The Impact of Diffusion and Disorder on the Single Photon Response of β-Ta OKIDs

Kian Peter Syaifoel

TU Delft

Preface

The element Tantalum, which is central to this thesis, was named after the mythical figure Tantalus. According to legend, he was greedy and tried to deceive the Olympian gods. In punishment for his actions, Tantalus was doomed to eternal hunger in Tartarus, where the fruits above his head and the water below his chin would forever retreat just outside his reach. From this story we have phrases such as 'tantalizing' and 'Tantalean punishment'. (NL: 'Tantaluskwelling')

Tantalum was discovered and named by Anders Ekeberg, after the tantalizing task of dissolving its naturally occurring oxides and isolating them from the chemically similar Niobium that occurs in the same ores. If Ekeberg had been working on optical MKIDs, I suspect he might have decided on the same name for Tantalum. Trying to use Tantalum as absorptive element in MKIDs namely mirrors Tantalus' punishment: Eager to capture more photons, the MKID's desired energy resolution retreats out of reach.

With the many iterations needed to make progress in any scientific effort, I would argue that scientific work is more of a 'Sisyphean task'. I hope the next person to haul this particular boulder up the hill is able to draw some useful ideas from my attempt at it.

I would like to thank everyone who supported me practically, emotionally or otherwise in my efforts in this project and in my education as a whole. This includes my friends; my partner and my family, whom I always feel at ease with; and my supervisors, who gave much appreciated advice and patience w.r.t. focus and planning. I will fondly look back on my time here.

The Impact of Diffusion and Disorder on the Single Photon Response of β-Ta OKIDs

Thesis for the completion of the M.Sc. degree in Applied Physics

by
Kian Peter Syaifoel
Student no. 4726502

conducted during
February 2024 - October 2024

assessed by

Daily Supervisor	ir. S.A.H. de Rooij
Supervisor	dr. ir. P.J. de Visser
Responsible Thesis Supervisor	prof. dr. ir. J.J.A. Baselmans
Independent Examiner	prof. dr. Y.M. Blanter

defended on
November 5th 2024

at

TU Delft

Abstract

Optical kinetic inductance detectors (OKIDs) are a type of superconducting single photon detector whose signal corresponds to the number of quasiparticles that are excited when the detector absorbs a photon. OKIDs whose absorptive element is made out of β-phase Tantalum (β-Ta), a disordered superconductor, exhibit faster than exponential (supra-exponential) decay of its pulse-shaped single photon response. Faster decay generally corresponds to a higher density of quasiparticles. These pulses cannot be explained by the equations typically used to describe how these quasiparticles decay in a process called recombination. One possible explanation is that the high degree of disorder of β-Ta causes the diffusion of quasiparticles through the material to be slow, which leads to a high local density of quasiparticles, causing the faster decay. The usual model for this recombination does not include this diffusion.

In this thesis, we model the OKID's single photon pulse with a local model of the recombination process, that does include this diffusion. To this end, we analyze the downconversion process, which describes the generation of particles due to a photon absorption. This analysis therefore gives information on the initial conditions of the recombination process. Next, we study and solve this diffusion and recombination model.

Next, we test the response pulse predicted by this model against measurement data of the single photon pulses of a β-Ta OKID, for different photon energies. We show that the model must assume that an unfeasibly high number of quasiparticles is excited in order to reproduce the data. Furthermore, the pulses generated by the model exhibit a dependence on the absorbed photon energy, which is not observed in the data. Therefore we reason that the model must be missing some photon energy dependent effect. We reason that this energy dependence could lie in the downconversion process, or that it could lie in the responsivity of the OKID.

We also investigate the pulses of OKIDs that are designed with an absorber composed of multiple small elements of β-Ta, which constrain the quasiparticles in space. We measure the correspondence of the size of the β-Ta elements and the decay time of the response pulse. We show that the decay time increases for larger elements, until it saturates. This behavior is also predicted by our model. We therefore conclude that the diffusion indeed plays a significant role in the cause of supra-exponential decay.

Contents

Chapter 1

Introduction

Central to this thesis is a type of superconducting device called a microwave kinetic inductance detector (MKID). One potential application of these devices is as a single-photon, energy resolving detector for visible to near-IR light. Such an Optical MKID (OKID), could be applied to perform spectrometry on extremely faint sources, e.g. to search for signs of life in the atmospheres of exoplanets. For this to be feasible the OKID ideally absorbs as many incident photons as possible, and it ideally has a high 'Resolving Power'. Resolving power (R) is a measure of how well the device can tell apart photons of different energy. Exciting the OKID with a known photon energy, R is defined as the average measured photon energy divided by the deviation in the measurements.

Currently, the OKIDs with highest R use aluminum as absorber of the photons. Aluminum is a well-understood superconductor. However, this material has a low normal-state resistivity, which makes for a high reflectivity. This limits the number of incident photons actually absorbed. Therefore β-phase tantalum (β-Ta), a superconductor with a higher normal-state resistivity, is being researched. Its higher resistivity corresponds to a higher absorption of photons[1, §2.5.2]. However, it also corresponds to a higher degree of disorder of the material[2]. 'Disorder' denotes a high presence of impurities, grain boundaries or defects in the material's structure, which electrons and phonons can scatter on.

For OKIDs based on β-Ta it was found that the device's energy resolution is limited by an unexpectedly quick decay of the signal compared to Al devices[3], see figure 1.1. A shorter pulse makes for a more difficult distinction between photon energies. To illustrate, at wavelengths $1545nm$ to $402nm$, β-Ta OKIDs reach a resolving power of only 4.5 to 6[3], whereas Al OKIDs on the same substrate reach a resolving power of 10 to 21[4]. The best performing Al OKIDs, from the same paper, are suspended on a membrane and reach $R = 19$ to 52.

The single photon response of an MKID relates to the number of excitations (quasiparticles) created by the absorbed photon[5, §2.2.3]. The signal decays as the created quasiparticles undergo a process called recombination. In this process they return to the ground state, in which the particles are bonded in pairs. This process continues until the equilibrium state of the material is reached again. As the quasiparticles must 'pair up', the rate of recombination depends on the density of quasiparticles[5, eq.2.29]. In figure 1.1 we saw that the Al OKID's signal decays exponentially, while the β-Ta OKID's signal initially decays faster than exponentially. For a situation where the excess number of quasiparticles is small compared to the equilibrium amount, an exponential decay is expected[6].

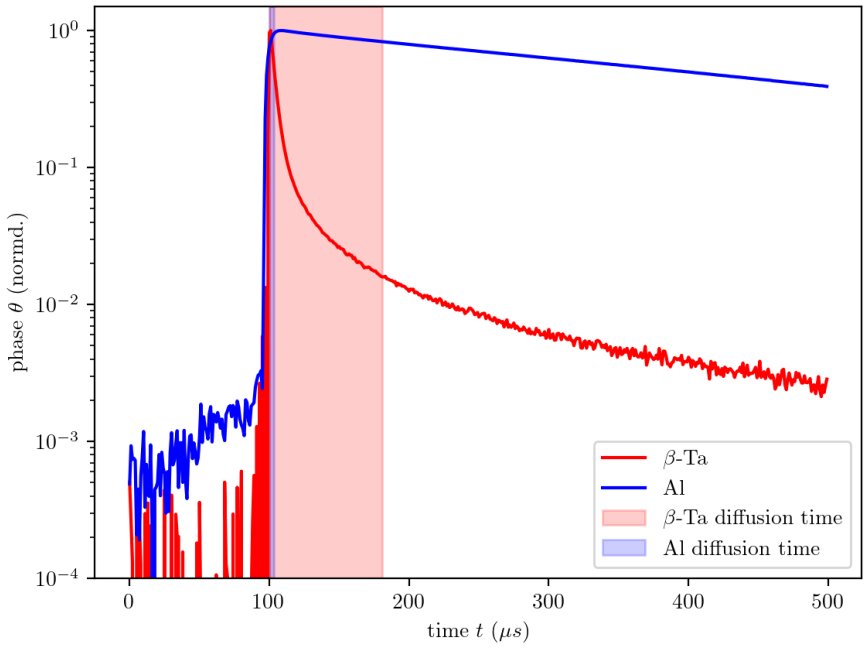


Figure 1.1: Averaged phase response of a $402nm$ photon absorption for a β-Ta and Al OKID. The maxima are normalized to 1rad for comparison. The shaded areas show the expected time of an electron to diffuse across the length of the corresponding Al or β-Ta absorber. The diffusion constants D of Al and β-Ta were measured to be 150 and $0.5cm^2/s$ resp. For 1D diffusion along the length ℓ of the detectors (310 and $90\mu m$ resp.), the shown diffusion time is calculated according to $\sqrt{MSD(t)} = \sqrt{2Dt} = \ell$

Remember that β-Ta was chosen for its higher absorption and that that corresponds to a higher resistivity. The resistivity, in turn, relates to slower diffusivity of particles within the material[7, §3.4.3]. A probable explanation for the faster-than-expected recombination is then that the quasiparticles, which are created in an area close to the photon absorption, are slow to diffuse throughout the β-Ta compared to the Al. Slow diffusion would locally create a higher density of quasiparticles and thus initially quicker recombination. The shaded areas in figure 1.1 show the negligible diffusion time needed for quasiparticles to spread across the Al OKID, and the significant diffusion time needed to spread across the β-Ta OKID.

In this thesis we test the hypothesis that the slower diffusion hypothesis causes the faster decay. We do so by building and testing a model of an MKID's single photon response, based on the diffusion and quasiparticle recombination occurring in the MKID. To do so, we first look into the downconversion process, which describes the excitation of quasiparticles due to a photon absorption. Then, we analytically and computationally model the processes that govern the recombination of quasiparticles. Lastly, we experimentally test the limitations of the model, compared to measured pulses. We will also, experimentally and using the model, observe how lowering the distance of β-Ta, over which quasiparticles can diffuse, contributes to the signal's decay.

We schematically present the structure of this thesis in table 1.1. Here we show the steps we take to produce a model of an MKID pulse, and to test it with actual measurement data. The main content chapters of this thesis (Ch.3, 4 and 5) encompass these steps. In addition to these content chapters, the thesis includes this chapter as introduction, an overview of preliminary theory in chapter 2, and an overview of conclusions made for each step in chapter 6.

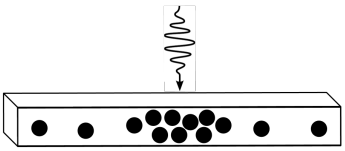
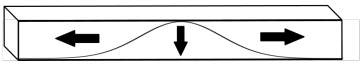
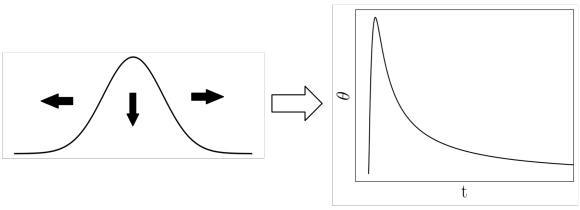
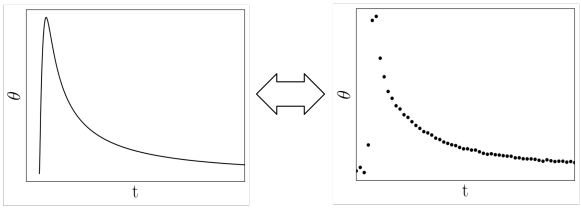
Subject	Description
 Downconversion	**Motivation:** 'Downconversion' is the process by which a single photon absorption leads to the excitation of many low energy quasiparticles. We investigate this process for disordered materials such as β-Ta in order to find out which initial conditions should be taken for the recombination process that follows it. **Sections:** Ch.3, App.A, App.B
 Recombination	**Motivation:** The recombination process entails how the created quasiparticles recombine to Cooper pairs, returning the material to equilibrium. We build and solve a diffusion-dependent model for this recombination process, in order to assess whether this addition explains the behavior of β-Ta MKIDs. **Sections:** Ch.4, App.C
 MKID Response	**Motivation:** The recombination model tells us how the distribution of quasiparticles changes in time. This must be converted to the MKID response that we can measure. We must include the 'ringing' time of the MKID oscillator as well as its responsivity. **Sections:** §2.3, §4.4.1, §5.3.2
 Evaluation w.r.t. Data	**Motivation:** In order to test the 'diffusion hypothesis' we compare the behavior of the produced model with measurement data of β-Ta MKIDs. **Sections:** Ch.5, App.D, App.E

Table 1.1: Overview of the subjects we cover, a short explanation of each, and the associated sections of the thesis.

Chapter 2

Preliminary Theory on Superconductivity and MKIDs

This chapter gives a short overview of theory and concepts used in the rest of this thesis. Kinetic inductance detectors (KIDs), first described in [8], rely on superconductivity for their detection mechanism. This type of detector owes its high sensitivity to the small binding energy of Cooper pairs in a superconductor[5, §1.1]. Breaking these Cooper pairs changes the conductivity of the material, which offers an elegant manner to read out a response when the material is incorporated into a resonator. We will now cover basic theory to describe these excitations and how they affect the MKIDs' conductive properties and thereby the observed signal.

2.1 Static Properties of Superconductors

The first microscopic theory which was able to explain superconductive properties was the Bardeen-Cooper-Schrieffer (BCS) theory, which pivots upon the existence of Cooper pairs, a bound state of electrons. The postulated mechanism behind this bonding is that each electron induces deformations in the surrounding structure of (positive) nuclei, which can indirectly attract another electron. It was proven by Cooper that any arbitrarily small binding energy between electrons allows for the existence of bound states, in which two electrons of momentum larger than the Fermi momentum k_F can still have a favorable combined energy, lower than twice the Fermi energy $2E_F$[1, §3.1]. At zero temperature, one can expect all the conduction electrons to form Cooper pairs, condensing into a macroscopic ground state. With these ideas, BCS theory provides a mechanism to explain many aspects of superconductivity[5, §2.1].

The interactions of Cooper pairs with thermal phonons or incident photons induce excitations of the ground state. These excitations, corresponding to the breaking of Cooper pairs, create so-called Bogoliubov quasiparticles which behave akin to charge carriers in a normal conductor. The thermal equilibrium quasiparticles density, at finite temperature T is given by [5, §2.1]

$$n_{qp} = 4N_0 \int_0^\infty N_s(E) f(E) dE \approx 2N_0 \sqrt{2\pi k_B T \Delta} \exp\left(-\Delta/k_B T\right), \tag{2.1}$$

which is calculated from the quasiparticles' density of states $N_s(E)$ and their Fermi-Dirac distribution $f(E)$. This is schematically displayed in figure 2.1. N_0 is the density of states at the Fermi level, k_B is the Boltzmann constant, and Δ is the superconductor's energy gap. According to BCS theory, at low temperatures Δ is related to the material's critical temperature by

$$\Delta = 1.764 k_B T_c. \tag{2.2}$$

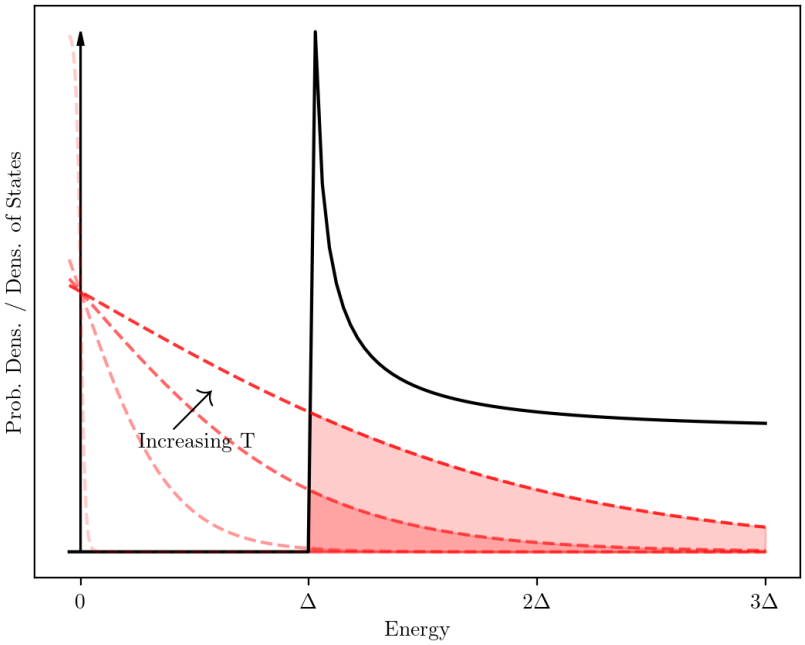


Figure 2.1: Illustration of the density of states (black) and the Fermi-Dirac distribution (red) of the quasiparticles in a superconductor. For $T \to 0$ all electrons are paired into Cooper pairs, and the superconductor is in its ground state. At finite temperatures, more and more quasiparticles will occupy the states above the energy gap Δ.

2.2 Dynamic Properties of Superconductors

On its own, BCS cannot explain dynamic effects such as conductivity under AC conditions, which we need in the description of a resonator. For this we can apply an extension of BCS theory developed by Mattis and Bardeen. First let us look at a qualitative description. When describing the total conductivity of a superconductor, one should consider both the current carried by the superconducting condensate, as well as the current carried by the quasiparticles. This two-fluid system can be described qualitatively with an adaptation of the Drude model of conductivity[1, §2.5.1]. Instead of a single species of charge carrier, one then considers two parallel types of charge carriers. These can be viewed as one type with a finite and one with an infinite mean free path, corresponding to the quasiparticles and the Cooper pairs respectively. Equation 2.3 gives the combined conductivity predicted by this Drude-like model.

$$\begin{aligned} \sigma(\omega) &= \sigma_1(\omega) - i\sigma_2(\omega) \\ \sigma_1(\omega) &= \frac{\pi n_s e^2}{2m}\delta(\omega) + \frac{n_n e^2 \tau_n}{m} \\ \sigma_2(\omega) &= \frac{n_s e^2}{m\omega} \end{aligned} \tag{2.3}$$

ω is the angular frequency of the current. n_s and n_n are respectively the particle density of superconducting charge carriers and normal charge carriers. e and m are the charge and mass of the carriers, and τ_n is the mean scattering time for the normal charge carriers. The main takeaways from this model are that, for $\omega > 0$, the real part is finite, meaning that there is energy dissipation due to the quasiparticles; a resistive effect. Furthermore, there is an imaginary part which gives an inductive response in the material. This is called 'kinetic inductance', as it relates to the inertia of the Cooper pairs. Both effects relate to the number of corresponding charge carriers and therefore the number of broken Cooper pairs.

Now let us return to the quantitative theory of dynamic superconductivity by Mattis and Bardeen. From this theory, for $k_BT, \hbar\omega < 2\Delta$, the conductivity at thermal equilibrium can be approximated by[5, §2.2.2]

$$\begin{aligned}\frac{\sigma_1}{\sigma_N} &= \frac{4\Delta}{\hbar\omega}\exp\left(-\Delta/k_BT\right)\sinh\left(\frac{\hbar\omega}{2k_BT}\right)K_0\left(\frac{\hbar\omega}{2k_BT}\right)\\ \frac{\sigma_2}{\sigma_N} &= \frac{\pi\Delta}{\hbar\omega}\left[1-2\exp\left(-\Delta/k_BT\right)\exp\left(\frac{-\hbar\omega}{2k_BT}\right)I_0\left(\frac{\hbar\omega}{2k_BT}\right)\right].\end{aligned} \tag{2.4}$$

The result is given as a fraction of the normal-state conductivity σ_N. I_0 and K_0 are the modified Bessel functions of the first and second kind, and ω is the angular frequency of the applied AC field. When a photon absorption occurs, more Cooper pairs are broken and quasiparticles are created and we are no longer in thermal equilibrium. We will assume that we can describe the excited material as being in thermal equilibrium for some higher effective temperature. Then, in combination with equation 2.1, the dependence of the conductivity on a change in n_{qp} can be derived from these equations, giving[5, §2.2.2]:

$$\begin{aligned}\frac{d\sigma_1}{dn_{qp}} &\simeq \sigma_N\frac{1}{N_0\hbar\omega}\sqrt{\frac{2\Delta_0}{\pi k_BT}}\sinh\left(\frac{\hbar\omega}{2k_BT}\right)K_0\left(\frac{\hbar\omega}{2k_BT}\right)\\ \frac{d\sigma_2}{dn_{qp}} &\simeq \sigma_N\frac{-\pi}{2N_0\hbar\omega}\left[1+\sqrt{\frac{2\Delta_0}{\pi k_BT}}\exp\left(\frac{-\hbar\omega}{2k_BT}\right)I_0\left(\frac{\hbar\omega}{2k_BT}\right)\right]\end{aligned} \tag{2.5}$$

Together, these equations can thus describe a change in conductivity when an excess of quasiparticles is created, e.g. in the event of a photon absorption. Focusing on σ_2, for small variations in n_{qp} and for $T < T_c/3$, $d\sigma_2/dn_{qp}$ is approximately constant[5, §2.2.2].

2.3 MKID response

The dependence of a superconductor's conductive properties on its excitation can be harnessed by incorporating it in a device such as an LC circuit. Its resonance frequency and its quality factor are then linked to the superconductor's excitation. Coupling such a resonator to a read-out line produces a notch filter with a pass band determined by the resonator's frequency and quality factor. The low resistance in a superconducting resonator allows it to achieve a high quality factor. The resulting narrow band allows many such resonators, tuned to different frequencies, to be read out on one line. An example design of such a KID is given in figure 2.2.

For the film of superconductor that gets excited in such a circuit, the conductivity can be expressed in terms of surface impedance Z_s, which consists of a surface resistance R_s and a surface inductance L_s. These depend on the complex conductivity shown before. This is given by[5, §2.2.3]:

$$R_s + i\omega L_s = \sqrt{\frac{i\mu_0\omega}{\sigma}}\coth\left(\sqrt{i\omega\mu_0\sigma}d\right) \tag{2.6}$$

Here, μ_0 is the magnetic permeability of free space and d is the thickness of the film. Calculating the impedance of such a circuit, including contributions from the superconductivity as well as the geometry of the device, one can calculate the expected transmission S_{21} across the read-out line. Measuring S_{21} for a frequency sweep, one can find the resonance frequency at which to probe S_{21}. From this calibration one can also define a relative amplitude A and phase change θ, as observable response of the KID to stimulation, see [5, §3.3]. In the end, those observables depend on the change in quasiparticles approximately according to

$$\begin{aligned}\frac{dA}{dN_{qp}} &= -\frac{\alpha_k\beta Q}{\sigma_2 V}\frac{d\sigma_1}{dn_{qp}}\\ \frac{d\theta}{dN_{qp}} &= -\frac{\alpha_k\beta Q}{\sigma_2 V}\frac{d\sigma_2}{dn_{qp}}\end{aligned} \tag{2.7}$$

where N_{qp} is the total number of quasiparticles, α_k is the ratio of kinetic inductance to total inductance, $\beta = 1 + 2d/\lambda/\sinh(2d/\lambda)$, λ is the magnetic penetration depth, Q is the quality factor, and V is the volume. Since the losses, due to σ_1, are predominantly determined by the number of quasiparticles, and the kinetic inductance, due to σ_2, is predominantly determined by the number of Cooper pairs, A and θ can be considered probes for respectively the quasiparticle and Cooper pair response. In this thesis we will focus mainly on the phase response (θ) of the MKIDs. As we will address in a later chapter, we usually approximate the phase responsivity given in equation 2.7 to be constant. This is because for low effective temperature and assuming Q remains constant, we know firstly that σ_2 varies only slowly, such that $-\alpha_k \beta Q/\sigma_2 V$ is approximately constant, and secondly that σ_2 varies approximately linearly, such that $d\sigma_2/dn_{qp}$ is approximately constant[5, §2.2.1].

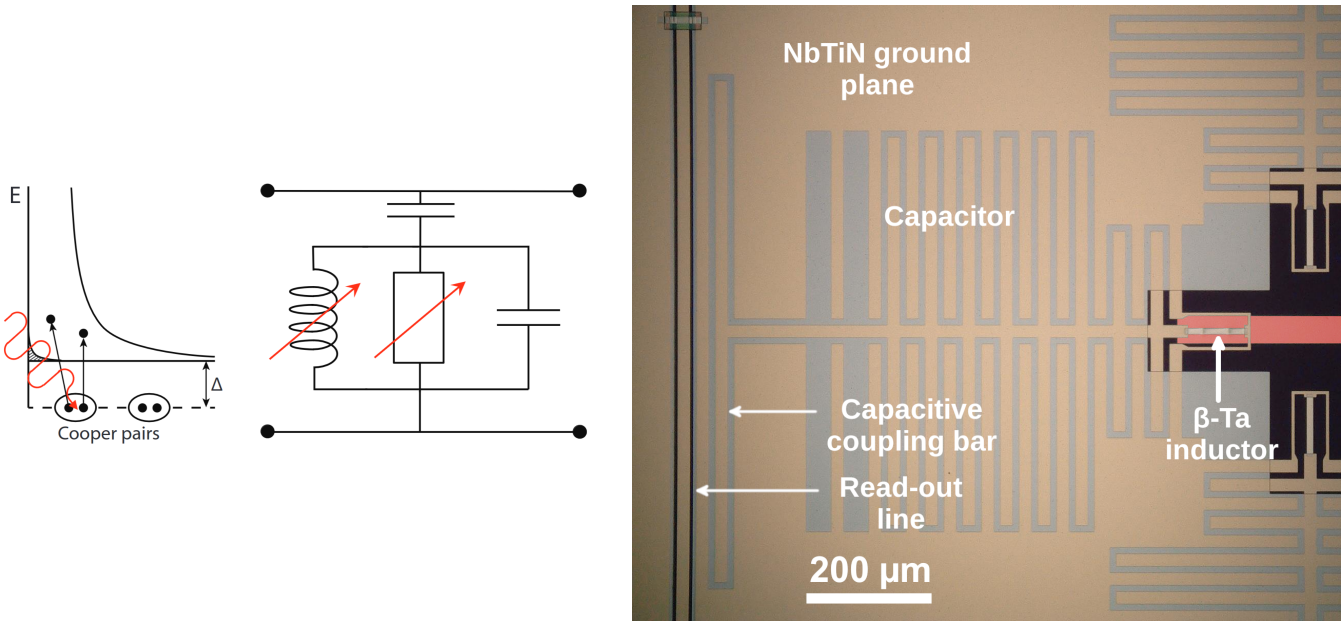


Figure 2.2: Shown (middle) is a circuit diagram of an MKID, in which the inductive and resistive components represent a strip of superconductor. As a photon breaks Cooper pairs and generates quasiparticles (left), this inductance and resistance increases. This figure was taken from [5]. On the right we present a physical example of an OKID based on a β-Ta absorber. The labels denote how the various parts of the design contribute to the shown circuit model.

Chapter 3

Downconversion Cascade

3.1 Introduction

We want to describe the decay of an MKID signal due to a single photon excitation. We have mentioned how this relates to the number of quasiparticles excited in the material. Before we describe the decay of excess quasiparticles, we first need to look into the generation of them. 'Downconversion' entails this process, in which the energy from an absorbed photon goes from being carried by a single particle, to being shared among a large distribution of particles. The aim of looking into this process is to determine the starting conditions of the recombination process that follows.

We will focus on two types of particle within the superconductor that are considered to play a role in this process. On the one hand we have the phonons, the particles that represent the modes of vibration which can be carried by the crystalline structure of a material. On the other hand we have the charge carriers in the superconductor. These come in the form of quasiparticles, or come bonded in Cooper pairs. Together, we will refer to these as electrons. The phonons and electrons interact with themselves, with each other, with imperfections in the material's structure, as well as the boundaries of the material. Through these interactions, the photon's energy, initially absorbed by a single electron, gets distributed among many of these phonons and electrons. As the energy per particle gradually decreases, different interactions between them are dominant to the process. In reference [9], Kozorezov et al. identify different stages in the downconversion process, based on which interactions are dominant at which energy.

We will summarize the process as they describe it and address its applicability to the downconversion process in a disordered superconductor such as β-Ta. We will complement this qualitative discussion with a comparison of calculations for β-Ta and for Al, an ordered superconductor. Lastly, we will use this information to determine the initial conditions or constraints in the modeling of the quasiparticle recombination process that comes after the downconversion.

3.2 Kozorezov's Downconversion Process

The pictures in figure 3.1 illustrate the stages of the downconversion as defined by Kozorezov. Let us cover these stages one by one in the subsections that follow.

