Time Calibration in Experiments with Networked Sensors

Olivier Mehani, Ronnie Taib, Benjamin Itzstein

Rebasing time series from networked measurement devices into a single timeline

Readings from multiple physiological sensors are reported by several computers to a centralised collection server [1].

Physiological sensors have varying time signatures and data sizes.

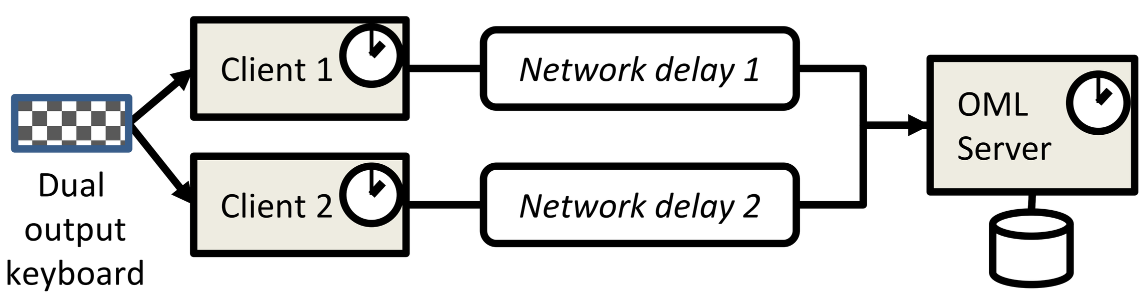
Sensor	Sampling rate	Data streams	Device
Eye tracking glasses	30 Hz	3 chan.: gaze direction, pupil dilation, blinks	SMI ETG
EEG	128 Hz	14 channels	Emotiv Epoc
GSR and BVP	512 Hz	2 chan.: skin conductance, blood volume pulse	Procomp Infiniti
Posture	500 Hz	16 channels: pressure, distance	2×Phidget 8/8/8
Simulator events	50 Hz	10+ channels	C# interface
Human annotations	125 Hz	1 channel: key logging	Keyboard

Post hoc Timestamp Correction

- Sample reporting event t_e is timestamped
  - at node n before transmission, $c_n(t_e) = t_e + \Delta_n(t_e)$, and
  - at server s incl. network delay, $c_s(t_e^{r_n}) = t_e + \Delta_{n,s}(t_e) + \Delta_s(t_e + \Delta_{n,s}(t_e))$.
- Use the TTP algorithm [3] to estimate actual time of event i
  - Use round-trip times to server d_j

$$T^n_{est}(i) = c_s(i) - \frac{1}{i}\sum_{j=1}^{i} c_s(j) + \frac{1}{i}\sum_{j=1}^{i} c_n(j) + \frac{1}{i}\sum_{j=1}^{i} d_j$$

- Experimental evaluation

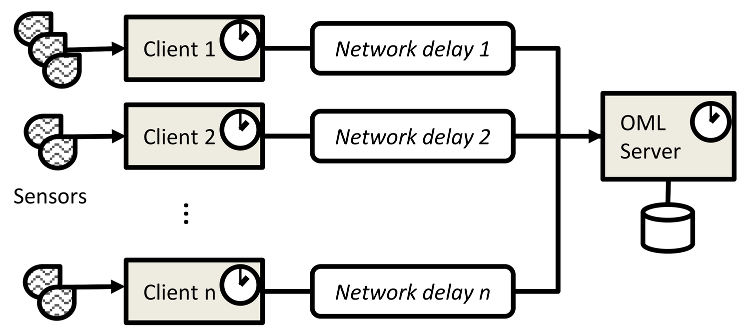


Experiment setup: known artificial clock offset and network delays are introduced, for later correction.

