Pensieve header: Computations in the ybax algebra using the Drinfel’d double (at $\epsilon^2=0$. Continues ExpDoubleEpsilonSquare5@@.nb.

The double at $\epsilon^2 = 0$

Utilities

Canonical Form:

```
In[∘]:= CF[sd_SeriesData] := MapAt[CF, sd, 3];
CF[ℰ_] := ExpandDenominator@
   ExpandNumerator@Together[Expand[ℰ] //. e^x_ e^y_ :→ e^(x+y) /. e^x_ :→ e^CF[x]];
```

The Kronecker δ:

```
In[∘]:= Kδ /: Kδ_{i_,j_} := If[i === j, 1, 0];
```

Equality and multiplication of perturbed Gaussians; $\mathbb{E}[L, Q, P]$ stands for $e^{L+Q}P$:

```
In[∘]:= 𝔼 /: 𝔼[L1_, Q1_, P1_] ≡ 𝔼[L2_, Q2_, P2_] :=
   CF[L1 == L2] ∧ CF[Q1 == Q2] ∧ CF[Normal[P1 - P2] == 0];
𝔼 /: 𝔼[L1_, Q1_, P1_] 𝔼[L2_, Q2_, P2_] := 𝔼[L1 + L2, Q1 + Q2, P1 * P2];
```

Zip and Bind

```
In[∘]:= {b*, y*, a*, x*, z*} = {β, η, α, ξ, ζ};
{β*, η*, α*, ξ*, ζ*} = {b, y, a, x, z}; (u_i_)* := (u*)_i;
```

```
In[∘]:= expand[sd_SeriesData] := MapAt[expand, sd, 3];
expand[ℰ_] := Expand[ℰ];
Zip_{}[P_] := P;
Zip_{ζ_,ζs___}[P_] := (expand[P // Zip_{ζs}] /. f_. ζ^d_. :→ ∂_{ζ*,d} f) /. ζ* → 0
```

QZip implements the “Q-level zips” on $\mathbb{E}(L, Q, P) = Pe^{L+Q}$. Such zips regard the L variables as scalars.

```
In[∘]:= QZip_{ζs_List,simp_}@𝔼[L_, Q_, P_] := Module[{ζ, z, zs, c, ys, ηs, qt, zrule, Q1, Q2},
    zs = Table[ζ*, {ζ, ζs}];
    c = Q /. Alternatives @@ (ζs ⋃ zs) → 0;
    ys = Table[∂_ζ (Q /. Alternatives @@ zs → 0), {ζ, ζs}];
    ηs = Table[∂_z (Q /. Alternatives @@ ζs → 0), {z, zs}];
    qt = Inverse@Table[Kδ_{z,ζ*} - ∂_{z,ζ}Q, {ζ, ζs}, {z, zs}];
    zrule = Thread[zs → qt.(zs + ys)];
    Q2 = (Q1 = c + ηs.zs /. zrule) /. Alternatives @@ zs → 0;
    simp /@ 𝔼[L, Q2, Det[qt] e^-Q2 Zip_ζs[e^Q1 (P /. zrule)]] ];
QZip_{ζs_List} := QZip_{ζs,CF};
```

Upper to lower and lower to Upper:

In[ ]:=
```
U2l = {B_{i_}^{p_·} → e^{-p b_i}, B^{p_·} → e^{-p b}, 𝒜_{i_}^{p_·} → e^{p α_i}, 𝒜^{p_·} → e^{p α}};
l2U = {e^{c_· b_{i_}+d_·} :→ B_i^{-c} e^d, e^{c_· b+d_·} :→ B^{-c} e^d,
    e^{c_· α_{i_}+d_·} :→ 𝒜_i^c e^d, e^{c_· α+d_·} :→ 𝒜^c e^d,
    e^{ℰ_} :→ e^{Expand@ℰ}};
```

LZip implements the “L-level zips” on $\mathbb{E}(L, Q, P) = Pe^{L+Q}$. Such zips regard all of Pe^Q as a single”P”. Here the z’s are b and α and the ζ’s are β and a.

