

Pensieve header: This is a pruned version of SL2PortfolioProgram.nb and SL2PortfolioTesting.nb from pensieve://Projects/SL2Portfolio/.

SL2Portfolio

Initialization / Utilities

SL2Portfolio

```
In[ ]:=
$p = 2; $k = 1; $U = QU; $E := {$k, $p};
$trim := {h^p_ /; p > $p -> 0, e^k_ /; k > $k -> 0};
SetAttributes[SS, HoldAll];
q_h = e^y e^h;
T2t = {T_{i-}^p_ -> e^{p h t_i}, T_{i-}^p_ -> e^{p h t}}; (* "T to lower t" *)
t2T = {e^{c_- t_i + b_-} -> T_{i-}^{c/h} e^b, e^{c_- t + b_-} -> T^{c/h} e^b, e^e_- -> e^{Expand@e}}; (* "t to upper T" *)
SS[e_-, op_] := Collect[
  Normal@Series[If[$p > 0, e, e /. T2t], {h, 0, $p}],
  h, op];
SS[e_] := SS[e, Together];
Simp[e_-, op_] := Collect[e, _CU | _QU, op];
Simp[e_] := Simp[e, SS[#, Expand] &];
Kd /: Kd_{i,j} := If[i == j, 1, 0];
c_Integer_{k_Integer} := c + 0[e]^{k+1};
```

SL2Portfolio

```
In[ ]:=
CF[e_] := ExpandDenominator@
  ExpandNumerator@Together[Expand[e] /. e^x_ e^y_ -> e^{x+y} /. e^x_ -> e^{CF[x]}];
```

SL2Portfolio

```
In[ ]:=
Unprotect[SeriesData];
SeriesData /: CF[sd_SeriesData] := MapAt[CF, sd, 3];
SeriesData /: Expand[sd_SeriesData] := MapAt[Expand, sd, 3];
SeriesData /: Simplify[sd_SeriesData] := MapAt[Simplify, sd, 3];
SeriesData /: Together[sd_SeriesData] := MapAt[Together, sd, 3];
SeriesData /: Collect[sd_SeriesData, specs__] := MapAt[Collect[#, specs] &, sd, 3];
Protect[SeriesData];
```

SL2Portfolio

DeclareAlgebra

SL2Portfolio

```
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;
B[x_, y_, e_] := B[x, y, e] = B[x, y];
```

SL2Portfolio

In[]:=

```

DeclareAlgebra[U_Symbol, opts__Rule] := Module[{gp, sr, g, cp, M, CE, k = 0,
  gs = Generators /. {opts},
  cs = Centrals /. {opts} /. Centrals -> {}},
  (#U = U@#) & /@gs;
  gp = Alternatives @@ gs; gp = gp | gp; (* gens *)
  sr = Flatten@Table[{g -> ++k, gi -> {i, k}}, {g, gs}]; (* sorting -> *)
  cp = Alternatives @@ cs; (* cents *)
  SetAttributes[M, HoldRest]; M[0, _] = 0; M[a_, x_] := a x;
  CE[_] := Collect[_U, Expand] /. $trim;
  Ui[_] := _ /. {t : cp -> ti, u_U -> (#i &) /@u};
  Ui[NCM[]] = U@{} = 1U = U[];
  B[U@(x_)i_, U@(y_)i_] := Ui@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_. 1U) := CE[c x]; (c_. 1U) ** x_ := CE[c x];
  (a_. U[xx___, x_]) ** (b_. U[yy___]) := If[OrderedQ[{x, y} /. sr],
    CE@M[a b /. $trim, U[xx, x, y, yy]],
    U@xx ** CE@M[a b /. $trim, U@y ** U@x + B[U@x, U@y, $E]] ** 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[_NonCommutativeMultiply] := U /@ _;
  OU[specs___, poly_] := Module[{sp, null, vs, us},
    sp = Replace[{specs}, L_List -> Lnull, {1}];
    vs = Join@@(First /@ sp);
    us = Join@@(sp /. L_s_ -> (L /. x_i_ -> xs));
    CE[Total[
      CoefficientRules[poly, vs] /. (p_ -> c_) -> c U@(us^p)
    ] /; x_nnull -> x];
  OU[specs___, E[L_, Q_, P_]] := OU[specs, SS@Normal[P e^{L+Q}]];
  sigma_rs[c_. * u_U] := (c /. (t : cp)_j -> t_j /. {rs}) U[List@@(u /. v_j_ -> v_j /. {rs})];
  m_j_to_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]);
  U /: c_. * u_U * v_U := CE[c u ** v];
  Si[c_. * u_U] := CE[(((c /. Si[U, Centrals]) DeleteCases[u, _i]) **
    Ui[NCM@@Reverse@Cases[u, x_i -> S@U@x])];
  Delta_i_to_j_k[c_. * u_U] := CE[(((c /. Delta_i_to_j_k[U, Centrals]) DeleteCases[u, _i]) **
    (NCM@@Cases[u, x_i -> sigma_1_to_j_2_to_k@Delta@U@x] /. NCM[] -> U[])];

