update on tests filling up full KNL with pileup events

4 messages

Slava Krutelyov <slava77@fnal.gov>   Fri, Mar 16, 2018 at 7:09 PM
To: "mic-trk-rd@cern.ch" <mic-trk-rd@cern.ch>

Hi,

I've looked at timing with 4K ttbar+PU70 events filling up the full KNL.

Back in Oct 6 (right after the full-det-tracking was merged) the best
time was based on
256 threads and 64 events in flight
- regular run   65.7 s in the event loop
- numactl bind to use HBM   51.0 s in the event loop
here HBM is the high bandwidth memory, which is not used at all in the
regular run.

Now, with the same 256 thread 64 events in flight
- regular run   56.2 s in the event loop
- numactl bind to use HBM   46.8 s in the event loop

So, the "50 sec per 4000 events" or about 80 events/s is still
approximately correct.

It looks like without HBM (high bandwidth memory) the gain in total time
is about 17%,
while it's only 9% with HBM.

perf stat reports that the number of instructions has decreased, which
 appears to be the largest factor in the speedup.
The instruction count is down
- by 9% without HBM
- by 6% with HBM
We run less code, which is likely due to the cleaning updates and other
optimizations.
The difference in the instruction count with and without HBM makes me a
bit suspicious what the instruction count is. It probably includes more
than just our code, but also some system overhead.

L1 data cache misses are down by 30% (times 1.3), which is the largest
change among the counters that I checked with perf stat.

Cheers

        --slava

--
****
 Vyacheslav (Slava) Krutelyov
 UCSD: 9500 Gillman Dr, Physics UCSD 0354, La Jolla, CA 92093-0354
 CERN: 42-R-027
 AIM/Skype: siava16   googleTalk: slava77@gmail.com
 (630) 291-5128 Cell (US)   +41 76 275 7116 Cell (CERN)

Kevin McDermott <kpm82@cornell.edu>   Fri, Mar 16, 2018 at 7:43 PM
To: Slava Krutelyov <slava77@fnal.gov>
Cc: "mic-trk-rd@cern.ch" <mic-trk-rd@cern.ch>

Hi Slava,

Dan made a number of updates that reduce the total loop time per event by moving the remap functions to vectors. Does
this analysis include the merged PR from Dan this morning?

peace
Kevin
[Quoted text hidden]

Slava Krutelyov <slava77@fnal.gov>   Fri, Mar 16, 2018 at 8:19 PM
To: Kevin McDermott <kpm82@cornell.edu>
Cc: "mic-trk-rd@cern.ch" <mic-trk-rd@cern.ch>

On 3/16/18 4:43 PM, Kevin McDermott wrote:
> Hi Slava,
>
> Dan made a number of updates that reduce the total loop time per event
> by moving the remap functions to vectors. Does this analysis include
> the merged PR from Dan this morning?

Hi Kevin,

No, this analysis is as of PR133, and matches what I was showing in the TSG.

        --slava

>
> peace
> Kevin
>
> On Fri, Mar 16, 2018 at 7:09 PM, Slava Krutelyov <slava77@fnal.gov
[Quoted text hidden]

Slava Krutelyov <slava77@fnal.gov>   Mon, Mar 19, 2018 at 6:53 PM
To: Kevin McDermott <kpm82@cornell.edu>
Cc: "mic-trk-rd@cern.ch" <mic-trk-rd@cern.ch>

Hi Kevin and all,

I checked the numbers as of pr135.

The single thread got worse (based on 200 events tested; using CE):
- build time   71 s   -> 87 s   [worse by 23%]
- loop time   107 s -> 111 s
At the level of the job total, perf stat says that the number of
instructions per cycle is down by about 13%, while the total number of
instructions went down 9%.

Performance with the full load of the KNL is significantly better, by
about 40%.
The same 4000 events as before
 256 threads and 64 events in flight:
- 40.6 s total, 39.1 s in the event loop without HBM

- 34.8 s total, 33.5 s in the event loop with HBM

Most of the gain in timing here comes from the improvement in the CPU utilization (the same as improved parallelization or reduction of serial code):
- as of pr133 the job without HBM was hanging around 144 CPUs (56%)
- now the CPU utilization is 193 CPUs

One interesting change is the L1-dcache-load-misses, which have increased by ~30%. It's unclear if we started using more data per computation or it is just the CPU efficiency improvement push the total up.

With the latest code the most optimal way to fill 256 threads is now closer to 36 events in flight (205 CPUs used on average), compared to the optimum around 56 events in flight as of pr133 (148 CPUs used on average).

Cheers

     --slava

On 3/16/18 4:43 PM, Kevin McDermott wrote:
> Hi Slava,
>
> Dan made a number of updates that reduce the total loop time per event
> by moving the remap functions to vectors. Does this analysis include
> the merged PR from Dan this morning?
>
> peace
> Kevin
>
> On Fri, Mar 16, 2018 at 7:09 PM, Slava Krutelyov <slava77@fnal.gov>
[Quoted text hidden]