Optimization of a Water Rocket in OpenMDAO/dymos

Bernardo Bahia Monteiro

Abstract

This work presents a model of a soda-bottle water rocket developed with NASA's OpenMDAO Dymos optimal control multidisciplinary framework. This is an accessible example that is able to highlight many of the benefits and challenges of multidisciplinary optimization and of collocation methods. Optimization results for flight range and height at apogee with respect to empty mass, initial water volume and launch angle are presented.

Nomenclature

$(\cdot)_0$	value of $(\cdot)$ at $t = 0$
$\dot{}$	time derivative
γ	flight path angle
ρ_w	water density
A_{out}	nozzle area
D	drag
g	acceleration of gravity
h	altitude
k	polytropic constant
m	mass
m_{empty}	empty mass
m_w	water mass
p	pressure in the rocket
p_a	ambient pressure
q	dynamic pressure
r	range
S	cross sectional area
T	thrust
t	time
v	flight speed
v_{out}	water exit speed at the nozzle
V_b	internal volume of the rocket
V_w	water volume in the rocket

1 Introduction

The soda-bottle water rocket, fig. 1, is a great problem to introduce basic physics principles to students, such as Newton's laws, ideal gas dynamics, Bernoulli's equation and aerodynamics [1]. The design of these rockets for optimal height or range, however, is not a trivial problem. It involves the numerical integration of a non-linear set of ordinary differential equations (ODE) and couples the disciplines of flight dynamics and propulsion, represented by the fluid dynamics inside the rocket bottle.

In spite of that, there are several water rocket competitions worldwide, so it is interesting to find a solution for this problem. In this work, we optimize a water rocket

Figure 1: Water Rocket

for range and height at the apogee, using design variables that are easily modifiable just before launch: the empty mass, which can be changed with the use of ballast, the initial water volume and the launch angle. We use this very accessible problem to show a simple but real world application of state of the art multidisciplinary optimization framework OpenMDAO [2] and its optimal control extension, Dymos [3].

This report is organized as follows. First, we present the model formulation with its division in components and implementation in OpenMDAO. We discuss the constants required by the model and our choice of optimization algorithm. Then, we proceed to investigate the model with the use of parameter sweeps and contour plots. Finally, we present and analyse the optimization results for both range and height at the apogee.

2 Problem Formulation

A natural objective function for a water rocket is the maximum height achieved by the rocket during flight, or the horizontal distance it travels, i.e. its range. The design of a water rocket is somewhat constrained by the soda bottle used as its "engine". This means that the volume available for water and air is fixed, the initial launch pressure is limited by the bottle's strength (since the pressure is directly related to the energy available for the rocket, it is easy to see that it should be as high as possible) and the nozzle throat area is also fixed. Given these manufacturing constraints, the design variables we are left with are the empty mass (it can be easily changed through adding ballast), the water volume at the launch, and the launch angle. With this considerations in mind, a natural formulation for the water rocket problem is

maximize	range or height
w.r.t.	empty mass, initial water volume, launch angle, trajectory
subject to	flight dynamics (eqs. (5) to (8))
	fluid dynamics inside the rocket (table 1)
	$0 <$ initial water volume $<$ volume of bottle
	$0^\circ <$ launch angle $< 90^\circ$
	$0 <$ empty mass

This problem is single objective (although the objective can be toggled), continuous, constrained, non-linear and deterministic.

2.1 Model

There are two main approaches to solving an optimization problems which involves time integration of a differential equation: direct integration (i.e. time marching) and collocation methods. In comparative study, Hwang and Munster [4] found that high order collocation methods such as Legendre-Gauss-Lobatto (LGL) are more robust and efficient than the alternatives.

For this work, we use the LGL collocation method to discretize the ordinary differential equations. In order to do this, each phase is discretized into a 21 node LGL segment, and the values of the states at the even collocation points are added as a design variable. The resulting approximation is differentiated with respect to time, and compared with the time derivative provided by the model. The difference between these values, called the *collocation defects* are added to the problem as *collocation constraints*, which the optimizer must drive to zero. Furthermore, the difference of the states in the end and beginning of subsequent phases are also added to the problem as *linkage constraints* [3].

This approach is a form of the Simultaneous Analysis and Design (SAND) approach, meaning that the optimizer is also responsible for solving the physics of the problem. This approach leads to a fairly large problem, in this case with 151 design variables and 151 functions of interest (1 objective function, 140 collocation constraints, and 10 linkage constraints).

In order to solve this problem formulation, we need a model capable of, given a trajectory and the remaining design variables, calculate the rocket's flight dynamic and fluid dynamic collocation and linkage defects and extract the maximum height and range achieved during flight. Such a model can be divided into three basic components: a water engine, responsible for modelling the fluid dynamics inside the rocket and returning its thrust; the aerodynamics, responsible for calculating the atmospheric drag of the rocket; and the equations of motion, responsible for propagating the rocket's trajectory in time, using Newton's laws and the forces provided by the other two components.

In order to integrate these three basic components, some additional interfacing components are necessary: an atmospheric model to provide values of ambient pressure for the water engine and air density to the calculation of the dynamic pressure for the aerodynamic model, and a component that calculates the instantaneous mass of the rocket

Table 1: Constitutive equations of the water engine model

Component	Equation
`water_exhaust_speed`	$v_{\text{out}} = \sqrt{2(p - p_a)/\rho_w}$ (1)
`water_flow_rate`	$\dot{V}_w = -v_{\text{out}} A_{\text{out}}$ (2)
`pressure_rate`	$\dot{p} = kp \frac{\dot{V}_w}{(V_b - V_w)}$ (3)
`water_thrust`	$T = (\rho_w v_{\text{out}})(v_{\text{out}} A_{\text{out}})$ (4)

by summing the water mass with the rocket's empty mass. The high level layout of this model is shown in fig. 2.

In the next section, the formulation of each component will be discussed. All models are analytically differentiated with OpenMDAO's framework.

