

# *Genetic Algorithm to Study Practical Quantum Adversaries*

**Walter O. Krawec**      Sam A. Markelon

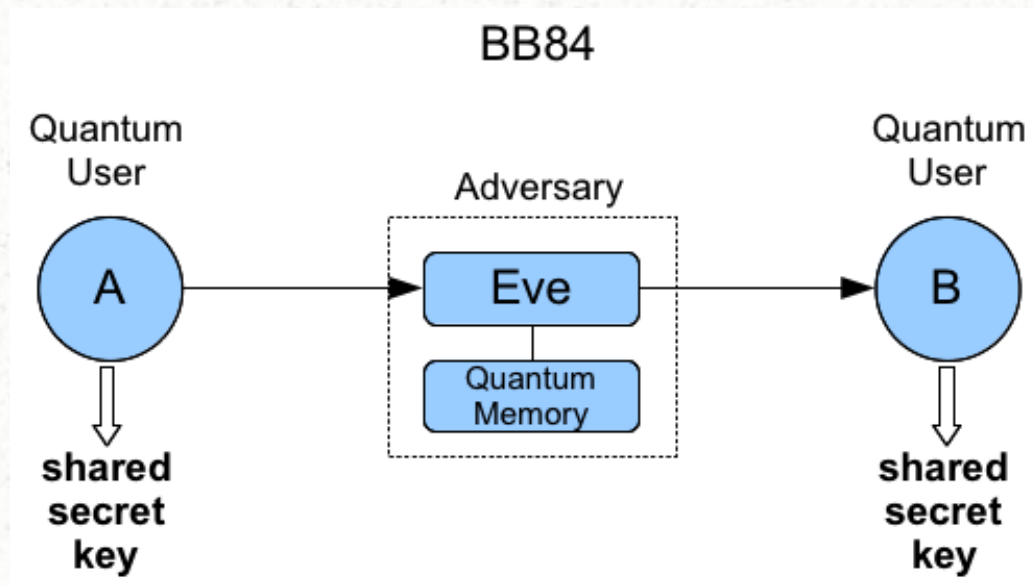
University of Connecticut, Storrs CT USA

walter.krawec@gmail.com  
walterkrawec.org

# *Quantum Key Distribution (QKD)*

- Allows two users – Alice (A) and Bob (B) – to establish a shared secret key
- Secure against an all powerful adversary
  - Does not require any computational assumptions
  - Attacker bounded only by the laws of physics
  - Something that is not possible using classical means only
- Accomplished using a *quantum communication channel*

# Quantum Key Distribution



# *QKD in Practice*

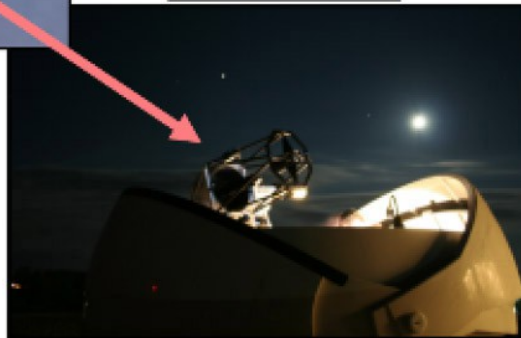
- Quantum Key Distribution is here already
- Several companies produce commercial QKD equipment
  - MagiQ Technologies
  - id Quantique
  - SeQureNet
  - Quintessence Labs
- Have also been used in various applications
- Cities are developing quantum networks
- Freespace QKD is possible...

