Pensieve header: A unified verification notebook for the sl_3 project.

Prolog

```
In[•]:= HL[ℰ_] := Style[ℰ, Background → Yellow];
```

Initialization / Utilities

The "degree carrier / filtration parameter" is $\hbar$, and all "coupling constants" are proportional to it.

TD

```
In[•]:= $p = 2; $k = 1;
        If[$k == 0, ϵ = 0, ϵ /: ϵ^k_ /; k > $k := 0];
        SetAttributes[{SS, SST}, HoldAll];
        TRule = {T_i_ → ℯ^(ħ t_i), T → ℯ^(ħ t)}; qrule = q → ℯ^(2 ϵ ħ);
        SS[ℰ_] := Block[{ħ, ϵ}, (* Shielded Series *)
           Collect[Normal@Series[ℰ, {ħ, 0, $p}], ħ, Together] ];
        SST[ℰ_] :=
          Block[{ħ, ϵ}, Collect[Normal@Series[ℰ /. TRule, {ħ, 0, $p}], ħ, Together] ];
        Simp[ℰ_, op_] := Collect[ℰ, _CU | _QU, op];
        Simp[ℰ_] := Simp[ℰ, Collect[Normal@Series[#, {ħ, 0, $p}], ħ, Expand] &];
        SimpT[ℰ_] :=
          Collect[ℰ, _CU | _QU, Collect[Normal@Series[# /. TRule, {ħ, 0, $p}], ħ, Expand] &];
```

Differential polynomials (DP):

Utils

```
In[•]:= DP_{α_→D_x, β_→D_y}[P_][λ_] :=
         Total[CoefficientRules[P, {α, β}] /. ({m_, n_} → c_) :⇒ c D[λ, {x, m}, {y, n}]]
```

DeclareAlgebra

QLImplementation

```
In[•]:= Unprotect[NonCommutativeMultiply]; Attributes[NonCommutativeMultiply] = {};
        (NCM = NonCommutativeMultiply)[x_] := x;
        NCM[x_, y_, z__] := (x ⊠ y) ⊠ z;
        0 ⊠ _ = _ ⊠ 0 = 0;
        (x_Plus) ⊠ y_ := (# ⊠ y) & /@ x; x_ ⊠ (y_Plus) := (x ⊠ #) & /@ y;
        B[x_, x_] = 0;
        B[x_, y_] := x ⊠ y - y ⊠ x;
```

