[ME EN 2450 Memo]

To:	ME EN 2450
From:	Kin Blandford, Hyrum Coleman, Austin Neff
Date:	Nov 24, 2022
Subject:	Compressed Air Train Project
Attachments:	main.py, rk4.py, optimize_method.py, train_motion.py

---

Summary

In order to find the best set of parameters for a single stroke, compressed air powered train—such that the train reaches a displacement of 10 meters in a minimum time and coasts to a stop before reaching a displacement of 12.5 meters—we implement a modified Monte Carlo optimization method.

The best set of parameters we discovered can be found in table 3. With these parameters, the train reaches the 10 meter mark in 5.63 seconds and coasts to a stop before reaching 12.5 meters. No parameter constrains were violated.

Numerical Methods

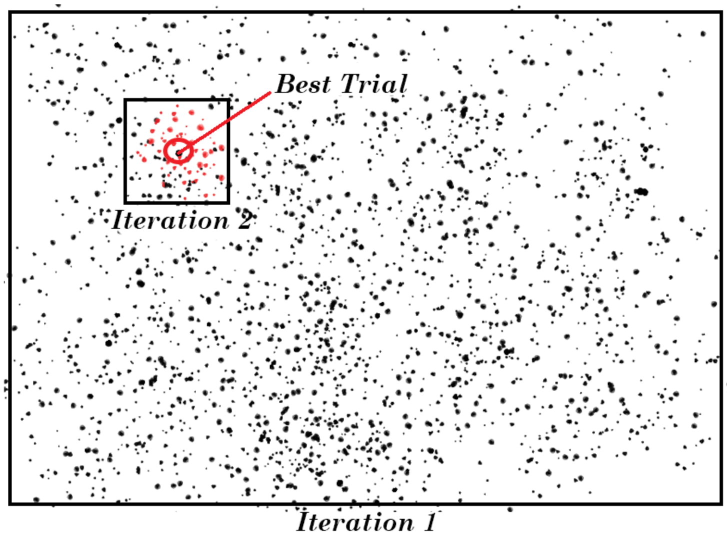
In order to simulate the motion of the train, we used the RK-4 numerical method and the following system of ODE's and a step size of 0.01s. The parameters in these ODE's can be found in Table 1. The train begins at rest, so the initial conditions are velocity=0 and acceleration=0.

$$\text{Acceleration:} \quad \frac{d^2x}{dt^2} = \frac{1}{m+m_w}\left[A_p\frac{r_g}{r_w}\left(\frac{P_0V_0}{V_0+A_p(r_g/r_w)x} - P_{\text{atm}}\right) - \frac{1}{2}C_d\rho A\left(\frac{dx}{dt}\right)^2 - C_r mg\right]$$

$$\text{Deceleration:} \quad \frac{d^2x}{dt^2} = \frac{1}{m}\left[-\frac{1}{2}C_d\rho A\left(\frac{dx}{dt}\right)^2 - C_r mg\right].$$

Equations of motion, where m is the total mass of the train, mw is the mass of the wheels, Ap is the area of the piston head, rg is the radius of the gear, rw is the radius of the wheel, P0 is the initial tank pressure, V0 is the initial tank volume, Patm is the atmospheric pressure, Cd is the coefficient of air resistance, A is the total frontal area of the train, Cr is the coefficient of rolling resistance, and g is the acceleration due to gravity.

To obtain the optimal set of parameters, we first defined a 'cost' function. The cost function checks if any design or situational constraints are violated and returns the time to complete the race. If any constraints are violated, the cost function returns an arbitrarily large number. The cost function is the function to be minimized, since the time to complete the race is the result to be minimized.



Then, in order to minimize the cost function, we implemented a multi-layer Monte Carlo optimization method. This method iteratively evaluates the cost function with a set of randomized parameters. The parameters are initially randomized within their allowable ranges. After the first iteration of optimization, the parameters are randomized in an increasingly small range around the best set of parameters found from each iteration (Supplementary figure 1).

