On the height of the Hunga Tonga-Hunga Ha‘Apai plume

Wout De Jonghe

August 2023

1 Abstract

The Hunga Tonga-Hunga Ha‘Apai volcano erupted violently in 2022, creating a massive plume reaching well into the mesosphere. This reports attempts to computationally recreate the conditions of the eruption to explain the plume‘s exceptional height and why it is so rare. It was found that the atmospheric conditions in Tonga were ideal for the formation of a plume with a very high equilibrium altitude. The collapse of the caldera caused the plume to reach this altitude with a high inertial velocity, hence the maximal height reached high above the equilibrium height. The Tonga volcano had access to a vast amount of water vapor due to its submarine nature. This high concentration of volatiles allowed for buoyancy under the aforementioned conditions, which would normally result in a collapse of the plume, explaining the rarity of the event.

2 Introduction

Despite a relatively large number of studies on the dynamics of volcanic plumes, it remains difficult to use quantitative tools to describe or predict many of the behaviours observed in some recent volcanic explosions. On the 15^{th} of January 2022, the Hunga Tonga-Hunga Ha‘Apai (HTHH) volcano erupted violently with the plume reaching a maximum height of about 57 km, well into the Mesosphere. Plumes this massive can have a large effect on air traffic and could potentially disturb the ozone layer in extreme cases. Models that could predict the dynamics of these plumes or even make estimations based on parameters known before the eruption have a potential to contribute in keeping the effects of eruptions under control, but they are not widely used at the moment. Especially the plume height is currently mostly used as a way to derive the mass eruption rate through a power law, initially proposed by Morton in 1956 [9]. The mass eruption rate is difficult to measure and the law is thus hard to test. It is also impossible to determine the mass eruption rate before an eruption or use it to predict any dynamics of the plumes. This makes the usefulness of this law questionable at best, especially since it has been shown to break down for plumes above 35 km [12]. Hence the need for a more sophisticated method using a complete dynamical model that can be solved numerically to describe the behaviour of plumes and potentially predict their future dynamics. In this report, a model described by Woods 1993 [14] will be implemented to justify the extraordinary maximum height of the HTHH volcano and describe some other phenomena observed in the plume. In the next section, the background and relevant information will be laid out, followed by a section on the model that will be used.

3 Hunga Tonga-Hunga Ha‘Apai eruption

The HTHH volcano lies in the Western South Pacific Ocean with coordinates 20.536°S, 175.382°W, about 2000km north of New Zeeland. The volcano has been relatively silent from 1900 to 2014 with only 5 recorded minor eruptions. After 2014, the volcano started to become more active, with a streak of eruptions in December 2021 [5]. Eventually culminating in the climactic eruption on the 15^{th} of January 2022, causing an eruption column up to 57km into the atmosphere and several tsunamis hitting nearby islands. [1]

The most notable features of the explosion were the caldera collapse and its submarine nature: the volcano was submersed under about 200 m of ocean. This in combination with the caldera collapsing allowed for a vast amount of water to enter the magma chamber, adding a large quantity of volatiles to the eruption plume.

At an altitude of approximately 30-35 km, clouds were observed around the volcanic plume, this observation was further confirmed by the measurement of large quantities of water in the stratosphere at around this altitude. [8]

4 The model

The model outlined in the following section was first implemented by Andrew W. Woods [14] and used for all the calculations in this report. The model is based on the conservation of mass, momentum, energy and water content and includes the effects of water vapor and its condensation. The first equation involves the conservation of mass:

$$\frac{d}{dz}(\rho u b^2) = 2\epsilon b u \rho_a$$

where b denotes the plume radius, u the vertical speed, ρ the plume density and the subscript "a" denotes that the quantity relates to the ambient atmosphere and not the plume itself. ϵ is the entrainment coefficient: a dimensionless value that denotes how much of the ambient air is integrated into the plume. For the gas-thrust region it is a function of the density but once the plume is buoyant the entrainment coefficient becomes constant. The HTHH eruption is Plinian, suggesting thatr the gas-thrust region is very short compared to the buoyant region and ϵ will thus be taken to be a constant 0.09 [14]. The following equation regards the conservation of momentum:

$$\frac{d}{dz}(\rho u^2 b^2) = b^2 g(\rho_a - \rho)$$

with the symbols having the same meaning as in the previous equation and g being the gravitational constant. As the effect of liquid water and water vapor are considered, an equation conserving the total water content is required:

$$\frac{d}{dz}(\rho u b^2 q) = 2\epsilon b u q_a \rho_a$$

where q represents the total mass fraction of water, i.e. liquid water and water vapor combined. The energy equation reads:

$$\frac{d}{dz}[\rho u b^2(C_p T + u^2/2 + gz)] = 2\epsilon b u \rho_a(C_p T_a + gz) - [S + (C_{pv} - C_{pw})(T - 273)]\frac{d}{dz}(u b^2 q_a q_v)$$

where C_p is the specific heat at constant pressure in the column and q_v is the mass fraction of water vapor. All other parameters are constants and are given with their value and justification in Appendix A. This concludes the differential equations necessary for the model, but there are still more variables than equations. This can be solved by introducing constraints to the model. The first is for the ambient temperature, which will be constraint by meteorological data, and the pressure can then be determined from the hydrostatic balance:

$$\frac{d}{dz}P = -\frac{gP}{R_a T}$$

which defines the density through the ideal gas law. It is assumed that small sediments in the plume won't have enough time to form larger rocks that would fall out of the plume during its rise, hence the amount of solid material is conserved:

$$\frac{d}{dz}((1 - n - q_w)\rho u b^2) = 0$$

where $q_w = q - q_v$ is the mass fraction of condensed water in the column. To find the water vapor level of the column and atmosphere the saturation vapor pressure is used:

$$e_s = A\exp(-B/T_a)$$

. It is assumed that the column can be considered as just saturated:

$$\frac{w}{1+w}P = e_s$$

This assumption is valid in the column, and hence $q_v = nw$. For the ambient air a constant, R, which varies between 0 and 1 is introduced to be multiplied by w, depending on the moist level of the atmosphere ($q_a = Rw$). The island lies in the tropical region, hence $R = 1$ will be used throughout the report. Next, it is possible to write the total specific heat as a weighted sum of the specific heat of the different phases:

