Package: hf (via r-universe)

May 13, 2026

Title Head Loss Calculations for Pipelines

Version 0.0.0.9000

Description Provides simple and efficient functions to calculate head loss in pipes using standard hydraulic equations.

License MIT + file LICENSE

Depends R (>= 4.1.0)

Encoding UTF-8

Language en-US

Roxygen list(markdown = TRUE)

URL https://github.com/joaobtj/hf, https://joaobtj.github.io/hf/

BugReports https://github.com/joaobtj/hf/issues

Suggests bsicons, bslib, knitr, rmarkdown, shiny, testthat (>= 3.0.0)

Config/testthat/edition 3

VignetteBuilder knitr

Config/roxygen2/version 8.0.0

Repository https://joaobtj.r-universe.dev

Date/Publication 2026-05-11 17:27:31 UTC

RemoteUrl https://github.com/joaobtj/hf

RemoteRef HEAD

RemoteSha 219856cfac71849b3a026e5ec988cf77cdeb9106

Contents

---

calc_diameter_darcy — *Calculate Required Pipe Diameter using Darcy-Weisbach*

Description

Calculates the required internal pipe diameter iteratively given a specific head loss and flow rate. By taking a functional approach, it supports any friction factor calculation function.

Usage

```
calc_diameter_darcy(
  loss,
  length,
  flow,
  roughness,
  friction_fun = NULL,
  viscosity = 1.004e-06,
  gravity = 9.81
)
```

Arguments

`loss`	Numeric. The target friction head loss (meters).
`length`	Numeric. The length of the pipe (meters).
`flow`	Numeric. The volumetric flow rate (cubic meters per second).
`roughness`	Numeric. The absolute internal roughness of the pipe (meters). Required unless `friction_factor` is provided.
`friction_fun`	Function. A function to calculate the friction factor (must accept `reynolds`, `roughness`, and `diameter`). If `NULL` (the default), it uses `calc_friction_cw`.
`viscosity`	Numeric. Kinematic viscosity of the fluid (sq. meters per sec). Default is 1.004e-6.
`gravity`	Numeric. Acceleration due to gravity (meters per second squared). Default is 9.81.

Value

A numeric vector representing the required internal diameter in meters.

Examples

```
# Find diameter using the default Colebrook-White function
calc_diameter_darcy(loss = 5, length = 100, flow = 0.02, roughness = 0.00026)

# Find diameter by injecting Swamee-Jain
calc_diameter_darcy(
  loss = 5, length = 100, flow = 0.02, roughness = 0.00026,
  friction_fun = calc_friction_sj
)
```

---

calc_diameter_flamant *Calculate Required Pipe Diameter using Flamant*

Description

Calculates the required internal pipe diameter given a specific target head loss, length, flow rate, and Flamant roughness coefficient.

Usage

```
calc_diameter_flamant(loss, length, flow, coef = 0.000135)
```

Arguments

loss	Numeric. The target friction head loss (meters).
length	Numeric. The length of the pipe (meters).
flow	Numeric. The volumetric flow rate (cubic meters per second).
coef	Numeric. The Flamant roughness coefficient (b). Default is 0.000135, which is typical for smooth plastic pipes (e.g., PVC, PE).

Value

A numeric vector representing the required internal diameter in meters.

Examples

```
# Find diameter for a 50m pipe with 1.5m allowable head loss and 0.0002 m^3/s flow
calc_diameter_flamant(loss = 1.5, length = 50, flow = 0.0002)
```

calc_diameter_hw — *Calculate Required Pipe Diameter using Hazen-Williams*

Description

Calculates the required internal pipe diameter given a specific target head loss, length, flow rate, and roughness coefficient based on the Hazen-Williams (hw) equation.

Usage

```
calc_diameter_hw(loss, length, flow, coef = 140)
```

Arguments

`loss`	Numeric. The target friction head loss (meters).
`length`	Numeric. The length of the pipe (meters).
`flow`	Numeric. The volumetric flow rate (cubic meters per second).
`coef`	Numeric. The Hazen-Williams roughness coefficient (dimensionless). Default is 140, which is typical for PVC pipes.

Value

A numeric vector representing the required internal diameter in meters.

Examples

```
# Find diameter for a 100m pipe with 2m allowable head loss, 0.02 m^3/s flow (C = 140)
calc_diameter_hw(loss = 2, length = 100, flow = 0.02)
```

calc_flow_darcy — *Calculate Flow Rate using Darcy-Weisbach*

Description

Calculates the volumetric flow rate iteratively given a specific head loss. This function uses a functional programming approach, allowing the injection of any friction factor function (e.g., Colebrook-White).

