
TECHNICAL REPORT

Editors
Sitaram Chamarty, Ben Gilad and Oded Margalit

Technical Report #19-01

July 2, 2019

The BGU-Negev Hi-Tech Faculty Startup Accelerator,
Department of Computer Science,
Ben-Gurion University, Beer Sheva, Israel
2019 INTERNATIONAL SYMPOSIUM ON CYBER SECURITY CRYPTOGRAPHY AND MACHINE LEARNING

JUNE 2019 • 27-28 • BEN-GURION UNIVERSITY ALON BUILDING FOR HI-TECH

PROF. SHLOMI DOLEV
DR. SACHIN LODHA
GENERAL CHAIRS
## Table of Contents

PhD Student Research Track ........................................................................................................ 1

Introduction .................................................................................................................................. 2

A New Family of Trapdoor Functions for Single Database Information-theoretic Private Block
Retrieval ....................................................................................................................................... 3
Radhakrishna Bhat and N R Sunitha

Towards in-vivo Energy Harvesting Computational Nanorobots ................................................. 8
Shlomi Dolev and Ram Prasadh Narayanan

Sampling for effective logging policies for database activity monitoring .................................... 12
Hagit Grushka, Ofer Biller, Bracha Shapira, Lior Rokach and Oded Sofer

Assume, Guarantee or Repair ....................................................................................................... 15
Hadar Frenkel, Orna Grumberg, Corina Pasareanu, and Sarai Sheinvald.

Self-Stabilizing Local Load Balancing ......................................................................................... 19
Shlomi Dolev and Manish Kumar

Reinforcement Learning Method for Computing the Capacity of Communication Channels with
Feedback ........................................................................................................................................ 23
Ziv Aharoni, Oron Sabag and Haim Permuter

Visual Analytics for Unsupervised Anomaly Detection of Multivariate Sensor Streams .................. 27
Efrat Vilenski and Jonathan Rosenblatt

LOCALIZATION OF DATA INJECTION ATTACKS ON DISTRIBUTED M-
estimation ................................................................................................................................. 31
Or Shalom, Amir Leshem and Anna Scaglione

Information-Theoretically Secure Quantum Gate Computation and Applications ......................... 36
Dor Bitan and Shlomi Dolev
Entrepreneurship Pitch Track

Introduction

Algorithm and IoT nitrate sensor to optimize agriculture yields and protect the environment

Shlomi Arnon

SocPro: Social Protector for Online Social Networks

Nadav Voloch

Cryptopus: Crypto currency monitoring application

Nadav Voloch

Computational Nano-robots as In-Vivo Medicine.

Ram Prasadh Narayanan

AdaptiveClimb – Adaptive Policy for Cache Replacement

Marina Sadesky

Exploiting compressed video domain redundancy for cyber protection and compression efficiency (casting covert channel in compressed video)

Yoram Segal/Ofer Hadar

Secret Computing: Privacy Preserving Machine Learning and Analytics

Nicolas Gama, Inpher and Mariya Georgieva

Stick ‘n Grip: A revolutionary robot gripper for the e-commerce industry

Amir Shapiro
PhD Student Research Track

Chair: Oded Margalit
Introduction

PhD Student Research Track chaired by Prof. Oded Margalit

CSCML has four diverse tracks: Academic, Entrepreneurship, CTF (Capture The Flag) and the PhD track. The PhD session is a great way to get a taste of the great research in the areas of Cyber Security; Cryptology; and Machine Learning.

This year we had nine short presentations, where researchers presented their work for peer-review. The topics were diverse: from the physical nano-particle world of robots that can fit into the human bloodstream; through the recent hot topic of Quantum computing; to the classic load balancing problems.

We got excellent speakers from the Technion (Israel); Bar Ilan University (Israel); Siddaganga Institute of Technology Tumakuru (India); and, of course, Ben Gurion University – the hosting institute. The first talk was given remotely, by a video conference.

I’m happy to note that one of the talks was done in collaboration with IBM, an industry sponsor of CSCML.

Looking forward to CSCML 2020.

Regards,

Prof. Oded Margalit, IBM Cyber Security Center of Excellence (CCoE)

PhD Student Research Track Chair
A New Family of Trapdoor Functions for Single Database Information-theoretic Private Block Retrieval
(An Efficient Transformation Capabilities of Single Database Private Block Retrieval)

Radhakrishna Bhat\textsuperscript{1,2} and N R Sunitha\textsuperscript{1,3}

\textsuperscript{1} Department of Computer Science and Engineering, Siddaganga Institute of Technology, Affiliated to Visveswaraya Technological University Belagavi, B H Road, Tumakuru, Karnataka, India 572103
\textsuperscript{2} rsb567@gmail.com
\textsuperscript{3} nrsunithasit@gmail.com

Abstract. The Private Information Retrieval (PIR) is one of the promising techniques to preserve user privacy in the presence of trusted-but-curious servers. The information-theoretically private query construction assures the highest user privacy over curious and unbounded computation servers. Therefore, the need of information-theoretic private retrieval was fulfilled by various schemes in variety of PIR settings. To augment previous work, we propose a combination of new bit connection methods called rail-shape and signal-shape and new quadratic residuosity assumption based family of trapdoor functions for generic single database Private Block Retrieval (PBR). The main goal of this work is to show that the possibility of mapping from computationally bounded privacy to information-theoretic privacy or vice-versa in a single database setting using newly constructed bit connection and trapdoor function combinations. The proposed bit connection and trapdoor function combinations have achieved the following results.

- **Single Database information-theoretic PBR (SitPBR):** The proposed combinations are used to construct SitPBR in which the user privacy is preserved through the generation of information-theoretic queries and data privacy is preserved using quadratic residuosity assumption.
- **Single Database computationally bounded PBR (ScPBR):** The proposed combinations are used to construct ScPBR in which both user privacy and data privacy are preserved using well-known intractability assumption called quadratic residuosity assumption.
- **Map(SitPBR)→ScPBR:** The proposed combinations can be used to transform (or map) SitPBR into ScPBR scheme by choosing appropriate function parameters.
- **Map(ScPBR)→SitPBR:** The proposed combinations can be used to transform (or map) ScPBR into SitPBR scheme by choosing appropriate function parameters.
Keywords: Private information retrieval · Information-theoretic privacy · User privacy · Private Block Retrieval · Oblivious transfer · Probabilistic encryption

1 Introduction

The goal of any privacy critical applications is to preserve the underlying privacy (like user privacy or server privacy or data privacy) with guaranteed confidentiality primitive (i.e., information-theoretic).

Among all other user privacy preserving techniques, Private Information Retrieval (PIR) is one of the prominent privacy preserving techniques to preserve both user privacy and data privacy introduced by Chor et.al [8,10]. The private information retrieval also called as special case of 1-out-of-n oblivious transfer involves two communicating parties: user and server in which user privately reads a single bit from server’s n bit database. The basic goal of Chor et.al [8,10] was to provide the highest confidentiality to the user’s interest (may be index, pattern, graph moves etc.) for real-time privacy applications. Since then, comprehensive research has been carried out in several dimensions of PIR including relaxing the privacy level from information-theoretic to a computationally bounded setting, reducing communication and computation overhead, reducing number of rounds and number of servers involved, extending to private write etc.

One of the natural extensions to PIR protocol is Private Block Retrieval (PBR) in which user privately reads a block (instead of a bit) from server’s database. Based on the level of privacy, the PIR protocol is broadly divided into two groups: information-theoretic PIR and computationally bounded PIR as described below.

- **Information-theoretic PIR** (itPIR): If the PIR protocol involves information-theoretically private queries with non-colluding replicated database server entities then such scheme is considered as information-theoretic PIR (itPIR) in which the user privacy is preserved through the information-theoretically private queries. Several information-theoretic schemes [21,6,13,14,2,1] and some PBR extensions [10,3,19,12,20] have concentrated on providing information-theoretic privacy using database replications.

- **Computationally bounded PIR** (cPIR): If the PIR protocol involves a computationally bounded (or computationally intractable) database server entities then such scheme is considered as computationally bounded PIR (cPIR) in which the privacy is preserved based on the well-defined cryptographic intractability assumption(s). Most of the research work [17,9,4,16,22,15] and [23,18,7,12,20,5] on cPIR concentrated on using a single intractability assumption to preserve both user privacy and data privacy.

There are following major problems in the existing single database PBR schemes (including both itPBR and cPBR).
- Lack of sufficient itPIR approaches: More research focus was on the construction of an efficient cPBR instead of itPBR in a single database setting. This leads to the lack of information-theoretic privacy guarantee to the user.

- Lack of independency between user and data privacy: Most of the existing cPBR schemes use a single intractability assumption (such as Quadratic residuosity, Phi-hiding, Lattices, Composite residuosity etc) to preserve both user privacy and data privacy. If the curious party breaks the underlying intractability assumption then both the privacy concerns are easily compromised without extra effort. For instance, the single database PIR protocol constructed by Kushilevitz and Ostrovsky [17] rely on the well-known intractability assumption called Quadratic Residuosity Assumption (QRA) to achieve both the user privacy (through the computationally intractable query inputs with quadratic residuosity properties) and the data privacy (through the quadratic residuosity ciphertexts). Note that compromising the QRA naturally reveals both privacy concerns (without extra effort). Therefore, there is a strong need of a generic scheme with efficient mapping from cPBR to itPBR in such a way that the underlying primitive of user privacy should also map from intractability assumption to information-theoretic privacy. Note that, Kushilevitz and Ostrovsky scheme does not support an efficient mapping cPBR to/from itPBR.

- Lack of generic framework that fulfil the above needs: Due to the lack of generic PBR framework (which can be used as a generic framework for several privacy critical applications such as PBR, oblivious transfer, asymmetric encryption etc), there is a strong need of a generic PBR scheme that can efficiently transform between several PBR extensions like information-theoretic PBR, computationally bounded PBR, oblivious transfer, asymmetric encryption etc.

With this thorough investigation, the natural question that arises is as follows.

*Is it possible to construct a generic single database Private Block Retrieval framework with reasonable performance that fulfils one or more privacy concerns (such as user privacy, data privacy, server privacy) of private block retrieval and oblivious transfer?*

**Our Single Database Private Block Retrieval Solution:** We introduce a new bit connection and QRA based trapdoor functions for a single database PBR with the following results.

- New quadratic residuosity based single bit *injective* and *lossy* trapdoor functions.

- New bit connection methods (BCMs) called *rail-shape* and *signal-shape* to interconnect the proposed trapdoor functions with the aid of quadratic residuosity based injective trapdoor functions introduced by Freeman et.al [11].

- The appropriate combination of the proposed bit connection methods and trapdoor functions serve as a generic framework to map between several PBR extensions such as information-theoretic PBR, computationally bounded PBR, oblivious transfer, asymmetric encryption etc.
− New single database information-theoretic PBR (SitPBR) schemes using the combination of proposed bit connection methods and trapdoor functions in which the communication cost of the first scheme is $O(u(v - 2) + 2u \log N)$ and it’s computation cost is $O(u(2v - 2))$ where $n = uv$ is the database size, $u =$ rows, $v =$ columns, and $N$ is the RSA composite. The communication cost of the second scheme is $O(u(v - 1) + u \log N)$ and it’s computation cost is $O(u(2v - 1))$.

− New single database computationally bounded PBR (ScPBR) schemes in which the communication cost of the first scheme is $O(u(v - 2) + 2u \log N)$ and it’s computation cost is $O(u(2v - 2))$. The communication cost of the second scheme is $O(u(v - 1) + u \log N)$ and it’s computation cost is $O(u(2v - 1))$.

− At their basic construction, all the proposed schemes are single round, memoryless, and plain database protocols.

References


Towards in-vivo Energy Harvesting Computational Nanorobots

Shlomi Dolev and Ram Prasad Narayanan
Department of Computer Science, Ben-Gurion University of the Negev, Israel
{dolev, narayan}@cs.bgu.ac.il

Abstract. We discuss the design of a nanorobot which can navigate, detect cancer cells in the blood and actuate the exposure of drugs. This concept is in line with the effort to design an autonomous computational nanorobot for in-vivo medical diagnosis and treatment. We present this facile approach to design a collective system to visualize the programability in nanorobots. Through this work, we present an overall picture towards an inorganic autonomous computational nanorobot for cancer diagnosis and treatment. An implementable model and simulation of an oscillating Carbon Nanotube (CNT) for radio communication in nanorobots is also presented. We develop a model to predict the oscillation frequency of the cantilever beam based on its properties, including the device geometry. By this, we provide an implementable design of in-vivo transceiver, suitable for future large-scale applications based on radio communicating nanorobots. A swarm of such nanorobots can be regarded as implementable programmable matter. Also, synchronization of nanorobots can be inspired from nature. We intend to discuss two propositions concerning the synchronization of nanorobots (also called particles) inspired by pulsating impulses exhibited by nature (fireflies and honeybee combs). Assuming arbitrary incoherency (initially) for all particles in the medium, it can be shown that coherency improves over time within each subset of particles due to stronger and mutual influence of one particle on the others.

Keywords: Nano-Electro-Mechanical System (NEMS) · Actuator · Cancer · Energy harvesting · Carbon-Nanotube (CNT) detector · Tumor detection

1 Introduction

Nanorobotics is emerging as a growing research field with applications in medicine, surgery, intelligent coating, smart materials, to name a few. The promising advances in electronic fabrication and research on nanoparticles have attributed

* We thank the Lynne and William Frankel Center for Computer Science, the Rita Altura Trust Chair in Computer Science, the Kreitman School of Advanced Graduate Studies, Ben-Gurion University of the Negev for their support. This research was also supported by a grant from the Ministry of Science & Technology, Israel and German Research Funding Organization (DFG, Grant#8767581199).
to vast improvements in this field. Until recently, the ability to realize theoretical concepts of nanorobotics into practice was limited to research stages. Recent research results in the field of DNA origami and computing [1–3], programmable matters [4, 5] and nanorobotic swarming systems [6] have provided evidences of implementable nanosystems. One prime example that can be defined as nanorobots would be the concept of programmable matters, defined as particles having certain computational capability that can either actively or passively involve in sensing, flocking and foraging. The ability of programmable matters to exhibit phenomenal changes to its properties upon action of chemical or bio-sensing [7, 8], or a combination of both, influences in realizing inorganic nanorobots for varied in-vivo sensing and actuation purposes.

2 Discussion: Design of Nanorobot Structure

In a previous work of ours [9, 10], we have discussed in detail, the design of a nanorobot and its possible use in cancer detection and treatment. We had provided results on numerical calculations and simulation for the same. We have also discussed an implementable design of in-vivo transceiver, suitable for future large-scale applications based on radio communicating nanorobots.

3 Radio Transmission and Self Synchronization

3.1 Proposition #1: Time Stamped Self Synchronization

Here, we assume that all particles are identical in geometry, physical parameters and computational capability. Each particle is equipped with a radio and an oscillator that allow it to send a beep at some fixed power every $T$ steps, for some fixed $T$. Initially, the particles may not be synchronized. So the goal will be to eventually arrive at a state where all particles beep at the same time. Each particle can only listen to the shared medium or transmit a beep, but it cannot do both. If a particle listens to the shared medium, it either notices an idle or a busy medium, depending on whether the cumulative signal strength of the beeps sent out at that time exceeds some fixed threshold or not. Depending on the outcome, it may then decide to reset its starting point for the beeps or not. We propose a simple greedy algorithm for the synchronization of the beeps. After a particle has sent out a beep, it will not listen to the medium for $T/3$ steps (so that, for example, it has some time to recharge). Afterwards, it keeps listening to the medium until it sends out another beep (because $T$ time steps have passed by since its last beep). If, during any of the time steps at which it is listening, the particle notices a busy medium, it resets the starting point of its beeping period to that step (in the hope that also $T$ steps later the medium will be busy again, so that its beep signal will contribute to strengthening that cumulative beep signal) and beeps in the next step (in order to cause also other particles to reset their beeping periods, even though this will not yet cause the particles to be perfectly synchronized). Note that when a particle beeps because it noticed a
busy channel, it just takes $T - 1$ steps before it beeps again because it wants to be synchronized with the received cumulative beep signal. Otherwise, the time between two beeps is $T$ steps. We noticed that in certain situations this algorithm will quickly cause the particles to transmit beeps in a synchronized fashion, and we are currently working on a formal analysis exploring these situations. For example, the greedy algorithm would work if all the nodes are sufficiently close to each other so that the graph $G$ in which two nodes $v$ and $w$ are connected if $v$ can notice the beep signal of $w$ forms a single connected component and $G$ has a diameter of at most $T/6$. An illustration of the progress of the synchronization is given in Fig 1.

![Image](image.png)

**Fig. 1.** (a) Each node is capable to transit and listen to its medium, using its radio. A chaotic scenario persists. (b) Each subset consists of nodes with the same firing impulse. (c) The subsets dissolve into each other to form a bigger set. (d) Synchronized single set

### 3.2 Proposition #2: Speculation on Self Synchronization by Electromagnetic Field

Interestingly realizing synchronization does not require symmetry breaking as it establishes symmetry. Therefore, it can be achieved with no leader and no means for symmetry breaking, such as unique identifiers or randomization. In a particle system, energy is limited and physical devices that waste that energy are eventually out of resources and do not participate in the transmission. Transmitting against the resistance of environmental EM field requires more energy use, forcing all the involved parties to find an optimal energy level (and therefore a synced frequency) of operation. This behavior was validated with models and experiments in the phase synchronization of pendulum clocks [11–15]. The energy (frequency) synchronization in a dynamical system is a function of its own source and the system parameters, defined by the supply power and the individual (and eventually, the collective) electromagnetic field of the nodes in the
system. Consider a distributed network, where the nodes fire at sporadic time intervals. Let \( e_m = \{a_1, a_2, \ldots, a_n\} \) be a vector, comprising the normalized electromagnetic field values of the respective nodes. We intend to further normalize the vector values to implement a threshold detection algorithm. The addition and cancellation of field values of the neighboring nodes forces the individual nodes to thrive for reaching the threshold (maximal transmitting frequency). This additive amplification of action due to a global EM application continues until a synchronized state is achieved in all nodes, that is, all nodes transmit with the maximum frequency, reach the maximal electromagnetic force value in a finite time. This (invisible) electromagnetic force is the added force acting on all the nodes, causing all the nodes to shift towards a particular direction, until all are synchronized and directed.

