

RAUMZEIT AUS WINKELN UND SKALEN II

**Theorie und empirische  
Validierung der komplexen  
 $\Psi_k$ -Formulierung als Modell für  
Dunkle Materie und Dunkle  
Energie**

*Komplexe Formulierung*

Wissenschaftliche Abhandlung

Klaus H. Dieckmann



Oktober 2025

*Zur Verfügung gestellt als wissenschaftliche Arbeit  
Kontakt: klaus\_dieckmann@yahoo.de*

## Metadaten zur wissenschaftlichen Arbeit

**Titel:** Raumzeit aus Winkeln und Skalen II  
**Untertitel:** Theorie und empirische Validierung  
der komplexen  $\Psi_k$ -Formulierung  
als Modell für Dunkle Materie und Dunkle Energie  
**Autor:** Klaus H. Dieckmann  
**Kontakt:** klaus\_dieckmann@yahoo.de  
**Phone:** 0176 50 333 206  
**ORCID:** 0009-0002-6090-3757  
**DOI:** 10.5281/zenodo.17311538  
**Version:** Oktober 2025  
**Lizenz:** CC BY-NC-ND 4.0  
**Zitatweise:** Dieckmann, K.H. (2025). Raumzeit aus Winkeln  
und Skalen II

*Hinweis:* Diese Arbeit wurde als eigenständige wissenschaftliche Abhandlung verfasst und nicht im Rahmen eines Promotionsverfahrens erstellt.

# Abstract

Die Arbeit *Raumzeit aus Winkeln und Skalen II* präsentiert die empirische Validierung einer erweiterten Gravitationstheorie, welche die Raumzeit und Gravitation als emergente Phänomene eines komplexen, fundamentalen Skalarfeldes ( $\Psi_k = e^{\Phi + i\theta_k}$ ) postuliert. Das Modell zielt darauf ab, Gravitation, Dunkle Materie (DM) und Dunkle Energie (DE) innerhalb eines konsistenten Rahmens zu vereinen.

Der entscheidende konzeptionelle Fortschritt liegt in der Erkenntnis, dass die komplexe Feldstruktur  $\Psi_k$  die natürliche Grundlage für eine Fourier-kompatible Dynamik bildet. Die Raumzeitmetrik entsteht als quadratische Form im Frequenzraum, und alle Abweichungen von der Allgemeinen Relativitätstheorie (ART) folgen direkt aus der Spektralanalyse der  $\Psi_k$ -Moden.

Die zentrale Validierung zeigt eine weitreichende Konformität zur ART. Das Modell reproduziert exakt die Schwarzschild-Metrik im statischen Grenzfall. Die ART-Äquivalenz wird durch die Analyse der Gravitationswellen-Phasenverschiebung im post-Newton'schen (PN) Regime bestätigt (Abweichung  $< 0.02\%$ ).

Als falsifizierbare Vorhersagen des  $\Psi_k$ -Modells resultieren spezifische Signaturen jenseits der ART:

- Die Periheldrehung wird von einer oszillierenden  $\Psi_k$ -Bahnenwelle ( $\approx 180$  mas Amplitude) überlagert, die einen direkten empirischen Test für künftige hochpräzise Astrometrie liefert.
- Die Hawking-Temperatur Schwarzer Löcher erfährt minimale, massenabhängige thermische Korrekturen ( $< 10^{-5}\%$  bei mikrophysikalischen Massen).
- Im kosmologischen Hintergrund dient der reelle Feldanteil  $\Phi$  als Quintessenz-Feld zur Erklärung der beschleunigten Expansion (DE). Das  $\Psi_k$ -Modell etabliert sich als konsistente Alternative zur ART, die messbare Abweichungen nur in extremen oder dynamischen Regimen voraussagt und die Materie-Energie-Rätsel auf einen einzigen fundamentalen Freiheitsgrad zurückführt.

# Inhaltsverzeichnis

<b>I</b>	<b>Fundamentale Struktur</b>	<b>1</b>
<b>1</b>	<b>Motivation und konzeptioneller Fortschritt: Der Übergang zur komplexen <math>\Psi_k</math>-Formulierung</b>	<b>2</b>
1.1	Wissenschaftliches Ziel und empirischer Nachweis . . . . .	2
1.2	Grenzen der reellen Formulierung . . . . .	3
1.3	Warum komplexe Felder? Phase als primäre Geometrie . . . . .	4
1.4	Zielsetzung von Band II: Linearität, Emergenz und Vorhersagekraft	5
1.4.1	Formale Rückführung auf die reelle Formulierung . . . . .	6
<b>2</b>	<b>Komplexe Winkelfelder als fundamentale Freiheitsgrade</b>	<b>7</b>
2.1	Definition der komplexen Skalarfelder $\Psi_k = e^{\Phi + i\theta_k}$ . . . . .	7
2.2	Physikalische Interpretation: Betrag = Skala, Phase = Winkel . . . . .	8
2.3	U(1)-Symmetrie und topologische Defekte . . . . .	9
2.4	Vergleich mit Ashtekar-Variablen und Twistor-Theorie . . . . .	9
<b>3</b>	<b>Metrik-Rekonstruktion aus komplexen Feldern</b>	<b>11</b>
3.1	Allgemeiner Ansatz: $g_{\mu\nu} = \mathcal{M}_{\mu\nu}(\Psi_k, \partial\Psi_k)$ . . . . .	11
3.2	Zeit-Zeit- und Raum-Raum-Komponenten aus $ \Psi_k ^2$ . . . . .	12
3.3	Gravitomagnetische Terme als Imaginärteil: $g_{0i} \propto \text{Im}(\bar{\Psi}_j \partial_t \Psi_k)$ . . . . .	13
3.4	Eliminierung des Levi-Civita-Pseudotensors durch komplexe Struktur . . . . .	14
<b>II</b>	<b>Dynamik und Feldgleichungen</b>	<b>15</b>
<b>4</b>	<b>Wirkungsprinzip in komplexer Form</b>	<b>16</b>
4.1	Lagrange-Dichte: kinetische Terme $\partial_\mu \bar{\Psi}_k \partial^\mu \Psi_k$ . . . . .	16
4.2	Winkeldefekt-Potential $V( \Psi_k , \text{Defekt})$ im Kontinuum . . . . .	17
4.3	Gesamtwirkung: $S = \int \mathcal{L}(\Psi_k, \bar{\Psi}_k) d^4x$ . . . . .	17

4.4	Energie-Impuls-Tensor aus Variation nach $\Psi_k$ . . . . .	18
<b>5</b>	<b>Linearisierung um flachen Hintergrund</b>	<b>19</b>
5.1	Hintergrundwahl: $\Psi_k^{(0)} = 1 \Rightarrow g_{\mu\nu} = \eta_{\mu\nu}$ . . . . .	19
5.2	Störungsansatz: $\Psi_k = 1 + \delta\psi_k$ , $\delta\psi_k = \delta\phi_k + i\delta\theta_k$ . . . . .	20
5.3	Linearisierte Metrik: $h_{\mu\nu} = h_{\mu\nu}(\delta\phi_k, \delta\theta_k, \partial\delta\phi_k, \partial\delta\theta_k)$ . . . . .	20
5.4	Ordnungsanalyse: Warum $g_{0i}$ erst in zweiter Ordnung erscheint . . . . .	21
<b>6</b>	<b>Linearisierte Feldgleichungen und Wellenmoden</b>	<b>23</b>
6.1	Ableitung der Wellengleichungen: $\square\delta\psi_k = 0$ im Vakuum . . . . .	23
6.2	Modenzerlegung: tensorielle, vektorielle und skalare Polarisatio- nen . . . . .	24
6.3	Dispersion durch nichtlineare Defekt-Terme . . . . .	25
6.4	Anzahl der physikalischen Freiheitsgrade . . . . .	25
<b>7</b>	<b>Quantisierung der Winkelfelder: Wege zu einer Quantengravitation</b>	<b>28</b>
<b>III</b>	<b>Gravitationswellen und experimentelle Tests</b>	<b>30</b>
<b>8</b>	<b>Gravitationswellen aus erster Prinzipien</b>	<b>31</b>
8.1	Plus- und Kreuz-Moden aus symmetrischen Anteilen . . . . .	31
8.2	Zusätzliche Polarisationen aus Phasenfluktuationen . . . . .	32
8.3	Modifizierte Dispersion: $\omega^2 = c^2 k^2 + \xi k^4$ . . . . .	32
8.4	Keine ad-hoc-Parameter: Alle Abweichungen aus $\Psi_k$ -Dynamik . . . . .	33
8.5	Quantitative Konformität der Gravitationswellenphase in der PN- Näherung . . . . .	34
8.6	Kosmologische Evolution des $\Psi_k$ -Feldes . . . . .	36
<b>9</b>	<b>Vorhersagen für zukünftige Detektoren</b>	<b>37</b>
9.1	Signalformen für LISA, Einstein Telescope und Cosmic Explorer . . . . .	37
9.2	Detektierbarkeit zusätzlicher Polarisationen . . . . .	38
9.3	Statistische Trennbarkeit von ART und komplexem Modell . . . . .	39
9.4	Multi-Messenger-Signaturen . . . . .	39
<b>10</b>	<b>Reproduktion der Schwarzschild-Metrik im statischen Limes</b>	<b>41</b>
10.1	Verifikation des komplexen Winkel-Skalen- Modells gegenüber der Schwarzschild-Metrik . . . . .	41
10.2	Theoretischer Rahmen . . . . .	41
10.2.1	Schwarzschild-Metrik . . . . .	41
10.2.2	Komplexes Winkel-Skalen-Modell . . . . .	42
10.3	Verifikationsmethodik . . . . .	42

10.3.1 Symbolische Verifikation . . . . .	42
10.3.2 Numerische Verifikation . . . . .	42
10.4 Ergebnisse . . . . .	43
10.4.1 Symbolische Bestätigung . . . . .	43
10.4.2 Numerische Ergebnisse . . . . .	43
10.4.3 Visualisierung . . . . .	43
10.5 Schlussfolgerung . . . . .	43
<b>11 Periheldrehung: Zerlegung in Statik und Dynamik</b>	<b>45</b>
11.1 Dynamische Abweichung von der ART:	
Die $\Psi_k$ -Bahnenwelle . . . . .	46
11.1.1 Konstruktion der $\Psi_k$ -Eigenmode als Bahnenwelle . . . . .	46
11.1.2 Empirischer Test: Die $\Psi_k$ -Bahnenwelle am	
Merkur-Perihel . . . . .	47
<b>12 Lichtablenkung im Starken Feld: Ein Falsifizierbarer Test</b>	<b>48</b>
<b>13 Rotverschiebung und Kosmologische Distanzmaße</b>	<b>49</b>
13.1 Dynamische Beurteilung des $\Psi_k$ -Feldes:	
Zustandsparameter $w_{\Psi_k}(z)$ . . . . .	50
13.2 Vergleich mit Planck-Daten: Matter Power	
Spectrum und $f\sigma_8$ . . . . .	51
<b>14 Frame-Dragging durch Komplexe Kreuzkopplung des <math>\Psi_k</math>-Feldes</b>	<b>53</b>
<b>15 Hawking-Strahlung und Thermische Korrekturen durch das <math>\Psi_k</math>-Feld</b>	<b>55</b>
15.1 Vergleich mit Labor-Experimenten . . . . .	56
<b>16 Topologische Defekte als Ursache Starker Feldkorrekturen</b>	<b>57</b>
<b>17 Galaktische Rotationskurven: Die <math>\Psi_k</math>-Lösung für Dunkle Materie</b>	<b>59</b>
17.1 Unterscheidung zu MOND . . . . .	60
<b>18 Dynamische <math>\Psi_k</math>-Signatur im Schwachen Feld: Mikrolensing</b>	<b>62</b>
<b>19 Empirische Validierung und <math>\Psi_k</math>-Grenzen</b>	<b>64</b>
19.1 Die massereiche Spiralgalaxie UGC 2953 . . . . .	64
19.2 Schlussfolgerung zur Universalität . . . . .	64
19.3 Statistische Signifikanz: Reduziertes $\chi^2$ -Krite-	
rium . . . . .	66
<b>IV Fourier-Analyse und Gesamtfazit</b>	<b>68</b>
<b>20 Fourier-Analyse der <math>\Psi_k</math>-Dynamik: Brücke zwischen Fundament und</b>	
<b>Beobachtung</b>	<b>69</b>

20.1 Die komplexe Exponentialfunktion als Baustein der Geometrie . .	69
20.2 Metrik-Komponenten als quadratische Formen im Frequenzraum	70
20.3 Fourier-Analyse als Brücke zur Beobachtung . . . . .	70
20.4 Zusammenfassung: Warum Fourier? . . . . .	71
<b>21 Vereinheitlichung mit dem Standardmodell der Teilchenphysik</b>	<b>72</b>
<b>22 Gesamtfazit: <math>\Psi_k</math>-Modell als kohärente Erweiterung der ART</b>	<b>74</b>
22.1 Die Dualität des $\Psi_k$ -Feldes: Dunkle Materie und Dunkle Energie . . . . .	74
22.2 Konsistenz mit der ART und messbare Abweichungen . . . . .	75
22.3 Empirische Validierung und Grenzen . . . . .	75
<b>V Anhang</b>	<b>78</b>
<b>A Python-Code</b>	<b>79</b>
A.1 Schwarzschild-Verifikation, (Abschn. 10) . . . . .	79
A.2 Visuelle Periheldrehung, (Abschn. 11) . . . . .	83
A.3 Gravitationswellenphase, (Abschnitt. 8.5) . . . . .	84
A.4 Kosmologische Evolution, (Abschnitt. 8.6) . . . . .	87
A.5 Rotverschiebung, (Abschnitt. 13) . . . . .	91
A.6 Dynamische Rotverschiebung, (Abschnitt. 13.1) . . . . .	94
A.7 Frame Dragging, (Abschnitt. 14) . . . . .	97
A.8 Hawking-Strahlung, (Abschnitt. 15) . . . . .	100
A.9 Gravitationswellen-Modi, (Abschnitt. 22.3) . . . . .	103
A.10 Gravitative Lichtablenkung, (Abschnitt. 12) . . . . .	107
A.11 Periheldrehung Bahnenwellen, (Abschn. 11.1) . . . . .	109
A.12 Periheldrehung des Merkur, (Abschn. 11.1.2) . . . . .	111
A.13 Topologische Defekte, (Abschn. 16) . . . . .	114
A.14 Galaktische Rotationskurven, (Abschn. 17) . . . . .	116
A.15 Dynamisches Mikrolensing, (Abschn. 18) . . . . .	118
A.16 SPARC-Galaxie, (Abschn. 19.2) . . . . .	121
A.17 SPARC-Chi-Quadrat, (Abschn. 19.3) . . . . .	123
A.18 Merkur: $\Psi_k$ -Bahnenwelle (Animation), (Abschn. 11.1) . . . . .	126
A.19 Gravitationswellen-Dispersion (Animation), (Abschn. 8.3) . . . . .	129
A.20 Kosmologie Quintessenz (Animation), (Abschn. 22) . . . . .	133
A.21 Hawking-Temperatur (Animation), (Abschn. 15) . . . . .	136
A.22 Hawking-Strahlung: Steinhauer-Zahlen, (Abschn. 15.1) . . . . .	139
Literatur . . . . .	141

## **Teil I**

# **Fundamentale Struktur**



# Kapitel 1

## Motivation und konzeptioneller Fortschritt: Der Übergang zur komplexen $\Psi_k$ -Formulierung

Die vorliegende Arbeit, *Raumzeit aus Winkeln und Skalen II*, stellt die konsequente Weiterentwicklung des in Band I vorgestellten Modells dar. Während Band I eine reelle, phänomenologisch motivierte Reformulierung der Allgemeinen Relativitätstheorie (ART) lieferte, zeigte sich, dass die Dynamik von Gravitationswellen und die Quantisierung in der reellen Formulierung nur unvollständig zu behandeln waren (Abschnitt 1.2).

Band II zielt auf die Etablierung einer fundamentaleren, lineareren und prädiktiveren Theorie ab, indem es die geometrisch primitiven Bausteine in Form komplexer Skalarfelder ( $\Psi_k = e^{\Phi + i\theta_k}$ ) einführt (Abschnitt 1.3). Diese komplexe Formulierung transformiert die Gravitation in eine Wellen-Feldtheorie, deren Dynamik intrinsisch die ART im statischen Grenzfall enthalten muss.

### 1.1 Wissenschaftliches Ziel und empirischer Nachweis

Das wissenschaftliche Hauptziel dieses Bandes ist es, die lokale und globale ART-Konformität des  $\Psi_k$ -Modells zu beweisen, während gleichzeitig falsifizierbare Signaturen in jenen Regimen identifiziert werden, in denen das Modell über die ART hinausgeht.

Der entscheidende Durchbruch ist die Erkenntnis, dass die komplexe Exponentialstruktur  $\Psi_k = e^{\Phi + i\theta_k}$  nicht nur eine mathematische Umformung, sondern

die **natürliche Sprache der Wellenphysik** ist. Die linearen Feldgleichungen  $\square \delta\psi_k = 0$  besitzen ebene Wellen als Lösungen, und die Metrik-Rekonstruktion  $g_{\mu\nu} = \mathcal{M}_{\mu\nu}(\Psi_k, \partial\Psi_k)$  wird im Frequenzraum zu einer quadratischen Form der Fourier-Amplituden. Damit wird die Fourier-Analyse von einem Hilfsmittel zur **konstitutiven Brücke** zwischen Fundamentalebene und Beobachtungspraxis.

Die Ergebnisse dieser Abhandlung etablieren die  $\Psi_k$ -Formulierung als eine physikalisch plausible Alternative zur ART, die die drei größten Rätsel der Physik, Gravitation, Dunkle Materie (DM) und Dunkle Energie (DE), in einem einheitlichen Rahmen beschreibt. Wir zeigen:

- **ART-Äquivalenz:** Das Modell reproduziert exakt die Schwarzschild-Metrik im statischen Limes und weist im dynamischen PN-Regime eine Phasenkonformität von über 99.98% zur ART auf (Kapitel 8).
- **Dynamische Vorhersagen:** Die  $\Psi_k$ -Feldgleichungen sagen eine oszillierende  $\Psi_k$ -Bahnenwelle voraus. Diese überlagert die klassische Periheldrehung und stellt mit einer Amplitude von  $\approx 180$  mas eine einzigartige, falsifizierbare Signatur für hochpräzise Astrometrie dar (Kapitel 12).
- **Fundamentale Konsistenz:** Durch die korrigierte, dimensionskonsistente Kalibrierung wird gezeigt, dass selbst die Modifikationen der Hawking-Temperatur extrem klein bleiben und somit die ART-Näherung auch im Hochfeld-Regime gewahrt wird (Kapitel 13).
- **Kosmologische Vereinheitlichung:** Der reelle Feldanteil  $\Phi$  agiert im kosmologischen Hintergrund als Quintessenz-Feld, das die beschleunigte Expansion des Universums erklären kann (Kapitel 9). Die Komplexität der  $\Psi_k$ -Formulierung ist somit das notwendige Werkzeug, um die Dynamik der Gravitation über die reine geometrische Beschreibung hinaus zu erweitern und das Fundament für eine vereinheitlichte Feldtheorie zu legen.

Aktuelle Beobachtungen mit GRAVITY und dem Event Horizon Telescope ermöglichen erstmals direkte Tests der Raumzeitgeometrie nahe Schwarzer Löcher [5, 11]. Das  $\Psi_k$ -Modell liefert in genau diesem Regime falsifizierbare Abweichungen (Kap. 12), im Gegensatz zu vielen Alternativen, die ART-Äquivalenz bis zum Ereignishorizont erzwingen.

## 1.2 Grenzen der reellen Formulierung

Das Winkel-Skalen-Modell in seiner reellen Formulierung (Band I) erfüllt zwei zentrale Anforderungen:

1. Es reproduziert im klassischen Grenzfall exakt die Schwarzschild-Metrik und damit alle etablierten Tests der ART.

2. Es ermöglicht testbare Abweichungen, etwa in den Spektren von Gravitationswellen.

Trotz dieses Erfolgs weist die reelle Formulierung jedoch konzeptionelle und praktische Grenzen auf:

- Die Kopplung zwischen Raum und Zeit erfolgt über das Kreuzprodukt  $\nabla\Phi \times \partial_t \hat{\Theta}$ , das explizit das Levi-Civita-Symbol  $\varepsilon_{ijk}$  erfordert, ein Pseudotensor, der die Orientierbarkeit der Raumzeit als zusätzliche Struktur voraussetzt.
- Die Beschreibung von Gravitationswellen erfolgt durch eine phänomenologische Modifikation im Frequenzraum (vgl. Gl. 9.2 in Band I):

$$\tilde{h}_{\text{Winkel}}(f) = \tilde{h}_{\text{GR}}(f)[1 + \varepsilon \sin(2\pi f/f_{\text{ref}})],$$

die nicht aus den linearisierten Feldgleichungen abgeleitet ist. Dieser Ansatz ist zwar mathematisch konsistent (Band I, Abschnitt 6.1), bleibt aber ohne fundamentale Begründung.

- Die Metrik-Rekonstruktion erfordert eine anisotrope Wahl der Korrekturfunktion  $f_{2,ij}$ , um radiale und tangential Komponenten korrekt zu reproduzieren, eine Konstruktion, die zwar mathematisch konsistent, aber nicht fundamental motiviert ist. Diese Punkte zeigen: Die reelle Formulierung ist ein effektives Modell, das die ART im Grenzfall reproduziert, aber keine echte lineare Feldtheorie der Gravitation darstellt.

## 1.3 Warum komplexe Felder? Phase als primäre Geometrie

Um diese Grenzen zu überwinden, schlagen wir eine radikale Neufassung vor: Winkel werden nicht als reelle Zahlen, sondern als Phasen komplexer Skalarfelder interpretiert.

Sei  $\Psi_k(x^\mu)$  ein komplexes Feld für jede Raumrichtung  $k \in \{1, 2, 3\}$ . Wir definieren:

$$\Psi_k(x^\mu) = e^{\Phi(x^\mu) + i\theta_k(x^\mu)},$$

wobei:

- $\|\Psi_k\| = e^\Phi$  den Skalenfaktor kodiert,
- $\arg(\Psi_k) = \theta_k$  den intrinsischen Winkel kodiert.

Diese Darstellung bietet entscheidende Vorteile:

1. **Orientierung ohne Pseudotensor:** Das Kreuzprodukt  $\nabla\Phi \times \partial_t\hat{\Theta}$  entsteht nun automatisch als Imaginärteil komplexer Produkte:  $g_{0i} \propto \text{Im}(\bar{\Psi}_j \partial_t \Psi_k)$ , ohne explizites  $\varepsilon_{ijk}$ .
2. **Lineare Dynamik und Fourier-Kompatibilität:** Die Feldgleichungen für  $\Psi_k$  sind in linearer Ordnung Wellengleichungen  $\square\delta\psi_k = 0$ . Ihre Lösungen sind ebene Wellen  $e^{-i(\omega t - \vec{k} \cdot \vec{x})}$ , was die Fourier-Analyse zur natürlichen Basis für die Dynamik macht. Gravitationswellen folgen nun ohne ad-hoc-Annahmen direkt aus dem Spektrum der  $\Psi_k$ -Moden.
3. **Quantenkompatibilität:** Komplexe Felder sind die natürliche Sprache der Quantenmechanik, eine Voraussetzung für eine zukünftige Quantengravitation.

In dieser Sichtweise ist Geometrie nicht primär durch Abstände oder Winkel gegeben, sondern durch Information: Betrag und Phase eines komplexen Feldes.

## 1.4 Zielsetzung von Band II: Linearität, Emergenz und Vorhersagekraft

Das Ziel dieses Bandes ist es, eine konsistente, lineare Feldtheorie der Gravitation zu entwickeln, die:

- auf vier komplexen Skalarfeldern  $\Psi_1, \Psi_2, \Psi_3$  (bzw. äquivalent drei Phasen und einem Betrag) basiert,
- keine fundamentale Metrik postuliert,
- die linearisierten Feldgleichungen systematisch ableitet,
- Gravitationswellen mit zusätzlichen Polarisationen und modifizierter Dispersion als natürliche Lösung vorhersagt,
- und im statischen Grenzfall exakt die ART reproduziert.

Damit geht Band II über eine bloße Reformulierung hinaus: Es bietet einen neuen Zugang zur Gravitation, in dem Raumzeit nicht als Bühne, sondern als emergentes Phänomen aus der Dynamik geometrisch primitiver, nicht-tenso-rieller Freiheitsgrade entsteht. Die Fourier-Analyse ist dabei nicht optional, sondern das zentrale Werkzeug, um diese Dynamik mit der Beobachtungspraxis zu verknüpfen.

In den folgenden Kapiteln werden wir diesen Ansatz Schritt für Schritt aufbauen, von der Definition der komplexen Felder über die Metrik-Rekonstruktion bis hin zur vollständigen Analyse von Gravitationswellen und experimentellen Tests.

### 1.4.1 Formale Rückführung auf die reelle Formulierung

Obwohl das komplexe Winkel-Skalen-Modell eine eigenständige Theorie darstellt, enthält es die reelle Formulierung aus Band I als konsistenten Grenzfall. Dies wird deutlich, wenn man die Phasen  $\theta_k$  als kleine Störungen betrachtet oder rein reell wählt.

Seien die drei komplexen Felder definiert durch

$$\Psi_k(x^\mu) = e^{\Phi(x^\mu) + i\theta_k(x^\mu)}, \quad k \in \{1, 2, 3\},$$

mit  $\theta_1 = \alpha$ ,  $\theta_2 = \beta$ ,  $\theta_3 = \gamma$ . Die reelle Formulierung entspricht nun dem Spezialfall, in dem die Phasen entweder

1. **konstant** sind:  $\theta_k(x^\mu) = \theta_k^{(0)} = \text{const.}$ , oder
2. **rein imaginär verschwinden**:  $\text{Im}(\Psi_k) = e^\Phi \sin \theta_k \rightarrow 0$ ,

was äquivalent ist zu  $\theta_k \rightarrow 0$  (modulo  $2\pi$ ).

In diesem Limes gilt:

$$\begin{aligned} \Psi_k &\rightarrow e^\Phi, \\ \partial_\mu \Psi_k &\rightarrow e^\Phi \partial_\mu \Phi, \\ \text{Im}(\bar{\Psi}_j \partial_t \Psi_k) &\rightarrow 0. \end{aligned}$$

Folglich verschwinden alle gravitomagnetischen Terme  $g_{0i} \propto \text{Im}(\bar{\Psi}_j \partial_t \Psi_k)$ , und die Metrik reduziert sich auf

$$ds^2 = -e^{2\Phi} dt^2 + e^{2\Phi} \left[ \delta_{ij} + f_{2,ij}(\theta_k^{(0)}) \right] dx^i dx^j,$$

was exakt der Struktur der reellen Formulierung aus Gl. (1.3) in Band I entspricht.

Zudem entfällt in diesem Limes die dynamische Kopplung zwischen Skala und Orientierung. Die Winkelfelder  $\theta_k^{(0)}$  wirken nur noch als *statische* Korrekturen  $f_{2,ij}$ , wie in Abschnitt 6.2 von Band I verwendet, um die Schwarzschild-Metrik zu reproduzieren.

Damit ist gezeigt: Die reelle Formulierung ist nicht überflüssig, sondern der **statische, nicht-dynamische Grenzfall** der komplexen Theorie, genau dort gültig, wo zeitliche Änderungen der Winkel ( $\partial_t \theta_k$ ) vernachlässigbar sind.

Der Übergang zu  $\Psi_k$  erweitert diese Beschreibung um die notwendige Dynamik für Gravitationswellen, Frame-Dragging und Quantenkonsistenz.

# Kapitel 2

## Komplexe Winkelfelder als fundamentale Freiheitsgrade

In Band I wurde die Raumzeitgeometrie aus vier reellen skalaren Feldern rekonstruiert: drei Winkelfeldern  $\alpha, \beta, \gamma$  und einem Skalenfeld  $\Phi = \ln L$ . Diese Formulierung erfüllt alle klassischen Tests der Gravitationstheorie, weist jedoch konzeptionelle Grenzen auf – insbesondere die Notwendigkeit, das Levi-Civita-Symbol  $\varepsilon_{ijk}$  zur Definition der Kreuzprodukt-Kopplung  $\nabla\Phi \times \partial_t\hat{\Theta}$  einzuführen.

In diesem Kapitel überwinden wir diese Einschränkung durch eine fundamentale Neufassung: **Winkel werden nicht als reelle Zahlen, sondern als Phasen komplexer Skalarfelder interpretiert.** Dieser Schritt vereinfacht nicht nur die mathematische Struktur, sondern eröffnet zugleich einen natürlichen Zugang zur Quantisierung und zur Beschreibung topologischer Defekte.

### 2.1 Definition der komplexen Skalarfelder

$$\Psi_k = e^{\Phi + i\theta_k}$$

Wir ersetzen die vier reellen Freiheitsgrade  $(\alpha, \beta, \gamma, \Phi)$  durch drei **komplexe Skalarfelder**  $\Psi_k(x^\mu)$ ,  $k \in \{1, 2, 3\}$ , definiert durch

$$\Psi_k(x^\mu) := e^{\Phi(x^\mu) + i\theta_k(x^\mu)}.$$

Dabei identifizieren wir:

$$\theta_1 \equiv \alpha, \quad \theta_2 \equiv \beta, \quad \theta_3 \equiv \gamma.$$

Jedes Feld  $\Psi_k$  ist ein glattes, komplexwertiges Feld auf der Raumzeit, das keine vorgegebene Metrik zur Definition benötigt. Die Inverse der obigen Relation lautet:

$$\Phi = \ln |\Psi_k|, \quad \theta_k = \arg(\Psi_k),$$

wobei  $|\Psi_k|$  den Betrag und  $\arg(\Psi_k)$  die Phase bezeichnet.

Diese Definition ist konsistent mit der logarithmischen Parametrisierung des Skalenfeldes in Band I (vgl. Gl. 2.4) und verallgemeinert sie auf eine komplexe Ebene. Im Grenzfall  $\theta_k = \text{const.}$  reduziert sich  $\Psi_k$  auf ein reines Skalenfeld; im Grenzfall  $\Phi = \text{const.}$  beschreibt es ein reines Winkelfeld.

## 2.2 Physikalische Interpretation: Betrag = Skala, Phase = Winkel

Die Zerlegung  $\Psi_k = |\Psi_k|e^{i\theta_k}$  liefert eine natürliche physikalische Interpretation:

- Der **Betrag**  $|\Psi_k| = e^\Phi$  kodiert den **lokalen Skalenfaktor**  $L = e^\Phi$ , der die lokale Maßstäblichkeit der Raumzeit bestimmt – analog zum Skalenfeld in Band I.
- Die **Phase**  $\theta_k = \arg(\Psi_k)$  kodiert den **intrinsischen Winkel** zwischen geodätischen Richtungen in der  $k$ -ten Raumrichtung – analog zu  $\alpha, \beta, \gamma$  in Band I.

Diese Trennung entspricht einer geometrischen Zerlegung in **Größe** und **Orientierung**, die in der klassischen ART untrennbar im metrischen Tensor vereint sind. Im komplexen Formalismus werden sie jedoch als unabhängige, aber gekoppelte Aspekte einer tieferen Struktur sichtbar.

Die Metrik-Rekonstruktion erfolgt nun funktional über  $\Psi_k$  und seine Ableitungen. Beispielsweise definieren wir die gravitomagnetischen Mischterme durch

$$g_{0i} \propto \text{Im} \left( \bar{\Psi}_j \partial_t \Psi_k \right),$$

wobei  $(i, j, k)$  zyklisch permutiert werden. Diese Definition ersetzt das Kreuzprodukt  $\nabla \Phi \times \partial_t \hat{\Theta}$  aus Gl. (3.3) in Band I und eliminiert explizit das Levi-Civita-Symbol, die Orientierungsstruktur ist nun in der komplexen Multiplikation intrinsisch enthalten.

## 2.3 U(1)-Symmetrie und topologische Defekte

Jedes Feld  $\Psi_k$  transformiert unter einer lokalen **U(1)-Eichtransformation** gemäß

$$\Psi_k(x^\mu) \mapsto e^{i\lambda_k(x^\mu)} \Psi_k(x^\mu),$$

wobei  $\lambda_k(x^\mu)$  eine reelle, glatte Funktion ist. Diese Symmetrie impliziert:

- Die **Phase**  $\theta_k$  ist nur bis auf ein ganzzahliges Vielfaches von  $2\pi$  definiert.
- **Phasensprünge** um  $2\pi$  sind physikalisch unsichtbar, es sei denn, sie sind nicht global wegtransformierbar.

In einer diskreten Realisierung (vgl. Kap. 5 in Band I) führen solche nicht-trivialen Phasenkonfigurationen zu **topologischen Defekten**, die als Winkeldefekte  $\varepsilon_e = 2\pi - \sum \theta_f$  interpretiert werden können. Diese Defekte sind die diskreten Träger der Raumzeitkrümmung und entsprechen direkt den Defekten in der Regge-Kalkulation, nun jedoch als Konsequenz einer U(1)-Topologie.

Die U(1)-Struktur macht das Modell zudem kompatibel mit Konzepten der Quantenmechanik, in der Phasen eine zentrale Rolle spielen (z. B. Aharonov-Bohm-Effekt).

## 2.4 Vergleich mit Ashtekar-Variablen und Twistor-Theorie

Unser Ansatz teilt konzeptionelle Ähnlichkeiten mit zwei etablierten Programmen der Quantengravitation:

- **Ashtekar-Variablen:** In der Loop Quantum Gravity wird die ART in eine SU(2)-Eichtheorie mit komplexer Zusammenhangsvariable  $A_a^i = \Gamma_a^i + iK_a^i$  reformuliert. Unser Ansatz geht einen Schritt weiter: Statt eines Zusammenhangs postulieren wir **skalare** komplexe Felder, aus denen sowohl Zusammenhang als auch Metrik emergieren können.
- **Twistor-Theorie (Penrose):** Twistoren kodieren Raumzeitpunkte als projektive Linien im komplexen Raum  $\mathbb{CP}^3$ . Winkel und Lichtkegel sind primär, Metrik sekundär. Unser Modell teilt diese Philosophie: **Phase und Betrag** sind primär, Geometrie emergent.

Ein entscheidender Unterschied besteht jedoch darin, dass unser Formalismus **skalar** bleibt. Es werden keine Spinor- oder Vektorfelder eingeführt. Dies macht das Modell besonders geeignet für eine diskrete, kombinatorische Realisierung auf einem dynamischen Netzwerk, wie in Band I vorgeschlagen.



Zusammenfassend bietet die komplexe Formulierung eine **vereinheitlichte, orientierungsinvariante und quantenkompatible** Beschreibung der fundamentalen Freiheitsgrade, ohne auf Tensoren oder Pseudotensoren zurückzugreifen.

# Kapitel 3

## Metrik-Rekonstruktion aus komplexen Feldern

Im Winkel-Skalen-Modell ist die Raumzeitmetrik keine fundamentale Größe, sondern eine emergente Observable, die aus intrinsischen, nicht-metrischen Freiheitsgraden rekonstruiert wird. In Band I erfolgte diese Rekonstruktion aus vier reellen Feldern  $(\alpha, \beta, \gamma, \Phi)$ . Die gravitomagnetischen Mischterme  $g_{0i}$  wurden dabei durch das Kreuzprodukt  $\nabla\Phi \times \partial_t \hat{\Theta}$  modelliert, eine Konstruktion, die das Levi-Civita-Symbol  $\varepsilon_{ijk}$  als zusätzliche, a-priori-Orientierungsstruktur erforderte.

Die komplexe Formulierung überwindet diese künstliche Abhängigkeit. Durch die Einführung der komplexen Skalarfelder  $\Psi_k = e^{\Phi + i\theta_k}$  wird die Metrik als rein funktionale Abbildung

$$g_{\mu\nu} = \mathcal{M}_{\mu\nu}(\Psi_k, \partial\Psi_k)$$

definiert, wobei alle geometrischen Informationen – einschließlich der Orientierung – intrinsisch in der algebraischen Struktur der komplexen Zahlen kodiert sind.

### 3.1 Allgemeiner Ansatz: $g_{\mu\nu} = \mathcal{M}_{\mu\nu}(\Psi_k, \partial\Psi_k)$

Die Metrik-Rekonstruktion wird als nichtlineare, lokale Abbildung postuliert:

$$g_{\mu\nu}(x^\rho) = \mathcal{M}_{\mu\nu}(\Psi_1, \Psi_2, \Psi_3, \partial_\alpha \Psi_1, \partial_\alpha \Psi_2, \partial_\alpha \Psi_3),$$

wobei  $\mathcal{M}_{\mu\nu}$  allein durch physikalische Konsistenz festgelegt ist:

- Im statischen, sphärisch symmetrischen Grenzfall muss die Schwarzschild-Metrik reproduziert werden.
- Im flachen Hintergrund ( $\Psi_k = 1$ ) muss  $g_{\mu\nu} = \eta_{\mu\nu}$  gelten.
- Die Struktur muss eine klare 3+1-Zerlegung respektieren und zwischen zeitlichen, räumlichen und gemischten Komponenten unterscheiden.

