State of Untethered Virtual Reality

Norman Ponte
Carnegie Mellon University
nponte@andrew.cmu.edu

ABSTRACT

With the data rate of wireless links increasing at exponential rates high-end virtual reality (VR) headsets still remain tethered to a console or pc. Many companies are searching for the way to remove the tether. With all the existing work being done in the graphics and network communities to the graphics pipeline and wireless links is it possible to remove the tether. Are all these optimizations enough to meet the high bandwidth and low latency requirement of virtual reality or are we stuck waiting for upcoming technologies to fix some of the issues. In this paper I examine the current state of VR and the future challenges that must be handled in order to remove the tether from virtual reality and deliver a similar experience [1].

1. INTRODUCTION

The field of virtual reality is fairly new and people are still figuring out what direction they want to take with this new technology. One common goal however is to remove the tether and allow for a virtual reality experience with no cables. The problems they face are iterations of problems that have been part of the network and graphics research and we can often find papers detailing different approaches to these problems. As it stands however removing the tether is still a very complex problem. Current systems use the tether to transmit data from high compute GPUs to the displays on the headset.In a virtual reality system the general consensus is that the pipeline goes like this.

1. Sample User Input
2. Run the Game Simulation
3. Render the Scene
4. Finish the Graphics of the Scene
5. VSync
6. Scanout to Display

The link between the headset and console/pc is usually ignored because it has the needed bandwidth and the latency is negligible. This tether gives them a link to transmit frame information with a high bandwidth and a very low latency. With the removal of this tether developers have to make sure they still meet the two main requirements for delivering a compelling virtual reality experience while still respecting the costs of the other parts of the pipeline. In general removing the tether comes down to delivering on two basic properties required from a virtual reality system.

1. Low Latency between Input Sampling and Video Scanout [$\approx 50ms$]
2. 90Hz High Resolution Video [$\approx 7Gbps$]

In this paper I will take the approach of seeking to solve the second requirement first and then determine what that cost was in terms of latency and then look for techniques that could cut latency from the system so that the first requirement can also be satisfied.

The rest of the paper is organized as follows. First I will report on the current state of virtual reality followed by upcoming technology that could change the landscape of virtual reality. Then I will look at four of the areas that could lead to the solution of this problem. Next I will examine standard wireless link technology and optimizations to increase bandwidth. Next I will look into techniques to reduce the size of the frames sent over the link to the headset mainly using compression. Finally I will examine graphics techniques to meet the latency requirements of the system. To finish the paper I will report on the end-to-end system viability of wireless virtual reality and describe future work in achieving the stated goal.

2. STATE OF VIRTUAL REALITY

Virtual Reality as a field has experienced a high degree of growth and re-factoring in recent years. As more companies begin their foray into Virtual Reality the field as a whole becomes more complicated and specialized. Although there are many different products in this field from many different companies on a high level their products follow the same rules and guidelines. If you look at *Table 1* you can see most of the top tier headsets follow the same pattern and most require the same graphics component from the attached (NVIDIA GTX 970). The key feature I want to focus on in this graph is that the data requirements are roughly the same for all the devices and they all rely on some version of the HDMI cable in order to transport the video from the pc or console to the headset.

Table 1: VR/AR Headsets

Product	Resolution [Per Eye]	Refresh Rate(Hz)	Color Depth(bit)	Bandwidth (Gbps)	Tether
Oculus Rift	2160x1200 [1080x1200]	90	10	7	HDMI 1.3
HTC Vive	2160x1200 [1080x1200]	90	8	7	HDMI 1.4
PlayStation VR	1920x1080 [960x1080]	90-120	8	6.22-8.29	HDMI 2.03
Meta2 (AR)	2560x1440 [12801440]	60	10	≈ 7.71	HDMI 1.4

Table 2: HDMI Latency + Bandwidth

Version	Bandwidth	Bandwidtd(No Overhead)
1.3/1.4	10.2 Gbps	8.16 Gbps
2.0	18 Gbps	14.4 Gbps

These data requirements are a factor of the video resolution they output and the color depth they export. This requirement of roughly 7 Gbps makes HDMI cables a very logic link to transfer the data as this is a higher rate then standard wireless technologies and a low enough link to not require specialized cables. Looking at *Table 2* you can see this fits well into the HDMI use case. Another key factor is that HDMI cables transmit data at about 80 percent of the speed of light meaning that we can assume the added latency over this link to be negligable.