In[ ]:=
```
LZip_{ζs_List,simp_}@𝔼[L_, Q_, P_] := Module[{ζ, z, zs, c, ys, ηs, lt, zrule, L1, L2, Q1, Q2},
    zs = Table[ζ*, {ζ, ζs}];
    c = L /. Alternatives @@ (ζs ⋃ zs) → 0;
    ys = Table[∂_ζ (L /. Alternatives @@ zs → 0), {ζ, ζs}];
    ηs = Table[∂_z (L /. Alternatives @@ ζs → 0), {z, zs}];
    lt = Inverse@Table[Kδ_{z,ζ*} - ∂_{z,ζ} L, {ζ, ζs}, {z, zs}];
    zrule = Thread[zs → lt.(zs + ys)];
    L2 = (L1 = c + ηs.zs /. zrule) /. Alternatives @@ zs → 0;
    Q2 = (Q1 = Q /. U2l /. zrule) /. Alternatives @@ zs → 0;
    simp /@ 𝔼[L2, Q2, Det[lt] e^{-L2-Q2} Zip_ζs[e^{L1+Q1} (P /. U2l /. zrule)]] //. l2U];
LZip_{ζs_List} := LZip_{ζs,CF};
```

In[ ]:=
```
Bind_{}[L_, R_] := L R;
Bind_{is__}[L_𝔼, R_𝔼] := Module[{n},
    Times[
        L /. Table[(v : b | B | a | x | y)_i → v_{n@i}, {i, {is}}],
        R /. Table[(v : β | α | 𝒜 | ξ | η)_i → v_{n@i}, {i, {is}}]
    ] // LZip_{Flatten@Table[{β_{n@i},a_{n@i}},{i,{is}}]} // QZip_{Flatten@Table[{ξ_{n@i},y_{n@i}},{i,{is}}]}];
B_{l_List}[L_, R_] := Bind_l[L, R]; B_{is___}[L_, R_] := Bind_{is}[L, R];
```

The two halves

In[∘]:=
```
(*Hopf algebra on the a,x side*)
tam[i_,j_->k_] := 𝔼[(α_i + α_j) a_k, (ⅇ^(-α_j) ξ_i + ξ_j) x_k, 1 + O[ϵ]^2]
taΔ[i_->j_,k_] := 𝔼[α_i (a_j + a_k), ξ_i (x_j + x_k), 1 + ϵ ξ_i x_k (-a_j + 1/2 ξ_i x_j) + O[ϵ]^2]
taS[i_] := 𝔼[-α_i a_i, - ⅇ^(α_i) ξ_i x_i, 1 - ϵ ⅇ^(α_i) ξ_i x_i (a_i + 1/2 ⅇ^(α_i) ξ_i x_i) + O[ϵ]^2]
taSi[i_] := 𝔼[-α_i a_i, -ⅇ^(α_i) ξ_i x_i, 1 - ϵ ⅇ^(α_i) ξ_i x_i (a_i - 1 + 1/2 ⅇ^(α_i) ξ_i x_i) + O[ϵ]^2]
(*Hopf algebra on the y,b side*)
tbm[i_,j_→k_] := 𝔼[(β_i + β_j) b_k, (η_i + η_j) y_k, 1 - ϵ η_j y_k β_i + O[ϵ]^2]
tbΔ[i_->j_,k_] := 𝔼[β_i (b_j + b_k), η_i (ⅇ^(-b_k) y_j + y_k), 1 + 1/2 ϵ η_i^2 y_j y_k ⅇ^(-b_k) + O[ϵ]^2]
tbS[i_] := 𝔼[-β_i b_i, -ⅇ^(b_i) η_i y_i, 1 - ϵ ⅇ^(b_i) η_i y_i (β_i + 1/2 ⅇ^(b_i) η_i y_i) + O[ϵ]^2]
tbSi[i_] := 𝔼[-β_i b_i, -ⅇ^(b_i) η_i y_i, 1 - ϵ ⅇ^(b_i) η_i y_i (β_i - 1 + 1/2 ⅇ^(b_i) η_i y_i) + O[ϵ]^2]
```

First check that on the generators this agrees with our conventions in SLPortfolio.pdf with $\hbar = \gamma = 1$

In[∘]:=
```
{
  "[a,x]" → ((𝔼[0, 0, a_2 x_1] ~B_{1,2}~ tam_{1,2→1})⟦3⟧ - (𝔼[0, 0, a_1 x_2] ~B_{1,2}~ tam_{1,2→1})⟦3⟧),
  "[b,y]" → ((𝔼[0, 0, y_2 b_1] ~B_{1,2}~ tbm_{1,2→1})⟦3⟧ - (𝔼[0, 0, y_1 b_2] ~B_{1,2}~ tbm_{1,2→1})⟦3⟧)
} /. {z_1 → z} // Simplify
(Δ[#] → Simplify@Normal@Last[𝔼[0, 0, #_1] ~B_1~ taΔ_{1→1,2}]) & /@ {a, x}
(Δ[#] → Simplify@Normal@Last[𝔼[0, 0, #_1] ~B_1~ tbΔ_{1→1,2}]) & /@ {b, y}
{
  " S(a) = " ((𝔼[0, 0, a_1] ~B_1~ taS_1)⟦3⟧),
  " S(x) = " ((𝔼[0, 0, x_1] ~B_1~ taS_1)⟦3⟧),
  " S(b) = " ((𝔼[0, 0, b_1] ~B_1~ tbS_1)⟦3⟧),
  " S(y) = " → ((𝔼[0, 0, y_1] ~B_1~ tbS_1)⟦3⟧)
} /. {z_1 → z} // Simplify
```

