Frequency-invariance gaps in current kernel

Zhang Rui <rui.zhang@intel.com>
Chen Yu <yu.c.chen@intel.com>
Frequency-Invariance background

Load Balancer

- find_busiest_group()
- group_is_overloaded()
- group_has_capacity()

Without Frequency Invariance

- util_avg/load_avg
- clock_pelt

Clock Task

- clock_task
- freq_cur
- freq_max

Clock_task can be bigger if the CPU is running at slower frequency
freq_max can be wrong
Bogus freq_max when frequency range changed

Workload: 80% Busy on CPU0/CPU2
Limit CPU2 frequency via sysfs

freq_max not updated

Under-estimated util-avg confuses load balancer
Bogus `freq_max` when turbo/throttled

- `freq_max` lower than actual max frequency
- `freq_max` higher than actual max frequency
- Under-estimated `util-avg` confuses load balancer
- `arch_freq_scale` clamped

**Workload:**
- CPU0/CPU2 90% Busy
- 90% Busy on other CPUs
under-estimated util_avg on Ecore CPU
confuses Load Balance

Workload:
90% Busy on PcoreCPU0/EcoreCPU12
90% Busy on other CPUs
Next steps?

Possible improvements:
- CPUs have different freq_max (Hybrid/SST-TF/SST-BF)
- Frequency range changes at runtime (CTDP/SST-PP/Userspace)
- Per cpu freq_max instead of globally unique value
- Allow updating freq_max at runtime

freq_max when turbo/throttle:
- Unknown
- Changes over time
- Need to be accurate

Idea?
CPU capacity affected by SMT sibling cannot be measured by frequency scaling. Ideas?

100% Busy on Core0smt0/Core1smt0
Increasing workload on Core0smt1
Backup2
frequency invariance on Intel

- frequency invariance
- (aperf_delta / mperf_delta) * base_freq
- base frequency in Non-turbo mode
- 4th entry (4Core Turbo) in Turbo mode
Backup 3

Lose accuracy when freq_max is low

Workload:
- Fixed task size on CPU0/CPU2 (yogini rate50)
  - CPU0: performance
  - CPU2: schedutil

util_avg is lower on CPU0 because CPU0 is running at a frequency higher than freq_max
Backup 4
utilization based freq_max estimation

- Assumption
  - either firmware or software is targeting for higher frequency when CPU is busier
  - Under-estimated util_avg does not impact much on CPUs with high Idle residency

- Solution
  - Weight current frequent in freq_max calculation
  - Weight is a variant based on Busy% (CPU utilization)
  - Busy% = mperf_delta / tsc_delta

Estimated freq_max value when CPU is throttled from 3G to 2G