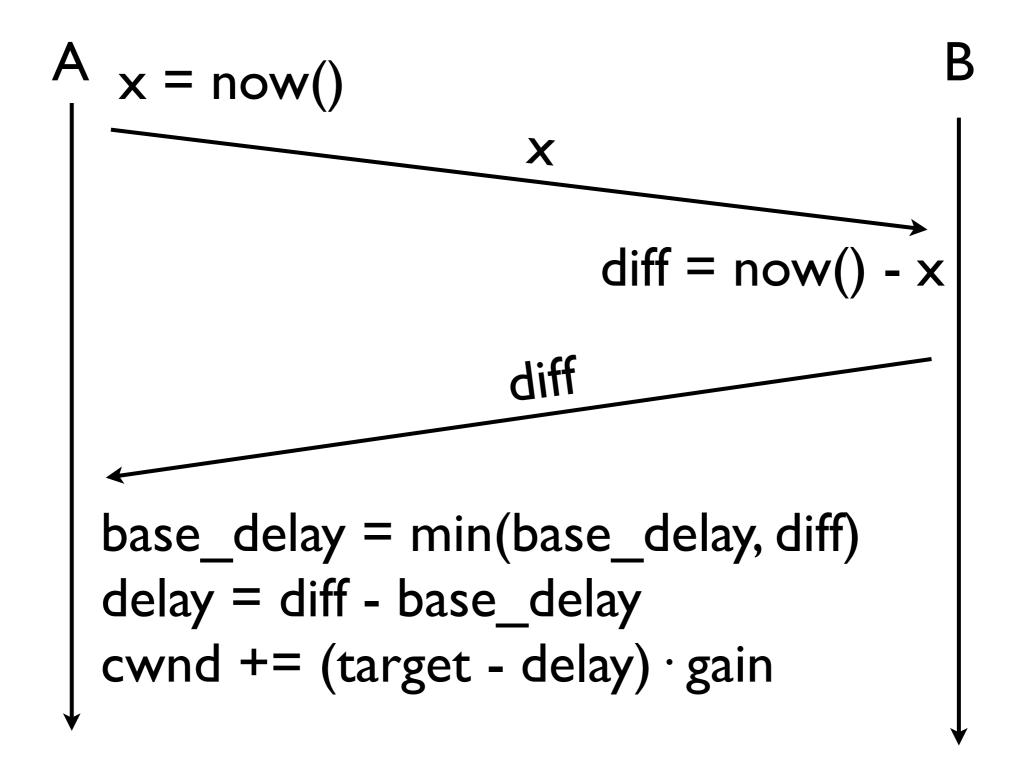
correcting for clock drift in uTP and LEDBAT

Arvid Norberg, arvid@bittorrent.com Bram Cohen, bram@bittorrent.com



- using a base delay of the lowest ever received diff assumes that at least one packet will get through with no buffering delay
- only ever adjusting base_delay downwards assumes no clock drift

base_delay [5461953] 0x015cfc18; UTP_Connect conn_seed;2660758874 packet_size;300 (B) target_delay;100 (ms) delay_history;3 delay_base_history;13 (minutes) socket; 0x015a9a60 4,28925e+89 4,28925e+09 actual_delay (us)-x1y1 delay base (us)-x1y1 4.2892e+09 4.2892e+09 4,28915e+09 4,28915e+09 4,2891e+09 4,2891e+89 4.28905e+09 4.28905e+09 4.289e+09 4.289e+09

250

time (s)