# QKD in Practice: Freespace

Alice

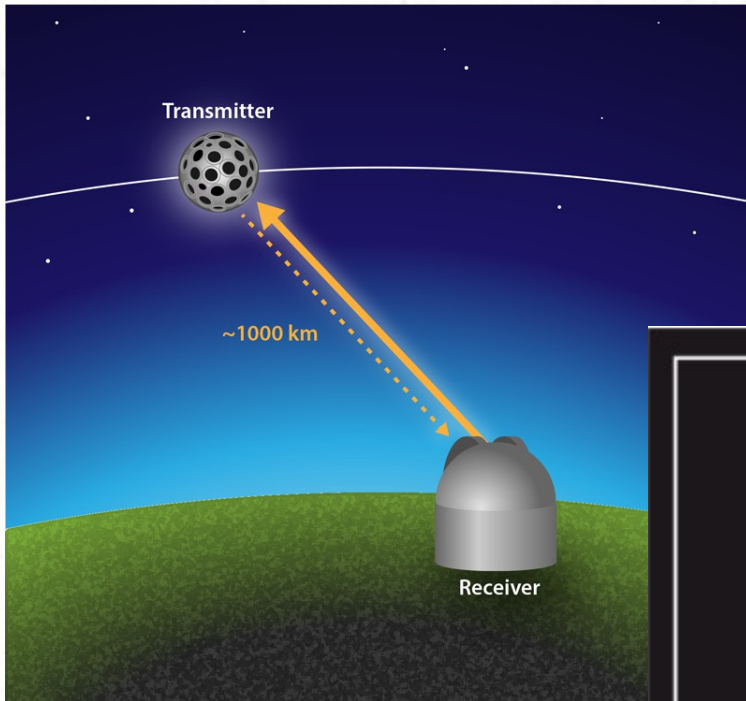


Bob



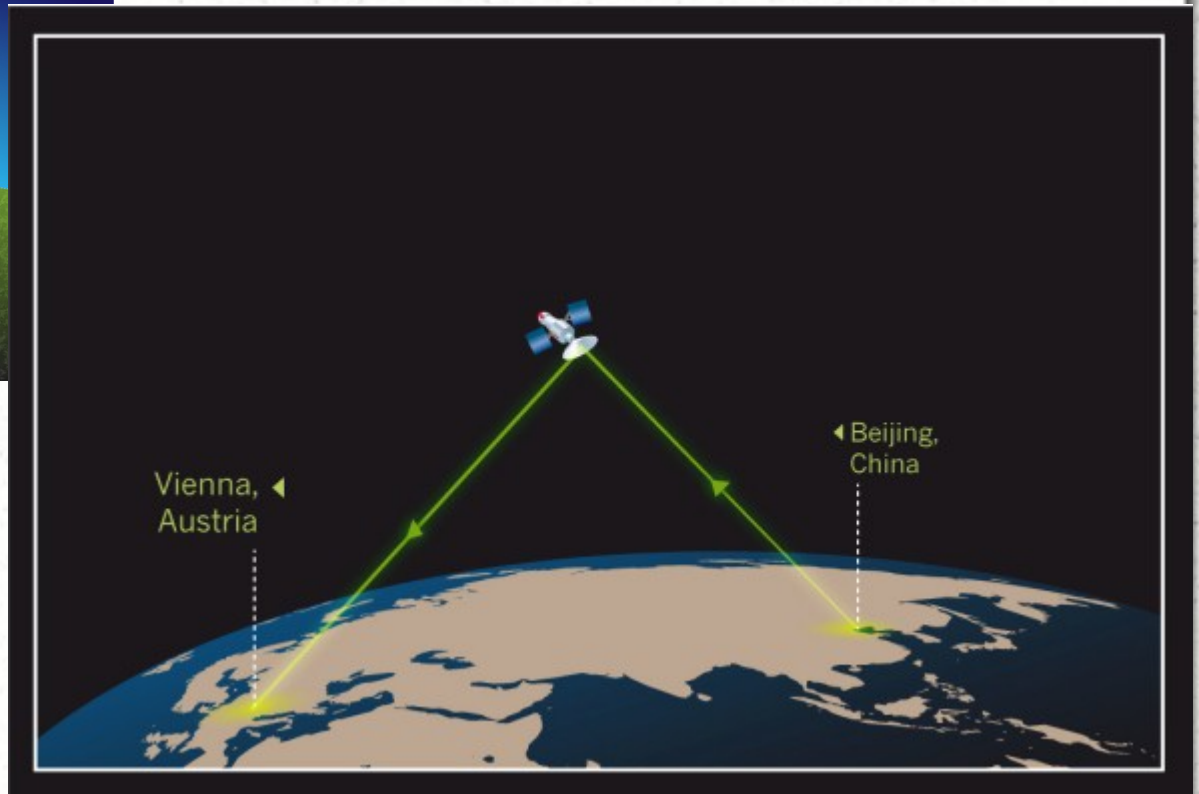
<http://spie.org/newsroom/5189-free-space-laser-5-system-for-secure-air-to-ground-quantum-communications>

# QKD in Practice



<http://www.nature.com/news/data-teleportation-the-quantum-space-race-1.11958>

<https://physics.aps.org/articles/v8/68>



# *Our Work*

- Currently, numerous QKD protocols exist, many with unconditional security proofs
  - Security against “***all-powerful***” adversaries
  - Proofs involve information theoretic arguments to compute the “***key-rate***” as a function of “***noise***”
  - Direct correlation between noise and information gained by an adversary
- Of great interest: a protocol's **noise tolerance**

# *Our Work*

- However, such “unconditional” security proofs assume the adversary has access to complex quantum technology such as:
  - Perfect quantum memories
  - The ability to perform optimal measurements of high-dimensional systems
- Analyzing QKD protocols with “practical” adversaries is an important question
  - But difficult!



# *Our Work*

- Our goal: Design a system (a genetic algorithm) that can take as input an *arbitrary* QKD protocol, and output its noise tolerance for *practical* adversaries
- Different models of “practical” adversaries – here we use a definition from [2]:
  - Adversary does not have access to a quantum memory system

## *Our Work: The Idea*

- We will use a GA to evolve actual **practical** attacks against a given input protocol.
- The GA will attempt to minimize the **induced noise** of the attack, while maximizing the **information gain**
  - This will lead to a bound on the noise-tolerance of the given protocol against practical adversaries
- Practical Benefit: noise tolerances are higher for practical adversaries, thus we may be able to operate these QKD protocols at higher rates!

## *Related Work*

- Evolutionary Algorithms have been used for some time to study quantum algorithms
- They also have seen use in studying **classical** cryptography
- We have used them to study the security of arbitrary QKD protocols against **all-powerful** adversaries
- We also have shown how a GA can be used to discover optimal QKD protocols.

## *Related Work*

- Other automated (non EA) tools exist to analyze QKD protocols in both all-powerful and practical scenarios
- However these other tools all require the QKD protocol to be converted into an **entanglement-based form**
  - Such a conversion requires complex user-knowledge
  - Furthermore, such a conversion is not known to be possible for all classes of QKD protocol!
- We are proposing a system that can take any arbitrary QKD protocol in it's basic form (i.e., not converted to an entanglement-based version) and analyze its maximal noise tolerance for practical adversaries.

# *Main Contributions*

- We show how a gate-based solution representation and a unitary-based representation can be used to study practical quantum adversaries against **arbitrary** QKD protocols
- Our evaluations show that evolutionary methods can produce the same, or similar, noise tolerances as current-known results
- We apply our techniques on protocols which do not admit a known entanglement based version – thus our methods can be applied to a much wider range of QKD protocols than current non-EA approaches are capable of.
- Finally, our approach does not require extensive technical knowledge of the mathematical foundations of quantum computation – thus, our system is potentially more applicable to a wider user base.

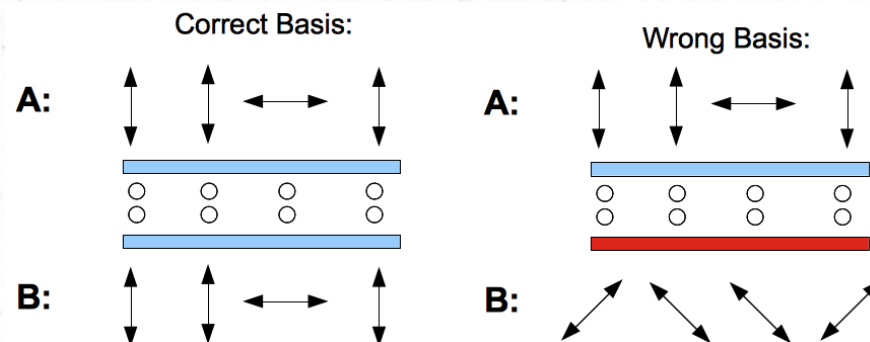
# *Background*

# *Bits vs. Qubits*

- Classical Bits:
  - May be 0 or 1
  - Can be read at any time
  - Can be copied
- Quantum Bits (*qubits*)
  - May be  $|0\rangle$ ,  $|1\rangle$ , or a *superposition* of both
  - Reading a qubit (called measuring) can destroy it and produce random output
  - Cannot copy a qubit
  - Modeled as a vector in  $C^2$

# Preparing and Measuring

- Qubits are modeled as vectors in  $C^2$
- Many ways to send (*prepare*) a qubit
  - May prepare using any orthonormal basis of  $C^2$
- Many ways to read (*measure*) a qubit
  - May read in any orthonormal basis of  $C^2$
- If you prepare and measure in the same basis, result is deterministic
- Otherwise it is random and original qubit “collapses” to the observed state