```

SL2Portfolio

DeclareMorphism

SL2Portfolio

```
In[*]:= DeclareMorphism[m_, U_ -> V_, ongs_List, oncs_List: {}] := (
  Replace[ongs,
    {(g_ -> img_) -> (m[U[g]] = img), (g_ -> img_) -> (m[U[g]] := img /. $trim)}, {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]] /. $trim;
```

SL2Portfolio

Meta-Operations

SL2Portfolio

```
In[*]:=  $\sigma_{rs\_}$ [_E_Plus] :=  $\sigma_{rs}$  /@ _E;
  m_{j -> j_} = Identity; m_{j -> k_}[0] = 0;
  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];
   $\Delta_{is\_}$ [_E_Plus] := Simp[ $\Delta_{is}$  /@ _E];
```

SL2Portfolio

Implementing CU = $\mathcal{U}(\mathfrak{sl}_2^{\mathbb{C}})$

SL2Portfolio

```
In[*]:= DeclareAlgebra[CU, Generators -> {y, a, x}, Centrals -> {t}];
  B[a_CU, y_CU] = -y_CU; B[x_CU, a_CU] = -x_CU;
  B[x_CU, y_CU] = 2 e a_CU - t 1_CU;
  (S@y_CU = -y_CU; S@a_CU = -a_CU; S@x_CU = -x_CU);
  S_i[CU, Centrals] = {t_i -> -t_i};
   $\Delta$ @y_CU = CU@y_1 + CU@y_2;  $\Delta$ @a_CU = CU@a_1 + CU@a_2;  $\Delta$ @x_CU = CU@x_1 + CU@x_2;
   $\Delta_{i -> j, k}$ [CU, Centrals] = {t_i -> t_j + t_k};
```

SL2Portfolio

Implementing QU = $\mathcal{U}_q(\mathfrak{sl}_2^{\mathbb{C}})$

SL2Portfolio

```
In[*]:= DeclareAlgebra[QU, Generators -> {y, a, x}, Centrals -> {t, T}];
  B[a_QU, y_QU] = -y_QU; B[x_QU, a_QU] = -y_QU @ x;
  B[x_QU, y_QU] := SS[q_n - 1] QU@{y, x} + O_QU[{a}, SS[(1 - T e^{-2 e a h}) / h]];
  (S@y_QU := O_QU[{a, y}, SS[-T^{-1} e^{h e a} y]]; S@a_QU = -a_QU; S@x_QU := O_QU[{a, x}, SS[-e^{h e a} x]]);
  S_i[QU, Centrals] = {t_i -> -t_i, T_i -> T_i^{-1}};
   $\Delta$ @y_QU := O_QU[{y_1, a_1}_1, {y_2}_2, SS[y_1 + T_1 e^{-h e a_1} y_2]];
   $\Delta$ @a_QU = QU@a_1 + QU@a_2;  $\Delta$ @x_QU := O_QU[{a_1, x_1}_1, {x_2}_2, SS[x_1 + e^{-h e a_1} x_2]];
   $\Delta_{i -> j, k}$ [QU, Centrals] = {t_i -> t_j + t_k, T_i -> T_j T_k};
```

