SC4052 Assignment 1

Gokul Ramesh U2222182H

Contents

1. Literary Survey

Transmission Control Protocol (TCP) algorithms are a core element in establishing regulations and standards for data transfers between multiple programs over networks (G, 2024). Currently, TCP algorithms are used in datacenters to help optimize the data center network (DCN) and manage congestion. Some currently used variants of TCP include Data Center TCP (DCTCP) (Alizadeh et al., 2010) and Data Center Quantized Congestion Notification (DCQCN) (Zhu et al., 2015) which help enhance overall network performance and improve network responsiveness and efficiency respectively. However, as demands change, data centers will evolve with them while incorporating new characteristics into their infrastructure.

The most recent trends pertaining to data centers are the focus on rapid expansions to meet future demand (Lee et al., 2025), and the focus on sustainability and carbon neutrality through greater usage of renewable energy resources (*TOP 10 Data Center Trends, 2025*). These trends highlight how much importance we have given to taking environmentally friendly approaches even while the demand for data centers continues to rapidly increase.

On the algorithm side, current research is heavily focused on using Artificial Intelligence (AI) and machine learning (ML) to optimize the various operations in a data center such as workload optimization and capacity planning (*The Impact of Artificial Intelligence on Data Centers: A Comprehensive Analysis*, n.d.). These trends suggest a high likelihood of future TCP algorithms working in conjunction with AI to achieve greater network efficiency, while reducing costs and improving the carbon footprint of data centers.

As an example, while the current DCTCP algorithm uses Explicit Congestion Notification (ECN) to estimate congestion levels and adjust its parameters, a futuristic version of DCTCP that leverages AI would be able to analyze real-time network conditions and data through reinforcement learning (J. Liu et al, 2*024)*, and more accurately adjust congestion control parameters for better efficiency, leading to better network performance.

Another example is while traditional TCP algorithms may face challenges in the data center environment which hosts a variety of applications, with some needed small, predictable latency, and other needing large, sustained throughput (Archiveddocs, 2016), we could instead use AI powered algorithms like PowerTCP which use machine learning models to leverage in-band network telemetry to achieve finer congestion control and be able to react to network changes without loss of throughput, which increases in importance in the future where data centers will experience larger and more diversified workloads with greater demands. (Addanki et al., 2021).

