

Validate Power: Westfall-Young

April 11, 2022

Explanation

Westfall-Young procedures

The Westfall-Young procedure was validated separately due to its unique complications and computational burden.

First, we wrote a series of careful unit tests ensuring that our code worked as expected for each step of the WY procedure. One of these tests compared results from our procedure to the WY procedure in the multtest package, and found it matched quite well.

Second, we also tested the WY procedures using the same simulation procedure described above. For constant effect and fixed effect models, the power estimation matches well between PUM and the simulations. However, for random effects models, the two methods diverge in some cases. We find that the simulated power matches the PUMP-calculated power only when (1) there is a large number of blocks/clusters, and (2) the user has a large number of WY permutations. Below, we discuss the single-step procedure because it is simpler, although the same concepts hold for the step-down procedure. Although it is difficult to verify why this discrepancy occurs, our hypothesis is that this behavior occurs due to the combination of the sensitivity of the WY procedure, and the instability of the random effects model.

The goal of the WY procedure is to estimate the adjusted p -value for outcome i

$$\tilde{p}_i = Pr \left(\min_{1 \leq j \leq k} P_j \leq p_i \mid H_0^C \right).$$

where P_j is a random variable representing a p -value for outcome j , and lowercase p_i is the observed realization for outcome i . We assume a set of k tests with corresponding null hypotheses H_{0i} for $i = 1, \dots, k$. The complete null hypothesis is the setting where all the null hypotheses are true: $H_0^C = \cap_{i=1}^k H_{0i} = \{\text{all } H_i \text{ are true}\}$. In order to estimate this p -value, the p -value is calculated across B permutations

$$\tilde{p}_i = \frac{1}{B} \sum_{b=1}^B \mathbb{I} \left(\min_{1 \leq j \leq k} p_{b,j}^* \leq p_i \right).$$

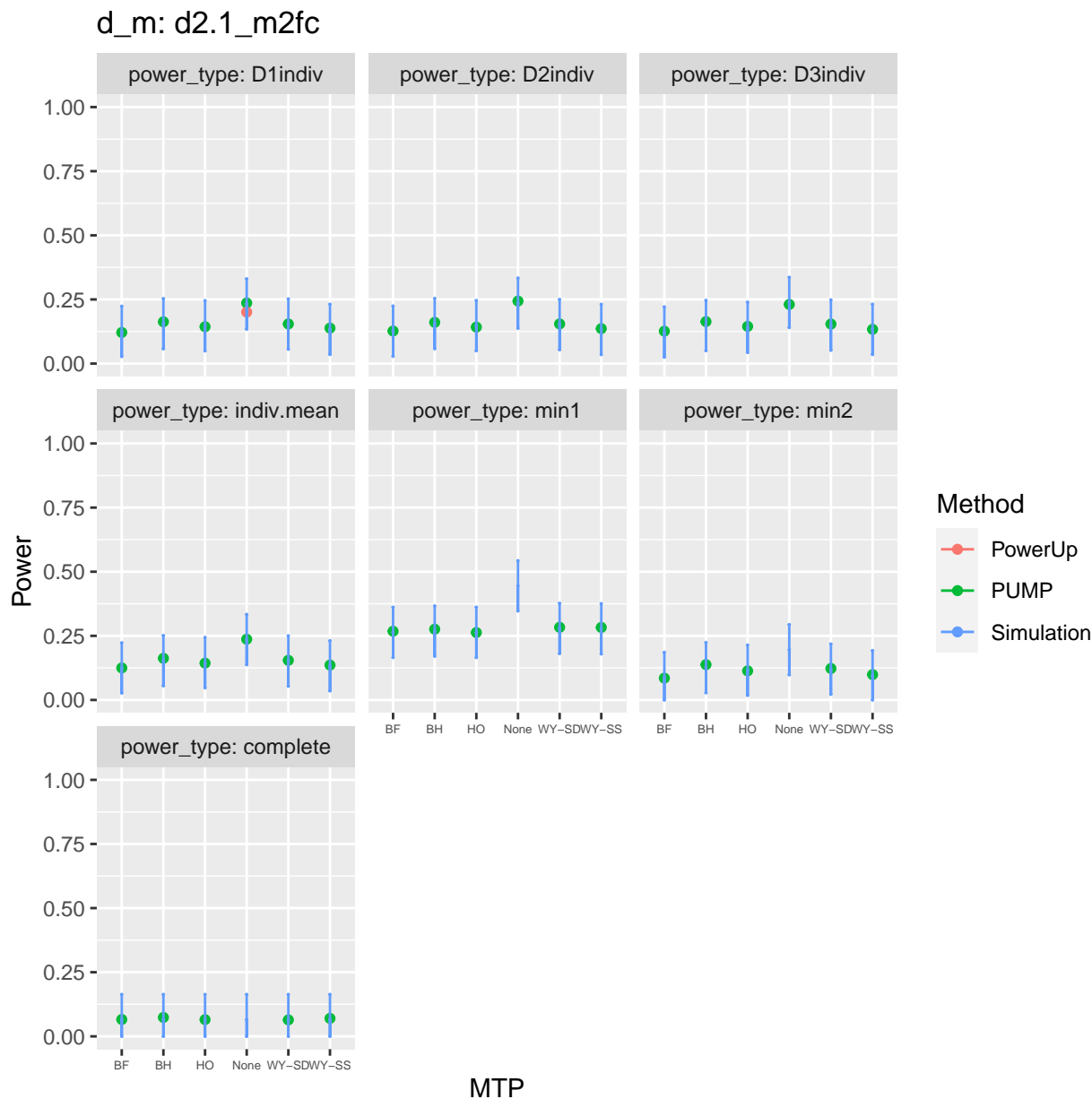
In order for the procedure to be correct, we should be generating the p -values P_j from the true distribution of p -values under the null hypothesis. We generally are testing outcomes that are truly significant, so the observed p -values are small. Thus, looking at these expressions, we are concerned about tail behavior; we are testing whether a small observed p -value is less than the minimum of the generated p -values of our multiple outcomes. This combination of factors means that a small discrepancy in the tails between the generated null distribution and the true distribution could have a large impact on the adjusted p -value.

Given that we are relying on tail behavior, this explains why we need both a large number of permutations, and a large number of blocks/clusters. The large number of permutations is required because we are trying to estimate a rare event. With a significant outcome, it is rare that the observed p -value will be less than the minimum of p -values generated from the null distribution. The large number of blocks/clusters is required to accurately estimate the tail behavior for random effects models. With a small number of blocks/clusters, we may not properly estimate the spread of the distribution, resulting in a poor estimate of the tail behavior.

Remark due to computational burden, for certain models we only tested WY-SS and not WY-SD. In some cases, we had to reduce the number of simulations, number of permutations, or the number of units in order to run a validation that was not computationally intractable.

