Characterization of Program Execution Time Distribution

Young-Kyoon Suh
Department of Computer Science
University of Arizona
{yksuh}@cs.arizona.edu

March 8, 2017

1 Observed Phenomena

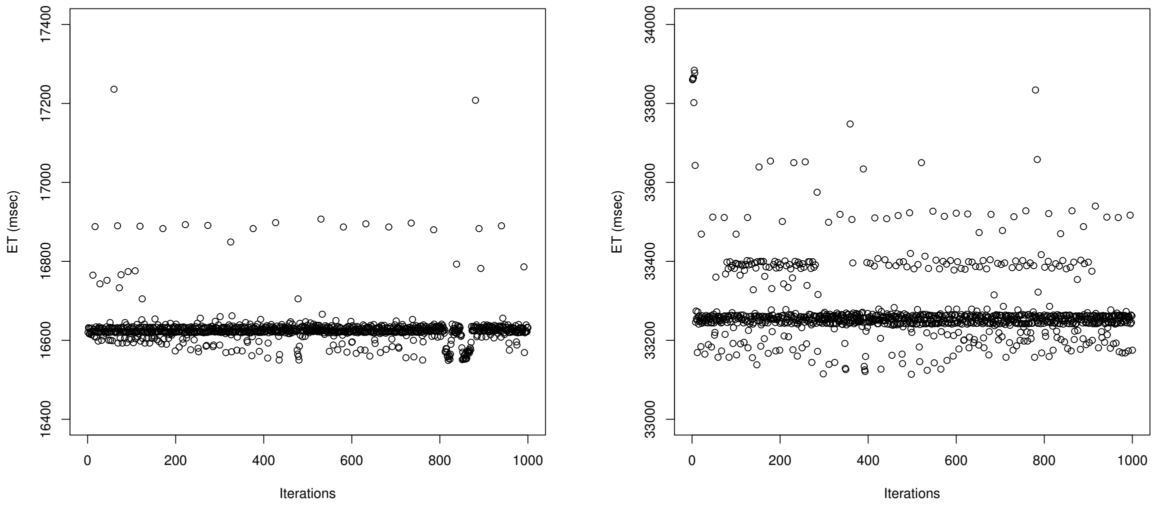
In this section, we discuss interesting phenomena observed from our measurements on a program-under-test (PUT) with a given task length. PUT is a simple compute-only program that runs a nested-for loop for a given amount of time (task length) and terminates.

1.1 Different Bands in Elapsed Time

We introduce an interesting phenomenon regarding the elapsed time (ET) of PUT. ET is computed as the difference of the wall-clock times measured before and after PUT execution.

Presumption Little structure in ET

Description Figure 1 shows the distribution of ET on PUT16 and PUT32. In this report, PUTx denotes the PUT with x-second task length. We observe from these data that some or many of ET records were concentrated within a certain interval, called *band* (or layer). The used data were obtained by EMPv2.



(a) Per-execution ET on PUT16

(b) Per-execution ET on PUT32

Figure 1: Per-execution ET on PUT16 and PUT32

Analysis As illustrated in Figure 1(a), there was the compact band formed around 16,650 msecs in the y-axis corresponding to most of the PUT executions. Another *weak* band around 16,900 msecs was formed as well. Furthermore, Figure 1(b) demonstrates this similar occurrence observed in PUT32. Most of the y-axis values were clustered around 33,250 msecs. There were also other higher bands consisting of around 33,400 or 33,500 msecs. For each experiment we indeed were able to see an intensively-clustered band (formed by around the median ET measurements) and a few bands weakly formed above and below the main band.

Hypothesis "*A high band seems formed due to daemon processes that periodically came up and ran for a moment, thus affecting ET at certain executions.*"

Test When we looked at the data in PUT32, a raw ET value (*e.g*, 33,500 msecs) exceeding its median appeared at the 547th, 573th, 600th, and 626th. The same ET value was seen at the 731th, 758th, 916th, 942th, 968th, and 995th. In other words, every 26 or 27 executions ET was higher than the median. In PUT16, the measured high ET value was also observed every 51 executions. Based on these investigations, relatively more executed processes were included in the high ET measurements. Considering there were no other processes except PUT, we infer that some daemon processes must have contributed to forming such a high band. This hypothesis, therefore, is supported.

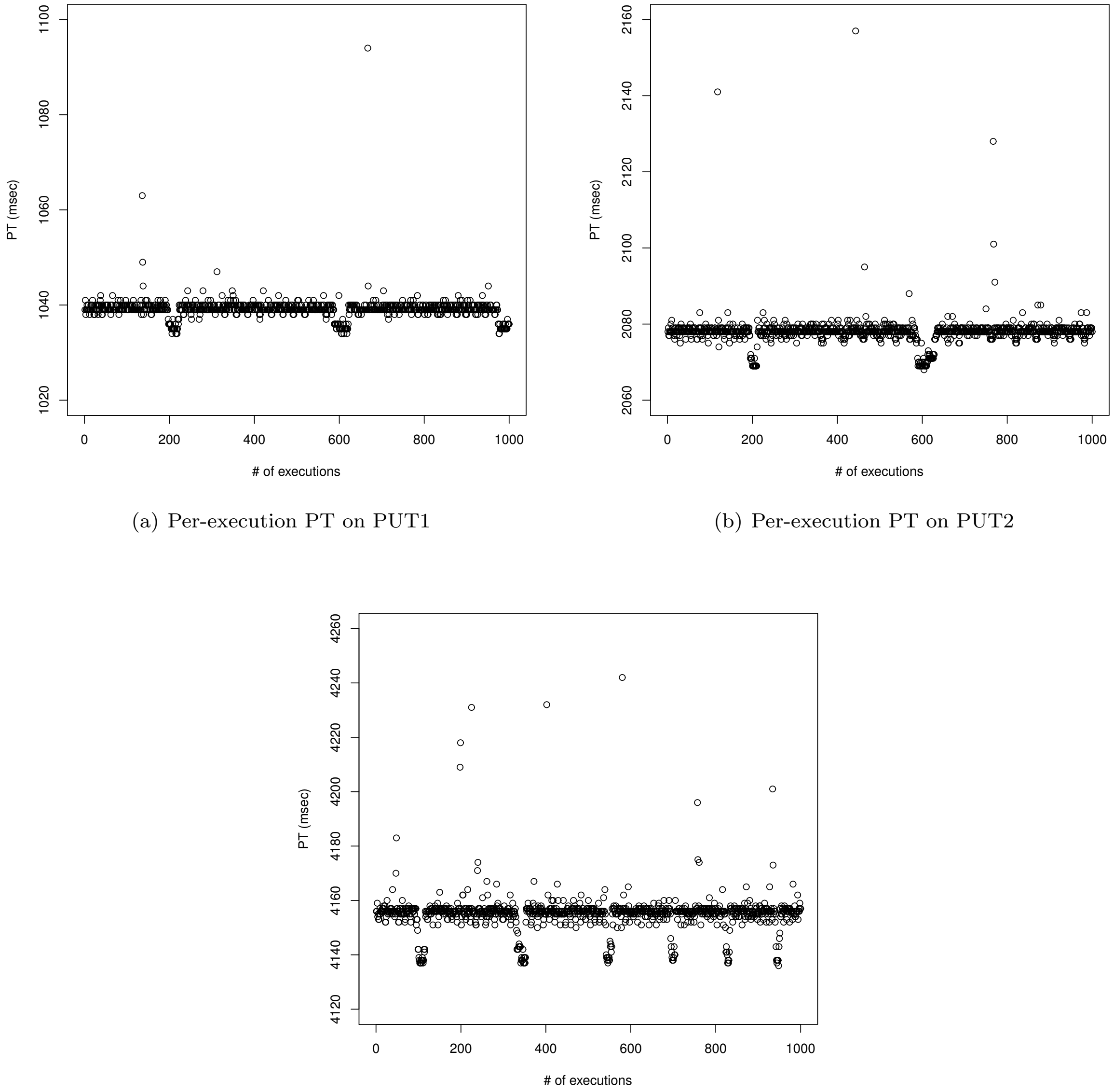
Status This cause is understood.

