HydDown

User guide and Technical reference

Anders Andreasen

Contents

1 Introduction

HydDown is an open source python tool for calculation of Hydrogen (or other pure gas phase species) vessel/container depressurization and filling. The HydDown logo shown in Fig. 1.1 visualizes the key parameters and transport phenomena during gas vessel filling or discharging. The thermodynamic state inside the vessel changes over time as seen from immediately observable variables temperature (T) and pressure (P). This is caused by change in fluid inventory (density) due to flow of gas either in- or out of the vessel. Further, heat is transfered from or to the surroundings via convective heat transfer on the in- and outside of the vessel - with heat being conducted thorugh the vessel wall.

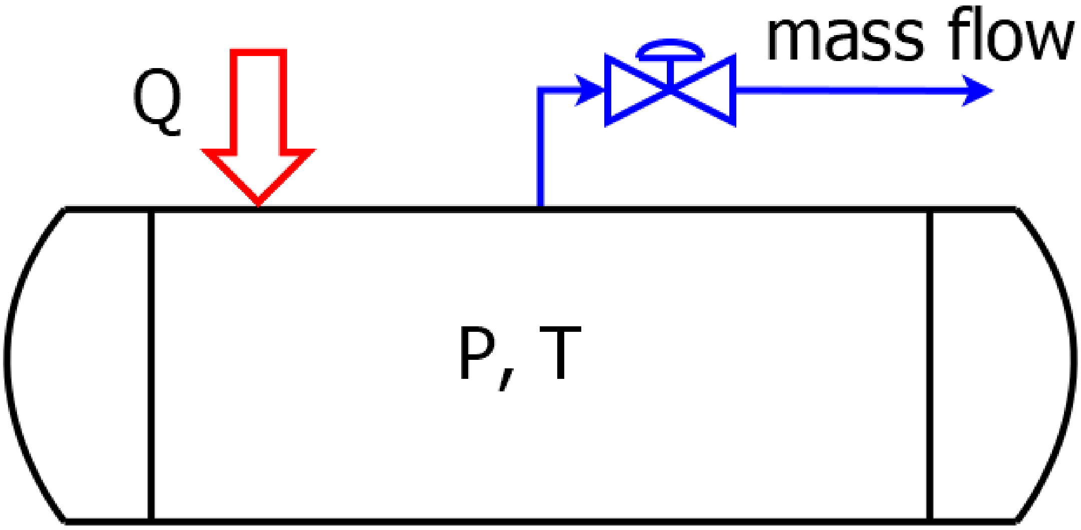

Figure 1.1: HydDown logo

Run the code as simple as:

```
python main.py input.yml
```

where main.py is the main script and input.yml is the input file in Yaml syntax.

1.1 Background

This is a small spare time project for calculation of vessel filling and depressurisation behaviour. This is mainly to demonstrate, that although perceived as a very tedious/difficult task to write your own code for such an apparent complex problem, actually a fairly limited amount of code is necessary if you have a good thermodynamic backend.

A few choices is made to keep things simple to begin with:

- Coolprop is used as thermodynamic backend
- Only pure substances are considered
- Gas phase only
- No temperture stratification in the gas phase
- No temperture gradient through vessel wall
- Heat transfer is modelled as simple as possible

These choices makes the problem a lot more simple to solve, First of all the the pure substance Helmholtz energy based equation of state (HEOS) in coolprop offers a lot of convenience in terms of the property pairs/state variables that can be set independently. Using only a single gas phase species also means that component balances is redundant and 2 or 3-phase flash calculations are not required. That being said the principle used for a single component is more or less the same, even for multicomponent mixtures with potentially more than one phase.

1.2 Requirements

- Python (3.8 - at least python3)
- Numpy
- matplotlib
- Coolprop (6.4.1)
- cerberus
- PyYaml
- pandas

The script is running on Windows 10 x64, with stock python installation from python.org and packages installed using pip. Should run om linux (it does on an Ubuntu image on GitHub) or in any conda environment as well, but I haven't checked.