2.1.1 Water engine

The water engine is modelled by assuming that the air expansion in the rocket follows an adiabatic process and the water flow is incompressible and inviscid, i.e. it follows Bernoulli's equation. We also make the following simplifying assumptions:

1. The thrust developed after the water is depleted is negligible
2. The area inside the bottle is much smaller than the nozzle area
3. The inertial forces do not affect the fluid dynamics inside the bottle

This simplified modelling can be found in Prusa [5]. A more rigorous formulation, which drops all these simplifying assumptions can be found in Wheeler [6], Gommes [7], Barrio-Perotti et al. [8]. Assumption 1 is known to underestimate the rocket performance, since the air left in the bottle after it is out of water is known to generate appreciable thrust [9]. This simplified model, however, produces physically meaningful results as will be shown in this text.

There are two states in this dynamical model, the water volume in the rocket V_w and the gauge pressure inside the rocket p. The constitutive equations are shown in table 1 and a diagram showing the model organization is in fig. 3.

2.1.2 Equation of motion component (`eom`)

The modelling of the rocket flight dynamics consist of a simple 2D model, described by the following equations

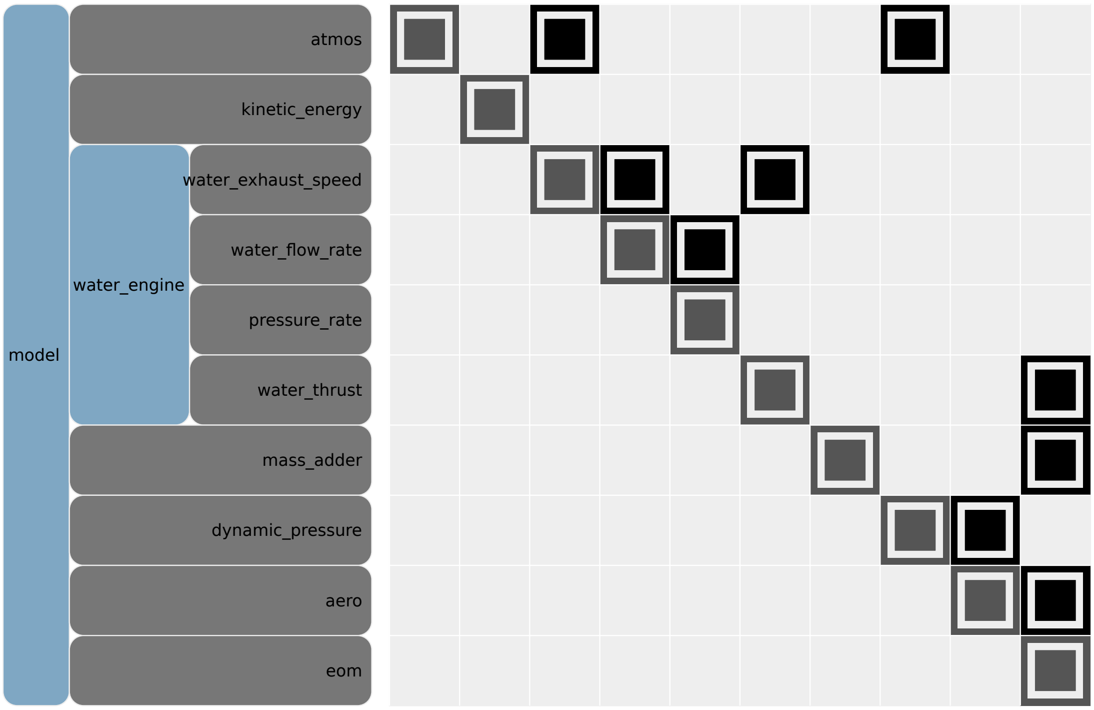


Figure 2: N2 diagram for the water rocket model

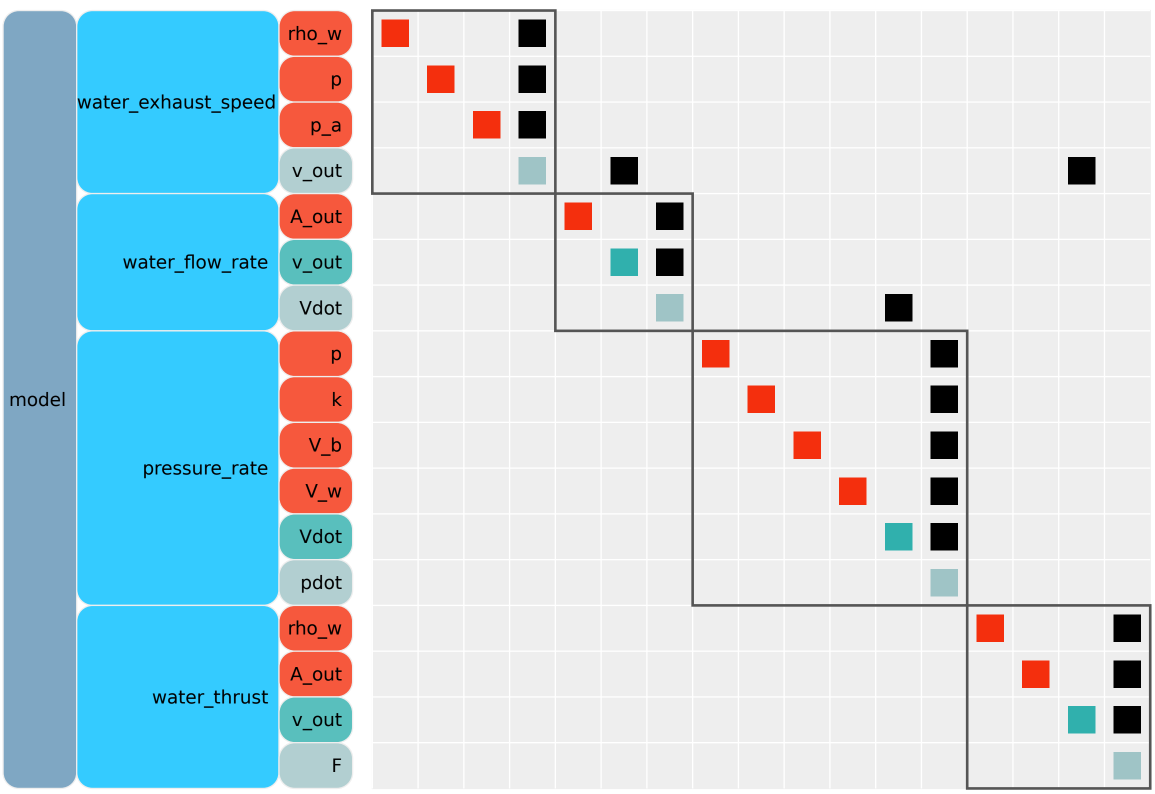


Figure 3: N2 diagram for the water engine group

$$
\begin{aligned}
\dot{v} &= \frac{T-D}{m} - g\sin\gamma && (5)\\
\dot{\gamma} &= \frac{T}{mv} - \frac{g\cos\gamma}{v} && (6)\\
\dot{h} &= v\sin\gamma && (7)\\
\dot{r} &= v\cos\gamma && (8)
\end{aligned}
$$

The model used by this work was implemented by Falck and Gray [3]. It is important to notice that this model has a singularity in the flight path angle derivative when the flight speed is zero. This happens because the rotational dynamics are not modelled.

2.1.3 Auxiliary components (`aero`, `dynamic_presure`, `mass_adder`, `atmos`)

The `atmos` group calculates the air density for use in the dynamic pressure calculation and the ambient pressure for the thrust calculation following the US 1976 model. It consists of several lookup tables and was developed by Falck and Gray [3], using data from https://www.digitaldutch.com/atmoscalc/index.htm. Since the range of altitudes achieved by the water rocket is very small (< 100m), the air density and pressure are practically constant, thus the use of an atmospheric model is not necessary. It was, however, included because it was readily available within Dymos' framework, and this was in fact easier than setting the values to constants.

The dynamic pressure component implements the dynamic pressure equation

$$q = \tfrac{1}{2}\rho v^2 \tag{9}$$

the aerodynamic model calculates drag via

$$D = qSC_D \tag{10}$$

and the mass adder calculates the rocket's instantaneous mass by summing the water mass with the rockets empty mass, i.e.

$$m = m_{\text{empty}} + \rho_w V_w \tag{11}$$

2.1.4 Phases

The flight of the water rocket is split in three distinct phases: propelled ascent, ballistic ascent and ballistic descent. If the simplification of no thrust without water were lifted, there would be an extra "air propelled ascent" phase between the propelled ascent and ballistic ascent phases.

Propelled ascent is the flight phase where the rocket still has water inside, and hence it is producing thrust. The thrust is given by eq. (4) from the water engine model, and fed into the flight dynamic equations eqs. (5) to (8). It starts at launch and finishes when the water is depleted, i.e. $V_w = 0$.

Table 2: Values for parameters in the water rocket model

Parameter	Value	Unit	Reference
