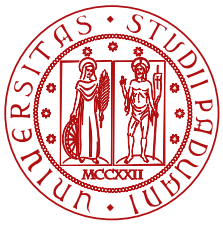


UNIVERSITÀ DEGLI STUDI DI PADOVA

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

CORSO DI LAUREA IN INGEGNERIA INFORMATICA

Characterization of single-photon detectors for Quantum Communications

Relatore
Prof. Vallone Giuseppe

Laureando
Conti Paolo

Correlatore
Bolanos Wagner Matias Ruben

ANNO ACCADEMICO 2023-2024

Data di laurea 27/09/2024

Abstract

The main goal of this thesis is the characterization of a set of single photon detectors (SPDs) by MPD, used in the field of Quantum Communications.
The analysis focuses on two aspects of the detectors: the jitter (electronical noise) and the tail of the detected pulse, related to the relaxation time of the detector. Moreover, we compare the performances of two versions of the same SPD model present in the *QuantumFuture* group at the University of Padua.
Data collection was performed with two setups. The first one, made to study the jitter, uses the SPDC effect - Spontaneous Parametric Down Conversion - to obtain two photons, which are detected by SPDs and then treated as coincidences; the analysis of these timestamps was performed with a small software developed as part of the thesis.
The second setup involves the application of a pulsed source to a single detector, and the resulting data is plotted properly in order to better show the tail of the distribution and compare that of the various PDMs.

L'obiettivo principale di questa tesi è la caratterizzazione di un set di rivelatori di singoli fotoni (SPD) prodotti da MPD, utilizzati nel campo della Comunicazione Quantistica.
L'analisi si concentra su due aspetti dei rivelatori: il jitter (rumore elettronico) e la coda dell'impulso rilevato, correlata al tempo di rilassamento del rivelatore. Inoltre, vengono confrontate le prestazioni di due versioni dello stesso modello di SPD presenti nel gruppo *QuantumFuture* dell'Università di Padova.
La raccolta dati è stata effettuata con due setup. Il primo, realizzato per studiare il jitter, utilizza l'effetto SPDC - Spontaneous Parametric Down Conversion - per ottenere due fotoni, che vengono rilevati dagli SPD e trattati come coincidenze; l'analisi di questi tempi è stata eseguita con un piccolo software sviluppato come parte della tesi.
Il secondo setup prevede l'applicazione di una sorgente a impulsi a un singolo rivelatore, e i dati risultanti vengono tracciati opportunamente per mostrare meglio la coda della distribuzione e confrontare quella dei vari PDM.

Contents

List of Figures

Chapter 1

Introduction to Photon Detection

1.1 Quantum Mechanics and Photons

Quantum mechanics is one of the most successful theories in modern physics. It provides a description of the interactions between atomic objects, introducing a *wave-particle duality*: the ”wave” portion of the theory refers to *wave functions*, representations of the probability that a particle exists at a location in a certain moment, while the ”particle” portion introduces *quanta* as the smallest possible unit of a phenomenon - therefore implying that particles can only have discrete values for energy or momentum.
When applied to light, quantum mechanics leads to the concept of *photons*.
A photon is defined as the smallest possible particle - quanta - of light, or electromagnetic energy in general; although the nature of light has been studied for centuries, the concept of photon was only established through Planck's equation, $E = hf$, and later confirmed by Einstein's research. In accordance to the theory, the behaviour of a photon can be either particle-like or wave-like. Photo-detectors, in order to register the arrival of a single one, treat them as particles; despite this, the issue with predicting when and where a photon will strike the detector is related to the probabilistic nature of events in quantum mechanics: instead of talking about a ”single photon”, it is better to refer to a ”single photon state”.

1.2 The importance of photons in QKD

Quantum Communications is one of the main research fields in which single photons are relevant. Given that photons can travel at the speed of light and can be manipulated through linear optics, they are an ideal solution to carry information by encoding it in one of its degrees of freedom, like polarization, energy, or angular momentum.
Quantum Key Distribution (QKD) is a particular field of development in which single photons -

thanks to their quantum nature and the laws of quantum mechanics - are used to create a theoretically secure shared key between two parties. QKD does not transmit messages, unlike classical protocols, but it allows traditional algorithms to use the greater security of a quantum key to share data: any attempt to intercept the key will be detectable, and the eavesdropper will get negligible to no information.

1.3 Single Photon Detectors

An *ideal* single photon detector is usually defined as one that:

- has a detection efficiency of 100%;
- has a null dark-count rate - meaning that it never generates an output without any incident photons;
- has a null jitter - as in, the variation of the delay between the optical and electrical signals;
- has a null dead time - the time during which photon detection is impossible after a measurement.

Furthermore, an ideal SPD has a certain degree of ”photon number resolution”, which allows it to distinguish between one and more photons in an incident pulse; in this description of the main types of detectors avalaible today, a first differentiation can be made between non-photon-number-resolving and photon-number-resolving SPDs:

- non-PNR: photomultiplier tubes; single-photon avalanche photodiodes; quantum-dot field-effect transistor-based detectors; superconducting nanowire detectors; up-conversion detectors.
- PNR: superconducting tunnel junction detectors; quantum-dot field-effect transistor detectors; superconducting nanowire detectors; superconducting transition edge sensors; visible light photon counters.

1.3.1 Photomultiplier tubes

Photomultiplier tubes, or PMTs (Fig. 1.1), rely on a photo-catode releasing an electron when being hit by a photon by photoelectric effect: said electron is then directed towards a chain of dinodes, causing an avalanche effect (Fig. 1.2) until the generated signal is strong enough to be detected by output electronics.

Figure 1.1: A PhotoMultiplier tube by ThorLabs

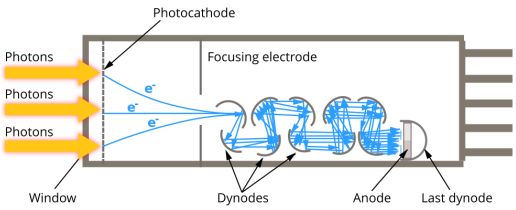


Figure 1.2: The electron cascade that forms in a PMT

The main downside of PMTs is its dependency on vacuum-state and vacuum-tube technology for all operations, which greatly reduces their lifetime and reliability. Despite this, these detectors have a decent efficiency, ranging from 10% to 40% depending on the wavelength of the photon and on the base material; they also support a larger detection area and have very low dark-counts levels and dead-time.

1.3.2 Single-photon Avalanche Photodiodes

Single-photon Avalanche Photodiodes, or SPADs (Fig 1.3), are based on the same principle as PMTs; the main difference is the absence of vacuum and the use of a diode in breakdown-state to achieve a better avalanche amplification of the first electron-hole pair, emitted after being hit by an incident photon.

Due to the presence of a diode under high tension - bigger than its breakdown threshold - SPADs usually have higher dead times compared to other detectors: this can be further explained by the existance of ”traps” in the propagation material, that tend to block electrons during the avalanche effect, only to release them during the dead time, creating a false photon detection event - known as *afterpulsing*.

SPADs have an overall better efficiency, compared to PMTs, reaching 85% when working with visible light; dark counts and jitters, instead, tend to be higher than average, but can be reduced by cooling the device.