3.2.1 Stage 1: $E_\lambda \rightarrow E^*$

The downconversion starts with the absorption of a photon by a single electron. The energy of the photon is labeled E_λ. Kozorezov states that the electron-electron (e-e) scattering times are below a femtosecond for the energies at this stage. By e-e scattering, the electron's energy gets distributed across several electrons. As the energy per particle decreases, the e-e scattering rate decreases. This stage

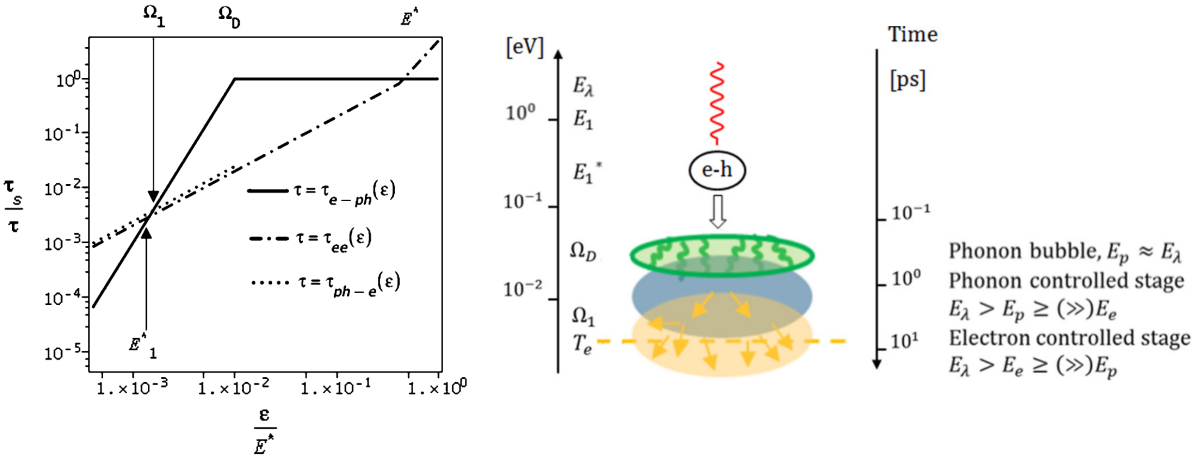


Figure 3.1: The graph on the left[10] plots the rates at which electrons scatter with themselves and with phonons. As the energy per particle decreases, different scattering processes become dominant, and the number of particles carrying the energy increases. The picture on the right[11] adds some interpretation to the former picture. It shows that the successive stages have increasing duration, and gives some properties of these stages. Initially a single electron carries the photon energy. When τ_{e-ph}^{-1} overtakes τ_{ee}^{-1}, many phonons at the Debye energy are created. Next the phonons have interactions among themselves. When τ_{ph-e}^{-1} overtakes τ_{e-ph}^{-1} many phonons scatter to excite electrons again.

is defined as lasting from the absorption until the e-e scattering rate dips below the electron-phonon (e-ph) scattering rate. The energy per particle at which this catch-up occurs is labeled E^*.

3.2.2 Stage 2: $E^* \rightarrow \Omega_D$

At this stage the e-ph scattering is the prevalent process. The electrons, still at rather high energy, excite phonons near the Debye energy Ω_D. Kozorezov argues that this stage has a short duration compared to the lifetime of the Debye phonons. Therefore he concludes that this stage results in a 'phonon bubble', i.e. a distribution of phonons that is energetically and spatially narrow. This state, modeled by a delta function, is taken as the initial condition for the next stage.

For stage 3 and 4, the evolution of the distributions of electrons and phonons are modelled using a pair of kinetic equations. Each differential equation models the respective particle distribution, their spatial evolution due to diffusion, the potential escape of particles and the coupling between the two species due to collisions. Under several assumptions and simplifications, which we will address in section 3.3, the behavior of the system can be solved for stage 3 and stage 4 separately.

3.2.3 Stage 3: $\Omega_D \rightarrow \Omega_1$

Given the high e-ph rates compared to ph-e rates at this stage, the time dependence of the aforementioned system of equations is dominated by the variation of the phonon distribution. In the course of this stage, the narrow distribution of Debye phonons broadens as more phonons are produced of lower energies. In e.g. reference [12], the resultant phonon distribution of this phase is taken to be a Debye distribution. The next stage commences as the e-ph rates are outpaced by the phonon-electron (ph-e) scattering rates. The energy per particle of this transition is labeled Ω_1.

3.2.4 Stage 4: $\Omega_1 \rightarrow 3\Delta$

In this phase, the ph-e scattering rate is dominant. This implies that the distribution of phonons will decrease, as the phonons excite electrons. Meanwhile, the electrons of themselves also undergo relaxation processes. A natural limit to define the end of this stage is given by the energy 3Δ, i.e. three

times the superconductor's band gap. This is namely the lowest energy at which an electron, when it decays to the band gap Δ, can still excite a phonon of 2Δ, the lowest energy capable of breaking a Cooper pair. Finally, we obtain a distribution of electrons just above the band gap. These low-energy excess electrons undergo recombination processes according to the Rothwarf-Taylor equations, which we will address in the next chapter.

3.3 Applicability to β-Ta

Kozorezov's model was construed for materials such as Aluminum and α-phase Tantalum, which are ordered superconductors. It is not a given whether the process as described by Kozorezov is directly applicable to β-Ta. β-Ta is namely a disordered material which may violate several of the assumptions used by Kozorezov. Let us address these concerns stage by stage. We will demonstrate that the specific assumptions and therefore the specific calculations by Kozorezov et al. are not directly applicable to β-Ta, which constrains their utility for us.

3.3.1 Stage 1: $E_\lambda \rightarrow E^*$

In this stage, Kozorezov argues that e-e collisions are so fast that the elastic collisions due to disorder are negligible. As such, he states that he can approximate the e-e scattering rate using the Landau-Pomeranchuk formula, which is valid for a free electron gas.

We can calculate the scattering rate due to elastic collisions for Al and β-Ta, and compare them to the e-e scatter rate from the Landau-Pomeranchuk formula at E^* to see if the elastic collisions are indeed negligible in comparison. We will calculate E^* using [9, eq1]. The elastic collision time is calculated from the measured quasiparticle diffusion constant, see τ_{elastic} and τ_{e-e} in figure 3.2 and the code in Appendix B. The results are shown in table 3.1.

	$\tau_{e-e}^{-1}(E^*)$	$\tau_{\text{elastic}}^{-1}$
Al	0.2 GHz	83 MHz
β-Ta	2.8 MHz	2.9 GHz

Table 3.1: Scatter rates evaluated at energy E^* for both Al and β-Ta.

While for Al, the e-e scatter rate is indeed an order of magnitude higher than the elastic scattering rate due to disorder, for β-Ta, the calculated elastic scattering rate eclipses the e-e scattering rate. These results demonstrate a contradiction in applying Kozorezov's assumption to β-Ta. The fact that this assumption does not seem to hold for β-Ta makes for several complications. Kozorezov's formula for E^* does not hold. The same goes for the formula for the duration of stage 2 (τ_1), as it depends on E^*.

3.3.2 Stage 2: $E^* \rightarrow \Omega_D$

The conclusion that this stage ends with the creation of a 'phonon bubble' depends on the assumption that the duration of this stage (τ_1) is much shorter than the lifetime of a Debye phonon. $\tau_1 \ll \tau_{ph}$ There are reasons to believe this would not hold up for β-Ta. For one, the disorder of the material would introduce temperature-dependent anharmonic scattering [13]. Kozorezov states that, at least for a 'typical anharmonic potential', the scatter time of this process may be comparable to the lifetime of Debye phonons. Conceivably this process is more dominant for β-Ta.

To fully check the condition for the 'phonon bubble', one should therefore take into account the anharmonic contribution to the phonon lifetime, using e.g. the work of [14]. Because this fell out of

the scope of the project, the next best check is whether at least the pair breaking contribution to the phonon lifetime would violate this condition. Our results for both Al and β-Ta are shown in table 3.2. The condition is not violated for either material. However, our calculations for this condition depend on Kozorezov's calculations for E^* and the duration of stage 2, calculations for which we have shown above that they do not apply to β-Ta.

	τ_1	$\tau_{ph-e,K}$	$\tau_{ph-e,L}$	$\tau_{ph-e,T}$
Al	0.47 ps	7.7 ps	1.8 ps	1.0 ps
β-Ta	3.1 ps	6.3 ns	5.4 ps	19 ps

Table 3.2: Checking Kozorezov's condition that the duration of the stage must be much smaller than the photon lifetime. ($\tau_1 \ll \tau_{ph}$) For τ_1 we use Kozorezov's calculation[9, §II], and for τ_{ph} we use our calculations for the pair breaking times in figure 3.2, implemented in Appendix B.

3.3.3 Stage 3 and Stage 4: $\Omega_D \to 3\Delta$

Kozorezov states that anharmonic effects are inversely proportional to the fifth power of energy while the phonon lifetime is inversely proportional to the energy. Therefore he argues that in these stages the anharmonicity is quickly outpaced by Cooper pair breaking, and can be neglected. Again this is for a 'typical anharmonic potential', so this assumption might not hold for β-Ta.

Another significant assumption is that the density of excess quasiparticles is small, so that recombination events are mainly with the equilibrium quasiparticles in the material, rather than between excess quasiparticles. This density depends on the number of quasiparticles at any point in time as well as on the amount of space over which they are distributed. This in turn depends on the rate of diffusion, which is much slower for β-Ta, which might influence the validity of the assumption.

Regarding the diffusion of quasiparticles and phonons: Initially, Kozorezov neglects the (quasi)diffusion of phonons, as the diffusion of quasiparticles is much faster for the materials he considers. For β-Ta, this may or may not be the case. If anharmonic scattering or ph-e scattering do not limit the phonons' mean free path, the mean free path is given by the average distance a phonon can travel before it hits an edge of the volume of material. For a slab of $10 \times 10 \times 0.04\mu m$, this is about $0.09\mu m$ (see Appendix B), which corresponds to a diffusion of $D = 135\mu m^2/\mu s$, which would be faster than the $75\mu m^2/\mu s$ measured for quasiparticles in β-Ta. This violates the assumption made by Kozorezov. However, we must remember that the potential role of anharmonic scattering was not taken into account.

The potential prevalence of anharmonic scattering increase the downconversion of phonons, which would affect the distribution of phonon energies such that the average ph-e scattering rate decreases faster. This would in turn affect the duration of stage 3. All that is to say that anharmonicity, strongly linked to disorder, may invalidate Kozorezov's calculations for application to a material like β-Ta.

3.3.4 Summary

We have shown that various assumptions made in [9] do not hold up for β-Ta. We also raised potential problems which we have not exhaustively considered. To sum up these issues for β-Ta per stage:

- In stage 1 the assumption that the e-e scattering rate is much faster than the elastic scattering rate is shown not to hold up.
- In stage 2 the assumption that the stage's duration is much shorter than phonon lifetime may hold up. However, in our check we utilized calculations from stage 1 that do not apply to β-Ta.

Furthermore, we only considered the pair-breaking component to the phonon lifetime, while we expect the anharmonic decay may play a large role.

- For stage 3 and 4, we again stress that anharmonicity may not be negligible for β-Ta. We also cast doubt on the assumption that the excess quasiparticle density is low. Lastly, the assumption that phonon diffusion is slow compared to quasiparticle diffusion is shown to be false.

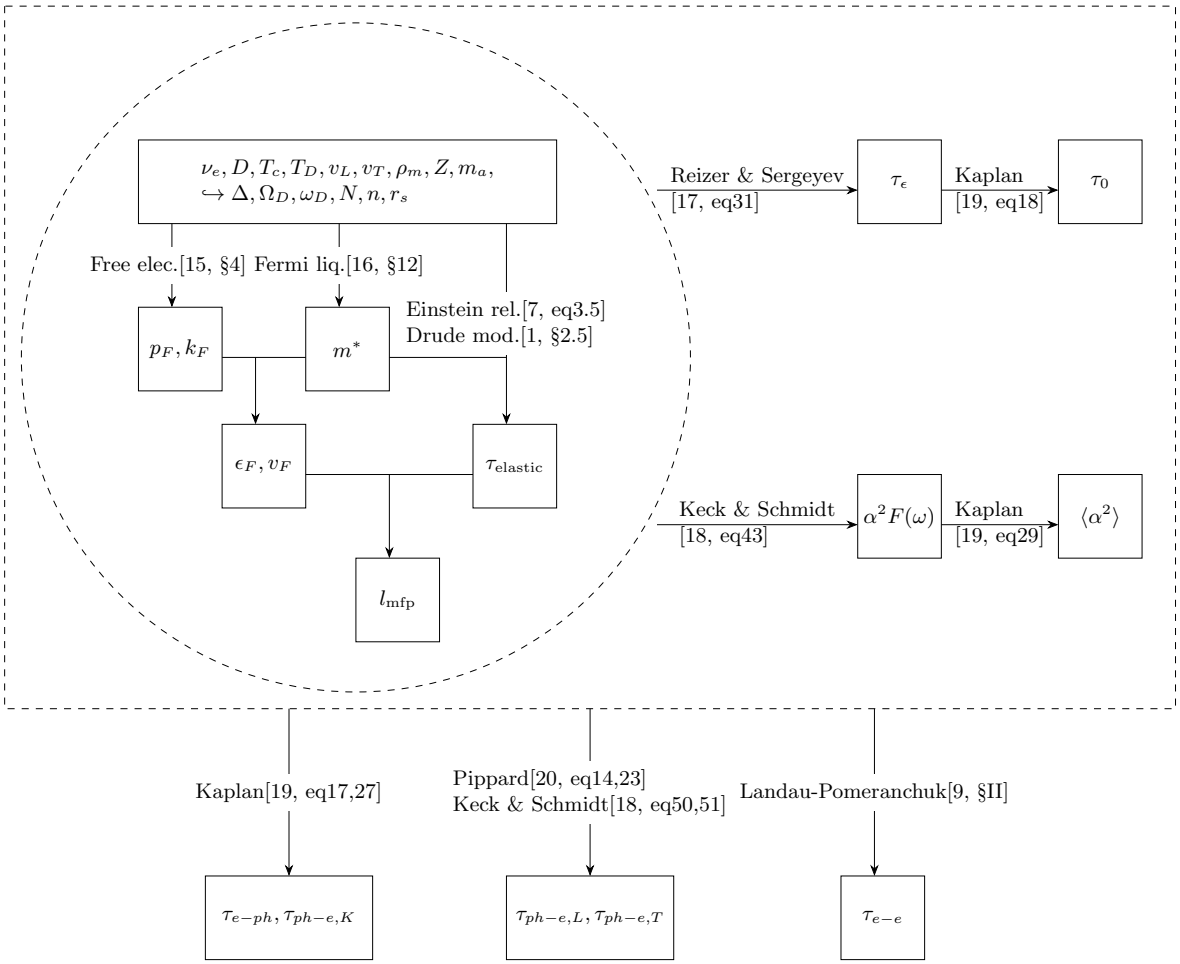


Figure 3.2: This flowchart schematically represents the calculations performed to obtain various scattering rates from a set of material parameters. For the values of the parameters and the code used in this calculation, see Appendix A and B.

3.4 Adapted Calculations for β-Ta

Since we cannot directly apply Kozorezov's assumptions and calculations to β-Ta, let us take a more qualitative view of the downconversion process, where we do still assume that downconversion approximately follows the stages described before. We will focus on the behavior of stage 3 and 4, as these will be most significant to formulating the initial conditions of the quasiparticle recombination process that will follow. To this end, we will calculate the e-ph and ph-e scattering rates for Al and β-Ta and compare these rates and the energy of their intersection. The schema of these calculations is shown in figure 3.2, the specific calculations are implemented in appendix B.

In order to calculate the necessary parameters from the ones we have available (see appendix A) we employ different underlying theories to describe the charge carriers. Our choices for these theories and

the values we calculate using them are shown in the dotted circle in figure 3.2. In order to perform calculations using the limited parameters we have, the charge carriers and their properties in these theories are sometimes identified with each other, even though they are not precisely the same concepts. The specific calculations we do can be found in appendix B.

Again, we do not include anharmonic scattering of the phonons, which could have a large influence on the specifics of Stage 3, the phonon-controlled stage. We calculate the e-ph and ph-e rates to see at what energy the transition occurs from stage 3 to 4. I.e. where the phonon distribution starts to create excess quasiparticles. In figure 3.2, we show the calculation method for these rates. We have a single calculation method for the e-ph rate (based on Kaplan[19]) and two methods for the ph-e rate (based on Pippard [20] and Kaplan [19]). The results of these calculations are presented in figure 3.3, which we interpret after explaining the calculation steps further.

3.4.1 Calculations based on Kaplan

For materials that are not disordered, there are energy dependent theoretical expressions available for τ_{e-ph} (which we associate with quasiparticle lifetime τ_s)[19, eq17] and τ_{ph-e} (which we associate with Cooper pair breaking time τ_B)[19, eq27]. For disordered superconductors, there is a theory for τ_{e-ph} by Reizer and Sergeyev[17]. However, the expression for τ_{e-ph} from this theory does not include energy dependence. Therefore we will adapt this theory as follows:

To obtain $\tau_{e-ph}(E, T = 0)$, we first calculate τ_ϵ according to [17, eq31]. τ_ϵ is the energy relaxation time of a quasiparticle at an energy much smaller than the thermal energy, and at temperature T_c. We can relate this to the characteristic time τ_0 according to $\tau_0 = \tau_s(0, T_c) \cdot 4.20$[19, eq18]. Using this characteristic time, we can then use Kaplan's expression for $\tau_{e-ph}(E, T = 0)$[19, eq17]. By doing so we assume that the energy dependence of this scattering is the same for a disordered material.

To obtain $\tau_{ph-e}(E)$, we first calculate the Eliashberg function $\alpha^2F(\omega)$ for β-Ta, using [18, eq43], a theory valid for disordered metals. We can then apply [19, eq29,27] to obtain an energy dependent expression for the ph-e rate. Again this assumes that this energy dependence behaves the same for a disordered material as for the ordered materials that the theory is made for.

3.4.2 Calculations based on Pippard

We also apply a second way to calculate the energy dependent ph-e scattering rate. For this we will apply the normal-state theory of the attenuation of ultrasonic waves in metals by interaction with conduction electrons. These attenuation rates α_L for longitudinal waves and α_T for transverse waves can be calculated according to Pippard's theory[20, eq14,23]. These attenuation rates are then converted to attenuation times by $\tau_{L,T}^{-1} = c_{L,T}\alpha_{L,T}$[18, eq50,51], where c is the sound velocity in the material. In taking these rates as representing pair breaking rates, we assume that the attenuation due to conduction electrons now corresponds to the interactions of phonons with electrons in the BCS ground state. At low temperatures, the density of the charges in the BCS ground state should be similar to the density of conduction electrons in the normal state, since these pair up to form the Cooper pairs. Furthermore, we consider phonons with high enough energy ($> 2\Delta$) to interact with and break these Cooper pairs. By doing this calculation for Al as well, we can gain some confidence that this equating this calculation to the pair-breaking rate is valid.

3.5 Naive IC Calculations for β-Ta

Let us momentarily put aside the objections raised in section 3.3, and apply some of Kozorezov's calculations to β-Ta at face value. We will use these to determine the duration of the downconversion stages. Using these durations and assuming the distributions of quasiparticles and phonons spread out according to their corresponding diffusivity, we can estimate the expected width σ_{IC} of the quasiparticle distribution at the end of the downconversion process. This distribution forms the initial condition

(IC) of the recombination process that follows. Lastly, we implement the method of [12] to calculate the expected pair breaking efficiency η_{PB} for β-Ta.

3.5.1 Duration

As the duration of the latter stages is expected to be more significant, we will focus on the last two stages. Kozorezov derives the durations of stage 3 and 4 as the following quantities respectively:

$$\begin{aligned} t_I &\approx 2\tau_B(\Omega_1) \\ t_{II} &\approx \tau_s(4.5\Delta) \end{aligned} \tag{3.1}$$

For the pair breaking time τ_B we will take our calculations for τ_{ph-e}, and for the electron scattering time τ_s we will take our calculation for τ_{e-ph}.

3.5.2 Final distribution width

We can approximate the width σ_{IC} of the final distribution of quasiparticles by calculating the mean distance due to phonon diffusion and quasiparticle diffusion for stage 3 and 4 respectively. For these, we will use the phonon diffusion $D = 135\mu m^2/\mu s$ as calculated in section 3.3.3. For the quasiparticle diffusivity, we have measurements amounting to $D = 75\mu m^2/\mu s$ for β-Ta. We can then use the mean distance formula for 2D diffusion $l = \sqrt{4Dt}$ to estimate the width σ_{IC} of the final quasiparticle distribution.

3.6 Results and Conclusions

3.6.1 Rate calculations

The results of the rate calculations explained above are shown in figure 3.3. From the Al graph in this figure, we can conclude that the Pippard calculations for τ_{ph-e} are about an order of magnitude off. Looking to the rates for β-Ta, we can conclude two things: Firstly, the rates are all a few orders of magnitude lower than for Al, across the range of energies plotted. Secondly, the transition energy Ω_1 from phonon-controlled to electron-controlled stage occurs at a much lower energy per particle. Depending on which ph-e rate to trust, Ω_1 even falls below the energy 3Δ, which marks the end of the downconversion process. We can make two main conclusions from this analysis.

1. If the transition occurs at such low energy, the phonon downconversion of stage 3 may result in the creation of phonons below 2Δ, where they cannot break more Cooper pairs. This loss may be exacerbated due to the anharmonic decay of phonons. Such phonon loss could decrease η_{PB}, yielding fewer quasiparticles, which would mean a higher Fano factor [12], which would reduce the achievable energy resolution of the device.

2. Because of the low transition energy Ω_1, we can expect stage 4 of the downconversion, in which many low energy quasiparticles are created, to continue into the recombination phase. For a material like Al, we can assume that all low energy quasiparticles have been generated before going into the recombination process. However, for β-Ta we can expect the generation of quasiparticles due to stage 4, and the recombination of quasiparticles to occur simultaneously. Therefore we need to simulate the recombination process while the excess quasiparticles still 'trickle in' from Stage 4 of the Downconversion phase. Because the phonon distribution at the end of Stage 3 decays exponentially with the pair-breaking time [9, eq17], we will simulate this overlap using an exponentially decaying influx of quasiparticles to the recombination stage.

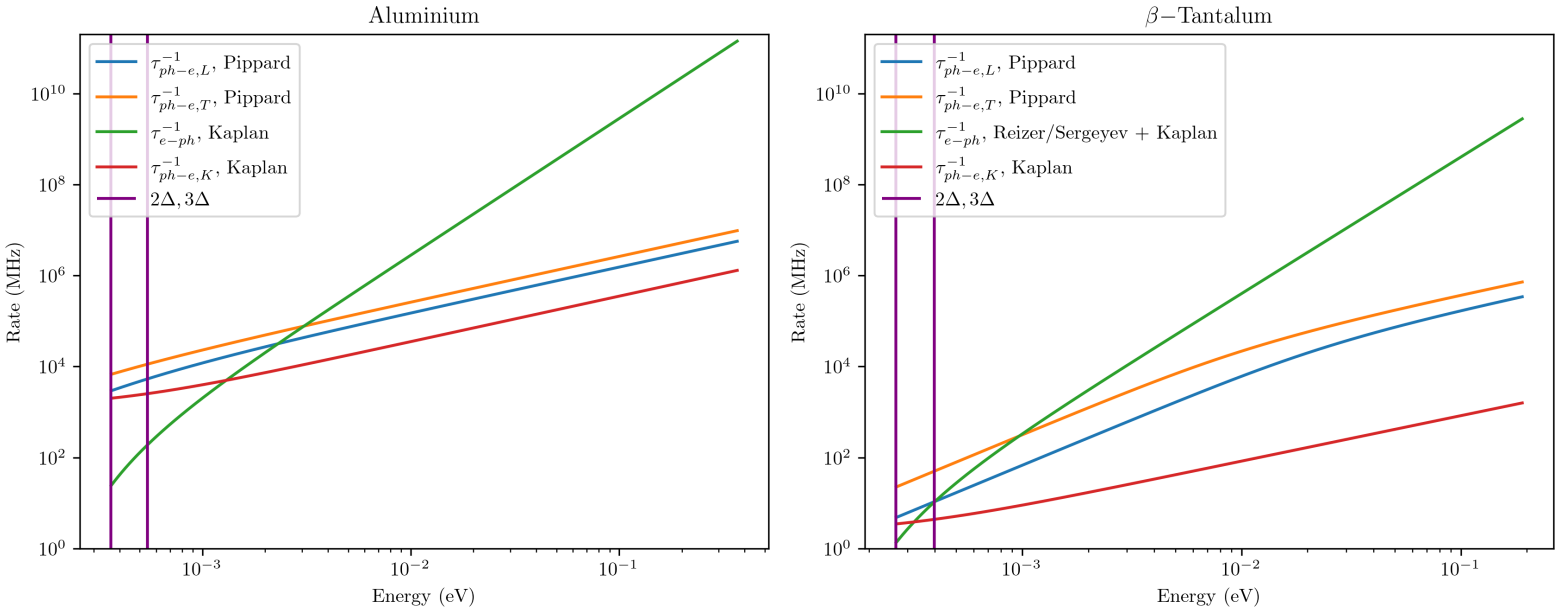


Figure 3.3: These graphs show the energy dependent e-ph and ph-e scattering rates calculated for Al and β-Ta using the method in figure 3.2. From the Al graph, we can see that the Pippard calculations do not exactly correspond to the Kaplan calculations, which are valid for Al. It is therefore hard to say whether the β-Ta results are trustworthy. Assuming that they are, we can conclude that the transition to Stage 4 happens at an energy much closer to the band gap for β-Ta, than for Al.

3.6.2 Naive IC Calculations

Duration

We fill in equation 3.1 as follows. We make three separate calculations for t_I by using our calculations for τ_{ph-e}. We evaluate these at the corresponding intersection points Ω_1 in figure 3.3. We calculate t_{II} with the adapted Kaplan function for τ_{e-ph}. These calculations are available in appendix B. The results are tabulated below:

t_I	$2\tau_{ph-e,K}(0.3meV)$	$0.5\mu s$
t_I	$2\tau_{ph-e,L}(0.4meV)$	$0.2\mu s$
t_I	$2\tau_{ph-e,T}(0.9meV)$	$8ns$
t_{II}	$\tau_{e-ph}(4.5\Delta)$	$19ns$

Table 3.3: Duration of stage 3 and 4, calculated using values from 3.2. For comparison, the values calculated for Al by Kozorezov are $t_I = 140ps$ and $t_{II} = 1.3ns$. We were able to reproduce $t_I = 160ps$ and $t_{II} = 1.0ns$ for Al, likely due to slightly differing parameters.

Link with τ_{IC}

As stated in the conclusions of section 3.6.1, we will model the overlap between the downconversion and recombination processes by an exponentially decaying source term of quasiparticles in the recombination model. The time constant of this source term we will call τ_{IC}. Kozorezov states that the number of phonons at the end of the downconversion process first linearly increases and then it expo-

nentially increases by breaking Cooper pairs. He concludes that the time over which this decay this occurs corresponds to $\tau_B(\Omega_1)$, i.e. the pair breaking time at the end of the phonon controlled stage.[9, §IV].

We will identify $\tau_B(\Omega_1)$ with τ_{IC}. Then, we can calculate τ_{IC} from half the time t_I in table 3.3. This means that, depending on the calculation method, we expect values from $4ns$ to $0.3\mu s$. This still leaves a large uncertainty in this parameter.

Final distribution width

We estimate the final distribution width σ_{IC} using the diffusion rates $D_{ph} = 135\mu m^2/\mu s$ and $D_e = 75\mu m^2/\mu s$, and the relation for the mean squared distance for diffusion in 2D, $4Dt$. Depending on how close Ω_1 is to the end of the downconversion process 3Δ, we can opt to include stage 4 ($\sigma_{IC} = \sqrt{4(D_{ph}t_I + D_e t_{II})}$) or exclude stage 4 ($\sigma_{IC} = \sqrt{4D_{ph}t_I}$). The results are tabulated below. The large range of results here give a large uncertainty in parameter σ_{IC}.

Calc. method	t_I	σ_{IC} excl. $t_{II} = 19ns$	σ_{IC} incl. $t_{II} = 19ns$
$2\tau_{ph-e,K}(0.3meV)$	$0.5\mu s$	$16\mu m$	$17\mu m$
$2\tau_{ph-e,L}(0.4meV)$	$0.2\mu s$	$10\mu m$	$11\mu m$
$2\tau_{ph-e,T}(0.9meV)$	$8ns$	$2\mu m$	$3\mu m$

Table 3.4: Duration of stage 3 and 4, calculated using values from 3.2. For comparison, the values calculated for Al by Kozorezov are $t_I = 140ps$ and $t_{II} = 1.3ns$. (We were able to reproduce $t_I = 160ps$ and $t_{II} = 1.0ns$, likely due to slightly differing parameters.)

Pair breaking efficiency

Plugging in our parameters for β-Ta into the code of [12], which is also based on Kozorezov's theory and may not hold for disordered superconductors, we obtain $\eta_{PB} \approx 59\%$, close to the maximum achievable pair breaking efficiency[21]. However, it may be that this method, being based on stage 3 and 4 as described by Kozorezov, is not reliably applicable to β-Ta.

Chapter 4

Quasiparticle Recombination

4.1 Introduction

The downconversion process as described in the previous chapter leaves us with a local distribution of excess quasiparticles. These quasiparticles carry part of the energy of an absorbed photon. The 'Recombination' process, which this chapter will describe, investigate and model, governs how the superconductor returns to a thermal equilibrium state, i.e. a state in which the excess quasiparticles have decayed and only a thermal distribution of quasiparticles remains.

We will model this decay according to the Rothwarf-Taylor equations[22], adapted to include spatial dependence due to the diffusion of the quasiparticles and phonons. We will simplify these equations to a single partial differential equation (PDE) under the assumption that some limits hold for the materials we consider. The resulting equation we will investigate analytically to confirm the influence of diffusion on its behavior. Lastly we will build a computational solution to the equation, which we will later use to compare to experimental data.

