Proceedings of the CTD/WIT 2019
PROC-2019-027
October 18, 2019

Level-1 Track Finding with an all-FPGA System at CMS for the HL-LHC

Thomas James on behalf of the CMS Collaboration

Department of Physics
Imperial College London, UK

ABSTRACT

The Compact Muon Solenoid (CMS) experiment at the Large Hadron Collider (LHC) is designed to study a wide range of high energy physics phenomena. It employs a large all-silicon tracker within a 3.8 T magnetic solenoid, which allows precise measurements of transverse momentum (p_{T}) and vertex position. This tracking detector will be upgraded to coincide with the operation of the High-Luminosity LHC, which will provide luminosities of up to $7.5 \times 10^{35}\,\mathrm{cm}^{-2}\mathrm{s}^{-1}$ to CMS, or 200 collisions per 25 ns bunch crossing. This new tracker must maintain the nominal physics performance in this more challenging environment. Novel tracking modules that utilise closely spaced silicon sensors to discriminate on charged particle p_{T} have been developed and allow the selective readout of hits compatible with tracks of $p_{\mathrm{T}} > 2 - 3\,\mathrm{GeV}$ to off-detector trigger electronics. This would allow the use of tracking information at the Level-1 trigger of the experiment, a requirement to keep the Level-1 triggering rate below the 750 kHz target, while maintaining physics sensitivity. This paper presents a concept for an all-FPGA based track finder using a fully time-multiplexed architecture. Hardware demonstrators have been assembled to prove the feasibility and capability of such a system. The performance for a variety of physics scenarios will be presented, as well as the proposed scaling of the demonstrators to the final system.

PRESENTED AT

Connecting the Dots and Workshop on Intelligent Trackers (CTD/WIT 2019)
Instituto de Física Corpuscular (IFIC), Valencia, Spain
April 2-5, 2019

1 CMS and the HL-LHC

The Compact Muon Solenoid (CMS) experiment [1] is a large, all purpose particle detector, designed to investigate a wide range of physics at the Large Hadron Collider (LHC) [2]. CMS was designed to operate with an average number of simultaneous collisions (pileup) of ~ 25, with a bunch crossing rate of 40 MHz. A major feature of CMS is the 1.2 m radius, 200 m^2 area silicon strip tracker, the largest silicon tracker in operation in the world. As the tracker sits within the 3.8 T superconducting solenoid, the transverse momentum, p_T, of charged particles can be measured from the track curvature.

CMS operates a Level-1 (L1) trigger to reject uninteresting events [3]. This enacts a rate reduction on the order of a factor of 400. At the time of construction, it would have been unfeasible to read-out the tracker data at 40 MHz to help in the Level-1 decision making, primarily due to the large data size and rate.

By 2026, the LHC will be upgraded in luminosity to about $5 - 7.5 \times 10^{34} \mathrm{cm}^{-2}\,\mathrm{s}^{-1}$ (or $140 - 200$ pileup). This will enable an increase in integrated luminosity of approximately 3 times by 2035, with respect to no such upgrade. This upgraded LHC machine is known as the High-Luminosity LHC (HL-LHC) [4], and is targeting a lifetime integrated luminosity of 3,000 - 4,000 fb^{-1}.

During the long-shutdown preceding HL-LHC operation, large sections of the CMS detector will be replaced. The silicon tracker will have accumulated significant radiation damage, and a replacement is being constructed [5, 6]. Simulation studies show that with this luminosity, a new handle is needed at the L1 trigger stage in order to maintain thresholds and sensitivity to interesting physics, while at the same time keeping the L1 accept rate within the expected limit of 750 kHz. Present thresholds without track information would approach a 4 MHz L1 accept rate at 200 pileup [7]. The new outer-tracker will therefore incorporate a novel design to allow the read-out of tracking information at 40 MHz to the L1 trigger.

2 The CMS Phase 2 Upgrade

New tracking modules, under development for the HL-LHC, utilise a pair of closely spaced silicon sensors ($1.6 - 4.0$ mm) and on-detector correlation logic in order to discriminate charged particle tracks with a p_T exceeding a threshold of $2 - 3$ GeV. The module design is depicted in Figure 1. A pair of clusters (one in each sensor) consistent with such a track is known as a 'stub'. Only the stubs are sent to the off-detector L1 trigger electronics (at 40 MHz), enacting a rate reduction of approximately a factor of ten. There remains, however, an average of 12,000 - 15,000 stubs per bunch crossing which must be processed, and matched to particle trajectories.

