Proceedings of CTD 2020
PROC-CTD2020-17
August 9, 2020

Parallelizable Track Pattern Recognition in High-Luminosity LHC

Philip Chang, Slava Krutelyov, Avi Yagil, Balaji Venkat Sathia Narayanan, Mario Masciovecchio, Matevz Tadel

University of California San Diego, CA, USA

ABSTRACT

The high instantaneous luminosity conditions in the High Luminosity Large Hadron Collider pose major computational challenges for the collider experiments. One of the most computationally challenging components is the reconstruction of charged-particle tracks. In order to efficiently operate under these conditions, it is crucial that we explore new and faster methods or implementations of charged-particle track reconstruction than what is being used today. Kalman-filter-based methods of the track pattern recognition that are currently used in the LHC experiments are inherently sequential and iterative and therefore cannot easily be accelerated through parallelization or vectorization by modern processors, such as graphics processing units or multicore processors. There have been attempts with great effort in vectorizing Kalman-filter-based methods of the track pattern recognition on modern processors with success. In this work, we instead start with a segment-linking-based algorithm that can be naturally parallelized and vectorized and is expected to run efficiently on modern processors. We established a preliminary segment-linking-based track pattern recognition for the CMS experiment using the Phase-II outer tracker and our findings and implications are presented here. This work is building on experience gained from a prototype of a similar approach studied in a different tracker layout geometry based on ideal detector simulation previously presented at CHEP2016.

PRESENTED AT

Connecting the Dots Workshop (CTD 2020)
April 20-30, 2020

1 Introduction

The reconstruction of charged-particle tracks is crucial for the experiments at the Large Hadron Collider (LHC) of CERN. Reconstruction of many objects, such as vertex, lepton, and jet reconstruction, as well as the b tagging, and the missing transverse momentum calculation all rely on the charged particle track reconstruction. Among the reconstruction steps, tracking is most time-consuming and its time scales exponentially with the multiplicity of the hits in the tracker. This leads to a computing challenge for the track reconstruction at the planned High-Luminosity LHC (HL-LHC) era, where the number of overlapping proton-proton (pp) collisions (pileup) will increase by severalfold. Therefore it is crucial to investigate ways to speed up the track reconstruction algorithms in order not to suffer loss in tracking efficiency negatively impacting physics results of the experiments.

The industry is providing modern processors that utilize parallelism more and more to provide higher computing power to the consumer. To address the tracking challenge of the HL-LHC it is therefore pertinent to investigate algorithms that can exploit parallelism to best leverage the gains from the industry.

Several novel algorithms are being developed for the HL-LHC environment with a common theme of exploiting parallelism. Patatrack uses a cellular automata-based tracking algorithm on graphic processor units (GPUs) and builds pixel track seeds in parallel [1]. Exa.TrkX uses graph neural network-based algorithm for track building, which can easily be computed in parallel through vectorizing matrix multiplication computations [2]. Even the inherently sequential Kalman filter-based tracking algorithm is being vectorized by seeds, events, and subregions of the detector in the mkFit project to exploit the parallelism of modern architectures [3].

In this work, we present an algorithm approach that is inherently parallel thereby enabling opportunity for efficient tracking on modern architectures. The presented algorithm has a root in the segment linking algorithm used in the central outer tracker of the CDF experiment. Previously a similar algorithm was explored in a different context of exploring grouped layer tracker layout design in Ref. [4, 5], which showed a promising result of being able to reduce the combinatorics drastically with the additional benefit of tracking algorithm being inherently parallel. Adopting from previous work in Ref. [4, 5], the presented algorithm builds mini-doublets by associating hits left by the charged particle in the detector with each other and recursively associates pair of mini-doublets to tracklets and pair of tracklets to track candidates. Each step of the algorithm presented here only requires information of immediate neighbors and therefore can be run in multiple threads in parallel with relative ease. As a proof of concept, the preliminary algorithm is being developed in the context of CMS Collaboration's HL-LHC outer tracker geometry [6], which is particularly suitable for applying such a parallel algorithm. We show a case study demonstrating the proof of concept of how such an algorithm would work in HL-LHC environment.