1.2 Periodic Dips in PT

We now turn our discussion to several phenomena regarding PT. PT is calculated as the sum of the user time and system time of PUT.

Presumption PT is totally flat.

Description In the measured data by EMPv2, we observed periodically-appearing dips. Figure 2 exhibits the distribution of PT in the experiments where the assigned task lengths were one, two, and four seconds.



(a) Per-execution PT on PUT1

(b) Per-execution PT on PUT2

(c) Per-execution PT on PUT4

Figure 2: Per-execution PT on PUT1, PUT2, and PUT4

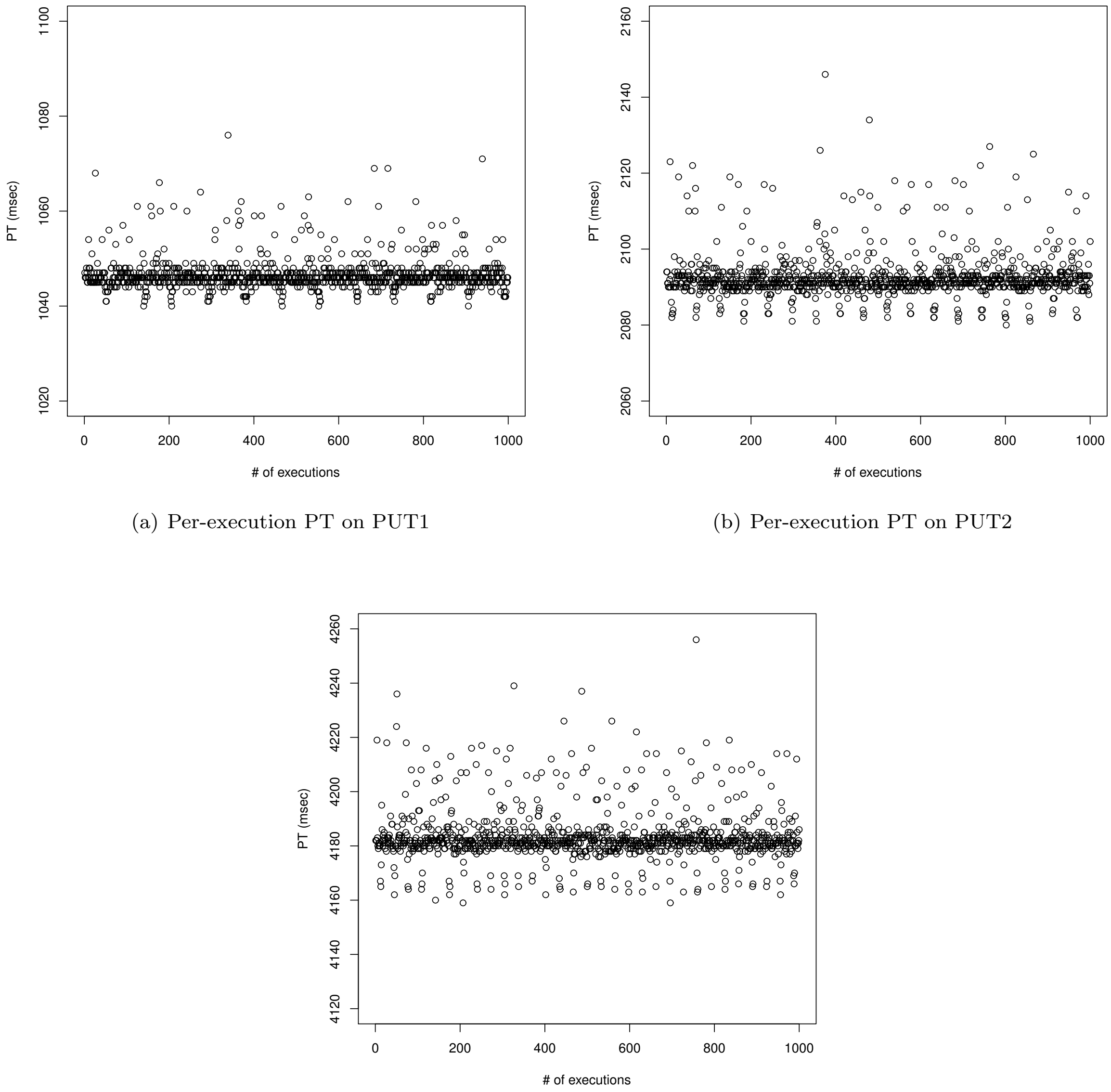
Analysis At first, the calculated PT values were plotted around the median. At some point, however, the PT values dropped to the minimum, and then the values bounced back up to the median. This pattern repeated after a certain number of executions; namely, this dip seemed to occur every around x seconds after the first appearance. Specifically, the dip first appeared after the 200th execution (about 200 seconds) and reappeared every approximately 400 seconds, as shown in Figure 2(a). In PUT2, these dips first emerged around the 200th execution, and they reoccurred after around 800 seconds, as illustrated in Figure 2(b). In Figure 2(c), the first dip was observed at around the 100th execution and repeated after approximately 800 seconds. Note that the x-axis values represents execution number, rather than actual time. The same execution number across different tasks do not mean that the respective actual times are same. The 200th execution in PUT2 and PUT4 could happen around 400 seconds and 800 seconds, respectively, after the tasks start.

Hypothesis I "*These dips are caused by adjustments by NTP (Network Time Protocol) [1] used to synchronize its local system time to a server in the network.*"

This hypothesis is based on the assumption that PT will be faster whenever the system time is corrected by the NTP daemon (`ntpd`).

Test To confirm this, we ran some additional experiments in which `ntpd` was disabled. In these experiments, EMPv1 was used.

The experiment results are shown in Figure 3. Unfortunately, the dips did not go away. Conversely, overall dips became thinner and appeared more frequently. For instance, the dips shown in Figure 3(a) were weaker and more frequent, compared with the ones in Figure 2(a). The same trend was seen in Figures 3(b) and 3(c) although less strong and somewhat seemingly random. In addition, the overall distribution of PT looked more dispersed, weakening the compactness of the bands.