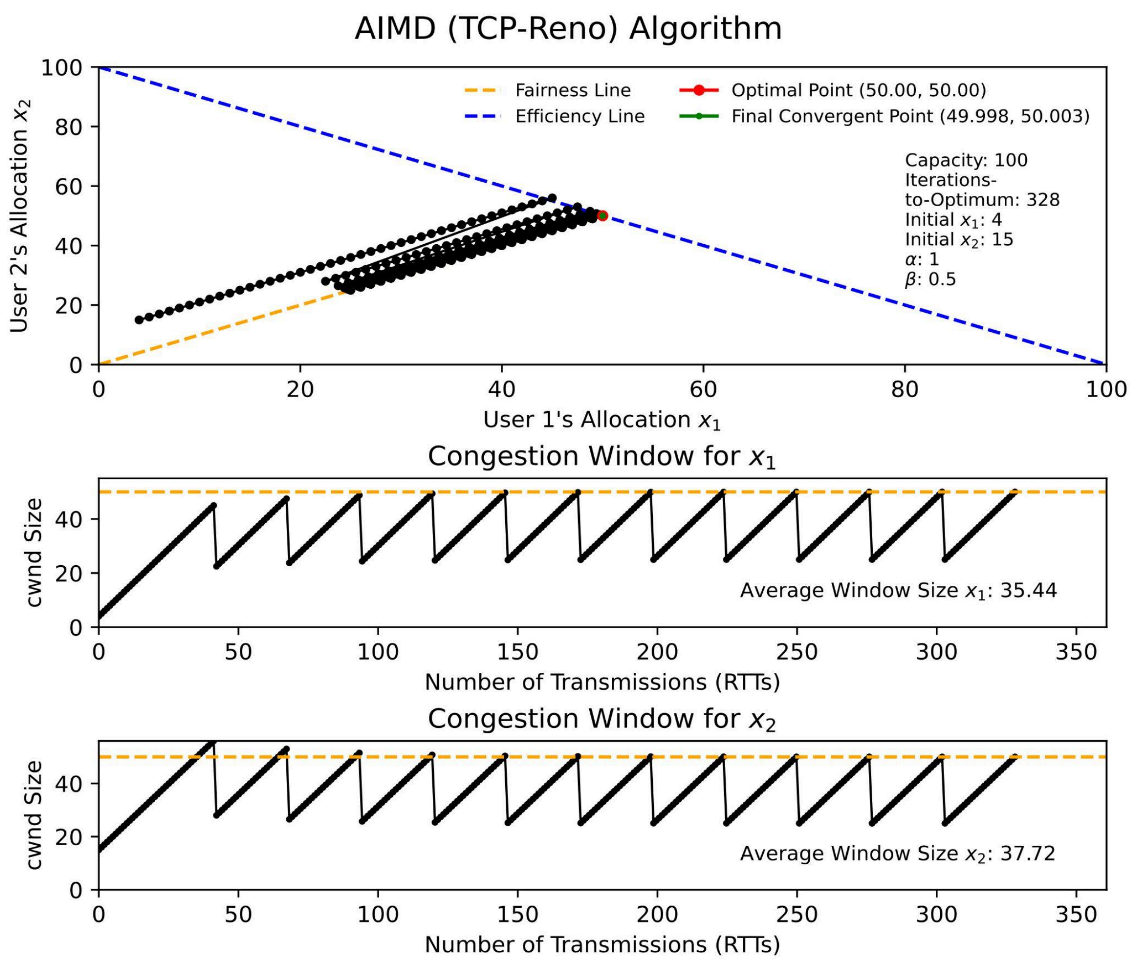
For the foreseeable future, it seems that any step forward will utilize AI in one way or another, which will unlock endless possibilities for what humanity can accomplish with bigger and faster machines. Next, I will discuss my experiments on various TCP algorithms.

2. Traditional AIMD (TCP-Reno)

The first algorithm that I experimented on was the standard Additive Increase Multiplicative Decrease (AIMD) algorithm that is also known as TCP-Reno. The 2 phases are represented by the following equations :

- Additive Phase : $cwnd = cwnd + \alpha$
- Multiplicative Phase : $cwnd = cwnd * \beta$

According to the traditional algorithm, I used the values of $\alpha = 0.5, \beta = 1$, for the implementation.



It can be observed from the first table that the AIMD algorithm is able to converge to the optimal point, which is the intersection of the fairness and efficiency line, in 328 iterations.

However, the network is not constantly fully utilised as the Bandwidth Utilization, which is the sum of the average window size over the total capacity, is only 73.16%, which is significantly far from full network utilization. This is because the linear increase is too slow in reaching the maximum capacity, while the 0.5x decrease is too large for the linear increase to quickly recover from. This leads to TCP-Reno being unfavourable for data centers, due to its poor throughput and high latency.

