



UNIVERSITÀ DEGLI STUDI ROMA TRE

Dipartimento di Ingegneria

Corso di Laurea in Ingegneria Informatica

Tesi Di Laurea

Sviluppo di un simulatore per la crittografia quantistica a variabili continue

Laureando

Kevin Santodonato

Matricola 548019

Relatore

Prof. Matteo Rosati

Anno Accademico 2022/2023

Questa è la dedica

Ringraziamenti

Grazie a tutti

Indice

Indice	iv
Elenco delle figure	vi
Introduzione	vii
1 Descrizione quantistica di un segnale ottico	1
1.1 Stati coerenti e le loro proprietà	1
1.2 Misurazione dei segnali e incertezza quantistica	3
2 Breve storia della QKD	5
2.1 CV-QKD con modulazione gaussiana	6
2.1.1 Produzione, trasmissione e ricezione del segnale quantistico	6
2.1.2 Stima dei parametri	8
2.1.3 Riconciliazione delle informazioni	9
2.2 Discussione della sicurezza contro attacchi di lettura del segnale	10
3 Simulazione	14
3.1 Preparazione, trasmissione e misura	15
3.1.1 Preparazione dello stato coerente	15

3.1.2	Trasmissione su canale rumoroso	15
3.1.3	Misura omodina del segnale	16
3.2	Stima dei parametri	16
3.2.1	Sifting	16
3.2.2	Stima delle varianze dei dati trasmessi, ricevuti e delle loro covarianza . . .	16
3.2.3	Stima mutue informazioni	16
3.2.4	Accertamento sicurezza trasmissione	16
3.3	Riconciliazione	16
3.3.1	Generazione stringa random di bit	16
3.3.2	Algoritmo Progressive Edge Growth	16
3.3.3	Messaggio fittizio	16
3.3.4	Algoritmo Sum-Product	16
Conclusioni e sviluppi futuri		17
Bibliografia		18

Elenco delle figure

1.1	Figura da inserire—> Stato coerente —>Da apportare modifica consigliata dal professore	2
1.2	Figura da inserire—> confronto tra gaussiano con spia e senza spia	4
2.1	Schema che descrive le fasi del protocollo facendo particolare attenzione al canale quantistico e canale classico.	
2.2	Schema di comunicazione tra Alice e Bob Sezione ??	11
2.3	Schema di comunicazione tra Alice e Bob Sezione ??	12
2.4	Schema di comunicazione tra Alice e Bob Sezione ??	12
2.5	Schema di comunicazione tra Alice e Bob Sezione ??	13

Introduzione

La crittografia, da migliaia di anni, ha costituito il fondamento delle comunicazioni segrete e della protezione dei dati sensibili. Nel corso della storia umana, dall'antico Egitto fino alla metà del secolo scorso, la crittografia veniva considerata una arte e tutti i sistemi crittografici e cifrari che venivano realizzati erano considerati sicuri fino al momento in cui qualcuno riusciva a trovare una falla nel sistema e decifrare il messaggio senza una chiave.

Nel 1949, grazie a Claude Shannon, inizia l'era moderna della crittografia la quale, da questo punto in poi della storia, viene considerata oltre che un arte anche una scienza. Shannon nel '49 pubblica un articolo denominato "Communication Theory of Secrecy Systems"[Sha49] nel quale comprova, per la prima volta, la sicurezza del cifrario di Vernam (comunemente chiamato One Time Pad) attraverso dimostrazioni matematiche.

Il cifrario di Vernam si basa su un concetto molto semplice: per trasmettere un messaggio cifrato tra due parti, che vengono comunemente chiamate Alice e Bob, viene prodotto una chiave della lunghezza del messaggio che si vuole trasmettere. La chiave deve essere condivisa un modo sicuro tra le due parti, ad esempio Alice e Bob possono incontrarsi e scambiare la chiave. Successivamente la chiave viene utilizzata da Alice per codificare il messaggio e da Bob per decodificarlo, a questo punto la chiave va scartata e per trasmettere un nuovo messaggio è necessario produrre una nuova chiave.

Questo tipo di cifrari rientrano nella classe dei crittosistemi simmetrici (a chiave privata) nei quali per codifica e decodifica viene utilizzata la stessa chiave. Di questa classe di crittosistemi la sicurezza è

garantita da quanto dimostrato da Shannon però si può subito notare un problema non indifferente: la chiave deve essere condivisa in qualche modo tra Alice e Bob. Condividere una chiave, in alcune situazioni, potrebbe risultare un'operazione molto dispendiosa ed è per questo che nel 1976 sono state gettate le basi per una nuova classe di crittosistemi denominati crittosistemi asimmetrici [DH76].

