EdgeDroid:

An Experimental Approach to Benchmarking Human-in-the-Loop Applications

M. Olguín Muñoz[†], J. Wang[‡], M. Satyanarayanan[‡] and J. Gross[†]

[†] KTH Royal Institute of Technology, Sweden
[‡] Carnegie Mellon University

Spring 2019 OEC Workshop. May 9–10, Carnegie Mellon University, Pittsburgh, PA.

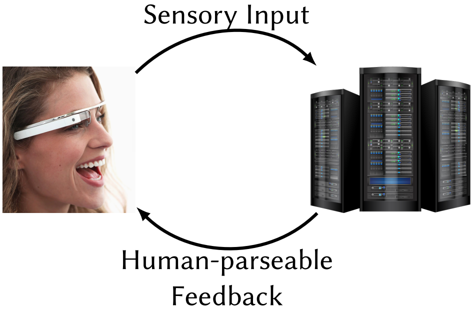

65
126ft
NE
354
73°
65
5
68°


SAMSUNG
Drowzee


VIVA TECHNOLOGY
bpifrance
bpifrance


65
126ft
NE
354


SAMSUNG

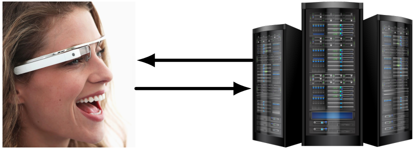

Sensory Input
Human-parseable
Feedback

Studying Human-in-the-Loop Applications

Need to understand and optimize these applications:

- How do they interact with each other?
- How do they interact with infrastructure?
- How do they scale?

With which methodology can we study these behaviors?

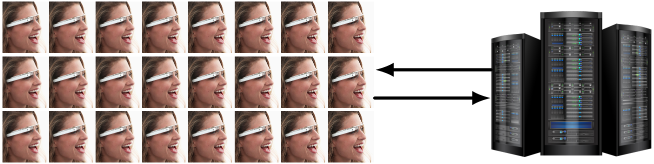


Studying Human-in-the-Loop Applications

Need to understand and optimize these applications:

- How do they interact with each other?
- How do they interact with infrastructure?
- How do they scale?

With which methodology can we study these behaviors?

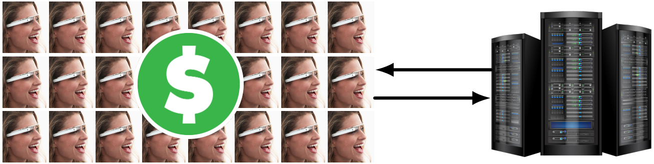


Studying Human-in-the-Loop Applications

Need to understand and optimize these applications:

- How do they interact with each other?
- How do they interact with infrastructure?
- How do they scale?

With which methodology can we study these behaviors?



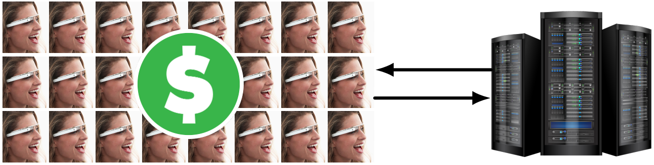
Costly

Studying Human-in-the-Loop Applications

Need to understand and optimize these applications:

- How do they interact with each other?
- How do they interact with infrastructure?
- How do they scale?

With which methodology can we study these behaviors?



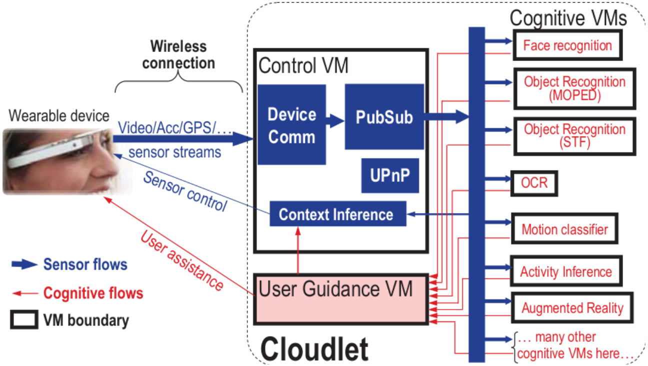
Costly, poor repeatability

Previous & Related Work

Prototyping [1]–[4]

Latencies [5]–[7]

Models [8], [9]