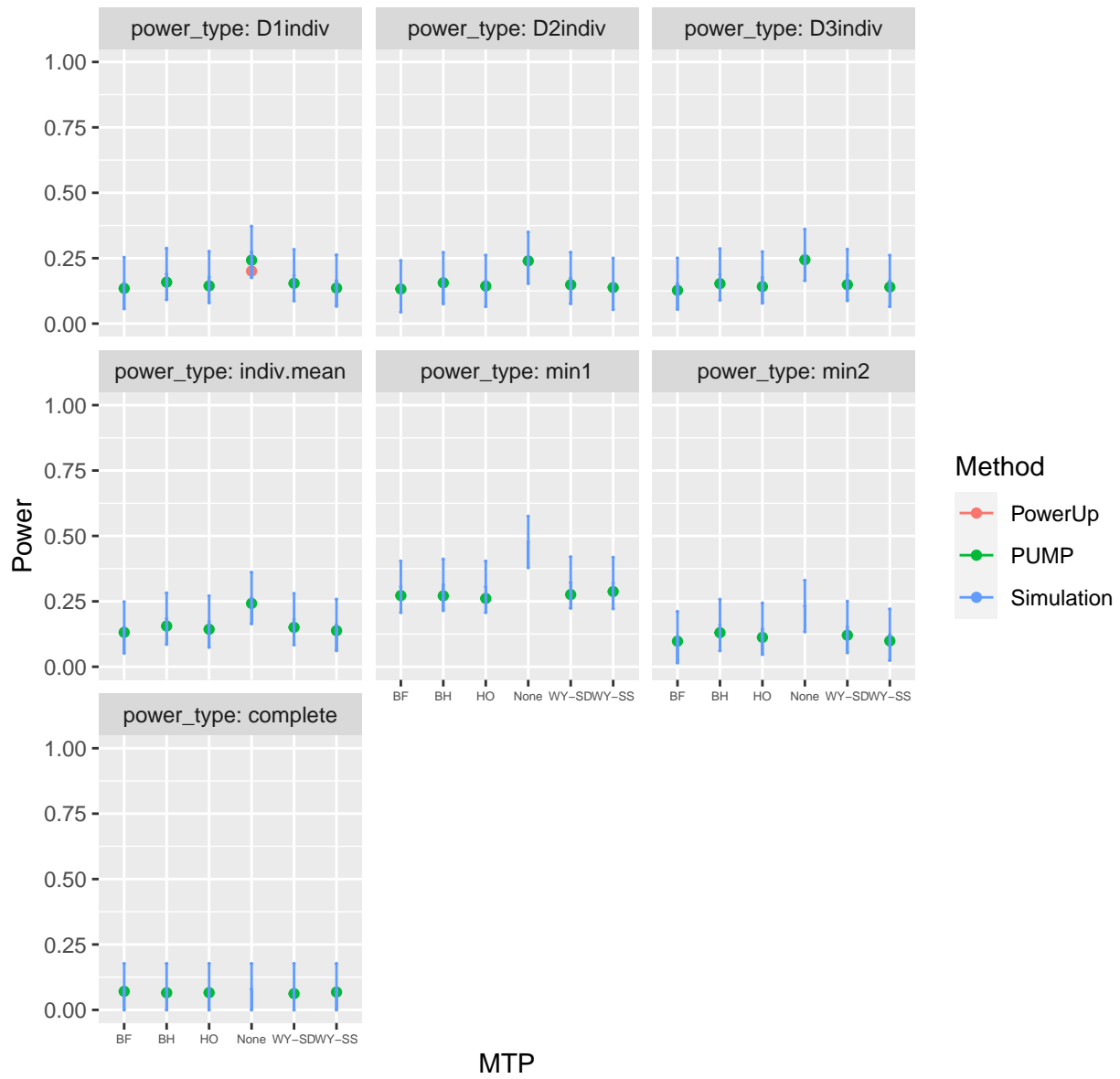
d2.1 models

Constant effects



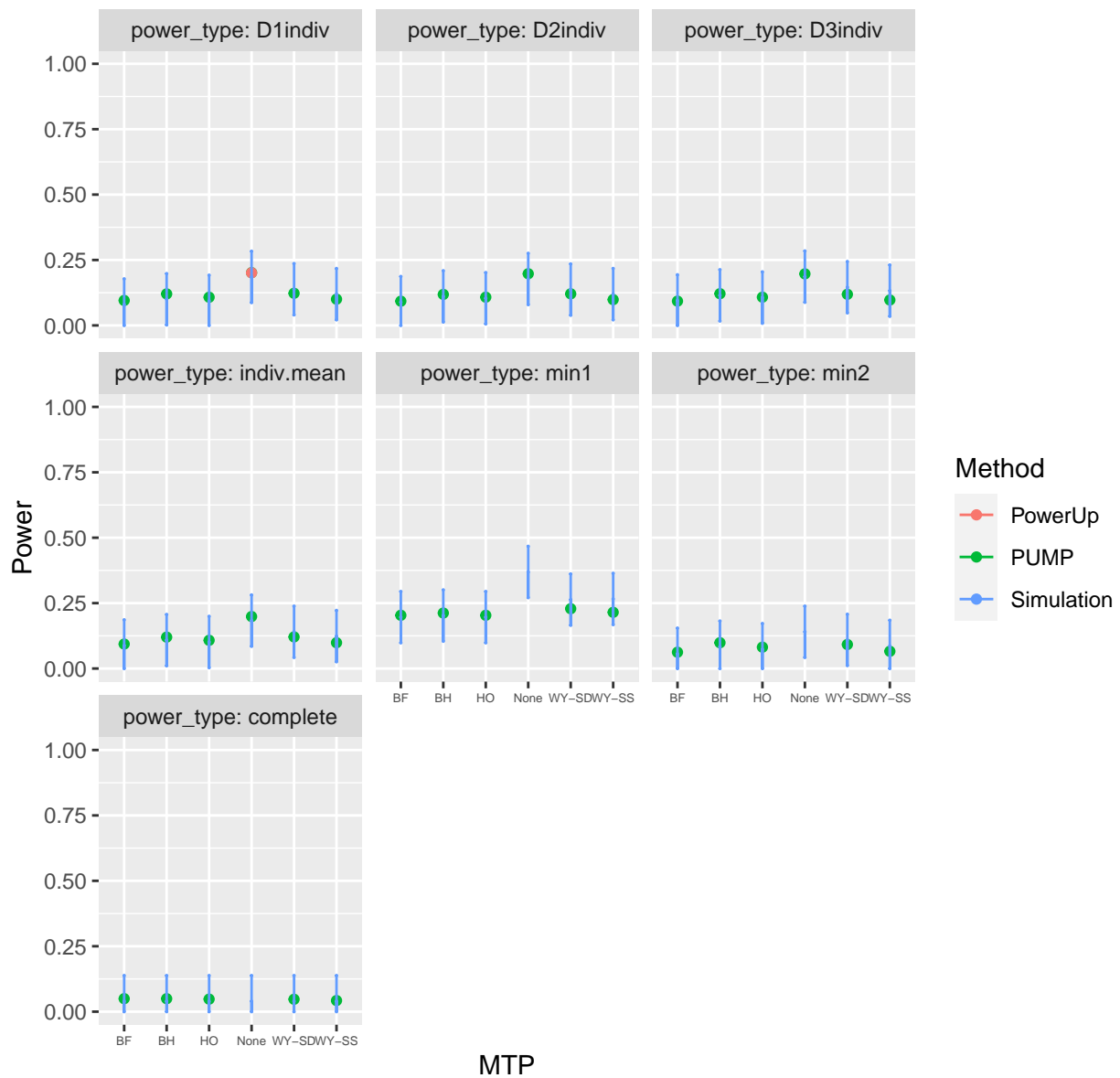
Fixed effects

d_m: d2.1_m2ff