References

Sampling for effective database activity monitoring and anomaly detection

Hagit Grushka-Cohen1, Ofer Biller2, Oded Sofer3, Lior Rokach1, and Bracha Shapira1

1 Ben - Gurion University of the Negev, the Department of Software and Information Systems Engineering, Beer-Sheva, Israel
{hgrushka@post,bshapira,liorrk}.bgu.ac.il
2 IBM Guardium Security Division and IBM Cyber Security Center of Excellence, BeerSheva, Israel
ofer.biller@il.ibm.com
https://www.research.ibm.com/haifa/ccoe
3 IBM Guardium Security Division, Tel-Aviv, Israel
odedso@il.ibm.com

Abstract. Databases are at the heart of organizational IT infrastructure. Database activity monitoring (DAM) system monitor database transactions for anomaly detection and activity logging. In addition data security and privacy protection systems are widely used to help implement security policies and detect attacks and data abuse. To prevent data leakage many organizations monitor database operations in real-time (such logs were crucial in the multi-billion dollar suit of Google vs. Uber4).

Anomaly detection systems for DAM model the normal behavior of users or transactions in order to alert on abnormalities or high-risk behavior. An anomaly detection system can only learn and detect anomalies for the data that has been saved to logs [3, 5]. Transaction data monitoring and analysis is challenging due to the high velocity nature of database systems (up to 100K transactions per second). As a result and due to the cost of storage and compute, such systems audit only a portion of the vast number of transactions that take place and cannot save most of them to logs. Current methods rely on a manual policy directing which transactions to save. The user’s risk profile carries a lot of weight in such polices as does the security officer (SO) based on their perception of each role/person. Since data collection is bounded by the available resources and the quality of the policy rules, the anomaly detection system makes decisions based on a restricted data set. This data set may not represent all of the users or provide an in-accurate assessment of the risk of each user. In other words, the system is only looking under the lamp post.

Previous work studied the effect of sampling on anomaly detection systems mostly in the domains of network traffic monitoring and web-page retrieval with attempts to develop smart methods for sampling [4, 5, 3, 4].

These studies show that sampling introduces bias to the anomaly detection. In our work we do not change or aim to improve the anomaly detection. We only examine the effect of the collected data on the recalls of the existing anomaly detection (already in use by the DAM system). The common approaches for transactions monitoring are static and focus all efforts on monitoring high-risk users or activities to maximize the monitored risk. The data collected is skewed towards users which the SO estimated pose a high-risk. This approach has three major drawbacks: (i) bootstrapping the estimation of user risk may be hard, (ii) user risk changes over time and the risk estimation requires constant updates and is unlikely to remain accurate over time (iii) when an un-tracked user is flagged for a violation, the SO has no historic data on their activity and risk profile.

The ideal sampling strategy needs to maximize the value created from the collected logs while maintaining the human expert SO in the loop, logging transactions most likely to be risky while capturing the necessary data to identify changes in user behavior. The static expert policy approach is hard to set up and maintain as it is not exploring beyond the policy. Balancing exploration and exploitation in order to maximize a reward is an important problem for many domains, including reinforcement learning for agents and optimizing ads. In the stochastic multi-armed bandit problem (MAB) the goal is to maximize the reward from a slot machine with multiple arms. Various strategies for balancing exploration/exploitation have been studied [6].

In this work we examine sampling strategies for monitoring as a MAB problem, where the risk of the transactions logged is viewed as reward. Unlike the classic MAB problem the risk distribution of a user is not static, it may naturally change when the user changes their role or can change due to hacking, malware, or an employee being compromised. Another difference in data sampling for DAM is that at each round we are not sampling one user but sampling multiple users to monitor, i.e. multiple arms are pulled at each round (the collector logs all the transactions for a list of users). The reward function we are maximizing is based on the user risk sampled, we aim to sample the transactions which pose the highest risk.

We would like to examine the impact of the different sampling policies in two dimensions: quality and breadth of the logs collected, which are important for investigating issues and their impact, and anomaly detection quality in the DAM domain. We present a novel variant of the MAB problem to incorporate capacity, pulling multiple levers at each time frame and a novel sampling strategy for sampling MAB with capacity. We empirically compare this method, C-ε-Greedy sampling, to the rule based (policy) baseline of using knowledge of prior risk per user as well as Gibbs-prior sampling [1] using two data sets: simulated user transaction risk data and a ”real” data set of air quality measurements. We find that there are significant differences among the strategies in terms of the yield of discovered security events, the information coverage of the system regarding the overall activities in the database and the robustness to the change in a user’s risk distribution.
We found that C-\(\epsilon\)-Greedy sampling performs better for both objectives of log coverage and anomaly detection. For anomaly detection, exploration is crucial and the random factor of \(\epsilon\)-Greedy determines the detection recall of anomalies in users outside the high risk groups. We found C-\(\epsilon\)-Greedy to yield the best reward significantly improving over the expert-policy as well as Gibbs-prior sampling approach [1] in both simulated data and the environmental data set.

In terms of risk monitoring and logging, the results show that if the SO is well acquainted with the users when setting the policy the static strategy system can achieve fine results (67% of optimal reward for DAM and 65% for AQI). However, when the SO is not familiar with the user or failing to update user risk profile we saw the average reward drop. This matches the reported pain point of deploying and maintaining DAM systems, a great effort by the SO to create a good policy on deployment and then a constant need to maintain it as users switch positions or change behavior. Sampling methods biased towards exploration quickly gathered behavior information about all the users.

C-\(\epsilon\)-Greedy approach with 80% of the capacity aimed at exploitation achieved the best reward while still capturing most of the anomalies. Initializing C-\(\epsilon\)-Greedy with the existing SO knowledge data can assist the system to avoid the cold start problem and increase the SO trust of the system. Providing the expert with a single knob to turn in the form of exploration / exploitation rates makes setup and maintenance of monitoring systems far simpler and easy.

Our evaluation on data from the environmental domain have shown that this algorithm can work for other problem areas outside of sampling for security. It is possible to define many monitoring problems where each sample costs money as a C-MAB problem and spread the budget accordingly, even in small data problems. Further work can explore improving C-\(\epsilon\)-Greedy on a task made of many such data sets.

References

Assume, Guarantee or Repair
Student submission

Hadar Frenkel\textsuperscript{1} \*, Orna Grumberg\textsuperscript{1}, Corina Pasareanu\textsuperscript{2} and Sarai Sheinvald\textsuperscript{3}

\textsuperscript{1} Department of Computer Science, The Technion, Israel
\textsuperscript{2} Carnegie Mellon University and NASA Ames Research Center, CA, USA
\textsuperscript{3} Department of Software Engineering, ORT Braude College, Israel

Abstract. We present Assume-Guarantee-Repair (AGR) – a novel framework which not only verifies that a program is error-free, but also repairs the program, in case the verification fails. We consider simple C-like programs, extended with synchronous communication actions over communication channels. Our method uses compositional approach to modularly check the system for errors, and to repair it. We fulfill the two tasks simultaneously: in every iteration of the procedure, we either make another step towards proving safety of the (current) system, or remove the current vulnerability in a way that brings it closer to being safe. We manage to handle infinite-state systems by using a finite abstract representation. We describe our method and demonstrate the effectiveness of AGR on several examples using existing SMT solvers, learning, and reachability analysis tools.

1 Introduction

For large-scale systems, verifying that a system is error-free is a main challenge in the field of formal verification. Often, the verification process of such a system does not scale well. Compositional verification aims to verify small components of a system separately, and from the safety of the individual components, to conclude the safety of the entire system. This, however, is not always possible, since the safety of a component often depends on the behavior of its environment.

The Assume-Guarantee (AG) style compositional verification \cite{5,8} suggests a solution to this problem. The simplest AG rule checks if a system composed of components $M_1$ and $M_2$ fulfills a safety requirement $P$ by checking that $M_1$ under assumption $A$ fulfills $P$, and that any system containing $M_2$ as a component fulfills the safety assumption $A$. Several frameworks have been proposed to support this style of reasoning. Finding a suitable assumption $A$ is then a common challenge in such frameworks.

In this work, we present Assume-Guarantee-Repair (AGR) – a fully automated framework which applies the Assume-Guarantee rule, and while seeking a suitable assumption $A$, incrementally repairs the given program in case a vulnerability is found. Our framework is inspired by \cite{6}, which presented a learning-based method to finding an assumption $A$, using the $L^*$ \cite{11} algorithm for learning regular languages.

Our AGR framework handles communicating programs. These are infinite-state C-like programs, extended with the ability to synchronously read and write messages over communication channels. Sending messages over common channels as well as the ability of local computations, make communicating programs a good model for security protocols. We

\* This work was funded in part by the Hiroshi Fujiwara Cyber Security Research Center.

\*\* Contact information: hfrenkel@cs.technion.ac.il, +972-4-829-4315
model such programs as finite-state automata, similar to a representation as a control-flow graph. Its advantage, however, is in the ability to exploit an automata-learning algorithm such as $L^*$. The $L^*$ algorithm aims at learning a (potentially unknown) regular language $U$. Its entities consist of a teacher – an oracle who answers membership queries (“is the word $w$ in $U$?”) and equivalence queries (“is $A$ an automaton whose language is $U$?”), and a learner, who iteratively constructs a finite deterministic automaton $A$ for $U$ by submitting a sequence of membership and equivalence queries to the teacher.

In using the $L^*$ algorithm for learning an assumption $A$ for the AG-rule, membership queries are answered according to the satisfaction of the safety requirement $P$: If $M_1$ composed with $t$ is safe, then the trace $t$ in hand should be in the assumption $A$. Otherwise, $t$ should not be in $A$. Once the learner constructs a stable system $A$, it submits an equivalence query. The teacher then checks whether $A$ is a suitable assumption, that is, whether $M_1$ composed with $A$ is safe according to $P$, and whether the language of $M_2$ is contained in the language of $A$. The learning procedure aims at learning the weakest assumption $A_w$, which contains all the traces that composed with $M_1$ fulfill $P$. The key observation that guarantees termination is that the components in this procedure – $M_1, M_2, P$ and $A_w$ – are all regular.

Our setting is more complicated than the usual, since the traces in the components contain constraints over program variables, which are to be checked semantically and not syntactically. Moreover, $A_w$ above may no longer be regular.

Our method manages to overcome this problem in a way that still guarantees termination in case the system is safe, and progress towards safety, otherwise.

As we have described above, our goal is not only to compositionally prove that a system is error-free, but also to remove vulnerabilities in case they exist. An AG-rule can either conclude that the system is safe, or return a counterexample in the form of a trace that contains a vulnerability of the system. In our case, instead of returning the counterexample, we repair $M_2$ in a way that eliminates this vulnerability. In order to do so, we infer new constraints on the system, in a process called abduction 

We add the learned constraints to the set of actions of $M_2$ through the learning process to eliminate the erroneous trace. We then return to the verification stage and try to prove that the repaired program is safe, and so on.

Thus, AGR operates in a verify-repair loop, where each iteration runs a learning-based process to determine whether the (current) system is safe according to the property $P$, and if not, eliminates errors from $M_2$ while enriching the set of constraints derived from these errors, which often leads to quicker convergence.

**Implementation** To demonstrate the effectiveness of AGR, we implemented our AGR framework and ran it on 16 examples designed to exercise different aspects of the approach. We used Spacer [3] as a model checker to answer verification questions, and Z3 [2] to infer new constraints by abduction. We also integrated $L^*$ implementation from the LTSA tool [4] in our AGR framework. The experiments provide proof of concept to our algorithm. In the future we plan to robustify our implementation and to evaluate it extensively in the context of autonomous systems and security protocols.

---

4 We refer to program behaviors as traces, as they are represented by traces of the automaton.
2 Example

We now present an example to demonstrate the process. Figure 1 presents a simple program, $M_2$, which reads a password as long as it has less than four digits. Once the password is long enough, the program encrypts the password and sends it through a communication channel. The figure demonstrates the program and its representation as an automaton.

```plaintext
1: while(true)  
2:   pass = ReadInput;  
3:   while(pass <= 999)  
4:     pass = ReadInput;  
5:   pass2 := encrypt(pass);  
6:   return pass2;
```

![Fig. 1: Modeling a communicating program as an automaton $M_2$](image)

As mentioned in Section 1, we use a compositional approach in order to handle large systems. In Figure 2, we present the environment of $M_2$, namely $M_1$, which performs the encryption of the password read by $M_2$. The components $M_1$ and $M_2$ synchronize over common channels, i.e., $enc$ and $return$. The channel $enc$ is used to communicate the value of the password from $M_2$ to $M_1$, while the channel $return$ is used to return the value of the encrypted password (i.e., $pass2$) from the whole program.

Since we use automata learning in order to detect errors and repair them, the safety property is too given as an automaton. The property $P$ presented in Figure 2 requires that if the password $pass$ was entered and $pass2$ is the password returned after encryption, then $pass \neq pass2$. That is, the original password is not exposed. The property also requires the encrypted password to be of at most 64 bits to avoid overflow. Nothing in the composition of $M_1$ and $M_2$ enforces the password to be of at most 64 bits, and thus the systems does not satisfy the property. A possible violation is the concrete trace $t = read(2^{63})$, $2^{63} > 999$, $encrypt(2^{63})$, $pass2 = 2^{63} \cdot 2$, $return(2^{64})$. The trace $t$ violates the property $P$ since $pass2$ overflows with 64 bits. Therefore, we wish to eliminate $t$. We then use logical abduction in order to learn the new constraint $pass < 2^{64}$ and locate it in $M_2$ in a way that ensures that for the encrypted password it holds that $pass2 < 2^{64}$.

![Fig. 2: The system environment $E$ and the property $P$ given as automata](image)
References

Abstract. In distributed systems, participants may cooperate in processing tasks, such that loads are balanced among them. We present a local algorithm that (repeatedly) uses local (imbalance) criteria to transfer loads and iterate until all loads are balanced. Furthermore, the algorithm can handle a stream of incoming loads while balancing the loads. Namely, balance the arbitrary initial loads and at the same time distribute the incoming loads using a unified balancing potential function. The potential function measures the absolute difference between every individual node and the local (neighborhood) average load. Each participant acts to reduce the potential, by suggesting to transfer/receive individual load while keeping the average neighborhood load intact. The actual transfer takes place when both neighbors can reduce the potential. When the potential of every participant is zero, the individual loads equal the neighborhoods load, which in turn implies (by the fact that loads are positive) that the average load is equal in all the system, namely that loads are totally balanced.

Keywords: Distributed Algorithms, Local Algorithms, Self-Stabilization, Load Balancing.

1 Introduction and Related Work

Distributed systems can be viewed as a collection of computers and resources shared by active users. When the demand for resources increases the load balancing problem become important. In this scenario, systems have two states. (i) legitimate state, when loads are totally balanced in the system. (ii) illegitimate state, when loads are not balanced (and may stay imbalanced forever even if only new tasks are evenly distributed). Self-stabilizing [1, 2] algorithms ensure that whenever unexpected fault occurs (e.g., illegal/undesired/unplanned transfer of loads) that leaves the system in an arbitrary global state of the system.
the system will eventually converge to legitimate state from any such possible illegitimate state.

**Related work.** Two self-stabilizing algorithms for transferring the load around the network are presented by Flatebo et. al. [3]. In the first algorithm, each node compares its own load with the load of its neighbors. When a neighbor momentarily has a lower load, the node transfers any (if exists) new received load to this neighbor. The second algorithm aims at transferring new received load globally to lightly loaded nodes. These algorithms are in fact (new) load scheduling algorithm that transfer the (new) load to the least loaded nodes, but do not (repeatedly and possibly locally, up to a certain radius) balance the already assigned loads as we suggest here. Our paper presents local self-stabilizing load balancing algorithm for an arbitrary network of nodes which distributes existing and incoming loads evenly over the system in order to balance the load.

In other work, Feuilloley et.al. [4] introduced load balancing algorithm for balancing discrete and fraction load. Both kind of load balancing algorithms needs prepossessing steps e.g.: identifying bipartite graph, maximal matching and based on the concept of Match-and-balance algorithms. While our Algorithm fits in dynamic settings where no pre-processing is needed. We start from any arbitrary node with any arbitrary graph where topology can be dynamic.

## 2 Load Balancing Algorithm

**Algorithm 1:** Self-Stabilizing Local Load Balancing

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Input:</strong></td>
<td>An undirected graph $G = (P, E)$. Each node in graph knows the number of its neighbors. A node also knows the load of each neighbor.</td>
</tr>
<tr>
<td><strong>Output:</strong></td>
<td>A graph $G = (P, E)$ which have balanced load on each node.</td>
</tr>
</tbody>
</table>

1. **Load** ($p$): Load on node $p$
2. **Avg** ($p$): Computes the average load of node $p$ and its direct neighbors
3. **$S_{\text{min}}$** = **LessLoadedNodes** ($p, q$):
   - which returns the set of any neighbors $q$ of $p$ such that $\text{Load}(q) < \text{Avg}(p)$
4. **$S_{\text{max}}$** = **MoreLoadedNodes** ($p, q$):
   - which returns the set of any neighbors $q$ of $p$ such that $\text{Load}(q) > \text{Avg}(p)$
5. **Diff**($p, q$): Returns the load difference between the node $p$ and $q$
6. Each node will execute line 7 to 16 synchronously
7. Compute $S_{\text{min}}$ and $S_{\text{max}}$
8. Compute **Avg**($p$) with respect to each node
9. **if** $\text{Avg}(p) \leq \text{Load}(p) \land \text{Load}(q) < \text{Load}(p)$ **then**
10. - Compute **ExtraLoad** = **Diff**($\text{Load}(p)$, **Avg**($p$))
11. - Node $p$ notify to $S_{\text{min}}$ nodes.
12. - Neighboring nodes $S_{\text{min}}$ and node $p$ may take **ExtraLoad** equally from node $p$
13. **if** $\text{Avg}(p) > \text{Load}(p) \land \text{Load}(q) > \text{Load}(p)$ **then**
14. - Compute **ExtraLoad** = **Diff**($\text{Avg}(p)$, $\text{Load}(p)$)
15. - Node $p$ notify to $S_{\text{max}}$ nodes.
16. - Neighboring nodes $S_{\text{max}}$ may give **ExtraLoad** equally to node $p$
Self-stabilizing local load balancing algorithm appears in Algorithm 1. It uses a variable called $Load$. Each node consists of the present value of $Load$. Whenever a node executes step of load balancing algorithm, it computes the average of $Load$ with its direct neighbors using $Avg(p)$. Also, computes the difference of two loads using $Diff(p,q)$. Each node consists of two types of neighbor, some of them has load less than $Avg(p)$, which information is in $S_{\text{min}} = \text{LessLoadedNodes}(p,q)\text{ and some has load more than }Avg(p)$, which are in $S_{\text{max}} = \text{MoreLoadedNodes}(p,q)$.

