PPr801 Project
Report
on

Superconducting Tunnel Junctions

Submitted by
P181211 **Gaurav Agarwal**

Under the guidance of,

Prof. Sangita Bose,
UM-DAE-CBS, Mumbai

and

Prof. Pratap Raychaudhuri,
TIFR, Mumbai

Department of Physical Sciences
Centre for Excellence in Basic Sciences
Mumbai, India - 400098

Semester 8, 2022

Acknowledgement

I wholeheartedly thank Prof. Sangita Bose and Prof. Pratap Raychaudhuri for mentoring me. They introduced me to the topic, allowed me to work in their labs and provided me with a lot of guidance for undertaking this project. I thank Prof. Sangita Bose for her patience with mistakes and reassurance which were showered on me in the project's entirety.
I thank Chandan Gupta for assisting me with the equipment. I thank Pritam Das for both assisting me with equipment as well as taking readings in magnetic fields which I was not able to do within time. I would also like to thank John Jesudasan sir and Vivas Bagwe sir for making me feel included at TIFR.

Gaurav Agarwal

May, 2022
UM-DAE-CBS, Mumbai

Abstract

One of the pioneering methods of confirming the band gap predicted by the BCS theory was using superconducting tunnel junctions. We outline the low-temperature techniques used and measure electron tunneling in normal(Ag) - superconductor(NbN) tunnel junctions. We confirm the predictions of the BCS theory. We demonstrate phenomenological factors that are intricately linked to the physics of superconductors. Further, we measure these tunnel junctions in magnetic fields and present Maki analysis on the data collected. Python programs are also developed for automatic data analysis and parameter extraction.

Contents

Chapter 1

Introduction

Superconductivity is a phenomenon that was discovered in Mercury by Kammerlingh Onnes in 1911. On cooling Mercury below 4 Kelvin using liquefied Helium it was found that the resistance of mercury dropped abruptly to an 'immeasurable' quantity. This phenomenon is named superconductivity, and the temperature at which it happens is termed critical temperature. Superconducting materials also expel magnetic fields akin to a diamagnet. These properties are onset by a phase transition in the material where the conjugate field can be temperature and magnetic field. Kammerlingh Onnes won the Nobel prize for this discovery in 1913. Thereafter, many metals like Lead, Niobium and alloys like Niobium Nitride, Strontium based compounds were found to be superconductive.

Theoretically, superconductivity was first classically explained by London in 1935 which explained the Meissner effect and penetration depth in superconductors. However, much later in 1950 a phenomenological framework - Ginzberg Landau theory - combined second-order phase transitions and was able to explain superconductivity macroscopically. Abrikosov showed that the theory also predicted type-II superconductors. Landau and Abrikosov won the Nobel prize in 2003 for this. A microscopic theory of superconductivity was suggested by Bardeen, Cooper and Schrieffer in 1957 which explained superconductors almost completely at the time. The theory won the Nobel prize in 1972.

Currently, the field is active in finding superconductors with high-temperature superconductors (such as Cuprates and Graphene at magic angle), studying mixed vortex states in type II superconductors, unconventional superconductivity etc. Theoretical models for high-temperature superconductivity are still not on solid footing and are an active field of research.

1.1 Basic Characteristics

1.1.1 Zero Resistance

The electrical resistivity of all metals arises due to the scattering of free electrons (plane waves) from phonons and defects in a crystal lattice [1]. On decreasing the temperature, the crystal lattice admits more order, resulting in less scattering. This

[1] In a perfect crystal, electrons will not experience any resistance other than phonon scattering.

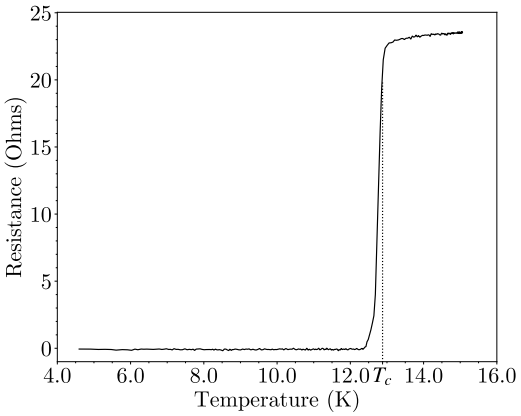


Figure 1.1: The resistance of a superconductor dropping to zero at the critical temperature (T_c).

leads to the lower electrical resistance of the material. In a pure crystal, one expects zero resistance at absolute zero temperature.

Superconducting materials, however, show an abrupt fall to zero resistance below a certain temperature. This temperature is called critical temperature. The phenomenon of zero resistance was verified by taking a superconductive ring with a flowing current. The decay of the current is proportional to $e^{-(R/L)t}$, and was found to be vanishingly small even after long periods of time (years)[4]. This can only imply that the resistance indeed is zero. An application of the zero resistance effect is that Lenz's law is perfectly obeyed, so as to not allow any change in the magnetic flux once the ring is superconductive. Hence, the net internal flux remains zero and can be used to shield equipment from changing magnetic fields.

A 'two-fluid' model of the electrons was suggested by thermodynamic arguments and states that the charge carriers in a superconductor are of two types - 'super-electrons' and electrons. It is assumed that super-electrons face no scattering and effectively short the normal electrons from carrying any current. A slight justification of this hypothesis was seen by applying AC voltages across superconducting samples and observing some inductive impedance. The super-electrons have some inertia that explains the impedance.

1.1.2 Perfect Diamagnetism

When a sample is cooled below its transition temperature, it loses all its resistance. This implies that the magnetic flux enclosed within an imaginary closed loop in the superconductor cannot change. Hence, the magnetic flux at any point within the sample cannot change. This is to say that the application of magnetic fields to a superconducting sample would lead to the generation of electrical current loops according to Lenz's law effectively nullifying the magnetic field inside the sample. These currents arise on the surface of the superconductor and are known as 'screening currents'. This is how a superconductor exhibits perfect diamagnetism [4].

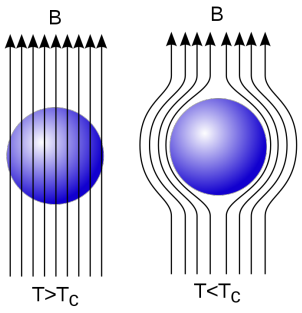


Figure 1.2: An image demonstrating Meissner effect

Meissner Effect

An important property of superconductors was found in addition to the diamagnetism. Even if there is a magnetic field applied to the sample before cooling, the sample expels all magnetic fields as soon as it is cooled under T_c. This phenomenon is called the Meissner effect. It was given classical phenomenological backing with the London equation:

$$\nabla^2 H = \lambda^{-2} H$$

which relates the penetration depth λ of the magnetic field with the external applied magnetic field obtained as a consequence of the minimization of energy of the system.

1.2 BCS Theory of Superconductivity

A microscopic theory of superconductivity was proposed by Bardeen, Cooper and Schrieffer in 1957 [1]. The theory was able to predict multiple phenomena of a superconductor such as the Meissner effect, specific heats, penetration depths etc. Experimentally, it was found that the transition temperature depended on the atomic mass of the atoms forming the crystal. This implied that there was some connection between the phonons and the electrons. Another key factor was the presence of critical temperature and critical magnetic field, which suggests the presence of a threshold gap. The electrons must be forbidden to access energy states within reach. Further, the Fermi statistics of the electrons would no longer be valid because electrons seemed to all occupy the ground state. They must have some Bosonic behaviour.

BCS theory suggested that electrons with opposite spins pair up under a weak attractive potential. The pair is called a 'Cooper pair'. The superconductivity arises as a macroscopic effect out of the condensation of these Cooper pairs. The attractive interaction between electrons is attributed to the interaction between electrons and phonons. An electron moving in a lattice must deform the crystal slightly, which causes an electron of the opposite spin to move into the region of higher positive charge density. This results in an interaction between the electrons and does not require the electrons to be physically close together. The energy required to break a single pair in the lattice is related to the energy required to break all the pairs, hence giving rise to a bandgap.

According to the BCS theory, the bandgap would be dependent on the temperature. As the temperature increases the quasi-particles increase. Since there are fewer pairs to break, their contribution to the energy gap reduces until it eventually becomes zero. This is experimentally verified as the frequency of absorption of light changes in the superconductor. The band gap's (Δ) dependence on the temperature is analytically given by [2]:

$$\frac{1}{N(0)V} = \int_0^{\hbar\omega_c} \frac{\tanh \frac{1}{2}\beta \left(\xi^2 + \Delta^2\right)^{1/2}}{\left(\xi^2 + \Delta^2\right)^{1/2}} d\xi \tag{1.1}$$

where $1/N(0)V = \eta$ is the inverse interaction strength (equal to 3.03 for weak coupling superconductors), and β renders the temperature dependence. This integral is analytically not solvable and can only be solved numerically.
At the critical temperature T_c the band gap drops to zero. We can obtain T_c by evaluating the above integral.

$$\frac{1}{N(0)V} = \int_0^{\beta_c \hbar\omega_c/2} \frac{\tanh x}{x} dx$$

This integral yields $\ln\left(A\beta_c\hbar\omega_c\right)$, where $A = 2e^{\gamma}/\pi \approx 1.13$ and γ here is Euler's constant $\gamma = 0.577$ [2]. Consequently,

$$kT_c = \beta_c^{-1} = 1.13\hbar\omega_c e^{-1/N(0)V} \tag{1.2}$$

Using equation (1.1) and (1.2) we numerically obtain the following graph for the temperature dependence of the band gap using code mentioned in A.1.1.
A way of measuring this band gap is using the tunneling of electrons from/to a superconducting material. Tunnel junctions are used in this report to study both tunneling as well as measure band gaps in superconductors.

1.3 Tunnel Junctions

Quantum mechanical tunneling provides a useful tool for studying the spectral properties of superconductors. When an electron tunnels across a thin insulating barrier, its energy remains unchanged and it can therefore be used to carry spectroscopic information. Tunnel junctions were first used by Giaver [3] for measuring the dependence of energy gap and the density of states in a superconducting metal. It helped confirm BCS's predictions and made a great contribution towards the understanding of superconductivity. Giaver won the Nobel prize for his studies.

An electron in one metal can 'jump' to another through quantum mechanical tunneling. The electron's wavefunction gets attenuated by an exponential decay outside the surface of the metal. If the metals are placed sufficiently close together, then the probability that an electron can 'jump' to another metal bulk is larger. This is called quantum mechanical tunneling. In similar normal metal junctions, the tunneling happens both ways. We can however, induce biased tunneling by raising the Fermi level of one of the metals. This is done by providing a voltage

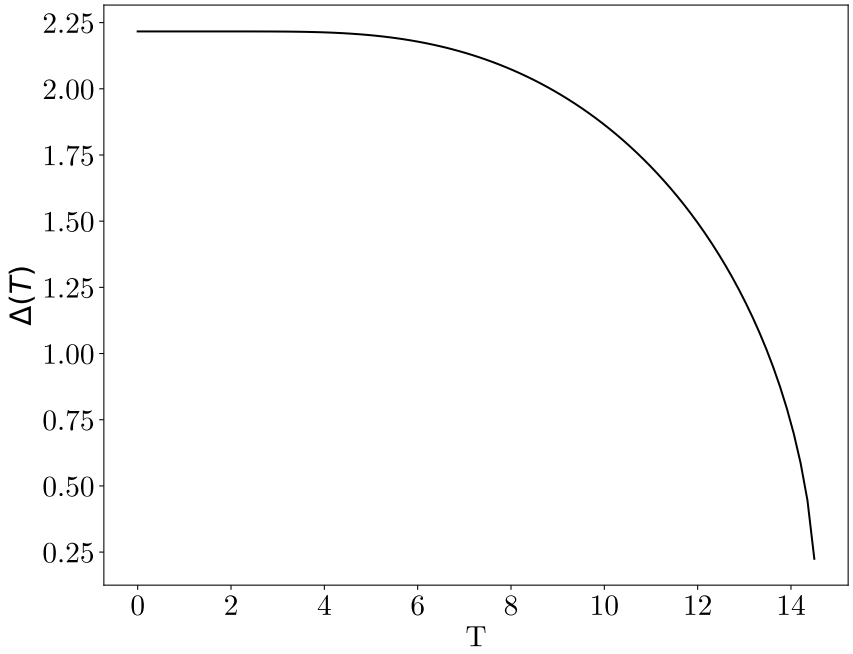


Figure 1.3: $\Delta(T)$ (meV) vs T (Kelvin) curve with $T_c = 14.5$ Kelvin and $\eta = 3.03$

bias to the metals. The tunneling current is found to increase linearly with the bias. Normal to normal metal tunneling current is given by

$$\begin{aligned} I_{nn} &= A|T|^2 N_1(0) N_2(0) \int_{-\infty}^{\infty} [f(E) - f(E+eV)] dE \\ &= A|T|^2 N_1(0) N_2(0) eV \equiv G_{nn} V \end{aligned}$$

where T is the transition matrix element, N the density of states and 'f' the Fermi-Dirac distribution.