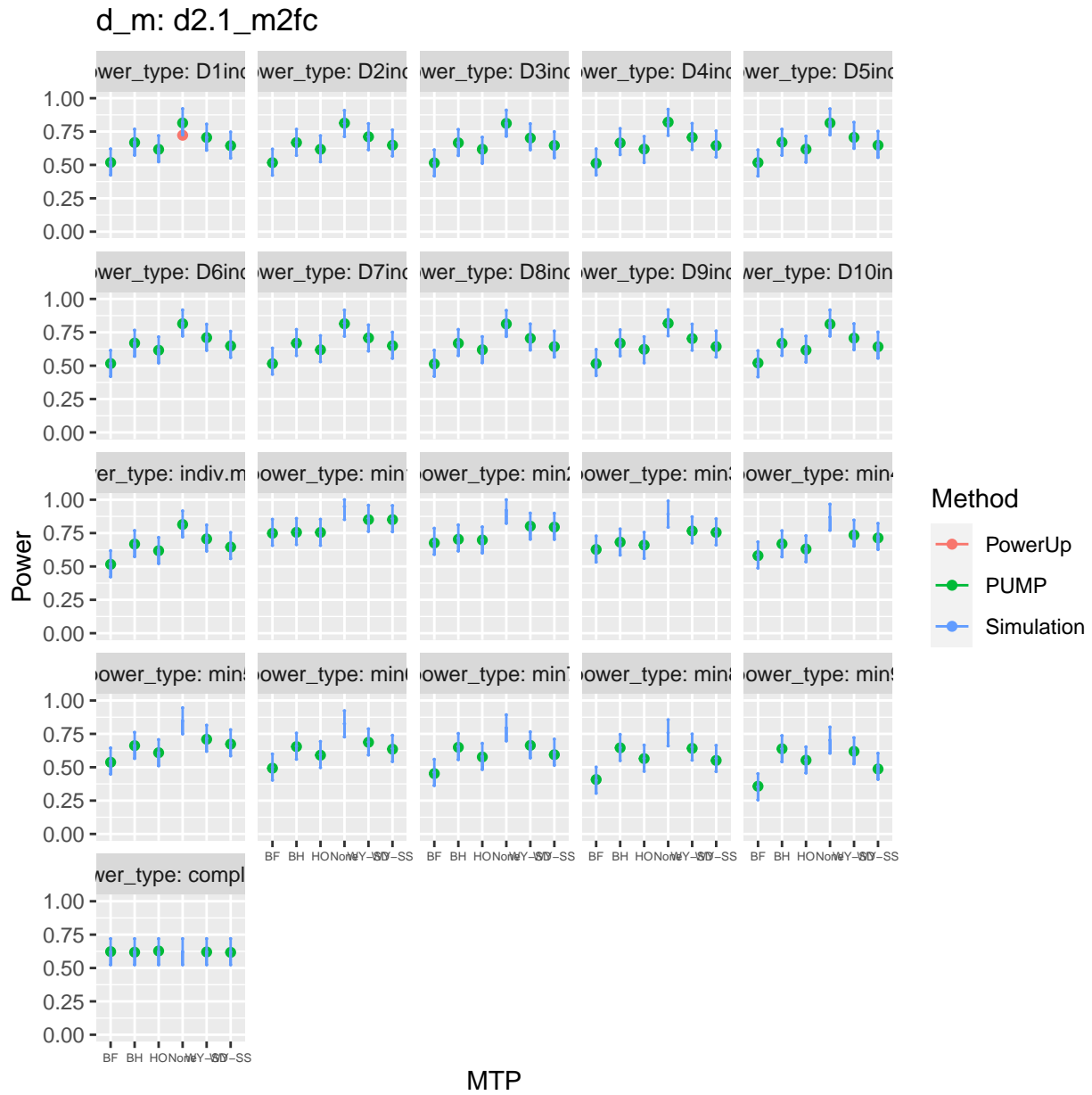


Random effects

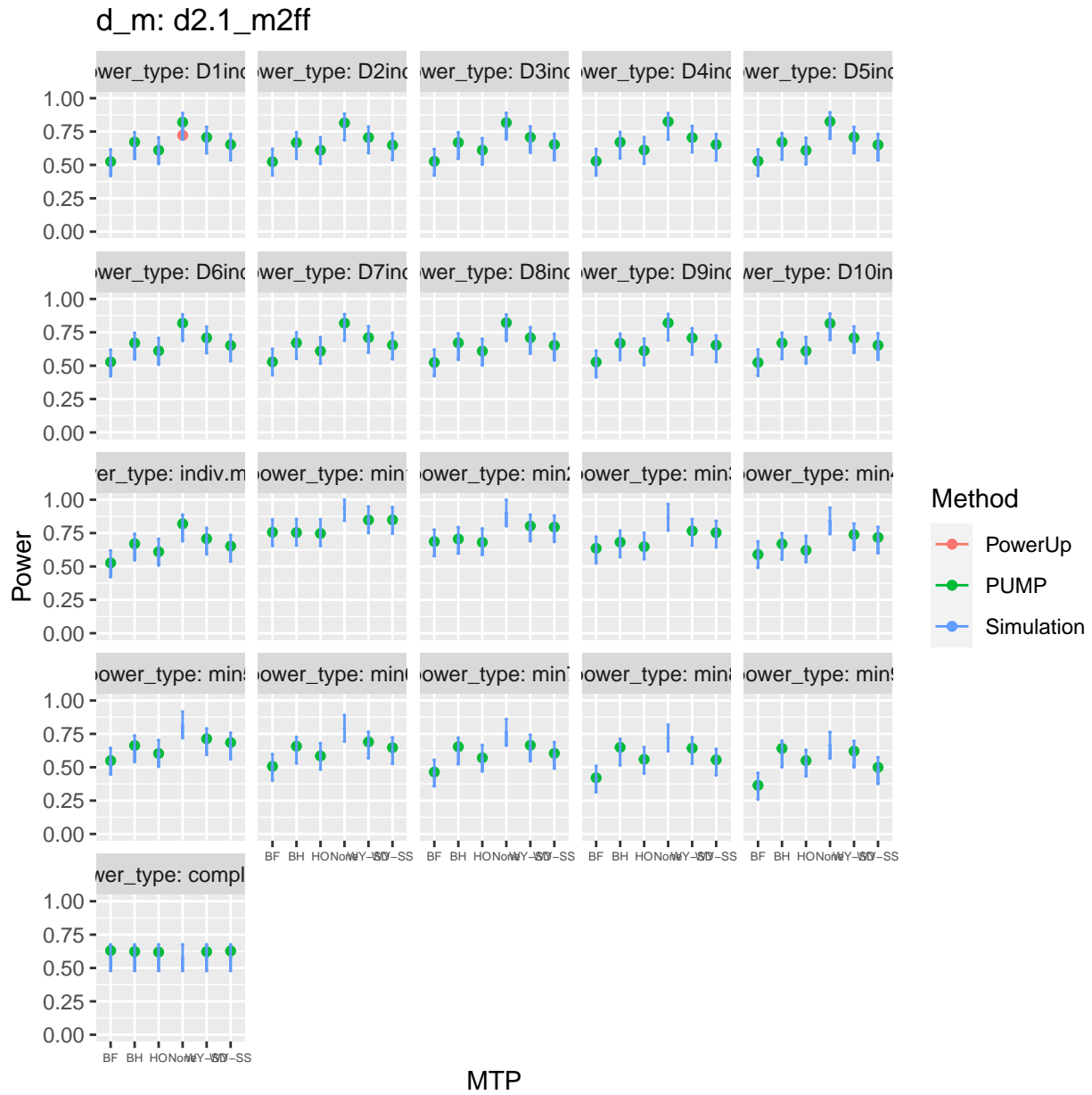
d_m: d2.1_m2fr



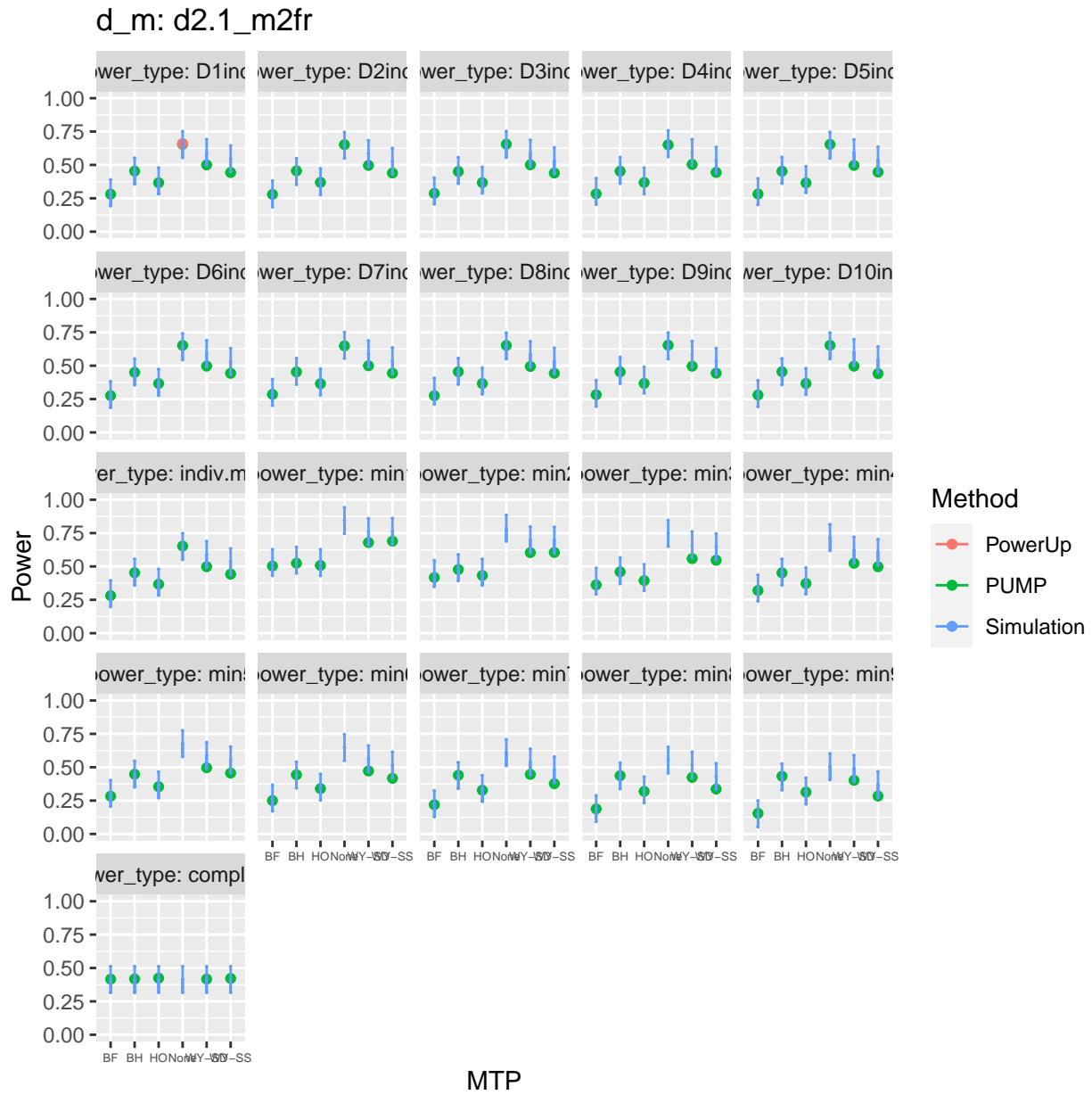