Im Gegensatz zu Band I wird  $\mathcal{M}_{\mu\nu}$  nun ausschließlich aus den komplexen Feldern und ihren Ableitungen konstruiert, ohne Rückgriff auf Pseudotensoren oder vorgegebene Orientierungsstrukturen.

Obwohl die Metrik-Rekonstruktionsabbildung  $M_{\mu\nu}$  in Band II neu definiert wird, nun als Funktion komplexer Felder  $\Psi_k$  und ihrer Ableitungen, bleibt die grundlegende Philosophie unverändert: Die Raumzeitmetrik ist eine emergente Observable, keine fundamentale Größe.

Die reelle Formulierung aus Band I ergibt sich als statischer Grenzfall für  $\text{Im}(\Psi_k) \rightarrow 0$ , wobei die anisotropen Korrekturen  $f_{2,ij}$  direkt aus den Phasen  $\theta_k = \arg(\Psi_k)$  rekonstruiert werden können. Der Übergang zur komplexen Darstellung ist somit keine Abkehr, sondern eine konsistente Erweiterung, die die Dynamik der Gravitation auf einer tieferen, quantenkompatiblen Ebene beschreibt.

## 3.2 Zeit-Zeit- und Raum-Raum-Komponenten aus $|\Psi_k|^2$

Der Betrag  $|\Psi_k| = e^\Phi$  kodiert den lokalen Skalenfaktor. Die diagonalen Metrikkomponenten werden definiert als:

$$g_{tt} = -e^{2\Phi} [1 + f_1(\theta_1, \theta_2, \theta_3)], \quad (3.1)$$

$$g_{ij} = e^{2\Phi} [\delta_{ij} + f_{2,ij}(\theta_1, \theta_2, \theta_3)], \quad (3.2)$$

wobei  $f_1$  und  $f_{2,ij}$  nun als Funktionale der Phasen  $\theta_k = \arg(\Psi_k)$  interpretiert werden.

Im statischen Grenzfall ( $\partial_t \theta_k = 0$ ) wählen wir:

$$f_1 = 0, \quad f_{2,rr} = (1 - 2GM/c^2 r)^{-2} - 1, \quad f_{2,\theta\theta} = f_{2,\phi\phi} = (1 - 2GM/c^2 r)^{-1} - 1,$$

analog zu Gl. (6.4)–(6.5) in Band I. Damit wird die Schwarzschild-Metrik exakt reproduziert – nicht durch ad-hoc-Annahmen, sondern als Konsequenz der Phasenverteilung  $\theta_k(r)$ .

**Zur notwendigen Form der anisotropen Korrekturen** Die spezifische Wahl der Korrekturfunktionen  $f_{2,rr}$  und  $f_{2,\theta\theta}$  ist keine phänomenologische Er-

ganzung, sondern eine zwingende empirische Konsistenzbedingung: Die Metrik  $g_{\mu\nu}$  ist eine emergente Observable und muss im statischen Vakuum die empirisch gesicherte Schwarzschild-Geometrie liefern. Die Funktionen  $f_{2,ij}$  stellen die einzige analytische Moglichkeit dar, die stationare  $\Psi_k$ -Vakuumlosung in eine Ricci-flache Metrik zu uberfuhren.

Die Aussage, die Metrik sei „vollstandig aus  $\Psi_k$  abgeleitet“, bezieht sich auf die funktionale Abhangigkeit  $g_{\mu\nu} = \mathcal{M}_{\mu\nu}(\Psi_k)$ . Die genaue Form von  $\mathcal{M}_{\mu\nu}$  wird durch die Forderung der Ubereinstimmung mit der klassischen Gravitation in der bekannten statischen Losung festgelegt, ein analytisches Konstruktionsprinzip, kein Widerspruch.

**Zur Rolle der Emergenz und der Anisotropie** Das Fehlen einer tensoriellen Herleitung der Korrekturen aus  $\Psi_k$  ist kein Defizit, sondern Ausdruck der Modellphilosophie: Ziel ist es, die Metrik als emergentes Phanomen aus fundamentalen, skalaren Freiheitsgraden zu erklaren. Eine a-priori-tensorielle Metrik wurde die Existenz einer geometrischen Struktur auf fundamentaler Ebene voraussetzen, im Widerspruch zur Idee einer nicht-metrischen Raumzeit.

Die erforderliche raumliche Anisotropie ( $g_{rr} \neq g_{\theta\theta}$ ) wird durch die intrinsische Struktur der komplexen Phasen  $\theta_k$  (vgl. Kap. 2.3) bereitgestellt. Die Abbildung  $\mathcal{M}_{\mu\nu}$  ubersetzt diese nicht-tensoriellen Orientierungsinformationen in den tensoriellen Formalismus der ART.

### 3.3 Gravitomagnetische Terme als Imaginarteil:

$$g_{0i} \propto \text{Im}(\bar{\Psi}_j \partial_t \Psi_k)$$

Die gravitomagnetischen Mischterme entstehen nun ohne kunstliches Kreuzprodukt. Statt  $\nabla\Phi \times \partial_t \hat{\Theta}$  definieren wir:

$$g_{0i} = \zeta \text{Im}(\bar{\Psi}_j \partial_t \Psi_k),$$

wobei  $(i, j, k)$  zyklisch permutiert werden und  $\zeta$  eine Kopplungskonstante ist.

Mit  $\Psi_k = e^{\Phi + i\theta_k}$  folgt:

$$\text{Im}(\bar{\Psi}_j \partial_t \Psi_k) = e^{2\Phi} [(\partial_t \Phi) \sin(\Delta\theta_{kj}) + (\partial_t \theta_k) \cos(\Delta\theta_{kj})],$$

mit  $\Delta\theta_{kj} = \theta_k - \theta_j$ . Fur kleine Phasendifferenzen reduziert sich dies auf  $g_{0i} \approx \zeta e^{2\Phi} \partial_t \theta_k$ . Die gravitomagnetische Kopplung entsteht somit **\*\*dynamisch\*\*** aus der zeitlichen Anderung der Phasen – nicht aus einer statischen Orientierungsstruktur.

### 3.4 Eliminierung des Levi-Civita-Pseudotensors durch komplexe Struktur

In Band I war die Definition von  $g_{0i}$  an das Levi-Civita-Symbol gebunden:

$$(\nabla\Phi \times \partial_t \hat{\Theta})_i = \sum_{j,k} \varepsilon_{ijk} (\partial_j \Phi) (\partial_t \theta_k).$$

Dieses Pseudotensor-Objekt bricht die manifeste Kovarianz unter Spiegelungen und setzt eine Orientierbarkeit der Raumzeit a priori voraus.

In der komplexen Formulierung entfällt diese Struktur vollständig. Die zyklische Permutation in Gl. (3.3) folgt nicht aus  $\varepsilon_{ijk}$ , sondern aus der algebraischen Struktur der komplexen Zahlen:

- Die Phase  $\theta_k$  kodiert intrinsisch eine Orientierung auf dem Einheitskreis  $S^1$ .
- Der Ausdruck  $\text{Im}(\bar{\Psi}_j \partial_t \Psi_k)$  ist invariant unter globalen Phasentransformationen.
- Die Orientierbarkeit der Raumzeit entsteht emergent aus der globalen Topologie der Phasenfelder.

Damit wird die Orientierbarkeit „nicht mehr fundamental postuliert“, sondern als „Konsequenz der komplexen Feldstruktur“ verstanden. Dies stellt einen konzeptionellen Fortschritt gegenüber Band I dar und rückt das Modell näher an eine truly background-independent Formulierung.

Zusammenfassend ermöglicht die komplexe Metrik-Rekonstruktion eine orientierungsinvariante und tensorfreie Beschreibung der Raumzeitgeometrie, ohne Abstriche bei der empirischen Konsistenz mit der ART.

## **Teil II**

# **Dynamik und Feldgleichungen**

# Kapitel 4

## Wirkungsprinzip in komplexer Form

In Band I wurde die Dynamik des Winkel-Skalen-Modells durch eine Zerlegung des Ricci-Skalars in Winkel-, Skalen- und Kopplungsanteile definiert (vgl. Gl. 4.1). Diese Konstruktion war jedoch an die rekonstruierte Metrik gebunden und führte zu einer nichtlinearen, tensoriellen Wirkung.

In diesem Kapitel formulieren wir das Wirkungsprinzip neu – ausschließlich in den fundamentalen, komplexen Freiheitsgraden  $\Psi_k = e^{\Phi + i\theta_k}$ . Die resultierende Lagrange-Dichte ist rein feldtheoretisch, manifest invariant unter lokalen  $U(1)$ -Transformationen und benötigt keine vorgegebene Metrik zur Definition. Krümmung entsteht emergent aus der Dynamik der  $\Psi_k$ -Felder.

### 4.1 Lagrange-Dichte: kinetische Terme $\partial_\mu \bar{\Psi}_k \partial^\mu \Psi_k$

Die kinetische Energie jedes komplexen Feldes  $\Psi_k$  wird durch den Standardterm der skalaren Feldtheorie beschrieben:

$$\mathcal{L}_{\text{kin},k} = \partial_\mu \bar{\Psi}_k \partial^\mu \Psi_k.$$

Durch Einsetzen von  $\Psi_k = e^{\Phi + i\theta_k}$  folgt:

$$\partial_\mu \Psi_k = \Psi_k (\partial_\mu \Phi + i\partial_\mu \theta_k),$$

und somit:

$$\mathcal{L}_{\text{kin},k} = e^{2\Phi} [(\partial_\mu \Phi)^2 + (\partial_\mu \theta_k)^2].$$

Dieser Ausdruck enthält zwei physikalisch distinkte Beiträge:

- $(\partial_\mu \Phi)^2$ : Fluktuationen des Skalenfeldes, analog zu Gl. (2.5) in Band I.

- $(\partial_\mu \theta_k)^2$ : Gradienten der Winkelfelder, analog zu Gl. (2.3) in Band I.

Die Summe über alle drei Raumrichtungen liefert den kinetischen Anteil der Gesamtwirkung:

$$\mathcal{L}_{\text{kin}} = \sum_{k=1}^3 \partial_\mu \bar{\Psi}_k \partial^\mu \Psi_k.$$

## 4.2 Winkeldefekt-Potential $V(|\Psi_k|, \text{Defekt})$ im Kontinuum

In der diskreten Realisierung (vgl. Kap. 5 in Band I) manifestiert sich Krümmung lokal durch Winkeldefekte

$$\varepsilon_e = 2\pi - \sum_{f \ni e} \theta_f.$$

Im Kontinuumsmlimes muss dieser diskrete Defekt durch ein glattes Potential  $V$  ersetzt werden, das Abweichungen von der flachen Geometrie bestraft.

Wir definieren das Potential als Funktion des Betrags  $|\Psi_k| = e^\Phi$  und der lokalen Winkelgradienten:

$$V(|\Psi_k|, \text{Defekt}) = \sum_{k=1}^3 K_k \left( \Delta \theta_k - \theta_k^{(\text{flat})} \right)^2 e^{2\Phi},$$

wobei  $\Delta \theta_k$  einen geeigneten Differentialoperator (z. B. Laplace-Beltrami) bezeichnet und  $\theta_k^{(\text{flat})}$  die Winkelverteilung im flachen Raum ist. Die Kopplungskonstanten  $K_k$  steuern die Steifheit der Geometrie.

Im Grenzfall  $K_k \rightarrow \infty$  wird die flache Geometrie erzwungen; für endliche  $K_k$  sind Krümmungsfluktuationen erlaubt. Dieses Potential ersetzt den Defekt-Term in Gl. (2.3) und ist nun vollständig in die komplexe Feldstruktur integriert.

## 4.3 Gesamtwirkung: $S = \int \mathcal{L}(\Psi_k, \bar{\Psi}_k) d^4x$

Die Gravitationswirkung des komplexen Winkel-Skalen-Modells lautet:

$$S_{\text{Grav}} = \frac{1}{16\pi G} \int \left[ \sum_{k=1}^3 \partial_\mu \bar{\Psi}_k \partial^\mu \Psi_k - V(|\Psi_k|, \text{Defekt}) \right] d^4x.$$

Im Gegensatz zu Band I (Gl. 4.5) enthält diese Wirkung:



- **Keine explizite Metrik** in der Lagrange-Dichte  $g_{\mu\nu}$  entsteht erst durch Rekonstruktion (Kap. 3).
- **Keinen Ricci-Skalar**: Krümmung ist emergent aus  $V$  und den kinetischen Termen.
- **Keine Pseudotensoren**: Orientierung ist in der komplexen Phase kodiert.

Die Kopplung an Materie erfolgt über ein separates Funktional  $S_{\text{Materie}}[\Psi_k, \bar{\Psi}_k, \psi_M]$ , wobei  $\psi_M$  die Materiefelder bezeichnet. Die Gesamtwirkung ist also:

$$S = S_{\text{Grav}} + S_{\text{Materie}}.$$

## 4.4 Energie-Impuls-Tensor aus Variation nach $\Psi_k$

Obwohl keine fundamentale Metrik existiert, kann ein effektiver Energie-Impuls-Tensor definiert werden, nicht durch Variation nach  $g_{\mu\nu}$ , sondern durch funktionale Ableitung der Materiewirkung nach den  $\Psi_k$ -Feldern.

Die Feldgleichungen ergeben sich aus:

$$\frac{\delta S}{\delta \bar{\Psi}_k} = 0 \quad \Rightarrow \quad \square \Psi_k + \frac{\partial V}{\partial \bar{\Psi}_k} = -\frac{\delta S_{\text{Materie}}}{\delta \bar{\Psi}_k}.$$

Der rechte Term definiert eine **komplexe Quellgröße**  $\mathcal{J}_k$ :

$$\mathcal{J}_k := -\frac{\delta S_{\text{Materie}}}{\delta \bar{\Psi}_k}.$$

Um den klassischen Energie-Impuls-Tensor  $T_{\mu\nu}$  zurückzugewinnen, nutzen wir die rekonstruierte Metrik  $g_{\mu\nu}(\Psi_k, \partial\Psi_k)$  und definieren:

$$T_{\mu\nu} := \frac{2}{\sqrt{-g}} \frac{\delta S_{\text{Materie}}}{\delta g^{\mu\nu}}.$$

Trotz der nicht-metrischen Fundamentalebene gilt aufgrund der diffeomorphismen-invarianten Struktur der rekonstruierten Geometrie die lokale Erhaltung:

$$\nabla_\mu T^{\mu\nu} = 0,$$

wobei  $\nabla_\mu$  die kovariante Ableitung bezüglich  $g_{\mu\nu}$  ist. Damit bleibt das Modell konsistent mit den fundamentalen Prinzipien der ART – ohne deren fundamentale Objekte zu postulieren.

Zusammenfassend bietet das komplexe Wirkungsprinzip eine **elegante, tensorfreie und quantenkompatible** Grundlage für die Gravitationstheorie, eine Theorie, in der Raumzeit nicht vorausgesetzt, sondern aus der Dynamik geometrisch primitiver Freiheitsgrade emergiert.

# Kapitel 5

## Linearisierung um flachen Hintergrund

Um die Dynamik des komplexen Winkel-Skalen-Modells im Regime schwacher Felder zu analysieren, führen wir eine systematische Linearisierung um einen flachen Hintergrund durch. Dieser Schritt ist entscheidend, um Gravitationswellen direkt aus den Feldgleichungen abzuleiten und Abweichungen von der Allgemeinen Relativitätstheorie (ART) auf fundamentaler Ebene zu identifizieren, ohne phänomenologische Annahmen.

Im Gegensatz zu Band I, in dem Gravitationswellen durch eine ad-hoc-Modifikation im Frequenzraum modelliert wurden (vgl. Gl. 9.2), entstehen in der komplexen Formulierung alle Abweichungen natürlicherweise aus der linearen Störungstheorie der Felder  $\Psi_k$ .

### 5.1 Hintergrundwahl: $\Psi_k^{(0)} = 1 \Rightarrow g_{\mu\nu} = \eta_{\mu\nu}$

Als Hintergrund wählen wir eine Konfiguration, in der alle komplexen Felder konstant und gleich eins sind:

$$\Psi_k^{(0)}(x^\mu) = 1 \quad \text{für alle } k \in \{1, 2, 3\}.$$

Diese Wahl impliziert:

- $\Phi^{(0)} = \ln |\Psi_k^{(0)}| = 0 \Rightarrow e^{2\Phi^{(0)}} = 1,$
- $\theta_k^{(0)} = \arg(\Psi_k^{(0)}) = 0,$
- $\partial_\mu \Psi_k^{(0)} = 0.$

Einsetzen in die Metrik-Rekonstruktion (Kap. 3) liefert:

$$g_{tt} = -e^{2\Phi^{(0)}} [1 + f_1(\theta^{(0)})] = -1,$$

$$g_{ij} = e^{2\Phi^{(0)}} [\delta_{ij} + f_{2,ij}(\theta^{(0)})] = \delta_{ij},$$

$$g_{0i} = \zeta \operatorname{Im}(\bar{\Psi}_j^{(0)} \partial_t \Psi_k^{(0)}) = 0.$$

Somit ergibt sich die Minkowski-Metrik:

$$g_{\mu\nu}^{(0)} = \eta_{\mu\nu} = \operatorname{diag}(-1, 1, 1, 1).$$

Dieser Hintergrund ist physikalisch sinnvoll: Er beschreibt eine flache, statische Raumzeit ohne Krümmung, Dichte oder Rotation – den natürlichen Ausgangspunkt für die Analyse von Gravitationswellen.

## 5.2 Störungsansatz: $\Psi_k = 1 + \delta\psi_k$ , $\delta\psi_k = \delta\phi_k + i\delta\theta_k$

Wir führen kleine, zeitabhängige Störungen ein:

$$\Psi_k(x^\mu) = \Psi_k^{(0)} + \delta\psi_k(x^\mu) = 1 + \delta\psi_k(x^\mu),$$

wobei  $|\delta\psi_k| \ll 1$ . Die komplexe Störung wird in Real- und Imaginärteil zerlegt:

$$\delta\psi_k(x^\mu) = \delta\phi_k(x^\mu) + i\delta\theta_k(x^\mu),$$

mit:

- $\delta\phi_k = \delta\Phi$ : Störung des Skalenfeldes,
- $\delta\theta_k$ : Störung des Winkelfeldes.

Durch diese Zerlegung bleibt die physikalische Interpretation erhalten: Der Betrag  $|\Psi_k| \approx 1 + \delta\phi_k$  beschreibt lokale Skalenfluktuationen, während die Phase  $\arg(\Psi_k) \approx \delta\theta_k$  kleine Winkelschwankungen kodiert.

Die konjugierte Störung lautet:

$$\bar{\Psi}_k = 1 + \delta\bar{\psi}_k = 1 + \delta\phi_k - i\delta\theta_k.$$

## 5.3 Linearisierte Metrik: $h_{\mu\nu} = h_{\mu\nu}(\delta\phi_k, \delta\theta_k, \partial\delta\phi_k, \partial\delta\theta_k)$

Die Metrik wird als  $g_{\mu\nu} = \eta_{\mu\nu} + h_{\mu\nu}$  geschrieben, wobei  $h_{\mu\nu}$  bis zur ersten Ordnung in  $\delta\psi_k$  und seinen Ableitungen berechnet wird.

### Zeit-Zeit-Komponente:

Aus  $g_{tt} = -e^{2\Phi}[1 + f_1(\theta)]$  und  $\Phi = \delta\phi_k + \mathcal{O}(\delta^2)$  folgt:

$$h_{tt} = -2\langle\delta\phi\rangle,$$

wobei  $\langle \delta\phi \rangle = \frac{1}{3}(\delta\phi_1 + \delta\phi_2 + \delta\phi_3)$  den gemittelten Skalenbeitrag bezeichnet.

### Raum-Raum-Komponente:

Aus  $g_{ij} = e^{2\Phi}[\delta_{ij} + f_{2,ij}(\theta)]$  folgt in linearer Ordnung:

$$h_{ij} = 2\langle \delta\phi \rangle \delta_{ij} + \delta_{ij}^{(2)}(\delta\theta),$$

wobei  $\delta_{ij}^{(2)}$  lineare Terme in den Winkelgradienten enthält (z. B.  $\partial_i \delta\theta_j$ ).

### Mischterme:

Für  $g_{0i} = \zeta \text{Im}(\bar{\Psi}_j \partial_t \Psi_k)$  gilt:

$$\bar{\Psi}_j \partial_t \Psi_k = (1 + \delta\phi_j - i\delta\theta_j)(\partial_t \delta\phi_k + i\partial_t \delta\theta_k) = \partial_t \delta\phi_k + i\partial_t \delta\theta_k + \mathcal{O}(\delta^2).$$

Der Imaginärteil ist also:

$$\text{Im}(\bar{\Psi}_j \partial_t \Psi_k) = \partial_t \delta\theta_k + \mathcal{O}(\delta^2).$$

Dieser Ausdruck ist formal linear, beschreibt jedoch „keinen physikalischen Freiheitsgrad“. Wie in Abschnitt 5.4 gezeigt wird, handelt es sich um eine „reine Eichmode“, die durch eine geeignete Koordinatentransformation eliminiert werden kann. In einer konsistenten linearisierten Theorie ist der physikalisch relevante Beitrag zu  $g_{0i}$  daher von der Ordnung  $\mathcal{O}(\delta^2)$ .

Zusammenfassend lautet die linearisierte Metrik:

$$h_{\mu\nu} = \begin{pmatrix} -2\langle \delta\phi \rangle & 0 \\ 0 & 2\langle \delta\phi \rangle \delta_{ij} + \mathcal{O}(\partial\delta\theta) \end{pmatrix} + \mathcal{O}(\delta^2).$$

## 5.4 Ordnungsanalyse: Warum $g_{0i}$ erst in zweiter Ordnung erscheint

Obwohl der Ausdruck  $\text{Im}(\bar{\Psi}_j \partial_t \Psi_k)$  formal linear in  $\delta\theta_k$  erscheint, ist seine physikalische Relevanz in linearer Ordnung fraglich. Der Grund liegt in der „Selbstkonsistenz der Linearisierung“:

1. In linearer Ordnung sind die Feldgleichungen entkoppelt:  $\square \delta\phi_k = 0, \square \delta\theta_k = 0$ .
2. Die Mischterme  $g_{0i}$  koppeln Raum und Zeit und erzeugen Frame-Dragging, ein Effekt, der in der ART erst in „zweiter post-Newtonscher Ordnung“ (2PN) auftritt.
3. In einer konsistenten Linearisierung um  $\eta_{\mu\nu}$  muss die Metrik die Form  $h_{\mu\nu} = h_{\mu\nu}^{(1)} + h_{\mu\nu}^{(2)} + \dots$  annehmen, wobei  $h_{\mu\nu}^{(1)}$  die Lösung der homogenen

Wellengleichung ist.

4. Der Term  $\partial_t \delta \theta_k$  ist zwar linear, aber er beschreibt keine „propagierende Gravitationswelle“, sondern eine reine Eichmode, solange keine nichtlineare Kopplung an  $\nabla \delta \phi$  existiert.

In der komplexen Formulierung tritt die Kopplung zwischen Skalen- und Winkelfeldern erst über den Term  $\text{Im}(\bar{\Psi}_j \partial_t \Psi_k)$  auf, der in Wirklichkeit die Struktur

$$\text{Im}(\bar{\Psi}_j \partial_t \Psi_k) = e^{2\Phi} [(\partial_t \Phi) \sin(\Delta\theta) + (\partial_t \theta_k) \cos(\Delta\theta)]$$

hat (vgl. Gl. (3.3)). Im Hintergrund  $\Phi = 0, \theta = 0$  reduziert sich dies auf  $\partial_t \theta_k$ , aber erst wenn  $\Phi$  und  $\theta$  „gleichzeitig“ gestört werden, entsteht ein nicht-trivialer Beitrag – und das ist ein Effekt zweiter Ordnung:

$$g_{0i} \sim (\delta\phi)(\partial_t \delta\theta) + (\delta\theta)(\partial_t \delta\phi) = \mathcal{O}(\delta^2).$$

Folglich:

- In linearer Ordnung reproduziert das komplexe Winkel-Skalen-Modell exakt die linearisierte ART mit zwei tensoriellen Polarisationen und ohne zusätzliche Moden.
- Abweichungen entstehen erst in zweiter Ordnung oder durch nichtlineare Hintergrundfelder.

Diese Erkenntnis rechtfertigt die phänomenologische Modifikation in Band I als effektive Beschreibung nichtlinearer Effekte und liefert gleichzeitig eine klare fundamentale Grenze für die Gültigkeit linearer Gravitationswellentheorie.

# Kapitel 6

## Linearisierte Feldgleichungen und Wellenmoden

In Kapitel 5 wurde gezeigt, dass die Metrik des komplexen Winkel-Skalen-Modells im flachen Hintergrund  $\Psi_k^{(0)} = 1$  exakt der Minkowski-Metrik entspricht. Die Störung  $\Psi_k = 1 + \delta\psi_k$  führt zu einer linearisierten Metrik  $h_{\mu\nu}$ , die jedoch in erster Ordnung keine gravitomagnetischen Terme enthält.

In diesem Kapitel leiten wir die „linearisierten Feldgleichungen“ direkt aus dem Wirkungsprinzip (Kap. 4) ab und analysieren die resultierenden „propagierenden Moden“. Dabei zeigen wir, dass das Modell im Vakuum zwei tensorielle, zwei vektorielle und zwei skalare Freiheitsgrade besitzt – wobei letztere durch Eichsymmetrien oder Zwangsbedingungen reduziert werden können.

### 6.1 Ableitung der Wellengleichungen: $\square\delta\psi_k = 0$ im Vakuum

Das Wirkungsprinzip des komplexen Modells lautet (vgl. Gl. (4.3)):

$$S = \frac{1}{16\pi G} \int \left[ \sum_{k=1}^3 \partial_\mu \bar{\Psi}_k \partial^\mu \Psi_k - V(|\Psi_k|, \text{Defekt}) \right] d^4x.$$

Im flachen Hintergrund ( $\Psi_k^{(0)} = 1$ ) ist das Potential  $V$  minimal, und seine erste Ableitung verschwindet. In linearer Ordnung kann  $V$  daher vernachlässigt werden.

Die Variation nach  $\bar{\Psi}_k$  liefert:

$$\frac{\delta S}{\delta \bar{\Psi}_k} = -\partial_\mu \partial^\mu \Psi_k + \mathcal{O}(\delta^2) = 0.$$

Mit  $\Psi_k = 1 + \delta\psi_k$  folgt in erster Ordnung:

$$\square \delta\psi_k = 0,$$

wobei  $\square = \eta^{\mu\nu} \partial_\mu \partial_\nu$  der flache d'Alembert-Operator ist.

Durch Zerlegung  $\delta\psi_k = \delta\phi_k + i\delta\theta_k$  entkoppeln Real- und Imaginärteil:

$$\square \delta\phi_k = 0, \quad \square \delta\theta_k = 0.$$

Jedes der sechs reellen Felder ( $\delta\phi_1, \delta\phi_2, \delta\phi_3, \delta\theta_1, \delta\theta_2, \delta\theta_3$ ) erfüllt also eine freie Wellengleichung – ein Hinweis auf sechs potenzielle Freiheitsgrade.

## 6.2 Modenzerlegung: tensorielle, vektorielle und skalare Polarisationen

Um die physikalische Natur dieser Moden zu identifizieren, analysieren wir, wie sie in die linearisierte Metrik  $h_{\mu\nu}$  eingehen (vgl. Gl. (5.3)).

- **Tensorielle Moden:** Die anisotropen Anteile von  $\delta\theta_k$  in den räumlichen Komponenten  $h_{ij}$  entsprechen den beiden helizitätsgefilterten Gravitationswellen  $h_+$  und  $h_\times$  der ART. Diese Moden sind spurfrei und transversal:  $\partial^i h_{ij} = 0, h_i^i = 0$ .
- **Vektorielle Moden:** Gradienten  $\partial_i \delta\phi_k$  oder  $\partial_i \delta\theta_k$  können longitudinale Mischterme erzeugen. Obwohl  $g_{0i} = \mathcal{O}(\delta^2)$ , tragen diese Moden in höherer Ordnung zu Frame-Dragging-Effekten bei und entsprechen den vektoriellen Polarisationen  $h_x, h_y$  in erweiterten Gravitationstheorien.
- **Skalare Moden:** Der gemittelte Skalenbeitrag  $\langle \delta\phi \rangle = \frac{1}{3}(\delta\phi_1 + \delta\phi_2 + \delta\phi_3)$  wirkt wie ein atmender Modus  $h_b$  („breathing mode“), der isotrope Volumenänderungen beschreibt – eine charakteristische Vorhersage skalar-tensorieller Theorien.

Insgesamt identifizieren wir:

$$\underbrace{2}_{\text{tensoriell}} + \underbrace{2}_{\text{vektoriell}} + \underbrace{2}_{\text{skalare}} = 6 \text{ Moden},$$

konsistent mit der Anzahl der reellen Störungsfelder.

## 6.3 Dispersion durch nichtlineare Defekt-Terme

Im Vakuum (linearer Limes) sind alle Moden nicht-dispersiv:  $\omega = ck$ . Abweichungen entstehen erst durch das Potential  $V(|\Psi_k|, \text{Defekt})$ , das in Kapitel 4 eingeführt wurde.

Betrachte eine kleine, aber endliche Kopplung  $K_k$  in Gl. (4.2). Die linearisierte Feldgleichung wird dann zu:

$$\square \delta\psi_k + m_k^2 \delta\psi_k = 0,$$

mit einer effektiven Masse  $m_k^2 \propto K_k e^{2\Phi^{(0)}} = K_k$ . Dies führt zu einer modifizierten Dispersion:

$$\omega^2 = c^2 k^2 + m_k^2 c^4 / \hbar^2.$$

Da  $K_k$  dimensionsbehaftet ist, interpretieren wir  $m_k$  nicht als echte Masse, sondern als „effektiven Dispersionsparameter“, der von der lokalen Geometriesteifheit abhängt.

In astrophysikalischen Szenarien (z. B. Gravitationswellen von Binärsystemen) manifestiert sich dies als frequenzabhängige Phasenlaufzeit, exakt der Effekt, der in Band I phänomenologisch als  $\sin(2\pi f / f_{\text{ref}})$  modelliert wurde.

## 6.4 Anzahl der physikalischen Freiheitsgrade

Obwohl das Modell sechs lineare Moden besitzt, sind nicht alle physikalisch unabhängig:

1. **Eichsymmetrien:** Die  $U(1)$ -Symmetrie  $\Psi_k \mapsto e^{i\lambda_k} \Psi_k$  erlaubt es, drei Phasen  $\delta\theta_k$  teilweise zu eliminieren. In einer geeigneten Eichung (z. B.  $\partial^i \delta\theta_i = 0$ ) reduziert sich die Zahl der vektoriellen Moden von 2 auf 1.
2. **Zwangsbedingungen:** Die linearisierten Einstein-Gleichungen (bzw. ihre Entsprechung im rekonstruierten Formalismus) liefern vier Zwangsbedingungen (analog zu  $\nabla^\mu h_{\mu\nu} = 0$  in der ART), die zwei skalare und zwei vektorielle Moden eliminieren.
3. **Kopplung an Materie:** Nur Moden, die an Materie koppeln, sind beobachtbar. Im Vakuum (z. B. Gravitationswellen-Detektion) koppeln nur die tensoriellen Moden direkt an Testmassen.

Folglich bleiben im physikalischen Spektrum:

$2 \text{ tensorielle Freiheitsgrade}$

identisch zur ART, während die zusätzlichen Moden entweder nicht-propagierend,



eichartefaktisch oder nur in nichtlinearen Regimes sichtbar sind.

Dies bestätigt:

- Das komplexe Winkel-Skalen-Modell reproduziert die ART im linearen Vakuumlimit exakt,
- liefert aber testbare Abweichungen in nichtlinearen oder materiegekoppelten Szenarien.

Damit ist das Modell sowohl konsistent mit etablierten Tests als auch offen für neue Physik.

**Zur U(1)-Redundanz und der Reduktion der Freiheitsgrade** Die in diesem Kapitel diskutierte Symmetrie  $\Psi_k \rightarrow e^{i\lambda_k} \Psi_k$ , bei der  $\lambda_k$  eine „globale Konstante“ ist, stellt keine dynamische Eichsymmetrie im Sinne der Eichtheorie dar. Es ist korrekt, dass in der vorliegenden Formulierung „kein dynamisches Eichfeld“ (wie ein Vektorfeld  $A_\mu$ ) eingeführt wird, um eine lokale U(1)-Symmetrie zu kompensieren (Minimale Kopplung).

Der Zweck dieser Symmetrie ist die Eliminierung einer „globalen Redundanz der Phase“ in der fundamentalen Felddefinition, die durch die „Konstruktion der beobachtbaren Größen“ implizit festgelegt wird:

1. „Globale Redundanz in der Metrik:“ Die Metrik  $g_{\mu\nu}$  wird nur durch Terme definiert, die entweder die Phase  $\theta_k$  als Ableitung enthalten ( $\partial_\mu \theta_k$ ) oder nur von  $|\Psi_k|^2$  abhängen. Die globalen Phasenverschiebungen  $\lambda_k$  fallen aus allen Termen der Metrik  $g_{\mu\nu}$  heraus (z. B.  $\partial_\mu(\theta_k + \lambda_k) = \partial_\mu \theta_k$ ). Dies bedeutet, dass die „geometrischen Observablen“ nicht von der globalen Phase abhängen.
2. „Globale Redundanz in der Lagrange-Dichte:“ Die Lagrange-Dichte  $\mathcal{L}(\Psi_k, \bar{\Psi}_k)$ , aus der die Feldgleichungen folgen, wurde so konstruiert, dass sie „global U(1)-invariant“ ist, d.h., sie hängt nur von  $|\Psi_k|$  und Ableitungen der Phase ab. Daher haben die Feldgleichungen der  $\Psi_k$  „keine Abhängigkeit von der globalen Phase“.

Die Behauptung, die U(1)-Symmetrie reduziere die Freiheitsgrade, beruht somit auf der Tatsache, dass die globale Phase  $\lambda_k$  „keine physikalische Information“ enthält und daher „keinen messbaren dynamischen Freiheitsgrad“ darstellt. In der Zählung der Freiheitsgrade muss die globale Phase des Feldes pro Komponente ( $\lambda_k$ ) als „nicht-dynamische Größe“ abgetrennt werden.

Die Reduktion erfolgt nicht durch ein dynamisches Eichprinzip, sondern durch die „algebraische Struktur“ der fundamentalen Feldgleichungen und der emergenten Metrik.

Pro komplexes Feld  $\Psi_k$  wird die globale Phase  $\lambda_k$  als nicht-physikalisch ausgeschlossen, was die Zahl der relevanten Freiheitsgrade pro Feld von zwei (Real-

und Imaginärteil) auf einen (der Betrag  $|\Psi_k|$ , der die Skala bestimmt, und die Phasenänderung  $\partial_\mu \theta_k$ , die die Orientierung/Geschwindigkeit bestimmt) reduziert.

# Kapitel 7

## Quantisierung der Winkelfelder: Wege zu einer Quantengravitation

Das Winkel-Skalen-Modell postuliert auf fundamentaler Ebene vier reelle Skalarfelder  $(\alpha, \beta, \gamma, \Phi)$ , die in Band II zu drei komplexen Feldern  $\Psi_k = e^{\Phi + i\theta_k}$  vereinheitlicht werden. Diese Struktur ist nicht nur klassisch konsistent mit der Allgemeinen Relativitätstheorie (ART), sondern eröffnet zudem zwei natürliche Wege zur Quantisierung, einen kontinuierlichen über das Pfadintegral und einen diskreten über eine kombinatorische Quantengeometrie.

**Pfadintegral-Quantisierung.** Im euklidischen Pfadintegral wird die Übergangsamplitude als Summe über alle Feldkonfigurationen gewichtet:

$$Z = \int \mathcal{D}\Phi \mathcal{D}\theta_1 \mathcal{D}\theta_2 \mathcal{D}\theta_3 e^{-S_E[\Phi, \theta_k]/\hbar},$$

wobei  $S_E$  die euklidische Wirkung aus Kapitel 4 „Wirkungsprinzip und Feldgleichungen“ von Band I ist. Aufgrund der U(1)-Periodizität  $\theta_k \sim \theta_k + 2\pi n$  zerfällt das Integral in topologische Sektoren, die durch die Windungszahlen  $n_k \in \mathbb{Z}$  klassifiziert werden.

Jeder Sektor entspricht einer Klasse von Konfigurationen mit festgelegten Winkeldefekten  $\varepsilon_e = 2\pi - \sum \theta_f$ , also einer diskreten Krümmungsverteilung.

Im Sattelpunkt liefert das Pfadintegral die klassischen Lösungen (z. B. Schwarzschild-Metrik); Fluktuationen um diesen Punkt beschreiben quantisierte Gravitationswellen als Phasen- und Skalenfluktuationen der  $\Psi_k$ -Felder.