SL2Portfolio

The representation ρ

SL2Portfolio

```
In[ ]:=
\rho@y_{CU} = \rho@y_{QU} = \begin{pmatrix} \theta & \theta \\ \epsilon & \theta \end{pmatrix}; \rho@a_{CU} = \rho@a_{QU} = \begin{pmatrix} \gamma & \theta \\ \theta & \theta \end{pmatrix};
\rho@x_{CU} = \begin{pmatrix} \theta & \gamma \\ \theta & \theta \end{pmatrix}; \rho@x_{QU} = \begin{pmatrix} \theta & (1 - e^{-\gamma \epsilon \hbar}) / (\epsilon \hbar) \\ \theta & \theta \end{pmatrix};
\rho[e^{\xi}] := MatrixExp[\rho[\xi]];
\rho[\xi_] := (\xi /. T2t /. t \to \gamma \epsilon /. (U : CU | QU) [u___]) \Rightarrow Fold[Dot, \begin{pmatrix} 1 & \theta \\ \theta & 1 \end{pmatrix}, \rho / @ U / @ {u}]
```

SL2Portfolio

tSW

SL2Portfolio

Goal. In either U , compute $F = e^{-\eta y} e^{\xi x} e^{\eta y} e^{-\xi x}$. First compute $G = e^{\xi x} y e^{-\xi x}$, a finite sum. Now F satisfies the ODE $\partial_\eta F = \partial_\eta (e^{-\eta y} e^{\eta G}) = -yF + FG$ with initial conditions $F(\eta = 0) = 1$. So we set it up and solve:

SL2Portfolio

```
In[ ]:=
SW_{xy}[U_, kk_] :=
SW_{xy}[U, kk] = Block[{$U = U, $k = kk, $p = kk}, Module[{G, F, fs, f, bs, e, b, es},
G = Simp[Table[\xi^k / k!, {k, 0, $k + 1}].NestList[Simp[B[x_U, #]] &, y_U, $k + 1]];
fs = Flatten@Table[f_{1,i,j,k}[\eta], {1, 0, $k}, {i, 0, 1}, {j, 0, 1}, {k, 0, 1}];
F = fs.(bs = fs /. f_{l_,i_,j_,k_}[\eta] \Rightarrow e^{l U} @ {y^i, a^j, x^k});
es = Flatten[
Table[Coefficient[e, b] == 0, {e, {F - 1_U /. \eta \to 0, F ** G - y_U ** F - \partial_\eta F}}, {b, bs}]];
F = F /. DSolve[es, fs, \eta][[1]];
E[0,
\xi x + \eta y + (U /. {CU \to -t \eta \xi, QU \to \eta \xi (1 - T) / \hbar}),
F + \theta_{\$k} /. {e \to 1, U \to Times}
] /. (v : \eta | \xi | t | T | y | a | x) \to v_1
]];
tSW_{xy,i_,j_ \to k_} := SW_{xy}[$U, $k] /. {\xi_1 \to \xi_i, \eta_1 \to \eta_j, (v : t | T | y | a | x)_1 \to v_k};
tSW_{xa,i_,j_ \to k_} := E[\alpha_j a_k, e^{-\gamma \alpha_j} \xi_i x_k, 1];
tSW_{ay,i_,j_ \to k_} := E[\alpha_i a_k, e^{-\gamma \alpha_i} \eta_j y_k, 1];
```

SL2Portfolio

Exponentials as needed.

SL2Portfolio

Task. Define $\text{Exp}_{U_i,k}[\xi, P]$ which computes $e^{\xi \mathcal{O}(P)}$ to ϵ^k in the algebra U_i , where ξ is a scalar, X is x_i or y_i , and P is an ϵ -dependent near-docile element, giving the answer in \mathbb{E} -form. Should satisfy $U @ \text{Exp}_{U_i,k}[\xi, P] == \mathbb{S}_U[e^{\xi X}, x \rightarrow \mathcal{O}(P)]$.

Methodology. If $P_0 := P_{\epsilon=0}$ and $e^{\xi \mathcal{O}(P)} = \mathcal{O}(e^{\xi P_0} F(\xi))$, then $F(\xi = 0) = 1$ and we have:
 $\mathcal{O}(e^{\xi P_0} (P_0 F(\xi) + \partial_\xi F)) = \mathcal{O}(\partial_\xi e^{\xi P_0} F(\xi)) = \partial_\xi \mathcal{O}(e^{\xi P_0} F(\xi)) = \partial_\xi e^{\xi \mathcal{O}(P)} = e^{\xi \mathcal{O}(P)} \mathcal{O}(P) = \mathcal{O}(e^{\xi P_0} F(\xi)) \mathcal{O}(P)$.
 This is an ODE for F . Setting inductively $F_k = F_{k-1} + \epsilon^k \varphi$ we find that $F_0 = 1$ and solve for φ .

