Homework 3 : Clock Synchronization

CS 675 – Spring 2018

Umang Desai

Analyzing the log, we get three numbers.
1) The average round-trip time between your client and your server.
--> 0.0013604165739567098

2) the packet loss rate between your client and your server, calculated as the percentage of interactions that failed.
--> 0%

3) the average rate at which your client's clock is drifting relative to your server's clock, measured in terms of microseconds per second. So, for example, a result of "-75 microseconds per second" means that for each second that ticks on the server's clock, your client's clock only ticks 0.999925 seconds.
--> As our average offset is 0.0005445103541898662, clients clock ticks only 0.999455 seconds for each second that ticks on the server's clock.

Produce a short writeup that includes the following:
1) What language did you use and why?
--> Python. I used python because it provides time in microsecond precision. I was going to use java before, but as java provides time only to millisecond precision I was forced to use python.

2) How do we compile / run your code?
-->Just run the server.py and client.py in command line using the python keywork.

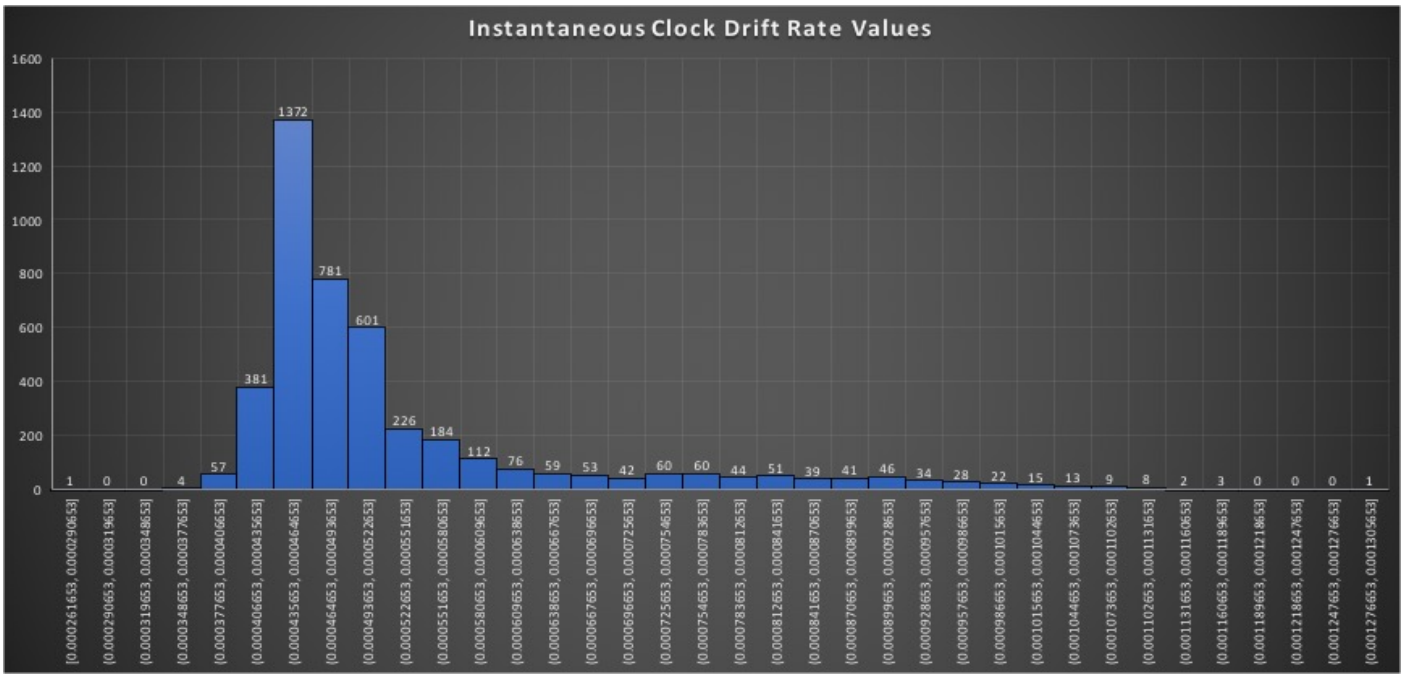
python server.py serverIpAddress serverPort
python client.py serverIpAddress serverPort

Here serverIpAddress and serverPort are the IP address and port where you want the server socket to bind to.

3) How did you calculate your client's average clock drift rate?
--> Adding all the offsets for each successful interaction and dividing it by total successful interactions. A basic average value of the set successful interactions' offsets.

4) What timeout did you pick to detect a failed interaction, and what happens if the server's response packet arrives after that timeout?
--> 1 second. I picked 1 second as it is more than enough for a interaction to execute between client and server. Delivery will only fail if there are failures on client or server side. Also if UDP packet takes too long to route, which did not seem to be the case in my experiment.

5) For each successful interaction, calculate the clients' average clock drift rate during the period since the previous successful interaction. Plot a histogram of these "instantaneous" clock drift rate values, and show that plot in your writeup. Hypothesize why the histogram is shaped as it is. (Picture also provided in folder for reference)



This seems to be a right skewed(non-symmetric) distribution where we can see that most of the concentration is at the centre. Most of the drifts are around 450 microseconds which is how it would be. If the drift is at milliseconds, then the clocks are working at much different rates. But since it is within half a millisecond, we can say that clocks are consistently drifting at an approximate rate of 450 microseconds. This also means that the network and consistency of delivery is good.