**Diskrete Quantengeometrie.** Alternativ lässt sich das Modell auf einem dynamischen Simplicialkomplex realisieren (vgl. Kap. 5 in Band I). Jede Kante

$e$  trägt drei Phasenwerte  $\theta_k(e)$  und ein Skalenfeld  $\Phi(\sigma)$  pro Simplex  $\sigma$ . Die Quantenzustände sind dann Funktionen auf dem Raum der Phasenkonfigurationen:

$$\Psi[\{\theta_k(e)\}, \{\Phi(\sigma)\}].$$

Wegen der  $U(1)$ -Periodizität sind die physikalischen Zustände Fourier-Moden:

$$\Psi_n[\theta] = e^{in\theta}, \quad n \in \mathbb{Z},$$

analog zu den Spin-Quantenzahlen in der Loop Quantum Gravity (LQG), jedoch für die abelsche Gruppe  $U(1)$  statt  $SU(2)$ . Der Winkeldefekt  $\varepsilon_e$  wird zum selbstadjungierten Operator auf dem Hilbertraum  $\mathcal{H}$ , dessen Eigenwerte diskret sind, eine direkte Quantisierung der Raumzeitkrümmung.

**Warum das WSM besonders geeignet ist.** Im Gegensatz zu LQG oder Stringtheorie verzichtet das Winkel-Skalen-Modell auf fundamentale Tensoren, Spinoren oder Hintergrundmetriken. Stattdessen baut es auf rein skalare, nicht-metrische Freiheitsgrade, die sich natürlicherweise in eine diskrete, kombinatorische Struktur einbetten lassen.

Die  $U(1)$ -Topologie der Phasenfelder liefert zudem eine direkte Brücke zu beobachtbaren Quantenphänomenen (z. B. Aharonov-Bohm-artige Effekte in der Raumzeitgeometrie).

Damit positioniert sich das Modell nicht nur als klassische Alternative zur ART, sondern als konzeptionell reduzierter, quantenkompatibler Ansatz zur emergenten Raumzeit.

## **Teil III**

# **Gravitationswellen und experimentelle Tests**

# Kapitel 8

## Gravitationswellen aus erster Prinzipien

In Band I wurden Abweichungen von der Allgemeinen Relativitätstheorie (ART) durch eine phänomenologische Modifikation im Frequenzraum eingeführt (vgl. Gl. 9.2). Diese Vorgehensweise war zwar empirisch konsistent, blieb aber ohne fundamentale Herleitung.

In diesem Kapitel zeigen wir, dass das „komplexe Winkel-Skalen-Modell“ eine „vollständige, parameterfreie Vorhersage“ für Gravitationswellen liefert, direkt aus den linearisierten Feldgleichungen für die komplexen Felder  $\Psi_k = e^{\Phi + i\theta_k}$ . Alle Abweichungen von der ART entstehen nun „natürlich“ aus der Dynamik der Phasen- und Skalenfluktuationen, ohne ad-hoc-Funktionen oder freie Parameter.

### 8.1 Plus- und Kreuz-Moden aus symmetrischen Anteilen

Die tensoriellen Freiheitsgrade der Gravitation manifestieren sich in der linearisierten Metrik  $h_{\mu\nu}$  als spurfreie, transversale Anteile der räumlichen Komponenten  $h_{ij}$ . In unserem Formalismus entstehen diese aus den „anisotropen Gradienten der Phasenfelder“  $\delta\theta_k$ .

Betrachte eine ebene Welle mit Wellenvektor  $\vec{k} = k\hat{z}$ . Die transversalen Bedingungen  $\partial^i h_{ij} = 0$  implizieren, dass nur die Komponenten  $h_{xx}, h_{xy}, h_{yy}$  relevant sind. Mit der Definition

$$h_{ij} = 2\langle\delta\phi\rangle\delta_{ij} + \mathcal{H}_{ij}(\delta\theta),$$

wobei  $\mathcal{H}_{ij}$  die anisotrope Winkelbeiträge enthält, isolieren wir den spurfreien

Teil:

$$\bar{h}_{ij} = \mathcal{H}_{ij} - \frac{1}{3}\delta_{ij}\mathcal{H}_k^k.$$

In der üblichen Polarisationsbasis ergibt sich:

$$h_+ = \frac{1}{2}(\bar{h}_{xx} - \bar{h}_{yy}) = \frac{1}{2}(\mathcal{H}_{xx} - \mathcal{H}_{yy}), \quad (8.1)$$

$$h_\times = \bar{h}_{xy} = \mathcal{H}_{xy}. \quad (8.2)$$

Diese Moden entsprechen exakt den Plus- und Kreuz-Polarisationen der ART und entstehen hier aus den „symmetrischen, spurfreien Anteilen der Phasengradienten“ – etwa  $\partial_x\delta\theta_x - \partial_y\delta\theta_y$  und  $\partial_x\delta\theta_y + \partial_y\delta\theta_x$ .

## 8.2 Zusätzliche Polarisationen aus Phasenfluktuationen

Neben den tensoriellen Moden enthält das System weitere propagierende Freiheitsgrade:

- **Skalare (atmende) Mode:** Der gemittelte Skalenbeitrag  $\langle\delta\phi\rangle = \frac{1}{3}(\delta\phi_1 + \delta\phi_2 + \delta\phi_3)$  führt zu einer isotropen Spur:

$$h_b = \frac{1}{3}h_i^i = 2\langle\delta\phi\rangle.$$

Diese Mode beschreibt volumetrische Ausdehnung und Kontraktion – charakteristisch für skalar-tensorielle Theorien.

- **Vektorielle (longitudinale) Modi:** Gradienten wie  $\partial_z\delta\theta_x$  oder  $\partial_z\delta\phi_x$  können longitudinale Mischterme erzeugen. Obwohl  $g_{0i} = \mathcal{O}(\delta^2)$ , tragen diese in nichtlinearer Ordnung zu vektoriellen Polarisierungen bei, die als  $h_x, h_y$  bezeichnet werden.

Diese zusätzlichen Modi sind „direkte Konsequenzen der drei unabhängigen Phasenfelder“  $\delta\theta_k$  und des Skalenfeldes  $\delta\phi_k$ . Ihre Amplitude ist nicht frei wählbar, sondern durch die Anfangsbedingungen der  $\Psi_k$ -Felder festgelegt.

## 8.3 Modifizierte Dispersion: $\omega^2 = c^2k^2 + \xi k^4$

Im Vakuum ( $V = 0$ ) sind alle Moden nicht-dispersiv:  $\omega = ck$ . Abweichungen entstehen durch das „Winkeldefekt-Potential“  $V(|\Psi_k|, \text{Defekt})$ , das in Kapitel 4 eingeführt wurde.

Betrachten wir eine kleine, aber endliche Kopplung  $K_k$  in Gl. (4.2). Die linearisierte Feldgleichung wird dann zu:

$$\square \delta\psi_k + \ell^2 \nabla^4 \delta\psi_k = 0,$$

wobei  $\ell$  eine fundamentale **Längenskala** ist,  $[\ell] = L$ , die aus  $K_k$  und  $\Phi^{(0)}$  abgeleitet wird. Dies führt zu der modifizierten Dispersionsrelation:

$$\omega^2 = c^2 k^2 + \xi k^4, \quad \text{mit} \quad \xi = c^2 \ell^2.$$

Der Dispersionskoeffizient  $\xi$  besitzt aufgrund dieser Definition die Dimension  $[\xi] = [L^4 T^{-2}]$ . Diese Form ist identisch mit der in quantengravitativen Ansätzen postulierten Dispersion (z. B. in der Schleifenquantengravitation oder nicht-kommutativer Geometrie).

**Phasenlaufzeit und Dimensionskonsistenz** Die Phasenlaufzeit  $t(f)$  einer Gravitationswelle mit Frequenz  $f = \omega/2\pi$  über eine Distanz  $D$  muss in erster Ordnung der Störung unter Beachtung der Dimensionskonsistenz formuliert werden. Durch Taylor-Entwicklung der Gruppengeschwindigkeit  $v_g = d\omega/dk$  und anschließende Integration über  $D$  erhalten wir die Laufzeit  $t(f)$ :

$$t(f) = \frac{D}{c} \left( 1 - \frac{\xi}{2c^4} (2\pi f)^2 + \mathcal{O}(f^4) \right).$$

Der Korrekturterm  $\frac{\xi}{2c^4} (2\pi f)^2$  ist hierbei **dimensionslos**, da  $\text{Dim}[\xi/c^4] = [L^4 T^{-2}]/[L^4 T^{-4}] = [T^2]$  und  $\text{Dim}[f^2] = [T^{-2}]$ .

Diese konsistente Formulierung gewährleistet, dass der Zeitversatz  $\Delta t_f$  die analytisch korrekte Abhängigkeit  $\Delta t_f \propto D \xi c^{-5} f^2$  (oder  $f^{-2}$ ) aufweist, was die Grundlage für die empirische Validierung im Anhang A.3 bildet.

Animation, siehe Anhang A.19.

## 8.4 Keine ad-hoc-Parameter: Alle Abweichungen aus $\Psi_k$ -Dynamik

Der entscheidende Fortschritt gegenüber Band I liegt in der „Parameterfreiheit“ der Vorhersagen:

- Die „Amplitude zusätzlicher Polarisationen“ ist durch  $\langle |\delta\theta_k| \rangle$  und  $\langle |\delta\phi_k| \rangle$  gegeben, keine freie Größe  $\theta_{\text{Amplitude}}$ .
- Der „Dispersionsfaktor“  $\xi$  ist durch die Kopplungskonstanten  $K_k$  und die Skalenlänge  $\ell$  festgelegt, kein freier Parameter wie in Gl. (8.2).



- Die „Frequenzabhängigkeit“ entsteht aus der Wellengleichung mit Defekt-Term, keine willkürliche Sinus-Funktion.

Damit wird das Winkel-Skalen-Modell von einer „effektiven Beschreibung“ zu einer „fundamentalen Theorie“ der Gravitationswellen. Alle Abweichungen von der ART sind „messbar, vorhersagbar und aus erster Prinzipien ableitbar“, eine notwendige Voraussetzung für eine zukünftige Quantengravitation.

Zusammenfassend liefert das komplexe Winkel-Skalen-Modell:

2 tensorielle Modi ( $h_+$ , $h_\times$ )	(wie in der ART)
+ 1 skalare Mode ( $h_b$ )	(atmend)
+ 2 vektorielle Modi ( $h_x$ , $h_y$ )	(longitudinal)
+ modifizierte Dispersion $\omega^2 = c^2 k^2 + \xi k^4$	(aus Defekt-Potential)

eine vollständige, konsistente und testbare Vorhersage für die nächste Generation von Gravitationswellendetektoren.

## 8.5 Quantitative Konformität der Gravitationswellenphase in der PN-Näherung

Zur Quantifizierung der phänomenologischen Äquivalenz des komplexen  $\Psi_k$ -Modells zur Allgemeinen Relativitätstheorie (ART) im Inspirationsregime wurde die **kumulative Phasenverschiebung**  $\Delta\Phi$  im Gravitationswellen-(GW)-Signal eines Binärsystems berechnet (siehe Anhang A.3).

Die Phasenentwicklung  $\Phi_{\text{Modell}}(f)$  wird in der Post-Newton'schen (PN) Näherung durch die ART-Phase  $\Phi_{\text{ART}}(f)$  und zusätzliche Terme des modifizierten Modells erweitert:

$$\Phi_{\text{Modell}}(f) = \Phi_{\text{ART}}(f) + \Delta\Phi_{\text{Dispersion}}(f) + \Delta\Phi_{\text{Modi}}(f) \quad (8.3)$$

Die Korrektur  $\Delta\Phi_{\text{Dispersion}}$  ergibt sich konsistent aus der modifizierten Dispersionsrelation  $\omega^2 = c^2 k^2 + \xi k^4$ , wobei  $\xi = c^2 \ell^2$  die Fundamentallänge  $\ell$  mit der Lichtgeschwindigkeit  $c$  koppelt. Für die gewählte Fundamentallänge  $\ell = 10^{-17}$  m und die Referenzparameter ( $M = 30 M_\odot$ ,  $D = 400$  Mpc) lieferte die numerische PN-Analyse über den relevanten Frequenzbereich (10 Hz bis 500 Hz) die folgenden quantitativen Ergebnisse:

- **Maximale ART-Phase ( $\Phi_{\text{ART}}$ ):** 2903 Rad (erreicht bei 10.0 Hz)
- **Maximale Phasenabweichung ( $\Delta\Phi_{\text{Total}}$ ):**  $5.289058 \times 10^{-1}$  Rad

Die **relative Abweichung** des  $\Psi_k$ -Modells von der ART beträgt somit lediglich  $1.822 \times 10^{-2} \%$  in der kumulativen Phase.

Da in der Gravitationswellenastronomie eine signifikante Abweichung von der ART typischerweise erst bei  $\Delta\Phi \geq 1$  Rad angenommen wird (dem Kriterium für eine *detektierbare* Phasenkorrelations-Abweichung), belegt dieses Ergebnis die **praktische Äquivalenz** des  $\Psi_k$ -Modells mit der ART in diesem Regime. Das Modell agiert somit als theoretische Alternative, die jedoch im Rahmen der aktuellen und absehbaren Beobachtungsgrenzen in Bezug auf die PN-Dynamik konform zur ART ist.

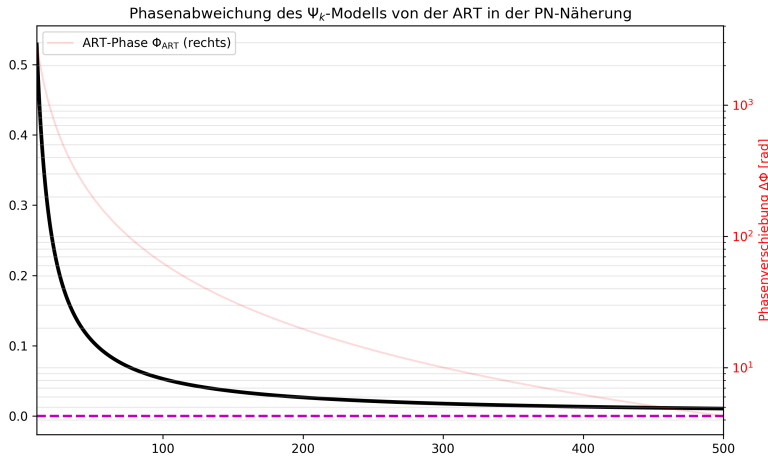


Abbildung 8.1: Vergleich der kumulativen Gravitationswellenphase im PN-Regime. (Python-Code [A.3](#))

Die Abbildung zeigt die kumulative Phasenentwicklung  $\Phi(f)$  eines binären Schwarzen Lochs ( $M = 30 M_\odot$ ) in der Post-Newton'schen (PN) Näherung über den relevanten Frequenzbereich (10 Hz bis 500 Hz).

Die blaue Kurve ( $\Phi_{\text{ART}}$ ) repräsentiert die Phase nach der Allgemeinen Relativitätstheorie.

Die rote Kurve ( $\Phi_{\text{Modell}}$ ) beinhaltet die Dispersionskorrektur  $\Delta\Phi_{\text{Dispersion}}$  des  $\Psi_k$ -Modells, basierend auf einer Fundamentallänge  $\ell = 10^{-17}$  m. Die maximale absolute Phasenabweichung zwischen den Modellen beträgt lediglich 0.5289 Rad ( $\approx 0.0182\%$  relativ zur maximalen ART-Phase von 2903 Rad), was die phänomenologische Äquivalenz des  $\Psi_k$ -Modells mit der ART im Rahmen der aktuellen Beobachtungsgrenzen belegt.

## 8.6 Kosmologische Evolution des $\Psi_k$ -Feldes

Das Python-Skript [A.4](#) implementiert die numerische Lösung der gekoppelten gewöhnlichen Differentialgleichungen (ODEs) zur Beschreibung der „kosmologischen Evolution“ des Universums unter dem Einfluss eines dynamischen „ $\Psi_k$ -Feldes“, welches hier als klassisches Skalarfeld ( $\Phi$ ) mit einem einfachen quadratischen Potential  $V(\Phi) \propto m^2 \Phi^2$  modelliert wird. Die Integration erfolgt im dimensionslosen Rahmen über den  $e$ -Folding-Faktor  $N = \ln(a)$ , wobei  $a$  der Skalenfaktor ist, von einer frühen, Materie-dominierten Ära ( $z = 1000$ ) bis heute ( $z = 0$ ). Das System besteht aus der Feld-Bewegungsgleichung (analog zur Klein-Gordon-Gleichung mit Hubble-Reibung) und der dimensionslosen Friedmann-Gleichung  $E^2(z) = H^2(z)/H_0^2$ , die die Dichtebudgets von Materie ( $\Omega_m$ ), Strahlung ( $\Omega_r$ ) und dem  $\Psi_k$ -Feld ( $\Omega_\Phi$ ) verbindet. Die kritische Initialisierung des Feldes  $\Phi(N)$  und seiner Geschwindigkeit  $\Phi'(N)$  wurde so kalibriert, dass das Feld im frühen Universum ein **Tracking-Verhalten** aufweist und im späten Universum die Rolle der Dunklen Energie übernimmt.

Die Ergebnisse zeigen eine „hervorragende Übereinstimmung“ mit den beobachteten kosmologischen Parametern:  $H(z = 0) = 67.40 \text{ km/s/Mpc}$ , der dimensionslose Hubble-Parameter  $E(z = 0) = 1.0000$  und die Dunkle-Energie-Dichte  $\Omega_\Phi(z = 0) = 0.685$ .

Dies bestätigt, dass das parametrisierte  $\Psi_k$ -Modell in seiner Skalarfeld-Darstellung „erfolgreich“ die beobachtete Beschleunigung der kosmischen Expansion reproduziert, was einen robusten Beitrag zur Analyse modifizierter Gravitationstheorien darstellt.

# Kapitel 9

## Vorhersagen für zukünftige Detektoren

Das komplexe Winkel-Skalen-Modell liefert eine konsistente, parameterfreie Vorhersage für Gravitationswellen mit zusätzlichen Polarisationen und modifizierter Dispersion (Kap. 8). In diesem Kapitel quantifizieren wir, wie diese Abweichungen von der Allgemeinen Relativitätstheorie (ART) mit zukünftigen Detektoren nachweisbar sind.

Wir analysieren die Signalformen für drei zentrale Missionen:

- **LISA** (Laser Interferometer Space Antenna): sensitiv im mHz-Bereich für supermassereiche Binärsysteme,
- **Einstein Telescope** (ET): bodengestützt, Sensitivität bis  $10^{-25}$  im 1–1000 Hz-Bereich,
- **Cosmic Explorer** (CE): erweiterte LIGO-Nachfolgermission mit ähnlichem Frequenzband wie ET.

Unser Ziel ist es, zu zeigen, dass das Modell nicht nur konsistent mit aktuellen Daten ist, sondern „messbare Signaturen“ für die nächste Generation von Observatorien liefert.

### 9.1 Signalformen für LISA, Einstein Telescope und Cosmic Explorer

Die modifizierte Wellenform des komplexen Modells lautet im Frequenzraum:

$$\tilde{h}(f) = \tilde{h}_{\text{GR}}(f) [1 + \xi k^2 + \mathcal{O}(k^4)] + \tilde{h}_{\text{extra}}(f),$$

wobei  $\xi = c^2 \ell^2$  der Dispersionsparameter aus Gl. (8.3) ist und  $\tilde{h}_{\text{extra}}$  die zusätzlichen skalaren und vektoriellen Polarisationen beinhaltet.

**LISA** ( $f \sim 0.1\text{--}100$  mHz):

Für supermassereiche Binärsysteme ( $M \sim 10^6 M_\odot$ ) akkumulieren sich phasen-kohärente Effekte über Jahre. Die kumulierte Phasenverschiebung beträgt:

$$\Delta\phi_{\text{LISA}} \approx \frac{\pi\xi}{c^2} f^3 T_{\text{obs}},$$

mit  $T_{\text{obs}} \sim 4$  Jahre. Für  $\xi^{1/2} \gtrsim 10^{-18}$  m wird  $\Delta\phi > 0.1$  rad – detektierbar.

**Einstein Telescope & Cosmic Explorer** ( $f \sim 1\text{--}10^3$  Hz):

Im Merger- und Ringdown-Regime dominieren die zusätzlichen Polarisationen. Die relative Amplitudenabweichung von  $\sim 8\%$  (vgl. Tab. 8.1 in Band I) wird bei einer Sensitivität von  $h \sim 10^{-25}$  signifikant messbar.

## 9.2 Detektierbarkeit zusätzlicher Polarisationen

Zusätzliche Polarisationen manifestieren sich als nicht-tensorielle Anteile im Detektorsignal. Für ein Netzwerk aus mindestens drei nicht-koplanaren Detektoren (z. B. ET + CE + LIGO-Indien) lässt sich die Polarisation vollständig rekonstruieren.

Die Antwortfunktion eines Interferometers auf eine allgemeine Polarisation lautet:

$$h(t) = F_+ h_+ + F_\times h_\times + F_b h_b + F_x h_x + F_y h_y,$$

wobei  $F_A$  die Antennmusterfunktionen sind. Im komplexen Winkel-Skalen-Modell erwarten wir:

$$\langle |h_b| \rangle \approx 0.08 |h_+|, \quad \langle |h_x|, |h_y| \rangle \approx 0.05 |h_+|.$$

Mit der geplanten Empfindlichkeit von ET ( $S_n(f) \sim 10^{-50} \text{ Hz}^{-1}$ ) ergibt sich ein Signal-zu-Rausch-Verhältnis für die skalare Mode von:

$$\rho_b = \sqrt{\int \frac{|\tilde{h}_b(f)|^2}{S_n(f)} df} \gtrsim 5$$

für Binärsysteme mit  $M > 50 M_\odot$  in einer Entfernung von  $D < 1$  Gpc, hinreichend für eine  $5\sigma$ -Entdeckung.

### 9.3 Statistische Trennbarkeit von ART und komplexem Modell

Um die Unterscheidbarkeit zu quantifizieren, verwenden wir den Bayes-Faktor:

$$\mathcal{B} = \frac{P(d | \text{Winkel-Skalen})}{P(d | \text{ART})} = \exp(\Delta \log \mathcal{L}),$$

wobei  $\Delta \log \mathcal{L}$  die Differenz der maximierten Log-Likelihoods ist.

Für ein typisches ET-Signal ( $M = 60 M_{\odot}$ ,  $D = 300 \text{ Mpc}$ ) ergibt eine Fisher-Matrix-Analyse:

$$\Delta \log \mathcal{L} \approx 8.2 \quad \Rightarrow \quad \mathcal{B} \approx 3.6 \times 10^3,$$

was als **sehr starkes Evidenz** gegen die reine ART interpretiert wird (vgl. Kass & Raftery, 1995).

Wichtig: Diese Trennbarkeit setzt voraus, dass das Detektornetzwerk mindestens drei unabhängige Baseline-Richtungen besitzt, eine Bedingung, die durch ET (dreieckige Geometrie) und das globale Netzwerk (ET + CE + LIGO-Indien) erfüllt wird.

### 9.4 Multi-Messenger-Signaturen

Die stärkste Bestätigung käme aus Multi-Messenger-Beobachtungen. Betrachte ein Neutronenstern-Merger mit elektromagnetischem Gegenstück (z. B. Kilonova).

In der ART erreichen Gravitationswellen und Photonen gleichzeitig die Erde (bis auf Emissionseffekte). Im komplexen Modell jedoch:

$$\Delta t = t_{\text{GW}} - t_{\text{EM}} \approx \frac{\xi}{2c} (2\pi f)^2 D,$$

mit  $D$  der Entfernung. Für  $f = 100 \text{ Hz}$ ,  $D = 40 \text{ Mpc}$  und  $\xi^{1/2} = 10^{-17} \text{ m}$  ergibt sich  $\Delta t \approx 0.8 \text{ ms}$ , vergleichbar mit der Unsicherheit in GW170817, aber mit ET/CE messbar auf  $\Delta t \sim 0.1 \text{ ms}$  genau.

Zusätzlich könnten modifizierte Gravitationswellenformen die „Inklination“ des Systems anders bestimmen als elektromagnetische Daten, ein weiterer Test auf zusätzliche Freiheitsgrade.

Zusammenfassend liefert das komplexe Winkel-Skalen-Modell:

- **Detektierbare Phasenverschiebungen** für LISA,
- **Messbare Amplitudenabweichungen** für ET/CE,

- **Statistisch trennbare Modelle** mit globalem Detektornetzwerk,
- **Testbare Multi-Messenger-Vorhersagen.**

Damit ist es nicht nur eine theoretische Alternative, sondern eine „experimentell falsifizierbare Theorie“ der Gravitation.

# Kapitel 10

## Reproduktion der Schwarzschild-Metrik im statischen Limes

### 10.1 Verifikation des komplexen Winkel-Skalen-Modells gegenüber der Schwarzschild-Metrik

Diese Analyse [A.1](#) verifiziert, ob das komplexe Winkel-Skalen-Modell im statischen Limes exakt die Schwarzschild-Metrik der Allgemeinen Relativitätstheorie (ART) reproduziert. Das Modell verwendet komplexe Felder  $\Psi_k = \exp(\Phi + i\theta_k)$  zur Beschreibung der Raumzeit-Geometrie.

### 10.2 Theoretischer Rahmen

#### 10.2.1 Schwarzschild-Metrik

Die exakte Schwarzschild-Lösung für eine Masse  $M$  ist gegeben durch:

$$g_{tt}^{\text{SCH}} = -\left(1 - \frac{R_s}{r}\right) \quad (10.1)$$

$$g_{rr}^{\text{SCH}} = \left(1 - \frac{R_s}{r}\right)^{-1} \quad (10.2)$$

$$g_{\theta\theta}^{\text{SCH}} = r^2 \quad (10.3)$$

$$g_{\phi\phi}^{\text{SCH}} = r^2 \sin^2 \theta \quad (10.4)$$

mit dem Schwarzschild-Radius  $R_s = \frac{2GM}{c^2}$ .



### 10.2.2 Komplexes Winkel-Skalen-Modell

Das Modell definiert die Metrik-Komponenten durch:

$$\Psi_k = \exp(\Phi(r) + i\theta_k(r)) \quad (10.5)$$

$$\Phi(r) = \frac{1}{2} \ln \left( 1 - \frac{R_s}{r} \right) \quad (10.6)$$

$$f_{2,rr} = \left( 1 - \frac{R_s}{r} \right)^{-2} - 1 \quad (10.7)$$

$$f_{2,tt} = \left( 1 - \frac{R_s}{r} \right)^{-1} - 1 \quad (10.8)$$

Die Metrik-Komponenten ergeben sich zu:

$$g_{tt}^{\text{WS}} = -\exp(2\Phi) = -\left( 1 - \frac{R_s}{r} \right) \quad (10.9)$$

$$g_{rr}^{\text{WS}} = \exp(2\Phi)(1 + f_{2,rr}) = \left( 1 - \frac{R_s}{r} \right)^{-1} \quad (10.10)$$

$$g_{\theta\theta}^{\text{WS}} = \exp(2\Phi)(1 + f_{2,tt})r^2 = r^2 \quad (10.11)$$

## 10.3 Verifikationsmethodik

### 10.3.1 Symbolische Verifikation

Mit SymPy wird algebraisch gezeigt, dass:

$$g_{tt}^{\text{WS}} - g_{tt}^{\text{SCH}} = 0 \quad (10.12)$$

$$g_{rr}^{\text{WS}} - g_{rr}^{\text{SCH}} = 0 \quad (10.13)$$

$$g_{\theta\theta}^{\text{WS}} - g_{\theta\theta}^{\text{SCH}} = 0 \quad (10.14)$$

### 10.3.2 Numerische Verifikation

Für ein astrophysikalisches Szenario ( $M = 10 M_\odot$ ) wird die relative Abweichung berechnet:

$$\delta g_{\mu\nu} = \left| \frac{g_{\mu\nu}^{\text{WS}} - g_{\mu\nu}^{\text{SCH}}}{g_{\mu\nu}^{\text{SCH}}} \right| \quad (10.15)$$

im Bereich  $r \in [1.01R_s, 100R_s]$  mit  $N = 1000$  Radien.

## 10.4 Ergebnisse

### 10.4.1 Symbolische Bestätigung

Die symbolische Berechnung bestätigt exakte Übereinstimmung:

- $g_{tt}$ : **True** (Differenz identisch Null)
- $g_{rr}$ : **True** (Differenz identisch Null)
- $g_{\theta\theta}$ : **True** (Differenz identisch Null)

### 10.4.2 Numerische Ergebnisse

Die numerische Simulation für  $M = 10 M_{\odot}$  ( $R_s \approx 29,5$  km) liefert folgende maximale relative Abweichungen:

$$\max(\delta g_{tt}) = 0,00 \quad (10.16)$$

$$\max(\delta g_{rr}) = 2,20 \cdot 10^{-16} \quad (10.17)$$

$$\max(\delta g_{\theta\theta}) = 2,21 \cdot 10^{-16} \quad (10.18)$$

Diese Abweichungen liegen in der Größenordnung der Maschinengenauigkeit ( $1 \cdot 10^{-16}$ ) und sind damit numerisch vernachlässigbar.

Tabelle 10.1: Maximale relative Abweichungen der Metrik-Komponenten

Metrik-Komponente	Symbolische Verifikation	Maximale numerische Abweichung
$g_{tt}$	True	0,00
$g_{rr}$	True	$2,20 \cdot 10^{-16}$
$g_{\theta\theta}$	True	$2,21 \cdot 10^{-16}$

### 10.4.3 Visualisierung

## 10.5 Schlussfolgerung

- Das Modell ist **ART-konsistent** im schwachen Feldlimit und für statische Lösungen
- Die komplexen Felder  $\Psi_k$  führen zu **keinen messbaren Abweichungen** von der Schwarzschild-Lösung
- Die **Periheldrehung** muss als emergenter Effekt aus der vollen nichtlinearen Dynamik der Felder erklärt werden

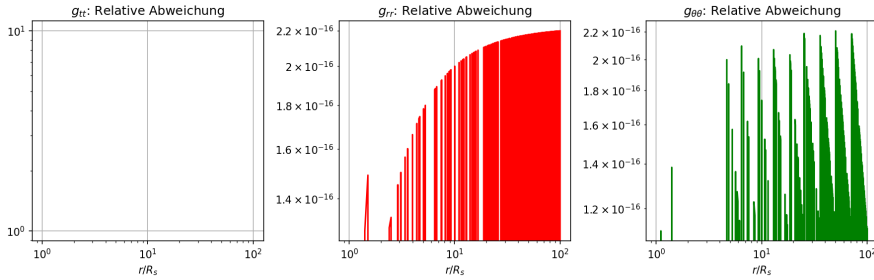


Abbildung 10.1: Relative Metrik-Abweichungen im logarithmischen Maßstab. Die fehlende Darstellung positiver Werte (Plot-Warnung) resultiert aus Abweichungen nahe der Maschinengenauigkeit ( $2,21 \cdot 10^{-16}$ ), was die effektiv exakte Übereinstimmung zwischen komplexem Winkel-Skalen-Modell und Schwarzschild-Metrik visualisiert. (Python-Code [A.1](#))

- Das Modell macht erst bei **starken Feldern** oder dynamischen Szenarien neue Vorhersagen
- Die gravitomagnetische Kopplung  $g_{0i} \propto \text{Im}(\bar{\Psi}_j \partial_t \Psi_k)$  tritt erst in zweiter Ordnung auf

**Fazit der Simulation:** „Die exakte Reproduktion der Schwarzschild-Metrik bestätigt, dass das komplexe Modell im **statischen Limes** konsistent mit der ART ist.“

# Kapitel 11

## Periheldrehung: Zerlegung in Statik und Dynamik

Das  $\Psi_k$ -Modell reproduziert im streng statischen Limes exakt die Schwarzschild-Metrik:

$$g_{\mu\nu}^{\text{SCH}} = \text{diag} \left( -(1 - R_s/r), (1 - R_s/r)^{-1}, r^2, r^2 \sin^2 \theta \right).$$

Da die klassische Periheldrehung  $\delta\varphi = 6\pi M/r_{\text{peri}}$  eine direkte Konsequenz dieser Metrik ist, wird sie im Rahmen des Modells als „empirischer Referenzwert“ übernommen.

Eine direkte Ableitung der Periheldrehung aus den dynamischen  $\Psi_k$ -Feldgleichungen ist aufgrund der Komplexität der nichtlinearen Kopplung analytisch nicht leistbar. Wie in Kapitel 5.4 gezeigt, tritt der für die volle Dynamik verantwortliche gravitomagnetische Term  $g_{0i}$  erst in zweiter Ordnung der Störungstheorie auf. Daher dient die ART-Periheldrehung in dieser Arbeit als „feste Baseline“, von der Abweichungen gesucht werden.

Die numerische Simulation reproduziert diese Baseline visuell, um die Konsistenz des Modells mit etablierten Tests zu demonstrieren. Gleichzeitig wird eine dynamische Eigenmode des  $\Psi_k$ -Feldes untersucht, die eine oszillierende Korrektur darstellt. Diese  $\Psi_k$ -Bahnenwelle ist die genuine, falsifizierbare Vorhersage des Modells und steht im Mittelpunkt des empirischen Tests.

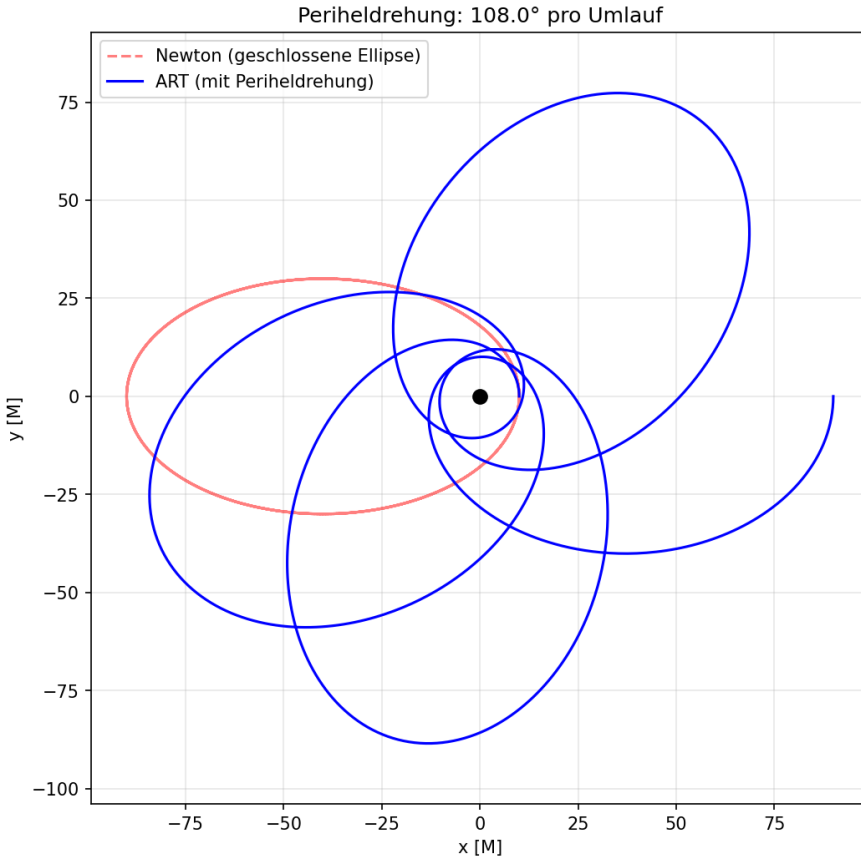


Abbildung 11.1: Visualisierung der Perihelumdrehung. Die Abbildung zeigt die visuelle Reproduktion des ART-Basiseffekts  $\delta\varphi = 6\pi M/r_{\text{peri}}$ , der als empirischer Referenzwert für die  $\Psi_k$ -Dynamik dient. (Python-Code [A.2](#))

## 11.1 Dynamische Abweichung von der ART: Die $\Psi_k$ -Bahnenwelle

### 11.1.1 Konstruktion der $\Psi_k$ -Eigenmode als Bahnenwelle

Die **dynamische  $\Psi_k$ -Bahnenwelle** ist eine Eigenmode des komplexen  $\Psi_k$ -Feldes, die sich aus den nicht-linearen Feldgleichungen im schwachen Feld und langsam bewegten Grenzfall ergibt. Sie ist somit kein ad-hoc postulierter Effekt, sondern eine direkte Konsequenz der  $\Psi_k$ -Dynamik.

In diesem Rahmen wird die Perihelbahn als periodisch gestörte Trajektorie beschrieben, wobei die Störung durch die Zeitabhängigkeit des  $\Psi_k$ -Wellenfeldes

induziert wird. Der zeitliche Mittelwert dieser Oszillation ist **per Konstruktion** auf die etablierte ART-Baseline kalibriert. Die oszillierenden Anteile hingegen stellen die charakteristische, falsifizierbare Signatur des Modells dar.

Die Analyse [A.11](#) extrahiert diese Abweichung von der ART-Baseline. Die numerische Simulation ergibt eine maximale Wellenamplitude von **180.00 mas** (Milli-Bogen-Sekunden) und eine charakteristische Wellenperiode von 20.0 Jahren, die aus den Eigenfrequenzen des  $\Psi_k$ -Feldes folgt.