SL2Portfolio

```
In[ ]:= (* Bug: The first line is valid only if  $0(e^{P_0}) == e^0(P_0)$ . *)
(* Bug:  $\xi$  must be a symbol. *)
Exp_{u_i, \theta}[\xi_, P_] := Module[{LQ = Normal@P /. \epsilon \to \theta},
  E[\xi LQ /. (x | y)_i \to \theta, \xi LQ /. (t | a)_i \to \theta, 1]];
Exp_{u_i, k}[\xi_, P_] := Block[{$U = U, $k = k},
  Module[{P0 = \varphi, \varphi_s, F, j, rhs, at0, at\xi},
    P0 = Normal@P /. \epsilon \to \theta;
    \varphi_s =
      Flatten@Table[\varphi_{j1, j2, j3}[\xi], {j2, 0, k}, {j1, 0, 2k + 1 - j2}, {j3, 0, 2k + 1 - j2 - j1}];
    F = Normal@Last@Exp_{u_i, k-1}[\xi, P] + \epsilon^k \varphi_s. (\varphi_s /. \varphi_{js}[\xi] \Rightarrow Times @@ {y_i, a_i, x_i}^{\{js\}});
    rhs = Normal@
      Last@m_{i, j \to i} [E[\xi P0 /. (x | y)_i \to \theta, \xi P0 /. (t | a)_i \to \theta, F + \theta_k] m_{i \to j} @ E[\theta, \theta, P + \theta_k]];
    at0 = (# == 0) & /@ Flatten@CoefficientList[F - 1 /. \xi \to \theta, {y_i, a_i, x_i}];
    at\xi = (# == 0) & /@ Flatten@CoefficientList[(\partial_\xi F) + P0 F - rhs, {y_i, a_i, x_i}];
    E[\xi P0 /. (x | y)_i \to \theta, \xi P0 /. (t | a)_i \to \theta, F + \theta_k] /.
      DSolve[And @@ (at0 \cup at\xi), \varphi_s, \xi][[1]]]
```

SL2Portfolio

Zip and Bind

SL2Portfolio

```
In[ ]:= E /: E[L1_, Q1_, P1_] \equiv E[L2_, Q2_, P2_] :=
  CF[L1 == L2] \wedge CF[Q1 == Q2] \wedge CF[Normal[P1 - P2] == 0];
E /: E[L1_, Q1_, P1_] E[L2_, Q2_, P2_] := E[L1 + L2, Q1 + Q2, P1 + P2];
```

SL2Portfolio

```
In[ ]:= {t*, y*, a*, x*, z*} = {\tau, \eta, \alpha, \xi, \zeta};
{\tau*, \eta*, \alpha*, \xi*, \zeta*} = {t, y, a, x, z}; (u_{-i})^* := (u^*)_i;
```

SL2Portfolio

```
In[ ]:= Zip_{\{}}[P_] := P; Zip_{\{\xi, \zeta, \dots\}}[P_] := (Expand[P // Zip_{\{\xi, \zeta\}}] /. f_ . \zeta^{d_} \Rightarrow \partial_{\{\xi^*, d\}} f) /. \zeta^* \to \theta
```

SL2Portfolio

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

SL2Portfolio

```
In[ ]:= QZip_{\xi_s List, simp} @ E[L_, Q_, P_] := Module[{\xi, z, zs, c, ys, \eta_s, qt, zrule, Q1, Q2},
  zs = Table[\xi^*, {\xi, \zeta_s}];
  c = Q /. Alternatives @@ (\zeta_s \cup zs) \to \theta;
  ys = Table[\partial_\xi (Q /. Alternatives @@ zs \to \theta), {\xi, \zeta_s}];
  \eta_s = Table[\partial_z (Q /. Alternatives @@ \zeta_s \to \theta), {z, zs}];
  qt = Inverse@Table[K\delta_{z, \xi^*} - \partial_{z, \xi} Q, {\xi, \zeta_s}, {z, zs}];
  zrule = Thread[zs \to qt.(zs + ys)];
  Q2 = (Q1 = c + \eta_s.zs /. zrule) /. Alternatives @@ zs \to \theta;
  simp /@ E[L, Q2, Det[qt] e^{-Q2} Zip_{\xi_s} [e^{Q1} (P /. zrule)]];
  QZip_{\xi_s List} := QZip_{\xi_s, CF};
```

SL2Portfolio

LZip implements the “L-level zips” on $\mathbb{E}(L, Q, P) = P\theta^{L+Q}$. Such zips regard all of $P\theta^Q$ as a single “P”. Here the z’s are t and α and the ζ ’s are τ and a .

