Results

Without losing generality we can consider rhoLeft greater than rhoRight otherwise the graphic will be reversed (the shock wave moving to the left, the rarefaction wave moving to the right, and the velocity with inversed sign)

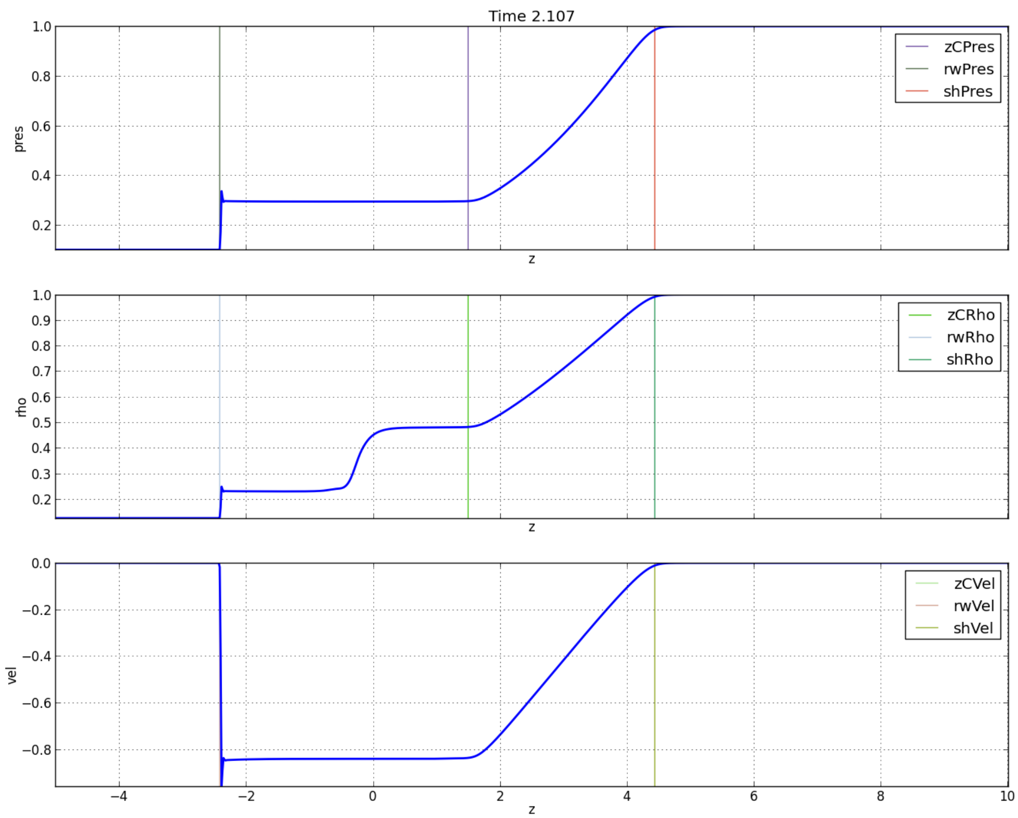

Figura 1: *shock tube reversed*

From now on we olly consider cases when shock wave is moving left and rw left(it's checked in initcond_riemann for rho1 >rho2)

The points in the task file : p1, p2, p3, p4, p5, p6 Obs:

1. pres1 = presRight

2. rho1 = rhoRight
3. vel1 = velRight
4. vel2 = vel3
5. pres2 = pres3
6. pres6 = presLeft
7. vel6 = velLeft
8. rho6 = rhoLeft
9. in the discontinuity part (p2-p3) density is not constant, dcPoint will mark where it changes(it's discontinuous) rho2 <rho1

1. rwPoint and shPoint are determned empirically(taking the points from left, right respective when the function pressure is not constant anymore) - analyze_functions.py
2. The rarefaction wave is moving with speed(csRW) velLeft - csLeft and rwPoint is also plotted like this (rwPointAn in the graphic: Point is Rho, Pres and Vel). The point is initialized with the value of rwPoint after the time = timeAfterAnPoints (defined in riemann_params.py)
3. When velLeft = velRight = 0(not in the case of complete problem) the shock wave is moving with speed(csShock) vel2 * rho2 / (rho2 - rhoRight) in the graphic marked as shPointAn initialized in the same way as rwPointAn(after timeAfterAnPoints)
4. dcPoint is moving with speed vel2 and it's initialized at the beginning = zC(it is analytically calculated. here analytically means that it was calculated and not empirically determined by observing the shaoes of the functions)
5. the expressions from tasks file are evaluated after timeAfterAnPoints so we should redirect output

