Measurement of Packet Train Arrival Conditions in High Latency Networks

'Etuate Cocker, Ulrich Speidel
Department of Computer Science, The University of Auckland, Private Bag 92019, Auckland, New Zealand
Email: ecoc005@aucklanduni.ac.nz, ulrich@cs.auckland.ac.nz

N. Rebenich[1], S. Neville[1], A. Gulliver[1], R. Eimann[2],
K. Nisar[3], S. Hassan[3], Z. Aziz[3], M.-C. Dong[4], V. Wong[4]
[1] University of Victoria, Canada, [2] in private capacity, Zurich, [3] Universiti Utara Malaysia, [4] University of Macau

***Abstract*—Real-time Internet applications such as telephony, video conferencing and remote control are increasing in importance. A critical requirement for such applications is the ability to receive data packets in correct order with minimal delay (latency) and loss of data. Most Internet Service Providers (ISP) try to achieve this by adding bandwidth in the form of additional infrastructure (links and routers) and load balancing to meet the continuous Internet traffic growth. For real-time protocols, such upgrades are not exclusively beneficial, however. They tend to increase the number of routers (and hence router queues) a packet has to pass through, and increase the potential for out-of-order delivery of packets. Our paper presents the baseline results of a longitudinal study investigating the effects of such infrastructure changes on international real-time traffic.**

I. Introduction

In recent years, Voice over IP (VoIP) protocols (e.g., Skype) and other real-time applications have revolutionised Internet use and pointed out the potential for others, such as video conferencing, remote control, monitoring, and manipulation in areas such as education, telemedicine, engineering, and so on.

For smooth operation, real-time data must be delivered over the Internet with as little delay and loss as possible. Many applications cannot afford the time required for retransmission of lost data. A packet traveling through the Internet may be delayed by long propagation distances, but also in the queues of congested routers. These delays are collectively known as latency. Other undesirable effects include:

- packet losses and queue-induced jitter at congested routers,
- jitter and out-of-order delivery as a result of load balancing creating multiple paths between source and destination. Reordering packets of real-time applications at the receiver requires buffering, which implies a further delay, and
- undesirable traffic such as malicious traffic and retransmissions after packet losses consuming bandwidth.

The traditional strategy of ISPs in dealing with such issues is to upgrade networks by adding alternative links in and out of their networks, replacing or augmenting long-latency satellite links by shorter fibre optic cables, and adding faster routers, and in some cases load balancing.

One result of such strategies is an Internet resembling an increasingly dense physical mesh of links, often with multiple physical paths between a source and a destination host. Packets do not necessarily travel along the physically shortest, least congested, or even a consistent path. Also, the number of routers along paths increases, meaning that even if individual router queues become shorter, packets now encounter more queues than before. The net effect of such measures on global latency and in-order delivery of packets is thus not necessarily that obvious. Even now, it can often be quite surprising to see which geographical route packets actually take.

Our research has its genesis in the connectivity situation in Pacific Island countries (PICs), mostly small, remote and isolated islands with small population and skill base, but often a large overseas diaspora. The benefits of Internet connectivity and in particular real-time protocols to PICs are obvious: Applications such as Skype let those in the islands stay in close touch with family and locally unavailable expert skills overseas.

Poverty and slow technological development [1] mean low Internet penetration [2]. This restricts many PIC ISPs to congested low bandwidth / high latency satellite links for international connectivity. In the islands, old and unreliable technology dominates local networks [3]. Combined with the small skill base, this acts as a further barrier to uptake while limiting practically achievable maintenance and reducing protection against malicious traffic. This in turn aggravates congestion. Together, these factors often prevent widespread uptake of real-time protocols. `ping` probes from New Zealand, where most overseas Pacific Islanders live, return round-trip times over 800 ms for many islands. Real-time communication with Skype is almost impossible.

Upgrades of the kind discussed above do happen in PICs, but are often slow to arrive and tend to use surplus equipment (such as the recently laid fibre-optic link between Samoa and Hawaii). New links tend to yield radically different latencies. The Pacific is therefore an ideal region to study the question as to whether network changes can influence the Internet's ongoing ability to deliver real-time traffic, and whether we can draw any conclusions about the long-term feasibility of

978-1-4799-0434-1/13/$31.00 ©2013 IEEE

real-time applications over a best-effort network?