SL2Portfolio

```
In[ ]:= LZip $\zeta$ s_List,simp_@E[L_, Q_, P_] := Module[{ $\zeta$ , z, zs, c, ys,  $\eta$ s, lt, zrule, L1, L2, Q1, Q2},
  zs = Table[ $\zeta^*$ , { $\zeta$ ,  $\zeta$ s}];
  c = L /. Alternatives@@ ( $\zeta$ s  $\cup$  zs)  $\rightarrow$  0;
  ys = Table[ $\partial_{\zeta}$  (L /. Alternatives@@ zs  $\rightarrow$  0), { $\zeta$ ,  $\zeta$ s}];
   $\eta$ s = Table[ $\partial_z$  (L /. Alternatives@@  $\zeta$ s  $\rightarrow$  0), {z, zs}];
  lt = Inverse@Table[K $\delta_{z,\zeta^*}$  -  $\partial_{z,\zeta}$ L, { $\zeta$ ,  $\zeta$ s}, {z, zs}];
  zrule = Thread[zs  $\rightarrow$  lt.(zs + ys)];
  L2 = (L1 = c +  $\eta$ s.zs /. zrule) /. Alternatives@@ zs  $\rightarrow$  0;
  Q2 = (Q1 = Q /. T2t /. zrule) /. Alternatives@@ zs  $\rightarrow$  0;
  simp /@ E[L2, Q2, Det[lt] e-L2-Q2 Zip $\zeta$ s[eL1+Q1 (P /. T2t /. zrule)]] // t2T];
LZip $\zeta$ s_List := LZip $\zeta$ s,CF;
```

SL2Portfolio

```
In[ ]:= Bind_{ } [L_, R_] := L R;
Bind_{is_} [L_IE, R_IE] := Module[{n},
  Times[
    L /. Table[{v : T | t | a | x | y}_i  $\rightarrow$  vn $\epsilon$ i, {i, {is}}}],
    R /. Table[{v :  $\tau$  |  $\alpha$  |  $\zeta$  |  $\eta$ }_i  $\rightarrow$  vn $\epsilon$ i, {i, {is}}]
  ] // LZipFlatten@Table[{ $\tau$ n $\epsilon$ i, an $\epsilon$ i}, {i, {is}}] // QZipFlatten@Table[{ $\zeta$ n $\epsilon$ i, yn $\epsilon$ i}, {i, {is}}] ];
B_L_List := Bind_L; B_is_ := Bind_{is};
Bind[ $\mathcal{E}$ _IE] :=  $\mathcal{E}$ ;
Bind[Ls_ ,  $\zeta$ s_List, R_] := Bind $\zeta$ s [Bind[Ls], R];
```

SL2Portfolio

Tensorial Representations

SL2Portfolio

```
In[ ]:= t $\eta$  = tI = E[0, 0, 1 + 0 $\delta_{kk}$ ];
```

SL2Portfolio

```
In[ ]:= tmi,j $\rightarrow$ k := Module[{tk},
  E[( $\tau_i + \tau_j$ ) tk +  $\alpha_i$  ak +  $\alpha_j$  ak,  $\eta_i$  yk +  $\xi_j$  xk, 1]
  (tSWxy,i,j $\rightarrow$ tk /. {ttk  $\rightarrow$  tk, Ttk  $\rightarrow$  Tk, ytk  $\rightarrow$  e- $\gamma$  $\alpha_i$  yk, atk  $\rightarrow$  ak, xtk  $\rightarrow$  e- $\gamma$  $\alpha_j$  xk}});
mj $\rightarrow$ k [ $\mathcal{E}$ _IE] :=  $\mathcal{E}$  ~ Bj,k ~ tmj,k $\rightarrow$ k;
```