1.3 Units of measure

The SI units are adapted for this project. The following common units are used in the present project and this also applies to the units used in the input files:

Table 1.1: Unit system

Property	Unit	Comment
Temperature	K	$^\circ$ C is used in plots
Pressure	Pa	bar is used in plots
Mass	kg	
Volume	m^3	
Time	s	
Energy	J	
Duty/power	W	
Length	m	
Area	m^2	
Heat flux	$\mathrm{W/m}^2$	
Heat transfer coefficient	$\mathrm{W/(m}^2$ K)	
Density	$\mathrm{kg/m}^3$	
Heat capacity	J/(kg K)	

1.4 License

2 Usage

2.1 Basic usage

Run the code as simple as:

```
python main.py input.yml
```

where main.py is the main script and input.yml is the input file in Yaml syntax.

The Yaml input file is edited to reflect the system of interest.

2.2 Calculation methods

The following methods are implemented:

- Isothermal i.e. constant temperature of the fluid during depressurisation (for a very slow process with a large heat reservoir)
- Isenthalpic/Adiabatic (no heat transfer with surroundings, no work performed by the expanding fluid)
- Isentropic (no heat transfer with surroundings, PV work performed by the expanding fluid)
- Constant internal energy
- Energy balance. This is the most general case and includes both the ability to transfer heat with surroundings as well as accounting for PV work.

For isothermal/isenthalpic/isentropic/isenergetic calculations the minimal input required are:

- Initial conditions (pressure, temperature)
- vessel dimensions (ID/length)
- valve parameters (Cd, diameter, backpressure)
- Calculation setup (time step, end time)
- Type of gas

If heat transfer is to be considered the calculation type "energybalance" is required. A few options are possible:

- Fixed U (U-value required, and ambient temperature)
- Fixed Q (Q to be applied to the fluid is requried)
- Specified h, the external heat transfer coefficient is provided and either the internal is provided or calculated from assumption of natural convection from a vertical cylinder at high Gr number. Ambient temperature is required.
- Fire (Stefan-Boltzmann equation heat duty)

2.3 Script

```python
import yaml
import sys
from hyddown import HydDown

if __name__ == "__main__":
    if len(sys.argv) > 1:
        input_filename = sys.argv[1]
    else:
        input_filename = "input.yml"

    with open(input_filename) as infile:
        input = yaml.load(infile, Loader=yaml.FullLoader)


    hdown=HydDown(input)
    hdown.run()
    hdown.verbose=1
    hdown.plot()
```

2.4 Module import

2.5 Input file

3 Theory

3.1 Thermodynamics

3.1.1 Equation of state

The equation of state used by HydDown is the Helmholtz energy formulation as implemented in CoolProp (Bell et al. 2014). Most of the text in the present section has been sourced from the CoolProp documentation to be as accurate and true to the source as possible. The Helmholtz energy formulation is a convenient construction of the equation of state because all the thermodynamic properties of interest can be obtained directly from partial derivatives of the Helmholtz energy.

It should be noted that the EOS are typically valid over the entire range of the fluid, from subcooled liquid to superheated vapor, to supercritical fluid. In general, the EOS are based on non-dimensional terms δ and τ, where these terms are defined by

$$\delta = \rho/\rho_c$$

$$\tau = T_c/T$$

where ρ_c and T_c are the critical density of the fluid if it is a pure fluid. For pseudo-pure mixtures, the critical point is typically not used as the reducing state point, and often the maximum condensing temperature on the saturation curve is used instead.

