

Strikte, menschenlesbare Beweise aus dem maschinenbewiesenen Pfad (Stand: v0.0604)

Vorbemerkung

Dieses Dokument formuliert die bereits im strikten Pfad maschinenbewiesenen Resultate in menschenlesbarer, aber eng an den Lean-Sätzen orientierter Form. Ich schreibe hier **nur** das, was im vorliegenden Stand formal gedeckt ist.

Präzisierung: In den sensiblen Gate-Bereichen spreche ich bewusst nur von dem, was die aktuelle Surface tatsächlich bündelt. Insbesondere ist die **Spectrum condition** hier ein Paket aus RepData plus SpectrumWitness, also **keine** bereits ausgearbeitete analytische Spektraltheorie; und die KMS/GNS-Schiene hängt formal an L **sowie** am aus Pillar A exportierten Wert $\beta = \text{betaFromA}$.

Wichtig ist insbesondere:

- **Isotonie** und **Locality** sind echte Gate-Beweise auf dem konkreten ToC-von-A-Brückennetz.
- Das **Split-Property** liegt aktuell als **expliziter Split-Witness / Marker** vor, nicht als tiefer analytischer von-Neumann-Satz.
- Die **Spectrum condition** ist im aktuellen Stand eine **gate-level witness construction**: ein Repräsentationsdatum plus ein expliziter Stützmengen-Nachweis im Vorwärtskegel.
- Die **basis-relative Additivität** ist **nicht** für das reiche Netz bewiesen, sondern für das **additive Observable-Netz**.
- Die **Nicht-Additivität des reichen Netzes** ist formal als echte Obstruktion bewiesen.

Gemeinsamer Kontext

Wir fixieren im strikten ToC-Pfad:

- einen Verzweigungsparameter $b : \text{Nat}$,
- eine Policy $p : \text{PillarA.LayerA.Policy } b$,
- wo erforderlich die Positivitätsannahme $[\text{Fact } (0 < b)]$.

Daraus werden im Repo die folgenden Objekte definiert:

- $\Omega := \text{ToC}.\Omega (b := b) (p := p)$ als endlicher Boundary-Träger,
- $L := \text{ToC}.L (b := b) (p := p)$ als aus Pillar A exportierte Randmatrix,
- $N_{\text{ToC}} := \text{ToC}.N_{\text{ToC}} (b := b) (p := p)$ als kanonisches reiches ToC-Brückennetz,
- $N_{\text{add_ToC}} := \text{ToC}.N_{\text{add_ToC}} (b := b) (p := p)$ als additives Observable-Netz,

- $B_add_ToC := ToC.B_add_ToC (b := b) (p := p)$ als transportierte Observable-Basis,
- $D_ToC := ToC.D_ToC (b := b) (p := p)$ als konkrete Disjunktheitsrelation.

Im Folgenden werden die Lean-Sätze jeweils in mathematischer Form paraphrasiert.

1. Isotonie

(1) Voraussetzungen

Gegeben sei das konkrete ToC-von-A-Brückennetz $N_ToC = ToC.N_ToC (b := b) (p := p)$.

Zu zeigen ist die Haag–Kastler-Isotonie im repo-spezifischen Sinn: für jede Regioneninklusion $a \leq c$ ist der Einschluss $N_ToC(a) \rightarrow N_ToC(c)$ injektiv.

Die maschinenbewiesene Fassung ist:

$ToC.isotony_ToC : Gates.HaagKastler.Isotony (N := N_ToC)$.

(2) Theorem

Theorem (Isotonie des konkreten ToC-Brückennetzes). Für alle Regionen a, c des Netzes N_ToC und jede Inklusion $hab : a \leq c$ ist die Abbildung $N_ToC.LAN.incl hab : N_ToC.LAN.Alg a \rightarrow N_ToC.LAN.Alg c$ injektiv.

(3) Beweis

Der formale Beweis reduziert direkt auf das zugrundeliegende Matrixnetz. Im Code wird $ToC.isotony_ToC$ durch Verweis auf `boundaryMatrixIncl_injective` bewiesen.