A current development field for SPADs involves linear mode, as opposed to breakdown-state: this would allow the output to be proportional to the number of incident photons, hence making new detectors photon-number-resolving as well; despite the theoretical advantage, though, these new detectors would have much larger noise margins and the amplitude of the

Figure 1.3: A SPAD built by Excelitas

output pulse wouldn't be useful in actual measurements.

It is important to note that SPAD detectors - in particular, Excelitas SPADs as the one shown in figure 1.3 and PDMs (Photo Detection Modules) by MPD - are among those used in the data analysis part of this thesis.

1.3.3 Quantum-dot field-effect transistor-based detectors

This type of detector, often referred to as QDOGFET (Fig. 1.4), is based on *quantum dot* technology: the key property of these materials, also called semiconductor nanocrystals, is that an electron can be excited to a higher state of energy every time the base is hit by light - UV light in particular.

A QDOGFET is structured as follows: an optical absorber provides electrons whenever a photon impacts the detector; these electrons are then sent towards a FET, with a thin layer of quantum dots between the gate and the channel. The dots trap the carriers, thus changing the potential of the gate and the conductivity of the channel, and allowing a software to detect the impact of a photon.

1.3.4 Superconducting nanowire single-photon detectors

These detectors, known as SNSPDs (Fig 1.5), use the properties of *superconducting wires*, a particular class of conductors in which the resistance instantly drops to zero when brought below a temperature threshold, and a current can be maintained in time without any source; supercon-

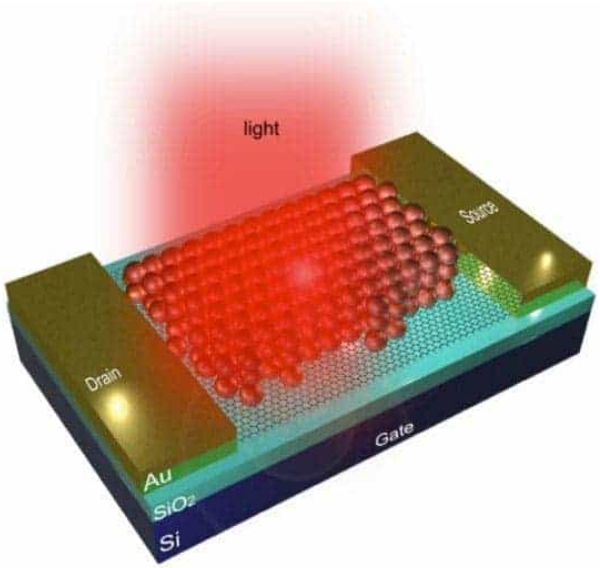


Figure 1.4: A simple working scheme of a QDOGFET

ducting wires also possess a critical current density, above which the wire's behaviour reverts back to a normal conductor.

This last property is at the base of an SNSPD: the detector maintains a superconductive wire just below its critical density, so that an incident photon will cause a small portion of the path to revert back to a normal state. The generation of this region with a finite, not null resistance causes a voltage peak, that can be read as the detection of an incident photon.

SNSPDs offer a great number of benefits, at the cost of significant expenses and operational challenges. They do not suffer from afterpulsing, despite locking in a self-heating state sometimes, and are the fastest single-photon detectors known today, with extremely low dark counts and jitters. The main downside is that, in order to achieve proper super-conductivity in the wire, the device must work at temperatures of around 4 K - or below, if possible.

1.3.5 Up-conversion single-photon detectors

The last case of a non-photon-number-resolving detector treated in this thesis is the Up-conversion SPD, which focuses on the detection of photons in the infrared (IR) frequency; the main principle is removing the challenges related to working with IR photons by applying a strong beam of light to the incident photon inside of a non-linear crystal, a crystal in which the polarization does not respond linearly to the energy of an incident light beam. The combination of these elements results in the emission of a photon in the visible frequency: The next step is simple detection using either SPADs or PMTs.

The efficiency of an up-conversion detector is close to 100% in the first part, involving the crystal, and it decreases to around 50% to 60% with the second half, depending on what

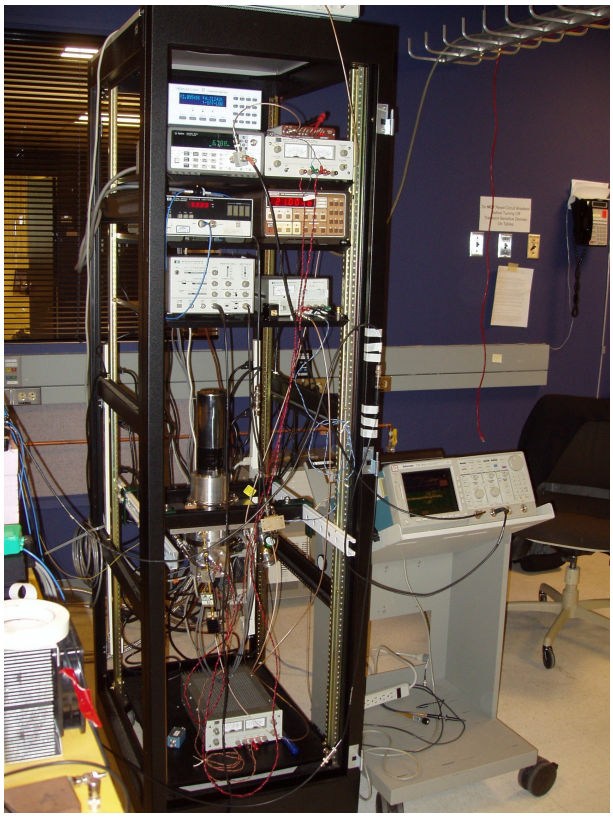


Figure 1.5: The complete setup of a commercial SNSPD

SPD is chosen for the final photon detection. The key motivation to work with these SPDs is the compromise between decent efficiency and low dark-count rates; the main downside is the creation of unwanted photons in the up-conversion process, which can be addressed by lowering the energy of the pump beam.

1.4 Photon-number-resolving detectors

We will now talk about the second main class of SPDs: photon-number-resolving detectors (PNR). An important note to be made is the definition of ”photon counting” that we will be considering: First of all, a PNR SPD does not necessarily have the ability to calculate the exact number of incident photons during an event. If we consider both the real efficiency and the presence of dark counts, the resulting photon count will be at best a good lower estimate. Furthermore, a distinction can be made between three categories of photon-number-resolving SPDs:

- *no-PNR* detectors;
- *some-PNR* detectors, which are a simple combination of multiple SPDs, that have no

photon counting resolution if taken individually;

- *full PNR* detectors, in which the output is actually proportional to the number of incident photons, no matter the efficiency and possible saturation levels.

A common, noteworthy approach to photon-number resolution is the use of *pixels*: the active detection area of an SPD is divided into multiple, independent parts, each capable of detecting an incident photon. This allows us to manage the dead time and jitters of a single part after every event, as well as being able to detect as many photons as the number of pixels available; the main issue with this approach is that, when using small numbers of pixels, subsequent photons may hit the same pixel, resulting in a single event being registered.