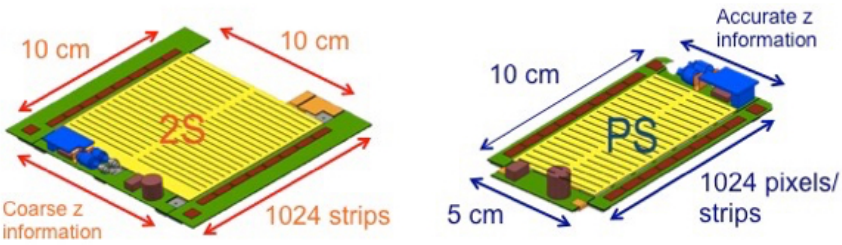

Figure 1: Layout of the two-strip (2S) and pixel-strip (PS) p_T modules, as being developed for the upgraded CMS Outer Tracker. The PS modules contain a layer of 1.5 mm macro-pixels, which give finer granularity in the direction perpendicular to the strips. These modules will be used in the regions $r < 60$ cm, where the hit occupancy is higher.

The upgraded L1 trigger for CMS at the HL-LHC will include a 'correlator' layer, which will combine information from the calorimeter, muon, and tracker systems, in order to make physics objects which can

be selected on with a menu-like global trigger. The combination of tracker and calorimeter objects greatly improves the pileup rejection capabilities at L1. Particle and pileup identification algorithms have been shown to significantly improve the performance of both transverse energy (E_{T}) and missing E_{T} triggers [8]. Vertex finding algorithms have also been implemented in FPGA logic, and have been demonstrated to perform well for physics events with high energy jets, even in the presence of $140-200$ pileup.

The latency budget of the L1 trigger is 12.5 μs, determined by the depth of the front-end buffers. However, it will take an estimated 1 μs to transmit the data between the front-ends of the detector and the back-end electronics. A further 1 μs is required to propagate the L1 accept signal back to the front-end chips. It is estimated to take 3.5 μs to correlate the trigger primitives from each sub-detector, and make a triggering decision. This leaves only 4 μs for the track-finding and fitting process, accounting for a 30% safety margin. On a L1 accept signal, all front-end buffers will be triggered to read out the information from the selected event to the data acquisition (DAQ) and High Level Trigger (HLT) farm. This architecture is illustrated in Figure 2.

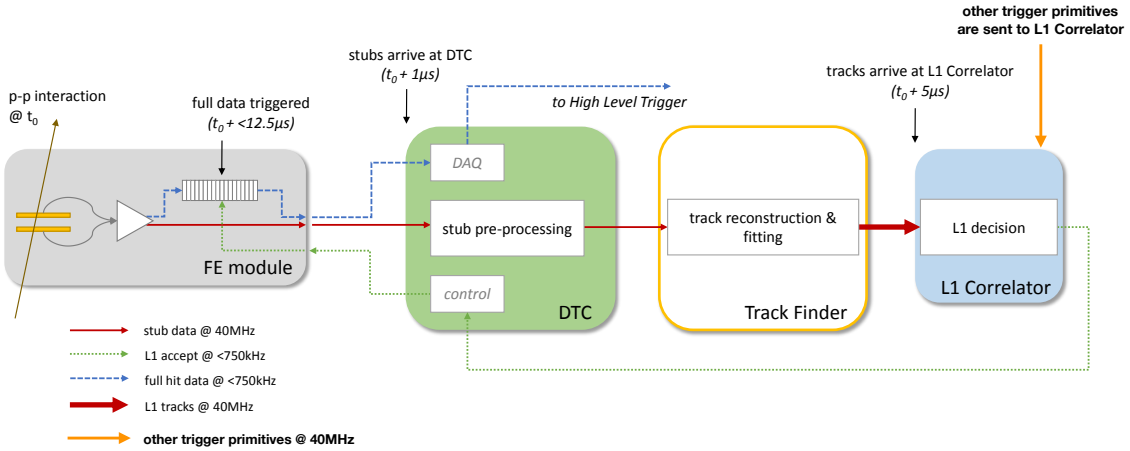

Figure 2: Illustration of dataflow and latency requirements from p_{T} modules through to the off-detector electronics that are dedicated to forming the L1 trigger decision.