QLImplementation

```
In[•]:= DeclareAlgebra[U_Symbol, opts__Rule] := Module[{gp, sr, g, cp, M, CE, pow, k = 0,
            gs = Generators /. {opts}, cs = Centrals /. {opts} },
           (#_U = U@#) & /@ gs;
           gp = Alternatives @@ gs; gp = gp | gp_; (* gens *)
```

```
sr = Flatten@Table[{g → ++k, g_i_ → {i, k}}, {g, gs}]; (* sorting → *)
cp = Alternatives @@ cs; (* cents *)
SetAttributes[M, HoldRest]; M[0, _] = 0; M[a_, x_] := a x;
CE[ℰ_] := Collect[ℰ, _U, (Expand[#] /. ħ^d_ /; d > $p :→ 0) &];
U_i_[ℰ_] := ℰ /. {t : cp :→ t_i, u_U :→ Replace[u, x_ :→ x_i, 1]};
U_i_[NCM[]] = pow[ℰ_, 0] = U@{} = 1_U = U[];
B[U@(x_)_i_, U@(y_)_i_] := B[U@x_i, U@y_i] = U_i@B[U@x, U@y];
B[U@(x_)_i_, U@(y_)_j_] /; i =!= j := 0;
B[U@y_, U@x_] := CE[-B[U@x, U@y]];
x_ ⊠ (c_. 1_U) := CE[c x]; (c_. 1_U) ⊠ x_ := CE[c x];
(*(a_.*x_U)**(b_.*y_U) := CE[M[a b,x**y]];*)
(a_. U[xx___, x_]) ⊠ (b_. U[y_, yy___]) := If[OrderedQ[{x, y} /. sr],
  CE@M[a b, U[xx, x, y, yy]],
  U@xx ⊠ CE@M[a b, U@y ⊠ U@x + B[U@x, U@y]] ⊠ U@yy
 ];
U@{c_. * (l : gp)^n_, r___} /; FreeQ[c, gp] := CE[c U@Table[l, {n}] ⊠ U@{r}];
U@{c_. * l : gp, r___} := CE[c U[l] ⊠ U@{r}];
U@{c_, r___} /; FreeQ[c, gp] := CE[c U@{r}];
U@{l_Plus, r___} := CE[U@{#, r} & /@ l];
U@{l_, r___} := U@{Expand[l], r};
U[ℰ_NonCommutativeMultiply] := U /@ ℰ;
𝒪_U[specs___, poly_] := Module[{sp, null, vs, us},
  sp = Replace[{specs}, l_List :→ l_null, {1}];
  vs = Join @@ (First /@ sp);
  us = Join @@ (sp /. l_s_ :→ (l /. x_i_ :→ x_s));
  CE[Total[
     CoefficientRules[poly, vs] /. (p_ → c_) :→ c U@(us^p)
    ]] /. x_null :→ x];
pow[ℰ_, n_] := pow[ℰ, n - 1] ⊠ ℰ;
𝕊_U[ℰ_, ss___Rule] := CE@Total[
   CoefficientRules[ℰ, First /@ {ss}] /.
    (p_ → c_) :→ c NCM @@ MapThread[pow, {Last /@ {ss}, p}]];
m_j_→k_[c_. * u_U] := CE[((c /. (t : cp)_j → t_k) DeleteCases[u, _j|k]) ⊠
   U @@ Cases[u, w_j :→ w_k] ⊠ U @@ Cases[u, _k]];
S_i_[c_. * u_U] := CE[((c /. S_i[U, Centrals]) DeleteCases[u, _i]) ⊠
   U_i[NCM @@ Reverse@Cases[u, x_i :→ S@U@x]]];
Δ_i_→j_,k_[c_. * u_U] := CE[((c /. Δ_i→j,k[U, Centrals]) DeleteCases[u, _i]) ⊠
   NCM @@ Cases[u, x_i :→ DD_j,k@U@x]];
P_i_,j_[c_. * u_U] := CE[(P[U@Cases[c U[], _i] ⊠ (U @@ Cases[u, _i]),
       (U@Cases[c U[], _j]) ⊠ (U @@ Cases[u, _j])]
     DeleteCases[c U[], _i|j] ⊠ U @@ DeleteCases[u, _i|j])] /. U[] → 1];
```

DeclareMorphism

QLImplementation

```
In[•]:= DeclareMorphism[m_, U_ → V_, ongs_List, oncs_List : {}] := (
          Replace[ongs, (g_ → img_) :↦ (m[U[g]] = img), {1}];
          m[1_U] = 1_V;
          m[U[g_i_]] := V_i[m[U@g]];
          m[U[vs__]] := NCM @@ (m /@ U /@ {vs});
          m[ℰ_] := Simp[ℰ /. oncs /. u_U :↦ m[u]]; )
```

Meta-Operations

QLImplementation

```
In[•]:= m_{j_→j_} = Identity;
        m_{j_→k_}[ℰ_Plus] := Simp[m_{j→k} /@ ℰ];
        m_{is___,i_,j_→k_}[ℰ_] := m_{j→k}@m_{is,i→j}@ℰ;
        S_i_[ℰ_Plus] := Simp[S_i /@ ℰ];
        Δ_{i_→j_,k_}[ℰ_Plus] := Simp[Δ_{i→j,k} /@ ℰ];
        P_{i_,j_}[ℰ_Plus] := Simp[P_{i,j} /@ ℰ];
        Δ_{i_→j_,k_}[c_ * QU[]] :=
          Collect[(c /. Δ_{i→j,k}[QU, Centrals]) QU[], _U, (Expand[#] /. ħ^d_ /; d > $p :↦ 0) &];
        Δ_{i_→j_,k_}[c_ * CU[]] :=
          Collect[(c /. Δ_{i→j,k}[CU, Centrals]) CU[], _U, (Expand[#] /. ħ^d_ /; d > $p :↦ 0) &];
```

QLImplementation

Rule::rhs : Pattern j_ appears on the right–hand side of rule j_ → j_. »