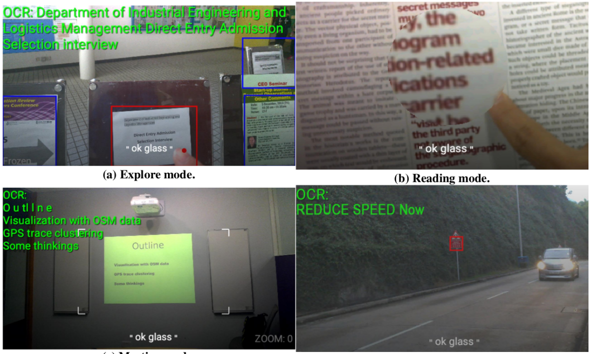
Gabriel Platform (CMU)

Previous & Related Work

Prototyping [1]–[4]

Latencies [5]–[7]

Models [8], [9]



(a) Explore mode. (b) Reading mode.

(c) Meeting mode. (d) Driving mode.

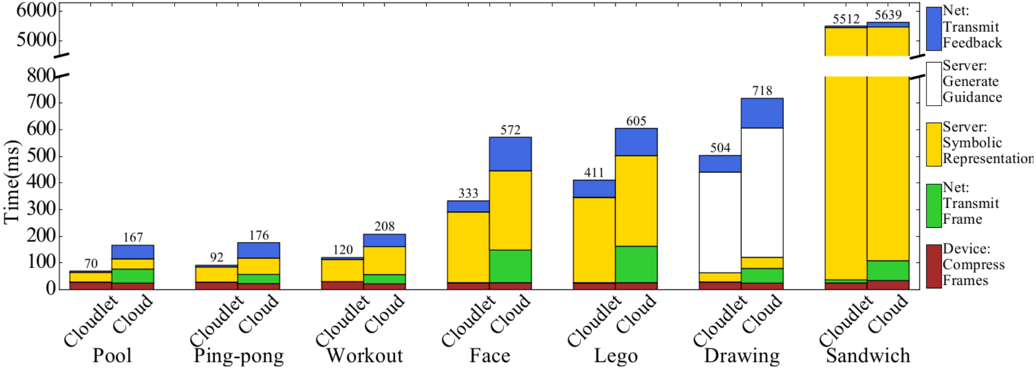
Hyperion Assistant (HKUST)

Previous & Related Work

Prototyping
[1]–[4]

Latencies
[5]–[7]

Models
[8], [9]



Source: Chen *et al.* [5]

Previous & Related Work

Prototyping
[1]–[4]

Latencies
[5]–[7]

Models
[8], [9]

- H. Al-Zubaidy *et al.*, “Performance of in-network processing for visual analysis in wireless sensor networks,” in *Proceedings of the IFIP Networking Conference*, ser. IFIP NETWORKING’15, 2015
- S. Schiessl *et al.*, “Finite-length coding in edge computing scenarios,” in *Proceedings of the International Workshop on Smart Antennas*, ser. ITG WSA ’17, 2017

Our Contributions

- A methodology for benchmarking human-in-the-loop applications.

Our Contributions

- A methodology for benchmarking human-in-the-loop applications.
- EdgeDroid: A synthetic workload generator for Human-in-the-Loop applications on Edge infrastructure.

Our Contributions

- A methodology for benchmarking human-in-the-loop applications.
- EdgeDroid: A synthetic workload generator for Human-in-the-Loop applications on Edge infrastructure.
- Experiments and measurements which show the effectiveness of the approach.

Benchmarking… what?

Key question:

In what way is system performance reflected in terms of user experience?

Benchmarking… what?

Key question:

In what way is system performance reflected in terms of user experience?

Answer: Time

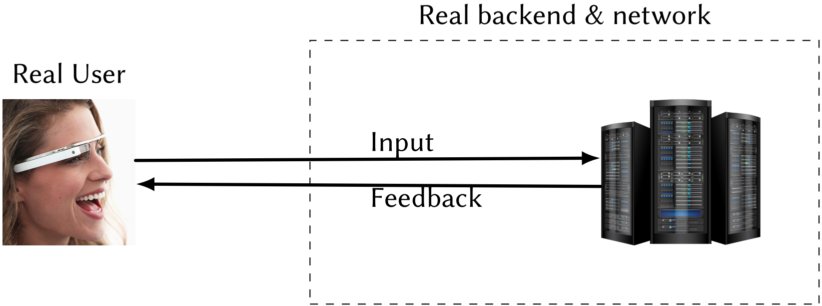
- Latency
- Jitter

Human time-perception is tail-driven…

Users tend to remember worst-case occurrences rather than averages.

