Analysis: Out of Gas

This was a hard problem to wrap your head around, as evidenced by the results. The final solution was, however, surprisingly simple (although the argumentation for it is not).

Let us first consider a single case, with a single acceleration value **a**, and an arbitrarily large **N**. Suppose some strategy S_1 takes you home in time **T**. Obviously $a*T^2 / 2 ≥ D$, as $a*T^2 / 2$ is the distance we travel if we start accelerating immediately and never brake.

We can propose an alternate strategy S_2: first stop and wait for time **T** - $sqrt$(2 **D**/ **A**), and then start accelerating full speed and never brake. This will also bring you home in time **T**.

We have to prove that if S_1 did not collide with the other car, neither will S_2. We prove this by checking that S_2 arrives at any point between the top of the hill and your home no earlier than S_1.

Assume S_1 was at some point **X** later than S_2. Notice that the speed of S_1 coming into **X** had to be larger than the speed of S_2 — otherwise even accelerating full speed will not let S_1 catch up with S_2. But it is impossible to achieve a faster speed at **X** than S_2: strategy S_2 accelerated all the way from the top of the hill to **X**.

So, now we only need to consider strategies such as S_1. We now just need to determine how long do we need to wait on the top of the hill.

One last thing to notice is that if the other car moves at constant speed between x_i and x_{i+1}, and our strategy does not put us in front of the other car at x_i or at x_{i+1}, then it does not intercept the other car in any intermediate point either. We know that because if we did intercept it in some intermediate point, it would mean that we were moving faster than the other car at the time of interception; and since we're accelerating, and the other car's speed is constant, we would end up in front of it at x_{i+1}. Therefore passing the other car between x_i and x_{i+1} leads to a contradiction, and it must be the case that we don't pass it.

With this reasoning complete, we now have two possible algorithms. One is a binary search: To check if a given waiting time **T** is long enough, we just need to check all the **N** points at which the other car can change its speed. To achieve a precision of 10^{-6}, we will need around 40 iterations in the binary search. This means we solve each test case in O(**N A**) time, with the constant 40 hidden in the big-O notation — fast enough.

If we are worried about the constant factor of 40, we can also iterate over all points x_i, and for each calculate the minimum waiting time needed not to intercept the other car at this point: `t`$_i$ `- sqrt(2`x_i`/a)` for all i such that x_i `< D` (you also have to include the moment where the other car reaches your house). This will make our solution run in O(**N A**) time without the pesky constant.