Implementing CU = $\mathcal{U}(\mathrm{sl}_3^\epsilon)$

CU

```
In[•]:= DeclareAlgebra[CU, Generators → {X, Y, Z, b, a, z, y, x}, Centrals → {ss, tt}];
B[xCU, aCU] = 2 CU@x;
B[xCU, bCU] = -CU@x;
B[yCU, bCU] = 2 CU@y;
B[yCU, aCU] = -CU@y;
B[zCU, aCU] = CU@z;
B[zCU, bCU] = CU@z;
B[xCU, yCU] = zCU;
B[xCU, zCU] = 0;
B[aCU, bCU] = 0;
B[yCU, zCU] = 0;
B[aCU, XCU] = 2 XCU;
B[bCU, XCU] = -XCU;
B[aCU, YCU] = -YCU;
B[bCU, YCU] = 2 YCU;
B[aCU, ZCU] = ZCU; B[bCU, ZCU] = ZCU;
B[xCU, XCU] = ss 1CU + 2 ∈ CU[a];
B[yCU, YCU] = tt 1CU + 2 ∈ CU[b];
B[zCU, ZCU] = (ss + tt) 1CU + 2 ∈ (aCU + bCU);
B[yCU, ZCU] = XCU; B[xCU, ZCU] = -CU[Y];
B[zCU, XCU] = 2 ∈ yCU;
B[zCU, YCU] = -2 ∈ CU[x]; B[yCU, XCU] = 0;
B[xCU, YCU] = 0;
B[XCU, YCU] = 2 ∈ ZCU;
B[XCU, ZCU] = 0;
B[ZCU, YCU] = 0;
```

Verifying associativity on triples of generators:

```
In[•]:= With[{bas = CU /@ {X, Y, Z, b, a, z, y, x}},
 Table[z1 ⊠ (z2 ⊠ z3) - (z1 ⊠ z2) ⊠ z3 // Simp // HL,
    {z1, bas}, {z2, bas}, {z3, bas}] // Flatten // Union]
Out[•]=
{0}
```

```
In[•]:= With[{bas = CU /@ {X, Y, Z, b, a, z, y, x}},
 Table[B[aa, B[bb, cc] // Expand] + B[bb, B[cc, aa] // Expand] + B[cc, B[aa, bb] // Expand] //
     Simp, {aa, bas}, {bb, bas}, {cc, bas}] // Flatten // Union]
Out[•]=
{0}
```

Verifying associativity on a “random” triple:

```
In[•]:= With[{z1 = CU[y, y, a, a, x, x], z2 = CU[y, a, x], z3 = CU[y, y, a, x]}, {
          (rhs = (z1 ⊠ z2) ⊠ z3 // Simp) // Short,
          z1 ⊠ (z2 ⊠ z3) - rhs // Simp // HL
         }] // Timing
Out[•]=
{0.09375, {36 CU[y, y, a, a, z, z, z, x] +
   24 CU[y, y, a, a, a, z, z, z, x] + <<9>> + CU[y, y, a, a, a, a, y, y, y, x, x, x, x], 0}}
```

Implementing QU = $\mathcal{U}_q(\mathrm{sl}_3^\epsilon)$

```
In[•]:= (*s=exp[2A-B+ϵa], t=exp[2B-A+ϵb]*)
```