350

400

450

500

300

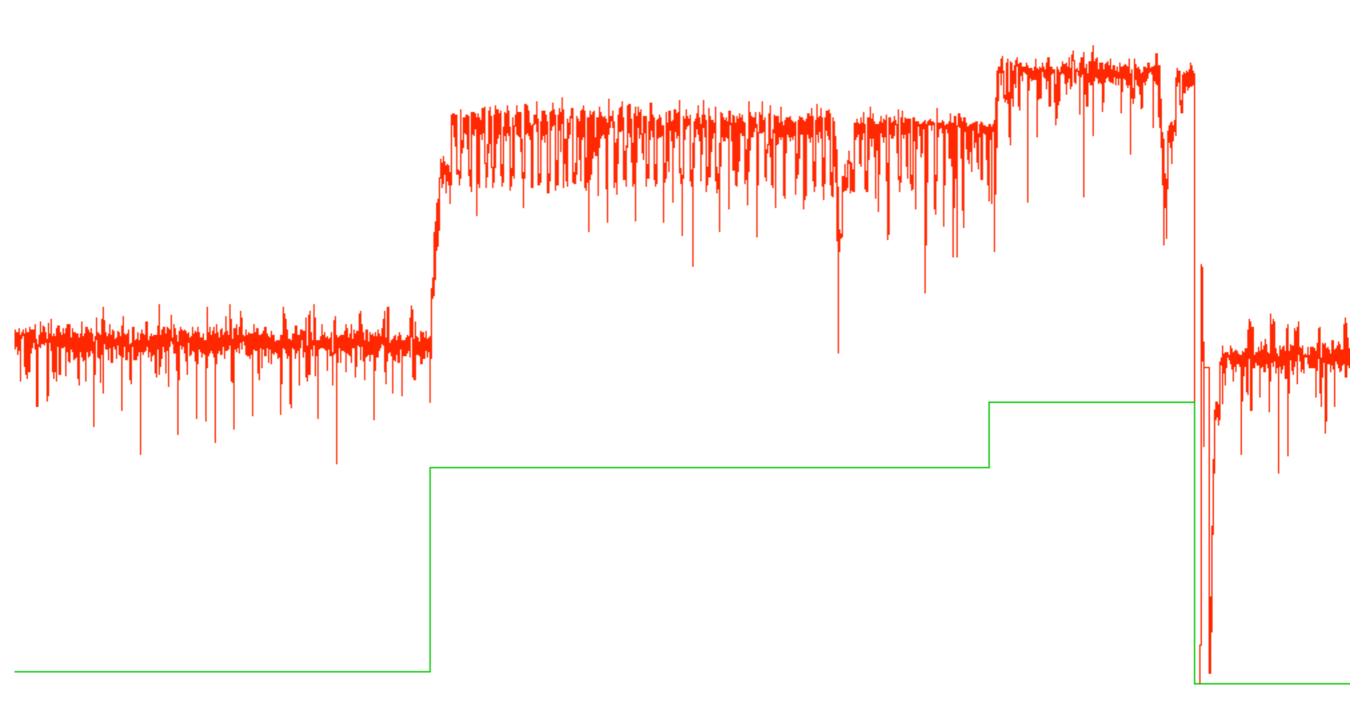
150

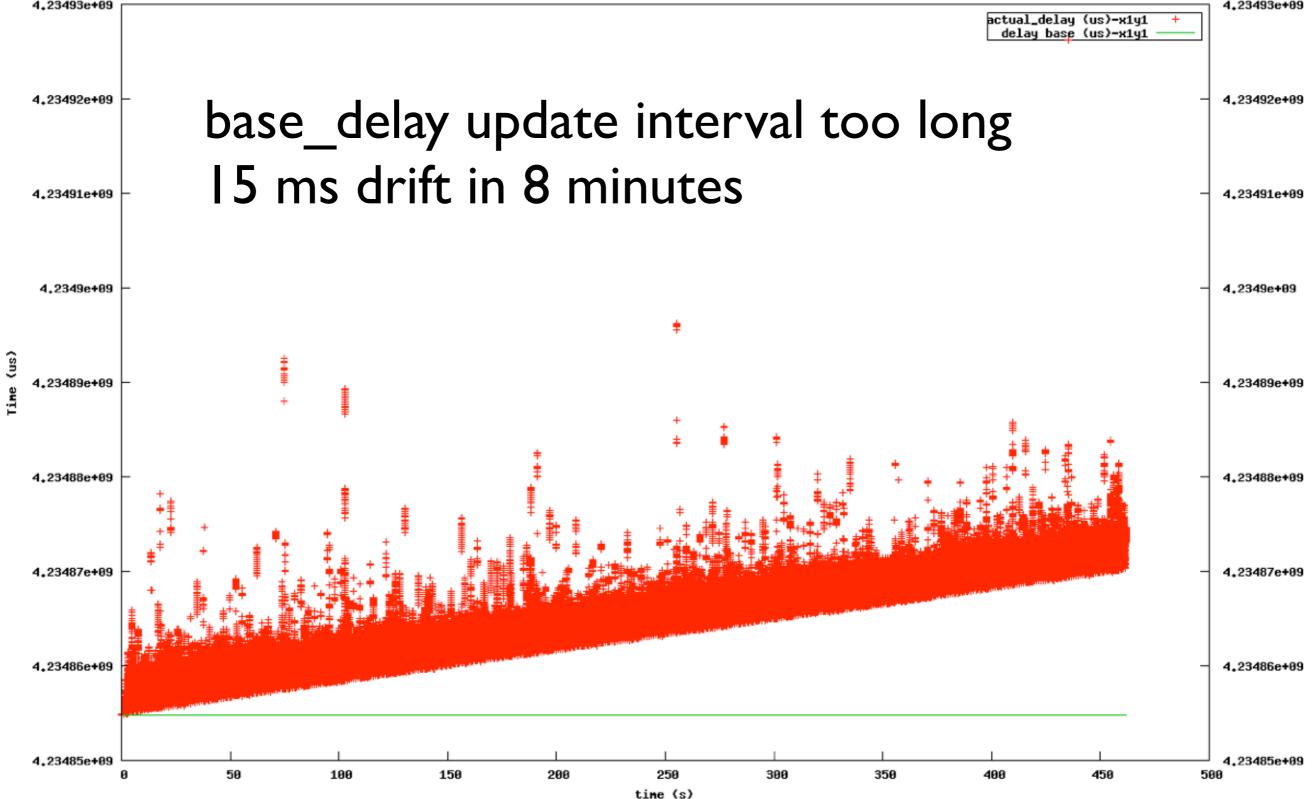
100

200

- One solution to clock drift is to update base_delay periodically with the lowest seen diff in the last period
- What should the interval be?
 - If it's too short, we'll set the base delay to a sample that isn't the true lowest value, and our measurements are unreliable
 - If it's too long, we'll allow a significant drift before adjusting, and our measurements are unreliable

delay_base update interval too short delays are artificially kept too low for certain periods





- Tests show that the update interval cannot be less than around 10 minutes on our DSL line
- Other tests on a LAN show a significant drift over 10 minutes
- There is no good magic number for the base_delay update interval

- However, if the clock drift is in your favor, it's not a problem
- Since the drift causes the delay measurements to appear smaller, the base_delay is updated with the drift

base_delay [853107] 0x0010b4f0: UTP_Connect conn_seed:44316 packet_size:300 (B) target_delay:100 (ms) delay_history:3 delay_base_history:13 (minutes socket: 0x00109890 3,307e+07 3.307e+07 actual_delay (us)-x1y1 