

Road Map. • Implement and verify scatter-level stitching.

• Guess/deduce glow level.

Deriving Gassner. \mathcal{L}^{2Dw} is $\mathbb{Q}[[b_i]]\langle a_{ij} \rangle$ modulo locality, $[a_{ij}, a_{ik}] = 0$, $[a_{ik}, a_{jk}] = -[a_{ij}, a_{jk}] = b_j a_{ik} - b_i a_{jk}$, and $(\text{mod } \langle a_{ii} \rangle) [a_{ij}, a_{ji}] = b_i a_{ji} - b_j a_{ij}$. Acts on $\mathbf{V} = \mathbb{Q}[[b_i]]\langle x_i = a_{i\infty} \rangle$ by $[a_{ij}, x_i] = 0$, $[a_{ij}, x_j] = b_i x_j - b_j x_i$. Hence $e^{\text{ad} a_{ij}} x_i = x_i$, $e^{\text{ad} a_{ij}} x_j = e^{b_i} x_j + \frac{b_j}{b_i} (1 - e^{b_i}) x_i$. Renaming $\bar{x}_i = x_i/b_i$, $\bar{t}_i = e^{b_i}$, get

$$[e^{\text{ad} a_{ij}}]_{\bar{x}_i, \bar{x}_j} = \begin{pmatrix} 1 & 1 - t_i \\ 0 & t_i \end{pmatrix}.$$

The \mathcal{L}^{2Dw} Adjoint representation. $e^{\text{ad} a_{ij}}$ acts by

$$a_{kl} \mapsto a_{kl}, \quad a_{ik} \mapsto a_{ik}, \quad a_{kj} \mapsto e^{-b_i} a_{kj} + \frac{b_k}{b_i} (1 - e^{-b_i}) a_{ij},$$

$$a_{ki} \mapsto a_{ki} + (1 - e^{-b_i}) a_{kj} + b_k \frac{e^{-b_i} - 1}{b_i} a_{ij},$$

$$a_{jk} \mapsto e^{b_i} a_{jk} + \frac{b_j}{b_i} (1 - e^{b_i}) a_{ik}, \quad a_{ji} \mapsto e^{b_i} a_{ji} + \frac{b_j}{b_i} (1 - e^{b_i}) a_{ij}.$$

Implementation/verification: pensieve://2015-04/nb/ZeroCo.pdf.

Adjoint Gassner. Renaming $\bar{a}_{ij} = a_{ij}/b_i$ and $t_i = e^{b_i}$, get $[\bar{a}_{ij}, \bar{a}_{ik}] = 0$, $[\bar{a}_{ik}, \bar{a}_{jk}] = -[\bar{a}_{ij}, \bar{a}_{jk}] = \bar{a}_{ik} - \bar{a}_{jk}$, and $(\text{mod } \langle \bar{a}_{ii} \rangle) [\bar{a}_{ij}, \bar{a}_{ji}] = \bar{a}_{ji} - \bar{a}_{ij}$, so

$$\bar{a}_{kj} \mapsto t_i^{-1} \bar{a}_{kj} + (1 - t_i^{-1}) \bar{a}_{ij},$$

$$\bar{a}_{ki} \mapsto \bar{a}_{ki} + (1 - t_i^{-1}) \bar{a}_{kj} + (t_i^{-1} - 1) \bar{a}_{ij},$$

$$\bar{a}_{jk} \mapsto t_i \bar{a}_{jk} + (1 - t_i) \bar{a}_{ik}, \quad \bar{a}_{ji} \mapsto t_i \bar{a}_{ji} + (1 - t_i) \bar{a}_{ij}.$$

Questions. • As Gassner is Γ calculus, Adjoint Gassner must factor through Gassner. **How?** • Interpretation? π_T -Artin?

$$\begin{array}{c} j \\ \swarrow \\ k \text{ --- } [a_{jk}, a_{kl}] \text{ --- } l \\ \uparrow \\ b \end{array} = \frac{b}{b_j a_{kl}} - \frac{\text{arc}}{b_k a_{jl}} - \frac{\text{Y-junction}}{\gamma_{jkl}}$$

OneCo Monoblog.

(151019a) Perhaps I should switch to a circuit algebra perspective, plus meta-monoid ops.

2Dv. b : bracket trace; c : cobracket trace; $\langle b, c \rangle = \delta \in \{0, 1\}$; $\deg b_i = \deg c_j = \deg a_{ij} = \deg \delta = 1$. Implementation/verification: pensieve://2015-08/nb/abc.pdf.

\mathcal{A}^{2Dv} is $\mathbb{Q}[[\delta]]FA(b_i, c_j, a_{ij})$ (so $\mathcal{L}^v = \{f + f^{ij} a_{ij}\}$) modulo locality,

tt.

$$[a_{jk}, a_{jl}] = c_l a_{jk} - c_k a_{jl},$$

hh.

$$[a_{jk}, a_{ik}] = b_i a_{jk} - b_j a_{ik},$$

Swinging. $\delta a_{ij} a_{kl} - \delta a_{il} a_{jk} = b_k c_l a_{ij} - b_i c_l a_{kj} - b_k c_j a_{il} + b_i c_j a_{kl}$

ht.

$$[a_{jk}, a_{kl}] = b_j a_{kl} - b_k a_{jl} - c_l a_{jk} + c_k a_{jl},$$

ab,ac.

$$\text{ad } a_{jk}: b_j, -b_k, -c_j, c_k \mapsto \gamma_{jk} := \delta a_{jk} - b_j c_k,$$

Backie.

$$[a_{jk}, a_{kj}] =$$

$$(b_j + c_k) a_{kj} - (b_k + c_j) a_{jk} + (b_j - c_j) a_{kk} - (b_k - c_k) a_{jj} + \gamma_{jk} - \gamma_{kj},$$

$$\text{with } \gamma_{jk} := \delta a_{jk} - b_j c_k,$$

bc.

$$[b_i, c_j] = 0.$$

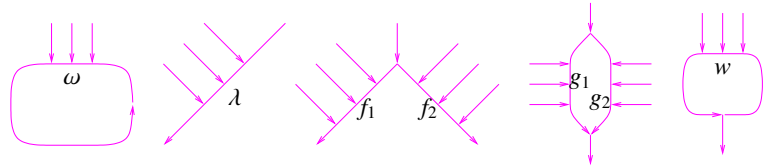
$$\text{So } a_{ij} f = f^\delta a_{ij} - \frac{b_i c_j}{\delta} (f^\delta - f), \quad [a_{ij}, f] = (f^\delta - f) \left(a_{ij} - \frac{b_i c_j}{\delta} \right),$$

$$\text{with } f^\delta := f // \begin{pmatrix} b_i \rightarrow b_i + \delta & b_j \rightarrow b_j - \delta \\ c_i \rightarrow c_i - \delta & c_j \rightarrow c_j + \delta \end{pmatrix}.$$

The Ascending Algebra \mathcal{A}_+^{2Dv} . Same but with only a_{ij} , $i < j$.

The OneCo Sub-Quotient is $\langle a_{ij} \rangle$ modulo $\delta^2 = \delta c_i = c_j c_k = 0$, so \mathcal{L}^{1co} is (coefficient functions non-central, in $\mathbb{Q}[[b_i]]$)

The 1co Graphs.



(151019c) Make the braid representation presentable?

(151019b) Switch to an EK basis?

(151019a) To do: Find and implement the group-like condition.