QU

```
In[•]:= DeclareAlgebra[QU, Generators → {X, Y, Z, b, a, z, y, x}, Centrals → {s, t}];
B[a_QU, x_QU] = -2 QU@x; B[b_QU, x_QU] = QU@x;
B[b_QU, y_QU] = -2 QU@y; B[a_QU, y_QU] = QU@y;
B[a_QU, z_QU] = -QU@z; B[b_QU, z_QU] = -QU@z;
B[y_QU, x_QU] = -z_QU + ϵ ħ QU[y, x];
B[z_QU, x_QU] = -ϵ ħ QU[z, x];
B[b_QU, a_QU] = 0;
B[z_QU, y_QU] = ϵ ħ QU[z, y];
B[a_QU, X_QU] = 2 X_QU; B[b_QU, X_QU] = -X_QU;
B[a_QU, Y_QU] = -Y_QU; B[b_QU, Y_QU] = 2 Y_QU;
B[a_QU, Z_QU] = Z_QU; B[b_QU, Z_QU] = Z_QU;
B[x_QU, X_QU] = -2 ϵ ħ QU@{X, x} + ((1 - s) QU[])/ħ + 2 s ϵ QU[a];
B[y_QU, Y_QU] = -2 ϵ ħ QU@{Y, y} + ((1 - t) QU[])/ħ + 2 t ϵ QU[b];
B[z_QU, Z_QU] = -2 ϵ ħ QU@{Z, z} + ((1 - s t) QU[])/ħ + 2 s t ϵ QU[a] + 2 s t ϵ QU[b];
B[y_QU, Z_QU] = X_QU - ϵ ħ QU@{Z, y};
B[x_QU, Z_QU] = -ϵ ħ QU@{Z, x} - (s + s ϵ ħ) QU[Y] + 2 s ϵ ħ QU[Y, a];
B[z_QU, X_QU] = 2 ϵ y_QU - ϵ ħ QU@{X, z};
B[z_QU, Y_QU] = -2 t ϵ QU[x] - ϵ ħ QU[Y, z]; B[y_QU, X_QU] = ϵ ħ QU[X, y];
B[x_QU, Y_QU] = ϵ ħ QU[Y, x];
B[X_QU, Y_QU] = 2 ϵ Z_QU - ϵ ħ QU[X, Y];
B[X_QU, Z_QU] = ϵ ħ QU[X, Z];
B[Z_QU, Y_QU] = ϵ ħ QU[Y, Z];
```

```
In[•]:= With[{bas = QU /@ {X, Y, Z, b, a, x, y, z}},
          Table[{z1, z2} → Simp[z1 ⊠ z2 - z2 ⊠ z1], {z1, bas}, {z2, bas}] ]
```