L'idea alla base dei crittosistemi asimmetrici (a chiave pubblica) è quella di utilizzare due chiavi diverse per la codifica e la decodifica. La prima implementazione di un protocollo che rientra in questa classe è stata realizzata nel 1978 e prende in nome di RSA [RSA78], il quale oggi è ampiamente utilizzato. In crittosistemi a chiave pubblica se Bob vuole ricevere dei messaggi codificati con una chiave pubblica per prima cosa deve generare per se stesso una chiave privata. Da questa chiave produce una chiave pubblica che rivela ad Alice la quale utilizza la chiave pubblica per codificare il messaggio e trasmetterlo a Bob. Una volta ricevuto il messaggio Bob lo decodifica utilizzando la propria chiave privata.

La sicurezza di questi sistemi, a differenza di quelli a chiave privata, non si basa su dimostrazioni matematiche ma solamente sulla complessità computazionale. Questo perché per la produzione delle chiavi vengono utilizzate delle funzioni matematiche molto semplici da calcolare ma molto difficili da invertire. In termini di complessità computazionale difficile sta a significare che il tempo necessario per eseguire un'operazione cresce esponenzialmente con la lunghezza della chiave, mentre per operazioni semplici il tempo di esecuzione è polinomiale rispetto alla lunghezza della chiave. Ad esempio, la sicurezza del protocollo RSA si basa proprio sulla difficoltà di fattorizzazione di interi molto grandi.

Non essendoci prove matematiche che garantiscano la sicurezza dei crittosistemi a chiave pubblica non è da escludere la probabilità che sia possibile realizzare un algoritmo classico che abbia una complessità computazionale polinomiale per la fattorizzazione di grandi interi. Di fatto la sicurezza di questa classe di protocolli è già minacciata da un algoritmo per computer quantistici, ideato da Peter Shor nel 1994, che permette la fattorizzazione di grandi interi con complessità polinomiale [Sho94].

Per questo al giorno d'oggi è sempre più importante la ricerca su protocolli di distribuzione quantistica di chiavi (QKD) che utilizzano principi della fisica quantistica con lo scopo di condividere chiavi su canali pubblici. Questi protocolli rientrano nella classe di crittosistemi simmetrici perché la codifica e la decodifica dei messaggi trasmessi viene effettuata con una chiave privata comune ad Alice e Bob. Il potere della QKD risiede nel fatto che grazie alla fisica quantistica si ha la possibilità di determinare se la condivisione dei messaggi necessari per la produzione della chiave è avvenuta in modo sicuro o meno, in parole povere si è in grado di determinare se una spia (Eve) è venuta in possesso di questi messaggi.

Il primo protocollo ad essere stato proposto per la distribuzione quantistica di chiavi è il BB84 che prende il nome dai suoi ideatori Charles H. Bennett e Gilles Brassard che lo hanno realizzato nel 1984 [BB84]. Questo protocollo per realizzare la comunicazione utilizza due canali di trasmissione, uno classico e pubblico e l'altro quantistico e privato. Alice trasmette sul canale quantistico a Bob una sequenza di fotoni polarizzati in modo casuale in due basi, ognuna delle quali ha due gradi di polarizzazione. Nella prima base il fotone può essere polarizzato a 0 o 90 gradi e verrà considerato come un bit 0, mentre nella seconda a 45 o 135 gradi e verrà considerato come bit 1. In ricezione Bob effettuerà la misura della polarizzazione del fotone scegliendo una base con probabilità uniforme dalle due basi disponibili. Alla fine della trasmissione Bob sarà in possesso dei gradi di polarizzazione misurati i quali verranno convertiti in 0 e 1 seguendo la stessa convenzione di Alice. Da notare che parte dei dati di Bob differiranno da quelli inviati da Alice perché Bob ha effettuato la misura utilizzando la base sbagliata. Successivamente Bob comunicherà ad Alice quale base ha utilizzato per la misura di ogni fotone ed Alice scarterà tutti quei fotoni la cui base di trasmissione e misura non combaciano, e lo comunicherà a Bob. Fatto ciò Alice e Bob saranno in possesso, almeno in teoria, della stessa stringa random di bit. Quello che potrebbe succedere è che durante la trasmissione Eve intercetti i fotoni inviati da Alice per effettuarne una misura e poi inviare il risultato della propria misura a Bob.

Tuttavia questo altererebbe il risultato della misura di Bob perchè Eve non può sapere con quale base è stato polarizzato il fotone, e di conseguenza non può sapere con quale base misurarlo. Misurando un fotone con la base sbagliata nel momento in cui lo ritrasmetterà a Bob egli riceverà un fotone polarizzato in modo differente rispetto a quello inviato da Alice. Questo permetterà ad Alice e Bob di determinare la presenza di una spia andando a confrontare una porzione di bit misurati con la stessa base utilizzata in trasmissione e quindi ritenuti uguali. Durante il confronto se si riscontrano bit diversi si determina la presenza di una spia. Se Eve ha intercettato troppi fotoni non viene prodotta nessuna chiave e si ricomincia il protocollo da capo, in caso contrario si utilizzano i bit rimasti segreti come chiave crittografica[Zha18].