1.4.1 Superconducting tunnel junction detectors

Superconducting tunnel junction, or STJ (Fig. 1.6), detectors are among the first PNR SPDs created; they consist of a double film of superconducting metal, divided by an insulating layer - hence the *tunnel junction* name. When working at temperatures of around 1 K and with a small voltage across the junction - to allow the flow of particles created between the two films - the impact of a photon causes a perturbation strong enough to create a change in the electrical intensity across the tunnel.

Figure 1.6: One of the first developed STJ detectors

Single-photon detection in a STJ is made possible by keeping the device well below the critical temperature threshold of the superconductor: by doing this, the number particles generated by thermal phenomena is much lower than that of the particles generated after an impact event. Further more, since the current generated by an incident photon grows with a known proportion after multiple impacts, STJ detectors also possess photon-counting capabilities.

The efficiency of these devices has been shown to be around 50% with fairly high count rates - around 10 kHz; background-generated counts are also very low and mostly related to electronic noise.

1.4.2 Quantum-dot field-effect transistor detector

This type of SPD has already been discussed as part of the non-photon-number-resolving detectors; its key property is that the potential of the gate changes whenever a photon hits the optical absorver, causing it to release electrons which are then trapped by the quantum dots between the gate and the channel.

Given that the proportion between the change in conductivity and the energy released by a single photon is known, it is relatively easy to upgrade a QDOGFET from a *no PNR* to a *some PNR* state. The overall efficiency for these detectors is not very significant, being just around 2% at a wavelength of 805 nm, but it does work extremely well with small number of incident photons, reaching an accuracy close to 90% when distinguishing 1 to 3 photons events.

Further development in this field aims to reduce dark count rates - which are around 2000 cps for standard devices - by lowering the working temperature of QDOGFETs from 80 K to 4 K.

1.4.3 Superconducting nanowire single photon detector

Similarly to quantum-dot SPDs, superconducting nanowire detectors (SNSPDs) were developed as part of non-photon-number-resolving SPDs and only later upgraded to have some degree of photon counting resolution. The main principle, in synthesis, is observing when and where a section of a superconducting wire - kept close to its critical current density - reverts back to normal behaviour and shows a not null resistance, revealing the impact of photon. The result, which only allows for *some PNR* capabilities, was achieved by using more nanowires to fill the active area of the detector, instead of just one; this can be done in two different ways, both using nanowires electrically connected in parallel (Fig 1.7).

The first solution considers the sum of the parallel currents, and the number of incident photons is obtained by calculating how many wires reverted back to normal conditions. This allows parallel SNSPDs to maintain the extremely high speed of the single-photon version, be-

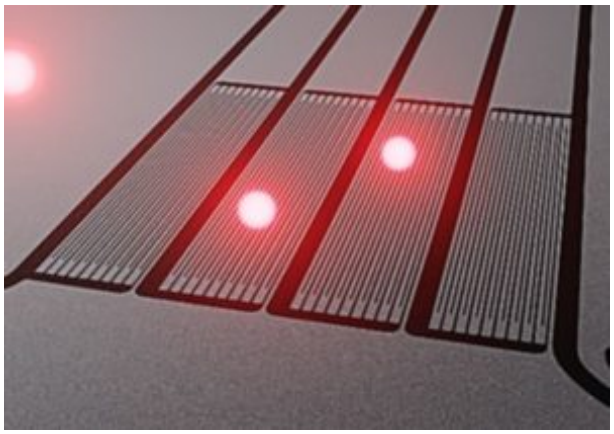


Figure 1.7: The scheme of a SSPD with parallel nanowires

cause having multiple wires instead of a single, longer one greatly reduces inductance - which is the limiting factor for the operational speed of this device. The photon-counting capability has been shown with up to four photons, while also maintaining very low dark counts and a good frequency (80 MHz); the only downside on the commercial side is the low efficiency (around 2%) of a parallel SNSPD.

The second variant does not sum the currents of the parallel wires, but instead keeps them separated and linked to individual detectors: the output is simply a digital combination of all single-photon-detection events. This system, also able to distinguish up to 4 photon events, showed a better efficiency than its alternative (up to 25%).

1.4.4 Superconducting transition edge sensor

A superconducting transition edge (TES) sensor operates in a similar way to other SPDs we've discussed in this chapter, such as QDOGFETs or SNSPDs. It is essentially a *bolometer*, that is a sensor for electromagnetic radiation, that detects absorbed energy as a rise in temperature and calculates the number of incident photons that brought said energy. Since the output is based on the energy released by a single photon, we can immediately see that this type of SPD does have PNR capabilities.

To obtain a combination of extreme heat sensitivity and a subsequent significant change in resistance, needed to get a notable output signal, the thermal sensor is made of a small film of superconductive material, kept at a temperature between its superconducting and normal states. By doing this, the small amount of heat released by a photon will cause a large increase in resistance. At this point, both the current flowing through the device and its Joule effect will be greatly reduced, signaling the impact of a photon; the accuracy of the detectors depends on the device used to measure the current change: the best solution has been shown to be quantum-interference devices (SQUIDs), the design of which is beyond the goals of this discussion.

STE sensors can be improved by reducing their heat capacity - that is, the relationship between energy supplied and change in temperature; this can be done either by fiber coupling the light source to the device, reducing the free space between the two down to $10\ \mu$m, and by operating at just a few K, which would be required anyway to keep the sensor in its intermediate state. In addition, some of the latest sensors also have self-aligning mechanisms for the fiber.

The main advantage of using STE detectors is that they can be adapted to different wavelength while keeping a high efficiency, for example both with IR and visible light photons. The main drawbacks are low counting frequencies (100 kHz) and slow response times (100 ns).

1.4.5 Visible light photon counter

Visible light photon counters, or VLPCs, are known for reaching the efficiency levels seen with TESs. The working principle is quite similar to that of a SPAD; an incident photon meets two layers, a *frontal* one and a *gain* one, and will liberate an electron from one of the two: said electron will then be accelerated by a set voltage and hit a sequence of layers, causing an avalanche effect that will release more energy, which is eventually detected as the output. The main difference between a TES and an SPAD, in addition to having photon-counting resolution, is the use of impurities instead of normal semiconductors.

The reason why the avalanche principle could not be simply applied to SPADs is the *multiplication noise*: single photons do not always release the same amount of energy when hit, and when dealing with an avalanche effect the difference can become significant. The advantage of a VLPC is that this noise is close to zero: given that only electrons are multiplied, while holes are ”left behind”, and that single carriers generate a smaller multiplication noise, VLPCs do not suffer from the same issue - note that, in a SPAD, both electrons and pairs are involved in the avalanche effect.

Because of the absence of noise, VLPCs show extremely high accuracy when detecting up to six photons (up to 88%). The dead time is also notably low, because a multiplicating electron only hits a small portion of the available surface, allowing the rest of it to wait for the effect of further photons.

The main downside of visible light photon counters are, again, low repetition rate (100 kHz) and high dark count rates (20 kHz).

An interesting development involving photon-number-resolving SPADs was the analysis of the slope of the avalanche rise before its saturation, which gives some information about the number of incident photons: this approach has been shown to work fairly well with up to four photon events.

1.5 Spontaneous Parametric Down Conversion

The final part of this chapter presents an optical phenomenon known as Spontaneous Parametric Down Conversion, or SPDC.

