

Package ‘diffeqr’

August 26, 2020

Type Package

Title Solving Differential Equations (ODEs, SDEs, DDEs, DAEs)

Version 1.0.0

Description An interface to 'DifferentialEquations.jl' <<https://diffeq.sciml.ai/dev/>> from the R programming language.

It has unique high performance methods for solving ordinary differential equations (ODE), stochastic differential equations (SDE), delay differential equations (DDE), differential-algebraic equations (DAE), and more. Much of the functionality, including features like adaptive time stepping in SDEs, are unique and allow for multiple orders of magnitude speedup over more common methods.

'diffeqr' attaches an R interface onto the package, allowing seamless use of this tooling by R users.

Depends R (>= 3.4.0)

Encoding UTF-8

License MIT + file LICENSE

URL <https://github.com/SciML/diffeqr>

LazyData true

SystemRequirements Julia (>= 1.5), DifferentialEquations.jl, ModelingToolkit.jl

Imports JuliaCall, stringr

RoxygenNote 7.1.1

Suggests testthat, knitr, rmarkdown

VignetteBuilder knitr

NeedsCompilation no

Author Christopher Rackauckas [aut, cre]

Maintainer Christopher Rackauckas <me@chrisrackauckas.com>

Repository CRAN

Date/Publication 2020-08-26 16:30:03 UTC

R topics documented:

diffeqgpu_setup	2
diffeq_setup	2
jitoptimize_ode	3
jitoptimize_sde	4

Index	5
--------------	----------

diffeqgpu_setup	<i>Setup DiffEqGPU</i>
-----------------	------------------------

Description

This function initializes the DiffEqGPU package for GPU-parallelized ensembles. The first time will be long since it includes precompilation.

Usage

```
diffeqgpu_setup()
```

Examples

```
## diffeq_setup() is time-consuming and requires Julia+DifferentialEquations.jl
degpu <- diffeqr::diffeqgpu_setup()
```

diffeq_setup	<i>Setup diffeqr</i>
--------------	----------------------

Description

This function initializes Julia and the DifferentialEquations.jl package. The first time will be long since it includes precompilation.

Usage

```
diffeq_setup(...)
```

Arguments

... Parameters are passed down to JuliaCall::julia_setup

Examples

```

## diffeq_setup() is time-consuming and requires Julia+DifferentialEquations.jl
diffeqr::diffeq_setup()

```

jitoptimize_ode *Jit Optimize an ODEProblem*

Description

This function JIT Optimizes and ODEProblem utilizing the Julia ModelingToolkit and JIT compiler.

Usage

```
jitoptimize_ode(de, prob)
```

Arguments

de	the current diffeqr environment
prob	an ODEProblem

Examples

```

## diffeq_setup() is time-consuming and requires Julia+DifferentialEquations.jl
de <- diffeqr::diffeq_setup()
f <- function(u,p,t) {
    du1 = p[1]*(u[2]-u[1])
    du2 = u[1]*(p[2]-u[3]) - u[2]
    du3 = u[1]*u[2] - p[3]*u[3]
    return(c(du1,du2,du3))
}
u0 <- c(1.0,0.0,0.0)
tspan <- c(0.0,100.0)
p <- c(10.0,28.0,8/3)
prob <- de$ODEProblem(f, u0, tspan, p)
fastprob <- diffeqr::jitoptimize_ode(de,prob)
sol <- de$solve(fastprob,de$Tsit5())

```

jitoptimize_sde *Jit Optimize an SDEProblem*

Description

This function JIT Optimizes and SDEProblem utilizing the Julia ModelingToolkit and JIT compiler.

Usage

```
jitoptimize_sde(de, prob)
```

Arguments

de	the current diffeqr environment
prob	an SDEProblem

Examples

```
## diffeq_setup() is time-consuming and requires Julia+DifferentialEquations.jl  
diffeqr::diffeq_setup()
```

Index

`diffeq_setup`, [2](#)
`diffeqgpu_setup`, [2](#)
`jitoptimize_ode`, [3](#)
`jitoptimize_sde`, [4](#)