Constant effects, $M = 10$ and $\rho = 0.85$



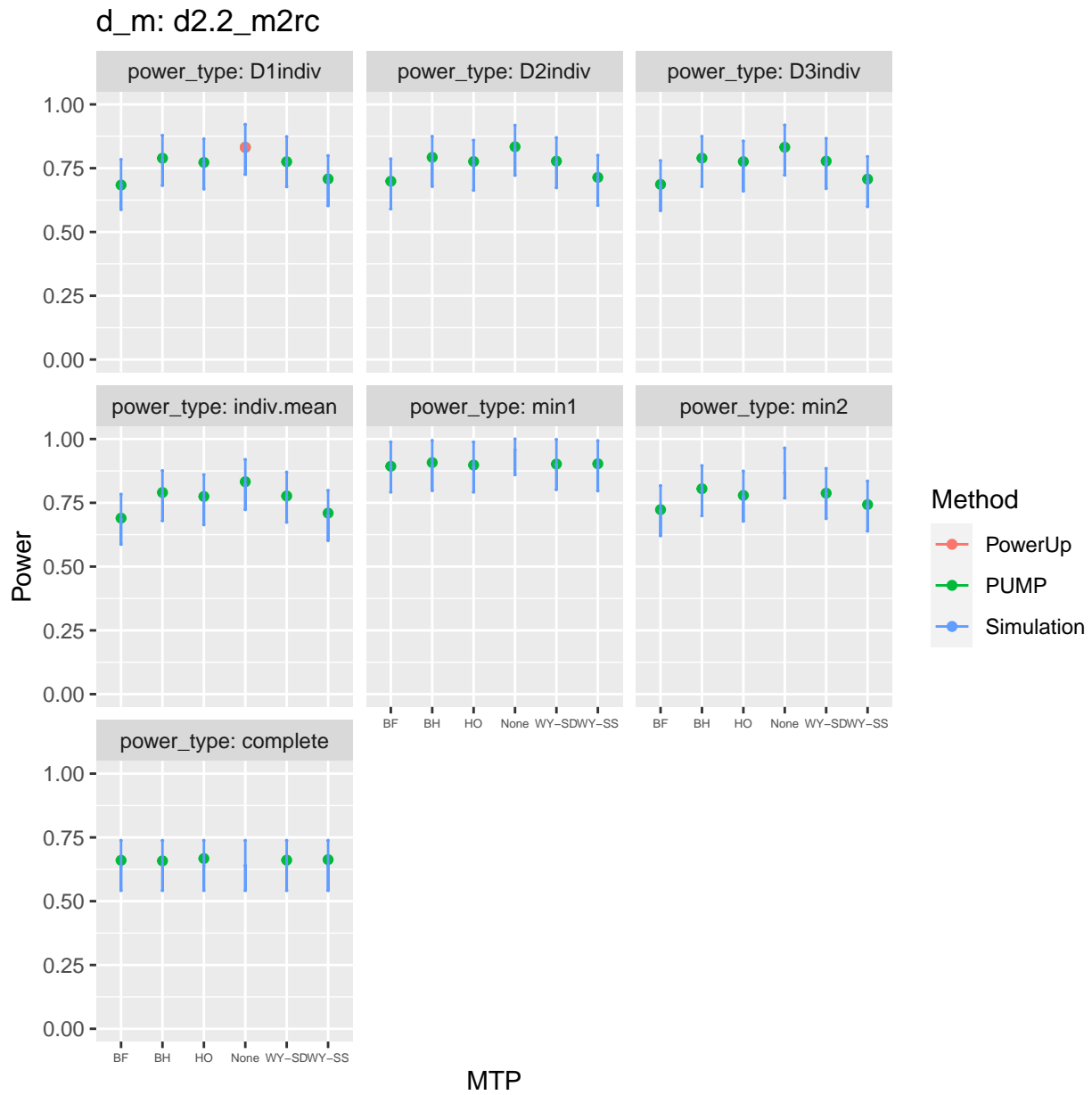
Fixed effects, $M = 10$ and $\rho = 0.85$



