CoTiMA 1.0.4: Power analysis updates

28 October 2026

1 Verified calculation routes and interpretation

The power-analysis implementation separates design validation, invariant-model fitting/loading, parameter extraction, implied covariance calculations, regression effects, power probabilities, and output formatting. These components are tested independently. Invariant CT fitting still uses the selected estimation backend.

For the subsequent discrete-time regression calculations, `powerMethod="auto"` checks that continuous Gaussian indicators are recorded and that the regression has every T0 predictor freely predicting every T1 outcome, with unrestricted covariance matrices within both time blocks. If eligible, regression coefficients and full/reduced residual variances are computed directly from the implied covariance matrix. No lavaan regression fits are needed.

Supported Gaussian regressions with fixed/omitted slopes or covariance restrictions use lavaan. Missing legacy indicator-type metadata also leads to lavaan with an explicit Gaussian-assumption note. `powerMethod="lavaan"` requests this route explicitly; `powerMethod="covariance"` requires the eligibility check to pass. Both routes print a CoTiMA-style console message and store their explanation in `result$summary$calculation_note`. The decision and numerical details are available in `regression_check`, `power_method`, and `regression_details`. Regression fits are reused across requested power levels.

`regressionModel=NULL` builds the unrestricted Gaussian regression. An optional lavaan syntax string can specify a supported restricted regression using variables `V1T0,...,VpT0,V1T1,...,VpT1`. For example, fixing a cross-time slope to zero prevents testing that slope and gives it NA power. Equality constraints/shared labels, latent measurement models and known binary or ordinal indicators are rejected: a one-coefficient Gaussian power formula does not automatically become valid merely by fitting them in lavaan. Use design-specific simulation for these cases. Restrictions on continuous-time coefficients are not automatically imposed as discrete-time zeros; indirect paths can generate a nonzero discrete-time effect.

The fast default uses the complete two-sided noncentral-t rejection probability, evaluated through its equivalent noncentral F distribution. A bounded integer bisection search finds the smallest admissible N achieving the requested power. Zero effects can require infinite N; very strong effects can already meet the target at the smallest admissible N. `useMBESS=TRUE` retains MBESS's legacy upper-tail probability with a two-sided critical value, while retaining the fast integer search. The historical internal `nestedProbFunT` was a root target: it returned calculated power minus requested power, with $f^2 = (R^2_{full} - R^2_{reduced})/(1 - R^2_{full})$. Its purpose was numerical sample-size inversion, not hierarchical nesting. The new helpers make these operations explicit and independently testable.

Only positive lags inside `timeRange` are analysed; observed lags within the bounds are included. The first positive lag is retained. Two-wave studies and models with three or more variables are supported. The legacy `Opt. Lag` row selects the first actual minimizing lag; all tied lags are returned in `summary$estimates[["All optimal lags"]]`. Missing post-hoc powers are NA, not zero; the Mean/Median rows summarize available first-interval powers. Lags outside the analysed range remain missing.

Recorded measurement-error variance, non-identity/free loadings, or augmented latent-state models are rejected rather than silently treated as directly observed state regressions. For restricted regressions the partial-R^2 power calculation remains an approximation, not a general SEM power test.

The results are plug-in power approximations for discrete-time regression coefficients. They do not propagate CT estimation uncertainty or constitute power calculations for the pooled CT drift test, a full multiwave design, or a hierarchical sampling design. The significance-range output uses a two-sided ML Wald criterion, which is distinct from the noncentral-t planning calculation. It reports the smallest/largest significant evaluated lag without asserting significance at every intermediate lag. Power calculations restore original input time units from invariant-fit scaling metadata. Legacy caches lacking that metadata are assumed to use original units.

Reduced-fit caching forces the lavaan route. Newly written caches include design and model metadata and are rejected if reused with another covariance/model. Old bare lavaan reduced-fit caches must be regenerated. The legacy `skipScaling=FALSE` and non-NULL `useSampleFraction` settings previously prepared data that the invariant fitting interface did not actually consume; they now raise an explicit error. Rescale/subset the input studies explicitly.