Infocommunications Journal

A PUBLICATION OF THE SCIENTIFIC ASSOCIATION FOR INFOCOMMUNICATIONS (HTE)

MARCH 2018 Volume X Number 1 ISSN 2061-2079

PAPERS FROM OPEN CALL

ADDITIONAL

Technically Co-Sponsored by

IEEE ComSoc™
IEEE Communications Society

hte

IEEE®
HUNGARY SECTION

Editorial Board

Indexing information

Infocommunications Journal is covered by Inspec, Compendex and Scopus.
**Infocommunications Journal is also included in the Thomson Reuters – Web of ScienceTM Core Collection,
Emerging Sources Citation Index (ESCI)**

Infocommunications Journal

Technically co-sponsored by IEEE Communications Society and IEEE Hungary Section

Supporters

FERENC VÁGUJHELYI – president, National Council for Telecommunications and Information Technology (NHIT)
GÁBOR MAGYAR – president, Scientific Association for Infocommunications (HTE)

Editorial Office (Subscription and Advertisements)**:**
Scientific Association for Infocommunications
H-1051 Budapest, Bajcsy-Zsilinszky str. 12, Room: 502
Phone: +36 1 353 1027
E-mail: info@hte.hu • Web: www.hte.hu

Articles can be sent also to the following address:
Budapest University of Technology and Economics
Department of Telecommunications and Media Informatics
Tel.: +36 1 463 1102, Fax: +36 1 463 1763
E-mail: vida@tmit.bme.hu

Subscription rates for foreign subscribers: 4 issues 10.000 HUF + postage

Publisher: PÉTER NAGY

HU ISSN 2061-2079 • Layout: PLAZMA DS • Printed by: FOM Media

www.infocommunications.hu

Time Synchronization Solution for FPGA-based Distributed Network Monitoring

Ferenc Nandor Janky and Pal Varga

***Abstract*—Distributed network monitoring solutions face various challenges with the increase of line speed, the extending variety of protocols, and new services with complex KPIs. This paper addresses one part of the first challenge: faster line speed necessitates time-stamping with higher granularity and higher precision than ever. Proper, system-wide time-stamping is inevitable for network monitoring and traffic analysis point of view. It is hard to find feasible time synchronization solutions for those systems that have nation-wide, physically distributed probes.**

Current networking equipment reside in server rooms, and have many legacy nodes. Access to GPS signal is complicated in these places, and Precision Time Protocol (PTP) does not seem to be supported by all network nodes in the near future – so high precision time-stamping is indeed a current problem. This paper suggests a novel, practical solution to overcome the obstacles.

The core idea is that in real-life, distributed network monitoring systems operate with a few, finite number of probe-clusters, and their site should have a precise clock provided by PTP or GPS somewhere in the building. The distribution of time information within a site is still troublesome, even within a server rack. This paper presents a closed control loop solution implemented in an FPGA-based device in order to minimize the jitter, and compensate the calculated delay.

***Keywords*—network monitoring, time synchronization, hardware acceleration, closed control loop**

I. INTRODUCTION

Network monitoring has a well-established practice at telecommunication operators. There are fundamentally different solutions available – depending on what kind of data are initially available and how they are gathered. The least flexible solutions are based on the functional networking elements: they can provide pre-digested reports, statistical counters, and occasionally (when not under heavy load), even detailed information on the actual messages. Some operators use standalone protocol analyzers, which do not suffer from the temporal, load-related bottlenecks – rather, they have spatial data capture issues: only a segment of the network is visible at any given time. On the other hand, complete traffic information can be gathered by network-wide traffic monitoring. These latter solutions are based on passive, distributed probes; central processing entities; and client software – also distributed – at the operating personnel. This paper discusses a peculiar problem of such systems: effective time synchronization among the entities.

Network traffic analysis requires the understanding of the order of the messages appearing in the network, even if they appear at different interfaces. This makes high resolution and high precision time-stamping the basic requirement, beside lossless message capture. While there are standardized network protocols available for tackling this issue, there are practical obstacles in their network-wide usage. Although the Network Time Protocol (NTP) is widely available [1], it cannot be used as a general purpose synchronization protocol. In fact, the message transfer delay between NTP clients and servers is not compensated, hence the different nodes end up setting their local time to a clock value with a random delay. The typical order of the forwarding delay in current core routers is in the 0.5-5 microseconds range, depending on the traffic volume – among other factors. Since the minimum packet interarrival-time is 0.672 microsecond even at a 1 Gbps link (and 67.2 nanoseconds for a 10 Gbps link), such delays cannot be left without compensation for time synchronization.

Precision Time Protocol (PTP), on the other hand covers the delay-compensation issue well [2]. Unfortunately, PTP is not at all wide-spread, even after 10 years of commercialization for PTPv2. The concept, however, necessitates that all network nodes in the path have PTPv2 capability. Otherwise – even if one node cannot compute and share its delay data –, compensation of time information is not possible.

Another solution could be to introduce time information of GPS (Global Positioning System) satellites into the nodes – this is not feasible, since rack cabinets in server rooms lack the line of sight.

We can suppose that at least one machine at each monitoring site has the possibility to get synchronized to the master clock of the network (e.g. through PTP or GPS). Nevertheless, synchronizing all clocks within the site with nanosecond-range precision, is still a challenge.

This paper presents a solution for the time synchronization issues of systems with FPGA-based monitoring probes. What makes FPGA a key player here is that hardware-acceleration removes the jitter of operating system and protocol-stack delay from the equation. The delay of handling time information within an FPGA is constant, we can calculate with it precisely – and compensate this delay for the time-stamp.

In this paper we focus on the time synchronization challenges of a monitoring site. The implemented solution is based on the practical pre-requisite that each site has a reference clock available for the monitoring system. This paper suggests an FPGA-based clock synchronization method for the distributed monitoring equipment, more precisely, its interface cards.

The authors are with the Department of Telecommunications and MediaInformatics, Faculty of Electrical Engineering and Informatics, Budapest University of Technology and Economics, Magyar tudósok körútja 2., 1117 Budapest, Hungary (phone: +36704213213; e-mail: fecjanky@gmail.com and pvarga@tmit.bme.hu)

DOI: 10.36244/ICJ.2018.1.1

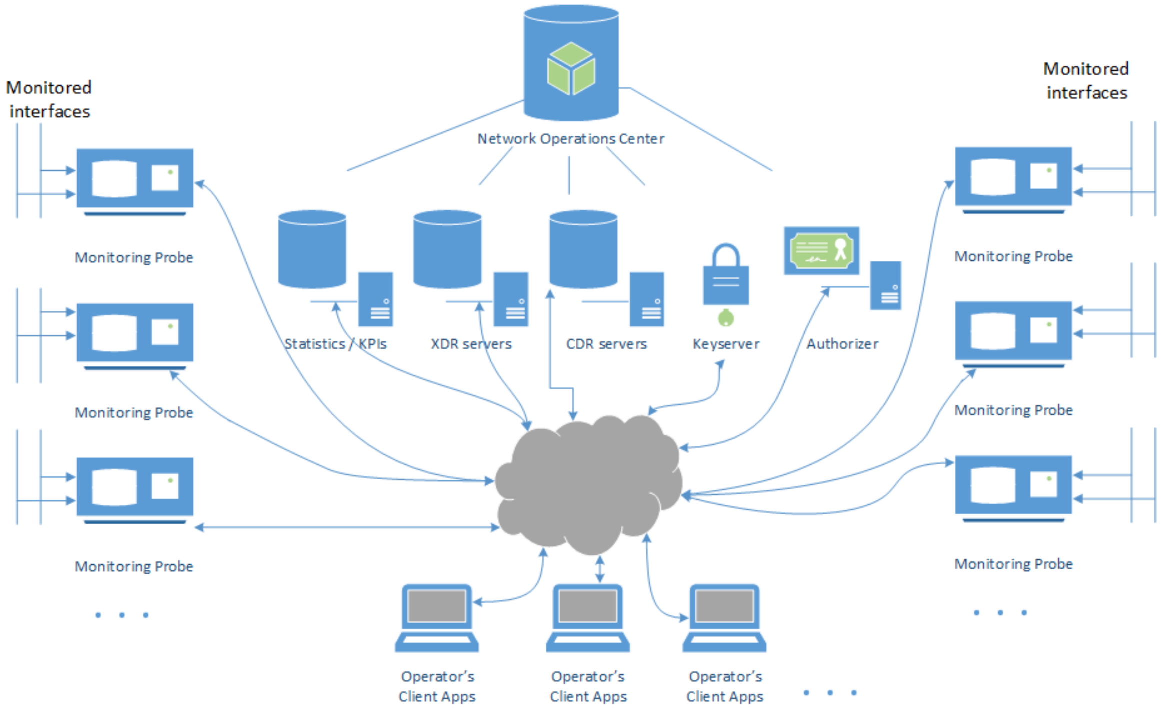


Fig. 1. A generic architecture for distributed network monitoring

II. Network Monitoring supported by FPGA-based Probes

A. *The Generic Concept of Distributed Network Monitoring*

The distributed network monitoring architecture depicted by Figure 1 supports local, probe-based pre-processing (time-stamping, requirement-based packet chunking, filtering criteria-based distribution) and central, deep analysis (correlation of messages and transactions, data record compilation, statistics generation), even on-the-fly. The time-based ordering and interleaving of messages are enabled by the hardware-accelerated time-stamping, providing nanosecond-range resolution with sub-microsecond precision. The information stored locally at the distributed Monitoring Probes can be accessed by client applications of the operator. Besides, the Monitoring Probes send pre-digested data to the Servers for correlation (creating e.g. Call Data Records, CDRs), as well as periodic reports containing their calculated statistics [3].

Since user data and control data are often carried over the same channels, their division requires message analysis on network- or transaction-level (e.g., IP- or TCP-level). The changing traffic patterns force the operators to look for new tools to process even the user traffic. The first step towards this is the compilation of XDRs (eXtended Data Records) based on control- and user-plane messages and transactions. These often contain message-level timestamps, as well. Based on these data, the deep traffic analysis tools provide valuable information towards business-intelligence and network optimization. Besides, all nodes can be configured to report directly to the NOC (Network Operations Center).

Operators use the network-wide, passive monitoring for fault detection, service quality assurance, and resource planning, among others [4]. Besides lossless data capture, network monitoring covers further functions, as well:

- precise time-stamping, ordering;
- compilation, search and fetch of Call Data Records (CDRs) and Extended Data Records (XDRs);
- calculation and reporting of Key Performance Indicators, KPIs;
- Call Tracing at various complexity levels;
- bit-wise message decoding for protocol analysis; etc.

All these functions are present in the network monitoring practice, since beside user-level data analysis, network analysis is important from connection-level to application-level, as well.

System elements of the described generic architecture can be implemented in many ways. In the SGA-7N system – which serves as the base implementation for the presented solution – monitoring probes of the presented system are called “Monitors”. These consist of three main building blocks: a high performance Field Programmable Gate Array (FPGA)-based custom hardware platform, a firmware dedicated for network monitoring, and the probe software [5].

B. *FPGA-based packet processing*

There are many features that make FPGAs useful in packet processing tasks [6]. The main concept itself allows *parallel processing* of the input data. Different, simultaneous tasks can be carried out at each clock cycle on the same data, which in this case is the packet header [7], [8]. Besides, the *input word length* is much greater for FPGAs (getting 90 bytes) than for modern CPUs (64 bits). Furthermore, FPGA are set up in hardware-defined languages, and they are indeed *reconfigurable* hardware: their internal wiring can be changed within milliseconds. These features enable FPGA-based hardware platforms to become high performance networking devices, e.g., network monitors, switches, routers, firewalls or intrusion detection systems [9]. Nevertheless, as a network monitoring system, it supports distributed and lossless packet level monitoring of Ethernet links for 1 or 10 Gbps.

Beside providing sufficient resources for switching and routing at 1 or 10 Gbps, the design of SGA-GPLANAR [10] and SGA-10GED [11] used in SGA-7N includes some special, network monitoring-related requirements, namely

– lossless packet capture,
– 64-bit time-stamping with sub-microsecond resolution,
– header-only capture: configurable depth of decoding,
– on-the-fly packet parsing by hardware [12],
– parameterized packet/flow generator for mass testing [13],[14].

Various applications then require other supported functionalities. As an example, the high-speed monitoring application [15] consists of the following sub-modules:

– time-stamping every frame upon reception;
– packet decoding from layer 2 up to the application layer;
– packet filtering with a reconfigurable rule-set to decide what we do with a given packet;
– packet chunking: packets can be truncated depending on the matching rule;
– packet distribution: to distribute packets by different criteria: IP flows, fragment steering, steering based on mobile core network parameters, etc.;
– packet encapsulation: monitoring information is stored in a specified header format.

These features and capabilities make the FPGA a suitable enabler of hardware acceleration within the Monitors.

III. Challenges and Requirements in Detail

For a distributed monitoring solution described in the previous sections, there is a strong requirement for having a *monotonic clock*. Otherwise, packet reordering would happen even with a single monitoring node (changing its clock) – and this is not feasible, since traffic analysis is heavily dependent upon packet timestamps. As a consequence, the need for monotonic system time is inherent.

Another challenge comes from the fact that a distributed monitoring system has its components geographically separated from each other, therefore the clock frequency and the time information of the clocks of the nodes have to be frequency- and phase-synchronized to each other with some given threshold. This problem has many solutions, e.g., using GPS based synchronization systems [16]. Although technically it can work well [17], as a drawback, this requires additional installation expenditures on an indoor site that has no installed antenna system to carry the GPS signal inside the building and could also result in extensive cabling work. A convenient alternative is to use network time synchronization that utilizes the telecommunication network for exchanging packets as per a designated protocol to achieve frequency and phase synchronization. Examples for this are Network Time Protocol (NTP) [1] and Precision Time Protocol (PTP) [2].

When speaking about time synchronization, the following properties describe a clock – which are in-line with the generic definition of clock properties [18]:

- accuracy – *i.e.* how good is the time information compared to some reference
- precision – *i.e.* how precise is a tick of the clock compared to some reference
- stability – *i.e.* how does the clock frequency change e.g., over time or based on external temperature changes etc.

The biggest challenge of all – as usual – is to adapt to the existing monitoring framework described in II with minimal modifications to the existing solution, while satisfying all the precision and accuracy related requirements. As mentioned before, the platform for proof-of-concept is the SGA-7N monitoring system, which utilizes FPGA-based monitoring cards. These are capable of capturing on high-speed network interfaces – with fine-grained time-stamping capabilities –, and they have their own, existing time-keeping facilities.

In order to tackle all the above mentioned issues with a solution fitting into the network monitoring architecture, we suggested to create a new FPGA-based card that implements these functions:

- network time synchronization,
- local time synchronization,
- interfacing with the existing nodes – OAMP functions.

The following sections describe this solution, and show its feasibility in the running monitoring system.

IV. Architecture of the Distributed Time Synchronized Monitoring System

A. Generic concept

For providing easy adaptation into the existing system, and also taking into account FPGA resource usage, a hybrid solution has been designed. This solution implements network time synchronization in a standalone card that distributes the digital timing information over a dedicated control bus, as illustrated by Figure 2.

The synchronization framework provides a platform-independent agent that can be integrated into the existing FPGA cards' top level VHDL (VHSIC Hardware Description Language, [19]) modules, and is used through a well-defined and portable interface.

The agent itself has low complexity, and as a result, the solution does not waste CLB (Configurable Logic Block) resources – as if the whole network synchronization stack were instantiated N times on all monitoring node cards. Furthermore, this results in better internal synchronization compared to the replicated stacks, since those can have skew to each other (within the boundaries), as specified by their protocol.

As shown by Figure 2, each node has its own network synchronization function, therefore the accuracy and precision

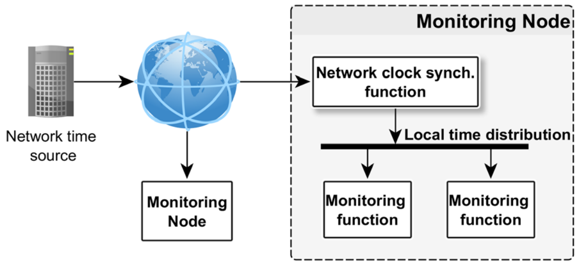


Fig. 2. Fitting the time synchronization function into the generic, distributed network monitoring concept

between two monitoring nodes can be guaranteed only to an extent that the utilized time synchronization protocol provides. Due to the uncompensated delay of routers, switches and transmission paths, this is in the magnitude of milliseconds of a software implementation of NTP. This precision, can be increased by using FPGAs for hardware acceleration. Depending on the PTP version and the underlying network capabilities, this can fall into the magnitude of nanoseconds.

The main idea of the solution is to install a local time-distribution bus between the nodes within a site. This allows us to achieve nanosecond-range synchronicity, as there is less perturbation between the hardware implementations of the transmitting and receiving ends – no OS scheduler, no network etc. Moreover, frequency synchronization can also be easily achieved by implementing a synchronous bus – i.e., transmitting the clock signal along with the data.

B. External time synch. subsystem design and implementation

When selecting the candidate for implementing the external time synchronization function, three protocols were considered:

- Network Time Protocol (NTP) [1],
- Precision Time Protocol v1 (PTPv1) [20],
- Precision Time Protocol v2 (PTPv2) [2].

In order to achieve the best synchronization between PTPv2 clocks, the protocol requires PTPv2-enabled switches/routers throughout the network. These do the bookkeeping of the processing delay values in the synchronization packets as they traverse through the network. Without this feature, the achievable synchronicity in a multi-hop network is around the same as by using PTPv1.

Since PTPv2 is not widely available in current networks, we concluded in either selecting NTP or PTPv1, due to their simplicity. PTPv1 has way more modes of operation when compared to NTP. Still, these two protocols are operating based on semantically the same principle when determining the round trip time and offset compared to a reference clock entity. Although there are significant differences originated from their packet structure, the time-stamp format and also the epoch that could result in more complex implementation if PTPv1 would be chosen. Still, the NTP time-stamp format includes a 32-bit unsigned seconds field spanning 136 years and a 32-bit fraction field resolving 232 picoseconds the prime epoch, or base date of era 0, is 0 h 1 January 1900 UTC – i.e., when all bits are zero.

Based on the requirements, the above considerations, and the Occam principle, the design decision led to selecting *NTP protocol to be used* for synchronizing the FPGA-based monitoring cards through a card that is responsible for implementing the external and internal (see Section IV-C) time synchronized function called SGA-Clock.

Each FPGA-based packet processing and networking protocol implementation has its own complexity. There are several readily available implementations that can be used for packet processing in FPGAs with some limited flexibility when it comes to interconnecting it with other modules. The one that has been used for the current implementation is a flexible solution for Protocol Implementations within FPGAs. The solution detailed in [21] provides a generic framework in VHSIC Hardware Description Language (VHDL) that enables rapid prototyping of networking protocols. Among many other things it provides the following main features:

- – supports protocol module interconnection via layering;
- – handles reception and transmission of Protocol Data Units (PDUs) with queuing;
- – provides a high level interface for separating and combining Protocol Control Information (PCI) and Service Data Unite (SDU), forwarding, pausing or dropping SDUs;
- – provides a unified way to handle Interface Control Information (ICI), SDU, and PDU events (e.g., error signalling) [22];
- – adds support of auxiliary information that travels along with messages
- – provides components for common tasks recurring during implementing networking protocols (de/serialization, arbitration etc.).

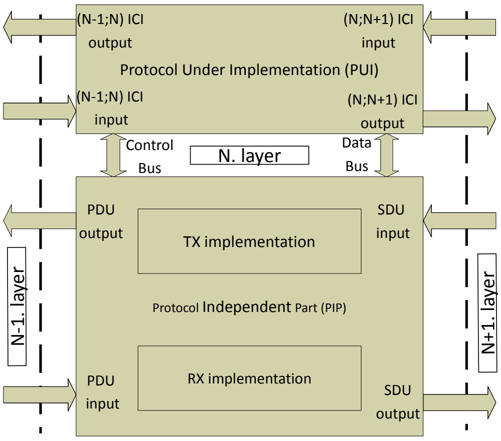


Fig. 3. Fundamental building block of the FPGA networking framework used for the Protocol Implementation

The framework's basic building block (shown by Figure 3) was used for implementing a pure FPGA-based UDP/IP protocol stack with ARP [23] on top of 802.3 Ethernet. It provides a platform with deterministic timing for the likewise FPGA-based implementation of NTP. For each of these protocols the corresponding protocol-specific parts have been described in VHDL, using the generic framework [21].