Out[∘]= $\{[a,x] \to -x + O[\epsilon]^2,\ [b,y] \to -y\,\epsilon + O[\epsilon]^2\}$

Out[∘]= $\{\Delta[a] \to a_1 + a_2,\ \Delta[x] \to x_1 + (1 - \epsilon\, a_1)\, x_2\}$

Out[∘]= $\{\Delta[b] \to b_1 + b_2,\ \Delta[y] \to B_2\, y_1 + y_2\}$

Out[∘]= $\{-\ S(a)\ =\ a + O[\epsilon]^2,\ -\ S(x)\ =\ x -\ S(x)\ =\ a\, x\, \epsilon + O[\epsilon]^2,\ -\ S(b)\ =\ b + O[\epsilon]^2,\ S(y)\ =\ \to -\frac{y}{B} + O[\epsilon]^2\}$

Hopf algebra axioms on both sides separately

(co)-associativity on both sides

In[∘]:= $\{(ta\Delta_{1\to1,2}\sim B_2\sim ta\Delta_{2\to2,3})\equiv(ta\Delta_{1\to1,3}\sim B_1\sim ta\Delta_{1\to1,2}),$
$(tb\Delta_{1\to1,2}\sim B_2\sim tb\Delta_{2\to2,3})\equiv(tb\Delta_{1\to1,3}\sim B_1\sim tb\Delta_{1\to1,2}),$
$(tam_{1,2\to1}\sim B_1\sim tam_{1,3\to1})\equiv(tam_{2,3\to2}\sim B_2\sim tam_{1,2\to1}),$
$(tbm_{1,2\to1}\sim B_1\sim tbm_{1,3\to1})\equiv(tbm_{2,3\to2}\sim B_2\sim tbm_{1,2\to1})\}$

Out[∘]= {True, True, True, True}

Δ is an algebra morphism

In[∘]:= $\{tam_{1,2\to1}\sim B_1\sim ta\Delta_{1\to1,2}\equiv(ta\Delta_{1\to1,3}\,ta\Delta_{2\to2,4})\sim B_{1,2,3,4}\sim(tam_{3,4\to2}\,tam_{1,2\to1}),$
$tbm_{1,2\to1}\sim B_1\sim tb\Delta_{1\to1,2}\equiv(tb\Delta_{1\to1,3}\,tb\Delta_{2\to2,4})\sim B_{1,2,3,4}\sim(tbm_{3,4\to2}\,tbm_{1,2\to1})\}$

Out[∘]= {True, True}

S is convolution inverse of id

In[∘]:= $\{(ta\Delta_{1\to1,2}\sim B_1\sim taS_1)\sim B_{1,2}\sim tam_{1,2\to1},\ (ta\Delta_{1\to1,2}\sim B_2\sim taS_2)\sim B_{1,2}\sim tam_{1,2\to1}\}$
$\{(tb\Delta_{1\to1,2}\sim B_1\sim tbS_1)\sim B_{1,2}\sim tbm_{1,2\to1},\ (tb\Delta_{1\to1,2}\sim B_2\sim tbS_2)\sim B_{1,2}\sim tbm_{1,2\to1}\}$

Out[∘]= $\{\mathbb{E}[0, 0, 1+O[\epsilon]^2], \mathbb{E}[0, 0, 1+O[\epsilon]^2]\}$

Out[∘]= $\{\mathbb{E}[0, 0, 1+O[\epsilon]^2], \mathbb{E}[0, 0, 1+O[\epsilon]^2]\}$

Si is the inverse of S

In[∘]:= $\{taSi_1\sim B_1\sim taS_1\equiv\mathbb{E}[a_1\alpha_1, x_1\xi_1, 1], taS_1\sim B_1\sim taSi_1\equiv\mathbb{E}[a_1\alpha_1, x_1\xi_1, 1]\}$
$\{tbSi_1\sim B_1\sim tbS_1\equiv\mathbb{E}[b_1\beta_1, y_1\eta_1, 1], tbS_1\sim B_1\sim tbSi_1\equiv\mathbb{E}[b_1\beta_1, y_1\eta_1, 1]\}$