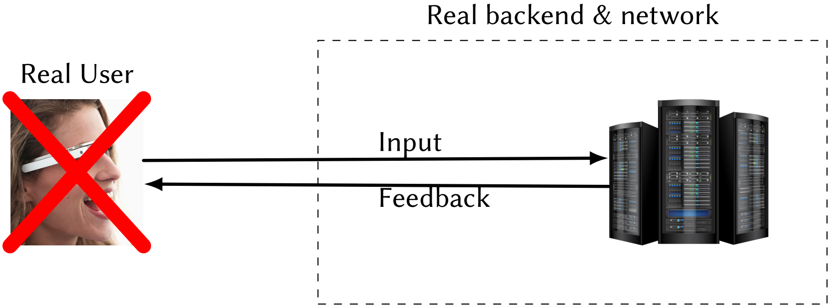
KPIs: Delays (total RTT and segment-wise)

Approach



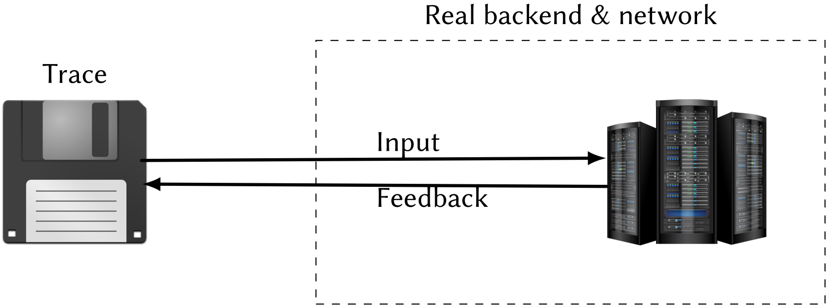
Benchmarking human-in-the-loop applications is HARD

Approach



What if we could do away with the human users?

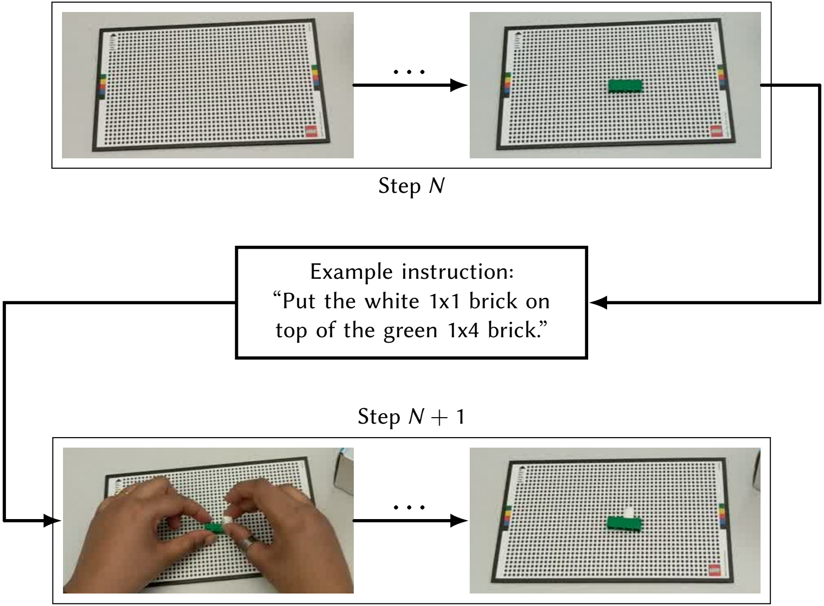
Approach



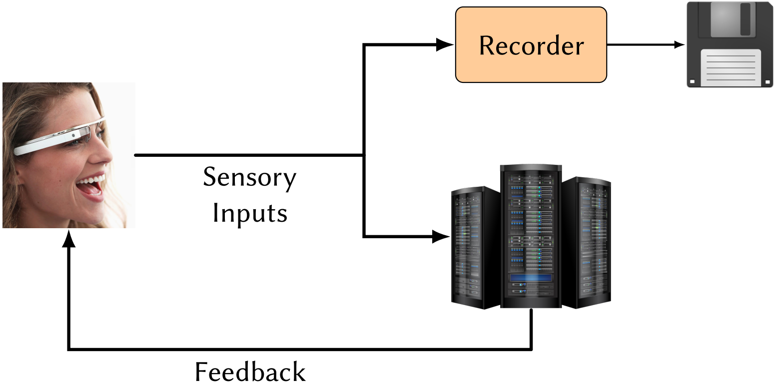
What if we could do away with the human users?

