# Package 'rkappa'

June 9, 2012

**Type** Package

**Title** Kappa simulation Project

**Version** 1.0.120609

**Date** 2012-06-07

**Author** Anatoly Sorokin

**Maintainer** Anatoly Sorokin <lptolik@gmail.com>

**Description** The package provides a infrastructure required to simulate kappa model with various parameter sets and analyse results of such simulations.

**License** GPL (>= 2)

**Copyright** Anatoly Sorokin 2012

**Suggests** randtoolbox,gdata,sensitivity,igraph

## R topics documented:

1

---

rkappa-package                      *Kappa simulation Project*

---

## Description

The package provides a infrastructure required to simulate kappa model with various parameter sets and analyse results of such simulations.

## Details

| | |
|---|---|
| Package: | rkappa |
| Type: | Package |
| Title: | Kappa simulation Project |
| Version: | 1.0.120609 |
| Date: | 2012-06-07 |
| Author: | Anatoly Sorokin |
| Maintainer: | Anatoly Sorokin <lptolik@gmail.com> |
| License: | GPL (>= 2) |
| Copyright: | Anatoly Sorokin 2012 |
| Suggests: | randtoolbox,gdata,sensitivity,igraph |

## Author(s)

Anatoly Sorokin

---

concurrent.sensitivity

*concurrent sensitivity*

---

## Description

function calculate PRCC sensitivity coefficients of the model parameters

## Usage

```
concurrent.sensitivity(res, obsSens, time = max(obsSens$time),
    outName = "Prot[0-9]+", nboot = 0)
```

## Arguments

| | |
|---|---|
| res | data frame of parameter sets. Each row is a set, each column is a parameter |
| obsSens | data frame of observables obtained in the model simulations |
| time | time point at which sensitiviti is going to be calculated |
| outName | name of observable to calculate sensitivity |
| nboot | number of bootstrap runs |

**Value**

list of parameter sensitivities

**Author(s)**

Anatoly Sorokin

---

makeIGraph                     *makeIGraph*

---

**Description**

function converts kappa string, defining the complex structure into iGraph graph

**Usage**

```
makeIGraph(kappa)
```

**Arguments**

kappa            complex definition in kappa language

**Value**

simplified iGraph graph of the complex

**Author(s)**

Anatoly Sorokin

---

makeSIF                        *makeSIF*

---

**Description**

function creates SIF (Simple Interaction Format) from kappa string

**Usage**

```
makeSIF(kappa)
```

**Arguments**

kappa            kappa string defining the complex structure

**Value**

simplified iGraph graph of the complex

**Author(s)**

Anatoly Sorokin

---

prepareProject                      *prepareProject*

---

### Description

Creates a infrastructure required to simulate kappa model with various parameter sets and generate correspondent folder infrastructure

### Usage

```
prepareProject(project = paste("multi", format(Sys.time(), "%Y%m%d%H%M%S"),
    sep = ""), numSets = 500, pTable = NA, constantfiles = c("main_rnap_def_rule.ka",
    "main_rnap_init.ka", "main_rnap_param.ka"), templatefiles = c("var_prom_def.ka",
    "var_prom_init.ka", "var_prom_rule.ka"), paramfile = c("var_prom_param.ka"),
    k_min = 0.1, k_max = 10, exec.path = "~/kasim3/KaSim", shFile = "run.sh.templ",
    jobFile = "job.sh.templ", jobCFile = "jobConc.sh", repReg = "_-",
    type = c("parallel", "concurrent", "both"))
```

### Arguments

| | |
|---|---|
| project | name of the project to be created, new folder will be created to contain the project files |
| numSets | number of parameter sets to be generated |
| pTable | Parameter ranges data frame. Should contain columns param with parameter names, Min and Max with parameter ranges. Names in param column should match names in the content of paramfile files. |
| constantfiles | list of file names containing constant part of the model |
| templatefiles | list of template file names to be used to create model |
| paramfile | list of parameter file names |
| k_min | minimum parameter value factor to be used if pTable==NA |
| k_max | maximum parameter value factor to be used if pTable==NA |
| exec.path | path to kappa language simulator executables in simulation environment |
| shFile | run script template file name |
| jobFile | job management job file template |
| jobCFile | job management job template file for concurrent simulations |
| repReg | regular expression to be replaced with number of parameter set |
| type | type of the project |

### Value

project object

### Author(s)

Anatoly Sorokin

| print.kproject | *print kproject* |
|---|---|

### Description

Kproject class print method

### Usage

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

### Arguments

x              object to print

...

### Author(s)

Anatoly Sorokin

| print.summary.kproject | |
|---|---|
| | *print summary kproject* |

### Description

Kproject.summary print method

### Usage

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

### Arguments

x              object to print

...

### Author(s)

Anatoly Sorokin

| project | *\it E. coli promoter transcription initiation model project* |
|---|---|

**Description**

The project is build around \it E. coli promoter transcription initiation model. The purpose of the project is to compare sensitivity of parameters of the model in parallel and concurrent regimes.

