Fratty, and Y. Saar "Cyber Security for LEO Satellite Networks", *BGU 2020 projects*, P-2022-020.

[6] M. K. Hanawal, M. J. Abdel-Rahman and M. Krunz, ”Joint Adaptation of Frequency Hopping and Transmission Rate for Anti-Jamming Wireless Systems” *IEEE Transactions on Mobile Computing, vol. 15, no. 9, pp. 2247-2259, Sep. 2016*. 10.1109/TMC.2015.2492556.

[7] W. Gau and H. Li, “Technologies for Spacecraft Antenna Engineering Design”, *Springer 2021*. 10.1007/978-981-15-5833-7

[8] P. Delos, B. Broughton, and J. Kraft Phased “Array Antenna Patterns”, *Analog Dialogue*, vol.54, May 2020.

[9] N. Maletic, V. Sark, M. H. Eissa, J. Gutierrez, E. Grass, and O. Bouchet, “Wireless communication systems in the 240 GHz band: Applications, feasibility and challenges”, *IEEE Proc. 16th Int. Symp. Wireless Commun. Syst. (ISWCS)*, Aug. 2019, pp. 436–440. 10.1109/ISWCS.2019.8877170

[10] Jakoby, Rolf, Alexander Gaebler, and Christian Weickhmann. 2020. "Microwave Liquid Crystal Enabling Technology for Electronically Steerable Antennas in SATCOM and 5G Millimeter-Wave Systems" Crystals 10, no. 6: 514. 10.3390/cryst10060514

10. Appendices

A) Grading Recommendation
B) MATLAB code
C) Chart of SNR for attack direction, when nulling is kept at 70°.

Appendix A:

המלצת ציון לדו"ח מסכם

אם יש צורך, לכל סטודנט/ית בנפרד

מספר הפרויקט: P-2023-009

הפרויקט: Cyber Security for next generation LEO Satellite Network Using Phased Array Antenna

שם המנחה: פרופ' שלומי ארנון

שם הסטודנט/ית: יערה הופמן ת.ז.: 206201022

שם הסטודנט/ית: אופיר יצחק ת.ז.: 205796238

%		חלש 55-64	בינוני 65-74	טוב 75-84	ט"מ 85-94	מצוין 95-100
20	הצגת גישת הפתרון, והתכנון ההנדסי					
20	הצגת התוצאות וניתוח השגיאות					
20	הסקת מסקנות					
10	גילוי יוזמה וחריצות					
20	פתרון בעיות, מקוריות ותרומה אישית (מעבר למילוי ההנחיות)					
10	עמידה בלו"ז ורמת הביצוע המעשי					

הערות נוספות:

Table of Contents


```
clc;
clear;
clear all;
```

Initiate uniform linear phased array

```
c = physconst('LightSpeed');
freq = 300e9;                                                          %
 carrier frequency 1GHz
lambda = c/freq;                                                       %
 wavelength
lookdir = 0;                                                           %
 look directions
noisedir = 40;                                                         %
 interference direction
```

Array & Steering Vector

```
array = phased.ULA('NumElements',30,'ElementSpacing',lambda/2);        %
 phased array with 10 antennas with spacing lambda/2
array.Element.BackBaffled = true;                                      %
 restricts the antenna response to azimuth angles in the interval [-90,90]
 degrees
```

generate friendly signal using a simple rectangular pulse, and recieve at the ULA

```
f = 10;                                                                 %
 set the frequency of the cosine signal to 10 Hz
t = linspace(0, 0.3, 300)';                                             %
 create a time vector
```

```
testsig = cos(2*pi*f*t);                                                    %
 generate a cosine signal
origsig = testsig;

npower = 0.05;
  % add white Gaussian noise at power 0.5
testsig = testsig + sqrt(npower/2)*(randn(size(testsig)) ...
          + 1i*randn(size(testsig)));                                       %
 simulate background noise

snr_in_db = calculate_snr(origsig, testsig);
fprintf('SNR: noised signal %.2f dB\n', snr_in_db);

angle_of_arrival = [0;0];                                                % signal
 arrives to antenna at 30deg and 0deg elevation
x = collectPlaneWave(array,testsig,angle_of_arrival,freq);                  %
 collect wave at the antenna (matrix with ten columns)
                                                                            %
 (Each column represents the received signal at one of the array elements)
```

