Discovery of Robust Protocols for Secure Quantum Cryptography

Walter O. Krawec
University of Connecticut
Storrs, CT USA
walter.krawec@uconn.edu

Sam A. Markelon
University of Connecticut
Storrs, CT USA

ABSTRACT

Quantum Key Distribution (QKD) allows for two parties to establish a shared secret key which is secure against an all-powerful adversary (a task impossible to achieve using only classical communication). Furthermore, any attack against these systems creates an observable "noise signature." This work develops a new solution representation for QKD protocols allowing a GA to evolve optimal protocols to counter observed attacks against the communication.

CCS CONCEPTS

• **Security and privacy** → **Cryptography**; • **Computing methodologies** → **Search methodologies**;

KEYWORDS

Genetic Algorithm, Quantum Computing, Quantum Cryptography

ACM Reference Format:
Walter O. Krawec and Sam A. Markelon. 2019. Discovery of Robust Protocols for Secure Quantum Cryptography. In *Genetic and Evolutionary Computation Conference Companion (GECCO '19 Companion), July 13–17, 2019, Prague, Czech Republic.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3319619.3321945

Introduction: Quantum Key Distribution (QKD) protocols allow two parties, Alice (A) and Bob (B), to establish a shared secret key secure against an all-powerful adversary. This is in strict contrast to classical key distribution where security always requires computationally bounded adversaries. Furthermore, QKD technology is a practical, real-world technology today [10].

Quantum communication, used in QKD, has the fascinating property that any attempt to learn information on the secret key causes a detectable disturbance. In fact, one may upper-bound an adversary's information based only on observable noise. Normally QKD protocols are designed and then analyzed to see how much noise they can tolerate. In [5, 6], the question was "flipped" to ask: given a particular amount of noise, can we design optimal protocols to counter this attack, taking into account practical limitations on user capabilities. Indeed, in those works, QKD protocols were produced, through evolutionary algorithms, which were superior to state-of-the-art QKD protocols in terms of secure communication efficiency. In fact, those systems found protocols which could successfully operate over channels (i.e., attacks), where ordinary QKD protocols would simply fail.

Evolutionary algorithms have seen great success in quantum algorithm design [11]; they have also been successfully applied to problems in classical cryptography [8, 9]. Only recently have they been applied to problems in *quantum* cryptography for both analyzing human-made protocols and designing optimal protocols [4–6]. In this work, we extend previous efforts to develop optimal protocols by devising an entirely new solution representation. Our new method is not as restrictive as the "template" based approach used in [5]; it is also easier to simulate (necessary for fitness calculations) than a circuit based approach used in [6]; finally, it is also a representation that makes practical sense with today's technology. We then use a GA to optimize protocols for certain channels and discover it can actually discover innovative strategies to improve key-rates above that for which standard QKD protocols support.

Solution Representation: We consider a QKD protocol to be a list of *encoders* and *decoders*. Encoders are responsible for choosing a quantum state to send based on A's choice of key-bit. This can be a probabilistic process and this probability choice is part of the encoder. These values are all later evolved by the GA (the state choice, and the probability of executing). Decoders, which are controlled by B, receive a quantum state, perform a *measurement* on it (note that, for an overview of basic quantum communication background, the reader is referred to [7]), and, based on this result, output a guess as to A's key-bit. The decoder may also report "inconclusive." Finally, both encoders and decoders pass a classical message to one-another. If this message matches (i.e., both send the same code-word), then both parties accept this iteration and use it to contribute towards their raw-key. Otherwise, both parties discard all results. Either way, this process is repeated until a sufficiently large *raw-key* is established of size N bits. From this, standard error-correction and privacy amplification are run [10] to distill an actual secret key of size

GECCO '19 Companion, July 13–17, 2019, Prague, Czech Republic
© 2019 Association for Computing Machinery.
ACM ISBN 978-1-4503-6748-6/19/07. . . $15.00
https://doi.org/10.1145/3319619.3321945

Channel Description	$(1,2,2)$				$(2,3,3)$				Ref. [5]		Ref. [5] (w/P)		BB84
	Avg.	σ	Max.	#	Avg.	σ	Max.	#	Avg.	σ	Avg.	σ	
10%	.1548	.00051	.1552	50	.1538	.00065	.1552	50	.152	10^{-7}	.155	10^{-5}	.152
13%	.0023	.00581	.017	7	.0022	.00554	.017	8	0	0	.017	10^{-5}	0
$C1$	.1745	.01177	.2152	50	.1675	.03750	.2106	49	.126	.04	.127	.035	0
$C2$	.1531	.02623	.1692	49	.1301	.03367	.1681	49	.157	.018	.157	.028	0

Table 1: Evaluating our algorithm and comparing to prior work. $(1,2,2)$ implies our algorithm assuming no code-words and two encoders/decoders max; $(2,3,3)$ implies code-words are allowed and three encoders/decoders max. $C1$ and $C2$ denote "Channel 1" and "Channel 2" from [5]. "#" denotes the number of successful runs out of 50. Higher numbers are better (as they imply a faster key-rate). "(w/P)" means with preprocessing manually turned on in the prior work of [5].

