**ANVESHAK COMMS**

So to plan the communication b/w rover and base station the main thing we need to see is the range of communication .

Firstly we planned to make a LAN setup in our rover such that all the microcontroller and jetson are connected to the router and they communicate with it instead of connecting the microcontrollers directly to the jetson. If this is done then we can transmit video and other sensor data independent of jetson and we can even do teleoperations on the rover independent of jetson.

So for this we have to increase the range of our communication.

So we thought about changing the antenna used.

We have to remember things like the gain of the antenna should be more ,for every 6db gain we can increase the double the range so (ex if gain is 24db we can transmit data up to 16 times the initial range) ,our rover has omni directional antenna whose gain is 13dbi which is less as we are not able to do comms properly ,and omni directional antenna is not that suitable for receiving data and is it has a small area for receiving the signal on the other hand we could use a parabolic antenna with high gain like AG-HP-2G20 (from ubiquiti this a airgrid) it has a gain of 20dbi

Advantages of parabolic antenna over omni:

* Long range communication with better strength is advantageous on both transmitting and receiving end but we have to make sure that parabolic antennas are in line of sight which is not required in omni (but the range is less).

To achieve this we can have our antennas fixed on a motor which automatically points towards the base station and from the base station towards the rover .

For this we can use map localization , like the rover knows where it is from the base station even though it is not connected so it can point automatically towards the base station and the base station knows where the rover is as the rover continuously updates its location to the base station which helps the base station antenna to point towards the rover .

Recovery Mode:

* If at any time the rover gets disconnected from the base station then it will enter into recovery mode as the rover knows its whereabouts from the base station it will point its antenna towards base station ,but the base station may not have exact location so it will rotate it’s antenna in all directions slowly until it receives signal from rover ,incase the we are unable to receive any data from rover in the base station then the rover will first calculate the distance from base station if the distance is more than the maximum range we got during our testing then it goes autonomous (if ready).

Hardware planned to use is :

* Ubiquiti air grid M2: AG-HP-2G20:https://dl.ui.com/datasheets/airgridm/airGrid\_HP.pdf
* Or Powerbeam ac :PBE-2AC-400 :<https://dl.ui.com/datasheets/PowerBeam_ac/PowerBeam_PBE-2AC-400_DS.pdf>

SRT Vs UDP Vs TCP

* SRT offers the better qualities of TCP and UDP in a single protocol. The biggest positive of SRT is that it provides error correction while having low latency.
* SRT also provides congestion control, and provides real time streaming of video with subsecond latency.
* SRT provides high reliability by recovering lost or corrupted packets, ensuring the delivery of data even in challenging network conditions.
* UDP offers fast transmission of data due to no connection and handshake process.
* UDP is unreliable because there is no error correction mechanism or retransmission of data packets.
* TCP is reliable because it ensures the transmission of data through error checking mechanisms, flow control and retransmission of data packets.
* TCP is relatively slow and has higher latency because it is a connection based protocol.

We read about the intra rover communication protocols which could be used. The major protocols we focussed on were I2C, SPI, UART, CAN, LAN and WAN. We also looked into LoRa, Zigbee, Sigmafox and many other protocols for rover to base station communication. We studied about how to create temporary LAN connections, ad hoc network, VLAN and WPLAN. We went deep into the working of CAN, and studied the softwares involved. We also saw about the implementation of TCP, and CAN. For TCP simulation, we had two ways shown on the internet. One used python socket library while the other one used WSIM. The python was easier and apparently the WSIM was a usage of TCP Client/Server. We committed a few errors by looking into cloud computing using the internet forgetting the fact that internet was not allowed in the competition. For video transmission we looked at a lot of protocols like TCP, UDP, SRT, Zixi, RTP, RTMP, RTSP, HTTP(HLS) and RTP over UDP. At some point we even thought of making a custom protocol, which might just be the combination of the goods of TCP and UDP, but then SRT was already made just for that purpose.

Communication Protocols.

COmParison report(Brief description plus advantages and disadvanatges)

UDP (User Datagram Protocol):

UDP is a fundamental yet ubiquitous protocol. It is lightweight, has low overhead, and offers minimal latency. But it comes with a catch!

Advantages:

We discovered that UDP's low overhead means it has a smaller header size compared to TCP. This results in less network congestion. Moreover, UDP's connectionless nature makes it perfect for applications that need low latency, such as real-time video streaming and gaming.

Disadvantages:

UDP lacks reliability. It does not have built-in error checking or retransmission, it can't guarantee that all packets will reach their destination or arrive in the right order. Plus, it doesn't have sophisticated congestion control, so it's prone to network congestion and degraded performance in busy environments.

Applicability:

Based on our research, UDP shines in situations where low latency and reduced overhead are essential. That's why it's commonly used in real-time multimedia streaming, VoIP, DNS, and gaming applications.

TCP (Transmission Control Protocol):

TCP is a slightly better version of the UDP protocol . It's all about ensuring data integrity and ordered delivery. But, as we found out, it comes with its own set of pros and cons.

Advantages:

TCP guarantees reliable and in-order delivery of data. It does this by using acknowledgments and retransmissions to ensure data integrity. TCP also has fancy flow control mechanisms to prevent data loss due to congestion, making it great for applications that need reliable transmission.

Disadvantages:

TCP's reliability comes at a cost. It has a larger header size compared to UDP, resulting in higher packet overhead and increased network congestion. Additionally, TCP's connection-oriented nature means it requires a three-way handshake for connection establishment, leading to higher latency compared to UDP. That makes it less suitable for real-time applications where low latency is crucial.

Applicability:

Our findings showed that TCP is widely used in applications like web browsing, file transfers, email, and other non-real-time applications that require reliable and ordered data delivery.

SRT (Secure Reliable Transport):

SRT is like the brilliant lovechild of UDP and TCP. It combines the best of both worlds and adds its own unique features. We were particularly excited to dive into SRT and explore its advantages and applications.

Advantages:

SRT blew us away with its reliability and low latency. It offers error recovery mechanisms like retransmission and forward error correction, ensuring reliable delivery of media streams even in challenging network conditions. And here's the cool part: SRT dynamically adapts to changing network conditions to optimize performance and stream quality. Plus, it supports end-to-end encryption for secure transmission and has bandwidth optimization features.

Applicability in a Rover for Communication:

In such scenarios like exploring a distant planet or handling rescue missions, reliable communication is vital. SRT's advantages make it a perfect fit. It ensures reliable transmission of critical data and video feeds, even in unpredictable and bandwidth-limited environments. Its low latency characteristics allow real-time control and monitoring of the rover, enabling quick responses to changing conditions.