Random effects, $M = 10$ and $\rho = 0.85$



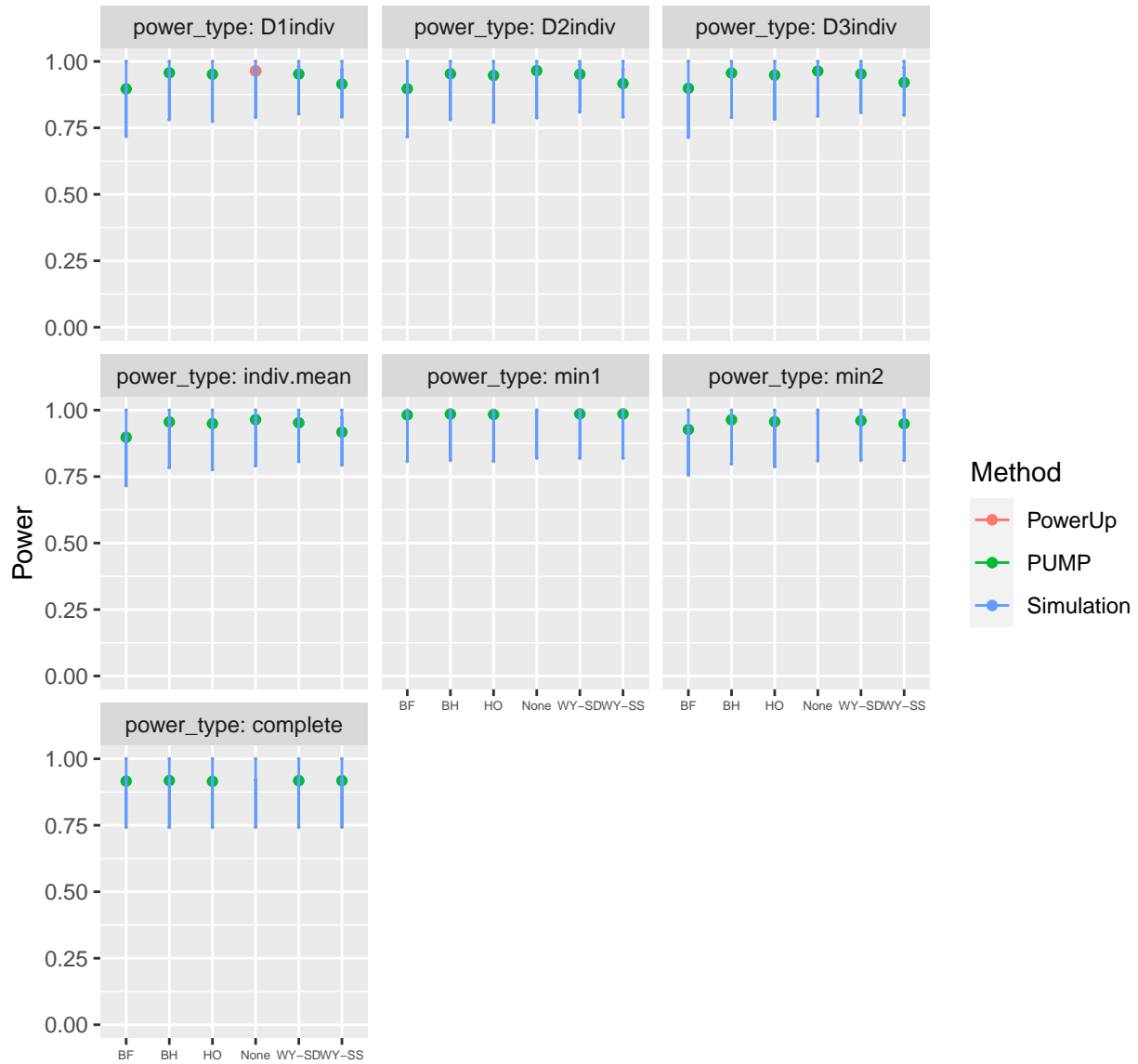
d2.2 models



d3.1 models

$K = 10, J = 10$

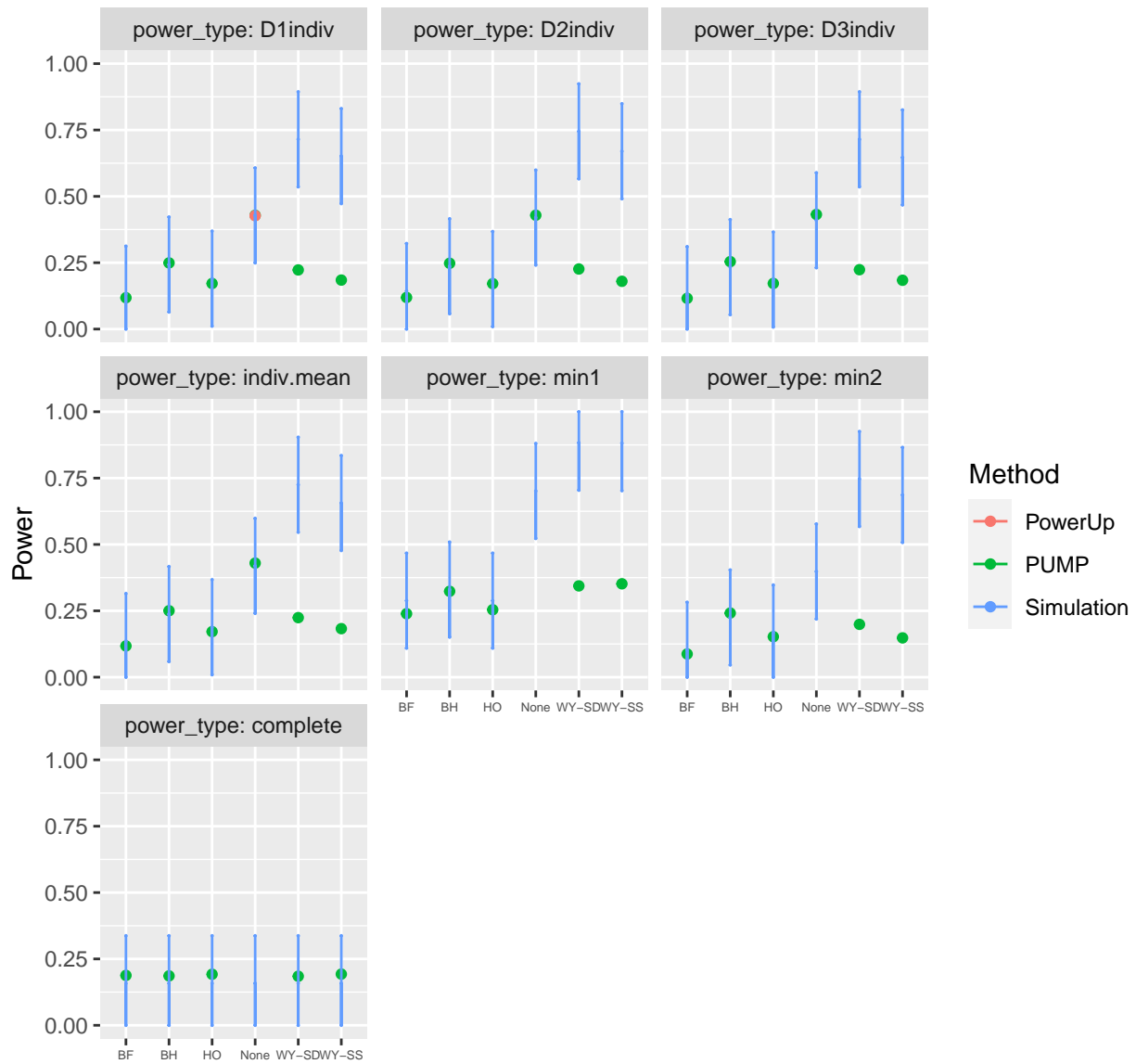