Each node synchronously computes $S_{\text{min}}$, $S_{\text{max}}$ and, $Avg(p)$ after each iteration and update its values. If condition $Avg(p) \leq Load(p) \land Load(q) < Load(p)$ (line 9) satisfy, node $p$ computes $ExtraLoad = Diff(Load(p), Avg(p))$ and notifies to $S_{\text{min}}$ nodes, who may take the $ExtraLoad$ equally with itself (node $p$) from node $p$. In other case, if condition $Avg(p) > Load(p) \land Load(q) > Load(p)$ (line 13) is satisfied, node $p$ computes $ExtraLoad = Diff(Avg(p), Load(p))$ and notify to $S_{\text{max}}$ nodes, who may give the $ExtraLoad$ equally to node $p$.

To build intuition on the operation of the algorithm we start with two lemmas/observations for closure (safety) properties. Then we use a potential function for convergence (liveness) properties.

**Lemma 1.** During the execution of Algorithm 1, the maximum individual load does not increase and minimum individual load does not decrease.

*Proof.* Consider an initial configuration of graph with individual load, where $L_{\text{max}}$ holds the maximum individual load and $L_{\text{min}}$ holds the minimum individual. In case of maximum individual loaded node ($p_{\text{max}}$), whenever $Avg(p_{\text{max}}) \leq Load(p_{\text{max}})$ and for every neighbor $q$, $Load(p_{\text{max}}) \geq Load(q)$ then according to the condition in line 9 of the algorithm, $p_{\text{max}}$ does not take load form $q$. Similarly, in case of minimum individual loaded node ($p_{\text{min}}$), whenever $Avg(p_{\text{min}}) \geq Load(p_{\text{min}})$ and for every neighbor $q$, $Load(p_{\text{min}}) \leq Load(q)$ then according to the condition in line 13 in of the algorithm $p_{\text{min}}$ does not give load to $q$.

**Lemma 2.** Algorithm 1 guarantees that no more nodes will join set of $L_{\text{min}}$ and $L_{\text{max}}$.

*Proof.* Define set of minimum individual load $S\mathcal{L}_{\text{min}} = \{p \in N | L_p = Val_{\text{min}} \}$, where $Val_{\text{min}} = \min_{p \in N}\{L_p\}$. Consider a new node $p_j$ with load $L_j$. If $(L_j > L_{\text{min}} \land Avg(P_j) > L_{\text{min}})$ satisfy then new node $p_j$ may get the load from neighbors by which $L_j$ will never go less than $L_{\text{min}}$ and will not join $S\mathcal{L}_{\text{min}}$.

Similarly, Define set of maximum individual load $S\mathcal{L}_{\text{max}} = \{p \in N | L_p = Val_{\text{max}} \}$, where $Val_{\text{max}} = \max_{p \in N}\{L_p\}$. Consider a new node $p_j$ with load $L_j$. If $(L_j < L_{\text{max}} \land Avg(P_j) < L_{\text{max}})$ satisfy then new node $p_j$ may take the load from neighbors by which $L_j$ will never exceed $L_{\text{max}}$ and will not join $S\mathcal{L}_{\text{max}}$.

For the sake of analyzing the convergence time we use a potential function. The potential function is the sum of absolute values of $Load(p) - Avg(p)$ over
all $p$. We first establish that starting in an arbitrary non-balanced configuration there is at least one node that transfers load, this is based on a path from the node $p$ with maximum load to a node $q$ with minimum load, observing that there must be two neighboring nodes $r$ and $s$ for which $\text{Load}(r) > \text{Avg}(r)$ and $\text{Load}(s) < \text{Avg}(s)$. Staring with $p$ for which $\text{Load}(p) \geq \text{Avg}(p)$ if the next nodes on the path, $u$, have $\text{Load}(u) \geq \text{Avg}(u)$ then the first node $w$ with $\text{Load}(w) = \text{Load}(q)$ must have $\text{load}(w) < \text{Avg}(w)$.

We show that any load transfer in our algorithm reduces the potential, and when the potential is zero all loads are equal (since if they are not, there must exist neighbors, $p$ and $q$, with different loads, where $q$ has a smaller load, since the average load of $q$ is the load of $q$ and since $p$ has greater load, $q$ must have a neighbor $r$ with load smaller than $q$ has, continuing such a chain we obtain a contradiction for loads being positive. Thus, when the load difference is bounded by a constant, and the minimal transferred load is one load unit (allowing at most one load unit difference in balanced global state, when balancing with subset of qualified neighbors), and even if we ignore the (very likely) possibility of concurrent load transfers we get $O(n)$ time for the convergence. Note, that in case there are $O(n)$ concurrent balancing actions in every pulse, the convergence time is constant.

**Lemma 3.** Starting from any state, Algorithm 1 guarantees that the system will reach a legitimate state in $O(n)$ time, where the load is balanced across the distributed system.

### 3 Future Work and Conclusion

Many details and possible extensions are omitted from this brief announcement version. Still we note that our scheme and concept can be easily extended to (i) transfer loads only locally up to a certain distance, (ii) to transfer loads only if the local difference is above a given threshold, (iii) to work in asynchronous settings (not only by the use of a self-stabilizing synchronizer or neighborhood mutual-exclusion/rendezvous). The load transfer to neighbors is not restricted to the policy of uniform proposals to all qualified neighbors, other policies are possible as long as the proof assertions hold.

### References

Reinforcement Learning Method for Computing the Capacity of Communication Channels with Feedback

Ziv Aharoni, Oron Sabag, and Haim Permuter

Ben-Gurion University Beer-Sheva, Israel 8410501
{zivah,oronsa}@post.bgu.ac.il
haimp@bgu.ac.il

Abstract. One of the fundamental problems in Information Theory is solving the feedback capacity of noisy channels with memory. These channels' capacity is expressed analytically by an optimization problem over a multi-letter objective. This is the main obstacle to directly solving the feedback capacity analytically. In the last decade, some channels with memory where solved by formulating the capacity objective as a Markov decision process, and then applying dynamic programming algorithms. However, those solutions were restricted by the cardinality of the channels and were computationally tractable only for channels with binary alphabet. In this paper, we propose a novel method to compute the feedback capacity of channels with memory using reinforcement learning (RL). The main advantage of this approach is its computational efficiency, even for channels with with large alphabet size. The outcome of the RL algorithm sheds light on the properties of the optimal solution, which in our case, is the optimal input distribution of the channel. These insights can be converted into analytic, single-letter capacity objectives by solving corresponding lower and upper bounds. We demonstrate the efficiency of this method by analytically solving the feedback capacity of the well-known Ising channel with a ternary alphabet. We also provide a simple coding scheme that achieves the feedback capacity. The proposed methodology is another step in developing a powerful set of tools to solve channels with large alphabet size.

1 Introduction

Computing the capacity of a finite state channel (FSC) is a difficult task that has been vigorously researched in recent decades [3]. With the presence of feedback, the feedback capacity of a FSC can be expressed using the directed information [5, 11]. Despite the fact that the directed information is a multi-letter expression, it was shown that it can be formulated as a Markov decision process (MDP), which enables its computability using known MDP algorithms [4].

When formulated as a MDP, the feedback capacity of a FSC can be computed using a variety of methods, such as value and policy iteration. These algorithms have been proven very effective for channels with relatively small alphabets of
the channel input, output and state \([4, 2, 7, 6, 10, 9, 12]\). However, a principal drawback is that their computational complexity grows with the cardinality of the channel alphabet. Indeed, even for channel parameters from the ternary alphabet, these algorithms might be intractable.

We propose a methodology that uses RL to compute the feedback capacity of unifilar FSCs. Initially, a RL algorithm, namely the deep deterministic policy gradient (DDPG), is used to numerically estimate the feedback capacity. Then, the outcome of the RL algorithm is used to conjecture the structure of the analytic solution, which is expressed by a directed graph. The conjectured graph, that is called a \(Q\)-graph, can be used to compute analytic lower and upper bounds of the feedback capacity \([8]\). The bounds are guaranteed to coincide to the feedback capacity, in the case that the specific \(Q\)-graph of the analytic solution is extracted. Furthermore, the \(Q\)-graph can be used to derive a simple, capacity-achieving coding scheme of the channel. In our work, the proposed methodology enabled us to compute the feedback capacity of the Ising channel with a ternary alphabet (Ising3), and derive a capacity achieving coding scheme.

2 Main Results

The following theorems constitute our main results.

**Theorem 1.** The feedback capacity of a unifilar FSC can be estimated by a RL algorithm.

**Remark 1.** Theorem 1 is a computational result. Specifically, while previous estimations of the capacity were constrained by the cardinality of the channel parameters, we show that the RL algorithm is dimensional free.

Using the numerical results from the RL algorithm, one can deduce the analytic solution structure by a \(Q\)-graph \([8]\), which is used to compute the feedback capacity.

The following theorem is an instance of a known channel that we were able to solve using the numerical results from the RL algorithm.

**Theorem 2.** The feedback-capacity of the Ising3 channel is given by

\[
C_{fb} = \max_{p \in [0,1]} 2 \frac{H_2(p) + 1 - p}{p + 3},
\]

where \(C_{fb} \approx 0.961227\) for \(p \approx 0.263805\).

Furthermore, we derive a simple coding scheme that achieves the feedback capacity in Theorem 2.

**Theorem 3.** There exists a simple coding scheme for the Ising channel with general alphabet \(X\), with the following achievable rate:

\[
R(X) = \max_{p \in [0,1]} 2 \frac{H_2(p) + (1 - p) \log_2 (|X| - 1)}{p + 3}.
\]
Note that for $|\mathcal{X}| = 3$, the coding scheme achieves the capacity in Theorem 2.

The coding scheme is described by a repeated procedure that is given by the following:

**Code construction and initialization:**

- The message is a stream of $n$ uniform bits.
- Transform the message into a stream of symbols from $\mathcal{X}$, denoted by $\nu_1, \nu_2, \ldots$
  
  with the following statistics:
  
  $\nu_i = \begin{cases} 
  \nu_{i-1}, & \text{w.p } p \\
  \text{Unif}[\mathcal{X}\setminus\{\nu_{i-1}\}], & \text{w.p } 1-p
  \end{cases}$  

  (3)

  In words, a new symbol equals the previous symbol with probability $p$ and, otherwise, it is randomly chosen from the remaining symbols. This mapping can be done, for instance, by using enumerative coding, as shown in [1].

- At the first time, the encoder transmits $\nu_1$ twice.
- The decoder, upon receiving $y_1, y_2$, decode $\hat{\nu}_1 = y_2$ and sets $c = 2$.

The transmission procedure is given by the following:

**Encoder:**

1. If $\nu_t = \nu_{t-1}$ transmit $\nu_t$ twice and move to the next symbol.
2. If $\nu_t \neq \nu_{t-1}$ transmit $\nu_t$ once and view the last feedback $y$.
   
   (a) If $y = \nu_t$ move to the next symbol.
   (b) If $y \neq \nu_t$ transmit $\nu_t$ again and move to the next symbol.

**Decoder:**

1. If $y_t \neq \hat{\nu}_{c-1}$ then $\hat{\nu}_c = y_t$, increment $c = c + 1$.
2. If $y_t = \hat{\nu}_{c-1}$ then wait for $y_{t+1}$, set $\hat{\nu}_c = y_{t+1}$, and increment $c = c + 1$.

3 Conclusions

We derived an estimation algorithm of the feedback capacity of a unifilar FSC using RL. The RL approach addresses the cardinality constraint and establishes RL as a useful tool for channels with high cardinality. We provided an example over the Ising3 channel, where we used the insights provided by the numerical results to analytically compute its feedback capacity. Furthermore, we showed a simple capacity-achieving coding scheme for the Ising3 channel with feedback.

In the future, we plan to generalize the results of the Ising3 to solve the Ising channel for any alphabet size. Then, we plan to solve different channels numerically and, hopefully, establish methods to induce their analytic solution, and capacity-achieving coding schemes. On the other hand, we plan to use the analytic solution of solved channels as a benchmark for improving RL algorithms.
Bibliography


Visual Analytics for Unsupervised Anomaly Detection of Multivariate Sensor Streams

Efrat Vilenski and Jonathan D. Rosenblatt
Dept. of Industrial Engineering and Management, Ben-Gurion University of the Negev, Israel. johnros@bgu.ac.il

Abstract. IoT revolution influenced many domains such as industrial, healthcare, energy and many more. New challenges arise, including: IOT analytics and dashboards for remote operation centers that have to make decisions based on multivariate sensor streams. Understanding multivariate data is a formidable task, as it can be very confusing for humans to comprehend and understand data with many variables.

We present a visual analytics framework, suited for unsupervised anomaly detection of multivariate sensor streams. The framework was developed and implemented in two real world use-cases of precision agriculture and precision livestock farming. Our work also provides visual analytics to aid analysts with tasks from the unsupervised anomaly detection pipeline.

Keywords: Multivariate Data · Unsupervised Anomaly Detection · Visual Analytics · Internet of Things · Alerts system

Advances in sensors technology and wireless communications have enabled low-cost installations of large sensor networks and contributed to the IoT revolution in many domains such as industrial, healthcare, energy, smart city, smart buildings and many more. New challenges arise, including: IOT analytics and dashboards for remote operation centers that need to analyze vast amount of incoming correlated streams; Maintaining sensors network integrity in order to make decisions only based on good sensors; Analyzing IoT stream that are based on dynamic processes in unstable environments.

Our research deals with unsupervised anomaly detection of multivariate sensor streams of real world use-cases from several domains: precision agriculture, precision livestock farming and machine predictive maintenance.

Many of today’s data sets are multivariate, where several measurements are taken per observation. In some cases, it will be sufficient to isolate each variable and analyze it separately. This “massively univariate” approach is simple and popular, but in many cases it is insensitive to sensor anomalies. The major flaw of a massively univariate analysis is the inability to detect multivariate anomalies. As an example, consider a patient in intensive care: it is possible that the pulse and temperature are each reasonable, but jointly their values are unprecedented. This type of anomaly, which clearly requires attention, cannot be detected by a massively univariate approach, and requires a multivariate approach.
In the past, when computational power was limited, most advances in multivariate data analysis were primarily mathematical, or relied on sums of squares. Advances in computer science increased the applicability of existing multivariate techniques, and renewed interest in the creation of new ones. Recent advances in visual analytics also enabled new options for interactive data exploration and opened opportunities for creating new techniques for understanding complex multivariate data.

1 Unsupervised Anomaly Detection

Our work focuses on visual analytics to aid with tasks from the unsupervised anomaly detection pipeline of multivariate sensor streams, which is a complex challenge that involves several aspects:

Data Quality Data from sensor networks, may have data quality issues. Important step is ensuring good data quality and if needed cleaning the data.

Anomaly Scoring Anomaly detection relies on scoring the abnormality of each sensor using various algorithms, possibly ensembling them into one single score. The larger the score, the stronger the indication of an anomaly.

Pre-processing and Transforming Our algorithms rely strongly on correlations. We put great effort into pre-processing the data and rescaling variables so that distributions are symmetric, relations (approximately) linear, etc.

Interactivity Interactive visualizations are a cornerstone of our pipeline. They help the analyst to understand and explain the source of an anomaly. Which is imperative with multivariate data. Visualizations also help in identifying anomalous patterns, and evaluating scoring algorithms.

Sliding Temporal Window Anomalies are declared based on sensor readings from recent history. This is achieved by choosing a time window of fixed length and sliding it along the streaming dataset. The anomaly scores are calculated per sliding window, the window’s length balances between detection power and speed of detection.

Ensemble Scores Ensembles of anomaly scores refers the aggregation of multiple scores into a single score. The motivation for ensembling is that each algorithm may be sensitive to a different departure from the steady state. For each type of signal/pattern/anomaly there is a different optimal scoring rule. Since we do not know a-priori the nature of the anomaly, we run several scoring rules simultaneously, and aggregate them.

Thresholds and Classification Given a set of classes (such as “Normal”, “Anomaly”), anomaly scores and thresholds, sensors can be classified. The challenge is to find thresholds that reduce misclassification errors.

2 About MultiNav

We developed prof of concept open source framework, suited for unsupervised anomaly detection of multivariate data. The framework is based on learning’s from specific use-cases, generalized for other potential new use-case.
MultiNav.js was designed and developed specifically for dashboards used in remote operation centers, that need to make constant decisions based on multivariate sensor streams. The framework was tested on two real world use cases:

**System for detecting anomalous dendrometer sensors.** Dendrometer sensors provide data used for plot-specific irrigation decisions, with critical implications for yields and water savings. To aid the identification of malfunctioning dendrometer sensors, we developed a pipeline for detecting various types of anomalies and investigating their root causes using visual analytics system. Our pipeline is unique not only in that it borrows from web technologies to provide interactivity, but also because it incorporates detection algorithms from several fields, such as robust multivariate statistics, unsupervised machine learning, and social-network analysis. The system was developed and implemented in production environment, with collaboration with Phytech, an agricultural plant-based IoT company.

**Alerting system for herd management.** Goals of the system: identify anomalous animals with multivariate methods; Provide visual tools to examine and understand the anomalies. The systems receives data from a novel system that contains electronic scales and drinking behavior sensor designed for automatic small-ruminant monitoring. The water source is an attraction point, that voluntarily attract the animals to the scale few times per day. The system enable creation of a cost effective alerts management tools that aid farmers in making evidence-based decisions, in real time, regarding herd nutrition, health and welfare. The system can track drinking behavior, water consumption and body weight measurements in small ruminants (sheep and goats). System is developed in collaboration with Volcani Reasrch center, PLF lab.

Based on Multiinav.js we are creating MultiNav R package, which is intended to aid power analyst / data scientist, with tasks related to unsupervised anomaly detection of sensor streams. While there are other packages that provide sufficient support for anomaly detection and multivariate data, they are not designed for interactivity. MultiNav provides both default algorithms for anomaly detection and visual aids to help with specific common tasks in the pipeline. Few examples:

- Exploratory data analysis for understanding multivariate data.
- Identify anomalies: Are there sensors that behave different from the rest? is the difference consistent across time or only manifested in a specific time window? Does a specific sensor behaves differently compared to his history?
- Evaluate anomaly scores: For a specific window, compare scores from several anomaly methods, Which methods have consistent scoring? which methods bring new information?
- Track performance of anomaly scores over time.
- Given an anomaly, understand the nature of the multivariate anomaly.

**References**

LOCALIZATION OF DATA INJECTION ATTACKS ON DISTRIBUTED M-ESTIMATION

Or Shalom\textsuperscript{*}, Amir Leshem\textsuperscript{*}, Anna Scaglione\textsuperscript{†}

\textsuperscript{*}Faculty of Engineering, Bar-Ilan University, Ramat Gan, Israel.
\textsuperscript{†}School of ECEE, Arizona State University, Tempe, AZ, USA.

