

# SS14 gas pump proposal

slarticodefast  
github.com/slarticodefast

July 29, 2024

## Abstract

This document is intended to serve as a design document for improved gas pump mechanics following proper physics. We will derive the necessary formulas needed for an implementation from the basic laws of thermodynamics.

TODO: problems with current implementation of gas pumps:  
no energy conservation, no soft clogging, arbitrary prioritization of one pump over another at low pressure, how does this fit into the atmos road map?

## 1 Ideal gases

Space station 14 uses ideal gases to simulate atmospherics. An ideal gas is defined by following the ideal gas law

$$pV = nRT, \tag{1.1}$$

where  $p$  is the pressure in  $kPa$ ,  $V$  is the volume in  $L$ ,  $n$  is the amount of substance of gas in  $mol$  with  $1mol = 6.02214076 \cdot 10^{23}$  particles,  $R = 8.314462618 \cdot L \cdot kPa \cdot K^{-1} \cdot mol^{-1}$  is the universal gas constant and  $T$  is the absolute temperature in  $K$ . The internal energy of an ideal gas is defined by

$$U(n, T) = c_V nT, \tag{1.2}$$

where  $c_V$  is the molar heat capacity at constant volume, defined in `gases.yml` for all in-game gases.

## 2 First law of thermodynamics

The first law of thermodynamics is a formulation of energy conservation

$$\Delta U = Q + W, \quad (2.1)$$

where  $\Delta U$  is the change in internal energy of the gas,  $Q$  the heat supplied or withdrawn from the system and  $W$  the work done on or by the system. We use the positive sign convention, where all net energy transfers to the system are positive and all net energy transfers from the system are negative. For an infinitesimally small change it takes the form

$$dU = \delta Q + \delta W, \quad (2.2)$$

where the work  $\delta W = -pdV$  can be calculated from the pressure  $p$  and change in volume  $dV$ .

## 3 Free expansion

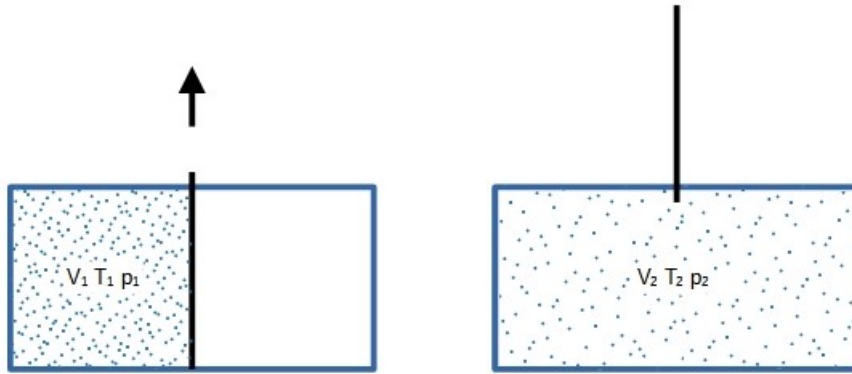


Figure 1: Free expansion

If we let a gas expand freely by removing a barrier (Figure 1), no work is being done  $W = 0$  and no heat is exchanged with the outside of the (thermally isolated) box  $Q = 0$ . Therefore the internal energy of the gas does not change  $\Delta U = 0$ . From 1.2 we follow that the temperature of the gas stays constant. We get

$$pV = nRT = \text{const.} \quad (3.1)$$

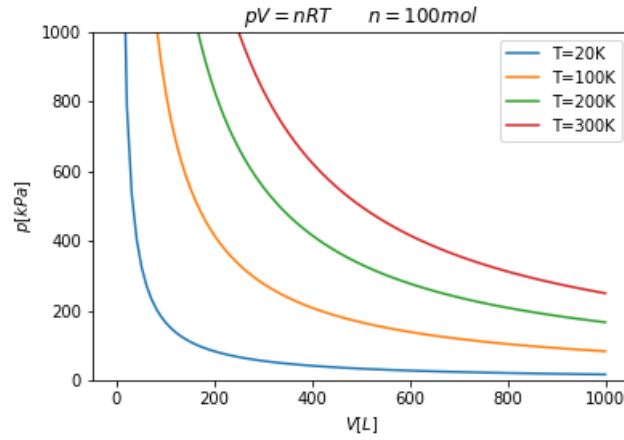


Figure 2: Lines of constant temperature (isothermals)

or equivalently

$$p_1 V_1 = p_2 V_2, \quad T_1 = T_2, \quad n_1 = n_2. \quad (3.2)$$

This process is not reversible as the gas cannot be forced back into the left part of the box without applying work. In SS14 free expansion happens if you fill an empty pipe network with a gas, for example by opening a valve or using a connector port.