2 Segment linking in CMS HL-LHC Outer Tracker

CMS Collaboration is upgrading its tracker for HL-LHC [6]. The proposed design contains double-layered silicon modules with a gap size of $\approx 2\,\mathrm{mm}$ dubbed p_T *modules* in the outer tracker that can reduce occupancy of hits by requiring coincidences of hits between two layers consistent with a minimum p_T threshold. This is illustrated in Figure 1. This design choice was originally driven by enabling hardware-based track trigger capability for tracks with minimum p_T of $2\,\mathrm{GeV}$ [6]. However, as the p_T modules look for coincidences of hits in each module separately it also enables a unique opportunity to apply a parallel tracking algorithm.

The main tracking algorithm steps through several stages of building pieces that tracks consist of. First, the algorithm builds the mini-doublets in each p_T module by associating hits in each layer consistent with the minimum p_T threshold of $1\,\mathrm{GeV}$. Then the mini-doublets in the neighboring modules on separate layers are linked together to form segments. Neighboring segments on distinct layer pairs are then linked together by applying a geometrical requirement that the pair of segments be consistent with a track hypothesis. The linked segments are dubbed tracklets. The segment linking can also occur skipping a layer to allow for missing hits by building tracklets with gaps. Figure 2 illustrates various objects: mini-doublets, segments, tracklets, and track candidates.

With various mini-doublets, segments, and tracklets built, one can build various kinds of track candidates

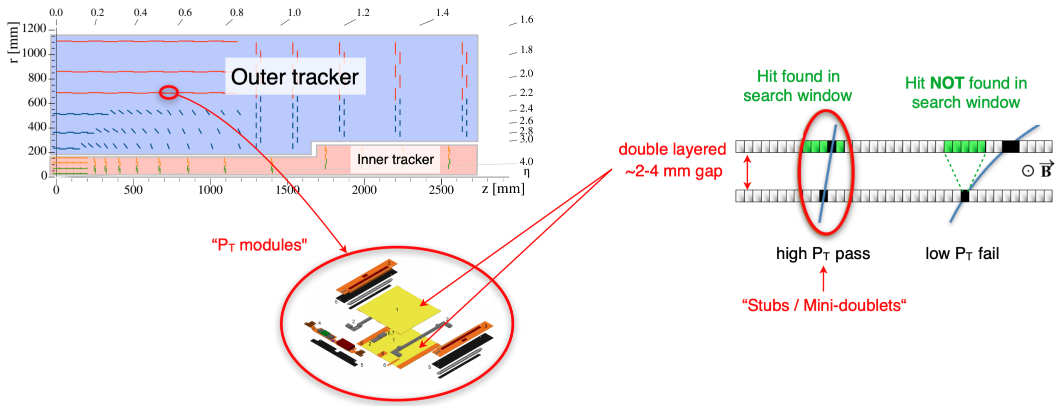

Figure 1: Proposed CMS tracker design for high luminosity LHC (HL-LHC) [6]. The p_T module concept is illustrated here. The double layered modules have a gap of size $\approx 2\,\mathrm{mm}$ and the coincidences of hits consistent with a track above a minimum p_T threshold can be applied to reduce occupancies.

by putting together the objects. For example, if no hits are lost for a given charged-particle trajectory, one can put together three segments across the six layers of the outer tracker to form a track candidate. If one or more hit is lost, then a linked segment with gaps can be used to form a track candidate. The seeds from inner pixel detectors can be used as “segments” as well to build track candidates but are not a requirement. Figure 2 illustrates the different kinds of track candidates that can be built from this algorithmic approach.

The benefits of such a segment linking algorithm in the outer tracker are apparent. The algorithm is easily parallelizable as the subroutines of the algorithm only require information from their immediate neighbors. Many tracking algorithms widely used today rely heavily on the inner tracker and its health, while the proposed algorithm does not. algorithm does not depend on the inner tracker or its health, while many tracking algorithms widely used today do. This also means that more extensive displaced tracking will come naturally.