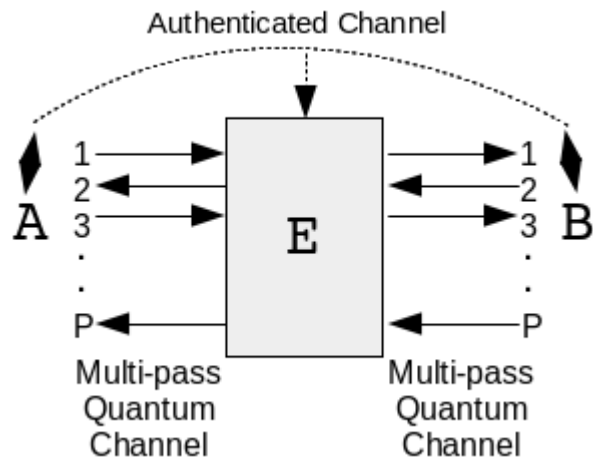
# Quantum Processes

- Two (equivalent) ways of thinking of quantum processes: circuit based and unitary based
- Circuit: A collection of rudimentary **gates** each applied to one or two **wires** (a wire holding one qubit).
- Unitary: A unitary matrix acting on  $C^n$
- We work with both models:
  - Circuit: Advantage is it describes a more practical system
  - Unitary: Advantage is it gives Eve potentially more power (unless the number of gates in the circuit is very large)

# *Quantum Key Distribution*

# QKD – Two Stages

- Quantum Communication Stage
  - Consists of numerous iterations, each leading to at most one key bit
  - Uses a P-pass quantum channel allowing qubits to travel from A to B “P” times
  - Also uses an authenticated classical channel
  - Output: a **raw-key** of size N-bits



# QKD – Two Stages

- Classical Post Processing:
  - Takes as input the N-bit **raw key** and outputs an L(N) bit **secret key**
  - We are interested in the **key-rate function**:

$$r = \lim_{N \rightarrow \infty} \frac{L(N)}{N}$$

# QKD – Two Stages


- Classical Post Processing:
  - Takes as input the N-bit **raw key** and outputs an L(N) bit **secret key**
  - We are interested in the **key-rate function**:

$$r = \lim_{N \rightarrow \infty} \frac{L(N)}{N}$$

- In our practical adversary setting, this is a classical system at the end, thus we may use the Csiszar-Korner bound [4]:

$$r = \lim_{N \rightarrow \infty} \frac{L(N)}{N} = H(A|E) - H(A|B)$$

# Goal

$$r = \lim_{N \rightarrow \infty} \frac{L(N)}{N} = H(A|E) - H(A|B)$$


Typically, as the noise increases, Eve's uncertainty drops causing  $r$  to decrease.

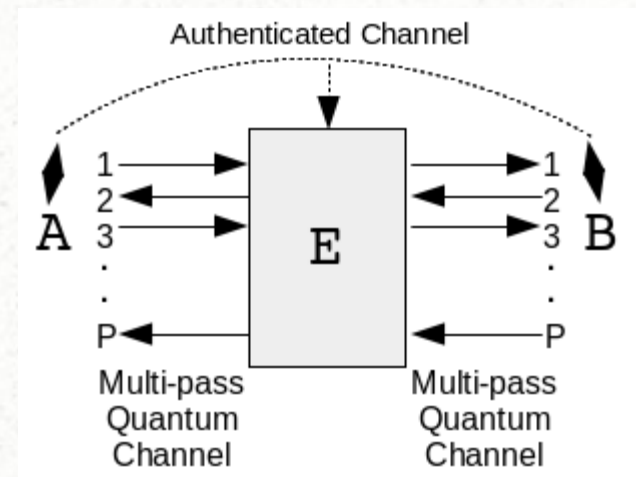
Question: When does  $r=0$ ?

Goal: find an attack which causes  $r$  to drop to zero while inducing a minimal level of noise. Thus, in practice, whenever this amount of noise is observed, one should **abort!**

# *The Algorithm*

# *Solution Representation*

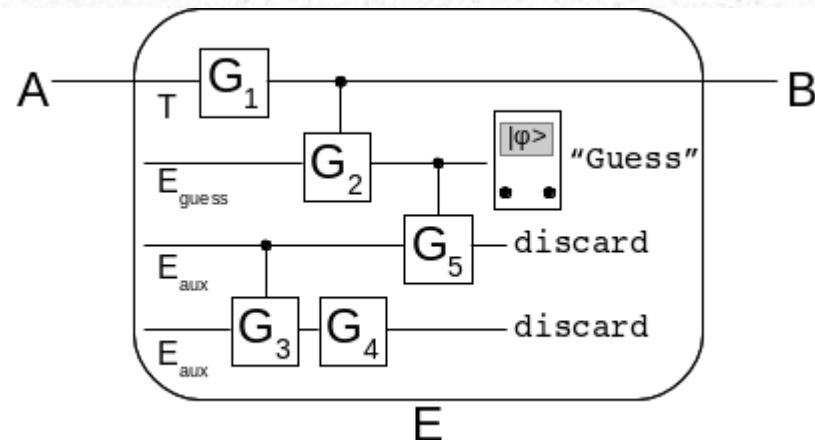
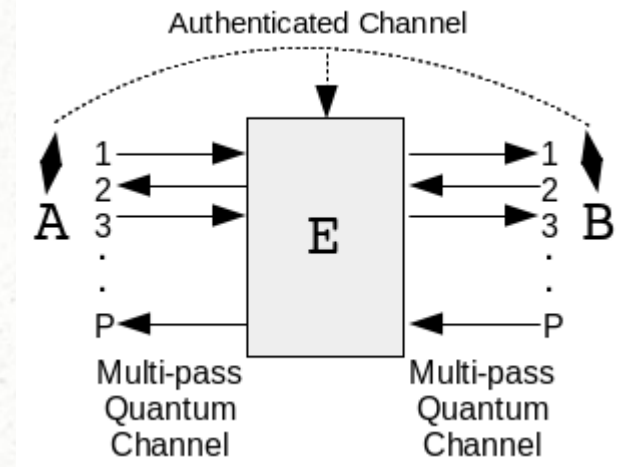
- For an arbitrary QKD protocol, we must evolve an attack consisting of  $P$  “probes” and a final **measurement strategy** yielding a **guess** of the key-bit being sent