## 4 Adiabatic process

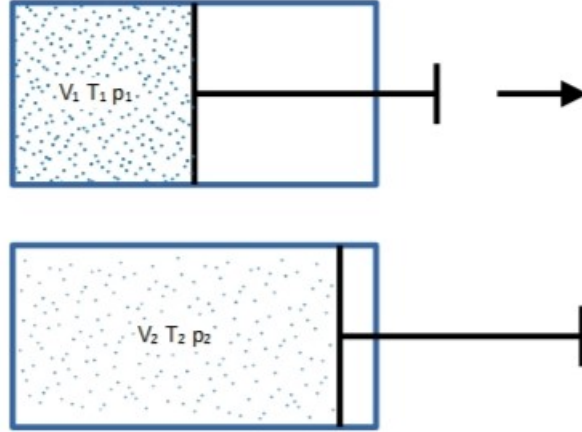


Figure 3: Adiabatic expansion

Now we expand the gas using a piston instead (Figure 3). The gas exchanges no heat with the environment outside the box and we have  $\delta Q = 0$ . The gas presses against the side of the piston as it expands, doing the work  $\delta W = -pdV$ . We get

$$c_V n dT = dU = \delta W \quad (4.1)$$

$$= -pdV. \quad (4.2)$$

Using the product rule of differentiation we get

$$dT = d\left(\frac{pV}{nR}\right) \quad (4.3)$$

$$= \frac{1}{nR} d(pV) \quad (4.4)$$

$$= \frac{1}{nR} (pdV + Vdp) \quad (4.5)$$

leading to

$$\frac{c_V}{R} (pdV + Vdp) = -pdV \quad (4.6)$$

$$\Leftrightarrow -\left(\frac{c_V + R}{c_V}\right) pdV = Vdp \quad (4.7)$$

$$\Leftrightarrow -\gamma \frac{dV}{V} = \frac{dp}{p} \quad (4.8)$$

with the heat capacity ration  $\gamma = \frac{c_p}{c_v} = \frac{c_v + R}{c_v}$ . This value is already defined for all gases in `gases.yml`, but currently unused. Integrating both sides gives us

$$-\gamma \ln \left( \frac{V_2}{V_1} \right) = \ln \left( \frac{p_2}{p_1} \right) \quad (4.9)$$

and exponentiating gives us

$$\left( \frac{V_2}{V_1} \right)^{-\gamma} = \frac{p_2}{p_1}. \quad (4.10)$$

By inserting the ideal gas law we finally get

$$p_1 V_1^\gamma = p_2 V_2^\gamma = \text{const.} \quad (4.11)$$

$$p_1^{1-\gamma} T_1^\gamma = p_2^{1-\gamma} T_2^\gamma = \text{const.} \quad (4.12)$$

$$T_1 V_1^{\gamma-1} = T_2 V_2^{\gamma-1} = \text{const.} \quad (4.13)$$

This process is reversible, which means we can compress the gas adiabatically and get back to the initial state using the same formula. To calculate the total work done during the process we use

$$W = - \int_{V_1}^{V_2} p dV \quad (4.14)$$

$$= - \int_{V_1}^{V_2} p_1 \left( \frac{V_1}{V} \right)^\gamma dV \quad (4.15)$$

$$= -p_1 V_1^\gamma \left[ \frac{1}{\gamma-1} V^{1-\gamma} \right]_{V_1}^{V_2} \quad (4.16)$$

$$= \frac{V_1 p_1 - V_2 p_2}{\gamma-1}. \quad (4.17)$$

For compression this value is positive, for expansion negative.

## 5 Mixtures of ideal gases

We now combine different ideal gases in a setup similar to Figure 1, but with the second gas in the right chamber. The resulting gas mix will act as an ideal gas with the properties

$$U_{mix} = \sum_i U_i \quad (5.1)$$

$$c_{V,mix} = \frac{\sum_i n_i c_{V,i}}{\sum_i n_i} \quad (5.2)$$

$$T_{mix} = \frac{\sum_i n_i c_{V,i} T_i}{\sum_i n_i c_{V,i}} \quad (5.3)$$

$$V_{mix} = \sum_i V_i \quad (5.4)$$

$$\gamma_{mix} = \frac{\sum_i n_i c_{p,i}}{\sum_i n_i c_{V,i}} \quad (5.5)$$

$$= \frac{\sum_i n_i (c_{V,i} + R)}{\sum_i n_i c_{V,i}}. \quad (5.6)$$

The internal energies and heat capacities are additive due to energy conservation. Note that merging a gas with an empty volume with  $V_2 > 0$ ,  $n_2 = 0$ ,  $p = 0$  is equivalent to free expansion.

