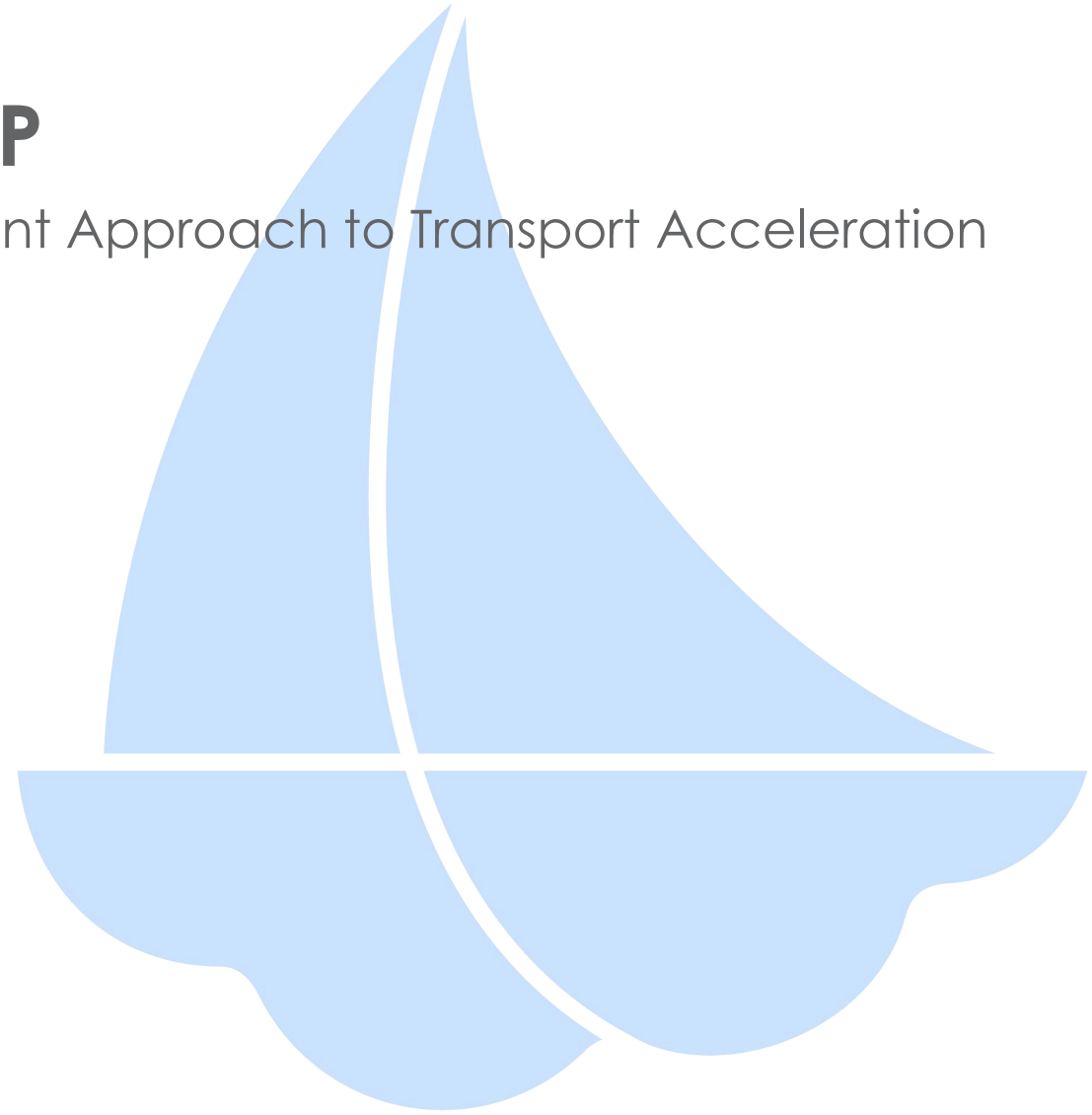
轻网科技
LightWAN

ZetaTCP

An Intelligent Approach to Transport Acceleration



Abstract

Transmission Control Protocol (TCP) was developed nearly four decades ago when the primary users of the Internet were DARPA and universities. Despite being used to deliver more than 90% of all Internet traffic, TCP has actually become a key bottleneck to today's fast web performance.