d_m: d3.1_m3rr2rr



MTP

$K = 5, J = 5$

d_m: d3.1_m3rr2rr

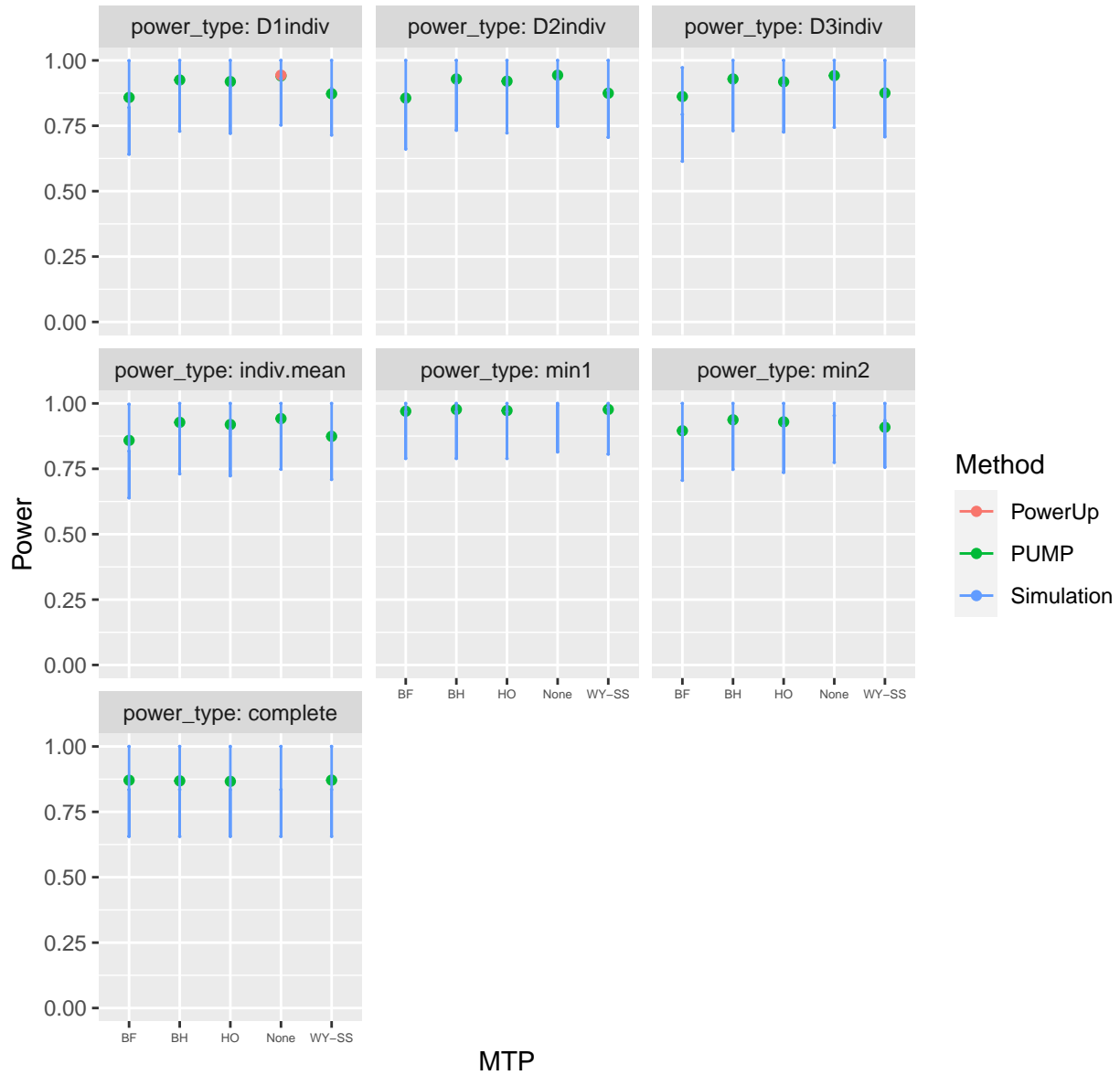


MTP

d3.3 models

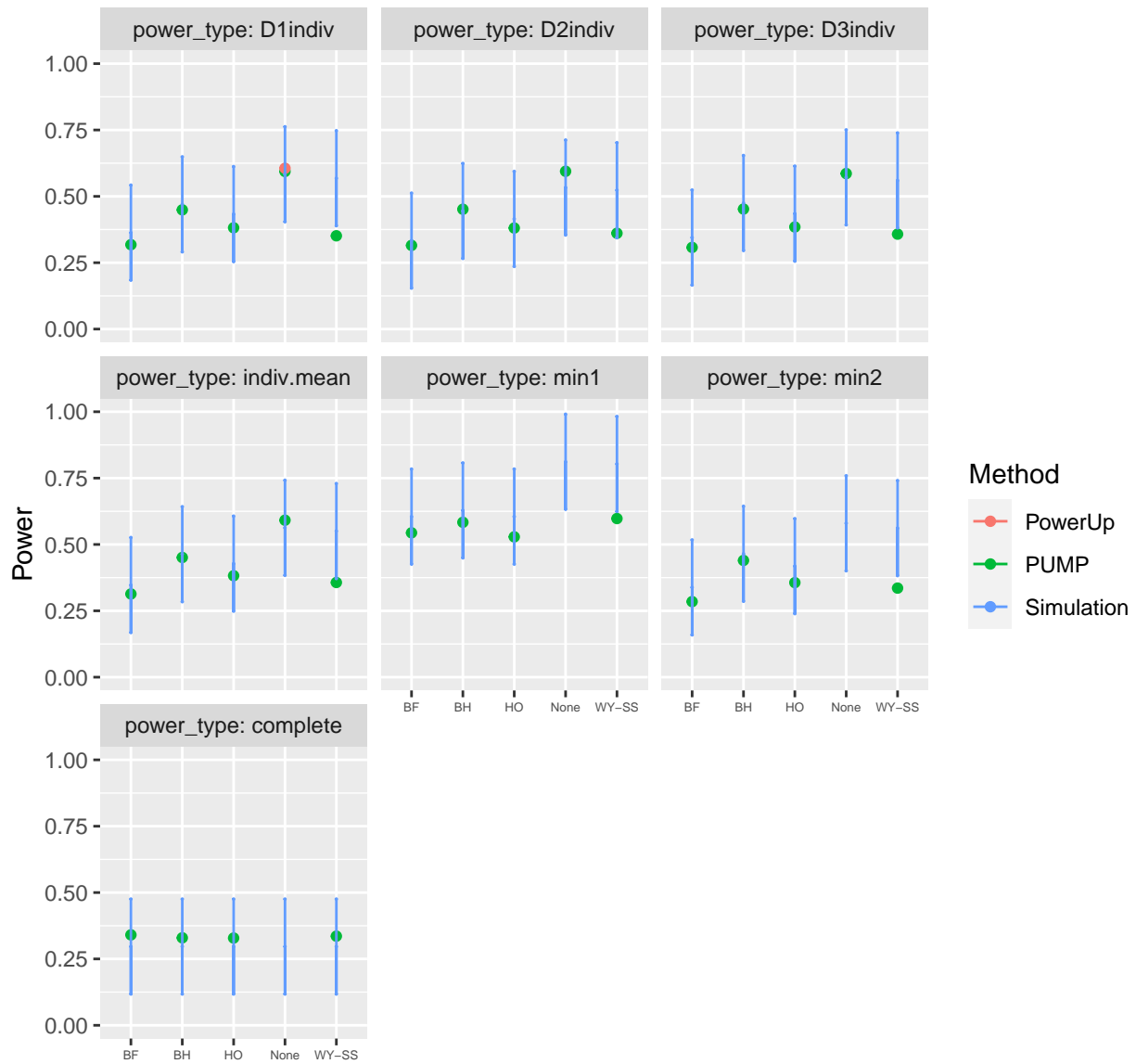
