NTNU – Trondheim
Norwegian University of
Science and Technology

Density of States and Critical Temperature in Superconductor/Ferromagnet Structures with Spin-Orbit Coupling

Jabir Ali Ouassou

Master of Science in Physics and Mathematics
Submission date: June 2015
Supervisor: Jacob Rune Wüsthoff Linder, IFY

Norwegian University of Science and Technology
Department of Physics

ABSTRACT

We consider superconductor/ferromagnet hybrid systems with intrinsic spin-orbit coupling, and investigate theoretically how the proximity effect manifests in two measurable quantities, namely the density of states and critical temperature. For concreteness, the spin-orbit coupling is taken to be of the experimentally common Rashba–Dresselhaus type. Our results are obtained via the quasiclassical theory of superconductivity, which has been extended to include spin-orbit coupling as an $\mathrm{SU}(2)$ gauge field. We present both an analytical discussion of the weak proximity regime, and a full numerical investigation of the strong proximity regime. The results show that the spin-orbit coupling leaves a clear trace in the density of states, which displays a highly nonmonotonic behaviour as a function of magnetization directions and phase differences. We also determine how the critical temperature is affected and, interestingly, demonstrate that one can achieve a spin-valve effect using a single ferromagnet. The critical temperature is found to exhibit a highly nonmonotonic behavior, both as a function of magnetization direction and type of spin-orbit coupling. Compared to the earlier inhomogeneously magnetized structures, this offers a new way to control the superconductivity of proximity structures.

SAMMENDRAG

I denne masteroppgaven granskes superleder/ferromagnet-hybridsystem med iboende spinn–bane-kobling, og det undersøkes spesielt hvordan samspillet deres påvirker tilstandstettheten og den kritiske temperaturen. Det antas at spinn–bane-koblingen er av Rashba–Dresselhaus-typen. Det tas utgangspunkt i den kvasiklassiske tilnærmingen til superledning, og mer spesifikt, en utvidet versjon som tar med spinn–bane-kobling som et $SU(2)$-justerfelt. Vi presenterer både en analytisk diskusjon ved svake vekselvirkninger, og en full numerisk granskning ved sterke vekselvirkninger. Resultatene viser at spinn–bane-koblingen har en klar påvirkning på tilstandstettheten, som får en sterkt ikke-monoton oppførsel som funksjon av både magnetiseringsvinkler og faseforskjeller. Vi kommer også fram til at den kritiske temperaturen har en sterkt ikke-monoton oppførsel som funksjon av magnetiseringsretning og typen spinn–bane-kobling. Dette kan lede til nye måter å kontrollere superledningen i hybridstrukturer på, som vil muliggjøre f.eks. spinnventiler bestående av bare én ferromagnet.

PREFACE

This document was submitted as a master thesis in theoretical condensed matter physics at the Norwegian University of Science and Technology, under supervision of professor Jacob Linder. It marks the conclusion of a 10 semester integrated programme in Applied Physics and Mathematics, and amounts to 30 ECTS-credits in the final semester of the programme. The thesis builds upon results from a 15 ECTS-credit specialization project in the 9th semester, which has been made available online for the readers' convenience.[1] The work presented herein has also resulted in a paper, which has been submitted to the journal Physical Review B, and is currently undergoing review. Meanwhile, a preprint is available on ArXiv,[2] and has been enclosed at the back. The perhaps most time-consuming part of the work has been to write, test, and execute programs that calculate the density of states and critical temperature of various superconducting hybrid structures; the resulting code base has also been enclosed.

First and foremost, I am grateful to my supervisor Jacob Linder for his insightful guidance over the past year. Our weekly discussions have been a great source of learning and inspiration for me, and I look forward to our continuing collaboration over the next few years. I would also like to thank Sol Jacobsen for our collaboration on the paper, and for very useful discussions regarding both the analytical derivations and computer simulations. I also wish to thank Einar Nyberg Karlsen for helping me proofread the thesis, and acknowledge very helpful discussions with Inge Madshaven and Ingvild Gomperud over the past year. Finally, I want to thank my parents for supporting and encouraging my studies.

This thesis was typeset in LaTeX, using the design `classicthesis` for the document itself, combined with a BibLaTeX style based on `nature`. Most of the figures were typeset in LaTeX using the package `tikz`. The numerical work was done in Matlab, and made heavy use of a 6th order MIRK-algorithm from the package `bvp6c` to solve the nonlinear boundary value problems.[3] The resulting data was either plotted in Matlab using the internal commands `plot` and `surf`, or in R using the package `ggplot2`.

Jabir Ali Ouassou
Trondheim, Norway
June 2015

Most of the mathematical notation in this thesis will follow the usual conventions in physics. In particular, both real and complex scalars are written with a light typeface as in $z = x + iy$, while three-dimensional vectors and matrices are written with a bold typeface as in $\boldsymbol{A}\boldsymbol{x} = \mathbf{b}$. However, the Cartesian unit vectors will be denoted by $\boldsymbol{e}_x, \boldsymbol{e}_y, \boldsymbol{e}_z$, without the conventional "hats", to avoid confusion with the notation for matrices in Nambu space. Partial derivatives are written in the short-hand form $\partial_x \equiv \partial/\partial x$, so the operator $\nabla \equiv \partial_x \boldsymbol{e}_x + \partial_y \boldsymbol{e}_y + \partial_z \boldsymbol{e}_z$. Other common notation includes using a superscript dagger† for Hermitian conjugation, a superscript asterisk* for complex conjugation, a superscript^T for matrix transposition, square brackets for the commutator $[\boldsymbol{A}\,,\,\mathbf{B}] \equiv \boldsymbol{A}\mathbf{B} - \mathbf{B}\boldsymbol{A}$, and curly braces for the anticommutator $\{\boldsymbol{A}\,,\,\mathbf{B}\} \equiv \boldsymbol{A}\mathbf{B} + \mathbf{B}\boldsymbol{A}$. Equations with the triple equality sign $\equiv$ should be taken as definitions, while $\cong$ is used for approximate equality.

In order to make the physical nature of the mathematical objects clear, we will also make use of some less common notation. Symbols with an underbar such as $\underline{S}$ will be used for 2×2 complex matrices in spin space, and symbols with a hat such as $\hat{P}$ will be used for similar matrices in Nambu space. Both these spaces are spanned by the usual Pauli matrices:

$$\underline{\sigma}_0 \equiv \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad \underline{\sigma}_x \equiv \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad \underline{\sigma}_y \equiv \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad \underline{\sigma}_z \equiv \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix};$$

$$\hat{\tau}_0 \equiv \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad \hat{\tau}_x \equiv \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad \hat{\tau}_y \equiv \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad \hat{\tau}_z \equiv \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}.$$

These basis matrices will also be combined to form the Pauli vectors, which possess both a three-dimensional vector structure in geometric space, and a matrix structure in either spin space or Nambu space:

$$\underline{\boldsymbol{\sigma}} \equiv \underline{\sigma}_x \boldsymbol{e}_x + \underline{\sigma}_y \boldsymbol{e}_y + \underline{\sigma}_z \boldsymbol{e}_z\,, \qquad \hat{\boldsymbol{\tau}} \equiv \hat{\tau}_x \boldsymbol{e}_x + \hat{\tau}_y \boldsymbol{e}_y + \hat{\tau}_z \boldsymbol{e}_z\,.$$

Objects with a structure in both spin space and Nambu space are spanned by the 16 matrices $\hat{\tau}_n \otimes \underline{\sigma}_m$ that are formed by the Kronecker products of the Pauli matrices above. Note that throughout this thesis, we use the convention that sums and products of incompatible matrices should be

resolved by taking Kronecker products with appropriate identity matrices. For instance, the expression $a + b\underline{\sigma}_x + c\hat{\tau}_y$ should be thought of as a short-hand notation for $a(\hat{\tau}_0 \otimes \underline{\sigma}_0) + b(\hat{\tau}_0 \otimes \underline{\sigma}_x) + c(\hat{\tau}_y \otimes \underline{\sigma}_0)$, and $\hat{\tau}_n\underline{\sigma}_m$ should be interpreted as $(\hat{\tau}_n \otimes \underline{\sigma}_0)(\hat{\tau}_0 \otimes \underline{\sigma}_m) = \hat{\tau}_n \otimes \underline{\sigma}_m$. Beware of trace expressions like $\mathrm{Tr}\{a + b\underline{\sigma}_x + c\hat{\tau}_y\}$, where the result depends on whether the trace is a sum over the diagonal in spin space, Nambu space, or both.

In theoretical physics, it is common practice to select a unit system where the most common physical constants get numerical values of one, so that these constants can be removed from the equations. Throughout this thesis, we will use *rationalized natural units* unless otherwise stated, which means that the following constants are normalized to unity: Planck's reduced constant $\hbar$; the rationalized gravitational constant $4\pi G$; the speed of light c; and Boltzmann's constant k. Since $c = 1/\sqrt{\epsilon_0\mu_0}$, we can also set the vacuum permittivity ϵ_0 and permeability μ_0 to unity. So the unit system can be summarized by the equation:

$$\hbar = k = 4\pi G = c = \epsilon_0 = \mu_0 = 1\,.$$

This equation uniquely defines a system of physical units, which can be related to any other unit system by plugging in the experimental values of the constants in the equation. So we can say that the philosophy of natural units is the exact opposite of conventional units: instead of *defining* the base units and *measuring* the physical constants, we define the physical constants and then determine the basic measures by experiment. A short list of common natural units, as well as conversion factors for equivalent SI units, is provided in table 1.

TABLE 1: Conversion table between natural units and some common SI units.

QUANTITY	NATURAL UNIT	CONVERSION FACTOR
Length	$\sqrt{4\pi G\hbar/c^3}$	5.73×10^{-35} m
Time	$\sqrt{4\pi G\hbar/c^5}$	1.91×10^{-43} s
Mass	$\sqrt{\hbar c/4\pi G}$	6.14×10^{-9} kg
Energy	$\sqrt{\hbar c^5/4\pi G}$	$5.52 \times 10^{+8}$ J
Temperature	$\sqrt{\hbar c^5/4\pi G k^2}$	$4.00 \times 10^{+31}$ K
Charge	$\sqrt{\hbar c \epsilon_0}$	5.29×10^{-19} C

CONTENTS

1 INTRODUCTION

1.1 BACKGROUND AND MOTIVATION

Superconductivity is one of the main research areas of modern condensed matter physics. Besides having interesting properties from a fundamental physics point of view, these materials have also found a wide range of practical applications. Their defining property is of course the dissipationless conductance of electric currents, which can greatly reduce the power consumption of electronic circuits, while at the same time reducing the cooling requirements of said circuits. Moreover, their other properties make them highly attractive for high-precision measurement instruments. For instance, magnetic field variations down to the attotesla scale can be detected by averaging SQUID measurements over a few days.[4] Another example would be superconducting cameras, which excite around 17,000 electrons per electronvolt when an incident photon is detected, permitting accurate measurements of individual photon colours.[5] Some suggestions for superconducting logic circuits also exist, including memory chips and computer processors that have already been demonstrated to work.[6–9]

Spintronics is another major research field in condensed matter physics, where the purpose is to process information by manipulating the flow of electron spins. Two main strategies are to produce spin-polarized electric currents using strong ferromagnets,[10] and conversely, to alter the field orientations of weaker ferromagnets using these spin-polarized currents.[11] Compared to conventional electronics, this technology may yield higher operation frequencies and a higher tolerance for unstable power supplies. As for commercially available technology, most harddrives are based on the giant magnetoresistive effect,[12,13] and MRAM memory chips have also been available for a decade.

Superconducting spintronics is a relatively new field, where the main goal is to combine the strengths of superconductivity and spintronics.[14,15] Both of these fields have been active for some time, and a lot of attention has also been given to their interplay. This might seem a bit strange, since superconductivity and magnetism are well-known to be two antagonistic orders. On one hand, superconductors expel magnetic fields according to the Meissner effect: an applied magnetic field induces perfect mirror cur-

rents in the superconductor, which in turn cancel the applied field inside the superconductor. On the other hand, conventional superconductivity arises due to a condensate of spin-0 electron pairs, and these pairs would be ripped apart by a ferromagnetic spin-splitting. Thus, the technologies might at first sight seem fundamentally incompatible.

While the conclusions above are accurate for interactions at the macroscopic level, the situation is quite different at the mesoscopic level.[a] In practice, there will always be a boundary layer around the interface where properties from both materials coalesce. In this region, spin-0 electron pairs from the superconductor fall under the influence of the magnetic field, and may be transformed into different kinds of spin-1 pairs. This solves the coexistence problem by superconductivity adapting to the presence of magnetic fields near the interface, yielding physical phenomena known as *triplet proximity effects*. However, advanced nanotechnology is required to create and manipulate such structures, so these effects were experimentally inaccessible until around 15 years ago.

While a homogeneous ferromagnet may transform electron pairs from the singlet state $|\uparrow\downarrow\rangle - |\downarrow\uparrow\rangle$ to the triplet state $|\uparrow\downarrow\rangle + |\downarrow\uparrow\rangle$, the electrons still have opposite spins, and eventually dissociate in a magnetic field. In order for superconductivity to penetrate deeper into the ferromagnet, the electron pairs would have to be rotated into one of the long-range triplet states $|\uparrow\uparrow\rangle$ or $|\downarrow\downarrow\rangle$, which consist of two electrons with the same spin. One way to accomplish this is using inhomogeneous magnetic fields, e.g. multiple homogeneous ferromagnets with noncollinear magnetizations.[16,17] While such experiments have been performed with success, it has nevertheless proven notoriously difficult to control the relative magnetization between different layers in mesoscopic multilayer structures, since the magnetic elements tend to spontaneously align. To avoid this problem, one alternative is to use a single ferromagnetic layer with an intrinsically inhomogeneous field, like the conical ferromagnet holmium. However, unlike the situation with two ferromagnets, this approach does not offer a simple way to directly control the triplet generation externally.

Another possibility which was recently suggested, is that long-range triplets may be generated using a single ferromagnetic material with a property known as *spin-orbit coupling*.[18,19] As we will see in later chapters,

[a] With *mesoscopic systems*, we mean materials of size 1 nm to 1 µm. These are typically small enough for quantum mechanical behaviour to start manifesting, and at the same time large enough for collective phenomena like superconductivity and ferromagnetism to take place. With the last statement, we refer to the fact that sub-nanometre structures are typically too small to exhibit this kind of large-scale ordering.

multilayer structures based on these materials not only permit production of long-range triplets under the right circumstances, but also enable an extensive external control over both singlet and triplet superconductivity by rotating the magnetic field.

1.2 SCOPE AND STRUCTURE

In this thesis, we consider mesoscopic superconductor/ferromagnet thin-film structures with spin-orbit coupling of the Rashba–Dresselhaus type. We investigate how the relationship between the singlet pairs and triplet pairs in the system changes as a function of the various parameters, and how this affects physical observables like density of states and critical temperature. The investigation is part analytical and part numerical, and ultimately results in physical predictions that may be tested empirically.

In the first half of chapter 2, we present the fundamental mathematical and physical formalism, including the concepts of superconductivity, ferromagnetism, spin-orbit coupling, Green's functions, and the equations that bind these concepts together. In the second half of the chapter, we proceed to review some of the well-known proximity-coupled systems that have been investigated before, including superconductivity in normal metals, ferromagnets, spin-active interfaces, and Josephson junctions. We focus on explaining their behaviour in an intuitive manner and how their physical properties would be observed in a laboratory setting. In chapter 3, we present a derivation of the self-consistency equation for the superconducting gap, both in the Keldysh and Matsubara formalisms. The original research starts with chapter 4, where an analytical investigation of spin-orbit coupling in superconductor/ferromagnet structures is performed. This is succeeded by a full numerical investigation of the density of states and critical temperature in such structures in chapter 5, where we consider the strong proximity regime. Finally, we conclude with a short summary and outlook in chapter 6.

After the content outlined above, we have also enclosed some extra material related to this thesis. The algorithm that was used to determine the critical temperature in chapter 5 is presented in appendix A. After the bibliography, we also include a paper that was written in conjunction with the work presented herein. Finally, MATLAB implementations of the programs that were used to generate the data in chapter 5 are enclosed at the end of the book.

2

FUNDAMENTAL CONCEPTS

2.1 SUPERCONDUCTIVITY

The physical origin of conventional low-temperature superconductivity is a phonon-mediated attraction between the electrons in a crystal.[20,21] In the BCS formalism,[22] this is described by the mean-field Hamiltonian

$$\mathcal{H} = -\int d^3\mathbf{x} \ \left[\Delta(\mathbf{x},t)\psi_\downarrow^\dagger(\mathbf{x},t)\psi_\uparrow^\dagger(\mathbf{x},t) + \Delta^*(\mathbf{x},t)\psi_\uparrow(\mathbf{x},t)\psi_\downarrow(\mathbf{x},t)\right], \quad (2.1)$$

where the mean-field is defined as

$$\Delta(\mathbf{x},t) \equiv \lambda\langle\psi_\uparrow(\mathbf{x},t)\psi_\downarrow(\mathbf{x},t)\rangle\,. \quad (2.2)$$

In the equations above, the operators $\psi_\sigma^\dagger(\mathbf{x},t)$ and $\psi_\sigma(\mathbf{x},t)$ create or annihilate quasielectrons with spin σ at position $\mathbf{x}$ and time t, and $\lambda > 0$ is the effective coupling constant of the electron–phonon–electron interaction that results in superconductivity. In general, the mean-field $\Delta = |\Delta|e^{i\varphi}$ is a complex function, where the norm $|\Delta|$ is called the *superconducting gap*, and the argument φ the *superconducting phase*. However, it is possible to perform a U(1) gauge transformation $\psi_\sigma \mapsto \psi_\sigma e^{i\varphi/2}$ in order to cancel the superconducting phase φ from the Hamiltonian. We will therefore assume a gauge where Δ is real, except when considering hybrid systems with multiple superconducting components such as Josephson junctions.

2.2 FERROMAGNETISM

Ferromagnetism is a phenomenon where so-called *exchange interactions* between the electrons in a material results in a macroscopic alignment of their spins.[1,23] This can be described by the mean-field Hamiltonian

$$\mathcal{H} = -\int d^3\mathbf{x} \ \sum_{\sigma\sigma'} \psi_\sigma^\dagger(\mathbf{x},t)\left[\mathbf{h}(\mathbf{x},t)\cdot\underline{\boldsymbol{\sigma}}\right]_{\sigma\sigma'}\psi_{\sigma'}(\mathbf{x},t)\,, \quad (2.3)$$

where the *exchange field* $\mathbf{h}(\mathbf{x},t)$ encapsulates the magnetic properties of the ferromagnet, and $\underline{\boldsymbol{\sigma}}$ is the well-known Pauli vector.

Spin-orbit coupling refers to processes where the spin $\underline{\sigma}$ and momentum $\mathbf{p}$ of each quasielectron become coupled, usually due to a broken spatial symmetry in the crystal. Such phenomena are described by the linearized single-particle Hamiltonian[18,19]

$$H = -\mathbf{p} \cdot \underline{\mathbf{A}}/m\,, \tag{2.4}$$

where the *spin-orbit field* $\underline{\mathbf{A}}$ is an SU(2) vector field, i.e. an object with a vector structure in real space and a 2×2 matrix structure in spin space. Moving on to quantum field theory, the presence of a spin-orbit field can be accounted for by replacing partial derivatives of Green's functions by their SU(2) gauge covariant equivalents,[1,18,19]

$$\nabla(\,\cdot\,) \mapsto \tilde{\nabla}(\,\cdot\,) \equiv \nabla(\,\cdot\,) - i\big[\hat{\underline{\mathbf{A}}}\,,\,\cdot\,\big]\,, \tag{2.5}$$

where the spin-orbit field in Spin–Nambu space is $\hat{\underline{\mathbf{A}}} \equiv \operatorname{diag}(\underline{\mathbf{A}}, -\underline{\mathbf{A}}^*)$.

For a concrete example of such an interaction, consider a thin crystal layer in the xy-plane, such that the growth direction is along the z-axis. This crystal would have a *structure inversion asymmetry* near the interfaces, which can be present throughout the structure if it is sufficiently thin. The broken inversion symmetry results in a *Rashba spin-splitting*,[24]

$$H_R = \alpha(p_y \underline{\sigma}_x - p_x \underline{\sigma}_y)\,, \tag{2.6}$$

where α is called the Rashba coefficient. Other couplings with the same symmetry structure can be produced by subjecting the crystal to a uniaxial strain,[25,26] or through interactions between different occupied subbands in the material.[27] Another kind of spin-orbit coupling arises due to *bulk inversion asymmetry*, which occurs in crystals that lack an inversion centre. One example of such a *Dresselhaus spin-splitting* is[26,28]

$$H_D = \beta(p_y \underline{\sigma}_y - p_x \underline{\sigma}_x)\,, \tag{2.7}$$

where β is called the Dresselhaus coefficient. Couplings of the same kind can also be produced by subjecting the crystal to a biaxial strain.[25,26] There is a plethora of other alternatives when it comes to spin-orbit coupling, especially in noncentrosymmetric crystals,[29] but we restrict our attention to the experimentally common Rashba–Dresselhaus type in this thesis.

For a Rashba–Dresselhaus coupling in the xy-plane, comparing eqs. (2.4), (2.6) and (2.7) reveals that the appropriate spin-orbit field is

$$\underline{\mathbf{A}} = m(\beta \underline{\sigma}_x + \alpha \underline{\sigma}_y)\boldsymbol{e}_x - m(\beta \underline{\sigma}_y + \alpha \underline{\sigma}_x)\boldsymbol{e}_y\,. \tag{2.8}$$

It will prove prudent to introduce a new notation for such couplings, which will let us distinguish between the physical effects that derive from the strength of the coupling, and those that derive from the geometry. For this purpose, we will employ a polar notation defined by

$$m\alpha \equiv a \sin\chi\,, \qquad\qquad m\beta \equiv a\cos\chi\,, \tag{2.9}$$

where we call a the *spin-orbit strength* and χ the *spin-orbit angle*. From the definitions, we see that $\chi = 0$ corresponds to a pure Dresselhaus coupling, while $\chi = \pm\pi/2$ results in a pure Rashba coupling, with the geometric interpretation of χ illustrated in fig. 2.1. In this notation, eq. (2.8) becomes

$$\underline{\mathbf{A}} = a(\underline{\sigma}_x \cos\chi + \underline{\sigma}_y \sin\chi)\boldsymbol{e}_x - a(\underline{\sigma}_y \cos\chi + \underline{\sigma}_x \sin\chi)\boldsymbol{e}_y\,, \tag{2.10}$$

or in component form,

$$\underline{\mathbf{A}} = a\boldsymbol{e}_x \begin{pmatrix} 0 & e^{-i\chi} \\ e^{i\chi} & 0 \end{pmatrix} + a\boldsymbol{e}_y \begin{pmatrix} 0 & ie^{i\chi} \\ -ie^{-i\chi} & 0 \end{pmatrix}. \tag{2.11}$$

Note that $\underline{A}_x^2 = \underline{A}_y^2 = a^2$, which means that $\underline{\mathbf{A}}^2 = 2a^2$. Another useful property of the parametrization is that the transformation $\chi \mapsto 3\pi/2 - \chi$ is equivalent to switching $\underline{A}_x \leftrightarrow \underline{A}_y$.

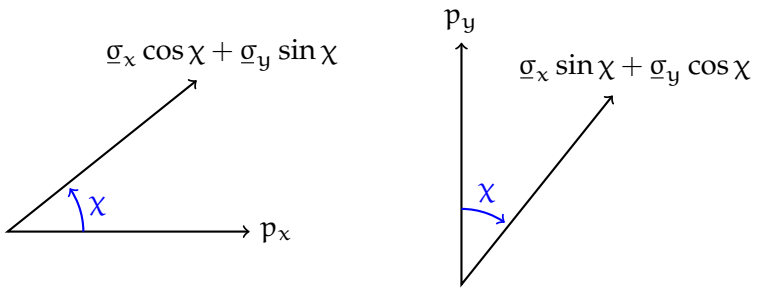


FIGURE 2.1: Geometric interpretation of the notation. The Hamiltonian couples the momentum component p_x to the spin component $\underline{\sigma}_x \cos\chi + \underline{\sigma}_y \sin\chi$ with a coefficient $-a/m$, and the momentum component p_y to the spin component $\underline{\sigma}_x \sin\chi + \underline{\sigma}_y \cos\chi$ with a coefficient $+a/m$.

Most modern treatments of condensed matter physics are expressed in the language of *Green's functions*, i.e. quantum field correlation functions. There are many different ways to define these functions, but this thesis is based on the Keldysh real-time formalism,[30–32] with the occasional translation to the Matsubara imaginary-time formalism.[33,34]

The cornerstone of the Keldysh formalism is a set of three Green's functions that describe the quantum transport of electrons and holes,

$$G^{R}_{\sigma\sigma'}(\mathbf{x},t;\mathbf{x}',t') \equiv -i\langle\{\psi_\sigma(\mathbf{x},t)\,,\,\psi^\dagger_{\sigma'}(\mathbf{x}',t')\}\rangle\theta(t-t')\,, \tag{2.12}$$

$$G^{A}_{\sigma\sigma'}(\mathbf{x},t;\mathbf{x}',t') \equiv +i\langle\{\psi_\sigma(\mathbf{x},t)\,,\,\psi^\dagger_{\sigma'}(\mathbf{x}',t')\}\rangle\theta(t'-t)\,, \tag{2.13}$$

$$G^{K}_{\sigma\sigma'}(\mathbf{x},t;\mathbf{x}',t') \equiv -i\langle[\psi_\sigma(\mathbf{x},t)\,,\,\psi^\dagger_{\sigma'}(\mathbf{x}',t')]\rangle\,, \tag{2.14}$$

along with their anomalous counterparts that describe Cooper pairs,

$$F^{R}_{\sigma\sigma'}(\mathbf{x},t;\mathbf{x}',t') \equiv -i\langle\{\psi_\sigma(\mathbf{x},t)\,,\,\psi_{\sigma'}(\mathbf{x}',t')\}\rangle\theta(t-t')\,, \tag{2.15}$$

$$F^{A}_{\sigma\sigma'}(\mathbf{x},t;\mathbf{x}',t') \equiv +i\langle\{\psi_\sigma(\mathbf{x},t)\,,\,\psi_{\sigma'}(\mathbf{x}',t')\}\rangle\theta(t'-t)\,, \tag{2.16}$$

$$F^{K}_{\sigma\sigma'}(\mathbf{x},t;\mathbf{x}',t') \equiv -i\langle[\psi_\sigma(\mathbf{x},t)\,,\,\psi_{\sigma'}(\mathbf{x}',t')]\rangle\,. \tag{2.17}$$

These matrices are then combined to form three matrices in Nambu space,

$$\underline{\hat{G}}^{R} \equiv \begin{pmatrix} \underline{G}^{R} & \underline{F}^{R} \\ \underline{F}^{R*} & \underline{G}^{R*} \end{pmatrix}, \tag{2.18}$$

$$\underline{\hat{G}}^{A} \equiv \begin{pmatrix} \underline{G}^{A} & \underline{F}^{A} \\ \underline{F}^{A*} & \underline{G}^{A*} \end{pmatrix}, \tag{2.19}$$

$$\underline{\hat{G}}^{K} \equiv \begin{pmatrix} \underline{G}^{K} & \underline{F}^{K} \\ -\underline{F}^{K*} & -\underline{G}^{K*} \end{pmatrix}, \tag{2.20}$$

which can be further conjoined to form a single matrix in Keldysh space,

$$\check{G} \equiv \begin{pmatrix} \underline{\hat{G}}^{R} & \underline{\hat{G}}^{K} \\ 0 & \underline{\hat{G}}^{A} \end{pmatrix}. \tag{2.21}$$

The most interesting physical observables in superconductor/ferromagnet hybrid systems, such as the density of states and electrical currents, can then be expressed in terms of the 8×8 matrix function $\check{G}(\mathbf{x},t;\mathbf{x}',t')$.

In practice, it is often prohibitively difficult to find an exact solution for the Green's functions. We will therefore follow the standard procedure of

switching to the Wigner representation, Fourier transforming the relative coordinates, performing a quasiclassical approximation, and taking the dirty limit.[1,35–37] The approximation can be summarized by the equation

$$\check{g}(\mathbf{x}, \epsilon, t) \equiv \frac{i}{\pi} \int d\xi_{\mathbf{p}} \langle \check{G}(\mathbf{x}, \mathbf{p}, \epsilon, t) \rangle_F \,, \tag{2.22}$$

where ϵ is the quasiparticle energy, $\mathbf{p}$ is the momentum, $\xi_{\mathbf{p}} \equiv \mathbf{p}^2/2m - \mu$ is the kinetic energy measured from the Fermi level μ, and $\langle \cdot \rangle_F$ denotes an average over the Fermi surface. The approximation above is based on two crucial assumptions:

(i) $\check{G}$ is dominated by particles near the Fermi surface;

(ii) $\check{G}$ is nearly isotropic with respect to momentum.

The first assumption can be justified using the Pauli principle. At low temperatures, most states inside the Fermi sphere $|\mathbf{p}| \leqslant p_F$ are occupied. Since the Pauli principle prevents double occupancy of the states, this implies that there are few available states with $|\mathbf{p}| \ll p_F$ that particles can be scattered *into*. On the other hand, most states outside the Fermi sphere are unoccupied, so there are also few states with $|\mathbf{p}| \gg p_F$ that particles may be scattered *from*. Conservation of momentum then dictates that most of the allowed scatterings have to be between different states with $|\mathbf{p}| \cong p_F$. These are the interactions that drive the transport processes in the system, and which therefore dominate the behaviour of the Green's functions. As for the second assumption, this holds for highly diffusive systems, where the frequent impurity scattering extinguishes the anisotropic components of the Green's functions.[a] We will restrict our attention to the special case of thermal equilibrium herein, so the description is further simplified by a time independence. Under the set of assumptions outlined above, the Keldysh component can be written

$$\hat{g}^K = (\hat{g}^R - \hat{g}^A) \tanh(\epsilon/2T) \,, \tag{2.23}$$

and the advanced component can be written

$$\underline{\hat{g}}^A = -\hat{\tau}_z \, \underline{\hat{g}}^{R\dagger} \hat{\tau}_z \,, \tag{2.24}$$

so all the physical characteristics of the system are actually encoded in the retarded component $\underline{\hat{g}}^R$ due to these symmetries. Most of this thesis

[a] Note that the isotropic approximation breaks down near interfaces, so boundary conditions for $\check{g}$ must be derived from a microscopic theory using the exact Green's functions.

is dedicated to calculating $\underline{\hat{g}}^R$ for superconducting hybrid systems, and using the results to calculate physical observables in equilibrium.

The 4×4 components of the retarded function $\underline{\hat{g}}^R$ are not independent, since symmetries between electron-like and hole-like excitations affects the Nambu space structure of $\underline{\hat{g}}^R$,[b]

$$\underline{\hat{g}}^R(\mathbf{x}, \epsilon) = \begin{pmatrix} \underline{g}^R(\mathbf{x}, +\epsilon) & \underline{f}^R(\mathbf{x}, +\epsilon) \\ -\underline{f}^{R*}(\mathbf{x}, -\epsilon) & -\underline{g}^{R*}(\mathbf{x}, -\epsilon) \end{pmatrix}. \tag{2.25}$$

Moreover, the matrix is subject to the normalization condition $(\underline{\hat{g}}^R)^2 = 1$, which further constrains the components of $\underline{\hat{g}}^R$ by relating $\underline{g}^R$ and $\underline{f}^R$,

$$\underline{g}^R(\mathbf{x}, \epsilon)\underline{g}^R(\mathbf{x}, \epsilon) - \underline{f}^R(\mathbf{x}, \epsilon)\,\underline{f}^{R*}(\mathbf{x}, -\epsilon) = 1\,, \tag{2.26}$$

$$\underline{g}^R(\mathbf{x}, \epsilon)\,\underline{f}^R(\mathbf{x}, \epsilon) - \underline{f}^R(\mathbf{x}, \epsilon)\underline{g}^{R*}(\mathbf{x}, -\epsilon) = 0\,. \tag{2.27}$$

Remarkably, by parametrizing $\underline{\hat{g}}^R$ in a way that automatically satisfies the symmetry and normalization requirements above, then the 4×4 matrix equations for $\underline{\hat{g}}^R$ can be reduced to just 2×2 matrix equations. In this thesis, we employ the so-called *Riccati parametrization* for this purpose,[38]

$$\underline{\hat{g}}^R = \begin{pmatrix} \underline{N} & 0 \\ 0 & -\underline{\tilde{N}} \end{pmatrix} \begin{pmatrix} 1 + \underline{\gamma}\underline{\tilde{\gamma}} & 2\underline{\gamma} \\ 2\underline{\tilde{\gamma}} & 1 + \underline{\tilde{\gamma}}\underline{\gamma} \end{pmatrix}, \tag{2.28}$$

where the normalization matrices $\underline{N} \equiv (1 - \underline{\gamma}\underline{\tilde{\gamma}})^{-1}$ and $\underline{\tilde{N}} \equiv (1 - \underline{\tilde{\gamma}}\underline{\gamma})^{-1}$, and the *tilde conjugation* $\tilde{\underline{\gamma}}(\mathbf{x}, \epsilon) \equiv \underline{\gamma}^*(\mathbf{x}, -\epsilon)$ is a combination of complex conjugation $i \mapsto -i$ and energy $\epsilon \mapsto -\epsilon$. Using the parametrization above, solving the equations for the 2×2 matrix function $\underline{\gamma}(\mathbf{x}, \epsilon)$ is sufficient to uniquely construct the whole Green's function $\underline{\hat{g}}^R(\mathbf{x}, \epsilon)$. It is also noteworthy that $\underline{\hat{g}}^R \to 1$ when $\underline{\gamma} \to 0$, while all the elements of $\underline{\hat{g}}^R$ diverge to infinity when $\underline{\gamma} \to 1$; so we see that a finite range of variation in $\underline{\gamma}$ parametrizes an infinite range of variation in the elements of $\underline{\hat{g}}^R$. This choice of parametrization has proven especially useful for numerical simulations of superconductivity because $\underline{\gamma}$ is single-valued and bounded.

In the Matsubara formalism, the field operators $\psi_\sigma(\mathbf{x}, t)$ and $\psi^\dagger_\sigma(\mathbf{x}, t)$ are analytically continued to imaginary times $t = -i\tau$ to describe systems in thermal equilibrium, where $|\tau| < 1/T$ at temperature T. We define the Matsubara Green's functions as

$$G^M_{\sigma\sigma'}(\mathbf{x}, \tau; \mathbf{x}', \tau') \equiv -i\langle T_\tau \psi_\sigma(\mathbf{x}, \tau)\bar{\psi}_{\sigma'}(\mathbf{x}', \tau')\rangle\,, \tag{2.29}$$

$$F^M_{\sigma\sigma'}(\mathbf{x}, \tau; \mathbf{x}', \tau') \equiv -i\langle T_\tau \psi_\sigma(\mathbf{x}, \tau)\psi_{\sigma'}(\mathbf{x}', \tau')\rangle\,, \tag{2.30}$$

[b] The reason eq. (2.25) has a different sign structure from eq. (2.18), is the imaginary unit in eq. (2.22), which results in an extra minus sign upon complex conjugation.

where $\psi_\sigma(\mathbf{x},\tau)$ is the analytic continuation of $\psi_\sigma(\mathbf{x},t)$, $\bar{\psi}_\sigma(\mathbf{x},\tau)$ is the analytic continuation of $\psi^\dagger_\sigma(\mathbf{x},t)$, and T_τ orders field operators such that τ increases from right to left. Like the other Green's functions discussed so far, the above can be combined to form a matrix $\hat{\underline{G}}^{\mathrm{M}}$ in Nambu space. After a quasiclassical approximation, the Matsubara functions can also be expressed in terms of the retarded and advanced functions,

$$\hat{\underline{g}}^{\mathrm{M}}(\mathbf{x},\omega) = \begin{cases} i\hat{\underline{g}}^{\mathrm{R}}(\mathbf{x},i\omega), & \omega \geqslant 0; \\ i\hat{\underline{g}}^{\mathrm{A}}(\mathbf{x},i\omega), & \omega < 0. \end{cases} \tag{2.31}$$

Note that the i's in eqs. (2.29) and (2.30) are unconventional, and also the source of the extra i in eq. (2.31). These definitions of $\underline{G}^{\mathrm{M}}$ and $\underline{F}^{\mathrm{M}}$ were chosen to eliminate some extraneous i's from the equations in chapters 3 and 4. This alternative definition of the Green's function is equivalent to rotating the superconducting gap $\Delta \mapsto i\Delta$ in the complex plane, which in turn is equivalent to a $\mathrm{U}(1)$ gauge transformation. We also note that since the imaginary time τ is restricted to the region $(-1/T, 1/T)$, ω can only be one of the discrete *Matsubara frequencies* $\omega_n = (2n+1)\pi T$ for $n \in \mathbb{Z}$. In superconductors, the relevant Matsubara frequencies are those below the Debye frequency ω_c, so $n \in (-n_c, n_c)$ with the cutoff $n_c \cong \omega_c/2\pi T$.

It is the anomalous Green's functions that describe the dynamics of Cooper pairs, and therefore the formation and behaviour of superconductivity. The anomalous function $\underline{f}$, which should be interpreted as either $\underline{f}^{\mathrm{R}}$ or $\underline{f}^{\mathrm{M}}$ depending on the formalism employed, can be parametrized as

$$\underline{f} = (f_s + \mathbf{f}_t \cdot \underline{\boldsymbol{\sigma}})i\underline{\sigma}_y, \tag{2.32}$$

or in component form,

$$\begin{pmatrix} f_{\uparrow\uparrow} & f_{\uparrow\downarrow} \\ f_{\downarrow\uparrow} & f_{\downarrow\downarrow} \end{pmatrix} = \begin{pmatrix} if_y - f_x & f_z + f_s \\ f_z - f_s & if_y + f_x \end{pmatrix}. \tag{2.33}$$

The *singlet component* f_s corresponds to conventional spin-0 electron pairs, and transforms as a scalar under spin rotations; while the *triplet component* $\mathbf{f}_t = (f_x, f_y, f_z)$ describes the more exotic spin-1 pairs, and transforms as a vector.[39,40] Furthermore, the triplet component can be further decomposed into a *short-range triplet component* $f_\parallel$ along the exchange field $\mathbf{h}$, which decays over a characteristic length scale $\sqrt{D/h}$ in a ferromagnet; and a *long-range triplet component* $f_\perp$ perpendicular to $\mathbf{h}$, which decays over the larger length scale $\sqrt{D/2\pi T}$. This decomposition of the Green's functions will play an important role in the derivations in chapter 4.

The fundamental physical quantity that describes the transport properties of a classical electric circuit is the electric current,

$$\mathbf{I}_e = GL\mathbf{E}\,, \tag{2.34}$$

where G is the conductance of the material, L the length of the material, and $\mathbf{E}$ the electric field in the material. Similarly, one may define a matrix current $\check{\mathbf{I}}$ in Keldysh space to describe the transport properties of quasiclassical spintronic circuits,

$$\check{\mathbf{I}} \equiv GL\check{g}\tilde{\nabla}\check{g}\,, \tag{2.35}$$

which is subdivided into the same components as the Green's functions,

$$\check{\mathbf{I}} \equiv \begin{pmatrix} \hat{\underline{\mathbf{I}}}^{R} & \hat{\underline{\mathbf{I}}}^{K} \\ 0 & \hat{\underline{\mathbf{I}}}^{A} \end{pmatrix}. \tag{2.36}$$

In this thesis, we consider thin-film structures that are homogeneous in the xy-plane, in which case $\check{I}_x = \check{I}_y = 0$, and only $\check{I} \equiv \check{I}_z$ is relevant. This matrix function encapsulates information about the flow of particles, energy, charge, spin, phase coherence, etc. throughout the system. For instance, the energy current I_ϵ, electric current I_e, and spin current $\mathbf{I}_\sigma$ can all be extracted from the Keldysh current $\hat{\underline{I}}^{K}$,[41–45]

$$I_\epsilon = \frac{1}{4e^2}\int d\epsilon\, \mathrm{Tr}\big[\epsilon\hat{\underline{I}}^{K}(\epsilon)\big], \tag{2.37}$$

$$I_e = \frac{1}{4e}\int d\epsilon\, \mathrm{Tr}\big[\hat{\tau}_z\hat{\underline{I}}^{K}(\epsilon)\big], \tag{2.38}$$

$$\mathbf{I}_\sigma = \frac{1}{8e^2}\int d\epsilon\, \mathrm{Tr}\big[\hat{\tau}_z\hat{\underline{\sigma}}\hat{\underline{I}}^{K}(\epsilon)\big], \tag{2.39}$$

where $\hat{\underline{\sigma}} = \mathrm{diag}(\underline{\sigma}, \underline{\sigma}^*)$ is the spin operator in Nambu space. It can also be shown straight from the definitions that

$$\partial_z\underline{\gamma} = \big[2GL\underline{N}\big]^{-1}\big[\underline{I}^{R}_{12} - \underline{I}^{R}_{11}\underline{\gamma}\big] + i\underline{A}_z\underline{\gamma} + i\underline{\gamma}\underline{A}_z^*\,, \tag{2.40}$$

where $\underline{I}^{R}_{ij}$ refers to the (i,j) component in Nambu space of the retarded current $\underline{I}^{R}$. The boundary conditions for a physical system in equilibrium can therefore be conveniently expressed as a retarded matrix current. The spin-orbit terms come from using the general definition $\check{\mathbf{I}} \equiv GL\check{g}\tilde{\nabla}\check{g}$ in terms of the covariant derivative in eq. (2.5). However, for the spin-orbit coupling in eq. (2.10), these terms are inconsequential since $\underline{A}_z = 0$.

All the systems considered herein consist of diffusive materials with a superconducting gap $\Delta(\mathbf{x})$, exchange field $\mathbf{h}(\mathbf{x})$, and spin-orbit field $\underline{\mathbf{A}}(\mathbf{x})$. The quasiclassical Green's function $\hat{\underline{g}}^{\mathrm{R}}(\mathbf{x}, \epsilon)$ for such a system can be calculated from the Usadel diffusion equation,[1,2,18,19,31,35,37,46,c]

$$i D \tilde{\nabla} \cdot \left(\hat{\underline{g}}^{\mathrm{R}} \tilde{\nabla} \hat{\underline{g}}^{\mathrm{R}} \right) = \left[\epsilon \hat{\tau}_z + \hat{\underline{\Delta}} + \mathbf{h} \cdot \hat{\underline{\boldsymbol{\sigma}}} \, , \, \hat{\underline{g}}^{\mathrm{R}} \right] , \tag{2.41}$$

where D is the diffusion constant of the material, $\tilde{\nabla}$ is the gauge covariant derivative defined in eq. (2.5), the spin operator in Nambu space is

$$\hat{\underline{\boldsymbol{\sigma}}} \equiv \begin{pmatrix} \underline{\boldsymbol{\sigma}} & 0 \\ 0 & \underline{\boldsymbol{\sigma}}^* \end{pmatrix} , \tag{2.42}$$

and the superconducting gap matrix is

$$\hat{\underline{\Delta}} \equiv \begin{pmatrix} 0 & 0 & 0 & \Delta \\ 0 & 0 & -\Delta & 0 \\ 0 & \Delta^* & 0 & 0 \\ -\Delta^* & 0 & 0 & 0 \end{pmatrix} . \tag{2.43}$$

It is worth noting that $\mathbf{h} \cdot \hat{\underline{\boldsymbol{\sigma}}} = \hat{\tau}_0 \otimes (\mathbf{h} \cdot \underline{\boldsymbol{\sigma}})$ for an exchange field in the xz-plane, and that $\hat{\underline{\Delta}} = \Delta(\hat{\tau}_x \otimes i\underline{\sigma}_y)$ when we work in a real gauge. Performing a Riccati parametrization yields the equation

$$\begin{aligned} D\left[(\partial_z^2 \underline{\gamma}) + 2(\partial_z \underline{\gamma}) \underline{\tilde{N}} \underline{\tilde{\gamma}} (\partial_z \underline{\gamma})\right] \\ = &- 2i\epsilon \underline{\gamma} - \Delta \underline{\sigma}_y + \underline{\gamma} \Delta^* \underline{\sigma}_y \underline{\gamma} - i\mathbf{h} \cdot (\underline{\boldsymbol{\sigma}} \underline{\gamma} - \underline{\gamma} \underline{\boldsymbol{\sigma}}^*) \\ &+ D\left[\underline{\mathbf{A}} \, \underline{\mathbf{A}} \underline{\gamma} - \underline{\gamma} \underline{\mathbf{A}}^* \underline{\mathbf{A}}^* + 2(\underline{\mathbf{A}} \underline{\gamma} + \underline{\gamma} \underline{\mathbf{A}}^*) \underline{\tilde{N}} (\underline{\mathbf{A}}^* + \underline{\tilde{\gamma}} \underline{\mathbf{A}} \underline{\gamma})\right] \\ &+ 2iD\left[(\partial_z \underline{\gamma}) \underline{\tilde{N}} (\underline{A}_z^* + \underline{\tilde{\gamma}} \underline{A}_z \underline{\gamma}) + (\underline{A}_z + \underline{\gamma} \underline{A}_z^* \underline{\tilde{\gamma}}) \underline{N} (\partial_z \underline{\gamma})\right] , \end{aligned} \tag{2.44}$$

where we have assumed a thin-film hybrid structure with the z-axis as the layering direction.[1,2] Note that $\underline{A}_z = 0$ for the Rashba–Dresselhaus coupling in eq. (2.10), in which case the last line of the equation vanishes. On the other hand, if the spin-orbit field only couples to spins in the xz-plane, then $\underline{\mathbf{A}} = \underline{\mathbf{A}}^*$, which implies that the second line of the right-hand side could be written more compactly as $[\underline{\mathbf{A}}^2 \, , \, \underline{\gamma}] + 2\{\underline{\mathbf{A}} \, , \, \underline{\gamma}\} \underline{\tilde{N}} (\underline{\mathbf{A}} + \underline{\tilde{\gamma}} \underline{\mathbf{A}} \underline{\gamma})$.[48]

[c] Note that the left-hand side of the Usadel equation is essentially just the derivative of the matrix current $\tilde{\nabla} \cdot \hat{\underline{\mathbf{I}}}^{\mathrm{R}}$, so the diffusion equation may also be interpreted as an equation that describes the leakage of a matrix current.[41,42,47]

To determine an appropriate boundary condition for the matrix current, one may in principle devise a microscopic model of the interface, and use this to derive a scattering matrix for each available transport channel n. However, in practice, knowing the transmission probability T_n for each channel is sufficient. Assuming elastic scattering at a general spin-inactive interface, the matrix current satisfies the Nazarov boundary condition,[41,42]

$$\check{I} = 2G_Q \sum_n \frac{T_n \left[\check{g}_L\,,\, \check{g}_R\right]}{4 - 2T_n + T_n\left\{\check{g}_L\,,\, \check{g}_R\right\}}\,, \tag{2.45}$$

where $\check{g}_L$ and $\check{g}_R$ are the quasiclassical Green's functions at the left and right sides of the interface, respectively, and the prefactor $G_Q \equiv e^2/\pi$ is the conductance quantum. Very high transmission probabilities can be achieved using a variety of experimental setups, such as superconductors connected to monocrystalline semiconductor nanowires ($T \cong 0.75$),[49] monocrystalline metallic nanowires ($T \cong 0.95$),[50] monolayers of the same material ($T \cong 0.98$),[51] or monolayers of graphene ($T \cong 0.98$).[52] However, it is easier to manufacture interfaces with low transmission probabilities, since e.g. crystal structure misalignments and interfacial impurities tend to lower the transmission probability. In the *tunneling limit* $T_n \ll 1$, the Nazarov boundary condition reduces to the commonly employed Kuprianov–Lukichev boundary condition,[41,42,53]

$$2\check{I} = G_T \left[\check{g}_L\,,\, \check{g}_R\right], \tag{2.46}$$

where we introduced the *tunneling conductance* $G_T \equiv G_Q \sum_n T_n$. In other words, we do not need to know the full transmission spectrum $\{T_n\}$, as the single scalar G_T fully describes the interface in this limit.

The boundary conditions at magnetic interfaces are more complicated, and has only recently been derived after decades of research on the topic. There are two main new physical effects that requires addressing. Firstly, the transmission probabilities at a magnetic interface depend on the spins of the quasiparticles, resulting in *spin-filtering*. We therefore need two parameters to characterize each transmission channel, namely the average transmission probability $T_n \equiv (T_{n\uparrow} + T_{n\downarrow})/2$ and spin-polarization $P_n \equiv (T_{n\uparrow} - T_{n\downarrow})/(T_{n\uparrow} + T_{n\downarrow})$. Secondly, quasiparticles that reflect off a magnetic interface acquire different phase shifts $\varphi_{n\uparrow}$ or $\varphi_{n\downarrow}$ depending on spin, resulting in *spin-dependent interfacial phase shifts* $\phi_n \equiv \varphi_{n\uparrow} - \varphi_{n\downarrow}$. By causing a relative phase shift between the electrons in a Cooper pair, the spin-dependent phase shifts can cause a singlet pair to transform into a triplet pair, thus causing *spin-mixing*. To linear order in the transmission

probabilities and spin-dependent phase shifts, but for arbitrarily high spin-polarizations, the boundary condition may be written[44,45,54–57]

$$\check{I} = G_Q \sum_n \left[\underline{\hat{t}}_n \check{g}_L \underline{\hat{t}}_n^\dagger - i\phi_n (\mathbf{m} \cdot \underline{\hat{\boldsymbol{\sigma}}}) \, , \, \check{g}_R \right] , \tag{2.47}$$

where the transmission matrix $\underline{\hat{t}}_n \equiv t_0 + t_1 (\mathbf{m} \cdot \underline{\hat{\boldsymbol{\sigma}}})$, $\mathbf{m}$ is a unit vector that describes the interface magnetization, and the transmission coefficients $t_{n0} \equiv (t_{n\uparrow} + t_{n\downarrow})/2$ and $t_{n1} \equiv (t_{n\uparrow} - t_{n\downarrow})/2$ are real numbers. If we expand this equation, it can be written in the more explicit form

$$\begin{aligned} 2\check{I} = {} & G_{T0} \left[\check{g}_L \, , \, \check{g}_R \right] + G_{T1} \left[\check{g}_L \, , \, (\mathbf{m} \cdot \underline{\hat{\boldsymbol{\sigma}}}) \check{g}_R (\mathbf{m} \cdot \underline{\hat{\boldsymbol{\sigma}}}) \right] \\ & + G_{MR} \left[\check{g}_L \, , \, \left\{ \check{g}_R \, , \, \mathbf{m} \cdot \underline{\hat{\boldsymbol{\sigma}}} \right\} \right] + i G_\phi \left[\check{g}_L \, , \, \mathbf{m} \cdot \underline{\hat{\boldsymbol{\sigma}}} \right] . \end{aligned} \tag{2.48}$$

To more accurately identify the interesting parameter regimes, it can be convenient to describe the system using the dimensionless parameters

$$\mu \equiv \frac{G_{MR}}{G_{T0}} = \frac{\sum_n T_n P_n}{\sum_n T_n (1 + \sqrt{1 - P_n^2})} , \tag{2.49}$$

$$\lambda \equiv \frac{G_\phi}{G_{T0}} = \frac{-2 \sum_n \phi_n}{\sum_n T_n (1 + \sqrt{1 - P_n^2})} , \tag{2.50}$$

$$\kappa \equiv \frac{G_{T1}}{G_{T0}} = \frac{\sum_n T_n (1 - \sqrt{1 - P_n^2})}{\sum_n T_n (1 + \sqrt{1 - P_n^2})} , \tag{2.51}$$

where the magnetoresistive factor $\mu \in [-1, 1]$ describes spin-polarization, the phase factor $\lambda \in (-\infty, \infty)$ describes spin-dependent phase shifts, and $\kappa \in [0, 1]$ is a correction factor for high polarizations. This version of the equation describes $\check{I}$ using a sum of commutators with $\check{g}_L$, but a similar equation for the other side of the interface can be found by letting $\check{I} \mapsto -\check{I}$ and $L \leftrightarrow R$ in the equations above.

For non-magnetized tunneling junctions, it is relatively straight-forward to obtain boundary conditions for the Riccati parameter $\underline{\gamma}$ by substituting eq. (2.46) into eq. (2.40),[1,2]

$$\partial_z \underline{\gamma}_L = \frac{1}{L_L \zeta_L} (1 - \underline{\gamma}_L \underline{\tilde{\gamma}}_R) \underline{N}_R (\underline{\gamma}_R - \underline{\gamma}_L) + i \underline{\mathbf{A}}_z \underline{\gamma}_L + i \underline{\gamma}_L \underline{\mathbf{A}}_z^* , \tag{2.52}$$

$$\partial_z \underline{\gamma}_R = \frac{1}{L_R \zeta_R} (1 - \underline{\gamma}_R \underline{\tilde{\gamma}}_L) \underline{N}_L (\underline{\gamma}_R - \underline{\gamma}_L) + i \underline{\mathbf{A}}_z \underline{\gamma}_R + i \underline{\gamma}_R \underline{\mathbf{A}}_z^* , \tag{2.53}$$

where we introduced the dimensionless interface parameters $\zeta_j \equiv G_j / G_T$. Throughout this thesis, we will assume the value $\zeta = 3$ for all material interfaces, and $\zeta = \infty$ for vacuum interfaces. Corresponding boundary conditions for $\underline{\tilde{\gamma}}$ are found by tilde conjugation of the equations above.

Performing an explicit Riccati parametrization of eq. (2.48) is far from trivial. To be able to implement such boundary conditions numerically, a viable alternative is to use eq. (2.40) directly instead,

$$\partial_z \gamma_L = \left[2G_L L_L \underline{N}_L\right]^{-1}\left[\underline{I}^R_{12} - \underline{I}^R_{11}\gamma_L\right] + i\underline{A}_z\gamma_L + i\gamma_L\underline{A}^*_z\,, \tag{2.54}$$

where the components $\underline{I}^R_{11}$ and $\underline{I}^R_{12}$ are extracted from eq. (2.48),

$$\begin{aligned}
\frac{2\underline{I}^R_{11}}{G_{TO}} &= \left[\underline{g}_L\underline{g}_R - \underline{f}_L\tilde{\underline{f}}_R\right]\left[1+\mu(\mathbf{m}\cdot\underline{\boldsymbol{\sigma}})\right] \\
&- \left[1+\mu(\mathbf{m}\cdot\underline{\boldsymbol{\sigma}})\right]\left[\underline{g}_R\underline{g}_L - \underline{f}_R\tilde{\underline{f}}_L\right] \\
&+ \left[\underline{g}_L(\mathbf{m}\cdot\underline{\boldsymbol{\sigma}})\underline{g}_R - \underline{f}_L(\mathbf{m}\cdot\underline{\boldsymbol{\sigma}}^*)\tilde{\underline{f}}_R\right]\left[\mu+\kappa(\mathbf{m}\cdot\underline{\boldsymbol{\sigma}})\right] \\
&- \left[\mu+\kappa(\mathbf{m}\cdot\underline{\boldsymbol{\sigma}})\right]\left[\underline{g}_R(\mathbf{m}\cdot\underline{\boldsymbol{\sigma}})\underline{g}_L - \underline{f}_R(\mathbf{m}\cdot\underline{\boldsymbol{\sigma}}^*)\tilde{\underline{f}}_L\right] \\
&+ i\lambda\left[\underline{g}_L(\mathbf{m}\cdot\underline{\boldsymbol{\sigma}}) - (\mathbf{m}\cdot\underline{\boldsymbol{\sigma}})\underline{g}_L\right],
\end{aligned} \tag{2.55}$$

$$\begin{aligned}
\frac{2\underline{I}^R_{12}}{G_{TO}} &= \left[\underline{g}_L\underline{f}_R - \underline{f}_L\tilde{\underline{g}}_R\right]\left[1+\mu(\mathbf{m}\cdot\underline{\boldsymbol{\sigma}}^*)\right] \\
&- \left[1+\mu(\mathbf{m}\cdot\underline{\boldsymbol{\sigma}})\right]\left[\underline{g}_R\underline{f}_L - \underline{f}_R\tilde{\underline{g}}_L\right] \\
&+ \left[\underline{g}_L(\mathbf{m}\cdot\underline{\boldsymbol{\sigma}})\underline{f}_R - \underline{f}_L(\mathbf{m}\cdot\underline{\boldsymbol{\sigma}}^*)\tilde{\underline{g}}_R\right]\left[\mu+\kappa(\mathbf{m}\cdot\underline{\boldsymbol{\sigma}}^*)\right] \\
&- \left[\mu+\kappa(\mathbf{m}\cdot\underline{\boldsymbol{\sigma}})\right]\left[\underline{g}_R(\mathbf{m}\cdot\underline{\boldsymbol{\sigma}})\underline{f}_L - \underline{f}_R(\mathbf{m}\cdot\underline{\boldsymbol{\sigma}}^*)\tilde{\underline{g}}_L\right] \\
&+ i\lambda\left[\underline{f}_L(\mathbf{m}\cdot\underline{\boldsymbol{\sigma}}^*) - (\mathbf{m}\cdot\underline{\boldsymbol{\sigma}})\underline{f}_L\right],
\end{aligned} \tag{2.56}$$

and $g_j = 2\underline{N}_j - 1$, $f_j = 2\underline{N}_j\gamma_j$. The corresponding boundary condition for $\partial_z\gamma_R$ is then found by letting $z \mapsto -z$ and $L \leftrightarrow R$ in all the equations above, and boundary conditions for $\partial_z\tilde{\gamma}_L$ and $\partial_z\tilde{\gamma}_R$ are found by tilde conjugation of the equations for $\partial_z\gamma_L$ and $\partial_z\gamma_R$.

