Last factor optimization in rLmin

In the rLmin function, a set columns c is given to the algorithm, to determine if it forms a rL-minimal design.

To perform this check, the algorithm compares the matrix added factors of c to the matrix of added factors formed by any permutation in c. Those permutations are the $\binom{|c|}{r}$ possible ways of choosing another set of r basic factors among c.

The idea of last factor optimization is that the last factor added to c must be present in the new set of basic factors. This reduces the number of permutations to test to $\binom{|c|-1}{r-1}$. However, we must ensure that, if a set c is **not** rL-minimal, the rL-smaller permutation of c, will be found in those $\binom{|c|-1}{r-1}$ permutations.

Example

Let us consider the set of columns $c = (1, 2, 4, 8, 3, 5, 9, 15)$. It is a candidate design generated by the search-table algorithm by adding column 15 to the parent design $c_p = (1, 2, 4, 8, 3, 5, 9)$. This set of factors is not rL-minimal, so a permutation c should output a rL-smaller matrix of added factors.

In c, there initial basic factors are $c_r = (1, 2, 4, 8)$ and the added factors are $c_k = (3, 5, 9, 15)$. The added factor matrix is thus

```
cr = [1 2 4 8];
ck = [3 5 9 15];
G = Gmat(4);
L = G(:,ck)
```

```
L = 4×4
     1     1     1     1
     1     0     0     1
     0     1     0     1
     0     0     1     1
```

Now, the basic algorithm tests all $\binom{8}{4} = 70$ combinations of 4 columns, among which, some yields the rL-smaller permutation of c, that produces the following added factor matrix

```
cr_min = [1 2 4 9];
ck_min = setdiff([cr ck],cr_min);
R = G(:,cr_min);
K = G(:,ck_min);
Lmin = mod(R\K,2)
```

```
Lmin = 4×4
     1     1     1     0
     1     0     0     1
     0     1     0     1
     0     0     1     1
```

Which is equivalent to the following added factors $c_k^* = (3, 5, 9, 14)$. This matrix is the only one that is rL-smaller than our matrix of added factors. By looking at all permutations we see that the following set of new basic factors yield the Lmin matrix.

```
M = [[1 2 4 9];[1 2 5 8];[1 3 4 8];[1 3 5 9]];
T = array2table(M)
```

T = 4×4 table

	M1	M2	M3	M4
1	1	2	4	9
2	1	2	5	8
3	1	3	4	8
4	1	3	5	9

And that none of them contains the last added factor (15).