Package ‘hydraulics’

August 2, 2020

Type Package

Title Basic Pipe and Open Channel Hydraulics

Version 0.2.1

Author Ed Maurer [aut, cre], Irucka Embry [aut, ctb] (iemisc code)

Maintainer Ed Maurer <emaurer@scu.edu>

Description Functions for basic hydraulic calculations related to water flow in circular pipes both flowing full (under pressure), and partially full (gravity flow), and trapezoidal open channels. For pressure flow this includes friction loss calculations by solving the Darcy-Weisbach equation for head loss, flow or diameter, and plotting a Moody diagram. The Darcy-Weisbach friction factor is calculated using the Colebrook (or Colebrook-White equation), the basis of the Moody diagram, the original citation being Colebrook (1939) <doi:10.1680/ijoti.1939.13150>. For gravity flow, the Manning equation is used, again solving for missing parameters. The derivation of and solutions using the Darcy-Weisbach equation and the Manning equation are outlined in many fluid mechanics texts such as Finnemore and Franzini (2002, ISBN:978-0072432022). For the Manning equation solutions, this package uses modifications of original code from the 'iemisc' package by Irucka Embry.

License GPL (>= 3)

Depends R (>= 3.6.0)

Encoding UTF-8

LazyData true

Imports ggplot2, grid, reshape2

Suggests

RoxygenNote 7.0.2

NeedsCompilation no

Repository CRAN

Date/Publication 2020-08-02 04:50:02 UTC

R topics documented:

`colebrook` *Calculates the Darcy-Weisbach Friction Factor f*

Description

This function calculates the Darcy-Weisbach friction factor and is only provided in this package for use with water in circular pipes while the equation is technically valid for any liquid or channel. As with many parts of this package, techniques and formatting were drawn from Irucka Embry's iemisc package, which includes some methods with similar functionality. Two utility functions are included for velocity and Reynolds Number.

Usage

```
velocity(D = NULL, Q = NULL)

reynolds_number(V = NULL, D = NULL, nu = NULL)

colebrook(ks, V, D, nu)
```

Arguments

`D`	numeric vector that contains the pipe diameter [m or ft] which should be D >=0.0025 m (0.0082 ft).
`Q`	(for velocity function only) numeric vector that contains the flow rate [$m^3 s^{-1}$ or $ft^3 s^{-1}$]
`V`	numeric vector that contains the average Velocity of flow in the pipe, equal to flow divided by area, $\frac{Q}{A}$ [$m\,s^{-1}$ or $ft\,s^{-1}$]
`nu`	numeric vector that contains the kinematic viscosity of water, [$m^2 s^{-1}$ or $ft^2 s^{-1}$]. Computed with a utility function in water_properties.R: kvisc(T=T, units=['SI' or 'Eng'])
`ks`	numeric vector that contains the 'equivalent sand roughness height sand roughness height. Units should be consistent with other input [m or ft]

Details

The Colebrook-White equation was developed to estimate the Darcy-Weisbach friction factor for commercial pipes under turbulent flow conditions. It is recommended for pipe diameters greater than 2.5 mm (0.1 inch). The equation is:

$$\frac{1}{\sqrt{f}} = -2\log\left(\frac{\frac{ks}{D}}{3.7D} + \frac{2.51}{Re\sqrt{f}}\right)$$

where $Re = \frac{VD}{nu}$ is the unitless Reynolds Number.

Value

f Returns a numeric vector containing the Darcy-Weisbach friction factor

Author(s)

Ed Maurer

See Also

`kvisc` for kinematic viscosity, `velocity` for calculating $V = \frac{Q}{A}$, `reynolds_number` for Reynolds number

Examples

```
# A Type 1 problem (solve for hf): US units
D <- 20/12   #diameter of 20 inches
Q <- 4       #flow in ft^3/s
T <- 60      #water temperature in F
ks <- 0.0005 #pipe roughness in ft

f <- colebrook(ks=ks,V=velocity(D,Q), D=D, nu=kvisc(T=T, units="Eng"))
```

`darcyweisbach` *Solves the Darcy-Weisbach Equation for the either head loss (hf), flow rate (Q), diameter (D), or roughness height (ks).*

Description

This function solves the Darcy-Weisbach friction loss equation for with water in circular pipes. the function solves for either head loss (hf), flow rate (Q), diameter (D),or roughness height, (ks) whichever is missing (not included as an argument).