Usage

```
calc_flow_darcy(
  loss,
  length,
  diameter,
  roughness,
  friction_fun = NULL,
  viscosity = 1.004e-06,
  gravity = 9.81
)
```

Arguments

`loss`	Numeric. The target friction head loss (meters).
`length`	Numeric. The length of the pipe (meters).
`diameter`	Numeric. The internal diameter of the pipe (meters).
`roughness`	Numeric. The absolute internal roughness of the pipe (meters). Required unless `friction_factor` is provided.
`friction_fun`	Function. A function to calculate the friction factor (must accept `reynolds`, `roughness`, and `diameter`). If `NULL` (the default), it uses `calc_friction_cw`.
`viscosity`	Numeric. Kinematic viscosity of the fluid (sq. meters per sec). Default is 1.004e-6.
`gravity`	Numeric. Acceleration due to gravity (meters per second squared). Default is 9.81.

Value

A numeric vector representing the volumetric flow rate in cubic meters per second.

Examples

```
# Default: Uses Colebrook-White to find the flow rate
calc_flow_darcy(loss = 5, length = 100, diameter = 0.1, roughness = 0.00026)

# Injecting Swamee-Jain function
calc_flow_darcy(
  loss = 5, length = 100, diameter = 0.1, roughness = 0.00026,
  friction_fun = calc_friction_sj
)
```

calc_flow_flamant *Calculate Flow Rate using Flamant*

Description

Calculates the volumetric flow rate in a pipe given a specific head loss, length, internal diameter, and Flamant coefficient.

Usage

```
calc_flow_flamant(loss, length, diameter, coef = 0.000135)
```

Arguments

loss	Numeric. The friction head loss (meters).
length	Numeric. The length of the pipe (meters).
diameter	Numeric. The internal diameter of the pipe (meters).
coef	Numeric. The Flamant roughness coefficient (b). Default is 0.000135, which is typical for smooth plastic pipes (e.g., PVC, PE).

Value

A numeric vector representing the volumetric flow rate in cubic meters per second.

Examples

```
# Find maximum flow rate for a 50m pipe with 15mm diameter and 1.5m head loss
calc_flow_flamant(loss = 1.5, length = 50, diameter = 0.015)
```

calc_flow_hw *Calculate Flow Rate using Hazen-Williams*

Description

Calculates the volumetric flow rate in a pipe given a specific head loss, length, internal diameter, and roughness coefficient based on the Hazen-Williams (hw) equation.

Usage

```
calc_flow_hw(loss, length, diameter, coef = 140)
```

Arguments

`loss`	Numeric. The friction head loss (meters).
`length`	Numeric. The length of the pipe (meters).
`diameter`	Numeric. The internal diameter of the pipe (meters).
`coef`	Numeric. The Hazen-Williams roughness coefficient (dimensionless). Default is 140, which is typical for PVC pipes.

Value

A numeric vector representing the volumetric flow rate in cubic meters per second.

Examples

```
# Find flow rate for a 100m pipe with 0.1m diameter, C = 140, and 2m head loss
calc_flow_hw(loss = 2, length = 100, diameter = 0.1)
```

`calc_friction_blasius` *Calculate Friction Factor using Blasius*

Description

Calculates the Darcy friction factor using the empirical Blasius equation. This formula is highly accurate for smooth pipes (e.g., PVC, glass) and turbulent flows with Reynolds numbers up to 100,000.

Usage

```
calc_friction_blasius(reynolds, roughness, diameter)
```

Arguments

`reynolds`	Numeric. The Reynolds number of the flow (dimensionless).
`roughness`	Numeric. The absolute internal roughness of the pipe (meters).
`diameter`	Numeric. The internal diameter of the pipe (meters).

Value

A numeric vector representing the Darcy friction factor.

Examples

```
# Calculate friction factor for a Reynolds number of 50,000
calc_friction_blasius(reynolds = 50000, roughness = 0, diameter = 0.1)
```

`calc_friction_cw` *Calculate Friction Factor using Colebrook-White*

Description

Calculates the Darcy friction factor iteratively using the implicit Colebrook-White equation for turbulent flow. For laminar flow (Re <= 2000), it returns the exact solution (64 / Re).

Usage

```
calc_friction_cw(reynolds, roughness, diameter)
```

Arguments

`reynolds`	Numeric. The Reynolds number of the flow (dimensionless).
`roughness`	Numeric. The absolute internal roughness of the pipe (meters).
`diameter`	Numeric. The internal diameter of the pipe (meters).

Value

A numeric vector representing the Darcy friction factor.

Examples

```
calc_friction_cw(reynolds = 100000, roughness = 0.00026, diameter = 0.1)
```

`calc_friction_haaland` *Calculate Friction Factor using Haaland*

Description

Calculates the Darcy friction factor using the explicit Haaland equation. This formula is a highly accurate explicit approximation of the Colebrook-White equation for turbulent flows.

Usage

```
calc_friction_haaland(reynolds, roughness, diameter)
```

Arguments

`reynolds`	Numeric. The Reynolds number of the flow (dimensionless).
`roughness`	Numeric. The absolute internal roughness of the pipe (meters).
`diameter`	Numeric. The internal diameter of the pipe (meters).