In attempt to remove the tether the approach some systems are taking is to move away from the traditional fully immersive virtual reality and settle with a subset of that experience. This is usually achieved with augmented reality and gear based virtual reality. Both of these systems remove parts of the virtual reality system and this allows them to exist without the tether. The first of these is gear based virtual reality. The two most common products here are the Samsung Gear VR and Google Cardboard. The main problem with these gear based VR is that the video processing is done on the headset which means you have much less graphical compute power this mean you are either outputting a low resolution display or completely removing positional tracking and limiting the field of view (FOV). The other direction companies such as Microsoft are taking is augmented reality. Augmented reality does not imply a lack of a tether as one of the most common AR headsets is the Meta2 which is also tethered due to the data requirements *Table 1*. Augmented Reality however can take the approach which is similar to the approach taken by gear based VR which is just to reduce the data requirements from the headset. The flagship product of untethered augmented reality is the Microsoft HoloLens. The HoloLens uses a custom HPU chip which is 200x faster then CPU in calculations used to formulate the augmented reality this allows them to meet the mentioned latency requirements. The data is then passed to the CPU which can output to the display. This combination of specialized hardware and lower data requirements (1268x720) allow Microsoft to exist without being tethered to high compute engine. As a whole it seems that companies are either straying away from classic untethered virtual reality experience or have innovations in the pipeline.

3. UPCOMING ADVANCES

Recently there has been some key advanced that can possibly lead to the end of the tethered experience without a loss in the quality of the virtual reality experience. One of the most logical ways of solving this problem is to move the processing power to the headset and for most of the lifespan of virtual reality this has not been a possibility because of the incurred round trip delay of the cost of splitting the computation between two devices [2]. This was mostly a result of the fact that chip technology for the headset could not output the needed graphics at the needed latency while complying to power requirements. This may no longer be the case. Both Qualcomm and Intel have in development their own prototypes of untethered virtual reality. Both of these prototypes are based around specialized processors the Snapdragon 835 and a chip from Movidius which is not yet been released. These advances could be the way to solve the need of the external compute power but until they have been released and the chips are analyzed we will not know what they have capable of or if it can deliver the same quality of an experience.

The second advance comes from MIT. The paper Cutting the Cord in Virtual Reality was released last year and looks to replace the HDMI cable as a means of transmitting video to the headset [3]. In the paper the students propose using mmWave technology to deliver the data. This idea is not original as many have pointed to high frequency RF signal as having the highest data rate of wireless signals. In the paper they solve the problem that blocked people in the past in the fact that mmWaves are very prone to being blocked by objects in the way. The authors show that with their MoVR reflector they can minimize the signal to noise ration allowing the high-data-rate link of mmWaves to be used to communicate with the VR headset even in presence of blockage.

Although both of these advances show a high amount of promise the question still remains what can we currently achieve with more common link technologies and more advanced recent optimizations. Is the dream of tethered virtual reality sidelined until these technologies become more standard and researched or can we achieve something close to that with current technology.

Table 3: Network Bandwidth

Location	TCP Rate	UDP Rate	Packet Loss
Gates	200 Mbits/sec	241 Mbits/sec	60%
Hammerschlag Hall	150 Mbits/sec	200 Mbits/sec	71%
Home	20 Mbits/sec	34 Mbits/sec	65%

4. INCREASE LINK BANDWIDTH

Without optimizing any other part of the pipeline is it possible to achieve the required data rate from standard wireless technologies or at the very least how close can we get to the required data rate. For our experiment we will be using 7 Gbps for the required data rate. This is what both the HTC Vive and the Oculus Rift require.