(a) Per-execution PT on PUT1

(b) Per-execution PT on PUT2

(c) Per-execution PT on PUT4

Figure 3: Per-execution PT on PUT1, PUT2, and PUT4 when `ntpd` was turned off

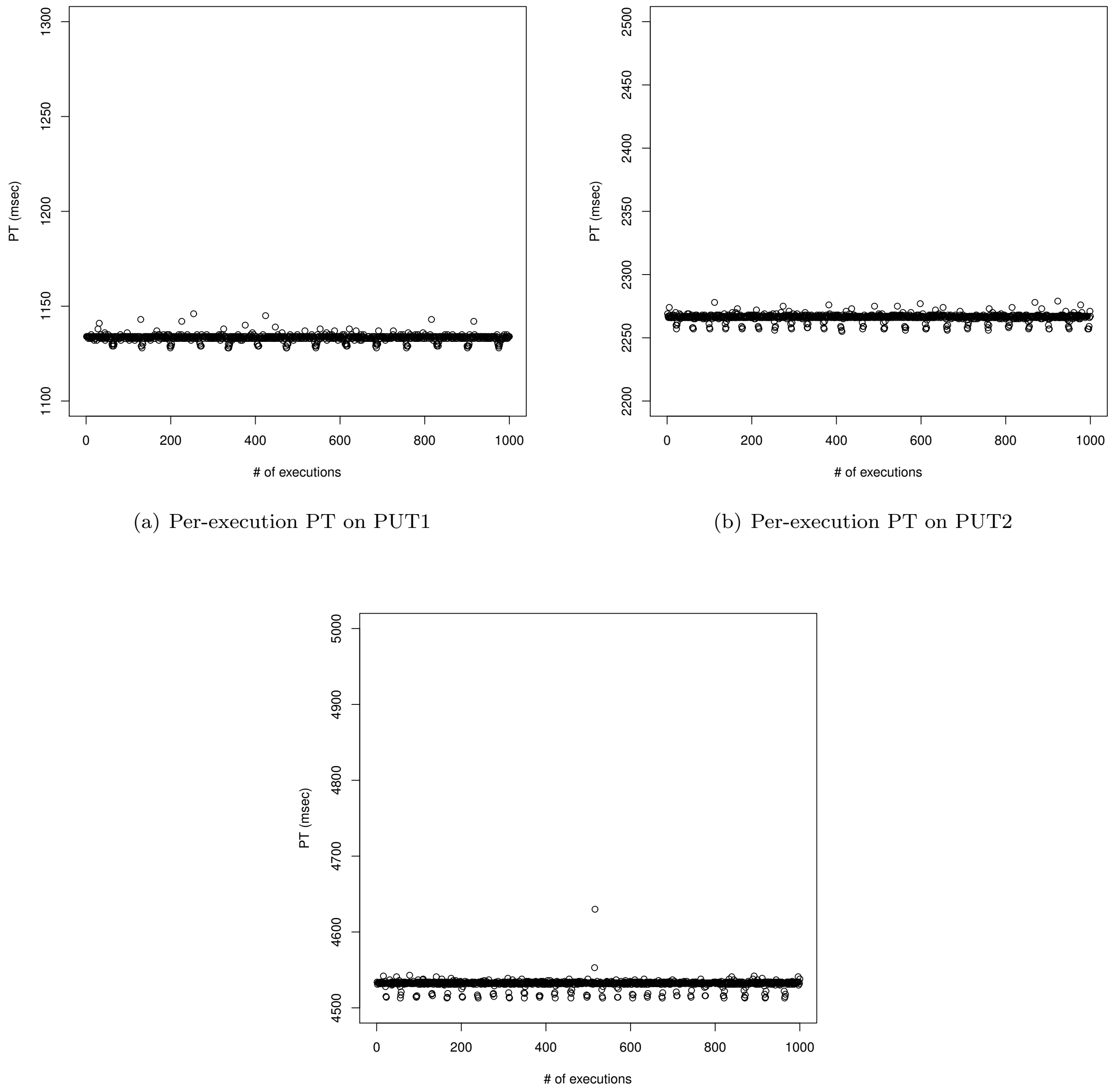
Turning off `ntpd` did not help us to figure out the cause of the dips. Our hypothesis, thus, is not supported.

Hypothesis II "*Disabling the Turbo mode set in a processor will make the dips go away.*"

Some processors support "turbo mode" and scale up the frequency and voltage at times. This hypothesis studies the impact of the frequency scaling caused by the turbo mode.

Test We found that the processor type of our experiment machine is Intel Core *i*7 CPU 870 @ 2.93GHz. This processor enabled by default the turbo mode through the feature named "Intel® Turbo Boost Technology". In addition, the turbo mode can be influenced by another feature of the processor, called "Enhanced Intel SpeedStep® Technology", which can reduce power usage/heat based on the load of the processor. To see if the Turbo mode was the cause of the dips, we disabled both features. The frequency scaling was stopped.

Figure 4 shows the distribution of PT on PUT1, PUT2, and PUT4. In these data, EMPv2 was used. Compared with Figure 2, overall the degree of dips became much more discrete and weaker. In addition, the standard variations of PT on the PUT tasks reduced to 1.65/2.78/5.75 from 2.39/4.33/7.52, respectively. This implies that turning off the turbo mode helped us reduce the variation of PT. As a result, most executions yielded flat PT measurements. Nevertheless, the dips were still observed.



(a) Per-execution PT on PUT1

(b) Per-execution PT on PUT2

(c) Per-execution PT on PUT4

Figure 4: Per-execution PT on PUT1, PUT2, PUT4, and PUT8 when the turbo mode was turned off

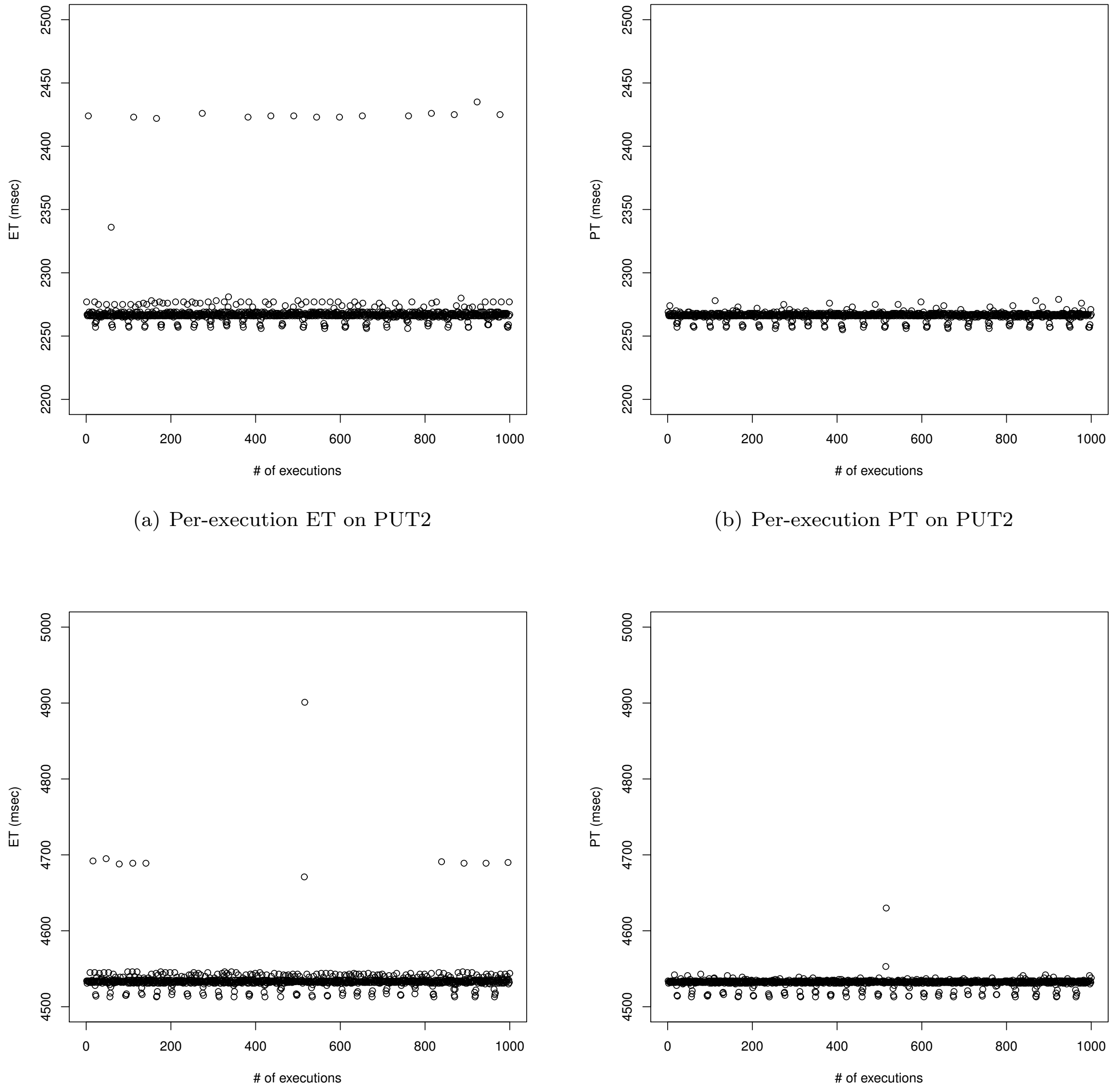
Status This cause is yet understood.

In subsequent sections, we show the experiment results after the Turbo mode was off.

1.3 Synchronized ET and PT

Presumption Totally correlated

Description Figure 5 shows ET and PT measurements on PUT2 and PUT4. The common phenomenon is that most of the measured ET and PT data looked synchronized. The phenomenon was observed from the EMPv3 data.