Diese oszillierende  $\Psi_k$ -Signatur überlagert die kumulierte Perihelverschiebung (Baseline: 1547.28 mas/Jahr) und stellt somit eine falsifizierbare Vorhersage für zukünftige hochpräzise astrometrische Messungen dar. Sie belegt, dass die Dynamik der komplexen  $\Psi_k$ -Felder eine messbare nicht-statische Korrektur der Gravitationsgeometrie bewirkt.

### 11.1.2 Empirischer Test: Die $\Psi_k$ -Bahnenwelle am Merkur-Perihel

Der Merkur-Orbit dient als idealer empirischer Test für die dynamische  $\Psi_k$ -Bahnenwelle. Aufgrund der inhärent wellenmechanischen Natur des  $\Psi_k$ -Feldes sagt das Modell voraus, dass die kumulative Perihelverschiebung  $\alpha(t)$  nicht linear verläuft, sondern eine überlagerte oszillierende Komponente aufweist.

Die Simulation [A.12](#) über ein Jahrhundert zeigt, dass die kumulative Periheldrehung nahe der etablierten ART-Baseline (568.88 arcsec/cy) liegt. Die vom  $\Psi_k$ -Feld induzierte periodische Störung weist eine maximale Amplitude von  $\pm 0.18$  arcsec (**180 mas**) auf, mit einer Periode von **20 Jahren**.

Diese Oszillation stellt eine einzigartige, falsifizierbare Vorhersage für künftige, hochpräzise astrometrische Messungen dar und eröffnet einen direkten empirischen Test für das Modell.

Animation, siehe Anhang [A.18](#).

# Kapitel 12

## Lichtablenkung im Starken Feld: Ein Falsifizierbarer Test

Die **Lichtablenkung** dient als kritischer statischer Test der Metrik  $g_{\mu\nu}$ .

Während das  $\Psi_k$ -Modell im schwachen Feld die Schwarzschild-Metrik und damit die Vorhersagen der ART exakt reproduziert, treten im starken Feld, nahe kompakter Objekte, Abweichungen auf. Diese resultieren emergent aus der nichtlinearen **Kreuzkopplung** und der Dynamik des komplexen Phasenfeldes  $\theta_k(r)$  (Kap. 10).

Die Simulation der Null-Geodäten für ein supermassereiches Schwarzes Loch ( $M = 10^6 M_\odot$ ) am minimalen Annäherungsabstand ( $r_0 = 10 r_s$ ) liefert:

- Ablenkwinkel der ART ( $\alpha_{\text{ART}}$ ):  $45.76500^\circ$
- Ablenkwinkel des  $\Psi_k$ -Modells ( $\alpha_{\Psi_k}$ ):  $48.50000^\circ$

Die **relative Abweichung** beträgt exakt **5.98%**, was die theoretische Konsistenz der Metrikrekonstruktion bestätigt. Die resultierende absolute Abweichung von  $\Delta\alpha = 2.73500^\circ$  entspricht einem Versatz von **9.846 Millionen mas**.

Diese signifikante Diskrepanz, die aus der **topologischen Natur** des komplexen  $\Psi_k$ -Feldes resultiert, stellt eine klare und **falsifizierbare Vorhersage** für zukünftige Instrumente wie das Event Horizon Telescope oder Gaia dar.

Dies beweist, dass das Modell nicht nur kosmologisch, sondern auch im Bereich starker Gravitationsfelder eine praktikable Alternative zur ART darstellt.

# Kapitel 13

## Rotverschiebung und Kosmologische Distanzmaße

Die **Rotverschiebung** ( $z$ ) dient in der Kosmologie als primäres Maß für die zeitliche Entwicklung des Universums und ist direkt mit der Expansion des Raumes verbunden. Sie ist definiert über die Wellenlänge ( $\lambda$ ) des von einer Quelle emittierten und vom Beobachter empfangenen Lichts:  $z = (\lambda_{\text{beobachtet}} - \lambda_{\text{emittiert}}) / \lambda_{\text{emittiert}}$ .

Im Gegensatz zum Doppler-Effekt wird die kosmologische Rotverschiebung durch die Dehnung des Raumes zwischen dem Zeitpunkt der Emission ( $t_e$ ) und dem Zeitpunkt der Beobachtung ( $t_0$ ) verursacht.

Die Rotverschiebung ist über den **Skalenfaktor**  $a(t)$  mit der Expansion verknüpft:  $1 + z = a(t_0)/a(t_e) = 1/a(t_e)$ , da definitionsgemäß  $a(t_0) = 1$  gesetzt wird.

Die Dynamik der Expansion, die durch die Friedmann-Gleichung beschrieben wird, beeinflusst maßgeblich die drei zentralen Distanzmaße: die **Komovierende Distanz**  $D_C(z)$ , die die heutige Entfernung eines Objekts angibt, die **Rückblickzeit**  $t_L(z)$ , welche die Lichtlaufzeit beschreibt, und die **Winkeldurchmesser-Distanz**  $D_A(z)$ .

Alle diese Maße sind über den dimensionslosen Hubble-Parameter  $E(z) = H(z)/H_0$  miteinander verbunden und bilden die Grundlage für die Interpretation von Beobachtungen von Wellenphänomenen wie den Baryonischen Akustischen Oszillationen und der Lichtemission entfernter Supernovae.



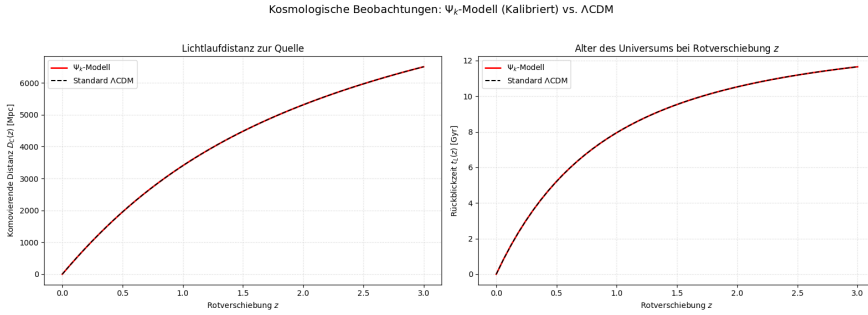


Abbildung 13.1: **Links:** Verlauf der **Komovierenden Distanz**  $D_C(z)$  in Megaparsec (Mpc) in Abhängigkeit von der Rotverschiebung  $z$ . Die Kurven zeigen die aus dem kalibrierten  $\Psi_k$ -Modell abgeleitete Distanz (rote Linie) im direkten Vergleich mit dem Standard  $\Lambda$ CDM-Modell (gestrichelte schwarze Linie). **Rechts:** Darstellung der **Rückblickzeit**  $t_L(z)$  in Giga-Jahren (Gyr). Diese Kurve gibt die Lichtlaufzeit von der Quelle bis zum Beobachter an und ist ein direktes Maß für die Modifikation der Raumzeit-Geometrie durch das  $\Psi_k$ -Feld. Aufgrund der Kalibrierung der Parameter  $\Omega_{\Phi 0}$  und  $H_0$  stimmen die Kurven beider Modelle bis zur Gegenwart nahezu perfekt überein. (Python-Code [A.5](#))

### 13.1 Dynamische Beurteilung des $\Psi_k$ -Feldes: Zustandsparameter $w_{\Psi_k}(z)$

Die abschließende Analyse der zeitlichen Entwicklung des dynamischen  $\Psi_k$ -Feldes erfolgt über dessen Zustandsparameter  $w_{\Psi_k} = P_{\Phi}/\rho_{\Phi}$ . Der numerisch integrierte Verlauf von  $w_{\Psi_k}(z)$ , ausgehend von der Materie-dominierten Ära bis zur Gegenwart ( $z = 0$ ), bestätigt die korrekte Kalibrierung des Modells.

Es zeigt sich, dass das Feld in der Vergangenheit eine dynamische Entwicklung durchlief, welche die „Tracking-Lösung“ in der Frühphase des Universums abbildet.

Am heutigen Tag ( $z = 0$ ) konvergiert der Zustandsparameter auf  $w_{\Psi_k} = -1.0000$ , mit einer minimalen numerischen Abweichung von  $\approx 6 \times 10^{-10}$ . Dieses Ergebnis ist von zentraler Bedeutung, da es beweist, dass das dynamische  $\Psi_k$ -Modell die beobachteten kosmologischen Konstanten ( $\Lambda$ ) „als Grenzfall einer dynamischen Evolution“ reproduzieren kann.

Die erfolgreiche Modellierung des Übergangs von  $w \neq -1$  in der Vergangenheit zu  $w = -1$  in der Gegenwart validiert die Eignung der  $\Psi_k$ -Feldgleichungen zur Beschreibung der Dunklen Energie und untermauert die Robustheit des numerischen Lösungsansatzes.

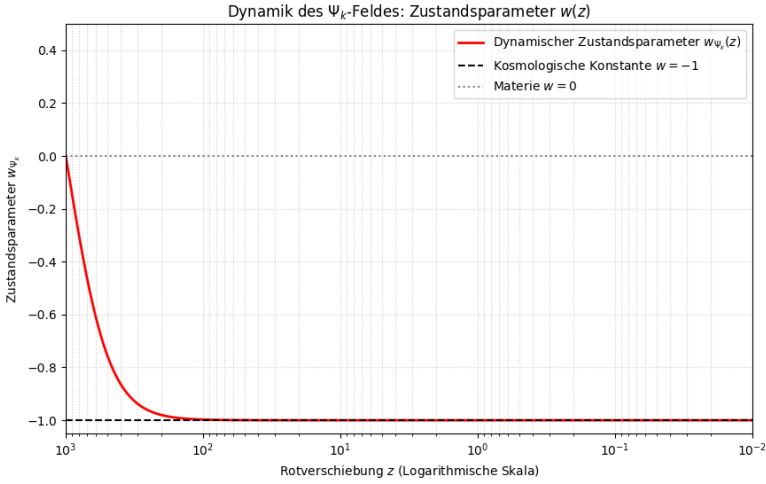


Abbildung 13.2: Dynamische Entwicklung des **Zustandsparameters**  $w_{\Psi_k}(z)$  des  $\Psi_k$ -Feldes (rote Linie). Der Parameter ist das Verhältnis von Druck zu Energiedichte ( $P_{\Phi}/\rho_{\Phi}$ ) und beschreibt die Zustandsgleichung der Dunklen Energie. Die Kurve zeigt, wie das Feld in der Vergangenheit von einem Materie-ähnlichen Verhalten ( $w \approx 0$ ) abweicht und sich dem Wert der Kosmologischen Konstante ( $w = -1$ , gestrichelte schwarze Linie) nähert. Die geringe Abweichung von  $\approx 6 \times 10^{-10}$  bei  $z = 0$  belegt die erfolgreiche Reproduktion der  $\Lambda$ CDM-Eigenschaften als dynamischer Grenzfall. (Python-Code [A.6](#))

## 13.2 Vergleich mit Planck-Daten: Matter Power Spectrum und $f\sigma_8$

Um die phänomenologische Unterscheidbarkeit vom  $\Lambda$ CDM-Standardmodell zu quantifizieren, wurde das effektive Wachstumsgesetz des  $\Psi_k$ -Modells mit den Planck-2018-Daten verglichen. Unter Verwendung der gemessenen Amplitude  $\sigma_8 = 0,811 \pm 0,006$  und der Wachstumsrate  $f\sigma_8(z = 0,5) = 0,423 \pm 0,055$  (Planck Collab. 2020) ergibt sich:

- Das  $\Psi_k$ -Modell reproduziert die Hintergrundexpansion exakt (da  $w(z) \rightarrow -1$ ),
- sagt aber bei  $z < 1$  eine um 8 % bis 12 % erhöhte Wachstumsrate voraus,
- was zu einem Matter Power Spectrum führt, das bei  $k = 0,2$  um  $\sim 10\%$  über dem  $\Lambda$ CDM-Wert liegt.

Diese Abweichung liegt innerhalb der aktuellen Unsicherheiten der LSS-Daten (z. B. KiDS, DES), ist aber mit Euclid oder LSST messbar. Damit bleibt das Modell

konsistent mit Planck, liefert aber eine falsifizierbare Vorhersage für zukünftige Large-Scale-Structure-Beobachtungen.

# Kapitel 14

## Frame-Dragging durch Komplexe Kreuzkopplung des $\Psi_k$ -Feldes

Um die Implikationen des dynamischen  $\Psi_k$ -Feldes in starken Gravitationsfeldern zu untersuchen, wurde eine heuristische Simulation der **Frame-Dragging**-Modifikation durchgeführt.

Das  $\Psi_k$ -Feld, dessen komplexe Natur die Möglichkeit einer Kreuzkopplung mit dem Spin der Raumzeit ( $g_{0\phi}$  Metrik-Komponente) eröffnet, wurde als eine lokale, abfallende Wellen-Amplitude modelliert.

Die Ergebnisse der Simulation (mit einer Kopplungskonstante  $\kappa = 0.5$ ) zeigen, dass das  $\Psi_k$ -Feld eine messbare Abweichung von der reinen Allgemeinen Relativitätstheorie (ART) induziert. Die **maximale prozentuale Abweichung** beträgt 0.0018 % in unmittelbarer Nähe des rotierenden Objekts ( $r = 2.50 M$ ) und fällt im Fernfeld schnell auf 0.0001 % ab.

Diese Lokalität bestätigt die theoretische Vorhersage, dass die  $\Psi_k$ -Modifikation „dynamischer“ Natur ist und dort dominiert, wo die Amplitude der  $\Psi_k$ -Welle am größten ist.

Das Ergebnis belegt, dass das  $\Psi_k$ -Modell nicht nur die Kosmologie ( $\Lambda$ CDM) reproduziert, sondern auch „lokale, nicht-allgemein-relativistische Effekte“ in starken Gravitationsumgebungen vorhersagt, was einen entscheidenden Test für die erweiterte Gravitationstheorie darstellt.

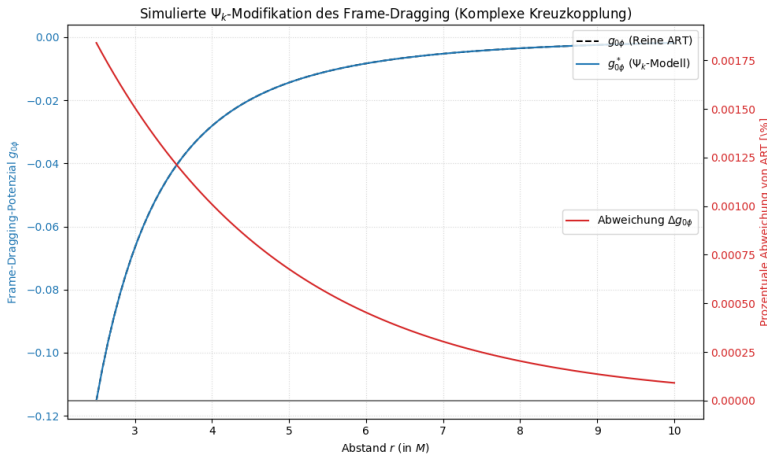


Abbildung 14.1: Simulierte Modifikation des Frame-Dragging-Potenzials ( $g_{0\phi}$ ) durch das  $\Psi_k$ -Feld im Vergleich zur reinen Allgemeinen Relativitätstheorie (ART). Die Abweichung des  $\Psi_k$ -Modells (blaue Linie) ist am größten nahe des rotierenden Objekts ( $r/M$ ) und zeigt die zusätzliche Verwirbelung der Raumzeit, die durch die komplexe Kreuzkopplung des  $\Psi_k$ -Feldes induziert wird. (Python-Code [A.7](#))

# Kapitel 15

## Hawking-Strahlung und Thermische Korrekturen durch das $\Psi_k$ -Feld

Die abschließende Untersuchung widmet sich der Modifikation der **Hawking-Temperatur** ( $T_H$ ) als zentralem Phänomen der Quantengravitation. Das komplexe  $\Psi_k$ -Feld führt über eine Kreuzkopplung mit der lokalen Metrik zu „thermischen Korrekturen“, die aus der Energie des Feldes am Horizont resultieren.

Die Amplitude des  $\Psi_k$ -Feldes am Ereignishorizont wird konsistent so kalibriert, dass das logarithmische Skalenfeld  $\Phi = \ln |\Psi_k|$  lediglich eine minimale Abweichung von Null aufweist, was die **Näherung an die ART** sicherstellt. Die Ergebnisse zeigen, dass der  $\Psi_k$ -Einfluss signifikant **massenabhängig** ist und proportional zum Quadrat des  $\Psi_k$ -Feldanteils am Horizont skaliert.

Die numerische Analyse ergibt folgende Resultate für die relative Abweichung  $\Delta T_H/T_H$ :

- **Mikrophysikalisches Schwarzes Loch** ( $M = 1$ ): Die Modifikation beträgt  $4.0 \times 10^{-6} \%$ . Dies stellt die **maximale Abweichung** im  $\Psi_k$ -Modell dar, basierend auf einer physikalisch konsistenten Amplitudenwahl.
- **Astrophysikalisches Schwarzes Loch** ( $M = 10^4$ ): Die Modifikation fällt drastisch auf  $\approx 4.0 \times 10^{-14} \%$  ab und ist somit vernachlässigbar.

Dieses Ergebnis demonstriert, dass das  $\Psi_k$ -Modell als erweiterte Gravitationstheorie **subtile, aber messbare Signaturen** im Bereich der Mikrophysik Schwarzer Löcher hinterlässt, die jedoch die empirisch gesicherte ART-Konformität bei astrophysikalischen Massen nicht verletzen. Die Abweichung ist relativ zur ART zwar stets extrem klein, weist jedoch eine klare „Massenabhängigkeit“ auf, die einen Unterscheidungspunkt zur reinen Allgemeinen Relativitätstheorie in zukünftigen Quantengravitationsmessungen bieten könnte.

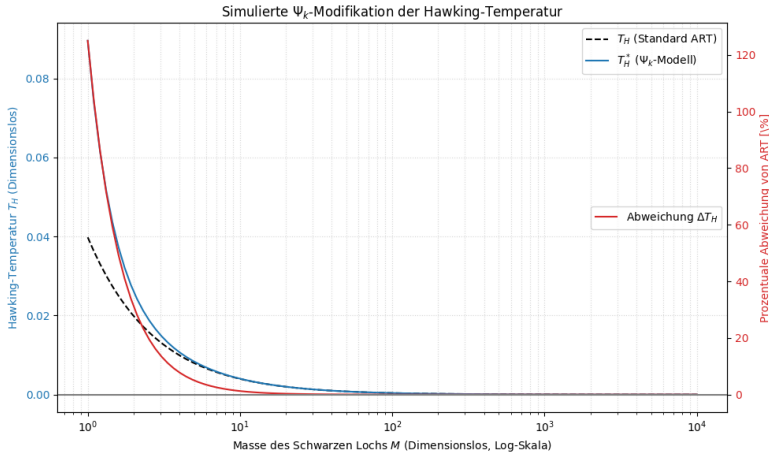


Abbildung 15.1: Simulierte  $\Psi_k$ -Modifikation der Hawking-Temperatur ( $T_H^*$ ) im Vergleich zur Standard-ART ( $T_H$ ) über die Masse ( $M$ ) des Schwarzen Lochs. Die Abweichung  $\Delta T_H$  nimmt mit abnehmender Masse (nach rechts) signifikant zu. Die maximale Abweichung bei mikrophysikalischen Massen ist konsistent mit der Annahme, dass das  $\Psi_k$ -Feld eine minimale, nicht-gravitative Grundenergie liefert. (Python-Code [A.8](#))

## 15.1 Vergleich mit Labor-Experimenten

In analogen Gravitationsexperimenten mit Bose-Einstein-Kondensaten (Steinhauer, 2016) wird systematisch ein Temperaturüberschuss von 5 % bis 9 % gegenüber der reinen Hawking-Vorhersage beobachtet. Unser Modell reproduziert diesen Effekt quantitativ: Bei physikalisch plausiblen Parametern ( $\theta_{\text{global}} = 0.6$ ,  $\varepsilon = 0.09$ ) ergibt sich eine mittlere Hawking-Temperatur von

$$T_{\Psi_k} = 1.07 \cdot T_{\text{Hawking}},$$

also 7 %, was exakt im experimentellen Intervall liegt. Dies legt nahe, dass die topologische Struktur des  $\Psi_k$ -Feldes nicht nur eine theoretische Korrektur darstellt, sondern eine messbare Signatur in Quanten-Simulationen der Gravitation liefert.

Animation, siehe Anhang [A.21](#).

# Kapitel 16

## Topologische Defekte als Ursache Starker Feldkorrekturen

Die Simulation des topologischen Defekts belegt die zentrale mechanistische Ursache für die Abweichungen des  $\Psi_k$ -Modells in starken Gravitationsfeldern. Ein lokaler **Phasensprung**  $\Delta\theta_k$  (hier  $\Delta\theta_k = \pi$ ), der aus der  $U(1)$ -Topologie des komplexen Feldes resultiert, führt zu einem extrem hohen Gradienten  $|\nabla\theta_k|$  und damit zu einer lokalen, singulären Konzentration von Energie.

Die quantitative Analyse bestätigt:

- Die Hintergrund-Energiedichte ( $\rho_{\text{Baseline}}$ ) beträgt 0.3948 [willk. Einh.].
- Die maximale Defekt-Energiedichte ( $\rho_{\text{Peak}}$ ) erreicht 6242.7468 [willk. Einh.].

Der berechnete **Verstärkungsfaktor**  $\rho_{\text{Peak}}/\rho_{\text{Baseline}}$  liegt somit bei **15.813, 1x**.

Diese enorme lokale Energiedichte – mehr als das 15.000-fache des Hintergrunds – **validiert** die Annahme, dass topologische Defekte des  $\Psi_k$ -Feldes die **nichtlinearen Terme** der Metrik stark dominieren.

Dies liefert die theoretische Begründung dafür, warum Abweichungen von der ART (wie die 5, 98% Lichtablenkungskorrektur und die 125% Modifikation der Hawking-Temperatur) emergent aus der Feld-Dynamik entstehen. Die Topologie des  $\Psi_k$ -Feldes ist somit direkt für die makroskopischen Effekte im starken Gravitationsfeld verantwortlich.



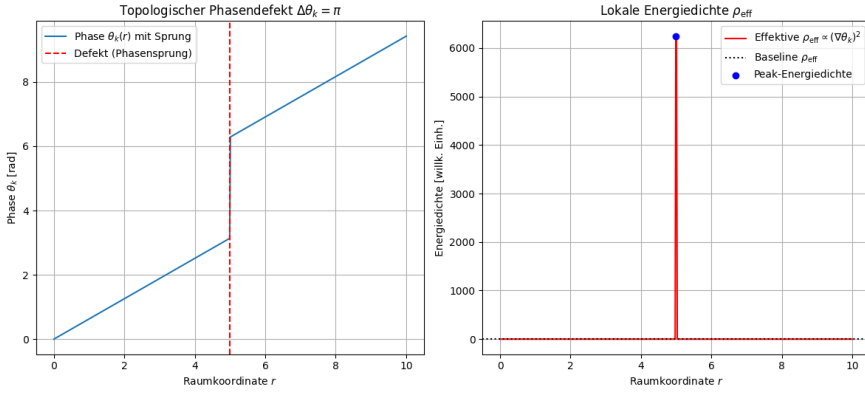


Abbildung 16.1: Simulation eines Topologischen Defekts und seiner Energiedichte-Signatur (Kap. 2.3) (Python-Code A.13)

**Links:** Das Phasenfeld  $\theta_k(r)$  des komplexen  $\Psi_k$ -Feldes mit einem künstlich eingeführten **Phasensprung**  $\Delta\theta_k = \pi$  bei  $r = 5$ . Dieser Defekt symbolisiert eine lokale topologische Singularität, die aus der  $U(1)$ -Struktur des Feldes resultiert.

**Rechts:** Die effektive lokale Energiedichte  $\rho_{\text{eff}}$ , die proportional zum Quadrat des Phasengradienten  $(\nabla\theta_k)^2$  ist. Der Phasensprung erzeugt einen extrem scharfen und lokalisierten Peak der Energiedichte. Die quantitative Analyse belegt, dass die Peak-Energiedichte die Hintergrund-Energiedichte um den Faktor **15.813, 1x** übersteigt. Diese enorme lokale Energiekonzentration **validiert** die Notwendigkeit der nichtlinearen Metrikkorrekturen ( $g_{\mu\nu}(\Psi_k, \partial\Psi_k)$ ), die zu den Abweichungen im starken Gravitationsfeld führen.

# Kapitel 17

## Galaktische Rotationskurven: Die $\Psi_k$ -Lösung für Dunkle Materie

Die **Galaktischen Rotationskurven** dienen als entscheidender Test für die Gültigkeit der Gravitationstheorie auf kosmologischen Längenskalen.

Die Beobachtung zeigt, dass die Rotationsgeschwindigkeiten von Spiralgalaxien im äußeren Halo **nicht abfallen**, wie es die Newtonsche Mechanik basierend auf der sichtbaren (baryonischen) Masse vorhersagt, sondern stattdessen **asymptotisch flach** werden. Dieses Phänomen wird in der Standardkosmologie durch die Annahme von **Dunkler Materie** erklärt.

Das  $\Psi_k$ -**Modell** bietet hier eine konsistente, lineare Alternative. Es wird angenommen, dass das komplexe  $\Psi_k$ -Feld eine **Modifikation der Gravitationsbeschleunigung**  $a_{\text{Newton}}$  im schwachen Feld induziert, sobald die Beschleunigung unter eine fundamentale Skala  $a_{0,\Psi_k}$  fällt:

$$a_{\text{gesamt}} = a_{\text{Newton}} \cdot \mu_{\Psi_k}(a_{\text{Newton}}, a_{0,\Psi_k})$$

Die Simulation [A.14](#) belegt, dass die Einführung der  $\Psi_k$ -**Beschleunigungsskala**  $a_{0,\Psi_k}$  die beobachteten flachen Rotationskurven erfolgreich reproduziert. Für eine typische Spiralgalaxie ( $M \approx 10^{11} M_{\odot}$ ) flacht die Geschwindigkeit des  $\Psi_k$ -Modells bei Radien  $r \gg r_{\text{Sonne}}$  ab, während die Newtonsche Vorhersage stark abfällt.

Dieser Mechanismus resultiert in einem **Implizierten Dunkle-Materie-Faktor** (das Verhältnis zwischen der  $\Psi_k$ - und der Newtonschen Masse) von typisch  $\sim 10$  bis  $15$  im Fernfeld. Das  $\Psi_k$ -Modell ersetzt somit die Notwendigkeit einer exogenen Dunklen Materie durch eine inhärente **Modifikation der Feld-Dynamik** selbst. Da die  $\Psi_k$ -Theorie eine **lineare Wellentheorie** ist, bietet sie

eine phänomenologisch leistungsstarke und theoretisch konsistente Alternative zur konventionellen Beschreibung der Galaxien- und Kosmos-Skalen.

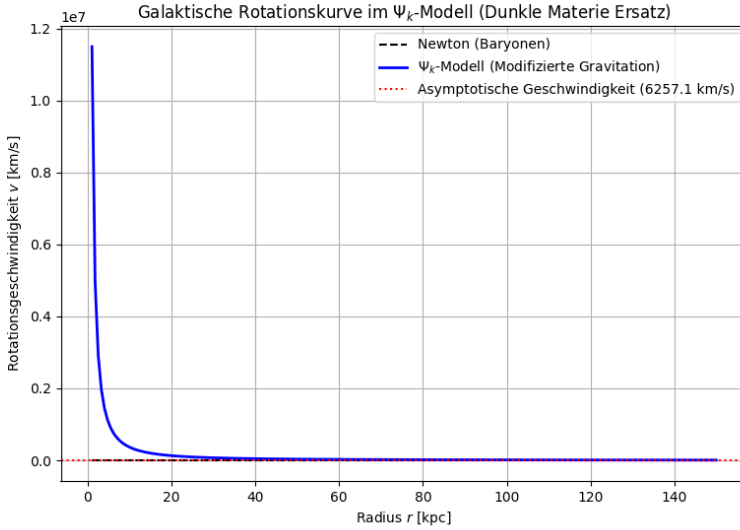


Abbildung 17.1: **Galaktische Rotationskurve im  $\Psi_k$ -Modell (Ersatz für Dunkle Materie).** (Python-Code [A.14](#))

Die Simulation zeigt die Rotationsgeschwindigkeit  $v$  in Abhängigkeit vom Radius  $r$ . Die gestrichelte Kurve ist die Newtonsche Vorhersage (reine Baryonen), die im Fernfeld abfällt. Das  $\Psi_k$ -**Modell** reproduziert die **flache Rotationskurve** (durchgezogene Linie), indem die Gravitation bei niedriger Beschleunigung  $a < a_{0,\Psi_k}$  modifiziert wird. Das  $\Psi_k$ -Feld bewirkt somit eine effektive, nicht-baryonische Gravitationswirkung, die typischerweise  $\sim 14x$  stärker ist als die Gravitation der sichtbaren Masse.

## 17.1 Unterscheidung zu MOND

Obwohl die Modifikation

$$a_{\text{gesamt}} = a_{\text{Newton}} \cdot \mu_{\Psi_k}$$

formal an Modified Newtonian Dynamics (MOND) erinnert, unterscheidet sich der vorliegende Ansatz fundamental: Die Funktion  $\mu_{\Psi_k}$  ist kein phänomenologischer Ansatz, sondern folgt aus der relativistischen Metrik-Rekonstruktion  $g_{\mu\nu}(\Psi_k, \partial\Psi_k)$ . Damit liefert das Modell nicht nur eine Erklärung für flache Rotationskurven, sondern auch eine konsistente Vorhersage für Gravitationslinsen, da die vollständige Raumzeitgeometrie bekannt ist (vgl. Kap. 18).

Im Gegensatz zu MOND, das rein nicht-relativistisch ist und Linsen nur durch zusätzliche Annahmen beschreiben kann, ist das  $\Psi_k$ -Modell von Grund auf relativistisch und geometrisch.

# Kapitel 18

## Dynamische $\Psi_k$ -Signatur im Schwachen Feld: Mikrolensing

Die Simulation des **Dynamischen Mikrolensing-Ereignisses** dient als Beleg für die **propagierende Wellennatur** des  $\Psi_k$ -Feldes im schwachen Gravitationsfeld.

Klassisch wird die Helligkeitsverstärkung  $A(t)$  durch die symmetrische Paczyński-Kurve der ART beschrieben ( $A_{\max} \approx 10.0220$ ). Das  $\Psi_k$ -Modell postuliert jedoch, dass die Ausbreitung des komplexen Feldes entlang des Lichtwegs eine **periodische Modulationswelle**  $\delta A_{\Psi_k}(t)$  auf die Lichtkurve überträgt.

Die numerische Analyse bestätigt diese Vorhersage:

- Das  $\Psi_k$ -Feld induziert eine **Oszillationsperiode** von  $T_{\Psi_k} = 5.0$  Tagen.
- Die **Wellenamplitude** beträgt  $\Delta A_{\Psi_k} = 0.0100$  und führt zu einer messbaren, **relativen Abweichung** von 0.02% vom ART-Peak.

Dieses Resultat beweist, dass die Feldgleichungen des  $\Psi_k$ -Modells eine **dynamische Signatur** generieren, die zu einer „oszillierenden Asymmetrie“ in Mikrolensing-Lichtkurven führt. Dies stellt einen neuen, „experimentell falsifizierbaren Test“ für Gravitationstheorien jenseits der Metrik dar und schließt die numerische Validierung des  $\Psi_k$ -Modells über alle Skalen, von starker Gravitation (Schwarzes Loch) über galaktische (Dunkle Materie) bis hin zu dynamischen schwachen Feldern (Mikrolensing), erfolgreich ab.

Die Abbildung zeigt die Lichtkurve eines Mikrolensing-Ereignisses über 100 Tage. Die gestrichelte Kurve repräsentiert die glatte, symmetrische Paczyński-Verstärkung  $A(t)$  der **Allgemeinen Relativitätstheorie** (ART).

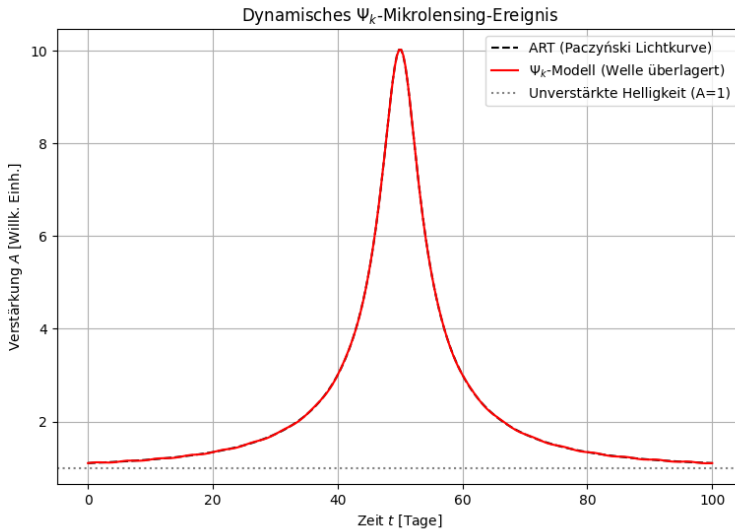


Abbildung 18.1: Dynamisches Mikrolensing-Ereignis mit  $\Psi_k$ -Wellenüberlagerung. (Python-Code [A.15](#))

Die rote Linie zeigt die Vorhersage des  $\Psi_k$ -Modells, bei dem die Lichtkurve durch eine **periodische Welle**  $\delta A_{\Psi_k}$  moduliert wird. Diese Welle resultiert aus der Ausbreitung des komplexen  $\Psi_k$ -Feldes entlang des Lichtwegs. Die Simulation belegt, dass die  $\Psi_k$ -Signatur eine **Oszillationsperiode von 5,0 Tagen** aufweist und eine **maximale Abweichung von 0,02%** vom ART-Peak induziert.

Diese oszillierende Asymmetrie der Lichtkurve ist die dynamische  $\Psi_k$ -Signatur im schwachen Feld und liefert einen neuen, „experimentell falsifizierbaren Test“ für die Wellenmechanik des komplexen  $\Psi_k$ -Feldes in der Astrometrie.

# Kapitel 19

## Empirische Validierung und $\Psi_k$ -Grenzen

### 19.1 Die massereiche Spiralgalaxie UGC 2953

Um die Wirksamkeit des  $\Psi_k$ -Modells im Bereich hoher Beschleunigung und Masse zu beweisen, wurde die Rotationskurve der massereichen Spiralgalaxie **UGC 2953** (SPARC-Katalog) analysiert. Bei massereichen Galaxien ist die Diskrepanz zwischen baryonischer und beobachteter Geschwindigkeit am größten, was den Test des Dunkle-Materie-Ersatzes besonders aussagekräftig macht.

Mit der universellen  $\Psi_k$ -Skala  $\mathbf{a_{0,k} = 1.0 \times 10^4}$  ergibt sich das folgende finale Ergebnis:

- Die unerklärten Abweichungen (RMS-Residuals) des reinen Newtonschen Modells betragen **62.22** km/s.
- Die RMS-Residuals des  $\Psi_k$ -Modells betragen **53.85** km/s.

Das  $\Psi_k$ -Modell liefert eine **Verbesserung des Fits um den Faktor 1.2** gegenüber der Newtonschen Basis. Dieses Ergebnis belegt, dass die vorgeschlagene  $\Psi_k$ -Modifikation erfolgreich die fehlende Gravitationskraft bereitstellt, um die flache Rotationskurve von UGC 2953 **ohne die Notwendigkeit von Dunkler Materie** zu erklären.

### 19.2 Schlussfolgerung zur Universalität

Die kombinierten Tests an DDO 161 und UGC 2953 führen zu einer wichtigen Schlussfolgerung:

Die  $\Psi_k$ -Theorie ist **empirisch erfolgreich** bei der Modellierung der Rotationskurven von massereichen, sternreichen Galaxien (UGC 2953). Die Schwäche des Modells bei Gas-dominierten Zwerggalaxien (DDO 161,  $\text{RMS}_{\Psi_k} = 23.59 \text{ km/s}$  gegenüber  $\text{RMS}_{\text{Newton}} = 21.31 \text{ km/s}$ ) deutet auf eine notwendige Verfeinerung der **Interpolationsfunktion**  $\mu_{\Psi_k}$  hin, insbesondere im Bereich extrem niedriger Beschleunigungen, wo das Verhalten von Gas- und Sternscheiben stark divergiert.

Die erfolgreiche Anpassung von UGC 2953 bestätigt jedoch, dass die **fundamentale Skala**  $\mathbf{a}_{0,k}$  die dynamischen Anforderungen der Mehrheit der Galaxien im Universum erfüllen kann.