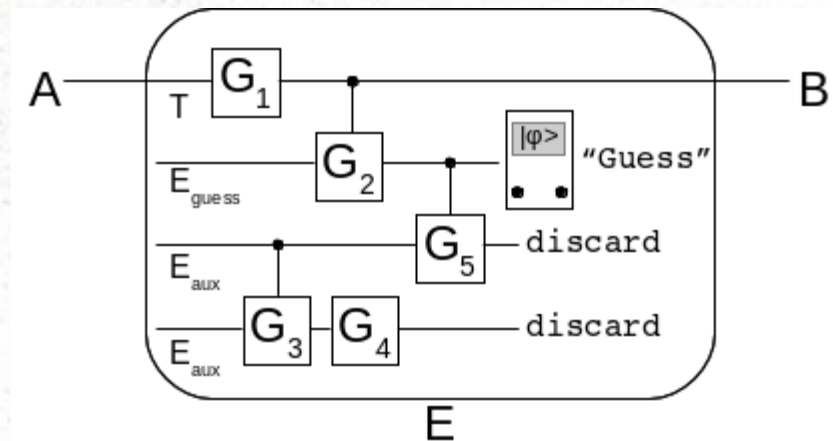
# Solution Representation

- Gate based Solution:
  - Evolve “P” **circuits**
  - Each act on  $M+1$  wires
  - After all  $P$  passes, the “+1” wire is **measured** yielding the guess (the other wires are discarded).



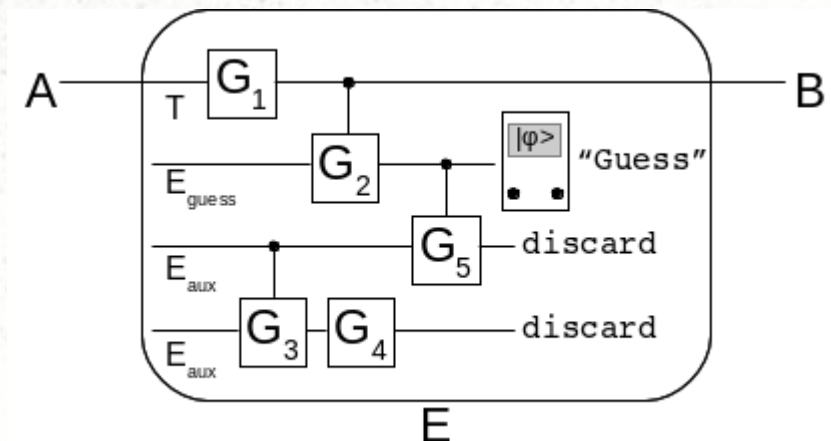
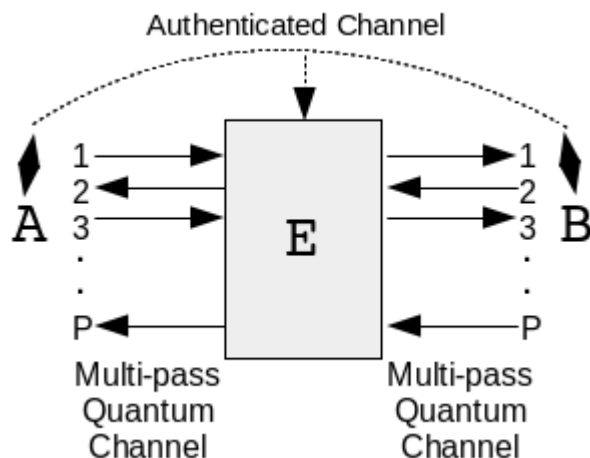
# *Solution Representation*

- We use a modified solution representation introduced in [10] originally used to evolve optimized **quantum algorithms**.
- Let  $G$  be a set of **allowed gates** (user-defined)
  - We use  $G = \{H, \text{CNOT}, R(p, t1, t2)\}$
- Abstractly a Gate is:
  - Type: integer
  - Wires: integer
  - Arguments: doubles



# *Solution Representation*

- A list of gates ( $G_1, G_2, \dots, G_K$ ) represents an attack strategy for one pass of the channel
- A candidate solution, then, is an array of  $P$  lists of gates
- The attack strategy is: Apply circuit 1 on pass 1 (between A and B); Apply circuit 2 on pass 2, etc. Finally: measure the “+1” wire and discard all others

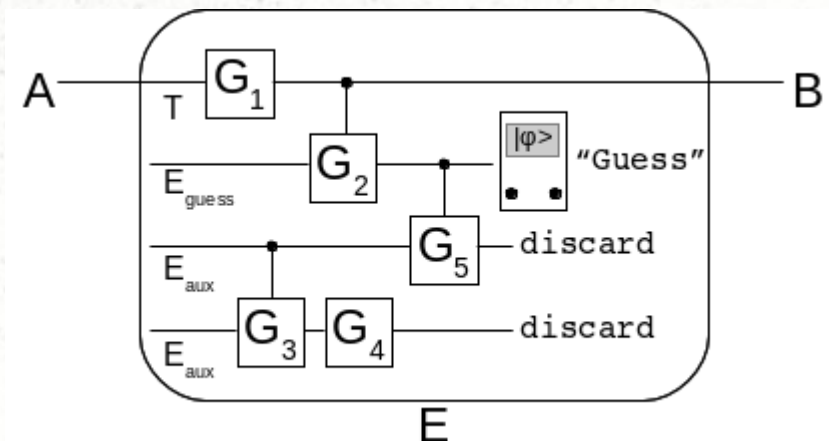
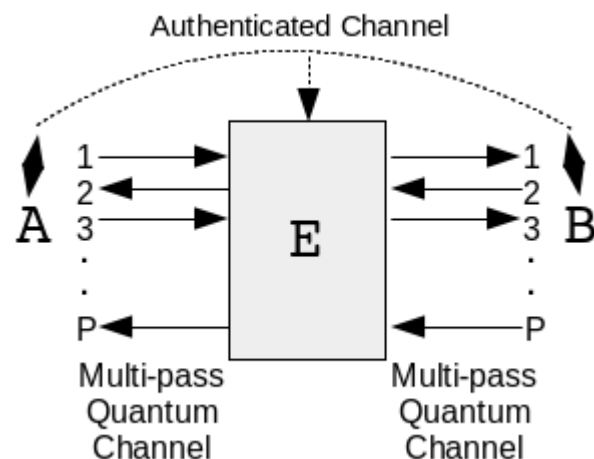


# Solution Representation

- Crossover: Choose  $P$  random crossover points and, for each list of gates, do one point cross-over