$$C_p = (1 - n - q_w)c_{vs} + nc_{vg} + q_w C_{pw}$$

.The density inside the column can be constraint by:

$$\rho = \left[\frac{n}{\rho_g} + \frac{1-n}{\rho_s} \right]^{-1}$$

where ρ_s is the combined density of solids and liquids in the plume. And ρ_g is the density of gasses in the plume and can also be found through the ideal gas law:

$$\rho_g = \frac{P}{R_{ga}T}$$

with $R_{ga} = wR_v + (1 - w)R_a$ being the bulk gas constant. For a gas mass fraction above 0.1 (which it is for the vast majority of the plume), the first term in the density equation starts dominating and the approximation

$$\rho = \frac{\rho_g}{n}$$

becomes valid.

5 Methods

The model will be used to explain the height the HTHH plume reached and attempt to understand the relevant parameters which have a large influence on the rise dynamics. The most important degrees of freedom in the model are the initial conditions for the parameters in the differential equations: u, b, T, n and q. For some of these parameters, it is possible to estimate the initial value from available data but most of them are hard to determine within a reasonable error. They will each be studied separately to deduce their effect on the plume height and dynamics. To achieve this a range of initial values will be used to find the velocity profile of the plume: this profile shows the speed of the particles of the plume at each altitude level of their ascent. The maximum height is where the velocity hits zero, and the equilibrium height of the cloud is where the gradient turns zero. The first parameter that will be studied is the initial vent radius, b_0. The eruption was accompanied by a caldera collapse, and the caldera radius of about 1.5 km can thus be taken as the initial vent radius for HTHH. To study the effect of this large radius, several values will be studied, including ones that would imply no caldera collapse. The second parameter is the initial velocity, u_0. A typical Plinian eruption has initial velocities between 75 and 250 m s^{-1}, and this range will be discussed.The initial temperature range that will be considered is 800 K-1200 K.

Due to the vast amount of water vapor added to the incoming magma, it can be assumed that $n_0 = q_0$. This approximation is justified as gas fractions purely from volcanic origin are of the order of 0.01. This value will be shown to be too low for the observed dynamics of the HTHH volcano. The extra gas needed to drive the eruption is expected to come from the large amount of sea water and vastly outnumbers any other gasses present. Because the exact amount of added water vapor is difficult to gauge and can have a large effect on the plume dynamics, its variation will be considered in each of the above cases. Looking at values between 0.01 (as a comparison with a normal eruption), and 0.8, for a large amount of water being dropped in.

A secundary objective of the report is to find a good estimation for the water vapor added to the eruption based on stratospheric water levels in the period around the eruption. Measurements from ACE on heavy water (HDO/D_2O) concentration will be used to compare the contribution of water coming from the vent, the ambient air or from the sea water. These sources each have different initial isotope concentrations:

Source	Isotope ratio (Delta D ‰)	Isotope ratio (ppm)
Sea water	0	156 [6]
Vent	700	265 [10]
Unperturbed stratosphere	-500	78
Stratosphere after eruption	-200	125

This data can be used to put constraints on the allowed compositions of water in the eruption plume in the stratosphere. Assuming that the 3 mentioned sources are the only ones that influence the final composition, a system with 2 equations and 3 unknowns was set up:

$$s + v + a = 1$$

$$156s + 265v + 78a = 125$$

Where s ,v and a refer to the relative fractions that comes from each of the 3 sources in the final composition. As there are 3 unknowns and only 2 equations, it is only possible to find trends and constraints on the parameters.

6 Results

6.1 Initial plume radius

	$b_0 = 50$	$b_0 = 100$	$b_0 = 150$	$b_0 = 750$	$b_0 = 1000$	$b_0 = 1500$
$n_0 = 0.01$	2107	1333	1263	1199	1196	1194
$n_0 = 0.05$	45033	46234	47231	1471	1448	1427
$n_0 = 0.1$	45534	45531	47354	2237	2061	1937
$n_0 = 0.2$	45199	46331	47318	52862	54832	58056
$n_0 = 0.4$	45224	45522	46193	52023	56000	60588
$n_0 = 0.6$	45611	45892	47015	53387	55800	61257
$n_0 = 0.8$	45276	46792	46515	52749	56812	61198

Table 1: The maximum height the plume reaches for the given values of initial plume radius and gas mass fraction, the initial water vapor mass fraction is assumed to be equal to the total gas mass fraction. Vent radii and maximum heights are given in m. $u_0 = 150$ m s^{-1} and $T_0 = 1000$ K.

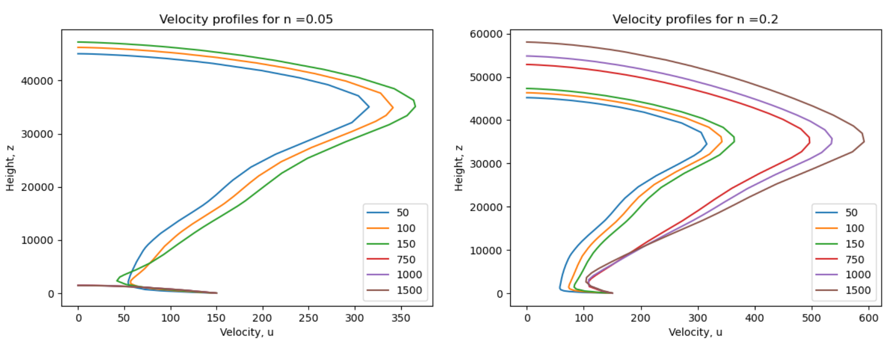

Figure 1: $T_0 = 1000$ K and $u_0 = 150$ m s^{-1}. Plot of the evolution of the velocity of the particles in the plume as they ascent to the top, with varying values for the initial plume radius.

As seen in Figure 1, for $n_0 = 0.05$, a ratio expected in most eruptions, only small vent eruptions manage to become buoyant but reach a smaller maximal altitude as they have lower inertia. Only for higher values of n_0, which can only realistically be reached in cases like HTHH, plumes with a high b_0 become buoyant.

