

Dissertation

submitted to the

Combined Faculties of Natural Sciences and Mathematics
of the Ruperto-Carola-University of Heidelberg. Germany

for the degree of

Doctor of Natural Sciences

Put forward by

Conrad Albrecht

born in Pirna, Germany

Oral examination: January 29, 2014

On a Numerical Framework
for Functional Renormalization
of Quantum Statistical Physics

*Referees: Prof. Dr. Christof Wetterich
Prof. Dr. Franz Wegner*

Zusammenfassung

Numerische Aspekte Funktionaler Renormierung für die quanten-statistische Physik. Die vorliegende Arbeit möchte einen Beitrag auf dem Gebiet der Funktionalen Renormierung mit Schwerpunkt auf der Statistischen Quantenphysik leisten. Speziellen Wert legen wir dabei auf ein flexibles Trunkierungsschema, welches zwei zentrale mathematische Objekte von physikalischer Relevanz beschreibt: Den inversen Propagator und das effektive Potential des Systems. Für erst genannten streben wir insbesondere eine genaue Impulsaufflösung an. Sie steht im Zusammenhang mit der Dispersion der Teilchen des Systems. Das effektive Potential hingegen enthält u.a. bedeutsame Informationen in Bezug auf die thermodynamische Zustandsgleichung und das Phasendiagramm des Systems.

Eine wesentliche Errungenschaft unserer Studie ist die Umsetzung einer numerischen Bibliothek, `libfrg`, die einen allgemeinen Rahmen für paralleles Hochleistungsrechnen in Verbindung mit der Funktionalen Renormierung ermöglicht. Sie steht unter der GNU GPL und ist somit geeignet, zukünftig von der Gemeinschaft der Forscher, die auf diesem Gebiet arbeiten, weiterentwickelt zu werden.

Abstract

On a Numerical Framework for Functional Renormalization of Quantum Statistical Physics. The subject of this thesis intends to investigate and put forward the method of functional renormalization within the field of quantum statistical physics. Our focus is on a (generic) truncation scheme that is suited to flexibly resolve two important mathematical objects of physical relevance: The (inverse) propagator and the effective potential, respectively. In the former case our effort aims at a proper resolution of the momentum dependence which is related to the particles dispersion relation. The effective potential contains valuable thermodynamic information on e.g. the equation of state and the system's phase diagram. A main achievement related to our study is the implementation of a numerical library, `libfrg`, which sets up a generic framework for high performance parallel computing in conjunction with the method of functional renormalization. By licensing it under the GNU GPL it is tailored to foster shared development by the community of scientists with research focus on this branch of physics.

Contents

Preface	page	ix
1 Functional renormalization for Statistical Physics	page	1
1.1 When Statistical Physics Meets Field Theory to Go Quantum		3
1.2 A Conceptual Introduction to the Flow of the Effective Action		8
2 On the Road to Non-Relativistic Interacting Fermions	page	25
2.1 Preliminaries		27
2.1.1 Setting the stage		27
2.1.2 An Intermezzo on Super-Matrices		28
2.1.3 On the <i>Truncation</i> and the Graphical Representation of the Flow Equation		33
2.1.4 Momentum Representation of the Effective Averaged Action Γ_k		42
2.1.5 An Important Property of Inverse Propagators		46
2.2 Doing the Math		51
2.2.1 Facing the Full Inverse Propagator $\Gamma_k^{(2)}$		51
2.2.2 The Flowing Effective Potential $U_k(\rho)$		53
2.2.3 Deriving the Flow of the Inverse Propagators		56
2.3 The Story Revised: Flowing Into the Broken Phase		61
3 libfrg — A Numerical Library for the Flow Equation	page	71
3.1 From Theory to Practice – The Call for Numerics		73
3.2 Aspects of the libfrg Library’s Implementation		75
3.2.1 The Code’s Conceptual Structure		75
3.2.2 When Things Grow Big		80
3.2.3 Logging and Documentation		89
3.2.4 The STr in $\dot{\Gamma} = \frac{1}{2} \text{STr} G \dot{R}$		92
3.3 Details on Implemented Classes of Physical Objects		98
3.3.1 The Effective Potential – Γ_k at Constant Field		98
3.3.2 The (inverse) Propagator – Quadratic Fluctuations of Γ_k		103
3.3.3 The Regulator – Driving the Flow		107
3.4 A Short Note on Licensing		118
4 First Application to Statistical Physics	page	121

4.1	Flow Equations for a Relativistic Gas of Interacting Bosons	123
4.2	Critical Physics of Superfluidity	129
4.2.1	Technically Benchmarking the Flow	130
4.2.2	Detecting Fixed Points & Recovering Mean Field Theory	134
4.3	The Story Revised: Scaling Form of the Flow Equations	140
	Conclusion & Perspectives	page 147
	A Remarks on the Fourier transform of the Flow Equation	page 151
	B Choice of Units	page 155
	C From Laurent to Fourier to Chebyshev via Runge	page 161
	D Flow Equations for Interacting Fermions with Spin-Imbalance	page 171
	Bibliography	page 179

Preface

It was shortly after I did finish my undergraduate studies in physics when I learned about complex dynamics that is just *around the corner* waiting to challenge/limit a purely mathematical approach to quite simple model systems. In those days a set of three rather straightforward, but non-linear first order, ordinary differential equations, the *Lorenz equations*^[Lor63], of the form

$$\frac{d\vec{x}}{dt} = \vec{f}(\vec{x}) \quad (1)$$

made a significant impact on my research direction as a physicist. The complexity hidden behind the *innocent-looking* mathematics becomes uncovered by simulating the evolution of \vec{x} . For a certain set of parameters it reveals the so called *Lorenz attractor*. In fact, numerical investigation crucially boosted the interest in the system and just recently, an issue from Steve Smale's *list of mathematical problems to be solved in the 21th century*^[Sma98] has been successfully cracked by a hybrid ansatz employing mathematics as well as computational algorithms^[Tuc02].

Another showcase based on a simple iterative procedure, but subject to complicated studies provides the so called *Mandelbrot set* which we depicted in [fig. 1](#). It visually demonstrates the improvement and success of numerical methods to investigate such problems. The top plot of the left row is an edited version of numerical computation from the late 1970s. Below is the same analysis with state of the art computing power/precision. In order to grasp the origin of the figure, take the series/iteration

$$z_{n+1} = z_n^2 + c \quad \text{with} \quad z_n, c \in \mathbb{C} \quad \text{for all} \quad n \in \mathbb{N} \quad , \quad (2)$$

where c denotes a fixed constant and the initial value z_0 equals zero. In the limit $n \rightarrow \infty$ the complex plane parametrized by c decays into two classes of points: a) $\lim_{n \rightarrow \infty} |z_n| \rightarrow \infty$ or b) $0 \leq |z_n| \leq \text{const.}$. Situation b) defines the Mandelbrot set (black stars). The constant is identified to be at most² 2.

The lesson I kept in mind: Numerics helps to understand theoretical models and the other way around. I started becoming more confident with computational aspects and the art of investigating physical

¹The pictures have been produced by [Wolfgang Beyer](#), LIFE Center, LMU Munich.

²This might be understood as follows: The complex map $z \mapsto z^2$ leaves the unit circle $|z| = 1$ invariant. The interior $|z| < 1$ is mapped into itself. The same holds for the exterior $|z| > 1$. Thus, for $c = 0$ the series, [eq. \(2\)](#), with $|z_1| > 1$ grows to infinity. $c \neq 0$ accounts for a linear shift of z^2 before remapping. Since any c with $|c| > 2$ shifts the unit disk $|z| \leq 1$ such that its intersection with $|z| \leq 1$ is empty, all $z_1 = c$ with $|z_1| > 2$ yield divergent series.

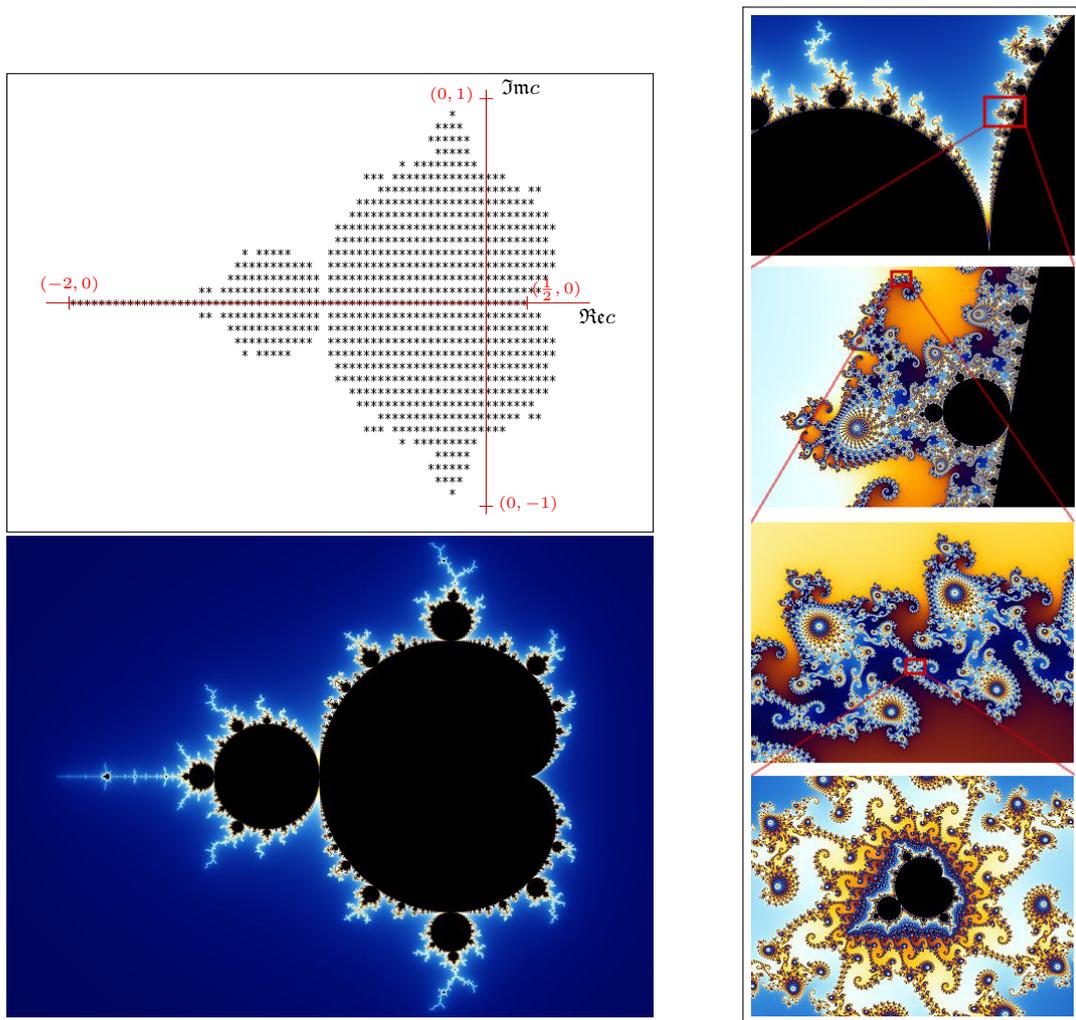


Figure 1: *Plotting the Mandelbrot set.* Illustration of the technical progress in visualizing/numerically computing the Mandelbrot set^[Man83] M from the late 1970s^[BM78] (left upper panel) to present¹ (left lower panel). The color encoding for modern plots stores how fast z_n escapes to infinity: The brighter the color the faster the divergence. In practice one might e.g. define a maximal number N for iterating eq. (2). The brightness of points $c \notin M$ can be defined through $1 - m/N$ with m the value of n where $|z_n| > 2$ for the first time. In order to get maximal contrast at the boundary of M which is drawn in black, it is convenient to invert the brightness to m/N .

The sequence to the right exemplifies the (nearly) *self-similar* character of the Mandelbrot set which is related to the concept of scaling for second order phase transitions of statistical physics. Zooming into the boundary region of M yields structures which are similar to the whole M . However, it is not exactly the same as expected for a (*perfect*) fixed point of second order phase transitions.

systems with the aid of computing power. With my focus on non-relativistic, many-particle quantum physics I got attracted by the tremendous progress of controlling cold quantum gases to simulate and study quantum phases of matter. This field rapidly grew since the first realization of a Bose–Einstein condensate^[DMA⁺95,AEM⁺95] in the mid 1990s. While my master thesis primarily aimed at pure numerics I realized the technical power of functional renormalization^[Wet93] to approach many-body physics from the perspective of quantum field theory.

Back to the structure of eq. (1) one might ask for so called *fixed points* \vec{x}_* where $\vec{f}(\vec{x}_*) = 0$. Then, the system becomes *static*, i.e. $\vec{x} = \text{const.}$ and its behavior near such special points is of particular importance for the system’s dynamics. In addition, the topology of fixed points in the space where \vec{x} is defined reveals major information on the system’s dynamics. In fact, integrating a single path $\vec{x}(t)$ with initial condition $\vec{x}(t_0) = \vec{x}_0$ exhibits the property of *deterministic chaos*^[Str94], i.e. small deviation from \vec{x}_0 exponentially grow with time t . Hence, an approach purely relying on computing power will eventually fail due to finite numerical precision.

The two examples from the beginning provide a convenient analogy to summarize some of the concepts related to functional renormalization. Turning to *time* steps $t = n\Delta t$ we might view eq. (2) as a discrete version of eq. (1) in the sense that a system’s state $\vec{x}(t)$ and z_n is *advanced* in time, respectively. There is a corresponding β -function which *drives* this evolution: $\vec{f}(\vec{x})$ and $z^2 + c$, respectively. As we will introduce in chapter 1 the so called *effective action* Γ is capable to characterize the (macroscopic) quantum state of a many-particle system. Functional renormalization provides an equation to evolve a given (microscopic) system defined through the initial (classical) action $S \sim \Gamma_{t=0}$ towards the macroscopic physics given by $\Gamma_{t \rightarrow -\infty}$ (the negative time direction is pure convention). The *flow equation* is exactly of the same type as eq. (1). Thus, fixed points correspond to scale invariance: On the path to the macroscopic phenomena the theory Γ_t stays fixed/unaltered. In a sense the system becomes *self-similar*. This property can be also found by studying the microscopic details of the Mandelbrot set, as shown by the right row of fig. 1. Sufficient *zooming* yields a structure that is similar to the whole set M .

Conceptually related to self-similarity is the notion of *second order phase transitions* where macroscopic structure continuously emerges. Imagine a scalar function $y(x)$ depending on a single variable x . If it has small fluctuations about a mean value \bar{y} , zooming out/rescaling/averaging the function will eventually lead to $y(x) = \bar{y}$ which stays constant regardless how much you zoom out/rescale/average—a fixed point is reached.

When we started diving into the subject of functional renormalization, we noticed the lack of a standard numerical library at hand. With what has been said so far, it was a quite natural step to initiate a project that intends to fill this gap between theoretical physics and computer science. The present thesis is the result of our effort towards this issue. We organized it as follows: From theory (chapter 2) to numerics (chapter 3) to application (chapter 4) with some more background information collected as appendices.

From a didactic point of view we hope this thesis is written as coherent as possible. Various footnotes try to provide background information to the main text. Since my education is at the interface between theoretical physics and computer science, strict mathematicians should bear with my less rigorous style of presenting mathematical ideas.

Functional renormalization for Statistical Physics

It is the aim of [section 1.1](#) to provide an elementary overview on the conceptual ideas this thesis incorporates. Basic notions from many-body quantum physics/(non-relativistic) quantum field theories are introduced, outlining the road from *density matrices* to *coherent states* to *path integrals* to the *effective action*—the central object quantum observables are derived from. Standard text books as e.g. [\[AS10,ZJO02,Zee05\]](#) contain this material. The reader familiar with all those concepts might safely skip this section, but it also prepares an example continued in [section 1.2](#).

We then turn to an illustrative introduction of the *functional renormalization group* (FRG) in [section 1.2](#). Apart from approaches presented in lecture notes like e.g. ref. [\[Del07\]](#) (with example calculations for $O(N)$ models) or ref. [\[Gie06\]](#) (with focus on gauge theories) we would like to generally comment on scaling by discussing a simple mathematical toy model and embed the FRG into the historical context of the renormalization group, first. Thereafter we discuss the central idea that leads to the *flow equation* by examine a zero-dimensional field theory with focus on *spontaneous symmetry breaking* (SSB) and different choices of *projection prescriptions*. Finally we investigate and illustrate the basis independence of the flow equation.

1.1	When Statistical Physics Meets Field Theory to Go Quantum	. . .	3
1.2	A Conceptual Introduction to the Flow of the Effective Action	. . .	8

1.1 When Statistical Physics Meets Field Theory to Go Quantum

To get started we would like to outline the road from classical to quantum statistical physics. Thereafter we discuss the notion of the *effective action* Γ . [Section 1.2](#) provides a specific computational method to derive it where [chapter 3](#) establishes corresponding numerics later on.

Classical statistical physics^[Man88] of a many-body system aims at computing the *partition function*

$$Z_{\text{cl}} = \sum_i \pi_i \quad , \quad (1.1)$$

where the quantity π_i reflects a statistical weight characterizing the system in state i . It is turned into a probability simply by

$$p_i \equiv \frac{\pi_i}{Z_{\text{cl}}} \quad \text{such that} \quad \sum_i p_i = 1 \quad . \quad (1.2)$$

Hence it proves convenient to define

$$W_{\text{cl}} \equiv \ln Z_{\text{cl}} \quad (1.3)$$

as we will recognize [below](#). If one fixes the system's total energy E , Z_{cl} represents the *microscopic ensemble*. However, one might be interested in the situation where the system is embedded into a thermodynamic *heat bath* which fixes the (inverse) temperature T (β), but allows for energy fluctuations (*canonical ensemble*). Moreover, there could be the need to include *particle number* N fluctuations where one fixes the *chemical potential* $\mu \equiv \frac{\partial E}{\partial N}$ (*grand canonical ensemble*). Then, the corresponding probabilities are computed according to the *Boltzmann weights*

$$\pi_i = \exp(-\beta E_i) \quad (\text{canonical}) \quad \text{or} \quad \pi_i = \exp[-\beta(E_i - \mu N_i)] \quad (\text{grand canonical}) \quad , \quad (1.4)$$

respectively. An exact expression for Z_{cl} is equivalent to the full thermodynamic information and physical observables O_i associated with a given state i are expressed as averaged quantities

$$\langle O \rangle \equiv \sum_i O_i p_i \quad (1.5)$$

due to energy (and particle) fluctuations. By way of example for the grand canonical case, we have:

$$\langle E \rangle - \mu \langle N \rangle = -\partial_\beta W_{\text{cl}} \quad \text{and} \quad \langle N \rangle = \frac{1}{\beta} \partial_\mu W_{\text{cl}} \quad , \quad \text{thus} \quad \langle E \rangle = \left(\frac{\mu}{\beta} \partial_\mu - \partial_\beta \right) W_{\text{cl}} \quad . \quad (1.6)$$

We reviewed this common concept to highlight W_{cl} ($= W_{\text{cl}}(\beta, \mu)$) as the mathematical object from which we easily derive physical observables by taking appropriate derivatives with respect to (inverse) temperature and chemical potential. Up to a suitable *Legendre transform* to be discussed [below](#), W_{cl} is the analog of the effective action/potential for the *path integral* formulation of quantum field theories (QFTs).

The transition to the quantum description of statistical physics involves the definition of the Hermitian, positive semi-definite *density operator/matrix* $\hat{\rho}$ which characterizes an ensemble of states $|\psi_i\rangle$ with associated probabilities p_i , i.e.

$$\hat{\rho} \equiv \sum_i p_i |\psi_i\rangle\langle\psi_i| \quad \text{and hence} \quad \text{Tr } \hat{\rho} = \sum_{i,j} p_i |\langle j|\psi_i\rangle|^2 = 1 \quad , \quad (1.7)$$

where we labeled some orthonormal basis states by $|j\rangle$. Nevertheless, the states $|\psi_i\rangle$ do not necessarily need to fulfill $\langle\psi_i|\psi_j\rangle = \delta_{ij}$ for $i \neq j$, but the *normalization condition* for $i = j$ was used in eq. (1.7). The symbol Tr denotes the trace operator

$$\text{Tr } M \equiv \sum_j M_{jj} \quad \text{of the matrix} \quad M_{jk} \equiv \langle j|M|k\rangle \quad . \quad (1.8)$$

It is important to recognize that $\hat{\rho}$ combines classical statistical as well as quantum physical concepts. Although its trace can be reduced to the sum of p_i , (*off-diagonal elements*)

$$\hat{\rho}_{jk} \equiv \langle j|\hat{\rho}|k\rangle = \sum_i p_i \langle j|\psi_i\rangle\langle\psi_i|k\rangle \quad (1.9)$$

capture information from quantum physics: In contrast to eq. (1.5), the expectation value $\langle\hat{O}\rangle$ of any observable represented by the (quantum) operator \hat{O} should be obviously determined by

$$\langle\hat{O}\rangle = \sum_i p_i \langle\psi_i|\hat{O}|\psi_i\rangle \quad . \quad (1.10)$$

Indeed, it can be expressed with the aid of $\hat{\rho}$ as

$$\langle\hat{O}\rangle = \text{Tr } \hat{\rho}\hat{O} = \sum_{jk} \hat{\rho}_{jk} \hat{O}_{kj} \quad . \quad (1.11)$$

Motivated by eqs. (1.1) and (1.4) we might generalize the partition function to

$$Z \equiv \text{Tr } \exp(-\beta H) \quad (\text{canonical}) \quad \text{or} \quad Z \equiv \text{Tr } \exp\left[-\beta(H - \mu\hat{N})\right] \quad (\text{grand canonical}) \quad , \quad (1.12)$$

where H represents the *microscopic* Hamiltonian, and \hat{N} is the (total) number operator. The corresponding density matrix can be defined by

$$\hat{\rho} \equiv \exp(-\beta H)/Z \quad (\text{canonical}) \quad \text{or} \quad \hat{\rho} \equiv \exp\left[-\beta(H - \mu\hat{N})\right]/Z \quad (\text{grand canonical}) \quad (1.13)$$

in order to generalize the classical p_i . If we (hypothetically) assume the quantum system to be fully specified by the Hamiltonian's eigenvalues E , and if we have total particle number conservation $[H, \hat{N}] = 0$ in addition, we reduce the information contained in the density matrix to the classical situation: $\hat{\rho} = \text{diag}(p_1, \dots, p_i, \dots)$.

To proceed the line of reasoning we need the machinery of *second quantization*¹, that expresses multi-particle quantum physics in terms of annihilation (creation) operators $a_i^{(\dagger)}$ whose (anti-)commutation

¹An easy accessible introduction provides e.g. [AS10], and a brief discussion of the connection to Schrödinger's multi-particle wave function formulation can be found in e.g. [Sre04].

relations

$$[a_i^\dagger, a_j]_{\mp} \equiv a_i^\dagger a_j \mp a_j a_i^\dagger = \delta_{ij} \quad \text{and} \quad [a_i^{(\dagger)}, a_j^{(\dagger)}]_{\mp} = 0 \quad (1.14)$$

specify *bosonic* and *fermionic* particles/statistics, respectively. When acting on a multi-particle state $|\psi\rangle$, the $a_i^{(\dagger)}$ annihilates (creates) a single particle in state² i . Let us label $|\psi\rangle = |n_1, \dots, n_i, \dots\rangle$ the state of definite particle numbers n_i in single particle states i : $a_i^\dagger a_i |\psi\rangle \equiv \hat{n}_i |\psi\rangle \stackrel{!}{=} n_i |\psi\rangle$. Now, take the (free) Hamiltonian

$$H_f(a^\dagger, a) = \sum_i \epsilon_i a_i^\dagger a_i \quad . \quad (1.15)$$

It obviously commutes with all \hat{n}_i —and therefore with total particle number $N(a^\dagger, a) = \sum_i \hat{n}_i$. Therefore, $|\psi\rangle$ is an eigenstate of H_f with corresponding energy $E = \sum_i \epsilon_i n_i$. However, $|n_1, \dots, n_i, \dots\rangle$ is not tailored to diagonalize the density matrices from eq. (1.12). We need to construct the eigenstates of a_i called *coherent states*³ $|\eta\rangle \equiv |\eta_1, \dots, \eta_i, \dots\rangle$ with $a_i |\eta\rangle \stackrel{!}{=} \eta_i |\eta\rangle$ to write the (grand) canonical partition function as a *path integral*

$$Z = \text{Tr} \exp \left[-\beta(H - \mu \hat{N}) \right] = \int \mathcal{D}\eta \exp(-S[\eta]) \quad (1.16)$$

with the *action*

$$S[\eta] \equiv \int_0^\beta d\tau \left[H(\eta^*, \eta) + \sum_i \eta_i^*(\tau) (\partial_\tau - \mu) \eta_i(\tau) \right] \quad (1.17)$$

by splitting the *imaginary time interval* $[0, \beta] \ni \tau$ into infinitesimal partitions and inserting the coherent state resolution of the identity operator

$$1 = \int \mathcal{D}\eta \exp \left[-\sum_i \eta_i^* \eta_i \right] |\eta\rangle \langle \eta| \quad (1.18)$$

Depending on the statistics, $\eta_i^{(*)} \leftrightarrow a_i^{(\dagger)}$ are *ordinary* and *Grassmann* numbers for bosons and fermions, respectively. Silently dropping the vast majority of (probably important) technical details we arrived at the field integral formulation of quantum statistical physics.

It remains to introduce a scheme allowing to derive physical observables. When discussing classical statistical physics [above](#) it turned out being convenient to define $W = \ln Z$ that depends on certain parameters (β, μ) which one can formally use to take derivatives from yielding averages of physical observables, cf. eq. (1.6). To this end we (formally) generalize Z to the functional $Z[J]$ by introducing the (*linear*) *sources* $J_i(\tau)$ as $S[\eta] \rightarrow S[\eta] - J\eta$ with $J\eta \equiv \int d\tau \sum_i \eta_i(\tau) J_i(\tau)$. To avoid diving into a sea of indices we adopt the notational convention that allows dropping labels as τ and i on fields and matrices

²The label i is taken as an arbitrary (*multi*-)index that fully characterizes the given single particle state. E.g. it might represent position x or momentum q plus *internal* degrees of freedom as *spin*.

³For details check standard literature as e.g. [AS10].

when obvious. Taking *functional derivatives* $\frac{\delta}{\delta J} \equiv \delta_J$ of J on $Z[J]$ reveals appropriate (connected) field expectation values:

$$\langle \phi \rangle_c \equiv W^{(1)} \equiv \delta_J W = \langle \eta \rangle \equiv \phi, \quad \langle \phi \phi \rangle_c \equiv W^{(2)} \equiv \delta_{JJ}^2 W = \langle \eta \eta \rangle - \phi \phi, \quad \dots \quad (1.19)$$

We implicitly introduced the *averaging symbol* by

$$\langle \dots \rangle = \int \mathcal{D}\eta(\dots) \exp(-S[\eta] + J\eta) \quad . \quad (1.20)$$

Since $\exp(W[J]) = \int \mathcal{D}\eta \exp(-S[\eta] + J\eta)$ one might regard $-W$ as $S_J \equiv S - J\eta$ *renormalized* by the *summation* $\int \mathcal{D}\eta(\dots)$ of *fluctuations* of η . However, $J \stackrel{!}{=} 0$ in the very end of every computation in order to return to the original partition function, eq. (1.16). Observe that to each fixed source/*external field*⁴ J there is a corresponding *field expectation value* ϕ . It is the purpose of the *effective action* Γ , to express Z as $\exp(-\Gamma[\phi])$ when $J = 0$, i.e. a given Γ is equivalent to the full *statistical information* of the theory: Z is obtained from evaluating $\Gamma[\phi]$ at the (physically observable, *macroscopic*) *field expectation value* ϕ_0 which, in turn, is determined by the extremum of Γ : $\delta_\phi \Gamma = 0$ (cf. eq. (1.22) below).

To achieve the desired goal use is taken of the *Legendre transform*^[ZRM08] of W defined by

$$\Gamma[\phi] = \sup_J (\phi J - W[J]) \quad \text{and, if well defined,} \quad \phi = \delta_J W \quad , \quad (1.21)$$

as expected. Of particular importance are the first and second (functional) derivatives of Γ , due to the relations

$$\Gamma^{(1)} \equiv \delta_\phi \Gamma = J \quad \text{and} \quad \Gamma^{(2)} \equiv \delta_{\phi\phi}^2 \Gamma = \delta_\phi J \Rightarrow 1 = W^{(2)} \Gamma^{(2)} \quad . \quad (1.22)$$

The (real-valued) symmetric matrix $W_{ij}^{(2)} = \delta^2 W / \delta J_i \delta J_j$ is the inverse of $\Gamma_{ij}^{(2)} = \delta^2 \Gamma / \delta \phi_i \delta \phi_j$. Diagonalizing $W^{(2)}$ yields

$$W^{(2)} = \text{diag}(w_1, \dots, w_i, \dots) \quad \text{with} \quad w_i = \langle \eta_i^2 \rangle - \phi_i^2 = \langle (\eta_i - \phi_i)^2 \rangle \geq 0 \quad , \quad (1.23)$$

i.e. $W^{(2)}$ is manifestly *semi-positive definite*, implying the *convexity* of $W[J]$. Since

$$\Gamma^{(2)} = \text{diag}(w_1^{-1}, \dots, w_i^{-1}, \dots) \quad (1.24)$$

follows, $\Gamma[\phi]$ is also a convex functional, i.e. the *quantum equation of motion* $\delta_\phi \Gamma = J = 0$ singles out a unique mean field ϕ_0 (if $w_i > 0$).

⁴According to the *action principle* $\delta S[\eta] = 0$ that leads to the classical *equation of motion* (EoM) for the field η (Euler-Lagrange equation), the source J might be regarded as some *external field* (linearly coupled to η) when considering S_J as representing a *physical system*. J is intimately linked to the subject of *linear response theory*^[AS10,Kub57] which is a significant concept relevant for experimental access to probe a physical system specified by S : A small perturbation of η by applying an external field J and measuring the response of certain observables $O(J)$ yields insight to the system. Therefore, introducing J is beyond a pure mathematical *trick* since it has a direct physical interpretation.

Let us illustrate these ideas with the simplest (trivial) case of the free/non-interacting theory given by the Hamiltonian H_f from eq. (1.15) reduced to a single bosonic degree of freedom ($i \in \{1\}$, $\epsilon \equiv \epsilon_1$) which translates to a $D = 0$ -dimensional (bosonic) Gaussian field integral

$$Z_f(J) = \int d\varphi \exp[-S_f(\varphi) + \varphi J] = \sqrt{\frac{\pi}{\epsilon}} \exp(J^2/4\epsilon) \quad \text{with} \quad S_f(\varphi) = \epsilon\varphi^2, \quad \epsilon > 0 \quad . \quad (1.25)$$

Perturbation theory ^[Sre04] would proceed by approximately determining an interacting theory $S_f(\varphi) \rightarrow S_f(\varphi) + S_{\text{int}}(\varphi)$ as follows:

$$Z(J) = \exp[-S_{\text{int}}(\partial_J)] Z_f(J) \xrightarrow{\text{e.g. } S_{\text{int}}(\varphi) = \lambda\varphi^4} Z(J) \approx Z_f - \lambda \partial_J^4 Z_f + \mathcal{O}(\lambda^2) \quad , \quad (1.26)$$

where $\partial_J^4 Z_f$ is proportional to the (disconnected) *4-point correlation function* $\langle \varphi^4 \rangle$ —up to the constant factor Z . But what if λ , a priori, can not be considered small? This question opens the arena for *non-perturbative* methods as the FRG, subsequently discussed in section 1.2 where we continue this exercise.

The expectation value due to eq. (1.25) reads $\phi = \partial_J \ln Z_f(J) = J/2\epsilon$ and the effective action eventually becomes

$$\Gamma_f(\phi) = \phi J(\phi) - W_f(J(\phi)) = \epsilon\phi^2 - \frac{1}{2} \ln(\pi/\epsilon) = S_f(\phi) + \text{const.} \quad . \quad (1.27)$$

Here, we trivially confirmed the fact that, up to an (irrelevant) constant, quantum fluctuations of φ do not modify the functional form of the effective (macroscopic) action Γ_f compared to the (microscopic) classical action S_f ; no *renormalization* of S_f takes place compared to Γ_f . What about the physical mean field $\phi_0 = \phi|_{J=0}$ (?): It is the trivial one, $\phi_0 = 0$ —the system is said to be within the *symmetric phase* (SYM). If $S \neq S_f$ becomes renormalized such that Γ obtains a nontrivial minimum $\phi_0 \neq 0$, we encounter the physics of *spontaneous symmetry breaking* (SSB). Note, that $\Gamma_f^{(2)} W_f^{(2)} = 2\epsilon \frac{1}{2\epsilon} = 1$ and $Z_f = \exp[-\Gamma(\phi_0)] = \sqrt{\pi/\epsilon}$ as expected.

1.2 A Conceptual Introduction to the Flow of the Effective Action

The idea of continuously connecting the physics at microscopic length scales Λ^{-1} represented by an appropriate action S_Λ to macroscopic observables is the subject of the *renormalization group* [Wil75,Fis74,Sha94b]. The quest for *scaling laws* [Car96], i.e. the formulation of (mathematical) rules that describe the change of some observable $O(x) \rightarrow \tilde{O}(\lambda x)$ under reparametrization of a suitable *scaling variable* $x \rightarrow \lambda x$, is a quite general concept that has been adapted to various branches of physics ranging from *high energy physics* [GML54] to *solid state/condensed matter physics* [Kad90,MK78] to *localization in disordered systems* [And58,AALR79] to *fractals and chaos* [Fei83,Str94]. A historically famous example that successfully employed simple scaling arguments to estimate the energy released by the first nuclear bomb (called *Trinity*) on basis of a time series of snapshots of the blast was worked out by G. Taylor [Tay50a,Tay50b]. A small set of (reasonable) physical assumptions and *scaling relations* sufficed to obtain an adequate answer—no detailed knowledge on e.g. fluid dynamics, gravitation, etc. was needed.

To illustrate the power of scaling let us derive the *Pythagorean theorem* $a^2 + b^2 = c^2$ valid for any right triangle with hypotenuse of length c in Euclidean geometry, cf. fig. 1.1. To do so we note that a triangle is uniquely specified by an angle α and its adjacent sides a and b . Hence the enclosed area is a function of these parameters, $\mathcal{A} = \mathcal{A}(a, b, \alpha)$. Since we are left with a single physical unit, namely *length* L , dimensional (scaling) analysis⁵ yields $[\mathcal{A}] = 2$. Due to the fact that $[a] = 1$, $[b] = 1$, and $[\alpha] = 0$, the triangle's area necessarily needs to fulfill

$$\mathcal{A} = C \left(\alpha, \frac{a}{b} \right) ab, \quad \text{where } [C] = 0 \quad (1.28)$$

is a dimensionless function we are not able to specify without further knowledge on Euclid's geometry—analogueous to a more detailed understanding of the physics involved in e.g. the nuclear blast concerned by Taylor mentioned above. From a pessimistic point of view one might call the factor ab the trivial scaling part of \mathcal{A} whereas C contains the most (interesting) information on the underlying geometry. Indeed, there is a prominent example from cold atomic physics: The so called *Bertsch parameter*⁶ ξ (a pure number) measures the non-trivial dependence of the ground state energy density ϵ_g of an interacting, non-relativistic fermionic gas [Bak99,Hei01] whereas the *Fermi energy* ϵ_F carries the trivial non-interacting part of physics involved: $\epsilon_g \equiv \frac{5}{3} \xi \epsilon_F$ [CCPS03].

However, in cases where there is only *weak* or even no dependence of C on the parameters of the problem we obtain the *scaling solution*

$$\mathcal{A} = \text{const. } ab \quad (1.29)$$

for the right triangle. In our case $\alpha = \frac{\pi}{2}$ is even fixed by definition. If one generally assumes

$$C = \text{const. } \sim 1 \quad (1.30)$$

⁵Our convention to specify a physical quantity/observable O is adopted to the notation $O = \{O\}[[O]]$ with *numerical value* $\{O\}$ and *dimension/physical unit* $[[O]]$. Performing *dimensional analysis*, $[[O]]$ is understood to represent the *power counting* of $[[O]]$ in terms of some basic observable O_b : After having defined (all) physical constants of the problem (e.g. the Planck constant \hbar , speed of light c , particle masses, ...) to be dimensionless, $[[O]]$ collapses to some power of $[[O_b]]$, i.e. $[[O]] = [[O_b]]^{[O]}$. Here we define O_b to be length L and $[L] \equiv 1$.

⁶According to a talk given by Aurel Bulgac (Department of Physics, University of Washington, Seattle, U.S.A.) at the *Bertschfest* (September 7-9, 2012) George F. Bertsch posed the challenge of computing ξ in 1999.

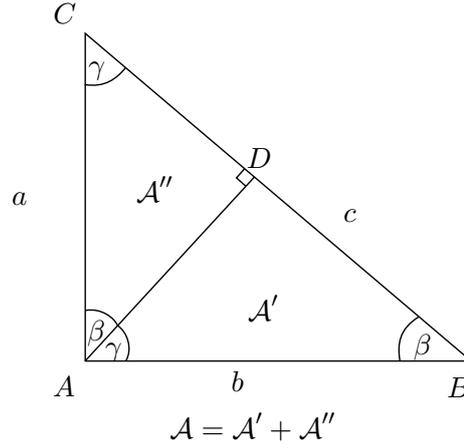


Figure 1.1: *Scaling in a right triangle.* Sketch to illustrate the scaling argument that supports Pythagoras' basic theorem. Since the triangle $\triangle ABC$ is composed of the similar/rescaled versions $\triangle ABD$ and $\triangle ADC$, respectively, the corresponding relation of the areae leads to $a^2 + b^2 = c^2$.

one would *just* miss a factor of order unity, $\frac{1}{2} \sin \alpha$, compared to the *true* area result $\mathcal{A}(a, b, \alpha)$. This sort of analysis worked equally well for the energy of *Trinity*. The upshot might be formulated as follows: The dependence of a physical observable $O \equiv O_c O_s$ can be split into a (trivial) *scaling part* O_s constructed from pure dimensional analysis and a (dimensionless) function O_c with $[O_c] = 0$ that encapsulates the physics beyond pure scaling. If the dependence of O_c on (physical) parameters drops, one might refer to O as some “*universal* quantity, independent from the (microscopic) physical details”. This term of jargon will become more transparent when we introduce the concept of the renormalization group [below](#).

Finally we are in position to argue for Pythagoras' theorem. We now know the scaling law of the area of a (generic) triangle in Euclidean geometry to satisfy

$$\mathcal{A}(\lambda a, \lambda b, \alpha) = \lambda^2 \mathcal{A}(a, b, \alpha) \quad . \quad (1.31)$$

From [fig. 1.1](#) we conclude that the triangle $\triangle ABC$ is composed from similar ones, namely $\triangle ABD$ and $\triangle ADC$ which are downscaled versions of $\triangle ABC$ by factors $\lambda_a = \frac{a}{c}$ and $\lambda_b = \frac{b}{c}$, respectively. Thus, the relation among the individual areas reads

$$\mathcal{A}_{\triangle ABC} = \mathcal{A}(a, b, \frac{\pi}{2}) \equiv \mathcal{A}_0 = \mathcal{A}_{\triangle ABD} + \mathcal{A}_{\triangle ADC} = \lambda_a^2 \mathcal{A}_0 + \lambda_b^2 \mathcal{A}_0 \quad (1.32)$$

from which we immediately read off the desired theorem.

Despite the power of scaling arguments it is the aim of the renormalization group⁷ (RG) to go beyond the approximation [eq. \(1.30\)](#) in the sense of (partially) including *quantum and/or statistical fluctuations*

⁷The term *group* is somehow misleading since the conceptional basis of the renormalization group scheme is loosely linked to the ideas of group theory^[Geo99] and its connection to the subject does not yield much fruitful insight to physics. Due to [\[AS10\]](#) the notion was unfortunately guided by the success of the *Eightfold Way*^[GM62] in particle physics during the 1960s. In the following we prefer the simple term *renormalization*.

to the *classical/microscopic* physics in order to *renormalize* its parameters like e.g. particle masses, local magnetization, etc. to compute an effective theory. This task is performed by successively mapping observables at one scale to another one which is considered to represent the physics at length-, energy-, ... scale of interest. The corresponding mapping is called a *scaling transformation*. Depending on this process one might distinguish different renormalization schemes for which we set up the following (perhaps incomplete and biased) list, [table 1.1](#):

century	field	renormalization scheme/step	ass. physicist(s)	init. paper(s)
1950s	diagrammatic QFT	perturbative ren. , introduces counter-terms to deal with ultraviolet divergences	M. Gell-Mann/ F. E. Low	[GML54]
1960s	(quantum) statistical physics	block spin ren. , averages spatially neighboring degrees of freedom to define a renormalized quantity (initially applied to the Ising model)	L. P. Kadanoff	[Kad66]
1970s	(quantum) field theories	momentum shell ren. , integrates out Fourier modes of the field theory from high energies to low ones	K. Wilson, F. Wegner/ A. Houghton	[Wil71] [WH73]
1980/90s	path/functional integrals	functional ren. , introduces a quadratic (mass) term that (smoothly) regulates the propagator of the theory	J. Polchinski, C. Wetterich	[Pol84] [Wet93]
1990s	quantum lattice systems	density matrix ren. , numerical technique that iteratively diagonalizes the Hamiltonian of specific sub-blocks of the lattice system to extract low energy/ground state physics	S. R. White	[Whi92]
	can. quant. theories	Hamiltonian ren. , introduces a unitary transformation that diagonalizes a given Hamiltonian	K. Wilson/ S. Głazek, F. Wegner	[GW93] [Weg94]

Table 1.1: It should serve as a historically ordered overview on the different flavors of renormalization. We do not claim completeness and for the sake of structure we tried to categorize the set of renormalization procedures due to their conceptual approach which will appear biased, because there are arguments to regroup the contributions from the papers listed. The critical reader should take it as an informative arrangement of initiating literature to different flavors of the renormalization technique in theoretical physics.

As pointed out by F. Wegner^[Weg01], his approach diagonalizes a given Hamiltonian by smoothly decreasing off-diagonal elements while Głazek's/Wilson's approach requires an exact vanishing of those elements when a predefined *energy parameter* drops below some threshold. Moreover, it is possible to group the listed approaches from the 1970s and 1980/90s under the umbrella of the notion of

momentum-space renormalization since these formalisms distinguish between *soft/infrared* (IR) and *hard/ultraviolet* (UV) Fourier modes of the corresponding (quantum) field theory. A *cutoff* is defined at scale k to divide momenta q into the corresponding categories.

On the one hand J. Polchinski's renormalization is similar to C. Wetterich's idea in the sense of introducing a smooth term to the Lagrangian that regulates the propagation of modes. On the other hand Polchinski's concept is much closer to Wilson's momentum shell integration since it redefines the Lagrangian's parameters/couplings, but demands the overall path integral/partition function being unaffected. This is different from Wetterich's equation that explicitly considers the (continuous) variation of the effective action. In addition, his functional differential equation is obtained by suppressing IR modes while Polchinski's cutoff is designed to damp contributions from the UV. Moreover, as we will see [below](#) Wetterich's regulator is not even confined to explicitly distinguish between different modes. As described in e.g. [\[Hal06\]](#), White's renormalization is an extension of Wilson's idea that overcomes certain numerical drawback on errors at the boundary of blocks the system is divided into. From the conceptual point of view it is near to Kadanoff's picture when applied to real-space lattices. Loosely speaking it is also possible to combine Kadanoff's renormalization in *real-space* with the momentum-space concepts, if one recaps that momenta q are associated with distances/wave lengths q^{-1} . Even in the case of Wegner's Hamiltonian renormalization one might argue for an inverse relation between energy (momentum) and length.

A lot more philosophy and perhaps even proofs of equivalence (?) on the interrelation of those methods can be added, but it primarily was our aim to embed all of them into a historical context. From now on we are dealing with the functional renormalization set up by C. Wetterich.

To grasp the central idea of the *flow equation* we will continue our simple⁸ toy model of a $D \equiv d + 1 = 0$ dimensional field theory from [section 1.1](#) with a single degree of freedom⁹ φ . The corresponding theory was given by a *microscopic/classical* action $S_\Lambda(\varphi) = S_f(\varphi)$ (cf. [eq. \(1.25\)](#)) with the *equation of motion* (EoM) $\delta S_\Lambda = 0$, i.e. $\partial_\varphi S_\Lambda = 0$. The corresponding *partition function* $Z(J)$ and the *effective action* $\Gamma(\phi)$ in terms of the *source* J and *expectation value/mean* $\phi \equiv \partial_J W$ read

$$Z(J) = \int d\varphi \exp(-S_\Lambda(\varphi) + J\varphi) \quad \text{with} \quad W(J) \equiv \ln Z(J) \quad \text{and} \quad \Gamma(\phi) = J\phi - W(J) \quad , \quad (1.33)$$

respectively. Since the source J is set to zero in the very end, we have

$$\partial_\varphi S_\Lambda(\varphi) = 0 \quad (\text{classically}) \quad \text{versus} \quad \partial_\phi \Gamma(\phi) = 0 \quad (\text{quantum}) \quad . \quad (1.34)$$

One might wonder if there is a natural way to continuously connect S_Λ and Γ by successively including fluctuations of φ around its mean ϕ . In a first step we use [eq. \(1.33\)](#) to express Γ as a (path) integral

$$\Gamma(\phi) - J\phi = -\ln \left[\int d\varphi \exp(-S(\varphi) + \varphi J) \right] \quad (1.35)$$

⁸Readers in favor of a formal approach may follow ref. [\[Paw07\]](#). Textbooks on (F)RG provide refs. [\[Sal99,KBS10\]](#).

⁹Here we assume $[\varphi] = 0$, and even more generally: All quantities are taken to be dimensionless. For $D > 0$ we have $S = \int \mathcal{L}(\varphi, \partial\varphi, \dots)$. Some *standard* ((non-)relativistic) *kinetic term* $\mathcal{L} \subset \varphi \partial^n \varphi$ sets the dimension of φ due to the requirement $[S] = 0$.

of *some* action S to be specified. If we could *get rid* of the integration, e.g. by inserting

- i) a function $f(\varphi)$ sharply peaked around ϕ , cf. $\delta(\varphi) \approx \exp(-\frac{R}{2}\varphi^2)$, or
- ii) introducing $f(\varphi)$ as highly oscillating for $\varphi \neq \phi$, cf. $\exp(i\frac{R}{2}\varphi^2) = \cos(R\varphi^2/2) + i \sin(R\varphi^2/2)$

with real and *large* R (roughly speaking, to be substantiated **below**) under the integral of eq. (1.35), we ask ourself: Is there the chance to arrive at a correspondence like “ $\Gamma(\phi) \approx S(\phi)$ ”? As we indicate, useful relations of the exponential function in accordance with a suitable shift $\varphi \rightarrow \varphi - \phi$ can be used to model appropriate $f(\varphi)$ s; a key ingredient of this construction: $R = 0$ leaves the original theory untouched and the full effective action, eq. (1.35), gets recovered. Note that it is possible to unify i) and ii) by allowing R to become complex valued with $\Re R > 0$. The prefactor $\frac{1}{2}$ is just for convenience. Labeling the *regulator* R by the so called *flow/scale parameter* $k \in \mathbb{R}$, we require the following general properties:

$$|R_k| \xrightarrow{k \rightarrow 0} 0 \quad \text{and} \quad |R_k| \xrightarrow{k \rightarrow \Lambda' \rightarrow \infty} \infty \quad . \quad (1.36)$$

Simple choices as $R_k = k^2$ or $R_k = ik^2$ satisfy eq. (1.36).¹⁰ In order to rewrite eq. (1.35), we express the integrand

$$\exp[-S(\varphi) + (\varphi - \phi)J] \quad \text{as} \quad \exp[\dots] \exp\left[-\frac{R_k}{2}(\varphi - \phi)^2\right] \quad . \quad (1.37)$$

Therefore, we (formally) decompose the quantities from eq. (1.35) according to the prescriptions

$$\begin{aligned} S(\varphi) &\rightarrow S_k(\varphi) = S_\Lambda(\varphi) + \frac{R_k}{2}\varphi^2 \\ \Gamma(\phi) &\rightarrow \Gamma_k(\phi) = \tilde{\Gamma}_k(\phi) + \frac{R_k}{2}\phi^2 \\ J(\phi) &\rightarrow J_k(\phi) = \tilde{J}_k(\phi) + R_k\phi \quad , \end{aligned} \quad (1.38)$$

which define the *explicit* dependence of the classical action¹¹ S , the effective action Γ and the source J on the regulator, respectively. The *bare* action S_Λ contains the microscopic physics for which we eventually would like to compute observables, and S_k describes the same physics, but with additional quadratic (*mass*) term $\frac{1}{2}R_k\varphi^2$. The symbol $\tilde{\Gamma}_k$ subtracts this term from the effective action Γ_k on the level of the (*macroscopic*) field ϕ . Furthermore, it turns out that $\tilde{\Gamma}_k$ (smoothly) interpolates S_Λ to

$$\Gamma = \tilde{\Gamma}_{k=0} = \Gamma_{k=0} \quad (1.39)$$

by means of the flow equation, eq. (1.48), derived below.

$$\tilde{\Gamma}_k(\phi) = -\ln \left[\int d\varphi \exp \left(-S_\Lambda(\varphi) - \frac{1}{2}R_k(\varphi - \phi)^2 + (\varphi - \phi)\tilde{J}_k \right) \right] \quad (1.40)$$

¹⁰ Again, when turning to $\varphi \rightarrow \varphi(x)$ there is the additional freedom to choose different R_k for e.g. all Fourier modes $\varphi(q)$. Typically, one writes $R_k(q) = k^2 r(q/k)$ with $r(y) \sim 1$ for $|y| \in [0, 1]$ and $r \rightarrow 0$ for $|y| \rightarrow \infty$ without a Fermi surface present. Those criteria imitate the Wilsonian momentum-shell renormalization. We address this issue in [section 3.3.3](#).

¹¹In order to avoid confusion, let us emphasize that $S_\Lambda \neq S_{k=\Lambda}$ —in contrast to our practice of abbreviating $\Gamma_{k=\Lambda}$ and $J_{k=\Lambda}$ by Γ_Λ and J_Λ , respectively.

is known as the *averaged effective action*¹². Thus, adding a quadratic (*mass*) term $\frac{1}{2}R_k\varphi^2$ to our (classical) theory described by S_Λ can be exploited to directly relate the classical action to the (averaged) effective action $\tilde{\Gamma}_k^{(\sim)}$. This happens for

$$\sigma_R \equiv |R|^{-1/2} \quad (1.41)$$

significantly below any scale

$$\sigma_\Lambda \equiv \Lambda^{-1} \quad (1.42)$$

set by the fluctuations of $S_\Lambda(\varphi)$. Then, no quantum fluctuations are included into $\tilde{\Gamma}_k^{(\sim)}$ at all. Figure 1.2 schematically depicts the idea behind a real and imaginary regulator R_k which stems from the method of *steepest descent*^[Deb09,AW05]. However, the second condition of eq. (1.36) has to be understood in the (loosely formulated) sense recently described: $S_\Lambda(\varphi)$ is considered to represent a parameter-dependent function that sets some (minimal) scale¹³ σ_Λ below which S_Λ does not significantly vary¹⁴, i.e. it is assumed to be save to apply the approximation $S_\Lambda(\varphi + \sigma_\Lambda) \approx S_\Lambda(\varphi)$. Therefore, when $\frac{\sigma_R}{\sigma_\Lambda} \ll 1$, typically when $k = \Lambda' \gg \Lambda$, the factor $\exp[-S_\Lambda(\phi)]$ pulls out of the integration over φ in eq. (1.40) where it remains to solve a Gaussian integral. In practice one will choose a finite Λ' to start the flow. This approximation adds Λ' -dependent counter-terms to the flowing parameters/couplings of $\tilde{\Gamma}_{k=\Lambda'}$ in the sense:

$$\tilde{\Gamma}_{\Lambda'}(\phi) \approx S_\Lambda(\phi) + \tilde{s}(\phi, \Lambda') + c(\Lambda') \quad \text{with} \quad \tilde{s}(\phi, \Lambda') \equiv -\frac{\tilde{J}_{\Lambda'}^2}{2R_{\Lambda'}}, \quad c(\Lambda') \equiv \frac{1}{2} \ln \frac{R_{\Lambda'}}{2\pi} \quad (1.43)$$

from the Gaussian integration for $R_k \xrightarrow{k=\Lambda' \rightarrow \infty} \infty$. Since $\tilde{s}(\phi, \Lambda') = J_{\Lambda'}\phi - \frac{1}{2}R_{\Lambda'}\phi^2 - J_{\Lambda'}^2/2R_{\Lambda'}$ we might reformulate eq. (1.43) as

$$-W_{\Lambda'} = \Gamma_{\Lambda'}(\phi) - J_{\Lambda'}(\phi)\phi \approx S_\Lambda(\phi) + s(\phi, \Lambda') + c(\Lambda') \quad \text{with} \quad s(\phi, \Lambda') \equiv -\frac{J_{\Lambda'}^2}{2R_{\Lambda'}} \quad (1.44)$$

¹²Most literature on FRG does refer to $\tilde{\Gamma}_k$ below as the *averaged effective action*; as well as labeling it by Γ_k . Although we stick to this convention it seems a bit misleading since $\tilde{\Gamma}_k$ is not the Legendre transform of $W_k = \ln \int d\varphi \exp(-S_k(\varphi) + J_k\varphi)$ for $k \neq 0$ and hence it does not fulfill typical properties like being convex in general—in contrast our definition of Γ_k is the Legendre transform of W_k .

¹³As an example for a function that introduces a scale one might consider the *spontaneously symmetry broken* (SSB) action $S(\varphi) = -a\varphi^2 + \lambda\varphi^4$ with parameters $a, \lambda > 0$ which set the *condensate* $\varphi_0^2 = a/2\lambda$ identified with the local minima of S . If one would *resolve/probe* $S(\varphi)$ on *distances* $\Delta\varphi > \varphi_0$ the feature of the local minimum $\varphi_0 \neq 0$ of S would not be appropriately incorporated into the problem. However, for e.g. $\Delta\varphi = \varphi_0/1000$ no new details of S will be explored compared to $\Delta\varphi = \varphi_0/100$. Nevertheless there are simple functions like e.g. an *algebraic decay* $a\varphi^{-1}$ where there is no obvious scale to be associated with. Such functions are commonly called *scale invariant* since any change in $\Delta\varphi$ does not shed new light on them. Another term in use in this context is *self-similarity*. It became popular in studies of fractal geometry like the *Mandelbrot set*^[Man83,Fal03].

¹⁴There is an additional aspect of Λ in the case of $D > 0$ where $\varphi \rightarrow \varphi(x)$ —with x a (continuous) label/parametrization of the (space) manifold of dimension D —the field variable(s), i.e. degrees of freedom, $\varphi(x)$ can be considered to be composed of Fourier modes $\varphi(q)$; another set of degrees of freedom to describe the physics. Then, the functional $S_\Lambda[\varphi] \in \mathbb{R}$ receives contributions from all these modes and $\sigma_\Lambda = \Lambda^{-1}$ represents a *distance measure* $|x| \in \mathbb{R}$ below which $\varphi(q)$ with $q > \Lambda$ do not (significantly) contribute to the value of S_Λ .

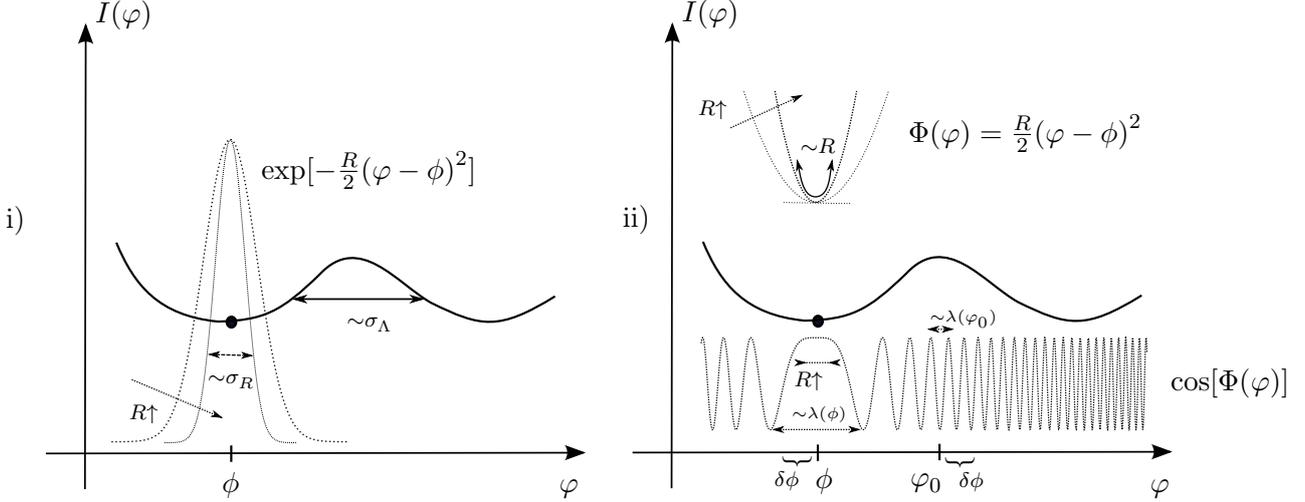


Figure 1.2: Schematics of the i) real and ii) imaginary regulator R . In the case of a real regulator the term $\mathcal{G}(\varphi) \equiv \exp[-\frac{R}{2}(\varphi - \phi)^2]$ multiplies a Gaussian to the integrand $I(\varphi) \equiv \exp[-S_\Lambda(\varphi) + \tilde{J}_k(\varphi - \phi)]$. If the smallest scale σ_Λ of S_Λ is negligible compared to the width σ_R of the Gaussian, eq. (1.41), the integration over φ collapses to a Gaussian integral with (constant) prefactor $\exp[-S_\Lambda(\phi)]$.

When R is purely imaginary, $\mathcal{G}(\varphi)$ becomes an oscillating function $\exp[-i\Phi(\varphi)]$ with phase $\Phi(\varphi) = \frac{R}{2}(\varphi - \phi)^2$. Although there is no meaningful notion to characterize $\mathcal{G}(\varphi)$ by a single (global) wavelength λ for arbitrary $\Phi(\varphi)$, it is possible to define this quantity locally by fixing some φ_0 to Taylor expand around: $\Phi(\varphi) = \Phi(\varphi_0) + \Phi'(\varphi_0)\delta\varphi + \mathcal{O}(\delta\varphi^2)$ with $\delta\varphi \equiv \varphi - \varphi_0$. The local wavelength is defined as $\lambda(\varphi_0) \equiv 2\pi/\Phi'(\varphi_0) = 2\pi/R(\varphi_0 - \phi)$. If $\lambda \lesssim \sigma_\Lambda$, eq. (1.42), i.e. \mathcal{G} oscillates on scales where $S_\Lambda(\varphi)$ (and hence $I(\varphi)$) is approximately constant, the contribution to the integral cancels. Therefore, we need large λ for $I(\varphi)$ to survive. This requirement translates to a stationary phase $\Phi'(\varphi_0) \approx 0$. In particular, in the limit where $R \rightarrow \infty$, we need $\phi_0 \rightarrow \phi$ in order for λ to stay finite. Thus, $\exp[-S_\Lambda(\phi)]$ becomes singled out again.

Note that the exact form of $\tilde{J}_k^{(\sim)}$ determines whether $\tilde{s}^{(\sim)} \approx 0$ is a reasonable approximation for $k \rightarrow \infty$. Indeed, as we will see below, the free theory yields $\tilde{s} \sim R_\Lambda^{-1}$ and $s \sim R_\Lambda'$, respectively. This observation is consistent with the introduced explicit splitting of J_k from eq. (1.38). In practice, physical observables from $\tilde{\Gamma}_{k=0}$ have to be employed in order to fix the parameters of $\tilde{\Gamma}_{k=\Lambda'}$. Since we know¹⁵/model S_Λ according to a given (microscopic) theory, an expression for the (differential) variation $\partial_k \tilde{\Gamma}_k$ is desired to integrate down to Γ . Due to the possible physical interpretation of the scale parameter k , it has momentum dimension $[k] = 1$. Hence, it is common to switch to the dimensionless (logarithmic) derivative

$$\partial_t \equiv k\partial_k \quad \text{with time } t \equiv \ln(k/\Lambda') \quad \text{and the notational abbreviation } \dot{\Gamma}_k \equiv k\partial_k \Gamma_k \quad . \quad (1.45)$$

¹⁵ Actually, there is an opposite point of view when applying the FRG idea to quantum gravity: There one starts with some known action Γ —the Einstein-Hilbert action—in the IR and extrapolates *back* to the UV to search for a possible theory that leads to Einstein's theory of general relativity when including quantum fluctuations.^[Reu98] At a possible fixed point Γ_∞ , the averaged effective action Γ_k converges to in the limit $k \rightarrow \infty$, is directly linked to the asymptotic safety scenario^[WHI79,Nie06] introduced by S. Weinberg.

Thus, the flowing time starts at $t = 0$ ($k = \Lambda'$) running down to $t \rightarrow -\infty$ ($k = 0$). Let us investigate the explicit variation of Γ_k with respect to t when R_k decreases/increases with k . We take eqs. (1.35) and (1.38) to conclude

$$\dot{\Gamma}_k = -\dot{W}_k + \dot{J}_k(\phi - \partial_J W_k) = +\frac{1}{2}\langle\varphi^2\rangle\dot{R}_k + 0 \quad . \quad (1.46)$$

Expressing $\langle\varphi^2\rangle$ in terms of the second derivative of W_k ($\partial_J^2 W = \langle\varphi^2\rangle - \phi^2$) as well as using the inverse relation of the second derivatives of the Legendre transforms, namely $W_k^{(2)}\Gamma_k^{(2)} = 1$, we arrive at

$$\dot{\Gamma}_k = \frac{1}{2}\left(W_k^{(2)} + \phi^2\right)\dot{R}_k \quad \Rightarrow \quad \partial_t\left(\Gamma_k - \frac{1}{2}R_k\phi^2\right) = \frac{1}{2}\left(\Gamma_k^{(2)}\right)^{-1}\dot{R}_k \quad . \quad (1.47)$$

An appropriate redefinition of Γ_k by

$$\tilde{\Gamma}_k \equiv \Gamma_k - \frac{1}{2}R_k\phi^2 \quad \text{yields the flow equation} \quad \dot{\tilde{\Gamma}}_k(\phi) = \frac{1}{2}\left(\tilde{\Gamma}_k^{(2)}(\phi) + R_k\right)^{-1}\dot{R}_k = \frac{1}{2}G_k\dot{R}_k \quad , \quad (1.48)$$

where we denote the (full) quantum propagator of the theory specified via $S_k(\varphi) \equiv S_\Lambda(\varphi) + \frac{1}{2}R_k\varphi^2$ by

$$G_k(\phi) \equiv W_k^{(2)}(J(\phi)) = \left(\Gamma_k^{(2)}(\phi)\right)^{-1} = \left(\tilde{\Gamma}_k^{(2)}(\phi) + R_k\right)^{-1} \geq 0 \quad , \quad (1.49)$$

which is positive as more generally pointed out in eq. (1.23). Note, that we did not explicitly use the *form* of the derivative—the same equation is valid for ∂_t substituted by ∂_k or any other derivative defined through $t = t(k)$. Moreover one should be aware of the fact that $\tilde{\Gamma}_k$ is only equal to the Legendre transform Γ_k of W_k at $k = 0$, cf. eq. (1.39).

Let us understand the derived result, eq. (1.48). It is an implicit, *self-consistent* equation, since the rhs. provides the variation of $\tilde{\Gamma}_k$ depending on (the second derivative of) $\tilde{\Gamma}_k$ itself. We illustrate the situation in fig. 1.3. In order to proceed any further, it is necessary to start with some ansatz for $\tilde{\Gamma}_k$. This step is commonly denoted as *truncation* within the *business* of the FRG. An obvious option at hand would be to assume that all n th derivatives above some value n_0 simply vanish: $\tilde{\Gamma}_k^{(n>n_0)} \stackrel{!}{=} 0$, i.e. one deals with a polynomial ansatz in ϕ for $\tilde{\Gamma}_k(\phi)$. But there is an intimate drawback coming along with this approach. Assume we have fixed a polynomial expansion of $\tilde{\Gamma}_k$ to order n_0 around some (at the moment arbitrarily chosen) field value ϕ_0 . Inserting it to the lhs. of the flow equation lets the ∂_t -derivative act on the polynomial's coefficients $\tilde{\gamma}_k$. On the other hand, an expansion of $(\tilde{\Gamma}_k(\phi) + R_k)^{-1}$ on the rhs. will yield some polynomial¹⁶ $\beta(\phi)$ to infinite order in ϕ . Depending on whether coefficients β_n in front of $(\phi - \phi_0)^n$ with $n > n_0$ are *significantly different* from zero or not the chosen polynomial truncation would turn out to be insufficient. In cases where one is allowed to ignore the $\beta_{n>n_0}$ —and this is by no means obvious—the resulting set of equations on the variation of the Taylor expansion coefficients read

$$\dot{\tilde{\gamma}}_n = \partial_\phi^n \dot{\tilde{\Gamma}}_k(\phi) \Big|_{\phi=\phi_0} = \frac{1}{2} \partial_\phi^n G_k(\phi) \dot{R}_k \Big|_{\phi=\phi_0} = \beta_n \quad \text{where} \quad 0 \leq n \leq n_0 \quad , \quad (1.50)$$

and the inner two expressions of the chain of equalities are denoted by a *projection prescription* of the flow equation. In cases where $D > 0$ one should even consider mixtures of polynomials in fields ϕ and

¹⁶The notation was chosen on purpose. One might cast the flow equation's structure into the form $\dot{y}(t, x) = \beta(y'')$ with the *velocity field* β aka. *beta-function* within the framework of renormalization.

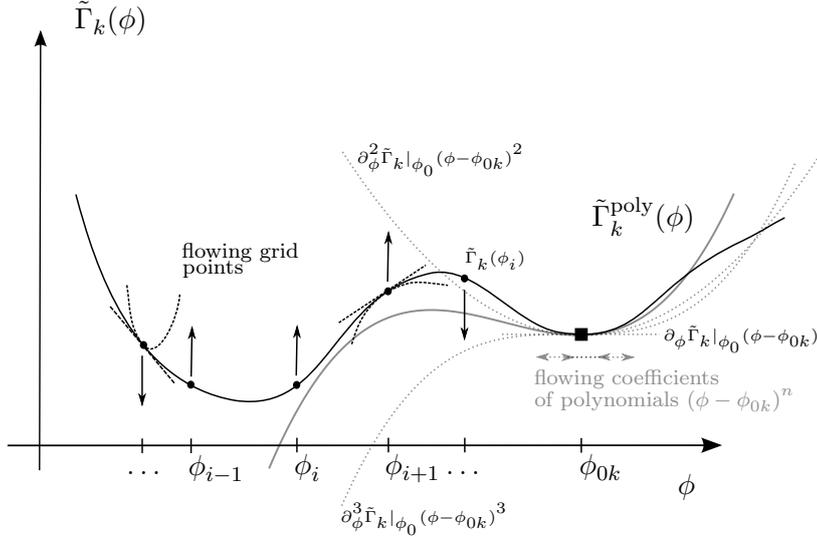


Figure 1.3: *Polynomial and discrete truncation.* The plot above sketches two schemes how to truncate the averaged effective action $\tilde{\Gamma}_k$ in order to obtain a practically tractable set of equations from the flow equation, eq. (1.48). One option (right part of figure) is to (dynamically) expand $\tilde{\Gamma}_k$ around some value ϕ_{0k} (indicated by the \blacksquare). Usually one would like to take a local minimum, since the quantities $\partial_\phi^n \tilde{\Gamma}_k|_{\phi_0}$ might be interpreted as (low energy) observables of physics at scale k . In this case one successively improves the *local* dependence of $\tilde{\Gamma}_k^{\text{poly}}$ on ϕ in the neighborhood of ϕ_{0k} towards the *real* $\tilde{\Gamma}_k$.

However, apart from technical issues, it might be of importance to capture *global* aspects of the averaged effective action, e.g. to handle first order phase transitions where one needs information on the global minimum of $\tilde{\Gamma}_k$. To this end, it is preferable to employ a (discrete) *grid* as a suitable truncation, see left part of sketch. Evaluating the rhs. of the flow equation for fixed ϕ_i yields the variation (indicated by $\uparrow\downarrow$) of the grid point $\tilde{\Gamma}_k(\phi_i)$ (labeled by \bullet) from the lhs. which depends on the second derivative $\tilde{\Gamma}_k^{(2)}(\phi_i)$ in turn (dashed parabolae).

their (spatial) derivatives $\partial\phi$ in addition: $\phi, \phi\phi, \dots, \partial\phi, \phi\partial\phi, \phi\phi\partial\phi, \dots, \partial\phi\partial\phi, \phi\partial\phi\partial\phi, \dots$; diving into a sea of numerous combinations of possible terms¹⁷. To reduce their cardinal number one might apply symmetry principles, (dimensional) power counting arguments as well as physical insight to estimate their relevance when flowing down to $k = 0$. However, in principle there is no *small parameter* compared to e.g. perturbation theory which one could take as a measure for the *reliability* of computed physical quantities. Nevertheless FRG calculations made their success public on various issues of (theoretical) physics as

- UV extension of QED^[GJ04],
- asymptotic safety in quantum gravity^[Reu98],
- the Higgs mass^[SW09],
- critical exponents of $O(N)$ models^[TW94],
- the phase diagram of QCD^[Paw10],

¹⁷... and names to possible truncation schemes^[BTW00]: *vertex expansion, derivative expansion, canonical expansion, ...*

- the BCS–BEC crossover^[DFG⁺10],
- the Hubbard-model^[HS01a],
- radio-frequency spectra of a polaron^[SE11],
- the Efimov-effect^[FMS11],
- disordered systems^[TT07],
- quantum non-equilibrium dynamics^[BM12],
- etc. .

Especially the fact that there is no small parameter restricted to be kept small enables the method to address non-perturbative aspects; however, with *biased* touch due to a truncation of $\tilde{\Gamma}_k$ that has to be appropriately chosen in order to capture the relevant physics which one a priori does not exactly know about. Intuition has to guide the computation. Hence FRG will not be the natural choice for exploring completely unknown territory of non-perturbative physics as *strongly correlated systems*¹⁸. In such a situation of *blindness* e.g. unbiased (quantum) Monte Carlo^[LB05] techniques might be the preferred tool to serve as a first scout. However, when a great deal is already known about a physical system inaccessible by perturbative methods, FRG develops its strength in conjunction with known physical facts. It is capable to exclude or stress the relevance of certain physical processes due to the intuitive interpretation of Γ_k and the graphical representation of the flows of $\check{\gamma}_n$ (see [chapter 2](#) for a detailed discussion). and therefore provide further physical insight.

Furthermore we did arrive at the very motivation of this thesis from a technical point of view. Back to the discussion of a polynomial truncation we could avoid trouble considering $\tilde{\Gamma}_k(\phi)$ being represented by a discrete set of values $\tilde{\Gamma}_{ki} \equiv \tilde{\Gamma}_k(\phi_i)$ plus a suitable interpolation in between. We then just numerically evaluate [eq. \(1.48\)](#) at a chosen finite set $\{\phi_i\}$ as previously depicted in [fig. 1.3](#) and there is no conceptual problem in adaptively varying the set at different scales k during integrating the flow equation down to Γ . Additionally, the computational effort is reduced in the sense that there is no necessity to apply an projection prescription like [eq. \(1.50\)](#) to the flow equation. This process is replaced by the choice $\{\phi_i\}$ in accordance with an appropriate interpolation scheme. The flows $\check{\Gamma}_{ki}$ are given by fixing $\phi = \phi_i$ on the rhs. of [eq. \(1.48\)](#). All the (involved) details provides [chapter 3](#).

Despite the advantages over a polynomial ansatz argued above the story of discretization becomes much more involved when $\varphi \rightarrow \varphi(x)$. Then, one has to face the practically impossible task of discretizing an infinite set of variables. Formulating the theory on the grid is twofold: First one has to select discrete values in space(-time) x_j and discretize each variable¹⁹ $\phi_j \equiv \phi(x_j)$ afterwards in order to arrive at a (multi-dimensional) grid of ϕ_{ji} ($i \dots$ grid index of *internal* degrees of freedom, as in [fig. 1.3](#)), $\tilde{\Gamma}_k$ depends on. (ij) might be viewed as a *multi-index*. Details are presented in [section 3.1](#) where we choose a *hybrid*

¹⁸By this term we loosely refer to (quantum) statistical systems where physical observables are determined by not only a few correlation functions $W^{(n)}$ with $n \sim 1$. Sometimes this phrase is confused with *strongly interacting* systems which might trace back to the fact that *free* (quantum) field theories are described by a *Gaussian* probability distribution $\exp(-S_\Lambda^f(\varphi))$ with $S_\Lambda^f = a\varphi^2$ where all $W^{(n>2)}$ vanish.

¹⁹Which, in turn, might be multi-dimensional due to *internal* degrees of freedom. E.g. the field $\phi(x)$ representing bosonic particles of spin 0 is complex-valued, cf. [eq. \(1.60\)](#).

truncation combining ideas from the polynomial ansatz and the grid. The approach is guided by the notion *from local to non-local* as described in place.

Now it is appropriate to continue our exercise from [section 1.1](#) in order to check the derivation of the flow equation. It provides a practical demonstration; Moreover, referring back to [eq. \(1.43\)](#), the *initial condition* turns into

$$\tilde{\Gamma}_{\Lambda'}(\phi) \approx S_f(\phi) + 0 + c(\Lambda') \approx \epsilon\phi^2 + \frac{1}{2} \ln(R_{\Lambda'}/2\pi) \quad , \quad (1.51)$$

where use was taken of the following facts:

$$\begin{aligned} a) \quad & S_k(\varphi) = S_f(\varphi) + \frac{1}{2} R_k \varphi^2 & = (\epsilon + R_k/2)\varphi^2 \\ b) \quad & \phi = \partial_J W_k & = J/(2\epsilon + R_k) \\ c) \quad & \tilde{J}_k = J_k - R_k \phi & = 2\epsilon\phi \quad . \end{aligned} \quad (1.52)$$

When fixing ϕ to some finite value we obtain: $\tilde{s}_{\Lambda'} = \tilde{J}_{\Lambda'}^2/2R_{\Lambda'} \rightarrow 0$ for $\Lambda'/\Lambda \rightarrow \infty$ and thus the approximation, [eq. \(1.51\)](#), assumes $R_{\Lambda'} \sim \Lambda'^2 \gg \epsilon \sim \Lambda^2$. Motivated by the form of $\tilde{\Gamma}_{\Lambda'}$ (and actually the known exact result, [eq. \(1.27\)](#)) we consider the polynomial truncation

$$\tilde{\Gamma}_k(\phi) \stackrel{!}{=} \epsilon_k \phi^2 + c_k \quad (1.53)$$

with *initial conditions*

$$\epsilon_{\Lambda} = \epsilon \quad \text{and} \quad c_{\Lambda} = \frac{1}{2} \ln(R_{\Lambda}/2\pi) \quad . \quad (1.54)$$

Note that we dropped the prime from Λ' here (as well as for the rest of the thesis) since confusion with the label from S_{Λ} should be excluded. Plugging the ansatz into the flow equation, [eq. \(1.48\)](#), yields

$$(\partial_k \epsilon_k) \phi^2 + \partial_k c_k = 0 \cdot \phi^2 + \frac{1}{2} \frac{\partial_k R_k}{2\epsilon_k + R_k} \quad . \quad (1.55)$$

Hence, $\epsilon_k = \epsilon$ does not get renormalized at all, i.e. it is independent of the flow parameter k . This result is in agreement with the exact computation from [section 1.1](#), [eq. \(1.27\)](#). The explicit integration for c_k is straightforward to perform:

$$c_0 - c_{\Lambda} = \frac{1}{2} \int_{\Lambda}^0 dk \partial_k \ln(2\epsilon + R_k) = \frac{1}{2} \ln \frac{2\epsilon}{2\epsilon + R_{\Lambda}} \rightarrow c_0 = \frac{1}{2} \ln \frac{\pi^{-1}}{\epsilon^{-1} + 2R_{\Lambda}^{-1}} \approx \frac{1}{2} \ln(\epsilon/\pi) \quad , \quad (1.56)$$

taking into account $R_{\Lambda} \rightarrow \infty$. Thus, also the constant term from [eq. \(1.27\)](#) is correctly reproduced—everything appears to be consistent, but: If we would have chosen a truncation different from [eq. \(1.53\)](#), e.g.

$$\tilde{\Gamma}_k(\phi) \stackrel{!}{=} \frac{\lambda_k}{4!} \phi^4 + \epsilon_k \phi^2 + c_k \quad , \quad (1.57)$$

ϵ_k as well as the new coupling λ_k and *higher order couplings* (cf. [eq. \(1.50\)](#))—dropped by hand, here—start to flow:

$$\frac{1}{4!} (\partial_k \lambda_k) \phi^4 + (\partial_k \epsilon_k) \phi^2 + \partial_k c_k = \frac{1}{2} \frac{\partial_k R_k}{\frac{1}{2} \lambda_k \phi^2 + 2\epsilon_k + R_k} = \frac{1}{2} \frac{\partial_k R_k}{2\epsilon_k + R_k} - \frac{\lambda_k}{4} \frac{\partial_k R_k}{(2\epsilon_k + R_k)^2} \phi^2 + (\dots) \phi^4 + \dots \quad (1.58)$$

This observation underlines the (serious) drawback discussed above and to be kept in mind when employing a polynomial truncation. However, increasing powers of $1/R_k$ might *damp* contributions from ϕ^{2n} for increasing $n \in \mathbb{N}$.

The ideas presented here quite naturally extend to multiple (infinitely many) degrees of freedom φ_i . S_Λ , $\tilde{\Gamma}_k$, G_k , \dots turn into multi-dimensional function(al)s, the regulator R_k and $\tilde{\Gamma}_k^{(2)}$ are considered to be (symmetric)²⁰ matrices in the field indices (i, j) , the partition function becomes represented by a path integral $\int \mathcal{D}\varphi \exp(-S_\Lambda[\varphi])$ and the flow equation is written as

$$\dot{\tilde{\Gamma}}_k[\phi] = \frac{1}{2} \text{STr} \left(\tilde{\Gamma}_k^{(2)}[\phi] + R_k \right)^{-1} \dot{R}_k = \frac{1}{2} \text{STr} G_k[\phi] \dot{R}_k \equiv \beta_k[\phi] \quad (1.59)$$

where the STr-operator represents the *super-trace* (cf. section 2.1.2) on the matrix $G_k \dot{R}_k$ over the diagonal (i, i) . Within the field of renormalization β_k is commonly referred to/know as β -*function(al)*. It might be regarded as a *velocity field* as it is the analog to $f(x)$ in the first order differential equation $\dot{y}(t) = f(y, t)$.

In order to discuss *spontaneous breaking of a continuous symmetry*²¹ within the framework of the FRG we need at least two (bosonic) degrees of freedom: denote them by *ochooseanotherbasis* (φ, φ^*) Within this context φ and φ^* are considered as independent degrees of freedom obtained from $\varphi_{1/2}$ by a linear transform other than (φ_1, φ_2) where those degrees of freedom are related by the follow

$$\begin{aligned} \varphi &= (\varphi_1 + i\varphi_2)/\sqrt{2} \\ \varphi^* &= (\varphi_1 - i\varphi_2)/\sqrt{2} \quad \Rightarrow \int \mathcal{D}\varphi \equiv \int d\varphi d\varphi^* = \int d\varphi_1 d\varphi_2 \left| \frac{\partial(\varphi, \varphi^*)}{\partial(\varphi_1, \varphi_2)} \right| = \int d\varphi_1 d\varphi_2 \quad . \end{aligned} \quad (1.60)$$

Let us suppose $S_\Lambda(\varphi_1, \varphi_2)$ and in particular $\tilde{\Gamma}_k(\phi_1, \phi_2)$ depend on

$$\rho \equiv \phi^* \phi = \frac{1}{2}(\phi_1^2 + \phi_2^2) \quad (1.61)$$

only. This statement is intimately linked to the physical constraint of particle number conservation in non-relativistic physics due to Noether's theorem discussed in standard text books on condensed matter field theory^[AS10]. Writing $\phi \equiv \sqrt{\rho} e^{i\alpha}$, the continuous $U(1)$ symmetry of $\tilde{\Gamma}_k$ under $\phi \rightarrow \phi e^{i\alpha}$, i.e.

$$\tilde{\Gamma}_k(\phi_1, \phi_2) \equiv U_k(\rho) \xrightarrow{U(1)\text{-transf.}} U_k(\rho) \quad , \quad (1.62)$$

²⁰To deal with fermionic degrees of freedom represented by Grassmann variables ψ , all matrices involved are actually considered to be *super-matrices*. Section 2.1.2 is concerned about a more detailed description.