$J = 20, K = 20$

d_m: d3.3_m3rc2rc



$J = 10, K = 10$

d_m: d3.3_m3rc2rc

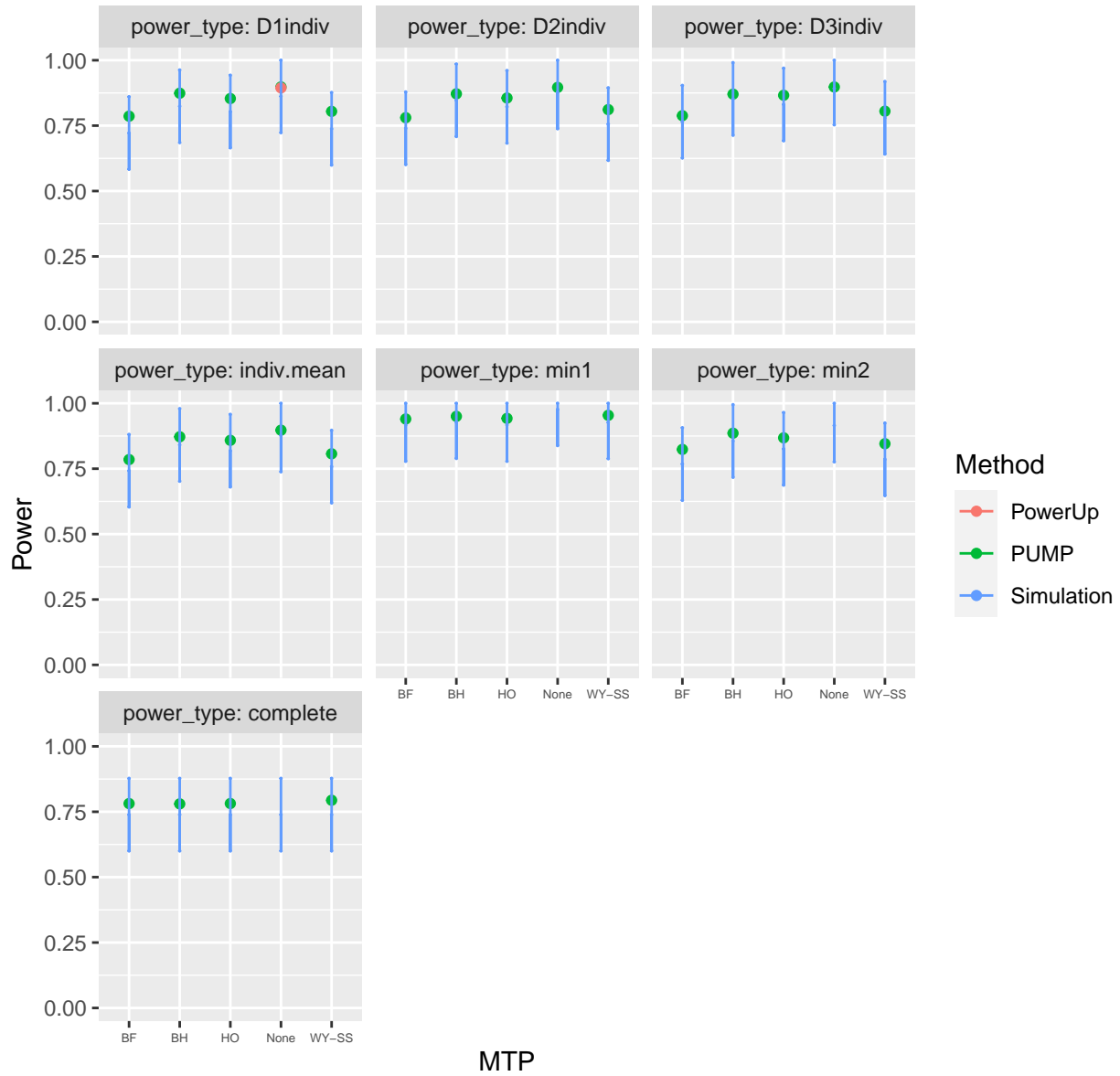


MTP

d3.2 models

Constant effects

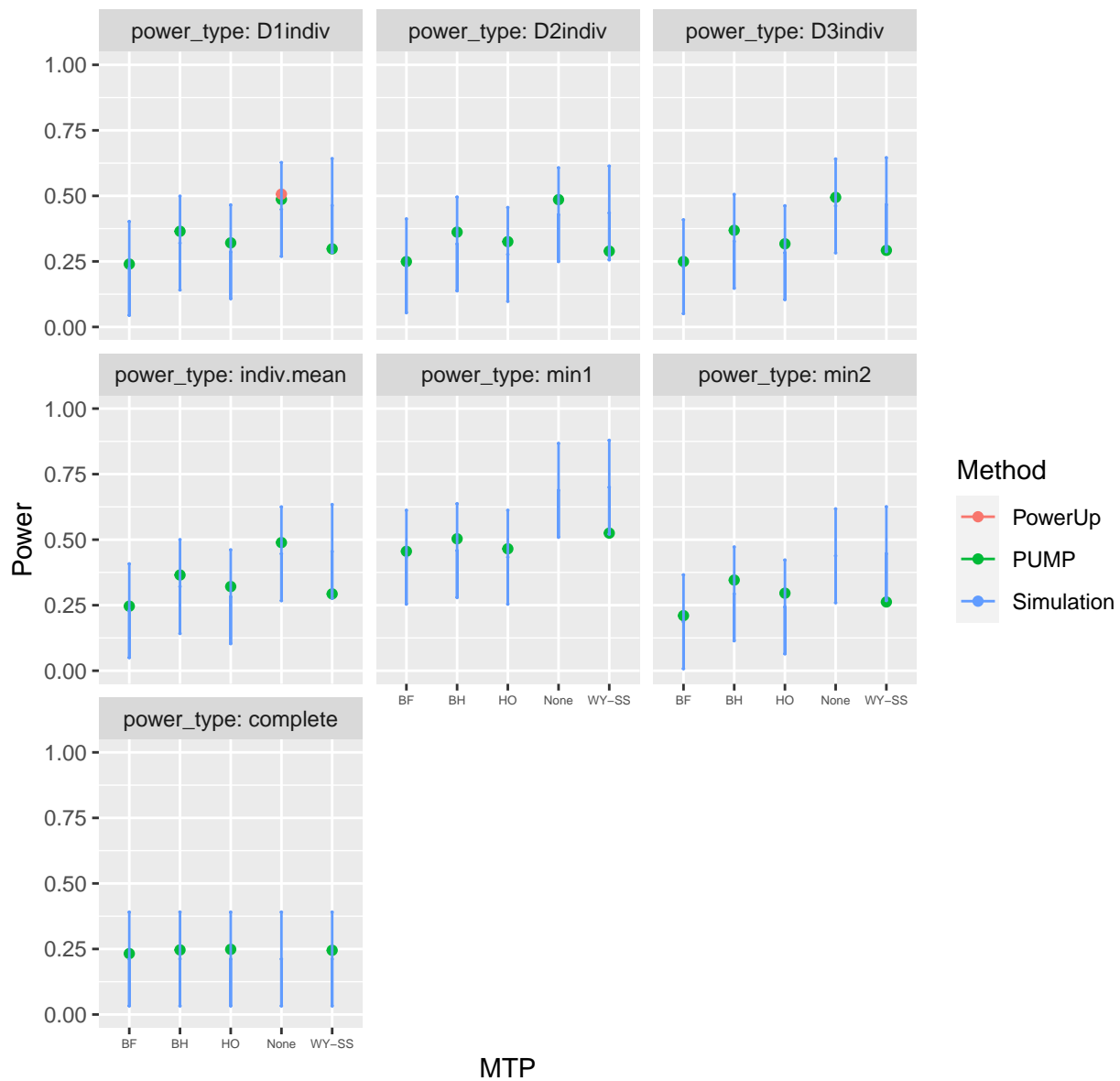
