Training process:

1. Preparation of the dataset:

For the training of our **LSTM** model, first the input features were decided. The input features chosen were normal reaction forces on the wheels and the centre of the rover, distance from an obstacle, velocity of the rover and height of the obstacle, which is negative if it's a ditch. The height/depth of an obstacle is split into two features, i.e, the height of the obstacle on the left of the rover's centre and on the right. This is done so that the model learns to move the left and right side of the suspension independently. Also, the height/depth for an obstacle is floored to 0 until the rover is 10 cm away from the obstacle since, because prior to this the suspension doesn't have to undergo any change and hence the model learns to do nothing until actually required. The output for the model are the control link angles.

5000 data points were extracted for the aforementioned features by running a dynamic simulation of the rover on Martian-like terrain on **SolidWorks**.

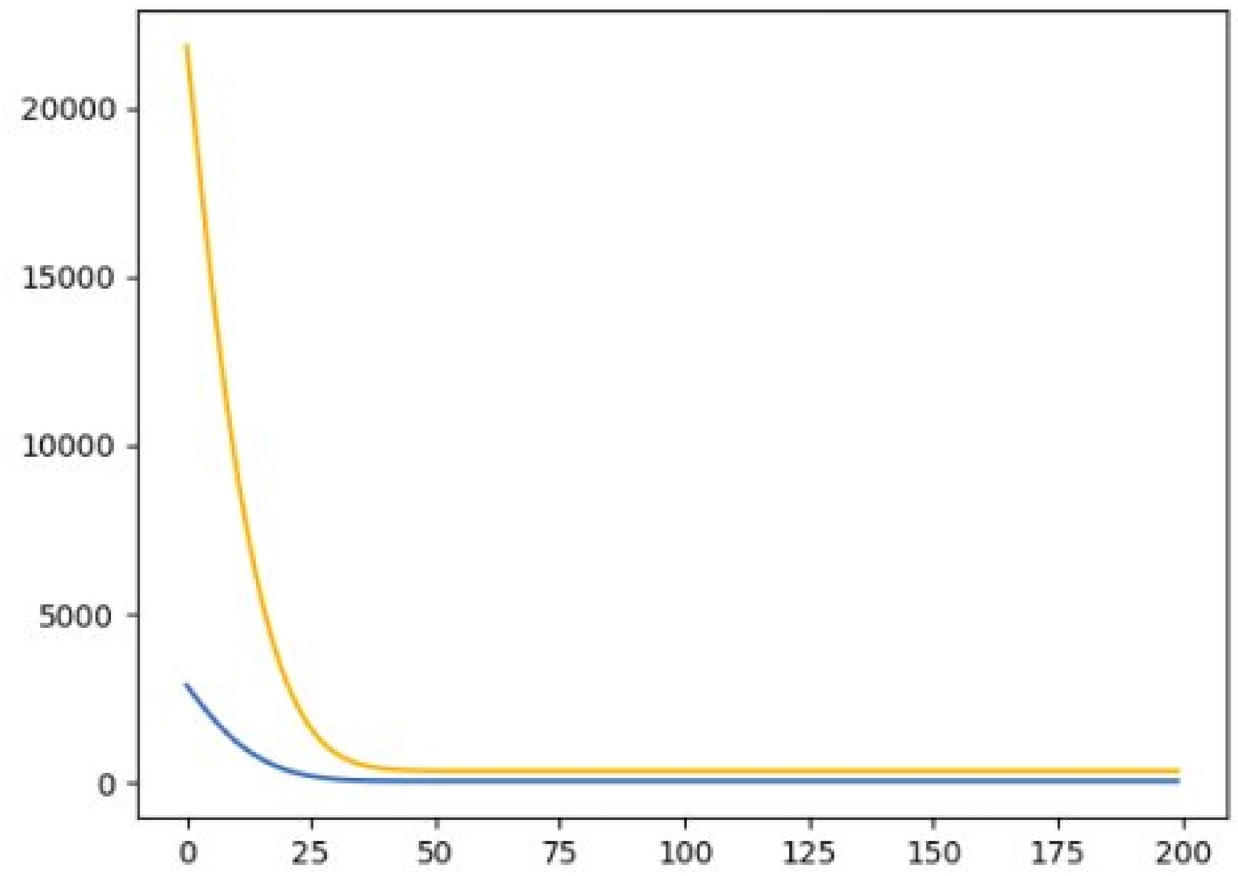

Fig 8 - Plot of MSE vs No. of Epochs for Train and Validation Sets

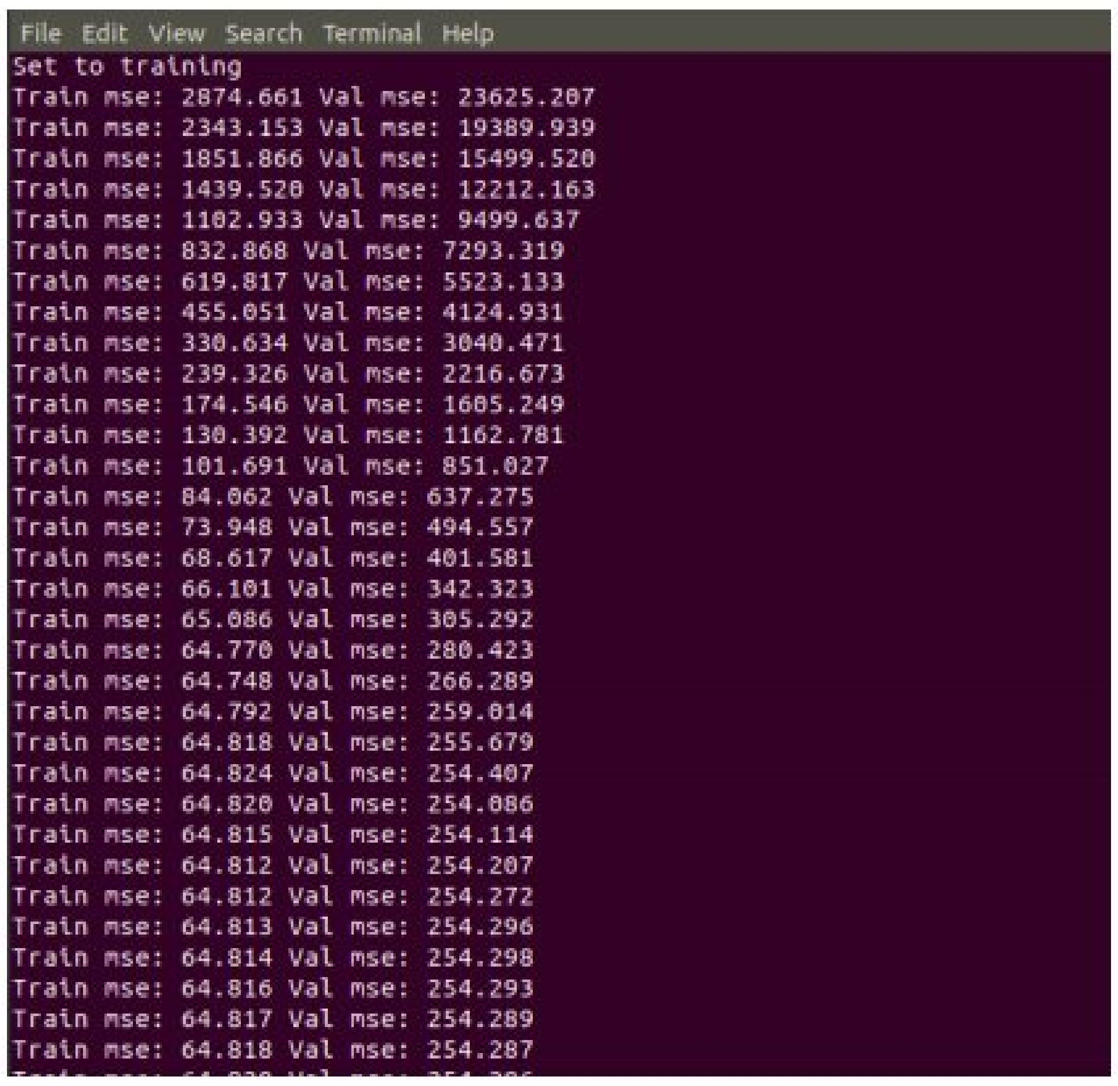

Fig 9 - Training of the model

2. Training and tuning:

Prior to training, a sliding window was used on the dataset to generate sequences of 42 consecutive time steps. Then the newly generated grouped dataset is split into train, validation and test set using a ratio split of 70-10-20.

Since the goal of the model is used to predict the control link angles at the next time-steps given the inputs of the 42 previous time-steps, the model has been trained accordingly. Hence, the loss function is the mean-squared error computed from the output produced from the last time step in a sequence and the label for the next time step.

The hyperparameters such as learning rate, number of epochs, number of hidden units and hidden layers were tuned to minimize the mse of the validation set.