TCP's primary goal is to ensure that every packet gets delivered, and that it gets delivered in the correct sequence. To accomplish this, a limited number of packets are transmitted at a time. These packets must be acknowledged before additional packets can be sent. Over time, the queue of packets waiting to be sent grows. However, standard TCP is governed by packet loss, so it actually depends on dropped packets to know when to slow down and eventually find the best speed. This limited flow control mechanism results in inefficient bandwidth utilization a poor end user experience on today's Internet.

While network bandwidth has greatly increased, application delivery demands and user expectations have also greatly increased. There are now a growing number of TCP optimizers that use various techniques to overcome network latency caused by the inherent inefficiencies of TCP. This whitepaper reviews some of these techniques and evaluates the pros and cons of each approach. We will also introduce LightWAN ZetaTCP technology, a next-generation approach to overcoming transport latency.

TCP Optimization

To date, there are three generations of TCP optimization:

- **Loss-based** – uses loss to govern speed
- **Delay-based** – uses queuing delay instead of loss to govern speed
- **Learning-based** – uses analytics and learned session observations to govern speed in real-time

Loss Based - Since network pipes are so much fatter today, most TCP optimizers use techniques to increase the initial congestion window size (CWND) and also seek to provide better congestion management and loss recovery. The problem with this approach is that even with these tweaks, TCP transport speed continues to be governed by loss, which results in inefficient throughput and an uneven flow. In addition, these optimized TCP stacks are still static by nature - the settings stay the same regardless of network conditions.

This "one-size-fits-all" optimization approach can no longer keep up with current web performance demands. Furthermore, treating loss as a signal of congestion can adversely worsen throughput, especially when packet loss are caused by non-congestion factors such as a weak mobile signal. Lastly, with the deep buffers that are contained in modern networking equipment, the possibility exists where congestion is masked until the buffers are full. In this case, waiting to slow down until loss actually happens can cause even greater packet loss and make the situation much worse than necessary.

Examples: TCP Reno and Cubic, both are standard TCP stacks in most Linux kernels.

Delay Based - An alternative approach is to govern TCP transport speed by queuing delay instead of loss. With this approach, congestion levels are proactively monitored based on the changing of round trip time (RTT), with the goal of preventing unnecessary rate decrease and avoiding loss altogether. In general, the delay-based approach fits today's networks better than a loss-based approach. It maximizes the average amount of data that can be sent over a single round trip time (RTT). In addition, it maximizes throughput in a high, steady flow, which can have a positive effect on maintaining a high bitrate for video streaming and reducing page load time for web applications.

While this technique is an improvement over loss-based TCP optimization, it can be disrupted by small queues in the data path or by frequent latency changes, which is typical with mobile networks. In addition, delay-based TCP optimization is mostly static in nature, with very limited ability to perform dynamic adjustments.

Example: FastTCP.

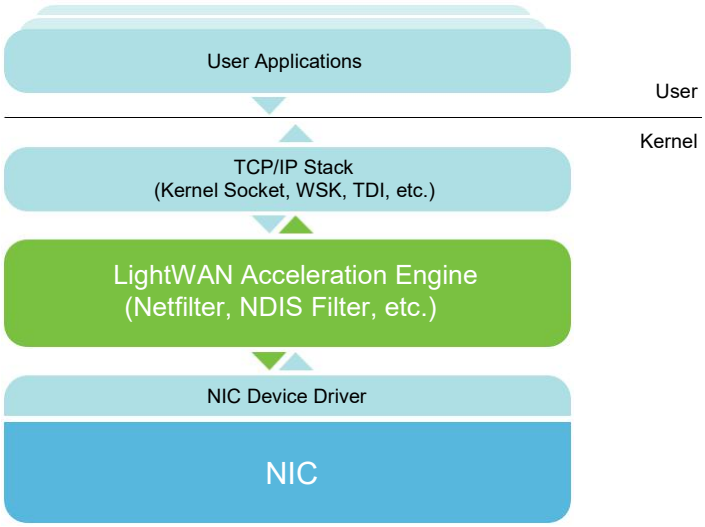
Learning Based - The next evolution of TCP improvement takes a learning-based approach, where the characteristics of every TCP flow are observed and analyzed in real-time in order to adaptively make intelligent in-session adjustments. In essence, each and every session is uniquely optimized for maximum performance. The learning-based approach overcomes the limitations of the static approaches described previously. It observes the transmission of the TCP flow to judge the true level of congestion intelligently and accurately. As a result, both congestion detection and handling as well as loss recovery is dealt with far more efficiently.