3. Exponential-to-Linear TCP

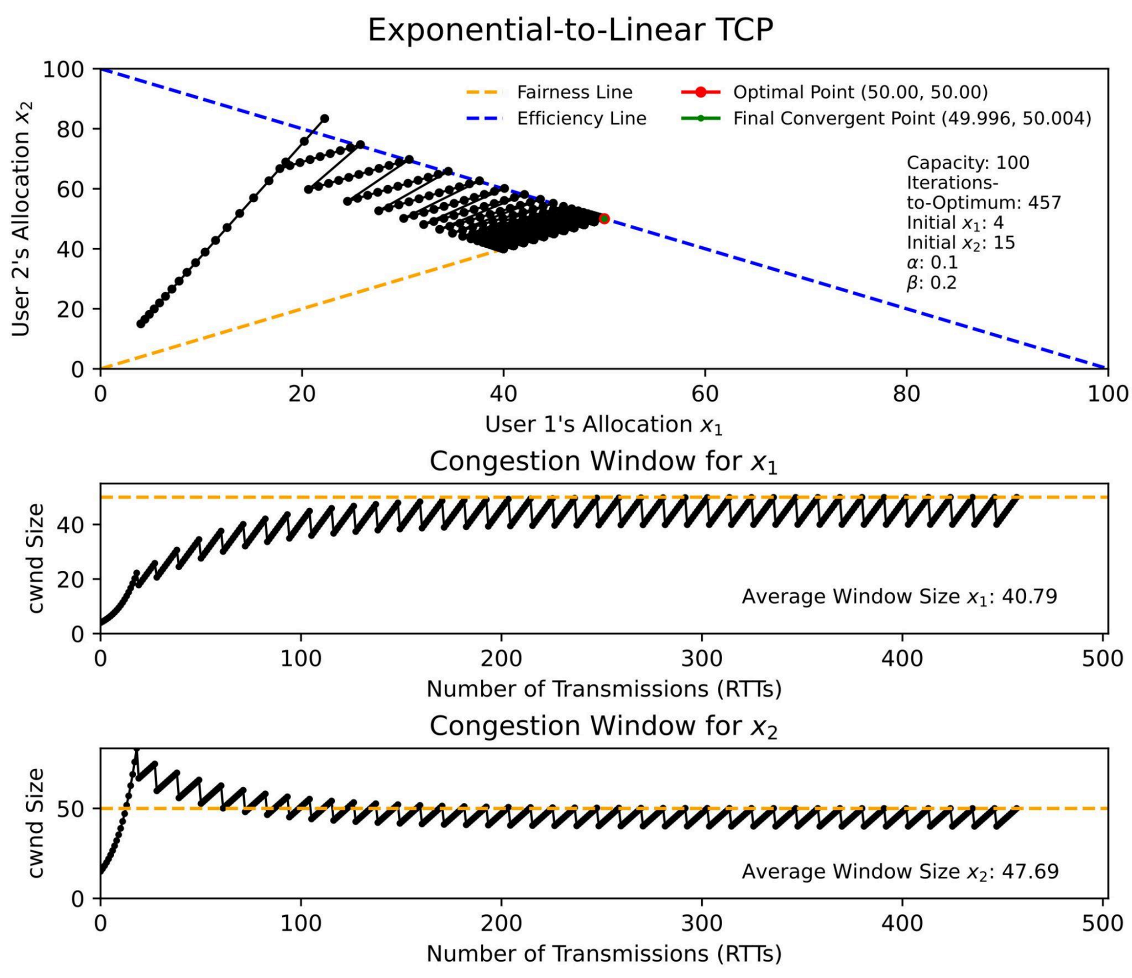
Next we explore a TCP algorithm that I found during my experimentation. These are its phases :

- Exponential Increase : $cwnd = cwnd + cwnd * \alpha$
- Multiplicative Decrease : $cwnd = cwnd * (1 - \beta)$

However, there is unique addition to this, where once the system encounters its first instance of congestion, a flag is set, causing the increase phase to shift to this equation :

- Additive Increase : $cwnd = cwnd + 1$

With my experimentation, I also found the parameters $\alpha = 0.1,\ \beta = 0.2$ to be optimal, while keeping the same starting points as the previous algorithm at $(4,\ 15)$.