ABSTRACT
This paper describes a distributed statistical estimation problem, corresponding to a network of agents. The network may be vulnerable to data injection attacks, in which attackers control legitimate nodes in the network and use them to inject false data. We have previously shown [1] that the detection metric by Wu et al. in [2], is vulnerable to sophisticated attacks where the attacker mixes normal behavior and false data injection. In this paper we propose a novel metric that can be computed locally by each agent to detect and localize the novel attack in the network in a single instance.

Index Terms—Distributed projected gradient, Decentralized optimization, Data injection attacks, Convex optimization, M-Estimators

I. INTRODUCTION

DECENTRALIZED multi-agent optimization is an important problem in distributed computation. These algorithms rely on local computations as well as in-neighborhood communication to achieve their common goal of minimizing a common cost function or converging to a stable point. As these networks gain popularity [3–11], it has become apparent that they are sensitive to false data injection which can steer the network’s final state, see [2], [12–22] for examples. The structure of an independently self-updating network, which has been the main advantage of these methods, can turn into a vulnerability by allowing an attacker which controls a single node to have a global impact. This type of attack cannot be detected using cryptographic techniques, since the attacker controls a legitimate node in the network. This paper focuses on the problem of localizing attacks on distributed statistical estimation, using M-estimators [23] in general and maximum-likelihood in particular, using the distributed projected gradient (DPG) algorithm. We begin with a novel data injection attack scheme, and its effects on decentralized optimization algorithms, and primarily DPG [3]. We propose a novel, more sophisticated attack scheme which is invisible to all previous detection methods. This attack scheme is shown to be always successful on communication networks, even when the network is dynamically changing over time. We then propose a new metric, computed locally by each agent over time, to detect and localize an attacker in the network, allowing the users to ignore the attacker and reach convergence to the true optimal state. In contrast to previously proposed techniques, our scheme can detect and mitigate the attack in a single run of the algorithm.

Notations: We use boldfaced letters to denote vectors and boldfaced uppercase letters to denote matrices. For a vector $\theta, [\theta]_i$ denote its $i$-th element, similarly, for a matrix $A$, $A_{ij}$ denotes its $(i, j)$-th element.

II. PROBLEM FORMULATION

Consider a grid of sensors measuring independent random processes that depend on a joint parameter. In order to extract this parameter, the sensors solve a M-Estimator problem

$$\arg\min_{\theta} \sum_i \rho(x_i, \theta)$$

where $\rho(x_i, \theta)$ is the $i$-th agent’s private objective function and $\theta$ is an unknown parameter vector. M-Estimators generalize the maximum-likelihood by replacing the likelihood function with a generalized function, $\rho(x_i, \theta)$, for each user $i \in V$. We assume that the process is i.i.d. between sensors. We consider a distributed setup where agents do not share their private information $x_i$.

II-A. Preliminaries

Consider an undirected, time varying graph $G(t) = (V, E(t))$ defining a network of $N$ agents, where $V = \{1, ..., N\}$ is a set of $N$ nodes (agents) and $E(t) \subseteq V \times V$ denotes the connections between the nodes for some time $t \in \mathbb{N}$. For each node $i$, we define $N_i \subset V$ as the neighborhood set of agent $i$, as $N_i \equiv \{j : (j, i) \in E\}$, note that $E = \cup_{t=1}^{\infty} E(t)$. We mark the $i$-th agent state for some time $t \geq 0$ as $\theta_i(t)$.

II-B. Distributed stochastic M-estimation

The $N$ agents share the common goal of minimizing a joint objective function in a distributed manner; i.e., solve the following optimization problem:

$$\min_{\theta} h(\theta) := \frac{1}{N} \sum_{i=1}^{N} h_i(\theta), \quad \text{ s.t. } \theta \in C$$

This work is supported by the NSF CCF–BSF 1714672 and grants ISF–1644/18 and ISF-NRF 2277/16.
where $\mathcal{C} \subseteq \mathbb{R}^p$ is a closed, convex, compact set and $h_i : \mathbb{R}^p \to \mathbb{R}, h_i(\theta) = \rho(x_i, \theta)$ is a private differentiable\(^1\) function over $\mathcal{C}$, known to the $i$-th agent alone. In our problem, we assume that $x_i$ are i.i.d. given any value of $\theta$; i.e., the objective function $\rho(x, \theta)$ is the same function, and is known to all nodes. However the specific realization $\rho(x_i, \theta)$ is private since each node has its own data. We mark the optimal solution of the optimization problem as $h^* = h(\theta^*)$, where $\theta^* \in \mathcal{C}$ is the optimal parameter. In this paper we assume that all $h_i(\theta)$ are convex. The M-estimation can be solved distributedly using a stochastic distributed projected gradient algorithm.

Let $G(t)$ be the graph associated with a weighted adjacency matrix, $W(t) \in \mathbb{R}^{N \times N}$, where $W(t)$ satisfies:

**Assumption 1:** $W(t)$ fulfills the next terms for $t \geq 0$:
- $W(t)$ is a symmetric, nonnegative, bi-stochastic matrix.
- If $(i, j) \in E(t)$ then $W_{i,j}(t) \geq \xi$ for some $\xi \in (0, 1)$.
- If $(i, j) \notin E(t)$ then $W_{i,j}(t) = 0$.

**Assumption 2:** There exists $B < \infty$ such that the graph $(V, \bigcup_{t=1}^{B}E(t + l))$ is connected.

The distributed projected gradient (DPG) method [3] solves the optimization problem shown in (2) by performing the recursion:

$$
\theta_i(t) = \sum_{j=1}^{N} W_{ij}(t) \theta_{j}(t), \quad \forall i \in V, t \geq 0,
$$

**Assumption 3:** $\eta(t)$ is a time-varying step size satisfying $\sum_{t=1}^{\infty} \eta(t) = \infty$ and $\sum_{t=1}^{\infty} \eta^2(t) < \infty$.

**Proposition 1:** Under assumptions 1-3, for a compact space, the joint objective function asymptotically reaches a minimum, as seen in [4], [12].

$$
\lim_{t \to \infty} h(\theta(t)) = h^*
$$

Our goal is to detect malicious nodes in the network that attempt to destroy the distributed computation by injecting false data.

**III. DATA INJECTION ATTACKS**

Consider a distributed M-estimation, where some nodes are malicious and inject false data into the network. We divide the set of nodes, $V$, into two subsets: $R \subset V$ is the set of reliable agents and $A := V \setminus R, A \neq \emptyset$ is the set of attackers. Let $n_\alpha = |A|$ be the number of attacking nodes. The attackers’ goal is to steer the network’s final state $\lim_{t \to \infty} \theta(t)$ to a target state of their choice, while remaining transparent to the network. To do so the attackers follow a deceiving update rule of their choice while the trustworthy agents follow the DPG update rule as shown in (3). A previous work [2] suggested a straightforward attack scheme, as well as a detection method. Unfortunately, the attack scheme can be modified to evade this detection method. In this section we present a novel improved attack method, and later we propose a combined detection and localization scheme, computed locally by each agent in a single instance of the algorithm.

**III-A. Novel Attack Scheme**

The new attack scheme proposed here is a mixture of two update rules:

- The trustworthy agents’ DPG update rule.
- The straightforward attacker’s update rule [2].

To combine both update rules we generate a new time-varying proportion coefficient marked as $g(t)$.

**Assumption 4:** The new proportion coefficient $g(t)$ fulfills the following conditions:
- For all $t \geq 0, 0 \leq g(t) \leq 1$.
- $g(t)$ decreases over time, i.e. $g(t+1) < g(t)$.
- $g(0) = 1, \lim_{t \to \infty} g(t) = 0$.

Note that the limitations on $g(t)$ is very minor, we have no assumptions neither on the convergence rate to 0, nor on any relation to other components in the network’s convergence process.

The new proposed attack scheme is

$$
\theta_j(t + 1) = g(t) \times DPG(\theta_j(t))
$$

$$
+ (1 - g(t)) \times (\alpha_0 + z_j(t + 1)), \quad \forall j \in A
$$

where $DPG(\theta_j(t))$ refers to (3), $\alpha_0$ is the attacker’s desired final state and $z_j(t)$ is a zero mean and $\sigma^2(t)I_p$ variance random noise, vanishing a.s. over time and satisfying the expected convergence rate of the graph for all $j \in A$.

The result of implementing the new attack scheme on the network forces the initial state of the attackers’ nodes to be similar to that of the trustworthy agents. Therefore, the detection scheme in [2] fails. The network’s convergence to the attacker’s desired state, under the new attack scheme, is demonstrated in Figure 1. Looking at Figure 1, we can see that the entire convergence process under the proposed attack scheme can be divided into 3 time periods. Prior to the attack ($t < T_g$), during the attack ($T_g \leq t < T_\infty$) and post convergence ($t \geq T_\infty$).

**Assumption 5:** The objective function gradient, $\nabla h_i$, is bounded for each $i \in V$, s.t.

$$
|\nabla h_i(\theta)| \leq \frac{q}{2}, \quad q = [q_1, \ldots, q_P]
$$

**Proposition 2.** Under the previous assumption and the proposed attack scheme in (5), the network converges to the attacker’s desired state $\alpha_0$.

$$
\lim_{t \to \infty} ||\theta_i(t) - \alpha_0 ||_\infty = 0, \quad \forall i \in V,
$$

Proof in [1, Appendix A].
IV. LOCALIZING THE ATTACKERS

In this section we propose a novel low-complexity metric, computed over time by each agent, to detect and localize the attackers instantaneously, running the algorithm for a single instance (as opposed to previous works, including [1]). Once we find the attackers we can ignore their data and have a trustworthy network solving (2), reaching an optimal state. We run the recursive DPG algorithm as seen in (3), where the attackers are following the proposed attack scheme shown in (5). The algorithm runs for some time index marked as $T_{\infty}$, sufficient for convergence.

In this method, each agent compares the state updates received from each agent in its neighborhood with the rest of the agents in the neighborhood over time, after reaching convergence. The agents are assumed to be identically distributed and therefore if an agent is malicious and updates differently, it will stand-out and be considered as an outlier.

Denote the two hypotheses:

$H_{i,j}^{0}$ - Agent $j \in \mathcal{N}_{i}$ is not an attacker; i.e., $j \notin A$.
$H_{i,j}^{1}$ - Agent $j \in \mathcal{N}_{i}$ is an attacker; i.e., $j \in A$.

The proposed metric, computed over time by each agent is given by

$$\Delta U_{i,j} = \frac{1}{\Delta T} \sum_{t} U_{i,j}(t) \bigg|_{H_{i,j}^{1}} - \delta_{u}$$

where

$$U_{i,j}(t) = u_{i,j}(t) - \text{median}\{u_{i,l}(t) : l \in \mathcal{N}_{i} \backslash \{j\}\}$$

$$u_{i,j}(t) = \frac{\|\theta_{i,j}(t+1) - \theta_{i,j}(t)\|}{\eta(t)} dt$$

$\theta_{i,j}$ is the data that agent $i$ receives from agent $j$ and $\delta_{u}$ is a predefined threshold. An example of the proposed detection and localization scheme in a single instance for different integration time, $\Delta T$, can be seen in Figure 2.

![Fig. 1. An example of the novel attack scheme (for $P = 1$). The network reaches convergence to an unstable state that drift over time to $\alpha_{0}$. Note that in this figure a weak attack was drawn in order to emphasize the three periods of the algorithm, generally the attack is controlled by the attackers.](image1)

![Fig. 2. An example of the new proposed detection and localization scheme, $\Delta U_{i,j}$, computed for different averaging time, $\Delta T$, in a dynamically changing random graph ($P = 1$). It is easy to notice that the attacker is exceptional.](image2)

V. LOCALIZATION SCHEME ANALYSIS

In the proposed localization scheme, we look at the tail of the algorithm (post convergence, at times $t \geq T_{\infty}$). We can show analytically, that the attacker will behave differently than other agents in most cases. Due to the nature of the algorithm, the attacker constantly oppose the trustworthy agents’ gradient update and therefore will standout off other agents.

For a trustworthy agent, $i \in R$, and an attacker, $j \in A$, the states after convergence are

$$\theta_{i}(t+1) = \alpha_{0} + \epsilon_{i}(t) - \eta(t) \nabla h_{i}(\alpha_{0})$$

$$\theta_{j}(t+1) = g(t)(\alpha_{0} + \epsilon_{j}(t) - \eta(t) \nabla h_{j}(\alpha_{0})) + (1 - g(t))(\alpha_{0} + z_{j}(t))$$

where $\forall i \in V$, $\epsilon_{i}$ is a random noise, vanishing according to the convergence rate of the network. To compute $u_{i,j}(t)$, we have to compute the following

$$[\theta_{i}(t+1) - \theta_{i}(t)]/\eta(t) = \epsilon_{i}(t)/\eta(t) - \nabla h_{i}(\alpha_{0})$$

$$[\theta_{j}(t+1) - \theta_{j}(t)]/\eta(t) = [z_{j}(t) - \epsilon_{j}(t-1)]/\eta(t)$$

where $\epsilon_{i}(t) = \epsilon_{i}(t) - \epsilon_{i}(t-1)$.

It is easy to see that from the definition of $\epsilon_{i}(t)$, $\forall i \in V$, and $z_{j}(t)$, they both rely on the convergence rate of the network; i.e. rely on $\eta(t)$. That means that the failure of the post-convergence localization scheme, relies solely on the objective function $h_{i}$. Therefore, in order for the attacker to
bypass the given localization test, the attacker has to steer the network to a state that satisfies

\[ \nabla h_i(\alpha_0) \approx \lim_{t \to \infty} \frac{e_i(t) - z_j(t)}{\eta(t)} \]  

(12)

VI. SIMULATIONS

This section presents the simulations conducted and the results are shown in the figures below. In the simulations we generated an “Erdos–Renyi” random-graph, consisting of \( N \) agents (\( N = 50, 100, 500 \)) with random edge probability, \( 0 \leq p \leq 1 \). We generated the adjacency matrix \( W(t) = I - \frac{1}{N} S + \frac{1}{N} (P + P^T) \), where \( P \) is a random \( N \times N \) matrix and \( S \) is a diagonal matrix consisting of the column sum of \( (P + P^T) \).

We assume that there is a majority of trustworthy agents in each neighborhood. In a case that the majority of neighbors in a trustworthy agent’s neighborhood are attackers, this agent is likely to consider the attackers as trustworthy agents.

VI-A. Example: Detecting, localizing and eliminating 5 coordinated attackers while estimating logistic distribution mean:

In this example we present detection and localization in a single instance. After the network converges, each agent periodically look for attackers using (8). If a trustworthy agent suspect another agent in his neighborhood, the suspicious agent’s data will be ignored for a given time period. When the given time period expires, the suspicious agent is being examined again with a more rigorous threshold, if the agent remain suspicious his data will still be ignored. If not, the trustworthy agent will use the data again (and so on). By ignoring the data from suspicious agents, we make sure that the network is reaching convergence to the true optimal state.

We assume that each agent holds a single measurement, (in this example \( P = 1, 3, 5 \) and 7) consisting of the desired signal with zero mean noise. Our goal is to extract the desired signal by eliminating the noise from the given measurement in a distributed manner. To simulate the problem we initialize the agents’ state with values generated from a logistic distribution with parameters \( (\mu, \Sigma) \) where \( \Sigma = \text{diag}(\sigma_1, \ldots, \sigma_P) \). The agents solve a distributed problem with the following private objective function:

\[ h_i(\theta) = 2 \sum_{p=1}^{P} \log \left( 2 \cosh \left( \frac{[x_i]_p - [\theta]_p}{2\sigma_p} \right) \right) + C \quad (13) \]

for all \( i \in V \), where \( C \) is a constant number known to all agents, \( x_i \) is the measured signal for some agent \( i \) and \( \theta = \mu \), the desired variable. An example of the network convergence to the true optimal state after attackers elimination can be see in Figure 3. The localization scheme ROC, for different \( P \), is depicted in Figure 4. Looking at the simulations, we see that it is easier to perform a successful yet transparent attack on high dimensional problems.

\[ \text{Fig. 3. Detection, localization and elimination of 5 coordinated attackers in a single instance, as explained in VI-A. In each subplot, presented different dimension of } \theta(t) \text{ (} P = 3 \text{). We see that the trustworthy agents converge to the true optimal state, } \theta^* \text{, while the attackers converge to } \alpha_0. \]

\[ \text{Fig. 4. ROCs temporal difference localization performance at the neighboring agents of the 5 coordinated attackers. } \theta \in \mathbb{R}^P, \, P = 1, 3, 5, 7. \]

VII. CONCLUSIONS

In this paper we presented a novel attack on distributed multi-agent optimization. We then presented a combined detection and localization method in the case of distributed M-Estimators with i.i.d agents data. In an extension of this work we present detailed proofs of the exponential bounds for \( P_{FA} \) and \( P_D \), as well as the propositions presented in this paper.
VIII. REFERENCES


[22] Wei Xu, Zhengqing Li, and Qing Ling, “Robust decentralized dynamic optimization at presence of malfunctioning agents,” .

Randomly Rotate Qubits Compute and Reverse
Information-Theoretically Secure Quantum Gate Computation and Applications
(PhD Track Technical Report Abstract)

Dor Bitan\textsuperscript{1} and Shlomi Dolev\textsuperscript{2}

\textsuperscript{1} Dept. of Mathematics, Ben-Gurion University of the Negev, Israel, dorbi@post.bgu.ac.il.
\textsuperscript{2} Dept. of Computer Science, Ben-Gurion University of the Negev, Israel dolev@cs.bgu.ac.il

\textbf{Abstract.} Quantum computations are typically based on representing the classical bits 0 and 1 as the pure state qubits of the computational basis $|0\rangle$ and $|1\rangle$, utilizing quantum phenomena of superposition and entanglement. We explore the case of using randomized bases (instead of the typically used computational basis) to represent classical bits and perform information-theoretically secure evaluation of quantum gates. We suggest an information-theoretically secure encryption scheme, based on randomized bases, and discuss its quantum homomorphic properties. We demonstrate the usefulness of using randomized bases in an efficient secure QKD protocol and other applications.

\textbf{Keywords:} Homomorphic encryption, Quantum computing, Information-theoretic security

\textsuperscript{†} This paper is a Ph.D. student paper submission to the third International Symposium on Cyber Security Cryptology and Machine Learning (CSCML 2019).