Out[•]=

$$\left\{\left\{\{QU[X],QU[X]\}\to 0,\ \{QU[X],QU[Y]\}\to 2\in QU[Z]-\in\hbar\, QU[X,Y],\right.\right.$$
$$\{QU[X],QU[Z]\}\to\in\hbar\, QU[X,Z],\ \{QU[X],QU[b]\}\to QU[X],\ \{QU[X],QU[a]\}\to -2\,QU[X],$$
$$\{QU[X],QU[x]\}\to\frac{(-1+s)\,QU[]}{\hbar}-2\,s\in QU[a]+2\in\hbar\, QU[X,x],$$
$$\left.\{QU[X],QU[y]\}\to -\in\hbar\, QU[X,y],\ \{QU[X],QU[z]\}\to -2\in QU[y]+\in\hbar\, QU[X,z]\right\},$$
$$\left\{\{QU[Y],QU[X]\}\to -2\in QU[Z]+\in\hbar\, QU[X,Y],\ \{QU[Y],QU[Y]\}\to 0,\right.$$
$$\{QU[Y],QU[Z]\}\to -\in\hbar\, QU[Y,Z],\ \{QU[Y],QU[b]\}\to -2\,QU[Y],\ \{QU[Y],QU[a]\}\to QU[Y],$$
$$\{QU[Y],QU[x]\}\to -\in\hbar\, QU[Y,x],\ \{QU[Y],QU[y]\}\to\frac{(-1+t)\,QU[]}{\hbar}-2\,t\in QU[b]+2\in\hbar\, QU[Y,y],$$
$$\left.\{QU[Y],QU[z]\}\to 2\,t\in QU[x]+\in\hbar\, QU[Y,z]\right\},\ \left\{\{QU[Z],QU[X]\}\to -\in\hbar\, QU[X,Z],\right.$$
$$\{QU[Z],QU[Y]\}\to\in\hbar\, QU[Y,Z],\ \{QU[Z],QU[Z]\}\to 0,\ \{QU[Z],QU[b]\}\to -QU[Z],$$
$$\{QU[Z],QU[a]\}\to -QU[Z],\ \{QU[Z],QU[x]\}\to (s+s\in\hbar)\,QU[Y]-2\,s\in\hbar\, QU[Y,a]+\in\hbar\, QU[Z,x],$$
$$\{QU[Z],QU[y]\}\to -QU[X]+\in\hbar\, QU[Z,y],$$
$$\left.\{QU[Z],QU[z]\}\to\frac{(-1+s\,t)\,QU[]}{\hbar}-2\,s\,t\in QU[a]-2\,s\,t\in QU[b]+2\in\hbar\, QU[Z,z]\right\},$$
$$\{\{QU[b],QU[X]\}\to -QU[X],\ \{QU[b],QU[Y]\}\to 2\,QU[Y],\ \{QU[b],QU[Z]\}\to QU[Z],$$
$$\{QU[b],QU[b]\}\to 0,\ \{QU[b],QU[a]\}\to 0,\ \{QU[b],QU[x]\}\to QU[x],$$
$$\{QU[b],QU[y]\}\to -2\,QU[y],\ \{QU[b],QU[z]\}\to -QU[z]\},$$
$$\{\{QU[a],QU[X]\}\to 2\,QU[X],\ \{QU[a],QU[Y]\}\to -QU[Y],\ \{QU[a],QU[Z]\}\to QU[Z],$$
$$\{QU[a],QU[b]\}\to 0,\ \{QU[a],QU[a]\}\to 0,\ \{QU[a],QU[x]\}\to -2\,QU[x],$$
$$\{QU[a],QU[y]\}\to QU[y],\ \{QU[a],QU[z]\}\to -QU[z]\},$$
$$\left\{\{QU[x],QU[X]\}\to\frac{(1-s)\,QU[]}{\hbar}+2\,s\in QU[a]-2\in\hbar\, QU[X,x],\ \{QU[x],QU[Y]\}\to\in\hbar\, QU[Y,x],\right.$$
$$\{QU[x],QU[Z]\}\to (-s-s\in\hbar)\,QU[Y]+2\,s\in\hbar\, QU[Y,a]-\in\hbar\, QU[Z,x],$$
$$\{QU[x],QU[b]\}\to -QU[x],\ \{QU[x],QU[a]\}\to 2\,QU[x],\ \{QU[x],QU[x]\}\to 0,$$
$$\left.\{QU[x],QU[y]\}\to QU[z]-\in\hbar\, QU[y,x],\ \{QU[x],QU[z]\}\to\in\hbar\, QU[z,x]\right\},$$
$$\left\{\{QU[y],QU[X]\}\to\in\hbar\, QU[X,y],\ \{QU[y],QU[Y]\}\to\frac{(1-t)\,QU[]}{\hbar}+2\,t\in QU[b]-2\in\hbar\, QU[Y,y],\right.$$
$$\{QU[y],QU[Z]\}\to QU[X]-\in\hbar\, QU[Z,y],\ \{QU[y],QU[b]\}\to 2\,QU[y],\ \{QU[y],QU[a]\}\to -QU[y],$$
$$\left.\{QU[y],QU[x]\}\to -QU[z]+\in\hbar\, QU[y,x],\ \{QU[y],QU[y]\}\to 0,\ \{QU[y],QU[z]\}\to -\in\hbar\, QU[z,y]\right\},$$
$$\left\{\{QU[z],QU[X]\}\to 2\in QU[y]-\in\hbar\, QU[X,z],\ \{QU[z],QU[Y]\}\to -2\,t\in QU[x]-\in\hbar\, QU[Y,z],\right.$$
$$\{QU[z],QU[Z]\}\to\frac{(1-s\,t)\,QU[]}{\hbar}+2\,s\,t\in QU[a]+2\,s\,t\in QU[b]-2\in\hbar\, QU[Z,z],$$
$$\{QU[z],QU[b]\}\to QU[z],\ \{QU[z],QU[a]\}\to QU[z],\ \{QU[z],QU[x]\}\to -\in\hbar\, QU[z,x],$$
$$\left.\{QU[z],QU[y]\}\to\in\hbar\, QU[z,y],\ \{QU[z],QU[z]\}\to 0\right\}\right\}$$

Verifying associativity on triples of generators:

```
In[•]:= With[{bas = QU /@ {Z, X, Y, a, b, y, x, z}},
          Table[HL[z1 ⊠ (z2 ⊠ z3) - (z1 ⊠ z2) ⊠ z3 // Simp],
             {z1, bas}, {z2, bas}, {z3, bas}] // Flatten // Union]
Out[•]=
        {0}
```