Capitolo I

Descrizione quantistica di un segnale ottico

I.1 Stati coerenti e le loro proprietà

I segnali elettromagnetici possono essere descritti sia attraverso onde che attraverso particelle, ovvero fotoni. Ad esempio, il segnale prodotto da un LASER è ben rappresentato da uno stato quantistico detto stato coerente. Esso comunemente è rappresentato $|\alpha_j\rangle = |q_j + ip_j\rangle$ attraverso la notazione introdotta da Paul Dirac che prende il nome di *bra-ket*. Dal punto di vista della fisica classica, il numero complesso α è strettamente legato all'ampiezza di oscillazione del campo elettrico corrispondente al segnale. Esso è costituito da due componenti reali, q e p , dette componenti di quadratura. Dal punto di vista della fisica quantistica, tuttavia, non è possibile assegnare un valore preciso a queste grandezze. Infatti, i valori di q e p che utilizziamo per rappresentare lo stato coerente sono solo valori medi. Quando proviamo a fare una misurazione delle grandezze fisiche associate, che indichiamo con \hat{q} e \hat{p} , esse si comportano come variabili aleatorie che rispondono alla stessa distribuzione di probabilità gaussiana che, in opportune unità di misura, ha varianza unitaria. Il motivo per cui non

rappresentano un valore esatto è perché, a differenza fisica classica, nella fisica quantistica le quantità, anche scalari, sono degli operatori che assumono un valore esatto solamente in fase di misura.

Ogni stato coerente presenta un numero medio di fotoni che può essere calcolato nel seguente modo:

$$\langle n_j \rangle = |a_j|^2 = q^2 + p^2 \quad (1.1)$$

Il numero medio di fotoni di uno stato coerente è associato all'energia del segnale di luce che viene inviato, maggiore è il numero di fotoni maggiore è l'energia del segnale.

Come detto precedentemente, nel nostro caso gli stati coerenti si presentano come una distribuzione di probabilità gaussiana e possono essere rappresentati graficamente nel piano di Gauss come una nuvola di punti i quali avranno appunto probabilità gaussiana di essere estratti.

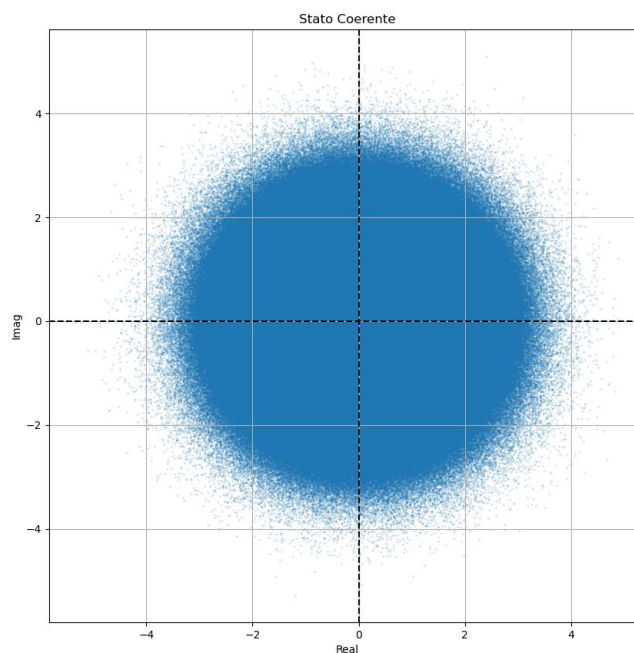


Figura 1.1: Figura da inserire—> Stato coerente —> Da apportare modifica consigliata dal professore

1.2 Misurazione dei segnali e incertezza quantistica

Partendo dal caso ideale, in cui il canale di comunicazione è perfetto e quindi non aggiunge rumore durante la trasmissione, Bob riceverà uno stato coerente con la forma vista in figura 1.1. Da notare che l'incertezza della misura è presente anche nel caso di un canale di comunicazione ideale, questo perché non è dovuta solamente al rumore esterno, come ci si aspetterebbe in un canale di comunicazione classico, ma è una caratteristica intrinseca di uno stato quantistico la quale non può essere rimossa **((o attenuata) da chiedere al professore)** in base al principio di indeterminazione di Heisenberg. Questa incertezza si presenta proprio in fase di misura, perché come detto nella sezione 1.1, i valori di quadratura q e p sono solamente valori medi mentre i valori \hat{q} o \hat{p} che otteniamo da una misura 2.1.1 si comportano come variabili aleatorie con distribuzione di probabilità gaussiana.

In un trasmissione reale, l'aggiunta di rumore esterno è inevitabile e in questo rumore è presente anche quello prodotto ad un eventuale spia (Eve) che tenta anch'essa di misurare il segnale modificandolo, questo rende la misura di Bob ancora più incerta.

Per concludere possiamo dire che le QKD utilizzano le caratteristiche dei segnali quantistici per cui c'è un'incertezza nella misura anche in assenza di rumore esterno e che una misura introduce rumore proprio per determinare l'eventuale intromissione di Eve nella comunicazione e quindi la possibilità o meno di creare una chiave crittografica sicura.