Der Kern ist folgender:

1. Man nehme $A, B \in N_ToC(a)$ mit $incl hab A = incl hab B$.
2. Da die lokalen Algebren als `RegionBlockMat`-Subtypen über konkreten Matrizen realisiert sind, genügt es, die Gleichheit der zugrundeliegenden Matrizen zu zeigen.
3. Dazu wird `congrArg RegionBlockMat.toMatrix` auf die Gleichung der eingeschlossenen Elemente angewendet.
4. Die Inklusion ändert die zugrundeliegende Matrix nicht; formal ist dies die Aussage, die über `boundaryMatrixLocalNet_incl_toMatrix` ausgerechnet wird.
5. Damit stimmen die Matrixrepräsentanten von A und B überein.
6. Mit dem Extensionalitätssatz `RegionBlockMat.ext` folgt $A = B$.

Also ist jeder lokale Einschluss injektiv, und damit gilt Isotonie.

2. Locality

(1) Voraussetzungen

Wir betrachten auf demselben Netz die konkrete Disjunktheitsrelation D_{ToC} , die als Trägerdisjunktheit der zugrundeliegenden finiten Regionen implementiert ist.

Die maschinenbewiesene Fassung ist:

`ToC.locality_ToC : Gates.HaagKastler.Locality (N := N_ToC) (D := D_ToC).`

(2) Theorem

Theorem (Locality / Mikrolokalität des konkreten ToC-Brückennetzes). Sind a und b disjunkte Regionen, dann kommutieren alle Elemente $x \in N_{ToC}(a)$ und $y \in N_{ToC}(b)$ nach Einbettung in die Algebra der Vereinigungsregion $a \sqcup b$.

(3) Beweis

Der Beweis ist konkret matrixtheoretisch und nicht nur formal-symbolisch. Er läuft im Code über `boundaryMatrixLocality` und verwendet drei Stufen.

Schritt 1: Disjunkt getragene Perturbationen multiplizieren zu Null.

Für lokal getragene Matrizen P auf a und Q auf b mit disjunkten Trägern wird gezeigt:

- $P * Q = \emptyset$,
- $Q * P = \emptyset$.

Der Grund ist die Summendarstellung der Matrixmultiplikation. In jedem Summanden ist mindestens einer der beiden Faktoren wegen der Trägerdisjunktheit Null.

Schritt 2: Lokale Blöcke zerfallen in „Skalar mal Eins + lokal getragene Perturbation“.

Jedes lokale Algebraelement wird als

$$A = \alpha \cdot 1 + P, B = \beta \cdot 1 + Q$$

mit lokal getragenen P, Q dargestellt.

Setzt man dies in AB und BA ein, dann:

- kommutieren die skalaren Teile automatisch,
- kommutieren die Mischterme wegen Zentralität von 1 ,
- verschwinden die reinen Perturbationsterme wegen Schritt 1.

Daher folgt bereits auf Matrixebene

$$\text{RegionBlockMat.toMatrix } A * \text{RegionBlockMat.toMatrix } B = \text{RegionBlockMat.toMatrix } B * \text{RegionBlockMat.toMatrix } A.$$

Schritt 3: Rücktransport in die lokale Algebra der Join-Region.

Die Elemente x und y werden über die Isotonieabbildungen in $N_ToC(a \sqcup b)$ eingebettet. Dort stimmt ihre zugrundeliegende Matrix mit den oben behandelten Matrixbildern überein. Also kommutieren sie auch als Elemente der Join-Algebra.

Damit ist die Haag–Kastler-Locality auf dem konkreten ToC-Netz bewiesen.

3. Split-Property-Marker

(1) Voraussetzungen

Wir betrachten zwei Regionen $a \leq b'$ im konkreten ToC-Brückennetz N_ToC .

Die relevanten maschinenbewiesenen Sätze sind:

- `ToC.splitWitness_ToC`,
- `ToC.splitWitness_ToC_isTypeI`,
- `ToC.splitProperty_ToC`.

Wichtig: Dies ist im aktuellen Stand ein **expliziter Split-Witness** auf Gate-Ebene. Es ist **kein** analytischer Typ-I-Satz über von-Neumann-Abschlüsse.

