**Resultados para robot de 2 DOF**

El presente informe muestra los resultados obtenidos para datos recolectados de robots de dos DOF. Se tienen dos casos principales, los cuales varían por el control de motores que realizan. Se utilizaron diferentes arquitecturas de redes neuronales para cada caso, especificándose los parámetros posteriormente.

Control PD + Gravity Compensation:

NN: Input (6) + Hidden (20) + Output(2). f\_act = tanh

Batch =  256, epochs\_max = 50, patience  = 1

Mean Absolute Error (Test data):

array([0.01144372, 0.01083645])

Last epoch:

Epoch 50/50

241601/241601 [==============================] - 1s 5us/step - loss: 3.3709e-04 - acc: 0.9977 - val\_loss: 2.1016e-04 - val\_acc: 0.9978

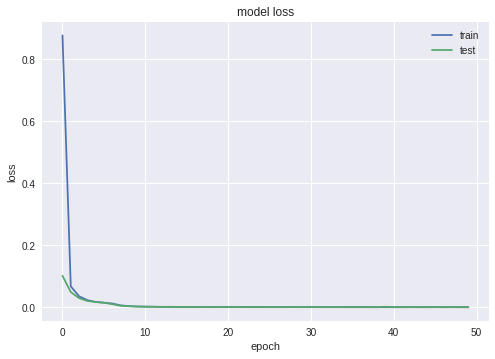


Fig. 1: Loss-train, loss-test

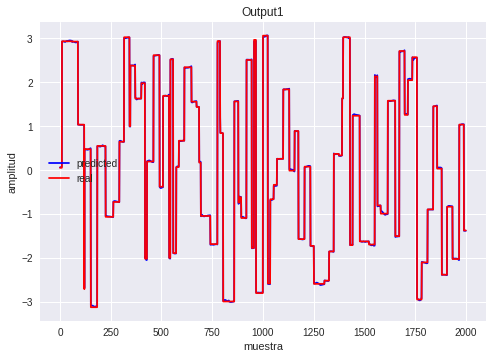


Fig. 2: Predicted (red) y Real (blue) para test data output 1

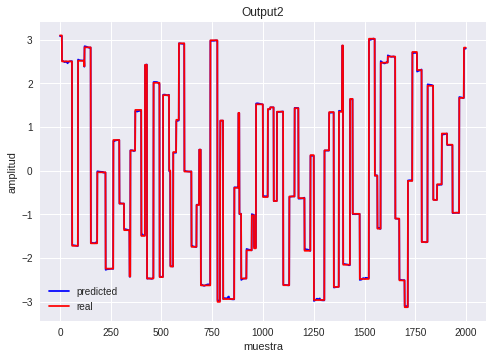


Fig. 3: Predicted (red) y Real (blue) para test data output 2

Feedback Linearization Controller:

1. NN: Input (6) + Hidden (32) + Output(2). f\_act = tanh

Batch =  256, epochs\_max = 1000, patience  = 1

Mean Absolute Error (Test data):

array([0.01593701, 0.01158849])

Last epoch:

Epoch 1000/1000

46855/46855 [==============================] - 0s 5us/step - loss: 2.2047e-04 - acc: 0.9978 - val\_loss: 3.8703e-04 - val\_acc: 0.9972