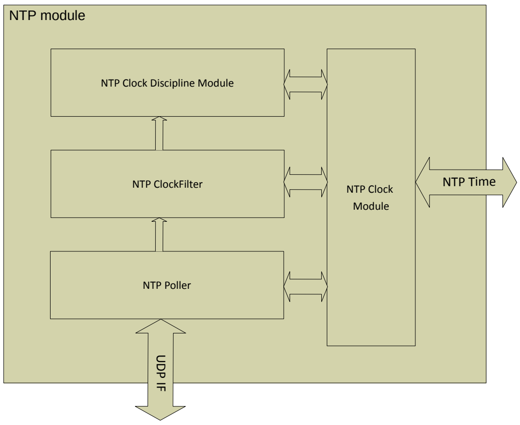


Fig. 4. NTP module block diagram of components

The internal structure of the NTP module is shown by Figure 4. The NTP Poller component is responsible for the NTP packet transmission and reception, and implementing the On-Wire protocol for determining the offset – based on the packet messages. The packet-handling part is also implemented through the Protocol Implementations framework. The NTP ClockFilter component is there to regulate the offset values presented by the poller by ordering the results based on delay, updating internal state variables, calculating jitter, and suppressing spikes based on jitter and last successful test time. If the offset data got passed the filter stage, it gets forwarded for further processing by the NTP Discipline module.

The NTP Discipline module controls the clock module – by adjusting the time increment – based on the filtered offset data. The NTP clock module provides an interface for controlling the time increment that itself is added to the clock register in each system clock cycle – thus implementing the clock functionality. The time information is fed back to each module as illustrated on Figure 4. This chain of modules with the feedback is another realization of a closed loop control chain described in the following section.

C. Internal time synch. subsystem design and implementation

Since time-stamping is done by the monitoring interface cards, the time synchronization information has to be spread around all interface cards of all monitoring units within the site. This time synchronization is an internal matter of the monitoring system. The relationship between “external” and “internal” time synchronization is shown by Figure 5.

The internal time information synchronization function is responsible for having all clocks in all monitoring functions to be completely synchronized *within a monitoring node*. Since this is an internal component, the amount of perturbation that potentially affects this subsystem is considered minimal compared to the external time synchronization subsystem.

The elements of this subsystem are:

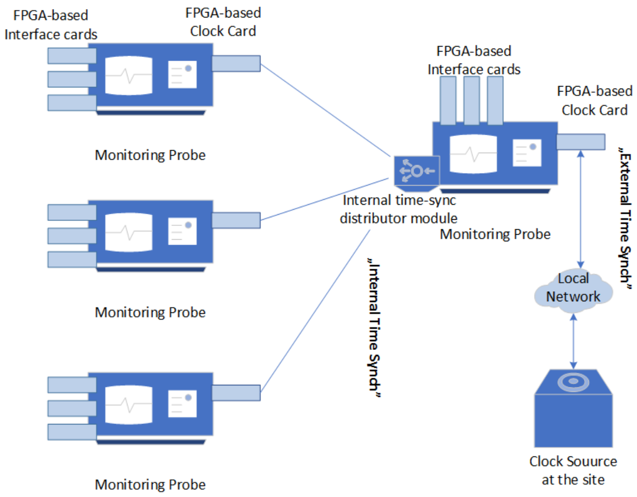


Fig. 5. Time synchronization within a monitoring site – methods for external and internal subsystems differ to allow high precision and accuracy in time-stamping

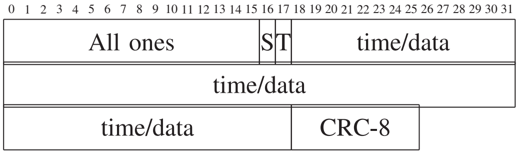


Fig. 6. High-Speed Time-stamp Interface frame format

- digital bus that is able to transmit time and status information;
- a driver module of that bus that resides in the Network clock synchronization function;
- receiver modules attached to that bus performing local time synchronization.

Internally to each monitoring probe, all FPGA boards that implement a monitoring function can operate from different power supply units. As a consequence, ground level isolation is necessary over the bus. For reducing the physical layer complexity, a point-to-point bus system has been designed. In order to be able to maximize the number of clients connected the bus, it utilizes an asynchronous serial communication using 2 wires that provides uni-directional communication – with this system bi-directional communication would require 4 wires. The communication protocol executed by the driver module (the internal time synch. distribution module in Figure 5) multiplexes arbitrary data units and the time information over the bus into frames – equipped with error detection code – in an alternating pattern. That results in periodic transmission of valid time information.

The parameters of the physical signalling are:

- LVCMOS33 (Low Voltage CMOS 3.3) levels for representing logical values;
- asymmetric signal transmission;
- 15.625 MHz clock frequency with 4x oversampling;
- NRZ line coding.

The frame format used on the bus is shown in Figure 6. The

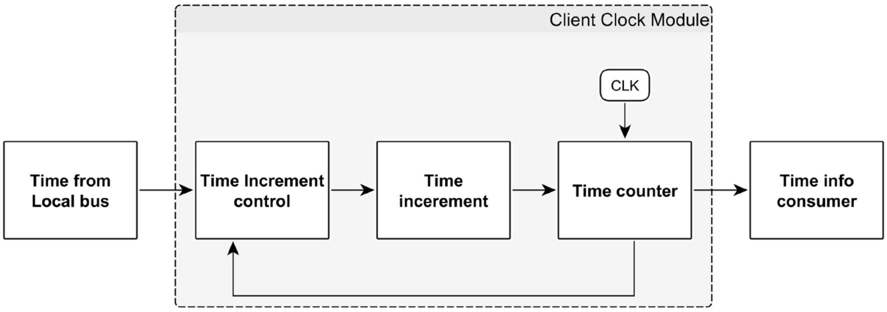


Fig. 7. Internal clock module diagram

frame starts with an all 1's preamble, and it is followed by a start bit with value 0. The type field is used to differentiate the payload types. When T=1 it indicates that the payload is time information otherwise it is data – hence an overlay data communication protocol can be used on this data channel. The time format is in line with the external time synchronization, i.e., it uses the NTP time format for representing the time information. For detecting transmission errors on the bus, a CRC-8 value is calculated for the *'Type'* and *'Payload'* fields and appended to the frame that is checked on frame reception for detecting transmission errors.

$$T_{frame} = N_{bits} \times T_{bit} \tag{1}$$

and

$$T_{bit} = 1/f_{signalling} \tag{2}$$

Where T_{bit} is the bit time, $f_{signalling}$ is the signalling frequency on the internal bus, and N_{bits} is the number of bits in the frame. Calculating (1) and (2) with the above given parameters, the frame time is 90/15.625 MHz = 5.76 µs. Since every other frame carries time information, the clock on the receiver side can be disciplined/controlled on a 11.52 µs basis. Under such short period of time even low quality, non-temperature controlled crystal oscillators have negligible drift. As a result, this update period is adequate for the network monitoring use case.

The receiver module (i.e., in the FPGA-based Clock Card in Figure 5) de-multiplexes the data and the time information from the payload of the received frame. It also verifies that the received frame's CRC-8 value matches the calculated one. If no errors were detected then it feeds the time information into a module that performs time synchronization executing the pseudo-code in algorithm 1

The client clock module – as shown in Figure 7 – is incrementing a clock counter with an increment value – corresponding to the nominal clock frequency in the internal time representation – in each system clock period. The clock module frequency can be adjusted through modifying the time increment itself. The $Delay_{static}$ constant can be measured for a given configuration and adjusted accordingly. The algorithm is illustrated by a sample waveform of the master and slave entity in Figure 8. Informally if the skew is less than the desired precision under one synchronization period – i.e. when the valid time is transmitted by the master entity – then the phase of time progresses in sync on the two entities.

Algorithm 1 Receiver local time disciple algorithm

```
$Increment \Leftarrow 1/f_{clk}$
$Delay_{static} \Leftarrow x$
$T_{local} \Leftarrow 0$
for Each rising edge of system clock do
  if Received valid time-stamp from Master then
    $T_{local} \Leftarrow T_{recv} + Increment + Delay_{static}$
  else
    $T_{local} \Leftarrow T_{local} + Increment$
  end if
end for
```

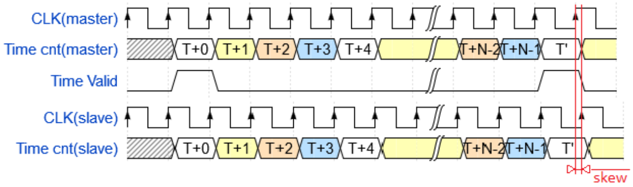


Fig. 8. Illustration of timing on the internal synchronization bus

To concede that this system can have nanosecond synchronization let us execute the algorithm: let T_n be the nth time point where synchronization occurs between the master and the slave entity. At time point T_n when the client receives a time-stamp the local clock will be in sync with the master clock – given that the $Delay_{static}$ constant was determined correctly. De-synchronization arises due to errors in the master and client clock oscillator frequency. To ensure that the desired level of synchronization is reached it has to be shown that the master and client clock would not diverge more than one nanoseconds under time interval (T_n, T_{n+1}) – since by definition at T_{n+1} the clocks will be in sync again and this process is periodic. Given a worst case calculation the following equation must hold true:

$$\begin{gathered}\left|\frac{1}{(1+\epsilon)f_{clk}}N - \frac{1}{(1-\epsilon)f_{clk}}N\right| \lesssim 1ns \\ \frac{N}{f_{clk}}\left|\frac{-2\epsilon}{1-\epsilon^2}\right| \lesssim 1ns \\ T_{ts}\left|\frac{-2\epsilon}{1-\epsilon^2}\right| \lesssim 1ns, where \\ N = \frac{T_{ts}}{T_{clk}}\end{gathered} \tag{3}$$

In equation 3 ϵ stands for the precision of the oscillator, f_{clk} is the system clock frequency and T_{ts} is the time-stamp frame time on the internal synchronization bus. In theory equation 3 can be satisfied for arbitrary ϵ if T_{ts} can be adjusted freely. Substituting parameters from the concrete implementation with $T_{ts} = 2T_{frame} = 11.52\,\mu\mathrm{s}$ and $\epsilon = 50$ ppm results in $1.15ns \lesssim 1ns$ which is approximately satisfying.

It is important to note that this accuracy and precision is only achieved over the *internal* time synchronization bus. If the external subsystem – that is completely orthogonal to the internal subsystem – synchronizes to its reference with $\mu\mathrm{s}$ accuracy then this results in the same accuracy for the monitoring probe vs. external reference relation. Even though the synchronicity will be still at the ns level in the monitoring probe vs. monitoring probe relation inside the same monitoring node driven by the same master.

D. Implementation

The realized system with all internal components is shown by Figure 9, where the external time synchronization – as presented in Section IV-B – is done by the SGA Clock card – visible in the bottom right part. Similarly, the internal time synchronization – described in Section IV-C – is performed over the local bus with agent modules. These modules run in all the FPGA-based monitoring cards acting as slaves at the high-speed time-stamp interfaces; all are driven by the SGA Clock card acting as a master.

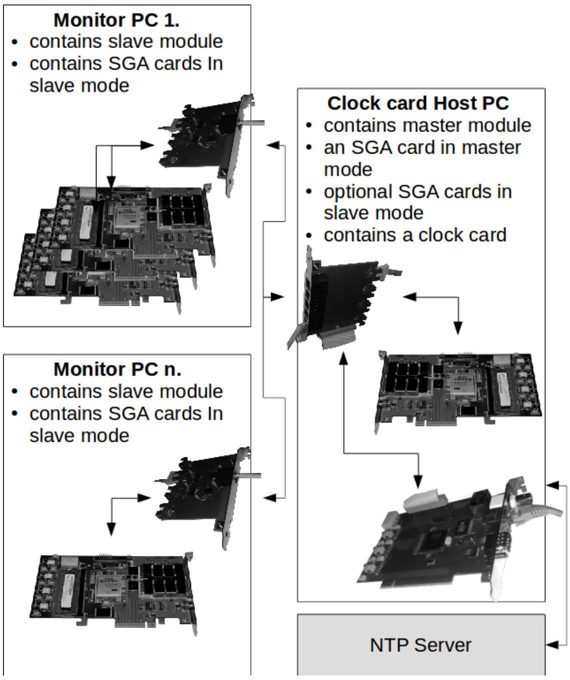


Fig. 9. The realized system with all internal components

V. Verification & Results

There has been extensive testing and measurements carried out for verifying the solution. In order to analyze the degree of synchronization to the master NTP clock, a packet capturer was installed on the Ethernet segment at which the FPGA implementation of the NTP slave was connected. The NTP packets used for synchronization were captured bidirectionally. This packet capture then was filtered for those NTP packets that had all 4 timestamps used in the On-Wire protocol to calculate the offset from the reference clock value. Our dedicated post-processing utility then extracted the offset information along with the time elapsed from the start of measurement – which is determined by the first NTP packet present in the packet capture.

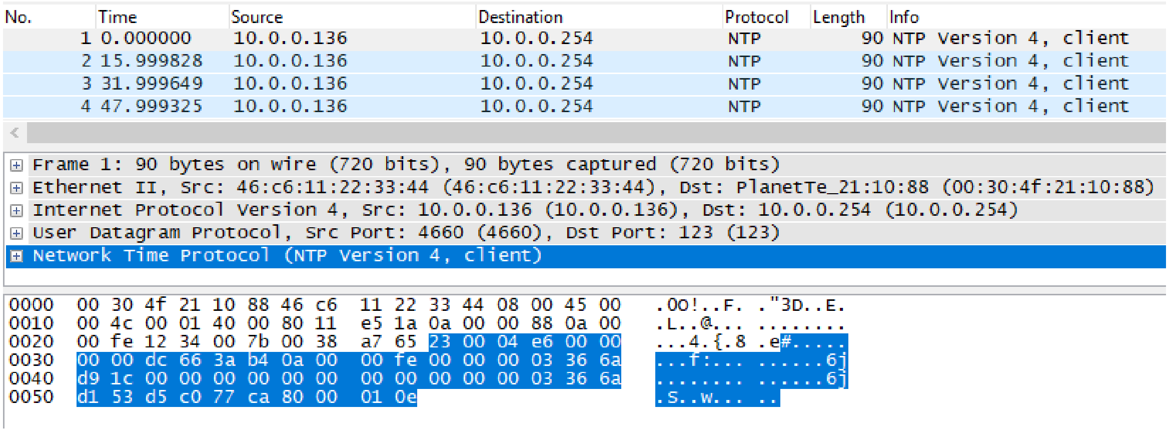
A sample packet capture is shown by Figure 10. The statistical parameters – like the clock drift and real offset – of the device was determined by fitting a linear curve on the offset values. There were various measurements carried out – for the actual measurement presented in this paper, the capture was taken for approximately 3 hours. The results and the fitted curve plot can be seen on Figure 11.

The curve fitted on this measurement shows that there was a fix $14.52\,\mu\mathrm{s}$ offset compared to the reference clock. As presented in Section III having a precise and stable clock – with known offset – is as good as having an accurate one. Besides, the first order stability of the device is $-0.7\,1/\mathrm{ns}$. This precision and stability are considered adequate for satisfying the requirements of the external time synchronization part.

As for the internal synchronization part, it is by design has accuracy and precision in the magnitude of $1\,\mathrm{ns}$ – see Section IV-C for details.

The $\sim\!1\,\mathrm{ns}$ accuracy over the internal synchronization bus satisfies the criteria of any monitoring system with 1, 10 or even 100 Gbit/s Ethernet, since packet inter-arrival time even of the latter case is 6.72ns [24] – almost one magnitude greater than the theoretical accuracy of the presented, implemented and verified time synchronization method.

As a real-life verification, the above described time-synchronization system has been put into operation at Magyar Telekom. The system hardware (with its FPGA firmware) has been installed beside the SGA-7N network monitoring system, and showed the expected result. The system provides accurate time information to the monitoring cards ever since, and it is planned to be expanded for covering all related monitoring cards, network-wide.



```
No.   Time        Source        Destination   Protocol  Length  Info
    1 0.000000    10.0.0.136    10.0.0.254    NTP           90 NTP Version 4, client
    2 15.999828   10.0.0.136    10.0.0.254    NTP           90 NTP Version 4, client
    3 31.999649   10.0.0.136    10.0.0.254    NTP           90 NTP Version 4, client
    4 47.999325   10.0.0.136    10.0.0.254    NTP           90 NTP Version 4, client

⊞ Frame 1: 90 bytes on wire (720 bits), 90 bytes captured (720 bits)
⊞ Ethernet II, Src: 46:c6:11:22:33:44 (46:c6:11:22:33:44), Dst: PlanetTe_21:10:88 (00:30:4f:21:10:88)
⊞ Internet Protocol Version 4, Src: 10.0.0.136 (10.0.0.136), Dst: 10.0.0.254 (10.0.0.254)
⊞ User Datagram Protocol, Src Port: 4660 (4660), Dst Port: 123 (123)
⊞ Network Time Protocol (NTP Version 4, client)

0000  00 30 4f 21 10 88 46 c6  11 22 33 44 08 00 45 00   .0O!..F. ."3D..E.
0010  00 4c 00 01 40 00 80 11  e5 1a 0a 00 00 88 0a 00   .L..@... ........
0020  00 fe 12 34 00 7b 00 38  a7 65 23 00 04 e6 00 00   ...4.{.8 .e#.....
0030  00 00 dc 66 3a b4 0a 00  00 fe 00 00 00 03 36 6a   ...f:... ......6j
0040  d9 1c 00 00 00 00 00 00  00 00 00 00 00 03 36 6a   ........ ......6j
0050  d1 53 d5 c0 77 ca 80 00  01 0e                     .S..w... ..
```

Fig. 10. Measurement result – a typical example of an NTP packet containing four timestamps

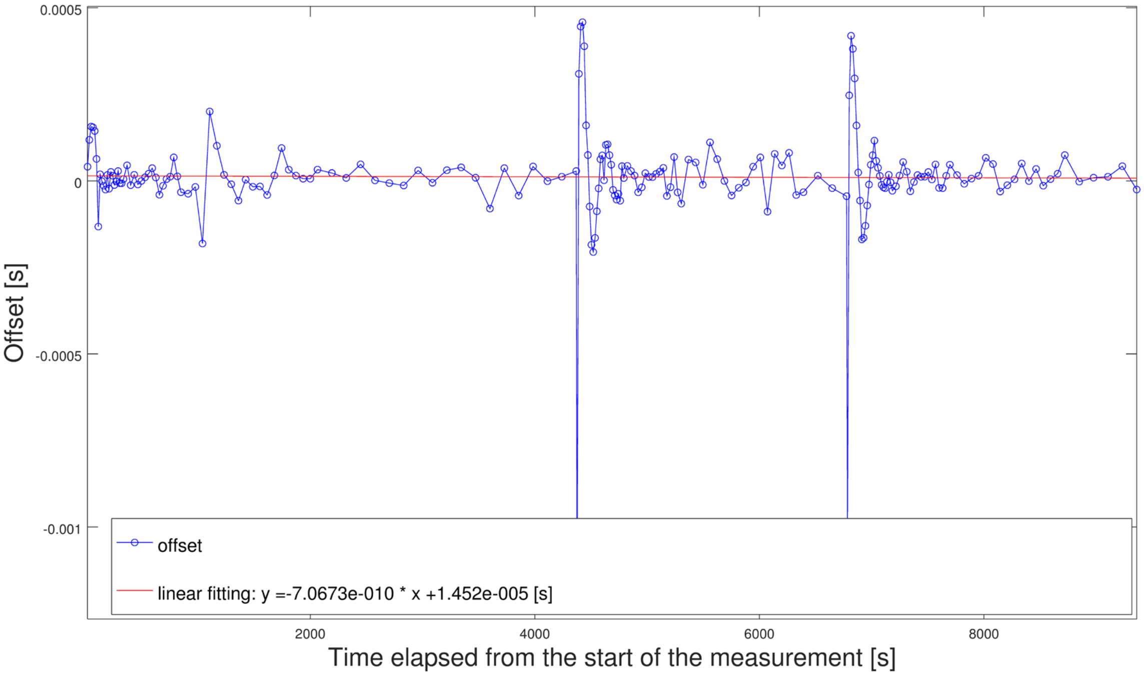


Fig. 11. Measurement result – plot of derived offset-measurement data

VI. Conclusion

In this paper, we introduced a general time synchronization solution for a high performance, lossless network monitoring system called SGA-7N that is based on a reconfigurable architecture. The probes of the system are called “Monitors”, which consists of three main building blocks: a high performance Field Programmable Gate Array (FPGA)-based custom hardware platform, a firmware dedicated for network monitoring, and the probe software. The reconfigurable property of the FPGA chip enables to turn the Monitor hardware platform into a high performance networking device – among others, a network monitoring probe. Beside supporting distributed and lossless packet level monitoring of Ethernet links for 1 or 10 Gbps of the described system, the FPGA serves as the base platform of the time synchronization solution for the interface cards of the Monitors.