6.2 Initial plume velocity

	$u_0 = 50$	$u_0 = 75$	$u_0 = 100$	$u_0 = 150$	$u_0 = 200$	$u_0 = 250$
$n_0 = 0.01$	132	297	529	1194	2130	3345
$n_0 = 0.05$	155	349	624	1427	2606	4255
$n_0 = 0.1$	196	447	810	1937	3907	57794
$n_0 = 0.2$	451	1223	55093	58056	58540	59130
$n_0 = 0.4$	53897	54996	57046	60588	60811	62073
$n_0 = 0.6$	56603	56818	58236	61257	61523	63852
$n_0 = 0.8$	55179	56608	58618	61198	63457	64465

Table 2: The maximum height the plume reaches for the given values of initial plume velocity and gas mass fraction, the initial water vapor mass fraction is assumed to be equal to the gas mass fraction. Initial velocities are given in m s^{-1} and maximal altitudes in m. $b_0 = 1500$ m and $T_0 = 1000$ K.

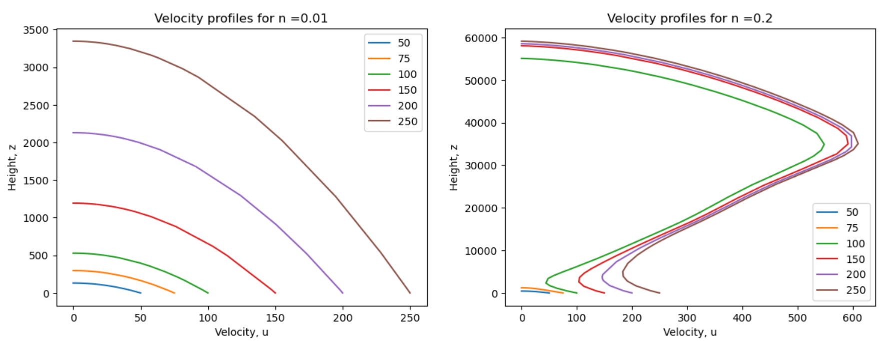

Figure 2: $T_0 = 1000$ K and $b_0 = 1500$ m. Plot of the evolution of the velocity of the particles in the plume as they ascent to the top, with varying values of v_0.

As seen in Figure 2, for $n_0 = 0.05$, none of the plumes become buoyant, this is due to the high value of b_0 chosen to conform with the HTHH case. A higher value of n_0 shows that once a minimum initial velocity of 100 m s^{-1} is reached such that the plume becomes buoyant, it has very little further effect on the dynamics of the plume.

6.3 Initial plume Temperature

	$T_0 = 700$	$T_0 = 800$	$T_0 = 900$	$T_0 = 1000$	$T_0 = 1100$	$T_0 = 1200$
$n_0 = 0.01$	1177	1182	1188	1194	1200	1206
$n_0 = 0.05$	1314	1347	1385	1427	1475	1530
$n_0 = 0.1$	1540	1642	1769	1937	2178	2585
$n_0 = 0.2$	2444	3387	56875	58055	56744	57885
$n_0 = 0.4$	61931	57934	58842	60588	58839	58441
$n_0 = 0.6$	60934	60604	60922	61257	60253	58946
$n_0 = 0.8$	62706	63040	61482	61198	59979	61049

Table 3: The maximum height the plume reaches for the given values of initial plume temperature and gas mass fraction, the initial water vapor mass fraction is assumed to be equal to the gas mass fraction. Initial temperatures in K and heights again in m. $u_0 = 150$ m s^{-1} and $b_0 = 1500$ m.

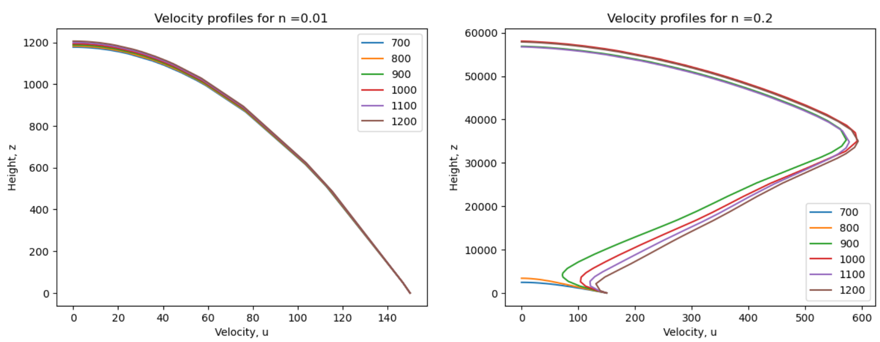

Figure 3: $u_0 = 150$ m s^{-1} and $b_0 = 1500$ m. Plot of the evolution of the velocity of the particles in the plume as they ascent to the top, with varying values of T_0.

For $n_0 = 0.05$, again the plumes do not become buoyant for similar reasons as in Figure 2 and the plume height is independent of temperature. For $n = 0.2$ a similar dependency as u_0 is observed: a threshold value of T_0 is required for buoyancy, but afterward the dynamics of the plume are very little affected by the initial temperature.

6.4 Water composition

Figure 4 shows that the percentage coming from sea water must lie in between 9.1% and 57%. The percentage of entrained air lies in between 41% and 70% and in between 0 and 21% for what percentage of the water comes from the volcanic vent.

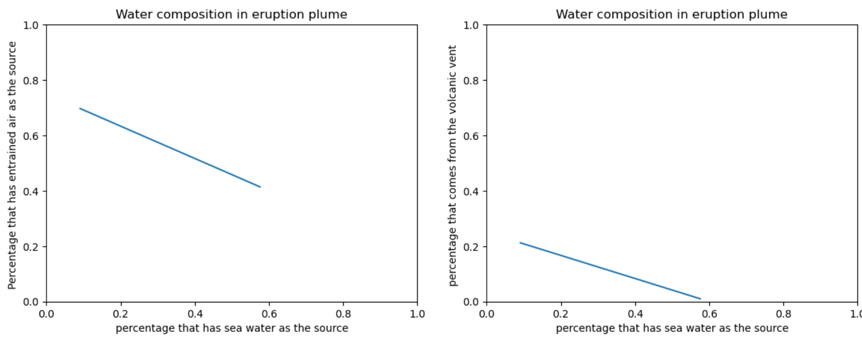

Figure 4: The constraints on the allowed composition of the water found in the HTHH eruption plume. To match the measured isotope ratios, the amount of water that came from the volcanic vent can not exceed 20%.