Value

A numeric vector representing the Darcy friction factor.

Examples

```
# Calculate friction factor for a Reynolds number of 100,000
calc_friction_haaland(reynolds = 100000, roughness = 0.00026, diameter = 0.1)
```

---

calc_friction_sj   *Calculate Friction Factor using Swamee-Jain*

Description

Calculates the Darcy friction factor using the explicit Swamee-Jain equation. This is a highly accurate approximation of the Colebrook-White equation that does not require iteration.

Usage

```
calc_friction_sj(reynolds, roughness, diameter)
```

Arguments

reynolds	Numeric. The Reynolds number of the flow (dimensionless).
roughness	Numeric. The absolute internal roughness of the pipe (meters).
diameter	Numeric. The internal diameter of the pipe (meters).

Value

A numeric vector representing the Darcy friction factor.

Examples

```
calc_friction_sj(reynolds = 100000, roughness = 0.00026, diameter = 0.1)
```

calc_head_loss_darcy *Calculate Head Loss using Darcy-Weisbach*

Description

Calculates the friction head loss in a pipe based on the universal Darcy-Weisbach equation. This function allowing the user to inject any friction factor calculation function (e.g., Colebrook-White or Swamee-Jain). Alternatively, a pre-calculated friction factor can be provided directly.

Usage

```
calc_head_loss_darcy(
  length,
  flow,
  diameter,
  roughness = NULL,
  friction_factor = NULL,
  friction_fun = NULL,
  viscosity = 1.004e-06,
  gravity = 9.81
)
```

Arguments

`length`	Numeric. The length of the pipe (meters).
`flow`	Numeric. The volumetric flow rate (cubic meters per second).
`diameter`	Numeric. The internal diameter of the pipe (meters).
`roughness`	Numeric. The absolute internal roughness of the pipe (meters). Required unless `friction_factor` is provided.
`friction_factor`	Numeric. An optional pre-calculated Darcy friction factor. If provided, `roughness` and `friction_fun` are ignored.
`friction_fun`	Function. A function to calculate the friction factor (must accept `reynolds`, `roughness`, and `diameter`). If `NULL` (the default), it uses `calc_friction_cw`.
`viscosity`	Numeric. Kinematic viscosity of the fluid (sq. meters per sec). Default is 1.004e-6.
`gravity`	Numeric. Acceleration due to gravity (meters per second squared). Default is 9.81.

Value

A numeric vector representing the head loss in meters.

Examples

```
# 1. Default: Uses Colebrook-White function automatically
calc_head_loss_darcy(length = 100, flow = 0.02, diameter = 0.1, roughness = 0.00026)

# 2. Functional Injection: Pass the Swamee-Jain function as an argument
calc_head_loss_darcy(
  length = 100,
  flow = 0.02,
  diameter = 0.1,
  roughness = 0.00026,
  friction_fun = calc_friction_sj
)

# 3. Direct Value: Provide the friction factor manually
calc_head_loss_darcy(length = 100, flow = 0.02, diameter = 0.1, friction_factor = 0.02)
```

---

calc_head_loss_flamant

Calculate Head Loss using Flamant Equation

Description

Calculates the friction head loss in a pipe based on the empirical Flamant equation. This formula is highly recommended for small-diameter plastic pipes (typically < 50 mm) commonly used in micro-irrigation systems.

Usage

```
calc_head_loss_flamant(length, flow, diameter, coef = 0.000135)
```

Arguments

`length`	Numeric. The length of the pipe (meters).
`flow`	Numeric. The volumetric flow rate (cubic meters per second).
`diameter`	Numeric. The internal diameter of the pipe (meters).
`coef`	Numeric. The Flamant roughness coefficient (b). Default is 0.000135, which is typical for smooth plastic pipes (e.g., PVC, PE).

Value

A numeric vector representing the head loss in meters.

Examples

```
# Calculate head loss for a 50m PE pipe with 15mm diameter and 0.2 L/s flow
calc_head_loss_flamant(length = 50, flow = 0.0002, diameter = 0.015)
```

calc_head_loss_hw *Calculate Head Loss using Hazen-Williams Equation*

Description

Calculates the friction head loss in a pipe based on the empirical Hazen-Williams (hw) equation. This is valid only for water at ordinary temperatures.

Usage

```
calc_head_loss_hw(length, flow, diameter, coef = 140)
```

Arguments

length	Numeric. The length of the pipe (meters).
flow	Numeric. The volumetric flow rate (cubic meters per second).
diameter	Numeric. The internal diameter of the pipe (meters).
coef	Numeric. The Hazen-Williams roughness coefficient (dimensionless). Default is 140, which is typical for PVC pipes.

Value

A numeric vector representing the head loss in meters.

Examples

```
# Calculate head loss for a 100m PVC pipe (C = 140) with 0.1m diameter and 0.02 m^3/s flow
calc_head_loss_hw(length = 100, flow = 0.02, diameter = 0.1)
```

Index