With a superconductor junction however, the existence of band gap and the pairing of electrons introduces new physics into the system. Superconductor's ground state only admits paired electrons and the band gap prohibits the existence of any electrons up till a certain level. In a system with normal - superconducting junction, an electron from the normal metal cannot jump to the superconductor's ground state. To jump to the superconductor, it's energy must be high enough to overcome the band gap of the superconductor. The tunneling current formula admits this change in the energy dependence of the DOS[4]:

$$\begin{aligned} I_{ns} &= A|T|^2 N_1(0) \int_{-\infty}^{\infty} N_{2s}(E) [f(E) - f(E+eV)] dE \\ &= \frac{G_{nn}}{e} \int_{-\infty}^{\infty} \frac{N_{2s}(E)}{N_2(0)} [f(E) - f(E+eV)] dE \end{aligned} \tag{1.3}$$

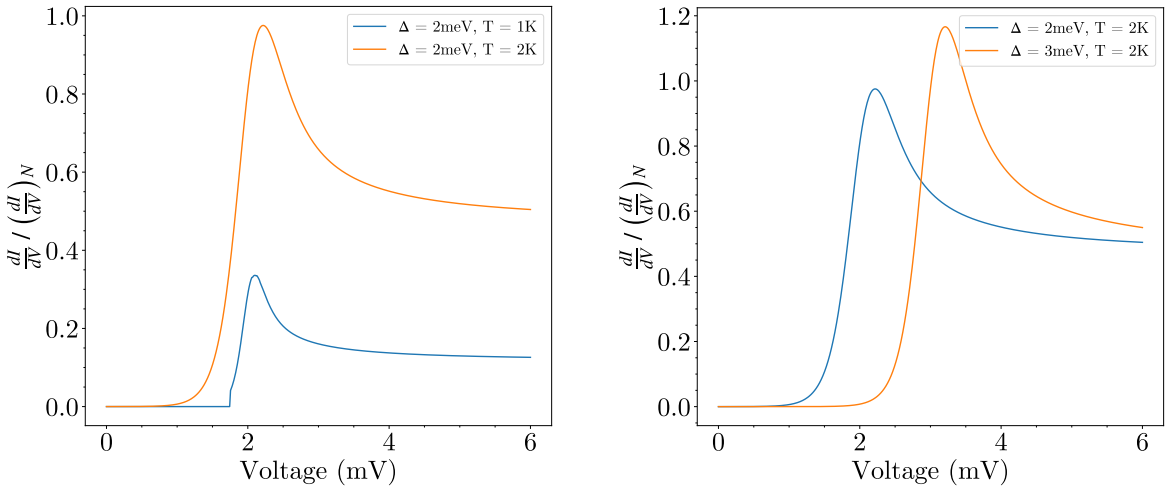


Figure 1.4: Simulated data for superconductor-normal metal junction. It is evident that tunneling current peaks when the band gap energy matches the bias provided to the tunnel junction.

At $T = 0$ there is no tunneling current until $e|V| > \Delta$. The conductance dI/dV is:

$$\frac{dI}{dV} = G_{nn} \int_{-\infty}^{\infty} \frac{N_{2s}(E)}{N_2(0)} f'(E + eV) dE \tag{1.4}$$

This provides an easy framework to measure the DOS and the band gap of a superconducting material [3].

Further, with increasing temperature it was found experimentally that the curves as demonstrated in Figure (1.4) broaden with temperature. This is interpreted by a broadening due to finite-lifetime effects of the quasi-particles at the gap edge [5]. It can be accounted for, by the introduction of an imaginary phenomenological parameter Γ:

$$\frac{N_s(E)}{N(0)} = \begin{cases} \mathrm{Re}\left[\frac{Abs[E+i\Gamma]}{\sqrt{[E+i\Gamma]^2-\Delta^2}}\right], & (|E| > \Delta) \\ 0 & , (|E| < \Delta) \end{cases} \tag{1.5}$$

1.3.1 Magnetic fields

In an external magnetic field H, the BCS quasiparticle energy changes by μH. The energy is then $E = (\epsilon_k^2 + \Delta^2)^{1/2} \pm \mu H$. The quasiparticle density distribution splits into two, spin up and spin down states:

$$N_s^{\uparrow} = \frac{1}{2} N(0) \frac{\epsilon_k - \mu H}{\left[(\epsilon_k - \mu H)^2 - \Delta^2\right]^{1/2}} \equiv \frac{1}{2} N(0) \rho_{s\uparrow} \tag{1.6}$$

$$N_s^{t} = \frac{1}{2} N(0) \frac{\epsilon_k + \mu H}{\left[(\epsilon_k + \mu H)^2 - \Delta^2\right]^{1/2}} \equiv \frac{1}{2} N(0) \rho_{s\downarrow} \tag{1.7}$$

Following the Giaver's model of tunneling conductance outlined before, we get:

$$g(V) = \frac{1}{2}\int_{-\infty}^{\infty} (\rho_s \uparrow + \rho_s \downarrow) \frac{\beta \exp[\beta(E+eV)]}{\{1+\exp[\beta(E+eV)]\}^2} dE \tag{1.8}$$

One can also include spin polarization in case of ferromagnetic materials [8] to get:

$$\begin{aligned} dI/dV \propto N_\uparrow |M_\uparrow|^2 \int_{-\infty}^{\infty} \rho_\uparrow(E,H) f'(E+eV) dE \\ + N_\downarrow |M_\downarrow|^2 \int_{-\infty}^{\infty} \rho_\downarrow(E,H) f'(E+eV) dE \end{aligned}$$

where $N_{\downarrow\uparrow}$ is the spin down (up) DOS of the ferromagnet at E_f, $M_{\downarrow\uparrow}$ is the spin down (up) matrix element for transmission, $\rho_{\downarrow\uparrow}$ is the spin down (up) superconducting density of states, and f' is the derivative of the Fermi function with respect to V. The polarization we measure is given by

$$P = \frac{|M_\uparrow|^2 N_\uparrow - |M_\downarrow|^2 N_\downarrow}{|M_\uparrow|^2 N_\uparrow + |M_\downarrow|^2 N_\downarrow}$$

Maki originally showed that the densities are given by:

$$\rho_{\downarrow\uparrow}(E) = \frac{\rho(0)}{2} \operatorname{sgn}(E) \operatorname{Re}\left(\frac{u_\pm}{(u_\pm^2 - 1)^{1/2}}\right),$$

where $u_+$and $u_-$are implicitly defined by

$$u_\pm = \frac{E \mp \mu H}{\Delta} + \frac{\zeta u_\pm}{(1-u_\pm^2)^{1/2}} + b\left(\frac{u_\mp - u_\pm}{(1-u_\mp^2)^{1/2}}\right). \tag{1.9}$$

where ζ is the orbital depairing parameter, and b is the spin-orbit scattering parameter.

Chapter 2

Experiment Details

To study and characterize superconducting tunnel junctions, one requires to fabricate the tunnel junction (via sputtering), and a cryostat to cool down the sample below T_c. For this project two cryostats - a dry system and a wet system - were used and are outlined below along with the standard experimental techniques used to measure and fabricate the tunnel junctions.

2.1 Cryogenic Systems

Two sets of cryo-stats were used to perform resistance v/s temperature (RT), mutual inductance v/s temperature (MT) and current v/s voltage (I-V) measurements. All the cryostats require a coolant/cryogen which in this case is He4. He4 has a boiling point of 4.2K at one atmosphere pressure and is below the T_c of our sample (NbN) thus making it an optimum choice. The systems used for the report are outlined:

2.1.1 Dry System

The dry system operates without any liquid helium in the system. The system consists of an inverted cold head mounted on a table top. The cold head is connected to a compressor that cycles Helium to the cold head in double walled lines. The cold head is also connected to a turbo molecular pump, and houses the sample mounts, heaters and sensors.
Working: The turbo molecular pump is used to maintain vacuum between the inner and outer shroud of the cold head. Vacuum (10^{-6} mbar) provides insulation as well as prevents any condensation/ice-formation as the system cools down to cryogenic temperature. After achieving vacuum, the compressor is switched on. Room temperature helium gas is first compressed and then supplied to the refrigerator via flexible gas lines. The compressed helium is cooled by expansion via a piston on top of the cold head and provides cooling. After cooling, the helium is returned to the compressor to repeat the cycle. The temperature achieved is 2.7 Kelvin.

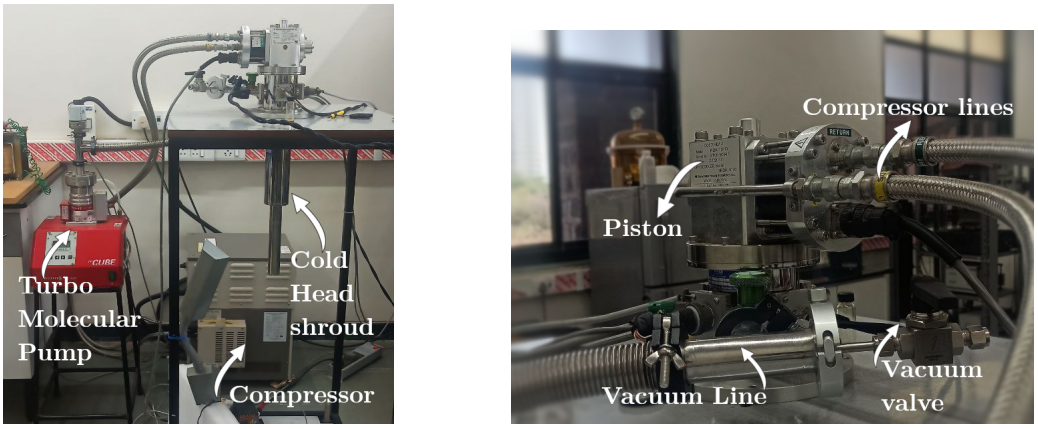


Figure 2.1: The cryo setup is shown with all necessary parts highlighted.

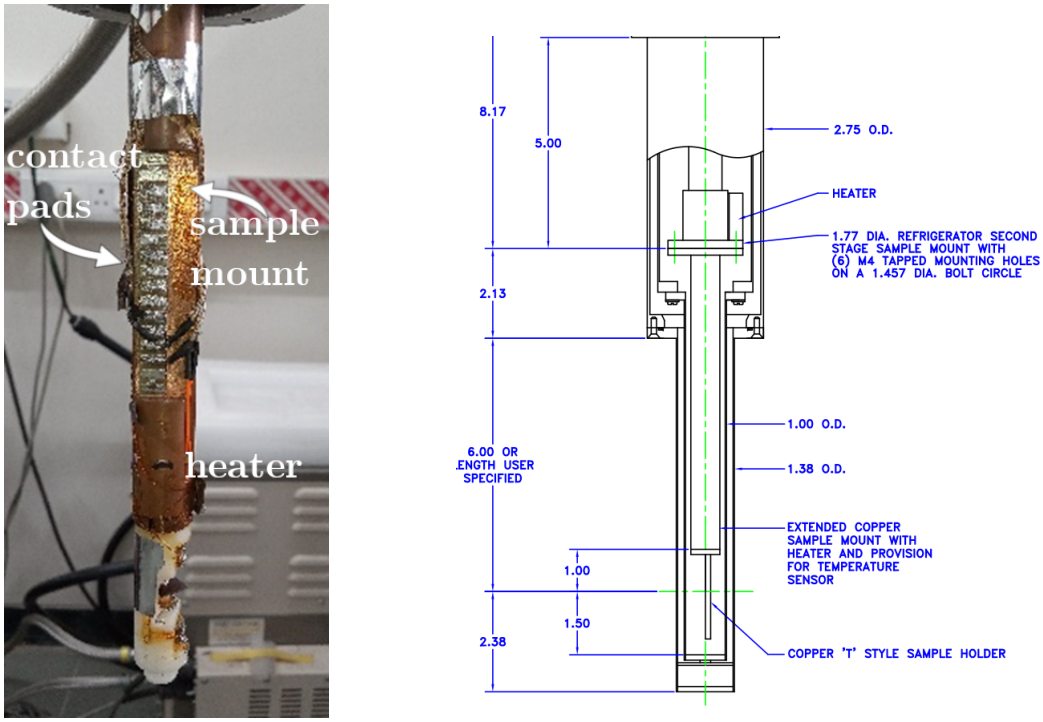


Figure 2.2: The schematic of the cold head is shown on the right. The image of extended sample mount is shown on the left. The schematic shows the sample mount surrounded by a smaller shroud, and further surrounded by another shroud to form an annular vacuum arrangement.

The cold head consists of the sample mount (shown above) and has 8 contact pads for doing 2 RT measurements simultaneously. In addition it has two thermally coupled resistive heaters, two thermally coupled zirconium oxynitride temperature sensors, one placed near sample and one placed near the refrigerator. Towards the ground, there is a quadrupolar-dipolar coil arrangement mounted on a polymer block to measure MT and penetration depth measurements. The cold head is finally connected to a PID based 335 Lakeshore temperature controller, SR830 lock-in amplifiers and Keithley 2400 current source meters.