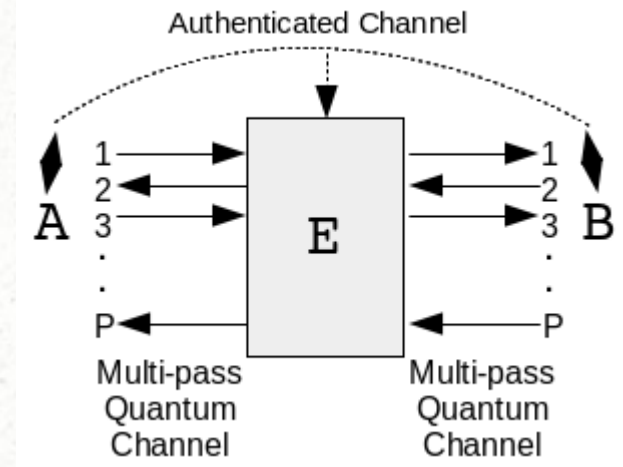
- Mutation:

Create Gate:	20%
Remove Gate:	30%
Change Wire:	70%
Change Gate Type:	20%
Change Gate Attribute:	80%



# *Solution Representation*

- Unitary-based solution:
  - For each  $P$  passes, evolve a unitary attack operator  $U_i$
  - Operators act on  $C^{2n}$
  - Such an operator **could** be constructed as a circuit if the allowed gate size is large enough
  - Apply each unitary operator for each pass
  - Measure the extra  $C^2$  subspace yielding a guess and discard the extra  $C^n$  sub-space



# *Solution Representation*

- Unitary-based solution:
  - We adopted a solution representation from [5]
  - Unitary matrices are decomposed into three arrays totaling  $16n^2$  real parameters
- Crossover: for each array choose a random crossover point
- Mutation: perturb 10% of the array elements by a randomly chosen number

*The Algorithm: Encoding (and  
simulating) a QKD Protocol*

# *QKD Protocol*

- There are two important aspects of any QKD protocol:
  - **computeNoise**
  - **computeKeyRate**
- These are both functions of the protocol itself (e.g., how Alice prepares and sends qubits) and the attack
- Both must be written by the user
- We extended a quantum simulator we initially developed in [6] which supports simple commands like **measure** or **attack**
  - Thus user does not need advanced mathematical abilities to use our system



# QKD Protocol

- Once both functions are written, the simulator will step through the protocol simulating it
  - Whenever an **attack** is called, the next list of gate elements or the next unitary operator (depending on solution representation) is simulated
- At the end, the “+1” wire (or subspace) is measured and the rest thrown out.
- Finally, we are left with a classical distribution with random variables A, B, and E. From this we can compute:

$$r = \lim_{N \rightarrow \infty} \frac{L(N)}{N} = H(A|E) - H(A|B)$$

- Noise is computed similarly.

*The Algorithm: Putting it all together...*

# *The Algorithm*

- First, a user writes a **computeNoise** and **computeKeyRate** function (using calls to our extended simulator)
- Next, GA will create initial population of 100 solutions (using either gate or unitary based approach – user specified)
- Selection for crossover is tournament selection, tournament size 3
- Mutation probability 80%
- Elitism used to keep best solution from previous generation.
- Fitness:  $fit = .55(r + .02)^2 + .45Q^2$
- Final output: min(Q) over all evolved attacks where  $r < 0$

# *Evaluations*

- We tested 5 very different QKD protocols
- Some QKD protocols have known noise tolerances in the memory-less scenario
- Some do not (and prior techniques cannot be used, since no entanglement-based version is known!)

# Evaluations

Experiment		BB84	Six-state BB84	SARG04	B92	SQKD
G(1,4)	Avg.	.173	.257	.221	.202	.126
	Min	.154	.211	.205	.174	.103
	Std.	.029	.045	.022	.022	.02
	#	17/20	15/20	16/20	16/20	7/10
G(3,4)	Avg.	.172	.26	.228	.225	.167
	Min	.154	.211	.206	.194	.167
	Std.	.025	.04	.022	.031	10E-17
	#	20/20	14/20	13/20	15/20	7/10
U(1)	Avg.	.159	.215	.189	.134	.131
	Min	.157	.211	.183	.124	.122
	Std.	.002	.006	.004	.006	.006
	#	20/20	20/20	20/20	20/20	10/10
U(2)	Avg.	.170	.227	.221	.203	.164
	Min	.161	.215	.208	.169	.142
	Std.	.004	.005	.01	.032	.011
	#	20/20	20/20	19/20	20/20	9/10
Known Tolerance [2]		.154	.204	.175	n/a	n/a

G(W, K) = Gate-Based with max wires “W”, max gates “K”

U(n) = Unitary-Based with dimension  $C^{2n}$

# Evaluations

Experiment		BB84	Six-state BB84	SARG04	B92	SQKD
G(1,4)	Avg.	.175	.257	.221	.202	.126
	Min	.154	.211	.205	.174	.103
	Std.	.029	.045	.022	.022	.02
	#	17/20	15/20	16/20	16/20	7/10
G(3,4)	Avg.	.172	.26	.228	.225	.167
	Min	.154	.211	.206	.194	.167
	Std.	.025	.04	.022	.031	10E-17
	#	20/20	14/20	13/20	15/20	7/10
U(1)	Avg.	.159	.215	.189	.134	.131
	Min	.157	.211	.183	.124	.122
	Std.	.002	.006	.004	.006	.006
	#	20/20	20/20	20/20	20/20	10/10
U(2)	Avg.	.170	.227	.221	.203	.164
	Min	.161	.215	.208	.169	.142
	Std.	.004	.005	.01	.032	.011
	#	20/20	20/20	19/20	20/20	9/10
Known Tolerance [2]		.154	.204	.175	n/a	n/a

For BB84, our algorithm finds a solution which agrees with prior, non EA work.

# Evaluations

Experiment		BB84	Six-state BB84	SARG04	B92	SQKD
G(1,4)	Avg.	.173	.257	.221	.202	.126
	Min	.154	.211	.205	.174	.103
	Std.	.029	.045	.022	.022	.02
	#	17/20	15/20	16/20	16/20	7/10
G(3,4)	Avg.	.172	.26	.228	.225	.167
	Min	.154	.211	.206	.194	.167
	Std.	.025	.04	.022	.031	10E-17
	#	20/20	14/20	13/20	15/20	7/10
U(1)	Avg.	.159	.215	.189	.134	.131
	Min	.157	.211	.183	.124	.122
	Std.	.002	.006	.004	.006	.006
	#	20/20	20/20	20/20	20/20	10/10
U(2)	Avg.	.170	.227	.221	.203	.164
	Min	.161	.215	.208	.169	.142
	Std.	.004	.005	.01	.032	.011
	#	20/20	20/20	19/20	20/20	9/10
Known Tolerance [2]		.154	.204	.175	n/a	n/a

For others, it's close, though higher; however our system is applicable to a wider-range of QKD protocols. It is also more flexible in terms of specifying adversary power.