Supplementary Figure 1.
A visual description of the iterative optimization method. The 2D space represents the design space. The size of the dots indicates the success of a trial. After each iteration is complete, the best trial becomes the center for the next region of randomized searching.

Eventually, after at least three iterations, the best set of parameters found, as well as the time to complete the race, is returned.

Table 1 lists all of the relevant design parameters in the problem. The ranges of values provided in the project description were used, in addition to the constant values found in the bottom half of the table.

Satisfying these ranges of values is not sufficient for this project, however. The train cannot slip at any time throughout the duration of its motion, as the system of ODE's does not account for the potential 'slipping mode' of motion. This is checked based on the acceleration of the train, and if the static force of friction can provide a sufficient amount of torque. The train must also be able to pass through the tunnel, so its height must not exceed 2.3 meters and its width must not exceed 2 meters. Also, the radius of the pinion gear must not be larger than the radius of the trains wheels. Finally, the trains overall length must not exceed 1.5 meters. These situational constraints are checked for in our cost function, which returns an arbitrarily large value if any of them are violated (as discussed earlier).

Table 1. Relevant physical parameters in the design problem

parameter	symbol	range of values	units
length of train	L_t	(0.2, 0.3)	m
outer diameter of train	D_o	(0.05, 0.2)	m
density of train material	ρ_t	choose material	kg/m^3
initial tank gage pressure	$P_{0_{\text{gage}}}$	(70000, 200000)	psig
pinion gear radius	r_g	(0.002, 0.01)	m
length of piston stroke	L_r	(0.1, 0.5)	m
diameter of piston	D_p	(0.04, 0.08)	m
air density	ρ_a	1	kg/m^3
atmospheric pressure	P_{atm}	14.6959	psi
drag coefficient	C_d	0.8	—
rolling friction coefficient	C_r	0.03	—
coefficient of static friction	μ_s	0.7	—
wheel diameter	D_w	40	mm
mass of wheels and axles	m_w	0.1	kg

Numerical Results

In the optimal trial, the train reached the finish line in 5.63 seconds, and slowed to a stop at 12.5 meters (right before the end of the track).

Figure 1 shows a plot of the distance traveled versus time for the optimal train design. The red dashed line shows the velocity of the train throughout the race. The blue solid line shows the position of the train throughout the race. The green dot and dashed lines show the finish time and position. The blue dashed line shows the end of the track.

Table 2. Summary of optimal train results

Computational time: 0.02 s (single run)
Optimization time: 1h+
Number of iterations: 200,000+
Maximum distance traveled: 12.5 m
Time to reach finish line: 5.63 s

Length of Train: 0.25 m
Outer Diameter of train: 0.2 m
Height of train: 0.23 m
Material of train: Titanium
Total Mass of Train: 28.9006 kg
Train frontal area: 0.0982 m^2
Initial Pressure: 200000 Pa
Initial tank volume: 0.0007 m^3
Pinion Gear Radius: 0.006 m
Length of stroke: 0.3 m
Total length of piston: 0.45 m
Diameter of piston: 0.06 m
Mass of piston: 1.5904 kg

Figure 1. Plot of distance traveled versus time for the optimal train design.

