In[•]:=
```
(*------------------------------CLEAN SETUP------------------------------*)
ClearAll["Global`*"]

(*Optimized 2-meter wormhole parameters*)
R0 = 2.0;    (*2 meter throat radius*)
A0 = 0.01;   (*optimized amplitude*)
w = 0.2;    (*optimized width*)
ε = 0.04;   (*regulator-not needed directly here*)

Print["Using optimized wormhole parameters:"];
Print["R0 = ", R0, ", A0 = ", A0, ", w = ", w, ", ε = ", ε];

(*Conformal potential Φ(r)*)
potential[r_, A0_, R0_, w_] := -A0 * (1 - R0 / r) * Exp[-((r - R0) / w)^2];

dPotential[r_, A0_, R0_, w_] := D[potential[r, A0, R0, w], r];
ddPotential[r_, A0_, R0_, w_] := D[potential[r, A0, R0, w], {r, 2}];

(*Ω integrand (same form as your 1mm version)*)
omegaIntegrand[r_, A0_, R0_, w_] :=
  Module[{phi, dphi, ddphi, rSafe}, rSafe = SetPrecision[r, 30];
   phi = potential[rSafe, A0, R0, w];
   dphi = dPotential[rSafe, A0, R0, w];
   ddphi = ddPotential[rSafe, A0, R0, w];
   Exp[2 phi] * (ddphi + dphi / rSafe) * rSafe / (8 Pi)];

Print["\nCalculating total Ω for the optimized 2-meter wormhole..."];

totalOmega =
  Quiet@NIntegrate[omegaIntegrand[r, A0, R0, w], {r, R0, R0 + 5 w}, Method → "GlobalAdaptive",
    WorkingPrecision → 30, PrecisionGoal → 8, AccuracyGoal → Infinity];

Print["\n-----------------------------------------------"];
Print["   ✔ Total Ω = ", ScientificForm[totalOmega, 5]];
Print["-----------------------------------------------"];
Print["Raw value: ", totalOmega];
```

Using optimized wormhole parameters:

R0 = 2., A0 = 0.01, w = 0.2, ε = 0.04

Calculating total Ω for the optimized 2-meter wormhole...

✔ Total $\Omega = 3.9753 \times 10^{-4}$

Raw value: 0.000397532449043051801994045440901