C_D	0.3450	-	[10]
S	$\pi 106^2/4$	mm^2	[10]
k	1.2	-	[9, 11, 12]
A_{out}	$\pi 22^2/4$	mm^2	[13]
V_b	2	L	-
ρ_w	1000	kg/m^3	-
p_0	6.5	bar	-
v_0	0.1	m/s	-
h_0	0	m	-
r_0	0	m	-

Ballistic ascent is the flight phase where the rocket is ascending ($\gamma > 0$) but produces no thrust. This phase begins at the end of the propelled ascent phase and ends at the apogee, defined by $\gamma = 0$.

Descent is the phase where the rocket is descending without thrust. It begins at the end of the ballistic ascent phase and ends with ground impact, i.e. $h = 0$.

2.2 Model parameters

The model requires a few constant parameters. The values used for those are in table 2. Values for the bottle volume V_b, its cross-sectional area S and the nozzle area A_{out} are determined by the soda bottle that makes the rocket primary structure, and thus are not easily modifiable by the designer. The polytropic coefficient k is a function of the moist air characteristics inside the rocket. The initial speed v_0 must be set to a value higher than zero, otherwise eq. (6) becomes singular. This issue arises from the angular dynamics of the rocket not being modelled. The drag coefficient C_D is sensitive to the aerodynamic design, but can be optimized by a single discipline analysis. The initial pressure p_0 should be maximized in order to obtain the maximum range or height for the rocket. It is limited by the structural properties of the bottle, which are modifiable by the designer, since the bottle needs to be available commercially. Finally, the starting point of the rocket is set to the origin.

2.3 Optimization algorithm and derivatives computation

Due to the size of the resulting problem and the presence of many equality constraints, it is imperative to use a gradient based optimization algorithm. We choose SciPy's SLSQP algorithm, due to its availability and adequate performance for moderatelly large, equality constrained problems. This optimizer is a sequential quadratic programming optimizer.

It should be noted that since the problem has as many design variables as functions of interest, it does not benefit from the adjoint method for calculating derivatives. Therefore, the total derivatives were calculate from the partial derivatives using the direct method.

3 Investigation on the coupled models

To qualitatively evaluate the model, parameters sweeps with respect to empty mass (50–300g) and initial water volume(0.5-1.5L) are shown in figs. 4 and 5, with contours of final range, apogee height and optimum launch angle for maximum range. The range of the sweeps were chosen to cover a representative range of the search space. The empty mass is limited from below by zero and unlimited from above, while the initial water volume is limited from below by zero and from above by the bottle volume, in this case 2L. By doing a thought experiment, it is easy to see that with zero or maximum water volume there will be no thrust generated and therefore both range and height will be zero. The optimum must lie between these extremes. Likewise, with a very high mass the weight will be higher than the thrust and the rocket will not leave the ground. The very low mass case is slightly more complicated, but if we consider that the propelled phase is very short, even if the rocket gains a lot of speed in this phase it will not cover much distance. A very low mass means that the motion will be quickly halted by atmospheric drag once the propelled phase is over. Therefore, we do not expect the zero empty mass case to be optimal. These engineering considerations are confirmed by the contour plots, which suggest that the problem is unimodal as expected. This means that the choice of a local optimizer such as the SciPy's SLSQP is adequate.