generate the attacking signal

```
jammer = 10*sqrt(2)*randn(300,1);                                           %
 jammer noise is AWGN
jammer_angle = [noisedir;0];                                                %
 attacking signal arrives to antenna at 60deg and 0deg elevation
jampower= 5;
noisePwr = 1e-5;                                                            %
 added white gaussian noise
noise = sqrt(noisePwr/2)*...                                                %
 to simulate background noises
    (randn(size(jammer)) + 1j*randn(size(jammer)));
jammer = (jammer*jampower + noise);
  % total jamming signal includes noise
jamsig = collectPlaneWave(array,jammer,jammer_angle,freq);                  %
 collect attacker wave at the antenna (matrix with ten columns)

snr_in_db = calculate_snr(origsig, jammer);
fprintf('SNR jammer1: %.2f dB\n', snr_in_db);
```

generate the attacking signal2

```
jammer2 = 10*sqrt(2)*randn(300,1);
  % jammer noise is AWGN
jammer_angle2 = [-70;0];                                                 %
 attacking signal arrives to antenna at 60deg and 0deg elevation
jampower2= 5;
noisePwr2 = 1e-5;                                                           %
 added white gaussian noise
noise2 = sqrt(noisePwr/2)*...                                               %
 to simulate background noises
    (randn(size(jammer2)) + 1j*randn(size(jammer2)));
```

```
jammer2 = (jammer2*jampower2 + noise);
     % total jamming signal includes noise
jamsig2 = collectPlaneWave(array,jammer2,jammer_angle2,freq);
  % collect attacker wave at the antenna (matrix with ten columns)

snr_in_db = calculate_snr(origsig, jammer2);
fprintf('SNR jammer2: %.2f dB\n', snr_in_db);
```

generate the attacking signal3

```
jammer3 = 10*sqrt(2)*randn(300,1);
  % jammer noise is AWGN
jammer_angle3 = [-50;0];                                                    %
 attacking signal arrives to antenna at 60deg and 0deg elevation
jampower3= 5;
noisePwr3 = 1e-5;                                                              %
 added white gaussian noise
noise3 = sqrt(noisePwr/2)*...                                                  %
 to simulate background noises
    (randn(size(jammer3)) + 1j*randn(size(jammer3)));
jammer3 = (jammer3*jampower3 + noise);
     % total jamming signal includes noise
jamsig3 = collectPlaneWave(array,jammer3,jammer_angle3,freq);
  % collect attacker wave at the antenna (matrix with ten columns)

snr_in_db = calculate_snr(origsig, jammer3);
fprintf('SNR jammer3: %.2f dB\n', snr_in_db);
```

generate the attacking signal4

```
jammer4 = 10*sqrt(2)*randn(300,1);
  % jammer noise is AWGN
jammer_angle4 = [-20;0];                                                    %
 attacking signal arrives to antenna at 60deg and 0deg elevation
jampower4= 5;
noisePwr4 = 1e-5;                                                              %
 added white gaussian noise
noise4 = sqrt(noisePwr/2)*...                                                  %
 to simulate background noises
    (randn(size(jammer4)) + 1j*randn(size(jammer4)));
jammer4 = (jammer4*jampower4 + noise);
     % total jamming signal includes noise
jamsig4 = collectPlaneWave(array,jammer4,jammer_angle4,freq);
  % collect attacker wave at the antenna (matrix with ten columns)

snr_in_db = calculate_snr(origsig, jammer3);
fprintf('SNR jammer3: %.2f dB\n', snr_in_db);
```