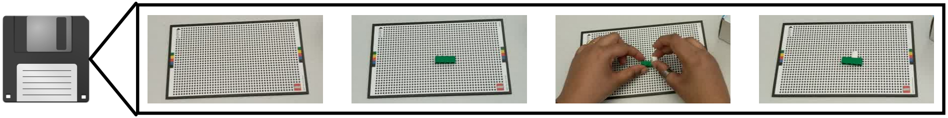
Repeatable, but is it scalable?

Example: Task Guidance Wearable Cognitive Assitance, LEGO [1]



Tracing





Trace Replay

Non-trivial Challenge

- Changes in system responsiveness require adapting trace.
- System delays affect user behavior as well.

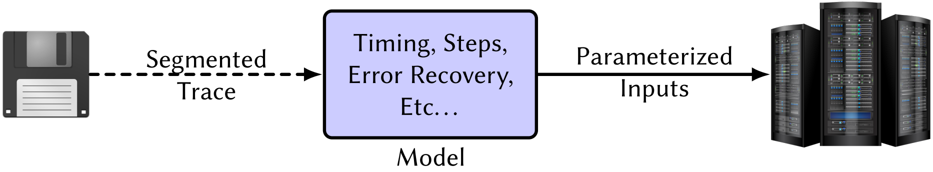
Trace Replay

Non-trivial Challenge

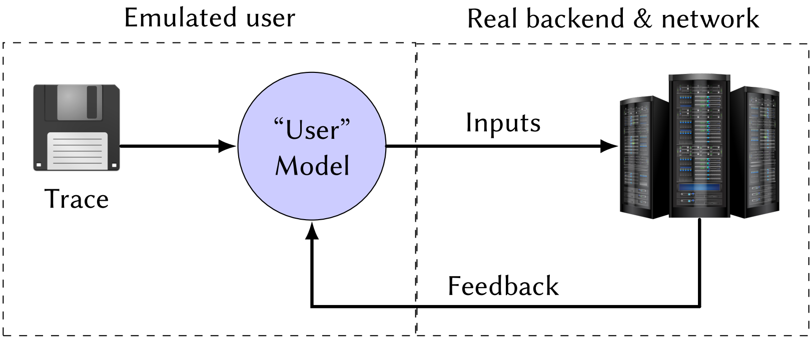
- Changes in system responsiveness require adapting trace.
- System delays affect user behavior as well.

Our Approach

- Segment trace into logical “steps”.
- Controlled replay of steps.

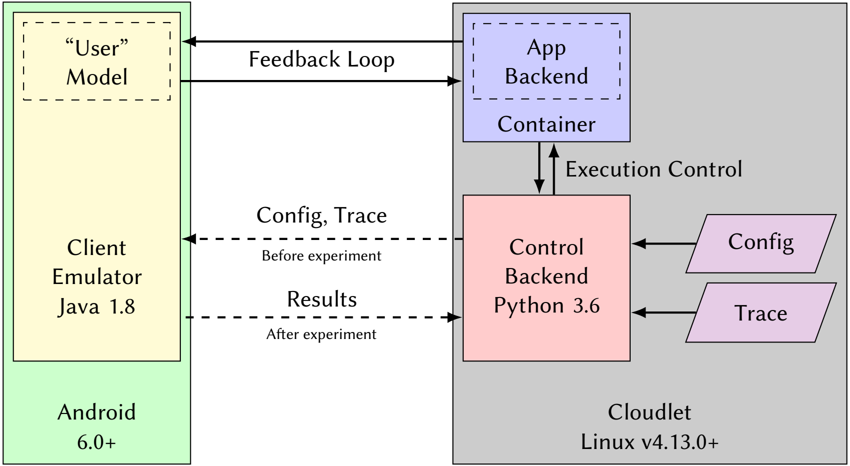


Approach



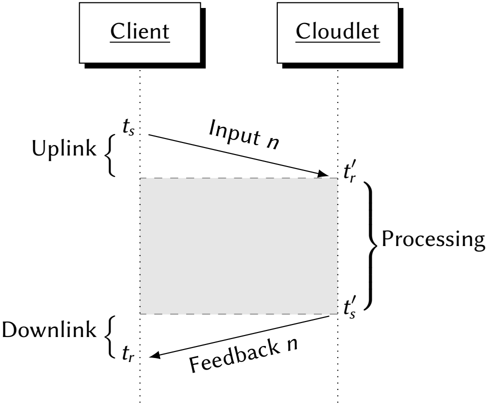
Repeatable, scalable!