2.1.2 Wet System

The wet system is called so, because the sample is put in a liquid helium bath. The system is a continuous flow cryostat and can achieve up to 2.2 Kelvin. In this cryostat, liquid 4He is introduced using the inner tube of a two-way transfer tube which is dipped directly into a helium Dewar. The outer tube is used for pumping and removal of the evaporated 4 He gas which goes into the 4 He recovery line. The extent of pumping is controlled by a flowmeter and the flow of liquid from the Dewar is controlled by a needle valve. The lowest temperature of 2.2K was attained by flooding the cryostat with liquid He, closing the needle valve and then pumping over the liquid using a rotary pump.

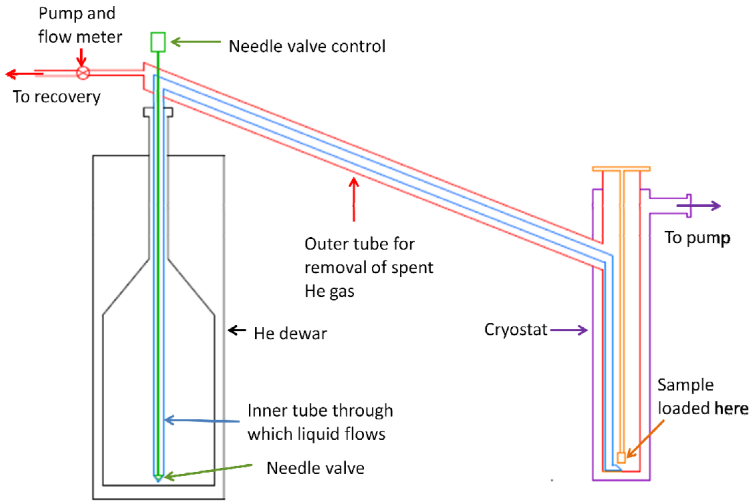


Figure 2.3: A working schematic of the continuous flow cryostat.

2.2 Sputtering system

Sputtering is a thin film deposition technique. The basic principle involves creation of a plasma by the application of high voltage with a negative potential on the target. Positive ions in the discharge are accelerated towards the target and dislodge individual atoms/molecules after hitting the target. This material then travels across the discharge and falls on a substrate where it condenses. This condensation eventually forms a thin film atom by atom. We can form thin films of required shape and size by using masks on the substrate.

In our case, we have used DC [1] magnetron sputtering. A static magnetic field is introduced by placing an array of magnets behind the target. The magnetic field modifies the paths of the charged particles into closed loops and increases the density of bombardment resulting in a higher deposition rate.

[1] Our target, Niobium is conductive.

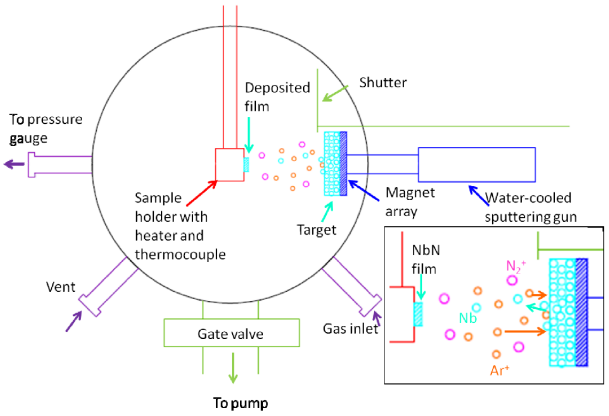


Figure 2.4: A working diagram of the sputter system.

2.3 Techniques

- **Soldering**: Excessive heat destroys the oxide layers in tunnel junctions. Therefore, we use Indium wire for making electrical contacts with the junctions. Indium wire melts at ∼150°C which is relatively low. The iron is also set to a lower temperature and care is taken to contact the sample only for a short period of time.

- **Four Probe Method:** The electrical transport measurements of the tunnel junctions are taken by the four probe method. Four equally-spaced, co-linear probes are set up on the material. A DC current is applied between the outer two probes and a voltage drop is measured between the inner two probes. This eliminates the counting of the contact resistance of the current supplying probes.

- **RT Measurements** : RT measurements are simply done by the four probe method with constant data acquisition of temperature, current and voltage. The current is set to $\sim 5 \times 10^{-6}$ amps so as to produce minimal local heating effects and the temperature is sweeped across the desired range.

- **T_c Measurement**: The RT measurement is done on a heating cycle, that allows a stable temperature control with a PID system. The temperature at which sample displays 90% of it's normal resistance is classified as the critical temperature.

- **MT Measurement**: MT measurements are done on thin film samples mounted between the quadrupolar-dipolar coil arrangement. An AC signal is passed onto the quadrupole coil and is picked up the dipole coil beneath the sample. The signal generation is done by a SR830 lock-in amplifier, which is then amplified and sent to the quadrupole coil. The current is measured by measuring voltage across a 10 Ohm resistor after amplification. The signal from the dipole coil is detected by another SR830 lock-in amplifier.

- **IV Measurement**: The tunnel junction's IV is measured by keeping temperature constant with the help of a PID based temperature controller and sweeping current across the junction. An average of 30 readings is taken at one set of parameters to reduce the noise. The cycle is then repeated at different temperatures.

2.4 Tunnel Junctions

We use NbN/native oxide/Ag as our superconductor - insulator - normal metal tunnel junction. The details of the junction are mentioned below.

2.4.1 Fabrication

A pure Niobium sample is loaded as the target and a [1 0 0] single crystal MgO substrate with the required mask is loaded on the substrate holder with silver paste. The substrate temperature is raised to and maintained at 600 ^{o}C using a PID loop. The sputter chamber is subsequently evacuated using a turbo-molecular pump to a pressure of $\sim 5 \times 10^{-6}$ Torr. A mixture of Ar and N_2 gas is let in, controlled by mass flow controllers. The power supply is switched on, ionizing the Argon around 380V. The Argon ions start hitting the target. Negatively charged Niobium ions fly away from the target and react with the Nitrogen gas in the environment before condensing on the substrate. The sputtering is done at a constant 250Watts until the desired thickness of the sample is condensed on the substrate.

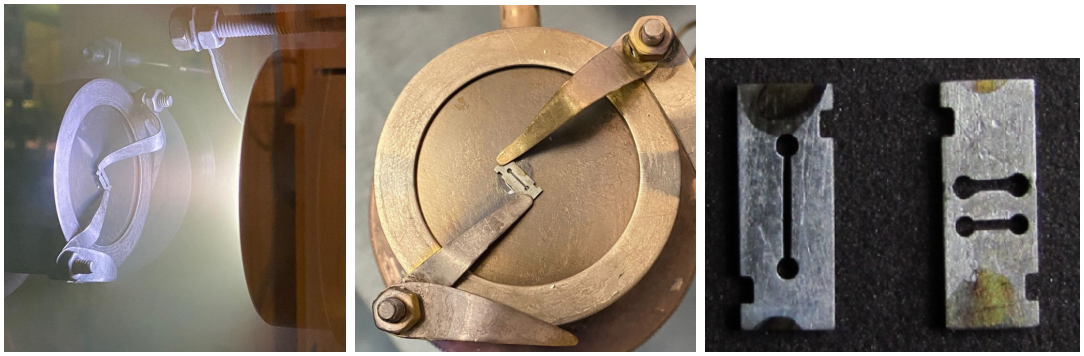


Figure 2.5: Images depicting the sputtering plasma glow (left), a loaded substrate with the NbN mask (middle) and the shadow masks for NbN and Ag (right).

NbN is deposited as a 300μm wide strip-line. The sample is then allowed to cool to 250 ^{o}C in vacuum. Once this temperature is reached, the chamber is vented and kept exposed to air at 250 ^{o}C for a period of 60-120 minutes to allow a thin oxide layer to form. This temperature was chosen as it was high enough to cause sufficient oxidation on the surface and yet not so high that the entire film gets oxidized or damaged. After the oxidation, the chamber is once again evacuated and the sample cooled to room temperature. The sample is then removed from the sputtering chamber and immediately loaded into an evaporation chamber with another shadow mask to deposit cross-strips of silver to make the tunnel junctions.

The typical base vacuum used in the evaporation chamber was $\sim 5 \times 10^{-5}$ Torr. The silver is evaporated by passing a current of ~65A through a tungsten boat or filament holding $\sim$ 60-70mg of silver wire. A shutter is used to cover the boat/filament during the first few seconds of evaporation to prevent any impurity on the surface of the silver wire from getting deposited on the sample. This step completes the fabrication of the tunnel junction.

2.4.2 Measurement

Standard techniques are used to measure RT and I-V characteristics of the tunnel junctions. Contacts are made on tunnel junctions with the Indium solder as shown in figure 2.6. IV data is taken by having two probes on the silver and two probes on the Niobium Nitride layer. The substrate with the junction is mounted on the sample mount with g-varnish that thermally couples the substrate to the system. The contacts from the junction are made to the contact pads and the cryostat is turned on for the measurement. The data acquisition is performed by pre-programmed LABVIEW programs that communicate with the electronics through the GPIB protocol.

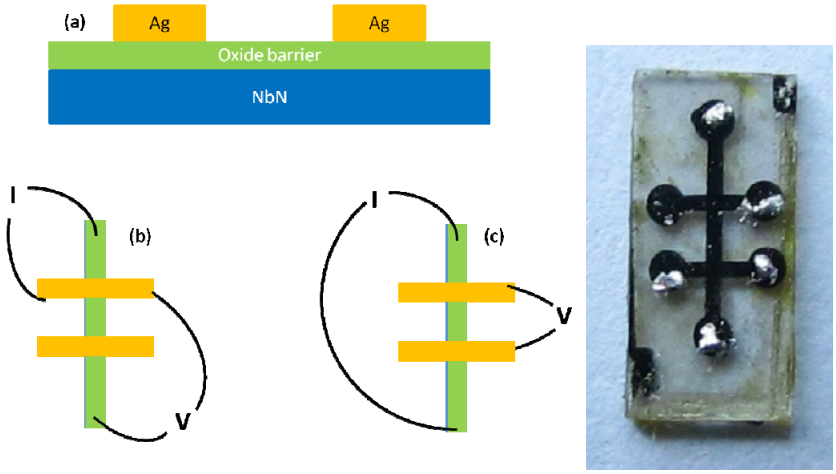


Figure 2.6: A schematic depicting the connections for measurement of the tunnel junction (left) and a close picture of a tunnel junction (right). The connections for RT measurements are shown in (c) and I-V measurements are shown in (b).

Chapter 3

Experiments and Analysis

In this chapter we present and analyze the data acquired through the experimental setup outlined in Chapter 2.
We start by measuring RT of a NbN sample and MT of a Nb sample to gain an idea of the critical temperature and penetration depths. Further data from different tunnel junctions is also presented.

3.1 Preliminary measurements

3.1.1 RT measurement

RT of a NbN sample is measured and the following data is acquired:

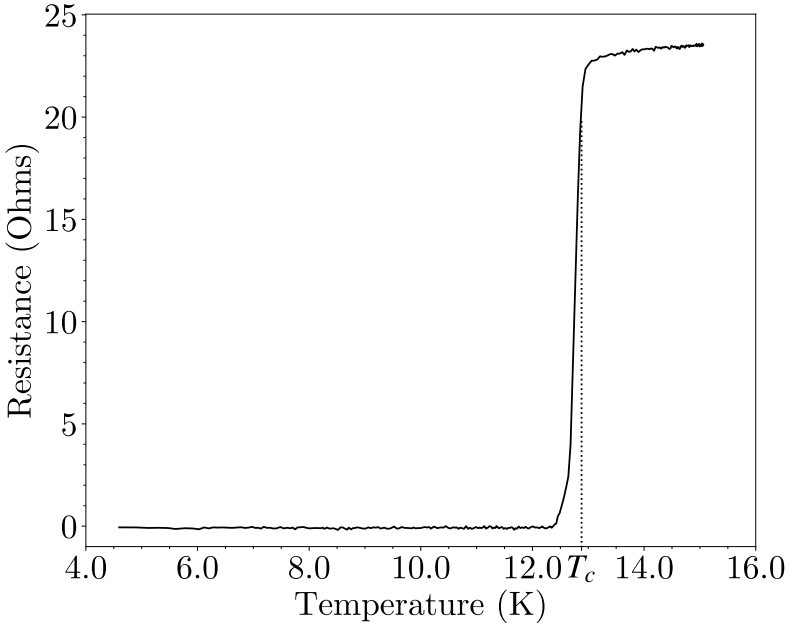


Figure 3.1: RT of a 14nm NbN film taken during a heating run.

90% of the resistance is reached at 12.83 ± 0.05 Kelvin which is designated as

our critical temperature.