generate the attacking signal5

```
jammer5 = 10*sqrt(2)*randn(300,1);
  % jammer noise is AWGN
```

```
jammer_angle5 = [60;0];                                                 %
 attacking signal arrives to antenna at 60deg and 0deg elevation
jampower5= 5;
noisePwr5 = 1e-5;                                                       %
 added white gaussian noise
noise5 = sqrt(noisePwr/2)*...                                           %
 to simulate background noises
    (randn(size(jammer5)) + 1j*randn(size(jammer5)));
jammer5 = (jammer5*jampower5 + noise);
     % total jamming signal includes noise
jamsig5 = collectPlaneWave(array,jammer5,jammer_angle5,freq);
  % collect attacker wave at the antenna (matrix with ten columns)


snr_in_db = calculate_snr(origsig, jammer5);
fprintf('SNR jammer3: %.2f dB\n', snr_in_db);
```

generate the attacking signal6

```
jammer6 = 10*sqrt(2)*randn(300,1);
  % jammer noise is AWGN
jammer_angle6 = [75;0];                                                 %
 attacking signal arrives to antenna at 60deg and 0deg elevation
jampower6= 5;
noisePwr6 = 1e-5;                                                       %
 added white gaussian noise
noise6 = sqrt(noisePwr/2)*...                                           %
 to simulate background noises
    (randn(size(jammer6)) + 1j*randn(size(jammer6)));
jammer6 = (jammer6*jampower6 + noise6);
      % total jamming signal includes noise
jamsig6 = collectPlaneWave(array,jammer6,jammer_angle6,freq);
  % collect attacker wave at the antenna (matrix with ten columns)


snr_in_db = calculate_snr(origsig, jammer6);
fprintf('SNR jammer3: %.2f dB\n', snr_in_db);
```

total recieved signal

```
rxsig = x + jamsig + jamsig2 + jamsig3 + jamsig4 + jamsig5 + jamsig6;
                                                  % total recieved signal is
 friendly signal and adversary signal
```

Implement the adaptive LCMV beamformer using the same ULA

```
Friendly = [0,5,15]; % in degrees
Hostile = [-70,-50,-20,60,75];
[merged, response] = merge_vectors(Friendly,Hostile);
```

```
steeringvector = phased.SteeringVector('SensorArray',array,...              %
 creates a steering vector object for our URA "array",
    'PropagationSpeed',c);                                                  %
 and sets the propagation speed to the speed of light.
LCMVbeamformer =
 phased.LCMVBeamformer('TrainingInputPort',true,'WeightsOutputPort',true);
LCMVbeamformer.Constraint = steeringvector(freq, merged);
LCMVbeamformer.DesiredResponse = response';
[yLCMV,wLCMV] = LCMVbeamformer(rxsig,jamsig);
snr_in_db = calculate_snr(origsig, yLCMV);
fprintf('SNR with nulling: %.2f dB\n', snr_in_db);
```

No nulling

```
beamformer = phased.PhaseShiftBeamformer('SensorArray',array,...
    'OperatingFrequency',freq,'PropagationSpeed',c,...
    'Direction',angle_of_arrival,'WeightsOutputPort',true);
[y,w] = beamformer(rxsig);
snr_in_db = calculate_snr(origsig, y);
fprintf('SNR without nulling: %.2f dB\n', snr_in_db);
```