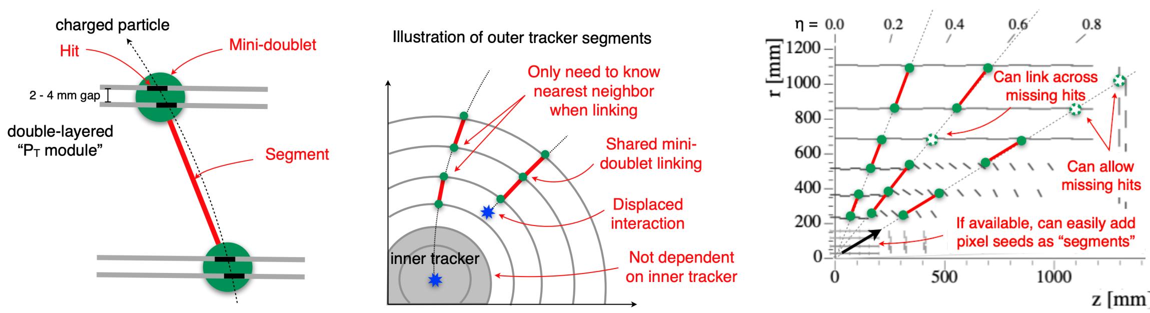

Figure 2: Illustrations of the segment linking algorithm in the outer tracker. (left) Illustration of mini-doublets and segments. (center) Illustration of tracker segments in the outer tracker and their potential benefits. (right) Illustration of non-exhaustive list of possible track candidates that can be built from the presented algorithm.

3 Case study: building barrel tracks via segment linking

As a start, we focus on track candidate reconstruction with no missing hits in the barrel outer tracker. There are a total of 6 layers in the barrel outer tracker with each module in each layer having two hits leading to a total of 12 hits in the barrel outer tracker. As mentioned before this kind of track candidate has the benefit of not relying on pixel track seeds. We will present the algorithm targeting these tracks and discuss the efficiency of the algorithm.

3.1 Mini-doublet building and efficiency

The first step is to build the mini-doublets using the p_T module concept. The target threshold is 1 GeV. In each module, the hits are associated and checked whether there is a pair of hits consistent with the minimum p_T threshold of 1 GeV. If they pass the requirements the pairs of hits are called mini-doublets. The efficiency of the mini-doublet for the innermost layer and the outermost layer is shown in Figure 3. The efficiency is defined to be the ratio of number of denominator tracks with a mini-doublet in a given layer reconstructed to the number of denominator tracks that leave all 12 hits in the barrel region. Figure 3 shows full efficiency soon after the targeted 1 GeV threshold. This mini-doublet formation is very important in reducing combinatorics. Most hits in the tracker come from low pt tracks (i.e. < 1 GeV tracks) and by forming mini-doublets one effectively filters out hits from low p_T tracks which are not targeted by this algorithm and only increase combinatorial background. The effect can be seen from the Table 1 where the multiplicities of hits from a representative pileup 200 $t\bar{t}$ event in each layer are reduced by factor 3 to 7.

Layer	1	2	3	4	5	6
Number of hits	36k	28k	21k	17k	12k	6k
Number of mini-doublets	5.9k	3.8k	3.1k	3.7k	3.3k	2.2k
Ratio	6.1	7.3	6.8	4.6	3.5	2.7

Table 1: Number of hits and mini-doublets in each layer of the CMS HL-LHC outer tracker from a representative sample of pileup 200 $t\bar{t}$ events.

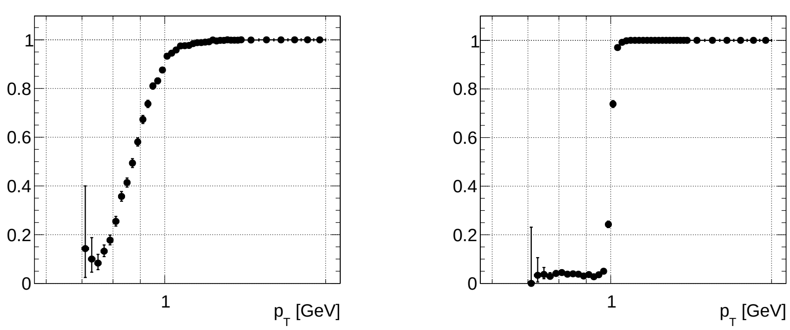

Figure 3: Mini-doublet efficiency as a function of p_T. Left is the mini-doublet efficiency for the innermost layer, and the right is the mini-doublet efficiency for the outermost layer.