The non-dimensional Helmholtz energy of the fluid is given by

$$\alpha = \alpha^0 + \alpha^r$$

where α^0 is the ideal-gas contribution to the Helmholtz energy, and α^r is the residual Helmholtz energy contribution which accounts for non-ideal behavior. For a given set of δ and τ, each of the terms α^0 and α^r are known. The exact form of the Helmholtz energy terms is fluid dependent, but a relatively simple example is that of Nitrogen, which has the ideal-gas Helmholtz energy of

$$\alpha^0 = \ln\delta + a_1\ln\tau + a_2 + a_3\tau + a_4\tau^{-1} + a_5\tau^{-2} + a_6\tau^{-3} + a_7\ln[1-\exp(-a_8\tau)]$$

and the non-dimensional residual Helmholtz energy of

$$\alpha^r = \sum_{k=1}^{6} N_k\delta^{i_k}\tau^{j_k} + \sum_{k=7}^{32} N_k\delta^{i_k}\tau^{j_k}\exp(-\delta^{l_k}) + \sum_{k=33}^{36} N_k\delta^{i_k}\tau^{j_k}\exp(-\phi_k(\delta-1)^2 - \beta_k(\tau-\gamma_k)^2)$$

and all the terms other than δ and τ are fluid-dependent correlation parameters.

The other thermodynamic parameters can then be obtained through analytic derivatives of the Helmholtz energy terms. For instance, the pressure is given by

$$p = \rho RT\left[1 + \delta\left(\frac{\partial\alpha^r}{\partial\delta}\right)_\tau\right]$$

and the specific internal energy by

$$\frac{u}{RT} = \tau\left[\left(\frac{\partial\alpha^0}{\partial\tau}\right)_\delta + \left(\frac{\partial\alpha^r}{\partial\tau}\right)_\delta\right]$$

and the specific enthalpy by

$$\frac{h}{RT} = \tau\left[\left(\frac{\partial\alpha^0}{\partial\tau}\right)_\delta + \left(\frac{\partial\alpha^r}{\partial\tau}\right)_\delta\right] + \delta\left(\frac{\partial\alpha^r}{\partial\delta}\right)_\tau + 1$$

which can also be written as

$$\frac{h}{RT} = \frac{u}{RT} + \frac{p}{\rho RT}$$

The specific entropy is given by

$$\frac{s}{R} = \tau\left[\left(\frac{\partial\alpha^0}{\partial\tau}\right)_\delta + \left(\frac{\partial\alpha^r}{\partial\tau}\right)_\delta\right] - \alpha^0 - \alpha^r$$

and the specific heats at constant volume and constant pressure respectively are given by

$$\frac{c_v}{R} = -\tau^2\left[\left(\frac{\partial^2\alpha^0}{\partial\tau^2}\right)_\delta + \left(\frac{\partial^2\alpha^r}{\partial\tau^2}\right)_\delta\right]$$

$$\frac{c_p}{R} = \frac{c_v}{R} + \frac{\left[1+\delta\left(\frac{\partial\alpha^r}{\partial\delta}\right)_\tau - \delta\tau\left(\frac{\partial^2\alpha^r}{\partial\delta\partial\tau}\right)\right]^2}{\left[1+2\delta\left(\frac{\partial\alpha^r}{\partial\delta}\right)_\tau + \delta^2\left(\frac{\partial^2\alpha^r}{\partial\delta^2}\right)_\tau\right]}$$

The EOS is set up with temperature and density as the two independent properties, but often other inputs are known, most often temperature and pressure because they can be directly measured. As a result, if the density is desired for a known temperature and pressure, it can be obtained iteratively.