For this purpose, we embarked on a longitudinal study in 2012 to monitor long-term trends in the condition in which a real-time packet stream (such as a VoIP call) arrives at a remote receiver after passing through a real network over a long distance ("arrival condition"). The study conducts active measurements by exchanging simulated real-time traffic between endpoints, collecting timing and other arrival information for the associated packets. Rather than settling on a method for analysing the data from the outset, we wanted to retain as much information on the packet propagation as we could in order to permit retrospective analysis at a later date.

For any conclusions to be transferable to the Internet in general, it was also important not to restrict the study to a particular pair of endpoints, or to the Pacific. In cooperation with numerous colleagues across the world, the study establishes a network of such endpoints ("beacons") between which the experiments of this study are conducted.

In this paper, we first discuss what we want to be able to measure and any resulting requirements for tools. We then review related work on Internet traffic measurement tools and describe the challenges that led to the development of our own software, before proposing means of evaluation and discussing some of the early observations made.

A. Active measurement of arrival condition

The arrival condition of a packet train is determined in part by the available bandwidth along the forwarding path(s), but also by the amount of jitter, packet loss and packet reordering introduced along the way. This required recording of timing information for packets, but also a mechanism to be able to detect packet loss and reordering. We were also interested in concrete evidence of path changes, which can be detected at least in some cases through changes in the TTL field in the IP headers of the arriving packets.

To be able to deploy a large number of beacons, the solution had to be flexible and scalable in terms of cost. Last but not least, as many of our beacons are deployed in locations where bandwidth is at an extreme premium, any measurement tool to be used thus had to limit its network load. Finally, we needed per-packet data logging in order to permit retrospective analysis.

Existing active measurement utilities tend to concentrate on the bandwith aspect: IPERF [4] measures throughput, packet loss and jitter, but provides no per-packet logging facility and can load the network quite heavily. TOPP [5] estimates bandwith with a small number of probe packet pairs, which does not allow a ready conclusion as to the arrival condition of a sustained packet train. The PathChirp [6] network model assumes no packet reordering and, like TOPP, works with a small number of packets only. Pathload [7] is similar to PathChirp, but generates a higher network load. OWAMP [8] measures packet loss, jitter, delay, hop count (TTL) and out-of-order arrivals while presenting a low network load. However, none of these tools log per-packet data for later analysis. We thus opted to design our own beacon software.

In order to deploy beacons at multiple endpoints worldwide, we considered PlanetLab [9] as an existing platform. However, as a shared platform, use would not have been exclusive to our study. This could have led to a situation where packets could have arrived at a PlanetLab node busy with other operations, leaving them "un-timed" until our share of the node got its turn at processing. Consequently, this implied that we had to establish an entirely new network.

II. The Beacon Network

The beacon network is implemented on a growing network of computers (25 at the time of writing) running our beacon software. The beacons conduct their experiments in pairs, with one side acting as the "initiator" (beacon transmitting the first packet in an experiment) and the other as the "responder". One beacon usually pairs with several other beacons, but will only conduct a single experiment with a single partner beacon at any given time to avoid interference between experiments. Most beacon computers used are dedicated standalone machines to permit timestamps to be created as close to events as possible.

The most basic experiment, on which we report here, generates a series of 10,000 packets (packet train) similar to those generated by a three minute VoIP call using Skype over UDP [10], albeit with entirely synthetic data: Each packet identifies the experiment and experiment run that it belongs to and carries a serial number and a transmit timestamp, as well as some padding with contact information for the investigators. The initiator beacon transmits its packet stream to the responder beacon, which sends its own packet stream in return once a packet from the initiator is received. Packets are transmitted every 20 ms. The transmitting beacon logs the packet's serial number, the transmission timestamp in the packet, and the time at which the operating system's send method returned (thus giving a closer time estimate). The receiving beacon logs the serial number, the received packet count at the time that the packet was received, the transmit timestamp from the packet, a receive timestamp, and the value of the IP TTL field.

Most experiments are run three times a day for each beacon pair. The data logged allows us to measure the variations in latency and to compute various indicators of quality for the received packet train, such as jitter (standard deviation of packet inter-arrival time), packet loss, number of out-of-order arrivals, number of different hop counts seen, using the time-to-live field in IP header (TTL) as an indicator, and inter-arrival time entropies. By repeating the experiments over an indefinite period of time, we are able to plot long term trends in these observables.

Our beacon software is also capable of running experiments based on TCP connections where data is passed to the transmitting TCP socket either at a regular rate ("VoIP" over TCP scenario) or as fast as the socket is able to dispatch the

data (TCP "download" scenario). The authors will discuss these experiments in a future paper.

