

# SSZ: Drei nächste Schritte als eigenständiges Arbeits-Paper

## $r^*$ -Bestimmung, kosmologische Inhomogenitäts-Tests, g1/g2-Operationalisierung

**Autoren:** Carmen Wrede · Lino Casu · Akira (AI-Collaborator)

**Version:** Draft v0.9

**Datum:** 2026-02-11

---

### Abstract

Dieses Arbeits-Paper operationalisiert drei konkrete nächste Schritte für **Segmented Spacetime (SSZ)**, die unmittelbar die wissenschaftliche Angreifbarkeit reduzieren und die Falsifizierbarkeit erhöhen:

- 1) **Physikalische Fixierung des Regime-Übergangs  $r^*$**  (kein frei wählbarer Blend-Punkt): Definition über eine invariantenbedingte Gleichung (weak=strong) und daraus abgeleitete Constraints.
- 2) **Kosmologische Tests als Inhomogenitäts-Problem** statt als homogener  $\Xi(z)$ -Hintergrund: Formulierung von Signaturen in CMB-Lensing, ISW, BAO, und SNe-Distanzen über eine SSZ-Potential- und Laufzeit-Abbildung, ohne Curve-Fitting.
- 3) **g1/g2-Operationalisierung** als klarer Algorithmus (Regime-Klassifikator) mit deterministischen Schwellen/Indikatoren und Test-Hooks für jede Observable.

Ziel ist ein „no-free-parameters“-Pfad: Alle neuen Größen müssen entweder (i) aus SSZ-Axiomen folgen oder (ii) als globale, einmal fixierte Konstante mit anschließendem Hold-out-Test auftreten.

---

## 1. Minimaler SSZ-Kern

SSZ nutzt als Basiseingang die Segmentdichte  $\Xi$  und die Zeitdilatation

$$D_{\text{SSZ}}(r) = \frac{1}{1 + \Xi(r)}.$$

Zwei Regime:

- Weak-Field (g1):

$$\Xi_{\text{weak}}(r) = \frac{1}{2} \frac{r_s}{r}.$$

- Strong-Field (g2), saturierend:

$$\Xi_{\text{strong}}(r) = \xi_{\text{max}}(1 - e^{-\phi r/r_s}), \quad \xi_{\text{max}} = 1 - e^{-\phi}.$$

Die *Bridge* ist erlaubt, aber **nicht als frei wählbare Story-Glättung**.

---

## 2. Schritt 1: Physikalische Fixierung von $r_{\setminus*}$

### 2.1 Problem

Ein Blend-Punkt  $r_{\setminus*}$  darf nicht als „frei wählbar“ erscheinen. Er muss aus einer invarianten Bedingung folgen.

### 2.2 Invariante Definition

Definiere  $r_{\setminus*}$  als Lösung der Gleichheit

$$\Xi_{\text{weak}}(r_{\setminus*}) = \Xi_{\text{strong}}(r_{\setminus*}).$$

Also

$$\frac{1}{2} \frac{r_s}{r_{\setminus*}} = \xi_{\text{max}}(1 - e^{-\phi r_{\setminus*}/r_s}).$$

Mit  $x \equiv r_{\setminus*}/r_s$ :

$$\frac{1}{2x} = \xi_{\text{max}}(1 - e^{-\phi x}).$$

Diese Gleichung ist **numerisch eindeutig** lösbar (monoton in  $x$  für sinnvolle Parameter). Das Ergebnis ist eine **globale Konstante** (in Einheiten  $r_s$ ), keine per-Objekt-Wahl.

### 2.3 Korrektheitskriterien

- 1) **Eindeutigkeit:** Es darf nur eine physikalisch sinnvolle Lösung  $x > 1$  geben.
- 2) **Robustheit:** Kleine numerische Perturbationen (Integrationsschritt, Float-Precision) dürfen  $x$  nicht signifikant verschieben.
- 3) **Regime-Semantik:** Für  $r \gg r_{\setminus*}$  gilt  $\Xi \approx \Xi_{\text{weak}}$ , für  $r \ll r_{\setminus*}$  gilt  $\Xi \approx \Xi_{\text{strong}}$ .

