In[22]:=

```
(* ============================================== *)
(* 3D Wormhole + 1D Brans-Dicke + EM Vortex + Matter *)
(* ============================================== *)

ClearAll["Global`*"];



(* ----------------------- *)
(* 1. Wormhole geometry      *)
(* ----------------------- *)

(* Wormhole scalar potential (your shape) *)
R0 = 1.0*^-3; (* throat radius: 1 mm *)
A  = 1.0;   (* amplitude scale      *)
w  = 10 R0;  (* Gaussian width       *)

ΦW[r_] := -A (1 - R0/r) Exp[-(r - R0)^2/w^2];

(* Metric ansatz (conformally flat spatial slices) *)
(* ds^2 = -e^{2Φ} dt^2 + e^{-2Φ} (dr^2 + r^2 dΩ^2) *)
f[r_]   := Exp[2 ΦW[r]];
finv[r_] := Exp[-2 ΦW[r]];

(* ----------------------- *)
(* 2. Brans-Dicke scalar     *)
(* ----------------------- *)

ωBD = 100.0; (* Brans-Dicke parameter *)

$Assumptions = r > 0;

BDop[f_] := Simplify[
 D[f[r], {r, 2}] + (2/r) D[f[r], r]
];

(* BD residual for wormhole scalar *)
bdResidual[r_] = Simplify[BDop[ΦW]];

(* Required trace T_req from BD scalar equation:
   Φ'' + 2 Φ'/r = (8 π / (3+2 ω)) T   *)
TtraceRequired[r_] := (3 + 2 ωBD) bdResidual[r]/(8 Pi);
```

```
(* ----------------------- *)
(* 3. EM double-vortex field *)
(*    (flat-space Maxwell)  *)
(* ----------------------- *)

(* Geometry of EM vortices in xy-plane *)
dEM  = 0.06;   (* separation along x [m] *)
R0EM  = 0.03;   (* vortex radius [m]      *)
ωEM  = 2 Pi 5.0;  (* angular frequency      *)

(* Paper-matched parameters (from your Python) *)
paperPars = {
 3.162*^-5,    (* A1 *)
 3.162*^-5,    (* A2 *)
 N[Pi],      (* φ *)
 0.0001,     (* mix1 *)
 0.0001,     (* mix2 *)
 0.00139359,   (* epsY *)
 0.0001,     (* chir1 *)
 -0.0001    (* chir2 *)
};

r2xy[x_, y_, cx_, cy_] := (x - cx)^2 + (y - cy)^2;
rxy[x_, y_, cx_, cy_] := Sqrt[r2xy[x, y, cx, cy]];
envEM[x_, y_, cx_, cy_] := Exp[-r2xy[x, y, cx, cy]/R0EM^2];

eOne[x_, y_, t_, cx_, cy_, Aamp_, phase_, mix_] :=
 Module[{rx = x - cx, ry = y - cy, rr, g, er, et},
  rr = Sqrt[rx^2 + ry^2];
  If[rr == 0., Return[{0., 0.}]];
  g = envEM[x, y, cx, cy];
  er = {rx, ry}/rr;
  et = {-ry, rx}/rr;
  Aamp g Cos[ωEM t + phase] ((1 - mix) er + mix et)
 ];

bOne[x_, y_, t_, cx_, cy_, Aamp_, phase_, chir_] :=
 Module[{rx = x - cx, ry = y - cy, rr, g, et},
  rr = Sqrt[rx^2 + ry^2];
  If[rr == 0., Return[{0., 0.}]];
  g = envEM[x, y, cx, cy];
```

```
 et = {-ry, rx}/rr;
 Aamp g Sin[ωEM t + phase] chir et
 ];

eField[x_, y_, t_, pars_List : paperPars] :=
 Module[{A1, A2, ϕ, mix1, mix2, epsY, chir1, chir2, c1, c2},
 {A1, A2, ϕ, mix1, mix2, epsY, chir1, chir2} = pars;
 c1 = {-dEM/2., 0.};
 c2 = { dEM/2., epsY};
 eOne[x, y, t, c1⟦1⟧, c1⟦2⟧, A1, 0., mix1] +
 eOne[x, y, t, c2⟦1⟧, c2⟦2⟧, A2, ϕ, mix2]
 ];

bField[x_, y_, t_, pars_List : paperPars] :=
 Module[{A1, A2, ϕ, mix1, mix2, epsY, chir1, chir2, c1, c2},
 {A1, A2, ϕ, mix1, mix2, epsY, chir1, chir2} = pars;
 c1 = {-dEM/2., 0.};
 c2 = { dEM/2., epsY};
 bOne[x, y, t, c1⟦1⟧, c1⟦2⟧, A1, 0., chir1] +
 bOne[x, y, t, c2⟦1⟧, c2⟦2⟧, A2, ϕ, chir2]
 ];

(* Maxwell stress-energy in flat space, Gaussian units, signature (-,+,+,+) *)
stressEnergyEM[x_, y_, t_, pars_List : paperPars] :=
 Module[{E, B, E2, B2, rho, S, sigma, T},
 E = Append[eField[x, y, t, pars], 0.]; (* {Ex,Ey,Ez} *)
 B = Append[bField[x, y, t, pars], 0.]; (* {Bx,By,Bz} *)
 E2 = E.E;
 B2 = B.B;
 rho = (E2 + B2)/(8. Pi);
 S = Cross[E, B]/(4. Pi);
 sigma = ConstantArray[0., {3, 3}];
 Module[{i, j, prefid = (E2 + B2)/(8. Pi)},
  Do[
   sigma⟦i, j⟧ =
    prefid If[i == j, 1., 0.] - (E⟦i⟧ E⟦j⟧ + B⟦i⟧ B⟦j⟧)/(4. Pi)
   , {i, 1, 3}, {j, 1, 3}]
  ];
 T = ConstantArray[0., {4, 4}];
```

```
 T[[1, 1]] = rho;
 T[[1, 2 ;; 4]] = S;
 T[[2 ;; 4, 1]] = S;
 T[[2 ;; 4, 2 ;; 4]] = sigma;
 T
 ];

(* Trace of EM stress-energy: T^μ_μ = -rho + p1 + p2 + p3 *)
TtraceEM[x_, y_, t_, pars_List : paperPars] :=
