Solving linearized 5-gon in emergent $\mathcal{P}$

In[◦]:=
```
SetDirectory["C:\\drorbn\\AcademicPensieve\\People\\Kuno"];
<< FreeLie.m
```

```
FreeLie` implements / extends
 {*, +, **, $SeriesShowDegree, ⟨⟩, ∫, ≡, ad, Ad, adSeries, AllCyclicWords, AllLyndonWords,
  AllWords, Arbitrator, AS, ASeries, AW, b, BCH, BooleanSequence, BracketForm, BS, CC, Crop,
  cw, CW, CWS, CWSeries, D, Deg, DegreeScale, DerivationSeries, div, DK, DKS, DKSeries, EulerE,
  Exp, FreeLieFormatting, Inverse, j, J, JA, LieDerivation, LieMorphism, LieSeries, LS, LW,
  LyndonFactorization, Morphism, New, RandomCWSeries, Randomizer, RandomLieSeries, RC, SeriesSolve,
  Support, t, tb, TopBracketForm, tr, UndeterminedCoefficients, αMap, Γ, ι, Λ, σ, τ, ℏ, ⌐, ⌢}.
```

```
FreeLie` is in the public domain. Dror Bar-Natan is committed
  to support it within reason until July 15, 2022. This is version 240218.
```

Extending FreeLie.m

1. The antipode

In[◦]:=
```
Antipode[ℰ_] := Expand[ℰ /. w_AW :> (-1)^Length[w] Reverse[w]]
```

The map R

1. The map R: FL --> FA (symmetric - horizontal)

In[ ]:=
```
R[_, 0] = 0;
R[A_, u_LW] := R[A, u] = Module[{w1, w2},
    If[Deg[u] === 1, 0,
     {w1, w2} = LyndonFactorization[u];
     Expand[
      b[ι[w1], R[A, w2]] + b[R[A, w1], ι[w2]]
       + Sum[(τ[LW[α], w2] ** AW[α] ** Antipode[τ[LW[α], w1]] -
           τ[LW[α], w1] ** AW[α] ** Antipode[τ[LW[α], w2]]) / 2,
         {α, A}]
      ]
     ]
    ];
R_A_[ℰ_] := Expand[ℰ /. LW[seq__] :→ R[A, LW[seq]]];
R_A_[ls_LieSeries] := R_A[ls] = New[ASeries[as],
    as[d_] := as[d] = R_A[ls[d + 1]]
    ];
```

In[ ]:=
```
R_{x,y}[LW[x, x, x, y]]
```

Out[ ]=
```
-AW[x, x, y] + AW[y, x, x]
```

In[ ]:=
```
R_{x,y,z}[b[b[LW[x], LW[y]], b[LW[x], LW[z]]]]
```

Out[ ]=

$$\frac{1}{2}\,\mathrm{AW}[y, x, z] - \frac{1}{2}\,\mathrm{AW}[z, x, y]$$

In[ ]:=
```
P3[ψ_LieSeries] := Module[{pd},
   pd = τ[LW[y], ψ];
   pd + LieMorphism[{LW[x] → LW[y], LW[y] → 0}]@pd -
    LieMorphism[{LW[x] → LW[x] + LW[y], LW[y] → 0}]@pd - 2 R_{x,y}[ψ]
  ]
```

In[ ]:=
```
ψ = LS[{LW@x, LW@y}, ψs]
```

Out[ ]=

$$\mathrm{LS}\left[\overline{x}\,\psi s[x] + \overline{y}\,\psi s[y],\ \overline{xy}\,\psi s[x, y],\ \overline{x\ \overline{xy}}\,\psi s[x, x, y] + \overline{\overline{xy}\,y}\,\psi s[x, y, y],\ \ldots\right]$$