   ```
   python main.py --timeEnd=10 > out
   ```

   The output for the paramers in tasks file is in files out_*

6. We can mark point p1, p2, p3, p4, p5, p6 - (set mark6Points to True in model_riemann.py) to see where actually the values are calculated - they are points right /and left of shPoint, zC, and rwPoint respectivelly (we can change them (moving them closer or farther) by changing delta in checkExpressions function in model_riemann.py

Shock tube

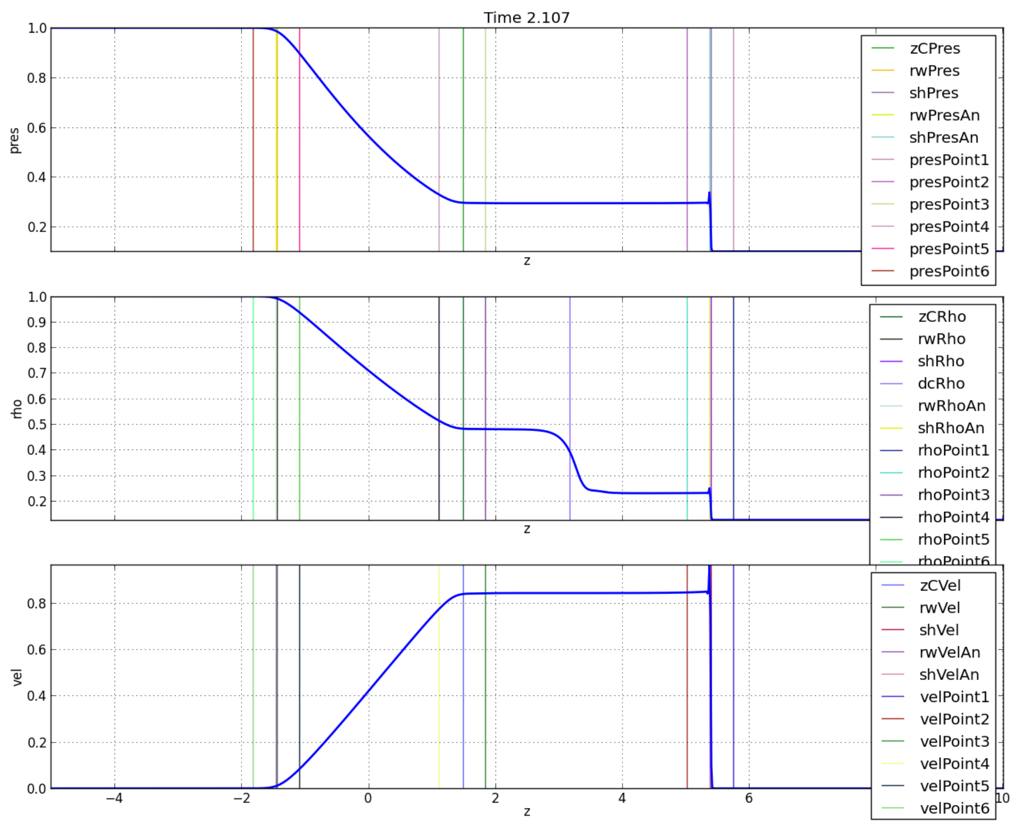

Figura 2: *shock tube*

Observations:

1. cs6 = -csRW
2. csShock >cs1 and csShock >cs2
3. relations fulfilled see out_shocktube file

