μ calculus clippings

July-30-12
8:51 PM

Swap $\mu \longleftrightarrow \lambda$!

Pensieve header: ~~λ~~ μ-calculus clippings - not to be executed.

```
hta[x_, y_, z_][λ[ω_, μ_]] := Module[{μx, Adμx},
   μx = MakeLieSeries[D[μ, h[x]]];
   Adμx =
    LieMorphism[{LW[y] → Ad[ScaleLieSeries[-1, μx]][LW[z]]}];
   λ[
    AddCWSeries[StableApply[Adμx, ω], J[LW[y], μx]],
    Collect[μ, _h, StableApply[Adμx, #] &]
   ]
  ];
hta[x_, y_][λ[ω_, μ_]] :=
  λ[ω, μ] // hta[x, y, ⟨"z"⟩] // LieMorphism[{LW["z"] → LW[y]}];
dm[x_, y_, z_][λ_] := λ // hta[y, x] // tm[x, y, z] // hm[x, y, z];

Ad[x_] := adSeries[E^(-ad), x];

J[-1, ___] = MakeCWSeries[0];
J[n_, y_LW, μ_LieSeries, s_] := J[n, y, μ, s] = Module[
    {sμ, μs},
    sμ = ScaleLieSeries[s, μ];
    μs = StableApply[
      LieMorphism[{y → Ad[ScaleLieSeries[-1, sμ]][LW[z]]}], μ];
    μs = μs // LieMorphism[{LW[z] → y}];
    IntegrateCWSeries[
     AddCWSeries[
      J[n - 1, y, μ, s] // LieDerivation[{y → b[μs, y]}],
      div[y, μs]
     ],
     {s, 0, s}
    ]
   ];
J[y_LW, μ_LieSeries] := J[y, μ] = Module[{cws, s},
    cws = Unique[J];
    cws[] = Hold[J[y, μ]];
    cws[d_Integer] := cws[d] = J[d - 1, y, μ, s][d] /. s → 1;
    CWSeries[cws]
   ];
```

$\mu_s = \mu // CC^{s\mu}_u$

needs a name! CC^{μ}_u ?

$$\mathrm{hta}^{xu}(\underset{(\omega,\mu)}{\lambda}) = \underset{(\omega,\mu)}{\lambda} \; //. \; \overbrace{u \mapsto e^{\mathrm{ad}\,\mu_x}(\bar{u}) \; /. \; \bar{u} \to u}$$

$+ (J_u(\mu_x), 0)$ "the spice"

what's $J_u(\mu)$? It's $J'_u(\mu)$ where

$$\mu_s = CC^{s\mu}_u(\mu)$$

$$J^0_u(\mu) = \mathrm{div}_u \mu$$

and $$\frac{d}{ds} J^s_u(\mu) = J^s_u // \mathrm{der}(u \mapsto [\mu_s, u]) + \mathrm{div}_u \mu_s$$

$$\frac{d}{ds} j^{xy}(s\mu) =$$

In older language:

$$= \underbrace{j^{xy}(s\mu) // \mathrm{der}(y \mapsto [\mu^s_x, y])}_{\text{First long}} + \underbrace{(\mathrm{div}_y \mu^s_x)}_{\text{then short}}$$

where $\mu^s_x := \mu_x //_0\ y \mapsto e^{s\mu_x} \bar{y} e^{-s\mu_x}$