(2) Theorem

Theorem (Split-Property-Marker des ToC-Brückennetzes). Für jede Inklusion $a \leq b'$ besitzt das Netz N_ToC einen expliziten Split-Witness, also eine Zwischen-* \rightarrow -Algebra F mit Homomorphismen

$$N_ToC(a) \rightarrow F \rightarrow N_ToC(b'),$$

so dass ihre Komposition genau den Isotonieeinschluss $N_ToC(a) \rightarrow N_ToC(b')$ ergibt; außerdem ist F im aktuellen Gate-Sinn als Type-I-tragend markiert.

(3) Beweis

Der Beweis ist vollständig explizit konstruktiv.

Schritt 1: Definition des Zwischenobjekts.

Man definiert den Split-Träger als Bild des kleineren lokalen Algebras in der größeren:

$$\text{boundaryMatrixSplitCarrier } \text{hab} := \{ y \in N(b') \mid \exists x \in N(a), \text{incl } \text{hab } x = y \}.$$

Das Zwischenobjekt ist also nicht abstrakt postuliert, sondern konkret als Bildmenge der Inklusion konstruiert.

Schritt 2: Wahl eines Urbildes.

Für jedes y im Bildträger wird mittels `Classical.choose` ein Urbild `boundaryMatrixSplitOutLeft hab y` gewählt. Der zugehörige Spezifikationssatz zeigt:

$\text{incl hab (SplitOutLeft hab } y) = y.$

Schritt 3: Äquivalenz zwischen $N(a)$ und dem Bildträger.

Die Abbildungen

- $\text{SplitIn} : N(a) \rightarrow F,$
- $\text{SplitOutLeft} : F \rightarrow N(a)$

sind invers zueinander. Daher erhält man eine explizite Äquivalenz

$\text{boundaryMatrixSplitEquiv hab} : N(a) \simeq F.$

Schritt 4: Transport der *-Algebra-Struktur.

Über diese Äquivalenz wird die *-Algebra-Struktur von $N(a)$ auf F übertragen. Das liefert $\text{boundaryMatrixSplitSA hab}.$

Schritt 5: Expliziter *-Isomorphismus.

Aus der Äquivalenz wird ein *-Isomorphismus

$\text{boundaryMatrixSplitIsoLocal hab} : N(a) \simeq^* F$

konstruiert.

Schritt 6: Einbettung in die größere Algebra.

Die zweite Abbildung $F \rightarrow N(b')$ ist einfach die Inklusion des Bildträgers in die größere Faser. Der Satz $\text{boundaryMatrixSplitToRightHom_apply}$ zeigt, dass dieses Bild genau das zugrundeliegende Element von $N(b')$ ist.

Schritt 7: Kompositionsgleichung.

Die Komposition

$N(a) \rightarrow F \rightarrow N(b')$

ist definitionsgemäß genau der ursprüngliche Isotonieeinschluss.

Schritt 8: Type-I-Marker.

Da F explizit *-isomorph zu $N(a)$ konstruiert ist, wird im aktuellen Gate-Sinn isTypeI durch die Existenz dieses *-Isomorphismus bewohnt.

Damit liegt für jede Regioneninklusion ein expliziter Split-Witness vor. Also gilt $\text{splitProperty_ToC}.$

4. Spectrum condition

(1) Voraussetzungen

Hier ist besonders wichtig: Die aktuelle Spectrum condition ist eine **Witness-Konstruktion**, nicht die Auswertung eines analytischen Spektralmaßes.

Die maschinenbewiesene Fassung ist:

```
ToC.spectrumCondition_ToC : Gates.HaagKastler.SpectrumCondition (N := N_ToC)
(T := TransGroup_ToC).
```

Dazu werden im Code konstruiert:

- $\text{TransGroup_ToC} := \text{ZMod } d$, wobei $d = (T (b := b) (p := p)).d$,
- ein Repräsentationsträger $\text{RepCarrier_ToC} := \text{TransGroup_ToC} \times \text{N_ToC}(\text{topRegion})$,
- eine Shift-Wirkung shiftEquiv_ToC ,
- ein Impulspunkt $\text{momentumPoint_ToC} := (\text{energyScale_ToC}, \theta, \theta, \theta)$,
- eine Spektralstütze $\text{support} := \{\text{momentumPoint_ToC}\}$.