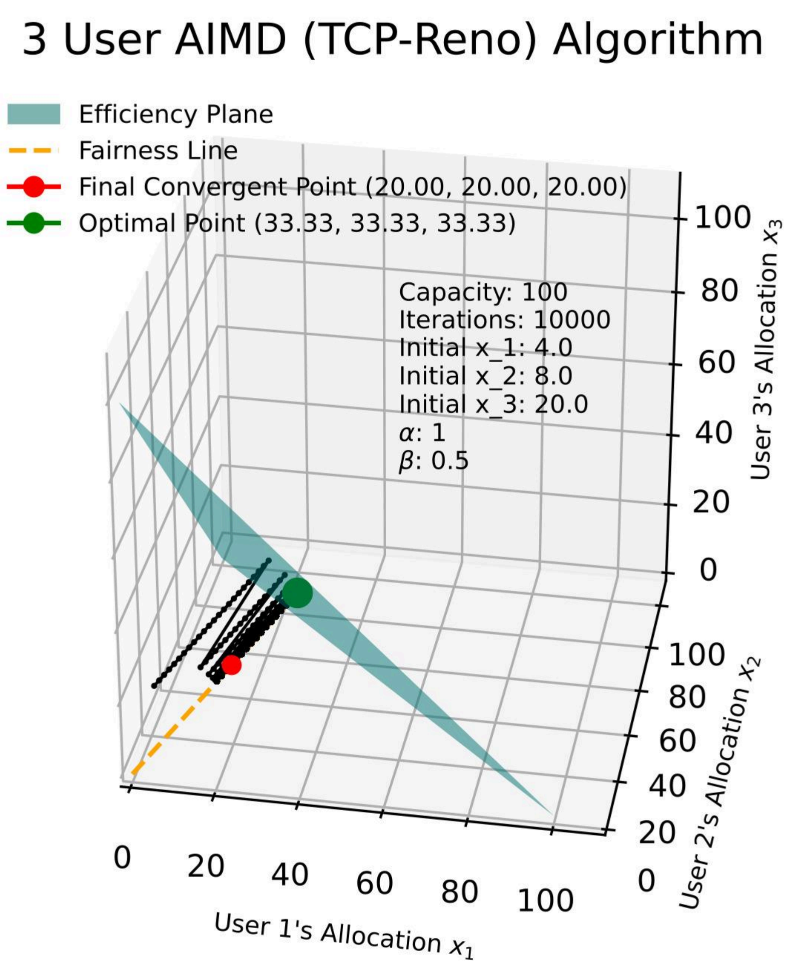
Similar to TCP-Reno, Exponential-to-Linear TCP could converge to the optimal point in 457 iterations. However, we see that the initial increase to congestion is much faster and more steep as compared to TCP-Reno due to the initial exponential increase.

The graphs of the individual users also show that the window size remains closer to the optimal line of 50 throughout the experiment, resulting in higher average window sizes, and thus a higher Bandwidth Utilization of 88.48%. This is a result of the values of the α, β values, where the smaller β leads to a smaller multiplicative decrease, leading to a faster recovery to maximum throughput.

This algorithm works especially well in data centers which contain window sizes much larger than 100, and usually measure in Gigabytes. The initial exponential increase ensures that the system approaches the threshold quickly while the small decrease ensures consistent high speed networking with low latency.

4. Three User AIMD

Next, we will tackle the case where there are 3 users that share a single bottleneck, as opposed to 2 users that have seen before. We will be using the TCP-Reno algorithm with values $\alpha = 1, \beta = 0.5$, for its parameters, while keeping capacity the same.



We can observe that the 3D model for 3 users is very similar to the 2D graph in the section of TCP-Reno where they both show the same back-and-forth behaviour. However, the main difference lies in the 2D efficiency plane, which differs from the 1D efficiency line on the 2D graphs in previous experiments.

Additionally, we can observe the system did not converge even after 10,000 iterations, but while there is no visual proof, it would be false to conclude that the model never converges. I have instead attached the analytical proof in Appendix A that proves that the TCP-Reno algorithm dynamics converge for 3 and even N users, using $\alpha = 1, \beta = 0.5$ as the parameters.

5. Conclusion

This report has looked at some ideas of what future TCP algorithms could look like based on what current trends indicate about the state of future data centers. It also looked at the experimental analysis of regular TCP-Reno and Exponential-to-Linear TCP and their effectiveness in a data center environment. While I only used Bandwidth Utilization for comparing between the algorithms, in reality there are many more factors that go into selecting an algorithm, such as Fairness Index, Retransmission rates etc. However, I believe that even here, AI can be utilised to look at many different factors at once to screen for the best algorithm according to a data center’s specifications.

