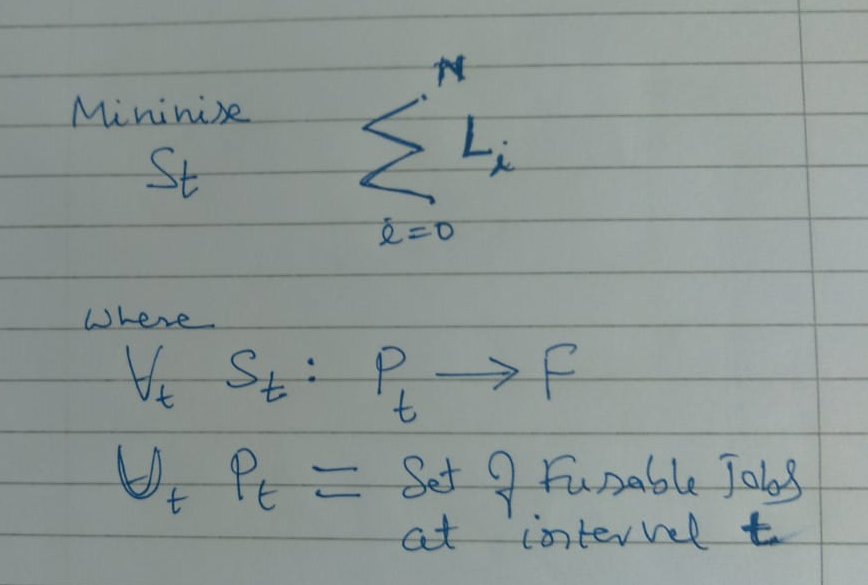
**Problem formulation for Job scheduling to improve QoS in IoT-Fog Network - using iFogSim simulator**

the problem can be formulated as:



**where:**

Total IoT-Fog system (Simulation) time **T** is divided into **N** equal duration intervals denoted as **It (I1, I2, I3…. In).**

We assume fixed number of fog devises and denote the set of fog devises by **F.**

The workloads are in the form of jobs, where **Jt** is the list of jobs arrived at time interval **t**,

All new jobs created at the interval **It** are denoted as **Nt** .

all active jobs being denoted as **At**

If job **j ∈ Nt** cannot be allocated to a fog devise then it is added to a wait queue **Wt**

All created jobs that are not active and are not in the wait queue are considered to be completed

We consider the problem of minimising Latency **L**i objective score accumulated across all intervals **It** **(I1, I2, I3…. In).** in a bounded time, experiment.

We denote the Latency score for interval (**It**) by **Lt** and consider a total **N** interval in an experiment.

We denote the utilization metrics of all fog devises in interval **It−1** as **Ut** . Now using **Ut** , we need to predict a scheduling decision **St**.

All tasks for jobs in **Nt ∪ Wt ∪ At** are called feasible tasks.

# **Job Scheduling Algorithm:**

Facts:

* Things(sensors) generate jobs, fog nodes also generate jobs, jobs are of two kinds periodic or event based.
* Jobs are scheduled at induvial Fog Nodes for beater QoS that are near to sensor (Things) that create job.
* We use FCFS algorithm to schedule jobs at each fog node and will observe the QoS (in terms of delay and Network usage)

Steps:

1. Add all jobs (arrives from senser/created by fog node) arrived at **i-th** interval to **JOBS\_LIST** based on arrival time.
2. Pick the first **JOB** from **JOBS\_LIST** for processing and remove it from to JOBS\_LIST.
3. Process the selected **JOB**, keep account of QoS parameter **Li** for that **JOB**.
4. Repeat step 1,2 and 3 for i = 0 to N intervals that will complete the simulation time.
5. Print the QoS parameters (average delay)