Implementation: EdgeDroid



https://github.com/molguin92/EdgeDroid

Timestamping



Clocks are synchronized previous to the experiment.

Timestamps at key points to obtain:

$$\Delta T_{\text{up}} = t'_r - t_s \quad (1)$$

$$\Delta T_{\text{proc}} = t'_s - t'_r \quad (2)$$

$$\Delta T_{\text{down}} = t_r - t'_s \quad (3)$$

$$\Delta T_{\text{total}} = \Delta T_{\text{up}} + \Delta T_{\text{proc}} + \Delta T_{\text{down}} = t_r - t_s \quad (4)$$

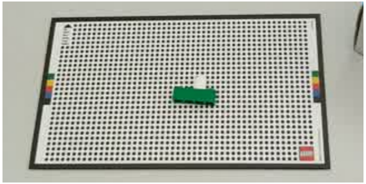
Evaluation

Key purpose:

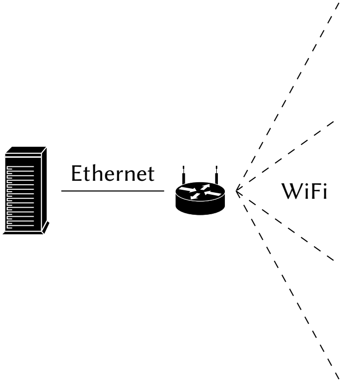
Demonstrate utility of EdgeDroid.

Evaluation: Setup

Application & Scenarios



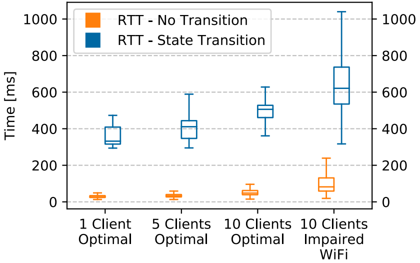
LEGO Assistant



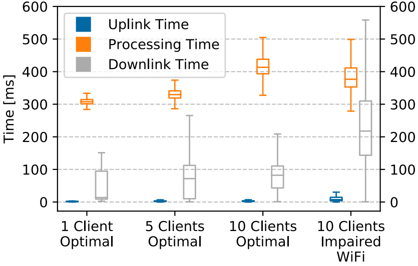
- Three *optimal* scenarios with 1, 5 and 10 devices.
- Weakened wireless link with 10 devices.

Use Cases

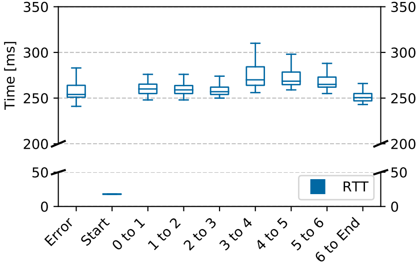
State change vs. no state change.



Times by pipeline segments.



RTT by task step.



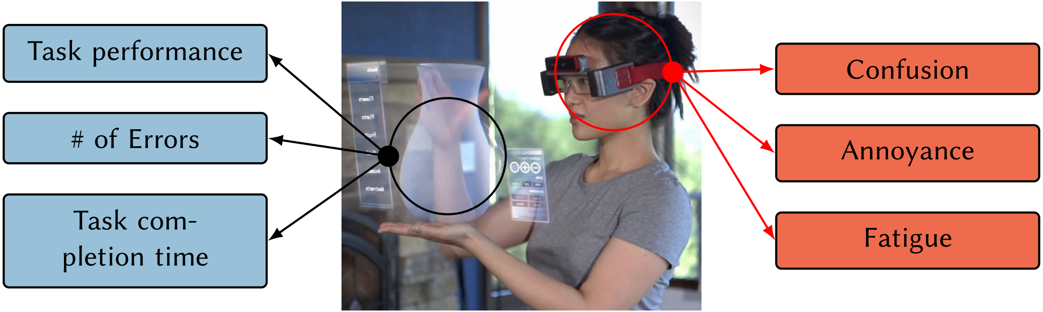
Future Work

- Extending EdgeDroid
  - Characterizing human behavior
  - Extending to other types of applications
- Characterizing Control Performance on Edge Computing Infrastructure