(a) Per-execution ET on PUT2

(b) Per-execution PT on PUT2

(c) Per-execution ET on PUT4

(d) Per-execution PT on PUT4

Figure 5: Per-execution ET/PT on PUT2 and PUT4

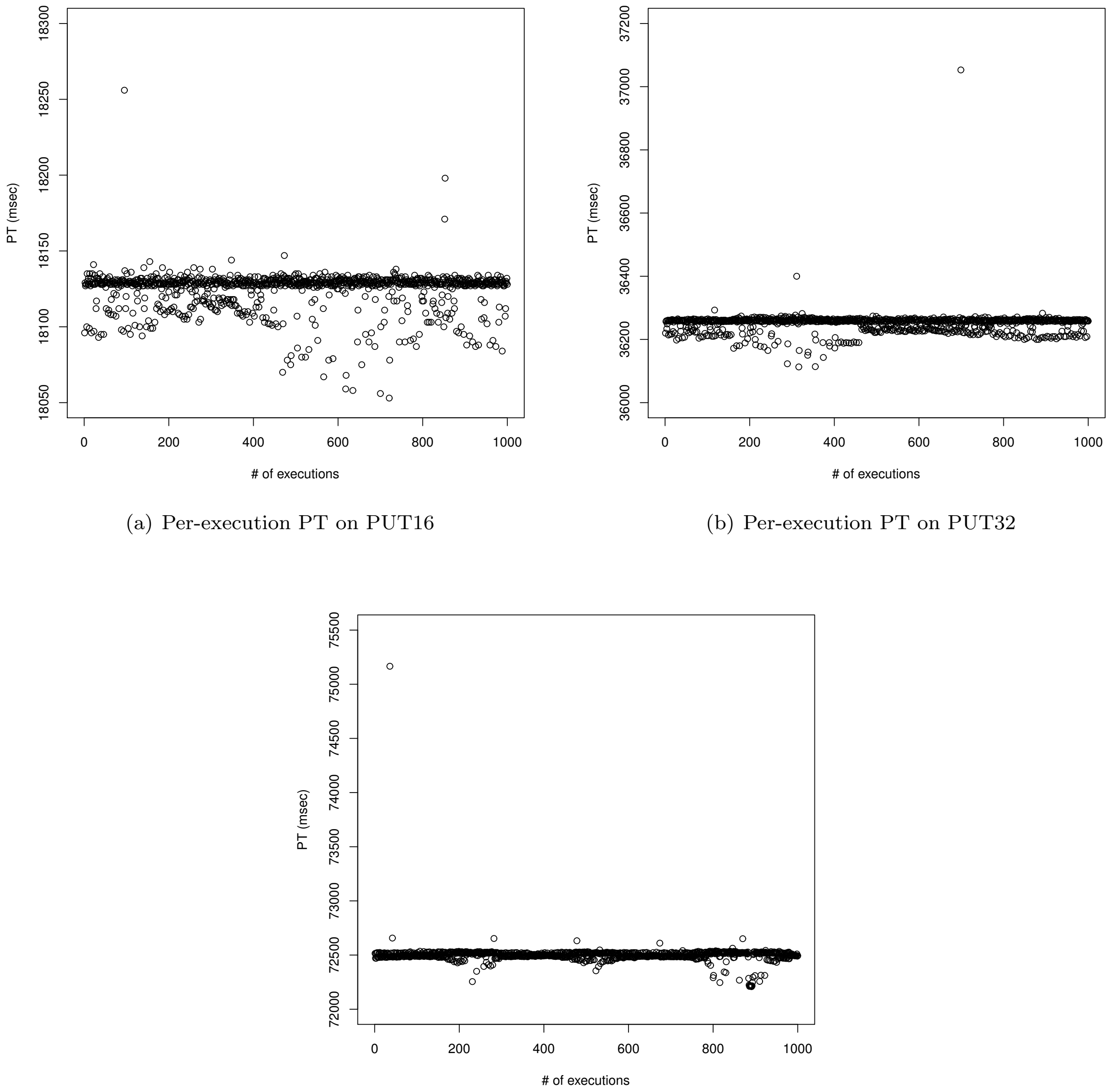
Analysis In PUT2, if we overlap Figures 5(a) and 5(b), we know that most executions have exactly the same ET and PT. This also holds true for PUT4, as shown in Figures 5(c) and 5(d). The synchronization does not take place in some executions, where some other daemon processes had indeed run together.

1.4 Outliers in PT

Presumption Large outlier in PT should not occur.

Description This phenomenon indicates that a couple of PT measurements showed exceedingly high peaks.

Analysis Figure 6 exhibits the PT measurements on PUT16, PUT32, and PUT64. These data were obtained by EMPv3. There were three outliers observed at around the 90th and the 820th executions, as shown in Figure 6(a). Likewise, a couple of outliers were also observed from the measurements on PUT32 and in PUT64, as illustrated in Figures 6(b) and 6(c).



(a) Per-execution PT on PUT16

(b) Per-execution PT on PUT32

(c) Per-execution PT on PUT64

Figure 6: Per-execution PT on PUT8, PUT16, PUT32, and PUT64

There are three common things that we can draw regarding this phenomenon. First of all, literally all the peaks are far above the main band. Second, every experiment reveals a few peaks. Lastly, their occurrence does not have any periodic cycle. In other words, the outliers are scattered all over the places. It looks irregular and appears randomly at a specific execution.

Hypothesis "*The outliers could be correlated with processes scheduled to run at some point during our experiment.*"

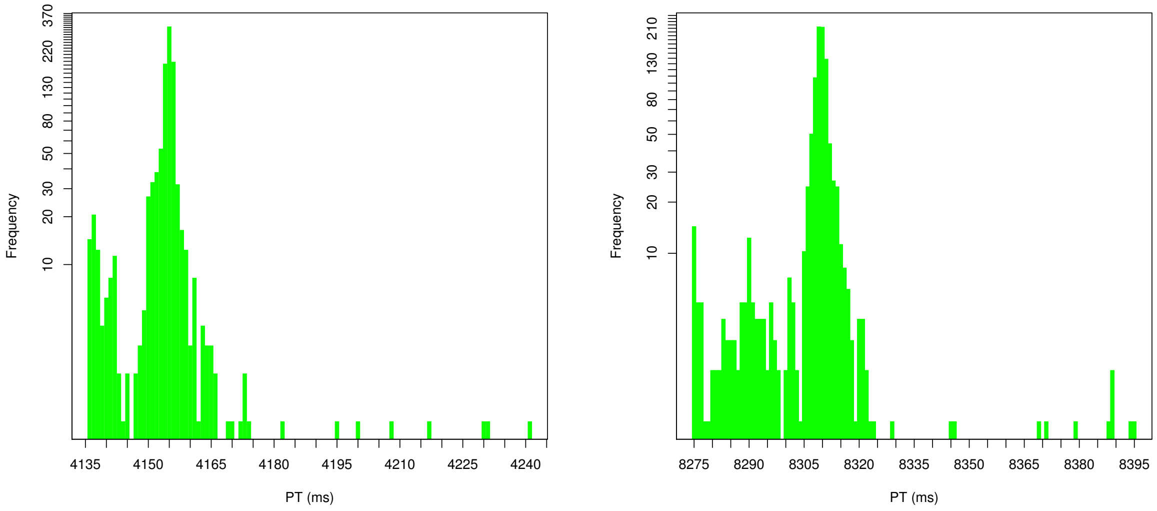
We looked at the data revealing the peaks in every experiment. Interestingly, *CPU_COUNT*, representing # of context swaps, is positively correlated with peak. PT grows linearly with the increase of # of context switches. The correlation coefficient of PT and the context switches is 0.67, which is quite high. (The correlation between PT and # of run processes is also positive.) Hence, this hypothesis is supported.

Status We have understood this.

1.5 Periodic Peaks in Frequency of PT

Presumption No peak except the median PT

Description Figure 7 shows the frequency of PT in the experiments whose respective expected running time is 4 and 8 seconds. This phenomenon indicates that a few peaks in frequency of PT are regularly seen. The phenomenon was observed from the EMPv2 data.



