

# Package ‘dfsaneacc’

May 13, 2024

**Title** Accelerated Derivative-Free Method for Large-Scale Nonlinear Systems of Equations

**Version** 1.0.3

**Date** 2024-05-13

**Author** Ernesto G. Birgin [aut, ctb],  
John L. Gardenghi [aut, cre, ctb, trl],  
Diaulas S. Marcondes [aut, ctb, trl],  
Jose M. Martinez [aut, ctb]

**Maintainer** John L. Gardenghi <john.gardenghi@unb.br>

**NeedsCompilation** yes

**Description** Secant acceleration applied to derivative-free Spectral Residual Methods for solving large-scale nonlinear systems of equations. The main reference follows: E. G. Birgin and J. M. Martinez (2022) <doi:10.1137/20M1388024>.

**License** GPL-3

## R topics documented:

dfsaneacc-package . . . . .	1
dfsaneacc . . . . .	2
<b>Index</b>	<b>5</b>

---

dfsaneacc-package	<i>Accelerated Derivative-Free Method for Large-Scale Nonlinear Systems of Equations</i>
-------------------	--

---

## Description

Secant acceleration applied to derivative-free Spectral Residual Methods for solving large-scale nonlinear systems of equations. The main reference follows: E. G. Birgin and J. M. Martinez (2022) <doi:10.1137/20M1388024>.

## Details

This package includes the function:

```
dfsaneacc function to solve large-scale nonlinear systems of
equations using a derivative-free approach with sequential
secant acceleration for spectral residual methods.
```

## Author(s)

Ernesto G. Birgin [aut, ctb], John L. Gardenghi [aut, cre, ctb, trl], Diaulas S. Marcondes [aut, ctb, trl], Jose M. Martinez [aut, ctb]

Maintainer: John L. Gardenghi <john.gardenghi@unb.br>

## References

J. Barzilai, and J. M. Borwein (1988), Two-point step size gradient methods, *IMA J Numerical Analysis*, 8, 141-148.

E. G. Birgin, J. M. Martinez (2022), Secant acceleration of sequential residual methods for solving large-scale nonlinear systems of equations, *SIAM Journal on Numerical Analysis*, 60(6), 3145-3180.

W. LaCruz, and M. Raydan (2003), Nonmonotone spectral methods for large-scale nonlinear systems, *Optimization Methods and Software*, 18, 583-599.

W. LaCruz, J. M. Martinez, and M. Raydan (2006), Spectral residual method without gradient information for solving large-scale nonlinear systems of equations, *Mathematics of Computation*, 75, 1429-1448.

M. Raydan (1997), Barzilai-Borwein gradient method for large-scale unconstrained minimization problem, *SIAM Journal on Optimization*, 7, 26-33.

---

dfsaneacc	<i>Accelerated derivative-free spectral residual method for nonlinear systems of equations</i>
-----------	--

---

## Description

Accelerated derivative-free algorithm to solve nonlinear systems of equations.

## Usage

```
dfsaneacc(x, evalr, nhlim = 6, epsf = 1e-06, maxit = Inf, iprint = -1, ...)
```

**Arguments**

<code>x</code>	initial estimate for the solution of the nonlinear system.
<code>evalr</code>	a function that computes the nonlinear system evaluated at a given point as parameter and return the evaluated value. See details below.
<code>nhlim</code>	an integer that determines how many previous iterates must be considered in the sequential secant acceleration step. The default is 6.
<code>epsf</code>	a real value determining the absolute convergence tolerance. The default is $1.0\text{e-}6$ . See details below.
<code>maxit</code>	an integer determining the maximum number of iterations. The default is Inf.
<code>iprint</code>	the output level. The default is -1. See details below.
<code>...</code>	represents additional arguments that must be passed to <code>evalr</code> .

**Details**

The function `dfsaneacc` implements sequential residual methods (La Cruz and Raydan 2003; La Cruz, Mart\’inez, and Raydan 2006) with sequential secant acceleration approach proposed by Birgin and Mart\’inez (2022).

Convergence of the algorithm is declared when  $\|F(x)\|_2^2 \leq \text{epsf}^2$ . The default value for `epsf` is  $1.0\text{e-}6$ .

The algorithm employ the function `evalr` to compute the value of the nonlinear system at a given point `x`. The function `evalr` must have the form `evalr (x, ...)`.

The function has four output levels, based on the value of the input parameter `iprint`: `iprint=-1` no output is generated, `iprint=0` means basic information at every iteration, `iprint=1` adds additional information related to the backtracking strategy, and `iprint=2` adds information related to the computation of the acceleration step. Its default value is `iprint=-1`.

**Value**

A list with

<code>x</code>	the final estimate to the solution.
<code>res</code>	the final nonlinear system value.
<code>normF</code>	the final nonlinear system value squared L2-norm.
<code>iter</code>	the total number of iterations.
<code>fcnt</code>	the total number of functional evaluations.
<code>istop</code>	an integer indicating the convergence type. Possible values are 0 for successful convergence (squared L2-norm of the residual) and 1 for maximum number of iterations exceeded.

**Author(s)**

Ernesto G. Birgin [aut, ctb], John L. Gardenghi [aut, cre, ctb, trl], Diaulas S. Marcondes [aut, ctb, trl], Jose M. Martinez [aut, ctb]

Maintainer: John L. Gardenghi <john.gardenghi@unb.br>

## References

- J. Barzilai, and J. M. Borwein (1988), Two-point step size gradient methods, *IMA J Numerical Analysis*, 8, 141-148.
- E. G. Birgin, J. M. Martínez (2022), Secant acceleration of sequential residual methods for solving large-scale nonlinear systems of equations, *SIAM Journal on Numerical Analysis*, 60(6), 3145-3180.
- W. LaCruz, and M. Raydan (2003), Nonmonotone spectral methods for large-scale nonlinear systems, *Optimization Methods and Software*, 18, 583-599.
- W. LaCruz, J. M. Martínez, and M. Raydan (2006), Spectral residual method without gradient information for solving large-scale nonlinear systems of equations, *Mathematics of Computation*, 75, 1429-1448.
- M. Raydan (1997), Barzilai-Borwein gradient method for large-scale unconstrained minimization problem, *SIAM Journal on Optimization*, 7, 26-33.

## Examples

```
n <- 3
x0 <- rep(1/n^2, n)

expfun2 <- function(x) {
  n <- length(x)
  f <- rep(NA, n)
  f[1] <- exp(x[1]) - 1.0
  f[2:n] <- (2:n)/10.0 * (exp(x[2:n]) + x[1:n-1] - 1)
  f
}

ret <- dfsaneacc(x=x0, evalr=expfun2, nhlim=6, epsf=1.0e-6*sqrt(n),
iprint=0)
ret
```

# Index

- \* **large-scale**

- dfsaneacc, [2](#)

- \* **package**

- dfsaneacc-package, [1](#)

dfsaneacc, [2](#)

dfsaneacc-package, [1](#)