delay base (us)-x1y1 3.3065e+07 3.3065e+07 3,306e+07 3.306e+07 3.3055e+07 3.3055e+07 3,305e+07 3,305e+07 3.3045e+07 3,3045e+07 3,304e+07 3.304e+07 3.3035e+07 3,3035e+07 3,303e+07 3.303e+07 0 100 200 300 400 500 600 time (s)

- We can take advantage of this fact by keeping track of the other end's delay as well as our own
- Whenever the other end's base_delay is adjusted downwards, we know it's adjusting for drift and we can adjust our own base_delay the same amount upwards

```
A = now()
                    X
                            y = now()
                  diff, y
base_delay = min(base_delay, diff)
delay = diff - base delay
their diff = now() - y
if (their diff < their base_delay) {</pre>
  base delay += their base delay - their diff
 their base delay = their diff
```

base_delay [6130562] 0x015a92f8; UTP_Connect conn_seed;2915978084 packet_size;300 (B) target_delay;100 (ms) delay_history;3 delay_base_history;13 (minutes) socket; 0x015a92f8 4.24935e+09 4.24935e+09 actual_delay (us)-x1y1 delay base (us)-x1y1 4,2493e+09 4.2493e+09 4,24925e+89 4,24925e+09 4,2492e+89 4.2492e+09 4,24915e+09 4.24915e+09 4,2491e+09 4.2491e+09 4.24905e+09 4.24905e+09 50 100 400 450 150 200 250 300 350 500 time (s)