- 2 measurement nodes
- Event: simultaneous keypress via modified hardware
- Introduce artificial clock offset and network delays
  - up to 1 s offset
  - up to 2 s RTT
  - both static and variable
  - alone or combined
- Measure RTT and report RTT from each node once per second
- Compare timestamp differences for the same event

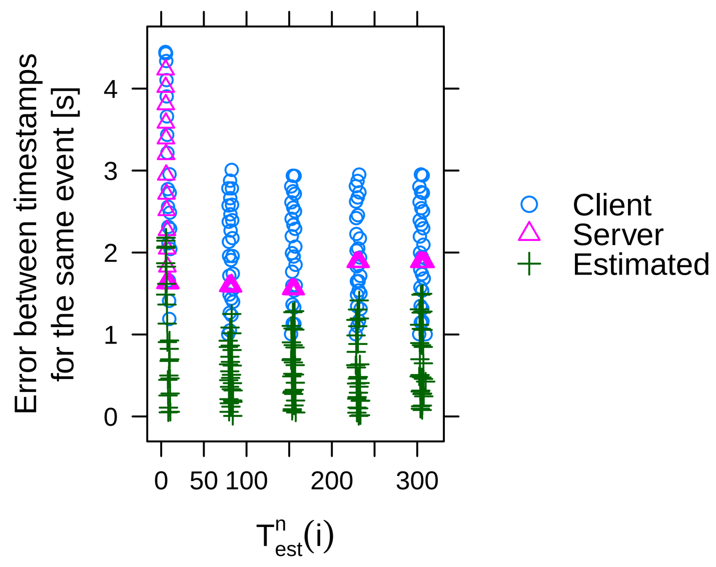
Physiological experiments framework

- Experiments with physiological sensors
  - Sample rates can be high ⇒ large volumes of data
  - Sampling rates vary across time scales ⇒ unaligned records
  - Sensors connected to different machines ⇒ synchronisation issues
  - Little support for cross-sensor synchronisation
- Framework for storing and synchronising physiological data from various equipments
  - Store high volumes of data in central location
  - Scalable to allow addition of new sensors
  - Low cost and easily deployable
    ⇒ The OML measurement framework [2] is a good candidate
  - Synchronisation mechanisms
    ⇒ Problem!
- Synchronisation is hard
  - clock discrepancies
  - network latencies
  - expensive solutions (*e.g.*, GPS)

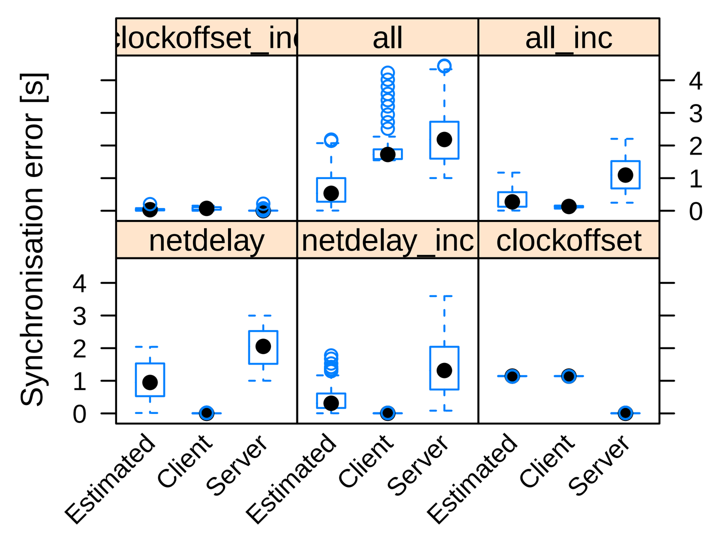


- Goal: Find a *post hoc* algorithm to resynchronise sample timestamps.

Results and future work



Pairwise comparisons of timestamps for a give event (static clock offset and network delay).



Errors in raw and estimated timestamps.

- Summary
  - Reduction of the estimated timestamp from the worst vantage point
  - Still not as good as the best vantage point (but which one is it?)
- Future work
  - Improve TPP feed-forward approach further
  - Study other synchronisation techniques
  - Integrate into OML

References

[1] R. Taib, J. Tederry, and B. Itzstein, "Quantifying driver frustration to improve road safety," in CHI 2014, Toronto, Ontario, Canada, Apr. 26, 2014. DOI: 10.1145/2559206.2581258.

[2] O. Mehani, G. Jourjon, T. Rakotoarivelo, and M. Ott, "An instrumentation framework for the critical task of measurement collection in the future Internet," Comput. Netw., vol. 63, Apr. 22, 2014. DOI: 10.1016/j.bjp.2014.01.007.

[3] K. Arvind, "Probabilistic clock synchronization in distributed systems," IEEE Trans. Parallel Distrib. Syst., vol. 5, no. 5, May 1994. DOI: 10.1109/71.282558.

From imagination to impact