4.2 Theory

4.2.1 Rothwarf-Taylor Equations

In equation 4.1, the Rothwarf-Taylor equations are given[22]. These are the reaction equations that model the recombination of quasiparticles to form Cooper pairs, the simultaneous excitation of phonons, as well as the pair breaking due to phonons. These equations hold when the quasiparticles present have energies close to the gap edge Δ, which links to the investigation in the previous chapter.

$$\begin{aligned} \frac{dN_{qp}}{dt} &= -\frac{RN_{qp}^2}{V} + 2\Gamma_B N_\omega \\ \frac{dN_\omega}{dt} &= \frac{RN_{qp}^2}{2V} - \Gamma_B N_\omega - \Gamma_{es}\left(N_\omega - N_\omega^0\right) \end{aligned} \tag{4.1}$$

N_{qp} is the number of quasiparticles near the band gap, and N_ω is the number of phonons of energy $> 2\Delta$. The first terms of both these rate equations describe how two quasiparticles can recombine to form a Cooper pair, which generates a phonon. The second terms describe that one such phonon can also break a Cooper pair to generate two quasiparticles. The third term in the second equation gives the rate at which phonons are lost. For example, they can escape the superconductor towards the environment or the substrate or decay to $< 2\Delta$ phonons, due to anharmonic decay.

For solving these equations, it is convenient to write $N_{qp} = N_{qp}^0 + \delta N_{qp}$ and $N_\omega = N_\omega^0 + \delta N_\omega$, where N_{qp}^0 denotes the thermal equilibrium value and δN_{qp} the excess. Different limiting cases can be viewed

to solve this nonlinear system.
For a small excess $\delta N_{qp} \ll N^0_{qp}$, the system is approximately linear, and δN_{qp} can be shown to decay exponentially[6]. As shown in the introduction chapter (Ch. 1), the exponential model is not suitable to describe the decay of a β-Ta OKID's signal. We will derive this exponential solution in 4.3.

If $\delta N_{qp} \not\ll N^0_{qp}$, the linearization is not applicable. However, other simplifications can be made, such as $d\delta N_\omega/dt \approx 0$, assuming the pair breaking is much faster than the quasiparticle recombination. Solving this system gives a decay of the form[6]

$$\delta N_{qp}(t) = \frac{2N^0_{qp}}{\left(1 + 2N^0_{qp}/\delta N_{qp}(0)\right) e^{t/\tau^*_{qp}} - 1}, \tag{4.2}$$

where τ^*_{qp} can be expressed in terms of the rate constants. We will derive this solution in 4.3. The solution decays proportionally to $1/t$ at first before assuming an exponential decay. This model can successfully be fitted to single photon pulses of β-Ta. For this, however, you have to fit the 'ringing time' of the MKID to values that do not match the measurements[6]. We show these results in figure 4.1. The mentioned ringing time will be further explained at equation 4.18. Therefore the Rothwarf-Taylor equations were found not to be a predictive theory for the behavior of β-Ta OKIDs.

In the next section, we propose an adapted system, which is a local version of the Rothwarf-Taylor equations. This system considers the number density of particles instead of the number over a given volume, and includes diffusion of these particles. We will test whether this model can successfully predict the behavior of β-Ta OKIDs.

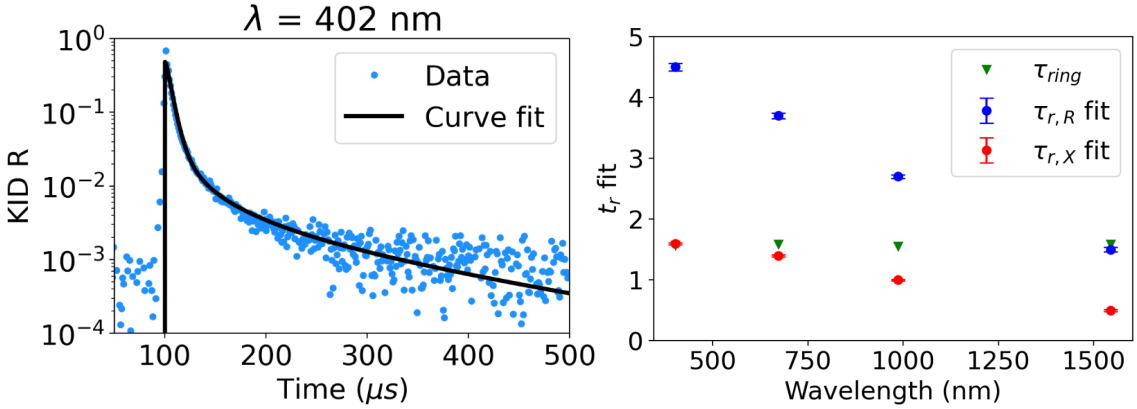


Figure 4.1: These results, from [6], show (left) that it is possible to rather nicely fit equation 4.2, to the β-Ta data. However (right) the fitted values for the ringing time largely disagree with the ringing time expected from the data.

4.2.2 Local RT-equations with diffusion

To approach the diffusion hypothesis theoretically, we will model an OKID's response based on the Rothwarf-Taylor equations[22], when including spatial terms due to diffusion. In essence, this will come down to solving the following system of differential equations.

$$\begin{aligned} \frac{dn_{qp}(\mathbf{r},t)}{dt} &= -Rn^2_{qp}(\mathbf{r},t) + 2\Gamma_B n_\omega(t) + D_{qp}\nabla^2 n_{qp}(\mathbf{r},t) \\ \frac{dn_\omega(\mathbf{r},t)}{dt} &= \frac{R}{2}n^2_{qp}(\mathbf{r},t) - \Gamma_B n_\omega(\mathbf{r},t) - \Gamma_{es}(n_\omega(t) - n_{\omega,0}) + D_\omega \nabla^2 n_\omega(\mathbf{r},t) \end{aligned} \tag{4.3}$$

In these equations, the particle densities n are taken to be both time and location dependent. R, Γ_B and Γ_{es} are rate coefficients of recombination, excitation and phonon escape. $n_{\omega,0}$ is the thermal equilibrium density of phonons.

It is unlikely this PDE system is solvable analytically, being a non-linear second order PDE. As such, we will restrict our analytic efforts to workable limiting cases in 1D. For the full picture we will computationally solve a simplification of these equations.

4.2.3 Simplifications

If we work under the assumption that the phonon processes are much faster than the quasiparticle processes, we can simplify the system in the following two ways.

Negligible phonon diffusion

In equation 4.3, there are diffusion terms for both the quasiparticles and phonons. The relative importance of these two processes depend on the diffusivity and the respective lifetime of the particle. We will neglect the diffusion of phonons, under the assumption that

$$D_{qp}\tau_{qp} \gg D_{\omega}\tau_{\omega}. \tag{4.4}$$

Let us validate this assumption. The necessary values are tabulated below. Filling in these values in condition 4.4, we obtain the equivalent conditions $n_{qp} \ll 1.6 \cdot 10^{10}\mu m^{-3}$ for Al and

$$n_{qp} \ll 6.9 \cdot 10^{6}\mu m^{-3} \tag{4.5}$$

for β-Ta.

Material	Al	β-Ta
Quasiparticle diffusion D_e	$15000\mu m^2/\mu s$	$75\mu m^2/\mu s$
Phonon diffusion D_ω	$200\mu m^2/\mu s$	$135\mu m^2/\mu s$
QP lifetime times QP density $\tau_{qp} \cdot n_{qp}$[5, eq2.29]	$31000\mu s/\mu m^3$	$345000\mu s/\mu m^3$
Phonon lifetime τ_ω	$0.15ns$	$28ns$

Table 4.1: Values needed to check our assumptions. The phonon lifetime is calculated using Pippard's ultrasonic attenuation theory[20] for longitudinal phonons at an energy of 2Δ. Values here are found in appendix A or calculated in appendix B

At sufficiently low quasiparticle densities, we can thus ignore the diffusion of phonons. Let us check this right away: Take a 402nm (3.1eV) photon. With perfect pair breaking efficiency it produces $3.1eV/\Delta \approx 23000$ excess quasiparticles in β-Ta. Assuming these are distributed in a 2D Gaussian with $\sigma_{IC} = 2\mu m$, the smallest value calculated in table 3.4, we obtain the highest possible quasiparticle density at the center of the Gaussian: $9700/2\pi\sigma_{IC} = 915\mu m^{-2}$. Lastly we convert this to a 3D density by dividing by a layer thickness of $0.04\mu m$. This gives a maximum excess quasiparticle density of $23 \cdot 10^3\mu m^{-3}$, which is far below condition 4.5.

To conclude, we can make this assumption, which allows us to ignore phonon diffusion. Therefore we obtain the following simplified system, where the spatial distribution of phonons is entirely defined by where they are created by quasiparticle recombination events.

$$\begin{aligned} \frac{dn_{qp}(\mathbf{r},t)}{dt} &= -Rn_{qp}^2(\mathbf{r},t) + 2\Gamma_B n_\omega(t) + D(\mathbf{r},t)\nabla^2 n_{qp}(\mathbf{r},t) \\ \frac{dn_\omega(\mathbf{r},t)}{dt} &= \frac{R}{2}n_{qp}^2(\mathbf{r},t) - \Gamma_B n_\omega(\mathbf{r},t) - \Gamma_{es}(n_\omega(\mathbf{r},t) - n_{\omega,0}) \end{aligned} \tag{4.6}$$

Phonon quasi steady state

We can make a similar simplification which allows us to assume the phonon equation is so fast compared to the quasiparticle equation that it immediately reaches steady state. For this we must assume that Γ_B or Γ_{es} is much larger than the quasiparticle recombination rate Rn_{qp}. Γ_{es} can be minimized in MKID design by e.g. suspending the detector on a membrane[4]. Therefore we will consider the condition $\Gamma_B \gg Rn_{qp}$, or equivalently:

$$\tau_{qp} \gg \tau_\omega$$

Using the data from the previous section, this amounts to the condition $n_{qp} \ll 2.1 \cdot 10^8 \mu m^{-3}$ for Al, and $n_{qp} \ll 1.2 \cdot 10^7 \mu m^{-3}$ for β-Ta. This is a similar requirement as for the previous simplification, and with our previous approximation for the maximum quasiparticle density being $23 \cdot 10^3 \mu m^{-3}$, this condition is also met.

We can use the phonon steady state assumption as follows. In equation 4.6, we add twice the second equation to the first.

$$\frac{\partial n_{qp}}{\partial t} - D\nabla^2 n_{qp} + 2\frac{\partial n_\omega}{\partial t} = -2\Gamma_{es}(n_\omega - n_{\omega,0})$$

Now we assume that the phonon rates Γ_B and Γ_{es} are much larger than the quasiparticle recombination rate Rn_{qp}. Changes in the quasiparticle density then happen at a longer timescale than changes in the phonon density. Therefore, whenever there is a change in n_{qp}, we can assume n_ω to quickly reach steady state, simplifying the equation above with $\frac{\partial n_\omega}{\partial t} \to 0$. Then, n_ω is given by

$$\frac{\partial n_{qp}}{\partial t} - D\frac{\partial^2 n_{qp}}{\partial x^2} = -2\Gamma_{es}(n_\omega - n_{\omega,0})$$

or

$$n_\omega = \frac{\frac{\partial n_{qp}}{\partial t} - D\nabla^2 n_{qp}}{-2\Gamma_{es}} + n_{\omega,0}$$

We substitute this in the first equation of 4.3. This gives the following PDE for n_{qp}. We thus reduced the system to a single PDE with one unknown.

$$\frac{\partial n_{qp}}{\partial t} - D\nabla^2 n_{qp} = \frac{2\Gamma_B n_{\omega,0} - Rn_{qp}^2}{1 + \frac{\Gamma_B}{\Gamma_{es}}} \tag{4.7}$$

Now let us express n_{qp} as $n_{qp,0} + \delta n_{qp}$. Here $n_{qp,0}$ is the thermal steady state value and δn_{qp} is the excess density of quasiparticles. Substituting in equation 4.7 gives the following equation.

$$\frac{\partial n_{qp,0}}{\partial t} - D\nabla^2 n_{qp,0} + \frac{\partial \delta n_{qp}}{\partial t} - D\nabla^2 \delta n_{qp} = \frac{2\Gamma_B n_{\omega,0} - Rn_{qp,0}^2}{1 + \frac{\Gamma_B}{\Gamma_{es}}} - \frac{2Rn_{qp,0}\delta n_{qp}}{1 + \frac{\Gamma_B}{\Gamma_{es}}} - \frac{R\delta n_{qp}^2}{1 + \frac{\Gamma_B}{\Gamma_{es}}}$$

The first terms on both sides of this equation correspond to equation 4.7 at steady state. Therefore these terms are 0. We thus obtain the nonlinear, homogeneous PDE below for the excess quasiparticle density n_{qp}.

$$\frac{\partial \delta n_{qp}}{\partial t} - D\nabla^2 \delta n_{qp} = -\frac{2Rn_{qp,0}\delta n_{qp}}{1 + \frac{\Gamma_B}{\Gamma_{es}}} - \frac{R\delta n_{qp}^2}{1 + \frac{\Gamma_B}{\Gamma_{es}}} \tag{4.8}$$

Let us change notation for conciseness. $Q = \delta n_{qp}$, $Q_0 = n_{qp,0}$, and $R' = R/(1 + \frac{\Gamma_B}{\Gamma_{es}})$

$$Q_t - D\nabla^2 Q = -2R'Q_0 Q - R'Q^2 \tag{4.9}$$

This is the equation we will work with for the rest of this thesis.

4.3 Analytical Examination

Since equation 4.9 is still not evident to solve, we can examine further limiting cases analytically to probe the behavior of the equation. In table 4.2, we present an overview of the relevant simplified cases that can be distinguished, as well as the full system. In the following sections, we will solve or approach these cases one by one.

Limiting cases	**Small excess qp density**	**Large excess qp density**
Fast diffusion	$Q_t = -2R'Q_0Q$ $Q(t_0) = A$	$Q_t = -2R'Q_0Q - R'Q^2$ $Q(t_0) = A$
Slow diffusion	$Q_t - D\nabla^2 Q = -2R'Q_0Q$ $Q(\mathbf{r}, t_0) = A\delta(\mathbf{r})$	$Q_t - D\nabla^2 Q = -2R'Q_0Q - R'Q^2$ $Q(\mathbf{r}, t_0) = A\delta(\mathbf{r})$

Table 4.2: Cases (in terms of differential equation and initial condition) that can be distinguished in the recombination process of quasiparticles. A high diffusivity means that we can consider the initial impulse of quasiparticles to be spread out instantaneously, which eliminates the diffusive term. A small excess quasiparticle density allows us to neglect the quadratic term.

4.3.1 Fast diffusion, small excess

This simple case describes the exponential decay that is observed at the tail of the quasiparticle recombination. For highly diffusive materials, with either a low energy photon absorption or a large bulk of material over which the quasiparticles can distribute, this model suffices in describing the quasiparticle recombination.

$$\begin{aligned} Q_t &= -2R'Q_0Q \\ Q(\mathbf{r}, 0) &= A \end{aligned}$$

This ODE is easily solved by the following expression. This is the same solution as the exponential solution mentioned in section 4.2.1.

$$Q(t) = Ae^{-2R'Q_0t} \tag{4.10}$$

This limit of the PDE is particularly useful, as it corresponds to the tail of the pulse response. This allows us to perform a simple fitting procedure for the product $R'Q_0$.

4.3.2 Fast diffusion, large excess

This case describes the quasiparticle decay for a highly diffusive material, regardless of the magnitude of the excess density of quasiparticles created by a photon absorption.

$$\begin{aligned} Q_t &= -2R'Q_0Q - R'Q^2 \\ Q(\mathbf{r}, 0) &= A \end{aligned}$$

We can solve this ODE through separation of variables. Rewriting gives the equation

$$-\frac{Q_t}{R'Q(2Q_0 + Q)} = 1$$

Integrate both sides over time from 0 to t.

$$-\int_0^t \frac{Q_\tau}{R'Q(2Q_0 + Q)} d\tau = \int_0^t d\tau$$

We can evaluate the r.h.s. and simplify the l.h.s. by changing to an integral over Q.

$$-\int_A^{Q(t)} \frac{dQ}{R'Q(2Q_0+Q)} = t$$

The remaining integral by e.g. decomposing the fraction. This gives an implicit solution for $Q(t)$.

$$\frac{1}{2Q_0R'}\left(\ln\left(\frac{2Q_0}{Q(t)}+1\right)-\ln\left(\frac{2Q_0}{A}+1\right)\right)=t$$

Solving for $Q(t)$ gives the result below. This is the same solution as the nonlinear solution mentioned in section 4.2.1.

$$Q(t)=\frac{2Q_0}{\left(\frac{2Q_0}{A}+1\right)e^{2Q_0R't}-1} \tag{4.11}$$

We will show in section 4.3.4 that this solution can be altered to find the solution for zero diffusion. This will prove to be a useful simplification to approach the full solution of the PDE.

4.3.3 Slow diffusion, small excess

Consider equation 4.9 in just one spatial dimension x, on an infinite domain. The initial condition (IC) is defined by is an impulse of magnitude A at $x=0$. Let us restrict ourselves to the simple case that the excess number of quasiparticles is much smaller than the equilibrium amount ($Q \ll Q_0$). Then equation 4.9 is approximately linear.

$$\begin{aligned} Q_t - DQ_{xx} &= -2R'Q_0Q \\ Q(x,0) &= A\delta(x) \end{aligned} \tag{4.12}$$

We can apply separation of variables $Q(x,t)=X(x)T(t)$.

$$XT_t - DTX_{xx} = -2R'Q_0XT$$

Isolate time dependent and location dependent parts.

$$\frac{T_t}{T}+2R'Q_0 = D\frac{X_{xx}}{X} = -\lambda$$

The l.h.s. is independent of x. The r.h.s. is independent of t. Therefore, both must be equal to some arbitrary constant $-\lambda$.

$$\begin{aligned} T_t &= -2R'Q_0T-\lambda T \\ X_{xx} &= -\frac{\lambda}{D}X \end{aligned}$$

These ODEs have the following solutions.

$$\begin{aligned} T &\propto e^{-\lambda t}e^{-2R'Q_0t} \\ X &\in \operatorname{span}\left\{e^{-ix\sqrt{\lambda/D}}, e^{ix\sqrt{\lambda/D}}\right\} \end{aligned}$$

Thus, the general solution to equation 4.12 is given below, where the integral over the eigenvalues $\lambda \geq 0$ is a generalized superposition [23, §10.2].

$$Q=\int_0^\infty \left(A(\lambda)e^{-ix\sqrt{\lambda/D}}e^{-\lambda t}e^{-2R'Q_0t}+B(\lambda)e^{ix\sqrt{\lambda/D}}e^{-\lambda t}e^{-2R'Q_0t}\right)d\lambda$$

If we substitute $k = \sqrt{\lambda/D}$, we can equivalently integrate over $k \in (-\infty, \infty)$.

$$Q = e^{-2R'Q_0 t} \int_{-\infty}^{\infty} C(k) e^{-ikx} e^{-Dk^2 t} dk \tag{4.13}$$

Now let us impose the initial condition $Q(x, 0) = A\delta(x)$.

$$A\delta(x) = \int_{-\infty}^{\infty} C(k) e^{-ikx} dk$$

We recognize this as inverse Fourier transform, and find $C(k)$ by transforming the IC.

$$C(k) = \frac{1}{2\pi} \int_{-\infty}^{\infty} A\delta(x) e^{ikx} dx = \frac{A}{2\pi}$$

Then we can obtain a closed expression for Q.

$$Q = e^{-2R'Q_0 t} \frac{A}{2\pi} \int_{-\infty}^{\infty} e^{-ikx} e^{-Dk^2 t} dk = \frac{A}{2\sqrt{\pi D t}} e^{-2R'Q_0 t} e^{-\frac{x^2}{4Dt}} \tag{4.14}$$

If we integrate this over x to find the excess number of quasiparticles, this is simply the same exponential decay as in section 4.3.1. Therefore on its own this limiting case of the PDE does not explain a non-exponential pulse shape.

4.3.4 Slow diffusion, large excess

Analytically we were unable to exactly solve this case. However, it is possible to discern its behavior by approaching it from the earlier described cases.

Quadratic term perturbation

To approximate the solution for a small but not-negligible quadratic term, we apply a perturbation approach to extend our linear solution given in eq. 4.14, which we now call L.

For a small parameter ϵ, we write the following PDE and ansatz.

$$\begin{aligned} Q_t - DQ_{xx} &= -2R'Q_0 Q - \epsilon R' Q^2 \\ Q &= L + \epsilon P \end{aligned}$$

Filling in the ansatz gives the equation

$$\begin{aligned} L_t - DL_{xx} + 2Q_0 R' L + \\ R'\epsilon^3 P^2 + 2R'\epsilon^2 LP + \\ \epsilon \left(P_t - DP_{xx} + 2Q_0 R' P + R' L^2 \right) = 0 \end{aligned}$$

In which we recognise the first line equals 0, as it is solved by the linear solution. The second line, containing the higher orders of ϵ, is neglected. This leaves us with the third line as linear PDE for the perturbation term P.

$$P_t - DP_{xx} + 2Q_0 R' P = -R' L^2$$

Except for the forcing term $-R'L^2$, this is the same PDE as the linearized case of equation 4.12. Therefore this equation will have the same Green's function. We can then directly obtain the Green's function from our linear solution, as that solution is the response function for an impulse at $t = 0$ and $x = 0$, scaled by A. Given the translational symmetry of the problem we can then find the Green's function by translating L and A [23, §11.3.8].

$$G(x, t; x_0, t_0) = \frac{1}{2\sqrt{\pi D(t - t_0)}} e^{-2R'Q_0(t-t_0)} e^{-\frac{(x-x_0)^2}{4D(t-t_0)}} \tag{4.15}$$

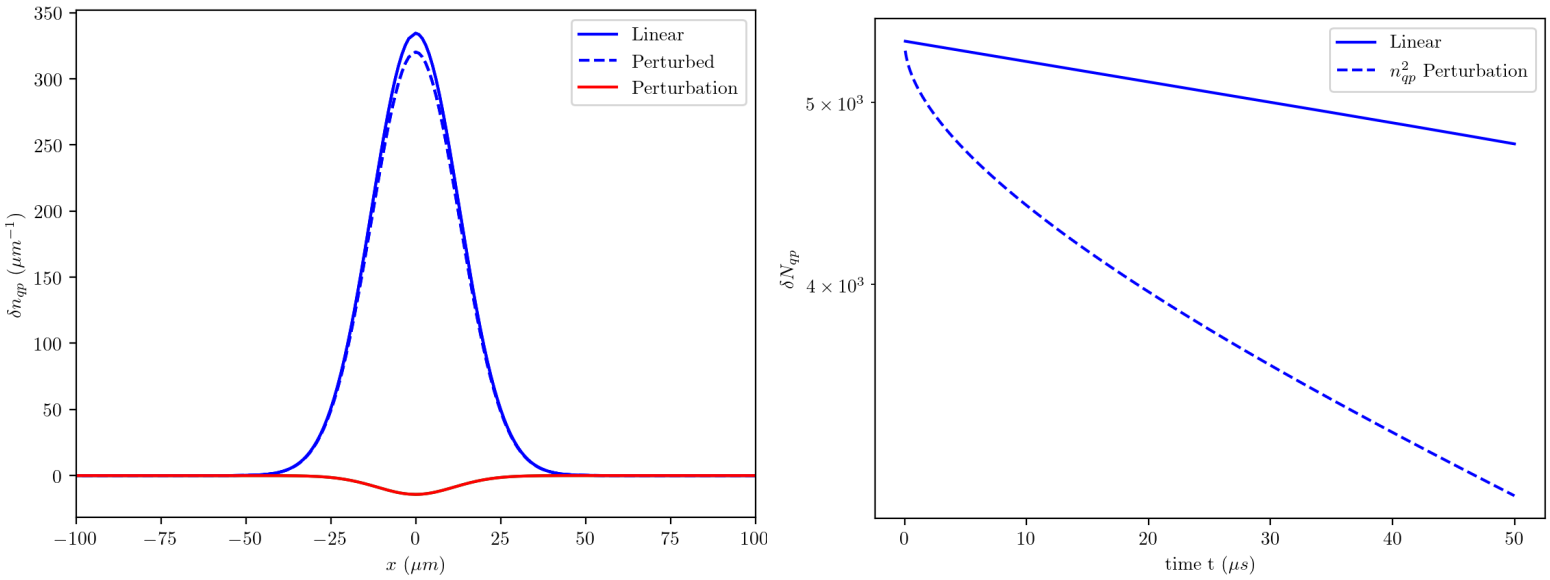


Figure 4.2: These plots qualitatively show the effect that the nonlinearity has on the behavior of the decay. On the left, at a single timestep, we observe the effect that the perturbed distribution has decayed faster. On the right, we see that this indeed introduces a faster-than-exponential decay.

This allows us to find the solution for P by evaluating the contribution from the initial condition as well as the forcing term.

$$P = \int_{-\infty}^{\infty} P(x_0, 0) G(x, t; x_0, 0) dx_0 + \int_{-\infty}^{\infty} \int_0^t -R' L^2(x_0, t_0) G(x, t; x_0, t_0) dt_0 dx_0$$

For the IC we take $P(x_0, 0) = 0$, as the IC for Q is exactly satisfied by our linear solution L so no perturbative correction is necessary at $t = 0$. Simplifying the second integral as far as possible gives the result below. Setting $\epsilon = 1$ then gives the approximate solution $Q = L + P$ to equation 4.9.

$$P = \int_0^t -\frac{A^2 R' e^{\frac{-2DQ_0 R' t_0 (t+t_0) + \frac{x^2 (t-t_0)}{2t-t_0} - \frac{x^2}{2}}{Dt_0}}}{4\pi\sqrt{D}\sqrt{t_0}\sqrt{2t-t_0}} dt_0 \tag{4.16}$$

The effect of introducing this nonlinear behavior to the recombination process is shown in figure 4.2.

Diffusion term perturbation

In section 4.3.2, we solved the system for exceedingly fast diffusion. In this section we will first show that the found solution for fast diffusion will give us a general solution to the situation without diffusion. From there, we will introduce a small amount of diffusion through a perturbation approach.