Time synchronization of the network monitoring nodes are crucial, since the analysis depends highly on the proper message sequence, which is determined mainly by the timestamps.

First, each monitoring site has to have a reference clock that is synchronized with other reference clocks at other sites. Naturally, the monitoring system has to be synchronized to the reference clock available at the physical site. In this paper we call it external time synchronization, and it is solved by an FPGA-based, NTP implementation that use data filtering and has a clock discipline module in order to output monotonous clock information. This avoids timestamps jumping backwards, or jumping forward too much within one step, hence the clock if the monitoring system becomes well-regulated. Each interface at the monitoring node has to get synchronized with this clock information. In this paper we call it internal time synchronization, and it is implemented through a proprietary time-synchronization protocol. Its sender, (or master) part works in a distributor-card residing at the main reference clock machine of the monitoring system, whereas the receiver (or slave) parts are realized within the FPGA of the monitoring cards.

As presented in the paper, the overall system shows sub-nanosecond accuracy and stability, meeting the requirements of 10 Gbps, or even 100 Gbps Ethernet-based packet monitoring. The presented solution is already installed in the network-wide, real-life monitoring at Magyar Telekom.

REFERENCES

[1] D. L. Mills, *Computer Network Time Synchronization: The Network Time Protocol*. Boca Raton, FL: CRC Press, 2006.

[2] *IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems (PTPv2)*, IEEE Standard 1588-2008.

[3] P. Tatai, G. Marosi, and L. Osvath, "A flexible approach to mobile telephone traffic mass measurement and analysis," in *18th IEEE Instrumentation and Measurement Technology Conference*, Budapest, Hungary, May 2001.

[4] D. Kozma, G. Soos, and P. Varga, "Supporting LTE Network and Service Management through Session Data Record Analysis," *Infocommunications Journal*, vol. 71, no. 2, pp. 11–16, June 2016.

[5] AITIA, "SGA-NETMON The GSM/GPRS/UMTS/LTE Network Monitoring System," White Paper, 2012. [Online]. Available: http://sga.aitia.ai/pdfs/SGA-NetMon.pdf

[6] J. W. Lockwood, N. McKeown, G. Watson, G. Gibb, P. Hartke, J. Naous, R. Raghuraman, and J. Luo, "NetFPGA - An Open Platform for Gigabit-rate Network Switching and Routing," in *IEEE MSE*, San Diego, CA, US, 2007.

[7] N. Possley, "Traffic Management in Xilinx FPGAs," White Paper, 2006. [Online]. Available: http://www.xilinx.com/support/documentation/white_papers/wp244.pdf

[8] W. Jiang and V. K. Prasanna, "Scalable Packet Classification on FPGA," *IEEE Transactions on Very Large Scale Integration Systems*, vol. 20, no. 9, pp. 1668–1680, August 2011.

[9] P. Orosz, T. Tothfalusi, and P. Varga, "C-GEP: Adaptive Network Management with Reconfigurable Hardware," in *14th IEEE International Symposium on Integrated Management (IM)*, Ottawa, Canada, 2015.

[10] *SGA-GPlanar product description*, Aitia International Inc., accessed: 2017-12-30. [Online]. Available: http://www.fpganetworking.com/gplanar/

[11] *SGA-10GED product description*, Aitia International Inc., accessed: 2017-12-30. [Online]. Available: http://www.fpganetworking.com/10ged/

[12] V. Pus, L. Kekely, and J. Korenek, "Low-Latency Modular Packet Header Parser for FPGA," in *ACM/IEEE Symposium on Architectures for Networking and Communications Systems*, Austin, TX, USA, October 2012.

[13] P. Olaszi, "Complex Load Testing of Mobile PS and CS Core," EuroNOG 2012, September 2012. [Online]. Available: http://www.data.proidea.org.pl/euronog/2edycja/materials/Peter_Olaszi-Complex_Load_Testing_of_Mobile_PS_and_CS_Core.pdf

[14] G. Soos and P. Varga, "Use Cases for LTE Core Network Mass Testing," in *5th Mesterproba*, Budapest, Hungary, 2016.

[15] P. Varga, L. Kovacs, T. Tothfalusi, and P. Orosz, "C-GEP: 100 Gbit/s Capable, FPGA-based, Reconfigurable Networking Equipment," in *16th IEEE International Conference on High Performance Switching and Routing (HPSR)*, Budapest, Hungary, 2015.

[16] B. Sterzbach, "GPS-based Clock Synchronization in a Mobile, Distributed Real-Time System," *Real-Time Systems*, vol. 12, no. 1, pp. 63—-75, January 1997.

[17] J. J. Garnica, V. Moreno, I. Gonzalez, S. Lopez-Buedo, F. J. Gomez-Arribas, and J. Aracil, "ARGOS: A GPS Time-Synchronized Network Interface Card based on NetFPGA," in *2nd North American NetFPGA Developers Workshop*, Stanford, CA, USA, August 2010.

[18] D. L. Mills, "Internet Time Synchronization: The Network Time Protocol," *IEEE Transactions on Communications*, vol. 39, no. 10, pp. 1482–1493, October 1991.

[19] *IEEE Standard VHDL Language Reference Manual*, IEEE Standard 1076-2008.

[20] *IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems*, IEEE Standard 1588-2002.

[21] F. N. Janky, "A Flexible Architecture for Protocol Implementations within FPGAs," in *25th Telecommunications Forum (TELFOR)*, Belgrade, Serbia, 2017.

[22] *Information technology - Open Systems Interconnection - Basic Reference Model: The Basic Model*, ISO/IEC Standard 7498-1:1994(E).

[23] *An Ethernet Address Resolution Protocol*, IETF Internet Standard RFC 826.

[24] P. Mooney, "40/100-Gigabit Ethernet: Watching the Clock," White Paper, 2009. [Online]. Available: http://www.lightwaveonline.com/articles/print/volume-26/issue-10/applications/40100-gigabit-ethernet-watching-the-clock.html

Ferenc Nandor Janky received the MSc degree in Electrical Engineering from BME, Budapest, Hungary, in 2013. He gained experienced while working for various telecommunication companies including Vodafone, AITIA International Inc. and Ericsson. His main areas of interest are network protocols, FPGA programming and software development. Ferenc is currently working as a C++ software developer for an international corporate bank. He is a member of the SmartComLab at BME TMIT.

Pal Varga is Associate Professor at BME, Hungary, where he received his Ph.D. (2011) from. Besides, he is director in AITIA International Inc. Earlier he was working for Ericsson Hungary and Tecnomen Ireland, as software design engineer and system architect, respectively. His main research interest include network performance measurements, root cause analysis, fault localization, traffic classification, end-to-end QoS and SLA issues, as well as hardware acceleration, and Internet of Things. He is also a member of the SmartComLab at BME TMIT.

Cryptanalysis and Improvement of a Two-Factor User Authentication Scheme

Devender Kumar, Satish Chand, and Bijendra Kumar

***Abstract*—Recently, Wang-Wang have discussed a two birds with one stone: two-factor authentication with security beyond conventional bound. We find that this scheme is vulnerable to the password exposure attack and also does not offer user anonymity, which is an important feature for some of the applications like e-healthcare services, e-banking, etc. In this paper, we provide the solution to these problems.**

***Index Terms*—Two-factor authentication, password exposure attack, user anonymity, smart card, offline password guessing attack, insider attack**

I. Introduction

In the era of internet, most of the resources and services are available online. However, the security is an important issue to access online resources and services. A remote user authentication scheme can help to access online resources and services securely. Such scheme allows a user and a server to authenticate each other over an insecure channel. In 1981, Lamport [1] developed the first remote user authentication scheme in which the server was required to keep a password table. Since then, many smart card based remote user authentication schemes [2], [3], [4], [5], [6], [7], [8] have been discussed that do not require password tables.

In 2009, Xu et al. discussed a user authentication scheme based on smart card [9] and claimed that it is secured even if the smart card is lost. Sood et al. [10] found that the scheme [9] is not resistant to forgery attack and they improved it by overcoming its weakness. The paper [11] cryptanalyzed the scheme [9] and found that it is not resistant to the impersonation attack if a valid but malicious user uses the information stored in his own smart card. They improved this scheme to overcome its limitation. Horng et al. [12] found that the scheme [11] is not resistant to the insider and offline password guessing attacks.

In 2014, Chen et al. [13] cryptanalyzed the schemes [10], [11] and they found that the scheme [10] does not offer mutual authentication and the scheme [11] is not resistant to the smart card loss and off-line guessing attacks. They designed an improved scheme to remove these flaws. Jiang et al. [14] found that the scheme [13] is not secured against the offline password guessing attack and designed an improved scheme to overcome this problem. Mishra et al. [15] discussed the security issues of the scheme [14] and showed that it is susceptible to the insider, user impersonation and password guessing attacks. They designed a scheme to overcome these security flaws.

Recently, Wang-Wang [16] have discussed a two factor authentication scheme and suggested twelve independent security criteria that a two-factor authentication scheme should satisfy as follows: (i) no verifier-table (ii) no password exposure (iii) no smart card loss attack (iv) password friendly (v) resistance to known attacks (vi) provision of key agreement (vii) sound repairability (viii) no clock synchronization (ix) mutual authentication (x) timely typo detection (xi) user anonymity (xii) forward secrecy. Out these, user anonymity and password exposure are the essential properties of an user authentication scheme. User anonymity means user identity-protection and un-traceability. That is the scheme should protect user identity and prevent user activities from tracing. Password exposure means that the privileged administrator cannot get the user's password. In this paper, we analyze the security of the scheme [16] and find that it is susceptible to the password exposure attack and also lacks user anonymity. We present an improved scheme to overcome its limitations.

A. Threat model

Here, we present the capabilities of an attacker A as follows:

- A can eavesdrop all the transmitted messages between the participants over a public channel.
- A can reroute, resend, delete, modify and insert the eavesdropped messages.
- A can take out all the information saved in the smart card of a valid user if it is obtained by A somehow [17], [18].
- A cannot know the user's password as well as steal the user's smart card at the same time.
- A can enumerate offline all possible elements in the cartesian product $D_{id} \times D_{pw}$ in a reasonable amount of time [16].
- The privileged administrator may act as an attacker A.

The remaining paper is arranged as follows: section II reviews the Wang-Wang's scheme in brief and section III presents its cryptanalysis. Section IV introduces our proposed scheme and its performance analysis is presented in section V. Its formal security analysis is same as that of the Wang-Wang's scheme as we do not change the parameters which are transmitted via a public channel and hence it is omitted. Finally, section VI concludes the paper.

II. Review of Wang-Wang's Scheme

Here, we briefly review the robust password authentication scheme using smart card by Wang-Wang [16] that consists of

D. Kumar is with the Division of Information Technology, NSIT, New Delhi, India-110078, Phone:+919013489217, Fax: +91-11-25099022, e-mail: dk_iitm@yahoo.co.in.

S. Chand is with School of Computer and Systems Sciences, JNU, New Delhi, India-110067.

B. Kumar is with the Division of Computer Engineering, NSIT, New Delhi, India-110078.

DOI: 10.36244/ICJ.2018.1.2

the following four phases. The notations used in this paper are given in Table I.

TABLE I
NOTATIONS USED IN PAPER

Notations	Description
U_i	i^{th} User
S	Remote server
A	Attacker
ID_i	U_i's identity
PW_i	U_i's password
x	S's secret key
y	S's public key
p, q	Large prime numbers
n_0	An integer
$Honey\_List$	Link list
m_0	Number of items in $Honey\_List$
$H_i(.)$	One-way hash function
g	Generator of a prime order cyclic group G
$\|\|$	Concatenation operator
SC	Smart card
$\oplus$	XOR operator

A. Registration phase

User U_i performs the below steps to register with server S:

1) U_i selects his identity ID_i, password PW_i, and a random string b.
2) He sends $\{ID_i, H_0(b||PW_i)\}$ to S through a private channel.
3) After obtaining the request from U_i at time T, S chooses a random number a_i and calculates $A_i = H_0((H_0(ID_i) \oplus H_0(b||PW_i)) \ mod \ n_0)$. S verifies if U_i is a registered user. If not, then S stores the information $\{ID_i, T_{reg} = T, a_i, Honey\_List = NULL\}$ in its database; otherwise, it replaces the value of T_{reg} with T, a_i with newly selected a_i, and $Honey\_List$ with $NULL$ in its database corresponding to U_i. S then calculates $N_i = H_0(b||PW_i) \oplus H_0(x||ID_i||T_{reg})$.
4) S stores the information $\{N_i, A_i, A_i \oplus a_i, q, g, y, n_0, H_0(.), ...H_3(.)\}$ in a SC and transmits it to U_i securely.
5) After obtaining the smart card, U_i stores b into it; thus, the smart card contains $\{N_i, A_i, A_i \oplus a_i, q, g, y, n_0, H_0(.), ..., H_3(.), b\}$

B. Login phase

The steps are performed as below in this phase:

1) U_i inputs his identity ID_i^* and password PW_i^* after inserting his SC into the card reader attached with the system.
2) SC calculates $A_i^* = H_0((H_0(ID_i^*) \oplus H_0(b||PW_i^*)) \ mod \ n_0)$ and checks if $A_i^* = A_i$. If it is not true, the session is terminated.
3) SC selects a random number u and calculates $C_1 = g^u \ mod \ p$, $Y_1 = y^u \ mod \ p$, $k = H_0(x||ID_i||T_{reg}) = N_i \oplus H_0(b||PW_i^*)$, $a_i = (A_i \oplus a_i) \oplus A_i$, $CID_i = ID_i^* \oplus H_0(C_1||Y_1)$, $CAK_i = (a_i||k) \oplus H_0(C_1||Y_1)$, and $M_i = H_0(Y_1||k||CID_i||CAK_i)$.
4) U_i sends the message $\{C_1, CID_i, CAK_i, M_i\}$ to S through a public channel.

C. Verification phase

On getting the login message $\{C_1, CID_i, CAK_i, M_i\}$ from U_i, S performs the below steps:

1) S calculates $Y_1 = (C_1)^x \ mod \ p$ and $ID_i = CID_i \oplus H_0(C_1||Y_1)$. It verifies the format of ID_i. If it is not found in correct format, then the session is terminated.
2) S calculates $k = H_0(x||ID_i||T_{reg})$ and $M_i^* = H_0(Y_1||k||CID_i||CAK_i)$, where T_{reg} is excerpted from its database corresponding to the entry ID_i. It checks if $M_i^* = M_i$. If it is false, the session is terminated.
3) S computes $a_i^{`}||k^{`} = CAK_i \oplus H_0(C_1||Y_1)$ and verifies if $a_i^{`}$ is equal to the stored $a_i$. If it is false, $S$ rejects the request; otherwise, it check if $k^{`} = k$. If it is true, then perform next step; otherwise, if $a_i^{`} = a_i$ and $k_i^{`} \neq k_i$, then S concludes that the card of U_i is corrupted with a probability $1 - \frac{1}{2^{n_0}}$. In that case, S either enters $k^{`}$ into Honey_List if $|Honey\_List| < m_0$ (e.g. $m_0 = 10$) or suspends the smart card of U_i until he re-registers (i.e. when $|Honey\_List| = m_0$).
4) S creates a random number v and calculates the temporary key $K_S = (C_1)^v \ mod \ p, C_2 = g^v \ mod \ p$ and $C_3 = H_1(ID_i||ID_S||Y_1||C_2||k||K_S)$. S sends the message $\{C_2, C_3\}$ to U_i via a public channel.
5) After getting the message $\{C_2, C_3\}$ from S, SC calculates $K_U = (C_2)^u \ mod \ p, C_3^* = H_1(ID_i||ID_S||Y_1||C_2||k||K_U)$, and checks if $C_3^* = C_3$. If it is right, U_i authenticates S and calculates $C_4 = H_2(ID_i||ID_S||Y_1||C_2||k||K_U)$. U_i sends the message $\{C_4\}$ to S via an insecure channel.
6) On obtaining the message $\{C_4\}$ from U_i, S calculates $C_4{}^* = H_2(ID_i||ID_S \ ||Y_1||C_2||k||K_S)$, and checks if $C_4^* = C_4$. If it is true, S authenticates U_i and accepts the login request; otherwise, the session is terminated.
7) U_i and S share the session key $sk_U = H_3(ID_i||ID_S||Y_1||C_2||k||K_U) = H_3(ID_i||ID_S||Y_1|| C_2||k||K_S) = sk_S$ for secured future communication.

D. Password change phase

User U_i performs the below steps in this phase:

1) U_i inputs his ID_i and PW_i after inserting his SC into the card reader attached with the system.
2) SC calculates $A_i^* = H_0((H_0(ID_i) \oplus H_0(b||PW_i)) \ mod \ n_0)$ and checks if $A_i^* = A_i$. If it is not true, the request for changing password is rejected.
3) Smart card prompts U_i to enter a new password PW_i^{new} and calculates $N_i^{new} = N_i \oplus H_0(b||PW_i) \oplus H_0(b||PW_i^{new})$ and $A_i^{new} = H_0((H_0(ID_i) \oplus H_0(b||PW_i^{new})) \ mod \ n_0)$. It replaces the values of N_i, A_i and $a_i \oplus A_i$ with N_i^{new}, A_i^{new} and $a_i \oplus A_i^{new}$, respectively. Thus the password is changed successfully.

III. CRYPTANALYSIS OF WANG-WANG'S SCHEME

We cryptanalyze the Wang-Wang's scheme [16] based on the threat model as given in section I-A and find the following security problems:

A. Password exposure attack

Since user U_i sends $\{ID_i, H_0(b||PW_i)\}$ to S in step (2) of registration phase, a malicious privileged administrator A has knowledge of these two parameters. Assume that A somehow gets access to the U_i's smart card [19], then he can find U_i's password PW_i as follows:

1) Choose a password PW_i^* and compute $H_0(b||PW_i^*)$
2) Check if $H_0(b||PW_i^*) = H_0(b||PW_i)$. If it is true, then A gets the correct password PW_i of U_i and stops the procedure. Otherwise, repeat the steps (1) and (2).

Hence, user U_i's password is not safe from a malicious privileged administrator in this scheme.

B. User anonymity

Since user U_i sends the message $\{ID_i, H_0(b||PW_i)\}$ to S in step (2) of registration phase, his identity ID_i is transmitted in plaintext. Thus, his identity is not anonymous from a malicious privileged administrator A.