Usage

```
darcyweisbach(
  Q = NULL,
  D = NULL,
  hf = NULL,
  L = NULL,
  ks = NULL,
  nu = NULL,
  units = c("SI", "Eng")
)
```

Arguments

`Q`	numeric vector that contains the flow rate [$m^3 s^{-1}$ or $ft^3 s^{-1}$]
`D`	numeric vector that contains the pipe diameter [m or ft]
`hf`	numeric vector that contains the head loss through the pipe section [m or ft]
`L`	numeric vector that contains the pipe length [m or ft],
`ks`	numeric vector that contains the equivalent sand roughness height. Units should be consistent with other input [m or ft]
`nu`	numeric vector that contains the kinematic viscosity of water, [$m^2 s^{-1}$ or $ft^2 s^{-1}$].
`units`	character vector that contains the system of units [options are `SI` for International System of Units and `Eng` for English (US customary) units. This is used for compatibility with iemisc package

Details

The Darcy-Weisbach equation was developed to estimate the head loss, h_f, due to friction over a length of pipe. For circular pipes it is expressed as:

$$h_f = \frac{fL}{D}\frac{V^2}{2g} = \frac{8fL}{\pi^2 g D^5}Q^2$$

where f is the friction factor (calculated with the `colebrook` function and g is the gravitational acceleration ($9.81\frac{m}{s^2}$ or $32.2\frac{ft}{s^2}$).

Value

Returns a list including the missing parameter (hf, Q, D, or ks):

- Q - flow rate.
- V - flow velocity.
- L - pipe length.
- hf - head loss due to friction
- f - Darcy-Weisbach friction factor
- ks - roughness height
- Re - Reynolds number

See Also

colebrook for friction factor calculation

Examples

```
#Type 2 (solving for flow rate, Q): SI Units
D <- .5
L <- 10
hf <- 0.006*L
T <- 20
ks <- 0.000046
darcyweisbach(D = D, hf = hf, L = L, ks = ks, nu = kvisc(T=T, units='SI'), units = c('SI'))

#Type 3 (solving for diameter, D): Eng (US) units
Q <- 37.5     #flow in ft^3/s
L <- 8000     #pipe length in ft
hf <- 215     #head loss due to friction, in ft
T <- 68       #water temperature, F
ks <- 0.0008  #pipe roughness, ft
darcyweisbach(Q = Q, hf = hf, L = L, ks = ks, nu = kvisc(T=T, units='Eng'), units = c('Eng'))
```

---

manningc — *Solves the Manning Equation for gravity flow in a circular pipe*

Description

This function solves the Manning equation for water flow in a circular pipe at less than full. Uniform flow conditions are assumed, so that the pipe slope is equal to the slope of the water surface and the energy grade line. This is a modification of the code prepared by Irucka Embry in his iemisc package. The iemisc::manningcirc function was adapted here for more limited cases commonly used in classroom exercises, additional checks were included to ensure the pipe is flowing less than full, and a cross-section figure is also available. The iemisc::manningcirc and iemisc::manningcircy functions were combined into a single function.

Usage

```
manningc(
  Q = NULL,
  n = NULL,
  Sf = NULL,
  y = NULL,
  d = NULL,
  y_d = NULL,
  units = c("SI", "Eng")
)
```

Arguments

`Q`	numeric vector that contains the flow rate [$m^3 s^{-1}$ or $ft^3 s^{-1}$]
`n`	numeric vector that contains the Manning roughness coefficient.
`Sf`	numeric vector that contains the slope of the pipe [unitless]
`y`	numeric vector that contains the water depth [m or ft]
`d`	numeric vector that contains the pipe diameter [m or ft]
`y_d`	numeric vector that contains the ratio of depth to diameter [unitless]
`units`	character vector that contains the system of units [options are `SI` for International System of Units and `Eng` for English (US customary) units. This is used for compatibility with iemisc package]

Details

The possible applications of this function for solving the Manning equation for circular pipes are:

Given	**Solve for**
y_d, Q, Sf, n	d
d, Sf, Q, n	y
y, d, Q, n	Sf
y, d, Sf, n	Q
d, Q, Sf, y	n

The Manning equation (also known as the Strickler equation) describes flow conditions in an open channel under uniform flow conditions. It is often expressed as:

$$Q = A\frac{C}{n}R^{\frac{2}{3}}{S_f}^{\frac{1}{2}}$$

where C is 1.0 for SI units and 1.49 for Eng (U.S. Customary) units. Critical depth is defined by the relation (at critical conditions):

$$\frac{Q^2 T}{g\,A^3} = 1$$

where T is the top width of the water surface. Since T equals zero for a full pipe, critical depth is set to the pipe diameter d if the flow Q exceeds a value that would produce a critical flow at $\frac{y}{d} = 0.99$.