For very fast diffusion, the quasiparticles that are created immediately spread out over the superconductor. Therefore equation 4.9 reduced to the following ODE with scalar initial condition.

$$Q_t = -2R' Q_0 Q - R' Q^2$$
$$Q(\mathbf{r}, 0) = A$$
$$\text{With solution } \quad Q(t) = \frac{2Q_0}{\left(\frac{2Q_0}{A} + 1\right) e^{2Q_0 R' t} - 1}$$

Now let us look at the limit where there is no diffusion. This gives us the same ODE but with an IC that can be spatially dependent.

$$\begin{aligned} Q_t &= -2R'Q_0Q - R'Q^2 \\ Q(\mathbf{r},0) &= A(\mathbf{r}) \end{aligned}$$

Since there is no spatial derivative in the ODE, the solution has exactly the same form as before, with the only spatial dependence given by the initial condition.

$$Q(\mathbf{r},t) = \frac{2Q_0}{\left(\frac{2Q_0}{A(\mathbf{r})}+1\right)e^{2Q_0R't}-1}$$

Let us call this unperturbed solution N. For a small parameter ϵ, we write the following PDE and ansatz.

$$\begin{aligned} Q_t - \epsilon D\nabla^2 Q &= -2R'Q_0Q - R'Q^2 \\ Q &= N + \epsilon P \end{aligned}$$

Filling in the ansatz gives the equation

$$\begin{aligned} N_t + 2Q_0R'N + R'N^2 \; + \\ \epsilon^2(-D\nabla^2 P + RP^2) \; + \\ \epsilon(P_t - D\nabla^2 N + 2R'Q_0P + 2R'NP) = 0 \end{aligned}$$

In which we recognise the first line equals 0, as it is the unperturbed equation, which is solved by N. The second line, containing the higher orders of ϵ, is neglected. This leaves us with the third line as first order, linear, inhomogeneous ODE for the perturbation term P.

$$P_t + 2R'(Q_0+N)P = D\nabla^2 N$$

The homogeneous solution is found by separation of variables.

$$\begin{aligned} \frac{P_{hom,t}}{P_{hom}} &= -2R'(Q_0+N) \\ \int_0^t \frac{P_{hom,\tau}}{P_{hom}}d\tau &= -2R'\int_0^t (Q_0+N(\mathbf{r},\tau))d\tau \\ \int_{P_{hom}(\mathbf{r},0)}^{P_{hom}(\mathbf{r},t)} \frac{1}{P_{hom}}dP_{hom} &= -2R'\int_0^t Q_0+N(\mathbf{r},\tau)d\tau \\ \ln|P_{hom}(\mathbf{r},t)| &= -2R'\left(Q_0t + \frac{-2Q_0R't + \ln\left(\frac{(2Q_0+A(\mathbf{r}))e^{2Q_0R't}-A(\mathbf{r})}{2Q_0}\right)}{R'}\right) + \ln|P_{hom}(\mathbf{r},0)| \\ P_{hom}(\mathbf{r},t) &= P_{hom}(\mathbf{r},0)e^{2R'Q_0t-2\ln\left(\frac{(2Q_0+A(\mathbf{r}))e^{2Q_0R't}-A(\mathbf{r})}{2Q_0}\right)} \end{aligned}$$

We can find a particular solution by multiplying the inhomogeneous ODE with $P_{hom}^{-1}(\mathbf{r},t)/P_{hom}(\mathbf{r},0) = e^{2R'\int_0^t(Q0+N)d\tau}$.

$$P_te^{2R'\int_0^t(Q0+N)d\tau} + 2R'(Q_0+N)e^{2R'\int_0^t(Q0+N)d\tau}P = D(\nabla^2N)e^{2R'\int_0^t(Q0+N)d\tau}$$

Using the product rule, we find

$$(Pe^{2R'\int_0^t(Q0+N)d\tau})_t = D(\nabla^2N)e^{2R'\int_0^t(Q0+N)d\tau}$$

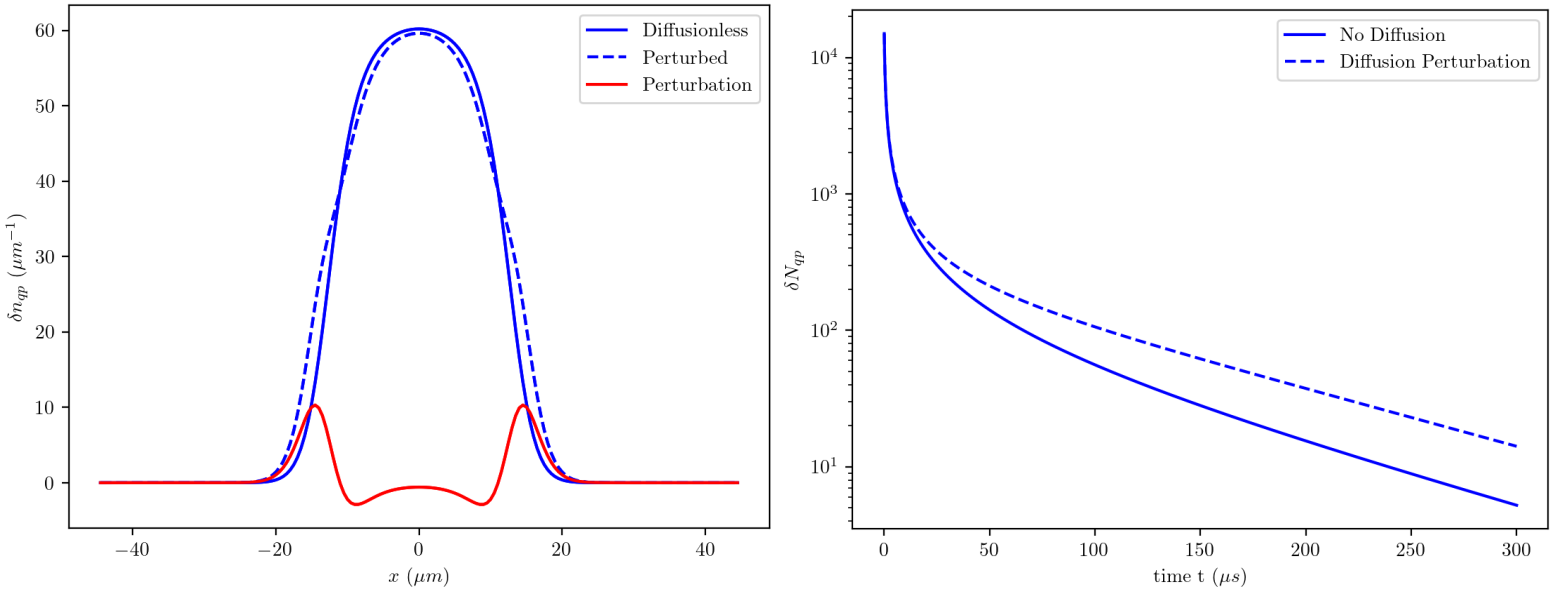


Figure 4.3: These plots show, by means of a diffusion perturbation on an exact nonlinear solution without diffusion, that introducing diffusion leads to a slower decay.

Integrating and solving for P gives

$$P_{part}(\mathbf{r},t) = De^{-2R'\int_0^t (Q0+N)d\tau}\int_0^t (\nabla^2 N(\mathbf{r},\tau))e^{2R'\int_0^\tau (Q0+N(\mathbf{r},\tau'))d\tau'}d\tau$$

$$P_{part}(\mathbf{r},t) = De^{2R'Q_0t-2\ln\left(\frac{(2Q_0+A(\mathbf{r}))e^{2Q_0R't}-A(\mathbf{r})}{2Q_0}\right)}\int_0^t (\nabla^2 N(\mathbf{r},t))e^{-2R'Q_0t+2\ln\left(\frac{(2Q_0+A(\mathbf{r}))e^{2Q_0R't}-A(\mathbf{r})}{2Q_0}\right)}dt$$

So the general solution to the ODE is given by

$$P(\mathbf{r},t) = P_{hom}(\mathbf{r},t) + P_{part}(\mathbf{r},t)$$

The IC requires that the perturbation P should be zero at $t = 0$, since the unperturbed solution is exact initially.

$$P(\mathbf{r},0) = P_{hom}(\mathbf{r},0) + P_{part}(\mathbf{r},0) = P_{hom}(\mathbf{r},0) + 0 = 0$$

Therefore we find that $P_{hom}(\mathbf{r},0) = 0$ and thus $P = P_{part}$.

$$P = D\frac{e^{2R'Q_0t}}{\left(\frac{(2Q_0+A(\mathbf{r}))e^{2Q_0R't}-A(\mathbf{r})}{2Q_0}\right)^2}\int_0^t e^{-2R'Q_0\tau}\left(\frac{(2Q_0+A(\mathbf{r}))\,e^{2Q_0R'\tau}-A(\mathbf{r})}{2Q_0}\right)^2\nabla^2 N(\mathbf{r},\tau)d\tau \quad (4.17)$$

The effect of introducing diffusion to the recombination process is shown in figure 4.3

4.3.5 Discussion of analytical methods

The diffusion of quasiparticles in β-Ta is neither near 0 nor exceedingly large. Furthermore, starting from a localized excitation as we expect from chapter 3, we cannot restrict ourselves to a low density of excess quasiparticles. Therefore none of the analytical solutions presented would be sufficient to simulate the response of a β-Ta OKID. The simplest case of exponential decay at low quasiparticle density is still useful to describe and get fitted parameters of the tail of a measurement. Regarding

the diffusionless model, from figure 4.3, we see that it would already decay too quickly compared to a model that includes even a small perturbation of diffusion. In order to fully capture the behavior of the system, as well as include other complicating factors such as border conditions, we need to numerically model the system.

4.4 Numerical Model

In order to capture the full behavior of the PDE we must resort to numerically solving it. We will implement a 1D finite volume approach. This is a convenient choice compared to finite difference, as we will see it naturally reduces the order of the spatial derivatives to deal with. Furthermore, it is much more straightforward to implement than finite element methods.

4.4.1 1D finite volume scheme

We start from the same PDE as before, now for generality also including a source term $S(x,t)$. Consider the domain Ω, an interval on the x-axis, corresponding to the length of absorber. Let us separate this interval into N smaller intervals, or 'volumes' Ω_i of length Δx, such that $\cap_i \Omega_i = \emptyset$ and $\cup_i \Omega_i = \Omega$. Let Ω_i and Ω_{i+1} border at location $x_{i+\frac{1}{2}}$. We start by integrating the PDE on one such volume.

$$\frac{\partial Q}{\partial t} = D\frac{\partial^2 Q}{\partial x^2} - R'Q^2 - 2R'Q_0Q + S$$
$$\int_{\Omega_i} \frac{\partial Q}{\partial t} dx = \int_{\Omega_i} D\frac{\partial^2 Q}{\partial x^2} dx - R' \int_{\Omega_i} Q^2 dx - 2R'Q_0 \int_{\Omega_i} Q dx + \int_{\Omega_i} S dx$$

We will work with averages over the volumes defined as $\langle Q \rangle_i = \frac{1}{\Delta x}\int_{\Omega_i} Q dx$. We can directly evaluate the integral of the diffusion term. Furthermore, we approximate $\frac{1}{\Delta x}\int_{\Omega_i} Q^2 dx \approx \langle Q \rangle_i^2$.

$$\Delta x \frac{\partial \langle Q \rangle_i}{\partial t} = \left[D\frac{\partial Q}{\partial x} \right]_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} - \Delta x R' \langle Q \rangle_i^2 - 2\Delta x R' Q_0 \langle Q \rangle_i + \Delta x \langle S \rangle_i$$

Next we approximate the remaining spatial derivative by a finite difference of the average values of adjacent volumes.

$$\Delta x \frac{\partial \langle Q \rangle_i}{\partial t} = D|_{x_{i+\frac{1}{2}}} \frac{\langle Q \rangle_{i+1} - \langle Q \rangle_i}{\Delta x} - D|_{x_{i-\frac{1}{2}}} \frac{\langle Q \rangle_i - \langle Q \rangle_{i-1}}{\Delta x} - \Delta x R' \langle Q \rangle_i^2 - 2\Delta x R' Q_0 \langle Q \rangle_i + \Delta x \langle S \rangle_i$$

We arrive at the following non-linear system of ODEs.

$$\frac{\partial \langle Q \rangle_i(t)}{\partial t} = D|_{x_{i+\frac{1}{2}}} \frac{\langle Q \rangle_{i+1}(t) - \langle Q \rangle_i(t)}{\Delta x^2} - D|_{x_{i-\frac{1}{2}}} \frac{\langle Q \rangle_i(t) - \langle Q \rangle_{i-1}(t)}{\Delta x^2} - R' \langle Q \rangle_i^2(t) - 2R'Q_0 \langle Q \rangle_i(t) + \langle S \rangle_i(t)$$

Time marching

In order to simulate the response in time, we should also discretize in time. Let us write the previous equation as:

$$\frac{\partial \mathbf{Q}(t)}{\partial t} = \mathbf{f}[\mathbf{Q}(t)]$$

We implemented four finite difference approaches to integrate this equation for a given IC. Explicit methods, which have a closed expression for the update step, would have preference. After all, they do not require any difficult systems of equations to be solved in the update step, which would slow down the process. The simplest conceivable method is the explicit Forward Euler method, with error of order $O(\Delta t)$, which gives

$$\mathbf{Q}(t + \Delta t) = \mathbf{Q}(t) + \Delta t \mathbf{f}[\mathbf{Q}(t)]$$

We realized this method is stable only for extremely small Δt ($< 1fs$), because of the quadratic term, which makes the equation particularly 'stiff'. Another option that was then implemented is the explicit Runge-Kutta method, with error $O(\Delta t^4)$.

$$\begin{aligned}\mathbf{Q}(t+\Delta t) &= \mathbf{Q}(t) + \frac{\Delta t}{6}(\mathbf{k}_1 + 2\mathbf{k}_1 + 2\mathbf{k}_3 + \mathbf{k}_4)\\ \mathbf{k}_1 &= \mathbf{f}[\mathbf{Q}(t)]\\ \mathbf{k}_2 &= \mathbf{f}[\mathbf{Q}(t) + \Delta t\mathbf{k}_1/2]\\ \mathbf{k}_3 &= \mathbf{f}[\mathbf{Q}(t) + \Delta t\mathbf{k}_2/2]\\ \mathbf{k}_4 &= \mathbf{f}[\mathbf{Q}(t) + \Delta t\mathbf{k}_3]\end{aligned}$$

Alas, also this higher order scheme did not appear to be sufficiently stable for the equation we are solving. Therefore we turned to implicit schemes, which are inherently more stable. The Backward Euler method gives the following system of quadratic equations as update step, with error $O(\Delta t)$.

$$\mathbf{Q}(t+\Delta t) = \mathbf{Q}(t) + \Delta t\mathbf{f}[\mathbf{Q}(t+\Delta t)]$$

The Crank-Nicolson method gives a similar implicit equation, but should be more accurate, as it has an error term of order $O(\Delta t^2)$.

$$\mathbf{Q}(t+\Delta t) = \mathbf{Q}(t) + \frac{\Delta t}{2}\left(\mathbf{f}[\mathbf{Q}(t)] + \mathbf{f}[\mathbf{Q}(t+\Delta t)]\right)$$

Since these implicit update equations are a system of quadratic equations, there is no general analytic solution. Therefore we approximate the solution using the `fsolve` function in `SciPy`, with $\mathbf{Q}(t)$ as initial guess. As both equations seem sufficiently stable and similarly computationally intensive, we will continue with using the Crank-Nicolson method, on account of its error term which is higher order in Δt.

Boundary conditions

Unlike in the analytical efforts, we cannot simulate an infinite domain. Therefore we should take into account boundary conditions. Since, typically, an absorber is used with a lower band gap than the surrounding material, we can assume that no quasiparticles diffuse outside the edges of the absorber into a material with higher band gap. Conversely, the material with higher band gap will have a smaller amount of thermal quasiparticles, so no quasiparticles will diffuse into the absorber either.

We can model this by imposing the Von Neumann boundary condition, by setting the diffusion to zero at the boundary of the domain. In the numerical model, we can implement this by padding the grid of volumes on both sides with an extra volume that matches the value of the boundary volume. This ensures the diffusion term is zero at the edges.

Conversion to signal

In order to convert the simulation results to the observable phase or amplitude measurements θ or A, we do the following. We integrate $Q(x,t)$ at each time step over x to obtain $\delta N_{qp}(t)$.

We should take into account the response time (or ringing time) of the resonator $\tau_{\text{ring}} = Q/\pi f_0$. Where Q is now the quality factor of the resonator, and f_0 the resonance frequency. Assuming this response time not only determines the response to an electrical signal impulse but also to an impulse in δN_{qp}, we can view the response to $\delta N_{qp}(t)$ as being filtered by a function $\propto \exp(-t/\tau_{\text{ring}})$. We can implement this according to the following convolution, where τ_{ring}^{-1} normalizes the exponential function, and H is the Heaviside step function.

$$\delta\hat{N}_{qp}(t) = \delta N_{qp}(t) * H(t)\tau_{\text{ring}}^{-1}\exp(-t/\tau_{\text{ring}}) \tag{4.18}$$

Next, assuming $\delta\hat{N}_{qp}$ is small enough, so that N_{qp} falls within the linear part of $\theta(N_{qp})$ and $A(N_{qp})$, we can use measurements for $d\theta/dN_{qp}$ or dA/dN_{qp} as conversion factors to obtain $\theta(t)$ or $A(t)$. Further assumptions made for this calculation are that the local changes in conductivity σ due to δn_{qp} are also linear, and that the current in the MKID is the same throughout the domain, such that the response due to changes in conductivity is independent of the location of the excess quasiparticles. We will go more into this in the next chapter.

4.4.2 Implementation in code

In appendix C, we present and explain a Python program that implements the finite volume scheme of the previous section. An example simulation of this code is shown in figure 4.4. The code also includes tools to obtain parameters from MKID measurement data, which are used in the simulation. The simulation was extended to include an option for energy dependent diffusivity, and an option for an exponentially decaying source term instead of a simple initial condition. Lastly, some optimizations were made to improve simulation time, so that we can use small dt and dx step sizes, which are desirable to keep the simulation stable.

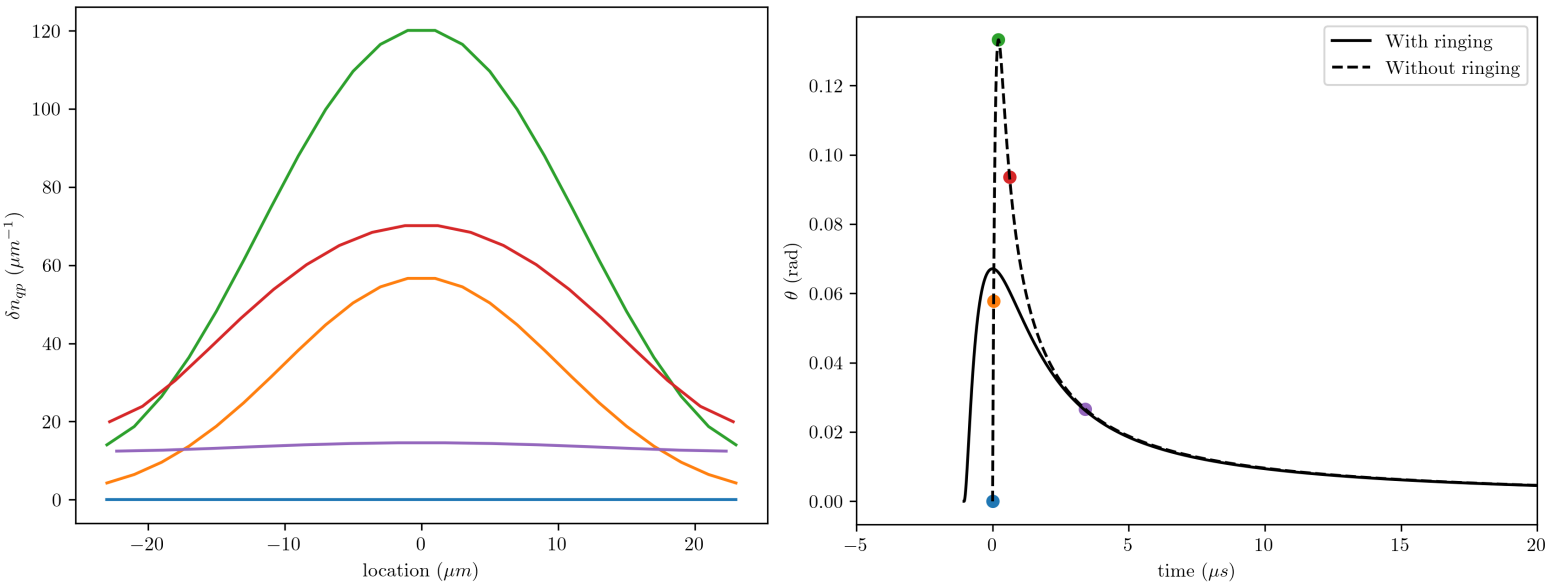


Figure 4.4: These plots show how the excess quasiparticle density varies as we move forward in the signal. The colors of the distributions on the left correspond to the marks on the right.

Additions for physical accuracy

- **Energy-dependent diffusion**
  For effective temperatures well below T_c, the diffusivity of quasiparticles is described by[24][25, eq3.8]

$$D_{\text{eff}} = D_N \sqrt{\frac{2k_B T_{eff}}{\pi\Delta}} \tag{4.19}$$

  Where D_N is the diffusivity of Fermi energy quasiparticles due to scattering with imperfections in the material, for which we have measurements.

  We use this relation to include a simple energy and location dependent D in our simulations as follows. At every timestep in our simulation, we calculate, on each volume element, the density of quasiparticles. That is the thermal equilibrium density n_{qp}, plus the excess density δn_{qp} due to the excitation. By inverting the relation for $n_{qp} = n_{qp}(T)$ given by [5, eq2.4] for each element,

using the `lambertw` function in SciPy, we obtain an effective temperature T_{eff} at each volume element. We can then use the relation above to calculate an energy dependent and location dependent diffusivity which we can use to calculate the next timestep in our simulation.

- **Initial condition and exponential forcing term**
  Unlike the simple analytical models we considered in section 4.3, in reality the delta peak is too rough of a simplification for an initial condition of the system. More accurately, as covered in chapter 3, if we consider that some diffusion of quasiparticles and phonons already took place during the downconversion, and if we assume the downconversion process is exceedingly fast compared to the recombination phase, we expect the initial condition to be some Gaussian distribution of excess quasiparticles, with a width σ_{IC}. The integral of this distribution represents the initial number of excess quasiparticles.

  However, in the downconversion chapter (3) we also concluded that we expect some overlap between the downconversion process and the start of the recombination process. Specifically, as the phonon distribution reduces exponentially with some time constant τ_{IC}, we expect a source term of excess quasiparticles which decays exponentially with the same time constant. In order to implement this, we define a source term which has a Gaussian shape spatially for some σ_{IC}, but which decreases exponentially with time. For this, we can simply multiply a given Gaussian IC by $H(t)\tau_{IC}^{-1}\exp(-t/\tau_{IC})$, where H is the step function. Multiplying by this normalized function ensures that the equivalent number of excess quasiparticles is introduced by the forcing term as the would have been introduced by the initial condition it replaces. In figure 4.5 the effect is shown that a forcing term slightly increases the pulse height and width compared to a simple IC.

- **2D-ness correction**
  Another assumption in our model is to model a narrow thin strip of superconductor as a 1D line. Depending on the width of the IC, this may initially not be the case. For example, imagine a cuboid piece of superconductor of the shape $100 \times 10 \times 0.04\mu m$ If the spot size of the IC after downconversion is much smaller than $10\mu m$, initial spatial effects may be too significant to ignore in a 1D model. We implemented the following approximate method to probe the significance of this effect. It is approximate, as we do not take into account border effects of the planar diffusion, let alone the effect of energy-dependent diffusion on the displacement.

  For a density of particles undergoing diffusion, the mean displacement of particles is $\text{MD}(t) = \sqrt{2nDt}$, where n is the number of dimensions in which the particles can diffuse. Therefore, for a distribution undergoing planar diffusion, the planar density must be proportional to time according to $\rho_p \propto \frac{1}{4Dt}$. Meanwhile, the linear density of our 1D model will be proportional to time according to $\rho_l \propto \frac{1}{\sqrt{2Dt}}$. We can therefore choose a fictitious diffusion $D_f(t) = 8D^2t$ in our 1D model for as long as the MD is smaller than half the width of the absorber. This emulates that, in the beginning of the simulation, the 1D density decreases with the same proportionality in time as the more realistic 2D density.

Optimizations

- **Symmetry**
  One simple optimization to make, provided that the pulse is centered in the middle of our simulation domain, is to take advantage of the symmetry. Simulating half the domain is a simple and 'lossless' way to cut down on computing time.

- **Adaptive timestep**
  Since the changes in the quasiparticle density are very rapid initially, it is crucial to take small timesteps in this part of the simulation. However, such small timesteps are wasteful for computation time once the faster-than-exponential phase of the decay is over. We implemented the following manner to adapt the timestep. The initially provided step dt is taken for the first timestep. Then we calculate the absolute change in quasiparticle number dN. After the first step we save the product $p = dNdt$. Initially the idea was to update the timestep according to

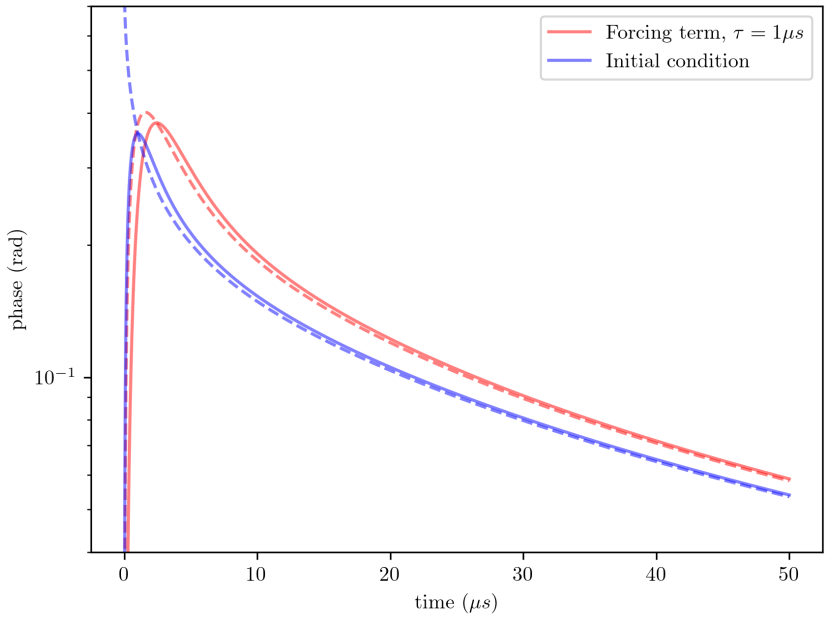


Figure 4.5: This graph shows two pulses which are equivalent except for the fact that one is simulated starting from a Gaussian IC, while the other has IC $\delta n_{qp} = 0$ and instead is excited by an exponentially decreasing Guassian forcing term. The dotted lines are the simulation time series, and the solid lines include the 'ringing' of the resonator.

$dt_{i+1} = p/dN_i$. This makes sure that the timestep increases as the changes dN become smaller, keeping the product approximately constant. However, we noticed some oscillation occurring, where dt started to alternate between a very small value one step and a large value the next. In order to counter this, we opted to update dt according to $dt_{i+1} = \frac{dt_i + p/dN_i}{2}$. This rolling average stabilizes the process. Because the timestep varies, we need to interpolate the signal to be able to calculate the 'ringing' convolution of equation 4.18. We will linearly interpolate with a stepsize of $dt_{interp} = 5ns$.

- **Adaptive geometry**
  The last optimization we implemented is to decrease how finely the x-axis is discretized as time progresses. The diffusion part of the PDE reduces the need for a high spatial resolution as time progresses. When a simulation starts, we first wait for several time constants of the source term to pass, so no significant localized peak is injected into the system anymore. After that we assume that the quasiparticle distribution gets wider according to the mean displacement (MD) due to a simple diffusion process. In 1D, this is $\text{MD}(t) = \sqrt{2Dt}$. For D, we use the minimum diffusivity. I.e. diffusivity calculated for a thermal quasiparticle distribution using equation 4.19. For the implementation, we compile a list which includes all possible values of dx, i.e. those that cleanly divide the domain length L_x. This list includes all values from the value closest to the requested initial dx value, to $dx = L_x$. Every time the MD becomes larger than one of these values, dx is updated accordingly.

While the symmetry optimization is exact, the two other ones potentially come at the cost of numerical errors. We test this in figure 4.6. This test is done for the same values of dx and dt which we will use in the following chapter. We can conclude that the optimizations do not significantly change the results of the simulation. Given the great improvement in simulation time, these optimizations will be necessary for the fitting procedures we will perform with this model.

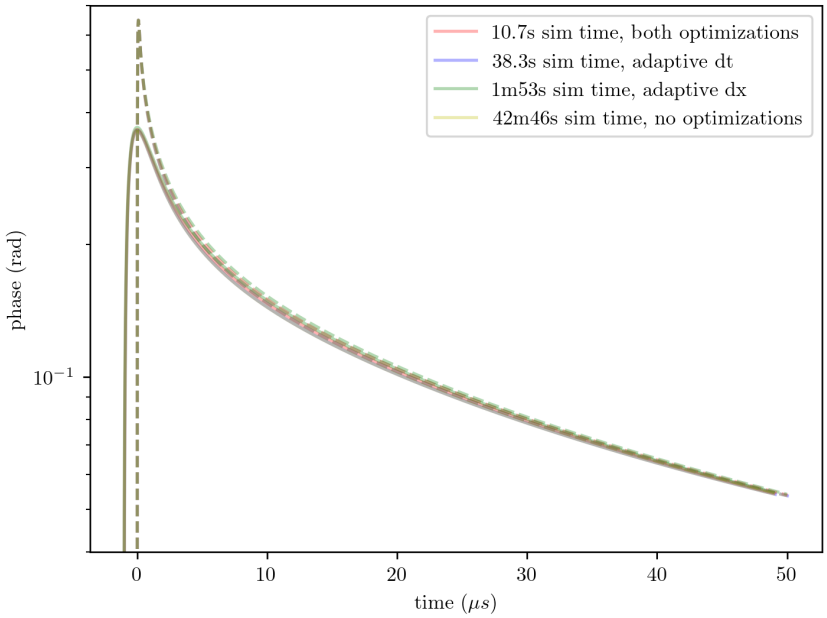


Figure 4.6: This graph shows that the simulations including the dt and dx optimizations do not differ greatly from the simulation without; they all overlap. The chosen values for these step sizes are $dt = 1ns$ and $dx = \sigma_{IC}/5$. The computation time is greatly reduced by these optimizations. All simulations above were done with the symmetry optimization. For each simulation, the plot including ringing is shifted so the maximum is at $t = 0$, unlike in figure 4.5.

Chapter 5

Experimental Evaluation of the Model

5.1 Introduction

The previous chapters have served to build a model of quasiparticle recombination and to formulate the choices and assumptions behind it. In this chapter we test how well the model we established can reproduce or predict the behavior of actual MKID measurements.

Firstly, we will present the parameters needed for the model. While we can obtain many from theoretical or measurement values, some, especially those to do with the downconversion process, have too much uncertainty to know in advance and are therefore obtained by fitting the model to the data.

This brings us to the second section, where we perform this fitting for two different MKIDs, and for various photon energies each. Based on what we obtain from these fits, we establish to what extent the model's behavior matches or differs from the data.

Lastly, we probe the role of diffusion in the duration of quasiparticle recombination. We do so by observing the pulses of MKIDs that we produced in which the lengths of the β-Ta absorption elements vary from a few μm to several hundred μm.

5.2 Parameters

The model constructed in the previous chapter relies on many parameters. We will go through them to assess which ones can be used from data or theory and which ones we cannot know in advance.

Known material properties

Below we list the known material parameters that we need to run the model.

- The electron diffusivity at the Fermi level which we use is $D_0 = 75\mu m^2/\mu s$. This was measured by the research group from the temperature dependence of the upper critical field[26].
- The single spin density of states at the Fermi level $N_0 = \nu_e = 30.2{\cdot}10^{-9} eV^{-1}\mu m^{-3}$ was calculated from the diffusivity by the Einstein relation[26].
- The critical temperature for each chip with MKIDs was measured for a separate layer of material deposited on the same wafer. For β-Ta: $T_c \approx 0.87K$.