²¹We encountered a first example of SSB of a *discrete* symmetry in footnote 13. There, the action S was invariant under *reflection* $\varphi \rightarrow -\varphi$. But the *ground state* $\varphi_0 \neq 0$ —in this context defined as a solution to the EoM $\partial_\varphi S = 0$ —changes sign under the reflection; only the *meta-stable* solution $\varphi_0 = 0$ respects this symmetry: SSB is the general statement that the (physical) equations remain invariant under certain transformation, but their solution (and hence physical observables) does not need to. Spontaneous breaking of a continuous symmetry refers to an invariance under $\varphi \rightarrow g_\alpha(\varphi)$ where g_α specifies some transformation continuously depending on the (set of) parameter(s) α . Such a situation is intimately linked to so called *Goldstone* bosons; one of the historically relevant papers might be that of Goldstone himself^[Gol61]. A reasonable presentation including a collection of references introducing SSB of continuous symmetries and its application in physics provides e.g. [Bur98]. Note, however, that for our zero dimensional toy model SSB is impossible as $k \rightarrow 0$, see e.g. [Mor11], ch. 2.4.

becomes particularly transparent. The quantum EoM of $\Gamma = \tilde{\Gamma}_{k=0}$ specifying the *mean field* (ϕ_0, ϕ_0^*) reduces to

$$\partial_\rho U_{k=0}(\rho_0) \equiv U'_{k=0} = 0 \quad (1.63)$$

which defines a condition on the *condensate* ρ_0 being the (local) extremum of U_k at $k = 0$. However, ρ_0 does not fix the *phase* α and the system's ground state—parametrized by e.g. (ϕ_{10}, ϕ_{20}) , (ϕ_0, ϕ_0^*) or (ρ_0, α_0) —will eventually pick an arbitrary value. Thus, the mean field ϕ_0 (as a solution to the quantum EoM) becomes rotated to $\sqrt{\rho_0} \exp[i(\alpha_0 + \alpha)]$ under $\phi \rightarrow \phi e^{i\alpha}$, i.e. it is not invariant under $U(1)$ transformations; the symmetry becomes spontaneously broken (for $\rho_0 \neq 0$).

An important point we would like to stress next is the fact that eq. (1.59) is a basis independent statement which should yield one unique flow equation whether we choose (ϕ_1, ϕ_2) or (ϕ, ϕ^*) to work with. Indeed, an explicit calculation verifies

$$\begin{array}{ll} \text{---} & \text{---} \\ (\phi_1, \phi_2)\text{---basis} & (\phi, \phi^*)\text{---basis} \\ R_k : & k^2 \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \\ G_k : & \frac{\begin{pmatrix} U''_k \phi_2^2 + U'_k + k^2 & -U''_k \phi_1 \phi_2 \\ -U''_k \phi_1 \phi_2 & U''_k \phi_1^2 + U'_k + k^2 \end{pmatrix}}{(U''_k \phi_1^2 + U'_k + k^2)(U''_k \phi_2^2 + U'_k + k^2) - U''_k{}^2 \phi_1^2 \phi_2^2} \\ \hookrightarrow \dot{U}_k : & 2k \frac{U''_k \rho + U'_k + k^2}{(U''_k \rho + U'_k + k^2)^2 - U''_k{}^2 \rho^2} \end{array} \quad (1.64)$$

Implementing the discretization idea introduced above we are able to directly follow the minimum ρ_{0k} during the flow of $U_k(\rho)$ down to $k = 0$ by reading it off from the interpolation of the flowing grid points $U_k(\rho_n)$.

To demonstrate the basis independence for the polynomial truncation previously discussed²² it is convenient to choose the expansion point ρ_0 to (adaptively) coincide with the extremum ρ_{0k} —(low energy) observables depend on $U_{k=0}^{(n)}(\rho_0)$ where $\rho_0 \neq 0$ within the SSB phase and the case $\rho_0 = 0$ is referred to as *symmetric phase* (SYM). From a pure mathematical point of view a polynomial $p_n(x)$ of *degree* n is uniquely specified by $n + 1$ parameters/coefficients $\{c_i\}$:

$$p_n(x) = \sum_{i=0}^n c_i x^i \quad , \quad (1.65)$$

but by introducing another parameter, the *expansion point* x_0 , we can easily switch to $\{a_i\}|_{x_0}$ with

$$p_n(x) = \sum_{i=0}^n a_i (x - x_0)^i \quad \text{where} \quad c_j = \frac{1}{j!} \sum_{i=j}^n a_i \frac{i!}{(i-j)!} (-x_0)^{i-j} \quad . \quad (1.66)$$

²²Apart from this introductory discussion our computation of the flow equation in chapter 2 incorporates some polynomial ansatz despite our primary aim to exploit the discretization idea from above. The functional $\tilde{\Gamma}_k[\phi]$ defines a complicated object that generally depends on infinitely many variables $\phi(x)$ whose full discretization is out of range for any practical implementation.

In particular for the effective action $U_k(\rho)$, we observe: $a_i = U_k^{(i)}(\rho_{0k})$ and eq. (1.63) (at some finite k) provides the appropriate condition to fix ρ_{0k} . However, the most general (polynomial) expansion of $\tilde{\Gamma}_k[\phi] = U_k(\rho)$ around ρ_{0k} that does not incorporate any symmetry reads

$$\tilde{\Gamma}_k[\phi] = \tilde{\Gamma}_k[\phi_{0k}] + 0 + \frac{1}{2} \sum_{a,b} \left[\partial_{\phi_a \phi_b}^2 \tilde{\Gamma}_{k,ab}[\phi_{0k}] \right] (\phi_a - \phi_{0k,a})(\phi_b - \phi_{0k,b}) + \dots \quad (1.67)$$

Here, the explicit zero marks the vanishing linear term $\sum_a \left[\partial_{\phi_a} \tilde{\Gamma}_k[\phi_{0k}] \right] (\phi_a - \phi_{0k,a})$ due to the extremum condition $\tilde{\Gamma}_k^{(n=1)} \Big|_{\phi_{0k}} \equiv \partial_{\phi_a} \tilde{\Gamma}_k[\phi_{0k}] \stackrel{!}{=} 0$ defining the $\phi_{0k,a}$ s (and hence ρ_{0k}). Indices a, b label the elements of the basis chosen for the (bosonic) degree(s) of freedom—the plural includes the generic case where $\phi \rightarrow \phi(x)$. Note that we will drop the tilde on $\tilde{\Gamma}_k$, its importance we did stress above, from now on since literature on FRG does.

Plugging the generic truncation eq. (1.67) into the lhs. of eq. (1.59) yields the projected flows, eq. (1.50), when extracting the $\Gamma_k^{(n)}$ s by the appropriate projection prescription on the rhs. of eq. (1.59):

$$\dot{\Gamma}_k^{(n)}[\phi_0] = \frac{1}{2} \partial_{\phi}^n \text{STr } G_k \dot{R}_k \Big|_{\phi_0} \quad \text{and in particular} \quad \dot{\Gamma}_{k,ab}^{(2)}[\phi_0] = \frac{1}{2} \partial_{\phi_a \phi_b}^2 \text{STr } G_k[\phi] \dot{R}_k \Big|_{\phi_0} \quad (1.68)$$

is the flow of the *inverse propagator* of the theory *evaluated at* the condensate—the object this thesis aims on resolving by a grid.

However, we would like to pause for a second in order to address an issue that brands eq. (1.68) an approximation. It traces back to the fact that, in general, ϕ_0 explicitly varies when $k \rightarrow 0$. Thus, eq. (1.68) receives an additional contribution proportional to $\partial_t \phi_0 = \dot{\phi}_0$. To grasp the idea we employ a rather abstract/sloppy notation that rewrites the expansion of $\Gamma_k[\phi]$ around a given field configuration ϕ_0 (cf. eq. (1.67)) as

$$\Gamma[\phi] = \Gamma[\phi_0] + \Gamma^{(1)}[\phi_0](\phi - \phi_0) + \frac{1}{2} \Gamma^{(2)}[\phi_0](\phi - \phi_0)(\phi - \phi_0) + \dots \quad , \quad (1.69)$$

where also the scale parameter k has been dropped. More abbreviation is added to this *basis independent*²³ formulation by the definition $\Delta\phi \equiv \phi - \phi_0$. Computing the variation of the effective action with respect to the scale t yields²⁴

$$\dot{\Gamma}[\phi] = \sum_{n=0}^{\infty} \frac{1}{n!} \left[\dot{\Gamma}^{(n)}[\phi_0] - \dot{\phi}_0 \Gamma^{(n+1)}[\phi_0] \right] \underbrace{\Delta\phi \dots \Delta\phi}_{n \text{ times}} = \frac{1}{2} \text{STr } G[\phi] \dot{R} = \beta \quad , \quad (1.70)$$

i.e. switching from one expansion point ϕ_0 to another on different scales adds a corresponding contribution to the projection prescriptions of the flow equation. Indeed, demanding $\Gamma[\phi_0]$ to stay a local extremum

²³ The notion *basis independent* is justified by the observation that under linear transformations $x' = Tx$ the derivative ∂ inversely transforms as $\partial' = T^{-1}\partial$. Our shorthand notation suppresses all (coordinate) indices, here. Hence $\partial_{\phi \dots \phi}^n \Gamma[\phi_0] \underbrace{(\phi - \phi_0) \dots (\phi - \phi_0)}_{n \text{ times}}$ stays invariant.

²⁴ Again, the summation $\sum_{n=0}^{\infty} \frac{1}{n!} (\dots)$ is rather symbolic. For each n there are different contributions $\delta^n \Gamma_k / \delta \phi_i^{n_i} \delta \phi_j^{n_j} \dots$ with $n = n_i + n_j + \dots$.

during the course of the flow, i.e. $\dot{\Gamma}^{(1)}[\phi_0] = 0$, results in

$$\dot{\phi}_0 = \left(\Gamma^{(2)}[\phi_0] \right)^{-1} \dot{\Gamma}^{(1)}[\phi_0] \quad (1.71)$$

and the more accurate result compared to eq. (1.68) reads

$$\dot{\Gamma}_{k,a_1\dots a_n}^{(n)}[\phi_{0k}] - \sum_b \Gamma_{k,a_1\dots a_n b}^{(n+1)}[\phi_{0k}] \dot{\phi}_{0k,b} = \frac{1}{2} \partial_{\phi_{a_1}\dots\phi_{a_n}}^n \text{STr} G_k[\phi] \dot{R}_k \Big|_{\phi_{0k}} = \beta_k^{(n)}[\phi_{0k}] \quad , \quad (1.72)$$

where we reinserted indices a_i and b analogous to a and b from eq. (1.67) as well as the label k for the scale dependence. Transferring this insight to the (frequently used) Taylor expansion/polynomial truncation of the effective potential (cf. section 3.3.1) adds another reason for the advantage of working with a grid instead.

In any case, following the flow of the (local) extremum of (a polynomially truncated) $\Gamma_k[\phi]$ by means of eq. (1.71) restricts the analysis to smoothly follow it. One is unable to detect the emergence of another global extremum/minimum. Hence one is *blind* to first order phase transitions. A truncation based on a grid, as depicted in fig. 1.3, offers the option to explore such discontinuous jumps in the *condensate* ϕ_0 .

To close the discussion, let us explicitly check the basis independence of the projected flow equation for the (inverse) propagator for our toy model $\Gamma_k[\phi] = U_k(\rho)$. On the lhs. of eq. (1.68) we have

$$\dot{\Gamma}_{k,ab}^{(2)}[\phi_0] : \quad \begin{pmatrix} \dot{U}'_k + \phi_{10}^2 \dot{U}''_k & \phi_{10} \phi_{20} \dot{U}''_k \\ \phi_{10} \phi_{20} \dot{U}''_k & \dot{U}'_k + \phi_{20}^2 \dot{U}''_k \end{pmatrix} \quad \begin{pmatrix} \phi_0^{*2} \dot{U}''_k & \dot{U}'_k + \rho_0 \dot{U}''_k \\ \dot{U}'_k + \rho_0 \dot{U}''_k & \phi_0^2 \dot{U}''_k \end{pmatrix} \quad (1.73)$$

Again, the ∂_t -derivative has to be taken with care: Although we demand $U'_k(\rho_0) = 0$ at each k , $\dot{U}'_k(\rho_0) \neq 0$ here—an appropriate equation specifying ρ_0 as an extremum, namely $\dot{\rho}_{0k} = -\dot{U}'_k/U''_k \Big|_{\rho_{0k}}$, compensates for it (cf. section 4.1). To compare the projected flows from the bases (ϕ_1, ϕ_2) and (ϕ, ϕ^*) we might combine the elements $\dot{\Gamma}_{k,ab}^{(2)}$ as follows ($\rho_0 \neq 0, \phi \notin \mathbb{R}$):

$$\begin{array}{ll} (\phi_1, \phi_2)\text{-basis} & (\phi, \phi^*)\text{-basis} \\ \dot{U}'_k(\rho_0) : & \dot{\Gamma}_{k,11}^{(2)}[\phi_0] - \frac{\phi_{10}}{\phi_{20}} \dot{\Gamma}_{k,12}^{(2)}[\phi_0] \qquad \dot{\Gamma}_{k,21}^{(2)}[\phi_0] - \frac{\phi_0^*}{\phi_0} \dot{\Gamma}_{k,22}^{(2)}[\phi_0] \end{array} \quad (1.74)$$

$$\begin{array}{ll} \dot{U}''_k(\rho_0) : & (\phi_{10} \phi_{20})^{-1} \dot{\Gamma}_{k,12}^{(2)}[\phi_0] \qquad \phi_0^{-2} \dot{\Gamma}_{k,22}^{(2)}[\phi_0] \quad . \end{array} \quad (1.75)$$

When applied to the rhs. of the flow equation, a straightforward but algebraically slightly involved computation²⁵ confirms the equivalence of the two projection prescriptions in eqs. (1.74) and (1.75), respectively. Would one have set $\dot{U}'_k = 0$ from the outset, there will be residual terms that prevent identical flow equations if one takes e.g. all $\dot{\Gamma}_{k,ab}^{(2)}[\phi_0]$ from the (ϕ, ϕ^*) -basis or $2\dot{\Gamma}_{k,11}^{(2)}[\phi_0]$ from (ϕ_1, ϕ_2) at $\phi \in \mathbb{R}$ to get an expression for $\dot{U}''_k(\rho_0)$. However, since the prescriptions eqs. (1.74) and (1.75) involve derivatives ∂_{ϕ_a} that act on $G_k[\phi]$ which itself is linked to the second derivative of Γ_k the resulting flow equations for \dot{U}'_k and \dot{U}''_k contain derivatives of U_k up to order 4, i.e. the equations have the form $\dot{y} = \beta(y'', y''', y^{(4)})$. Thus, one needs to project further and further; practically one is enforced to stop

²⁵Some simplification is obtained by choosing the condensate's phase $\alpha = \frac{\pi}{2}$ such that $\phi_0^* = -\phi_0$ and $\phi_{10} = 0$ (assuming finite $\phi_{20} \dot{U}''_k$) for eq. (1.74). Furthermore, the aid of a suitable computer algebra system may help.

this recursion at a certain $\dot{U}_k^{(n_0)}$ assuming that $\dot{U}_k^{(n)} = 0$ for $n > n_0$. All those problems do not arise when dealing with the grid truncation of U_k .

This observation closes our technical remarks on the flow equation. Further details, especially on the treatment of fermionic degrees of freedom, are given e.g. in [MSH⁺11] and [section 2.1.2](#) later on.

On the Road to Non-Relativistic Interacting Fermions

The following discussion intends to derive a set of equations based on the theory of functional renormalization we conceptually introduced in [chapter 1](#). They are tailored to model the non-relativistic quantum field theory of a gas of two species of interacting fermionic particles $\psi_{\uparrow\downarrow}$ coupled to a bosonic field φ through a local (Yukawa-type) interaction. Our ansatz for the effective action Γ_k is focused on a rather generic momentum dependence $P_{k,\varphi/\psi_{\uparrow\downarrow}}(q)$ of the inverse propagator $\Gamma_k^{(2)}$ as well as a generic expression for the effective potential $U_k(\rho) = \Gamma_k|_{\psi_{\uparrow\downarrow}=0, \rho=\phi\phi^*=\text{const.}}$. Both mathematical objects are implemented for numerical studies in [section 3.3](#).

On our way to the final result listed in [appendix D](#) we discuss aspects of the involved math of *super-algebra* in [section 2.1.2](#). Moreover we outline details of the flow equation's graphical representation in [section 2.1.3](#) which becomes extended in the presence of a condensate, i.e. when φ acquires a vacuum expectation value ϕ . A reduced version of the equations derived in [section 2.3](#) become applied in [chapter 4](#) as a first benchmark of the numerics set up in [chapter 3](#).

2.1	Preliminaries	27
2.1.1	Setting the stage	27
2.1.2	An Intermezzo on Super–Matrices	28
2.1.3	On the <i>Truncation</i> and the Graphical Representation of the Flow Equation	33
2.1.4	Momentum Representation of the Effective Averaged Action Γ_k	42
2.1.5	An Important Property of Inverse Propagators	46
2.2	Doing the Math	51
2.2.1	Facing the Full Inverse Propagator $\Gamma_k^{(2)}$	51
2.2.2	The Flowing Effective Potential $U_k(\rho)$	53
2.2.3	Deriving the Flow of the Inverse Propagators	56
2.3	The Story Revised: Flowing Into the Broken Phase	61

2.1 Preliminaries

Actually, before we deduce any formula characterizing the flow of an interacting gas of non-relativistic fermions, we would like to comment on various aspects related to this subject. They are meant to prepare the calculation in [section 2.2](#) and provide some background information as well. We deal with algebraic aspects and introduce an useful graphical representation to visualize (physical) processes that drive the flow. Alongside these facets we comment on the truncation scheme to be used in [section 2.2](#).

2.1.1 Setting the stage

We start with the most general expression of the *flow equation*, namely

$$\partial_k \Gamma_k[\eta] = \frac{1}{2} \text{STr}(\Gamma_k^{(2)}[\eta] + R_k)^{-1} \partial_k R_k = \frac{1}{2} \text{⊙} \quad , \quad (2.1)$$

which is an operator equation for the functional (*quantum*) *averaged effective action* $\Gamma_k = \Gamma_k[\eta]$ depending on some (generalized) quantum field η at certain momentum scale k . Recall that we have dropped—and will drop from now on—the tilde on Γ_k which we used in [chapter 1](#). The symbol $\Gamma_k^{(2)}$ is defined as

$$\Gamma_k^{(2)}[\eta] \equiv \overleftarrow{\frac{\delta}{\delta\eta}} \Gamma_k[\eta] \overrightarrow{\frac{\delta}{\delta\eta}} \quad , \quad (2.2)$$

where we introduced the *field derivative* with respect to η which is either bosonic (φ) or fermionic (ψ). The arrows \Leftrightarrow denote whether $\frac{\delta}{\delta\eta}$ acts to the left- or right-hand side. This becomes important when dealing with fermionic fields¹. R_k is the so called *regulator* that suppresses certain field modes. We met a *primordial* version of this quantity in [section 1.2](#) first; and we will discuss it in much more detail in [section 3.3.3](#). From the derivation of the flow equation one can deduce the properties:

$$\Gamma_k = \begin{cases} \text{the (ordinary) effective action } \Gamma, & \text{for } k = 0 \\ \text{the classical (microscopic) action}^2 S, & \text{for } k \rightarrow \infty \end{cases} \quad . \quad (2.3)$$

The origin of [eq. \(2.3\)](#) is due to the fact that one adds a *mass-like* term

$$\Delta S_k[\eta] \equiv \frac{k^2}{2} \eta^\dagger r_k \eta \quad , \quad \text{and therefore} \quad R_k \equiv \overleftarrow{\frac{\delta}{\delta\eta}} \Delta S_k \overrightarrow{\frac{\delta}{\delta\eta}} = k^2 r_k \quad , \quad (2.4)$$

to $S[\eta]$ and derives the (modified) effective action, labeled by the symbol Γ_k . η^\dagger is the Hermitian conjugate of the complex field η . When we write $\eta = (\eta_{\bar{a}})$ the subscript \bar{a} refers to several fields that represent different *types* of particles with either bosonic or fermionic character³. The *shape function*⁴

¹Sometimes we will just write $\frac{\delta}{\delta\eta}$ instead of $\overleftarrow{\frac{\delta}{\delta\eta}}$ in order not to overload the notation. However, for the general case the *order* of field derivatives in an expression like $\frac{\delta^2}{\delta\eta_2 \delta\eta_1} \equiv \frac{\delta}{\delta\eta_2} \frac{\delta}{\delta\eta_1}$ matters. Here, we mean “Differentiate with respect to η_1 first and to η_2 afterwards.”. For $\frac{\delta^2}{\delta\eta_2 \delta\eta_1} \equiv \overleftarrow{\frac{\delta}{\delta\eta_2}} \overleftarrow{\frac{\delta}{\delta\eta_1}}$ it is the other way around, since the derivative operation acts to the left.

²A precise treatment of the limit $k \rightarrow \infty$ was given back in [section 1.2](#).

³In cases where we write *non-bared* field indices we separately label the different fields and its complex conjugate (or an equivalent basis of the complex field $\eta_{\bar{a}}$). For instance, if we have a theory with one bosonic (ϕ) and one fermionic (ψ) field we write $\eta = (\eta_{\bar{1}}, \eta_{\bar{2}}) = (\eta_1, \eta_2, \eta_3, \eta_4) = (\phi, \phi^*, \psi, \psi^*)$ with $\eta_{\bar{1}} = (\phi, \phi^*)$ or $\eta_{\bar{1}} = (\phi_1, \phi_2)$ and $\eta_{\bar{2}} = (\psi, \psi^*)$.

⁴In the (discrete) space of fields, η can be viewed as a (finite dimensional) vector and accordingly r_k is a matrix.

r_k does not need to be specified at this stage—except that it is not depending on the field η , but may vary with *frequency* and *momentum*⁵. By the construction of the flow equation, r_k is chosen such that it does only *connect* η_a^* to $\eta_{a'}$ with $a = a'$. R_k separately regulates the propagators of the theory, i.e. it is diagonal in the field indices \bar{a} . Notice that $R_k = k^2 r_k$ does not depend on the field η as well. Finally, the *super-trace* STr denotes a trace operation that runs over all (discrete and continuous) field indices introducing an additional minus sign when summing over (discrete) *fermionic* field indices a .

For convenience let us introduce a graphical notation where we abbreviate the (*regulated*) *generalized propagator*⁶

$$G_k[\eta] \equiv (\Gamma_k^{(2)}[\eta] + R_k)^{-1} \equiv \text{=====} \quad (2.5)$$

and the *regulator insertion* $\partial_k R_k \equiv \bullet$. Hence we can write $G_k \partial_k R_k$ as ===== and the *generalized loop-integration* symbolically represented by STr finally yields the graphical expression, eq. (2.1). In particular the graphical representation of the flow, eq. (2.1), provides a useful tool to get intuition on which physical processes contribute to the flow of Γ_k after having defined a suitable *truncation scheme*, to be discussed in a moment.

2.1.2 An Intermezzo on Super-Matrices

Let us pause to develop some machinery that is capable of dealing with *super-matrices*, i.e. we focus on matrices with the formal structure

$$S_I = \left(\begin{array}{c|c} B_1 & F_1 \\ \hline F_2 & B_2 \end{array} \right) \quad \text{or} \quad S_{II} = \left(\begin{array}{c|c} F_1 & B_1 \\ \hline B_2 & F_2 \end{array} \right), \quad (2.6)$$

where $B_{1/2}$ and $F_{1/2}$ are (block-)matrices of ordinary numbers (*c-numbers*) and Grassmann numbers (*anti-commuting c-numbers*, we will call them *a-numbers*), respectively. We plotted dashed lines inside the matrices since we will occasionally use them in the same way for our derivation to explicitly separate the $B_{1/2}$ and $F_{1/2}$ blocks for visual convenience. Assuming that the super-matrix $(S_I)_{ij}$ has size $n \times n$ the horizontal and vertical dashes separate blocks $(B_1)_{1\dots r, 1\dots r}$, $(F_1)_{1\dots r, r+1\dots n}$, $(F_2)_{r+1\dots n, 1\dots r}$ and $(B_2)_{r+1\dots n, r+1\dots n}$ with $0 \leq r \leq n$. The corresponding relations for S_{II} are defined in total analogy.

The *splitting parameter* r allows to treat the four a/c-valued blocks \mathcal{S}_i ($i = 1 \dots 4$) of super-matrices S as individual matrix elements, i.e. it *respects* the usual rules for addition, scalar multiplication and matrix multiplication as follows:

$$\alpha S = \begin{pmatrix} \alpha \mathcal{S}_1 & \alpha \mathcal{S}_2 \\ \alpha \mathcal{S}_3 & \alpha \mathcal{S}_4 \end{pmatrix}, \quad S + \tilde{S} = \begin{pmatrix} \mathcal{S}_1 + \tilde{\mathcal{S}}_1 & \mathcal{S}_2 + \tilde{\mathcal{S}}_2 \\ \mathcal{S}_3 + \tilde{\mathcal{S}}_3 & \mathcal{S}_4 + \tilde{\mathcal{S}}_4 \end{pmatrix} \quad \text{and} \quad S\tilde{S} = \begin{pmatrix} \mathcal{S}_1\tilde{\mathcal{S}}_1 + \mathcal{S}_2\tilde{\mathcal{S}}_3 & \mathcal{S}_1\tilde{\mathcal{S}}_2 + \mathcal{S}_2\tilde{\mathcal{S}}_4 \\ \mathcal{S}_3\tilde{\mathcal{S}}_1 + \mathcal{S}_4\tilde{\mathcal{S}}_3 & \mathcal{S}_3\tilde{\mathcal{S}}_2 + \mathcal{S}_4\tilde{\mathcal{S}}_4 \end{pmatrix}. \quad (2.7)$$

Especially the last equality leads to the important statement that all those operations *preserve* the *type* of the super-matrices, i.e. $S_{I/II} \tilde{S}_{I/II} = \tilde{S}_I$. Furthermore one has $S_{I/II} \tilde{S}_{II/I} = \tilde{S}_{II}$ and $S_{II/I} \tilde{S}_{I/II} = \tilde{S}_{II}$. Another operation from *ordinary* matrix algebra one needs to extend to super-matrices is the trace

⁵By momentum and frequency we mean conjugate variables of space-time due to an appropriate Fourier transform.

⁶In the following we will use $\text{---}\xrightarrow{\psi}\text{---}$ for fermionic and $\text{---}\xrightarrow{\varphi}\text{---}$ for bosonic field propagators and for notational convenience we will sometimes neglect the k label that just indicates the energy/momentum scale dependence of the (general) propagator.

Tr with the property $\text{Tr } B_1 B_2 = \text{Tr } B_2 B_1$ with $(B)_{ij} \in \mathbb{C}$. In contrast, matrices F with a-numbers as elements $(F)_{ij}$ exhibit $\text{Tr } F_1 F_2 = -\text{Tr } F_2 F_1$ since $(F_1)_{ij}(F_2)_{ji} = -(F_2)_{ij}(F_1)_{ji}$. In total analogy we obtain $\text{Tr } BF = \text{Tr } FB$.

It is appropriate to define the (discrete) super-trace sTr according to

$$\text{sTr } S_I \equiv \text{Tr } B_1 - \text{Tr } B_2 \quad . \quad (2.8)$$

We can use the last relation of eq. (2.7) in conjunction with eq. (2.6) to explicitly verify the stated commutation property, i.e.

$$\begin{aligned} \text{sTr } S_I \tilde{S}_I &= \text{Tr}(B_1 \tilde{B}_1 + F_1 \tilde{F}_2 - F_2 \tilde{F}_1 - B_2 \tilde{B}_2) \\ &= \text{Tr}(\tilde{B}_1 B_1 - \tilde{B}_2 B_2) - \text{Tr}(\tilde{F}_2 F_1 - \tilde{F}_1 F_2) = \text{sTr } \tilde{S}_I S_I \quad . \end{aligned} \quad (2.9)$$

If we commute S_{II} -matrices under the super-trace an extra overall minus sign is produced:

$$\text{sTr } S_{II} \tilde{S}_{II} = -\text{sTr } \tilde{S}_{II} S_{II} \quad . \quad (2.10)$$

The commutation of a *mixed* super-matrix product $S_I S_{II}$ under sTr is a bit more involved. Introducing some notational convention helps to handle this case and it will be of practical importance for concrete computations as well. Let us define

$${}^-S_I \equiv \begin{pmatrix} B_1 & -F_1 \\ -F_2 & B_2 \end{pmatrix} \quad \text{and} \quad {}^-S_{II} \equiv \begin{pmatrix} -F_1 & B_1 \\ B_2 & -F_2 \end{pmatrix} \quad , \quad (2.11)$$

i.e. we negate all $(S_{I/II})_{ij}$ -elements which are a-numbers. Obviously ${}^-S$ is of the same type as S and we have

$${}^{--}S_{I/II} \equiv {}^+S_{I/II} = S_{I/II} \quad . \quad (2.12)$$

Hence we can write

$$\text{sTr } S_I \tilde{S}_{II} = \text{sTr } \tilde{S}_{II} {}^-S_I \quad , \quad (2.13)$$

which is the analogue to eqs. (2.9) and (2.10) for mixed super-matrix products.

Another operation with super-matrices S of interest to us is *inversion*, i.e. finding matrices such that we naturally define

$$1 \equiv \begin{pmatrix} 1 & | & 0 \\ \hline - & | & - \\ 0 & | & 1 \end{pmatrix} \stackrel{!}{=} S_L^{-1} S \stackrel{!}{=} S S_R^{-1} \quad , \quad (2.14)$$

where we distinguish between the *left-* and *right-inverse* $S_{L,R}^{-1}$ for a moment. But as obvious from matrix multiplication we have $1S = S1 = S$ and by evaluating $S_L^{-1} S S_R^{-1}$ using the associative property of (super-)matrix multiplication it turns out that

$$S_L^{-1} = S_R^{-1} \equiv S^{-1} \quad . \quad (2.15)$$

Furthermore, we immediately check that the inversion of S yields a matrix of the same type as S , e.g.

$$S_I S^{-1} = \begin{pmatrix} B_1 & F_1 \\ F_2 & B_2 \end{pmatrix} \begin{pmatrix} S_1^{-1} & S_2^{-1} \\ S_3^{-1} & S_4^{-1} \end{pmatrix} = \begin{pmatrix} B_1 S_1^{-1} + F_1 S_3^{-1} & \dots \\ \dots & F_2 S_2^{-1} + B_2 S_4^{-1} \end{pmatrix} \stackrel{!}{=} 1 \quad . \quad (2.16)$$

From the off-diagonal blocks 0 of the unit-matrix we are unable to deduce information on whether the \mathcal{S}_i^{-1} are matrices with elements that are a- or c-numbers. On the other hand the blocks 1 are ordinary matrices B and therefore the products $B_1\mathcal{S}_1^{-1}$, $F_1\mathcal{S}_3^{-1}$, $F_2\mathcal{S}_2^{-1}$ and $B_2\mathcal{S}_4^{-1}$ need to be ordinary matrices as well. Since $(B\tilde{B})_{ij}$ and $(F\tilde{F})_{ij}$ are c-numbers and $(BF)_{ij}$ are a-numbers we finally have $S^{-1} = S_I^{-1}$. Similarly we argue for $S^{-1} = S_{II}^{-1}$ in the case of $S = S_{II}$. It is also straightforward to check that

$$^{-}S^{-1}(S^{-1}) = 1 \quad , \quad (2.17)$$

i.e. the inverse of ^{-}S is just the inverse of S with a negation to all elements $(S^{-1})_{ij}$ which are a-numbers. A last ingredient necessary to derive the flow equations of a theory that involves fermions and bosons concerns (functional) derivatives $\frac{\delta}{\delta\eta}$; or more precisely *ordinary/bosonic* derivatives $\frac{\delta}{\delta\varphi}$ and *fermionic* derivatives $\frac{\delta}{\delta\psi}$ with respect to a-numbers. To be consistent with $\psi_1\psi_2 = -\psi_2\psi_1$ we have to require $\frac{\delta}{\delta\psi_1}\psi_2 = -\psi_2\frac{\delta}{\delta\psi_1}$, i.e. the derivative $\frac{\delta}{\delta\psi}$ anti-commutes with a-numbers and similarly we convince ourself that it commutes with c-numbers. Therefore we obtain

$$\frac{\delta}{\delta\varphi}S = S\frac{\delta}{\delta\varphi} \quad \text{and} \quad \frac{\delta}{\delta\psi}S = ^{-}S\frac{\delta}{\delta\psi} \quad , \quad (2.18)$$

if the super-matrix S of either type I or II is independent of φ and ψ , respectively. In the same fashion we can write the *product rules* as

$$\frac{\delta}{\delta\varphi}S\tilde{S} = \left(\frac{\delta}{\delta\varphi}S\right)\tilde{S} + S\frac{\delta}{\delta\varphi}\tilde{S} \quad \text{and} \quad \frac{\delta}{\delta\psi}S\tilde{S} = \left(\frac{\delta}{\delta\psi}S\right)\tilde{S} + ^{-}S\frac{\delta}{\delta\psi}\tilde{S} \quad (2.19)$$

in the case where the super-matrices S and \tilde{S} are ϕ - and ψ -dependent. Moreover we recognize that fermionic derivatives change the type of super-matrices:

$$\frac{\delta}{\delta\varphi}S_{I/II} = \tilde{S}_{I/II} \quad \text{and} \quad \frac{\delta}{\delta\psi}S_{I/II} = \tilde{S}_{II/I} \quad . \quad (2.20)$$

This observation is important if one evaluates higher-order fermionic derivatives and a specific example will demonstrate that we have to extend our notation from eq. (2.11). Let us compute

$$\frac{\delta^3}{\delta\psi\delta\psi\delta\psi}S_I(\psi)\tilde{S}_I(\psi) = \frac{\delta^2}{\delta\psi\delta\psi} \left[S_{II}^{(1)}\tilde{S}_I + ^{-}S_I\tilde{S}_{II}^{(1)} \right] \quad , \quad (2.21)$$

where we introduced the shorthand

$$S^{(n)} \equiv \underbrace{\frac{\delta^n}{\delta\psi\delta\psi\dots}}_{n \text{ times}} S(\psi) \quad (2.22)$$

which hides the order of the anti-commuting derivatives $\delta/\delta\psi$. Moreover, we dropped indices a labeling different field variables ψ_a . To this end, the notation of eqs. (2.24) and (2.25) becomes ambiguous⁷. We accept this drawback in order to avoid diving into a sea of notation that masks the conceptual

⁷Concerning the additional minus sign arising from $\delta^2/\delta\psi_a\delta\psi_b = -\delta^2/\delta\psi_b\delta\psi_a$: We do not modify the order of $\delta^3/\delta\psi_a\delta\psi_b\delta\psi_c$ when distributing the derivatives by means of the product rule in the computation of eq. (2.21), continued by eqs. (2.24) and (2.25).

idea for introducing eq. (2.23). The Roman subscript in eq. (2.21) explicitly indicates the type of the *derivative of the super-matrix* denoted by $\tilde{S}^{(n)}$; it is not the type of \tilde{S} itself! Now, if we perform the next fermionic derivative in eq. (2.21), we encounter a notational problem when acting on the second summand of eq. (2.21): If $\frac{\delta}{\delta\psi}$ is applied to ${}^{-}S_I$ its type is modified and one has to indicate that the left superscript '–' refers to type I super-matrices although $\frac{\delta}{\delta\psi}{}^{-}S_I$ is of type II . We do so by extending the left upper label of the super-matrix symbol, eq. (2.11), i.e.

$${}^{-II}S_I \equiv \begin{pmatrix} -B_1 & F_1 \\ F_2 & -B_2 \end{pmatrix} \quad \text{and} \quad {}^{-I}S_{II} \equiv \begin{pmatrix} F_1 & -B_1 \\ -B_2 & F_2 \end{pmatrix} . \quad (2.23)$$

Therefore we proceed the evaluation of eq. (2.21) with

$$\frac{\delta}{\delta\psi} \left[S_I^{(2)} \tilde{S}_I + {}^{-}S_{II}^{(1)} \tilde{S}_{II}^{(1)} + {}^{-I}S_{II}^{(1)} \tilde{S}_{II}^{(1)} + {}^{+}S_I \tilde{S}_I^{(2)} \right] . \quad (2.24)$$

We finally obtain

$$\frac{\delta^3 S_I \tilde{S}_I}{\delta\psi \delta\psi \delta\psi} = S^{(3)} \tilde{S} + {}^{-}S^{(2)} \tilde{S}^{(1)} + {}^{-II}S^{(2)} \tilde{S}^{(1)} + S^{(1)} \tilde{S}^{(2)} + {}^{-}S^{(2)} \tilde{S}^{(1)} - S^{(1)} \tilde{S}^{(2)} + S^{(1)} \tilde{S}^{(2)} + {}^{-}S \tilde{S}^{(3)} \quad (2.25)$$

where we neglect the subscript since it can be deduced from $S = S_I$ and $\tilde{S} = \tilde{S}_I$ in combination with the number n of fermionic derivatives $\delta/\delta\psi$ acting on the super-matrix. We are not allowed to further simplify the expression, due to the ambiguous notation from eq. (2.22). Note, that the result, eq. (2.25), differs from the bosonic analogon by a couple of minus signs, i.e.

$$\delta^3 S_I \tilde{S}_I / \delta\varphi \delta\varphi \delta\varphi = S^{(3)} \tilde{S} + 3 \times S^{(2)} \tilde{S}^{(1)} + 3 \times S^{(1)} \tilde{S}^{(2)} + S \tilde{S}^{(3)} . \quad (2.26)$$

Here, $3 \times S^{(i)} \tilde{S}^{(j)}$ with $i = 1, 2$ and $j = 2, 1$ ($i + j = 3$) denotes all three combinations of distributing the three derivatives $\delta/\delta\varphi$ on S and \tilde{S} . Anyway, it merely was our intention to demonstrate the need for the left upper label defined by eqs. (2.11) and (2.23). The ambiguous notation, eq. (2.22), will not be used any more. In particular, symbols as $\delta^n \Gamma_k / \delta\eta \delta\eta \dots \equiv \Gamma_k^{(n)}$ refer to matrices in the indices a, b, c, \dots of the functional derivatives $\delta/\delta\eta_a, \delta/\delta\eta_b, \delta/\delta\eta_c, \dots$

Our consideration up to this point allow for a useful extension of an identity that connects the derivative of a matrix with the derivative of its inverse to the super-matrix formalism. Consider

$$\frac{\delta}{\delta\psi} S S^{-1} = \left(\frac{\delta S}{\delta\psi} \right) S^{-1} + {}^{-}S \frac{\delta}{\delta\psi} S^{-1} = \frac{\delta}{\delta\psi} 1 = 0 \quad \Rightarrow \quad \frac{\delta S}{\delta\psi} = - {}^{-}S \left(\frac{\delta}{\delta\psi} S^{-1} \right) S \quad (2.27)$$

and together with eq. (2.17) we sum up to have

$$\frac{\delta}{\delta\psi} \pm S = - \mp S \left[\frac{\delta}{\delta\psi} \pm (S^{-1}) \right] \pm S , \quad (2.28)$$

which differs from the ordinary/bosonic derivative by

$$\frac{\delta}{\delta\varphi} S = -S \left[\frac{\delta}{\delta\varphi} S^{-1} \right] S . \quad (2.29)$$

However, recognizing that $\Gamma_k[\eta]$ is a real valued functional of bosonic and fermionic fields η_a , we observe that $\Gamma_k^{(2)}[\eta]$ is a (finite dimensional) matrix in the (discrete) field indices a with elements which are either ordinary complex numbers or Grassmann valued. Arranging the discrete field index $a = 1 \dots n$ such that all η_a with $1 \leq a \leq r$ are bosonic and the remaining η_a represent fermionic fields (cf. eq. (2.6) and the text to follow), we conclude that $\Gamma_k^{(2)}$ is of the same type as the super-matrices S_I above. Then, from eq. (2.20), it is straightforward to read off the corresponding type of an arbitrary vertex $\Gamma_k^{(n)}$. Therefore we will use the developed notational tools to deduce some useful identities for practical computations. From eq. (2.29) we immediately get

$$\frac{\delta}{\delta\psi} \pm G_k = - \mp G_k \left(\frac{\delta}{\delta\psi} \pm \Gamma_k^{(2)} \right) \pm G_k \quad (2.30)$$

as well as an analogous relation for the bosonic derivative. Another identity concerns rewriting the rhs. of the flow equation in a convenient way for practical calculations. We define the symbol $\tilde{\partial}_k$ such that it acts as a derivative on the regulator R_k only. Hence, due to eq. (2.9)—and without mathematical rigor—we can write

$$\begin{aligned} \tilde{\partial}_k \text{sTr} \ln G_k^{-1} &= \text{sTr} \sum_{n=1}^{\infty} \frac{(-)^{n-1}}{n} \tilde{\partial}_k (G_k^{-1} - 1)^n = \text{sTr} \left[\tilde{\partial}_k G_k^{-1} \right] \sum_{n=0}^{\infty} (-)^n (G_k^{-1} - 1)^n \\ &= \text{sTr} (\partial_k R_k) G_k = \text{sTr} G_k \partial_k R_k \quad , \end{aligned} \quad (2.31)$$

which is exactly twice the rhs. of the flow equation. We obtain the last and most important relation for practical purposes if we consider to take the fermionic derivative of eq. (2.31). In addition to the rules we used for the $\tilde{\partial}_k$ -derivative acting on $\text{sTr} \ln G$ a moment before we also have to respect eq. (2.18) in addition. To this end, let us consider $\frac{\delta}{\delta\psi} \text{sTr} S_I^n$ which becomes

$$\text{sTr} \frac{\delta S_I^n}{\delta\psi} = \text{sTr} \sum_{m=0}^{n-1} -S_I^m \underbrace{\frac{\delta S_I}{\delta\psi}}_{\equiv \tilde{S}_{II}} S_I^{n-1-m} \quad . \quad (2.32)$$

and by using eq. (2.13) for the summands ($p \equiv n - 1 - m$) we obtain

$$\text{sTr} -S_I \underbrace{-S_I^{m-1} \tilde{S}_{II} S_I^p}_{\tilde{S}_{II}} = \text{sTr} -S_I^{m-1} \tilde{S}_{II} S_I^{p+1} = \dots = \text{sTr} \frac{\delta S_I}{\delta\psi} S_I^{n-1} = \text{sTr} -S_I^{n-1} \frac{\delta S_I}{\delta\psi} \quad , \quad (2.33)$$

which allows us to write

$$\text{sTr} \frac{\delta S_I^n}{\delta\psi} = \text{sTr} \left[\frac{\delta S_I}{\delta\psi} \right] n S_I^{n-1} \quad (2.34)$$

and therefore the final identity reads

$$\text{sTr} \frac{\delta}{\delta\eta} \ln G_k^{-1} = \text{sTr} \frac{\delta \Gamma_k^{(2)}}{\delta\eta} G_k = \text{sTr} -G_k \frac{\delta \Gamma_k^{(2)}}{\delta\eta} \quad . \quad (2.35)$$

where we set the former S_I to $G_k^{-1} - 1$ (since $\Gamma_k^{(2)}$ is an appropriate type-I super-matrix); and we repeated algebraic steps similar to the computation which led to eq. (2.31).

2.1.3 On the *Truncation* and the Graphical Representation of the Flow Equation

Let us now introduce the physical picture for the QFT model of the BCS–BEC crossover: We consider an ultra–cold gas of interacting spin-up/spin-down fermions $\psi_{\uparrow,\downarrow}$ in the *balanced case*, i.e. the *chemical potential* μ is equal for both types of fermions: $\mu_{\downarrow} = \mu_{\uparrow}$. For the fermionic interaction we assume it to be point-like/*local*, since we are interested in the low energy physics of the problem that does not resolve details of the interaction potential. It is rendered by a single parameter, the *scattering length*^[KZ08] a . Since for positive a the fermions can form a bosonic bound state φ , we consider the following ψ - φ *Yukawa-type* interaction:

$$h_{\varphi}\varphi^{\dagger}\psi_{\downarrow}\psi_{\uparrow} = \begin{array}{c} \psi_{\downarrow} \\ \swarrow \\ \bullet \\ \nearrow \\ \psi_{\uparrow} \end{array} \text{---} \varphi \quad . \quad (2.36)$$

However, in the language of functional renormalization φ is just considered as a bosonic degree of freedom composed from two fermionic ones. In particular for $a \rightarrow -\infty$ we might identify φ with Cooper pairs. Mathematically this is justified by the so-called *Hubbard–Stratonovich*^[Hub59,Str57,AS10,Kle11] transformation that is applied to decouple a local 4–fermionic interaction $\psi_{\downarrow}^{\dagger}\psi_{\uparrow}^{\dagger}\psi_{\downarrow}\psi_{\uparrow}$ by an auxiliary (static) bosonic field φ . On the other hand we want to allow for dynamics of the bosonic field φ due to the physical picture in which it represents a bound/composite state of two fermionic degrees of freedom; to this end the model goes beyond a pure Hubbard–Stratonovich transformation. Thus, the (renormalizable) dynamics/kinetic term of the bosonic φ –field allows to capture *non-local* features of the macroscopic physics of the (pure) fermionic theory, e.g. by the process:

$$\begin{array}{c} \psi_{\downarrow} \\ \swarrow \\ \bullet \\ \nearrow \\ \psi_{\uparrow} \end{array} \text{---} \varphi \text{---} \begin{array}{c} \psi_{\downarrow} \\ \swarrow \\ \bullet \\ \nearrow \\ \psi_{\uparrow} \end{array} \quad . \quad (2.37)$$

In addition we allow for (point-like) interactions of the (dynamical) bosons, meaning that we assume an *effective potential*

$$U[\rho] = m^2\rho + \frac{\lambda}{2}\rho^2 + \dots = \text{---} \xrightarrow{m^2} \text{---} + \begin{array}{c} \text{---} \\ \swarrow \\ \bullet \\ \nearrow \\ \text{---} \end{array} \lambda + \dots \quad \text{with } \rho \equiv \varphi^{\dagger}\varphi \quad (2.38)$$

that generates higher order, particle number conserving interactions⁸.

A suitable *truncation* of Γ_k is guided by the underlying physics we described above and therefore we

⁸Formally, this fact manifests itself due to the invariance of $U(\rho)$ under global phase rotations $\varphi \rightarrow e^{i\alpha}\varphi$, where $\rho = \varphi^{\dagger}\varphi \rightarrow \rho$. Since Noether’s theorem ensures a conservation law to each continuous symmetry one obtains the continuity equation $\partial_t\rho = -\nabla \cdot \mathbf{j}$.

restrict the averaged effective action to

$$\Gamma_k \stackrel{!}{=} \sum_{\sigma=\uparrow,\downarrow} \psi_\sigma^* P_\psi^k \psi_\sigma + \varphi^* P_\varphi^k \varphi + h_\varphi \varphi^* \psi_\uparrow \psi_\downarrow + U_k[\rho] + c.c. \quad , \quad (2.39)$$

where P_η^k denotes the *inverse propagator*⁹ and *c.c.* ensures $\Gamma_k^* = \Gamma_k$ as appropriate for a quantum action to describe a physical system¹⁰. While the φ are ordinary complex numbers to represent the bosonic degrees of freedom in the field integral formulation of QFT, the ψ have to be (anti-commuting) Grassmann variables to implement the *Pauli exclusion principle* with associated *anti-commutation relations* for fermions, cf. eq. (1.14) with a_i substituted by $\psi_{\uparrow\downarrow}$. The sub-/superscript k indicates if the corresponding operators/observables O contribute to the flow, i.e. if $\partial_k O \neq 0$. For our choice of Γ_k , eq. (2.39), the (real valued) coupling h_φ does not flow at all ($h_\varphi^k = h_\varphi$). One can use the introduced graphical representation in eq. (2.1) to deduce this statement for the so called *symmetric phase* (SYM), see our discussion below. However, when it comes to calculate the physics in the *spontaneously symmetry broken phase* (SSB) there exist diagrams that contribute to the flow of the Yukawa coupling. We extend the graphical notation from this paragraph in section 2.3 and the corresponding flow equation for h_φ^k is given by eq. (2.193). In order to proceed with eq. (2.39) it is necessary to specify an appropriate basis. This step turns the abstract notation, in particular eq. (2.1), into a form which allows for concrete computation. Section 2.1.4 is dedicated to this topic.

Let us spend a minute to comment on truncation schemes in general to absorb our ansatz, eq. (2.39). Clearly, the choice of the truncation will have (probably serious) impact on the results obtained from solving the flow equation. A sensible truncation is expected to capture all relevant physical effects such that observables O_k have to correctly reflect the physics under investigation. As we tried to point out in section 1.2 picking a suitable truncation scheme is far from trivial and one needs to skillfully balance computational effort versus a treatment close to the underlying physics. Especially the latter aspect turns out to be the crux of the matter: Theoretically, the flow drives Γ_k to the *full* quantum action Γ including the entire *macro*-physics of the problem in question. However, it is just impossible for a given truncation to capture the most general Γ one can think of. Thus one needs to restrict to an ansatz for Γ_k such that it incorporates *the relevant degrees of freedom*.

From our vague formulation it should become obvious that there is no straightforward/unique way to establish the idea of functional renormalization in practice. To our knowledge there is no cookbook available that allows for blindly setting up and applying an *optimal* truncation. Moreover, there is the free choice of the regulator function R_k which affects the course of the flow. Yet, there exists an idea to define an *optimization* criterion^[Lit01b,Lit01a,Lit01c] for R_k . We discuss several options to define an appropriate regulator in section 3.3.3.

Due to the ambiguous process of specifying a truncation tailored to fit the physical problem to be

⁹By definition P_η is diagonal in the basis of distinct fields (η_a^*, η_a) .

¹⁰In the field integral formulation of quantum many-particle physics this traces back to require physical observables to be Hermitian, especially the Hamiltonian should satisfy $H^\dagger = H$. In particular the inverse propagators P_η^k correspond to the *free* field theory, i.e. a setup without interaction between different species of particles represented by (η_a^*, η_a) . Therefore $H^\dagger = H$ translates to $P_\eta^\dagger = P_\eta$.

solved, there are different schemes on the market. Three of them widely used in the literature are briefly discussed in [BTW00]. Irrespective of the ambiguity attached to the notion *truncation* there should be a systematic prescription to *improve* it. Typically it consists of including increasingly many parameters to let flow. It is advisable to let these extensions respect the symmetries imposed on Γ by the physical setup. Typically, the inclusion follows a given philosophy, e.g. it successively adds polynomials of fields $\eta, \eta\eta, \eta\eta\eta, \dots$ of increasing degree to Γ_k where the corresponding prefactors become flowing quantities. For us we would like to adopt the phrase *from local to global interaction*: On the microscopic level one might start with a purely fermionic theory where the spin-up/spin-down fermions $\psi_{\uparrow\downarrow}$ interact by a local (*contact*) interaction $\sim \lambda\psi_{\uparrow}^{\dagger}\psi_{\downarrow}^{\dagger}\psi_{\uparrow}\psi_{\downarrow}$. The Hubbard–Stratonovich transformation introduces an additional bosonic field φ . By adding dynamics in the form of P_{φ}^k to φ , the local interaction parameter λ is substituted by a non-local one. Systematically improving φ 's dispersion/momentum-dependence (cf. section 2.1.4) through a finer resolution of P_{φ}^k by an (adaptive) grid (cf. section 3.3.2) captures this non-local character increasingly well. Since $U[\rho]$ specifies (particle number conserving) local interactions of the *interaction mediating* field φ , its discretization in terms of increasingly many grid points (cf. section 3.3.1) contributes to an accurate description of the non-local nature of the interaction between fermions. A momentum-dependent Yukawa-type coupling h_{φ} would add even more *non-locality*, but such a step is beyond the scope of our approach within this thesis. We would turn to a never-ending story when it comes to list all contributions allowed by non-relativistic symmetries as e.g. particle number conservation.

However, given the interpretation of φ as the bosonic field composed of ψ_{\uparrow} and ψ_{\downarrow} , the most simplest extension to non-locality with respect to the fermionic interaction is given by a correct description of φ 's propagator $P_{\varphi}^k{}^{-1}$. In a next step one might add local interactions of φ encapsulated in the (real-valued) function $U_k(\rho)$. This constitutes the logic we preferred to phrase by *from local to global* above. Within the approximation specified a moment ago the systematic classification of improving our truncation consists in successively enhancing the resolution of P_{φ}^k and $U_k(\rho)$. From this perspective, the resolution of P_{ψ} in momentum space in a way analogous to P_{φ} does not directly follow the line. It might be regarded as a byproduct. Nevertheless it is of physical relevance to study the fermionic particle's dispersion. Our approach outlined in order to set up a reasonable truncation is also based on the insight of previous studies which invested substantial effort to quantitatively improve physical observables in the BCS-BEC crossover. In terms of qualitative success there was only partial success. A summary provides review [DFG⁺10].

Now, let us introduce the concept of *projecting* the flow on a certain operator O which appears in the truncated version of Γ_k . Fixing the truncation of Γ_k is in some sense at the heart of the projection procedure from a totally unspecified averaged effective action Γ_k whose flow is determined by the most general eq. (2.1). Starting with some (known) Γ_{Λ} at energy scale $k = \Lambda$, eq. (2.1) will in general *produce* additional couplings when flowing to different energy regimes:

$$\Gamma_{\Lambda+\delta k}[\eta] = \Gamma_{\Lambda}[\eta] + \frac{\delta k}{2}\Delta\Gamma_{\Lambda}[\eta] + \mathcal{O}(\delta k^2) \quad \text{with} \quad \Delta\Gamma_{\Lambda}[\eta] \equiv \text{STr}(\Gamma_{\Lambda}^{(2)}[\eta] + R_k)^{-1}\partial_k R_k \quad , \quad (2.40)$$

where $\Delta\Gamma_{\Lambda}[\eta]$, as a functional of the field η , will in general not assume the same functional form as

$\Gamma_\Lambda[\eta]$. But that is implicitly assumed when choosing a specific truncation Γ_k for the whole energy interval $k \in [0, \Lambda]$. To this end, we systematically neglect certain contributions to the (final) effective action Γ when starting with some known theory of the (microscopic) physics at length scale Λ^{-1} , i.e. formally defining¹¹ $\Gamma_\Lambda = S$, and integrating eq. eq. (2.1) down to $k = 0$. All information on the difference between S and Γ is projected to (generalized couplings) that become *renormalized* under the flow. In our specific case, eq. eq. (2.39), it is the operators P_φ^k and P_ψ^k as well as the scalars m_k^2 , λ_k , ... and $h_\varphi^k = h_\varphi$.

Therefore a “wise choice” for an ansatz of Γ_k refers to a projection of the most general case to a truncated version of the averaged effective action that is sensitive to the physics especially at *large* length scales $k^{-1} \gg \Lambda^{-1}$. On the one hand suitably defining Γ_k is a hard task that needs sufficient physical intuition, but on the other hand if one compares experimentally obtained quantities to the results derived from Γ , it offers the opportunity to extract the *essential physics* by adjusting the truncation as long as one reaches sufficient accordance between theory and experiment.

However, back to the graphical representation of eq. eq. (2.1), we would like to write it in the *discrete* basis of fields $\eta = (\eta_a)$ and therefore $\Gamma_k^{(2)}$ acquires two *field indices*:

$$\Gamma_{k,ab}^{(2)}[\eta] \equiv \frac{\overleftarrow{\delta}}{\delta\eta_a} \Gamma_k[\eta] \frac{\overleftarrow{\delta}}{\delta\eta_b} \quad . \quad (2.41)$$

Furthermore, by *regulating* $\Gamma_k^{(2)}$ with R_k and inverting it, the (generalized) propagator $G_k = G_{k,ab}$ is again a matrix in the field indices expression to the right of STr on the rhs. of eq. (2.1) reads $\sum_b G_{k,ab}(\partial_k R_k)_{bc}$. Adopting the convention to write $a \equiv B$ for bosonic and $a \equiv F$ for fermionic indices we have

$$\text{STr} G_k(\partial_k R_k) = \sum_{a,B} \text{tr} G_{k,Ba}(\partial_k R_k)_{aB} - \sum_{a,F} \text{tr} G_{k,Fa}(\partial_k R_k)_{aF} \quad (2.42)$$

with tr symbolically denoting the trace/integration over all remaining (continuous) indices not specified yet¹². In addition to the introduced summation symbols STr and tr we will use Tr , referring to the trace over the space of discrete indices a excluding the additional minus which STr assigns to fermionic indices $a = F$. Tr_B and Tr_F are restricted to bosonic and fermionic (discrete) indices, respectively. With these definitions at hand we are allowed to write the condensed expression

$$\text{STr} = \text{tr} \left[\text{Tr}_B - \text{Tr}_F \right] \quad . \quad (2.43)$$

As stated above, the regulator $R_{k,ab}$ is diagonal with respect to field indices by construction (and thus $\partial_k R_k$ is) and therefore eq. (2.42) simplifies to

$$\text{tr} G_{B_i B'_i}(\partial_k R_k)_{B'_i B_i} - \text{tr} G_{F_i F'_i}(\partial_k R_k)_{F'_i F_i} = \text{diagram} + \dots - \text{diagram} - \dots \quad , \quad (2.44)$$

¹¹According to eq. (2.3) one should know the *fundamental physics*, i.e. the action S in the limit $\Lambda \rightarrow \infty$. On the other hand one is allowed to regard S as some *effective theory* where all fluctuations of an underlying (fundamental) theory like e.g. QCD, string theory etc. on length scales $k^{-1} < \Lambda^{-1}$ are properly integrated out by the flow eq. (2.1).

¹²For the equations to come we declare to implicitly sum over all possible values of repeating discrete (field-)indices a , which adapts some sort of *Einstein summation rule* and is in use to reduce notation. Nevertheless, in cases where the \sum -symbol helps to keep track on the indices to sum over we will restore it.

where B_i and F_i are the (two) bosonic and fermionic field indices referring to one specific field/particle φ_i and ψ_i , respectively. In addition we presented the graphical notation one is used to in this language. Remembering the graphical convention it becomes now obvious why $\partial_k \Gamma_k$ is determined by a *one-loop structure*: Take the propagator $G_{ab} = \equiv$, couple it to $(\partial_k R_k)_{ab} = \bullet$ by $G_{ab}(\partial_k R_k)_{bc} = \equiv \bullet$ and *built up the loop* by coupling it back to the propagator $G_{k,ab}(\partial_k R_k)_{ba} = \bigcirc$. As a next step, we can derive an expression for the functional derivative of the propagator G with respect to a fixed field η_i . With eq. (2.31) in mind it is straightforward to conclude

$$\frac{\delta}{\delta \eta_i} G[\eta]_{ab}(\partial_k R_k)_{ba} = - \zeta_i G_{ac} \left[\frac{\delta}{\delta \eta_i} \Gamma_{k,cd}^{(2)} \right] G_{db}(\partial_k R_k)_{ba} = - \eta_i \Gamma_k^{(3)} \bigcirc, \quad (2.45)$$

where we used the (common) definition $\zeta_i = \pm 1$ for a bosonic and fermionic index i , respectively. It is important to note that one can not read off its impact on the propagators Γ_k in the graphical representation used here and therefore, the developed diagrammatic approach should serve just as a convenient way to represent the flow equation — unless one does not include a suitable additional notation, which will not be necessary in our case.

Again, one can represent the expression with an appropriate diagram, defining a new element, the *vertex* (matrix)

$$\Gamma_{k,ab}^{(n)} \equiv \frac{\delta^{n-2}}{\delta \eta_{n-2} \dots \delta \eta_2 \delta \eta_1} \Gamma_{k,ab}^{(2)}[\eta] \equiv : \begin{array}{c} \eta_1 \quad \eta_b \\ \diagdown \quad \diagup \\ \Gamma_k^{(n)} \\ \diagup \quad \diagdown \\ \eta_a \\ \dots \\ \eta_{n-2} \end{array} \quad (n > 2) \quad , \quad (2.46)$$

with $n - 2$ external legs labeled by the fields η_i ($i = 1 \dots n - 1$). The one-loop structure of higher order derivatives of $G \partial_k R_k$ is similarly obtained by successively applying eqs. (2.18) and (2.30): E.g. the 3rd order term reads

$$\begin{aligned} \frac{\delta^3}{\delta \eta_3 \delta \eta_2 \delta \eta_1} \bigcirc &= - \bigcirc + 2 \begin{array}{c} \eta_2 \\ \diagdown \quad \diagup \\ \bigcirc \\ \diagup \quad \diagdown \\ \eta_1 \quad \eta_3 \end{array} + 2 \begin{array}{c} \eta_1 \quad \eta_3 \\ \diagdown \quad \diagup \\ \bigcirc \\ \diagup \quad \diagdown \\ \eta_2 \end{array} + 2 \begin{array}{c} \eta_3 \\ \diagdown \quad \diagup \\ \bigcirc \\ \diagup \quad \diagdown \\ \eta_1 \quad \eta_2 \end{array} \\ &- 2 \begin{array}{c} \eta_1 \\ \diagdown \quad \diagup \\ \bigcirc \\ \diagup \quad \diagdown \\ \eta_2 \quad \eta_3 \end{array} - 2 \begin{array}{c} \eta_1 \\ \diagdown \quad \diagup \\ \bigcirc \\ \diagup \quad \diagdown \\ \eta_3 \quad \eta_2 \end{array} - 2 \begin{array}{c} \eta_2 \\ \diagdown \quad \diagup \\ \bigcirc \\ \diagup \quad \diagdown \\ \eta_3 \quad \eta_1 \end{array} \quad . \end{aligned} \quad (2.47)$$

It is worth mentioning for practical purposes that since $\partial_k R_k$ just couples to the same fields η_a , we always have $\psi_i \rightarrow \bullet \rightarrow \psi_i$ or $\varphi_i \rightarrow \bullet \rightarrow \varphi_i$.

Let us finally point to some useful notation to get rid of $\partial_k R_k$ -terms in the flow equation diagrams. Due to the identity eq. (2.31) we may write the flow equation as

$$\partial_k \Gamma_k = \frac{1}{2} \tilde{\partial}_k \text{STr} \ln G^{-1} \quad , \quad (2.48)$$

with $\tilde{\partial}_k$ acting on the regulator R_k only. Hence our diagrammatic approach allows us to write

$$\bigcirc = \tilde{\partial}_k \text{STr} \ln G^{-1} \quad (2.49)$$

and acting with the (general) field derivative $\frac{\delta}{\delta\eta}$ on it yields

$$\frac{\delta}{\delta\eta} \textcircled{\bullet} = \eta \textcircled{\bullet} = \tilde{\partial}_k \eta \textcircled{\bullet} \quad , \quad (2.50)$$

where the last equality follows from the action of $\frac{\delta}{\delta\eta}$ on the rhs. of the former relation according to eq. (2.35). Therefore we should agree to write the formal expression

$$\frac{\delta^n}{\delta\eta^n} \textcircled{\bullet} = \tilde{\partial}_k \frac{\delta^{n-1}}{\delta\eta^{n-1}} \textcircled{\bullet} \quad (n \geq 1) \quad , \quad (2.51)$$

which significantly reduces the number of propagators in the diagrams contributing to $\Gamma^{(n)}$ due to the absence of the regulator insertion \bullet (also followed by the generalized propagator $\textcircled{=}$). As we will demonstrate below, one may benefit from it in the discussion whether certain vertices flow at all. Moreover, eq. (2.51) reduces the computational effort to explicitly derive the projected flow equations at first sight since the number of super-matrix multiplications decreases. However, the price to pay is the additional $\tilde{\partial}_k$ -derivative to be evaluated in a second step after having fixed the regulator R_k .

Let us pause for a moment to add some remarks on the scaling of the total number of diagrams with the number n of functional derivatives. First, we recall that eq. (2.50) effectively reduces the amount of diagrams to write down for the projection of the flow. The numerous diagrams for $\Gamma_k^{(n)}$, as e.g. in eq. (2.47), are actually produced by the action of $\frac{\delta^n}{\delta\eta^n}$ on the rhs. of the flow, eq. (2.1), due to the product rule on one hand and eq. (2.30) on the other hand. If we consider any diagram (γ, g) with γ vertices $\Gamma_k^{(i)}$ and g propagators G_k the functional derivative will *produce* γ diagrams (γ, g) as well as g diagrams $(\gamma + 1, g + 1)$, both with one more external leg, i.e.

$$\frac{\delta}{\delta\eta}(\gamma, g) \rightarrow \gamma(\gamma, g) + g(\gamma + 1, g + 1) \quad . \quad (2.52)$$

This recursion helps to estimate the total number of *general diagrams*, e.g. as in eq. (2.47), but without the regulator insertion $\partial_k R_k = \bullet$.

Starting with the flow of $\Gamma_k^{(1)}$ ($n = 1$) we observe that it is directly determined by eq. (2.50) where $\gamma = g = 1$. Hence we define $((m)) \equiv (\gamma, g)|_{\gamma=g=m}$ and eq. (2.52) becomes

$$\frac{\delta}{\delta\eta}((m)) \rightarrow m((m)) + m((m + 1)) \quad . \quad (2.53)$$

Including the *initial condition* $\Gamma_k^{(1)}$ eventually yields

$$N_n = \sum_{\substack{a_1=1 \\ a_i \leq a_{i+1} \leq a_i+1}} \prod_{i=1}^n a_i \quad (2.54)$$

as the total number of general diagrams that have to be evaluated for $\partial_k \Gamma_k^{(n)} \sim \tilde{\partial}_k \sum_{1 \leq m \leq n} ((m))$. The

which is obviously more rapidly increasing with n than N_n . More precisely, our estimate states that the ratio between diagrams without $\partial_k R_k$ and those with regulator insertion decreases as $\mathcal{O}\left(\frac{1}{n}\right)$.

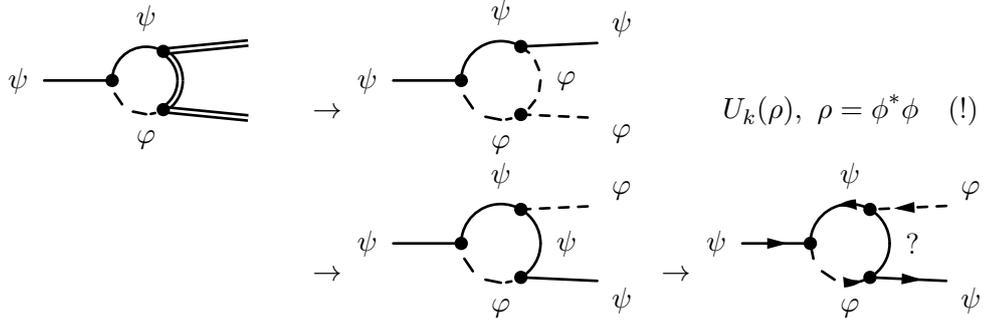
Having established these general technical ideas let us turn back to our original problem, especially to the the specific case where we project the flow onto the coupling h_φ . An appropriate expression for $\partial_k h_\varphi$ is obtained by inserting Γ_k from eq. (2.39) into the lhs. of eq. (2.1), differentiating with respect to the fields φ^* , ψ_\uparrow and ψ_\downarrow and setting all fields to zero ($\eta = 0$) afterwards. If we apply the same procedure to the rhs. of eq. (2.1) we complete the construction of the flow projection onto h_φ and we are immediately able to read off an expression for $\partial_k h_\varphi$:

$$\begin{aligned}
 h_\varphi = \frac{\delta^3 \Gamma_k[\eta]}{\delta\psi_\downarrow \delta\psi_\uparrow \delta\varphi^*} \Big|_{\eta=0} &\rightarrow \partial_k h_\varphi = \frac{1}{2} \tilde{\partial}_k \frac{\delta^3}{\delta\psi_\downarrow \delta\psi_\uparrow \delta\varphi^*} \text{STr} \ln G^{-1} \Big|_{\eta=0} \\
 &\sim \tilde{\partial}_k \left[\text{Diagram 1} \Big|_{\eta=0} + \tilde{\partial}_k \left[\text{Diagram 2} \Big|_{\eta=0} + \tilde{\partial}_k \left[\text{Diagram 3} \Big|_{\eta=0} \right. \right. \right. \\
 &\qquad\qquad\qquad \left. \left. \left. \right. \right. \right.
 \end{aligned} \tag{2.62}$$

where in the last step the \sim indicates that we neglect all numerical prefactors of the diagrams that appear when applying the functional derivatives, to highlight the pure structure of the equation. Of course, the external legs have to be labeled by the fields $\psi_{\uparrow,\downarrow}$ and φ . The crucial observation is now that we can exclude certain diagrams by referring to our truncation Γ_k , eq. (2.39):

1. We did not include any vertex $\Gamma_k^{(5)}$, in particular there is no one with at least the two fermionic fields ψ_\downarrow and ψ_\uparrow
 \rightarrow rules out the first diagram
2. Using this argument again we know that the left vertex of the second diagram must be purely bosonic (arising from the effective potential $U_k(\rho)$, i.e. the flow of λ_k).
 \rightarrow this would imply two external bosonic legs, but we do not consider it for the flow of h_φ with two fermionic legs
3. Let us discuss the last diagram step by step:

First, we should label all external legs. Choosing the left one to be fermionic implies the corresponding vertex to have one further fermionic and one bosonic (both internal, i.e. part of the *one-loop structure* of the flow equation) leg, since our truncation Γ_k allows for exactly one coupling between bosons and fermions, eq. (2.36). In a next step we may label the upper right external leg fermionic, but this leads to a bosonic interaction that involves an odd number of bosons which is excluded due to particle conservation (global U(1) symmetry) when closing the loop. If we would have chosen the upper right external leg to be bosonic we do not encounter this problem, but we may proceed by taking into account the interaction process in more detail. At each vertex the Yukawa-type interaction does only allow the bosonic bound state to *split* into its two composite fermionic parts or the other way around. We denote this fact by arrows along the propagator lines and we obviously see no chance to construct a consistent diagram:



There is one important assumption with respect to our graphical notation which is often fulfilled in practice but it should be explicitly checked when arguing the way we did. From the most general perspective $G_k^{-1} = (\Gamma_k^{(2)} + R_k)$ is not diagonal in the discrete field indices \bar{a} and thus G will not be as well, but the diagrammatic reasoning directly relies on the fact of *mismatching*, i.e. we constructed a contradiction by concatenating the vertices $\Gamma_k^{(n)}$ ($n > 2$) by a propagator $G_k = G_{k,ab}$ that does not vanish for field indices a and b corresponding to the fields $\eta_{\bar{a}}$ only. In our case this statement is ensured by setting the fields η to zero in eq. (2.62) and one explicitly checks it by computing $\Gamma_k^{(2)}\big|_{\eta=0}$ from eq. (2.39) and remembering that $R_{k,ab}$ always satisfies the required diagonal structure in $(\eta_{\bar{a}})$ by construction. Now, the inversion of G^{-1} reduces to invert the blocks

$$\begin{pmatrix} 0 & \pm P_{\eta_{\bar{a}}}^k \\ P_{\eta_{\bar{a}}}^k & 0 \end{pmatrix} \text{ which obviously yields } \begin{pmatrix} 0 & P_{\eta_{\bar{a}}}^{k-1} \\ \pm P_{\eta_{\bar{a}}}^{k-1} & 0 \end{pmatrix} \quad (2.63)$$

with the *elementary propagators* $P_{\eta_{\bar{a}}}^{-1}$ are from $\eta_a^* P_{\eta_{\bar{a}}} \eta_a$ and \pm holds for bosonic and fermionic degrees of freedom, respectively.

4. Summing everything up we conclude to have

$$\partial_k h_\varphi = 0 \quad (2.64)$$

since there is no diagram to contribute to the flow of the Yukawa coupling¹³.

We demonstrated that the graphical representation of the flow equation is useful to determine the contributions to the flow before actually evaluating the rhs. of eq. (2.1). Nevertheless, for concrete computations of the described projection it is often more suitable to start with eq. (2.44) that has *factorized* in the field indices due to the form of the regulator R_k . Before inverting $(\Gamma_k^{(2)} + R_k)$ to get G one should also take into account that one normally sets all fields to zero after the projection prescription and one therefore has to consider $\Gamma_k^{(2)}\big|_{\eta=0}$ for the inversion in this specific case, only.

¹³The vanishing flow of h_φ is actually the reason why we neglected the k -label on it at all.

2.1.4 Momentum Representation of the Effective Averaged Action Γ_k

Our compact notation in eq. (2.39) becomes more explicit when choosing the *space* on which the generalized field η should be defined. For a first intuitive perspective we take the $D \equiv (d + 1)$ -dimensional (Euclidean) *real space* we label by

$$x \equiv (\tau, \mathbf{x}) = (x_0, x_1, \dots, x_d) \quad (2.65)$$

as unified notation for the d spatial dimensions denoted by \mathbf{x} and one *imaginary time*¹⁴ direction $x_0 = \tau \in \mathbb{R}_+$. Furthermore we abbreviate corresponding integrals over the whole range of available x values by¹⁵

$$\int_x \equiv \int_\tau \int_{\mathbf{x}} \equiv \int_0^\beta d\tau \int d^d \mathbf{x} \quad . \quad (2.66)$$

Back to our ansatz for Γ_k , eq. (2.39), and with $\eta = (\varphi, \varphi^*, \psi, \psi^*)$, $\psi = (\psi_\uparrow, \psi_\downarrow)$ we write

$$\begin{aligned} \Gamma_k[\eta] &= \Gamma_k^U[\rho(x)] + \int_{x,y} \left\{ \varphi^*(y) P_{\varphi,k}(y,x) \varphi(y) + \sum_{\sigma=\uparrow,\downarrow} \psi_\sigma^*(y) P_{\psi,k}(y,x) \psi_\sigma(x) \right\} \\ &\quad + \int_{x,y,z} h_\varphi(x,y,z) \varphi^*(x) \psi_\uparrow(y) \psi_\downarrow(z) + c.c. \\ &= \int_x U_k(\rho(x)) + \int_{x,y} \left\{ \varphi^*(y) P_{\varphi,k}(y-x) \varphi(y) + \sum_{\sigma=\uparrow,\downarrow} \psi_\sigma^*(y) P_{\psi,k}(y-x) \psi_\sigma(x) \right\} \\ &\quad + h_\varphi \int_x \varphi^*(x) \psi_\uparrow(x) \psi_\downarrow(x) + c.c. \quad , \end{aligned} \quad (2.67)$$

where we introduced translational invariance of the inverse propagators P_η and restricted the boson-fermion interaction to be point-like with a coupling h_φ independent from x , as apparent from the second line. $\Gamma_k^U[\rho(x)]$ denotes the part of the (flowing) effective action $\Gamma_k[\eta]$ that contributes to the (flowing) effective potential $U_k(\rho)$ where we restrict to the physical situation of *local* bosonic interactions as described next.

To understand our truncation eq. (2.67) in more detail, let us generally comment on the notion of the *effective potential* itself. Formally, $U(\rho)$ is defined as the effective action $\Gamma[\eta]$ evaluated at constant

¹⁴The concept of imaginary time is motivated by relating the functional integral representation of the (canonical) partition function $Z_{stat} \equiv \text{Tr} \exp(-\beta H) = \int \mathcal{D}\eta \exp S[\eta]$ with $S = \int_0^\beta d\tau L$ (L is the system's Lagrangian) of quantum statistical physics with the (many-particle) Hamiltonian H at temperature $T \equiv \beta^{-1}$ (Boltzmann constant $k_B = 1$, see appendix B) to the quantum partition function $Z_{dyn} \equiv \lim_{t \rightarrow \infty} \text{Tr} \exp(-iHt) = \int \mathcal{D}\eta \exp(iS)$ that encodes real time t dynamics of a multi-particle system. For $T = 0$, $\tau \leftrightarrow it$ suffices to establish the connection, but at *finite temperature* $T \neq 0$ one has to impose certain boundary conditions on $\eta(\tau, \mathbf{x})$: Namely, $\varphi(\beta, \mathbf{x}) = \varphi(0, \mathbf{x})$ for bosons and $\psi(0, \mathbf{x}) = -\psi(\beta, \mathbf{x})$ for fermions, i.e. S_{dyn} becomes a multiple of S_{stat} when *compactifying* the imaginary time direction by $\tau \stackrel{\perp}{=} \tau + n\beta$, $n \in \mathbb{Z}$. Roughly speaking, this multiplication factor can be absorbed into the definition of the *measure* $\mathcal{D}\eta$ producing an overall normalization of Z_{dyn} that does not effect physical observables derived from the partition functions. Nevertheless one should always keep in mind that the mathematical procedure of *analytic continuation*^[Nee97,Haz95] of results obtained in one language is not always applicable. Thus the simple substitution $\tau \leftrightarrow it$ might fail. The term *Euclidean* is due to the fact that the Minkowskian *distance* from special relativity $ds^2 \equiv -dt^2 + d\mathbf{x}^2$ becomes $d\tau^2 + d\mathbf{x}^2$.

¹⁵In general, our notation \int_a assumes an integral (summation) over the whole domain of a -values.

fields

$$\eta_a(x) = (\phi, \phi^*, 0, 0, 0) \equiv \Phi \quad , \quad \text{i.e.} \quad \Gamma[\Phi] \equiv V_D U(\phi^* \phi = \rho) \quad , \quad (2.68)$$

where the explicit ρ -dependence is chosen to preserve total particle number in our non-relativistic system. Furthermore, we introduced the appropriate definition of the *space-(imaginary)time volume*

$$V_D \equiv \frac{V_d}{T} \quad \text{with} \quad D = d + 1 \quad , \quad (2.69)$$

where we labeled the space-time volume to integrate over by $V\beta = \frac{V_d}{T}$. The subscript d in V_d reminds us of the general case of a d -dimensional *space volume*. By virtue of this definition it is obvious that terms involving derivatives of $\phi(x)$ do not contribute to $U(\rho)$. On the other hand *interactions* like

$$\prod_{i=1}^n \int_{x_{2i-1}} \int_{x_{2i}} c_i \varphi^*(x_{2i-1}) \varphi(x_{2i}) \quad \supset \Gamma[\eta] \quad \text{with some constants} \quad c_i \quad (2.70)$$

that are invariant under *global* $U(1)$ symmetry $\varphi(x) \rightarrow \varphi(x)e^{i\alpha}$ add to the effective potential. Nevertheless, our physical model restricts to *local* interactions like e.g. $\int_x [\varphi^*(x)\varphi(x)]^n$. More generally we may write

$$\Gamma[\eta] \subset \int_x \bar{U}(\rho(x)) = \sum_n c_n \int_x \rho^n(x) \quad \rightarrow \quad \Gamma[\Phi] = V_D \bar{U}(\rho = \phi^* \phi) \quad , \quad (2.71)$$

where $\bar{U}(\rho)$ represents a function of a single variable ρ and x becomes a pure label for each $\rho(x) = \varphi^*(x)\varphi(x)$ for which \bar{U} is evaluated; the outcome is integrated over, afterwards. Thus, the effective action properly reduces to the definition of the effective potential, i.e. $\bar{U}(\rho) = U(\rho)$.

To get the flow equation it is necessary to derive $\Gamma_k^{(2)}$ first and to invert it afterwards. Therefore we Fourier transform eq. (2.67) with the conjugate quantities defined as

$$q \equiv (q_0, \mathbf{q}) \quad , \quad (2.72)$$

storing all factors of 2π into the definition of the integrals:

$$\int_q \equiv \int_{q_0} \int_{\mathbf{q}} \equiv \int_{q_0} \int \frac{d^d \mathbf{q}}{(2\pi)^d} \quad . \quad (2.73)$$

Furthermore one should take into account that at finite temperature $T = \beta^{-1} \neq 0$ the Fourier transform of τ has to be performed on a finite interval $[0, \beta]$ and hence

$$\int_{q_0} \equiv \begin{cases} \int_{-\infty}^{\infty} \frac{dq_0}{2\pi} & T = 0 \\ \frac{1}{\beta} \sum \omega_n & T > 0 \end{cases} \quad \text{with} \quad \omega_n = \begin{cases} 2n\pi T & \text{for bosons} \\ (2n+1)\pi T & \text{for fermions} \end{cases} \quad (n \in \mathbb{Z}) \quad , \quad (2.74)$$

known as *Matsubara frequencies*. Our convention¹⁶ writes the Fourier transform of each complex field η as

$$\begin{aligned} \eta(x) &= \int_q \eta(q) e^{iq \cdot x} \\ \eta(q) &= \int_x \eta(x) e^{-iq \cdot x} \end{aligned} \quad \text{with} \quad q \cdot x \equiv q_0 \tau + \sum_{i=1}^d q_i x_i = \sum_{i=0}^d q_i x_i \quad (2.75)$$

¹⁶In fact one is allowed to define the Fourier transform differently. It just needs to be consistent. [Appendix A](#) deals with a more careful treatment of the subject.

and correspondingly for (the independent)¹⁷ η^*

$$\eta^*(x) = \int_q \eta^*(q) e^{-iq \cdot x} \quad \text{and} \quad \eta^*(q) = \int_x \eta^*(x) e^{iq \cdot x} \quad , \quad (2.76)$$

which is by some means adopted to *energy-momentum conservation* in the graphical representation we introduced (cf. eq. (2.80)). Moreover, the *Dirac delta distribution* reads

$$\delta(x) = \int_q e^{iqx} \quad \text{and} \quad \delta(q) = (2\pi)^{-D} \int_x e^{iqx} \quad \text{with} \quad \int_x \delta(x) = \int_q (2\pi)^D \delta(q) = 1 \quad . \quad (2.77)$$

One of the factors $1/2\pi$ in $1/(2\pi)^D$ is understood to be substituted by T at finite temperature.