7 Discussion

7.1 Initial plume radius

Out of the 6 values chosen, the first three represent a case without caldera collapse with relatively small vent radii. The higher values represent very large initial vents which would correspond to a caldera collapsing. The collapsed HTHH caldera was estimated to have a 1.5 km radius, hence the final column from Table 1 is most likely to be aligned with the studied case. For gas mass fraction around 0.02 only plumes with relatively small vent radii become buoyant. Higher values of b_0 reach low peak altitudes as the column collapses. Only for a vapor mass fraction higher than to be expected from a normal eruption can these plumes become buoyant. These plumes reach a much higher maximum velocity than the small vent cases. Despite the equilibrium height being at roughly the same altitude, inertia manages to push the plumes with a large initial vent much higher, up to 58 km.

7.2 Initial plume velocity

Similar to the last section, Figure 2 shows that a small vapor mass fraction doesn‘t allow for buoyancy to occur (given a 1500 m initial vent radius as with HTHH). Then the maximum height is heavily reliant on initial velocity as the motion is almost purely ballistic. For a larger vapor mass fraction, as in the HTHH case, buoyancy occurs at reasonable initial velocities. For a vapor mass fraction of 0.2, an initial velocity of 100 m s^{-1} is required. Figure 2 shows that the peak altitude is not heavily dependent on initial velocity once the plumes become buoyant. This suggests that it is not necessary to acquire precise measurements on initial velocities, which can be hard to determine, as long as the buoyancy conditions are met.

7.3 Initial plume Temperature

Temperature has a similar effect as initial velocity on the maximum height reached by the plume. For low vapor mass fractions the plumes are unable to become buoyant and only reach low altitudes. For higher vapor mass fractions the initial temperature is only relevant to determine whether the plume becomes buoyant. Once this condition is met, the variable has little effect on the eventual plume height. For $n_0 = 0.2$, buoyancy occurs from 900 K onwards.

7.4 Equilibrium height

The equilibrium height can be inferred by finding the altitude at which the velocity gradient is zero, and hence the plume starts slowing down. Non-buoyant plumes have no equilibrium height as they collapse ballistically. In the buoyant case, there will be a height at which the plume density equals the ambient air density and equilibrium is reached. The inertial speed can cause the maximum height of the plume to be much higher than the equilibrium height, but most of the material will oscillate back and forth until they settle on the equilibrium altitude. Figures 1-3 show that the previously discussed factors (initial radius, speed and temperature) have little effect on the equilibrium height if the plume becomes buoyant. In all cases, the equilibrium height is roughly at the 35 km threshold, as expected from the data on vapor loading in the atmosphere. What does have a big effect on the equilibrium height is the temperature profile of the atmosphere. It is possbile to investigate the effect of different atmospheric temperature profiles by keeping the stratosphere constant (as it is mostly invariant under tropospheric changes), but changing the ground level temperatures, together with the lapse rate to ensure that the temperature is the same at the tropopause. As seen in Figures 5 and 6, different atmospheres based on different ground-level temperatures give vastly different results for equilibrium and maximum heights. With notably 298 K (the forecasted temperature for the time of the eruption) giving the highest maximal altitude and an equilibrium height at the expected 35 km. This explains the stratosphere water levels and clouds being formed at this altitude as most of the water being carried through the plume would end up at the equilibrium height. Figure 5 shows that the ground-level temperature has no effect on whether the plume manages to become buoyant. It only affects what equilibrium height and maximal height the plumes will reach.

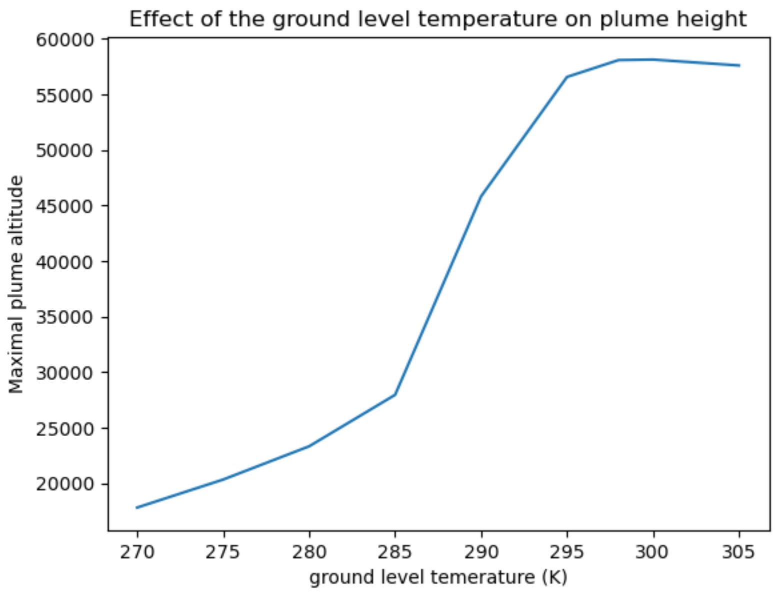

Figure 5: The effect of the ambient temperature at ground level on the maximal plume height for the values expected in the HTHH eruption (u_0 = 150 m s^{-1}, b_0 = 1500 m, T_0 = 1000 K and n_0 = 0.2). Note the steep rise from about 290 K, with a maximum at around 298K.