**Usage**

```
data(project)
```

**Format**

The format is: List of 13 $ name : chr "lot.of.dna1" $ date : chr "20120607181510" $ const-Lines :List of 3 ..$ main_rnap_def_rule.ka: chr [1:27] "#basic promoter part description" " ..$ main_rnap_init.ka : chr [1:3] " ..$ main_rnap_param.ka : chr [1:7] " $ templateLines:List of 3 ..$ var_prom_def.ka : chr [1:15] "#variable promoter part description" " ..$ var_prom_init.ka: chr [1:2] " ..$ var_prom_rule.ka: chr [1:23] "#var_prom_rule.ka" "'promoter_- nonspec' RNAP (dna,st~f),promoter_- (ns,s35,s10~f)-> RNAP (dna!1,st~n),promoter_- (ns!1,s35,s10~f) @ ('kns'*'pns_-'"| __truncated__ "'prom_- ns diss' RNAP (dna!1,st~n),promoter_- (ns!1,s35,s10~f) -> RNAP (dna,st~f),promoter_- (ns,s35,s10~f) @ ('k_ns'*'p_ns_-')"| __truncated__ "'recognitionUp_-' RNAP (sigmaS!_,dna!1,st~n,alpha),promoter_- (ns!1,up) -> RNAP (sigmaS!_,dna!1,st~pr,alpha!2),promoter_- (ns!"| __truncated__ ... $ replaceRegexp: chr "_-" $ nRep : num 10 $ nSets : num 500 $ execPath : chr "/exports/home/osorokin/kasim3/KaSim" $ type : chr "both" $ shLines :List of 3 ..$ run.sh.templ: chr [1:22] "#!/bin/bash" "numEv=10" "time=1000" "if [ \"$1\" != \"\" ]; then" ... ..$ jobConc.sh : chr [1:21] "#!/bin/bash" "###############################" "# #" "# SGE MPI job script for ECDF Cluster #" ... ..$ job.sh.templ: chr [1:21] "#!/bin/bash" "######################################" "# #" "# SGE MPI job script for ECDF Cluster #" ... $ paramLines :List of 1 ..$ var_prom_param.ka: chr [1:19] "#var_prom_param.ka" " $ pTable :'data.frame': 18 obs. of 5 variables: ..$ i : num [1:18] 1 2 3 4 5 6 7 8 9 10 ... ..$ name: chr [1:18] "p_ns_-" "pns_-" "k35_-" "kb10_-" ... ..$ str : num [1:18] 2 3 4 5 6 7 8 9 10 11 ... ..$ min : num [1:18] 0.01 0.0001 0 0 1.7 0.67 0.05 3.6 0.14 0.35 ... ..$ max : num [1:18] 10000 100 0 0 170 67 5 360 14 35 ... $ paramSets :'data.frame': 500 obs. of 18 variables: ..$ p_ns : num [1:500] 2912 1153 9848 7942 1672 ... ..$ pns : num [1:500] 27.2 77.9 23.2 88.4 12 ... ..$ k35 : num [1:500] 0 0 0 0 0 0 0 0 0 0 ... ..$ kb10 : num [1:500] 0 0 0 0 0 0 0 0 0 0 ... ..$ k_35 : num [1:500] 169 100.9 30.9 140.3 70.4 ... ..$ k_10 : num [1:500] 42.61 29.55 62.98 4.86 37.78 ... ..$ kb_10 : num [1:500] 2.305 0.469 4.99 0.938 3.913 ... ..$ k35i : num [1:500] 220 113 292 352 176 ... ..$ kb10i : num [1:500] 6.55 3.42 8.81 11.41 4.29 ... ..$ k10 : num [1:500] 10.52 18.09 4.42 16.42 32.26 ... ..$ kelong : num [1:500] 0.763 0.22 0.615 0.37 0.964 ... ..$ ksynt : num [1:500] 93.9 19.3 105.4 40.3 127.7 ... ..$ kuntouch: num [1:500] 225 165 290 129 338 ... ..$ kUP : num [1:500] 177.8 127 68.6 190.4 40.7 ... ..$ k_UP : num [1:500] 32.1 275.4 138.8 188 51.1 ... ..$ kreset : num [1:500] 9.16e-04 3.60e-04 1.19e-03 1.46e-03 6.89e-05 ... ..$ kinit : num [1:500] 1.461 3.348 0.448 1.976 2.803 ... ..$ ksigma : num [1:500] 2.487 1.585 0.145 0.743 2.183 ... - attr(*, "class")= chr "kproject"

**Examples**

```
data(project)
## maybe str(project) ; plot(project) ...
```

---

read.observables   *read observables*

---

### Description

function to read and parse KaSim simulation results

### Usage

```
read.observables(file)
```

### Arguments

file    name of the observable file

### Value

data frame of observables

### Author(s)

Anatoly Sorokin

---

read.snapshot   *read snapshot*

---

### Description

function to read KaSim snapshot kappa files

### Usage

```
read.snapshot(file)
```

### Arguments

file    name of the snapshot file

### Author(s)

Anatoly Sorokin

---

| summary.kproject | *summary kproject* |
|---|---|

---

### Description

Kproject class summary method

### Usage

```
## S3 method for class 'kproject'
summary(x, ...)
```

### Arguments

| x | object to create summary for |
|---|---|
| ... | |

### Value

kproject.summary object

### Author(s)

Anatoly Sorokin

---

| timed.concurrent.sensitivity | |
|---|---|
| | *timed concurrent sensitivity* |

---

### Description

function calculate PRCC sensitivity coefficients of the model parameters along whole timeline of the model execution,

### Usage

```
timed.concurrent.sensitivity(res, obsSens, outName = "Prot[0-9]+",
    nboot = 0, plot = FALSE)
```

### Arguments

| res | data frame of parameter sets. Each row is a set, each column is a parameter |
|---|---|
| obsSens | data frame of observables obtained in the model simulations |
| outName | name of observable to calculate sensitivity |
| nboot | number of bootstrap runs |
| plot | flag should be true, if graphical representation is required |

**Value**

data frame of parameter PRCC coefficients at each time point

**Author(s)**

Anatoly Sorokin

# Index