Now, we would like to proceed with representation of the (flowing) effective potential $U_k(\rho)$ in frequency/momentum-space. To understand its Fourier transform we study

$$\begin{aligned} \int_x [\varphi^*(x)\varphi(x)]^n &= \int_{x, q_1, \dots, q_{2n}} \exp \left[ix \cdot \sum_{i=1}^{2n} (-)^i q_i \right] \prod_{i=1}^n \varphi^*(q_{2i-1}) \varphi(q_{2i}) \\ &= (2\pi)^D \int_{q_1, \dots, q_{2n}} \delta \left(\sum_{i=1}^{2n} (-)^i q_i \right) \prod_{i=1}^n \varphi^*(q_{2i-1}) \varphi(q_{2i}) \quad , \end{aligned} \quad (2.78)$$

where we used $\varphi(x) = \phi = \text{const.}$, and therefore $\varphi(q) = \phi \delta(q)$, in the last step. Assuming, as above, that U_k is expandable in a Taylor series in ρ (more generally around some minimum ρ_0), the Fourier transform immediately follows from this expression.

In a next step we transform parts of Γ_k involving the inverse propagators P_η , picking out one particular (η_i, η_i^*) as a representative:

$$\begin{aligned} \int_{x,y} \eta^*(y) P_\eta(y-x) \eta(x) &= \int_{x,y,q,q',q''} \eta^*(q'') P_\eta(q') \eta(q) e^{iq \cdot x} e^{iq' \cdot (y-x)} e^{-iq'' \cdot y} \\ &= (2\pi)^{2D} \int_{q,q',q''} \eta^*(q'') P_\eta(q') \eta(q) \delta(q-q') \delta(q'-q'') \\ &= \int_q \eta^*(q) P_\eta(q) \eta(q) \quad . \end{aligned} \quad (2.79)$$

Similarly we obtain

$$\int_x \varphi^*(x) \psi_\uparrow(x) \psi_\downarrow(x) = \int_{q,q'} \varphi^*(q) \psi_\uparrow(q') \psi_\downarrow(q-q') \quad (2.80)$$

¹⁷The terminology of *independent* has to be understood as follows: From the point of view of the functional $\Gamma[\eta]$, complex fields $\eta \in \mathbb{C}$ carry two additional degrees of freedom per remaining (discrete and continuous) labels for particle species a and e.g. space-time x , respectively. All $\eta_a^*(x)$ are treated as independent variables that Γ depends on and nothing prevents us from using $\eta_{1,2} \in \mathbb{R}$ defined by $\eta = \eta_1 + i\eta_2$ instead of η and its complex conjugate itself. On the other hand, each variable $\eta_a(x)$ ((a,x) fixed) takes a complex value which we are definitely able to derive a complex conjugate from. Therefore, the definition eq. (2.76) is necessary in order to stay consistent: Since the Fourier transform of a complex quantity $\eta(x) = \eta_1(x) + i\eta_2(x)$ can be treated as individually transforming the real functions $\eta_{1,2}(x)$, it follows that $\eta^*(x) \equiv [\eta(x)]^* = \int_q \eta^*(q) e^{-iq \cdot x}$ with $\eta^*(q) = [\eta(q)]^*$.

and end up with the *momentum representation* of the effective averaged action truncation

$$\begin{aligned} \Gamma_k[\eta] = & U_k(\rho) \int_q \delta(q) \delta(q) + \int_q \varphi^*(q) P_{\varphi,k}(q) \varphi(q) + \sum_{\sigma=\uparrow,\downarrow} \int_q \psi_\sigma^*(q) P_{\psi,k}(q) \psi_\sigma(q) \\ & + h_\varphi \int_{q,q'} \{ \varphi^*(q) \psi_\uparrow(q') \psi_\downarrow(q-q') - \varphi(q) \psi_\uparrow^*(q') \psi_\downarrow^*(q-q') \} \quad , \end{aligned} \quad (2.81)$$

where the relative minus sign in the last term arises due to the Grassmann nature of the fermionic fields: $(\psi_\uparrow \psi_\downarrow)^* = \psi_\downarrow^* \psi_\uparrow^* = -\psi_\uparrow^* \psi_\downarrow^*$.

With a view to the flow equation of the effective potential, $\partial_k U_k(\rho)$ (cf. [section 2.2.2](#)), we would like to slightly rewrite our truncation of $\Gamma_k[\eta]$ with respect to the bosonic degrees of freedom represented by $\varphi(x)$. Suppose, we would like to decompose the bosonic field into a space-time independent part ϕ and fluctuations $\tilde{\varphi}(x)$, i.e.

$$\varphi(x) \equiv \phi + \tilde{\varphi}(x) \quad . \quad (2.82)$$

As defined in [eq. \(2.75\)](#) the corresponding Fourier transform yields $\varphi(q=0) = \int_x \varphi(x)$. Namely, the zero-frequency/momentum component represents the tempo-spatial average of a given bosonic field configuration $\varphi(x)$. Now, we associate this number with the (arbitrary) constant ϕ from above and demand

$$\phi \stackrel{!}{=} \frac{1}{V_D} \int_x \varphi(x) = \frac{T}{V_d} \varphi(q=0) \quad . \quad (2.83)$$

Hence, it follows that $\tilde{\varphi}(q) \equiv \int_x \tilde{\varphi}(x) e^{-iq \cdot x}$ vanishes for $q=0$, since the definition [eq. \(2.82\)](#) in accordance with [eq. \(2.83\)](#) induces:

$$\varphi(q) = \int_x \varphi(x) e^{-iq \cdot x} = \phi \delta(q) + \tilde{\varphi}(q) \quad \rightarrow \quad \varphi(q=0) = V_D \phi + \tilde{\varphi}(q=0) \quad . \quad (2.84)$$

Finally we sum up to have

$$\varphi(q) = \phi \delta(q) + \tilde{\varphi}(q) = \begin{cases} V_D \phi = \int_x \varphi(x) & , q=0 \\ \tilde{\varphi}(q) & , q \neq 0 \end{cases} \quad . \quad (2.85)$$

As an important remark we note that ϕ determines the amplitude $\varphi(q)$ of the bosonic field's *zero-momentum mode*, i.e. $\rho = \phi^* \phi$ accounts for the density of particles that occupy the state characterized by $q=0$. Therefore $V_D \rho$ defines the total number of composite (bosonic) particles $\varphi = (\psi_\downarrow \psi_\uparrow)$ that are *condensed*. They built the so called *condensate* which may refer to e.g. Cooper pairs or molecular bound states of the fermionic degrees of freedom.

An equation similar to [eq. \(2.85\)](#) holds for $\varphi^*(q)$ and therefore, we rewrite

$$\begin{aligned} \int_q \varphi^*(q) P_\varphi^k(q) \varphi(q) &= \int_q [\phi^* \delta(q) + \tilde{\varphi}^*(q)] P_\varphi^k(q) [\phi \delta(q) + \tilde{\varphi}(q)] \\ &= V_D \phi^* \phi P_\varphi^k(q=0) + \int_{q \neq 0} \varphi^*(q) P_\varphi^k(q) \varphi(q) + 0 \quad , \end{aligned} \quad (2.86)$$

where we employed the fact that $\tilde{\varphi}(q) = \varphi(q)$ for $q \neq 0$ and the explicit 0 represents terms like $\phi^* P_\varphi^k(0) \tilde{\varphi}(q=0)$ that drop to zero since $\tilde{\varphi}(q)$ vanishes at $q=0$. The upshot of this line of reasoning

becomes clear from eq. (2.86): $P_\varphi^k(q=0)$ directly contributes to the linear part of $U_k(\rho)$. Formulated more precisely, we just define the *slope* $c_{k,1}$ of the linear part of $U_k(\rho) = \sum_n c_{k,n}\rho^n$ as the zero-momentum contribution to $P_\varphi^k(q)$ which in the so called *symmetric phase* (SYM) coincides with a quantity referred to as *mass term* or *gap*. It is denoted by

$$m_{k,\varphi}^2 = c_{k,1} = P_\varphi^k(q=0) \geq 0 \quad (\text{SYM}) \quad . \quad (2.87)$$

However, $c_{k,1}$ might become negative and in this situation one enters the regime of *spontaneously symmetry breaking* (SSB) which we will discuss in more detail in section 2.3.

Extracting the essence from the previous discussion we redefine our truncation from eq. (2.81) as

$$\begin{aligned} \Gamma_k[\eta] = & \int_x U_k(\rho(x)) + \int_{q \neq 0} \varphi^*(q) P_{\varphi,k}(q) \varphi(q) + \sum_{\sigma=\uparrow,\downarrow} \int_q \psi_\sigma^*(q) P_{\psi,k} \psi_\sigma(q) \\ & + h_\varphi \int_{q,q'} \{ \varphi^*(q) \psi_\uparrow(q') \psi_\downarrow(q-q') - \varphi(q) \psi_\uparrow^*(q') \psi_\downarrow^*(q-q') \} \quad , \end{aligned} \quad (2.88)$$

where, now, $P_\varphi^k(q=0) \stackrel{!}{=} 0$ since its contribution (to the flow) is explicitly encoded in the linear part of $\tilde{U}_k(\rho)$. To this end we have $U_k(\rho) \rightarrow U_k(\rho) + P_\varphi^k(q=0)\rho$ when passing from eq. (2.81) to eq. (2.88). In the following we will implicitly adopt

$$P_\varphi^k(q=0) \equiv 0 \quad . \quad (2.89)$$

In particular, if we start with a bosonic *ultraviolet* propagator $P_\varphi^{k=\Lambda}(q) = iq_0 + \mathbf{q}^2 + m_{\Lambda,\phi}^2$ at the *microscopic scale* $k = \Lambda$, where the mass term was fixed by e.g. scattering properties in the *infrared regime* $k = 0$, we have

$$U_\Lambda(\rho) = m_{\Lambda,\varphi}^2 \rho \quad . \quad (2.90)$$

2.1.5 An Important Property of Inverse Propagators

As a closing remark we want to discuss an important restriction on the functional form of the inverse propagator $P(q, q') = P(q)\delta(q - q')$ in momentum space in the case of translational symmetry for $P(x, x') = P(x - x')$ in space-time. As a suitable starting point we adopt a portion of abstract notation in order to avoid masking the basic idea. Therefore, let us declare implicit summation convention on repeated indices. Moreover, we drop $(2\pi)^D$ -factors and space-time/momentum-frequency indices that appear in the definition of the Fourier transform. Additionally, we use a uniform notation for discrete and continuous indices and we adopt Dirac's *bra-ket notation*. To this end, we may write an inverse propagator contribution to the effective action as e.g. chosen in our ansatz for the flowing effective action, eq. (2.67):

$$\Gamma^P = \langle \eta | P | \eta \rangle \quad \text{with Hermitian } P = P^\dagger \quad \text{since } \Gamma = \Gamma^* \quad \text{has to be real.} \quad (2.91)$$

Diving into the space-time representation of the effective action we formulate the completeness of the basis vectors $|x\rangle$ by

$$\langle x | x' \rangle = \delta_{xx'} \quad \text{and the decomposition of the unit operator reads } 1 = \int_x |x\rangle \langle x| \quad . \quad (2.92)$$

The Fourier transform (dropping factors of 2π) is easily introduced by the definition

$$\mathbb{U}_{xq} \equiv \langle x|q \rangle \equiv e^{iq \cdot x} \quad \text{and therefore} \quad \langle q|q' \rangle = \int_x e^{ix \cdot (q-q')} = \delta_{qq'} \quad \text{as well as} \quad \mathbb{U}_{xq}^* \mathbb{U}_{qx'} = \delta_{xx'} \quad , \quad (2.93)$$

which allows us to associate the Fourier transform with the unitary operator \mathbb{U} , i.e. $\mathbb{U}\mathbb{U}^\dagger = 1$ that bridges from the orthonormal basis $\{|x\rangle\}$ of space-time to frequency/momentum $\{|q\rangle\}$. In order to turn eq. (2.91) to a basis-specific representation we define

$$\eta_x \equiv \langle x|\eta \rangle \quad \text{and hence we have} \quad \eta_x^* = \langle \eta|x \rangle \quad , \quad (2.94)$$

and corresponding relations hold for η_q . Note, that the property $\langle \eta|\eta' \rangle^* = \langle \eta'|\eta \rangle$ remains valid for fermions due to $(\psi_x \psi_{x'})^* = \psi_{x'}^* \psi_x$ of Grassmann-valued variables ψ_x . More precisely, we have $\langle \eta|\eta' \rangle^* = \langle \eta|1|\eta' \rangle^* = (\eta_x^* \eta'_x)^* = \eta'_x \eta_x = \langle \eta'|\eta \rangle$.

Back to eq. (2.91) we are in position to obtain the space-time representation of the effective action with the aid of eqs. (2.92) and (2.93):

$$\Gamma^P = \eta_x^* P_{xx'} \eta_{x'} \quad \text{with} \quad \eta_x = \mathbb{U}_{xq} \eta_q \quad \text{and} \quad P_{xx'} = \mathbb{U}_{xq} P_{qq'} \mathbb{U}_{x'q'}^* \quad (2.95)$$

where the space-time representation of the inverse propagator is defined as $P_{xx'} \equiv \langle x|P|x' \rangle$ with its Fourier transform $P_{qq'}$.

Having established these definitions, we conclude that

$$P_{xx'}^* = P_{x'x} \in \mathbb{R} \quad (2.96)$$

from eq. (2.91). As announced above, we would like to take into account translational invariance in space-time, where, for the moment, we focus on *real* time t that is related to *imaginary* time τ by $\tau = it$ (cf. footnote 14), i.e. $x = (t, \mathbf{x})$ and $q = (\omega, \mathbf{q})$ with *real* frequencies ω (different from Matsubara frequencies q_0)¹⁸. In this case, we have

$$P_{xx'} = P(x - x') \quad (2.97)$$

and the property of a Hermitian inverse propagator reads

$$P^*(y) = P(-y) \quad \text{with} \quad y = x - x' \quad . \quad (2.98)$$

However, we would like to further specify $P(x - x')$. Suppose it is written as $P(x - x') = \delta_{xx'} \bar{P}_x$ where \bar{P}_x needs to be translationally invariant. Unless being constant, an ordinary (algebraic) function $\bar{P}(x)$ will not obviously do the job. On the other hand we may choose e.g. $\bar{P}_x = \bar{P}(\partial_x)$ since the derivatives

$$\partial_x \equiv (\partial_\tau, \partial_{\mathbf{x}}) \quad (2.99)$$

¹⁸By the way: The scalar product $x \cdot q$ has *Minkowskian signature*, now: $x \cdot q = \omega t - \mathbf{q} \cdot \mathbf{x}$; in contrast to the *Euclidean metric* in imaginary time $x \cdot q = q_0 \tau + \mathbf{q} \cdot \mathbf{x}$.

are translationally invariant operators. Written as finite difference operator for the one-dimensional case ($x \in \mathbb{R}$) in e.g. the basis $\{x_n = n\epsilon, n \in \mathbb{Z}\}$ the derivative of a scalar function f takes the form

$$\partial_x f = \lim_{\epsilon \rightarrow 0} \frac{1}{\epsilon} \begin{pmatrix} \ddots & & & & & \\ & -1 & 1 & & & \\ & & -1 & 1 & & \\ & & & -1 & 1 & \\ & & & & \ddots & \ddots \end{pmatrix} \begin{pmatrix} \vdots \\ f_{n-1} \\ f_n \\ f_{n+1} \\ \vdots \end{pmatrix} \quad (2.100)$$

with $f_n \equiv f(x_n)$ and $\epsilon = x_{n+1} - x_n$. Therefore we notice that $P_{x'x} = \delta_{xx'} \bar{P}(-\partial_x)$ and the property of the inverse propagator being Hermitian requires

$$\bar{P}(\partial_x) = \bar{P}^*(-\partial_x) \quad . \quad (2.101)$$

Since $\bar{P}(\partial_x)$ is defined by the Taylor expansion of the algebraic function $\bar{P}(x)$ (around $x = 0$) we may compensate the minus sign in front of ∂_x by $\partial_x \rightarrow i\partial_x$, i.e. space-time translation invariance of the inverse propagator with diagonal structure

$$P_{xx'} = P(x - x') = \delta_{xx'} \bar{P}_x \quad \text{requires} \quad \bar{P}_x = \bar{P}(i\partial_t, -i\partial_{\mathbf{x}}) \quad (2.102)$$

which ensures that $P_{xx'}$ is Hermitian.

The additional minus sign was introduced just for convenience and it matches standard notation from quantum mechanics. Let us use this point to recall some philosophy behind quantum physics¹⁹: According to de Broglie's hypothesis^[dB70], energy E and momentum \mathbf{p} of a particle are proportional to some frequency ω and momentum \mathbf{q} , i.e. $E = \hbar\omega$ and $\mathbf{p} = \hbar\mathbf{q}$ with Planck constant \hbar which we set to 1 in our system of units (cf. appendix B). As it turned out, it seems to be sensible to describe quantum physics of a single (spinless) particle by a complex field $\Psi(t, \mathbf{x})$ whose absolute value $|\Psi|^2$ determines the particle's (tempo-spatial) probability distribution. In the *free* case, *plain waves*²⁰ $\Psi_f \sim \exp[\pm i(\omega t - \mathbf{q} \cdot \mathbf{x})]$, are associated with *non-interacting* particles which obey a *dispersion* $\omega = \omega(\mathbf{q})$, i.e. an energy-momentum relation $E = E(\mathbf{p})$ ²¹. Furthermore, we assume $\lim_{\mathbf{p} \rightarrow \infty} E(\mathbf{p}) \rightarrow +\infty$ in order to ensure a physically *stable* system; otherwise, the *principle of least energy* might eventually drive the particle's velocity above the speed of light.

The *wave front* which is defined by a constant *phase* $\alpha \equiv \omega t - \mathbf{q} \cdot \mathbf{x} = \text{const.}$ moves into the direction of \mathbf{q} at the absolute speed of $\omega(\mathbf{q})/|\mathbf{q}|$ as obvious from $\frac{d\alpha}{dt} = 0$. One might have been associated $-\mathbf{q}$ as propagation direction as well, and therefore the relative minus sign between ωt and $\mathbf{x} \cdot \mathbf{q} = -\mathbf{x} \cdot (-\mathbf{q})$ would have been absorbed. Actually, an analogous trick is used in relativistic physics where for each

¹⁹Standard literature as [Sha94a] should cover those ideas in much more detail. We just like to outline the rough concept in order to obtain intuition on the inverse propagator \bar{P} .

²⁰Adopting notation from special relativity with space-time 4-vector $x^\nu = (t, \mathbf{x})$, energy-momentum 4-vector $q^\nu = (\omega, \mathbf{q})$ and Minkowskian metric $g_{\mu\nu} = \text{diag}(1, -1, -1, -1)$ we have $\Psi_f \sim \exp[\pm i x \cdot q]$.

²¹Note, although we may associate a well defined (*phase*) *velocity* $v = \omega(\mathbf{q})/|\mathbf{q}|$ to a plain wave there is no meaningful notion of *position* which reflects a particular case of Heisenberg's principle of uncertainty.

$E = \omega$ there exists an $\bar{E} = -E$ due to the relation $q \cdot q = m^2$ for particles of mass m (speed of light $c = 1$) and it seems impossible to construct a stable system. But the solution to this problem is to write $\pm\omega t = |\omega|(\pm t)$ and interpret particles with negative energy as having positive energy but traveling backwards in time. However, it is appropriate to choose \mathbf{q} parallel to the direction of propagation and therefore we stick to the (standard) convention of having a relative minus sign.

Now, a plain wave decomposition of Ψ will pave the way to a recipes in order to guess a suitable differential equation for the dynamics of Ψ as follows: Demand the Fourier transform $\Psi(\omega, \mathbf{q}) = \int_{t, \mathbf{x}} \Psi(t, \mathbf{x}) \exp[iq \cdot x]$ to obtain $\omega\Psi = \omega(\mathbf{q})\Psi$ and therefore we associate $i\partial_t \leftrightarrow \omega = E$ and $-i\partial_{\mathbf{x}} \leftrightarrow \mathbf{q} = \mathbf{p}$. Note, that we end up with the Schrödinger equation $i\partial_t\Psi = E(-i\partial_{\mathbf{x}})\Psi \equiv H\Psi$ where H denotes the *Hamilton operator*. More generally, one might define the translationally invariant²² inverse propagator $\bar{P} = -E + H = \bar{P}(i\partial_t, -i\partial_{\mathbf{x}})$ and the Schrödinger equation becomes $\bar{P}\Psi = 0$ which one refers to as *on-shell condition*. As an example take the *free-particle* inverse propagator from a quadratic dispersion relation $\omega = \omega(\mathbf{q}) = \mathbf{q}^2/2m$ that reads $\bar{P} = -E + H_0 = -i\partial_t - \partial_{\mathbf{x}}^2/2m$ with $H_0 = \frac{1}{2m}(i\partial_{\mathbf{x}}) \cdot (i\partial_{\mathbf{x}})$ the single particle Hamiltonian of mass m . Following the route to *many-particle quantum physics/quantum field theory* (cf. e.g. ref. [AS10]) a species of indistinguishable particles is still described by some field²³ in space-time whose Fourier modes are interpreted as particles with energy $\omega(\mathbf{q})$. However, by virtue of the *path integral* there is no on-shell condition $\bar{P} = 0$ any more which is called *off-shell* and associated 'particles' are referred to as *virtual*.

However, according to eqs. (2.93) and (2.95) the momentum representation of the inverse propagator eq. (2.102) reads²⁴

$$P_{qq'} = \mathbb{U}_{qx}^* \bar{P}_x \mathbb{U}_{xq'} = \delta_{qq'} \bar{P}_{q'} \quad \text{with} \quad \bar{P}_q \equiv \bar{P}(-\omega, -\mathbf{q}) \in \mathbb{R} \quad (2.103)$$

which is in accordance with the general property $P_{qq'} = P_{q'q}^*$. The final twist to the story enters when we turn back to imaginary time where $-i\partial_t \rightarrow \partial_\tau \in [0, \beta = T^{-1})$. Moreover, we require *spatial isotropy* in the sense that \bar{P}_q depends on \mathbf{q}^2 instead of \mathbf{q} , i.e. only the magnitude of the particle's momentum but not its spatial direction is relevant to the energy-momentum dispersion relation. Hence, we have

$$P_{xx'} = \delta_{xx'} \bar{P}(-\partial_\tau, -\partial_{\mathbf{x}}^2) \quad . \quad (2.104)$$

Applying the Fourier transform with (real valued) Matsubara frequencies²⁵ q_0 yields

$$P_{qq'} = \delta_{qq'} \bar{P}(-iq_0, \mathbf{q}^2) \equiv \delta_{qq'} \bar{\bar{P}}(q_0, \mathbf{q}^2) \quad \text{with} \quad \bar{\bar{P}}^*(q_0, \mathbf{q}^2) = \bar{\bar{P}}(-q_0, \mathbf{q}^2) \quad (2.105)$$

since $\bar{\bar{P}}^*(iq_0, \mathbf{q}^2) = \bar{\bar{P}}(-iq_0, \mathbf{q}^2)$ and $P_{qq'} = P_{q'q}^*$. Due to the fact that $\bar{\bar{P}}(q)$ depends on \mathbf{q}^2 we state for an inverse propagator of a time translationally invariant (thermodynamics!) and spatial isotropic system

²²Nothing prevents us from picking a \bar{P} that explicitly depends on \mathbf{x} and t in addition. Indeed, the former becomes relevant for e.g. *external potential* $V = V(\mathbf{x})$ and the latter introduces when studying *non-equilibrium* where e.g. *dissipation* reduces the system's total energy.

²³Which field has to be used depends on the particle's statistics which in turn is linked to the particle's spin by virtue of the *Spin-Statistics-Theorem* [Pau40]. Most prominently, spinless particles (bosons) are represented by complex scalar fields and spin-1/2 particles (fermions) by Grassmann fields.

²⁴The derivatives of \bar{P}_x act on $\mathbb{U}_{xq'}$ such that $\bar{P}(i\partial_t, -i\partial_{\mathbf{x}}) \rightarrow \bar{P}(-\omega', -\mathbf{q}')$.

²⁵Note, that we can rewrite $\omega t = (-i\omega)(it) = q_0\tau$.

in Matsubara formalism:

$$P_{qq'} = \delta_{qq'} \bar{P}(q) \quad \text{with} \quad \bar{P}^*(q) = \bar{P}(-q) \quad \text{where} \quad q = (q_0, \mathbf{q}) \quad . \quad (2.106)$$

Note, that this implies the identities

$$\Re \bar{P}(-q) = \Re \bar{P}(q) \quad (\text{symmetric}) \quad \text{and} \quad \Im \bar{P}(-q) = -\Im \bar{P}(q) \quad (\text{anti-symmetric}) \quad . \quad (2.107)$$

2.2 Doing the Math

We now turn to the derivation of the set of equations following from the flow equation, eq. (2.1). They are adapted to describe the physics of a non-relativistic gas of two species of fermions within the truncation specified by eq. (2.67). After a warm-up that restricts to the symmetric phase (SYM, $\varphi = \psi_{\uparrow\downarrow} = 0$) we extend our findings to the spontaneously broken phase (SSB, $\varphi = \phi, \psi_{\uparrow\downarrow} = 0$) in section 2.3. An even more general setting where the two fermionic species $\psi_{\uparrow\downarrow}$ are treated differently with respect to their corresponding (inverse) propagators $P_{\uparrow\downarrow}(q)$ is given in appendix D.

2.2.1 Facing the Full Inverse Propagator $\Gamma_k^{(2)}$

As a first step to set up the flow equation we have to specify $\Gamma_k^{(2)}$ in an appropriate basis. It turns out that the *momentum representation* eq. (2.88) is suitable to easily obtain G_k since $\Gamma_k^{(2)}$ is diagonal in q and the regulator R_k is defined via an additive diagonal term in momentum space. Being more precise and following the notation of eqs. (2.2) and (2.41), we define

$$\Gamma_{k,ab}^{(2)}(q_1, q_2) \equiv \frac{\overrightarrow{\delta}}{\delta\eta_a(-q_1)} \Gamma_k \frac{\overleftarrow{\delta}}{\delta\eta_b(q_2)} \quad \text{in the basis} \quad \eta_a(q) \equiv (\varphi(q), \varphi^*(-q), \psi(q), \psi^*(-q))^{26} \quad (2.108)$$

with $\psi(q) \equiv (\psi_{\uparrow}(q), \psi_{\downarrow}(q))$. The contribution $U_k^{(2)}$ of the effective potential U_k to $\Gamma_k^{(2)}$ becomes clear, if we take the functional derivative of eq. (2.78) with respect to $\varphi(\tilde{q})$. Having in mind that $\varphi(x) = \phi = \text{const.}$ we conclude:

$$\begin{aligned} \frac{\delta}{\delta\varphi(\tilde{q})} \int_x [\varphi^*(x)\varphi(x)]^n &= n\phi^{n-1}\phi^{*n} \int_{q_1, \dots, q_{2n-1}} \delta \left(\sum_{i=1}^{2n-1} (-)^i q_i + \tilde{q} \right) \prod_{i=1}^{2n-1} (2\pi)^D \delta(q_i) \\ &= \phi^{*n} \left(\frac{\partial \phi^n}{\partial \phi} \right) \delta(\tilde{q}) \quad , \end{aligned} \quad (2.109)$$

which (without mathematical rigor) allows us to write

$$\frac{\delta^n}{\delta\varphi^*(q_n) \dots \delta\varphi^*(q_{m+1}) \delta\varphi(q_m) \dots \delta\varphi(q_1)} \int_x U_k(\rho(x)) = (2\pi)^{(1-n)D} \frac{\partial^n U_k(\rho)}{\partial \phi^{*n-m} \partial \phi^m} \delta \left(\sum_{i=1}^m q_i - \sum_{i=m+1}^n q_i \right) . \quad (2.110)$$

Hence, eq. (2.108) yields

$$U_{k,BB'}^{(2)}(q_1, q_2) = \left(\begin{array}{cc} \frac{\delta^2}{\delta\varphi(-q_1)\delta\varphi(q_2)} & \frac{\delta^2}{\delta\varphi(-q_1)\delta\varphi^*(-q_2)} \\ \frac{\delta^2}{\delta\varphi^*(q_1)\delta\varphi(q_2)} & \frac{\delta^2}{\delta\varphi^*(q_1)\delta\varphi^*(-q_2)} \end{array} \right) \int_x U_k = (2\pi)^{-D} \left(\begin{array}{cc} \frac{\partial^2 U_k}{\partial \phi^2} & \frac{\partial^2 U_k}{\partial \phi \partial \phi^*} \\ \frac{\partial^2 U_k}{\partial \phi^* \partial \phi} & \frac{\partial^2 U_k}{\partial \phi^{*2}} \end{array} \right) \delta(q_1 - q_2) \quad (2.111)$$

and

$$U_{k,FB}^{(2)}(q_1, q_2) = U_{k,BF}^{(2)}(q_1, q_2) = 0 \quad \text{as well as} \quad U_{k,FF'}^{(2)}(q_1, q_2) = 0 \quad , \quad (2.112)$$

²⁶The additional minus signs for the q -basis is manually introduced to get the diagonal structure of $\Gamma_k^{(2)}$ in momentum space. If one changes the discrete basis to e.g. two real fields φ_1 and φ_2 for the bosonic field $\varphi = \varphi_1 + i\varphi_2$ one should define $(\varphi_1(q), \varphi_2(q))$ in order to get the same diagonal structure for $\Gamma_k^{(2)}$, see also appendix A. A perhaps more transparent treatment provides appendix A.

since U_k just contains pure bosonic interaction terms and does not couple to the *fermionic sector* $a = F$. As above we used the notational convention to write B and F for bosonic and fermionic field indices, respectively. Since the discrete and continuous indices *decoupled* in eq. (2.112), i.e. we have a diagonal structure in momentum space, we are used to write

$$U_{k,ab}^{(2)}(q_1, q_2) \equiv U_{k,ab}^{(2)} \delta(q_1 - q_2) \quad . \quad (2.113)$$

Deriving the contribution P_η^k of the inverse propagators (second and third summand of eq. (2.88)) we agree to have

$$P_{\eta,BB'}^k(q_1, q_2) = (2\pi)^{-D} \begin{pmatrix} 0 & P_\varphi^k(-q_1) \\ P_\varphi^k(q_1) & 0 \end{pmatrix} \delta(q_1 - q_2) \quad , \quad P_{\eta,BF}^k(q_1, q_2) = P_{\eta,FB}^k(q_1, q_2) = 0 \quad (2.114)$$

and

$$P_{\eta,FF'}^k(q_1, q_2) = (2\pi)^{-D} \begin{pmatrix} 0_{2 \times 2} & -P_\psi^k(-q_1) 1_{2 \times 2} \\ P_\psi^k(q_1) 1_{2 \times 2} & 0_{2 \times 2} \end{pmatrix} \delta(q_1 - q_2) \quad . \quad (2.115)$$

The 2×2 -label indicates that $0_{2 \times 2}$ and $1_{2 \times 2}$ are the zero and unit matrix, respectively. If it is rather clear from the context, we will hide such cumbersome notation in the following text. The relative minus sign in eq. (2.115) compared to eq. (2.114) is again caused by the anti-commuting nature of the Grassmann variables (ψ, ψ^*) . At this stage the origin of the flow equation, the regulator R_k (cf. eq. (2.4)) reenters. As mentioned it acts like a mass contribution to the inverse propagators P_η^k and we therefore just have to *regulate* them by

$$\begin{aligned} P_\varphi^{k,reg}(q) &\equiv P_\varphi^k(q) + k^2 r_{k,\varphi} \left(\frac{q}{k}\right) \\ P_\psi^{k,reg}(q) &\equiv P_\psi^k(q) + k^2 r_{k,\psi} \left(\frac{q}{k}\right) \end{aligned} \quad \text{and for} \quad P_{\eta,ab}^{k,reg}(q_1, q_2) \equiv P_{\eta,ab}^k(q_1, q_2) + R_{k,ab} \delta(q_1 - q_2) \quad (2.116)$$

we explicitly obtain the expression

$$(2\pi)^{-D} \begin{pmatrix} 0 & P_\varphi^{k,reg}(-q_1) & & \\ P_\varphi^{k,reg}(q_1) & 0 & & \\ \hline & & 0 & -1P_\psi^{k,reg}(-q_1) \\ & & 1P_\psi^{k,reg}(q_1) & 0 \end{pmatrix} \delta(q_1 - q_2) \quad (2.117)$$

for $P_{\eta,ab}^{k,reg}(q_1, q_2)$ and, for visual convenience, we separated the fermionic from the bosonic contributions by dashed lines. The *mixed* part is left blank since it is identically zero. Similar to eq. (2.113) we would like to define

$$P_{\eta,ab}^{k,reg}(q_1, q_2) \equiv P_{\eta,ab}^{k,reg}(q_1) \delta(q_1 - q_2) \quad . \quad (2.118)$$

Turning to the last contribution to $\Gamma_k^{(2)}$ from the Yukawa interaction between bosonic and fermionic degrees of freedom we arrive at

$$\frac{h_\varphi}{(2\pi)^{2D}} \left(\begin{array}{cc|cc} & & 0 & 0 & \psi_\downarrow^*(q_2 - q_1) & -\psi_\uparrow^*(q_2 - q_1) \\ & & -\psi_\downarrow(q_1 - q_2) & \psi_\uparrow(q_1 - q_2) & 0 & 0 \\ \hline 0 & \psi_\downarrow(q_1 - q_2) & 0 & \varphi^*(q_2 - q_1) & & \\ 0 & -\psi_\uparrow(q_1 - q_2) & -\varphi^*(q_2 - q_1) & 0 & & \\ \hline -\psi_\downarrow^*(q_2 - q_1) & 0 & & & 0 & -\varphi(q_1 - q_2) \\ \psi_\uparrow^*(q_2 - q_1) & 0 & & & \varphi(q_1 - q_2) & 0 \end{array} \right). \quad (2.119)$$

Blank entries are again zero elements and we will abbreviate this expression by $H_{\varphi,ab}(q_1, q_2)$ for the calculations to come.

As a last ingredient to derive the rhs. of the flow equation we mention to have

$$(2\pi)^{-D} \left(\begin{array}{cc|cc} 0 & \partial_k k^2 r_{k,\varphi} \left(-\frac{q_1}{k}\right) & & \\ \hline \partial_k k^2 r_{k,\varphi} \left(\frac{q_1}{k}\right) & 0 & & \\ \hline & & 0 & -1\partial_k k^2 r_{k,\psi} \left(-\frac{q_1}{k}\right) \\ & & 1\partial_k k^2 r_{k,\psi} \left(\frac{q_1}{k}\right) & 0 \end{array} \right) \delta(q_1 - q_2) \quad (2.120)$$

for $\partial_k R_{k,ab}(q_1, q_2)$ in total analogy to the expression, eq. (2.117).

2.2.2 The Flowing Effective Potential $U_k(\rho)$

It is now the aim to invert

$$G_{k,ab}^{-1}(q_1, q_2) = \Gamma_{k,ab}^{(2)}(q_1, q_2) + R_{k,ab} \delta(q_1 - q_2) = \left[U_{k,ab}^{(2)} + P_{\eta,ab}^{k,reg}(q_1) \right] \delta(q_1 - q_2) + H_{\varphi,ab}(q_1 - q_2) \quad , \quad (2.121)$$

but we can reduce the computational effort by recalling the projection procedure discussed parallel to the introduction of the diagrammatic representation of the flow equation. Much simplification in G_k^{-1} will be gained by setting fields to be constant or even zero. We therefore continue by deriving expressions for the projection to our flowing quantities $U_k(\rho)$, $P_\phi^k(q)$ and $P_\psi^k(q)$ from our truncation Γ_k . Differentiating eq. (2.88) with respect to the *flow parameter* k and demanding bosonic fields to be constant in real space x ($\varphi^{(*)}(x) = \phi^{(*)}$) as well as setting all fermionic fields $\psi^{(+)}$ to zero we check to have

$$\partial_k \Gamma_k \Big|_{\substack{\varphi^{(*)}(x) = \phi^{(*)} \\ \psi^{(+)} = 0}} = V_D \partial_k U_k(\rho) \quad . \quad (2.122)$$

An appropriate expression for the flow of the effective potential \tilde{U}_k is calculated by evaluating the rhs. of the flow equation and setting the fields $\eta(x)$ to be constant or identically zero (for fermions). In momentum space this corresponds to $\varphi^{(*)}(q) = (2\pi)^D \phi^{(*)} \delta(q)$ and $\psi(q) = 0$. To sum up we have to

evaluate

$$V_D \partial_k U_k = \frac{1}{2} \text{STr} G_k \partial_k R_k \Big|_{\substack{\varphi = \phi \\ \psi = 0}}, \quad (2.123)$$

where we simplified the notation to focus on the main aspect. Referring to eq. (2.119) we obtain an important simplification demanding the fields to be constant:

$$H_{\varphi, FF'}(q_1, q_2) = \frac{h_\varphi}{(2\pi)^D} \begin{pmatrix} \epsilon \phi^* & 0 \\ 0 & -\epsilon \phi \end{pmatrix} \delta(q_1 - q_2) \quad \text{and} \quad H_{\varphi, BF}(q_1, q_2) = H_{\varphi, FB}(q_1, q_2) = 0 \quad (2.124)$$

with the *total antisymmetric* (two-dimensional) symbol $\epsilon = \epsilon_{ij}$ ($i, j = 1, 2$). Therefore the propagator G_k^{-1} acquires the structure

$$G_{k,ab}^{-1}(q_1, q_2) = (2\pi)^{-D} \begin{pmatrix} G_{k, BB'}^{-1}(q_1) & 0 \\ 0 & G_{k, FF'}^{-1}(q_1) \end{pmatrix} \delta(q_1 - q_2) \quad (2.125)$$

with

$$G_{k, BB'}^{-1}(q) = \begin{pmatrix} \phi^{*2} \partial_\rho^2 U_k & \partial_\rho U_k + \rho \partial_\rho^2 U_k + P_\varphi^{k, reg}(-q) \\ \partial_\rho U_k + \rho \partial_\rho^2 U_k + P_\varphi^{k, reg}(q) & \phi^2 \partial_\rho^2 U_k \end{pmatrix} \quad (2.126)$$

and

$$G_{k, FF'}^{-1}(q) = \begin{pmatrix} \epsilon \phi^* h_\varphi & -1 P_\psi^{k, reg}(-q) \\ 1 P_\psi^{k, reg}(q) & -\epsilon \phi h_\varphi \end{pmatrix}. \quad (2.127)$$

Following the derivation of the flow equation, G_k is defined as

$$\int_z G_k(x, z) G_k^{-1}(z, y) = \delta(x - y) \quad (2.128)$$

in real space, hiding the discrete index structure. In momentum space, this leads to the requirement²⁷

$$\int_q G(q_1, q) G^{-1}(q, q_2) = (2\pi)^D \delta(q_1 - q_2) \quad (2.129)$$

Writing

$$G_{k,ab}(q_1, q_2) = (2\pi)^{3D} \begin{pmatrix} G_{k, BB'}(q_1) & 0 \\ 0 & G_{k, FF'}(q_1) \end{pmatrix} \delta(q_1 - q_2) \quad (2.130)$$

we immediately see that $\sum_c \int_q G_{k,ac}(q_1, q) G_{k,cb}^{-1}(q, q_2) = (2\pi)^D \delta_{ab} \delta(q_1 - q_2)$ and thus it remains to invert the discrete matrices, eqs. (2.126) and (2.127). The negative of the determinant of $G_{k, BB'}^{-1}(q)$ reads

$$\begin{aligned} |G_B^{-1}(q)| &\equiv -\det[G_{k, BB'}^{-1}(q)] \\ &= \rho U_k'' U_k' + (U_k' + \rho U_k'') [P_\varphi^{k, reg}(q) + P_\varphi^{k, reg}(-q) + U_k'] + P_\varphi^{k, reg}(q) P_\varphi^{k, reg}(-q) \end{aligned} \quad (2.131)$$

²⁷One may check the relation by: $\delta(y - x) = \int_q e^{iq \cdot (y-x)}$ and on the other hand we have $\int_z G(x, z) G^{-1}(z, y) = \int_{z, q, q', q'', q'''} G(q, q') G^{-1}(q'', q''') e^{-iq \cdot x} e^{iz \cdot (q' - q''')} e^{iq'' \cdot y} = \int_{q, q'''} \left[\int_{q'} G(q, q') G^{-1}(q', q''') \right] e^{i(y \cdot q'' - x \cdot q)}$. The relative minus sign when Fourier transforming the first argument of $G(x, y)$ is due to the convention in eq. (2.108).

with $U_k^{(n)} \equiv \partial_\rho^n U_k$ and the property $|G_B^{-1}(q)| = |G_B^{-1}(-q)|$. According to Laplace's formula we have

$$G_{k,BB'}(q) = \frac{1}{|G_B^{-1}(q)|} \begin{pmatrix} -\phi^2 U_k'' & U_k' + \rho U_k'' + P_\varphi^{k,reg}(-q) \\ U_k' + \rho U_k'' + P_\varphi^{k,reg}(q) & -\phi^{*2} U_k'' \end{pmatrix}. \quad (2.132)$$

For fermions we employ the fact that all 2×2 -sub-matrices in eq. (2.127) commute, i.e. $[1, 1] = [1, \epsilon] = [\epsilon, \epsilon] = 0$ which allows us to write the inverse matrix as

$$G_{k,FF'}(q) = \frac{1}{|G_F^{-1}(q)|} \begin{pmatrix} -\epsilon \phi h_\varphi & 1 P_\psi^{k,reg}(-q) \\ -1 P_\psi^{k,reg}(q) & \epsilon \phi^* h_\varphi \end{pmatrix} \quad (2.133)$$

with

$$\frac{1}{|G_F^{-1}(q)|} \equiv [-\epsilon^2 \rho h_\varphi^2 + 1 P_\psi^{k,reg}(q) P_\psi^{k,reg}(-q)]^{-1} = \frac{1}{\rho h_\varphi^2 + P_\psi^{k,reg}(q) P_\psi^{k,reg}(-q)} \quad (2.134)$$

and again $|G_F^{-1}(q)| = |G_F^{-1}(-q)|$. Finally we have to evaluate

$$\begin{aligned} \text{STr } G_k \partial_k R_k &= \int_{x,y} \zeta_a G_{k,ab}(x,y) \partial_k R_{k,ba}(y,x) \\ &= \int_{q,q'} \zeta_a G_{k,ab}(q,q') \partial_k R_{k,ba}(q',q) \\ &= V_D \int_q \left[G_{k,BB'}(q) \partial_k R_{k,B'B}(q) - G_{k,FF'}(q) \partial_k R_{k,F'F}(q) \right] \end{aligned} \quad (2.135)$$

where we defined ζ_a to be ± 1 for $a = B$ and $a = F$, respectively. Therefore the flow equation for the effective potential becomes

$$\begin{aligned} \partial_k U_k(\rho) &= \frac{1}{2} \int_q \frac{1}{|G_B^{-1}(q)|} \left[(U_k' + \rho U_k'') \left(\partial_k k^2 r_{k,\varphi}(\frac{q}{k}) + \partial_k k^2 r_{k,\varphi}(-\frac{q}{k}) \right) \times \right. \\ &\quad \left. + P_\varphi^{k,reg}(-q) \partial_k k^2 r_{k,\varphi}(\frac{q}{k}) + P_\varphi^{k,reg}(q) \partial_k k^2 r_{k,\varphi}(-\frac{q}{k}) \right] \\ &\quad - \int_q \frac{1}{|G_F^{-1}(q)|} \left[P_\psi^{k,reg}(-q) \partial_k k^2 r_{k,\psi}(\frac{q}{k}) + P_\psi^{k,reg}(q) \partial_k k^2 r_{k,\psi}(-\frac{q}{k}) \right] \end{aligned}$$

with

$$\begin{aligned} U_k^{(n)} &= \partial_\rho^n U_k(\rho), \quad \rho = \phi^* \phi \\ |G_B^{-1}(q)| &= \rho U_k'' U_k' + (U_k' + \rho U_k'') [P_\varphi^{k,reg}(q) + P_\varphi^{k,reg}(-q) + U_k'] + P_\varphi^{k,reg}(q) P_\varphi^{k,reg}(-q) \\ |G_F^{-1}(q)| &= \rho + P_\psi^{k,reg}(q) P_\psi^{k,reg}(-q) \\ P_\eta^{k,reg}(q) &= P_\eta^k(q) + k^2 r_{k,\eta}(\frac{q}{k}) \\ \int_q &= \int_{q_0} \int \frac{d^d q}{(2\pi)^d}, \quad \int_{q_0} = \begin{cases} \int_{-\infty}^{\infty} \frac{dq_0}{2\pi} & T = 0 \\ \frac{1}{\beta} \sum_{\omega_n} & T > 0 \end{cases}, \quad \omega_n = \begin{cases} 2n\pi T & \text{bosons} \\ (2n+1)\pi T & \text{fermions} \end{cases}. \end{aligned} \quad (2.136)$$

2.2.3 Deriving the Flow of the Inverse Propagators

Next, we compute the flow of $P_\phi^k(q)$, since it appears in the flow of the effective potential, eq. (2.136). The corresponding projection prescription reads

$$\partial_k \tilde{P}_\varphi^k(q) \equiv \partial_k P_\varphi^k(q) + \left. \frac{\partial^2 \partial_k U_k}{\partial \phi^* \partial \phi} \right|_{\phi=0} = (2\pi)^{2D} V_D^{-1} \left. \frac{\delta^2 \partial_k \Gamma_k}{\delta \varphi(q) \delta \varphi^*(q)} \right|_{\eta=0}. \quad (2.137)$$

Referring to the rhs. of the flow equation we have (cf. eq. (2.51))

$$\begin{aligned} \partial_k \tilde{P}_\varphi^k(q) &= \frac{1}{2} (2\pi)^{2D} V_D^{-1} \tilde{\partial}_k \left. \frac{\delta^2}{\delta \varphi(q) \delta \varphi^*(q)} \text{STr} \ln G_k^{-1} \right|_{\eta=0} \\ &= \frac{1}{2} (2\pi)^{2D} V_D^{-1} \tilde{\partial}_k \left\{ \varphi(q) \text{---} \text{---} \text{---} \text{---} \varphi(q) \left|_{\eta=0} - \varphi(q) \text{---} \text{---} \text{---} \text{---} \varphi(q) \left|_{\eta=0} \right. \right\} \end{aligned} \quad (2.138)$$

and we therefore have to compute $\left. \frac{\delta \Gamma_k^{(2)}}{\delta \varphi(q)} \right|_{\eta=0}$, $\left. \frac{\delta \Gamma_k^{(2)}}{\delta \varphi^*(q)} \right|_{\eta=0}$ and $\left. \frac{\delta^2 \Gamma_k^{(2)}}{\delta \varphi^*(q) \delta \varphi(q)} \right|_{\eta=0}$ in addition to $G_k|_{\eta=0}$. Using eq. (2.125) and demanding all fields η to vanish we deduce

$$G_{k,ab}^0(q_1, q_2) \equiv G_{k,ab}(q_1, q_2)|_{\eta=0} = (2\pi)^{3D} \begin{pmatrix} G_{k,BB'}^0(q_1) & 0 \\ 0 & G_{k,FF'}^0(q_1) \end{pmatrix} \delta(q_1 - q_2) \quad (2.139)$$

with the bosonic contribution

$$G_{k,BB'}^0(q) = \begin{pmatrix} 0 & \frac{1}{U'_k|_{\rho=0} + P_\varphi^{k,reg}(q)} \\ \frac{1}{U'_k|_{\rho=0} + P_\varphi^{k,reg}(-q)} & 0 \end{pmatrix} \quad (2.140)$$

and the fermionic one,

$$G_{k,FF'}^0(q) = \begin{pmatrix} 0 & \frac{1}{P_\psi^{k,reg}(q)} \\ -\frac{1}{P_\psi^{k,reg}(-q)} & 0 \end{pmatrix}. \quad (2.141)$$

Both immediately follow from $\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} 0 & A \\ B & 0 \end{pmatrix} \begin{pmatrix} 0 & B^{-1} \\ A^{-1} & 0 \end{pmatrix}$. Alternatively, one may just take the results, eqs. (2.130) to (2.134), and set all fields η to zero.

Back to eq. (2.121) we recognize to have two contributions to $\left. \frac{\delta \Gamma_k^{(2)}}{\delta \varphi(q)} \right|_{\eta=0}$: One is received from H_φ , eq. (2.119), and another from $U_k^{(2)}$, eqs. (2.111) and (2.112). But a closer look to the latter term leads us to state that it does not contribute. In fact, U_k is a function of $\rho = \phi^* \phi$ to ensure conservation of the total number of particles (cf. eq. (2.38)) and therefore every odd number of derivatives either directly vanishes or consists of a sum of terms that contain at least one bosonic field $\phi^{(*)}$. Setting all fields to

zero ($\phi = 0$) beats down such contributions in the end²⁸. Thus we explicitly have

$$\Gamma_{q,ab}^{(3)}(q_1, q_2) \equiv \left. \frac{\delta \Gamma_k^{(2)}}{\delta \varphi(q)} \right|_{\eta=0} = (2\pi)^{-2D} \begin{pmatrix} 0 & \vdots & & & \\ \vdots & 0 & 0 & & \\ \vdots & \vdots & \vdots & \ddots & \\ \vdots & 0 & -h_\varphi \epsilon & & \end{pmatrix} \delta(q_1 - q_2 - q) \quad (2.142)$$

and similarly

$$\Gamma_{q,ab}^{(3^*)}(q_1, q_2) \equiv \left. \frac{\delta \Gamma_k^{(2)}}{\delta \varphi^*(q)} \right|_{\eta=0} = (2\pi)^{-2D} \begin{pmatrix} 0 & \vdots & & & \\ \vdots & h_\varphi \epsilon & 0 & & \\ \vdots & \vdots & \vdots & \ddots & \\ \vdots & 0 & 0 & & \end{pmatrix} \delta(q_1 - q_2 + q) \quad (2.143)$$

The remaining quantity to derive reads

$$\Gamma_{k,ab}^{(4)}(q_1, q_2) \equiv \left. \frac{\delta^2 \Gamma_k^{(2)}}{\delta \varphi^*(q) \delta \varphi(q)} \right|_{\eta=0} = (2\pi)^{-3D} 2 U_k''|_{\phi=0} \begin{pmatrix} 0 & 1 & \vdots & & \\ \vdots & 0 & \vdots & \ddots & \\ \vdots & \vdots & \vdots & \vdots & \ddots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & 0 \end{pmatrix} \delta(q_1 - q_2 + q - q) \quad (2.144)$$

where now just $U_k^{(2)}$ contributes and by having in mind the structure of U_k we conclude that $\frac{\partial^{2n} U_k}{\partial \phi^{*n} \partial \phi^n}$ does not vanish after setting the field contents to zero.

Back to eq. (2.138) we have to evaluate

$$\begin{aligned} \text{STr } \Gamma_k^{(4)} G_k^0 &= \int_{q, q'} \zeta_a \Gamma_{k,ab}^{(4)}(q, q') G_{k,ba}^0(q', q) \\ &= (2\pi)^{-2D} V_D \left(2 U_k''|_{\rho=0} \right) \int_q \left[\frac{1}{U_k'|_{\rho=0} + P_\varphi^{k,reg}(q)} + \frac{1}{U_k'|_{\rho=0} + P_\varphi^{k,reg}(-q)} \right] \end{aligned} \quad (2.145)$$

and

$$\begin{aligned} \text{STr } \Gamma_q^{(3)} G_k^0 \Gamma_q^{(3^*)} G_k^0 &= \sum_{a,b,c,d} \zeta_a \int_{q_1, q'} \delta(q_1 - q' - q) \delta(q' - q_1 + q) \Gamma_{ab}^{(3)} G_{k,bc}^0(q') \Gamma_{cd}^{(3^*)} G_{k,da}^0(q_1) \\ &= -(2\pi)^{-2D} V_D \int_{q'} \sum_{FF'F''F'''} \Gamma_{FF'}^{(3)} G_{k,FF''}^0(q') \Gamma_{F''F'''}^{(3^*)} G_{k,F'''}^0(q + q') \end{aligned} \quad (2.146)$$

²⁸From a more formal perspective we may consider $\left. \frac{\partial^{m+n} U_k}{\partial \phi^{*m} \partial \phi^n} \right|_{\phi=0}$ assuming $n \geq m$ w.l.o.g. since we can *complex conjugate* our result, i.e. $n \leftrightarrow m$ and $\phi \leftrightarrow \phi^*$, to obtain the corresponding expression for $m > n$. Accounting for Leibniz rule²⁹ of the m -th derivative of a product we have $\frac{\partial^{m+n} U_k}{\partial \phi^{*m} \partial \phi^n} = \frac{\partial^m}{\partial \phi^{*m}} U_k^{(n)} \phi^{*n} = \phi^{*n-m} \sum_{r=0}^m \binom{m}{r} \frac{n!}{(n-r)!} \rho^{m-r} U_k^{(n+m-r)}$ and for $m > n$ we just replace the prefactor in front of the sum by ϕ^{m-n} and exchange m and n in the summation as well. For a term that does not explicitly contain $\phi^{(*)}$ we have to require $m = n$, i.e. the *degree of the derivative* $m+n = 2n$ must be even. Furthermore we need to have $r = m$ ($r = n$) for $\rho^0 = 1$ and hence we write $\left. \frac{\partial^{m+n} U_k}{\partial \phi^{*m} \partial \phi^n} \right|_{\phi=0} = n! U_k^{(n)}|_{\phi=0} \delta_{mn}$.

²⁹A heuristic proof to Leibniz's rule might work as follows: Consider the product $f(x)g(y)$; The operator $(\partial_x + \partial_y)$ mimics the product rule as $(\partial_x + \partial_y)fg = f'g + fg'$ and we obtain $(\partial_x + \partial_y)^n fg = \sum_{m=0}^n \binom{n}{m} \partial_x^m \partial_y^{n-m} (fg) = \sum_{m=0}^n \binom{n}{m} f^{(m)} g^{(n-m)}$ which reduces to the rule when setting $x = y$.

where the (discrete) matrix multiplication under the integral reads

$$-h_\varphi^2 \begin{pmatrix} 0 & 0 \\ 0 & -\epsilon \end{pmatrix} \begin{pmatrix} 0 & 1 \frac{1}{P_\psi^{k,reg}(q')} \\ -1 \frac{1}{P_\psi^{k,reg}(-q')} & 0 \end{pmatrix} \begin{pmatrix} \epsilon & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 1 \frac{1}{P_\psi^{k,reg}(q'+q)} \\ -1 \frac{1}{P_\psi^{k,reg}(-q'-q)} & 0 \end{pmatrix} \quad (2.147)$$

and its trace is

$$+ \frac{h_\phi^2}{P_\psi^{k,reg}(-q')P_\psi^{k,reg}(q'+q)} \text{Tr} \begin{pmatrix} 0 & \\ & -\epsilon^2 \end{pmatrix} = \frac{2h_\varphi^2}{P_\psi^{k,reg}(-q')P_\psi^{k,reg}(q'+q)} \quad . \quad (2.148)$$

In eq. (2.146) we used the fact that according to eqs. (2.140), (2.142) and (2.143) the bosonic and fermionic sector are completely decoupled and $\Gamma^{(3)}$ does not carry any bosonic contribution.

Summing up, we have the flow of the (modified) bosonic propagator as

$$\begin{aligned} \partial_k \tilde{P}_\varphi^k(q) &= U_k''|_{\rho=0} \tilde{\partial}_k \int_{q'} \left[\frac{1}{\tilde{P}_\varphi^{k,reg}(q')} + \frac{1}{\tilde{P}_\varphi^{k,reg}(-q')} \right] \\ &\quad - h_\varphi^2 \tilde{\partial}_k \int_{q'} \frac{1}{P_\psi^{k,reg}(q')P_\psi^{k,reg}(q-q')} \end{aligned} \quad (2.149)$$

with

$$\tilde{P}_\varphi^{k,reg}(q) = P_\varphi^{k,reg}(q) + U_k'|_{\rho=0} \quad . \quad (2.150)$$

Reiterating the same conceptual steps for the flow of the fermionic propagator will yield the corresponding flow of P_ψ^k . It is at this stage where our additional notation, eqs. (2.6) and (2.11), become useful. Moreover it will be convenient to use the (standard) representation of the (Hermitian and traceless) *Pauli matrices*

$$\sigma_1 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad \sigma_2 = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad \sigma_3 = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \quad \text{and}^{30} \quad \sigma_\pm \equiv \frac{1}{2}(\sigma_1 \pm i\sigma_2), \quad 1^\pm \equiv \frac{1}{2}(1 \pm \sigma_3) \quad . \quad (2.151)$$

As an appropriate projection prescription we choose

$$\begin{aligned} \partial_k P_\psi^k(q) &= \frac{1}{2}(2\pi)^{2D} V_D^{-1} \tilde{\partial}_k \frac{\delta^2}{\delta\psi_\uparrow(q)\delta\psi_\uparrow^*(q)} \text{STr} \ln G_k^{-1} \Big|_{\eta=0} \\ &= \frac{1}{2}(2\pi)^{2D} V_D^{-1} \tilde{\partial}_k \frac{\delta}{\delta\psi_\uparrow(q)} \text{STr} \frac{\delta\Gamma_k^{(2)}}{\delta\psi_\uparrow^*(q)} G_k \Big|_{\eta=0} \\ &= 0 - \frac{1}{2}(2\pi)^{2D} V_D^{-1} \tilde{\partial}_k \text{STr} \left[-\Gamma_{q_\uparrow}^{(3*)} - G_k^0 \Gamma_{q_\uparrow}^{(3)} G_k^0 \right], \end{aligned} \quad (2.152)$$

where the explicit 0 indicates that our truncation does not include a *4-fermion coupling* $\Gamma_k^{(4)}$ that may

³⁰We could also have written the former (2-dimensional) antisymmetric tensor as $\epsilon = i\sigma_2$.

contribute as \bigcirc to the flow of the fermionic inverse propagator P_ψ^k . Furthermore we defined

$$-\Gamma_{q_\uparrow, ab}^{(3*)}(q_1, q_2) \equiv \left. \frac{\delta \Gamma_k^{(2)}}{\delta \psi_\uparrow^*(q)} \right|_{\eta=0} = \frac{h_\varphi}{(2\pi)^{2D}} \begin{pmatrix} 0 & -\sigma_\pm \\ \sigma_- & 0 \end{pmatrix} \delta(q_1 - q_2 + q) \quad (2.153)$$

$$\Gamma_{q_\uparrow, ab}^{(3)}(q_1, q_2) \equiv \left. \frac{\delta \Gamma_k^{(2)}}{\delta \psi_\uparrow(q)} \right|_{\eta=0} = \frac{h_\varphi}{(2\pi)^{2D}} \begin{pmatrix} 1^- & 0 \\ -1^- & 0 \end{pmatrix} \delta(q_1 - q_2 - q) \quad , \quad (2.154)$$

where in this case our newly introduced notation does not affect the first equality since the additional minus sign appears for the vanishing entries of $\Gamma_{q_\uparrow}^{(3*)}$ only. Furthermore the propagators in eq. (2.152) read

$$\pm G_{k, ab}^0(q_1, q_2) = (2\pi)^{3D} \begin{pmatrix} G_{k, BB'}^0(q_1) & 0 \\ 0 & G_{k, FF'}^0(q_1) \end{pmatrix} \delta(q_1 - q_2) \quad , \quad (2.155)$$

i.e. they stay unchanged compared to eq. (2.139). Thus the immediate task is to explicitly evaluate

$$-\frac{1}{2}(2\pi)^{2D} V_D^{-1} \tilde{\delta}_k h_\varphi^2 \int_{q', q''} \delta(q'' - q' + q) \delta(q' - q'' - q) \times \\ \text{sTr} \begin{pmatrix} 0 & -\sigma_\pm \\ \sigma_- & 0 \end{pmatrix} \begin{pmatrix} G_\varphi^0(q') & \\ & G_\psi^0(q') \end{pmatrix} \begin{pmatrix} 1^- & 0 \\ -1^- & 0 \end{pmatrix} \begin{pmatrix} G_\varphi^0(q'') & \\ & G_\psi^0(q'') \end{pmatrix} \quad ,$$

with an obvious abbreviation for the (discrete) propagator matrices $G_\varphi^0 \equiv G_{k, BB'}^0$ and $G_\psi^0 \equiv G_{k, FF'}^0$. In a first step of simplification we may write

$$-\frac{h_\varphi^2}{2} \tilde{\delta}_k \int_{q'} \text{sTr} \begin{pmatrix} -(0 \ \sigma_+) G_\psi^0(q') & \\ \begin{pmatrix} 0 \\ \sigma_- \end{pmatrix} G_\varphi^0(q') & \end{pmatrix} \begin{pmatrix} (1^- \ 0) G_\psi^0(q' - q) & \\ - \begin{pmatrix} 1^- \\ 0 \end{pmatrix} G_\varphi^0(q' - q) & \end{pmatrix} \quad . \quad (2.156)$$

Converting the (discrete) super-trace into the (ordinary) trace Tr takes us to

$$+\frac{h_\varphi^2}{2} \tilde{\delta}_k \int_{q'} \text{Tr} \left\{ \begin{pmatrix} 0 \\ 1 \end{pmatrix} \sigma_- G_\varphi^0(q') (1^- \ 0) G_\psi^0(q' - q) - \sigma_+ (0 \ 1) G_\psi^0(q') \begin{pmatrix} 1^- \\ 0 \end{pmatrix} G_\varphi^0(q' - q) \right\} \quad . \quad (2.157)$$

We have

$$\begin{aligned} \sigma_- G_\varphi^0(q) &= 1^- / \tilde{P}_\varphi^{k, reg}(q) & 1^- G_\varphi^0(q) &= \sigma_- / \tilde{P}_\varphi^{k, reg}(-q) \\ (1^- \ 0) G_\psi^0(q) &= (0 \ 1^-) / P_\psi^{k, reg}(q) & (0 \ 1) G_\psi^0(q) &= -(1 \ 0) / P_\psi^{k, reg}(-q) \end{aligned} \quad (2.158)$$

and hence the Tr in eq. (2.157) simplifies to

$$\frac{1}{\tilde{P}_\varphi^{k, reg}(q') P_\psi^{k, reg}(q' - q)} \text{Tr} \left\{ \begin{pmatrix} 0 \\ 1 \end{pmatrix} (0 \ 1^-) + (\sigma_+ \ 0) \begin{pmatrix} \sigma_- \\ 0 \end{pmatrix} \right\} \quad (2.159)$$

after an appropriate redefinition of the *loop-momentum* q' (cf. eq. (2.160))³¹. Evaluating the Tr operation simply yields a factor of 2 due to $1^- \otimes 1^-$ for the first matrix and $\sigma_+ \sigma_- = 1^\dagger$ for the second one under the trace. Therefore the flow of the fermionic inverse propagator reads

$$\partial_k P_\psi^k(q) = h_\varphi^2 \tilde{\delta}_k \int_{q'} \frac{1}{\tilde{P}_\varphi^{k,reg}(q') P_\psi^{k,reg}(q' - q)} \sim \psi_{\uparrow\downarrow}(q) \begin{array}{c} \varphi(q') \\ \circlearrowleft \\ \psi_{\downarrow\uparrow}(q' - q) \end{array} \psi_{\uparrow\downarrow}(q) \quad (2.160)$$

with the $\tilde{P}_\varphi^{k,reg}(q)$ from eq. (2.150).

³¹Since q' is integrated over from minus to plus infinity, a linear shift $q' \rightarrow q' + q$ and reflections $q' \rightarrow -q'$ do not affect the measure of $\int_{q'}$. At finite temperature T those manipulations are still valid in virtue of the same equidistant spacing for bosonic and fermionic Matsubara frequencies, namely $2\pi T$, as well as their reflection property: $\omega_{-n} = -\omega_n$ (bosons) and $\omega_{-(n+1)} = -\omega_n$ (fermions).

2.3 The Story Revised: Flowing Into the Broken Phase

When previously deriving the flow of the effective potential $U_k(\rho)$ as well as the particle's (inverse) propagators \tilde{P}_φ^k and P_ψ^k , we restricted our result to the situation of vanishing field expectation values $\eta = 0$. But in the end we are interested in *spontaneous symmetry breaking* (SSB) manifest by a bosonic *ground state* $\varphi(x) = \phi$ that is spatially homogeneous due to translational invariance of the problem. Since the (flowing) quantum action $\Gamma_k[\eta]$ represents a functional that takes a given field configuration $\eta = \eta_a(x)$ mapping it to the manifold of real numbers (as an *action* does), we should take revision on our projection prescription of the inverse propagators:

By definition, the full propagator G_k is a functional that is obtained from the inverse of the second derivative of the effective action Γ_k (cf. eqs. (2.108) and (2.128)). Therefore one may view it as an (infinite-dimensional) matrix that depends on the fields (cf. eq. (2.119)), i.e. one has to evaluate G_k according to a given field configuration $\eta_a(x)$. As we are mainly interested in SSB it is reasonable to study the flow equations at $\eta_a(x) = (\phi, \phi^*, 0, 0, 0, 0) \equiv \Phi$ with constant (real)³² bosonic *condensate*

$$\phi = \phi^* = \sqrt{\rho} = \text{const.} > 0 \quad (\text{SSB phase}) \quad (2.161)$$

and vanishing fermionic field expectation value $\psi_{\uparrow\downarrow}(x) = 0$ (due to the Pauli exclusion principle fermions are not allowed to condensate). To this end, G_k from eq. (2.130) has to be used for the projection of the flow to the (inverse) fermionic and bosonic propagators and thus the previously derived equations, namely eq. (2.149) and eq. (2.160), are special cases where the physical ground state fulfills $\phi = 0$. This situation is commonly referred to as *symmetric phase* (SYM).

By means of the *quantum equation of motion* $\frac{\delta\Gamma}{\delta\eta} = 0$, SSB is determined by the (global) extremum ρ_0 of the effective potential $U(\rho)$ when restricting to the translationally invariant situation of the ground state. Expanding this principle to the flowing quantum action Γ_k , consistency requires evaluating the rhs. of the flow equation at the corresponding ρ_0 that is determined by the (flowing) effective potential $U_k(\rho)$. Adopting this paradigm we have to generalize the flow equations for $P_{\varphi/\psi}^k$. Moreover, our argumentation for the vanishing flow of the Yukawa coupling h_φ (cf. eq. (2.62)) is not valid any more. The reason is discussed in the [item 3](#) and traces back to the non-diagonal structure of the propagator G_k in field space as apparent from eqs. (2.132) and (2.133). In the following we provide the more general flow equations for the inverse propagators and h_φ which reduce to the previously computed ones when setting $\phi = 0$. Furthermore, we introduce a diagrammatic approach that allow for a discussion of flow equations in the SSB phase. Of course, all one-loop diagrams properly generalize our notation from e.g. [item 3](#), i.e. they reduce to the SYM phase diagrams when $\phi \rightarrow 0$.

However, compared to eqs. (2.142) to (2.144) the bosonic contribution of the discrete matrix structure becomes non-zero:

$$\Gamma_{BB'}^{(3)} = \begin{pmatrix} \alpha^* & \beta^* \\ \beta^* & \beta \end{pmatrix}, \quad \Gamma_{BB'}^{(3*)} = \begin{pmatrix} \beta^* & \beta \\ \beta & \alpha \end{pmatrix} \quad \text{and} \quad \Gamma_{BB'}^{(4)} = \begin{pmatrix} \gamma^* & d \\ d & \gamma \end{pmatrix} \quad (2.162)$$

³²One may choose an (unimportant) phase angle α for the constant field $\phi = \sqrt{\rho}e^{i\alpha} = \text{const.}$ where $\rho = \phi\phi^*$, but for simplicity we set it to zero.

with corresponding *condensate couplings*

$$\alpha \equiv U_k''' \phi^3, \quad \beta \equiv 2U_k'' \phi + U_k''' \rho \phi, \quad \gamma \equiv 3U_k''' \phi^2 + U_k^{(4)} \phi^2 \rho \quad \text{and} \quad d = d^* \equiv 2U_k'' + 4U_k''' \rho + U_k^{(4)} \rho^2 \quad (2.163)$$

whose diagrammatic representation is discussed in detail below³³. For the moment we just note that all bosonic vertex contributions vanish in the symmetric phase, except for the part of d which does not couple to the condensate ϕ .

Concerning the propagators from eqs. (2.131) to (2.134), it appears to be convenient to use the property $P^*(q) = P(-q)$ of translationally invariant inverse propagators P (cf. eq. (2.106)). Therefore, the full bosonic and fermionic propagators (evaluated at Φ) read

$$G_{k,BB'}(q) = \frac{1}{|G_B^{-1}(q)|} \begin{pmatrix} -\theta & [\tilde{P}_\varphi^{k,reg}(q)]^* \\ \tilde{P}_\varphi^{k,reg}(q) & -\theta^* \end{pmatrix} \quad \text{with} \quad |G_B^{-1}(\pm q)| = |\tilde{P}_\varphi^{k,reg}(q)|^2 - |\theta|^2 \quad (2.164)$$

where

$$\theta \equiv U_k'' \phi^2 \quad \text{as well as} \quad \tilde{P}_\varphi^{k,reg}(q) \equiv U_k' + \rho U_k'' + P_\varphi^{k,reg}(q) \quad (2.165)$$

and

$$G_{k,FF'}(q) = \frac{1}{|G_F^{-1}(q)|} \begin{pmatrix} -\epsilon h_{\varphi,k} \phi & 1 [P_\psi^{k,reg}(q)]^* \\ -1 P_\psi^{k,reg}(q) & \epsilon h_{\varphi,k} \phi^* \end{pmatrix} \quad \text{with} \quad |G_F^{-1}(\pm q)| = |P_\psi^{k,reg}(q)|^2 + h_{\varphi,k}^2 \rho \geq 0. \quad (2.166)$$

The Yukawa coupling $h_{\varphi,k}$ —as we will see—will become scale dependent in the SSB phase. We note, that the fermionic determinant $|G_F^{-1}(q)|$ is always non-zero for a properly regularized fermionic (inverse) propagator, i.e. $P_\psi^{k,reg}(q) \neq 0$ for all q . In particular the *Fermi surface* $P_\psi^{k,reg}(q)|_{q_0=0} \stackrel{!}{=} 0$ has to be modified by an appropriate regulator function $R_k(q)$ (at zero temperature $T = 0$). A discussion on that issue can be found in [section 3.3.3](#).

A quick calculation also paves the way to a condition for a non-vanishing of the (negative) bosonic determinant $-|G_B^{-1}(q)|$: Taking into account that we would like to evaluate the rhs. of the flow equations at the minimum of the effective potential $U_k(\rho)$ for the SSB phase ($\rho \neq 0$), we set $U_k'(\rho_0) = 0$ and assume $U_k''(\rho_0) \geq 0$. Therefore, the (negative of the) bosonic determinant reduces to

$$|G_B^{-1}(q)| = 2\rho_0 U_k''(\rho_0) \Re P_\varphi^{k,reg}(q) + |P_\varphi^{k,reg}(q)|^2 \quad (2.167)$$

and we need to have a positive real part of the regularized, bosonic inverse propagator, i.e. $\Re P_\varphi^{k,reg}(q) > 0$. If it would vanish, $\Im P_\varphi^{k,reg}(q) \neq 0$ for all q which can not be satisfied due to the anti-symmetric property, eq. (2.107).

At this point we would like to properly introduce the graphical notation we used e.g. in eq. (2.160). Due to space-time translational invariance that is reflected in appropriate δ -distributions for the propagator $G_{k,ab}(q_1, q_2)$ (cf. eq. (2.130)) and vertices $\frac{\delta^n}{\delta \eta^1(q) \delta \eta^2(q') \dots \delta \eta^n(q^{(n)})} \Gamma_{k,ab}^{(2)}(q_1, q_2)$ (cf. e.g. eqs. (2.153) and (2.154)) the

³³As previously defined we stick to the convenient abbreviation $U_k^{(n)} \equiv \partial_\rho^n U_k(\rho)$.

graphical representation of propagators just carry one momentum index q and vertices $(n + 1)$ q -labels, respectively. Moreover we have to represent the discrete field indices where for each η_a there exists an η_a^* . Remember that we grouped those fields by $\eta_{\bar{a}}$ and the graphical notation will be similarly arranged. More precisely, we label propagator and vertex lines by $\eta_{\bar{a}}$ and an arrow on the line will indicate if it refers to η_a^* or η_a which may be associated to particle creation and destruction³⁴, respectively:

$$G_{\eta\eta^*}(q) \sim \begin{array}{c} \xrightarrow{\eta(q)} \\ \hline \end{array} \sim \frac{\tilde{P}_\eta^*(q)}{|G_\eta^{-1}(q)|} \xrightarrow{\text{SYM}} \frac{1}{P_\eta(q)} \quad (2.168)$$

$$G_{\eta\eta}(q) \sim \begin{array}{c} \xleftarrow{\eta(q)} \\ \textcircled{\times} \\ \downarrow \\ \textcircled{\times} \mathcal{C} \end{array} \sim \frac{\mathcal{C}}{|G_\eta^{-1}(q)|} \xrightarrow{\text{SYM}} 0 \quad , \quad (2.169)$$

where \mathcal{C} is related to the *condensate* of the SSB phase which vanishes when switching to the SYM phase. Similarly, $\tilde{P}_\eta(q)$ represents the modified momentum structure of the inverse propagator $P_\eta(q)$ in the presence of the condensate with $\tilde{P}_\eta(q) \xrightarrow{\text{SYM}} P_\eta(q)$. Note that $|G_\eta^{-1}(q)| \xrightarrow{\text{SYM}} |P_\eta(q)|^2$ and the wiggled line in eq. (2.169) in combination with the *blobby* vertex represents some condensate coupling that is specified in detail by the quantity \mathcal{C} . In particular, if referring back to eqs. (2.164) and (2.165) for bosons we have $\mathcal{C} = U''\phi^2$. Therefore:

$$G_{\varphi\varphi^*} \sim \begin{array}{c} \xrightarrow{\varphi(q)} \\ \text{---} \end{array} \quad , \quad G_{\varphi^*\varphi} \sim \begin{array}{c} \xleftarrow{\varphi(q)} \\ \text{---} \end{array} \quad , \quad G_{\varphi\varphi} \sim \begin{array}{c} \xrightarrow{\varphi(q)} \\ \bullet \\ \swarrow \searrow \\ \textcircled{\times} \phi \quad \textcircled{\times} \phi \end{array} \quad , \quad G_{\varphi^*\varphi^*} \sim \begin{array}{c} \xleftarrow{\varphi(q)} \\ \bullet \\ \swarrow \searrow \\ \phi \textcircled{\times} \quad \phi \textcircled{\times} \end{array} \quad (2.170)$$

and equivalent diagrams are obtained for fermions where one has $\mathcal{C} = h_\varphi\phi$ as obvious from eq. (2.166):

$$G_{\psi_{\uparrow\downarrow}\psi_{\uparrow\downarrow}^*} \sim \begin{array}{c} \xrightarrow{\psi_{\uparrow\downarrow}(q)} \\ \text{---} \end{array} \quad , \quad G_{\psi_{\uparrow\downarrow}^*\psi_{\uparrow\downarrow}} \sim \begin{array}{c} \xleftarrow{\psi_{\uparrow\downarrow}(q)} \\ \text{---} \end{array} \quad , \quad G_{\psi_{\uparrow\downarrow}\psi_{\uparrow\downarrow}} \sim \begin{array}{c} \xrightarrow{\psi_{\uparrow\downarrow}(q)} \\ \bullet \\ \downarrow \\ \textcircled{\times} \phi \end{array} \quad , \quad G_{\psi_{\uparrow\downarrow}^*\psi_{\uparrow\downarrow}^*} \sim \begin{array}{c} \xleftarrow{\psi_{\uparrow\downarrow}(q)} \\ \bullet \\ \downarrow \\ \phi \textcircled{\times} \end{array} \quad (2.171)$$

Of course, there is impact of the condensate on vertices as well. In general we would like to define *ordinary* couplings $\Gamma^{(n)}$ and *condensate* couplings $\Gamma_{\mathcal{C}}^{(n)}$ ($n > 2$) with the suggestive property $\Gamma_{\mathcal{C}}^{(n)} \xrightarrow{\text{SYM}} 0$.

In our specific system we identify ($n = 3, 4$):

$$\Gamma_{\varphi^*\psi_{\uparrow\downarrow}\psi_{\uparrow\downarrow}}^{(3)} \sim \begin{array}{c} \psi_{\uparrow\downarrow}(q_2) \\ \swarrow \\ \bullet \\ \searrow \\ \psi_{\uparrow\downarrow}(q_1) \end{array} \xrightarrow{h_\varphi} \varphi(q_1+q_2) \quad \text{and} \quad \Gamma_{\varphi^*\varphi^*\varphi\varphi}^{(4)} \sim \begin{array}{c} \varphi(q_2) \quad \varphi(q_2) \\ \swarrow \quad \searrow \\ \bullet \\ \swarrow \quad \searrow \\ \varphi(q_1) \quad \varphi(q_1) \end{array} \xrightarrow{2U''} \quad , \quad (2.172)$$

cf. eqs. (2.142) to (2.144), and as apparent from eqs. (2.162) and (2.163) the SSB phase reveals further

³⁴Recall that in virtue of the coherent state path integral there is an identification between annihilation (creation) operators $a^{(+)}$ and (complex conjugate) fields $\eta^{(*)}$ by $\eta^{(*)} \leftrightarrow a^{(+)}$.

coupling contributions. Exemplarily we depict the second summand from β and d , respectively:

$$\Gamma_{\mathcal{C}\varphi^*\varphi^*\varphi}^{(3)} \sim \begin{array}{c} \varphi(q_1) \text{ ---} \\ \text{---} \varphi(q_2) \\ \text{---} \varphi(q_1 - q_2) \\ \text{---} \varphi(q_2) \\ \text{---} \phi \\ \text{---} \phi \\ \text{---} \phi \end{array} U''' \quad \text{and} \quad \Gamma_{\mathcal{C}\varphi^*\varphi^*\varphi\varphi}^{(4)} \sim \begin{array}{c} \varphi(q_2) \text{ ---} \\ \text{---} \varphi(q_1) \\ \text{---} \varphi(q_1) \\ \text{---} \phi \\ \text{---} \phi \end{array} 4U''' \quad . \quad (2.173)$$

For convenience, we introduced the shorthand notation $\Gamma_{\eta_{a_1} \dots \eta_{a_n}}^{(n)} \equiv \frac{\delta^n}{\delta \eta_{a_1} \dots \delta \eta_{a_n}} \Gamma$ and dropped the momenta in addition. Note, that condensate particles ϕ do not transfers any momentum q . In the process to the left from eq. (2.173) a (non-condensed) boson scatters off a condensate particle by transferring momentum $q_1 - q_2$ to it. The diagram to the right shows that the condensate can also *passively* contribute to scattering of non-condensed bosons.

Now, due to the generic one-loop structure of renormalization flow diagrams³⁵ all flowing quantities have to be constructed from the elements discussed above. Technically, the instruction reads: Appropriately concatenate propagators and vertices into a loop (trace) by ensuring that one (full) propagator connects two (full) vertices and the *external legs* of the corresponding one-loop diagram fit to the quantity whose flow is to be considered. Directly concatenating two vertices (propagators) is not allowed due to the generic structure of the flow equations:

$$\partial_k \Gamma_k^{(n \geq 1)} \sim \tilde{\partial}_k \text{STr} \prod_i [G_k \Gamma_k^{(m_i)}] \quad \text{with} \quad 2 \leq m_i \leq n + 2 \quad (\text{cf. eq. (2.30)}) \quad . \quad (2.174)$$

As we will explicitly show in a minute, the Yukawa coupling will flow in the SSB phase and in accordance with our result, eq. (2.64), $\partial_k h_\varphi = 0$ for the SYM phase all contributing diagrams contain propagators and vertices that vanish in the limit $\phi \rightarrow 0$. One of the terms that contribute reads

$$\partial_k h_{\varphi,k} \propto h_{\varphi,k}^5 \rho^2 U_k'' \tilde{\partial}_k \int_q \frac{1}{|G_F^{-1}(q)|^2 |G_B^{-1}(q)|} \sim \tilde{\partial}_k \begin{array}{c} \psi_{\uparrow\downarrow}(0) \text{ ---} \\ \text{---} \psi_{\uparrow\downarrow}(q) \\ \text{---} \psi_{\uparrow\downarrow}(q) \\ \text{---} \psi_{\uparrow\downarrow}(q) \\ \text{---} \psi_{\uparrow\downarrow}(q) \\ \text{---} \psi_{\uparrow\downarrow}(0) \end{array} \begin{array}{c} \phi \\ \phi \\ \phi \\ \phi \\ \phi \end{array} \begin{array}{c} \psi_{\uparrow\downarrow}(q) \\ \psi_{\uparrow\downarrow}(q) \\ \psi_{\uparrow\downarrow}(q) \\ \psi_{\uparrow\downarrow}(q) \\ \psi_{\uparrow\downarrow}(q) \end{array} \begin{array}{c} \phi \\ \phi \\ \phi \\ \phi \\ \phi \end{array} \begin{array}{c} h_{\varphi,k} \\ h_{\varphi,k} \\ h_{\varphi,k} \\ h_{\varphi,k} \\ h_{\varphi,k} \end{array} U_k'' \begin{array}{c} \phi \\ \phi \\ \phi \\ \phi \\ \phi \end{array} \varphi(0) \xrightarrow{\text{SYM}} 0 \quad . \quad (2.175)$$

Note, that our truncation did not allow for the momentum resolution of $h_{\varphi,k}$ and therefore we projected to the term $h_{\varphi}\varphi^*(0)\psi_{\downarrow}(0)\psi_{\uparrow}(0)$ (cf. eq. (2.186)) from our ansatz for the effective action Γ_k (cf. eq. (2.88)) in favor of some term that carries external momenta. As apparent by diagrammatic reasoning the process that drives the flow of the Yukawa coupling involves two (composite) condensate particles that split up into their fermionic constituents to provide scattering partners for the incoming fermions on the one hand and ones to rejoin for the outgoing boson with vanishing momentum on the other hand. Finally, the remaining two bosons scatter back into the condensate. Therefore this particular process provides a mechanism to scatter spin-up/spin-down fermions into the condensate. Equally, the condensate may loose particles by the reverse process. However, due to momentum conservation it is straightforward

³⁵Cf. section 2.1.3, and the important equation eq. (2.30) from section 2.1.2.

to extend eq. (2.175) to a general momentum structure of the external legs. But for us, this would (unnecessary) blow up the momentum integration by external labels q_1 and q_2 . One lesson to take is the fact that the condensate particles provide a way to contribute to the flow of certain quantities, but in the end—by virtue of particle number conservation—each participating condensate particle that enters the loop has to appropriately leave it again. Metaphorically speaking, the condensate provides a *medium* in which some scattering process takes place and therefore it *passively* affects its physics³⁶. Another contribution to the flow of $h_{\varphi,k}$ in the SSB phase may be constructed according to the diagram

$$\psi_{\uparrow\downarrow}(0) \quad \psi_{\uparrow\downarrow}(q) \quad \phi \quad \psi_{\uparrow\downarrow}(q) \quad \psi_{\uparrow\downarrow}(0) \quad \varphi(q) \quad h_{\varphi k} \quad h_{\varphi k} \quad h_{\varphi k} \quad \varphi(0) \sim h_{\varphi,k}^3 \rho \xrightarrow{\text{SYM}} 0 \quad . \quad (2.176)$$

$\times \beta^* = 2U_k'' \phi^* + U_k''' \rho \phi^*$

Note, that compared to eq. (2.175), eq. (2.176) involves a condensate coupling β^* besides propagator contributions emerging from the condensate \mathcal{C} . It takes one of the two bosons that had been created with the aid of a condensate particle that previously split up to provide corresponding scattering partners to the incoming fermions while the other one constitutes the outgoing boson. Obviously, U_k'' and U_k''' may be associated with appropriate coupling constants. As apparent from the additional factor ρ , another condensate particle is involved in the latter case. To this end the process is at least proportional to the condensate density and therefore it vanishes in the SYM phase. By the way, the coupling $\alpha^{(*)}$ is principally not allowed to play any role here, since it does not couple to a vertex with two ingoing and one outgoing leg. Nevertheless it is possible to construct further diagrams for the flow of the Yukawa coupling that involves bosonic condensate propagators instead of $---\blacktriangleright---$ we depicted here (cf. eq. (2.170)). For the complete set of diagrams that drive the renormalization of the Yukawa coupling we refer to eq. (2.193).