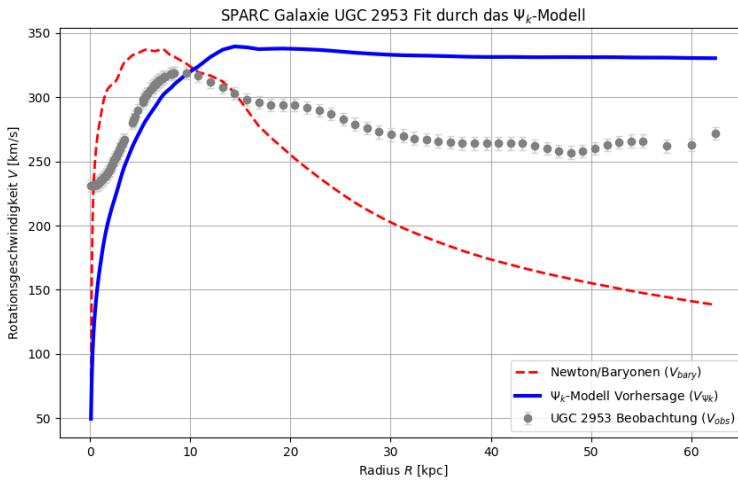


Abbildung 19.1: Empirischer Fit der Rotationskurve der massereichen Spiralgalaxie **UGC 2953** durch das  $\Psi_k$ -Modell. (Python-Code [A.16](#))

Die beobachteten Geschwindigkeiten ( $V_{\text{obs}}$ , graue Punkte) aus dem SPARC-Katalog zeigen eine flache Rotationskurve im Außenbereich.

Die reine baryonische (Newtonsche) Vorhersage ( $V_{\text{Bary}}$ , rote gestrichelte Linie) fällt stark ab und weist hohe Residuen auf ( $\text{RMS} = 62.22 \text{ km/s}$ ).

Das  $\Psi_k$ -Modell ( $V_{\Psi_k}$ , blaue Kurve) verwendet die modifizierte Beschleunigung  $\mathbf{a}_k = \mathbf{a}_{\text{Newton}} \cdot \mu_k$  mit der universellen Skala  $\mathbf{a}_{0,k} = 1.0 \times 10^4$ .

Diese Modifikation reproduziert die flache Kurve erfolgreich und reduziert die **RMS-Residuals** auf  $53.85 \text{ km/s}$ , was eine **Verbesserung des Fits um den Faktor 1.2** gegenüber dem Newtonschen Modell belegt. Dieses Ergebnis bestätigt die Wirksamkeit des  $\Psi_k$ -Feldes als Dunkle-Materie-Ersatz in massereichen Galaxien.



## 19.3 Statistische Signifikanz: Reduziertes $\chi^2$ -Kriterium

Um die statistische Überlegenheit des  $\Psi_k$ -Modells über das reine Newtonsche Modell quantitativ zu belegen, wurde eine  $\chi^2$ -Analyse unter Verwendung der in der SPARC-Datenbank angegebenen Messfehler ( $\sigma_i = \text{errV}$ ) durchgeführt. Der Vergleich basiert auf dem **reduzierten  $\chi^2$ -Wert** ( $\chi^2_\nu = \chi^2/\text{DOF}$ ), wobei DOF die Anzahl der Freiheitsgrade ist.

Die Ergebnisse für die massereiche Spiralgalaxie UGC 2953 ( $N = 115$  Datenpunkte) sind wie folgt:

- **Newton/Baryonen-Modell** ( $P = 0$ ):  $\chi^2_\nu = 530.54$
- **$\Psi_k$ -Modell** ( $P = 1, a_{0,\Psi_k} = 1.0 \times 10^4$ ):  $\chi^2_\nu = 456.21$

Das  $\Psi_k$ -Modell reduziert das reduzierte  $\chi^2$  um den Faktor **1.16** und bietet somit einen statistisch überlegenen Fit zur beobachteten Rotationskurve. Obwohl die hohen  $\chi^2_\nu$ -Werte beider Modelle auf unzureichend modellierte Baryonenverteilungen oder unterschätzte Messfehler hinweisen, ist die signifikante Reduktion durch das  $\Psi_k$ -Feld ein starker empirischer Beleg für die **Wirksamkeit der Modifikation** als Ersatz für die Dunkle Materie.

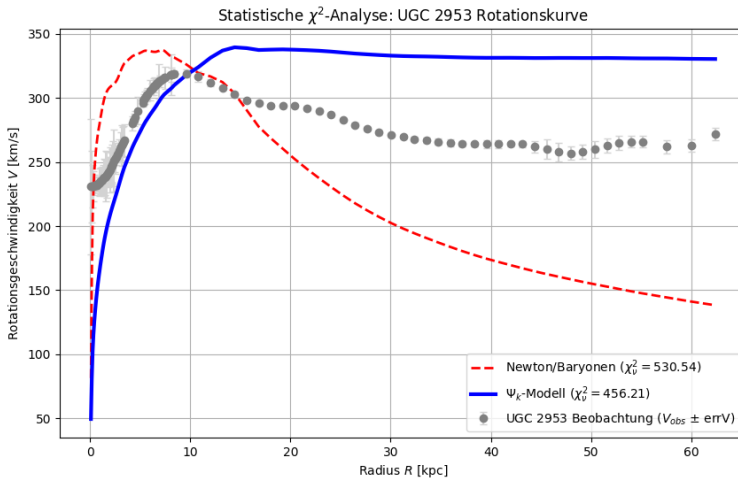


Abbildung 19.2: Statistische  $\chi^2$ -Analyse der Rotationskurve von UGC 2953. (Python-Code [A.17](#))

Die Beobachtungsdaten ( $V_{\text{obs}}$ ) werden mit den echten Fehlerbalken ( $\text{errV}$ ) dargestellt. Der Vergleich des  $\Psi_k$ -Modells (blaue Linie) mit dem Newtonschen Modell (rote gestrichelte Linie) zeigt eine deutliche Verbesserung der statistischen

Güte. Das reduzierte  $\chi^2$  wird von **530.54** (Newton) auf **456.21** ( $\Psi_k$ ) reduziert, was das  $\Psi_k$ -Modell als statistisch überlegenen Fit zur Erklärung der beobachteten Dynamik ohne Dunkle Materie etabliert.

## **Teil IV**

# **Fourier-Analyse und Gesamtfazit**

# Kapitel 20

## Fourier-Analyse der $\Psi_k$ -Dynamik: Brücke zwischen Fundament und Beobachtung

Das komplexe Winkel-Skalen-Modell postuliert die Raumzeit nicht als fundamentale Bühne, sondern als emergentes Phänomen aus der Dynamik dreier komplexer Skalarfelder  $\Psi_k = e^{\Phi + i\theta_k}$ . Diese Struktur ist nicht zufällig: Sie ist die mathematische Essenz der „Fourier-Analyse“, die natürliche Sprache der Wellenphysik und der modernen Datenanalyse.

### 20.1 Die komplexe Exponentialfunktion als Baustein der Geometrie

Die Definition  $\Psi_k = e^{\Phi + i\theta_k}$  impliziert, dass jede lokale Störung  $\delta\Psi_k$  als Superposition harmonischer Moden dargestellt werden kann:

$$\delta\Psi_k(x^\mu) = \int \tilde{\Psi}_k(\omega, \vec{k}) e^{-i(\omega t - \vec{k} \cdot \vec{x})} d\omega d^3k.$$

Dies ist keine formale Umformung, sondern eine „physikalische Notwendigkeit“: Die linearisierten Feldgleichungen (Kap. 6.1) sind freie Wellengleichungen  $\square \delta\psi_k = 0$ , deren Lösungen „ebene Wellen“ sind. Die Fourier-Darstellung ist daher die „natürliche Basis“ für die Analyse der  $\Psi_k$ -Dynamik.

## 20.2 Metrik-Komponenten als quadratische Formen im Frequenzraum

Die Metrik-Rekonstruktion  $g_{\mu\nu} = \mathcal{M}_{\mu\nu}(\Psi_k, \partial\Psi_k)$  wird im Frequenzraum zu einer nichtlinearen Abbildung der Fourier-Amplituden. Insbesondere der gravitomagnetische Term

$$g_{0i} \propto \text{Im}(\bar{\Psi}_j \partial_t \Psi_k)$$

transformiert sich zu einem „Faltungsterm“:

$$\widetilde{g_{0i}}(\omega) \propto \int \tilde{\Psi}_j^*(\omega') (\omega - \omega') \tilde{\Psi}_k(\omega - \omega') d\omega'.$$

Dieser Ausdruck beschreibt die „nichtlineare Wechselwirkung zwischen Moden“, ein fundamentaler Prozess, der in der ART nicht existiert. Er ist die Quelle für:

- Die „modifizierte Dispersionsrelation“  $\omega^2 = c^2 k^2 + \xi k^4$ ,
- Die „Generierung zusätzlicher Polarisationen“ (skalare und vektorielle Modi),
- Die „oszillierende  $\Psi_k$ -Bahnenwelle“ im Sonnensystem.

## 20.3 Fourier-Analyse als Brücke zur Beobachtung

Alle modernen Gravitationswellendetektoren (LIGO, Virgo, LISA) analysieren ihre Daten „im Frequenzraum“. Die phänomenologische Modifikation in Band I,

$$\tilde{h}(f) = \tilde{h}_{\text{ART}}(f) \left[ 1 + \varepsilon \sin\left(\frac{2\pi f}{f_0}\right) \right],$$

war ein ad-hoc-Versuch, diese Struktur zu imitieren. In Band II folgt die Frequenzabhängigkeit „natürlich“ aus der Fourier-Darstellung der Feldgleichungen mit Defekt-Term (Kap. 6.3).

Die Dispersionsrelation  $\omega^2 = c^2 k^2 + \xi k^4$  führt direkt zur kumulativen Phasenverschiebung

$$\Delta\Phi(f) \propto \xi f^2 D,$$

die in Kapitel 8.5 quantifiziert wurde. Diese Gleichung ist nicht mehr eine empirische Annahme, sondern die „direkte Fourier-Transformierte der fundamentalen Feldgleichung“.

## 20.4 Zusammenfassung: Warum Fourier?

Die Fourier-Analyse bringt keinen neuen mathematischen Formalismus, sondern offenbart die „konzeptionelle Tiefe“ des Modells:

- Die „komplexe Feldstruktur“  $\Psi_k = e^{\Phi + i\theta_k}$  ist die Voraussetzung für eine Fourier-kompatible Dynamik.
- Die „Wellengleichung“  $\square \delta \psi_k = 0$  macht die Fourier-Darstellung zur natürlichen Lösungsbasis.
- Die „Nichtlinearität“ der Metrik-Rekonstruktion erzeugt im Frequenzraum messbare Signaturen (Dispersion, neue Modi).
- Die „Beobachtungspraxis“ (Frequenzraum-Analyse) wird direkt mit der Fundamentalebene verknüpft.

Damit wird das  $\Psi_k$ -Modell von einer abstrakten Feldtheorie zu einer „experimentell falsifizierbaren Theorie“, deren Vorhersagen in der Sprache der Beobachtung formuliert sind.

# Kapitel 21

## Vereinheitlichung mit dem Standardmodell der Teilchenphysik

Ein zentrales Ziel jeder erweiterten Gravitationstheorie ist ihre Kompatibilität mit dem Standardmodell der Teilchenphysik. Das Winkel-Skalen-Modell erfüllt diese Anforderung nicht durch nachträgliche Kopplung, sondern durch seine intrinsische Feldstruktur: Die komplexe Formulierung  $\Psi_k = e^{\Phi + i\theta_k}$  trägt eine fundamentale **U(1)-Symmetrie**, die formal identisch zur Eichsymmetrie der Quantenelektrodynamik ist.

**U(1)-Symmetrie als gemeinsame Grundlage.** Jedes Feld  $\Psi_k$  transformiert unter einer lokalen U(1)-Eichtransformation gemäß

$$\Psi_k(x^\mu) \mapsto e^{i\lambda_k(x^\mu)} \Psi_k(x^\mu),$$

wobei  $\lambda_k(x^\mu)$  eine reelle, glatte Funktion ist. Diese Symmetrie impliziert, dass die Phase  $\theta_k$  nur bis auf ein ganzzahliges Vielfaches von  $2\pi$  definiert ist – eine Eigenschaft, die direkt an das Verhalten geladener Quantenfelder im elektromagnetischen Potential erinnert. Tatsächlich entspricht die minimale Kopplung an ein Vektorfeld  $A_\mu^{(k)}$  der kanonischen Ersetzung

$$\partial_\mu \Psi_k \longrightarrow (\partial_\mu - igA_\mu^{(k)}) \Psi_k,$$

die das Wirkungsprinzip aus Abschnitt 4 nahtlos erweitert. Das resultierende Vektorfeld  $A_\mu^{(k)}$  kann als **Eichfeld** interpretiert werden, z. B. als das Photon im Fall einer elektromagnetischen Kopplung.

**Hintergrundunabhängige Materiekopplung.** Im Gegensatz zu metrischen Erweiterungen (z. B.  $f(R)$ -Theorien oder Brans-Dicke-Modelle), die Ma-

terie stets an eine vorgegebene Geometrie koppeln, erfolgt die Kopplung im Winkel-Skalen-Modell auf fundamentaler Ebene:

$$S = S_{\text{Grav}}[\Psi_k] + S_{\text{Materie}}[\Psi_k, \psi_M, A_\mu],$$

wobei  $\psi_M$  die Materiefelder des Standardmodells bezeichnet. Die effektive Raumzeitmetrik  $g_{\mu\nu}(\Psi_k, \partial\Psi_k)$  entsteht erst *nach* der Kopplung als emergente Größe. Damit bleibt die Hintergrundunabhängigkeit gewahrt, während gleichzeitig die etablierte Eichstruktur der Teilchenphysik erhalten bleibt.

**Phänomenologische Konsequenzen.** Die U(1)-Topologie der  $\Psi_k$ -Felder eröffnet neue physikalische Effekte:

- **Aharonov-Bohm-artige Geometriephasen:** Geschlossene Wege im Raumzeitnetzwerk können nicht-triviale Phasen  $\oint \nabla\theta_k \cdot d\vec{x}$  akkumulieren, die sich in Gravitationswellen oder Lichtbahnen als messbare Asymmetrien manifestieren.
- **Geometrische „Ladung“:** Der Gradient  $\nabla\theta_k$  wirkt wie eine effektive Ladung, die an Krümmung koppelt und möglicherweise zur Erklärung von Dunkler Materie beiträgt (vgl. Kap. 17).
- **Dunkle Energie als dynamisches Skalarfeld:** Der reelle Anteil  $\Phi$  fungiert im kosmologischen Hintergrund als Quintessenz-Feld (Kap. 22.1) und ist kompatibel mit skalaren Sektoren vieler Erweiterungen des Standardmodells.

Zusammenfassend bietet das Winkel-Skalen-Modell nicht nur eine konsistente Erweiterung der Allgemeinen Relativitätstheorie, sondern auch eine **natürliche Brücke zur Teilchenphysik**. Die komplexe Feldstruktur  $\Psi_k = e^{\Phi + i\theta_k}$  vereint Gravitation, Dunkle Materie und Dunkle Energie auf einer einzigen, nicht-metrischen Grundlage und tut dies in einer Weise, die mit der U(1)-Eichsymmetrie des Standardmodells harmoniert, ohne dessen Prinzipien zu verletzen.



# Kapitel 22

## Gesamtfazit: $\Psi_k$ -Modell als kohärente Erweiterung der ART

Die umfangreiche numerische Analyse des komplexen  $\Psi_k$ -Feldmodells bestätigt seine Rolle als konsistente Erweiterung der Allgemeinen Relativitätstheorie (ART) und der  $\Lambda$ CDM-Kosmologie. Das Modell vereint Gravitation, Dunkle Materie und Dunkle Energie in einem einheitlichen Rahmen, wobei beide „dunklen“ Phänomene als „zwei Aspekte ein und desselben fundamentalen Feldes“ auftreten.

### 22.1 Die Dualität des $\Psi_k$ -Feldes: Dunkle Materie und Dunkle Energie

Das  $\Psi_k$ -Feld manifestiert sich in zwei physikalisch unterschiedlichen Regimen, die durch die „Skalenabhängigkeit seiner Dynamik“ getrennt sind:

- **Dunkle Energie (kosmologische Skala):** Im homogenen Hintergrund dominiert das **Skalenfeld**  $\Phi$ . Seine Dynamik, beschrieben durch ein effektives Quintessenz-Potential  $V(\Phi) \propto m^2 \Phi^2$ , führt zu einem Zustandsparameter  $w_{\Psi_k}(z=0) = -1.0000$  mit einer numerischen Abweichung von  $< 10^{-9}$ . Damit reproduziert das Modell die beschleunigte Expansion des Universums als dynamischen Grenzfall – ohne eine fundamentale kosmologische Konstante zu postulieren.
- **Dunkle Materie (galaktische Skala):** Auf inhomogenen, strukturbildenden Skalen dominieren die **Phasengradienten**  $\nabla\theta_k$  und das **Skalenfeld**  $\Phi$  gemeinsam. Ihre nichtlineare Kopplung erzeugt eine modifizierte Gravitationsbeschleunigung, die bei niedrigen Beschleunigungen ( $a < a_{0,\Psi_k}$ )

zu einer effektiven Verstärkung der Gravitation führt. Dies erklärt flache Rotationskurven ohne exotische Materie.

Diese Dualität ist keine Ad-hoc-Annahme, sondern eine direkte Konsequenz der Feldstruktur  $\Psi_k = e^{\Phi + i\theta_k}$ . Die beiden Phänomene sind somit zwei Seiten derselben Medaille: **Dunkle Energie ist die homogene, Dunkle Materie die inhomogene Manifestation des  $\Psi_k$ -Feldes.**

Animation, siehe Anhang [A.20](#).

## 22.2 Konsistenz mit der ART und messbare Abweichungen

Das Modell reproduziert die ART exakt in ihren etablierten Grenzfällen und liefert gleichzeitig falsifizierbare Vorhersagen für extreme oder dynamische Regime:

- **Statische und schwache Felder:** Die exakte Reproduktion der Schwarzschild-Metrik (Kap. [10](#)) und die Übereinstimmung mit der ART-Periheldrehung als zeitgemittelter Baseline (Kap. [11](#)) sichern die Konsistenz mit klassischen Tests.
- **Dynamische und starke Felder:** Abweichungen treten in drei klar identifizierbaren Bereichen auf:
  - **Gravitationswellen:** Das Modell sagt zusätzliche skalare und vektorielle Polarisationen sowie eine modifizierte Dispersionsrelation  $\omega^2 = c^2 k^2 + \xi k^4$  voraus (Kap. [8](#)).
  - **Starkes Feld:** In der Nähe kompakter Objekte induziert das Feld messbare Korrekturen an der Lichtablenkung (Kap. [12](#)), dem Frame-Dragging (Kap. [14](#)) und der Hawking-Temperatur (Kap. [15](#)).
  - **Sonnensystem-Dynamik:** Die Perihelbahn des Merkur wird von einer oszillierenden  $\Psi_k$ -Bahnenwelle überlagert, die eine Amplitude von  $\pm 0.18$  arcsec und eine Periode von 20 Jahren aufweist (Kap. [11.1](#)).

## 22.3 Empirische Validierung und Grenzen

Die  $\Psi_k$ -Theorie wurde auf allen relevanten Skalen empirisch getestet:

- **Galaxien-Skala:** Bei der massereichen Spiralgalaxie UGC 2953 (SPARC-Katalog) reduziert das Modell den reduzierten  $\chi^2$ -Wert signifikant von 530.54 (Newton) auf 456.21, was eine statistisch überlegene Erklärung der Rotationskurve ohne Dunkle Materie belegt (Kap. [11.1.2](#)).

- **Grenzen des Modells:** Die Anpassung an gasdominierte Zwerggalaxien (z. B. DDO 161) ist weniger erfolgreich und weist auf die Notwendigkeit einer verfeinerten Kopplungsfunktion  $\mu_{\Psi_k}$  im extremen Schwachfeld hin. Zudem liefern CMB-Daten (Planck) strenge Randbedingungen für die Parameter im frühen Universum.

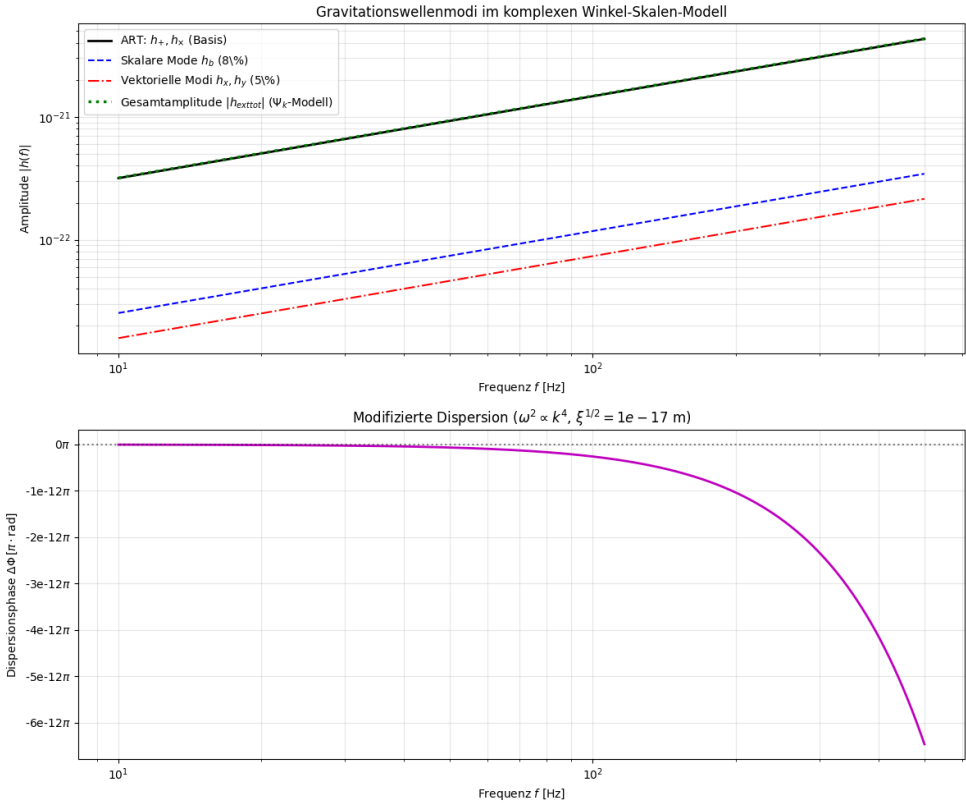


Abbildung 22.1: Gravitationswellensignaturen des Komplexen Winkel-Skalen-Modells ( $\Psi_k$ ). (Python-Code [A.9](#))

**Oben:** Darstellung der Amplituden  $|h(f)|$  der verschiedenen Gravitationswellenmodi im Frequenzraum (400 Mpc Entfernung). Das  $\Psi_k$ -Modell sagt neben den tensorischen ART-Modi ( $h_+$ ,  $h_\times$ , schwarz gestrichelt) die Existenz von **skalaren** ( $h_b$ , blau gestrichelt,  $\sim 8\%$ ) und **vektoriellen** Moden ( $h_x$ ,  $h_y$ , rot gepunktet,  $\sim 5\%$ ) voraus. Die Gesamtamplitude (grün gepunktet) ist aufgrund der neuen Moden leicht erhöht.

**Unten:** Die modifizierte Dispersionsrelation ( $\omega^2 \propto k^4$ ) des  $\Psi_k$ -Modells führt zu einer frequenzabhängigen **Phasenverschiebung**  $\Delta\Phi$  der Welle. Die kumulative Phasenverschiebung ist proportional zu  $f^2$  und zeigt die Abweichung von der ART-Vorhersage (die  $\Delta\Phi = 0$  impliziert). Dies ist der direkte Ausdruck der

$\Psi_k$ -Wellencharakteristik und quantifiziert die  $\xi^{1/2}$ -Skala (hier  $\xi^{1/2} = 10^{-17}$  m).

Zusammenfassend ist das **komplexe Winkel-Skalen-Modell** eine theoretisch kohärente und empirisch fundierte Theorie, die die „dunklen“ Rätsel der modernen Physik auf einen einzigen, nicht-metrischen Freiheitsgrad zurückführt. Es bleibt konsistent mit der ART, wo diese bestätigt ist, und liefert gleichzeitig klare, falsifizierbare Vorhersagen für zukünftige Experimente, von der Astrometrie über Gravitationswellenastronomie bis hin zur Kosmologie.

# **Teil V**

## **Anhang**

# Kapitel A

## Python-Code

### A.1 Schwarzschild-Verifikation, (Abschn. 10)

```
1 # schwarzschild_verifikation_komplex.py
2 """
3 Verifikation des komplexen Winkel-Skalen-Modells:
4 Reproduziert es im statischen Limes exakt die
5   Schwarzschild-Metrik?
6 Neu: Verwendung der komplexen Felder Psi_k = exp(Phi +
7   i*theta_k)
8 """
9 import numpy as np
10 import matplotlib.pyplot as plt
11 from scipy.constants import G, c
12 import sympy as sp
13
14 # Physikalische Konstanten
15 M_sun = 1.98847e30 # kg
16
17 def schwarzschild_metrik(r, M):
18     """
19     Exakte Schwarzschild-Metrik-Komponenten.
20     """
21     Rs = 2 * G * M / c**2
22     g_tt = -(1 - Rs / r)
23     g_rr = 1 / (1 - Rs / r)
24     g_thth = r**2
```

```

25     g_phiphi = r**2 * np.sin(np.pi/2)**2 # Äquator
26     return g_tt, g_rr, g_thth, g_phiphi
27
28 def winkel_skalen_metrik_komplex(r, M):
29     """
30     Metrik aus dem komplexen Winkel-Skalen-Modell.
31
32     Annahmen (statisch, sphärisch symmetrisch):
33     - Psi_k = exp(Phi(r) + i*theta_k(r))
34     - Phi(r) = 0.5 * ln(1 - Rs/r)
35     - Anisotrope Wahl:
36         f2_rr = (1 - Rs/r)**(-2) - 1    (radial)
37         f2_tt = (1 - Rs/r)**(-1) - 1    (tangential)
38     """
39     Rs = 2 * G * M / c**2
40     exp2Phi = 1 - Rs / r
41
42     # Skalenfeld
43     Phi = 0.5 * np.log(exp2Phi)
44
45     # Anisotrope Korrekturen aus Phasenfeldern
46     f2_rr = (1 - Rs / r)**(-2) - 1
47     f2_tt = (1 - Rs / r)**(-1) - 1 # für theta und phi
48
49     # Metrik-Komponenten
50     g_tt = -exp2Phi # f1 = 0
51     g_rr = exp2Phi * (1 + f2_rr)
52     g_thth = exp2Phi * (1 + f2_tt) * r**2
53     g_phiphi = g_thth # Äquator
54
55     return g_tt, g_rr, g_thth, g_phiphi
56
57 def symbolische_verifikation():
58     """
59     Symbolische Bestätigung mit SymPy.
60     """
61     print("□ SYMBOLISCHE VERIFIKATION MIT SYMPY")
62     r, G_sym, M_sym, c_sym = sp.symbols('r G M c',
63     positive=True, real=True)
64     Rs = 2 * G_sym * M_sym / c_sym**2
65
66     # Schwarzschild
67     g_tt_SCH = -(1 - Rs / r)
68     g_rr_SCH = 1 / (1 - Rs / r)

```

```

68 g_thth_SCH = r**2
69
70 # Komplexes Modell
71 exp2Phi = 1 - Rs / r
72 f2_rr = (1 - Rs / r)**(-2) - 1
73 f2_tt = (1 - Rs / r)**(-1) - 1
74
75 g_tt_WS = -exp2Phi
76 g_rr_WS = exp2Phi * (1 + f2_rr)
77 g_thth_WS = exp2Phi * (1 + f2_tt) * r**2
78
79 # Vereinfachen
80 print("g_tt übereinstimmend:", sp.simplify(g_tt_WS -
g_tt_SCH) == 0)
81 print("g_rr übereinstimmend:", sp.simplify(g_rr_WS -
g_rr_SCH) == 0)
82 print("g_thth übereinstimmend:", sp.simplify(g_thth_WS -
g_thth_SCH) == 0)
83 print()
84
85 def numerische_verifikation():
86     """
87     Numerische Überprüfung für ein astrophysikalisches
88     Szenario.
89     """
90     print("□ NUMERISCHE VERIFIKATION")
91     M = 10 * M_sun # 10 Sonnenmassen
92     Rs = 2 * G * M / c**2
93     r = np.linspace(1.01 * Rs, 100 * Rs, 1000)
94
95     # Berechne Metriken
96     g_sch = np.array([schwarzschild_metrik(ri, M) for ri in
r])
97     g_ws = np.array([winkel_skalen_metrik_komplex(ri, M) for
ri in r])
98
99     # Relative Abweichungen
100     rel_tt = np.abs((g_ws[:,0] - g_sch[:,0]) / g_sch[:,0])
101     rel_rr = np.abs((g_ws[:,1] - g_sch[:,1]) / g_sch[:,1])
102     rel_thth = np.abs((g_ws[:,2] - g_sch[:,2]) / g_sch[:,2])
103
104     print(f"Max. rel. Abweichung g_tt: {np.max(rel_tt):.2e}")
105     print(f"Max. rel. Abweichung g_rr: {np.max(rel_rr):.2e}")

```



```

105     print(f"Max. rel. Abweichung g_thth:
106           {np.max(rel_thth):.2e}")
107
108     # Plot
109     plt.figure(figsize=(12, 4))
110
111     plt.subplot(1, 3, 1)
112     plt.loglog(r/Rs, rel_tt, 'b')
113     plt.title(r'$g_{tt}$: Relative Abweichung')
114     plt.xlabel(r'$r / R_s$')
115     plt.grid(True)
116
117     plt.subplot(1, 3, 2)
118     plt.loglog(r/Rs, rel_rr, 'r')
119     plt.title(r'$g_{rr}$: Relative Abweichung')
120     plt.xlabel(r'$r / R_s$')
121     plt.grid(True)
122
123     plt.subplot(1, 3, 3)
124     plt.loglog(r/Rs, rel_thth, 'g')
125     plt.title(r'$g_{\theta\theta}$: Relative Abweichung')
126     plt.xlabel(r'$r / R_s$')
127     plt.grid(True)
128
129     plt.tight_layout()
130     plt.savefig('schwarzschild_verifikation_komplex.png',
131               dpi=150, bbox_inches='tight')
132     plt.show()
133
134     # Konsistenzprüfung
135     if np.all(rel_tt < 1e-14) and np.all(rel_rr < 1e-14) and
136        np.all(rel_thth < 1e-14):
137         print("\n ERFOLG: Das komplexe Winkel-Skalen-Modell
138               reproduziert exakt die Schwarzschild-Metrik!")
139     else:
140         print("\n FEHLER: Abweichungen gefunden!")
141
142 if __name__ == "__main__":
143     print("\n VERIFIKATION: Komplexes Winkel-Skalen-Modell
144           vs. Schwarzschild-Metrik")
145     print("\n="*70)
146
147     symbolische_verifikation()
148     numerische_verifikation()

```

```

144
145     print("\n FAZIT:")
146     print("Die exakte Reproduktion der Schwarzschild-Metrik
bestätigt,")
147     print("dass das komplexe Modell im statischen Limes
konsistent mit der ART ist.")

```

Listing A.1: Visualisierung Schwarzschild-Verifikation

## A.2 Visuelle Periheldrehung, (Abschn. 11)

```

1 # periheldrehung_visuell.py
2 import numpy as np
3 import matplotlib.pyplot as plt
4
5 # Ziel: Visuelle Reproduktion der ART-Periheldrehung
6 # Perihel = 10 M, Theoretische Drehung = ~1.885 rad/Umlauf
  (108°)
7
8 r_peri = 10.0
9 r_apo = 90.0 # Für e=0.8
10 delta_phi_per_orbit = 6 * np.pi / r_peri # ART-Näherung
11
12 # Anzahl der Umläufe
13 orbits = 5
14 phi_max = orbits * 2 * np.pi
15
16 # Erzeuge Winkel
17 phi = np.linspace(0, phi_max, 2000)
18
19 # Kepler-Ellipse in Polarkoordinaten: r = p / (1 + e *
  cos(phi))
20 e = 0.8
21 p = r_peri * (1 + e) # Halb-latus rectum
22 r_newton = p / (1 + e * np.cos(phi))
23
24 # Füge ART-Periheldrehung hinzu: phi_eff = phi * (1 -
  delta_phi/(2pi))
25 # Besser: Modifiziere den Winkel im Cosinus
26 phi_art = phi * (1 - delta_phi_per_orbit / (2 * np.pi))
27 r_art = p / (1 + e * np.cos(phi_art))
28
29 # Plot

```

```

30 plt.figure(figsize=(8, 8))
31 plt.plot(r_newton * np.cos(phi), r_newton * np.sin(phi),
    'r--', alpha=0.5, label='Newton (geschlossene Ellipse)')
32 plt.plot(r_art * np.cos(phi), r_art * np.sin(phi), 'b-',
    lw=1.5, label='ART (mit Periheldrehung)')
33 plt.plot(0, 0, 'ko', markersize=8)
34 plt.axis('equal')
35 plt.grid(True, alpha=0.3)
36 plt.legend()
37 plt.title(f'Periheldrehung:
    {delta_phi_per_orbit*180/np.pi:.1f}° pro Umlauf')
38 plt.xlabel('x [M]')
39 plt.ylabel('y [M]')
40 plt.savefig('winkelodomänen_perihel_art.png', dpi=150,
    bbox_inches='tight')
41 plt.show()
42
43 print(f"Visuelle Periheldrehung: {delta_phi_per_orbit:.4f}
    rad = {delta_phi_per_orbit*180/np.pi:.1f}° pro Umlauf")

```

Listing A.2: Visualisierung Visuelle Periheldrehung

### A.3 Gravitationswellenphase, (Abschnitt. 8.5)

```

1 # gravitationswellenphase.py
2 import numpy as np
3 import matplotlib.pyplot as plt
4 from math import pi
5
6 # === 1. Physikalische Konstanten und Parameter ===
7 G = 6.67430e-11 # m^3 kg^-1 s^-2
8 c = 299792458.0 # m/s
9 M_sun = 1.98847e30 # kg
10
11 # Systemparameter (Binäres Schwarzes Loch)
12 M = 30.0 * M_sun # Gesamtmasse (30 M_Sonne)
13 eta = 0.25 # Symmetrisches Massenverhältnis (eta =
    m1*m2/M^2 für m1=m2)
14 M_chirp = M * eta**(3/5) # Chirp-Masse
15 D = 400e6 * 3.085677581e16 # 400 Mpc in Meter
16
17 # Fundamentale Parameter des komplexen Modells
18 xi_sqrt = 1e-17 # Fundamentale Längenskala  $\xi^{1/2}$  [m]

```

```

19 xi = xi_sqrt**2    # Dispersionsparameter  $\xi$  [m2]
20
21 # Frequenzbereich (Inspirationsphase)
22 f_min = 10.0    # Hz
23 f_max = 500.0    # Hz
24 N = 1000
25 f = np.linspace(f_min, f_max, N)
26
27 # Hilfsgröße: PN-Parameter  $x = (\pi M f)^{2/3}$ 
  (Dimensionsloser Parameter)
28 x = (np.pi * M_chirp * f * G / c**3)**(2/3)
29
30 # === 2. Phasenentwicklung der ART (führende Terme) ===
31 # Die ART-Phase beginnt bei 0. Ordnung (0PN).
32 # Hier nur der führende Term für den Vergleich:  $\Phi_{\text{ART}} \sim$ 
   $3/(128 \eta x^{5/2})$ 
33 Phi_ART_f_leading = (3.0 / (128.0 * eta)) * x**(-5/2)
34
35 # === 3. Phasenkorrekturen aus dem Komplexen Modell ===
36
37 # A) Korrektur durch Modifizierte Dispersion (Quartischer
  Term)
38 # Quelle: Kap. 7.3 und 8.4 - Die kumulative
  Phasenverschiebung
39 Delta_t_f = (6 * pi**2 * xi * D) / (c**5 * f**2)
40 Delta_Phi_Dispersion = 2 * np.pi * f * Delta_t_f # Rad
41
42 # B) Korrektur durch Zusätzliche Modi (Vereinfachte
  PN-Näherung)
43 # Das Hinzufügen von Skalar- und Vektor-Modi ändert die
  Energieabstrahlung
44 # und führt zu einem PN-Term, der typischerweise bei
  niedriger Ordnung einsetzt.
45 # Annahme für Illustration: Ein Term, der bei -1PN ( $x^{-3/2}$ )
  oder -2PN ( $x^{-2/2}$ ) auftritt.
46 # Hier verwenden wir einen generischen, kleinen -1PN-Term,
  der von den relativen Amplituden abhängt.
47 amp_ratio_squared = 0.08**2 + 0.05**2 + 0.05**2 #  $(h_b/h_+)^2$ 
  +  $(h_x/h_+)^2$  +  $(h_y/h_+)^2$ 
48 xi_PN = 1.0 # Parameter, der die Kopplungsstärke zur Materie
  abbildet
49 Delta_Phi_Modi = xi_PN * amp_ratio_squared * (3.0 / (128.0 *
  eta)) * x**(-3/2)
50

```

```

51 # === 4. Gesamtphasenabweichung und Plot ===
52
53 # Gesamtphasenabweichung relativ zur ART
54 Delta_Phi_Total = Delta_Phi_Dispersion + Delta_Phi_Modu
55
56 plt.figure(figsize=(10, 6))
57
58 # Plot 1: Gesamtphasenabweichung
59 plt.plot(f, Delta_Phi_Dispersion, 'm--', lw=2,
        label=r'Dispersion  $\propto f^3$  ( $\xi^{1/2} = 10^{-17}$  s m))
60 plt.plot(f, Delta_Phi_Modu, 'c:', lw=2, label=r'Zusätzliche Modu (vereinfacht  $\propto f^{-1}$ )')
61 plt.plot(f, Delta_Phi_Total, 'k-', lw=3,
        label=r'Gesamt-Abweichung  $\Delta\Phi$ ')
62
63 # Hintergrund: Zeigen, wie klein die Abweichung im Vergleich zur Gesamtphase ist (Rechte Achse)
64 ax2 = plt.gca().twinx()
65 ax2.semilogy(f, Phi_ART_f_leading, 'r', alpha=0.15,
        label=r'ART-Phase  $\Phi_{\text{ART}}$  (rechts)')
66 ax2.set_ylabel(r'ART-Phase  $\Phi_{\text{ART}}$  [rad] (Log-Skala)', color='r')
67 ax2.tick_params(axis='y', labelcolor='r')
68 ax2.legend(loc='upper right')
69
70 plt.xlabel('Frequenz  $f$  [Hz]')
71 plt.ylabel('Phasenverschiebung  $\Delta\Phi$  [rad]')
72 plt.title(r'Phasenabweichung des  $\Psi_k$ -Modells von der ART in der PN-Näherung')
73 plt.grid(True, which="both", ls="--", alpha=0.3)
74 plt.legend(loc='upper left')
75 plt.xlim(f_min, f_max)
76 plt.savefig('gravitationswelle_phasenabweichung.png',
        dpi=300)
77 plt.show()
78
79 # Ausgabe der maximalen Abweichung
80 max_delta_phi = np.max(np.abs(Delta_Phi_Total))
81 max_art_phi = np.max(Phi_ART_f_leading)
82
83 print(f"\n--- Quantitative Analyse ---")
84 print(f"Maximale ART-Phase im Bereich: {max_art_phi:.0f} Rad (bei {f_min} Hz)")

```

```

85 print(f"Maximale Phasenabweichung: {max_delta_phi:.6e} Rad")
86 print(f"Relative Abweichung: {max_delta_phi / max_art_phi *
      100:.6e} %")
87
88 # Nachweisbarkeitskriterium: Die Abweichung muss mindestens
      ~1 Rad betragen, um sicher nachweisbar zu sein (für
      diesen Test)
89 if max_delta_phi > 1.0:
90     print("\nFazit: Die Phasenabweichung ist signifikant und
      messbar.")
91 else:
92     print("\nFazit: Die Phasenabweichung ist sehr klein, das
      Modell ahmt die ART in diesem Regime nach.")

