Quessions:

Regarding the API

1. For receiving data(to be stored in influxdb), are you ok with the sample input expected by the API, as given below?

    http://localhost:8080/main/save?line=temp,machine=machine1 c=105 1578248588000000000

2. Will the data sender, send the time precision also, like below?

    http://localhost:8080/main/save?**precision=ns&**line=temp,machine=machine1 c=105 1578248588000000000

or do we need to set it to a specific precision?

3. Are we going to send back the data to industry? If so can please address next 2 questions
4. For sending data, what is the output format of data, can we send lineprotocol data as such,

or the industry/machine expects data in specific format?

5. What data need to be send, does the industry send as the query(filter conditions) to fetch the data from influxdb
6. I presume you have MATLAB simulator and Influxdb. This API is to send data from MATLAB to influxdb and vice versa. Is my understanding right?
7. I understand that the data in influxdb will be sampled and sent to server via MQTT. May you please address below questions on this subject
    - What kind of server it is?
    - How do we sample the data that needs to sent to the server, need the logic to sample the data
    - What does the server do with the data sent?
    - Why MQTT is choosen?

Regarding future part

1. MES and maintenance data that will be sent as time-series data?
2. As you say, example data are "how long machine is running, problems faced",

To get better understanding, it would be great if you can convert this into time-series data (as it needs to stored in influxdb)

3. For data-preprocessing (like outliers removal), do you have any algorithm in mind to do it?
4. For data-processing
    - Can you elaborate more on the diagnostic engine/machine learning part
    - Can you elaborate more on the prognostics engine/machine learning part
5. Diagnostics and prognostics data are sent for overall analysis. What kind of analysis it is?

Answers:

1. Sample input into Influxdb

http://localhost:8080/main/save?line=temp,machine=machine1 c=105 1578248588000000000

The above link is of localhost. I can't access that data.

2. Data with time precision:

http://localhost:8080/main/save?**precision=ns&**line=temp,machine=machine1 c=105 1578248588000000000

We need to set it to a precision.

Line Protocol Syntax:

```
<measurement>[,<tag_key>=<tag_value>[,<tag_key>=<tag_value>]]
<field_key>=<field_value>[,<field_key>=<field_value>]
[<timestamp>]
```

Example data:

Sample freq. = 20000Hz
Simple python, read from csv and return from csv.
Every sec. 20k of data should pass to influx.
Every 1/20k rows should be stored in the database.
Accelerometer data and fault class given, sampling freq is 20000Hz.

(https://in.mathworks.com/help/predmaint/ug/condition-monitoring-and-prognostics-using-vibration-signals.html) OPEN LINK FOR REF

3 & 4. Sending data back to the industry & Output format of data:

We set up a server and the analytics will be done within the intranet. As it's sensitive data we should only send/process within the intranet. And the maintenance data will be taken from the industry such as a particular machine's historical data with when the machine got set up in the industry, any faults/breakdown occurred previously, how's the working condition, etc.

Some devices will send data encoded in binary. Some will send data represented as JSON or XML. Some may even just send the data as text.

The data will depend on the type of sensor.

Many sensors are very slow, small, low power devices - imagine a contact sensor for a door which connects using Zigbee. In that case it would be insane to choose XML as the format for the sensor data - XML is much too heavy.

If you're asking about the actual message format,
As I'll be using MQTT protocol, we can actually reduce the header size to a fraction of what HTTP uses.
We can also define the payload of your message.

Most applications use an application-specific arrangement that optimizes the data in the payload down to the byte level.

For example, I deployed a remote temperature sensor that worked with MQTT a while back.

The data I needed to send was:

1. The temperature reading

2. The vibration readings (acceleration, gyro)

2. The timestamp

3. Sensor ID

4. Key

Since I was measuring ambient temperature with a 0.1deg C accuracy, I used a 2-byte allocation for the temperature value, a 4-byte allocation for the Hex Timestamp, a 2-byte allocation for the Sensor ID and a 16-byte allocation for the internal application key.

So this was a total of 24 bytes plus some MQTT headers.

The message can be modified based on your application and message requirements, but it is quite flexible from a development point of view. Some developers like to add start and stop bytes, checksum information and other message validation data on the payload as well.