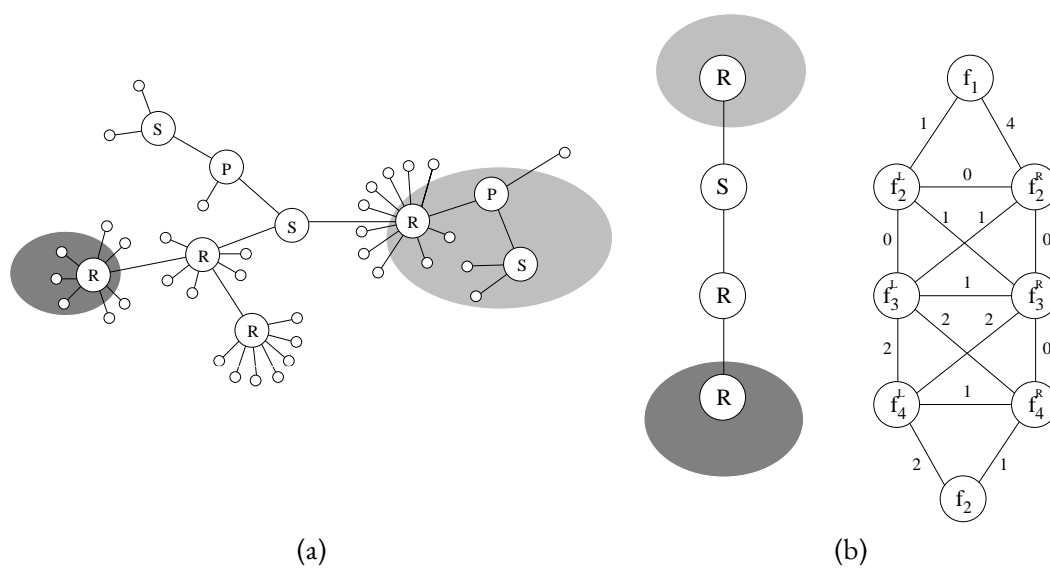


Figura 1.2: Figura da inserire—> confronto tra gaussiano con spia e senza spia

Capitolo 2

Breve storia della QKD

La QKD (Quantum Key distribution), o Distribuzione Quantistica delle Chiavi, è una tecnologia di crittografia quantistica che permette la distribuzione di chiavi crittografiche con un livello di sicurezza inattaccabile anche da computer quantistici. La QKD può essere suddivisa in due macro-classi DV-QKD e CV-QKD. Nella DV-QKD (Discrete Variable Quantum Key Distribution) le informazioni vengono trasmesse utilizzando stati quantici discreti, ad esempio la polarizzazione degli fotoni, che viene misurata attraverso apposite apparecchiature per ogni fotone; al contrario nella CV-QKD (Continuous Variable Quantum Key Distribution) vengono misurate variabili continue delle onde elettromagnetiche come ad esempio la fase.

La sicurezza è garantita dai principi della fisica quantistica che ci permettono di rilevare eventuali intercettazioni da parte di Eve (spia) nella trasmissione su canale quantistico tra Alice (mittente) e Bob (destinatario).

La rilevazione è dovuta al fatto che durante la misura da parte di Eve lo stato quantistico viene alterato facendo diminuire l'energia del segnale e introducendo del rumore, quindi dell'incertezza. All'atto pratico Eve intercetta lo stato coerente, rappresentato come in figura 1.1, ne effettua una misura per poi ritrasmetterlo sul canale. Bob riceverà uno stato coerente con del rumore aggiunto e un minor

numero di fotoni, ciò comporta che la gaussiana che lo rappresenta sarà caratterizzata da una varianza superiore a quella stimata e un valor medio più vicino allo zero. Per questo motivo, si effettuano numerosi run di trasmissione quantistica. In una fase preliminare si utilizza un certo numero di run per stimare i parametri di rumore del canale e, da essi, decidere se la trasmissione sta avvenendo in modo sicuro oppure no (si veda Sez. 2.1.2). Nel caso in cui il controllo non vada a buon fine, la trasmissione corrente viene abortita perché non è possibile garantire la sicurezza della chiave crittografica.

2.1 CV-QKD con modulazione gaussiana

Il protocollo di distribuzione quantistica di chiavi a variabili continue con modulazione gaussiana è un protocollo molto indicato per lo sviluppo e l'utilizzo su larga scala nel mondo reale data la sua affinità con le infrastrutture oggi già esistenti, come ad esempio, i canali di comunicazione in fibra ottica. Come enunciato all'inizio del capitolo con questo protocollo si effettuano misure su variabili continue delle onde elettromagnetiche e in particolare nei protocolli con modulazione gaussiana gli stati coerenti da trasmettere sul canale di comunicazione vengono estratti da una distribuzione normale centrata in zero e una certa deviazione standard che viene definita in base ad alcuni fattori.