3.2 Segment building and efficiency

The next step is to build the segments by associating two mini-doublets. To reduce combinatorics in associating the mini-doublets a module map is defined from a large number of simulated muon gun events. The module map is defined such that for a given reference module in a lower layer a list of compatible modules from a neighboring layer is assigned. When creating segments a mini-doublet in a reference module is only compared against mini-doublets from the compatible modules. The angles in the mini-doublets between the two hits and the angle between the mini-doublets are required to be consistent with a track hypothesis. The efficiency of the segment is shown in Figure 4. The efficiency is defined as the ratio of number of denominator tracks with a segment in a given combination of layers reconstructed to the number of denominator tracks that leave all 12 hits in the barrel region. Soon after the targeted minimum p_T threshold, a good efficiency is achieved.

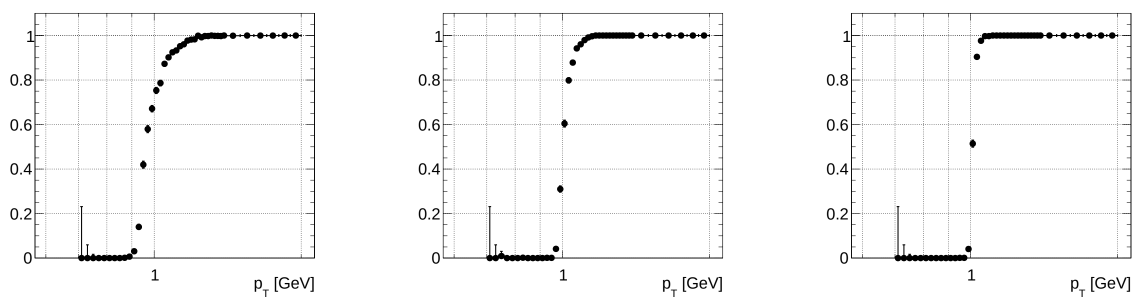

Figure 4: Segment efficiency as a function of p_T. (Left) Efficiency of segments formed by mini-doublets in layer 1 and 2 from the algorithm. (Center) Efficiency of segments formed by mini-doublets in layer 3 and 4 from the algorithm. (Right) Efficiency of segments formed by mini-doublets in layer 5 and 6 from the algorithm.

3.3 Segment linking (tracklet building) and efficiency

The next step is to build the tracklets by linking two segments. The same module maps used in the segment building are used to determine which segments to attempt to link. This is done by checking the compatibility between the outer module of the inner segment to the inner module of the outer segment of the two segments being linked. If these modules are deemed compatible then various geometrical constraints are checked. If the two segments pass all the requirements then the two segments are considered linked and become a tracklet. The geometrical constraints include the β angles the segments make against the chord drawn in $r - \phi$ plane between the innermost mini-doublet to the outermost mini-doublet and a constraint in $r - z$ plane between the segments and a few more angle compatibility requirements. If the two segments at hand are segments belonging to a single true charged particle track, the $\Delta\beta$ between the β angles will take a value close to zero, while for a pair of segments stemming from combinatorial backgrounds the value of the $\Delta\beta$ will not have a preferred value. This is illustrated in Figure 5 where the peak is visible above the flat combinatorial background from a representative pileup 200 $t\bar{t}$ events. The efficiency for the tracklets are reported for muon gun events, and also for pions and electrons from the pileup 200 $t\bar{t}$ events in Figure 6.

3.4 Track candidate building and efficiency

The track candidate with all hits in six layers for the barrel outer tracker can be simply built by putting together the tracklets from layer 1 to 4 with tracklets from layer 3 to 6 that have a common segment in layers 3-4. The case study presented here focuses on tracks with no hits missing. Other types of track candidates that allow for missing hits will be studied in the future. The efficiency of track candidates with no missing hits are reported for muon gun events, and also for pions and electrons from the pileup 200 $t\bar{t}$ events in Figure 7.