IV. Our scheme

In this section, we present our improved scheme by overcoming the weaknesses of the scheme [16]. In the registration phase of our scheme, we send the hash value of the user's identity ID_i and random string b instead of sending ID_i directly in plaintext to provide user anonymity. To resist from password exposure attack, we store the encrypted value of the random string b using XOR operation in the memory of the smart card in the registration phase. Our scheme consists of the following four phases:

A. Registration phase

User U_i executes the below steps to register with server S:

1) U_i selects his identity ID_i, password PW_i, and a random string b.
2) He sends $\{H_0(b||ID_i), H_0(b||PW_i)\}$ to S through a secure channel.
3) After obtaining the registration message from U_i at time T, S selects a random number a_i and calculates $A_i = H_0((H_0(b||ID_i) \oplus H_0(b||PW_i))\ mod\ n_0)$. S verifies from its database whether U_i is a registered user. If not, S stores the information $\{H_0(b||ID_i), T_{reg} = T, a_i, Honey\_List = NULL\}$ in its database; otherwise, it replaces the value of T_{reg} with T, a_i with newly selected a_i, and $Honey\_List$ with NULL in its database corresponding to U_i. Then, S calculates $N_i = H_0(b||PW_i) \oplus H_0(x||H_0(b||ID_i)||T_{reg})$.
4) S stores the information $\{N_i, A_i, A_i \oplus a_i, q, g, y, n_0, H_0(.), ...H_3(.)\}$ in a SC and sends it to U_i securely.
5) After obtaining the smart card, U_i computes $c = b \oplus H_0(ID_i \oplus PW_i)\ mod\ n_0$ and stores c into it and finally the SC contains the data $\{N_i, A_i, A_i \oplus a_i, q, g, y, n_0, H_0(.), ...H_3(.), c\}$

B. Login phase

The following steps are executed in this phase:

1) User U_i inputs his identity ID_i^* and password PW_i^* after inserting his SC into the card reader attached with the system.
2) SC calculates $b = c \oplus H_0(ID_i^* \oplus PW_i^*)\ mod\ n_0$ and $A_i^* = H_0((H_0(b||\ ID_i^*) \oplus H_0(b||PW_i^*))\ mod\ n_0)$ and checks if $A_i^* = A_i$. If it is not true, the session is terminated.
3) SC selects a random number u and computes $C_1 = g^u\ mod\ p$, $Y_1 = y^u\ mod\ p$, $k = H_0(x||H_0(b||ID_i^*)||T_{reg}) = N_i \oplus H_0(b||PW_i^*)$, $a_i = (A_i \oplus a_i) \oplus A_i^*$, $CID_i = H_0(b||ID_i^*) \oplus H_0(C_1||Y_1)$, $CAK_i = (a_i||k) \oplus H_0(C_1||Y_1)$, and $M_i = H_0(Y_1||k||CID_i||CAK_i)$.
4) U_i sends the message $\{C_1, CID_i, CAK_i, M_i\}$ to S via a public channel.

C. Verification phase

On obtaining the login request $\{C_1, CID_i, CAK_i, M_i\}$ from U_i, S executes the following steps:

1) S calculates $Y_1 = (C_1)^x\ mod\ p$ and $H_0(b||ID_i) = CID_i \oplus H_0(C_1||Y_1)$. It checks the entry of $H_0(b||ID_i)$ in its database. If it is not found, the session is rejected.
2) S calculates $k = H_0(x||H_0(b||ID_i)||T_{reg})$ and $M_i^* = H_0(Y_1||k||CID_i||CAK_i)$, where T_{reg} is excerpted from its database corresponding to the entry $H_0(b||ID_i)$. It checks if $M_i^* = M_i$. If it is false, the session is terminated.
3) S computes $a_i^{‘}||k^{‘} = CAK_i \oplus H_0(C_1||Y_1)$ and verifies if $a_i^{‘}$ is equal to the stored a_i. In case of inequality, S denies the request; otherwise, it check if $k^{‘} = k$. If it is true, then perform next step; otherwise, if $a_i^{‘} = a_i$ and $k_i^{‘} \neq k_i$, then S concludes that the card of U_i is corrupted with a probability $1 - \frac{1}{2^{n_0}}$. In that case, S either enters $k^{‘}$ into Honey_List if $|Honey\_List| < m_0$ (e.g. $m_0 = 10$) or suspends the smart card of U_i until he re-registers (i.e. when $|Honey\_List| = m_0$).
4) S creates a random number v and calculates the temporary key $K_S = (C_1)^v\ mod\ p, C_2 = g^v\ mod\ p$ and $C_3 = H_1(H_0(b||ID_i)||ID_S||Y_1||C_2||k||K_S)$. S sends the message $\{C_2, C_3\}$ to U_i via an insecure channel.
5) After obtaining the message $\{C_2, C_3\}$ from S, the SC calculates $K_U = (C_2)^u\ mod\ p, C_3^* = H_1(H_0(b||ID_i)||ID_S||Y_1||C_2||k||K_U)$, and checks if $C_3^* = C_3$. If it is true, U_i authenticates S and calculates $C_4 = H_2(H_0(b||ID_i)\ ||ID_S||Y_1||C_2||k||K_U)$. U_i sends the message $\{C_4\}$ to S via a public channel.
6) After obtaining the message $\{C_4\}$ from U_i, S calculates ${C_4}^* = H_2(H_0(b||\ ID_i)||ID_S||Y_1||C_2||k||K_S)$, and checks if $C_4^* = C_4$. If it is true, S authenticates U_i and accepts his login request; otherwise, the session is terminated.
7) U_i and S share the session key $sk_U = H_3(H_0(b||ID_i)||\ ID_S||Y_1||C_2||k||K_U) = H_3(H_0(b||ID_i)||ID_S||Y_1||C_2\ ||k||K_S) = sk_S$ for secured future communication.

D. Password change phase

User U_i performs the following steps to change his password:

1) U_i inputs his ID_i and PW_i after inserting his SC into the card reader attached with the system.
2) SC calculates $b = c \oplus H_0(ID_i \oplus H_0(PW_i)$ and $A_i^* = H_0((H_0(b||ID_i) \oplus H_0(b||PW_i)) \ mod \ n_0)$ and checks if $A_i^* = A_i$. If it is not true, the request for changing password is rejected.
3) Smart card prompts U_i to enter a new password PW_i^{new} and calculates $N_i^{new} = N_i \oplus H_0(b||PW_i) \oplus H_0(b||PW_i^{new})$ and $A_i^{new} = H_0((H_0(b||ID_i) \oplus H_0(b||PW_i^{new})) \ mod \ n_0)$. It replaces the values of N_i, A_i and $a_i \oplus A_i$ with N_i^{new}, A_i^{new} and $a_i \oplus A_i^{new}$, respectively. Thus the password is changed successfully.

V. PERFORMANCE ANALYSIS

In this section, we compare our scheme with that of the related schemes [20], [14], [21], [22], [16] in terms of communication cost, computational cost and security fetures. Like in other works, we have not considered the cost of lightweight operations like exclusive-or and concatenation operations. We have taken the length of parameter n_0 as $32\,bits$ and the user identity ID_i, password PW_i, random numbers, timestamps, and output of hash function have taken as $128\,bits$ long each; while the lengths of y and g are taken as 1024 $bits$ each, similar to that in the scheme [16].

From Table II, it is evident that the scheme [14] has the highest communication cost ($3456\,bits$). The communication cost of our scheme is same as that of the scheme [16]; however, the scheme [16] does not provide the security features like password exposure and user anonymity as shown in Table IV. The scheme [22] has the least communication cost, i.e. (1792 $bits$); however, it does not provide the security features like password exposure, smart card loss attack, sound repairability and user anonymity. Thus, our scheme has better performance than the related schemes [20], [14], [21].

TABLE II
COMMUNICATION COST

Scheme	Communication cost(bits)
Islam [20]	$1408 + 1408 = 2816$
Jiang et al. [14]	$2304 + 1152 = 3456$
Bym [21]	$2176 + 1152 = 3328$
Truong [22]	$640 + 1152 = 1792$
Wang-Wang [16]	$1536 + 1152 = 2688$
Ours	$1536 + 1152 = 2688$

Table III presents the computational cost of our scheme along with the related schemes [20], [14], [21], [22], [16] in login and authentication phases. The computational cost of the schemes [20], [14], [21], [22], [16] and our scheme are, respectively, $5t_e + 6t_h$, $6t_e + 8t_h$, $10t_e + 2t_s + 8t_h$, $4t_c + 14t_h$, $6t_e + 16t_h$ and $6t_e + 17t_h$. The scheme [21] has the higher computaional cost as compared to that of ours and does not offer the security features like verifier table, password friendly and timely typo detection. The scheme [20] has the least computation cost; but it suffers from smart card loss attack. The scheme [16] only takes one hash function less than ours; however, it does not provide the security features like password exposure and user anonymity as shown in Table IV. Thus, our scheme satisfies all the security features while others do not as given in Table IV.

TABLE III
COMPUTATION COST

Scheme	User	Server	Sum
Islam [20]	$3t_e + 3t_h$	$2t_e + 3t_h$	$5t_e + 6t_h$
Jiang et al. [14]	$4t_e + 4t_h$	$2t_e + 4t_h$	$6t_e + 8t_h$
Bym [21]	$5t_e + t_s + 5t_h$	$5t_e + t_s + 3t_h$	$10t_e + 2t_s + 8t_h$
Truong [22]	$t_c + 7t_h$	$3t_c + 7t_h$	$4t_c + 14t_h$
Wang-Wang [16]	$3t_e + 9t_h$	$3t_e + 7t_h$	$6t_e + 16t_h$
Ours	$3t_e + 10t_h$	$3t_e + 7t_h$	$6t_e + 17t_h$

t_h: time complexity of hash operation; t_e: time complexity of exponentiation operation; t_s: time complexity of encryption/decryption of symmetric key cryptography; t_c: time complexity of Chebysev polynomial

TABLE IV
SECURITY FEATURES

Security features	[20]	[14]	[21]	[22]	[16]	Ours
Verifier-table	Yes	Yes	No	Yes	Yes	Yes
Password exposure	Yes	No	Yes	No	No	Yes
Password friendly	Yes	Yes	No	Yes	Yes	Yes
Smart card loss attack	No	Yes	Yes	No	Yes	Yes
Known attacks	Yes	Yes	Yes	Yes	Yes	Yes
Provision of key agreement	Yes	Yes	Yes	Yes	Yes	Yes
Timely typo detection	Yes	Yes	No	Yes	Yes	Yes
Clock synchronization	Yes	No	Yes	Yes	Yes	Yes
Sound repairability	Yes	No	Yes	No	Yes	Yes
Mutual authentication	Yes	Yes	Yes	Yes	Yes	Yes
Forward secrecy	Yes	No	Yes	Yes	Yes	Yes
User anonymity	Yes	No	Yes	No	No	Yes

VI. CONCLUSION

In this paper, we have cryptanalyzed the security of the Wang-Wang's scheme and found that it does not provide user anonymity and suffers from the password exposure attack. We have improved this scheme by overcoming its limitations. Further, we have shown that our scheme is more secured than the existing schemes.

REFERENCES

[1] L. Lamport, "Password authentication with insecure communication," *Communications of the ACM*, vol. 24, no. 11, pp. 770–772, 1981.

[2] H.-Y. Chien, J.-K. Jan, and Y.-M. Tseng, "An efficient and practical solution to remote authentication: smart card," *Computers & Security*, vol. 21, no. 4, pp. 372–375, 2002.

[3] E.-J. Yoon, E.-K. Ryu, and K.-Y. Yoo, "Further improvement of an efficient password based remote user authentication scheme using smart cards," *IEEE Transactions on Consumer Electronics*, vol. 50, no. 2, pp. 612–614, 2004.

[4] C.-I. Fan, Y.-C. Chan, and Z.-K. Zhang, "Robust remote authentication scheme with smart cards," *Computers & Security*, vol. 24, no. 8, pp. 619–628, 2005.

[5] X.-M. Wang, W.-F. Zhang, J.-S. Zhang, and M. K. Khan, "Cryptanalysis and improvement on two efficient remote user authentication scheme using smart cards," *Computer Standards & Interfaces*, vol. 29, no. 5, pp. 507–512, 2007.

[6] K.-H. Yeh, C. Su, N.-W. Lo, Y. Li, and Y.-X. Hung, "Two robust remote user authentication protocols using smart cards," *Journal of Systems and Software*, vol. 83, no. 12, pp. 2556–2565, 2010.

[7] S. Kumari and M. K. Khan, "Cryptanalysis and improvement of 'a robust smart-card-based remote user password authentication scheme'," *International Journal of Communication Systems*, vol. 27, no. 12, pp. 3939–3955, 2014.

[8] V. Odelu, A. K. Das, M. Wazid, and M. Conti, “Provably secure authenticated key agreement scheme for smart grid,” *IEEE Transactions on Smart Grid*, 2016.

[9] J. Xu, W.-T. Zhu, and D.-G. Feng, “An improved smart card based password authentication scheme with provable security,” *Computer Standards & Interfaces*, vol. 31, no. 4, pp. 723–728, 2009.

[10] S. K. Sood SK, Sarje AK, “An improvement of xu et al.’s authentication scheme using smart cards,” in *Proceedings of The Third Annual ACM Bangalore Conference, Bangalore, Karnataka, India*, pp. 1–5, 2010.

[11] R. Song, “Advanced smart card based password authentication protocol,” *Computer Standards & Interfaces*, vol. 32, no. 5, pp. 321–325, 2010.

[12] W.-B. Horng, C.-P. Lee, and J.-W. Peng, “Security weaknesses of song’s advanced smart card based password authentication protocol,” in *Progress in Informatics and Computing (PIC), 2010 IEEE International Conference on*, vol. 1, pp. 477–480, IEEE, 2010.

[13] B.-L. Chen, W.-C. Kuo, and L.-C. Wuu, “Robust smart-card-based remote user password authentication scheme,” *International Journal of Communication Systems*, vol. 27, no. 2, pp. 377–389, 2014.

[14] Q. Jiang, J. Ma, G. Li, and X. Li, “Improvement of robust smart-card-based password authentication scheme,” *International Journal of Communication Systems*, vol. 28, no. 2, pp. 383–393, 2015.

[15] D. Mishra, A. Chaturvedi, and S. Mukhopadhyay, “Cryptanalysis and improvement of jiang et al.’s smart card based remote user authentication scheme,” *arXiv preprint arXiv:1312.4793*, 2013.

[16] D. Wang and P. Wang, “Two birds with one stone: Two-factor authentication with security beyond conventional bound,” *IEEE Transactions on Dependable and Secure Computing*, 2016.

[17] P. Kocher, J. Jaffe, and B. Jun, “Differential power analysis,” in *Annual International Cryptology Conference*, pp. 388–397, Springer, 1999.

[18] T. S. Messerges, E. A. Dabbish, and R. H. Sloan, “Examining smart-card security under the threat of power analysis attacks,” *IEEE transactions on computers*, vol. 51, no. 5, pp. 541–552, 2002.

[19] A. K. Das, “A secure and robust temporal credential-based three-factor user authentication scheme for wireless sensor networks,” *Peer-to-peer Networking and Applications*, vol. 9, no. 1, pp. 223–244, 2016.

[20] S. Islam, “Design and analysis of an improved smartcard-based remote user password authentication scheme,” *International Journal of Communication Systems*, vol. 29, no. 11, pp. 1708–1719, 2016.

[21] J. W. Byun, “Privacy preserving smartcard-based authentication system with provable security,” *Security and Communication Networks*, vol. 8, no. 17, pp. 3028–3044, 2015.

[22] T.-T. Truong, M.-T. Tran, A.-D. Duong, and I. Echizen, “Chaotic chebyshev polynomials based remote user authentication scheme in client-server environment,” in *IFIP International Information Security Conference*, pp. 479–494, Springer, 2015.

Devender Kumar received his M.Sc. (Mathematics) from Panjab University, Chandigarh and M.Tech. (Computer Science and Engineering) from IIT, Madras. Currently, he is an Assistant Professor in Netaji Subhas Institute of Technology, New Delhi and pursuing his Ph.D. in Information Technology from University of Delhi. His current research interests include cryptography and network security.

Satish Chand did his M.Sc. (Mathematics) from IIT, Kanpur and M.Tech. (Comp. Sc.) from IIT, Kharagpur and Ph.D. (Comp. Sc.) from Jawaharlal Lal Nehru University, Delhi. Currently he is a Professor in School of Computer and Systems Sciences, Jawaharlal Nehru University, New Delhi. He is Director, placement cell, Jawaharlal Nehru University, besides being a member in several other committees. He has been made Chairman, All India Board of Information Technology, by AICTE. He has credit to publish about hundred research papers in international/national journals and conferences of high repute. He works in different areas that include video processing, image processing, video broadcasting, Digital steganography, Image forensics, wireless sensor networks, network security, etc.

Bijendra Kumar did his Bachelor of Engineering from H.B.T.I. Kanpur, Uttar Pradesh, India and his Ph.D. from University of Delhi, New Delhi, India. Presently, he is a Professor in Division of Computer Engineering at Netaji Subhas Institute of Technology, New Delhi, India. His areas of research interests are Video applications, cryptography, watermarking, and design of algorithms.

Cost-Efficient Resource Allocation Method for Heterogeneous Cloud Environments

Marton Szabo[1], David Hajay[2] and Mark Szalayz[3]

***Abstract*—In this paper we present a novel on-line NFV (Network Function Virtualization) orchestration algorithm for edge computing infrastructure providers that operate in a heterogeneous cloud environment. The goal of our algorithm is to minimize the usage of computing resources which are offered by a public cloud provider (e.g., Amazon Web Services), while fulfilling the required networking related constraints (latency, bandwidth) of the services to be deployed. We propose a reference network architecture which acts as a test environment for the evaluation of our algorithm. During the measurements, we compare our results to the optimal solution provided by an ILP-based solver.**

***Index Terms*—orchestration; network algorithm; heterogeneous cloud, fog computing, cloud computing**

I. Introduction

In the field of telecommunications many new emerging trends can be observed. For example, IoT (Internet of Things) aims to make traditional devices smart and connected to the Internet; this is a major trend already present nowadays. In the field of transportation, we can also find many exciting new solutions (e.g., remote driving, autonomous drones, etc.). The appearance of 5G networks is expected to enable even more revolutionary services to be built [1]. These can be for example the tactile Internet and on-line augmented reality applications, where the low response time is a crucial prerequisite. These services require not only the evolution of the radio interface, but also necessitates certain modifications in the topology of the back-haul network in order to serve the large number of new devices and the network traffic generated by them, and provide near real-time response times.

Today’s widely deployed telecommunications networks are not flexible enough to fulfill these expected new challenges. For example, running network functions are currently binded to the special purpose hardware elements located in the core of the network (e.g. firewalls, carrier grade NAT platforms), which means unbearably high latency for most of the new applications. The NFV (Network Function Virtualization) concept aims to overcome this challenge [2], [3]: virtualization of the network functions makes it possible to run these services on general purpose hardware (e.g., x86 based servers with high compute capacity), thus removing the limitations coming from the physical location of the devices. The virtual network functions are expected to be one the fundamental building blocks of the future 5G networks.

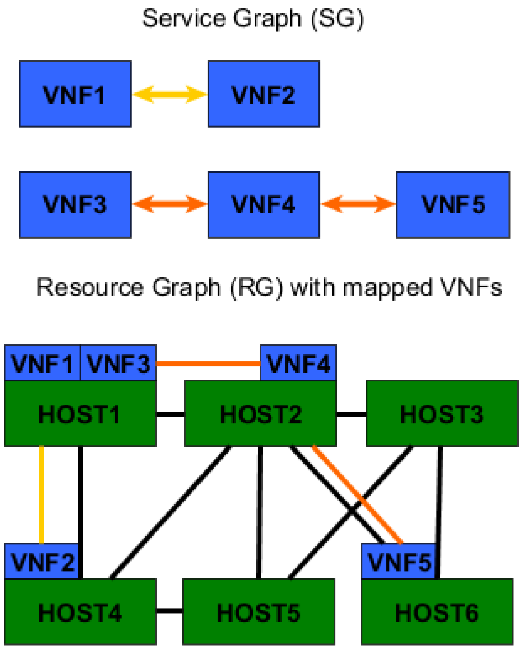


Figure 1. Mapping Service Function Chain to the infrastructure

Furthermore, by extending the traditional cloud concept with compute nodes at edge of the network – often called Mobile Edge Computing (MEC) [4] – using together with the high amount of resources in the core data centers enables many new applications for the service providers. This way, it is possible to run certain network functions near to the end-users with very low latency guaranteed, while other components of a service – that are not so sensitive to latency – can be deployed in core data centers instead of placing those in the limited capacity edge nodes [5]. One service consists of elementary functions connected with each other in a given order. This is called Service Function Chaining (SFC), and its model defines different requirements to the underlying network and virtualization environments (required CPU, RAM, storage, constraints on bandwidth, latency between nodes) [6]. The process that maps multiple service graphs (SGs) composed of different virtual network functions (VNFs) to a common physical infrastructure, represented by the Resource Graph (RG), is called Virtual Network Embedding (VNE). An example of the placement of VNFs and logical connections to the nodes and links of the physical infrastructure is shown in Fig. 1.

