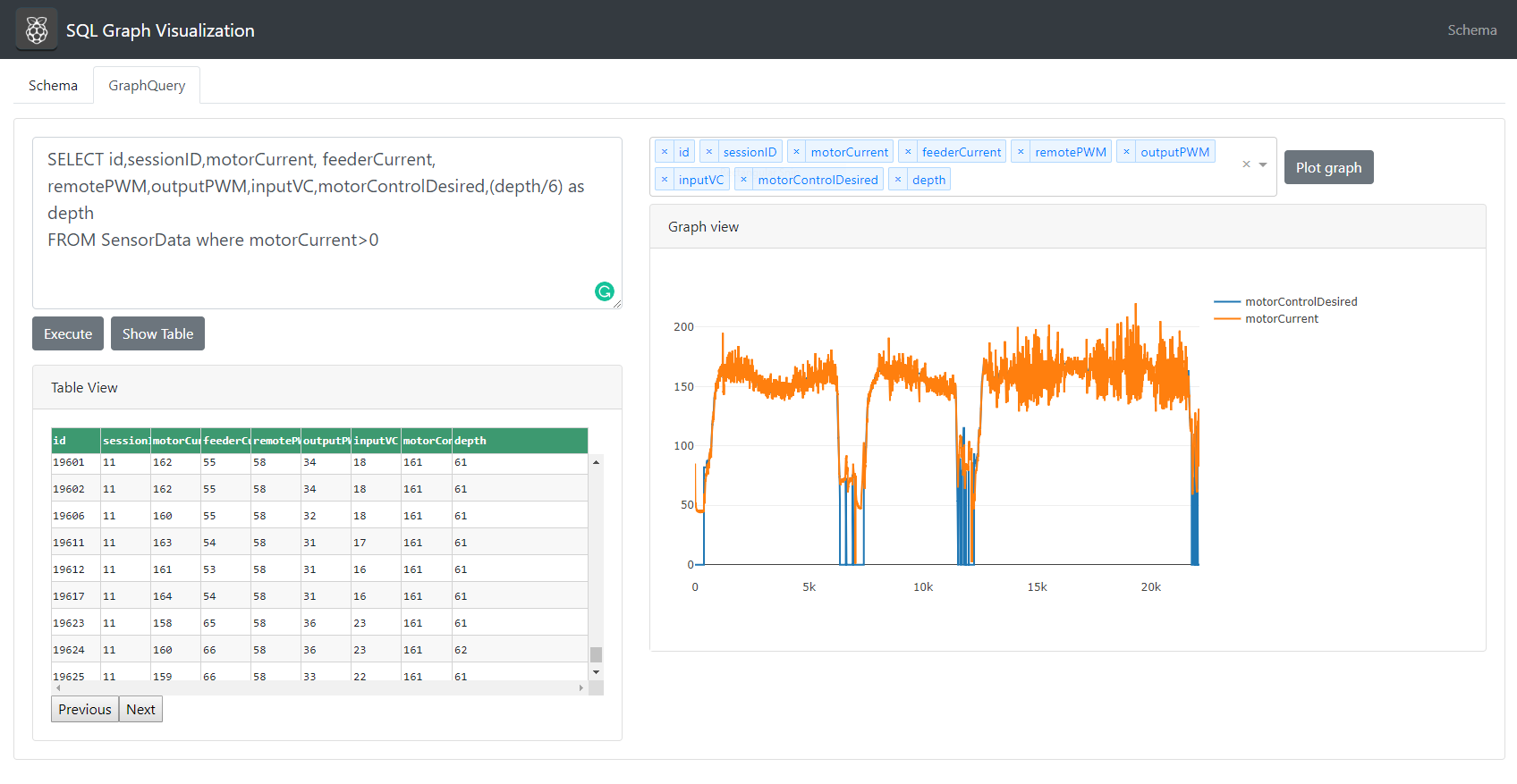
Intermediate report CIT

2019 Week 10

* Drilling session (2019-03-07): 3 drilling sessions with different positions of the accelerometer.
* SQL-database: made most of the progress on storing the data from XBEE module, accelerometer and time of flight (ToF) sensor.
* Half way through fixing the time of flight (ToF) sensor.
* SQL Graph visualization tool :
  + Features: can be launched from raspberry pi, can run query on the database, runs on PC browser.
  + Currently under operational condition but need to fix the performance for larger data sets (screenshot attached).
* In-progress for next week:
  + Improving the efficiency of SQL Graph visualization tool.
  + constructing the data pipeline for connecting the SQL data base and the machine learning algorithm.