## 6 Piston pump

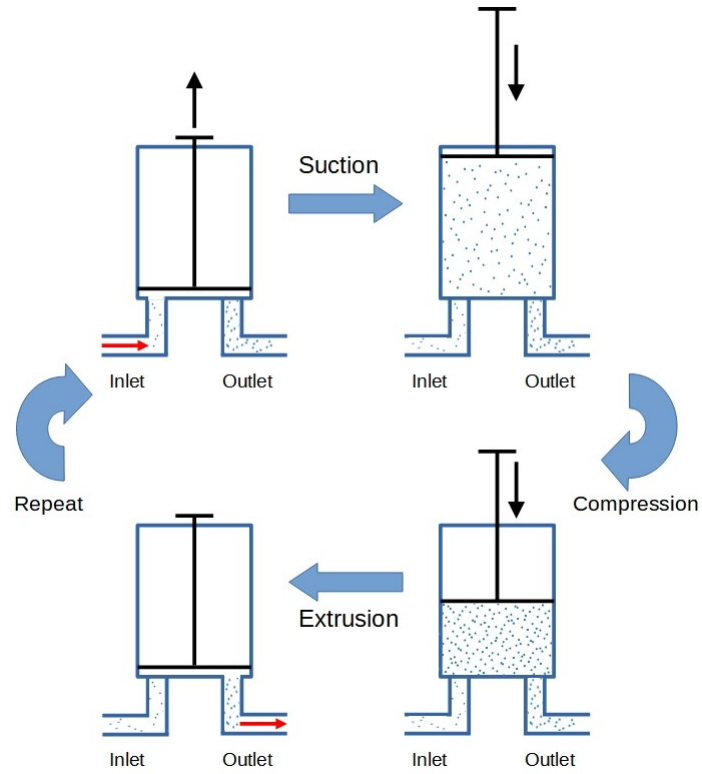


Figure 4: Adiabatic expansion

A simple piston pump has the following cycle (see Figure 4):

1. Suction: Open valve 1 and adiabatically expand the inlet gas from  $V_{in}$  to  $V_{in} + V_{pump}$ . Then close valve 1, which splits off the volume  $V_{pump}$  from the inlet gas.

$$V_{in} \longrightarrow V_{in} \quad (6.1)$$

$$n_{in} \longrightarrow n_{in} \left( \frac{V_{in}}{V_{in} + V_{pump}} \right) \quad (6.2)$$

$$p_{in} \longrightarrow p_{in} \left( \frac{V_{in}}{V_{in} + V_{pump}} \right)^{\gamma_{in}} = p_{pump} \quad (6.3)$$

$$T_{in} \longrightarrow T_{in} \left( \frac{V_{in}}{V_{in} + V_{pump}} \right)^{\gamma_{in}-1} = T_{pump} \quad (6.4)$$

$$n_{in} \left( \frac{V_{pump}}{V_{in} + V_{pump}} \right) = n_{pump} \quad (6.5)$$

2. Compression: Adiabatically compress the gas inside the pump from until it equals the output pressure. This step can be skipped if the output pressure is lower than the pressure inside the pump.

$$V_{pump} \longrightarrow V_{pump} \left( \frac{p_{pump}}{p_{out}} \right)^{\frac{1}{\gamma_{in}}} = V'_{pump} \quad (6.6)$$

$$n_{pump} \longrightarrow n_{pump} \quad (6.7)$$

$$p_{pump} \longrightarrow p_{out} \quad (6.8)$$

$$T_{pump} \longrightarrow T_{pump} \left( \frac{p_{pump}}{p_{out}} \right)^{\frac{1-\gamma_{in}}{\gamma_{in}}} = T'_{pump} \quad (6.9)$$

3. Extrusion: Open valve 2, causing the pump gas and outlet gas to mix, then compress the mixed gas adiabatically from  $V_{out} + V'_{pump}$  to  $V_{out}$  by moving the piston to its initial position. Close valve 2 and the cycle can restart.

$$V_{out} + V'_{pump} \longrightarrow V_{out} \quad (6.10)$$

$$n_{mix} \longrightarrow n_{mix} \quad (6.11)$$

$$p_{out} \longrightarrow p_{out} \left( \frac{V_{out} + V'_{pump}}{V_{out}} \right)^{\gamma_{mix}} \quad (6.12)$$

$$T_{out} \longrightarrow T_{out} \left( \frac{V_{out} + V'_{pump}}{V_{out}} \right)^{\gamma_{mix}-1} \quad (6.13)$$

TODO: Sum of work done for all three steps of the cycle. Find closed form solution to limit the pump volume  $V_{pump}$  by the pumps power usage.