(2) Theorem

Theorem (Spectrum-condition-Gate auf dem strikten ToC-Pfad). Für das Netz N_ToC existieren auf der aktuellen Gate-Ebene

1. ein konkretes RepData -Paket über TransGroup_ToC , also ein Repräsentationsträger zusammen mit einer Familie von Äquivalenzen $U(g)$,
2. ein expliziter SpectrumWitness mit Stütze $\{\text{momentumPoint_ToC}\}$,

so dass diese Stütze vollständig im Vorwärtskegel liegt.

Wichtig ist: Formal wird damit **nicht** schon ein analytisches Spektrum einer unitären Darstellung konstruiert; bewiesen ist die aktuelle Witness-Form der Spectrum condition.

(3) Beweis

Der Beweis ist eine direkte Konstruktion.

Schritt 1: Übersetzungsgruppe.

Als Translationsgruppe wird $\text{ZMod } d$ gewählt. Das ist eine endliche, aus dem $A \rightarrow B$ -Handoff abgeleitete Trägergruppe.

Schritt 2: Repräsentationsträger.

Als Repräsentationsträger dient das Produkt aus Gruppenelement und Top-Algebraelement:

```
RepCarrier_ToC = TransGroup_ToC × N_ToC(topRegion).
```

Schritt 3: Kandidat für die Translationswirkung.

Ghostwriter: ChatGPT

Die Familie $U(g)$ wird durch die Shift-Abbildung

$$(x.1, x.2) \mapsto (x.1 + g, x.2)$$

mit Inverser

$$(x.1, x.2) \mapsto (x.1 - g, x.2)$$

definiert. Formal wird dies als Familie von Äquivalenzen `shiftEquiv_ToC g` in `RepData` eingetragen.

Wichtig ist erneut: Die aktuelle Struktur `RepData` verlangt auf Gate-Ebene nur H und $U : G \rightarrow H \simeq H$; die Gruppenaxiome einer Darstellung sind hier **nicht zusätzlich** als eigener Satz ausformuliert.

Schritt 4: Wahl des Spektralpunktes.

Als kanonischer Momentumspunkt wird

$$\text{momentumPoint_ToC} = (\text{energyScale_ToC}, \theta, \theta, \theta)$$

festgelegt, wobei

$$\text{energyScale_ToC} = \|L\|_F$$

die Frobeniusnorm der aus Pillar A exportierten Matrix ist.

Schritt 5: Positivität.

Aus der allgemeinen Eigenschaft der Frobeniusnorm folgt

$$\theta \leq \text{energyScale_ToC}.$$

Schritt 6: Vorwärtskegel-Nachweis.

Die Spektralstütze ist die Einermenge $\{\text{momentumPoint_ToC}\}$. Für ihr einziges Element sind beide Bedingungen des Vorwärtskegels erfüllt:

1. die Energiekomponente ist nichtnegativ,
2. die Lorentz-artige Ungleichung folgt hier trivial, weil die Raumkomponenten Null sind.

Im Lean-Beweis wird dies mittels

- `energyScale_nonneg_ToC`,
- `sq_nonneg`,
- `nlinarith`

ausgerechnet.

Also ist die gesamte Stützmenge im Vorwärtskegel enthalten. Damit ist die gate-level Spectrum condition bewiesen.

5. Basis-relative Additivität auf dem additiven Observable-Netz

(1) Voraussetzungen

Wir betrachten nicht das reiche Netz, sondern das additive Observable-Netz $N_{\text{add_ToC}}$ mit der transportierten Basis $B_{\text{add_ToC}}$.

Die maschinenbewiesene Fassung ist:

$\text{ToC.additivity_basis_ToC_add} : \text{AdditivityOnBasis } (N := N_{\text{add_ToC}}) (B_{\text{add_ToC}}).$

(2) Theorem

Theorem (Basis-relative Additivität des additiven Observable-Netzes). Für jede Region r wird jede Observable in $N_{\text{add_ToC}}(r)$ durch die Bilder der Basisregion-Algebren innerhalb von r erzeugt.