3 The Back-end System

The tracker back-end system is responsible for constructing tracks from stubs at every bunch-crossing. It must also control and receive the data from the front-end, and provide a DAQ path for the full dataset on an L1 trigger. The system is divided into two layers of FPGA-based processing: the DAQ, Trigger and Control (DTC) boards; and the Track Finder Processors (TFPs).

A total of 216 DTC boards will be used to read-out the data from the 13,296 modules of the outer-tracker. Up to 72 modules will be connected per board, with bi-directional optical links at 2.56 Gb/s to detector, and either 5.12 or 10.24 Gb/s from detector. The DTC boards are envisaged to host dual Xilinx Ultrascale+ FPGAs [9], allowing for a total bandwidth of about 0.7 Tb/s per card. This layer of the tracker back-end is also responsible for stub pre-processing, which includes: the conversion from a local module-level coordinate scheme to a global position; the sorting of stub data into ϕ-sectors; and the time-multiplexing of the data into one of 18 time-nodes [10]. The cabling between detector and DTC ensures that each of nine sets of 24 DTCs receives data from a single nonant (40 degree region) of the tracker in the azimuthal angle ϕ. In this scheme, the 216 DTC boards will occupy 18 ATCA [11] crates in total, corresponding to two crates (one rack) per nonant.

The TFP layer of the tracker back-end accumulates data from 48 DTCs; one 25 Gb/s optical link from each. There are expected to be 162 TFP boards in total (excluding spares/redundant nodes), each responsible for one nonant in ϕ, and one out of every 18 bunch-crossings (the time-multiplexing period). These boards

will host one or two large FPGA(s), delivering enough processing power to find and fit the track candidates. As the majority of stubs are associated with pileup, the data-rate between the TFP and the L1 correlator is reduced down to around 30 Gb/s.

4 Track-finder Algorithms

Several track-finding algorithms have been studied for Level-1 tracking. Hardware demonstrators have been constructed to prove the feasibility of two candidate algorithms: a Hough Transform followed by a Kalman Filter (KF) [12, 13]; and a 'tracklet' seeding and projection followed by a χ^2 minimisation fit [14].

A hybrid of these two algorithm options is currently under development. This solution consists of a tracklet seeding and projection, followed by a Kalman Filter.

- Tracklet seeds are formed from pairs of stubs in adjacent tracker layers and disks. Using the constraint that the track originates at the interaction point, a set of tracklet helix parameters can be calculated.
- The tracklet parameters are projected to the remaining layers and disks of the tracker, and stubs inside the projection windows are matched to form track candidates.
- Track candidates that share stubs in three or more tracker layers or disks are merged into a single candidate. This removes duplicate candidates prior to the track fit.
- The tracklet helix parameters are then used to seed the Kalman Filter state and covariance matrices. The matched stubs are then applied one-by-one to the KF. The KF selects the set of stubs that best fit a track, and calculates four helix parameters: p_{T}, η and ϕ, and the longitudinal impact parameter z_0.

