



Arm Solutions at Lightspeed

Remaining EEVDF scheduling latency and lag improvement

Scheduler and Real-Time MC

Linux Plumber 2026

Vincent Guittot

Agenda

Remaining short task scheduling issues

- Remaining issues with a single task
- Issues with multiple tasks
- Open issues

Race between wake up and pick

Remaining issues with single task

- Short task wakes up and cancels protection
 - More tasks enqueue w/ positive lag before pick_next_task()
 - Short task becomes ineligible
- Wakeup on idle cpu
 - Any positive vlag disappears (nothing to apply against)
 - Do not set buddy because CPU is idle
 - More tasks enqueue before pick_next_task
 - Short task becomes ineligible

What to do ?

Remaining issues with single task

- Somehow unfair
 - Only an ordering issue
 - `pick_next_task()` not called yet
- Clear vlag when enqueue on an idle CPU ?
 - Unfair when load balancing migrates the task
 - Positive vlag from contentions on prev CPU
- Check at enqueue for wakeup
 - Worth for short slice tasks

Eternal positive lag

Remaining issues with single task

- Negative lag can disappear by delaying the dequeue
 - But not positive lag
- Long sleeping task can keep large positive vlag forever
 - positive lag no more relevant

What to do ?

Remaining issues with single task

- Keep tasks “enqueued” like negative lag
- Linear decay w/ raw clock:
- Compare vruntime at dequeue with avg vruntime at enqueue ?
- Idle CPUs trigger positive lag reset
 - Already the case for negative lag

Race between wake up and pick

Multi tasks issues

- Wake up cancels protect
 - Set next buddy
 - More short slice tasks enqueue before `pick_next_task()`
 - Next buddy overwritten
- Multi short tasks on same CPU
 - slice is not shorter
 - Preempt short doesn't work

What to do ?

Multi tasks issues

- Add task in pushable list
 - In addition to `set_next_buddy`
 - Look for another CPU if not picked
- Select best CPU
 - Need lockless access to min/max slice
 - Take min slice into account in `select_task_rq_fair()`
 - Take min slice into account inside `find_energy_efficient_cpu`
 - Do not consider only energy

Opportunistic lag credit

Open issues

- Extra positive lag
 - TA accumulate large positive lag
 - TB enqueued
 - TA run few usec and dequeue
 - TB receive its share of this positive lag
 - lag increased while it never competed with it



linaro.org

Thank you

add social media links or your personal contacts