Beacons are identified by a two-letter country code and a number (e.g., the first New Zealand beacon is NZ1). When placing the beacons, Pacific Island countries were of particular interest, because their international connectivity often features long latencies coupled with small bandwidths and high levels of congestion.

III. A NOTE ABOUT CLOCKS

As discussed above, both endpoints contribute timestamps from their clocks to the experiment. This raises the question of clock synchronisation, which can be achieved either via a highly accurate clock at both beacons (atomic or GPS), or with less accuracy via network time protocol (NTP).

To exchange packets between a specific beacon pair, we execute and terminate beacon processes on the respective hosts at specified times. Beacons must thus be sufficiently synchronised so they do not miss each other's transmissions. NTP suffices for this purpose. An NTP client synchronises to an estimate of the actual time, based on the NTP response and an estimate of the return propagation time as half the client-server round trip time. Several of our sites experience NTP request RTTs of 100's of milliseconds and asymmetric request-response paths, causing offsets in the dozens of milliseconds. However, this is still sufficient as beacons start listening a short time before they transmit, and need to keep listening beyond the end of their own transmissions to account for overall latency and any late incoming packets.

Measuring total latency would require highly accurate synchronisation, but is not a primary observable in our case. Jitter is our only primary observable which includes time stamps from both ends. As long we can assume a constant offset between receiver and transmitter clocks, the offset appears in both individual packet latency and average latency, and thus cancels out in the jitter computation. The only remaining requirement is thus a small relative clock drift between beacons, a requirement easily met by standard quartz clocks.

IV. OBSERVATIONS

This section looks at a number of preliminary observations on a selection of beacon pairs, using our longest-serving beacons. Some graphs thus start later than others due to beacons being commissioned at different times. Also, network connectivity issues and hardware problems can cause prolonged outages and intermittent data quality issues. Our plots show these periods as diagonal ramps or, in some cases, as graphs terminating early.

A. Packet loss

Not unexpectedly, one tends to see more packet loss on crowded satellite links to islands than on long-distance fibre paths. Figure 1 shows packet loss data collected between New Zealand and Canada (fibre), and between Tuvalu and Canada (satellite). While the overall packet loss is higher over time

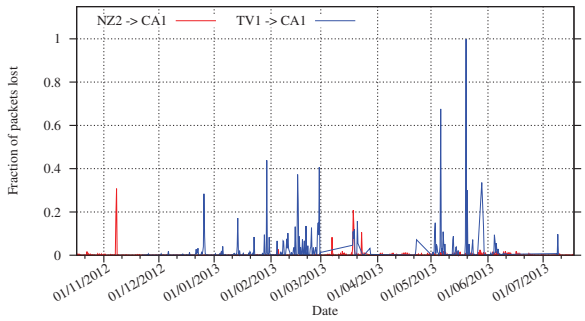

Fig. 1. Packet loss recorded at the University of Victoria (Canada) for UDP packet trains transmitted from the University of Auckland (New Zealand) and the Government of Tuvalu.

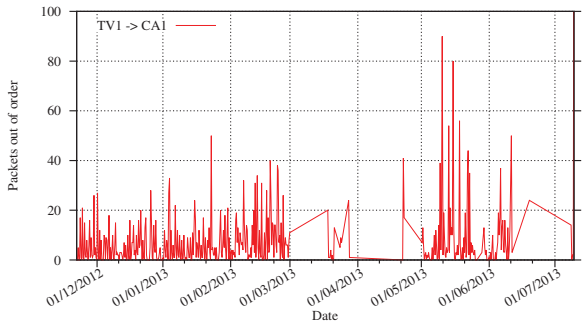

Fig. 2. Out-of-order arrivals at the University of Victoria from the Government of Tuvalu

for packets originating in Tuvalu, we note that packet loss from New Zealand can spike at similar values, albeit less frequently.

B. Out-of-order packet arrivals

An out-of-order packet arrival consists of a packet arriving ahead of another packet that was transmitted before it. Between many endpoint pairs, especially in the developed world, we have yet to not register out-of-order arrivals. However, they are a relatively common occurrence in Pacific Island traffic (see Fig. 2), albeit at a low level ($< 0.5\%$).

