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

Continues pensieve://2017-06/ and pensieve://2017-08/.

```
SetDirectory["C:\\drorbn\\AcademicPensieve\\Projects\\PPSA"];
```

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

TD

```
$TℏD = ∞; ℏ /: ℏ^d_. /; d > $TℏD := 0;
$TϵD = 2; ϵ /: ϵ^d_. /; d > $TϵD := 0;
```

Implementing $\mathcal{U}_{\gamma\epsilon;\hbar}$

With $q = e^{\hbar\gamma\epsilon}$, $A = e^{-\hbar\epsilon a}$, and $[f, g]_q := fg - qgf$, our algebra is $\mathcal{U}_{\gamma\epsilon;\hbar} = \langle t, y, a, x\rangle/\mathcal{R}$, where $\mathcal{R} = \left([t, *] = 0,\ [a, y] = -\gamma y,\ [x, y]_q = \hbar^{-1}\left(1 - TA^2\right),\ [x, a] = -\gamma x\right)$.

QLImplementation

```
q := Sum[(ℏ γ ϵ)^k / k!, {k, 0, Min[$TϵD, $TℏD]}];
AlgebraAtom = y | a | x;
PBWRule = {y → 1, a → 2, x → 3};
B[U@a, U@y] = -γ U@y; B[U@x, U@a] = -γ U@x;
B[U@x, U@y] := (q - 1) UU[y, x] + UU[ℏ^-1 (1 - T^Min[$TϵD,$TℏD] Sum[(-2 a ϵ ℏ)^k / k!, {k, 0}])];
```

QLImplementation

```
x_ ≼ y_ := OrderedQ[{x, y} /. PBWRule]; x_ ≺ y_ := ! OrderedQ[{y, x} /. PBWRule];
Simp[ℰ_] := Collect[ℰ, _U, Expand];
```

QLImplementation

```
U_i_[ℰ_] := ℰ /. {t → t_i, T → T_i, u_U :> Replace[u, x_ :> x_i, 1]};
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[x_, x_] = 0;
B[U[y_], U[x_]] = Simp[-B[U[x], U[y]]];
B[x_, y_] := x ** y - y ** x;
```

QLImplementation

```
Unprotect[NonCommutativeMultiply];
NonCommutativeMultiply[x_] := x;
0 ** _ = _ ** 0 = 0;
x_ ** U[] := x; U[] ** x_ := x;
(a_ * x_U) ** (b_ * y_U) := If[a b === 0, 0, Simp[a b (x ** y)]];
(a_ * x_U) ** y_ := Simp[a (x ** y)]; x_ ** (a_ * y_U) := Simp[a (x ** y)];
(x_Plus) ** y_ := (# ** y) & /@ x; x_ ** (y_Plus) := (x ** #) & /@ y;
```

QLImplementation

```
U[xx___, x_] ** U[y_, yy___] :=
  If[x ≼ y, U[xx, x, y, yy], U@xx ** (U@y ** U@x + B[U@x, U@y]) ** U@yy];
```

QLImplementation

```
UU[c_. * (l : AlgebraAtom)^n_, r___] /; FreeQ[c, AlgebraAtom] :=
  Expand[c UU[Sequence @@ Table[l, {n}], r]];
UU[c_. * l : AlgebraAtom, r___] := Expand[c U[l] ** UU[r]];
UU[c_, r___] /; FreeQ[c, AlgebraAtom] := Expand[c UU[r]];
UU[] = U[];
UU[l_Plus, r___] := UU[#, r] & /@ l;
UU[l_, r___] := UU[Expand[l], r];
```

QLImplementation

```
O[poly_, specs___] := Module[{vs, us, z},
  vs = Join @@ (First /@ {specs});
  us = Join @@ ({specs} /. (l_ → s_) :> (l /. x_i_ :> x_s));
  Simp@Total[CoefficientRules[Normal@Series[poly, {ħ, 0, $TD}], vs] /.
      (p_ → c_) :> c UU @@ (us^p)]]
```

```
$TϵD = 5; B[U@x, U@y] // Simp
```

$$\left(\frac{1}{\hbar}-\frac{T}{\hbar}\right)U[\,]+2\,T\epsilon\,U[a]-2\,T\epsilon^2\,\hbar\,U[a,a]+\left(\gamma\epsilon\hbar+\frac{1}{2}\gamma^2\epsilon^2\hbar^2+\frac{1}{6}\gamma^3\epsilon^3\hbar^3+\frac{1}{24}\gamma^4\epsilon^4\hbar^4+\frac{1}{120}\gamma^5\epsilon^5\hbar^5\right)U[y,x]+\frac{4}{3}T\epsilon^3\hbar^2\,U[a,a,a]-\frac{2}{3}T\epsilon^4\hbar^3\,U[a,a,a,a]+\frac{4}{15}T\epsilon^5\hbar^4\,U[a,a,a,a,a]$$

```
$TϵD = 2; z1 = U[y, y, a, a, x, x]; z2 = U[y, a, x]; z3 = U[y, y, a, x];
z1 ** (z2 ** z3) - (z1 ** z2) ** z3 // Simp
```

```
0
```