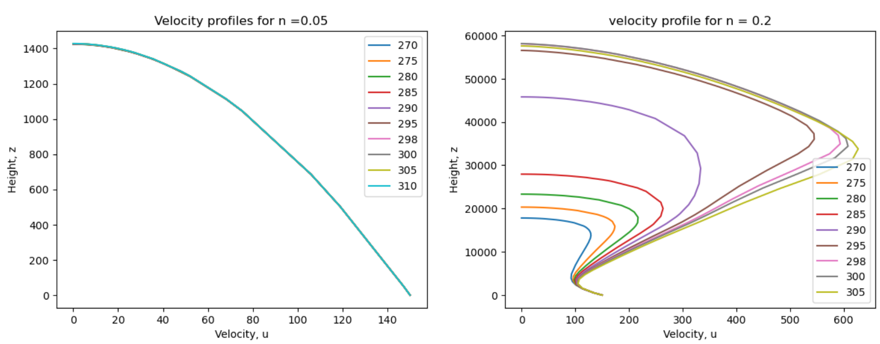

Figure 6: u_0 = 150 m s^{-1}, T_0 = 1000 K and b_0 = 1500 m. Plots of the velocity profiles for different atmosphere temperature profiles by changing the ground-level temperature and tropospheric lapse rate to ensure that the profile is invariant from the stratosphere onwards. Note that the forecasted temperature in Tonga on the day of the eruption was 298 K

8 Conclusion

In light of the previous results, an explanation for the extraordinary height of the HTHH plume can be given. In general, the eventual maximal plume height is dependent on the equilibrium height of the plume and its inertial velocity when going through this point, but only if the buoyancy conditions are met. The equilibrium height is almost exclusively defined by the properties of the ambient air and not much by the initial properties of the plume itself. The inertia of the plume when traversing the equilibrium point is dependent on the initial properties of the plume, the initial velocities and temperature have a relatively minor effect within the expected ranges of a volcanic eruption. The initial plume radius on the other hand has a much larger effect. Since the radius can differ by several orders of magnitude depending on the vent‘s dimensions or a potential caldera collapse, the inertia can differ drastically. This explains the apparent correlation between plume height and mass eruption rate discussed in previous work [9]. The last factor to be considered is the previously mentioned buoyancy conditions. Plumes with high initial vent radii tend to collapse more often than smaller plumes, they require a much higher water vapor concentration to become buoyant.

Comparing these general results with the HTHH observations, its eventual height can be explained. The forecasted ground temperature of 298 K together with the entire temperature profile were ideal for a maximally large plume as given in Figure 5. Due to the collapse of the caldera an exceptionally broad plume was formed that allowed the plume to reach equilibrium altitude with a high enough velocity and inertia to reach its eventual height. And lastly, these conditions didn‘t result in a plume collapse, which would normally be expected, due to the added water vapor from HTHH‘s submarine nature. Hence the conditions were ideal for a high plume to be formed. Using the expected values for the HTHH parameters gives the velocity profile seen in Figure 7, with a maximum altitude of 58 km. Several quantities related to the plume dynamics can be extracted from the velocity. For example by integrating the inverse velocity over the height, the total time it takes for the first particles to reach the top can be determined by using the programme shown in appendix X. For the profile from Figure 7, the expected time would be about 4 minutes. For less climactic eruptions without caldera collapse, it tends to take closer to 10 minutes.

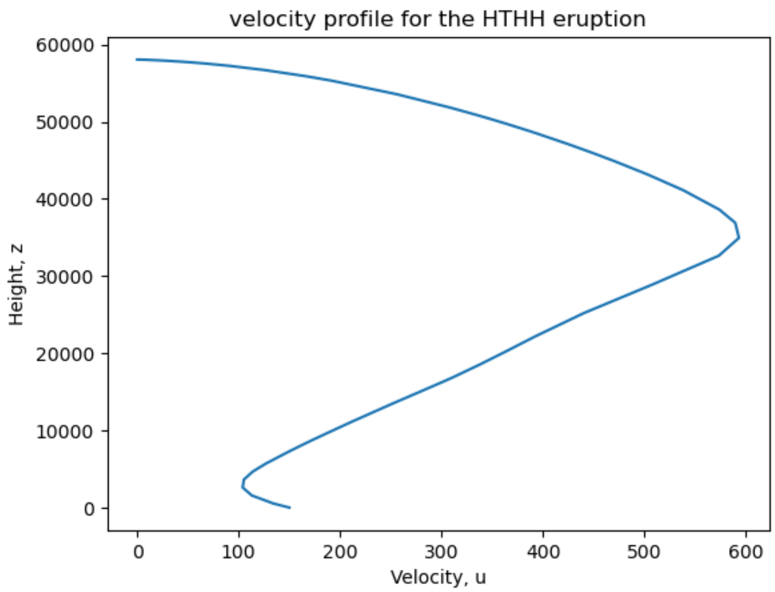

Figure 7: Velocity profile for the HTHH eruption. Initial conditions of $u_0 = 150$ ms^{-1}, $b_0 = 1500$ m, $T_0 = 1000$ K, $n_0 = 0.2$ and $q_0 = 0.2$ were used, in light of the expected values for HTHH. The equilibrium height comes out at 35 km, and the maximum height at 58km.

To use the model to potentially predict the dynamics of other plumes some amendments would need to be made in future research. For example, the effect from strong winds should be included as they are shown to have an effect on plume height [3][13]. Due to the tropic nature of the Tonga region, a moist atmosphere approximation can be made, this might not always be the case. The easiest solution would be to multiply the water vapor mass fraction by a constant between 0 and 1, as discussed by Woods [14]. The general use of the model should stay the same: the model can use weather forecasts to determine the likely equilibrium height for a potential eruption of an active volcano. The likelihood of collapse or buoyancy can be determined through the size of the vent or caldera in case it collapses. The hardest parameters to determine would be the vapor concentration and the probability of caldera collapse, but it should be possible to make reasonable estimations.