Complete

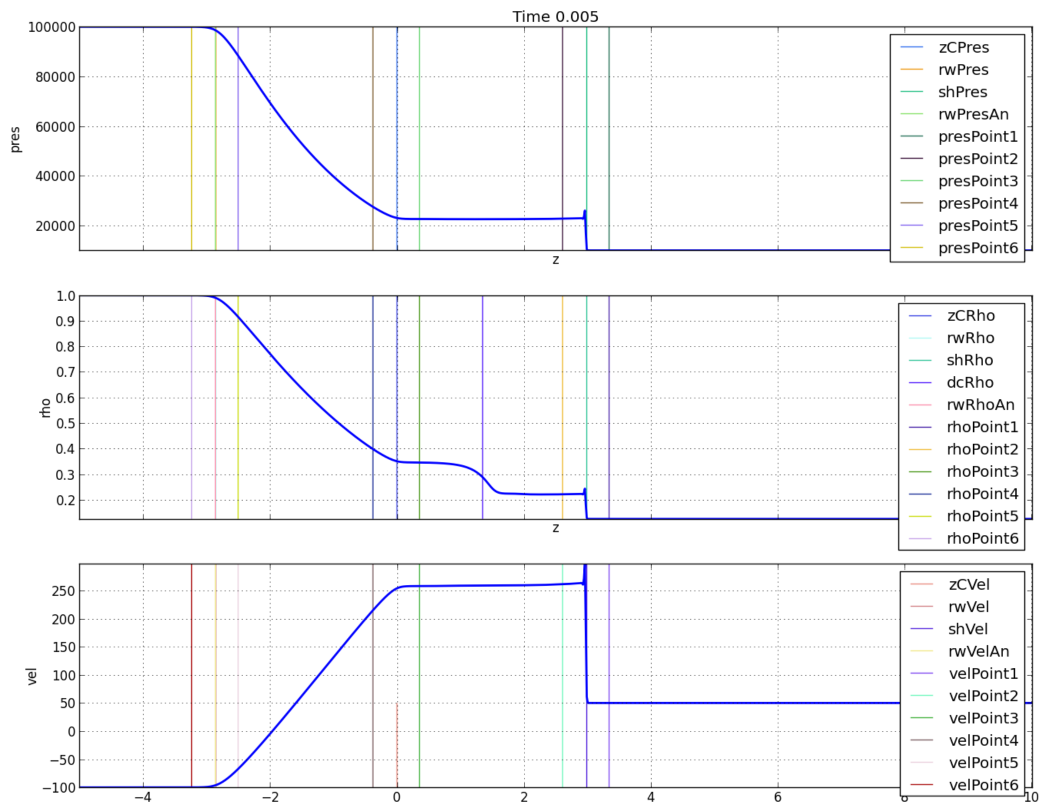

Figura 3: *complete*

Observations:

1. csRW = cs6 - velLeft (this should always be true)

2. I don't know how to determine analytically csShock in this case when velocities are not 0(but I think there is a way), that's why shPointAn is not shown in the graph in this case. I could calculate csShock from empirical determination of shPoint (csShock = (shPoint - shPointPrevious) / dt) , but this not done yet(to compare with cs1 and cs2). Calculating csShock as sqrt(gamma * pres / rho) in shPoint is not practical because of the discontinuity and oscillations

3. relations fulfilled see out_complete file

4. making velRight = -300 (greater than csShock) will make appear oscillations

5. But this was the case when velRight <0 and velLeft >0, when changing signs(fluids not approaching each other) this oscillation does not appear

Expansion into vacuum

Case a

Case b

Observations:

1. fluid from left is not filling instantly the right side

2. velocity grows slower in case b

3. density falls faster in case a - because of low density shPointAn is not very well calculated

TODO

1. move legend out of the plot

2. calculate shPoint velocity (csShock) in case of complete problem (I don't know the analyitical solution in this case)