The next section provides detail on ZetaTCP, the learning-based approach developed by LightWAN.

ZetaTCP

LightWAN began development of ZetaTCP, an advanced learning-based approach to TCP acceleration. ZetaTCP was developed with the following design goals:

Figure 1. LightWAN Accieleration Engine



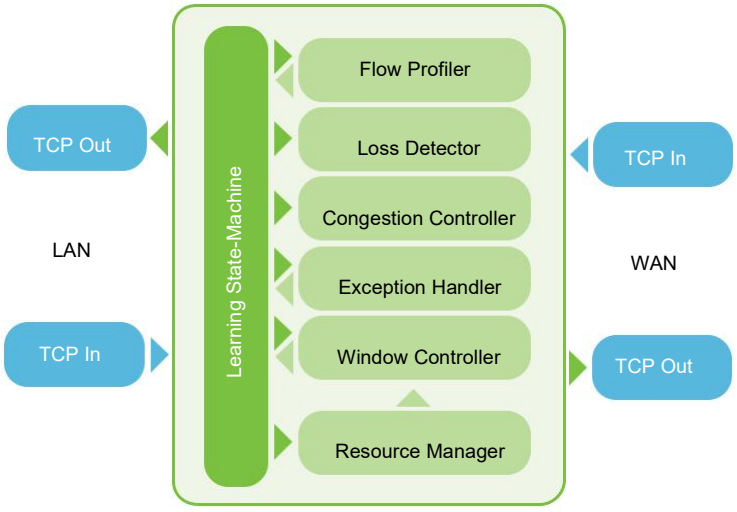
Transparency: Designed to be completely compatible with standard TCP stacks without replacing them, ZetaTCP boosts TCP performance and performs all the optimization transparently. A ZetaTCP installation takes in the TCP flows, “super-charging” and pumping them out without the receiving end even being aware of its existence. ZetaTCP can be embedded into client devices such as smart phones, tablets, and laptops, or installed on servers, load balancers, or any caching devices. ZetaTCP sits between the standard TCP stack and network interface driver as shown in Figure 1. All applications continue to communicate with standard TCP stacks. ZetaTCP is completely transparent to all applications and network devices in any deployment or embedded scenario.

Learning-based: ZetaTCP has a complete set of patented algorithms that study and learn the characteristics of the TCP flow patterns on-the-fly in order to make accurate judgments and well-informed decisions. This learned intelligence increases the TCP transport performance to levels previously unattainable.

Scalability: ZetaTCP can easily scale up or down. On the high-end, ZetaTCP can be built into industrial grade network equipment, such as routers, gateways and application delivery controllers, supporting multiple CPUs/cores and boosting millions of TCP flows with tens of gigabits of throughput. On the low-end, it can be embedded into consumer devices, such as smartphones, tablets, laptops or wireless routers, requiring only a few megabytes of memory.

These design goals ensure that ZetaTCP is the most advanced and adaptable TCP acceleration solution capable of delivering unparalleled network performance.

Figure 2. ZetaTCP Engine



How It Works

ZetaTCP engine houses most of the optimization intelligence and control functions. Packets are queued at the TCP in/out modules while most of the logic is focused on the WAN-side processing.

ZetaTCP consists of the following major components:

Learning state-machine: The intelligence hub of ZetaTCP. Accumulates knowledge about the network path and enables real-time session-specific decisions, such as how fast to transmit data and when to re-transmit data.

Flow profiler: Extracts and learns the characteristics of each TCP flow, records and maintains the learning state-machine.

Loss detector: Watches for packet loss and decides the most likely reason for the loss based on the learning state-machine, e.g., whether the loss was caused by a simple random drop or by network congestion.

Congestion controller: Executes the core congestion control logic based on the learning state-machine.

Exception handler: Instead of directly accelerating the TCP flows, the exception handler uses the intelligence from the learning state-machine to detect the flaws in peer TCP stacks or certain devices along the data path. There are erroneous TCP implementations in real-world deployments. This module is added to detect these abnormal TCP implementations to ensure maximum acceleration. Its built-in intelligence also contributes to the learning state-machine.

Window controller: Computes the TCP advertised window size, which controls the incoming packets from both the LAN and the WAN sides, and balances between the level of acceleration and system resources.