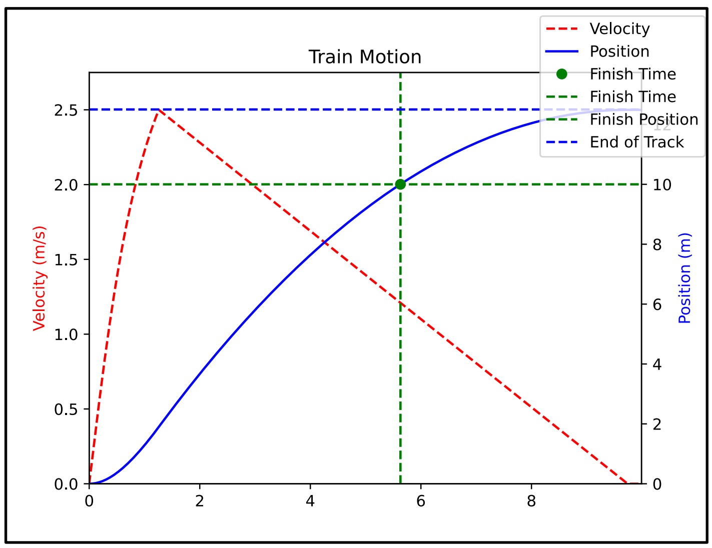


Table 3. Optimal set of physical parameters

parameter	symbol	optimum value	units
length of train	L_t	0.228	m
outer diameter of train	D_o	0.109	m
height of train	H_t	0.134	m
material of train	—	Copper	—
total mass of train	m	14.679	kg
train frontal area	A	0.018	m^2
initial tank pressure	P_0	79287.92	pa
tank volume	$\mathcal{V}_0$	0.00067	m^3
pinion gear radius	r_p	0.0053	m
length of piston stroke	L_r	0.4055	m
total length of piston	L_p	0.6082	m
diameter of piston	D_p	0.0611	m
mass of piston	M_p	2.233	kg

Realistic Train Design

The optimal parameters found have a large degree of precision which is realistically unobtainable. In addition, certain optimal parameters may not be physically attainable due to off-the-shelf availability.

Table 4 lists parts which are commercially available to build an optimal train. However, these parts do not exactly match the optimum parameters found.

The largest titanium pipe we could find had an outside diameter of 4” (0.1683m), which is less than the 4.29” (0.11m) optimal value.

The best rack and pinion available has a gear radius of 0.28” (0.007m), which is notably larger than the optimal 0.21” (0.005 m).

An ideal piston has a stroke length of 15.75” (0.4m), whereas the best piston we could find has a stroke length of 12” (0.3048m).

When simulating the motion of the train with these chosen components, the train is just barely unable to reach the finish line in 10 seconds. However, no other constraints are violated.

Table 4. Parts list for final train design

Component	Vendor	Price	Model Number	Spec
Piston	MSCDirect	$242.94	MSC# 85495935	12” Ls, 2” D
Tank / Train Body	Online Metals	$439.12	18370	4” OD
Rack and Pinion	McMaster	$15.73	7880K11	0.28” Rg
Wheels	Digi-Key	$3.95	2183-1087-ND	1” D
Axels	McMaster	$4.34	1327K84	1/16”
End caps	McMaster	$5.29	9275K68	For 4” OD

Attached Scripts
The following files are the python scripts used in the simulation and optimization of the train motion:

main.py, rk4.py, optimize_method.py, train_motion.py

The driver script, main.py, initializes global physical constants, parameter bounds, and describes all parameters in their unabbreviated form. The driver script can be run in two modes. The first mode performs optimization in the manner described in the 'Numerical Methods' section. The second mode runs the simulation of the trains motion with a specified set of parameters. In either case, the physical parameters are printed to the console and the trains motion is plotted.

The numerical ODE solver is contained in rk4.py. The method is general and can solve a system of any number of ODE's.

The optimization functions and the cost function are contained in optimize_method.py, in addition to helper functions which create parameter bounds and randomize parameter values in specified ranges. A result class is also contained in optimize_method.py, which is the standard format of trial results in our program. An exhaustive search optimization method is contained in optimize_method.py, but that was not used to find our optimal set of parameters due to unsatisfactory results.

The trains motion is computed in train_motion.py in the train_motion function. This is the function that is solved numerically using the RK-4 method. It is generalized to take parameters in the format of lists, dictionaries, or lists of type np.ndarray. It solves either the acceleration or deceleration ODE depending on the position on the track.