

The Fusion Gauge of the Stabilizer Reconstruction Formula

2026-09-26

Verdict: local-correction (resolved).

1 The source statement

Let a group G act transitively on a set X of blocks, fix $x_0 \in X$ with stabilizer H , and choose representatives k_x with $k_x x_0 = x$ and $k_{x_0} = e$. Let ω be a three-cocycle on G . Ref. [GRLM23, source lines 740–747] shows that the restriction $\omega|_H$ is a coboundary $d\beta$, and that in the fusion gauge fixed by β the basepoint restriction $\psi = L/\beta$ is a two-cocycle on H . Equation (20) of the same reference, at source lines 752–759, then reconstructs every component of the L -symbol:

$$L_{g_1, g_2}^x = \frac{\omega(g_1 g_2 k_x, h_2^{-1}, h_1^{-1}) \psi(h_1, h_2)}{\omega(g_1, g_2 k_x, h_2^{-1}) \omega(g_1, g_2, k_x)}, \quad (1)$$

where $h_2 = k_{g_2 x}^{-1} g_2 k_x$ and $h_1 = k_{g_1 g_2 x}^{-1} g_1 k_{g_2 x}$. Line 761 says that “given ω and β by fixing some gauge” the solutions of the compatibility equation

$$L_{g, hk}^x L_{h, k}^x = \omega(g, h, k) L_{g, h}^{kx} L_{g h, k}^x \quad (2)$$

form a torsor under $H^2(H, \mathbb{C}^\times)$.

2 The gap

Formula (1) with a two-cocycle ψ satisfies (2) when $\omega(a, b, c) = 1$ for all $a, b, c \in H$. It does not do so for a general three-cocycle with $\omega|_H = d\beta$. At $x = x_0$ and $a, b \in H$ the formula gives $L_{a, b}^{x_0} = F(a, b)\psi(a, b)$ with

$$F(a, b) = \frac{\omega(ab, b^{-1}, a^{-1})}{\omega(a, b, b^{-1}) \omega(a, b, e)}, \quad (3)$$

while (2) restricted to x_0 and H forces $\omega|_H = d(L^{x_0}|_H) = dF$ with F as in (3). This can fail.

Example. Let $G = H = \mathbb{Z}_3 = \{e, s, s^2\}$ act on a single block, so $k = e$ and $h_1 = g_1, h_2 = g_2$. Let $\beta(s, s) = -1$ and $\beta = 1$ at every other pair, and put $\omega = d\beta$, with $(d\beta)(g, h, k) = \beta(g, hk)\beta(h, k)/(\beta(g, h)\beta(gh, k))$. Then $\omega(s, s, s^2) = -1$, while $F(s, s) = F(s^2, s^2) = -1$ and $F = 1$ at the other pairs, so $dF = 1$ identically. Hence for every two-cocycle ψ the formula violates (2) at $(g, h, k) = (s, s, s^2)$. Numerical checks on $A_4 \supset V_4$ and $S_3 \supset A_3$, with ω twisted by a random normalized coboundary, show violations of order one in the same way.

3 The local fix

Before applying (1), change the fusion gauge so that ω is one on $H \times H \times H$. Such a gauge always exists: if $\omega|_H = d\beta_0$ with β_0 on H , extend β_0^{-1} to G by one outside $H \times H$ and apply the resulting fusion gauge. In that gauge $F = 1$ on H , the basepoint restriction of (1) is ψ itself, and a direct four-term three-cocycle computation shows that (1) satisfies (2); no normalization of ω is needed. Transporting back by the inverse fusion gauge gives a solution for the original ω . Equivalently, one may keep ω and replace ψ in (1) by $\psi\beta F^{-1}$, which matches the source's reading $L|_H = \psi\beta$.

The torsor statement of line 761 is correct for classes under action-tensor gauges with the fusion gauge held fixed, as the source says. If fusion gauges preserving ω are also allowed, a two-cocycle β on G acts, and the classes collapse to the quotient of $H^2(H, \mathbb{C}^\times)$ by the restriction of $H^2(G, \mathbb{C}^\times)$; this remark is not formalized.¹

Verdict: resolved local fix. Formula (1) is correct in the fusion gauge in which ω is trivial on the stabilizer, and that gauge always exists. The classification and the torsor statement hold as stated for action-tensor gauge classes.

References

- [GRLM23] José Garre-Rubio, Laurens Lootens, and András Molnár. Classifying phases protected by matrix product operator symmetries using matrix product states. *Quantum*, 7:927, 2023. Source coordinates in this repository use arXiv:2203.12563v3.

¹Formal declarations: `TNLean.Algebra.StabilizerRepresentatives.reconstructedLSymbol`, `TNLean.Algebra.StabilizerRepresentatives.reconstructedLSymbol_isCompatible`, `TNLean.Algebra.StabilizerRepresentatives.exists_actionGauge_eq_reconstructedLSymbol`, `TNLean.Algebra.StabilizerRepresentatives.h2EquivSolutionClasses` and `TNLean.Algebra.ScalarThreeCochain.exists_fusionGauge_eq_one_of_isTrivialGaugeClass_comap` in `TNLean/Algebra/StabilizerCocycleReconstruction`.