9 Appendix A. List of constants with justifications

- $g = 9.81$ m s^{-2}: gravitational acceleration, assumed to be at a constant global average as horizontal or vertical differences will be minor compared to other errors.
- $R_v = 462$ J kg^{-1} K^{-1}: gas constant for vapor [14]
- $R_a = 462$ J kg^{-1} K^{-1}: gas constant for air [14]
- $A = 2.5310^8$ kg m^{-1} s^{-2} and $B = 5.4210^3$ K: empirically determined values for the constants in the saturation vapor pressure equation [14]
- $S = 2.2610^6$ J kg^{-1}: latent heat of vaporization of water at 273 K, assumed to be constant [7]
- $C_{pv} = 1400$ J K^{-1}: specific heat of vapour [14]
- $C_{pw} = 4170$ J K^{-1}: specific heat of liquid water [14]
- $c_{vs} = 1617$ J K^{-1}: typical heat capacity of solids in the volcanic explosion [11]
- $c_{vg} = 1155$ J K^{-1}: typical heat capacity for the gasses in a volcanic plume. Expected to differ due to the extra water vapor but not included as it doesn‘t affect the results significantly [11]
- $\rho_s = 1700$ kg m^{-3} : a typical density value for volcanic solid. [2]
- ϵ: entrainment coefficient (0.1 for buoyant and 0.06-0.07 for jets), taken as a constant 0.09 in the calculations [4]

References

[1] E. Astafyeva, B. Maletckii, T. D. Mikesell, E. Munaibari, M. Ravanelli, P. Coïsson, F. Manta, and L. Rolland. The 15 january 2022 Hunga Tonga eruption history as inferred from ionospheric observations. *Geophysical Research Letters*, 49(10):e2022GL098827, 2022.

[2] P. Dellino, F. Dioguardi, R. Isaia, R. Sulpizio, and D. Mele. The impact of pyroclastic density currents duration on humans: The case of the AD 79 eruption of Vesuvius. *Scientific Reports*, 11(1):4959, 2021.

[3] B. Devenish. Using simple plume models to refine the source mass flux of volcanic eruptions according to atmospheric conditions. *Journal of volcanology and geothermal research*, 256:118–127, 2013.

[4] S. A. Fagents, T. K. Gregg, and R. M. Lopes. *Modeling volcanic processes: the physics and mathematics of volcanism.* Cambridge University Press, 2013.

[5] H. T.-H. Ha'Apai. Report on Hunga Tonga-Hunga Ha'Apai (tonga)—february 2022. *Bulletin of the Global Volcanism Network*, 47(2), 2022.

[6] R. Hagemann, G. Nief, and E. Roth. Absolute isotopic scale for deuterium analysis of natural waters. absolute d/h ratio for smow 1. *Tellus*, 22(6):712–715, 1970.

[7] M. B. Kirkham. *Principles of soil and plant water relations.* Elsevier, 2023.

[8] L. Millan, M. L. Santee, A. Lambert, N. J. Livesey, F. Werner, M. J. Schwartz, H. C. Pumphrey, G. L. Manney, Y. Wang, H. Su, et al. The Hunga Tonga-Hunga Ha'Apai hydration of the stratosphere. *Geophysical Research Letters*, 49(13):e2022GL099381, 2022.

[9] B. R. Morton, G. I. Taylor, and J. S. Turner. Turbulent gravitational convection from maintained and instantaneous sources. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, 234(1196):1–23, 1956.

[10] W. Shanks, J. K. Boehlke, R. Seal, and S. Humphris. Stable isotopes in mid-ocean ridge hydrothermal systems: Interactions between fluids, minerals, and organisms. *Geophysical Monograph-American Geophysical Union*, 91:194–194, 1995.

[11] Y. J. Suzuki, T. Koyaguchi, M. Ogawa, and I. Hachisu. A numerical study of turbulent mixing in eruption clouds using a three-dimensional fluid dynamics model. *Journal of Geophysical Research: Solid Earth*, 110(B8), 2005.

[12] L. Wilson and G. Walker. Explosive volcanic eruptions-VI. Ejecta dispersal in plinian eruptions: the control of eruption conditions and atmospheric properties. *Geophysical Journal International*, 89(2):657–679, 1987.

[13] M. Woodhouse, A. Hogg, J. Phillips, and R. Sparks. Interaction between volcanic plumes and wind during the 2010 Eyjafjallajökull eruption, Iceland. *Journal of Geophysical Research: Solid Earth*, 118(1):92–109, 2013.

[14] A. W. Woods. Moist convection and the injection of volcanic ash into the atmosphere. *Journal of Geophysical Research: Solid Earth*, 98(B10):17627–17636, 1993.