SPDC is a non-linear effect of second order in which an incident pump photon with energy E_p is converted into two photons, named signal and idler, with lower energies E_s and E_i respectively (Fig. 1.8). These two generated photons are known to be *entangled*, but that topic is beyond the scope of this thesis.

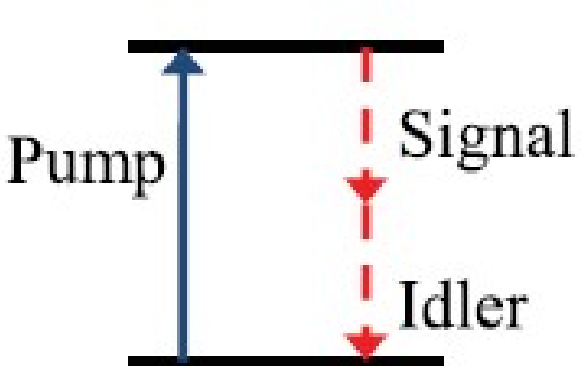


Figure 1.8: Energy correlation between the three photons

For SPDC to happen, the pump and generated photons must obey the *phase-matching* conditions, which are the conservation of energy and linear momentum, such that

$$E_p = E_s + E_i$$

$$\vec{p}_p = \vec{p}_s + \vec{p}_i$$

$$\lambda_p = \lambda_s = \lambda_i$$

where p, s, i refer to pump, signal and idler photons, while $E, \vec{p}, \lambda$ are respectively the energy, the momentum and the wavelength.
If we also consider $E = h \cdot \nu$ (Planck's equation) and $\vec{p} = h \cdot \vec{k}$, with $\vec{k}$ being the *wavenumber* of a photon - the reciprocal of the wavelength - we can write the previous equations in terms of frequency and wavenumber (Fig. 1.9), such that

$$\nu_p = \nu_s + \nu_i$$

$$\vec{k}_p = \vec{k}_s + \vec{k}_i$$

SPDC is an extremely weak process, meaning that only a small portion of incident photons will generate a couple of entagled particles (with a generation probability of around $1.6 \cdot 10^{-13}$).

Figure 1.9: Correlation between the photons, in terms of frequency and wavenumber

The main setup used to obtain SPDC involves a pump laser directed towards a non-linear material, often a crystal, as shown in fig. 1.10.

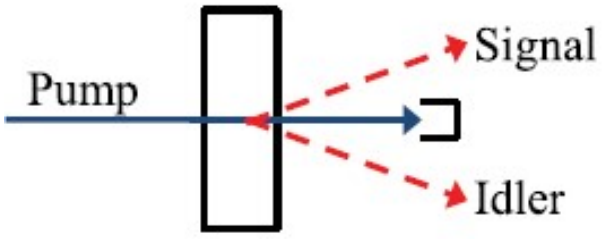


Figure 1.10: The scheme of a basic setup for SPDC

SPDC had an important role in the data analysis part of this thesis - a role that will be explained in Chapter 2; the reason for this is the extremely low uncertainty in the creation time of the generated pair of photons, which is in the order of femtoseconds. Therefore, including SPDC in the data collection setup allowed us to consider the two photons as ”created in the same instant” and made the uncertainty (or noise) related to the photon-generation neglibile.

Chapter 2

The jitter

2.1 What is the jitter

The focus of this chapter is the characterization of the new SPDs available at the University of Padua by calculating their *jitter*.

The jitter of a detector is defined as the random variation in the time difference between the impact of a photon on the active surface of the SPD - *impact event* - and the emission of the corresponding electric signal on the output side - *detection event*. Therefore, jitter provides an excellent measure of the reactivity of these new detectors, allowing for an idea of how convenient their application can be.

2.2 Setup and use of SPDC

In order to characterize the jitter of a photon detector, we chose to use an SPDC source - as explained in section 1.5 - to study two SPDs at the same time.

The common approach to this analysis would have been the use of a pulsed laser source to a single detector: unfortunately, this requires a source with a FWHM - *Full Width at Half Maximum* (fig. 2.1) - smaller than the jitter of the detector. This is because if the FWHM were bigger than the studied jitter, the uncertainty in the photon-detection time would be mostly caused by the source and not by the SPDs, removing the whole purpose of the analysis. Instead, if the FWHM is small enough, any change in the SPD's times can be attributed to the detector itself. The use of SPDC in the setup provides a solution to this problem. Instead of measuring a single channel distribution, we measured the distribution of the events detected *in coincidence*[1] between two detectors, corresponding to a pair of photons created by SPDC.

[1] When two events are detected within a given time window, known as coincidence window, they are said to be detected in coincidence.

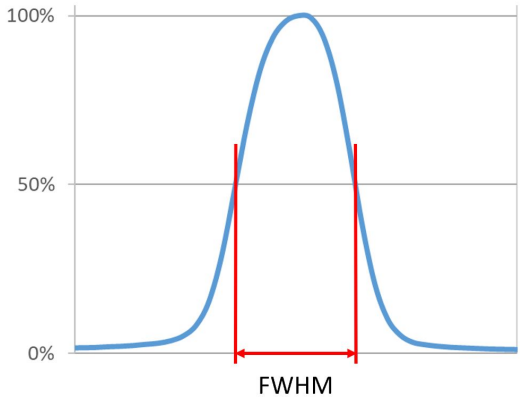


Figure 2.1: FWHM as a measure of the duration of the laser pulse

The definitive setup consisted of the following elements, as shown in fig. 2.2:

- A continuous wave laser source, with a wavelength of $404nm$ (purple light);
- A non-linear crystal, which was pumped by the laser beam and allowed SPDC to happen, resulting in the generation of the two entagled photons - each with double the initial wavelength, 808 nm (infra-red light) in this case;
- Two optic-fiber paths, to guide each of two photons towards the chosen detector and to reduce the need for free-space propagation;
- Two SPDs, chosen to get a complete system of equations in order to calculate the jitters. Those involved in this analys were two PMD-IR detectors by MPD, based on the principle of Single Photon Avalanche Detection;
- A quTAG - a time-tagger - to record the timestamps of photon-detection events for both SPDs.

2.3 Data, code and analysis

During the characterization of the jitter of the new MPDs, collected data was saved as *.txt* files, named following the pattern *TimeTags_SN..._SN...* to identify the two detectors paired in that set of measurements. *SN* stands for serial number, and the four detectors used in this study were *SN07*, *SN08*, *SN10* and *SN11*.

Every text file was composed with the same structure:

- a header (fig. 2.3) with basic information about the set of timestamps recorded, such as date and time, the unit of measure used, and the format needed to extract the data for further use.