One of the key observable properties of a superconducting hybrid system is the local density of states $N_\sigma(\epsilon, z)$, i.e. the number of quantum states at position z that a quasiparticle with energy ϵ and spin σ could occupy. This quantity is proportional to the real part of the Green's function,[58]

$$N_\sigma(\epsilon, z) = N_0 \operatorname{Re}\{g^R_{\sigma\sigma}(\epsilon, z)\}, \tag{2.57}$$

where N_0 is the density of states per spin at the Fermi level when the material is in its normal state. Technically, N_0 should have been a function of energy, but in quasiclassical theory it is assumed that the quasiparticle energy ϵ is much smaller than the Fermi energy, in which case N_0 is nearly constant.[20] Since typical superconductors have gaps of order 1 meV,[59,60] while the Fermi energy is around 1–10 eV,[61] this usually holds very well. Numerically, it is more convenient to study the normalized density of states $D_\sigma \equiv N_\sigma/N_0$, and in most cases the system is adequately described by the spin-independent density of states $D \equiv (D_\uparrow + D_\downarrow)/2$. We therefore focus on the normalized and spin-independent density of states

$$D(\epsilon, z) = \frac{1}{2} \operatorname{Re} \operatorname{Tr}\{\underline{g}^R(\epsilon, z)\} \tag{2.58}$$

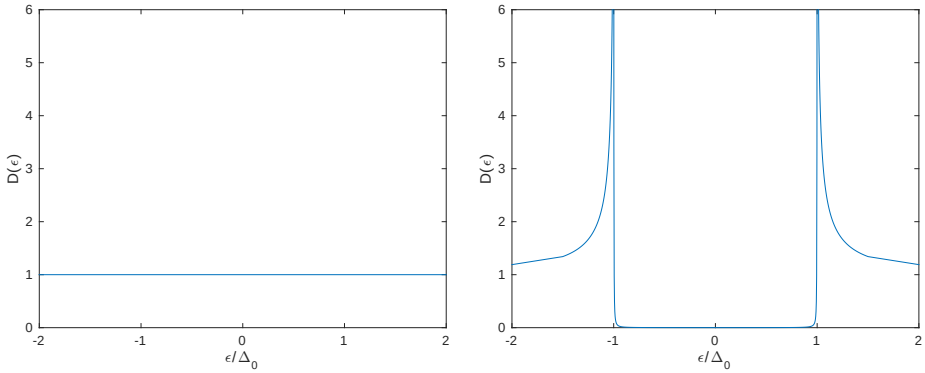
throughout this thesis, which after Riccati parametrization takes the form

$$D(\epsilon, z) = \frac{1}{2} \operatorname{Re} \operatorname{Tr}\{\underline{N}(1 + \underline{\gamma}\tilde{\underline{\gamma}})\}. \tag{2.59}$$

Although the density of states as defined above is a local quantity, we will frequently refer to $D(\epsilon)$ without explicitly specifying the location, in which case it is implied that it is measured at the center of the material. Notably, in the limit of weak superconductivity $\|\underline{f}^R\| \ll 1$, the zero-energy density of states $D(0)$ can be written in the form

$$D(0) \cong 1 - \frac{1}{2}\left|f^R_s(0)\right|^2 + \frac{1}{2}\left|f^R_t(0)\right|^2. \tag{2.60}$$

The characteristic density of states is therefore flat and featureless for a normal metal (fig. 2.2a), displays a zero-energy gap for singlet superconductors (fig. 2.2b), and has a zero-energy peak for triplet superconductors. Thus, measuring the density of states provides a simple yet effective way to determine whether the system is dominated by singlet or triplet pairs.



(a) Normal metal; (b) Superconductor.

FIGURE 2.2: Density of states $D(\epsilon)$ for a normal metal and a bulk superconductor. In the former case, the density of states is flat and featureless; in the latter, it is suppressed within the gap $\epsilon \in (-\Delta, +\Delta)$, but enhanced at the edges $\epsilon = \pm\Delta$.

Empirically, the density of states can be directly probed using tunneling spectroscopy.[20,21,62] The basic concept is that an electron with energy ϵ can only tunnel between two materials with a nonzero density of states $N(\epsilon)$, so the tunneling probability depends on a product $TN_L(\epsilon)N_R(\epsilon)$ between the interface transparency T, left density of states $N_L(\epsilon)$, and right density of states $N_R(\epsilon)$. However, when a voltage V is applied to the junction, the electron acquires the energy eV when tunneling, thus shifting the product to $TN_L(\epsilon)N_R(\epsilon + eV)$ instead. By using a normal metal with a constant density of states N_L as the left material, and a superconductor with an energy-dependent density of states $N_R(\epsilon)$ as the right material, measuring the tunneling probability as a function of voltage provides a way to probe the density of states of a superconductor. More precisely, what an experimentalist would measure is the net tunneling current

$$I = 2\pi T \int d\epsilon \, N_L(\epsilon) \big[n(\epsilon) - n(\epsilon + eV) \big] N_R(\epsilon + eV) \,, \tag{2.61}$$

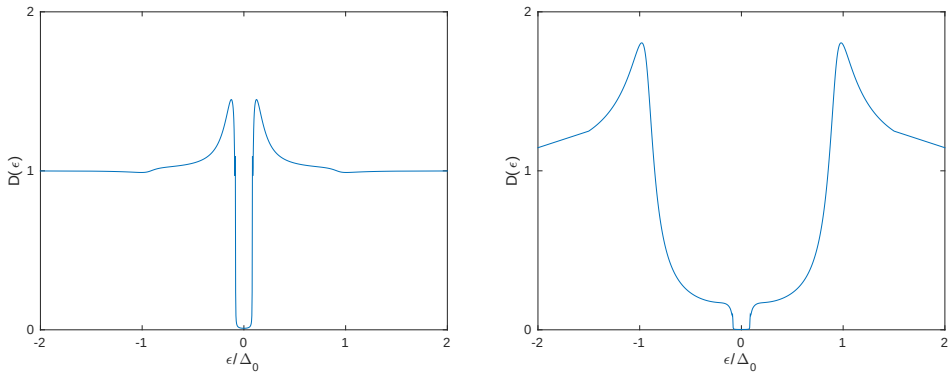
where $n(\epsilon) = 1/\big[1 + \exp(\epsilon/T)\big]$ is the Fermi–Dirac distribution function. At sufficiently low temperatures and voltages, this implies that

$$\frac{dI}{dV} \cong 2\pi e T \, N_L(0) \, N_R(eV) \,, \tag{2.62}$$

so measuring the differential conductance dI/dV provides a direct insight into the density of states in the superconductor for the energy $\epsilon = eV$.[20,21] These measurements can be performed with around 100 nm precision using metallic probes at predefined locations,[63] or in principle all the way down to atomic precision using a scanning tunneling microscope.[64,65]

When a superconductor is brought into contact with a normal metal, it can induce superconducting properties in the normal metal near the interface, while the same properties are suppressed in the superconductor. This can intuitively be understood as a leakage of Cooper pairs from the superconductor into the normal metal, or conversely, as a drainage of Cooper pairs from the superconductor by the normal metal. The microscopic explanation is given by Andreev reflections at the interface.[66–68] When an electron in the normal metal reaches the interface to a superconductor, and has a subgap energy $|\epsilon| < |\Delta|$, it cannot enter the material as a normal electron since the density of states is zero there. Instead, there are two possible outcomes: the electron can either be reflected as an electron, or Andreev reflected as a hole. In the latter case, what actually happens is that the electron joins up with another electron at the interface, and is transmitted into the superconductor as a Cooper pair. The net effect of this process is to transfer two subgap electrons from the normal metal to the superconductor, thus lowering the density of states in the normal metal, and inducing a minigap near the interface. Conversely, a hole can also be Andreev reflected off the interface as an electron, effectively removing a Cooper pair from the superconductor, and suppressing the superconductivity of the material. Combined, the Andreev reflection of electrons and holes explains how a minigap is induced in normal metals, and how the gap is suppressed in superconductors (see figs. 2.3 and 2.4). This *proximity effect* can be studied by measuring supercurrents up to roughly one coherence length from the interface, i.e. $\xi_S = \sqrt{D/\Delta}$ for superconductors, $\xi_T = \sqrt{D/2\pi T}$ for normal metals at temperature T, and $\xi_F = \sqrt{D/h}$ for ferromagnets with exchange field $\mathbf{h}$.

The figures we present on the next page highlight both the forward proximity effect, which is how the superconductor affects the normal metal, and the inverse proximity effect, which is how the superconductor reacts. Numerically, this requires a self-consistent approach, where we alternate between solving the equations in the superconductor and normal metal until the solution stabilizes. However, for sufficiently large superconductors and low interface transparencies, the density of states in the normal metal can be calculated to a close approximation while neglecting the inverse proximity effect, saving a tremendous amount of computation time. In the rest of this thesis, this simplification is made when computing the density of states. We emphasize that this is adequate for calculating observables in the normal metal in equilibrium, such as tunnelling currents and density of states, but not for calculating the critical temperature or non-equilibrium properties.



(a) Normal metal; (b) Superconductor.

FIGURE 2.3: Proximity effect. The superconductor induces a so-called minigap in the normal metal, and since the calculation was performed self-consistently, the normal metal simultaneously induces subgap states in the superconductor.

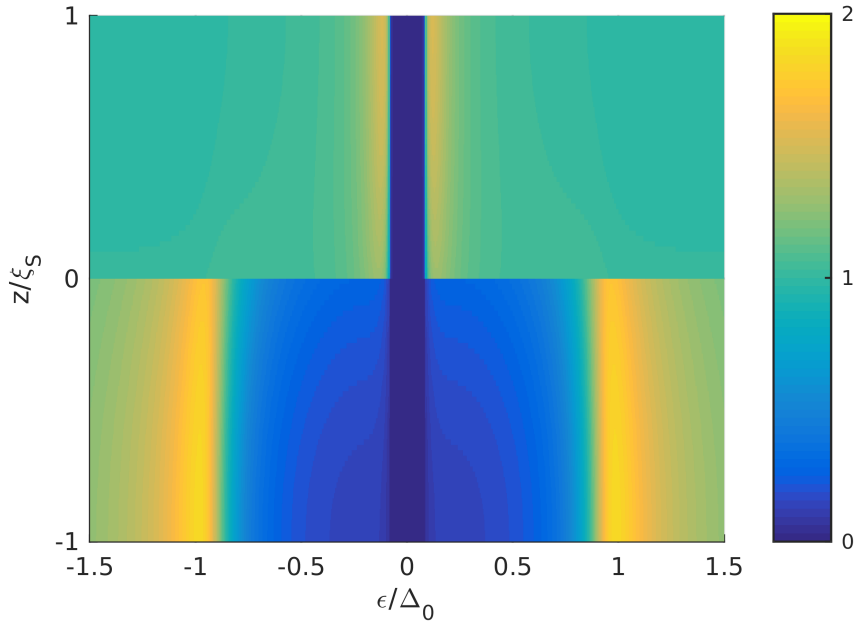


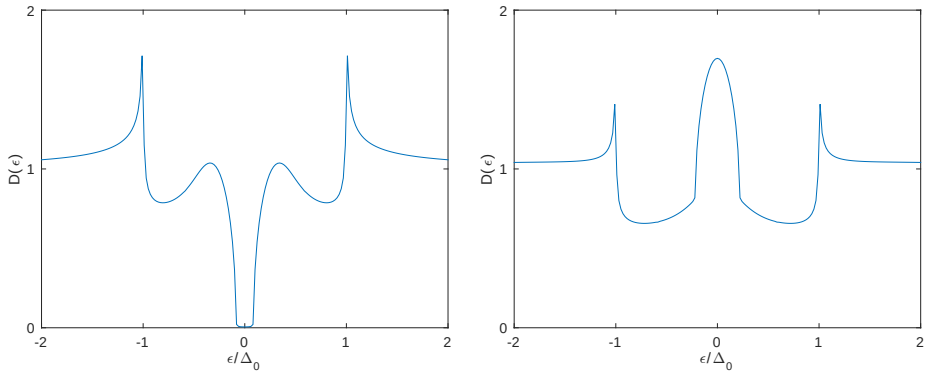
FIGURE 2.4: Local density of states $D(\epsilon, z)$ for a superconductor/ferromagnet bilayer. Since the length of the system is only $2\xi_S$, the superconductor manages to induce a minigap throughout the normal metal ($z > 0$), while the gap is weakened throughout the superconductor ($z < 0$).

When a superconductor is brought into contact with a ferromagnet, the proximity effect can manifest in radically different ways.[14,16,17,69–74] The essential physics can again be understood as a leakage of Cooper pairs into the adjacent material. In a conventional BCS superconductor, all the Cooper pairs reside in a singlet state $|\uparrow\downarrow\rangle - |\downarrow\uparrow\rangle$. However, in a ferromagnet the exchange field $\mathbf{h}$ produces a spin-splitting of the density of states, which causes the pairs to acquire a finite momentum $q = 2mh_z/p$. This in turn leads to spatial oscillations in the superconducting wave function, so the singlet state $|\uparrow\downarrow\rangle - |\downarrow\uparrow\rangle$ can transform into the triplet state $|\uparrow\downarrow\rangle + |\downarrow\uparrow\rangle$ as the pairs penetrate into the ferromagnet:

$$|\uparrow\downarrow\rangle - |\downarrow\uparrow\rangle \quad \mapsto \quad \cos qz \left[|\uparrow\downarrow\rangle - |\downarrow\uparrow\rangle \right] + i \sin qz \left[|\uparrow\downarrow\rangle + |\downarrow\uparrow\rangle \right]. \tag{2.63}$$

This is known as *spin-mixing*. The presence of ferromagnetic order also modifies the fundamental process of Andreev reflection. The reason is that an incoming particle flips its spin under Andreev reflection, thus switching between the two spin bands $N_\uparrow(\epsilon)$ and $N_\downarrow(\epsilon)$ in the process. In strong ferromagnets, the difference between the spin bands can be large enough to suppress Andreev processes, resulting in a very different behaviour from normal metals.[73] In the extreme case of half-metals, which are 100% spin-polarized ferromagnets, conventional Andreev reflections are entirely forbidden.[75]

Note that the triplet pairs $|\uparrow\downarrow\rangle + |\downarrow\uparrow\rangle$ which are generated by the proximity effect in a ferromagnet are what we call *short-range triplet components*, which decay over the typically short distance scale ξ_F in the ferromagnet. For practical applications, generating the *long-range triplet components* $|\uparrow\uparrow\rangle$ and $|\downarrow\downarrow\rangle$ is a lot more interesting. These pairs consist of particles from the same spin band, and decay over the much longer distance scale ξ_T, making applications in mesoscopic spintronic circuits possible. They can be produced by a *spin-rotation* of short-range triplet pairs, which may be caused by either inhomogeneous magnetic fields,[16,17] or materials with spin-orbit coupling,[18,19] where the latter is the subject that will be further investigated in this thesis. One common feature for both the short-range and long-range triplet components, is that their spectroscopic signature is a zero-energy peak in the density of states (see figs. 2.5 and 2.6), which follows directly from eq. (2.60).[76]



(a) Singlet-dominated ($L_F = 0.19\xi_S$); (b) Triplet-dominated ($L_F = 0.23\xi_S$).

FIGURE 2.5: Proximity effect in a ferromagnet with the exchange field $h = 3\Delta_0$. The density of states exhibits either an induced minigap or a zero-energy peak, depending on whether the singlet or triplet pairs dominate in eq. (2.60).

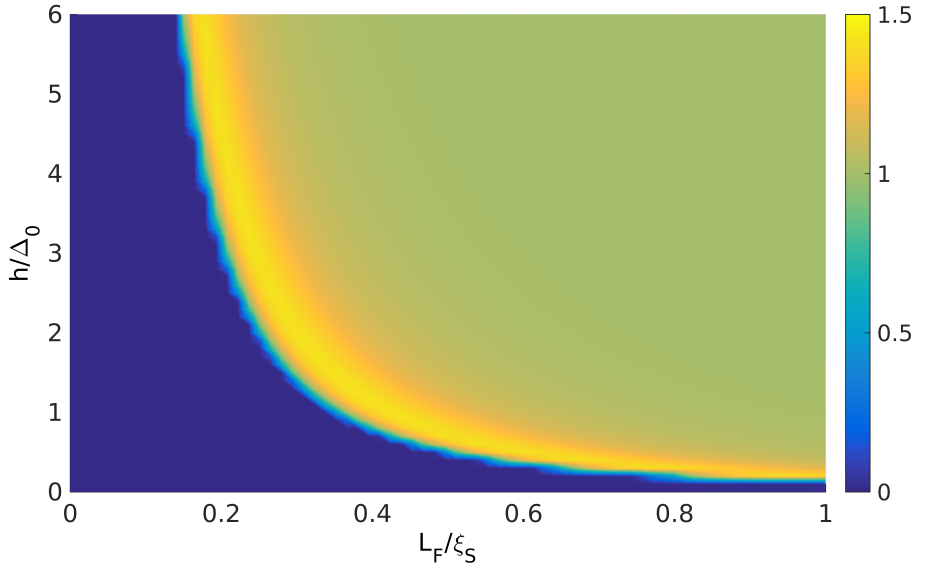
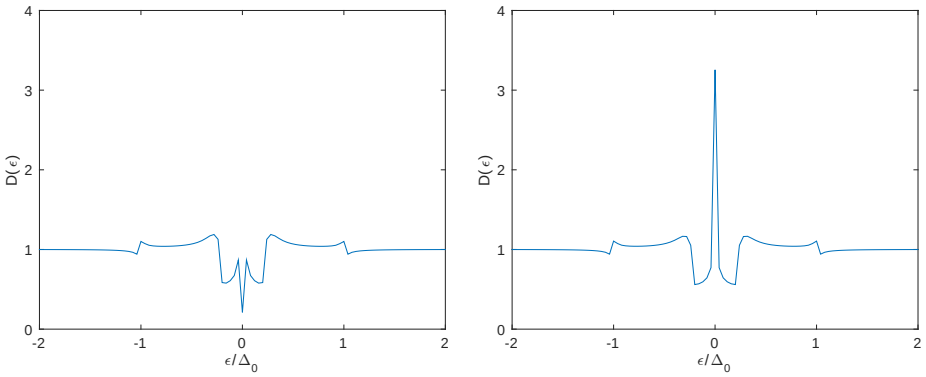


FIGURE 2.6: Zero-energy density $D(0)$ as a function of the ferromagnet length L_F and exchange field strength h. For sufficiently large parameters, there is a sudden onset of triplet pairing in the ferromagnet, resulting in a large zero-energy peak. As the system parameters increases past this point, the density of states gradually flattens and converges towards that of a normal metal.

Another way to generate triplet superconductivity is to exploit materials with *spin-active interfaces*, which typically take the form of ferromagnetic insulators a few nanometers thick.[43,54–56] The spin-mixing is then caused by the spin-dependent interfacial phase shifts, which produces a sudden onset of triplet superconductivity when $\lambda = \pm 1$ (see figs. 2.7 and 2.8).[77,78] For practical applications, spin-active interfaces between superconductors and ferromagnets are especially interesting, since the combination can be used to generate long-range triplets if the ferromagnetic insulator and conductor are noncollinearly magnetized.



(a) Singlet-dominated ($\lambda = 0.95$); (b) Triplet-dominated ($\lambda = 1.05$).

FIGURE 2.7: Proximity effect in a normal metal with spin-active interface ($P = 0$). There is either a minigap or a zero-energy peak, depending on the value of λ.

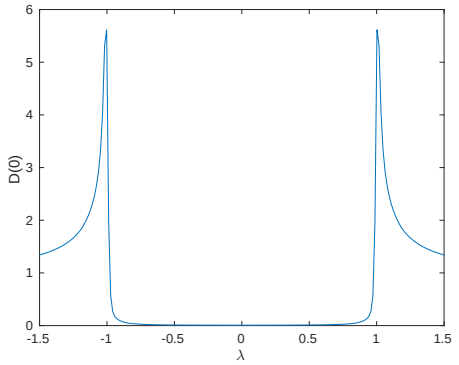


FIGURE 2.8: Zero-energy density $D(0)$ as a function of λ. The superconductor induces a minigap in the normal metal, which abruptly transforms into a peak for $\lambda = \pm 1$. For larger values of λ, the peak gradually flattens to $D(0) = 1$.

An especially interesting scenario occurs when two superconductors couple through a region of normal metal, forming a *Josephson junction*.[79] Cooper pairs can tunnel between the two superconductors, and due to interference effects, the dynamics of the junction depends strongly on the phase difference $\phi \equiv \varphi_R - \varphi_L$ between the Cooper pairs in the right and left superconductor.[20,21] These effects can also be understood in terms of multiple Andreev reflections.[70] When a hole in the normal metal reflects off the interface to a superconductor as an electron, it annihilates one of the Cooper pairs in the superconductor. The electron then moves throughout the normal metal, and reflects off the second superconductor as a hole, thus producing a new Cooper pair in that superconductor. Through this process of successive Andreev reflections, the net effect is that a Cooper pair has been transmitted in a phase-coherent manner through a region of normal metal by an intermediary electron/hole loop. Depending on the phases of the two superconductors, the resulting interference can be either constructive or destructive, which manifests as an enhancement or suppression of the minigap in the normal metal (see fig. 2.9).

If the interfaces are opaque enough to be treated in the tunneling limit,[d] and no magnetic interfaces or materials are involved, then the equations that govern the behaviour of the Josephson junction are

$$I = I_c \sin\phi\,, \qquad \frac{d\phi}{dt} = 2eV\,. \tag{2.64}$$

These equations show that applying a constant voltage V over the junction causes the phase difference to vary linearly with time, and this produces a sinusoidally varying current I. So a conventional Josephson junction is a perfect voltage-to-frequency converter. We also note that the current through the junction is maximized for $\phi = \pm\pi/2$, in which case it reaches the critical current I_c. By integrating the electric power $P = VI$ over time, we find the energy of the junction to be

$$E = \frac{I_c}{2e}(1 - \cos\phi)\,, \tag{2.65}$$

which shows that energy can be stored in the junction by adjusting the phase difference ϕ. In all conventional junctions, the critical current I_c is a positive quantity, so the ground state has to be $\phi = 0$. These systems are therefore also referred to as 0-junctions.

[d] For a discussion of high-transparency Josephson junctions, see Nazarov & Blanter.[42]

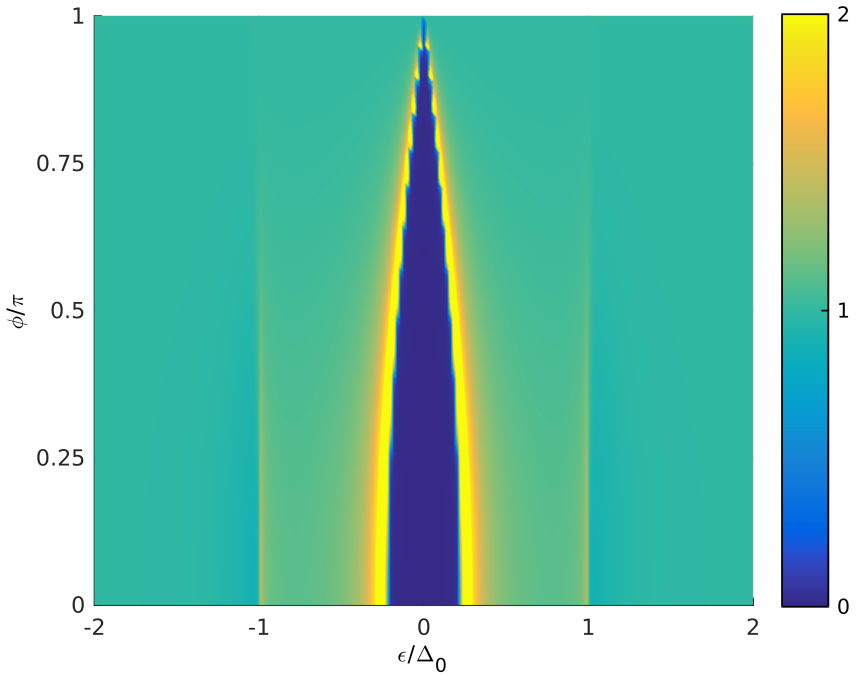


FIGURE 2.9: Density of states $D(\epsilon)$ for a conventional Josephson junction as a function of the phase difference ϕ. The minigap shrinks as the phase difference increases, and when it reaches $\phi = \pi$, the density of states becomes entirely flat.

In Josephson junctions with magnetic elements, a whole range of new effects are possible. First of all, the critical current I_c can become negative due to the spatial modulation of the superconducting wave function. This shifts the ground state to $\phi = \pm\pi$, resulting in a so-called π-junction.[69,70] Furthermore, by combining 0-junctions and π-junctions, it is also possible to create φ-junctions with degenerate ground states $\phi = \pm\varphi$.[74] Moreover, by exploiting materials with the right spin-orbit coupling, it should be possible to create φ_0-junctions, which would have a single but arbitrary ground state $\phi = \varphi_0$.[80,81] Both π-junctions and ϕ-junctions have already been fabricated,[82,83] while the φ_0-junction remains a prediction for now. Another recent prediction is that spin-orbit coupled Josephson junctions can have a significant triplet proximity effect, even when $\phi = \pi$.[84] In other words, Josephson junctions with ferromagnetic and spin-orbit coupled components presents a rich set of novel physical phenomena, which may have applications in both digital and quantum computing.[85]

In addition to density of states and electrical currents, superconducting hybrid systems exhibit another key observable that should be addressed: the critical temperature. In conventional bulk superconductors, the gap Δ starts at a material-dependent maximum Δ_0 at absolute zero, and then gradually decreases as the temperature of the system is increased. When the temperature reaches a critical value T_{cs}, then the system undergoes a phase transition to a non-superconductor, where the gap Δ is the order parameter of the transition.[20–22] However, this process is modified in mesoscopic hybrid systems. In figs. 2.3 and 2.4, it was shown that the inverse proximity effect leads to the induction of subgap states in a superconductor near the interface. This leads to a reduction of the superconducting gap at all temperatures, which in turn shifts the phase transition to a lower temperature $T_c < T_{cs}$. The magnitude of this effect is strongly dependent on the composition of the hybrid system, and is usually maximized when the superconducting material is small (fig. 2.10), when the adjoining material is large (fig. 2.11), when the interface transparency is high, and in the presence of magnetic fields. In some extreme cases, the critical temperature can even drop to $T_c = 0$, in which case the superconducting properties are destroyed regardless of the temperature.

These effects have led to the suggestion of a *superconducting spin-valve*, which consists of a superconductor interlocked between homogeneous ferromagnets.[86,87] When the magnets have a parallel field configuration, the net magnetic field is quite strong, while in the antiparallel configuration, the net magnetic field is zero. Since the two scenarios have vastly different effects on the superconductor inbetween, the critical temperature of the system can have a strong dependence on the relative magnetization angle. When operated in the correct temperature range, this permits the superconductivity of the system to be toggled on and off by a rotation of the magnetic field, thus providing a transistor-like switching effect based on a magnetic input signal. Several such devices have already been manufactured by experimentalists,[88–93] and may have important applications in the emerging field of superconducting spintronics.

We emphasize that critical temperature calculations always require a self-consistent approach, thus making it a much more time-consuming task than density of states calculations. However, the computation time can be reduced by orders of magnitude when using a suitable algorithm. Such an algorithm based on the principle of binary search is suggested in appendix A, and an implementation in MATLAB is enclosed at the back.

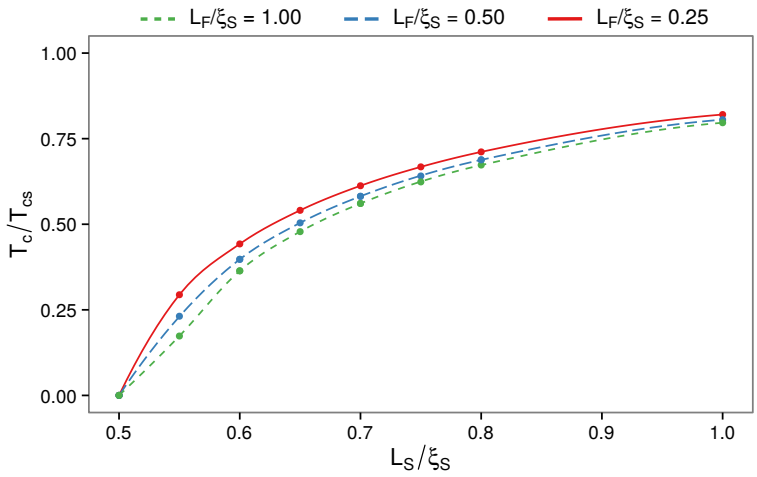


FIGURE 2.10: Critical temperature of a superconductor/ferromagnet bilayer as a function of the length L_S of the superconductor. The adjoining ferromagnet has three different lengths L_F and an exhange field $h = 10\Delta_0$. Note that the superconductivity vanishes entirely for $L_S \cong 0.5\xi_S$, but reaches nearly 50% of the bulk value already at $L_S \cong 0.6\xi_S$, indicating that the system is very sensitive to parameter changes in this region.

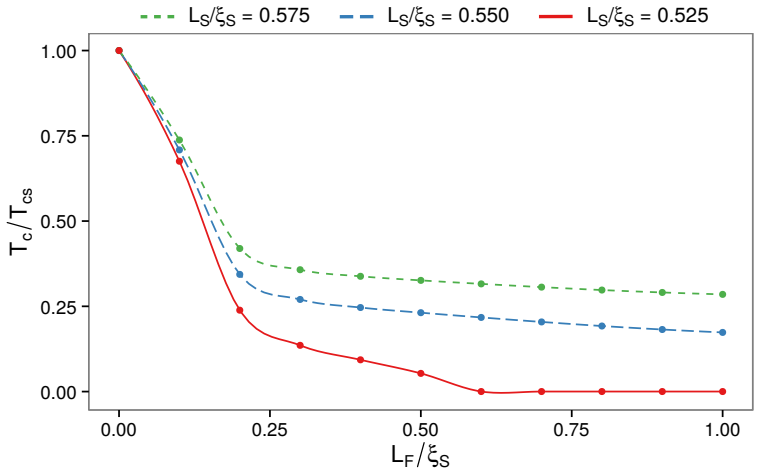


FIGURE 2.11: Critical temperature of a superconductor/ferromagnet bilayer as a function of the length L_F of the ferromagnet. The exchange field is $h = 10\Delta$, and the superconductor has three different lengths L_S. For $L_S = 0.525\xi_S$, the critical temperature drops to zero for $L_F \cong 0.6\xi_S$. No indications of nonmonotonic behaviour are observed; this is consistent with the results of Fominov et al., who only reported such behaviour for systems where either the interface parameter ζ or exchange field h is much smaller than for the systems considered here.[94]

3

SUPERCONDUCTING GAP

In this chapter, we derive a self-consistency equation for the gap $\Delta(\mathbf{x})$, which will enable the gap to be calculated from the Green's functions of the material. We will presume a conventional BCS superconductor with an s-wave symmetry. Applying the results to a bulk superconductor then produces self-consistency conditions that relate the Debye frequency ω_c, zero-temperature gap Δ_0, and critical temperature T_c of the material. These results are finally substituted back into the equations from whence they came, which reduces the number of independent parameters in the equations. The derivation is initially restricted to the Keldysh formalism, but the results are later translated to the Matsubara formalism as well.

3.1 KELDYSH FORMALISM

The starting point for deriving the gap equation is the Keldysh Green's function, and specifically the anomalous component defined in eq. (2.17). By evaluating this function at opposite spins but coinciding spacetime coordinates, and then rewriting commutators according to the fermionic anticommutation relations, we obtain the equations

$$F^K_{\uparrow\downarrow}(\mathbf{x}, t; \mathbf{x}, t) = -2i\langle\psi_\uparrow(\mathbf{x}, t)\psi_\downarrow(\mathbf{x}, t)\rangle\,, \tag{3.1}$$

$$F^K_{\downarrow\uparrow}(\mathbf{x}, t; \mathbf{x}, t) = +2i\langle\psi_\uparrow(\mathbf{x}, t)\psi_\downarrow(\mathbf{x}, t)\rangle\,. \tag{3.2}$$

The superconducting gap was defined as $\Delta(\mathbf{x}, t) \equiv \lambda\langle\psi_\uparrow(\mathbf{x}, t)\psi_\downarrow(\mathbf{x}, t)\rangle$, so the above immediately suggests two different ways to calculate the gap,

$$\Delta(\mathbf{x}, t) = +\frac{i\lambda}{2}F^K_{\uparrow\downarrow}(\mathbf{x}, t; \mathbf{x}, t)\,, \tag{3.3}$$

$$\Delta(\mathbf{x}, t) = -\frac{i\lambda}{2}F^K_{\downarrow\uparrow}(\mathbf{x}, t; \mathbf{x}, t)\,. \tag{3.4}$$

Note that the only assumption we have made so far, is that the material is a conventional BCS superconductor, i.e. that it is described by eq. (2.1). Insofar as this assumption is satisfied, the gap equations above are exact.

Now that we know the superconducting gap as a function of the exact Keldysh Green's function $\underline{F}^K$, the next step on the agenda is to derive a

similar equation for the quasiclassical version $\underline{f}^K$. First, we introduce the Wigner mixed coordinates,

$$\underline{F}^K(\mathbf{x}, t; \mathbf{x}, t) = \lim_{\zeta\to 0}\lim_{\tau\to 0} \underline{F}^K(\mathbf{x}+\zeta/2, t+\tau/2; \mathbf{x}-\zeta/2, t-\tau/2)\,, \tag{3.5}$$

where the limits ensure that the relative coordinates ζ and τ are set to zero, which is necessary because the absolute coordinates were coinciding. We then proceed to Fourier transform the relative coordinates,

$$\underline{F}^K(\mathbf{x}, t; \mathbf{x}, t) = V \lim_{\zeta\to 0}\lim_{\tau\to 0} \int \frac{d\epsilon}{(2\pi)} \int \frac{d^3\mathbf{p}}{(2\pi)^3} e^{i(\mathbf{p}\cdot\zeta - \epsilon\tau)} \underline{F}^K(\mathbf{x}, \mathbf{p}, t, \epsilon)\,, \tag{3.6}$$

where V is the volume of the system. Evaluating the limits, we then get

$$\underline{F}^K(\mathbf{x}, t; \mathbf{x}, t) = V \int \frac{d\epsilon}{(2\pi)} \int \frac{d^3\mathbf{p}}{(2\pi)^3}\, \underline{F}^K(\mathbf{x}, \mathbf{p}, t, \epsilon)\,. \tag{3.7}$$

The momentum integral can be parametrized using spherical coordinates,

$$\underline{F}^K(\mathbf{x}, t; \mathbf{x}, t) = \frac{V}{4\pi^3} \int d\epsilon \int dp\, p^2 \int \frac{d\Omega}{4\pi}\, \underline{F}^K(\mathbf{x}, \mathbf{p}, t, \epsilon)\,, \tag{3.8}$$

where the last integral is an angular average over momentum directions,

$$\underline{F}^K(\mathbf{x}, t; \mathbf{x}, t) = \frac{V}{4\pi^3} \int d\epsilon \int dp\, p^2 \left\langle \underline{F}^K(\mathbf{x}, \mathbf{p}, t, \epsilon)\right\rangle\,. \tag{3.9}$$

The kinetic energy of the quasiparticles is defined as $\xi \equiv \mathbf{p}^2/2m - \mu$, where μ is the Fermi energy, so this may also be written as a ξ-integration,

$$\underline{F}^K(\mathbf{x}, t; \mathbf{x}, t) = \frac{V}{4\pi^3} \int d\epsilon \int d\xi\, \sqrt{2m^3(\xi+\mu)}\, \left\langle \underline{F}^K(\mathbf{x}, \mathbf{p}, t, \epsilon)\right\rangle\,. \tag{3.10}$$

Under the quasiclassical approximation, it is assumed that the excitation energy ξ is much smaller than the Fermi energy μ, so the above becomes

$$\underline{F}^K(\mathbf{x}, t; \mathbf{x}, t) \cong \frac{V}{4\pi^3}\sqrt{2\mu m^3} \int d\epsilon \int d\xi\, \left\langle \underline{F}^K(\mathbf{x}, \mathbf{p}, t, \epsilon)\right\rangle\,, \tag{3.11}$$

or written in terms of the density of states per spin N_0 at the Fermi level,

$$\underline{F}^K(\mathbf{x}, t; \mathbf{x}, t) \cong \frac{N_0}{2\pi} \int d\epsilon \int d\xi\, \left\langle \underline{F}^K(\mathbf{x}, \mathbf{p}, t, \epsilon)\right\rangle\,. \tag{3.12}$$

Substituting in eq. (2.22), we finally obtain a direct relationship between the exact function $\underline{F}^K(\mathbf{x}, t; \mathbf{x}, t)$ and its quasiclassical counterpart $\underline{f}^K(\mathbf{x}, t, \epsilon)$,

$$\underline{F}^K(\mathbf{r}, t; \mathbf{r}, t) \cong -\frac{i}{2} N_0 \int d\epsilon\, f^K(\mathbf{x}, t, \epsilon)\,. \tag{3.13}$$

Substituting this into eqs. (3.3) and (3.4), we get the quasiclassical results

$$\Delta(x,t) = +\frac{1}{4}N_0\lambda\int d\epsilon\, f^K_{\uparrow\downarrow}(x,t,\epsilon)\,, \tag{3.14}$$

$$\Delta(x,t) = -\frac{1}{4}N_0\lambda\int d\epsilon\, f^K_{\downarrow\uparrow}(x,t,\epsilon)\,. \tag{3.15}$$

For reasons that will become apparent later, it will prove prudent to average the two equations above, such that we obtain the single equation:

$$\Delta(x,t) = \frac{1}{8}N_0\lambda\int d\epsilon\left[f^K_{\uparrow\downarrow}(x,t,\epsilon) - f^K_{\downarrow\uparrow}(x,t,\epsilon)\right]. \tag{3.16}$$

So far, we have assumed a singlet superconductor in the quasiclassical regime—but we have not yet made any mention of thermal equilibrium. Thus, the gap equation above is equally valid for non-equilibrium systems.

We will now restrict our attention to thermal equilibrium, in which case eqs. (2.23) and (2.24) provide a direct relationship between the Keldysh Green's function $\underline{f}^K$ and the retarded Green's function $\underline{f}^R$,

$$\underline{f}^K(x,\epsilon) = \left[\underline{f}^R(x,\epsilon) + \underline{f}^{RT}(x,-\epsilon)\right]\tanh(\epsilon/2T)\,. \tag{3.17}$$

We begin by extracting the antidiagonal elements of the matrix equation,

$$f^K_{\uparrow\downarrow}(x,\epsilon) = \left[f^R_{\uparrow\downarrow}(x,\epsilon) + f^R_{\downarrow\uparrow}(x,-\epsilon)\right]\tanh(\epsilon/2T)\,, \tag{3.18}$$

$$f^K_{\downarrow\uparrow}(x,\epsilon) = \left[f^R_{\downarrow\uparrow}(x,\epsilon) + f^R_{\uparrow\downarrow}(x,-\epsilon)\right]\tanh(\epsilon/2T)\,, \tag{3.19}$$

and then perform a singlet/triplet decomposition according to eq. (2.32),

$$f^K_{\uparrow\downarrow}(x,\epsilon) = \left[f^R_z(x,\epsilon) + f^R_s(x,\epsilon) + f^R_z(x,-\epsilon) - f^R_s(x,-\epsilon)\right]\tanh(\epsilon/2T)\,, \tag{3.20}$$

$$f^K_{\downarrow\uparrow}(x,\epsilon) = \left[f^R_z(x,\epsilon) - f^R_s(x,\epsilon) + f^R_z(x,-\epsilon) + f^R_s(x,-\epsilon)\right]\tanh(\epsilon/2T)\,. \tag{3.21}$$

When we substitute these identities into eq. (3.16), the triplet components cancel, and we are therefore left with an equation that depends solely on the singlet component of the retarded Green's function,

$$\Delta(x) = \frac{1}{4}N_0\lambda\int d\epsilon\left[f^R_s(x,\epsilon) - f^R_s(x,-\epsilon)\right]\tanh(\epsilon/2T)\,. \tag{3.22}$$

If the integral above is performed for all real values of ϵ, it turns out to be logarithmically divergent for a bulk superconductor. However, physically, the energy spectrum that should be considered is restricted by the energy spectra of the phonons that mediate the attractive electron–electron interaction in the superconductor. More precisely, the attractive interaction λ

only exists in a limited regime $(-\omega_c, \omega_c)$ for the energy transfer in the electron–electron scattering process, where the cutoff ω_c is the Debye frequency. We therefore restrict the integration domain to this range,

$$\Delta(\mathbf{x}) = \frac{1}{4} N_0 \lambda \int_{-\omega_c}^{\omega_c} d\epsilon \left[f_s^R(\mathbf{x}, \epsilon) - f_s^R(\mathbf{x}, -\epsilon) \right] \tanh(\epsilon/2T) \,. \tag{3.23}$$

Note that both $\tanh(\epsilon/2T)$ and $f_s^R(\mathbf{x}, \epsilon) - f_s^R(\mathbf{x}, -\epsilon)$ are odd functions of ϵ, such that their product has to be an even function of ϵ. Exploiting this symmetry, we can constrain the domain of integration to positive energies:

$$\Delta(\mathbf{x}) = \frac{1}{2} N_0 \lambda \int_{0}^{\omega_c} d\epsilon \left[f_s^R(\mathbf{x}, \epsilon) - f_s^R(\mathbf{x}, -\epsilon) \right] \tanh(\epsilon/2T) \,. \tag{3.24}$$

We have not yet made the assumption that $\Delta(\mathbf{x})$ is real, so eq. (3.24) holds when the superconductor has a phase as well. Even though the integral is only taken over positive values of ϵ, because of the term $f_s^R(\mathbf{x}, -\epsilon)$, we still need to know the Green's function at negative energies to calculate the gap. This nuisance can be alleviated by exploiting another symmetry,

$$f_s^R(\mathbf{x}, \epsilon) = -f_s^{R*}(\mathbf{x}, -\epsilon^*) \,, \tag{3.25}$$

which holds in the gauge where $\Delta(\mathbf{x})$ is real. Substituted into eq. (3.24), we obtain a particularly convenient version of the gap equation, which only depends on the real part of the singlet component $f_s^R(\mathbf{x}, \epsilon)$ for $\epsilon \geqslant 0$:

$$\Delta(\mathbf{x}) = N_0 \lambda \int_{0}^{\omega_c} d\epsilon \, \mathrm{Re}\left\{ f_s^R(\mathbf{x}, \epsilon) \right\} \tanh(\epsilon/2T) \,. \tag{3.26}$$

Now that a suitable self-consistency equation is available, the next step is to apply it to a bulk superconductor. In this kind of system, the Usadel equation has the exact solution

$$\underline{\hat{g}}_{\mathrm{BCS}}^R(\theta) = \begin{pmatrix} \cosh(\theta)\underline{\sigma}_0 & \sinh(\theta) i\underline{\sigma}_y \\ \sinh(\theta) i\underline{\sigma}_y & -\cosh(\theta)\underline{\sigma}_0 \end{pmatrix}, \tag{3.27}$$

which has been parametrized by $\theta(\epsilon) \equiv \mathrm{atanh}(\Delta/\epsilon)$.[35–37] Substituting the top-right block $\underline{f}^R = \sinh(\theta) i\underline{\sigma}_y$ into eq. (2.32), solving for f_s^R, and using $\sinh(\mathrm{atanh}(u)) = u/\sqrt{1-u^2}$ to untangle the result, we for $\epsilon > 0$ find that

$$f_s^R(\epsilon) = \frac{\Delta}{\sqrt{\epsilon^2 - \Delta^2}} \,. \tag{3.28}$$

Note that this function is purely real for $|\epsilon| \geqslant |\Delta|$, and purely imaginary otherwise, which means that the real part is $\mathrm{Re}\{f_s^R(\epsilon)\} = f_s^R(\epsilon)\theta(|\epsilon| - |\Delta|)$. By substituting this into eq. (3.26), and dividing the result by $\Delta N_0 \lambda$, we obtain the self-consistency equation for the gap of a bulk superconductor,

$$\frac{1}{N_0\lambda} = \int_{\Delta}^{\omega_c} d\epsilon \, \frac{\tanh(\epsilon/2T)}{\sqrt{\epsilon^2 - \Delta^2}} \,. \tag{3.29}$$

Let us consider the zero-temperature limit. The gap approaches some finite upper bound $\Delta \to \Delta_0$ as $T \to 0$, and the hyperbolic tangent is eliminated by the convergence $\tanh(\epsilon/2T) \to \mathrm{sgn}(\epsilon)$, so the equation becomes

$$\frac{1}{N_0\lambda} = \int_{\Delta_0}^{\omega_c} \frac{d\epsilon}{\sqrt{\epsilon^2 - \Delta_0^2}} \,. \tag{3.30}$$

The integral can be solved by the substitution $\epsilon = \Delta_0 \cosh u$, followed by some standard hyperbolic identities. Reordering the results to isolate the Debye frequency ω_c, we then obtain an equation that can be used to calculate it from the zero-temperature gap Δ_0:

$$\omega_c = \Delta_0 \cosh(1/N_0\lambda) \,. \tag{3.31}$$

Hence there is one less parameter that has to be measured experimentally.

We then turn to the critical temperature T_c of the bulk superconductor, which is the temperature where the gap vanishes. Going back to eq. (3.29), and letting $T \to T_c$ and $\Delta \to 0$, we get

$$\frac{1}{N_0\lambda} = \int_{0}^{\omega_c} d\epsilon \, \frac{\tanh(\epsilon/2T_c)}{\epsilon} \,, \tag{3.32}$$

which can be cast in a more convenient form by an integration by parts,[21]

$$\frac{1}{N_0\lambda} = \tanh(\omega_c/2T_c)\ln(\omega_c/2T_c) - \frac{1}{2T_c}\int_{0}^{\omega_c} d\epsilon \, \frac{\ln(\epsilon/2T_c)}{\cosh^2(\epsilon/2T_c)} \,. \tag{3.33}$$

In the weak-coupling limit $\omega_c \gg T_c$, the factor $\tanh(\omega_c/2T_c)$ converges to unity, and it follows that the first term can be approximated by $\ln(\omega_c/2T_c)$. As for the second term, the integral converges rapidly to $\ln(\pi/4e^\gamma)$, where $\gamma \cong 0.57722$ is the *Euler–Mascheroni constant*. Ergo, the self-consistency equation above may be approximated as

$$\frac{1}{N_0\lambda} = \ln(\omega_c/2T_c) - \ln(\pi/4e^\gamma) \,, \tag{3.34}$$

and solving for the Debye frequency ω_c we obtain:

$$\omega_c = \frac{\pi T_c}{2e^\gamma} \exp(1/N_0\lambda)\,. \tag{3.35}$$

This provides an alternative to eq. (3.31) by enabling the calculation of the appropriate cutoff from the critical temperature T_c instead of the gap Δ_0. Since we assumed the weak-coupling limit $\omega_c \gg T_c$, the above suggests that $N_0\lambda \ll 1$, in which case $\exp(1/N_0\lambda) \cong 2\cosh(1/N_0\lambda)$. Note that the accuracy of this approximation varies drastically, even among typical low-temperature superconductors; for instance, the deviation is less than 0.02% for beryllium ($N_0\lambda \cong 0.23$), about 0.5% for aluminum ($N_0\lambda \cong 0.38$), and almost 9% for niobium ($N_0\lambda \cong 0.82$).[95] There also exist strong-coupled superconductors with $N_0\lambda > 1$, such as the niobium alloys Nb_3Al ($N_0\lambda \cong 1.7$) and Nb_3Sn ($N_0\lambda \cong 1.8$),[96] for which the approximation would be off by more than 30%. Some caution should therefore be exercised before making the assumption $N_0\lambda \ll 1$. However, assuming for the time being that it is satisfied, we may combine eqs. (3.31) and (3.35) to obtain the well-known BCS ratio between the zero-temperature gap Δ_0 and critical temperature T_c:[21,22]

$$\frac{\Delta_0}{T_c} = \frac{\pi}{e^\gamma}\,. \tag{3.36}$$

With eqs. (3.31) and (3.36) in our repertoire, measuring $N_0\lambda$ and either the Debye frequency ω_c, zero-temperature gap Δ_0, or critical temperature T_c is sufficient to determine the rest of these parameters—thus halving the number of independent parameters in the model. Substitution of these results into eq. (3.26) produces the improved self-consistency equation:

$$\Delta(\boldsymbol{x}) = N_0\lambda \int_0^{\Delta_0 \cosh(1/N_0\lambda)} d\epsilon\, \mathrm{Re}\big\{f_s^R(\boldsymbol{x},\epsilon)\big\} \tanh\left(\frac{\pi}{2e^\gamma}\frac{\epsilon/\Delta_0}{T/T_c}\right). \tag{3.37}$$

This equation is particularly well-suited for numerical simulations. One advantage is that we measure all energies relative to the zero-temperature gap Δ_0, and temperatures relative to the critical temperature T_c. The calculations may therefore be executed in a material-agnotic way, and then adapted to a number of physical materials later by tuning Δ_0 and T_c to their empirical values. The only physical parameter that we need to know at the time of the calculation, is the dimensionless coupling constant $N_0\lambda$, which measures the relative significance of BCS interactions in a material.

We also emphasize that the Green's function is evaluated only at positive energies, which entails that it is sufficient to solve the Usadel equation for $0 \leqslant \epsilon \leqslant \Delta_0 \cosh(1/N_0\lambda)$ to determine the gap. Finally, the fact that the integrand is real can be convenient implementation-wise, as not all numerical integration/interpolation routines support complex functions. We emphasize that this result is only valid for quasiclassical isotropic BCS superconductors in equilibrium in a real gauge.

3.2 MATSUBARA FORMALISM

We will use eq. (3.23) as a starting point for deriving a gap equation in the Matsubara formalism. Closing the integration contour along a half-circle in the upper complex half-plane as shown in fig. 3.1, but neglecting the integral along the half-circle in the weak coupling limit $\omega_c \gg T_c$, we get

$$\Delta(\mathbf{x}) = \frac{1}{4} N_0 \lambda \oint_\Gamma d\epsilon \left[f_s^R(\mathbf{x}, \epsilon) - f_s^R(\mathbf{x}, -\epsilon) \right] \tanh(\epsilon/2T) \,. \tag{3.38}$$

Since the hyperbolic tangent has poles at the imaginary energies $\epsilon = i\omega_n$, where $\omega_n \equiv (2n+1)\pi T$ are the Matsubara frequencies, we can recast the integral as a sum using Cauchy's residue theorem,

$$\Delta(\mathbf{x}) = \pi i T N_0 \lambda \sum_{n=0}^{n_c} \left[f_s^R(\mathbf{x}, i\omega_n) - f_s^R(\mathbf{x}, -i\omega_n) \right] , \tag{3.39}$$

where $n_c \cong \omega_c / 2\pi T$ is the index of the largest Matsubara frequency ω_n below the Debye frequency ω_c. However, this result is a bit unsettling: eq. (2.31) only relates the retarded functions to the Matsubara functions at positive Matsubara frequencies, while the second term above is evaluated at negative frequencies. The solution to this problem is to select different integration contours for the two terms in eq. (3.23),

$$\Delta(\mathbf{x}) = \frac{1}{4} N_0 \lambda \left\{ \oint_\Gamma d\epsilon \, f_s^R(\mathbf{x}, \epsilon) \tanh(\epsilon/2T) - \oint_{\Gamma'} d\epsilon \, f_s^R(\mathbf{x}, -\epsilon) \tanh(\epsilon/2T) \right\} , \tag{3.40}$$

where the new contour Γ' consists of a half-circle in the lower complex half-plane as shown in fig. 3.2. The first term in eq. (3.40) is equivalent to the first term in eq. (3.38), but the second term has two new properties. Firstly, since the contour Γ' has negative orientation, we have to remember to include an extra minus sign upon invocation of the residue theorem. Secondly, since the poles enclosed by Γ' are located at $\epsilon = -i\omega_n$, we

have $f_s^R(x, -\epsilon) = f_s^R(x, +i\omega_n)$, which is evaluated at positive Matsubara frequencies. Thus, applying the residue theorem to eq. (3.40) produces

$$\Delta(x) = 2\pi i T N_0 \lambda \sum_{n=0}^{n_c} f_s^R(x, i\omega_n)\,. \tag{3.41}$$

In conjunction with eq. (2.31), the above can be rewritten as an equation for the superconducting gap in terms of the Matsubara Green's function:

$$\Delta(x) = 2\pi T N_0 \lambda \sum_{n=0}^{n_c} f_s^M(x, \omega_n)\,. \tag{3.42}$$

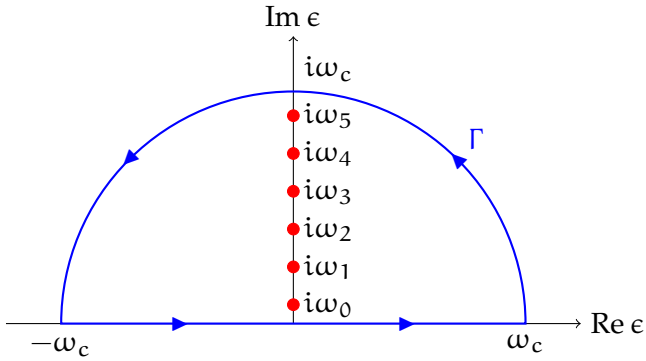


FIGURE 3.1: The integration contour Γ. The radius of the contour is the Debye frequency ω_c, and the indicated poles are the Matsubara frequencies $\epsilon = i\omega_n$.

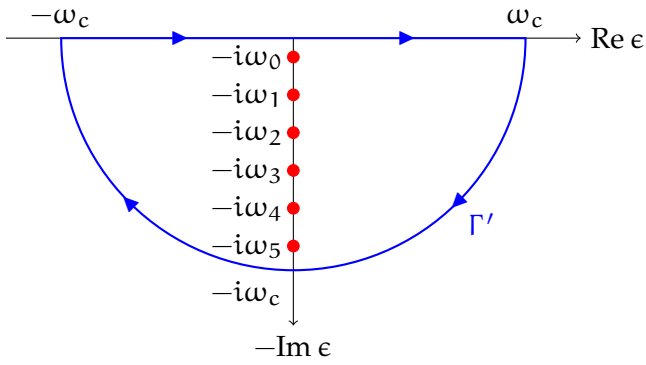


FIGURE 3.2: The integration contour Γ'. The radius of the contour is again ω_c, but the enclosed poles are now at the negative Matsubara frequencies $\epsilon = -i\omega_n$.