In our UDP experiments, packet latencies must differ by $\approx$20 ms or more for an out-of-order arrival to occur. Comparing out-of-order arrivals with packet loss, as shown in Fig. 3, we can see that these effects are not normally associated: Load balancing causes out of order arrivals but has a mitigating effect on downstream router load. Fig. 3 also shows a significant number of out-of-order arrivals between beacons TO1 and TO2. These beacons are physically only a few hundred metres apart, but as Tonga does not have a peering exchange, traffic between them leaves Tonga and thus incurs the latency differences observed.

C. Jitter

Jitter is not uniquely defined in the literature [11]. We use a "transit jitter" J_t, based on the following equation:

$$J_t = \frac{\sum_{i=0}^{n-1} \phi(i)|\sigma_i|}{n'}. \quad (1)$$

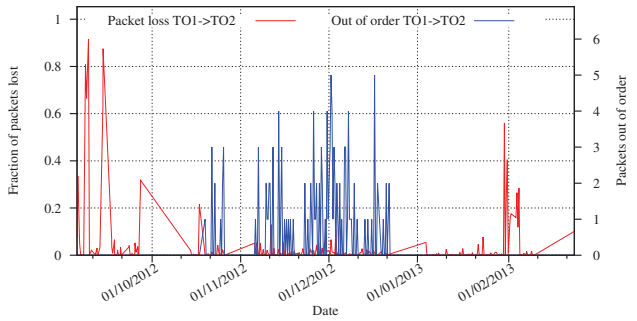

Fig. 3. Packet loss and out-of-order arrivals between the the Tongan beacons TO1 and TO2

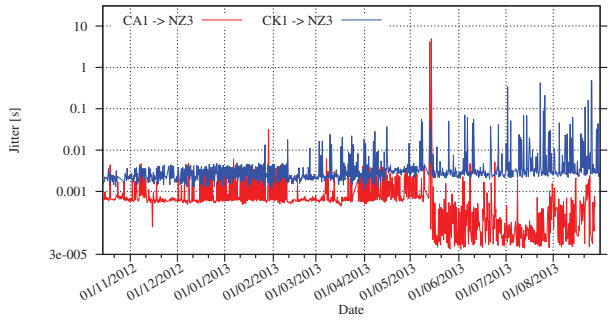

Fig. 4. Jitter between Canada and New Zealand (fibre) and the Cook Islands and New Zealand (satellite)

where n is the number of packets transmitted, n' the number of these that were received, $\phi(i) = 1$ if the i'th packet was received and $\phi(i) = 0$ otherwise. σ_i is given by

$$\sigma_i = r_i - t_i - \frac{1}{n'} \sum_{j=0}^{n-1} \phi(j)(r_j - t_j). \qquad (2)$$

where t_i and r_i are the times at which the i'th transmitted packet was transmitted / received, according to the respective beacon's clock. This definition has the advantage that it is invariant under constant offset between the beacon clocks. The jitter is thus a time value, and can be used to estimate the size of the buffer required at the receiver. Figure 4 shows the jitter between Canada and New Zealand and the Cook Islands and New Zealand. Overall, jitter from the Cook Island appears to be on the increase. The shape of the latency distribution associated with a jitter value is not necessarily clear, however. Queueing theory tells us that a packet stream passing through a series of router queues of varying length will have a latency variation distribution with a significant tail beyond the jitter value. In this case, a receive buffer will typically have to hold packets equivalent to a multiple of the jitter time seen.

However, queues are not the only possible cause for jitter. Consider, e.g., packets arriving via several different paths with different physical link latencies but little queueing-related delay on each path. In such cases, the packet latency distribution peaks around the individual path latencies but has no significant tail, meaning that actual delays seen are

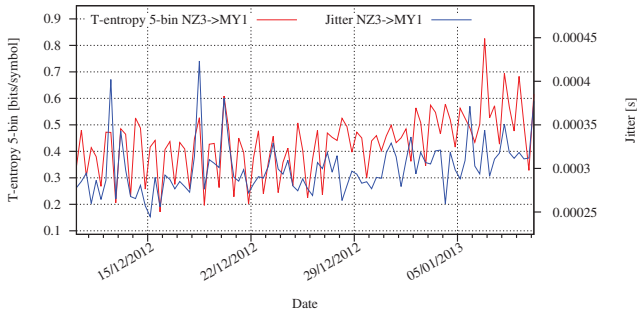

Fig. 5. Jitter and T-entropy between New Zealand and Malaysia. Note the daily cycles for both – but jitter and entropy do not always track each other.

not much larger than the jitter value itself. This "systematic" jitter thus allows us to restrict buffer sizes to the order of the jitter value itself. The next section discusses how we may distiguish between queue-induced and systematic jitter.