These plots also show that the contours are smooth, which suggests low solver noise. This is confirmed by the good convergence tolerance achieved, in the order of 10^{-8} at moderate initial flight path angles and 10^{-6} up to 89°. For launch angles higher than this, convergence is very hard because the flight speed at the apogee becomes very low thus the flight path angle differential equation 6 becomes ill conditioned. In light of this, the value of γ was bounded to 89° for the optimization.

We found that the choice of transcription is the factor that most impact the optimizer performance. This happens because the transcription controls not only the size of the problem by introducing many collocation design variables and constraints, but also its stiffness and its coupling. The size of the problem is determined by the number of collocation points in the transcription; the coupling by the order of the transcription, since every point in the one transcription segment is coupled to the others; and the stiffness of the problem was found to be related to how accurate the chosen transcription can represent the true solution: if the transcription is too course the optimizer will have a hard time reducing the collocation defects. It was found that high order segments are ideal for having high accuracy in the representation of the solution while using fewer points than a low order transcription of similar accuracy, and the reduced problem size compensated the higher coupling. However, it was observed empirically that this came at the cost of more sensitivity to initial guesses, although more work is required to quantify these observations.

4 Optimization results

The optimum design variables, optimizer performance parameters and other selected parameters of interest can be seen in table 3. These results show good agreement with the parameters sweeps shown in section 3 It is interesting to notice that both the optimal

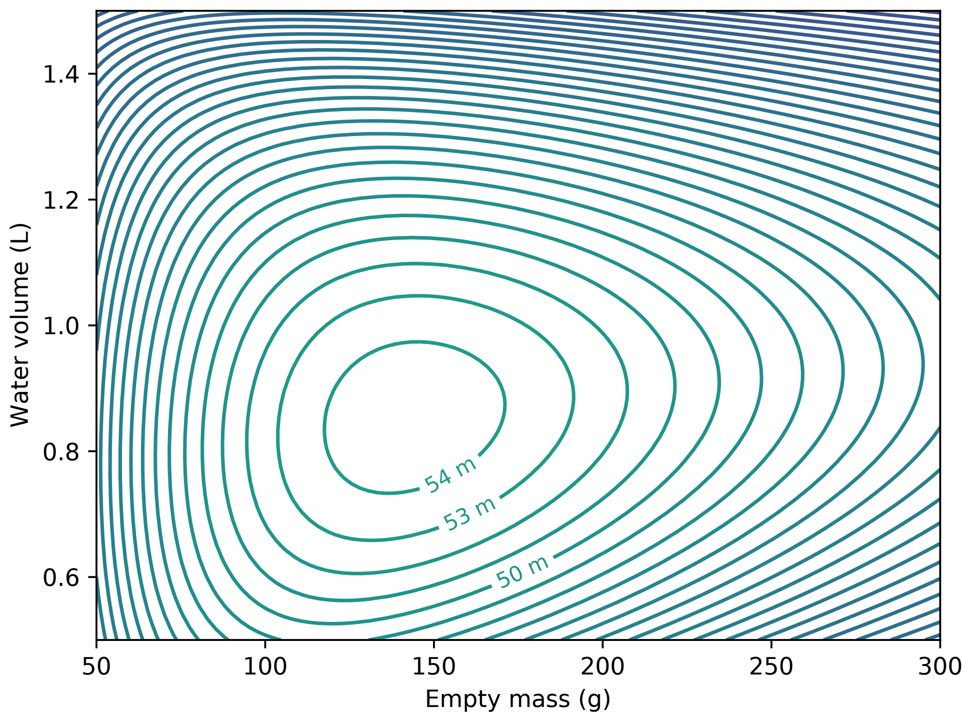


Figure 4: Height at apogee contours for the water rocket problem with launch angle fixed to 89°

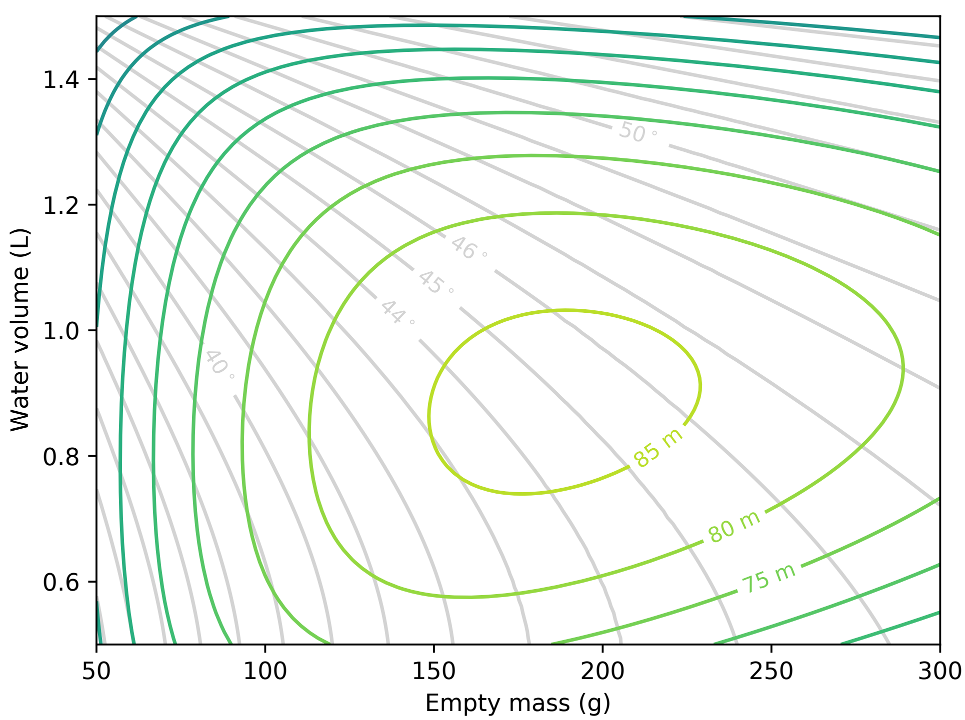


Figure 5: Range contours in green-yellow for the water rocket problem with optimal launch angle. Launch angle contours are in grey.