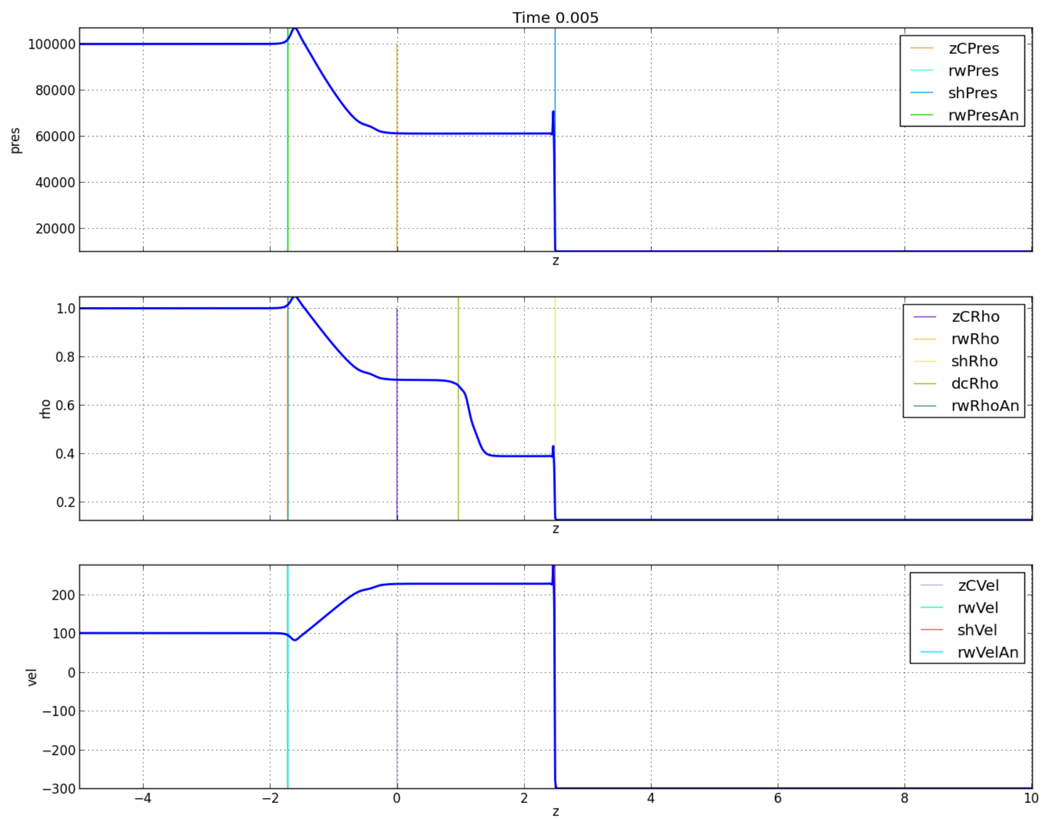

Figura 4: *complete abs(velRight) bigger*

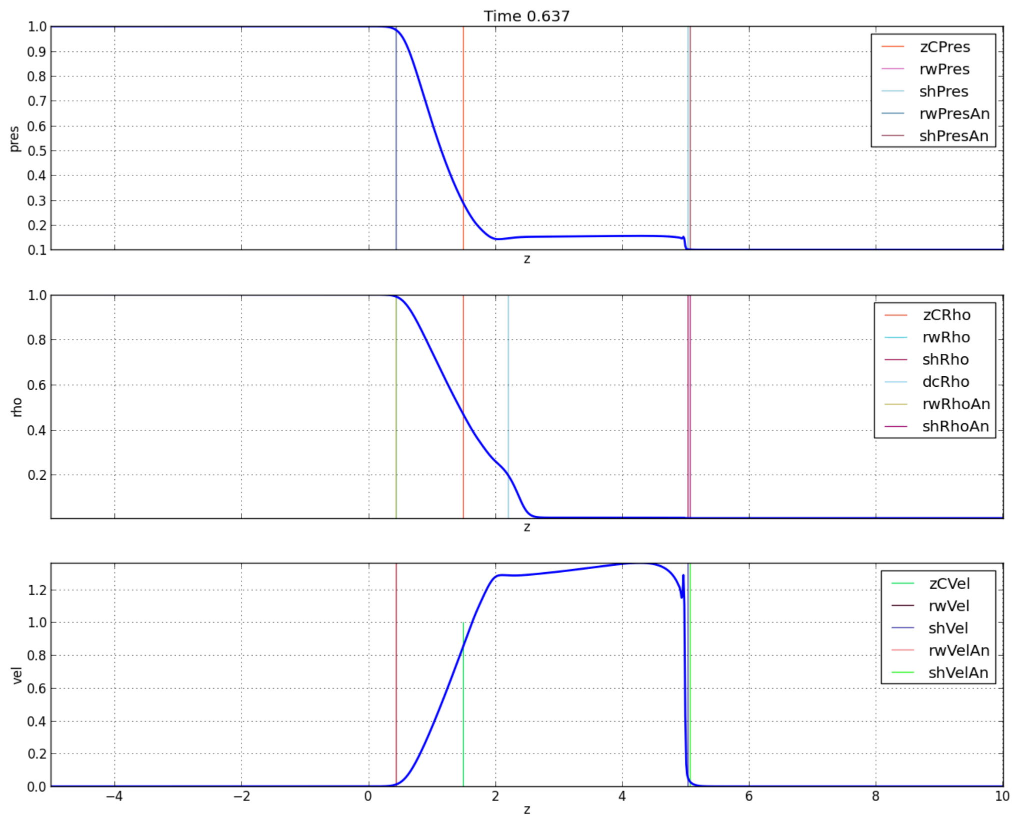