SL2Portfolio

```
In[ ]:= tm1,2 $\rightarrow$ 3
```

SL2Portfolio

$$\text{Out[]} = \mathbb{E} \left[\mathbf{a}_3 \alpha_1 + \mathbf{a}_3 \alpha_2 + \mathbf{t}_3 (\tau_1 + \tau_2), y_3 \eta_1 + e^{-\gamma \alpha_1} y_3 \eta_2 + e^{-\gamma \alpha_2} x_3 \xi_1 + \frac{(1 - T_3) \eta_2 \xi_1}{\hbar} + x_3 \xi_2, \right. \\ \left. 1 + \frac{1}{4 \hbar} \eta_2 \xi_1 (8 \hbar \mathbf{a}_3 T_3 + 4 e^{-\gamma \alpha_1 - \gamma \alpha_2} \gamma \hbar^2 x_3 y_3 + 2 e^{-\gamma \alpha_1} \gamma \hbar y_3 \eta_2 - 6 e^{-\gamma \alpha_1} \gamma \hbar T_3 y_3 \eta_2 + \right. \\ \left. 2 e^{-\gamma \alpha_2} \gamma \hbar x_3 \xi_1 - 6 e^{-\gamma \alpha_2} \gamma \hbar T_3 x_3 \xi_1 + \gamma \eta_2 \xi_1 - 4 \gamma T_3 \eta_2 \xi_1 + 3 \gamma T_3^2 \eta_2 \xi_1) \right] \in + O[\epsilon]^2$$

SL2Portfolio

```
In[ ]:= S[U_, kk_] := S[U, kk] = Module[{OE},
  OE = m_{3,2,1 \to 1} [Exp_{QU_1, $k} [\eta, S_1[QU[y_1]] /. QU \to Times]
    Exp_{QU_2, $k} [\alpha, S_2[QU[a_2]] /. QU \to Times] Exp_{QU_3, $k} [\xi, S_3[QU[x_3]] /. QU \to Times]];
  E[-t_1 \tau_1 + OE[[1]], OE[[2]], OE[[3]]] /. {\eta \to \eta_1, \alpha \to \alpha_1, \xi \to \xi_1};
  tS_i_ := S[$U, $k] /. {(v : \tau | \eta | \alpha | \xi)_1 \to v_i, (v : t | T | y | a | x)_1 \to v_i};
```

SL2Portfolio

```
In[ ]:= tS_1
```

SL2Portfolio

$$\text{Out[]} = E \left[-a_1 \alpha_1 - t_1 \tau_1, \frac{1}{\hbar T_1} \left(-e^{\gamma \alpha_1} \hbar y_1 \eta_1 - e^{\gamma \alpha_1} \hbar T_1 x_1 \xi_1 + e^{\gamma \alpha_1} \eta_1 \xi_1 - e^{\gamma \alpha_1} T_1 \eta_1 \xi_1 \right), \right. \\ \left. 1 + \frac{1}{4 \hbar T_1^2} \left(4 e^{\gamma \alpha_1} \gamma \hbar^2 T_1 y_1 \eta_1 - 4 e^{\gamma \alpha_1} \hbar^2 a_1 T_1 y_1 \eta_1 - 2 e^{2\gamma \alpha_1} \gamma \hbar^2 y_1^2 \eta_1^2 - 4 e^{\gamma \alpha_1} \hbar^2 a_1 T_1^2 x_1 \xi_1 - \right. \right. \\ \left. \left. 4 e^{\gamma \alpha_1} \gamma \hbar T_1 \eta_1 \xi_1 + 8 e^{\gamma \alpha_1} \hbar a_1 T_1 \eta_1 \xi_1 + 4 e^{\gamma \alpha_1} \gamma \hbar T_1^2 \eta_1 \xi_1 - 4 e^{2\gamma \alpha_1} \gamma \hbar^2 T_1 x_1 y_1 \eta_1 \xi_1 + \right. \right. \\ \left. \left. 6 e^{2\gamma \alpha_1} \gamma \hbar y_1 \eta_1^2 \xi_1 - 2 e^{2\gamma \alpha_1} \gamma \hbar T_1 y_1 \eta_1^2 \xi_1 - 2 e^{2\gamma \alpha_1} \gamma \hbar^2 T_1^2 x_1^2 \xi_1^2 + 6 e^{2\gamma \alpha_1} \gamma \hbar T_1 x_1 \eta_1 \xi_1^2 - \right. \right. \\ \left. \left. 2 e^{2\gamma \alpha_1} \gamma \hbar T_1^2 x_1 \eta_1 \xi_1^2 - 3 e^{2\gamma \alpha_1} \gamma \eta_1^2 \xi_1^2 + 4 e^{2\gamma \alpha_1} \gamma T_1 \eta_1^2 \xi_1^2 - e^{2\gamma \alpha_1} \gamma T_1^2 \eta_1^2 \xi_1^2 \right) \epsilon + O[\epsilon]^2 \right]$$