Out[∘]= {True, True}

Out[∘]= {True, True}

S is an algebra anti-(co)morphism

In[∘]:= $\{tam_{1,2\to1}\sim B_1\sim taS_1\equiv(taS_1\,taS_2)\sim B_{1,2}\sim tam_{2,1\to1}, tbm_{1,2\to1}\sim B_1\sim tbS_1\equiv(tbS_1\,tbS_2)\sim B_{1,2}\sim tbm_{2,1\to1}\}$
$\{taS_1\sim B_1\sim ta\Delta_{1\to1,2}\equiv ta\Delta_{1\to2,1}\sim B_{1,2}\sim(taS_1\,taS_2), tbS_1\sim B_1\sim tb\Delta_{1\to1,2}\equiv tb\Delta_{1\to2,1}\sim B_{1,2}\sim(tbS_1\,tbS_2)\}$

Out[∘]= {True, True}

Out[∘]= {True, True}

Pairing

In[∘]:= $tP_{i\_,j\_} := \mathbb{E}\left[\beta_i\,\alpha_j, \eta_i\,\xi_j, 1+\frac{1}{4}\epsilon\,\eta_i^2\,\xi_j^2\right]$

In[∘]:=
```
qfac[k_, q_] := (1 - q)^-k QPochhammer[q, q, k] // FunctionExpand
qfe[k_] := Normal[Series[qfac[k, E^ρ], {ρ, 0, 1}]] /. {ρ → ϵ}
Table[𝔼[0, 0, y1^r b1^s a2^t x2^u] ~ B1,2 ~ tP1,2 ≡ 𝔼[0, 0, Kδr,u Kδs,t qfe[r] s!],
   {r, 0, 4}, {s, 0, 4}, {t, 0, 4}, {u, 0, 4}] // Flatten // Union
```

Out[∘]= {True}

Pairing axioms

In[∘]:=
$$\{(\mathrm{tbm}_{1,2\to1}\,\mathbb{E}[\alpha_3 a_3,\ \xi_3 x_3,\ 1])\sim B_{1,3}\sim tP_{1,3} \equiv (\mathbb{E}[\beta_1 b_1,\ \eta_1 y_1,\ 1]\ \mathbb{E}[\beta_2 b_2,\ \eta_2 y_2,\ 1]\ \mathrm{ta}\Delta_{3\to4,5})\sim B_{1,4}\sim tP_{1,4}\sim B_{2,5}\sim tP_{2,5},\ (\mathrm{tb}\Delta_{1\to1,2}\,\mathbb{E}[\alpha_3 a_3,\ \xi_3 x_3,\ 1]\ \mathbb{E}[\alpha_4 a_4,\ \xi_4 x_4,\ 1])\sim B_{1,3}\sim tP_{1,3}\sim B_{2,4}\sim tP_{2,4} \equiv (\mathbb{E}[\beta_1 b_1,\ \eta_1 y_1,\ 1]\ \mathrm{tam}_{3,4\to3})\sim B_{1,3}\sim tP_{1,3}\}$$

Out[∘]= {True, True}

In[∘]:=
$$\{(\mathrm{tbS}_1\,\mathbb{E}[\alpha_2 a_2,\ \xi_2 x_2,\ 1])\sim B_{1,2}\sim tP_{1,2} \equiv (\mathbb{E}[\beta_1 b_1,\ \eta_1 y_1,\ 1]\ \mathrm{taS}_2)\sim B_{1,2}\sim tP_{1,2},\ (\mathrm{tbSi}_1\,\mathbb{E}[\alpha_2 a_2,\ \xi_2 x_2,\ 1])\sim B_{1,2}\sim tP_{1,2} \equiv (\mathbb{E}[\beta_1 b_1,\ \eta_1 y_1,\ 1]\ \mathrm{taSi}_2)\sim B_{1,2}\sim tP_{1,2}\}$$

Out[∘]= {True, True}

The Double

The double multiplication (should really bind the a’s and b’s separately)

In[∘]:=
```
Block[{i, j, k}, tdm_{i_,j_→k_} =
  Simplify /@ Expand /@ ((𝔼[β_i b_i + α_j a_j, η_i y_i + ξ_j x_j, 1] (taΔ_{i→h1,h2} ~ B_{h2} ~ taΔ_{h2→h2,h3})
       (tbΔ_{j→t1,t2} ~ B_{t2} ~ tbΔ_{t2→t2,t3})) ~ B_{h3} ~ taSi_{h3} ~ B_{t1,h3} ~ (tP_{t1,h3}) ~
     B_{t3,h1} ~ (tP_{t3,h1}) ~ B_{h2,j,i,t2} ~ (tam_{h2,j→k} tbm_{i,t2→-k}) /. {u_-k :> u_k}) ]
```