The Matsubara Green's function for a bulk superconductor can be found by substituting $\epsilon = i\omega_n$ and $if_s^R(i\omega_n) = f_s^M(\omega_n)$ into eq. (3.28),

$$f_s^M(\omega_n) = \frac{\Delta}{\sqrt{\omega_n^2 + \Delta^2}}\,. \tag{3.43}$$

Substituting the above into eq. (3.42), and subsequently dividing by $\Delta N_0\lambda$, we obtain a self-consistency equation for bulk superconductors,

$$\frac{1}{N_0\lambda} = \sum_{n=0}^{n_c} \frac{2\pi T}{\sqrt{\omega_n^2 + \Delta^2}}\,. \tag{3.44}$$

We will then consider the limit of critical temperature $T \to T_c$, which means that $\Delta \to 0$. Reinstating $\omega_n = (2n+1)\pi T$, the equation reduces to

$$\frac{1}{N_0\lambda} = \sum_{n=0}^{n_c} \frac{1}{n+1/2}\,, \tag{3.45}$$

which may be written in a closed form using the *digamma function* $\psi(\,\cdot\,)$,

$$\frac{1}{N_0\lambda} = \psi\left(n_c + \frac{1}{2}\right) - \psi\left(\frac{1}{2}\right). \tag{3.46}$$

This result was derived at the critical temperature T_c. However, the cutoff $n_c = \omega_c/2\pi T - 1/2$ is inversely proportional to temperature, so we may as well rewrite the result in a form that is valid at arbitrary temperatures:

$$\frac{1}{N_0\lambda} = \psi\left(\frac{T}{T_c}\left[n_c + \frac{1}{2}\right]\right) - \psi\left(\frac{1}{2}\right). \tag{3.47}$$

This result is the Matsubara equivalent to eq. (3.31), in that it provides a method of determining an appropriate cutoff n_c from the coupling $N_0\lambda$. Alternatively, the cutoff n_c may be calculated from the explicit equation

$$n_c = \frac{1}{4e^\gamma}\left(\frac{T_c}{T}\right)\exp\left(\frac{1}{N_0\lambda}\right) - \frac{1}{2}\,, \tag{3.48}$$

which follows by substitution of eq. (3.35) into $n_c = \omega_c/2\pi T - 1/2$.[a] This result shows that one in principle has to include infinitely many terms in the sum as $T \to 0$, regardless of the coupling strength $N_0\lambda$. On the other hand, as $T \to T_c$ fewer and fewer terms are required, and the Matsubara

[a] The calculated value of the cutoff n_c should of course be truncated to an integer, regardless of whether we used eq. (3.47) or (3.48) to calculate it.

formalism becomes increasingly efficient. This behaviour can intuitively be understood by consulting figs. 3.1 and 3.2; since the spacing between the Matsubara frequencies $\omega_n = (2n+1)\pi T$ increases with T, more and more frequencies will be pushed out of the integration contours as the temperature rises, resulting in fewer enclosed poles at high temperatures.

We will now use the results for a bulk superconductor to rewrite the non-bulk self-consistency equation in a more convenient form. First, we rewrite eq. (3.45) in the form

$$\frac{1}{N_0\lambda} \cong \sum_{n=0}^{n_c'} \frac{1}{n+1/2} - \int_{n_c}^{n_c'} \frac{dn}{n+1/2}\,, \tag{3.49}$$

where we have increased the summation cutoff from $n_c = \omega_c/2\pi T_c - 1/2$ to $n_c' = \omega_c/2\pi T - 1/2$, and approximated the difference as an integral. Rewriting the summand as $2\pi T/\omega_n'$, and evaluating the integral, we get

$$\frac{1}{N_0\lambda} = 2\pi T \sum_{n=0}^{n_c'} \frac{1}{\omega_n'} + \ln\left(\frac{T}{T_c}\right). \tag{3.50}$$

Finally, we multiply eq. (3.50) by $\Delta(x)$, divide eq. (3.42) by $N_0\lambda$, and then subtract the results to obtain the improved self-consistency equation:[47]

$$\Delta(x)\ln\left(\frac{T}{T_c}\right) = 2\pi T \sum_{n=0}^{n_c} \left(f_s^M(x,\omega_n) - \frac{\Delta(x)}{\omega_n} \right). \tag{3.51}$$

In the above, $\omega_n = (2n+1)\pi T$ and $\omega_c = (2n_c+1)\pi T$, i.e. the primes were dropped, as there should be no risk of confusion with $(2n+1)\pi T_c$ here. This form of the equation is particularly popular for critical temperature calculations in superconducting hybrid systems.[87,94] As a concluding remark, the above equation may also be solved explicitly for the gap $\Delta(x)$,

$$\Delta(x) = \frac{2e^\gamma (T/T_c)\Delta_0}{\ln(T/T_c) + \psi(n_c+1/2) - \psi(1/2)} \sum_{n=0}^{n_c} f_s^M(x,\omega_n)\,, \tag{3.52}$$

where we evaluated the sum over $2\pi T/\omega_n$ in terms of digamma functions, and assumed that the BCS coupling is weak enough for eq. (3.36) to hold. This result has the same form as eq. (3.42); the difference is that $N_0\lambda$ has been replaced by a function of the relative temperature T/T_c and cutoff n_c, where the latter abstracts all information about $N_0\lambda$ according to eq. (3.48).

4

WEAK PROXIMITY EFFECT

In this chapter, we consider layered structures with superconducting and magnetic elements, and include the effects of in-plane Rasha–Dresselhaus couplings. We linearize the diffusion equations of the system, and then separate the equations into singlet and triplet projections. The results are valid in two notable regimes. The first is near the critical temperature T_c, when superconductivity is suppressed in the entire structure. The second is when the interface transparency is low, in which case superconductivity is suppressed in the magnetic regions, while the superconducting regions may be approximated by BCS bulk solutions. As we will see in chapter 5, these equations also provide a very good qualitative explanation for the behaviour of the system in the strong proximity regime.

4.1 LINEARIZATION OF THE DIFFUSION EQUATION

While the Green's functions are $\underline{g} = 1$ and $\underline{f} = 0$ for normal metals, superconductivity is associated with the appearance of a nonzero anomalous component $\underline{f}$ in the system. The assumption of a *weak proximity effect* may therefore be translated to the mathematical statements $\underline{g} \cong 1$ and $\|\underline{f}\| \ll 1$, which justifies a linearization of the diffusion equation in this regime. In terms of the Riccati parametrization, this means that $\underline{N} \cong 1$ and $\|\underline{\gamma}\| \ll 1$. Neglecting all factors of $\underline{N}$ and $\underline{\tilde{N}}$ and all higher-order $\underline{\gamma}$ and $\underline{\tilde{\gamma}}$ terms in eq. (2.44), we obtain the linearized Usadel equation

$$\begin{aligned} D\,\partial_z^2\underline{\gamma} = &-2i\epsilon\underline{\gamma} - \Delta\underline{\sigma}_y - i\mathbf{h}\cdot(\underline{\boldsymbol{\sigma}}\underline{\gamma} - \underline{\gamma}\underline{\boldsymbol{\sigma}}^*) \\ &+ D\left[2\underline{\mathbf{A}}\underline{\gamma}\underline{\mathbf{A}}^* + \underline{\mathbf{A}}\,\underline{\mathbf{A}}\underline{\gamma} + \underline{\gamma}\underline{\mathbf{A}}^*\underline{\mathbf{A}}^*\right] \\ &+ 2iD\left[\underline{A}_z(\partial_z\underline{\gamma}) + (\partial_z\underline{\gamma})\underline{A}_z^*\right]. \end{aligned} \tag{4.1}$$

For simplicity, we focus on the Rashba–Dresselhaus coupling in eq. (2.10). Note that each product involving $\underline{\mathbf{A}}$ above should be interpreted as a dot product of the vector structure and matrix product of the matrix structure, which means that $\underline{\mathbf{A}}\underline{\gamma}\underline{\mathbf{A}}^* = \underline{A}_x\underline{\gamma}\,\underline{A}_x^* + \underline{A}_y\underline{\gamma}\,\underline{A}_y^*$ and $\underline{\mathbf{A}}^2 = \underline{A}_x^2 + \underline{A}_y^2$. Clearly $\underline{\mathbf{A}}\,\underline{\mathbf{A}} = \underline{\mathbf{A}}^*\underline{\mathbf{A}}^* = 2\alpha^2$ and $\underline{\mathbf{A}}_z = \underline{\mathbf{A}}_z^* = 0$, so eq. (4.1) simplifies to

$$D\,\partial_z^2\underline{\gamma} = -2i\epsilon\underline{\gamma} - \Delta\underline{\sigma}_y - i\mathbf{h}\cdot(\underline{\boldsymbol{\sigma}}\underline{\gamma} - \underline{\gamma}\underline{\boldsymbol{\sigma}}^*) + 4D\alpha^2\underline{\gamma} + 2D\underline{\mathbf{A}}\underline{\gamma}\underline{\mathbf{A}}^*. \tag{4.2}$$

In the weak proximity regime, the anomalous Green's function $\underline{f}^R$ and Riccati parameter $\underline{\gamma}$ are related by $\underline{f}^R = 2\underline{\gamma}$. Combining this with eq. (2.32), we get the singlet/triplet decomposition

$$\underline{\gamma} = \frac{1}{2}(f_s^R + \mathbf{f}_t^R \cdot \underline{\boldsymbol{\sigma}})i\underline{\sigma}_y\,, \tag{4.3}$$

from which we derive the trace identities

$$if_s^R = \mathrm{Tr}\{\underline{\gamma}\underline{\sigma}_y\}\,, \qquad if_t^R = \mathrm{Tr}\{\underline{\gamma}\underline{\sigma}_y\underline{\boldsymbol{\sigma}}\}\,, \tag{4.4}$$

$$i\mathbf{h}\cdot\mathbf{f}_t^R = \mathrm{Tr}\{(\mathbf{h}\cdot\underline{\boldsymbol{\sigma}})\underline{\gamma}\underline{\sigma}_y\}\,, \qquad i\mathbf{h}f_s^R = \mathrm{Tr}\{(\mathbf{h}\cdot\underline{\boldsymbol{\sigma}})\underline{\gamma}\underline{\sigma}_y\underline{\boldsymbol{\sigma}}\}\,, \tag{4.5}$$

$$-i\mathbf{h}\cdot\mathbf{f}_t^R = \mathrm{Tr}\{\underline{\gamma}(\mathbf{h}\cdot\underline{\boldsymbol{\sigma}}^*)\underline{\sigma}_y\}\,, \qquad -i\mathbf{h}f_s^R = \mathrm{Tr}\{\underline{\gamma}(\mathbf{h}\cdot\underline{\boldsymbol{\sigma}}^*)\underline{\sigma}_y\underline{\boldsymbol{\sigma}}\}\,. \tag{4.6}$$

To derive these relations, substitute eq. (4.3) into the expressions, and use the identities $\mathrm{Tr}\{a + \mathbf{b}\cdot\underline{\boldsymbol{\sigma}}\} = 2a$, $\mathrm{Tr}\{(a + \mathbf{b}\cdot\underline{\boldsymbol{\sigma}})\underline{\boldsymbol{\sigma}}\} = 2\mathbf{b}$, and $\underline{\sigma}_y\underline{\boldsymbol{\sigma}}^*\underline{\sigma}_y = -\underline{\boldsymbol{\sigma}}$. If we right-multiply eq. (4.2) by $\underline{\sigma}_y$ and $\underline{\sigma}_y\underline{\boldsymbol{\sigma}}$, respectively, and then use the list of identities above to evaluate their traces, we obtain the equations

$$\frac{i}{2}D\,\partial_z^2 f_s^R = \epsilon f_s^R + \mathbf{h}\cdot\mathbf{f}_t^R + 2iDa^2 f_s^R + D\,\mathrm{Tr}\{\underline{\mathbf{A}}\underline{\gamma}\underline{\mathbf{A}}^*\underline{\sigma}_y\} - \Delta\,, \tag{4.7}$$

$$\frac{i}{2}D\,\partial_z^2 \mathbf{f}_t^R = \epsilon\mathbf{f}_t^R + \mathbf{h}f_s^R + 2iDa^2\mathbf{f}_t^R + D\,\mathrm{Tr}\{\underline{\mathbf{A}}\underline{\gamma}\underline{\mathbf{A}}^*\underline{\sigma}_y\underline{\boldsymbol{\sigma}}\}\,. \tag{4.8}$$

As for the remaining trace terms, an explicit matrix calculation shows that

$$\underline{\mathbf{A}}\underline{\gamma}\underline{\mathbf{A}}^*\underline{\sigma}_y = -ia^2 f_s^R - ia^2(f_x^R\underline{\sigma}_y + f_y^R\underline{\sigma}_x)\sin 2\chi + ia^2 f_z^R\underline{\sigma}_z\,, \tag{4.9}$$

from which it follows directly that

$$D\,\mathrm{Tr}\{\underline{\mathbf{A}}\underline{\gamma}\underline{\mathbf{A}}^*\underline{\sigma}_y\} = -2iDa^2 f_s^R\,, \tag{4.10}$$

$$D\,\mathrm{Tr}\{\underline{\mathbf{A}}\underline{\gamma}\underline{\mathbf{A}}^*\underline{\sigma}_y\underline{\boldsymbol{\sigma}}\} = -2iDa^2(f_x^R\boldsymbol{e}_y + f_y^R\boldsymbol{e}_x)\sin 2\chi + 2iDa^2 f_z^R\boldsymbol{e}_z\,. \tag{4.11}$$

Substituting these results back into eqs. (4.7) and (4.8), we finally obtain the linearized Usadel equations for the components f_s^R and $\mathbf{f}_t^R$,

$$\frac{i}{2}D\,\partial_z^2 f_s^R = \epsilon f_s^R + \mathbf{h}\cdot\mathbf{f}_t^R - \Delta\,, \tag{4.12}$$

$$\frac{i}{2}D\,\partial_z^2 \mathbf{f}_t^R = \epsilon\mathbf{f}_t^R + \mathbf{h}f_s^R + 2iDa^2\boldsymbol{\Omega}(\chi)\mathbf{f}_t^R\,, \tag{4.13}$$

where we have defined the spin-orbit interaction matrix

$$\boldsymbol{\Omega}(\chi) \equiv \begin{pmatrix} 1 & -\sin 2\chi & 0 \\ -\sin 2\chi & 1 & 0 \\ 0 & 0 & 2 \end{pmatrix}. \tag{4.14}$$

Note that all the spin-orbit terms in the singlet equation cancelled, while the corresponding terms in the triplet equation could be neatly factored into a strength-dependent factor a^2 and a geometry-dependent factor $\mathbf{\Omega}(\chi)$. The interaction matrix $\mathbf{\Omega}(\chi)$ describes two different physical effects. The first kind is caused by the diagonal terms, which modifies the effective energy of the triplet component f_k^R to $E_k = \epsilon + 2iDa^2\Omega_{kk}$. Imaginary energies are usually associated with inelastic scatterings that destroy Cooper pairs, and are well-known to have a detrimental effect on superconductivity. The diagonal elements of $\mathbf{\Omega}(\chi)$ therefore *suppress* triplet superconductivity in a material. The second effect is caused by off-diagonal terms, which causes a mixing of the components f_x^R and f_y^R. If the exchange field $\mathbf{h}$ has a projection in the xy-plane, these terms may cause the short-range triplet component $f_\parallel^R$ along $\mathbf{h}$ to oscillate into a long-range triplet component $f_\perp^R$ perpendicular to $\mathbf{h}$, leading to a manifestation of long-range triplet superconductivity in the material. The off-diagonal elements may therefore *enhance* triplet superconductivity. These triplet mixing terms are proportional to $\sin 2\chi$, which suggests that the mixing will be strongest for $\chi = \pm\pi/4$, but vanishing for $\chi = 0$ and $\chi = \pm\pi/2$. In other words, as long as both the exchange field and spin-orbit field are confined to the xy-plane, a pure Rashba coupling or pure Dresselhaus coupling is insufficient to create any triplet mixing, while a Rashba and Dresselhaus coupling of equal magnitude maximizes the triplet mixing.

4.2 LINEARIZATION OF THE BOUNDARY CONDITIONS

We will now consider the Kuprianov–Lukichev boundary conditions in the weak proximity limit. Linearizing eqs. (2.52) and (2.53), we get

$$\partial_z \gamma_L = \frac{1}{L_L \zeta_L}(\gamma_R - \gamma_L) + i\underline{A}_z \gamma_L + i\gamma_L \underline{A}_z^*, \tag{4.15}$$

$$\partial_z \underline{\gamma}_R = \frac{1}{L_R \zeta_R}(\gamma_R - \underline{\gamma}_L) + i\underline{A}_z \gamma_R + i\underline{\gamma}_R \underline{A}_z^*. \tag{4.16}$$

Inserting $\underline{A}_z = 0$ and $\underline{f}^R = 2\underline{\gamma}$, performing a singlet/triplet decomposition like before, and assuming that the interface is located at $z = 0$, we get:

$$L_L \zeta_L \partial_z f_s^R(0^-) = L_R \zeta_R \partial_z f_s^R(0^+) = f_s^R(0^+) - f_s^R(0^-); \tag{4.17}$$

$$L_L \zeta_L \partial_z \mathbf{f}_t^R(0^-) = L_R \zeta_R \partial_z \mathbf{f}_t^R(0^+) = \mathbf{f}_t^R(0^+) - \mathbf{f}_t^R(0^-). \tag{4.18}$$

So the derivative changes by a factor $L_L\zeta_L/L_R\zeta_R$ across the interface, and will on both sides be proportional to the difference across the interface.

In this section, we consider the special case when the ferromagnet has an exchange field along the z-axis, and where the spin-orbit coupling is restricted to the ferromagnetic region. First of all, starting with a gap Δ in the superconductor, eq. (4.12) implies that we necessarily also have a singlet component f_s^R there, and according to eq. (4.17), this component leaks into the ferromagnet as well. Once inside the ferromagnet, eq. (4.13) shows that the presence of a singlet component also induces a short-range triplet component f_z^R. However, the spin-orbit term in eq. (4.13) does not provide any way for f_z^R to rotate into one of the long-range components f_x^R and f_y^R, so these components decouple entirely from the equations. This means that we only need two equations in the superconductor,

$$\frac{i}{2} D\, \partial_z^2 f_s^R = \epsilon f_s^R - \Delta\,, \tag{4.19}$$

$$\frac{i}{2} D\, \partial_z^2 f_z^R = \epsilon f_z^R\,, \tag{4.20}$$

and two equations in the ferromagnet,

$$\frac{i}{2} D\, \partial_z^2 f_s^R = \epsilon f_s^R\,, \tag{4.21}$$

$$\frac{i}{2} D\, \partial_z^2 f_z^R = E_z f_z^R\,, \tag{4.22}$$

where we defined the effective triplet energy

$$E_z \equiv \epsilon + 4iD\alpha^2\,, \tag{4.23}$$

and the boundary conditions are given by eqs. (4.17) and (4.18). As was pointed out in the previous section, an imaginary energy contribution is detrimental to the presence of Cooper pairs, so the net effect of eq. (4.23) is to make the triplet pairs less energetically favourable than singlet pairs. This effect increases with the spin-orbit strength α, but is notably independent of the spin-orbit angle χ. The latter implies that the effect should be equally significant for systems with Rashba coupling and Dresselhaus coupling, as long as the strength of the coupling is the same. Following this argument a bit further, the implication would be a suppression of triplets for high values of α, which would lead to a larger induced minigap in the ferromagnet. If this suppression also reduces the leakage of singlet pairs from the superconductor, we would also expect an increase in critical temperature of the system for increasing α.

We will now consider a slightly different geometry, where the exchange field of the ferromagnet is constrained to the xy-plane instead,

$$\mathbf{h} = h(\cos\theta\, \boldsymbol{e}_x + \sin\theta\, \boldsymbol{e}_y)\,. \tag{4.24}$$

According to eq. (4.13), the presence of a singlet component f_s^R will again produce a short-range triplet component $f_\parallel^R$ in the ferromagnet, but this time along the direction $(\cos\theta, \sin\theta, 0)$ in the xy-plane. Depending on the particular values of the magnetization angle θ and spin-orbit angle χ, the spin-orbit term in eq. (4.13) may then produce a long-range triplet component $f_\perp^R$ along the perpendicular direction $(-\sin\theta, \cos\theta, 0)$. However, the triplet component f_z^R decouples completely from the equations since the spin-orbit term only mixes f_x^R and f_y^R. This is in other words the opposite situation from what we considered in the previous section, where it was f_x^R and f_y^R that decoupled from the rest of the equations.

This can be made more explicit by rewriting the diffusion equation in terms of $f_\parallel^R$ and $f_\perp^R$, which can be expressed as

$$f_\parallel^R = \quad \cos\theta\, f_x^R + \sin\theta\, f_y^R\,, \tag{4.25}$$

$$f_\perp^R = -\sin\theta\, f_x^R + \cos\theta\, f_y^R\,. \tag{4.26}$$

Rewriting eq. (4.12) using $\mathbf{h}\cdot\mathbf{f}_t^R = h f_\parallel^R$, and projecting eq. (4.13) along the unit vectors $(\cos\theta, \sin\theta, 0)$ and $(-\sin\theta, \cos\theta, 0)$, respectively, we obtain

$$\frac{i}{2} D\, \partial_z^2 f_s^R = \epsilon f_s^R + h f_\parallel^R - \Delta\,, \tag{4.27}$$

$$\begin{aligned}\frac{i}{2} D\, \partial_z^2 f_\parallel^R = {} & \left[\epsilon + 2iD a^2 (1 - \sin 2\theta \sin 2\chi)\right] f_\parallel^R \\ & - 2iD a^2 \cos 2\theta \sin 2\chi\, f_\perp^R + h f_s^R\,,\end{aligned} \tag{4.28}$$

$$\begin{aligned}\frac{i}{2} D\, \partial_z^2 f_\perp^R = {} & \left[\epsilon + 2iD a^2 (1 + \sin 2\theta \sin 2\chi)\right] f_\perp^R \\ & - 2iD a^2 \cos 2\theta \sin 2\chi\, f_\parallel^R\,.\end{aligned} \tag{4.29}$$

From this result, we get three decoupled equations in the superconductor,

$$\frac{i}{2} D\, \partial_z^2 f_s^R = \epsilon f_s^R - \Delta\,, \tag{4.30}$$

$$\frac{i}{2} D\, \partial_z^2 f_\parallel^R = \epsilon f_\parallel^R\,, \tag{4.31}$$

$$\frac{i}{2} D\, \partial_z^2 f_\perp^R = \epsilon f_\perp^R\,, \tag{4.32}$$

and three coupled equations in the ferromagnet,

$$\frac{i}{2}D\,\partial_z^2 f_s^R = \epsilon f_s^R + h f_\parallel^R\,, \tag{4.33}$$

$$\frac{i}{2}D\,\partial_z^2 f_\parallel^R = E_\parallel f_\parallel^R + X f_\perp^R + h f_s^R\,, \tag{4.34}$$

$$\frac{i}{2}D\,\partial_z^2 f_\perp^R = E_\perp f_\perp^R + X f_\parallel^R\,, \tag{4.35}$$

where we defined the effective energies and mixing factor

$$E_\parallel \equiv \epsilon + 2iD\alpha^2(1 - \sin 2\theta \sin 2\chi)\,, \tag{4.36}$$

$$E_\perp \equiv \epsilon + 2iD\alpha^2(1 + \sin 2\theta \sin 2\chi)\,, \tag{4.37}$$

$$X \equiv -2iD\alpha^2 \cos 2\theta \sin 2\chi\,. \tag{4.38}$$

The boundary conditions are still given by eqs. (4.17) and (4.18).

These equations clearly elucidate the interplay between the different kinds of pairing in the system. If we start with a singlet component f_s^R, the presence of an exchange field $\mathbf{h}$ causes a singlet/triplet mixing, which produces the short-range triplet component $f_\parallel^R$. The spin-orbit field then causes a triplet/triplet mixing given by $X \sim \cos 2\theta \sin 2\chi$, which produces the long-range component $f_\perp^R$. However, contrary to the case in ferromagnets without spin-orbit coupling, the long-range triplet component is not necessarily the most stable species in the system. Notably, when $\operatorname{sgn}\theta = \operatorname{sgn}\chi$, eqs. (4.36) and (4.37) show that the long-range component has a much larger imaginary energy contribution than the short-range component, suggesting that a strong spin-orbit interaction might make the short-range component the most stable of the triplet components. It is only when $\operatorname{sgn}\theta = -\operatorname{sgn}\chi$ that both the exchange field and spin-orbit coupling favours the long-range component, so this is the parameter regime where we would expect a significant triplet proximity effect. Curiously, for $\theta = -\operatorname{sgn}(\chi)\,\pi/4$, there is no way to generate the long-range triplets because $X = 0$, while $E_\parallel$ is simultaneously maximized. This suggests that we would observe a suppression of all triplet components for these parameters, making the singlet component the dominant species.

While $\chi = \pm\pi/4$ is the optimal spin-orbit angle for generating long-range triplets, it is not immediately obvious what an optimal value of θ would be. One one hand, $|X|$ is maximized for the axial field configurations $\theta = 0$ and $\theta = \pm\pi/2$ (see fig. 4.1); but on the other hand, $E_\parallel/E_\perp$ is maximized for the intermediate value $\theta = \operatorname{sgn}(\chi)\pi/4$ (see fig. 4.2). As we will see in chapter 5, the result of the competition between these two factors is a highly nonmonotonic dependence on θ.

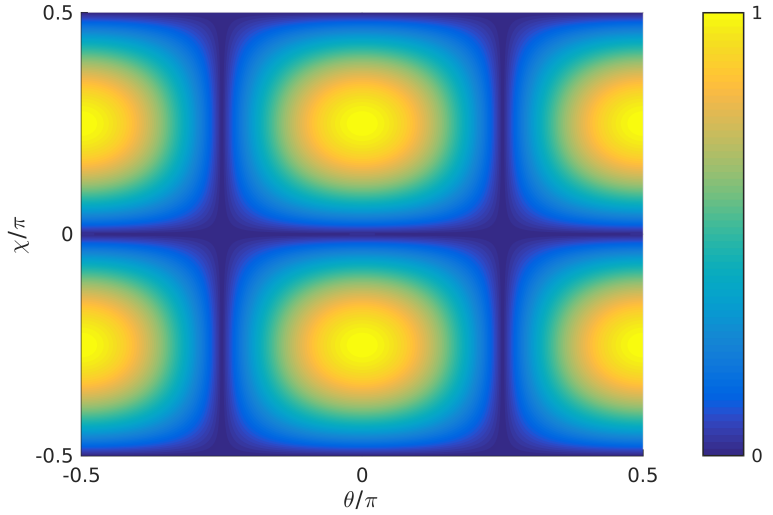


FIGURE 4.1: Plot of the absolute value $|X|/2Da^2$ of the mixing factor as a function of the magnetization angle θ and spin-orbit angle χ. We see that $|X|$ drops to zero for $\chi = 0$ and $\chi = \pm\pi/2$, so there is no triplet mixing for pure Rashba or Dresselhaus couplings. The same happens for $\theta = \pm\pi/4$. This is consistent with the results of Bergeret and Tokatly for in-plane field configurations.[19]

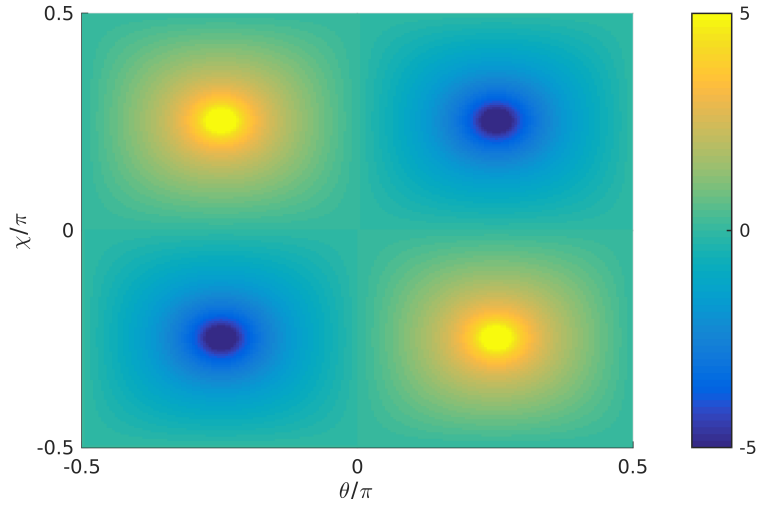


FIGURE 4.2: Plot of the logarithmic energy ratio $\ln(E_\parallel/E_\perp)$ as a function of the magnetization angle θ and spin-orbit angle χ when $\epsilon = 0$. In the bright yellow regions where $\operatorname{sgn}\theta = -\operatorname{sgn}\chi$, we see that $E_\parallel \gg E_\perp$, making the long-range triplet components highly energetically favoured. The exact opposite situation $E_\parallel \ll E_\perp$ occurs in the dark blue regions where $\operatorname{sgn}\theta = \operatorname{sgn}\chi$. In both cases, the most extreme behaviour clearly happens when $\chi = \pm\pi/4$ and $\theta = \pm\pi/4$.

If we extract and combine the anomalous components of eqs. (2.24) and (2.31), we see that the Matsubara Green's function $\underline{f}^{M}$ can be expressed as

$$\underline{f}^{M}(z,\omega) = \begin{cases} i\underline{f}^{R}(z,i\omega), & \omega \geqslant 0, \\ -i\underline{f}^{RT}(z,-i\omega), & \omega < 0, \end{cases} \tag{4.39}$$

which after a singlet/triplet decomposition using eq. (2.32) becomes

$$f_s^{M}(z,\omega) = i f_s^{R}(z,i|\omega|)\,, \tag{4.40}$$
$$\mathbf{f}_t^{M}(z,\omega) = i\mathbf{f}_t^{R}(z,i|\omega|)\operatorname{sgn}\omega\,. \tag{4.41}$$

Note that the singlet component f_s^{M} is an even function of the Matsubara frequency ω, while the triplet component $\mathbf{f}_t^{M}$ is an odd function. Letting $\epsilon \mapsto i|\omega|$ in eqs. (4.12) and (4.13), and substituting in eqs. (4.40) and (4.41), we obtain the linearized diffusion equations in the Matsubara formalism,

$$\frac{1}{2}D\,\partial_z^2 f_s^{M} = |\omega| f_s^{M} - i\operatorname{sgn}\omega\,\mathbf{h}\cdot\mathbf{f}_t^{M} - \Delta\,, \tag{4.42}$$
$$\frac{1}{2}D\,\partial_z^2 \mathbf{f}_t^{M} = |\omega|\mathbf{f}_t^{M} - i\operatorname{sgn}\omega\,\mathbf{h} f_s^{M} + 2D a^2 \boldsymbol{\Omega}(\chi)\mathbf{f}_t^{M}\,. \tag{4.43}$$

The boundary conditions take the exact same form as eqs. (4.17) and (4.18).

In general, the equations above have to be accompanied by eq. (3.51). However, the linearized equations are only valid when either the interface transparency is very low, or the system is close to the critical temperature. In the first case, using a *bulk approximation* $\Delta \cong \Delta_0$ in the superconducting region is justified. In the second case, it is common to use the *single-mode approximation* $\Delta \cong \delta\cos\big[k(z+L_s)\big]$ for the superconducting region, where L_s is the length of the superconductor, and the modulation k is determined by solving the Usadel equation in both regions.[94,97] The latter can also be generalized to a *multi-mode approximation* by adding higher-order terms of the kind $\delta_n \cosh\big[\kappa_n(z+L_s)\big]$.[94] We will not pursue the analytic solution of the linearized equations further in this thesis; instead, we move on to the full numerical solution of the nonlinear equations in the next chapter, but will use the linearized equations for qualitative explanations of the observed phenomena.

5

STRONG PROXIMITY EFFECT

In this chapter, we present numerical results for the density of states and critical temperature of superconductor/ferromagnet hybrid systems with in-plane Rashba–Dresselhaus coupling. We find that the observables have a highly nonmonotonic dependence on magnetization angles and phase differences, and that this behaviour can be qualitatively explained using the linearized equations from the previous chapter.

5.1 DENSITY OF STATES IN BILAYERS

First, we will consider a superconductor/ferromagnet bilayer, where the ferromagnetic region has an intrinsic spin-orbit coupling (see fig. 5.1). The simulations are performed using a zero-temperature bulk solution in the superconducting region, and the proximity effect is studied in the form of the central density of states in the ferromagnetic region. We use a ferromagnet of length $L_F = \xi_S/2$ which has a homogeneous in-plane magnetization $h = 3\Delta_0(\cos\theta\, \boldsymbol{e}_x + \sin\theta\, \boldsymbol{e}_y)$, and study the behaviour of the system as a function of the magnetization angle θ. The two materials are adjoined by a tunneling interface with a relative resistance $\zeta = 3$, and the structure is surrounded by a vacuum of infinite resistance $\zeta = \infty$. We assume an in-plane Rasbha–Dresselhaus coupling as defined in eq. (2.10), which is parametrized by a spin-orbit strength α and spin-orbit angle χ. Without the spin-orbit coupling, the density of states for a system with these parameters is quite flat ($D(0) = 1.025$), but an explicit calculation of the singlet and triplet projections confirms that both are present.

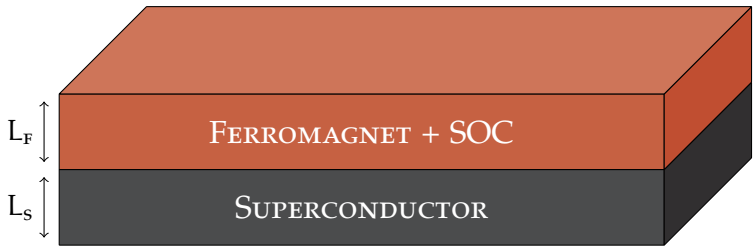


FIGURE 5.1: The physical system considered in this section.

The first plot that we present, is of the zero-energy density of states $D(0)$ as a function of the spin-orbit angle χ and magnetization angle θ (fig. 5.2). In the upper-right and bottom-left quadrants of the plot, i.e. the regions where $\mathrm{sgn}(\theta/\chi) = +1$, the spin-orbit coupling seems to have little to no effect on the zero-energy density, resulting in flat regions with $D(0) \cong 1$. However, the situation is drastically different for the quadrants where $\mathrm{sgn}(\theta/\chi) = -1$. In these regions, $D(0)$ increases as $\theta \to -\mathrm{sgn}(\chi)\,\pi/4$, and the zero-energy peaks indicate a long-range triplet proximity effect for these parameters. When θ gets *too close* to $-\mathrm{sgn}(\chi)\,\pi/4$ on the other hand, $D(0)$ abruptly drops to zero. This means that the density of states has been split by a minigap, and that we have had a sudden transition from a triplet-dominated to singlet-dominated state. There is nothing special about the parameters chosen here; the behaviour persists even in very strong ferromagnets with $h = 20\Delta_0$, as long as a is scaled proportionally.[a]

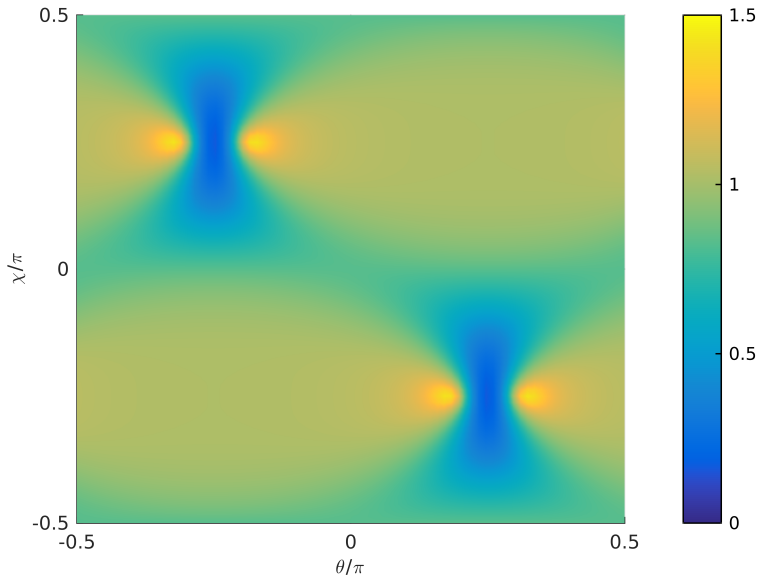
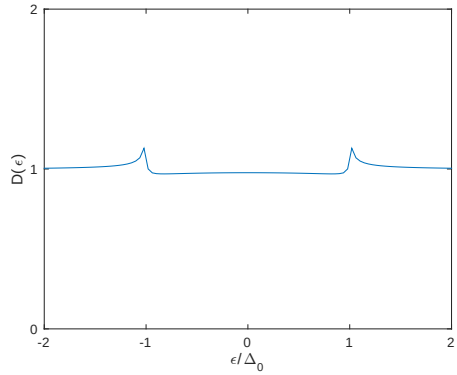


FIGURE 5.2: Zero-energy density $D(0)$ as a function of the spin-orbit angle χ and magnetization angle θ. Note the nonmonotonic dependence on magnetization angle for $\chi = \pm\pi/4$, which corresponds to a Rashba and Dresselhaus coupling of equal strength. This plot was generated for the spin-orbit strength $a\xi_s = 2$.

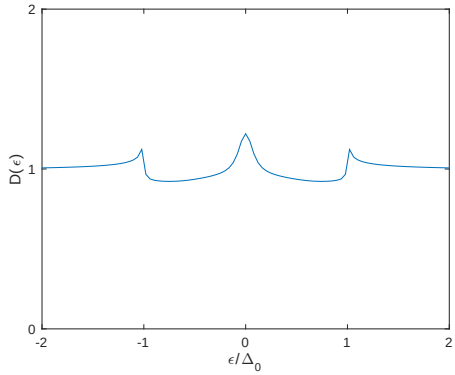
[a]Numerically, the behaviour persists for even higher field configurations than this. However, the quasiclassical approximation is only valid when the spin-orbit strength a is well below the Fermi wavenumber, which is of the order 10^{10} Å^{-1} in typical metals.[61] Assuming a superconducting coherence length of $\xi_s \cong 30$ nm, this implies that we must require $a\xi_s \ll 300$, so we should expect to obtain reasonable results as long as $a\xi_s \lesssim 15$.

This behaviour is in excellent agreement with what we would expect from eqs. (4.36) to (4.38). When $\mathrm{sgn}(\theta/\chi) = +1$, the imaginary energy of the short-range triplets $\mathrm{Im}\{E_{\parallel}\} \sim (1 - \sin 2\theta \sin 2\chi)$ is minimized, while the corresponding quantity $\mathrm{Im}\{E_{\perp}\} \sim (1 + \sin 2\theta \sin 2\chi)$ for the long-range triplets is maximized. In other words, the short-range triplets are nearly unaffected by the presence of a spin-orbit coupling, while the long-range triplets are strongly suppressed. In this regime, we therefore get the same result $D(0) \cong 1$ as without spin-orbit coupling. When $\mathrm{sgn}(\theta/\chi) = -1$ on the other hand, the same equations show that short-ranged triplets are penalized by a large imaginary energy, while the long-range triplets become energetically favourable. This results in a large-scale conversion from short-ranged to long-ranged triplets in the system, causing the pronounced zero-energy peaks that we see in the figure. However, the triplet mixing factor $X \sim \cos 2\theta \sin 2\chi$ that is responsible for this conversion vanishes as $\theta \to \pm\pi/4$, so the long-range triplets decouple from the equations in this limit. Since the short-range triplets are already heavily suppressed by the spin-orbit coupling, the most stable species in the system will be the singlet component, and this manifests as a minigap with $D(0) < 1$. For a visualization of these competing effects, we note that the top-left and bottom-right quadrants of fig. 5.2 can be thought of as some kind of product between figs. 4.1 and 4.2, which again highlights the qualitative agreement between the linearized equations and numerical results.

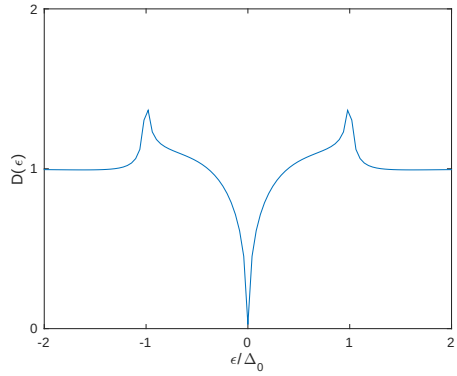
According to fig. 5.2, the optimal spin-orbit angles for observing a large θ-dependence are $\chi = \pm\pi/4$, corresponding to a Rashba and Dresselhaus coupling of equal strength. We also see that $\theta \cong -0.17\pi$ and $\theta = -0.25\pi$, respectively, correspond to one of the greatest zero-energy peaks and deepest minigaps for $\chi = \pi/4$. We therefore focus on these two extrema, and display the density of states $D(\epsilon)$ for these configurations in fig. 5.3. When $\theta = -0.25\pi$, the qualitative features of the density of states are altered as the spin-orbit strength α is increased. When $\alpha\xi_s = 1$, we get the same result $D(0) \cong 1$ as without a spin-orbit coupling, which means that the coupling is not strong enough to suppress the short-range triplet components. When the strength is increased to $\alpha\xi_s = 2$, we seem to have reached some critical value where a minigap with $D(0) = 0$ forms. If the strength is further increased to $\alpha\xi_s = 4$, then the minigap opens completely, indicating a strong singlet proximity effect. When $\theta = -0.17\pi$ on the other hand, a zero-energy peak forms in all three cases, suggesting long-range triplet proximity effects. *Thus, we may toggle between a minigap and zero-energy peak by rotating the magnetization. This is a novel phenomenon caused by the spin-orbit coupling, and one of the main results presented herein.*



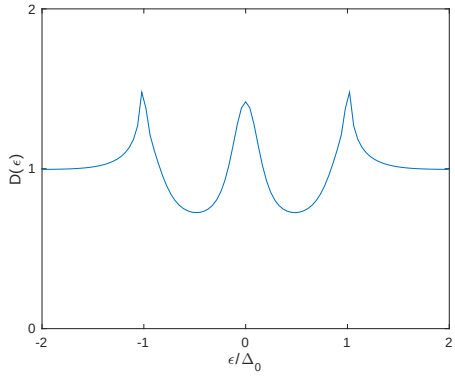
(a) $a\xi_s = 1$ and $\theta = -0.25\pi$;



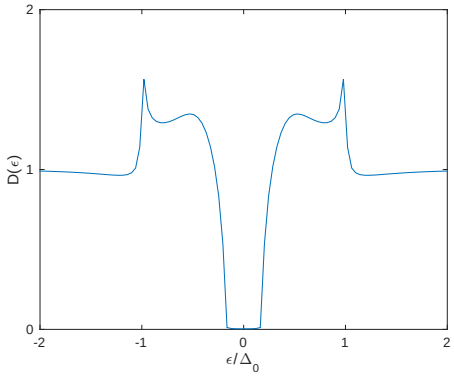
(b) $a\xi_s = 1$ and $\theta = -0.17\pi$;



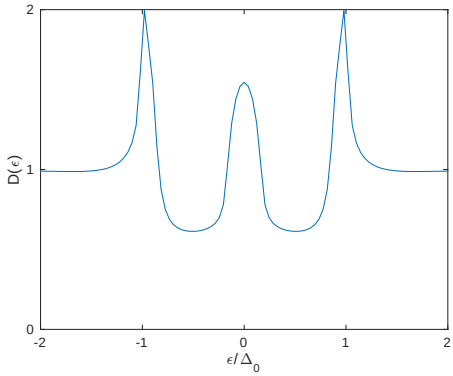
(c) $a\xi_s = 2$ and $\theta = -0.25\pi$;



(d) $a\xi_s = 2$ and $\theta = -0.17\pi$;



(e) $a\xi_s = 4$ and $\theta = -0.25\pi$;



(f) $a\xi_s = 4$ and $\theta = -0.17\pi$.

FIGURE 5.3: Density of states $D(\epsilon)$ for different spin-orbit strengths a and magnetization angles θ. These plots were generated for the spin-orbit angle $\chi = \pi/4$.

We will now consider a trilayer consisting of a ferromagnet interlocked between two bulk superconductors (see fig. 5.4), where the materials have the same dimensions and properties as in the previous section. The system is a kind of Josephson junction, and is sensitive to the phase difference ϕ between the superconductors. There are therefore two interesting parameters to vary in experiments on such junctions: the magnetization angle θ, which can be rotated by an external magnetic field; and the phase difference ϕ, which can be altered by an external electric field.

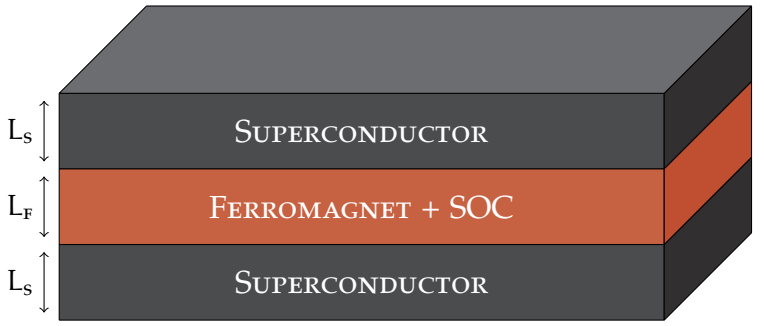
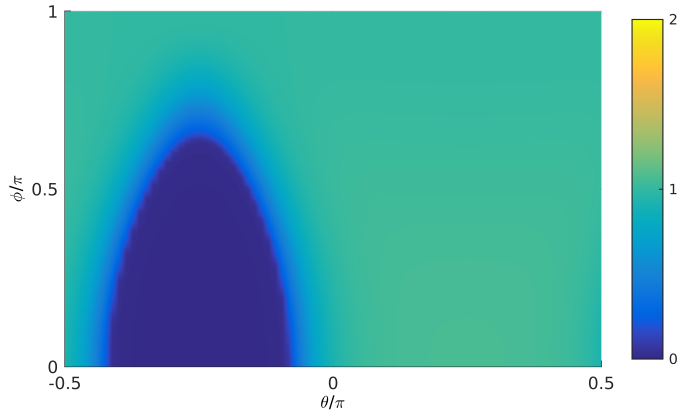
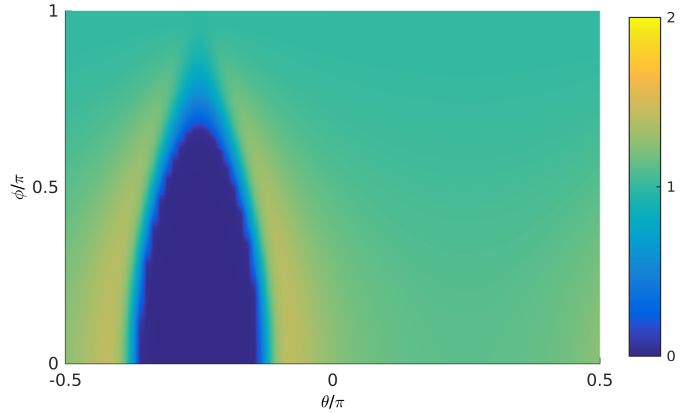


FIGURE 5.4: The physical system considered in this section.

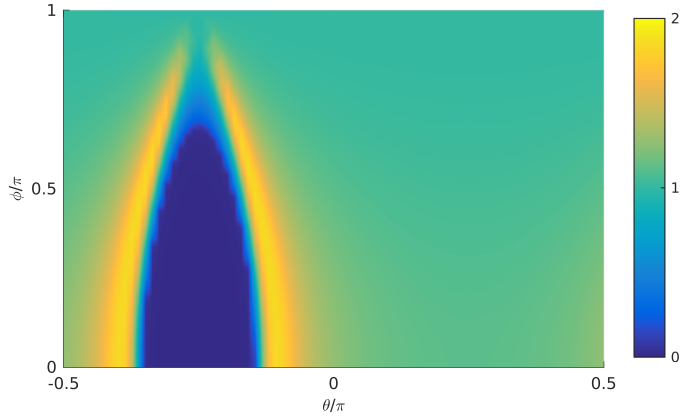
Following the same strategy as in the previous section, we will start the investigation by considering how the zero-energy density of states $D(0)$ varies with the system parameters. The results are shown in fig. 5.5, and are consistent with the results in fig. 5.2. First of all, a minigap forms near $\theta = -\pi/4$, which is explained by a suppression of short-range triplets in the system. In fact, the effect is even stronger for the trilayer than it was for the bilayer, which might be explained by having an influx of singlet pairs from two neighboring superconductors instead of one. Furthermore, the singlet-dominated region is bordered by a triplet-dominated region, where we interpret a zero-energy peak as a long-range triplet proximity effect. These zero-energy peaks are nonexistent for $\chi = 0.15\pi$, and maximized as $\chi \to 0.25\pi$, so the most interesting configuration is again when the Rashba and Dresselhaus coupling are of roughly equal strength. We will therefore focus on the special case $\chi = \pi/4$ for the rest of this section.



(a) $\chi = 0.15\pi$;



(b) $\chi = 0.22\pi$;

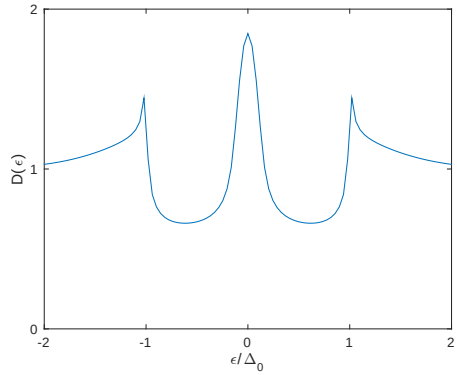


(c) $\chi = 0.25\pi$.

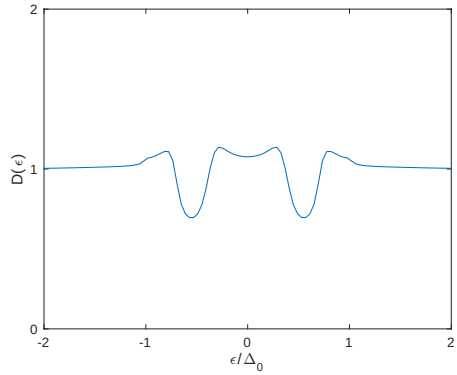
FIGURE 5.5: Zero-energy density $D(0)$ as a function of the phase difference ϕ and magnetization angle θ. These plots were generated for $a\xi_s = 2$.

It is particularly noteworthy that the system exhibits a nonmonotonic behaviour as a function of the magnetization angle *and* phase difference. The first of these is visible in any horizontal section of constant $\phi \lesssim 0.7\pi$ in fig. 5.5c; we observe that $D(0) \cong 1$ for $\theta = 0$, $D(0) \cong 2$ for $\theta = -0.10\pi$, and $D(0) \cong 0$ for $\theta = -0.25\pi$. *So by simply rotating the magnetization by $\pi/4$ radians, the system passes through a neutral region, strongly triplet-dominated region, and strongly singlet-dominated region, thus offering a convenient way to switch between these regimes in a laboratory setting.* The nonmonotonic dependence on phase difference is clearly visible in the vertical sections $\theta = -0.20\pi$ and $\theta = -0.30\pi$, where we observe $D(0) \cong 0$ for $\phi = 0$, $D(0) \cong 2$ for $\phi = 0.75\pi$, and $D(0) = 1$ for $\phi = \pi$.

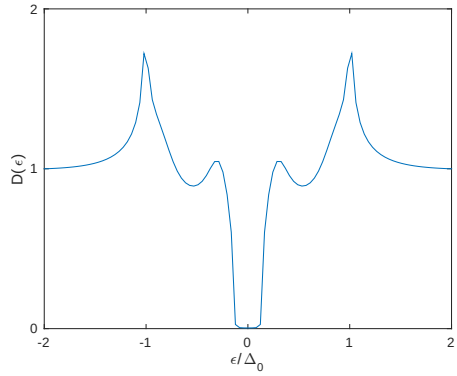
To better illustrate the nonmonotonic behaviour discussed above, we present a more detailed comparison of the density of states in fig. 5.6. For $\theta = -0.10\pi$, there is a significant zero-energy peak when $\phi = 0$, but the peak disappears when we increase the phase difference to $\phi = 0.75\pi$. This means that passing a current through the junction should lead to a diminishing zero-energy peak, which is similar to how a ferromagnetic junction without spin-orbit coupling would behave. For $\theta = -0.25\pi$, there is a significant minigap for $\phi = 0$, which is weakened when we increase the phase difference to $\phi = 0.75\pi$. Passing larger current through this junction should lead to a diminishing minigap, which is how a junction with neither exchange field nor spin-orbit coupling would behave. However, for the intermediate value $\theta = -0.20\pi$, something entirely different happens: we get a significant minigap for $\phi = 0$, but a significant peak for $\phi = 0.75\pi$. *Passing a current through this junction should therefore lead to a transition from a singlet-dominated to triplet-dominated state. This is a new kind of behaviour, and does not appear in junctions without spin-orbit coupling.* The behaviour outlined above also persists in much stronger exchange fields as long as the spin-orbit strength is scaled accordingly (see fig. 5.7).



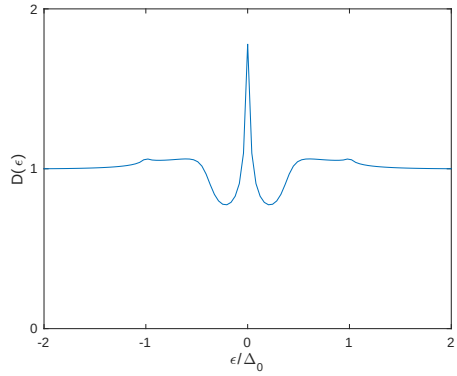
(a) $\theta = -0.10\pi$ and $\phi = 0$;



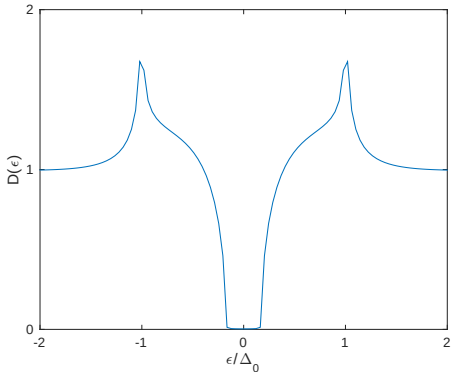
(b) $\theta = -0.10\pi$ and $\phi = 0.75\pi$;



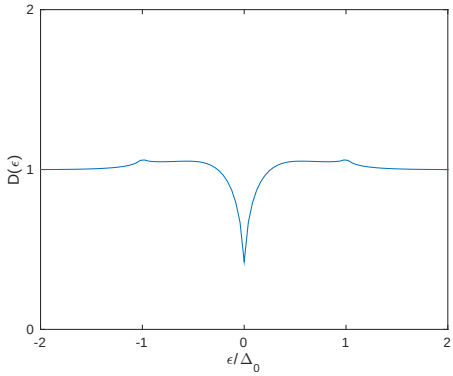
(c) $\theta = -0.20\pi$ and $\phi = 0$;



(d) $\theta = -0.20\pi$ and $\phi = 0.75\pi$;

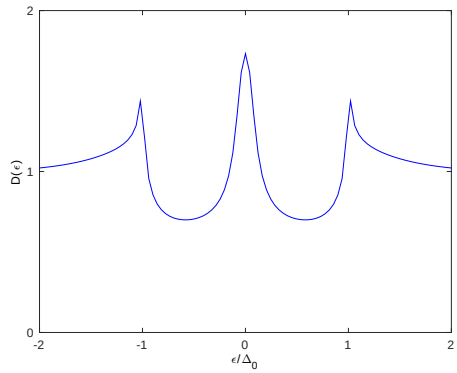


(e) $\theta = -0.25\pi$ and $\phi = 0$;

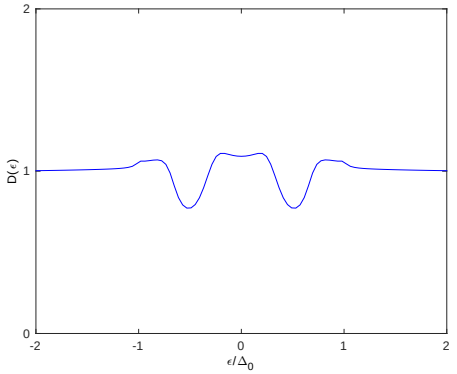


(f) $\theta = -0.25\pi$ and $\phi = 0.75\pi$.