4 Conclusion and outlook

The reconstruction of charged-particle tracks is crucial for the LHC experiments to be successful. In the planned HL-LHC era, the increase in pileup will bring a computational challenge to tracking. To exploit the gains from the industry, we presented here a naturally parallelizable algorithm for the HL-LHC in the

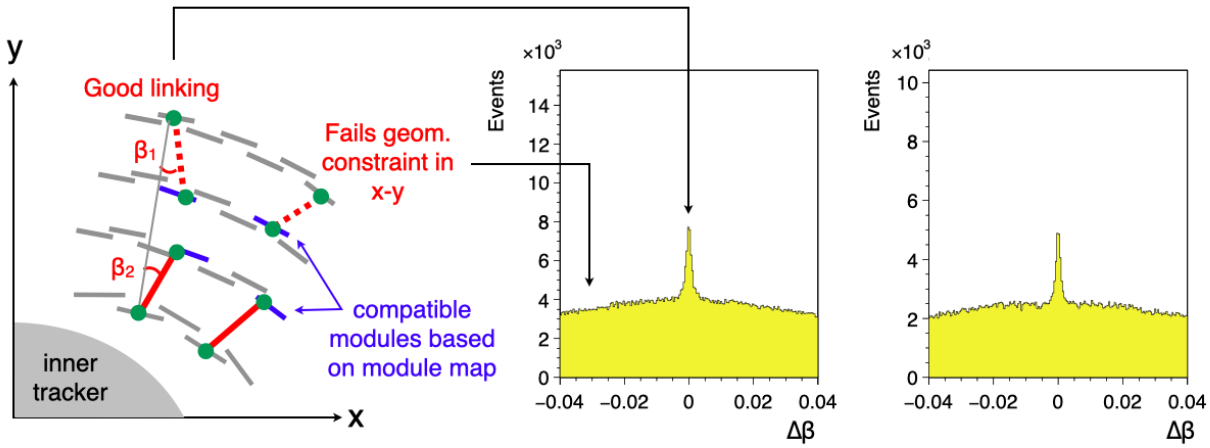

Figure 5: $\Delta\beta$ distribution of tracklets in the barrel outer tracker. (Left) Illustration of how a good and bad segment linking looks like (Center) $\Delta\beta$ distribution for segment linking between segments in layer 1-2 to layer 3-4. (Right) $\Delta\beta$ distribution for segment linking between segments in layer 2-3 to layer 4-5. The $\Delta\beta$ value of correct pairing results in a peak near zero, while the combinatorial background from bad linking is uniformly distributed.