Out[∘]=
$$\mathbb{E}\Big[a_k(\alpha_i+\alpha_j)+b_k(\beta_i+\beta_j),\ y_k\Big(\eta_i+\frac{\eta_j}{\mathcal{A}_i}\Big)-(-1+B_k)\eta_j\xi_i+x_k\Big(\frac{\xi_i}{\mathcal{A}_j}+\xi_j\Big),$$
$$1+\frac{1}{4\mathcal{A}_i\mathcal{A}_j}\Big(2y_k\eta_j\big(2x_k\xi_i+\mathcal{A}_j(-2\beta_i+(1-3B_k)\eta_j\xi_i)\big)+\mathcal{A}_i\xi_i\big(x_k(-4\beta_j+2(1-3B_k)\eta_j\xi_i)+\mathcal{A}_j\eta_j(4a_kB_k+(1-4B_k+3B_k^2)\eta_j\xi_i)\big)\Big)\epsilon+O[\epsilon]^2\Big]$$

In[∘]:=
```
(*Deriving tdS using tdm*)
Block[{i}, tdS_{i_} = ((tbSi_1 taS_2) ~ B_{1,2} ~ tdm_{2,1→i}) /. {z_1|2 → z_i}]
```

Out[∘]=
$$\mathbb{E}\Big[-a_i\alpha_i-b_i\beta_i,\ \frac{-y_i\mathcal{A}_i\eta_i-B_ix_i\mathcal{A}_i\xi_i+\mathcal{A}_i\eta_i\xi_i-B_i\mathcal{A}_i\eta_i\xi_i}{B_i},$$
$$1+\frac{1}{4B_i^2}\big(4B_iy_i\mathcal{A}_i\eta_i-4B_iy_i\mathcal{A}_i\beta_i\eta_i-2y_i^2\mathcal{A}_i^2\eta_i^2-4a_iB_i^2x_i\mathcal{A}_i\xi_i-4B_i^2x_i\mathcal{A}_i\beta_i\xi_i-4B_i\mathcal{A}_i\eta_i\xi_i+4a_iB_i\mathcal{A}_i\eta_i\xi_i+4B_i^2\mathcal{A}_i\eta_i\xi_i-4B_ix_iy_i\mathcal{A}_i^2\eta_i\xi_i+4B_i\mathcal{A}_i\beta_i\eta_i\xi_i-4B_i^2\mathcal{A}_i\beta_i\eta_i\xi_i+6y_i\mathcal{A}_i^2\eta_i^2\xi_i-2B_iy_i\mathcal{A}_i^2\eta_i^2\xi_i-2B_i^2x_i^2\mathcal{A}_i^2\xi_i^2+6B_ix_i\mathcal{A}_i^2\eta_i\xi_i^2-2B_i^2x_i\mathcal{A}_i^2\eta_i\xi_i^2-3\mathcal{A}_i^2\eta_i^2\xi_i^2+4B_i\mathcal{A}_i^2\eta_i^2\xi_i^2-B_i^2\mathcal{A}_i^2\eta_i^2\xi_i^2\big)\epsilon+O[\epsilon]^2\Big]$$

In[∘]:=
```
(*Deriving tdΔ using tdm*)
Block[{i, j, k}, tdΔ_{i_→j_,k_} = (tbΔ_{i→3,1} taΔ_{i→2,4}) ~ B_{1,2,3,4} ~ (tdm_{3,4→k} tdm_{1,2→j})]
```

Out[∘]=
$$\mathbb{E}\Big[a_j\alpha_i+a_k\alpha_i+b_j\beta_i+b_k\beta_i,\ y_j\eta_i+B_jy_k\eta_i+x_j\xi_i+x_k\xi_i,\ 1+\frac{1}{2}\big(B_jy_jy_k\eta_i^2-2a_jx_k\xi_i+x_jx_k\xi_i^2\big)\epsilon+O[\epsilon]^2\Big]$$

First check the double formulas on the generators agree with SL2Portfolio.pdf:

In[•]:=
```
{
  "[a,y]" → ((𝔼[0, 0, y2 a1] ~B1,2~ tdm1,2→1)〚3〛 - (𝔼[0, 0, y1 a2] ~B1,2~ tdm1,2→1)〚3〛),
  "[b,x]" → ((𝔼[0, 0, x2 b1] ~B1,2~ tdm1,2→1)〚3〛 - (𝔼[0, 0, x1 b2] ~B1,2~ tdm1,2→1)〚3〛),
  "xy-qyx" → ((𝔼[0, 0, x1 y2] ~B1,2~ tdm1,2→1)〚3〛 - (1 + ϵ) (𝔼[0, 0, y1 x2] ~B1,2~ tdm1,2→1)〚3〛)
 } /. {z_1 → z} // Expand // Factor
{
  "Δ(a)" → ((𝔼[0, 0, a1] ~B1~ tdΔ1→1,2)〚3〛),
  "Δ(x)" → ((𝔼[0, 0, x1] ~B1~ tdΔ1→1,2)〚3〛),
  "Δ(b)" → ((𝔼[0, 0, b1] ~B1~ tdΔ1→1,2)〚3〛),
  "Δ(y)" → ((𝔼[0, 0, y1] ~B1~ tdΔ1→1,2)〚3〛)
 } // Simplify
{
  "S(a)" → ((𝔼[0, 0, a1] ~B1~ tdS1)〚3〛),
  "S(x)" → ((𝔼[0, 0, x1] ~B1~ tdS1)〚3〛),
  "S(b)" → ((𝔼[0, 0, b1] ~B1~ tdS1)〚3〛),
  "S(y)" → ((𝔼[0, 0, y1] ~B1~ tdS1)〚3〛)
 } /. {z_1 → z} // Simplify
```

Out[•]= $\{[a,y] \to -y + O[\epsilon]^2,\ [b,x] \to x\,\epsilon + O[\epsilon]^2,\ xy-qyx \to (1-B) + a\,B\,\epsilon + O[\epsilon]^2\}$

Out[•]= $\{\Delta(a) \to (a_1 + a_2) + O[\epsilon]^2,\ \Delta(x) \to (x_1 + x_2) - a_1 x_2\,\epsilon + O[\epsilon]^2,$
$\Delta(b) \to (b_1 + b_2) + O[\epsilon]^2,\ \Delta(y) \to (y_1 + B_1 y_2) + O[\epsilon]^2\}$

Out[•]= $\{S(a) \to -a + O[\epsilon]^2,\ S(x) \to -x - a\,x\,\epsilon + O[\epsilon]^2,\ S(b) \to -b + O[\epsilon]^2,\ S(y) \to -\frac{y}{B} + \frac{y\,\epsilon}{B} + O[\epsilon]^2\}$

Hopf algebra axioms on double

(co)-associativity

In[•]:=
```
{(tdΔ1→1,2 ~B2~ tdΔ2→2,3) ≡ (tdΔ1→1,3 ~B1~ tdΔ1→1,2),
 (tdm1,2→1 ~B1~ tdm1,3→1) ≡ (tdm2,3→2 ~B2~ tdm1,2->1)}
```

Out[•]= {True, True}

Δ is an algebra morphism

In[•]:=
```
tdm1,2→1 ~B1~ tdΔ1→1,2 ≡ (tdΔ1→1,3 tdΔ2→2,4) ~B1,2,3,4~ (tdm3,4→2 tdm1,2→1)
```

Out[•]= True

S is convolution inverse of id

In[•]:=
```
{(tdΔ1→1,2 ~B1~ tdS1) ~B1,2~ tdm1,2→1, (tdΔ1→1,2 ~B2~ tdS2) ~B1,2~ tdm1,2→1}
```

Out[•]= $\{\mathbb{E}[0, 0, 1 + O[\epsilon]^2],\ \mathbb{E}[0, 0, 1 + O[\epsilon]^2]\}$

S is a (co)-algebra anti-morphism

In[•]:=
```
{tdm1,2→1 ~B1~ tdS1 ≡ (tdS1 tdS2) ~B1,2~ tdm2,1→1,
 tdS1 ~B1~ tdΔ1→1,2 ≡ tdΔ1→2,1 ~B1,2~ (tdS1 tdS2)} // Expand
```

Out[•]= {True, True}

R-matrix

In[∘]:=
```
𝕖_{q_,k_}[x_] := 𝕖^(Sum[(1 - q)^j x^j / (j (1 - q^j)), {j, 1, k + 1}])
tR_{i_,j_} := 𝔼[b_i a_j, y_i x_j, 1 - ϵ (1/4) y_i^2 x_j^2 + O[ϵ]^2]
  (*First two terms in Faddeev-Quesne formula*)
```

In[∘]:=
```
Series[𝕖_{q,1}[z] /. {z → y_i x_j, q → 1 + ρ}, {ρ, 0, 1}] /. {ρ → ϵ}
```