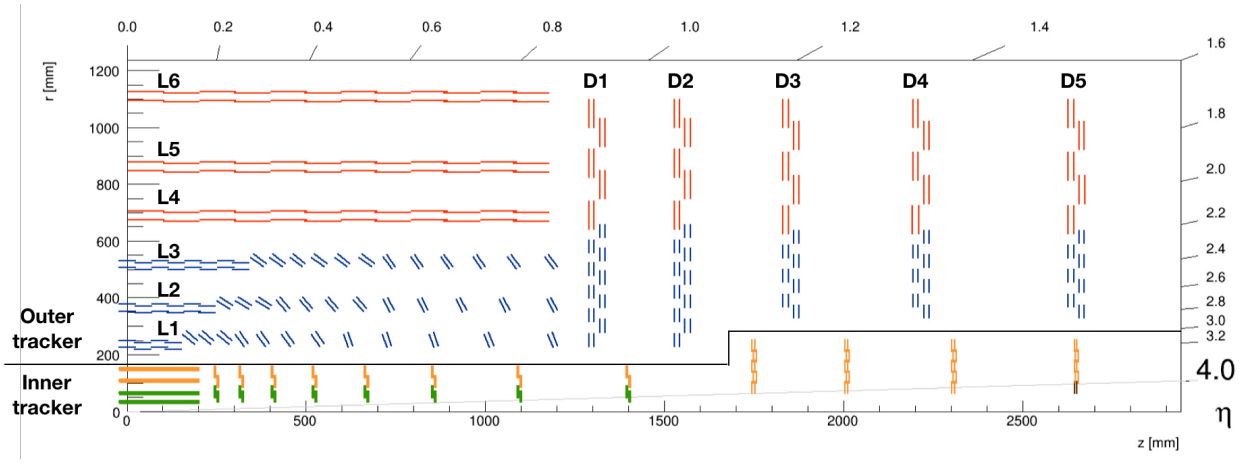

Figure 3: One quadrant of the upgraded tracker geometry and layout. The region within around 20 cm is the Inner Tracker, and will not output data to the L1 trigger path. Six barrel layers and five endcap disks each side make up the Outer Tracker, which consist of p_{T} modules.

Seeds are generated in redundant layer and disk-pairs, to allow for inefficiencies. Figure 3 depicts one quadrant of the upgraded tracker geometry. The following pairs of layers and disks of the outer tracker are used to seed tracking (where 'L' denotes a barrel layer, and 'D' an endcap disk): L1L2, L3L4, L5L6, L1D1, L2D1, D1D2, D3D4. As the KF is able to check all possible combinations of matched stubs to build and fit the best track, wider search windows can be used in the matching stage than with the simpler χ^2 minimisation approach. Stubs used to build the initial tracklet are not re-fitted by the KF, saving latency. The merging of duplicates before the fit reduces the number of track candidates that must be processed

by the KF by a factor of three (from around 570 to 175 candidates). Most duplicates (71%) are generated by the same particle being found in different seeding layers. The remainder are generated from multiple stub combinations associated with a single particle being found within the same seeding configuration. The merging operation costs about 1.5% in track-finding efficiency as a result of nearby tracks from separate particles being combined.

In contrast to the previous demonstrator projects which were written entirely in standard Hardware Description Languages (HDLs), the 'hybrid' solution is being implemented with a High Level Synthesis (HLS) language developed by Xilinx. This change should allow for easier and faster algorithm development, quicker uptake of non-experts, and improved long-term maintainability. As the Xilinx HLS code can be run outside of an FPGA, it also acts as a software emulator for the implemented algorithms. Some aspects of the firmware design, in particular the low-level board and link infrastructure (typically developed by engineers rather than physicists) are expected to remain in HDL. The hybrid algorithm implementation is currently targeting a 250 MHz clock frequency.

4.1 The Kalman Filter

Each Kalman Filter worker (an independent firmware block that that runs the KF, and can process one stub per clock cycle) consists of a state updater, connected to data-flow controlling logic. A simplified block-diagram of a single worker is shown in Figure 4. Incoming stubs are stored in FIFO 1. They are later retrieved by the stub-state associator, which matches stubs to states, in order of increasing radii. The states are managed by the state control, which multiplexes partially worked states (from FIFO 3), and incoming seeds (from FIFO 2). Each state-stub combination is then passed through the state updater, which produces new state and covariance matrices from the weighted average of the seed, previous stub inputs, and new stub inputs. Finally, the most appropriate state for each input track candidate is selected, primarily based on the χ^2 of the fit.

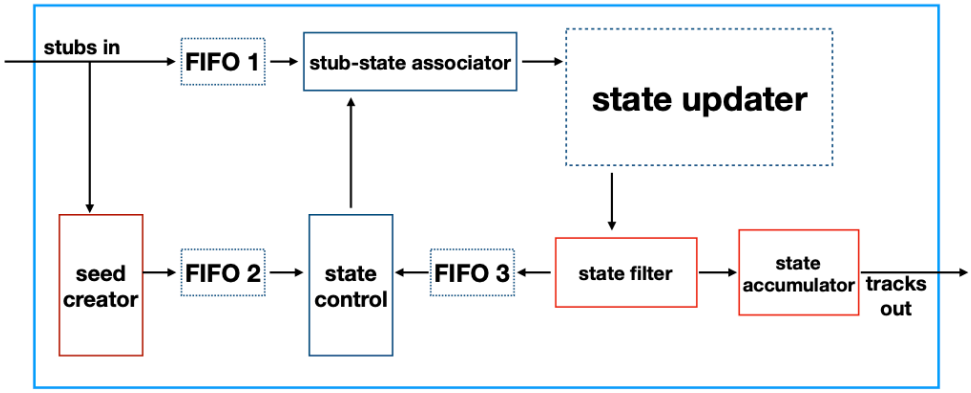

Figure 4: Simplified block diagram of a single KF worker.

