This template serves as a guide with the correct format and content to write a manuscript for submission to the *Journal of Visualized Experiments*. You may begin writing directly in this document. Guiding notes are provided in **grey** throughout this document; *please remove these notes before submission.*

Please strictly follow the guidelines in our [Manuscript Instructions for Authors](http://www.jove.com/files/Instructions_for_Authors.pdf) document. Failure to follow these guidelines will delay the editorial and peer review process and extend your publication turnaround time. Should you have any questions about writing your manuscript, please contact your editor, or JoVE’s [Editorial Office](mailto:submissions@jove.com). Step-by-step instructions to submit can be found [here](http://www.jove.com/publish/ready-to-start).

**Required formatting**

* *File Type:* Microsoft Word is JoVE’s preferred format for text. All manuscripts must be submitted as a .doc or .docx file
* *Font:* 12 pt Arial, Calibri, or Times New Roman throughout the entire document
* *Page margins:* 1 inch on all sides
* *Line spacing:* single-spaced with a space between each paragraph/bullet point

**TITLE:** Extending the Physics Reach of the LHC with Parallel Computing

**AUTHORS:** Valerie Halyo, Patrick LeGresley, Paul Lujan

Valerie Halyo

Department of Physics,

Princeton University

vhalyo@princeton.edu

Patrick LeGresley

??

Paul Lujan

Department of Physics,

Princeton University

plujan@princeton.edu

**CORRESPONDING AUTHOR:** Valerie Halyo

**KEYWORDS:** Trigger algorithms; parallel computing; vertexing algorithms; Hough transform

**SHORT ABSTRACT:** *(10 words minimum, 50 words maximum)*

At the Large Hadron Collider, the trigger is required to decide very quickly whether to keep or discard an event, limiting our ability to search for new physics. Parallel computing can enhance the capabilities of the trigger and enable us to search for new models which previously were not possible.

**LONG ABSTRACT:** *(150 words minimum, 300 words maximum)*

At the Large Hadron Collider (LHC), the trigger systems for the detectors must be able to process a very large amount of data in a very limited amount of time, so that the nominal collision rate of 40 MHz can be reduced to a data rate that can be stored and processed in a reasonable amount of time. This need for high performance places very stringent requirements on the complexity of the algorithms that can be used for identifying events of interest in the trigger system, which potentially limits the ability to trigger on signatures of various new physics models. In this paper, we present an alternative tracking algorithm, based on the Hough transform, which avoids many of the problems associated with the standard combinatorial track finding currently used. The Hough transform is also well-adapted for parallel computing using Graphics Processing Unit (GPU) or Many Integrated Core (MIC) architecture, and such systems could be easily integrated into the existing High-Level Trigger (HLT). This algorithm offers the ability to trigger on topological signatures of new physics currently not practical to reconstruct, such as events with jets or black holes significantly displaced from the primary vertex. We show the performance of the Hough transform algorithm on models with displaced tracks, and consider a specific model of new physics which is not currently accessible at the LHC to show the considerable advantages that can be realized using this new algorithm.

**INTRODUCTION:** *(150 words minimum, 1500 words maximum)*

The successful discovery of the Higgs boson at the Large Hadron Collider (LHC) by the ATLAS and CMS experiments has been a triumph for the Standard Model (SM) of particle physics. As we begin to explore the ramifications of this discovery, there is great potential for discovery of new physics by considering scenarios in which the Higgs decays into new, previously-undiscovered particles or exhibits other behavior beyond the SM. Although the current measurements show no signs of deviating from the SM predictions, many channels remain unexplored, and many possibilities remain for new physics to be discovered in future LHC data.

It is thus important that we make the most out of the abilities of the LHC to discover this potential new physics. One particularly critical component of a modern particle physics experiment such as ATLAS or CMS is the trigger and data acquisition (DAQ) system, which is responsible for selecting events of interest from the raw collision data. As the collisions occur at a rate of 40 MHz and events can only be stored and processed at a rate of ~100 Hz, the trigger must thus quickly and efficiently select events while rejecting background at a rate of at least 400,000 : 1. At both CMS and ATLAS, this is obtained by implementing a multi-level hierarchical trigger system. For example, at CMS, the first level, Level 1 (L1), is implemented in hardware and reads out only a subset of the detector to quickly make a coarse decision and reduce the incoming rate to ~100 kHz. This is then followed by the High-Level Trigger (HLT), which fully reconstructs the event to perform a more complete analysis and make a final decision.