V. Symbolic Mappings and Entropy

Systematic jitter effects make packet arrivals more predictable than queue-induced delays as they introduce *patterns*. The amount of patterning can be assessed via a variety of techniques, and we propose the use of entropy / complexity estimators in this context.

The fundamental idea is this: Sort the interarrival times (or the latencies seen) into bins, as if to produce a histogram. Associate each bin with a symbol (such as the letters 'A', 'B', 'C' and so on), and concatenate the respective symbols for each interarrival time into a string, a technique known as *symbolic mapping*. For example, code interarrival times of less than 15 ms as 'A', 15-18 ms as 'B', 18-22 ms as 'C', 22-25 ms as 'D', and anything longer as 'E'. A "perfect" packet train with 20ms intertransmission times would thus result in a string "CCCCCC…". In a queue-delayed packet train, one would expect a more random assortment of all symbols, whereas a train with systematic jitter might look as regular as a "BDBDBDB…".

We can then put a figure on the degree of pattern repetition in the string by a variety of means: We can use a computable string complexity such as the Lempel-Ziv production complexity [12], an information measure such as the compressed length of the string after compression with a universal data compressor such as LZ77 [13] or LZ78 [14], or an entropy estimate such as the associated compression ratio. We use the T-entropy [15] in our examples here.

Figure 5 shows the T-entropy and jitter for traffic between beacon NZ3 at the University of Auckland and MY1 at Universiti Utara Malaysia. In this case, both exhibit approximately daily cycles, and in many cases jitter and entropy peak at the same time, indicating that the jitter is queue-induced. On the other hand, on some days, jitter peaks coincide with entropy troughs, indicating that in these cases, the interarrival times follow patterns and the jitter is predominantly systematic in nature. As can be seen in Figures 5 and 6, the T-

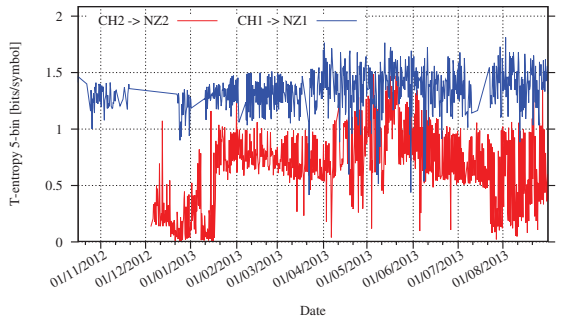

Fig. 6. T-entropy between two Swiss beacons and two of their New Zealand counterparts: Medium term fluctuations or long term trend?

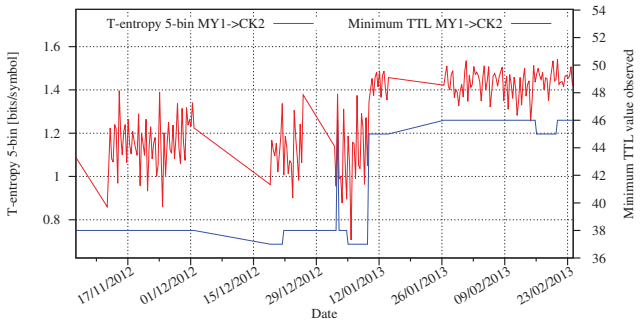

Fig. 7. T-entropy increases can reflect changes in routing (as evidenced by the time-to-live field). Here, the entropy rises despite the lower number of routers involved.

entropy of traffic can vary strongly and quickly with time – a common observation across our various beacon pairs. With such strong short-term fluctuations, any long-term trend in this category will take some time to determine. In some cases, e.g., as shown in Fig. 7, we can observe step-like changes in entropy, which clearly correlate with changes in the minimum time-to-live observed among the packets). However, such an association is not possible for all increases.

VI. Evidence of Route Diversity

Most of our UDP experiments to date have recorded only a single TTL value for each 10,000 packet train, indicating that all packets of the train may have travelled along the same route. However, a small number of experiments show more than one TTL value for some packet trains, proving that packets travelled via more than one route. This effect often occurs over several subsequent repetitions of the experiment, indicating that we do not merely observe route updates coinciding with the experiment, but actual concurrent use of multiple paths.