The latency of the state updater, in three configurations of clock frequency and FPGA choice are given in Table 1. The resource utilisation of a single state updater block is given in Table 2. For the KU115 FPGA, the HDL control flow logic runs at 320 MHz, and takes 18 clocks (56 ns).

Device	Frequency [MHz]	Number of Clocks	Latency [ns]
KU115	320	46	144
KU115	440	55	125
VU9P	440	40	91

Table 1: Latency of a single Kalman state updater, in three configurations of clock frequency and FPGA choice.

Recent improvements have been made to the Kalman Filter firmware implementation. Corrections for non-radial endcap strips improve performance at high p_{T}. A simple implementation of uncertainties due to multiple scattering have been is added, whereby the hit errors in ϕ are inflated by $0.75\,\mathrm{mrad}/p_{\mathrm{T}}$. The off-diagonal terms in the higher order circle expansion terms are replaced by constant shifts in ϕ. These constants can be read from the tracklet seed, and give performance approximately equivalent to the full simulation, without significantly increasing FPGA resource utilisation. A five-parameter configuration of the fit has also been developed, which includes the transverse impact parameter (d_0). The KF is capable of fitting between four and six stubs per track.

Object	DSPs	BRAM (36 Kb)
4 parameter state updater	52	1.5
5 parameter state updater	67	2.0
HDL control-flow	1	14.5

Table 2: Resource utilisation of the Kalman Filter.

Each Kalman Filter worker is independent, and no more than 18 are expected to be required per TFP. In total, this implementation uses 13% BRAM, 3% LUTs, and 17% DSPs of a KU115 FPGA for the four parameter option, and runs at 320 MHz.

5 Performance Results

In simulations of top quark pairs with 200 superimposed pileup events, an average track finding efficiency of about 95% has been accomplished for tracks with p_{T} above 2 GeV. This is shown in Figure 5. With this same sample, an average of 60 (200) tracks are found above 3 (2) GeV, per bunch crossing. The distribution per bunch crossing is shown in Figure 7. Of the tracks found, 11% are 'fake' or incorrectly reconstructed, and about 4% are duplicates. Figure 6 shows the tracking efficiency for leptons. The efficiency for muon track finding is in excess of 97% above a p_{T} of 2 GeV. Above 10 GeV, the efficiency for electron reconstruction is about 90%.

As can be seen in Figure 6, the track finding algorithm has been demonstrated to work well up to 300 pileup, showing significant margin for scenarios in which the HL-LHC delivers higher than expected luminosity.

6 Displaced Track Finding

A modification of the hybrid algorithm is being developed to allow for displaced track finding for $d_0 < 10\,\mathrm{cm}$ [15]. As the interaction-point constraint can no longer be applied, tracklets must instead be formed from triplets of stubs. The following triplet combinations are being considered: L2L3L4, L2L3D1, L3L5L6, L2D1D2. These seeding layers would be run in parallel to those described in Section 4, however, in this scenario some of the prompt seeding combinations may be made redundant, and could therefore be removed with optimisation.