10 Appendix B. Code used for the general model

```
{
 "cells": [
  {
   "cell_type": "code",
   "execution_count": 3,
   "id": "9c2e4340",
   "metadata": {},
   "outputs": [],
   "source": [
    "### libraries being used for modelling and subsequent analysis ###\n",
    "\n",
    "import numpy as np\n",
    "from scipy.integrate import solve_ivp\n",
    "from scipy.optimize import fsolve\n",
    "from scipy.linalg import solve\n",
    "from matplotlib import pyplot as plt\n",
    "import scipy.integrate as integrate\n",
    "import math"
   ]
  },
  {
   "cell_type": "code",
   "execution_count": 4,
   "id": "fb797731",
   "metadata": {},
   "outputs": [],
   "source": [
    "### List of constants used in the model ###\n",
    "\n",
    "\n",
    "g = 9.81 #gravitational constant\n",
    "Rv = 462 # massive gas constant for water vapout\n",
    "Ra = 285 # massive gas constant for dry air\n",
    "R = 1 # moist level of the atmosphere (0 dry and 1 is moist)\n",
    "A = 2.53*10**8 # experimentally determined constand for the equilibrium pressure (es)\n
    "B = 5.42*10**3 # experimentally determined constand for the equilibrium pressure (es)\n
    "eps = 0.09 # entrainment coefficient\n",
    "cvs = 1617 # heat capacity of solids in the plume\n",
    "cvg = 1155 # heat capacity of liquids in the plume\n",
    "Cpv = 1400 # heat capacity of water vapour\n",
    "Cpw = 4170 # heat capacity of water\n",
    "Ls = 0.0012 #temperature lapse rate in stratosphere\n",
    "Lm = 0.00247 # temperature lapse rate in mesosphere\n",
    "T0=298 # initial air temperature at ground level\n",
```

```
  "p0=101325 # initial pressure at ground level\n",
  "rhos=1700 # density of pyroclasts\n",
  "S = 2260000 # latent heat of vapourisation at 273K\n",
  "\n",
  "### Temperature lapse rates at different marks for air pressure, determined from meteo
  "\n",
  "\n",
  "Pa = [100000,92500,85000,75000,70000,65000,60000,55000,50000,45000,40000,35000,30000,25
  "L = [1, 0.0005466666666666697, 0.0004533333333333033, 0.00055, 0.0004600000000000023,
 ]
},
{
 "cell_type": "code",
 "execution_count": 24,
 "id": "49982383",
 "metadata": {},
 "outputs": [],
 "source": [
  "### Model to solve the differentiql equation ###\n",
  "\n",
  "### Solver requires a guess for initial derivatives, should not effect runtime but a go
  "derivs_guess = [-1,-1,100,-1,0.01, -10, -5,0.01, 1, -1, -1, 0.01, 0.01,0.01, 1, -1, -1]
  "\n",
  "\n",
  "    \n",
  "### Next the system is defined, variables used in differential equations are ###\n",
  "### given as input for the function, whereas variables defined through constraints  ###
  "### are seperately written as such ###\n",
  "\n",
  "def system(t, TarhoabunTPqqw): # necessary to write all differential variables as 1 str
  "    Ta, rhoa, b, u, n, T, P, q, qw = TarhoabunTPqqw # string is unpact here\n",
  "    Cp = (1-n)*cvs + n*cvg\n",
  "    es = A*np.exp(-B/Ta)\n",
  "    w = es/(P-es)\n",
  "    qa = R*w\n",
  "    qv = n*w\n",
  "    Rga = w*Rv + (1-w)*Ra\n",
  "    rhog = P/(Rga*T)\n",
  "    rho = (n/rhog + (1-n)/rhos)**(-1)\n",
  "    \n",
  "    \n",
  "    ### all the equations need to be wriiten as differential equations ###\n",
  "    ### (including the constraints) and are given below ###\n",
  "    \n",
  "    \n",
  "    def derivatives(derivs):\n",
```

```
"        dTa, drhoa, db, du, dn, dT, dP, dq, dCp, des, dw, dqa, dqv, dqw, dRga, drhog, c
"        if P > Pa[1]:   # the rate of change of atmospheric pressure is dependent on th
"            eq1 = L[1]*dP - dTa\n",
"        if P > Pa[2] and P < Pa[1]:\n",
"                eq1 = L[2]*dP - dTa\n",
"        if P > Pa[3] and P < Pa[2]:\n",
"                eq1 = L[3]*dP - dTa\n",
"        if P > Pa[4] and P < Pa[3]:\n",
"            eq1 = L[4]*dP - dTa\n",
"        if P > Pa[5] and P < Pa[4]:\n",
"            eq1 = L[5]*dP - dTa\n",
"        if P > Pa[6] and P < Pa[5]:\n",
"            eq1 = L[6]*dP - dTa\n",
"        if P > Pa[7] and P < Pa[6]:\n",
"            eq1 = L[7]*dP - dTa\n",
"        if P > Pa[8] and P < Pa[7]:\n",
"            eq1 = L[8]*dP - dTa\n",
"        if P > Pa[9] and P < Pa[8]:\n",
"            eq1 = L[9]*dP - dTa\n",
"        if P > Pa[10] and P < Pa[9]:\n",
"            eq1 = L[10]*dP - dTa\n",
"        if P > Pa[11] and P < Pa[10]:\n",
"            eq1 = L[11]*dP - dTa\n",
"        if P > Pa[12] and P < Pa[11]:\n",
"            eq1 = L[12]*dP - dTa\n",
"        if P > Pa[13] and P < Pa[12]:\n",
"            eq1 = L[13]*dP - dTa\n",
"        if P > Pa[14] and P < Pa[13]:\n",
"            eq1 = L[14]*dP - dTa\n",
"        if P > Pa[15] and P < Pa[14]:\n",
"            eq1 = L[15]*dP - dTa\n",
"        if P > Pa[16] and P < Pa[15]:\n",
"            eq1 = L[16]*dP - dTa\n",
"        if P > Pa[17] and P < Pa[16]:\n",
"            eq1 = L[17]*dP - dTa\n",
"        if P > Pa[18] and P < Pa[17]:\n",
"            eq1 = L[18]*dP - dTa\n",
"        if P > Pa[19] and P < Pa[18]:\n",
"            eq1 = L[19]*dP - dTa\n",
"\n",
"        elif P < Pa[19] and t < 50000:\n",
"            eq1 = Ls - dTa\n",
"        if t > 50000:\n",
"            eq1 = -Lm - dTa\n",
"        eq2 = dP/(Ra*Ta) - P*dTa/(Ra*Ta*Ta) - drhoa # ideal gas law to find atmosphere
"        eq3 = eps*rhoa/rho - drho*b/(2*rho) - du*b/(2*u) - db # conservation of mass\n'
```

```
"        eq4 = 9.81*(rhoa - rho)/(2*u*rho) - drho*u/(2*rho) - u*db/b - du # conservation
"        eq5 = (1-n-qw)*(drho/rho + du/u + 2*db/b) - dqw - dn # conservation of solids\n
"        eq6 = 2*eps*rhoa*(g*t/Cp +Ta)/(b*rho) - (S+(Cpv-Cpw)*(T-273))/(Cp*rho)*(du*rhoa
"        eq7 = -g*P/(Rga*Ta) - dP # hydrostatic balance for atmosphere pressure\n",
"        eq8 = 2*eps*qa*rhoa/rho - q*drho/rho - 2*q*db/b - q*du/u - dq # conservation of
"        eq9 = dn*cvg -dn*cvs - dCp #heat capacity (constraint 1)\n",
"        eq10 = es*B*dTa/(Ta**2) - des #aturation pressure (constraint 2)\n",
"        eq11 = des/(P-es)-es*(dP-des)/(P-es)**2 -dw #constraint 3\n",
"        eq12 = R*dw - dqa #(constraint 4)\n",
"        eq13 = dn*w + dw*n - dqw #(constraint 5)\n",
"        eq14 = dq-dqv-dqw #rate of change of vapor must be opposite of rate of change o
"        eq15 = dw*Rv - dw*Ra - dRga #(constraint 6)\n",
"        eq16 = dP/(Rga*T) - P*dT/(Rga*T*T) - drhog #(constraint 7)\n",
"        eq17 = (dn/rhos + n*drhog/(rhog**2) - dn/rhog)*(n/rhog + (1-n)/rhos)**(-2) -drh
"        return [eq1, eq2, eq3, eq4, eq5, eq6, eq7, eq8, eq9, eq10, eq11, eq12, eq13, eq
"    \n",
"    global derivs_guess\n",
"    \n",
"    \n",
"    ### Loops to ensure the solver does it work. All variables need to be considered bu
"    ### variables are returned as the constraints can be retrieved later ###\n",
"    \n",
"    derivs = fsolve(derivatives, x0=derivs_guess)\n",
"    derivs_guess = derivs\n",
"    dTa, drhoa, db, du, dn, dT, dP, dq, dCp, des, dw, dqa, dqv, dqw, dRga, drhog, drho
"    return [dTa, drhoa, db, du, dn, dT, dP, dq, dqw] # dont return the constraint varia
"\n",
"\n",
"\n",
"### The solver solve_ivp is used to solve the system, the method requires an input of i
"### for the differential variables (as the constraint initial conditions can be calcula
"    \n",
"solution = solve_ivp(system, t_span=[0,90000], y0=[298, 1.225, 1500, 150, 0.2, 1000, 10
"\n",
"    \n",
"### Finally assign the solutions of the solver to the relevant parameters and redefine
"\n",
"t = solution.t\n",
"Ta,rhoa, b, u, n, T, P, q, qw = solution.y\n",
"Cp = (1-n)*cvs + n*cvg\n",
"es = A*np.exp(-B/Ta)\n",
"w = es/(P-es)\n",
"qa = R*w\n",
"qv = n*w\n",
"Rga = w*Rv + (1-w)*Ra\n",
"rhog = P/(Rga*T)\n",
```

```
    "rho = (n/rhog + (1-n)/rhos)**(-1)\n",
    "\n",
    "\n",
    "### These solutions can be plotted against each other or the height (t) itself to study
   ]
  }
 ],
 "metadata": {
  "kernelspec": {
   "display_name": "Python 3 (ipykernel)",
   "language": "python",
   "name": "python3"
  },
  "language_info": {
   "codemirror_mode": {
    "name": "ipython",
    "version": 3
   },
   "file_extension": ".py",
   "mimetype": "text/x-python",
   "name": "python",
   "nbconvert_exporter": "python",
   "pygments_lexer": "ipython3",
   "version": "3.10.12"
  }
 },
 "nbformat": 4,
 "nbformat_minor": 5
}
```