(a) PT frequency on PUT4

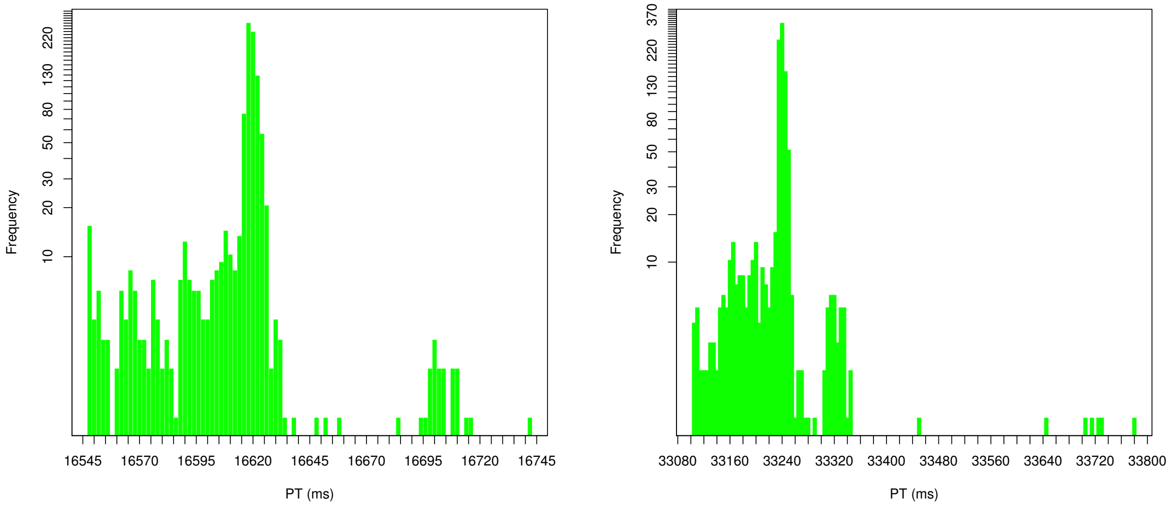
(b) PT frequency on PUT8

Figure 7: PT frequency on PUT4 and PUT8

Analysis In Figure 7(a), we could see two distinct peaks formed around 4,138 msecs and 4,142 msecs before the highest peak around 4,155 msecs. This pattern was also observed in Figure 7(b), where there were about three peaks shown at 8,275 msecs, 8,292 msecs, and 8,300 msecs before the highest one around 8,318 msecs. The peaks appeared to repeat every around 8 msecs. This periodicty seems formed by the same PT values on PUT4 and PUT8. In particular, these PT values were below the median. The low PT values were calculated when PT periodically went down to the minimum, as discussed in Section 1.2. As a consequence, periodic peaks in frequency were observed, as illustrated in Figure 7.

Hypothesis "*Heavier PUT will also have similar periodic peaks in frequency due to their low PT values.*"

Test Such similar peaks in frequency periodically appeared in the measurements on PUT16 and PUT32, as depicted in Figure 8. Again, EMPv2 was used for this test.



(a) PT frequency on PUT16

(b) PT frequency on PUT32

Figure 8: PT frequency on PUT16 and PUT32

Therefore, this hypothesis is supported.

Status The cause is understood.

1.6 Increasing Minimum and Maximum Difference in PT

Presumption Almost zero or very small constant

Description This phenomenon indicates that the difference between the maximum and minimum of the measured PT values grows linearly over increasing task length, as shown in Figure 9. Note that xy axes are in log scale. This phenomenon was observed from the EMPv2 data.

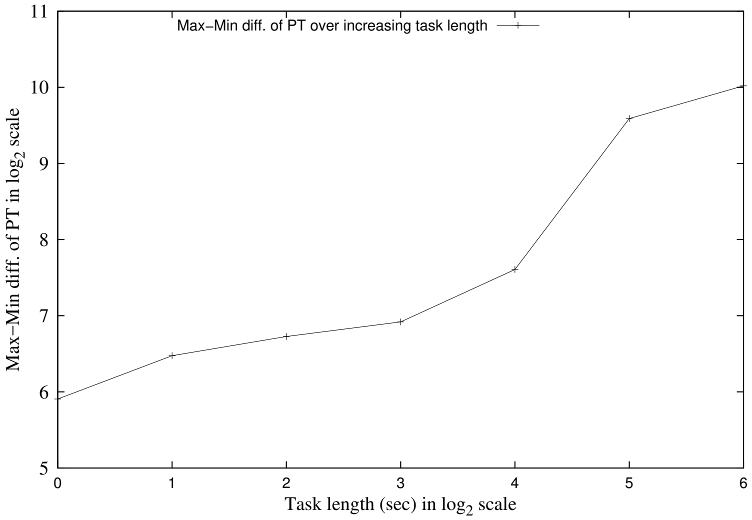


Figure 9: Increasing difference over increasing task length

Analysis As task length increases, the PT difference increases. Considering that the y-axis values are plotted in log scale, the difference grows linearly. It appears to be synchronized with the degree of increasing task length. Our expectation on PT was flat, and thus, the increasing difference was strange.

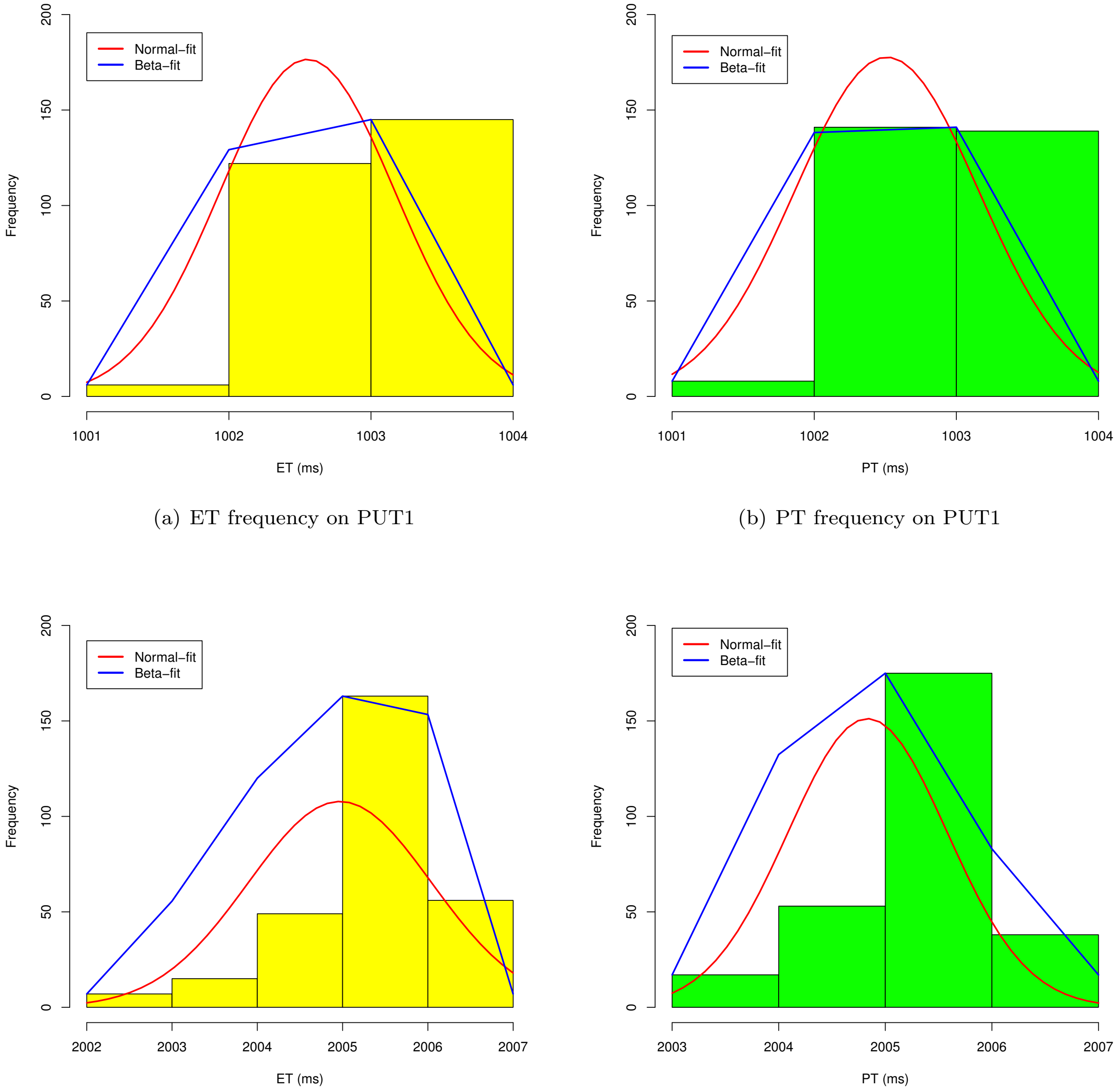
Hypothesis ?