SL2Portfolio

```
In[ ]:= \Delta[U_, kk_] := \Delta[U, kk] = Module[{OE},
  OE = Block[{$k = kk, $p = kk + 1},
    m_{1,3,5 \to 1} @ m_{2,4,6 \to 2} @ Times [ (* Warning: wrong unless $p \ge $k+1! *)
      ReplacePart[1 \to 0] @ Exp_{QU_1, $k} [\eta, \Delta_{1 \to 1,2}[QU[y_1]] /. QU \to Times],
      ReplacePart[2 \to 0] @ Exp_{QU_3, $k} [\alpha, \Delta_{3 \to 3,4}[QU[a_3]] /. QU \to Times],
      ReplacePart[1 \to 0] @ Exp_{QU_5, $k} [\xi, \Delta_{5 \to 5,6}[QU[x_5]] /. QU \to Times]
    ] /. {\eta \to \eta_1, \alpha \to \alpha_1, \xi \to \xi_1};
  E[\tau_1 (t_1 + t_2) + \alpha_1 (a_1 + a_2), OE[[2]], OE[[3]]];
  t\Delta_{i \to j, k_} :=
  \Delta[$U, $k] /. {(v : \tau | \eta | \alpha | \xi)_1 \to v_i, (v : t | T | y | a | x)_1 \to v_j, (v : t | T | y | a | x)_2 \to v_k};
```

SL2Portfolio

```
In[ ]:= t\Delta_{1 \to 1,2}
```

SL2Portfolio

$$\text{Out[]} = E \left[(a_1 + a_2) \alpha_1 + (t_1 + t_2) \tau_1, y_1 \eta_1 + T_1 y_2 \eta_1 + x_1 \xi_1 + x_2 \xi_1, \right. \\ \left. 1 + \frac{1}{2} \left(-2 \hbar a_1 T_1 y_2 \eta_1 + \gamma \hbar T_1 y_1 y_2 \eta_1^2 - 2 \hbar a_1 x_2 \xi_1 + \gamma \hbar x_1 x_2 \xi_1^2 \right) \epsilon + O[\epsilon]^2 \right]$$

SL2Portfolio

The Faddeev-Quesne formula:

SL2Portfolio

```
In[ ]:= e_{q, k} [x_] := e ^ \left( \sum_{j=1}^{k+1} \frac{(1-q)^j x^j}{j (1-q^j)} \right); e_{q, $k} [x]
```

SL2Portfolio

```
In[*]:= R[QU, kk_] := R[QU, kk] = E[-(h a2 t1)/gamma, h x2 y1,
Series[e^{h gamma^{-1} t1 a2 - h y1 x2} (e^{h b1 a2} e_{qn, kk}[h y1 x2] /. b1 -> gamma^{-1} (epsilon a1 - t1)), {epsilon, 0, kk}]]];
tR_{i,j}_ := R[$U, $k] /. {(v:t|T|y|a|x)_1 -> v_i, (v:t|T|y|a|x)_2 -> v_j};
tR_{i,j}_ := tR_{i,j} = tR_{i,j} ~ B_j ~ tS_j;
```

SL2Portfolio

```
In[*]:= {tR_{1,2}, tR_{1,2}}
```

SL2Portfolio

```
Out[*]:= {E[-(h a2 t1)/gamma, h x2 y1, 1 + ((h a1 a2)/gamma - 1/4 gamma h^3 x2^2 y1^2) epsilon + O[epsilon]^2], E[(h a2 t1)/gamma, -h x2 y1/T1,
1 + 1/(4 gamma T1^2) (-4 h a1 a2 T1^2 - 4 gamma h^2 a1 T1 x2 y1 - 4 gamma h^2 a2 T1 x2 y1 - 3 gamma^2 h^3 x2^2 y1^2) epsilon + O[epsilon]^2]}
```

SL2Portfolio

tC is the counterclockwise spinner; tC is its inverse.