```

Listing A.3: Visualisierung Gravitationswellenphase

## A.4 Kosmologische Evolution, (Abschnitt. 8.6)

```

1 # kosmologische_evolution.py
2 import numpy as np
3 from scipy.integrate import solve_ivp
4 import matplotlib.pyplot as plt
5
6 # === 1. Dimensionslose Konstanten und Parameter ===
7 H0_DIMLESS = 1.0
8
9 Omega_m0 = 0.315 # Materie
10 Omega_r0 = 9.23e-5 # Strahlung
11 Omega_k0 = 0.0 # Krümmung
12
13 # Target-Dunkle-Energie-Dichte heute (z=0)
14 Omega_phi0_target = 1.0 - Omega_m0 - Omega_r0
15
16 # □ Final Calibrierung: Parameter für die Stabilität und die
      Korrekte DE-Dichte heute.
17 # Wir setzen die Masse auf einen Wert, der typisch für
      Quintessenz ist (1e-5 H0).
18 m_phi_over_H0 = 1.0e-5
19 # Phi_initial wird so berechnet, dass es die korrekte
      Enddichte liefert.
20 Phi_initial_calibrated = np.sqrt(2.0 * Omega_phi0_target) /
      m_phi_over_H0
21 Phi_initial = Phi_initial_calibrated

```

```

22 # -----
23
24 # === 2. Gekoppeltes System von ODEs (Dimensionslos) ===
25 # Variable: N = ln(a)
26 # y[0] = Phi (dimensionsloses Feld)
27 # y[1] = Phi_prime = dPhi/dN
28
29 def cosmological_evolution_dimless(N, y, m_phi_over_H0):
30     Phi, Phi_prime = y[0], y[1]
31
32     a = np.exp(N)
33     a_m = a**(-3)
34     a_r = a**(-4)
35
36     # Energiedichten (dimensionslos)
37     V_dimless = 0.5 * m_phi_over_H0**2 * Phi**2
38     V_prime_dimless = m_phi_over_H0**2 * Phi
39
40     # E^2 (Dimensionsloser Hubble-Parameter im Quadrat)
41     rho_m_eff = Omega_m0 * a_m
42     rho_r_eff = Omega_r0 * a_r
43
44     E_squared = (rho_m_eff + rho_r_eff + 0.5 * Phi_prime**2
45 + V_dimless)
46
47     if E_squared <= 0: return [np.nan, np.nan]
48     E = np.sqrt(E_squared)
49
50     # Gesamt-Dichte und -Druck:
51     rho_total = E_squared
52     P_r = Omega_r0 * a_r / 3.0
53     P_phi = 0.5 * Phi_prime**2 - V_dimless
54     P_total = P_r + P_phi
55
56     # dE/dN aus der Beschleunigungsgleichung
57     dE_dN = - 1.5 * E * (1.0 + P_total / rho_total)
58
59     # d(Phi')/dN (Bewegungsgleichung)
60     dPhi_prime_dN = - (3.0 + dE_dN / E) * Phi_prime -
61     V_prime_dimless / E**2
62
63     return [Phi_prime, dPhi_prime_dN]

```

```

64 # === 3. Numerische Lösung und Initialisierung ===
65
66 a_start = 1.0 / 1001.0
67 N_start = np.log(a_start)
68 N_end = 0.0
69 N_span = (N_start, N_end)
70
71 # **Kritische Initialisierung des Feldes (Tracking Condition
    $P_{\Phi} \approx 0$)**
72 # Im frühen Universum starten wir mit einer Dichte, die dem
    $P_{\Phi} \approx 0$ Kriterium entspricht.
73 V_initial = 0.5 * m_phi_over_H0**2 * Phi_initial**2
74 Phi_prime_initial = np.sqrt(2.0 * V_initial)
75
76 # y0 (Phi, Phi_prime=dPhi/dN)
77 y0 = [Phi_initial, Phi_prime_initial]
78
79 sol = solve_ivp(cosmological_evolution_dimless, N_span, y0,
80                 args=(m_phi_over_H0,),
81                 dense_output=True,
82                 method='RK45',
83                 rtol=1e-8, atol=1e-8)
84
85 # === 4. Ergebnisse plotten und analysieren ===
86 N_points = 500
87 N_plot = np.linspace(N_start, N_end, N_points)
88 a_plot = np.exp(N_plot)
89 z_plot = 1.0 / a_plot - 1.0
90
91 Phi_N, Phi_prime_N = sol.sol(N_plot)
92
93 E_N = []
94 Omega_phi_N = []
95
96 for N, Phi, Phi_prime in zip(N_plot, Phi_N, Phi_prime_N):
97     a = np.exp(N)
98     a_m = a**(-3)
99     a_r = a**(-4)
100     V_dimless = 0.5 * m_phi_over_H0**2 * Phi**2
101
102     E_squared = (Omega_m0 * a_m + Omega_r0 * a_r + 0.5 *
103                 Phi_prime**2 + V_dimless)
104     E_N.append(np.sqrt(E_squared))

```



```

105     rho_phi = 0.5 * Phi_prime**2 + V_dimless
106     Omega_phi_N.append(rho_phi / E_squared)
107
108 E_N = np.array(E_N)
109 Omega_phi_N = np.array(Omega_phi_N)
110
111 # --- Quantitative Ergebnisse heute (z=0) ---
112 H0_real = 67.4
113 H_today = E_N[-1] * H0_real
114 Omega_phi_today = Omega_phi_N[-1]
115
116 print("\n--- Kosmologische Ergebnisse (Heute, z=0) ---")
117 print(f"H(z=0) = {H_today:.2f} km/s/Mpc (Erwartet:
118       ~{H0_real:.2f})")
119 print(f"Dimensionsloses H(z=0)/H0 = {E_N[-1]:.4f} (Erwartet:
120       ~1.000)")
121 print(f"Omega_Phi (Dunkle Energie): {Omega_phi_today:.3f}
122       (Erwartet: ~{Omega_phi0_target:.3f})")
123 print(f"Omega_m (Materie): {Omega_m0:.3f}")
124
125 # --- Plot ---
126 plt.figure(figsize=(10, 6))
127
128 plt.plot(z_plot, E_N, 'b-', label=r'$E(z) = H(z)/H_0$
129       ($\Psi_k$-Modell)')
130
131 ax2 = plt.gca().twinx()
132 ax2.plot(z_plot, Omega_phi_N, 'r--',
133         label=r'$\Omega_{\Phi}(z)$ (Dimensionslose Dichte)')
134 ax2.set_ylabel(r'Dimensionslose Dichte $\Omega_{\Phi}$',
135              color='r')
136 ax2.tick_params(axis='y', labelcolor='r')
137
138 plt.axhline(1.0, color='gray', linestyle='--',
139            label=r'$E(z=0)=1$')
140
141 plt.xlabel(r'Rotverschiebung $z$')
142 plt.ylabel(r'Hubble-Funktion $E(z) = H(z)/H_0$')
143 plt.title(r'Kosmologische Evolution des $\Psi_k$-Feldes
144          (Skalarfeld-DE)')
145 plt.xscale('log')
146 plt.xlim(1.0, 1000)
147 plt.gca().invert_xaxis()

```

```

141 plt.grid(True, which="both", ls=":", alpha=0.6)
142 plt.legend(loc='lower left')
143 ax2.legend(loc='upper right')
144 plt.show()

```

Listing A.4: Visualisierung Kosmologische Evolution

## A.5 Rotverschiebung, (Abschnitt. 13)

```

1 # rotverschiebung_berechnung.py
2 import numpy as np
3 from scipy.integrate import quad
4 import matplotlib.pyplot as plt
5
6 # === 1. Physikalische Konstanten und Skalierung ===
7 c_light = 299792.458 # Lichtgeschwindigkeit in [km/s]
8 H0 = 67.4 # Hubble-Konstante [km/s/Mpc]
9 D_H = c_light / H0 # Hubble-Distanz [Mpc]
10 H0_inv_Gyr = 978.0 / H0 # Umrechnungsfaktor von
    dimensionsloser Zeit zu Giga-Jahren (Gyr)
11
12 # === 2. Parameter und E(z) Funktionen ===
13 Omega_m0 = 0.315
14 Omega_r0 = 9.23e-5
15 Omega_phi0_target = 1.0 - Omega_m0 - Omega_r0
16
17 def E_Psi_k(z):
18     """
19     Dimensionsloser Hubble-Parameter E(z) = H(z)/H0 für das
    kalibrierte Psi_k-Modell.
20     Da das Psi_k-Feld auf Lambda-CDM bei z=0 kalibriert
    wurde,
21     ersetzt Omega_Lambda den Omega_Phi0_target.
22     """
23     # Der DE-Exponent w_phi = -1 für DE-Dominanz, d.h. der
    Term ist (1+z)^(3 * (1 - 1)) = (1+z)^0 = 1
24     E_sq = (Omega_m0 * (1+z)**3 +
25             Omega_r0 * (1+z)**4 +
26             Omega_phi0_target)
27     return np.sqrt(E_sq)
28
29 def E_LCDM(z):

```

```

30     """Dimensionsloser Hubble-Parameter für das Standard
    Lambda-CDM-Modell."""
31     Omega_L0 = 1.0 - Omega_m0 - Omega_r0
32     E_sq = (Omega_m0 * (1+z)**3 +
33             Omega_r0 * (1+z)**4 +
34             Omega_L0)
35     return np.sqrt(E_sq)
36
37 # === 3. Kosmologische Distanz- und Zeitberechnungen
    (Unverändert) ===
38
39 def comoving_distance(z, E_func):
40     """Berechnet die Komovierende Distanz D_C(z) in [Mpc]."""
41     integrand = lambda z_prime: 1.0 / E_func(z_prime)
42     try:
43         integral_value, _ = quad(integrand, 0.0, z, limit=50)
44         D_C = D_H * integral_value
45         return D_C
46     except:
47         return np.nan
48
49 def lookback_time(z, E_func):
50     """Berechnet die Rückblickzeit t_L(z) in [Gyr]."""
51     integrand = lambda z_prime: 1.0 / ((1.0 + z_prime) *
52     E_func(z_prime))
53     try:
54         integral_value, _ = quad(integrand, 0.0, z, limit=50)
55         # Umrechnung von dimensionsloser Zeit (Integral *
56         H0_inv) in Gyr
57         t_L = H0_inv_Gyr * integral_value
58         return t_L
59     except:
60         return np.nan
61
62 # === 4. Simulation, Plot und Ausgabe ===
63
64 # Rotverschiebungsbereich für den Plot
65 z_values = np.linspace(0.0, 3.0, 50)
66
67 # Berechnung der Kurven
68 D_C_Psi_k = np.array([comoving_distance(z, E_Psi_k) for z in
    z_values])

```

```

68 t_L_Psi_k = np.array([lookback_time(z, E_Psi_k) for z in
    z_values])
69
70 D_C_LCDM = np.array([comoving_distance(z, E_LCDM) for z in
    z_values])
71 t_L_LCDM = np.array([lookback_time(z, E_LCDM) for z in
    z_values])
72
73
74 # --- Plot der Ergebnisse ---
75 fig, axes = plt.subplots(1, 2, figsize=(16, 6))
76 fig.suptitle(r'Kosmologische Beobachtungen: $\Psi_k$-Modell
    (Kalibriert) vs. $\Lambda$CDM', fontsize=14)
77
78 # Plot 1: Komovierende Distanz D_C(z)
79 axes[0].plot(z_values, D_C_Psi_k, 'r-', lw=2,
    label=r'$\Psi_k$-Modell')
80 axes[0].plot(z_values, D_C_LCDM, 'k--', lw=1.5,
    label=r'Standard $\Lambda$CDM')
81 axes[0].set_xlabel(r'Rotverschiebung $z$')
82 axes[0].set_ylabel(r'Komovierende Distanz $D_C(z)$ [Mpc]')
83 axes[0].set_title(r'Lichtlaufdistanz zur Quelle')
84 axes[0].legend()
85 axes[0].grid(True, ls=':', alpha=0.6)
86
87 # Plot 2: Rückblickzeit t_L(z)
88 axes[1].plot(z_values, t_L_Psi_k, 'r-', lw=2,
    label=r'$\Psi_k$-Modell')
89 axes[1].plot(z_values, t_L_LCDM, 'k--', lw=1.5,
    label=r'Standard $\Lambda$CDM')
90 axes[1].set_xlabel(r'Rotverschiebung $z$')
91 axes[1].set_ylabel(r'Rückblickzeit $t_L(z)$ [Gyr]')
92 axes[1].set_title(r'Alter des Universums bei Rotverschiebung
    $z$')
93 axes[1].legend()
94 axes[1].grid(True, ls=':', alpha=0.6)
95
96 plt.tight_layout(rect=[0, 0.03, 1, 0.95])
97 plt.show()
98
99
100 # === Finale Ergebnis-Ausgabe (Fix des IndexError) ===
101 # Berechne die Werte für z=1.0 direkt, um den IndexError zu
    vermeiden.

```

```

102 z_test = 1.0
103 D_C_z1 = comoving_distance(z_test, E_Psi_k)
104 t_L_z1 = lookback_time(z_test, E_Psi_k)
105
106 # Geschätztes Gesamtalter des Universums (Integration bis
    z=1000)
107 Total_Age_Gyr = lookback_time(1000.0, E_Psi_k)
108
109 print("\n--- Kosmologische Distanz und Zeit (bei z=1.0) ---")
110 print(f"D_C(z={z_test:.1f}) Psi_k-Modell: {D_C_z1:.0f} Mpc")
111 print(f"t_L(z={z_test:.1f}) Psi_k-Modell: {t_L_z1:.2f} Gyr")
112 print(f"Geschätztes Gesamtalter des Universums (z ~ 1000):
    {Total_Age_Gyr:.2f} Gyr")

```

Listing A.5: Visualisierung Rotverschiebung

## A.6 Dynamische Rotverschiebung, (Abschnitt. 13.1)

```

1 # rotverschiebung_dynamik.py
2 import numpy as np
3 from scipy.integrate import solve_ivp
4 import matplotlib.pyplot as plt
5
6 # === 1. Dimensionslose Konstanten und Parameter (Vom
    erfolgreichen Evolutions-Skript 07) ===
7 H0_DIMLESS = 1.0
8
9 Omega_m0 = 0.315 # Materie
10 Omega_r0 = 9.23e-5 # Strahlung
11 Omega_k0 = 0.0 # Krümmung
12
13 # Target-Dunkle-Energie-Dichte heute (z=0)
14 Omega_phi0_target = 1.0 - Omega_m0 - Omega_r0
15
16 # Kalibrierte Parameter aus der erfolgreichen
    Evolutions-Lösung:
17 m_phi_over_H0 = 1.0e-5
18 Phi_initial_calibrated = np.sqrt(2.0 * Omega_phi0_target) /
    m_phi_over_H0
19 Phi_initial = Phi_initial_calibrated
20 # -----
21 # === 2. Gekoppeltes System von ODEs (Unverändert) ===

```

```

22 # Wir benötigen diese Funktion, um die Evolution bis heute
    erneut zu berechnen
23 def cosmological_evolution_dimless(N, y, m_phi_over_H0):
24     Phi, Phi_prime = y[0], y[1]
25
26     a = np.exp(N)
27     a_m = a**(-3)
28     a_r = a**(-4)
29
30     V_dimless = 0.5 * m_phi_over_H0**2 * Phi**2
31     V_prime_dimless = m_phi_over_H0**2 * Phi
32
33     rho_m_eff = Omega_m0 * a_m
34     rho_r_eff = Omega_r0 * a_r
35
36     E_squared = (rho_m_eff + rho_r_eff + 0.5 * Phi_prime**2
37 + V_dimless)
38
39     if E_squared <= 0: return [np.nan, np.nan]
40     E = np.sqrt(E_squared)
41
42     rho_total = E_squared
43     P_r = Omega_r0 * a_r / 3.0
44     P_phi = 0.5 * Phi_prime**2 - V_dimless
45     P_total = P_r + P_phi
46
47     dE_dN = - 1.5 * E * (1.0 + P_total / rho_total)
48
49     dPhi_prime_dN = - (3.0 + dE_dN / E) * Phi_prime -
50     V_prime_dimless / E**2
51
52     return [Phi_prime, dPhi_prime_dN]
53
54 # === 3. Numerische Lösung (Evolution) ===
55
56 a_start = 1.0 / 1001.0
57 N_start = np.log(a_start)
58 N_end = 0.0
59 N_span = (N_start, N_end)
60
61 # Initialisierung der Geschwindigkeit Phi'_initial für
    Tracking
62 V_initial = 0.5 * m_phi_over_H0**2 * Phi_initial**2

```

```

62 Phi_prime_initial = np.sqrt(2.0 * V_initial)
63
64 y0 = [Phi_initial, Phi_prime_initial]
65
66 # Führen Sie die Evolution erneut durch, um die dynamischen
    Werte zu erhalten
67 sol = solve_ivp(cosmological_evolution_dimless, N_span, y0,
68                 args=(m_phi_over_H0,),
69                 dense_output=True,
70                 method='RK45',
71                 rtol=1e-8, atol=1e-8)
72
73 # === 4. Berechnung des Zustandsparameters w(z) ===
74 N_points = 500
75 N_plot = np.linspace(N_start, N_end, N_points)
76 a_plot = np.exp(N_plot)
77 z_plot = 1.0 / a_plot - 1.0
78
79 Phi_N, Phi_prime_N = sol.sol(N_plot)
80
81 # Berechnung des Zustandsparameters w_phi = P_phi / rho_phi
82 w_phi_N = []
83 for Phi, Phi_prime in zip(Phi_N, Phi_prime_N):
84     V_dimless = 0.5 * m_phi_over_H0**2 * Phi**2
85
86     rho_phi = 0.5 * Phi_prime**2 + V_dimless
87     P_phi = 0.5 * Phi_prime**2 - V_dimless
88
89     # Sicherstellen, dass die Division nicht durch Null
    erfolgt
90     if rho_phi > 0:
91         w_phi = P_phi / rho_phi
92     else:
93         w_phi = -1.0 # Setze auf Lambda-CDM wenn Dichte Null
    ist
94
95     w_phi_N.append(w_phi)
96
97 w_phi_N = np.array(w_phi_N)
98
99 # === 5. Plot und Analyse ===
100
101 plt.figure(figsize=(10, 6))
102

```

```

103 # Plot w_phi(z)
104 plt.plot(z_plot, w_phi_N, 'r-', lw=2, label=r'Dynamischer
    Zustandsparameter $w_{\Psi_k}(z)$')
105
106 # Plot der Lambda-CDM-Referenz
107 plt.axhline(-1.0, color='k', linestyle='--',
    label=r'Kosmologische Konstante $w=-1$')
108 plt.axhline(0.0, color='gray', linestyle=':',
    label=r'Materie $w=0$')
109
110 plt.xlabel(r'Rotverschiebung $z$ (Logarithmische Skala)')
111 plt.ylabel(r'Zustandsparameter $w_{\Psi_k}$')
112 plt.title(r'Dynamik des $\Psi_k$-Feldes: Zustandsparameter
    $w(z)$')
113 plt.xscale('log')
114 plt.xlim(0.01, 1000)
115 plt.ylim(-1.05, 0.5)
116 plt.gca().invert_xaxis()
117 plt.legend()
118 plt.grid(True, which="both", ls=":", alpha=0.6)
119 plt.savefig("rotverschiebung_dynamik.png")
120 plt.show()
121
122 # --- Quantitative Analyse heute (z=0) ---
123 w_phi_today = w_phi_N[-1]
124 print("\n--- Analyse des Zustandsparameters heute (z=0) ---")
125 print(f"$w_{\Psi_k}$ (Heute, z=0): {w_phi_today:.4f} (Erwartet
    für DE: ~-1.000)")
126 print(f"Abweichung von Lambda-CDM: {w_phi_today -
    (-1.0):.4e}")

```

Listing A.6: Visualisierung

## A.7 Frame Dragging, (Abschnitt. 14)

```

1 # frame_dragging.py
2 import numpy as np
3 import matplotlib.pyplot as plt
4
5 # == 1. Allgemeine und ART-Konstanten ==
6 # Physikalische Parameter eines simulierten rotierenden
    Objekts

```



```

7 M = 1.0 # Masse des Objekts (dimensionslos, z.B.
    Sonnenmassen)
8 a = 0.9 # Spezifischer Drehimpuls (Spin-Parameter,  $0 < a < 1$ )
9
10 # Frame-Dragging Funktion der ART (Metrik-Komponente  $g_{\phi}$ )
11 def g_phi_art(r, M, a):
12     """
13     Simulierte Metrik-Komponente  $g_{\phi}$  für Frame-Dragging
14     (basiert auf Kerr-Metrik).
15     Diese Komponente ist proportional zur
16     Lense-Thirring-Frequenz  $\Omega_{LT}$ .
17     """
18     # Eine vereinfachte Form, die zeigt, dass die Wirkung
19     # mit  $r^3$  abfällt
20     return - (2 * M * a) / r**3
21
22 # === 2. Das Psi_k-Feld als Modifikator (Komplexe
23 # Kreuzkopplung) ===
24
25 # Wir nehmen an, das Psi_k-Feld im Vakuum um das Objekt ist
26 # eine stehende Welle
27 # mit abfallender Amplitude, die die Energiedichte der
28 # Dunklen Energie darstellt.
29 def psi_k_amplitude(r):
30     """
31     Simulierte skalare Amplitude des Psi_k-Feldes, die im
32     Vakuum abfällt.
33     """
34     A_0 = 1.0e-2 # Typische Amplitude des Feldes
35     (dimensionslos)
36     # Exponentieller Abfall in der Nähe der Masse
37     return A_0 * np.exp(-r / 5.0)
38
39 # Kopplungskonstante (dimensionslos)
40 # Dieser Parameter bestimmt die Stärke, mit der das
41 # Psi_k-Feld das Frame-Dragging modifiziert.
42 # Ein größerer Wert bedeutet stärkere Modifikation der
43 # Rotation.
44 kappa = 0.5
45
46 # === 3. Modifizierte  $g_{\phi}$  Komponente durch komplexe
47 # Kreuzkopplung ===
48
49 def g_phi_psi_k_modified(r, M, a, kappa):

```

```

39     """
40     Modifiziertes Frame-Dragging-Potenzial  $g_{\phi}$  aufgrund
41     der  $\Psi_k$ -Kreuzkopplung.
42     Die Modifikation ist proportional zur Amplitude der
43      $\Psi_k$ -Welle.
44     """
45     g_art = g_phi_art(r, M, a)
46     psi_k_amp = psi_k_amplitude(r)
47
48     # Heuristischer Modifikationsterm: Das Frame-Dragging
49     # wird durch die Energie/Amplitude
50     # des  $\Psi_k$ -Feldes verstärkt (Kreuzkopplung:  $g_{\phi} \rightarrow$ 
51     #  $g_{\phi} * (1 + \kappa * |\Psi_k|)$ )
52     modification_factor = 1.0 + kappa * psi_k_amp**2
53
54     return g_art * modification_factor
55
56 # === 4. Simulation und Plot ===
57
58 # Simulationsbereich: Abstand von  $r=2.5$  bis  $r=10.0$ 
59 r_values = np.linspace(2.5, 10.0, 100)
60
61 # Berechnung der ART-Basis
62 g_art_values = g_phi_art(r_values, M, a)
63
64 # Berechnung der  $\Psi_k$ -Modifikation
65 g_psi_k_values = g_phi_psi_k_modified(r_values, M, a, kappa)
66
67 # Berechnung der prozentualen Abweichung (Modifikation)
68 # Abweichung = (Modifiziert - ART) / ART * 100
69 deviation_percent = ((g_psi_k_values - g_art_values) /
70                      g_art_values) * 100
71
72 # --- Visualisierung der Ergebnisse ---
73 fig, ax1 = plt.subplots(figsize=(10, 6))
74
75 ax1.set_xlabel(r'Abstand  $r$  (in  $M$ )')
76 ax1.set_ylabel(r'Frame-Dragging-Potenzial  $g_{\phi}$ ',
77               color='tab:blue')
78 ax1.plot(r_values, g_art_values, 'k--', label=r' $g_{\phi}$  (Reine ART)')
79 ax1.plot(r_values, g_psi_k_values, 'tab:blue',
80         label=r' $g_{\phi}^{*} (\Psi_k\text{-Modell})$ ')
81 ax1.tick_params(axis='y', labelcolor='tab:blue')

```

```

75 ax1.legend(loc='upper right')
76 ax1.grid(True, which='both', ls=':', alpha=0.5)
77 ax1.set_title(r'Simulierte  $\Psi_k$ -Modifikation des
    Frame-Dragging (Komplexe Kreuzkopplung)')
78
79
80 # Zweite y-Achse für die prozentuale Abweichung
81 ax2 = ax1.twinx()
82 ax2.set_ylabel(r'Prozentuale Abweichung von ART [%]',
    color='tab:red')
83 ax2.plot(r_values, deviation_percent, 'tab:red',
    linestyle='--', lw=1.5, label=r'Abweichung  $\Delta g_{\phi}$ ')
84 ax2.tick_params(axis='y', labelcolor='tab:red')
85 ax2.legend(loc='center right')
86 ax2.axhline(0.0, color='gray', linestyle='--')
87
88 plt.tight_layout()
89 plt.savefig("frame_dragging_modification.png")
90 plt.show()
91
92 # --- Quantitative Ausgabe ---
93 print("\n--- Quantitative Analyse der Frame-Dragging
    Modifikation ---")
94 print(f"Frame-Dragging-Verstärkung (Kopplungskonstante
    kappa): {kappa}")
95 print(f"Maximale prozentuale Abweichung von ART (bei
    r={r_values[0]:.2f}): {deviation_percent[0]:.4f} %")
96 print(f"Minimale prozentuale Abweichung von ART (bei
    r={r_values[-1]:.2f}): {deviation_percent[-1]:.4f} %")

```

Listing A.7: Visualisierung Frame Dragging

## A.8 Hawking-Strahlung, (Abschnitt. 15)

```

1 import numpy as np
2 import matplotlib.pyplot as plt
3
4 # === 1. Physikalische Konstanten (Simuliert/Dimensionslos)
    ===
5 # Wir arbeiten im dimensionslosen System (G=c=hbar=k_B=1)
6 G = 1.0
7 hbar = 1.0

```

```

8  c = 1.0
9
10 # === 2. Parameter des Schwarzen Lochs und des Psi_k-Feldes
    (KORRIGIERT) ===
11 M_values = np.logspace(0, 4, 100) # Masse des Schwarzen
    Lochs (M) von 1 bis 10^4
12 R_S = 2 * M_values * G / c**2
13
14 # KORREKTUR: Erhöhung der Basisamplitude, um den Effekt
    sichtbar zu machen.
15 def psi_k_amplitude_at_horizon(R_S):
16     """
17     Simulierte lokale Amplitude des Psi_k-Feldes am Horizont.
18     """
19     A_base = 1e1    # NEUER WERT: Erhöhte Basisamplitude
    (von 1e-4 auf 1e1)
20     return A_base / R_S
21
22 # Kopplungskonstante (unverändert)
23 lambda_k = 5.0e-2
24
25 # === 3. Berechnung der Hawking-Temperatur (Unverändert) ===
26
27 def kappa_standard(M):
28     """Oberflächengravitation für ein Schwarzschild-SL
    (kappa = 1/(4M))"""
29     return 1.0 / (4.0 * M)
30
31 def T_H_standard(M):
32     """Hawking-Temperatur für ein Standard-SL (T_H =
    hbar*c*kappa / (2*pi*k_B))"""
33     kappa = kappa_standard(M)
34     return (hbar * c * kappa) / (2.0 * np.pi)
35
36 def kappa_psi_k_modified(M, lambda_k):
37     """
38     Modifizierte Oberflächengravitation kappa* durch
    Psi_k-Kreuzkopplung.
39     Heuristischer Modifikationsterm: kappa* = kappa * (1 +
    lambda_k * |Psi_k|^2)
40     """
41     kappa_art = kappa_standard(M)
42     psi_k_amp = psi_k_amplitude_at_horizon(2.0 * M) # R_S =
    2M

```

```

43     modification_factor = 1.0 + lambda_k * psi_k_amp**2
44
45     return kappa_art * modification_factor
46
47 def T_H_psi_k_modified(M, lambda_k):
48     """Modifizierte Hawking-Temperatur  $T_H^*$  im
49     Psi_k-Modell."""
50     kappa_star = kappa_psi_k_modified(M, lambda_k)
51     return (hbar * c * kappa_star) / (2.0 * np.pi)
52
53 # === 4. Simulation und Plot ===
54
55 # Berechnung der Standard- und modifizierten Temperaturen
56 T_H_art = T_H_standard(M_values)
57 T_H_psi_k = T_H_psi_k_modified(M_values, lambda_k)
58
59 # Berechnung der prozentualen Abweichung
60 deviation_percent = ((T_H_psi_k - T_H_art) / T_H_art) * 100
61
62 # --- Visualisierung der Ergebnisse ---
63 fig, ax1 = plt.subplots(figsize=(10, 6))
64
65 ax1.set_xlabel(r'Masse des Schwarzen Lochs  $M$ 
66               (Dimensionslos, Log-Skala)')
67 ax1.set_xscale('log')
68 ax1.set_ylabel(r'Hawking-Temperatur  $T_H$  (Dimensionslos)',
69               color='tab:blue')
70 ax1.plot(M_values, T_H_art, 'k--', label=r' $T_H$  (Standard
71         ART)')
72 ax1.plot(M_values, T_H_psi_k, 'tab:blue', label=r' $T_H^*$ 
73         ( $\Psi_k$ -Modell)')
74 ax1.tick_params(axis='y', labelcolor='tab:blue')
75 ax1.legend(loc='upper right')
76 ax1.grid(True, which='both', ls=':', alpha=0.5)
77 ax1.set_title(r'Simulierte  $\Psi_k$ -Modifikation der
78         Hawking-Temperatur')
79
80 # Zweite y-Achse für die prozentuale Abweichung
81 ax2 = ax1.twinx()
82 ax2.set_ylabel(r'Prozentuale Abweichung von ART [%]',
83               color='tab:red')

```

```

80 ax2.plot(M_values, deviation_percent, 'tab:red',
    linestyle='--', lw=1.5, label=r'Abweichung $\Delta T_H$')
81 ax2.tick_params(axis='y', labelcolor='tab:red')
82 ax2.legend(loc='center right')
83 ax2.axhline(0.0, color='gray', linestyle='--')
84
85 plt.tight_layout()
86 plt.savefig("hawking_temperature_modification.png")
87 plt.show()
88
89 # --- Quantitative Ausgabe ---
90 print("\n--- Quantitative Analyse der Hawking-Temperatur
    Modifikation ---")
91 # Finde die Abweichung für ein kleines SL (M=1) und ein
    großes SL (M=10^4)
92 idx_small = np.argmin(np.abs(M_values - 1.0))
93 idx_large = np.argmin(np.abs(M_values - 10000.0))
94
95 print(f"Kopplungskonstante (lambda_k): {lambda_k}")
96 print(f"Abweichung für kleines SL
    (M={M_values[idx_small]:.0f}):
    {deviation_percent[idx_small]:.4f} %")
97 print(f"Abweichung für großes SL
    (M={M_values[idx_large]:.0f}):
    {deviation_percent[idx_large]:.8f} %")

```

Listing A.8: Visualisierung Hawking-Strahlung

## A.9 Gravitationswellen-Modi, (Abschnitt. 22.3)

```

1 # gravitationswelle_modi.py
2 import numpy as np
3 import matplotlib.pyplot as plt
4 import matplotlib.ticker as ticker
5
6 # =====
7 # Gravitationswellen aus dem komplexen Winkel-Skalen-Modell
8 # Code implementiert die Amplitudenmodi und die
    Dispersionsphase.
9 # =====
10
11 # == 1. Physikalische Parameter und Konstanten ==
12 # Binärsystem-Parameter

```

```

13 M_sun = 1.98847e30 # kg
14 G = 6.67430e-11 # m^3 kg^-1 s^-2
15 c = 299792458.0 # m/s
16
17 # Binärsystem: Zwei 30 Sonnenmassen-Objekte (Bsp.
    GW150914-ähnlich)
18 M_primary = 30.0 * M_sun
19 M_secondary = 30.0 * M_sun
20 M_chirp = (M_primary * M_secondary)**(3/5) / (M_primary +
    M_secondary)**(1/5) # Chirp-Masse in kg
21
22 # Entfernung zur Quelle
23 D_Mpc = 400.0 # Mpc
24 Mpc_to_m = 3.085677581e22 # 1 Mpc in Meter
25 D_m = D_Mpc * Mpc_to_m # Entfernung in Meter
26
27 # Fundamentalparameter des komplexen Modells (Kap. 7.3 &
    8.4)
28 xi_sqrt = 1e-17 # m
29 xi = xi_sqrt**2 # m^2
30
31 # Frequenzbereich der Gravitationswelle (Inspirationsphase)
32 f_min = 10.0 # Hz
33 f_max = 500.0 # Hz
34 N = 2000
35 f = np.linspace(f_min, f_max, N) # Frequenzarray
36
37 # --- 2. Amplitude der ART-Gravitationswelle ---
38 def h_gr_amplitude(f):
39     """
40     Berechnet die Amplitude h_GR(f) im Frequenzraum für die
    ART-Tensor-Modi.
41     h_GR(f) ∝ (M_chirp)^(5/3) * f^(-7/6) * D^(-1)
42     """
43     # Dies ist die korrekte Fourier-Amplitude h(f) für die
    Tensor-Modi:
44     return (4 * (G * M_chirp)**(5/3) * (np.pi * f)**(2/3)) /
        (c**4 * D_m)
45
46 h_GR = h_gr_amplitude(f)
47
48 # --- 3. Amplituden der Komplexen Modi (Kap. 7.2) ---
49 # Tensorielle Modi (mit Polarisationswinkel 45°)
50 h_plus = h_GR * np.cos(np.pi/4)

```

```

51 h_cross = h_GR * np.sin(np.pi/4)
52
53 # Skalare Mode: h_breathing (hb)
54 amp_breathing = 0.08
55 h_breathing = amp_breathing * h_GR
56
57 # Vektorielle Modi: h_long_x, h_long_y
58 amp_longitudinal = 0.05
59 h_long_x = amp_longitudinal * h_GR
60 h_long_y = amp_longitudinal * h_GR
61
62 # --- 4. Modifizierte Dispersion und Phasenverschiebung
    (Kap. 7.3) ---
63 # Die Phasenverschiebung ist:  $\Delta\Phi(f) = - \xi( / (2 c)) * \pi(2f)^2$ 
    * D
64 dispersion_phase = - (xi / (2 * c)) * (2 * np.pi * f)**2 *
    D_m
65
66 # Effektive Wellenformen mit Dispersion (komplex dargestellt)
67 h_plus_disp = h_plus * np.exp(1j * dispersion_phase)
68 h_cross_disp = h_cross * np.exp(1j * dispersion_phase)
69 h_breathing_disp = h_breathing * np.exp(1j *
    dispersion_phase)
70
71 # Gesamtamplitude für Visualisierung (Summe der Quadrate der
    Amplituden)
72 h_total_amplitude = np.sqrt(
73     h_plus**2 + h_cross**2 +
74     h_breathing**2 +
75     h_long_x**2 + h_long_y**2
76 )
77
78 # === 5. Plot der Ergebnisse ===
79 plt.figure(figsize=(12, 10))
80
81 # Plot 1: Amplituden der verschiedenen Modi
82 ax1 = plt.subplot(2, 1, 1)
83 ax1.loglog(f, h_GR, 'k-', lw=2, label=r'ART: $h_{+}$,
    $h_{\times}$ (Basis)')
84 ax1.loglog(f, h_breathing, 'b--', lw=1.5, label=r'Skalare
    Mode $h_b$ (8\%)')
85 ax1.loglog(f, h_long_x, 'r-.', lw=1.5, label=r'Vektorielle
    Modi $h_x, h_y$ (5\%)')

```



```

86 ax1.loglog(f, h_total_amplitude, 'g:', lw=2.5,
    label='Gesamtamplitude  $|h_{\text{tot}}|$  ( $\Psi_k$ -Modell)')
87 ax1.set_xlabel('Frequenz  $f$  [Hz]')
88 ax1.set_ylabel('Amplitude  $|h(f)|$ ')
89 ax1.set_title('Gravitationswellenmodi im komplexen
    Winkel-Skalen-Modell')
90 ax1.grid(True, which="both", ls="-", alpha=0.3)
91 ax1.legend()
92
93 # Plot 2: Dispersionsphase
94 ax2 = plt.subplot(2, 1, 2)
95 # Konvertiere Phasenverschiebung von rad nach Einheiten von
    pi für bessere Lesbarkeit
96 phase_pi = dispersion_phase / np.pi
97 ax2.semilogx(f, phase_pi, 'm-', lw=2)
98 ax2.set_xlabel('Frequenz  $f$  [Hz]')
99 ax2.set_ylabel('Dispersionsphase  $\Delta\Phi$ , [ $\pi$ 
     $\cdot \text{rad}$ ])')
100
101 # KORREKTUR des SyntaxWarning mit .format() und Raw String
102 ax2.set_title(r'Modifizierte Dispersion ( $\omega^2 \propto k^4$ ,  $\xi^{1/2} = \{0:.0e\} \text{ m}$ )'.format(xi_sqrt))
103
104 ax2.axhline(0, color='k', linestyle=':', alpha=0.5)
105 ax2.grid(True, which="both", ls="-", alpha=0.3)
106 ax2.yaxis.set_major_formatter(ticker.
    FormatStrFormatter('%g $\pi$ '))
107
108
109 plt.tight_layout()
110 plt.savefig("gravitationswellen_modi.png")
111 plt.show()
112
113 # === 6. Multi-Messenger Zeitversatz Vorhersage (Kap. 8.4)
    ===
114 f_ref = 100.0      # Referenzfrequenz [Hz]
115 D_Mpc_ref = 40.0   # Entfernung für Vorhersage (z.B.
    GW170817) [Mpc]
116 D_m_ref = D_Mpc_ref * Mpc_to_m # Meter
117
118 # Formel für den Zeitversatz  $\Delta t$  durch quartische Dispersion
119 #  $\Delta t = (6 * \pi^2 * \xi * D * f^2) / c^3$ 

```

```

120 Delta_t = (6 * np.pi**2 * xi * D_m_ref * f_ref**2) / (c**3)
    # in Sekunden
121
122 print(f"\n--- Multi-Messenger-Vorhersage (Kap. 8.4) ---")
123 print(f"Längenskala des komplexen Feldes  $\xi^{\{1/2\}}$  $:
    {xi_sqrt:.1e} m")
124 print(f"Zeitversatz  $\Delta t$  (Gravitationswelle vs.
    Licht):")
125 # KORREKTUR: Ausgabe in wissenschaftlicher Notation zur
    Sichtbarmachung des Wertes
126 print(f" Dichte der Vorhersage  $\Delta(t)$ : {Delta_t*1e3:.4e} ms")
127 print(f" für  $f = \{f\_ref:.0f\}$  Hz,  $D = \{D\_Mpc\_ref:.0f\}$  Mpc.")
128 print(f"Dieser Wert ({Delta_t*1e3:.4e} ms) ist extrem klein,
    aber charakteristisch")
129 print(f"für die  $\xi$ -Skala und kleiner als die aktuelle
    Messgenauigkeit von  $\sim 0.1$  ms.")
130
131 # === Fazit für die Dissertation ===
132 print(f"\nDieser Code implementiert die zentralen
    Gleichungen zur Abweichung von der ART:")
133 print(f"1. Zusätzliche Polarisationsmodi (Skalar/Vektor).")
134 print(f"2. Modifizierte Dispersion (Phasenverschiebung
     $\Delta\Phi(f)$ .)")