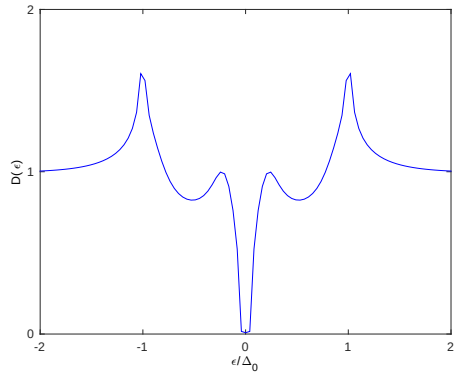
FIGURE 5.6: Density of states $D(\epsilon)$ for various magnetization angles θ and phase differences ϕ. Depending on magnetization angle, the effect of increasing the phase difference can be classified as either a triplet suppression, singlet/triplet transition, or singlet suppression. These plots were generated for the exchange field $h = 3\Delta_0$, spin-orbit strength $a\xi_s = 2$, and spin-orbit angle $\chi = \pi/4$.



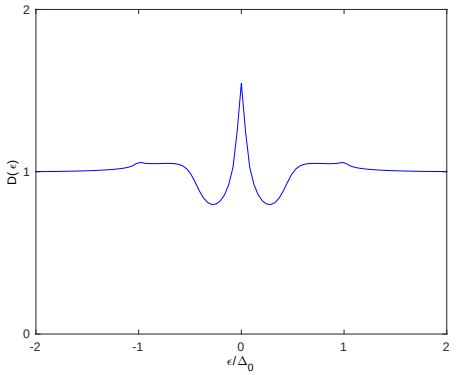
(a) $\theta = -0.21\pi$ and $\phi = 0$;



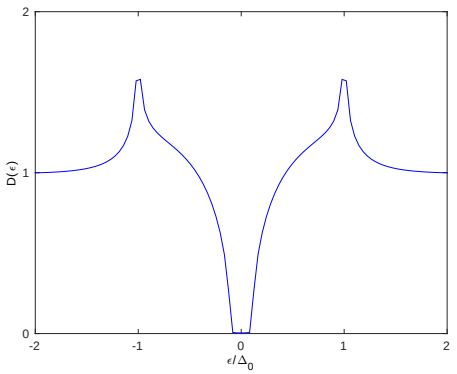
(b) $\theta = -0.21\pi$ and $\phi = 0.75\pi$;



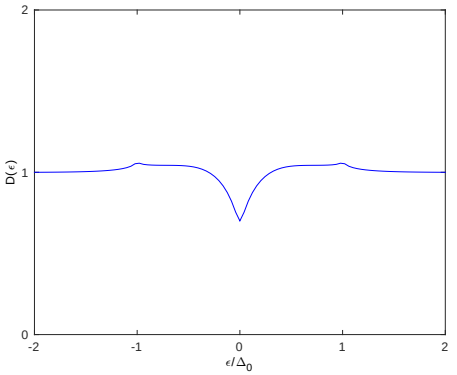
(c) $\theta = -0.23\pi$ and $\phi = 0$;



(d) $\theta = -0.23\pi$ and $\phi = 0.75\pi$;



(e) $\theta = -0.25\pi$ and $\phi = 0$;



(f) $\theta = -0.25\pi$ and $\phi = 0.75\pi$.

FIGURE 5.7: Density of states $D(\epsilon)$ for different magnetization angles θ and phase differences ϕ. These plots are similar to those in fig. 5.6, but were generated for the stronger exchange field $h = 10\Delta_0$ and spin-orbit strength $a\xi_s = 6$.

In this section, we will return to the superconductor/ferromagnet bilayer shown in fig. 5.1, and discuss the critical temperature T_c of the structure. The interface parameters are the same as in section 5.1, but we consider a shorter ferromagnet $L_F = 0.2\xi_S$ with a stronger exchange field $h = 10\Delta_0$. Since a self-consistent approach is required when determining the critical temperature, the superconductor can no longer be approximated by a bulk solution. Instead, we use eq. (3.37) to determine the gap, where we assume a material with the BCS coupling strength $N_0\lambda = 0.2$. The critical temperature was calculated using algorithm 3 in appendix A with $N = 12$ iterations, yielding T_c/T_{cs} to a precision of about 0.0001.

The first plots, figs. 5.8 and 5.9, show how the critical temperature T_c varies with the spin-orbit angle χ for an exchange field in the z-direction. The critical temperature is clearly independent of the spin-orbit angle χ, and behaves the same way for any combination of Rashba and Dresselhaus couplings with the same net strength. Furthermore, when the spin-orbit strength a increases, we see that the critical temperature T_c increases as well. This behaviour can be explained by the linearized equations from the previous chapter. According to eq. (4.23), the imaginary part of the effective triplet energy is $\operatorname{Im} E_z = 4Da^2$. The imaginary energy term suppresses the triplet component more and more as the spin-orbit strength a is increased, regardless of the spin-orbit angle χ. This effect reduces the amount of triplets in the ferromagnet, and as the triplet proximity channel is suppressed, the critical temperature is restored to higher values.

The same situation for an exchange field along the x-axis is shown in figs. 5.10 and 5.11. For this geometry, we observe a somewhat smaller critical temperature for all $a > 0$ and all χ compared to figs. 5.8 and 5.9. This can again be explained by the linearized equations. According to eqs. (4.36) and (4.37), the effective energies of the triplet components have the imaginary parts $\operatorname{Im} E_\parallel = \operatorname{Im} E_\perp = 2Da^2$ when $\theta = 0$. In other words, the triplets are less suppressed for an exchange field along the x-axis than z-axis because $\operatorname{Im} E_z$ is twice as large as $\operatorname{Im} E_\parallel$ and $\operatorname{Im} E_\perp$. We also observe a drop in critical temperature as $\chi \to \pm\pi/4$. According to eq. (4.38), this is exactly when the triplet mixing factor $X \sim \sin 2\chi$ is maximized. The explanation is therefore that long-range triplets are generated for these parameters, and as a new triplet proximity channel opens, the critical temperature is reduced. Note that for $\theta = 0$, there is no change in the effective energies $E_\parallel$ and $E_\perp$ when we vary χ, so the critical temperature reduction observed here is solely due to changes in the mixing factor X.

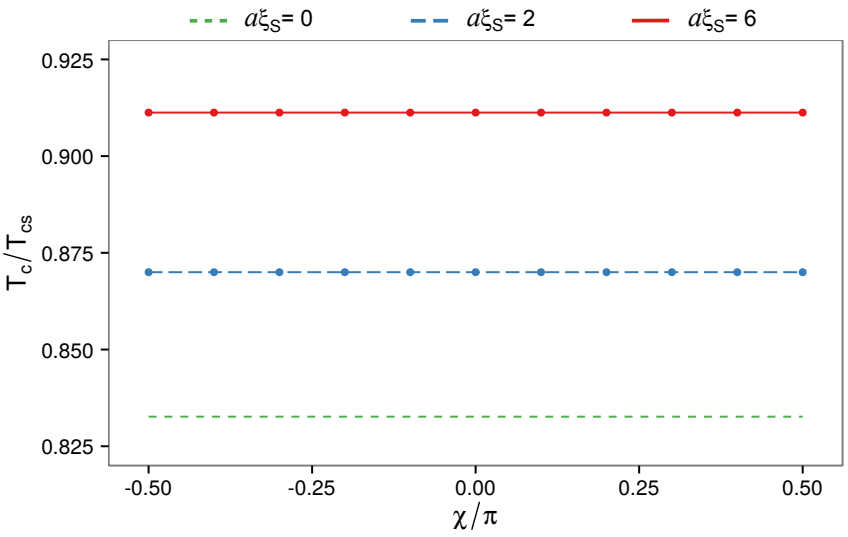


FIGURE 5.8: Plot of the critical temperature as a function of the spin-orbit angle χ when the superconductor length $L_s = 1.00\xi_s$ and exchange field $\mathbf{h} \sim \mathbf{e}_z$.

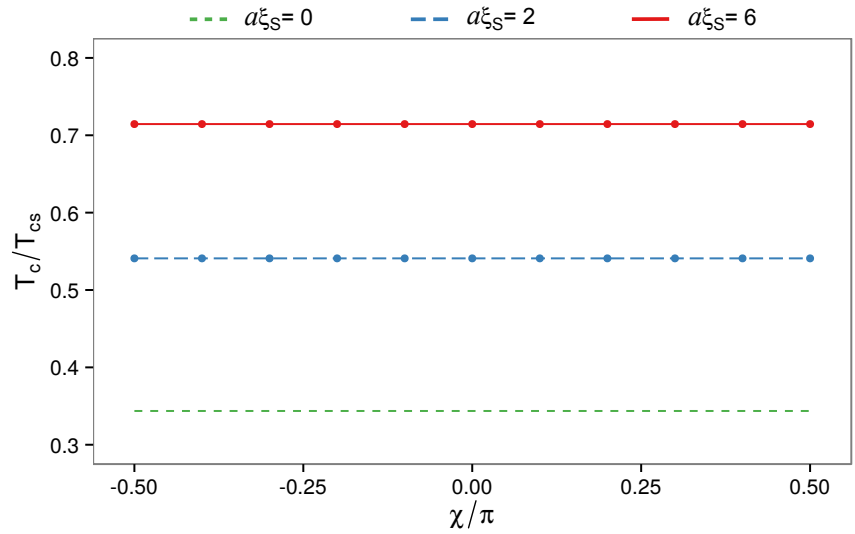


FIGURE 5.9: Plot of the critical temperature as a function of the spin-orbit angle χ when the superconductor length $L_s = 0.55\xi_s$ and exchange field $\mathbf{h} \sim \mathbf{e}_z$.

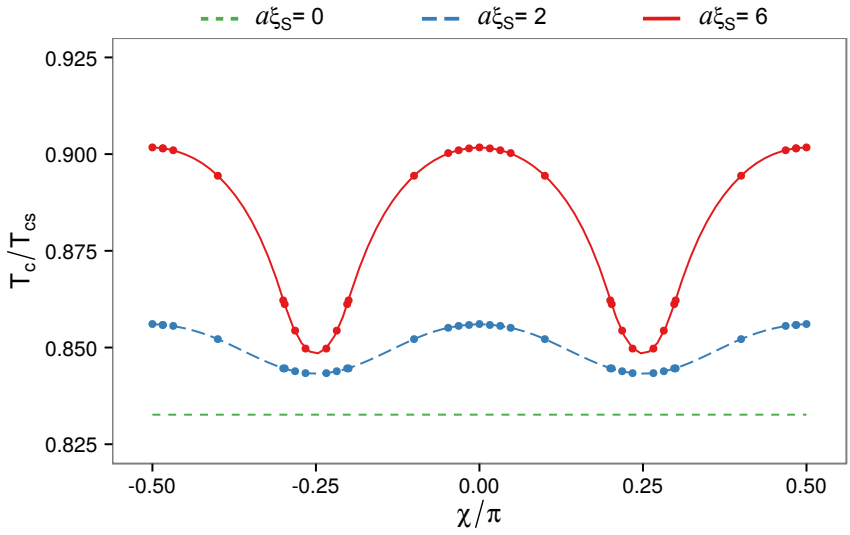


FIGURE 5.10: Plot of the critical temperature as function of the spin-orbit angle χ when the superconductor length $L_s = 1.00\xi_s$ and exchange field $\mathbf{h} \sim \mathbf{e}_x$.

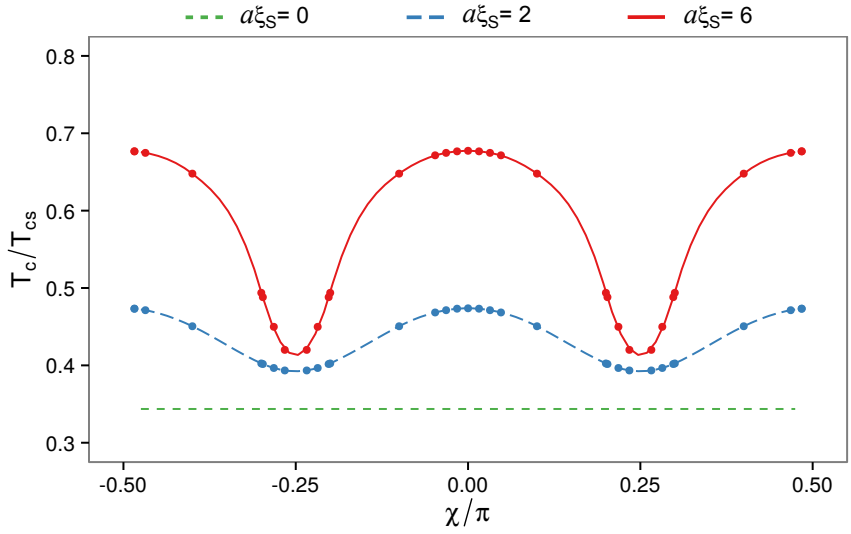


FIGURE 5.11: Plot of the critical temperature as function of the spin-orbit angle χ when the superconductor length $L_s = 0.55\xi_s$ and exchange field $\mathbf{h} \sim \mathbf{e}_x$.

In figs. 5.12 and 5.13, we examine how the critical temperature varies with the magnetization angle θ in the xy-plane. We include plots for the three representative spin-orbit angles $\chi = 0$ and $\chi = \pm\pi/4$. For $\chi = 0$, i.e. a pure Dresselhaus coupling, the critical temperature is independent of the magnetization angle. For $\chi = \pm\pi/2$ (not shown), i.e. a pure Rashba coupling, the results are identical. If $\chi = \pm\pi/4$, on the other hand, we find a more interesting behaviour: the critical temperature has a minimum at $\theta = +\chi$ and maximum at $\theta = -\chi$. Furthermore, for the choice of physical parameters in fig. 5.13, the effect results in a striking trough-to-peak increase of nearly 60%. As shown in fig. 5.12, the effect persists in larger structures as well, but is then weaker.

These results may again be interpreted using the linearized equations. Both the minimum and maximum are located at points where the mixing factor $X \sim \cos 2\theta \sin 2\chi$ vanishes, which indicates that the explanation cannot be related to the long-range triplets in this case. We will therefore focus on the relationship between the singlets and short-range triplets. When $\theta = +\chi$, the imaginary term in eq. (4.36) vanishes, so the short-range triplets have the energy $E_{\parallel} = \epsilon$. In other words, the spin-orbit coupling has no effect on the short-range triplets for these parameters, so we get the same low critical temperature as without spin-orbit coupling. However, when $\theta = -\chi$, the imaginary term in eq. (4.36) is maximized, and we find that $E_{\parallel} = \epsilon + 4iD\alpha^2$. This means that the spin-orbit coupling has a destructive influence on the short-range triplets. The suppression of the triplet proximity channel limits the leakage of Cooper pairs from the superconductor into the ferromagnet, increasing the critical temperature.

The results discussed above show that the critical temperature can be controlled via the magnetization direction of a single ferromagnetic layer. This is a new result originating from the presence of spin-orbit coupling: in conventional superconductor/ferromagnet bilayers, the critical temperature is independent of the magnetization direction of the ferromagnet. By using a spin-valve setup with a superconductor interlocked between two ferromagnets, it is well-known that the relative magnetization configuration between the ferromagnetic layers will tune the critical temperature of the system. *In contrast, in our case such a spin-valve effect can be obtained with a single ferromagnet: by rotating the magnetization by $\pi/2$ radians, the critical temperature goes from a maximum to a minimum. This is one of the main results presented in this thesis.* The fact that only one ferromagnet is required is of practical importance since it can be difficult to control multiple magnetization directions within a single mesoscopic structure.

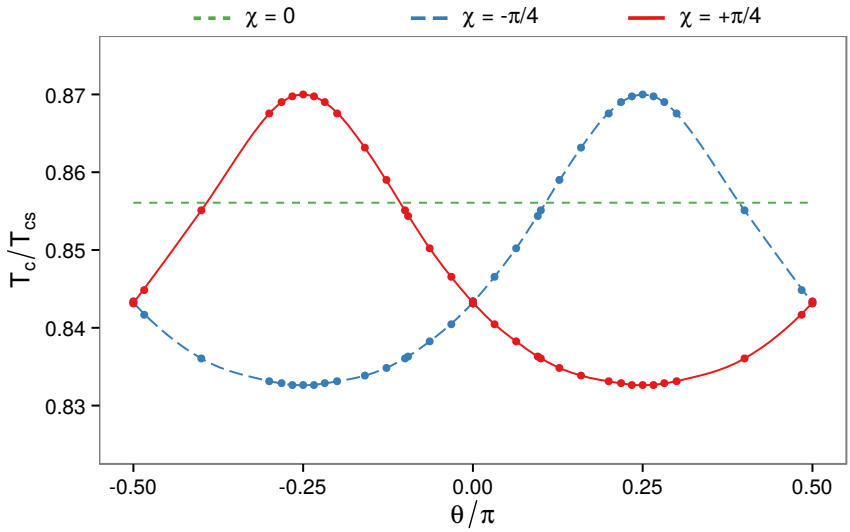


FIGURE 5.12: Plot of critical temperature as function of the exchange field angle θ when the superconductor length $L_s/\xi_s = 1.00$ and spin-orbit strength $a\xi_s = 2$.

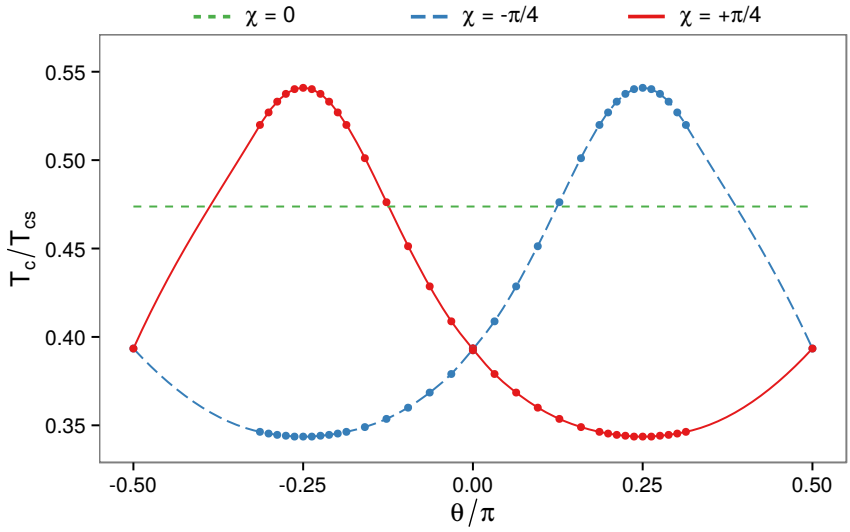


FIGURE 5.13: Plot of critical temperature as function of the exchange field angle θ when the superconductor length $L_s/\xi_s = 0.55$ and spin-orbit strength $a\xi_s = 2$.

6

SUMMARY AND OUTLOOK

In this thesis, we have demonstrated that the physics of superconducting hybrid structures can be drastically altered in the presence of a spin-orbit coupling. The effects were especially pronounced for the case of a Rashba and Dresselhaus coupling of similar strength, which suggests focusing on this class of systems in experimental setups. We also noted that these effects persist even in very strong ferromagnets, as long as there is also a proportionally strong spin-orbit coupling in the system.

For superconductor/ferromagnet bilayers, we observed that a rotation of the in-plane magnetization could cause the system to switch between strongly singlet-dominated and triplet-dominated regimes. This was indicated by the appearance of either a minigap or zero-energy peak in the density of states. Furthermore, self-consistent calculations revealed a very large variation in the critical temperature during such a rotation, at least if the superconductor is sufficiently small. This suggests that it should be possible to construct a superconducting spin-valve consisting of only one superconductor and one ferromagnet, in contrast to previous designs which required two ferromagnets. This is beneficial in laboratory settings, since it is quite difficult to control the relative magnetization between the components in a mesoscopic multilayer structure.

For the Josephson junctions, we observed that the system had a non-monotonic behaviour as a function of both the magnetization and the phase difference. By rotating the magnetization, it is therefore possible to switch between a singlet-dominated and triplet-dominated junction. A particularly interesting phenomenon occurs for intermediate magnetization directions, where increasing the phase difference causes a minigap to transform into a zero-energy peak. Since the phase difference is related to the amount of electricity flowing through the junction, passing a current through the junction should lead to a transformation from a singlet-dominated to triplet-dominated state for this magnetization.

The conclusion from the observations above is that spin-orbit coupling offers exciting new ways to exert control over superconductivity in hybrid structures, which may have important applications in the emerging field of superconducting spintronics.

We have presented how in-plane Rashba–Dresselhaus couplings affect the properties of superconducting hybrid systems. The natural next step would be to check the behaviour of related structures. For instance, we would intuitively expect superconductor/normal-metal/ferromagnet trilayers with a spin-orbit coupling in the normal metal to behave similarly to the superconductor/ferromagnet bilayer considered in this thesis. Another possibility would be superconductor/normal-metal bilayers with highly polarized spin-active interfaces, where we again have a spin-orbit coupling in the normal metal. Such configurations, where the magnetism and spin-orbit coupling reside in different layers, would provide the flexibility for experimentalists to select optimal materials for each purpose.

Another possibility which has been neglected in this thesis, is the case of an out-of-plane spin-orbit coupling, which can occur in noncentrosymmetric crystals. In such systems, a whole range of new effects enter the equations due to the derivative terms in eq. (2.44) and nonzero terms in eqs. (2.52) and (2.53). Recent results for such a system have already shown that there is a significant triplet proximity effect for Josephson junctions where the superconductors are in antiphase, which is a novel phenomenon which has not been reported for other structures before.[84]

This thesis has provided a thorough investigation of how superconductivity manifests in ferromagnets with spin-orbit coupling. However, it is by no means a complete investigation of the topic; there are still plenty of unexplored situations left to consider, and it is likely that even more novel phenomena will be discovered in these systems with time.

A ALGORITHM

In this appendix, we describe and compare algorithms which can be used to calculate the critical temperature of superconducting hybrid structures.

A.1 LINEAR SEARCH ALGORITHM

In order to introduce the notation, we will begin the discussion with a simple linear search algorithm, as shown in algorithm 1 below.

ALGORITHM 1: Linear search algorithm for finding the critical temperature.

```
procedure CriticalTemperature
    T ← 0
    Δ ← Δ_0
    γ ← γ_0
    while max|Δ| > δ do
        T ← T + τ
        repeat
            γ ← Usadel(γ, Δ)
            Δ_* ← Δ
            Δ ← Gap(γ, T)
        until max|Δ − Δ_*| < δ
    end while
end procedure
```

When the procedure is executed, we initialize the current temperature T to the absolute zero, the gap Δ to the corresponding bulk value Δ_0, and the Riccati parameter γ to the BCS bulk solution γ_0. Since $\gamma = \gamma(z, \epsilon)$ is actually a function of position and energy, a numerical implementation would have to discretize these variables, and store one copy of γ for each position and energy on the lattice. To obtain reliable results, we found that the calculations had to be performed for around 500 energies in the range $(0, 2\Delta_0)$, and 100 more energies in the range $(2\Delta_0, \omega_c)$, where the Debye cutoff $\omega_c \cong 74\Delta_0$ if we use the coupling constant $N_0\lambda = 0.2$. As for the resolution in position space, around 100–150 evenly spaced points

in each material seemed to be sufficient for most parameters. In practice, it is also convenient to store a copy of $\tilde{\gamma}$, as well as the first derivatives $\partial_z \gamma$ and $\partial_z \tilde{\gamma}$, at each lattice point. However, for brevity, we will simply refer to all this information about the Riccati parameters by the symbol γ.

After initializing the state variables, the temperature is increased by a small amount τ. The Riccati parameter γ is then recalculated by solving the Usadel equation, using the current value for the gap Δ as a constant background source, and using the previous value of γ as an initial guess for a solution.[a] The gap Δ is then recalculated using the new value for γ, by solving the gap equation at the current temperature T. This procedure is then repeated until the gap converges, which is measured by checking whether or not the current value of Δ has changed from the previous value Δ_* by less than some threshold δ. The algorithm will then continue to increase the temperature by τ at every iteration, until the maximum gap $|\Delta(z)|$ in the superconductor goes below the threshold δ. When this occurs, the system must have reached the critical temperature, so the current value for T is the final result for T_c.

The linear search algorithm presented here gets the job done, but is not very efficient. For instance, if we wish to calculate the critical temperature with a relative accuracy of 0.0001, we would need to test up to 10,000 temperatures between $T = 0$ and $T = T_{cs}$. For each of these temperatures, we have to solve a nonlinear differential equation at roughly 300 locations and 600 energies—and then repeat this procedure again and again until the gap converges. In other words, this is truly a formidable computation task, and may in practice take days to complete depending on hardware, implementation, and parameters. Note that all of this was required just to calculate a single number T_c; in order to gain physical insight into the behaviour of hybrid systems, the whole procedure would also have to be repeated for several different geometries and field strengths, adding a whole new dimension to the required computation time.

[a] In the case of e.g. a superconductor/ferromagnet bilayer, *solving the Usadel equation* involves first solving the equation in the ferromagnet using the old superconductor state as a boundary condition, and then solving the equation in the superconductor using the updated ferromagnet state as a boundary condition.

To reduce the required computation time, one alternative is to switch to a binary search algorithm instead, as outlined in algorithm 2 below.

ALGORITHM 2: Binary search algorithm for finding the critical temperature.

procedure CRITICALTEMPERATURE
  $T_1 \leftarrow 0$ ▷ $T_1 \leqslant T_c$
  $T_2 \leftarrow T_{cs}$ ▷ $T_2 \geqslant T_c$
  for $n \leftarrow 1, \ldots, N$ **do**
    $T \leftarrow (T_1 + T_2)/2$
    $\underline{\gamma} \leftarrow \underline{\gamma}_0$
    $\Delta \leftarrow \text{GAP}(\underline{\gamma}, T)$
    repeat
      $\underline{\gamma} \leftarrow \text{USADEL}(\underline{\gamma}, \Delta)$
      $\Delta_* \leftarrow \Delta$
      $\Delta \leftarrow \text{GAP}(\underline{\gamma}, T)$
    until $\max|\Delta - \Delta_*| < \delta$
    if $\max|\Delta| < \delta$ **then** ▷ $T \geqslant T_c$
      $T_2 \leftarrow T$
    else
      $T_1 \leftarrow T$
    end if
  end for
  $T \leftarrow (T_1 + T_2)/2$ ▷ $|T - T_c| \leqslant |T_2 - T_1|/2^{N+1}$
end procedure

For this algorithm to successfully determine the critical temperature, three crucial assumptions about the problem have to be made:

(i) $0 \leqslant T_c \leqslant T_{cs}$;

(ii) $\forall T < T_c : |\Delta| > 0$;

(iii) $\forall T \geqslant T_c : |\Delta| = 0$.

The first assumption, that the critical temperature T_c of a hybrid system has to be between the absolute zero and the bulk value T_{cs}, is quite trivial. It is well-known that the effect of an inverse proximity effect is to suppress the superconductivity of a material. If somehow an exception were to occur, the symptom would be the result $T_c = T_{cs}$, which could prompt

new calculations with the initial value $T_2 > T_{cs}$ to verify the result. The two other assumptions may be taken as a definition of critical temperature, together with the requirement that it is a single-valued quantity. It is more subtle to justify the single-valuedness, since the gap can be a reentrant function of temperature in e.g. bilayers with holmium.[98] However, no signs of reentrant behaviour were observed in the systems considered herein, possibly due to the choice of interface transparency $\zeta = 3$. We stress that Δ can still exhibit a nonmonotonic dependence on temperature, and a reentrant dependence on any other system parameters—just not a reentrant dependence on temperature. If this assumption is violated, the algorithm would likely return *one* of the critical temperatures, but *which one* would not necessarily be well-defined. Such a violation could therefore manifest as a discontinuous behaviour of the critical temperature.

In this algorithm, we also define variables that give a lower limit T_1 and upper limit T_2 for the critical temperature. Just like in algorithm 1, we assume that the critical temperature is between the absolute zero and the bulk value, and initialize these variables accordingly. For every iteration of the binary search, we then set the current temperature of the system to the midpoint $T = (T_1 + T_2)/2$, initialize the system to a zero-temperature bulk solution $\underline{\gamma}_0$, and perform a self-consistent calculation like before. If the gap Δ drops to zero, expressed numerically as dropping below the threshold δ, we conclude that $T \geqslant T_c$ according to assumption (ii), and update the upper limit to $T_2 = T$. If not, meaning that the gap must have converged to some finite value $|\Delta| > \delta$, we conclude that $T < T_c$ according to assumption (i), and update the lower limit to $T_1 = T$. By repeating this procedure of testing the midpoint $T = (T_1 + T_2)/2$ of the interval, and updating the limits T_1 and T_2 based on the results, the possible location of the critical temperature is confined to an interval (T_1, T_2) that decreases exponentially with the number of iterations N. For instance, while a precision of 0.0001 requires up to 10,000 iterations with algorithm 1, the number is reduced to just 12 iterations with algorithm 2, resulting in a remarkable increase in efficiency for high precision requirements.

While algorithm 2 was a remarkable improvement upon algorithm 1, it is still converges relatively slowly for two main reasons:

(i) For every iteration, we restart the calculation from a bulk solution $\underline{\gamma}_0$;

(ii) For temperatures $T > T_c$, the convergence $\Delta \to 0$ takes a nearly exponential amount of time, often changing by just 1–2 % per iteration.

To solve the first issue, it may be tempting to initialize $\underline{\gamma}$ to a bulk solution only at the start of the program, and then reuse the last calculated value of $\underline{\gamma}$ when switching temperature. The problem with this approach, is that the non-superconducting solution with $\underline{\gamma} = 0$ and $\Delta = 0$ satisfies the Usadel equation at all temperatures, but this is not the solution branch that is interesting when $T < T_c$. Thus, if the system converges to $\underline{\gamma} = 0$ for some $T > T_c$, and we later use this as an initial guess for some $T < T_c$, then we risk ending up on the wrong solution branch $\underline{\gamma} = 0$, yielding unphysical results. The solution to this problem is to save a copy of the last *nonzero* solution to a backup variable $\underline{\gamma}_*$. This should be done at the end of each iteration where we have confirmed that $T < T_c$, so that we always use a value of $\underline{\gamma}$ from a lower temperature as the initial guess.

To handle the second issue, we make an additional assumption about the behaviour of the system, namely that the gap decreases monotonically to zero if $T \geqslant T_c$. In other words, for supercritical temperatures $T > T_c$, we assume that every time we solve the Usadel equation and recalculate the gap, the new value for the gap must be lower than the previous one. This assumption should be quite reasonable; an exception would have to be a system with a highly nonlinear dependence on the gap, such that the amount of Cooper pairs spontaneously increases when the gap is in a range (Δ_1, Δ_2), but decreases for all other values of the gap. This would imply the existence of metastable states with $\Delta = \Delta_2$, in addition to the critical state with $\Delta = 0$, making the gap a multivalued function of temperature. The author is unaware of any superconductors where this sort of behaviour occurs, and we therefore neglect this possibility herein.

Given the assumption above, we do not need to obtain a self-consistent solution at all temperatures. If we ever observe that the gap *increases* between two iterations, the new assumption leads to the conclusion $T < T_c$, regardless of whether the gap changed by more than a threshold δ during

the iteration or not.[b] The strategy for reducing the computation time is as follows: we reduce the gap $\Delta \mapsto \Delta/2$, and then perform two iterations of the self-consistent solver.[c] The next step is to perform M iterations of the usual self-consistent solver, where we declare that $T < T_c$ if the gap increases during an iteration, and $T \geqslant T_c$ if it goes below the threshold δ. Remarkably, by performing the operation $\Delta \mapsto \Delta/2$ regularly, the number of iterations before the gap drops to zero for $T \geqslant T_c$ goes from roughly exponential to logarithmic, saving a tremendous amount of computation time. On the other hand, for $T < T_c$, the only information required is whether the gap converges towards a finite value or not. Letting $\Delta \mapsto \Delta/2$ regularly, and simply checking whether the gap then increases between iterations, permits us to obtain this information much quicker.

Pseudocode describing the resulting algorithm is given in algorithm 3 on the next page. An implementation in MATLAB has also been enclosed at the back of this thesis. The attachment `critical_superconductor.m` illustrates the basic concept for a superconductor in vacuum, and was primarily used as a guinea pig during the initial tests of the algorithm. The more interesting `critical_bilayer.m` implements the same algorithm for superconductor/ferromagnet bilayers with spin-orbit coupling, and was used to calculate all the critical temperature data presented in this thesis. We used $M = 6$ iterations between each time the gap was reduced to allow the system a few iterations to stabilize numerically, and performed in total $N = 12$ iterations of the binary search algorithm.

[b] The converse is however not true: if the gap *decreases* between two iterations, then both $T < T_c$ and $T \geqslant T_c$ are possible conclusions, so more calculations are required.

[c] If we performed only one iteration after reducing the gap, we might in principle observe an increasing gap due to Cooper pairs leaking back into the superconductor from the ferromagnet. By performing two iterations, we ensure that the effect of reducing the gap has had an impact on both the ferromagnetic region and the superconducting region.

ALGORITHM 3: Optimized algorithm for finding the critical temperature.

```
procedure CriticalTemperature
    T_1 ← 0                                                    ▷ T_1 ⩽ T_c
    T_2 ← 1                                                    ▷ T_2 ⩾ T_c
    γ  ← γ_0
    γ_* ← γ_0
    for n ← 1, ..., N do
        T  ← (T_1 + T_2)/2
        Δ  ← Gap(γ, T)
        m ← 0
        loop
            Δ_* ← Δ
            m  ← m + 1
            if m mod M = 0 then
                Δ ← Δ/2
                γ ← Usadel(γ, Δ)
                Δ ← Gap(γ, T)
            end if
            γ ← Usadel(γ, Δ)
            Δ ← Gap(γ, T)
            if mean(|Δ| − |Δ_*|) > 0 then                      ▷ T ⩽ T_c
                T_1 ← T
                γ_* ← γ
                break loop
            else if max |Δ| < δ then                           ▷ T ⩾ T_c
                T_2 ← T
                γ  ← γ_*
                break loop
            end if
        end loop
    end for
    T ← (T_1 + T_2)/2                                  ▷ |T − T_c| ⩽ |T_2 − T_1|/2^{N+1}
end procedure
```

BIBLIOGRAPHY

1. J. A. Ouassou.
Full proximity effect in spin-textured superconductor/ferromagnet bilayers.
Specialization Project (Norwegian University of Science and Technology, 2014).
http://www.pvv.org/~jabirali/academic/master/project.pdf

2. S. H. Jacobsen, J. A. Ouassou & J. Linder.
Critical temperature and tunneling spectroscopy of superconductor/ferromagnet hybrids with intrinsic Rashba–Dresselhaus spin-orbit coupling.
Enclosed Preprint (2015).
ARXIV: 1503.06835 [cond-mat]

3. N. Hale & D. R. Moore.
Sixth-order extension to the Matlab package BVP4C.
Technical Report (Oxford University Computing Laboratory, 2008).
https://www.cs.ox.ac.uk/files/720/NA-08-04.pdf

4. S. K. Range.
Gravity Probe B: Exploring Einstein's universe with gyroscopes (NASA, 2008)
http://einstein.stanford.edu/content/education/GP-B_T-Guide4-2008.pdf

5. D. Martin, P. Verhoeve, A. Peacock, A. Dordrecht, J. Verveer & R. Hijmering.
A 12×10 pixels superconducting tunnel junction array based spectro-photometer for optical astronomy.
Nuclear Instruments and Methods in Physics Research A **520,** 512–515 (2004).
DOI: 10.1016/j.nima.2003.11.301

6. K. K. Likharev & V. K. Semenov.
RSFQ logic/memory family: A new Josephson-junction technology for sub-terahertz-clock-frequency digital systems.
IEEE Transactions on Applied Superconductivity **1,** 3–28 (1991).
DOI: 10.1109/77.80745

7. S. Nagasawa, Y. Hashimoto, H. Numata & S. Tahara.
A 380 ps, 9.5 mW Josephson 4-Kbit RAM operated at a high bit yield.
IEEE Transactions on Applied Superconductivity **5** (1995).
DOI: 10.1109/77.403086

8. M. Dorojevets, P. Bunyk & D. Zinoviev.
FLUX chip: Design of a 20 GHz 16-bit ultrapipelined RSFQ processor prototype based on 1.75 μm LTS technology.
IEEE Transactions on Applied Superconductivity **11,** 326–332 (2001).
DOI: 10.1109/77.919349

9. M. Tanaka, T. Kondo, N. Nakajima, T. Kawamoto, Y. Yamanashi, Y. Kamiya, A. Akimoto, A. Fujimaki, H. Hayakawa, N. Yoshikawa, H. Terai, Y. Hashimoto & S. Yorozu.
Demonstration of a single-flux-quantum microprocessor using passive transmission lines.
IEEE Transactions on Applied Superconductivity **15,** 400–404 (2005).
DOI: 10.1109/TASC.2005.849860

10. M. Johnson & R. H. Silsbee.
Interfacial charge-spin coupling: Injection and detection of spin magnetization in metals.
Physical Review Letters **55,** 1790–1793 (1985).
DOI: 10.1103/PhysRevLett.55.1790

11. D. C. Ralph & M. D. Stiles.
Spin transfer torques.
Journal of Magnetism and Magnetic Materials **320,** 1190–1216 (2008).
DOI: 10.1016/j.jmmm.2007.12.019

12. M. Baibich & J. Broto.
Giant magnetoresistance of (001)Fe/(001)Cr magnetic superlattices.
Physical Review Letters **61,** 2472–2475 (1988).
DOI: 10.1103/PhysRevLett.61.2472

13. G. Binasch, P. Grünberg, F. Saurenbach & W. Zinn.
Enhanced magnetoresistance in layered magnetic structures with antiferromagnetic interlayer exchange.
Physical Review B **39,** 4828–4830 (1989).
DOI: 10.1103/PhysRevB.39.4828

14. J. Linder & J. W. A. Robinson.
Superconducting spintronics.
Nature Physics **11,** 307–315 (2015).
DOI: 10.1038/nphys3242

15. M. Eschrig.
Spin-polarized supercurrents for spintronics.
Physics Today **64,** 43–49 (2011).
DOI: 10.1063/1.3541944

16. F. Bergeret, A. Volkov & K. Efetov.
Long-range proximity effects in superconductor-ferromagnet structures.
Physical Review Letters **86,** 4096–4099 (2001).
DOI: 10.1103/PhysRevLett.86.4096

17. F. Bergeret, A. Volkov & K. Efetov.
Odd triplet superconductivity and related phenomena in superconductor-ferromagnet structures.
Reviews of Modern Physics **77,** 1321–1373 (2005).
DOI: 10.1103/RevModPhys.77.1321

18. F. Bergeret & I. Tokatly.
Singlet-triplet conversion and the long-range proximity effect in superconductor-ferromagnet structures with generic spin dependent fields.
Physical Review Letters **110,** 117003 (2013).
DOI: 10.1103/PhysRevLett.110.117003

19. F. Bergeret & I. Tokatly.
Spin-orbit coupling as a source of long-range triplet proximity effect in superconductor-ferromagnet hybrid structures.
Physical Review B **89,** 134517 (2014).
DOI: 10.1103/PhysRevB.89.134517

20. M. Tinkham.
Introduction to superconductivity 2nd ed. (McGraw–Hill, 1996).
ISBN: 978-0-07-064878-4

21. K. Fossheim & A. Sudbø.
Superconductivity: Physics and applications 1st ed. (Wiley, 2004).
ISBN: 978-0-470-84452-6

22. J. Bardeen, L. N. Cooper & J. R. Schrieffer.
Theory of superconductivity.
Physical Review **108,** 1175–1204 (1957).
DOI: 10.1103/PhysRev.108.1175

23. J. Keeling.
Quantum magnetism.
Lecture Notes (University of St. Andrews, 2008).
http://www.st-andrews.ac.uk/~jmjk/keeling/teaching/magnetism-notes.pdf

24. Y. A. Bychkov & E. I. Rashba.
Oscillatory effects and the magnetic susceptibility of carriers in inversion layers.
Journal of Physics C **17,** 6039–6045 (1984).
DOI: 10.1088/0022-3719/17/33/015

25. B. A. Bernevig & S. C. Zhang.
Spin splitting and spin current in strained bulk semiconductors.
Physical Review B **72,** 115204 (2005).
DOI: 10.1103/PhysRevB.72.115204

26. M. D. Studer.
Spin manipulation in two-dimensional electron and hole gases.
PhD thesis (ETH Zürich, 2010).
http://e-collection.library.ethz.ch/eserv/eth:2507/eth-2507-02.pdf

27. E. Bernardes, J. Schliemann, M. Lee, J. C. Egues & D. Loss.
Spin-orbit interaction in symmetric wells with two subbands.
Physical Review Letters **99,** 076603 (2007).
DOI: 10.1103/PhysRevLett.99.076603

28. G. Dresselhaus.
Spin-orbit coupling effects in zinc blende structures.
Physical Review **100,** 580–586 (1955).
DOI: 10.1103/PhysRev.100.580

29. K. V. Samokhin.
Spin-orbit coupling and semiclassical electron dynamics in noncentrosymmetric metals.
Annals of Physics **324,** 2385–2407 (2009).
DOI: 10.1016/j.aop.2009.08.008

30. T. Kita.
Introduction to nonequilibrium statistical mechanics with quantum field theory.
Progress of Theoretical Physics **123,** 581–658 (2010).
DOI: 10.1143/PTP.123.581

31. J. Rammer.
Quantum field-theoretical methods in transport theory of metals.
Reviews of Modern Physics **58,** 323–359 (1986).
DOI: 10.1103/RevModPhys.58.323

32. L. V. Keldysh.
Diagram technique for nonequilibrium processes.
Soviet Physics JETP **20,** 1018–1026 (1965)

33. G. D. Mahan.
Many-particle physics 3rd ed. (Springer, New York, 2000).
ISBN: 978-0-306-46338-9

34. T. Matsubara.
A new approach to quantum statistical mechanics.
Progress of Theoretical Physics **14,** 351–378 (1955)

35. V. Chandrasekhar.
Introduction to the quasiclassical theory of superconductivity for diffusive proximity-coupled systems.
Online Preprint (2004).
ARXIV: 0312507 [cond-mat]

36. W. Belzig, F. K. Wilhelm, C. Bruder, G. Schön & A. D. Zaikin.
Quasiclassical Green's function approach to mesoscopic superconductivity.
Superlattices and Microstructures **25,** 1251–1288 (1999).
DOI: 10.1006/spmi.1999.0710

37. J. P. Morten.
Spin and charge transport in dirty superconductors.
Diploma Thesis (Norwegian University of Science and Technology, 2005)

38. N. Schopohl.
Transformation of the Eilenberger equations of superconductivity to a scalar Riccati equation.
Online Preprint (1998).
ARXIV: 9804064 [cond-mat]

39. A. P. Mackenzie & Y. Maeno.
The superconductivity of Sr_2RuO_4 and the physics of spin-triplet pairing.
Reviews of Modern Physics **75,** 657–712 (2003).
DOI: 10.1103/RevModPhys.75.657

40. R. Balian & N. R. Werthamer.
Superconductivity with pairs in a relative p-wave.
Physical Review **131,** 1553–1564 (1963).
DOI: 10.1103/PhysRev.131.1553

41. Y. V. Nazarov.
Novel circuit theory of Andreev reflection.
Superlattices and Microstructures **25,** 1221–1231 (1999).
DOI: 10.1006/spmi.1999.0738

42. Y. V. Nazarov & Y. M. Blanter.
Quantum transport 1st ed. (Cambridge University Press, Cambridge, England, 2009).
ISBN: 978-0-521-83246-5

43. I. Gomperud & J. Linder.
Spin-supercurrent and phase-tunable triplet Cooper pairs via magnetic insulators.
Online Preprint (2015).
ARXIV: 1503.08229v1 [cond-mat]

44. P. Machon, M. Eschrig & W. Belzig.
Nonlocal thermoelectric effects and nonlocal onsager relations in a three-terminal proximity-coupled superconductor-ferromagnet device.
Physical Review Letters **110,** 047002 (2013).
DOI: 10.1103/PhysRevLett.110.047002

45. P. Machon, M. Eschrig & W. Belzig.
Supplemental: Nonlocal thermoelectric effects and nonlocal onsager relations in a three-terminal proximity-coupled superconductor-ferromagnet device.
Physical Review Letters **110,** 1–5 (2013).
DOI: 10.1103/PhysRevLett.110.047002

46. K. D. Usadel.
Generalized diffusion equation for superconducting alloys.
Physical Review Letters **25,** 507–509 (1970).
DOI: 10.1103/PhysRevLett.25.507

47. I. A. N. Kellett.
Circuit theory of mesoscopic superconducting components.
Master Thesis (Chalmers University of Technology, 2011)

48. J. Arjoranta & K. Rummukainen.
Spin-orbit coupling in superconductor-normal metal-superconductor junctions.
Master Thesis (University of Helsinki, 2014)

49. Y. J. Doh, J. A. van Dam, A. L. Roest, E. P. A. M. Bakkers, L. P. Kouwenhoven & S. De Franceschi.
Tunable supercurrent through semiconductor nanowires.
Science **309,** 272–275 (2005).
DOI: 10.1126/science.1113523

50. M. Jung, H. Noh, Y. J. Doh, W. Song, Y. Chong, M. S. Choi, Y. Yoo, K. Seo, N. Kim, B. C. Woo, B. Kim & J. Kim.
Superconducting junction of a single-crystalline Au nanowire for an ideal Josephson device.
ACS Nano **5,** 2271–2276 (2011).
DOI: 10.1021/nn1035679

51. V. Cherkez, J. C. Cuevas, C. Brun, T. Cren, F. Debontridder, V. Stolyarov & D. Roditchev.
Proximity effect between two superconductors spatially resolved by scanning tunneling spectroscopy.
Physical Review X **4,** 011033 (2014).
DOI: 10.1103/PhysRevX.4.011033

52. G. H. Lee, S. Kim, S. H. Jhi & H. J. Lee.
Ultimately short ballistic vertical graphene Josephson junctions.
Nature Communications **6,** 6181 (2015).
DOI: 10.1038/ncomms7181

53. M. Y. Kuprianov & V. F. Lukichev.
Influence of boundary transparency on the critical current of dirty SS'S structures.
Soviet Physics JETP **67,** 1163–1168 (1988)

54. M. Eschrig, A. Cottet, W. Belzig & J. Linder.
General boundary conditions for quasiclassical theory of superconductivity in the diffusive limit: application to strongly spin-polarized systems.
Online Preprint (2015).
ARXIV: 1504.06258

55. A. Cottet.
Spectroscopy and critical temperature of diffusive superconducting/ferromagnetic hybrid structures with spin-active interfaces.
Physical Review B **76,** 224505 (2007).
DOI: 10.1103/PhysRevB.76.224505

56. A. Cottet, D. Huertas-Hernando, W. Belzig & Y. V. Nazarov.
Spin-dependent boundary conditions for isotropic superconducting Green's functions.
Physical Review B **80,** 184511 (2009).
DOI: 10.1103/PhysRevB.80.184511

57. A. Cottet, D. Huertas-Hernando, W. Belzig & Y. Nazarov.
Erratum: Spin-dependent boundary conditions for isotropic superconducting Green's functions.
Physical Review B **83,** 184511 (2011).
DOI: 10.1103/PhysRevB.83.139901

58. R. Fazio & C. Lucheroni.
Local density of states in superconductor-ferromagnetic hybrid systems.
Europhysics Letters **45,** 707–713 (1998).
DOI: 10.1209/epl/i1999-00225-3

59. I. Giaever & K. Megerle.
Study of superconductors by electron tunneling.
Physical Review **122,** 1101–1111 (1961).
DOI: 10.1103/PhysRev.122.1101

60. P. Townsend & J. Sutton.
Investigation by Electron Tunneling of the Superconducting Energy Gaps in Nb, Ta, Sn, and Pb.
Physical Review **128,** 591–595 (1962)

61. N. W. Ashcroft & N. D. Mermin.
Solid state physics 1st ed. (Saunders College, 1976).
ISBN: 978-0-03-083993-1

62. I. Giaever.
Energy gap in superconductors measured by electron tunneling.
Physical Review Letters **5,** 147–148 (1960).
DOI: 10.1103/PhysRevLett.5.147

63. H. Pothier, S. Guéron, N. O. Birge, D. Esteve & M. H. Devoret.
The superconducting proximity effect probed on a mesoscopic length scale.
Czechoslovak Journal of Physics **46,** 2319–2320 (1996).
DOI: 10.1007/BF02571151

64. N. Moussy, H. Courtois & B. Pannetier.
Local spectroscopy of a proximity superconductor at very low temperature.
Europhysics Letters **55,** 861 (2001).
DOI: 10.1209/epl/i2001-00361-2

65. H. Le Sueur, P. Joyez, H. Pothier, C. Urbina & D. Esteve.
Phase controlled superconducting proximity effect probed by tunneling spectroscopy.
Physical Review Letters **100,** 197002 (2008).
DOI: 10.1103/PhysRevLett.100.197002

66. A. F. Andreev.
The thermal conductivity of the intermediate state in superconductors.
Soviet Physics JETP **19,** 1228–1231 (1964)

67. G. E. Blonder, M. Tinkham & T. M. Klapwijk.
Transition from metallic to tunneling regimes in superconducting microconstrictions: Excess current, charge imbalance, and supercurrent conversion.
Physical Review B **25,** 4515–4532 (1982).
DOI: 10.1103/PhysRevB.25.4515

68. M. Octavio, M. Tinkham, G. E. Blonder & T. M. Klapwijk.
Subharmonic energy-gap structure in superconducting constrictions.
Physical Review B **27,** 6739–6746 (1983).
DOI: 10.1103/PhysRevB.27.6739

69. A. I. Buzdin.
Proximity effects in superconductor-ferromagnet heterostructures.
Reviews of Modern Physics **77,** 935–976 (2005).
DOI: 10.1103/RevModPhys.77.935

70. A. A. Golubov, M. Y. Kuprianov & E. Il'ichev.
The current-phase relation in Josephson junctions.
Reviews of Modern Physics **76,** 411–469 (2004).
DOI: 10.1103/RevModPhys.76.411

71. E. A. Demler, G. B. Arnold & M. R. Beasley.
Superconducting proximity effects in magnetic metals.
Physical Review B **55,** 15174–15182 (1997).
DOI: 10.1103/PhysRevB.55.15174

72. M. Eschrig & T. Löfwander.
Triplet supercurrents in clean and disordered half-metallic ferromagnets.
Nature Physics **4,** 138–143 (2008).
DOI: 10.1038/nphys831

73. M. J. M. De Jong & C. W. J. Beenakker.
Andreev reflection in ferromagnet-superconductor junctions.
Physical Review Letters **74,** 1657–1660 (1995).
DOI: 10.1103/PhysRevLett.74.1657

74. E. Goldobin, D. Koelle, R. Kleiner & R. G. Mints.
Josephson junction with a magnetic-field tunable ground state.
Physical Review Letters **107,** 227001 (2011).
DOI: 10.1103/PhysRevLett.107.227001

75. C. Visani, Z. Sefrioui, J. Tornos, C. Leon, J. Briatico, M. Bibes, A. Barthélémy, J. Santamaría & J. E. Villegas.
Equal-spin Andreev reflection and long-range coherent transport in high-temperature superconductor/half-metallic ferromagnet junctions.
Nature Physics **8,** 539–543 (2012).
DOI: 10.1038/nphys2318

76. S. Kawabata, Y. Asano, Y. Tanaka & A. A. Golubov.
Robustness of Spin-Triplet Pairing and Singlet–Triplet Pairing Crossover in Superconductor/Ferromagnet Hybrids.
Journal of the Physical Society of Japan **82,** 124702 (2013).
DOI: 10.7566/JPSJ.82.124702

77. J. Linder, T. Yokoyama, A. Sudbø & M. Eschrig.
Pairing symmetry conversion by spin-active interfaces in magnetic normal-metal-superconductor junctions.
Physical Review Letters **102,** 107008 (2009).
DOI: 10.1103/PhysRevLett.102.107008

78. J. Linder, T. Yokoyama & A. Sudbø.
Theory of superconducting and magnetic proximity effect in S/F structures with inhomogeneous magnetization textures and spin-active interfaces.
Physical Review B **79,** 054523 (2009).
DOI: 10.1103/PhysRevB.79.054523

79. B. Josephson.
Possible new effects in superconductive tunnelling.
Physics Letters **1,** 251–253 (1962).
DOI: 10.1016/0031-9163(62)91369-0

80. A. Buzdin.
Direct coupling between magnetism and superconducting current in the Josephson φ_0 junction.
Physical Review Letters **101,** 107005 (2008).
DOI: 10.1103/PhysRevLett.101.107005

81. F. S. Bergeret & I. V. Tokatly.
Theory of diffusive φ_0 Josephson junctions in the presence of spin-orbit coupling.
Online Preprint (2014).
ARXIV: 1409.4563v1

82. V. Ryazanov, V. Oboznov, A. Rusanov, A. Veretennikov, A. Golubov & J. Aarts.
Coupling of two superconductors through a ferromagnet: evidence for a π junction.
Physical Review Letters **86,** 2427–2430 (2001).
DOI: 10.1103/PhysRevLett.86.2427

83. H. Sickinger, A. Lipman, M. Weides, R. G. Mints, H. Kohlstedt, D. Koelle, R. Kleiner & E. Goldobin.
Experimental evidence of a φ Josephson junction.
Physical Review Letters **109,** 107002 (2012).
DOI: 10.1103/PhysRevLett.109.107002

84. S. Jacobsen & J. Linder.
Giant triplet proximity effect in π-biased Josephson junctions with spin-orbit coupling.
Online Preprint (2015).
ARXIV: 1503.03500v1

85. A. K. Feofanov, V. A. Oboznov, V. V. Bol'ginov, J. Lisenfeld, S. Poletto, V. V. Ryazanov, A. N. Rossolenko, M. Khabipov, D. Balashov, A. B. Zorin, P. N. Dmitriev, V. P. Koshelets & A. V. Ustinov.
Implementation of superconductor/ferromagnet/superconductor π-shifters in superconducting digital and quantum circuits.
Nature Physics **6,** 593–597 (2010).
DOI: 10.1038/nphys1700

86. D. Huertas-Hernando, Y. V. Nazarov & W. Belzig.
Absolute spin-valve effect with superconducting proximity structures.
Physical review letters **88,** 047003 (2002).
DOI: 10.1103/PhysRevLett.88.047003

87. Y. V. Fominov, a. a. Golubov, T. Y. Karminskaya, M. Y. Kupriyanov, R. G. Deminov & L. R. Tagirov.
Superconducting triplet spin valve.
JETP Letters **91,** 308–313 (2010).
DOI: 10.1134/S002136401006010X

88. J. Y. Gu, C. Y. You, J. S. Jiang, J. Pearson, Y. B. Bazaliy & S. D. Bader.
Magnetization-orientation dependence of the superconducting transition temperature in the ferromagnet-superconductor-ferromagnet system: CuNi/Nb/CuNi.
Physical review letters **89,** 267001 (2002).
DOI: 10.1103/PhysRevLett.89.267001

89. I. C. Moraru, W. P. Pratt & N. O. Birge.
Magnetization-dependent Tc shift in ferromagnet/superconductor/ferromagnet trilayers with a strong ferromagnet.
Physical Review Letters **96,** 037004 (2006).
DOI: 10.1103/PhysRevLett.96.037004

90. J. Zhu, I. N. Krivorotov, K. Halterman & O. T. Valls.
Angular dependence of the superconducting transition temperature in ferromagnet-superconductor-ferromagnet trilayers.
Physical Review Letters **105,** 207002 (2010).
DOI: 10.1103/PhysRevLett.105.207002

91. P. V. Leksin, N. N. Garif'Yanov, I. a. Garifullin, Y. V. Fominov, J. Schumann, Y. Krupskaya, V. Kataev, O. G. Schmidt & B. Büchner.
Evidence for triplet superconductivity in a superconductor-ferromagnet spin valve.
Physical Review Letters **109,** 057005 (2012).
DOI: 10.1103/PhysRevLett.109.057005

92. A. Singh, S. Voltan, K. Lahabi & J. Aarts.
Colossal proximity effect in a superconducting triplet spin valve based on halfmetallic ferromagnetic CrO_2.
Online Preprint (2014).
ARXIV: 1410.4973v1

93. N. Banerjee, C. B. Smiet, R. G. J. Smits, A. Ozaeta, F. S. Bergeret, M. G. Blamire & J. W. A. Robinson.
Evidence for spin selectivity of triplet pairs in superconducting spin valves.
Nature Communications **5,** 3048 (2014).
DOI: 10.1038/ncomms4048

94. Y. Fominov, N. Chtchelkatchev & A. Golubov.
Nonmonotonic critical temperature in superconductor/ferromagnet bilayers.
Physical Review B **66,** 014507 (2002).
DOI: 10.1103/PhysRevB.66.014507

95. W. L. McMillan.
Transition temperature of strong-coupled superconductors.
Physical Review **167,** 331–344 (1968).
DOI: 10.1103/PhysRev.167.331

96. J. Kwo & T. Geballe.
Superconducting tunneling into the A15 Nb_3Al thin films.
Physical Review B **23,** 3230–3239 (1981).
DOI: 10.1103/PhysRevB.23.3230

97. C. Cirillo, S. L. Prischepa, M. Salvato, C. Attanasio, M. Hesselberth & J. Aarts.
Superconducting proximity effect and interface transparency in Nb/PdNi bilayers.
Physical Review B **72,** 1–7 (2005).
DOI: 10.1103/PhysRevB.72.144511

98. C. T. Wu, O. T. Valls & K. Halterman.
Reentrant superconducting phase in conical-ferromagnet-superconductor nanostructures.
Physical Review Letters **108,** 117005 (2012).
DOI: 10.1103/PhysRevLett.108.117005

Critical Temperature and Tunneling Spectroscopy of Superconductor/Ferromagnet Hybrids with Intrinsic Rashba–Dresselhaus Spin-Orbit Coupling

Sol H. Jacobsen,[1*] Jabir Ali Ouassou,[1*] and Jacob Linder[1]

[1]*Department of Physics, Norwegian University of Science and Technology, N-7491 Trondheim, Norway*
[*]*These authors contributed equally to this work.*

We investigate theoretically how the proximity effect in superconductor/ferromagnet hybrid structures with intrinsic spin-orbit coupling manifests in two measurable quantities, namely the density of states and critical temperature. To describe a general scenario, we allow for both Rashba and Dresselhaus type spin-orbit coupling. Our results are obtained via the quasiclassical theory of superconductivity, extended to include spin-orbit coupling in the Usadel equation and Kupriyanov–Lukichev boundary conditions. Unlike previous works, we have derived a Riccati parametrization of the Usadel equation with spin-orbit coupling which allows us to address the full proximity regime and not only the linearized weak proximity regime. First, we consider the density of states in both SF bilayers and SFS trilayers, where the spectroscopic features in the latter case are sensitive to the phase difference between the two superconductors. We find that the presence of spin-orbit coupling leaves clear spectroscopic fingerprints in the density of states due to its role in creating spin-triplet Cooper pairs. Unlike SF and SFS structures without spin-orbit coupling, the density of states in the present case depends strongly on the direction of magnetization. Moreover, we show that the spin-orbit coupling can stabilize spin-singlet superconductivity even in the presence of a strong exchange field $h \gg \Delta$. This leads to the possibility of a magnetically tunable minigap: changing the direction of the exchange field opens and closes the minigap. We also determine how the critical temperature T_c of an SF bilayer is affected by spin-orbit coupling and, interestingly, demonstrate that one can achieve a spin-valve effect with a single ferromagnet. We find that T_c displays highly non-monotonic behavior both as a function of the magnetization direction and the type and direction of the spin-orbit coupling, offering a new way to exert control over the superconductivity of proximity structures.