# Intermediate report CIT

# 2019 Week 11

* Drilling session (2019-03-14): 6 drilling sessions with different positions of the accelerometer and different concrete hardness levels (2 bars and 1 bar).
* All the sensors (ToF, accelerometer) and Xbee module are perfectly working.
* SQL-database: recorded all 6 drilling sessions data in the SQL database.
* SQL Graph visualization tool:
  + Features: can be launched from raspberry pi, can run query on the database, runs on PC browser.
  + Performance issues w.r.t larger datasets is fixed with improved efficiency.
* In-progress for next week:
  + Constructing the data pipeline for connecting the SQL data base and the machine learning algorithm.
  + Testing different algorithm strategies for automatic control (PS: This might take a while)
  + Adding more functionality to the SQL visualization tool (histograms, multivariate plots, etc.)

Visualization of the 6 drilling sessions (recorded this week) plotted by SQL tool is presented below:



# Intermediate report CIT

# 2019 Week 12

* In-progress for this week and next week:
  + Database migration and synchronization (updating main database with new
  + Constructing online (from sensors) and offline (from database) drill environment for training and testing.
    - Constructing the data pipeline for connecting the SQL data base and the machine learning algorithm.
    - Cleaning the data offline and online with solutions like Kalman filter.
    - Generating clean state space and discretized action space
  + Testing different algorithm strategies for automatic control (PS: This might take a while)
  + Adding more functionality to the SQL visualization tool (histograms, multivariate plots, etc.)

# Intermediate report CIT

# 2019 Week 13

* In-progress for this week and next week:
  + Cleaning database to handle e.g. missing data entries
  + Converting potentiometer readings to discretized action taken
  + Started automatic control tests using 1D-CNN based network using inputs from Xbee and outputs from the generated action space
  + Strategies for improving the tests;
    - Better data sampling, e.g. to reduce skewness of data
    - Ignore bad data entries, such as data acquired form retracting the drill bit, which should not be considered during training

# Intermediate report CIT

# 2019 Week 14

* Strategies that improved the tests:
  + Better data sampling, e.g. to reduce skewness of data
  + Ignored bad data entries, such as data acquired form retracting the drill bit, which should not be considered during training
  + Script for automatic labeling of the drilling data based on the depth map of the block.
* Results from improved strategies:
  + A material detection algorithm with good accuracy based on the data from AD10.

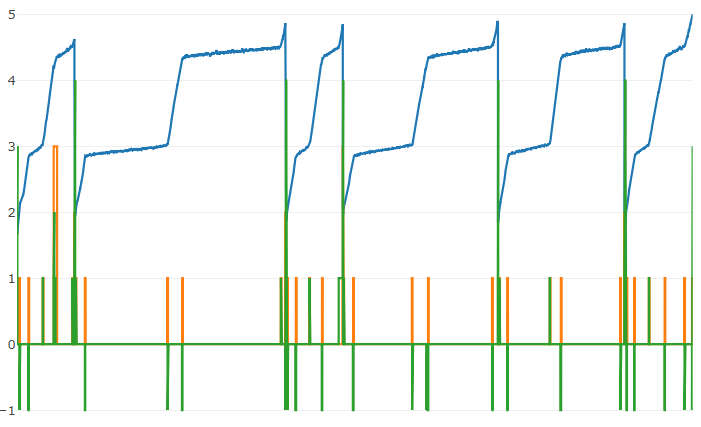


Figure 1 Result of the material classifier. The green line shows the correctness of the result, a zero means correct results. The green line is correct except where the orange is nonzero, indicating pars not usable for material classification.

* In-progress for this week and next week:
  + Checking the convergence of different supervised learning algorithms
  + Starting to build Reinforcement learning framework based the result from convergence test with supervised learning.
  + Build environment for drilling (reward function, terminal states, transitions etc.)
  + Build loss function based on the paper [DQfD](https://arxiv.org/pdf/1704.03732.pdf) to initially train on the expert demonstration data and then continue training on the drill setup.
  + Implement a prioritized sampler based on the importance sampling weights and priorities ([prioritized experience replay](https://arxiv.org/pdf/1511.05952.pdf) based on expert demonstrations).