Out[∘]= $e^{x_j y_i} - \frac{1}{4}\left(e^{x_j y_i} x_j^2 y_i^2\right) \epsilon + O[\epsilon]^2$

Quasi-triangular axiom 1:

In[∘]:=
```
tR_{1,2} ~ B_1 ~ tdΔ_{1->1,3} ≡ (tR_{1,4} tR_{3,2}) ~ B_{2,4} ~ tdm_{2,4→2}
```

Out[∘]= True

Quasi-triangular axiom 2:

In[∘]:=
```
((tdΔ_{1->1,2} tR_{3,4}) ~ B_{1,2,3,4} ~ (tdm_{1,3→1} tdm_{2,4→2})) ≡
  ((tdΔ_{1→2,1} tR_{3,4}) ~ B_{1,2,3,4} ~ (tdm_{3,1→1} tdm_{4,2→2}))
```

Out[∘]= True

Reidemeister 3:

In[∘]:=
```
((tR_{1,2} tR_{4,3} tR_{5,6}) ~ B_{1,4} ~ tdm_{1,4→1} ~ B_{2,5} ~ tdm_{2,5→2} ~ B_{3,6} ~ tdm_{3,6→3}) ≡
  ((tR_{1,6} tR_{2,3} tR_{4,5}) ~ B_{1,4} ~ tdm_{1,4→1} ~ B_{2,5} ~ tdm_{2,5→2} ~ B_{3,6} ~ tdm_{3,6→3})
```

Out[∘]= True

In[∘]:=
```
Block[{i, j}, \overline{tR}_{i_,j_} = Expand /@ tR_{i,j} ~ B_j ~ tdS_j]
```

Out[∘]= $\mathbb{E}\left[-a_j b_i, -\frac{x_j y_i}{B_i}, 1 + \frac{\left(-4 a_j B_i x_j y_i - 3 x_j^2 y_i^2\right) \epsilon}{4 B_i^2} + O[\epsilon]^2\right]$

Reidemeister 2

In[∘]:=
```
{(\overline{tR}_{1,2} tR_{3,4}) ~ B_{1,2,3,4} ~ (tdm_{1,3→1} tdm_{2,4→2}), (tR_{1,2} \overline{tR}_{3,4}) ~ B_{1,2,3,4} ~ (tdm_{1,3→1} tdm_{2,4→2})}
```

Out[∘]= $\{\mathbb{E}[0, 0, 1 + O[\epsilon]^2], \mathbb{E}[0, 0, 1 + O[\epsilon]^2]\}$

In[∘]:=
```
dm = tdm; dS = tdS; dΔ = tdΔ; R = tR;
```

Deriving the Drinfeld element u and its inverse ui

In[∘]:=
```
Block[{i}, {
  u_i_ = R_{1,2} ~ B_1 ~ dS_1 ~ B_{1,2} ~ dm_{2,1→i},
  ui_i_ := R_{1,2} ~ B_2 ~ dS_2 ~ B_2 ~ dS_2 ~ B_{1,2} ~ dm_{2,1→i}
 }]
```

Out[∘]= $\left\{\mathbb{E}\left[-a_i b_i,\ -\frac{x_i y_i}{B_i},\ B_i + \frac{\left(-4 a_i B_i^2 - 4 B_i x_i y_i - 4 a_i B_i x_i y_i - 3 x_i^2 y_i^2\right)\epsilon}{4 B_i} + O[\epsilon]^2\right],\ \text{Null}\right\}$

u and ui are inverses

In[∘]:= $(u_1\, ui_2) \sim B_{1,2} \sim dm_{1,2\to 1}$

Out[∘]= $\mathbb{E}\left[0,\ 0,\ 1 + O[\epsilon]^2\right]$

The ribbon element v satisfies $v^2 = S(u)\,u$. The spinner C=uv^{-1}.
It is convenient to compute $z = S(u)\,u^{-1}$ which is something easy.

In[∘]:= $((u_1 \sim B_1 \sim dS_1)\ ui_2) \sim B_{1,2} \sim dm_{1,2\to 1}$

Out[∘]= $\mathbb{E}\left[0,\ 0,\ \frac{1}{B_1} + \frac{a_1 \epsilon}{B_1} + O[\epsilon]^2\right]$

(* Needs fixing! *) So in our case S(u) = u z so S(u)u = $u^2\, z$ and v = u$z^{\frac{1}{2}}$ and finally C = uv^{-1} = $z^{-\frac{1}{2}} = e^{t_1/2}(1 - \epsilon\, a_1)$.