3.1.2 MT measurement

MT of a thin NbN film is measured and the following data is acquired:

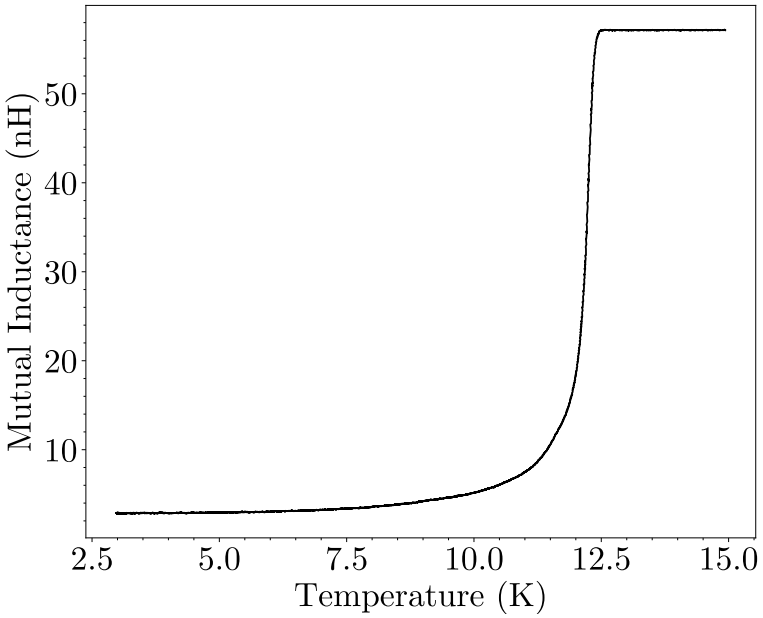


Figure 3.2: MT of a 5nm NbN film taken during a heating run.

90% of the mutual inductance is reached at 12.36±0.05 Kelvin. This is very close to the previous RT experiment too. Further a clear demonstration of Meissner effect (section 1.1.2) is evident as the mutual inductance drops. The magnetic fields are not able to penetrate the sample as it turns superconductive. Lower temperatures result in lower penetration depths. Penetration depth measurements can be done with multiple sets of data for different thickness samples, however, it is out of the scope of this report.

3.2 Tunnel Junctions

The I-V characteristics of tunnel junctions were studied at different temperatures. The process of data analysis is highlighted below, which will be repeated for each set of data later:

3.2.1 Data Analysis example

A sample set of I-V data at a temperature of 4 Kelvin with analysis is shown below:

The I-V data shown on the right was collected at the temperature of 5 Kelvin. The current sweep ranges from -0.8mA to +0.8mA. Looking at the I-V curve it is evident that current starts flowing after a certain potential bias is reached.

To study the rise of the current, we numerically differentiate our I-V data with respect to voltage and get conductance. For our purposes, we have also removed the background (discussed in appendix A.1.1) , smoothened and normalized the data about a point far from the peaks such that the slope of current v/s voltage is constant. The peak of dI/dV gives us the maximal conductance i.e. it corresponds to the point where the external potential bias matches the band gap of the superconductor to allow tunneling. Thus, distance of the peak from the origin corresponds to the bandgap of the superconductor. To extract the bandgap both qualitatively and quantitatively, we can fit it to equation (1.4).

We can therefore, simply fit the curve obtained for parameters Δ and Γ according to equations (1.4) and (1.5). An automatic program for fitting these parameters was developed and is explained in Appendix A.1.1.
The fit generated by the program is shown as the last image, with the parameters extracted as $\Delta = 3.002 \pm 0.003$meV and $\Gamma = 0.123 \pm 0.002 meV$ [1].

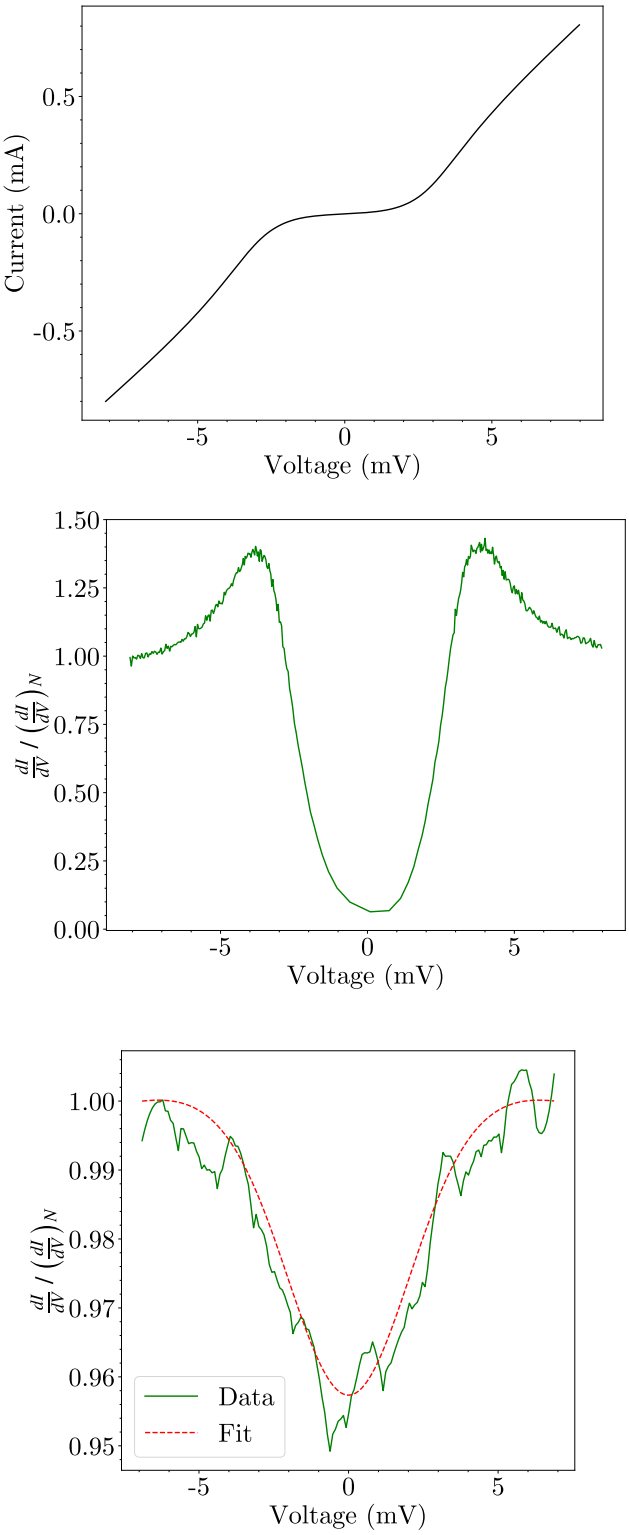
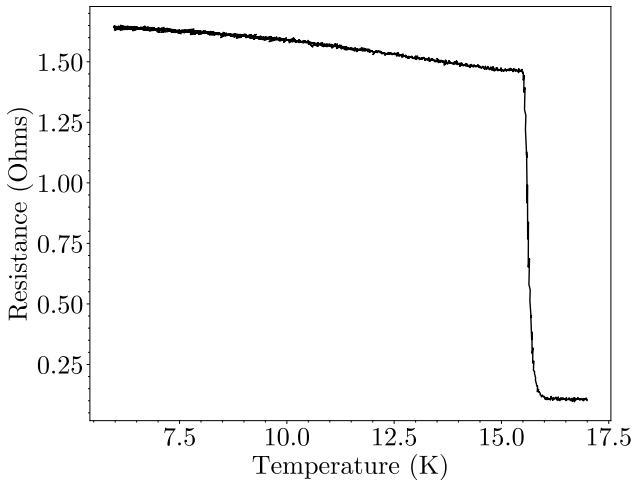


Figure 3.3: The I-V and conductance data collected from the tunnel junction at 4 K.

[1]The errors are calculated through a least square fitting program, and do not account for errors in the instruments which are much smaller than the errors listed.

3.2.2 Tunnel Junction 1

On collecting the RT data of a tunnel junction we get the following:



The resistance of the tunnel junction drops as soon as the NbN transitions to the normal state. This happens because the band gap ceases to exist and therefore electrons are able to tunnel through even at a minimal potential bias. The T_c is found to be ~ 15.8 Kelvin.
Further, the I-V measurements are taken, differentiated and smoothened to obtain:

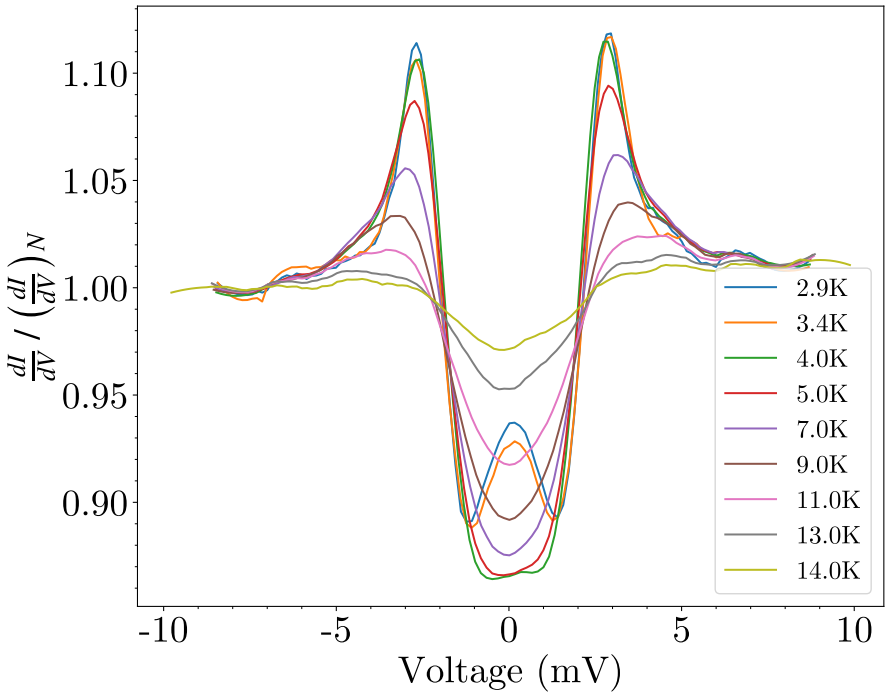


Figure 3.4: Conductance data collected at different temperatures.

The conductance curves at lower temperatures show an unexpected behaviour. There is a peak at around null voltage for lower temperatures. It corresponds to a phenomenon called *Andreev Reflection*. The reflection suggests that there are parts of the superconducting film in direct contact with the metal i.e. the oxide layer is punctured. This could be due to longer period spent in soldering or simply the tunnel junction being bad to begin with. Other conductance curves are as expected, however, noise is present even after smoothening. The data must be averaged over more number of points. We proceed to measure another tunnel junction keeping in mind these corrections.

3.2.3 Tunnel Junction 2

Another tunnel junction was fabricated and measured. The conductance of the junction came out as:

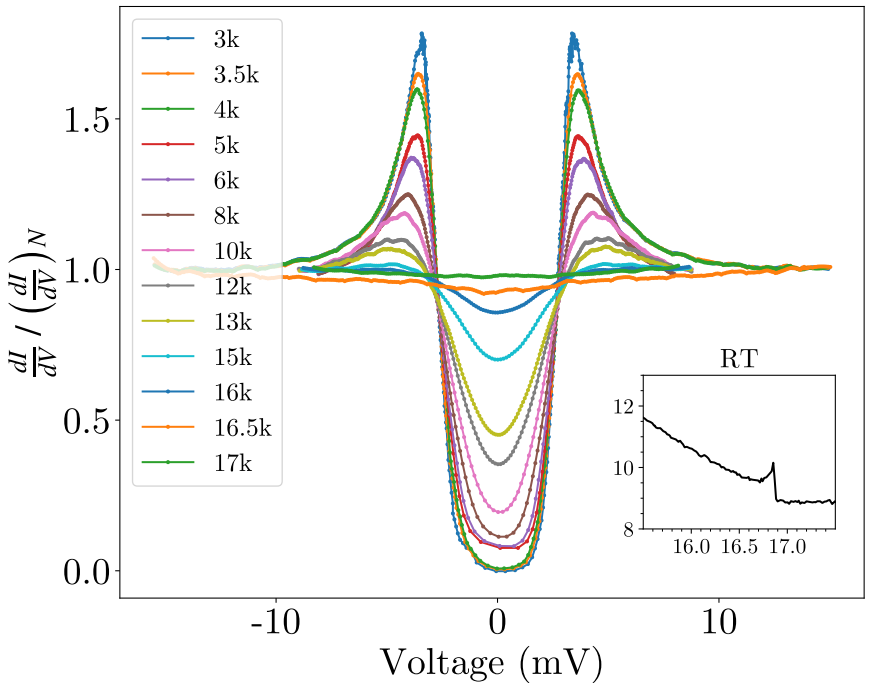


Figure 3.5: Conductance data collected at different temperatures.