Department of Telecommunications and Media Informatics, Budapest University of Technology and Economics, Hungary, 1117 Budapest, Magyar Tudosok krt. 2.

[1] szabo.marton@tmit.bme.hu, [2] yhaja.david@tmit.bme.hu,
[3] zmark.szalay@tmit.bme.hu

DOI: 10.36244/ICJ.2018.1.3

By applying the previously described technologies together with dynamically reconfigurable, software-based networks (Software Defined Networks, SDN), limitations caused by the current rigid network architectures can be eliminated, thus making the introduction of the new-generation network services possible. These can be for example remote driving cars and industrial robots controlled from the cloud, edge content caching, smart cities or on-line augmented reality applications.

An other interesting aspect of the future 5G networks is the resource sharing between different service providers, which would enable their users to be served independently of their actual physical location (e.g., in case of roaming). In such a multi-provider cloud environment the goal of the participants is to utilize their own infrastructure the most efficiently, thus minimizing the expenses caused by using external resources.

In this paper, we propose a new online resource orchestration algorithm which finds proper placement for the network functions of online services while minimizing the costs to be paid for external resources taken at third-party infrastructure providers. In order to evaluate our algorithm, we implemented a framework where we tested its performance in various simulation scenarios. We compare our results to an ILP-provided optimal offline solution.

The paper is organized as follows: Sec. II overviews the related work. Sec. III describes our reference architecture and the the optimization problem in the form of an ILP to be solved. Our online heuristic algorithm is explained in Sec. IV. Performance measurements are evaluated in Sec. V. We conclude our work in Sec. VI.

II. Related Work

Virtual Network Embedding (VNE) is known to be NP-hard [7], which means finding the optimal solution cannot be done within reasonable time in case of large input, for example, when many services are deployed in a large infrastructure. Two different approaches exist to solve the VNE problem: i) exact solutions that find the optimum but these can be applied to limited scale problems, ii) approximation-based algorithms that trade the optimal solution for better runtime. [8] summarizes many of the possible solutions to the VNE problem.

Several approaches use Integer Linear Programming (ILP) to solve the VNE problem. In [9] the authors implemented an ILP formula to minimize the cost of embedding in terms of edge costs while maximizing the acceptance ratio. Reconfiguration of the existing mapping by enabling VNF migrations formed as MILP (Mixed ILP) were studied in [10].

Many different approaches solve the VNE problem with heuristic algorithms. Most of them perform the mapping in two steps: node mapping stage and edge mapping stage, thus physical nodes that have been selected to host neighboring network functions in the node mapping state may be multiple hops away from each other. Many algorithms aim to solve this problem by minimizing link utilization, e.g., [11], [12]. Authors of [13] proposed a hybrid algorithm, which first solves a relaxation of the original problem by using linear programming in polynomial time. Then they use deterministic and randomized rounding techniques on the solution of the linear program to approximate the values of the variables in the original MILP. A decomposing mapping algorithm proposed in [14] aims to minimize the mapping cost by making a selection of the available decompositions during the node mapping stage.

By Edge Computing, we mean a new network functionality, that extends the traditional cloud computing paradigm with additional computing capacity placed close to the end users. These resources are distributed in the service provider's network edge, for example co-located with an Internet Edge PoP (Point of Presence). This new approach makes possible to serve the users at the edge of the network rather than routing over the whole Internet backbone to the data centers located in the core, where all the computing capacity is concentrated. This ensures significant latency reduction and bandwidth savings on the backbone links, thus better QoS (Quality of Service) can be provided. The Open Edge Computing Initiative [15] is the responsible for driving the development of the Edge Computing technology.

The Open Fog Consortium [16] has proposed a possible reference architecture for the 5G ecosystem, called Fog Computing. The architecture [17] can be divided into three main layers. The top layer includes the central clouds that can be either the ISP's own private cloud or a public cloud provider's (e.g., Amazon Web Services, Microsoft Azure) infrastructure. The Fog Nodes, which can be found distributed in the ISP's network are located in the middle layer. They have less computing capacity compared to the previous layer, but can host applications with strict requirement on response time. While having limited resources, Fog Nodes can be used to enhance the performance of the end devices, or to offload the computation intensive tasks from them, thus ensuring better battery lifetime and response times. The bottom layer hosts the end-devices that consume the compute and network resources of the ISP. The devices are usually connected to the network via wireless interface. Further features of the end equipment are location-independence, limited hardware resources and large quantity.

Fog Computing also defines an ideal architecture for one of today's most important emerging paradigm, which is the IoT (Internet of Things) [18], [19]. In IoT, sensor devices in the bottom layer usually monitor different environmental variables, then send the measurement data to a central entity located in the network. This entity may send control messages back to the devices in order to change their state, then it aggregates and transmits the data to an other unit, that for example stores the data for later data processing and big-data applications. This other unit may be suitable to be placed in the central cloud infrastructure, because of the high storage requirements and the computational intensive data processing methods that can be performed more efficiently there. Other

Fog Computing related use-cases and challenges, for example from security point of view can be read in [20] and [21].

III. HETEROGENEOUS NETWORKS

By heterogeneous networks we mean an infrastructure, where different service providers are present. In this section, we introduce our network model based on Fog/Edge Computing, then we give the formal statement of the emerging resource allocation problem in heterogeneous networks.

A. Reference network

Based on the previously described considerations we created our own simplified network model for the three-tier Fog Computing architecture: a network consists of a given number of Fog Nodes (or Edge Computing Clusters) and Central Clouds. Each Fog Node contains a random number of servers with given computing capabilities (CPU, RAM and storage) and two gateway nodes. Each Fog Node has a SAP (Service Access Point) attached to it via the SAP-Gateway. The SAP represents the connection point to the network from the perspective of the end devices. They can reach the network resources through this interface (the SAP can be understood as for example a mobile base station). Within a Fog Node the servers are connected in a full mesh topology. We assume that bandwidth is not the bottleneck therein, because the nodes that belong to the same Fog Node are located close to each other and the blocking-less feature can be provided by choosing the right data center topology. The Fog Nodes and the central data centers are interconnected with each other via the Core Network. Each link has its custom delay and bandwidth characteristics assigned to it. The central cloud can be hosted by a public infrastructure provider (e.g., Amazon Web Services, Microsoft Azure) or the ISP's own data center. A topology may contain any number of central clouds. We assume that they have unlimited compute, storage and memory capacity, but the service provider may needs to pay a fee for the consumed resources. Fig. 2 shows an example topology with four fog clusters and one central cloud node.

B. Problem statement

We are searching for a solution to the following problem: How can we deploy service chains to a previously described heterogeneous cloud environment in a cost effective way? Let us assume that we are an ISP with Fog Nodes scattered around our network with given computing capabilities. Furthermore, we have access to one or more public cloud provider's infrastructure through the Internet. Because of economic reasons, it may be beneficial to have a contract with more than one provider, thus better prices can be achieved for the allocation. We expose our network to the end-users, who can then initiate SFC deployments with various QoS requirements. In this case, an efficient algorithm, which allocates physical resources to the components of the SFC, is necessary. The goal of the algorithm is to minimize the cost to be payed for consuming external resources by fulfilling the QoS requirements and other constraints that are dictated by the capabilities of the network. Finding an optimal solution is known to be NP-hard as this problem can be treated as a generalized version of the previously described VNE problem (arbitrary resource cost assigned to each node).

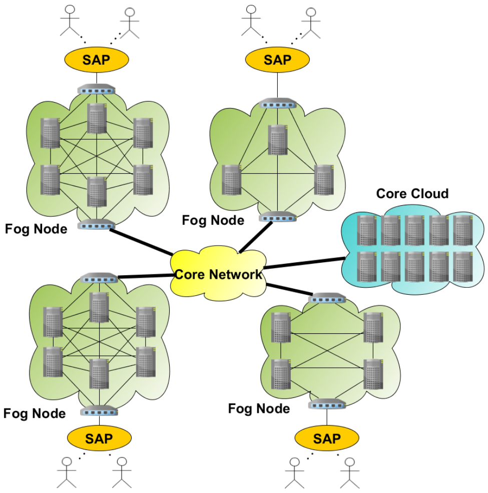


Figure 2. Example to the modeled architecture

Table I
NOTATIONS USED

Notation	Description
V_s, E_s	NFs and links of the service graph
V_r, E_r	Nodes and links of the resource graph
$(i,j) \in E_s$	SG link between NF i and j
$(u,v) \in E_r$	RG link between node u and v
$x_u^i : i \in V_s, u \in V_r$	1 if NF i has been mapped to node u
$y_{u,v}^{i,j}$	1 if (u,v) is in the physical path of (i,j)
$\vec{r_i}$	Resources required by NF i
$\vec{\rho_j}$	Available resources in Node j
$\delta_{u,v}$	Delay of physical link (u,v)
$d^{i,j}$	Maximal delay between NF i and j
$\beta_{u,v}$	Available bandwidth on (u,v) link
$b^{i,j}$	Required bandwidth between NF i and j
$\varrho_s \subset V_s$	SAPs in the SG
$\varrho_r \subset V_r$	SAPs in the RG
$\varrho_s \subseteq \varrho_r$	SG sAPs can be found in the RG also
c_u^i	The cost of running NF i on Node u
$\gamma \subset V_r$	Core Cloud nodes

Equations 1-8 describe the problem as an ILP in the following section. The notations we use in the formal problem description are summarized in Table I. We provide the intuitive meaning of each line of the ILP in the following.

$$\forall i \in V_s : \sum_{u \in V_r} x_u^i = 1 \tag{1}$$

$$\forall (i,j) \in E_s, \forall u \in V_r :$$
$$\sum_{v:(u \to v) \in V_r} y_{u,v}^{i,j} - \sum_{w:(w \to u) \in V_r} y_{w,u}^{i,j} = x_u^i - x_u^j \tag{2}$$

$$\forall u \in V_r, \forall i \in V_s : \sum_{i \in V_s} x_u^i \vec{r_i} \leq \vec{\rho_u} \tag{3}$$

$$\forall (i,j) \in E_s : \sum_{(u,v) \in E_r} y_{u,v}^{i,j} \delta_{u,v} \leq d^{i,j} \tag{4}$$

$$\forall (u,v) \in E_r : \sum_{(i,j) \in E_s} y_{u,v}^{i,j} b^{i,j} \leq \beta_{u,v} \tag{5}$$

$$\forall i \in \varrho_s, : x_i^i = 1 \tag{6}$$

$$\min \sum_{u \in V_r} \sum_{i \in V_s} x_u^i c_u^i \tag{7}$$

As the result of the mapping, each VNF is assigned to exactly one physical node (1). The flow constraint is given by (2). The total amount of resources required by the VNFs mapped to a given node cannot exceed the resources available at the node (3). The total delay on the physical path of a SG link cannot exceed the requested delay between the two VNFs (4). Similarly, the total bandwidth of virtual links mapped to the same physical link cannot be greater than the available bandwidth (5). The SG SAPs are mapped to the RG SAPs with the same ID (6). The objective function of the optimization is to minimize the external resource costs (7). The cost of deploying a VNF is calculated based on the formula:

$$c_u^i = \alpha_u * CPU^i + \beta_u * RAM^i + \gamma_u * BW^i + \delta_u * STR^i, \tag{8}$$

where α, β, γ and δ are the cost parameters of the given physical node. The CPU, RAM, BW and STR are the resources requested by the VNF, where BW is the sum bandwidth on all links connected to the VNF, and STR is the allocated storage. In that case, when a node belongs to a fog nodes, we set the parameters to 0, thus ensuring that using our own infrastructure does not imply any cost. However, in real cloud environments currently Virtual machines usually created by using instance types with predefined CPU and memory resources and the additional bandwidth costs calculated based on the data volume moved to/from the cloud. In our model, we are using a more granular cost function. We entered the problem to an ILP solver by customizing the program that was used in [22] with our own cost calculation method.

IV. OUR ONLINE ALGORITHM

In this section we propose a novel orchestration algorithm that aims to minimize resource usage from the external core clouds, thus utilizing our own infrastructure in the most efficient way. VNF migration is an important feature of our approach, which gives option to migrate a given set of network functions to the cloud, thus freeing up network resources in the fog nodes in order to serve more latency and bandwidth sensitive requests. We introduce the $migrationcost$ attribute in (9):

$$migrationcost^i = \min \lambda_u * STR^i, u \in \gamma, \tag{9}$$

as there is a cost penalty to be payed for moving VNF_i to the cloud from one of the fogs. The rationale is that a migration process requires redundant resources to be allocated caused by the duplicating the state of the network function to be transfered to the cloud. We derive this cost from the migration coefficient of a node (λ) and the allocated storage to VNF^i. The main steps of our algorithm taking into account migration costs are described in the following, and pseudo-code is provided in Alg. 1 and 2.

A. SG preprocessing

In the first step of our algorithm, we calculate the order of execution. The ORDERSUBCHAINS method splits the incoming service request to the list of triplets containing the links and their connected nodes, i.e., the VNFs. The method starts with the first available SAP and collects all the neighboring nodes and connected edges, then appends the link with the strictest bandwidth requirement to the list together with its endpoints. After that it collects the available nodes and edges that became reachable via the new node.

B. VNF mapping

The next step step is the mapping of the service requests to the physical infrastructure. The MAP method iterates trough the previously ordered list of edges. Depending on the status of the nodes connected by the link, three different cases are possible. If both ends have already been allocated to a computing resource previously, then only a suitable path for the virtual edge needs to be found. To achieve this we run a Dijkstra algorithm between the hosts in the physical topology.

If one of the end nodes is not in mapped_vnodes yet and it is not a SAP either, then the node needs to be mapped. In the MAPVNF method the program tries to find a suitable place for the VNF. First, it filters the available physical nodes based on computing resources, and after that it checks if the candidate is reachable from the previous node via any sequence of edges. If the path does not satisfy the delay requirement, or any of the edges does not have enough free bandwidth, then the node is removed from the list of candidates. When the list of compatible physical nodes is available, they are sorted based on the resource cost of hosting the actual VNF. After the host node is determined, the link can also be mapped with the previously seen method. In that case, when the actual element is a SAP, then the algorithm calculates the path with the lowest latency, where the required bandwidth is available on all edges. If the path fulfills the latency requirement between the previously mapped VNF and the SAP, then the link mapping is performed.

It may occur, that one of the steps above fails. For example, none of the nodes have enough resource to host a given VNF, or the network related requirements cannot be met. In that case, the algorithm tries to step back to a previous state. This step is performed by the ROLLBACK method. In order to ensure better runtime, limiting the number of rollback steps may be necessary. We can do that by setting the max_rollback constant to an appropriate value. The ROLLBACK method

Algorithm 1 Service graph mapping to resource graph

```
$running \leftarrow$ copy($RG$)
$mapped\_vnodes \leftarrow \emptyset$
$map\_list \leftarrow$ORDERSUBCHAINS($SG$)
$mapped\_vnodes.insert(\varrho_s.first)$
$rollback\_level$ = 0
for all $(u, v, link) \in map\_list$ do
    if $(u, v) \in mapped\_vnodes$ then
        $success \leftarrow$ MAPVIRTUALLINK($link$)
    else if $u \notin \varrho_s$ then
        $success \leftarrow$ MAPVNF($u, v, link$)
    else                ▷ This means actual_element is a SAP
        $success \leftarrow$ MAPVLINK2SAP($u, v, link$)
    end if
    if $\neg success$ and $rb\_level \geq max\_rb$ then
        $success \leftarrow$ MIGRATINGEDGE2CORE($cable, u, v$)
    else
        $success \leftarrow$ ROLLBACK($u, v, link$)
        $rollback\_level+ = 1$
    end if
    if $success$ then
        $mapped\_vnodes.insert(v)$
    end if
end fordone
```

Algorithm 2 VNF migration from fog to cloud

```
procedure MIGRATINGEDGE2CORE($cable, u, v$)
    $compatible\_fogs \leftarrow \emptyset$
    $fog\_list \leftarrow$ GETFOGSFROMVNFS($cable$)
    for fog $\in$ fog_list do
        if $fog$ == $u.fog$ then
            $compatible\_fogs$.insert($fog$)
        else
            $vlink \leftarrow$ GETVLINK($u, v$)
            $phy\_path \leftarrow$ GETPATH($u.fog,\ fog$)
            if CHECKPATH($phy\_path, vlink$) then
                $compatible\_fogs$.insert($fog$)
            end if
        end if
    end for
    DELUNCOMPVNFS($cable, compatible\_fogs$)
    $possible\_nodes \leftarrow$ GETPHYNODES($cable$)
    $mig\_list \leftarrow$ ordered empty list
    for node $\in$ possible_nodes do
        $vnf\_list \leftarrow$ ordered empty list
        for vnf $\in$ node.corable_vnfs do
            if ISMIGRABLE($vnf$) then
                $vnf.mig\_cost \leftarrow$ GETMIGCOST($vnf$)
                $mig\_list$.add($vnf$)
            end if
        end for
        EXPANDMIGLIST($mig\_list, node.cable\_vnfs$)
    end for
    for mig_opt $\in$ mig_list do
        DOMIGRATING($mig\_opt$)
        if ISMIGWASSUC($mig\_opt$) then
            return True
        else
            RESTOREMIG($mig\_opt$)
        end if
    end for
    return False
end procedure
```

restores the state when the previous VNF was mapped, then chooses an other candidate from the list of the suitable nodes, and continues the mapping from the modified state. If the number of rollbacks exceeds the limit, then the algorithm tries to migrate one or more already mapped VNFs to the central cloud, thus freeing up resources in the fog nodes. The migration process is described in the next section.

C. Migrating VNFs to the cloud

The method that performs the migration can be divided into three parts. The first part collects the possible fog nodes containing VNFs that can be moved to the central cloud without violating the latency and bandwidth constraints. The GETFOGSFROMVNFS method returns the fog nodes which contain movable VNFs. If the fog node of the previously mapped VNFs is in the result, then that fog node is surely suitable to host the actual VNF in terms of network related constraints. In other cases, the GETPATH method collects the sequence of the physical links between the host of the previous VNF and the actual fog node. After that, if the performance of the physical path is conform with the virtual link requirements, then the fog node is stored as a possible element.

The second part of the method determines the migration options from the fog nodes selected previously. The DELUNCOMPVNF method removes the VNFs that belong to incompatible fog nodes from the list containing movable VNFs. After that, iterating through the list of the physical nodes that have movable VNFs on it, if by migrating the VNF from a node enough resources can be freed up, then we calculate the migration cost of the VNF and insert it to the migration list. The migration list is ordered by the migration cost of the contained elements. It may occur that by moving only one VNF to the cloud at a time the required amount of resources cannot be provided to the new NF. Because of that, checking different subset of VNFs mapped to the same physical node may be necessary. Instead of checking all the possible subsets, in order to reduce runtime we only test the neighboring pairs, triplets, and so on in the EXPANDMIGLIST method. If any of these subsets grants enough free resources, then we insert it to the migration list with the total migration cost of all the affected VNFs. The result of the process is an ordered list that contains the possible migrating options.

In the last step, the DOMIGRATING method tries to execute the migrating options from the migration list, starting with the cheapest option. The ISMIGRATING method checks in each iteration if the migration was successful. If the remapped physical paths fulfill the corresponding virtual link requirements, then the method returns true. Else in RESTOREMIG we restore the previous state and continue with the next migration option from the list.

V. MEASUREMENT RESULTS

We have run measurements to demonstrate how our algorithm works in different topology setups with various requests. We had six scenarios each of them contained different number of fog nodes between 1 and 20. During our simulations we posted the requests (the service graphs) one by one, till the first failure occurred. We indicated failure when there was no way to deploy a service graph completely to the remaining available resource set.