Test ?

Status Not understood.

2 Exploring a Proper Distribution for Measured Data

Figures 10, 11, 12, 13, 14, 15, and 16 represent the histograms of ET and PT on the 1 ∼ 4096 second tasks. Nnormal and beta distributions are fitted on the histograms as well. For these histograms, we used EMPv5, which identified and removed outliers in its measurements.



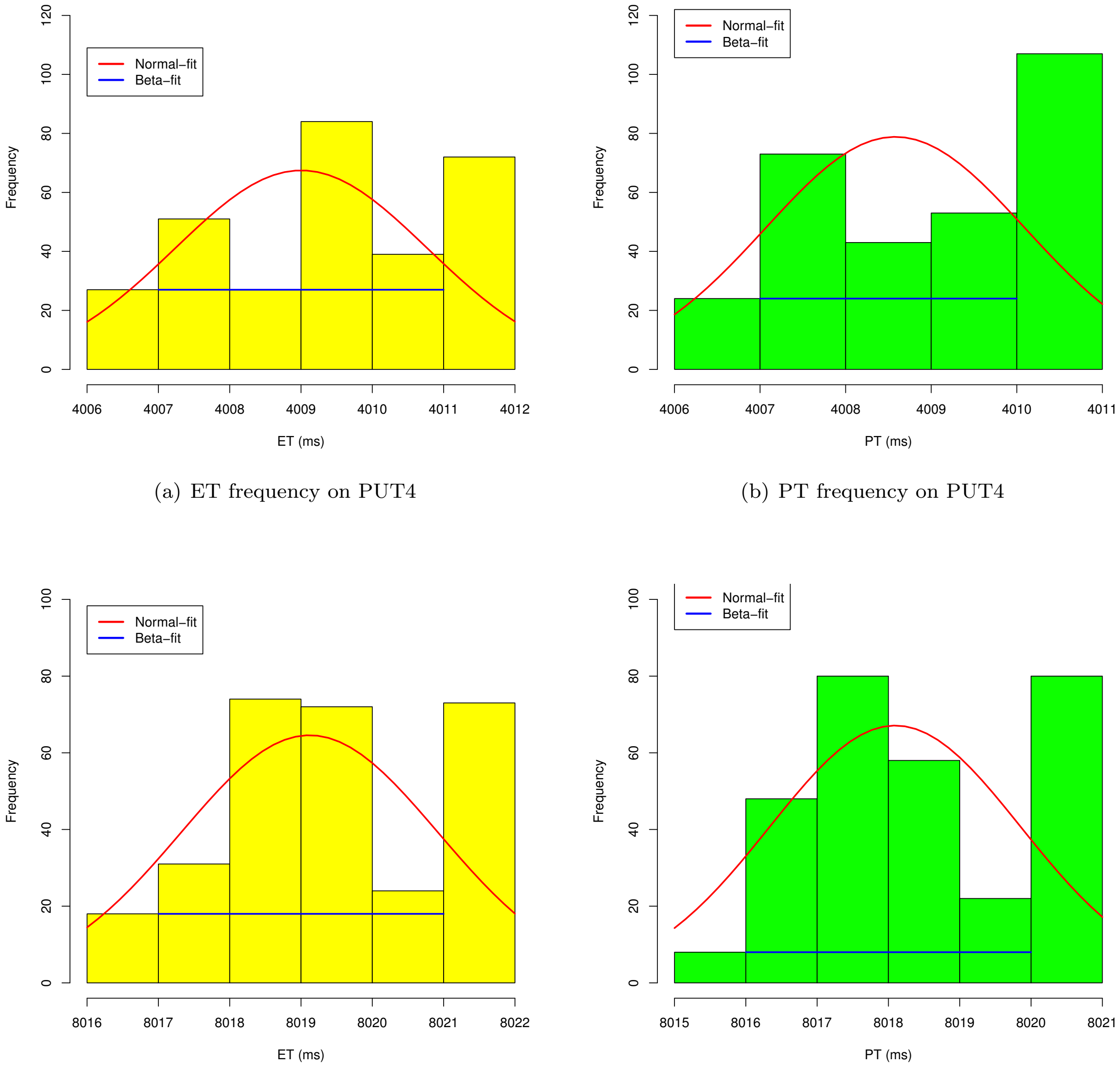
(a) ET frequency on PUT1

(b) PT frequency on PUT1

(c) ET frequency on PUT2

(d) PT frequency on PUT2

Figure 10: Normal and beta distributions fitting onto the histograms of the measurements of PUT1 and PUT2