Intermediate report CIT

2019 Week 18

Work done in past few weeks:

* labelled the latest drilling session but testing needs to be done on the new data set.
* A material detection algorithm with good accuracy based on the data from AD10.
* Starting to build Reinforcement learning framework based the result from convergence test with supervised learning.
* Implemented a prioritized sampler based on the importance sampling weights and priorities ([prioritized experience replay](https://arxiv.org/pdf/1511.05952.pdf) based on expert demonstrations).
* Built a loss function based on the paper deep q-learning from demonstrations ([DQfD](https://arxiv.org/pdf/1704.03732.pdf)) to initially train on the expert demonstration data and then continue training on the drill setup.
* Tested the dqfd using the openAI gym lunarlander-v2 environment.

1. Using the already trained simple dqn algorithm as expert, the demonstration data needed for dqfd algorithm is collected.
2. Now the DQfD algorithm is pre-trained using just the demonstration data and after it is trained, it allowed run on the virtual environment directly.
3. The pre-trained DQfD algorithm started to perform well from the start, which can be seen from the accumulated reward of each episode.

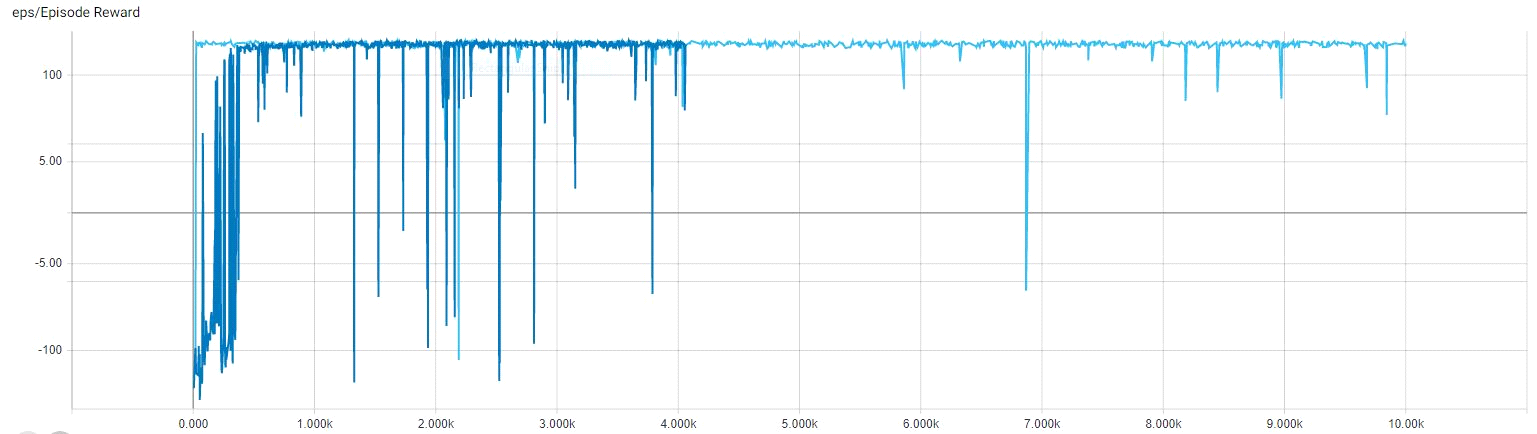


Fig1: In this figure the dark blue represents the training of simple dqn's and the light blue represents the DQfD's accumulated rewards per each episode of lunar-landing virtual environment. (x-axis: number of episodes/sessions, y-axis: cumulative reward per episode)

In-progress for next 2 weeks:

* Implement a real-time and testable material detection algorithm.
* Build environment for drilling (reward function, terminal states, transitions etc.)
* Train the DQfD algorithm on the cleaned and pre-processed drill data and finally test it on the drill once everything looks good.

Intermediate report CIT

2019 Week 20

Work done in last week and this week:

* A material detection algorithm with good accuracy based on the data from AD10.
* Implemented a real-time and testable material detection algorithm.
* Built a simple automatic drill controller based on material detection, T.O.F data (depth) and readings from ad10
* Tested material detection and simple automatic drill controller today (15th May ’19).
* Both the material detection and the simple controller produced good results (i.e. we have working versions of material detection and automatic controller :D) but must be fine-tuned/tested for consistency and better performance.

In-progress for next 2 weeks:

* Build smart phone app for control, visualization.
* Finetune the material detection and automatic drill controller.
* Build environment for drilling (reward function, terminal states, transitions etc.)
* Train the DQfD algorithm on the cleaned and pre-processed drill data and finally test it on the drill once everything looks good.