Characterizing human behavior…

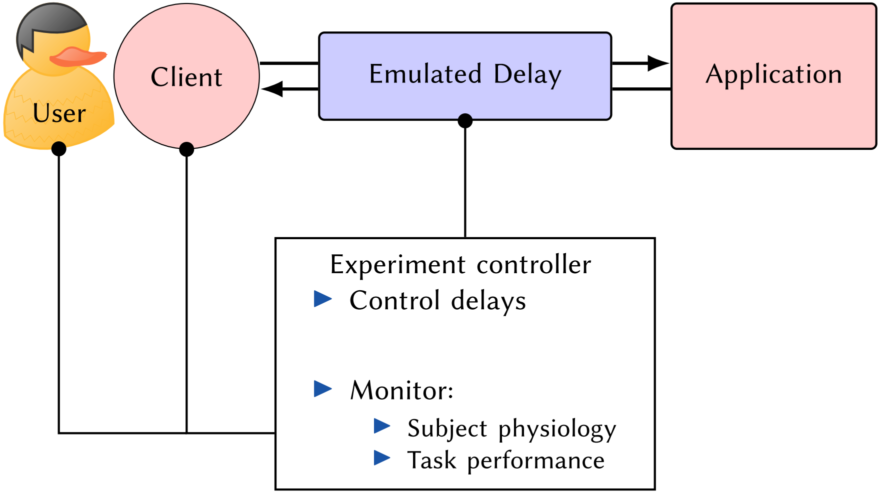
… in the presence of delays in human-in-the-loop applications.

Open research question: how do these delays affect users?



Dabrowski and Munson, "40 Years of Searching for the Best Computer System Response Time" [10]

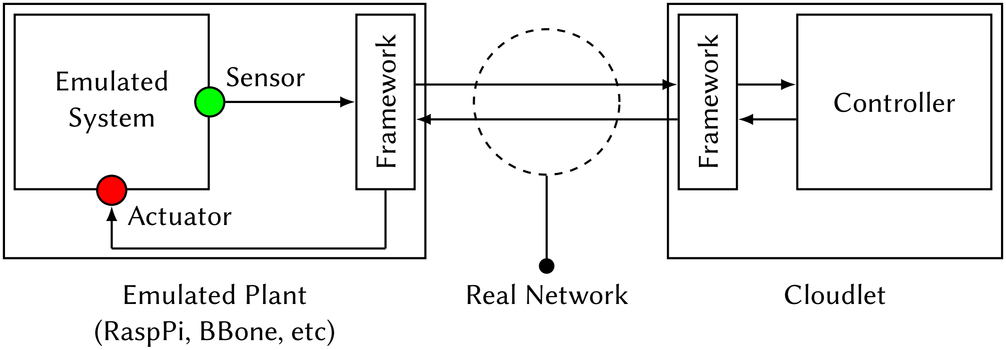
Proposed Experiment Setup



Control Performance on Edge Computing

EdgeDroid for Control Applications

Main focus: study achievable latencies and reliability with off-the-shelf infrastructure. Similar to the NCSBench platform developed at TUM[1][11].



[1] https://github.com/tum-lkn/NCSbench/

Conclusions

Summary

- Need to study the scaling of Human-in-the-Loop applications.
  - Difficult due to human users.
- Methodology + tool suite for benchmarking:
  - **EdgeDroid**
  - Trace based.
  - Model of human behavior.
- Results which show the utility of EdgeDroid.

Future Work

- Extending EdgeDroid
  - Characterizing human behavior
  - Extending to other types of applications
- Characterizing Control Performance on Edge Computing Infrastructure

Acknowledgements

Part of an ongoing collaboration with the Elijah Group at Carnegie Mellon University, led by Prof. Mahadev Satyanarayanan.

Carnegie Mellon University
Computer Science Department

This work was recently presented at HotMobile'19 [12].