3.1.2 Isothermal process

3.1.3 Isentropic process

3.1.4 Isenthalpic process

3.1.5 General energy balance

3.2 Flow devices

3.2.1 Restriction Orifice

3.2.2 Relief valve

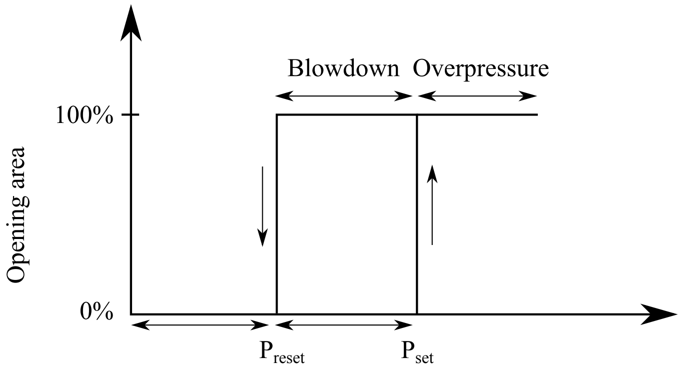

Figure 3.1: Relief valve hysteresis

3.2.3 Control Valve

3.3 Heat transfer

3.3.1 Natural convection

To determine the heat transfer for the gas-wall interface, Newton's law of cooling is applied, as given in equation eq. 3.1.

$$\frac{dQ}{dt} = -hA(T_s - T_{gas}) \tag{3.1}$$

- dQ is the change in thermal energy due to convective heat transfer. [J]
- dt is the change in time during the heat transfer. [s]
- h is the convective heat transfer. [W/m^2 · K]
- A is the area normal to the direction of the heat transfer. [m^2]
- T_s is the surface temperature of the geometry. [K]
- T_{gas} is the temperature of the surrounding gas. [K]

Equation [eq. **??**} indicates the heat transfer to the surface of any geometry such as a plate or cylindrical wall, by means of heat convection from its surroundings (Çengel and Ghajar 2011)(Geankoplis 1993).

The convective heat transfer will need to be estimated for the the gas-wall interface, by the use of empirical relations for the Nusselt number. The Nusselt number describes the ratio of convective heat transfer to conductive heat transfer, normal to a surface area, as given in equation eq. 3.2.

$$Nu = \frac{hL}{k} \tag{3.2}$$

- Nu is the Nusselt number. [-]
- h is the convective heat transfer. [W/m^2·K]
- L is a characteristic length of the geometry. [m]
- k is the thermal conductivity of the gas. [W/m·K]

The characteristic length L used is the height of the gas volume.

The empirical correlations used to calculate the Nusselt number of the gas-wall interface is a function of the Rayleigh number, which can be defined by the Grashof number and Prandtl number, as in equation eq. 3.3.

$$Ra = Gr \cdot Pr \tag{3.3}$$

- Ra is the Rayleigh number. [-]
- Gr is the Grashof number. [-]
- Pr is the Prandtl number. [-]

The Grashof number is a dimensionless number which approximates the ratio of the buoyancy forces to viscous forces, as given in equation [eq. 3.4}. The Prandtl number is a dimensionless number defined as the ratio of the momentum diffusivity to thermal diffusivity, as given in equation eq. 3.5.

$$Gr = \frac{\beta g \rho^2 L^3 \Delta T}{\mu^2} \tag{3.4}$$

$$Pr = \frac{c_p \mu}{k} \tag{3.5}$$

- β is the coefficient of volume expansion. [1/K]
- g is the standard acceleration of gravity. [m/s^2]
- ρ is the gas density. [kg/m^3]
- L is the characteristic length. [m]
- ΔT is the temperature difference of the surface and gas. [K]
- μ is the dynamic viscosity. [kg/m·s]
- c_p is the heat capacity of gas. [J/kg·K]
- k is the thermal conductivity of gas. [J/m·K]

3.3.2 Mixed convection

3.3.3 Conduction

3.3.4 Fire heat loads

The heat transfer from the flame to the shell is modelled using the recommended approach from Scandpower (Hekkelstrand and Skulstad 2004). The heat transfer from the flame to the vessel shell is divided into radiation, convection and reradiation as seen in equation eq. 3.6.

$$q_f = \underbrace{\alpha_s \cdot \varepsilon_f \cdot \sigma \cdot T_f^4}_{\text{Radiation}} + \underbrace{h_f \cdot (T_f - T_s(t))}_{\text{Convection}} - \underbrace{\varepsilon_s \cdot \sigma \cdot T_s(t)^4}_{\text{Reradiation}} \tag{3.6}$$