Resource manager: Tracks and controls system resources, including memory and computation power, and dynamically balances the consumption of system resources among all active TCP flows, which is the key to ZetaTCP's fairness and scalability to millions of flows. The resource management also takes input from the learning state-machine.

ZetaTCP's learning-based algorithms primarily improve the TCP transport in the following ways:

Improved Congestion Detection and Handling

Standard TCP's congestion avoidance is based on an algorithm that was designed decades ago. The algorithm starts with the assumption that all loss is a signal of congestion. This assumption oversimplifies what is really happening in the network and forces standard TCP to take the most conservative strategy to deal with packet loss. In reality, most packet losses in today's networks is not caused by congestion. This is especially true in wireless/ mobile networks, where fading channels introduce inevitable packet loss. Standard TCP does not perform well in such environments.

The delay-based approach (see page 1), on the other hand, uses increased latency as an indication of congestion, which performs better than standard TCP's loss-based approach where packet loss is caused by non-congestion factors. But again, this is still a static approach and does not fit the constantly changing network scenarios.

For example, when a shallow queue on the network path becomes overwhelmed, the delay-based approach does not detect the congestion because latency does not build up and it does not back off the sending rate. This leads to massive packet loss and can take a very long time to recover, which blocks the TCP sliding window and causes low throughput. Another scenario is when a network device on the path introduces occasional variable packet-processing delay. This is considered to be congestion by the delay-based approach and can cause unnecessary slowdowns, leading to lower throughput.

To overcome the shortcomings of the above static congestion control mechanisms, LightWAN ZetaTCP introduces a number of intelligent algorithms that are built into its learning state-machine to track and categorize the congestion scenarios based on real-time traffic statistics. The dynamic learning happens on a per TCP connection basis. ZetaTCP takes many factors into consideration for congestion detection, such as:

- **RTT**: Round Trip Time
- **ACK**: Interval between Acknowledgements
- **SACKs**: Severity of packet disorder reflected by duplicated ACKs and selective ACKs

The intelligence regarding the specific network path of the TCP connection builds up during the transmission, and the congestion can be detected more accurately and promptly. As a result, ZetaTCP reacts to congestion more effectively to avoid massive packet loss. The intelligence built up in the learning state-machine is also used to control the transmission after the congestion. This allows a smoother recovery and a more efficient usage of the available bandwidth on the network path.