SL2Portfolio

```
In[*]:= tC_{i_} := E[{0, 0, T1^{1/2} e^{-epsilon a_i h} + 0 $k}];
tC_{i_} := E[{0, 0, T1^{-1/2} e^{epsilon a_i h} + 0 $k}];
```

SL2Portfolio

```
In[*]:= Block[{$k = 3}, {tC_1, tC_2}]
```

SL2Portfolio

```
Out[*]:= {E[{0, 0, sqrt(T1) - h a1 sqrt(T1) epsilon + 1/2 h^2 a1^2 sqrt(T1) epsilon^2 - 1/6 (h^3 a1^3 sqrt(T1)) epsilon^3 + O[epsilon]^4],
E[{0, 0, 1/sqrt(T2) + h a2 epsilon/sqrt(T2) + h^2 a2^2 epsilon^2/(2 sqrt(T2)) + h^3 a2^3 epsilon^3/(6 sqrt(T2)) + O[epsilon]^4}]}
```

SL2Portfolio

```
In[*]:= Kink[QU, kk_] := Kink[QU, kk] = Block[{$k = kk}, (tR_{1,3} tC_2) ~ B_{1,2} ~ tm_{1,2+1} ~ B_{1,3} ~ tm_{1,3+1}];
tKink_{i_} := Kink[$U, $k] /. {(v:t|T|y|a|x)_1 -> v_i};
Kink[QU, kk_] := Kink[QU, kk] = Block[{$k = kk}, (tR_{1,3} tC_2) ~ B_{1,2} ~ tm_{1,2+1} ~ B_{1,3} ~ tm_{1,3+1}];
tKink_{i_} := Kink[$U, $k] /. {(v:t|T|y|a|x)_1 -> v_i}
```

SL2Portfolio

Alternative Algorithms

SL2Portfolio

```
In[*]:= lambda_{alt,k}_[CU] := If[k == 0, 1, Module[{eq, d, b, c, so},
eq = rho @ e^{epsilon x cu} . rho @ e^{eta y cu} == rho @ e^{d y cu} . rho @ e^{c (t1 cu - 2 epsilon a cu)} . rho @ e^{b x cu};
{so} = Solve[Thread[Flatten[eq], {d, b, c}]] /. C @ 1 -> 0;
Series[e^{-eta y - epsilon x + eta epsilon t + c t + d y - 2 epsilon c a + b x} /. so, {epsilon, 0, k}]]];
```


SL2Portfolio

The Trefoil

```
In[ ]:= Block[{$k = 1},
  Z = tR1,5 tR6,2 tR3,7 tC4 tKink8 tKink9 tKink10;
  Do[Echo@{$k, k}; Z = Z ~ B1,k ~ tm1,k→1, {k, 2, 10}];
  Z]
" {1, 2}
" {1, 3}
" {1, 4}
" {1, 5}
" {1, 6}
" {1, 7}
" {1, 8}
```

SL2Portfolio

```
In[ ]:= Block[{$k = 1},
  Z = tR1,5 tR6,2 tR3,7 tC4 tKink8 tKink9 tKink10;
  Do[Z = Z ~ B1,k ~ tm1,k→1, {k, 2, 10}]; Z]
```

SL2Portfolio

$$\text{Out[]} = \mathbb{E} \left[\theta, \theta, \frac{T_1}{1 - T_1 + T_1^2} + \left((-2 \hbar a_1 T_1 - \gamma \hbar T_1^2 + 2 \hbar a_1 T_1^2 + 2 \gamma \hbar T_1^3 - 3 \gamma \hbar T_1^4 - 2 \hbar a_1 T_1^4 + 2 \gamma \hbar T_1^5 + 2 \hbar a_1 T_1^5 - 2 \gamma \hbar^2 T_1 x_1 y_1 - 2 \gamma \hbar^2 T_1^4 x_1 y_1) \epsilon \right) / (1 - 3 T_1 + 6 T_1^2 - 7 T_1^3 + 6 T_1^4 - 3 T_1^5 + T_1^6) + \mathcal{O}[\epsilon]^2 \right]$$