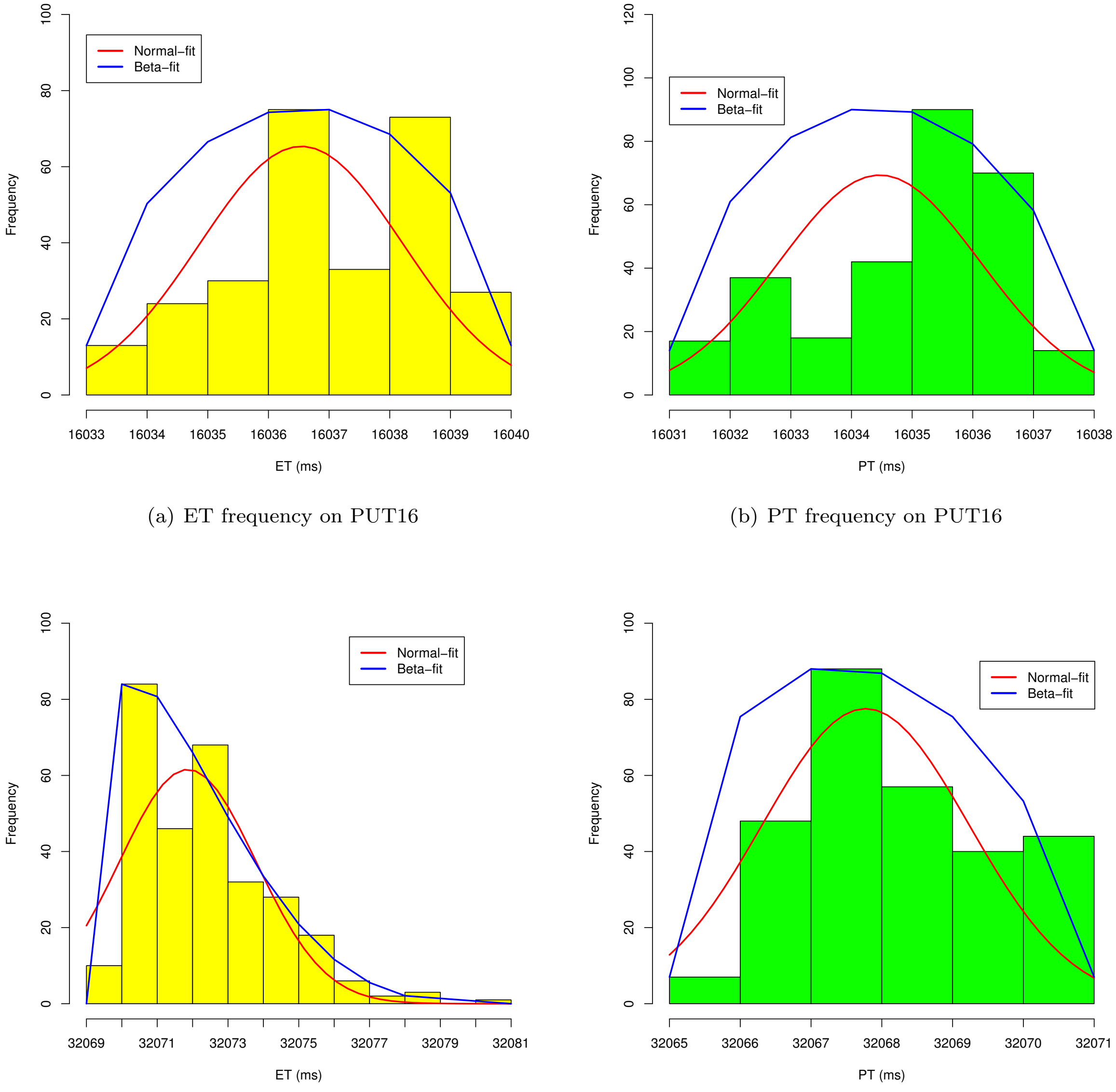
(a) ET frequency on PUT4

(b) PT frequency on PUT4

(c) ET frequency on PUT8

(d) PT frequency on PUT8

Figure 11: Normal and beta distributions fitting onto the histograms of the measurements of PUT4 and PUT8



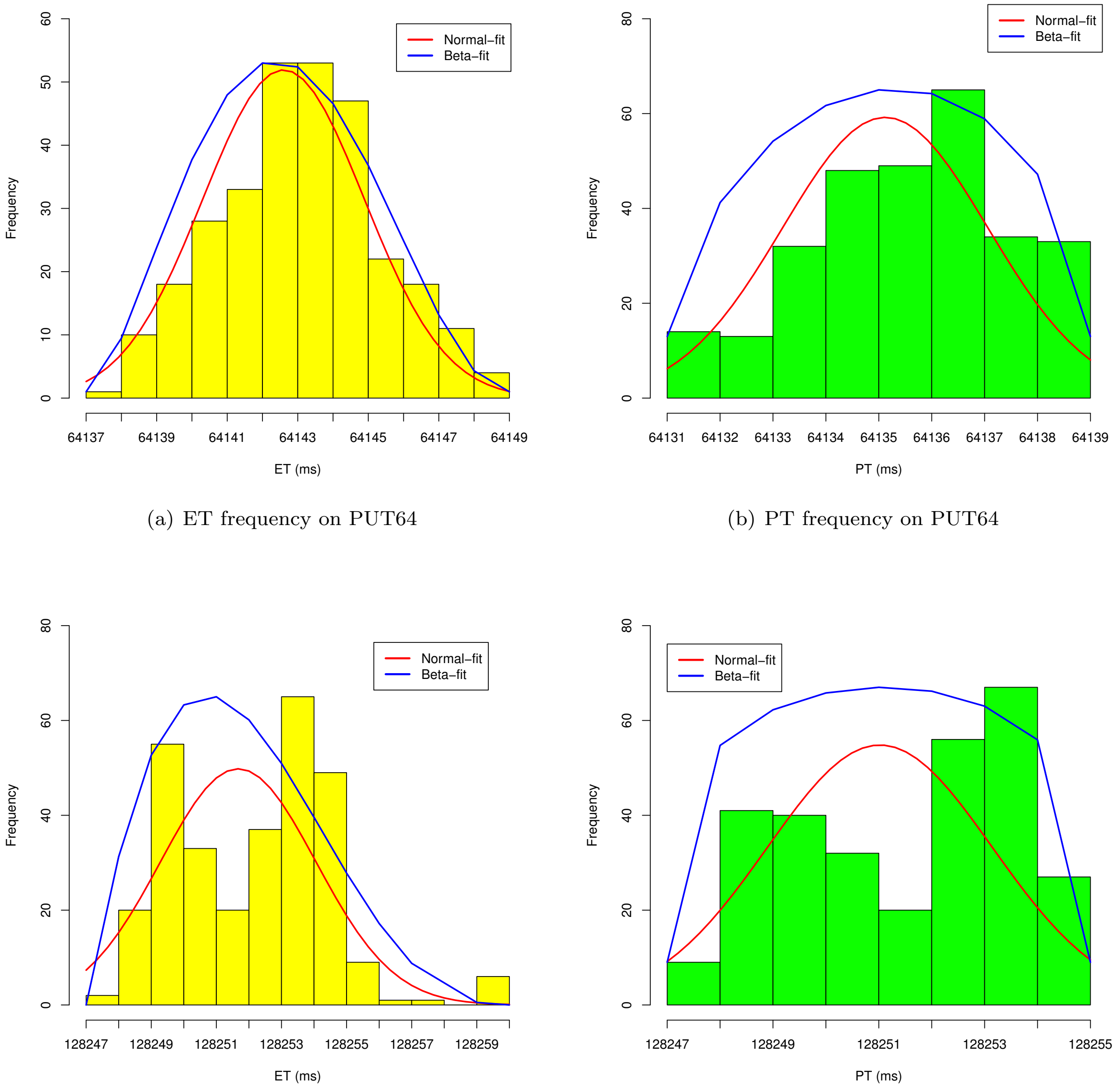
(a) ET frequency on PUT16

(b) PT frequency on PUT16

(c) ET frequency on PUT32

(d) PT frequency on PUT32

Figure 12: Normal and beta distributions fitting onto the histograms of the measurements of PUT16 and PUT32