Il protocollo può essere suddiviso in due sezioni: la prima parte ha effettivamente a che fare con segnali quantistici mentre la seconda parte opera con segnali e dati classici. La prima sezione comprende la scelta e la trasmissione degli stati coerenti su fibra ottica da parte di Alice e termina nel momento in cui Bob effettua la misura; la seconda parte comprende il sifting, la stima dei parametri e la riconciliazione.

2.1.1 Produzione, trasmissione e ricezione del segnale quantistico

Alice prepara gli stati coerenti andando a estrarre i valori delle componenti di quadratura q e p da una distribuzione normale centrata in zero ($Q \sim P \sim N(0, V_{mod})$) [LPF⁺18]. Da notare che la

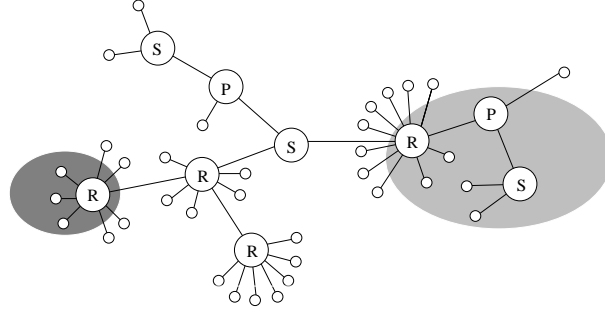


Figura 2.1: Schema che descrive le fasi del protocollo facendo particolare attenzione al canale quantistico e canale classico.

distribuzione normale dalla quale Alice estrae le componenti di quadratura non ha nulla a che fare con le distribuzioni normali che caratterizzano gli stati coerenti ma rappresenta in modo in cui i dati vengono modulati. Dopo la preparazione Alice trasmette a Bob gli stati coerenti attraverso un canale quantistico.

Il canale è caratterizzato da un certo valore di trasmittanza T e di rumore ξ . A causa della trasmittanza durante la trasmissione il segnale perde potenza quindi le gaussiane che descrivono lo stato coerente saranno centrata in un valore più vicino allo zero rispetto al momento della preparazione, mentre il rumore che viene introdotto fa aumentare la varianza delle gaussiane così da accrescere l'incertezza nelle misure da parte di Bob.

Per dare una rappresentazione matematica di quello che avviene ad uno stato coerente durante la trasmissione, possiamo andare a considerare un generico stato scelto da Alice $|a_i\rangle = |q_A + ip_A\rangle$. Come abbiamo detto nella sezione 1.1 le componenti di quadratura rispondono ad una distribuzione

di probabilità normale, quindi avremo:

$$\begin{aligned} q_A &\sim N(q_A, 1) \\ p_A &\sim N(p_A, 1) \end{aligned} \tag{2.1}$$

In ricezione, a causa del canale, Bob andrà ad effettuare la misura su uno stato coerente alterato $|b_i\rangle = |q_B + ip_B\rangle$ dove le componenti di quadratura rispondono alle seguenti distribuzioni gaussiane:

$$\begin{aligned} q_B &\sim N(\sqrt{T}q_B, 1 + \xi) \\ p_B &\sim N(\sqrt{T}p_B, 1 + \xi) \end{aligned} \tag{2.2}$$

Per la misura dello stato coerente possono essere adottati due metodi diversi:

- **misura omodina:** Bob decide se misurare la componente di quadratura q o p , la decisione viene presa attraverso una distribuzione di probabilità uniforme.
- **misura eterodina:** questa misura permette di misurare entrambe le componenti contemporaneamente, però c'è un prezzo da pagare perché effettuando questa misura viene aggiunto del rumore e la varianza delle distribuzioni normali ha un incremento di 1.

In caso di misura omodina è necessaria un'operazione addizionale cioè il sifting. A questo punto del protocollo Alice è in possesso del doppio dei dati di Bob (in quanto Bob misura solo una della due componenti per ogni round) e per far sì che abbiano la stessa quantità di dati correlati Bob rivela ad Alice, su canale classico pubblico, quale componente ha misurato ad ogni round; Alice prende nota della componenti misurate e scarta le altre.[MFZG18]

2.1.2 Stima dei parametri

Dopo avere terminato la fase di trasmissione degli stati, Alice e Bob rivelano una porzione random di dati in modo tale da confrontare ciò che è stato effettivamente inviato da Alice con le misure di Bob. Da questo confronto sono in grado di stimare l'effettiva trasmittanza e rumore in eccesso del

canale di trasmissione e con queste stime possono calcolare l'informazione mutua I_{AB} (informazione mutua tra Alice e Bob) e l'informazione mutua tra Eve e Bob χ_{EB} . Nel caso in cui χ_{EB} risulti essere maggiore di βI_{AB} ¹ il protocollo viene abortito.

2.1.3 Riconciliazione delle informazioni

Se la stima dei parametri ha avuto successo Alice e Bob effettuano una correzione degli errori dei segnali scambiati. La riconciliazione può avvenire in due forme:

- **diretta:** i dati di Alice sono i dati primari e Bob corregge i suoi dati di conseguenza.
- **inversa:** in questo caso sono i dati di Bob ad essere primari, vengono inviati ad Alice la quale li utilizza per correggere i propri dati.