Back to our initial aim, the derivation of the flow equations for the BCS–BEC crossover in the SSB phase, we would like to extend the renormalization of the bosonic (inverse) propagator first. Compared to eq. (2.138) we now project to a finite condensate ϕ and thus for non-vanishing momentum q we have (cf. our truncation, eq. (2.88))³⁷

$$\partial_k \tilde{P}_\varphi^k(q) = \frac{1}{2} \tilde{\partial}_k \left\{ \text{STr} \Gamma_k^{(4)} G_k - \text{STr} \Gamma_{k,q}^{(3)} G_k \Gamma_{k,q}^{(3*)} G_k \right\} \Big|_{\eta=\Phi} \equiv \beta_{\varphi,B}^{(4)} + \beta_{\varphi,B}^{(3)}(q) + \beta_{\varphi,F}^{(3)}(q) \quad (q \neq 0) \quad . \quad (2.177)$$

While the first term introduces purely bosonic contributions the second one decouples into a bosonic and a fermionic β -function³⁸. Note, that $\beta_{\varphi,B}^{(4)}$ does not explicitly depend on the external momentum

³⁶Playing pool in the pub is certainly different from a match under water—aside from the technical requirement the players have to fulfill . . .

³⁷We dropped the $(2\pi)^D V_D$ -factor related to the Fourier transform since we previously checked that it does not enter the final flow equation. We will assume this notational convention from now on. When explicitly necessary we will appropriately reinsert it and stress its presence.

³⁸The notion of the β -function stems from (perturbative) renormalization where the change of (renormalized) couplings

of the inverse bosonic propagator and therefore it just provides an overall shift to each mode $q \neq 0$. Moreover, \tilde{P}_φ^k is not regularized which is in contrast to its emergence on the rhs. of the flow equation, i.e. within the context of the β -function. Becoming explicit, we state

$$\beta_{\varphi,B}^{(4)} = \tilde{\partial}_k \int_{q'} d \frac{\Re \tilde{P}_\varphi^{k,reg}(q') - \theta \gamma^*}{|G_B^{-1}(q')|} \xrightarrow{\text{SYM}} 2U_k'' \tilde{\partial}_k \int_{q'} \frac{\Re \tilde{P}_\varphi^{k,reg}(q')}{|\tilde{P}_\varphi^{k,reg}(q')|^2} = 2U_k'' \tilde{\partial}_k \int_{q'} \frac{1}{\tilde{P}_\varphi^{k,reg}(q')} \quad (2.178)$$

with appropriate notation from eqs. (2.163) to (2.165); in particular $\theta \gamma^* = U_4 U_2 \rho^3 + 3U_3 U_2 \rho^2 \in \mathbb{R}$ where we introduced the notational abbreviation

$$U_n \equiv U_k^{(n)} = \partial_\rho^n U_k(\rho) \quad . \quad (2.179)$$

We did not assign arrows to the loop and the condensate particles of the second diagram on purpose: Depending on the condensate coupling γ one has to appropriately arrange them in order to fulfill particle number conservation³⁹. E.g. two condensate particles enter the loop via the coupling U_2 and are scattered back into the condensate by γ .

Following the line, we similarly derive the fermionic contribution to the flow of the bosonic (inverse) propagator using our notation from eq. (2.166):

$$\beta_{\varphi,F}^{(3)}(q) = -h_{\varphi,k}^2 \tilde{\partial}_k \int_{q'} \frac{P_\psi^{k,reg}(q' - q) [P_\psi^{k,reg}(q')]^*}{|G_F^{-1}(q' - q)| |G_F^{-1}(q')|} \xrightarrow{\text{SYM}} -h_{\varphi}^2 \tilde{\partial}_k \int_{q'} \frac{1}{P_\psi^{k,reg}(q - q') P_\psi^{k,reg}(q')} \quad (2.180)$$

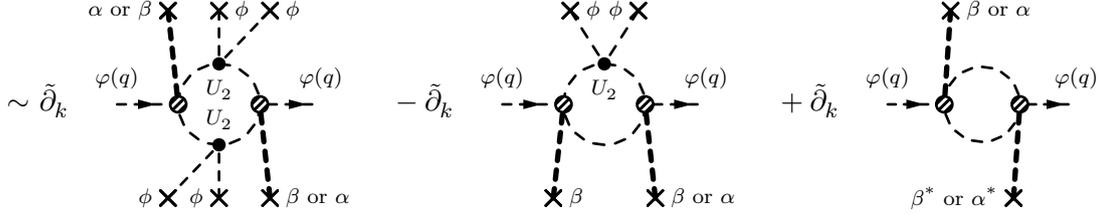
The remaining β -function identically vanishes in the SYM phase, i.e. it's diagrammatic representation

$\lambda_k = (\lambda_{1,k}, \lambda_{2,k}, \dots)$ depending on (momentum) scale k is described by a *velocity field* $\partial_k \lambda_k = \beta(\lambda_k)$. The label β must not be confused with the condensate coupling from eq. (2.163). But a distinction will be obvious from the context.

³⁹This condition is ensured by construction of the ansatz for the effective action Γ_k since it remains invariant under global *phase rotation* $\eta \rightarrow e^{i\alpha} \eta$.

purely contains condensate vertices and propagators. We obtain⁴⁰

$$\begin{aligned} \beta_{\varphi,B}^{(3)}(q) = & \mathcal{C}_1 \tilde{\partial}_k \int_{q'} \frac{1}{|G_B^{-1}(q')||G_B^{-1}(q'-q)|} - 2\mathcal{C}_2 \tilde{\partial}_k \int_{q'} \frac{\tilde{P}_\varphi^{k,reg}(q')}{|G_B^{-1}(q')||G_B^{-1}(q'+q)|} - 2\mathcal{C}_3 \tilde{\partial}_k \int_{q'} \frac{\tilde{P}_\varphi^{k,reg}(q')}{|G_B^{-1}(q')||G_B^{-1}(q'-q)|} \\ & + \frac{|\alpha|^2}{2} \tilde{\partial}_k \int_{q'} \frac{\tilde{P}_\varphi^{k,reg}(q')\tilde{P}_\varphi^{k,reg}(q'-q')}{|G_B^{-1}(q')||G_B^{-1}(q'-q)|} + \frac{|\beta|^2}{2} \tilde{\partial}_k \int_{q'} \frac{\tilde{P}_\varphi^{k,reg}(q')\tilde{P}_\varphi^{k,reg}(q'-q)+2[\tilde{P}_\varphi^{k,reg}(q')]^* \Re \tilde{P}_\varphi^{k,reg}(q'-q)}{|G_B^{-1}(q')||G_B^{-1}(q'-q)|} \end{aligned} \quad (2.181)$$



$$\text{with } \mathcal{C}_1 = |\beta|^2 |\theta|^2 + \Re(\alpha\beta\theta^{*2}), \quad \mathcal{C}_2 = \Re(\beta^2\theta^*) \quad \text{and} \quad \mathcal{C}_3 = \Re(\alpha^*\beta\theta) \quad (2.182)$$

which we may explicitly write by reinserting the corresponding expressions for the condensate couplings α , β and θ in terms of derivatives of the effective potential and powers of the condensate density ρ :

$$\begin{aligned} \beta_{\varphi,B}^{(3)}(q) = & -2 \left[(U_3 U_2)^2 \rho^5 + 3U_3 U_2^3 \rho^4 + 2U_2^4 \rho^3 \right] \tilde{\partial}_k \int_{q'} \frac{1}{|G_B^{-1}(q')||G_B^{-1}(q'-q)|} \\ & + 4 \left[U_3^2 U_2 \rho^4 + 2U_3 U_2^2 \rho^3 \right] \tilde{\partial}_k \int_{q'} \frac{\Re \tilde{P}_\varphi^{k,reg}(q')}{|G_B^{-1}(q')||G_B^{-1}(q'-q)|} \\ & + 4 \left[U_3 U_2^2 \rho^3 + 2U_2^3 \rho^2 \right] \tilde{\partial}_k \int_{q'} \frac{[\tilde{P}_\varphi^{k,reg}(q')]^*}{|G_B^{-1}(q')||G_B^{-1}(q'-q)|} \\ & - 2U_3^2 \rho^3 \tilde{\partial}_k \int_{q'} \frac{\Re \tilde{P}_\varphi^{k,reg}(q'-q) \Re \tilde{P}_\varphi^{k,reg}(q')}{|G_B^{-1}(q')||G_B^{-1}(q'-q)|} \\ & - 2 \left[U_3 U_2 \rho^2 + U_2^2 \rho \right] \tilde{\partial}_k \int_{q'} \frac{2\tilde{P}_\varphi^{k,reg}(q'-q) \Re \tilde{P}_\varphi^{k,reg}(q') + [\tilde{P}_\varphi^{k,reg}(q'-q) \tilde{P}_\varphi^{k,reg}(q')]^*}{|G_B^{-1}(q')||G_B^{-1}(q'-q)|} . \end{aligned} \quad (2.183)$$

Now, we head on to specify the flow of the fermionic (inverse) propagator whose β -function includes just one contribution that properly reduces to the previously result of the SYM phase when $\phi \rightarrow 0$. Its derivation closely follows the computational steps performed in [section 2.2.3](#). In particular we derive

$$\begin{aligned} \partial_k P_\psi^k(q) = & h_{\varphi,k}^2 \tilde{\partial}_k \int_{q'} \frac{[\tilde{P}_\varphi^{k,reg}(q') P_\psi^{k,reg}(q'-q)]^*}{|G_B^{-1}(q')||G_F^{-1}(q'-q)|} \xrightarrow{\text{SYM}} h_\varphi^2 \tilde{\partial}_k \int_{q'} \frac{1}{\tilde{P}_\varphi^{k,reg}(q') P_\psi^{k,reg}(q'-q)} \\ \sim & \tilde{\partial}_k \psi(q) \begin{array}{c} \text{---} \bullet \text{---} \\ \text{---} \bullet \text{---} \\ \text{---} \bullet \text{---} \end{array} \psi(q) \quad . \end{aligned} \quad (2.184)$$

As we did notice earlier the Yukawa coupling starts to flow in the SSB phase. Part of the corresponding flow equation was given in [eq. \(2.175\)](#). Here, we would like to outline the computational route to the

⁴⁰Among others we exploit linear shifts and reflection of the *loop momenta* q' in order to condense different terms that emerge when computing the trace.

full result. Starting from a suitable projection prescription we derive

$$\partial_k h_{\varphi,k} = \frac{\delta^3 \partial_k \Gamma_k}{\delta \psi_{\downarrow}(q-q') \delta \psi_{\uparrow}(q') \delta \varphi^*(q)} \Big|_{\eta=\Phi} = \frac{1}{2} \tilde{\partial}_k \text{STr} \frac{\delta^3 \ln G_k^{-1}}{\delta \psi_{\downarrow}(0) \delta \psi_{\uparrow}(0) \delta \varphi^*(0)} \Big|_{\eta=\Phi} \quad (2.185)$$

where the second equality uses the momentum independence of the Yukawa coupling (by definition of our truncation, eq. (2.88)) and therefore we set $q = q' = 0$. Remember that $G_k^{-1} = \Gamma_k^{(2)} + R_k$. To this end, we compute

$$\partial_k h_{\varphi,k} = \frac{1}{2} \tilde{\partial}_k \text{STr} \left[\frac{\delta \Gamma_k^{(2)}}{\delta \varphi^*} G_k \left(\frac{\delta}{\delta \psi_{\downarrow}} \Gamma_k^{(2)} \right) - G_k \frac{\delta \Gamma_k^{(2)}}{\delta \psi_{\uparrow}} G_k + \frac{\delta \Gamma_k^{(2)}}{\delta \varphi^*} G_k \frac{\delta \Gamma_k^{(2)}}{\delta \psi_{\uparrow}} - G_k \frac{\delta \Gamma_k^{(2)}}{\delta \psi_{\downarrow}} G_k \right] \Big|_{\eta=\Phi} \quad (2.186)$$

where, for reasons of notational clarity, we dropped the momentum index ($q = 0$), incorporated the fact that $\frac{\delta \Gamma_k^{(2)}}{\delta \varphi^*}$ does not depend on Grassmann variables (see below) and extensively employed the relation, eq. (2.30). Furthermore, we note that the propagator evaluated at constant (condensate) field Φ obeys $\Gamma_k|_{\eta=\Phi} = G_k|_{\eta=\Phi}$. Let us proceed by analyzing the first summand of eq. (2.186) under the super-trace. Aside the (full) propagator that is given by eq. (2.130) we derive the following expressions:

$$\frac{\delta \Gamma_k^{(2)}}{\delta \varphi^*} ab(q_1, q_2) = \left(\begin{array}{c|c} \mathcal{B} & \\ \hline & h_{\varphi,k} \mathcal{E} \end{array} \right) \delta(q_1 - q_2) \quad \text{with} \quad \mathcal{B} \equiv \begin{pmatrix} \beta^* & \beta \\ \beta & \alpha \end{pmatrix} \quad \text{and} \quad \mathcal{E} \equiv \begin{pmatrix} \epsilon & \\ & 0 \end{pmatrix}, \quad (2.187)$$

$$\frac{\delta \Gamma_k^{(2)}}{\delta \psi_{\downarrow}} ab(q_1, q_2) = h_{\varphi,k} \left(\begin{array}{c|c} & -\sigma_- & 0 \\ \hline & & \\ \sigma_+ & & \\ \hline & & \\ 0 & & \end{array} \right) \delta(q_1 - q_2) \quad \text{where} \quad \frac{\delta}{\delta \psi_{\downarrow}} \Gamma_k^{(2)} = -\frac{\delta \Gamma_k^{(2)}}{\delta \psi_{\downarrow}}, \quad (2.188)$$

$$\frac{\delta \Gamma_k^{(2)}}{\delta \psi_{\uparrow}} ab(q_1, q_2) = h_{\varphi,k} \left(\begin{array}{c|c} & 1^- & 0 \\ \hline & & \\ -1^- & & \\ \hline & & \\ 0 & & \end{array} \right) \delta(q_1 - q_2). \quad (2.189)$$

Since all matrices are diagonal in momentum space all momentum dependent quantities carry the same (loop) momentum q' , i.e. we have $\text{STr} = \int_{q'} \text{sTr}$ and we focus on the matrix multiplication and super-trace operation in (discrete) field space. More specifically we compute

$$-\text{sTr} \frac{\delta \Gamma_k^{(2)}}{\delta \varphi^*} G_k \frac{\delta \Gamma_k^{(2)}}{\delta \psi_{\downarrow}} G_k \frac{\delta \Gamma_k^{(2)}}{\delta \psi_{\uparrow}} G_k \equiv \bar{\beta}_1 + \bar{\beta}_2 \quad \text{with} \quad \beta\text{-functions defined as} \quad \beta_{1/2} \equiv \tilde{\partial}_k \int_{q'} \bar{\beta}_{1/2}, \quad (2.190)$$

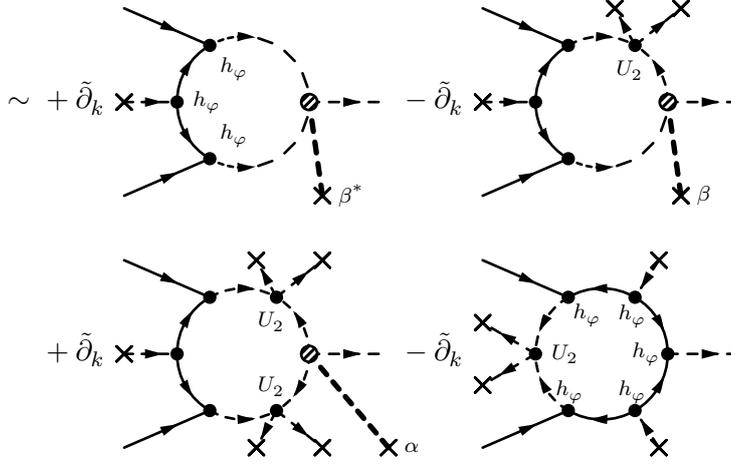
i.e. $\partial_k h_{\varphi,k} = \beta_1 + \beta_2$ where

$$\bar{\beta}_1 = -h_{\varphi,k}^2 \text{Tr} \mathcal{B} G_B(\sigma_- 0) G_F \begin{pmatrix} 1^- \\ 0 \end{pmatrix} G_B = +h_{\varphi,k}^3 \phi \frac{|\tilde{P}_{\varphi}^{k,reg}(q')|^2 \beta^* - 2\beta\theta^* \Re \tilde{P}_{\varphi}^{k,reg}(q') + \alpha\theta^{*2}}{|G_F^{-1}(q')| |G_B^{-1}(q')|^2} \quad \text{and} \quad (2.191)$$

$$\bar{\beta}_2 = +h_{\varphi,k}^3 \text{Tr} \mathcal{E} G_F \begin{pmatrix} \sigma_+ \\ 0 \end{pmatrix} G_B(1^- 0) G_F = -h_{\varphi,k}^5 \frac{\phi^2 \theta^*}{|G_B^{-1}(q')| |G_F^{-1}(q')|^2}. \quad (2.192)$$

Repeating the procedure for the second term of eq. (2.186) we recognize to have exactly the same $\bar{\beta}_{1/2}$. Thus, we finally state

$$\begin{aligned}
\partial_k h_{\varphi,k} = & + h_{\varphi,k}^3 (U_3 \rho^2 + 2U_2 \rho) \tilde{\partial}_k \int_{q'} \frac{|\tilde{P}_{\varphi}^{k,reg}(q')|^2}{|G_F^{-1}(q')||G_B^{-1}(q')|^2} \\
& - h_{\varphi,k}^3 (2U_3 U_2 \rho^3 + 4U_2^2 \rho^2) \tilde{\partial}_k \int_{q'} \frac{\Re \tilde{P}_{\varphi}^{k,reg}(q')}{|G_F^{-1}(q')||G_B^{-1}(q')|^2} \\
& + h_{\varphi,k}^3 U_3 U_2^2 \rho^4 \tilde{\partial}_k \int_{q'} \frac{1}{|G_F^{-1}(q')||G_B^{-1}(q')|^2} \\
& - h_{\varphi,k}^5 U_2 \rho^2 \tilde{\partial}_k \int_{q'} \frac{1}{|G_B^{-1}(q')||G_F^{-1}(q')|^2} \xrightarrow{\text{SYM}} 0 \quad (2.193)
\end{aligned}$$



and the corresponding diagrams are a bit less explicit compared to eqs. (2.175) and (2.176) in order to avoid overloading the message of the condensate processes they represent.

It remains to apply the $\tilde{\partial}_k$ derivative which involves quite some algebraic manipulation, but does not carry much interesting physics. A comprehensive list of the final flow equations for the effective potential $U_k(\rho)$ and the propagators $\tilde{P}_{\varphi,k}(q)$ and $P_{\psi,k}(q)$ provides [appendix D](#) where we include the equation's simplification for the SYM phase. Moreover we argue how one extends those equations for the case of the *imbalanced Fermi gas* ($\mu_{\downarrow} \neq \mu_{\uparrow}$) which was confirmed by an ab initio calculation with an ansatz for the effective action Γ_k where we distinguish the propagators for the spin-up (ψ_{\uparrow}) and spin-down (ψ_{\downarrow}) fermions.

libfrg — A Numerical Library for the Flow Equation

This chapter is devoted to research activities related to the present thesis that covered a major fraction of every day’s working flow. Considerable effort has been spent to establish a numerical library that is capable of providing a framework to solve physical problems tractable with the flow equation. After a short motivation exposed in [section 3.1](#), several aspects of the code’s implementation such as parallel computing and abstract modeling by classes are discussed in [section 3.2](#). [Section 3.3](#) finally turns to objects that are relevant for concrete physical simulations as performed in [chapter 4](#) later on. In particular we focus on the momentum resolution of the inverse propagator, reasonable choices for regulator functions as well as the Chebyshev approximation of the effective potential.

3.1	From Theory to Practice – The Call for Numerics	73
3.2	Aspects of the libfrg Library’s Implementation	75
3.2.1	The Code’s Conceptual Structure	75
3.2.2	When Things Grow Big	80
3.2.3	Logging and Documentation	89
3.2.4	The STr in $\dot{\Gamma} = \frac{1}{2} \text{STr} G\dot{R}$	92
3.3	Details on Implemented Classes of Physical Objects	98
3.3.1	The Effective Potential – Γ_k at Constant Field	98
3.3.2	The (inverse) Propagator – Quadratic Fluctuations of Γ_k	103
3.3.3	The Regulator – Driving the Flow	107
3.4	A Short Note on Licensing	118

3.1 From Theory to Practice – The Call for Numerics

If one starts scanning available options to solve the flow equation, (exact) analytic approaches turn out to become rapidly involved when switching to elaborated truncation schemes. For sure: A set of non-linear, coupled differential equations of the form $\dot{y}(x, t) = f(t, y, y', \dots)$ will, in general, contain a rich spectrum of features and it is not expected to be solved blindfold, if any. Perhaps the most popular example being the dynamics of the *Lorenz attractor*^[Lor63]. Therefore turning to a numerical solution is a natural step to do.

It is the pronounced aim of this chapter to present a library that is tailored to pin down the lack of an existing generic code within the context of functional renormalization calculations. The idea is to establish a platform that allows for continuous expansion and development. It was inspired by existing codes within other communities in physics such as the business of *Monte Carlo Simulations*^[LB05] where codes as the *CUBA* library^[Hah05] exist or the massive collection of specialized codes for astrophysical computations and numerical solutions in cosmology¹.

We are aware of the dimension of our perspective and therefore the following should be treated as a basis that needs long-term support, further development in order to become a successfully established tool for calculations using functional renormalization.

A collection of important issues that guided our choice of tools to build `libfrg` reads:

- flexibility** Since the idea of functional renormalization involves (abstract) mathematical concepts, there is the need to transfer them to an appropriate coding paradigm. Therefore we utilize the C++ programming language² which offers object oriented concepts. This feature will be essential when modeling the effective potential $U_k(\rho)$ or the (inverse) propagator $P_k(q)$ (cf. [section 3.3](#)).
On the one hand, aspects of numerical implementation play a seminal role on the way to describing physics with `libfrg` compared to e.g. comprehensive software packages as *MATLAB* or *Mathematica*. But on the other hand, `libfrg` is aiming at numerics and has been specialized to its needs to remain lightweight.
- control** Due to `libfrg`'s design it is fully controlled by the developers (and users), i.e. technical problems are transparent down to the level of e.g. memory access (c array pointers) and inter-process communication (message passing interface). Access to routine parameters support adaptability to the physical problem to be solved.
- speed** Adequate control of the libraries implementation as well as optimization during code compilation offers the potential of speed enhancement. Moreover the wide compatibility of C++ to C enables access to well established/long-term tested routines³ and integration of even more robust routines written in FORTRAN⁴ is relatively straightforward.

¹A comprehensive list provides <http://asterisk.apod.com/wp/> as of Aug 2013.

²We assume, at least, the C++03 standard^[iso03] as far as it is supported by the *GNU Compiler Collection* (GCC).

³The language was published in the late 1980s by scientists from *Bell Labs*^[KR88].

⁴Originally developed by *IBM*, the first official programmer's manual appeared in the mid-1950s^[IBM56].

- portability & long-term support** Another notable aspect concerns hardware compatibility and future use. Since the *Linux kernel*—which (open source) *Linux* distributions as *Debian* are built on—is written in C and the free *GCC* is based on C++, we expect that C/C++ remains supported in the long run on various platforms. It also helps in realizing parallel computing in heterogeneous computer architecture environments (cf. [section 3.2.2](#)).
- expenses** Moreover, *GNU/Linux* in combination with *GCC* serves as a free framework to further develop `libfrg` free of charge. It therefore is independent of fees, does not rely on commercial third party products with restrictive licenses—a suitable starting point to foster and forward independent research. An overview on all software packets included by `libfrg` is spread in [section 3.4](#). In particular, we list information on licensing.

From a first perspective one might regret this pure approach due to its (formidable) drawback of time-consuming ab initio coding. It involves computer engineering related challenges far from its original motivation to tackle physical questions with the aid of functional renormalization. But along with the reasoning above we judged it worth the effort since there is the lack of such a code within the functional renormalization community. If one intends to discretize the effective action Γ_k for *enterprise application* (cf. [section 3.2.2](#)), we need to address issues on high performance computing for our approach to pay off.

3.2 Aspects of the `libfrg` Library's Implementation

We now aim at outlining the general structure and conceptual idea of `libfrg` as a framework that serves for numerical computations related to functional renormalization. Thereafter, [section 3.3](#) turns to a detailed description of classes implemented by the library.

3.2.1 The Code's Conceptual Structure

This subsection intends to provide a generic overview on the conceptual layout of the `libfrg` code's structure in order to serve as a guide. The very details are spread within the code's inline documentation which one can generate a *doxygen* manual from. [Section 3.2.3](#) provides some remarks on how to.

As mentioned in [section 3.1](#) we employ the paradigm of *object oriented* programming to account for the abstract formulation of functional renormalization. If we recap the flow equation's structure

$$\dot{\Gamma}_k = \frac{1}{2} \text{STr} G_k \dot{R}_k = \beta_k \quad (3.1)$$

it becomes evident that there are the following computational tasks to be rendered:

**ordinary
differential
equation
integration**

In order to advance Γ_k from scale k to $k + \Delta k$ we take advantage of the *GNU Scientific Library* (*GSL*) which provides an adaptive *Runge–Kutta integrator* of type *Cash-Karp (4,5)*^[CK90], i.e. it uses 6 function evaluation to advance the system by a step with accuracy of order four and five in Δk , respectively. Thereafter, an error estimate is drawn from these results. However, there is the option to switch to any other stepping method available in the *GSL* library by specifying the corresponding preprocessor directive in `frg_std_include.hpp`, [listing 3.1](#). Additionally there is control on the adaptive step size by setting appropriate relative and absolute error bounds with the `rkeps{rel|abs}`⁵ attributes of the `FlowParams` structure encapsulated by the `Flow` class described below, see [fig. 3.1](#).

⁵In this chapter we get used to some sort of regular expression notation when referring to functions, classes, etc. written in typewriter font style. The following list serves as a summary to this sort of shorthand notation:

<code>foo{bar1 bar2}</code>	denotes <code>foo</code> bar1 or <code>foo</code> bar2
<code>foo*bar</code>	labels any expression that starts with <code>foo</code> and has trailing <code>bar</code> as e.g. <code>fooanythingbar</code> , ambiguity of the symbol <code>*</code> to the reference symbol (of pointers) should be resolved by the context
<code>{0-4}</code> and <code>{c-f}</code>	matches one of the five numbers <code>0...4</code> and one of the letters <code>c,d,...f</code> , respectively
<code>foo[]</code> and <code>bar()</code>	are associated with the array named <code>foo</code> and the function denoted by the name <code>bar</code> , respectively.
<code><foobar></code>	<code>foobar</code> is a (human readable) descriptor for some variable contents to be substituted, e.g. <code><name of the array>[]</code> ; the distinction from templates, e.g. <code>complex<double></code> , should be obvious from the context

summation & multi-dimensional integration

Within the *momentum representation* the *super-trace* `STr` involves a Matsubara summation at non-zero temperature as well as a d -dimensional (spatial) integration. `libfrg` provides the routine `fullTrace()` that incorporates these operations described in [section 3.2.4](#). Besides multi-dimensional Monte-Carlo-based integrators, the *CUBA library*^[Hah05] (developed at [Max-Planck-Institute for Physics, Munich](#)) contains the deterministic routine `(ll)Cuhre()` we depicted for `libfrg`. As specified for the ODE integrator above, error bounds are set by suitable attributes within the `Flow` class, namely: `Flow::FlowParams.eps{abs|rel}`.

The determination of the Matsubara summation is followed by an (experimental) algorithm implemented by `matSum()`. We present it in [section 3.2.4](#). In a nutshell it selects a given number (`Flow::FlowParams.N0`) of summands and obtains intermediate ones by interpolation.

discretization & interpolation

The effective potential $U_k(\rho)$ and the (inverse) propagators $P_k(q)$ depend on the continuous variables: (bosonic) field expectation value ρ and momentum vector q in free space, respectively. However, one needs to parametrize them to a finite set of flowing quantities for a numerical treatment. Either one votes for a Taylor expansion around a given expansion point that leaves the *continuous nature* of ρ and q intact or one directly discretizes P_k and U_k . Here, we prefer the second option for reasons discussed in [section 1.2](#).

In general, the integration and summation from `STr` requires arbitrary values of ρ and q , and hence there is the need for a suitable interpolation on the grid of U_k and P_k . To be specific: The two-dimensional⁶ interpolation of $P_k(q)$ relies on a combination of one-dimensional cubic splines⁷ designed to meet two requirements: speed and *global* interpolation. The one-dimensional case for U_k is performed using the Chebyshev approximation (see [footnote 7](#)) in order to account for well defined derivatives of the effective potential.

The abstract design of `libfrg` is illustrated in [fig. 3.1](#) where classes (names in boxes) are related by a) inheritance (solid arrows) or b) class attributes whose specific names are printed next to the dashed

⁶So far, the implementation of the (inverse) propagator P_k given by the class `Propagator` is restricted to a two-dimensional grid, e.g. to separately represent frequency q_0 and the magnitude of the spatial momentum \mathbf{q}^2 for non-relativistic, spatially homogeneous setups. We would like to note that it is straightforward to adapt it to the situation of $O(N)$ symmetry (leaving $q^2 = q_0^2 + \mathbf{q}^2$ invariant): Restrict the q_0 -grid to the single point $q_0 = 0$ and work with the \mathbf{q}^2 -direction left.

⁷A basic introduction to the subject of spline interpolation is given in e.g. [\[Ran06\]](#), ch. 2.3 and ch. 2.6 discuss the Chebyshev approximation.

⁸The class contains an array of function pointers `frgInt*` that have to be set by the user with routines that return the integrand's value of a given flowing object (e.g. the (inverse) propagator `Propagator`). In addition `frgInt()`s take two structures of types `ExtParamIntGr` and `LoopParamIntGr` which pool indices related to the flowing object itself (e.g. the *external* momentum p , cf. [fig. 3.6](#)) and labels related to the integration/summation from the trace operation `STr` (e.g. `loop-momentum q`), respectively.

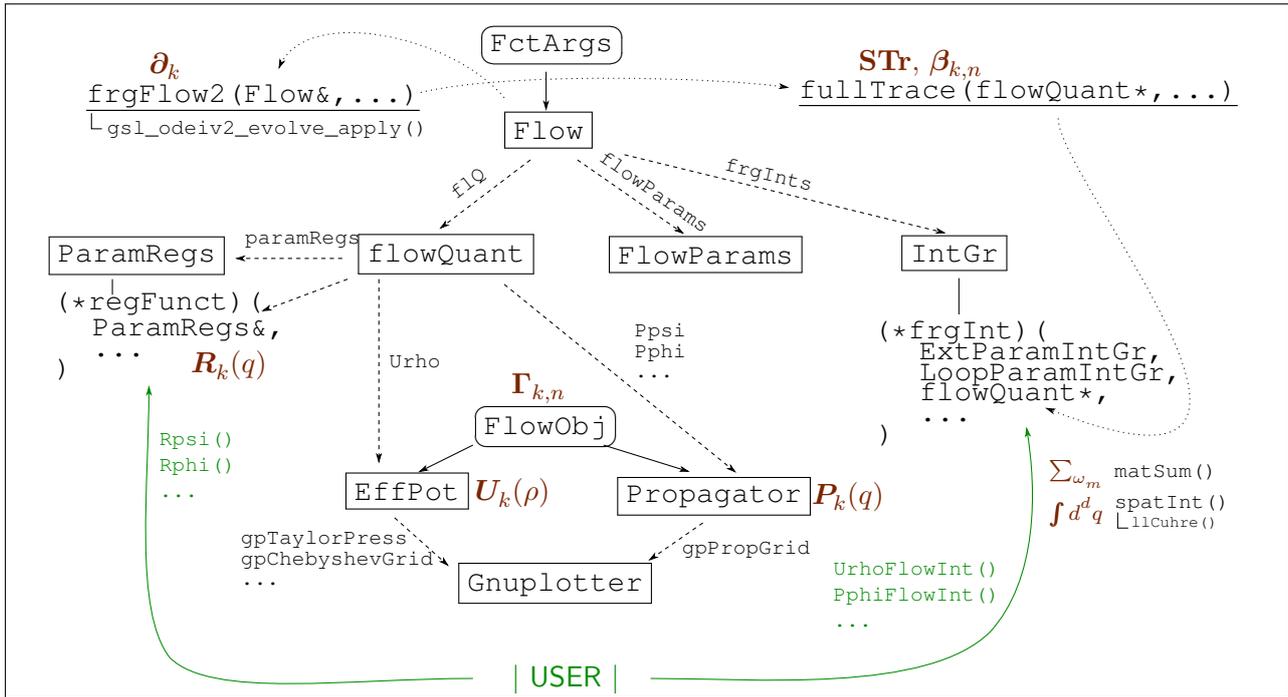


Figure 3.1: *Conceptual design of libfrg*. Collaboration diagram providing an overview on the abstract design of libfrg. The (brown) symbols represent the mathematical objects behind the code's structure. They are linked against the corresponding sections in this chapter. Solid (black) arrows indicate class inheritance and dashed ones track class attribute types corresponding to instances (next to the dashed line) of classes. Virtual classes are framed by oval boxes. Dotted arrows suggest how the structures/classes interact with the key routines of libfrg (underlined).

The top level object is an instance of class `Flow` that encapsulates all necessary objects to advance the flow equation, eq. (3.1): The integrands `frgInts` represented by the class `IntGr`⁸, a collection of all flowing quantities bundled by the instance `f1Q` of the class `flowQuant`, and (technical) parameters `flowParams` like numerical error bounds, number of Matsubara summands to explicitly evaluate, etc.. `Flow` instances are passed as arguments to the top level routine to advance the system, `frgFlow2()`. Besides function pointers to bosonic and fermionic regulator functions which take arguments of type `ParamRegs`, the `flowQuant`-class contains a vector attribute with pointers to the abstract class `FlowObj`. Each flowing object as e.g. the effective potential class `EffPot` or the (inverse) propagator class `Propagator` should inherit from `FlowObj`. Unless the implementation of the flowing object does not provide all generic/(pure) virtual functions declared by `FlowObj`, it stays virtual—it is not allowed to be used by means of instantiation.

The library's user has to provide two sets of functions: The regulator functions $R_{\dots}(q)$ on the one hand and the integrands $I(U_k(\rho), P_{k,\varphi}, R_{k,\varphi}, \dots)$ to be traced on the other hand. Both are included into the instance of `Flow` by setting appropriate function pointers. A collection of regulator functions is implemented in `frg_regs.cpp`

arrows. The global object that merges all information necessary to solve the flow of the physical system, eq. (3.1), is instantiated from the class `Flow`. In order to uniquely mark `Flow` instances as valid function arguments, the class inherits from the abstract base class `FctArgs`⁹. To evolve the flow equation represented by the instance `flow` of type `Flow`, `libfrg` provides the function

```
frgFlow2(Flow& flow, double t0, double t1, bool (intr*)(flowQuant*, bool))10
```

which advances the objects collected in the `Flow` instance from (logarithmic) scale $t_0 = \ln k_0/\Lambda$ to t_1 . The numerical value of Λ is set by the constant `double`-value `Lambda` in `frg_std_include.hpp`. The interrupt function `intr()` is optional, i.e. it has a default value; it provides the opportunity to stop `frgFlow2()` due to some user defined criterion on e.g. the instance `flow`. `Flow` itself contains, among others, three main attributes:

<code>flowParams</code>	An instance of the <code>FlowParams</code> class that groups various settings for the evolution/solution of the flow equation,
<code>flQ</code>	the collection of flowing quantities represented by the class <code>flowQuant</code> ¹¹ ,
<code>frgInts</code>	and a set of functions declared in class <code>IntGr</code> that represent the integrands under <code>STr</code> on the rhs. of the flow equation (and corresponding projected flows).

Schematically one might rewrite eq. (3.1) as

$$\partial_t \text{flQ} = \text{fullTrace}(\text{frgInts}(\text{flQ}))|_{\text{flowParams}} \quad . \quad (3.2)$$

In practice the integrands in `frgInts` will be integrated by some function that incorporates the `STr` operation. To this end `libfrg` provides `fullTrace()` specified in section 3.2.4. The corresponding value (rhs. of eq. (3.1)) is used by the *GSL* ODE integrator to advance the flowing quantities in `flQ`. The required *GSL* function pointer

```
Flow::beta of type int(*) (double, const double*, double*, void*)
```

has to be passed to the `Flow` constructor when a corresponding object is instantiated.

The discrete set of flowing quantities represented by the instance `flQ` are logically grouped into so called *flowing objects*. As we will see in a moment, this abstraction layer is introduced for straightforward addition of arbitrary new classes to characterize any (numerical) truncation of Γ_k by suitable object oriented methods and attributes. The only restriction for new classes to be integrated into and used by `frgFlow2()` through `Flow::flowQuant` is the requirement that it has to inherit from the abstract

⁹To some extent this step is redundant. Nevertheless, we introduced it as an analogue to a function argument `void*` in C. Any function `func(FctArgs ¶ms)` has to typecast the generic argument `params` to its needs.

¹⁰For historical reasons there is also a predecessor `frgFlow()`. Both functions are defined in `frg_flow.cpp`.

¹¹For historical reasons the class name starts without a capital letter, here. The opposite case constitutes the standard form we use for such labels.

base class `FlowObj`. This class declares a number of virtual functions. Unless any of them remains undefined those stay virtual and are not allowed to be used as valid flowing objects.

The reason behind this behavior becomes transparent as follows: The concept of `libfrg` linearly orders all flowing quantities into a (giant) vector with coefficients $\Gamma_{k,n}$ where $n = 0, 1, \dots$ denotes the corresponding discrete index; k remains the (continuous) scale parameter, as before. Classes representing flowing objects are expected to linearly order their flowing quantities. This requirement is fulfilled by defining the interface functions:

```
FlowObj::getLinIndex()  takes the structure GridCoord12 to translate it into a
                        linear index, and
FlowObj::getGridCoord() reverses the operation of getLinIndex() .
```

Following the same spirit there are additional functions that manage *live plotting* during processing and getting/setting the (complex) values of the flowing quantities, respectively. Moreover, `FlowObj::setExtParamIntGr()` was declared for the purpose of specifying external parameters that are important for the integrand stored within an instance of the class `IntGr`.

Back to the class `flowQuant`: It contains an instance of the standard C++ vector class template with `FlowObj` pointers. These are defined during construction of a `flowQuant` instance. The order of the vector dictates the order of the `IntGr::frgInts` array in the sense:

$$\text{IntGr::frgInts}[m] \leftrightarrow \text{flowQuant::flObj}[m] \quad , \quad (3.3)$$

i.e. the number of integrands and that of the flowing objects is required to coincide. Thus, eq. (3.2) narrows down to

$$\partial_t \text{flQ.flObj}[m] = \text{fullTrace}(\text{frgInts.frgInts}[m](\text{flQ}))|_{\text{flowParams}} \quad . \quad (3.4)$$

Note that we chose the index m instead of n from $\Gamma_{k,n}$ on purpose: While the former one labels the flowing objects represented by the class `FlowObj`¹³, the latter one counts individual flowing quantities collected within one/different flowing objects.

In order to manage the files which contain the code for the mandatory components of `libfrg` listed in table 3.1, there is a global `Makefile` which compiles simply by typing the command

```
user@bash, FRGv1> make
```

II.1

`Makefile` is configured by the text file `conf.mk`, documentation included inline. It contains settings as:

- local compilation and usage of the *GSL*, *GMP*, *CUBA* and *IPM* libraries that ship with the software package of `libfrg` (cf. table 3.3)

¹²It is simply a tuple of `int` values e.g. to label a multi-dimensional grid of flowing quantities, cf. the `Propagator` class, section 3.3.2.

¹³... as e.g. the effective potential $U_k(\rho)$ (e.g. $m=0$) or the (inverse) propagator $P_k(q)$ (e.g. $m=1, 2, 3$ for a bosonic and two fermionic particles).

src/	
frg_errLog.cpp	basic capabilities for logging the library's activity, section 3.2.3
frg_std_include.cpp	see frg_std_include.hpp below
effPot/	
frg_effPot.cpp	class EffPot representing the effective potential $U_k(\rho)$, section 3.3.1
flow/	
frg_flow.cpp	contains the top level functions to run the flow, namely frgFlow2(), section 3.2.1
frg_livePlot.cpp	class Gnuplotter for data printing and live plotting, section 3.2.3
frg_regs.cpp	collection of regulator functions, section 3.3.3
headers/*.hpp	pool of header files, e.g. for inclusion on C/C++ code when (statically) linking against the library libfrg
frg_std_include.hpp	contains includes for necessary libraries (table 3.3) plus generic settings to run the flow and basic inline functions
parallel/	
frg_parallel.cpp	routines for parallel computing using <i>MPI</i> , section 3.2.2
prop/	
frg_prop.cpp	class Propagator to represent the (inverse) propagator $P_k(q)$, section 3.3.2
frg_spline.cpp	functions for one- and two-dimensional spline interpolation
frg_struct.cpp	collects various structures and classes for abstract handling of the flow and its mathematical objects; among them: the <i>global</i> class Flow covering all sub-classes, see section 3.2.1 and fig. 3.1
trace/	
frg_trace	routines to perform the STR-operation, section 3.2.4

Table 3.1: *Source code files of libfrg*. Summary of the file structure which contributes to the components necessary for compiling libfrg.

- names of standard directories
- specification of compilers in use (for C, C++, and FORTRAN¹⁴)
- general compiler and linking options for e.g. profiling and debugging .

3.2.2 When Things Grow Big

Any serious attempt aiming at high performance numerical computation will eventually end up scheduling tasks in parallel. In general, advancing $\Gamma_{k,n}$ from scale k to $k + \Delta k$ is dictated by a set of (strongly) coupled differential equations

$$\dot{\Gamma}_{k,n} = \beta_{k,n}(\Gamma_{k,0}, \Gamma_{k,1}, \dots) \quad n = 0, 1, \dots, \quad (3.5)$$

¹⁴Exclusively for compiling the optional *IPM* library.

where the solution $\Gamma_{k_1,n}$ depends on all¹⁵ $\Gamma_{k_0,n}$ with $\Lambda \geq k_0 > k_1$ up to the initial (microscopic) scale Λ . However, it is definitely an option to compute $\Gamma_{k,n} \rightarrow \Gamma_{k+\Delta k,n}$ for every n in parallel.

To this end we developed a dynamic parallelization scheme that follows a hybrid ansatz mixing multi-threading with *OpenMP* and inter-process communication based on *MPI*. It follows a hierarchic model where a single *master MPI* process dynamically schedules the $\Gamma_{k,n}$ to a number of *slave MPI* processes. The slaves report their number of available compute cores J first. Then, they process the portion of (maximally) J jobs

$$\Gamma_{k,n} \rightarrow \Gamma_{k+\Delta k,n} \quad (3.6)$$

given a tuple of n -values determined by the master. When finished, a slave distributes its results to all *MPI* processes and signals its standby state to the master. In turn, the master redistributes work.

There are two main benefits which triggered our design:

1. *OpenMP* parallelization allows using shared memory which reduces consumption of such resources. Note that every *MPI* process needs to take its own copy of `flQ`. Furthermore we recall: The evolution step, eq. (3.6), is obliged to incorporate the full information of all $\Gamma_{k,n}$.
2. *MPI* enhances flexibility of parallel evaluation. Different computing resources are allowed to be connected in a network transparent manner; The underlying infrastructure might be supported by either a *virtual private network (VPN)* on top of the internet or—in total contrast concerning speed and issues of reliability— a compute cluster's high speed connection such as *InfiniBand*.

The fact that every new result $\Gamma_{k+\Delta k,n}$ is directly communicated among all *MPI* processes distributes the network traffic over time. It tends to avoid peaked communication in contrast to the situation where the master is responsible for collecting and spreading the updates of $\Gamma_{k,n}$. To get our hands on a lower bound of the total amount of network traffic to be expected, let us roughly estimate the situation for $N = 1K \equiv 2^{10}$ flowing quantities $\Gamma_{k,n}$ advanced by $S = 2^6 \approx 60$ steps with the aid of $M_{\text{mpi}} = 2^4 \approx 20$ *MPI* (slave) processes. The update information for each flowing quantity consists of at least $|u| = (2 + 2 \cdot 8)\text{B} \gtrsim 2^4\text{B}$ (`1 x size_t, 2 x double`). Each *MPI* process needs to get the full information on the update of the flowing quantities after every successive step. Hence the total load L due to updating¹⁶ evaluated data reads

$$L_{M_{\text{mpi}}}^{\text{eval}}(N \cdot S) = S \cdot N \cdot M_{\text{mpi}} \cdot |u| \stackrel{\text{example}}{\gtrsim} 2^{24}\text{B} = 16 \text{ MB} \quad . \quad (3.7)$$

In addition there is traffic from the distribution of *OpenMP* jobs among the *MPI* slaves by the master: $|j| \gtrsim 2\text{B}$ (`1 x size_t` plus at most 2 bit that signal the slave state), i.e. there is the need for *MPI* related

¹⁵By the way: The physics of *critical phenomena* near *second order phase transitions* is characterized by the effect of *loss of memory*, i.e. the correlation between different Γ_k as $k \rightarrow 0$ becomes weak in the sense that it is sufficient to parametrize the physics by few numbers, called *critical exponents*. They are related to symmetry properties of the system. ^[ZJO02,AS10]

¹⁶For simplicity we do not account for the portion of redundant traffic due to the fact that a slave does not need to send its own calculation to itself. In fact, our implementation does transmit these data. For each step $k \rightarrow k + \Delta k$ on would reduce $L_{M_{\text{MPI}}}^{\text{eval}}$ by N . Therefore one gains a fraction of $1/M_{\text{MPI}}$ in efficiency concerning network traffic.

communication data of order

$$L^{\text{com}}(N \cdot S) = S \cdot N \cdot |j| \stackrel{\text{example}}{\gtrsim} 2^{17} \text{B} = 128 \text{ KB} \quad . \quad (3.8)$$

According to eqs. (3.7) and (3.8) the ratio of *useful data* to *communication overhead* is given by

$$\frac{L_{M_{\text{mpi}}}^{\text{eval}}}{L^{\text{com}}} = M_{\text{mpi}} \frac{|u|}{|j|} > 1 \quad . \quad (3.9)$$

Note, that $|u|/|j| = \frac{22}{6} \dots \frac{18}{2} \approx 4 \dots 9$ depending on the number of bits 16...64 used for type `size_t` (`int...long long`). In our example above we used 16 bit that is enough to label the (signed) number $N = 1000$ of flowing quantities.

Since eq. (3.5) is separately advanced from k to $k + \Delta k_n$ for fixed n by using an adaptive Runge-Kutta algorithm with step width Δk_n , the overall

$$\Delta k_{\min} \equiv \min(\Delta k_0, \Delta k_1, \dots) \quad (3.10)$$

needs to be determined by the master after having received corresponding information from the slaves in order to respect the error bounds. Subsequently, the *final* evolution $\Gamma_{k,n} \rightarrow \Gamma_{k+\Delta k_{\min},n}$ is coordinated by the master again. These conceptual steps are coordinated by the function `frgFlow2()` from `libfrg`. Assuming complex valued $\Gamma_{k,n}$, the routine

```
parallSchedProcArrMPIOpenMPHybrid(
    double gRe[], double gIm[],
    bool neg[], size_t nG,
    FctArgs &arg, parallEval f
)
with g{Re|Im}[] the arrays of length nG representing
     $\Gamma_{k,n} = \Re\Gamma_{k,n} + i\Im\Gamma_{k,n} \quad ,$ 
```

encapsulates the core ability for distributed computing. It is called within `frgFlow2()` and an overview on the *MPI* communication protocol is given by fig. 3.2. The array `neg[]` of same size as `g{Re|Im}[]` provides the option of specifying single $\Gamma_{k,n}$ s to be neglected. The master traverses this array and distributes jobs with index n according to `!neg[n]` and uses it to track jobs left for processing at the same time. `neg[n]` equals to `true` for $n=0, \dots, nG-1$ when `parallSchedProcArrMPIOpenMPHybrid()` returns. The core job of slaves consists in executing the routine `f` with generic argument `arg` and storing the computation's result to `g{Re|Im}[n]`. The incarnation¹⁷ of `f` in `libfrg` reads

```
parallRoutine(size_t n, FctArgs &arg, double &gRe, double &gIm)
```

¹⁷One might employ `parallSchedProcArrMPIOpenMPHybrid()` in a context independent from solving the flow equation.

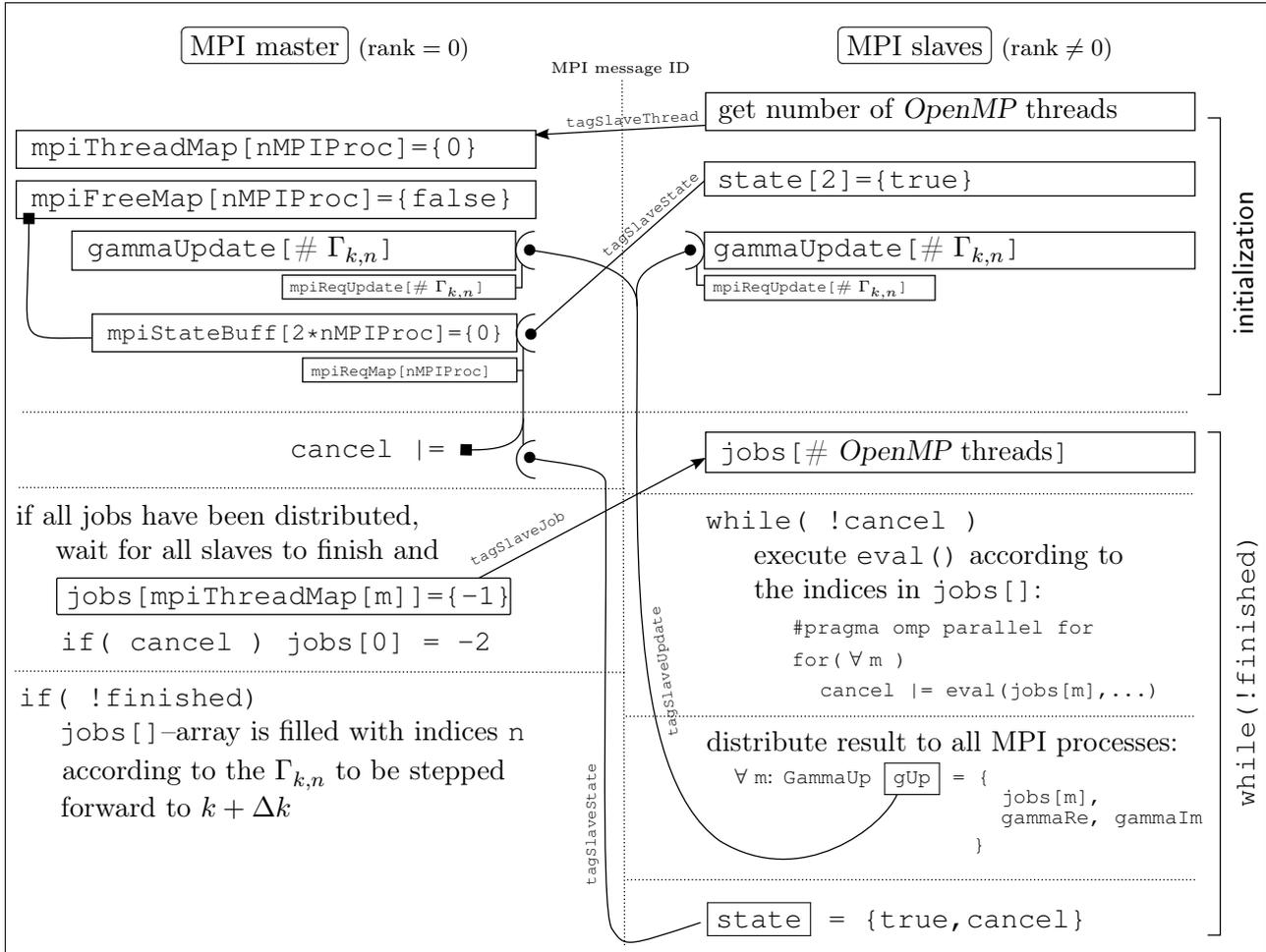


Figure 3.2: *Hybrid parallel computing design of libfrg.* MPI master–slave communication protocol of `parallSchedProcArrMPIOpenMPHybrid()` for distributed computation of the evolution step $\Gamma_{k,n} \rightarrow \Gamma_{k+\Delta k,n}$. All send operations are synchronous (`MPI_Ssend()`) except for updates of $\Gamma_{k,n}$. These are communicated via the purely blocking routine `MPI_Send()`. Non-blocking receives (`MPI_Irecv()`) are marked by bullets (•) and their blocking counterparts `MPI_Recv()` are represented by arrows (→). Memory attached to the buffers get labeled by a filled square (■).

defined in `frg_flow.cpp`.

Let us estimate the theoretical speedup compared to the serial execution of the code. Concerning N flowing quantities $\Gamma_{k,n}$, we assume the amount of time to step forward a single $\Gamma_{k,n}$ corresponds to complexity class $\mathcal{O}(N)$, at a minimum¹⁸. Hence, $\Gamma_{k,n} \rightarrow \Gamma_{k+\Delta k,n}$ will take time $\mathcal{O}(N^2)$. The more successive steps we take, the longer the program will run. Therefore the adaptive step size algorithm

¹⁸Referring to eq. (3.5), $\beta_{k,n}$ depends on all flowing quantities $\Gamma_{k,n}$. Thus, the minimal effort for computing a single $\beta_{k,n}$ generally requires to retrieve all value $\Gamma_{k,n=1\dots N}$ once. In practice the complicated ST_r operation might consume a considerable larger number of function evaluation on the one hand and $\beta_{k,n}$ does not need to depend on all flowing quantities on the other hand.

helps to wisely spent time at those steps that need to be small in order to stay within the given error bounds and to speed up if the *slope* $\beta_{k,n}$ is *sufficiently small*, i.e. $|(\Gamma_{k+\Delta k_{\min,n}} - \Gamma_{k,n})/\Delta k_{\min}| \lesssim 1$. However, whether we consider the performance of a single step or a series of these will not depend on the number N of flowing quantities. The total computing time $T(N)$, eq. (3.11), becomes proportional to the number S of single steps taken.

Now, the fraction of parallel execution depends on the total number M_{tot} of threads distributed among the *MPI* slaves. The degree of parallelization increases according to the fraction M_{tot}/N and saturates to 1 for $M_{\text{tot}} \geq N$. Then, all flowing quantities are stepped forward in parallel consuming time $\mathcal{O}(N)$ and $M_{\text{tot}} - N$ central processing units/threads/cores become idle. The parallelization speedup is accompanied by increasing *MPI* communication summarized by eq. (3.9). Each *MPI* process keeps its own copy of `flQ` and therefore needs to stay updated. Depending on the network's bandwidth B , additional time $\mathcal{O}(M_{\text{mpi}}N/B)$ is needed. It is crucial to take B as an abstract quantity that is not necessarily equivalent to the (theoretical) bandwidth of the physical network such as $\sim 100\text{Mbit/s}$ on *Ethernet* or $\sim 100\text{Gbit/s}$ for *InfiniBand*. One crucial reason for this statement was addressed [above](#) when discussing network traffic due to *MPI* communication: Spreading the updates of the flowing quantities $\Gamma_{k,n}$ takes place distributed over the period of parallel computing. However, this effect decreases with increasing M_{tot} and at $M_{\text{tot}} = N$ communication becomes *peaked* again. As a (crude) estimate we incorporate an *effective bandwidth* by the substitution: $B \rightarrow B \cdot N/M_{\text{mpi}}$. Therefore we end up with time complexity

$$T(N) = T_{M_{\text{tot}}}^{\text{eval}}(N) + T_{M_{\text{mpi}}}^{\text{com}}(N) = \mathcal{O}\left(\frac{N^2}{M_{\text{tot}}}\right) + \mathcal{O}\left(\frac{M_{\text{mpi}}^2(N)}{B}\right)$$

with $M_{\text{mpi}} \leq M_{\text{tot}} \leq N$, S fixed . (3.11)

To be more specific, we outline the following cases:

<i>serial execution</i>	$B \rightarrow \infty$ $M_{\text{tot}} = 1$	$t(N) = \mathcal{O}(N^2)$		
<i>single multi-core</i>	$B \rightarrow \infty$ $M_{\text{tot}} = N$	$t(N) = \mathcal{O}(N)$		
<i>hybrid computing</i>	B finite $M_{\text{tot}} = N$	<i>i) full MPI</i>	$M_{\text{mpi}} \sim N$	$t(N) = \mathcal{O}(N) + \mathcal{O}(N^2/B)$
		<i>ii) full hybrid</i>	$M_{\text{mpi}} \approx M_{\text{omp}}$	$t(N) = \mathcal{O}(N) + \mathcal{O}(N/B)$

For simplicity the last case defines

$$M_{\text{tot}} = M_{\text{mpi}} \cdot M_{\text{omp}} \tag{3.12}$$

with M_{omp} the number of threads per *MPI* slave¹⁹. Note, that our calculation dropped the time consumed by `MPI_*()` routines. Such an assumption requires them to be negligible compared to `fullTrace()`. However, we tried to argue for our hybrid ansatz of parallelization. The upshot might be formulated as follows: As of today, available multi-core processors contain ~ 10 cores and

¹⁹Nevertheless the code in `libfrg` is as generic as the master is able to support different numbers of *OpenMP* threads per *MPI* slave.

parallelization based on pure *OpenMP* becomes quite restrictive for systems with $N \sim 10^3$. Relying on *MPI* only introduces reasonable network traffic decreasing speed and massive usage of memory.

We would like to continue addressing some practical issues:

- In order to carry out performance measurements the software package including *libfrg* ships with the library *IPM*. It is dynamically loaded through shell variables, e.g.:

```

user@bash, ~> LD_PRELOAD=<path/to/libipm.so> \                               II.2
                IPM_LOGFILE=<ipm.log> \
                IPM_LOGDIR=<path/where/to/store/ipm/logging> \
                mpirun <options> <binaries>

```

Figure 3.3 provides part of the data from the performance measurement using *IPM*. In particular the plot to the right underlines the *master–slave*–model set up for parallelization with (*Open*)*MPI*. Here, the process with rank 0 corresponds to the master which is mainly involved in distributing and coordinating jobs to the slaves. It spends almost all of its processing time for calling `MPI_*()` routines. A detailed profiling analysis separated by master and slave processes reflects **fig. 3.4**. It is crucial to note that a major fraction of computing power is spent to evaluate the integrand of type `frgInt`. It has to be provided by the library's user. Therefore speed performance of the compiled binaries relies, among others, on an efficient implementation²¹ of those routines.

- The mandatory *CUBA* library offers parallelization through *POSIX Threads*^[CO95] (`pthread`). To avoid any interference with *OpenMP* one should disable this feature. In addition, we loosely observed that `llCuhre()` from the *CUBA* library fires `syscalls` to the *Linux kernel* which thwarts speed of computation. Setting the shell environment variable `CUBACORES=0` is suggested before running the code.
- Depending on the number N of flowing quantities one should decide on the type of data that represents the label n of $\Gamma_{k,n}$. It needs to be a signed integer (not of type `unsigned`) in order to allow the *MPI* master process to send messages on having finished distribution of jobs to the slaves; or to handle cases where a slave encountered an error. Taking e.g. `long` for $N = 10^3$ incorporates quite some overhead/redundant bits for network traffic. If N exceeds the capacity of the data type, unpredictable behavior is expected due to integer overflow. A wise choice is suggested and has to be specified in `frg_std_include.hpp`, see **listing 3.1** including comments therein.

²⁰The plot does not provide time-resolved information.

²¹Especially `spatInt()` which calls `llCuhre()` from the *CUBA* library extensively invokes the user defined integrand functions of type `frgInt`. Details of computing its return value affect the code's efficiency. For instance, multiple evaluation of specific function values (as `sin(π)`) should be avoided. Storing it to a (temporal) variable proves beneficial for the code's speed evaluation—if not optimized out by the compiler anyway.

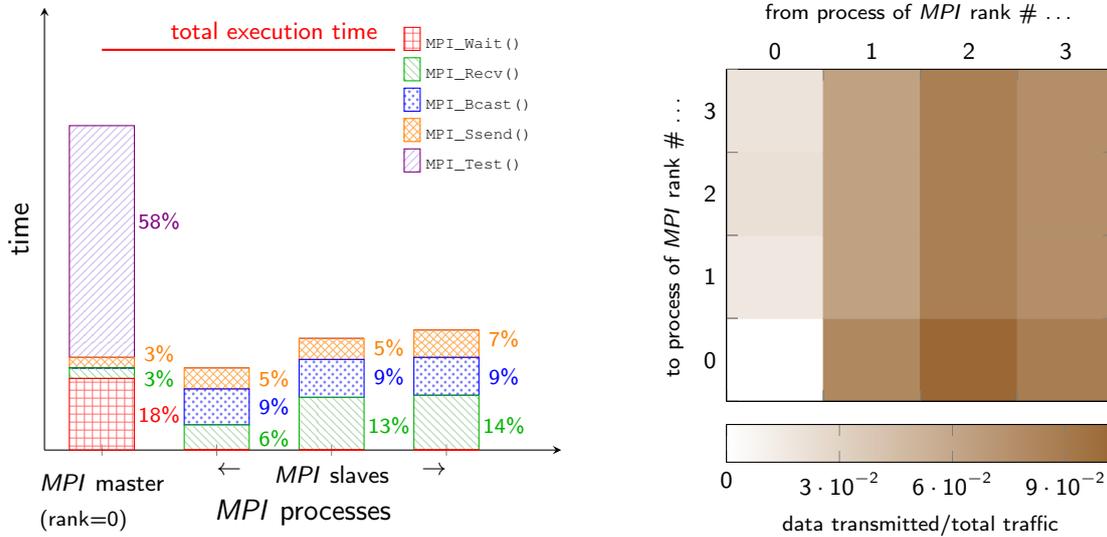


Figure 3.3: *MPI performance measurements for parallel computing in libfrg.* Collection of IPM output from evaluating a test setup of flowing quantities at zero temperature. The panel to the left provides timing information with respect to *MPI* routines. Time measurements are normalized to the total time of execution and communication that consumes less than 1% of execution time (as e.g. `MPI_Ssend()`) is dropped. As opposed to the slave processes ($\text{rank} > 0$) the master almost exclusively spends computing resources on coordinating fractions of the work load from `frgFlow2()`. Our benchmark took a scenario where the evaluation of the distributed jobs labeled by n was trivial, i.e. `eval(n, ...)` simply returns a constant value. Hence the fraction of time t_{MPI} spent for `MPI_*()` calls and that for `eval(n, ...)`, t_{eval} , is of same order: $t_{\text{MPI}} \sim t_{\text{eval}}$. It helps to obtain convenient scales for the plot to the right. *Real life* scenarios typically shift the load according to $t_{\text{eval}} \gg t_{\text{MPI}}$ (as desired).

The right panel contains information on the amount of network traffic due to *MPI* communication expanded in terms of the *MPI* topology, i.e. the colored square at coordinate (i, j) encodes the amount of communication between processes n and m relative to the total network traffic. The upshot on the graph (partially)²⁰ reflects our parallel computing design’s intention: Avoid peak traffic by spreading update information on $\Gamma_{k,n}$ distributed over the time of evaluation. In fact, the *MPI* master ($\text{rank}=0$) purely sends job distribution information to the slaves (column $(0, j)$) which consumes about 5% of network bandwidth. Nearly the same amount, $\sim 4\%$, is added to the communication of updates of $\Gamma_{k,n}$ spread from the slaves (row $i, 0$) to all *MPI* processes. The remaining fraction accounts for updating (square $(0, 1) \times (3, 3)$). Therefore the ratio “bytes of *update data* to bytes of *job distribution*” reads $\sim 10/1$, cf. eq. (3.9). Distribution of the jobs depends on the slave’s rank, the number of *OpenMP* threads available to them, and the speed on which they perform the jobs. Hence, the slaves’ *MPI* communication activity differs in general (darker column $(n, 2)$ compared to the neighboring ones.).

Listing 3.1: fraction from `frg_std_include.hpp`

```

    ///set the GSL ODE inetration method to be used
148 //select RK4(5) with Cash-Karp error estimate as integration routine
    #define GSL_ODEIV_STEP_TYPE gsl_odeiv2_step_rkck
150 /*}}}}*/
    //fold: PARALLEL COMUTING/*{}}*/
152 ///OpenMP writelock for logging functions (to be initialized in main()-routine)
    extern omp_lock_t writelock;
154 /// \brief MPI data type used for registering the MPI data type derived from
    /// the structure GammaUp in frg_parallel.cpp
156 ///
    /// roughly, it represents the index of the (linearized) array
158 /// gamma(Re|Im)[] taken by parallSchedProcArrMPIOpenMPHybrid(), but
    /// needs to adapt negative values in order to communicate the messages
160 /// "finished" and "cancel" from the MPI master to the slaves,\n
    /// <b>note:</b> the index of gamma(Re|Im)[] is declared as size_t whose
162 /// size depends on the machine which the library was compiled for
    /// (adressable memory), therefore the maximal value of the (necessarily)
164 /// signed type PARALLEL_JOB_INDEX_T_MPI (along with the type
    /// parallel_job_index_t) must not exceed the maximal value of size_t on
166 /// any machine involved into the parallel computation! Otherwise,
    /// unpredicted behavior may follow: e.g. gamma(Re|Im)[n] may cause a
168 /// segmentation fault
    #define PARALLEL_JOB_INDEX_T_MPI MPI_LONG_LONG_INT
170 /// \brief type that is used to index the parallel jobs distributed by
    /// parallSchedProcArrMPIOpenMPHybrid()
172 ///
    /// it needs to be equivalent to the corresponding MPI type
174 /// PARALLEL_JOB_INDEX_T_MPI above
    typedef long long int parallel_job_index_t;

```


3.2.3 Logging and Documentation

libfrg provides support for logging the routine's activity by means of logfiles. This feature is implemented by `frg_errLog.cpp` which houses the functions `logMsg()` and `errMsg()` for generic logging information and error messages, respectively. Both need to be supplied with the calling function's name as well as a (meaningful) string of information that specifies a reason its invocation. Their difference is nearly trivial: The former takes a file name to write to²² and the latter increments the *error counter* `errCount`²³ and uses `ERR_LOG_FN` set in `frg_std_include.hpp` as `extern`, i.e. it is expected to be defined e.g. within the `main()`-file of an application that binds to libfrg. In any case it stores the file to the directory given by `getRootDir()`²⁴.

The routines `{err|log}Msg()` are designed to create thread save logging with respect to *OpenMP* avoiding scrambling of messages on output (to files). The same procedure for *MPI* would include additional message passing among the parallel *MPI* processes. We took a back door to circumvent such network overhead simply by attaching logging information from different *MPI* processes to individual files of the form `<path/to/log/file>.<MPI rank>`.

A (rough) indicator to detect the origin of an avalanche of error messages is given by the following format of the log/error message:

```
Proc. <MPI rank>.<OpenMP thread ID> <date/time> - <mic. sec.>, <routine>: <message>
```

It is appended to the appropriate file specified a second ago. Each function that contributes to libfrg is expected to report on possible errors twofold: Firstly, it should throw a corresponding message through the supplied routines previously described and secondly, it needs to return an error code. For historical reasons this was a `bool` value²⁵. This is why most of the library's functions do so up to now. To become a bit more sophisticated we switched to (compatible) `int` values whose error code translation is defined in `frg_std_include().hpp`, see [listing 3.2](#).

For tracking the progress of `frgFlow2()`, the *MPI* master process calls `FlowObj::livePlot()` on each flowing object after each evolution step from k to $k + \Delta k_{\min}$. Thus it is mandatory to provide the design of a new flowing object with `{init|quit}LivePlot()`—whether it is left blank or it supports certain activity as e.g. saving intermediate states of the flowing object. The constant `saveFlowQuantIntermed` from `frg_std_include.hpp` serves as a trigger for this feature.

Along with the capability described, libfrg provides a suitable class, namely `Gnuplotter`, if *Gnuplot* is available on the machine where the *MPI* master is executed. Corresponding parameters are defined in `frg_std_include.hpp`, cf. [listing 3.2](#). If these settings are inconsistent plotting is dropped. An

²²Under *Linux* one might take e.g. `/dev/stdout` or `/dev/stderr` for redirection of logging to output on command line. As depicted by [listing 3.2](#), using the `logMode` constant from `frg_std_include.hpp` is an alternative.

²³It is an arbitrary precision variable from the *GMP library* to support unlimited large integers. Up to the point of writing it is the exclusive reason which calls for including the *GMP library*. Regarding future use of libfrg it is available to increase precision of calculations. However, one might argue against the meaningfulness of including *GMP*.

²⁴When called for the very first time, an argument that specifies the directory's name is allowed to be supplied. If neglected, `./` is set, i.e. logging is stored to the base directory where the calling binaries are executing from.

²⁵`true` for evaluation success and `false` else

Listing 3.2: fraction from `frg_std_include.hpp`

```

    ///set the gnuplot executable location
262 #define GNUPLOT "/usr/bin/gnuplot"
    ///set gnuplot command line options
264 #define GNUPLOT_OPTIONS ""//"-p"
    ///terminal to be used for gnuplot output
266 #define GNUPLOT_TERM "post_eps_enhanced_color"//"wxt"

268 /**  \brief  sets the mode for logging

270     -1      no logging (in fact every value different from 0 or 1
           results in no logging)\n
272     0      logs to standard output\n
           1      output to file
274 */
    const int logMode = 1;
276
    ///error codes for advanced error handling (introduced on 2013-08-03)
278 ///frg-function's return value for success
    ///(compatible with boolean value for evaluation success)
280 #define FRG_SUCCESS 1
    ///frg-function's return value for (general) failure
282 ///(compatible with boolean value for evaluation failure)
    #define FRG_FAILURE 0
284 ///frg-function's return value for division by zero
    #define FRG_DIVBYZERO 2

```

instance of `Gnuplotter` ships with various routines to visualize arrays on one- and two-dimensional domains. As an example we present [fig. 3.5](#) obtained from invoking `Gnuplotter::plot2DArray-FlowQuant()`. The data of the plot are stored as temporary files (`*.dat.tmp`) in the sub-folder of the return value of `getRootDir()`. This value is set by the external constant `DATA_FOLDER`. The file's name is determined from the instance's name. Similarly, the *Gnuplot* output is specified with trailing `*LivePlotting.tmp`.

When writing the source code of `libfrg` we stuck to some styling guidelines²⁶ we suggest for further development of the library:

- Indentation uses exactly three space characters and is extensively used to highlight different code blocks, list function's arguments, ...
- More structure provide (nested) *folds*. Blocks of code are embraced according to:

```

    //fold: <short summary> /*{{{*/ <block of code> /*}}*/

```

²⁶Since these rules did establish over time they are applied *almost* everywhere.

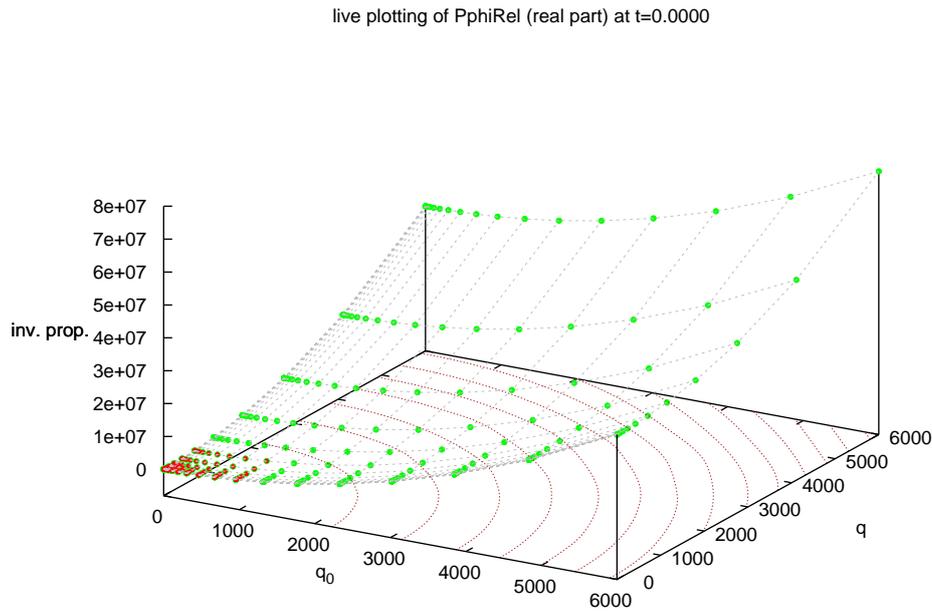


Figure 3.5: *Inverse propagator printed by the Gnuplotter class.* One-to-one output of `Gnuplotter::plot2DArrayFlowQuant()` on input of a two-dimensional array of complex valued quantities (we depict the plot of the real part). The routine incorporates the capability of highlighting a given region/set of grid points (red dots) of the domain which might help to separate *flowing* quantities from *non-flowing* ones. A visual guideline provides an additional contour plot (brown).

Here we depicted the (relativistic) inverse propagator $P_\varphi(q) \sim q^2 = q_0^2 + \mathbf{q}^2$ at *initial* (momentum) scale $k = \Lambda$ ($t = \ln k/\Lambda$). Note that the implementation of the Propagator class, [section 3.3.2](#), allows for a variable distribution of grid points in q_0 - and $|\mathbf{q}|$ -direction. The need for non-flowing quantities (green dots) is due to $\dot{P}_\eta(p) = \int_q I \left[P_\eta^{-1}(p - q) \right]$ in where the integrand I depends on (inverse) propagators P_η that itself depend on the *external* momentum p (red dots) and the *loop*-momentum q (green dots), cf. e.g. [fig. 3.6](#) and [eq. \(2.160\)](#).

which logically groups code.

- Naming of variables, constants, classes, etc. follows *camel casing*, i.e. we use medial capitals as in words like *CamelCase* or *BahnCard*. Preprocessor definitions as `FRG_SUCCESS` are exclusively constructed by capital letters. Different words are separated by an underscore.
- Class/structure names start with capital letter²⁷; instances of classes/structures, variables, etc. with lower case letter, i.e. the instance `class` of class `Class` is invoked by `Class class (...)`;
- Declaration of classes, structures, functions, etc. are documented according to *doxygen's* syntax²⁸ within corresponding header files `*.hpp`.

²⁷For historical reasons the structure `flowQuant` provides an exception to this rule.

²⁸Note, that *doxygen* allows for *HTML tags* `<tag>...</tag>` and *L^AT_EX-formulae* within `\f$... \f$`.

Concerning the documentation of `libfrg` we provide comprehensive information with the aid of *doxygen*²⁹. It might be generated with the `Makefile` that ships with `libfrg`'s source code:

```
user@bash, FRGv1> make makedoxy readdoc II.3
```

The argument `readdoc` should be omitted if the command `firefox` does not relate to a *HTML* browser (with *Javascript* support). To keep the standard it will be of reasonable importance to stay with a sufficient level of *doxygen-format-ready* comment lines when adding more contributions in the future. At the time of writing (Aug 2013) the ratio of code to comment is estimated³⁰ as $\lesssim 2/1$ whose current value can be checked at any time using the `Makefile`:

```
user@bash, FRGv1> make codestatistics II.4
```

3.2.4 The STr in $\dot{\Gamma} = \frac{1}{2} \text{STr } G\dot{R}$

The super-trace operation entails a main challenge for the numerical solution of the flow, eq. (3.1). Adopting some sort of abstract (sloppy) language, STr constitutes a *generic summation over the full propagator bounded/weighted by the regulator*. Since $G \equiv G_{k=0}$ is the *Green's function* of the full quantum theory in the sense $G\Gamma^{(2)} = 1$ a parametrization by a suitable *truncation* Γ_k , in general, will become a highly complicated object; it needs to be approximated by sufficiently many *flowing quantities/degrees of freedom* to sum over. The more we are able to perform this job by hand the less we are plagued by time-consuming numerical operations. But, as outlined a minute ago, the involved structure of G limits chances for analytic solutions. As carried out in chapter 2 our situation allows for an explicit computation of the *super* of the super-trace, i.e. we perform the summation over the (discrete) field index of bosons and fermions. This procedure leaves a summation/integration over the (continuous) spatial degrees of freedom. The transformation to *momentum space* yields a $D \equiv (d+1)$ -dimensional integration

$$\text{tr} \equiv \int_{q_0} \int \frac{d^d \mathbf{q}}{(2\pi)^d} \quad (3.13)$$

on momenta \mathbf{q} and (Matsubara) frequencies q_0 which become discrete³¹ at finite temperature $T \neq 0$ with spacing $\Delta q_0 = 2\pi T$. To this end `libfrg` provides `spatInt()` and `matSum()` which are combined

²⁹Project home page: <http://doxygen.org> as of Aug 2013

³⁰We scanned all source files constituting the library `libfrg` using regular expression filters:

```
^(\\*|\\s)*$           We dropped empty and meaningless lines,
(^\\s*(/\\*|//))|\\#    identified single C/C++/Makefile/Bash comment lines as well as
((?!/\\*) .)*\\*+//    mixtures of code and comment within one line.
```

Additionally, we subtracted a total of 10 lines per file due to the *GNU GPL* license notification. However, in this rough estimate, complex multi-line comments are treated as code lines (we employ the `egrep`-tool which scans files line by line). Therefore we used the symbol “less or approximately equal to” in the main text.

³¹Recall the notation $\int_{q_0} = T \sum_{\omega_n}$ from eq. (2.74) where $\omega_n = \pi T n$ with odd or even integer n for fermions and bosons, respectively. In the limit $T \rightarrow 0$ we have $\int_{q_0} \rightarrow \frac{1}{2\pi} \int_{-\infty}^{\infty} dq_0$

to `fullTrace()`. Depending on the *flowing object* (cf. `FlowObj`, section 3.2.1) there is symmetry left to further simplify the trace `tr`. In particular we are concerned about two cases:

1. d -dimensional³² rotational symmetry and
2. a symmetry that becomes broken due to a direction specified by an *external momentum* \mathbf{p} .