- q_f is the flame heat flux. [W/m^2]
- α_s is the vessel surface absorptivity. [-]
- ε_f is the flame emissivity. [-]
- σ is the Stefan-Boltzmann constant, σ = $5.67 \cdot 10^{-8}$ [W/m^2· K^4]

- T_f is the flame temperature. [K]
- h_f is the convection heat transfer coefficient between the flame and the surface. [W/m^2· K]
- $T_s(t)$ is the time dependent surface temperature. [K]
- ε_s is the surface emissivity. [-]

This model assumes that the pressure vessel is fully engulfed by the flame. This means that the view factor for the radiation is unity and is therefore not taken into consideration. The convective heat transfer coefficients for a jet fire and a pool fire, and recommended values for the emissivity and absorptivity, are given by Scandpower as (Hekkelstrand and Skulstad 2004)

- $h_{jet\ fire}$ = 100 [W/m^2·K]
- $h_{pool\ fire}$ = 30 [W/m^2·K]
- α_s = 0.85
- ε_s = 0.85
- ε_f = 1.0 (optical thick flames, thickness > 1 m)

The flame temperature is found by solving equation eq. 3.7 for the incident heat flux in relation to the ambient conditions. The flame temperature is kept constant throughout the simulation.

$$q_{total} = \sigma \cdot T_f^4 + h_f \cdot (T_f - T_{amb}) \tag{3.7}$$

- q_{total} is the incident flame heat flux as given in table tbl. 3.1. [W/m^2]
- T_{amb} is the ambient temperature $\approx$ 293 K (20° C)

The heat flux used to calculate the flame temperature is given in table tbl. 3.1.

Table 3.1: Incident heat fluxes for various fire scenarios given by Scandpower (Hekkelstrand and Skulstad 2004)

	Small jet fire [kW/m^2]	Large jet fire [kW/m^2]	Pool fire [kW/m^2]
Peak heat load	250	350	150
Background heat load	0	100	100

3.4 Model implementation

A simple (naive) explicit Euler scheme is implemented to integrate the mass balance over time, with the mass rate being calculated from an orifice/valve equation. For each step, the mass relief/ left in the vessel is known. Since the volume is fixed the mass density is directly given. For the calculation methods (isentropic,isenthalpic,isenergetic etc), Coolprop allows specifying density and either H,S or U directly - this is very handy and normally only TP, PH, TS property pairs are implemented, and you would need to code a second loop to make it into am UV, VH or SV calculation. Coolprop is very convenient for this, however for a cubic EOS and for multicomponent Helmholtz energy EOS coolprop only supports a subset of state variables to be specified directly (T,P,quality). For this reason single component HEOS is the main target of this project.

4 Validation

4.1 Nitrogen discharge

The code is mainly for demonstration, and is provided as-is. However, a comparison has been made to experiment I1 from Haque et al. (see full ref below). The results are shown below.

4.2 Hydrogen filling

4.3 Air discharge/filling

References

Bell, Ian H., Jorrit Wronski, Sylvain Quoilin, and Vincent Lemort. 2014. "Pure and Pseudo-Pure Fluid Thermophysical Property Evaluation and the Open-Source Thermophysical Property Library CoolProp." *Industrial & Engineering Chemistry Research* 53 (6): 2498–508. https://doi.org/10.1021/ie4033999.

Çengel, Yunus A., and Afshin J. Ghajar. 2011. *Heat and Mass Transfer - Fundementals and Applications*. Fourth. McGraw-Hill.

Geankoplis, C. J. 1993. *Transport Processes and Unit Operations*. Chemical Engineering. PTR Prentice Hall.

Hekkelstrand, B., and P. Skulstad. 2004. *Guidelines for the Protection of Pressurised Systems Exposed to Fire*. Scandpower Risk Management AS.