# Mixing Random with pMEDICI for CIT Models without constraints

## Previous considerations

* Previous experiments were conducted on simple models, having less than 10 parameters.
* It seems that the Colbourn formula works well with combinatorial problems with medium-high complexity, i.e., with more than 8 parameters (and/or high strength or high alphabet size). In that cases, our mix random-pMEDICI performs better than the baseline approach using only pMEDICI.
* For *v*=2 the prediction of the seeds size given by the Colbourn formula seems to fit the best scenario we measured with our previous experiments.
* Sometimes we saw that the formula gives a different number of seeds if compared to the best scenario, but then, the results in terms of time and size are comparable. **For this reason, in this report, I will only report the comparison of time obtained with the number of seeds given by the formula, with that of the baseline and of the best scenario.**

## Objective of this new report

* Investigate whether the Colbourn formula can be somehow weighted in order to work well with models having *v* greater than 2
* Investigate the generalizability of the formula on models in two additional categories:
  + Medium-complexity models (with *k* between 10 and 30)
  + High-complexity models (with *k* between 30 and 100)

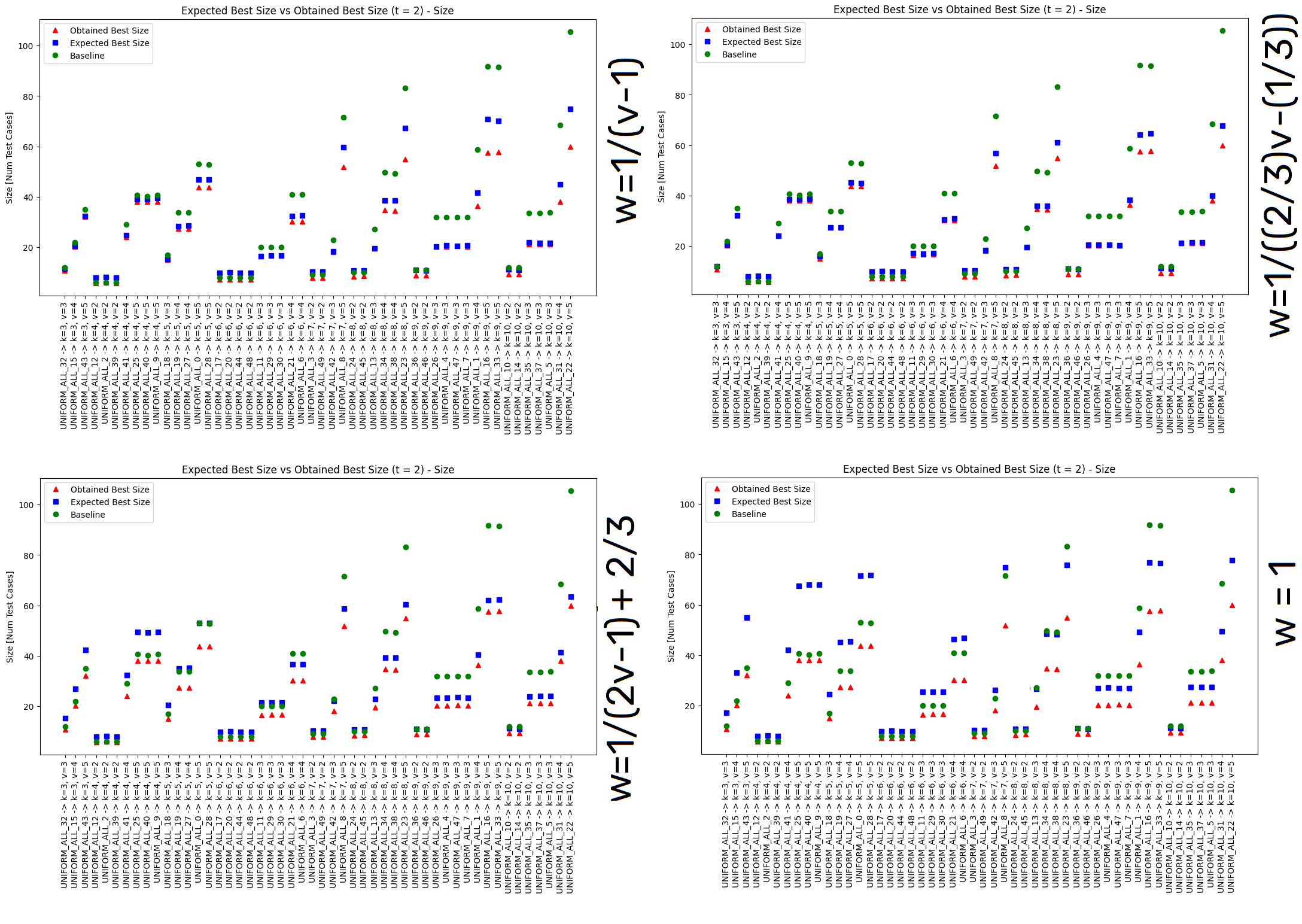
In all cases, I’m going to use *v* between 2 and 7