3. Results:

The **LSTM** model produced decent results which can be deployed on a rover and easily improved as more training data is collected during actual traversal on Mars. The metric chosen to represent the results is root-mean-square error as we are aiming to regress the control link angles. The model achieved an **RMSE** of 13°. The error while still a bit significant due to the low training data used can be overcome

Scanned with CamScanner

by the structure of the suspension and due to the presence of a software threshold of 30°, above which the motor will actuate the control links of the suspension to overcome any obstacle.

Deployment:

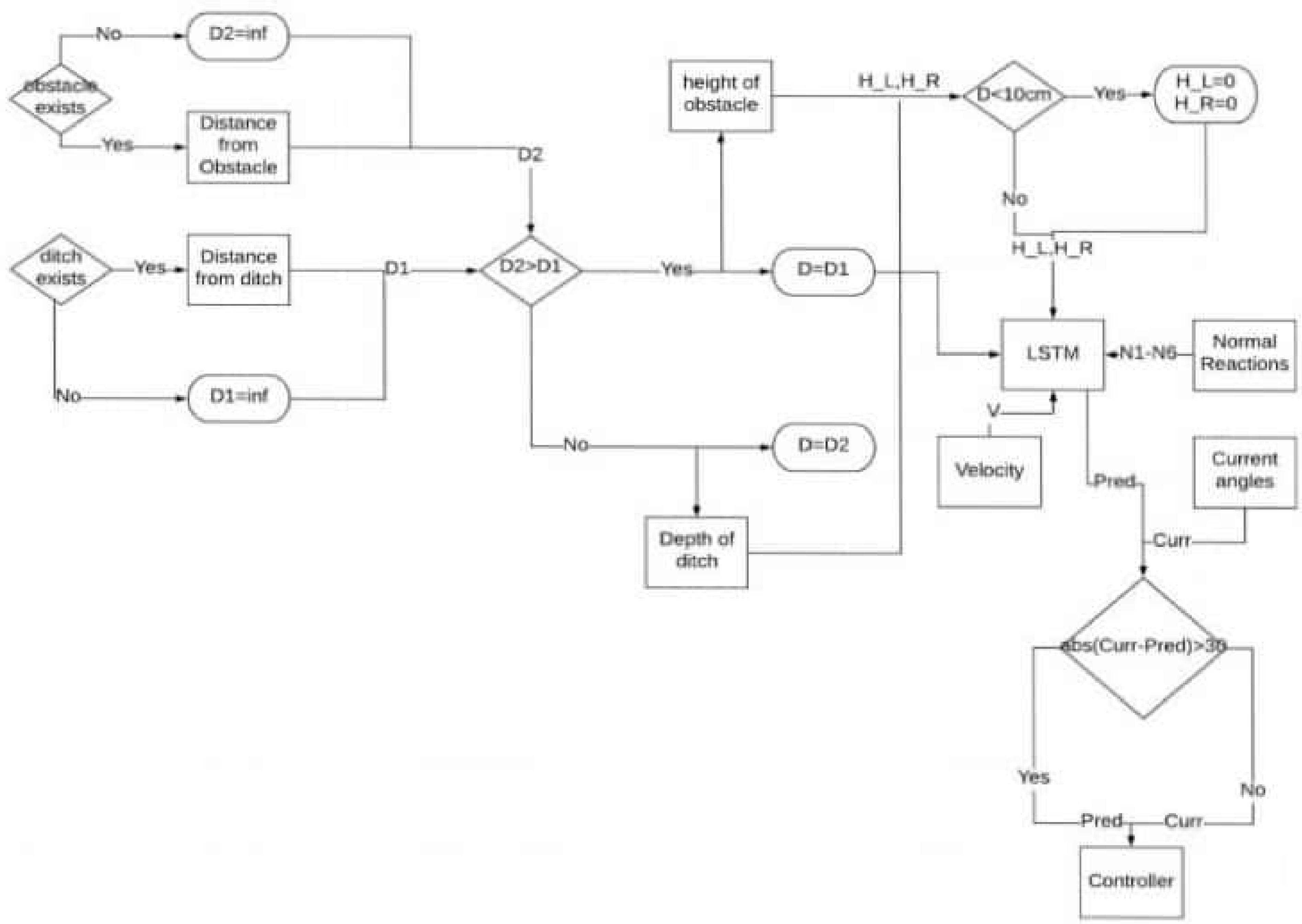

Fig 10 - Deployment flowchart for LSTM

As seen in Fig 11, the model only takes obstacles/ditches that are closer than 10 cm into consideration, to only react when necessary. Obstacle/ditch information such as distance from it or height/depth is given from the processed feed of the stereo cam. The normal reactions are given by force sensors appropriately placed on the wheels. The controller only makes a change to the suspension if the change is greater than a fixed threshold.

Scanned with CamScanner