Anders gesagt: $N_{\text{add_ToC}}$ erfüllt Additivität **relativ zur ausgezeichneten L-abgeleiteten Basis**.

(3) Beweis

Der Beweis benutzt die Definition des additiven Netzes als **von Basisgeneratoren erzeugter Subnetzpfad**.

Sei also r eine Region und U eine $*$ -Unteralgebra von $N_{\text{add_ToC}}(r)$, welche alle Basisgeneratoren aus basisGenSet enthält. Es ist zu zeigen, dass dann jedes Element $A \in N_{\text{add_ToC}}(r)$ bereits in U liegt.

Schritt 1: Lift von U ins reiche Netz.

Man definiert eine $*$ -Unteralgebra V der entsprechenden reichen Faser, indem man die additive Zusatzstruktur vergisst:

$\text{boundaryMatrixAdditiveSubalgebraLift } r \ U.$

Ein Element des reichen Netzes liegt genau dann in V , wenn es als additives Element vorliegt und dort zu U gehört.

Schritt 2: Jedes reiche Basisgeneratorbild besitzt ein additives Urbild.

Der Hilfssatz $\text{rich_basisGenSet_has_additive_preimage}$ besagt: Jeder Basisgenerator des reichen Netzes in r wird durch ein additives Element mit demselben zugrundeliegenden Wert repräsentiert.

Da alle additiven Basisgeneratoren nach Voraussetzung in U liegen, folgt: alle entsprechenden reichen Basisgeneratoren liegen in V .

Schritt 3: Nutzung der Definition der additiven Faser.

Ein Element $A \in N_{\text{add_ToC}}(r)$ ist per Konstruktion gerade dadurch charakterisiert, dass sein zugrundeliegender reicher Wert durch diese Basisgeneratoren erzeugt ist.

Formal ist das genau der Inhalt der in A.2 gespeicherten Erzeugbarkeitsinformation.

Da v alle Basisgeneratoren enthält, folgt aus dieser Erzeugbarkeitsinformation, dass der zugrundeliegende reiche Wert von A in v liegt.

Schritt 4: Rückübersetzung in die additive Faser.

Aus der Definition von v erhält man nun, dass das zugehörige additive Element selbst in u liegt. Über Subtyp-Extensionalität identifiziert man dieses Element wieder mit A .

Folglich liegt jedes $A \in N_{\text{add_ToC}}(r)$ in jeder $*$ -Unteralgebra, die alle Basisgeneratoren enthält. Genau dies ist `AdditivityOnBasis`.

6. GNS / KMS / Heisenberg-Dynamik aus L und β_{FromA}

(1) Voraussetzungen

Wir arbeiten auf dem strikten ToC-Pfad mit `[Fact (0 < b)]` und der aus Pillar A exportierten symmetrischen Matrix L .

Relevante maschinenbewiesene Definitionen und Sätze sind insbesondere:

- `ToC.L_symm_ToC`,
- `ToC.matrixDynMinus_ToC`, `ToC.matrixDyn_ToC`,
- `boundaryDensityState_isKMS_minus_atBeta`,
- `ToC.matrixFullKMSMinus_ToC`, `ToC.matrixKMS_ToC`,
- `ToC.gnsDatum_top_ToC`,
- `ToC.state_top_ToC_from_cyclic`.

(2) Theorem

Theorem (Strikte Ableitung von Heisenberg-Dynamik, KMS-Zustand und GNS-Datum aus L und β_{FromA}). Aus der aus Pillar A exportierten symmetrischen Randmatrix L und dem ebenfalls aus A exportierten Wert $\beta = \text{betaFromA}$ werden auf dem strikten ToC-Pfad formal abgeleitet:

1. eine minus-orientierte volle Heisenberg-Dynamik,
2. ein dazu passender KMS-Zustand bei $\beta = \text{betaFromA}$,
3. ein konkretes GNS-Datum der Top-Algebra.

(3) Beweis

Teil A: Heisenberg-Dynamik aus L

Die minus-orientierte Dynamik ist definitionsgemäß

`matrixDynMinus_ToC = fullBoundaryDynMinus L L_symm_ToC.`

Die zugrundeliegende Zeitentwicklung ist also die Adjungierungswirkung

$$A \mapsto U_t^* A (U_t^*)^\dagger$$

mit

$$U_t^* = \exp(-i t H)$$

und $H = L$.