Value

Returns a list including the missing parameter:

- Q - flow rate
- V - flow velocity
- A - cross-sectional area of flow
- P - wetted perimeter
- R - hydraulic radius (A/P)
- y - flow depth

- d - pipe diameter
- Sf - slope
- n - Manning's roughness
- yc - critical depth
- Fr - Froude number
- Re - Reynolds number
- Qf - Full pipe flow rate

Author(s)

Ed Maurer, Irucka Embry

See Also

xc_circle for a cross-section diagram of the circular channel

Examples

```
#Solving for flow rate, Q: SI Units
manningc(d = 0.6, n = 0.013, Sf = 1./400., y = 0.24, units = "SI")
#returns 0.1 m3/s

#Solving for Sf, if d=600 mm and pipe is to flow half full
manningc(d = 0.6, Q = 0.17, n = 0.013, y = 0.3, units = "SI")
#returns required slope of 0.003

#Solving for diameter, d when given y_d): Eng (US) units
manningc(Q = 83.5, n = 0.015, Sf = 0.0002, y_d = 0.9, units = "Eng")
#returns 7.0 ft required diameter

#Solving for depth, d when given Q: SI units
manningc(Q=0.01, n=0.0013, Sf=0.001, d = 0.2, units="SI")
#returns depth  y = 0.042 m, critical depth, yc = 0.085 m
```

---

manningt — *Solves the Manning Equation for water flow in an open channel*

Description

This function solves the Manning equation for water flow in an open channel with a trapezoidal shape. Uniform flow conditions are assumed, so that the channel slope is equal to the slope of the water surface and the energy grade line. This is a modification of the code prepared by Irucka Embry in his iemisc package. Specifically the iemisc::manningtrap, iemisc::manningrect, and iemisc::manningtri were combined and adapted here for cases commonly used in classroom exercises. Some auxiliary variables in the iemisc code are not included here (shear stress, and specific energy), as these can be calculated separately. A cross-section figure is also available.

Usage

```
manningt(
  Q = NULL,
  n = NULL,
  m = NULL,
  Sf = NULL,
  y = NULL,
  b = NULL,
  units = c("SI", "Eng")
)
```

Arguments

`Q`	numeric vector that contains the flow rate [$m^3 s^{-1}$ or $ft^3 s^{-1}$]
`n`	numeric vector that contains the Manning roughness coefficient
`m`	numeric vector that contains the side slope of the channel (m:1 H:V) [unitless]
`Sf`	numeric vector that contains the slope of the channel [unitless]
`y`	numeric vector that contains the water depth [m or ft]
`b`	numeric vector that contains the channel bottom width [m or ft]
`units`	character vector that contains the system of units [options are `SI` for International System of Units and `Eng` for English (US customary) units. This is used for compatibility with iemisc package.

Details

The Manning equation (also known as the Strickler equation) describes flow conditions in an open channel under uniform flow conditions. It is often expressed as:

$$Q = A\frac{C}{n}R^{\frac{2}{3}}{S_f}^{\frac{1}{2}}$$

where C is 1.0 for SI units and 1.49 for Eng (U.S. Customary) units. Critical depth is defined by the relation (at critical conditions):

$$\frac{Q^2T}{g\,A^3} = 1$$

where T is the top width of the water surface.

Value

Returns a list including the missing parameter:

- Q - flow rate
- V - flow velocity
- A - cross-sectional area of flow
- P - wetted perimeter
- R - hydraulic radius

- y - flow depth (normal depth)
- b - channel bottom width
- m - channel side slope
- Sf - slope
- B - top width of water surface
- n - Manning's roughness
- yc - critical depth
- Fr - Froude number
- Re - Reynolds number

Author(s)

Ed Maurer, Irucka Embry

See Also

spec_energy_trap for specific energy diagram and xc_trap for a cross-section diagram of the trapezoidal channel

Examples

```
#Solving for flow rate, Q, trapezoidal channel: SI Units
manningt(n = 0.013, m = 2, Sf = 0.0005, y = 1.83, b = 3, units = "SI")
#returns Q=22.2 m3/s

#Solving for roughness, n, rectangular channel: Eng units
manningt(Q = 14.56, m = 0, Sf = 0.0004, y = 2.0, b = 4, units = "Eng")
#returns Manning n of 0.016

#Solving for depth, y, triangular channel: SI units
manningt(Q = 1.0, n = 0.011, m = 1, Sf = 0.0065, b = 0, units = "SI")
#returns 0.6 m normal flow depth
```

---

moody — *Creates a Moody diagram with optional manually added points*

Description

This function plots a standard Moody diagram, and allows additional points to be added by including arguments Re and f.