plotting

```
figure
subplot(411)
plot(t,origsig, "color", 'b')
title('Original Signal');
hold on
grid on

subplot(412)
plot(t,testsig, "color", 'b')
title('Noised Channel Signal');
hold on
grid on

subplot(413)
plot(t,y, "color", 'b')
title('Received Signal Without Nulling');
hold on
grid on

subplot(414)
plot(t,yLCMV, "color", 'b')
title('Received Signal With Nulling');
hold on
grid on

figure
subplot(211)
pattern(array,freq,[-180:180],0,'PropagationSpeed',c,...
    'CoordinateSystem','rectangular','Type','powerdb','Normalize',true)
title('Array Response without Beamforming Weights')
```

```
subplot(212)
pattern(array,freq,[-180:180],0,'PropagationSpeed',c,...
    'type','directivity')

figure
subplot(211)
pattern(array,freq,[-180:180],0,'PropagationSpeed',c,...
    'CoordinateSystem','rectangular','Type','powerdb','Normalize',true,...
    'Weights',wLCMV)
title('Array Response with Beamforming Weights');

figure
pattern(array,freq,[-180:180],0,'PropagationSpeed',c,...
    'type','directivity','Weights',wLCMV)

function [directions, responsevec] = merge_vectors(FriendlyVec, HostileVec)
    % Check that all angles are in the valid range
    if any(FriendlyVec > 90) || any(FriendlyVec < -90) || any(HostileVec > 90)
 || any(HostileVec < -90)
        error('Angles must be in the range [-90, 90]');
    end

    % Check that there are no matching angles in the two vectors
    if any(ismember(FriendlyVec, HostileVec))
        error('Friendly and Hostile vectors cannot contain the same angles');
    end

    % Append a 0 to each element of the input vectors
    FriendlyVec2 = [FriendlyVec; zeros(size(FriendlyVec))];
    HostileVec2 = [HostileVec; zeros(size(HostileVec))];

    % Merge the vectors and sort by angle size
    directions = sortrows([FriendlyVec2 HostileVec2].');
    directions = directions.';

    % Create the response vector
    responsevec = ones(size(directions));
    responsevec = responsevec(1,:);
    % Find the indices where the angles originated from the Hostile vector
    HostileIndices = ismember(directions(1,:), HostileVec);

    % Set the corresponding values in the response vector to 0
    responsevec(HostileIndices) = 0;
end

function snr_db = calculate_snr(original_signal, noisy_signal)
    % Calculate the power of the original signal
    original_power = mean(abs(original_signal).^2);

    % Calculate the power of the noise signal
    noise_power = mean(abs(noisy_signal - original_signal).^2);

    % Calculate the SNR in dB
```

```
    snr_db = 10 * log10(original_power ./ noise_power);
end
```

Published with MATLAB® R2021b

Appendix C:

Chart of SNR for attack direction

when nulling is kept at 70°

Direction	SNR
60	-19.5261190507922
60.1	-18.6987149127307
60.2	-18.6907266447968
60.3	-18.0890266932679
60.4	-17.8164709855209
60.5	-17.2620021441566
60.6	-17.9725696587894
60.7	-17.4221120661599
60.8	-16.7120168300556
60.9	-17.4126493939636
61	-16.1149688269589
61.1	-15.3103451040106
61.2	-15.8677779388473
61.3	-15.4972333469069
61.4	-15.1557376855837
61.5	-14.7650651675525
61.6	-14.2839437847097
61.7	-14.8897219089399
61.8	-13.8108195329716
61.9	-13.6682344446312
62	-13.4911527236580
62.1	-13.0813409508403
62.2	-12.8764215759874
62.3	-12.4884301432638
62.4	-12.8819545130100
62.5	-12.3026373724146
62.6	-12.2225235029594
62.7	-12.1060711672204
62.8	-11.5740220949254
62.9	-11.6224829179684
63	-11.2388814866214
63.1	-11.5284916331000
63.2	-11.1899981571644
63.3	-10.9911593636247
63.4	-11.2437345024573
63.5	-11.9488126789470
63.6	-11.5457530309228
63.7	-10.3713763460890
63.8	-11.1275284994813
63.9	-11.4293265987304
64	-11.3883609990549

Direction	SNR
64.1	-11.4687445817046
64.2	-11.2444479932585
64.3	-11.8943054943070
64.4	-11.4231411347526
64.5	-11.1429093242034
64.6	-11.7159239453843
64.7	-11.4530716239659
64.8	-11.4754118698476
64.9	-11.0634500627898
65	-11.6990798965500
65.1	-11.7454450055620
65.2	-11.3400148463072
65.3	-11.3685075734818
65.4	-11.5824203096814
65.5	-11.7650927272244
65.6	-10.9775547451886
65.7	-11.0894960390559
65.8	-11.1362560276282
65.9	-11.1077464178087
66	-10.8106128349609
66.1	-10.9132811986750
66.2	-11.0667877677877
66.3	-10.4500855318378
66.4	-10.2821304228436
66.5	-9.91694824480793
66.6	-9.79418301812019
66.7	-9.64961041340060
66.8	-9.60084520246116
66.9	-9.23931761466782
67	-9.42825786928528
67.1	-9.54797721716291
67.2	-9.14491458973536
67.3	-8.79149256304965
67.4	-9.07872850779242
67.5	-8.16211167888784
67.6	-7.71842505986214
67.7	-7.28501815965351
67.8	-6.77350491070631
67.9	-7.16705183875881
68	-7.10240341236106
68.1	-6.86150526612776

