Timeout Strategies for Distributed Systems

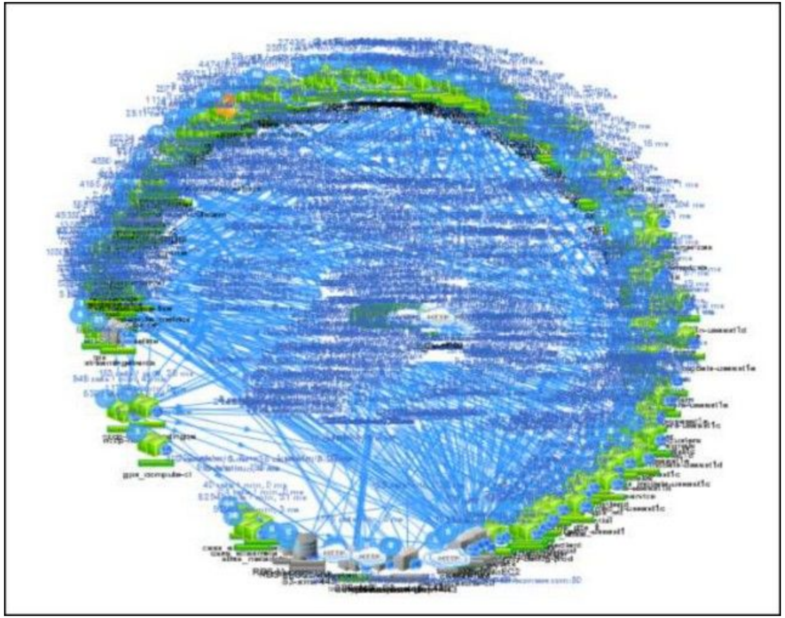
Govind Velamoor

Mentors: Lan (Max) Liu, Zhaoqi (Roy) Zhang,
Prof. Raja Sambasivan

MIT PRIMES Spring Conference, 5/18/2025

Distributed systems are powerful but complex

Netflix's distributed systems



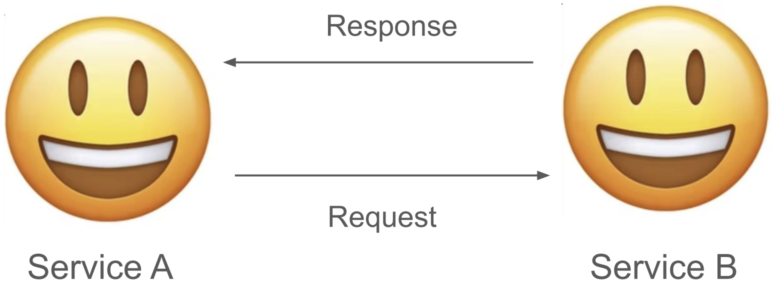
Benefits:

Scaling

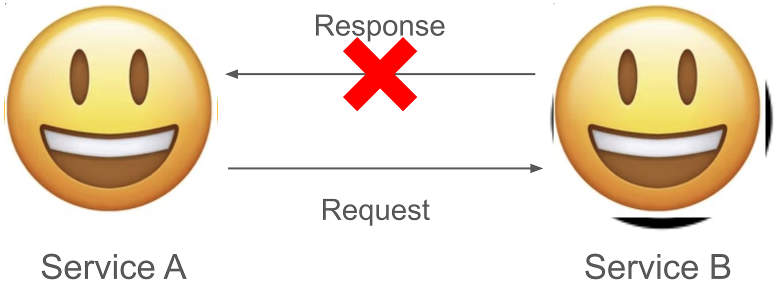
Performance

Challenge:
Debugging and failure-tolerance

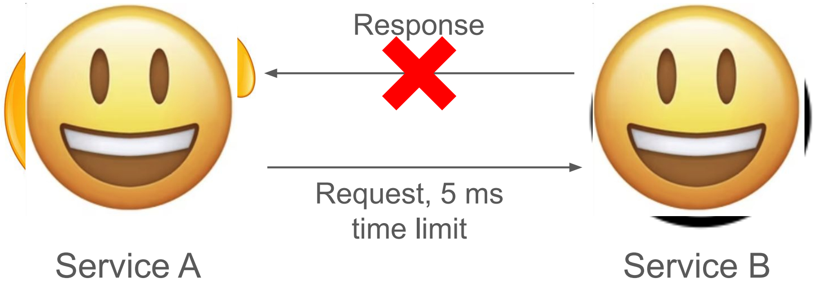
Why are timeouts important for distributed systems?



Normal Request



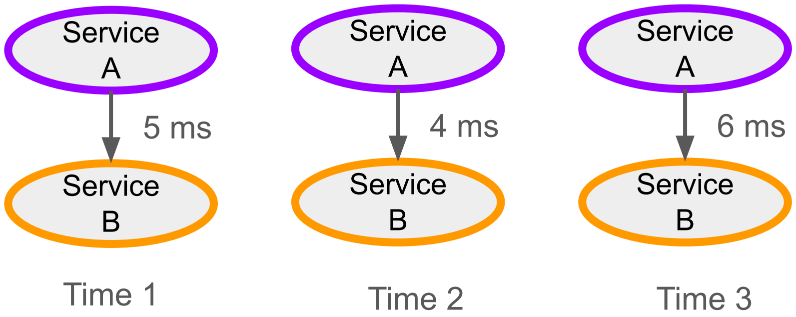
Failed Request



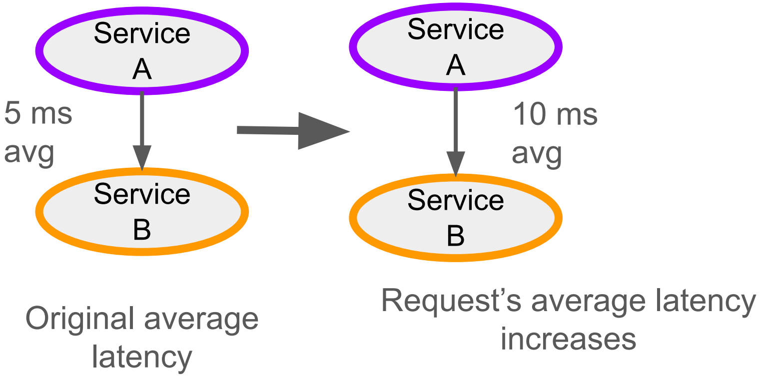
Request with a timeout

Challenges with setting timeouts

Nonconstant latency



Developers can change services' implementation



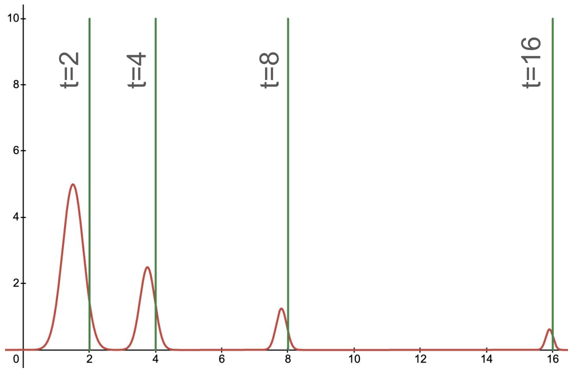
Other ways latency can change:

- Cache hit or miss.
- Systems can get overloaded.