I. INTRODUCTION

Material interfaces in hybrid structures give rise to proximity effects, whereby the properties of one material can "leak" into the adjacent material, creating a region with properties derived from both materials. In superconductor/ferromagnet (SF) hybrid structures[1], the proximity effect causes superconducting correlations to penetrate into the ferromagnetic region and vice versa. These correlations typically decay over short distances, which in diffusive systems is of the order $\sqrt{D/h}$, where D is the diffusion coefficient of the ferromagnet and h is the strength of the exchange field. However, for certain field configurations, the singlet correlations from the superconductor may be converted into so-called long-range triplets (LRTs)[2]. These triplet components have spin projection parallel to the exchange field, and decay over much longer distances. This results in physical quantities like supercurrents decaying over the large length scale $\sqrt{D/T} \gg \sqrt{D/h}$, where T is the temperature. This distance is independent of h, and at low temperatures it becomes increasingly large, which allows the condensate to penetrate deep into the ferromagnet. The isolation and enhancement of this feature has attracted much attention in recent years as it gives rise to novel physics and possible low-temperature applications by merging spintronics and superconductivity[3].

It is by now well-known that the conversion from singlet to long-range triplet components of the superconducting state can happen in the presence of magnetic inhomogeneities[4,5], *i.e.* a spatially varying exchange field, and until recently such inhomogeneities were believed to be the primary source of this conversion[6–15], although other proposals using *e.g.* non-equilibrium distribution functions and intrinsic triplet superconductors also exist[16–19]. However, it has recently been established that another possible source of LRT correlations is the presence of a finite spin-orbit (SO) coupling, either in the superconducting region[20] or on the ferromagnetic side[21,22]. In fact, it can be shown that an SF structure where the magnetic inhomogeneity is due to a Bloch domain wall, as considered in *e.g.* Refs. 50–52, is gauge equivalent to one where the ferromagnet has a homogeneous exchange field and intrinsic SO coupling[21]. It is known that SO scattering can be caused by impurities[23], but this type of scattering results in purely isotropic spin-relaxation, and so does not permit the desired singlet-LRT conversion. To achieve such a conversion, one needs a rotation of the spin pair into the direction of the exchange field[24]. This can be achieved by using materials with an intrinsic SO coupling, either due to the crystal structure in the case of noncentrosymmetric materials[25], or due to interfaces in thin-film hybrids[26], where the latter also modifies the fundamental process of Andreev reflection[27,28].

In this paper, we establish how the presence of spin-orbit coupling in SF structures manifests in two important experimental observables: the density of states $D(\varepsilon)$ probed via tunneling spectroscopy (or conductance measurements), and the critical temperature T_c. A common consequence for both of these quantities is that neither becomes independent of the magnetization direction. This is in contrast to the case without SO coupling in conventional monodomain ferromagnets, where the results are invariant with respect to rotations of the magnetic exchange field. This symmetry is now lifted due to SO coupling: depending on the magnetization direction, LRT Cooper pairs are created in the system which leave clear fingerprints both spectroscopically and in terms of the T_c behavior. On the technical side, we will present in this work for the first time a Riccati parametrization of the Usadel equation and its corresponding boundary conditions that include SO coupling. This is an important advance in terms of exploring the full

physics of triplet pairing due to SO coupling as it allows for a solution of the quasiclassical equations without any assumption of a weak proximity effect. We will also demonstrate that the SO coupling can actually protect the singlet superconducting correlations even in the presence of a strong exchange field, leading to the possibility of a minigap that is magnetically tunable via the orientation of the exchange field.

The remainder of the article will be organised as follows: In Section II, we introduce the relevant theory and notation, starting from the quasiclassical Usadel equation, which describes the diffusion of the superconducting condensate into the ferromagnet. We also motivate our choice of intrinsic SO coupling in this section, and propose a new notation for describing Rashba–Dresselhaus couplings. The section goes on to discuss key analytic features of the equations in the limit of weak proximity, symmetries of the density of states at zero energy, and analytical results needed to calculate the critical temperature of hybrid systems. We then present detailed numerical results in Section III: we analyze the density of states of an SF bilayer in III A (see Fig. 1), and of an SFS Josephson junction in III B (see Fig. 2). We study different orientations and strengths of the exchange field and SO coupling, and in the case of the Josephson junction, the effect of altering the phase difference between the condensates. Then, in Section III C, we continue our treatment of the SF bilayer in the full proximity regime by including a self-consistent solution in the superconducting layer, and focus on how the presence of SO coupling affects the critical temperature of the system. We discover that the SO coupling allows for spin-valve functionality with a single ferromagnetic layer, meaning that rotating the magnetic field by $\pi/2$ induces a large change in T_c. Finally, we conclude in Section IV with a summary of the main results, a discussion of some additional consequences of the choices made in-text, as well as possibilities for further work.

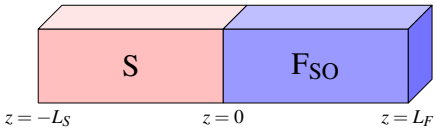


FIG. 1: The SF bilayer in III A and III C. We take the thin-film layering direction along the z-axis, and assume an xy-plane Rashba–Dresselhaus coupling in the ferromagnetic layer.

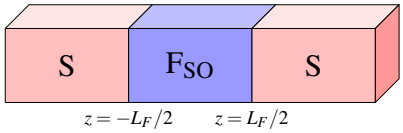


FIG. 2: The SFS trilayer in III B. We take the thin-film layering direction along the z-axis, and assume an xy-plane Rashba–Dresselhaus coupling in the ferromagnetic layer.

II. THEORY

A. Fundamental concepts

The diffusion of the superconducting condensate into the ferromagnet can be described by the Usadel equation, which is a second-order partial differential equation for the Green's function of the system[29]. Together with appropriate boundary conditions, the Usadel equation establishes a system of coupled differential equations that can be solved in one dimension. We will consider the case of diffusive equilibrium, where the retarded component $\hat{g}^R$ of the Green's function is sufficient to describe the behaviour of the system[30,31]. We start by examining the superconducting correlations in the ferromagnet, and use the standard Bardeen–Cooper–Schrieffer (BCS) bulk solution for the superconductors. In particular, we will clarify the spectroscopic consequences of having SO coupling in the ferromagnetic layer.

In the absence of SO coupling, the Usadel equation[29] in the ferromagnet reads

$$D_F \nabla(\hat{g}^R \nabla \hat{g}^R) + i\left[\varepsilon\hat{\rho}_3 + \hat{M}, \hat{g}^R\right] = 0, \tag{1}$$

where the matrix $\hat{\rho}_3 = \mathrm{diag}(1,-1)$, and ε is the quasiparticle energy. The magnetization matrix $\hat{M}$ in the above equation is

$$\hat{M} = \begin{pmatrix} \underline{h}\cdot\underline{\sigma} & 0 \\ 0 & (\underline{h}\cdot\underline{\sigma})^* \end{pmatrix},$$

where $\underline{h} = (h_x, h_y, h_z)$ is the ferromagnetic exchange field, $(^*)$ denotes complex conjugation, $\underline{\sigma} = (\sigma_x, \sigma_y, \sigma_z)$ is the Pauli vector, and σ_k are the usual Pauli matrices. The corresponding Kupriyanov–Lukichev boundary conditions are[32]

$$2L_j\zeta_j \hat{g}_j^R \nabla \hat{g}_j^R = [\hat{g}_1^R, \hat{g}_2^R], \tag{2}$$

where the subscripts refer to the different regions of the hybrid structure; in the case of an SF bilayer as depicted in Fig. 1, $j = 1$ denotes the superconductor, and $j = 2$ the ferromagnet, while ∇ denotes the derivative along the junction $1 \to 2$. The respective lengths of the materials are denoted L_j, and the interface parameters $\zeta_j = R_B/R_j$ describe the ratio of the barrier resistance R_B to the bulk resistance R_j of each material.

We will use the Riccati parameterisation[33] for the quasiclassical Green's function $\hat{g}^R$,

$$\hat{g}^R = \begin{pmatrix} N(1+\gamma\tilde{\gamma}) & 2N\gamma \\ -2\tilde{N}\tilde{\gamma} & -\tilde{N}(1+\tilde{\gamma}\gamma) \end{pmatrix}, \tag{3}$$

where the normalisation matrices are $N = (1-\gamma\tilde{\gamma})^{-1}$ and $\tilde{N} = (1-\tilde{\gamma}\gamma)^{-1}$. The tilde operation denotes a combination of complex conjugation $i \to -i$ and energy $\varepsilon \to -\varepsilon$. The Riccati parameterisation is particularly useful for numerical computation because the parameters are bounded, contrary to the multi-valued θ-parameterisation[30]. In practice, this means that for certain parameter choices the numerical routines will converge in the Riccati parametrization but not the θ-parametrization.

To include intrinsic SO coupling in the Usadel equation, we simply have to replace all the derivatives in Eq. (1) with their gauge covariant counterparts:[21,35]

$$\nabla(\cdot) \mapsto \tilde{\nabla}(\cdot) \equiv \nabla(\cdot) - i[\hat{\underline{A}}, \cdot]. \tag{4}$$

This is valid for any SO coupling linear in momentum. We consider the leading contribution; higher order terms, *e.g.* those responsible for the SU(2) Lorentz force, are neglected here. Such higher order terms are required to produce so-called φ_0 junctions which have attracted interest of late[36], and consequently we will see no signature of the φ_0 effect in the systems considered herein. The object $\hat{\underline{A}}$ has both a vector structure in geometric space, and a 4×4 matrix structure in Spin–Nambu space, and can be written as $\hat{\underline{A}} = \mathrm{diag}(\underline{A}, -\underline{A}^*)$ in terms of the SO field $\underline{A} = (A_x, A_y, A_z)$, which will be discussed in more detail in the next subsection. SO coupling in the context of quasiclassical theory has also been discussed in Refs. 34,35. When we include a SO coupling in this way, we derive the following form for the Riccati-parametrized Usadel equation:

$$\begin{aligned} D_F\left(\partial_k^2\gamma + 2(\partial_k\gamma)\tilde{N}\tilde{\gamma}(\partial_k\gamma)\right) &= -2i\varepsilon\gamma - i\underline{h}\cdot(\underline{\sigma}\gamma - \gamma\underline{\sigma}^*) \\ &\quad + D_F\left[\underline{A}\underline{A}\gamma - \gamma\underline{A}^*\underline{A}^* + 2(\underline{A}\gamma + \gamma\underline{A}^*)\tilde{N}(\underline{A}^* + \tilde{\gamma}\underline{A}\gamma)\right] \\ &\quad + 2iD_F\left[(\partial_k\gamma)\tilde{N}(A_k^* + \tilde{\gamma}A_k\gamma) + (A_k + \gamma A_k^*\tilde{\gamma})N(\partial_k\gamma)\right], \end{aligned} \tag{5}$$

where the index k indicates an arbitrary choice of direction in Cartesian coordinates. The corresponding equation for $\tilde{\gamma}$ is found by taking the tilde conjugate of Eq. (5). Similarly, the boundary conditions in Eq. (2) become:

$$\begin{aligned} \partial_k\gamma_1 &= \frac{1}{L_1\zeta_1}(1-\gamma_1\tilde{\gamma}_2)N_2(\gamma_2-\gamma_1) + iA_k\gamma_1 + i\gamma_1A_k^*, \\ \partial_k\gamma_2 &= \frac{1}{L_2\zeta_2}(1-\gamma_2\tilde{\gamma}_1)N_1(\gamma_2-\gamma_1) + iA_k\gamma_2 + i\gamma_2A_k^*, \end{aligned} \tag{6}$$

and the $\tilde{\gamma}$ counterparts are found in the same way as before. For the details of these derivations, see Appendix A.

We will now discuss the definition of current in the presence of spin-orbit interactions. Since the Hamiltonian including SO coupling contains terms linear in momentum (see below), the velocity operator $v_j = \partial H/\partial k_j$ is affected. We stated above that the Kupriyanov–Lukichev boundary conditions are simply modified by replacing the derivative with its gauge covariant counterpart including the SO interaction. To make sure that current conservation is still satisfied, we must carefully examine the Usadel equation. In the absence of SO coupling, the quasiclassical expression for electric current is given by

$$\underline{I}_e = I_0\int_{-\infty}^{\infty} \mathrm{d}\varepsilon\,\mathrm{Tr}\{\rho_3(\check{g}\nabla\check{g})^K\}, \tag{7}$$

where $\check{g}$ is the 8×8 Green's function matrix in Keldysh space

$$\check{g} = \begin{pmatrix} \hat{g}^R & \hat{g}^K \\ \hat{0} & \hat{g}^A \end{pmatrix}, \tag{8}$$

and I_0 is a constant that is not important for this discussion. Current conservation can now be proven from the Usadel equation itself. We show this for the case of equilibrium, which is relevant for the case of supercurrents in Josephson junctions. In this case $\hat{g}^K = (\hat{g}^R - \hat{g}^A)\tanh(\beta\varepsilon/2)$ and we get

$$\underline{I}_e = I_0\int_{-\infty}^{\infty} \mathrm{d}\varepsilon\,\mathrm{Tr}\{\rho_3(\hat{g}^R\nabla\hat{g}^R - \hat{g}^A\nabla\hat{g}^A)\}\tanh(\beta\varepsilon/2). \tag{9}$$

Performing the operation $\mathrm{Tr}\{\rho_3\cdots\}$ on the Usadel equation, we obtain

$$D\nabla\cdot\mathrm{Tr}\{\rho_3(\hat{g}^R\nabla\hat{g}^R) + i\,\mathrm{Tr}\{\rho_3[\varepsilon\rho_3 + \hat{M}, \hat{g}^R]\} = 0. \tag{10}$$

Now, inserting the most general definition of the Green's function $\hat{g}^R$, one finds that the second term in the above equation is zero. Thus, we are left with

$$\nabla\cdot\mathrm{Tr}\{\rho_3(\hat{g}^R\nabla\hat{g}^R)\} = 0, \tag{11}$$

which expresses precisely current conservation since the same analysis can be done for $\hat{g}^A$. Now, let us include the SO coupling. The current should then be given by

$$\underline{I}_e = I_0\int_{-\infty}^{\infty} \mathrm{d}\varepsilon\,\mathrm{Tr}\{\rho_3(\check{g}\tilde{\nabla}\check{g})^K\}, \tag{12}$$

so that the expression for the charge current is modified by the presence of SO coupling, as is known. The question is now if this current is conserved, as it has to be physically. We can prove that it is from the Usadel equation by rewriting it as

$$\begin{aligned} &D\nabla\cdot(\hat{g}^R\tilde{\nabla}\hat{g}^R) \\ &\quad = D[\underline{A}, \hat{g}^R\nabla\hat{g}^R] + D[\underline{A}, [\underline{A}, \hat{g}^R]] - i[\varepsilon\rho_3 + \hat{M}, \hat{g}^R], \end{aligned} \tag{13}$$

and then performing the operation $\mathrm{Tr}\{\rho_3\cdots\}$,

$$D\nabla\cdot\mathrm{Tr}\{\rho_3(\hat{g}^R\tilde{\nabla}\hat{g}^R)\} = 0, \tag{14}$$

so we recover the standard current conservation law $\nabla\cdot\underline{I}_e = 0$.

B. Spin-orbit field

The precise form of the generic SO field $\underline{A}$ is imposed by the experimental requirements and limitations. As the name suggests, spin-orbit coupling couples a particle's spin with its motion, and more specifically its momentum. As mentioned in the Introduction, the SO coupling in solids can originate from a lack of inversion symmetry in the crystal structure. Such spin-orbit coupling can be of both Rashba and Dresselhaus type and is determined by the point group symmetry of the crystal[67,68]. It is also known that the lack of inversion symmetry due to surfaces, either in the form of interfaces to other materials or to vacuum, will give rise to antisymmetric spin-orbit coupling of the Rashba type. For sufficiently thin structures, the SO coupling generated in this way can permeate the entire structure. Intrinsic inversion asymmetry arises naturally due to interfaces between materials in thin-film hybrid structures such as the ones considered herein. Noncentrosymmetric crystalline structures provide an alternative source for intrinsic asymmetry, and are considered in Ref. 43. In thin-film hybrids, the Rashba spin

splitting derives from the cross product of the Pauli vector $\underline{\sigma}$ with the momentum $\underline{k}$,

$$H_R = -\frac{\alpha}{m}(\underline{\sigma}\times\underline{k})\cdot\hat{\underline{z}}, \tag{15}$$

where α is called the Rashba coefficient, and we have chosen a coordinate system with $\hat{\underline{z}}$ as the layering direction. Another well-known type of SO coupling is the Dresselhaus spin splitting, which can occur when the crystal structure lacks an inversion centre. For a two-dimensional electron gas (quantum well) confined in the $\hat{\underline{z}}$-direction, then to first order $\langle k_z\rangle = 0$, so the Dresselhaus splitting becomes

$$H_D = \frac{\beta}{m}(\sigma_y k_y - \sigma_x k_x), \tag{16}$$

where β is called the Dresselhaus coefficient. In our structure, we consider a thin-film geometry with the confinement being strongest in the z-direction. Although there may certainly be other terms contributing to the Dresselhaus SO coupling in such a structure, we consider the standard form Eq. (16) as an approximation that captures the main physics in the problem. This is a commonly used model in the literature to explore the effects originating from SO coupling in a system. When we combine both interactions, we obtain the Hamiltonian for a general Rashba–Dresselhaus SO coupling,

$$H_{RD} = \frac{k_x}{m}(\alpha\sigma_y - \beta\sigma_x) - \frac{k_y}{m}(\alpha\sigma_x - \beta\sigma_y). \tag{17}$$

In this work, we will restrict ourselves to this experimentally common form of SO coupling. As explained in Ref. 21, the SO coupling acts as a background SU(2) field, *i.e.* an object with both a vector structure in geometric space, and a 2×2 matrix structure in spin space. We can therefore identify the interaction above with an effective vector potential $\underline{A}$ which we will call the *SO field*,

$$H_{RD} \equiv -\underline{k}\cdot\underline{A}/m, \tag{18}$$

from which we derive that

$$\underline{A} = (\beta\sigma_x - \alpha\sigma_y, \alpha\sigma_x - \beta\sigma_y, 0). \tag{19}$$

At this point, it is convenient to introduce a new notation for describing Rashba–Dresselhaus couplings, which will let us distinguish between the physical effects that derive from the strength of the coupling, and those that derive from the geometry. For this purpose, we employ polar notation defined by the relations

$$\begin{aligned}\alpha &\equiv -a\sin\chi,\\ \beta &\equiv \ \ a\cos\chi,\end{aligned} \tag{20}$$

where we will refer to a as the *SO strength*, and χ as the *SO angle*. Rewritten in the polar notation, Eq. (19) takes the form:

$$\underline{A} = a(\sigma_x\cos\chi + \sigma_y\sin\chi)\hat{\underline{x}} - a(\sigma_x\sin\chi + \sigma_y\cos\chi)\hat{\underline{y}}. \tag{21}$$

From the definition, we can immediately conclude that $\chi = 0$ corresponds to a pure Dresselhaus coupling, while $\chi = \pm\pi/2$ results in a pure Rashba coupling, with the geometric interpretation of χ illustrated in Fig. 3. Note that $A_x^2 = A_y^2 = a^2$, which means that $\underline{A}^2 = 2a^2$. Another useful property is that we can switch the components $A_x \leftrightarrow A_y$ by letting $\chi \to 3\pi/2 - \chi$.

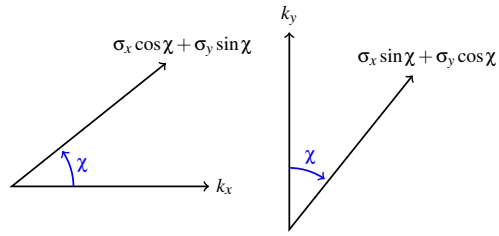


FIG. 3: Geometric interpretation of the SO field (21) in polar coordinates: the Hamiltonian couples the momentum component k_x to the spin component $(\sigma_x\cos\chi + \sigma_y\sin\chi)$ with a coefficient $+a/m$, and the momentum component k_y to the spin component $(\sigma_x\sin\chi + \sigma_y\cos\chi)$ with a coefficient $-a/m$. Thus, a determines the magnitude of the coupling, and χ the angle between the coupled momentum and spin components.

The appearance of LRTs in the system depends on the interplay between SO coupling and the direction of the exchange field. Recall that the LRT components are defined as having spin projections parallel to the exchange field, as opposed to the short-ranged triplet (SRT) component which appears as long as there is exchange splitting[37] but has spin projection perpendicular to the field and is therefore subject to the same pair-breaking effect as the singlets[3,24], penetrating only a very short distance into strong ferromagnets. Thus if we have an SO field component along the layering direction, *e.g.* if we had $A_z \neq 0$ in Figs. 1 and 2, achievable with a noncentrosymmetric crystal or in a nanowire setup, then a non-vanishing commutator $[\underline{A}, \underline{h}\cdot\underline{\sigma}]$ creates the LRT. However, we will from now only consider systems where $A_z = 0$, in which case the criterion for LRT is[21] that $[\underline{A}, [\underline{A}, \underline{h}\cdot\underline{\sigma}]]$ must not be parallel to the exchange field $\underline{h}\cdot\underline{\sigma}$. Expanding, we have

$$\begin{aligned}[\underline{A}, [\underline{A}, \underline{h}\cdot\underline{\sigma}]] &= 4a^2(\underline{h}\cdot\underline{\sigma} + h_z\sigma_z)\\ &\quad - 4a^2(h_x\sigma_y + h_y\sigma_x)\sin 2\chi,\end{aligned} \tag{22}$$

from which it is clear that no LRTs can be generated for a pure Dresselhaus coupling $\chi = 0$ or Rashba coupling $\chi = \pm\pi/2$ when the exchange field is in-plane. However, the effect of SO coupling becomes increasingly significant for angles close to $\pm\pi/4$ (see Fig. 5 in Section III A). We also see that no LRTs can be generated for in-plane magnetization in the special case $h_x = h_y$ and $h_z = 0$, since $h_x\sigma_y + h_y\sigma_x$ can then be rewritten as $h_x\sigma_x + h_y\sigma_y$, which is parallel to $\underline{h}$. There is no LRT generation for the case $h_x = h_y = 0$ and $h_z \neq 0$ for similar reasons. In general however, the LRT will appear for an in-plane magnetization as long as $h_x \neq h_y$ and the SO coupling is not of pure Dresselhaus or pure Rashba type. Once the condition for long-range triplet generation is satisfied, increasing the corresponding exchange field will also increase the proportion of long-range triplets compared with short-range

triplets. Whether or not the presence of long-range triplets can be observed in the system, *i.e.* if they retain a clear signature in measurable quantities such as the density of states when the criteria for their existence is fulfilled, depends on other aspects such as the strength of the spin-orbit coupling and will be discussed later in this paper. Thus, a main motivation for this work is to take a step further than discussing their existence, and make predictions for when long-ranged triplet pairs can actually be observed via spectroscopic or T_c measurements in SF structures with spin-orbit coupling. However, we will also demonstrate that the presence of SO coupling offers additional opportunities besides the creation of LRTs. We will show both analytically and numerically that the SO coupling can protect the singlet component even in the presence of a strong exchange field, which normally would suppress it. This provides the possibility to tune the well-known minigap occuring both in bilayer and Josephson junctions *magnetically* simply by altering the direction of the magnetization.

C. Weak proximity effect

In order to establish a better analytical understanding of the role played by SO coupling in the system before presenting the spectroscopy and T_c results, we will now consider the limit of weak proximity effect, which means that $|\gamma_{ij}| \ll 1$, $N \approx 1$ in the ferromagnet. The anomalous Green's function in general is given by the upper-right block of Eq. (3), $f = 2N\gamma$, which we see reduces to $f = 2\gamma$ in this limit. It will also prove prudent to express the anomalous Green's function using a singlet/triplet decomposition, where the singlet component is described by a scalar function f_s, and the triplet components encapsulated in the so-called d-vector[44,45],

$$f = (f_s + \underline{d}\cdot\underline{\sigma})i\sigma_y\,. \tag{23}$$

Combining the above with the weak proximity identity $f = 2\gamma$, we see that the components of γ can be rewritten as:

$$\gamma = \frac{1}{2}\begin{pmatrix} id_y - d_x & d_z + f_s \\ d_z - f_s & id_y + d_x \end{pmatrix}. \tag{24}$$

Under spin rotations, the singlet component f_s will then transform as a scalar, while the triplet component $\underline{d} = (d_x, d_y, d_z)$ transforms as an ordinary vector. Another useful feature of this notation is that it becomes almost trivial to distinguish between short-range and long-range triplet components; the projection $d_\parallel = \underline{d}\cdot\hat{\underline{h}}$ along the exchange field corresponds to the SRTs, while the perpendicular part $d_\perp = |\underline{d}\times\hat{\underline{h}}|$ describes the LRTs, where $\hat{\underline{h}}$ here denotes the unit vector of the exchange field. For a concrete example, if the exchange field is oriented along the z-axis, then d_z will be the short-range component, while both d_x and d_y are long-ranged components. In the coming sections, we will demonstrate that the LRT component can be identified from its density of states signature, as measurable by tunneling spectroscopy.

In the limit of weak proximity effect, we may linearize both the Usadel equation and Kupriyanov–Lukichev boundary conditions. Using the singlet/triplet decomposition in Eq. (24), and the Rashba–Dresselhaus coupling in Eq. (19), the linearized version of the Usadel equation can be written:

$$\frac{i}{2}D_F\partial_z^2 f_s = \varepsilon f_s + \underline{h}\cdot\underline{d}\,, \tag{25}$$

$$\frac{i}{2}D_F\partial_z^2\underline{d} = \varepsilon\underline{d} + \underline{h}f_s + 2iD_F a^2\underline{\Omega}(\chi)\,\underline{d}\,, \tag{26}$$

where we for brevity have defined an SO interaction matrix

$$\underline{\Omega}(\chi) = \begin{pmatrix} 1 & -\sin 2\chi & 0 \\ -\sin 2\chi & 1 & 0 \\ 0 & 0 & 2 \end{pmatrix}. \tag{27}$$

We have now condensed the Usadel equation down to two coupled differential equations for f_s and $\underline{d}$, where the coupling of course is proportional to the exchange field and the SO interaction term. The latter has been written as a product of a factor $2iD_Fa^2$ depending on the strength a and a factor $\underline{\Omega}(\chi)\underline{d}$ depending on the angle χ in the polar notation. The matrix $\underline{\Omega}(\chi)$ becomes diagonal for a Dresselhaus coupling with $\chi = 0$ or a Rashba coupling with $\chi = \pm\pi/2$, which implies that there is no triplet mixing for such systems. In contrast, the off-diagonal terms are maximal for $\chi = \pm\pi/4$, which suggests that the triplet mixing is maximal when the Rashba and Dresselhaus coefficients have the same magnitude. In addition to the off-diagonal triplet mixing terms, we see that the diagonal terms of $\underline{\Omega}(\chi)$ essentially result in imaginary energy contributions $2iD_Fa^2$. As we will see later, this can in some cases result in a suppression of all the triplet components in the ferromagnet.

We will now consider exchange fields in the xy-plane,

$$\underline{h} = h\cos\theta\,\underline{\hat{x}} + h\sin\theta\,\underline{\hat{y}}\,. \tag{28}$$

Since the linearized Usadel equations show that the presence of a singlet component f_s only results in the generation of triplet components along $\underline{h}$, and the SO interaction term only mixes the triplet components in the xy-plane, the only nonzero triplet components will in this case be d_x and d_y. The SRT amplitude $d_\parallel$ and LRT amplitude $d_\perp$ can therefore be written:

$$d_\parallel = \;\; d_x\cos\theta + d_y\sin\theta\,, \tag{29}$$

$$d_\perp = -d_x\sin\theta + d_y\cos\theta\,. \tag{30}$$

By projecting the linearized Usadel equation for $\underline{d}$ along the unit vectors $(\cos\theta, \sin\theta, 0)$ and $(-\sin\theta, \cos\theta, 0)$, respectively, then we obtain coupled equations for the SRTs and LRTs:

$$\frac{i}{2}D_F\partial_z^2 f_s = \varepsilon f_s + hd_\parallel\,, \tag{31}$$

$$\begin{aligned}\frac{i}{2}D_F\partial_z^2 d_\parallel = {}& [\varepsilon + 2iD_Fa^2(1 - \sin 2\theta\,\sin 2\chi)]\,d_\parallel \\ & - 2iD_Fa^2\cos 2\theta\,\sin 2\chi\,d_\perp + hf_s\,,\end{aligned} \tag{32}$$

$$\begin{aligned}\frac{i}{2}D_F\partial_z^2 d_\perp = {}& [\varepsilon + 2iD_Fa^2(1 + \sin 2\theta\,\sin 2\chi)]\,d_\perp \\ & - 2iD_Fa^2\cos 2\theta\,\sin 2\chi\,d_\parallel\,.\end{aligned} \tag{33}$$

These equations clearly show the interplay between the singlet component f_s, SRT component $d_\parallel$, and LRT component $d_\perp$. If

we start with only a singlet component f_s, then the presence of an exchange field h results in the generation of the SRT component $d_\parallel$. The presence of an SO field can then result in the generation of the LRT component $d_\perp$, where the mixing term is proportional to $a^2\cos 2\theta \sin 2\chi$. This implies that in the weak proximity limit, LRT mixing is absent for an exchange field direction $\theta = \pi/4$, corresponding to $h_x = h_y$, while it is maximized if $\theta = \{0, \pi/2, \pi\}$ and at the same time $\chi = \pm\pi/4$. In other words, the requirement for maximal LRT mixing is therefore that the exchange field is aligned along either the x-axis or y-axis, while the Rashba and Dresselhaus coefficients should have the same magnitude. It is important to note here that although the mixing between the triplet components is maximal at $\theta = \{0, \pi/2, \pi\}$, this does not necessarily mean that the signature of the triplets in physical quantities is most clearly seen for these angles, as we shall discuss in detail later.

Moreover, these equations show another interesting consequence of having an SO field in the ferromagnet, which is unrelated to the LRT generation. Note that the effective quasiparticle energies coupling to the SRTs and LRTs become

$$E_\parallel = \varepsilon + 2iD_F a^2(1 - \sin 2\theta \sin 2\chi)\,, \tag{34}$$

$$E_\perp = \varepsilon + 2iD_F a^2(1 + \sin 2\theta \sin 2\chi)\,. \tag{35}$$

When $\theta = \chi = \pm\pi/4$, then the SRTs are entirely unaffected by the presence of an SO coupling; the triplet mixing term vanishes for these parameters, and $E_\parallel$ is clearly independent of a. However, when $\theta = -\chi = \pm\pi/4$, the situation is drastically different. There is still no possibility for LRT generation, however the SRT energy $E_\parallel = \varepsilon + 4iD_F a^2$ will now obtain an imaginary energy contribution which destabilizes the SRTs. In fact, numerical simulations show that this energy contribution destroys the SRT components as a increases. As we will see in Section III C, this effect results in an increase in the critical temperature of the bilayer. Thus, switching between $\theta = \pm\pi/4$ in a system with $\chi \simeq \pm\pi/4$ may suggest a novel method for creating a triplet spin valve.

When $\chi = \pm\pi/4$ but $\theta \neq \pm\pi/4$, the triplet mixing term proportional to $\cos 2\theta \sin 2\chi$ will no longer vanish, so we get LRT generation in the system. We can then see from the effective triplet energies that as $\theta \to \mathrm{sgn}(\chi)\pi/4$, the imaginary part of $E_\parallel$ vanishes, while the imaginary part of $E_\perp$ increases. This leads to a relative increase in the amount of SRTs compared to the amount of LRTs in the system. In contrast, as $\theta \to -\mathrm{sgn}(\chi)\pi/4$, the imaginary part of $E_\perp$ vanishes, and the imaginary part of $E_\parallel$ increases. This means that we would expect a larger LRT generation for these parameters, up until the point where the triplet mixing term $\cos 2\theta \sin 2\chi$ becomes so small that almost no LRTs are generated at all. The ratio of effective energies coupling to the triplet component at the Fermi level $\varepsilon = 0$ can be written

$$\frac{E_\perp(0)}{E_\parallel(0)} = \frac{1 + \sin 2\theta \sin 2\chi}{1 - \sin 2\theta \sin 2\chi}. \tag{36}$$

D. Density of states

The density of states $D(\varepsilon)$ containing all spin components can be written in terms of the Riccati matrices as

$$D(\varepsilon) = \mathrm{Tr}[N(1 + \gamma\tilde{\gamma})]/2\,, \tag{37}$$

which for the case of zero energy can be written concisely in terms of the singlet component f_s and triplet components $\underline{d}$,

$$D(0) = 1 - |f_s(0)|^2/2 + |\underline{d}(0)|^2/2\,. \tag{38}$$

The singlet and triplet components are therefore directly competing to lower and raise the density of states[46]. Furthermore, since we are primarily interested in the proximity effect in the ferromagnetic film, we will begin by using the known BCS bulk solution in the superconductor,

$$\hat{g}_{BCS} = \begin{pmatrix} \cosh(\theta) & \sinh(\theta) i\sigma_y e^{i\phi} \\ \sinh(\theta) i\sigma_y e^{-i\phi} & -\cosh(\theta) \end{pmatrix}, \tag{39}$$

where $\theta = \mathrm{atanh}(\Delta/\varepsilon)$, and ϕ is the superconducting phase. Using Eq. (24) and the definition of the tilde operation, and comparing $\hat{g}^R$ in Eq. (3) with its standard expression in a bulk superconductor Eq. (39), we can see that at zero energy the singlet component $f_s(0)$ must be purely imaginary and the asymmetric triplet $d_z(0)$ must be purely real if the superconducting phase is $\phi = 0$.

By inspection of Eq. (26), we can see that a transformation $h_x \leftrightarrow h_y$ along with $d_x \leftrightarrow d_y$ leaves the equations invariant, as would be expected for a one-dimensional system aligned along the z-axis. The density of states, which is given by Eq. (38), will therefore be unaffected by such permutations,

$$D[\underline{h} = (a, b, 0)] = D[\underline{h} = (b, a, 0)]\,, \tag{40}$$

while in general

$$D[\underline{h} = (a, 0, b)] \neq D[\underline{h} = (b, 0, a)]\,. \tag{41}$$

However, whenever one component of the planar field is exactly twice the value of the other component, one can confirm that the linearized equations remain invariant under a rotation of the exchange field

$$\underline{h} = (a, 2a, 0) \to \underline{h} = (a, 0, 2a)\,, \tag{42}$$

with associated invariance in the density of states.

E. Critical temperature

When superconducting correlations leak from a superconductor and into a ferromagnet in a hybrid structure, there will also be an inverse effect, where the ferromagnet effectively drains the superconductor of its special properties. Physically, this effect is observable in the form of a reduction in the superconducting gap $\Delta(z)$ near the interface at all temperatures. Furthermore, if the temperature of the hybrid structure is somewhat close to the bulk critical temperature T_{cs} of the superconductor, this *inverse proximity effect* can be strong enough to

make the superconducting correlations vanish entirely throughout the system. Thus, proximity-coupled hybrid structures will in practice always have a critical temperature T_c that is lower than the critical temperature T_{cs} of a bulk superconductor. Depending on the exact parameters of the hybrid system, T_c can sometimes be significantly smaller than T_{cs}, and in some cases it may even vanish ($T_c \to 0$).

To quantify this effect, it is no longer sufficient to solve the Usadel equation in the ferromagnet only. We will now also have to solve the Usadel equation in the superconductor,

$$D_S \partial_z^2 \gamma = -2i\varepsilon\gamma - \Delta(\sigma_y - \gamma\sigma_y\gamma) - 2(\partial_z\gamma)\tilde{N}\tilde{\gamma}(\partial_z\gamma) \,, \qquad (43)$$

along with a self-consistency equation for the gap $\Delta(z)$,

$$\Delta(z) = N_0\lambda \int_0^{\Delta_0 \cosh(1/N_0\lambda)} d\varepsilon \, \mathrm{Re}\{f_s(z,\varepsilon)\} \tanh\left(\frac{\pi}{2e^\gamma}\frac{\varepsilon/\Delta_0}{T/T_{cs}}\right) , \qquad (44)$$

where N_0 is the density of states per spin at the Fermi level, and $\lambda > 0$ is the electron-electron coupling constant in the BCS theory of superconductivity. For a derivation of the gap equation, see Appendix B.

To study the effects of the SO coupling on the critical temperature of an SF structure, we therefore have to find a self-consistent solution to Eq. (5) in the ferromagnet, Eq. (6) at the interface, and Eqs. (43) and (44) in the superconductor. In practice, this is done by successively solving one of the equations at a time numerically, and continuing the procedure until the system converges towards a self-consistent solution. To obtain accurate results, we typically have to solve the Usadel equation for 100–150 positions in each material, around 500 energies in the range $(0, 2\Delta_0)$, and 100 more energies in the range $(2\Delta_0, \omega_c)$, where the Debye cutoff $\omega_c \approx 76\Delta_0$ for the superconductors considered herein. This procedure will then have to be repeated up to several hundred times before we obtain a self-consistent solution for any given temperature of the system. Furthermore, if we perform a conventional linear search for the critical temperature T_c/T_{cs} in the range $(0,1)$ with a precision of 0.0001, it may require up to 10,000 such iterations to complete, which may take several days depending on the available hardware and efficiency of the implementation. The speed of this procedure may, however, be significantly increased by performing a binary search instead. Using this strategy, the critical temperature can be determined to a precision of $1/2^{12+1} \approx 0.0001$ after only 12 iterations, which is a significant improvement. The convergence can be further accelerated by exploiting the fact that $\Delta(z)$ from iteration to iteration should decrease monotonically to zero if $T > T_c$; however, the details will not be further discussed in this paper.

III. RESULTS

We consider the proximity effect in an SF bilayer in III A, and the corresponding SFS Josephson junction in III B, using the BCS bulk solution for the superconductors. We take the thin-film layering direction to be oriented in the z-direction and fix the spin-orbit coupling to Rashba–Dresselhaus type in the xy-plane as given by Eq. (19). For these calculations, we choose the superconductor coherence length to be $\xi_S = 30$ nm, and use a ferromagnet of length $L_F = 15$ nm. We solve the equations using MATLAB with the boundary value differential equation package `bvp6c` and examine the density of states $D(\varepsilon)$ for energies normalised to the superconducting gap Δ. For brevity of notation, we include the normalization factor in the coefficients α and β in these sections. This normalization is taken to be the length of the ferromagnetic region L_F, so that for instance $\alpha = 1$ in the figure legends means $\alpha L_F = 1$. Finally, in Section III C, we calculate the dependence of the critical temperature of an SF bilayer as a function of the different system parameters.

A. SF Bilayer

Consider the SF bilayer depicted in Fig. 1. In section II B we introduced the conditions for the LRT component to appear, and from Eq. (22) it is clear that no LRTs will be generated if the exchange field is aligned with the layering direction, *i.e.* $\underline{h} \sim \underline{\hat{z}}$, since Eq. (22) will be parallel to the exchange field. Conversely, the general condition for LRT generation with in-plane magnetisation is both that $h_x \neq h_y$ and that the SO coupling is not of pure Rashba or pure Dresselhaus form. However, it became clear in Section II C that the triplet mixing was maximal for equal Rashba and Dresselhaus coupling strengths, and in fact the spectroscopic signature is quite sensitive to deviations from this.

For SF bilayers without SO coupling and a homogeneous exchange field, one expects to see a spectroscopic minigap whenever the Thouless energy is much greater than the strength of the exchange field. The minigap in SF structures closes when the resonant condition $h \sim E_g$ is fulfilled where E_g is the minigap occuring without an exchange field and a zero-energy peak emerges instead[41]. The minigap E_g depends on both the Thouless energy and the resistance of the junction. For stronger fields we will have an essentially featureless density of states (see *e.g.* Ref. 39 and references therein). This is indeed what we observe for $\alpha = \beta = 0$ in Fig. 6. With purely out-of-plane magnetisation $\underline{h} \sim \underline{\hat{z}}$, the effect of SO coupling is irrespective of type: Rashba, Dresselhaus or both will always create a minigap. With in-plane magnetisation however, the observation of a minigap above the resonant condition[39,41] indicates that dominant Rashba or dominant Dresselhaus coupling is present. The same is true for SFS trilayers, and thus to observe a signature of long-range triplets the Rashba–Dresselhaus coefficients must be similar in magnitude, and in the following we shall primarily focus on this regime. To clarify quantitatively how much the Rashba and Dresselhaus coefficients can deviate from each other before destroying the low-energy enhancement of the density of states, which is the signature of triplet Cooper pairs in this system, we have plotted in Fig. 4 the density of states at the Fermi level ($\varepsilon = 0$) as a function of the spin-orbit angle χ and the magnetization direction θ. For purely Rashba or Dresselhaus coupling ($\chi = \{0, \pm\pi/2\}$), the deviation from the normal-state value is small. However, as soon as both components are present a highly non-monotonic behavior is observed.

This is particularly pronounced for $\chi \to \pm\pi/4$, although the conversion from dominant triplets to dominant singlets as one changes θ is seen to occur even away from $\chi = \pm\pi/4$.

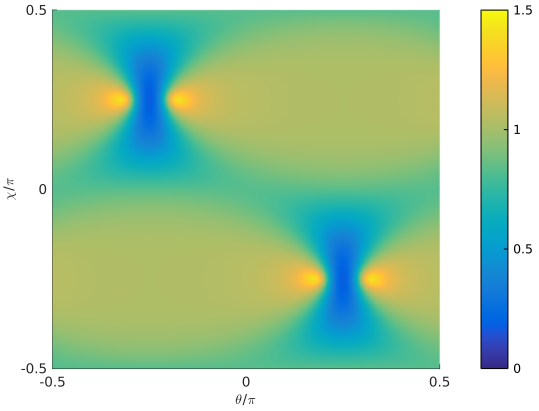


FIG. 4: Zero-energy density of states $D(0)$ as a function of the spin-orbit angle χ and magnetization angle θ. We have used a ferromagnet of length $L_F/\xi_S = 0.5$ with an exchange field $h/\Delta = 3$ and a spin-orbit magnitude $a\xi_S = 2$.

With either $\underline{h} = h\underline{\hat{x}} \neq 0$, or equivalently $\underline{h} = h\underline{\hat{y}} \neq 0$, LRTs are generated provided $\alpha\beta \neq 0$, and in Fig. 7 we can see that the addition of SO coupling introduces a peak in the density of states at zero energy, which saturates for a certain coupling strength. This peak manifests as sharper around $\varepsilon = 0$ than the zero-energy peak associated with weak field strengths of the order of the gap (*i.e.* as evident from $\alpha = \beta = 0$ in Fig. 7), which occurs regardless of magnetisation direction or texture[39,41]. By analysing the real components of the triplets, for a gauge where the superconducting phase is zero, we can confirm that this zero-energy peak is due to the LRT component, in this case d_x, also depicted in Fig. 7, in agreement with the predictions for *textured* magnetisation *without* SO coupling[4,39]. However, it is also evident from Fig. 7 that increasing the field strength rapidly suppresses the density of states towards that of the normal metal, making the effect more difficult to detect experimentally. However, as we see in Fig. 5, the introduction of SO coupling means the *direction* of the exchange field is crucially important, and allows for a dramatic spectroscopic signature for fields without full alignment with the x- or y-axes.

Fig. 5 shows how the density of states at zero energy varies with the angle θ between h_x and h_y at zero energy; with $\theta = 0$ the field is aligned with h_x, and with $\theta = \pi/2$ it is aligned with h_y. We see that the inclusion of SO coupling introduces a nonmonotonic angular dependance in the density of states, with increasingly sharp features as the SO coupling strength increases, although the optimal angle at approximately $\theta = 7\pi/32$ and $\theta = 9\pi/32$ varies minimally with increasing SO coupling. Clearly the ability to extract maximum LRT conversion from the inclusion of SO coupling is highly sensitive to the rotation angle, with near step-function behaviour delineating the regions of optimal peak in the density of states and an energy gap for strong SO coupling. It is remarkable to see how $D(0)$ vs. θ formally bears a strong resemblance to the evolution of a fully gapped BCS density of states $D(\varepsilon)$ vs. ε to a flat density of states as the SO coupling decreases.

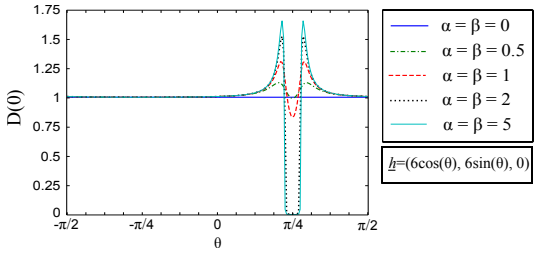


FIG. 5: The dependence of the density of states of the SF bilayer at zero energy on the angle θ between the x and y components of the exchange field $\underline{h}/\Delta = 6(\cos\theta, \sin\theta, 0)$ for increasing SO coupling. As the strength of the coupling increases, we see increasingly sharp variations from an optimal peak at around $\theta \approx 7\pi/32$ and $\theta \approx 9\pi/32$ to a gap at $\theta = \pi/4$.

These results can again be explained physically by the linearized equations (31)–(33). Since the case $\alpha = \beta$ corresponds to $\chi = -\pi/4$ in the notation developed in the preceding sections, Eq. (36) implies that $E_\perp(0) > E_\parallel(0)$ when $\theta < 0$, while $E_\perp(0) < E_\parallel(0)$ when $\theta > 0$. In other words, for negative θ, the SO coupling suppresses the LRT components, and the exchange field suppresses the other components. Since the singlet and SRT components have opposite sign in Eq. (38), this renders the density of states essentially featureless. However, for positive θ, both the SO coupling and the exchange field suppress the SRT components, meaning that LRT generation is energetically favoured. Note that $E_\perp/E_\parallel \to \infty$ as $\theta \to +\pi/4$, which explains why the LRT generation is maximized in this regime. Since the triplet mixing term in Eq. (33) is proportional to $\cos 2\theta \sin 2\chi$, the LRT component vanishes when the value of θ gets too close to $+\pi/4$. Furthermore, since $E_\parallel$ has a large imaginary energy contribution in this case, the SRTs are also suppressed at $\theta = +\pi/4$. Thus, despite LRTs being most energetically favored at this exact point, we end up with a system dominated by singlets due to the SRT suppression and lack of LRT production pathway Nevertheless, one would conventionally expect that exchange fields of a magnitude $h \gg \Delta$ as depicted in Fig.5 would suppress any features in the density of states, while we observe an obvious minigap. It thus appears as if the singlet correlations are much more resilient against the pair-breaking effect of the exchange field when spin-orbit coupling is present. To identify the physical origin of this effect, we solve the linearized equations (31)–(33) along with their corresponding boundary conditions for the specific case $\varepsilon = 0$, $\theta = -\chi = \pi/4$. We consider a bulk superconductor occupying the space $x < 0$ while the F has a length L that is so large that one in practice only needs to keep the decaying parts of the anomalous Green's function in the F region. We then find the following expression for the singlet component at the

SF interface in the absence of SO coupling:

$$f_s^0 = \frac{s}{2\zeta L}\sqrt{\frac{D}{h}}. \tag{45}$$

With increasing h, the singlet correlations are suppressed in the conventional manner. However, we now incorporate SO coupling in the problem. For more transparent analytical results, we focus on the case $2(a\xi)^2 \gg h/\Delta$. This condition can be rewritten as $2Da^2 \gg h$. In this case, a similar calculation gives the singlet component at the SF interface in the presence of SO coupling:

$$f_s = f_s^0\sqrt{\frac{Da^2}{2h}}. \tag{46}$$

Clearly, the SO coupling enhances the singlet component in spite the presence of an exchange field since $\sqrt{Da^2/h} \gg 1$. This explains the presence of the conventional zero energy gap for large SO coupling even with a strong exchange field. A consequence of this observation is that SO coupling in fact provides a route to a *magnetically tunable minigap*. Fig. 5 shows that when both an exchange field and SO coupling is present, the direction of the field determines when a minigap appears. This holds even for strong exchange fields $h \gg \Delta$ as long as the SO coupling is sufficiently large as well.

Concerning the LRT component, we recall that is defined as the component of $\underline{d}$ that is perpendicular to $\underline{h}$, and this perpendicular component $d_\perp$ can be found easily from the cross product of the two vectors: $d_\perp = |\underline{d} \times \hat{\underline{h}}|$. We saw above that the spectroscopic signature of LRT generation is strongly dependent on the angle of the field, and this angle is a tunable parameter for sufficiently weak magnetic anisotropy. In Fig. 8 we see an example of the dramatic effect this rotation can have on the spectroscopic signature of LRT generation. Comparing the effect of increasing the exchange field $\underline{h} = \Delta\hat{\underline{y}} \to 3\Delta\hat{\underline{y}}$ in Fig. 7 , when the exchange field is changed from $\underline{h} = (6\Delta, 3\Delta, 0) \to (6\Delta, 5\Delta, 0)$ (*i.e.* changing the direction of the field), as shown in Fig. 8, we see that a strong zero-energy peak now emerges due to the presence of LRT in the system. If one were to remove the SO coupling, the low-energy DOS would have no trace of any superconducting proximity effect, which demonstrates the important role played by the SO interactions here. Finally, for completeness we include an example of the effect of rotating the field to have a component along the junction in Fig. 9. Comparing the case of $\underline{h} = (0, 3\Delta, 6\Delta)$ in Fig. 9 with $\underline{h} = (6\Delta, 3\Delta, 0)$ in Fig. 8, we see that the two cases are identical, as predicted in the limit of weak proximity effect, and increasing the magnitude of the out-of-plane z component of the field has no effect on the height of the zero-energy peak, which is instead governed by the in-plane y component.

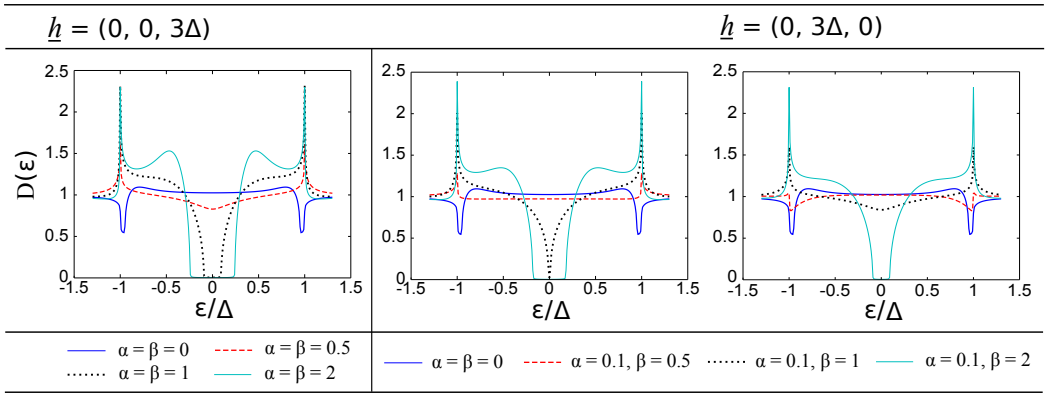


FIG. 6: Density of states $D(\varepsilon)$ for the SF bilayer with energies normalised to the superconducting gap Δ and SO coupling normalised to the inverse ferromagnet length $1/L_F$. The table shows the spectroscopic effect of increasing SO with $\alpha = \beta$ when the magnetisation $\underline{h} = 3\Delta\hat{z}$, *i.e.* with the field perpendicular to the interface, and the effect of increasing difference between the Rashba and Dresselhaus coefficients for both $\underline{h} = 3\Delta\hat{z}$ and $\underline{h} = 3\Delta\hat{y}$. Although the conditions for LRT generation are fulfilled in the latter case, it is clear that no spectroscopic signature of this is present.