4.1 New Wireless Link Technologies

As discussed before high frequency waves are one of the solutions being considered by many in this industry. These mmWaves have Low Latency and High Bandwidth which are the two main desires. 60 GHz connection can be used for short rage and offer up to 2.5 Gbps while the rest of them 71-76,81-86,92-95 GHz can offer up to 10 Gbps but require a transmitting license to use. These high frequency waves must maintain line-of-sight (LOS) between the receiver and the transmitter or the link is not effective. As discussed before this problem has been solved by the researchers at MIT. The problem is that communication at this frequency requires high frequency components such as amplifiers and mixers which are not as available for commercial use which hinders mmWaves as the solution to the problem.

In the world of Internet of Things (IoT) Li-Fi has gained a high degree of attention. There are many advantages to Li-Fi with perhaps the best of them being the advertised data rates. Li-Fi has been shows to achieve a rate of 3.43 Gbps for a single RBG triplet of LD [4]. They have been shown to get up to 14 Gbps in well-illuminated areas and up to 100 Gbps when using WDM with 36 parallel streams. The only negatives around Li-Fi is that since it uses Light waves it does not bounce well and thus LOS is required. Li-Fi has the potential to be the solution to the problem we face but a mass roll out of Li-Fi is not coming anytime soon so we have to look elsewhere.

4.2 Existing Link Technologies

For this paper I wanted to focus on widespread wireless use. The most common wireless link technology is Wi-Fi which is based on the IEEE 802.11 standard. In **Table 5** you can see the theoretical link rate using the different versions of the 802 protocol. While these numbers seem promising the problem is that these advertised rates are often very far from the actual data rates achieved between connections. This is because the high overhead in the IEEE 802 protocols combined with the cost of TCP re-transmissions leads to a much smaller effective throughput [5]. For my end-to-end system I wanted to document data rate speeds on both my home network and on the campus network. Using network analytic tools such as iPerf [6] tcpdump [7] and traceroute analyzed the data rate and packet loss using UDP and TCP protocols with the optimal packet size found. In *Table 3* you can see that the rates for both TCP and UDP are very low. Packet loss was a huge problem in all three of these cases. For both of the campus network measurements I found I was being routed through three hops which meant the possibility for bit error was even greater then the single hop example of the home network. Partial packet recovery such as Maranello have been shown to improve throughput of links by up to a factor of two but that still would only put us at about 400 Mbit/s [8] Latency was also very important for my calculations so in all three locations I measured the RTT **Table 4**. What stood out here was that the usual time was fine for all of the locations but the outliers would lead to large problems in the virtual reality experience. While some of the numbers might seem to be bugs with the network it has been shown that RTT can be sampled accurately over a TCP connection [9].

Table 4: Network RTT[ms]

Location	Mean	Median	Max
Gates	8.65	4.6	140
Hammerschlag Hall	6.5	3.7	18.4
Home	5.25	18.7	281.7

Table 5: IEEE 802.11

Protocol	Data Rate
802.11a	54 Mbps
802.11b	11 Mbps
802.11g	54 Mbps
802.11n	600 Mbps
802.11ac	1.3 Gbps

Table 6: glTF Compression

Algorithm	WoodTower(640Kb)	Man(1.6Mb)	SciFiHelmet(4Kb)
Speed[us]: compression + decompression			
lz4fast 1.7.5 -17	3+1	394+101	1062+316
lz4fast 1.7.5 -3	3+1	718+149	1756+439
zstd 1.1.4 -1	10+6	2232+877	4506+2018
shrinker 0.1	4+2	3084+871	5620+1285
pythy 2011-12-24 -6	6+1	1965+389	2456+567
Compression rate[%]			
lz4fast 1.7.5 -17	49.31	81.67	67.15
lz4fast 1.7.5 -3	33.71	73.11	53.39
zstd 1.1.4 -1	22.26	49.56	31.91
shrinker 0.1	28.81	63.40	45.54
pythy 2011-12-24 -6	30.01	66.85	45.22