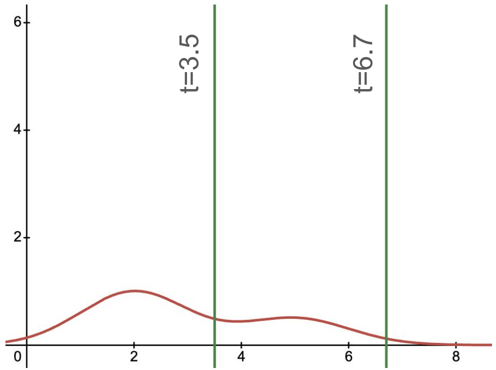
Optimal Timeouts

Previous work: Why ALTO is optimal

- **ALTO**: **A**nalyzer of **L**atency for **T**imeout **O**ptimization
- 40% faster than industry standard in worst-case
- Send “diagnostic” requests, those without timeout values, to monitor the real state of each service



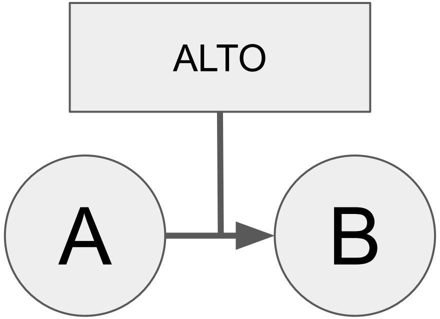
Distribution for which exponential backoff is optimal



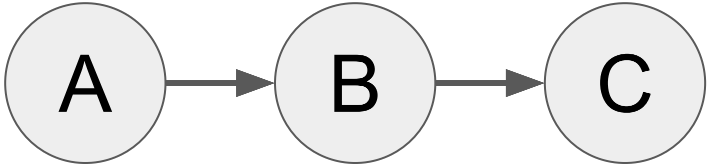
Arbitrary distribution and optimal timeouts computed by ALTO

Current work: Research Goals

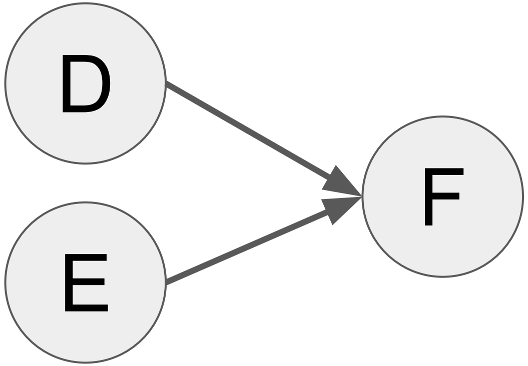
1. Determining optimal timeout values



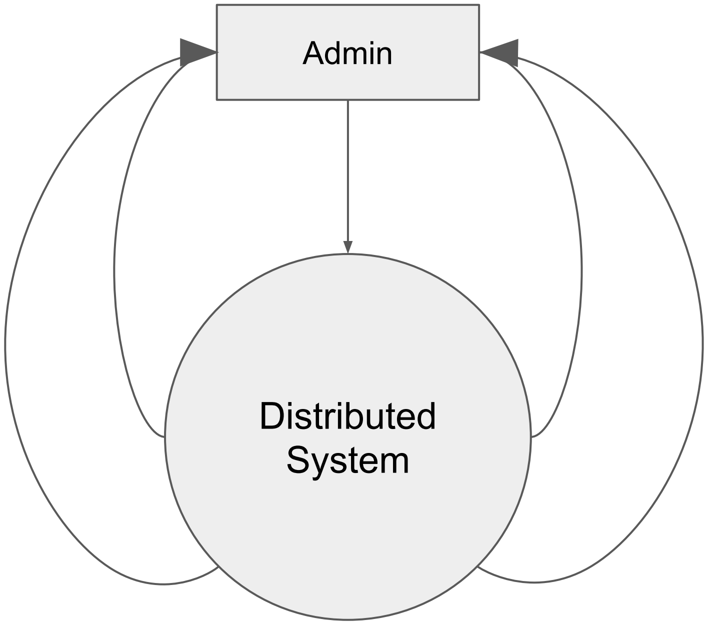
2. Fast timeout adaptability

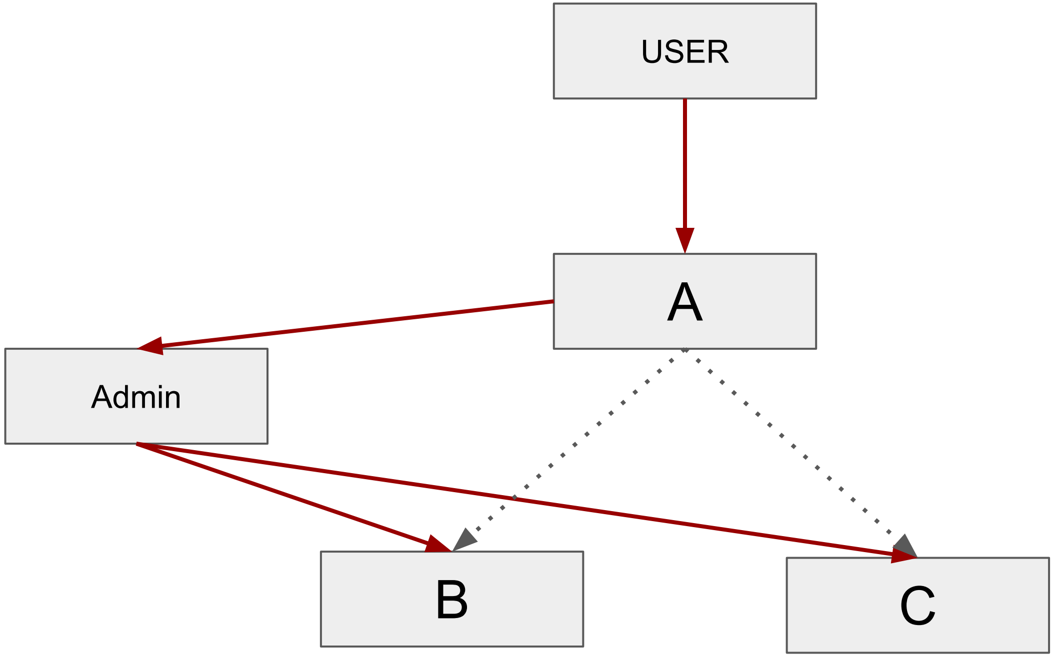


3. Efficiently computing timeout values across the entire distributed system



Global implementation of ALTO in our testbed

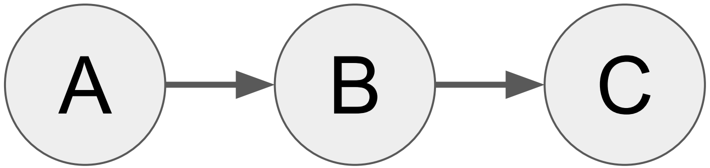


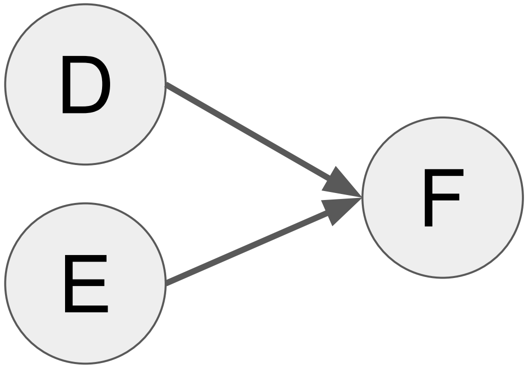


Computing timeouts globally & adapting to new situations

1. Find distributions from all services
2. Compute timeouts to each service based on distributions
3. Determine contribution from each service

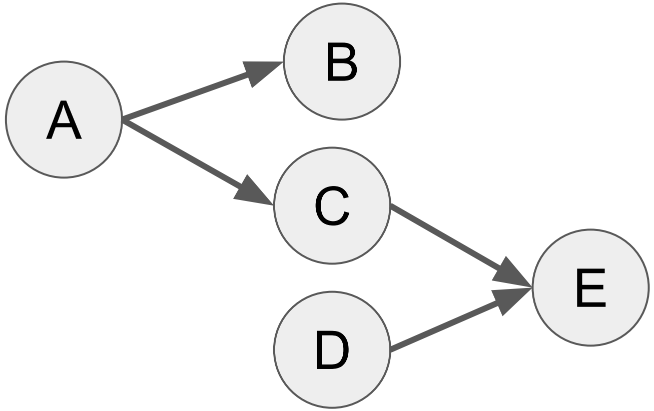
$$C(\mathrm{N}) = D(\mathrm{N}) - \sum_{n \in \mathrm{N.V}} D(n)$$