The data above does not show any peaks and is free of noise to allow analysis. It is evident that the change in conductance becomes fully normalized at 17 Kelvin i.e. the NbN is in the normal state. The same can be confirmed form the inset RT graph. To begin our analysis, we begin by smoothening the data and removing linear bias:

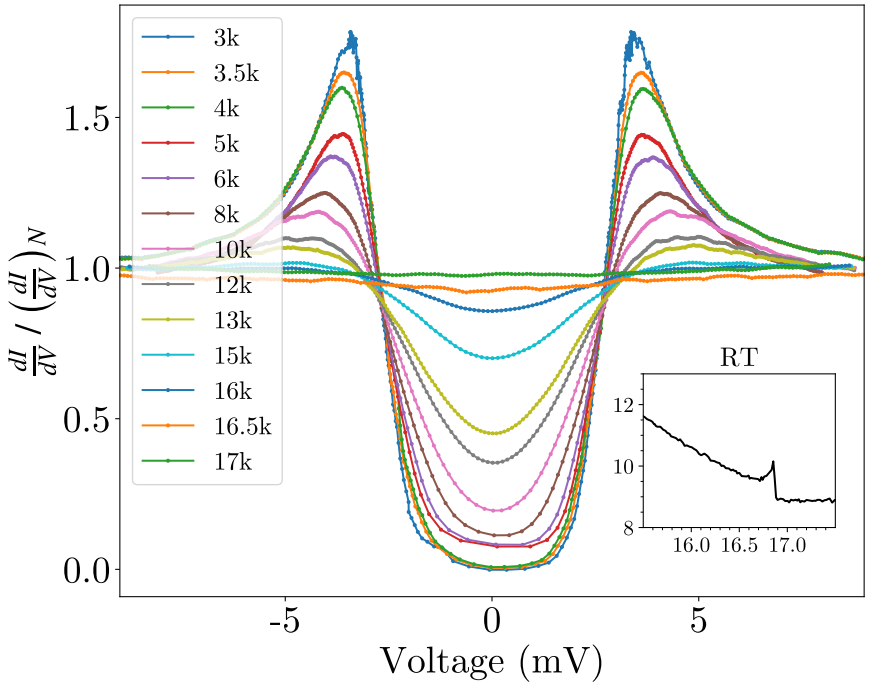


Figure 3.6: Conductance data smoothened and linear bias removed

One of the main features highlighting the physics of these junctions can be noted from the above graph. The conductance peaks reduce in height with increasing temperature which goes hand in hand with the quasiparticle interactions increasing with temperature. The band gap decreases as we go higher in temperature. This will be evident once we extract the parameters by fitting with the program developed and outlined in A.1.1. We get the following outputs:

Temperature (K)	Γ (meV)	±Γ	Δ(meV)	±Δ
3	0.168	0.004	3.050	0.004
3.5	0.186	0.005	3.045	0.005
4	0.186	0.004	3.061	0.004
5	0.123	0.003	3.002	0.003
6	0.114	0.003	2.991	0.003
8	0.131	0.003	2.942	0.004
10	0.130	0.004	2.830	0.005
12	0.259	0.005	2.594	0.006
13	0.358	0.008	2.435	0.008
15	0.761	0.029	1.871	0.021
16	1.008	0.062	1.420	0.016
16.5	1.520	0.344	0.902	0.055

Table 3.1: The parameters extracted from the fit.

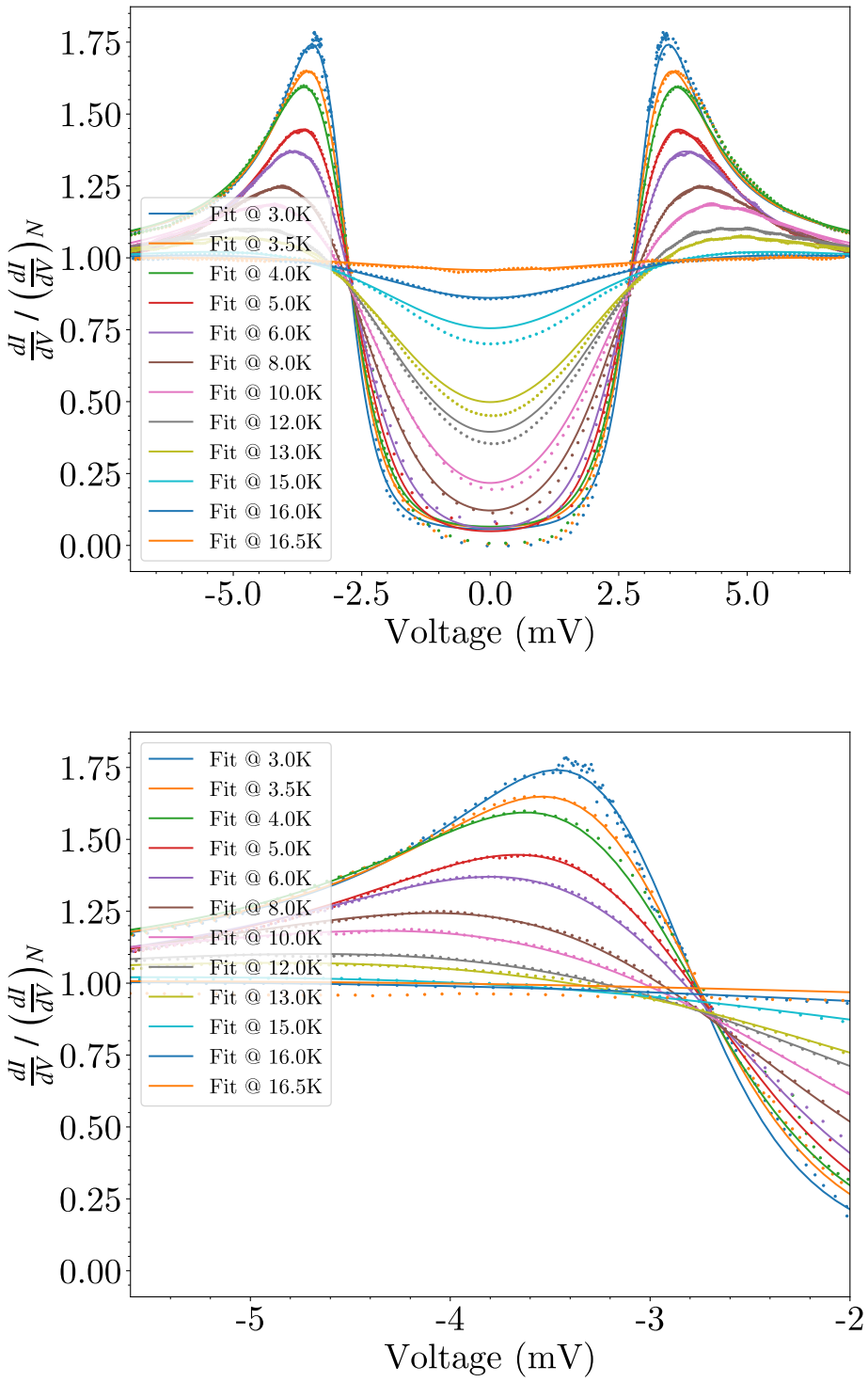


Figure 3.7: Fit of the collected data after removing the background, smoothening and normalising.

There are certain patterns that can be noted from the Table 3.1. Firstly, Γ values which depict the broadening and lifetime of the quasiparticles [5] are very small. The broadening of the gap edge, is also very clear as the temperature is

raised. At temperature close to the critical temperature, Γ blows up relevant to the physics of the quasiparticle lifetimes.

Similarly, Δ shows a clear downward trend as the temperature increases. We verify it against equation (1.1) below:

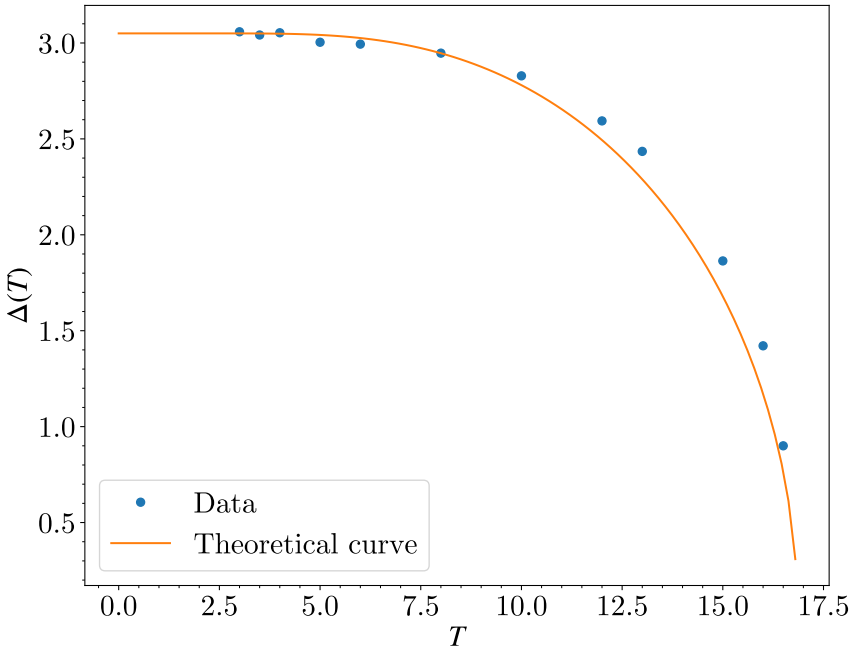


Figure 3.8: Δ v/s T theoretical curve compared with the Δ v/s T parameters extracted from the experiment and tabulated in Table 3.1. T_c = 16.8 K for the theoretical curve.

Hence, we find that the analysis that we performed agrees strongly with the BCS microscopic theory of superconductivity.

3.2.4 Magnetic field measurements

We take I-V data again, however, with magnetic field applied perpendicular to the tunnel junction. It gives rise to other physical phenomena such as Zeeman splitting, effects of pair breaking, effect of spin-orbit interaction and gives us a measure of parameters such as the orbital de-pairing parameter according to the Maki theory mentioned in[6].

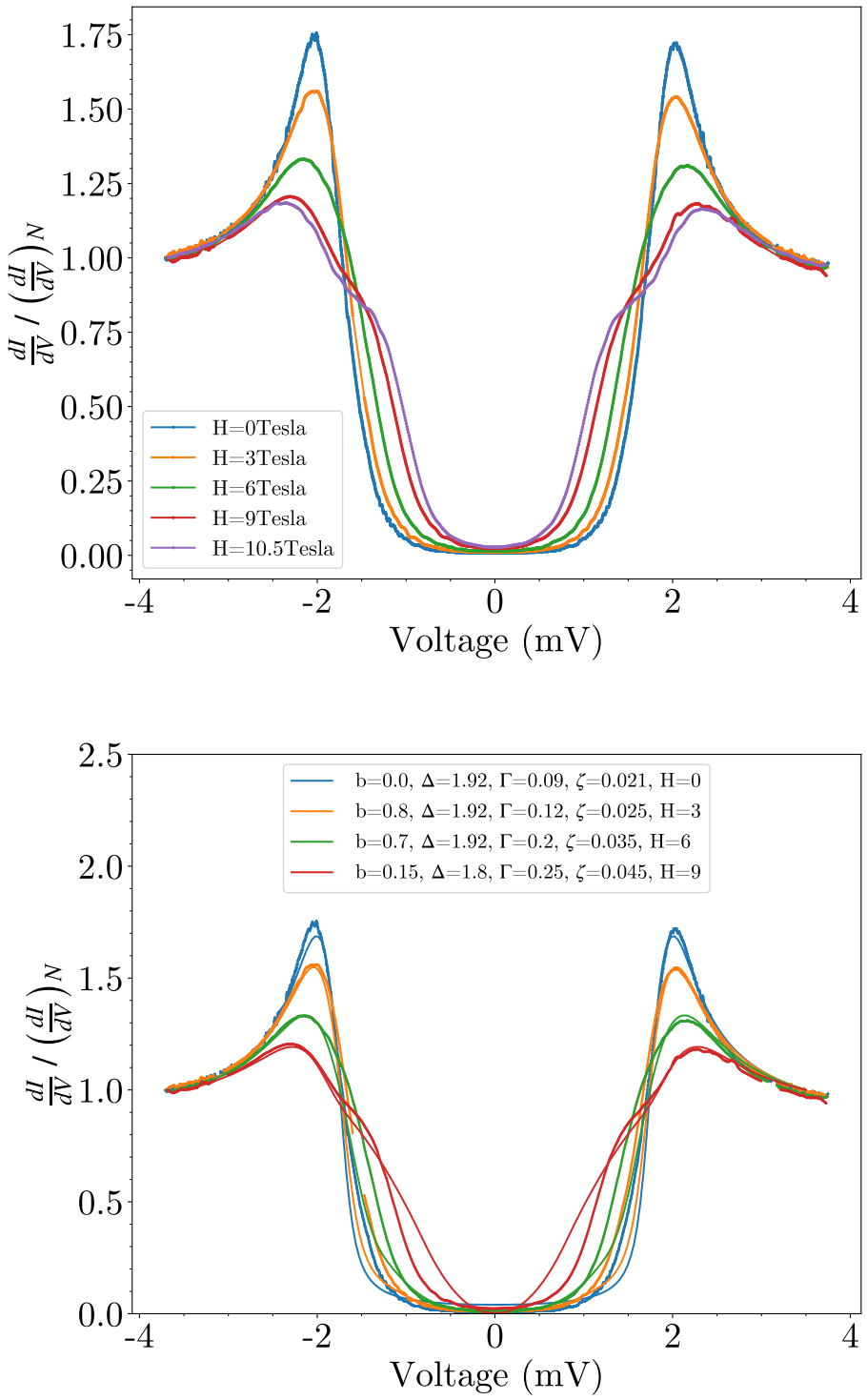


Figure 3.9: The dI/dV data collected with magnetic field perpendicular to the tunnel junction. Next image shows the fit generated through the program in appendix A.3. The data was collected at 400mK temperature.