Linear decay time

From the analyses in chapter 4, we know that at the tail end of each photon pulse, the pulse approaches the form of exponential decay with a rate $2R'Q_0$, with $R' = R/(1 + \frac{\Gamma_B}{\Gamma_{es}})$. In this expression, Q_0 is the equilibrium density of quasiparticles, and R, Γ_B, Γ_{es} are the recombination rate, the pair breaking rate and the phonon escape rate. As these are hard to quantify, it is easier to simply work with $2R'Q_0$ as a whole. Or equivalently, we can work with the effective quasiparticle decay time $\tau^*_{qp} = 1/(2R'Q_0)$. We obtain this value by fitting an exponential $\propto \exp(-t/\tau^*_{qp})$ to the tail end of a given measurement of a pulse.

Resonator response time

For high quality factor resonators such as MKIDs, a resonator's response time is dominated by its 'ringing' time. This depends on its quality factor and resonance frequency according to $\tau_{\text{ring}} = Q/(\pi f_0)$. Data for the quality factor and resonance frequency are available for each MKID, measured at multiple temperatures.

The argument to apply this response time assumes that a change in quasiparticles affects the resonance frequency only marginally. For the MKIDs we look at $\delta f/f_0 \approx 10^{-4}$. The total quality factor Q is composed of the coupling quality factor Q_c and the internal quality factor Q_i, which is the contribution that responds to a change in quasiparticles. Because $Q_c \ll Q_i$, $Q \approx Q_c$, so Q is approximately independent of the change in quasiparticles. All in all, a change in quasiparticles is therefore assumed not to significantly affect the ringing time. The electrical signal following the change of the resonator properties thus follows a ringing time that is approximately constant.

Pair breaking efficiency

The pair breaking efficiency η_{PB} describes what fraction of the photon energy makes it into the distribution of low energy quasiparticles at the start of the recombination process. For ordered conductors, for which Kozorezov's theory can be applied, an upper limit to this efficiency can be found at approximately 0.59[12]. As stated in the downconversion chapter (Ch. 3), repeating these calculations for β-Ta also begets a value $\eta_{PB} \approx 0.59$. However, this result relies on Kozorezov's theory for ordered superconductors, which we have shown is not applicable to β-Ta. In chapter 3 we have also reasoned that we expect the efficiency to be lowered by loss of phonons below 2Δ. In conclusion, we do not truly know the pair breaking efficiency of β-Ta in advance, and therefore we will leave it to be found as a fitting parameter constrained between 0 and 1.

IC width

In chapter 3, about the downconversion process, we have made calculations of the expected width σ_{IC} of the initial quasiparticle distribution or source term. Due to the lack of disordered theory for energy dependent phonon-electron scattering times, we approached the calculation of these times using two adapted theories. Using these scattering times, we calculated σ_{IC} ranging from $2\mu m$ to $17\mu m$. See table 3.4. Since, as explained in chapter 3, there is little reason to trust one of these calculations over the others, we are left with a high uncertainty in σ_{IC}. Therefore we leave this value as another fit parameter, which we bound to values above $0.2\mu m$. Any lower than that, the simulation becomes prohibitively slow.

IC time constant

A situation similar to that of σ_{IC} holds for the time constant τ_{IC}. This is the time constant of the forcing term that models the influx of excess quasiparticles from the downconversion to the recombination process. In chapter 3, on downconversion, we calculated different values for τ_{IC}, based on different calculation methods. The values calculated for τ_{IC} range from $4ns$ to $0.3\mu s$. As we do not have good arguments to put more trust into any one of these calculations, we also have a large uncertainty in τ_{IC}. Therefore we also leave this parameter as a fitting parameter. We bound the fit between $10ns$

and $10\mu s$, the lower bound was chosen to keep an order of magnitude above the simulation timestep $\Delta t = 1ns$. We choose an upper bound so that the fitting procedure does not attempt too high values of τ_{IC}. For high values, the code optimizations in adaptive dx are namely paused for a longer time, which makes the simulations prohibitively lengthy.

Thermal quasiparticle density

Another unknown is the thermal quasiparticle density Q_0. You may think we could calculate this from the temperature of the device, using [5, eq2.4]. However, there are a few things that cause uncertainty. First of all, this equation is extremely sensitive to small variations in temperature. For example, for a β-Ta MKID $10\mu m$ wide and $0.04\mu m$ thick, a linear density of $0.1\mu m^{-1}$ corresponds to a temperature of $106mK$, and a tenfold of that, $1\mu m^{-1}$, corresponds to a temperature of $125mK$. If only the temperature played a role, this could have been manageable, but the equilibrium amount of quasiparticles is also influenced by other factors such as e.g. the power of the readout signal[5, §4.5].

Evidence for this is given in figure 5.1. In this figure, we show that, for a particular MKID, $\tau^*_{qp} = 1/2R'Q_0$ is approximately constant for low temperatures. Because R' should not depend on temperature, the expected temperature dependence of Q_0 must not hold up. Since the MKID measurements we will use are conducted below $100mK$, and we do not have a complete overview of the other factors that could contribute to Q_0, we cannot calculate this value in advance.

As a result, we also opt for fitting the value for Q_0. We provide a lower bound of $0.1\mu m^{-1}$, to avoid extremely low Q_0. For exceedingly low Q_0, $R' = (\tau^*_{qp})^-1/2Q_0$ namely becomes exceedingly large, which causes the simulation to be more unstable due to the increased weight of the nonlinear term of the PDE (equation 4.9).

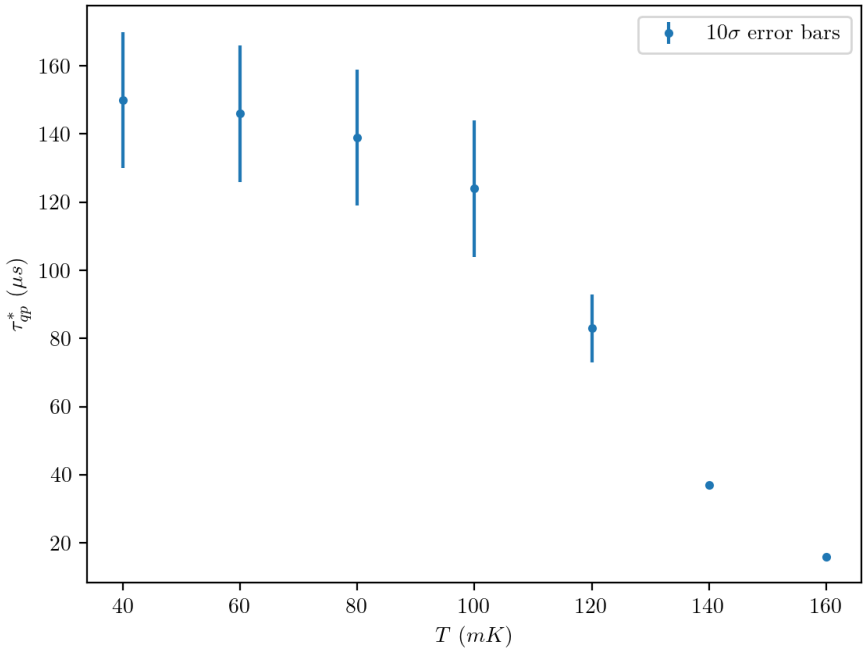


Figure 5.1: This graph shows τ^*_{qp} fitted to the tails of the pulses of MKID7, read out at -109dBm, excited by 673nm photons, on chip `bTa_LT278W2chip5`. $\tau^*_{qp} = 1/2R'Q_0$ stays approximately constant at some point as the temperature decreases. Because we expect the rate R' to remain constant, this suggests that Q_0 stops following the temperature of the device at some point.

Responsivity

The responsivity of the measured phase response to the number of excess quasiparticles is expressed as[5, eq2.27]

$$\frac{d\theta}{dN_{qp}} = -\frac{\alpha_k \beta Q}{\sigma_2 V}\frac{d\sigma_2}{dn_{qp}}.$$

Using Mattis-Bardeen theory, $\frac{d\sigma_2}{dn_{qp}}$ can be found to be approximately constant for low variations in n_{qp}[5, §2.2.2]. This assumption is dubious depending on how localized the excess quasiparticles are for slow diffusion in β-Ta, we will come back to this in section 5.3.2. A measurement value for the low-δn_{qp} responsivity is available for each MKID, which is what we will use for now.

5.3 Assessment of the Model's Predictive Power

In this section we will attempt to recreate the single photon response of two β-Ta MKIDs, for different photon energies. The specific MKID measurements we use are shown in table 5.1. We will run our model at the step sizes tested in figure 4.6: $dt_{init} = 1ns, dx_{init} = \sigma_{IC}/5$.

First we fit the model to single photon pulses. Then we check whether the fact that the model is 1D severely limits its validity to the data. Next we look at the photon energy dependence of the model, and how that compares to the data's photon energy dependence. Lastly we formulate two possible explanations to the discrepancy between the model's energy dependence and that of the data.

Chip	MKID	Temperature	P_{read}	β-Ta elm. length×width×thickness
`bTa_LT278W2chip5`	5	20mK	-99dBm	$90\mu m \times 10\mu m \times 0.04\mu m$
`bTa_LT354chip3`	14	20mK	-92dBm	$48\mu m \times 6\mu m \times 0.04\mu m$

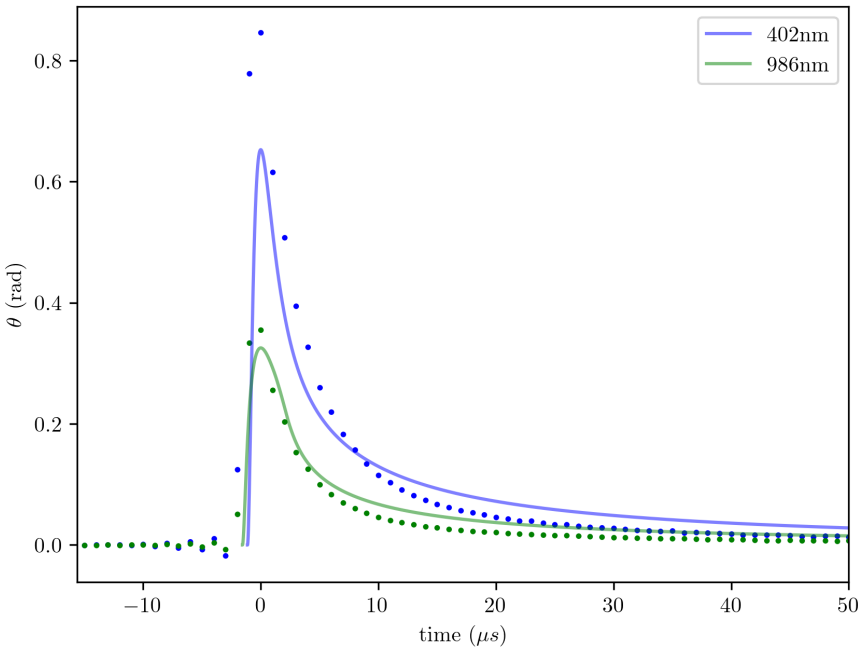
Table 5.1: The measurement data we will use in this section. Important for this section is that the β-Ta elements of the MKIDs have different line widths: $10\mu m$ for one and $6\mu m$ for the other.

5.3.1 Simulating single photon pulses

We will first attempt to recreate the single photon pulse measurements for the $10\mu m$-wide MKID of table 5.1. As argued in the previous section, we will fit $\eta_{PB}, \sigma_{IC}, \tau_{IC}$ and Q_0. All extensions and optimizations of the model are used by default, except the 2D-approximation. The exact fitting procedure is provided in Appendix D.

Single photon energy pulse

For the $10\mu m$-wide MKID, we fit the model to the average 986nm and 402nm photon pulse. The results are shown in figure 5.2.



λ_{ph}	986nm	402nm
η_{PB} (-)	0.9999 ± 0.0007	0.9 ± 1.7
σ_{IC} (μm)	10.7 ± 0.3	10 ± 46
τ_{IC} (μs)	0.4141 ± 0.0003	0.2 ± 0.2
Q_0 (μm^{-1})	0.53 ± 0.01	1.6 ± 0.6

Figure 5.2: Comparison of data and model for the shown parameters. The parameters were fitted for the $10\mu m$-wide MKID at 986nm and 402nm photon pulses.

Focusing on the 986nm pulse, we see that, although the model is able to reproduce the pulse reasonably well, the pulse height does not quite reach the experimental pulse height. And this undershoot happens even though the fitting algorithm ends up assuming near 100% pair breaking efficiency. This is unrealistic considering the $\sim 60\%$ limit for ordered superconductors[12][21], and given our reasoning in chapter 3 that we expect the pair breaking efficiency for disordered materials to be lower.

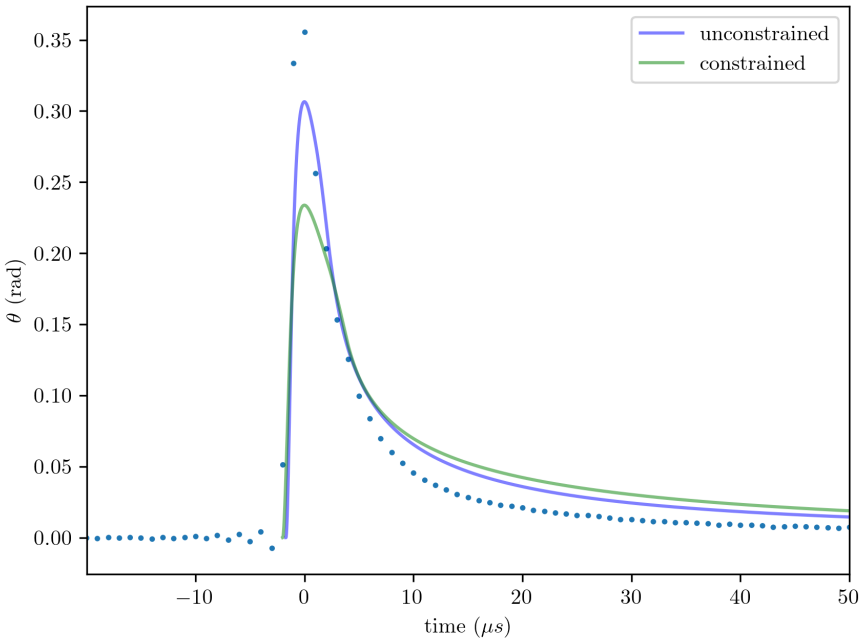
Now let us look at the performance of fitting the model to a higher energy photon pulse, at 402nm wavelength. This time the algorithm also tends to high pair breaking efficiency for the best fitting result. The found parameters have much larger uncertainties and the fitted model is less successful at matching the height of the peak in the data than in the lower energy case.

The influence of 2D diffusion

Our model is fundamentally one dimensional, so it cannot take any potentially important 2D diffusion effects into account. 2D diffusion, however, might be relevant if the width of the initial condition σ_{IC} is much narrower than the width of the absorber. Let us look into this potential contribution by two methods.

- **Initial 2D approximation**

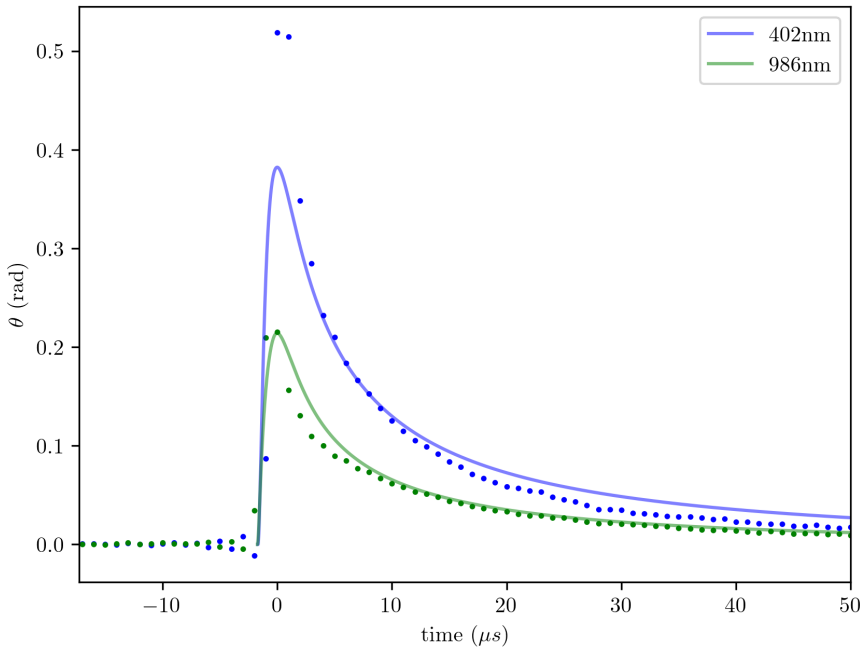
We can use the option we implemented in the model to introduce a 'correction' to the diffusivity. This option emulates the 2D diffusion when the quasiparticle distribution is more narrow than the absorber width. The fitting results for the model including this addition are shown in figure 5.3. When we fit this model to the low energy 986nm pulse, starting from an initial guess of $3\mu m <$ width/2, the fitting algorithm again finds an optimum fit with $\sigma_{IC} \approx 10\mu m >$ width/2. So any 2D effects are irrelevant for these found parameters. If we try fitting under the constraint that $\sigma_{IC} <$ width/2. We obtain the other fit in figure 5.3, which matches the data much worse. This allows us to gain some confidence that there does not exist another optimum fit below this constraint.



$\sigma_{IC} <$ width/2?	unconstrained	constrained
η_{PB} (-)	0.9999 ± 0.0007	0.9542 ± 0.0003
σ_{IC} (μm)	10.7 ± 0.3	3.131 ± 0.002
τ_{IC} (μs)	0.4141 ± 0.0003	0.7247 ± 0.0002
Q_0 (μm^{-1})	0.53 ± 0.01	0.7766 ± 0.0002

Figure 5.3: Fit of model with the initial 2D diffusion approximation enabled, both with and without constraining the fit to $\sigma_{IC} <$ width/2.

- **Narrow MKID data**
  We can also fit the model to data of the MKID design with a narrower line width of $6\mu m$, see table 5.1. This reduces the influence of 2D diffusion on the data. The results of this fitting procedure are shown in figure 5.4. We see that the same trends hold for this narrower device as for the previous device: The low energy photon pulse fits quite well, although with extremely high pair breaking efficiency. For the high energy photon pulse, the optimum fit undershoots the height of the data pulse.



λ_{ph}	986nm	402nm
η_{PB} (-)	0.90 ± 0.05	0.7 ± 0.3
σ_{IC} (μm)	6.1 ± 0.8	6 ± 7
τ_{IC} (μs)	0.0894 ± 0.0005	0.2 ± 0.2
Q_0 (μm^{-1})	4.8 ± 0.4	6.7 ± 0.8

Figure 5.4: Optimal fits of the narrow ($6\mu m$-wide) MKID for a 986nm and 402nm photon pulse

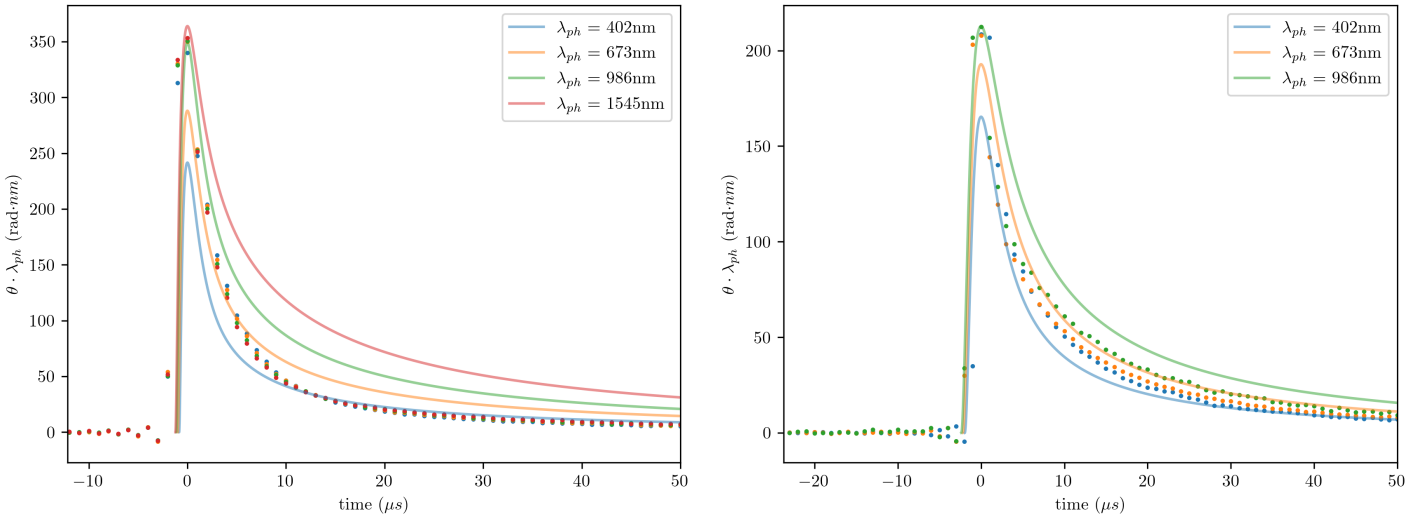
In conclusion of the above: The 2D correction does not seem to improve the optimum fit of the $10\mu m$-wide MKID, as the optimal σ_{IC} that is found is far larger than $5\mu m$. As for the narrow MKID data, we can see the same trends in the results as before: 1) The pair breaking efficiency must be near 100% to fit the data 2) The model cannot reproduce the behavior of a high photon energy pulse. Therefore we can cautiously infer that the ignorance of 2D diffusion in the initial phase of the recombination process, is not the most important inaccuracy of our model.

Pulses of multiple photon energies

So far, we have only investigated how well the model can reproduce pulses of a single photon energy. Next we will investigate the photon energy dependence of the pulses that the model can simulate. We do so by fitting the model to pulses of multiple photon energies concurrently. For visualizing the plots of this section, we 'normalize' the plots of pulses by the respective phonon energy, by multiplying the phase of each point in the plot by the respective photon wavelength. The results of the fitting procedure for both the $6\mu m$ and $10\mu m$-wide MKIDs are shown in figure 5.5. For these fits we see that the simulated pulses are photon energy dependent, while the data is not.

Besides the discrepancy in dependence on photon energy, we again observe in all simulations of this section that the model can reproduce the peak heights of low photon energy data reasonably well, although with high pair breaking efficiency. The model cannot capture the peak heights of the higher

energy photon pulses.



MKID	10μm-wide	6μm-wide
η_{PB} (-)	0.95 ± 0.05	0.9 ± 0.2
σ_{IC} (μm)	11 ± 2	8 ± 5
τ_{IC} (μs)	0.0797 ± 0.0003	0.5 ± 0.1
Q_0 (μm^{-1})	1.0 ± 0.1	5.0 ± 0.3

Figure 5.5: Optimal concurrent fit of the wide (10μm-wide) and narrow (6μm-wide) MKID for all available photon wavelengths.

5.3.2 Analysis and conclusion

We observed two main aspects in running the model that point towards the idea that the model does not completely capture the behavior of the actual β-Ta pulses. The first is that the model does not manage to match the pulse height for the higher energy photons, even when assuming suspiciously high pair breaking efficiency. Secondly, the model does not produce pulse shapes that are independent of the photon energy, while the pulse data *is* photon energy independent.

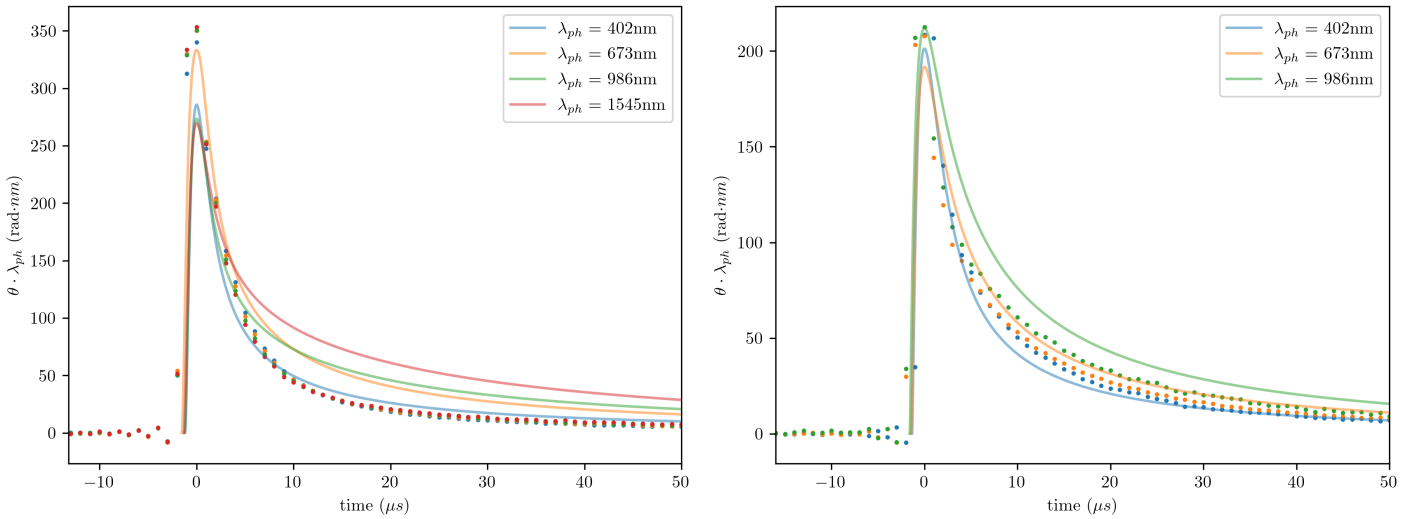
It is rather remarkable that the pulse shapes of the measured β-Ta MKIDs do not appear to be dependent on photon energy. After all, the PDE that we solve using the model is non-linear. One solution scaled by a constant should not generally be another solution to the PDE. Since we do observe this in the data, there must be other aspects to the recombination process that are dependent on the photon energy.

Photon energy dependent downconversion

One possibility is that the downconversion has photon energy dependent properties. If that is the case, the model parameters σ_{IC}, τ_{IC} and η_{PB} which we fit could be dependent on the photon energy. Since η_{PB} maxes out to fit the data in the previous sections, there is little leeway to vary it. For now we will also assume σ_{IC} plays a larger role than τ_{IC}, on account of its higher uncertainties in the multi-energy fits in figure 5.5. Therefore, let us investigate the potential photon energy dependence of the width

of the initial condition. To do so we will repeat the procedure for fitting pulses of multiple phonon energies, but this time fit a separate σ_{IC} for each wavelength of photon.

The results of these fits are shown in figure 5.6, and the energy dependence of σ_{IC} suggested by these fits is shown in figure 5.7. From these results, one could argue that the energy-dependent σ_{IC} makes for a better overlap of the normalized pulses, indicating that part of the missing energy dependence of the model could hide in the downconversion phase. If this is the case, these fitting results suggest the trend that σ_{IC} increases as photon energy increases. One possible mechanism for this would be that, during the downconversion, a higher photon energy excites quasiparticles of higher energy, which could be more mobile and therefore contribute to a larger σ_{IC}.



MKID	$10\mu m$-wide	$6\mu m$-wide
η_{PB} (-)	0.99 ± 0.06	0.99 ± 0.06
σ_{402nm} (μm)	15 ± 3	12 ± 8
σ_{673nm} (μm)	12 ± 2	6 ± 2
σ_{986nm} (μm)	4 ± 2	6 ± 2
σ_{1545nm} (μm)	1 ± 5	-
τ_{IC} (μs)	0.283 ± 0.002	0.052 ± 0.003
Q_0 (μm^{-1})	1.0 ± 0.1	5.0 ± 0.2

Figure 5.6: Optimal concurrent fit of the $10\mu m$-wide MKID (left) and of the $6\mu m$-wide MKID (right) for all available photon wavelengths, with a separate σ_{IC} per photon energy.

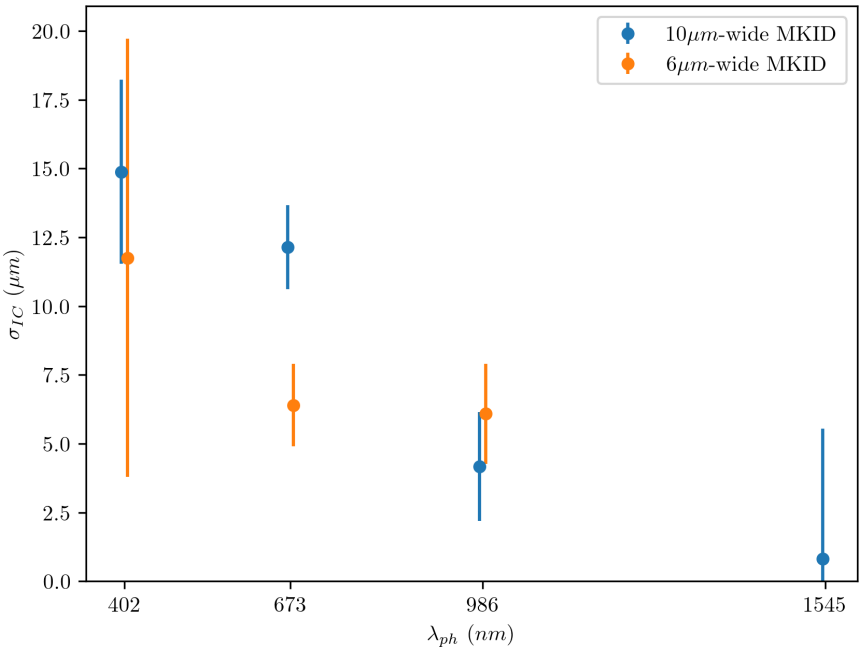


Figure 5.7: Here we plot the photon wavelength against the fitted initial condition width. The fitting results suggest that, *if* this is indeed the lacking energy dependence in our model, then the spread of the IC must be larger for larger photon energy.

