

Author's Reply

April 3, 2017

Overview

I appreciate your review and address your remaining question below. Note that my response to your question is incorporated into the revision and marked in [blue](#).

Reviewer

<< Reviewer's comments to the author(s) >>

The manuscript is well revised.

However, I have a question how to estimate each daemon's influence as time. The estimation is important for accuracy of your proposed method.

For acceptance, please clarify the estimation method and its validity.

If I understand your question right, we actually *measure* each daemon's process time (PT) for each timing (sample) of a program-under-test, rather than estimate the influence of that daemon as time. Remind that PT is computed based on the measures of user and system time received from the kernel via the Netlink socket. To check whether the influence (in your terminology) was significant on each sample, we came up with an idea of the *cutoff* method by which we could eliminate a certain sample containing a daemon that ran over the respective cutoff time.

To determine the cutoff associated with each daemon, we first identify infrequent, long-running daemons (Steps 2—8). We then compute the cutoff of each so-identified daemon (Step 9). The detailed cutoff computation is specified in the second paragraph in the right column on page 3. The final cutoff data that we obtained for our identified daemon are exhibited in Table 1. For more details about the identification and cutoff calculation, please revisit the blue text in Section 2.

We applied the table to discard a sample involving a daemon whose PT is above the respective cutoff. Dropping a certain sample means that there happened such a daemon's significant influence on that sample. This indicates that we can't trust and include that sample in the base data to calculate the execution time of the given program.

The validity of our protocol using the cutoff was assessed on real-world programs (sort and matrix multiplication) and the SPEC CPU2006 benchmark. As shown in Figure 5 and Table 3, our protocol obtained better measurement quality for those real workloads, compared with that of the existing timing scheme using elapsed time.

Note that for better clarification raised from your question, I slightly modified the Contribution paragraph, provided a single-line comment about the cutoff computation, and finally added the last sentence in parentheses in the prose of Step 9. (Due to a space limit, more text could not be inserted.)

I hope this response suffices, but if I misunderstood your question, please kindly let me know. I will re-address.