# Package: MFRCD (via r-universe)

November 7, 2026

**Type** Package

**Title** Optimal Row-Column Designs for Asymmetrical Factorial Experiments

**Version** 0.1.1

**Maintainer** Sukanta Dash <sukanta.iasri@gmail.com>

**Description** Constructs and analyzes optimal row-column designs for mixed-level factorial experiments under square and rectangular field layouts. For square field layouts, the package implements direct common-factor constructions by first forming two component treatment arrays, one for each factor or super-factor, and then combining them through a symbolic cell-wise product following Gopinath, Parsad and Mandal (2018) <doi:10.1080/03610926.2017.1376091>. For rectangular field layouts, the package constructs designs by extracting a balanced principal block from a mixed-level block design, treating it as the principal column, taking the complete treatment set as the principal row, and generating the full row-column design by cyclic modular development. The package also includes repair utilities for improving disconnected or partially connected row-column designs through bounded treatment-swap searches while preserving the row-column layout structure. The package provides diagnostic tools for connectedness, orthogonal factorial structure, balance, estimability, and selected optimality criteria for row-column designs.

**License** GPL-3

**Encoding** UTF-8

**Depends** R (>= 4.0.0)

**Imports** stats

**Suggests** mixedfact, roxygen2, testthat (>= 3.0.0)

**RoxygenNote** 7.3.3

**Config/testthat/edition** 3

**NeedsCompilation** no

**Author** Archana A [aut], Sukanta Dash [aut, cre]

**Repository** https://sukantaiasri.r-universe.dev

**Date/Publication** 2026-05-11 08:39:33 UTC

**RemoteUrl** https://github.com/cran/MFRCD

**RemoteRef** HEAD

**RemoteSha** d38bf6e5f1b95a85a94194453bc480d8efccc72e

# Contents

analyse_rcd . . . 2
check_rc_factorial_n . . . 3
make_connected_rcd_bounded . . . 4
make_D2_from_complete_sets . . . 5
mfrcd . . . 6
mfrcd_check_repair . . . 7
print.mfrcd_check_repair . . . 9
print.rcd_analysis . . . 9
rc_factorial_n . . . 10
rc_from_principal_block . . . 11
rc_method1 . . . 12
rc_method2 . . . 12
rcd_analysis_summary . . . 13
rcd_feasibility_check . . . 14
summarize_rcd_layout . . . 15
tetra_connectedness . . . 15
validate_bounded_rcd . . . 16
verify_mfrcd . . . 17

**Index** **19**

---

`analyse_rcd` *Analyze a factorial row-column design*

---

## Description

Analyze a factorial row-column design.

## Usage

```
analyse_rcd(
  design,
  factor_levels = NULL,
  input = c("auto", "number", "factor_label"),
  sep = "",
  tol = 1e-08
)
```

### Arguments

| | |
|---|---|
| design | A row-column design matrix. |
| factor_levels | Integer vector of factor levels. |
| input | Coding used by the supplied design. |
| sep | Character separator used in treatment labels. |
| tol | Numerical tolerance. |

### Value

A list, matrix, data frame, or classed object containing the requested row-column design information.

### Examples

```
D <- matrix(c(1, 2, 3, 4), nrow = 2)
res <- analyse_rcd(D, factor_levels = c(2, 2), input = "number")
stopifnot(inherits(res, "rcd_analysis"))
```

---

## check_rc_factorial_n *Check basic row-column factorial design properties*

---

### Description

Check basic row-column factorial design properties.

### Usage

```
check_rc_factorial_n(D)
```

### Arguments

| | |
|---|---|
| D | A design object or design matrix. |

### Value

A list, matrix, data frame, or classed object containing the requested row-column design information.

### Examples

```
ex <- rc_factorial_n(c(2, 2))
chk <- check_rc_factorial_n(ex)
stopifnot(is.list(chk))
```

`make_connected_rcd_bounded` *Repair a row-column design under bounded swaps*

## Description

Repair a row-column design under bounded swaps.