VII. Conclusion

The selection of early observations from our beacon network in this paper backs up much of the anecdotal evidence about Internet traffic in the Pacific and around the world. We see high packet losses across narrowband satellite links and in some cases out-of-order arrivals indicating either route diversity or extremely overloaded routers. Entropy analysis lets us distinguish between queue-induced and systematic jitter, and we can also report evidence for the occurrence of both. Our baseline data is extremely noisy, meaning that long-term observations will be needed to derive any global trend. Current work includes growing the number of beacons (19 at the time of writing, with three further beacons under construction), as well as interfaces for beacon data processing.

VIII. Acknowledgments

The authors would like to acknowlegde the assistance from the following organisations: Internet NZ, Pacific Island Partners (PIP), the Internet Society (ISOC), Ministry of Infrastructure and Planning (Cook Islands), Telecom Cook Islands, Telecom Fiji Ltd, Government of Kiribati, Government of Tuvalu, Ministry of Lands, Survey, and Natural Resources (Tonga), E.M. Jones Ltd. (Tonga), Ministry of Revenue (Tonga).

References

[1] Zwimpfer Communications Ltd: *Internet Infrastructure and e-governance in Pacific Islands countries, A survey on the development and use of the Internet*, United Nations Educational, Scientific and Cultural Organisation (UNESCO), Apia, 2002, pp. 1-118.

[2] D.H.R. Spennemann: *Digital divides in the Pacific Islands*, IT and Society, Vol 1, 2004, pp. 46-65.

[3] PIFS: *Pacific ICT survey report*, Pacific Islands Forum Secretariat(PIFS), Suva, 2002, pp. 1-14.

[4] O. Olvera-Irigoyen, A. Kortebi , L. Toutain and D. Ros: *Available bandwidth probing in hybrid home networks*, IEEE Workshop on Local and Metropolitan Area Networks (LANMAN), Chapel Hill, U.S.A, 13-14 Oct 2011. pp. 1-7.

[5] B. Melander, M. Bjorkman and P. Gunningberg: *A new end-to-end probing and analysis method for estimating bandwidth bottlenecks*, IEEE GLOBECOM'00, San Francisco, U.S.A, 27 Nov-01 Dec 2000. pp. 415-420.

[6] V. Fajardo, Y. Cheng , G. Parmar and Y. Ohba: *An efficient and loss tolerant method for measuring available bandwidth*, IEEE GLOBECOM 2009, Hawaii, U.S.A, 30 Nov-04 Dec 2009. pp. 1-6.

[7] M. Jain and C. Dovrolis: *End-to-end available bandwidth: measurement methodology, dynamics, and relation with TCP throughput*, IEEE/ACM Transactions on Networking, Vol 11, No. 4, 2003, pp. 537-549.

[8] S. Shalunov, B. Teitelbaum, A. Karp, J. Boote, and M. Zekauskas: *A one-way active measurement protocol (OWAMP).* Network Working Group – RFC, 4656 (2006), http://tools.ietf.org/html/rfc4656.

[9] B. Chun, D. Culler, T. Roscoe, A. Bavier, L. Peterson, M. Wawrzoniak, M. Bowman: *PlanetLab: an overlay testbed for broad-coverage services*, SIGCOMM Comput. Commun. Rev., Vol. 33, No. 3, 2003, pp. 3–12.

[10] Skype: *Skype SILK Datasheet*, https://developer.skype.com/resources/SILKDataSheet.pdf, last accessed: 31/8/2013.

[11] C. Demichelis, P. Chimento: *RFC3393: IP Packet Delay Variation Metric for IP Performance Metrics (IPPM)*, http://tools.ietf.org/html/rfc3393

[12] A. Lempel and J. Ziv, "On the complexity of finite sequences," *IEEE Trans. Inf. Theory*, vol. 22, no. 1, pp. 75–81, 1976.

[13] J. Ziv and A. Lempel: *A Universal Algorithm for Sequential Data Compression*, IEEE Trans. Inform. Theory, Vol 23, No. 3, May 1977, pp. 337-343.

[14] J. Ziv and A. Lempel: sl Compression of Individual Sequences via Variable-Rate Coding, IEEE Trans. Inform. Theory, Vol 24, No. 5, September 1978, pp. 530-536.

[15] M. R. Titchener, "Deterministic computation of string complexity, information and entropy," in *Proc. IEEE Int. Symp. on Inf. Theory*, Cambridge, MA, 1998, p. 326.