Proposed evaluation

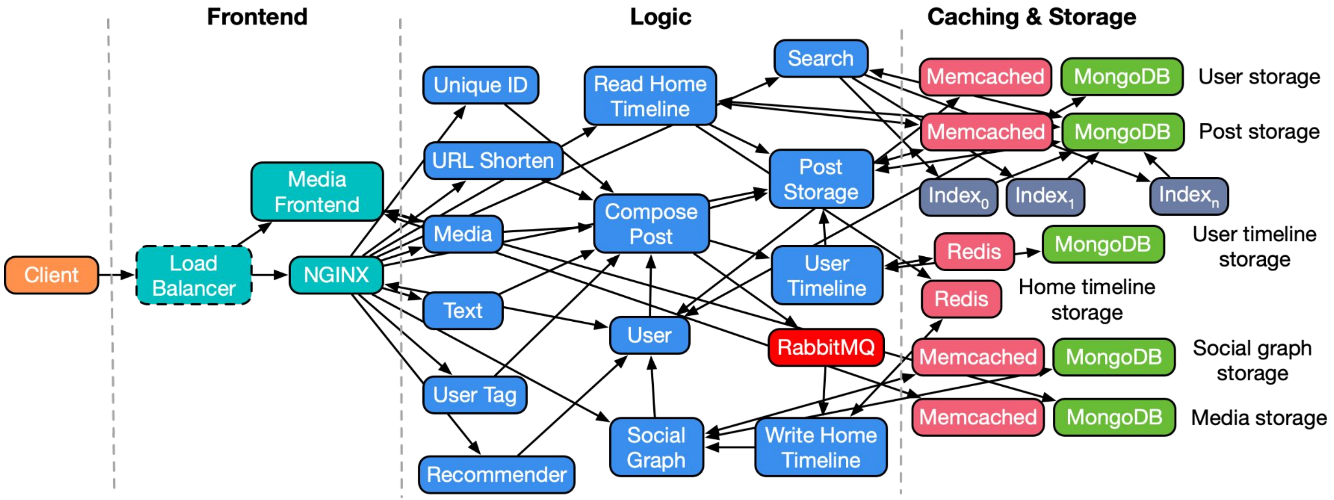
- Evaluate whether the global approach is faster-adapting and more resource-efficient than the local approach
- Consider:
  - The number of timed out requests (measure of lag in adapting timeout values)
  - The closeness of the timeout value to latency



Methodology
- Use my custom application
- Inject latency increases/decreases
- See how long services take to adapt

Future Work

- Implement ALTO global in Social Network, a larger and industry-standardized distributed system



Conclusions

Goal \ Timeout Algorithm	Exponential Backoff (Industry standard)	ALTO (previous work)	ALTO, Global (current work)
Optimal timeout values	✗	✓	✓
Efficient computation	✓	✗	✓
Fast adaptability	✗	✗	✓

Please reach out! **gvelamoor@gmail.com**

Acknowledgements

My mentors Max, Roy, and Prof. Sambasivan



My family and friends

MIT PRIMES

References

[1] Barroso L. A. Dean, J. The tail at scale. Communications of the ACM, 56:74–80, 2013.

[2] E. Troubitsyna. Model-driven engineering of fault tolerant microservices. Fourteenth Int. Conf. Internet Web Appl. Serv, 2019.

[3] Martinek P. Al-Debagy, O. A comparative review of microservices and monolithic architectures. 18th IEEE International Symposium on Computational Intelligence and Informatics, pages 000149–000154, 2019.

[4] Ojdowska A. Przybylek A. Blinowski, G. Model driven engineering of fault tolerant microservices. Fourteenth Int. Conf. Internet Web Appl. Serv, pages 1–6, 2019.

[5] Vishal Varshney Anton Ilinchik. All you need to know about timeouts: How to set a reasonable timeout for your microservices to achieve maximum performance and resilience. Zalando Engineering Blog, 2023.

[6] Tcp congestion control algorithms. https://www.tetcos.com/pdf/v13/Experiments/TCP-Congestion-Control-Algorithms.pdf

[7] B. Gregg. Frequency trails. https://www.brendangregg.com/FrequencyTrails/modes.html

[8] pyms. https://python-microservices.github.io/home/

[9] J. Richards. wrk2. https://github.com/giltene/wrk2

[10] The second law of latency: Latency distributions are never normal.

[11] Li Q. Yang, B. Enhanced particle swarm optimization algorithm for sea clutter parameter estimation in generalized pareto distribution. Appl. Sci., 2023.

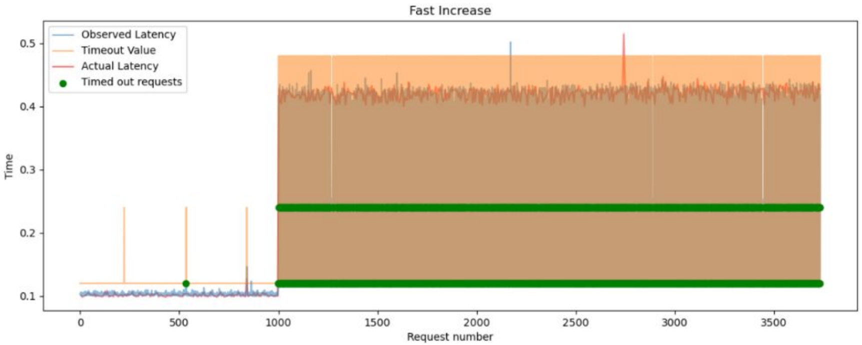
[12] He J. Zhang, W. Modeling end-to-end delay using pareto distribution. Second International Conference on Internet Monitoring and Protection (ICIMP 2007), 2007.

[13] Zhang Y. Cheng D. Shetty A. et al. Gan, Y. Death star bench repository.

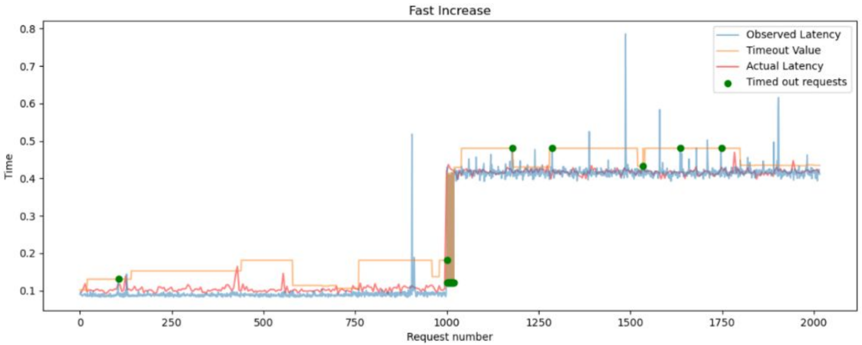
[14] Zhang Y. Cheng D. Shetty A. et al. Gan, Y. An open-source benchmark suite for microservices and their hardware-software implications for cloud edge systems. ASPLOS ’19: Proceedings of the Twenty-Fourth International Conference on Architectural Support for Programming Languages and Operating Systems, 2019.

Backup Slides

Our previous work, ALTO performs significantly better than industry standard



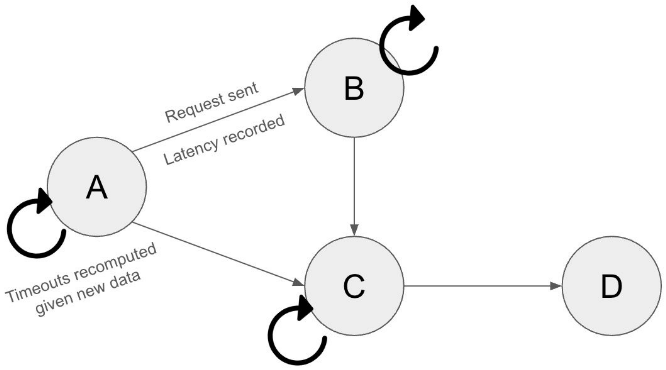
Exponential Backoff



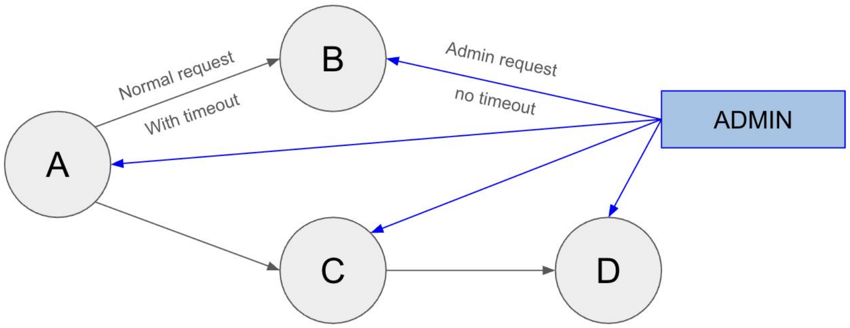
ALTO

Local vs Global timeout algorithms

Local algorithms operate strictly between two services



Global algorithms have data about the entire distributed system



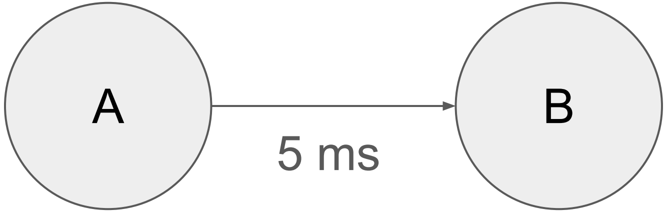
When is a timeout **optimal**?

- As systems evolve, timeouts change.