In[∘]:=
```
Block[{i},
 {CC_i_ = 𝔼[0, 0, B_i^(1/2) e^(-ϵ a_i/2) + O[ϵ]^2],
  CC̄_i_ = 𝔼[0, 0, B_i^(-1/2) e^(ϵ a_i/2) + O[ϵ]^2]
 }]
```

Out[∘]= $\left\{\mathbb{E}\left[0,\ 0,\ \sqrt{B_i} - \frac{1}{2}\left(a_i \sqrt{B_i}\right)\epsilon + O[\epsilon]^2\right],\ \mathbb{E}\left[0,\ 0,\ \frac{1}{\sqrt{B_i}} + \frac{a_i\,\epsilon}{2\sqrt{B_i}} + O[\epsilon]^2\right]\right\}$

In[∘]:=
```
Block[{i, j}, {
  Kink_i_ = (R_{1,3} CC̄_2) ~ B_{1,2} ~ dm_{1,2→1} ~ B_{1,3} ~ dm_{1,3→i},
  Kink̄_j_ = (R̄_{1,3} CC_2) ~ B_{1,2} ~ dm_{1,2→1} ~ B_{1,3} ~ dm_{1,3→j}
 }]
```

Out[∘]= $\left\{\mathbb{E}\left[a_i b_i,\ x_i y_i,\ \frac{1}{\sqrt{B_i}} + \frac{\left(2 a_i - x_i^2 y_i^2\right)\epsilon}{4\sqrt{B_i}} + O[\epsilon]^2\right],\right.$
$\left.\mathbb{E}\left[-a_j b_j,\ -\frac{x_j y_j}{B_j},\ \sqrt{B_j} + \frac{\left(-2 a_j B_j^2 - 4 a_j B_j x_j y_j - 3 x_j^2 y_j^2\right)\epsilon}{4 B_j^{3/2}} + O[\epsilon]^2\right]\right\}$

In[∘]:=
```
k2 = (R_{3,1} CC_2) ~ B_{1,2} ~ dm_{1,2→1} ~ B_{1,3} ~ dm_{1,3→i} /. e → E;
k4 = (R̄_{3,1} CC̄_2) ~ B_{1,2} ~ dm_{1,2→1} ~ B_{1,3} ~ dm_{1,3→j} /. e → E;
Simplify@{Kink_i ≡ k2, Kink̄_j ≡ k4, (Kink_i Kink̄_j) ~ B_{i,j} ~ dm_{i,j→1}}
```

Out[∘]= $\left\{\text{True},\ \text{True},\ \mathbb{E}\left[0,\ 0,\ 1 + O[\epsilon]^2\right]\right\}$

Reidemeister 2:

In[∘]:= $\left(R_{1,2}\,\overline{R}_{3,4}\right) \sim B_{1,3} \sim dm_{1,3\to1} \sim B_{2,4} \sim dm_{2,4\to2}$

Out[∘]= $\mathbb{E}\left[0,\,0,\,1+O[\epsilon]^2\right]$

Cyclic Reidemeister 2:

In[∘]:= $\left(R_{1,4}\,\overline{R}_{5,2}\,\overline{CC}_3\right) \sim B_{2,4} \sim dm_{2,4\to2} \sim B_{1,3} \sim dm_{1,3\to1} \sim B_{1,5} \sim dm_{1,5\to1} \equiv \overline{CC}_1$

Out[∘]= True

Trefoil

In[∘]:=
```
Timing[
 Z = R1,5 R6,2 R3,7 CC4 Kink8 Kink9 Kink10;
 Do[Print["doing ", r]; Z = Z~B1,r~dm1,r→1, {r, 2, 10}];
 Simplify@Z[[3]] ]
```

$Z = R_{1,5}\,R_{6,2}\,R_{3,7}\,\overline{CC}_4\,\overline{Kink}_8\,\overline{Kink}_9\,\overline{Kink}_{10}$; Do[Print["doing ", r]; $Z = Z \sim B_{1,r} \sim dm_{1,r\to1}$, {r, 2, 10}];

doing 2

doing 3

doing 4

doing 5

doing 6

doing 7

doing 8

doing 9

doing 10

Out[∘]= $\left\{71.3125,\ \frac{B_1}{1-B_1+B_1^2}+\left(B_1\left(-B_1+2B_1^2+2B_1^4+a_1\left(-1+B_1-B_1^3+B_1^4\right)-2x_1y_1-B_1^3\left(3+2x_1y_1\right)\right)\epsilon\right)\Big/\left(1-B_1+B_1^2\right)^3+O[\epsilon]^2\right\}$