Table 3: Optimization results

	Maximum range	Maximum height
Empty mass	0.1861kg	0.1429kg
Water volume	0.9045L	0.8704L
Launch angle	45.0292°	89.0000°
Flight angle at end of propelled phase	38.3402°	88.8437°
Maximum range	86.4126m	3.2027m
Maximum height	23.5110m	54.6032m
Maximum velocity	42.4133m/s	47.2591m/s
Iterations	105	419
Function evaluations	584	3988
Gradient evaluations	105	419
Tolerance	1×10^{-8}	5×10^{-5}

empty mass and water volume are larger for the range optimization than for the height optimization. This suggests that the problem is truly multidisciplinary: if the water engine could be optimized by itself, for example by maximizing its impulse, the water volume would be the same for both cases. Instead, we see that this is not the case because the propulsion and flight dynamics disciplines are couple through mass.

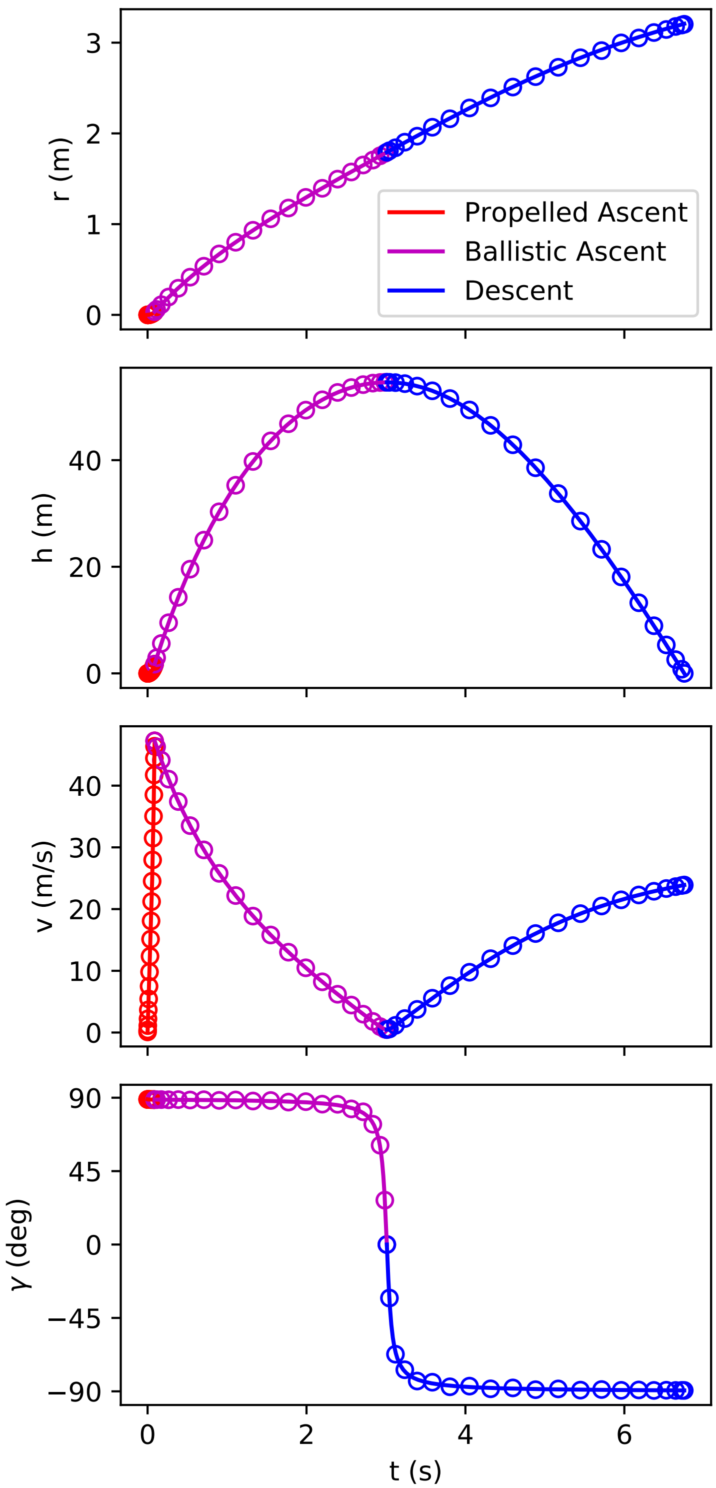
Due to the singularity at $\gamma = 90°$, the convergence for the height optimization is much worse than for the range optimization, both in number of iterations and function evaluations, even with a lower tolerance set for the former case.

It is interesting to notice that for the maximum range optimization, the launch angle is not smaller than 45° as would be expect for a ballistic flight with drag. This happens because at the small speeds of the beginning of the propelled ascent phase, the flight path angle changes very fast as shown in fig. 9. Because of this effect, at the end of the propelled phase the flight path angle is smaller than 45°, which agrees with the intuition from the ballistic case. This high non-linearity in the flight path angle also explains the need for a high order transcription.