\textsuperscript{*} We thank the Lynne and William Frankel Center for Computer Science, the Rita Altura Trust Chair in Computer Science. This research was also partially supported by a grant from the Ministry of Science and Technology, Israel & the Japan Science and Technology Agency (JST), and the German Research Funding (DFG, Grant #8767581199). We also thank Dani Berend for discussions, comments and suggestions throughout the research.
Introduction. Delegation of computation, while preserving confidentiality of the data (and sometimes even the program), is a challenging practical task. In particular, there is a lot of use in perfectly secure schemes that can delegate data and computation to be processed by an untrusted server. Existing solutions are based on distributed secure multi-party computation or computationally secure fully homomorphic schemes. Still, there exists no single server information-theoretically secure solution. In light of numerous results, e.g. [Sho94,Gro96,Jor18], established in the growing field of quantum computation, it is natural to ask whether an efficient information-theoretically secure solution may be found in the scope of quantum computing. In 2014, it was shown in [YPDF14] that it is impossible to construct an efficient information-theoretically secure quantum homomorphic encryption scheme. Specifically, it was proved that, the size of the encryption of an information-theoretically secure quantum homomorphic encryption scheme must grow exponentially in the input size. Hence, practical perfectly secure encryption systems can only be used to evaluate a subset of all possible functions, e.g., [Lia13,TKO+16,OTF18]. Several works used computationally-secure classical fully homomorphic encryption schemes to construct computationally-secure quantum homomorphic encryption schemes, e.g., [DSS16].

Our contribution. We suggest here a new approach to encrypt and outsource storage of classical data while enabling quantum gate computations over the encrypted data. Our method is based on using randomized bases to represent strings of classical bits, instead of the typically used computational basis \{0,1\}, and is perfectly secure (rather than only computationally secure).

The Random-Basis Encryption Scheme. The inability of determining the coordinates of an arbitrary qubit, given a realization of it, gives rise to the following encryption scheme, which allows a user to outsource storage of confidential (classical) information to an untrusted quantum server. We assume that both the user and the server can: (a) generate physical realizations of qubits in the computational basis; (b) manipulate these physical realizations of qubits using quantum gates (unitary operators); (c) transmit these qubits between each other; (d) measure qubits. The message space, \(M\), and the ciphertext space, \(C\), are both \{0,1\}. The key space, \(K\), is \([0,2\pi]^2\). The scheme consists of the following three algorithms:

**Gen (key generation):** Output a uniformly random pair \((\theta,\varphi)\) from \([0,2\pi]^2\).

**Enc (encryption):** On input message \(b \in M\) and a key \(k = (\theta,\varphi)\), first generate the qubit \(|b\rangle\). Now, Let 
\[
K = \begin{pmatrix}
\cos \frac{\theta}{2} & e^{i\varphi} \sin \frac{\theta}{2} \\
-e^{-i\varphi} \sin \frac{\theta}{2} & e^{i\varphi} \cos \frac{\theta}{2}
\end{pmatrix} \in M_2(C),
\]
and apply \(K\) to \(|b\rangle\) to obtain \(|q\rangle = K |b\rangle\). Finally, output \(|q\rangle\).

**Dec (decryption):** On input ciphertext \(|\psi\rangle\) and a key \(k = (\theta,\varphi)\), let \(K^\dagger\) denote the conjugate transpose of \(K\), where \(K^\dagger\) is as in \(\text{Enc}\) and apply \(K^\dagger\) to \(|\psi\rangle\). Then, measure \(K^\dagger |\psi\rangle\) in reference to the computational basis and output the outcome of the measurement.

The matrix \(K\) defined in the scheme is the unitary matrix whose columns are the elements of an orthonormal basis denoted \(B_{(\theta,\varphi)}\). We refer to \(\text{Enc}\) as taking the elements of the computational basis to the elements of the random orthonormal basis \(B_{(\theta,\varphi)}\) and \(\text{Dec}\) takes the elements of \(B_{(\theta,\varphi)}\) to the elements of the computational basis. Analysis of the security and correctness of the scheme appear in the full version of this paper. A user may use our scheme to outsource the storage of a string of bits, \(x = b_1 b_2 \ldots b_l\), to an untrusted quantum server by encrypting each bit \(b_j\), and transmit to the server the tensor product \(|q\rangle = |q_1 q_2 \ldots q_l\rangle\).

Quantum Gate Computations. We explore the consequences of homomorphically applying quantum gates to the ciphertext by the cloud. First we note that any gate which commutes with the family of the encryption gates \(K\), may be homomorphically and securely applied to the encrypted data. Several unitary operations are typically used in quantum computing. We investigate the consequences of applying some of these typically-used quantum gates to a randomized basis \(B_{(\theta,\varphi)}\) encryption of classical data. The results of that investigation are listed below.

The **NOT gate.** The NOT gate is the unitary transformation that interchanges the elements of the computational basis: \(|b\rangle \rightarrow |1-b\rangle\). In the case \(\varphi = \pm \frac{\pi}{2}\), applying a NOT gate to elements of \(B_{(\theta,\varphi)}\) we get the same effect as when applying it to an element of the computational basis. Consequently, NOT gates may be homomorphically applied to encrypted data. The algorithm \(\text{Gen}\) may be adjusted to output keys with \(\varphi = \pm \frac{\pi}{2}\) without compromising the security. From now on we assume \(\varphi = \pm \frac{\pi}{2}\).

The **CNOT gate.** The CNOT gate is the two-qubit gate which when applied to the elements of the computational basis of \(\mathbb{H}^2\otimes\mathbb{H}^2\), if the first (control) qubit is \(|0\rangle\) then the second (target) qubit is left unchanged, and if the first qubit is \(|1\rangle\), then a NOT gate is applied to the second qubit. Applying a CNOT gate to the elements of a partially-randomized basis \(\{0,1\} \otimes B_{(\theta,\pm \frac{\pi}{2})}\) of \(\mathbb{H}^2\otimes\mathbb{H}^2\) keeps the target-control structure. Hence, CNOT gates may be homomorphically applied to systems of two qubits when the control qubit is
an element of the computational basis and the target qubit is an element of $B(\theta, \pm \frac{\pi}{2})$. Encrypted bits cannot be used as control qubits since any ordered (orthonormal) basis may be chosen as a key, including pairs of bases composed of the same elements in reversed order.

**CNOT gates.** For $n \in \mathbb{N}$, the $C^n$ CNOT gate is an $n+1$ qubit gate. When applied to elements of the computational basis of $\mathcal{H}^{\otimes(n+1)}$, if the first $n$ qubits are $\ket{1}$ then a CNOT gate is applied to the last qubit. Otherwise, that element is left unchanged. $C^n$ CNOT gates may be homomorphically applied to systems of qubits when the control qubits are elements of the computational basis and the target qubit is in $B(\theta, \pm \frac{\pi}{2})$.

**The Hadamard gate.** The Hadamard gate is the unitary transformation which takes the elements of the computational basis to the elements of $B(\frac{\pi}{4}, 0)$. When measuring any of the elements of $B(\frac{\pi}{4}, 0)$ in reference to the computational basis, the probabilities of obtaining zero or one are both $\frac{1}{2}$. A Hadamard gate cannot be applied to the encrypted data homomorphically. Nevertheless, by using an ancillary qubit, we construct a quantum gate which takes elements of every orthonormal basis to an equally weighted superposition of the elements of that basis (assuming $\varphi = \pm \frac{\pi}{2}$). This gate, denoted by $D$, is established by first applying an Hadamard gate to an ancillary $\ket{0}$ qubit, and then a CNOT gate to that system of two qubits, where the ancillary qubit is the control qubit and the second is the target qubit. The $D$ gate may be homomorphically applied to the elements of a randomized basis, using an ancillary $\ket{0}$ qubit, resulting in the same effect as when applying a Hadamard gate to the elements of the computational basis – creating a superposition of the elements of that basis with equal probabilities.

**Applications.** We now present several applications that may be achieved using our encryption scheme.

**Coalitions-resilient secure multi-party XOR computation.** Consider the following scenario. Each of $N$ honest-but-curious parties, $\mathcal{P}_i$, $1 \leq i \leq N$, is holding a bit $b_i \in \{0,1\}$. The parties are interested in learning the XOR of their bits, $b_1 \oplus \cdots \oplus b_N$, without revealing their own bits. One trivial solution to that problem is as follows. One of the parties, say $\mathcal{P}_1$, picks $b_0 \in \{0,1\}$ uniformly at random. For $1 \leq i \leq N$: $\mathcal{P}_i$ computes $b_i' := b_{i-1}' \oplus b_i$ and sends the result to the next party. $\mathcal{P}_i$ computes $b_i'' \oplus b_0(= b_1 \oplus \cdots \oplus b_N)$, and sends the result to the other parties. This solution is vulnerable to attacks of coalitions of honest-but-curious parties, trying to gain information regarding the bits of other parties. E.g., $\mathcal{P}_{k-1}$ and $\mathcal{P}_{k+1}$ can learn $\mathcal{P}_k$’s bit by computing $b_{k-1}' \oplus b_k'$. More generally, $\mathcal{P}_m$ and $\mathcal{P}_{m+1}$ can learn the XOR of the bits of the parties $\mathcal{P}_{m+1}, \ldots, \mathcal{P}_{m+t-1}$. One application of our random-basis encryption scheme is the following solution to the multi-party XOR computation problem, which is resilient to such attacks of coalitions.

- $\mathcal{P}_1$ picks $b \in \{0,1\}$ and uses the random-basis encryption scheme to generate an encryption $\ket{\psi_b}$ of $b$.
- For $1 \leq i \leq N$, if $b_i = 1$, then $\mathcal{P}_i$ applies a CNOT gate to the received qubit. $\mathcal{P}_i$ transmits it to $\mathcal{P}_{i+1}$.
- $\mathcal{P}_1$ decrypts the received qubit to obtain an outcome $b'$. Computing $b \oplus b'$, she obtains the desired XOR of the bits of all the parties and sends the result to them.

At each stage, the qubit received by $\mathcal{P}_i$ is an encryption of a random bit. Since our encryption scheme is perfectly secure, measuring that encryption-qubit, $\mathcal{P}_i$ obtains zero and one with equal probabilities, regardless of the actual value of the encrypted bit. Hence, using our perfectly secure random-basis encryption scheme, coalitions of honest-but-curious parties cannot gain any information regarding the bits of the other parties.

**A Quantum Key Distribution (QKD) scheme.** The random-basis encryption scheme requires that the parties hold a shared key. Nevertheless, it may also be used to construct a two-stage QKD scheme, in which one party sends to another information in the form of a string of classical bits. Suppose Alice holds a bit $b \in \{0,1\}$, and wishes to send it privately to Bob. To this end, Alice and Bob may follow the following single-bit two-stage encryption scheme.

- Bob randomly picks $b'$ from $\{0,1\}$, uses (the $\varphi = \pm \frac{\pi}{2}$ version of) the random-basis encryption scheme to generate an encryption $\ket{\psi_{b'}}$ of $b'$, and then transmits $\ket{\psi_{b'}}$ to Alice.
- If $b = 1$ Alice applies a CNOT gate to $\ket{\psi_{b'}}$; otherwise, she leaves it unchanged.
- Alice sends the qubit back to Bob, who decrypts it and obtains a bit, $b''$.
- Bob computes $b'' \oplus b'$ to obtain $b$.

(Obviously, the algorithm may be applied bit-wise to a sequence of bits of any length.)

**A secure Quantum Pseudo-Telepathy scheme.** The phrase Quantum Pseudo-Telepathy was first introduced in [BBT03], and refers to the use of quantum entanglement to eliminate the need for communication in specific multi-party tasks. A comprehensive coverage of the subject may be found in [BBT05]. The simplest example of quantum pseudo-telepathy comes from the Mermin-Peres magic square game [Mer90]. In that
game, two parties, Alice and Bob, are presented with a 3×3 table. Each of them is required to fill in a part of the table, as follows. Alice is given an input $i$, $1 \leq i \leq 3$, and needs to put either 0 or 1 at each entry of the $i$-th row, in such a way that the sum of the three entries will be even. Similarly, Bob is given a $j$, $1 \leq j \leq 3$, and needs to fill in the $j$-th column with the constraint that the sum be odd. Alice and Bob win the game if they place the same number in the intersection of the row and the column that they fill. The parties do not know $i$ and $j$ ahead of the game, and they cannot communicate after being given these values. They are allowed to communicate before the game begins and send information to each other. It was shown in [BBT05] that there is no classical algorithm that lets Alice and Bob win the game with probability greater than $\frac{2}{3}$, whereas there exists a quantum algorithm that lets them win the game with probability 1.

A winning quantum algorithm was introduced in [Mer90]. This algorithm is based on having each of the parties hold two qubits out of an entangled system of four qubits, and its stages are as follows. Before the game begins, the parties generate a system of four entangled qubits and share it in such a way that Alice holds the first two qubits of the system and Bob holds the other two. The game begins, and the parties are given their inputs. Then, each party applies one of several predetermined quantum gates to his/her qubits according to the input. Next, the parties measure their qubits (in reference to the computational basis of $\mathbb{I}$) and fill in the first two entries of their row/column according to the outcomes of their measurements. Each of them fills the last entry of her/his row/column according to the parity condition defined above. It was proved in [Mer90] that, by following this algorithm, Alice and Bob are guaranteed to win the game.

Assume that Alice and Bob are two distant parties, willing to participate in the game. To use the algorithm described above, they must share an entangled four-qubit state. They may ask a third party, Charlie, to generate such an entangled state and transmit two qubits to each of them. In that case, two concerns may arise. First, Charlie might be untrustworthy. Second, two adversaries, Eve and Mallory, might intercept Charlie’s transmission and use the entangled qubits sent by Charlie for a game of their own, or any other purpose. To overcome the possibility that Charlie is untrustworthy, Alice and Bob may decide that one of them, say Alice, will generate the desired four-qubit entangled state and transmit two of the qubits to Bob. This does not solve the second concern. A single adversary, Eve, may intercept the transmission and use the qubits to engage in the magic square game with Alice instead of Bob. Two distant parties may securely generate and share a system of entangled qubits using our random-basis encryption scheme:

- Alice uses our scheme to generate independent encryptions of two 0 bits and two 1 bits.
- Alice generates a pair of ancillary $|0\rangle$ qubits and applies to the system of six qubits a specific quantum gate (we omit the details of the gate from this technical abstract technical report version).
- The first two output qubits are the ancillary qubits, and are not used in the next stages of the scheme.
- Alice keeps the next two output qubits to herself and transmits the last two to Bob.
- Alice and Bob engage in our QKD scheme, during which Alice shares with Bob the keys she used to generate the encrypted qubits in the first stage of this scheme.
- Alice and Bob decrypt the qubits they hold and obtain a system of four entangled qubits.

If an adversary intercepts the transmission and possesses the qubits, then the adversary cannot use them to engage in the game in place of Bob. The last four output qubits constitute a system of four entangled and encrypted qubits, which may be used (after decryption) to win the magic square game.

**Discussion.** In this abstract technical report of the paper we briefly describe a perfectly secure encryption scheme of classical data using quantum computers, based on using randomized bases. Our scheme allows a user to outsource confidential data to a distrustful quantum server and homomorphically apply $\text{NOT}$ gates and controlled-$\text{NOT}$ gates to encrypted data, where the control qubits are plaintexts and the target qubits are encrypted. We suggested a quantum gate which allows applying a modified version of the Hadamard gate to encrypted data. We have suggested ways to use our scheme for performing multi-party computation of binary XOR perfectly secure under an attack of coalitions of honest-but curious parties. We suggested a two-stage QKD scheme. We have suggested a protocol enabling two distant parties securely obtain an entangled pair to be used in a quantum pseudo telepathy game. In general, quantum entanglement is an important resource vastly used in many quantum protocols. Once generated, this resource should be secured in a way that will ensure that only the rightful owners of it will be able to use it. The scheme suggested above enables securing that resource not only for the magic square game, but also for any other purpose. We believe that our new approach and techniques suggest a possible direction for future research on perfectly secure quantum homomorphic encryption.
References


Entrepreneurship Pitch Track

Chair: Ben Gilad
Introduction

Entrepreneurship Pitch Track chaired by Ben Gilad and Shlomi Dolev

Information security practitioners have always had to take a wide-angle view of the world, because there is no aspect of life that is so disconnected from information security that it is immune to problems, or at least the possibility of improvements.

This year’s CSCML Entrepreneurship Pitch Track is a perfect example of this. It is clear that the entrants had a pulse on the current situation in information security, and are gearing up, (in some cases already have geared up) to meet the challenge head-on, and in the process protect all of us.

Entries ranged from cryptocurrency, Nanorobots, IoT, social networks, secret computing, and scheduling. It was heartening, this year as well as in past years, to note that, even in a business focused track, there were entries that could justifiably be considered “for the greater good of the people” – that is, even if they had business motives and priorities, they would still end up benefiting all of us.

These entrepreneurs deserve all the encouragement that we in the community can give them, in whatever form is suitable.

As was the case last years, the Entrepreneurship Pitch Track at CSCML 2019 did an excellent job of fulfilling this objective and consequently was a great success. It received sponsorship from leading VCs (JVP, BaseCamp Innovation Center,) and corporations (Oracle, DELL EMC, IBM). Ten start-ups pitched in the event, out of which “Towards in-vivo Energy Harvesting Computational Nanorobots” was selected by the Entrepreneurship Pitch Track Committee as the leading entry and won the $500 prize donated by-Prof. Bezalel Gavish (ATSMA). “Towards in-vivo Energy Harvesting Computational Nanorobots” was one of the three selected by a committee I chaired. All three pitched in front of the entire audience who voted for the first, second and third places.

“Towards in-vivo Energy Harvesting Computational Nanorobots” presented readiness to turn their technological solution into a business and emerged as the winner! All three leading projects received a certificate of recognition! “Stick ‘n Grip: A revolutionary robot gripper for the e-commerce industry” received second places, and “Algorithm and IoT nitrate sensor to optimize agriculture yields and protect the environment” placed third.

We look forward to an even better CSCML 2020.

Regards, Ben Gilad
Tata Consultancy Services
Entrepreneurship Pitch Track Chairs
Algorithm and IoT nitrate sensor to optimize agriculture yields and protect the environment

Prof. Shlomi Arnon, Prof. Ofer Dahan Mr. Yeshno Elad

And Dr. Kobi Inbar
Background

- Nowadays most of the agriculture world is using fertilizers in their fields and crops.

- Excessive use of fertilizers is expensive and contaminates the groundwater.

- Our solution - standalone device that gives real time Information on Nitrate concentration in the soil in order to reduce the pollution and maximize the yields.
Our Core Technology

- **IoT Electro optics sensor to monitor nitrate 24/7 in the ground** (second generation in final development stage).

- **App to support the measurement remotely** (first generation in medium development stage)

- **Algorithm and big data on a cloud to analyze the measurement and predicate nitrate spatial distribution** (early stage).
Market

- The precision farming market is expected to grow from USD 3.20 Billion in 2015 to USD 7.87 Billion by 2022, at a CAGR of 13.47%.

