APPENDIX

Variable	Definition
v_0	Desired velocity
T	Safe time headway
a	Maximum acceleration
b	Comfortable deceleration
δ	Minimum distance
l	Vehicle length
Δv_α	Velocity difference with front vehicle

TABLE III: Variable descriptions for the Intelligent Driver Model.

A. *Jacobian of the IDM*

For ease of calculation, we represent the simulation state vector q at a certain time step t to be a $(1, 2N)$ vector instead of a $(N, 2)$ vector, where N is the number of vehicles. Thus, the simulation state will take on the form

$$q_t = \begin{bmatrix} x_1 & v_1 & \dots & x_N & v_N \end{bmatrix}$$

Then, we can expect the Jacobian relating one state to the next to be a vector of dimension $(2N, 2N)$. For a particular vehicle α indices from the front vehicle, the Jacobian of the IDM forward simulation is derived. Let

$$f(x_\alpha, v_\alpha) = \dot{x}_\alpha$$
$$g(x_\alpha, v_\alpha) = \dot{v}_\alpha$$

Then the Jacobian of the IDM with respect to state values position x and velocity v will take on the form:

$$J_{idm}(q_0, q_1) = \begin{pmatrix} \frac{\delta vehicle1_{t=1}}{\delta vehicle1_{t=0}} & \dots & \frac{\delta vehicle1_{t=1}}{\delta vehicleN_{t=0}} \\ \vdots & \ddots & \vdots \\ \frac{\delta vehicleN_{t=1}}{\delta vehicle1_{t=0}} & \dots & \frac{\delta vehicleN_{t=1}}{\delta vehicleN_{t=0}} \end{pmatrix}$$

Recall from Equation 1 that the state of a single agent or vehicle comprises both a position and a velocity component. For sake of readability, the derivative below is always taken with respect to the previous timestep. Then, the Jacobian above can then be expanded:

$$\begin{pmatrix} \frac{\partial f(x_1,v_1)}{\partial x_1} & \frac{\partial f(x_1,v_1)}{\partial v_1} & \dots & \dots & \frac{\partial f(x_1,v_1)}{\partial x_N} & \frac{\partial f(x_1,v_1)}{\partial v_N} \\ \frac{\partial g(x_1,v_1)}{\partial x_1} & \frac{\partial g(x_1,v_1)}{\partial v_1} & \dots & \dots & \frac{\partial g(x_1,v_1)}{\partial x_N} & \frac{\partial g(x_1,v_1)}{\partial v_N} \\ \vdots & \vdots & \ddots & \ddots & \vdots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \vdots & \vdots \\ \frac{\partial f(x_N,v_N)}{\partial x_1} & \frac{\partial f(x_N,v_N)}{\partial v_1} & \dots & \dots & \frac{\partial f(x_N,v_N)}{\partial x_N} & \frac{\partial f(x_N,v_N)}{\partial v_N} \\ \frac{\partial g(x_N,v_N)}{\partial x_1} & \frac{\partial g(x_N,v_N)}{\partial v_1} & \dots & \dots & \frac{\partial g(x_N,v_N)}{\partial x_N} & \frac{\partial g(x_N,v_N)}{\partial v_N} \end{pmatrix}$$

This resulting Jacobian ends up being a lower triangular matrix. This is because any entries above the main 2-by-2 diagonal represent the relation between a vehicle and the vehicles behind it. In car-following models, vehicle position and velocity are not affected by vehicles behind. Thus, the upper half of the Jacobian is zeroed out. Additionally, in the context of IDM, a vehicle is only influenced by the vehicle directly in front of it. Thus, any partial derivatives between a vehicle and any vehicle more than 1 position ahead is also zeroed out. An example of the Jacobian for a 3-vehicle simulation is shown below:

$$\begin{pmatrix} \frac{\partial f(x_1,v_1)}{\partial x_1} & \frac{\partial f(x_1,v_1)}{\partial v_1} & 0 & 0 & 0 & 0 \\ \frac{\partial g(x_1,v_1)}{\partial x_1} & \frac{\partial g(x_1,v_1)}{\partial v_1} & 0 & 0 & 0 & 0 \\ \frac{\partial f(x_2,v_2)}{\partial x_1} & \frac{\partial f(x_2,v_2)}{\partial v_1} & \frac{\partial f(x_2,v_2)}{\partial x_2} & \frac{\partial f(x_2,v_2)}{\partial v_2} & 0 & 0 \\ \frac{\partial g(x_2,v_2)}{\partial x_1} & \frac{\partial g(x_2,v_2)}{\partial v_1} & \frac{\partial g(x_2,v_2)}{\partial x_2} & \frac{\partial g(x_2,v_2)}{\partial v_2} & 0 & 0 \\ 0 & 0 & \frac{\partial f(x_3,v_3)}{\partial x_2} & \frac{\partial f(x_3,v_3)}{\partial v_2} & \frac{\partial f(x_3,v_3)}{\partial x_3} & \frac{\partial f(x_3,v_3)}{\partial v_3} \\ 0 & 0 & \frac{\partial g(x_3,v_3)}{\partial x_2} & \frac{\partial g(x_3,v_3)}{\partial v_2} & \frac{\partial g(x_3,v_3)}{\partial x_3} & \frac{\partial g(x_3,v_3)}{\partial v_3} \end{pmatrix}$$