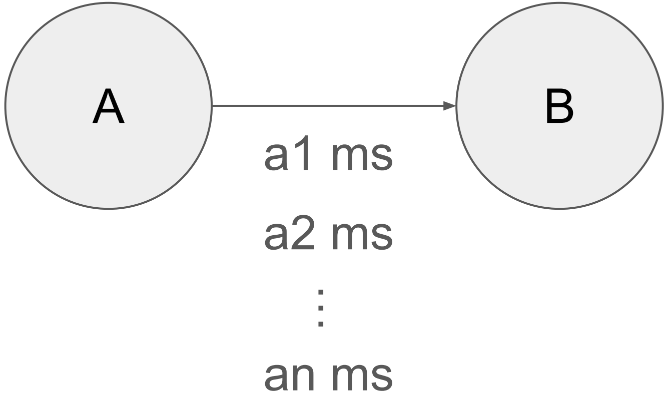
- An **optimal timeout** is a timeout that results in the minimal possible average amount of time before a response is received.
  - Too short → wasting work since we have to reissue requests
  - Too long → wasting time when request should have been discarded

- We **continuously update** the timeout values to adapt accordingly.

Increasing timeout values allow for precise hedging



Normal timeouts



Sequential timeouts

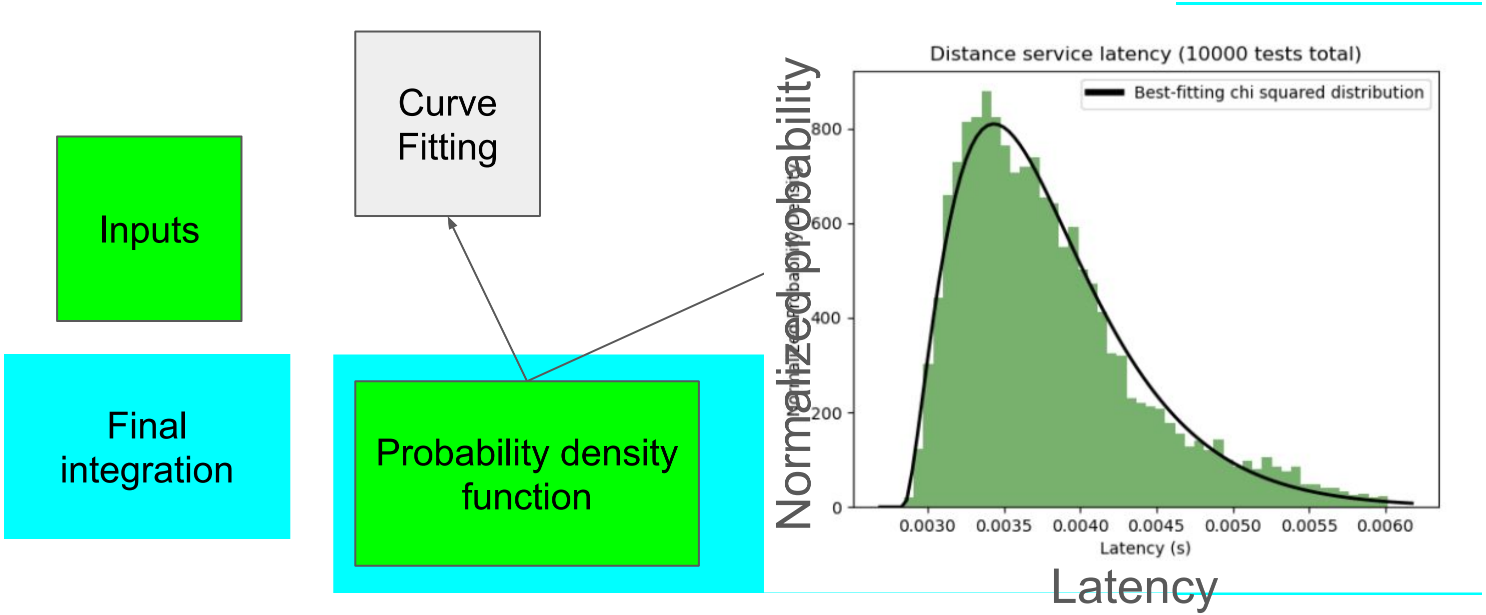
Failure conditions

- Temporary increase
- System failure

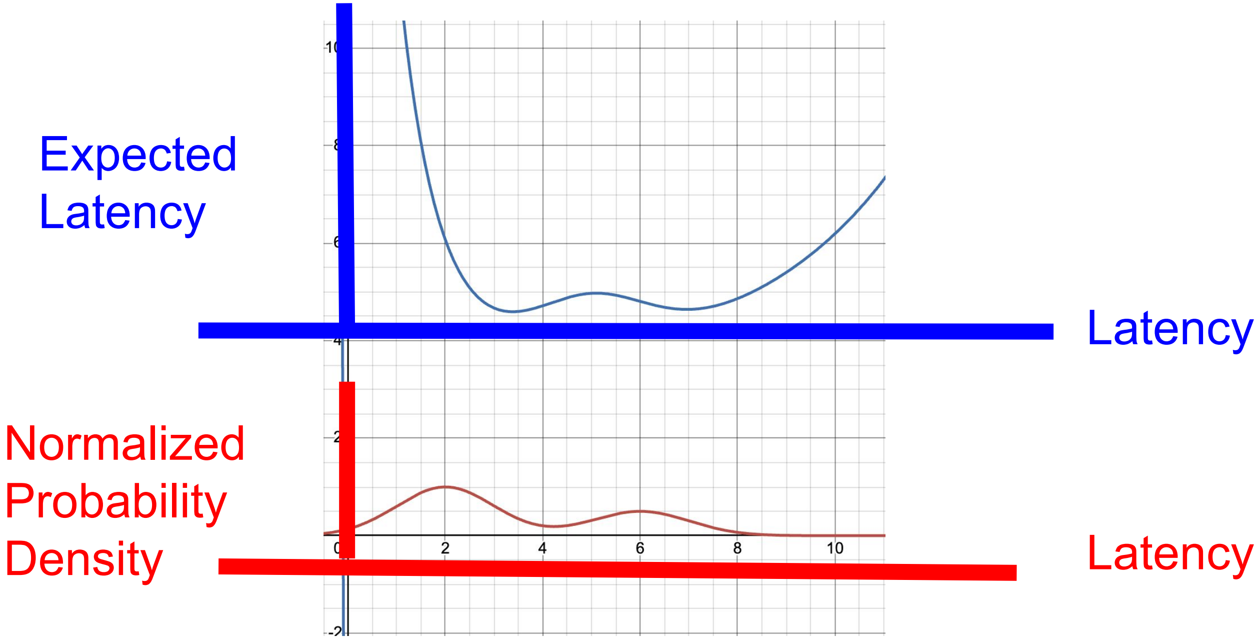
Increase timeout value

- Sensitivity reduction
- Failure confirmation

Mathematical Model for Latency



Math behind mathematical model (1)



Math behind mathematical model (2)

$$E(t) = \frac{\int_t^{\infty} f(x)\mathrm{d}x}{\int_0^{\infty} f(x)\mathrm{d}x}(t + E(t)) + \frac{\int_0^{t} f(x)\mathrm{d}x}{\int_0^{\infty} f(x)\mathrm{d}x} \cdot \frac{\int_0^{t} x f(x)\mathrm{d}x}{\int_0^{t} f(x)\mathrm{d}x} + g(t)$$

Case where we time out

Case where the request is successful

Cost function

Derivation

$$E(t) = \frac{\int_t^\infty f(x)\mathrm{d}x}{\int_0^\infty f(x)\mathrm{d}x}(t + E(t)) + \frac{\int_0^t f(x)\mathrm{d}x}{\int_0^\infty f(x)\mathrm{d}x} \cdot \frac{\int_0^t x f(x)\mathrm{d}x}{\int_0^t f(x)\mathrm{d}x} + g(t)$$

$$E(t) = \frac{\int_t^\infty f(x)\mathrm{d}x}{\int_0^\infty f(x)\mathrm{d}x} t + \frac{\int_t^\infty f(x)\mathrm{d}x}{\int_0^\infty f(x)\mathrm{d}x} E(t) + \frac{\int_0^t f(x)\mathrm{d}x}{\int_0^\infty f(x)\mathrm{d}x} \cdot \frac{\int_0^t x f(x)\mathrm{d}x}{\int_0^t f(x)\mathrm{d}x} + g(t)$$