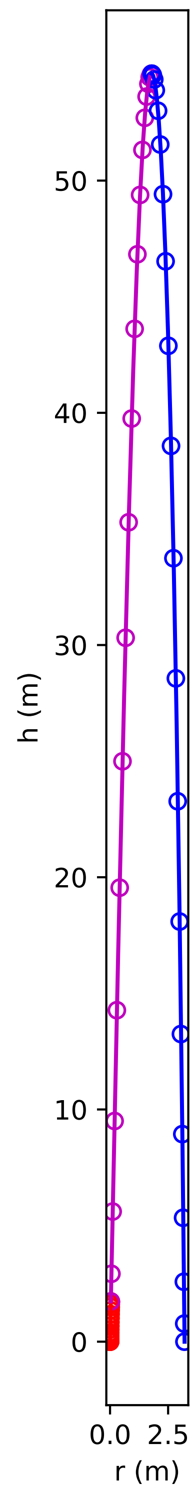
The trajectory of the rocket optimized for range can be seen in fig. 8, and the trajectory and selected states for the propelled phase for the rocket optimized for height can be seen in figs. 6 and 7 respectively. In the latter case, notice how fast the flight path angles change at the apogee. This due to the drastic reduction in speed at the apogee that is characteristic of launch angles close to vertical. Once again, these high non-linearities justify the choice of a high order transcription

5 Conclusion