Item 1 reduces to

$$\text{tr}_1 \equiv \frac{\Omega_d}{(2\pi)^d} \int_{q_0} \int_0^\infty d|\mathbf{q}| |\mathbf{q}|^{d-1} \quad (3.14)$$

with Ω_d the d -dimensional *full solid angle*. Ω_d is expressed in terms of the surface of the unit sphere embedded into the d -dimensional space. With the generalized factorial, the *Gamma-function*³³ $\Gamma(x)$, we have

$$\Omega_d \equiv \frac{2\pi^{d/2}}{\Gamma(d/2)} \quad \text{and hence} \quad \text{tr}_1 = C_1^{-1}(d) \int_{q_0} \int_0^\infty d|\mathbf{q}| |\mathbf{q}|^{d-1} \quad (3.15)$$

$$\text{where} \quad C_1(d) \equiv \frac{(2\pi)^d}{\Omega_d} = 2^{\lfloor \frac{d-1}{2} \rfloor} \pi^{\lfloor \frac{d+1}{2} \rfloor} (d-2)!! , \quad d > 1 \quad . \quad (3.16)$$

We used the *double factorial*³⁴ $n!! = n(n-2)!!$ for $n \in \mathbb{N}$ as well as the *ceiling function* $\lceil x \rceil$ for $x \in \mathbb{R}$ returning the (unique) integer $m \in \mathbb{Z}$ with $x \leq m < x + 1$. Analogously we defined the *floor function* $\lfloor x \rfloor$ which returns the (unique) integer $m \in \mathbb{Z}$ satisfying $x - 1 < m \leq x$.

Although we could have stopped at the version for $C_1(d)$ that leaves $\Gamma(d/2)$ untouched, our reformulation is in favor of a numerical implementation. The double factorial is easily embedded into a `while`-loop and the ceiling of the division of two integers `n` and `m` is almost naturally implemented in C/C++ through the native support of the floor function in integer arithmetic: `n/m` returns $\lfloor \frac{n}{m} \rfloor$. For $n \neq 0$ a first guess might be `n/m+1`, but if n is a multiple of m , i.e. $m \mid n$, the return value exceeds the result by one, since: $\lfloor \frac{n}{m} \rfloor = \lceil \frac{n}{m} \rceil = \frac{n}{m}$ if n and m are not prime to each other. We need to introduce some modification to `n/m+1` that cures this error, but leaves the result intact otherwise.

To this end, let us decompose n as $km + \bar{n}$ with $k \in \mathbb{N}$ and $m \nmid \bar{n}$ with \bar{n} non-negative. We observe

$$\lfloor \frac{n-1}{m} \rfloor + 1 = k + 1 + \lfloor \frac{\bar{n}-1}{m} \rfloor = \begin{cases} k & , \bar{n} = 0 \\ k + 1 & , 0 < \bar{n} < m \end{cases} . \quad (3.17)$$

Thus the final ceiling routine for `n!=1` reads: `1+(n-1)/m`.

The ($d > 2$)-dimensional symmetry broken case is a bit more involved. First of all, we introduce (generalized) spherical coordinates according to fig. 3.6 (right panel) which satisfy the following recurrence:

³² For $T = 0$ we simultaneously substitute $d \rightarrow D$ and $\int_{q_0} \rightarrow 1$.

³³We assume the difference from the symbol of the effective action Γ_k to be obvious from the context. There is a definition of $\Gamma(z)$ for complex argument z with poles at negative integers including zero by means of analytic continuation^[Nee97,Haz95]. However, we are concerned about $x \in \mathbb{R}$ for $x > 0$ only. Apart from an explicit expression there is the important recurrence $\Gamma(x+1) = x\Gamma(x)$ with *initial condition* $\Gamma(1) = 1$ and $\Gamma(\frac{1}{2}) = \sqrt{\pi}$ for positive integers and positive half-integral values x , respectively.

³⁴By definition $0!! = 1!! = 1$.

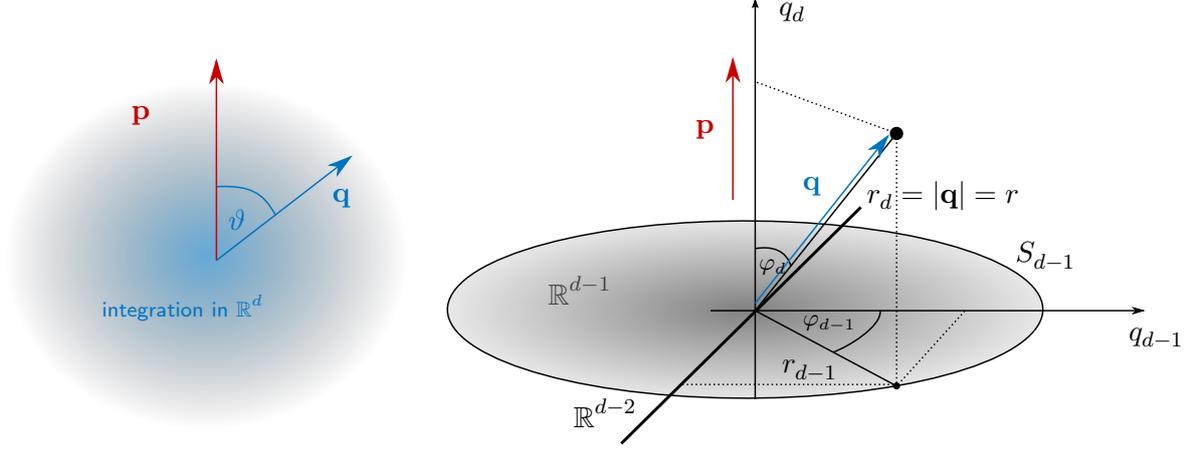


Figure 3.6: *Definition of hyper-cylindrical coordinates.* Sketch displaying our convention for parametrizing the d -dimensional (Euclidean) space. On the one hand we exploit the spherical symmetry (left) and on the other hand we employ Cartesian coordinates $\mathbf{q} = (q_1, \dots, q_d)$ and hyper-cylindrical coordinates $\varphi = (r, \varphi_d, \dots, \varphi_2)$, respectively, to handle the *symmetry-broken* situation due to *external momentum* \mathbf{p} .

$$\begin{aligned} q_i &= r_i \cos \varphi_i & \text{with } r_i &\in [0, \infty) & \text{for } i = d, d-1, \dots, 3, \\ r_{i-1} &= r_i \sin \varphi_i & \varphi_i &\in [0, \pi] \end{aligned} \quad (3.18)$$

and the initial condition reads

$$\begin{aligned} q_2 &= r_2 \cos \varphi_2 & \text{with } r_2 &\in [0, \infty) \\ q_1 &= r_2 \sin \varphi_2 & \varphi_2 &\in [0, 2\pi) \end{aligned} \quad (3.19)$$

$\mathbf{q} = (q_1, \dots, q_d)$ denotes (standard) Cartesian coordinates with $q_i \in \mathbb{R}$ and $\varphi = (r, \varphi_d, \dots, \varphi_2)$ where we set $r \equiv r_d = |\mathbf{q}|$. As fig. 3.6 depicts, the external momentum vector \mathbf{p} is aligned along the integration variable/axis q_d and the angle ϑ between vectors \mathbf{p} and \mathbf{q} is associated with φ_d .

The volume element transforms according to

$$d^d \mathbf{q} = |D_d| dr d\varphi_d \dots d\varphi_2 \quad \text{with } D_d \equiv \det \frac{\partial \mathbf{q}}{\partial \varphi} \quad (3.20)$$

Explicitly computing the *Jacobian* $\partial \mathbf{q} / \partial \varphi$ yields an almost triangular matrix with one additional non-zero (lower) off-diagonal. Applying Laplace's formula for determinants we end up with the recurrence

$$\tilde{D}_i = (-)^{i-1} \sin^{i-2} \varphi_i \tilde{D}_{i-1} \quad \text{for } \tilde{D}_i \equiv D_i / r^{i-1}, \quad i = d, d-1, \dots, 3, \quad (3.21)$$

where the initial condition becomes

$$\tilde{D}_2 = \begin{vmatrix} \sin \varphi_2 & \cos \varphi_2 \\ \cos \varphi_2 & -\sin \varphi_2 \end{vmatrix} = -1 \quad (3.22)$$

The spherical volume element directly follows as

$$d^d \mathbf{q} = r^{d-1} \sin^{d-2} \varphi_d \sin^{d-3} \varphi_{d-1} \dots \sin \varphi_3 dr d\varphi_d \dots d\varphi_3 d\varphi_2 \quad (3.23)$$

Given a function $f(\mathbf{q}) = f(r, x)$ that depends on the magnitude $r = |\mathbf{q}|$ (spherical symmetry ...) and an angle $x \equiv \cos \vartheta$ (... broken by \mathbf{p}) to a preferred direction only, we conclude

$$\text{tr}_2 \equiv C_2^{-1}(d) \int_{q_0} \int_0^\infty r^{d-1} dr \int_{-1}^1 dx (1-x^2)^{\frac{d-3}{2}} \quad (3.24)$$

$$\begin{aligned} \text{with } C_2^{-1}(d) &\equiv \frac{1}{(2\pi)^d} \int_0^{2\pi} d\varphi_2 \int_0^\pi d\varphi_3 \dots \int_0^\pi d\varphi_{d-1} \sin^{d-3} \varphi_{d-1} \dots \sin \varphi_3 \\ &= \frac{\Omega_{d-1}}{(2\pi)^d} = [2\pi C_1(d-1)]^{-1} \\ \hookrightarrow C_2(d) &= 2^{\lfloor \frac{d}{2} \rfloor} \pi^{\lfloor \frac{d+2}{2} \rfloor} (d-3)!! , \quad d > 2 \quad . \end{aligned} \quad (3.25)$$

The integral in C_2 characterizes the integration over a sphere of unit radius embedded into $(d-1)$ -dimensional Euclidean space. The explicit expression C_2 analogous to C_1 has an intuitive interpretation, see [fig. 3.6](#). For zero-temperature calculations (cf. [footnote 32](#)) we are in position to simplify tr for our flow equations, now. Notably we have in mind

1. [eq. \(2.136\)](#) to determine $U_k(\rho)$ (spherical case) and
2. e.g. [eq. \(2.184\)](#) for the (inverse) propagators $P_k(p)$ which singles out a direction specified by $p = (p_0, |\mathbf{q}|)$.

So far, `spatInt()` from `frg_trace.cpp` turns formula [eq. \(3.24\)](#) into an usable source code for the physically relevant cases $d = 3$ and $d = 4$, e.g. $T = 0$ non-relativistic or *classical* relativistic physics. Furthermore it captures $0 \leq d < 3$ whose formulae are well known and therefore straightforward to compute. The dimension d of the system is stored as variable of the `flowQuant` structure, namely: `flowQuant::spatDim`. Since the regulator function R_k typically cuts of the integration, `spatInt()` provides arguments to restrict the domain of $r = |\mathbf{q}| \in [q_{\min} > 0, q_{\max} \geq q_{\min}]$.

It remains to handle the Matsubara summation at $T \neq 0$. As previously mentioned, \int_{q_0} becomes a discrete summation of ω_n with

$$\Delta q_0 = \omega_{n+1} - \omega_n = 2\pi T \quad . \quad (3.26)$$

Theoretically the sum runs over infinitely many ω_n . However, it remains impossible to carry out this task in practice. Even worse: As $T \rightarrow 0$ the range $\Delta \equiv 2\pi T(N-1)$ covered by a fixed number of N successive ω_n shrinks down to zero. It becomes even impossible to numerically sum a finite Δ . We provide a solution by `matSum()` included in `frg_trace.cpp`. It works as follows:

1. Choose a set of N (arbitrary) ω_n that are compatible with bosonic/fermionic Matsubara frequencies³⁵.
2. Apply `spatInt()` (or whatever operation is necessary) in order to reduce $\text{STr} \dots$ to $T \sum_{\omega_n} \dots$.

³⁵Here, the index n refers to adjacent Matsubara frequencies ω_n . Below, j in n_j label the selected frequencies ω_{n_j} which do not necessarily satisfy $\delta\omega_j = \omega_{n_{j+1}} - \omega_{n_j} = 2\pi T$, but $\delta\omega_j = 2\pi T(n_{j+1} - n_j)$.

Note, that $S_{j-1}(\omega_{n_j}) = S_j(\omega_{n_j}) = s_{n_j} = a_{j-1,0}$. Reconstructing the Matsubara summation $T \sum_{\omega_n} \dots$ on the whole interval $[\omega_{n_1}, \omega_{n_N}]$ is expressed as

$$\mathcal{S} \equiv s_{n_1} + \sum_{j=1}^{N-1} \sigma_j \quad \text{with} \quad \sigma_j \equiv \sum_{\omega_n \in (\omega_{n_j}, \omega_{n_{j+1}}]} S_j(\omega_n) \quad . \quad (3.28)$$

Some algebraic massaging yields

$$\begin{aligned} \sigma_j &= \frac{1}{2\pi} \sum_{i=0}^3 \frac{(-)^i a_{ji}}{i+1} \delta\omega_j^{i+1} \cdot T^0 && \text{(zero temperature contribution)} \\ &+ \frac{1}{2} (a_{j1} - a_{j2} \delta\omega_j + a_{j3} \delta\omega_j^2) \delta\omega_j \cdot T && \text{(first order finite temperature correction)} \\ &+ \pi \left(\frac{1}{3} a_{j2} - \frac{1}{2} a_{j3} \delta\omega_j \right) \delta\omega_j \cdot T^2 && \text{(second order finite temperature correction)} \end{aligned} \quad (3.29)$$

where

$$\delta\omega_j \equiv \omega_{n_{j+1}} - \omega_{n_j} > 0 \quad (3.30)$$

is defined.

Some remarks are in order here.

$$\mathcal{S} = c_0 + c_1 T + c_2 T^2 \quad , \quad (3.31)$$

i.e. eqs. (3.28) and (3.29) got rid of evaluating numerically instable expressions of the form “ $0 \cdot \infty$ ” (cf. fig. 3.7). Instead, the zero temperature contribution explicitly reads c_0 and finite temperature corrections $c_{0 < i < M}$ are added on top. M depends on the degree of the interpolating polynomial which is larger by one. The generalization of eq. (3.29) involves the Bernoulli numbers^[AW05,Haz95], but adds no conceptual news to the idea itself—it rather blows up the explicit formula.

The shrinking interval $\Delta q_0 = 2\pi T$ as $T \rightarrow 0$ got replaced by fixed intervals $\delta\omega_j$. In fact, the first term in eq. (3.29) is simply the integration $\int_{q_0} = \frac{1}{2\pi} \int dq_0$ over the cubic splines eq. (3.27)³⁶. Only the *non-constant portion* of $S(q_0)$ yields finite temperature corrections, i.e. a_{j0} from the $S_j(q_0)$ do not contribute to $c_{i \neq 0}$. Metaphorically speaking, it is not *felt* by temperature. In essence we interpolated missing information $S(q_0)$ at $\omega_n \in (\omega_{n_j}, \omega_{n_{j+1}})$ by a set of spline coefficients a_{ij} to end up with the (semi-)analytic eq. (3.29). This approach avoids explicit summation. It saves us from evaluating increasingly many $S(q_0)$ or shrinking the interval covered by a finite number of ω_{n_j} when $T \rightarrow 0$, respectively.

Despite the advantage outlined so far it remains to wisely choose the n_j such that the cubic spline interpolation correctly approximates missing *intermediate* $n_j < n < n_{j+1}$. Since it strongly depends on the specific form of $S(q_0)$, the numerical evaluation should successively increase the total number N of ω_{n_j} and check for the convergence of \mathcal{S} .

³⁶The additional minus sign is due to the reversed definition of $\delta\omega_j$ compared to the $(q_0 - \omega_{n_{j+1}})$ -term in eq. (3.27).

3.3 Details on Implemented Classes of Physical Objects

The following subsections aim at a more detailed analysis of the main objects implemented for `libfrg` so far. In particular we are concerned about the effective potential U_k and the (inverse) propagator P_k which we are going to accurately resolve in momentum space. Moreover we spend some pages to illuminate several aspects of a proper choice for the regulator function R_k .

3.3.1 The Effective Potential – Γ_k at Constant Field

For us the effective potential $U_k(\rho)$ defined through

$$\Gamma_k[\eta] \Big|_{\psi=0, \varphi=\phi=\text{const.}} \equiv V_D U_k(\rho) \quad (3.32)$$

(cf. discussion following eq. (2.68)) is a real-valued function of a single³⁷, real-valued variable $\rho = \phi\phi^* \geq 0$. Without loss of generality one might set $\phi \in \mathbb{R}$ such that $\phi = \sqrt{\rho}$. The space-time volume V_D encapsulates the *collapse* of the summation/integration over all spatial-temporal degrees of freedom entering the value of Γ_k for a given field configuration η . As discussed at the beginning of section 2.3 as well as being sketched in section 1.1 (check derivation eqs. (1.21) to (1.24)) the *quantum equation of motion* singles out a ground state ρ_0 which is characterized by an *extremum principle*. When restricting to (spatially) constant field configurations as in eq. (3.32), this constraint translates to determining a (local) extremum $\partial_\rho U_k(\rho_0) = 0$. Then, the field configuration $\phi_0 = \sqrt{\rho_0} e^{i\alpha}$ with an arbitrary (fixed) α constitutes the (translationally invariant) *ground state/condensate* of the system. For $\rho_0 \neq 0$ we encounter SSB. No doubt: During the course of the flow ρ_0 will vary and hence we label the condensate *scale dependent*, i.e. $\rho_0 = \rho_{0k}$.

Transferring the illustration fig. 1.3 to $U_k(\rho)$ there is the option of either (Taylor) expanding the effective potential around e.g. ρ_{0k} or alternatively evaluating $U_k(\rho)$ on a grid of several values ρ_i . The class `EffPot` from `frg_effPot.cpp` provides a realization of both approaches. Nevertheless, there are the following (technical) advantages favoring the latter representation of U_k :

1. Given an arbitrary set of $\{\rho_i\}$ the flow of the corresponding $U_k(\rho_i)$ s are easily obtained: An expression for $\partial_k U_k(\rho)$ has to be derived only once. Afterwards one simply sets the (external) parameter ρ to ρ_i . In contrast, a Taylor expansion of $U_k(\rho)$ around some ρ_{0k} requires to *project* the flow equation to the scale dependent expansion coefficients which adds computational work for each of them.
2. Starting the flow at $k = \Lambda$ one needs to set an appropriate Taylor expansion point $\rho = \rho_{0\Lambda}$ such that $\partial_\rho U_\Lambda(\rho_{0\Lambda}) \equiv m_\Lambda^2 \geq 0$. For the ground state we choose the condensate $\rho_{0\Lambda}$ to reside where the global minimum of U_Λ is attained—a stronger condition than the extremum principle mentioned above. As long as $0 \leq m_k^2 > 0$, $0 \leq \rho_{0k} = 0$ has to be fulfilled. At a *second order phase transition* $m_k^2 = 0$ we need to explicitly enforce $m_k^2 = 0$ if it tends to decrease below zero during the course

³⁷When dealing with various (constant) bosonic fields, the effective potential becomes a scalar function depending on multiple variables ϕ_1, ϕ_2, \dots

of the flow. Then, the expansion point ρ_{0k} stays at a (variable) local minimum. As a result, ρ_{0k} starts to increase (SSB) according to a corresponding flow equation. Likewise, explicitly switching from SYM to SSB has to be reversed when ρ_{0k} approaches zero again. Note that the flow equation of m_k^2 becomes replaced by that of ρ_{0k} and vice versa. Moreover, we are bound to follow the local minimum we did choose to be global at the beginning of the flow ($k = \Lambda$). [Figure 4.1](#) from the following chapter illustrates the procedure of explicitly switching the flow equations $\dot{\rho}_{0k} \leftrightarrow \dot{m}_k^2$. All these issues vanish into thin air if we take U_k on a grid including a suitable interpolation scheme. Now, it is possible to follow the global trend of the effective potential up to a specified maximal ρ -value. Hence, we are enabled to identify *first order phase transitions* during the flow which we define³⁸ as discontinuous drops of the condensate value ρ_{0k} from a finite value to zero or the other way around.

To date an object instantiated from the class `EffPot` is capable to represent an effective potential of the form

$$U_k(\rho) = \sum_{n=0}^2 \frac{1}{n!} \partial_\rho^n U_k(\rho_{0k}) (\rho - \rho_{0k})^n = \begin{cases} -p_k + m_k^2 \rho + \frac{\lambda_k}{2} \rho^2 & , \rho_{0k} = 0 \text{ and } m_k^2 \geq 0 \quad (\text{SYM phase}) \\ -p_k + \frac{\lambda_k}{2} (\rho - \rho_{0k})^2 & , \rho_{0k} \neq 0 \text{ and } m_k^2 = 0 \quad (\text{SSB phase}) \end{cases} . \quad (3.33)$$

Here, p_k , m_k^2 or ρ_{0k} , and λ_k are the flowing quantities, cf. e.g. [section 4.1](#). It is conceptually straightforward to extend the numerics to higher order Taylor polynomials, but due to the previous [list](#) our focus is on the discretization of U_k . The class `EffPotFromTaylor` has been implemented in `frg_effPot.cpp` to represent a (generic) Taylor expansion around some ρ_{0k} . It is used by the `GSL` compatible function `effPotFromTaylor()` in order to supply the constructor `EffPot::EffPot()` with a function for determining suitable interpolation points $(\rho_i, U_k(\rho_i))$ on the grid to be discussed in a minute.

Explicitly switching between SYM and SSB mentioned by [item 2](#) from above calls for specific data structures in `EffPot` as well as an appropriate insertion to the procedural sequence of the source code that drives the flow. First of all there is the need for a routine that detects whether the Taylor expansion, [eq. \(3.33\)](#), indicates SSB or SYM. This issue is answered by `EffPot::getSYMT()`. For m_k^2 or ρ_{0k} exclusively becoming zero the decision is obvious, but how to treat the situation where both quantities vanish (*phase transition*)? Then it depends on the variation of the *mass parameter*, \dot{m}_k^2 as well as the *direction* of the flow, i.e. if k (or $t \sim \ln k$) increases or decreases. If k increases, a decreasing \dot{m}_k^2 needs to be compensated by an increase of ρ_{0k} . In this case $\dot{m}_k^2 < 0$ indicates flowing into the SSB phase. For increasing mass parameter we are left without a finite condensate. For decreasing³⁹ k the

³⁸Recall that, in the sense of [section 1.2](#), Γ_k (thus U_k) interpolates between the microscopic action $S = \Gamma_{k=\Lambda}$ and the full (quantum) effective action $\Gamma = \Gamma_{k=0}$. However, only Γ allows to identify whether the system exhibits SSB. The notion of quantities derived from $\Gamma_{k \neq 0}$ strongly depends on the choice of the regulator R_k (check [section 3.3.3](#)). To this end we do not deal with a phase transition in the conventional sense of condensed matter physics.

³⁹This corresponds to our setting where $k = \Lambda \rightarrow 0$.

whole story is the other way around. Summing up⁴⁰:

$$\rho_{0k} = m_k^2 = 0, \quad \begin{cases} \dot{m}_k^2 \geq 0 \rightarrow \text{SYM} & \text{for increasing } t \\ \dot{m}_k^2 < 0 \rightarrow \text{SSB} & \text{_____} \\ \dot{m}_k^2 \leq 0 \rightarrow \text{SYM} & \text{for decreasing } t \\ \dot{m}_k^2 > 0 \rightarrow \text{SSB} \end{cases} \quad (3.34)$$

The nature of numerics implies the continuous variation of the scale parameter k to become discrete. Therefore it will be a rather rare event for m_k or ρ_{0k} to exactly hit zero from above. It is this insight that introduces two threshold values `EffPot::M2Trans` and `EffPot::Rho0Trans` which are used by `parallRoutine()` (check [section 3.2.2](#)) through `frgFlow2()` (cf. [section 3.2.1](#)) when evolving the flow. More precisely, there is the routine `stepEstimate()` in `frg_flow.cpp` that aims at estimating a new step width for iteratively approaching a *critical evolution step*. When m_k^2 or ρ_{0k} drop below zero we need to decrease the step width such that $0 \leq m_k^2 < \text{EffPot::M2Trans}$ or $0 \leq \rho_{0k} < \text{EffPot::Rho0Trans}$ is fulfilled. The idea is sketched in [fig. 3.8](#).

Concerning a suitable interpolation for the discrete set of $(\rho_i, U_k(\rho_i))$ one may consult *one-dimensional (natural) cubic splines* [[Haz95,Ran06,PFTV92](#)]. However, the structure of the flow equation is such that $\dot{\Gamma}_k = \beta_k [\Gamma_k^{(2)}]$. Since we are unable to directly discretize the functional $\Gamma_k[\eta]$ which would involve interpolating $\Gamma_k[\eta_i]$ where the index i labels infinitely many variables⁴¹, we are bound to select a suitable truncation first. If manageable, we are free to discretize objects of this truncation, e.g. as the effective potential $U_k(\rho)$ or the (inverse) propagator $P_k(q)$. The need for a truncation as a first step typically involves projection prescriptions to derive the flow of $\Gamma_k^{(n \geq 2)}$. The projection utilizes functional derivatives on both sides of the flow equation. Schematically written:

$$\dot{\Gamma}_k[\eta] = \beta_k [\Gamma_k^{(2)}[\eta]] \quad \rightarrow \quad \frac{\delta^n}{\delta \eta^n} \dot{\Gamma}_k[\eta] = \dot{\Gamma}_k^{(n)}[\eta] = \beta_k^{(n)} [\Gamma_k^{(2)}[\eta], \Gamma_k^{(3)}[\eta], \dots, \Gamma_k^{(n+2)}[\eta]] \quad . \quad (3.35)$$

Therefore, we will need to compute derivatives $\partial_\rho^n U_k(\rho_{0k})$ of the effective potential beyond $n = 2$. Any polynomial of *degree*⁴² n in ρ has vanishing derivatives $\partial_\rho^{m > n}$. Since cubic splines establish continuous 2nd derivatives at most, they are not primarily suited to represent a reasonable interpolation for U_k at $\rho \neq \rho_i$. For sure, we might consider *quartic, quintic, ...* splines, in order to increase the *smoothness* at the ρ_i , but the corresponding computational effort considerably increases [[Ran06,PFTV92](#)].

There is another option associated with the name *Chebyshev*, a Russian mathematician of the 19th century. In a nutshell: Polynomial interpolation faces a severe thread when it comes to convergence properties with increasing degree. Given a function $f(x) \in \mathbb{R}$ the *error* $|f(x) - p_n(x)|$ of a polynomial $p_n(x)$ of degree n strongly depends on the choice of the n *coordinates* (x_i, f_i) it has to interpolate through, namely $p_n(x_i) = f_i$. Even if f is sufficiently smooth, p_n does not need to converge to f

⁴⁰Note, that we do assign the ambiguous case $\dot{m}_k^2 = 0$ to SYM by definition. It is equally justified for `EffPot::getSYMT()` to indicate SSB in this situation.

⁴¹For instance, $\eta = \eta(x)$ —a set of ininitely many variables. Even if we discretize x we are left with a function $\Gamma_k[\eta_i]$ with N variables whose values have to be discretized again.

⁴²Here, the term *degree* n of a polynomial $p_n(x)$ refers to the maximal power of the variable x in p_n .

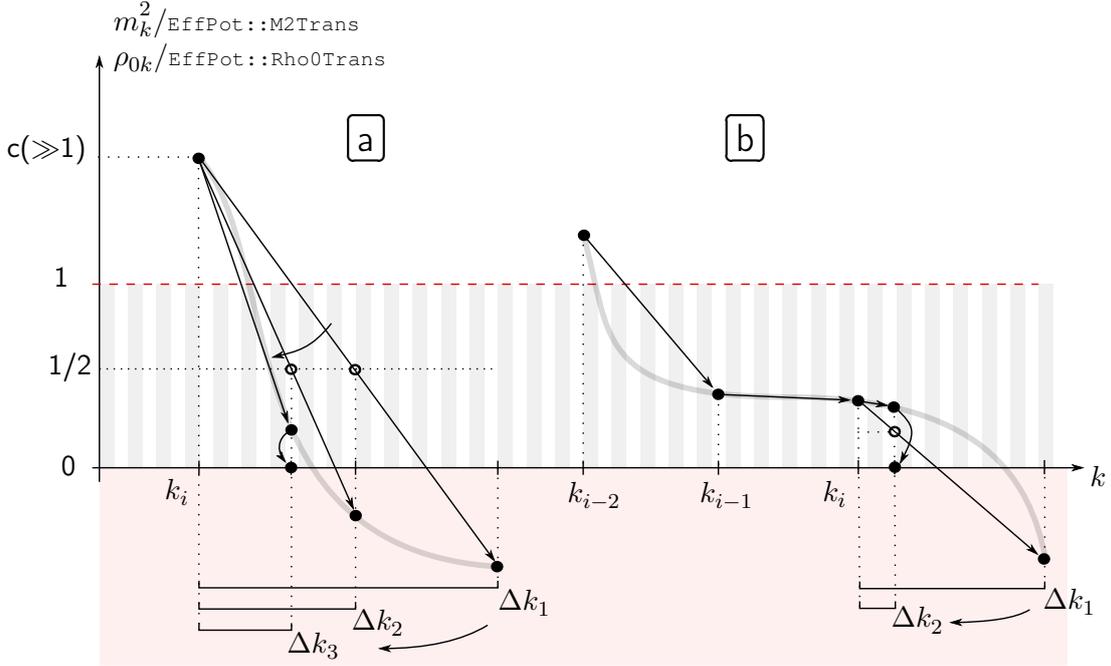


Figure 3.8: *The action of `stepEstimate()` for the explicit switching $SYM \leftrightarrow SSB$ when Taylor expanding U_k . The requirement for the mass parameter m_k^2 (or the condensate ρ_{0k}) not to drop below zero introduces some numerical effort when it comes to advancing this quantity where the scale parameter k varies according to (adaptive) discrete steps Δk . To this end we (the user of `EffPot`) have to set two *threshold* values `EffPot::M2Trans` and `EffPot::Rho0Trans` (red dashed line, $---$) when an instance of the effective potential class is invoked. The function `parallRoutine()` employed by the routine `frgFlow2()` that drives (straight arrows, \rightarrow) the flowing quantities (filled dots, \bullet) recognizes the scale k_i where m_k^2 (or ρ_{0k}) becomes negative (flimsy shaded region in red) when being advanced by Δk_1 (horizontal square bracket, \longleftarrow). It then calls `stepEstimate()` in order to obtain a new guess, Δk_2 . Afterwards, the flowing quantity is recalculated at $k_i + \Delta k_2$. This procedure is iterated as long as the mass parameter or the condensate becomes non-negative.*

Essentially there are two qualitative distinct situations. a) Either the flowing quantity drops from above the threshold or b) from below. Note, that as long as the mass parameter (condensate) stays positive no *phase transition* is triggered and hence `stepEstimate()` is not called at all. The flowing quantity is allowed to *dive* below the threshold without $SYM \leftrightarrow SSB$. However, as soon as it becomes negative, a phase transition is detected (case b)). Since we need $\Delta k > 0$ to perform a (finite) step at all, an estimate Δk is suggested by `stepEstimate()`. It relies on linearly interpolating between k_i and $k_i + \Delta k_1$ such that half of the (positive) value of the flowing quantity at k_i is approached (black circle, \circ). The procedure is repeated up to the point where advancing the flowing quantity actually yields a value above or equal to zero. When the flowing quantity is out of the threshold region (gray bold solid bars)—before becoming negative—it is allowed to be much larger in magnitude than the threshold itself (case a), $c \gg 1$). Now `stepEstimate()` aims at suggesting an appropriate Δk to *catch* the flowing quantity within the threshold region near half of the threshold value. As before, the procedure is iterated until the negative value of the flowing quantity at $k_i + \Delta k$ becomes non-negative again.

in the limit $n \rightarrow \infty$. In particular, equidistant $x_i \in [-1, 1]$ constitute a terribly bad choice for the approximation of $f(x) = (1 + 25x^2)^{-1}$. This is known as *Runge's phenomenon*^[Run01]. When selecting the x_i to match the roots or (local) extrema⁴³ of the *Chebyshev polynomial*⁴⁴

$$T_n(x) \equiv \cos[n \arccos x] \quad (3.36)$$

exponential decrease of $|f - p_n|$ for $n \rightarrow \infty$ is ensured for f *Lipshitz continuous*^[Haz95, Tre13]. The unique polynomial $p_n(x)$ might be decomposed by a set of n *orthonormal base polynomials* (see [appendix C](#)). One option is to use the $T_n(x)$, i.e.

$$p_n(x) = \sum_{i=0}^{n-1} c_i T_i(x) \quad (3.37)$$

with coefficients c_n . When learning about all that we discovered [Tre13]⁴⁵ as an inspiring source to the topic. An (exciting) survey of the story collecting knowledge from literature as well as explicit computations from our own hands is given in [appendix C](#). To whet your appetite: Under the spell of Laurent series from complex analysis there is a quite unifying approach to the convergence properties of (real-valued) interpolating polynomials to a given function f . There is a neat physical analogy to electrostatics that rules these convergence properties.

However, in practice we (partially) employ the *GSL* library to implement the interpolation on the Chebyshev grid, in this case, defined by zeros of [eq. \(3.36\)](#). The routines `gsl_cheb_*()` follow the line of [Bro73]. There, the user needs to supply a function f to be approximated. Since we might have to exclusively rely on the discrete set $\{(\rho_i, U(\rho_i))\}$, we directly perform the computation of the coefficients c_n from [eq. \(3.37\)](#) by `EffPot::interpolU()` equivalently to the procedure in `gsl_cheb_init()`, but without explicitly retrieving f at the Chebyshev points x_i .

The naive computation of the c_n (cf. [appendix C](#)) with $n = 0 \dots N - 1$ amounts for computational effort of order $\mathcal{O}(N^2)$. This is our approach and that of the corresponding *GSL* routines at the moment. Nevertheless, there is the opportunity for enhancing performance to complexity class $\mathcal{O}(N \log N)$, if one realizes that the derivation of the c_n from the set of $\{U_k(\rho_i)\}$ is closely related to the *Fast Fourier Transform*^[CT65] (touched at the end of [appendix C](#)). More precisely, one should consult the *Fast Cosine Transform*^[ANR74] (of type II). Although not at the top of our priority list⁴⁶ for developing `libfrg`, the corresponding source code extension might become realized in the future.

After the initialization of the coefficients c_i by `EffPot::interpolU()`, `EffPot::getU()` is able to return any interpolated value $U_k(\rho)$ for $0 \leq \rho \leq \text{EffPot::rhoMax}$. The routine `getUGSLFunc()`

⁴³ ... which are the roots of the derivative $T'_n(x) = \partial_x T_n(x)$. Literature^[Haz95] denotes $U_n(x) = \frac{1}{n+1} T'_{n+1}(x)$ as *Chebyshev polynomials of the second kind*.

⁴⁴The German language writes the Russian character **Т** (international phonetic alphabet: **te**) at the beginning of the Russian name *Чебышев* in Latin letters as **Tsch**. Phonetically it is equivalent to the **ch** in the English word *chip*. Therefore the letter *T* for the polynomial $T_n(x)$. [Appendix C](#) justifies that $T_n(x)$ is indeed a polynomial of degree n .

⁴⁵As of Sept. 2013, the book is available chapter by chapter at <http://www2.maths.ox.ac.uk/chebfun/ATAP/> as *MATLAB* *.m files which one can generate L^AT_EX documents from.

⁴⁶We judge it much more important to care about the time-consuming integration/summation from *S*Tr we present in [section 3.2.4](#).

from `frg_effPot.cpp` provides a *GSL* compatible wrapping function (data type `gsl_function`) to `EffPot::getU()`. We made it available for use in conjunction with any other *GSL* routine which take it as an argument.

The discrete values of U_k are modified by `EffPot::setUn()`. Note, that the Chebyshev grid fixes the ρ_i to specific values in $[0, \text{rhoMax}]$; one is not allowed to arbitrarily vary them. One rather has to supply the constructor of `EffPot` with a maximal value for ρ that becomes internally stored as `EffPot::rhoMax`. It is allowed to be called/modified by appropriate `get*()/set*()` routines. `EffPot::getRhon()` provides the option to retrieve the ρ_i value depending on the `EffPot::rhoMax` previously set.

Calling `EffPot::getDerU()` incorporates `gsl_cheb_calc_deriv()` from the *GSL* library to derive any $\partial_\rho^n U_k(\rho)$ with $0 \leq \rho \leq \text{rhoMax}$. There is an efficient scheme that allows to determine the coefficients c'_n of the derivative $p'_n(x) = \partial_x p_n(x)$ due to a recursion relation among the $T_n(x)$ and $T'_n(x)$. Details are spread in [appendix C](#). Instances of `EffPot` automatically keep track of previously calculated coefficients by internal validity flags in order to avoid unnecessary recalculation. In general, the updating strategy for invalid coefficients is performed not before explicit usage. Actually, in the case where one retrieves a specific interpolating value, updates take place. We adopt such a *conservative* strategy in favor of time efficiency.

3.3.2 The (inverse) Propagator – Quadratic Fluctuations of Γ_k

A lesson to be learned from [section 3.2.2](#), especially [fig. 3.4](#), is the fact that the user of `libfrg` significantly contributes to the code's runtime efficiency. The reason is related to the resource intensive computation of the STr-operation. It massively makes use of function calls of pointer type `(frgInt*)()` as apparent from [fig. 3.1](#). Therefore it is advisable to focus on speed when implementing a two-dimensional grid in order to support the discrete values π_n of any non-relativistic, inverse propagator of the form $P(q) = P(q_0, \mathbf{q}^2) \in \mathbb{C}$. On the opposite face of the coin there is the desire to interpolate the (inverse) propagator *as a whole*. To be more specific: Consider a one-dimensional function $f(x)$ that crosses through given coordinates $(x_n, f(x_n))$ on the interval $[a, b]$. A (natural) cubic spline $s(x)$ is characterized by coefficients of polynomials⁴⁷ of degree three. They are determined such that the whole curve follows a principle of *minimal bending* in the sense that the *cumulated curvature* $\int_a^b dx [\partial_x^2 s(x)]^2$ becomes minimal^[Ran06, Haz95]. Therefore there is a criterion that specifies the approximation $s(x)$ of $f(x)$ *as a whole*. By the way: The minimization property of the cumulated curvature is the reason that reduces oscillation effects like Runge's phenomenon mentioned when discussing the interpolation of the effective potential $U_k(\rho)$ on a grid in [section 3.3.1](#).

According to has been said we develop an appropriate interpolation scheme for two-dimensional surfaces, `getZFrom2DSpline()` from `frg_spline.cpp`. The routine is based on one-dimensional (natural) cubic splines (`nat_spline1D()`) in order to account for the global aspect of the given set of discrete π_n -values. However, our approach with emphasize on speed optimization is different from *bicubic spline interpolation*^[PFTV92].

⁴⁷Some practically useful facts to keep in mind when considering approximations by polynomials is entertainingly presented in [\[Tre11\]](#).

The `Propagator` class—whose source code resides in `frg_prop.cpp`—splits P_k into its real and imaginary part. The arbitrary, but increasingly ordered q_0 and \mathbf{q}^2 values are stored by private arrays `w[]` of size N and `q[]` of size M , respectively. Both become established at the time of construction of a `Propagator` instance. Various methods⁴⁸ as e.g. `Propagator::set{W|Q}()` provide tools to modify the grid defined via `w[] × q[]`. However, N and M stay fixed. The matrix structure that stores the complex valued π_n ($n = 1, \dots, N \cdot M$) derives from an array of array pointers that point to pointers of data type/structure `complex<double>`. This construct is initiated by `createProp()` from `frg_struct.cpp`.

Moreover, the `Propagator` object maintains several attributes that reflect the validity of the instance depending on e.g. the order of the values in `{w|q}[]` or consistency of spline coefficients for (polynomial) interpolation. `Propagator::getFullStatus()` retrieves the validity of the `Propagator` instance based on all these flags. `bool Propagator::get*()` methods rely on status information for determining their evaluation success to be returned.

A set of `Propagator::plot*()` routines allow for writing out the (inverse) propagator's state to files. In addition, the `Propagator` class inherits from the abstract base class `FlowObj`. Thus it implements all (virtual) routines as (partially) described in [section 3.2.1](#). Therefore, recording snapshots of a `Propagator` object is almost automatically done. If one needs to duplicate a given instance of a propagator the class provides `Propagator::copyP()` for deep copies.

Since the integration involves arbitrary values of $P(q)$, there is the call for a scheme to interpolate the given values π_n which approximate P_k . For us we develop a method that incorporates a set of one-dimensional (natural) cubic splines^[Ran06]. A fictive (inverse) propagator is plotted in [fig. 3.9](#); it summarizes the key ingredients on which we build to represent $P(q)$.

The general procedure in order to approximate a two-dimensional surface by some polynomial starts with an ansatz (substitute $q_0 \rightarrow x$ and $\mathbf{q}^2 \rightarrow y$)

$$P(x, y) = \sum_{i,j} c_{ij} x^i y^j \quad . \quad (3.38)$$

Imposing certain *boundary conditions* specifies the set of coefficients c_{ij} . For simplicity we restrict P to be real valued, here. For $P \in \mathbb{C}$ we split it into real and imaginary part and separately perform the procedure we are going to discuss in subsequent paragraphs.

A task that renders the key question, might be formulated as follows:

Given four values at the edge of a square $[0, 1] \times [0, 1]$, interpolate them for (x, y) inside the square.

Since we are left with the four conditions $P(0, 0) = \pi_{00}, \dots, P(1, 1) = \pi_{11}$, we are restricted to a (bi)linear ansatz⁴⁹ where $i, j = 0, 1$ in [eq. \(3.38\)](#). Given a whole grid we could separately repeat this procedure for each square with edges $\pi_{nm}, \dots, \pi_{n+1m+1}$. Here, we substituted n from π_n by (n, m)

⁴⁸Details are documented directly in the code. They are formatted to match *doxygen's* convention.

⁴⁹In general, there is no (two-dimensional) plane that interpolates four arbitrary points in three-dimensional space. But the bilinear ansatz incorporates a quadratic component: Take the term xy for $x = y$.

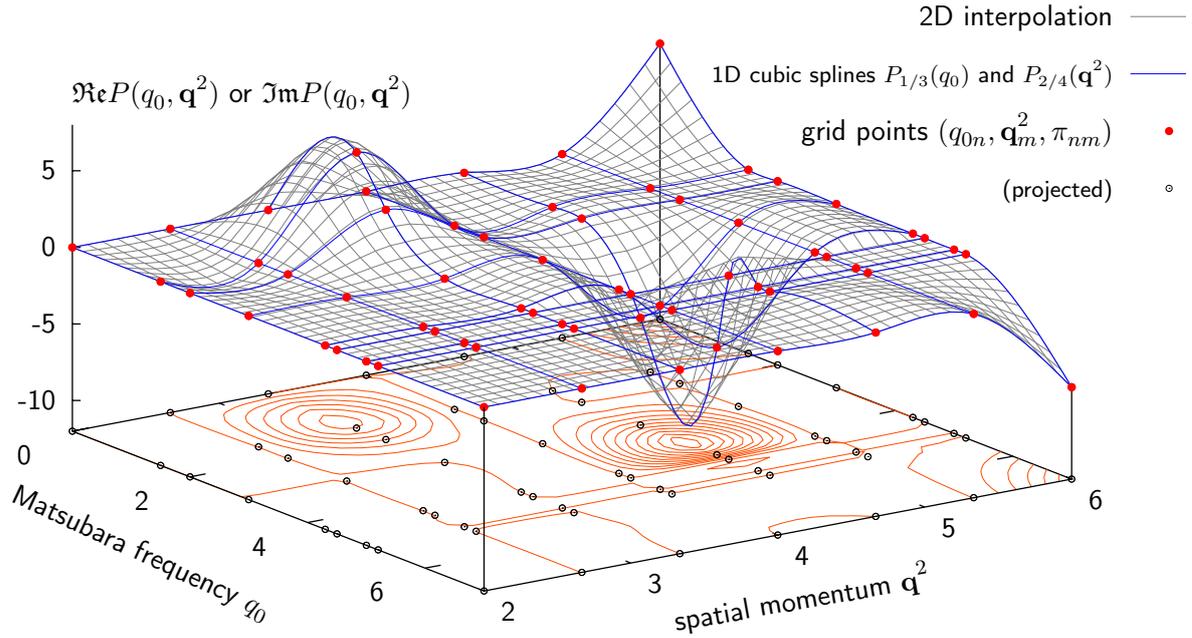


Figure 3.9: *Grid for the momentum-resolved (inverse) propagator.* The plot demonstrates our conceptual approach (no physical data) to interpolate the grid of the discrete (inverse) propagator values $(q_{0n}, \mathbf{q}_m^2, \pi_{nm})$. First of all we split the π_{nm} into their real and imaginary part. Therefore we obtain two sets of three-dimensional grid points. We present one of them by (red) filled dots. Their projection $(q_{0n}, \mathbf{q}_m^2, \text{const.})$ is printed as (black) empty dots along with a (orange) contour plot of the two-dimensional interpolation. We mark the one-dimensional (natural) cubic splines $P_{1\dots 4}$ by (blue) lines. They are computed such that they pass through the (red) grid points. Then, we explicitly derive the two-dimensional interpolation (gray lines) for each patch enclosed by four segments of these splines on the basis of eq. (3.42). Note that the variation of a π_{nm} -value affects two of the one-dimensional (natural) cubic splines. Accordingly, they have to be updated for a valid two-dimensional interpolation. The state of the one-dimensional splines is recorded by appropriate bool arrays $\{w|q\}\text{Spl}[\]$ used by `Propagator::interpP()` in order to refresh the spline coefficients stored by $\{w|q\}\text{SplA}\{0-4\}\{\text{Re}| \text{Im}\}P[\]$.

in order to differently label the x - and y -direction, i.e. we deal with the set of discrete coordinates $\{(x_n, y_m, \pi_{nm})\}$.

However, by incorporating the boundary values of the squares only, we restrict to *local* information. Surrounding points get involved when we consider polynomials beyond the linear order. For example, we could choose $i, j = 0, \dots, 3$; the so called *bicubic* interpolation. There are $4^2 = 16$ coefficients for which we need corresponding conditions in order to solve a system of linear equations obtained from eq. (3.38). We could do so, if we include derivatives of $P(x, y)$ into several directions, e.g. at the edges of the square in question. They have to be approximated in some way by surrounding grid points, e.g. through *finite differences* such as $\partial_x P(x_n, y_m) \approx \frac{\pi_{nm} - \pi_{n-1m}}{x_n - x_{n-1}}$. In the end, this procedure—called *bicubic interpolation*—amounts to perform matrix inversions. If the partial derivatives are obtained via one-dimensional spline interpolation, literature commonly refers to the algorithm as *bicubic spline*

interpolation.

As we are interested in both, the *global* aspect of the interpolation on the one hand and speed optimization on the other hand, our approach is slightly different, i.e. we establish one-dimensional (natural) cubic-splines through the grid with coordinates defined by the discrete $\{q|w\} []$ values, cf. [fig. 3.9](#). From this it remains to interpolate the interior of a square given the values at the boundary. In contrast to bicubic spline interpolation we do not exclusively use one-dimensional (natural) splines to determine derivatives at the grid points. The whole information from the spline's polynomials become involved. More precisely, the square enclosing the coordinate (q_0, \mathbf{q}^2) for which $P(q)$ is to be approximated, is incorporated into our interpolation scheme:

First of all, let us reparametrize the square $(x, y) \in [w[i], w[i+1]] \times [q[i], q[i+1]]$ such that $(x, y) \rightarrow (x, y) \in [0, 1] \times [0, 1] = [0, 1]^2$. We label the functions at the four boundaries by

$$P_1(x) \equiv P(x, 0) , \quad P_2(y) \equiv P(1, y) , \quad P_3(x) \equiv P(x, 1) , \quad \text{and} \quad P_4(y) \equiv P(0, y) \quad . \quad (3.39)$$

Fixing y , a straightforward interpolation from $P_4(y)$ to $P_2(y)$ reads $(1-x)P_4(y) - xP_2(y)$. The same procedure might be repeated for $P_1(x)$ and $P_3(x)$ at constant x . A naive guess for the whole interpolation scheme of $P(x, y)$ might be written down as

$$P(x, y) \sim [(1-y)P_1(x) + yP_3(x)][(1-x)P_4(y) + xP_2(y)] \quad . \quad (3.40)$$

It remains to account for the unwanted constant factors from the first or second term in square brackets at the boundary of $[0, 1]^2$. Defining

$$a \equiv P_1(0) = P_4(0) , \quad b \equiv P_2(0) = P_1(1) , \quad c \equiv P_3(1) = P_2(1) , \quad \text{and} \quad d \equiv P_4(1) = P_3(0) , \quad (3.41)$$

an educated guess leads to

$$P(x, y) = \frac{[(1-y)P_1(x) + yP_3(x)][(1-x)P_4(y) + xP_2(y)]}{(1-y)[(1-x)a + xb] + y[(1-x)d + xc]} \quad . \quad (3.42)$$

We are left with the inspection of zeros of the denominator. To this end, let us rewrite this expression as follows:

$$a + \beta x(1-y) + \delta y(1-x) + \gamma xy \quad \text{with} \quad \beta \equiv b - a , \quad \gamma \equiv c - a , \quad \text{and} \quad \delta \equiv d - a \quad . \quad (3.43)$$

Without loss of generality we are free to choose the constant a to represent the minimum of all the values at the edges, i.e.

$$a \equiv \min \equiv \min\{a, b, c, d\} , \quad \text{and} \quad \beta, \gamma, \delta \quad \text{become non-negative.} \quad (3.44)$$

Note that a *global* shift of $P(x, y) \rightarrow P(x, y) + \Delta$ does not affect the values β, γ , and δ . Since the coordinate (x, y) is taken from the *unit square* $[0, 1]^2$, we set the shift to

$$\Delta = \max - 2a \quad \text{with} \quad \max \equiv \max\{a, b, c, d\} \quad . \quad (3.45)$$

Hence, the minimal value of the denominator becomes $max - min \geq 0$. The case where $min - max$ vanishes refers to the trivial interpolation $P(x, y) \equiv 0$. Due to the *global* nature of our proposed interpolation scheme, it might happen that, although $max - min = 0$, the one-dimensional splines from eq. (3.39) do not vanish identically. Therefore we extend max and min to include the whole boundary of $[0, 1]^2$ to be interpolated. Since we are dealing with cubic polynomials $P_{1/3}(x), P_{2/4}(y)$, we are able to explicitly read off the positions x_e of local extrema. Assuming a polynomial $p_3(x) = \sum_{i=0}^3 a_i x^i$ the constraint $\partial_x p_3 = 0$ leads to

$$\begin{aligned} a_3 \neq 0: \quad x_{e\pm} &= -\frac{1}{3a_3} \left(a_2 \pm \sqrt{a_2^2 - 3a_1a_3} \right) \\ a_3 = 0, a_2 \neq 0: \quad x_e &= -\frac{a_1}{2a_2} \\ a_3 = a_2 = 0: \quad a_1 = 0 &\rightarrow \exists! x_e \quad . \end{aligned} \quad (3.46)$$

The non-relativistic symmetry $P^*(q_0, \mathbf{q}^2) = P(-q_0, \mathbf{q}^2)$ (cf. eq. (2.107)) discussed in section 2.1.5, restricts the numerical implementation of the (inverse) propagator to $q_0 \geq 0$ (and, of course, $\mathbf{q}^2 \geq 0$). P -values for $q_0 < 0$ given to `Propagator::getPwq()` are derived by means of this property.

When counting the total number of coefficients that store the information contents of $P(q_0, \mathbf{q})$, we obtain: $2 \cdot 4 \cdot [(N-1)(M-1) + N + M - 2]$, i.e. we have to interpolate the real and imaginary part of P by N plus M splines that are represented by $4(M-1)$ and $4(N-1)$ coefficients, respectively. The complexity of this algorithm is characterized by $\mathcal{O}(N \cdot M)$: The determination of a single one-dimensional (natural) cubic spline through M and N coordinates ($x_n, y_m = \text{const.}$) and ($x_n = \text{const.}, y_m$) takes time $\mathcal{O}(M)$ and $\mathcal{O}(N)$, respectively. The routine `Propagator::interpP()` performs this task in practice. Attention: After constructing an object from the `Propagator` class, one has to explicitly call `interpP()` in order to establish valid spline coefficients. If the function's argument `full` is set to `true`, all splines are updated, irrespective whether their state represented by the private arrays `{w|q}Spl[]` indicates the validity of the spline or not. For `full=false` only those splines which got modified by a previous variation of a corresponding grid value π_{nm} become computed again. Such an update involves $2 \cdot 4 \cdot (N+M-2)$ spline coefficients saving considerable amount of compute power compared to the computation of all splines.

Whenever an arbitrary value of $P(q)$ is retrieved via `Propagator::getPwq()` a refresh of the outdated splines is automatically established. We implement the search for the correct (discrete) coordinate (n, m) —such that (x, y) fulfills $x_n \leq x < x_{n+1}$ and $y_m \leq y < y_{m+1}$ —according to a partition scheme which uses nested intervals as follows: By means of the discrete values $x_0 = x_{\min} < x_1 < \dots < x_n < \dots < x_N = x_{\max}$, cut the interval $[x_{\min}, x_{\max}]$ in half and recursively repeat this step for the new interval that contains x as long as $x_n \leq x < x_{n+1}$. The same procedure is repeated for y . Therefore we end up with an additional contribution to the time complexity, namely $\mathcal{O}(\log N + \log M) = \mathcal{O}(\log N \cdot M)$.

3.3.3 The Regulator – Driving the Flow

As introduced in section 1.2 the regulator R_k for (bosonic) degrees of freedom φ is merely used to smoothly connect the (full) effective action $\Gamma_k[\phi]$ to its *classical* counterpart, $S_k[\varphi]$ in the sense of eqs. (1.39)

and (1.43). We briefly reported in footnote 10 of chapter 1 that, aside the mild restrictions eq. (1.36), additional arbitrariness in the choice of R_k emerges when turning to a (d -dimensional) field theory with (infinitely many) degrees of freedom $\varphi(x)$. The natural extension of the substitutions, eq. (1.38), consists in replacing $\frac{1}{2}R_k\varphi^2$ by

$$\frac{R_k}{2} \int_x \varphi^2(x) \quad \text{or} \quad \frac{R_k}{2} \int_q \varphi(q)\varphi(-q) \quad (3.47)$$

when switching to momentum space; a reasonable choice to be discussed below. However, the extension from φ to $\varphi(x)$ adds more ambiguity to the choice of the regulator, i.e. instead of a single quantity R_k we are dealing with a function $R_k(q)$, now. For instance, we are allowed to individually *weight* the Fourier modes $\varphi(q)$ according to

$$\frac{1}{2} \int_q R_k(q)\varphi(q)\varphi(-q) \equiv \frac{k^2}{2} \int_q r_k(y)\varphi(q)\varphi(-q) \quad (3.48)$$

with a scale k dependent, dimensionless *shape function* $r_k(y)$ where⁵⁰ $y = q/k$. At a theoretical level this freedom is welcome since it does not affect the flow equation in the sense that $\Gamma_{k=0} = \Gamma$. As long as eq. (1.36) holds for all x the relation to the microscopic theory, eq. (1.43), remains valid.

Nevertheless, in practice this ambiguity comes along with a (*painful*) price to pay. Although the flow equation is an exact statement, in general, truncation schemes spoil it. We already discussed some aspects of this challenge in section 1.2 which provided a bold motivation to set up the numerical framework implemented by the library `libfrg`. Different choices of R_k result in different *trajectories* $(\Gamma_{k,1}, \dots, \Gamma_{k,n}, \dots)$ ⁵¹ traversed from the microscopic model encoded in $S_\Lambda[\varphi]$ to the full quantum theory represented by $\Gamma_{k=0}[\phi]$. Since it is impossible to capture the most general form of $\Gamma_k[\phi]$ with a finite set of flowing quantities, the flow of the discrete $\Gamma_{k,n}$ will miss contributions adding to the full result $\Gamma_{k=0}[\phi]$. An optimal choice of R_k minimizes such deviation.

Despite spending quite some effort at the beginning of the 2000s towards a reasonable definition for an *optimal* regulator^[Lit01b,Lit01a,Lit01c], the subject remains challenging. The term *optimal* is associated with whether R_k yields results *close to the physical* $\Gamma[\phi]$ or not within a given truncation. Technically, the reason for the complexity of providing a suitable measure consists in the infinite number of degrees of freedom x which renders pinning down the function $R_k(x)$ a tough task. The only restriction at hand being eq. (1.36) and optionally the optimization criterion formulated within the references cited a moment ago. Loosely speaking, one brands R_k optimal if it modifies the (inverse) propagator $\Gamma_k^{(2)}[\phi] + R_k$ such that its minimal value (when being evaluated at some field ϕ) becomes maximal for varying q in the momentum-space representation. The idea behind this criterion might become more transparent if we

⁵⁰ $y = q/k$ has to be understood in a rather symbolic way, i.e. y represents all frequency/momentum components made dimensionless by appropriate powers of k . In our case of non-relativistic physics with spatial isotropy we have $q = (q_0, |\mathbf{q}|)$ and therefore $y = (q_0, \mathbf{q}^2)/k^2$, for example.

⁵¹ Remember the notion of the flowing quantities $\Gamma_{k,n}$ from section 3.2.1 where they are interpreted as components of a vector which parametrizes the truncation of the effective action Γ_k .

recall the flow equation

$$\dot{\Gamma}_k[\phi] = \frac{1}{2} \text{STr} \underbrace{\left[\Gamma_k^{(2)}[\phi] + R_k \right]^{-1}}_{G_k[\phi]} \dot{R}_k \equiv \beta_k[\phi] \quad (3.49)$$

again. Let us focus on the role of R_k . Its impact on β_k , hence the variation of Γ_k , is twofold. On the one hand it modifies the inverse propagator and therefore offers the ability to remove zeros of $\Gamma_k^{(2)}$, *i.e.* poles/singularities of the propagator which have to be integrated over by means of the super-trace operation. On the other hand \dot{R}_k multiplies G_k permitting limits on the integration/summation range of the operator STr ; the situation is depicted in [fig. 3.10](#). It is the former aspect that inspired Litim's optimization condition: Taking the momentum dependence of $\Gamma_k^{(2)}$ to be encoded in $P_k(q)$, zeros of this object yield poles to be integrated over to obtain the *velocity* β_k . Thus, $P_k(q) = 0$ is related to a numerically demanding situation implying the challenge of instabilities originating from “1/0”. Therefore it is desirable to *cure/remove* these infinities *as much as possible*, *i.e.* maximize the minimal value of $|P_k(q) + R_k(q)|$ with respect to q .

Despite its simplicity and general formulation, in general, the optimization criterion stated above is neither trivial to satisfy nor it singles out a unique R_k when it comes to application. First of all it strongly relies on the shape of $P_k(q)$. If one approaches a physical problem with a given functional form of $P_k(q)$ with flowing parametrization (as it is not the case for us) one might succeed in specifying a suitable $R_k(q)$. However, one needs to parametrize the regulator by a single quantity to find an optimized value. As soon as $\Gamma_k^{(2)}[\phi]$ involves more than $P_k(q)$ the story becomes complicated even more. The bottom line we would like to stress: Results from (practically) applying the functional renormalization machinery have to be treated with care. They should include a reasonable portion of physical background. To our knowledge there is no generic recipe that allows for *blindly* (and safely) running the flow. A pragmatic method to gain some intuition on the stability of a specific regulation scheme set by R_k provides *the principle of minimum sensitivity*^[BHLM95,Paw07]. It follows the heuristic idea:

“Let your regulator depend on (several) parameter(s) α . If the flowing quantities $\Gamma_{k=0,n}$ weakly depend on variation of α the impact of R_k on the flow is assumed to be small, too. This behavior is expected from theory and might stem from the fact that the truncation correctly picks the main contribution when flowing towards the full quantum theory.”

No doubt that this approach lacks a solid argument with respect to optimization, but it is the way to go in our situation where $P_k(q)$ and $U_k(\rho)$ are given maximally flexibility by means of *living* on a flexible grid.

In the case of bosons it is expected that $P_k^{-1}(q)$ includes an infrared divergence at $q = 0$ —the classical, non-relativistic inverse propagator reads⁵²

$$P_{\varphi,\Lambda}(q) = iq_0 + \frac{\mathbf{q}^2}{2} . \quad (3.50)$$

⁵²Please consult [appendix B](#) for our choice of units in use here.

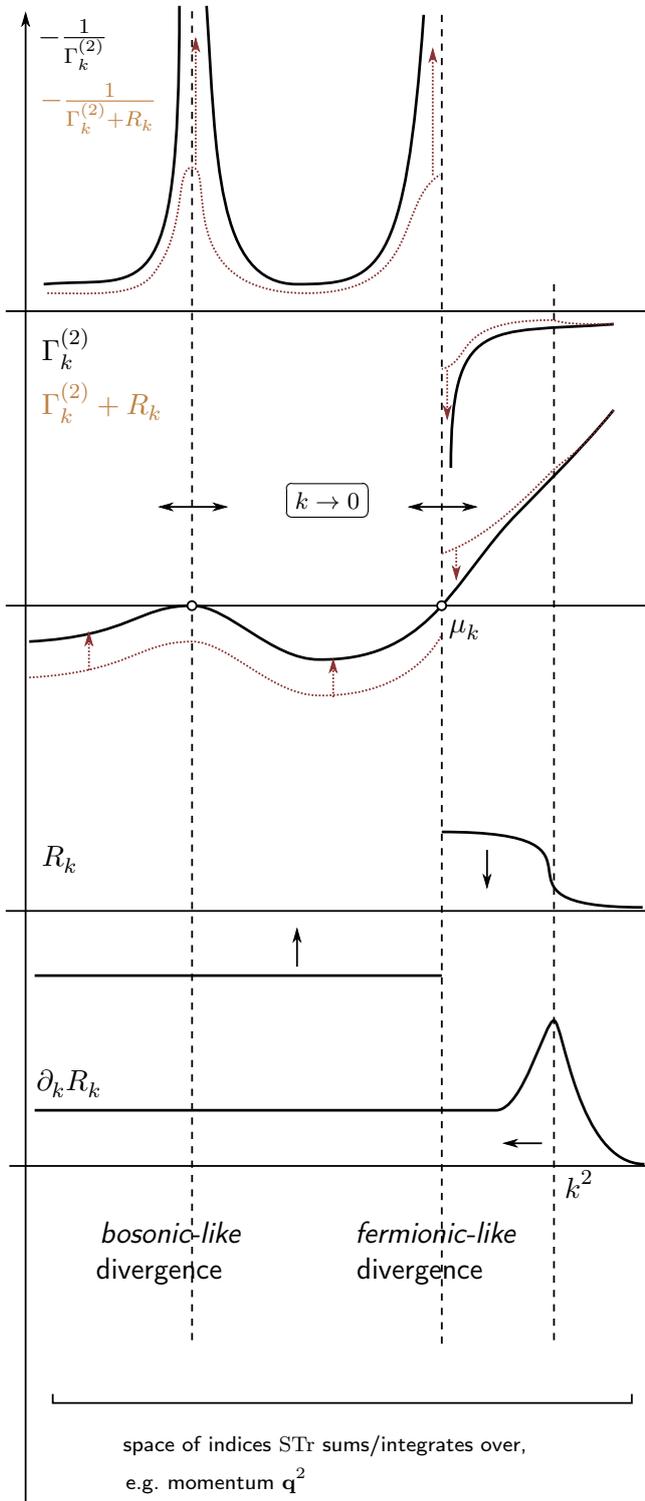


Figure 3.10: *Technical aspects of regulators.* This sketch intends to depict the (twofold) significance of R_k on the technical level. On the one hand, it softens/removes poles of the propagator G_k and on the other hand it cuts off the summation/integration from STr. As we argue [below](#), restricting the frequency/momentum summation/integration “ \sum_q ” up to scale k permits Γ_k to be interpreted as representing the physics at length scale k^{-1} , roughly speaking. However, to some extent this point of view is limited: Enforcing $\sum_{q \leq k}$ eventually misses poles of G_k when $k \rightarrow 0$. Since zeros of $\Gamma_k^{(2)}$ might dominate the contributions from STr to the β_k -function, the course of the flow quantitatively deviates from a situation where the corresponding pole(s) are included. From this perspective a regulator that does not *weight* the modes of the theory (cf. [eq. \(3.47\)](#)) seems to be more neutral compared to a choice like e.g. [eq. \(3.47\)](#).

The lesson to learn: It is a wise idea to properly keep track of G_k 's poles when flowing down to $k = 0$. The term *physics at scale k* should be associated with reference points defined by the pole structure of the propagator. In particular, a (flowing) Fermi surface μ_k needs to be regularized. The *relevant* momenta become $q \pm k$. Moreover, the exact position of μ_k is necessary to remove the related divergence. A simple *shift* of $\Gamma_k^{(2)}$ by R_k through $\Gamma_k^{(2)} + R_k$ that regulates the *infrared* divergence of e.g. bosonic propagators fails—it merely shifts the position μ_k . Nevertheless, the option to utilize an *imaginary* regulator remains, cf. [eq. \(3.53\)](#).

$P_{\varphi,k}(q \neq 0)$ should not develop zeros due to the absence of any *Fermi surface* for integer spin particles. Hence, adding a positive (real) number k^2 to the bosonic (inverse) propagator removes the *infrared* divergence of $P_{\varphi,k}^{-1}(q)$ at $q = 0$. But of course, for fermions we are faced with poles of the propagator at non-vanishing momentum,

$$P_{\psi,\Lambda}(q) = iq_0 + \mathbf{q}^2 - \mu \quad . \quad (3.51)$$

Since the flow demands $k^2 \rightarrow 0$, adding this term helps curing the divergence at the *Fermi energy* $\mu = \mathbf{q}_F^2$ as long as $k^2 > \mu$, only—if the renormalized (inverse) propagator does not get modified at all! Would we have been able to parametrize $P_k(q)$ such that zeros of P_k are explicitly given by analytic expressions—i.e. there is an explicit formula that identifies the Fermi surface which we associate/define by the set of q_F where P_k vanishes—we could explicitly adapt $R_k(q)$. However, this option is not at our disposal. Numerically tracking the zero of P_k entails an implicit dependence of R_k on k . Hence \dot{R}_k needs to be evaluated live during the flow introducing a source for numerical instability. But that is not the be-all and end-all to the story: According to fig. 1.2, R_k is actually allowed to be considered complex valued.

If we decompose a general complex valued regulator into its real and imaginary contribution as

$$R_k(q) \equiv R_{k,1}(q) + iR_{k,2}(q) \quad , \quad (3.52)$$

$R_{k,i}(q)$ needs to be symmetric ($i = 1$, *real* regulator) and antisymmetric ($i = 2$, *imaginary* regulator) in q , respectively; a requirement originating from the space-time translationally invariant, non-relativistic symmetry to be respected by P_k in order to stay with a physically reasonable interpretation of $\Gamma_{k \neq 0}$. In particular, at $P_{\psi}(q_0 = 0, \mathbf{q}) = 0$, the inverse propagator P_{ψ} is real valued. Therefore adding ik^2 properly regulates its divergence. Owing to this insight our approach to regulate fermions consists of

$$R_{\psi,k,1}(q) = 0 \quad \text{and} \quad R_{\psi,k,2}(q) = k^2 r_{\psi}(y) \quad (3.53)$$

with a shape function r_{ψ} to be discussed [below](#). The basic idea behind this choice is close to [BBW04]. It relates to the fact that fermionic Matsubara frequencies obey $q_0 \neq 0$ when $T \neq 0$. To this end temperature regulates terms as iq_0 . For the sake of completeness let us mention that there is another regularization scheme related to temperature. It is commonly referred to as *interaction regularization* or *temperature cutoff* [HS01b]. However, the anti-symmetry $\Im \mathbf{m} P_{\psi}(-q_0, \mathbf{q}^2) = -\Im \mathbf{m} P_{\psi}(q_0, \mathbf{q}^2)$ imposes that, in general, $\Im \mathbf{m} P_{\psi}(q_0, \mathbf{q}^2) \sim c(\mathbf{q}) q_0^{2n+1} \xrightarrow{q_0 \rightarrow 0} 0$ with some $n \in \mathbb{N}$ as long as $\Im \mathbf{m} P_{\psi}$ is allowed to be represented by a Taylor expansion around $q_0 = 0$. To this end, regulating the Fermi surface by an imaginary contribution R_k to the inverse propagator P_k is, to some extent, a reasonable idea.

Let us return to the situation where P_k vanishes at some q_F ; for the sake of illustration let us assume $q_F = (q_0 = 0, \mathbf{q}_F^2)$. In contrast to the bosonic case discussed a minute ago, this time we are faced with a *zero crossing* since q_0 takes positive and negative values as well. Hence it is not sufficient to just add a positive constant ik^2 which simply *shifts the divergence away* from $q_0 = 0$. We need to *push* $\Im \mathbf{m} P_{\psi}(q)$ for $q_0 > 0$ and $q_0 < 0$ into opposite directions. In addition we should establish the anti-symmetry

property for the regulator term: $i \operatorname{sign}(q_0)k^2$ will do the job⁵³. To complete our line of reasoning: Equation (3.53) for bosons discussed a moment ago reads

$$R_{\varphi,k,1}(q) = k^2 r_{\varphi}(y) \quad \text{and} \quad R_{\varphi,k,2}(q) = 0 \quad . \quad (3.54)$$

It remains to direct the focus on suitable one-dimensional regulator shape functions $r(y)$ and motivate their (straightforward) multi-dimensional generalization as well. After having discussed/specified the role of R_k in eq. (3.49) as removing the propagator's poles, we would like to spend few words on limiting STr through \dot{R}_k . Some physical interpretation is in order here: When switching from a single degree of freedom to a field (theory) with infinitely many degrees of freedom labeled by a (continuous) spatial index/variable x , we employ the Fourier transformation to end up with the *conjugate variable*/index q specifying the field's *modes*. Quantum field theory is designed to represent (classical) identical particles at momentum \mathbf{q} and energy q_0 as ((non-linear) interacting) modes of a given field $\varphi(x)$ or $\psi(x)$. Functional renormalization is constructed such that the regulator actually suppresses the propagation of certain modes q . Check cf. fig. 1.2 for the concept of suppressing a single degree of freedom.

Functional renormalization intends to relate the classical action $S_{k=\Lambda}$ at some *microscopic* momentum scale Λ to its full quantum counterpart $\Gamma_{k=0}$. The desired goal is reached by modifying the propagator $\Gamma_k^{(2)}$ by adding the regulator R_k to become a *modified* propagator. Regarding $\Gamma_k^{(2)} + R_k$ as the propagator of a given theory at scale k from the outset, the former technical trick which inserted R_k converts to a physical picture. It associates Γ_k to *some* theory at length scale k^{-1} . *Some* refers to the fact that the course of the flow $S_{\Lambda} \rightarrow \Gamma$ may strongly depend on the choice of the regulator, although the classical and quantum action remain the same in theory⁵⁴.

Let us clarify the notion of *some*: We could think of S_{Λ} as being an effective theory derived from a *more fundamental* one at length scales $k^{-1} > \Lambda^{-1}$. All modes q larger in magnitude than k (symbolic notation, see footnote 50) are already included by (flowing) parameters of S_{Λ} , i.e. $S_{\Lambda} = \Gamma_{k=\Lambda}$. Therefore we just need the regulator term $\frac{k^2}{2} \int_{q \leq \Lambda} \eta^{\dagger}(q) \eta(q)$, i.e. we differently weight the modes by k^2 . This defines the interpretation of $\Gamma_{0 \leq k \leq \Lambda}$. Mathematically it is given by the shape functions $r_k(y)$. A shape function whose corresponding Γ_k includes all modes with $q > k$ and none of $k < q$ needs to vanish and returns infinity, respectively. The associated k -derivative is proportional to $\delta(q - k)$. Hence it picks a single value of G_k under the integral STr that contributes to β_k , see eq. (3.49). In general we write

$$R_k(q) = k^2 r_k(y) \quad \text{and} \quad \partial_k R_k(q) = 2k[r_k(y) - y \cdot \nabla r_k(y)] \equiv 2k u_k(y) \quad (3.55)$$

with the definition of the *derivative* of the shape function $u_k(y)$. y is a n -dimensional variable $y = (y_1, \dots, y_n)$ and $y \cdot \nabla \equiv \sum_i y_i \partial_i$ where $\partial_i \equiv \partial/\partial y_i$. A moderate variant of $r_k(y)$ introduced a minute ago—we call it *sharp regulator*—becomes unity for $q < k$ instead of growing beyond any finite

⁵³The sign function $\operatorname{sgn}(x)$ is defined to return 1 for positive x and -1 otherwise—except for $x = 0$ where it vanishes identically. Note, that this choice does not remove the divergence at $q_0 = 0$ itself. However, $P_{\psi} + R_{\psi}$ is evaluated under the integral \int_q at zero temperature where altering a single value of the integrand does not modify the integral itself.

⁵⁴Practically we face a bunch of challenging technical barriers to overcome.

value. Nevertheless this washes out the former interpretation: During the course of the flow, modes with $q < k$ are *partially* included into Γ_k .

Various $r_k(y)$ might be constructed. For us there is a technical issue that motivates the definition of a (new) class of (bosonic) shape functions illustrated in [fig. 3.11](#) and referred to as *polynomial regulators*. Its implementation in `libfrg` is provided by the routines `relBosRegPoly()` (R_k) and `relBosRegPolyDer()` ($\partial_k R_k$). Both are part of the regulator collection in `frg_regs.cpp`. We define the one-dimensional (!) version of the polynomial regulator through

$$r_k^\alpha(y) \equiv (1-y)^\alpha \Theta(1-y) \quad \text{with} \quad u_k^\alpha(y) = [1 + (\alpha - 1)y] (1-y)^{\alpha-1} \Theta(1-y) \quad \text{where} \quad \alpha > 0 \quad . \quad (3.56)$$

The symbol $\Theta(x)$ refers to the unit step function that returns 1 for $x > 0$ and vanishes otherwise. y is assumed to be positive. In cases it becomes negative we substitute $y \rightarrow |y|$. Then, the ill-defined derivative at $y = 0$ is insignificant in so far as u_k is evaluated under an integral only. $\alpha \geq 0$ introduces a parameter which might be varied in $(1, \infty)$ (the reason is given in a moment) in order to estimate the sensitivity of $\Gamma_{k=0}$ on R_k . By means of the principle of minimum sensitivity one might select the best approximation of $\Gamma_{k=0}$ to the *real* effective action. Both, $r_k^\alpha(y)$ and $u_k^\alpha(y)$, are plotted for $\alpha \in [0, 6]$ in the right panels of [fig. 3.11](#).

[Equation \(3.56\)](#) originates from the fact that the sharp regulator as well as the so called *Litim regulator* (see [fig. 3.11](#))—which are actually the special cases $\alpha = 0, 1$ —develop discontinuities in β_k at finite temperature. The reason becomes obvious from the corresponding $u_k(y)$ that regulates the super-trace. Both exhibit at least a discontinuity at $y = 1$, i.e. $q = k$. For the sharp regulator there is a δ -peak on top in addition. At finite temperature q_0 adopts discrete values ω_n with equal spacing $\Delta q_0 = 2\pi T$. But k is still continuously decreasing during the course of the flow. Due to the discontinuity the summands of \sum_{ω_n} from STr do not get continuously weighted to zero when $k \rightarrow 0$. In contrast, when k passes a bosonic/fermionic q_0 its corresponding contribution to β_k becomes instantaneously dropped.

The technical reason behind the unwanted behavior of $u_k(y)$ at k is the explicit use of the unit step function $\Theta(1-y)$ that cuts off functions multiplied by—unless they do not converge to zero at $y = 1$ in a rather smooth manner such that its k -derivative continuously approaches zero for $y \rightarrow 1$ ($y < 1$). The benefit of using Θ is an explicit limitation of the integration/summation of STr . It is of significant importance to numerical evaluations. However, if we do not insist on this (numerically profitable) property there is a useful one-dimensional R_k to be found in the literature^[TW94,BTW00], called (generalized) *exponential regulator*. Its parametric dependence on κ is plotted in [fig. 3.12](#). The corresponding formulae read

$$r_k^\kappa(y) = \frac{z}{e^z - 1} \quad \text{and} \quad u_k^\kappa(y) = \frac{z}{(e^z - 1)^2} [(\kappa - 1) + (1 + \kappa(z - 1))e^z] \quad \text{where} \quad z \equiv y^\kappa \quad . \quad (3.57)$$

It is implemented in analogy to the polynomial regulator by the function `relBosRegExp()` and `relBosRegExpDer()` in `frg_regs.cpp`. Again, κ is a real and positive parameter to study the

⁵⁵The indefinite integral of $u_k^\alpha(y)$ reads $-(1-y)^\alpha [2 + (\alpha - 1)y] / (\alpha + 1)$ for $\alpha > 0$.

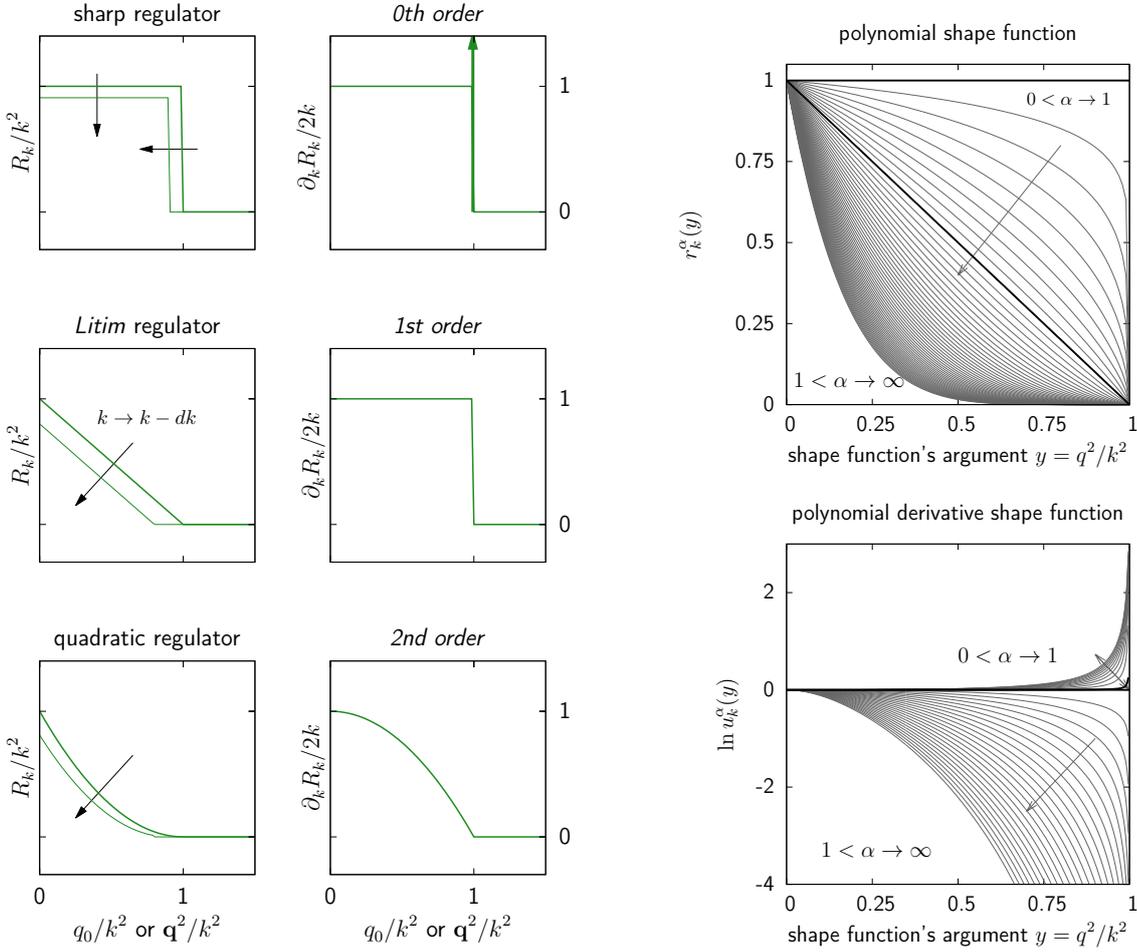


Figure 3.11: *The polynomial regulator.* The panel to the left reflects the idea behind the introduction of the polynomial regulator defined by eq. (3.56). When the flow parameter k is lowered towards zero the regularization scheme defined through R_k , or equivalently r_k , sets the limits of the STr-operation by virtue of $\partial_k R_k$, i.e. u_k . Unless the variation of r_k from k to $k - dk$ at k is not smooth enough (sharp and Litim regulator) there will be a discontinuity in $u_k(y)$ at $y = 1$ which potentially fosters numerical instability as we explain in the main text.

The two plots to the right scan the parameter $\alpha \geq 0$ of the polynomial regulator to demonstrate the dependence of the regulator shape function $r_k^\alpha(y)$ (upper panel) and its associated (logarithm of the) derivative $u_k^\alpha(y)$. $\alpha = 0$ corresponds to the *sharp regulator* $\Theta(1 - y)$ and the case $\alpha = 1$ is known as the *Litim regulator* (bold lines). Let us assume $\alpha > 0$. For $0 < \alpha < 1$, $u_k^\alpha(y)$ diverges when $y \rightarrow 1$. However, the integral $\int_0^\infty dy u_k^\alpha(y)$ stays finite⁵⁵ for all $\alpha > 0$, namely $2/(1 + \alpha)$. It approaches zero for $\alpha \rightarrow \infty$ and hence there is no use to choose α e.g. larger than, say, ~ 10 .