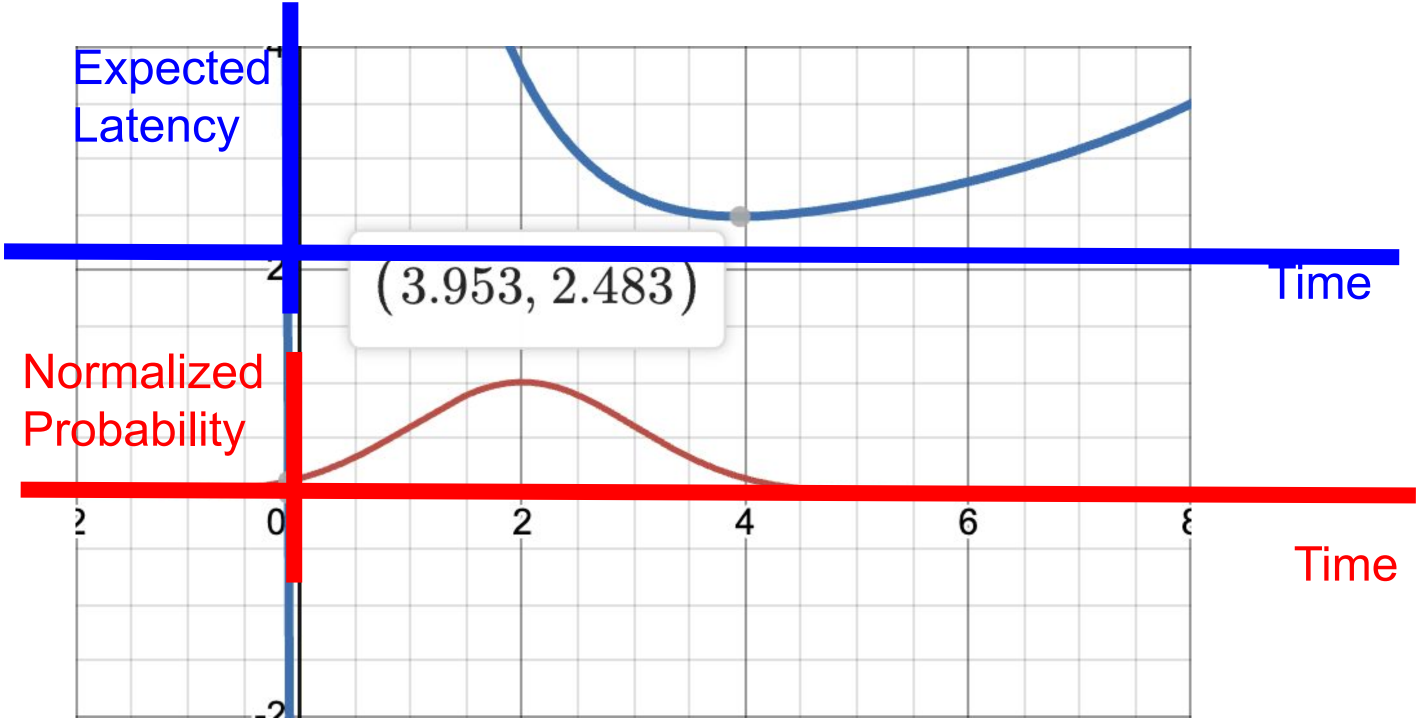
$$E(t)\left(1 - \frac{\int_t^\infty f(x)\mathrm{d}x}{\int_0^\infty f(x)\mathrm{d}x}\right) = \frac{\int_t^\infty f(x)\mathrm{d}x}{\int_0^\infty f(x)\mathrm{d}x} t + \frac{\int_0^t f(x)\mathrm{d}x}{\int_0^\infty f(x)\mathrm{d}x} \cdot \frac{\int_0^t x f(x)\mathrm{d}x}{\int_0^t f(x)\mathrm{d}x} + g(t)$$

$$E(t)\left(\frac{\int_0^t f(x)\mathrm{d}x}{\int_0^\infty f(x)\mathrm{d}x}\right) = \frac{\int_t^\infty f(x)\mathrm{d}x}{\int_0^\infty f(x)\mathrm{d}x} t + \frac{\int_0^t f(x)\mathrm{d}x}{\int_0^\infty f(x)\mathrm{d}x} \cdot \frac{\int_0^t x f(x)\mathrm{d}x}{\int_0^t f(x)\mathrm{d}x} + g(t)$$

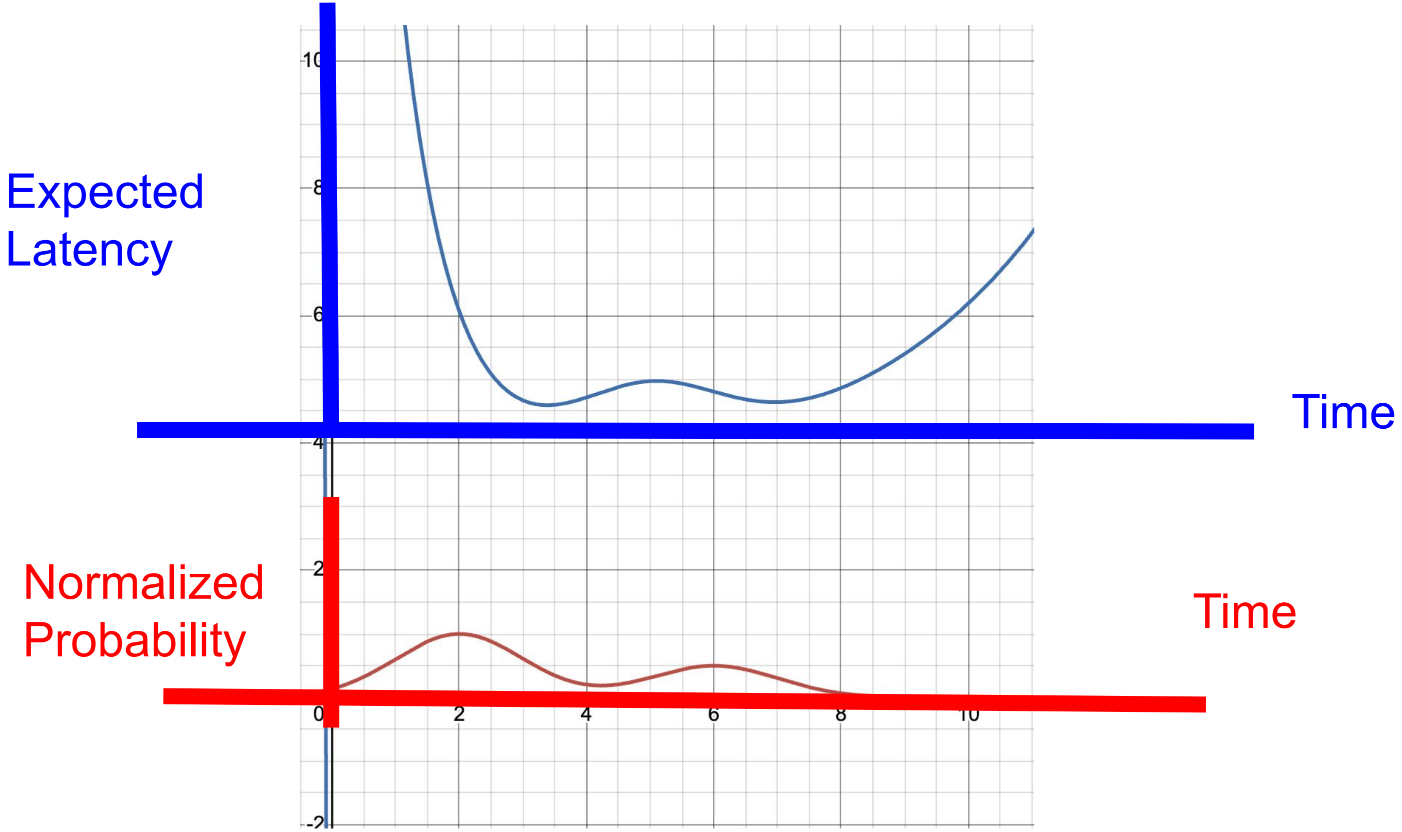
$$E(t)\int_0^t f(x)\mathrm{d}x = \int_t^\infty f(x)\mathrm{d}x \cdot t + \int_0^t f(x)\mathrm{d}x \cdot \frac{\int_0^t x f(x)\mathrm{d}x}{\int_0^t f(x)\mathrm{d}x} + g(t)\int_0^\infty f(x)\mathrm{d}x$$

$$E(t) = \frac{t\int_t^\infty f(x)\mathrm{d}x + \int_0^t x f(x)\mathrm{d}x + g(t)\int_0^\infty f(x)\mathrm{d}x}{\int_0^t f(x)\mathrm{d}x}$$

