

Package ‘syncons’

August 5, 2026

Title Efficient and Low-Cost Construction of Synthetic Communities

Version 0.1.1

Description Provides functions and a Shiny application to design plate layouts for constructing synthetic communities (SynComs) from a pool of strains using 24-, 96- and 384-well microplates. The method generates all possible strain combinations in a scalable, low-cost workflow suitable for high-throughput experiments.
Reference: Tian et al. (2024) <[doi:10.21769/BioProtoc.2405576](https://doi.org/10.21769/BioProtoc.2405576)>.

URL <https://github.com/gaospecial/syncons>,
<https://doi.org/10.21769/BioProtoc.2405576>,
<https://bio-spring.shinyapps.io/SynComsConstructor/>

BugReports <https://github.com/gaospecial/syncons/issues>

License GPL (>= 3)

Encoding UTF-8

RoxygenNote 7.3.3

Imports dplyr, DT, glue, purrr, shiny, stringr, tibble, tidyr, utils

NeedsCompilation no

Author Chun-Hui Gao [aut, cre] (ORCID:
<<https://orcid.org/0000-0002-1445-7939>>)

Maintainer Chun-Hui Gao <gaospecial@gmail.com>

Depends R (>= 4.1.0)

Repository CRAN

Date/Publication 2026-08-05 17:30:02 UTC

Contents

assign_plate	2
combinations	2
one_plate	3
run_shinyApp	4
sort_combination	4
strain_combination	5

Index	6
--------------	----------

assign_plate	<i>Assign multiple strains to one or more specified plates</i>
--------------	--

Description

Assign multiple strains to one or more specified plates

Usage

```
assign_plate(  
  strains,  
  strain_sep = "/",  
  plate_type = c("24", "96", "384"),  
  plate_prefix = "P",  
  return_layout = FALSE  
)
```

Arguments

- strains a character vector specifying the strain names
- strain_sep separator used to join strains into combination
- plate_type specify plate type, either 24, 96, or 384.
- plate_prefix plate prefix
- return_layout return the layout of plate, instead of a sample table if TRUE

Value

a tibble

Examples

```
assign_plate(letters[1:5])
```

combinations	<i>all possible combinations of n sets</i>
--------------	--

Description

all possible combinations of n sets

Usage

```
combinations(n)
```

Arguments

- n dim

one_plate

*The composition of community in a single 24-, 96- or 384-well plate***Description**

We use a fast, extensible method to generate multiple SynComs in every possible composition. The smallest unit is either a 4 x 4 plate (24 well) for 16 SynComs, a 8x8 plate (96 well) for 64 SynComs, or a 16 x 16 plate (384 well) for 256 SynComs. Based on this, one can generate as many as plates, and in that cases the combination of SynComs could be multiplied in batch. For each time we add an extra strain, the number of total plates will double to fulfill all additional strain combinations. In other words, the possible combination of n strain equals to 2^n .

Usage

```
one_plate(
  plate_type = c("24", "96", "384"),
  plate_name = NULL,
  strain_base = NULL,
  strain_add = NULL,
  strain_sep = "/",
  return_layout = FALSE
)
```

Arguments

plate_type	specify plate type, either 24, 96, or 384.
plate_name	will be used as the prefix in combination id
strain_base	the base strain or community in a single plate
strain_add	the id of added strains
strain_sep	separator used to join strains into combination
return_layout	return the layout of plate, instead of a sample table if TRUE

Value

a tibble

Examples

```
one_plate("24")
one_plate("96")
one_plate("384")
```

run_shinyApp	<i>Run Shiny App locally</i>
--------------	------------------------------

Description

Run Shiny App locally

Usage

```
run_shinyApp()
```

Value

a shiny app

sort_combination	<i>Sort combination in a string</i>
------------------	-------------------------------------

Description

String is used as combination name. However, the combination of A and B can be "A/B" or "B/A", this would make it difficult in comparison of combination names. By using this function, the name of strains included in a combination will be sorted alphabetically, so that the combination names can be identical and human-friendly.

Usage

```
sort_combination(strain_combination, strain_sep = "/")
```

Arguments

strain_combination	
	a string or a character vector
strain_sep	separator

Value

a string

Examples

```
sort_combination("A1/A11/A2/B3/D5/C11")  
sort_combination(c("A1/A11/A2/B3/D5/C11", "A1/A11/A25/B3/D18/C11") )
```

strain_combination	<i>All the possible combination of multiple strains</i>
--------------------	---

Description

All the possible combination of multiple strains

Usage

```
strain_combination(strains, strain_sep = "/")
```

Arguments

strains	names of strains, can be any integers
strain_sep	separator used to join strains into combination

Value

a character vector

Examples

```
strain_combination(c("A","B","C"))
```

Index

`assign_plate`, [2](#)

`combinations`, [2](#)

`one_plate`, [3](#)

`run_shinyApp`, [4](#)

`sort_combination`, [4](#)

`strain_combination`, [5](#)