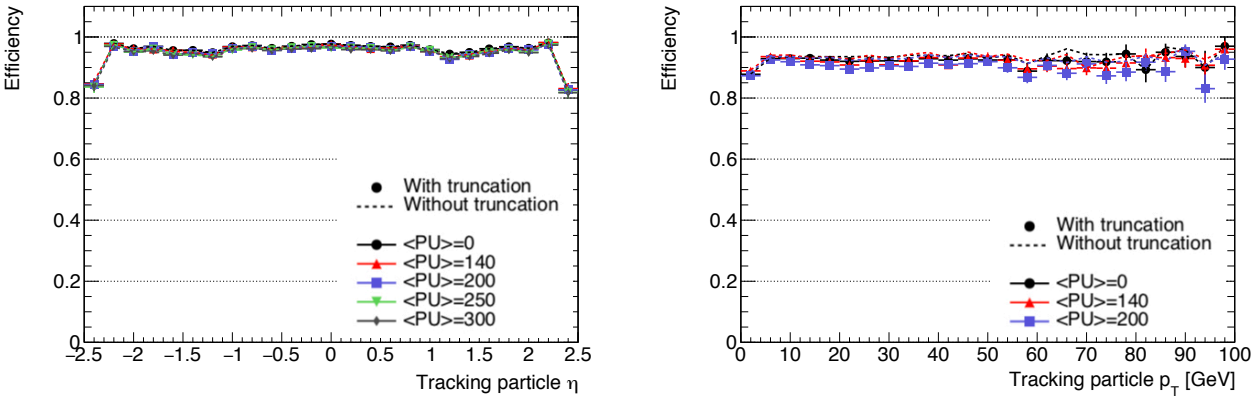

Figure 5: Track finding efficiency against particle η for particles with p_{T} above 2 GeV, produced in top quark pair production events, in simulated conditions of 0, 140, 200, 250 and 300 pileup. (left); Track finding efficiency against particle p_{T}, for particles produced in simulated top quark pair production events, in conditions of 0, 140, and 200 pileup (right). The efficiency is shown including data loss due to the fixed latency cut-off (with truncation), and without such effects (without truncation).

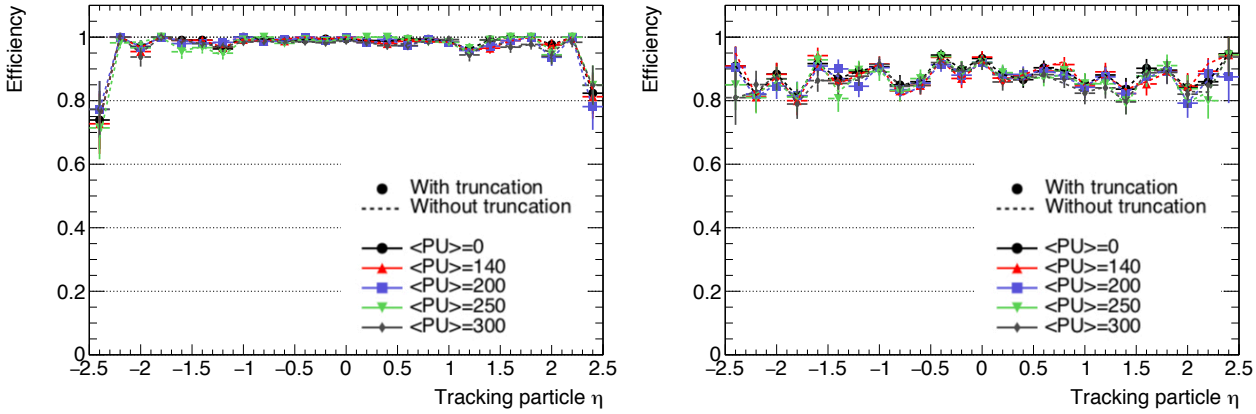

Figure 6: Track finding efficiency against particle η for muons (left), and electrons (right). Particles with a p_{T} above 2 GeV are included, in simulated conditions of 0, 140, 200, 250 and 300 pileup. The efficiency is shown including data loss due to the fixed latency cut-off (with truncation), and without such effects (without truncation).

A rate increase of 1.2 (1.4) times is observed when running with displaced seeding, with respect to prompt seeding only, followed by the 5 parameter (4 parameter) Kalman filter.

7 Conclusions

In order to maintain physics performance under HL-LHC conditions, CMS requires tracking at Level-1 of the triggering chain. Flexible and scalable track-finder and track-fitting algorithms, running on FPGA devices have been developed, and have been operated successfully in currently available hardware.

