

# Experimental study on round trip times of web applications

Rafi Khaled, Rui Meireles  
{rkhaled, rui.meireles}@vassar.edu

Department of Computer Science, Vassar College, USA

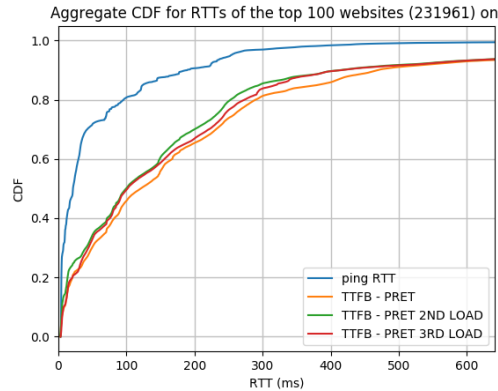


Fig. 1. Example figure

**Abstract—Replace by concrete abstract.**

## I. INTRODUCTION

Introduction.

Figure 1 is an example figure.

## II. RELATED WORK

Related work.

## III. METHODOLOGY

Collection of data was conducted using two Python libraries: `pytyping` and `PycURL`. `PycURL` is a Python interface to `libcurl`, which is a client-side URL transfer library. `PycURL` allows one to fetch various objects identified by a URL: in our case, objects corresponding to the URLs of various websites hosted on servers across the world [1].

First, we wanted to calculate the RTT as measured by a calculation involving, at the highest level, a websites pretransfer time and time to first byte: the RTT was calculated as the pretransfer time subtracted from the time to first byte, both of which were obtained with `PycURL`. To do this, it is as simple as creating a new `Curl` Object from `PycURL`, setting the relevant options, i.e. the name of the website and the `FOLLOWLOCATION` to 1. Setting the `FOLLOWLOCATION` to 1 tells the library to follow any `Location:` header that the server sends as part of a HTTP header in a 3xx response. The `Location:` header can specify a relative or an absolute URL to follow. The library will issue another request for the new

URL and follow new `Location:` headers all the way until no more such headers are returned [2].

The next calculation we wanted to observe is the RTT on the second load of a website. This calculation has the same equation of RTT as described above, except with the appropriate second load of the pretransfer time and the time to first byte. In order to measure this, we took advantage of `PycURL` allowing for reuse of `Curl` Objects: one only needs to reset the relevant options identically to the first load and can then measure the speed of the connection

## IV. EVALUATION

Evaluation.

## V. CONCLUSIONS

Conclusions.

## REFERENCES

- [1] (2016, Dec) `Pycurl` 7.43.0.1. [Online]. Available: <http://pycurl.io/>
- [2] `Curlopt`followlocation` explained`. [Online]. Available: [https://curl.haxx.se/libcurl/c/CURLOPT\\_FOLLOWLOCATION.html](https://curl.haxx.se/libcurl/c/CURLOPT_FOLLOWLOCATION.html)