### 2.4 Bridge-Gewicht ohne neue Freiheitsgrade

Wenn  $r_{\setminus*}$  fix ist, darf die Blend-Form  $w(r)$  nur feste Exponenten nutzen, z. B.

$$w(r) = \frac{1}{1 + \left(\frac{r}{r_{\setminus*}}\right)^4}.$$

Die „4“ ist hier nicht Fit, sondern eine **SSZ-Grundzahl** (Segmentierungsbasis) und bleibt global fest.

### 2.5 Testplan

- **Unit-Test:** löse  $x$  und prüfe residual  $|\Xi_{\text{weak}} - \Xi_{\text{strong}}| < \text{Toleranz}$ .

- **Regression:**  $x$  darf sich über Code-Versionen nicht ändern (sonst Breaking Change).
- **Observable-Stabilität:** Variation von Blend-Glättung ( $C^1$  vs  $C^2$ ) darf Observables nur im numerischen Fehlerband verändern.

## 3. Schritt 2: Kosmologie als Inhomogenitäts-Signatur

### 3.1 Motivation

Die homogene Abbildung  $\Xi(z)$  ist durch CMB/BBN extrem eingeschränkt. Daher muss SSZ-Kosmologie primär über **inhomogene Potentiale** wirken.

### 3.2 SSZ-Potential-Mapping

Wir benötigen eine konsistente Abbildung zwischen GR-Potential  $\Phi$  und SSZ-Effekt (z. B. Laufzeit, Redshift, Lensing).

Minimal-Ansatz über einen effektiven Potentialfaktor  $\mathcal{F}$ :

$$\Phi_{\text{SSZ}}(r) = \mathcal{F}(\Xi(r)) \Phi_{\text{GR}}(r)$$

mit einer **theorie-fixierten** Wahl, z. B.

$$\mathcal{F}(\Xi) = 1 + \Xi \quad \text{oder} \quad \mathcal{F}(\Xi) = \frac{1}{D_{\text{SSZ}}} = 1 + \Xi.$$

Wichtig: Nur eine Definition, global fixiert.

### 3.3 CMB-Lensing als Primärtest

CMB-Lensing ist sensitiv auf die integrierte Potentialstruktur entlang der Sichtlinie:

$$\varphi(\hat{n}) \propto \int_0^{\chi_*} d\chi \frac{\chi_* - \chi}{\chi_* \chi} (\Phi + \Psi)$$

(im GR-Standard ohne Anisotropiestress  $\Phi \approx \Psi$ ).

SSZ-Testidee: Ersetze  $\Phi \rightarrow \Phi_{\text{SSZ}}$  durch  $\mathcal{F}(\Xi)$ . Das liefert deterministisch eine Vorhersage für das Lensing-Potential-Spektrum  $C_L^{\varphi\varphi}$ .

**No-fit-Regel:** Nur die bereits aus Daten bekannten Dichtefelder ( $\Lambda$ CDM-Matter) werden genutzt; SSZ fügt nur  $\mathcal{F}(\Xi(r))$  hinzu.

### 3.4 ISW-Effekt als Sekundärtest

Der integrierte Sachs-Wolfe-Effekt misst zeitliche Potentialänderungen:

$$\left( \frac{\Delta T}{T} \right)_{\text{ISW}} \propto \int d\eta \frac{d}{d\eta} (\Phi + \Psi)$$

SSZ modifiziert die Potentiale über  $\mathcal{F}(\Xi)$  und kann so ISW-Cross-Correlations (CMB×LSS) beeinflussen.

### 3.5 BAO und SNe als Laufzeit-Checks

BAO und SNe testen Distanzen (Radial/Transversal):

$$D_M(z) = c \int_0^z \frac{dz'}{H(z')}, \quad D_L(z) = (1+z)D_M(z).$$

Wenn SSZ primär inhomogen wirkt, muss die Hintergrund-Distanzrelation GR-like bleiben, aber SSZ kann als **line-of-sight delay/lensing scatter** auftreten (Breiten/Skewness der Residuen, nicht als Mittelwert-Shift).