Photon energy dependent responsivity

The main issue we saw with the model is that it has difficulty reaching the same peak heights as the data, even when the pair breaking efficiency is taken to be almost 100%. In trying to fit to the data height, it is possible that accuracy to the rest of the curve is sacrificed. We propose that the explanation may lie in nonlinearities in the responsivity. As stated before, in our model we assume the responsivity $\frac{d\theta}{dN_{qp}}$ is approximately constant for low δn_{qp}, and for $\theta < 1$ rad.

Let us investigate the assumption of low excess quasiparticles. The responsivity can be written as[5, eq2.27]

$$\frac{d\theta}{dN_{qp}} = -\frac{\alpha_k \beta Q}{\sigma_2 V}\frac{d\sigma_2}{dn_{qp}}.$$

In this equation, σ_2 is dependent on the quasiparticle density according to figure 5.8. Above approximately an effective temperature of $0.4T_c$, the responsivity is not linear.

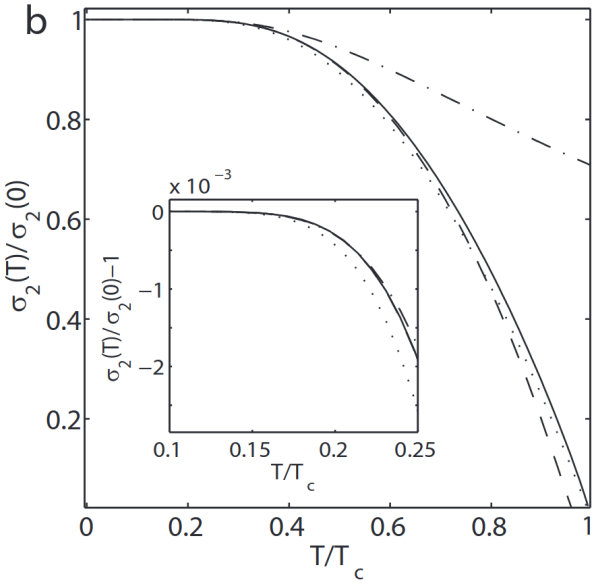


Figure 5.8: The solid line marks the dependence of the imaginary part of the conductivity, σ_2, on the (effective) temperature which corresponds to the density of quasiparticles, as calculated using the Mattis-Bardeen equations[5, eq2.12]. Figure taken from [5, §2.2.1].

Let us check the phase and the maximum T_{eff} that we expect locally for a $402nm$ excitation. We will calculate the expected maximum excess quasiparticle density for a 40nm thick β-Ta sheet, sweeping the value of σ_{IC}. This calculation is performed in the code in appendix B.

1. We assume a near-optimal pair breaking efficiency of $\eta_{PB} \approx 0.6$[12][21]. Which means that after downconversion we obtain approximately $0.6 \cdot E_{ph}/\Delta = 13961$ low energy excess quasiparticles.
2. Now we assume $\theta_{\max}$ occurs when δN_{qp} is distributed according to a 2D Gaussian of width σ_{IC}. Then, in the center of the Gaussian, the maximum excess planar density is found according to:

$$\delta n_{\text{qp,max}} = \frac{\delta N_{\text{qp,max}}}{2\pi\sigma_{IC}^2}$$

3. Lastly, we invert $n_{qp}(T_{\text{eff}})$[5, eq2.4] (multiplied by MKID film thickness so we deal with planar density). Figure 5.1 suggests an effective temperature of thermal n_{qp} is given at $T_{\text{eff}} \approx 100mK$. We plug in $\delta n_{\text{qp,max}} + n_{\text{qp,0}}$ to obtain the maximum expected local effective temperature. The results of this calculation are given in figure 5.9.

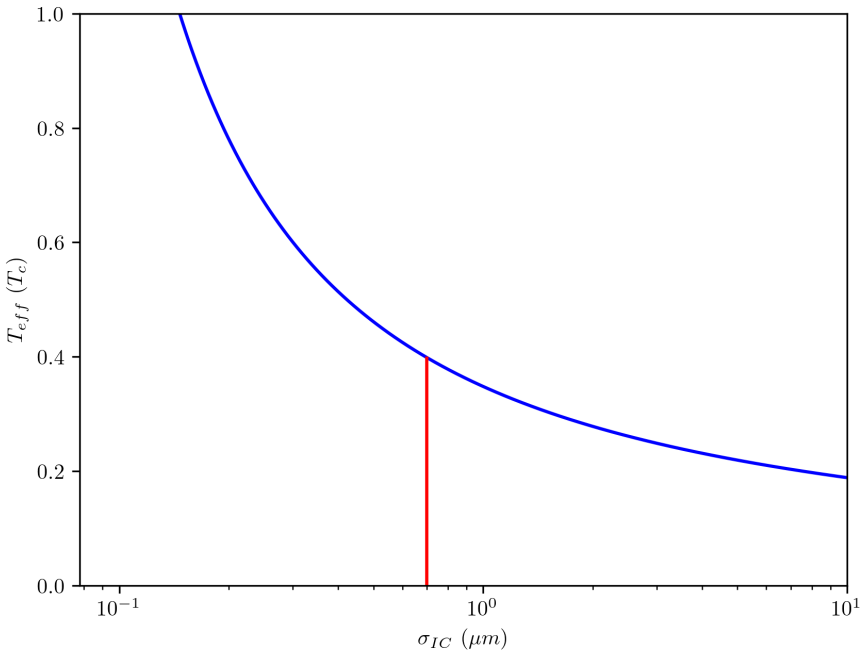


Figure 5.9: The resultant effective temperature as fraction of T_c, sweeping σ_{IC}. We see that for $\sigma_{IC} < 0.7$, the effective temperature is larger than $0.4T_c$, which would cause non-linear responsivity.

Depending on what the exact values for σ_{IC} and η_{PB} turn out to be, the constant responsivity assumption in our model may be invalid. This would not be the case for any of our calculations in section 3.6.2 of the chapter on downconversion. The potentially high T_{eff} may also pose another issue: The expression for energy dependent diffusion that we use, is dependent on the condition $T_{\text{eff}} \ll T_c$[25, eq3.8]. Therefore we may also need to look into a more general expression for energy dependent diffusion.

5.4 Assessment of Diffusion and Volume Constraint in MKIDs

Even though we are dealing with a model whose correspondence with the data is limited, we will apply it to investigate the influence that the diffusion in our model has on the decay time of a photon pulse. We do so as a measurement of how faster-than-exponential the decay is.

5.4.1 Experimental setup

We will study β-Ta MKIDs that each contain the same volume of β-Ta absorber, but the absorbers are subdivided into varying sub-elements of β-Ta. The β-Ta elements are separated by strips of the higher T_c NbTiN to ensure the quasiparticles excited in one of them are trapped there. The lumped element MKIDs are designed to have approximately constant current density across the absorber, so that the photon response should be the same regardless of which element absorbed the photon. The β-Ta absorbers of the different MKIDs have a line width of $6\mu m$ and a thickness of $40nm$. The MKIDs have sublengths varying from $6\mu m$ to $930\mu m$, and were designed for both a low quality factor $Q \approx 20000$ and a high quality factor $Q \approx 60000$. See figure 5.10 for an example of such an MKID and appendix E for an overview and more info of the MKIDs whose data we used. The MKIDs' single photon responses were measured at 402nm, 673nm and 986nm light, at 20mK, at various readout powers, and at a sampling frequency of 1MHz and 4MHz. Unfortunately, many of the measurements taken were unusable, especially those at 4MHz sampling. We discarded the data for pulses whose decay did not appear to be monotonous.

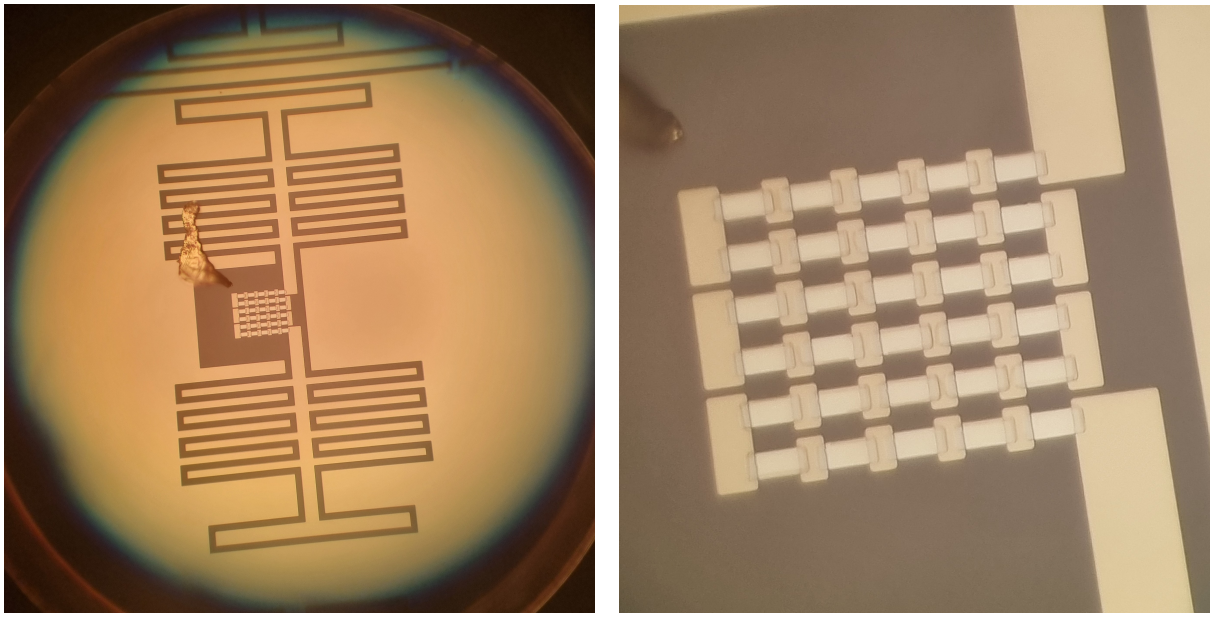


Figure 5.10: Example of one of the produced MKIDs. Left is the entire MKID, right is a zoomed in picture of the absorptive element. The lighter colored metal is β-Ta, the more yellowish material is NbTiN, and the dark area is the substrate. Note the distribution of the IDC on both sides of the absorber, which is beneficial to the homogeneity of the current throughout the absorber. The flake of material on top of the device is some unwanted piece of dust that likely ended up there after the measurements were taken.

5.4.2 Analysis

We compare the degree of faster-than-exponential decay occurring in these MKIDs to the predictions of our diffusion model. If the slow diffusion of quasiparticles is primarily responsible for the faster-than-exponential decay of the signals of β-Ta MKID, both the data and the model should show the following: The decay time should increase for larger sublength, until the sublength becomes so large, that it is the diffusion that constrains the space over which the quasiparticles are spread out. We expect this because the decay rate relates to the quasiparticle density[5, eq2.29]. For small volumes, the quasiparticles are constrained by the MKID itself, for larger volumes, at some point, the only constraint in the spread of quasiparticles could be due to slow diffusion.

In order to do this comparison, we scale the phase data of both the model and the MKIDs so that the maxima are set to $\theta = 1$rad. We then record the time until the threshold $\theta = 0.1$rad is reached starting from the peak maximum. If there is more faster-than-exponential behavior, the overall decay is faster and the threshold will be reached sooner. For the model, we use the parameters of the fit we already made for one of the sub-volume devices, namely that of figure 5.4. We will run this model for various lengths and apply the same thresholding to probe its decay rate. The 0.1rad threshold is a rather arbitrary number chosen as low number that still seemed to be above the noise level of the pulse tails. For different choices between ca. 0.01 and 0.2rad, we observed similar trends, shifted to lower or higher threshold times.

First let us take a look at the raw data. We will limit ourselves to the low quality factor MKIDs for now. For all measurements of these MKIDs, we normalize the phase data to 1rad at the maximum. Then, for each KID, we average all these normalized pulses. We plot these average pulses in figure 5.11. We see that almost all data points in this figure follow the trend that a higher sublength is related to a slower decay.

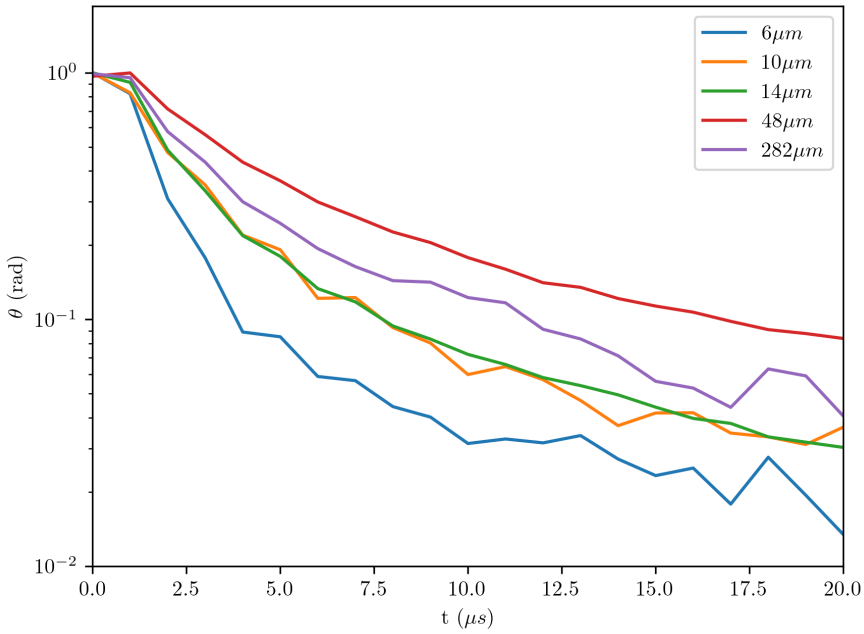


Figure 5.11: The shape of the signal decay, normalized to 1rad per measurement and averaged for each sublength. We see that the decay appears faster for smaller sublengths. This trend holds except for the order of the $48\mu m$ and $282\mu m$ lines. In this figure we only included data from the low quality factor MKIDs.

In figure 5.12 we show the threshold times of all available data for each wavelength of light, as well as the model's threshold time simulated for a 986nm photon pulse. We observe an initial increase in decay time that matches the model's trend until $48\mu m$. Due to the diffusion in the model, we expect the increase in decay time to saturate for large enough lengths. This trend also saturates for the data, but the threshold time even decreases, which is not something we expect. Perhaps this experiment should be repeated, especially for the higher sublength MKIDs. This may allow us to conclude whether the inflection in the decay time of our actual MKIDs corresponds with the inflection expected for the diffusivity of β-Ta.

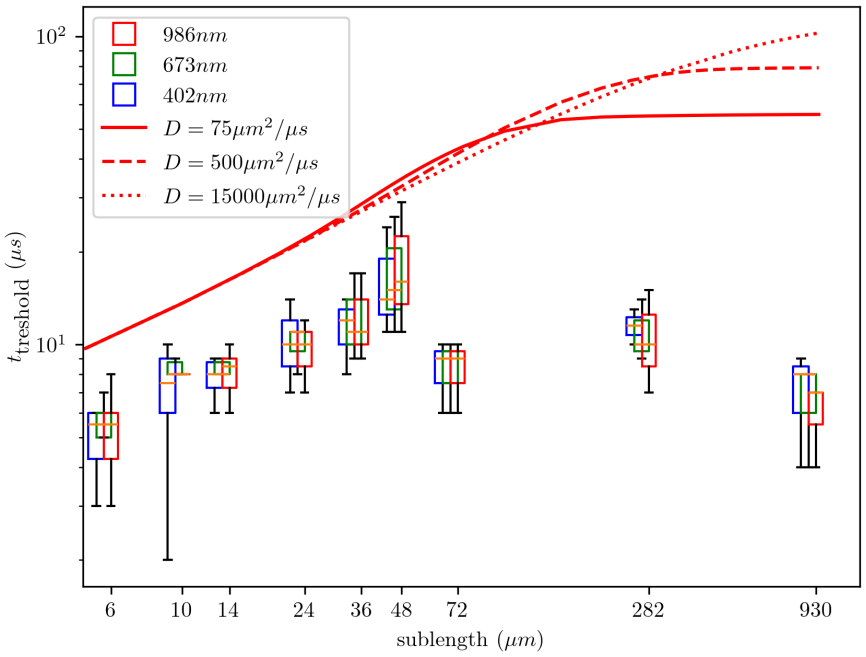


Figure 5.12: Comparison of the diffusion model's decay time until the 0.1rad threshold and that of the MKID measurements. The model was fitted to 986nm data of the 48μm sublength MKID. We ran this model for three different diffusivities, the lowest corresponding to β-Ta and the highest to Al. The different colors of boxes are shifted slightly besides each other for legibility, they belong to the same sublength.

5.4.3 Conclusion

The data from the low sublength MKIDs seem to suggest a rise in decay time as the β-Ta elements' lengths increase, which we also expect from the model. The increase in decay time also appears to saturate for both the data and the model. However, the MKIDs of longer β-Ta length have surprisingly fast decay compared to the model's prediction. It would require more trustworthy measurement data to make a solid conclusion, but the limited results seen here suggest that the decay time, and therefore the more-than-exponentiality of the quasiparticle recombination, is closely related to the diffusion.

5.5 Summary of conclusions

The results of the investigations in this chapter can be summarized in three parts:

1. The model cannot reproduce the pulse heights of the data without assuming unfeasibly high η_{PB}.
2. The pulse shapes produced by the model are dependent upon photon energy, whereas the measured pulses of β-Ta MKIDs do not have this dependence.
3. In MKIDs with different lengths of β-Ta elements, the decay gets slower as the length increases. Our model predicts a saturation to this trend, and the data confirms such a saturation. This in turn suggests that diffusion indeed plays a key role in causing the faster-than-exponential decay.

Chapter 6

Conclusions, Discussion and Outlook

In this chapter, we first summarize the conclusions that were made throughout this thesis. Then we make note of relevant points of discussion, such as assumptions made for our model or observed discrepancies between the model and data. Lastly, we suggest future research into some of these points of discussion.

6.1 Conclusions

6.1.1 Downconversion

In section 3.3, we show that the downconversion process as described by Kozorezov[9] cannot directly be applied to β-Ta. We therefore choose to qualitatively focus on the last two stages of the downconversion, which are respectively dominated by the electron-phonon scattering time and the phonon-electron pair-breaking time. We note a lack of theory for disordered materials that would give energy dependent theoretical expressions of these rates. In section 3.4 we therefore work out two methods to approximate these scattering rates for β-Ta.

In section 3.6.1, based on these calculations for the scattering rates, we show that the transition energy Ω_1 between the last two downconversion stages occurs near or even below the energy 3Δ which marks the end of the downconversion process. Therefore we can expect the last stage of the downconversion, in which many low energy quasiparticles are created, to continue into the recombination phase. As the phonon distribution at the end of Stage 3 decays exponentially with the pair-breaking time [9, eq17], we decide to simulate this overlap using an exponentially decaying influx of quasiparticles in the recombination model that we create.

In section 3.6.2, we use Kozorezov's expressions[9] for the calculation of the duration of the downconversion stages. In combination with our rate calculations, we then calculate values that describe the expected final distribution of quasiparticles after the downconversion. The final distribution generated by the downconversion corresponds to the initial condition for the recombination process. Our estimations for the width σ_{IC} of this distribution range from $2\mu m$ to $17\mu m$. Likewise, we calculate values for the time constant τ_{IC} of the abovementioned exponentially decaying quasiparticle source. We calculate values for τ_{IC} ranging from $4ns$ to $0.3\mu s$. Lastly, we apply the code of [12], to calculate the pair breaking efficiency η_{PB} for β-Ta. This calculation gives a near-ideal value of $\sim 59\%$. However, we argue that this calculation is not applicable to disordered materials either, and in section 3.6.1 we reason that we expect η_{PB} to be lower than ideal for β-Ta.

6.1.2 Experimental Evaluation of the Model

After building the model of the recombination process, we test it against measurements of the single photon pulses of β-Ta MKIDs for various photon energies. In section 5.2, we conclude that four parameters of the model cannot be accurately determined. These are the parameters σ_{IC}, τ_{IC} and η_{PB}, mentioned in the downconversion section, as well as the steady state quasiparticle density Q_0.

In section 5.3, we therefore use these as fit parameters to model the single photon pulses of β-Ta MKIDs. We show that the model can quite accurately reproduce low energy photon pulses ($\lambda_{ph} = 986nm$), but only when assuming a near 100% pair breaking efficiency. For high energy photon pulses ($\lambda_{ph} = 402nm$) the model is unable to reproduce the pulse height of the data.

Testing the model for pulses of different photon energies reveals that the pulse shapes generated by the model are dependent on photon energy, whereas the pulses from the data are not. This behavior of the data is remarkable, because the PDE that we solve in the model is nonlinear and therefore multiplying one solution by a scalar should not generally also be a solution to the PDE. Therefore we conclude that there must be other photon energy dependent aspects relevant to the recombination process which we have not taken into account. In section 5.3.2, we provide some arguments that these energy dependent aspects could lie in the downconversion process or in the nonlinearity of the MKID's responsivity.

In section 5.4, we investigate the single photon pulses of MKIDs whose absorber is composed of multiple small β-Ta elements. These elements constrain the quasiparticles in space. We show that the decay time of these MKIDs is longer for β-Ta elements that are larger in size. However, this correspondence eventually saturates for large enough β-Ta elements. With our model, we show that such an increase and subsequent saturation is indeed expected for a material with slow diffusion. From this correspondence we conclude that the diffusion indeed plays a significant role in the cause of faster than exponential decay in β-Ta OKIDs.

6.2 Discussion

6.2.1 Downconversion

Lack of disordered theory

Across the entirety of our treatment of the downconversion process, we run into the issue that there is no comprehensive theory available for the downconversion process in disordered superconductors. One downside of this is that there was no established calculation available for the pair breaking efficiency of β-Ta. Therefore we had to substitute a calculation meant for ordered superconductors.

Anharmonicity

In sections 3.3 and 3.6.1, we make no explicit calculations, but we make conjectures about the possible effects of the anharmonic phonon scattering on the downconversion process.

Uncertainty in rate calculations

Regarding our rate calculations in section 3.6.1, we find ourselves in the unfortunate situation that for aluminum, the electron-phonon rates calculated using Pippard (section 3.4.2) did not match the pair breaking rate from Kaplan (section 3.4.1), which is valid for aluminum. Therefore, we cannot trust either calculation method at face value for β-Ta. Thus, because of the large spread in the calculated β-Ta rates, our corollary calculations for σ_{IC} and τ_{IC} were also highly uncertain.

6.2.2 Recombination Model

One dimensionality of the model

One of the main assumptions in our model for the quasiparticle decay is that we can approximate the β-Ta absorber as a one dimensional domain. Indeed, running the model for a $10\mu m$-wide as well as a $6\mu m$-wide MKID in section 5.3 showed similar performance of the model in both cases. This allows us to assume that ignorance of higher dimensional diffusion effects is likely not the most important inaccuracy of our model

Energy dependent diffusion

Another assumption of the model that deserves some attention is our method of implementing location and energy dependent diffusion, according to equation 4.19. The condition for this relation to hold is that $T_{eff} \ll T_c$. This may not hold up, depending on the exact width σ_{IC} of the initial condition, see figure 5.9. Moreover, for our implementation of this energy dependent diffusion, we assume that locally the quasiparticles' energy follows a thermal distribution. We did not check the validity of this assumption.

6.2.3 Experimental Evaluation of the Model

Data alignment in the fitting method

In our routine for fitting the recombination model to the data, we shift both the model and data so that their maxima are located at $t = 0$. However, the decay has a very sharp peak, which we only sample at 1MHz. Therefore, the highest data point may not coincide with the true location of the peak maximum. You can see this occurring in e.g. the 986nm photon pulse in figure 5.4.

Implementation of the ringing time

In our attempts at fitting our model to the data, we repeatedly observe that an excessively high η_{PB} is needed to recreate the pulse data. In our modeling efforts we notice that the inclusion of the MKID's ringing time can severely attenuate our results as well as widen the peak. See figures 4.4, 4.5 and 4.6. We would have liked to take a further look at the response time of MKIDs and re-evaluate whether this is correctly implemented in our model.

Nonlinearity of the MKID responsivity

Throughout this thesis, we use the assumption that the responsivity $d\theta/dN_{qp}$ is constant. At the end of section 5.3.2, we investigate one of the necessary conditions for this: Whether the excess density of quasiparticles is small enough that $T_{eff} \ll T_c$. We show that, for a 402nm photon and for 60% pair breaking efficiency, this is not the case if $\sigma_{IC} < 0.7$. Therefore, depending on the specifics of the downconversion process, the responsivity may behave nonlinearly.

6.3 Outlook

Possible sources of photon energy dependence

Of all the complications discussed in the previous section, we believe the following two interrelated problems are most interesting to investigate further.

1. We should assess the height of the local maximum quasiparticle density caused by a photon absorption. If this density corresponds to an effective temperature that violates the condition $T_{eff} \ll T_c$, we must take into account nonlinear effects in the responsivity of the MKID, and we must rethink our implementation of the energy dependent diffusion.

2. The local maximum quasiparticle density depends on the initial distribution width σ_{IC} and on the pair breaking efficiency η_{PB}. Therefore it would also be necessary to look into methods to accurately describe the downconversion process for disordered materials like β-Ta. Besides, it would be interesting to see whether there indeed is a photon energy dependent aspect of the downconversion, as was suggested in section 5.3.2.

Anharmonicity in the downconversion process

Another more specific direction of inquiry that we suggest is relevant to the downconversion, is the effect of anharmonic phonon scattering. In particular, it would be interesting to see its effects on the rate of phonon downconversion. It would also be valuable to see its potential effect of 'losing' phonons that downconvert to energies below 2Δ, as this would decrease the pair breaking efficiency.

Increasing phonon diffusion

Lastly, in section 3.3, we argued that the phonon diffusion during downconversion is not negligible for β-Ta. For our treatment of this, we assumed that the most significant contribution to scattering the phonons are the outer edges of the material. Especially the thickness of material contributes to a lower mean free path and therefore to a reduction in the diffusion. If this assumption behind calculating the phonon diffusivity holds up, it would be interesting to try to make use of this by making MKIDs with thicker β-Ta films. This would namely allow for faster phonon diffusion and therefore increase the spread of the generated quasiparticles. In turn this could reduce their rate of decay, which would be beneficial to the energy resolution.

Bibliography

[1] M. Tinkham. *Introduction to Superconductivity*. McGraw Hill, 1996. ISBN 978-0-07-064878-4.

[2] N. Kovaleva, D. Chvostova, and A. Dejneka. Localization Phenomena in Disordered Tantalum Films. *Metals*, 7(7):257, July 2017. ISSN 2075-4701. doi: 10.3390/met7070257. URL https://www.mdpi.com/2075-4701/7/7/257. Number: 7 Publisher: Multidisciplinary Digital Publishing Institute.

[3] K. Kouwenhoven, D. Fan, E. Biancalani, S.A.H. De Rooij, T. Karim, C.S. Smith, V. Murugesan, D.J. Thoen, J.J.A. Baselmans, and P.J. De Visser. Resolving Power of Visible-To-Near-Infrared Hybrid β-Ta / NbTiN Kinetic Inductance Detectors. *Physical Review Applied*, 19(3):034007, March 2023. ISSN 2331-7019. doi: 10.1103/PhysRevApplied.19.034007. URL https://link.aps.org/doi/10.1103/PhysRevApplied.19.034007.

[4] P.J. De Visser, S.A.H. De Rooij, V. Murugesan, D.J. Thoen, and J.J.A. Baselmans. Phonon-Trapping-Enhanced Energy Resolution in Superconducting Single-Photon Detectors. *Physical Review Applied*, 16(3):034051, September 2021. ISSN 2331-7019. doi: 10.1103/PhysRevApplied.16.034051. URL https://link.aps.org/doi/10.1103/PhysRevApplied.16.034051.

[5] P.J. de Visser. *Quasiparticle dynamics in aluminium superconducting microwave resonators | TU Delft Repository*. PhD Thesis, TU Delft, Delft, 2014. URL https://resolver.tudelft.nl/uuid:eae4c9fc-f90d-4c12-a878-8428ee4adb4c.

[6] T. Rugers. *Quasiparticle Dynamics in Disordered Superconductors*. BSc Thesis, TU Delft, Delft, 2022. URL https://resolver.tudelft.nl/uuid:3097e7ce-73aa-4312-ad5c-eb105cc97091.

[7] P.C.J.J. Coumou. *Electrodynamics of strongly disordered superconductors*. PhD Thesis, TU Delft, Delft, 2015. URL https://resolver.tudelft.nl/uuid:ad01901d-511f-4496-ae66-301d0db92caa.

[8] P.K. Day, H.G. LeDuc, B.A. Mazin, A. Vayonakis, and J. Zmuidzinas. A broadband superconducting detector suitable for use in large arrays. *Nature*, 425(6960):817–821, October 2003. ISSN 0028-0836, 1476-4687. doi: 10.1038/nature02037. URL https://www.nature.com/articles/nature02037.