From this example, we can see that the Jacobian can be found by generalizing the partial derivative on the 2-by-2 diagonal, as well as the partial derivatives one 2-by-2 row directly below the diagonal. There are thus eight generalized terms to build the Jacobian matrix of one simulation time step. One entry on the 2-by-2 diagonal of the Jacobian matrix can be intuitively defined as the Jacobian of a vehicle's state with respect to itself from the previous time step. Thus, every 2-by-2 entry on the diagonal takes this general form for a particular vehicle in index α from the front:

$$J_{idm}[2\alpha, 2\alpha] = \begin{pmatrix} \frac{\partial f}{\partial x_\alpha} & \frac{\partial f}{\partial v_\alpha} \\ \frac{\partial g}{\partial x_\alpha} & \frac{\partial g}{\partial v_\alpha} \end{pmatrix}$$
$$\frac{\partial f}{\partial x_\alpha} = 0$$
$$\frac{\partial f}{\partial v_\alpha} = 1$$
$$\frac{\partial g}{\partial x_\alpha} = \frac{-2as^*(v_\alpha, \Delta v_\alpha)^2}{s_\alpha^3}$$
$$\frac{\partial g}{\partial v_\alpha} = \frac{-a\delta v_\alpha^{\delta-1}}{v_0^\delta} + \frac{-2a}{s_\alpha^2}\left(T + \frac{\Delta v_\alpha + v_\alpha}{2\sqrt{ab}}\right) s^*(v_\alpha, \Delta v_\alpha)$$

And, likewise, the diagonal one 2-by-2 row directly beneath it will take the form (excluding the first row, which represents the leading vehicle):

$$J_{idm}[2\alpha, 2\alpha - 2] = \begin{pmatrix} \frac{\partial f}{\partial x_{\alpha-1}} & \frac{\partial f}{\partial v_{\alpha-1}} \\ \frac{\partial g}{\partial x_{\alpha-1}} & \frac{\partial g}{\partial v_{\alpha-1}} \end{pmatrix}$$
$$\frac{\partial f}{\partial x_{\alpha-1}} = 0$$
$$\frac{\partial f}{\partial v_{\alpha-1}} = 0$$
$$\frac{\partial g}{\partial x_{\alpha-1}} = \frac{2as^*(v_\alpha, \Delta v_\alpha)^2}{s_\alpha^3}$$
$$\frac{\partial g}{\partial v_{\alpha-1}} = \frac{2as^*(v_\alpha, \Delta v_\alpha)v_\alpha}{2s_\alpha^2\sqrt{ab}}$$

With this analytically derived form of the Jacobian, we can now compute the Jacobian at any time step directly without the use of Autograd, given the current state as input. For N vehicles, we calculate $4N$ values on the main diagonal, $4(N-1)$ values on the ”subdiagonal”, and zero out every other value to attain the $2N \times 2N$ Jacobian matrix for a particular time step.

Note that sometimes, IDM will produce negative velocities in intermediate or end states. To address this issue, any submatrices where IDM yields a negative value at the next time step will have a gradient of zeroes. This addresses negative velocity clipping in the forward simulation, where negative velocities after the simulation update are clipped to zero.

The boost in speed per iteration using the analytical gradient versus autodifferentiation is visualized in Figure 5.

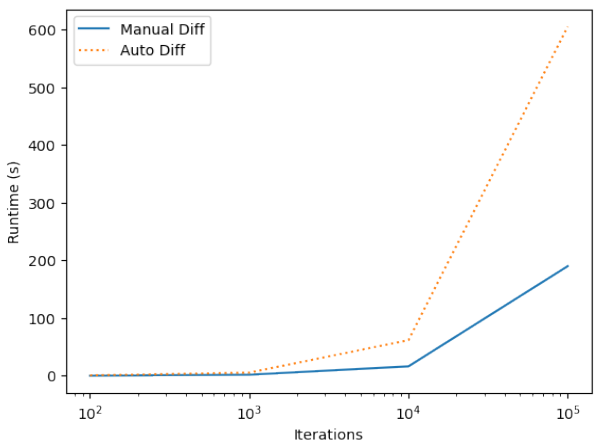

Fig. 5: **Comparison on runtime (s) over number of iterations for analytical differentiation versus auto differentiation of the car-following model IDM.**