In this work a model of a water rocket implement using the state of the art OpenMDAO Dymos framework was presented and optimized for both range and height at the apogee. This is a simple model that is able to illustrate well various concepts and challenges of multidisciplinary optimization and physical principles.



(a) flight dynamics states



(b) trajectory

Figure 6: Optimized trajectory for maximum height at apogee

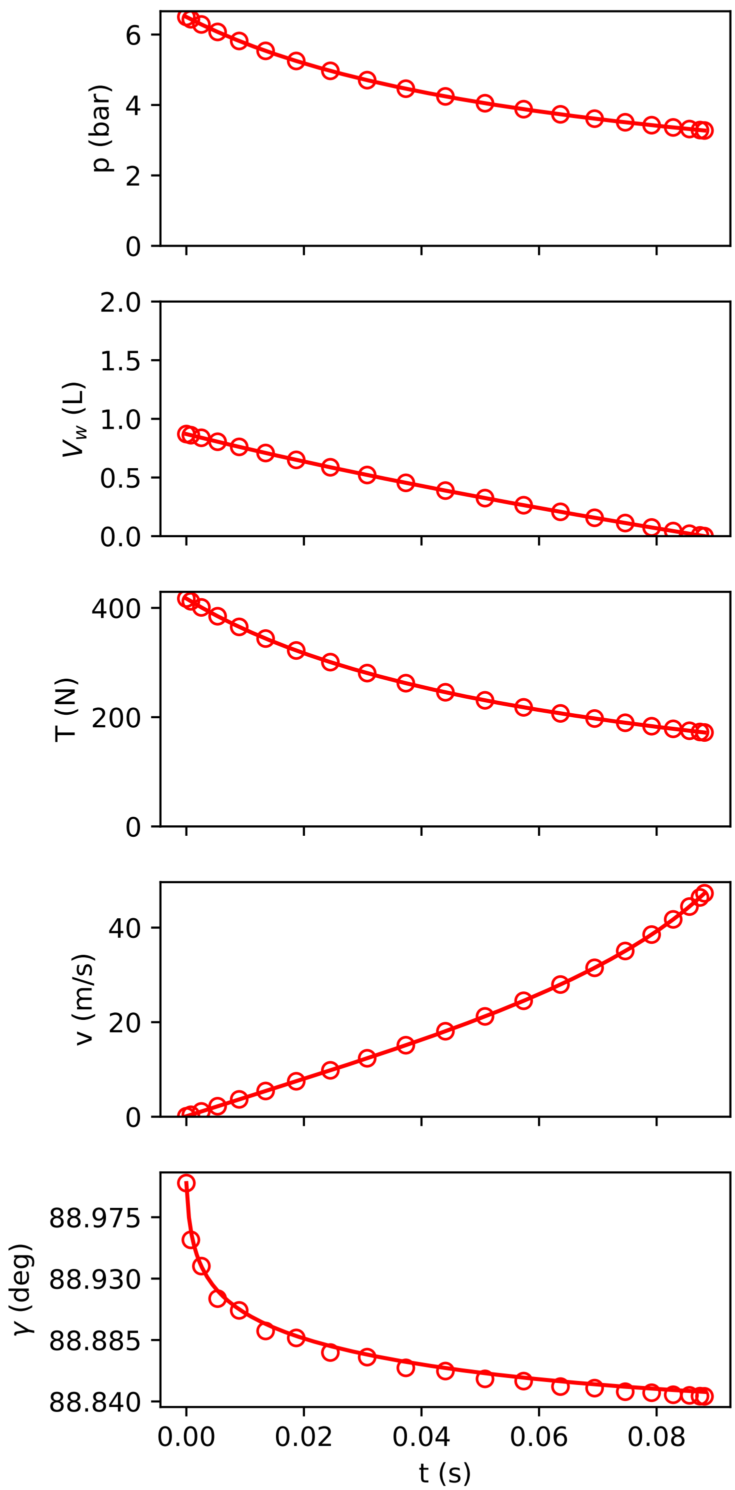
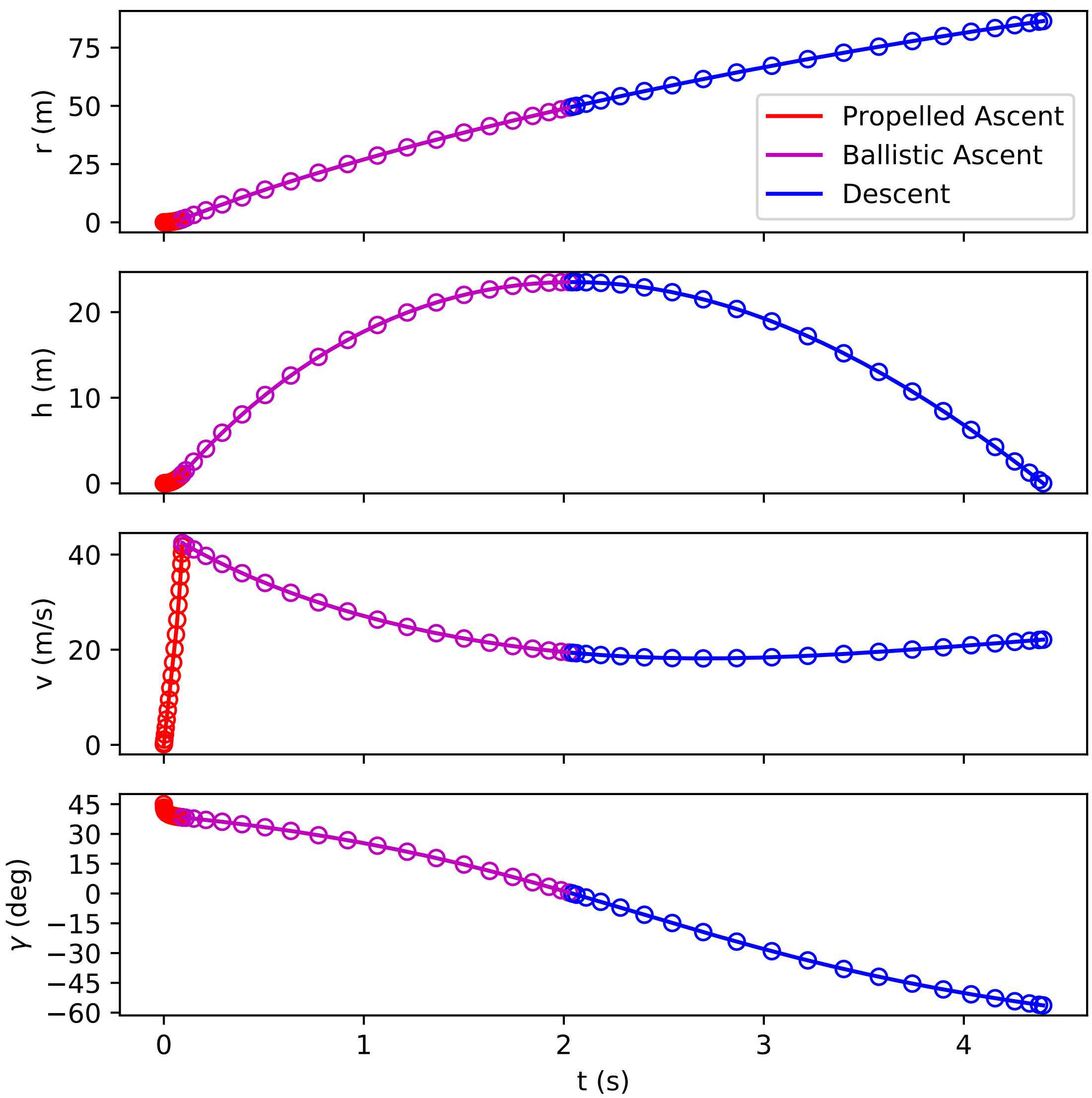
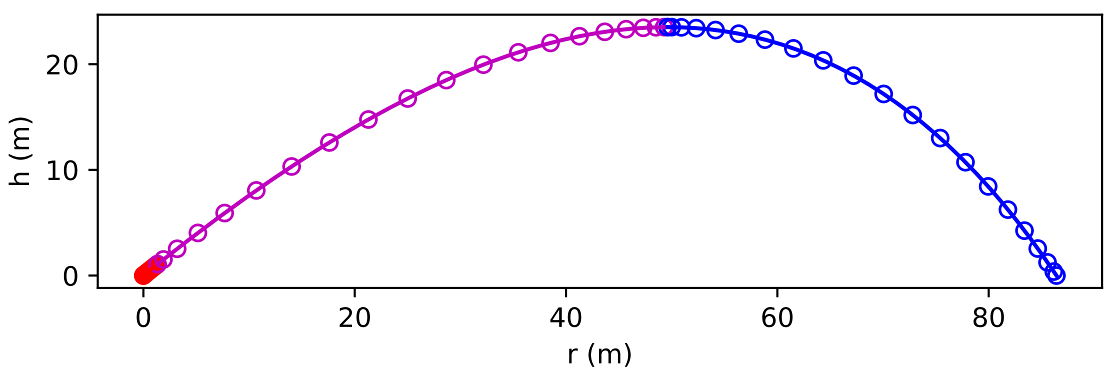