Figure 3. TCP Optimization Comparisons

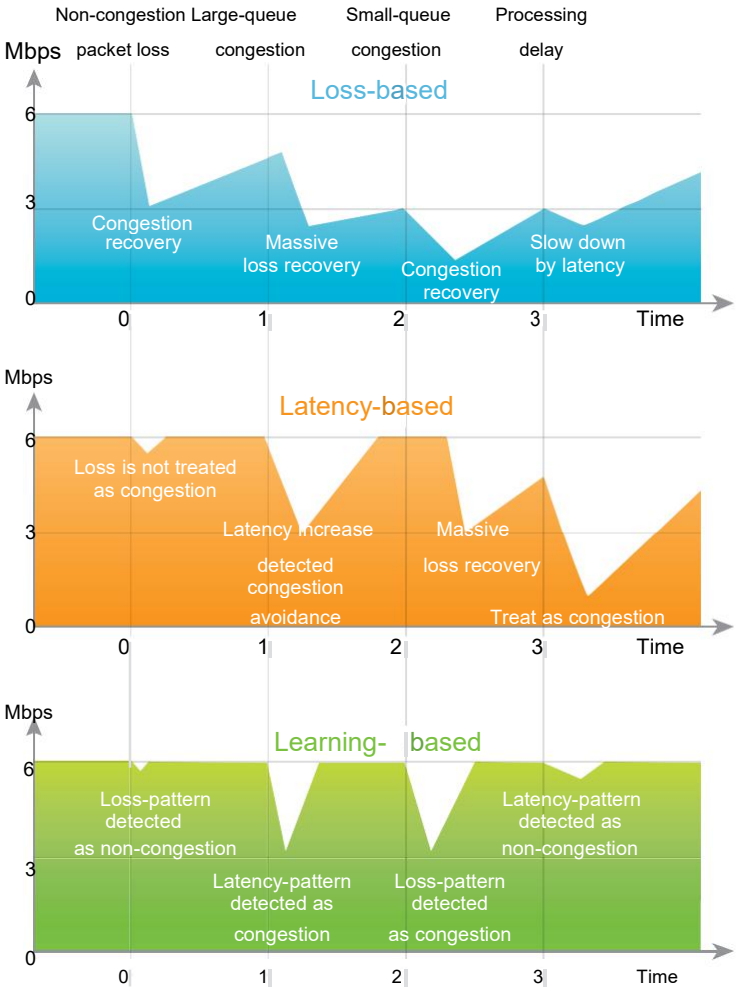


Figure 3 compares the reactions to congestion detection and handling of loss-based, delay-based and learning-based approaches under four typical network path scenarios: non-congestion packet loss, large-queue congestion, small queue congestion and processing delay.

From the charts, though the delay-based approach improves throughput in the first two scenarios compared to the loss-based approach, it fails to react to congestion on the small queue in the third scenario. In the fourth scenario, it mistakenly treats processing delay in some devices on network path as congestion. This leads to significant throughput loss. ZetaTCP is able to detect congestion correctly in all four scenarios and recover from loss and latency increases quickly, reaching a much higher average throughput compared to the first two approaches.

The ZetaTCP learning based approach is also effective in scenarios other than the four shown above.

Accurate and Predictive Loss-Detection with Rapid Loss Recovery

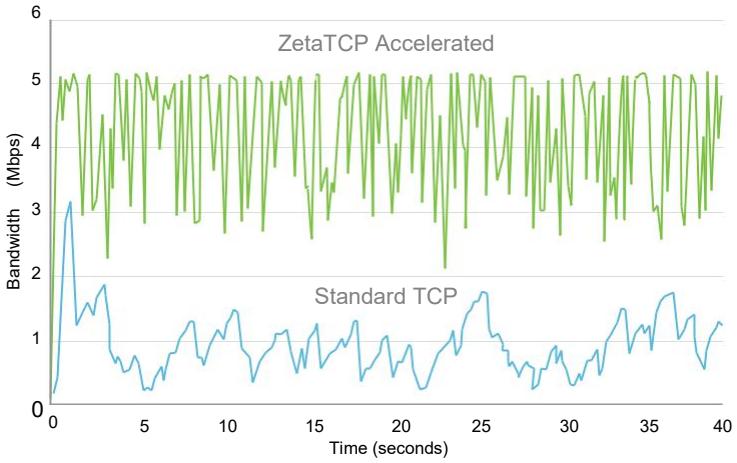
Packet loss is rarely distributed evenly across the duration of the transmission. Rather, loss tends to happen in a bursty manner. It is very common for more loss to happen in the fast recovery region and/or during the recovery.

Standard TCP, which is primarily based on RFC 2582 and SACK-supported algorithms such as RFC 3517, is rather mechanized and often fails to identify individual packet loss among clustered loss. This results in failure to retransmit lost packets promptly, adding more round trips to the recovery cycle.

Another scenario is retransmission timeout (RTO). The last piece of logic that the standard TCP wants to invoke is RTO retransmission. When this happens, standard TCP will back off the transmission to the point of the last acknowledgement (ACK). RTOs are not usually invoked except under extreme network conditions such as heavy congestion, loss of connectivity, etc. Falling back to retransmit everything that has just been transmitted usually makes the situation even worse.

ZetaTCP has introduced advanced intelligence to loss detection. It is able to detect packet loss more accurately and rapidly than standard TCP or delay-based implementations, especially under heavy loss scenarios. Similar to congestion detection, the intelligence for loss detection is built up through dynamic learning on a per-TCP connection basis. The learning state-machine tracks and categorizes the packet loss scenarios based on traffic statistics. Each packet loss instance further educates the learning state-machine to detect future packet loss more accurately and rapidly. As a result, ZetaTCP is able to recover from packet loss faster and keep the TCP sliding window moving more smoothly, which results in
higher throughput.