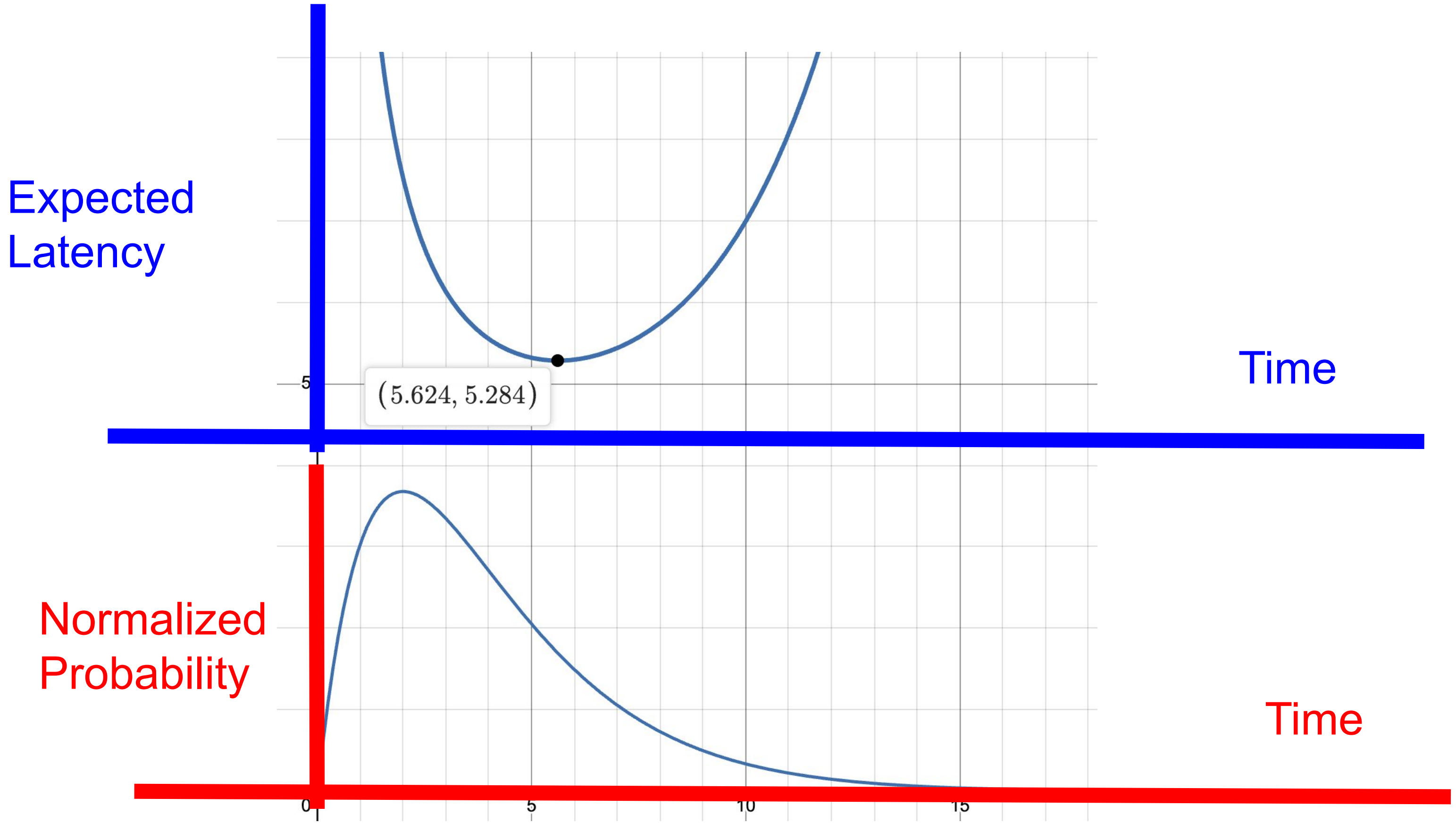
Performance ($f(x) = e^{-\frac{(x-2)^2}{2}}$, $g(x) = \frac{e^{\left(\frac{x}{3}\right)}}{10}$)



Performance ($f(x) = e^{-\frac{(x-2)^2}{2}} + \frac{e^{-\frac{(x-6)^2}{2}}}{2}$, $g(x) = \frac{e^{\left(\frac{x}{3}\right)}}{10}$)



Performance (chi squared, $g(x) = \frac{e^{\left(\frac{x}{3}\right)}}{10}$)



Extensions to a sequence of timeouts

Let's define a sequence of timeouts $\{t_n\}_{n\geq 0}$

We can modify the original equation in the following way:

$$E_n = \frac{\displaystyle\int_{t_n}^{\infty} f(x)\mathrm{d}x}{\displaystyle\int_{0}^{\infty} f(x)\mathrm{d}x}(t_n + E_{n+1}) + \frac{\displaystyle\int_{0}^{t_n} f(x)\mathrm{d}x}{\displaystyle\int_{0}^{\infty} f(x)\mathrm{d}x} \cdot \frac{\displaystyle\int_{0}^{t_n} xf(x)\mathrm{d}x}{\displaystyle\int_{0}^{t_n} f(x)\mathrm{d}x} + g(t_n)$$

From now on, let $E_n = a_n + b_n E_{n+1}$ where

$$a_n = \frac{\displaystyle\int_{0}^{t_n} xf(x)\mathrm{d}x + t_n \int_{t_n}^{\infty} f(x)\mathrm{d}x}{\displaystyle\int_{0}^{\infty} f(x)\mathrm{d}x} + g(t_n)$$

And

$$b_n = \frac{\displaystyle\int_{t_n}^{\infty} f(x)\mathrm{d}x}{\displaystyle\int_{0}^{\infty} f(x)\mathrm{d}x}$$

(these are both functions of only t_n)

(Derivation for a_n)

$$a_n = t_n \frac{\int_{t_n}^{\infty} f(x)\mathrm{d}x}{\int_0^{\infty} f(x)\mathrm{d}x} + \frac{\int_0^{t_n} f(x)\mathrm{d}x}{\int_0^{\infty} f(x)\mathrm{d}x} \cdot \frac{\int_0^{t_n} xf(x)\mathrm{d}x}{\int_0^{t_n} f(x)\mathrm{d}x} + g(t_n)$$

$$a_n = t_n \frac{\int_{t_n}^{\infty} f(x)\mathrm{d}x}{\int_0^{\infty} f(x)\mathrm{d}x} + \frac{\int_0^{t_n} xf(x)\mathrm{d}x}{\int_0^{\infty} f(x)\mathrm{d}x} + g(t_n)$$

$$a_n = \frac{\int_0^{t_n} xf(x)\mathrm{d}x + t_n \int_{t_n}^{\infty} f(x)\mathrm{d}x}{\int_0^{\infty} f(x)\mathrm{d}x} + g(t_n)$$

Extensions to a sequence of timeouts

The problem with the current equation is that it never ends.

$$E_n = a_n + b_n E_{n+1}$$

So, let's define a "terminal" timeout t_z : it's the last timeout in the list, and the timeout will never increase past it.

$$E_z = a_z + b_z E_z$$

$$E_z = \frac{a_z}{1 - b_z}$$

We now have the piecewise function $$E_n = \begin{cases} a_n + b_n E_{n+1} & n < z \\ \dfrac{a_z}{1 - b_z} & n \geq z \end{cases}$$

Extensions to a sequence of timeouts

How do we actually compute the timeouts?