Usage

```
moody(Re = NULL, f = NULL)
```

Arguments

Re	(optional) numeric vector that contains the Reynolds numbers of points to be manually added
f	(optional) numeric vector (same length as Re) that contains the Darcy-Weisbach friction factors corresponding to the points to be manually added

Value

a Moody diagram, with the optional added (Re, f) points

Author(s)

Ed Maurer

Examples

```
# Draw canonical Moody diagram
moody()

# Draw Moody diagram plotting two additional points
Re = c(10000, 100000)
f = c(0.04, 0.03)
moody( Re = Re, f = f )
```

---

spec_energy_trap *Creates a specific energy diagram for a trapezoidal channel*

Description

This function plots a specific energy diagram of a trapezoidal (including rectangular and triangular) channel, with annotation of critical depth and minimum specific energy.

Usage

```
spec_energy_trap(
  Q = NULL,
  b = NULL,
  m = NULL,
  scale = 3,
  units = c("SI", "Eng")
)
```

Arguments

`Q`	flow rate [$m^3 s^{-1}$ or $ft^3 s^{-1}$]
`b`	bottom width [m or ft]
`m`	side slope (H:1) [unitless]
`scale`	multiplier (of yc) for axis scales (default is 3)
`units`	character vector that contains the system of units [options are `SI` for International System of Units and `Eng` for English (US customary) units.

Details

Specific Energy, E, is the energy, expressed as a head (i.e., the mechanical energy per unit weight of the water, with units of length) relative to the channel bottom. It is calculated as:

$$E = y + \alpha\frac{Q^2}{2g\,A^2} = y + \alpha\frac{V^2}{2g}$$

where y is flow depth, A is the cross-sectional flow area, $V = \frac{Q}{A}$, and and α is a kinetic energy correction factor to account for non-uniform velocities across the cross-section; $\alpha = 1.0$ in this function (as is commonly assumed).

Value

a specific energy diagram

Author(s)

Ed Maurer

Examples

```
# Draw a specific cross-section with flow 1, width 2, side slope 3:1 (H:V)
spec_energy_trap(Q = 1.0, b = 2.0, m = 3.0, scale = 4, units = "SI")
```

`waterprops` *Functions to calculate water properties: density, dynamic and kinematic viscosity*

Description

This function calculates water properties that are used in other functions.

Usage

```
dvisc(T = NULL, units = c("SI", "Eng"))

dens(T = NULL, units = c("SI", "Eng"))

kvisc(T = NULL, units = c("SI", "Eng"))
```

Arguments

T	numeric vector that contains the water temperature [$^{\circ}C$ or $^{\circ}F$]
units	character vector that contains the system of units [options are SI for International System of Units and Eng for English (US customary) units. This is used for compatibility with iemisc package

Value

rho, the density of water for the dens function [$kg\,m^{-3}$ or $slug\,ft^{-3}$]

mu, the dynamic viscosity of water for the dvisc function [$N\,s\,m^{-2}$ or $lbf\,s\,ft^{-2}$]

nu, the kinematic viscosity of water for the kvisc function [m^2s^{-1} or ft^2s^{-1}].

Author(s)

Ed Maurer

Examples

```
#Find kinematic viscocity for water temperature of 55 F
nu = kvisc(T = 55, units = 'Eng')

#Find kinematic viscocity assuming default water temperature of 68 F
nu = kvisc(units = 'Eng')

#Find water density for water temperature of 25 C
rho = dens(T = 25, units = 'SI')
```

---

xc_circle — *Creates a cross-section plot for a partially filled pipe*

Description

This function plots a cross-section of a circular pipe, shaded as filled to the level indicated by the depth and diameter values passed to it.

Usage

```
xc_circle(y = NULL, d = NULL)
```

Arguments

y	water depth [m or ft]
d	pipe diameter [m or ft]

Value

a cross-section diagram

Author(s)

Ed Maurer

Examples

```
# Draw a cross-section with diameter 1.0 and depth 0.7
xc_circle(y = 0.7, d = 1.0)
```

---

xc_trap *Creates a cross-section plot for an open channel*

Description

This function plots a cross-section of a (trapezoid, rectangle, triangle), shaded as filled to the level indicated by the values passed to it.

Usage

```
xc_trap(y = NULL, b = NULL, m = NULL)
```

Arguments

y	water depth [m or ft]
b	bottom width [m or ft]
m	side slope (H:1)

Value

a cross-section diagram

Author(s)

Ed Maurer

Examples

```
# Draw a cross-section with depth 1, width 2, side slope 3:1 (H:V)
xc_trap(y = 1.0, b = 2.0, m = 3.0)
```

Index