In[ ]:=
```
pd = τ[LW[y], ψ]
```

Out[ ]=
```
AS[AW[] ψs[y], AW[x] ψs[x, y],
 AW[x, x] ψs[x, x, y] + AW[x, y] ψs[x, y, y] - 2 AW[y, x] ψs[x, y, y],
 AW[x, x, x] ψs[x, x, x, y] + AW[x, x, y] ψs[x, x, y, y] - 2 AW[x, y, x] ψs[x, x, y, y] +
  AW[x, y, y] ψs[x, y, y, y] - 3 AW[y, x, y] ψs[x, y, y, y] + 3 AW[y, y, x] ψs[x, y, y, y]]
```

In[ ]:=
```
LieMorphism[{LW[x] → LW[y], LW[y] → LW[x]}][AW[y, x, y]]
```

Out[ ]=
```
AS[0, 0, 0, AW[x, y, x]]
```

In[∘]:= **LieMorphism[{LW[x] → LW[y], LW[y] → 0}]@pd**

Out[∘]=

```
AS[0, AW[y] ψs[x, y], AW[y, y] ψs[x, x, y], AW[y, y, y] ψs[x, x, x, y]]
```

In[∘]:= **ψ**

Out[∘]=

```
LS[x̄ ψs[x] + ȳ ψs[y], x̄ȳ ψs[x, y], x̄ x̄ȳ ψs[x, x, y] + x̄ȳ y ψs[x, y, y], ...]
```

In[∘]:= **P3[ψ]**

Out[∘]=

```
AS[AW[] ψs[y], 0, 2 AW[x, y] ψs[x, y, y] - AW[y, x] ψs[x, y, y],
 AW[x, x, y] ψs[x, x, x, y] - AW[x, y, x] ψs[x, x, x, y] - AW[x, y, y] ψs[x, x, x, y] -
  3 AW[y, x, x] ψs[x, x, x, y] - AW[y, x, y] ψs[x, x, x, y] - AW[y, y, x] ψs[x, x, x, y] +
  2 AW[x, x, y] ψs[x, x, y, y] - 2 AW[x, y, x] ψs[x, x, y, y] +
  AW[x, y, y] ψs[x, x, y, y] - AW[y, x, x] ψs[x, x, y, y] - AW[y, y, x] ψs[x, x, y, y] +
  3 AW[x, y, y] ψs[x, y, y, y] - 3 AW[y, x, y] ψs[x, y, y, y] + AW[y, y, x] ψs[x, y, y, y]]
```

In[∘]:= **AS[0]**

Out[∘]=

```
AS[0, 0, 0, 0]
```

In[∘]:= **sol = SeriesSolve[{ψ}, ℏ (P3[ψ] ≡ AS[0]), Arbitrator → 0]**

Out[∘]=

```
FreeLie`Private`MessageStream$4248
```

In[∘]:= **Timing[ψ@{5}]**

SeriesSolve: In degree 1 arbitrarily setting {ψs[x] → 0, ψs[y] → 0}.

SeriesSolve: In degree 2 arbitrarily setting {ψs[x, y] → 0}.

SeriesSolve: In degree 3 arbitrarily setting {ψs[x, x, y] → 0}.

General: Further output of SeriesSolve::ArbitrarilySetting will be suppressed during this calculation.

Out[∘]=

```
{0.265625, LS[0, 0, 0, 0, 0, ...]}
```

In[∘]:= **Read[sol] // Column**

Out[∘]=

```
{ArbitrarilySetting, 1, {Hold[ψs[x]] → 0, Hold[ψs[y]] → 0}}
{ArbitrarilySetting, 2, {Hold[ψs[x, y]] → 0}}
{ArbitrarilySetting, 3, {Hold[ψs[x, x, y]] → 0}}
{ArbitrarilySetting, 4, {}}
{ArbitrarilySetting, 5, {Hold[ψs[x, x, x, x, y]] → 0}}
```

In[∘]:= **P3[ψ][5]**

Out[∘]=

```
0
```

In[ ]:= ψ[5]

Out[ ]=

0