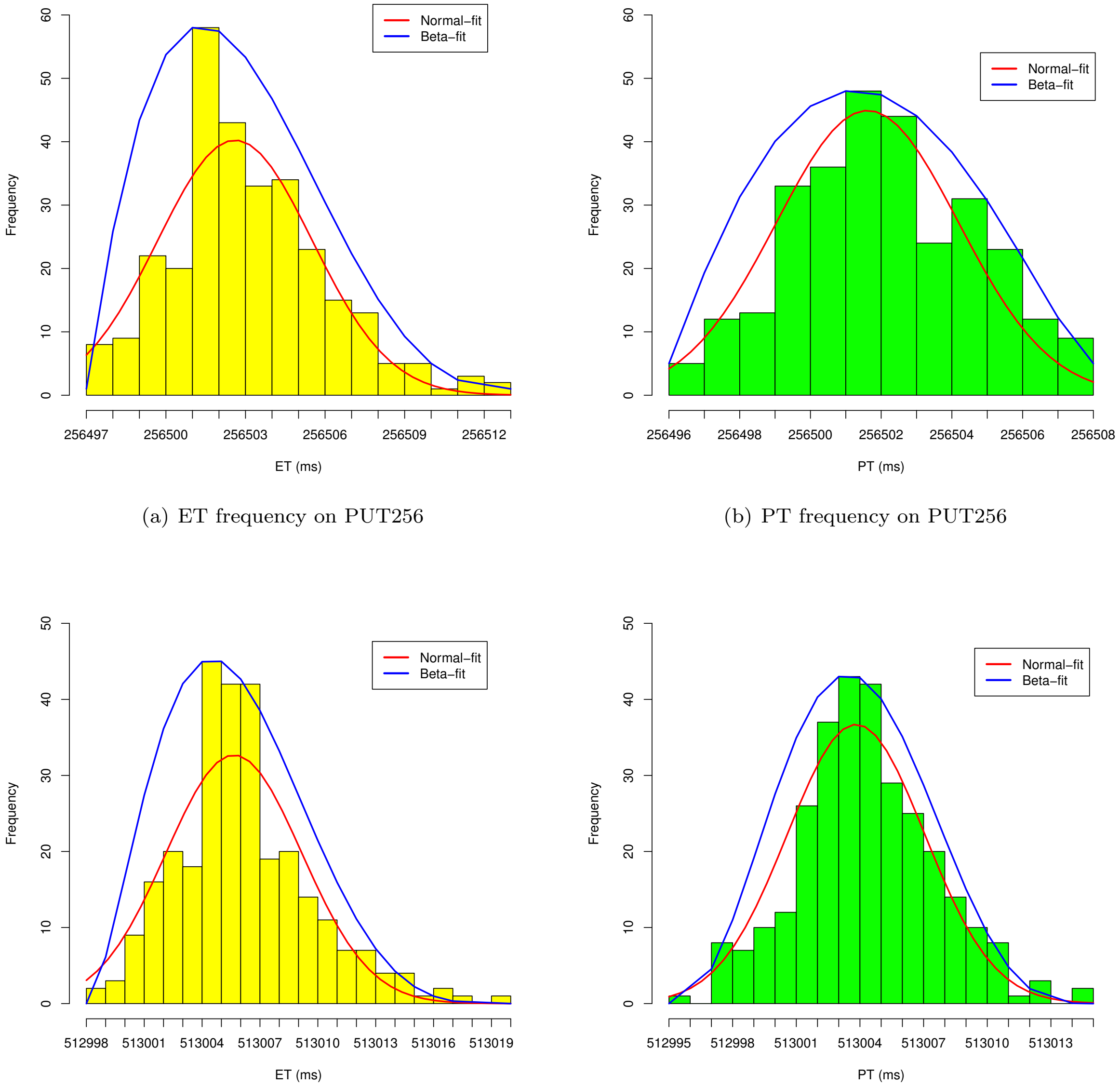
(a) ET frequency on PUT64

(b) PT frequency on PUT64

(c) ET frequency on PUT128

(d) PT frequency on PUT128

Figure 13: Normal and beta distributions fitting onto the histograms of the measurements of PUT64 and PUT128



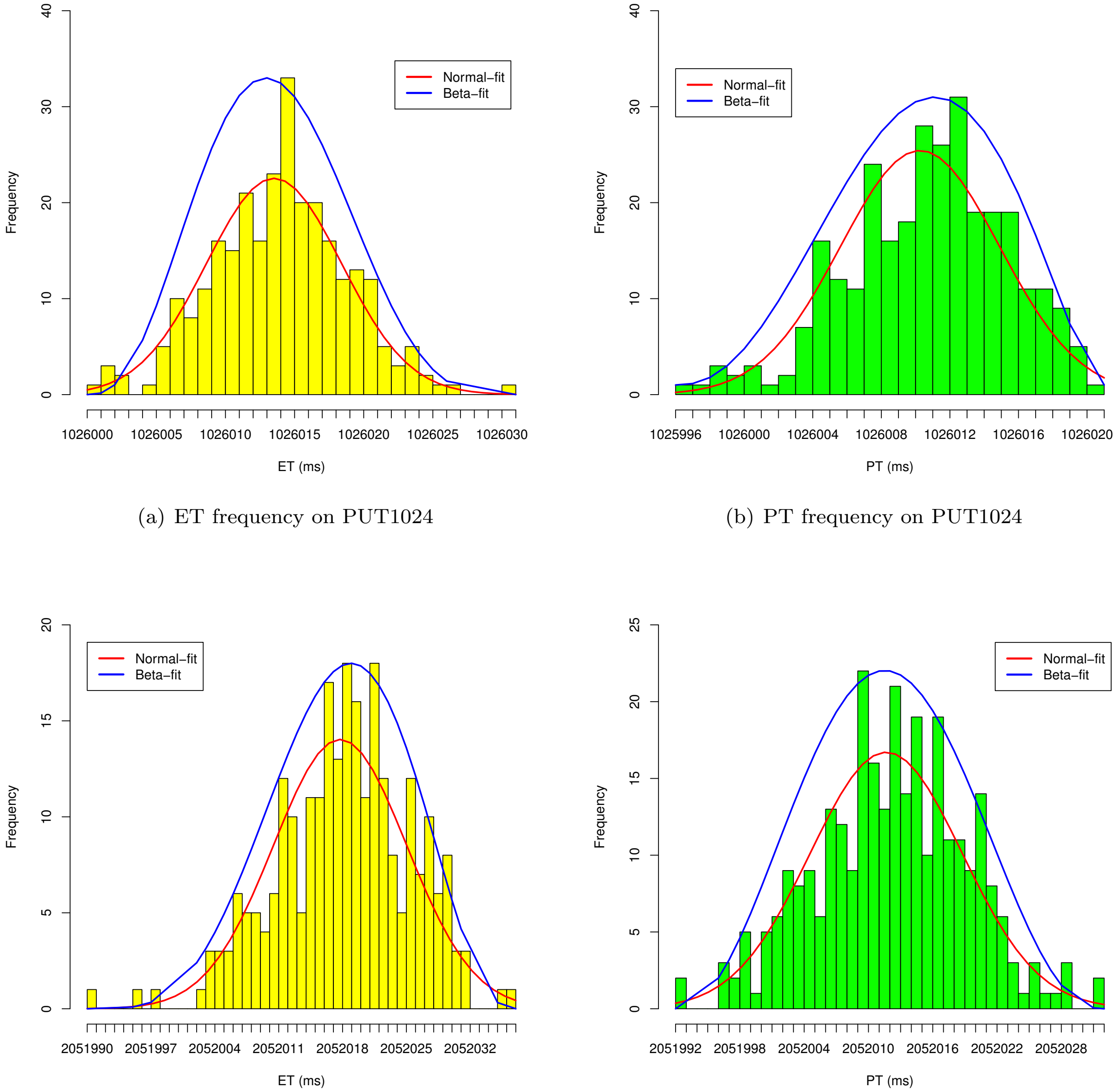
(a) ET frequency on PUT256

(b) PT frequency on PUT256

(c) ET frequency on PUT512

(d) PT frequency on PUT512

Figure 14: Normal and beta distributions fitting onto the histograms of the measurements of PUT256 and PUT512



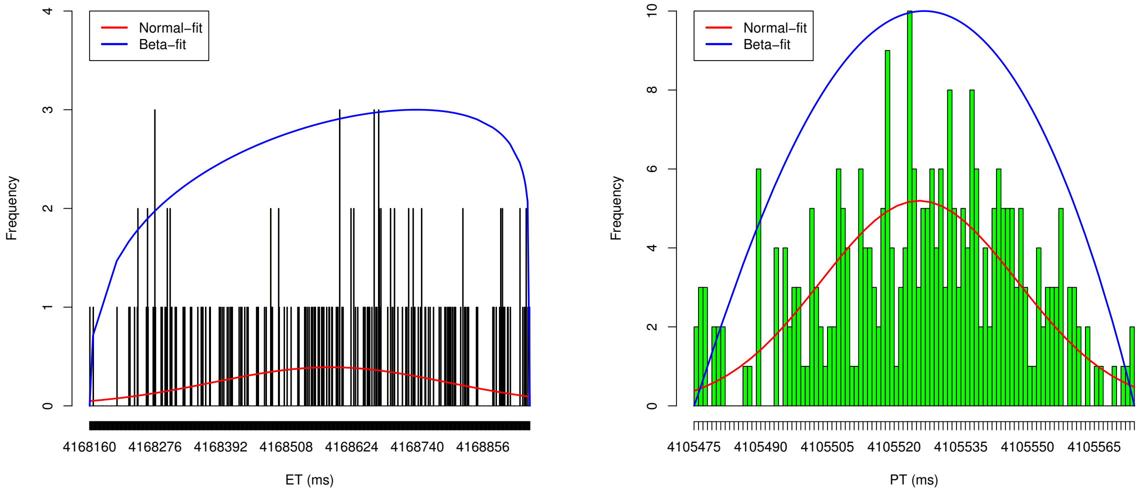
(a) ET frequency on PUT1024

(b) PT frequency on PUT1024

(c) ET frequency on PUT2048

(d) PT frequency on PUT2048

Figure 15: Normal and beta distributions fitting onto the histograms of the measurements of PUT1024 and PUT2048



(a) ET frequency on PUT4096

(b) PT frequency on PUT4096

Figure 16: Normal and beta distributions fitting onto the histograms of the measurements of PUT4096

References

[1] David L. Mills, "Internet Time Synchronization: The Network Time Protocol," in *IEEE Trans. on Comm.*, Vol. 39, No. 10, pp. 1482–1493, October, 1991.