- The global fertilizer market, with all its components (organic and inorganic), is expected to reach US $ 152 billion by 2020.

- The Specialty Fertilizers market is expected to reach 20 billion dollars in 2020, with an annual growth rate of about 7%.

- The relative share of fertilizers and chemicals out of the total farming expenditure in the United States can reach up to 20-30%.

- This is a significant expenditure and any improvement in utilizing these resources can in turn lead to improved profitability and competitiveness for the farmer.
Product Objectives

- Low power consumption
- Inexpensive
- Accurate measurements
- Small size
- User friendly and easy to use
Modern precision agriculture primarily depends on water and nutrient availability in the soil. While technological means to monitor and control the soil-water conditions are well developed and affordable commercially, monitoring and controlling the nutrient availability in the soil is still far from achieving the required resolution for both optimal plant growth and prevention of water resources pollution.

For the first time, to the best knowledge of the inventors, a real time and inexpensive in-situ continuous measurement system of nitrates was developed.
SocPro
Social Protector for Online Social Networks

Nadav Voloch, Ehud Gudes

Collaborators:
Eyal Pickholz, Omer Sella, Sagiv Mapgavker, Alexander Chinyan, Hagai Ortner
Intro- Social Network data privacy

• During the last years social network privacy issues has been a major concern for users and organizations.

• Cambridge Analytica scandal: data harvesting, personal data trading for political purposes and election interference.
Goal – personal data privacy with free secure connections

• We wish to maintain our personal freedom in the network.
• But we also wish our data will remain secure and will not get to unwanted entities.
• This can be done by a third party software app that identifies potential network hazards, in a user-adapted context.
Social networks today (Facebook, Twitter, Instagram, etc.) use widely common security preferences.

The privacy settings are either:

1. Administrator settings based on reports and other methods.
2. Dependent on user preferences – need user action.
Most of the users are unaware of their privacy settings

<table>
<thead>
<tr>
<th>Privacy Settings and Tools</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Your Activity</strong></td>
<td>$\text{Who can see your future posts?} : \text{Public}$</td>
</tr>
<tr>
<td></td>
<td>$\text{Review all your posts and things you're tagged in}$</td>
</tr>
<tr>
<td></td>
<td>$\text{Limit the audience for posts you've shared with friends of friends or Public?}$</td>
</tr>
<tr>
<td><strong>How People Find and Contact You</strong></td>
<td>$\text{Who can send you friend requests?} : \text{Everyone}$</td>
</tr>
<tr>
<td></td>
<td>$\text{Who can see your friends list?} : \text{Public}$</td>
</tr>
<tr>
<td></td>
<td>$\text{Who can look you up using the email address you provided?} : \text{Everyone}$</td>
</tr>
<tr>
<td></td>
<td>$\text{Who can look you up using the phone number you provided?} : \text{Everyone}$</td>
</tr>
<tr>
<td></td>
<td>$\text{Do you want search engines outside of Facebook to link to your profile?} : \text{Yes}$</td>
</tr>
</tbody>
</table>
SocPro – personal adaptive solution

• Mapping the user’s personal network, and giving Trust scores to his friends and their friends (networks).

• We use efficient graph algorithms for that purpose.
Applicative method – identifying weak links

- Identifying problematic users and connections by their attributes
- Today: based on reports, image and text analysis
- Our method: using deeper analysis of the network
Third party Social Network applications exist today but mostly for marketing and traffic boosters.

No apps for personal user benefits and privacy.
Business model

• Today users do not benefit from the third party use of their personal data.

• The users can buy the app to have a real friend network in which all data is safe.

• The users can decide giving their social data to marketing companies or other interested sides for payment.

• From this exchange the app gets a commission by percentage.
Current status

- Two teams:
  1. Image anonymization team:
      Prototype ready at the end of July 2019.
  2. Algorithmic network modeling team:
      Beta modeling system ready at the end of August 2019.
1. Context identification by Natural Language Processing (NLP).
2. Full adaptive intertwined add-on.
3. Modular software for all platforms (Social Networks).
Cryptopus" - Crypto currency "monitoring application

Nadav Voloch

Collaborators: Maor Hajaj, Hagai Ortner, Tiberiu Rosenberg, Ronen Hayat
Cryptocurrencies

- Most familiar: Bitcoin
- But there are thousands of other, different crypto coins
  - Ethereum, Ripple, Monero, Litecoin, etc./and more…
- Unstable “jumps”; have taken huge losses
- But also rapid, extreme gains
Cryptopus goal is to “catch” huge jumps

- monitor and analyze cryptocurrency trends
- provide real time information and live notification
- recommend investment opportunities in growing market
Smart investment by fishing jumps
Current solutions

• Monitor currencies but involve user choices for notifications while the user usually does not have preliminary knowledge.

• Cryptopus has “fishing” purposes – catching the big jumps.
Technologies used to create the app

- **React Native**: JavaScript-based, multi-platformed: No need to modulate to iOS or Web
- **Python**: an interpreted, high-level, general-purpose programming language.
- **CoinMarketCap API**: World's leading authority, most accurate, up-to-date listing prices, available supply, trade volume, market capitalization
Main Features

• Currency browser.
• Personal dashboard.
• Massive changes notifier.
Main Features

• Currency page: includes value and stats
• Notification choices: daily/weekly/monthly, positive/negative jump limits
• Activity graph
Business model

• “Money machine” – self sustainable software that creates profit.
• Adds from Crypto exchange companies (cex.io, coniexchange.io, etc..)
• Later development: exchange commissions for the app’s trading features
Our team

• Nadav Voloch: Project instructor, Ph.D candidate at BGU Computer Science department.
• Tiberiu Rosenberg, Ronen Hayat: Software developers, last year undergraduates from Ruppin academic college.
• Maor Hajaj: Software developer, last year undergraduates from The center for academic studies.
• Hagai Ortner: Business consultant, years of experience in the tech start-up field.
Current status

- A basic prototype software is currently ready, without GUI.
- First working prototype - at the end of September.
- By the end of December – full Beta version.
Development

- Developing smart machine learning algorithms to predict future crypto movements.
- Auto – investments: no human contact.
- Full dynamic coverage of several crypto markets and other platforms.
Towards Considering Diseases (Cancer), as Chronic Ailments Through Computational Nano-Robots.

Shlomi Dolev, Ram Prasad Narayanan
In 2019, there will be an estimated 1,762,450 new cancer cases diagnosed and 606,880 cancer deaths in the United States.

**Cancer Facts & Figures 2019 – American Cancer Society**

---

**Can Cancer Be Prevented?**

**How Much Progress Has Been Made against Cancer?**

Roughly $87.8 billion was spent in 2014 in the U.S. on cancer-related health care.

The Challenge

THE THERAPY

Current model of drugs – sustained with tested, but limited effectiveness.

Doomed to become obsolete as cells develop drug resistance.

THE NEED

The need, is to have an innovative strategy and idea to provide a generic and agile therapeutic treatment.

End user (Patient), acquires the ability to monitor their ‘ailment,’ become more aware of the cure process.

ADVANTAGES

Targets a single generic characteristic expression.

Can transmit information to Patients.
Existing ‘out-of-the-box’ Ideas (Competition)

NewPhase, SaNP, Gold Coated Nano-Particles

Microsoft computer scientists and researchers are working to ‘solve’ cancer

http://www.innovex.co.il/_Uploads/dbsAttachedFiles/NewPhasePresentationframeiNOVEX2016.pdf
https://news.microsoft.com/stories/computingcancer/
https://www.aummune.com/
Idea - Design of Producible Autonomous Nano-Robots

Blood Energy Harvester

NEM Actuator

Communication Module

Bio-Detection
Computational Nano-Robots as In-Vivo Medicine

TOWARDS CONSIDERING DISEASES (CANCER), AS CHRONIC AILMENTS THROUGH COMPUTATIONAL NANO-ROBOTS.

THANK YOU
AdaptiveClimb
Adaptive Policy for Cache Replacement

Marina Kogan-Sadetsky
Ph.D. Student
Department of Computer Science, BGU
What is Cache

A cache stores data so that future requests for that data can be served faster.

The more requests that can be served from the cache (cache hits), the faster the system performs.
Cache market

- All Operating Systems – Windows, Unix, MAC OS, and so on
- CPU – central processing unite of each computer
- databases
- software systems with slow memory
- web applications
- server-client applications
- and so on...

Better cache replacement algorithm leads to faster system performance.
Cache Management Challenges

- Populate the cache
- Keep the cache content relevant
- Manage cache size

request sequence: \( r_1, r_{34}, r_2, r_{234}, r_1, r_3, \ldots \)

CACHE (fast memory):

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>...</th>
<th>K</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>data</td>
<td>data</td>
<td>data</td>
<td>data</td>
<td>data</td>
<td>...</td>
<td>data</td>
</tr>
</tbody>
</table>

PAIN: to find the right victim to swap out during a page miss with fully occupied frames.

Slow data store
Solution: AdaptiveClimb

Adaptive Policy for Cache Replacement
We developed two easily implementable cash Cache Replacement algorithms – AdaptiveClimb and its dynamic version. Both algorithms outperform well-known Cache Replacement algorithms.
The CLIMB algorithm has a very long stabilization period, resulting in a slow cache. However, during periods of stable distribution, CLIMB has a high hit ratio. The graph shows the hit ratio stabilization over time for CLIMB, LRU, and AdaptiveCLIMB.
LRU algorithm:

- Does not stabilize
  - Slow cache!

But LRU adapts fast to dynamically changing distribution.

Hit ratio stabilization

- CLIMB
- LRU
- Adaptive Climb

Hit ratio vs. time (0 to 100,000)
AdaptiveClimb algorithm:
short stabilization time and high hit ratio → effective cache!
CLIMB algorithm: starts its slow stabilization again $\rightarrow$ slow cache

first round – number of heavy hitters fits initial cache size

second round – initial cache is small
The LRU algorithm does not distinguish between heavy hitters and noise, resulting in a slow cache.

First round – number of heavy hitters fits initial cache size.

Second round – initial cache is small.
Dynamic AdaptiveClimb algorithm: dynamically fits cache size to number of heavy hitters → effective cache!

First round – number of heavy hitters fits initial cache size

Second round – initial cache is small
Thank You
Covert Channel Cyber-attack over video stream DCT payload*
(Copyright protection algorithm for video and audio streams)

Y. Segal and O. Hadar

Communication Systems Engineering Department, Ben Gurion University of the Negev (BGU), Beer-Sheva, 84105, Israel
yoramse@post.bgu.ac.il

Abstract

The two main cyber-attack techniques via video packets are based on using the packet header or the payload. Most of the standard software protection tools easily detect anomalies in headers since there are fewer places to embed the malicious content. Moreover, due to the relatively small header size, such attacks are limited by the data volumes that can transfer. On the other hand, a cyber-attack that uses video packets’ payload can effectively conceal much more information and produce covert channels. Multimedia covert channels provide reasonable bandwidth and long-lasting transmission streams, suitable for planting malicious information and therefore used as an exploit alternative. The primary focus of this article is a proof of concept of cyber-attack that conceals malicious data in a video payload in the compressed domain, using steganography (in real time). This malicious data is extracted using a covert channel and a malware (that had previously planted at the end user side), on the other side. Additionally, after the implementation of the attack, it is necessary to review its parameters and conclude what the optimal parameters to use in different video scenarios.
In this paper, we will demonstrate attacks that take advantage of compressed domain video payload.
It is important to note that this method can be used as a method of copyright protection.

Keywords: Exploit; Invisible Covert Channel; Steganography; Watermarking; Cyber; Steno objects; Intra prediction; Inter prediction; Discrete Cosine Transform; DCT; Motion Vectors.

* This work was supported by the Israel National Cyber Bureau.
1 Introduction and Motivation

New multimedia platforms are introduced to our lives frequently (e.g., Spotify, CellcomTV, Netflix) and the relative part of digital media in internet traffic is increasing. Current studies \cite{1} show that video traffic reached up to 73 percent of consumer internet traffic in 2016 and predicted to reach 82 percent by the year 2021.

H.264, also known as MPEG-4 AVC (Advanced Video Coding) and H.265, are widely used in the new multimedia platforms. H.264/H.265 is suitable for a wide range of applications such as video conferencing, TV, storage, video streaming, surveillance and others. Video steganography over H264 is the process of secretly inserting and concealing data within videos. Steganography has been helpful in protecting media copyrights (via digital watermarks). On end, sophisticated users have used steganography as means of communication, transmitting hidden messages without anyone, but the intended recipient/s, being aware of it. Lately, newspaper reports have indicated that some users are using malicious software to break into smartphones, computers and even internet-connected televisions.

Multiple techniques have been reported for steganography and watermarking. An overview of digital image steganography is presented in \cite{2}. In \cite{3} basic building blocks for steganography in the compressed video were examined: the embedding operation and the choice of embedding alternatives.

It is shown in \cite{4} that Facebook Cover Photos can effectively hide information using Discrete Cosine Transform (DCT) coefficient embedding algorithms \cite{5}. Watermarking solves the challenge of illegal video distribution and manipulation. Watermark’s robustness is critical for avoiding attackers’ watermark disruption \cite{6} \cite{7}. Some methodologies developed in \cite{5} for compressing the robustness of different watermarking techniques. The watermarking algorithm presented in \cite{9} is embedding the watermark into the video by adjusting intermediate frequency coefficients.

An innovative approach for cyber-attack applying a Smart threshold and Anomaly Correction to compressed domain DCT coefficients described in \cite{9}. In this paper, we focus on manipulations of compressed domain Error estimation of DCT coefficients.

Video compression protocols, such as H.264, for example, divide video frames to Macro Blocks (MB), perform pixel predictions, calculate errors between predicted values and original values, perform DCT transform on the error results and then quantization of the obtained DCT coefficients. We conceal malicious data in a video payload using steganography \cite{11} algorithm that operates in the frequency domain and embeds binary codewords into a selected set of DCT coefficients.

The Cyber-attack algorithm takes advantage of lack of sensitivity of movie viewers to small deviations of Macro-Block (object) values from their original ones. Viewers are not likely to notice the minor noise of MBs. Moreover, since the viewer does not know the accurate real value of MBs in the original video movie, they are not likely to notice minor changes that affect MBs values accuracy.
2 Glossary

To complement the needed background, fundamental glossaries are presented in the following section.

**Steganography**: The art of data hiding within data. Steganography is an encryption technique that means to conceal the very existence of a message in oppose to cryptography that means to protect the content of a message.

**Watermarking**: The method of embedding data into digital multimedia content, not necessarily in a hidden manner. This is used to verify the credibility of the content or to recognize the identity of the content's owner. Watermarking has an additional requirement of robustness to possible attacks.

**Covert channel**: Communication paths that were neither designed nor intended to transfer information at all. These channels are typically used by untrustworthy programs to leak information to their owner while performing a service for another program.

**Compression**: The conversion of information to a representation or form that requires fewer bits than the original. This is useful for transmitting across network as well as for storing. Two types of compression: Lossless compression: decompressed image is the same as the original. Lossy compression: decompressed image is not the same as the original but looks similar.

**Intra prediction**: Prediction of a current video data block (e.g. macroblock) is created from previously coded block in the same frame. Exploiting the similarity to neighboring blocks, spatial redundancies.

**Inter prediction**: Prediction of a current video data block (e.g. macroblock) is created from one or more past or future frames (i.e. reference frames). The accuracy of the prediction can usually be improved by compensating for motion between the reference frame(s) and the current frame. Exploiting the temporal redundancies.

**DCT (Discrete Cosine Transform)**: A Fourier related transform which expresses a finite sequence of data points in terms of a sum of cosine functions oscillating at different frequencies. DCT is highly useful for lossy compression most of the signal information tends to be concentrated in a few low-frequency components of the DCT, and small high-frequency components can be discarded. The transform is calculated using a scaling matrix ($S_f$), combined with the quantization process into ($M_f$) where:

$$M_f \approx \frac{S_f \times 2^{15}}{Q_{step}}$$  \hspace{1cm} (1)

**Quantization**: Mapping of a signal with a range of values $X$ to a quantized signal with a reduced range of values $Y$. Using fewer bits to represent the same signal in a lossy manner. Usually preformed after DCT conversion and described as division by a quantization step size, $Q_{step}$, then rounding the result:

$$Q = round\left(\frac{1}{Q_{step}}\right)$$  \hspace{1cm} (2)
Entropy coding: Removes statistical redundancy in the data by representing commonly occurring code words by short binary codes in a lossless manner. Huffman coding is a type of entropy coding that use prefix code for each symbol in an efficient way.

Motion Vectors: A two-dimensional vector used for inter-prediction that provides an offset from the coordinates in the decoded picture to the coordinates in a reference picture.

Median Vector: A two-dimensional vector (2D) that represents the estimated motion vector of a macroblock. It calculated from its neighbors. In video compression, instead of transmitting the macroblock motion vector, we are carrying the difference between the original motion vector and its median vector (predicted motion vector). A 2D motion vector has two components - the X component and the Y component. The median vector calculated per component. All X components from all neighbors motion vectors grouped into one sorted array. The value in the array that separates the higher half from the lower half is determinate as the X component value of the median vector. The Y component will be calculating in the same way.

YUV: YUV is a color encoding system that takes human perception into account in the encoding process, allowing reduced bandwidth for chrominance components and enabling transmission errors and compression artifacts to be more efficiently masked by the human perception than using a RGB representation. The YUV model defines a color space in terms of one luma component (Y) which represents the brightness, and two chrominance components (UV) which represent color. YUV is computed from linear RGB as follows:

\[
Y = 0.299R + 0.587G + 0.114B \tag{3}
\]

\[
U \approx 0.492(B - Y) \tag{4}
\]

\[
V \approx 0.877(R - Y) \tag{5}
\]

DES Encryption: The Data Encryption Standard (DES) uses a block cipher algorithm that takes a fixed length string of plaintext bits and transforms it through a series of complicated operations into another cipher text bit string of the same length. A key is used to customize the transformations, so decryption is only possible with the knowledge of the key used in the encryption process.

Reed-Solomon code: Reed-Solomon codes are group of error correcting codes commonly used in commercial and consumer technologies and communications. The codes operate on a block of data treated as a set of finite field elements called symbols. The codes can detect and correct multiple symbol errors, depending on the number of symbols used in the code for checking: By adding t check symbols to the data, a Reed–Solomon code can detect any combination of up to and including t erroneous symbols, or correct up to and including t/2 symbols.

Data Packet Structure: A Typical data packet contains a header, payload and trailer. A header usually contains instructions about the data carried in the packet (such as length, numbering information and source / destination address). The payload represents the body of the packet and contains the actual data the packet is delivering to the
destination. A trailer usually contains bits marking the end of the data in the payload and error checking bits.