## Usage

```
make_connected_rcd_bounded(
  design,
  factor_levels,
  input = c("factor_label", "number", "auto"),
  sep = "",
  tol = 1e-08,
  replication = "auto",
  horizontal_reps = 1,
  vertical_reps = 1,
  max_per_row = horizontal_reps,
  max_per_column = vertical_reps,
  max_iter = 100,
  max_candidates = NULL,
  seed = NULL,
  verbose = FALSE
)
```

## Arguments

| | |
|---|---|
| `design` | A row-column design matrix. |
| `factor_levels` | Integer vector of factor levels. |
| `input` | Coding used by the supplied design. |
| `sep` | Character separator used in treatment labels. |
| `tol` | Numerical tolerance. |
| `replication` | Replication target or "auto". |
| `horizontal_reps` | Maximum horizontal replication bound. |
| `vertical_reps` | Maximum vertical replication bound. |
| `max_per_row` | Maximum allowed occurrences of a treatment in a row. |
| `max_per_column` | Maximum allowed occurrences of a treatment in a column. |
| `max_iter` | Maximum number of repair iterations. |
| `max_candidates` | Maximum number of candidate swaps to examine. |
| `seed` | Optional random seed. |
| `verbose` | Logical; if TRUE, emit progress or summary messages. |

## Value

A list, matrix, data frame, or classed object containing the requested row-column design information.

## Examples

```
D <- matrix(c(1, 2, 3, 4), nrow = 2)
res <- make_connected_rcd_bounded(
  D, factor_levels = c(2, 2), input = "number",
  max_iter = 1, max_candidates = 5
)
stopifnot(is.list(res))
```

---

# make_D2_from_complete_sets

*Create a D2 design from complete treatment sets*

---

## Description

Create a D2 design from complete treatment sets.

## Usage

```
make_D2_from_complete_sets(
  design,
  factor_levels,
  input = c("factor_label", "number", "auto"),
  sep = "",
  prefer_margin = c("rows", "columns"),
  tol = 1e-08,
  max_solutions_checked = 100000L,
  treatment_universe = c("auto", "factorial", "observed")
)
```

## Arguments

| | |
|---|---|
| `design` | A row-column design matrix. |
| `factor_levels` | Integer vector of factor levels. |
| `input` | Coding used by the supplied design. |
| `sep` | Character separator used in treatment labels. |
| `prefer_margin` | Preferred margin for complete-set reshuffling. |
| `tol` | Numerical tolerance. |
| `max_solutions_checked` | Maximum number of candidate partitions to check. |
| `treatment_universe` | Treatment universe used for connectedness and repair. |

## Value

A list, matrix, data frame, or classed object containing the requested row-column design information.

## Examples

```
D <- mfrcd(c(2, 2))
res <- make_D2_from_complete_sets(D, factor_levels = c(2, 2))
stopifnot(is.list(res))
```

---

`mfrcd` *Generate a mixed-level factorial row-column design*

---

## Description

Generate a mixed-level factorial row-column design.

## Usage

```
mfrcd(
  levels,
  block_size = NULL,
  sep = "",
  details = FALSE,
  check_properties = FALSE,
  use_mixedfact = TRUE,
  mixedfact_method = "auto",
  choose_block = "best",
  group1 = NULL,
  group2 = NULL,
  prefer = c("auto", "direct", "mixedfact"),
  tol = 1e-07
)
```

## Arguments

| | |
|---|---|
| `levels` | Integer vector of factor levels. |
| `block_size` | Block size used in the construction. |
| `sep` | Character separator used in treatment labels. |
| `details` | Logical; if TRUE, return construction details. |
| `check_properties` | Logical; if TRUE, compute design-property diagnostics. |
| `use_mixedfact` | Logical; if TRUE, allow the mixedfact-based construction route. |
| `mixedfact_method` | Method passed to mixedfact when that route is used. |

| | |
|---|---|
| choose_block | Rule for choosing a block when several are available. |
| group1 | Optional integer vector of factor indices in the first group. |
| group2 | Optional integer vector of factor indices in the second group. |
| prefer | Construction route preference. |
| tol | Numerical tolerance. |

## Value

A list, matrix, data frame, or classed object containing the requested row-column design information.

## Examples

```
d <- mfrcd(c(2, 2))
stopifnot(is.matrix(d))
```

---

# mfrcd_check_repair *Generate, check, and optionally repair an MFRCD design*

---

## Description

Generate, check, and optionally repair an MFRCD design.