Figura 5: *exp vacuum case a*

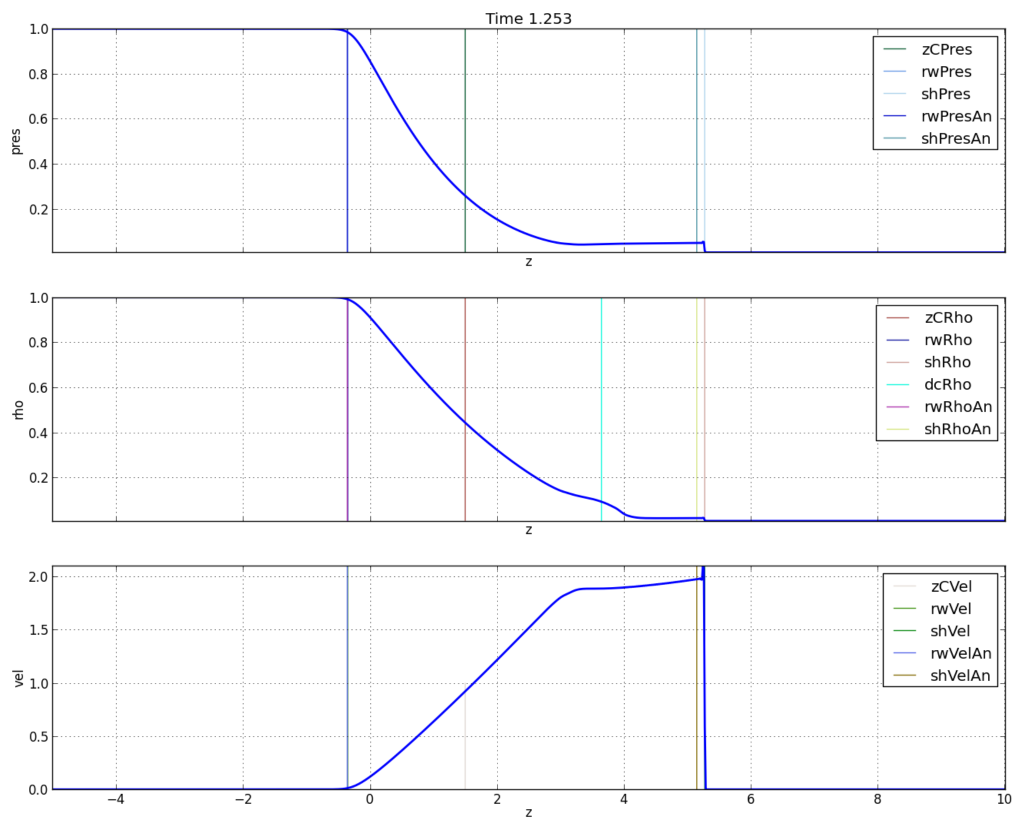

Figura 6: *exp vacuum case b*