5. DECREASING DATA REQUIREMENTS

In general there are three ways of decreasing the data requirements one of which is just outputting a lower quality of video which can include lowering the number of bits used to represent color or decreasing the resolution of the video. This will not be covered as this does not deliver the same type of experience as what we currently have. The second way is an active area of research which is modifying what information is sent to the headset to represent a frame or moment in the virtual reality world. Many companies have their own proprietary solutions for this and this problem is very application and hardware dependent so for this general analysis I will skip this as well. The final way and the most common way is compression and decompression.

5.1 Compression

Compression is a very desirable idea when it comes to wireless virtual reality. As I discussed earlier in the paper with the amount of data we have to transmit we are most likely going to need to compress the frames in order to meet our throughput requirements. The main issue with compression is the issue that remains at the heart of most of these discussions that being latency. When compressing and then decompressing a frame we add a latency cost that we are unable to circumvent by doing in parallel or by pipe lining it with other parts of the end-to-end system. The key to compression algorithms is the trade off between how much an object is compressed and how long the process takes. In **Table 4** [10][11][12][13][14] you can see many of the premier compression algorithms and their effects of compressing three different glTF scene which vary in size and material. Many virtual reality implementations vary on what information is sent over the link between the compute and headset elements and this is still an active area of research. For this paper I choose a glTF format which is very similar to the ovrscene that oculus uses. Most companies keep what information they are sending secret and glTF [15] is an open source effort to not only provide a general format for these 3D scenes but to optimize how much information is actually needed to generate these scenes. From the graph the compression and decompression times are very reasonable compared to our overall goal of 50 ms when compressing by moderate amounts when we compress much more heavily the time added is greater but still somewhat near our range. Overall compression seems to be a definite part of the final system implementation.

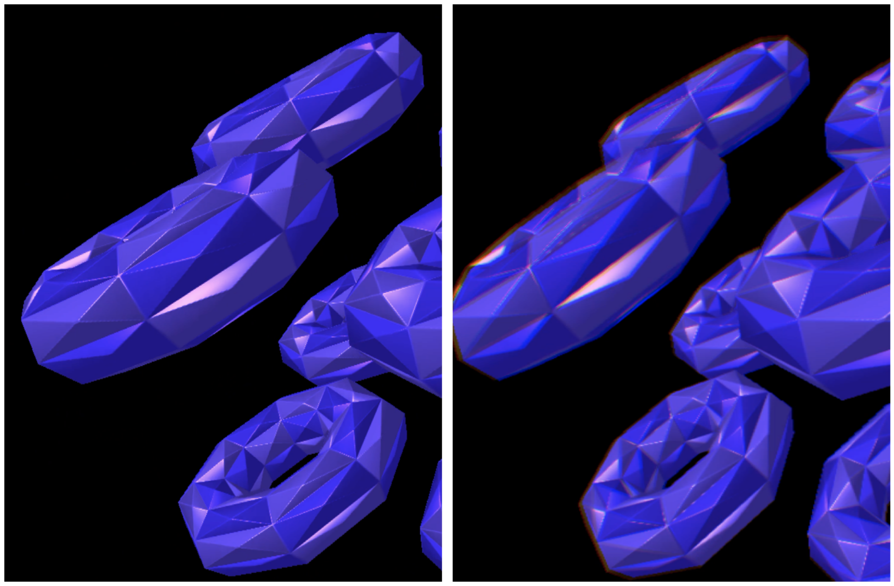

Figure 1: Normally Rendered vs ATW Rendered

Table 7: ATW Rendering Time

Scene Complexity	Normal	ATW
Simple	$\approx 6ms$	$\approx 2ms$
Medium	$\approx 24ms$	$\approx 12ms$
High	$\approx 31ms$	$\approx 18ms$

6. DECREASING LATENCY

Latency is a very important requirement for virtual reality and it is changed heavily with the removal of the tether. The human eye will start to notice the latency in generation frames at around 60 ms giving the system about that much time to output the frame to the display. A more conservative goal is around 50 ms and this is the approach taken by many virtual reality products.