[9] A. G. Kozorezov, A. F. Volkov, J. K. Wigmore, A. Peacock, A. Poelaert, and R. den Hartog. Quasiparticle-phonon downconversion in nonequilibrium superconductors. *Physical Review B*, 61 (17):11807–11819, May 2000. doi: 10.1103/PhysRevB.61.11807. URL https://link.aps.org/doi/10.1103/PhysRevB.61.11807. Publisher: American Physical Society.

[10] A. Kozorezov. Energy Down-Conversion and Thermalization in Metal Absorbers. *Journal of Low Temperature Physics*, 167(3):473–484, May 2012. ISSN 1573-7357. doi: 10.1007/s10909-011-0426-1. URL https://doi.org/10.1007/s10909-011-0426-1.

[11] A.G. Kozorezov, C. Lambert, F. Marsili, M.J. Stevens, V.B. Verma, J.P. Allmaras, M.D. Shaw, R.P. Mirin, and S.W. Nam. Fano fluctuations in superconducting-nanowire single-photon detectors. *Physical Review B*, 96(5):054507, August 2017. doi: 10.1103/PhysRevB.96.054507. URL

https://link.aps.org/doi/10.1103/PhysRevB.96.054507. Publisher: American Physical Society.

[12] A. Dantuma. *Pushing the Fano limit*. BSc Thesis, TU Delft, Delft, 2024.

[13] T. Held, I. Pfeiffer, and W. Kuhn. Influence of isotopic disorder on phonon frequencies and phonon linewidths of an anharmonic crystal. *Physical Review B*, 55(1):231–236, January 1997. doi: 10.1103/PhysRevB.55.231. URL https://link.aps.org/doi/10.1103/PhysRevB.55.231. Publisher: American Physical Society.

[14] H.R. Seyf, L. Yates, T.L. Bougher, S. Graham, B.A. Cola, T. Detchprohm, M. Ji, J. Kim, R. Dupuis, W. Lv, and A. Henry. Rethinking phonons: The issue of disorder. *npj Computational Materials*, 3(1):1–8, November 2017. ISSN 2057-3960. doi: 10.1038/s41524-017-0052-9. URL https://www.nature.com/articles/s41524-017-0052-9. Publisher: Nature Publishing Group.

[15] S.H. Simon. *The Oxford solid state basics*. Oxford University Press, Oxford, 2013. ISBN 978-0-19-968076-4.

[16] N.W. Ashcroft and N.D. Mermin. *Solid State Physics*. Harcourt College Publishers, 1976. ISBN 0-03-083993-9. URL http://archive.org/details/AshcroftSolidState.

[17] M.Y. Reizer and A.V. Sergeyev. Electron-phonon interaction in impure metals and superconductors. *Journal of Experimental and Theoretical Physics*, 63, 1986. URL http://jetp.ras.ru/cgi-bin/dn/e_063_03_0616.pdf.

[18] B. Keck and A. Schmid. Superconductivity and electron-phonon interaction in impure simple metals. *Journal of Low Temperature Physics*, 24(5):611–629, September 1976. ISSN 1573-7357. doi: 10.1007/BF00657170. URL https://doi.org/10.1007/BF00657170.

[19] S.B. Kaplan, C.C. Chi, D.N. Langenberg, J.J. Chang, S. Jafarey, and D.J. Scalapino. Quasiparticle and phonon lifetimes in superconductors. *Physical Review B*, 14(11):4854–4873, December 1976. ISSN 0556-2805. doi: 10.1103/PhysRevB.14.4854. URL https://link.aps.org/doi/10.1103/PhysRevB.14.4854.

[20] A.B. Pippard. CXXII. Ultrasonic attenuation in metals. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 46(381):1104–1114, October 1955. ISSN 1941-5982. doi: 10.1080/14786441008521122. URL https://doi.org/10.1080/14786441008521122. Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/14786441008521122.

[21] T. Guruswamy, D. J. Goldie, and S. Withington. Quasiparticle generation efficiency in superconducting thin films. *Superconductor Science and Technology*, 27(5):055012, March 2014. ISSN 0953-2048. doi: 10.1088/0953-2048/27/5/055012. URL https://dx.doi.org/10.1088/0953-2048/27/5/055012. Publisher: IOP Publishing.

[22] A. Rothwarf and B.N. Taylor. Measurement of Recombination Lifetimes in Superconductors. *Physical Review Letters*, 19(1):27–30, July 1967. ISSN 0031-9007. doi: 10.1103/PhysRevLett.19.27. URL https://link.aps.org/doi/10.1103/PhysRevLett.19.27.

[23] R. Haberman. *Applied Partial Differential Equations with Fourier Series and Boundary Value Problems*. Pearson, March 2018. ISBN 978-0-13-499543-4.

[24] J. Hu, J. Matin, P. Nicaise, F. Boussaha, C. Chaumont, M. Piat, V.D. Pham, and P. Bonifacio. Investigation of quasi-particle relaxation in strongly disordered superconductor resonators. *Superconductor Science and Technology*, 37(5):055014, April 2024. ISSN 0953-2048. doi: 10.1088/1361-6668/ad3f80. URL https://dx.doi.org/10.1088/1361-6668/ad3f80. Publisher: IOP Publishing.

[25] V. Narayanamurti, R.C. Dynes, P. Hu, H. Smith, and W.F. Brinkman. Quasiparticle and phonon propagation in bulk, superconducting lead. *Physical Review B*, 18(11):6041–6052, December 1978. doi: 10.1103/PhysRevB.18.6041. URL https://link.aps.org/doi/10.1103/PhysRevB.18.6041. Publisher: American Physical Society.

[26] Steven A. H. de Rooij, Remko Fermin, Kevin Kouwenhoven, Tonny Coppens, Vignesh Murugesan, David J. Thoen, Jan Aarts, Jochem J. A. Baselmans, and Pieter J. de Visser. Recombination of localized quasiparticles in disordered superconductors, October 2024. URL http://arxiv.org/abs/2410.18802. arXiv:2410.18802.

[27] G. Abadias, J.J. Colin, D. Tingaud, P. Djemia, L. Belliard, and C. Tromas. Elastic properties of α- and β-tantalum thin films. *Thin Solid Films*, 688:137403, October 2019. ISSN 00406090. doi: 10.1016/j.tsf.2019.06.053. URL https://linkinghub.elsevier.com/retrieve/pii/S0040609019304225.

Appendix A

Material Parameters

Below is an overview of the material parameters used for calculations in this thesis. The density of states and diffusivity at the Fermi level were measured by the research group[26], as were the critical temperatures. v_L, v_T and T_D are from [27]. The valence electrons and atomic mass were taken from the respective Wikipedia pages. The values for α^2_{av} and τ_0 for Aluminum are taken from [19].

Parameter	**Aluminum**	**β-Tantalum**
ν_e, N_0 ($\mathrm{eV}^{-1}\,\mu\mathrm{m}^{-3}$)	17.2×10^9	30.3×10^9
D ($\mu\mathrm{m}^2\,\mu\mathrm{s}^{-1}$)	15000	75
T_c (K)	1.19	0.87
T_D (K)	428.21	221
v_L ($\mu\mathrm{m}\,\mu\mathrm{s}^{-1}$)	6420	4350
v_T ($\mu\mathrm{m}\,\mu\mathrm{s}^{-1}$)	3040	1740
ρ_m ($\mathrm{g\,cm}^{-3}$)	2.7	17.1
Z (-)	3	2
m_a (u)	26.982	180.94788
α^2_{av} (eV)	1.93×10^{-3}	–
τ_0 (μs)	0.11	–

Table A.1: Material parameters for Aluminum and Tantalum

Appendix B

Theoretical Calculations

All my code is available at `https://github.com/10483/MTP`, if this repo is not available anymore, the code in this latex document can be downloaded from the folder `code` at `https://www.overleaf.com/read/gvtbjnkcmsvh#469188`. Below is one of the more important Python notebooks we made, `calc_theory.ipynb`, which implements the theory calculations outlined in figure 3.2, as well as several following calculations made for this thesis.

Preamble

```
import numpy as np
from scipy.integrate import dblquad
from scipy.integrate import quad
import matplotlib.pyplot as plt
plt.rcParams.update({
    "text.usetex": True,
    "font.family": "Computer Modern",
    "figure.dpi": 100,
})

class fixed_values: # constants in my choice of units
  def __init__(self):
    self.h=4.135667e-9 #eV us
    self.hbar=self.h/(2*np.pi) #eV us rad-1
    self.c=299792458 #um/us
    self.k_B=8.617343e-5 #eV/K
    self.eta_pb_max=0.59
    self.energy_au=27.211386 #eV
    self.time_au=2.418884e-11 #us
    self.length_au=5.291772e-5 #um
    self.N_A = 6.022e23
    self.m_e = 5.68563006e-12 #eV s2 m-2
consts = fixed_values()
```

Material Parameters

```
def calc_RS(metal,T):
    beta = (2/3 * metal.e_fermi)**2 * metal.nue/(2*metal.rho_m*metal.v_L**2)
    tau_epsilon = 1/(np.pi**4*beta/5 *
    ↪  metal.p_fermi*metal.l*T**4/(metal.p_fermi*metal.v_T)**3 *
    ↪  (1+3/2*(metal.v_L/metal.v_T)**5) * consts.k_B**4 / consts.hbar**2)
    tau_0 = 4.20*tau_epsilon # kaplan 18
```

```
    return tau_0

def calc_KSK(metal):
  qD = metal.E_D/(consts.hbar*metal.v_L)

  Phi_L = lambda x : 2/np.pi * ((x*np.arctan(x))/(x-np.arctan(x)) - 3/x)
  Phi_T = lambda x : 3/np.pi * (2*x**3 + 3*x - 3*(x**2+1) * np.arctan(x))/x**4
  gL =  metal.p_fermi**2 / (3*consts.m_e*np.sqrt(metal.rho_m)*metal.v_L)

  alpha2F = lambda omega : gL**2 * (metal.mstar*qD**2*omega**2)/(4*metal.p_fermi**2
  ↪  *metal.T_D**2) * (Phi_L(omega*metal.l/metal.v_L)+(metal.v_L/metal.v_T)**4 * 2 *
  ↪  Phi_T(omega*metal.l/metal.v_T)) * consts.hbar**3 * consts.k_B**-2

  alpha2_av = quad(alpha2F,0,metal.omega_D)[0] / 3
  return alpha2_av

class superconductor():
  def __init__(self,metaldict,disordered=False):
    self.nue = metaldict['nue']
    self.D = metaldict['D']
    self.Tc = metaldict['Tc']
    self.Delta = 1.768*consts.k_B*self.Tc #eV
    self.T_D = metaldict['T_D']
    self.omega_D = consts.k_B*self.T_D/consts.hbar #in MHz rad
    self.E_D = consts.k_B*self.T_D
    self.v_L = metaldict['v_L']
    self.v_T = metaldict['v_T']

    rho_m = metaldict['rho_m (g cm-3)']
    self.Z = metaldict['Z']
    m_a = metaldict['m_a (u)']

    self.N = consts.N_A*rho_m/m_a * 1e-12 # atoms um-3
    self.n = self.N*self.Z # electrons um-3
    self.rho_m = rho_m*6.242e15 *1e-12#eV us2 um-2 um-3

    self.rs = (3/(4*np.pi*self.n))**(1/3)
    self.rs_au = self.rs / consts.length_au

    self.p_fermi = np.sqrt((consts.hbar)**2*(3*np.pi**2*self.n)**(2/3))
    self.k_fermi = self.p_fermi/consts.hbar
    self.mstar = 2*consts.hbar**2*self.nue*np.pi**2/self.k_fermi # from suppl, fermi liquid
    ↪  model
    self.v_fermi = self.p_fermi/self.mstar
    self.e_fermi = (consts.hbar)**2/(2*self.mstar)*(3*np.pi**2*self.n)**(2/3)

    self.tau_elastic = self.nue*self.D*self.mstar/self.n # <= Einstein relation (sigma = e^2
    ↪  N(0) D) and drude effective mass, tau elastic time

    self.l=self.v_fermi*self.tau_elastic


    if disordered==False:
      self.tau_0 = 0.11 #us
      self.alpha2_av = metaldict['alpha2_av']
    else:
      self.tau_0 = calc_RS(self,self.Tc)
      self.alpha2_av = calc_KSK(self)
```

```
59
60  aluminium = {
61      'nue' : 17.2e9, #eV-1 um-3
62      'D' : 15000, #um2 us-1
63      'Tc' : 1.19, #K
64      'T_D' : 428.21, #K
65      'v_L' : 6420, #um/us
66      'v_T' : 3040, #um/us
67      'rho_m (g cm-3)' : 2.7, #g cm-3
68      'Z' : 3, #valence electrons per atom
69      'm_a (u)' : 26.982, #u
70      'alpha2_av' : 1.93e-3, #eV
71      'tau_0' : 0.11, #us
72  }
73
74  tantalum = {
75      'nue' : 30.3e9, #eV-1 um-3
76      'D' : 75, #um2 us-1
77      'Tc' : 0.87, #K
78      'T_D' : 221, #K
79      'v_L' : 4.35e3, #um/us
80      'v_T' : 1.74e3, #um/us
81      'rho_m (g cm-3)' : 17.1, #g cm-3
82      'Z' : 2, #valence electrons per atom
83      'm_a (u)' : 180.94788, #u
84    }
85
86  T=0.02 #K # all calculations requiring a non-zero temperature done at 20mK
87  Al = superconductor(aluminium,disordered=False)
88  bTa = superconductor(tantalum,disordered=True)
```

Functions

```
1   def calc_LP(metal,energy):
2       tau_ee = 1/((energy**2 / (consts.hbar*metal.e_fermi))*metal.rs_au**0.5 / 7.96)
3       return tau_ee
4
5   def calc_pippard(metal,energy):
6       omega = energy/consts.hbar
7
8       k_L=omega/metal.v_L
9       alpha_L = metal.n*metal.mstar/(metal.rho_m*metal.v_L*metal.tau_elastic) * (1/3 *
        ↪  (k_L**2*metal.l**2*np.arctan(k_L*metal.l)/(k_L*metal.l-np.arctan(k_L*metal.l))) - 1)
10      tau_ph_L = 1/(metal.v_L*alpha_L)
11
12      k_T=omega/metal.v_T
13      g = 3/(2*k_T**2*metal.l**2)*((k_T**2*metal.l**2+1)/(k_T*metal.l)*np.arctan(k_T*metal.l)-
        ↪  1)
14      alpha_T = metal.n*metal.mstar/(metal.rho_m*metal.v_T*metal.tau_elastic) * (1-g)/g
15      tau_ph_T = 1/(metal.v_T*alpha_T)
16      return tau_ph_L,tau_ph_T
17
18  @np.vectorize
19  def calc_kaplan(metal,energy,T):
20      tau_s = metal.tau_0/((metal.Delta/(consts.k_B*metal.Tc))**3*(1/3*((energy/metal.Delta)**
        ↪  2-1)**(3/2)+5/2*((energy/metal.Delta)**2-1)**(1/2)-metal.Delta/(2*energy)*(1+4*(ener
        ↪  gy/metal.Delta)**2)*np.log(energy/metal.Delta+((energy/metal.Delta)**2-1)**(1/2))))
21
```

```
    f = lambda e : 1/(np.exp(e/(consts.k_B*T))+1)

    RB_int = lambda omega : 1/(omega**2 - metal.Delta**2)**0.5 *
    ↪  (omega*(energy-omega)+metal.Delta**2)/((energy-omega)**2-metal.Delta**2)**0.5 *
    ↪  (1-f(omega)-f(energy-omega))
    Rphs_int = lambda omega : 1/(omega**2 - metal.Delta**2)**0.5 *
    ↪  (omega*(energy+omega)-metal.Delta**2)/((energy+omega)**2-metal.Delta**2)**0.5 *
    ↪  (f(omega)-f(energy+omega))

    RB = 4*np.pi*metal.nue*metal.alpha2_av/(consts.hbar*metal.n) *
    ↪  quad(RB_int,metal.Delta,energy-metal.Delta)[0]
    Rphs = 8*np.pi*metal.nue*metal.alpha2_av/(consts.hbar*metal.n) *
    ↪  quad(Rphs_int,metal.Delta,np.inf)[0]


    return tau_s, 1/RB#, 1/Rphs
```

Tantalum Rates Plot

```
E_range = np.geomspace(2.01*bTa.Delta,10*bTa.E_D,1000)
tau_ee = calc_LP(bTa,E_range)
tau_ph_L,tau_ph_T = calc_pippard(bTa,E_range)
tau_s,tau_B = calc_kaplan(bTa,E_range,T)

plt.figure()
#plt.loglog(E_range,1/tau_ee,label=r'£\frac{1}{\tau_{e-e}}£')
plt.loglog(E_range,1/tau_ph_L,label=r'$\tau_{ph-e,L}^{-1}$, Pippard')
plt.loglog(E_range,1/tau_ph_T,label=r'$\tau_{ph-e,T}^{-1}$, Pippard')
plt.loglog(E_range,1/tau_s,label=r'$\tau_{e-ph}^{-1}$, Reizer/Sergeyev + Kaplan')
plt.loglog(E_range,1/tau_B,label=r'$\tau_{ph-e,K}^{-1}$, Kaplan')
#plt.loglog(E_range,1/tau_phs,label=r'£\tau_{phs}^{-1}£, Kaplan')

plt.vlines([2*bTa.Delta,3*bTa.Delta],1,2e11,colors='purple',label=r'$2\Delta,3\Delta$')
plt.ylim(1,2e11)

plt.legend()
plt.title(r'$\beta-$Tantalum')
plt.xlabel('Energy (eV)')
plt.ylabel('Rate (MHz)')
plt.show()
```

Aluminum Rates Plot

```
E_range = np.geomspace(2.01*Al.Delta,10*Al.E_D,1000)
tau_ee = calc_LP(Al,E_range)
tau_ph_L,tau_ph_T = calc_pippard(Al,E_range)
tau_s,tau_B = calc_kaplan(Al,E_range,T)

plt.figure()
#plt.loglog(E_range,1/tau_ee,label=r'£\frac{1}{\tau_{e-e}}£')
plt.loglog(E_range,1/tau_ph_L,label=r'$\tau_{ph-e,L}^{-1}$, Pippard')
plt.loglog(E_range,1/tau_ph_T,label=r'$\tau_{ph-e,T}^{-1}$, Pippard')
plt.loglog(E_range,1/tau_s,label=r'$\tau_{e-ph}^{-1}$, Kaplan')
plt.loglog(E_range,1/tau_B,label=r'$\tau_{ph-e,K}^{-1}$, Kaplan')
#plt.loglog(E_range,1/tau_phs,label=r'£\tau_{phs}^{-1}£, Kaplan')

plt.vlines([2*Al.Delta,3*Al.Delta],1,2e11,colors='purple',label=r'$2\Delta,3\Delta$')
plt.ylim(1,2e11)

```

```
plt.legend()
plt.title('Aluminium')
plt.xlabel('Energy (eV)')
plt.ylabel('Rate (MHz)')
plt.show()
```

Stage 1 Checks

```
# Al
E1 = 67*Al.E_D
tau_ee = calc_LP(Al,E1)
print('tau_ee\t\t',tau_ee)
print('tau_elastic\t',Al.tau_elastic)
print(tau_ee**-1)
print(Al.tau_elastic**-1)

# bTa
lambda_au = (1/(bTa.E_D*bTa.tau_0))*(bTa.E_D/(consts.k_B*bTa.Tc))**3 *
↪  consts.energy_au*consts.time_au

E1 = 2.82*bTa.E_D*bTa.rs_au**(-1/4)*(lambda_au/3 * bTa.e_fermi/bTa.E_D)**(1/2)

tau_ee = calc_LP(bTa,E1)
print('tau_ee\t\t',tau_ee)
print('tau_elastic\t', bTa.tau_elastic) # = drude scattering time, based on diffusion
↪  measurement
print(tau_ee**-1)
print(bTa.tau_elastic**-1)
```

Stage 2 check

```
#Al, replacing some values with those directly from Kozorezov, to compare 1-to-1 with his
↪  results

E_D=36.9e-3
Delta=0.17e-3
E1 = 67*E_D

lambda0=(1/(E_D*Al.tau_0))*(E_D/(consts.k_B*Al.Tc))**3
tau1 = (3/(lambda0*E_D))*(E1/E_D)
print('tau1 = ',tau1*1e6,'ps')

tau_phD = calc_pippard(Al,E_D)
tau_phDK = calc_kaplan(Al,E_D,T)[1]
print(tau_phD[0]*1e6,'ps')
print(tau_phD[1]*1e6,'ps')
print(tau_phDK*1e6,'ps')


#bTa
lambda0=(1/(bTa.E_D*bTa.tau_0))*(bTa.E_D/(consts.k_B*bTa.Tc))**3
lambda_au = (1/(bTa.E_D*bTa.tau_0))*(bTa.E_D/(consts.k_B*bTa.Tc))**3 *
↪  consts.energy_au*consts.time_au
E1 = 2.82*bTa.E_D*bTa.rs_au**(-1/4)*(lambda_au/3 * bTa.e_fermi/bTa.E_D)**(1/2)
tau1 = (3/(lambda0*bTa.E_D))*(E1/bTa.E_D)
print('tau1 = ',tau1*1e6,'ps')

tau_phD = calc_pippard(bTa,bTa.E_D)
tau_phDK = calc_kaplan(bTa,bTa.E_D,T)[1]
```

```
print(tau_phD[0]*1e6,'ps')
print(tau_phD[1]*1e6,'ps')
print(tau_phDK*1e6,'ps')
```

Stage 3 check - phonon diffusion

```
length = 10
width = 10
height = 0.04

def distfunc(phi,theta):
    return np.min(np.stack([height/2/np.sqrt(1-np.sin(phi)**2),length/2/np.sqrt(1-np.sin(the
    ↪  ta)**2),width/2/np.sqrt(1-np.sin(np.pi/2-theta)**2)],axis=-1),axis=-1)

mfp = dblquad(distfunc,0,np.pi/2,0,np.pi/2)[0]*4/np.pi**2
D_ph = mfp*bTa.v_L/3
print(mfp)
print(bTa.v_L)
print(D_ph, '= D (um2/us)')
```

Phonon lifetimes at Ω_1 - and σ_{IC}

```
_,tau_B = calc_kaplan(Al,1.5e-3,T)
print('tau_B = ',tau_B*1e3,'ns')

_,tau_B_Omega1 = calc_kaplan(bTa,3e-4,T)
print('tau_B_Omega1 = ',tau_B_Omega1*1e3,'ns')
tau_ph_L_Omega1,_ = calc_pippard(bTa,4e-4)
print('tau_ph_L_Omega1 = ',tau_ph_L_Omega1*1e3,'ns')
_,tau_ph_T_Omega1 = calc_pippard(bTa,9e-4)
print('tau_ph_T_Omega1 = ',tau_ph_T_Omega1*1e3,'ns')

```

Qp lifetimes

```
print('Al:',Al.tau_0*Al.nue*(consts.k_B*Al.Tc)**3/(2*Al.Delta**2),'/ n_qp') # P de Visser
↪  2014 eq 2.29
print('bTa:',bTa.tau_0*bTa.nue*(consts.k_B*bTa.Tc)**3/(2*bTa.Delta**2),'/ n_qp')
```

Duration calculations

```
#Al
c = Al.v_L
nuph = 3*Al.E_D**2/(2*np.pi**2*c**3)/consts.hbar**3
lambda1 = (lambda0*Al.nue/(2*nuph)) #eV-1 us-1
Omega1 = Al.E_D*np.sqrt(3*lambda1/lambda0) #eV
print(Omega1)

_,tau_B_Omega1 = calc_kaplan(Al,Omega1,T)
t1=2*tau_B_Omega1
print('t1 = ',t1*1e6,'ps')

t2,_ = calc_kaplan(Al,4.5*Al.Delta,T)
print('t2 = ',t2*1e3,'ns')


#bTa

_,tau_B_Omega1 = calc_kaplan(bTa,3e-4,T)
```

```
t1=2*tau_B_Omega1
print('t1_K = ',t1*1e3,'ns')

tau_ph_L_Omega1,_ = calc_pippard(bTa,4e-4)
t1=2*tau_ph_L_Omega1
print('t1_L = ',t1*1e3,'ns')

_,tau_ph_T_Omega1 = calc_pippard(bTa,9e-4)
t1=2*tau_ph_T_Omega1
print('t1_T = ',t1*1e3,'ns')

t2,_ = calc_kaplan(bTa,4.5*bTa.Delta,T)
print('t2 = ',t2*1e3,'ns')

print('MD = ',np.sqrt(4*D_ph*t1)+np.sqrt(4*D_ph*t2),'um')
```

Responsivity nonlinearity

```
from scipy.special import lambertw

def resp_nonlin_part_MB(metal,F0,Teff):
    omega_0 = 2*np.pi*F0
    xi=consts.hbar*omega_0/(2*consts.k_B*Teff)
    nonlin = np.sqrt(2*metal.Delta/(np.pi*consts.k_B*Teff))*np.exp(-xi)*np.i0(xi)
    return nonlin


def nqp_to_T(nqp,N0,Delta,height):
    a = 2*N0*height*np.sqrt(2*np.pi*consts.k_B*Delta)
    b = Delta/consts.k_B
    return np.real(2*b/lambertw(2*a**2*b/(nqp**2)))

def T_to_nqp(T,N0,Delta,height):
    return 2*N0*height*np.sqrt(2*np.pi*consts.k_B*T*Delta)*np.exp(-Delta/(consts.k_B*T))

Eph = consts.h*consts.c/0.402
dNqpinit = 0.6*Eph/bTa.Delta

print(dNqpinit)

nqp_0=T_to_nqp(0.1,bTa.nue,bTa.Delta,0.04)

sigmalist=np.linspace(0.1,15,10000)
Tefflist=[]
for sigma in sigmalist:
    dnqpmax = dNqpinit/(2*np.pi*sigma**2)
    #print(dnqpmax)
    Tcfrac = nqp_to_T(dnqpmax+nqp_0,bTa.nue,bTa.Delta,0.04)/bTa.Tc
    #print(Tcfrac)
    Tefflist.append(Tcfrac)
Tefflist=np.array(Tefflist)

plt.figure()
plt.plot(sigmalist,Tefflist,color='b')
plt.vlines(0.7,0,0.4,colors=['red'])
plt.semilogx()
plt.ylim(0,1)
plt.xlim(0,10)
```

```
41  plt.xlabel(r'$\sigma_{IC}$ ($\mu m$)')
42  plt.ylabel(r'$T_{eff}$ ($T_c$)')
43
44  #print(resp_nonlin_part_MB(bTa,8500,Tcfrac*bTa.Tc))
```

Appendix C

Recombination Simulation

All my code is available at `https://github.com/10483/MTP`, if this repo is not available anymore, the code in this latex document can be downloaded from the folder `code` at `https://www.overleaf.com/read/gvtbjnkcmsvh#469188`. Below we pasted the file `Finite_volume_method_1D.py`. This Python code contains the 1D recombination model that includes diffusion, which is the main product of this thesis.

Structure and functionality

All dimensions, unless the variable name states otherwise, are in units μm, μs, K, eV or units we derive from those. Therefore, we defined the class `fixed_values` with which we make only one object called `consts`, which contains useful constants in our choice of units.

Then we define a class `KID_data`. When we create an instance of this class, we must supply the constructor with a path to a directory containing measurements of a certain chip, as well as a target measurement (KID number, readout power, temperature, and laser wavelength). It then copies from that measurement the average pulse data and parameters we need such as the dimensions of the absorber, the resonance frequency, the quality factor, the responsivity of the device and the critical temperature. The class also has a method to fit an exponential over a subdomain of choice in the average pulse data. We should fit this over the tail of the decay, where is has become predominantly exponential, to obtain τ^*_{qp}, which corresponds with $1/2R'Q_0$ and is one of the parameters we need in the simulation.

The next class to use is `KID_params`. Its constructor gathers parameters from a provided `KID_data` object and/or manually selected parameters. When both are given, priority is given to the manually inserted parameter. It gathers the result in one object which we use as input to the simulation.