E_0 is a function of every timeout, and we need to minimize it over $t_0, t_1, t_2, \cdots t_z$

$$E_0 = a_0 + b_0 \left(a_1 + b_1 \left(a_2 + b_2 \left(\cdots \frac{a_z}{1 - b_z} \cdots \right) \right) \right)$$

$$a_n = \frac{\int_0^{t_n} x f(x) \mathrm{d}x + t_n \int_{t_n}^{\infty} f(x) \mathrm{d}x}{\int_0^{\infty} f(x) \mathrm{d}x} + g(t_n)$$

$$b_n = \frac{\int_{t_n}^{\infty} f(x) \mathrm{d}x}{\int_0^{\infty} f(x) \mathrm{d}x}$$

Finding minima

We require that $\nabla E_0 = \mathbf{0}$.

$$\frac{\partial E_0}{\partial t_0} = \frac{\partial E_0}{\partial t_1} = \frac{\partial E_0}{\partial t_2} = \cdots = \frac{\partial E_0}{\partial t_{z-1}} = \frac{\partial E_0}{\partial t_z} = 0$$

$$\frac{\partial E_0}{\partial t_0} = \frac{\partial \left(a_0 + b_0 E_1\right)}{\partial t_0} = \frac{\mathrm{d}a_0}{\mathrm{d}t_0} + b_0 \frac{\partial E_1}{\partial t_0} + E_1 \frac{\mathrm{d}b_0}{\mathrm{d}t_0} = \frac{\mathrm{d}a_0}{\mathrm{d}t_0} + E_1 \frac{\mathrm{d}b_0}{\mathrm{d}t_0}$$

$$\frac{\partial E_0}{\partial t_i} = \frac{\partial \left(a_0 + b_0 \left(a_1 + b_1 \left(a_2 + b_2 \left(\cdots a_i + b_i E_{i+1} \cdots\right)\right)\right)\right)}{\partial t_i}$$

$$\frac{\partial E_0}{\partial t_i} = \frac{\partial \left(b_0 b_1 \cdots b_{i-1} \left(a_i + b_i E_{i+1}\right)\right)}{\partial t_i}$$

$$\frac{\partial E_0}{\partial t_i} = b_0 b_1 \cdots b_{i-1} \frac{\partial \left(a_i + b_i E_{i+1}\right)}{\partial t_i}$$

$$\frac{\partial E_0}{\partial t_i} = b_0 b_1 \cdots b_{i-1} \left(\frac{\mathrm{d}a_i}{\mathrm{d}t_i} + E_{i+1} \frac{\mathrm{d}b_i}{\mathrm{d}t_i}\right)$$

Finding minima

Thus,

$$\frac{\partial E_0}{\partial t_i} = 0 \implies \frac{\mathrm{d}a_i}{\mathrm{d}t_i} + E_{i+1}\frac{\mathrm{d}b_i}{\mathrm{d}t_i} = 0,$$

As b_i are strictly positive.

Finding minima

Well, what are $\frac{\mathrm{d}a_i}{\mathrm{d}t_i}$ and $\frac{\mathrm{d}b_i}{\mathrm{d}t_i}$?

$$\frac{\mathrm{d}a_i}{\mathrm{d}t_i} = \frac{\mathrm{d}}{\mathrm{d}t_i}\left(\frac{\int_0^{t_i} xf(x)\mathrm{d}x + t_i\int_{t_i}^{\infty} f(x)\mathrm{d}x}{\int_0^{\infty} f(x)\mathrm{d}x} + g(t_i)\right)$$

$$\frac{\mathrm{d}a_i}{\mathrm{d}t_i} = \frac{t_i f(t_i) + t_i(-f(t_i)) + \int_{t_i}^{\infty} f(x)\mathrm{d}x}{\int_0^{\infty} f(x)\mathrm{d}x} + g'(t_i)$$

$$\frac{\mathrm{d}a_i}{\mathrm{d}t_i} = \frac{\int_{t_i}^{\infty} f(x)\mathrm{d}x}{\int_0^{\infty} f(x)\mathrm{d}x} + g'(t_i)$$

$$\frac{\mathrm{d}b_i}{\mathrm{d}t_i} = \frac{\mathrm{d}}{\mathrm{d}t_i}\left(\frac{\int_{t_i}^{\infty} f(x)\mathrm{d}x}{\int_0^{\infty} f(x)\mathrm{d}x}\right) = \frac{-f(t_i)}{\int_0^{\infty} f(x)\mathrm{d}x}$$

Finding minima

So,

$$\frac{\int_{t_i}^{\infty} f(x)\mathrm{d}x}{\int_{0}^{\infty} f(x)\mathrm{d}x} + g'(t_i) + E_{i+1}\left(\frac{-f(t_i)}{\int_{0}^{\infty} f(x)\mathrm{d}x}\right) = 0.$$

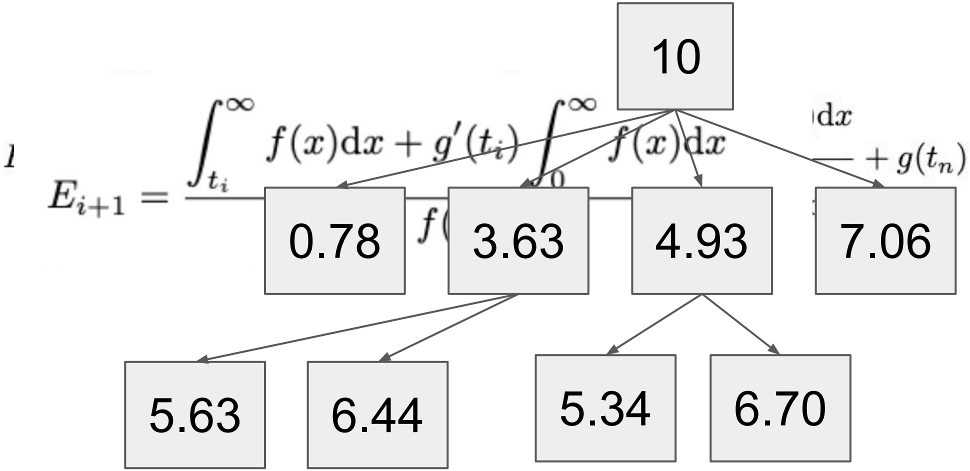
$$\int_{t_i}^{\infty} f(x)\mathrm{d}x + g'(t_i)\int_{0}^{\infty} f(x)\mathrm{d}x - f(t_i)E_{i+1} = 0$$

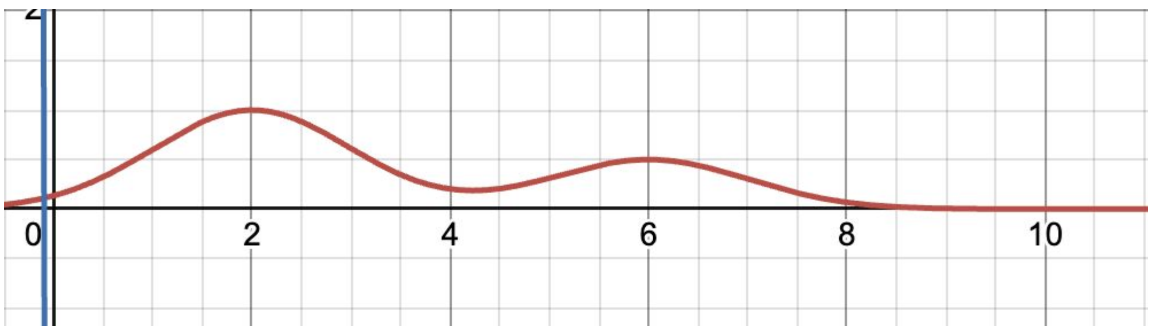
$$E_{i+1} = \frac{\int_{t_i}^{\infty} f(x)\mathrm{d}x + g'(t_i)\int_{0}^{\infty} f(x)\mathrm{d}x}{f(t_i)}$$

As we know t_z, and have that $E_z = \frac{a_z}{1-b_z}$, we can thus work our way backwards to determine all t_i.

Math behind mathematical model (3)

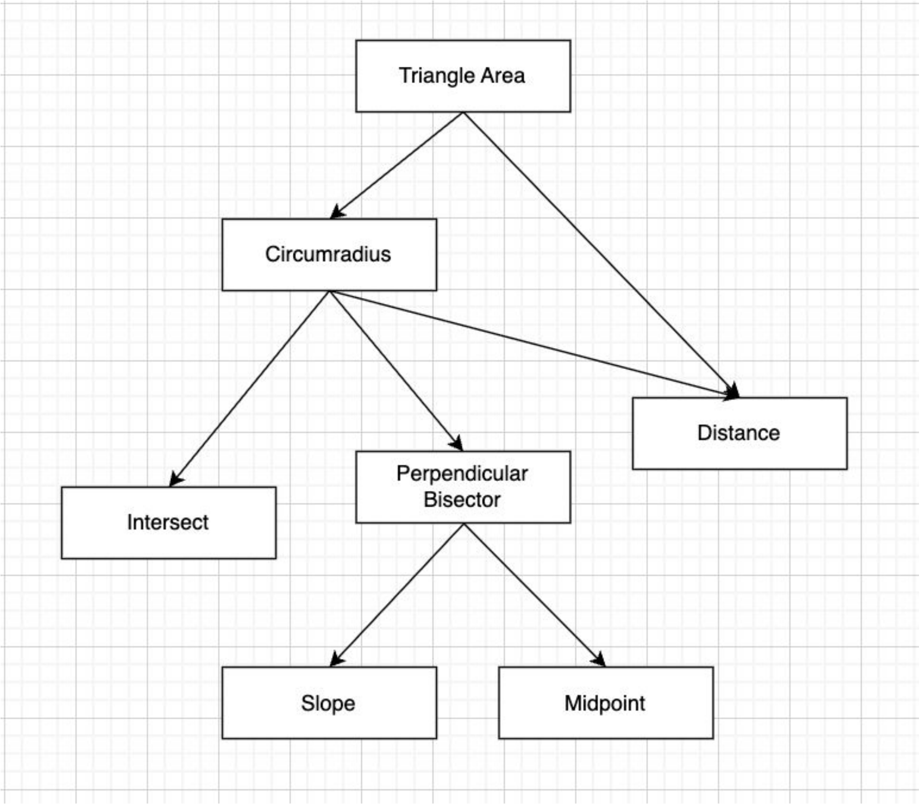
$$E_{i+1} = \frac{\int_{t_i}^{\infty} f(x)\mathrm{d}x + g'(t_i)\int_{0}^{\infty} f(x)\mathrm{d}x}{f(t_i)}$$