La riconciliazione diretta presenta un problema: non può essere utilizzata per valori di trasmittanza troppo bassi, questo perché Eve avrebbe più informazione sui dati di Alice rispetto a Bob e quindi risulterà impossibile creare una chiave crittografica sicura. D'altro canto la riconciliazione inversa non presenta questo problema quindi è possibile utilizzarla anche con valori bassi di trasmittanza, però è da tenere in considerazione che più bassa è la trasmittanza più sarà distruttivo l'effetto del rumore del canale. [LPF⁺18]

A questo punto del protocollo Alice e Bob sono possesso del subset di dati non utilizzati per la stima dei parametri che andiamo a chiamare \mathbf{X}_0 e \mathbf{Y}_0 ; prima di effettuare la riconciliazione vengono prodotte, a partire dai dati in loro possesso, altre due sequenze di dati correlati.

Quello che ci aspettiamo è che le varianze dei dati in possesso ad Alice e Bob siano rispettivamente:

$$\begin{aligned} V_{X_0} &= V_{mod} + 1 \\ V_{Y_0} &= TV_{mod} + 1 + \xi \end{aligned} \tag{2.3}$$

¹ β rappresenta l'accuratezza nel processare i dati dopo la trasmissione e viene calcolata come il rapporto tra il code-rate scelto e la mutua informazione tra Alice e Bob

Detto questo possiamo normalizzare \mathbf{X}_0 e \mathbf{Y}_0 in questo modo:

$$\begin{aligned}\mathbf{X} &= \frac{\mathbf{X}_0}{\sqrt{V_{mod}}} \\ \mathbf{Y} &= \frac{\mathbf{Y}_0}{\sqrt{TV_{mod} + 1 + \xi}}\end{aligned}\tag{2.4}$$

Così facendo abbiamo ottenuto altre due sequenze di dati \mathbf{X} e \mathbf{Y} , correlate tra loro dall'equazione $\mathbf{X} = \mathbf{Y} + \mathbf{Z}$, dove \mathbf{Z} è una variabile aleatoria che risponde ad una distribuzione normale centrata in zero e con varianza $V_z = \frac{1}{V_{mod}}$. Quindi normalizzando in questo modo abbiamo ottenuto lo stesso effetto che avremmo avuto se Bob avesse inviato ad Alice i proprio dati normalizzati \mathbf{Y} su un canale che aggiunge solamente rumore gaussiano [MFZG18].

Il passo successivo per Bob sarà quello di generare una stringa di bit random $\underline{s} \in \{0, 1\}^k$. Attraverso algoritmi di codifica LDPC viene calcolata una matrice \mathbf{H} necessaria per aggiungere alla stringa \underline{s} dei bit ridondanti di parità ottenendo la stringa $\underline{c} \in \{0, 1\}^n$ dove n corrisponde alla lunghezza della sottosequenza di dati non utilizzati per la stima dei parametri.

Successivamente \underline{c} viene utilizzata per modulare il segno della sequenza \mathbf{Y} ottenendo così un nuovo messaggio \underline{m} tale che $m_i = Y_i(-1)^{c_i}$ per $i = 1, 2, \dots, n$.

Il messaggio \underline{m} viene trasmesso ad Alice su canale classico pubblico che assumiamo non introduca errori. Alice estrae da \underline{m} un messaggio fittizio \underline{r} con $r_i = \frac{m_i}{\mathbf{X}_i}$ che corrisponde alla trasmissione su un canale Gaussiano con rumore di varianza $V_i = \frac{V_z}{|\mathbf{X}_i|}$.

Infine verrà utilizzato un algoritmo LDPC di decodifica per ottenere una stima $\hat{\underline{s}}$ della stringa \underline{s} prodotta da Bob.

2.2 Discussione della sicurezza contro attacchi di lettura del segnale

Questo protocollo è particolarmente sicuro perché consentente il rilevamento di un eventuale spia (Eve). Nel protocollo si assume che Eve sia in possesso di un computer quantistico e che abbia acces-

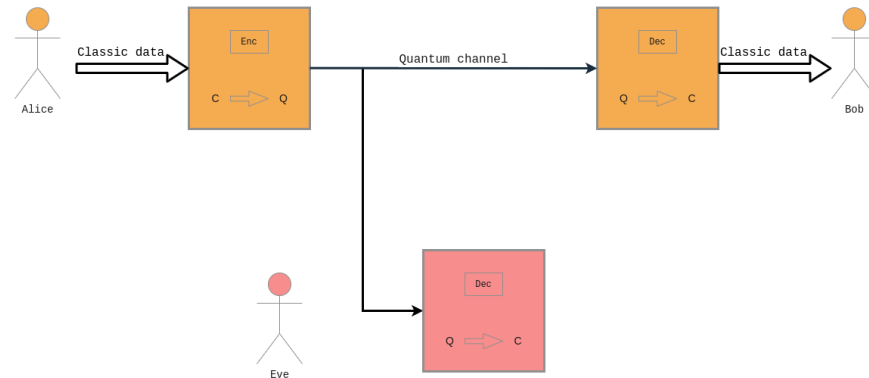


Figura 2.2: Schema di comunicazione tra Alice e Bob Sezione ??.

so completo al canale di comunicazione ed anche con queste assunzioni si è in grado di garantire la sicurezza per diversi tipi di attacchi.