First of all, we compared our algorithm's efficiency with and without the migration function. In the comparison we examined how many CPUs our algorithm can deploy to the same resource set. It is easy to see that migrating is mostly used for cost efficiency, because if we turn off this feature then our algorithm has two options for deploying VNFs. First option: our algorithm deploys cloud compatible VNFs directly in the cloud. This is obviously not cost efficient because we have to pay for allocated resources to the third-party from the beginning. Second option: we want our algorithm to be cost efficient, so it puts all the VNFs in the fog nodes. It is definitely cheap for us, however our resources in the fog nodes will quickly get exhausted.

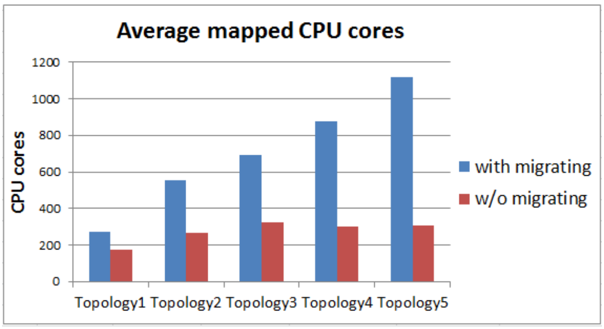


Figure 3. CPUs mapped with and without migration

Fig. 3 shows how many virtual CPU our algorithm can deploy with migrating and without migrating them if we want to keep our cost efficiency. Each bar on the figure represents the average number of mapped virtual CPUs on the quoted topologies. As you can see, we deploy more virtual CPUs when the migration feature is turned on.

Next, we wanted to examine what the cost difference between our algorithm and an optimal solution is. The optimal solution comes from the offline algorithm, which solves the previously defined ILP problem. Both algorithms were used with the same input, which contained the same resource graph and the same set of request graphs. We executed our algorithm and collected all the successfully deployed requests till the first failure. After that, we executed the offline algorithm with the same resource graph and the collected request set. Both algorithms return with a number that defines the price of the request set on the given topology. We compared these outputs and plotted those in a scatter plot shown in Fig. 4: each point represents a cost difference for a given request sequence. It is noticeable how the request randomization affects the deployed CPU number. It is also remarkable how our algorithm scales: with the increase of the number of fog nodes the difference between the two solutions does not go higher than 20%.

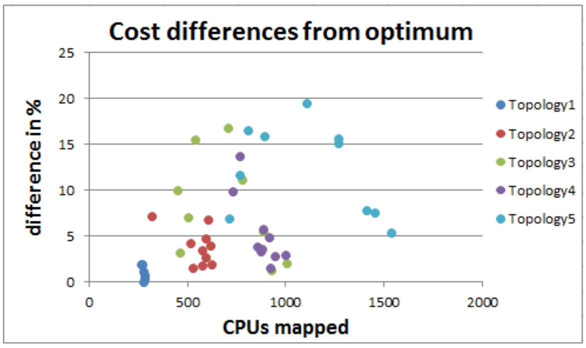


Figure 4. Differences of cost between the online and offline algorithms

Table II
RELATIVE STANDARD DEVIATION OF COSTS

	Mean of cost differences	Relative Standard Dev.
Topology1	0,9841	0,7931
Topology2	3,8272	0,5235
Topology3	8,0787	0,7032
Topology4	5,2071	0,7231
Topology5	12,1877	0,4061

Table II shows the calculated relative standard deviation based on the results displayed in Fig. 4. It tells us how the different results in each measurement are scattered around the mean cost. We can observe that as we increase the number of fog nodes significantly, the relative standard deviation starts to decrease, which also a proof of the good scalability.

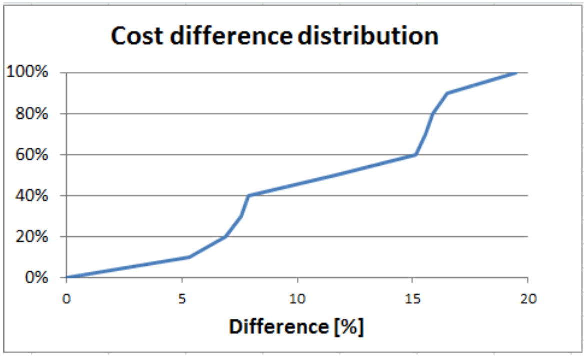


Figure 5. Cost difference distribution

In Fig. 5 we depict the distribution of cost differences. The x axis represents the cost difference (in percentage) between the offline and our algorithm. The optimal solution contains

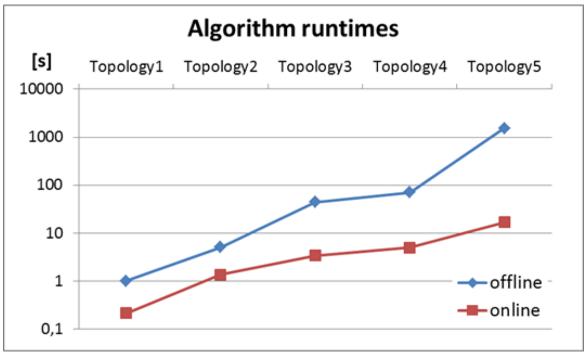


Figure 6. Runtimes of the algorithms

only the price of the allocated resource in the third-party environment (cloud). In our solution we calculate a migration cost as well. It is remarkable how the migration cost affects our final price. If the migration cost is higher than the resource allocation cost, then it is worthy to put a cloud compatible VNF directly to the core cloud, so we can avoid high migration cost.

Finally, we compared the execution times for both algorithms. The result is represented in Fig. 6. The offline algorithm's execution time is increasing exponentially with the number of fog nodes, however our algorithm execution time grows only linearly.

VI. Conclusion

In this work we examined how future networks should handle IoT services and their deployment. We made network topology examples that could be used to run such services. For handling the service creation on these topologies we created an online algorithm that can deploy services cost efficiently and that is also able to handle networking requirements of the services. Our algorithm runs in polynomial time, thus scales as well, while it provides close-to-optimal orchestration results.

References

[1] N. Panwar *et al.*, "A Survey on 5G," *Phys. Commun.*, vol. 18, no. P2, pp. 64–84, Mar. 2016. [Online]. Available: http://dx.doi.org/10.1016/j.phycom.2015.10.006

[2] ETSI, "White Paper: Network Functions Virtualisation (NFV)," 2013. [Online]. Available: http://portal.etsi.org/nfv/nfv_white_paper2.pdf

[3] ETSI GS NFV-PER 001, Dec 2014, *Network Functions Virtualisation (NFV); Architectural Framework*, ETSI, 2014 Dec.

[4] G. Brown, "Mobil edge computing use cases and deployment options," Tech. Rep., 2016.03.01. [Online]. Available: https://www.juniper.net/assets/us/en/local/pdf/whitepapers/2000642-en.pdf

[5] P. Mach *et al.*, "Mobile edge computing: A survey on architecture and computation offloading," *IEEE Communications Surveys and Tutorials*, vol. 19, no. 3, pp. 1628–1656, 2017.

[6] J. Halpern *et al.*, "Service Function Chaining (SFC) Architecture," IETF RFC 7665, Oct. 2015.

[7] E. Amaldi *et al.*, "On the computational complexity of the virtual network embedding problem," *Electronic Notes in Discrete Mathematics*, vol. 52, pp. 213 – 220, 2016, INOC 2015 – 7th International Network Optimization Conference.

[8] A. Fischer *et al.*, "Virtual network embedding: A survey," *IEEE Communications Surveys and Tutorials*, vol. 15, no. 4, pp. 1888–1906, 2013.

[9] I. Houidi *et al.*, "Virtual network provisioning across multiple substrate networks," *Computer Networks*, vol. 55, no. 4, pp. 1011–1023, 2011.

[10] G. Schaffrath *et al.*, "Optimizing long-lived cloudnets with migrations," in *UCC '12, ACM*. ACM, 2012, pp. 99–106.

[11] H. Cui *et al.*, "A virtual network embedding algorithm based on virtual topology connection feature," *Wireless Personal Multimedia Communications*, 2013.

[12] G. Wang *et al.*, "A virtual network embedding algorithm based on mapping tree," *Communications and Information Technologies*, 2013.

[13] M. Chowdhury *et al.*, "Vineyard: Virtual network embedding algorithms with coordinated node and link mapping," *IEEE/ACM Transactions on Networking*, vol. 20, no. 1, pp. 206–219, 2012.

[14] S. Sahhaf *et al.*, "Network service chaining with optimized network function embedding supporting service decompositions," *Computer Networks*, vol. 93, no. 3, pp. 492–505, 2015.

[15] Open Edge Computing Initiative. [Online]. Available: http://openedgecomputing.org

[16] OpenFog Consortium, "Openfog reference architecture for fog computing," Tech. Rep., 2017.02.08. [Online]. Available: https://www.openfogconsortium.org/wp-content/uploads/OpenFog_Reference_Architecture_2_09_17-FINAL.pdf

[17] C. C. Byers, "Architectural imperatives for fog computing: Use cases, requirements, and architectural techniques for fog-enabled iot networks," *IEEE Communications Magazine*, vol. 55, no. 8, pp. 14–20, 2017.

[18] A. V. Dastjerdi *et al.*, "Fog computing: Helping the internet of things realize its potential," *IEEE Computer*, vol. 49, no. 8, pp. 112–116, 2016.

[19] F. Bonomi *et al.*, "Fog computing and its role in the internet of things," *MCC '12*, pp. 13–16, 2012.

[20] S. Yi *et al.*, "A survey of fog computing: Concepts, applications and issues," in *Mobildata '15, ACM*. ACM, 2015, pp. 37–42.

[21] L. M. Vaquero *et al.*, "Finding your way in the fog: Towards a comprehensive definition of fog computing," *ACM SIGCOMM Computer Communication Review*, vol. 44, no. 5, pp. 27–32, 2014.

[22] B. Németh *et al.*, "Efficient Service Graph Embedding: A Practical Approach," in *Second IEEE International Workshop on Orchestration for Software Defined Infrastructures (O4SDI @ IEEE NFV-SDN 2016)*, Nov 2016.

Marton Szabo received his M.Sc. degree in informatics from Budapest University of Technology and Economics (BME) in 2017. During his studies he was a member of the High Speed Networks Laboratory hosted by the Department of Telecommunications and Media Informatics. After working in the telecom industry for one year he joined the newly formed MTA-BME Network Softwarization research group founded by the Hungarian Academy of Sciences (MTA). His current research focuses on network softwarization and virtualization in 5G.

Mark Szalay is a graduate student at Budapest University of Technology and Economics. He is a member of the High Speed Network Laboratory (http://hsnlab.tmit.bme.hu/) in the Department of Telecommunications and Media Informatics. His main research interests include Hardware (Router/switch/NIC) design, Network programming, Software-Defined Networking (SDN) and Network Function Virtualization (NFV). Mark has been working in this field for more than two years.

David Haja is an undergraduate student at Budapest University of Technology and Economics. He is a member of the High Speed Network Laboratory (http://hsnlab.tmit.bme.hu/) in the Department of Telecommunications and Media Informatics. His main research interests include Software-Defined Networking (SDN), Network Function Virtualization (NFV) and Resource Orchestrartion.

The Evolution of Free-Space Quantum Key Distribution

Tamas Bisztray and Laszlo Bacsardi

***Abstract*—In this paper we are looking at the milestones that were achieved in free−space quantum key distribution as well as the current state of this technology. First a brief overview introduces the technical prerequisites that will help to better understand the rest of the paper. After looking into the first successful demonstrations of short range free space QKD both indoor and outdoor, we are examining the longer range terrestrial QKD experiments. In the next step we look at some experiments that were aiming to take free space QKD to the next level by placing the sender or the receiver on moving vehicles. After the terrestrial demonstrations we focus on satellite based experiments. Finally, we explore hyper-dimensional QKD, utilising energy−time, polarization and orbital angular momentum (OAM) degrees of freedom.**

***Index Terms*—free−space quantum communication, quantum key distribution**

I. Introduction

QUANTUM key distribution (QKD) is an emerging technology which was born to solve one of the biggest information security issues, namely the obsoletion of conventional public key cryptography. Although public key cryptographic primitives such as RSA (Rivest−Shamir−Adleman protocol) and DH (Diffie−Hellman protocol) are still used today, in the near future when quantum computers reach quantum supremacy these protocols will not provide any security since the algorithms to break them are already developed, we are just waiting for quantum computers to catch up. However, information security is not only endangered in the future but in the present as well. Messages that are recorded today can be deciphered when a quantum computer becomes available. Quantum key distribution therefore, should be implemented in advance before quantum computers arrive. But what is exactly quantum key distribution and how does it solve this problem. In short, it replaces the conventional public key algorithms and establishes a perfectly secure secret pair of keys between the communicating parties. This shared secret key than can be used in symmetric cryptographic protocols to encrypt secret messages and this can be sent through a regular channel same as today. The security of this technology is based on the principles of quantum physics, unlike in public key algorithms where the security relies on assumptions such that a certain mathematical problem is hard, like discrete logarithm or prime factoring. There are two major obstacles in the way of implementing QKD on a large scale. The first on is that this technology is expensive and still not perfectly mature and a lot of things needed to be developed, such as quantum memory or quantum repeaters. The second is that in a network we would like to establish secure communication between every node, therefore a secure key−exchange between every node. To achieve this, we need to have a direct cable connection without breakpoints if we want to achieve maximum security. This is not possible in most of the cases since if we would like to describe a network it is usually a k-vertex or edge connected graph so if some parts of the network fail the other nodes can still reach each other. A subgraph of the network might be complete but the whole network is usually not. Moreover, it might not be even possible to establish direct connection between some of the nodes with optical cable, due to geographical separation. This is a major problem that needs a solution. Free space quantum key distribution can be the answer to this challenge, where instead of optical cables all we need is line of sight. The nodes moreover are not fixed to work in one pair. After the secret key is established between two they can turn in other directions. This area is further important for satellite communication which is an absolute necessity for building a global quantum encryption networks. It also has a lot of challenges as the weather or day and night cycles can influence the key exchange rate to name a few.

II. Technical background

In classical information technology information is encoded in bits, zeros and ones. In quantum communication the information is encoded in qubits [1]. These can contain information about zero and one at the same time. An example of such can be photon polarization or magnetic moment. Figure 1. shows a simple visualisation for the following experiment that will help the reader to understand the nature of qubits. Here the line represents the path of the photons (from left to right) while the boxes are the polarizers. In the first experiment we polarize light vertically with the first box. Then we apply another vertical polarizer and what we see is that all of the light comes through and the intensity remains unchanged. In the second experiment we change the second box to a horizontal polarizer, and this time we see that no light comes through. The third time a diagonal polarizer is used that has a 45° angle with the vertical polarizers reference frame. Now we see that light comes through but only with 50% intensity, meaning that for the individual photons there is a 50% chance of passing

T. Bisztray is with the Department of Networked Systems and Services, Budapest University of Technology and Economics. E-mail: t.bisztray@gmail.com L. Bacsárdi is with the Institute of Informatics and Economics, University of Sopron. E-mail: bacsardi@inf.uni-sopron.hu The research was supported by the Hungarian Scientific Research Fund – OTKA PD-112529. The research is connected to COST Action CA15220 Quantum Technologies in Space. The research was further supported by the National Research Development and Innovation Office of Hungary (Project No. 2017-1.2.1-NKP-2017-00001).

DOI: 10.36244/ICJ.2018.1.4

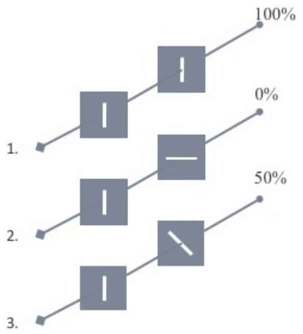


Fig. 1: *Experiment with photon polarization. 1. First light is polarized vertically and then it goes through another vertical polarizer (measurement). Here the light comes through with unchanged intensity. 2. After being vertically polarized now a horizontal measurement is performed. In this case no light comes through. 3. Here the second measurement is diagonal with 50% of the light coming through.*

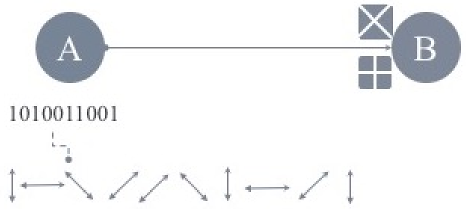


Fig. 2: *Schematic overview of the BB84 protocol. Alice sends random bit values to Bob encoding the information into photon polarization. She is randomly alternating between the rectilinear and diagonal bases as each bit value can be represented in both.* $\longleftrightarrow = \nearrow = 0$ *and* $\updownarrow = \nwarrow = 1$*. Bob is also randomly alternating between the measurement devices when detecting the incoming photons.*

or not passing the second polarizer. To be precise the actual probability of successfully measuring a photon in a β state which was prepared in an α state can be described with the following simple formula

$$|\langle\beta|\alpha\rangle|^2 = cos^2(\Theta),$$

where Θ is the angle difference between the shared reference frame [2]. It is easy to see now that in the first experiment both polarizer was vertical so the angle between them was zero. Thus $cos^2(0) = 1^2 = 1$ which means that the light will pass through with probability 1. The bra–ket $\langle.|.\rangle$ notation is the standard notation for describing quantum states. Ket $|.\rangle$ is a column vector while bra $\langle.|$ is a row vector. This is known as the Dirac notation. Bra and ket are each others Hermitian conjugate.

A. BB84 protocol, a prepare and measure approach

The question is now how can we use this property of light to establish a shared secret between the communicating parties. The BB84 protocol [3] was the first one to implement this. The schematic overview of the following steps is shown in Figure 2.

1) Alice encodes her random sequence of bits in horizontal, vertical, diagonal and anti$-$diagonal polarization where $\longleftrightarrow = \nearrow = 0$ and $\updownarrow = \nwarrow = 1$ (Alternating randomly between the bases)
2) Bob chooses randomly between rectilinear and diagonal measurement bases for each event
3) Bob will have a binary sequence as the result of the measurements. (It might be less then the amount Alice sent)
4) They compare their measurement bases
5) Alice and Bob keep the results only if they used similar bases
6) This binary sequence is called the shifted key

The security of this communication builds on Heisenberg's uncertainty principle which says that measuring one of these properties such as rectilinear or diagonal polarization, randomizes the value of the other property. It can be also proven that both of these measurements cannot be performed at the same time. The eavesdropper has a probabilistic chance to get information on this key but by performing an intermediate measurement on the transmitted information she introduces errors. The first challenge is that the results by default can contain some errors due to the imperfection of physical components, but this error rate can be determined and the protocol should allow to recover from this. The eavesdropper further increases this and to discover her presence Alice and Bob can compare a random subset of they shifted key with the assumption that the errors are evenly distributed. If the error rate is high the whole key is discarded, otherwise Alice and Bob will perform post processing steps to exclude all errors and further reduce the probabilistic information Eve might possess to arrive to their private shared secret. These post processing steps as well as the error estimation is communicated through a conventional public communication channel with a strong assumption on it's authenticity, such that Eve cannot perform a man in the middle attack nor can compromise the integrity of the messages. The second challenge is that it's difficult to produce single photons whose arrival times are not randomly distributed. Therefore, one approach is to use incoherent weak light pulses. The problem with this is that Eve can split the pulse into two or more photons measuring only one and letting the others arrive to Bob. This way she will introduce no additional error and can gain significant amount of partial information. Alice and Bob can estimate Eve's partial information on the string both from the detected error frequency and the optical pulse intensity. The post processing steps therefore must include privacy amplification [4] to further reduce Eve's partial information. During the error correction step which is usually done by LDPC algorithms (low density parity check) the key string is divided into chunks with the assumption that the errors are evenly distributed and each chunk contains only one error with very high probability. The parties compare these parities until they can do 5 round without finding any error. Of course after each round the string is permuted and new chunks are selected. When the parity of two chunks are compared one random bit has to be discarded so Eve cannot gain information on the parities. Unfortunately, this means that a lot of key$-$bits are discarded.

During these steps together with the privacy amplification Eve's actual information on the secret key becomes negligible.

For summary the key that the users get by simply detecting the incoming transmission is called the raw key. This contains detection events even from not compatible bases. Next during the shifting step, when only measurements performed in the compatible bases are kept, is when we acquire the shifted key. If there were no eavesdropper and no errors these would be identical, but even without an eavesdropper this contains errors. The rate of this error is significant not only for eavesdropper detection but if it is high the error correction would take a lot of rounds discarding a lot of key−bits. For this it is important to differentiate between the shifted key and the secret key which we get from the shifted key after the post processing steps.