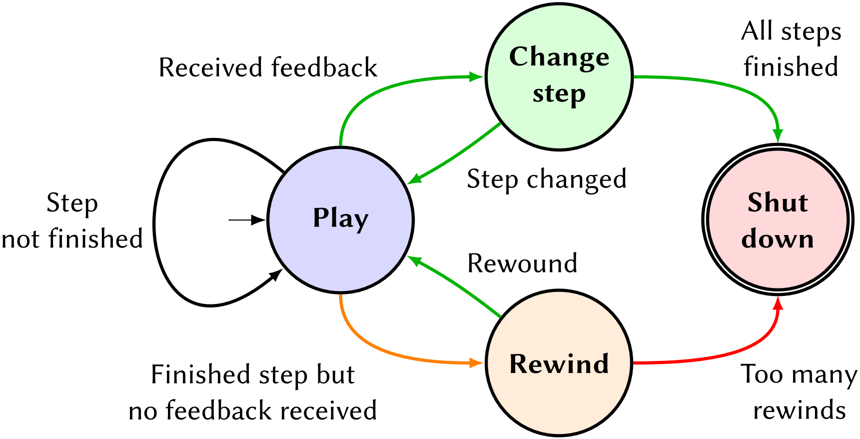
Extra Slides

Requirements

- Generate realistic, high-dimensional, real-time inputs.
- Correctly and realistically react to feedback.
- KPI: Delays.

Trace of pre-recorded inputs & a model of user behavior

User Model



Future work: more elaborate models.

References I

[1] K. Ha *et al.*, "Towards wearable cognitive assistance," in *Proceedings of the 12th Annual International Conference on Mobile Systems, Applications, and Services*, ser. MobiSys '14, Bretton Woods, New Hampshire, USA: ACM, 2014, pp. 68–81, ISBN: 978-1-4503-2793-0. DOI: 10.1145/2594368.2594383. [Online]. Available: http://doi.acm.org/10.1145/2594368.2594383.

[2] Z. Chen *et al.*, "Early implementation experience with wearable cognitive assistance applications," in *Proceedings of the 2015 Workshop on Wearable Systems and Applications*, ser. WearSys '15, Florence, Italy: ACM, 2015, pp. 33–38, ISBN: 978-1-4503-3500-3. DOI: 10.1145/2753509.2753517. [Online]. Available: http://doi.acm.org/10.1145/2753509.2753517.

[3] D. Chatzopoulos *et al.*, "Hyperion: A wearable augmented reality system for text extraction and manipulation in the air," in *Proceedings of the 8th ACM on Multimedia Systems Conference*, ser. MMSys'17, Taipei, Taiwan: ACM, 2017, pp. 284–295, ISBN: 978-1-4503-5002-0. DOI: 10.1145/3083187.3084017. [Online]. Available: http://doi.acm.org/10.1145/3083187.3084017.

[4] S. Jalaliniya *et al.*, "Designing wearable personal assistants for surgeons: An egocentric approach," *IEEE Pervasive Computing*, vol. 14, no. 3, pp. 22–31, 2015, ISSN: 1536-1268. DOI: 10.1109/MPRV.2015.61.

References II

[5] Z. Chen *et al.*, "An empirical study of latency in an emerging class of edge computing applications for wearable cognitive assistance," in *Proceedings of the Second ACM/IEEE Symposium on Edge Computing*, ser. SEC '17, San Jose, California: ACM, 2017, 14:1–14:14, ISBN: 978-1-4503-5087-7. DOI: 10.1145/3132211.3134458. [Online]. Available: http://doi.acm.org/10.1145/3132211.3134458.

[6] J. Dolezal *et al.*, "Performance evaluation of computation offloading from mobile device to the edge of mobile network," in *2016 IEEE Conference on Standards for Communications and Networking (CSCN)*, 2016, pp. 1–7. DOI: 10.1109/CSCN.2016.7785153.

[7] D. Chatzopoulos *et al.*, "Mobile augmented reality survey: From where we are to where we go," *IEEE Access*, vol. 5, pp. 6917–6950, 2017, ISSN: 2169-3536. DOI: 10.1109/ACCESS.2017.2698164.

[8] H. Al-Zubaidy *et al.*, "Performance of in-network processing for visual analysis in wireless sensor networks," in *Proceedings of the IFIP Networking Conference*, ser. IFIP NETWORKING'15, 2015.

[9] S. Schiessl *et al.*, "Finite-length coding in edge computing scenarios," in *Proceedings of the International Workshop on Smart Antennas*, ser. ITG WSA '17, 2017.