# Evaluations

Experiment		BB84	Six-state BB84	SARG04	B92	SQKD
G(1,4)	Avg.	.173	.257	.221	.202	.126
	Min	.154	.211	.205	.174	.103
	Std.	.029	.045	.022	.022	.02
	#	17/20	15/20	16/20	16/20	7/10
G(3,4)	Avg.	.172	.26	.228	.225	.167
	Min	.154	.211	.206	.194	.167
	Std.	.025	.04	.022	.031	10E-17
	#	20/20	14/20	13/20	15/20	7/10
U(1)	Avg.	.159	.215	.189	.134	.131
	Min	.157	.211	.183	.124	.122
	Std.	.002	.006	.004	.006	.006
	#	20/20	20/20	20/20	20/20	10/10
U(2)	Avg.	.170	.227	.221	.203	.164
	Min	.161	.215	.208	.169	.142
	Std.	.004	.005	.01	.032	.011
	#	20/20	20/20	19/20	20/20	9/10
Known Tolerance [2]		.154	.204	.175	n/a	n/a

No single setting led to best answer for all protocols in our trials

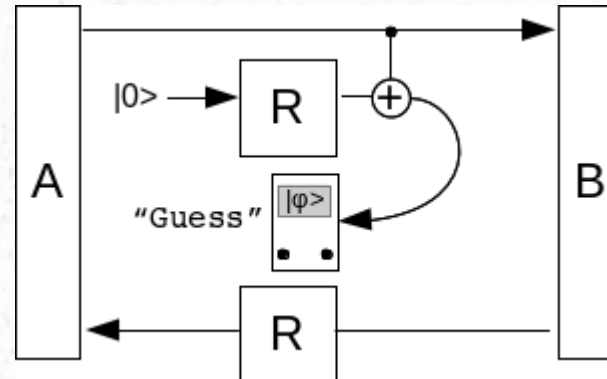


# Evaluations

Experiment		BB84	Six-state BB84	SARG04	B92	SQKD
G(1,4)	Avg.	.173	.257	.221	.202	.126
	Min	.154	.211	.205	.174	.103
	Std.	.029	.045	.022	.022	.02
	#	17/20	15/20	16/20	16/20	7/10
G(3,4)	Avg.	.172	.26	.228	.225	.167
	Min	.154	.211	.206	.194	.167
	Std.	.025	.04	.022	.031	10E-17
	#	20/20	14/20	13/20	15/20	7/10
U(1)	Avg.	.159	.215	.189	.134	.131
	Min	.157	.211	.183	.124	.122
	Std.	.002	.006	.004	.006	.006
	#	20/20	20/20	20/20	20/20	10/10
U(2)	Avg.	.170	.227	.221	.203	.164
	Min	.161	.215	.208	.169	.142
	Std.	.004	.005	.01	.032	.011
	#	20/20	20/20	19/20	20/20	9/10
Known Tolerance [2]		.154	.204	.175	n/a	n/a

Very easy to analyze new protocols unlike prior work.

# Evaluations



Sample attack found by our GA against SQKD (a 2-pass protocol).

For this particular protocol, the key is transmitted on the first pass while the reverse is used for error checking.

Our GA found an attack which incorporates this by using the forward channel only for E's guess while trying to "reverse" the noise in the reverse channel.

This property of the protocol was not specifically described to the GA – we only encoded the steps of the protocol into the system and this attack strategy was evolved.

Thus, GA was able to take advantage of the structure of the protocol to discover an optimal attack.

# *Closing Remarks*

- We showed how genetic algorithms may be used to study the security of QKD protocols when faced with practical memory-less adversaries
- Our method can be used to study a wide-range of protocols without requiring complex mathematical reductions to entanglement-based versions
  - One simply enters the protocol's steps and specifies when an adversary has an opportunity to attack
- We evaluated on five very different QKD protocols and compared to current known noise tolerances (when such knowledge is available)
  - Our evaluations also showed the GA is able to take advantage of the structure of the QKD protocol.

# *Future Work*

- Evolve attacks based on actual optical devices
  - E.g., evolve optical devices/experiments instead of abstract gates
- Take advantage of device imperfections (both in A/B's devices and also in E's devices)
  - Photon loss, noisy state preparations, faulty measurements
- Improve efficiency of software implementation and improve GA parameters
- **Ultimate goal: have a suite of tools allowing researchers and practitioners to test security of QKD protocols quickly in a variety of security scenarios (which are difficult to analyze mathematically).**

*Thank you! Questions?*

# References

- [2] A. Bocquet and R. Alleaume and A. Leverrier. Optimal eavesdropping on quantum key distribution without quantum memory. *Journal of Physics A*. 45 2 (2011), 025305
- [4] I. Csiszar and J. Korner. Broadcast channels with confidential messages. *IEEE Trans. Info. Theory*. 24, 3, (1978) 339-348
- [5] S.R. Hutsell and G.W. Greenwood. Applying Evolutionary Techniques to Quantum Computing Problems. In *Proc. IEEE CEC 2007* pp. 4081-4085
- [6] W. Krawec. A Genetic Algorithm to analyze the security of quantum cryptographic protocols. In *Proc. IEEE CEC 2016*. pp. 2098-2105
- [10] B. Rubinstein. Evolving quantum circuits using genetic programming. In *Proc. Evolutionary Computation*. Vol. 1 (2001) pp. 144-151