 Module[{T, rho, spatial, evals},
 T = stressEnergyEM[x, y, t, pars];
 rho = T[[1, 1]];
 spatial = T[[2 ;; 4, 2 ;; 4]];
 evals = Eigenvalues[(spatial + Transpose[spatial]) / 2.];
 -rho + Total[evals]
 ];

(* For BD scalar we identify r with y in the equatorial plane (x=0). *)
TtraceEMRadial[r_, t_ : 0.] := TtraceEM[0., r, t, paperPars];

(* ----------------------- *)
(* 4. Matter model + energy conditions *)
(* ----------------------- *)

(* Matter trace needed at radius r: T_mat = T_req - T_EM *)
TtraceMatter[r_] := TtraceRequired[r] - TtraceEMRadial[r];

(* Simple isotropic fluid ansatz:
   T^μ_ν(matter) = diag(-ρ, p, p, p),
   trace: T = -ρ + 3p.
   We'll choose ρ = |T|, then solve p so that trace matches:
     T = -ρ + 3p  →  p = (T + ρ)/3
   Then we check WEC/NEC/SEC/DEC for this toy fluid.
*)

rhoMatter[r_] := Abs[TtraceMatter[r]];
pMatter[r_]  := (TtraceMatter[r] + rhoMatter[r]) / 3.0;

(* Energy condition checks for isotropic fluid in orthonormal frame *)
wecQ[r_] := Module[{ρ = rhoMatter[r], p = pMatter[r]}, ρ ≥ 0 && ρ + p ≥ 0];
necQ[r_] := Module[{ρ = rhoMatter[r], p = pMatter[r]}, ρ + p ≥ 0];
secQ[r_] := Module[{ρ = rhoMatter[r], p = pMatter[r]}, ρ + p ≥ 0 && ρ + 3 p ≥ 0];
```

```
decQ[r_] := Module[{ρ = rhoMatter[r], p = pMatter[r]}, ρ ≥ 0 && ρ ≥ Abs[p]];

(* ----------------------- *)
(* 5. Sample radii and summary table *)
(* ----------------------- *)

sampleRs = R0 {1, 2, 3, 4, 5, 10, 50, 100};

dataRows = Table[
  Module[{rr = rVal, bd, Treq, Tem, Tmat, ρ, p, wec, nec, sec, dec},
   bd  = N[bdResidual[r] /. r → rr];
   Treq = N[TtraceRequired[r] /. r → rr];
   Tem  = N[TtraceEMRadial[rr]];
   Tmat = N[TtraceMatter[r] /. r → rr];
   ρ   = N[rhoMatter[r] /. r → rr];
   p   = N[pMatter[r] /. r → rr];
   wec  = wecQ[rr];
   nec  = necQ[rr];
   sec  = secQ[rr];
   dec  = decQ[rr];
   {rr, bd, Treq, Tem, Tmat, ρ, p, wec, nec, sec, dec}
  ],
  {rVal, sampleRs}
];

header = {"r (m)", "BD residual Φ''+2Φ'/r", "Ttrace_req", "Ttrace_EM",
    "Ttrace_matter", "ρ_matter", "p_matter", "WEC", "NEC", "SEC", "DEC"};

Print["============================================="];
Print["Wormhole + Brans-Dicke + EM vortex + matter"];
Print["=============================================\n"];
Print["Parameters:"];
Print["  R0   = ", R0, " m (throat radius)"];
Print["  A    = ", A, " (scalar amplitude scale)"];
Print["  w    = ", w, " m (Gaussian width)"];
Print["  ω_BD = ", ωBD];
Print["  EM vortex parameters (paper-matched):"];
Print["    ", paperPars, "\n"];

Print["Table columns:"];
Print["  r: radial position"];
```

```
Print["  BD residual: Φ''(r) + 2 Φ'(r)/r (0 => BD vacuum)"];
Print["  Ttrace_req: trace BD demands as source"];
Print["  Ttrace_EM: trace of EM vortex stress-energy"];
Print["  Ttrace_matter: what matter must supply"];
Print["  ρ_matter, p_matter: isotropic fluid model"];
Print["  WEC/NEC/SEC/DEC: energy condition flags.\n"];

Grid[Prepend[dataRows, header], Frame → All,
 Alignment → {Left, Baseline}]

(* Optional: plot of BD residual and matter trace near the throat *)
bdPlot = Plot[
 Evaluate[bdResidual[r]], {r, R0, 10 R0},
 PlotRange → All,
 AxesLabel → {"r (m)", "BD residual"},
 PlotLabel → "Brans-Dicke residual Φ''(r)+2 Φ'(r)/r"
];

tPlot = Plot[
 Evaluate[{TtraceRequired[r], TtraceMatter[r]}],
 {r, R0, 10 R0},
 PlotRange → All,
 PlotLegends → {"Ttrace_req", "Ttrace_matter"},
 AxesLabel → {"r (m)", "trace"},
 PlotLabel → "Required vs matter trace (EM trace ≈ 0)"
];

{bdPlot, tPlot}
```

```
==============================================
Wormhole + Brans-Dicke + EM vortex + matter
==============================================

Parameters:
  R0   = 0.001 m (throat radius)
  A    = 1. (scalar amplitude scale)
  w    = 0.01 m (Gaussian width)
  ω_BD = 100.
  EM vortex parameters (paper-matched):
```