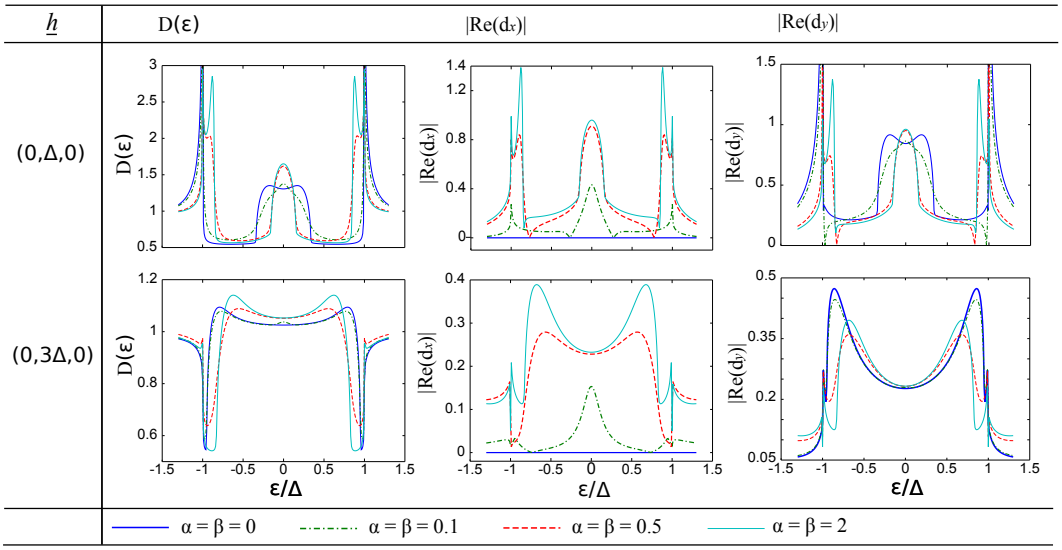


FIG. 7: Density of states $D(\varepsilon)$ for the SF bilayer with energies normalised to the superconducting gap Δ and SO coupling normalised to the inverse ferromagnet length $1/L_F$. The table shows the spectroscopic effect of equal Rashba–Dresselhaus coefficients when the magnetisation is oriented entirely in the y-direction, and also the correlation between the SO-induced zero-energy peak with the long-range triplet component $|\mathrm{Re}(d_x)| \equiv \mathrm{Re}(d_\perp)$. It is clear that the predominant effect of the LRT component, which appears only when the SO coupling is included, is to increase the peak at zero energies. Increasing the field strength rapidly suppresses the density of states towards that of the normal metal.

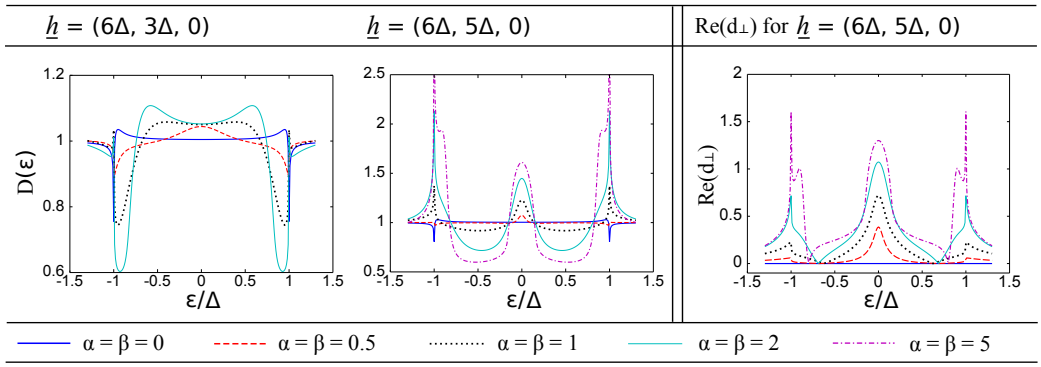


FIG. 8: Density of states $D(\varepsilon)$ in the SF bilayer for energies normalised to the superconducting gap Δ and SO coupling normalised to the inverse ferromagnet length $1/L_F$. The table shows the spectroscopic features of the SF bilayer with rotated exchange field in the xy-plane. Again we see a peak in the density of states at zero energy due to the LRT component, *i.e.* the component of $\underline{d}$ perpendicular to $\underline{h}$, $d_\perp$. The height of this zero-energy peak is strongly dependent on the angle of the field vector in the plane, as shown in Fig. 5. For near-optimal field orientations increasing the SO coupling leads to a dramatic increase in the peak of the density of states at zero energy.

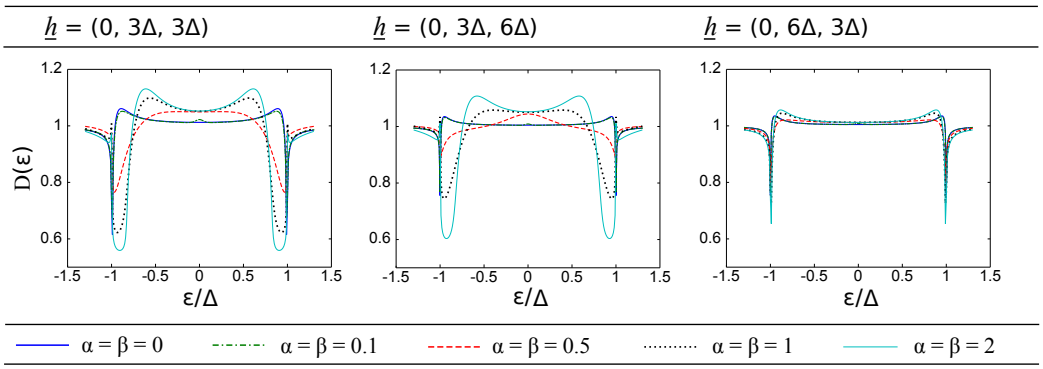


FIG. 9: Density of states $D(\varepsilon)$ in the SF bilayer for energies normalised to the superconducting gap Δ and SO coupling normalised to the inverse ferromagnet length $1/L_F$. The table shows the spectroscopic features of the SF bilayer with a rotated exchange field in the $xz \equiv yz$-plane. Note that when the field component along the junction is twice the component in the y-direction, here $\underline{h} = (0, 3\Delta, 6\Delta)$, the density of states is equivalent to the case $\underline{h} = (6\Delta, 3\Delta, 0)$ illustrated in Fig. 8, as predicted in the limit of weak proximity effect.

B. Josephson junction

By adding a superconducting region to the right interface of the SF bilayer we form an SFS Josephson junction. It is well known that the phase difference between the superconducting regions governs how much current can flow through the junction[48], and the density of states for a diffusive SNS junction has been measured experimentally with extremely high precision[47]. Here we consider such a transversal junction structure as depicted in Fig.2, again with intrinsic SO coupling in the *xy*-plane (Eq. 19) in the ferromagnet and with BCS bulk values for each superconductor. In III B 1 we consider single orientations of the exchange field along the principal axes of the system, and in III B 2 we consider rotated fields.

Let us first recapitulate some known results. We saw in Section II that the spin-singlet, SRT and LRT components compete to raise and lower the density of states at low energies. Their relative magnitude is affected by the magnitude and direction of both the exchange field and SO coupling and results in three distinctive qualitative profiles: the zero-energy peak from the LRTs, the singlet-dominated regime with a minigap, and the flat, featureless profile in the absence of superconducting correlations. In the Josephson junction, the spectroscopic features are in addition sensitive to the phase difference ϕ between the superconductors. In junctions with an interstitial normal metal, the gap decreases as $\phi = 0 \rightarrow \pi$, closing entirely at $\phi = \pi$ such that the density of states is that of the isolated normal metal; identically one[47,49]. Without an exchange field the density of states is unaffected by the SO coupling. This is because without an exchange field the equations governing the singlet and triplet components are decoupled and thus no singlet-triplet conversion can occur. From a symmetry point of view, it is reasonable that the time-reversal invariant spin-orbit coupling does not alter the singlet correlations.

Without SO coupling and as long as the exchange field is not too large, changing the phase difference can qualitatively alter the density of states from minigap to peak at zero energy (see Fig. 12), a highly useful feature permitting external control of the quasiparticle current flowing through the junction. Beyond a system specific threshold strength of exchange field, the minigap can no longer be sustained and increasing the phase difference simply lowers the density of states towards that of the normal metal. Amongst the features we outline in the following subsections, one of the effects of adding SO coupling is to make this useful gap-to-peak effect accessible with stronger exchange fields, *i.e.* for a greater range of materials. At the same time, the SO coupling cannot be *too strong* since the triplet correlations are suppressed in this regime leaving only the minigap and destroying the capability for qualitative change in the spectroscopic features.

1. Josephson junction with uniform exchange field in single direction

Consider first the case in which the exchange field is aligned in a single direction, meaning that we only consider exchange fields purely along the principal axes of the system. If we again restrict the form of the SO field to (19), aligning $\underline{h}$ in the z-direction will not result in any LRTs. In this case the spectroscopic effect of the SO coupling is dictated by the singlet and short-range triplet features, much as in the SF bilayer case (Fig. 6). This is demonstrated in Fig. 12, where again we see a qualitative change in the density of states for low fields where the resonant condition $h \sim E_g$ can be met.

We will now examine the effect of increasing the exchange field aligned in the x- or, equivalently, the y-direction. In this case, we have generation of LRT Cooper pairs. If $\underline{h}$ is sufficiently weak to sustain a gap independently of SO coupling, introducing weak SO coupling will increase the gap at zero phase difference while maintaining a peak at zero energy for a phase difference of 0.75π (see Fig. 12). Increasing the SO coupling increases this peak at zero energy up to a saturation point before the short-range behaviour dominates, manifesting in increasing side-peaks. As the exchange field increases sufficiently beyond the resonant condition to keep the gap closed, increasing the SO coupling increases the zero-energy peak at all phases, again due to the LRT component, eventually reaching a saturation point. As the phase difference $\phi = 0 \rightarrow \pi$, the density of states reduces towards that of the normal metal, closing entirely at $\phi = \pi$ as expected[42,43,49]. As the value of the density of states at zero energy saturates for increasing SO coupling, fixed phase differences yield the same drop at zero energy regardless of the strength of SO coupling.

We note in passing that when the SO coupling field has a component along the junction direction (z), it can qualitatively influence the nature of the superconducting proximity effect. As very recently shown in Ref. 43, a giant triplet proximity effect develops at $\phi = \pi$ in this case, in contrast to the normal scenario of a vanishing proximity effect in π-biased junctions.

2. Josephson junction with rotated exchange field

With two components of the exchange field $\underline{h}$, *e.g.* from rotation, it is again useful to separate the cases with and without a component along the junction direction. When the exchange field lies in-plane (the *xy*-plane), and provided we satisfy the conditions $h_x \neq h_y$ and $\alpha\beta \neq 0$, increasing the SO coupling drastically increases the zero energy peak as shown in Fig. 13, again due to the LRT component. This is consistent with the bilayer behavior, where the maximal generation of LRT Cooper pairs occurs at an angle $0 < \theta < \pi/4$. As the phase difference approaches π, the proximity-induced features are suppressed in the centre of the junction. This can be understood intuitively as a consequence of the order parameter averaging to zero since it is positive in one superconductor and negative in the other.

The 2D plots in this paper of the local density of states are given for the centre of the junction ($z = 0$), where one naturally expects the relative proportion of LRTs to be greatest. How-

ever, it is interesting to note that the zero-energy peak—the signature of the LRTs—is maintained throughout the ferromagnet. This is shown in Fig. 10, for the case $\alpha = \beta = 1$ and $\underline{h} = (1.5\Delta, 3.5\Delta, 0)$, where the maximal peak for $\phi = 0$ is almost twice the normal-state value. The depletion of this peak is surprisingly small at the superconductor interfaces.

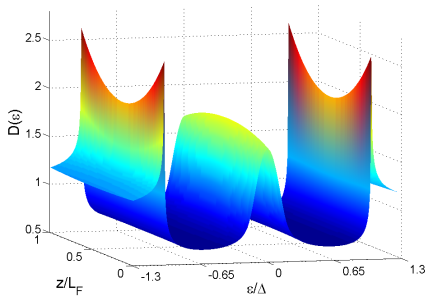


FIG. 10: Spatial distribution of the density of states $D(\varepsilon)$ throughout the ferromagnet of an SFS junction with phase difference $\phi = 0$, spin-orbit coupling $\alpha = \beta = 1$ and magnetisation $\underline{h} = (1.5\Delta, 3.5\Delta, 0)$.

With one component of the exchange field along the junction and another along either x or y, a phase-dictated gap-to-peak transition at zero energy is possible with stronger fields than with the field aligned in a single direction, as shown in Fig. 13. Notice that in this case increasing the phase difference $\phi = 0 \rightarrow 0.5\pi$ gives an increase in the peak at zero energy before reducing towards the normal metal state. For higher field strengths we find once again that increasing the SO coupling increases the peak at zero energy, up to a system-specific threshold, and increasing phase difference reduces the density of states towards that of the normal metal.

It is also useful to consider how the zero-energy density of states depends simultaneously on the phase-difference and magnetization orientation. To this end, we show in Fig. 11 a contour plot of the density of states at the Fermi level ($\varepsilon = 0$) as a function of the superconducting phase difference ϕ across the junction and the magnetization direction θ. The proximity effect vanishes in the centre of the junction at $\phi = \pi$ for any value of the exchange field orientation, giving the normal-state value. Just as in the bilayer case (Fig. 4), we see that the proximity effect is strongly suppressed for the range of angles $\theta > 0$. Rotating the field in the opposite direction, $\theta < 0$, strongly non-monotonic behavior emerges. For zero phase-difference, the physics is qualitatively similar to the bilayer situation. In this case, we proved analytically that the LRT is not produced at all when $\theta = -\pi/4$. Accordingly, Fig. 11 shows a full minigap there.

Whether or not a clear zero-energy peak can be seen due to the LRT depends on the relative strength of the Rashba and Dresselhaus coupling. In the top panel, we have dominant Dresselhaus coupling in which case the low-energy density of states show either normal-state behavior or a minigap. Interestingly, we see that the same opportunity appears in the present case of a Josephson setup as in the bilayer case: a magnetically tunable minigap appears. This effect exists as long as the phase difference is not too close to π, in which case the minigap closes. In the bottom panel corresponding to equal magnitude of Rashba and Dresselhaus, however, a strong zero-energy enhancement due to long-range triplets emerges as one moves away from $\theta = -\pi/4$. With increasing phase difference, the singlets are seen to be more strongly suppressed than the triplet correlations since the minigap region (dark blue) vanishes shortly after $\phi/\pi \simeq 0.6$ while the peaks due to triplets remain for larger phase differences.

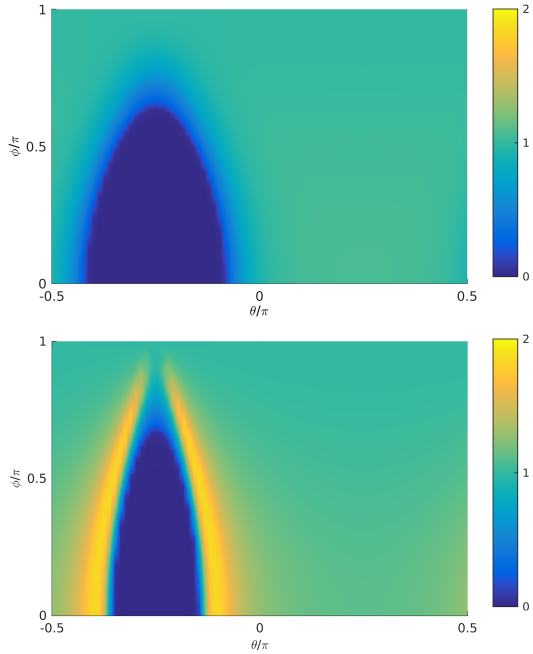


FIG. 11: Zero-energy density of states $D(0)$ as a function of the phase-difference ϕ and magnetization angle θ, both tunable parameters experimentally. The other parameters used are $L_F/\xi_S = 0.5$, $h/\Delta_0 = 3$, $a\xi_S = 2$. The top panel shows a plot for $\chi = 0.15\pi$, and the bottom one $\chi = 0.25\pi$.

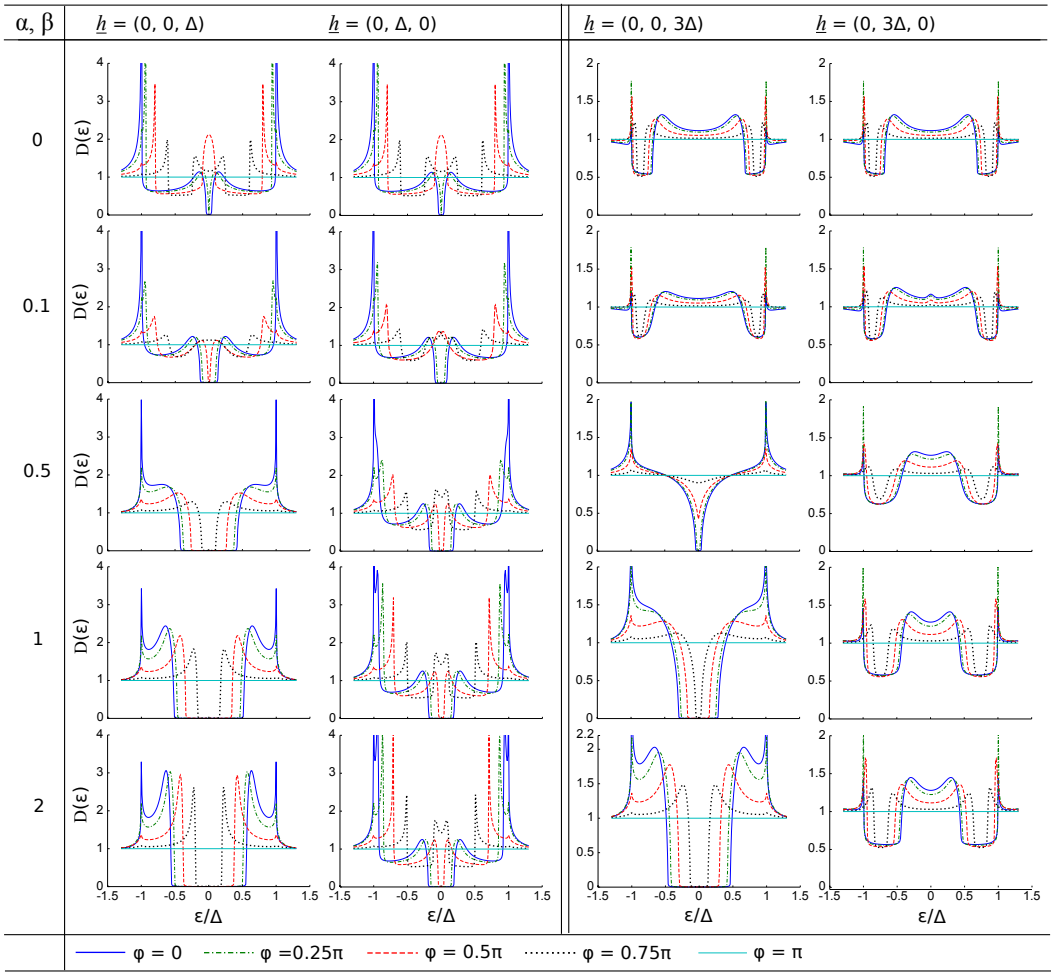


FIG. 12: The table shows the density of states $D(\varepsilon)$ in the SFS junction with increasing SO coupling and exchange field in a single direction, with $D(\varepsilon)$ normalised to the superconducting gap Δ and SO coupling normalised to the ferromagnet length L_F. With no SO coupling and very weak exchange field we see a phase-dictated gap-to-peak qualitative change in the density of states at zero energy. When the field is strong enough to destroy this gap, increasing the phase difference simply lowers the density of states towards that of the normal metal, which is achieved at a phase difference of $\phi = \pi$. With the addition of SO coupling we see a clear difference in the density of states due to the long range triplet component, which is present when the field is oriented in y but not in z. When LRTs are present with weak exchange fields, a phase-dictated gap-to-peak feature is retained and increased as the strength of SO coupling increases the gap, with the peak shown here at a phase difference of 0.75π. For stronger exchange fields, increasing the SO coupling produces the minigap when there is no LRT component, whereas the existence of an LRT component again introduces an increasing peak at zero energy when no minigap is present.

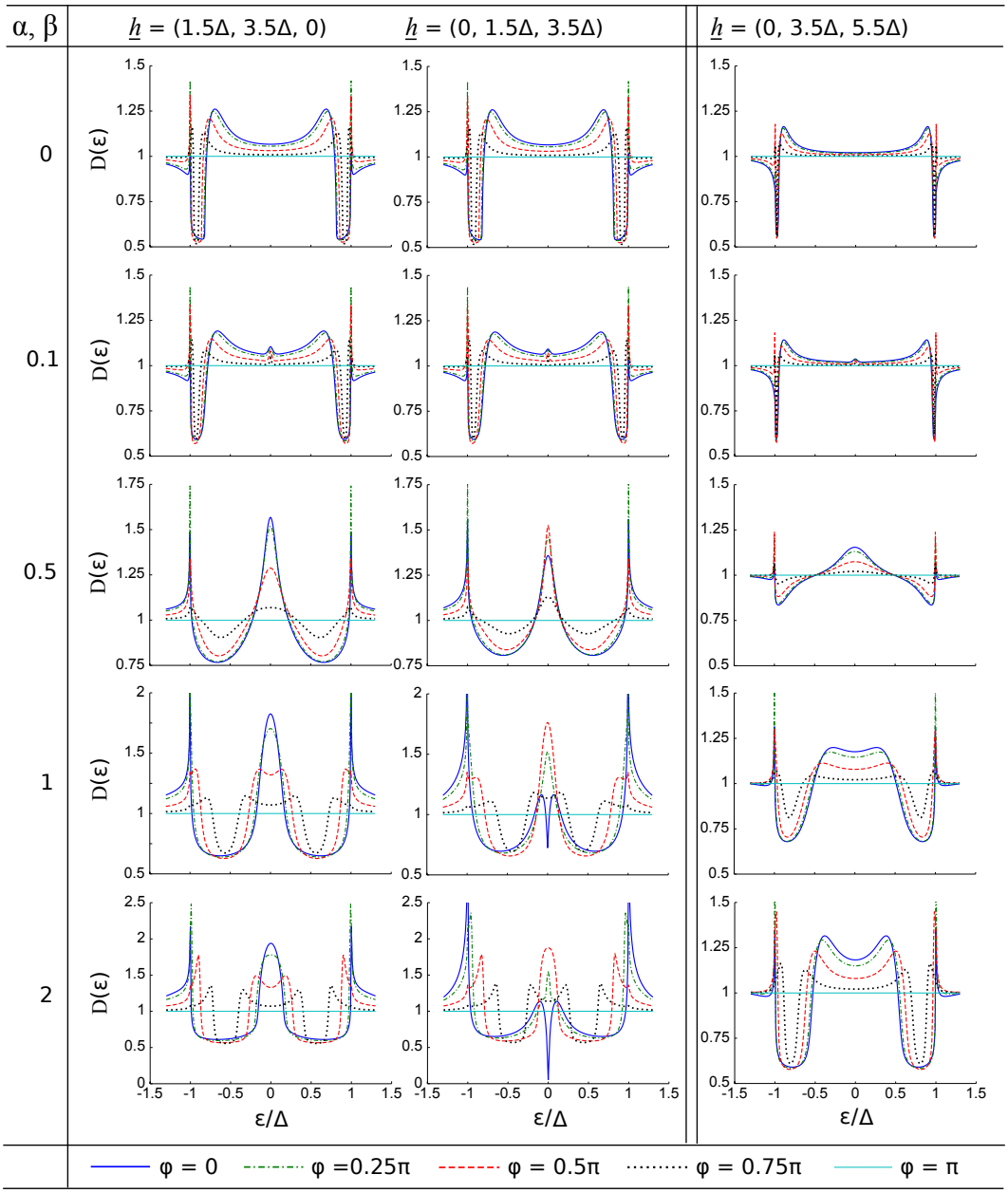


FIG. 13: Density of states $D(\varepsilon)$ in the SFS junction for energies normalised to the superconducting gap Δ and SO coupling normalised to the ferromagnet length L_F. The table shows the spectroscopic effects of increasing SO coupling in SFS with rotated exchange field. In the absence of SO coupling, the density of states is flat and featureless at low energies. Increasing the SO coupling again leads to a strong increase in the peak of the density of states at zero energy, while increasing the phase difference reduces the peak and shifts the density of states weight toward the gap edge for higher SO coupling strengths. With a component of the field in the junction direction a qualitative change in the density of states from strongly suppressed to enhanced at zero energy can be achieved by altering the phase difference between the superconductors. This change can occur in the presence of stronger exchange fields when SO coupling is included. Increasing the exchange field destroys the ability to maintain a gap in the density of states and the LRT component of the SO coupling increases the zero-energy peak as it did in the bilayer case.

C. Critical temperature

In this section, we present numerical results for the critical temperature T_c of an SF bilayer. The theory behind these investigations is summarized in Section II E, and discussed in more detail in Appendix B. An overview of the physical system is given in Fig. 1. In all of the simulations we performed, we used the material parameter $N_0\lambda = 0.2$ for the superconductor, the exchange field $h = 10\Delta_0$ for the ferromagnet, and the interface parameter $\zeta = 3$ for both materials. The other physical parameters are expressed in a dimensionless form, with lengths measured relative to the superconducting correlation length ξ_S, energies measured relative to the bulk zero-temperature gap Δ_0, and temperatures measured relative to the bulk critical temperature T_{cs}. This includes the SO coupling strength a, which is expressed in the dimensionless form $a\xi_S$. The plots presented in this subsection were generated from 12–36 data points per curve, where each data point has a numerical precision of 0.0001 in T_c/T_{cs}. The results were smoothed with a LOESS algorithm.

Before we present the results with SO coupling, we will briefly investigate the effects of the ferromagnet length L_F and superconductor length L_S on the critical temperature, in order to identify the interesting parameter regimes. The critical temperature as a function of the size of the superconductor is shown in Fig. 14.

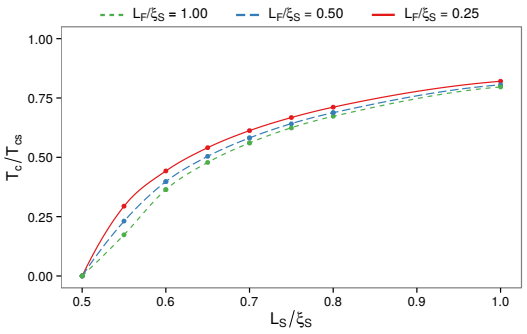


FIG. 14: Plot of the critical temperature T_c/T_{cs} as a function of the length L_S/ξ_S of the superconductor for $a\xi_S = 0$.

First of all, we see that the critical temperature drops to zero when $L_S/\xi_S \approx 0.5$. After this, the critical temperature increases quickly, already reaching nearly 50% of the bulk value when $L_S/\xi_S = 0.6$, indicating that the superconductivity is very sensitive to small changes in parameters for this region.

The next step is then to observe how the behaviour of the system varies with the size of the ferromagnet, and these results are presented in Fig. 15.

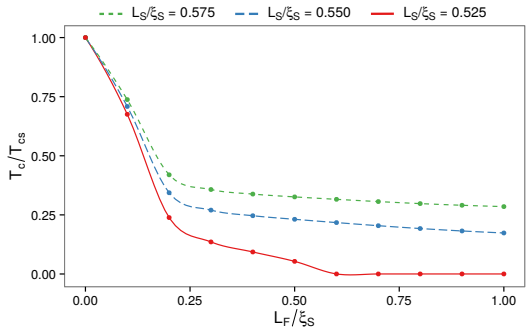


FIG. 15: Plot of the critical temperature T_c/T_{cs} as a function of the ferromagnet length L_F/ξ_S for $a\xi_S = 0$.

We again observe that the critical temperature increases with the size of the superconductor, and decreases with the size of the ferromagnet. The critical temperature for a superconductor with $L_S/\xi_S = 0.525$ drops to zero at $L_F/\xi_S \approx 0.6$, and stays that way as the size of the ferromagnet increases. Thus we do not observe any strongly nonmonotonic behaviour, such as reentrant superconductivity, for our choice of parameters. This is consistent with the results of Fominov *et al.*, who only reported such behaviour for systems where either the interface parameter or the exchange field is drastically smaller than for the bilayers considered herein[53].

We now turn to the effects of the antisymmetric SO coupling on the critical temperature, which has not been studied before. Figs. 16 and 17 show plots of the critical temperature as a function of the SO angle χ for an exchange field in the z-direction. The critical temperature is here independent of the SO angle χ. This result is reasonable, since the SO coupling is in the xy-plane, which is perpendicular to the exchange field for this geometry. We also observe a noticeable increase in critical temperature for larger values of a. This behaviour can be explained using the linearized Usadel equation. According to Eq. (26), the effective energy E_z coupling to the triplet component in the z-direction becomes

$$E_z = \varepsilon + 4iD_F a^2 \,; \tag{47}$$

so in other words, the SRTs obtain an imaginary energy shift proportional to a^2. However, as shown in Eq. (25), there is no corresponding shift in the energy of the singlet component. This effect reduces the triplet components relative to the singlet component in the ferromagnet, and as the triplet proximity channel is suppressed the critical temperature becomes restored to higher values.

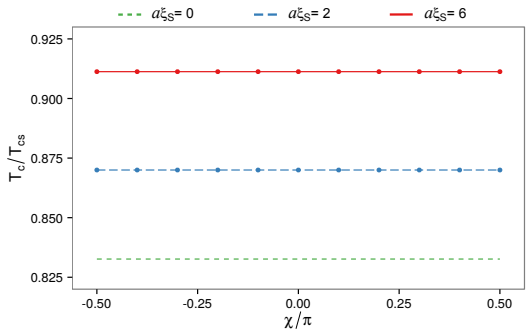


FIG. 16: Plot of the critical temperature T_c/T_{cs} as a function of the SO angle χ, when $L_S/\xi_S = 1.00$, $L_F/\xi_S = 0.2$, and $\underline{h} \sim \underline{\hat{z}}$.

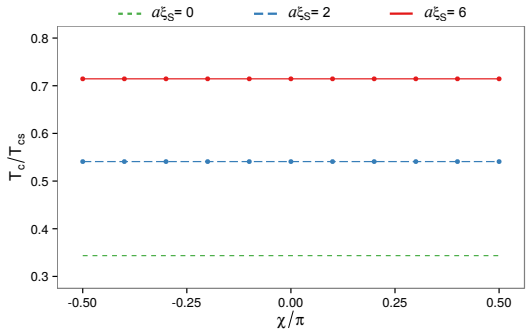


FIG. 17: Plot of the critical temperature T_c/T_{cs} as a function of the SO angle χ, when $L_S/\xi_S = 0.55$, $L_F/\xi_S = 0.2$, and $\underline{h} \sim \underline{\hat{z}}$.

The same situation for an exchange field along the x-axis is shown in Figs. 18 and 19. For this geometry, we observe a somewhat smaller critical temperature for all $a > 0$ and all χ compared to Figs. 16 and 17. This can again be explained by considering the linearized Usadel equation in the ferromagnet, which suggests that the effective energy E_x coupling to the triplet component in the x-direction should be

$$E_x = \varepsilon + 2iD_F a^2 \,, \tag{48}$$

which has a smaller imaginary part than the corresponding equation for E_z. Furthermore, note the drop in critical temperature as $\chi \to \pm\pi/4$. Since the linearized equations contain a triplet mixing term proportional to $\sin 2\chi$, which is maximal precisely when $\chi = \pm\pi/4$, these are also the geometries for which we expect a maximal LRT generation. Thus, this decrease in critical temperature near $\chi = \pm\pi/4$ can be explained by a net conversion of singlet components to LRTs in the system, which has an adverse effect on the singlet amplitude in the superconductor, and therefore the critical temperature.

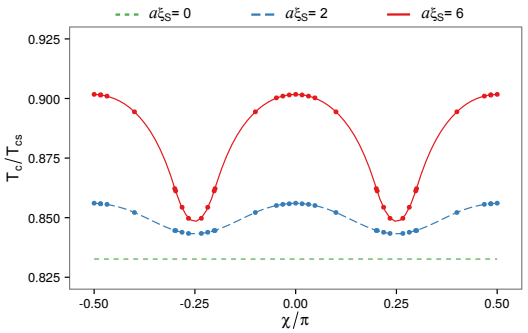


FIG. 18: Plot of the critical temperature T_c/T_{cs} as a function of the SO angle χ, when $L_S/\xi_S = 1.00$, $L_F/\xi_S = 0.2$, and $\underline{h} \sim \underline{\hat{x}}$.

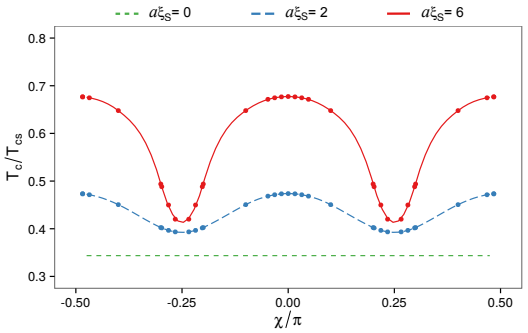


FIG. 19: Plot of the critical temperature T_c/T_{cs} as a function of the SO angle χ, when $L_S/\xi_S = 0.55$, $L_F/\xi_S = 0.2$, and $\underline{h} \sim \underline{\hat{x}}$.

In Figs. 20 and 21 we present the results for a varying exchange field $\underline{h} \sim \cos\theta\,\underline{\hat{x}} + \sin\theta\,\underline{\hat{y}}$ in the xy-plane. In this case, we observe particularly interesting behaviour: the critical temperature has extrema at $|\chi| = |\theta| = \pi/4$, where the extremum is a maximum if θ and χ have the same sign, and a minimum if they have opposite signs. Since $\theta = \pm\pi/4$ is precisely the geometries for which we do not expect any LRT generation, triplet mixing cannot be the source of this behaviour. For the choice of physical parameters chosen in Fig. 21, this effect results in a difference between the minimal and maximal critical temperature of nearly 60% as the magnetization direction is varied. As shown in Fig. 20, the effect persists qualitatively in larger structures as well, but is then weaker.

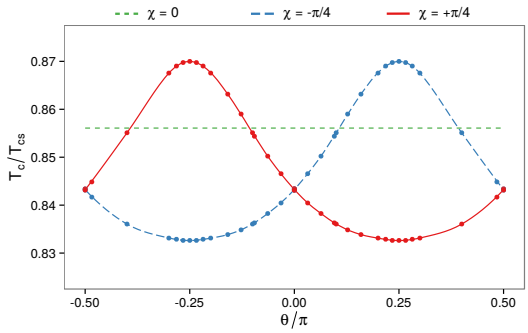


FIG. 20: Plot of critical temperature as function of the magnetization angle θ, when $L_S/\xi_S = 1.00$, $L_F/\xi_S = 0.2$, and $a\xi_S = 2$.

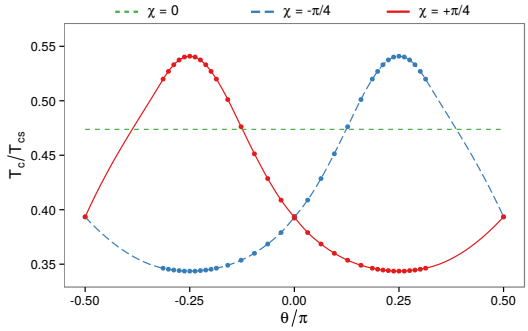


FIG. 21: Plot of critical temperature as function of the magnetization angle θ, when $L_S/\xi_S = 0.55$, $L_F/\xi_S = 0.2$, and $a\xi_S = 2$.

Instead, these observations may be explained using the theory developed in Section II. When we have a general exchange field and SO field in the xy-plane, Eq. (34) reveals that that the effective energy of the SRT component is

$$E_{\parallel} = \varepsilon + 2iD_F a^2(1 - \sin 2\theta \, \sin 2\chi) \, . \tag{49}$$

Since the factor $1 - \sin 2\theta \sin 2\chi$ vanishes for $\theta = \chi = \pm\pi/4$, we get $E_{\parallel} = \varepsilon$ for this case. This geometry is also one where we do not expect any LRT generation, since the triplet mixing factor $\cos 2\theta \sin 2\chi = 0$, so the conclusion is that the SO coupling has no effect on the behaviour of SRTs for these parameters—at least according to the linearized equations. However, since $1 - \sin 2\theta \sin 2\chi = 2$ for $\theta = -\chi = \pm\pi/4$, the situation is now dramatically different. The SRT effective energy is now $E_{\parallel} = \varepsilon + 4iD_F a^2$, with an imaginary contribution which again destabilizes the SRTs, and increases the critical temperature of the system. We emphasize that the variation of T_c with the magnetization direction is present when $\chi \neq \pi/4$ as well, albeit with a magnitude of the variation that gradually declines as one approaches pure Rashba or pure Dresselhais coupling.

D. Triplet spin-valve effect with a single ferromagnet

The results discussed in the previous section show that the critical temperature can be controlled via the magnetization direction of one single ferromagnetic layer. This is a new result originating from the presence of SO coupling. In conventional SF structures, T_c is independent of the magnetization orientation of the F layer. By using a spin-valve setup such as FSF[57–61], it has been shown that the relative magnetization configuration between the ferromagnetic layers will tune the T_c of the system. In contrast, in our case such a spin-valve effect can be obtained with a single ferromagnet (see Figs. 20 and 21): by rotating the magnetization an angle $\pi/2$, T_c goes from a maximum to a minimum. The fact that only a single ferromagnet is required to achieve this effect is of practical importance since it can be challenging to control the relative magnetization orientation in magnetic multilayered structures.

IV. SUMMARY AND DISCUSSION

It was pointed out in Ref. 21 that for the case of transversal structures as depicted in Fig. 2, pure Rashba or pure Dresselhaus coupling and arbitrary magnetisation direction are insufficient for long range triplets to exist. However, although these layered structures are more restrictive in their conditions for LRT generation than lateral junctions they are nevertheless one of the most relevant for current experimental setups[10,11,54], and herein we consider the corresponding experimentally accessible effects of SO coupling as a complement to the findings of Ref. 21. We have provided a detailed exposition of the density of states and critical temperature for both the SF bilayer and SFS junction with SO coupling, highlighting in particular the signature of long range triplets.

We saw that the spectroscopic signature depends nonmonotonically on the angle of the magnetic exchange field, and that the LRT component can induce a strong peak in the density of states at zero energy for a range of magnetization directions. In addition to the large enhancement at zero energy, we see that by carefully choosing the SO coupling and exchange field strengths in the Josephson junction it is again possible to control the qualitative features of the density of states by altering the phase difference between the two superconductors *e.g.* with a loop geometry[47].

The intrinsic SO coupling present in the structures considered herein derives from their lack of inversion symmetry due to the *e.g.* junction interfaces, so-called interfacial asymmetry, and we restricted the form of this coupling to the experimentally common and, in some cases, tunable Rashba-Dresselhaus form. A lack of inversion symmetry can also derive from intrinsic noncentrosymmmetry of a crystal. This could in principle be utilised to provide a component of the SO-field in the junction direction, but to date we are not aware of such materials having been explored in experiments with SF hybrid materials. However, analytic and numerical data suggest that these materials could have significant importance for spintronic applications making use of a large triplet Cooper pair population[43].

It is also worth considering the possibility of separating the spin-orbit coupling and ferromagnetic layers, which would arguably be easier to fabricate, and we are currently pursuing this line of investigation. In this case, we would expect similar conclusions regarding when the long-range triplets leave clear spectroscopic signatures and also regarding the spin-valve effect with a single ferromagnet, as found when the SO coupling and exchange field coexist in the same material. One way to practically achieve such a setup would be to deposit a very thin layer of a heavy normal metal such as Au or Pt between a superconductor and a conventional homogeneous ferromagnet. The combination of the large atomic number Z and the broken structural inversion symmetry at the interface region would then provide the required SO coupling. With a very thin normal metal layer (of the order of a couple of nm), the proximity effect would be significantly stronger, and thus analysis of this regime is only possible with the full Usadel equations in the Riccati parameterisation developed herein.

The current analysis pertains to thin film ferromagnets. Upon increasing the length of ferromagnetic film one will increase the relative proportions of long-range to short-range triplets in the middle of the ferromagnet. For strong ferromagnets where the exchange field is a significant fraction of the Fermi energy, the quasiclassical Usadel formalism may no longer describe the system behaviour appropriately, since it assumes that the impurity scattering rate is much larger than the other energy scales involved, and the Eilenberger equation should be used instead[55].

In the previous section, we also observed that the presence of SO coupling will in many cases *increase* the critical temperature of a hybrid structure. This effect is explained through an increase in the effective energy coupled to the triplet component in the Usadel equation, which destabilizes the triplet pairs and closes that proximity channel. However, for the special geometry $\theta = -\chi = \pm\pi/4$, the linearized equations suggest that the SRTs are unaffected by the presence of SO coupling, and this is consistent with the numerical results. We also note that for the geometries with a large LRT generation, such as $\theta = 0$ and $\chi = \pm\pi/4$, the LRT generation reduces the critical temperature again. Thus, for the physical parameters considered herein, we see that there is a very slight increase in critical temperature for these geometries, but not as large as for the geometries without LRT generation.

One particularly striking result from the critical temperature calculations is that when the Rashba and Dresselhaus contribution to the SO coupling is of similar magnitude, one observes that the critical temperature can change by as much as 60% of upon changing $\theta = -\pi/4$ to $\theta = +\pi/4$, *i.e.* by a 90° rotation of the magnetic field. This implies that it is possible to create a novel kind of triplet spin valve using an SF bilayer, where the ferromagnet has a homogeneous exchange field and Rashba–Dresselhaus coupling. This is in contrast to previous suggestions for triplet spin valves, such as the one described by Fominov *et al.*, which have required trilayers with different homogeneous ferromagnets[56]. The construction of such a device is likely to have possible applications in the emerging field of superconducting spintronics[3].

Acknowledgments

The authors thank Matthias Eschrig, Camilla Espedal, and Iryna Kulagina for useful discussions and gratefully acknowledge support from the 'Outstanding Academic Fellows' programme at NTNU and COST Action MP-1201' Novel Functionalities through Optimized Confinement of Condensate and Fields'. J.L. was supported by the Research Council of Norway, Grant No. 205591 (FRINAT) and Grant No. 216700.

Appendix A: Riccati parametrization of the Usadel equation and Kupriyanov–Lukichev boundary conditions

The 4×4 components of the retarded Green's function $\hat{g}$ are not entirely independent, but can be expressed as

$$\hat{g}(z,\varepsilon)=\begin{pmatrix} g(z,+\varepsilon) & f(z,+\varepsilon) \\ -f^*(z,-\varepsilon) & -g^*(z,-\varepsilon)\end{pmatrix}, \tag{A1}$$

which suggests that the notation can be simplified by introducing the *tilde conjugation*

$$\tilde{g}(z,+\varepsilon)\equiv g^*(z,-\varepsilon). \tag{A2}$$

Moreover, the normalization condition $\hat{g}^2=1$ further constrains the possible form of $\hat{g}$ by relating the g components to the f components,

$$gg-f\tilde{f}=1, \qquad gf-f\tilde{g}=0. \tag{A3}$$

Remarkably, if we pick a suitable parametrization of $\hat{g}$, which automatically satisfies the symmetry and normalization requirements above, then both the Usadel equation and the Kupriyanov–Lukichev boundary conditions can be reduced from 4×4 to 2×2 matrix equations. In this paper, we employ the so-called *Riccati parametrization* for this purpose, which is defined by

$$\hat{g}=\begin{pmatrix} N & 0 \\ 0 & -\tilde{N}\end{pmatrix}\begin{pmatrix} 1+\gamma\tilde{\gamma} & 2\gamma \\ 2\tilde{\gamma} & 1+\tilde{\gamma}\gamma\end{pmatrix}, \tag{A4}$$

where the normalization matrices are $N\equiv(1-\gamma\tilde{\gamma})^{-1}$ and $\tilde{N}\equiv(1-\tilde{\gamma}\gamma)^{-1}$. Solving the Riccati parametrized equations for the function $\gamma(z,\varepsilon)$ in spin space is then sufficient to uniquely construct the whole Green's function $\hat{g}(z,\varepsilon)$. It is noteworthy that $\hat{g}\to1$ when $\gamma\to0$, while the elements of $\hat{g}$ diverge to infinity when $\gamma\to1$; so we see that a finite range of variation in γ parametrizes an infinite range of variation in $\hat{g}$.

We begin by deriving some basic identities, starting with the inverses of the two matrix products $N\gamma$ and $\gamma\tilde{N}$:

$$(N\gamma)^{-1}=\gamma^{-1}N^{-1}=\gamma^{-1}(1-\gamma\tilde{\gamma})=\gamma^{-1}-\tilde{\gamma}; \tag{A5}$$

$$(\gamma\tilde{N})^{-1}=\tilde{N}^{-1}\gamma^{-1}=(1-\tilde{\gamma}\gamma)\gamma^{-1}=\gamma^{-1}-\tilde{\gamma}. \tag{A6}$$

By comparison of the results above, we see that $N\gamma=\gamma\tilde{N}$. Similar calculations for other combinations of the Riccati matrices reveal that we can always move normalization matrices past gamma matrices if we also perform a tilde conjugation in the process:

$$N\gamma=\gamma\tilde{N}, \quad \tilde{N}\gamma=\gamma N, \quad N\tilde{\gamma}=\tilde{\gamma}\tilde{N}, \quad \tilde{N}\tilde{\gamma}=\tilde{\gamma}N. \tag{A7}$$

Since we intend to parametrize a differential equation, we should also try to relate the derivatives of the Riccati matrices. This can be done by differentiating the definition of N using the matrix version of the chain rule:

$$\begin{aligned}\partial_zN&=\partial_z(1-\gamma\tilde{\gamma})^{-1}\\ &=-(1-\gamma\tilde{\gamma})^{-1}\left[\partial_z(1-\gamma\tilde{\gamma})\right](1-\gamma\tilde{\gamma})^{-1}\\ &=(1-\gamma\tilde{\gamma})^{-1}\left[(\partial_z\gamma)\tilde{\gamma}+\gamma(\partial_z\tilde{\gamma})\right](1-\gamma\tilde{\gamma})^{-1}\\ &=N\left[(\partial_z\gamma)\tilde{\gamma}+\gamma(\partial_z\tilde{\gamma})\right]N.\end{aligned} \tag{A8}$$

Performing a tilde conjugation of the equation above, we get a similar result for $\partial_z\tilde{N}$. So the derivatives of the normalization matrices satisfy the following identities:

$$\partial_zN=N\left[(\partial_z\gamma)\tilde{\gamma}+\gamma(\partial_z\tilde{\gamma})\right]N, \tag{A9}$$

$$\partial_z\tilde{N}=\tilde{N}\left[(\partial_z\tilde{\gamma})\gamma+\tilde{\gamma}(\partial_z\gamma)\right]\tilde{N}. \tag{A10}$$

In addition to the identities derived above, one should note that the definition of the normalization matrix $N=(1-\gamma\tilde{\gamma})^{-1}$ can be rewritten in many forms which may be of use when simplifying Riccati parametrized expressions; examples of this include $\gamma\tilde{\gamma}=1-N^{-1}$ and $1=N-N\gamma\tilde{\gamma}$.

Now that the basic identities are in place, it is time to parametrize the Usadel equation in the ferromagnet,

$$D_F\tilde{\nabla}(\hat{g}\tilde{\nabla}\hat{g})+i\left[\varepsilon\hat{\rho}_3+\hat{M},\hat{g}\right]=0, \tag{A11}$$

where we for simplicity will let $D_F=1$ in this appendix. We begin by expanding the gauge covariant derivative $\tilde{\nabla}(\hat{g}\tilde{\nabla}\hat{g})$, and then simplify the result using the normalization condition $\hat{g}^2=1$ and its derivative $\{\hat{g},\partial_z\hat{g}\}=0$, which yields the result

$$\begin{aligned}\tilde{\nabla}\cdot(\hat{g}\tilde{\nabla}\hat{g})=\partial_z(\hat{g}\partial_z\hat{g})-i\partial_z(\hat{g}\hat{A}_z\hat{g})\\ -i[\hat{A}_z,\hat{g}\partial_z\hat{g}]-[\hat{A},\hat{g}\hat{A}\hat{g}].\end{aligned} \tag{A12}$$

We then write $\hat{g}$ in component form using Eq. (A1), and also write $\hat{A}$ on the same form using $\hat{A}=\mathrm{diag}(\underline{A},-\underline{A}^*)$. In the rest of this appendix, we will for simplicity assume that $\underline{A}$ is real, so that $\hat{A}=\mathrm{diag}(\underline{A},-\underline{A})$; in practice, this implies that $\underline{A}$ can only depend on the spin projections σ_x and σ_z. The derivation for the more general case of a complex $\hat{A}$ is almost identical. The four terms in Eq. (A12) may then be written as follows:

$$\begin{aligned}&\partial_z(\hat{g}\partial_z\hat{g})\\ &=\begin{bmatrix} \partial_z(g\partial_zg-f\partial_z\tilde{f}) & \partial_z(g\partial_zf-f\partial_z\tilde{g}) \\ \partial_z(\tilde{g}\partial_z\tilde{f}-\tilde{f}\partial_zg) & \partial_z(\tilde{g}\partial_z\tilde{g}-\tilde{f}\partial_zf)\end{bmatrix};\end{aligned} \tag{A13}$$

$$\begin{aligned}&\partial_z(\hat{g}\hat{A}\hat{g})\\ &=\begin{bmatrix} \partial_z(gAg+fA\tilde{f}) & \partial_z(gAf+fA\tilde{g}) \\ -\partial_z(\tilde{g}A\tilde{f}+\tilde{f}Ag) & -\partial_z(\tilde{g}A\tilde{g}+\tilde{f}Af)\end{bmatrix};\end{aligned} \tag{A14}$$

$$\begin{aligned}&[\hat{A},\hat{g}\partial_z\hat{g}]\\ &=\begin{bmatrix} [A,g\partial_zg-f\partial_z\tilde{f}] & \{A,g\partial_zf-f\partial_z\tilde{g}\} \\ -\{A,\tilde{g}\partial_z\tilde{f}-\tilde{f}\partial_zg\} & -[A,\tilde{g}\partial_z\tilde{g}-\tilde{f}\partial_zf]\end{bmatrix};\end{aligned} \tag{A15}$$

$$\begin{aligned}&[\hat{A},\hat{g}\hat{A}\hat{g}]\\ &=\begin{bmatrix} [A,gAg+fA\tilde{f}] & \{A,gAf+fA\tilde{g}\} \\ \{A,\tilde{g}A\tilde{f}+\tilde{f}Ag\} & [A,\tilde{g}A\tilde{g}+\tilde{f}Af]\end{bmatrix}.\end{aligned} \tag{A16}$$

Substituting these results back into Eq. (A12), we can find the

upper blocks of the covariant derivative $\tilde{\nabla}\cdot(\hat{g}\tilde{\nabla}\hat{g})$,

$$\begin{aligned}&[\tilde{\nabla}\cdot(\hat{g}\tilde{\nabla}\hat{g})]^{(1,1)}\\&=\partial_z(g\partial_z g-f\partial_z\tilde{f})-i\partial_z(gA_z g+fA_z\tilde{f})\\&-i[A_z,\,g\partial_z g-f\partial_z\tilde{f}]-[\underline{A},\,g\underline{A}g+f\underline{A}\tilde{f}]\,,\end{aligned}\tag{A17}$$

$$\begin{aligned}&[\tilde{\nabla}\cdot(\hat{g}\tilde{\nabla}\hat{g})]^{(1,2)}\\&=\partial_z(g\partial_z f-f\partial_z\tilde{g})-i\partial_z(gA_z f+fA_z\tilde{g})\\&-i\{A_z,\,g\partial_z f-f\partial_z\tilde{g}\}-\{\underline{A},\,g\underline{A}f+f\underline{A}\tilde{g}\}\,.\end{aligned}\tag{A18}$$

In this context, the notation $\hat{M}^{(n,m)}$ refers to the n'th row and m'th column in Nambu space. Since the Green's function $\hat{g}$ and background field $\hat{A}$ also have a structure in spin space, the $(1,1)$ element in Nambu space is the upper-left 2×2 block of the matrix, and the $(1,2)$ element is the upper-right one.