Headers and latency is affected when you use SSL or other encryption with your messages. It depends on what applications we will be using.
---

5. Fetch data from database

Sensor data is to be recorded to the influxdb database. We will send the data as queries to fetch data to/from influxdb.
I use sensors like Accelerometer, gyro, temperature and gather the data at the server.

6. I am working with Matlab & Simulator and Influxdb. Yes, the API's which I want you to develop need to send data from Sensor to influxdb and vice versa.

7.

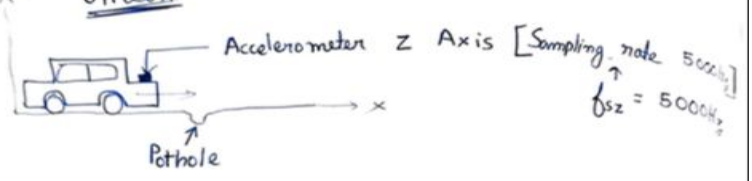
A. The data in influxdb will be sampled and sent to the server via MQTT. The server is a junction where the data from the sensors and the maintenance data of the machines will be accumulated and stored in their respective databases (influxdb). And then the analytics will be done on that data.

Server is a system with all the access to the data from sensors, MES, Maintenance. Using that we do analytics on data.

B. Sampling the data:

Sampling

Given



Problem :- Find out when car hit pothole with very less processing power.

Solution

① $f_s \geq 2 f_m$ [Sampling theorem]

$f_s \rightarrow$ Sampling freq.

$f_m \rightarrow$ Highest freq. content in the signal that's getting sampled.

! Consider $|\overline{f_m}|_{max} = 100\,Hz$.

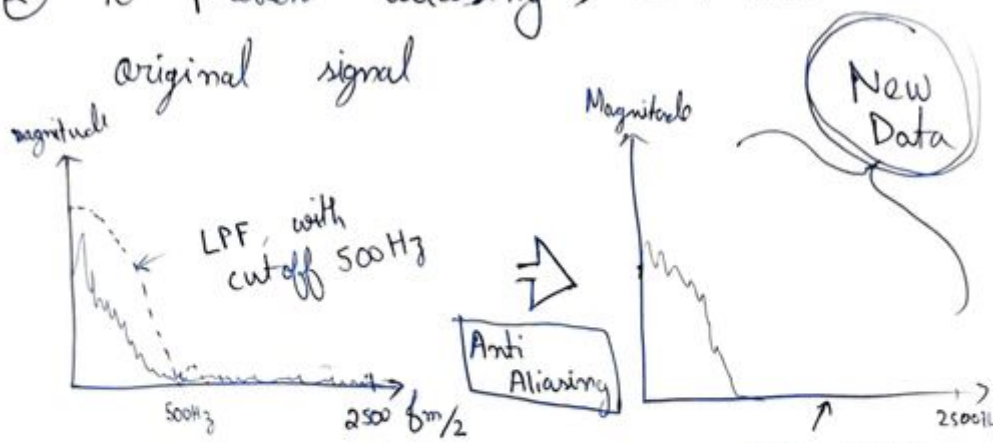
So, $f_s \geq 2 \times 100 \Rightarrow f_s \geq 200Hz$.

! Note if we make $f_s = 500Hz$, we would still be able to reduce no. of samples from 5000/s [5000Hz] to only 500/s [500Hz]

! This means we need downsampling.

$D = f_{sz_{old}} / f_{sz_{new}} = \frac{5000}{500} = 10$. So downsampling factor, D = 10