{0.00003162, 0.00003162, 3.14159, 0.0001, 0.0001, 0.00139359, 0.0001, -0.0001}

Table columns:

r: radial position

BD residual: Φ''(r) + 2 Φ'(r)/r (0 => BD vacuum)

Ttrace_req: trace BD demands as source

Ttrace_EM: trace of EM vortex stress-energy

Ttrace_matter: what matter must supply

ρ_matter, p_matter: isotropic fluid model

WEC/NEC/SEC/DEC: energy condition flags.

Out[76]=

r (m)	BD residu⋱al Φ''+2Φ'/r	Ttrace_r⋱eq	Ttrace_EM	Ttrace_m⋱atter	ρ_matter	p_matter	WEC	NEC	SEC	DEC
0.001	2.77556×10^{-12}	2.24185×10^{-11}	0.	2.24185×10^{-11}	2.24185×10^{-11}	1.49457×10^{-11}	True	True	True	True
0.002	29 503.5	238 303.	0.	238 303.	238 303.	158 869.	True	True	True	True
0.003	37 406.7	302 138.	-3.23117×10^{-27}	302 138.	302 138.	201 426.	True	True	True	True
0.004	38 659.3	312 255.	1.29247×10^{-26}	312 255.	312 255.	208 170.	True	True	True	True
0.005	36 539.9	295 137.	9.69352×10^{-27}	295 137.	295 137.	196 758.	True	True	True	True
0.01	11 050.3	89 254.3	1.29247×10^{-26}	89 254.3	89 254.3	59 502.9	True	True	True	True
0.05	-0.00003⋱298	-0.00026⋱6383	0.	-0.00026⋱6383	0.000266⋱383	0.	True	True	True	True
0.1	-1.04008×10^{-36}	-8.40082×10^{-36}	2.82119×10^{-37}	8.74568×10^{-36}	8.74568×10^{-36}	5.83045×10^{-36}	True	True	True	True

Out[79]=