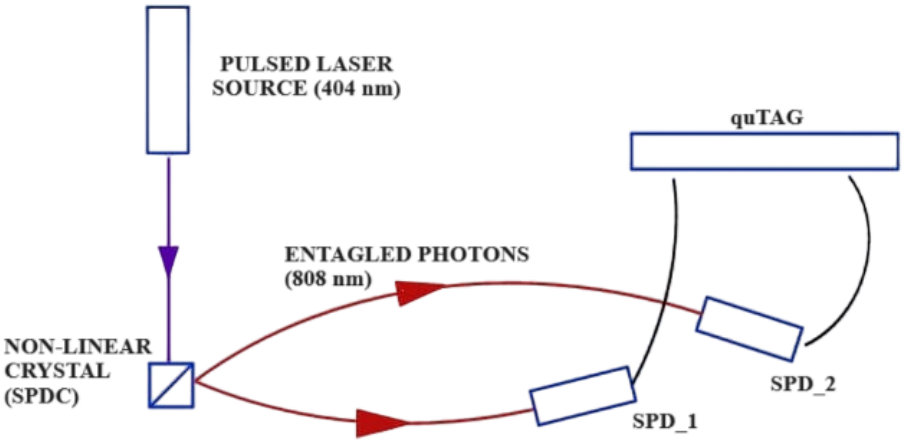


Figure 2.2: A scheme of the setup using SPDC

- the list of photon-detection timestamps obtained from the electrical signal generated in the MPD (fig. 2.4); the format for this part - already defined in the header - is very simple: the first half provides the detection time in picoseconds, while the second half indicates in which channel it was detected. The value of the channel can be either 1 or 2, indicating which of the two detectors detected a photon.

```
# Time Tags
# 2024-07-11 10:56:11
# Unit Time Tag: 1.000ps
# Unit Channel: 1
# Time Tag ; Channel
```

Figure 2.3: The header of the file TimeTags_SN07_SN08

The first step of the analysis was to extract the timestamps from the .txt file and to divide them between the two channels; this was achieved simply by skipping the first lines, which contain the header, and by looping through the following ones, reading the first part - before the ”;” - as the timestamp and the second as the detector that recorded it (Algorithm 1).

The next step was to remove the offset from one of the detectors (algorithm 2). This offset is present because each detector contains different path lengths, both optically and electrically (usually negligible, as they are in the order of nanoseconds), which become relevant when the coincidence window is set to units of nanoseconds or smaller.

The only way - or the most practical in this case - to determine the correct offset was to initially run the whole code without any correction, check the distance between the peak of the

```
2364126012196938 ;                    2
2364126021437034 ;                    1
2364126112611926 ;                    1
2364126140350506 ;                    2
2364126148228893 ;                    2
2364126172625952 ;                    1
2364126231137874 ;                    2
2364126272472034 ;                    1
2364126290466775 ;                    1
2364126304520352 ;                    2
2364126314788839 ;                    2
```

Figure 2.4: The first timestamps of the file TimeTags_SN07_SN08

Algorithm 1: Saving each timestamp to the array of its channel and counting them

```
Open source_path as file;
Read all lines into lines;
foreach line after line 5 do
    Split line into two parts using ";" as separator;
    if number of parts < 2 then
        Skip (corrupted line);
    else
        Assign timestamp as first part, removing spaces;
        Assign channel as second part, removing spaces;
        if channel is 1 then
            Append timestamp to timestamps_channel_1;
        else
            Append timestamp to timestamps_channel_2;
Set len_1 to the length of timestamps_channel_1;
Set len_2 to the length of timestamps_channel_2;
```

distribution peak - explained in the following paragraphs - and the origin of the x axis, and then remove said distance from a channel among the two. The choice is not random, as decreasing the correct set of timestamps will remove mentioned distance and doing it with the wrong one will instead double it; therefore, offset correction requires - worst case - to run the base code once, read the plotted graph and try the solution on both channels.

Having fixed the offset, SPDC could be applied to the characterization of the jitters. In order to do this, we had to determine *coincidences* between the two channels: a coincidence is defined as a simultaneous photon-detection event in both SPDs - in practice, the theoretical detection of two correlated photons generated by the down-conversion.

The first step in this process is the choice of a *coincidence window*, that is the maximum time difference between two events that makes them count as a coincidence. For this analysis,

Algorithm 2: Pseudocode: offset correction on channel 1

```
Set offset to the distance between the original Gaussian fit and the origin of the x axis
  (in ps);
foreach timestamp in timestamps_channel_1 do
    Set timestamp to timestamp - offset;
```

we chose a window of 1000 ps.

Then, the mapping of all simultaneous events between the two channels was performed with the *Two Pointer technique*. The technique uses the following steps - used as shown in algorithm 3:

- two pointers are initiated, one at the beginning of each array;
- compute the time difference between the pointed elements in *timestamps_channel_1* and *timestamps_channel_2*;
- if the difference is smaller than the coincidence window, store it;
- in any case, increase the pointer that refers to the smaller timestamp of the two.

Algorithm 3: Pseudocode: Two Pointer technique for coincidences detection

```
Initiate an empty array time_diffs;
Set COINCIDENCE_WINDOW to 1000 (ps);
Set the indexes i, j to 0;
while i, j are inside of the arrays do
    Set time_1 to the timestamp in timestamps_channel_1;
    Set time_2 to the timestamp in timestamps_channel_2;
    Set time_diff to time_1 - time_2;
    if Absolute value of time_diff < COINCIDENCE_WINDOW then
        Append time_diff to time_diffs;
        if time_1 < time_2 then
            Increase i by 1;
        else
            Increase j by 1;
    else
        if time_1 < time_2 then
            Increase i by 1;
        else
            Increase j by 1;
```

For the Two Pointer technique to work, the first assumption required is that the two arrays are sorted in ascending order: this is obviously true, since timestamps cannot be recorded in decreasing order in any case. Therefore, advancing a pointer means that the new pointed number will be greater than the previous.

The advancement logic applied to the pointers i and j (related to channels 1 and 2 respectively) uses the idea of always finding a larger timestamp from the smaller one, reducing the current time difference:

- if $|time\_1 - time\_2| < COINCIDENCE\_WINDOW$ and the time difference is small enough to count as a coincidence, advancing the pointer of the smaller timestamp gives us a chance to find another value closer to the current timestamp in the other channel - and therefore another coincidence;

- if, instead, $|time\_1 - time\_2| > COINCIDENCE\_WINDOW$, we want to find a new timestamp that reduces the time difference and brings it inside the time window; by increasing the pointer of the smaller one, the next comparison will involve a larger timestamp, potentially closer to the current one in the other channel.

Finally, the Two Pointer technique does not skip any coincidences, allowing for a complete analysis of the dataset: by always advancing the pointer of the smaller timestamp, we ensure that the algorithm is checking all possible pairs that could give a time difference inside the coincidence window. Further more, using two sorted arrays and systematically advancing the pointers implies that no acceptable pairs of timestamps will be skipped.

One last important consideration about the Two Pointer technique is regarding its complexity. The easiest solution to this analysis - in terms of code length - would have been a single *for* loop, as shown in algorithm 4, but the complexity of this solution is $O(n \cdot m)$, with n, m being the lengths of the two arrays of timestamps, due to the observation of every possible pair of elements. The technique used in this case, instead, only loops through the two arrays once, therefore giving the algorithm a complexity of $O(n + m)$.

Algorithm 4: Pseudocode: a standard ”for” loop to find coincidences

```
Initiate an empty array time_diffs;
Set COINCIDENCE_WINDOW to 1000 (ps);
foreach timestamp in timestamps_channel_1 do
    foreach timestamp in timestamps_channel_2 do
        if Absolute value of time_diff < COINCIDENCE_WINDOW then
            Append time_diff to time_diffs;
```

At this point of the analysis, the array *time_diffs* contained all the correlated photons recorded by the two SPDs; assuming that these events follow a normal distribution, we were able to calculate the jitter of every set of measurements by fitting a Gaussian curve to the histogram of the coincidences.