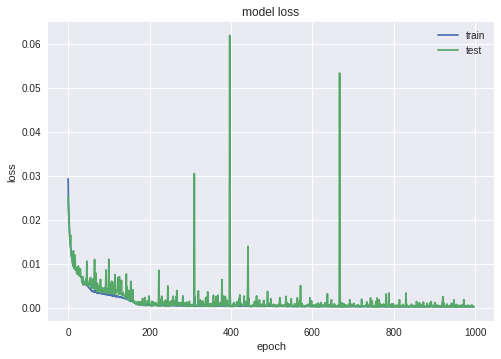


Fig. 4: Loss-train, loss-test



Fig. 5: Predicted (red) y Real (blue) para test data output 1

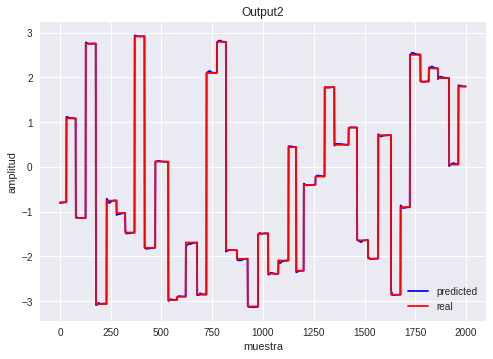


Fig. 6: Predicted (red) y Real (blue) para test data output 2

1. NN: Input (6) + Hidden (24) + Output(2). f\_act = tanh

Batch =  256, epochs\_max = 1000, patience  = 1

Mean Absolute Error (Test data):

array([0.01643259, 0.0180078 ])

Last epoch:

Epoch 1000/1000

46855/46855 [==============================] - 0s 5us/step - loss: 3.0533e-04 - acc: 0.9979 - val\_loss: 5.4879e-04 - val\_acc: 0.9858

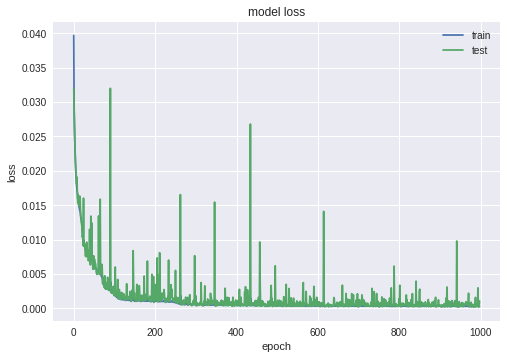


Fig. 7: Loss-train, loss-test

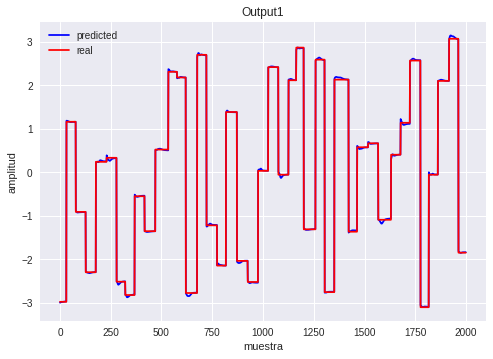


Fig. 8: Predicted (red) y Real (blue) para test data output 1

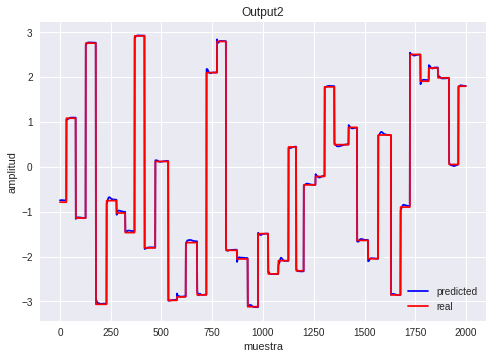


Fig. 9: Predicted (red) y Real (blue) para test data output 2

1. Polynomial Regression – Order 5.

With a polynomial combination on each variable.

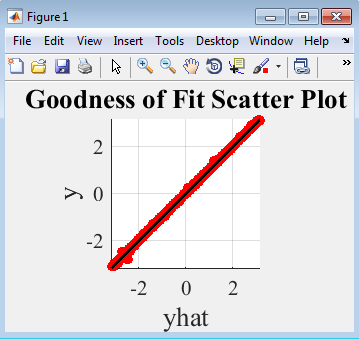


Fig. 10: Fit Scatter Plot

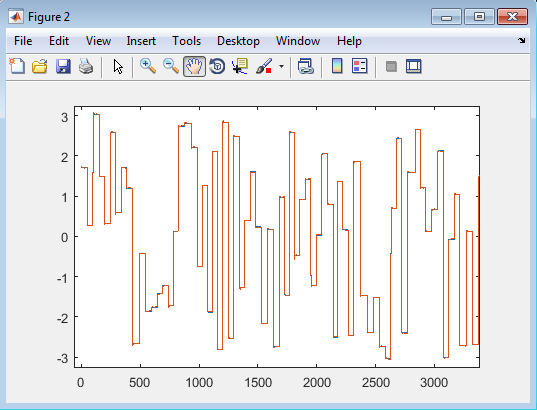
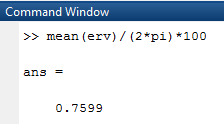