**ARP spoofing**: ARP spoofing, ARP cache poisoning, or ARP poison routing, is a technique by which an attacker sends (spoofed) Address Resolution Protocol (ARP) messages onto a local area network. Generally, the aim is to associate the attacker’s MAC address with the IP address of another host, such as the default gateway, causing any traffic meant for that IP address to be sent to the attacker instead. ARP spoofing may allow an attacker to intercept data frames on a network, modify the traffic, or stop all traffic. Often the attack is used as an opening for other attacks, such as denial of service, man in the middle, or session hijacking attacks. The attack can only be used on networks that use ARP, and requires attacker have direct access to the local network segment to be attacked (Wikipedia).

**MSE (Mean Square Error)**: The mean squared error (MSE) measures the average of the squares of the errors—that is, the average squared difference between the estimated values $\hat{Y}_i$ and the real value $Y_i$. The MSE is a measure of the quality of an estimator—it is always non-negative, and values closer to zero are better. If a vector of n predictions generated from a sample of n data points on all variables, $Y$ is the vector of observed values of the variable being predicted $Y^*$, then the within-sample MSE of the predictor is computed as:

$$MSE = \frac{1}{n} \sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2 \quad (6)$$

**PSNR (Peak signal-to-noise ratio)**: Peak signal-to-noise ratio describes the ratio between the maximum possible power of a signal and the power of corrupting noise that affects the fidelity of its representation and is usually expressed in terms of the logarithmic decibel scale. PSNR is most easily defined via the mean squared error (MSE). If Here, MAXI is the maximum possible pixel value of the image then the PSNR (in dB) is defined as:

$$PSNR = 10 \log_{10} \left( \frac{MAXI^2}{MSE} \right) \quad (7)$$

**SSIM (Structural Similarity Index)**: The structural similarity (SSIM) index is a method for predicting the perceived quality of digital images and videos and is used for measuring the similarity between two images. The measurement is based on a comparison to an unaltered image that is used as a reference. SSIM is designed to improve on traditional methods such as PSNR and MSE. The measure between two windows x and y of common size NxN is (Wikipedia):

$$s(x, y) = \frac{(2\mu_x\mu_y+C_1)+(2\sigma_{xy}+C_2)}{[\mu_x^2+\mu_y^2+C_1][\sigma_x^2+\sigma_y^2+C_2]} \quad (8)$$

With: $\mu_x, \mu_y$ the average and $\sigma_x^2, \sigma_y^2$ the variance, $\sigma_{xy}$ the covariance of x and y. $C_1, C_2$ two variables to stabilize the division with weak denominator (see Wikipedia for more details).
3 Objective

The primary objective of this article is to supply proof of concept to a covert channel that is based on H264 DCT coefficients manipulations. We demonstrate a cyber-attack that conceals malicious data in a video payload, in the compressed domain, using steganography (in real time). It will be used as a remote-control tool of malicious code that already exist at the victim side. The video stream allows as communicate with the malicious code via our covert channel. This new method let us take advantage on the user device without any operating system constrains or firewall restrictions.

After the implementation of the attack, it is necessary to review its parameters and conclude what are the optimal parameters to be used in different scenarios contexts.

The suggested attack will be implemented in the H.264 standard since it is widely used and offer flexibility in the compression process. The H.264 standard defines a syntax for compressed video and a method for decoding this syntax to produce a displayable video sequence. The covert channel that connects the malware and the adversary will be the positions of DCT coefficients in the block, a known dictionary and the malicious data concealed within the coefficients’ values.

The attack was implemented in MATLAB environment with an H.264 open source codec.

To measure the quality of the attack, we performed several tests to ensure high accuracy in detection of the malicious data upon receiving the infected H.264 bitstream. Other quality metrics used are the well-known MSE (mean square error) and PSNR (peak sound to noise ratio) metrics to ensure that the additive infected data do not increase bitrate to a noticeable level.

Since this work deals with concealment of malicious data within DCT coefficients, our primary focus is on the I-macroblocks, in which the residual DCT coefficients are coded and transmitted rather than the motion vectors. P and B-frames consist of I-macroblocks as well as predicted macroblocks, while I-frames merely consist of I-macroblock.

Cyber protection algorithms have two basic models: Detection and Prevention. Detection is an alerting algorithm that typically uses signature analysis or statistical anomaly detection methods. It has the advantage of being attack specific but may not be able to generalize. The generalization gap is overcome by incorporating some automatic adaptation in the detection processor implementing some learning cycles, which might consider an attack as standard data. Prevention is a process that prevents malicious data from penetrating the site or the system. The Prevention process operates on a regular basis regardless of the existence or non-existence of attack, therefore, providing more general protection compared to the Detection process.

In this research work, we are exploring a real-time Prevention algorithm for H.264 video streams. It is part of a more General Prevention Research (GPR) against attacks that use the video or audio stream payload as a malicious data container.

Payload manipulations produce some artifacts that can be described as noise addition to original video stream images. Modern video coding techniques employ lossy coding schemes, which often create compression artifacts that may lead to degradation of perceived video quality. Payload attack takes advantage of naturally introduced
compression artifacts and assumes that the user will not be able to distinguish between compression artifacts and malicious data of covert channel artifacts.

3.1 Attack perspective

To be able to prevent attack via video, it is necessary to analyze and understand the attacker point of view. Video-based Cyber-attacks are divided into two stages: first, the planting of hostile malware which will perform offensive actions such as: taking control of the device, deleting information, denial of service and so on. The second stage is establishing of a hidden communication channel (covert channel), capable of communicating with the malicious software that was preinstalled and sending to it remote operation commands, such as timing the attack and determining the type of attack. In advanced attacks, the covert channel can be used to manage a rolling event, whereas the attack develops according to the victim’s responses. The paper is focused on offensive prevention of the second cyber-attack stage (the covert channel), assuming that the hostile software already exists on the victim system. The first cyber-attack stage is out of the scope of this paper.

Attackers objective is maximizing covert channel bandwidth, thus maximize the amount of malicious data delivered in the stream payload, while minimizing the noise level. There are two types of such video attacks – Online and Offline. Offline attacks based on recorded movies. The attackers have access to or have some movies that they promote. This situation provides attackers with all the time that they need to plant malicious data in the video. Online attacks are much more complicated because of attacks based on intervening between the content streaming server and the user (man-in-the-middle attack) [11]. The online interference needs to guarantee very low latency. Brute-force payload manipulation requires online video transcoding process (decoding and encoding). The transcoding process consumes processing time and increases the latency. Therefore, online attacks will usually be done in the compressed domain and accomplished by manipulating the DCT and the MV components. Unlike the transcoding process, extracting DCT and MV consumes only 10% of the resources that required for full stream transcoding.

Our research to prevent such attacks focused on preventing MV and DCT manipulation. The fundamental concept is based on a random selection of MV and DCT coefficients and performing minor random changes of their values.

The prevention concept is mostly a self-immunization process by which an immune system becomes fortified against some types of malicious data (known as the immunogen). This process described as self-attacking with random parameters such that any attack will be impacted and destroyed by those random changes.

3.2 Infrastructure implementation Method

In general, the attack method is to establish a covert channel between the attacker and a VLC media player malware that located at the end user PC - a VLC that implement the attack code that created with Matlab and converted to C code (see Fig. 1). The VLC had been planted on the victims’ host using social engineering or other known
The second step is to partially decompress a bitstream (which addressed to the malware) and conceal malicious data in a video payload using steganography algorithm that operates in the frequency domain and embeds a binary codeword into a selected set of DCT coefficients. Finally, upon receiving the incoming bitstream, the malware must successfully extract the malicious data while decoding the video.

Fig. 1. Real time man in the middle attack

H.264 provides a clearly defined syntax for representing compressed video and related information (see Fig. 2). Upon receiving a bitstream, a decoder parses the syntax, extracts the parameters and data elements and can then proceed to decode and display video. The syntax is organized hierarchically, from a complete video sequence at the highest level, down to coded macroblocks and blocks.

At the top level, an H.264 sequence consists of a series of ‘packets’ or Network Adaptation Layer Units (NAL). These can include parameter sets containing key parameters that are used by the decoder to correctly decode the video data and slices, which are coded video frames or parts of video frames. At the next level, a slice represents all or part of a coded video frame and consists of several coded macroblocks, each containing compressed data corresponding to a 16x16 block of displayed pixels in a video frame. At the lowest level, a macroblock contains type information describing the choice of methods used to code the macroblock, prediction information such as coded motion vectors or intra prediction mode information and coded residual data. Understanding the syntax is crucial for accessing the desired location while working in real time and analyzing a bitstream.
Fig. 2. H.264 protocol hierarchy syntax and data casting location
3.3 Research Structure and Lab Setup

The research program includes the following components:

1. Defense algorithm
2. Attack algorithm
3. Attack envelope (computers, smartphones, IoT)

Our research to prevent such attacks is focused on preventing MV and DCT manipulation. The fundamental concept is based on random selection of MV and DCT coefficients and performing minor random changes of their values.

The prevention concept is essentially a self-immunization process by which an immune system becomes fortified against some types of malicious data (known as the immunogen). This process can be described as self-attacking with random parameters such that any attack will be impacted and destroyed by those random changes.

As part of this research phase, we focus on attacks, initiated from within the LAN environment, thus performed from inside the organization. In order to evaluate covert channel available bandwidth and corresponding video quality degradation and in order to measure video delivery delays due to the attack, we created an attack envelope that uses ARP spoofing [11] for “hijacking” the user requested live channel video stream and replacing it by the infected one (see Fig. 3 and Fig. 4).

![Fig. 3. Attack lab setup indicating stage A of the ARP spoofing used for hijacking the stream](image)

![Fig. 4. Attack lab setup indicating stage B of the ARP spoofing used for hijacking the stream](image)
Our measurement results indicate that the delay introduced to the requested live video stream, due to stream hijacking, ranges between 10’s to 100’s msec, which is unnoticeable and may be attributed to regular network delays.

Using this method, some of the ARP updates will still arrive from the original real live streaming content server. Therefore, we can expect some temporary disruptions in covert channel transmission (when the user client switches back to the original server). The switch is transparent to the user and only means that covert channel will deliver its malicious content only part of the time when it is the selected streaming server choice.

We use this attack to route the victim’s HTTP request through the attacker who manipulates the data. The HTTP video requests are filtered with IPTABLES (Linux networking tool) and changed to a different destination port (and/or address).

As it can see from Fig. 3 and from Fig. 4, In our lab, we performed the following Proof of Concept measurements, to understand the effect of a MITM (Man-in-The-Middle) on the innocent viewer.

Switch the traffic routing through another computer between the Client and Video Server (Layer 3 re-routing only); Perform of simple proxying with SOCAT (TCP listening and forwarding tool). (Layer 4 proxying only); Repacking the video (from one type of stream to another) via FFmpeg, and full transcoding via FFmpeg.

3.4 Attack Algorithm and Related Work

The attack method in the article is based on [10] [6] [7]. In this section we will point out the similarities and differences between the approaches and how they were managed.

The papers mentioned above discuss embedding a watermark in single JPEG picture, therefore use different approach in embedding the watermark. The transition from handling pictures to handling videos required several adjustments such as:

– Work locally (macroblock by macroblock and frame by frame) rather than globally over an entire picture [6] [7]. This was done to comply with the MATLAB based H.264 codec.
– The basic unit on which the algorithm operates is 4x4 macroblock rather than 8x8 as in [9] to comply with the MATLAB based H.264 codec.
– The algorithm mentioned in [6] [7] embedded the watermark to the 1000 largest DCT coefficients (DC excluded). Since our algorithm operates macroblock by macroblock we decided to select DCT coefficients in the medium range frequencies as the algorithm mentioned in [10].
– By embedding the watermark to the medium frequencies, we achieve: robustness to noisy channel – if a noise is added to the video it will not affect the correlation results. Robustness to operations such as high and low pass filter.

To keep a robust attack while adjusting a picture algorithm to videos we have made few assumptions:
A great advantage for this attack is the lack of need of reference video to determine if a watermark is present in a video or to determine which watermark was embedded [6].

There is a symmetric secret key that must share between the adversary and the users' decoder. The symmetric secret key contains the location of the DCT coefficients in which we embedded the watermark, the length of a word and many times it duplicated.

To extract and detect the watermark the decoder must have a copy of the dictionary [7]. The dictionary must be known to both sides but not necessary be secretive.

3.5 Implementation

The malicious data is concealed in the bitstream by embedding a single codeword to luma macroblocks. Although the term 'watermark' is usually used to describe a verification or an authenticity measure, in this subsection, we will refer to the malicious codeword to concealed in a macroblock as a watermark for simplicity.

The DCT watermarking technique in this project has two main characteristics: it operates in the transform domain instead of the spatial domain, and it can extract the watermark from a frame without comparison to an original unmarked image. The implemented technique is suitable for luminance samples of a source frame, or more precisely, for the luminance residual samples.

The H.264 codec used in the MATLAB environment operates on different macroblocks sizes according to the standard [12] [13] [14]. However, the residual macroblock which goes through the encoding process has a size of 4x4 (See Table 1).

<table>
<thead>
<tr>
<th>DC</th>
<th>2</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>5</td>
<td>8</td>
<td>13</td>
</tr>
<tr>
<td>4</td>
<td>9</td>
<td>12</td>
<td>14</td>
</tr>
<tr>
<td>10</td>
<td>11</td>
<td>15</td>
<td>16</td>
</tr>
</tbody>
</table>

Table 1. 4x4 macroblock DCT coefficients

The watermark consists of a codeword sequence, which superimposed to some of the coefficients of the full-frame DCT transform. The mark always superimposed to the same set of coefficients in each block. The set of coefficients, which the watermark superimposed to can be in every frequency; however, due to the tradeoff between perceptual invisibility and robustness to compression and other conventional image processing techniques, it is essential to choose the coefficients carefully.

To regain some robustness properly choosing the set of DCT values the mark is superimposed to, and by perceptually hiding it in image areas characterized by high luminance variance is essential. For that reason, the set of coefficients belongs to the medium range frequencies.

There is a tradeoff between the size of the dictionary too and the length of a codeword versus the perceptual quality of a video, long codeword cause greater degradation and use higher bitrate to transmit a single symbol. A part of the research was to examine and construct a codebook with unique codewords. Each codeword is composed of M
integers, and in our case since the code is binary, $M$ bits. There are many ways to construct such dictionary, yet after extensive trial and error period the optimal dictionary included 16 codewords, each has 8 bits length. 14 of the codewords used as symbols and two more are signaling words for beginning or ending a transmission between the adversary and the malware.

The dictionary that used in this work is presented in Table 2, the first and last codewords used as signaling symbols:

<table>
<thead>
<tr>
<th>Table 2. Dictionary</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. 0 0 0 0 1 1 1 1</td>
</tr>
<tr>
<td>2. 0 0 1 1 0 0 1 1</td>
</tr>
<tr>
<td>3. 0 0 1 1 1 1 0 0</td>
</tr>
<tr>
<td>4. 0 1 0 1 0 1 0 1</td>
</tr>
<tr>
<td>5. 0 1 0 1 1 0 1 0</td>
</tr>
<tr>
<td>6. 0 1 1 0 0 1 1 0</td>
</tr>
<tr>
<td>7. 0 1 1 0 1 0 0 1</td>
</tr>
<tr>
<td>8. 1 1 1 1 0 0 0 0</td>
</tr>
<tr>
<td>9. 1 1 0 0 1 1 0 0</td>
</tr>
<tr>
<td>10. 1 1 0 0 0 0 1 1</td>
</tr>
<tr>
<td>11. 1 0 1 0 1 0 1 0</td>
</tr>
<tr>
<td>12. 1 0 1 0 0 1 0 1</td>
</tr>
<tr>
<td>13. 1 0 0 1 1 0 0 1</td>
</tr>
<tr>
<td>14. 1 0 0 1 0 1 1 0</td>
</tr>
<tr>
<td>15. 0 0 0 0 0 0 0 0</td>
</tr>
<tr>
<td>16. 1 1 1 1 1 1 1 1</td>
</tr>
</tbody>
</table>

The primary objective in creating the dictionary was to achieve a highly successful detection rate by the malware. Although a dictionary that is formed by 8 bits codeword can consist of 256 unique codewords, the difference (bitwise) between any two codewords is a single bit. By creating greater Hamming distance (the number of bit differences between two codewords) between any two codewords, the detection, which uses correlation method, produce a significant difference in the correlation result when extracting the suspected codeword and correlating it with all the words in the dictionary. Another means to regain robustness and overcome false detections was to use redundant bits by duplicating a single codeword multiple times and referring to it as a single codeword. Denote the rule in which the adversary embeds the watermark as an embedding rule, and it goes as follows:

Let $I$ be the luma Macro Block (MB) on which the embedding process will perform, $T$ to be the coefficient set in the original bitstream (after integer transform & quantization) and $X$ to be the codeword to embed in the MB.

$$X = \{x_i\} \text{, } i \in \text{coefficient set}$$

$$T = \{t_i\} \text{, } i \in \text{coefficient set}$$

The data $T$ is added by modifying the selected DCT coefficients according to one of the following formulas:

$$t'_i = t_i + x_i$$

$$t'_i = t_i (1 + x_i)$$

$$t'_i = t_i + |t_i|x_i$$
Let $T'$ be the selected manipulated coefficient set to reinsert to the MB, and $Y$ to be the codewords dictionary table.

$$z = \frac{YT^*}{M} = \frac{1}{M} \sum_{i \text{Coefficient set}} Y_i T_i^*$$

(14)

The above formula will allow us to determine which codeword was embedded. The $Y$ codeword that yields the most significant $z$ correlation is assumed to be the present in the MB. To find the exact location of the coefficients set we want to mark in a bitstream, a partial decompression is required. The bitstream is composed from NAL (Network Abstraction Layer) unit in the highest syntax layer which is composed of Slices and other control parameter units for the decoder. One layer under is the Slice layer in which a coded unit (frame) is made up of one or more slices. Each slice consists of a Slice Header and Slice Data. The Slice Data is a series of coded macroblocks and skip macroblock that contains no data. In the lowest layer, the Macroblock layer, each coded macroblock contains the following parameters:

- MB type: I, B or P type.
- Prediction information: prediction mode for I MB or reference frames for B/P MB.
- Coded Block Pattern (CBP): indicates which luma and chroma blocks contain non-zero residual elements.
- Quantization parameter: quantization step size.
- Residual data.

Denote the rule in which the malware identifies and extracts the codeword as extracting rule, and it goes as follows: Let $I^*$ be the infected luma MB from which we need to extract the codeword and let $T^*$ be the corrupted coefficient set. To identify the correct codeword, we will measure the correlation, between the corrupted coefficient set and every possible codeword, $Y$, from the dictionary (codebook).