# References

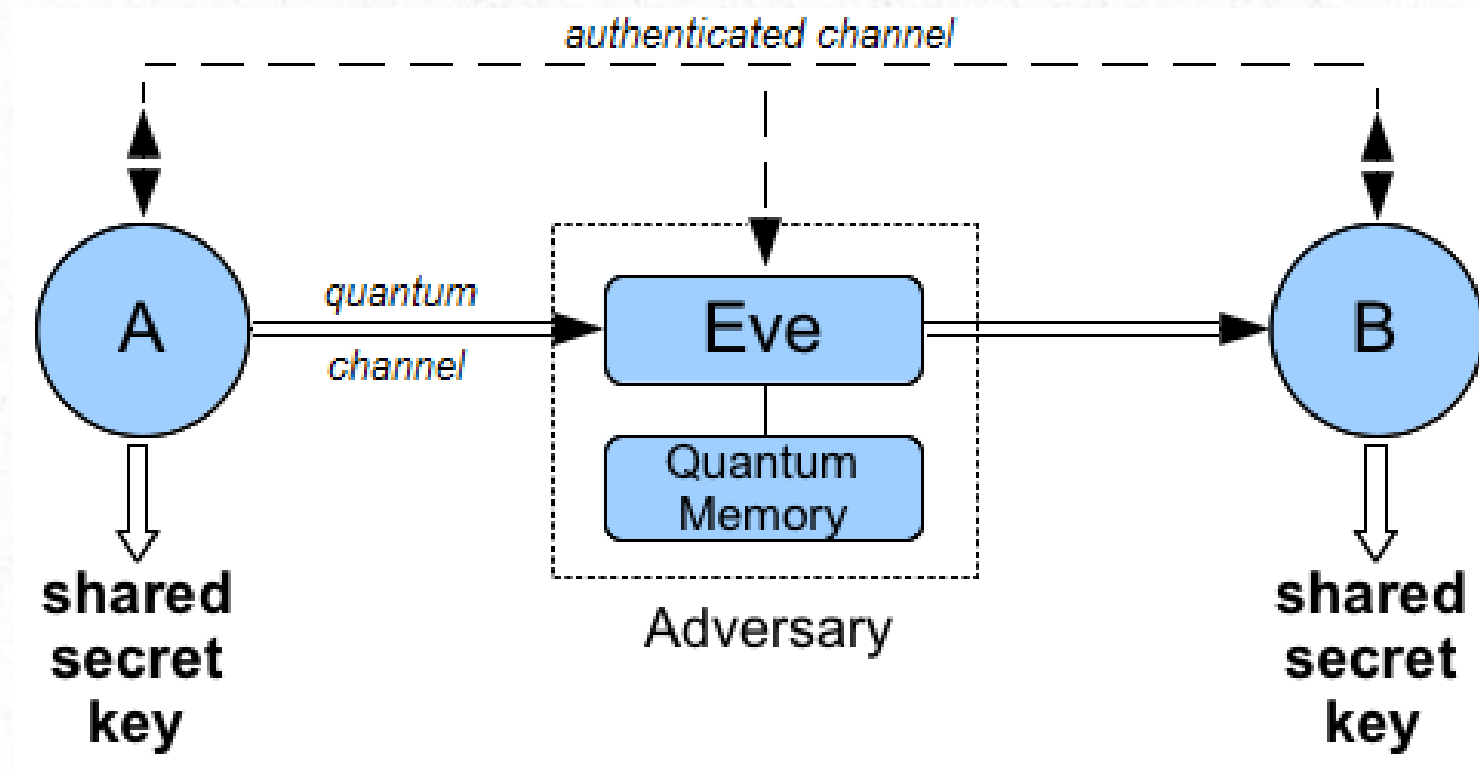
- C.H. Bennett and G. Brassard, 1984, Quantum cryptography: Public key distribution and coin tossing. in Proc. IEEE Int. Conf. on Computers, Systems, and Signal Processing. Vol 175, NY.
- C.H. Bennett, 1992, Quantum cryptography using any two nonorthogonal states. Phys. Rev. Lett., 68:3121-3124.
- M. Boyer, D. Kenigsberg, and T. Mor, 2007, Quantum Key Distribution with classical bob, in ICQNM.
- C.H.F. Fung and H.K. Lo, 2006, Security proof of a three-state quantum key distribution protocol without rotational symmetry. Phys. Rev. A, 74:042342.
- J. Bae and A. Acin. Key distillation from quantum channels using two-way communication protocols. Physical Review A, 75(1):012334, 2007.
- W.O. Krawec, 2017, Quantum Key Distribution with Mismatched Measurements over Arbitrary Channels. To appear, Quantum Information & Computation (arXiv:1608.07728)

## *References (cont.)*

- H. Lu and Q.-Y. Cai, 2008, Quantum key distribution with classical Alice, *Int. J. Quantum Information* 6, 1195.
- R. Renner, N. Gisin, and B. Kraus, 2005, Information-theoretic security proof for QKD protocols. *Phys. Rev. A*, 72:012332.
- R. Renner, 2007, Symmetry of large physical systems implies independence of subsystems, *Nat. Phys.* 3, 645.
- V. Scarani, A. Acin, G. Ribordy, and N. Gisin, 2004, *Phys. Rev. Lett.* 92, 057901.
- Z. Xian-Zhou, G. Wei-Gui, T. Yong-Gang, R. Zhen-Zhong, and G. Xiao-Tian, 2009, Quantum key distribution series network protocol with m-classical bobs, *Chin. Phys. B* 18, 2143.
- Xiangfu Zou, Daowen Qiu, Lvzhou Li, Lihua Wu, and Lvjun Li, 2009, Semiquantum key distribution using less than four quantum states. *Phys. Rev. A*, 79:052312.



# Quantum Key Distribution



# BB84: Basic Idea

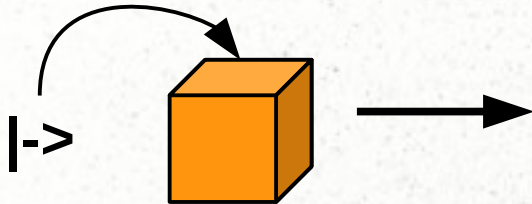
$$\begin{aligned} 0 &== \{ |0\rangle, |+\rangle \} \\ 1 &== \{ |1\rangle, |-\rangle \} \end{aligned}$$

**Alice**

**Eve**

**Bob**

Key-bit = 1  
Basis = X



$$\begin{aligned} 0 &== \{ |0\rangle, |+\rangle \} \\ 1 &== \{ |1\rangle, |-\rangle \} \end{aligned}$$

# BB84: Basic Idea

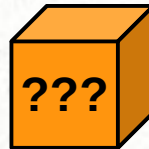
$$\begin{aligned} 0 &== \{ |0\rangle, |+\rangle \} \\ 1 &== \{ |1\rangle, |-\rangle \} \end{aligned}$$

**Alice**

Key-bit = 1  
Basis = X

**Eve**

Key-guess = ?  
Basis = ???  
Basis-Guess = Z



**Bob**

$$\begin{aligned} 0 &== \{ |0\rangle, |+\rangle \} \\ 1 &== \{ |1\rangle, |-\rangle \} \end{aligned}$$

# BB84: Basic Idea

$$\begin{aligned} 0 &== \{ |0\rangle, |+\rangle \} \\ 1 &== \{ |1\rangle, |-\rangle \} \end{aligned}$$