Figure 7: Thrust and selected states during propelled ascent phase optimized for height.



(a) flight dynamics states



(b) trajectory

Figure 8: Trajectory for maximum range rocket

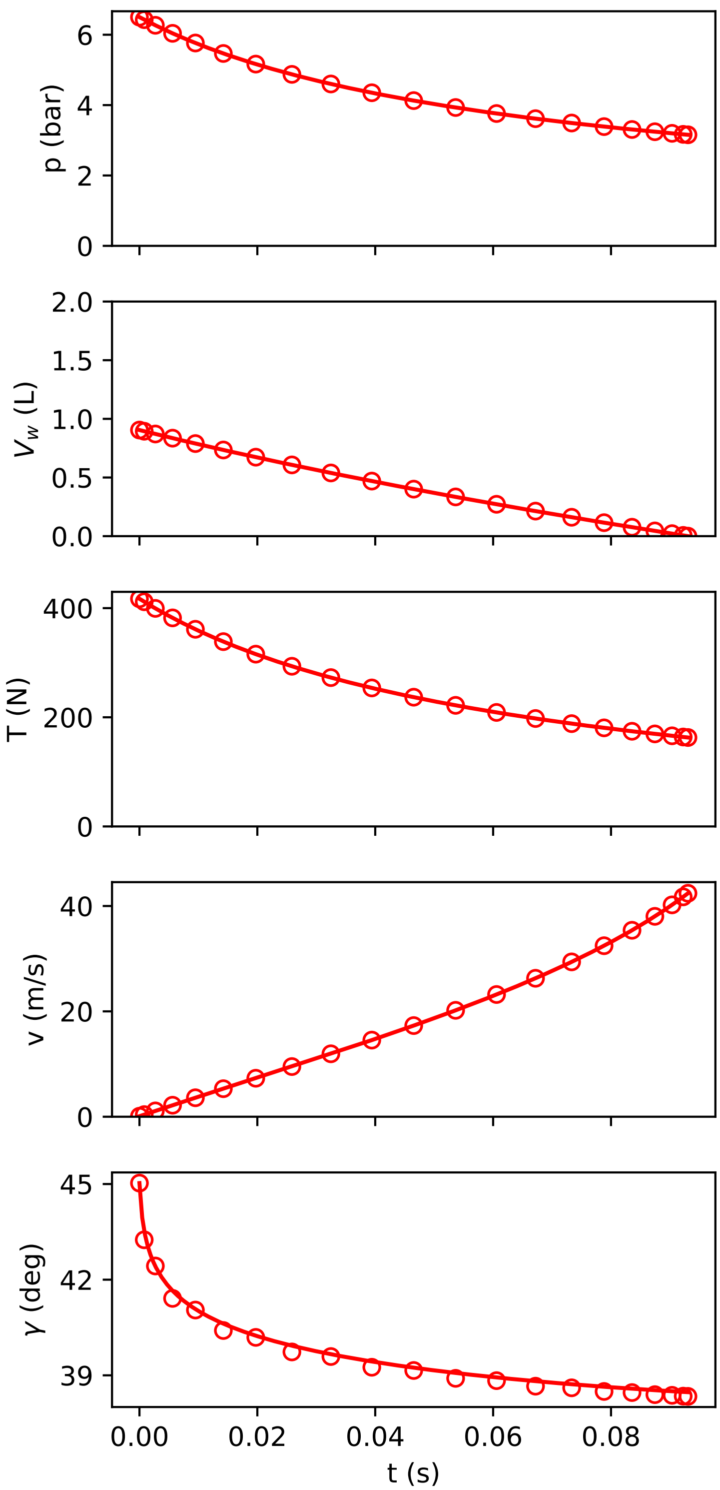


Figure 9: Thrust and selected states during propelled ascent phase optimized for range.

It was found that the problem is truly multidisciplinary in nature. It was also shown how the choice of optimization function affects the optimum and the convergence qualities of the problem. Furthermore, the usefulness of high order transcriptions for dynamic models was highlighted.

In spite of having a simple modelling, the water rocket can generate a fairly large optimization problem if collocations methods are used. In order to achieve acceptable accuracy, the problem must have hundreds of design variables and functions. Therefore, a gradient based optimization method with high quality derivatives is needed. This work highlights how an organized framework can ease the development effort of a model with such requirements.

Finally, we expect that this accessible example of the use of OpenMDAO and Dymos will help to increase the reach of these technologies, an we expect this example to be incorporated into Dymos' source code in the near future.

References

[1] David Kagan, Louis Buchholtz, and Lynda Klein. Soda-bottle water rockets. *The Physics Teacher*, 33(3):150–157, mar 1995. doi:10.1119/1.2344175.

[2] Justin S. Gray, John T. Hwang, Joaquim R. R. A. Martins, Kenneth T. Moore, and Bret A. Naylor. OpenMDAO: An open-source framework for multidisciplinary design, analysis, and optimization. *Structural and Multidisciplinary Optimization*, 59 (4):1075–1104, April 2019. doi:10.1007/s00158-019-02211-z.

[3] Robert D. Falck and Justin S. Gray. Optimal control within the context of multidisciplinary design, analysis, and optimization. In *AIAA Scitech 2019 Forum*. American Institute of Aeronautics and Astronautics, jan 2019. doi:10.2514/6.2019-0976.

[4] John T. Hwang and Drayton Munster. Solution of ordinary differential equations in gradient-based multidisciplinary design optimization. In *2018 AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*. American Institute of Aeronautics and Astronautics, jan 2018. doi:10.2514/6.2018-1646.

[5] Joseph M. Prusa. Hydrodynamics of a water rocket. *SIAM Review*, 42(4):719–726, jan 2000. doi:10.1137/s0036144598348223.

[6] Dean Wheeler. Modeling the thrust phase, 2002. URL `https://www.et.byu.edu/~wheeler/benchtop/pix/thrust_eqns.pdf`.

[7] Cedric J. Gommes. A more thorough analysis of water rockets: Moist adiabats, transient flows, and inertial forces in a soda bottle. *American Journal of Physics*, 78 (3):236–243, mar 2010. doi:10.1119/1.3257702.

[8] R Barrio-Perotti, E Blanco-Marigorta, J Fernández-Francos, and M Galdo-Vega. Theoretical and experimental analysis of the physics of water rockets. *European Journal of Physics*, 31(5):1131–1147, jul 2010. doi:10.1088/0143-0807/31/5/015.

[9] Glen E. Thorncroft, John R. Ridgely, and Christopher C. Pascual. Hydrodynamics and thrust characteristics of a water-propelled rocket. *International Journal of Mechanical Engineering Education*, 37(3):241–261, jul 2009. doi:10.7227/ijmee.37.3.9.

[10] R Barrio-Perotti, E Blanco-Marigorta, K Argüelles-Díaz, and J Fernández-Oro. Experimental evaluation of the drag coefficient of water rockets by a simple free-fall test. *European Journal of Physics*, 30(5):1039–1048, jul 2009. doi:10.1088/0143-0807/30/5/012.

[11] Laura Fischer, Thomas Günther, Linda Herzig, Tobias Jarzina, Frank Klinker, Sabine Knipper, Franz-Georg Schürmann, and Marvin Wollek. On the approximation of D.I.Y. water rocket dynamics including air drag. arXiv:2001.08828, 2020.

[12] Alejandro Romanelli, Italo Bove, and Federico González Madina. Air expansion in a water rocket. *American Journal of Physics*, 81(10):762–766, oct 2013. doi:10.1119/1.4811116.

[13] Nozzles. `http://www.aircommandrockets.com/nozzles.htm`, 2013.