As shown in figure 3.9, we have almost a good fit of magnetic tunneling conductance.we were able to extract the following parameters:

b	Δ (meV)	Γ (meV)	ζ	H (Tesla)
0.0	1.92	0.09	0.021	0
0.8	1.92	0.12	0.025	3
0.7	1.92	0.2	0.035	6
0.15	1.8	0.25	0.045	9

Clear trends for all the parameters are evident as the magnetic field increases. 'b', the spin-orbit scattering parameter has an increasing trend with the increasing field. 'ζ', the orbital de-pairing also has an increasing trend with the magnetic field. These trends are as expected. 'Γ' values representing the lifetime of quasi particles also increase with increasing magnetic field, which is in accordance with the BCS theory. **NOTE**: The parameters extracted were obtained from a program that involved Taylor approximations. It was later found that the Taylor approximation are not valid, and as a result, the fits depicted are not true fits. A corrected code is implemented using substitutions from J. Alexander's work [7, equation 77] and mentioned in A.3. Lastly, we can also calculate the Zeeman splitting by differentiating the conductance curve:

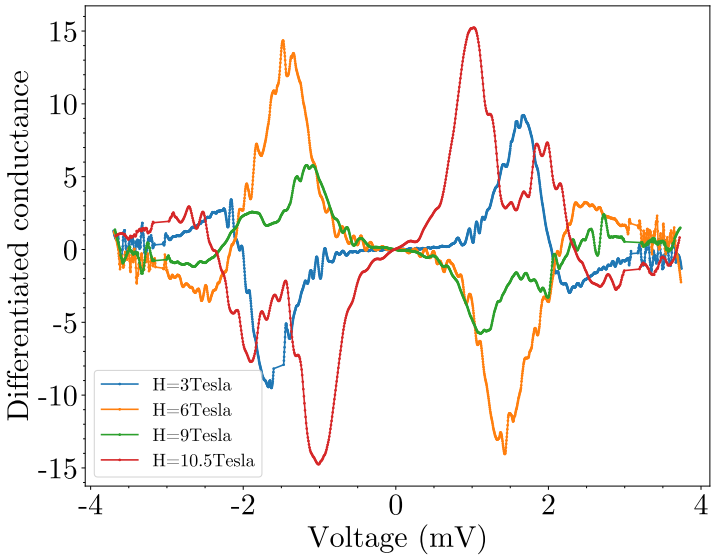


Figure 3.10: The differentiated conductance curves for determining the Zeeman splitting as the difference of the peaks.

The Zeeman splitting was found to be:
3 Tesla : 0.64mV
6 Tesla : 1.07mV
9 Tesla : 0.77mV
10.5 Tesla : 0.94mV
Except one outlier at 6 Tesla, the Zeeman splitting is found to be proportional to the magnetic field. More data would have lead to better statistics.

Chapter 4

Conclusion

Superconductors have been historically probed with tunneling spectroscopy. Tunneling processes are fundamental to quantum mechanics and can provide information directly linked with the physics of the samples. One of the simplest spectrums that can be obtained is the change of the current with the energy of the electrons i.e. voltage. The slope of the I-V curve at each voltage corresponds to the electron density of states (equation (1.4)). We demonstrated electron tunneling spectroscopy by measuring I-V characteristics of a superconductor-normal metal tunnel junction. The same can be done for superconductor-superconductor junctions where other interesting phenomena such as the Josephson effect arise.

In this three month project, we were able to learn equipment and techniques related to (1) fabrication of thin films and tunnel junctions via sputtering and thermal evaporation, (2) using wet and dry 4He based cryogenic systems and techniques of measurement at cryogenic temperatures and (3) program development for data analysis and extraction of parameters depicting physics of the system.

We started with RT and MT measurements of NbN thin films for determining the critical temperature T_c. After getting an idea of working with low-temperature systems, we began measuring tunnel junctions in section 3.2. Tunnel junctions were first measured for I-V characteristics in zero field and band gap energy Δ and quasiparticle lifetimes Γ were extracted (Table 3.1). The dependence of the band gap to temperature was found in perfect agreement with the BCS theory (Figure 3.8). Further we did I-V measurements in perpendicular magnetic fields and used Maki theory [6][8] to extract orbital depairing (ζ) and spin-orbit scattering (b) parameters along with the band gap (Δ) and broadening (Γ) (Figure 3.9). Further, the Zeeman splitting in magnetic fields was also determined in the conductance data acquired.

Future work:
The data analysis of tunnel junctions in magnetic fields was done under a Taylor approximation which is not valid for higher voltage biases. To simulate the tunnel junction one requires to numerically solve highly non-linear coupled equations [8]. The invalid approximations were employed in the non-linear root finding. However, using substitutions from [7, equation 77] one can reduce the non-linearity by encoding the two equations into four. An implementation of these substitutions with the non-linear root finding working and an almost ready conductance simulation is

mentioned in the Appendix A.3.
This analysis will further be applied to spin polarized tunneling, with ferromagnetic materials in place of normal metal (Ag) in our junctions which gives rise to new physics and is an active area of research.

Appendix A

Python simulations and data analysis codes

A.1 Simulation and analysis of the tunneling data

A.1.1 Fitting the tunneling data

Importing all the necessary libraries including the module 'confitmodule' written for the simulation (discussed later):

```
import numpy as np
import matplotlib.pyplot as plt
import confitmodule as cm
from scipy.signal import savgol_filter

%matplotlib inline
plt.rcParams['figure.figsize'] = [10, 8]
plt.rcParams.update({   'font.size': 28,
  "font.family": "serif",
  "font.serif": ["CMU serif"],
  "font.sans-serif" : ["CMU sans serif"],
})
plt.rc('axes', unicode_minus=False)
plt.rc('axes', unicode_minus=False)
plt.rc('pgf', texsystem='pdflatex')
```

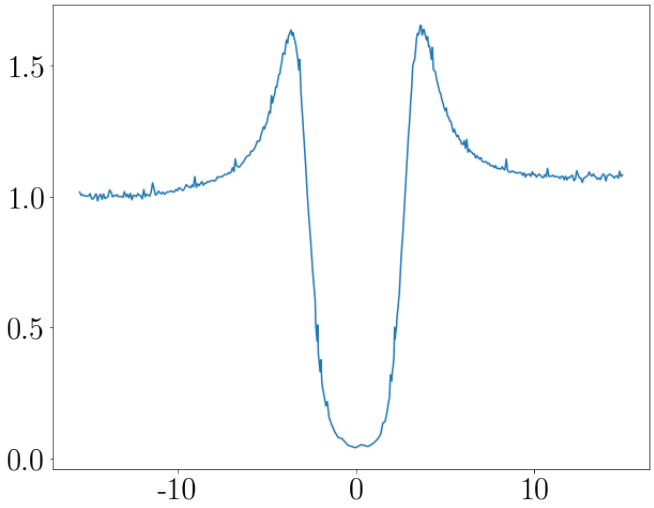
Defining a function 'diff' for numerical differentiation of the IV data:

```
def diff(x,y):
return [(y[i+1]-y[i])/(x[i+1]-x[i]) for i in range(x.size -1)]
```

Reading, differentiating and normalizing the IV data:

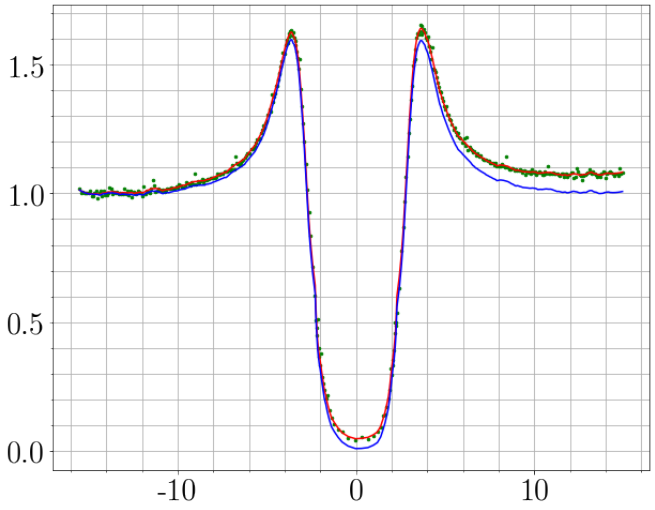
```
path = "/mnt/localdiskd/Semester 8/Project - Superconductivity/
  Code/CBS/data/goodtunneljunc-NbNAloAg_2903+0704/iv4k.txt"
data = np.transpose(np.loadtxt(path, usecols=(0,1),skiprows=1))

didv = diff(data[0],data[1])

dat = np.array([data[0,1:],didv])
dat[1] = dat[1]/np.average(dat[1,5:15])

```

```
9   plt.plot(dat[0],dat[1])
10
```



Using a polynomial approximator based on window length to reduce noise in the data. Further, we remove linear bias in the conductance data. Green dots represents original data, red curve the smoothened data, and blur curve is the data with linear bias removed:

```
1   yhat = savgol_filter(dat[1], 15, 2) # window size, polynomial
      order
2
3   plt.plot(dat[0],dat[1],'.g',markersize=5)
4   plt.plot(dat[0],yhat,'r')
5   plt.minorticks_on()
6   plt.grid(which='both')
7
8   dat[1] = yhat
9
10  slope_shift = (np.average(dat[1][-20:])-np.average(dat[1][:20]))
      /(dat[0][-1]-dat[0][0])
11  plt.plot(dat[0],yhat - (dat[0]-dat[0][0])*slope_shift,'b')
12  dat[1] = yhat - (dat[0]-dat[0][0])*slope_shift
```

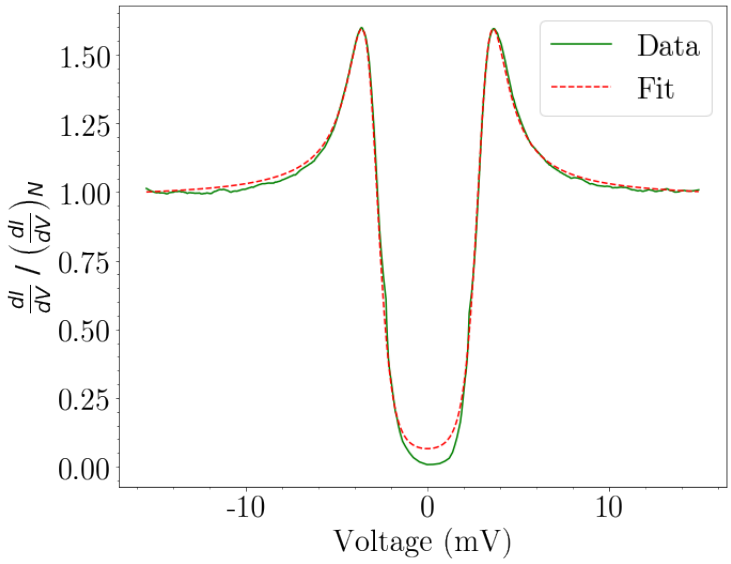


We proceed to fit the peaks using a fititng function in the 'confitmodule' we haev written and discussed later. We develop a mask that assigns higher weight to the points around peaks for better fitting of the band gap and broadening parameters. Then we call the fitting function and print the fit parameters and the associated errors.

```
T=4

lcut1 = -5.5
lcut2 = -2.5
rcut1 = 2.5
rcut2 = 5.5
wt=100

mask = (lcut2 > dat[0])*(dat[0] > lcut1)+(rcut1<dat[0])*(dat[0]<
  rcut2)
unc = mask*1/wt + ~mask*(np.max(dat[1])-np.max(dat[0]))/10

fit,err = cm.Conductancefit(dat,T,unc,inter='False')
print(fit)
print(2.355*np.sqrt(np.diag(err)))

V = np.linspace(dat[0][0],dat[0][-1],500)
G_values = [cm.G_cond(i,0.18617034,3.06114686,T) for i in V]
G_values = np.array(G_values)/G_values[0]
# np.savetxt("abc.txt", G_values)
# plt.plot(dat[0],dat[1]*mask, '.b')
plt.plot(dat[0],dat[1],'-g',label='Data')
plt.plot(V,G_values,'--r',label='Fit')

plt.xlabel('Voltage (mV)')
plt.ylabel(r'${\frac{dI}{dV}} \; / \; {\left(\frac{dI}{dV}\right)
  _N}$')
plt.legend()
plt.minorticks_on()
plt.tight_layout()
plt.show()
# plt.savefig('/mnt/localdiskd/Semester 8/Project -
```

```
Superconductivity/Report/images/tunintrofit.pdf')
```

```
[0.17617034 3.06114686]
[0.00376638 0.00402185]
```