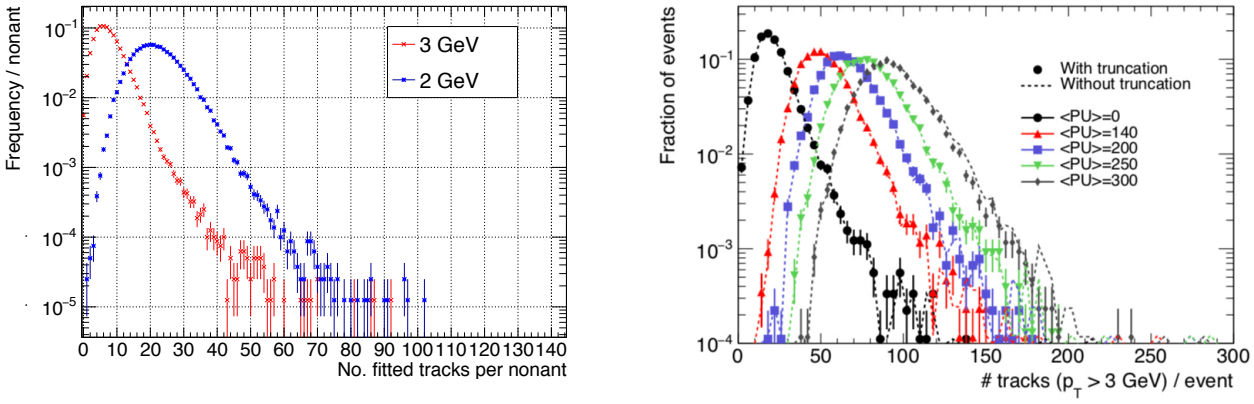

Figure 7: Track finding rates per nonant, per bunch crossing, for tracking in top quark pair production events under conditions of 200 pileup (left). Track finding rates per bunch crossing, for tracking above 3 GeV (right) in top quark pair production events under conditions of 0, 140, 200, 250 and 300 pileup (right). The rate is shown including data loss due to the fixed latency cut-off (with truncation), and without such effects (without truncation).

ACKNOWLEDGEMENTS

The author is supported by the Science & Technology Facilities Council (STFC). This work was done within the context of the CMS Collaboration.

References

[1] CMS Collaboration, "The CMS Experiment at the CERN LHC," JINST **3**, S08003 (2008). [doi: 10.1088/1748-0221/3/08/S08004].

[2] L. Evans et al., "LHC Machine," JINST **3**, (2008). [doi:10.1099/1748/3/08/S08001].

[3] CMS Collaboration, "The CMS Trigger System," JINST **12**, P01020 (2017) [arXiv:1609.02366v2].

[4] G. Apollinari et al., "High-Luminosity Large Hadron Collider (HL-LHC): preliminary design report," CERN-2015-005, (2015). [doi:10.5170/CERN-2015-005].

[5] CMS Collaboration, "Technical Proposal for the Phase-II Upgrade of the CMS Detector," CERN-LHCC-2015-10, (2015). [http://inspirehep.net/record/1614097].

[6] CMS Collaboration, "The Phase-2 Upgrade of the CMS Tracker," CERN-LHCC-2017-009, (2017) [https://cds.cern.ch/record/2272264].

[7] CMS Collaboration, "The Phase-2 Upgrade of the CMS L1 Trigger Interim Technical Design Report," CERN-LHCC-2017-013, (2017). [https://cds.cern.ch/record/2283192].

[8] B. Kreis for the CMS Collaboration, "Particle Flow and PUPPI in the Level-1 Trigger at CMS for the HL-LHC," Proceedings of Connecting the Dots, (2018) [arXiv:1808.02094v1].

[9] Xilinx Inc., "UltraScale Architecture and Product Data Sheet: Overview v2.11," (2017). [https://www.xilinx.com/support/documentation/data_sheets/ds890-ultrascale-overview.pdf].

[10] G. Hall, "A time-multiplexed track-trigger for the CMS HL-LHC upgrade," Nucl. Inst. and Meth. A **824**, 292-295 (2016) [doi:10.1016/j.nima.2015.09.075].

[11] PICMG, "Advanced Mezzanine Card Short Form Specification," (2006). [http://www.picmg.org/pdf/AMC.0_R2.0_Short_Form.pdf].

[12] R. Aggleton et al., "An FPGA Based Track Finder for the L1 Trigger of the CMS Experiment at the High Luminosity LHC," JINST **12**, P12019 (2017) [http://inspirehep.net/record/1643724].

[13] T. James, "A Hardware Track-Trigger for CMS at the High Luminosity LHC," CERN-THESIS-2018-241, (2018) [https://cds.cern.ch/record/2647214].

[14] E. Bartz et al., "FPGA-Based Tracklet Approach to Level-1 Track Finding at CMS for the HL-LHC," EPJ Web of Conferences **150**, 00016 (2017) [arXiv:1706.09225v1].

[15] Y. Gershtein, "CMS Hardware Track Trigger: New Opportunities for Long-Lived Particle Searches at the HL-LHC," Phys. Rev. D **96**, 035027 (2017) [http://cds.cern.ch/record/2647987].