Optimize idle CPU scan during task wake up

Chen Yu <yu.c.chen@intel.com>
problems during idle CPU scan

- problem1: overscan on CPUs
  - limit CPU scan depth
    - partially addressed by SIS_UTIL in 2022
  - restrict CPU scan set
    - idle CPU mask in LPC2021
- problem2: cross CPU wakeup
  - How to fix?

SIS_FILTER under evaluation in 2022
cross runqueue lock causes high idle percentage

will-it-scale context_switch test, system is overloaded

8.94% raw_spin_rq_lock_nested.constprop.0
try_to_wake_up;default_wake_function
autoremove_wake_function
__wake_up_common
__wake_up_common_lock
__wake_up_sync_key
pipe_write
new_sync_write
vfs_write
ksys_write
write

5.41% raw_spin_rq_lock_nested.constprop.0
__sched_text_start
schedule_idle
do_idle

high idle percentage ('top' command)

%Cpu0 : 2.7 us, 66.7 sy, 0.0 ni, 30.7 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
wakelist mitigates lock contention, but still long idle

6.97% update_cfs_group
   dequeue_entity
   dequeue_task_fair
   __sched_text_start
   schedule;pipe_read
   vfs_read
   ksys_read
   read

6.32% update_cfs_group
   enqueue_entity
   enqueue_task_fair
   ttwu_do_activate
   sched_ttwu_pending
   __flush_smp_call_function_queue
   flush_smp_call_function_queue
   do_idle

%Cpu0 : 5.3 us, 60.7 sy, 0.0 ni, 34.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
speculated scenario for long idle duration:

1: sched_ttwu_pending() is costly
2: idle duration is long
3: SIS_UTIL suggests waker to scan more CPUs
4: wakee waits for more time to be woken up
Lower the bar to find an “idle” CPU, avoid cross CPU wakeup?

1. The task on current CPU and the wakee task are both short running tasks
2. System is busy
3. `avg_load` or `avg_util` > threshold?
4. `sum_nr_running` > threshold?
5. Average running time < `sysctl_sched_min_granularity`? (no preemption)
benchmarks

**netperf improvement%**

- Number of netperf threads: 28, 56, 84, 112, 140, 168, 196, 224
- Improvement values:
  - TCP_RR: 0, 0, 0, 0, 200, 180, 160, 140
  - UDP_RR: 0, 0, 0, 0, 140, 120, 100, 80

**hackbench improvement%**

- Groups: 1, 2, 4, 8
- Improvement values:
  - Process-pipe: 0, 0, 0, 14
  - Process-sockets: 0, 0, 0, 4
Thank you!