There are two kinds of expressions that recur in the equations above, namely the components of $\hat{g}\partial_z\hat{g}$, and the components of $\hat{g}\hat{A}\hat{g}$. After we substitute in the Riccati parametrization $g=2N-1$ and $f=2N\gamma$, these components take the form:

$$\begin{aligned}[\hat{g}\partial_z\hat{g}]^{(1,1)}&=g\partial_z g-f\partial_z\tilde{f}\\&=2N\left[(\partial_z\gamma)\tilde{\gamma}-\gamma(\partial_z\tilde{\gamma})\right]N\,;\end{aligned}\tag{A19}$$

$$\begin{aligned}[\hat{g}\partial_z\hat{g}]^{(1,2)}&=g\partial_z f-f\partial_z\tilde{g}\\&=2N\left[(\partial_z\gamma)-\gamma(\partial_z\tilde{\gamma})\gamma\right]\tilde{N}\,;\end{aligned}\tag{A20}$$

$$\begin{aligned}[\hat{g}\hat{A}\hat{g}]^{(1,1)}&=g\underline{A}g+f\underline{A}\tilde{f}\\&=4N(\underline{A}+\gamma\underline{A}\tilde{\gamma})N-2\{\underline{A},\,N\}+\underline{A}\,;\end{aligned}\tag{A21}$$

$$\begin{aligned}[\hat{g}\hat{A}\hat{g}]^{(1,2)}&=g\underline{A}f+f\underline{A}\tilde{g}\\&=4N(\underline{A}\gamma+\gamma\underline{A})\tilde{N}-2\{\underline{A},\,N\gamma\}\,.\end{aligned}\tag{A22}$$

If we explicitly calculate the commutators of $\hat{A}$ with the two matrices $\hat{g}\partial_z\hat{g}$ and $\hat{g}\hat{A}\hat{g}$, then we find:

$$\begin{aligned}[\hat{A},\,\hat{g}\partial_z\hat{g}]^{(1,1)}&=[\underline{A},\,g\partial_z g-f\partial_z\tilde{f}]\\&=2N(1-\gamma\tilde{\gamma})\underline{A}N\left[(\partial_z\gamma)\tilde{\gamma}-\gamma(\partial_z\tilde{\gamma})\right]N\\&-2N\left[(\partial_z\gamma)\tilde{\gamma}-\gamma(\partial_z\tilde{\gamma})\right]N\underline{A}(1-\gamma\tilde{\gamma})N\,;\end{aligned}\tag{A23}$$

$$\begin{aligned}[\hat{A},\,\hat{g}\partial_z\hat{g}]^{(1,2)}&=\{\underline{A},\,g\partial_z f-f\partial_z\tilde{g}\}\\&=2N(1-\gamma\tilde{\gamma})\underline{A}N\left[(\partial_z\gamma)-\gamma(\partial_z\tilde{\gamma})\gamma\right]\tilde{N}\\&+2N\left[(\partial_z\gamma)-\gamma(\partial_z\tilde{\gamma})\gamma\right]\tilde{N}\underline{A}(1-\tilde{\gamma}\gamma)\tilde{N}\,;\end{aligned}\tag{A24}$$

$$\begin{aligned}[\hat{A},\,\hat{g}\hat{A}\hat{g}]^{(1,1)}&=[\underline{A},\,g\underline{A}g+f\underline{A}\tilde{f}]\\&=4\underline{A}N(\underline{A}+\gamma\underline{A}\tilde{\gamma})N\\&-4N(\underline{A}+\gamma\underline{A}\tilde{\gamma})N\underline{A}\\&-2[\underline{A}^2,\,N]\,;\end{aligned}\tag{A25}$$

$$\begin{aligned}[\hat{A},\,\hat{g}\hat{A}\hat{g}]^{(1,2)}&=\{\underline{A},\,g\underline{A}f+f\underline{A}\tilde{g}\}\\&=4\underline{A}N(\underline{A}\gamma+\gamma\underline{A})\tilde{N}\\&+4N(\underline{A}\gamma+\gamma\underline{A})\tilde{N}\underline{A}\\&-4\underline{A}N\gamma\underline{A}-2\{\underline{A}^2,\,N\gamma\}\,.\end{aligned}\tag{A26}$$

If we instead differentiate the aforementioned matrices with respect to z, we obtain:

$$\begin{aligned}[\partial_z(\hat{g}\partial_z\hat{g})]^{(1,1)}&=\partial_z(g\partial_z g-f\partial_z\tilde{f})\\&=2N[(\partial_z^2\gamma)+2(\partial_z\gamma)\tilde{N}\tilde{\gamma}(\partial_z\gamma)]\tilde{\gamma}N\\&-2N\gamma[(\partial_z^2\tilde{\gamma})+2(\partial_z\tilde{\gamma})N\gamma(\partial_z\tilde{\gamma})]N\,;\end{aligned}\tag{A27}$$

$$\begin{aligned}[\partial_z(\hat{g}\partial_z\hat{g})]^{(1,2)}&=\partial_z(g\partial_z f-f\partial_z\tilde{g})\\&=2N[(\partial_z^2\gamma)+2(\partial_z\gamma)\tilde{N}\tilde{\gamma}(\partial_z\gamma)]\tilde{N}\\&-2N\gamma[(\partial_z^2\tilde{\gamma})+2(\partial_z\tilde{\gamma})N\gamma(\partial_z\tilde{\gamma})]\gamma\tilde{N}\,;\end{aligned}\tag{A28}$$

$$\begin{aligned}[\partial_z(\hat{g}\underline{A}\hat{g})]^{(1,1)}&=\partial_z(g\underline{A}g+f\underline{A}\tilde{f})\\&=2N(1+\gamma\tilde{\gamma})\underline{A}N[\gamma(\partial_z\tilde{\gamma})+(\partial_z\gamma)\tilde{\gamma}]N\\&+2N[\gamma(\partial_z\tilde{\gamma})+(\partial_z\gamma)\tilde{\gamma}]N\underline{A}(1+\gamma\tilde{\gamma})N\\&+4N\gamma\underline{A}\tilde{N}[(\partial_z\tilde{\gamma})+\tilde{\gamma}(\partial_z\gamma)\tilde{\gamma}]N\\&+4N[(\partial_z\gamma)+\gamma(\partial_z\tilde{\gamma})\gamma]\tilde{N}\underline{A}\tilde{\gamma}N\,;\end{aligned}\tag{A29}$$

$$\begin{aligned}[\partial_z(\hat{g}\underline{A}\hat{g})]^{(1,2)}&=\partial_z(g\underline{A}f+f\underline{A}\tilde{g})\\&=2N(1+\gamma\tilde{\gamma})\underline{A}N[(\partial_z\gamma)+\gamma(\partial_z\tilde{\gamma})\gamma]\tilde{N}\\&+2N[(\partial_z\gamma)+\gamma(\partial_z\tilde{\gamma})\gamma]\tilde{N}\underline{A}(1+\tilde{\gamma}\gamma)\tilde{N}\\&+4N\gamma\underline{A}\tilde{N}[\tilde{\gamma}(\partial_z\gamma)+(\partial_z\tilde{\gamma})\gamma]\tilde{N}\\&+4N[\gamma(\partial_z\tilde{\gamma})+(\partial_z\gamma)\tilde{\gamma}]N\underline{A}\tilde{\gamma}\tilde{N}\,.\end{aligned}\tag{A30}$$

Combining all of the equations above, we can express Eqs. (A17) and (A18) using Riccati matrices. In order to isolate the second-order derivative $\partial_z^2\gamma$ from these, the trick is to multiply Eq. (A17) by γ from the right, and subsequently subtract the result from Eq. (A18):

$$\begin{aligned}&\frac{1}{2}N^{-1}\left\{[\tilde{\nabla}\cdot(\hat{g}\tilde{\nabla}\hat{g})]^{(1,2)}-[\tilde{\nabla}\cdot(\hat{g}\tilde{\nabla}\hat{g})]^{(1,1)}\gamma\right\}\\&\quad=\partial_z^2\gamma+2(\partial_z\gamma)\tilde{N}\tilde{\gamma}(\partial_z\gamma)\\&\quad-2i(A_z+\gamma A_z\tilde{\gamma})N(\partial_z\gamma)-2i(\partial_z\gamma)\tilde{N}(A_z+\tilde{\gamma}A_z\gamma)\\&\quad-2(\underline{A}\gamma+\gamma\underline{A})\tilde{N}(\underline{A}+\tilde{\gamma}\underline{A}\gamma)-\underline{A}^2\gamma+\gamma\underline{A}^2\,.\end{aligned}\tag{A31}$$

If we finally rewrite $[\tilde{\nabla}\cdot(\hat{g}\tilde{\nabla}\hat{g})]^{(1,1)}$ and $[\tilde{\nabla}\cdot(\hat{g}\tilde{\nabla}\hat{g})]^{(1,2)}$ in the equation above by substituting in the Usadel equation (A11), then we obtain the following equation for the Riccati matrix γ:

$$\begin{aligned}\partial_z^2\gamma=&-2i\varepsilon\gamma-i\underline{h}\cdot(\underline{\sigma}\gamma-\gamma\underline{\sigma}^*)-2(\partial_z\gamma)\tilde{N}\tilde{\gamma}(\partial_z\gamma)\\&+2i(A_z+\gamma A_z\tilde{\gamma})N(\partial_z\gamma)+2i(\partial_z\gamma)\tilde{N}(A_z+\tilde{\gamma}A_z\gamma)\\&+2(\underline{A}\gamma+\gamma\underline{A})\tilde{N}(\underline{A}+\tilde{\gamma}\underline{A}\gamma)+\underline{A}^2\gamma-\gamma\underline{A}^2\,.\end{aligned}\tag{A32}$$

The corresponding equation for $\tilde{\gamma}$ can be found by tilde conjugation of the above. After restoring the diffusion coefficient D_F, and generalizing the derivation to a complex SO field $\underline{A}$, the above result takes the form shown in Eq. (5).

After parametrizing the Usadel equation, the next step is to do the same to the Kupriyanov–Lukichev boundary conditions. The gauge covariant version of Eq. (2) may be written

$$2L_n\zeta_n\hat{g}_n\tilde{\nabla}\hat{g}_n=[\hat{g}_1,\hat{g}_2]\,,\tag{A33}$$

which upon expanding the covariant derivative $\hat{g}\tilde{\nabla}\hat{g}$ becomes

$$\hat{g}_n\partial_z\hat{g}_n = \frac{1}{2}\Omega_n[\hat{g}_1,\hat{g}_2] + i\hat{g}_n[\hat{A}_z,\hat{g}_n]\,, \tag{A34}$$

where we have introduced the notation $\Omega_n \equiv 1/L_n\zeta_n$ for the interface parameter. We will now restrict our attention to the (1,1) and (1,2) components of the above,

$$\begin{aligned} g_n\partial_z g_n - f_n\partial_z\tilde{f}_n &= \frac{1}{2}\Omega_n(g_1g_2 - g_2g_1 - f_1\tilde{f}_2 + f_2\tilde{f}_1) \\ &\quad + ig_n[A_z,g_n] + if_n\{A_z,\tilde{f}_n\}\,, \end{aligned} \tag{A35}$$

$$\begin{aligned} g_n\partial_z f_n - f_n\partial_z\tilde{g}_n &= \frac{1}{2}\Omega_n(g_1f_2 - g_2f_1 - f_1\tilde{g}_2 + f_2\tilde{g}_1) \\ &\quad + ig_n\{A_z,f_n\} + if_n[A_z,\tilde{g}_n]\,. \end{aligned} \tag{A36}$$

Substituting the Riccati parametrizations $g_n = 2N_n - 1$ and $f_n = 2N_n\gamma_n$ in the above, we then obtain:

$$\begin{aligned} N_n[(\partial_z\gamma_n)\tilde{\gamma}_n - \gamma_n(\partial_z\tilde{\gamma}_n)]N_n &= \Omega_nN_1(1-\gamma_1\tilde{\gamma}_2)N_2 \\ &\quad - \Omega_nN_2(1-\gamma_2\tilde{\gamma}_1)N_1 \\ &\quad - iN_n(1-\gamma_n\tilde{\gamma}_n)\underline{A}N_n \\ &\quad - iN_n\underline{A}(1-\gamma_n\tilde{\gamma}_n)N_n \\ &\quad + 2iN_n(\underline{A}+\gamma_n\underline{A}\tilde{\gamma}_n)N_n\,, \end{aligned} \tag{A37}$$

$$\begin{aligned} N_n[(\partial_z\gamma_n) - \gamma_n(\partial_z\tilde{\gamma}_n)\gamma_n]\tilde{N}_n &= \Omega_nN_1(1-\gamma_1\tilde{\gamma}_2)\gamma_2\tilde{N}_2 \\ &\quad - \Omega_nN_2(1-\gamma_2\tilde{\gamma}_1)\gamma_1\tilde{N}_1 \\ &\quad + iN_n(1+\gamma_n\tilde{\gamma}_n)\underline{A}\gamma_n\tilde{N}_n \\ &\quad + iN_n\gamma_n\underline{A}(1+\tilde{\gamma}_n\gamma_n)\tilde{N}_n\,. \end{aligned} \tag{A38}$$

If we then multiply Eq. (A37) by γ_n from the right, subtract this from Eq. (A38), and divide by N_n from the left, then we obtain the following boundary condition for γ_n:

$$\begin{aligned} \partial_z\gamma_n &= \Omega_n(1-\gamma_1\tilde{\gamma}_2)N_2(\gamma_2-\gamma_n) \\ &\quad + \Omega_n(1-\gamma_2\tilde{\gamma}_1)N_1(\gamma_n-\gamma_1) \\ &\quad + i\{A_z,\gamma_n\}\,. \end{aligned} \tag{A39}$$

When we evaluate the above for $n=1$ and $n=2$, then it simplifies to the following:

$$\partial_z\gamma_1 = \Omega_1(1-\gamma_1\tilde{\gamma}_2)N_2(\gamma_2-\gamma_1) + i\{A_z,\gamma_1\}\,, \tag{A40}$$

$$\partial_z\gamma_2 = \Omega_2(1-\gamma_2\tilde{\gamma}_1)N_1(\gamma_2-\gamma_1) + i\{A_z,\gamma_2\}\,. \tag{A41}$$

The boundary conditions for $\partial_z\tilde{\gamma}_1$ and $\partial_z\tilde{\gamma}_2$ are found by tilde conjugating the above. If we generalize the derivation to a complex SO field $\underline{A}$, and substitute back $\Omega_n \equiv 1/L_n\zeta_n$ in the result, then we arrive at Eq. (6).

Appendix B: Derivation of the self-consistency equation for Δ

For completeness, we here present a detailed derivation of the self-consistency equation for the BCS order parameter[62] in a quasiclassical framework. Similar derivations can also be found in Refs. 48,63–66. In this paper, we follow the convention where the Keldysh component of the anomalous Green's function is defined as

$$F^K_{\sigma\sigma'}(\underline{r},t;\underline{r}',t') \equiv -i\langle[\psi_\sigma(\underline{r},t),\psi_{\sigma'}(\underline{r},t)]\rangle\,, \tag{B1}$$

where $\psi_\sigma(\underline{r},t)$ is the spin-dependent fermion annihilation operator, and the superconducting gap is defined as

$$\Delta(\underline{r},t) \equiv \lambda\langle\psi_\uparrow(\underline{r},t)\,\psi_\downarrow(\underline{r},t)\rangle\,, \tag{B2}$$

where $\lambda > 0$ is the electron–electron coupling constant in the BCS theory. For the rest of this appendix, we will also assume that we work in an electromagnetic gauge where Δ is a purely real quantity. Comparing Eqs. (B1) and (B2), and using the fermionic anticommutation relation

$$\psi_\uparrow(\underline{r},t)\,\psi_\downarrow(\underline{r},t) = -\psi_\downarrow(\underline{r},t)\,\psi_\uparrow(\underline{r},t)\,, \tag{B3}$$

we see that the superconducting gap $\Delta(\underline{r},t)$ can be expressed in terms of the Green's functions in two different ways,

$$\Delta(\underline{r},t) = \frac{i\lambda}{2}F^K_{\uparrow\downarrow}(\underline{r},t;\underline{r},t)\,, \tag{B4}$$

$$\Delta(\underline{r},t) = -\frac{i\lambda}{2}F^K_{\downarrow\uparrow}(\underline{r},t;\underline{r},t)\,. \tag{B5}$$

We may then perform a quasiclassical approximation by first switching to Wigner mixed coordinates, then Fourier transforming the relative coordinates, then integrating out the energy dependence, and finally averaging the result over the Fermi surface to obtain the isotropic part. The resulting equations for the superconducting gap are

$$\Delta(\underline{r},t) = \frac{1}{4}N_0\lambda\int d\varepsilon\, f^K_{\uparrow\downarrow}(\underline{r},t,\varepsilon)\,, \tag{B6}$$

$$\Delta(\underline{r},t) = -\frac{1}{4}N_0\lambda\int d\varepsilon\, f^K_{\downarrow\uparrow}(\underline{r},t,\varepsilon)\,, \tag{B7}$$

where $f^K_{\sigma\sigma'}$ is the quasiclassical counterpart to $F^K_{\sigma\sigma'}$, ε is the quasiparticle energy, and N_0 is the density of states per spin at the Fermi level.

In the equilibrium case, the Keldysh component $\hat{g}^K$ can be expressed in terms of the retarded and advanced components of the Green's function,

$$\hat{g}^K = (\hat{g}^R - \hat{g}^A)\tanh(\varepsilon/2T)\,, \tag{B8}$$

and the advanced Green's function may again be expressed in terms of the retarded one,

$$\hat{g}^A = -\rho_3\hat{g}^{R\dagger}\rho_3\,, \tag{B9}$$

which implies that the Keldysh component can be expressed entirely in terms of the retarded component,

$$\hat{g}^K = (\hat{g}^R - \rho_3\hat{g}^{R\dagger}\rho_3)\tanh(\varepsilon/2T)\,. \tag{B10}$$

If we extract the relevant anomalous components $f^K_{\uparrow\downarrow}$ and $f^K_{\downarrow\uparrow}$ from the above, we obtain the results

$$f^K_{\uparrow\downarrow} = [f^R_{\uparrow\downarrow}(\underline{r},+\varepsilon) + f^R_{\downarrow\uparrow}(\underline{r},-\varepsilon)]\tanh(\varepsilon/2T)\,, \tag{B11}$$

$$f^K_{\downarrow\uparrow} = [f^R_{\downarrow\uparrow}(\underline{r},+\varepsilon) + f^R_{\uparrow\downarrow}(\underline{r},-\varepsilon)]\tanh(\varepsilon/2T)\,. \tag{B12}$$

We then switch to a singlet/triplet-decomposition of the retarded component f^R, where the singlet component is described by a scalar function f_s, and the triplet component by the so-called d-vector (d_x, d_y, d_z). This parametrization is defined by the matrix equation

$$f^R = (f_s + \underline{d}\cdot\underline{\sigma})i\sigma_y\,, \tag{B13}$$

or in component form,

$$\begin{pmatrix} f^R_{\uparrow\uparrow} & f^R_{\uparrow\downarrow} \\ f^R_{\downarrow\uparrow} & f^R_{\downarrow\downarrow} \end{pmatrix} = \begin{pmatrix} id_y - d_x & d_z + f_s \\ d_z - f_s & id_y + d_x \end{pmatrix}. \tag{B14}$$

Parametrizing Eqs. (B11) and (B12) according to Eq. (B14), we obtain

$$\begin{aligned} f^K_{\uparrow\downarrow}(\underline{r},\varepsilon) &= [d_z(\underline{r},+\varepsilon) + f_s(\underline{r},+\varepsilon) \\ &\quad + d_z(\underline{r},-\varepsilon) - f_s(\underline{r},-\varepsilon)]\tanh(\varepsilon/2T)\,, \end{aligned} \tag{B15}$$

$$\begin{aligned} f^K_{\uparrow\downarrow}(\underline{r},\varepsilon) &= [d_z(\underline{r},+\varepsilon) - f_s(\underline{r},+\varepsilon) \\ &\quad + d_z(\underline{r},-\varepsilon) + f_s(\underline{r},-\varepsilon)]\tanh(\varepsilon/2T)\,. \end{aligned} \tag{B16}$$

The triplet component d_z can clearly be eliminated from the above equations by subtracting Eq. (B15) from Eq. (B16),

$$f^K_{\uparrow\downarrow} - f^K_{\downarrow\uparrow} = 2[f_s(\underline{r},\varepsilon) - f_s(\underline{r},-\varepsilon)]\tanh(\varepsilon/2T)\,, \tag{B17}$$

and a matching expression for the superconducting gap can be acquired by adding Eqs. (B6) and (B7),

$$2\Delta(\underline{r}) = \frac{1}{4}N_0\lambda\int \mathrm{d}\varepsilon\,[f^K_{\uparrow\downarrow}(\underline{r},\varepsilon) - f^K_{\downarrow\uparrow}(\underline{r},\varepsilon)]\tanh(\varepsilon/2T)\,. \tag{B18}$$

By comparing the two results above, we finally arrive at an equation for the superconducting gap which only depends on the singlet component of the quasiclassical Green's function:

$$\Delta(\underline{r}) = \frac{1}{4}N_0\lambda\int \mathrm{d}\varepsilon\,[f_s(\underline{r},\varepsilon) - f_s(\underline{r},-\varepsilon)]\tanh(\varepsilon/2T)\,. \tag{B19}$$

If the integral above is performed for all real values of ε, it turns out to be logarithmically divergent *e.g.* for a bulk superconductor. However, physically, the range of energies that should be integrated over is restricted by the energy spectra of the phonons that mediate the attractive electron–electron interactions in the superconductor. This issue may therefore be resolved by introducing a Debye cutoff ω_c, such that we only integrate over the region where $|\varepsilon| < \omega_c$. Including the integration range, the gap equation is therefore

$$\Delta(\underline{r}) = \frac{1}{4}N_0\lambda\int_{-\omega_c}^{\omega_c} \mathrm{d}\varepsilon\,[f_s(\underline{r},\varepsilon) - f_s(\underline{r},-\varepsilon)]\tanh(\varepsilon/2T)\,. \tag{B20}$$

The equation above can however be simplified even further. First of all, both $f_s(\varepsilon) - f_s(-\varepsilon)$ and $\tanh(\varepsilon/2T)$ are clearly antisymmetric functions of ε, which means that the product is a symmetric function, and so it is sufficient to perform an integral over positive values of ε,

$$\Delta(\underline{r}) = \frac{1}{2}N_0\lambda\int_{0}^{\omega_c} \mathrm{d}\varepsilon\,[f_s(\underline{r},\varepsilon) - f_s(\underline{r},-\varepsilon)]\tanh(\varepsilon/2T)\,. \tag{B21}$$

However, because of the term $f_s(\underline{r},-\varepsilon)$, we still need to know the Green's function for negative values of ε before we can calculate the gap. On the other hand, the singlet component of the quasiclassical Green's functions also has a symmetry when the superconducting gauge is chosen as real

$$f_s(\underline{r},\varepsilon) = -f_s^*(\underline{r},-\varepsilon)\,, \tag{B22}$$

which implies that

$$f_s(\underline{r},\varepsilon) - f_s(\underline{r},-\varepsilon) = 2\,\mathrm{Re}\{f_s(\underline{r},\varepsilon)\}\,. \tag{B23}$$

Substituting Eq. (B23) into Eq. (B21), the gap equation takes a particularly simple form, which only depends on the real part of the singlet component $f_s(\underline{r},\varepsilon)$ for positive energies ε:

$$\Delta(\underline{r}) = N_0\lambda\int_{0}^{\omega_c} \mathrm{d}\varepsilon\,\mathrm{Re}\{f_s(\underline{r},\varepsilon)\}\tanh(\varepsilon/2T)\,. \tag{B24}$$

Let us now consider the case of a BCS bulk superconductor, which has a singlet component given by the equation

$$f_s(\varepsilon) = \frac{\Delta}{\sqrt{\varepsilon^2 - \Delta^2}}\,, \tag{B25}$$

so that the gap equation may be written as

$$\Delta = N_0\lambda\int_{0}^{\omega_c} \mathrm{d}\varepsilon\,\mathrm{Re}\left\{\frac{\Delta}{\sqrt{\varepsilon^2 - \Delta^2}}\right\}\tanh(\varepsilon/2T)\,. \tag{B26}$$

The part in the curly braces is only real when $|\varepsilon| \geq \Delta$, which means that the equation can be simplified by changing the lower integration limit to Δ. After also dividing the equation by $\Delta N_0\lambda$, we then obtain the self-consistency equation

$$\frac{1}{N_0\lambda} = \int_{\Delta}^{\omega_c} \mathrm{d}\varepsilon\,\frac{\tanh(\varepsilon/2T)}{\sqrt{\varepsilon^2 - \Delta^2}}\,. \tag{B27}$$

For the zero-temperature case, where $T \to 0$ and $\Delta \to \Delta_0$, performing the above integral and reordering the result yields

$$\omega_c = \Delta_0\cosh(1/N_0\lambda)\,. \tag{B28}$$

Using the above equation for ω_c, and the well-known result

$$\frac{\Delta_0}{T_c} = \frac{\pi}{e^\gamma}\,, \tag{B29}$$

where $\gamma \approx 0.57722$ is the Euler–Mascheroni constant, we can finally rewrite Eq. (B24) as:

$$\Delta(\underline{r}) = N_0\lambda\int_{0}^{\Delta_0\cosh(1/N_0\lambda)} \mathrm{d}\varepsilon\,\mathrm{Re}\{f_s(\underline{r},\varepsilon)\}\tanh\left(\frac{\pi}{2e^\gamma}\frac{\varepsilon/\Delta_0}{T/T_c}\right). \tag{B30}$$

This version of the gap equation is particularly well-suited for numerical simulations. One advantage is that we only need to know the Green's function for positive energies, which halves the number of energies that we need to solve the Usadel equation for. The equation also takes a particularly simple form if we use energy units where $\Delta_0 = 1$ and temperature units where $T_c = 1$, which is common practice in such simulations.

[1] A. I. Buzdin, Rev. Mod. Phys. **77**, 935 (2005).
[2] F.S. Bergeret, A.F. Volkov and K.B. Efetov, Rev. Mod. Phys. **77**, 1321 (2005).
[3] J. Linder and J. W. A. Robinson, Nature Physics **11**, 307 (2015).
[4] F. Bergeret, A. Volkov, and K. Efetov, Phys. Rev. Lett. **86**, 4096 (2001).
[5] M. Eschrig, J. Kopu, J. C. Cuevas, and Gerd Schön, Phys. Rev. Lett. **90**, 137003 (2003)
[6] K. Halterman, P. H. Barsic, and O. T. Valls, Phys. Rev. Lett. **99**, 127002 (2007)
[7] A. Cottet, Phys. Rev. Lett. **107**, 177001 (2011).
[8] Y. Asano, Y. Sawa, Y. Tanaka, and A. A. Golubov, Phys. Rev. B **76**, 224525 (2007).
[9] Y. Kalcheim *et al.*, Phys. Rev. B **89**, 180506(R) (2014).
[10] J. W. A. Robinson, J. D. Witt, and M. Blamire, Science **329**, 59 (2010)
[11] T. S. Khaire *et al.*, Phys. Rev. Lett. **104**, 137002 (2010).
[12] M. Alidoust and J. Linder, Phys. Rev. B **82**, 224504 (2010); I. B. Sperstad *et al.*, Phys. Rev. B **78**, 104509 (2008).
[13] Luka Trifunovic and Zoran Radović Phys. Rev. B **82**, 020505(R) (2010)
[14] I. Sosnin, H. Cho, V. T. Petrashov, and A. F. Volkov Phys. Rev. Lett. **96**, 157002 (2006)
[15] Z. Shomali, M. Zareyan, W. Belzig, NJP **13**, 083033 (2012).
[16] G. Annunziata *et al.*, Phys. Rev. B **83**, 060508(R) (2011)
[17] I. V. Bobkova and A. M. Bobkov, Phys. Rev. Lett. **108**, 197002 (2012)
[18] F. Konschelle, J. Cayssol, and A. Buzdin, Phys. Rev. B **82**, 180509(R) (2010)
[19] M. Houzet, Phys. Rev. Lett. **101**, 057009 (2008)
[20] G. Annunziata, D. Manske, and J. Linder, Phys. Rev. B **86**, 174514 (2012).
[21] F.S. Bergeret and I.V. Tokatly, Phys. Rev. B. **89**, 134517 (2014).
[22] F.S. Bergeret and I.V. Tokatly, Phys. Rev. Lett. **110**, 117003 (2013).
[23] A.A. Abrikosov and L.P. Gor'kov, Sov. Phys. JETP **15**, 752 (1962).
[24] M. Eschrig, Physics Today **64**(1), 43 (2011).
[25] K.V. Samokhin, Ann. Phys. **324**, 2385 (2009).
[26] V.M. Edelstein, Phys. Rev. B. **67**, 020505 (2003).
[27] J. Linder and T. Yokoyama, Phys. Rev. Lett. **106**, 237201 (2011).
[28] K. Sun and N. Shah, Phys. Rev. B 91, 144508 (2015)
[29] K. Usadel, Phys. Rev. Lett. **25**, 507 (1970).
[30] W. Belzig *et al.*, Superlattices and Microstructures, **5**, 25 (1999).
[31] V. Chandrasekhar, arXiv:0312507.
[32] M.Y. Kupriyanov and V.F. Lukichev, Sov. Phys. JETP **67**, 1163 (1988).
[33] N. Schopohl and K. Maki, Phys. Rev. B. **52**, 490 (1995); N. Schopohl, arXiv:cond-mat/9804064.
[34] F. Konschelle, Eur. Phys. J. B 87, 119 (2014)
[35] C. Gorini, P. Schwab, R. Raimondi, and A. L. Shelankov, Phys. Rev. B 82, 195316 (2010)
[36] A.I. Buzdin, Phys. Rev. Lett. **101**, 107005 (2008); A.A. Raynoso *et al.*, Phys. Rev. B. **86**, 214519 (2012); I. Kulagina and J. Linder, Phys. Rev. B **90**, 054504 (2014); F.S. Bergeret and I.V. Tokatly, preprint, arXiv:1409.4563.
[37] P. Fulde and R.A. Ferrell, Phys. Rev. **135**, A550 (1964); A.I. Larkin and Y.N. Ovchinnikov, Sov. Phys. JETP **20** 762 (1965).
[38] X. Liu, J. K. Jain, and C.-X. Liu, Phys. Rev. Lett. **113**, 227002 (2014).
[39] S. Kawabata *et al.*, J. Phys. Soc. Jpn. **82**, 124702 (2013).
[40] B. Pannetier and H. Courtois, J. Low. Temp. Phys. **118**, 599 (2000).
[41] T. Yokoyama, Y. Tanaka and A.A. Golubov, Phys. Rev. B **72**, 052512 (2005);T. Yokoyama, Y. Tanaka and A.A. Golubov, Phys. Rev. B **73**, 094501 (2006).
[42] J. C. Hammer, J. C. Cuevas, F. S. Bergeret, and W. Belzig, Phys. Rev. B **76**, 064514 (2007)
[43] S.H. Jacobsen and J. Linder, arXiv:1503.03500.
[44] R. Balian and N.R. Werthamer, Phys. Rev. **131**, 1553 (1963).
[45] A.P. Mackenzie and Y. Maeno, Rev. Mod. Phys. **75**, 657 (2003).
[46] Y. Tanaka and A. Golubov, Phys. Rev. Lett. **98**, 037003 (2007).
[47] H. le Sueur *et al*, Phys. Rev. Lett. **100**, 197002 (2008).
[48] M. Tinkham, *Introduction to Superconductivity*, (McGraw-Hill, 1996), 2nd Edition.
[49] F. Zhou, P. Charlat, and B. Pannetier, J. Low Temp. Phys. **110**, 841 (1998).
[50] M. Eschrig, J. Kopu, A. Konstandin, J.C. Cuevas, M. Fogelström, G. Schön, Advances in Solid State Physics Vol. 44, pp. 533-546 (2004)
[51] I. S. Burmistrov and N. M. Chtchelkatchev, Phys. Rev. B **72**, 144520 (2005).
[52] J. Linder and K. Halterman, Phys. Rev. B **90**, 104502 (2014)
[53] Y. Fominov *et al.*, Phys. Rev. B. **66**, 014507 (2002).
[54] T. Kontos *et al.*, Phys. Rev. Lett. **86**, 304 (2001); V.V. Ryazanov *et al.*, Phys. Rev. Lett. **86**, 2427 (2001).
[55] G. Eilenberger, Sov. Phys. JETP **214**, 195 (1968)
[56] Y. Fominov *et al.*, JETP Lett., **91**, 308 (2010).
[57] J. Y. Gu *et al.*, Phys. Rev. Lett. **89**, 267001 (2002).
[58] I. Moraru, W. P. Pratt Jr., and N. Birge, Phys. Rev. Lett. **96**, 037004 (2006).
[59] J. Zhu, I. N. Krivorotov, K. Halterman, and O. T. Valls, Phys. Rev. Lett. **105**, 207002 (2010).
[60] P. V. Leksin *et al.*, Phys. Rev. Lett. **109**, 057005 (2012).
[61] N. Banerjee *et al.*, Nature Commun. **5**, 3048 (2014).
[62] J. Bardeen, L. N. Cooper, and J. R. Schrieffer, Phys. Rev. **108**, 1175 (1957).
[63] N. Kopnin. *Theory of Nonequilibrium Superconductivity* (2001).
[64] A. A. Abrikosov, L.P. Gorkov, I.E. Dzyaloshinski. *Quantum Field Theoretical Methods in Statistical Physics* (Volume 4 of International Series of Monographs in Natural Philosophy) (1965).
[65] J. W. Serene and D. Rainer, Phys. Rep. **101**, 4 (1983).
[66] P. G. deGennes. *Superconductivity Of Metals And Alloys*. (1999).
[67] G. Dresselhaus, Phys. Rev. 100, 580 (1955); E. Rashba, Fiz. Tverd. Tela (Leningrad) 2, 1224 (1960), [Sov. Phys. Solid State 2, 1109 (1960)].
[68] E. Bauer, M. Sigrist. *Non-Centrosymmetric Superconductors: Introduction and overview.* Springer (2012).

MATLAB CLASSES

The following pages contain the implementation of five MATLAB classes. The class `SpinVector` implements a data structure for describing objects with a vector structure in geometric space and a matrix structure in spin space. This encompasses e.g. the Pauli vector $\underline{\boldsymbol{\sigma}}$ and the spin-orbit field $\underline{\mathbf{A}}$. The class `State` implements a data structure for describing the state of a physical system at a given position and energy. This is essentially just a wrapper class for the Riccati parameters $\underline{\gamma}$ and $\underline{\tilde{\gamma}}$ and their derivatives, but also includes some useful methods for e.g. calculating the singlet/triplet projections, calculating the local density of states, and converting Riccati matrices to/from the 16 element state vectors that `bvp6c` expects.

The by far largest class is `Metal`. This class describes the physical state throughout a metal, where the core structure is a matrix of `State` objects that correspond to a discretized range of positions and energies. The class also contains subroutines for solving the Usadel equation with three different boundary conditions: tunneling interfaces, spin-active interfaces, or transparent interfaces. Finally, the class contains a range of methods for visualizing the internal state, such as e.g. plotting the density of states.

The perhaps most important class is `Superconductor`. This class inherits the internal structure of `Metal`, but modifies the Usadel equation to contain a superconducting gap $\Delta(z)$, and includes subroutines for self-consistently calculating the gap. The purpose of this class is to be used in conjunction with other `Metal`-based objects, and permits simulations of both the forward and inverse proximity effects.

Finally, the class `Ferromagnet` inherits the internal structure of `Metal`, but modifies the Usadel equation and tunneling boundary conditions to account for exchange fields and spin-orbit fields.