Gli attacchi presi in considerazione tipicamente sono tre e differiscono per la loro efficacia:

- **attacchi individuali:** Eve effettua operazioni singolarmente su ogni segnale che viene scambiato tra Alice e Bob.
- **attacchi collettivi:** Eve salva nella memoria del suo computer quantistico un certo numero di stati sui quali effettua una misura collettiva. Questo attacco è più potente del precedente perché secondo i principi della fisica quantistica effettuando una misura collettiva si può potenzialmente ricavare più informazione che dalla misura del singolo stato.
- **attacchi coerenti:** sono attacchi più potenti dei collettivi perché Eve, oltre alla lettura collettiva, può utilizzare delle correlazioni quantistiche, cioè entanglement, per mettere in relazione gli stati di più run.

Tornando al motivo per cui effettivamente si riesce a rilevare Eve possiamo, anche questa volta, dare il merito ai principi della fisica quantistica. Questo perché in caso di misura da parte di Eve lo

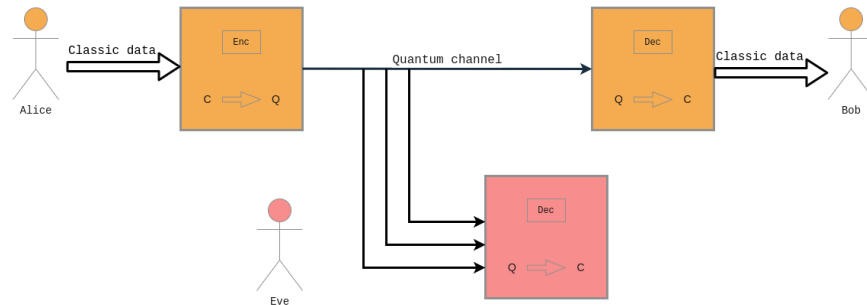


Figura 2.3: Schema di comunicazione tra Alice e Bob Sezione ??.

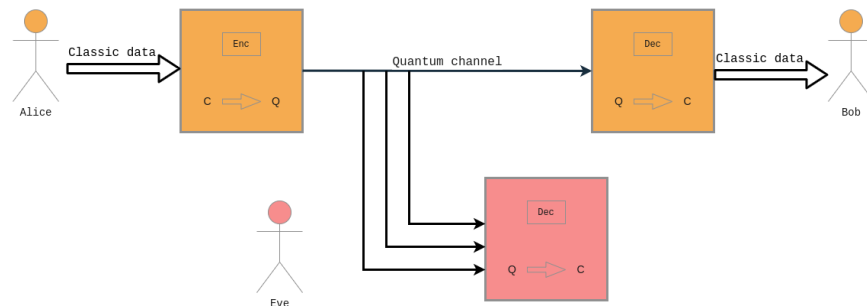


Figura 2.4: Schema di comunicazione tra Alice e Bob Sezione ??.

stato coerente trasmesso verrà alterato. Due esempi di attacchi che Eve può mettere in atto e le loro conseguenze sono:

- sottrarre una parte del segnale per effettuare una misura sullo stesso, questo comporta una diminuzione di fotoni del segnale ricevuto da Bob
- misura il segnale attraverso una misura eterodina per stimare entrambe le componenti di quadratura. Utilizzando le componenti misurate come valori medi, realizza un nuovo stato coerente che trasmette a Bob. In questo modo siccome vengono misurati dei valori randomici e siccome la misura eterodina stessa introduce ulteriore rumore la varianza degli stati coerenti ricevuti da Bob sarà superiore a quella originaria.

In entrambi i casi, un attacco da parte di Eve non va a far altro che aumentare gli effetti distruttivi del canale di trasmissione.

Tutte queste alterazione, in fase di stima dei parametri, produrranno dei valori indesiderati come $\chi > I_{AB}$. Queste incongruenze tra le stime e ciò che ci si aspettava dalla scelta dei parametri iniziali portano a pensare che una spia si sia intromessa nella comunicazione.

