We consider the classical algorithm in Clawpack with the minmod TVD limiter and solve the p-system:

$$\epsilon_t - u_x = 0, \tag{1a}$$
$$(\rho(x)u)_t - \sigma_x(\epsilon, x) = 0, \tag{1b}$$

where $\sigma(\epsilon, x) = \exp(K(x)\epsilon) - 1$,

$$\rho(x) = \frac{\rho_A + \rho_B}{2} + \frac{\rho_A - \rho_B}{2}\sin(2\pi x),$$
$$K(x) = \frac{K_A + K_B}{2} + \frac{KA - KB}{2}\sin(2\pi x).$$

The domain is given by $\Omega = [0, 400]$.

Generation of stegotons

To generate a stegoton we start with a zero initial condition and consider the following left boundary condition:

$$\epsilon(0,t) = 0,$$
$$u(0,t) = \begin{cases} -0.1\left[1 + \cos\left(t_0\pi\right)\right], & \text{if } |t_0| \leq 1, \\ 0, & \text{otherwise,} \end{cases}$$

where $t_0 = \frac{t-2.5}{2.5}$. Note that once $t > 5$, the left boundary is zero. To produce the stegotons in the paper, we proceed as follows:

- **Generation of the main stegoton.** Run create_stegotons/run_psystem.py, which solves (1) up to a final time $t = 400$. The resolution is given by $\Delta x = 1/1024$.

- **Isolation of the stegoton.** Inside cut_stegoton, create a folder called _output and copy create_stegoton/_output/*0400* to the newly created folder. Finally, run cut_stegoton/cut.py. The isolated stegoton will be placed in cut_stegoton/_output_cut_steg. The stegoton is isolated based on the stress. We locate the peak of the wave and move to the left and right until the solution is smaller than 10^{-12}.

- **Refinement of the isolated stegoton.** Inside refine_cut_stegoton, create a folder called _output_cut_steg and copy all the files inside cut_stegoton/_output_cut_steg to the newly created folder. Run refine_cut_stegoton/refine_steg.py. To refine the stegoton, we start with the isolated stegoton with $\Delta x = 1/1024$ and perform a high-order polynomial reconstruction. Using the reconstruction, we obtain the cell averages of the solution for different refinements with $\Delta x = 1/2048, 1/4096, 1/8192$.

After following these steps, the data (in h5 files) for the multiple refinements will be placed in refine_cut_stegoton/. We place the refined stegotons in the original domain $\Omega = [0, 400]$ and in a smaller domain $\Omega = [0, 20]$. We use the refined stegotons to obtain the initial condition for the pseudospectral simulations in the manuscript.

Measurement of the speed of stegotons

We need to estimate the speed of a highly refined stegoton. We do that using the refinement with $\Delta x = 1/8192$ and run the simulations in the modified version of Clawpack in https://github.com/manuel-quezada/pyclaw/tree/compute_L1_error_wrt_init_cond. We proceed as follows:

- Inside propagate_cut_stegoton, create a folder called _output_refn3 and copy refine_cut_stegoton/_output_small_domain/*refn3* into it.
- Change the name of the copied files from 'init_refn3...' to 'claw...'.
- Run propagate_cut_stegotons/prop_stegoton.py. Doing so will create a file called 'file.csv'.
- Run propagate_cut_stegotons/measure_speed.py to estimate the speed of the stegoton.