A.1.2 confitmodule

```
import numpy as np
import scipy.constants as const
import scipy.integrate as scint
import scipy.optimize as sc
import matplotlib.pyplot as plt
from numba import njit
from scipy import interpolate


#Constants:
kb = const.physical_constants["Boltzmann constant in eV/K"][0]*1
    e3


@njit
def Dfermi(E,V,T):
return (1/(np.cosh((E+V)/(2*kb*T)) )**2 * (4*kb*T))

@njit
def dos(E,Gam,Delta):
return np.real((np.abs(E)+complex(0,Gam))/np.sqrt((np.abs(E)+
    complex(0,Gam))**2 - Delta**2))

def G_cond(V,Gam,Delta,T):
tmp = lambda E : dos(E,Gam,Delta)*Dfermi(E,V,T)
return scint.quad(tmp,-100,100)[0]


```

```
27  def Conductancefit(data,T,sig,inter=True):
28  t1 = data[0]
29  t2= data[1]
30
31  def vectorize1(V,Gam,Delta):
32  #faster fitting but could have errors as it interpolates in
     between
33  small_V = np.linspace(min(V),max(V),300)
34  out = [G_cond(i, Gam, Delta,T) for i in small_V]
35  def cond(x):
36  tck = interpolate.splrep(small_V, out)
37  return interpolate.splev(x, tck)
38  ans = cond(V)
39  return np.array(ans)/ans[0]
40
41
42  def vectorize2(V,Gam,Delta):
43  #slow fitting, only use when data points are around 400
44  out = [G_cond(i, Gam, Delta,T) for i in V]
45
46  return np.array(out)/out[0]
47
48  bounds = ([0,0],[5 ,5])
49  if inter == True:
50  fit,err= sc.curve_fit(vectorize1,t1,t2,bounds=bounds,sigma=sig)
51  else:
52  fit,err= sc.curve_fit(vectorize2,t1,t2,bounds=bounds,sigma=sig)
53
54  return fit,err
```

A.2 Simulation of Δ v/s T

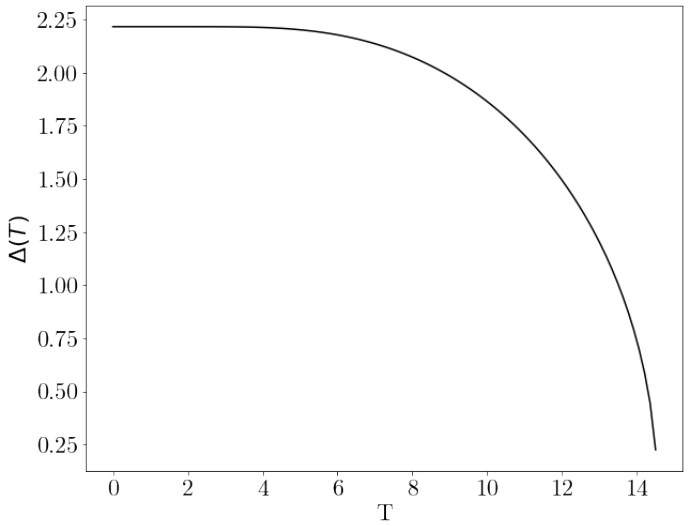
We import all the libraries necessary and set all the variables necessary.

```
1  import numpy as np
2  import matplotlib.pyplot as plt
3  import scipy.constants as const
4  from scipy.integrate import quad as scint
5  import scipy.optimize as sco
6
7  %matplotlib inline
8  plt.rcParams['figure.figsize'] = [10, 8]
9  plt.rcParams.update({   'font.size': 22,
10   "font.family": "serif",
11   "font.serif": ["CMU serif"],})
12 plt.rc('axes', unicode_minus=False)
13 plt.rc('pgf', texsystem='pdflatex')
14 kb = const.physical_constants["Boltzmann constant in eV/K"][0]*1
    e3  #in meV/K
15 hbar = 6.58211951 * 10**(-13)  #in meV . sec
```

Define a function 'deltaTsim' that simulates Δ v/s T curve according to the equation . 'deltaint' performs the integration given a T, Delta and $\hbar\omega$ value. We employ another function to numerically solve the equation for Δ by giving RHS as 'deltaint'.

Thus we effectively get the Δ v/s T values.

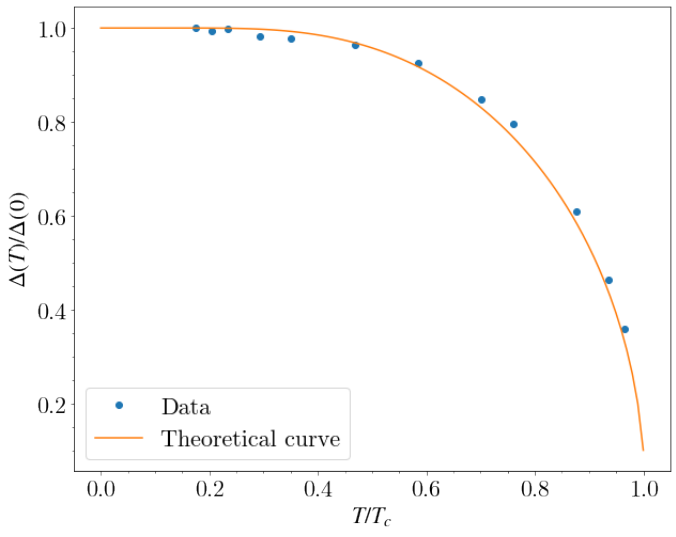
```
def deltaTsim(Tc,eta,Tarr):
#E in meV , Delta in meV, eta is inverse interaction strength, tc
  in kelvin
E = (kb * Tc)/(1.13*np.exp(-eta))


def deltaint(T,Delta,E):
integ = lambda x : (np.tanh(0.5/(kb*T)*(x**2+Delta**2)**0.5))/(x
 **2+Delta**2)**0.5
return scint(integ,0,E)[0]

def sim(E,Tarr):
fit = np.zeros_like(Tarr)
for i,T in enumerate(Tarr):
func = lambda Delta : [deltaint(T,Delta,E) - eta]
fit[i] = sco.fsolve(func, [0.05])
return fit


return sim(E,Tarr)

Tc =14.5
N =100
eta = 1/0.33 #inverse interaction strength
Tarr = np.linspace(0,Tc,N)
simdata = deltaTsim(Tc,eta,Tarr)


plt.plot(Tarr,simdata,'k')
plt.xlabel('T')
plt.ylabel('$\Delta(T)$')
# plt.savefig('/mnt/localdiskd/Semester 8/Project -
 Superconductivity/Report/images/deltavstcurveintro.pdf')
```



Finally we import the data we extracted from tunnel junction measurements and

set a critical temperature to try and match our obtained values with the predicted curve.

```
plt.rcParams['mathtext.fontset'] = 'stix'

data = np.transpose(np.loadtxt('/mnt/localdiskd/Semester 8/
 Project - Superconductivity/Code/CBS/data/
 Tunneljunction_NbNAloAg_29032022/TGamDelta.txt',usecols=(0,2),
 skiprows=1))

def fitdeltaT(data,eta):
x, y = data[0], data[1]


funct = lambda T,Tc : deltaTsim(Tc, eta, T)
fitpar = sco.curve_fit(funct , x, y)


return fitpar


Tc =16.8
N =100
eta = 3.03
Tarr = np.linspace(0,Tc,N)
fitdata = deltaTsim(Tc,eta,Tarr)

plt.plot(data[0]/Tc,data[1]/data[1][0],'o',label='Data')
plt.plot(Tarr/Tc,fitdata/fitdata[0],label='Theoretical curve')
plt.legend()
plt.minorticks_on()
plt.xlabel('$T/T_c$')
plt.ylabel('$\Delta(T)/\Delta(0)$')
plt.savefig('/mnt/localdiskd/Semester 8/Project -
 Superconductivity/Report/images/deltavstcurvetun.pdf')
```



A.3 Maki Analysis of tunneling data in magnetic fields

A.3.1 With Taylor expansion

The code below is written with taylor expansion for the 'b' part of the equation (5) in [8]. We define various functions such as the differentiated Fermi function 'Dfermi', density of states function 'dos', a function to numerically solve for the non-linear problem of u_+, u_- where (invalid) taylor expansions are used, a function 'upum' to vectorize the numerical solver, and finally a 'dI_dV' function to combine and integrate everything to get conductance curves. Different sections of the code are highlighted and produce different outputs such as the u_+ and u_- values , dI/dV values where a parallel computation is implemented for faster processing, and an interpolation section for kinks near zero bias, which arise when $\Gamma \neq 0$.

```
from pyexpat import error
import numpy as np
import scipy as sc
import scipy.constants as const
import scipy.optimize as sco
import scipy.integrate as scint
from joblib import Parallel, delayed
import multiprocessing
import time


kb = const.physical_constants["Boltzmann constant in eV/K"][0]*1e3
     #total value in milli eV
mu = 0.057883818012
pi = const.pi

def Dfermi(E,V,T):
# return (1/(np.cosh((E+V)/(2*kb*T)) )**2 * (4*kb*T))
return (-np.exp((E+V)/(kb*T))/(kb*T*(np.exp((E+V)/(kb*T))+1)**2))

def dos(E,Gam,Delta,xi,b,H,Pu,Pd):

up,um = upum((E),Gam,Delta,xi,b,H)
# print(up,um)
return 1/2 * np.sign(E) * np.real( Pu*(up)/(up**2 - 1)**0.5 + Pd*(
    um)/(um**2 - 1)**0.5)


def u_solve(E,Gam,Delta,xi,b,H):
#x1 -> Re(eqn 5)-Re(u+), x2 -> Im(eqn 5) - Im(u+), y1 -> Re(Eqn5)-
    Re(u-), y2 ->Im(eqn 5) - Im (u-)
# x is a vector containing [ Re(u+), Im(u+), Re(u-), Im(u-)]

#Scipy solution without transform with binomial
def utemp(x):
x1 = lambda x : (E-mu*H)/Delta + xi*np.real(complex(x[0],x[1]) /
    (1-complex(x[0],x[1])**2)**0.5) +b*np.real((complex(x[2],x[3])-
    complex(x[0],x[1]))*(1+1/2*(complex(x[2],x[3]))**2 )) - x[0]
```

```
34 x2 = lambda x : (Gam)/Delta + xi*np.imag(complex(x[0],x[1]) / (1-
       complex(x[0],x[1])**2)**0.5) +b*np.imag((complex(x[2],x[3])-
       complex(x[0],x[1]))*(1+1/2*(complex(x[2],x[3]))**2 )) - x[1]
35 y1 = lambda x : (E+mu*H)/Delta + xi*np.real(complex(x[2],x[3]) /
       (1-complex(x[2],x[3])**2)**0.5) +b*np.real((complex(x[0],x[1])-
       complex(x[2],x[3]))*(1+1/2*(complex(x[0],x[1]))**2 )) - x[2]
36 y2 = lambda x : (Gam)/Delta + xi*np.imag(complex(x[2],x[3]) / (1-
       complex(x[2],x[3])**2)**0.5) +b*np.imag((complex(x[0],x[1])-
       complex(x[2],x[3]))*(1+1/2*(complex(x[0],x[1]))**2 )) - x[3]
37 return [x1(x),x2(x),y1(x),y2(x)]
38 temp = sco.root(utemp, [0.02,0.02,0.02,0.02], method='lm').x
39 return [temp[0],temp[1],temp[2],temp[3]]
40
41 def upum(E,Gam,Delta,xi,b,H):
42 tmp = u_solve(E,Gam,Delta,xi,b,H)
43 return (complex(tmp[0],tmp[1]),complex(tmp[2],tmp[3]))
44
45
46 def fermi(E,V,T):
47 return 1/(np.exp((E+V)/(kb*T))+1)
48
49
50 def dI_dV(V,Gam,T,Delta,xi,b,H,P):
51
52 Pu = (1+P)/np.sqrt(2+2*P**2)
53 Pd = ((1-P**2)/np.sqrt(2+2*P**2))/(1+P)
54 tmp2 = lambda E : dos(E,Gam,Delta,xi,b,H,Pu,Pd) * Dfermi(E,V,T)
55 return scint.quad(tmp2,-10,10)[0]
56
57
58
59 # ================= u_p, u_m ====================================
60
61 # %%
62
63 E = np.linspace(-3.5, 3.5, 200)
64 # Gam = 0.0
65 # T = 0.245
66 # Delta = 0.39
67 # xi = 0.0
68 # b = 0.05
69 # H = 3
70
71 T = 0.31
72 Delta = 0.39
73 P = 0.72
74 H = 3
75
76 Gam = 0.00
77 xi = 0.024
78 b = 0.05
79
80
81 sol = [u_solve(i, Gam=Gam, Delta=Delta, xi=xi, b=b, H=H) for i in E
       ]
82 sol = np.array(sol)
```

```
# print(sol)
plt.plot(E, np.abs(sol[:, 0]))
plt.show()
plt.plot(E, np.abs(sol[:, 1]))
plt.show()

#============================================================
# %%

# LOAD DATA:
path = './data/5nmnbnandt0_28k'
file = open(f'{path}/0T.csv', 'r')
data1 = np.transpose(np.loadtxt(file, delimiter=','))
data1[1] = data1[1]/data1[1][0]


# ============== dI/dV=================================
# %%

# file = open('./TIFR/temp/1.txt','w')
n = 100
V = np.linspace(-1, 1, n)

T = 0.31
Delta = 0.39
P = 0.72
H = 3

Gam = 0.00
xi = 0.024
b = 0.05

# file.write(f'Gam={Gam} T={T} Delta={Delta} xi={xi} b={b} H={H} \n
    (Relevant values in meV and Tesla) \n V (meV)      dI/dV \n')


stime = time.time()
# didv = [dI_dV(i,Gam,T,Delta,xi,b,H,P) for i in V]


def conloop(i):
return dI_dV(i, Gam, T, Delta, xi, b, H, P)


num_cores = multiprocessing.cpu_count()
didv = Parallel(n_jobs=num_cores-1)(delayed(conloop)(i) for i in V)

DIDV = np.array(didv)
DIDV = DIDV/(DIDV[0])
print(time.time()-stime, " seconds elapsed.")
# np.savetxt(file, np.transpose(np.vstack((V,DIDV))))

plt.plot(
V, DIDV, label=f' b={b} ,\n $\\Delta$={Delta},\n $\\Gamma$={Gam},\n
    $\\zeta$={xi},\n H={H}')