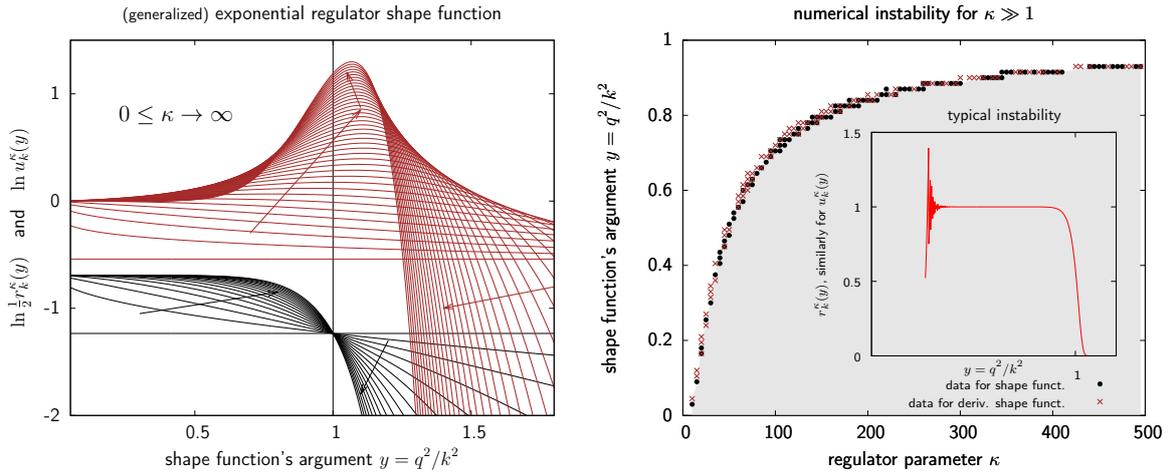


Figure 3.12: *The exponential regulator.* Left: Semi-logarithmic plots of eq. (3.57) for varying parameter κ , i.e. we present the shape functions of (0.5 times) the (generalized) exponential regulator $\frac{1}{2}r_k^\kappa$ (black, lower set of curves) and its derivative u_k^κ (brown), respectively. The arrows (\rightarrow) indicate the *direction of variation* when increasing κ . For $\kappa > 1$, u_k^κ develops a (local) maximum that shifts from $y = 0$ above $y = 1$ while becoming more pronounced at the same time. In the limit $\kappa \rightarrow \infty$ it converges to $y = 1$ from above. Both functions exponentially decay for $y \rightarrow \infty$. When $\kappa = 0$ the regulator's shape function becomes constant and for $\kappa \rightarrow \infty$ the sharp regulator $r_k^{\alpha=0}$ is recovered.

Although eq. (3.57) seems to be straightforward to be implemented, a numerical instability arises due to finite numerical precision which causes $\exp(y^\kappa) \mapsto 1$ for some finite $0 < y < 1$, especially when $\kappa \gg 1$. Hence the numerics is faced with division by zero. The dots in the figure to the right mark the κ -dependent y -values where the instability becomes serious. The inset demonstrates the emergence of such an instability which is accompanied by increasing oscillations around $r_k^\kappa(y) \approx 1$. Our implementation of the exponential regulator checks for the value of $e^y - 1$. If it falls below $1e^{-13}$, R_k and its derivative become substituted by their corresponding (theoretical) value at $y = q = 0$, namely k^2 and $2k$, respectively.

flow's sensitivity on the choice of the regulator. $\kappa = 0$ covers the trivial case $r_k = \text{const.}$ sometimes referred to as k^2 -regulator. $\kappa = 1$ denotes the *classical/standard* exponential regulator. In the limit $\kappa \rightarrow \infty$ eq. (3.57) turns into the sharp regulator $\Theta(1 - y)$. In general, the limiting cases $y \ll 1$ (such that $z = y^\kappa \ll 1$ in addition) and $y \gg 1$ (such that $z \gg 1$) yield the approximations $r_k^\kappa(y) \approx 1 - z/2 \approx 1$ and $r_k^\kappa(y) \approx ze^{-z}$, respectively. Actually, the idea behind the parameter κ might be phrased as follows: Take $z/(e^z - 1)$ with its limits written down a moment ago and assume $\kappa \gg 1$. Since $z = y^\kappa \ll y$ for $y < 1$ and $z \gg y$ when y exceeds unity, the validity of the approximations extends to a larger range when increasing κ . The interval around $y = 1$ where $r_k^\kappa(y)$ behaves differently shrinks. We finally end up with a constant plateau for $0 \leq y \lesssim 1$ as well as a rapid exponential decay for $y = 1 + \epsilon$ with $0 < \epsilon \ll 1$.

If we follow the same procedure for $u_k^\kappa(y)$ again, there is a subtle difference concerning the limits, i.e.

they explicitly depend on κ :

$$u_k^\kappa(y) \xrightarrow{y \rightarrow 0} \frac{1 + \frac{\kappa+1}{2}z + \dots}{1 + z + \dots} \quad \text{and} \quad u_k^\kappa(y) \approx \kappa z^2 e^{-z} \xrightarrow{z \rightarrow \infty} 0 \quad \text{for} \quad z \gg 1, \quad z \gg \kappa^{-1} = \text{const.} \quad . \quad (3.58)$$

In particular we note that if $\kappa > 1$ then $u_k^\kappa(y) > 1$ increases for small enough y and $u_k^\kappa(0) = 1$. Thus, there needs to be at least one local maximum in order to smoothly converge to zero for large y . It turns out⁵⁶ that the maximum emerging for $\kappa > 1$ drifts away from $y = 0$ exceeding $y = 1$. Finally, it converges to $y = 1$ from above.

We would like to add a comment on the implementation of the (generalized) exponential regulator: Namely, we encountered an instability related to division by (nearly) zero due to finite numerical precision. A detailed analysis provides the right panel of [fig. 3.12](#). There, we describe how to prevent this unwanted effect. Nevertheless, it is an issue for large κ , say $\gtrsim 20$, only. `libfrg` defines the (extendable) structure `ParamRegs` that contains attributes to specify the parameters for implemented regulator (shape) functions. The instance `paramRegs` is an attribute of the `flowQuant` class whose instance itself is part of the `Flow` class, see [fig. 3.1](#). In particular we define: `nDegPol` for α and `kappa` for κ in `ParamRegs`. In addition there is the technically relevant variable `maxX` which sets a y -value above which e.g. $r_k^\kappa(y) \stackrel{!}{=} 0$ is enforced (consult `relBosRegExp()` for details).

After having discussed several choices of one-dimensional regulators/shape functions, it is quite straightforward to extend those ideas to the case where y denotes a n -dimensional variable. There are two obvious approaches: a) one takes some *norm*, e.g.

$$|y| \equiv \sqrt{\sum_{i=1}^n y_i^2} \quad \text{with} \quad y_i = \frac{q_i}{k^2} \quad \rightarrow \quad R_k(q) = k^2 r_k(|y|) \quad \text{and} \quad \partial_k R_k(q) = 2k u_k(|y|) \quad (3.60)$$

and employs it as the argument to the regulator shape function or b) one directly multiplies the one-dimensional shape functions according to

$$R_k(q) = k^2 r_k^{(n)}(y) \quad \text{with} \quad r_k^{(n)}(y) \equiv \prod_{i=1}^n r_k(y_i) \quad . \quad (3.61)$$

The corresponding derivative shape function is obtained by explicit computation using [eq. \(3.55\)](#). For the polynomial regulator, we have in the case of two variables $y = (q_0, \mathbf{q}^2)/k^2 = (y_1, y_2)$

$$r_k^\alpha(y) = (1 - y_1)^\alpha (1 - y_2)^\alpha \Theta^{(2)}(1 - y) \quad \text{where} \quad \Theta^{(2)}(1 - y) \equiv \Theta(k^2 - |q_0|) \Theta(k^2 - \mathbf{q}^2) \quad (3.62)$$

⁵⁶If we explicitly solve $\partial_y u_k^\kappa(y) = 0$ we arrive at

$$0 = (1 - \kappa) + \left[2(\kappa - 1) + z(1 - 3\kappa) - \kappa z^2 \right] e^z + \left[(1 - \kappa) + z(3\kappa - 1) - \kappa z^2 \right] e^{2z} \quad ; \quad (3.59)$$

apart from solutions $y_0 = 0$. For $\kappa \gg 1$ there are two distinct cases: $y < 1$ ($z \ll 1$) and $y > 1$ ($z \gg 1$). The former situation turns [eq. \(3.59\)](#) into a trivial statement $0 \approx 0$, but the latter yields $0 \approx (1 - \kappa) + z(3\kappa - 1) - \kappa z^2$. An approximative solution reads $y_0^\kappa = z_0 \approx \frac{1}{2}(3 \pm \sqrt{5})$ finite. The minus sign from “ \pm ” is excluded due to $y > 1$. Finally, we have $y_0 \gtrsim 1 \rightarrow 1$ for $\kappa \rightarrow \infty$.

and

$$\begin{aligned}
 u_k^\alpha(y) &= [1 + (\alpha - 1)(y_1 + y_2) + (1 - 2\alpha)y_1y_2] [(1 - y_1)(1 - y_2)]^{\alpha-1} \Theta^{(2)}(1 - y) \\
 &= \left[1 + (\alpha - 1)(|q_0| + \mathbf{q}^2)/k^2 + (1 - 2\alpha)|q_0| \mathbf{q}^2/k^4 \right] \left[\left(1 - \frac{|q_0|}{k^2} \right) \left(1 - \frac{\mathbf{q}^2}{k^2} \right) \right]^{\alpha-1} \Theta^{(2)}(1 - y) \quad .
 \end{aligned}
 \tag{3.63}$$

As expected the derived result is invariant under $|q_0| \leftrightarrow \mathbf{q}$. Again, explicit restriction of the domain of integration/summation due to the explicit insertion of the Θ -step function proves convenient for numerical purposes.

software/library	written in	license	purpose	opt.
<i>GNU Scientific Library (GSL)</i>	C	GPL	scientific computing (integration, ODE solver, interpolation, ...)	
<i>GNU Multiple Precision Arithmetic Library (GMP)</i>	C/C++	LGPL ⁵⁷	arithmetic beyond the machine word precision	
<i>CUBA library</i>	C	LGPL	(Monte Carlo) integration routines	
<i>GNU Compiler Collection (OpenMP implementation)</i>	C++	GPL	cross-platform C/C++ compiler and multi-core parallel computing	
<i>Open MPI</i>	C	New BSD ⁵⁸	message passing interface for parallel computing	
<i>Integrated Performance Monitoring (IPM)</i>	C	LGPL	performance measure for parallel computation with <i>MPI</i>	X
<i>doxygen + Graphviz</i>	C++	EPL ⁵⁹	code documentation + code structure diagrams	X
<i>Git</i>	C, sh, Tcl, Perl	GPL	source code management	X
<i>Gnuplot</i>	C	own license ⁶⁰	2D and 3D data visualization	X

Table 3.3: Summary of licenses under which software included by and used in developing `libfrg` is distributed. The column labeled **opt**(ional) indicates if it is mandatory to the library.

3.4 A Short Note on Licensing

To close our discussion on numerical aspects of this thesis, we consider it worth adding few words on the conditions under which the library `libfrg` will be released. Although not a prior focus of scientific consideration, we carefully inspected this issue to end up selecting the *GNU General Public License* or *GNU*⁶¹ *GPL* for short—or even shorter: *GPL*. It is published by the *Free Software Foundation*

⁵⁷In a nutshell: This license is an extension of the GPL that allows for integration of LGPL licensed software into proprietary products under the obligation that the integrated parts stay free/LGPL licensed.

⁵⁸The *Berkeley Software Distribution* (a *Unix*-like operating from the *UC Berkeley*) licenses are compatible with the GPL.

⁵⁹The *Eclipse Public License* is a free, but GPL-incompatible license. It contains limited copyleft features.

⁶⁰Roughly: Open source license with source code modification granted. Details provides <http://gnuplot.info> as of Aug 2013.

⁶¹*GNU* is a recursive acronym for *GNU is Not Unix* naming a project that intends to provide a free version of the *Unix* operating system. It was initiated by physicist Richard Stallman who graduated from *MIT* in 1974. Well known and widely used software from *GNU* includes e.g. the *GNU Compiler Collection (GCC)* and the *GNU Emacs* editor. *GNU* intends to provide a free version of the *Unix* operating system and was subsequently extended by the Finnish computer scientist Linus Torvalds who wrote the *Linux kernel* and is also known for the distributed version control system *Git*. Today one refers to both efforts as *GNU/Linux*. A reasonable reference to explore the history of the *free software*

(FSF) and we encourage the reader to fly over its contents e.g. during a coffee break—in particular the preamble condenses the motivation and purpose of this license. As of Aug, 2013 the text is printed online at <http://www.gnu.org/licenses/gpl-3.0.html>.

The pragmatic reason for choosing the *GPL* is the very fact that `libfrg` includes software licensed under the *GPL* as well, see [table 3.3](#). Its terms simply enforce derived work to be published under the same terms (see paragraph 5 of the *GPL*). This property follows from a mechanism called *copyleft*. It aims at preserving the freedom to use, distribute and modify a work based on legal copyright law⁶².

In turn this entails the main purpose we had in mind when selecting this license—in fact the following drove our effort to rely on free software only: The user of `libfrg` have to release their code when based on the library under the *GPL* as well. It is our aim to ensure that everybody is allowed to forward research in the field of functional renormalization. The community is encouraged to further improve, extend and (re)share the code to foster scientific progress. It is public domain and should stay as such forever. As a side effect, publications in scientific journals become much more transparent with source code available.

Last but not least the *GPL* protects the developers from any warranty claims within applicable law, i.e. contributors to `libfrg` do not need to *fear* recourse due to non-reliable action when executing code compiled from the `libfrg`'s source. The exact legal text is written in paragraphs 15.–17. of the *GPL*.

movement provides [Moo01].

⁶²Another prominent example for such sort of licenses tailored to share creative works is the set of *Creative Commons Licenses* (CC). A similar, but weaker framework is set up by the *GNU Lesser General Public License* (LGPL). It allows for inclusion of the software into proprietary products under the constraint that the LGPL licensed code remains LGPL licensed.

First Application to Statistical Physics

We are finally going to utilize the concept of functional renormalization which has been introduced in [chapter 1](#) and received theoretical justification in [chapter 2](#). The main purpose of the following analysis is to provide a first benchmark for our numerical library from [chapter 3](#). Therefore we reduce the theoretical effort to a physical setup that renders bosons with relativistic dispersion in [section 4.1](#). It exhibits $O(N = 2)$ symmetry and corresponding models have been extensively studied for three spatial dimensions ($d=3$). On the subject there exists literature by means of Monte Carlo methods ^[GH93,BFMMM96], perturbative schemes like the ϵ -expansion ^[ZJO02], high-temperature expansion ^[BC97], and non-perturbative treatments as the functional renormalization ^[TW94,Lit02]. Access to critical exponents through experiments with superfluid helium is reported in [[SCL92,GA92](#)].

In [section 4.2](#) we intend to study the critical physics of the superfluid phase transition which falls into the $O(2)$ universality class. Instead of challenging the most advanced results for critical exponents we start with the most basic numerical setting to test `libfrg`. It opens the scene for a detailed investigation employing the full power of the numerics developed so far.

Finally, we switch to the dimensionless formulation of the flow equation for the $O(2)$ model in [section 4.3](#). Here we demonstrate the quantitative accuracy of the numerics by comparing the values of critical exponents derived from the *stability matrix* ^[BTW00,Str94] and the approach outlined in [section 4.2](#), respectively. Moreover we investigate the behavior of these values for the $O(N)$ symmetry in the limit $N \rightarrow \infty$.

4.1	Flow Equations for a Relativistic Gas of Interacting Bosons	123
4.2	Critical Physics of Superfluidity	129
4.2.1	Technically Benchmarking the Flow	130
4.2.2	Detecting Fixed Points & Recovering Mean Field Theory	134
4.3	The Story Revised: Scaling Form of the Flow Equations	140

4.1 Flow Equations for a Relativistic Gas of Interacting Bosons

In order to benchmark the numerical library `libfrg` we considerably reduce our theoretical setup from [chapter 2](#) and [appendix D](#). More precisely, we fall back to the much simpler physicals of an $O(N)$ -symmetric scalar model^[Sta68] with $N = 2$. It translates to bosons with relativistic dispersion in terms of the language promoted at the beginning of [section 2.1.3](#). In fact, the complex variable $\varphi(x)$ might be decomposed into a real and imaginary component $\varphi_{1/2}(x)$ by means of *locally* applying [eq. \(1.60\)](#) for each x . A simultaneous, *global* rotation of the two $\varphi_{1/2}(x)$ such that $\rho(x) = \frac{1}{2} [\varphi_1^2(x) + \varphi_2^2(x)] = \varphi^*(x)\varphi(x)$ remains constant is given by the $O(N = 2)$ transformation

$$\begin{pmatrix} \varphi_1(x) \\ \varphi_2(x) \end{pmatrix} \rightarrow \begin{pmatrix} \cos \alpha & -\sin \alpha \\ \sin \alpha & \cos \alpha \end{pmatrix} \begin{pmatrix} \varphi_1(x) \\ \varphi_2(x) \end{pmatrix} \quad \text{and} \quad \begin{pmatrix} \varphi(x) \\ \varphi^*(x) \end{pmatrix} \rightarrow \begin{pmatrix} e^\alpha & 0 \\ 0 & e^{-\alpha} \end{pmatrix} \begin{pmatrix} \varphi(x) \\ \varphi^*(x) \end{pmatrix}, \quad (4.1)$$

respectively. Clearly, any function of $\rho(x)$ stays invariant as well. Hence, this symmetry constraint restricts the most general local¹ contribution entering our truncation to $\int_x U_k(\rho(x))$. Concerning derivatives of the bosonic field, each $\partial_x \varphi(x)$ needs to be accompanied/multiplied by a term that compensates for the transformation factor² e^α under transformation [eq. \(4.1\)](#).

More symmetry is added by insisting on Lorentz symmetry. When switching from Minkowski space of special relativity to Euclidean space by means of imaginary time (cf. [footnote 14](#) of [chapter 2](#)) the invariance of the *wave operator* $\partial^2 = \partial_t^2 - \partial_{\mathbf{x}}^2$ turns out to require that field derivatives can be rewritten in terms of the *Laplace operator*

$$\partial_x^2 \equiv \partial_\tau^2 + \partial_{\mathbf{x}}^2 = \partial_\tau^2 + \partial_{x_1}^2 + \dots + \partial_{x_d}^2, \quad (4.2)$$

only. Therefore, the most general term for a truncation of the effective action that captures (semi-)local contributions might be written as

$$\int_x \mathcal{L}(\rho(x), \partial_x^2) \quad . \quad (4.3)$$

It has *internal* $O(2)$ symmetry³ and the corresponding theory exhibits real-time dynamics which obeys the space-time symmetry of special relativity.

However, we do not necessarily have to consult the correspondence between quantum dynamics and quantum statistical physics through $\tau \leftrightarrow it$. There exists another mapping which relates d -dimensional quantum statistical physics to $D = (d + 1)$ -dimensional classical statistical physics. Roughly speaking, the idea goes back to the construction of the field integral from the quantum partition function briefly sketched in [section 1.1](#). Instead of interpreting the field integral, [eq. \(1.16\)](#), with action, [eq. \(1.17\)](#), as originating from the reformulating the quantum partition function, [eq. \(1.12\)](#), with degrees of freedom η_i at (quantum) temperature $T = \beta^{-1}$, one might regard it as defining a classical partition function which sums over all degrees of freedom $\eta_i(\tau)$, including τ . Then, T determines the geometry of the

¹Non-local terms as e.g. $\varphi^*(x)\varphi(y)$ with $x \neq y$ are perfectly allowed, but we drop them here.

²Since α does not depend on x we have $\partial_x \varphi(x) \rightarrow e^\alpha \partial_x \varphi(x)$.

³The extension to $O(N)$ is straightforward by defining $\rho(x)$ proportional to $\sum_{i=1}^N \varphi_i^2(x)$.

D -dimensional space where the classical theory with Hamiltonian $S[\eta]/T_{\text{cl}}$ is defined on. We introduced the *classical* temperature T_{cl} , which is absorbed into $S[\eta]$ or simply set to unity—there is no direct access to it, but as we will see in [section 4.2](#), especially by [fig. 4.8](#), there is a way to indirectly recover T_{cl} for critical physics. Concerning the geometry of the D -dimensional space: For $T \neq 0$ it is represented by $S_\beta \times \mathbb{R}^d$ where S_β is a circle of circumference β . In the limit $T \rightarrow 0$ one should expect to converge to the corresponding physics in \mathbb{R}^D . A more careful as well as pedagogical treatment is presented in e.g. [\[Sac11\]](#) which also covers $O(2)$ symmetric models.

Transferring the previous thoughts to our $O(2)$ -symmetric relativistic physics we might interpret the case $\beta \rightarrow \infty$ as being related to a classical system in D dimensions. Since [eq. \(4.3\)](#) is still way too complicated being practically tractable, let us further reduce to a setting `libfrg` is capable to manage at the moment, namely

$$\Gamma_k[\varphi, \varphi^*] \stackrel{!}{=} \int_x U_k(\rho(x)) + \int_x \varphi^*(x) P_k(\partial_x^2) \varphi(x) \stackrel{4}{=} \int_x U_k(\rho(x)) + \sum_{a=1,2} \int_x \varphi_a(x) P_k(\partial_x^2) \varphi_a(x) \quad . \quad (4.4)$$

In fact, dropping the fermionic degrees of freedom from our initial ansatz of the effective action we are left with bosons. We further simplify [eq. \(2.67\)](#) to end up with [eq. \(4.4\)](#) by selecting the translationally invariant (inverse) propagator $P_{\varphi,k}(y-x) = \delta(y-x) P_k(\partial_x^2)$.

If we interpret $\varphi_{1/2}(x)$ as components of local two-dimensional classical spins

$$\vec{\varphi}(x) \equiv \begin{pmatrix} \varphi_1(x) \\ \varphi_2(x) \end{pmatrix} \quad , \quad (4.5)$$

we obtain physics which should help to understand the long-distance behavior of the so called (classical) *XY-model*^[ZJO02,AS10,Sac11] in D spatial dimensions. Especially when this system is tuned to a second order phase transition, we expect that the divergence of the *correlation length* ξ suppresses details of the microscopic model. Starting the flow with $P_{k=\Lambda}(\partial_x^2)$ proportional to ∂_x^2 , the *kinetic* term assumes a form that treats inhomogenities in $\vec{\varphi}(x)$ as energetically unfavorable: With the aid of partial integration we rewrite

$$\int_x \vec{\varphi}(x) \cdot \partial_x^2 \vec{\varphi}(x) \quad \text{as being proportional to} \quad \sum_{\substack{a=1,2 \\ i=\tau, x_1, \dots, x_d}} \int_x [\partial_i \varphi_a(x)]^2 \quad (4.6)$$

which exclusively vanishes for $\vec{\varphi}(x) = \text{const.}$. Another way of phrasing this: Fourier transforming ∂_x^2 yields a term proportional to $q^2 = q_0^2 + \mathbf{q}^2$, i.e. modes of the bosonic field with $q \neq 0$ add to the value of the inverse propagator and thus suppress/*damp* propagation.

Let us further restrict our ansatz of the effective potential U_k to be a polynomial in ρ up to quadratic order—we simply have a φ^4 -theory^[PS95]. More precisely: Taking the flow of U_k that follows from $\partial_k \Gamma_k|_{\eta=\Phi}$ we assume the ansatz

$$U_k(\rho = \varphi^* \varphi) = -p_k + m_k^2 (\rho - \rho_{0k}) + \frac{\lambda_k}{2} (\rho - \rho_{0k})^2 \quad (4.7)$$

⁴As a side remark: In the more general (non-relativistic) case of $P_k(\partial_x)$ there is *mixing* between the $\varphi_{1/2}$ -components. The technical reason being that odd powers of ∂_x in terms $\varphi_{1/2}(x) \partial_x^{2n+1} \varphi_{2/1}(x)$ ($n \in \mathbb{N}$) acquire an additional minus sign when applying partial integration under \int_x . Thus, *cancellation* of mixing terms in $\varphi^* P \varphi$ is prevented.

and ρ_{0k} is engineered to determine the minimum of U_k . Equation (4.7) is regarded as an expansion around the *physical*⁵ mean ρ_{0k} :

- $p_k \equiv -U_k(\rho_{0k})$ (pressure),
- $m_k^2 \equiv U'_k(\rho_{0k})$ (mass parameter⁶) and
- $\lambda_k \equiv U''_k(\rho_{0k})$ (interaction parameter) as well as
- ρ_{0k} (condensate density)

are the flowing parameters characterizing the effective potential. This truncation obviously demands $U_k^{(n>2)} \stackrel{!}{=} 0$ which—from the computational point of view—enormously reduces the rhs. of the flow equations. Moreover, being just quadratic in ρ allows U_k for a single minimum only. Hence the truncation will not be able to capture first order phase transitions. Here, the focus is on second order phase transitions.

As discussed in section 2.3 one might distinguish the cases $\rho_0 = 0$ (SYM) and $\rho_0 \neq 0$ (SSB). Due to the definition of ρ_{0k} as representing the minimum of U_k , m_k^2 needs to vanish: Explicitly computing ρ_{\min} from eq. (4.7) by demanding $U'_k(\rho_{\min}) = 0$ for $\rho_{0k} \neq 0$ we get $\rho_{\min} = \rho_{0k} - m_k^2/\lambda_k$ (w.l.o.g. $\lambda_k > 0$). Hence

$$m_k^2 = U'_k(\rho_{0k} \neq 0) \stackrel{!}{=} 0 \quad (4.8)$$

by construction within the SSB phase. However, for the SYM phase it is necessary to have $m_k^2 > 0$ in order for the minimum to exist⁷ at $\rho = 0$. On the other hand there is obviously no flow of ρ_{0k} by definition of this phase. Hence we reduced the infinite set of parameters from *theory space* of Γ_k to *five* flowing quantities⁸: $P_k(q^2)$, p_k , m_k^2 , λ_k , and ρ_{0k} where, on vanishing of the condensate density, $m_k^2 \geq 0$ triggers the transition from the SYM to the SSB phase and $\rho_{0k} \geq 0$ does so for the reverse case. The approach is sketched in fig. 4.1 in order illustrate the following:

Starting with $m_{k=\Lambda}^2 > 0$ the condensate density will vanish during the flow in the case of sponaneous symmetry breaking. Exactly at $m_k^2 = 0$ one needs the flow equation of the minimum of the effective

⁵Recall from section 1.2 (here we use the notation from that section!): The equation of motion for the effective action without external sources $\frac{\delta}{\delta\phi}\Gamma_k[\phi] = 0$ determines the expectation value ϕ of the microscopic fields φ . The term *physical* is a bit sloppy. At scale k the microscopic action reads $S_k[\varphi] = S[\varphi] + k^2 \int_q r(q/k)\varphi^*(q)\varphi(q)$ where S represents the physical system of interest and S_k determines the physics of that system modified by a cutoff $R_k(q) = k^2 r(q/k)$. Typically the cutoff is constructed such that it mainly suppresses the propagation of low momentum modes by an additional *mass* k^2 .

⁶The notation m_k^2 is pure convention from relativistic physics. In particular the square does not need to be positive from the formal point of view. As a coefficient of a Taylor expansion it might turn negative. However, as to become clear in a minute, $m_k^2 \geq 0$ by construction.

⁷For ρ sufficiently small the linear term of U_k (besides its constant offset, minus the pressure p_k) dominates. Since $\rho = \varphi^* \varphi \geq 0$, the *slope* $U'_k(\rho = 0)$ needs to be positive for $U_k(\rho = 0)$ to constitute a minimum. As being easily checked by taking eq. (4.7), U'_k is positive for $\rho < m_k^2/|\lambda_k|$ whenever $m_k^2 > 0$ —irrespective of the sign of λ_k . Of course, $\lambda_k < 0$ is physically instable by means of $\lim_{\rho \rightarrow \infty} U_k \rightarrow -\infty$.

⁸Of course, the *grid resolution* of $q = \{q_i\}$ on which the inverse propagator is (numerically) evaluated in the end specifies the final number of flowing parameters. Note, that the Euclidean version of *Lorentz covariance*, i.e. $P_k = P_k(\partial_x^2)$, allows for a one-dimensional grid since the corresponding P_k in momentum space respecting this symmetry should depend on q^2 only.

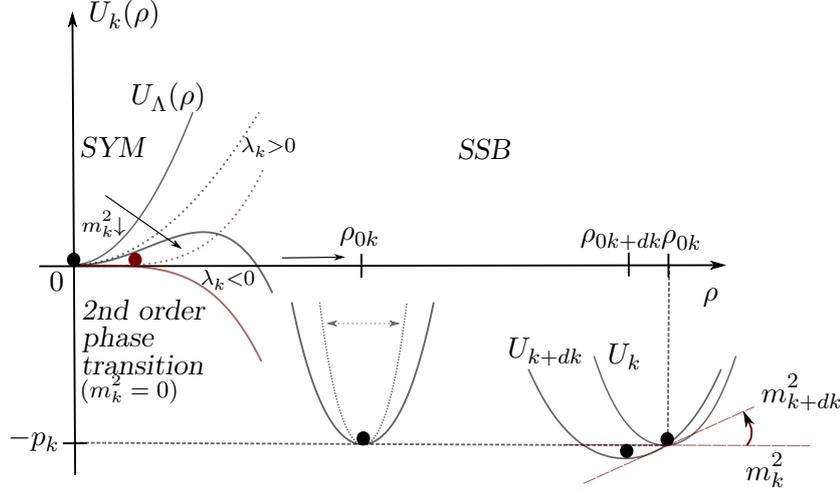


Figure 4.1: *SSB for polynomial truncation/Taylor expansion of the effective potential.* To follow the flow of *physical* quantities it is convenient to dynamically expand the effective potential U_k around the condensate ρ_{0k} . This implies explicitly *switching* the flowing parameters from the SYM to the SSB phase and the other way around. More precisely, the flow of the *mass parameter* m_k^2 is substituted by the flow of the condensate ρ_{0k} for SYM→SSB. The right part of the plot demonstrates the mechanism that shifts ρ_{0k} from k to $k + dk$.

potential, $\rho_0 = \rho_{0k}$, since eq. (4.7) *dynamically* expands around this particular value; it does not parametrize the global shape of U_k . Again, at this stage we explicitly see why the truncation from above is not able to capture first order phase transitions: We reside at the flowing (local) minimum that starts to (continuously) *move* away from $\rho = 0$. As the *derivative expansion* constitutes an expansion of the (inverse) propagator $P(q)$ around $q = 0$ focusing on the *low energy modes* of the problem, eq. (4.7) represents a Taylor expansion around ρ_0 of the microscopic fields. In order to keep track of ρ_{0k} we exploit its definition being a local minimum of U'_k for all k of the SSB phase: At each k we have $U'_{k+\Delta k}(\rho_{0k+\Delta k}) = U'_k(\rho_k) = 0$ where U'_k is modified by its explicit k dependence and the shift of $\rho_{0k+\Delta k} = \rho_{0k} + \Delta\rho_{0k}$ when $k \rightarrow k + \Delta k$. In particular, when Δk becomes infinitesimal we have

$$0 = \partial_k U'_k(\rho_{0k}) + U''_k(\rho_{0k}) \partial_k \rho_{0k} \quad \Rightarrow \quad \partial_k \rho_{0k} = -\frac{\partial_k m_k^2}{\lambda_k} . \quad (4.9)$$

At first sight this result might seem confusing since we have eq. (4.8), but this is exactly how it works: When the shape of U_k changes from k to $k + dk$ the zero slope m_k^2 at ρ_{0k} , in general, acquires a finite value which slightly shifts the potential's minimum to smaller ρ for increasing m_k^2 and to larger ρ for decreasing slope. The minus sign in eq. (4.9) explicitly accounts for this observation, cf. also fig. 4.1. Note that we assume a local minimum at ρ_{0k} in the sense that $\lambda_k = U''_k(\rho_{0k}) > 0$.

From this arguments one might also have guessed the result—up to a (positive) numerical prefactor—by dimensional analysis as follows: Taking our truncation, eq. (4.7), the *rate of change* $\dot{\rho}_{0k}$ should depend on \dot{m}_k^2 to first order, since the *shape* of a smooth U_k is approximated by the (linear) slope U'_k . However, the absolute *position* specified by the *coordinates* (ρ_{0k}, p_k) should not affect it. For a relativistic D -dimensional theory with momentum dimension $[q^\mu] \stackrel{!}{=} 1$ we have $[\rho_{0k}] = D - 2$, $[\partial_t] = 0$

and $[m_k^2] = 2$. In order to proceed we necessarily have to include the parameter λ_k that specifies the next to leading order contribution to U_k 's shape at ρ_{0k} . We have $[\lambda_k] = 4 - D$ and therefore we construct $[m_k^2/\lambda_k] = 2 - 4 + D = D - 2 = [\rho_{0k}]$, thus: $\dot{\rho}_{0k} \sim -\dot{m}_k^2/\lambda_k$. In essence we observe: To keep m_k^2 zero within the SSB phase its change has to be absorbed by a shift of the local minimum ρ_{0k} .

We have an expression for the flow of the (inverse) propagator $P(q^2)$ and the effective potential U_k from [appendix D](#). We also know how to determine the flow of the effective potential's minimum ρ_{0k} in the SSB phase from the parametrization, [eq. \(4.7\)](#). It remains to project the (general) flow of U_k to the pressure, the mass and the interaction parameter which is established by appropriate ρ -derivatives:

$$\partial_k p_k = -\partial_k U_k(\rho_{0k}) , \quad \partial_k m_k^2 = \partial_k U_k'(\rho_{0k}) \quad \text{and} \quad \partial_k \lambda_k = \partial_k U_k''(\rho_{0k}) \quad . \quad (4.10)$$

Adopting the notation from [appendix D](#), dropping the flow parameter index k and implicitly assuming to evaluate $U_k^{(n)}(\rho)$ ($n = 0, 2$, higher derivatives vanish due to our ansatz for U_k) at ρ_{0k} the final set of equations read⁹

$$\begin{array}{ll} \text{SSB } (\rho_0 \neq 0, m^2 \stackrel{!}{=} 0) & \text{SYM } (\rho_0 = 0, m^2 > 0) \\ \dot{p} = - \int \frac{\tilde{P}_1}{G_1} \dot{R}_1 & \rightarrow \quad - \int \frac{\dot{R}_1}{\tilde{P}_1} \end{array} \quad (4.11)$$

$$\dot{m}^2 = -2\lambda \int \frac{\tilde{P}_1^2 - \tilde{P}_1 \lambda \rho_0 + (\lambda \rho_0)^2}{G_1^2} \dot{R}_1 \quad \rightarrow \quad -2\lambda \int \frac{\dot{R}_1}{\tilde{P}_1^2} \quad (4.12)$$

$$\hookrightarrow \dot{\rho}_0 = -\dot{m}^2/\lambda \quad \rightarrow \quad 0 \quad (4.13)$$

$$\dot{\lambda} = 2\lambda^2 \int \frac{5\tilde{P}_1^3 - 12\tilde{P}_1^2(\lambda\rho_0) + 15\tilde{P}_1(\lambda\rho_0)^2 - 4(\lambda\rho_0)^3}{G_1^3} \dot{R}_1 \quad \rightarrow \quad 10\lambda^2 \int \frac{\dot{R}_1}{\tilde{P}_1^3} \quad (4.14)$$

$$\begin{aligned} \dot{P}(q^2) = & - \left(2\lambda \int \frac{\tilde{P}_1^2}{G_1^2} \dot{R}_1 \right) + \left(12\lambda \int \frac{\tilde{P}_1^2 \tilde{P}_2}{G_1^2 G_2} \dot{R}_1 \right) (\lambda\rho_0) \\ & - 2\lambda \left(\int \frac{\dot{R}_1}{G_1^2} + 4 \int \frac{\tilde{P}_1^2 + 2\tilde{P}_1 \tilde{P}_2}{G_1^2 G_2} \dot{R}_1 \right) (\lambda\rho_0)^2 \\ & + \left(4\lambda \int \frac{4\tilde{P}_1 + 3\tilde{P}_2}{G_1^2 G_2} \dot{R}_1 \right) (\lambda\rho_0)^3 - \left(8\lambda \int \frac{\dot{R}_1}{G_1^2 G_2} \right) (\lambda\rho_0)^4 \quad \rightarrow \quad -2\lambda \int \frac{\dot{R}_1}{\tilde{P}_1^2} \end{aligned} \quad (4.15)$$

$$\hookrightarrow P(q^2) = \tilde{P}(q^2) - \lambda\rho_0 - R(q^2) \quad \tilde{P}(q^2) = m^2 - R(q^2) \quad . \quad (4.16)$$

A crucial fact we exploited for this equations is $P_k(q^2) \in \mathbb{R}$. This is certainly true for the microscopic (inverse) propagator $P_{k=\Lambda}(q^2) = q^2$ and thus the flow will not introduce any imaginary contribution as

⁹As a reminder: $\tilde{P}_k(q') \Big|_{\rho=\rho_{0k}} \equiv U_1 + U_2 \rho_{0k} + P_k^{reg}(q'^2) \equiv m_k^2 + \lambda_k \rho_{0k} + P_k^{reg}(q'^2) \equiv m^2 + \lambda \rho_0 + P_1 \equiv \tilde{P}_1$, cf. [eq. \(2.165\)](#) and $G_1 \equiv \tilde{P}_1^2 - (\lambda\rho)^2$ ([eq. \(2.164\)](#)), we set $\varphi^* = \varphi = \sqrt{\rho}$. We adapted the convention to the additional definitions and conventions from this paragraph.

long as $\dot{R}_1 \in \mathbb{R}$. On the other hand we have the general property $P^*(q) = P(-q)$ from eq. (2.106). Taking $P_k = P_k(q^2)$ into account it necessarily follows that the inverse (relativistic) propagator has to be real. The choice of an imaginary regulator would be odd in this case. As a remark, we note that neither the flow of the (modified, inverse) propagator \tilde{P}_k , the condensate density ρ_{0k} nor the interaction parameter λ_k do depend on p_k . Since we are mainly focusing on the propagator's momentum dependence we might eventually neglect eq. (4.11) from this point of view.

4.2 Critical Physics of Superfluidity

To get our hands on a first application of the library `libfrg` which we developed as a general purpose framework for numerical studies with functional renormalization, we stick to the most simplest truncation for the physics introduced in the previous [section 4.1](#). To this end, we benchmark the system of bosons with relativistic dispersion in $d = 3$ spatial dimensions neglecting the flow of the (inverse) propagator P_k , [eq. \(4.15\)](#). For sure, this setup is superficially simple and we do not expect to end up with quantitative results at the cutting edge of research. Here, our intention is primarily to provide a *proof of concept* that motivates further, more physical and advanced truncations, which are straightforward to apply with the aid of the newly developed numerical library `libfrg`.

In order to approach the critical physics of the $O(2)$ model we need to recapture the idea of scaling which is expected to become relevant near and at a second order phase transition. The canonical approach introduced in [\[TW94\]](#) and followed by most of the succeeding publications motivates the reformulation of the flow equations by appropriately rescaling all flowing observables to dimensionless quantities. In order not to follow this procedure on a one-to-one basis we decide to stay with the dimensional [eqs. \(4.12\) to \(4.14\)](#) to cross-check if, at least at a qualitative level¹⁰, the outcome is acceptable—as it should.

Let us apply a bit of jargon to our specific setup. Since we are going to detect a *second order phase transition* there needs to be a corresponding *order parameter*. As we identified $\vec{\varphi}(x)$ with a local, classical spin, $\rho(x)$ is proportional to its magnitude. Since the derived flow equations are evaluated at constant $\rho(x) = \rho_0$, a finite condensate density $\rho_0 \neq 0$ indicates a phase transition in the sense that the ground state configuration of the $\vec{\varphi}(x)$ leads to a non-vanishing expectation value on averaging. Due to the fact, that $U_k(\rho)$ is approximated around ρ_0 at each scale k , we are confined to follow continuous variation of the condensate density, i.e. *first order phase transitions* are excluded. At the macroscopic level, $\rho_{0k=0} > 0$ indicates spontaneous symmetry breaking (SSB) and $\rho_{0k=0} = 0$ characterizes the system in the symmetric phase (SYM), respectively. At the moment when an *ordered state* $\rho_{0k=0} \neq 0$ builds up, the system is said to be *strongly correlated*, i.e. the orientation of a microscopic $\vec{\varphi}(x)$ is correlated with other $\vec{\varphi}(y)$ over distances $|x - y|$ characterized by the so called *correlation length* ξ which becomes divergent at the phase transition. Thus one might expect that microscopic parameters lose their relevance such that they drop from physical equations which are then governed by *universal* properties such as the dimension and (internal) symmetries of the system, only. Then the flow which computes quantities at scale k should be governed by the *canonical dimension* of these quantities itself. We refer to such an assumption as *scaling hypothesis*. *Anomalous dimensions* measure deviation from this expectation. They become introduced by rescaling the field φ in order to stay with a standard kinetic term $\frac{1}{2}q^2$. However, in our basic/naive approximation we neglect this effect. In $d = 3$ corrections to the scaling hypothesis are known to be small^{[\[TW94\]](#)}. While a system at a second order phase transition

¹⁰Clearly, since the scaling form of the flow equation is tailored to become independent from the explicit k -dependence, it is particularly adapted to extract critical exponents with quantitative precision. Due to our crude reduction of the truncation as well as sticking to the dimensional flow we do not expect to challenge quantitative results obtained by functional renormalization to date.

is expected to be dominated by the number of dimension and symmetries, one encounters quantities referred to as *critical exponents* to tackle the physics close to the second order phase transition.

As far as we have introduced the conceptual idea behind critical physics, we deduce that an observable O_k of canonical dimension $[O]$ should scale proportional to $k^{[O]}$ in cases the system becomes critical, i.e. if it encounters a second order phase transition. Hence, the dimensionless quantity

$$o_k \equiv O_k/k^{[O]} \quad (4.17)$$

should stay constant. In terms of flow dynamics^[Str94] this situation is referred to as *fixed point* $o_k = \text{const.} \equiv o^*$. Switching to the dimensionless derivative $\partial_t = k\partial_k$ of the flow parameter $t = \ln k/\Lambda$, the flow equation $\dot{O}_k = \beta_O$ with the β -function determined by the rhs. of eq. (2.1) is reformulated as

$$\dot{o}_k \equiv \partial_t o_k = \frac{\beta_O - [O]O_k}{k^{[O]}} \quad (4.18)$$

For the fixed point $\dot{o}_k = 0$, the β -function is equivalent to $[O]O_k$ and the flow of O_k is ruled by

$$\dot{O}_k = [O]O_k \quad \Rightarrow \quad O_k \sim \exp[O]t \sim k^{[O]} \quad (4.19)$$

4.2.1 Technically Benchmarking the Flow

Before turning our attention to the critical physics let us check some technical aspects of the numerical output of `libfrg`. As mentioned earlier we set the dimension of the classical system¹² to $d = 3$ where the anomalous dimension is expected to be small. The flow always starts at the *microscopic* scale $k = \Lambda$ and runs down to the *macroscopic* physics at¹³ $k = 0$, i.e. $t = 0 \rightarrow -\infty$. Our numerical analysis employs the exponential regulator, eq. (3.57), with $\kappa = 1$. Numerical data are represented by points and lines are purely drawn to guide the eye. In cases where we plot the line only, it refers to analytic functions. A detailed description/discussion of the figures is given by their caption.

First of all we use the option of `frgFlow2()` to execute an arbitrary user routine at the end of each evolution step from t to $t + \Delta t$ in order to print the integrand of the β -function of the interaction parameter λ_k at several intermediate steps. The result is shown in fig. 4.2 and it serves as a proof of concept for the successful interplay of the numerical components of `libfrg`. Next, we would like to check the convergence of the numerical flow depending on precision parameters. We are allowed to set them through the class `FlowParams` whose instance has to be supplied to the constructor `Flow::Flow()` in order to properly build an instance that comprises all necessary ingredients to

¹¹Although the scale derivative of the (one-dimensional) exponential regulator is peaked around $\mathbf{q}^2/k^2 = 1$ it does not vanish for $\mathbf{q}^2/k^2 < 1$.

¹²The (quantum) temperature is set to zero such that the Matsubara frequencies q_0 become continuous. Due to the Lorentz symmetry, q_0 is treated on equal footing as the spatial momenta \mathbf{q} . Therefore we technically reduce $P_k(q_0, \mathbf{q})$ to the one dimensional grid $P_k(0, \mathbf{q}^2)$, neglecting the Matsubara summation when evaluating the trace operation. As a consequence we only need the one-dimensional version of a regulator. Furthermore we substitute D by d . The corresponding routines of `libfrg`, namely `fullTrace()` and `spatInt()`, serve special options to do so. If `N0=0`, `fullTrace()` omits the Matsubara summation.

¹³Please note: To label flowing quantities we interchangeably use k and t , i.e. $O_t = O_k$.

¹⁴By construction of the code, the initial step size is set to `FlowParams::rkepsrel` which is $1/10$ in this case.

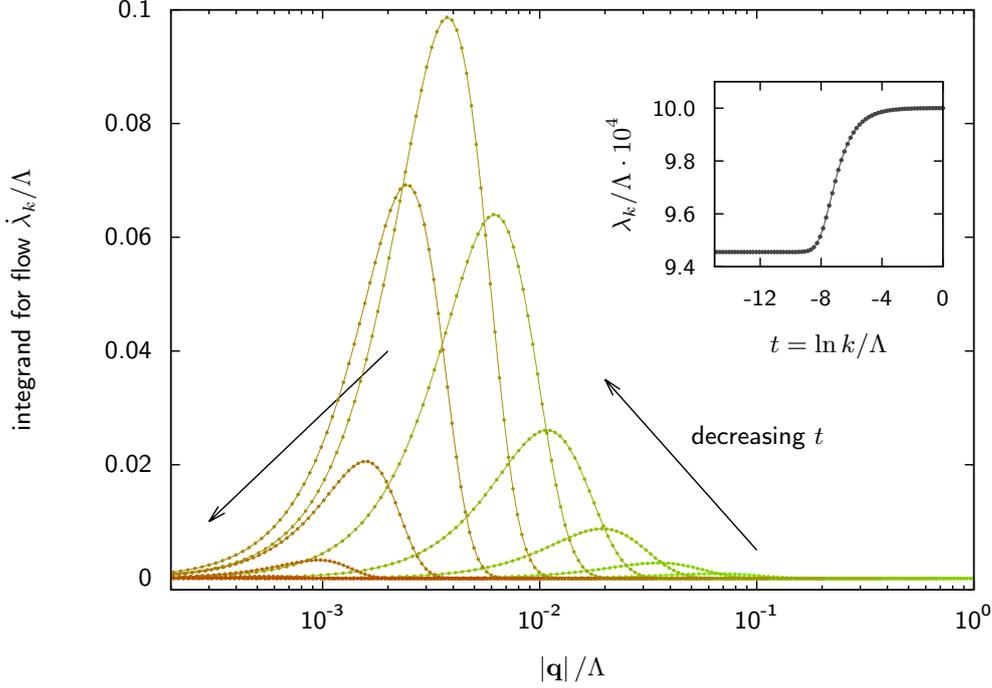


Figure 4.2: *Flow of the interaction parameter λ .* We set the system to start in SSB, i.e. $\rho_{0\Lambda} = 10^{-2}\Lambda$, but it enters instantaneously (after one step of iteration from $t = 0$ to $t < 0$) into the symmetric phase (SYM). According to eq. (4.14), $\beta_\lambda \sim +\lambda^2$, hence decreasing t leads to monotonically decreasing λ , which is confirmed by the numerical data presented by the inset.

The main panel illustrates the evolution of the integrand \dot{R}_1/\tilde{P}_1^3 to be integrated in order to obtain β_λ . From $t \approx 0$ (green) to $k \approx 0$ (red) its main contribution is peaked around $q \approx k$ which mainly arises from the interplay of two aspects: a) \dot{R}_1 (approximately) exponentially decays for $^{11}\mathbf{q}^2 > k^2$ and b) for $d = 3$ the integration \int comes along with an additional factor \mathbf{q}^2 . These issues impact the integrands shape through the whole range of t -values. The suppression of the integrand's maximum for small t is related to large momenta $|\mathbf{q}| \sim \Lambda$ in $1/\tilde{P}_1^3$. On the other hand, the proportionality of \dot{R}_1 to $k^2 \sim e^{2t}$ scales the integrand down for $t \rightarrow -\infty$. Thus the flow finally *dies away* which should not be confused with a fixed point we discussed in eq. (4.18) for the dimensionless quantity λ_k/k .

The arbitrarily chosen numerical value of the microscopic momentum scale Λ serves as a technical degree of freedom to tune the numerical value of the integrand which, in turn, triggers the numerical accuracy of β_λ . In principle, Λ can be arbitrarily set. Indeed, for the specific flow plotted here, we observed independence of $\lambda_{t \rightarrow -\infty}/\Lambda$ over a wide range of Λ -values, namely $\sim 10^{-3} \lesssim \Lambda \lesssim 10^2$. However, if the numerical value of Λ becomes too large or negligible small, numerical rounding errors arising during the evaluation of the β -function's integrand. In consequence they introduce numerical instabilities. Since this behavior strongly depends on the integrand itself, the optimal choice needs to be figured out *experimentally*, e.g. by some principle of *minimum sensitivity*, i.e. minimal deviation of results on variation of Λ .

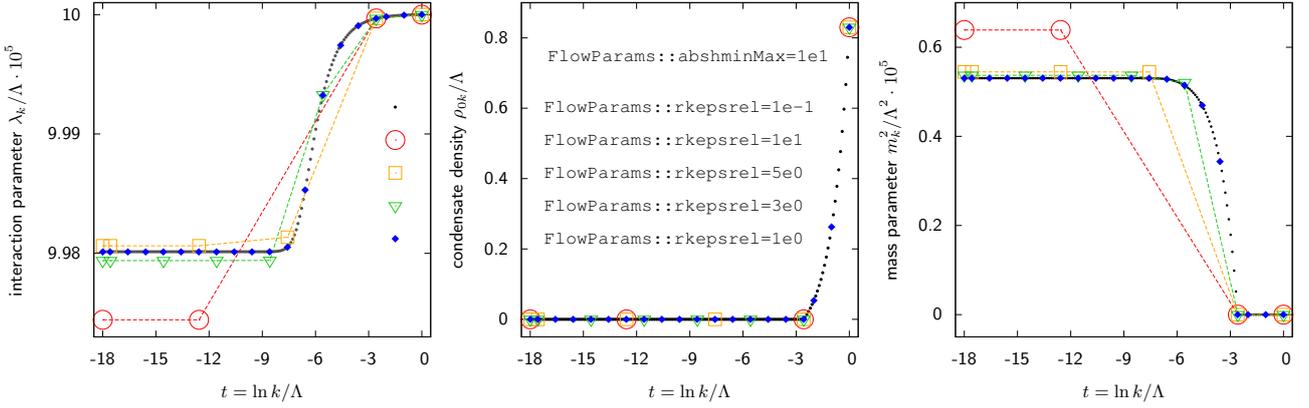


Figure 4.3: *Accuracy of the evolution of the flow by the routine `frgFlow2()`.* We plot the simultaneous evolution of the interaction parameter λ_k (left), the condensate density ρ_{0k} (center), and the mass parameter m_k^2 (right) versus the renormalization scale t . Compared to [fig. 4.4](#) we show the reverse situation where we fix the control parameter `FlowParams::abshminMax` to 10—i.e. the discrete stepsize is nearly unrestricted, $\Delta t \leq 10$ (!)—and the relative precision of advancing the flow, `FlowParams::rkepsrel`, is varied. The legend is spread out to the panels to the left (data symbols) and to the center (precision parameter value). The convergence of the result on reducing the error for advancing the flow becomes apparent. Although Δt is allowed to take large values at the level of `frgFlow2()` by means of the upper limit `FlowParams::abshminMax`, the underlying *GSL* routine adapts the corresponding step size such that the error bound `FlowParams::rkepsrel` stays fulfilled. Note: For each Δt there is a corresponding *pre*-evolution (cf. [section 3.2.2](#)) in order to determine a unique evolution step that complies with the error bound regarding all flowing quantities. Even if the β -function of one quantity indicates a *slow* evolution, a single *rapid* change in one flowing quantity enforces a small Δt for advancing all quantities in the end.

advance a given set of flowing quantities by means of `frgFlow2()` (cf. [fig. 3.1](#)). The tests are summarized by [figs. 4.3](#) and [4.4](#). Both plots contain an explicit switching between SSB and SYM during the flow, where [fig. 4.4](#) explicitly demonstrates the mutual exclusive flow of the condensate density ρ_{0k} and the mass parameter m_k^2 , respectively. Besides these control parameters there is even more options to tune the numerical flow. As mentioned in the caption of [fig. 4.2](#), the numerical value of Λ controls the numerical stability of determining the β -functions. In addition, we have `FlowParams::eps{abs|rel}` at our disposal. They determine the accuracy of the (multi-dimensional) spatial integration which is implemented by the *CUBA* library.

At this stage we feel the need to underline our agenda for the significance of carefully building a library for functional renormalization: The ability to easily control the numerics down to its very bottom distinguishes `libfrg` from approaches which include *ready to go* software packages as e.g. *Mathematica*. With respect to high performance computing, a serious realization needs accessible checks from scanning technical specification parameters. To this end, each investigation of physical results has to be cross-checked against such technical issues. These benchmarks provide a measure for

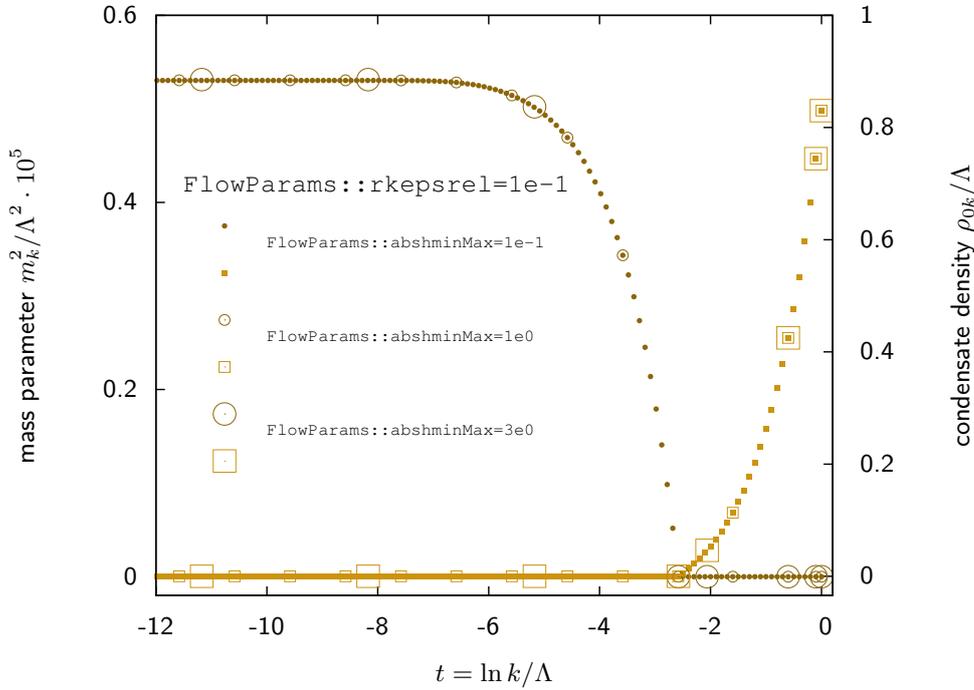


Figure 4.4: *Explicit phase transition and adaptive step size.* This plot demonstrates the mutual exclusive flow of the condensate density ρ_{0k} (ordinate to the right, ocher squares) and the mass parameter m_k^2 (left y -axis, olive-green circles), respectively. As obvious from the plot we start the system within the sponaneous symmetry broken phase ($\rho_{0t=0} \neq 0$, $m_{t=0}^2 = 0$). After the condensate density has dropped to zero ($t \approx -2.5$), m_k^2 starts to flow until the flow *dies away* (cf. fig. 4.2). Would the mass parameter have been dropped to zero at some $t < -2.5$ before, a condensate would build up again.

We also check the dependence of the flow on limiting the maximal step size allowed for advancing flowing quantities. It is controlled via `FlowParams::abshminMax`. The larger the data symbols the larger the upper bound for the discrete evolution steps Δt . The relative precision of advancing the flow via the *GSL* Runge–Kutta integrator `FlowParams::rkepsrel` is fixed to $1/10$. Here, all results collapse to a single curve to demonstrate the quality of the chosen *Cash-Karp* (4,5) routine. Observe the dynamics of the adaptive step size algorithm (cf. discussion around eq. (3.10)): When large step sizes are allowed (large squares/circles), Δt increases from¹⁴ $t = 0$ up to the phase transition. The procedure `stepEstimate()` depicted in fig. 3.8 ensures a precise determination of an accurate t value where `frgFlow2()` switches between ρ_{0k} and m_k^2 —even if previous steps have been large compared to the distance to the phase transition at $t \approx -2.5$. Hence, all data points coincide there. For $t \lesssim -2.5$ the step size nearly instantaneously reaches its maximal value allowed by `FlowParams:abshminMax`.

estimating systematic errors which in turn serve as a tool to specify the reliability of numerical results.

4.2.2 Detecting Fixed Points & Recovering Mean Field Theory

To grasp the existence of fixed points, let us reexamine the set of eqs. (4.12) to (4.14). The flow of the interaction parameter λ_k in the SYM phase is proportional to λ^2 if, loosely speaking, the flow of the mass parameter m_k^2 is *weak*. The β -function $\dot{\lambda} = \beta_\lambda \sim \lambda^2$ obviously exhibits a (trivial/Gaussian¹⁵) fixed point $\beta_\lambda = 0$ at $\lambda = 0$, only. But as we stressed around eqs. (4.17) and (4.18) scaling at a second order phase transition is associated by fixed points of the dimensionless/rescaled version of a flowing observable/quantity. Since λ_k/k introduces another contribution $\sim -\lambda$, the dimensionless interaction parameter is governed by a β -function of the type $-\lambda + \lambda^2$. The qualitative feature of this non-trivial fixed point is investigated by fig. 4.5. However, in the phase with SSB the situation is less obvious. We will discuss this issue when tuning the system to criticality in figs. 4.6 and 4.7.

Inspired by the flow equations in the limit $\rho_0 \rightarrow 0$ one might introduce the convenient definition

$$\tilde{\rho}_0 = \lambda \rho_0 \quad (4.20)$$

to highlight the structure of the β -functions, e.g. that of λ_k as follows:

$$\dot{\lambda} \equiv 2\lambda^2 \int I_\lambda(q, \tilde{\rho}_0) \sim \lambda^2 k^5 I_\lambda(k, \tilde{\rho}_0) \quad \text{with}^{16} \quad \dot{\int} \equiv \int \dot{R} \quad . \quad (4.21)$$

A rough treatment of the scale dependent derivative of the exponential regulator (cf. fig. 3.12) approximates it strongly peaked around $k = q$ such that the integration collapses to momenta of order k . Besides the explicit λ^2 -term the λ -dependence β_λ of $\dot{\lambda}$ depends on the value of $\tilde{\rho}_0$ only. A plot of I_λ at fixed k (i.e. fixed $q = k$ in P_1) is shown by the right panel of fig. 4.7. In the special case where $\lambda_k \rho_{0k}$ stays constant during the flow, the $\beta_\lambda \sim \lambda^2$ is a reasonable assumption, but as soon as $\tilde{\rho}_{0k}$ starts to flow the fixed point structure of β_λ might become modified.

Let us address the system's behavior at and near the phase transition. Figures 4.6 and 4.8 summarize the main results. From a quantitative point of view the outcome seems to be *disappointing*, but it should be taken into account that we intended to start our numerical survey by the most simplest setting in order to provide a first test of the numerical framework given by `libfrg`. However, our *poor man's approximation* confirms qualitative features of the underlying physics and, concerning the critical exponents, we reproduced at least mean field results. In particular, by means of functional renormalization techniques, those values have been observed by the same crude approximation for the Ising model ($N = 1$) in [CDMV03], Fig. 3, data for $n=2$. Moreover, we should keep in mind that starting with a set of dimensionless flow equations from the outset is much more adapted to the quest for quantitatively precise critical exponents. This observation explicitly confirms the bit of jargon “*It is convenient to introduce/work with renormalized/dimensionless variables ...*”. frequently found in the literature without further justification. In particular the effect of the flow *dieing away* (cf. fig. 4.2) is

¹⁵ $\lambda = 0$ corresponds to a non-interacting theory which is represented by a partition function with Gaussian weights: λ reflects the coefficients in front of the φ^4 -term which is dropped for $\lambda = 0$ and one stays with the quadratic coefficient m^2 .

¹⁶The notation $q = k$ is understood in the loose sense described in footnote 50 of chapter 3.

not present: Integrals run over q^2/k^2 being regulated by $u_k(q^2/k^2)$ (cf. eq. (3.55)) such that they do not become suppressed for $k \rightarrow 0$.

It is rather surprising that the numerics works that stable when sticking to the dimensionful description and explicitly dividing the flowing observable O_k by the appropriate power of k , namely $k^{[O]}$. Hence, we are convinced that further effort will improve the results.

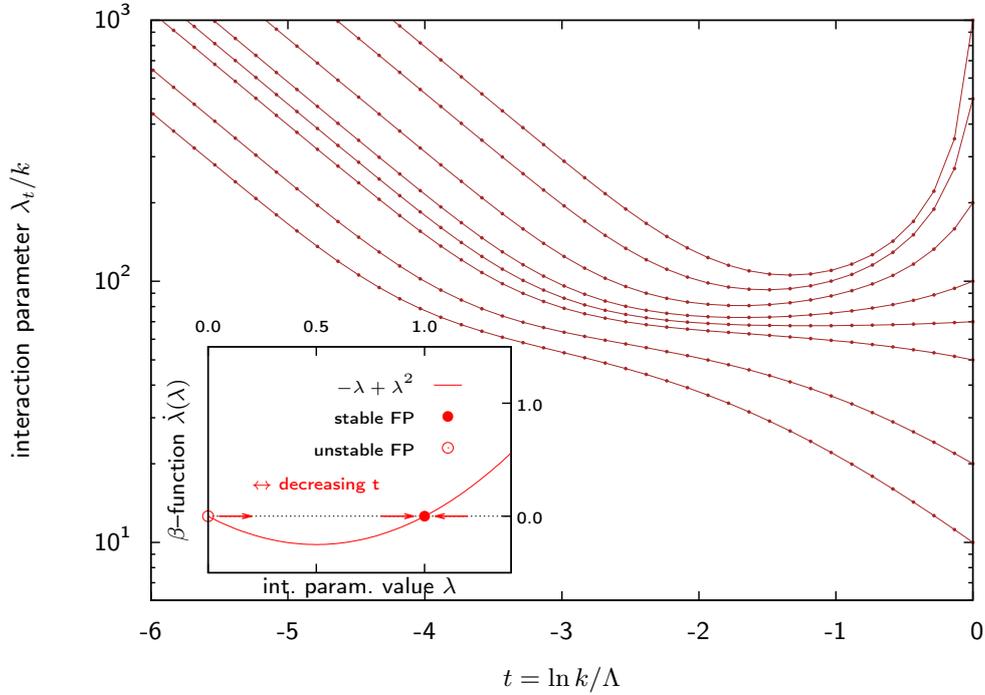


Figure 4.5: *Stable fixed point for λ_t/k in the symmetric phase.* We show the flow of the dimensionless interaction parameter in the SYM phase, where the corresponding β -function is of type $-\lambda + \lambda^2$ (red line of inset). Then, there exists a non-zero fixed point λ_* (filled, red dot of inset) for which $\dot{\lambda} = 0$. Moreover, it is stable for decreasing t , i.e. in the (*long-time*) limit $t \rightarrow -\infty$, λ_t evolves towards $\lambda_* \neq 0$. In contrast, $\beta_{\lambda_*=0} = 0$ is unstable (open, red dot of inset) since decreasing t refers to positive *tangent* $-\partial_\lambda \beta_\lambda|_{\lambda_*=0}$. The main figure plots *trajectories* λ_k for several initial values $\lambda_{k=\Lambda}$. The one that stays approximately constant is closest to the stable fixed point. But why does it start to deviate from a horizontal line around $t = -2.5$? It goes back to the fact that the flow eventually dies away while there remains an explicit division by k . In fact the plot is logarithmic in k and λ_k ; the abscissa prints $t \sim \ln k$ and the plot's ordinate is logarithmically scaled. Therefore, all curves end up as straight lines for t sufficiently small: The flow dies away and we get $\sim 1/k$ scaling.

Deviating from the fixed point solution by starting with smaller or larger $\lambda_{k=\Lambda}$ shows the qualitative attractive character of the fixed point. Even for one order of magnitude away from λ_* the trajectories flow towards the fixed point solution until the flow dies away.

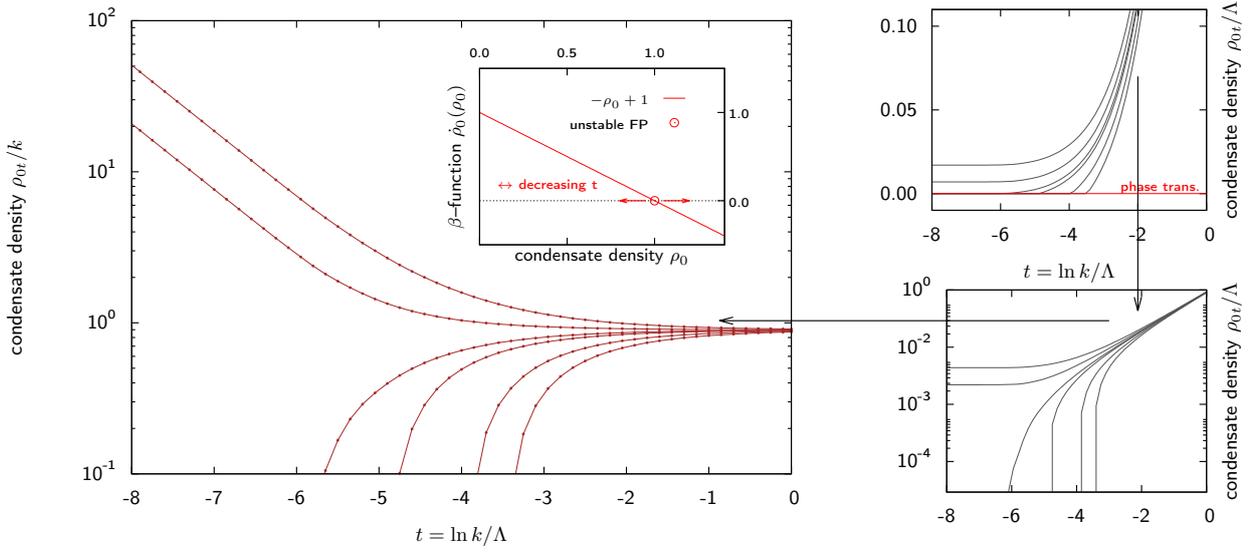


Figure 4.6: *Scaling of the condensate density ρ_{0k} at the phase transition—the unstable fixed point.* Here, we explicitly tune the system to the second order phase transition. As discussed at the beginning of [section 4.2](#) a phase transition coincides with a vanishing condensate density ρ_{0k} in the limit $t \rightarrow -\infty$. To approach this situation we implemented a procedure in `reLON.cpp` that realizes nested intervals. It works as follows: Start with an arbitrary value $\rho_{0t=0} \neq 0$ and advance it up to a t -value where the flow has been definitely died away (here, we took $t = -15$). If ρ_{0t} drops to zero, we record the initial value as lower bound ρ_{\min} for the critical condensate density ρ_{crit} . In cases $k \rightarrow 0$ yields SSB we appropriately update the upper bound ρ_{\max} which is set to infinity at the beginning of the procedure. The new estimate for ρ_{crit} for which this process gets repeated reads $(\rho_{\max} - \rho_{\min})/2$ if $\rho_{\max} < \infty$ and $2\rho_{\min}$, otherwise. According to the standard *IEEE 754-1985*^[EE85], 64 bit floating point numbers `double` store a mantissa of 52 bit. When ρ_{\max} becomes finite, the estimation procedure for ρ_{crit} , on average, decreases the length of the interval $[\rho_{\min}, \rho_{\max}]$ by a factor $1/2$, i.e. the precision of ρ_{crit} becomes increased by one bit, roughly. Therefore we iterate the procedure described above about 52 times.

The right upper diagram illustrates a series of flows close to the phase transition. Part of them hit the line $\rho_{0t} = 0$ (red, points for data omitted, see main plot), the rest stays with a finite condensate density up to the point where the flow dies away. In order to visualize the fixed point we need to logarithmically rescale the ordinate (right lower plot): The power law decrease of ρ_{0k} becomes transparent by a straight line (abscissa is logarithmically scaled due to $t \sim \ln k$), but does it scale like $k^{[\rho]} = k$ itself? Explicitly dividing ρ_{0k} by k and replotting (main panel to the left) visually confirms the scaling hypothesis.

However, compared to [fig. 4.5](#) we encounter an unstable fixed point resulting from a β -function schematically depicted by the inset (red line). Slight deviation drives the system to either SYM or SSB. A fact which one might exploit in order to get one's hand on the classical temperature T_{cl} discussed at the beginning of [section 4.1](#). Since $1/T_{\text{cl}}$ becomes absorbed into the microscopic couplings which define $S[\eta] \sim \Gamma_{k=\Lambda}$, it is a reasonable assumption that they linearly depend on $T_{\text{cl}} - T_{\text{crit}}$ for small deviation. T_{crit} denotes the critical (classical) temperature where the second order phase transition takes place. In particular we expect $(\rho_{\text{crit}} - \rho_{0k=\Lambda})/\rho_{\text{crit}} \sim (T_{\text{cl}} - T_{\text{crit}})/T_{\text{crit}}$; for $T_{\text{cl}} < T_{\text{crit}}$ (SSB) the *microscopic* condensate density needs to be above the critical value ρ_{crit} and vice versa (SYM).

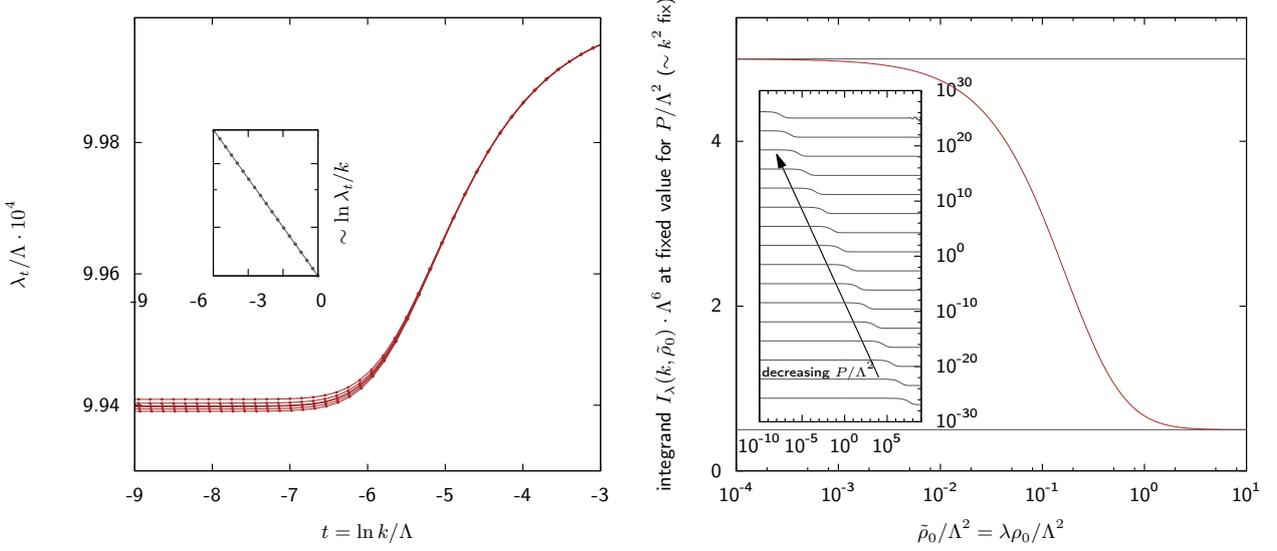


Figure 4.7: Is λ_t really scaling with k , i.e. is a fixed point for λ guaranteed? By this plot we would like to demonstrate the limitation of our simple truncation. By means of the scaling hypothesis it is expected that λ_k scales as k , if we are close to the second order phase transition. But the double-logarithmic inset of the left plot does not indicate a fixed point. There are two main aspects that might hide the scaling of the interaction parameter: a) The flow dies away too fast for λ_k to reach the stable fixed point demonstrated in fig. 4.5. This option underlines the efficiency of working with the scaling form of the flow equations if the focus is on the fixed point structure of the underlying theory. b) The truncation is insufficient to correctly account for all features of the *real*/physical fixed point(s).

From a technical point of view, let us rate the relevance of option b). As briefly touched by eq. (4.21) we might interpret the emergence of a finite condensate density ρ_{0k} as a modification of the β -function β_λ which is proportional to λ^2 in the SYM phase, i.e. a rough estimate yields: $\dot{\lambda} \sim \lambda^2 \rightarrow \lambda^2 I_\lambda(k, \rho_0 \lambda)$ for SSB where we explicitly have

$$I_\lambda(k, \tilde{\rho}_0) = \frac{5\tilde{P}^3 - 12\tilde{P}^2\tilde{\rho}_0 + 15\tilde{P}\tilde{\rho}_0^2 - 4\tilde{\rho}_0^3}{(\tilde{P}^2 - \tilde{\rho}_0^2)^3}, \quad \text{and} \quad \tilde{P} = P_1|_{q=k} + \tilde{\rho}_0$$

represents the modified inverse propagator evaluated at momentum $q = k$. We dropped *dieing away factors* k from \dot{R} and the integration volume element, i.e. $\dot{J} \sim k^5$. Furthermore, we restrict to the broken phase such that $\rho_0 > 0$. Note the following two facts: i) I_λ is not singular due to $\tilde{P}^2 - \tilde{\rho}_0^2 = P(P + 2\tilde{\rho}_0)$ (the regularized bosonic (inverse) propagator $P \equiv P_1|_{q=k} > 0$) and ii) for $\tilde{\rho}_0 \rightarrow \infty$ we obtain the limit $I_\lambda \rightarrow 1/2 = \text{const.}$. The main plot of the right panel shows I_λ for $P/\Lambda^2 = 1$ in order to illustrate that it significantly varies for $\tilde{\rho}_0/\Lambda^2 \approx 10^{-3} \dots 1$ only. The inset depicts the dependence of I_λ on P/Λ^2 which takes discrete values $10^{-7}, 10^{-6}, \dots, 10^8$. By inspection, the non-constant region of I_λ approximately linearly shifts on variation of P . We have $P|_{q=k} \sim k^2$. As explicitly demonstrated by fig. 4.6, ρ_0 scales as k at the phase transition. According to the flow of the interaction parameter (left plot), we have $\lambda_k \sim 10^{-3} \sim \text{const.}$ and thus $\tilde{\rho}_0$ eventually ends in a region where $I_\lambda(k, \tilde{\rho}_0) \approx \text{const.} = 1/2$. Therefore the fixed point structure of β_λ becomes stable for $t \rightarrow -\infty$. We conclude that missing the scaling solution for the interaction parameter seems to be an artefact of the dimensionful flow which dies away for sufficient small scales k .

However, our consideration should also demonstrate that the condensate is capable to significantly modify β -functions: If we fix ρ_0 and P at some scale k there is a region inbetween the *plateaus* (cases $\tilde{\rho}_0 \rightarrow 0$ and $\tilde{\rho}_0 \rightarrow \infty$) of I_λ where we might approximate $I_\lambda \sim \text{const.} - \ln \lambda$.

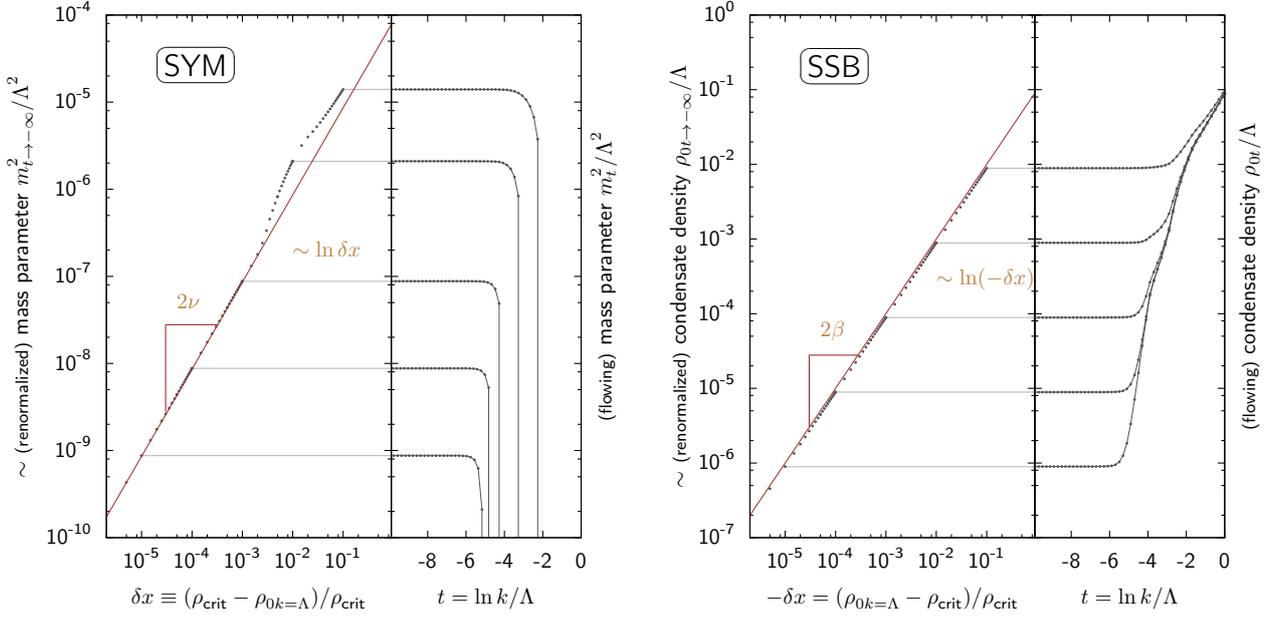


Figure 4.8: Mean field critical exponents ν (left) and β (right) from the simplest local potential approximation. To close the discussion, we intend to extract two critical exponents from our numerics related to a) the divergence of the correlation length ξ when approaching the phase transition from the symmetric phase (left panel) and b) the vanishing of the condensate near the transition within the broken phase (right panel), respectively. As we already stressed: Results from the cutting edge of present research are not expected, since we (intentionally) stayed with the dimensionful flow equations, we employ the most simplest truncation and did check that the flow dies too early in order for the interaction parameter to become relaxed to its stable fixed point.

Nevertheless, our data illuminate to which extent our approach is able to capture critical behavior of the system. In order to obtain the value of the critical exponent ν we need to start the flow near the phase transition such that we end up with a finite mass parameter in the limit $t \rightarrow -\infty$. Associating $\sqrt{m_{k=0}^2} \sim \xi^{-1}$ and recalling the proportionality of $(\rho_{\text{crit}} - \rho_{0k=\Lambda}) / \rho_{\text{crit}} \sim (T_{\text{cl}} - T_{\text{crit}}) / T_{\text{crit}}$ we are able to extract the scaling of $\xi \sim (T_{\text{cl}} - T_{\text{crit}})^{-\nu}$ by plotting $\ln m_{t \rightarrow -\infty}^2$ versus $\ln \delta x = \ln(\rho_{\text{crit}} - \rho_{0k=\Lambda})$. In the limit $\delta x \rightarrow 0$ the corresponding $m_{k=0}^2$ values should converge to a straight line with slope 2ν : $\ln m_{k=0}^2 \sim -2 \ln \xi \sim +2\nu \ln(T_{\text{cl}} - T_{\text{crit}}) \sim 2\nu \ln(\rho_{\text{crit}} - \rho_{0k=\Lambda})$. As obvious from the left part of the left panel the expected linear behavior (brown line) sets in for $\delta x \lesssim 10^{-3}$. The right panel of the left plot depicts demonstrative flows that lead to the data to the left. Unfortunately the result does not reach beyond the mean field approximation where $\nu = 1/2$. The origin has been extensively discussed above. The same procedure is easily repeated for the critical exponent β , if we start the initial value of $\rho_{0k=\Lambda}$ slightly above ρ_{crit} (right diagram). Then, we end up with a finite condensate density. Its square root characterizes the magnetization M which itself relates to the critical exponent β : $M \sim (T_{\text{crit}} - T_{\text{cl}})^\beta$, hence $\ln \rho_{0k=0} \sim 2 \ln M \sim 2\beta \ln(\rho_{0k=\Lambda} - \rho_{\text{crit}})$. Again we recover the mean field value $\beta = 1/2$.