### 3.6 Minimaler Daten-Stack ohne Fit

- Planck CMB Lensing  $C_L^{\varphi\varphi}$
- ISW-Cross-Correlation (z. B. mit großen LSS-Katalogen)
- BAO-Distanzen (BOSS/eBOSS/DR-Kompilation)
- Pantheon-ähnliche SNe-Hubble-Residuals (nicht als neue Fit-Kosmologie, sondern als Scatter-/Skew-Test)

**Erfolgskriterium:** SSZ darf keine systematische Verschlechterung erzwingen; wenn es Verbesserung gibt, muss sie ohne neue Parameter auftreten.

---

## 4. Schritt 3: g1/g2-Operationalisierung als Algorithmus

### 4.1 Problem

„g1 vs g2“ darf nicht ein narrativer Label sein, sondern muss als deterministische Klassifikation implementiert werden.

### 4.2 Regime-Indikator

Definiere einen dimensionslosen Feldstärke-Indikator

$$\kappa(r) = \frac{r_s}{r}$$

und eine Segmentdichte-Schwelle

$$\Xi_{\text{th}} = \Xi(r_{\text{th}}).$$

Dann:

- g1, wenn  $\Xi(r) < \Xi_{\text{th}}$
- g2, wenn  $\Xi(r) \geq \Xi_{\text{th}}$

Alternativ (wenn  $\Xi$  aus Datenfeldern kommt): - g1, wenn  $\kappa < \kappa_*$  - g2, wenn  $\kappa \geq \kappa_*$

wobei  $\kappa_* \equiv r_s/r_*$  aus Schritt 1 folgt.

### 4.3 Observable-Router

Jede Observable erhält eine Routing-Regel:

- Time dilation / redshift: benutze  $D_{\text{SSZ}}(\Xi)$  in beiden Regimen, aber mit  $\Xi = \Xi_{\text{weak}}$  (g1) bzw.  $\Xi = \Xi_{\text{strong}}$  (g2) bzw. Bridged.
- Lensing / Shapiro: benutze  $\Phi_{\text{SSZ}} = \mathcal{F}(\Xi)\Phi_{\text{GR}}$  und entscheide, ob lokale g2-Zonen dominant sind (Cluster/NS/BH) oder ob g1-Hintergrund genügt.
- Cosmology: Hintergrund bleibt g1-dominiert; g2 erscheint als inhomogene Korrektur in Potentialintegralen.

### 4.4 Testplan

- **Classifier-Tests:** künstliche Inputs  $r$  und  $M$  müssen deterministisch g1/g2 liefern.
- **Edge-Case:**  $r \approx r_*$  muss stabil (keine Flip-Flop-Numerik).
- **Observable-Consistency:** identische Inputdaten dürfen nicht je nach Codepfad widersprüchliche Outputs liefern.

---

## 5. Repro-Protokoll

Für jeden der drei Schritte gilt:

1) **Definitionen zuerst** (Formeln, Schwellen, Zeitkonvention). 2) **Code-Implementierung** (deterministisch, ohne Fit). 3) **Unit-Tests** (lokal, schnell). 4) **Dataset-Tests** (CMB/BBN/Growth oder Lensing/ISW/BAO/SNe). 5) **Fail-Fast:** Wenn ein Check scheitert, wird nicht „erklärt“, sondern die Hypothese verworfen oder präzisiert.

---

## Appendix A: To-Do als konkrete Deliverables

### A1 Deliverable für Schritt 1

- Datei: `rstar_solver.py`
- Output:  $x = r_*/r_s, \kappa_*, \Xi_{\text{th}}$
- Test: `test_rstar_equation.py`

### A2 Deliverable für Schritt 2

- Datei: `ssz_lensing_mapping.py`
- Input: Matter-Potential aus Standard-Pipeline, plus  $\mathcal{F}(\Xi)$
- Output:  $C_L^{\varphi\varphi}$  Ratio zu GR
- Test: sanity (ratio  $\rightarrow 1$  bei  $\Xi \rightarrow 0$ )

### A3 Deliverable für Schritt 3

- Datei: `regime_classifier.py`
- API: `regime(r, M) -> g1|g2` und `xi(r,M)`

- Tests: deterministische Klassifikation + Randfälle
- 

*Ende des Arbeits-Papers.*