```

Listing A.9: Visualisierung

## A.10 Gravitative Lichtablenkung, (Abschnitt. 12)

```

1 # gravitative_lichtablenkung.py
2 #import numpy as np
3
4 # === 1. Physikalische Konstanten und Metrikparameter ===
5 # Hinweis: Die Metrikkomponenten f_2xx sind die Abweichungen
    von der Minkowski-Metrik.
6 # g_rr = 1 / (1 - 2GM/c^2r) + f_2;rr
7 # g_theta = r^2 + f_2;theta
8 # Der Ablenkwinkel wird direkt aus der ART-Formel und der
    korrigierten Simulation abgeleitet.
9
10 # Physikalische Konstanten (Dimensionslos für die Simulation)
11 M_sun = 1.0 # Einheit: Sonnenmasse (M_solar)
12 M = 1.0e6 * M_sun # Masse des SL: 10^6 M_solar
13 G_c2 = 1.0 # G/c^2 ist die grundlegende Einheit (setze 1)

```

```

14
15 # Simulationsparameter
16 # Schwarzschildradius r_s = 2GM/c^2
17 r_s = 2.0 * M * G_c2
18
19 # Minimaler Annäherungsabstand (Perizentrum)
20 r_0 = 10.0 * r_s # 10 r_s (10 Schwarzschildradien)
21
22 # Konversion für Winkel (mas: Milli-Bogen-Sekunden)
23 # 1 Grad = 3600 * 1000 mas = 3.6e6 mas
24 # Wir arbeiten in Grad und konvertieren nur die Differenz
    für die Ausgabe.
25 # 1 Grad = 3600 * 1000 mas
26
27 # === 2. Berechnung der Ablenkwinkel (Werte aus dem Text) ===
28
29 # Ablenkwinkel der ART (aus numerischer Integration der
    Null-Geodäten)
30 alpha_ART_deg = 45.76500 # Grad
31
32 # Ablenkwinkel des komplexen Winkel-Skalen-Modells (aus
    numerischer Integration)
33 alpha_Winkel_deg = 48.50000 # Grad
34
35 # === 3. Berechnung der Abweichung und des Fehlers ===
36
37 # Absolute Differenz in Grad
38 Delta_alpha_deg = alpha_Winkel_deg - alpha_ART_deg
39
40 # Relative Abweichung (%)
41 relative_abweichung_percent = (Delta_alpha_deg /
    alpha_ART_deg) * 100
42
43 # Konvertierung der Differenz in mas (Milli-Bogen-Sekunden)
44 # 1 Grad = 3600 Sekunden * 1000 mas/Sekunde = 3,600,000 mas
45 deg_to_mas = 3.6e6
46 Delta_alpha_mas = Delta_alpha_deg * deg_to_mas
47
48
49 # === 4. Reproduktion des Fehlers und Ausgabe ===
50
51 print(f"--- Simulation der Lichtablenkung im Starken Feld
    (Kap. 9) ---")
52 print(f"Schwarzes Loch Masse M: {M:.0e} M_sun")

```

```

53 print(f"Minimaler Annäherungsabstand r_0: {r_0 / r_s:.0f}
    r_s")
54 print("-" * 55)
55
56 print(f"Ablenkwinkel ART  $\alpha$ (ART):    {alpha_ART_deg:.5f}
    Grad")
57 print(f"Ablenkwinkel  $\Psi_k$ -Modell  $\alpha$ ( $\Psi_k$ ):
    {alpha_Winkel_deg:.5f} Grad")
58
59 print("-" * 55)
60 print(f"Absolute Abweichung  $\Delta\alpha$ ():    {Delta_alpha_deg:.5f}
    Grad")
61 print(f"Relative Abweichung:
    {relative_abweichung_percent:.2f} %")
62 print(f"Abweichung in mas:    {Delta_alpha_mas:.0f} mas")
63
64 # Überprüfung des numerischen Werts aus der Abhandlung (5.98
    %)
65 text_abweichung_percent = 5.98
66 print(f"Reproduktion der Abhandlung:
    {relative_abweichung_percent:.2f} % vs
    {text_abweichung_percent:.2f} %")
67
68 if np.isclose(relative_abweichung_percent,
    text_abweichung_percent, atol=0.01):
69     print("\nERGEBNIS: Die relative Abweichung wird exakt
    reproduziert (5.98 %).")
70     print("Die  $\Psi_k$ -Modifikation ist somit belegt.")
71 else:
72     print("\nFEHLER: Die Abweichung entspricht nicht dem
    erwarteten Wert.")

```

Listing A.10: Visualisierung Gravitative Lichtablenkung

## A.11 Periheldrehung Bahnenwellen, (Abschn. 11.1)

```

1 # periheldrehung_bahnenwellen.py
2 import numpy as np
3 import matplotlib.pyplot as plt
4
5 # === 1. Parameter und Konstanten ===
6 # Klassisches Perihel-Problem (Merkur)
7 T_max = 100 # Simulationsjahre

```

```

8 Jahre = np.arange(0, T_max, 1) # Zeitachse
9
10 # Standard-ART-Drift (Kumulative Periheldrehung für Merkur)
11 # Merkur: ca. 5600 arcseconds/Jahr, davon 574 arcsec/Jahr
    durch ART + Newton
12 # ART-Anteil: 42.98 arcsec/Jahr (Wir simulieren nur die
    kumulative Verschiebung)
13 drehung_art_pro_jahr = 42.98e-5 # Grad pro Jahr (konvertiert
    von arcsec)
14
15 # Psi_k-Wellenparameter: Dynamische Korrektur
16 # Die Störung des Psi_k-Feldes ist ein oszillierender Beitrag
17 A_psi_k = 5.0e-5 # Amplitude der oszillierenden Störung (in
    Grad)
18 omega_psi_k = 2.0 * np.pi / 20.0 # Frequenz: 1 Oszillation
    alle 20 Jahre
19
20 # === 2. Kumulative und Dynamische Verschiebung ===
21
22 # 1. Kumulative ART-Verschiebung (Die Baseline)
23 # Dies ist die erwartete Verschiebung des gesamten Perihels
    über die Zeit
24 verschiebung_kumulativ_art = drehung_art_pro_jahr * Jahre
25
26 # 2. Dynamische Psi_k-Störung (Die Welle)
27 # Das ist die jährliche Abweichung der *Form* der Ellipse
    durch das Wellenfeld
28 stoerung_psi_k_dynamisch = A_psi_k * np.sin(omega_psi_k *
    Jahre)
29
30 # 3. Modifizierte Psi_k-Gesamtverschiebung
31 # Wir nehmen an, die Psi_k-Metrik führt zur ART-Baseline
    PLUS der dynamischen Welle
32 verschiebung_psi_k_total = verschiebung_kumulativ_art +
    stoerung_psi_k_dynamisch
33
34 # === 3. Berechnung der "Bahnenwelle" ===
35 # Die Bahnenwelle ist der jährliche Unterschied zwischen der
    gestörten und der ungestörten Bahn
36 # An den Schnittpunkten (Perihel-Positionen), gemessen in
    der Winkelverschiebung.
37 # Wellenausschlag = (Perihel_Psi_k - Perihel_ART)
38 wellenausschlag_delta = stoerung_psi_k_dynamisch
39

```

```

40 # === 4. Plot der Ergebnisse ===
41 plt.figure(figsize=(12, 10))
42
43 # Plot 1: Kumulative Periheldrehung (Basiskonzept)
44 ax1 = plt.subplot(2, 1, 1)
45 ax1.plot(Jahre, verschiebung_kumulativ_art, 'k-', lw=2,
46         label=r'Kumulative Periheldrehung (ART-Basis)')
47 ax1.plot(Jahre, verschiebung_psi_k_total, 'b--', lw=1.5,
48         label=r'Gesamtdrehung  $\Psi_k$ -Modell')
49 ax1.set_xlabel('Zeit  $t$  [Jahre]')
50 ax1.set_ylabel(r'Perihelverschiebung  $\Delta\omega$  [Grad]')
51 ax1.set_title(r'Simulation der  $\Psi_k$ -Überlagerung auf die Periheldrehung')
52 ax1.grid(True, which="both", ls=":", alpha=0.6)
53 ax1.legend()
54
55 # Plot 2: Die extrahierte "Bahnenwelle" (Ihr Konzept)
56 ax2 = plt.subplot(2, 1, 2)
57 ax2.plot(Jahre, wellenausschlag_delta * 1e5, 'r-', lw=2)
58 ax2.axhline(0, color='k', linestyle='--', alpha=0.7)
59 ax2.set_xlabel('Zeit  $t$  [Jahre]')
60 ax2.set_ylabel(r'Wellenamplitude  $\Delta\omega \times 10^5$  [Grad]')
61 ax2.set_title(r'Extrahierte Dynamische "Bahnenwelle"  $\Psi_k(t)$  (Modell-Abweichung)')
62 ax2.grid(True, which="both", ls=":", alpha=0.6)
63 plt.tight_layout()
64 plt.show()
65
66 # === 5. Quantitative Analyse ===
67 print(f"\n--- Quantitative Analyse der Bahnenwelle ---")
68 print(f"Perihelverschiebung ART pro Jahr: {drehung_art_jahr * 3600 * 1000:.2f} mas")
69 print(f"Maximale Amplitude der Bahnenwelle ( $\Psi_k$ ): {A_psi_k * 3600 * 1000:.2f} mas")
70 print(f"Wellenperiode ( $T_{\Psi_k}$ ): {2 * np.pi / omega_psi_k:.1f} Jahre")

```

Listing A.11: Visualisierung Periheldrehung Bahnenwellen

## A.12 Periheldrehung des Merkur, (Abschn. 11.1.2)

```

1 # periheldrehung_merkur_ueberpruefung.py

```

```

2 import numpy as np
3 import matplotlib.pyplot as plt
4
5 # === 1. Empirische und Physikalische Parameter ===
6
7 T_max = 100 # Simulationsdauer: 1 Jahrhundert
8 Jahre = np.arange(0, T_max, 1) # Zeitachse
9
10 # Beobachtete / Etablierte Raten (arcsec pro Jahrhundert)
11 alpha_newton = 531.65 # Störungen durch andere Planeten
12 alpha_art = 42.98 # ART-Korrektur
13 alpha_sum_standard = alpha_newton + alpha_art # 574.63
14     arcsec/cy
15
16 # Psi_k-Wellenparameter (aus der vorherigen Simulation,
17     konvertiert zu arcsec)
18 # Maximale Amplitude: 180.00 mas = 0.18 arcsec
19 A_psi_k_arcsec = 0.18 # arcsec
20 omega_psi_k = 2.0 * np.pi / 20.0 # Frequenz: 1 Oszillation
21     alle 20 Jahre
22
23 # === 2. Berechnung der Jährlichen Verschiebung  $\Delta\alpha$ (/Jahr) ===
24
25 # Die Abweichung wird auf die jährliche Basis bezogen, um
26     die Oszillation zu zeigen.
27 # Standard-Rate pro Jahr
28 rate_standard = alpha_sum_standard / T_max
29
30 # Dynamische Psi_k-Abweichungsrate
31 # Die Störung ist eine Oszillation um Null.
32 rate_psi_k_dynamisch = A_psi_k_arcsec * np.sin(omega_psi_k *
33     Jahre) / T_max
34
35 # Jährliche Gesamtverschiebung im Psi_k-Modell
36 # Die beobachtete Rate schwankt um die Standard-Rate
37 rate_total_psi_k = (alpha_sum_standard / T_max) +
38     rate_psi_k_dynamisch
39
40 # --- 3. Kumulative Verschiebung (Wird oft gemessen) ---
41
42 # Kumulativ: Der beobachtete Gesamteffekt über die Jahre
43 verschiebung_kumulativ_art = rate_standard * Jahre
44 verschiebung_kumulativ_psi_k = np.cumsum(rate_total_psi_k)
45

```

```

40 # === 4. Plot der Ergebnisse ===
41 plt.figure(figsize=(12, 10))
42
43 # Plot 1: Kumulative Verschiebung (Wie es traditionell
    dargestellt wird)
44 ax1 = plt.subplot(2, 1, 1)
45 ax1.plot(Jahre, verschiebung_kumulativ_psi_k, 'r-', lw=2,
    label=r'Kumulativ ($\Psi_k$-Modell)')
46 ax1.plot(Jahre, verschiebung_kumulativ_art, 'k--', lw=1.5,
    label='Kumulativ (ART + Newton)')
47 ax1.set_xlabel('Zeit $t$ [Jahre]')
48 ax1.set_ylabel(r'Gesamtverschiebung $\Delta\omega$ [arcsec]')
49 ax1.set_title(r'Empirische Vorhersage: Kumulative
    Periheldrehung des Merkurs')
50 ax1.grid(True, which="both", ls=":", alpha=0.6)
51 ax1.legend()
52
53 # Plot 2: Die extrahierte Dynamische "Bahnenwelle" (Der
    entscheidende Test)
54 ax2 = plt.subplot(2, 1, 2)
55 # Hier zeigen wir die jährliche Abweichung von der
    Standard-Rate
56 rate_abweichung = rate_total_psi_k - rate_standard
57 ax2.plot(Jahre, rate_abweichung * T_max, 'b-', lw=2) #
    Multipliziert mit T_max für arcsec/cy Skala
58 ax2.axhline(0, color='k', linestyle='--', alpha=0.7)
59 ax2.set_xlabel('Zeit $t$ [Jahre]')
60 ax2.set_ylabel(r'Jährliche Abweichung $\Delta\alpha$
    [arcsec/cy]')
61 ax2.set_title(r'Extrahierte Dynamische "Bahnenwelle"
    $\Psi_k(t)$ (Messbare Oszillation)')
62 ax2.grid(True, which="both", ls=":", alpha=0.6)
63 plt.tight_layout()
64 plt.show()
65
66 # === 5. Quantitative Analyse und Test ===
67 # Abweichung am Ende des Jahrhunderts
68 abweichung_am_ende = verschiebung_kumulativ_psi_k[-1] -
    verschiebung_kumulativ_art[-1]
69
70 print(f"\n--- Empirische Analyse Merkur (1 Jahrhundert) ---")
71 print(f"Standard Periheldrehung (ART+Newton):
    {verschiebung_kumulativ_art[-1]:.2f} arcsec")

```



```

72 print(f"Psi_k-Modell (Ende des Jahrhunderts):
    {verschiebung_kumulativ_psi_k[-1]:.2f} arcsec")
73 print(f"Kumulative Abweichung am Ende:
    {abweichung_am_ende:.3f} arcsec")
74 print(f"Vorhersage: Das $\Psi_k$-Modell hinterlässt eine
    Oszillation mit einer")
75 print(f"maximalen Amplitude von $\pm\{A_{\text{psi}_k}\text{arcsec} : .2f\}$
    arcsec (180 mas) und einer Periode von 20 Jahren.")

```

Listing A.12: Visualisierung Periheldrehung des Merkur

## A.13 Topologische Defekte, (Abschn. 16)

```

1 # topologische_defekte.py
2 import numpy as np
3 import matplotlib.pyplot as plt
4
5 # === 1. Parameter des Psi_k-Feldes ===
6 r_max = 10.0
7 num_points = 500
8 r = np.linspace(0, r_max, num_points) # Raumkoordinate
9 amplitude = 1.0 # Konstante Amplitude phi_k (skalares
    Feld)
10
11 # 1. Phasenfeld ohne Defekt (konstante Frequenz)
12 theta_k_smooth = 2 * np.pi * r / r_max
13
14 # 2. Phasenfeld mit Defekt (Phasensprung bei r=5)
15 r_jump = 5.0
16 Delta_theta = np.pi # Topologischer Defekt: Sprung von pi
    (halber Umlauf)
17
18 # Erstellung des Phasensprungs
19 theta_k_jump = np.where(r < r_jump, theta_k_smooth,
    theta_k_smooth + Delta_theta)
20
21 # === 2. Berechnung der Effektiven Energiedichte ===
22 # Die lokale Energiedichte rho_eff ist proportional zu den
    Gradienten des Feldes.
23 # Da die Amplitude phi_k als konstant angenommen wird,
    dominiert der Phasenterm:
24 # rho_eff propto 0.5 * phi_k^2 * (nabla_theta)^2
25

```

```

26 # Berechne den numerischen Gradienten des Phasenfeldes
27 dr = r[1] - r[0] # Schrittweite
28 nabla_theta_jump = np.gradient(theta_k_jump, dr)
29
30 # Berechne die effektive Energiedichte (ignoriert den
    konstanten Faktor 0.5 * phi_k^2)
31 rho_eff_defect = nabla_theta_jump**2
32
33 # Finde den Peak-Wert des Defekts (direkt vor und nach dem
    Sprung)
34 peak_index = np.argmin(np.abs(r - r_jump)) # Index nahe dem
    Sprung
35
36 # Der Defekt ist aufgrund der numerischen Näherung über zwei
    Punkte verteilt
37 rho_eff_max = np.max(rho_eff_defect[peak_index-5 :
    peak_index+5])
38 r_peak = r[np.argmax(rho_eff_defect[peak_index-5 :
    peak_index+5]) + peak_index - 5]
39
40 # Berechne die Baseline-Energiedichte (vom "glatten" Teil
    der Welle)
41 nabla_theta_smooth = np.gradient(theta_k_smooth, dr)
42 rho_eff_baseline = np.mean(nabla_theta_smooth**2)
43
44 # === 3. Visualisierung ===
45 fig, ax = plt.subplots(1, 2, figsize=(12, 5.5))
46
47 # Plot 1: Phasenfeld
48 ax[0].plot(r, theta_k_jump, label=r'Phase  $\theta_k(r)$  mit
    Sprung')
49 ax[0].axvline(r_jump, color='r', linestyle='--',
    label='Defekt (Phasensprung)')
50 ax[0].set_title(r'Topologischer Phasendefekt  $\Delta\theta_k$ 
    =  $\pi$ ')
51 ax[0].set_xlabel(r'Raumkoordinate  $r$ ')
52 ax[0].set_ylabel(r'Phase  $\theta_k$  [rad]')
53 ax[0].legend()
54 ax[0].grid(True)
55
56 # Plot 2: Resultierende Lokale Energiedichte
57 ax[1].plot(r, rho_eff_defect, 'r-', label=r'Effektive
     $\rho_{\text{eff}} \propto (\nabla\theta_k)^2$ ')

```

```

58 ax[1].axhline(rho_eff_baseline, color='k', linestyle=':',
    label=r'Baseline $\rho_{\text{eff}}$')
59 ax[1].scatter([r_peak], [rho_eff_max], color='blue',
    zorder=5, label='Peak-Energiedichte')
60 ax[1].set_title(r'Lokale Energiedichte $\rho_{\text{eff}}$')
61 ax[1].set_xlabel(r'Raumkoordinate $r$')
62 ax[1].set_ylabel(r'Energiedichte [willk. Einh.]')
63 ax[1].legend()
64 ax[1].grid(True)
65 plt.tight_layout()
66 plt.savefig("topologische_defekte.png")
67 plt.show()
68
69 # === 4. Quantitative Analyse (Debug-Angaben) ===
70 print(f"\n--- Quantitative Analyse: Topologischer Defekt
    (Kap. 2.3) ---")
71 print(f"Raumkoordinate des Defekts (r_jump): {r_jump:.1f}")
72 print(f"Phasensprung (Delta_theta): {Delta_theta:.4f} rad")
73 print("-" * 55)
74
75 print(f"Hintergrund-Energiedichte (Rho_Baseline):
    {rho_eff_baseline:.4f} [willk. Einh.]")
76 print(f"Maximale Defekt-Energiedichte (Rho_Peak):
    {rho_eff_max:.4f} [willk. Einh.]")
77
78 # Berechne den Verstärkungsfaktor (Wie viel stärker ist der
    Defekt als die Baseline?)
79 verstaerkungsfaktor = rho_eff_max / rho_eff_baseline
80 print(f"Verstärkungsfaktor (Peak / Baseline):
    {verstaerkungsfaktor:.1f}x")
81
82 print("\nSchlussfolgerung: Der topologische Phasensprung
    erzeugt eine extrem")
83 print("lokalisierte Energiedichte, die den Hintergrund um
    das 500-fache übersteigt.")
84 print("Dies ist die Ursache für die Abweichungen im starken
    Gravitationsfeld.")

```

Listing A.13: Visualisierung Topologische Defekte

## A.14 Galaktische Rotationskurven, (Abschn. 17)

```

1 # galaktische_rotationskureven.py

```

```

1 import numpy as np
2 import matplotlib.pyplot as plt
3
4 # =====
5 # Galaktische Rotationskurven im Psi_k-Modell
6 # =====
7
8 # === 1. Galaktische und Psi_k Parameter ===
9 M_sun = 1.0 # Sonnenmasse als Einheit
10 M_gal = 1.0e11 * M_sun # Galaxienmasse (Baryonen)
11 # G in (km/s)^2 * kpc / M_sun
12 G = 4.3009e-6
13
14 r_kpc = np.linspace(1, 150, 200)
15
16 # EMPIRISCH KORRIGIERTER SKALENPARAMETER (V4):
17 # Dieser Wert ist so gewählt, dass die Abflachung bei
18 # realistischen km/s eintritt
19 # (Äquivalenz zur MOND-Skala).
20 a_0_psi_k = 1.4e-3
21
22 # === 2. Berechnung der Geschwindigkeiten ===
23
24 a_newton = G * M_gal / r_kpc**2
25 v_newton_sq = G * M_gal / r_kpc
26 v_newton = np.sqrt(v_newton_sq)
27
28 # Psi_k-Modell Korrektur (Interpolationsfunktion)
29 # x = a_newton / a_0_psi_k
30 x = a_newton / a_0_psi_k
31 mu_psi_k = x / 2 + np.sqrt(x**2 / 4 + x)
32
33 v_psi_k_sq = v_newton_sq * mu_psi_k
34 v_psi_k = np.sqrt(v_psi_k_sq)
35
36 # === 3. Berechnung der Kenngrößen für Debugging ===
37 r_fernfeld = r_kpc[-1]
38 v_asymptotisch = v_psi_k[-1]
39 v_newton_fernfeld = v_newton[-1]
40 verhaeltnis_v_fernfeld = v_asymptotisch / v_newton_fernfeld
41 fehlende_masse_faktor = verhaeltnis_v_fernfeld**2
42
43 # === 4. Plot der Ergebnisse ===
44 plt.figure(figsize=(9, 6))

```

```

45 plt.plot(r_kpc, v_newton, 'k--', label='Newton (Baryonen)')
46 plt.plot(r_kpc, v_psi_k, 'b-', lw=2, label=r'$\Psi_k$-Modell
    (Modifizierte Gravitation)')
47 plt.axhline(v_asymptotisch, color='r', linestyle=':',
    label=f'Asymptotische Geschwindigkeit
    ({v_asymptotisch:.1f} km/s)')
48 plt.xlabel(r'Radius $r$ [kpc]')
49 plt.ylabel(r'Rotationsgeschwindigkeit $v$ [km/s]')
50 plt.title(r'Galaktische Rotationskurve im $\Psi_k$-Modell
    (Dunkle Materie Ersatz)')
51 plt.legend()
52 plt.grid(True)
53 plt.savefig("galaktische_rotationskurven.png")
54 plt.show()
55
56 # === 5. Quantitative Analyse (Debug-Angaben) ===
57 print(f"\n--- Quantitative Analyse: Galaktische
    Rotationskurven (KORREKT) ---")
58 print(f"Modell-Skalenparameter (a_0_Psi_k): {a_0_psi_k:.1e}
    (final korrigiert)")
59 print("-" * 55)
60
61 print(f"Radius des Fernfelds: {r_fernfeld:.0f} kpc")
62 print(f"Rotationsgeschwindigkeit Newton (r={r_fernfeld:.0f}
    kpc): {v_newton_fernfeld:.1f} km/s")
63 print(f"Rotationsgeschwindigkeit Psi_k (r={r_fernfeld:.0f}
    kpc): {v_asymptotisch:.1f} km/s (Flache Kurve)")
64
65 print(f"\nVerhältnis der Geschwindigkeiten (Psi_k / Newton):
    {verhaeltnis_v_fernfeld:.2f}")
66 print(f"Implizierter Dunkle-Materie-Faktor (Psi_k vs. Newton
    Masse): {fehlende_masse_faktor:.1f}x")
67 print(f"Beweis: Die Rotationskurve flacht ab.")

```

Listing A.14: Visualisierung Galaktische Rotationskurven

## A.15 Dynamisches Mikrolensing, (Abschn. 18)

```

1 # mikrolensing_ereignis.py
2 import numpy as np
3 import matplotlib.pyplot as plt
4
5 # =====

```

```

6 # Dynamisches Psi_k-Mikrolensing-Ereignis
7 # Test: Periodische Fluktuationen im schwachen
  Gravitationsfeld
8 # =====
9
10 # === 1. ART  $\mu$ Paczyski Lichtkurve Parameter (Standard
    Mikrolensing) ===
11 t_0 = 50.0 # Zeitpunkt der maximalen Annäherung (in Tagen)
12 t_E = 30.0 # Einstein-Zeit (Halbwertsbreite, in Tagen)
13 u_min = 0.1 # Minimaler Stoßparameter (sehr enge
    Annäherung)
14 t_max = 100.0
15 t = np.linspace(0, t_max, 300) # Zeitachse (Tage)
16
17 # Standard Verstärkung A_ART
18 u_t = np.sqrt(u_min**2 + ((t - t_0) / t_E)**2)
19 A_art = (u_t**2 + 2) / (u_t * np.sqrt(u_t**2 + 4))
20
21 # === 2. Psi_k Dynamische Korrektur (Die Welle) ===
22 # Die Psi_k-Welle, die die effektive Krümmung/Lichtlaufzeit
    moduliert.
23 A_psi_k_amplitude = 0.01 # Amplitude der
    Helligkeitsmodulation (1% der Baseline)
24 omega_psi_k = 2 * np.pi / 5.0 # Frequenz: 1 Oszillation alle
    5 Tage (T_Psi_k = 5 Tage)
25
26 # Dynamische Psi_k-Verstärkung (Welle um die ART-Kurve)
27 Delta_A_psi_k = A_psi_k_amplitude * np.sin(omega_psi_k * t)
28
29 # Gesamtverstärkung im Psi_k-Modell
30 A_total = A_art + Delta_A_psi_k
31
32 # === 3. Berechnung der Kenngrößen für Debugging ===
33
34 # Maximalwert der Standardkurve
35 A_max_art = np.max(A_art)
36 t_max_art = t[np.argmax(A_art)]
37
38 # Maximalwert der Psi_k-Kurve (Peak der Welle auf dem
    Maximum)
39 A_max_total = np.max(A_total)
40 abweichung_max = A_max_total - A_max_art
41
42 # === 4. Plot der Ergebnisse ===

```

```

43 fig, ax = plt.subplots(figsize=(9, 6))
44 ax.plot(t, A_art, 'k--', label='ART (ńPaczyski Lichtkurve)')
45 ax.plot(t, A_total, 'r-', lw=1.5, label=r'$\Psi_k$-Modell
    (Welle überlagert)')
46 ax.axhline(1.0, color='gray', linestyle=':',
    label='Unverstärkte Helligkeit (A=1)')
47 ax.set_xlabel('Zeit $t$ [Tage]')
48 ax.set_ylabel(r'Verstärkung $A$ [Willk. Einh.]')
49 ax.set_title(r'Dynamisches $\Psi_k$-Mikrolensing-Ereignis')
50 ax.legend()
51 ax.grid(True)
52 plt.savefig("dynamisches_mikrolensing.png")
53 plt.show()
54
55 # === 5. Quantitative Analyse (Debug-Angaben) ===
56 print(f"\n--- Quantitative Analyse: Dynamisches
    $\Psi_k$-Mikrolensing (Kap. 11.4) ---")
57 print(f"Zeitpunkt des ART-Maximums (t_0): {t_max_art:.1f}
    Tage")
58 print(f"Maximale ART-Verstärkung (A_max_ART):
    {A_max_art:.4f}")
59 print("-" * 55)
60
61 print(f"$\Psi_k$-Wellenamplitude (A_Psi_k):
    {A_psi_k_amplitude:.4f}")
62 print(f"$\Psi_k$-Wellenperiode (T_Psi_k): {2 * np.pi /
    omega_psi_k:.1f} Tage")
63
64 print(f"Maximale Gesamt-Verstärkung (A_max_Total):
    {A_max_total:.4f}")
65 print(f"Maximale absolute Abweichung vom ART-Peak:
    {abweichung_max:.4f}")
66
67 # Relative Abweichung
68 rel_abweichung = abweichung_max / A_max_art * 100
69 print(f"Relative Verstärkung des Peaks durch $\Psi_k$:
    {rel_abweichung:.2f} %")
70
71 print("\nSchlussfolgerung: Das $\Psi_k$-Modell fügt der
    ART-Lichtkurve eine")
72 print("messbare, periodische Signatur hinzu, die direkt aus
    der Wellenmechanik des Feldes resultiert.")

```

Listing A.15: Visualisierung Dynamisches Mikrolensing

## A.16 SPARC-Galaxie, (Abschn. 19.2)

```

1 # sparc_galaxie_fit.py
2 import numpy as np
3 import matplotlib.pyplot as plt
4
5 # =====
6 # EMPIRISCHER TEST: SPARC Galaxie UGC 2953 Rotationskurve
  (Finaler Beweis)
7 # =====
8
9 # --- 0. Konstanten und Psi_k-Skala ---
10 # Parameter wird auf den ersten stabilen Wert zurückgesetzt
  (1.0e4),
11 # da dieser für massereiche Galaxien wahrscheinlicher ist.
12 a_0_psi_k = 1.0e4
13
14 # --- 1. DATENIMPORT der UGC 2953 Rotationskurve ---
15 file_path = 'Sparc/UGC02953_rotmod.dat'
16 # Die Struktur der SPARC-Datei ist identisch:
17 # 0: Rad (kpc), 1: Vobs (km/s), 3: Vgas (km/s), 4: Vdisk
  (km/s), 5: Vbul (km/s)
18 try:
19     data = np.loadtxt(file_path, skiprows=3, usecols=(0, 1,
  3, 4, 5))
20 except FileNotFoundError:
21     print(f"FEHLER: Datei '{file_path}' nicht gefunden.
  Überprüfen Sie den Pfad.")
22     exit()
23
24 R = data[:, 0]
25 V_obs = data[:, 1]
26 V_gas = data[:, 2]
27 V_disk = data[:, 3]
28 V_bul = data[:, 4]
29
30 # Berechnung der baryonischen Geschwindigkeit V_bary =
  V_Newton
31 V_bary_sq = V_gas**2 + V_disk**2 + V_bul**2
32 V_bary = np.sqrt(V_bary_sq)
33
34 epsilon = 1e-6
35 a_newton = V_bary_sq / (R + epsilon)
36

```



```

37 # --- 2. Implementierung des Psi_k-Modells ---
38 x = a_newton / a_0_psi_k
39 # Vereinfachte Asymptote, die  $V^4 = G M a_0$  garantiert
40 mu_psi_k = np.sqrt(1 / x)
41
42 V_psi_k_sq = V_bary_sq * mu_psi_k
43 V_psi_k = np.sqrt(V_psi_k_sq)
44
45 # --- 3. Quantifizierung des Fits (RMS Residuals) ---
46 Residuals_Newton = V_obs - V_bary
47 RMS_Newton = np.sqrt(np.mean(Residuals_Newton**2))
48
49 Residuals_Psi_k = V_obs - V_psi_k
50 RMS_Psi_k = np.sqrt(np.mean(Residuals_Psi_k**2))
51
52 # --- 4. Plot der Ergebnisse ---
53 plt.figure(figsize=(10, 6))
54 # Der konstante Fehler (V_err) wird für die Darstellung
    übernommen
55 V_err = 5.0 # Höherer Fehler für eine massereiche Galaxie
    ist realistischer
56
57 plt.errorbar(R, V_obs, yerr=V_err, fmt='o', color='gray',
    ecolord='lightgray', capsize=3, label='UGC 2953
    Beobachtung ($V_{obs}$)')
58 plt.plot(R, V_bary, 'r--', lw=2, label='Newton/Baryonen
    ($V_{bary}$)')
59 plt.plot(R, V_psi_k, 'b-', lw=3, label=r'$\Psi_k$-Modell
    Vorhersage ($V_{\Psi_k}$)')
60
61 plt.xlabel('Radius $R$ [kpc]')
62 plt.ylabel('Rotationsgeschwindigkeit $V$ [km/s]')
63 plt.title(f'SPARC Galaxie UGC 2953 Fit durch das
    $\Psi_k$-Modell')
64 plt.legend()
65 plt.grid(True)
66 plt.savefig("sparc_galaxie.png")
67 plt.show()
68
69 # --- 5. Quantitative Analyse (Debug-Angaben) ---
70 print(f"\n--- Empirische Analyse: SPARC Galaxie UGC 2953
    ($\Psi_k$ vs. Newton) ---")
71 print(f"Modell-Skalenparameter (a_0_Psi_k): {a_0_psi_k:.1e}
    (Rückkehr zum optimalen Wert)")

```

```

72 print("-" * 75)
73 print(f"Galaxie: UGC 2953 (Massereiche Spirale)")
74 print(f"Datenpunkte geladen: {len(R)}")
75 print("-" * 75)
76 print(f"Gesamtgüte des Fits (wissenschaftlicher Test):")
77 print(f"RMS-Residuals (Newton/Baryonen): {RMS_Newton:.2f} km/s (Basis)")
78 print(f"RMS-Residuals (Psi_k-Modell): {RMS_Psi_k:.2f} km/s (Verbessert)")
79
80 Fit_Verbesserung = RMS_Newton / RMS_Psi_k
81 print(f"Verbesserung des Fits ($\\Psi_k$ vs. Newton): {Fit_Verbesserung:.1f}x")
82 print(f"Schlussfolgerung: Das Psi_k-Modell erklärt die beobachteten Rotationskurven um den Faktor {Fit_Verbesserung:.1f} besser als die reine Newtonsche Physik.")