11 Appendix C. Code to find it takes for the plume to rise


```
{
 "cells": [
  {
   "cell_type": "code",
   "execution_count": null,
   "id": "7b05e5fd",
   "metadata": {},
   "outputs": [],
   "source": [
    "### code to find the total time taken for the first particles to reach the top of the p
    "### the veolcity array from the solver is integrated using a simple trapzium summation
```

```
    "### against the travelled distanc to find time ###\n",
    "\n",
    "\n",
    "time = 0\n",
    "y = np.reciprocal(u)\n",
    "for i in range(0, len(t)-1):\n",
    "    time += (y[i]+ y[i+1])*(t[i+1]-t[i])/2\n",
    "print(time)"
   ]
  }
 ],
 "metadata": {
  "kernelspec": {
   "display_name": "Python 3 (ipykernel)",
   "language": "python",
   "name": "python3"
  },
  "language_info": {
   "codemirror_mode": {
    "name": "ipython",
    "version": 3
   },
   "file_extension": ".py",
   "mimetype": "text/x-python",
   "name": "python",
   "nbconvert_exporter": "python",
   "pygments_lexer": "ipython3",
   "version": "3.10.12"
  }
 },
 "nbformat": 4,
 "nbformat_minor": 5
}
```

12 Appendix D. Code to find the constraints on water distribution

```
{
 "cells": [
  {
   "cell_type": "code",
   "execution_count": 1,
   "id": "88cf087b",
   "metadata": {},
```

```
   "outputs": [],
   "source": [
    "### required libraries for the calculations and subsequent analysis ###\n",
    "\n",
    "import numpy as np\n",
    "import matplotlib.pyplot as plt\n",
    "import math"
   ]
  },
  {
   "cell_type": "code",
   "execution_count": 2,
   "id": "c6f37264",
   "metadata": {},
   "outputs": [],
   "source": [
    "### Solution of the system of equations given at the end of section 5. ###\n",
    "### Eventual arrays are used in Figure 4 to find the constraints on the distribution ##
    "\n",
    "seaperc = np.linspace(0, 1, 100)\n",
    "ventperc = np.linspace(0, 1, 100)\n",
    "airperc = np.linspace(0, 1, 100)\n",
    "s_perc = []\n",
    "v_perc = []\n",
    "a_perc = []\n",
    "for z in airperc:\n",
    "    for y in ventperc:\n",
    "        for x in seaperc:\n",
    "            if math.isclose(x + y + z, 1) and math.isclose(155.76*x + 264.792*y + 77.88
    "                s_perc.append(x)\n",
    "                v_perc.append(y)\n",
    "                a_perc.append(z)"
   ]
  }
 ],
 "metadata": {
  "kernelspec": {
   "display_name": "Python 3 (ipykernel)",
   "language": "python",
   "name": "python3"
  },
  "language_info": {
   "codemirror_mode": {
    "name": "ipython",
    "version": 3
   },
```

```
   "file_extension": ".py",
   "mimetype": "text/x-python",
   "name": "python",
   "nbconvert_exporter": "python",
   "pygments_lexer": "ipython3",
   "version": "3.10.12"
  }
 },
 "nbformat": 4,
 "nbformat_minor": 5
}
```