The simulation itself is written as a class `KID_sim`, which is used to create an object with the simulation results. The simulation results are calculated on the basis of the parameters provided through a `KID_params` object and a number of simulation parameters and options, that define step size, usage of optimizations, simulation time, etc.

```
import numpy as np
from scipy.optimize import fsolve
from scipy.special import lambertw
from scipy.integrate import quad
import matplotlib.pyplot as plt
import os
import re

# everything in um, us, K and eV  and combinations of them unless stated differently
```

```
class fixed_values:
  def __init__(self):
    self.h=4.135667e-9 #eV us
    self.hbar=self.h/(2*np.pi) #eV us rad-1
    self.c=299792458 #um/us
    self.k_B=8.617343e-5 #eV/K
    self.eta_pb_max=0.59
    self.energy_au=27.211386 #eV
    self.time_au=2.418884e-11 #us
    self.length_au=5.291772e-5 #um
    self.N0_bTa = 30.3e9 #eV-1 um-3
    self.D0_bTa = 75 #um2 us-1
    self.N0_Al = 17.2e9 #eV-1 um-3
    self.D0_Al = 15000 #um2 us-1
consts = fixed_values()

#gather data from specific KID necessary for simulation and data comparison.
class KID_data:
  def __init__(self,chippath,lambda_ph_in_nm,filename,length,samplefreq_in_MHz=1,FFT_power_path=False,material='bTa'):
    #copy stuff
    self.filename = filename
    self.samplefreq_in_MHz = samplefreq_in_MHz
    self.dt = 1/self.samplefreq_in_MHz
    self.chippath = chippath
    self.FFT_power_path = FFT_power_path
    self.lambda_ph_in_nm = lambda_ph_in_nm
    self.lambda_ph = lambda_ph_in_nm/1000
    nums = list(map(int, re.findall(r'\d+', filename)))
    self.KIDno = nums[0]
    self.readout_power=nums[1]
    self.temp_in_mK = nums[2]
    self.temp = self.temp_in_mK/1000
    self.length = length
    if material == 'bTa':
      self.N0 = consts.N0_bTa
      self.D0 = consts.D0_bTa
    elif material == 'Al':
      self.N0 = consts.N0_Al
      self.D0 = consts.D0_Al
    else:
      raise ValueError()
    #get pulse data
    self.getpulsedata()
    #get resonator data
    if self.FFT_power_path==False:
      self.getresdata_S21()
    else:
      print('Warning! You provided FFT_power_path, likely because S21 measurements are lacking. Make sure to manually provide all parameters (except tau_ringing) to KID_params().', self.Tdeppath)
      self.getresdata_FFT()

    '''
    #deconvolve
    self.ringing = np.exp(-self.t_full/self.tau_ringing)
    self.ringing /= np.sum(self.ringing)
```

```
64        self.phase, _ =
          ↪  deconvolve(np.pad(self.phase,(0,len(self.t_full)-1),constant_values=0),self.ringing)
65        self.amp, _ =
          ↪  deconvolve(np.pad(self.amp,(0,len(self.t_full)-1),constant_values=0),self.ringing)
66        '''
67
68      def getpulsedata(self):
69        self.datapath = self.chippath+str(self.lambda_ph_in_nm)+'nm/'+self.filename
70        self.data = np.genfromtxt(self.datapath,skip_header=1,delimiter=',')
71        self.amp = self.data[:,0]
72        self.ampstd = self.data[:,1]
73        self.phase = self.data[:,2]
74        self.phasestd = self.data[:,3]
75        self.t_full = np.arange(len(self.phase))*self.dt
76
77      def getresdata_S21(self):
78        self.Tdeppath =
          ↪  self.chippath+'S21/2D/KID'+str(self.KIDno)+'_'+str(self.readout_power)+'dBm_Tdep.csv'
79        if os.path.exists(self.Tdeppath):
80          self.Tdepdata = np.genfromtxt(self.Tdeppath,skip_header=1,delimiter=',')
81        else:
82          filenames = os.listdir(self.chippath+'S21/2D/')
83          pattern = re.compile(r'KID'+str(self.KIDno)+r'.*Tdep.*')
84          candidates = [filename for filename in filenames if pattern.match(filename)]
85          powers = np.array([int(re.findall(r'\d+',name)[1]) for name in candidates])
86          closest = powers[np.argmin(np.abs(powers-self.readout_power))]
87          self.Tdeppath =
            ↪  self.chippath+'S21/2D/KID'+str(self.KIDno)+'_'+str(closest)+'dBm_Tdep.csv'
88          self.Tdepdata = np.genfromtxt(self.Tdeppath,skip_header=1,delimiter=',')
89          print('Readout power not found, instead taking resonator data at closest readout power
            ↪  available.\npath: ', self.Tdeppath)
90        TdepT=self.Tdepdata[:,1]
91        argT = np.abs(TdepT-self.temp).argmin()
92        self.Quality=self.Tdepdata[argT,2]
93        self.F0=self.Tdepdata[argT,5]*1e-6 #convert to /us from Hz
94        self.T_c=self.Tdepdata[argT,21]
95        self.volume = self.Tdepdata[argT,14]
96        self.height = self.Tdepdata[argT,25]
97        self.width = self.volume/self.height/self.length
98        self.dthetadN=self.Tdepdata[argT,10]
99        self.dAdN=self.Tdepdata[argT,18]
100       self.tau_ringing=self.Quality/(np.pi*self.F0)
101       self.Delta = 1.768*consts.k_B*self.T_c
102
103     def getresdata_FFT(self):
104       path_to_data =
          ↪  self.chippath+self.FFT_power_path+'KID'+str(self.KIDno)+'_0dBm__all_td_averaged.dat'
105       with open(path_to_data,'r') as file:
106           lines = file.readlines()
107       line3=lines[2].strip()
108       line4=lines[3].strip()
109       self.F0 = float(re.findall(r"\d*\.?\d+",line3)[0]) * 1000 #from Ghz to us-1
110       self.Quality = float(re.findall(r"\d*\.?\d+",line4)[0])
111       self.tau_ringing = self.Quality/(np.pi*self.F0)
112
113     def fit_tail(self,start=0,end=-1,showplots=True):
114       self.phasefit = self.phase[start:end]
115       self.t_fit = self.t_full[start:end]
```

```
116       phasefit = self.phasefit[self.phasefit>0]
117       t_fit = self.t_fit[self.phasefit>0]
118       self.fitpars,self.fitcov=np.polyfit(t_fit,np.log(phasefit),1,cov=True)
119       a=np.exp(self.fitpars[1])
120       b=self.fitpars[0]
121       b_std=np.sqrt(np.diag(self.fitcov)[0])
122       self.tauqpstar=-1/b
123       self.L=1/self.tauqpstar
124       self.tauqpstarstd=b_std/b**2
125       if showplots:
126         plt.figure()
127         plt.semilogy(self.phasefit)
128         plt.semilogy(a*np.exp(-self.t_fit/self.tauqpstar))
129         plt.show()
130
131   class KID_params():
132     def __init__(self,eta_pb,sigma_IC,Teff=False,QO=False,KID=False,lambda_ph=False,tau_ringin
      ↪  g=False,dthetadN=False,NO=False,DO=False,L=False,Delta=False,length=False,height=False
      ↪  ,width=False,trickle_time=False):
133       # Copy data from KID. If optional param is given, overwrite KID value with individually
       ↪  specified value.
134       if KID == False:
135         print('Warning! No KID data given. Make sure to check whether all other optional
          ↪  parameters are provided, otherwise garbage may be produced in the simulation.')
136
137       self.eta_pb = eta_pb
138       self.sigma_IC = sigma_IC
139       self.lambda_ph = lambda_ph if (KID==False) or (lambda_ph!=False) else KID.lambda_ph
140       self.tau_ringing = tau_ringing if (KID==False) or (tau_ringing!=False) else
       ↪  KID.tau_ringing
141       self.dthetadN = dthetadN if (KID==False) or (dthetadN!=False) else KID.dthetadN
142       self.NO = NO if (KID==False) or (NO!=False) else KID.NO
143       self.DO = DO if (KID==False) or (DO!=False) else KID.DO
144       self.L = L if (KID==False) or (L!=False) else KID.L
145       self.Delta = Delta if (KID==False) or (Delta!=False) else KID.Delta
146       self.length = length if (KID==False) or (length!=False) else KID.length
147       self.height = height if (KID==False) or (height!=False) else KID.height
148       self.width = width if (KID==False) or (width!=False) else KID.width
149       self.trickle_time = trickle_time
150
151       if (Teff==False) and (QO==False):
152         raise ValueError('Either Teff or QO should be specified.')
153       self.QO = QO if (QO!=False) else self.T_to_nqp(Teff)
154
155     def T_to_nqp(self,Teff):
156       return 2*self.NO*np.sqrt(2*np.pi*consts.k_B*Teff*self.Delta)*np.exp(-self.Delta/(consts.
       ↪  k_B*Teff))*self.height*self.width
157
158     def print(self):
159       print('eta_pb: \t',self.eta_pb)
160       print('sigma_IC: \t',self.sigma_IC)
161       print('QO: \t\t', self.QO)
162       print('lambda_ph: \t',self.lambda_ph)
163       print('tau_ringing: \t',self.tau_ringing)
164       print('dthetadN: \t',self.dthetadN)
165       print('NO: \t\t',self.NO)
166       print('DO: \t\t',self.DO)
167       print('L: \t\t',self.L)
```

```
168       print('Delta: \t\t',self.Delta)
169       print('lxhxw: \t\t',self.length,'x',self.height,'x',self.width)
170       print('trickle_time: \t', self.trickle_time)
171
172
173   class KID_sim():
174     def __init__(self,params,dt_init,dx_or_fraction,dt_max=10,simtime_approx=100,adaptivedx=Tr
      ↪  ue,adaptivedt=True,usesymmetry=True,D_const=False,ringingdtinterp=0.001,approx2D=False
      ↪  ): # 2D option is sketchy, don't trust without more checking. If loading after a sim is
      ↪  too slow or requires too much space, increase ringingdtinterp.
175       #copy physical parameters from params object
176       self.params=params
177
178       # calculate other parameters
179       self.K = self.params.L/(2*self.params.Q0)
180       self.E_ph = consts.h*consts.c/self.params.lambda_ph
181       self.Nqp_init = self.params.eta_pb*self.E_ph/self.params.Delta
182
183       # Optimization setting
184       self.adaptivedx = adaptivedx #Increase the courseness of the grid according to the
      ↪  expected width of the distribution. => also sets dx = sigma_IC*dx_or_fraction
185       self.usesymmetry = usesymmetry #Simulates only half the domain, efficient for a
      ↪  symmetrical situation
186       self.adaptivedt = adaptivedt #Keeps the product of dt and DN constant at each step, such
      ↪  that for smaller expected changes in N (in the tail of the decay), larger timesteps
      ↪  are used.
187
188       # Initialize time axis and dt
189       dt = dt_init
190       self.t_axis=[0]
191       self.dtlist=[dt_init]
192
193       # set geometry
194       if self.adaptivedx: # if using adaptive dx option
195         dx = self.params.sigma_IC*dx_or_fraction # set dx as fraction
196       else: # otherwise set constant dx value
197         dx = dx_or_fraction
198       if self.usesymmetry: # calculate list of possible dx values that divide the domain
      ↪  cleanly, from small to large
199         maxdiv = int(np.ceil(self.params.length/2/dx))
200         valid_dx_list = self.params.length/2/np.arange(1,maxdiv+0.5)[::-1]
201       else:
202         maxdiv = int(np.ceil(self.params.length/dx))
203         valid_dx_list = self.params.length/np.arange(1,maxdiv+0.5)[::-1]
204       dx = valid_dx_list[0] # update dx to valid value close to the one set before
205       self.dxlist = [dx] # store dx in new output list which will contain dx at each timestep
206
207       self.set_geometry(dx,self.params.length) # calculate self.x_centers and self.x_borders
208       self.x_centers_list = [self.x_centers] # store x_centers in new output list
209
210       # initialize state variables
211       if self.params.trickle_time: # if using forcing term instead of simple IC
212         self.Qintime = [np.zeros_like(self.x_centers)] # set IC to zero
213       else:
214         self.Qintime = [np.exp(-0.5*(self.x_centers/self.params.sigma_IC)**2)*self.Nqp_init/(s
          ↪  elf.params.sigma_IC*np.sqrt(2*np.pi))] # set IC to
          ↪  Nqp_init
```

```
215         self.Qintime[0] = self.Qintime[0]*self.Nqp_init/self.integrate(self.Qintime[0],dx) #
            ↪  correct total Nqp if boundary clips off tails due to large sigma
216
217       self.Nqpintime = [self.integrate(self.Qintime[0],dx)] # calculate integral of density,
          ↪  store in new list
218
219       # calc thermal density of quasiparticles
220       Teff_thermal = self.nqp_to_T(self.params.Q0,self.params.N0,self.params.Delta,self.params
          ↪  .height,self.params.width) #calculate effective temperature at each
          ↪  volume
221       Dfinal = self.params.D0*np.sqrt(2*consts.k_B*Teff_thermal/(np.pi*self.params.Delta))
          ↪  #calculate D array for steady state
222
223       # run simulation
224       i=0 # keeps track of simulation step
225       self.t_elapsed=0 # keeps track of elapsed time (us)
226       t_elapsed_D=0 # keeps track of elapsed time but specifically for adapting dx with time
227       sqrtMSD2D=self.params.sigma_IC
228       if self.params.trickle_time: # pause adaptive dx as long as the forcing term is still
          ↪  large
229         dxAdaptPause=True
230       else:
231         dxAdaptPause=False
232
233       while True: # kind of a do-while loop
234         if (self.t_elapsed>3*self.params.trickle_time) and dxAdaptPause: # the integral from 0
            ↪  to 3*tau already contains >95% of the surface under the exponential.
235           dxAdaptPause=False # resume the adaptive dx optimization
236
237         # handle adaptive dx
238         if self.adaptivedx and (i!=0) and (dx!=valid_dx_list[-1]) and (dxAdaptPause==False):
239           sqrtMSD = np.sqrt(2*Dfinal*t_elapsed_D)+self.params.sigma_IC #mean squared distance
              ↪  expected from diffusion only (after forcing is negligible)
240           dx = valid_dx_list[valid_dx_list <= sqrtMSD*dx_or_fraction][-1] #update to the
              ↪  largest allowed dx value below the requested fraction of mean distance
241           self.set_geometry(dx,self.params.length) # calculate new geometry of simulation
242           Qprev = np.interp(self.x_centers,x_centersprev,self.Qintime[i]) # update the qp
              ↪  distribution to new geometry
243           Qprev *= self.integrate(self.Qintime[i],self.dxlist[i])/self.integrate(Qprev,dx) #
              ↪  correct for any potential loss of total Nqp by normalizing
244           t_elapsed_D+=dt # update elapsed diffusion time
245         else:
246           Qprev = self.Qintime[i]
247
248         # update diffusion
249         if D_const:
250           D=self.params.D0
251         else: # calculate local effective temperature
252           Teff_x = self.nqp_to_T(Qprev+self.params.Q0,self.params.N0,self.params.Delta,self.pa
              ↪  rams.height,self.params.width)
253           D = self.calc_D(self.params.D0,Teff_x,self.params.Delta) # calculate new location
              ↪  dependent diffusion
254         # 2D approximation
255         sqrtMSD2D += 4*self.params.D0*dt
256         if approx2D and (sqrtMSD2D<params.width/2):
257           D=8*D**2*(self.t_elapsed+dt) # use higher diffusion rate to simulate 2D diffusion if
              ↪  required
258
```

```
259         # do simulation step
260         self.dxlist.append(dx)
261         self.dtlist.append(dt)
262         self.Qintime.append(self.CN_step(dt,dx,D,self.params.L,self.K,Qprev))
263         self.Nqpintime.append(self.integrate(self.Qintime[i+1],dx))
264         self.x_centers_list.append(self.x_centers)
265 
266         x_centersprev = self.x_centers
267         self.t_elapsed+=dt
268         self.t_axis.append(self.t_elapsed)
269         print(f'\rIteration: {i}\tSimtime (us): {self.t_elapsed}', end='') # print and update
            ↪  progress counter
270 
271         if self.t_elapsed>simtime_approx: # check whether simulation has reached required time
272           break
273 
274         # handle adaptive dt
275         if self.adaptivedt and (dt<=dt_max):
276           dN = np.abs(self.Nqpintime[i]-self.Nqpintime[i+1]) # calculate difference in Nqp from
              ↪  previous step.
277           if i==0:
278             dNdt = dN*dt # set at beginning of simulation
279           if dN != 0:
280             dt = (dNdt/dN+dt)/2 # update dt, taking the mean stabilizes oscillations due to
                ↪  trickle and adaptive dt both depending on dt value.
281         i+=1
282       print()
283 
284       self.t_axis = np.array(self.t_axis) # save time array
285       self.Nqpintime=np.array(self.Nqpintime) # and data array
286 
287       if ringingdtinterp: # if desired: convolve with ringing with interpolation of dt =
          ↪  ringingdtinterp
288         self.t_axis_interp = np.arange(0,self.t_axis[-1],ringingdtinterp)
289         Nqp_interp = np.interp(self.t_axis_interp,self.t_axis,self.Nqpintime)
290         self.phaseintime = self.ringing(self.t_axis_interp,Nqp_interp,self.params.tau_ringing)
            ↪  *self.params.dthetadN
291         self.t_start = -self.t_axis_interp[np.argmax(self.phaseintime)]
292         self.t_axis_interp += self.t_start # align maximum to t=0
293       else: # otherwise simply multiply Nqp with responsivity
294         self.t_axis_interp = self.t_axis
295         self.phaseintime = self.Nqpintime*self.params.dthetadN
296         self.t_start = -self.t_axis_interp[np.argmax(self.phaseintime)]
297         self.t_axis_interp += self.t_start # align maximum to t=0
298 
299     def set_geometry(self,dx,length):
300       if self.usesymmetry: # set geometry for half the MKID
301         self.x_borders=np.arange(0,length/2+dx/2,dx)
302         self.x_centers=np.arange(dx/2,length/2,dx)
303       else: # set geometry for a full MKID
304         self.x_borders=np.arange(-length/2,length/2+dx/2,dx)
305         self.x_centers=np.arange(-length/2+dx/2,length/2,dx)
306       return
307 
308     def nqp_to_T(self,nqp,N0,Delta,height,width): # inverts n_qp(T)
309       a = 2*N0*height*width*np.sqrt(2*np.pi*consts.k_B*Delta)
310       b = Delta/consts.k_B
311       return np.real(2*b/lambertw(2*a**2*b/(nqp**2)))
```

```
312
313   def calc_D(self,D0,Teff_x,Delta): # calculates energy dependent D at elements, interpolates
      ↪  to borders
314     return np.interp(self.x_borders,self.x_centers,D0*np.sqrt(2*consts.k_B*Teff_x/(np.pi*Del
        ↪  ta)))
315
316   def diffuse(self,dx,D,Q_prev): # apply diffusion
317     Q_temp = np.pad(Q_prev,(1,1),'edge') #Assumes von Neumann BCs, for Dirichlet use
      ↪  np.pad(Q_prev,(1,1),'constant', constant_values=(0, 0)), disable 'usesymmetry' for
      ↪  this
318     gradient = D*np.diff(Q_temp)/dx
319     return (-gradient[:-1]+gradient[1:])/dx
320
321   def source(self,dt,dx): # apply quadratically decaying source term
322     S_next = np.exp(-0.5*(self.x_centers/self.params.sigma_IC)**2)/(self.params.sigma_IC*np.
      ↪  sqrt(2*np.pi))
323     S_prev = (self.Nqp_init/self.params.trickle_time)*np.exp(-(self.t_elapsed)/self.params.t
      ↪  rickle_time)*np.exp(-0.5*(self.x_centers/self.params.sigma_IC)**2)/(self.params.sigm
      ↪  a_IC*np.sqrt(2*np.pi))
324     integ_next = self.integrate(S_next,dx)
325     integ_prev = self.integrate(S_prev,dx)
326     if (integ_next > 1e-6) and (integ_prev > 1e-6):
327       S_next = S_next*(self.Nqp_init/self.params.trickle_time)*np.exp(-(self.t_elapsed+dt)/s
        ↪  elf.params.trickle_time)/integ_next
328       S_prev = S_prev*(self.Nqp_init/self.params.trickle_time)*np.exp(-(self.t_elapsed)/self
        ↪  .params.trickle_time)/integ_prev
329     return S_next,S_prev
330
331   def CN_eqs_source(self,dt,dx,D,L,K,Q_prev,Q_next): # the Crank-Nicolson update equations in
      ↪  case of a source term
332     S_next,S_prev = self.source(dt,dx)
333     return Q_prev - Q_next + 0.5*dt*(self.diffuse(dx,D,Q_next) - K*Q_next**2 - L*Q_next +
      ↪  S_next +
334                                      self.diffuse(dx,D,Q_prev) - K*Q_prev**2 - L*Q_prev +
                                         ↪  S_prev)
335
336   def CN_eqs(self,dt,dx,D,L,K,Q_prev,Q_next): # the Crank-Nicolson update equations without
      ↪  source term
337     return Q_prev - Q_next + 0.5*dt*(self.diffuse(dx,D,Q_next) - K*Q_next**2 - L*Q_next +
338                                      self.diffuse(dx,D,Q_prev) - K*Q_prev**2 - L*Q_prev)
339
340   def CN_step(self,dt,dx,D,L,K,Q_prev): # fsolve the CN equations, with the previous step/2
      ↪  as initial guess
341     if self.params.trickle_time: # if we have a source term:
342       return fsolve(lambda Q_next : self.CN_eqs_source(dt,dx,D,L,K,Q_prev,Q_next), Q_prev/2)
343     else:
344       return fsolve(lambda Q_next : self.CN_eqs(dt,dx,D,L,K,Q_prev,Q_next), Q_prev/2)
345
346   def integrate(self,Q,dx): # integrate nqp to find Nqp
347     if self.usesymmetry:
348       Nqp = np.sum(Q,axis=-1)*dx*2
349     else:
350       Nqp = np.sum(Q,axis=-1)*dx
351     return Nqp
352
353   def ringing(self,t_axis,phaseintime,tau_ringing): # convolve signal with ringing time to
      ↪  obtain phase response.
354     lenT = len(t_axis)
```

```
355         padded = np.pad(phaseintime,(lenT,lenT),constant_values=(0,0))
356         convring = np.exp(-t_axis/tau_ringing)
357         convring /= np.sum(convring)
358         return np.convolve(padded,convring,'valid')[:lenT]
```

Appendix D

Fitting Procedure

All my code is available at `https://github.com/10483/MTP`, if this repo is not available anymore, the code in this latex document can be downloaded from the folder `code` at `https://www.overleaf.com/read/gvtbjnkcmsvh#469188`. Below we pasted a typical application of the recombination model. With some explanation. It is a python notebook containing the fitting routine we frequently used in the course of this thesis, which is able to fit a single set of model parameters $\eta_{PB}, \sigma_{IC}, \tau_{IC}, Q_0$ to multiple photon energy pulses of a single MKID measurement simultaneously.

Preamble

We import the recombination model and `joblib` for parallelization.

```
import numpy as np
import matplotlib.pyplot as plt
#%matplotlib widget
from joblib import Parallel, delayed
from scipy.optimize import curve_fit
plt.rcParams.update({
    "text.usetex": True,
    "font.family": "Computer Modern",
    "figure.dpi": 300
})

from Finite_volume_method_1D import *
```

Load MKID data and set simulation parameters

Here we select the Chip, MKID and measurement. Then we set some simulation parameters and the wavelengths for which we have measurements. Then the MKID data is loaded and the tail of the data is fitted to obtain τ_{qp}^*. An average of the rate $L = 1/\tau_{qp}^*$ is calculated so all simulations use the same value.

```
chippath = '../../DataKian/bTa_LT278W2chip5/'
filename = 'KID5_99dBm__TmK20_avgpulse_ampphase.csv'
length = 90
simtime = 100
dtinit=0.001
dxfrac=1/5

lambdasinnm = [402,673,986,1545]

KIDdict = {}
```

```
for lambdanm in lambdasinnm:
    KIDdict[lambdanm] = KID_data(chippath,lambdanm,filename,length)
    KIDdict[lambdanm].fit_tail(showplots=True) # just to plot full pulses
for lambdanm in lambdasinnm:
    KIDdict[lambdanm] = KID_data(chippath,lambdanm,filename,length)
    KIDdict[lambdanm].fit_tail(start=250,end=400,showplots=True) # actually calculate tauqp*

Ls=[]
for lambdanm in lambdasinnm:
    print(KIDdict[lambdanm].tauqpstar,KIDdict[lambdanm].tauqpstarstd)
    print(KIDdict[lambdanm].L)
    Ls.append(KIDdict[lambdanm].L)
L=np.mean(Ls) # use the same linear decay rate for all wavelengths
```

Fitting and plotting functions

`prep_data` takes the data for the desired simulation time, and calculates a common time axis `xfit`, and concatenates the data for each wavelength into one array `yfit`. `func_to_parallelize` runs the simulation and interpolates the result at the data points. `fitfunc` runs the previous function for each wavelength in parallel, and concatenates the results so that it can be compared with `yfit` by `curve_fit`

```
def prep_data(offset):
    ylist = []
    for lambdanm in lambdasinnm:
        xfit = np.arange(offset,offset+simtime-KIDdict[lambdanm].dt/2,KIDdict[lambdanm].dt)
        indmin = np.argmax(KIDdict[lambdanm].phase)+offset
        indmax = indmin + simtime
        ylist.append(KIDdict[lambdanm].phase[indmin:indmax])
    yfit = np.array(ylist).ravel()
    return xfit, yfit

def func_to_parallelize(t,params):
     SIM = KID_sim(params,dtinit,dxfrac,simtime_approx=simtime,D_const=False,approx2D=False,
     ↪  ringingdtinterp=0.005)
     return np.interp(t,SIM.t_axis_interp,SIM.phaseintime)

def fitfunc(t,eta,sigma,tau,Q0):
    print('eta (-):\t',eta)
    print('sigma (um):\t',sigma)
    print('tau (us):\t',tau)
    print('Q0 (um-1):\t',Q0)
    datalist = Parallel(n_jobs=len(lambdasinnm))(delayed(func_to_parallelize)(t,KID_params(e
    ↪  ta_pb=eta,sigma_IC=sigma,trickle_time=tau,Q0=Q0,KID=KIDdict[lambdanm],L=L)) for
    ↪  lambdanm in lambdasinnm)
    return np.concatenate(datalist)

def par_func_plot(params):
     SIM = KID_sim(params,dtinit,dxfrac,simtime_approx=simtime,D_const=False,approx2D=False,
     ↪  ringingdtinterp=0.005)
     return SIM

def plotsim(eta,sigma,tau,Q0):
    datalist = Parallel(n_jobs=4)(delayed(par_func_plot)(KID_params(eta_pb=eta,sigma_IC=sigm
    ↪  a,trickle_time=tau,Q0=Q0,KID=KIDdict[lambdanm],L=L)) for lambdanm in
    ↪  lambdasinnm)
    plotdata = {}
    for obj,lambdanm in zip(datalist,lambdasinnm):
```

```
            plotdata[lambdanm]=obj
    return plotdata
```

Curve fit and print parameters

Here we prepare the data and run `curve_fit` and print the results.

```
xfit,yfit=prep_data(-5) #e.g. -5 means take 5 data points before the max into account for the data
popt, pcov = curve_fit(fitfunc,xfit,yfit,p0=[0.6,10,0.2,0.5],bounds=[[0.01,0.2,0.01,0.1],[1, length/6,10,np.inf]]) #with tau
print('\nDone!')

# Display parameters, standard deviation, and correlation coefficient
parameters = ['eta','sigma','tau','Q0']
print("Parameters:")
for name, param, std in zip(parameters,popt,np.sqrt(np.diag(pcov))):
    print(f"{name}:\t{param:.10f}\t+/-\t{std:.10f}")

print("\nCorrelation Coefficients:")
for i in range(len(popt)):
    for j in range(i+1, len(popt)):
        print(f"{parameters[i]} vs {parameters[j]}: {pcov[i,j]/np.prod(np.sqrt(pcov[i,i]*pcov[j,j])):.10f}")
```

Run optimized model and plot along with data

In this part we compare the data with the fitted model.

```
simdata = plotsim(popt[0],popt[1],popt[2],popt[3])

plt.figure()
for lambdanm in lambdasinnm:
    phase = KIDdict[lambdanm].phase
    t = KIDdict[lambdanm].t_full
    t -= t[np.argmax(phase)]
    plt.plot(simdata[lambdanm].t_axis_interp,simdata[lambdanm].phaseintime,alpha=0.5,label=r'$\lambda_{ph}=$\
    '+str(lambdanm)+r'nm')
    plt.scatter(t,phase,marker='.',s=10)
plt.xlim(10*simdata[lambdanm].t_start,50)
plt.xlabel(r'time ($\mu s$)')
plt.ylabel(r'$\theta$ (rad)')
#plt.ylim(1e-3)
#plt.semilogy()
plt.legend()
plt.show()
plt.figure()
for lambdanm in lambdasinnm:
    phase = KIDdict[lambdanm].phase
    t = KIDdict[lambdanm].t_full
    t -= t[np.argmax(phase)]
    plt.plot(simdata[lambdanm].t_axis_interp,simdata[lambdanm].phaseintime*lambdanm,alpha=0.5,label=r'$\lambda_{ph}=$\
    '+str(lambdanm)+r'nm')
    plt.scatter(t,phase*lambdanm,marker='.',s=10)
plt.xlim(10*simdata[lambdanm].t_start,50)
plt.xlabel(r'time ($\mu s$)')
```

```
26  plt.ylabel(r'$\theta\cdot\lambda_{ph}$ (rad$\cdot nm$)')
27  #plt.ylim(1e-3)
28  #plt.semilogy()
29  plt.legend()
30  plt.show()
```

Appendix E

MKIDs with β-Ta Subvolumes

An overview of the MKIDs with β-Ta subvolumes whose data is used in this project. Steven de Rooij and I came up with the concept. They were designed by Steven and produced by Tonny Coppens.

MKID no	elm. length
1 (high Q)	6 μm
2 (high Q)	10 μm
3 (high Q)	14 μm
4 (high Q)	24 μm
5 (high Q)	36 μm
6 (high Q)	71.6 μm
7 (high Q)	930 μm
10 (low Q)	6 μm
11 (low Q)	10 μm
12 (low Q)	14 μm
13 (low Q)	24 μm
14 (low Q)	48 μm
15 (low Q)	282 μm

Table E.1: The sub-element lengths per β-Ta MKID on chip `bTa_LT354chip3`.