4 Results

The video database was formed by movies with different characteristic such as: magnitude of motion between frames, amount of details or changes in a frame in the spatial domain (detailed frames tend to hold in the medium and high frequencies range more data than frames with greater smooth areas), natural and artificial objects in the video, etc.

All videos are of the same frame size of 144x176 pixels per frame (i.e., QCIF) and frame rate of 30 fps. The videos were initially loaded to the codec in raw (YUV) representation and then encoded to H.264 format to match the MATLAB based codec. The GOP (group of pictures) size set to 4 without B frames.

The codeword that embedded in the videos is of index 2 in the dictionary (i.e., 0 0 1 1 0 1 1 1). The same word was embedded 742 times in each video to examine the probability of detection and false alarm when given a fixed code word within a wide range of video scenes. Fig. 5 demonstrates the mean correlation over all the word
received versus all the words in the dictionary, while each plotline is a different video. The peak of all lines is in the codeword of index two that yields the highest mean correlation. The codeword of index 9 has the lowest correlation to codeword since it is the one’s complement to the embedded codeword. All other codewords have mean correlation values around zero.

![Figure 5: Mean correlation values vs. dictionary codewords](image)

**Fig. 5.** Mean correlation values vs. dictionary codewords

Fig. 6 demonstrates the hit (successful extraction and identification of the embedded codeword) and miss (false detection of the embedded codeword) rate for different videos. The mean successful detection is 92%. That result improved by adding residual data, but it would cause higher MSE (if more DCT coefficients will change per macroblock) or creating a smaller dictionary with fewer symbols that would make us use more symbols to embed the same data in a video. The lowest successful detection is 88%, and the highest successful detection is 96%. This result means that even the lowest successful detection it is quite an excellent method to conceal and detect embedded data.

![Figure 6: Successful detection rate for several videos](image)

**Fig. 6.** Successful detection rate for several videos
To avoid false detection, in addition to maximal computing correlation of the suspected manipulated DCT coefficients with the dictionary words it is necessary to establish a correlation threshold. If all correlation results are less than the threshold, we will assume no codeword sent.

To establish this threshold, we looked for the maximum correlation result with the minimum value over all codewords. Every codeword embedded with uniform distribution and a total of over 12000 codewords concealed in a variety of videos. The threshold that calculated was 0.0606.

Threshold calculation method:

1. Compute correlation for every suspected coefficient set with all words in the dictionary.
2. Find maximal correlation.
3. If the maximal correlation points the codeword that was embedded (successful identification) save its value.
4. Among all correlation values that saved find the minimal and this is the threshold.

For the watermark casting simulation, we used formula 11 that produced the highest successful identification rate as well as the lowest MSE. In the following graph - Fig. 7, we can see the effect of the MSE differences between the embedded frames (that contain malicious data) and clear frames with no additional information. As we can see the effect of the embedded data is to increase MSE (~350 or PSNR 22db) in videos which includes minor motions and lower MSE (~65 or PSNR 30db) in videos with enough motion. It happens since as the motion in a video increase, the residual macroblocks contain more information and more DCT coefficients have non-zero values. So, when adding to the coefficients the binary word, it would effect on the MSE only half of the time (only when we add 1 and no effect by adding 0).

![Fig. 7. MSE vs. frame index per video](image-url)
At the initial design of the algorithm the manner in which the attack operates relied mostly on the methods described in [9] [6] [7]. After the changes and additional code to embed and extract malicious codewords to the codec, the success ratio and correlation results did not fully comply with the results in the papers above. The results of these experiments are presented in Fig. 8. The extraction of a suspected codeword from an incoming bitstream first step is to produce a corrupted set of DCT coefficients with an added codeword with the same size. The second step is to calculate the correlation of the corrupted coefficients with the codewords in the dictionary. With the use of these results, we created a false-positive and true-positive graph that describe number of hits versus the correlation value (see Fig. 8). The cut point between the two graphs is our selected threshold. The first version of the dictionary was composed of binary codewords with 8 bits, and the total size of the dictionary was 256. That means the minimal Hamming distance between the words was 1 bit. However, with the use of this method, the success rate was around 62%.

Changes in the implementation were needed to adjust the gap between the papers and the work with H.264 codec. Therefore, we made changes at the dictionary and extended the Hamming distance in the expense of dictionary size that led to lower bitrate but increased the mean successful detection rate to 92%.

5 Comparative analysis DCT steganography techniques

In paper [16] various video steganography techniques in compressed domain are discussed. Researchers used videos for steganography, which are compressed in MPEG-2, MPEG-4 or H.264/AVC format, although H.265 format is also available but still not utilized for steganography. In video compressed domain, the commonly used methods for steganography are categorized according to the literature and for embedding secret data researchers utilized motion vector, macroblocks, variable length code etc. based
techniques. These video steganography techniques are discussed by highlighting various quality parameters.

Yang et al. [17] proposed an algorithm to hide data in videos using 4 x 4 DCT coefficients. They used vector quantization for hiding 1 bit of secret data in each 4 x 4 DCT block and the hiding was done in the low frequency components of the subblocks. After hiding data, the stego video was compressed using different quantization parameters using H.264/AVC coding standard. Experimental results showed that this technique was highly robust against compression. Shao-dao et al. [18] proposed an approach for video steganography based on high bitrate hiding algorithm to hide video as a secret message. They embedded 1 bit of secret message in each 4x4 macroblocks of DCT using vector quantization. The utilized 8 low frequency coefficients for embedding the information and the extraction was a blind retrieval for this scheme. By analyzing the result, it can be concluded that the scheme was highly robust against compression and PSNR was degraded by only 0.22dB on average and BER at receiver’s end was only 0.015%. But in terms of capacity, this scheme was able to hide only 2 frames of QCIF format in 96 frames of CIF format. Ma et al. [19] presented a technique based on intra-frame distortion drift introduced after embedding in H.264/AVC videos. In this technique the intra-frame distortion was introduced after embedding but not propagated to the neighboring blocks. They deployed the I-frame DCT quantized coefficients to hide data in the 4x4 luminance blocks and there was no intra-frame distortion drift to the covert video. They used block coefficient pairs for embedding with one used for embedding the secret data and the other one was used to fix the level of distortion. The obtained results demonstrated that the embedding capacity of the scheme was high and average PSNR was above 40 db. Esen et al. [20] proposed an adaptive block-based technique by utilizing forbidden zone hiding and selective embedding. The de-synchronization occurred because of adaptive block selection was handled by Repeat Accumulate (RA) codes. For embedding Y component of the frame was utilized and middle-frequency. A comparative summary of the various methods can be seen in Table 3.

**Table 3. Comparative Analysis of Steganography Techniques**

<table>
<thead>
<tr>
<th>Hiding Scheme</th>
<th>Quality Parameters</th>
<th>PSNR</th>
</tr>
</thead>
<tbody>
<tr>
<td>[17] 4 x 4 DCT macroblock coefficients</td>
<td>PSNR, Bit Error Rate</td>
<td>-</td>
</tr>
<tr>
<td>[18] 4 x 4 DCT macroblocks</td>
<td>PSNR, Bit Error Rate</td>
<td>42db</td>
</tr>
<tr>
<td>[19] 4 x 4 DCT block paired coefficients</td>
<td>Capacity, PSNR</td>
<td>40db</td>
</tr>
<tr>
<td>[20] Y components of middle frequency band of DCT</td>
<td>Capacity, PSNR</td>
<td>37db</td>
</tr>
<tr>
<td><strong>Our</strong>  DCT block correlation coefficients</td>
<td>PSNR</td>
<td><strong>30db @ 150Kbps</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>45db @ 80Kbps</strong></td>
</tr>
</tbody>
</table>
6 Conclusion and future research directions

In this paper we suggested a covert channel technic, that is based on video stream. Such method can be used for remote-control cyber-attack without any operating system dependency. The new idea based on manipulating DCT of compressed H.264 standard video streams. The paper offers to prevent such attack by generate random data within the potential DCT. Future work in this area can consider a hybrid technique in which the watermark added in the frequency domain, but spatial information is also exploited by marking only a subset of the image blocks in which there is a lot of changes and details. This hybrid technique can increase the robustness of the watermark, as well as better perceptual quality. Future work in this area can consider a hybrid technique in which the watermark or stego information added in the frequency domain, but spatial information is also exploited by marking only a subset of the image blocks in which there is a lot of changes and details. This hybrid technique can increase the robustness of the watermark, as well as better perceptual quality. Another future work can be to consider CDR (Content Disarm & Reconstruction) techniques that may cope with such watermarks.

7 Acknowledgment

This work was supported by the Israel National Cyber Bureau. The authors gratefully thank Mr. Lior Yahav for implementing the attack algorithm.

References

14. [Online], “H.264 encoder decoder scheme”.
Secret Computing
Privacy-Preserving Machine Learning and Analytics
Inpher has pioneered cryptographic Secret Computing© technology for secure, privacy-preserving analytics and machine learning.

**Leadership**

**Dr. Jordan Brandt**  
CEO & Cofounder  
+ Prev. Cofounder and CEO at Horizontal (acq. by Autodesk)  
+ Prev. Consulting Prof., Stanford University  
+ Ph.D. – Harvard University

**Dr. Dimitar Jetchev**  
CTO & Cofounder  
+ Prof. Math & Cryptography, EPFL Switzerland  
+ Prev. Microsoft Research  
+ Ph.D. – University of California, Berkeley

**Dr. Nicolas Gamma**  
Chief Computer Scientist  
+ Prof. Cryptography & CS  
Université de Versailles  
+ Ph.D. Ecole Normale Supérieure  
+ Lead author TFHE Lib.

**Alexander Petrie, CISSP**  
VP Solutions  
+ Prev. Security Engineer and Technical Manager at Open Systems AG  
+ M.Sc. – EPFL

**Dr. Mariya Georgieva**  
Director of Security Innovation  
+ Associate Researcher at EPFL  
+ Prev. Cryptography Research Engineer at Gemalto  
+ Ph.D. – Université de Caen

**Kevin McCarthy**  
VP Business Development  
+ Prev. Chief Customer Officer at DemystData  
+ BSBA – Bucknell University

**Team of 20:** 9 PhDs, with 300+ papers published in math, cryptography, and CS

**Investors**

JPMorgan Chase & Co.  
Crosslink Capital  
B/O Bowery Capital  
Alpama Ventures

WSJ: JPMorgan Invests in Startup Tech That Analyzes Encrypted Data
Secure Multiparty Computation (MPC) Challenges

**Challenge:** Multiple parties want to collaborate for data analysis and building predictive models but they are unable to share data for myriad reasons.
• Predicting collisions of satellites

• Satellite trajectories are private

• Satellite operators nonetheless perform conjunction analysis

• Need to evaluate non-linear functions with high numerical precision
Iridium 33 and Kosmos-2251 Satellite Collision

- Collision - 2009
- 11,700 m/s
- 789 km above Syberia
- More than 2000 debris
- ISS special maneuvers
For classifying rare events, numerical accuracy does matter!

Client needs: good predictions

<table>
<thead>
<tr>
<th>TRUE 1</th>
<th>PREDICTED 1</th>
<th>TRUE 0</th>
<th>PREDICTED 0</th>
</tr>
</thead>
<tbody>
<tr>
<td>TP = 0</td>
<td>FN = 1</td>
<td>FP = 0</td>
<td>TN = 9,999</td>
</tr>
</tbody>
</table>

accuracy ≈ 1, recall = 0, F1-score = 0
**Cross-industry Customer Use-Cases**

### Feature aggregation across private datasets

**Objective:** Better serve customers by combining their data across companies without compromising privacy, data security or using a trusted third party.

**Solution:** With XOR, secretly compute the common customers across banking, insurance and telecoms providers to train more accurate models for credit scoring, risk assessment and targeted marketing using the combined feature set.

### Anomaly Detection for Satellite Operators

- Pool data for anomaly detection across operators without sharing
- Predict and avoid interruptions in communications and operations
- Four manufacturer/operators involved with ESA lead

### Privacy-Compliant Machine Learning at ING

- Improve customer marketing models by leveraging training data with PI across multiple restricted jurisdictions without anonymizing any features.

**Logistic Regression for Customer Prediction**

- 24,000 data points
- 8% of customers who buy the lending product
  - Original Model = 30%
  - Secret Computed Model = 50%
- 40% increase in data quality because all features were preserved with XOR.

- Meet/exceed compliance under GDPR and Swiss DPA data transfer rules.
- Explore zero-knowledge multicloud computing.

### Secure Predictive-Maintenance for Distributed Fleets

- Privately aggregate data across fleets to improve maintenance models.
- Improve uptime while reducing operational costs.
- Deliver value to fleet manager without compromising supplier data.
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Identify outliers</strong></td>
<td>PRECISION</td>
<td>+</td>
<td>-</td>
<td>+</td>
</tr>
<tr>
<td><strong>Arithmetic accuracy</strong></td>
<td>CORRECTNESS</td>
<td>-</td>
<td>+</td>
<td>-</td>
</tr>
<tr>
<td><strong>Retain data fields</strong></td>
<td>FEATURES</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td><strong>No re-identification</strong></td>
<td>PRIVACY</td>
<td>+</td>
<td>+</td>
<td>-</td>
</tr>
</tbody>
</table>

1) [http://science.sciencemag.org/content/347/6221/536.full](http://science.sciencemag.org/content/347/6221/536.full)
2) This widely cited paper demonstrated that in a clinical setting, tuning DP to an adequate level of privacy yielded grossly inaccurate outputs which would have led to fatal dosage recommendations for patients.
   [https://www.ncbi.nlm.nih.gov/pmc/articles/PMC402719](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC402719)
3) [https://blog.cryptographyengineering.com/2015/06/15/what-is-differential-privacy/](https://blog.cryptographyengineering.com/2015/06/15/what-is-differential-privacy/)
4) For mathematical description of the protocol with benchmarks and F-Score results on logistic regression models, see Inpher's Financial Cryptography 2018 paper [https://eprint.iacr.org/2017/1214.pdf](https://eprint.iacr.org/2017/1214.pdf)

*Underlined text indicates the feature that is being compared.*
Computation Budget Allocation
Secure Multiparty Computation
Customer-hosted Analyst Platform (cloud or on-prem)  

Inpher-hosted XOR Service (never exposed to data)

- frontend (API, UI)
- compiler
- trusted dealer

engine

Data Source 1
Data Source 2
Data Source n...

Analyst submits operations to XOR Service and selects data sources
Customer-hosted Analyst Platform (cloud or on-prem)

Inpher-hosted XOR Service (never exposed to data)

Data Source 1

Data Source 2

Data Source n...

Operations compiled into a ‘circuit’ and distributed as a binary
Customer-hosted Analyst Platform (cloud or on-prem)

Inpher-hosted XOR Service (never exposed to data)

Data Source 1

Data Source 2

Data Source n...

Offline Phase: Random triplets are generated and distributed
Customer-hosted Analyst Platform (cloud or on-prem)

Inpher-hosted XOR Service (never exposed to data)

Online Phase: Data sources secretly compute with random numbers
Customer-hosted Analyst Platform
(cloud or on-prem)

Inpher-hosted XOR Service
(never exposed to data)

Partial results sent to Analyst Platform to construct final output
Secret Shared Data
Reveal Secret Shared Data
### Logistic Regression Training dataset with singularities

<table>
<thead>
<tr>
<th>Dimensions</th>
<th>Time Plaintext</th>
<th>Time Offline</th>
<th>Memory Offline</th>
<th>Time Online</th>
<th>Memory Online</th>
<th>Abs Precision</th>
</tr>
</thead>
<tbody>
<tr>
<td>100k x 31</td>
<td>5 s</td>
<td>1 mins 38 s</td>
<td>0.4 GB</td>
<td>36 s</td>
<td>0.6 GB</td>
<td>$7.6 \times 10^{-8}$</td>
</tr>
<tr>
<td>100k x 72</td>
<td>1 mins 15 s</td>
<td>4 mins 58 s</td>
<td>0.9 GB</td>
<td>1 mins 40 s</td>
<td>1.1 GB</td>
<td>$2.8 \times 10^{-7}$</td>
</tr>
<tr>
<td>1 Million x 50</td>
<td>1 mins 5 s</td>
<td>36 mins 40 s</td>
<td>6.87 GB</td>
<td>18 mins 30 s</td>
<td>10.1 GB</td>
<td>$1.7 \times 10^{-8}$</td>
</tr>
</tbody>
</table>
Thank you!

Join Inpher! 😊
https://inpher.io
Stick n’ Grip

A revolutionary grasping solution for the e-commerce industry
The team

“The why are we the ones to change the robotic manipulation field?”

Noam Hasson
Mechanical designer and algorithm planner.

Prof. Amir Shapiro
Board Advisor, Associate Professor and director of the robotics laboratory at Ben-Gurion University.

Prof. Elon Rimon
Board Advisor, Associate Professor at the Technion. Author of a robot grasping book

Yoav Golan
CTO, Mechanical Engineering Student-Fourth Year
Our Vision

We aim to make the best robot hand for the e-commerce industry
Picking robots have not been invented yet.

Today, picking is done by hand.
More problems..

Human workers suffer from variety of disadvantages

**Discontinuous work**
Human workers demand brakes and works in shifts

**Physical limitations**
Human cannot reach high objects,

**Quality control**
Humans work quality and efficiency can be influenced by variety of external causes
We design picking tools
We use innovative mechanical concepts to allow robotic arms to easily pick objects from a shelf.
how it works?

- Robotic arm
- Packages on a shelf
- Sticker Gripper
- Stickers dispenser
Robot get near a desired shelf
Robot gets near a desired shelf
Picking a sticker
2 Picking a sticker
Picking a sticker

Adhesive layer
3 Positioning in front of the package
4 Sticker is attached to the package
Package is manipulated
When finished, we let go of the sticker!
When finished, we let go of the sticker!
The robot moves on to the next task
Some Numbers

- The cost of a human picking station is $70,000 per year, not including insurance expenses.
- The cost of a robot picking station is $42,000 per year.

The customer can save more than $28,000 per year for every picking station!
Business model

**Sticker sales**
- Production cost of a sticker: 0.034$
- We will sell each sticker at cost of 0.045$

**Gripper sales**
- We will sell the grippers with minimum profit
- Encourage companies to use our product
0.011$
Profit per sticker

3,100,000,000$
Packages a year*

34,100,000$
Income per year

*Estimated delivery count of Amazon at 2017
Thanks!

Any questions?

You can find us at
- ashapiro@bgu.ac.il
- rimon@technion.ac.il
- noamhason@gmail.com
- yoavgo@post.bgu.ac.il