Dass dies wirklich eine *-Dynamik ist, ist im Modul `BoundaryMatrixFullDynamics` formal nachgewiesen:

- Erhaltung von $\emptyset, 1$, Addition, Skalarmultiplikation,
- Verträglichkeit mit Multiplikation,
- Verträglichkeit mit $*$,
- Zeitadditivität $\alpha_{\{s+t\}} = \alpha_s \circ \alpha_t$,
- Identität bei $t = \emptyset$.

Damit ist die Heisenberg-Dynamik formal aus L konstruiert.

Teil B: KMS-Zustand aus L

Aus L und dem von Pillar A exportierten Wert $\beta = \text{betaFromA}$ wird der Zustand

`boundaryDensityStateAtBeta L β`

gebildet; dies ist der normierte Gibbs-Zustand der Dichtematrixform

$$\rho_\beta \propto \exp(-\beta L).$$

Der zentrale analytische Satz ist

`boundaryDensityState_isKMS_minus_atBeta.`

Sein Beweis besteht formal aus drei Komponenten:

1. **Invarianz** des Zustands unter der minus-Dynamik,
2. **KMS-Kern** $F(z)$ als komplexe Fortsetzung,
3. Nachweis der **Randwerte** auf der reellen Achse und auf der verschobenen Randlinie $\text{Im}(z)=\beta$.

Dabei wird der KMS-Kern explizit als Spur-Ausdruck mit komplexen Exponentialen von L konstruiert. Die obere Randbedingung wird mit Hilfe von

- Exponentialadditionsformeln,
- Gibbs-Identitäten,
- Spur-Zyklizität,
- Verträglichkeit der Dynamik mit der Matrixmultiplikation

bewiesen.

Anschließend wird dieses Resultat in

`matrixFullKMSMinus_ToC`

als starkes C^* -KMS-Bündel und in

`matrixKMS_ToC`

als repo-weite `KMSState`-Oberfläche verpackt.

Insbesondere gilt formal:

- die inverse Temperatur ist exakt der aus `A` importierte Wert,
- die Dynamik ist exakt `matrixDyn_ToC`,
- der Zustand ist exakt der aus `L` gebildete Gibbs-Zustand.

Teil C: GNS-Datum aus `L`

Das Top-GNS-Datum ist

`gnsDatum_top_ToC = boundaryMatrixTopGNSDatum L β hL`.

Es ist konkret gegeben durch:

- Hilbertraumträger $H = \text{Mat } \Omega$,
- Darstellung $\pi(A)(X) = A \cdot X$, also Linksmultiplikation,
- zyklischer Vektor $\Omega_{\text{GNS}} = 1$.

Zyklizität. Für jedes $X : \text{Mat } \Omega$ wird als lokales Top-Element einfach $A := \text{matrixToTopRegion } X$ gewählt. Dann gilt

$$\pi(A)(1) = X.$$

Das ist genau der Satz `boundaryMatrixTopGNS_cyclic`.

Zustandsrekonstruktion. Der Top-Zustand ist durch das GNS-Skalarprodukt am zyklischen Vektor wiedergegeben:

$$\omega(A) = \langle\langle \Omega_{\text{GNS}}, \pi(A) \Omega_{\text{GNS}} \rangle\rangle.$$

Formal ist das der Satz `boundaryMatrixTop_state_from_cyclic`; auf dem `ToC`-Pfad erscheint er als

`state_top_ToC_from_cyclic`.

Somit sind Dynamik, KMS-Struktur und GNS-Realisierung sämtlich explizit aus `L` konstruiert und formal miteinander verknüpft.

7. Formale Additivitätsobstruktion des reichen Netzes

(1) Voraussetzungen

Wir betrachten nun ausdrücklich das **reiche** Netz `boundaryMatrixRichNet`, nicht das additive Observable-Netz.

Die entscheidenden maschinenbewiesenen Sätze sind:

- `pairOffDiagWitness_not_generated_from_singletons`,
- `boundaryMatrixRichNet_not_additive_at_pair`,
- `boundaryMatrixRichNet_not_additive_of_exists_ne`.