4.3 The Story Revised: Scaling Form of the Flow Equations

As we learned from [section 4.2](#) dimensionful quantities are not suited to extract critical exponents near (second order) phase transitions. Since [section 4.2.2](#) solely left us with qualitative features of the critical physics near such a transition, we eventually aim at providing a quantitative benchmark to demonstrate the abilities of `libfrg`. It is our purpose to reiterate the analysis of the previous section with the same most simplest ansatz for the effective action, i.e. we take

$$\Gamma_k[\varphi] = \frac{1}{2} \sum_{a=1}^N \int_x [\partial_x \varphi_a(x)]^2 + \int_x U_k(\rho(x)) \quad \text{where} \quad U(\rho(x)) = \begin{cases} m_k^2 \rho(x) + \frac{1}{2} \lambda_k \rho(x)^2 & \text{(SYM)} \\ \frac{1}{2} \lambda_k (\rho(x) - \rho_{0k})^2 & \text{(SSB)} \end{cases} \quad (4.22)$$

with $\rho(x) = \frac{1}{2} \sum_{a=1}^N \varphi_a(x)^2$ for an $O(N)$ -symmetric model. Adapting our notation to that of [appendix A](#) we compute

$$\Gamma_{k,(a,x)(b,y)}^{(2)} = [U_k''(\rho_x) \varphi_{(a,x)} \varphi_{(b,y)} + \delta_{ab} U_k'(\rho_x)] \delta_{xy} - \frac{\delta_{ab}}{2} [\partial_x^2 \delta_{xy} + \partial_y^2 \delta_{xy}] \quad . \quad (4.23)$$

Fourier transforming, setting the condensate density from $\rho(x) = \text{const.}$ to $\rho_{0k} \stackrel{!}{=} \frac{1}{2} \varphi_{1k}^2$, and $R_{k,(a,q)(b,q')} = R_k(q) \delta_{-qq'} \delta_{ab}$ yields the (scale dependent) propagator

$$G_{k,(a,q)(b,q')} = [q^2 + R_k(q) + U_k'(\rho_{0k}) + 2\rho_{0k} U_k''(\rho_{0k}) \delta_{1a}]^{-1} \delta_{-qq'} \delta_{ab} \quad . \quad (4.24)$$

Finally the flow equation leaves us with

$$\dot{U}_k = \frac{1}{2} \int_q \left[\frac{1}{q^2 + R_k(q) + M_1} + \frac{N-1}{q^2 + R_k(q) + M_0} \right] \dot{R}(q) \quad \text{with} \quad \begin{cases} M_0 \equiv U_k'(\rho_{0k}) \\ M_1 \equiv U_k'(\rho_{0k}) + 2\rho_{0k} U_k''(\rho_{0k}) \end{cases} \quad (4.25)$$

Switching to rescaled/dimensionless quantities incorporates the definitions

$$\begin{aligned} u_k &\equiv U_k/k^d, & \tilde{\rho}_k &\equiv \rho/k^{d-2}, & \tilde{M}_{0/1} &\equiv M_{0/1}/k^2 = \begin{cases} \partial u_k / \partial \tilde{\rho}_k & \equiv u'_k \\ \partial u_k / \partial \tilde{\rho}_k + 2\tilde{\rho}_k \partial^2 u_k / \partial \tilde{\rho}_k \partial \tilde{\rho}_k & \equiv u'_k + 2\tilde{\rho}_k u''_k \end{cases} \\ \tilde{m}_k^2 &\equiv m_k^2/k^2, & \tilde{\lambda}_k &\equiv \lambda_k/k^{4-d}, & \text{and} & r_k(q) &\equiv R_k(q)/q^2 \quad . \end{aligned} \quad (4.26)$$

When selecting the Litim regulator $R_k(q) = (k^2 - q^2)\Theta(k^2 - q^2)$ we arrive at the flow of the dimensionless effective potential as

$$\dot{u} = -du + (d-2)\tilde{\rho}u' + \frac{1}{dC_1(d)} \left[1/(1 + \tilde{M}_1) + (N-1)/(1 + \tilde{M}_0) \right] \quad (4.27)$$

with $C_1(d)$ from eq. (3.16)¹⁷. We dropped the k -label, $u' \equiv \partial u / \partial \tilde{\rho}$, $u'' \equiv \partial^2 u / \partial \tilde{\rho}^2$, ... is understood. Taking appropriate $\partial_{\tilde{\rho}}$ -derivatives and evaluating eq. (4.27) at $\tilde{\rho}_0$ leads to the flow equations

$$\dot{m}^2 = \beta_{\tilde{m}^2} \equiv \begin{cases} -2\tilde{m}^2 - \frac{N+2}{dC_1(d)} \tilde{\lambda} / (1 + \tilde{m}^2)^2 & \text{(SYM)} \\ 0 & \text{(SSB)} \end{cases} \quad (4.28)$$

$$\dot{\rho}_0 = \beta_{\tilde{\rho}_0} \equiv \begin{cases} 0 & \text{(SYM)} \\ (2-d)\tilde{\rho}_0 + \frac{1}{dC_1(d)} \left[3/(1 + 2\tilde{\rho}_0\tilde{\lambda})^2 + N - 1 \right] & \text{(SSB)} \end{cases} \quad (4.29)$$

$$\dot{\lambda} = \beta_{\tilde{\lambda}} \equiv \begin{cases} (d-4)\tilde{\lambda} + \frac{2(N+8)}{dC_1(d)} \tilde{\lambda}^2 / (1 + \tilde{m}^2)^3 & \text{(SYM)} \\ (d-4)\tilde{\lambda} + \frac{2}{dC_1(d)} \left[9/(1 + 2\tilde{\rho}_0\tilde{\lambda})^3 + N - 1 \right] \tilde{\lambda}^2 & \text{(SSB)} \end{cases} \quad (4.30)$$

For our szenario $N = 2$ and $d = 3$ there are three fixed points in the broken phase when solving $\beta_{\tilde{\rho}_0} = \beta_{\tilde{\lambda}} = 0$. We get three fixed points $(\tilde{\rho}_{0*}, \tilde{\lambda}_*)$

$$\left(1/3\pi^2, \frac{3}{2}(\pm\sqrt{3} - 1)\pi^2 \right)_{\pm} \quad \text{and} \quad \left(2/3\pi^2, 0 \right) . \quad (4.31)$$

The last one is *trivial* in the sense that the (dimensionless) interaction parameter $\tilde{\lambda}_*$ vanishes. Since a stable minimum of the effective potential implies a positive interaction parameter we drop $(\tilde{\rho}_{0*}, \tilde{\lambda}_*)_-$. Computing the *stability matrix*^[BTW00,Str94] in the broken phase

$$\left(\begin{array}{cc} \partial\beta_{\tilde{\rho}_0}/\partial\tilde{\rho}_0 & \partial\beta_{\tilde{\rho}_0}/\partial\tilde{\lambda} \\ \partial\beta_{\tilde{\lambda}}/\partial\tilde{\rho}_0 & \partial\beta_{\tilde{\lambda}}/\partial\tilde{\lambda} \end{array} \right) \Big|_{(1/3\pi^2, \frac{3}{2}(\sqrt{3}-1)\pi^2)} \quad (4.32)$$

at the fixed point of interest yields the two eigenvalues $-1.6106\dots$ and $0.3841\dots$. The inverse of the absolute value of the smallest negative one corresponds to the critical exponent that characterizes the divergence of the correlation length at the phase transition. We denote it by $\nu_{\text{SM}} = 0.62085\dots$

It follows from the *scaling relations*^[AS10] of the critical exponents

$$\gamma = (2 - \eta)\nu, \quad \alpha = 2 - d\nu, \quad \text{and} \quad 2 = \alpha + 2\beta + \gamma \quad \Rightarrow \quad \beta = (d-2)\frac{\nu}{2} \quad \text{assuming } \eta = 0 \quad (4.33)$$

which is enforced by our truncation, eq. (4.22). Therefore our computation of the critical exponent that specifies the vanishing of the condensate density reads $\beta_{\text{SM}} = \nu_{\text{SM}}/2 = 0.31042\dots$

Based on the SSB cases of eqs. (4.29) and (4.30) we now have analytical values at hand to be compared with the outcome of the numerical flow advanced by `libfrg`. Figure 4.9 plots an example of a flow tuned to the *non-trivial* fixed point $(\tilde{\rho}_{0*}, \tilde{\lambda}_*)_+$. Exhausting the numerical precision of 64 bit double floating point values by nested intervals as discussed in fig. 4.6, we tune to the critical value¹⁸ $\tilde{\rho}_{\text{crit}}$ defined at the microscopic scale $k = \Lambda$.

¹⁷Recall that we developed a formula for fast implementation of $C_1(d)$ in C/C++: see `solidAngleRotSym()` from `relON.cpp`. Since the β -functions are intensively called through `frgFlow2()` to advance the flow, it is worth to do so (cf. fig. 3.4).

¹⁸To increase the *time* interval t where the flow stays at $(\tilde{\rho}_{0*}, \tilde{\lambda}_*)_+$ one would need to work with numerical precision higher than provided by the `double` format. This implies e.g. `long double`, but there is no standard specifying the number of bits for the mantissa of `long double`. Therefore the precision becomes platform dependent—the reason why *GSL*'s routines do not support floating point precision beyond `double`. To access arbitrary precision numerics, the *GMP* library can be used. Then, routines for computing the flow need to be substituted to support it. A *GPL* compatible option is *ALGLIB*.

In analogy to the procedure introduced by fig. 4.8 we then slightly deviate from $\tilde{\rho}_{\text{crit}}$. Approaching $\tilde{\rho}_{\text{crit}}$ from below yields a finite and constant value of the *renormalized* mass parameter $\tilde{m}^2 k^2$ for k sufficiently small: After $\tilde{\rho}_0$ has dropped to zero, \tilde{m}^2 starts to flow by means of eq. (4.28), cf. the upper plot of fig. 4.9, purple curve. For $\tilde{m}^2 \rightarrow \infty$ such that $\tilde{\lambda}/\tilde{m}^2 \rightarrow \text{const.}$ we are left with $\dot{\tilde{m}}^2 \approx -2\tilde{m}^2$. Indeed, in this case $\dot{\tilde{\lambda}} \approx -\tilde{\lambda}$ and hence \tilde{m}^2 diverges twice as fast as $\tilde{\lambda}$ for $t \rightarrow -\infty$, i.e. $\tilde{m}^2 \sim e^{-2t} \sim k^{-2}$ and $\tilde{\lambda} \sim k^{-1}$, thus

$$m^2 \equiv k^2 \tilde{m}^2 \quad (4.34)$$

becomes constant and $\tilde{\lambda}/\tilde{m}^2 \rightarrow 0$ for $k \rightarrow 0$.

When tuning¹⁹ the dimensionless condensate density at $t = 0$ towards $\tilde{\rho}_{\text{crit}}$ from above we stay with a positive $\tilde{\rho}_0$ in the limit $t \rightarrow -\infty$. As $\tilde{\rho}_0 \rightarrow \infty$, $\tilde{\lambda} > 0$ the symmetry broken versions of eqs. (4.29) and (4.30) assume the form $\dot{\tilde{\rho}}_0 \approx -\tilde{\rho}_0 + c_1$ and $\dot{\tilde{\lambda}} \approx -\tilde{\lambda} + c_2$ with positive constants $c_{1/2}$ vanishing for $N = 1$. When $N > 1$, $\tilde{\lambda}$ flows to a stable (non-trivial) fixed point (cf. inset of fig. 4.5) and $\tilde{\rho}_0$ diverges—the stated approximative form of the flow equations remains valid. For the case of the Ising model ($N = 1$) the dimensionless interaction parameter diverges which also does not spoil the form of the approximated equations. Hence we get a finite condensate density

$$\rho_0 \equiv k \tilde{\rho}_0 \quad \text{in the limit } k \rightarrow 0 \quad . \quad (4.35)$$

According to the relation between the deviation from the critical value of $\tilde{\rho}_0$ at $k = \Lambda$

$$\delta \tilde{\rho}_0 \equiv \tilde{\rho}_{\text{crit}} - \tilde{\rho}_{0k=\Lambda} \sim \rho_{\text{crit}} - \rho_{0k=\Lambda} \sim T_{\text{cl}} - T_{\text{crit}} \quad , \quad (4.36)$$

discussed in fig. 4.6 and at the beginning of section 4.1, and the *classical* temperature T_{cl} we employ the definitions of the critical exponents $\xi^{-1} \sim (T_{\text{cl}} - T_{\text{crit}})^\nu \sim \sqrt{m_{k=0}^2}$ for SYM and $\rho_{0k=0} \sim (T_{\text{crit}} - T_{\text{cl}})^{2\beta}$ for SSB, again. This time we explicitly compute the exponents by (small) finite differences according to

$$2\nu(\delta \tilde{\rho}_0) \approx \frac{\ln m^2(\delta \tilde{\rho}_0 + \epsilon) - \ln m^2(\delta \tilde{\rho}_0)}{\ln(\delta \tilde{\rho}_0 + \epsilon) - \ln \delta \tilde{\rho}_0} \quad \text{and} \quad 2\beta(-\delta \tilde{\rho}_0) \approx \frac{\ln \rho_0(\delta \tilde{\rho}_0 + \epsilon) - \ln \rho_0(\delta \tilde{\rho}_0)}{\ln(\delta \tilde{\rho}_0 + \epsilon) - \ln \delta \tilde{\rho}_0} \quad (4.37)$$

with $\delta \tilde{\rho}_0 \ll \tilde{\rho}_{\text{crit}}$ and ϵ smaller than the smallest value of $\delta \tilde{\rho}_0$ taken. ρ_0 and m^2 are taken close to $k = 0$. $\delta \tilde{\rho}_0$ needs to be positive and negative for ν and β , respectively. The upper panels in fig. 4.10 provide the numerical results for applying eq. (4.37). The approach of the critical exponents to the corresponding values obtained by the exact diagonalization of the stability matrix, eq. (4.32), and the scaling relations, eq. (4.33), becomes obvious. The relative error of

$$\nu \equiv \lim_{\delta \tilde{\rho}_0 \rightarrow 0} \nu(\delta \tilde{\rho}_0) \quad \text{and} \quad \beta \equiv \lim_{\delta \tilde{\rho}_0 \rightarrow 0} \beta(\delta \tilde{\rho}_0) \quad (4.38)$$

compared to ν_{SM} and $\nu_{\text{SM}}/2$ is -0.3% and $+0.5\%$, respectively.

To close our benchmarking we vary the number N of bosonic fields φ_a and repeat the procedure from above to exclude that the quality of the result purely arised by coincidence. We present our findings in

¹⁹For the concrete evaluation of the critical exponents in the following we arbitrarily select $\tilde{\lambda}_{t=0} = 1$.

the lower diagram of [fig. 4.10](#). Note that [[CDMV03](#)] obtains a value $\nu = 0.50\dots$ ($n = 2$ in [Fig. 3](#)) for $N = 1$ with the same truncation as used in this section. Our result is 0.502. To convince the reader that this outcome is no coincidence we extend [eq. \(4.22\)](#) to contain an effective potential with cubic ρ -dependence. Hence [eqs. \(4.28\) to \(4.30\)](#) are enlarged by the flow of a coupling $\tilde{\lambda}_3$. In this case, our calculation yields $\nu = 0.717$ which deviates by approximately -1.8% from the literature where we read off $\nu \approx 0.73$ from [Fig. 3](#) at $n = 3$. Details on our analysis provides [fig. 4.11](#).

For $N = 3$ [[BTW00](#)] mentions $\nu = 0.74$ for vanishing anomalous dimension (p. 31, top paragraph). Our computation yields 0.727 which corresponds to a relative deviation of -1.7% . Compared to the analytical value $\nu_{\text{SM}} = 0.72981063\dots$ our outcome deviates by -0.4% .

In the case $N \rightarrow \infty$ we can simplify the flow of the dimensionless condensate density and the dimensionless interaction parameter in the broken phase according to ($d = 3$)

$$\dot{\tilde{\rho}}_0 \approx -\tilde{\rho}_0 + \bar{N} \quad \text{and} \quad \dot{\tilde{\lambda}} \approx -\tilde{\lambda} + 2\bar{N}\tilde{\lambda}^2 \quad \text{with} \quad \bar{N} \equiv N/6\pi^2, \quad \text{respectively.} \quad (4.39)$$

The non-trivial fixed point reads $(\tilde{\rho}_{0*}, \tilde{\lambda}_*)_+ = (\bar{N}, 1/2\bar{N})$. Hence $2\tilde{\rho}_{0*}\tilde{\lambda}_* = 1$ which justifies our approximation neglecting powers of the term $\sim 1/(1 + 2\tilde{\rho}_0\tilde{\lambda})$ when being added to \bar{N} . The stability matrix at the fixed point becomes $\text{diag}(-1, 1)$ and hence $\nu_{\text{SM}} \xrightarrow{N \rightarrow \infty} 1/|-1| = 1$. The lower panel of [fig. 4.10](#) confirms this convergent trend when increasing the number of fields. The same holds for $\beta_{\text{SM}} \xrightarrow{N \rightarrow \infty} \frac{1}{2}$.

As we explicitly demonstrated in this section, the rescaled, dimensionless version of the flow equations are much more suited to extract sensible values for the critical exponents on a quantitative level. The origin can be traced back to numerical precision. Even for the dimensionless flow there is numerical limitation in the sense that the instable direction of the (non-trivial) fixed point *scans* every digit of $\tilde{\rho}_{\text{crit}}$ when $t \rightarrow -\infty$. However, the precision of double is sufficient to tune $(\tilde{\rho}_{0t}, \tilde{\lambda}_t)$ to the fixed point in the interval $t \in [-15, -20]$, approximately. We depict this statement by the side panels of [fig. 4.9](#).

When working with the dimensionful formulation in [section 4.2](#) there is an additional obstacle that decreases numerical precision. In [fig. 4.2](#) we noted the effect of *dieing away* of the integrand for $k \rightarrow 0$ that contributes to the corresponding β -function. In particular for small k the values become negligibly small, eventually decreasing below the numerical precision which leads to constant flowing quantities at some finite t . This fact provides the reason why the trajectories λ_t/k and ρ_{0t}/k start to become straight lines in the logarithmic plot of [fig. 4.5](#) ($t \sim \ln k$) around $t = -2\dots -4$ and for [fig. 4.6](#) at $t \approx -6$, respectively. But as we observed in [fig. 4.9](#) scaling behavior sets in for $t \lesssim -15$. The top plots of [fig. 4.10](#) underline that it is mandatory to stay quite close to the phase transition in order to obtain critical exponents independent of the precise deviation $\delta\tilde{\rho}_0$. Otherwise values substantially different from the correct outcome are possible—see in particular the right top plot for the exponent β : It drops from ~ 0.41 to ~ 0.31 which corresponds to a relative decrease of about 24%.

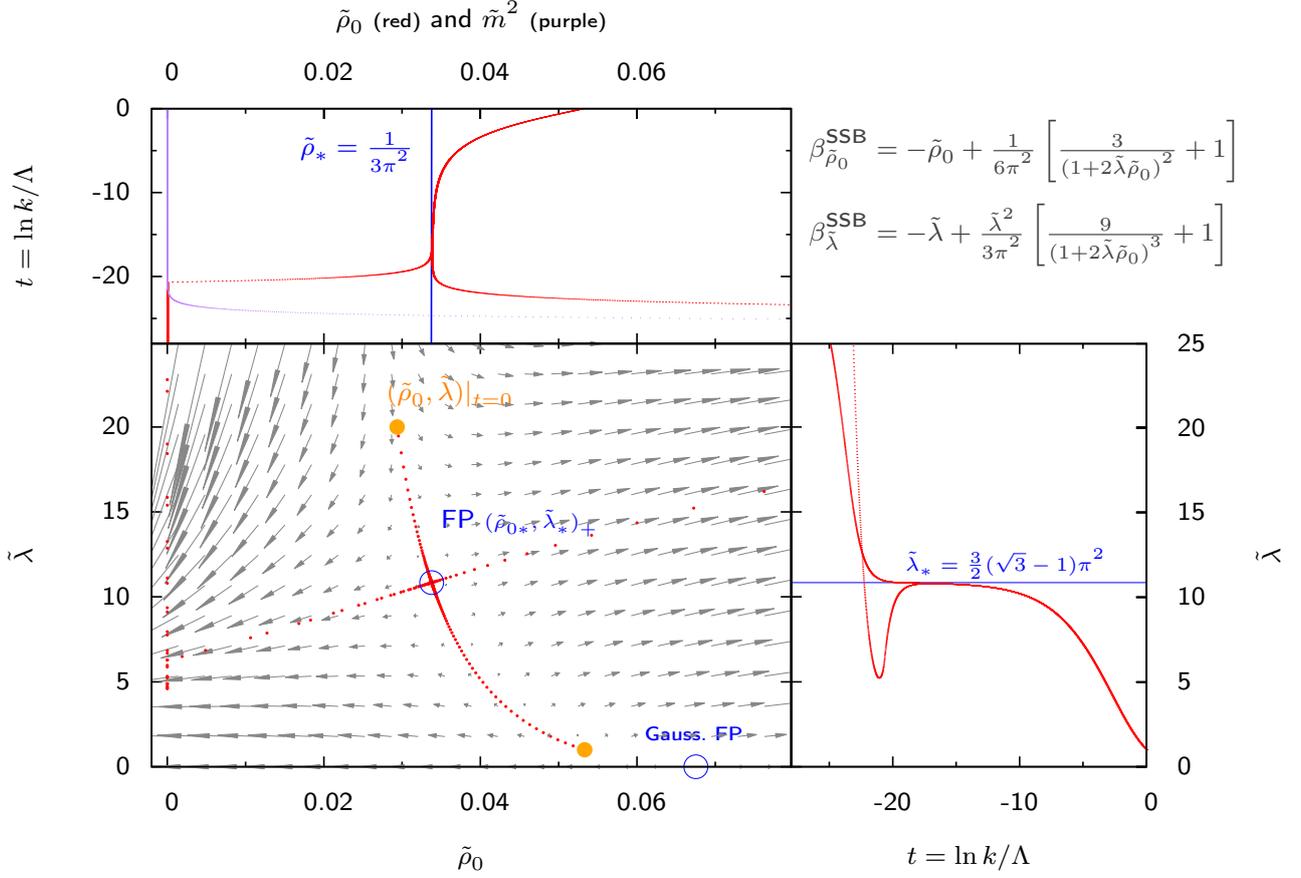


Figure 4.9: *Dimensionless flow to the fixed point for SSB.* This figure provides an illustration of the flow tuned to the phase transition. The gray arrows in the main plot indicate the *velocity field* due to the β -functions for SSB from eqs. (4.29) and (4.30) in three spatial dimensions and $O(2)$ symmetry. They are explicitly written down in the left upper corner. The open circles (blue) mark the possible fixed points (FP) $\beta_{\tilde{\rho}_0}^{\text{SSB}} = \beta_{\tilde{\lambda}}^{\text{SSB}} = 0$ for $\tilde{\rho}_{0*}$ and $\tilde{\lambda}_{*}$ positive. The *Gaussian* fixed point refers to $\tilde{\lambda}_{*} = 0$. However, the (*non-trivial*) fixed point is of interest for the critical physics. Its coordinates are explicitly given by blue lines in the adjacent plots on top of and left to the main panel. We chose two initial conditions (orange filled circles) where $\tilde{\lambda}_{t=0}$ equals to 1 and 20, respectively. The side panels show the evolution starting from $\tilde{\lambda}_{t=0} = 1$, only.

Employing nested intervals as in fig. 4.6 we tune $\tilde{\rho}_{0t=0}$ such that the flow (red dots) approaches the non-trivial fixed point. Data are plotted for equally spaced time- t -intervals to visually demonstrate that the flow spends a long time near the phase transition (increasing density of points near $(\tilde{\rho}_{0*}, \tilde{\lambda}_{*})_+$). Since the $\tilde{\rho}_0$ -direction is unstable, numerical precision limits the fine tuning such that the flow eventually quickly deviates from the critical point. Depending on whether $\tilde{\rho}_{0t=0}$ is slightly below or above $\tilde{\rho}_{\text{crit}}$, $\tilde{\rho}_{0t}$ drops to zero or diverges to infinity, respectively. Both cases are plotted in the side panels. For $\tilde{\rho}_{0t}$ dropping to zero there is an explicit switching of the flow equations to SYM where \tilde{m}^2 starts to flow (purple dots in the upper panel).

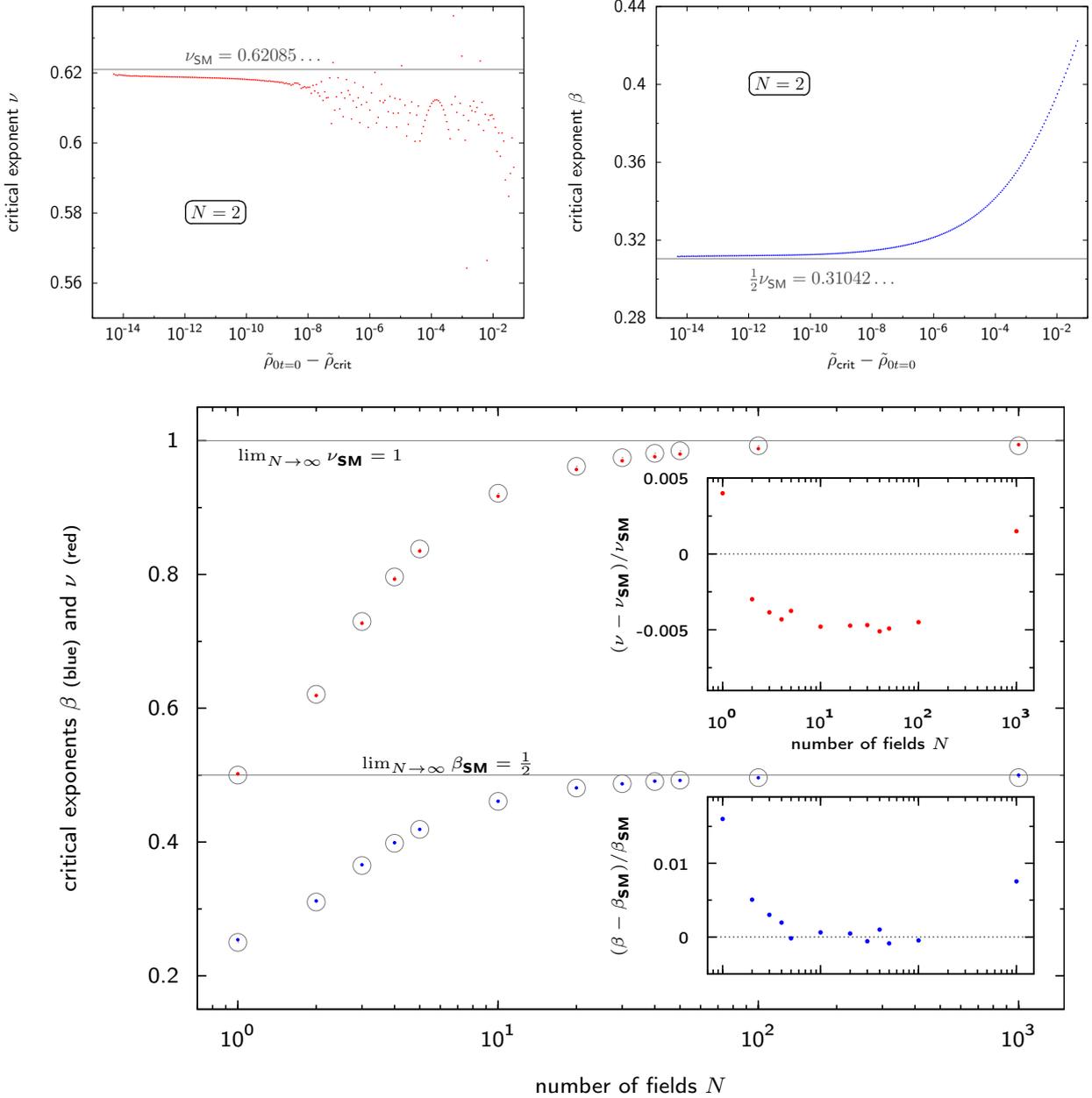


Figure 4.10: *Critical exponents from the dimensionless flow.* Computing critical exponents by two different methods we provide a quantitative benchmark of `libfrg`. The upper plots demonstrate how we extract critical exponents for SYM (top left) and SSB (top right). According to the finite differences formulae, eq. (4.37), we compute the tangents of $\ln k^2 \tilde{m}^2 \sim 2\nu \ln |\delta \tilde{\rho}_0|$ and $\ln k \tilde{\rho}_0 \sim 2\beta \ln |\delta \tilde{\rho}_0|$ in the limit $k \rightarrow 0$ for various small deviation $\delta \tilde{\rho}_0$ from the critical value $\tilde{\rho}_{\text{crit}}$ at the microscopic scale $k = \Lambda$. For $\delta \tilde{\rho}_0 \rightarrow 0$ this procedure converges to the critical exponents we aim at. The horizontal lines (gray) indicate the critical exponents ν_{SM} and $\beta_{\text{SM}} = \nu_{\text{SM}}/2$ obtained from (analytically) linearizing the β -functions around the non-trivial fixed point and assuming the validity of the scaling relations for vanishing anomalous dimension, eq. (4.33).

The lower main plot summarizes our findings when generalizing from $O(2)$ to arbitrary N . The dots denote data for the critical exponents derived from the flow as illustrated in the two top panels. The gray open circle's center represents the corresponding values from the analysis of the stability matrix. For quantitative comparison we show the relative error between the analytical computation and the numerical results. Almost all of them are clearly below the 1% threshold. Moreover, the convergence of the exponents for $N \rightarrow \infty$ derived in the main text is confirmed.

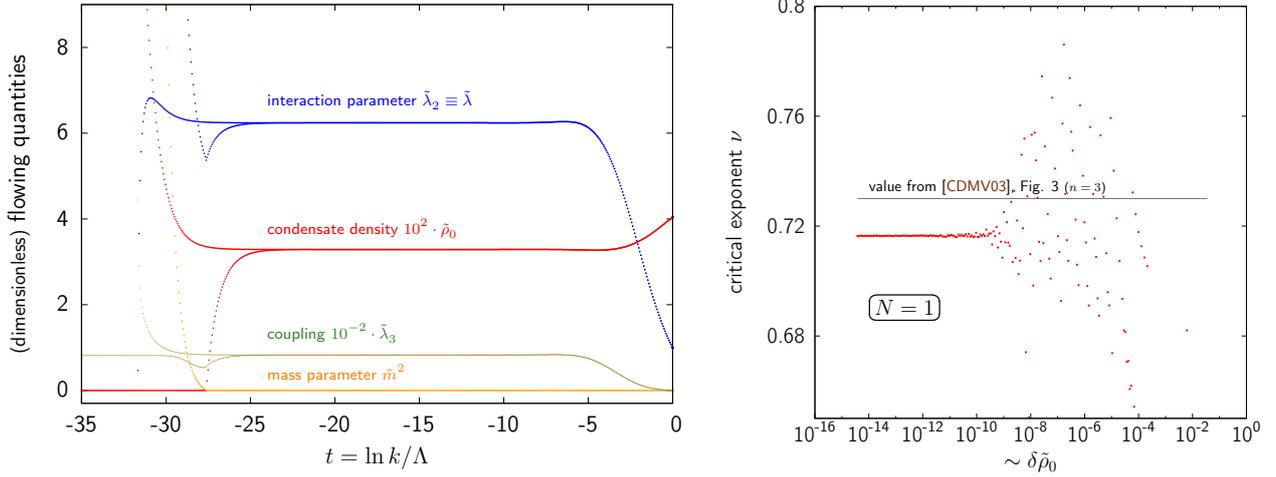


Figure 4.11: *Critical exponent ν for truncation with cubic potential in $\tilde{\rho}_0$.* We show the numerical results for the Ising model ($N = 1$) in $d = 3$ dimensions at *local potential approximation* (anomalous dimension $\eta = 0$). The data are based on an extended version of eqs. (4.28) to (4.30) where we include a coupling $\tilde{\lambda}_3$ for the cubic term in u_k . The equations read:

$$\dot{\tilde{m}}^2 = \begin{cases} -2\tilde{m}^2 - \frac{1}{2\pi^2} \frac{\tilde{\lambda}_2}{(1+\tilde{m}^2)^2} & \text{(SYM)} \\ 0 & \text{(SSB)} \end{cases}, \quad \dot{\tilde{\lambda}}_3 \equiv \frac{1}{\pi^2} \begin{cases} \frac{15}{(1+\tilde{m}^2)^3} \tilde{\lambda}_2 \tilde{\lambda}_3 - \frac{27}{(1+\tilde{m}^2)^4} \tilde{\lambda}_2^3 & \text{(SYM)} \\ \frac{15\tilde{\lambda}_2 \tilde{\lambda}_3 + 10\tilde{\rho}_0 \tilde{\lambda}_3^2}{(1+2\tilde{\rho}_0 \tilde{\lambda}_2)^3} - \frac{(3\tilde{\lambda}_2 + 2\tilde{\rho}_0 \tilde{\lambda}_3)^3}{(1+2\tilde{\rho}_0 \tilde{\lambda}_2)^4} & \text{(SSB)} \end{cases}$$

$$\dot{\tilde{\rho}}_0 = \begin{cases} 0 & \text{(SYM)} \\ -\tilde{\rho}_0 + \frac{1}{6\pi^2} \frac{3+2\tilde{\rho}_0 \tilde{\lambda}_3 / \tilde{\lambda}_2}{(1+2\tilde{\rho}_0 \tilde{\lambda}_2)^2} & \text{(SSB)} \end{cases}, \quad \dot{\tilde{\lambda}}_2 \equiv -\tilde{\lambda}_2 + \frac{1}{6\pi^2} \begin{cases} \frac{15\tilde{\lambda}_2^2}{(1+\tilde{m}^2)^3} - \frac{5\tilde{\lambda}_3}{(1+\tilde{m}^2)^2} & \text{(SYM)} \\ 2 \frac{(3\tilde{\lambda}_2 + 2\tilde{\rho}_0 \tilde{\lambda}_3)^2}{(1+2\tilde{\rho}_0 \tilde{\lambda}_2)^3} - 2\tilde{\lambda}_3 \frac{\tilde{\rho}_0 \tilde{\lambda}_3 / \tilde{\lambda}_2 - 1}{(1+2\tilde{\rho}_0 \tilde{\lambda}_2)^2} & \text{(SSB)} \end{cases}$$

The plot to the left shows flows of all quantities tuned slightly below and above the critical surface. We now reiterate the analysis of fig. 4.10 in order to compare the numerical outcome obtained with the aid of `libfrg` to the literature. The plot to the right draws a horizontal line. We deviate by about -1.8% from reference value in [CDMV03].

Conclusion & Perspectives

The present thesis has been devoted to establishing a numerical framework to solve physical problems by means of the method of functional renormalization. The developed code, organized as the library `libfrg`, provides a flexible and extendable tool to approach e.g. non-relativistic quantum statistical systems. It meets high performance computing standards in order to incorporate massive parallelization.

Before tackling issues on the practical implementation of a suitable numerical setup, we introduced the theoretical foundation of functional renormalization in [chapter 1](#). Apart from the canonical approach we addressed questions in particularly relevant for our numerical treatment. Taking the example of a zero-dimensional theory we examined the basis independence of projecting flow equations. Moreover, we argued for a discretization procedure of objects as the effective potential and the (inverse) propagator in favor of the common method of Taylor expanding around a preferred reference state/value. Especially first order phase transitions become accessible by a discretized effective potential. Hence, `libfrg` provides a new and flexible scheme to analyze the quality of truncations for functional renormalization.

[Chapter 2](#) dealt with computing a set of flow equations that are capable to render the physics of a two-component gas of interacting fermions with focus on discretizing the (inverse) propagator's momentum structure. We developed convenient matrix notation to handle the anti-commuting character of fermionic fields ψ . Furthermore we substantiated the graphical representation of the flow equation. In particular, we set the diagrammatics in the spontaneously broken phase on solid grounds.

With a view to future application we derived corresponding equations for the case of *spin imbalance* ([appendix D](#)). Since recent high precision experiments [\[NNCS10,KSCZ12\]](#) investigated the thermodynamics of the *unitary Fermi gas* it is one of our long-term perspectives to study this system with the aid of the newly developed numerics which opens the scene for improvements of quantities as e.g. the critical temperature or the *Bertsch parameter* whose theoretical value needs to be updated concerning the outcome of computations based on functional renormalization [\[DFG⁺10,BDS11\]](#). In fact, this discrepancy initiated our effort to realize a flexible library for numerics with functional renormalization.

[Chapter 3](#) intended to comment on the numerical library `libfrg` which is implemented by the paradigm of object oriented programming with C++. We succeeded in developing a flexible library that offers physicists in the field of functional renormalization a tool that is easily adaptable and extendable to

their needs. Our licensing policy ensures the freedom to use, modify and redistribute the code. Binding to proprietary software is avoided in order to guarantee independence from licensing fees. Due to the embedded high-quality (code to comment ratio: 2/1) documentation by *doxygen*, shared development by e.g. *Git* becomes practically available. A stringent treatment of logging and error messages helps users to identify runtime issues. Moreover, there is the option to remotely check the progress of the flow by functions that write back the current state of the flowing quantities to EPS-graphics files produced by *gnuplot*.

Parallel computing has been embedded to meet requirements for scientific computing on high performance clusters. The hybrid ansatz that incorporates (*Open*)*MPI* as well as *OpenMP* allows to fit the numerics to the computing grid's topology. We argued for this paradigm on the basis of time complexity and network bandwidth. Pushing this ansatz to its extreme, virtual private networks allow for connecting heterogeneous computing resources over the internet to communicate via *MPI*. If *MPI* becomes available for operating systems like *Android* or *iOS* clustering cell phones comes into reach.

Relating to the theoretical achievements from [chapter 2](#) we implemented the technical components to advance the flow given by the equations of functional renormalization. We did incorporate spatial integration routines for dimensions $0 \leq d \leq 4$ for the case of breaking the rotational symmetry by an external momentum which singles out a specific direction. A theoretical formula for arbitrary d has been derived which exploits native operations of C++ for fast evaluation. In order to correctly determine Matsubara summations we provide a semi-analytic method that enables a stable computation for $T \rightarrow 0$ while keeping the number of discrete frequencies to be evaluated fix. To capture the momentum dependence of non-relativistic (inverse) propagators we establish a two-dimensional grid in Matsubara frequency and absolute value of momentum. It is encapsulated by a class with corresponding infrastructure to intelligently manage the update of individual grid points at low computational cost. Similarly there exists a class for taking the effective potential to either a one-dimensional grid based on the Chebyshev approximation or to approximate it by Taylor expansions. To this end we implemented automatic switching from SYM to SSB and vice versa. Since the regulator is a fundamental object of functional renormalization we spent a careful survey summarizing possible variants for bosons and fermions. In the bosonic case we developed an *algebraic* regulator that generalizes the Litim and sharp cutoff. In addition we discussed a numerical instability for the implementation of the so called *exponential* regulator.

Having established `libfrg` opens the door for serious numerical studies, since there is access to each abstraction layer of the code with full control of technical parameters. `libfrg` provides the opportunity to become a standard tool as it exists for other techniques as *Quantum Monte Carlo*. Although the major challenge for reasonable results requires a sensible truncation which accurately captures the underlying physics, functional renormalization is not plagued by the *sign problem*—a severe challenge for the *Quantum Monte Carlo* community. The nearly straightforward extension from *vacuum* ($T = 0$, vanishing particle density) to many-body, finite temperature physics with the aid of functional renormalization provides a promising perspective to investigate various challenging systems in condensed matter; among them, the unitary Fermi gas.

Chapter 4 finally addressed a first benchmark of `libfrg`. To this end we considerably reduced the complexity of the theory derived in chapter 2 to a purely bosonic system with relativistic dispersion. In other words: We studied the critical physics of $O(2)$ symmetric models. By setting up the most simplest truncation we investigated critical exponents of the theory. But before turning to the physical aspects we carefully checked the flow's dependence on technical parameters that specify the numerical precision. Excellent accuracy has been confirmed.

From a conceptual point of view we stick to dimensionful equations rather than to employ the canonical approach of advancing the scaling form of the flow equation, first. As a result we discuss the limitation of this approach which supports the scaling form for determining critical exponents. Indeed, the most simplest truncation in combination with the dimensionful flow is not capable of going beyond mean field results. Nevertheless, the qualitative fixed point structure is partly observed.

In order to support the quantitative accuracy of the numerics provided by `libfrg` we recomputed critical exponents of the $O(N)$ model from the dimensionless version of the flow equations. This benchmark yielded values deviating on the sub-percentage level compared to analytical results based on the analysis of the *stability matrix* at the non-trivial fixed point. Values found in the literature that compute critical exponents in $d = 3$ spatial dimensions under the assumption of a negligible anomalous dimension deviate less than 2% from our findings. In the limit $N \rightarrow \infty$ we confirmed $\nu \rightarrow 1$ and $\beta \rightarrow \frac{1}{2}$ as transparent from an inspection of the stability matrix.

The near future calls for more explicitly testing the features of `libfrg`. Is it possible to improve the results concerning the critical exponents by including a wave-function renormalization factor extracted from the flowing momentum dependence of the inverse propagator? Is the flow of the discretized effective potential comparable to the Taylor expansion setup? ...

With the scientific progress put forward by this thesis we have a powerful tool at hand to contribute to various branches of physics which are formulated by a corresponding path integral. As stated above our focus is on non-relativistic physics, in particular condensed matter systems whose accessibility has been boosted by numerous experiments with cold atoms^[KZ08] during the past decade after the first realization of a Bose–Einstein condensate. A subject of particular interest is the equation of state and related quantities. In fact, first computations with functional renormalization have been recently published, see e.g. [RD12b, RD12a]. The straightforward inclusion of finite temperature and finite density within the framework of functional renormalization promises rich scenarios of application. The fact that we implemented an advanced routine that computes Matsubara sums enables us to go beyond (limited) analytical calculations.

The momentum dependence of (inverse) propagators becomes important when the anomalous dimension goes beyond a small perturbation to the canonical scaling. In two spatial dimensions this non-perturbative character has been reported in e.g. [GW01] for $O(N)$ models and [MBvS12] for a simplified model that approximates graphene. Hence a detailed analysis of the full momentum structure provides a challenging issue to investigate the underlying physics. Just recently, an approximation scheme that aims at resolving the momentum dependence of the propagator, the *Blaizot–Mendez–Wschebor approximation*^[BMGW06], has been successfully applied^[BBC⁺12] to the $O(N)$ model in $d = 3$. In order to

confirm the quality of the approximation, it is highly desirable to compete with our *exact* numerics.

The initial release of `libfrg` intends to foster joint effort on further developing this framework by the community of researchers putting forward functional renormalization. It serves as a basis to become a standard tool for high performance computing as it exists for techniques as e.g. *Quantum Monte Carlo* simulations or code to study e.g. fluid dynamics and classical many-body mechanics in Astrophysics & Cosmology.

Remarks on the Fourier transform of the Flow Equation

If one practically starts to apply the machinery of projecting the flow equation, eq. (1.59), for the first time, perhaps one struggles with switching between different sets of bases. In particular, one frequently encounters the *tempo-spatial* basis $x = (\tau, \mathbf{x})$ versus the frequency-momentum $q = (q_0, \mathbf{q})$ one. The following remarks intends to introduce a systematic approach to the subject.

Recalling footnote 23 from chapter 1, the combination

$$\Gamma_k^{(n)}[\eta_0] \underbrace{(\eta - \eta_0) \dots (\eta - \eta_0)}_{n \text{ times}} \quad (\text{A.1})$$

is invariant under linear, unitary transformation of the bosonic or fermionic fields

$$\eta \rightarrow \eta' \equiv \mathcal{U}\eta \quad , \quad (\text{A.2})$$

but $\Gamma_k^{(n)}[\eta_0]$ itself clearly represents a basis dependent quantity. We adopt the basis independent notation introduced by eq. (1.69) in section 1.2, here. Note, however, that physical observables need to be represented by basis independent statements. E.g. the system's *dispersion* (particle's energy-momentum dependence) is defined by $\det \Gamma^{(2)}[\eta_0] = 0$.

Let us (naturally) assume that \mathcal{U} is *scale invariant*, i.e. not depending on the flow parameter k . Then, the transformation

$$\Gamma_k^{(n)}[\eta] \rightarrow \underbrace{\mathcal{U}^\dagger \dots \mathcal{U}^\dagger}_{n \text{ times}} \Gamma_k^{(n)}[\eta] \quad \text{with} \quad \mathcal{U}\mathcal{U}^\dagger = 1 \quad (\text{A.3})$$

applies to $\dot{\Gamma}_k^{(n)}$ as well. Therefore we are allowed to determine the projection of the flow, eq. (1.59), in the basis of our choice. Converting to a another one is merely a matter of applying matrix algebra with the correct \mathcal{U} .

To be more specific: Consider the Cartesian product of all discrete and continuous field indices,

$$\mathcal{F} = \mathcal{I} \otimes \mathcal{S} \quad . \quad (\text{A.4})$$

For us, we split the *multi-index* from \mathcal{F} into an *internal* (discrete) field index and the (continuous) *space-time related* label, e.g. x or q correspond to \mathcal{S} and a discrete field index $a \in \mathcal{I}$ denotes the real or

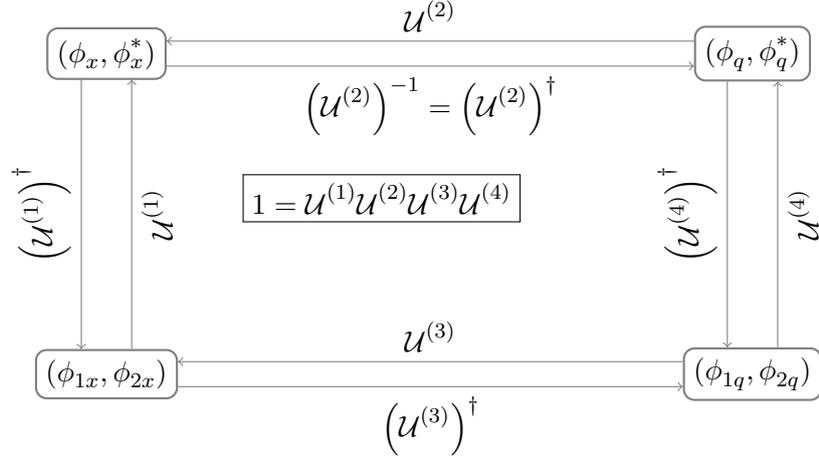


Figure A.1: *Scheme to relate different bases to each other.* Unitary transformation $\mathcal{U}^{(1\dots 4)}$ relate the components of $\eta^{(1\dots 4)}$ from eq. (A.5) to each other. Three of the four relations might be arbitrarily specified as e.g. defined by eqs. (A.6) to (A.8). However the remaining *connection* is uniquely determined by $1 = \prod_{n=1}^4 \mathcal{U}^{(n)}$. We explicitly compute it in eqs. (A.9) to (A.12). If we compute $\Gamma^{(n)}$ in some basis it is then a matter of algebra to transform it to another basis. If the transformation of the fields is governed by \mathcal{U} , $\Gamma^{(n)}$ transforms accordingly, check eq. (A.3).

complex ($a = 2n$) and imaginary or complex conjugate ($a = 2n + 1$) component of the n complex valued fields¹ $\eta_{\bar{a}}$.

We write the Fourier transform of a given field η_a introduced by eq. (2.75) in the spirit of eq. (2.93). For definiteness let us run through the following program by selecting the four bases

$$\begin{aligned}
 \eta^{(1)} : & \quad (\phi_x, \phi_x^*) && \text{complex-conjugated complex space-time basis} \\
 \eta^{(2)} : & \quad (\phi_{1x}, \phi_{2x}) && \text{real-imaginary space-time basis} \\
 \eta^{(3)} : & \quad (\phi_q, \phi_q^*) && \text{complex-conjugated complex frequency-momentum basis} \\
 \eta^{(4)} : & \quad (\phi_{1q}, \phi_{2q}) && \text{real-imaginary frequency-momentum basis} \quad . \quad (\text{A.5})
 \end{aligned}$$

of a bosonic field ϕ . It is now up to us how to define the Fourier transform that relates $\eta^{(1)} \leftrightarrow \eta^{(3)}$ and $\eta^{(2)} \leftrightarrow \eta^{(4)}$, respectively. Moreover, we are free to define the transformation $\eta^{(1)} \leftrightarrow \eta^{(2)}$ which implies $\eta^{(3)} \leftrightarrow \eta^{(4)}$. fig. A.1 aims at illustrating the defined relations. The $\mathcal{U}^{(n)}$ are calculated below.

To explicitly calculate the transformation \mathcal{U} let us define the following (unitary) transformation (summation/integration on repeated indices is understood):

$$\phi_x = \mathbf{U}_{xq} \phi_q \quad \text{and} \quad \phi_x^* = \mathbf{U}_{xq}^* \phi_q^* \quad \text{with}^2 \quad \mathbf{U}_{xq} = \mathbf{U}_{qx} = \frac{e^{iq \cdot x}}{\sqrt{2\pi^D}} \quad \text{and} \quad \mathbf{U}_{xq}^* = \mathbf{U}_{qx}^* = \frac{e^{-iq \cdot x}}{\sqrt{2\pi^D}} \quad , \quad (\text{A.6})$$

$$\phi_{1x} = \mathbf{U}_{xq} \phi_{1q} \quad \text{and} \quad \phi_{2x} = \mathbf{U}_{xq} \phi_{2q} \quad \text{as well as} \quad (\text{A.7})$$

$$\phi_x = \frac{1}{\sqrt{2}}(\phi_{1x} + i\phi_{2x}) \quad \text{and} \quad \phi_x^* = \frac{1}{\sqrt{2}}(\phi_{1x} - i\phi_{2x}) \quad . \quad (\text{A.8})$$

¹The *bar*-notation is introduced in footnote 3 of chapter 2.

There is an aspect quite crucial to grasp the intention of this appendix: Here, the symbol $*$ is not necessarily associated with the operation of complex conjugation. It just labels two components of the complex field. Nevertheless we defined the transformation eqs. (A.6) and (A.8) such that the *familiar* relations become recovered. But as we will see in a minute, the transformation $\eta^{(3)} \leftrightarrow \eta^{(4)}$ is slightly different from what one would naively expect. The message to be taken away: Happily define as it comes to your mind as long as possible; the rest needs to be determined consistently.

However, while eqs. (A.6) and (A.7) affect indices from \mathcal{S} , eq. (A.8) refers to a transformation of the indices of \mathcal{I} . With the aid of these equations it is straightforward to define corresponding unitary operators acting on \mathcal{F} :

$$\eta_\alpha^{(1)} = \eta_{(a,x)}^{(1)} = \mathcal{U}_{\alpha\alpha}^{(1)} \eta_{\alpha'}^{(2)} = \mathcal{U}_{(a,x)(a',x')}^{(1)} \eta_{(a',x')}^{(2)} \quad \text{with} \quad \mathcal{U}_{(a,x)(a',x')}^{(1)} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & i \\ 1 & -i \end{pmatrix}_{aa'} \delta_{xx'} \quad (\text{A.9})$$

$$\eta_\alpha^{(1)} = \mathcal{U}_{\alpha\alpha}^{(2)} \eta_{\alpha'}^{(3)} = \mathcal{U}_{(a,x)(a',q')}^{(2)} \eta_{(a',q')}^{(3)} \quad \text{with} \quad \mathcal{U}_{(a,x)(a',q')}^{(2)} = \frac{1}{\sqrt{2\pi^D}} \begin{pmatrix} e^{iq \cdot x} & 0 \\ 0 & e^{-iq \cdot x} \end{pmatrix}_{aa'} \quad (\text{A.10})$$

$$\eta_\alpha^{(2)} = \eta_{(a,x)}^{(2)} = \mathcal{U}_{\alpha\alpha}^{(3)} \eta_{\alpha'}^{(4)} = \mathcal{U}_{(a,x)(a',q')}^{(3)} \eta_{(a',q')}^{(4)} \quad \text{with} \quad \mathcal{U}_{(a,x)(a',q')}^{(3)} = \delta_{aa'} e^{iq \cdot x} / \sqrt{2\pi^D} \quad (\text{A.11})$$

$$\eta_\alpha^{(4)} = \eta_{(a,q)}^{(4)} = \mathcal{U}_{\alpha\alpha}^{(4)} \eta_{\alpha'}^{(3)} = \mathcal{U}_{(a,q)(a',q')}^{(4)} \eta_{(a',q')}^{(3)} \quad \text{where} \quad \mathcal{U}_{(a,q)(a',q')}^{(4)} \quad \text{is equal to}$$

$$\left(\mathcal{U}^{(3)}\right)_{(a,q)(a'',x)}^\dagger \left(\mathcal{U}^{(1)}\right)_{(a'',x)(a''',x'')}^\dagger \mathcal{U}_{(a''',x'')(a',q')}^{(2)} = \frac{1}{\sqrt{2}} \begin{pmatrix} \delta_{qq'} & \delta_{-qq'} \\ -i\delta_{qq'} & i\delta_{-qq'} \end{pmatrix}_{aa'} \quad . \quad (\text{A.12})$$

The multi-index $\alpha \in \mathcal{F}$ is split according to eq. (A.4), $\alpha = (a, x)$ ($\alpha = (a, q)$). Consequently, we obtain

$$\phi_{1q} = \frac{1}{\sqrt{2}}(\phi_q + i\phi_{-q}^*) \quad \text{and} \quad \phi_{2q} = \frac{1}{\sqrt{2}i}(\phi_q - i\phi_{-q}^*) \quad , \quad (\text{A.13})$$

or equivalently

$$\phi_q = \frac{1}{\sqrt{2}}(\phi_{1q} + i\phi_{2q}) \quad \text{and} \quad \phi_{-q}^* = \frac{1}{\sqrt{2}}(\phi_{1q} - i\phi_{2q}) \quad . \quad (\text{A.14})$$

As a demonstration let us recompute the contribution of the effective potential $U_k(\rho)$ to $\Gamma_k^{(2)}$. In section 2.2.1 we employed $\eta^{(3)}$ for the bosonic degrees of freedom. The same analysis from the perspective of $\eta^{(1)}$ becomes³

$$U_{k,(a,x)(a',y)}^{(2)} = \begin{pmatrix} \frac{\delta^2}{\delta\phi_x \delta\phi_y} & \frac{\delta^2}{\delta\phi_x \delta\phi_y^*} \\ \frac{\delta^2}{\delta\phi_x^* \delta\phi_y} & \frac{\delta^2}{\delta\phi_x^* \delta\phi_y^*} \end{pmatrix}_{aa'} \int_{x'} U_k(\rho_{x'} = \phi_{x'}^* \phi_{x'}) = \begin{pmatrix} U_k''(\rho_x) \phi_x^2 & U_k'(\rho_x) + U_k''(\rho_x) \rho_x \\ U_k'(\rho_x) + U_k''(\rho_x) \rho_x & U_k''(\rho_x) \phi_x^2 \end{pmatrix}_{aa'} \delta_{xy} \quad . \quad (\text{A.15})$$

²Compared to eq. (2.73) we equally distribute the factor $(2\pi)^{-D}$ among a function and its Fourier transform, here. Therefore these factors become absorbed into \mathbb{U} such that $\mathbb{U}\mathbb{U}^\dagger = 1$ corresponds to $U_{qx}U_{xq}^* = \delta_{qq'} \equiv \delta(q - q') = \frac{1}{2\pi} \int d^D x \exp[x \cdot (q - q')]$.

³Although it should be clear from the context, please note the following ambiguity in notation: While \mathcal{U} and \mathbb{U} refer to unitary transformation, U is the symbol for the effective potential. Moreover: U' , U'' , ... indicate the 1st, 2nd, ... derivative of the effective potential with respect to ρ . In contrast, primed indices like a' , a'' , ... are just different labels.

Inserting the constant field $\phi = \phi^* = \phi_0 = \text{const.} \in \mathbb{R}$ as well as applying the transformation $\mathcal{U}^{(2)}$ to momentum space yields

$$\begin{aligned}
U_{k,(a,q)(a',q')}^{(2)} \Big|_{\phi_0} &= \mathcal{U}_{(a,q)(a'',x)}^{(2)} \mathcal{U}_{(a',q')(a''',y)}^{(2)} U_{k,(a'',x)(a''',y)}^{(2)} \Big|_{\phi_0} \\
&= \begin{pmatrix} \mathfrak{U}_{qx} & 0 \\ 0 & \mathfrak{U}_{qx}^* \end{pmatrix}_{aa''} \begin{pmatrix} U_k''(\rho_0)\rho_0 & U_k'(\rho_0)+U_k''(\rho_0)\rho_0 \\ U_k'(\rho_0)+U_k''(\rho_0)\rho_0 & U_k''(\rho_0)\rho_0 \end{pmatrix}_{a''a'''} \begin{pmatrix} \mathfrak{U}_{q'y} & 0 \\ 0 & \mathfrak{U}_{q'y}^* \end{pmatrix}_{a'''a'} \\
&= \begin{pmatrix} U_k''(\rho_0)\rho_0 & \delta_{-qq'} & U_k'(\rho_0)+U_k''(\rho_0)\rho_0 & \delta_{qq'} \\ U_k'(\rho_0)+U_k''(\rho_0)\rho_0 & \delta_{qq'} & U_k''(\rho_0)\rho_0 & \delta_{-qq'} \end{pmatrix}_{aa'} \tag{A.16}
\end{aligned}$$

This derivation is more transparent/stringent than the calculation performed in [section 2.2.1](#), [eqs. \(2.109\)](#) and [\(2.110\)](#). In particular [eq. \(A.16\)](#) illuminates the rather counter-intuitive definition, [eq. \(2.108\)](#): Roughly speaking, it allows to pull $\delta_{-qq'}$ and $\delta_{qq'}$ out of the matrix $(\dots)_{aa'}$.

Choice of Units

This additional notes are intended to justify the basic notion of a *physical quantity* O as far as we understand it. We introduce the choice of units used for this thesis. It should help to convert them to other systems of units.

In contrast to mathematics, physical variables/observables/quantities are generally written as $O = \{O\}[[O]]$ with (mathematical) numerical value $\{O\}$ and *physical unit*¹ $[[O]]$. The former one originates from the abstract concepts of mathematics which are *just given* by definition². The latter one is closely related to a physicist's intention to build a bridge from mathematical concepts to the laws of nature in order to exploit achievements and insights from mathematics. Contact is made by measurements on properties of the object(s) of study. This process results in assigning the observable O to the object's property. Axioms are introduced as a set of equations $A_1(O_1, O_2, \dots) = 0, A_2(O_1, O_2, \dots) = 0, \dots$ which establish given relations among different observables. *Physical insight* is gained by e.g. identifying redundant axioms in order to remove them—the driving force behind *grand unified theories* (GUT). Another purpose of physical studies to understand nature is to solve the set $A_i(O_j)$ for given setups. Plotting an observable depending on a parametrization that specifies these setups results in *phase diagrams*. Note that the $A_i(O_j) = 0$ do not refer to the (artificial) decomposition $O = \{O\}[[O]]$. As long as one sticks juggling the formulae, units do not play a role.

However, in order to enable different observers to compare the outcome $o_{1/2}$ of their measurement³ of O in the sense: $o_1 \stackrel{?}{=} o_2$, we need to specify the equipment used. It is encoded in $[[O]]$. If $[[o_1]] = [[o_2]]$ then $o_1 = o_2$ if the numerical values $\{o_{1/2}\}$ coincide. To become specific: If someone tells you that the length of his arm is 2 then you need to know the procedure that assigned this pure number $\{o\}$ to the property

¹We reserve $[O]$ for dimensional analysis, see footnote 5 of chapter 1.

²We do not care about a *deeper meaning* here in order to avoid diving into philosophical questions that we feel impotent to answer.

³In addition one needs a law that specifies a transformation rule for O to switch from one *frame of reference* to another where the measurement takes place. Here we are concerned with different measurements within a given reference frame or *frame invariant* O . A prominent (counter-intuitive) example is the notion of *simultaneity* in special relativity where a unique *event* $x = (t, \mathbf{x})$ might yield different coordinates t_1 and t_2 when measured by two observers moving relative to each other.

length of arm. If he took a specific rod that exactly fits twice the length of his arm, the measurement procedure consists of: *How many units of this rod have to be laid out after another to match the length of the arm*. With this information at hand you can perform the same measurement with your arm as well; it becomes sensible to directly compare both numbers $\{o_1\}$ and $\{o_2\}$ to each other. E.g. if you get a length of 1 you might state that your arm is half as long as his one. But, if you would have chosen a rod that fits half the length of the original rod you could conclude that both of your arms are of same length. A dangerous conclusion when it comes to brawl on which rod to use as a standard. After having released from hospital you might have realized that the fight for your rod as *fundamental unit* was not worth the effort. Both are related by a simple rescaling $[[o_1]] = 2[[o_2]]$ —roughly speaking, that is the story about *cm*, *m*, *km*, etc.. Introducing the original rod as a reference and denoting its length as unit *rod*, length's of arms become represented by the (physical) observable “ $\{O\} \cdot rod$ ”.

Imagine, your girlfriend told you about her arm's length as being $5s$, five seconds. Again, you need to know the procedure of measurement that led to her result. Without, you have little chance to understand or even compare this outcome o_3 to the previous ones, $o_{1/2}$. Before returning to your desk to forward physics, you ask about her approach after lunch and she replies: *Well, it is the time my scorpion typically needs to travel along the length of my arm*⁴. Since you do not intend to check in for a stay in hospital again, you ask her for the (constant) speed c at which her beloved pet typically tends to travel. Upon the very definition of speed as well as your girlfriend's procedure of measuring arm lengths, you obtain the following transformation rule: $[[o_{1/2}]] = c[[o_3]]$ where $c = \{c\} \frac{rod}{s}$. What happened? We simply applied the law of uniform motion, $s(t) = vt$, from Newtonian mechanics to express a single property in terms of different units. This is exactly what one does when switching from one *system of units* to another one.

Historically, a bunch of different measuring devices have been introduced. Hence there exist various systems of units. As our previous example tried to point out, one might relate different units to each other. Analogous to the aim of GUTs the quest for a *minimal* set of units of *fundamental importance* comes to mind. To this end there are different opinions among physicists we do not want to touch. An entertaining paper on that subject provides [DOV01] including various standard references concerning the history of physical units. It is written by three authors fighting for different points of view.

Irrespective of such rather philosophical aspects, several fields of physics just do introduce new properties of objects to measure along with some characteristic constants. Take [table B.1](#) as a (incomplete) list that comes to our mind. All written constants are specific values of observables. In an attempt to define *universal* units one might like to express observables of a physical system in terms of units *widely available*. Back to our example: Since your girlfriend's scorpion is quite *special* in the sense that a scientist crosstown does not have access to it, one should prefer to c as being e.g. the speed of light which is the maximal speed for propagation of information in space anywhere.

So called *natural units* express all quantities of a physical system in terms of a set of system intrinsic,

⁴Yes, it seems far from being a precise method to measure arm's lengths, but your girlfriend is pragmatic enough not to build up a setting that employs the speed of light.

⁶Classical calculation with quantization of the absolute value of angular momentum by $n\hbar$, $n \in \mathbb{N}$, see *Bohr model* from standard undergraduate textbooks on quantum mechanics like [Sha94a].

branch of physics	observable O	example for characteristic constant	SI units ^[BIP06,MTN12]
Newtonian mechanics	length, time, mass	G ... Gravitational constant, proportionality coefficient that relates the force between two objects to their masses and distance	$\approx 6.67 \cdot 10^{-11} N(m/kg)^2$
electrodynamics	electric charge	e ... charge of the electron	$\approx 1.60 \cdot 10^{-19} As$
thermodynamics	temperature	k_B ... Boltzmann constant, relates temperature to the mean (kinetic) energy carried by the thermodynamic system's particles	$\approx 8.62 \cdot 10^{-5} eV/K$
special relativity	—	c ... (reference frame independent) constant speed of light	$=^5 299792458 m/s$
quantum mechanics	—	\hbar ... Planck constant, fundamental quantum of action	$\approx 6.58 \cdot 10^{-16} eVs$
atomic physics	—	a_0 ... Bohr radius, smallest (semi-classical) ⁶ radius of the electron's orbit in the hydrogen atom	$\approx 5.29 \cdot 10^{-11} m$
		m_e ... electron's rest mass	$\approx 9.11 \cdot 10^{-31} kg$
		α ... fine structure constant, electron's (semi-classical) velocity on orbit with Bohr radius a_0 in units of c	$\approx 1/137$

Table B.1: *Some characteristic constants of several branches of physics.* Part of them are used in table B.2.

characteristic constants/scales S_i . Technically this means: $S_i = 1 \cdot [[S_i]]$. Even more radically,

$$S_i \stackrel{!}{=} 1 \tag{B.1}$$

with the consequence that different units are allowed to be used in order to describe a single observable⁷. If the physical system under consideration has N independent units and we impose $M \leq N$ conditions of type eq. (B.1), each of the N units is allowed to be substituted by⁸ M of them. In our example we introduced c as a characteristic constant that relates the units of the observables length L and time T through $[[L]] = c \cdot [[T]]$. With $c \stackrel{!}{=} 1$ velocities v become pure numbers:

$$v = \{v\}[[v]] = \{v\}[[L]]/[[T]] = \{v\}c = \{v\}\{c\}[[c]] \stackrel{!}{=} \{v\}[[c]] \stackrel{!}{=} \{v\} \quad . \tag{B.2}$$

Now, the number $\{v\}$ measures v in units of the speed c . Moreover, the units of length and time *collapse* to a single one: $[[L]] \stackrel{!}{=} [[T]]$. In summary: The constraints eq. (B.1) might alter numerical

⁷ $\stackrel{!}{=}$ should be read as *is explicitly set to/defined as* in order to avoid notational confusion linked to the equal sign, e.g. $\hbar \approx 6.6 \cdot 10^{-34} Js = 1$ seems inappropriate to write.

⁸As stressed, the constraints eq. (B.1) need to be independent. Some more stringent treatment contains [Buc14].

value as well as units of observables. They define the transition from one system of units to another. In order to uniquely reverse eq. (B.1) it is mandatory for each $[[S_i]]$ to be independent from the remaining ones in the sense that $[[S_{i \neq j}]]$ can not be combined by multiplication and exponentiation to yield $[[S_i]]$. We label the unit system without eq. (B.1) by a and the other one by a' . Thus

$$[[O]]_a \equiv [[O]]_{a'} \prod_i [[S_i]]_a^{\alpha_i} \stackrel{!}{=} [[O]]_{a'} \cdot 1 \quad \text{with} \quad \alpha_i \in \mathbb{Q} \quad (\text{B.3})$$

and restoring O from the unit system a' reads

$$O_a = \{O\}_{a'} [[O]]_{a'} \prod_i S_{ia}^{\alpha_i} \quad . \quad (\text{B.4})$$

Said in words:

In order to obtain the observable O in unit system a , take the units $[[S_i]]_a$, appropriately combine them to match $[[O]]_a/[[O]]_{a'}$ and multiply the representation of O in unit system a' by the corresponding combination of S_i (in a').

To cut the story short, let us apply those considerations to the situation of the unitary Fermi gas. It involves quantum physics and thermodynamics/statistical physics. In addition to the characteristic constants \hbar and k_B we have the mass $2M_f$ of the bosonic particles composed from a spin-up and a spin-down fermion. According to the list from eq. (B.1) we deal with four independent observables, e.g. length, time, mass and temperature. By

$$\hbar \stackrel{!}{=} 1 \quad , \quad (\text{B.5})$$

$$k_B \stackrel{!}{=} 1 \quad , \quad \text{and} \quad (\text{B.6})$$

$$2M_f \stackrel{!}{=} 1 \quad (\text{B.7})$$

we allow their units to be represented by a single one. We denote the corresponding observable $O = O_b$ as *basic* with $[O_b] = 1$ the *power counting*. In our unit system, defined through eqs. (B.5) to (B.7) every observable has units in terms of powers of the basic unit $[[O_b]]$:

$$O = \{O\} [[O_b]]^{[O]} \quad \text{with} \quad [O] \in \mathbb{Q} \quad , \quad \text{hence} \quad [\hbar] = [k_B] = [M_f] = 0 \quad . \quad (\text{B.8})$$

A common choice in condensed matter physics is to take length; high energy physics prefers energy and momentum, respectively. We adopt momentum p :

$$[p] = 1 \quad (\text{B.9})$$

Let us comment on some of the consequences of eqs. (B.5) to (B.7). Since the (inverse) propagator $\Gamma_k^{(2)}$ is linked to the *dispersion* of the system defined by $\det \Gamma_k^{(2)} = 0$, the observables energy and momentum are of significance to us. Due to eq. (B.5) the units of momentum p and length x are inverse to each other:

$$[x] = -[p] \quad . \quad (\text{B.10})$$

An analogous relation holds for the units of energy E and time t , namely

$$[E] = -[t] \quad . \quad (\text{B.11})$$

This becomes transparent from the uncertainty relations $\hbar \sim \Delta p \Delta x$ and $\hbar \sim \Delta E \Delta t$, respectively. The connection of energy and momentum follows from the non-relativistic⁹, classical dispersion relation $E = p^2/2M$ of free particles of mass M . Application of eq. (B.7) yields

$$[E] = 2[p] + 0 \quad , \text{ therefore } [t] = -2[p] \quad . \quad (\text{B.12})$$

Concerning the dimension of fields η we investigate the (non-relativistic) standard kinetic term of the (classical) action $S[\eta] \subset \int dt d^d x \quad \eta^\dagger(\mathbf{x}, t) \frac{\Delta}{2M} \eta(\mathbf{x}, t)$. We deduce

$$[S] = [\hbar] = 0 = 2[\eta] + [t] + (d-2)[x] \quad \Rightarrow \quad [\eta] = \frac{1}{2}(2[p] + (d-2)[p]) = \frac{d}{2}[p] \quad . \quad (\text{B.13})$$

The Fourier transformed field $\tilde{\eta}_q = \frac{1}{2\pi} \int_x \eta_x e^{x \cdot q}$ satisfies

$$[\tilde{\eta}] = [\eta] - (d+2)[p] = -\frac{d+2}{2}[p] \quad . \quad (\text{B.14})$$

Last but not least, eq. (B.6) allows to associate temperature T with energy, i.e.

$$[E] = [T] = 2[p] \quad . \quad (\text{B.15})$$

Although the flow scale k does not need to be associated with a physical unit from the outset¹⁰, the regulator's choice $R_k = k^2 r(y)$ (cf. section 3.3.3) with $[r] = 0$ does so: Since R_k modifies the inverse propagator $P_k \rightarrow P_k + R_k$, we need to satisfy $[[P_k]] = [[R_k]]$. P_k is measured in units of energy, thus

$$[k] = \frac{1}{2}[R_k] = \frac{1}{2}[E] = [p] \quad . \quad (\text{B.16})$$

According to eq. (B.10), we conclude that k is proportional to inverse length scale, i.e. lowering the flow scale $k \rightarrow 0$ increases the associated length scale. Loosely speaking: *The flow interpolates from micro- to macro-physics.*

⁹In the case of relativistic physics where $E^2 = p^2 + m_0^2$ (speed of light $c \stackrel{\text{def}}{=} 1$, m_0 particle's invariant mass) we have $[E] = [p]$, in contrast to eq. (B.12). Explicitly setting $[c] = 0$ results in $[E] = [m_{(0)}]$ from $E = mc^2$. Thus masses are measured in units of energy/momentum. A dimensionless speed of light substitutes eq. (B.7) when dealing with relativistic physics. Moreover $c = 1$ requires $[x] = [t]$ which is in agreement with eqs. (B.10) and (B.11): $[x] = -[p] = -[E] = +[t]$. The first part of eq. (B.15), namely $[E] = [T]$, remains untouched.

¹⁰For the technical derivation of the flow equation k is just an arbitrary parameter.

¹¹Actually, when it comes to scales of interacting ultra-cold fermions (typically alkali atoms, e.g. ⁶Li), the *long-distant* part $V(r) = -C_6/r^6 + \dots$ of the van der Waals potential V is of importance. The van der Waals coefficient C_6 introduces an additional scale. Kinetic energy includes scales \hbar and M_f , the fermion's mass. In combination one could furnish a length scale, the *van der Waals length* $l_W = (M_f C_6 / \hbar^2)^{1/4}$. From the literature^[MB03] one obtains a typical value of C_6 for ⁶Li atoms that approximately reads 1400 in atomic units. Within those natural units, \hbar , the electron's mass m_e and its charge e as well as the Coulomb constant^[MTN12] k_C are set to 1. Hence length is measured in units of $a_0 = \hbar / \alpha m_e c$ with $\alpha = e^2 k_C / \hbar$. The atomic mass of ⁶Li is approximately six times the proton's mass m_p which itself is about 1836 times m_e . Therefore $M_f \approx 1.1 \cdot 10^4$ in atomic units and we end up with $l_W \approx 60 a_0$.

observable	non-relativistic physics		relativistic physics	
	substitution	SI unit system example	substitution	SI unit system example
momentum p	$\{p\} \cdot \Lambda$	$\approx \{p\}\{\Lambda\} \cdot 2.0 \cdot 10^{-24} \text{ kg m/s}$	$\{p\} \cdot \Lambda$	$\approx \{p\}\{\Lambda\} \cdot 5.0 \cdot 10^{-19} \text{ kg m/s}$
mass m	$\{m\} \cdot 2M_f$	$\approx \{m\} \cdot 2.0 \cdot 10^{-26} \text{ kg}$	$\{m\} \cdot \Lambda/c$	$\approx \{m\}\{\Lambda\} \cdot 1.7 \cdot 10^{-27} \text{ kg}$
length x	$\{x\} \cdot \hbar/\Lambda$	$\approx \{x\}/\{\Lambda\} \cdot 5.3 \cdot 10^{-11} \text{ m}$	$\{x\} \cdot \hbar/\Lambda$	$\approx \{x\}/\{\Lambda\} \cdot 2.1 \cdot 10^{-16} \text{ m}$
time t	$\{t\} \cdot 2M_f \cdot \hbar/\Lambda^2$	$\approx \{t\}/\{\Lambda\}^2 \cdot 5.3 \cdot 10^{-13} \text{ s}$	$\{t\} \cdot \hbar/\Lambda c$	$\approx \{t\}/\{\Lambda\} \cdot 7.0 \cdot 10^{-25} \text{ s}$
energy E	$\{E\} \cdot \Lambda^2/2M_f$	$\approx \{E\}\{\Lambda\}^2 \cdot 2.0 \cdot 10^{-22} \text{ J}$	$\{E\} \cdot \Lambda c$	$\approx \{E\}\{\Lambda\} \cdot 1.5 \cdot 10^{-10} \text{ J}$
temperature T	$\{T\} \cdot \Lambda^2/2M_f k_B$	$\approx \{T\}\{\Lambda\}^2 \cdot 14 \text{ K}$	$\{T\} \cdot \Lambda c/k_B$	$\approx \{T\}\{\Lambda\} \cdot 1.1 \cdot 10^{13} \text{ K}$

Table B.2: *Table to restore dropped constants from numerical values.* In order to translate numerical values from e.g. computer calculations to a physical unit system we need to restore the action of eqs. (B.5) to (B.7). Moreover, one needs to specify the *microscopic momentum scale* Λ whose numerical value $\{\Lambda\}$ is set by `Lambda` in `frg_std_include.hpp` of `libfrg`.

The 3rd/5th column provides an example of values which might be used in practice. In order to specify Λ we assume *atomic magnitudes* (cf. table B.1) for non-relativistic physics, i.e. the typical microscopic length scale¹¹ is given by the Bohr radius a_0 which we associate with a corresponding momentum $p_0 \equiv \hbar/a_0 = \alpha m_e c$. For relativistic physics we set $p_0 \equiv m_p c$. In both cases we have $\Lambda = \{\Lambda\} p_0$. Moreover, we take ${}^6\text{Li}$ atoms with approximated mass $M_f = 6m_p$ where the proton mass $m_p = 1.67 \cdot 10^{-27} \text{ kg}$. When fixing `Lambda=1e3` we need $\{T\} \approx 10^{-16}$ to reach the $n\text{K}$ regime for non-relativistic gases. In contrast, for relativistic physics $\{T\}\{\Lambda\} = 10^3$ characterizes temperatures corresponding to energies of the order 1TeV —the energy scale reached by the *Large Hadron Collider* at *CERN*.

Eventually we need to perform numerical calculations that operate with pure numbers, only. Therefore we have to specify a suitable momentum reference and set it to a numeric value thereafter. We select the *microscopic scale* Λ set by the *classical action* $S = S_{k=\Lambda}$. In `libfrg`, its numerical value is set by `Lambda` defined in `frg_std_include.hpp`. Moreover one might take the dimensionless *logarithmic scale parameter* t (not to be confused with time from above!)

$$t \equiv \ln k/\Lambda \quad \Rightarrow \quad \partial_t = k\partial_k, \quad [t] = [\partial_t] = 0 \quad . \quad (\text{B.17})$$

Let us finally list the substitution rules in order to restore the full physical observable O from the numbers $\{O\}$ spilled out by a computer programs based on `libfrg`. Constants \hbar , $2M_f$ and k_b have to be interpreted according to the unit system one would like to convert to. The numerical value of Λ is required to match `Lambda` and $[[\Lambda]]$ has to be taken in accordance to the unit of momentum within the unit system under consideration. Following eq. (B.4) we end up with table B.2.

From Laurent to Fourier to Chebyshev via Runge

As mentioned when discussing the practical implementation of a grid to represent the effective potential in [section 3.3.1](#), there is an appealing line of reasoning to grasp the convenient properties of real-valued polynomials. In particular, $p_n(x)$ of degree $n-1$ approximates a given function $f(x)$ under the constraint $p_n(x_i) = f(x_i)$ with $i = 0, \dots, n-1$ where the x_i constitute zeros or extrema of the so called *Chebyshev polynomials*. To shed light on this topic we encounter basics of complex analysis^[Nee97], and in particular Cauchy's theorem becomes applied. Bypassing Laurent series there is a unifying point of view with respect to Chebyshev approximation and Fourier series^[Tre13].

Since [chapter 3](#) deals with numerical issues, we comment on the idea of the *Fast Fourier Transform*^[CT65,CLW67] as well. It might be applied to efficiently compute coefficients of an expansion of f in Chebyshev polynomials. A *threat* intimately linked to the quest for a *good* approximation of a function f —or similarly, to the interpolation of $\{x_i, f_i\}$ —is *Runge's phenomenon*^[Run01]. There is an argument^[Tre13] highlighting the analogy to the physics of electrostatics.

Aside from consulting the references mentioned above as well as [PFTV92,Ran06,Haz95], we like to perform certain explicit calculations serving as a surplus of this survey. We focus on a deeper, but entertaining treatment of some terms briefly touched in [section 3.3.1](#). It is our aim to reflect part of our personal insight we gained when dealing with the issue of interpolating numerical data. For sure, we do not claim to completely render this subject! We rather pick the (relevant) part related to this thesis.

Let us start with the real-valued polynomial

$$p_n(x) = \sum_{i=0}^{n-1} a_i x^i \quad \text{under the constraint} \quad p_n(x_i) = f_i \quad \text{where} \quad i = 0, \dots, n-1 \quad . \quad (\text{C.1})$$

Naively, one might proceed to explicitly solve the corresponding set of n linear equations for the coefficients a_i . Then one would employ e.g. *Gaussian elimination*^[Haz95] which takes time $\mathcal{O}(n^3)$, in general. There is theoretical work that pushes the asymptotic complexity of matrix inversion towards the boundary $\mathcal{O}(n^2)$: One needs to evaluate all $n \times n$ values at least once. Enthusiastic readers are referred to the initiating papers [Str69] (Strassen, German mathematician) and [CW90] (Coppersmith/Winograd, IBM fellows).

However, there is a clever alternative due to Lagrange which goes under the name of *barycentric interpolation*^[BT04]. Since the polynomial

$$l(x) = \prod_{i=0}^{n-1} (x - x_i) \quad (\text{C.2})$$

vanishes at all $x = x_i$, we can construct polynomials of degree $n - 1$ in x such that

$$l_j(x) \equiv \prod_{\substack{i=0 \\ i \neq j}}^{n-1} \frac{x - x_i}{x_j - x_i} \quad \text{with} \quad l_j(x_i) \equiv \delta_{ij} \quad \text{for all} \quad i, j = 0, \dots, n-1 \quad . \quad (\text{C.3})$$

The denominator acts as a *normalization* in the sense that we obtain the Kronecker delta δ_{ij} instead of something proportional to.

Adopting the *bra-ket* notation from quantum mechanics we could define the *base vectors* $|l_j\rangle$ which *decompose the unit operator* on the discrete set of positions $|i\rangle \equiv |x_i\rangle$ according to

$$1 \equiv \sum_i |i\rangle\langle i| = \sum_j |l_j\rangle\langle l_j| \quad \text{with} \quad l_j(x_i) = \langle i|l_j\rangle \quad . \quad (\text{C.4})$$

Here, the summation indices i, j implicitly run from 0 to $n - 1$. The decomposition of 1 is to be understood as: *An arbitrary (real-valued) function $f(x)$ is defined/given only at the discrete set of points x_i :*

$$f_i \equiv \langle i|f\rangle \quad . \quad (\text{C.5})$$

Due to eq. (C.4) we get

$$|f\rangle = 1 |f\rangle = \sum_j \langle l_j|f\rangle |l_j\rangle = \sum_{i,j} \langle l_j|i\rangle \langle i|f\rangle |l_j\rangle = \sum_i f_i |l_i\rangle \quad . \quad (\text{C.6})$$

From this result, interpolation/extrapolation might be regarded as (naturally) extending the discrete set $\{f_i\}$ to $f(x)$ with $x \neq x_i$. In fact, we use that we did start with the $l_j(x)$ from eq. (C.3) defined for all $x \in \mathbb{R}$:

$$|i\rangle \rightarrow |x\rangle \quad , \quad 1 = \int dx |x\rangle\langle x| \quad \Rightarrow \quad f_i \rightarrow p_n(x) \equiv \langle x|f\rangle = \sum_{i=0}^{n-1} f_i \langle x|l_i\rangle = \sum_i f_i l_i(x) \quad . \quad (\text{C.7})$$

For sure: If we are given no more information than $\{f_i\}$, any $p_n(x)$ (or even more radically: any $f(x)$) that fulfills $p_n(x_i) = f_i$ ($f(x_i) = f_i$) might be regarded as an acceptable interpolation. Such an ambiguity is known under the term of *aliasing*. Evidence that eq. (C.7) might not be rejected as a totally insensible procedure provides the fact that

$$1 = \sum_j l_j(x) \quad \text{for any} \quad x \quad . \quad (\text{C.8})$$

Due to eq. (C.3) the sum of all $l_j(x)$ passes through 1 at the n x_i -values; moreover, it is still a polynomial of degree n . Thus it needs to be constant¹.