Direction	SNR
68.2	-6.08665995113408
68.3	-5.66203297377297
68.4	-5.31385513907018
68.5	-4.54405058809949
68.6	-4.20265117534845
68.7	-3.84984475552565
68.8	-2.78106918451961
68.9	-2.05065211733748
69	-1.38218960152316
69.1	0.00221363626286922
69.2	0.560963242782333
69.3	0.768118749114539
69.4	2.51835650395146
69.5	3.99464586774844
69.6	4.95668095488986
69.7	6.57789792974057
69.8	8.58790035979558
69.9	9.34137043995761
70	10.1365372464506
70.1	9.13811740049446
70.2	8.19763181831083
70.3	6.34096609493527
70.4	5.07202814927057
70.5	3.66789477619012
70.6	1.85178085689040
70.7	1.10305209548683
70.8	0.0900729582972013
70.9	0.111346649652662
71	-1.18216206918860
71.1	-1.73852928995899
71.2	-2.92331514236314
71.3	-3.18773711007767
71.4	-3.95943645075578
71.5	-4.36260220971806
71.6	-5.24714524618478
71.7	-5.09485524258804
71.8	-5.91683129590565
71.9	-6.00739388954108
72	-6.20149453842616
72.1	-7.24737474719694
72.2	-7.39835039492118
72.3	-7.17963632195837
72.4	-7.69513586086311
72.5	-8.40248786541938
72.6	-8.06728930146132

Direction	SNR
72.7	-7.98534550317406
72.8	-8.91481882555384
72.9	-9.60182047490332
73	-9.29324923634911
73.1	-9.74890568753886
73.2	-9.61875019886988
73.3	-10.0690658656133
73.4	-10.8339925417235
73.5	-10.1549332441707
73.6	-10.9266062342231
73.7	-10.5595407724531
73.8	-10.9777297272875
73.9	-11.3245820168178
74	-11.4680256555856
74.1	-11.7503805512763
74.2	-11.3910969474008
74.3	-11.2389431229920
74.4	-11.0246977224274
74.5	-11.2802064251870
74.6	-12.2669585485630
74.7	-11.5348013996381
74.8	-12.3739349100420
74.9	-12.4534224228076
75	-11.6922008268914
75.1	-11.7373582631707
75.2	-11.6792296001955
75.3	-12.4847787667489
75.4	-12.4818922374678
75.5	-12.0273873372788
75.6	-12.0777967508286
75.7	-12.5887279095932
75.8	-11.6472169199491
75.9	-12.7840865879794
76	-12.0427613038631
76.1	-12.8824295876161
76.2	-12.6551688075282
76.3	-12.5929635636945
76.4	-12.5525856742134
76.5	-11.7448297708209
76.6	-12.6004588210444
76.7	-12.3268474892675
76.8	-12.3541111472992
76.9	-12.2567599289670
77	-12.8897591087403
77.1	-12.6948235482112

Direction	SNR
77.2	-11.8885500760016
77.3	-12.3466460314883
77.4	-12.4098993613160
77.5	-12.7181010089999
77.6	-12.4429411697047
77.7	-13.1936932695793
77.8	-13.1066083852134
77.9	-13.2682267983157
78	-12.6868007674609
78.1	-12.6392529966828
78.2	-11.7410463138084
78.3	-12.9363800194334
78.4	-12.7944911975065
78.5	-12.7320562226775
78.6	-12.8538173325726
78.7	-12.5712032522020
78.8	-12.6849561217900
78.9	-12.0145860581036
79	-11.8359712757189
79.1	-12.8676621155847
79.2	-12.4940672065466
79.3	-12.4856440490028
79.4	-12.1083850072489
79.5	-12.4125220978464
79.6	-12.5372000142359
79.7	-12.4617916783780
79.8	-12.9480186158570
79.9	-12.0178663791633
80	-11.9289205588558