We have two types of pumps in SS14: One is the volume pump, which has a maximum output pressure of  $9000kPa$  and a selectable transfer volume per cycle. The other one is the pressure pump, which has an infinite maximum transfer volume, but only transfers gas up to a selectable output pressure (maximum  $4500kPa$ ).

For the pump rework we would make both pumps work as described above with one cycle in each update step. The pressure pump should pump until a certain pressure threshold is met and then stop, but have a constant (but maybe higher) internal pump volume. The volume pump should have the internal pump volume  $V_{pump}$  be selectable like before, but have no maximum output pressure. If we limit the pumps throughput by its maximum energy usage, its effectiveness will be reduced at high pressure. These diminishing returns will stop the player from creating extremely high-pressured areas. Breaking pipes at extremely high pressures has also been suggested in the atmos roadmap.

## 7 Examples

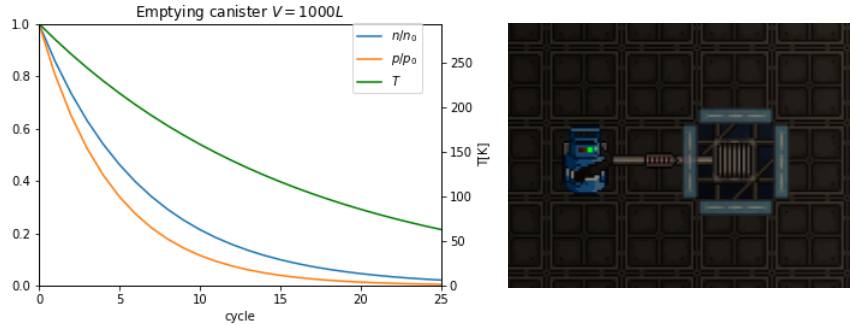


Figure 5: Removing gas from a canister

- Lowering the pressure in a gas container using a pump will now properly cool down the gas. The effects of the adiabatic expansion are shown in Figure 5. The pumps efficiency becomes lower with lower input pressure. The gas amount inside will decay exponentially, meaning that most of the gas can be removed quickly and the remaining trace amounts should not matter for any practical purposes. If you want to completely empty a container quickly you will have to connect it to space.

- Filling an empty pipe network with gas using a pump will not cause it to cool down since it is a free expansion. However, compressing it to a pressure higher than that of the pump input will cause it to heat up. This is good from a gameplay perspective: Atmos techs can no longer 'overclock' distro by making it extremely high pressure without also regulating the temperature. Leaving it at normal pressure won't influence the temperature and requires no additional setup. **TODO: diagram**

## 8 Implementation notes

- The pressure is a computed property in the `GasMixture` class and is only calculated when needed using  $pV = nRT$ .
- All the basic thermodynamic processes for gases should have a corresponding function in `AtmosphereSystem.Gases.cs`. Any state change done to gases falls into one of these categories, so this can prevent a lot of code duplication.
- The `Merge` function in `AtmosphereSystem.Gases.cs` should be renamed as it does not merge two gas mixtures, but instead only transfers the heat and moles from one gas to another, ignoring the volume. A new merge function should be written which properly combines two gas mixtures while adding their volume. This is often needed, for example when opening valves or connecting pipe networks when building. With this more code duplication can be prevented.
- Due to the theorem of equipartition of energy an ideal gas should have the molar heat capacity  $c_{v,m} = \frac{1}{2}fR$  where  $f$  is the number of degrees of freedom of the molecule. For a monoatomic ( $f = 3$ , 3 translational d.o.f.), diatomic ( $f = 5$ , 3 translational and 2 rotational d.o.f.) and triatomic ( $f = 6$ , 3 translational and 3 rotational d.o.f.) ideal gas this would result in  $c_{v,m} = \frac{3}{2}R$ ,  $\frac{5}{2}R$  and  $\frac{6}{2}R$  respectively. Our gases have different values meaning they do not behave like an ideal gas in all cases.
- Pumps currently run 8 times faster than shown to the player. This speedup is set in the `atmos.speedup` cvar. So if a volume pump is set to  $200L/s$  it actually pumps  $1600L/s$ , which is not shown to the player. This should be corrected by either removing the cvar or showing the correct speed in the clients interface. Also for energy conservation

reasons the speedup should not be able to circumvent the maximum transfer speed given by the pumps power usage.

- Currently filters and gas mixers also work as pumps. These will be changed to be flow based according to the atmos roadmap. This should probably be merged at the same time as this proposal.
- **TODO:** Automatically calculate  $\gamma$  from  $\gamma = \frac{c_{p,m}}{c_{v,m}} = \frac{c_{v,m}+R}{c_{v,m}}$  instead of using a datafield? Possibly necessary for energy conservation? Check the physics of non-ideal gases for that.