6.1 Asynchronous Time Warp

Asynchronous time warp (ATW) is a new technique that many are looking toward to offset the latency issues that are arising with more complicated virtual reality implementations. The basics of ATW is that if a frame is not able to be developed before vsync then ATW produces a new frame based on warping the new image with the new positional tracking information. The main power of ATW is to smooth out inconsistent frame rates. The problem with ATW is that it is a very expensive solution to still deliver a image that is going to suffer from jutter with near proximity objects. To generate a frame based on the last frame synchronously with generating what would be the real frame is expensive and is not well supported on current GPUs [16]. If done on a CPU it is not possible to offer a high performance time warp due to the power constraints and on a GPU we must be able to preempt normal rendering to run time warp which cannot be done well on existing GPU architecture. As with many of the other optimizations in this paper ATW is stuck waiting for advances.

In the experiments I ran to determine the effect of asynchronous time warp on both the quality of the image displayed and the latency times I found some interesting truths [17]. The first of these truths was the effect that asynchronous time warp has on latency. In **Table 7** you can see the speed of asynchronous time warp for generating frames compared to the regular speed of the frames. In the case where we want to achieve 90 Hz atw can allow us to not drop the frame rate even in the face of the renderer not being able to render the new image in time. This shows how quickly atw can provide frames bringing the whole system latency down to about 13 ms. There are two important facts from this chart I want to highlight. The first of these is the need to perform these operations in parallel. In the medium complexity scene the atw was able to make the cutoff to maintain the frame rate (16 ms) but combines (24+12) this would not make the cutoff. The second trend to notice is that the atw rendering cost is relative to the normal rendering costs and given a rendering of a complex scene atw will still add non-negligible latency to the system.

The negatives of atw are very evident however in **Figure 1** it is very easy to see the effects of atw has on the image that is being displayed compared to the scene when no atw is being used. This image relates to the medium scene complexity in the table. In this experiment the delta in input parameters is fairly constant and can be estimated well by the atw unit. In cases where the input parameters are not as constant atw is unable to be used as the image that is output is not an accurate depiction of the scene.

7. END-TO-END SYSTEM

Putting it all together into a cohesive system turned out unsuccessful. In my case I was unable to find a wireless link that gave the other components of the system a chance to work. This is mostly due the fact that wireless link technologies in the real world have very low effective data rates and most people turn to wires or rely on compression techniques to stream high quality content on their devices without the worry of input commands and render time. Even though in this case the link is too much of a limiting factor the other optimizations have put us in a place where if the link can improve we could realistically have wireless virtual reality.

For most of these optimizations if combined in the correct way and with little error we could achieve our goal. Using aggressive compression we could bring down the amount of information sent over the wire by a factor of 2 even more in some cases. If we were able to find a wireless link with an effective throughput of around 3 Gbps then we could have a functioning system. The extra latency added by this compress and decompress could be made up by using asynchronous time warp letting us meet the frame rate requirements. In the case were the user input is unpredictable we are unable to use atw and we would not meet the time requirement with these extra compression costs. Another unsolved problem is where the network packet get dropped and take a long time to reach the headset (201 ms RRT for some packets) this would lead to missing the latency deadline. As a whole if any of these negative scenarios happen or happen simultaneously the system will fail so it seems for now that people will prefer the safety and reliability of the cord until we can solve the problems in wireless connections or in improving graphics rendering techniques.