Figure 4. Real Data Comparison Test



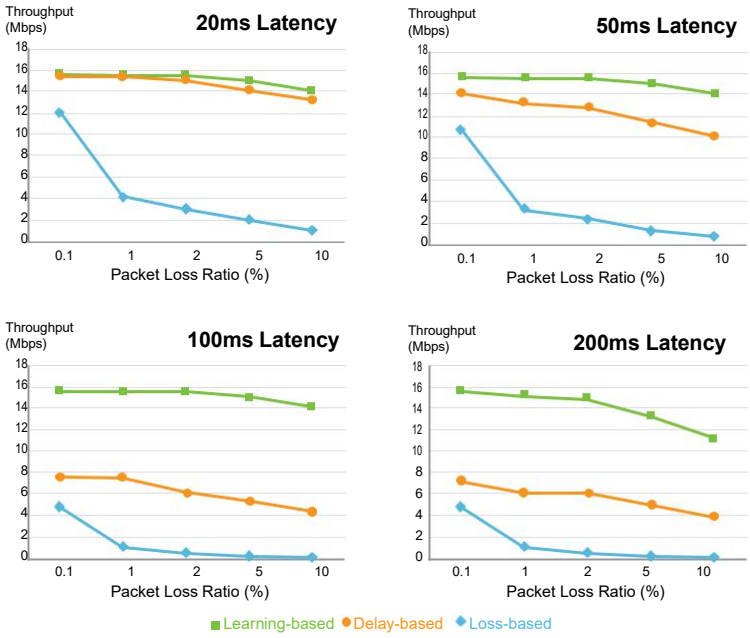
Leveraging this intelligence built up through the learning state-machine, ZetaTCP also helps to avoid RTO by first attempting to determine which packets are likely missing and re-transmitting just those packets. Even if ZetaTCP has to invoke RTO, it doesn't simply fall back to the last ACK and retransmit everything again, which is extremely inefficient and causes increased latency. Instead, it calculates a starting point and reitransmits only those packets that are likely missing. This mechanism not only allows a much faster recovery from massive packet loss, but also generates significantly less retransmission traffic than standard TCP or delay-based TCP, thus greatly increasing bandwidth efficiency.

Figure 4 is generated from real test data of 100-millisecond RTT and 1% average loss rate, with a max receive window size 65535. This limits the bandwidth ceiling to about 5.24Mbps.

Standard TCP does not recover from packet loss fast enough, leading to a much lower speed. ZetaTCP acceleration, on the other hand, detects and recovers packet loss quickly, so that the max bandwidth capacity is reached even under significant packet loss and round trip latency.

ZetaTCP's accurate and rapid detection of packet loss capability is especially valuable with the explosive growth of mobile networks. The flaky last mile fading channel to the mobile devices creates frequent bulk packet loss. Such loss triggers standard and delay-based TCP to jam more packets into the network, which causes more problems. ZetaTCP, however, analyzes the situation intelligently, recovers from packet loss rapidly and efficiently, and enables a smoother transmission and maximum throughput.

Figure 5. Third Party Comparison Tests



Third party performance comparison tests

The diagrams in Figure 5 compares the performance of loss-based, delay-based and learning-based approaches. The tests were performed by a third party. Some typical network scenarios under a max bandwidth of 16Mbps were simulated in a lab environment. Under those different network environments, the throughput of a single TCP connection was measured. Each sample was the average throughput of the TCP connection with multiple test runs. New cubic TCP, which is used in most Linux distributions today, was chosen for the loss-based approach. A widely used delay-based TCP implementation was chosen for delay-based approach. LightWAN ZetaTCP was chosen for the learning-based approach.

The results clearly indicate that the loss-based approach performed poorly under all of these long latency and lossy environments, while the performance of the delay-based approach was significantly better than that of the loss-based approach. The learning-based approach provided the highest through put under all environments in the test.

Conclusion

While TCP is relied upon to carry more than 90% of all Internet traffic, it has become a key performance bottleneck in keeping up with today's demands to effectively deliver latency-sensitive web applications to increasingly impatient end-users. In order to keep up with these demands, a number of TCP optimization approaches have been developed to provide the boost needed. Applying optimization techniques to standard, loss-based TCP provides some improvement. But as long as network speed is governed by loss, a high data-rate and stable throughput will be impossible to achieve.

A more modern delay-based approach provides some fundamental improvement to dealing with network latency, but for the most part, it remains a static, one-size-fits-all approach. In order to conquer the inefficiencies of the previous approaches, an advanced learning-based approach is needed that is capable of applying session-specific transport optimizations on-the-fly.

LightWAN ZetaTCP is the world's most advanced learning-based TCP acceleration engine. It is already used by hundreds of companies and millions of users to accelerate their latency-sensitive applications. ZetaTCP is able to provide these capabilities in a completely transparent manner while remaining fully compatible with existing TCP stacks.

轻网科技
— LightWAN —