## Usage

```
mfrcd_check_repair(
  levels,
  block_size = NULL,
  sep = "",
  use_mixedfact = TRUE,
  mixedfact_method = "auto",
  choose_block = "best",
  group1 = NULL,
  group2 = NULL,
  prefer = c("auto", "direct", "mixedfact"),
  tol = 1e-08,
  repair_if_disconnected = TRUE,
  replication = "auto",
  horizontal_reps = NULL,
  vertical_reps = NULL,
  max_per_row = NULL,
  max_per_column = NULL,
  max_iter = 100,
  max_candidates = NULL,
  seed = NULL,
  complete_set_prefer = c("rows", "columns"),
```

```
  max_complete_set_search = 100000L,
  treatment_universe = c("auto", "factorial", "observed"),
  verbose = FALSE
)
```

## Arguments

| | |
|---|---|
| levels | Integer vector of factor levels. |
| block_size | Block size used in the construction. |
| sep | Character separator used in treatment labels. |
| use_mixedfact | Logical; if TRUE, allow the mixedfact-based construction route. |
| mixedfact_method | Method passed to mixedfact when that route is used. |
| choose_block | Rule for choosing a block when several are available. |
| group1 | Optional integer vector of factor indices in the first group. |
| group2 | Optional integer vector of factor indices in the second group. |
| prefer | Construction route preference. |
| tol | Numerical tolerance. |
| repair_if_disconnected | Logical; if TRUE, attempt repair when the design is disconnected. |
| replication | Replication target or "auto". |
| horizontal_reps | Maximum horizontal replication bound. |
| vertical_reps | Maximum vertical replication bound. |
| max_per_row | Maximum allowed occurrences of a treatment in a row. |
| max_per_column | Maximum allowed occurrences of a treatment in a column. |
| max_iter | Maximum number of repair iterations. |
| max_candidates | Maximum number of candidate swaps to examine. |
| seed | Optional random seed. |
| complete_set_prefer | Preferred direction for complete-set reshuffling. |
| max_complete_set_search | Maximum number of complete-set partitions to inspect. |
| treatment_universe | Treatment universe used for connectedness and repair. |
| verbose | Logical; if TRUE, emit progress or summary messages. |

## Value

A list, matrix, data frame, or classed object containing the requested row-column design information.

## Examples

```
res <- mfrcd_check_repair(c(2, 2), repair_if_disconnected = FALSE, max_iter = 1, max_candidates = 5)
stopifnot(inherits(res, "mfrcd_check_repair"))
```

## print.mfrcd_check_repair *Print an MFRCD check-and-repair result*

### Description

Print an MFRCD check-and-repair result.

### Usage

```
## S3 method for class 'mfrcd_check_repair'
print(x, ...)
```

### Arguments

| | |
|---|---|
| x | An object. |
| ... | Additional arguments passed to or from methods. |

### Value

Invisibly returns ‘x‘.

### Examples

```
res <- mfrcd_check_repair(c(2, 2), repair_if_disconnected = FALSE, max_iter = 1, max_candidates = 5)
print(res)
```

## print.rcd_analysis *Print a row-column design analysis*

### Description

Print a row-column design analysis.

### Usage

```
## S3 method for class 'rcd_analysis'
print(x, digits = 6, ...)
```

### Arguments

| | |
|---|---|
| x | An object. |
| digits | Number of digits used for printing. |
| ... | Additional arguments passed to or from methods. |

## Value

Invisibly returns ‘x‘.

## Examples

```
D <- matrix(c(1, 2, 3, 4), nrow = 2)
res <- analyse_rcd(D, factor_levels = c(2, 2), input = "number")
print(res)
```

---

`rc_factorial_n` *Construct a mixed-level factorial row-column design*

---

## Description

Construct a mixed-level factorial row-column design.

## Usage

```
rc_factorial_n(levels, sep = "", details = FALSE, group1 = NULL, group2 = NULL)
```

## Arguments

| | |
|---|---|
| `levels` | Integer vector of factor levels. |
| `sep` | Character separator used in treatment labels. |
| `details` | Logical; if TRUE, return construction details. |
| `group1` | Optional integer vector of factor indices in the first group. |
| `group2` | Optional integer vector of factor indices in the second group. |

## Value

A list, matrix, data frame, or classed object containing the requested row-column design information.