Together, `Metal`, `Ferromagnet`, and `Superconductor` can be used to simulate a variety of multilayer structures, either interactively or in scripts. The layers may have properties such as superconductivity, ferromagnetism, interfacial spin-filtering, interfacial spin-mixing, and spin-orbit coupling, and the combined multilayer structures can be used for calculations of e.g. density of states and critical temperature.

```
% Define a data structure to describe objects with both a 3x1 vector
% structure in geometric space and a 2x2 matrix structure in spin
% space, such as the Pauli vector, and in general SU(2) vector fields.
%
% Written by Jabir Ali Ouassou <jabirali@switzerlandmail.ch>
% Created 2015-02-14
% Updated 2015-02-16

classdef SpinVector
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    % Define the internal variables for the data structure
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    properties (GetAccess=public, SetAccess=public)
        x = zeros(2);
        y = zeros(2);
        z = zeros(2);
    end


    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    % Define the internal methods and operator overloading
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    methods
        function self = SpinVector(Ax, Ay, Az)
            % Constructs a spin vector from three spin matrices Ax, Ay, Az.
            % The arguments are multiplied by 2x2 spin matrices in case
            % scalars are provided instead of matrices.
            self.x = Ax * eye(2);
            self.y = Ay * eye(2);
            self.z = Az * eye(2);
        end


        % Overloading of display functions
        function display(self)
            name = inputname(1);
            disp(sprintf(':: %s.x:', name));
            disp([self.x]);
            disp(sprintf('\n:: %s.y:', inputname(1)));
            disp([self.y]);
            disp(sprintf('\n:: %s.z:', inputname(1)));
            disp([self.z]);
        end


        % Overloading of unary operators and functions
        function self = conj(self)
            % This overloads the complex conjugation function
            self.x = conj(self.x);
            self.y = conj(self.y);
            self.z = conj(self.z);
        end

        function self = transpose(self)
            % This overloads the matrix transposition function
            self.x = self.x.';
            self.y = self.y.';
            self.z = self.z.';
        end

        function self = ctranspose(self)
            % This overloads the complex transposition function
            self.x = self.x';
            self.y = self.y';
            self.z = self.z';
        end
```

```
function self = uplus(self)
    % This overloads the unary plus operator
end

function self = uminus(self)
    % This overloads the unary minus operator
    self.x = -self.x;
    self.y = -self.y;
    self.z = -self.z;
end


% Overloading of binary operators
function lhs = plus(lhs, rhs)
    % This overloads the plus operator for spin vectors
    lhs.x = lhs.x+rhs.x;
    lhs.y = lhs.y+rhs.y;
    lhs.z = lhs.z+rhs.z;
end

function lhs = minus(lhs, rhs)
    % This overloads the minus operator for spin vectors
    lhs.x = lhs.x-rhs.x;
    lhs.y = lhs.y-rhs.y;
    lhs.z = lhs.z-rhs.z;
end

function result = times(lhs, rhs)
    % This overloads the arraywise multiplication operator for spin vectors
    if isobject(lhs)
        if isobject(rhs)
            lhs.x = lhs.x*rhs.x;
            lhs.y = lhs.y*rhs.y;
            lhs.z = lhs.z*rhs.z;
            result = lhs;
        else
            lhs.x = lhs.x*rhs(1);
            lhs.y = lhs.y*rhs(2);
            lhs.z = lhs.z*rhs(3);
            result = lhs;
        end
    else
            rhs.x = lhs(1)*rhs.x;
            rhs.y = lhs(2)*rhs.y;
            rhs.z = lhs(3)*rhs.z;
            result = rhs;
    end
end
```

```
    function result = mtimes(lhs, rhs)
        % This overloads the matrix multiplication operator for spin vectors

        if isobject(lhs)
            % Left-hand side is a spin vector
            if isobject(rhs)
                % Right-hand side is a spin vector
                result = lhs.x*rhs.x + lhs.y*rhs.y + lhs.z*rhs.z;
            elseif length(rhs) == 3
                % Right-hand side is a vector
                result = lhs.x*rhs(1) + lhs.y*rhs(2) + lhs.z*rhs(3);
            else
                % Right-hand side is a matrix or scalar
                lhs.x = lhs.x*rhs;
                lhs.y = lhs.y*rhs;
                lhs.z = lhs.z*rhs;
                result = lhs;
            end
        else
            % Right-hand side is a spin vector, left-hand side is not
            if length(lhs) == 3
                % Left-hand side is a vector
                result = rhs.x*lhs(1) + rhs.y*lhs(2) + rhs.z*lhs(3);
            else
                % Left-hand side is a matrix or scalar
                rhs.x = rhs.x*lhs;
                rhs.y = rhs.y*lhs;
                rhs.z = rhs.z*lhs;
                result = rhs;
            end
        end
    end

    function result = mpower(lhs,rhs)
        % This overloads the matrix power operator. Note that the
        % current definition is optimized for even exponents, and
        % will not produce correct output for odd powers!

        result = (lhs.x^2 + lhs.y^2 + lhs.z^2)^(rhs/2);
    end

    function lhs = mrdivide(lhs,rhs)
        % This overloads the matrix division operator
        lhs.x = lhs.x/rhs;
        lhs.y = lhs.y/rhs;
        lhs.z = lhs.z/rhs;
    end
end
```

```
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    % Define constant properties (available without object instantiation)
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    properties (Constant)
          Pauli = SpinVector([0,1;1,0], [0,-i;i,0], [1,0;0,-1]);
    end


    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    % Define static methods (available without object instantiation)
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    methods (Static)
        function result = RashbaDresselhaus(strength, angle)
        % This function returns an SU(2) vector field that describes the
        % a Rashba--Dresselhaus coupling in the xy-plane. The coupling
        % constants are given in polar coordinates, so that the Rashba constant
        % is strength*sin(angle), and the Dresselhaus one strength*cos(angle).

        % Define the Rashba--Dresselhaus SU(2) field
        result = SpinVector( strength*(cos(angle)*SpinVector.Pauli.x   ...
                                     + sin(angle)*SpinVector.Pauli.y), ...
                            -strength*(cos(angle)*SpinVector.Pauli.y   ...
                                     + sin(angle)*SpinVector.Pauli.x), ...
                             0 );
        end
    end
end
```

```
% Define a data structure to describe the state of the physical system
% for a given position and energy. This is done by describing the
% Riccati parametrized Green's function 'g', it's tilde conjugate 'gt',
% and their first derivatives 'dg' and 'dgt' for that configuration.
%
% This class is mainly intended for use with differential equation
% solvers, and therefore provides the method 'vectorize' to pack the
% internal variables in a vector format, and constructor State(...)
% that is able to unpack this vector format. Alternatively, the
% vectorization can be performed without instantiating the class at all,
% by calling the static methods 'pack' and 'unpack'.
%
% Written by Jabir Ali Ouassou <jabirali@switzerlandmail.ch>
% Created 2015-02-14
% Updated 2015-05-06


classdef State
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    % Define the internal variables of the data structure
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    properties (GetAccess=public, SetAccess=public)
        g   = zeros(2);
        dg  = zeros(2);
        gt  = zeros(2);
        dgt = zeros(2);
    end



    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    % Define the internal methods and overloaded operators
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    methods
        function self = State(varargin)
            % This is the default constructor, which takes as its input either:
            %  (i)   four 2x2 matrices, which should correspond to the
            %        matrices g, dg, gt, dgt, in that order;
            %  (ii)  one 16-element complex vector, which should be produced
            %        by either the 'vectorize' method or the 'pack' method;
            %  (iii) no arguments, i.e. the empty constructor.
            switch nargin
                case 1
                    % If we get one input, then assume that we got a vector
                    % created by 'vectorize', and reverse the procedure

                    [self.g,self.dg,self.gt,self.dgt] = self.unpack(varargin{1});

                case 4
                    % If we get four input arguments, then assume that
                    % these correspond to g, dg, gt, and dgt, respectively.
                    % Multiply with a 2x2 identity matrix in case the
                    % input arguments were scalars and not matrices.

                    self.g   = varargin{1} * eye(2);
                    self.dg  = varargin{2} * eye(2);
                    self.gt  = varargin{3} * eye(2);
                    self.dgt = varargin{4} * eye(2);

                otherwise
                    % In any other case, we assume that we were called as
                    % an empty constructor, so do nothing to the members.
            end
        end
```

```
% Overloading of the display function
function display(self)
    name = inputname(1);
    fprintf(':: %s.g:            [  Riccati parameter g  ]\n', name);
    disp([self.g]);
    fprintf('\n:: %s.dg:           [   Derivative  dg/dz   ]\n', inputname(1));
    disp([self.dg]);
    fprintf('\n:: %s.gt:           [  Riccati parameter g~ ]\n', inputname(1));
    disp([self.gt]);
    fprintf('\n:: %s.dgt:          [   Derivative  dg~/dz  ]\n', inputname(1));
    disp([self.dgt]);
end

% Definition of other useful methods
function result = vectorize(self)
    % Convert the internal data structure to a vector shape
    result = self.pack(self.g,self.dg,self.gt,self.dgt);
end

function g = eval_g(self)
    % Return the Green's function matrix g, i.e. convert the
    % Riccati parameter self.g to a normal Green's function.
    g = ( eye(2) - self.g*self.gt ) \ ( eye(2) + self.g*self.gt );
end

function gt = eval_gt(self)
    % Return the Green's function matrix g~, i.e. convert the
    % Riccati parameter self.gt to a normal Green's function.
    gt = ( eye(2) - self.gt*self.g ) \ ( eye(2) + self.gt*self.g );
end

function f = eval_f(self)
    % Return the anomalous Green's function matrix f, i.e. convert
    % the Riccati parameter self.g to a normal Green's function.
    f = ( eye(2) - self.g*self.gt )  \ (2 * self.g);
end

function ft = eval_ft(self)
    % Return the anomalous Green's function matrix f~, i.e. convert
    % the Riccati parameter self.g to a normal Green's function.
    ft = ( eye(2) - self.gt*self.g )  \ (2 * self.gt);
end

function result = eval_ldos(self)
    result = trace(real(self.eval_g))/2;
end

function result = singlet(self)
    % Calculate the singlet component of the Green's function,
    % i.e. the component proportional to iσ^y.
    f = self.eval_f;
    result = (f(1,2) - f(2,1))/2;
end

function result = singlett(self)
    % Calculate the singlet component of the t.c. Green's function
    % i.e. the component proportional to iσ^y.
    f = self.eval_ft;
    result = (f(1,2) - f(2,1))/2;
end
```

```
    function result = triplet(self)
        % Calculate the triplet component of the Green's function,
        % i.e. the component proportional to [σ^x,σ^y,σ^z] iσ^y.
        f = self.eval_f;
        result = [(f(2,2) - f(1,1))/2,                ...
                  (f(1,1) + f(2,2))/2i,               ...
                  (f(1,2) + f(2,1))/2];
    end

    function result = triplett(self)
        % Calculate the triplet component of the t.c. Green's function,
        % i.e. the component proportional to [σ^x,σ^y,σ^z] iσ^y.
        f = self.eval_ft;
        result = [(f(2,2) - f(1,1))/2,                ...
                  (f(1,1) + f(2,2))/2i,               ...
                  (f(1,2) + f(2,1))/2];
    end

    function result = srtc(self, exchange)
        % This method returns the short-range triplet component, i.e.
        % the triplet component *along* the exchange field, where the
        % exchange field should be provided as an argument.
        unitvec = exchange/(norm(exchange)+1e-16);
        result  = dot(unitvec,self.triplet) .* unitvec;
    end

    function result = srtct(self, exchange)
        % This method returns the t.c. short-range triplet component,
        % i.e. the component *along* the exchange field, where the
        % exchange field should be provided as an argument.
        unitvec = exchange/(norm(exchange)+1e-16);
        result  = dot(unitvec,self.triplett) .* unitvec;
    end

    function result = lrtc(self, exchange)
        % This method returns the long-range triplet component, i.e.
        % the triplet component *perpendicular* to the exchange field,
        % where the exchange field should be provided as argument.
        result = self.triplet - self.srtc(exchange);
    end

    function result = lrtct(self, exchange)
        % This method returns the t.c. long-range triplet component,
        % i.e. the component *perpendicular* to the exchange field,
        % where the exchange field should be provided as argument.
        result = self.triplett - self.srtct(exchange);
    end
end
```

```
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    % Define static methods (available without object instantiation)
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    methods (Static)
        function [g,dg,gt,dgt] = unpack(vector)
            % This method is used to unpack a 16 element state vector into
            % the Riccati parametrized Green's function that it represents.
            args = reshape(vector, 2, 8);
            g    = args(:,1:2);
            dg   = args(:,3:4);
            gt   = args(:,5:6);
            dgt  = args(:,7:8);
        end

        function vector = pack(g,dg,gt,dgt)
            % This method is used to pack Riccati parametrized Green's
            % functions into a 16 element complex state vector.
            vector = reshape([g dg gt dgt], 16, 1);
        end
    end
end
```

```
% This defines a data structure that describes the physical state of a
% metal for a given range of positions and energies. The purpose of this
% class is mainly to be used as a base class for more interesting material
% classes, such as those that describe superconductors and ferromagnets,
% or to be used in conjunction with such materials in hybrid structures.
%
% Written by Jabir Ali Ouassou <jabirali@switzerlandmail.ch>
% Created 2015-02-23
% Updated 2015-05-07

classdef Metal < handle
    properties (GetAccess=public, SetAccess=public)
        %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        % Define the internal state variables of the Metal class
        %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

        % Vector of positions in the material
        positions       = [];

        % Vector of energies for the material
        energies        = [];

        % Riccati parameters and derivatives for each position and energy
        states          = State.empty(0,0);

        % Riccati parameters and derivatives at the left boundary
        boundary_left   = State.empty(0);

        % Riccati parameters and derivatives at the right boundary
        boundary_right  = State.empty(0);

        %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        % Define the basic physical parameters of the metal
        %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

        % Interface parameter zeta at the left boundary
        interface_left  = inf;

        % Interface parameter zeta at the right boundary
        interface_right = inf;

        % Thouless energy of the system
        thouless        = 1;

        % Whether we should use transparent boundary conditions
        transparent     = false;

        % Whether we should use spin-active boundary conditions
        spinactive      = false;

        %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        % Define variables related to spin-active interfaces
        %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

        % Magnetization vector at left interface  (unit vector)
        magnetization_left  = [0,0,0];

        % Magnetization vector at right interface (unit vector)
        magnetization_right = [0,0,0];

        % Polarization at left interface  [-1,+1]
        polarization_left   = 0;

        % Polarization at right interface [-1,+1]
        polarization_right  = 0;
```

```
    % Spin-dependent phase-shift factor at left interface  [-inf,inf]
    phaseshift_left     = 0;

    % Spin-dependent phase-shift factor at right interface [-inf,inf]
    phaseshift_right    = 0;

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    % Internal variables that are automatically set during simulations
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    % Coefficients in the differential equations for gamma
    coeff1  = {};

    % Coefficients in the differential equations for gamma~
    coeff2  = {};

    % Partially evaluated Jacobian functions
    jc      = {};

    % Partially evaluated boundary conditions
    bc      = {};

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    % Internal variables that determine the simulation behaviour
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    % Maximum absolute error when simulating
    error_abs = 1e-6;

    % Maximum relative error when simulating
    error_rel = 1e-6;

    % Maximum grid size to use in simulations
    grid_size = 32768;

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    % Internal variables that determine the the program behaviour
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    % Whether to show progress and intermediate results
    debug = true;

    % Whether to plot intermediate results or not
    plot  = true;

    % How long to wait between program iterations
    delay = 0;
end
```

```
methods
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    % Define methods that instantiate the object
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    function self = Metal(positions, energies, thouless)
        % This method constructs a Metal instance from a vector of
        % positions, a vector of energies, and the Thouless energy.

        % Set the internal properties to the provided values
        self.positions = positions;
        self.energies  = energies;
        self.thouless  = thouless;

        % Initialize the internal state of the metal
        self.states(length(self.positions), length(self.energies)) = State;
        self.init_superconductor;

        % Set the boundary conditions to vacuum states
        self.boundary_left(length(energies))  = State;
        self.boundary_right(length(energies)) = State;
    end

    function init_metal(self)
        % Initialize the internal state to a normal metal. This is useful
        % as an initial guess when simulating *weak* proximity effects.
        for i=1:length(self.positions)
           for j=1:length(self.energies)
               self.states(i,j) = State;
           end
        end
    end

    function init_superconductor(self)
        % Initialize the internal state to a bulk superconductor with
        % superconducting gap 1. This is useful as an initial guess
        % when simulating *strong* proximity effects in energy units
        % where the zero-temperature gap is normalized to unity.
        for i=1:length(self.positions)
           for j=1:length(self.energies)
               self.states(i,j) = Superconductor.Bulk(self.energies(j), 1, 0);
           end
        end
    end
```

```
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    % Define methods that update the internal state of the object
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    function update_coeff(self)
        % This function updates the functions and coefficients passed to
        % the functions 'jacobian' and 'boundary' when solving equations.

        % Partially evaluate the Jacobian matrix and boundary conditions
        % for the different material energies, and store the resulting
        % anonymous functions in an array. These functions are passed on
        % to bvp6c when solving the equations.
        self.jc = {};
        self.bc = {};
        for m=1:length(self.energies)
            self.jc{m} = @(x,y) self.jacobian(self,x,y,self.energies(m));
            if self.transparent
                % If 'transparent' is true, use transparent b.c.
                self.bc{m} = ...
                   @(a,b) self.boundary_transparent(self,a,b,self.energies(m));
            elseif self.spinactive
                % If 'spinactive' is true, use spin-active b.c.
                self.bc{m} = ...
                   @(a,b) self.boundary_spinactive(self,a,b,self.energies(m));
            else
                % Else, use standard Kuprianov-Lukichev b.c. instead
                self.bc{m} = ...
                   @(a,b) self.boundary(self,a,b,self.energies(m));
            end
        end
    end

    function update_boundary_left(self, other)
        % This function updates the boundary condition to the left
        % based on the current state of an adjoining material.
        self.boundary_left(:) = other.states(end,:);
    end

    function update_boundary_right(self, other)
        % This function updates the boundary condition to the right
        % based on the current state of an adjoining material.
        self.boundary_right(:) = other.states(1,:);
    end
```

```
function update_state(self)
    % This function solves the Usadel equation numerically for the
    % given position and energy range, and updates the current
    % estimate for the state of the system.

    % Set the accuracy of the numerical solution
    options = bvpset('AbsTol',self.error_abs,...
                     'RelTol',self.error_rel,...
                     'Nmax',self.grid_size);

    for m=1:length(self.energies)
        % Progress information
        self.print('[ %3.f / %3.f ]  E = %2.4f ', ...
                   m, length(self.energies), self.energies(m));

        % Vectorize the current state of the system for the given
        % energy, and use it as an initial guess for the solution
        current = zeros(16,length(self.positions));
        for n=1:length(self.positions)
            current(:,n) = self.states(n,m).vectorize;
        end
        initial = bvpinit(self.positions', current);

        % Solve the differential equation, and evaluate the
        % solution on the position vector of the metal
        solution = deval(bvp6c(self.jc{m},self.bc{m},initial,options),...
                         self.positions);

        % Update the current state of the system based on the solution
        for n=1:length(self.positions)
            self.states(n,m) = State(solution(:,n));
        end

        % Time delay between iterations (reduces load on the system)
        pause(self.delay);
    end
end


function update(self)
    % This function updates the internal state of the
    % metal object by calling the other update methods.

    % Update the state
    self.update_coeff;
    self.update_state;

    % Plot the current density of states (if 'plot' is set to true)
    if self.plot
        self.plot_dos_surf;
    end
end
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Define miscellaneous helper methods for the class
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function print(self,varargin)
    % This function is used to print messages, such as debug info
    % and progress, if the 'debug' flag is set to 'true'. The
    % message is preceded by the name of the class, which lets you
    % distinguish the output of 'Metal' instances from the output
    % of instances of daughter classes.

    if self.debug
        fprintf(':: %s: %s\n', class(self), sprintf(varargin{:}));
    end
end

function index = position_index(self, position)
    % Returns the vector index corresponding to a given energy value
    index = find(abs(self.positions-position) < 1e-8, 1, 'first');
end

function index = energy_index(self, energy)
    % Returns the vector index corresponding to a given energy value
    index = find(abs(self.energies-energy) < 1e-8, 1, 'first');
end

function backup = backup_save(self)
    % Returns a backup of the state of the metal
    backup = self.states(:,:);
end

function backup_load(self, backup)
    % Restores the state of the metal from a backup
    % NB: Remember to run 'update' after a call to 'backup_load' to
    %     make sure that the rest of the properties are consistent!
    self.states(:,:) = backup;
end
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Define methods for printing and plotting the internal state
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function plot_dos_left(self)
    % Calculate the *left* density of states for the system.
    % Note: we assume that we only have data for positive energies,
    %       and that the negative energy region is symmetric.

    N = length(self.positions);
    M = length(self.energies);
    dos = zeros(2*M-1);
    erg = zeros(2*M-1);

    for m=1:M
        erg(M-m+1) = -self.energies(m);
        dos(M-m+1) = self.states(1,m).eval_ldos;

        if ~self.debug && dos(M-m+1) < 0
            dos(M-m+1) = 1e3;
        end

        erg(M+m-1) = self.energies(m);
        dos(M+m-1) = dos(M-m+1);
    end

    % Plot the results
    plot(erg, dos);
    xlabel('\epsilon/\Delta_0')
    ylabel('D(\epsilon)')
    set(gca, 'XTick', -20:20);
    set(gca, 'YTick', -20:20);
    set(gca, 'XLim',  [-2 2]);
    set(gca, 'YLim',  [ 0 2]);
end

function plot_dos_right(self)
    % Calculate the *right* density of states for the system.
    % Note: we assume that we only have data for positive energies,
    %       and that the negative energy region is symmetric.

    N = length(self.positions);
    M = length(self.energies);
    dos = zeros(2*M-1);
    erg = zeros(2*M-1);
    for m=1:M
        erg(M-m+1) = -self.energies(m);
        dos(M-m+1) = self.states(end,m).eval_ldos;

        if ~self.debug && dos(M-m+1) < 0
            dos(M-m+1) = 1e3;
        end

        erg(M+m-1) = self.energies(m);
        dos(M+m-1) = dos(M-m+1);
    end

    % Plot the results
    plot(erg, dos);
    xlabel('\epsilon/\Delta_0')
    ylabel('D(\epsilon)')
    set(gca, 'XTick', -20:20);
    set(gca, 'YTick', -20:20);
    set(gca, 'XLim',  [-2 2]);
    set(gca, 'YLim',  [0 2]);
end
```

```
function plot_dos_center(self)
    % Calculate the *central* density of states for the system.
    % Note: we assume that we only have data for positive energies,
    %       and that the negative energy region is symmetric.

    N = length(self.positions);
    M = length(self.energies);
    dos = zeros(2*M-1);
    erg = zeros(2*M-1);
    pos = floor(length(self.positions)/2);
    for m=1:M
        erg(M-m+1) = -self.energies(m);
        dos(M-m+1) = self.states(pos,m).eval_ldos;

        if ~self.debug && dos(M-m+1) < 0
            dos(M-m+1) = 1e3;
        end

        erg(M+m-1) = self.energies(m);
        dos(M+m-1) = dos(M-m+1);
    end

    % Plot the results
    plot(erg, dos);
    xlabel('\epsilon/\Delta_0')
    ylabel('D(\epsilon)')
    set(gca, 'XTick', -20:20);
    set(gca, 'YTick', -20:20);
    set(gca, 'XLim',  [-2 2]);
    set(gca, 'YLim',  [0 2]);
end

function plot_dos_surf(self)
    % Calculate the density of states throughout the system.
    % Note: we assume that we only have data for positive energies,
    %       and that the negative energy region is symmetric.

    N = length(self.positions);
    M = length(self.energies);
    dos = zeros(N, 2*M-1);
    for n=1:N
        dos(n,M) = self.states(n,1).eval_ldos;
        for m=2:M
            dos(n,M-m+1) = self.states(n,m).eval_ldos;
            if ~self.debug && dos(n,M-m+1) < 0
                dos(n,M-m+1) = 1e3;
            end
            dos(n,M+m-1) = dos(n,M-m+1);
        end
    end

    % Plot the results
    surf([fliplr(-self.energies) self.energies(2:end)], ...
         self.positions, dos, 'EdgeColor', 'none');
    shading('interp');
    colormap(parula(256));
    caxis([0 2]);
    view(7.5,30);

    set(gca, 'XTick', -20:20);
    set(gca, 'YTick', -20:20);
    set(gca, 'ZTick', -20:20);
    set(gca, 'XLim',  [-2 2 ]);
    set(gca, 'ZLim',  [ 0 2 ]);
end
```

```
    function plot_dist(self)
        % Calculate the singlet and triplet distributions.
        %
        % NB: This implementation only *adds* the contribution from
        %     every energy, and does not perform a proper integral!

        singlet = zeros(length(self.positions), 1);
        triplet = zeros(length(self.positions), 1);
        for m=1:length(self.energies)
            for n=1:length(self.positions)
                singlet(n) = singlet(n) + norm(self.states(n,m).singlet);
                triplet(n) = triplet(n) + norm(self.states(n,m).triplet);
            end
        end

        % Plot cubic interpolations of the results
        positions = linspace(self.positions(1), self.positions(end), 100);
        plot(positions, pchip(self.positions, singlet, positions), ...
             positions, pchip(self.positions, triplet, positions));
        xlabel('Relative position');
        ylabel('Distribution');
        legend('Singlet', 'Triplet');
    end
end


methods (Static)
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    % Define static methods (available without object instantiation)
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    function dydx = jacobian(self, x, y, energy)
        % This function takes a Metal object 'self', the position 'x',
        % the current state vector 'y', and an energy as inputs, and
        % calculates the Jacobian of the system. This is performed
        % using the Riccati parametrized Usadel equations.

        % Extract the Riccati parameters and their derivatives
        [g,dg,gt,dgt] = State.unpack(y);

        % Calculate the normalization matrices
        N  = inv( eye(2) - g*gt );
        Nt = inv( eye(2) - gt*g );

        % Calculate the second derivatives of the Riccati parameters
        % according to the Usadel equation in the metal
        d2g  = -2 * dg*Nt*gt*dg - (2i/self.thouless) * (energy+1e-3i)*g;
        d2gt = -2 * dgt*N*g*dgt - (2i/self.thouless) * (energy+1e-3i)*gt;

        % Pack the results into a state vector
        dydx = State.pack(dg,d2g,dgt,d2gt);
    end
```

```
function residue = boundary(self, y1, y2, energy)
    % This function takes a Metal object 'self', the position 'x',
    % the current state vector 'y', and an energy as inputs, and
    % calculates the Kuprianov-Lukichev boundary conditions. This
    % function will be used as b.c. when 'transparent' is false.

    % State in the material to the left
    s0   = self.boundary_left(self.energy_index(energy));
    g0   = s0.g;
    dg0  = s0.dg;
    gt0  = s0.gt;
    dgt0 = s0.dgt;

    % State at the left end of the material
    [g1,dg1,gt1,dgt1] = State.unpack(y1);

    % State at the right end of the material
    [g2,dg2,gt2,dgt2] = State.unpack(y2);

    % State in the material to the right
    s3   = self.boundary_right(self.energy_index(energy));
    g3   = s3.g;
    dg3  = s3.dg;
    gt3  = s3.gt;
    dgt3 = s3.dgt;

    % Calculate the normalization matrices
    N0  = inv( eye(2) - g0*gt0 );
    Nt0 = inv( eye(2) - gt0*g0 );

    N3  = inv( eye(2) - g3*gt3 );
    Nt3 = inv( eye(2) - gt3*g3 );

    % Calculate the deviation from the Kuprianov--Lukichev b.c.
    dg1  = dg1  - (1/self.interface_left)*( eye(2) - g1*gt0 )*N0*(  g1  - g0  );
    dgt1 = dgt1 - (1/self.interface_left)*( eye(2) - gt1*g0 )*Nt0*( gt1 - gt0 );

    dg2  = dg2  - (1/self.interface_right)*( eye(2) - g2*gt3 )*N3*(  g3  - g2  );
    dgt2 = dgt2 - (1/self.interface_right)*( eye(2) - gt2*g3 )*Nt3*( gt3 - gt2 );

    % Vectorize the results of the calculations, and return it
    residue = State.pack(dg1,dgt1,dg2,dgt2);
end
```

```
function residue = boundary_spinactive(self, y1, y2, energy)
    % This function takes a Metal object 'self', the position 'x',
    % the current state vector 'y', and an energy as inputs, and
    % calculates the spin-active version of the Kuprianov-Lukichev
    % boundary conditions. This function will be used as b.c. when
    % the property 'spinactive' is set to 'true'.

    % State in the material to the left
    s0   = self.boundary_left(self.energy_index(energy));
    g0   = s0.g;
    dg0  = s0.dg;
    gt0  = s0.gt;
    dgt0 = s0.dgt;

    % State at the left end of the material
    [g1,dg1,gt1,dgt1] = State.unpack(y1);

    % State at the right end of the material
    [g2,dg2,gt2,dgt2] = State.unpack(y2);

    % State in the material to the right
    s3   = self.boundary_right(self.energy_index(energy));
    g3   = s3.g;
    dg3  = s3.dg;
    gt3  = s3.gt;
    dgt3 = s3.dgt;

    % Calculate the Green's functions
    I = eye(2);

    G0  = (I - g0*gt0) \ (I + g0*gt0);
    Gt0 = (I - gt0*g0) \ (I + gt0*g0);
    F0  = (I - g0*gt0) \ (2*g0);
    Ft0 = (I - gt0*g0) \ (2*gt0);

    G1  = (I - g1*gt1) \ (I + g1*gt1);
    Gt1 = (I - gt1*g1) \ (I + gt1*g1);
    F1  = (I - g1*gt1) \ (2*g1);
    Ft1 = (I - gt1*g1) \ (2*gt1);

    G2  = (I - g2*gt2) \ (I + g2*gt2);
    Gt2 = (I - gt2*g2) \ (I + gt2*g2);
    F2  = (I - g2*gt2) \ (2*g2);
    Ft2 = (I - gt2*g2) \ (2*gt2);

    G3  = (I - g3*gt3) \ (I + g3*gt3);
    Gt3 = (I - gt3*g3) \ (I + gt3*g3);
    F3  = (I - g3*gt3) \ (2*g3);
    Ft3 = (I - gt3*g3) \ (2*gt3);

    % Calculate the interface parameters
    LS = self.magnetization_left * SpinVector.Pauli;
    LT = self.magnetization_left * conj(SpinVector.Pauli);
    LM = self.polarization_left                ...
        /(1+sqrt(1-self.polarization_left^2));
    LK = (1-sqrt(1-self.polarization_left^2))...
        /(1+sqrt(1-self.polarization_left^2));
    LL = 1i*self.phaseshift_left;

    RS = self.magnetization_right * SpinVector.Pauli;
    RT = self.magnetization_right * conj(SpinVector.Pauli);
    RM = self.polarization_right               ...
        /(1+sqrt(1-self.polarization_right^2));
    RK = (1-sqrt(1-self.polarization_right^2))...
        /(1+sqrt(1-self.polarization_right^2));
    RL = 1i*self.phaseshift_right;
```

```
    % Calculate the left interface matrices
    L1  = (G1*G0 - F1*Ft0)*(I+LM*LS)                ...
        - (I+LM*LS)*(G0*G1 - F0*Ft1)                ...
        + (G1*LS*G0 - F1*LT*Ft0)*(LM + LK*LS)       ...
        - (LM + LK*LS)*(G0*LS*G1 - F0*LT*Ft1)       ...
        + LL*(G1*LS - LS*G1);

    Lt1 = (Gt1*Gt0 - Ft1*F0)*(I+LM*LT)              ...
        - (I+LM*LT)*(Gt0*Gt1 - Ft0*F1)              ...
        + (Gt1*LT*Gt0 - Ft1*LS*F0)*(LM + LK*LT)     ...
        - (LM + LK*LT)*(Gt0*LT*Gt1 - Ft0*LS*F1)     ...
        - LL*(Gt1*LT - LT*Gt1);

    L2  = (G1*F0 - F1*Gt0)*(I+LM*LT)                ...
        - (I+LM*LS)*(G0*F1 - F0*Gt1)                ...
        + (G1*LS*F0 - F1*LT*Gt0)*(LM+LK*LT)         ...
        - (LM+LK*LS)*(G0*LS*F1 - F0*LT*Gt1)         ...
        + LL*(F1*LT - LS*F1);

    Lt2 = (Gt1*Ft0 - Ft1*G0)*(I+LM*LS)              ...
        - (I+LM*LT)*(Gt0*Ft1 - Ft0*G1)              ...
        + (Gt1*LT*Ft0 - Ft1*LS*G0)*(LM+LK*LS)       ...
        - (LM+LK*LT)*(Gt0*LT*Ft1 - Ft0*LS*G1)       ...
        - LL*(Ft1*LS - LT*Ft1);

    % Calculate the right interface matrices
    R1  = (G2*G3 - F2*Ft3)*(I+RM*RS)                ...
        - (I+RM*RS)*(G3*G2 - F3*Ft2)                ...
        + (G2*RS*G3 - F2*RT*Ft3)*(RM + RK*RS)       ...
        - (RM + RK*RS)*(G3*RS*G2 - F3*RT*Ft2)       ...
        + RL*(G2*RS - RS*G2);

    Rt1 = (Gt2*Gt3 - Ft2*F3)*(I+RM*RT)              ...
        - (I+RM*RT)*(Gt3*Gt2 - Ft3*F2)              ...
        + (Gt2*RT*Gt3 - Ft2*RS*F3)*(RM + RK*RT)     ...
        - (RM + RK*RT)*(Gt3*RT*Gt2 - Ft3*RS*F2)     ...
        - RL*(Gt2*RT - RT*Gt2);

    R2  = (G2*F3 - F2*Gt3)*(I+RM*RT)                ...
        - (I+RM*RS)*(G3*F2 - F3*Gt2)                ...
        + (G2*RS*F3 - F2*RT*Gt3)*(RM+RK*RT)         ...
        - (RM+RK*RS)*(G3*RS*F2 - F3*RT*Gt2)         ...
        + RL*(F2*RT - RS*F2);

    Rt2 = (Gt2*Ft3 - Ft2*G3)*(I+RM*RS)              ...
        - (I+RM*RT)*(Gt3*Ft2 - Ft3*G2)              ...
        + (Gt2*RT*Ft3 - Ft2*RS*G3)*(RM+RK*RS)       ...
        - (RM+RK*RT)*(Gt3*RT*Ft2 - Ft3*RS*G2)       ...
        - RL*(Ft2*RS - RT*Ft2);

    % Calculate the deviation from the Kuprianov--Lukichev b.c.
    % with spin-active interface terms from Machon et al.
    dg1  = dg1  + (0.25/self.interface_left)*(I - g1*gt1)*(L2  - L1*g1);
    dgt1 = dgt1 + (0.25/self.interface_left)*(I - gt1*g1)*(Lt2 - Lt1*gt1);

    dg2  = dg2  - (0.25/self.interface_right)*(I - g2*gt2)*(R2 - R1*g2);
    dgt2 = dgt2 - (0.25/self.interface_right)*(I - gt2*g2)*(Rt2 - Rt1*gt2);

    % Vectorize the results of the calculations, and return it
    residue = State.pack(dg1,dgt1,dg2,dgt2);
end
```

```
        function residue = boundary_transparent(self, y1, y2, energy)
            % This function takes a Metal object 'self', the position 'x',
            % the current state vector 'y', and an energy as inputs, and
            % calculates the transparent boundary conditions. This function
            % will be used as b.c. when 'transparent' is set to true. Note
            % that if the interface parameters are infinite, we assume a
            % vacuum interface, and therefore use the derivative boundary
            % condition dg = 0 instead of the transparency condition.

            % State in the material to the left
            s0   = self.boundary_left(self.energy_index(energy));
            g0   = s0.g;
            dg0  = s0.dg;
            gt0  = s0.gt;
            dgt0 = s0.dgt;

            % State at the left end of the material
            [g1,dg1,gt1,dgt1] = State.unpack(y1);

            % State at the right end of the material
            [g2,dg2,gt2,dgt2] = State.unpack(y2);

            % State in the material to the right
            s3   = self.boundary_right(self.energy_index(energy));
            g3   = s3.g;
            dg3  = s3.dg;
            gt3  = s3.gt;
            dgt3 = s3.dgt;

            % Calculate the deviation from the boundary conditions. We use
            % the derivative boundary condition dg = 0 when there is an
            % infinite interface parameter (i.e. vacuum interface), and the
            % transparent boundary condition g1 = g2/interface_param when
            % there is a finite interface parameter.

            if isinf(self.interface_left)
                r1  = dg1;
                rt1 = dgt1;
            else
                r1  = g1  - g0/self.interface_left;
                rt1 = gt1 - gt0/self.interface_left;
            end

            if isinf(self.interface_right)
                r2  = dg2;
                rt2 = dgt2;
            else
                r2  = g2  - g3/self.interface_right;
                rt2 = gt2 - gt3/self.interface_right;
            end

            % Vectorize the results of the calculations, and return it
            residue = State.pack(r1,rt1,r2,rt2);
        end
    end
end
```

```
% This defines a data structure that describes the physical state of a
% superconducting material for a given range of positions and energies.
% This class inherits the internal structure of the 'Metal' class.
%
% Written by Jabir Ali Ouassou <jabirali@switzerlandmail.ch>
% Inspired by a similar program written by Sol Jacobsen
% Created 2015-02-15
% Updated 2015-05-05

classdef Superconductor < Metal
    properties (GetAccess=public, SetAccess=public)
        %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        % Define the internal variables of the Superconductor class
        %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

        % Temperature of the system (relative to the bulk critical temperature)
        temperature = 0;

        % Strength of the superconductor (material constant N₀λ in the gap equation)
        strength    = 0.2;

        % Superconducting gap as a function of position (relative to the bulk gap)
        gap         = griddedInterpolant([0,1],[1,1]);

        % Superconducting phase as a function of position (defaults to zero)
        phase       = griddedInterpolant([0,1],[0,0]);

        % Whether we use a gauge where the superconducting phase is zero
        complex     = false;

        % Whether the superconducting gap and phase are locked to constant values
        locked      = false;
    end


    methods
        %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        % Define methods that instantiate the object
        %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

        function self = Superconductor(positions, energies, thouless, strength)
            % This method constructs a Superconductor instance from a vector
            % of positions, a vector of energies, the Thouless energy, and
            % the strength of the superconductivity (material constant N₀λ).
            %
            % Note: for self-consistent solutions to work properly, the energy
            %       vector should extend up to the Debye cutoff cosh(1/N₀λ).

            % Initialize the Metal superclass
            self@Metal(positions, energies, thouless);

            % Set the internal variables based on constructor arguments
            self.strength = strength;
        end
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Define methods which are useful for working with superconductors
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function gap_reduce(self)
    % This function reduces the current value of the superconducting
    % gap everywhere in the material. (This can be useful for
    % accelerating the convergence of programs that search
    % for the critical temperature of a hybrid system.)

    self.gap.Values = self.gap.Values./2;
    self.update_coeff;
    self.update_state;
    self.update_gap;
end

function result = gap_left(self)
    % This function returns the left superconducting gap.

    result = abs(self.gap.Values(1));
end

function result = gap_right(self)
    % This function returns the right superconducting gap.

    result = abs(self.gap.Values(end));
end

function result = gap_mean(self)
    % This function returns the mean superconducting gap.

    result = mean(abs(self.gap.Values));
end

function result = gap_max(self)
    % This function returns the maximal superconducting gap.

    result = max(abs(self.gap.Values));
end

function result = gap_min(self)
    % This function returns the minimal superconducting gap.

    result = min(abs(self.gap.Values));
end

function phase_set(self, phase)
    % This function sets the superconducting phase at all positions,
    % and updates the internal state of the superconductor to a bulk
    % material with the correct phase.

    % Set the superconducting phase
    self.phase = griddedInterpolant([0,1],[phase,phase]);

    % Set the internal state to a bulk superconductor with a phase
    for i=1:length(self.positions)
        for j=1:length(self.energies)
            self.states(i,j) = Superconductor.Bulk(self.energies(j), 1, phase);
        end
    end
end
```

```
    function plot_gap(self)
        % Plot the superconducting gap as a function of position
        xs = linspace(self.positions(1), self.positions(end));
        plot(xs, self.gap(xs));
        xlabel('Relative position');
        ylabel('Superconducting gap');
    end

    function plot_phase(self)
        % Plot the superconducting phase as a function of position
        xs = linspace(self.positions(1), self.positions(end));
        plot(xs, self.phase(xs)/pi);

        axis([0 1 -1 1]);
        set(gca, 'XTick', [0,1/4,1/2,3/4,1]);
        set(gca, 'YTick', [-1,-1/2,0,1/2,1]);

        xlabel('Relative position');
        ylabel('Superconducting phase (\pi)');
    end


    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    % Define methods that update the internal state of the object
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    function update_gap(self)
        % This function updates the gap function, which contains the
        % current estimate of the superconducting gap in the material.

        if ~self.locked
            % Calculate the gap and phase at every position in the system
            gaps   = zeros(1,length(self.positions));
            phases = zeros(1,length(self.positions));
            for n=1:length(self.positions)
                [gaps(n),phases(n)] = self.calculate_gap(self, self.positions(n));
            end

            % Create a piecewise cubic interpolation of the results
            self.gap   = griddedInterpolant(self.positions, gaps,   'pchip');
            self.phase = griddedInterpolant(self.positions, phases, 'pchip');
        end
    end

    function update_coeff(self)
        % This function updates the vector of coefficients passed to
        % the functions 'jacobian' and 'boundary' when solving equations.

        % Call the standard 'Metal' version of the method
        update_coeff@Metal(self);

        % Coefficients in the equations for the Riccati parameter gamma
        self.coeff1{1} = -2i/self.thouless;
        self.coeff1{2} = -SpinVector.Pauli.y/self.thouless;

        % Coefficients in the equations for the Riccati parameter gamma~
        self.coeff2{1} =  self.coeff1{1};
        self.coeff2{2} = -self.coeff1{2};
    end
```

```
    function update(self)
        % This function updates the internal state of the superconductor
        % by calling the other update methods. Always run this after
        % updating the boundary conditions or physical properties of
        % the superconductor, or after restoring from a backup.

        % Update the state
        self.update_gap;
        self.update_coeff;
        self.update_state;
        self.update_gap;

        % Plot the current DOS
        if self.plot
            self.plot_dos_surf;
        end
    end
end

methods (Static)
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    % Define static methods (available without object instantiation)
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    function dydx = jacobian(self, x, y, energy)
        % This function takes a Superconductor object 'self', the
        % position 'x', the current state vector 'y', and an energy
        % as inputs, and calculates the Jacobian of the system. This
        % is performed using the Riccati parametrized Usadel equations.

        % Extract the Riccati parameters and their derivatives
        [g,dg,gt,dgt] = State.unpack(y);

        % Retrieve the superconducting gap at this point
        gap  = self.gap(x) * exp(i*self.phase(x));
        gapt = conj(gap);

        % Calculate the normalization matrices
        N  = inv( eye(2) - g*gt );
        Nt = inv( eye(2) - gt*g );

        % Calculate the second derivatives of the Riccati parameters
        % according to the Usadel equations in the superconductor
        d2g  = -2 * dg*Nt*gt*dg + self.coeff1{1} * (energy+1e-3i)*g ...
             + gap * self.coeff1{2} + gapt * g*self.coeff2{2}*g;

        d2gt = -2 * dgt*N*g*dgt + self.coeff2{1} * (energy+1e-3i)*gt...
             + gapt * self.coeff2{2} + gap * gt*self.coeff1{2}*gt;

        % Pack the results into a state vector
        dydx = State.pack(dg,d2g,dgt,d2gt);
    end

    function [gap,phase] = calculate_gap(self, position)
        % This function returns a vector [gap,phase] with the
        % superconducting gap and phase at the given position.
        % This is done by invoking either 'calculate_gap_real'
        % or 'calculate_gap_complex', depending on whether
        % the parameter 'complex' is true or false.

        if self.complex
                [gap,phase] = self.calculate_gap_complex(self, position);
        else
                [gap,phase] = self.calculate_gap_real(self, position);
        end
    end
```

```
function [result_gap,result_phase] = calculate_gap_real(self, position)
    % This function extracts the singlet components of the Green's
    % function at a given position, and then uses the gap equation
    % to calculate the superconducting gap at that point.
    % [This version works for purely real values of the gap.]

    % Extract the singlet components from the states
    singlets = zeros(size(self.energies));
    index    = self.position_index(position);
    for m=1:length(self.energies)
        singlets(m) = real(self.states(index,m).singlet);
    end

    % Create a cubic interpolation of the numerical data above
    singlet = griddedInterpolant(self.energies, singlets, 'pchip');

    % This is the expression for the gap equation integrand. Using
    % eq. (4.34) in Fossheim & Sudbø, we have rewritten the argument
    % of tanh(E/2T) such that E is measured relative to the
    % zero-temperature gap, while T is measured relative to Tc.
    kernel = @(E) singlet(E) .* tanh(0.881939 * E/self.temperature);

    % Perform a numerical integration of the interpolation up to
    % the Debye cutoff. The Debye cutoff has been calculated from
    % the superconductor strength using eq. (3.34) in Tinkham.
    % The reason for cosh(1/N₀λ) instead of sinh(1/N₀λ), is that we
    % integrate the quasiparticle energy and not the kinetic energy.
    result_gap   = self.strength                                    ...
                 * integral(kernel, self.energies(1), cosh(1/self.strength));
    result_phase = 0;
end
```

```
        function [result_gap,result_phase] = calculate_gap_complex(self, position)
            % This function uses the gap equation to calculate the super-
            % conducting gap and superconducting phase at a given position.
            % [This version works for general complex values of the gap.]

            % Calculate the singlet component of the anomalous Green's
            % functions and its tilde conjugate, and then their difference.
            index    = self.position_index(position);
            singlets = zeros(size(self.energies));
            for m=1:length(self.energies)
                singlets(m) = self.states(index,m).singlet/2     ...
                            - conj(self.states(index,m).singlett)/2;
            end

            % Create cubic interpolations of the numerical data above
            singletR = griddedInterpolant(self.energies, real(singlets), 'pchip');
            singletI = griddedInterpolant(self.energies, imag(singlets), 'pchip');

            % This is the expression for the gap equation integrand. Using
            % eq. (4.34) in Fossheim & Sudbø, we have rewritten the argument
            % of tanh(E/2T) such that E is measured relative to the
            % zero-temperature gap, while T is measured relative to Tc.
            kernelR = @(E) singletR(E) .* tanh(0.881939 * E/self.temperature);
            kernelI = @(E) singletI(E) .* tanh(0.881939 * E/self.temperature);

            % Perform a numerical integration of the interpolation up to
            % the Debye cutoff. The Debye cutoff has been calculated from
            % the superconductor strength using eq. (3.34) in Tinkham.
            % The reason for cosh(1/N₀λ) instead of sinh(1/N₀λ), is that we
            % integrate the quasiparticle energy and not the kinetic energy.
            gapR = self.strength                                          ...
                 * integral(kernelR, self.energies(1), cosh(1/self.strength));
            gapI = self.strength                                          ...
                 * integral(kernelI, self.energies(1), cosh(1/self.strength));

            % Extract the superconducting gap and phase from the results
            result       = gapR + 1i*gapI;
            result_gap   = norm(result);
            result_phase = phase(result);
        end

        function result = Bulk(energy, gap, phase)
            % This function takes as its input an energy and a superconducting
            % gap, and returns a State object with Riccati parametrized Green's
            % functions that correspond to a BCS superconductor bulk state.

            theta = atanh(gap/(energy+1e-3i));
            c     = cosh(theta);
            s     = sinh(theta);
            g     = s/(1+c);

            result = State([0,  g; -g, 0]*exp( 1i*phase), 0, ...
                           [0, -g;  g, 0]*exp(-1i*phase), 0);
        end
    end
end
```

```
% This defines a data structure that describes the physical state of a
% ferromagnetic material with spin-orbit coupling for a given range
% of positions and energies. This class inherits the internal structure
% of the 'Metal' class.
%
% Written by Jabir Ali Ouassou <jabirali@switzerlandmail.ch>
% Inspired by a similar program written by Sol Jacobsen
% Created 2015-02-16
% Updated 2015-02-26

classdef Ferromagnet < Metal
    properties (GetAccess=public, SetAccess=public)
        %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        % Define the internal variables of the Ferromagnet class
        %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

        % Exchange field in the ferromagnet
        exchange        = [0,0,0];

        % Spin-orbit field in the ferromagnet
        spinorbit       = SpinVector(0,0,0);
    end


    methods
        %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        % Define methods that instantiate the object
        %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

        function self = Ferromagnet(positions, energies, thouless, exchange, spinorbit)
            % This method constructs a Ferromagnet instance from a vector
            % of positions, a vector of energies, the Thouless energy, the
            % exchange field, and the spin-orbit field in the material.

            % Initialize the Metal superclass
            self@Metal(positions, energies, thouless);

            % Set the internal variables based on constructor arguments
            self.exchange  = exchange;
            self.spinorbit = spinorbit;
        end
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Define methods which are useful for working with ferromagnets
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function plot_dist(self)
    % Calculate the singlet and triplet distributions.
    % NB: This implementation only *adds* the contribution from
    %     every energy, and does not perform a proper integral!

    singlet = zeros(length(self.positions), 1);
    triplet = zeros(length(self.positions), 1);
    srtc    = zeros(length(self.positions), 1);
    lrtc    = zeros(length(self.positions), 1);
    for m=1:length(self.energies)
        for n=1:length(self.positions)
            singlet(n) = singlet(n) + norm(self.states(n,m).singlet);
            triplet(n) = triplet(n) + norm(self.states(n,m).triplet);
            srtc(n) = srtc(n) + norm(self.states(n,m).srtc(self.exchange));
            lrtc(n) = lrtc(n) + norm(self.states(n,m).lrtc(self.exchange));
        end
    end

    % Plot cubic interpolations of the results
    positions = linspace(0, self.positions(end), 100);
    if norm(self.exchange) == 0
        % If there is no exchange field, don't distinguish SRT/LRT
        plot(positions, pchip(self.positions, singlet, positions),  ...
             positions, pchip(self.positions, triplet, positions));
        legend('Singlet', 'Triplet');
    else
        % If there is an exchange field, do distinguish SRT/LRT
        plot(positions, pchip(self.positions, singlet, positions),  ...
             positions, pchip(self.positions, srtc,    positions),  ...
             positions, pchip(self.positions, lrtc,    positions));
        legend('Singlet', 'Short-Range Triplet', 'Long-Range Triplet');
    end
    xlabel('Relative position');
    ylabel('Distribution');
end
```

```
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    % Define methods that update the internal state of the object
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    function update_coeff(self)
        % This function updates the vector of coefficients passed to
        % the functions 'jacobian' and 'boundary' when solving equations.

        % Call the standard 'Metal' version of the method
        update_coeff@Metal(self);

        % Coefficients in the equations for the Riccati parameter gamma
        self.coeff1{1} = -2i/self.thouless;
        self.coeff1{2} = (-i*self.exchange/self.thouless)*SpinVector.Pauli;
        self.coeff1{3} = self.spinorbit^2/self.thouless;
        self.coeff1{4} = sqrt(2/self.thouless)    * self.spinorbit.x;
        self.coeff1{5} = sqrt(2/self.thouless)    * self.spinorbit.y;
        self.coeff1{6} = sqrt(2/self.thouless)    * self.spinorbit.z;
        self.coeff1{7} = (2i/sqrt(self.thouless)) * self.spinorbit.z;

        % Coefficients in the equations for the Riccati parameter gamma~
        self.coeff2{1} = self.coeff1{1};
        self.coeff2{2} = conj(self.coeff1{2});
        self.coeff2{3} = conj(self.coeff1{3});
        self.coeff2{4} = conj(self.coeff1{4});
        self.coeff2{5} = conj(self.coeff1{5});
        self.coeff2{6} = conj(self.coeff1{6});
        self.coeff2{7} = conj(self.coeff1{7});
    end
end
```

```
    methods (Static)
        %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        % Define static methods (available without object instantiation)
        %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

        function dydx = jacobian(self, x, y, energy)
            % This function takes a Ferromagnet object 'self', the
            % position 'x', the current state vector 'y', and an energy
            % as inputs, and calculates the Jacobian of the system. This
            % is performed using the Riccati parametrized Usadel equations.

            % Extract the Riccati parameters and their derivatives
            [g,dg,gt,dgt] = State.unpack(y);

            % Calculate the normalization matrices
            N  = inv( eye(2) - g*gt );
            Nt = inv( eye(2) - gt*g );

            % Calculate the second derivatives of the Riccati parameters
            % according to the Usadel equations in the ferromagnet
            d2g  = -2 * dg*Nt*gt*dg + self.coeff1{1} * (energy+1e-3i)*g  ...
                 + (self.coeff1{2} + self.coeff1{3}) * g                 ...
                 + g * (self.coeff2{2} - self.coeff2{3})                 ...
                 + (self.coeff1{4} * g + g * self.coeff2{4})             ...
                   * Nt * (self.coeff2{4} + gt * self.coeff1{4} * g)     ...
                 + (self.coeff1{5} * g + g * self.coeff2{5})             ...
                   * Nt * (self.coeff2{5} + gt * self.coeff1{5} * g)     ...
                 + (self.coeff1{6} * g + g * self.coeff2{6})             ...
                   * Nt * (self.coeff2{6} + gt * self.coeff1{6} * g)     ...
                 + (self.coeff1{7} - g * self.coeff2{7} * gt) * N * dg   ...
                 + dg * Nt * (gt * self.coeff1{7} * g - self.coeff2{7});

            d2gt = -2 * dgt*N*g*dgt + self.coeff2{1} * (energy+1e-3i)*gt ...
                 + (self.coeff2{2} + self.coeff2{3}) * gt                ...
                 + gt * (self.coeff1{2} - self.coeff1{3})                ...
                 + (self.coeff2{4} * gt + gt * self.coeff1{4})           ...
                   * N * (self.coeff1{4} + g * self.coeff2{4} * gt)      ...
                 + (self.coeff2{5} * gt + gt * self.coeff1{5})           ...
                   * N * (self.coeff1{5} + g * self.coeff2{5} * gt)      ...
                 + (self.coeff2{6} * gt + gt * self.coeff1{6})           ...
                   * N * (self.coeff1{6} + g * self.coeff2{6} * gt)      ...
                 + (self.coeff2{7} - gt * self.coeff1{7} * g) * Nt * dgt ...
                 + dgt * N * (g * self.coeff2{7} * gt - self.coeff1{7});

            % Pack the results into a state vector
            dydx = State.pack(dg,d2g,dgt,d2gt);
        end
```

```
        function residue = boundary(self, y1, y2, energy)
            % This function takes a Metal object 'self', the position 'x',
            % the current state vector 'y', and an energy as inputs, and
            % calculates the Kuprianov-Lukichev boundary conditions.
            %
            % Note: this is an extension of 'Metal.boundary' which also
            %       works in the presence of spin-orbit coupling.

            % State in the material to the left
            s0   = self.boundary_left(self.energy_index(energy));
            g0   = s0.g;
            dg0  = s0.dg;
            gt0  = s0.gt;
            dgt0 = s0.dgt;

            % State at the left end of the material
            [g1,dg1,gt1,dgt1] = State.unpack(y1);

            % State at the right end of the material
            [g2,dg2,gt2,dgt2] = State.unpack(y2);

            % State in the material to the right
            s3   = self.boundary_right(self.energy_index(energy));
            g3   = s3.g;
            dg3  = s3.dg;
            gt3  = s3.gt;
            dgt3 = s3.dgt;

            % Calculate the normalization matrices
            N0  = inv( eye(2) - g0*gt0 );
            Nt0 = inv( eye(2) - gt0*g0 );

            N3  = inv( eye(2) - g3*gt3 );
            Nt3 = inv( eye(2) - gt3*g3 );

            % Calculate the deviation from the Kuprianov--Lukichev b.c.
            dg1  = dg1  - (1/self.interface_left)*( eye(2) - g1*gt0 )  ...
                          * N0  * ( g1  - g0  )                        ...
                        - (self.coeff1{7} * g1  - g1  * self.coeff2{7})/2;
            dgt1 = dgt1 - (1/self.interface_left)*( eye(2) - gt1*g0 )  ...
                          * Nt0 * ( gt1 - gt0 )                        ...
                        - (self.coeff2{7} * gt1 - gt1 * self.coeff1{7})/2;

            dg2  = dg2  - (1/self.interface_right)*( eye(2) - g2*gt3 ) ...
                          * N3  * (  g3  - g2  )                       ...
                        - (self.coeff1{7} * g2  - g2  * self.coeff2{7})/2;
            dgt2 = dgt2 - (1/self.interface_right)*( eye(2) - gt2*g3 ) ...
                          * Nt3 * ( gt3 - gt2 )                        ...
                        - (self.coeff2{7} * gt2 - gt2 * self.coeff1{7})/2;

            % Vectorize the results of the calculations, and return it
            residue = State.pack(dg1,dgt1,dg2,dgt2);
        end
    end
end
```

MATLAB SCRIPTS

The following pages contains a number of MATLAB scripts, which builds upon the classes that were introduced on the preceding pages. The scripts can be classified into three groups:

(i) `critical_*.m` perform self-consistent calculations, and use them to determine the critical temperature of a system;

(ii) `simulate_bilayer_*.m` calculates the density of states in bilayers;

(iii) `simulate_josephson_*.m` calculates the density of states in trilayers.

These scripts were used to create the vast majority of data presented in this thesis. However, a number of plots—notably the contour plots—were generated by interactive use of the `Metal/Superconductor/Ferromagnet` classes, and the corresponding code has therefore not been included here.

```
% This script calculates the critical temperature of a bulk superconductor,
% by performing a binary search for the temperature where the gap vanishes
% numerically. The result should be numerically one in the given unit
% system, so this script can be used to calibrate other simulation parameters.
%
% Written by Jabir Ali Ouassou <jabirali@switzerlandmail.ch>
% Created 2015-02-28
% Updated 2015-03-04


function critical_superconductor(superconductor_length, superconductor_strength)
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %                   DEFINE PARAMETERS FOR THE SIMULATION
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    % Vectors of positions and energies that will be used in the simulation
    positions = linspace(0, 1, 10);
    energies  = [linspace(0.000,1.500,500) linspace(1.501,cosh(1/superconductor_strength),100)];

    % Number of iterations of the binary search to perform
    iterations    = 8;

    % Number of iterations that the system needs to stabilize
    stabilization = 5;

    % Upper and lower limits for the binary search
    lower = 0.00;
    upper = 1.50;

    % Filename where results will be stored
    output = 'critical_superconductor.dat';



    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %                     PREPARATIONS FOR THE SIMULATION
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    % Make sure that all required classes and methods are in the current path
    initialize;

    % Create a superconductor object based on the parameters above
    s = Superconductor(positions, energies, 1/superconductor_length^2, superconductor_strength);

    % Enable or disable various debugging flags
    s.delay = 0;
    s.debug = 1;
    s.plot  = 0;

    % This variable is used to keep a backup of the last non-critical object
    sb = s.backup_save;
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%         PERFORM A BINARY SEARCH FOR THE CRITICAL TEMPERATURE
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

tic;
for n=1:iterations
    % Set the current temperature to the average of the two previous values
    s.temperature = (upper+lower)/2;

    % Keep updating the internal state of the superconductor until we
    % either reach a phase transition, or the gap starts to increase
    loop = 0;
    gaps = [ 1 ];
    while true
        % If too many iterations have passed without convergence, then
        % proceed to accelerate the convergence by reducing the gap
        loop = loop + 1;
        if rem(loop,stabilization) == 0
            % Status information
            fprintf(':: [ %3d / %3d ] [ Temp: %.6f ] [ Time: %2d min ] Accelerating!\n',...
                     n, iterations, s.temperature, floor(toc/60));

            % Reduce the superconducting gap everywhere in the system
            s.gap_reduce;
        end

        % Status information
        fprintf(':: [ %3d / %3d ] [ Temp: %.6f ] [ Time: %2d min ] [ Gap: %.6f ]\n',...
                 n, iterations, s.temperature, floor(toc/60), s.gap_max);

        % Update the superconductor state
        s.update;
        gaps(end+1) = s.gap_mean;

        if (gaps(end)-gaps(end-1)) > 0
            % The gap increased during the last iteration, so we must be
            % below the critical temperature. Updating the lower estimate,
            % and using the current state as an initial guess in the next
            % simulation. Then terminate the loop.

            lower = s.temperature;
            sb    = s.backup_save;
            break;

        elseif (s.gap_max < 0.005)
            % The gap is so small that we must have reached critical
            % temperature. Update upper estimate, load a noncritical
            % state from backup, and terminate loop.

            upper = s.temperature;
            s.backup_load(sb);
            break;

        else
            % If the superconductor is not critical, then use the current
            % state as an initial guess in the next simulation.

            sb = s.backup_save;
        end
    end
end
```

```
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %           SAVE RESULTS AND CLEAN UP AFTER THE SIMULATION
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    % The final estimate of the critical temperature is the mean of the current
    % upper and lower limits obtained by the above calculations
    critical_temperature = (upper+lower)/2;

    % Output the final result
    fprintf('Critical temperature: %.6f\nLower limit: %.6f\nUpper limit: %.6f\n:',...
             critical_temperature, lower, upper);

    % Save the results to file
    save(output);
end
```

```
% This script calculates the critical temperature of a superconductor/ferromagnet
% bilayer with spin-orbit coupling, by performing a kind of binary search for
% the temperature where the superconducting gap vanishes numerically.
%
% Written by Jabir Ali Ouassou <jabirali@switzerlandmail.ch>
% Created 2015-03-01
% Updated 2015-03-04



function critical_bilayer(superconductor_length, ferromagnet_length, ...
                          strength, exchange, spinorbit, angle)

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %                   DEFINE PARAMETERS FOR THE SIMULATION
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    % Vectors of positions and energies that will be used in the simulation
    positions     = linspace(0, 1, 150);
    energies      = [linspace(0.000,1.500,500) linspace(1.501,cosh(1/strength),100)];

    % Number of iterations of the binary search to perform
    iterations    = 12;

    % Number of iterations that the system needs to stabilize
    stabilization = 6;

    % Upper and lower limits for the binary search
    lower = 0;
    upper = 1;

    % Filename where results will be stored
    output = 'critical_bilayer.dat';


    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %                     PREPARATIONS FOR THE SIMULATION
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    % Make sure that all required classes and methods are in the current path
    initialize;

    % Instantiate and initialize superconductor/ferromagnet objects
    s = Superconductor(positions, energies, 1/superconductor_length^2, strength);
    f = Ferromagnet(positions, energies, 1/ferromagnet_length^2, exchange, ...
                    SpinVector.RashbaDresselhaus(spinorbit, angle));

    s.temperature     = 0;
    s.interface_right = 3;
    f.interface_left  = 3;

    % This enables or disables various debugging options
    s.delay = 0;
    s.debug = 1;
    s.plot  = 0;
    f.delay = 0;
    f.debug = 1;
    f.plot  = 0;
```

```
% Print out parameters for verification purposes
fprintf('SUPERCONDUCTOR:\n');
fprintf(' :: Length:          %1.6f\n',   superconductor_length);
fprintf(' :: Thouless energy: %1.6f\n',   s.thouless);
fprintf(' :: Interface:       %1.6f\n',   s.interface_right);
fprintf(' :: Strength:        %1.6f\n\n', s.strength);
fprintf('FERROMAGNET:\n');
fprintf(' :: Length:          %1.6f\n',   ferromagnet_length);
fprintf(' :: Thouless energy: %1.6f\n',   f.thouless);
fprintf(' :: Interface:       %1.6f\n',   f.interface_left);
fprintf(' :: Exchange field h:\n');
disp(f.exchange);
fprintf(' :: Spin-orbit field Ax:\n')
disp(f.spinorbit.x);
fprintf(' :: Spin-orbit field Ay:\n')
disp(f.spinorbit.y);
fprintf(' :: Spin-orbit field Az:\n')
disp(f.spinorbit.z);


% Initialize the bilayer by performing 'stabilization' iterations at
% zero temperature, to make sure that we do get the proximity effect
tic;
for n=1:stabilization
    % Status information
    fprintf('[ %3d / %3d ] [ Temp: %.6f ] [ Time: %2d min ] Initializing...\n', ...
            n, stabilization, s.temperature, floor(toc/60));

    % Update the boundary condition and state of the ferromagnet
    f.update_boundary_left(s);
    f.update;

    % Update the boundary condition and state of the superconductor
    s.update_boundary_right(f);
    s.update;
end

% These variables are used to keep a backup of the last non-critical object
sb = s.backup_save;
fb = f.backup_save;


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%           PERFORM A BINARY SEARCH FOR THE CRITICAL TEMPERATURE
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

for n=1:iterations
    % Set the current temperature to the average of the two previous values
    s.temperature = (upper+lower)/2;

    % Keep updating the internal state of the superconductor until we
    % either reach a phase transition, or the gap starts to increase
    loop = 0;
    gaps = [ 1 ];
    while true
        % If too many iterations have passed without convergence, then
        % proceed to accelerate the convergence by reducing the gap
        loop = loop + 1;
        if rem(loop,stabilization) == 0
            % Status information
            fprintf('[ %3d / %3d ] [ Temp: %.6f ] [ Time: %2d min ] Accelerating...\n',...
                    n, iterations, s.temperature, floor(toc/60));

            % Reduce the superconducting gap everywhere in the system
            s.gap_reduce;
        end
```

```
            % Status information
            fprintf('[ %3d / %3d ] [ Temp: %.6f ] [ Time: %2d min ] [ Gap: %.6f ]\n',...
                    n, iterations, s.temperature, floor(toc/60), s.gap_max);

            % Update the ferromagnet boundary conditions and state
            f.update_boundary_left(s);
            f.update;

            % Update the superconductor boundary conditions and state
            s.update_boundary_right(f);
            s.update;

            % Store the current superconductor mean gap in 'gaps'
            gaps(end+1) = s.gap_mean;

            % This is the logic that controls the while loop
            if (gaps(end)-gaps(end-1)) > 0
                % The gap increased during the last iteration, so we must be
                % below the critical temperature. Updating the lower estimate,
                % and using the current state as an initial guess in the next
                % simulation. Then terminate the loop.

                lower = s.temperature;
                sb    = s.backup_save;
                fb    = f.backup_save;
                break;

            elseif (s.gap_max < 0.005)
                % The gap is so small that we must have reached critical
                % temperature. Update upper estimate, load a noncritical
                % state from backup, and terminate the loop.

                upper = s.temperature;
                s.backup_load(sb);
                f.backup_load(fb);
                break;

            else
                % If the superconductor is not critical, then use the current
                % state as an initial guess in future simulations.

                sb = s.backup_save;
                fb = f.backup_save;
            end
        end
    end


    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %            SAVE RESULTS AND CLEAN UP AFTER THE SIMULATION
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    % The final estimate of the critical temperature is the mean of the current
    % upper and lower limits obtained by the above calculations
    critical_temperature = (upper+lower)/2;

    % Output the final result
    fprintf('Critical temperature: %.6f\nLower limit: %.6f\nUpper limit: %.6f\n:', ...
            critical_temperature, lower, upper);

    % Save the results to file
    save(output);
end
```

```
% This script simulates the proximity effect in a normal metal.
%
% Written by Jabir Ali Ouassou <jabirali@switzerlandmail.ch>
% Created 2015-05-06
% Updated 2015-05-07

function simulate_bilayer()
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %                   DEFINE PARAMETERS FOR THE SIMULATION
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    % Vectors of positions and energies that will be used in the simulation
    positions     = linspace(0.0, 1.0, 50);
    energies      = linspace(0.0, 2.0, 50);

    % Filename where results will be stored
    output = 'simulate_bilayer.dat';

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %                     PREPARATIONS FOR THE SIMULATION
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    % Make sure that all required classes and methods are in the current path
    initialize;

    % Create a superconductor
    s = Superconductor([0], energies, 1, 0.2);

    % Create a normal metal connected to the superconductor above
    m = Metal(positions, energies, 1);
    m.interface_left  = 3;
    m.update_boundary_left(s);

    % This enables or disables various debugging options
    m.delay = 0;
    m.debug = 1;
    m.plot  = 0;

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %                         PERFORM THE SIMULATION
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    % Update the internal state of the metal
    m.update;

    % Plot the results
    figure;
    m.plot_dos_surf;
    figure;
    m.plot_dos_center;

    % Save the results of the simulation
    save(output);
end
```

```
% This script simulates the proximity effect in a metal with a spin-active interface.
%
% Written by Jabir Ali Ouassou <jabirali@switzerlandmail.ch>
% Created 2015-05-06
% Updated 2015-05-07

function simulate_bilayer_spinactive(interface_polarization, interface_phase)
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %                   DEFINE PARAMETERS FOR THE SIMULATION
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    % Vectors of positions and energies that will be used in the simulation
    positions     = linspace(0.0, 1.0, 100);
    energies      = linspace(0.0, 2.0,  50);

    % Filename where results will be stored
    output = 'simulate_bilayer_spinactive.dat';

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %                     PREPARATIONS FOR THE SIMULATION
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    % Make sure that all required classes and methods are in the current path
    initialize;

    % Create a superconductor
    s = Superconductor([0], energies, 1, 0.2);

    % Create a normal metal connected to the superconductors above
    m = Metal(positions, energies, 1);
    m.spinactive         = 1;
    m.interface_left     = 3;
    m.magnetization_left = [0,0,1];
    m.polarization_left  = interface_polarization;
    m.phaseshift_left    = interface_phase;
    m.update_boundary_left(s);

    % This enables or disables various debugging options
    m.delay = 0;
    m.debug = 1;
    m.plot  = 0;

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %                         PERFORM THE SIMULATION
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    % Update the internal state of the metal
    m.update;

    % Plot the results
    figure;
    m.plot_dos_surf;
    figure;
    m.plot_dos_center;

    % Save the results of the simulation
    save(output);
end
```

```
% This script simulates the proximity effect in a ferromagnet with spin-orbit coupling.
%
% Written by Jabir Ali Ouassou <jabirali@switzerlandmail.ch>
% Created 2015-02-06
% Updated 2015-05-07

function simulate_bilayer_spinorbit(exchange_strength, exchange_angle,...
                                    spinorbit_strength, spinorbit_angle)

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %                   DEFINE PARAMETERS FOR THE SIMULATION
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    % Vectors of positions and energies that will be used in the simulation
    positions     = linspace(0.0, 1.0, 100);
    energies      = linspace(0.0, 2.0,  50);

    % Filename where results will be stored
    output = 'simulate_bilayer_spinorbit.dat';

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %                    PREPARATIONS FOR THE SIMULATION
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    % Make sure that all required classes and methods are in the current path
    initialize;

    % Create a superconductor
    s = Superconductor([0], energies, 1, 0.2);

    % Create a ferromagnet connected to the superconductor above
    m = Ferromagnet(positions, energies, 1/0.5^2,                                   ...
                    exchange_strength*[cos(exchange_angle), sin(exchange_angle), 0],...
                    SpinVector.RashbaDresselhaus(spinorbit_strength, spinorbit_angle));
    m.interface_left = 3;
    m.update_boundary_left(s);

    % This enables or disables various debugging options
    m.delay = 0;
    m.debug = 1;
    m.plot  = 0;

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %                          PERFORM THE SIMULATION
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    % Update the internal state of the metal
    m.update;

    % Plot the results
    figure;
    m.plot_dos_surf;
    figure;
    m.plot_dos_center;

    % Save the results of the simulation
    save(output);
end
```

```
% This script simulates the proximity effect in a metal with both a spin-active interface
% to a bulk superconductor, and a Rashba-Dresselhaus spin-orbit coupling in the metal.
%
% Written by Jabir Ali Ouassou <jabirali@switzerlandmail.ch>
% Created 2015-05-20
% Updated 2015-05-20

function simulate_bilayer_spinorbitactive(interface_polarization,          ...
                                          interface_phase, interface_angle,...
                                          spinorbit_strength, spinorbit_angle)

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %                  DEFINE PARAMETERS FOR THE SIMULATION
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    % Vectors of positions and energies that will be used in the simulation
    positions     = linspace(0.0, 1.0, 150);
    energies      = linspace(0.0, 2.0,  50);

    % Filename where results will be stored
    output = 'simulate_bilayer_spinorbitactive.dat';

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %                    PREPARATIONS FOR THE SIMULATION
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    % Make sure that all required classes and methods are in the current path
    initialize;

    % Create a superconductor
    s = Superconductor([0], energies, 1, 0.2);

    % Create a normal metal connected to the superconductors above
    m = Ferromagnet(positions, energies, 1/0.5^2, [0,0,0], ...
                    SpinVector.RashbaDresselhaus(spinorbit_strength, spinorbit_angle));
    m.spinactive         = 1;
    m.interface_left     = 5;
    m.magnetization_left = [cos(interface_angle),sin(interface_angle),0];
    m.polarization_left  = interface_polarization;
    m.phaseshift_left    = interface_phase;
    m.update_boundary_left(s);
    m.init_metal;

    % This enables or disables various debugging options
    m.delay = 0;
    m.debug = 1;
    m.plot  = 0;

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %                          PERFORM THE SIMULATION
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    % Update the internal state of the metal
    m.update;

    % Plot the results
    figure;
    m.plot_dos_surf;
    figure;
    m.plot_dos_center;

    % Save the results of the simulation
    save(output);
end
```

```
% This script simulates the proximity effect in a normal metal connected
% to two superconductors with a constant phase difference between them.
%
% Written by Jabir Ali Ouassou <jabirali@switzerlandmail.ch>
% Created 2015-05-06
% Updated 2015-05-07

function simulate_josephson(phase_difference)
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %                   DEFINE PARAMETERS FOR THE SIMULATION
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    % Vectors of positions and energies that will be used in the simulation
    positions     = linspace(0.0, 1.0, 50);
    energies      = linspace(0.0, 2.0, 50);

    % Filename where results will be stored
    output = 'simulate_josephson.dat';

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %                     PREPARATIONS FOR THE SIMULATION
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    % Make sure that all required classes and methods are in the current path
    initialize;

    % Create two superconductors with 'phase_difference' between them
    s1 = Superconductor([0], energies, 1, 0.2);
    s1.complex = true;
    s1.phase_set(-phase_difference/2);

    s2 = Superconductor([0], energies, 1, 0.2);
    s2.complex = true;
    s2.phase_set(+phase_difference/2);

    % Create a normal metal connected to the superconductors above
    m = Metal(positions, energies, 1);
    m.interface_left  = 3;
    m.interface_right = 3;
    m.update_boundary_left(s1);
    m.update_boundary_right(s2);

    % This enables or disables various debugging options
    m.delay = 0;
    m.debug = 1;
    m.plot  = 0;

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %                         PERFORM THE SIMULATION
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    % Update the internal state of the metal
    m.update;

    % Plot the results
    figure;
    m.plot_dos_surf;
    figure;
    m.plot_dos_center;

    % Save the results of the simulation
    save(output);
end
```

```
% This script simulates the proximity effect in a normal metal connected
% to two superconductors with a constant phase difference between them,
% where both interfaces are assumed to be spin-active.
%
% Written by Jabir Ali Ouassou <jabirali@switzerlandmail.ch>
% Created 2015-05-06
% Updated 2015-05-07

function simulate_josephson_spinactive(phase_difference, interface_polarization, ...
                                       interface_phase, interface_angle)

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %                  DEFINE PARAMETERS FOR THE SIMULATION
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    % Vectors of positions and energies that will be used in the simulation
    positions     = linspace(0.0, 1.0, 100);
    energies      = linspace(0.0, 2.0,  50);

    % Filename where results will be stored
    output = 'simulate_josephson_spinactive.dat';


    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %                    PREPARATIONS FOR THE SIMULATION
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    % Make sure that all required classes and methods are in the current path
    initialize;

    % Create two superconductors with 'phase_difference' between them
    s1 = Superconductor([0], energies, 1, 0.2);
    s1.complex = true;
    s1.phase_set(-phase_difference/2);

    s2 = Superconductor([0], energies, 1, 0.2);
    s2.complex = true;
    s2.phase_set(+phase_difference/2);

    % Create a normal metal connected to the superconductors above
    m = Metal(positions, energies, 1);
    m.spinactive          = 1;
    m.interface_left      = 3;
    m.interface_right     = 3;
    m.magnetization_left  = [cos(+interface_angle/2), sin(+interface_angle/2), 0];
    m.magnetization_right = [cos(-interface_angle/2), sin(-interface_angle/2), 0];
    m.polarization_left   = interface_polarization;
    m.polarization_right  = interface_polarization;
    m.phaseshift_left     = interface_phase;
    m.phaseshift_right    = interface_phase;
    m.update_boundary_left(s1);
    m.update_boundary_right(s2);

    % This enables or disables various debugging options
    m.delay = 0;
    m.debug = 1;
    m.plot  = 0;
```

```
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %                          PERFORM THE SIMULATION
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    % Update the internal state of the metal
    m.update;

    % Plot the results
    figure;
    m.plot_dos_surf;
    figure;
    m.plot_dos_center;

    % Save the results of the simulation
    save(output);
end
```

```
% This script simulates the proximity effect in a ferromagnet connected
% to two superconductors with a constant phase difference between them.
%
% Written by Jabir Ali Ouassou <jabirali@switzerlandmail.ch>
% Created 2015-05-06
% Updated 2015-05-07

function simulate_josephson_spinorbit(phase_difference,                 ...
                                      exchange_strength, exchange_angle,...
                                      spinorbit_strength, spinorbit_angle)

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %                  DEFINE PARAMETERS FOR THE SIMULATION
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    % Vectors of positions and energies that will be used in the simulation
    positions     = linspace(0.0, 1.0, 100);
    energies      = linspace(0.0, 2.0,  50);

    % Filename where results will be stored
    output = 'simulate_josephson_spinorbit.dat';


    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %                    PREPARATIONS FOR THE SIMULATION
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    % Make sure that all required classes and methods are in the current path
    initialize;

    % Create two superconductors with 'phase_difference' between them
    s1 = Superconductor([0], energies, 1, 0.2);
    s1.complex = true;
    s1.phase_set(-phase_difference/2);

    s2 = Superconductor([0], energies, 1, 0.2);
    s2.complex = true;
    s2.phase_set(+phase_difference/2);

    % Create a normal metal connected to the superconductors above
    m = Ferromagnet(positions, energies, 1/0.5^2,                                   ...
                    exchange_strength*[cos(exchange_angle), sin(exchange_angle), 0],...
                    SpinVector.RashbaDresselhaus(spinorbit_strength, spinorbit_angle));
    m.interface_left  = 3;
    m.interface_right = 3;
    m.update_boundary_left(s1);
    m.update_boundary_right(s2);

    % This enables or disables various debugging options
    m.delay = 0;
    m.debug = 1;
    m.plot  = 0;
```

```
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %                          PERFORM THE SIMULATION
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    % Update the internal state of the metal
    m.update;

    % Plot the results
    figure;
    m.plot_dos_surf;
    figure;
    m.plot_dos_center;

    % Save the results of the simulation
    save(output);
end
```

```
% This script simulates the proximity effect in a normal metal connected
% to two superconductors with a constant phase difference between them.
% It is assumed that the interfaces are spin-active, and that the normal
% metal has a Rashba-Dresselhaus coupling in the xy-plane.
%
% Written by Jabir Ali Ouassou <jabirali@switzerlandmail.ch>
% Created 2015-05-22
% Updated 2015-05-22

function simulate_josephson_spinorbitactive(phase_difference,                          ...
                                            interface_polarization, interface_phase,   ...
                                            interface_angle_left, interface_angle_right,...
                                            spinorbit_strength, spinorbit_angle)

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %                    DEFINE PARAMETERS FOR THE SIMULATION
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    % Vectors of positions and energies that will be used in the simulation
    positions     = linspace(0.0, 1.0, 100);
    energies      = linspace(0.0, 2.0,  50);

    % Filename where results will be stored
    output = 'simulate_josephson_spinorbitactive.dat';



    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %                     PREPARATIONS FOR THE SIMULATION
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    % Make sure that all required classes and methods are in the current path
    initialize;

    % Create two superconductors with 'phase_difference' between them
    s1 = Superconductor([0], energies, 1, 0.2);
    s1.complex = true;
    s1.phase_set(-phase_difference/2);

    s2 = Superconductor([0], energies, 1, 0.2);
    s2.complex = true;
    s2.phase_set(+phase_difference/2);

    % Create a normal metal connected to the superconductors above
    m = Ferromagnet(positions, energies, 1, [0,0,0], ...
                    SpinVector.RashbaDresselhaus(spinorbit_strength, spinorbit_angle));
    m.spinactive          = 1;
    m.interface_left      = 5;
    m.interface_right     = 5;
    m.magnetization_left  = [cos(interface_angle_left),  sin(interface_angle_left),  0];
    m.magnetization_right = [cos(interface_angle_right), sin(interface_angle_right), 0];
    m.polarization_left   = interface_polarization;
    m.polarization_right  = interface_polarization;
    m.phaseshift_left     = interface_phase;
    m.phaseshift_right    = interface_phase;
    m.update_boundary_left(s1);
    m.update_boundary_right(s2);

    % This enables or disables various debugging options
    m.delay = 0;
    m.debug = 1;
    m.plot  = 0;
```

```
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %                          PERFORM THE SIMULATION
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    % Update the internal state of the metal
    m.update;

    % Plot the results
    figure;
    m.plot_dos_surf;
    figure;
    m.plot_dos_center;

    % Save the results of the simulation
    save(output);
end
```