8. CONCLUSION AND FUTURE WORK

In the end it seems that we are left waiting on new advances to make this idea a possibility. Asynchronous time warp, compression algorithms, and more compact frame representations are steps in the right direction and those parts of the system offer much promise in losing the tether. This problem is defined by a series of trade off but ultimately it is at the hands of the weakest link of the system which is the wireless link itself which cannot in a conventional setting provide the service needed. Testing on both campus and home wireless networks put me very far away from the needed bandwidth even with aggressive compression and asynchronous time warp. Akami reported that the network speed for the average American is 11 Mbps and this includes the overhead and the inefficiencies discussed earlier. Even with all the new optimizations in the fields of both graphics and networks there are intrinsic properties in wireless links that mean with existing technology and algorithms we are unable to remove the tether while providing a experience of the same quality.

For future research I would want to finish implementing the end to end system with the presence of a high speed link. I was unable to combine many parts of the system and was left evaluating the integral parts of the system by their own and forming conclusions based on the separate parts. Another integral part to research in this field is that most companies do not release their internal algorithms and code base or if they do its tied to their product and relatively undocumented custom calls and procedures. This is a relatively new field and most companies want to keep their research and ideas proprietary and as such algorithms vary heavily from device to device and there is often not a base algorithm or work flow that is applied. For the future I would take a specific product and attempt to dive into that specific product and code base instead of coming to an overarching conclusion on the state of virtual reality.

9. REFERENCES

[1] Carmack, John. "Latency Mitigation Strategies." *AltDevBlogADay*. N.p., n.d. Web. 08 May 2017.
[2] Banerjee, K.s., and E. Agu. "Remote execution for 3D graphics on mobile devices." *2005 International Conference on Wireless Networks, Communications and Mobile Computing* (n.d.).
[3] Abari, Omid, Dinesh Bharadia, Austin Duffield, and Dina Katabi. "Cutting the Cord in Virtual Reality." *Proceedings of the 15th ACM Workshop on Hot Topics in Networks - HotNets '16* (2016).
[4] Tsonev, Dobroslav, Stefan Videv, and Harald Haas. "Towards a 100 Gb/s visible light wireless access network." *Optics Express 23.2* (2015).
[5] Chen, Qi, Felix Schmidt-Eisenlohr, Daniel Jiang, Marc Torrent-Moreno, Luca Delgrossi, and Hannes Hartenstein. "Overhaul of ieee 802.11 modeling and simulation in ns-2." *Proceedings of the 10th ACM Symposium on Modeling, analysis, and simulation of wireless and mobile systems - MSWiM '07* (2007).
[6] IPerf https://iperf.fr/
[7] Tcpdump http://www.tcpdump.org/tcpdump_man.html
[8] Bo Han, Aaron Schulman, Francesco Gringoli, Neil Spring, Bobby Bhattacharjee, Lorenzo Nava, Lusheng Ji, Seungjoon Lee, Robert Miller. "Maranello: Practical Partial Packet Recovery for 802.11".
[9] Karn, P., and C. Partridge. "Improving round-trip time estimates in reliable transport protocols." *Proceedings of the ACM workshop on Frontiers in computer communications technology - SIGCOMM '87* (1988).
[10] lz4 https://github.com/lz4/lz4.
[11] Zstandard http://facebook.github.io/zstd/.
[12] Shrinker http://www.rubydoc.info/gems/shrinker/0.1.2.
[13] pythy https://docs.python.org/2/library/archiving.html.
[14] lzbench https://github.com/inikep/lzbench.
[15] glTF https://github.com/KhronosGroup/glTF.
[16] Waveren, J. M. P. Van. "The asynchronous time warp for virtual reality on consumer hardware." *Proceedings of the 22nd ACM Conference on Virtual Reality Software and Technology - VRST '16* (2016).
[17] atw https://github.com/KhronosGroup/Vulkan-Samples.