## Examples

```
ex <- rc_factorial_n(c(2, 2))
stopifnot(is.matrix(ex))
```

rc_from_principal_block *Construct a row-column design from a principal block*

## Description

Construct a row-column design from a principal block.

## Usage

```
rc_from_principal_block(
  levels,
  block_size,
  principal_block = NULL,
  use_mixedfact = TRUE,
  mixedfact_method = "auto",
  choose_block = "best",
  sep = "",
  details = FALSE
)
```

## Arguments

| | |
|---|---|
| levels | Integer vector of factor levels. |
| block_size | Block size used in the construction. |
| principal_block | Optional principal block of treatment labels. |
| use_mixedfact | Logical; if TRUE, allow the mixedfact-based construction route. |
| mixedfact_method | Method passed to mixedfact when that route is used. |
| choose_block | Rule for choosing a block when several are available. |
| sep | Character separator used in treatment labels. |
| details | Logical; if TRUE, return construction details. |

## Value

A list, matrix, data frame, or classed object containing the requested row-column design information.

## Examples

```
pb <- matrix(c(0, 0, 0, 1), ncol = 2, byrow = TRUE)
ex <- rc_from_principal_block(c(2, 2), block_size = 2, principal_block = pb, details = TRUE)
stopifnot(is.matrix(ex$design))
```

## rc_method1 *Construct a two-factor row-column design using Method 1*

### Description

Construct a two-factor row-column design using Method 1.

### Usage

```
rc_method1(s1, s2, f = NULL, sep = "")
```

### Arguments

| | |
|---|---|
| s1 | Number of levels for the first factor or factor group. |
| s2 | Number of levels for the second factor or factor group. |
| f | Common factor used by Method 1. |
| sep | Character separator used in treatment labels. |

### Value

A list, matrix, data frame, or classed object containing the requested row-column design information.

### Examples

```
ex <- rc_method1(2, 4)
stopifnot(is.matrix(ex$design))
```

## rc_method2 *Construct a two-factor row-column design using Method 2*

### Description

Construct a two-factor row-column design using Method 2.

### Usage

```
rc_method2(s1, s2, sep = "")
```

### Arguments

| | |
|---|---|
| s1 | Number of levels for the first factor or factor group. |
| s2 | Number of levels for the second factor or factor group. |
| sep | Character separator used in treatment labels. |

## Value

A list, matrix, data frame, or classed object containing the requested row-column design information.

## Examples

```
ex <- rc_method2(2, 4)
stopifnot(is.matrix(ex$design))
```

---

`rcd_analysis_summary` *Summarize row-column design analysis*

---

## Description

Summarize row-column design analysis.

## Usage

```
rcd_analysis_summary(ans)
```

## Arguments

| | |
|---|---|
| `ans` | Analysis result. |

## Value

A list, matrix, data frame, or classed object containing the requested row-column design information.

## Examples

```
D <- matrix(c(1, 2, 3, 4), nrow = 2)
ans <- analyse_rcd(D, factor_levels = c(2, 2), input = "number")
res <- rcd_analysis_summary(ans)
stopifnot(is.data.frame(res))
```

`rcd_feasibility_check` *Check feasibility of a row-column design*

## Description

Check feasibility of a row-column design.

## Usage

```
rcd_feasibility_check(
  design,
  factor_levels,
  input = c("factor_label", "number", "auto"),
  sep = "",
  tol = 1e-08
)
```

## Arguments

| | |
|---|---|
| `design` | A row-column design matrix. |
| `factor_levels` | Integer vector of factor levels. |
| `input` | Coding used by the supplied design. |
| `sep` | Character separator used in treatment labels. |
| `tol` | Numerical tolerance. |

## Value

A list, matrix, data frame, or classed object containing the requested row-column design information.

## Examples

```
D <- matrix(c(1, 2, 3, 4), nrow = 2)
res <- rcd_feasibility_check(D, factor_levels = c(2, 2), input = "number")
stopifnot(is.list(res))
```

## summarize_rcd_layout *Summarize a row-column design layout*

### Description

Summarize a row-column design layout.