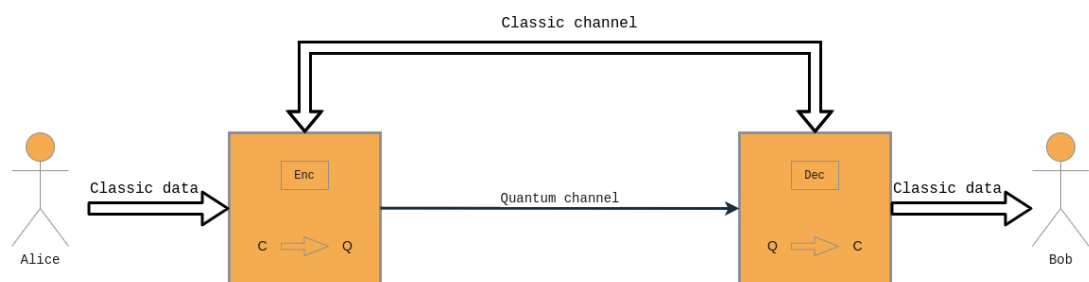


Figura 2.5: Schema di comunicazione tra Alice e Bob Sezione ??.

Capitolo 3

Simulazione

La realizzazione di questo simulatore per la distribuzione quantistica di chiavi con modulazione gaussiana di prefigge lo scopo di realizzare una chiave crittografica sicura, la quale successivamente potrà essere utilizzate per la codifica e decodifica di un messaggio trasmesso su un canale classico.

Come tecnologie è stato utilizzato il linguaggio di programmazione C++. I motivi principali di questa scelta sono tre: la velocità del linguaggio di programmazione, la disponibilità di librerie che implementano le distribuzioni di probabilità necessarie alla simulazione e la presenza di librerie esterne che implementano due algoritmi indispensabili per la correzione degli errori.

Anche se il progetto è stato realizzato utilizzando C++ è stato deciso di illustrare quanto è stato fatto attraverso dello psdeudo-codice in modo tale da non legare la simulazione ad una specifico linguaggio di programmazione ed agevolarne la rimplementazione in qualunque linguaggio di programmazione si voglia utilizzare, di fatto verrà utilizzato della pseudo-codice anche per descrivere il due algoritmi per la correzione degli errori.

3.1 Preparazione, trasmissione e misura

3.1.1 Preparazione dello stato coerente

Vengono scelti due valori che rappresentano i valori medi delle gaussiane che realizzano lo stato coerente. I valori vengono estratti a loro volta da una distribuzione di probabilità normale centrata in zero e con varianza V_{mod} che in questa specifica simulazione ha valore 5.226. Questo valore di varianza è stato scelto, insieme al valore della trasmittanza e quello del rumore, per raggiungere un rapporto segnale rumore pari a $SNR = 0.52$.

3.1.2 Trasmissione su canale rumoroso

Per ogni stato coerente estratto si simula la trasmissione in un canale rumoroso. In particolare i valori di rumore e trasmittanza scelti sono rispettivamente $\xi = 0.005$ e $T = 0.1$, scelti sempre per garantire un $SNR = 0.52$

3.1.3 Misura omodina del segnale

3.2 Stima dei parametri

3.2.1 Sifting

3.2.2 Stima delle varianze dei dati trasmessi, ricevuti e delle loro covarianza

3.2.3 Stima mutue informazioni

3.2.4 Accertamento sicurezza trasmissione

3.3 Riconciliazione

3.3.1 Generazione stringa random di bit

3.3.2 Algoritmo Progressive Edge Growth

Algorithm 1 : Classic PEG algorithm

pippo

Algorithm 2 : Sum-product decoding algorithm

pippo

3.3.3 Messaggio fittizio

3.3.4 Algoritmo Sum-Product

Algorithm 3 : Sum-product decoding algorithm

pippo

Conclusioni e sviluppi futuri

La tesi è finita

Bibliografia

- [BB14] Charles H. Bennett and Gilles Brassard. Quantum cryptography: Public key distribution and coin tossing. *Theoretical Computer Science*, 560:7–11, dec 2014.
- [DH76] W. Diffie and M. Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, 22(6):644–654, 1976.
- [LPF⁺18] Fabian Laudenbach, Christoph Pacher, Chi-Hang Fred Fung, Andreas Poppe, Momtchil Peev, Bernhard Schrenk, Michael Hentschel, Philip Walther, and Hannes Hübel. Continuous-variable quantum key distribution with gaussian modulation—the theory of practical implementations. *Advanced Quantum Technologies*, 1(1):1800011, 2018.
- [MFZG18] Mario Milicevic, Chen Feng, Lei M. Zhang, and P. Glenn Gulak. Key Reconciliation with Low-Density Parity-Check Codes for Long-Distance Quantum Cryptography. *npj Quantum Inf*, 4(1):21, April 2018. arXiv:1702.07740 [quant-ph].
- [RSA78] Ronald L Rivest, Adi Shamir, and Leonard Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2):120–126, 1978.
- [Sha49] C. E. Shannon. Communication theory of secrecy systems. *The Bell System Technical Journal*, 28(4):656–715, 1949.

- [Sho94] P.W. Shor. Algorithms for quantum computation: discrete logarithms and factoring. In *Proceedings 35th Annual Symposium on Foundations of Computer Science*, pages 124–134, 1994.
- [Zha18] Yusheng Zhao. Development of Quantum Key Distribution and Attacks against It. *J. Phys.: Conf. Ser.*, 1087:042028, September 2018.