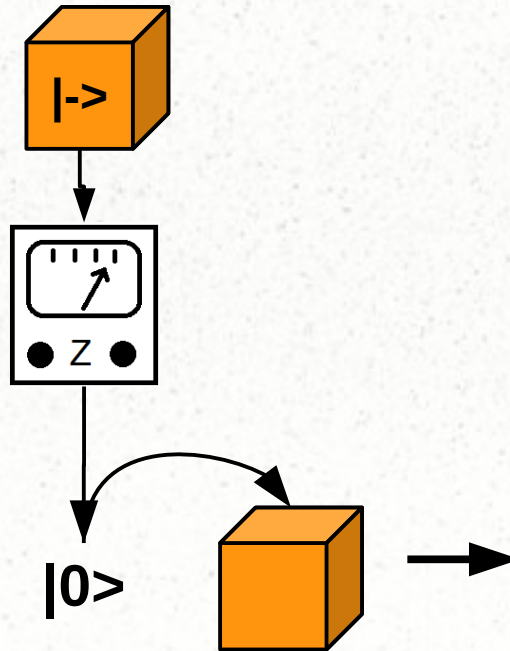
**Alice**

Key-bit = 1  
Basis = X

**Eve**

Key-guess = 0  
Basis = ???  
Basis-Guess = Z

**Bob**



$$\begin{aligned} 0 &== \{ |0\rangle, |+\rangle \} \\ 1 &== \{ |1\rangle, |-\rangle \} \end{aligned}$$

# BB84: Basic Idea

$$\begin{aligned} 0 &== \{ |0\rangle, |+\rangle \} \\ 1 &== \{ |1\rangle, |-\rangle \} \end{aligned}$$

**Alice**

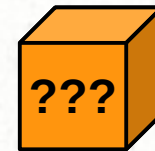
Key-bit = 1  
Basis = X

**Eve**

Key-guess = 0  
Basis = ???  
Basis-Guess = Z

**Bob**

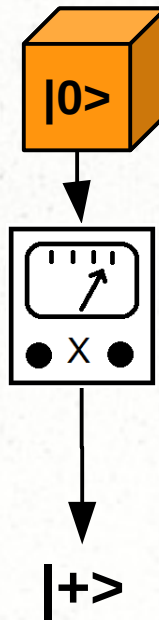
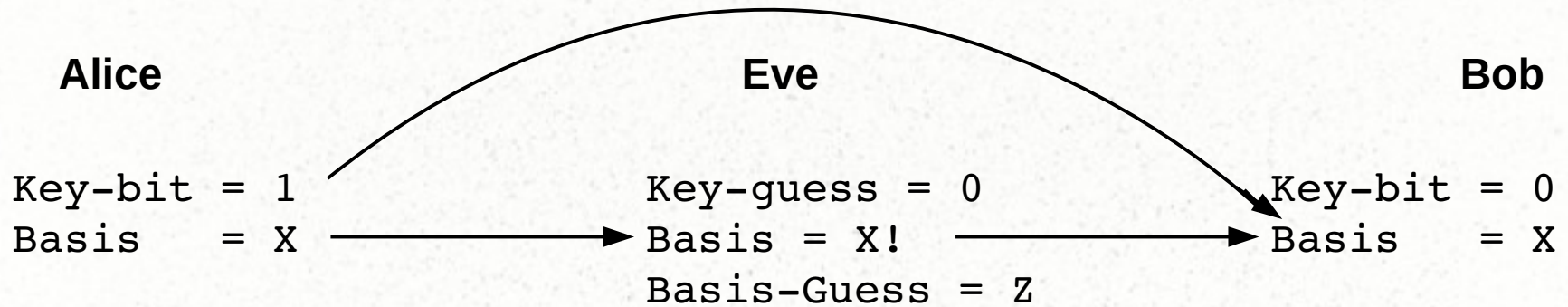
Key-bit = ?  
Basis = ?



$$\begin{aligned} 0 &== \{ |0\rangle, |+\rangle \} \\ 1 &== \{ |1\rangle, |-\rangle \} \end{aligned}$$

# BB84: Basic Idea

$$\begin{aligned} 0 &== \{ |0\rangle, |+\rangle \} \\ 1 &== \{ |1\rangle, |-\rangle \} \end{aligned}$$

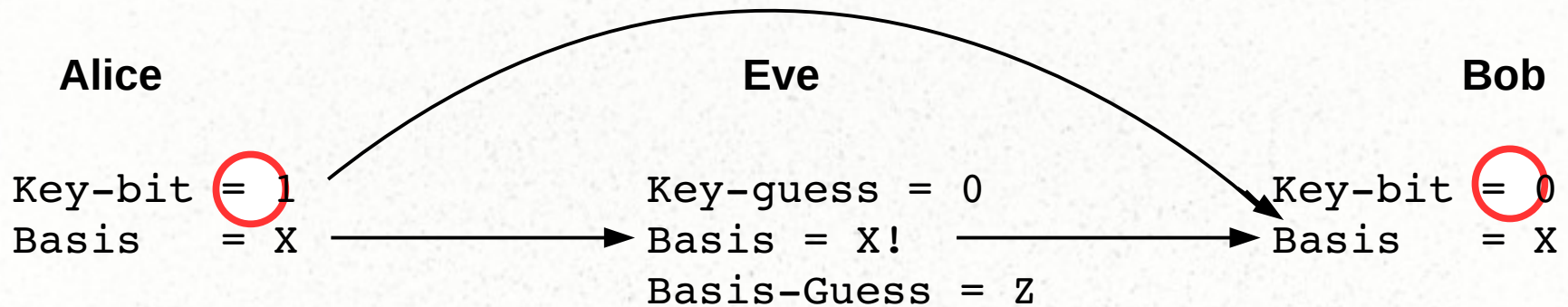


$$\begin{aligned} 0 &== \{ |0\rangle, |+\rangle \} \\ 1 &== \{ |1\rangle, |-\rangle \} \end{aligned}$$

# BB84: Basic Idea

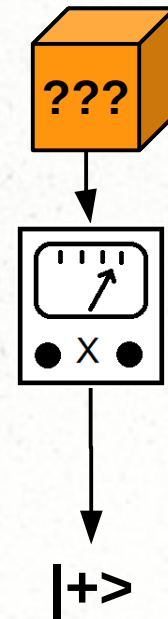
$$0 == \{ |0\rangle, |+\rangle \}$$

$$1 == \{ |1\rangle, |-\rangle \}$$



*Any attack induces errors in the quantum channel which A and B may detect!*

*Goal: Bound E's information gain as a function of this error rate.*



$$0 == \{ |0\rangle, |+\rangle \}$$

$$1 == \{ |1\rangle, |-\rangle \}$$