### Usage

```
summarize_rcd_layout(
  design,
  factor_levels,
  input = c("factor_label", "number", "auto"),
  sep = ""
)
```

### Arguments

| | |
|---|---|
| design | A row-column design matrix. |
| factor_levels | Integer vector of factor levels. |
| input | Coding used by the supplied design. |
| sep | Character separator used in treatment labels. |

### Value

A list, matrix, data frame, or classed object containing the requested row-column design information.

### Examples

```
D <- matrix(c(1, 2, 3, 4), nrow = 2)
res <- summarize_rcd_layout(D, factor_levels = c(2, 2), input = "number")
stopifnot(is.list(res))
stopifnot(is.data.frame(res$total_replication_by_treatment))
```

## tetra_connectedness *Check connectedness of a row-column design*

### Description

Check connectedness of a row-column design.

### Usage

```
tetra_connectedness(
  design,
  factor_levels = NULL,
  input = c("auto", "number", "factor_label"),
  sep = "",
  tol = 1e-08
)
```

### Arguments

| | |
|---|---|
| design | A row-column design matrix. |
| factor_levels | Integer vector of factor levels. |
| input | Coding used by the supplied design. |
| sep | Character separator used in treatment labels. |
| tol | Numerical tolerance. |

### Value

A list, matrix, data frame, or classed object containing the requested row-column design information.

### Examples

```
D <- matrix(c(1, 2, 3, 4), nrow = 2)
res <- tetra_connectedness(D, factor_levels = c(2, 2), input = "number")
stopifnot(is.logical(res$connected))
```

---

## validate_bounded_rcd *Validate row-column replication bounds*

---

### Description

Validate row-column replication bounds.

### Usage

```
validate_bounded_rcd(
  design,
  factor_levels,
  input = c("factor_label", "number", "auto"),
  sep = "",
  replication = "auto",
  max_per_row = 1,
  max_per_column = 1
)
```

### Arguments

| | |
|---|---|
| design | A row-column design matrix. |
| factor_levels | Integer vector of factor levels. |
| input | Coding used by the supplied design. |
| sep | Character separator used in treatment labels. |
| replication | Replication target or "auto". |
| max_per_row | Maximum allowed occurrences of a treatment in a row. |
| max_per_column | Maximum allowed occurrences of a treatment in a column. |

### Value

A list, matrix, data frame, or classed object containing the requested row-column design information.

### Examples

```
D <- matrix(c(1, 2, 3, 4), nrow = 2)
res <- validate_bounded_rcd(D, factor_levels = c(2, 2), input = "number")
stopifnot(is.list(res))
```

---

## verify_mfrcd *Verify a mixed-level factorial row-column design*

---

### Description

Verify a mixed-level factorial row-column design.

### Usage

```
verify_mfrcd(
  x,
  levels = NULL,
  sep = "",
  tol = 1e-07,
  max_order = NULL,
  verbose = FALSE
)
```

### Arguments

| | |
|---|---|
| x | An object. |
| levels | Integer vector of factor levels. |
| sep | Character separator used in treatment labels. |
| tol | Numerical tolerance. |
| max_order | Maximum factorial effect order to include. |
| verbose | Logical; if TRUE, emit progress or summary messages. |

## Value

A list, matrix, data frame, or classed object containing the requested row-column design information.

## Examples

```
d <- mfrcd(c(2, 2), details = TRUE, check_properties = FALSE)
res <- verify_mfrcd(d, levels = c(2, 2))
stopifnot(is.list(res))
```

# Index

analyse_rcd, 2

check_rc_factorial_n, 3

make_connected_rcd_bounded, 4
make_D2_from_complete_sets, 5
mfrcd, 6
mfrcd_check_repair, 7

print.mfrcd_check_repair, 9
print.rcd_analysis, 9

rc_factorial_n, 10
rc_from_principal_block, 11
rc_method1, 12
rc_method2, 12
rcd_analysis_summary, 13
rcd_feasibility_check, 14

summarize_rcd_layout, 15

tetra_connectedness, 15

validate_bounded_rcd, 16
verify_mfrcd, 17