It is also important to note that over large transmission distances the fibre attenuation can be very significant and this effect cannot be mitigated by amplifying the weak signals since quantum information cannot be amplified and thus no repeaters can be built to prolong the coverable distance.

B. Entanglement based key establishment

There is a spooky quantum physical phenomena called entanglement that can be also used to securely establish a secret key between Alice and Bob [5]. To obtain entangled pairs a strong laser shoots at a nonlinear crystal. This shooting is periodical with a certain pump frequency. The majority of the photons pass through the crystal but some of them undergo spontaneous parametric down conversion and two weaker pulse will leave the crystal. By the law of conservation of energy and the law of conservation of momentum, the pair have combined energies and momenta which is equal to the energy and momentum of the original photon. The laser has to be adjusted such that after the down conversion the average photon number in the weaker pulses must be between 0.1 and 0.5. Depending on the crystal used the correlation between the polarization can be Type I, where the photons share the same polarization or Type II where they have perpendicular polarization. But what do we mean by entanglement? The polarization of pairs is not determined, it is neither horizontal, vertical or in any other well definable state. If we send the pairs far apart from each other and we perform a measurement on one of them, in case of a Type I pair the polarization of the other half will be instantaneously determined to be polarized in the same way as the result of the measurement on the other half. Meaning if we perform the same measurement we will get the same result 100% of the time. The strange thing about this phenomena is that the pairs can be space like separated in a way that when the measurement is performed on one half, the same measurement is performed on the other before light could reach from one half to the other. However, information cannot be transmitted with this method faster than the speed of light since the parties need to communicate to choose the same measurement base in order to extract information out of this phenomena. This entangled state can be written as

$$|\Psi\rangle = \frac{1}{\sqrt{2}} |H\rangle_A |V\rangle_B - \frac{1}{\sqrt{2}} |H\rangle_A |V\rangle_B ,$$

for a Type II correlation with probability amplitudes $\frac{1}{\sqrt{2}}$, meaning there is a 50% chance to measure either horizontal or a vertical result but then the state of the other half is determined to be the opposite.

The photons are now anti−correlated. To test their entanglement, we can preform a Bell−experiment. To understand how that works lets do a game between Alice and Bob. They first get an input from the set $\{0{,}1\}$ (randomly). Then they output 0 or 1 as they wish. Let's call Alice's input A and output a, while Bob's input is B and output is b. The rules are the following:

- If $A \vee B = 0$ they win if $a = b$ // $(A \cdot B = 0) \rightarrow$ win if the answers are correlated
- If $A = B = 1$ they win if $a \neq b$ // $(A \cdot B = 1) \rightarrow$ win if answers are anti−correlated

If they win they get +1 coin if they loose they get nothing. We can write down the expected value of their winnings by looking at the probabilities of correlation

$$S = P_c(A_0B_0) + P_c(A_0B_1) + P_c(A_1B_0) + P_a(A_1B_1).$$

Here A_0 denotes that $A = 0$ and P_c stands for the probability of a correlated answer while P_a is for anti−correlated. Alice and Bob cannot communicate during the game. If they play randomly they win only 50% if the time so they previously agreed on making a 0 output no matter what the input is. Then the previous equation becomes: $S = 1{+}1{+}1{+}0$. From this it is obvious that classically $1 \leq S \leq 3$.

Note: In the most common Bell test called the CHSH inequality they get -1 for an uncorrelated answer. With that the equation is

$$P_c(A_iB_i) = P(a = 1, b = 1|A_iB_i) + P(a = 0, b = 0|A_iB_i)$$
$$P_a(A_iB_i) = P(a = 1, b = 0|A_iB_i) + P(a = 0, b = 1|A_iB_i)$$
$$E(A_iB_i) = P_c(A_iB_i) - P_a(A_iB_i).$$

In this case, the expected winnings are

$$S = E(A_0B_0) + E(A_0B_1) + E(A_1B_0) - E(A_1B_1),$$

with $S \leq 2$.

Alice and Bob would like to do better than that so although they cannot communicate classically they can share a Bell state (Type I). $|\Psi\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$. The measurement they perform on this will be the random input A and B. For Alice $A_0 = 0°, A_1 = 45°$ are the measurement angles, for Bob $B_0 = 22{,}5°, B_1 = -22{,}5°$. Now we would like to get correlated outcomes for A_0B_0, A_1B_0, A_0B_1 but uncorrelated results for A_1B_1. In the cases of A_0B_0, A_1B_0, A_0B_1 the angle is $\pi/8$ between the reference frames. For A_1B_1 it is $3\pi/8$.

In the case : $A \vee B = 0$ they win if $a = b$. According to the experiment in Figure 1, and our formula: $|\langle\beta|\alpha\rangle|^2 = cos^2(\Theta)$, we know that the probability that they win is $cos^2(\frac{\pi}{8})$.

In case of: $A = B = 1$ they win if $a \neq b$. Getting the same result is $cos^2(\frac{3\pi}{8})$. But now they win if the answers are anti−correlated. The probability of that is $1 - cos^2(\frac{3\pi}{8})$ which can be show to be equal to $cos^2(\frac{\pi}{8})$.

This gives

$$S = P_c(A_0B_0) + P_c(A_0B_1) + P_c(A_1B_0) + P_a(A_1B_1) \leq$$

$$\leq 4 \cdot cos^2(\pi/8).$$

Since $cos^2(\pi/8) \approx 0.85$ it follows that $S \leq 3.4$.

If an experiment is conducted and from the collected data we see that the coincidences (winnings) are greater than what a classical experiment would allow and the classical inequality is violated than we can be sure that indeed the incoming photons were entangled since we previously showed that classically $1 \leq S \leq 3$. This requires a large number of measurements with relatively high detection efficiency to successfully determine the violation. In case of the CHSH inequality a value grater than 2 is needed to prove the violation. The most commonly used protocol for entanglement based QKD is the Ekert protocol. In this case the coincident detections where the parties used the same measurement bases is used for the key establishment. In the case of not compatible measurement angles the results are recorder and used for the CHSH inequality or another Bell test.

III. Achievements in the 1990's

The first implementation of a quantum key distribution system that used free air as the optical path took place in 1991 [6]. Here the authors used a prepare and measure protocol to transmit the information from Alice to Bob. The bases that the information is encoded in are the rectilinear basis (horizontal vs vertical polarization) and the circular basis (left circular vs right circular) which can be used instead of the diagonal base in an equivalent way. Incoherent pulses are produced by a green LED that is filtered and directed on a horizontal polarizer. This light is then modulated Pockels Cells −(an optical component that can change the light's polarization direction as a function of applied voltage)− to achieve one of the four polarization states which is then detected by Bob. The intensity of this light is around 0.1 photon per pulse. This disallows the eavesdropper to further split the pulse into more photons. In this scenario the quantum channel was 32 cm free air.

This light intensity was good for demonstration and the given short distance, but such weak pulses would be lost due to noise and channel attenuation over larger distances. The low efficiency of detectors (9%) used in the experiment further limited the key rate of transmission. As a result, over 715.000 pulses were sent and only around 4000 were detected. This means that the laser has fired 715.000 times but due to channel attenuation or because of imperfection in the physical components combined with the low detection efficiency the detectors fired only 4000 times. Approximately half of the detections took place in the correct basis and the process took 10 minutes of real time. Without an eavesdropper the parties ended up with 754 bits of shared secret. With an active eavesdropper this was reduced to 105 bits leaving a lot of room for improvement.

The first demonstration for a successful free space quantum key−exchange in an outdoor environment was published in 1996. [7] The approach was similar to the previous one in terms of information encoding but instead of circular polarization a diagonal base state were used by adding a second Pockels cell. The experiment was conducted under bright daylight conditions over a 75m distance. After traveling through the air the single photon was focused back into an optical fiber. The small fibre diameter $(3 - \mu m)$ limited the angle through light could arrive which prevented background light coupling into the system. Using two silicon avalanche photodiodes with (50%) efficiency the achieved transmission rate was 1 kHz.

The next big step in free space QKD was the experiment done by a group of physicists at University of California, Los Alamos National Laboratory in 1998 [8]. Similarly to the previous experiments a prepare and measure protocol was used namely the B92. The maximal distance achieved was 950m under nighttime. An average photon number of ≤ 0.1 were used per pulse for transmission. The achieved BER with this setup was 1.5% which was lowered to 0.7% at a 240m distance. Here a laser was used to generate a large number of photons (10^5) with a $\sim$ 1-ns optical pulse which was then attenuated in such a way to reach a 2-photon probability of less then 0.5% and this implies that less than 6 of every 100 detectable pulses could contain 2 or more photons. The laser was temperature adjusted to get a wavelength of 772 nm, which is good against depolarizing effects of atmospheric turbulence. On the transmitter side a beam expander is used to magnify the beam that is directed into a telescope in the receiver side. With the transmitter pulsed at a 20 kHz the achieved bitrate was 50 Hz. The authors further argue that this experiment shows the feasibility of a ground station to satellite transmission. They suggested that under nighttime conditions a 35-450 Hz key generation rate is possible. To mitigate the effects of background photons narrow time windows within which we look for the incoming photons are important. To accurately determine the photon arrival time a bright (classical) precursor reference pulse was used which allows the receiver to set a 1-ns time window.

IV. The early 2000's

In 2002 the Los Alamos National Laboratory took their experiment a step further making a quantum key−exchange over a 9.81 km free air channel [9]. The experiment was conducted both during daylight and nighttime conditions. During the day the average photon number (μ) was between $0.2 < \mu < 0.8$ and $0.1 < \mu < 0.2$ during the night since the probability that the photon will be successfully detected also depend on the atmospheric transmission efficiency. The other important factor is the detection efficiency which is dependent on the physical apparatus on the receiver side and it's sensitivity towards noise and other interfering factors that makes the system deviate from an ideal setup. These factors can be however calculated to a degree by conducting an experiment with $\mu = 0$ transmission and comparing the results for day and night background generated noise. In this experiment the BB84 protocol was used. Some of the parameters such as the wavelength was unchanged from the previous experiment. The background radiance was mitigated by using spectral, spatial and temporal filtering. In this experiment however no polarization switching techniques were used. Here cryptographic monolithic randomizer generates two random bits to determine which of the four temperature−controlled diode laser should fire. Each laser corresponds to a state either in the rectilinear or diagonal basis. The lasers will emit a $\sim$ 1-ns, 772-nm optical pulse. On each cycle a 1-ns 1550-nm timing pulse was sent. The authors claim that this setup serves both simplicity and security. The malfunction of the random number generator could however sabotage the security of the system. Moreover, if the adversary manages to determine through some means, which laser fired she will not only know the base but the exact key bit that is transmitted. The photon detectors are silicon avalanche photodiodes cooled to $-20\ C^\circ$ operation temperature, with a single-photon detection efficiency of η_{det} $\sim$ 0.61 and a dark count rate of $\sim$ 1.6 kHz. After a 1 s transmission of 10^6 bits for each transmission a 6 second post processing step is necessary only to produce the shifted key resulting in 100 − 2,000 sifted key bits per 1-s quantum transmission. During this experiment 207 1-s transmission was performed during daylight with $\mu = 0.49$ and a $\sim 5.0\%$ BER. This resulted in 394,004 shifted key−bits from which 50,783 secret key−bits was constructed. In comparison 236 1-s transmission was done during nighttime conditions but with $\mu = 0,14$. This decreased the number of shifted key−bits to 192,925. However, since the BER was only $\sim 2,1\%$ the error correction steps discarded much less from the shifted key producing 118,064 bits of secret key. In a practical crypto−system this is sufficient to feed an AES encryption or for very short messages in OTP communication.

A huge stepping stone in free space QKD came in 2006 [10]. This experiment exceeded all previous ones in terms of transmission distance by more than one order of magnitude. The polarization entangled photons were generated at Alices's side 2400 m above sea level. A picosecond−pulsed laser used with a special crystal created energy degenerate entangled photon pairs of 710 nm wavelength. Using the Ekert protocol, −(an entanglements based key sharing protocol)− the photons are detected in the rectilinear or diagonal bases. The singlet state as previously is written as

$$|\Psi\rangle = \frac{1}{\sqrt{2}}\,|H\rangle_A\,|V\rangle_B - \frac{1}{\sqrt{2}}\,|H\rangle_A\,|V\rangle_B\,,$$

where $|H\rangle$ and $|V\rangle$ are the horizontal and vertical polarizations states while the lower indices indicate the spatial modes. One pair is detected by Alice right after generation since the source was located at her side. The other half is transmitted to Bob via a telescope over a 144 km long free space path. The alignment is automatically adjusted with a beacon laser based tracking system to mitigate beam drifting. The optical link efficiency is further attenuated by diffraction, absorption and imperfections in the physical components. The atmospheric losses are around 0.07 dB/Km at these altitudes. Adding up the various factors the attenuation of the whole channel was −25 dB in the best case with 25% single photon detector efficiency that is equivalent to a −6 dB attenuation. On both sides each detection event is labelled with a time tag that Bob sends to Alice who can see which subset of the transmissions arrived to Bob. To check for the presence of entanglement the evaluation of the CHSH inequality is necessary and it's violation was confirmed with a value of 2,5. The experiment however violates the locality loophole to some degree since the detection of the first photon took place with the other photon being still a few meters away, nevertheless the measurement of the other photon was space−like separated. As a result over a 75-s time period key−generation procedure 178 bits of distilled secure key was established which is $\sim$2.3 bit/s. Such key−rates are not sufficient to feed a modern crypto−system however it shows the feasibility of key exchange over large distances. Considering that over a satellite to earth communication the minimal distance is 400 km, the overall atmospheric thickness is about one order of magnitude less than in this experiment.

Satellite to ground station communication is now becoming a closer reality with these experimental results, however there are still challenges that needs to be solved such as how to aim a narrow beam from a moving object to a fixed station. The feasibility of the latter was demonstrated in 2012 [11], with an aircraft−to−ground downlink experiment from a fast moving airborne platform using the BB84 protocol with $\mu = 0.5$ sending 10 Mpps. This is a significant improvement over the previous experiment where the pulse rate was 1 million per second. The plane was between 1100 and 1300 m and the distance between the transmitter and receiver apparatus was 20 km. Scintillation, beam wandering and broadening was negligible for this experiment.

The total attenuation introduced during the process is constituted by several components. This comprises a free−space loss of 15 dB, atmospheric attenuation between 6−39 dB (9 in average), tracking loss on both the transmitter and receiver side 3 and 1 dB respectively. Furthermore, the attenuation caused by imperfections in optical parts adds another 3 dB and the coupling loss of the diode in the focal plane of Bob adds 2 dB. With this the total attenuation of the 20 km channel length is 33 dB in average but can vary between 30 and 63 dB depending on the weather conditions.

Tracking is accomplished by a GPS based closed−loop tracking system with active course tracking. The downlink transmissions wavelength is 850 nm with 10 MHz pulses where $\mu = 0.5$. The polarizations is done by a four−path laser diode. Bob is using a four−diode single photon detector for the incoming photons. The experiment was conducted after sunset and under new moon conditions. The achieved shifted key rate was 145 bit/s with the actual secure key−rate being 4.8 bit/s with a 4.5% QBER which is sufficient to encrypt transmissions over 1 Git/s.

V. RECENT ACHIEVEMENTS

Experiments to test the feasibility of satellite communication are important to overcome challenges introduced by the channel attenuation, tracking inaccuracy or imperfections in the physical parts. In satellite communication there are different "schools" , shown in Figure 3, that use different methods, each has certain advantages over the others. The first approach is the downlink communication introduced in the previous experiment. In this scenario the transmitter (Alice) is located on the satellite and the receiver (Bob) is on the ground station. The other is the uplink method where Alice is in the

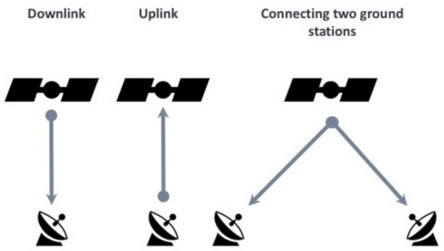


Fig. 3: Schematic overview of satellite based QKD transmission directions. In the downlink scenario the sender is located on the satellite and the ground station is the receiver. This is reversed at the uplink connection. The third scenario is different, here the satellite sends an entangled photon pair at two distant ground stations, where these stations can perform a Bell−type measurement on their half.

ground. As the atmospheric density is lower as we ascend, the downlink communication suffers less from diffraction and beam wandering as these effects are only introduced in the lower atmospheric layers. However, the receiver telescope is facing up and it is more exposed to noise that can come from, for example from a full moon. Both the downlink and uplink connection can serve as trusted node to establish keys between two ground stations or it can simply do a QKD between a ground station and the satellite. First the satellite establishes a secret key with an earth station and later can perform QKD with another earth station where this time it sends the established key as the secret message. The third approach is to directly connect two ground stations with the help of entanglement where the satellite generates an entangled pair and sends it over to the ground stations. In this scenario theoretically the satellite can act as an untrusted node since the parties will be able to detect any malicious activity by performing a Bell test.

A successful demonstration of a ground station based transmitter where the satellite was imitated by a moving truck was done in 2015 [12]. The distance between Alice and Bob was 650 m with the truck moving at 33 km/h to match the angular speed of a satellite traveling at a 600 km altitude with up to $0.75°/s$. On both sides a beacon and a tracking algorithm is applied to synchronize the alignment of the apparatus. The used protocol is a decoy state [13] BB84 where intensity and polarization of the photons are modulated. Weak coherent pulses at 532 nm are sent through a sum−frequency generation method combining a pulsed 810nm and a CW 1550 nm laser with the advantage that the 1550 nm laser removes the phase correlation between the pulses. The 532 nm pulses are sent to the transmitter via an optical fiber. The fiber transmission can introduce rotations in the polarization due to temperature fluctuations and the motion of the fiber. To mitigate this effect, the authors used a polarization characterization and compensation system based on a modified optical chopper which as it rotates lets through 50% of pulses unchanged, 30% is blocked and 20% is polarized in the horizontal, vertical, diagonal, antidiagonal, left circular or right circular polarization. 10% of the passed signals are sent to a single photon detector through a beam splitter which allow the tomograpical characterization of the polarization state to implement a real time polarization drift correction of the states. This drift can be caused by the fiber or other birefringent elements. Before the signal leaves the transmitter any phase or rotation is compensated using a set of motorized wave plates. With a total 30,6 dB attenuation during a 4 second transmission and a signal average photon number of 0.495, a decoy average photon number of 0.120, with a signal QBER of 6.55% and a decoy QBER of 5.49%. From a 5844 bit long shifted key a 160 bit long secure key was obtained through LDPC error correction and privacy amplification.

The biggest milestone in free space quantum key distribution comes from an actual satellite QKD demonstration made in 2016 with satellite Micius. [14] Fiber and terrestrial free space links exponentially reduce the transmission efficiency as they introduce a lot of attenuation. However, in empty space the photon loss is negligible. Therefore, successful demonstrations of QKD were conducted with satellite altitudes between 500 and 1200 km. The protocol as in the previous experiment was a decoy state BB84 at an 850 nm transmitter wavelength. A method for increasing the transmission efficiency would be to increase the signal power, however this is not possible for security reasons described before. The other approach is to mitigate the channel attenuation at all stages possible. The advantage of the downlink method that was used is that beam wandering becomes significant only at the end of the transmission channel due to atmospheric turbulence, but at that point the beam size is larger due to diffraction than the wandering and it won't significantly effect the transmission rate. For a 1200 km transmission the attenuation introduced by diffraction is around 22 dB. To establish the link between the parties a high−precision acquiring, pointing and tracking system is used with beacon leasers. The cascaded multi−stage APT system that was designed by the authors, reduces the pointing error below 3 dB. To reduce background noise temporal and spectral filtering is added to the system, plus the beacon leaser is used for ATP synchronization and to get timing information that is used to tag the received signal photons within a 2-ns time window. During the transmission motorized dynamical polarization compensation must be used to compensate the rotation angle offset induced by the relative motion of the satellite and the ground station. The BB84 decoy state protocol uses 3 intensity levels with 50% of the time signal, 25−25% of the time the two decoy states are transmitted with $\mu_s = 0.8,\ \mu_1 = 0.1,\ \mu_2 = 0$. The single photon detection efficiency of the ground station detectors is 50%. The overall optical efficiency considering the receiving telescope and the fibre coupling on the ground station is ˜16%. The classical communication is conducted through radio transmission between the satellite and the ground station. Every night the satellite passes the ground station an approximately 5−minute time period is open for the key exchange. In a short 273 s QKD transmission the ground station can detect 3,551,136 detection events from which 1,671,072 bits of sifted key is acquired. With the satellite at higher altitudes the shifted key rate decreases from 12 kbit/s at 645 km, to around 1 kbit/s at 1200 km and has a peak rate of 40.2 kbits/s at 530 km. The QBER varies between 1.1−3% depending on the altitude. For comparison if we would like to cover the same distance with a single fibre and conduct the same experiment even with perfect single photon sources and 100% efficient single photon detectors to obtaining 1 bit of shifted key over 1200 km would take 6 million years.