② To prevent aliasing, LPF the original signal

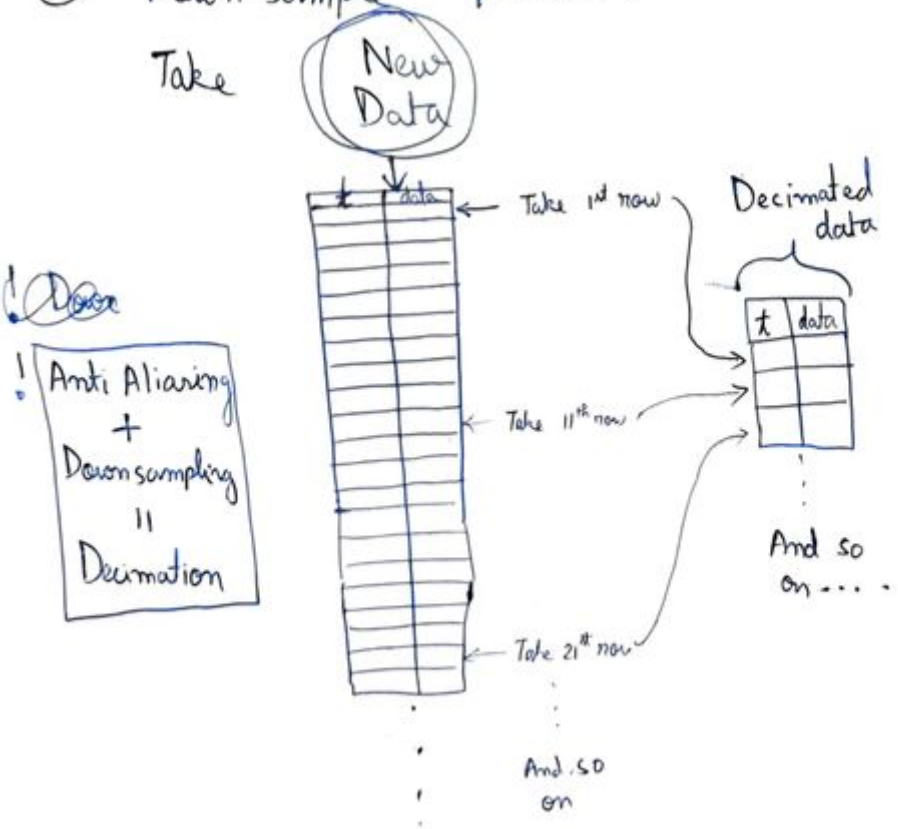


Notice noise also removed along with unrequired high frequencies.

! We have still retained 99.9% of all data

③ Down sample process :-

Take



! Down

! Anti Aliasing + Down sampling = Decimation

C. The server can be installed by us, or installed as an application on the client's server. At the server data processing will be done to the raw data. We use ML algorithms by doing data analytics for diagnostics (fault detection) and through diagnostics we determine prognostics (remaining useful life of a machine). All the data should be stored separately in different databases. Everything should be dynamically be able to be monitored at the dashboard.
D. **Reason for choosing MQTT**: It is a lightweight protocol for sending simple data flows from sensors to applications and middleware. MQTT suits small, cheap, low-memory and low-power devices.MQTT specializes in low-bandwidth, high-latency or unreliable networks, it is an ideal protocol for machine-to-machine (M2M) communication.

Regarding future part:

1. MES and maintenance data is what we get or gather from the company. It comprises the machines history, how long the machine has been running, when did the machine get installed, did the machine breakdown anytime in the past, did the machine get any service done, etc. All that data needs to be stored in different databases.
2. I work on machine's temperature, vibration data, for example I'll work on the gearbox. I'll place the sensors on the gearbox and will take the vibration data.

For example I have taken the gearbox fault diagnosis data. I have attached the dataset. Please go through it.

- Gearbox Fault Diagnosis Data set includes the vibration dataset recorded by using SpectraQuest's Gearbox Fault Diagnostics Simulator. Dataset has been recorded with the help of 4 vibration sensors placed in four different directions. Data set has been recorded under variation of load from '0' to '90' percent. Data set has been recorded in two different scenario: 1) Healthy condition and 2) Broken Tooth Condition

3. For removal of outliers, etc, I still haven't got any algorithm yet. I'm thinking of using Autocorrelation, regression, etc.

4. **For data-processing:**

- I'll check for the faults of machines in the diagnostics engine.
- I'll check for how long the machine will be healthy and what would be it's status in the future in the prognostics engine.
- By using ML algorithms, Data analysis is done after storing the diagnostics data and prognostics data.

Suggestions:

On the graph
Show small star mark for error/fault … if it hovers
Vibration increase - hover over red dot…

Fmea - what symptom , what error, explanation so, in ui when subsystem was about to fail, do fmea, and give description
Send notification to phone - actionable insight

I need One config file where you can Package code: which sets up db, data handling in db, everything can be done using that config file.

Master code:
With Python influxdb instance open
Create python file,
Script to automatically enter credentials if any

We will be showing the data to the industry. We set up the server at the industry.