Voraussetzung ist lediglich, dass der Träger mindestens zwei verschiedene Punkte besitzt:

$$\exists i, j : \Omega, i \neq j.$$

(2) Theorem

Theorem (Formale Additivitätsobstruktion des reichen Netzes). Sobald der Träger Ω zwei verschiedene Punkte enthält, erfüllt das reiche Boundary-Matrix-Netz die unbeschränkte binäre Additivität **nicht**.

(3) Beweis

Der Beweis ist explizit und konstruiert einen konkreten Gegenzeugen.

Schritt 1: Zweipunktregion.

Für zwei verschiedene Punkte $i \neq j$ definiert man die Zweipunktregion

$$\text{pairRegion}(i, j) = \{i\} \cup \{j\}.$$

Schritt 2: Offdiagonaler Zeuge.

Betrachte die Matrixeinheit E_{ij} , also die Matrix mit Eintrag 1 an Position (i, j) und sonst \emptyset .

Diese Matrix ist in der Zweipunktregion lokalisiert und liefert daher ein lokales Algebraelement

$$\text{pairOffDiagWitness}(i, j) \in A(\text{pairRegion}(i, j)).$$

Wesentliche Eigenschaft:

- an der Stelle (i, j) steht 1,
- also ist dieses Element nicht diagonal auf dem Zweipunktblock.

Schritt 3: Null-Offdiagonal-Eigenschaft für aus Singletonen erzeugte Elemente.

Man definiert die Eigenschaft `pairOffDiagZero`:

Ein Element hat auf dem Zweipunktblock an den Stellen (i, j) und (j, i) den Wert \emptyset .

Dann wird formal gezeigt:

1. Bilder aus der linken Singleton-Algebra haben diese Eigenschaft.
2. Bilder aus der rechten Singleton-Algebra haben diese Eigenschaft.
3. Die Eigenschaft ist abgeschlossen unter
 - $\emptyset, 1$,
 - Addition,
 - Negation,
 - Stern,
 - Multiplikation.

Folglich besitzt **jedes** Element der von den beiden Singleton-Bildern erzeugten *-Unteralgebra diese Null-Offdiagonal-Eigenschaft.

Schritt 4: Der Zeuge liegt nicht in dieser erzeugten Unteralgebra.

Der Zeuge `pairOffDiagWitness(i, j)` verletzt die Null-Offdiagonal-Eigenschaft, weil sein (i, j) -Eintrag gleich 1 ist.

Also kann er nicht in der von den beiden Singleton-Bildern erzeugten *-Unteralgebra liegen.

Formal ist das der Satz

```
pairOffDiagWitness_not_generated_from_singletons.
```

Schritt 5: Widerspruch zur binären Additivität.

Unbeschränkte binäre Additivität würde gerade fordern, dass jedes Element der Algebra von `pairRegion(i, j)` aus den Bildern der beiden Singleton-Algebren erzeugt wird.

Da der explizite Zeuge dies nicht erfüllt, ist binäre Additivität falsch.

Daher gilt:

```
¬ Additivity (N := boundaryMatrixRichNet).
```

Das ist die formale Additivitätsobstruktion des reichen Netzes.

Schlussbemerkung

Der formale Gesamtbefund des aktuellen strikten Pfades lautet damit präzise:

- Das **reiche ToC-Brückennetz** trägt bereits echte Isotonie, Locality, Split-Witness-Marker und eine gate-level Spectrum condition.

- Die **volle additive Observable-Struktur** ist **nicht** auf dem reichen Netz wahr, sondern auf dem eigens konstruierten **additiven Observable-Netz**.
- Die **GNS/KMS/Heisenberg-Schiene** ist strikt aus der exportierten Matrix L aufgebaut.
- Die **Nicht-Additivität des reichen Netzes** ist kein bloßer Verdacht, sondern formal durch einen expliziten offdiagonalen Gegenzeugen bewiesen.

Gerade diese Trennung zwischen reichem/completion-artigem Netz und additivem Observable-Netz ist im Stand v0.0604 keine lose Interpretation mehr, sondern Teil des tatsächlich maschinenbewiesenen Pfades.