References III

[10] J. Dabrowski *et al.*, “40 years of searching for the best computer system response time,” *Interact. Comput.*, vol. 23, no. 5, pp. 555–564, Sep. 2011, ISSN: 0953-5438. DOI: 10.1016/j.intcom.2011.05.008. [Online]. Available: https://doi.org/10.1016/j.intcom.2011.05.008.

[11] S. Gallenmüller *et al.*, “Benchmarking networked control systems,” in *CPSBench2018 (CPSWeek 2018)*, Porto, Portugal, 2018.

[12] M. O. J. Olguín Muñoz *et al.*, “EdgeDroid: An Experimental Approach to Benchmarking Human-in-the-Loop Applications,” in *Proceedings of the 20th International Workshop on Mobile Computing Systems and Applications*, ser. HotMobile ’19, Santa Cruz, CA, USA: ACM, 2019, pp. 93–98, ISBN: 978-1-4503-6273-3. DOI: 10.1145/3301293.3302353. [Online]. Available: http://doi.acm.org/10.1145/3301293.3302353.

[13] M. Satyanarayanan *et al.*, “The case for VM-based cloudlets in mobile computing,” *IEEE Pervasive Computing*, vol. 8, no. 4, 2009.

[14] J. Flinn, “Cyber foraging: Bridging mobile and cloud computing,” *Synthesis Lectures on Mobile and Pervasive Computing*, vol. 7, no. 2, pp. 1–103, 2012.

[15] K. Sasaki *et al.*, “Vehicle control system coordinated between cloud and mobile edge computing,” in *2016 55th Annual Conference of the Society of Instrument and Control Engineers of Japan (SICE)*, 2016, pp. 1122–1127. DOI: 10.1109/SICE.2016.7749210.

References IV

[16] ——,“Layered vehicle control system coordinated between multiple edge servers,” in *2017 IEEE Conference on Network Softwarization (NetSoft)*, 2017, pp. 1–5. DOI: 10.1109/NETSOFT.2017.8004199.

[17] T. Bittmann, “The edge will eat the cloud,” *Gartner Research*, no. G00338633, 2017.

[18] K. Kumar *et al.*, “Cloud computing for mobile users: Can offloading computation save energy?” *IEEE Computer*, vol. 43, no. 4, pp. 51–56, 2010.

[19] E. Cuervo *et al.*, “Maui: Making smartphones last longer with code offload,” in *Proceedings of the International Conference on Mobile Systems, Applications, and Services*, ser. ACM MOBISYS'10, 2010.

[20] K. Ha *et al.*, “The impact of mobile multimedia applications on data center consolidation,” in *2013 IEEE International Conference on Cloud Engineering (IC2E)*, 2013, pp. 166–176. DOI: 10.1109/IC2E.2013.17.

[21] K. Ha *et al.*, “Just-in-time provisioning for cyber foraging,” in *Proceeding of the 11th Annual International Conference on Mobile Systems, Applications, and Services*, ser. MobiSys '13, Taipei, Taiwan: ACM, 2013, pp. 153–166, ISBN: 978-1-4503-1672-9. DOI: 10.1145/2462456.2464451. [Online]. Available: http://doi.acm.org/10.1145/2462456.2464451.

References V

[22] (2018). Docker, [Online; accessed 14. Aug. 2018], [Online]. Available: https://www.docker.com.

[23] (2018). Network Time Protocol, [Online; accessed 24. Sep. 2018], [Online]. Available: https://www.eecis.udel.edu/~mills/ntp/html/index.html.

[24] (2018). TOML, [Online; accessed 25. Sep. 2018], [Online]. Available: https://github.com/toml-lang/toml1.

[25] K. Kim *et al.*, "Workload synthesis: Generating benchmark workloads from statistical execution profile," in *2014 IEEE International Symposium on Workload Characterization (IISWC)*, 2014, pp. 120–129. DOI: 10.1109/IISWC.2014.6983051.

[26] E. Deniz *et al.*, "Minime: Pattern-aware multicore benchmark synthesizer," *IEEE Transactions on Computers*, vol. 64, no. 8, pp. 2239–2252, 2015, ISSN: 0018-9340. DOI: 10.1109/TC.2014.2349522.

[27] M. Olguín *et al.*, "Demo: Scaling on the Edge – A Benchmarking Suite for Human-in-the-Loop Applications," in *Proceedings of The Third ACM/IEEE Symposium on Edge Computing*, ser. SEC '18, Accepted Submission, Extended Abstract, 2018. [Online]. Available: https://olguin.se/files/demo-scaling-edge.pdf.