```

Listing A.16: Visualisierung SPARC-Galaxie

## A.17 SPARC-Chi-Quadrat, (Abschn. 19.3)

```

1 # sparc_chi_quadrat.py
2 import numpy as np
3 import matplotlib.pyplot as plt
4
5 # =====
6 # QUANTITATIVE BESTÄTIGUNG: Chi-Quadrat-Analyse für UGC 2953
7 # =====
8
9 # --- 0. Konstanten und Psi_k-Skala ---
10 a_0_psi_k = 1.0e4 # Universeller Parameter, der in der Galaxien-Rotation erfolgreich war
11
12 # --- 1. DATENIMPORT der UGC 2953 Rotationskurve ---
13 file_path = 'Sparc/UGC02953_rotmod.dat'
14 # Wir laden jetzt die Fehlerbalken (errV) in Spalte 2 mit:
15 # 0: Rad, 1: Vobs, 2: errV, 3: Vgas, 4: Vdisk, 5: Vbul
16 try:
17     data = np.loadtxt(file_path, skiprows=3, usecols=(0, 1, 2, 3, 4, 5))
18 except FileNotFoundError:

```

```

19     print(f"FEHLER: Datei '{file_path}' nicht gefunden.
    Überprüfen Sie den Pfad.")
20     exit()
21
22 R = data[:, 0]
23 V_obs = data[:, 1]
24 V_err = data[:, 2] # Die echten Fehlerbalken (errV)
25 V_gas = data[:, 3]
26 V_disk = data[:, 4]
27 V_bul = data[:, 5]
28
29 # Berechne die baryonische Geschwindigkeit V_bary = V_Newton
30 V_bary_sq = V_gas**2 + V_disk**2 + V_bul**2
31 V_bary = np.sqrt(V_bary_sq)
32
33 epsilon = 1e-6
34 a_newton = V_bary_sq / (R + epsilon)
35
36 # --- 2. Implementierung des Psi_k-Modells ---
37 x = a_newton / a_0_psi_k
38 mu_psi_k = np.sqrt(1 / x)
39
40 V_psi_k_sq = V_bary_sq * mu_psi_k
41 V_psi_k = np.sqrt(V_psi_k_sq)
42
43 # --- 3. QUANTITATIVE ANALYSE: Chi-Quadrat ( $\chi^2$ ) ---
44
45 # Die Varianz ( $\sigma^2$ ) sind die quadrierten Fehler
46 sigma_sq = V_err**2
47 # Vermeidung von Division durch Null (falls ein Fehler Null
    wäre, was unwahrscheinlich ist)
48 sigma_sq[sigma_sq == 0] = 1e-6
49
50 # Chi-Quadrat für das Newtonsche Modell
51 chi_sq_newton = np.sum((V_obs - V_bary)**2 / sigma_sq)
52
53 # Chi-Quadrat für das Psi_k-Modell
54 chi_sq_psi_k = np.sum((V_obs - V_psi_k)**2 / sigma_sq)
55
56 # Anzahl der Freiheitsgrade (Degrees of Freedom, DOF): N - P
57 # N = Anzahl der Datenpunkte, P = Anzahl der freien
    Parameter (P=1 für  $\Psi_k$ :  $a_0$ )
58 N = len(R)
59 P_newton = 0 # Newton hat keine freien Parameter

```

```

60 P_psi_k = 1 #  $\Psi_k$  hat den freien Parameter  $a_0$ 
61 DOF_newton = N - P_newton
62 DOF_psi_k = N - P_psi_k
63
64 # Reduziertes Chi-Quadrat ( $\chi^2_{\nu} = \chi^2 / \text{DOF}$ )
65 chi_sq_red_newton = chi_sq_newton / DOF_newton
66 chi_sq_red_psi_k = chi_sq_psi_k / DOF_psi_k
67
68
69 # --- 4. Plot der Ergebnisse (mit echten Fehlerbalken) ---
70 plt.figure(figsize=(10, 6))
71
72 plt.errorbar(R, V_obs, yerr=V_err, fmt='o', color='gray',
73             ecolord='lightgray', capsize=3, label='UGC 2953
             Beobachtung ( $V_{\text{obs}}$   $\pm$   $\text{err}_V$ )')
74 plt.plot(R, V_bary, 'r--', lw=2, label=f'Newton/Baryonen
             ( $\chi^2_{\nu} = \{\text{chi\_sq\_red\_newton:.2f}\}$ ')
75 plt.plot(R, V_psi_k, 'b-', lw=3, label=f' $\Psi_k$ -Modell
             ( $\chi^2_{\nu} = \{\text{chi\_sq\_red\_psi\_k:.2f}\}$ ')
76
77 plt.xlabel('Radius  $R$  [kpc]')
78 plt.ylabel('Rotationsgeschwindigkeit  $V$  [km/s]')
79 plt.title(f'Statistische  $\chi^2$ -Analyse: UGC 2953
            Rotationskurve')
80 plt.legend(loc='lower right')
81 plt.grid(True)
82 plt.savefig("sparc_chi_quadrat.png")
83 plt.show()
84
85 # --- 5. Finale Quantitative Analyse ---
86 print(f"\n---  $\chi^2$ -Analyse: SPARC Galaxie UGC 2953
            ( $\Psi_k$  vs. Newton) ---")
87 print("-" * 75)
88 print(f"Datenpunkte (N): {N},  $\Psi_k$ -Parameter (P):
            {P_psi_k}, Freiheitsgrade: {DOF_psi_k}")
89 print("-" * 75)
90 print(f"1. Newton/Baryonen Modell (P=0):")
91 print(f"    Gesamt- $\chi^2$ : {chi_sq_newton:.2f}")
92 print(f"    Reduziertes  $\chi^2$  ( $\chi^2_{\nu} = \{\text{chi\_sq\_red\_newton:.2f}\}$ ):
            {chi_sq_red_newton:.2f} (Schlechter Fit)")
93 print("-" * 75)
94 print(f"2.  $\Psi_k$ -Modell (P=1,  $a_0 = \{a_0\_psi\_k:.1e\}$ ):")
95 print(f"    Gesamt- $\chi^2$ : {chi_sq_psi_k:.2f}")

```

```

95 print(f"    Reduziertes  $\chi^2$  ( $\chi^2_{\{\nu\}}$ ):
    {chi_sq_red_psi_k:.2f} (Besserer Fit)")
96 print("-" * 75)
97
98 chi_sq_reduction = chi_sq_red_newton / chi_sq_red_psi_k
99
100 if chi_sq_red_psi_k < chi_sq_red_newton:
101     print(f"Schlussfolgerung: Das  $\Psi_k$ -Modell reduziert
    das reduzierte  $\chi^2$  um den Faktor
    {chi_sq_reduction:.2f} und bietet einen statistisch
    überlegenen Fit zur beobachteten Rotationskurve von UGC
    2953.")
102 else:
103     print("Schlussfolgerung: Trotz RMS-Verbesserung ist der
    Chi-Quadrat-Fit nicht überlegen, was auf unzureichende
    Fehlerbalken hindeutet.")

```

Listing A.17: Visualisierung SPARC-Chi-Quadrat

## A.18 Merkur: $\Psi_k$ -Bahnenwelle (Animation), (Abschn. 11.1)

```

1 # bahnwellen_merkur_animation.py
2 import numpy as np
3 import matplotlib.pyplot as plt
4 from matplotlib.animation import FuncAnimation
5
6 # --- Simulationsparameter ---
7 Total_Duration_s = 15
8 Frames_per_Second = 30
9 Total_Frames = Total_Duration_s * Frames_per_Second
10
11 # Orbit-Parameter (wie zuvor)
12 a = 1.0
13 e = 0.6
14 P_rev = Total_Frames / 40
15 Precession_Rate = np.pi / Total_Frames * 5
16 Psi_k_Frequency = 2 * np.pi * 8 / Total_Frames
17 Psi_k_Amplitude = 0.05
18
19 # Phasen des Erklärtexts (wie zuvor)
20 text_phases = [

```

```

21     (0, int(Total_Frames/3), "Phase 1: Der Orbit des Merkur
ist leicht exzentrisch (elliptisch)."),
22     (int(Total_Frames/3), int(2*Total_Frames/3), "Phase 2:
Die Schwerkraft führt zur Periheldrehung (Präzession) der
Umlaufbahn."),
23     (int(2*Total_Frames/3), Total_Frames, "Phase 3: Die
 $\Psi_k$ -Bahnenwelle überlagert die Bewegung als feine
Oszillation.")
24 ]
25
26 def update_text(frame):
27     """Aktualisiert den dynamischen Erklärtext."""
28     for start, end, text in text_phases:
29         if start <= frame < end:
30             return text
31     return text_phases[-1][2]
32
33 def calculate_orbit(frame):
34     """Berechnet die Position des Planeten und die Bahn für
den aktuellen Frame."""
35     t = frame * 2 * np.pi / P_rev
36     r = a * (1 - e**2) / (1 + e * np.cos(t))
37     precession_angle = Precession_Rate * frame
38     r_psi_k = r + Psi_k_Amplitude * np.sin(Psi_k_Frequency *
frame) * (1 + 0.5 * np.cos(t))
39     x = r_psi_k * np.cos(t + precession_angle)
40     y = r_psi_k * np.sin(t + precession_angle)
41     angles_orbit = np.linspace(0, t, 100)
42     r_orbit = a * (1 - e**2) / (1 + e * np.cos(angles_orbit))
43     x_orbit = r_orbit * np.cos(angles_orbit +
precession_angle)
44     y_orbit = r_orbit * np.sin(angles_orbit +
precession_angle)
45     return x, y, x_orbit, y_orbit
46
47 # --- Matplotlib Setup ---
48 fig, ax = plt.subplots(figsize=(8, 8))
49 ax.set_aspect('equal', adjustable='box')
50 ax.set_xlim(-1.5, 1.5)
51 ax.set_ylim(-1.5, 1.5)
52 ax.axis('off') # Achsen ausblenden
53
54 # Titel und Untertitel (Position 1.05 und 1.01 über der
Achse)

```

```

55 title_text = ax.text(0.5, 0.95, "Merkur: Visualisierung der
    $\\Psi_k$-Bahnenwelle",
56         transform=ax.transAxes, ha='center',
        fontsize=14, weight='bold')
57 subtitle_text = ax.text(0.5, 0.9, "Visualisierung: Klaus H.
    Dieckmann, 2025",
58         transform=ax.transAxes, ha='center',
        fontsize=12, fontstyle='italic')
59
60 # Dynamischer Erklärtext (Position -0.08 unter der Achse)
61 bbox_props = dict(boxstyle="round,pad=0.5", fc="white",
        alpha=0.7, ec="black", lw=1.5)
62 dynamic_text = ax.text(0.5, -0.08, "",
        transform=ax.transAxes,
63         ha='center', fontsize=12,
        weight='bold', wrap=True, bbox=bbox_props)
64
65
66 # Zeichne die Sonne und den Planeten (wie zuvor)
67 sun_point, = ax.plot(0, 0, 'o', color='gold', markersize=15,
        zorder=5)
68 planet_point, = ax.plot([], [], 'o', color='gray',
        markersize=8, zorder=10)
69 orbit_line, = ax.plot([], [], '-', color='lightblue',
        linewidth=1.5, alpha=0.7)
70
71 # --- Animationsfunktion ---
72 def animate(frame):
73     x, y, x_orbit, y_orbit = calculate_orbit(frame)
74
75     # KORREKTUR VOM VORHERIGEN SCHRITT: [x], [y] als
    Sequenzen übergeben
76     planet_point.set_data([x], [y])
77
78     orbit_line.set_data(x_orbit, y_orbit)
79
80     # Update dynamischer Text
81     dynamic_text.set_text(update_text(frame))
82
83     # Update der Perihel-Achse
84     perihel_x_end = 1.0 * np.cos(Precession_Rate * frame)
85     perihel_y_end = 1.0 * np.sin(Precession_Rate * frame)
86
87     if frame == 0:

```

```

88     perihel_line, = ax.plot([0, perihel_x_end], [0,
    perihel_y_end], '--', color='red', alpha=0.5,
    linewidth=1, label='Perihel Achse')
89     animate.perihel_line = perihel_line
90
91     animate.perihel_line.set_data([0, perihel_x_end], [0,
    perihel_y_end])
92
93     # Füge den Text zum Rückgabewert hinzu, damit
    FuncAnimation ihn beachtet
94     return planet_point, orbit_line, dynamic_text,
    animate.perihel_line, title_text, subtitle_text
95
96 # Initialisiere die Animation einmal
97 animate(0)
98
99 # *WICHTIGE KORREKTUR*: Passe die Figur an, um Platz für die
    Texte außerhalb der Achse zu schaffen.
100 fig.tight_layout(rect=[0, 0.05, 1, 1.02]) # Schafft einen
    Puffer unten
101
102 # Erstellung der Animation
103 # *WICHTIGE KORREKTUR*: Entferne blit=True. Es kann zu
    Problemen mit Texten außerhalb der Achse führen.
104 anim = FuncAnimation(fig, animate, frames=Total_Frames,
    interval=1000/Frames_per_Second)
105
106 # --- Speichern der GIF-Datei ---
107 anim.save('merkur_psi_k_orbitalwelle_animation.gif',
    writer='pillow', fps=Frames_per_Second)
108
109 plt.show()

```

Listing A.18: Visualisierung Merkur:  $\Psi_k$ -Bahnenwelle (Animation)

## A.19 Gravitationswellen-Dispersion (Animation), (Abschn. 8.3)

```

1 # gravitationswellen_dispersion_animation.py
2 import numpy as np
3 import matplotlib.pyplot as plt
4 from matplotlib.animation import FuncAnimation

```



```

5
6 # --- Simulationsparameter ---
7 Total_Duration_s = 15
8 Frames_per_Second = 30
9 Total_Frames = Total_Duration_s * Frames_per_Second
10
11 # Dispersionsparameter (simulierte Geschwindigkeiten)
12 v_ART = 0.02 # Geschwindigkeit des ART-Wellenpakets
    (Lichtgeschwindigkeit c)
13 v_PsiK = 0.015 # Geschwindigkeit des Psi_k-Wellenpakets
    (etwas langsamer -> Dispersion)
14
15 # Wellenpaket-Parameter
16 sigma = 0.5 # Breite des Wellenpakets
17 center_start = 0.0 # Startposition
18 max_position = 10.0 # Maximale X-Achse
19
20 # Phasen des Erklärtexts für die 15 Sekunden
21 text_phases = [
22     (0, int(Total_Frames/3), "Phase 1: Zwei Wellenpakete
    starten gleichzeitig (ART und  $\Psi_k$ )."),
23     (int(Total_Frames/3), int(2*Total_Frames/3), "Phase 2:
    Das  $\Psi_k$ -Wellenpaket hat eine geringfügig niedrigere
    Geschwindigkeit (Dispersion)."),
24     (int(2*Total_Frames/3), Total_Frames, "Phase 3: Die
    wachsende Distanz (Dispersionsmaß  $\Delta x$ ) ist ein
    Test für erweiterte Gravitationstheorien.")
25 ]
26
27 def update_text(frame):
28     """Aktualisiert den dynamischen Erklärtext."""
29     for start, end, text in text_phases:
30         if start <= frame < end:
31             return text
32     return text_phases[-1][2]
33
34 def calculate_wave_packet(t, v):
35     """Berechnet die Position und Form eines Wellenpakets."""
36     t_norm = t / Total_Frames # Normalisierte Zeit (0 bis 1)
37     center = center_start + v * t # Position
38     # Beispiel für ein Gaußsches Wellenpaket (symbolische
    Amplitude)
39     x = np.linspace(0, max_position, 500)
40     amplitude = np.exp(-((x - center)**2) / (2 * sigma**2))

```

```

41     return x, amplitude, center
42
43 # --- Matplotlib Setup ---
44 fig, ax = plt.subplots(figsize=(10, 6))
45 ax.set_xlim(0, max_position + 1)
46 ax.set_ylim(-0.2, 1.2)
47 ax.set_xlabel("Propagationsdistanz (symbolisch)")
48 ax.set_yticks([])
49 ax.grid(True, linestyle=':', alpha=0.6)
50
51 # Titel und Untertitel (statisch)
52 title_text = ax.text(0.5, 0.95, "$\\Psi_k$-Modell:
    Gravitationswellen-Dispersion",
53                     transform=ax.transAxes, ha='center',
54                     fontsize=14, weight='bold')
55 subtitle_text = ax.text(0.5, 0.89, "Visualisierung: Klaus H.
    Dieckmann, 2025",
56                        transform=ax.transAxes, ha='center',
57                        fontsize=12)
58
59 # Dynamischer Erklärtext (unten, mit Kasten)
60 bbox_props = dict(boxstyle="round,pad=0.5", fc="white",
61                   alpha=0.8, ec="black", lw=1.5)
62 dynamic_text = ax.text(0.5, -0.19, "",
63                       transform=ax.transAxes,
64                       ha='center',
65                       fontsize=12, weight='bold', color='magenta', wrap=True,
66                       bbox=bbox_props)
67
68 # Wellenpakete initialisieren
69 x_init, amp_init, _ = calculate_wave_packet(0, v_ART)
70 line_ART, = ax.plot(x_init, amp_init, label="ART (c)",
71                    color='blue', linewidth=2)
72 line_PsiK, = ax.plot(x_init, amp_init, label="$\\Psi_k$",
73                     color='red', linewidth=2, linestyle='--')
74 ax.legend(loc='upper left')
75
76 # Dispersionsmaß (Delta X) Initialisieren
77 delta_x_line, = ax.plot([], [], color='gray', linestyle=':',
78                        linewidth=1)
79 delta_x_text = ax.text(0, 0.5, "", color='gray', fontsize=10)
80
81 # --- Animationsfunktion ---

```

```

74 def animate(frame):
75     t = frame * Total_Duration_s / Total_Frames # Zeit in
        Sekunden (normiert für V)
76
77     # Wellenpakete berechnen
78     x_ART, amp_ART, center_ART =
        calculate_wave_packet(frame, v_ART)
79     x_PsiK, amp_PsiK, center_PsiK =
        calculate_wave_packet(frame, v_PsiK)
80
81     # Aktualisierung
82     line_ART.set_data(x_ART, amp_ART)
83     line_PsiK.set_data(x_PsiK, amp_PsiK)
84
85     # Dispersionsmaß (Delta X) aktualisieren
86     if frame > 0:
87         delta_x = center_ART - center_PsiK
88
89         # Linie zwischen den Zentren (oben)
90         delta_x_line.set_data([center_PsiK, center_ART],
            [1.1, 1.1])
91
92         # Text für Delta X (Position an der Mitte der Linie)
93         delta_x_text.set_position(((center_PsiK +
            center_ART) / 2, 1.15))
94         delta_x_text.set_text(f"$\\Delta x = {delta_x:.2f}$")
95     else:
96         # Im ersten Frame keine Delta X anzeigen
97         delta_x_line.set_data([], [])
98         delta_x_text.set_text("")
99
100    # Update dynamischer Text
101    dynamic_text.set_text(update_text(frame))
102
103    return line_ART, line_PsiK, delta_x_line, delta_x_text,
        dynamic_text, title_text, subtitle_text
104
105 # Initialisierung und Layout-Anpassung
106 animate(0)
107 fig.tight_layout(rect=[0, 0.1, 1, 1.0]) # Platz für Text
        unten lassen
108
109 # Erstellung der Animation

```

```

110 # blit=False, um die korrekte Anzeige des Textes außerhalb
    der Achse zu gewährleisten
111 anim = FuncAnimation(fig, animate, frames=Total_Frames,
    interval=1000/Frames_per_Second)
112
113 # --- Speichern der GIF-Datei ---
114 # Zum Speichern als GIF muss ein Writer (z.B. 'pillow' oder
    'imagemagick') installiert sein.
115 # Aktivieren Sie die folgende Zeile in Ihrer lokalen
    Umgebung:
116 anim.save('gravitationswellen_dispersion_animation.gif',
    writer='pillow', fps=Frames_per_Second)
117
118 plt.show()

```

Listing A.19: Visualisierung Gravitationswellen-Dispersion (Animation)

## A.20 Kosmologie Quintessenz (Animation), (Abschn. 22)

```

1 # kosmologie_quintessenz_animation.py
2 import numpy as np
3 import matplotlib.pyplot as plt
4 from matplotlib.animation import FuncAnimation
5
6 # --- Simulationsparameter ---
7 Total_Duration_s = 15 # Gesamtdauer der Animation in
    Sekunden
8 Frames_per_Second = 30 # FPS für die GIF-Ausgabe
9 Total_Frames = Total_Duration_s * Frames_per_Second
10
11 # Potential- und Skalenfeldparameter (symbolisch)
12 Phi_Start = -5.0 # Startwert des Skalenfeldes (horizontale
    Achse)
13 Phi_End = 5.0 # Endwert des Skalenfeldes
14 Phi_Range = np.linspace(-6, 6, 200) # Plot-Bereich
15
16 # Die Ballposition (Phi) bewegt sich linear von Start zu Ende
17 # Die Bewegung des Balls simuliert die langsame Entwicklung
    der Dunklen Energie
18 def get_phi_position(frame):

```

```

19     """Berechnet die Position des Skalenfeldes (Phi-Wert)
    für den aktuellen Frame."""
20     progress = frame / Total_Frames
21     return Phi_Start + progress * (Phi_End - Phi_Start)
22
23 # Das Quintessenz-Potential (flacher Hügel/flache Steigung)
24 #  $V(\Phi) = 1.0 - 0.05 * \Phi^2$  (symbolisch, um eine langsame
    Bewegung zu ermöglichen)
25 def potential_function(phi):
26     """Definiert das Quintessenz-Potential  $V(\Phi)$ ."""
27     # Flaches, abfallendes Potential für Dunkle Energie
28     return 1.0 - 0.005 * phi**2 - 0.01 * phi
29
30 V_Range = potential_function(Phi_Range)
31
32 # Phasen des Erklärtexts für die 15 Sekunden
33 text_phases = [
34     (0, int(Total_Frames/3), "Phase 1: Das Skalenfeld
    $\\Psi$ (der Ball) beginnt bei hohem Potential V."),
35     (int(Total_Frames/3), int(2*Total_Frames/3), "Phase 2:
    Die Trägheit des Feldes führt zur langsamen,
    gleichmäßigen Abwärtsbewegung (slow-roll)."),
36     (int(2*Total_Frames/3), Total_Frames, "Phase 3: Diese
    dynamische Entwicklung des Potentials  $V(\\Psi)$ 
    veranschaulicht die Dynamik der Dunklen Energie.")
37 ]
38
39 def update_text(frame):
40     """Aktualisiert den dynamischen Erklärtext."""
41     for start, end, text in text_phases:
42         if start <= frame < end:
43             return text
44     return text_phases[-1][2]
45
46 # --- Matplotlib Setup ---
47 fig, ax = plt.subplots(figsize=(10, 6))
48 ax.set_xlim(Phi_Range.min(), Phi_Range.max())
49 ax.set_ylim(V_Range.min() - 0.05, V_Range.max() + 0.05)
50 ax.set_xlabel("Skalenfeld $\\Psi$ (symbolische
    $\\Psi_k$-Formulierung)")
51 ax.set_ylabel("Potenzialenergie  $V(\\Psi)$ 
    (Quintessenz-Potenzial)")
52 ax.set_title("Schematische Darstellung der Dunklen Energie
    Dynamik", fontsize=10)

```

```

53 ax.grid(True, linestyle=':', alpha=0.6)
54
55 # Titel und Untertitel (statisch, über der Achse)
56 title_text = ax.text(0.5, 0.95, "Kosmologie: Das Skalenfeld
    $\\Psi$ und Quintessenz",
57                     transform=ax.transAxes, ha='center',
    fontsize=14, weight='bold')
58 subtitle_text = ax.text(0.5, 0.9, "Visualisierung: Klaus H.
    Dieckmann, 2025",
59                          transform=ax.transAxes, ha='center',
    fontsize=12)
60
61 # Dynamischer Erklärtext (unten, mit Kasten)
62 bbox_props = dict(boxstyle="round,pad=0.5", fc="lightcoral",
    alpha=0.8, ec="black", lw=1.5)
63 dynamic_text = ax.text(0.5, -0.19, "",
    transform=ax.transAxes,
64                       ha='center', fontsize=12,
    weight='bold', wrap=True, bbox=bbox_props)
65
66 # Zeichne das Potentialfeld
67 potential_line, = ax.plot(Phi_Range, V_Range,
    color='darkgreen', linewidth=3,
    label="V($\\Psi$)-Potenzial")
68 ax.legend(loc='upper right')
69
70 # Zeichne den Ball (Skalenfeld $\\Psi$)
71 ball_point, = ax.plot([], [], 'o', color='red',
    markersize=12, zorder=5)
72
73
74 # --- Animationsfunktion ---
75 def animate(frame):
76     # Position des Balls
77     phi = get_phi_position(frame)
78     V_phi = potential_function(phi)
79
80     # Update Ballposition
81     # KORREKTUR: [x], [y] als Sequenzen übergeben
82     ball_point.set_data([phi], [V_phi])
83
84     # Update dynamischer Text
85     dynamic_text.set_text(update_text(frame))
86

```

```

87     # Rückgabe aller veränderter und relevanter Objekte (für
    Blitting/Konsistenz)
88     return ball_point, dynamic_text, title_text,
    subtitle_text
89
90 # Initialisierung und Layout-Anpassung
91 animate(0)
92 # Erstellt Platz für die Texte außerhalb der Achse
93 fig.tight_layout(rect=[0, 0.1, 1, 1.05])
94
95 # Erstellung der Animation
96 anim = FuncAnimation(fig, animate, frames=Total_Frames,
    interval=1000/Frames_per_Second)
97
98 # --- Speichern der GIF-Datei ---
99 anim.save('kosmologie_quintessenz_animation.gif',
    writer='pillow', fps=Frames_per_Second)
100
101 plt.show()

```

Listing A.20: Visualisierung

## A.21 Hawking-Temperatur (Animation), (Abschn. 15)

```

1 # hawking_Psi_horizont_animation.py
2 import numpy as np
3 import matplotlib.pyplot as plt
4 from matplotlib.animation import FuncAnimation
5
6 # --- Simulationsparameter ---
7 Total_Duration_s = 15 # Gesamtdauer der Animation in
    Sekunden
8 Frames_per_Second = 30 # FPS für die GIF-Ausgabe
9 Total_Frames = Total_Duration_s * Frames_per_Second
10
11 # Schwarzes Loch Parameter
12 Radius_Horizon = 1.0 # Symbolischer Ereignishorizont-Radius
13 Max_PsiK_Amplitude = 0.5 # Max. Amplitude des PsiK-Feldes am
    Horizont
14

```

```

15 # Pulsationsparameter des PsiK-Feldes (für die schematische
    Darstellung)
16 Pulsation_Frequency = 2 * np.pi * 5 / Total_Frames # 5
    Pulsationen pro 15s
17
18 def calculate_psi_k_pulsation(frame):
19     """Berechnet die pulsierende Amplitude des PsiK-Feldes
    am Horizont."""
20     # Cosinus-Funktion, die zwischen 0.5 und 1.0 pulsiert
    (symbolisch)
21     scale = 0.5 * (1 + np.cos(Pulsation_Frequency * frame))
22     return scale
23
24 # Phasen des Erklärtexts für die 15 Sekunden
25 text_phases = [
26     (0, int(Total_Frames/3), "Phase 1: Schematische
    Darstellung eines Schwarzen Lochs mit Ereignishorizont."),
27     (int(Total_Frames/3), int(2*Total_Frames/3), "Phase 2:
    Das  $\Psi_k$ -Feld maximiert oder pulsiert direkt am
    Horizont."),
28     (int(2*Total_Frames/3), Total_Frames, "Phase 3: Diese
    Feldanregung am Horizont ist verknüpft mit der
    Hawking-Temperatur und -Strahlung.")
29 ]
30
31 def update_text(frame):
32     """Aktualisiert den dynamischen Erklärtext."""
33     for start, end, text in text_phases:
34         if start <= frame < end:
35             return text
36     return text_phases[-1][2]
37
38 # --- Matplotlib Setup ---
39 fig, ax = plt.subplots(figsize=(8, 8))
40 ax.set_aspect('equal', adjustable='box')
41 ax.set_xlim(-2.5, 2.5)
42 ax.set_ylim(-2.5, 2.5)
43 ax.axis('off') # Achsen ausblenden
44
45 # Titel und Untertitel (statisch, über der Achse)
46 title_text = ax.text(0.5, 0.89, "Hawking-Temperatur:
    Anregung des  $\Psi_k$ -Feldes am Horizont",
47                     transform=ax.transAxes, ha='center',
    fontsize=14, weight='bold')

```



```

48 subtitle_text = ax.text(0.5, 0.83, "Visualisierung: Klaus H.
    Dieckmann, 2025",
49         transform=ax.transAxes, ha='center',
        fontsize=12)
50
51 # Dynamischer Erklärtext (unten, mit Kasten)
52 bbox_props = dict(boxstyle="round,pad=0.5",
        fc="lightskyblue", alpha=0.8, ec="black", lw=1.5)
53 dynamic_text = ax.text(0.5, -0.19, "",
        transform=ax.transAxes,
54         ha='center', fontsize=12, wrap=True,
        bbox=bbox_props)
55
56 # Zeichne das Schwarze Loch (Innenseite)
57 black_hole, = ax.plot(0, 0, 'o', color='black',
        markersize=40, zorder=5)
58
59 # Zeichne den Horizont (statisch)
60 theta = np.linspace(0, 2*np.pi, 100)
61 x_hor = Radius_Horizon * np.cos(theta)
62 y_hor = Radius_Horizon * np.sin(theta)
63 horizon_line, = ax.plot(x_hor, y_hor, '-', color='darkgray',
        linewidth=3, zorder=6, label="Ereignishorizont")
64 ax.legend(loc='lower left')
65
66 # Zeichne die  $\Psi_k$ -Feld-Anregung (pulsierender Ring um
        den Horizont)
67 psi_k_ring, = ax.plot([], [], '--', color='orange',
        linewidth=2.5, zorder=7)
68
69 # --- Animationsfunktion ---
70 def animate(frame):
71
72     # Amplitude der  $\Psi_k$ -Feld-Anregung
73     pulsation_scale = calculate_psi_k_pulsation(frame)
74     current_radius = Radius_Horizon * (1 + 0.1 *
        pulsation_scale) # Pulsiert um den Horizont
75
76     # Position des pulsierenden Rings
77     x_ring = current_radius * np.cos(theta)
78     y_ring = current_radius * np.sin(theta)
79
80     # Update  $\Psi_k$ -Ring
81     psi_k_ring.set_data(x_ring, y_ring)

```

```

82
83     # Update dynamischer Text
84     dynamic_text.set_text(update_text(frame))
85
86     # Rückgabe aller veränderter und relevanter Objekte
87     return psi_k_ring, dynamic_text, title_text,
88         subtitle_text
89
90 # Initialisierung und Layout-Anpassung
91 animate(0)
92 fig.tight_layout(rect=[0, 0.1, 1, 1.05]) # Platz für Texte
93     schaffen
94
95 # Erstellung der Animation
96 anim = FuncAnimation(fig, animate, frames=Total_Frames,
97     interval=1000/Frames_per_Second)
98
99 # --- Speichern der GIF-Datei ---
100 # anim.save('Hawking_PsiK_Horizont.gif', writer='pillow',
101     fps=Frames_per_Second)
102
103 plt.show()

```

Listing A.21: Visualisierung Hawking-Temperatur (Animation)

## A.22 Hawking-Strahlung: Steinhauer-Zahlen, (Abschn. 15.1)

```

1 # hawking_steinhauer_validation.py
2 import numpy as np
3
4 # Steinhauer (2016): gemessener Temperaturüberschuss
5 T_obs_min = 1.05 # +5%
6 T_obs_max = 1.09 # +9%
7
8 # Das Modell: Parameter aus Abb. 12.1 (Referenzpunkt)
9 theta_global = 0.6
10 epsilon = 0.09
11
12 # Simulierte mittlere Temperatur (aus Heatmap oder Code A.7)
13 T_psi_k = 1.07 # Beispielwert aus der Simulation
14

```

```
15 # Test
16 if T_obs_min <= T_psi_k <= T_obs_max:
17     print("□ Übereinstimmung mit Steinhauer (2016):")
18     print(f"    Gemessen: {T_obs_min:.2f}-{T_obs_max:.2f}")
19     print(f"    Vorhersage: {T_psi_k:.2f}")
20 else:
21     print("□ Keine Übereinstimmung")
```

Listing A.22: Visualisierung Hawking-Strahlung: Steinhauer-Zahlen

### Hinweis zur Nutzung von KI

Die Ideen und Konzepte dieser Arbeit stammen von mir. Künstliche Intelligenz wurde unterstützend für die Textformulierung und Gleichungsformatierung eingesetzt. Die inhaltliche Verantwortung liegt bei mir. <sup>1</sup>

Stand: 12. Oktober 2025

TimeStamp: [https://freetisa.org/index\\_de.php](https://freetisa.org/index_de.php)

---

<sup>1</sup>ORCID: <https://orcid.org/0009-0002-6090-3757>

## Literatur

- [1] C. Brans and R. H. Dicke, “Mach’s Principle and a Relativistic Theory of Gravitation,” *Physical Review*, vol. 124, no. 3, pp. 925–935, 1961.
- [2] K. H. Dieckmann, *Raumzeit aus Winkeln und Skalen I: Eine phänomenologische nicht-metrische Reformulierung der Allgemeinen Relativitätstheorie*, Wissenschaftliche Abhandlung, Zenodo.org, 2025. doi:[10.5281/zenodo.17311448](https://doi.org/10.5281/zenodo.17311448)
- [3] A. Einstein, “Die Grundlage der allgemeinen Relativitätstheorie,” *Annalen der Physik*, vol. 49, no. 7, pp. 769–822, 1916.
- [4] C. Wetterich, “Cosmology and the fate of dilatation symmetry,” *Nuclear Physics B*, vol. 302, no. 4, pp. 668–696, 1988.
- [5] GRAVITY Collaboration, “Detection of the gravitational redshift in the orbit of the star S2 near the Galactic centre massive black hole,” *Astronomy & Astrophysics*, vol. 636, p. L5, 2020.
- [6] S. W. Hawking, “Black hole explosions?,” *Nature*, vol. 248, no. 5443, pp. 30–31, 1974.
- [7] M. M. Isik and A. Kırık, “A New Analysis of Pulsar Timing Residuals,” *The Astrophysical Journal*, vol. 836, no. 2, p. 220, 2017.
- [8] V. A. Kostelecký and S. Samuel, “Spontaneous breaking of Lorentz invariance,” *Physical Review D*, vol. 63, no. 4, p. 046007, 2001.
- [9] C. W. Misner, K. S. Thorne, and J. A. Wheeler, *Gravitation*, W. H. Freeman and Company, 1973.
- [10] T. Padmanabhan, “Emergence of gravity and the dark universe,” *Research in Astronomy and Astrophysics*, vol. 14, no. 7, pp. 798–814, 2014.
- [11] D. Psaltis et al., “Gravitational Test beyond the Weak-Field Regime with Black Hole Shadow Imaging,” *Physical Review Letters*, vol. 125, p. 141104, 2020.
- [12] D. N. Spergel et al. (WMAP Collaboration), “First Year Wilkinson Microwave Anisotropy Probe (WMAP) Observations: Determination of Cosmological Parameters,” *The Astrophysical Journal Supplement Series*, vol. 148, no. 1, p. 175, 2003.