* Investigate whether the same conclusions we draw for UNIFORM models can be applied to MCA models, where the alphabet size is different for each parameter

## Experiments on UNIFORM models

### Weighting the formula

The idea behind the chosen weights is that since the formula was performing well with *v = 2*, then I wanted to find a weighting factor to be multiplied for every occurrence of *v* in Colbourn’s formula that led to no change when *v = 2*, but decreased the predicted value for higher values of *v*.



The two weighting factors in the first row seem to be the best ones (among those considered) since with high values of *v* the predicted values are closer to that obtained in the best scenario.

However, in further experiments with more complex models, I observed that using weight 1 (i.e., by not changing the formula) seems to be the best solution even with higher values for *v*.

### Generalizability of the formula for medium-complexity models

### Immagine che contiene testo, schermata, diagramma, linea Descrizione generata automaticamente Immagine che contiene testo, schermata, diagramma, linea Descrizione generata automaticamente

### Generalizability of the formula for high-complexity models

## Experiments on MCA models

When dealing with MCA models, since *v* is not a single value, but assumes a different value for each parameter, we have to decide which *v* to use in the formula.

I evaluated three possibilities:

* The minimum v
* The medium v
* The maximum v

With the first possibility, I think that the results should be the best one since we underestimate the number of seeds, but still we include some of them. This should be justified also by the fact that our objective is to do better than the baseline approach, and not (even if it would be desirable) as in the best scenario possible.

Instead, if one uses the medium of maximum value of *v*, in my opinion, we would overestimate the number of seeds and, thus, we may have test suites with many more test cases than those needed.

For this reason, in the following experiments, I computed the Colbourn formula using the minimum *v*.

The experiments have been performed multiple times, according to this schema:

* 100 repetitions for low-complexity models
* 50 repetitions for medium-complexity models
* 10 repetitions for high-complexity models

These numbers are justified by the experiments performed by Ludwing showing that when the complexity (i.e., size or number of parameters) of the models increases, the variability of random arrays decreases.

### Generalizability of the formula for low-complexity models

Immagine che contiene schermata, testo, diagramma, linea

Descrizione generata automaticamente

Immagine che contiene schermata, testo, linea, diagramma

Descrizione generata automaticamente

Also for MCA models, it seems that for most models having *k* > 8 the number given by Colbourn’s formula and that of the best scenario lead to comparable time/size. Moreover, in most models with *k* > 8, both approaches perform better than the baseline.

### Generalizability of the formula for medium-complexity models

Immagine che contiene schermata, testo, diagramma, linea

Descrizione generata automaticamente

Immagine che contiene schermata, testo, Policromia, diagramma

Descrizione generata automaticamente

For all medium-complexity models, even if they are MCA and not UNIFORM, the number given by Colbourn’s formula and that of the best scenario lead to comparable time/size. Moreover, in all models, both approaches perform better than the baseline.

### Generalizability of the formula for high-complexity models

## Immagine che contiene schermata, testo, linea, Policromia Descrizione generata automaticamente

Immagine che contiene schermata, testo, linea, Carattere

Descrizione generata automaticamente

Also in the case of high-complexity MCA models, the number given by Colbourn’s formula leads to a size very close to that of the best scenario. Similar considerations hold for time. Moreover, in all models, both approaches perform better than the baseline.