Regarding the mathematical definition of jitter, during this analysis, we will be using the following formula valid for Gaussian distributions:

$$jitter(FWHM) = 2\sqrt{2\ln 2}\sigma = 2.335 \cdot \sigma$$

where σ refers to the standard deviation. The reason behind this formula is that, when working with a normal distribution of values - as in this case - observing how wide the Gaussian curve is can provide an accurate enough idea of the speed of the detector. In particular, this is an optimal solution when achieving ideal working conditions isn't possible and extreme precision in the measurement of the jitter isn't required.

We then plotted *time_diffs* in a histogram - with bin size equal to the range of time differences found - and normalized it: this gave us a probability distribution instead of a count distribution, allowing for a more homogeneous analysis of different datasets with completely different number of events. After plotting the time differences in a histogram, we had to find the parameters of the Gaussian function that best represented the normal distribution of those probabilities: in particular, finding the standard deviation of that curve would have immediately given us the jitter of that measurement, as explained in the following paragraphs.

In order to do this, we defined a Gaussian function inside of the code, as

$$\phi(x) = a \cdot e^{-\frac{(x-\mu)^2}{2\cdot\sigma^2}}$$

with a being the amplitude, μ the mean and σ the standard deviation. We then used the *curve_fit* function - defined in the *scipy.optimize library* - to recursively calculate the optimal parameters that best approximated the real behavior of the found coincidences. The only requirement for this step to work was the definition of a *starting guess*, that is a set of initial parameters to start the recursion with. The chosen guess, obtained with the *mean* and *std* packages in Python's *numpy* library to maximize precision, was:

- the maximum value in the histogram as the amplitude a;
- *np.mean(time_diffs)* as the mean μ:
- *np.std(time_diffs)* as the standard deviation σ.

Finally, the jitter was calculated as previously explained, with

$$jitter = 2.335 \cdot \sigma$$

and the histogram and Gaussian curve were plotted together. The pseudocode for this last part is shown in algorithm 5.

Algorithm 5: Pseudocode: Gaussian fit to find standard deviation and jitter

```
Define gaussian as the Gaussian function to use in the recursion;
Define params_guess as the initial set of hypothetical values;
Run the recursive algorithm with curve_fit(gaussian, bin_centers, hist,
 params_guess;
Save in popt, pcov the optimal parameters and their variance returned by the
 recursion;
Extract amplitude, mean, std_dev from popt;
Calculate jitter_measurement as 2.335·std_dev;
Plot the histogram and the Gaussian fit;
```

The results are illustrated in Figs. 2.5a, 2.5b, 2.5c and 2.5d.

As shown in the previous figures, the resulting jitters for the four sets of measurements were:

$$jitter_{07,08} = 141.96\ ps$$

$$jitter_{08,11} = 147.24\ ps$$

$$jitter_{10,08} = 139.53\ ps$$

$$jitter_{11,07} = 141.01\ ps$$

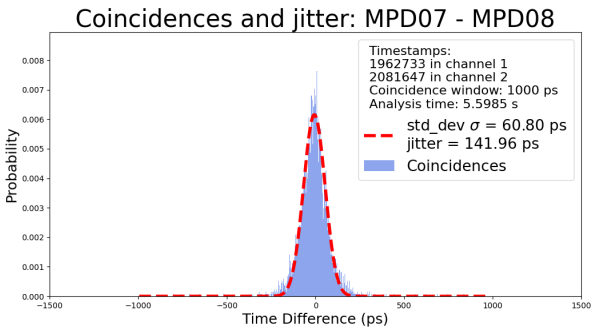
The individual jitters of the elements forming the setup add quadratically, following the equation:

$$jitter^2_{measurement} = jitter^2_{SPDC} + jitter^2_{SPD-1} + jitter^2_{SPD-2} + 2 \cdot jitter^2_{quTAG}$$

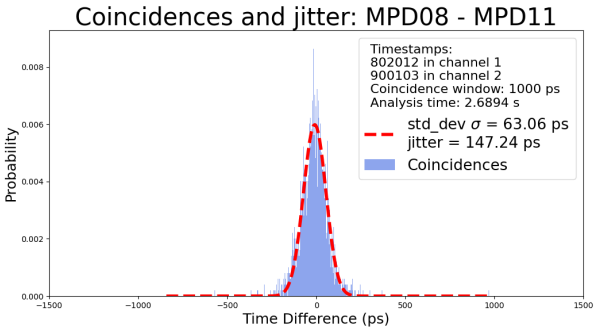
In this equation, $jitter_{SPDC}$ is in the order of femtoseconds ($10^{-15}s$) and can be approximated to 0 when compared to the other jitters - which are in picoseconds ($10^{-12}s$). This is because the creation time of the two correlated photons is extremely faster than the detection process inside the SPDs, and therefore does not influence the measurements.

The jitter $jitter_{quTAG}$, which has to be added twice to the equation because of the double *quTAG - SPD* connection build in the setup, is given by the company that builds the device:

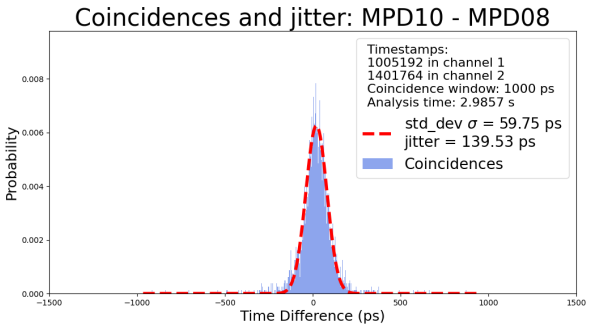
$$jitter_{quTAG} = 7.1\ ps$$



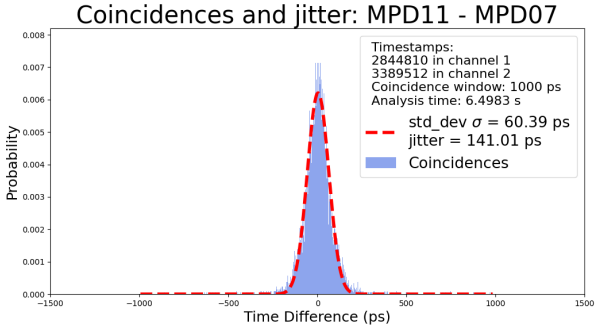
(a) SPDs 07 and 08



(b) SPDs 08 and 11



(c) SPDs 10 and 08



(d) SPDs 11 and 07

Figure 2.5: Comparison of measured jitter for different SPD pairs

We then obtain the following system of four equations with four variables:

$$jitter^2_{07,08} = jitter^2_{07} + jitter^2_{08} + 2 \cdot jitter^2_{quTAG}$$

$$jitter^2_{08,11} = jitter^2_{08} + jitter^2_{11} + 2 \cdot jitter^2_{quTAG}$$

$$jitter^2_{10,08} = jitter^2_{10} + jitter^2_{08} + 2 \cdot jitter^2_{quTAG}$$

$$jitter^2_{11,07} = jitter^2_{11} + jitter^2_{07} + 2 \cdot jitter^2_{quTAG}$$

which can be solved using the *fsolve* function to recursively find the best-fitting set of solutions (algorithm 6).