Channel Description	$(1 \to 2,2,2)$				From [5]
	Avg.	σ	Max.	#	
$C1$	.1869	.01665	.2035	50	.127
$C2$	.1587	.01377	.1685	50	.157

Table 2: Showing improvement when using one code-word initially, then later increasing the search space.

$\ell(N)$. The secret key-rate is computed using the Devetak-Winter equation [2] which states that, for N sufficiently large, the ratio $\ell(N)/N$ approaches $S(A|E) - H(A|B)$, where $S(A|E)$ is the conditional von Neumann entropy and $H(A|B)$ is the conditional Shannon entropy. We wish to maximize this ratio, thus we use this equation as our fitness function later. Note that an algorithm for computing $S(A|E)$, given a specific protocol and observed noise, is described in [5]; computing $H(A|B)$ is trivial given the protocol and observed noise.

A candidate solution is then a list of encoders and decoders. Crossover involves simple one-point crossover on each list of encoder and decoder for the two parent protocols. Mutation will add/remove an encoder or add/remove a decoder (probability 10% each); or it will mutate one of the internal parameters of the given encoders and decoders (e.g., altering the state it prepares) with probability 40%. Our GA uses tournament selection with a tournament size of 5; population size of 100; and a stopping criteria of 150 generations.

We evaluate on *symmetric channels* and *asymmetric channels*. The first gives us a solid benchmark to compare with as it is known that BB84 is optimal without preprocessing [1]. The asymmetric channels were also evaluated in [5, 6] giving good comparison cases. Results are shown in Table 1. We note that our approach outperformed the template based approach in [5] without preprocessing activated, on all tests except for Channel 2. We also note that it automatically discovered preprocessing strategies whereas in prior work, this had to be manually turned on within a template.

We also experimented with incremental evolution [3]. Here, we evolved the system for 100 generations with code-words deactivated; then we increased the search-space size by allowing code-words for an additional 100 iterations. This test then allowed our system to surpass [5] on Channel 2 as shown in Table 2. Note that our system surpassed results from [6] on all tests but this is not a fair comparison as the circuit-based approach in [6] did not allow for classical communication in addition to quantum (instead, their approach consisted of elementary quantum gates creating protocols with a very modular design that may be potentially easier to realize practically than our approach).

Closing Remarks: Many interesting problems remain open. Considering finite-key affects and practical devices are perhaps the most important, though this would require more involved simulations of the evolved protocol to evaluate fitness.

REFERENCES

[1] Joonwoo Bae and Antonio Acín. 2007. Key distillation from quantum channels using two-way communication protocols. *Physical Review A* 75, 1 (2007), 012334.

[2] Igor Devetak and Andreas Winter. 2005. Distillation of secret key and entanglement from quantum states. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Science* 461, 2053 (2005), 207–235.

[3] Alex Satoru Fukunaga and Andrew B Kahng. 1995. *Improving the performance of evolutionary optimization by dynamically scaling the evaluation function.* UCLA Computer Science Department.

[4] Walter O Krawec. 2016. A genetic algorithm to analyze the security of quantum cryptographic protocols. In *Evolutionary Computation (CEC), 2016 IEEE Congress on.* IEEE, 2098–2105.

[5] Walter O Krawec, Michael G Nelson, and Eric P Geiss. 2017. Automatic generation of optimal quantum key distribution protocols. In *Proceedings of the Genetic and Evolutionary Computation Conference.* ACM, 1153–1160.

[6] Walter O Krawec, Stjepan Picek, and Domagoj Jakobovic. 2019. Evolutionary Algorithms for the Design of Quantum Protocols. In *To appear, EvoApplications 2019.*

[7] M.A. Nielsen and I.L. Chuang. 2000. *Quantum Computation and Quantum Information.* Cambridge University Press, Cambridge, MA.

[8] Stjepan Picek and Marin Golub. 2011. On evolutionary computation methods in cryptography. In *MIPRO, 2011 Proc. 34th International Convention.* IEEE, 1496–1501.

[9] Stjepan Picek, Luca Mariot, Alberto Leporati, and Domagoj Jakobovic. 2017. Evolving S-boxes based on cellular automata with genetic programming. In *Proc. GECCO 2017 Companion.* ACM, 251–252.

[10] Valerio Scarani, Helle Bechmann-Pasquinucci, Nicolas J. Cerf, Miloslav Dušek, Norbert Lütkenhaus, and Momtchil Peev. 2009. The security of practical quantum key distribution. *Rev. Mod. Phys.* 81 (Sep 2009), 1301–1350. Issue 3.

[11] L. Spector. 2004. *Automatic Quantum Computer Programming: A Genetic Programming Approach.* Kluwer Academic Publishers, Boston, MA.