In order to proceed with any reasonable statement on the quality of a (polynomial) interpolation p_n through values $\{(x_i, f_i)\}$ we have to formulate some assumptions on the underlying f . An example from signal processing reads, roughly: *Given a function $f(x)$ whose Fourier transform is non-zero up to a specific frequency ν only, equidistant sampling with $x_i = \text{const.} + i/2\nu$ is able to exactly reconstruct/interpolate the original signal.* This statement, the *sampling theorem*², is discussed at the beginning of a (famous) paper on information theory, namely [Sha98]. Phrased more technically, $f(x)$ is completely determined by a finite Fourier series whose coefficients are equal to $f(x_i)$. Note that the equal spacing of the x_i is essential here. The simple but important lesson to learn: Given a function $f(x)$ that is known to be specified by a finite number of parameters a_i , the correct choice of sampling points x_i is significant to uniquely specify these a_i . In fact, we will discuss a connection relating the equidistant sampling from Fourier analysis to the sampling of Chebyshev interpolation in a moment. To this end we recap a central result from complex analysis—namely *Cauchy’s theorem*, i.e. *Stokes’ theorem* for complex analysis—and its implication for analytic functions $f(z)$ of a complex variable z which satisfy

$$\oint_{\mathcal{C}} dz f(z) = 0 \tag{C.9}$$

on a closed path \mathcal{C} . As a consequence we are allowed to *extract* f ’s value at z_0 by introducing a *simple pole* (non-analyticity) z^{-1} at that point:

$$f(z_0) = \frac{1}{2\pi i} \oint_{\mathcal{C}} dz \frac{f(z)}{z - z_0} \quad , \tag{C.10}$$

where \mathcal{C} encloses z_0 counterclockwise once. Using $\int_0^{2\pi} d\varphi e^{in\varphi} = 2\pi\delta_{0n}$ for some integer n it is relatively straightforward to extend eq. (C.10) to

$$f^{(n)}(z_0) \equiv \left. \frac{d^n f}{dz^n} \right|_{z_0} = \frac{n!}{2\pi i} \oint_{\mathcal{C}} dz f(z) (z - z_0)^{-1-n} \quad . \tag{C.11}$$

Given a function $g(z)$ that has an *isolated*³ pole of *order* n at z_0 , i.e. its behavior at that point is proportional to the divergence of z^{-n} at zero, we can set $f(z) = g(z)(z - z_0)^n$ and apply eq. (C.11) in the vicinity of z_0 ; that is, \mathcal{C} encloses the pole at z_0 only. Depending on the pole structure of $g(z)$, we might identify annuli in the complex plane where the *Laurent series*

$$\sum_{n=-\infty}^{\infty} c_n z^n \quad \text{with} \quad c_n = \frac{1}{2\pi i} \oint_{\mathcal{C}} dz f(z) z^{-1-n} \tag{C.12}$$

¹A bit more formally: Define $L(x) = -1 + \sum_{j=0}^{n-1} l_j(x)$, hence $L(x_i) = 0$. The ansatz $L(x) = \sum_{k=0}^{n-1} a_k x^k$ yields a system of n linear equations for the coefficients a_k compactly written as $M\mathbf{a} = 0$. The vector \mathbf{a} contains the coefficients a_k . Without loss of generality the matrix M_{ik} has elements $x_i^k \neq 0$ (global shift of all x_i by a constant). It follows that $\mathbf{a} = 0$, thus $L(x) = 0$.

²From the quantum mechanical point of view the *sampling theorem* is another face of *Heisenberg’s inequality/uncertainty principle* $\Delta x \Delta p \geq \hbar/2$.

³We associate “ $f(z)$ is analytic in an (open) region around z_0 ” with the notion *isolated*.

converges to f within these annuli which have to be free of poles/divergences. Stated differently: If f is analytic within the chosen annulus \mathcal{A} (and \mathcal{C} resides in \mathcal{A}), the Laurent series exactly reproduces f in \mathcal{A} . In cases where the inner circle of the annulus does not contain a singularity, \mathcal{A} can be extended to a disc and eq. (C.12) turns to the (complex) Taylor series, i.e.

$$f(z) = \sum_{n=0}^{\infty} c_n z^n = \sum_{n=0}^{\infty} \frac{1}{n!} f^{(n)}(z) \Big|_{z=0} z^n \quad . \quad (\text{C.13})$$

Assuming that f is analytic within an annulus containing $|z| = 1$, the convergence of the (complex) Fourier series

$$f(\varphi) = \sum_{n=-\infty}^{\infty} c_n e^{in\varphi} \quad \text{with} \quad c_n = \frac{1}{2\pi} \int_{-\pi}^{\pi} d\varphi f(\varphi) e^{-in\varphi} \quad (\text{C.14})$$

follows from eq. (C.12) with $z = e^{i\varphi}$. Imposing symmetry constraints as $f^* = f$ (f real valued) or (in addition) $f(\varphi) = \pm f(-\varphi)$, eq. (C.14) further reduces to well known formulae of Fourier analysis. Of practical importance is the requirement of f to be analytic. In particular all derivatives of $f(\varphi)$ need to exist for the convergence of the Fourier series. A popular counterexample being the *Gibbs phenomenon*^[Haz95] when interpolating a discontinuous function as e.g. a (periodic) unit step $f(\varphi) = \Theta(\varphi)$ with $f(\varphi + 2\pi) = f(\varphi)$.

Even for continuous, real-valued functions $f(x)$, the corresponding Taylor series do not need to converge. This observation is related to *Runge's phenomenon* which we are going to discuss below. It traces back to singularities of f when it is analytically continued^[Haz95] from $f(x)$ on the real line to $f(z)$ in the complex plane. Neither does $(1+x^2)^{-1}$ exhibit singularities nor does it contain discontinuities for $x \in \mathbb{R}$. Nevertheless $(1+z^2)^{-1}$ has simple poles at $z_0 = \pm i$. Therefore convergence of $\sum_{n=0}^{\infty} \frac{1}{n!} f^{(n)}(x) \Big|_{x=0} x^n$ for $|x| \geq 1$ is not guaranteed.

Transplanting these observations to the task of interpolating a (real-valued) function f through $\{(x_i, f_i)\}$ by a polynomial $p_n(x)$ is performed by combining the results eqs. (C.7) and (C.10). More precisely, the analytic continuation $l(z)$ of $l(x)$, eq. (C.2), serves to create simple poles at the x_i and hence⁴,

$$p_n(x) = \sum_i f_i l_i(x) = \frac{1}{2\pi i} \sum_i \oint_{\mathcal{C}_i} dz \frac{l(x)f(z)}{l(z)(x-z)} \quad , \quad (\text{C.15})$$

which is known as *Hermite's interpolation formula*. The contours \mathcal{C}_i enclose the singularity at x_i only. Due to eq. (C.9) we are allowed to substitute the \mathcal{C}_i by a single \mathcal{C} enclosing all $z = x_i$, but the term $(x-z)^{-1}$ in eq. (C.15) accounts for an additional contribution $f(x)$. Thus, we end up with a convenient equation that quantifies the difference between the interpolation p_n and f at x :

$$p_n(x) - f(x) = \frac{1}{2\pi i} \oint_{\mathcal{C}} dz \frac{l(x)f(z)}{l(z)(x-z)} \quad . \quad (\text{C.16})$$

⁴There is a representation of $l_i(x) = \prod_{j \neq i} (x - x_j)/(x_i - x_j)$ in terms of a contour integral. Consider the expression $l(x) \oint_{\mathcal{C}_i} dz / [(x-z)l(z)]$, where \mathcal{C}_i encloses $z = x_{j=i}$ only. Up to a factor of $2\pi i$ it is equivalent to $l(x) / [(x - x_i) \prod_{j \neq i} (x_i - x_j)]$.

Note, that f is assumed to be analytic in the region enclosed by \mathcal{C} excluding x, x_0, \dots, x_n . In order to inspect the dependence of the error $|p_n(x) - f(x)|$ on n it is of importance to detect that $|l(x)/l(z)|$ under the integral on the rhs. of eq. (C.16) is depending on the number of interpolation points, only. Its magnitude for $n \rightarrow \infty$ determines whether $p_n(x)$ converges to $f(x)$ or not. In order to normalize the increase/decrease of $\left| \frac{l(x)}{l(z)} \right|$ for fixed z, x due to the product $\prod_{i=0}^{n-1}$, it is sensible to investigate

$$\sqrt[n]{\left| \frac{l(x)}{l(z)} \right|} \quad \text{or equivalently its logarithm} \quad \kappa_n(x, z) = \frac{1}{n} \sum_{i=0}^{n-1} \ln |x - x_i| - \ln |z - x_i| \quad . \quad (\text{C.17})$$

Therefore, we have

$$\left| \frac{l(x)}{l(z)} \right| \equiv \exp [n\kappa_n(x, z)] \quad \text{with} \quad z \in \mathcal{C} \quad \text{and} \quad x \in \mathbb{R}, \text{ the real line in the complex plane.} \quad (\text{C.18})$$

It follows that the error $|p_n - f|$ exponentially decreases/increases for κ_n negative/positive when increasing the number of interpolation points x_i —provided that $0 < |\kappa_n(x, z)| < \kappa$ stays bounded for fixed x and all z on the (fixed) contour \mathcal{C} .

Now, physics enters the stage by defining the *potential*

$$u(z) = \lim_{n \rightarrow \infty} \frac{1}{n} \sum_{i=0}^{n-1} \ln |z - x_i| \equiv \int dx \rho(x) \ln |z - x| \quad , \quad (\text{C.19})$$

where the *charge density* on the real line satisfies

$$1 = \int dx \rho(x) \quad . \quad (\text{C.20})$$

The latter is defined by the distribution of the interpolation points $\{x_i\}$ for $n \rightarrow \infty$, e.g. for equidistant spacing $x_{i+1} - x_i = 2/n$ with $x_i \in [-1, 1]$ we get $\rho(x) = \frac{1}{2}\Theta(1 - |x|)$. The notion *charge* is motivated by electrostatics in two dimensions. Since analytic functions

$$f(z) = u(z) + iv(z) \quad \text{with} \quad z = x + iy \quad \text{where} \quad x, y, u, v \in \mathbb{R} \quad (\text{C.21})$$

need to satisfy the *Cauchy–Riemann equation*⁵

$$\nabla f = 0 \quad \text{with} \quad \nabla \equiv \partial_x + i\partial_y \quad , \quad (\text{C.22})$$

it follows the *Laplace equation*

$$\bar{\nabla} \nabla f \equiv \Delta f = 0 \quad \text{where} \quad \bar{\nabla} \equiv \partial_x - i\partial_y \quad \text{and} \quad \Delta = \partial_x^2 + \partial_y^2 \quad . \quad (\text{C.23})$$

Hence analytic functions satisfy the equation for the electrostatic potential. Excluding $z = 0$, $u(z) = \ln |z|$ fulfills $\Delta u = 0$. Adopting this spirit, we regard eq. (C.19) as the potential due to a given

⁵Stokes' theorem $\int_V d\omega = \int_{\partial V} \omega$ of the *differential form* $d\omega = d(fdz) = i\nabla f dx dy = 0$ on the *volume* V with the *boundary* $\partial V = \mathcal{C}$ yields eq. (C.9).

distribution $\rho(x)$ of infinitely many infinitesimal charges $\rho(x)dx$ located at x . Therefore, infinitely many interpolation points are characterized by

$$\kappa(x, z) \equiv \lim_{n \rightarrow \infty} \kappa_n(x, z) = u(x) - u(z) \quad . \quad (\text{C.24})$$

For

$$\kappa(x) \equiv \max_{z \in \mathcal{C}} \kappa(x, z) < 0 \quad , \quad (\text{C.25})$$

$p_n(x)$ converges exponentially to $f(x)$ at the rate $\kappa(x)$. If we consider the convergence properties of an interpolation with $a = x_0 < x_1 < \dots < x_n = b$ (\mathcal{C} encloses the real line segment $[a, b]$), eq. (C.25) must be true for all $x \in [a, b]$. Since $\kappa(x, z)$ is a potential difference, this condition is satisfied by $z = x + i0 \in [a, b]$ a global minimum of $u(z)$; in particular: $u(x)|_{x \in [a, b]} = \text{const.}$. Thus, we end up with a condition that determines $\rho(x)$.

In physical terms, the segment $[a, b]$ on the real line represents a metallic wire of total charge 1. Without loss of generality, we set $-a = b = 1$ from now on. By definition, differences of the electric potential vanish within a metal—there is no voltage that courses the charge carriers to move. Therefore we have to calculate the charge distribution for this equilibrium situation. Since the Laplace equation is invariant under *conformal mappings*⁶ $\omega(z)$, there is a technique to map the solution outside a metallic ring/surface with $|z| = 1$ to the real line segment $[-1, 1]$ which is the situation we would like to inspect. The former case is solved by the potential $u(z) = \ln|z|$; explicit calculation confirms $\Delta u = 0$, and clearly, $u(e^{i\varphi}) = \text{const.} = 0$. The unit circle $|z| = 1$ can be *stereographically* projected to the real axis by $(z + z^*)/2$. However, z^* is not analytic, but since $z = |z|e^{i\varphi}$ we are saved by $z^* \rightarrow z^{-1}$. Extending this projection to $|z| > 1$, we notice that⁷

$$\omega(z) = \frac{1}{2}(z + z^{-1}) \quad \text{maps} \quad \{z \in \mathbb{C}, |z| \leq 1\} \quad \text{to} \quad \{z \in \mathbb{C}, z \neq x \in [-1, 1]\} \quad . \quad (\text{C.26})$$

In the *far field* $\lim_{|z| \rightarrow \infty} u(z)$, the metallic wire on the real line and the unit sphere should produce the same $u(z)$, i.e. ω is required to be proportional to the *unit map* $z \mapsto z$, which is readily confirmed. Now, the potential u of the wire can be computed by inverting eq. (C.26),

$$u(z) = \ln \left| z \pm \sqrt{z^2 - 1} \right| + \text{const.} \quad (\text{C.27})$$

⁶Analytic functions ω are characterized by a well defined derivative $\omega^{(1)}(z) = d\omega/dz = |\omega^{(1)}(z)|e^{i\varphi(z)}$. Thus, the mapping $z \mapsto \omega(z)$ constitutes a (local) stretching by $|\omega^{(1)}|$ plus a rotation by φ of elements dz with $|dz| \ll 1$. This operation preserves angles.

Assuming a function $f(z) = f(\omega(z)) = f(\omega)$ analytic in z , we have $\Delta_z f = 0$ where $\nabla_z \equiv \partial_x + i\partial_y$. We analogously define $\nabla_\omega \equiv \partial_{\omega_1} + i\partial_{\omega_2}$ with $\omega = \omega_1 + i\omega_2$, $\omega_{1/2} \in \mathbb{R}$. The chain rule implies $\nabla_z = \omega^{(1)*}\nabla_\omega$ and $\bar{\nabla}_z = \omega^{(1)}\bar{\nabla}_\omega$. If ω is analytic, $f(\omega)$ is: $0 = \bar{\nabla}_z \nabla_z f(z) = |\omega^{(1)}|^2 \Delta_\omega f(\omega) + 0$, and the zero is due to the product rule from $\nabla_\omega \omega$. $d\omega/dz$ should not vanish in order for $z \mapsto \omega$ to be invertible.

⁷The magnitude $|\omega|^2 = |z|^2 + |z|^{-2} + 2 \cos 2\varphi$ takes all values larger than zero. This is confirmed by the fact that $h(x) \equiv x + x^{-1} > 2$ for $x > 1$. For sure, $h(2) = \frac{5}{2} > 2$, and $h(x)$ attains its local minimum 2 at $x = 1$ for all positive values of the argument where it is continuous. For $x \gg 1$ $h(x)$ grows as $\mathcal{O}(x)$. The (tangent of the) angle φ of z , namely y/x , becomes modified by a factor $(|z|^2 - 1)/(|z|^2 + 1)$ under the mapping $z \mapsto \omega$. For $|z| \gg 1$ this correction is nearly negligible.

which, again, is constant for $z = x = \cos \varphi \in [-1, 1]$: $\ln |e^{\pm i\varphi}| = 0$.

Here we are! This projection prescription eq. (C.26) encodes the choice of the Chebyshev interpolation points x_i . Its density $\rho(x)$ in the limit of infinitely many x_i is computed by stereographically projecting the uniform distribution from the unit sphere $|z| = 1$ to the real line segment $[-1, 1]$. The conformal mapping $\omega(z)$ helps to explicitly determine $\rho(x)$. Restricting $d\omega/dz$ to $|z| = 1$, we get $dz = dx + i0$. Then, the ratio $|dz|/dx = 1/|\omega^{(1)}|$ is proportional to the charge density of the metallic wire. We have

$$|d\omega/dz| = |1 - z^2| = |e^{i\varphi} - e^{-i\varphi}| \sim \sqrt{1 - \cos^2 \varphi} = \sqrt{1 - x^2} \quad , \quad (\text{C.28})$$

and finally

$$\rho(x) \sim 1/\sqrt{1 - x^2} \quad . \quad (\text{C.29})$$

Referring back to eq. (C.16), the contour \mathcal{C} must not contain singularities of f . Otherwise, a finite contribution adds to the error, which prevents $|p_n - f|$ to vanish for $n \rightarrow \infty$. The larger the *distance* $\max_{x \in [-1, 1], z \in \mathcal{C}} |x - z|$, the faster the convergence. Therefore, the pole structure of f affects the rate of the exponential vanishing error. Nevertheless, for interpolation points x_j projected according to eq. (C.26) from uniformly distributed $z_j = \exp(i2\pi j/n)$ on the ring $|z| = 1$, there is always a contour \mathcal{C} where f is analytic inside—provided that f has isolated singularities in the vicinity of the real line segment $[-1, 1]$ only. To this end, the *Chebyshev grid*

$$x_j = \cos[2\pi j/n + \text{const.}] \quad (\text{C.30})$$

is *optimal*. In the limit $n \rightarrow \infty$ the density eq. (C.29) is approached. Zeros or extrema of the *Chebyshev polynomials*

$$T_n(x) = \cos[n \arccos x] \quad (\text{C.31})$$

provide such interpolation points. It is this property that we focus on to numerically represent the effective potential $U_k(\rho)$.

If we would have chosen another distribution of interpolation points, the corresponding potential u will deviate from the global minimum configuration within the real line segment $[-1, 1]$. Then, in general, it is not ensured that $\kappa(x)$ from eq. (C.25) stays negative. For example: Equidistant x_j exhibit a potential $u(z)$ such that functions f which are analytic on the real line are not converged to by p_n . This is known as *Runge's phenomenon*. Numerical investigation shows⁸ that the first equipotential line $u(z) = \text{const.}$ which encloses the real line segment $[-1, 1]$ with $\rho(x) = \text{const.}$, crosses the imaginary axis at $\approx \pm 0.53$. Since $(1 + 25z^2)^{-1}$ has poles at $\pm 0.25i$, \mathcal{C} necessarily needs to enclose these singularities. Convergence of p_n to f is impossible on $[-1, 1]$, here. Note, that f is analytic on the whole real line. There is no discontinuity as in the case of *Gibbs phenomenon*.

To some extent $u(z)$ is an extension of the convergence criterion for Taylor series of a real valued function $f(x)$. While Taylor expansion incorporates data on n derivatives $f^{(i)}(x_0)$ at a given point x_0 ,

⁸Directly applying eq. (C.19) yields $u(z) \sim \Re[(z + 1) \ln(z + 1) - (z - 1) \ln(z - 1)]$ from $\partial_x \Re[z \ln z] = \partial_x [x \ln |z| - y \arctan(y/x)] = 1 + \ln |z|$.

(Chebyshev) interpolation employs n values f_i of the function at different points x_i . In the former case, the potential u produces circular equipotential lines according to $\ln|z - x_0|$. It is exactly the far field of u for the latter one or equivalently shrinking the interpolation interval to zero around x_0 .

A polynomial of degree n is unique when specified by $\{(x_i, f_i)\}$. According to the beginning of this section, eqs. (C.1) to (C.8), one might prefer a certain basis, $\{1, x, x^2, \dots\}$ or $\{l_i(x)\}$ or \dots . While Lagrange polynomials are suited to directly interpolate through a given set of points, it is rather involved to compute their derivatives. Since we need $U'_k(\rho)$, $U''_k(\rho)$, \dots for our numerics there is another type of polynomials written down in eq. (C.31). An expansion in $T_n(x)$ is nothing but another face of Fourier series for real valued functions f which are 2π -periodic with the additional symmetry $f(\varphi) = f(-\varphi)$. Equation (C.26) provides the link: $x = \cos \varphi$ and thus $f(\varphi) = f(\arccos x)$ as well as⁹ $dx = \pm \sin \varphi d\varphi = \pm \sqrt{1 - x^2} d\varphi$, then

$$f(\varphi) = \sum_{n=0}^{\infty} a_n \cos(n\varphi) + 0 \cdot \sin(n\varphi) = \sum_{n=0}^{\infty} a_n T_n(x) = f(x) \quad . \quad (\text{C.32})$$

The coefficients a_n follow from eq. (C.14) and $f = f^*$, i.e.

$$a_n = c_n + c_n^* = \frac{1}{\pi} \int_0^{2\pi} d\varphi f(\varphi) \cos n\varphi = \frac{2}{\pi} \int_{-1}^1 dx \frac{f(x) T_n(x)}{\sqrt{1 - x^2}} \quad . \quad (\text{C.33})$$

In order to confirm that $T_n(x)$ is a polynomial of degree n , we state:

$$T_0(x) = 1, \quad T_1(x) = x, \quad \text{and} \quad T_{n+1}(x) + T_{n-1}(x) = 2xT_n(x) \quad , \quad (\text{C.34})$$

which is a direct consequence of $\cos \alpha + \cos \beta = 2 \cos(\alpha + \beta)/2 \cos(\alpha - \beta)/2$.

From eq. (C.31) we deduce that $T_{n>1}(x)$ takes values in $[-1, 1]$ with n zeros at $\cos[\pi(i + \frac{1}{2})/n]$ (*Chebyshev points of the first kind*, $i = 0, \dots, n - 1$) and alternating minima/maxima ± 1 at $\cos[\pi i/n]$ (*Chebyshev points of the second kind*, $i = 0, \dots, n$). Let us address the question, if the $\{T_i(x)\}$, $i = 0, \dots, n - 1$ constitute a basis for n function values f_i evaluated at the Chebyshev grid points (of 1st/2nd kind).

To this end we have to confirm

$$\langle T_i | T_j \rangle = \sum_k \langle T_i | k \rangle \langle k | T_j \rangle = \sum_k T_i(x_k) T_j(x_k) = \text{const.} \delta_{ij} \quad , \quad (\text{C.35})$$

where k runs over all Chebyshev points of first or second kind. Indeed, $\cos \alpha \cos \beta = \frac{1}{2} [\cos(\alpha + \beta) + \cos(\alpha - \beta)]$ leads to two summations over equidistant argument values of the cosine centered around $\varphi = \pi/2$ from which $\cos(\varphi)$ is antisymmetric, i.e. $\cos(\varphi - \frac{\pi}{2}) = -\sin \varphi$. Only in the case $i = j$ the summation due to $\cos(\alpha - \beta) = 1$ does not get *averaged* to zero; it yields the constant factor $n/2$. $T_0(x) = 1$ is special in so far that the constant needs to be replaced by twice its value, $n/2 \rightarrow n$. Therefore we may write

$$1 = \frac{2}{n} \sum_{i=0}^{n-1} |T_i \rangle \langle T_i| - \frac{1}{n} |T_0 \rangle \langle T_0| \quad (\text{C.36})$$

⁹Both signs are necessary when traversing the unit circle counterclockwise. While x decreases from 1 to -1 for increasing $\varphi \in [0, \pi)$, it increases up to 1 again for $\pi \leq \varphi < 2\pi$.

in order to conclude

$$\begin{aligned}
 |f\rangle &= \frac{2}{n} \sum_i \langle T_i | f \rangle |T_i\rangle - \frac{1}{n} \langle T_0 | f \rangle |t_0\rangle \\
 &= \frac{2}{n} \sum_{i,k} \langle T_i | k \rangle \langle k | f \rangle |T_i\rangle - |T_0\rangle \frac{1}{n} \sum_k \langle T_0 | k \rangle \langle k | f \rangle \\
 &= \sum_i |T_i\rangle \left[\frac{2}{n} \sum_k f_k T_i(x_k) \right] - |T_0\rangle \frac{1}{2} \frac{2}{n} \sum_k 1 \cdot f_k \quad . \quad (C.37)
 \end{aligned}$$

Defining the expression [...] as b_i and computing $p_n(x) = \langle x | f \rangle$ we obtain the *Chebyshev interpolation*

$$p_{n-1}(x) = \sum_{i=0}^{n-1} b_i T_i(x) - \frac{b_0}{2} \quad (C.38)$$

with

$$b_i = \frac{2}{n} \sum_k f_k T_i(x_k) = \frac{2}{n} \sum_k f_k \cos(\pi \bar{k}) \quad \text{where} \quad \bar{k} \equiv \begin{cases} k + \frac{1}{2} & \text{for Cheb. grid of 1st kind.} \\ k & \text{for Cheb. grid of 2nd kind.} \end{cases} \quad (C.39)$$

Numerically, the b_i amount for evaluating a *Discrete Cosine Transform*^[ANR74]. Similar to the *Fast Fourier Transform* there is an efficient algorithm that operates in time $\mathcal{O}(n \log n)$. Given the set of coefficients $\{b_i\}$ there is a method to effectively compute $p_n(x)$ due to Clenshaw^[Cle55]. It incorporates the recurrence eq. (C.34).

When it comes to evaluate derivatives of p_n , the Chebyshev interpolation turns out to be quite handy, since there is a straightforward relation between the b_i and b'_i where we implicitly defined

$$\partial_x p_n(x) = \sum_{j=0}^n b_j \partial_x T_j(x) \equiv \sum_{j=0}^n b_j j \tilde{T}'_j \equiv \sum_{j=0}^{n-1} b'_j T_j(x) - \frac{b'_0}{2} \quad . \quad (C.40)$$

Let $T'_m \equiv m \tilde{T}'_m$ denote the derivative of T_m with respect to x . Is there a recurrence similar to eq. (C.34) that holds for the T'_m ? Written in terms of $\varphi = \arccos x$ and $T_m = \cos m\varphi$ we have $m \neq 0$

$$\tilde{T}'_m = \frac{T'_m}{m} = \frac{\sin m\varphi}{\sin \varphi} \quad , \quad \text{and thus} \quad \tilde{T}'_{m\pm 1} = x T'_m \pm T_m \quad \text{for} \quad m > 1 \quad . \quad (C.41)$$

It is verified by applying $\sin(\alpha \pm \beta) = \sin \alpha \cos \beta \pm \sin \beta \cos \alpha$. It directly follows that

$$2T_m = \tilde{T}'_{m+1} - \tilde{T}'_{m-1} \quad \text{for} \quad m \geq 0 \quad \text{with} \quad \tilde{T}'_0 \equiv 0 \quad \text{and} \quad \tilde{T}'_{-1} \equiv -1 \quad . \quad (C.42)$$

Using eq. (C.40) a quick calculation confirms

$$\begin{aligned}
 p'_{n+1} &= \sum_{j=1}^n [j b_j] \tilde{T}'_j = -\frac{b'_0}{2} + \sum_{j=0}^{n-1} b'_j T_j \\
 &= +\frac{b'_0}{2} \tilde{T}'_{-1} + \sum_{j=0}^{n-1} b'_j (\tilde{T}'_{j+1} - \tilde{T}'_{j-1})/2 \\
 &= 0 + \frac{1}{2} b'_{n-1} \tilde{T}'_n + \sum_{j=1}^{n-1} \left[\frac{1}{2} (b'_{j-1} - b'_{j+1}) \right] \tilde{T}'_j \quad . \quad (C.43)
 \end{aligned}$$

With the initial condition $b'_{n+1} = b'_n = 0$ a recursive computation of the b'_j is available from the set of $\{b_j\}$, namely

$$b'_{j-1} = b'_{j+1} + 2jb_j \quad \text{where } 0 < j \leq n \quad . \quad (\text{C.44})$$

Last but not least, let us outline the idea of the *Fast Fourier Transform* which might be used to compute the b_j from the (discrete) function values f_i . There exist various flavors adapted to the specific symmetries f may possess. For the generic complex case the aim is to compute n Fourier coefficients c_k according to

$$c_k = \sum_{j=0}^{n-1} f_j e^{-i2\pi kj/n} \quad \text{with } k = 0, \dots, n-1 \quad . \quad (\text{C.45})$$

The naive approach sums n products n times, i.e. we have computation complexity $\mathcal{O}(n^2)$. The key observation is the fact that $e^{i\varphi}$ is 2π -periodic and $\varphi/2\pi n = jk \pmod n$ takes all integer values from 0 to $n-1$ multiple times since $0 \leq jk \leq (n-1)^2$.

In particular we ask, if we have computed the Fourier coefficients c_k, d_k, \dots of n_1 functions evaluated at n_2 angles $\varphi = 2\pi j/n_2$ to f_j, g_j, \dots : Is there a way to combine those values to a new function with $n = n_1 n_2$ values h_j at $\varphi = 2\pi j/n$ such that the resulting Fourier coefficients e_k are a linear combination of the c_k, d_k, \dots ? This would enable us to compute eq. (C.45) starting with a small number of f_j that takes time $\mathcal{O}(1)$. Iterating the answer we would be able to compute each c_k for an exponentially increasing number of f_j in linear time.

Starting with

$$\underbrace{c_k = \sum_{j_0=0}^{n_2-1} f_{j_0} e^{-i2\pi k j_0/n_2}, \quad d_k = \sum_{j_1=0}^{n_2-1} g_{j_1} e^{-i2\pi k j_1/n_2}, \quad \dots}_{n_1 \text{ times}} \quad \text{to engineer} \quad e_K = \sum_{J=0}^{n-1} h_J e^{-i2\pi K J/n} \quad (\text{C.46})$$

we trivially extend the summation of c_k, d_k, \dots to $n = n_1 n_2$ by rewriting the exponent's argument $-i2\pi k j/n_2 = -i2\pi k (j n_1)/n$ with j any of the summation indices j_0, j_1, \dots . Can we unify the n_1 summations with n_2 terms for fixed k to yield e_k ? Up to now the modification of the exponent's argument do ambiguously map the n indices j_l with $l = 0, \dots, n_1-1$ to J when setting $e_k = c_k + d_k + \dots$. If we could achieve a mapping according to $J = j_l n_1 + l$ with $l = 0, \dots, n_2-1$, we would obtain a one-to-one correspondence of the j_0, j_1, \dots to J . Since c_k, d_k, \dots are n_2 periodic in k , the factor $\exp(-i2\pi K l/n)$ saves our day:

$$f_{j_0} \mapsto h_{j_0 n_1 + l}, \quad g_{j_1} \mapsto h_{j_1 n_1 + l}, \quad \dots \quad \text{and} \quad e_K = e^0 c_{K \bmod n_2} + e^{-i2\pi K/n} d_{K \bmod n_2} + \dots \quad . \quad (\text{C.47})$$

Due to the binary representation of computer algebra, $n_1 = 2$ is a common choice. Then, the final number n_{\max} of f_j is often taken to be a power of 2. However, eq. (C.47) describes a single iteration/recursion step and one may select different n_1 for each of them. Fixing n_1 , the procedure is fastest for^[CT65] $n_1 = 3$. $n_1 = 2, 4$ follow in efficiency and are preferred due to the computer's arithmetic.

Flow Equations for Interacting Fermions with Spin-Imbalance

As an overview we provide a comprehensive collection of flow equations that characterize a two-component Fermi gas. In particular it renders the physics of the *unitary Fermi gas* as well as the phase diagram of the imbalanced gas ($\mu_\downarrow \neq \mu_\uparrow$).

In a manner similar to eq. (2.67) we assume a truncation of the effective action $\Gamma_k[\eta]$ that incorporates (full) momentum resolved propagators $P_{\uparrow\downarrow,k}(q)$ of the fermionic particles $\psi_{\uparrow\downarrow}$ and a composite bosonic degree of freedom φ . The *molecule* φ couples to the fermions by a Yukawa-type interaction $\varphi\psi_\uparrow\psi_\downarrow$ with strength h_φ . As usual the *effective potential* $U_k(\rho = \phi\phi^*)$ is defined via the evaluation of Γ_k at *constant field*

$$\eta = \text{const.} \equiv \Phi \equiv (\varphi = \phi, \varphi^* = \phi^*, \psi_\uparrow = 0, \psi_\uparrow^* = 0, \psi_\downarrow = 0, \psi_\downarrow^* = 0) \quad . \quad (\text{D.1})$$

Furthermore, our truncation allows for an arbitrary shape of the function U_k .

For the sake of avoiding inflating notation we would like to introduce some abbreviations and conventions:

$$\begin{aligned} U_n &\equiv \partial_\rho^n U_k(\rho) && \text{the } n\text{th derivative of the } \textit{effective potential}, \\ \int &\equiv \int_{q'} && \text{integration over loop momentum } q' \text{ (cf. eq. (2.73)),} \\ &&& q \text{ remains reserved for external momentum and we define} \\ P_1^\varphi &\equiv \tilde{P}_{\varphi,k}^{reg}(q') && \text{as the (regulated and } \textit{modified}) \text{ bosonic propagator at momentum } q' \text{ as well as} \\ P_2^{\uparrow\downarrow} &\equiv P_{\uparrow\downarrow,k}^{reg}(q' - q) && \text{as the (regulated) fermionic propagators under the loop integral. We associate} \\ &&& \text{subscripts with momentum indices: } 1 \leftrightarrow q' \text{ and } 2 \leftrightarrow q' - q. \text{ Similarly,} \\ \dot{R}_1^{\uparrow\downarrow\varphi} &\equiv \partial_k R_{\uparrow\downarrow\varphi,k}(q') && \text{abbreviates the } \textit{scale-}k\text{-derivative of the particles } \textit{regulator functions.} \\ &&& \text{Another useful shorthand in use reads} \\ G_2^{\uparrow\downarrow/\varphi} &\equiv \left| G_{\uparrow\downarrow/\varphi}^{-1}(q' - q) \right| && \text{which refers to the fermionic/bosonic propagator determinant} \\ &&& \text{(cf. eqs. (2.164) and (2.166)).} \end{aligned} \quad (\text{D.2})$$

However, the set of equations becomes a bit more involved when explicitly distinguishing the propagators for spin-up and spin-down fermions. In particular the fermionic part of the (full) propagator evaluated at Φ reads

$$G_{k,FF'}(q)|_{\Phi} = \begin{pmatrix} -\mathcal{H}_{\varphi}(q) & \mathcal{P}_{\psi}^*(q) \\ -\mathcal{P}_{\psi}(q) & \mathcal{H}_{\varphi}^*(q) \end{pmatrix} \quad \text{with} \quad \mathcal{H}_{\varphi}(q) \equiv \begin{pmatrix} 0 & \frac{h_{\varphi}\phi}{G^{\uparrow\downarrow}(q)} \\ \frac{h_{\varphi}\phi}{G^{\uparrow\downarrow}(q)} & 0 \end{pmatrix}$$

$$\text{and} \quad \mathcal{P}_{\psi}(q) \equiv \begin{pmatrix} \frac{P_{\downarrow,k}(q)}{G^{\uparrow\downarrow}(q)} & 0 \\ 0 & \frac{P_{\uparrow,k}(q)}{G^{\uparrow\downarrow}(q)} \end{pmatrix} \quad (\text{D.3})$$

where

$$G^{\uparrow\downarrow}(q) \equiv h_{\phi}^2 \rho + P_{\uparrow,k}(q)P_{\downarrow,k}^*(q) = G^{\uparrow\downarrow}(-q) = [G^{\uparrow\downarrow}(q)]^* \xrightarrow{P_{\downarrow}=P_{\uparrow} \equiv P_{\psi}} |G_{\psi}^{-1}(q)| \quad . \quad (\text{D.4})$$

Besides the most general form of the equations we provide its reduction when regulating the bosonic degrees with a real-valued, symmetric regulator function $R_{\varphi,k}(q) = R_{\varphi,k}(-q) \in \mathbb{R}$ and fermions with the aid of anti-symmetric *imaginary regulators* $R^{\uparrow\downarrow} = i\bar{R}^{\uparrow\downarrow}$ where $\bar{R}_{\uparrow\downarrow}(-q) = -\bar{R}_{\uparrow\downarrow}(q) \in \mathbb{R}$ i.e. $R^{\uparrow\downarrow}(-q) = R^{\uparrow\downarrow*}(q)$. With this conventions at hand we should be able to read [table D.1](#).

Table D.1: Comprehensive list of flow equations in use for the investigation of an interacting Fermi gas. The symbol $\stackrel{reg}{\equiv}$ denotes the choice of specific regulator functions R as: $R_{\varphi,k}(q) = R_{\varphi,k}(-q) \in \mathbb{R}$ and $R^{\uparrow\downarrow} = i\tilde{R}^{\uparrow\downarrow}$ with $\tilde{R}_{\uparrow\downarrow}(q) \in \mathbb{R}$. Furthermore, $\stackrel{bal}{\equiv}$ reflects the transition from the *imbalanced* ($\mu_{\downarrow} \neq \mu_{\uparrow}$) to the *balanced* case. Notational abbreviations are given in eq. (D.2). The diagrammatics is quite sketchy—its foundation are discussed in sections 2.1.3 and 2.3.

flow. quant.	flow	SYM phase	diagrammatics
	$\Gamma_k[\eta] = \int_x U_k(\rho(x)) + \int_{q \neq 0} \varphi^*(q) P_{\varphi,k}(q) \varphi(q) + \sum_{\sigma=\uparrow,\downarrow} \int_q \psi_{\sigma}^*(q) P_{\sigma,k}(q) \psi_{\sigma}(q) + h_{\varphi} \int_{q,q'} \{ \varphi^*(q) \psi_{\uparrow}(q') \psi_{\downarrow}(q - q') - \varphi(q) \psi_{\uparrow}^*(q') \psi_{\downarrow}^*(q - q') \}$		
eff.	$= \frac{1}{2} \int \frac{P_1^{\varphi*} \tilde{R}_1^{\varphi} + P_1^{\varphi} \tilde{R}_1^{\varphi*}}{G_1^{\varphi}} - \int \frac{P_1^{\uparrow*} \tilde{R}_1^{\uparrow} + P_1^{\uparrow} \tilde{R}_1^{\uparrow*}}{G_1^{\parallel}}$		
pot.	$\stackrel{reg}{\equiv} \int \frac{\Re P_1^{\varphi}}{G_1^{\varphi}} \tilde{R}_1^{\varphi} - \int \frac{P_1^{\uparrow*}}{G_1^{\parallel}} \tilde{R}_1^{\uparrow} + \int \frac{P_1^{\uparrow*}}{G_1^{\parallel}} \tilde{R}_1^{\uparrow}$	$\rightarrow \int \frac{\tilde{R}_1^{\varphi}}{P_1^{\varphi}} - \int \frac{\tilde{R}_1^{\uparrow}}{P_1^{\uparrow}} - \int \frac{\tilde{R}_1^{\uparrow}}{P_1^{\uparrow}}$	
$\partial_k U_k$	$\stackrel{bal}{\equiv} \int \frac{\Re P_1^{\varphi}}{G_1^{\varphi}} \tilde{R}_1^{\varphi} - 2 \int \frac{\Im P_1^{\psi}}{G_1^{\psi}} \tilde{R}_1^{\psi}$	$\rightarrow \int \frac{\tilde{R}_1^{\varphi}}{P_1^{\varphi}} - 2 \int \frac{\tilde{R}_1^{\psi}}{P_1^{\psi}}$	
ferm.	$= h_{\varphi} \tilde{\partial}_k \int \frac{P_1^{\varphi*} P_2^{\uparrow\downarrow*}}{G_1^{\varphi} G_2^{\parallel\parallel}}$		
(inv.)	$= -\rho^2 h_{\varphi}^2 U_2 \int \frac{P_2^{\uparrow\downarrow*} \tilde{R}_1^{\varphi*}}{G_2^{\parallel\parallel} G_1^{\varphi 2}} + \rho h_{\varphi}^4 \int \frac{P_2^{\varphi} \tilde{R}_1^{\uparrow}}{G_2^{\varphi} G_1^{\parallel\parallel 2}}$		
prop.	$- h_{\varphi}^2 \int \frac{P_1^{\varphi* 2} P_2^{\uparrow\downarrow*}}{G_1^{\varphi 2} G_2^{\parallel\parallel}} \tilde{R}_1^{\varphi} - h_{\varphi}^2 \int \frac{P_2^{\varphi} P_1^{\uparrow\downarrow 2}}{G_2^{\varphi} G_1^{\parallel\parallel 2}} \tilde{R}_1^{\uparrow*}$	$\rightarrow -h_{\varphi}^2 \int \frac{\tilde{R}_1^{\varphi}}{P_1^{\varphi 2} P_2^{\uparrow* 2}} - h_{\varphi}^2 \int \frac{\tilde{R}_1^{\uparrow}}{P_1^{\varphi} P_2^{\uparrow* 2}}$	
$\partial_k P_{\uparrow\downarrow,k}$	$\stackrel{reg}{\equiv} -h_{\varphi}^2 \int \frac{(\rho^2 U_2^2 + P_1^{\varphi* 2}) P_2^{\psi*}}{G_1^{\varphi 2} G_2^{\psi}} \tilde{R}_1^{\varphi} + h_{\varphi}^2 \int \frac{(h_{\varphi} \rho + P_1^{\psi 2}) P_2^{\psi}}{G_1^{\psi 2} G_2^{\psi}} \tilde{R}_1^{\psi}$	$\rightarrow -h_{\varphi}^2 \int \frac{\tilde{R}_1^{\varphi}}{P_1^{\varphi 2} P_2^{\psi}} - h_{\varphi}^2 \int \frac{\tilde{R}_1^{\psi}}{P_1^{\psi} P_2^{\psi 2}}$	
bos.	$\partial_k P_{\varphi,k}(q) \equiv \beta_{\varphi,F}^{\parallel}(q) + \beta_{\varphi,B}^{(4)} + \beta_{\varphi,F}^{(3)}(q)$		
(inv.)			
prop.			

Table continued on next page ...

... table D.1 continued from previous page

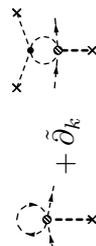
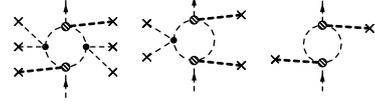
flow. quant.	flow	SYM phase	diagrammatics
$\beta_{\varphi,F}^{\parallel}$	$= -h_{\varphi}^2 \tilde{\delta}_k \int \frac{P_1^{\dagger*} P_2^{\dagger}}{G_1^{\parallel} G_2^{\parallel}}$ $= -h_{\varphi}^2 \rho \int \frac{P_2^{\dagger} \tilde{R}_1^{\dagger*}}{G_2^{\parallel} G_1^{\parallel 2}} - h_{\varphi} \rho \int \frac{P_2^{\dagger} \tilde{R}_1^{\dagger*}}{G_2^{\parallel} G_1^{\parallel 2}}$ $+ h_{\varphi}^2 \int \frac{P_2^{\dagger} P_1^{\dagger*2}}{G_2^{\parallel} G_1^{\parallel 2}} \tilde{R}_1^{\dagger} + h_{\varphi} \int \frac{P_2^{\dagger} P_1^{\dagger*2}}{G_2^{\parallel} G_1^{\parallel 2}} \tilde{R}_1^{\dagger}$	$\rightarrow +h_{\varphi}^2 \int \frac{\tilde{R}_1^{\dagger}}{P_2^{\dagger*} P_1^{\dagger 2}} + h_{\varphi}^2 \int \frac{\tilde{R}_1^{\dagger}}{P_2^{\dagger*} P_1^{\dagger 2}}$	$\sim \tilde{\delta}_k$ 
$\beta_{\varphi,F}^{\parallel}$	$\stackrel{reg.}{=} +2h_{\varphi} \int \frac{h_{\varphi} \rho P_2^{\psi} + P_2^{\psi} P_1^{\psi*2}}{G_2^{\psi} G_1^{\psi 2}} \tilde{R}_1^{\psi}$ $\stackrel{bal.}{=} 2h_{\varphi} \rho I_{F1} + 2h_{\varphi}^2 I_{F2}$	$\rightarrow +2h_{\varphi}^2 \int \frac{\tilde{R}_1^{\psi}}{P_2^{\psi*} P_1^{\psi 2}}$	
$\beta_{\varphi,B}^{(4)}$	$= -(U_4 U_2 \rho^4 + 4U_3 U_2 \rho^3 + 2U_2^2 \rho^2) \int \frac{\Im \tilde{R}_1^{\dagger}}{G_1^{\varphi 2}}$ $+ 2(U_4 U_2 \rho^3 + 3U_3 U_2 \rho^2) \int \frac{P_1^{\varphi*} \tilde{R}_1^{\dagger}}{G_1^{\varphi 2}}$ $- (U_4 \rho^2 + 4U_3 \rho + 2U_2) \int \frac{P_1^{\varphi*2} \tilde{R}_1^{\dagger}}{G_1^{\varphi 2}}$	$\rightarrow -2U_2 \int \frac{\tilde{R}_1^{\dagger}}{P_1^{\varphi 2}}$	$\sim \tilde{\delta}_k$ 
$\beta_{\varphi,B}^{(3)}$	$= -4(U_3^2 U_2^3 \rho^6 + 3U_3 U_2^4 \rho^5 + 2U_2^5 \rho^4) \int \frac{\tilde{R}_1^{\rho}}{G_1^{\varphi 2} G_2^{\rho}}$ $+ 8((U_3 U_2)^2 \rho^5 + 3U_3 U_2^3 \rho^4 + 2U_2^4 \rho^3) \int \frac{\Re P_1^{\rho} \tilde{R}_1^{\rho}}{G_1^{\varphi 2} G_2^{\rho}}$ $- 4(U_3 U_2)^2 \rho^5 \int \frac{\Im \tilde{R}_2^{\rho}}{G_1^{\varphi 2} G_2^{\rho}} \tilde{R}_1^{\rho}$ $+ 4(U_3 U_2^3 \rho^4 + U_2^4 \rho^3) \int \frac{P_2^{\rho} + 2\Re P_2^{\rho}}{G_1^{\varphi 2} G_2^{\rho}} \tilde{R}_1^{\rho}$ $- 4(U_3 U_2^2 \rho^3 + 2U_2^3 \rho^2) \int \frac{P_1^{\rho*2} + 2P_2^{\rho} \Re P_1^{\rho}}{G_1^{\varphi 2} G_2^{\rho}} \tilde{R}_1^{\rho}$ $- 4U_3^2 \rho^3 \int \frac{\Re P_2^{\rho} [(3\Im P_1^{\rho})^2 - (\Im \tilde{R}_1^{\rho})^2]}{G_1^{\varphi 2} G_2^{\rho}} \tilde{R}_1^{\rho}$ $- 4(U_3^2 U_2 \rho^4 + 2U_3 U_2^2 \rho^3) \int \frac{\Re P_1^{\rho*2} + 2\Re P_2^{\rho} \Re P_1^{\rho}}{G_1^{\varphi 2} G_2^{\rho}} \tilde{R}_1^{\rho}$ $+ 4(U_3 U_2 \rho^2 + U_2^2 \rho) \int \frac{2\Re P_2^{\rho} P_1^{\rho*2} + P_1^{\rho 2} P_2^{\rho}}{G_1^{\varphi 2} G_2^{\rho}} \tilde{R}_1^{\rho}$	$\rightarrow 0$	$\sim \tilde{\delta}_k$ 

Table continued on next page ...

flow. quant.	flow	SYM phase	diagrammatics
$\begin{array}{l} \text{reg,} \\ \text{bal} \end{array}$	$2[h_\varphi^2 I_{F2} - U_2 I_{B3}^{(4)}] \rho^0 + 2[h_\varphi^4 I_{F1} - 2U_3 I_{B3}^{(4)} + 2U_2^2 I_{B8}^{(3)}] \rho^1$	$[2U_3 U_2 (3I_{B2}^{(4)} + 2I_{B8}^{(3)}) - U_4 I_{B3}^{(4)} - 2U_2^3 (I_{B1}^{(4)} + 4I_{B5}^{(3)})] \rho^2$	
$\partial_k I_{\varphi,k}$	$+2[U_4 U_2 I_{B2}^{(4)} - 2U_3 U_2^2 (I_{B1}^{(4)} + 2I_{B5}^{(3)}) + 2U_2^4 (4I_{B2}^{(3)} + I_{B4}^{(3)}) - 2U_3^2 I_{B6}^{(3)}] \rho^3$	$[4U_3 U_2^3 (I_{B4}^{(3)} + 6I_{B2}^{(3)}) - U_4 U_2^2 I_{B1}^{(4)} - 4U_3^2 U_2 I_{B7}^{(3)} - 8U_2^5 I_{B1}^{(3)}] \rho^4$	
	$+4[(U_3 U_2)^2 (2I_{B2}^{(3)} - I_{B3}^{(3)}) - 6U_3 U_2^4 I_{B1}^{(3)}] \rho^5$	$- 4U_3^2 U_2^3 I_{B1}^{(3)} \rho^6$	

Table D.1: Comprehensive list of flow equations in use for the investigation of an interacting Fermi gas. The symbol $\overset{reg}{\equiv}$ denotes the choice of specific regulator functions R as: $R_{\varphi,k}(q) = R_{\varphi,k}(-q) \in \mathbb{R}$ and $R^{\uparrow\downarrow} = i\bar{R}^{\uparrow\downarrow}$ with $\bar{R}_{\uparrow\downarrow}(q) = -\bar{R}_{\uparrow\downarrow}(-q) \in \mathbb{R}$. Furthermore, $\overset{bal}{\equiv}$ reflects the transition from the *imbalanced* ($\mu_\downarrow \neq \mu_\uparrow$) to the *balanced* case. Notational abbreviations are given in eq. (D.2). The diagrammatics is quite sketchy—its foundation are discussed in sections 2.1.3 and 2.3.

Dank an . . .

In aller erster Linie möchte ich **Christof Wetterich** meinen Dank aussprechen. Er hat mir durch seine fachliche Begleitung die vorliegende Dissertation in seiner Arbeitsgruppe mit kompetentem physikalischen Rat und großzügigem Freiraum für die Ausgestaltung meiner Forschung ermöglicht. Die gesammelten Erfahrungen haben mich als Wissenschaftler wesentlich und unschätzbar gefördert.

Für das Interesse an meiner Arbeit und die sorgfältige Begutachtung dieser Promotionschrift bin ich **Franz Wegner** zu großem Dank verpflichtet. Ebenso schätze ich die Zeit, die **Selim Jochim** und **Werner Aeschbach–Hertig** “geopfert” haben, um im Rigorosum mit mir zu streiten.

Auf dem Weg zu dieser Doktorarbeit bereicherte mich der fachlicher Austausch mit Experten unterschiedlicher Fachgebiete. Insbesondere halfen mir **Sergej Moroz** und **Richard Schmidt**, einen guten Einstieg in die nicht-relativistische Physik kalter Quantengase vom Standpunkt der Funktionalen Renormierungsgruppe zu gewinnen. Sergejs stete Hilfsbereitschaft und Diskussion zur Physik des Unitären Fermi-Gases sowie Richards Erfahrung zur Impulsauflösung des (inversen) Propagators waren wegweisend für das vorliegende Projekt. Erfahrungsaustausch mit **Kai-Uwe Giering** zu technischen Aspekten der numerischen Implementierung der Flussgleichungen formte mein Konzept für den entwickelten Quellcode. Die vertiefte Ausarbeitung bestimmter Gesichtspunkte des Renormierungskonzepts wurde durch Diskussionen mit **David Schnörr** angeregt. **Konrad Rosenbaum**, **Richard Wolsch** und **Robert Lauer** halfen mit ihrer Erfahrung dazu beizutragen, dass die Bibliothek `libfrg` gewissen programmier-technischen Ansprüchen genügt. Die vorliegende Schrift haben auch alle diejenigen unterstützt, die mit Ihrer Lebenszeit die Entwicklung quell-offener und freier Software vorantreiben und voran getrieben haben. Ohne die Existenz von Initiativen wie dem *GNU Projekt* oder der *Linux Foundation* wäre diese Arbeit nicht möglich gewesen.

Eine hervorragende Umgebung zum unabhängigen und kreativen forschen boten mir—neben der breit aufgestellten Arbeitsgruppe von Christof am Institut für Theoretische Physik—die Heidelberg Graduiertenschule für Fundamentalphysik, sowie Stipendien der Studienstiftung des Deutschen Volkes und der International Max-Planck Research School für Quantendynamik. Allen, die Zeit und Kraft für die Existenz dieser Einrichtungen investiert haben, gilt mein herzlicher Dank.

Den Geist für die Naturwissenschaft erfrischte der Kreis meiner Freunde und Familie, indem mir interessierte (unkonventionelle) Fragen zum Thema sowie Wertschätzung meines Schaffens entgegen gebracht wurde. Hoch rechne ich meiner Frau **Juliane** und unserem Sohn **Timon** ihre Liebe und Zuwendung an, mit der sie meine Passion für die Physik mittragen.

Bibliography

Note: Explicit web links of this thesis are provided with access dates (as of $\langle month \rangle \langle year \rangle$) to cope with the dynamics in contents. Although we try to include hosts that are expected to exist on a long-term basis (extrapolated from their existence so far) we can not exclude unavailability of the provided resources. The project archive.org might serve as an option in the case of a dead link.

- [AALR79] Elihu Abrahams, Philip W. Anderson, Don C. Liccardello, and Tiruppattur V. Ramakrishnan, *Scaling Theory of Localization: Absence of Quantum Diffusion in Two Dimensions*, Physical Review Letters **42** (1979), no. 10, 673–676.
citation at page(s): 8
- [AEM⁺95] M H Anderson, J R Ensher, M R Matthews, C E Wieman, and E a Cornell, *Observation of bose-einstein condensation in a dilute atomic vapor.*, Science (New York, N.Y.) **269** (1995), no. 5221, 198–201.
citation at page(s): xi
- [And58] Philip W. Anderson, *Absence of Diffusion in Certain Random Lattices*, Physical Review **109** (1958), no. 5, 1492–1505.
citation at page(s): 8
- [ANR74] N. Ahmed, T. Natarajan, and K.R. Rao, *Discrete Cosine Transform*, IEEE Transactions on Computers **C-23** (1974), no. 1, 90–93.
citation at page(s): 102, 169
- [AS10] Alexander Altland and Ben Simons, *Condensed Matter Field Theory*, 2nd ed., Cambridge University Press, 2010.
citation at page(s): 1, 4, 5, 6, 9, 19, 33, 49, 81, 124, 141
- [AW05] G. B. Arfken and H. J. Weber, *Mathematical Methods for Physicists*, 3rd ed., Academic Press, 2005.
citation at page(s): 13, 97
- [Bak99] George Baker, *Neutron matter model*, Physical Review C **60** (1999), no. 5, 054311.
citation at page(s): 8
- [BBC⁺12] F. Benitez, J.-P. Blaizot, H. Chaté, B. Delamotte, R. Méndez-Galain, and N. Wschebor, *Nonperturbative renormalization group preserving full-momentum dependence: Implementation and quantitative evaluation*, Physical Review E **85** (2012), no. 2, 026707, [arXiv:1110.2665](https://arxiv.org/abs/1110.2665).
citation at page(s): 149
- [BBW04] Tobias Baier, Eike Bick, and Christof Wetterich, *Temperature dependence of antiferromagnetic order in the Hubbard model*, Physical Review B **70** (2004), no. 12, 125111, [arXiv:cond-mat/0309715](https://arxiv.org/abs/cond-mat/0309715).
citation at page(s): 111
- [BC97] P Butera and M Comi, *N-vector spin models on the simple-cubic and the body-centered-cubic lattices: A study of the critical behavior of the susceptibility and of the correlation length by high-temperature series extended to order β_{21}* , Physical Review B **56** (1997), no. 13, 8212–8240, [arXiv:hep-lat/9703018](https://arxiv.org/abs/hep-lat/9703018).
citation at page(s): 121
- [BDS11] Jens Braun, Sebastian Diehl, and Michael M. Scherer, *Finite-size and particle-number effects in an ultracold Fermi gas at unitarity*, Physical Review A **84** (2011), no. 6, 063616.
citation at page(s): 147

- [BFMMM96] H.G. Ballesteros, L.A. Fernández, V. Martín-Mayor, and A. Muñoz Sudupe, *Finite size effects on measures of critical exponents in $d = 3$ $O(N)$ models*, Physics Letters B **387** (1996), no. 1, 125–131, [arXiv:cond-mat/9606203](#).
citation at page(s): 121
- [BHLM95] Richard D Ball, Peter E Haagensen, JoséL. Latorre, and Enrique Moreno, *Scheme independence and the exact renormalization group*, Physics Letters B **347** (1995), no. 1-2, 80–88.
citation at page(s): 109
- [BIP06] BIPM, *International System of Units (SI)*, 8th ed., Bureau International des Poids et Mesures, December 2006.
citation at page(s): 157
- [BM78] Robert Brooks and J. Peter Matelski, *The Dynamics of 2-Generator Subgroups of $PSL(2, C)$* , Annals of Mathematics Studies **97** (1978), 65.
citation at page(s): x
- [BM12] Jürgen Berges and David Mesterházy, *Introduction to the nonequilibrium functional renormalization group*, Nuclear Physics B - Proceedings Supplements **228** (2012), 37–60, [arXiv:1204.1489](#).
citation at page(s): 17
- [BMGW06] Jean-Paul Blaizot, Ramón Méndez-Galain, and Nicolás Wschebor, *A new method to solve the non-perturbative renormalization group equations*, Physics Letters B **632** (2006), no. 4, 571–578, [arXiv:hep-th/0503103](#).
citation at page(s): 149
- [Bro73] R. Broucke, *Algorithm: ten subroutines for the manipulation of Chebyshev series*, Communications of the ACM **16** (1973), no. 4, 254–256.
citation at page(s): 102
- [BT04] Jean-Paul Berrut and Lloyd N. Trefethen, *Barycentric Lagrange Interpolation*, SIAM Review **46** (2004), no. 3, 501–517.
citation at page(s): 162
- [BTW00] Jürgen Berges, Nikolaos Tetradis, and Christof Wetterich, *Non-Perturbative Renormalization Flow in Quantum Field Theory and Statistical Physics*, Physics Reports **363** (2000), no. 4-6, 178, [arXiv:hep-ph/0005122](#).
citation at page(s): 16, 35, 113, 121, 141, 143
- [Buc14] E. Buckingham, *On Physically Similar Systems; Illustrations of the Use of Dimensional Equations*, Physical Review **4** (1914), no. 4, 345–376.
citation at page(s): 157
- [Bur98] C. P. Burgess, *Goldstone and Pseudo-Goldstone Bosons in Nuclear, Particle and Condensed-Matter Physics*, Physics Reports **330** (1998), no. 4, 119, [arXiv:hep-th/9808176](#).
citation at page(s): 19
- [Car96] John Cardy, *Scaling and Renormalization in Statistical Physics*, vol. 5, Cambridge University Press, 1996.
citation at page(s): 8
- [CCPS03] J. Carlson, S.-Y. Chang, V. Pandharipande, and K. Schmidt, *Superfluid Fermi Gases with Large Scattering Length*, Physical Review Letters **91** (2003), no. 5, 050401, [arXiv:physics/0303094](#).
citation at page(s): 8
- [CDMV03] Léonie Canet, Bertrand Delamotte, Dominique Mouhanna, and Julien Vidal, *Optimization of the derivative expansion in the nonperturbative renormalization group*, Physical Review D **67** (2003), no. 6, 065004.
citation at page(s): 134, 143, 146
- [CK90] J. R. Cash and Alan H. Karp, *A variable order Runge-Kutta method for initial value problems with rapidly varying right-hand sides*, ACM Transactions on Mathematical Software **16** (1990), no. 3, 201–222.
citation at page(s): 75
- [Cle55] C. W. Clenshaw, *A note on the summation of Chebyshev series*, Mathematics of Computation **9** (1955), no. 51, 118–118.
citation at page(s): 169

- [CLW67] J.W. Cooley, P.A.W. Lewis, and P.D. Welch, *Historical notes on the fast Fourier transform*, Proceedings of the IEEE **55** (1967), no. 10, 1675–1677.
citation at page(s): 161
- [CO95] IEEE Portable Applications Standards Committee and Others, *IEEE Std 1003.1 c-1995, Threads Extensions*, 1995.
citation at page(s): 85
- [CT65] James W. Cooley and John W Tukey, *An Algorithm for the Machine Calculation of Complex Fourier Series*, Mathematics of Computation **19** (1965), no. 90, 297.
citation at page(s): 102, 161, 170
- [CW90] Don Coppersmith and Shmuel Winograd, *Matrix multiplication via arithmetic progressions*, Journal of Symbolic Computation **9** (1990), no. 3, 251–280.
citation at page(s): 161
- [dB70] Louis de Broglie, *The Reinterpretation of Wave Mechanics*, Foundations of Physics **1** (1970), no. 1, 5–15.
citation at page(s): 48
- [Deb09] Peter J. W. Debye, *Näherungsformeln für die Zylinderfunktionen für große Werte des Arguments und unbeschränkt veränderliche Werte des Index*, Mathematische Annalen **67** (1909), no. 4, 535–558.
citation at page(s): 13
- [Del07] Bertrand Delamotte, *An Introduction to the Nonperturbative Renormalization Group*, preprint, 2007.
citation at page(s): 1
- [DFG⁺10] Sebastian Diehl, Stefan Floerchinger, Holger Gies, J.M. Pawłowski, and Christof Wetterich, *Functional renormalization group approach to the BCS-BEC crossover*, Annalen der Physik **656** (2010), no. 9, arXiv:0907.2193.
citation at page(s): 17, 35, 147
- [DMA⁺95] K. Davis, M. Mewes, M. Andrews, N. van Druten, D. Durfee, D. Kurn, and W. Ketterle, *Bose-Einstein Condensation in a Gas of Sodium Atoms*, Physical Review Letters **75** (1995), no. 22, 3969–3973.
citation at page(s): xi
- [DOV01] Michael J Duff, Lev B Okun, and Gabriele Veneziano, *Dialogue on the number of fundamental constants*, Journal of High Energy Physics **2002** (2001), no. 03, 023–023, arXiv:physics/0110060.
citation at page(s): 156
- [EE85] The Institute of Electrical Engineers and Electronics, *IEEE Standard for Binary Floating-Point Arithmetic*, preprint, 1985.
citation at page(s): 137
- [Fal03] Kenneth Falconer, *Fractal Geometry*, John Wiley & Sons, Ltd, Chichester, UK, September 2003.
citation at page(s): 13
- [Fei83] Mitchell J. Feigenbaum, *Universal Behavior in Nonlinear Systems*, Physica D: Nonlinear Phenomena **7** (1983), no. 1-3, 16–39.
citation at page(s): 8
- [Fis74] Michael E. Fisher, *The Renormalization Group in the Theory of Critical Behavior*, Reviews of Modern Physics **46** (1974), no. 4, 597–616.
citation at page(s): 8
- [FMS11] Stefan Floerchinger, Sergej Moroz, and Richard Schmidt, *Efimov Physics from the Functional Renormalization Group*, Few-Body Systems **51** (2011), no. 2-4, 153–180, arXiv:1102.0896v2.
citation at page(s): 17
- [GA92] Lori Goldner and Guenter Ahlers, *Superfluid fraction of He4 very close to T λ* , Physical Review B **45** (1992), no. 22, 13129–13132.
citation at page(s): 121

- [Geo99] Howard Georgi, *Lie algebras in particle physics*, Frontiers in Physics, Perseus Books, 1999.
citation at page(s): 9
- [GH93] Aloysius P. Gottlob and Martin Hasenbusch, *Critical behaviour of the 3D XY-model: a Monte Carlo study*, Physica A: Statistical Mechanics and its Applications **201** (1993), no. 4, 593–613.
citation at page(s): 121
- [Gie06] Holger Gies, *Introduction to the Functional RG and Applications to Gauge Theories*, Quantum (2006), 1–60, [arXiv:hep-ph/0611146v1](#).
citation at page(s): 1
- [GJ04] Holger Gies and Joerg Jaeckel, *Renormalization flow of QED*, Physical Review Letters **93** (2004), no. 11, 4, [arXiv:hep-ph/0405183](#).
citation at page(s): 16
- [GM62] Murray Gell-Mann, *Symmetries of Baryons and Mesons*, Physical Review **125** (1962), no. 3, 1067–1084.
citation at page(s): 9
- [GML54] Murray Gell-Mann and Francis E. Low, *Quantum Electrodynamics at Small Distances*, Physical Review **95** (1954), no. 5, 1300–1312.
citation at page(s): 8, 10
- [Gol61] J Goldstone, *Field theories with « Superconductor » solutions*, Il Nuovo Cimento **19** (1961), no. 1, 154–164.
citation at page(s): 19
- [GW93] Stanisław Głazek and Kenneth Wilson, *Renormalization of Hamiltonians*, Physical Review D **48** (1993), no. 12, 5863–5872.
citation at page(s): 10
- [GW01] G. Gersdorff and C. Wetterich, *Nonperturbative renormalization flow and essential scaling for the Kosterlitz-Thouless transition*, Physical Review B **64** (2001), no. 5, 054513.
citation at page(s): 149
- [Hah05] T. Hahn, *Cuba—a library for multidimensional numerical integration*, Computer Physics Communications **168** (2005), no. 2, 78–95, [arXiv:hep-ph/0404043](#).
citation at page(s): 73, 76
- [Hal06] Karen Hallberg, *New Trends in Density Matrix Renormalization*, Advances in Physics **55** (2006), no. 5-6, 54, [arXiv:cond-mat/0609039](#).
citation at page(s): 11
- [Haz95] Michiel Hazewinkel, *Encyclopaedia of Mathematics*, Encyclopaedia of Mathematics (closed) Series, no. Bd. 10, Kluwer Academic Publishers, 1995.
citation at page(s): 42, 93, 97, 100, 102, 103, 161, 164
- [Hei01] Henning Heiselberg, *Fermi systems with long scattering lengths*, Physical Review A **63** (2001), no. 4, 043606, [arXiv:cond-mat/0002056](#).
citation at page(s): 8
- [HS01a] Carsten Honerkamp and Manfred Salmhofer, *Magnetic and superconducting instabilities of the Hubbard model at the van Hove filling*, Physical Review Letters **87** (2001), no. 18, 4, [arXiv:cond-mat/0109392](#).
citation at page(s): 17
- [HS01b] ———, *Temperature-flow renormalization group and the competition between superconductivity and ferromagnetism*, Physical Review B **64** (2001), no. 18, 184516.
citation at page(s): 111
- [Hub59] John Hubbard, *Calculation of Partition Functions*, Physical Review Letters **3** (1959), no. 2, 77–78.
citation at page(s): 33

- [IBM56] IBM, *The Fortran Automatic Coding System For The IBM 704 EDPM*, IBM Corporation, 1956.
citation at page(s): 73
- [iso03] *ISO/IEC 14882:2003: Programming languages: C++*, 2003.
citation at page(s): 73
- [Kad66] Leo P. Kadanoff, *Scaling laws for Ising models near T_c* , *Physics* **2** (1966), no. 6, 263.
citation at page(s): 10
- [Kad90] ———, *Scaling and Universality in Statistical Physics*, *Physica A: Statistical Mechanics and its Applications* **163** (1990), no. 1, 1–14.
citation at page(s): 8
- [KBS10] Peter Kopietz, Lorenz Bartosch, and Florian Schütz, *Introduction to the functional renormalization group*, vol. 798, Springer Verlag, 2010.
citation at page(s): 11
- [Kle11] Hagen Kleinert, *Hubbard-Stratonovich Transformation: Successes, Failure, and Cure*, *Electronic Journal of Theoretical Physics* **8** (2011), no. 25, 57–64, [arXiv:1104.5161](https://arxiv.org/abs/1104.5161).
citation at page(s): 33
- [KR88] Brian W. Kernighan and Dennis M. Ritchie, *The C programming Language*, vol. 8, Prentice-Hall, 1988.
citation at page(s): 73
- [KSCZ12] Mark J H Ku, Ariel T Sommer, Lawrence W Cheuk, and Martin W Zwierlein, *Revealing the superfluid lambda transition in the universal thermodynamics of a unitary Fermi gas.*, *Science (New York, N.Y.)* **335** (2012), no. 6068, 563–7.
citation at page(s): 147
- [Kub57] Ryogo Kubo, *Statistical-Mechanical Theory of Irreversible Processes. I. General Theory and Simple Applications to Magnetic and Conduction Problems*, *Journal of the Physical Society of Japan* **12** (1957), no. 6, 570–586.
citation at page(s): 6
- [KZ08] Wolfgang Ketterle and Martin W Zwierlein, *Making, probing and understanding ultracold Fermi gases*, lecture notes (2008), [arXiv:0801.2500](https://arxiv.org/abs/0801.2500).
citation at page(s): 33, 149
- [LB05] David P. Landau and Kurt Binder, *A Guide to Monte Carlo Simulations in Statistical Physics*, vol. 38, Cambridge University Press, Cambridge, 2005.
citation at page(s): 17, 73
- [Lit01a] Daniel F. Litim, *Mind The Gap*, *International Journal of Modern Physics A* **16** (2001), no. 11, 6, [arXiv:hep-th/0104221](https://arxiv.org/abs/hep-th/0104221).
citation at page(s): 34, 108
- [Lit01b] Daniel F Litim, *Optimised Renormalisation Group Flows*, *Physics Letters B* **486** (2001), no. 1–2, 92–99.
citation at page(s): 34, 108
- [Lit01c] Daniel F. Litim, *Optimized renormalization group flows*, *Physical Review D* **64** (2001), no. 10, 105007, [arXiv:hep-th/0103195](https://arxiv.org/abs/hep-th/0103195).
citation at page(s): 34, 108
- [Lit02] ———, *Critical exponents from optimised renormalisation group flows*, *Nuclear Physics B* **631** (2002), no. 1-2, 128–158.
citation at page(s): 121
- [Lor63] Edward N. Lorenz, *Deterministic Nonperiodic Flow*, *Journal of the Atmospheric Sciences* **20** (1963), no. 2, 130–141.
citation at page(s): ix, 73
- [Man83] Benoit B. Mandelbrot, *The Fractal Geometry of Nature*, vol. 51, W. H. Freeman, 1983.
citation at page(s): x, 13

- [Man88] Franz Mandl, *Statistical Physics*, 2nd ed., Wiley, 1988.
citation at page(s): 3
- [MB03] J. Mitroy and M. Bromley, *Semiempirical calculation of van der Waals coefficients for alkali-metal and alkaline-earth-metal atoms*, *Physical Review A* **68** (2003), no. 5, 052714.
citation at page(s): 159
- [MBvS12] D. Mesterházy, J. Berges, and L. von Smekal, *Effect of short-range interactions on the quantum critical behavior of spinless fermions on the honeycomb lattice*, *Physical Review B* **86** (2012), no. 24, 245431, [arXiv:1207.4054v3](#).
citation at page(s): 149
- [MK78] Humphrey J. Maris and Leo P. Kadanoff, *Teaching the Renormalization Group*, *American Journal of Physics* **46** (1978), no. 6, 652.
citation at page(s): 8
- [Moo01] Glyn Moody, *Rebel Code*, The Penguin Press, 2001.
citation at page(s): 119
- [Mor11] Sergej Moroz, *Few-Body Physics with Functional Renormalization*, Ph.D. thesis, Heidelberg University, 2011.
citation at page(s): 19
- [MSH⁺11] Walter Metzner, Manfred Salmhofer, Carsten Honerkamp, Volker Meden, and Kurt Schoenhammer, *Functional renormalization group approach to correlated fermion systems*, *Reviews of Modern Physics* **84** (2011), no. 1, 59, [arXiv:1105.5289](#).
citation at page(s): 23
- [MTN12] Peter J. Mohr, Barry N. Taylor, and David B. Newell, *CODATA recommended values of the fundamental physical constants: 2010*, *Reviews of Modern Physics* **84** (2012), no. 4, 1527–1605, [arXiv:1203.5425](#).
citation at page(s): 157, 159
- [Nee97] Tristan Needham, *Visual Complex Analysis*, Oxford University Press, 1997.
citation at page(s): 42, 93, 161
- [Nie06] Max Niedermaier, *The Asymptotic Safety Scenario in Quantum Gravity – An Introduction*, *Classical and Quantum Gravity* **24** (2006), no. 18, R171–R230, [arXiv:gr-qc/0610018](#).
citation at page(s): 14
- [NNCS10] N Navon, S Nascimbène, F Chevy, and C Salomon, *The equation of state of a low-temperature Fermi gas with tunable interactions.*, *Science (New York, N.Y.)* **328** (2010), no. 5979, 729–32.
citation at page(s): 147
- [Pau40] Wolfgang Pauli, *The Connection Between Spin and Statistics*, *Physical Review* **58** (1940), no. 8, 716–722.
citation at page(s): 49
- [Paw07] Jan M. Pawłowski, *Aspects of the functional renormalisation group*, *Annals of Physics* **322** (2007), no. 12, 2831–2915, [arXiv:hep-th/0512261](#).
citation at page(s): 11, 109
- [Paw10] ———, *The QCD phase diagram: Results and challenges*, *AIP conference proceedings* **1343** (2010), 6, [arXiv:1012.5075](#).
citation at page(s): 16
- [PFTV92] William Press, Brian Flannery, Saul Teukolsky, and William Vetterling, *Numerical Recipes In C: The Art of Scientific Computing*, vol. 29, November 1992.
citation at page(s): 100, 103, 161
- [Pol84] Joseph Polchinski, *Renormalization and effective lagrangians*, *Nuclear Physics B* **231** (1984), no. 2, 269–295.
citation at page(s): 10
- [PS95] Michael E. Peskin and Daniel V. Schroeder, *An introduction to quantum field theory*, *Advanced book classics*, Addison-Wesley Publishing Company, 1995.
citation at page(s): 124

- [Ran06] Rolf Rannacher, *Einfuehrung in die Numerische Mathematik (Numerik 0)*, no. April, 2006.
citation at page(s): 76, 96, 100, 103, 104, 161
- [RD12a] a. Rançon and N. Dupuis, *Quantum criticality of a Bose gas in an optical lattice near the Mott transition*, Physical Review A **85** (2012), no. 1, 011602, arXiv:1107.1314v2.
citation at page(s): 149
- [RD12b] ———, *Universal thermodynamics of a two-dimensional Bose gas*, Physical Review A **85** (2012), no. 6, 063607, arXiv:1203.1788v2.
citation at page(s): 149
- [Reu98] Martin Reuter, *Nonperturbative evolution equation for quantum gravity*, Physical Review D **57** (1998), no. 2, 971–985, arXiv:hep-th/9605030.
citation at page(s): 14, 16
- [Run01] Carl Runge, *Über empirische Funktionen und die Interpolation zwischen äquidistanten Ordinaten*, Zeitschrift für Mathematik und Physik **46** (1901), no. 224-243, 20.
citation at page(s): 102, 161
- [Sac11] Subir Sachdev, *Quantum phase transitions*, 2nd ed., Cambridge University Press, 2011.
citation at page(s): 124
- [Sal99] Manfred Salmhofer, *Renormalization*, Springer Verlag, 1999.
citation at page(s): 11
- [SCL92] D. Swanson, T. Chui, and J. Lipa, *Propagation of second sound near $T\lambda$* , Physical Review B **46** (1992), no. 14, 9043–9050.
citation at page(s): 121
- [SE11] Richard Schmidt and Tilman Enss, *Excitation spectra and rf-response near the polaron-to-molecule transition from the functional renormalization group*, Physical Review A **83** (2011), no. 6, 15, arXiv:1104.1379.
citation at page(s): 17
- [Sha94a] Ramamurti Shankar, *Principles of quantum mechanics*, vol. 233, Plenum Press New York, 1994.
citation at page(s): 48, 156
- [Sha94b] ———, *Renormalization-group approach to interacting fermions*, Reviews of Modern Physics **66** (1994), no. 1, 129–192.
citation at page(s): 8
- [Sha98] Claude E. Shannon, *Communication In The Presence Of Noise*, Proceedings of the IEEE **86** (1998), no. 2, 447–457.
citation at page(s): 163
- [Sma98] Steve Smale, *Mathematical problems for the next century*, The Mathematical Intelligencer **20** (1998), no. 2, 7–15.
citation at page(s): ix
- [Sre04] Mark Srednicki, *Quantum Field Theory: Spin Zero*, preprint, 2004.
citation at page(s): 4, 7
- [Sta68] H. Stanley, *Dependence of Critical Properties on Dimensionality of Spins*, Physical Review Letters **20** (1968), no. 12, 589–592.
citation at page(s): 123
- [Str57] Ruslan L. Stratonovich, *On a Method of Calculating Quantum Distribution Functions*, Soviet Physics Doklady **2** (1957), 416.
citation at page(s): 33
- [Str69] Volker Strassen, *Gaussian elimination is not optimal*, Numerische Mathematik **13** (1969), no. 4, 354–356.
citation at page(s): 161

- [Str94] Steven H. Strogatz, *Nonlinear Dynamics And Chaos: With Applications To Physics, Biology, Chemistry And Engineering*, Studies in nonlinearity, Westview Press, 1994.
citation at page(s): xi, 8, 121, 130, 141
- [SW09] Mikhail Shaposhnikov and Christof Wetterich, *Asymptotic safety of gravity and the Higgs boson mass*, Physics Letters B **683** (2009), no. 2-3, 8, [arXiv:0912.0208](https://arxiv.org/abs/0912.0208).
citation at page(s): 16
- [Tay50a] Geoffrey Taylor, *The formation of a blast wave by a very intense explosion I. Theoretical discussion*, Proceedings of the Royal Society London A **201** (1950), no. 1065, 159–174.
citation at page(s): 8
- [Tay50b] ———, *The Formation of a Blast Wave by a Very Intense Explosion II. The Atomic Explosion of 1945.*, Proceedings of the Royal Society London A **201** (1950), no. 1065, 175–186.
citation at page(s): 8
- [Tre11] Lloyd N. Trefethen, *Six Myths of Polynomial Interpolation and Quadrature*, preprint, 2011.
citation at page(s): 103
- [Tre13] ———, *Approximation Theory and Approximation Practice*, Siam, 2013.
citation at page(s): 102, 161
- [TT07] Gilles Tarjus and Matthieu Tissier, *Non-Perturbative Functional Renormalization Group for Random Field Models and Related Disordered Systems. I: Effective Average Action Formalism*, Physical Review B **78** (2007), no. 2, 20, [arXiv:0712.3550](https://arxiv.org/abs/0712.3550).
citation at page(s): 17
- [Tuc02] Warwick Tucker, *A Rigorous ODE Solver and Smale's 14th Problem*, Foundations of Computational Mathematics **2** (2002), 53–117.
citation at page(s): ix
- [TW94] Nikolaos Tetradis and Christof Wetterich, *Critical exponents from the effective average action*, Nuclear Physics B **422** (1994), no. 3, 541–592.
citation at page(s): 16, 113, 121, 129
- [Weg94] Franz J. Wegner, *Flow Equations for Hamiltonians*, Annals of Physics **3** (1994), 77–91
note: hyperlink refers to available reprint in Physics Reports.
citation at page(s): 10
- [Weg01] ———, *Flow equations for Hamiltonians*, Physics Reports **348** (2001), no. 1-2, 77–89.
citation at page(s): 10
- [Wet93] Christof Wetterich, *Exact evolution equation for the effective potential*, Physics Letters B **301** (1993), no. 1, 90–94.
citation at page(s): xi, 10
- [WH73] Franz J. Wegner and Anthony Houghton, *Renormalization Group Equation for Critical Phenomena*, Physical Review A **8** (1973), no. 1, 401–412.
citation at page(s): 10
- [WHI79] Steven Weinberg, S W Hawking, and W Israel, *General relativity: an Einstein centenary survey*, Cambridge UP, Cambridge (1979), 790.
citation at page(s): 14
- [Whi92] Steven R. White, *Density matrix formulation for quantum renormalization groups*, Physical Review Letters **69** (1992), no. 19, 2863–2866.
citation at page(s): 10
- [Wil71] Kenneth G. Wilson, *Renormalization Group and Critical Phenomena. I. Renormalization Group and the Kadanoff Scaling Picture*, Physical Review B **4** (1971), no. 9, 3174–3183.
citation at page(s): 10

- [Wil75] ———, *The Renormalization Group: Critical Phenomena and the Kondo Problem*, Rev. Mod. Phys. **47** (1975), no. 4, 773–840.
citation at page(s): 8
- [Zee05] Anthony Zee, *Quantum field theory in a nutshell*, Universities Press, 2005.
citation at page(s): 1
- [ZJO02] Jean Zinn-Justin and Others, *Quantum field theory and critical phenomena*, vol. 142, Clarendon Press Oxford, 2002.
citation at page(s): 1, 81, 121, 124
- [ZRM08] R. K. P. Zia, Edward F. Redish, and Susan R. McKay, *Making Sense of the Legendre Transform*, preprint, 2008.
citation at page(s): 6

Erklärung/Affirmation

Ich versichere, dass ich diese Arbeit selbstständig verfasst habe und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

I affirm this thesis as independent research performed exclusively on my own. I have only used the sources indicated.

Heidelberg, den

C o n r a d A l b r e c h t