Fig. 11: Predicted (blue) y Real (red) para test data output 1

Data test was random generated with the next function:

NewDataPoint=[pi\*(2\*rand(1,2)-1) 4\*(2\*rand(1,2)-1) 40\*(2\*rand(1,2)-1)];

The mean of the absolute value of the errors in the possible output range () is:



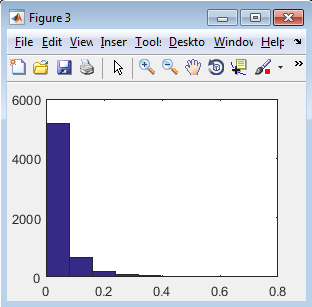


Fig. 12: Histograms of error in test.

A problem arises when the controller uses high gains for the error, specifically the coefficients of . In this cases, even when we have little errors on the estimation of , the control gains will highly increase this errors, so it’s hard to track low values of torque, but easy to track high values. Note that sometimes tracking fails because it is impossible to reach that values. This is a direct result of our limits on the values we can apply to the desired position

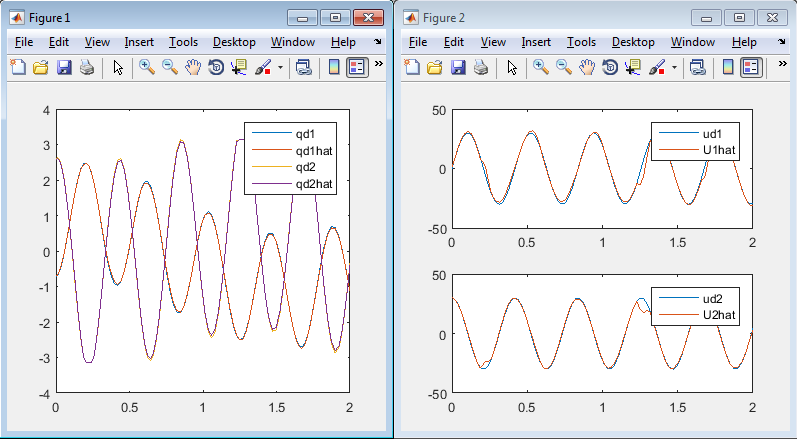


Fig. 13: Tracking high amplitude torques. Good tracks, with low relative error.

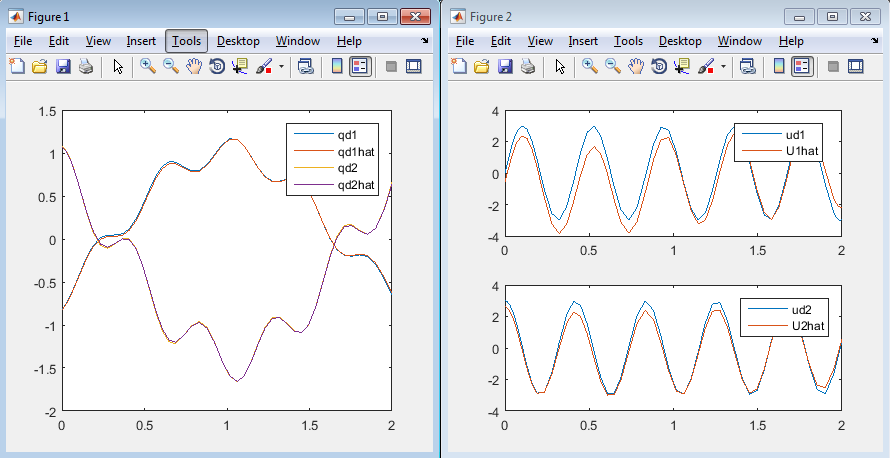


Fig. 14: Tracking low amplitude torques. Note that we are very close to the real values , but we have high relative errors.