d_m: d3.2_m3ff2rc



Random effects

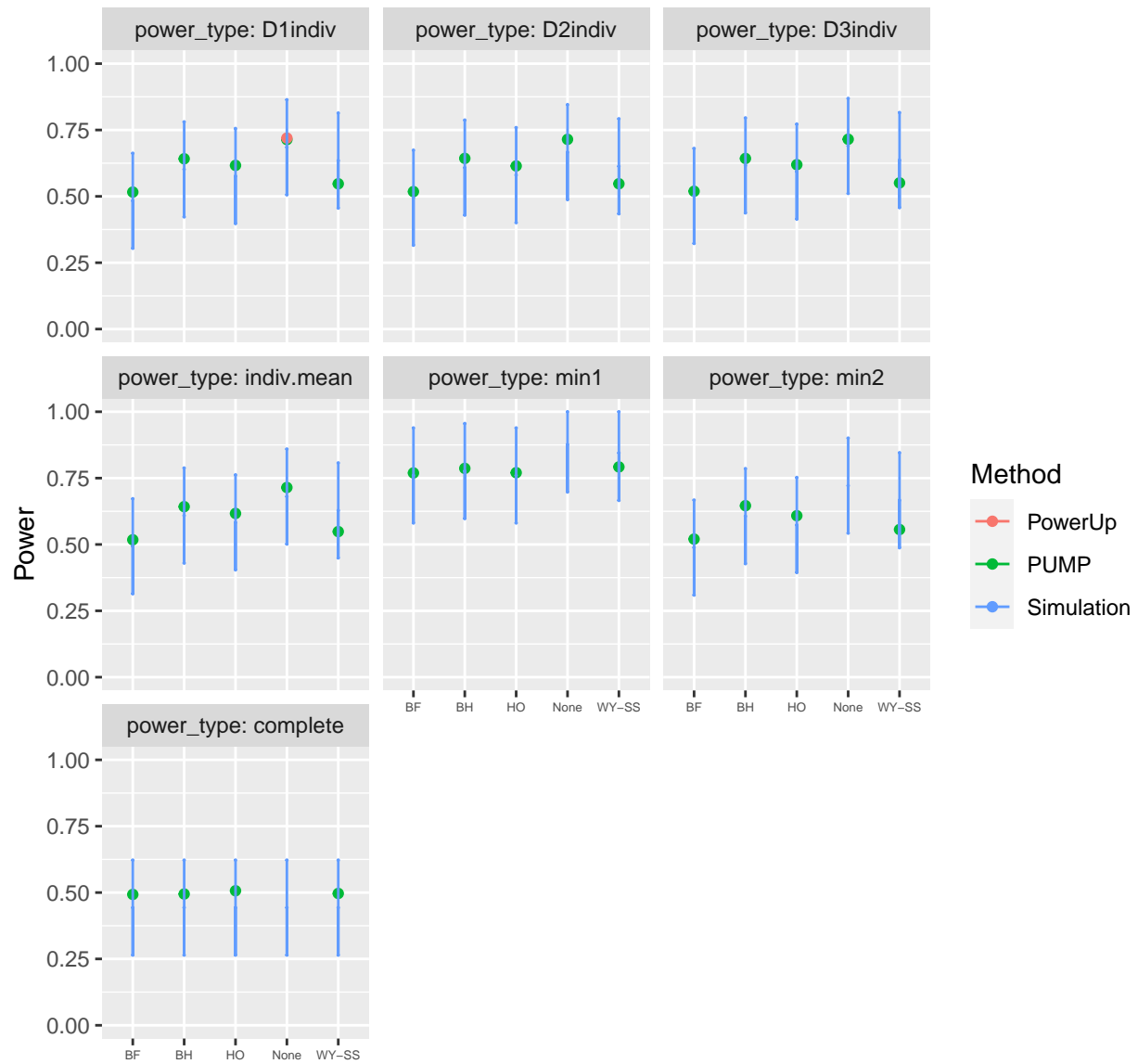
$J = 10, K = 10$

d_m: d3.2_m3rr2rc



$J = 20, K = 20$

d_m: d3.2_m3rr2rc



MTP