Plus, the report has also shown that AI can be used to engineer a TCP ex machina for data centers by showing the ongoing efforts to leverage AI in data centers, and referencing algorithms that leverage AI which are already in use, such as the previously mentioned PowerTCP, and Remy (Winstein & Balakrishnan, 2013), which uses AI to formulate specific congestion-control algorithms at different endpoints, and importantly has already shown to outperform human-designed algorithms like TCP Cubic, TCP Vegas etc.

With the growing emphasis on AI, and its widespread adoption, I believe that AI-powered algorithms will become the industry standard. I also believe businesses will be pressured to incorporate AI into their products to remain competitive in an increasingly AI-driven landscape.

References

1. Addanki, V., Michel, O., & Schmid, S. (2021). PowerTCP: Pushing the performance limits of datacenter networks. *arXiv (Cornell University)*. https://doi.org/10.48550/arxiv.2112.14309
2. Alizadeh, M., Greenberg, A., Maltz, D. A., Padhye, J., Patel, P., Prabhakar, B., Sengupta, S., & Sridharan, M. (2010). Data center TCP (DCTCP). *ACM SIGCOMM Computer Communication Review*, *40*(4), 63–74. https://doi.org/10.1145/1851275.1851192
3. Archiveddocs. (2016, August 31). *Data Center Transmission Control Protocol (DCTCP)*. Microsoft Learn. https://learn.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2012-r2-and-2012/hh997028(v=ws.11)
4. G, N. (2024, March 14). *TCP Protocol: Understanding what transmission control protocol is and how it works*. Hostinger Tutorials. https://www.hostinger.com/tutorials/tcp-protocol#Why_TCP_Is_Important
5. Lee, V., Seshadri, P., O'Neill, C., Choudhary, A., & Deutscher, S. A. (2025, January 20). *Breaking Barriers to Data Center Growth*. bcg.com. https://www.bcg.com/publications/2025/breaking-barriers-data-center-growth#:~:text=The%20growing%20number%20and%20size,the%20years%202020%20through%202023.
6. Liu, J. (n.d.). *An adaptive and interpretable congestion control service based on Multi-Objective Reinforcement Learning*. IEEE Journals & Magazine | IEEE Xplore. https://ieeexplore.ieee.org/document/10798476/authors#authors

7. Pentikousis, K., Badr, H., & Andrade, A. (2010). A COMPARATIVE STUDY OF AGGREGATE TCP RETRANSMISSION RATES. *International Journal of Computers and Applications*, *32*(4). https://doi.org/10.2316/journal.202.2010.4.202-2660
8. *The Impact of Artificial intelligence on data centers: A Comprehensive analysis*. (n.d.). https://www.digitalrealty.com/resources/articles/data-center-ai
9. *TOP 10 Data center trends in 2025: A Researcher Perspective – CAP @ NTU*. (n.d.). https://www.ntu-cap.org/2025/01/top-10-data-center-trends-in-2025-a-researcher-perspective/
10. Winstein, K., & Balakrishnan, H. (2013). TCP ex machina. *ACM SIGCOMM Computer Communication Review*, *43*(4), 123–134. https://doi.org/10.1145/2534169.2486020
11. Zhu, Y., Eran, H., Firestone, D., Guo, C., Lipshteyn, M., Liron, Y., Padhye, J., Raindel, S., Yahia, M. H., & Zhang, M. (2015). Congestion control for Large-Scale RDMA deployments. *ACM SIGCOMM Computer Communication Review*, *45*(4), 523–536. https://doi.org/10.1145/2829988.2787484

Appendix

1. Mathematical Proof of Convergence for 3 Users using AIMD (TCP-Reno) : https://github.com/xGokull/SC4052/blob/main/Assignment%201/3UsersProof_AIMD.pdf

2. Jupyter Notebook file with Code : https://github.com/xGokull/SC4052/blob/main/Assignment%201/SC4052%20Gokul%20Assignment%201.ipynb