```

```
# plt.plot(data1[0], data1[1], '--k', label='expt')
# plt.title(f'H={H} Tesla - 10nm , T={T} Kelvin')
plt.legend()
plt.minorticks_on()
plt.grid(which='minor')
plt.show()
# plt.savefig(f'./plots/G{Gam}T{T}D{Delta}Z{xi}b{b}H{H}.png')

# %%

# =====interpolation for V near zero==========

olim = 1.45  # outer limit
ilim = 0.05

vvalmask = ((V > -olim)*(V < -ilim)+(V < olim)*(V > ilim))
vvals = V*(~vvalmask)
nDiDv = (~vvalmask)*DIDV

vvals = vvals[vvals != 0]
nDiDv = nDiDv[nDiDv != 0]

inter = interp1d(vvals, nDiDv, kind='quadratic')
plt.plot(V, inter(V), '-k',
label=f' b={b} ,\n $\\Delta$={Delta},\n $\\Gamma$={Gam},\n $\\zeta$
    ={xi},\n H={H}')
# plt.plot(vvals,nDiDv,'o')
plt.title(f'H={H} Tesla - 10nm , T={T} Kelvin')
plt.plot(data1[0], data1[1], '--b', label='expt')
plt.legend()

plt.show()
```

A.3.2 Corrected code

The code below uses equation 77 from [7]:

$$\varepsilon y_2 + y_1 + y_4 h^2 - \frac{\alpha_{\text{orb}}}{\pi}\left(y_1 y_2 - y_3 y_4 h^2\right) = 0$$

$$\varepsilon y_4 + y_2 - y_3 + \frac{\alpha_{\text{orb}}}{\pi}\left(y_2 y_3 - y_1 y_4\right)$$

$$-\frac{2}{3\tau_{\text{s.o}}\pi}\left(y_1 y_4 + y_2 y_3\right) = 0$$

$$y_1^2 - y_2^2|\Delta|^2 + y_3^2 h^2 - y_4^2 h^2 |\Delta|^2 = -\pi^2$$

$$y_1 y_3 + y_2 y_4 |\Delta|^2 = 0$$

Using the following substitutions:

$$y_1 \pm h y_3 = -\pi u_{\pm}\left(1 - u_{\pm}^2\right)^{-1/2}$$

$$|\Delta|\left(y_2 \mp h y_4\right) = \pi\left(1 - u_{\pm}^2\right)^{-1/2}$$

u_+ and u_- can now be solved numerically. The implementation is shown below:

```
"""
Created on Thu May  5 19:13:52 2022

@author: gaurav
"""
import numpy as np
import scipy.constants as const
import matplotlib.pyplot as plt
import scipy.integrate as scint
from joblib import Parallel, delayed
import multiprocessing
import time


kb = const.physical_constants["Boltzmann constant in eV/K"][0]*1e3   #total value in milli eV
mu = 0.057883818012
pi = const.pi



def upum(E,Gam,Delta,xi,b,H):
mu = 0.057883818012
pi = const.pi
h2 = (mu*H)**2
xi = np.sign(E)*xi
def jacinv(y1,y2,y3,y4):
jac = np.array([[1-Delta*xi/pi * y2, complex(E,Gam)-Delta*xi/pi*y1, Delta*xi/pi*y4*h2, h2+Delta*xi/pi*h2*y3],
[-Delta*xi/pi *y4+2*b*Delta*y4/pi, Delta*xi/pi *y3 +1+2*b*Delta*y3/pi, -1+Delta*xi*y2/pi+2*b*Delta*y2/pi, complex(E,Gam)-Delta*xi*y1/pi+2*b*Delta*y1/pi],
[2*y1,-Delta**2 * 2*y2, 2*h2*y3, -2*h2*Delta**2 *y4],
[y3,y4*Delta**2,y1,Delta**2 *y2]])
return np.linalg.inv(jac)
# return jac

tp1 = lambda y1,y2,y3,y4 : complex(E,Gam)*y2+y1 + y4*h2 - Delta*xi/pi * (y1*y2-y3*y4*h2)
tp2 = lambda y1,y2,y3,y4 : complex(E,Gam)*y4+y2 -y3 + Delta*xi/pi * (y3*y2-y1*y4) +2*b*Delta/pi * (y1*y4+y2*y3)
tp3 = lambda y1,y2,y3,y4 : y1**2 - y2**2 * Delta**2 + y3**2 *h2 - y4**2 * h2 * Delta**2 +pi**2
tp4 = lambda y1,y2,y3,y4 : y1*y3 + y2*y4*Delta**2
y1,y2,y3,y4 = complex(0.03,-0.01),complex(0.375,0.6),complex(-0.1,7.0),complex(-0.4,-0.5)

err=1
while (err>1e-12):
sol = [y1,y2,y3,y4] - 0.9*(jacinv(y1, y2, y3, y4)) @ [tp1(y1,y2,y3,y4),tp2(y1,y2,y3,y4),tp3(y1,y2,y3,y4),tp4(y1,y2,y3,y4)]
y1 = sol[0]; y2=sol[1];y3=sol[2];y4=sol[3]
err = np.abs(tp1(*sol))+np.abs(tp2(*sol))+np.abs(tp3(*sol))+np.abs(tp4(*sol))

up = (y1+mu*H*y3)/(-pi*Delta*(y2-mu*H*y4))
```

```
47  um = (y1-mu*H*y3)/(-pi*Delta*(y2+mu*H*y4))
48
49  return up,um
50
51
52  def Dfermi(E,V,T):
53  return (-np.exp((E+V)/(kb*T))/(kb*T*(np.exp((E+V)/(kb*T))+1)**2))
54
55  def dos(E,Gam,Delta,xi,b,H,P):
56  Pu = (1+P)/2
57  Pd=(1-P)/2
58  up,um = upum(E,Gam,Delta,xi,b,H)
59  dosd = 0.5 * np.sign(E) * np.real( Pd*(up)/(up**2 - 1)**0.5)
60  dosu = 0.5 * np.sign(E) * np.real( Pu*(um)/(um**2 - 1)**0.5)
61
62  return dosu,dosd
63
64
65  def dI_dV(V,Gam,T,Delta,xi,b,H,P):
66
67  def tmp2(E):
68  tpp = dos(E,Gam,Delta,xi,b,H,P)
69  return (tpp[0]+tpp[1]) * Dfermi(E,V,T)
70
71  return scint.quad(tmp2,-10,10,limit=500)[0]
72
73
74
75  #Parameters=====================================================
76
77  Delta = 0.39
78  P = 0.72
79  H = 3
80  T = 0.31
81
82  Gam = 0.0
83  xi = 0.024
84  b = 0.05
85
86
87
88
89  # # dIdV sim=====================================================
90  stime = time.time()
91  # def conloop(i):
92  #     return dI_dV(i, Gam, T, Delta, xi, b, H, P)
93
94
95  # V = np.linspace(-1.5,1.5,300)
96  # num_cores = multiprocessing.cpu_count()
97  # didv = Parallel(n_jobs=num_cores-1)(delayed(conloop)(i) for i
     in V)
98  # DIDV = np.array(didv)
99  # DIDV = DIDV/(DIDV[0])
100
101
```

```
102
103 # plt.plot(V,DIDV)
104 # plt.title('didv')
105 # plt.show()
106 # #============================================================
107
108
109 # # dos=========================================================
110
111 E1 = np.linspace(-1.5,1.5,300)
112 dos = [dos(i,Gam,Delta,xi,b,H,P) for i in E1]
113 dos=np.array(dos)
114
115
116 plt.plot(E1,dos[:,0],label='down')
117 plt.plot(E1,dos[:,1],label='up')
118 plt.legend()
119 plt.title('dos')
120 plt.show()
121 # #============================================================
122
123
124 #=========upum=================================================
125 # E1 = np.linspace(-1.5,1.5,300)
126 # y1 = np.array([upum(i,Gam,Delta,xi,b,H) for i in E1])
127
128 # upr = np.abs(np.real(y1[:,0]))
129 # upi = np.abs(np.imag(y1[:,0]))
130 # umr = np.abs(np.real(y1[:,1]))
131 # umi = np.abs(np.imag(y1[:,1]))
132
133 # plt.plot(E1,upi,label=f'Gam={Gam},xi={xi},b={b},H={H},Delta={
    Delta}')
134 # plt.legend()
135 # plt.title('upum')
136 # plt.show()
137 #============================================================
138 print(time.time()-stime, " seconds elapsed.")
```

References

[1] J. Bardeen, L. N. Cooper, and J. R. Schrieffer. "Theory of Superconductivity". In: *Physical Review* 108.5 (Dec. 1957), pp. 1175–1204. DOI: 10.1103/physrev.108.1175. URL: https://doi.org/10.1103/physrev.108.1175.

[2] Michael Tinkham. *Introduction to superconductivity*. Mcgraw Hill, 1996.

[3] Ivar Giaever. "Energy Gap in Superconductors Measured by Electron Tunneling". In: *Physical Review Letters* 5.4 (Aug. 1960), pp. 147–148. DOI: 10.1103/physrevlett.5.147. URL: https://doi.org/10.1103/physrevlett.5.147.

[4] Alistair C Rose-Innes and E H Rhoderick. *Introduction to superconductivity*. Oxford Pergamon Press, 1994.

[5] R. C. Dynes, V. Narayanamurti, and J. P. Garno. "Direct Measurement of Quasiparticle-Lifetime Broadening in a Strong-Coupled Superconductor". In: *Physical Review Letters* 41.21 (Nov. 1978), pp. 1509–1512. DOI: 10.1103/physrevlett.41.1509. URL: https://doi.org/10.1103/physrevlett.41.1509.

[6] R. Meservey and P.M. Tedrow. "Spin-polarized electron tunneling". In: *Physics Reports* 238.4 (Mar. 1994), pp. 173–243. DOI: 10.1016/0370-1573(94)90105-8. URL: https://doi.org/10.1016/0370-1573(94)90105-8.

[7] J. A. X. Alexander et al. "Theory of Fermi-liquid effects in high-field tunneling". In: *Physical Review B* 31 (May 1985), pp. 5811–5825. DOI: 10.1103/physrevb.31.5811. (Visited on 04/22/2022).

[8] D. C. Worledge and T. H. Geballe. "Maki analysis of spin-polarized tunneling in an oxide ferromagnet". In: *Physical Review B* 62 (July 2000), pp. 447–451. DOI: 10.1103/physrevb.62.447. (Visited on 04/22/2022).