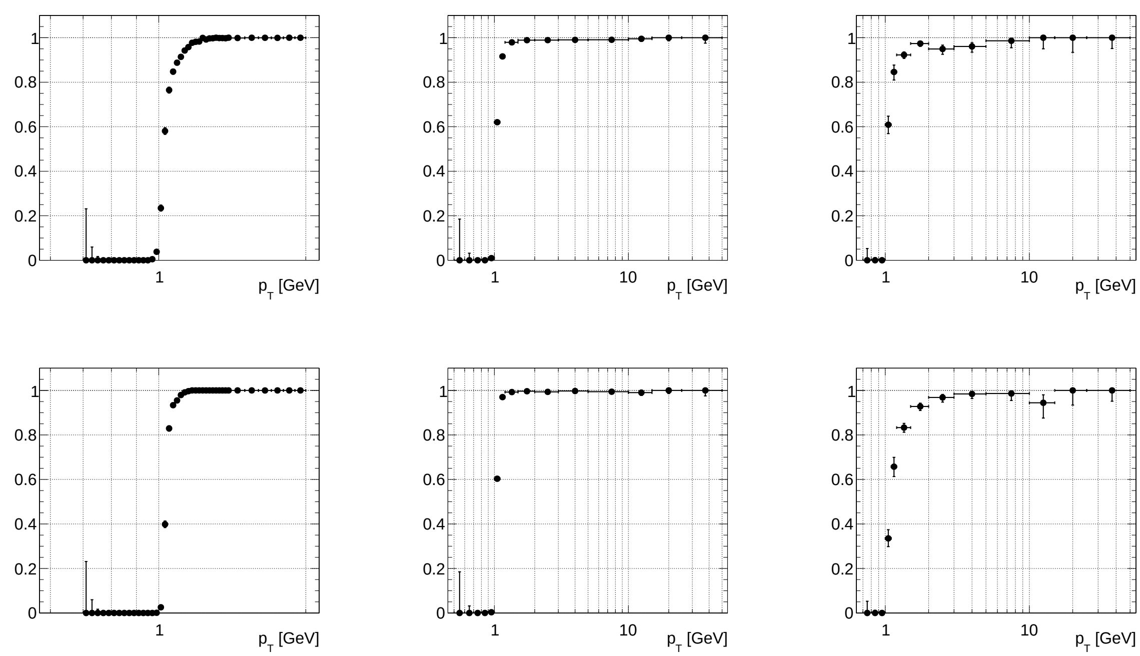

Figure 6: Tracklets efficiency as a function of p_T. The first (second) row shows the tracklet efficiencies for tracklets formed from layer 1-2-3-4 (3-4-5-6) for (left) muon-gun events, (center) pions, (right) electrons from a representative sample of pileup 200 $t\bar{t}$ events.

context of CMS experiment. This algorithm links hits to segments, segments to tracklets, and ultimately builds track candidates, where each step can be executed in parallel. The specific algorithm presented here is applied to the outer tracker of the CMS tracker for the HL-LHC. The benefit from building track candidates

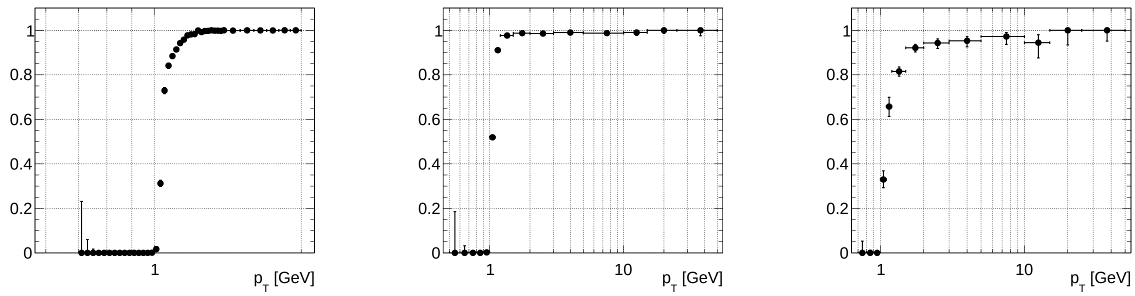

Figure 7: Efficiency of track candidates without missing hits as a function of p_T.

with the outer tracker includes complementarity to the inner tracker seeded tracking and a natural extension to displaced tracks among others. As a case study, track candidates with no missing hits in the barrel outer tracker have been studied. The study targeted tracks down to 1 GeV and shows that a good efficiency can be achieved. For future, the performance of the algorithm in the endcaps will be checked as well as building different kinds of track candidates that can allow missing hits and incorporate pixel seed tracks and extend the algorithm to cover wider phase-space.

References

[1] CMS Collaboration, "Patatrack Heterogeneous Computing 2018 Demonstrator: Pixel Tracks", CMS–DP–2018–059 https://cds.cern.ch/record/2646774.

[2] X. Ju et al., "Graph Neural Networks for Particle Reconstruction in High Energy Physics detectors", in *33rd Annual Conference on Neural Information Processing Systems*. 3, 2020. [arXiv:2003.11603].

[3] G. Cerati et al., "Speeding up Particle Track Reconstruction in the CMS Detector using a Vectorized and Parallelized Kalman Filter Algorithm", in *Connecting the Dots and Workshop on Intelligent Trackers*. 6, 2019. [arXiv:1906.11744].

[4] V. Krutelyov et al., "Impact of tracker layout on track reconstruction with high pileup", *J. Phys. Conf. Ser.* **898** (2017), no. 4, 042023.

[5] V. Krutelyov et al., "Impact of tracker layout and algorithmic choices on cost of computing at high pileup", *PoS* **ICHEP2016** (2016) 194.

[6] CMS Collaboration, "The Phase-2 Upgrade of the CMS Tracker", CERN–LHCC–2017–009, CMS–TDR–014.