The conclusion of this analysis was that the jitters of the four single-photon detectors are:

$$jitter_{MPD07} = 95.54\ ps \qquad jitter_{MPD08} = 104.52\ ps$$

$$jitter_{MPD10} = 91.89\ ps \qquad jitter_{MPD11} = 103.22\ ps$$

Algorithm 6: Pseudocode: System resolution with recursive function fsolve

Define `equations_system` according to the four equations found and the respective variables;

Set `jitter_quTAG` to 7.1;

Set `jitter_0708, jitter_0811, jitter_1008, jitter_1107` to the correct values;

Set `initial_guess` to [0, 0, 0, 0];

Save as `x_07, x_08, x_10, x_11` the four solutions returned by `fsolve(equations_system, initial_guess)`;

Chapter 3

The tail

3.1 Tail of a Gaussian distribution

This last chapter of the thesis explains the study of the *tail*, or *peripheral events*, related to the photon-detection events reported by the SPDs involved in the analysis.

As explained in the previous chapter, both the time differences found with SPDC coincidences and the individual detection timestamps follow a Gaussian distribution: said distribution is characterized by a central peak - the mean - and two tails that extend on both sides. The tails are particularly relevant when studying the jitter - or, equivalently, the reaction speed - of a detector, as they represent less probable events, in which the delay between the impact of the photon and the emission of the output signal significantly deviates from the expected detection time.

Therefore, an ideal single photon detector would be expected to have extremely small tails: the normal distribution of its detection events would be concentrated around the peak, and the probability of having significant delay would be around zero.

The following sections explain the two methods used to measure the *skewedness* of the Gaussian distributions plotted from the data, followed by a comparison between the latest - studied in Chapter 2 - and the previous versions of the same SPD model used at the University of Padua.

3.2 Kurtosis

The first method used to study the un-evenness of the found distributions was computing their *Kurtosis*, a statistical measure that studies the shape of distribution, focusing in particular on the relation between the peak and the tails. Kurtosis can give us a good esteem of the *heaviness* of the tail, showing the probability of having time differences with values that differ a lot from the expected mean. Further more, this measure also gives information about the peak:

- a distribution with low Kurtosis (Platykurtic) will have a less pronounced peak; the data is more evenly distributed around the mean, and the outliers are less frequent than the standard case;

- a distribution with high Kurtosis (Leptokuric), instead, will present a sharper - taller - peak: this implies that the majority of values will be much closer to the mean, but at the same time outliers will occur with greater probability than in a normal curve.

This simple distinction already gives us a first idea of what to expect from this analysis, regarding the MPDs studied in Chapter 2: the results shown in fig. 3.1 can be clearly associated with the second situation, as they show a defined peak around the mean.

It is important to note that the "standard" case, when measuring the Kurtosis of a distribution, is a proper Gaussian one: in this situation, the reference value is set to 3.

In this analysis we calculated the *Excess Kurtosis* of our distributions, which simply represents how much the tailedness of a curve deviates from that of a standard Gaussian; regarding the mathematical expression, we considered the following formula:

$$Excess\ Kurtosis = \frac{n(n+1)}{(n-1)(n-2)(n-3)} \cdot \sum_{i=1}^{n} \left(\frac{x_i - \bar{x}}{s} \right)^4 - \frac{3(n-1)^2}{(n-2)(n-3)}$$

with

- n being the size of the data collection;

- x_i the i^{th} value;

- $\bar{x}$ the mean;

- s the standard deviation.

A full demonstration of this formula is beyond the scope of this thesis, but the reasoning behind it is that the first addend represents the *fourth moment*[1] of the distribution, which contains information on the heaviness of the tails and the relevance of the peak, while the second addend accounts for the error when applying the formula to a reduced sample of data instead of a complete distribution - the second term behaves so that, if computed on a proper Gaussian, it drops the excess Kurtosis to zero.

The application of this expression in our analysis was very simple: the code shown in Algorithm 7 was added after the study of the jitter, in order to compute the tailedness of the coincidences' distributions.

[1] The four moments are, in order: mean, variance, skewdness, Kurtosis

Algorithm 7: Pseudocode: computing the Kurtosis of the MPDs' distributions

Define a `kurtosis` function with arguments `data_set, mean, std_dev`;
foreach *value in the dataset* **do**
    compute the sum in the first addend
Adjust the result by including the first coefficient and the second addend;
Run the function as `kurtosis(time_diffs, mean, std_dev` for every set of time differences;

3.3 Residuals

The second part of the study of the tail considered the *residuals*, defined as

$$Residual = Observed\ value - Expected\ value$$

This provides useful information by comparing the fitted Gaussian - Chapter 2 - to the concrete data distribution; this can be connected to the Kurtosis measures as a complementary analysis: if the values found show the presence of a defined peak and consistent tails, we then use the residuals to show if and how the SPD's timestamps vary from an ideal Gaussian distribution.

The expected result - in particular with the new MPDs - would be to have both the presence of a tail and low residual values, indicating that the detectors do not suffer from delayed detection with high probabilities.

3.4 MPD results

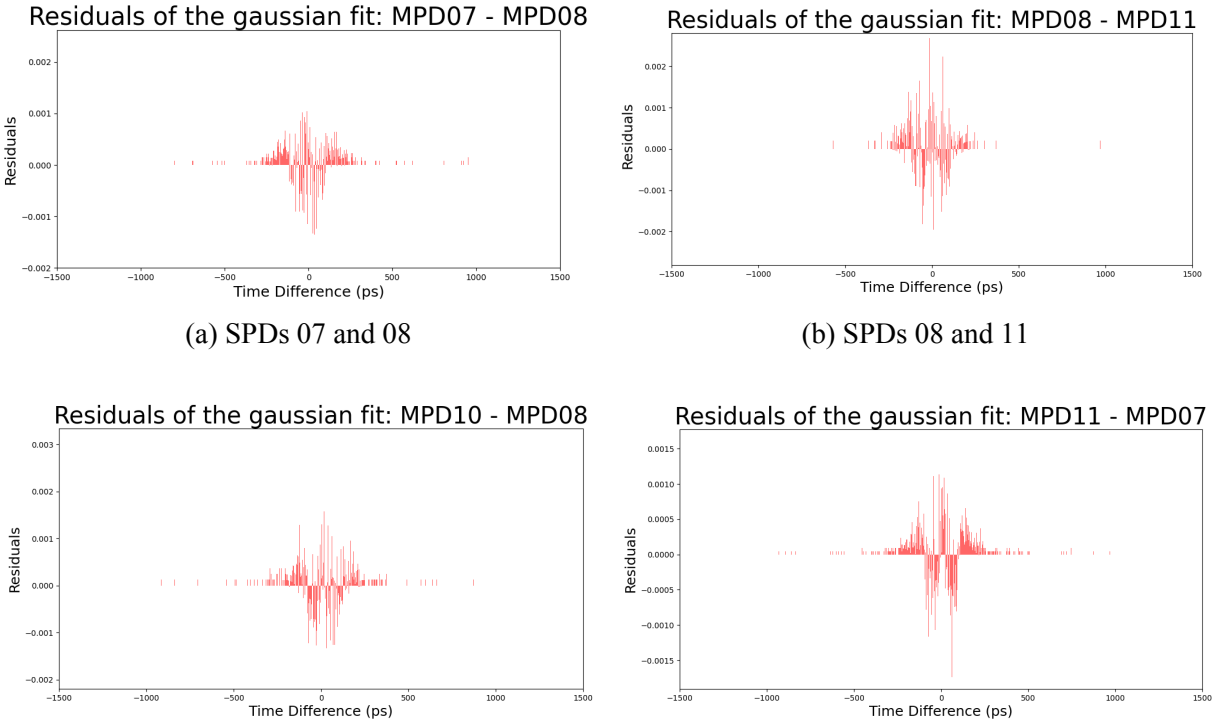
Regarding the new MPDs, the calculation of the Kurtosis value and the plotting of the residuals on the coincidences data (Fig. 3.1) gave the following results:

$$Kurtosis_{MPD07-08} = 97.8810$$

$$Kurtosis_{MPD08-11} = 62.0918$$

$$Kurtosis_{MPD10-08} = 104.3681$$

$$Kurtosis_{MPD11-07} = 117.2173$$



(a) SPDs 07 and 08

(b) SPDs 08 and 11

(c) SPDs 10 and 08

(d) SPDs 11 and 07

Figure 3.1: Residuals for the four coincidence datasets

3.5 Old MPDs results

The speed-related measurements shown in this chapter were also applied to the old MPDs previously used at the University of Padua, as shown in fig. 1.3, to give an idea of the improvements in efficiency provided by the purchase of the new devices.

The starting data-set was obtained in the same format used in the previous analysis (fig. 2.3 and 2.4), but with a single active photon detector and without any application of SPDC and coincidences. The following data manipulation, with the plot of a histogram and the fit of a Gaussian function, followed the same steps, with the only difference being a *modulo pulse period* operation on the timestamps: this was necessary to visualize the delay between photon emission and photon detection when working with a pulsed source.

The resulting graph is in fig. 3.2: the jitter of the old detectors resulted being:

$$jitter_{OLD\_SPD} = 150.32\ ps$$

This analysis resulted in an Excess Kurtosis value of

$$Kurtosis_{OLD\_SPD} = 3.86 \cdot 10^6$$

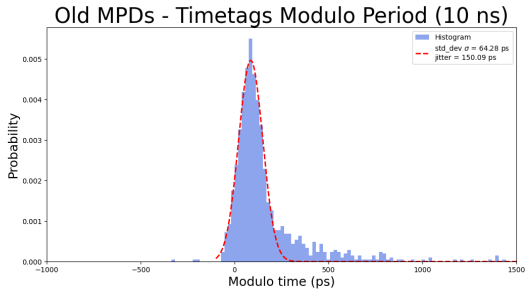


Figure 3.2: Old MPDs: histogram and Gaussian fit

and the Gaussian residuals are shown in fig. 3.3.

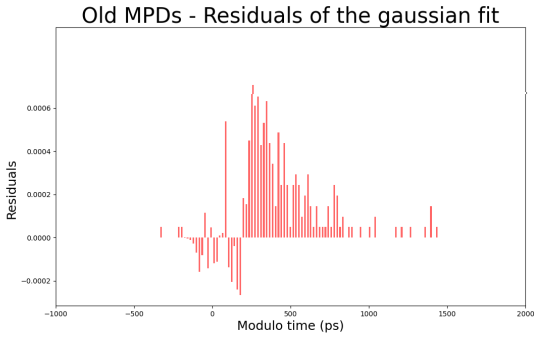


Figure 3.3: Old MPDs: Gaussian residuals

3.6 Detectors comparison

Having studied three key characteristics of the two types of Single Photon Detectors involved in this thesis, we can now compare the results:

- the average jitter of the new MPDs is significantly lower - around 33% less - than that of the old detectors:

$$jitter_{MPD} = 98.79\ ps \quad < \quad jitter_{Old\_MPD} = 150.32\ ps$$

  showing that MPDs have a faster reaction time when hit by a photon;

- the Excess Kurtosis index calculated from the distributions also confirms this result, since we have that:

$$Kurtosis_{MPD} = 95.39 \quad << \quad Kurtosis_{Old\_MPD} = 3.86 \cdot 10^6$$

This implies that detection events recorded by older detectors tend to create pseudo-Gaussian distributions with a heavier tail than those generated by the MPDs;

- finally, residuals plotting gives a more intuitive and visual proof of this conclusion: the four plots generated by the analysis of paired MPDs clearly present lower differences between the real values and those of the Gaussian fit - in particular when moving along the x-axis - compared to the plot based on the old data, in which the residuals form a long tail along the positive direction of said axis.

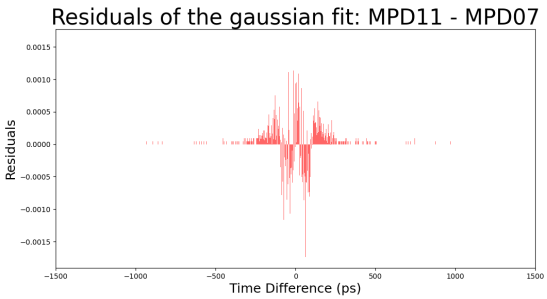


Figure 3.4: New SPDs residuals

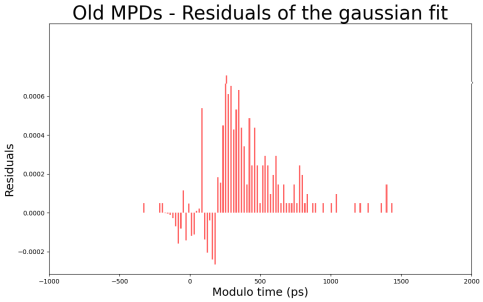


Figure 3.5: Old SPDs residuals

Chapter 4

Conclusion

In this thesis we characterized two different versions of Single Photon Detectors (SPDs) used by the *QuantumFuture* group at the University of Padua: the analysis focused on the jitter of the devices, the tail of the resulting timestamps distribution and the residuals of the comparison to a Gaussian fit.

The study saw a direct application of Spontaneous Parametric Down Conversion (SPDC) in the first part, and the use of a pulsed laser source during the second.

Our analysis showed that new detectors built by MPD present overall better performance when compared to the ones previously used: the jitter is significantly lower, meaning that the delay between photon-impact and photon-detection varies less than before, resulting in improved accuracy in the collected timestamps.

The analysis of Gaussian tails and residuals also confirmed this conclusion, showing that peripheral events tend to appear with lower probabilities and without an important detection delay.

In conclusion, this thesis showed that the new version of SPDs provides better accuracy and efficiency - and should therefore be the favourite choice between the two types of detectors - while also leaving space for improvement; despite having lower performance, the old SPDs can still provide acceptable results when having ideal working conditions is not possible or a slightly higher margin of error can be accepted.