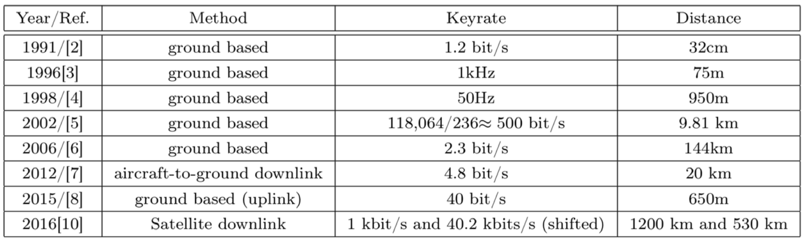
When the information is encoded in photon polarization the alignment of the transmitter and receiver apparatus becomes necessary. For this reason, they must have a shared reference frame (SRF). In polarization encoding this would ideally mean that if the transmitter sends a horizontally polarized photon the receiving apparatus should be able to detect it in case of a rectilinear detection, with probability 1, since the probability of correct detection is $cos^2\phi$ where ϕ is the angle difference between the devices reference frame, which should be zero. Constantly monitoring the rotation introduced by the apparatus is important, however during the transmission further rotation can be added by the channel and decoherence can worsen the rate of correct detection even if the parties used the same bases. A solution for this problem was proposed and demonstrated in reference [15]. The idea behind this experiment is to send rotational invariant photonic states by combining photon polarization and orbital angular momentum. OAM is a different degree of freedom in angular momentum compared to spin angular momentum which is associated with polarization. In this case the electromagnetic field is described by a twisted wavefront which has a helical shape and it's composed of ℓ intertwined helices with an optical vortex in the center, where photons carry $\ell\hbar$ OAM. Using this mode increases the channel cross−section size and it is only suitable for free space experiments, as single mode fibers are not compatible with this mode over long distances. OAM is conserved in vacuum, however it is affected by atmospheric turbulence. Rotational invariance is achieved with compensating misalignments in polarization with misalignments in spatial modes. A 210m QKD transmission was performed using BB84 decoy state protocol sending the qubits in the decoherence−free subspace of the four−dimensional OAM−polarization product Hilbert space. At the transmitter side there are 4 polarized attenuated lasers that can send photons in horizontal, vertical, right circular or left circular polarization states. Next the photons are transformed into rotational invariant states by a q−plate (space−variant birefringent plate with topological charge q). At the receiver side a second q−plate transforms back the incoming photon into the original polarized state which is then observed by polarizers and single photon detectors. This setup is insensitive to the relative reference frames of the users. The experiment was performed in different setups by rotating the transmitter telescope with $0°$, $15°$, $45°$, $60°$. Only compensation of polarization alterations coming from fiber distortions at Alice's side was performed, without making changes at the receiver side. Without using the rotational invariant states the QBER would depend on the relative angle Θ between Alice's and Bob's reference frames and would make QKD infeasible above a rotation angle of $15°$. However with this approach the key exchange was successfully performed even in the other scenarios.

Satellite based quantum communication would require a cheap and easy to deploy satellite network that could connect terrestrial locations that are not otherwise connectable using fiber based links. Using nano−satellites that could perform orbit−to−ground transmission of QKD both with single and entangled photons, is the proposal of the CubeSat Quantum Communications Mission (CQuCoM) [16]. The approach is to lunch miniaturized satellites massing only a few kilograms called nanosats, that are much easier to lunch, cheaper to develop and operate, and therefore highly reduces the cost of such missions. The CubeSat platform is highly accessible as all parts can be ordered online that are necessary to build a fully functional satellite. CubeSats are rapidly lunched for scientific, commercial and governmental purposes with 120 launched in 2015 and 118 in 2014. Ideally the CubeSats are capable of WCP BB84 QKD (decoy state) and entanglement based QKD where one−half of each pair is used in the downlink transmission and the other half is retained. The major challenge is the high pointing accuracy required to minimize data loss. The CQuCoM CubeSat platform is based on PICosatellite for Atmospheric and Space Science Observations (PICASSO) system which is a 3U (three unit) system but can be modified into 6U.

Using traditional radio frequencies or laser communication the mass of the spacecraft scales with the maximum data rate achievable from space. Small satellites are very limited in bandwidth using radio frequencies. Low orbit cubesats can reach only tens of Mbps. The National Institute of Information and Communications Technology (NICT) in Japan has launched the SOTA (Small Optical Transponder) mission [17], to experimentally prove the feasibility of high−bitrate laser-communication from a micro−satellite platform. SOCRATES (Space Optical Communications Research Advanced Technology Satellite) is a 48 kg microsatellite with Sun−synchronous near−circular orbit at an altitude of $\sim$600 km. Although the primary purpose was to perform high−speed transmission of data using lasercom a QKD experiment was conducted as well using non−orthogonal, linearly polarized laser sources at $\lambda = 1549$ nm wavelength.

One big challenge in free space QKD is that during daytime transmission the background noise is significantly higher and it can increase the dark count rate which results in a higher QBER and that lowers the key−rate. In an experiment at the Spanish National Research Council [18], researchers performed a 24 hour experiment and monitored how the background noise effects the QBER and thus the secret key−rate. The results show that there is a significant difference between the daylight and nighttime transmission. Between 21:30 and 6:30 there is almost zero background noise and the QBER is only around 2% with a high secret key rate. As soon as the sun rises, the background noise rises, and as a consequence the QBER, making the secret key rate much lower.

From these experiments it is clear that the free space link attenuation may vary between day and night transmission, but the channel itself cannot be improved. The photon detection efficiency due to the imperfections in single photon detectors, like dead time or dark counts, also puts a limit on the number of photons that can be successfully transmitted in one second. Encoding more than one bit of information in each qubit can be a promising method with the aim of increasing the key−rate of the QKD transmission. High dimensional QKD



Year/Ref.	Method	Keyrate	Distance
1991/[2]	ground based	1.2 bit/s	32cm
1996[3]	ground based	1kHz	75m
1998/[4]	ground based	50Hz	950m
2002/[5]	ground based	118,064/236≈ 500 bit/s	9.81 km
2006/[6]	ground based	2.3 bit/s	144km
2012/[7]	aircraft-to-ground downlink	4.8 bit/s	20 km
2015/[8]	ground based (uplink)	40 bit/s	650m
2016[10]	Satellite downlink	1 kbit/s and 40.2 kbits/s (shifted)	1200 km and 530 km

Fig. 4: QKD transmission method, performance and distance overview

using hyperentangled photons that use polarization and at the same time energy−time entanglement −(here the photon arrival times contain the extra information)− as an extra degree of freedom, were successfully performed with a 1.2−km long free−space link across Vienna [19]. This approach not only increases the channel captivity, but it is further beneficial for improving noise and eavesdropping resistance. Information can be encoded in different photonic degrees of freedom, such as transverse orbital angular momentum, discrete photon arrival time bins or continuous variable energy–time mode. Hyperentanglement can be achieved by spontaneous parametric down−conversion in nonlinear crystals and can be described as the tensor product of the two lower dimensional Hilbert spaces, thus acquiring a four−dimensional hyperentangled state. Photon A was measured locally while B was transmitted to the receiver over a 1.2 km free space link, overlapped with a 532-nm beacon laser for pointing, acquisition and tracking. On both sides the parties used a polarization analyzer and an optional transfer setup, that coupled the energy−time degree of freedom to the polarization degree of freedom. The coincidence rate at Alice's side was ~ 400 kcps while Bob detected ~ 350 kcps and an average ~ 20 kcps two−photon detections per second. The average link transmission efficiency was around 18% calculating all losses from source to receiver. To verify the presence of entanglement and evince the presence of an eavesdropper hyperentanglement−assisted Bell−state measurements can be used.

VI. CONCLUSION

Quantum communication and free space QKD is still in an early R&D stage and it will take a lot of time and investment as well as miniaturization and cheaper components, until it can become a true alternative to classical public key algorithms on a large scale. As it is the only method to establish keys with theoretical unconditional security it will be the go to option for a lot of sectors where security is at−most important, such as governmental institutions, banking, military or for strictly confidential company secrets. There are a lot of different approaches and protocols within free space quantum key distribution and at this stage it would be hard to predict which will be the go to direction. As free space quantum key distribution is making it's baby steps there were significant improvements over the last 20 years both in terms of performance and achieved distance. Figure 4. summarizes some of the results that were achieved in this time period.

ACKNOWLEDGMENT

The research was supported by the Hungarian Scientific Research Fund – OTKA PD−112529. The research is connected to COST Action CA15220 Quantum Technologies in Space.

REFERENCES

[1] S. Imre and F. Balázs. Quantum Computing and Communications – An Engineering Approach. Wiley, 2004

[2] L. Hanzo; H. Haas, S. Imre, D. O'Brien, M. Rupp and L. Gyongyosi, "Wireless Myths, Realities, and Futures: From 3G/4G to Optical and Quantum Wireless", Proceedings of the IEEE, Volume: 100 , Issue: Special Centennial Issue, pp. 1853-1888.

[3] C. H. Bennett and G. Brassard, "Quantum cryptography: Public key distribution and coin tossing," in Proceedings of the IEEE International Conference on Computers, Systems and Signal Processing. Bangalore, India: IEEE, 1984

[4] C. H. Bennett, G. Brassard and J-M Robert, "Privacy amplification by public discussion", SIAM Journal on Computing Vol. 17, no. 2 April 1988, pp. 210-229

[5] Michael A. Nielsen and Isaac L. Chuang. Quantum Computation and Quantum Information: 10th Anniversary Edition. Cambridge University Press, New York, NY, USA, 10th edition, 2011.

[6] C. H. Bennett, F. Bessette, G. Brassard, L. Salvail and John Smolin, Experimental Quantum Cryptography, Journal of Cryptology, 1992, volume 5, pages 3-28.

[7] B. C. Jacobs and J. D. Franson, Quantum cryptography in free space, Opt. Lett. 21, 1854-1856 (1996), https://doi.org/10.1364/OL.21.001854.

[8] W. T. Buttler, et al. Practical Free-Space Quantum Key Distribution over 1 km, Phys. Rev. Lett. Vol. 81, issue 15, pages = 3283--3286, 1988

[9] J. Richard at al. Practical free-space quantum key distribution over 10 km in daylight and at night, Physics Division, Los Alamos National Laboratory, Los Alamos, NM 87545, USA

[10] R. Ursin at al. Free-Space distribution of entanglement and single photons over 144 km, 2006

[11] Florian Moll, Sebastian Nauerth at al. Communication system technology for demonstration of BB84 quantum key distribution in optical aircraft downlinks. DOI: 10.1117/12.929739, 2012.

[12] J-P. Bourgoin, at al. Free-space quantum key distribution to a moving receiver, Opt. Express 23, 33437-33447 2015

[13] Hoi-Kwong Lo, Xiongfeng Ma, Kai Chen, Decoy State Quantum Key Distribution. Phys. Rev. Lett. 94, 230504 2005

[14] Sheng-Kai Liao at al. Satellite-to-ground quantum key distribution, Nature 10.1038/nature23655, 2017

[15] V. Giuseppe, at al. Free-Space Quantum Key Distribution by Rotation-Invariant Twisted Photons, Phys. Rev. Lett. vol. 113, issue: 6, doi = 10.1103/PhysRevLett.113.060503

[16] Daniel KL Oi at al. CubeSat quantum communications mission. EPJ Quantum Technology 2017, volume 1.

[17] A. Carrasco-Casado at al. LEO-to-ground optical communications using SOTA (Small Optical TrAnsponder) – Payload verification results and experiments on space quantum communications http://dx.doi.org/10.1016/j.actaastro.2017.07.030

[18] A. Carrasco-Casado; V. Fernández; N. Denisenko, Chapter from the book "Optical Wireless Communications", pp. 589-607 doi: 10.1007/978-3-319-30201-0_27 Springer, 2016

[19] F. Steinlechner, at al. Distribution of high-dimensional entanglement via an intra-city free-space link, DOI: 10.1038/ncomms15971

László Bacsárdi obtained M.Sc. degree in computer engineering at Budapest University of Technology and Economics (BME) in 2006. He holds an associate professor position at the University of Sopron, where he is the Head of the Institute of Informatics and Economics. He wrote his PhD thesis on the possible connection between space communications and quantum communications at the BME Department of Telecommunications in 2012. His current research interests are in mobile ad hoc communication, quantum computing and quantum communications. He is the Secretary General of the Hungarian Astronautical Society (MANT), which is the oldest Hungarian non-profit space association founded in 1956. He is member of the board of a Hungarian scientific newspaper ('World of Nature') and he is the publisher of a non-profit Hungarian space news portal ('Space World'). Furthermore he is member of IEEE, AIAA and the HTE as well as alumni member of the UN established Space Generation Advisory Council (SGAC).

Tamás Bisztray obtained B.Sc. degree in mathematics and an M.Sc. degree in computer science at Eötvös Loránd Univesity (ELTE). He is a PhD student at Budapest University of Technology and Economics in the topic of Quantum communication algorithms. He is working at Ericsson as an intern on Quantum Communication methods. His current research interests are in quantum communication algorithms and how these can be improved to better serve and enable a quantum encryption network.

Guidelines for our Authors

Format of the manuscripts

Original manuscripts and final versions of papers should be submitted in IEEE format according to the formatting instructions available on

http://www.ieee.org/publications_standards/publications/authors/authors_journals.html#sect2,
"Template and Instructions on How to Create Your Paper".

Length of the manuscripts

The length of papers in the aforementioned format should be 6-8 journal pages.
Wherever appropriate, include 1-2 figures or tables per journal page.

Paper structure

Papers should follow the standard structure, consisting of *Introduction* (the part of paper numbered by "1"), and *Conclusion* (the last numbered part) and several *Sections* in between.
The Introduction should introduce the topic, tell why the subject of the paper is important, summarize the state of the art with references to existing works and underline the main innovative results of the paper. The Introduction should conclude with outlining the structure of the paper.

Accompanying parts

Papers should be accompanied by an *Abstract* and a few *index terms (Keywords)*. For the final version of accepted papers, please send the *short cvs* and *photos* of the authors as well.

Authors

In the title of the paper, authors are listed in the order given in the submitted manuscript. Their full affiliations and e-mail addresses will be given in a footnote on the first page as shown in the template. No degrees or other titles of the authors are given. Memberships of IEEE, HTE and other professional societies will be indicated so please supply this information. When submitting the manuscript, one of the authors should be indicated as corresponding author providing his/her postal address, fax number and telephone number for eventual correspondence and communication with the Editorial Board.

References

References should be listed at the end of the paper in the IEEE format, see below:

a) Last name of author or authors and first name or initials, or name of organization
b) Title of article in quotation marks
c) Title of periodical in full and set in italics
d) Volume, number, and, if available, part
e) First and last pages of article
f) Date of issue

[11] Boggs, S.A. and Fujimoto, N., "Techniques and instrumentation for measurement of transients in gas-insulated switchgear," IEEE Transactions on Electrical Installation, vol. ET-19, no. 2, pp.87–92, April 1984.

Format of a book reference:

[26] Peck, R.B., Hanson, W.E., and Thornburn, T.H., Foundation Engineering, 2nd ed. New York: McGraw-Hill, 1972, pp.230–292.

All references should be referred by the corresponding numbers in the text.

Figures

Figures should be black-and-white, clear, and drawn by the authors. Do not use figures or pictures downloaded from the Internet. Figures and pictures should be submitted also as separate files. Captions are obligatory. Within the text, references should be made by figure numbers, e.g. "see Fig. 2."
When using figures from other printed materials, exact references and note on copyright should be included. Obtaining the copyright is the responsibility of authors.

Contact address

Authors are requested to submit their papers electronically via the EasyChair system. The link for submission can be found on the journal's website:
www.infocommunications.hu/for-our-authors

If you have any question about the journal or the submission process, please do not hesitate to contact us via e-mail:

Rolland Vida – Editor-in-Chief:
vida@tmit.bme.hu

Árpád Huszák – Associate Editor-in-Chief:
huszak@hit.bme.hu

LEGRAND WIRELESS CHARGERS

legrand®

100%

EVERYWHERE, WITH TOTAL FREEDOM

PRACTICAL

DISCREET

ROBUST

VERSATILE

WIRELESS CHARGER FLUSH WALL-MOUNTED
IP66 - IK8

WIRELESS
CHARGER
legrand

WIRELESS CHARGER IN FLUSH
WALL-MOUNTED
POWER STRIP

WIRELESS CHARGER, INSTALLABLE
IN FURNITURE

CATALOGUE
legrand

WIRELESS CHARGER BUILT INTO
A MULTI-SOCKET TOWER

www.legrand.hu

SCIENTIFIC ASSOCIATION FOR INFOCOMMUNICATIONS

hte

FOUNDED IN 1949

SCIENTIFIC
ASSOCIATION FOR
INFOCOMMUNICATIONS
HUNGARY

Who we are

Founded in 1949, the Scientific Association for Infocommunications (formerly known as Scientific Society for Telecommunications) is a voluntary and autonomous professional society of engineers and economists, researchers and businessmen, managers and educational, regulatory and other professionals working in the fields of telecommunications, broadcasting, electronics, information and media technologies in Hungary.

Besides its 1000 individual members, the Scientific Association for Infocommunications (in Hungarian: Hírközlési és Informatikai Tudományos Egyesület, HTE) has more than 60 corporate members as well. Among them there are large companies and small-and-medium enterprises with industrial, trade, service-providing, research and development activities, as well as educational institutions and research centers.

HTE is a Sister Society of the Institute of Electrical and Electronics Engineers, Inc. (IEEE) and the IEEE Communications Society.

What we do

HTE has a broad range of activities that aim to promote the convergence of information and communication technologies and the deployment of synergic applications and services, to broaden the knowledge and skills of our members, to facilitate the exchange of ideas and experiences, as well as to integrate and harmonize the professional opinions and standpoints derived from various group interests and market dynamics.

To achieve these goals, we…

- contribute to the analysis of technical, economic, and social questions related to our field of competence, and forward the synthesized opinion of our experts to scientific, legislative, industrial and educational organizations and institutions;
- follow the national and international trends and results related to our field of competence, foster the professional and business relations between foreign and Hungarian companies and institutes;
- organize an extensive range of lectures, seminars, debates, conferences, exhibitions, company presentations, and club events in order to transfer and deploy scientific, technical and economic knowledge and skills;
- promote professional secondary and higher education and take active part in the development of professional education, teaching and training;
- establish and maintain relations with other domestic and foreign fellow associations, IEEE sister societies;
- award prizes for outstanding scientific, educational, managerial, commercial and/or societal activities and achievements in the fields of infocommunication.

Contact information

President: **GÁBOR MAGYAR, PhD** • *elnok@hte.hu*
Secretary-General: **ERZSÉBET BÁNKUTI** • *bankutie@ahrt.hu*
Operations Director: **PÉTER NAGY** • *nagy.peter@hte.hu*
International Affairs: **ROLLAND VIDA, PhD** • *vida@tmit.bme.hu*

Address: H-1051 Budapest, Bajcsy-Zsilinszky str. 12, HUNGARY, Room: 502
Phone: +36 1 353 1027
E-mail: *info@hte.hu*, Web: *www.hte.hu*