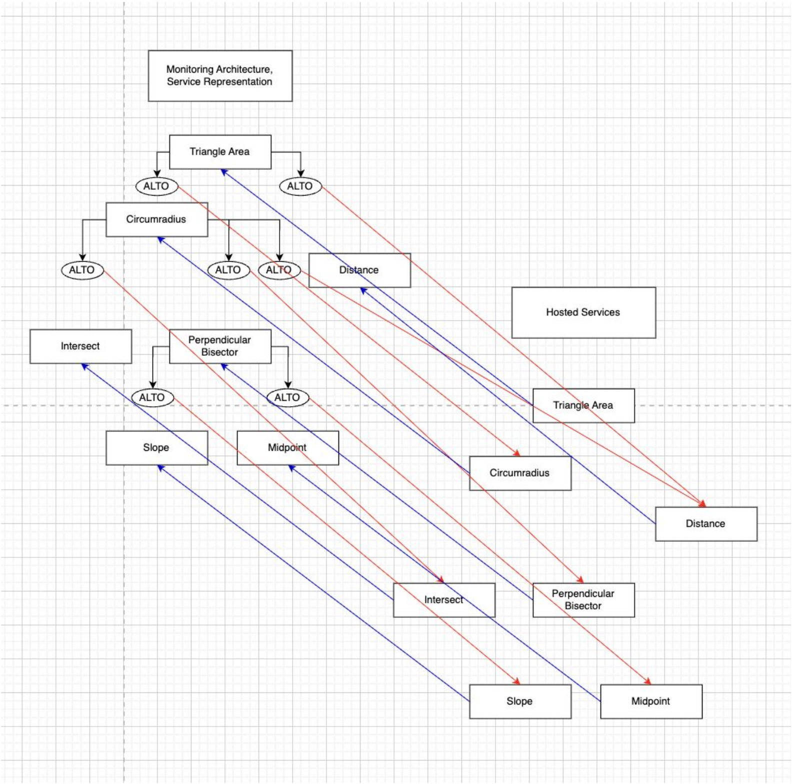


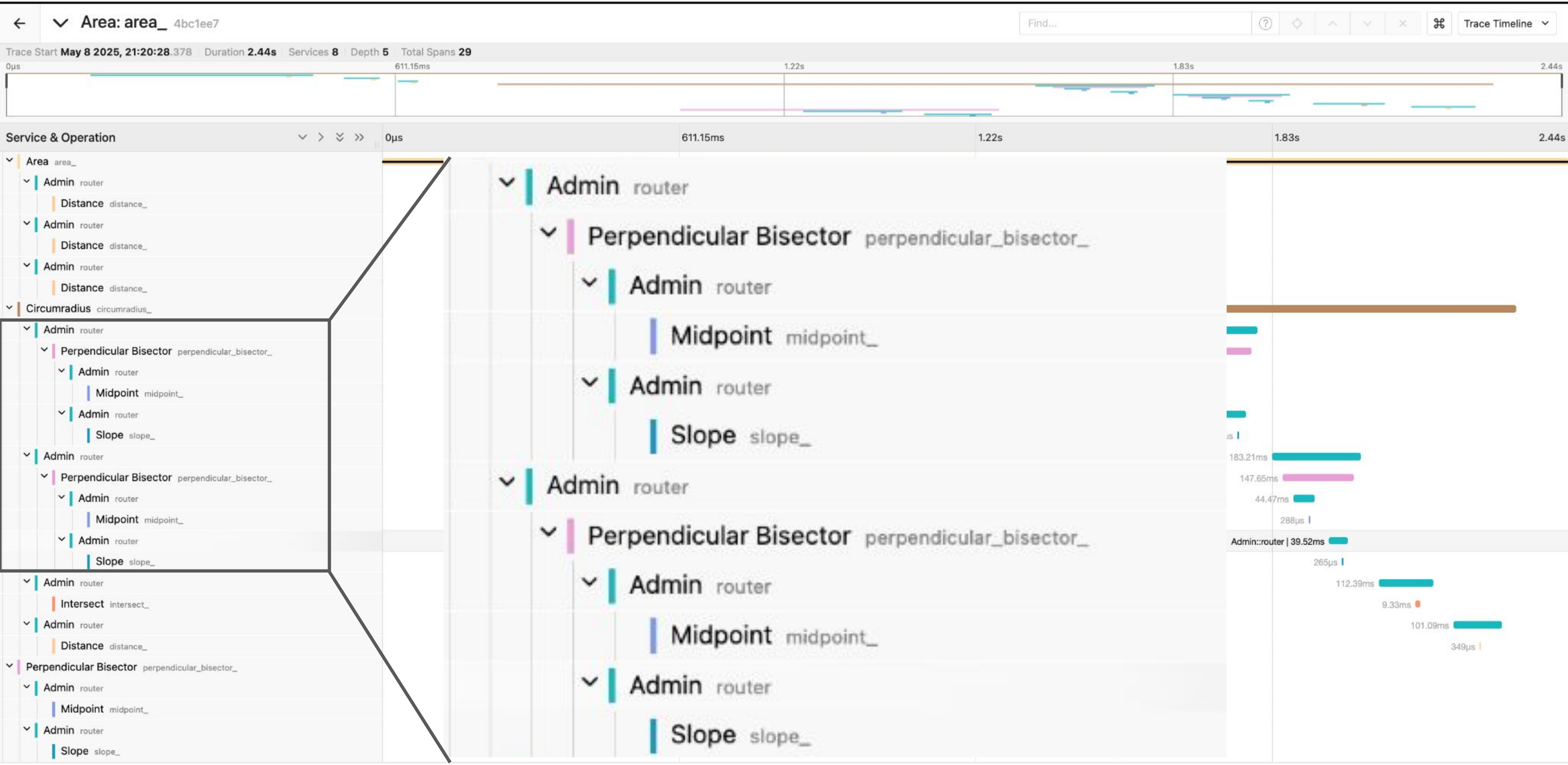

```
0.78
    10
    ---> 6.33
3.63
    5.63
        10
        ---> 3.53
    6.44
        10
        ---> 3.49
    10
    ---> 3.84
4.93
    5.34
        10
        ---> 3.89
    6.7
        10
        ---> 3.84
    10
    ---> 4.16
7.06
    10
    ---> 3.93
10
---> 5.61
```

Testbed Architecture



Global Architecture





Area: area_ 4bc1ee7
Find...
Trace Timeline
Trace Start May 8 2025, 21:20:28.378 Duration 2.44s Services 8 Depth 5 Total Spans 29
0µs
611.15ms
1.22s
1.83s
2.44s
Service & Operation
Area area_
Admin router
Distance distance_
Admin router
Distance distance_
Admin router
Distance distance_
Circumradius circumradius_
Admin router
Perpendicular Bisector perpendicular_bisector_
Admin router
Midpoint midpoint_
Admin router
Slope slope_
Admin router
Perpendicular Bisector perpendicular_bisector_
Admin router
Midpoint midpoint_
Admin router
Slope slope_
Admin router
Intersect intersect_
Admin router
Distance distance_
Perpendicular Bisector perpendicular_bisector_
Admin router
Midpoint midpoint_
Admin router
Slope slope_
183.21ms
147.65ms
44.47ms
288µs
Admin::router | 39.52ms
265µs
112.39ms
9.33ms
101.09ms
349µs