Because of the intense computing demands in the trigger, certain assumptions must be made in order to process events within the available time. These assumptions mean that the event reconstruction is not necessarily perfect, which limits our ability to search for new physics. Extending the capabilities of the trigger so that it could reconstruct events better would thus allow us to identify and analyze events from new physics models that are currently not possible to consider at the LHC.

One particularly striking signature of new physics would be the observation of a large number of tracks in the detector originating at a vertex significantly displaced from the original collision. Such an event would be a clear signal of physics beyond the Standard Model. A wide variety of theoretical models predict such events: for instance, a “hidden valley” model in which a Higgs boson decays to a new long-lived particle; production and decay of microscopic black holes; supersymmetric (SUSY) models with long-lived neutralinos; or “boosted jet” scenarios. Unfortunately, the current trigger at CMS is ill-adapted to observe such events, as it is not designed for reconstruction of such displaced tracks. Thus it is possible that with the current detector abilities, we would miss such events and hence the signature of new physics.

In this paper, we propose a new technique for integrating parallel computing, in the form of Graphics Processing Unit (GPU) or Many Integrated Core (MIC) architecture, into the trigger system. Because the current HLT system is built from ordinary commodity PC hardware, it is amenable to adding new hardware in the form of coprocessors. This would increase the computing power of the trigger by not only speeding up the existing algorithms in the trigger, but also by allowing us to introduce entirely new algorithms which could reconstruct events such as those containing displaced tracks more efficiently and accurately. Such an addition would thus allow us to search for new physics at the LHC in ways not previously possible.

**PROTOCOL:**

In this section, we discuss the current experimental configuration of the CMS tracker and trigger; for illustrative purposes, we use CMS as an example, but the principles are the same between the CMS and ATLAS experiments, and the results obtained should be applicable to either.

Both the CMS and ATLAS inner trackers include a silicon pixel detector and a silicon strip detector. All tracker layers provide two-dimensional hit position measurements, but only the pixel tracker and a subset of the strip tracker layers provide three-dimensional hit position measurements. The CMS pixel detector includes three barrel layers and two forward disks on either end of the detector; outside the pixel detector is the strip detector, consisting of ten layers in the barrel plus three inner disks and nine forward disks at each end of the detector. Owing to the strong magnetic field and the high granularity of the silicon tracker, promptly-produced charged particles with transverse momentum pT = 100 GeV/c are reconstructed with a resolution in pT of 1.5% and in transverse impact parameter d0 of 15 mm. The track reconstruction algorithms are able to reconstruct displaced tracks with transverse impact parameters up to 25 cm from particles decaying up to 50 cm from the beam line. The performance of the track reconstruction algorithms has been studied with data. The silicon tracker is also used to reconstruct the primary vertex position with a precision of 20 mm in each dimension.

The CMS track reconstruction, known as the Combinatorial Track Finder (CTF), uses an iterative algorithm, with earlier iterations searching for tracks that are more easily found (relatively higher pT tracks close to the interaction region). The hits in these tracks are then removed, allowing later iterations to search for lower momentum or highly displaced tracks without the number of possible combinations becoming too large. Each iteration consists of four steps: a seeding step, a track finding step, a track fitting step, and a selection step. In the first step, a seed, consisting of three hits or two and a primary vertex constraint, is constructed to create an initial estimate of the trajectory parameters for the track. Second, the track finding is performed using a global Kalman filter. This step performs a fast propagation of the track candidate to propagate the track through the layers of the detector, search for compatible nearby hits, and attach them to the candidate. After all candidates in this step have been found, the track candidate collection is then cleaned to remove duplicate tracks or tracks which share a large number of hits. The third step is to perform a full Kalman fit over the whole track to obtain the best estimate of the track parameters at all points along the trajectory. The filter begins at the innermost hits, and then iterates outward through each hit to update the track trajectory estimate and its uncertainty. After this first fit is complete, a smoothing stage is then performed running backwards from the last hit to apply the information from the later hits to the earlier ones. This step uses a Runge-Kutta propagator to account for the effect of material interactions and an inhomogeneous magnetic field. Finally, track selection requirements are applied to reduce the fake rate of the resulting track candidates.

As a demonstration of how computationally intensive 2D tracking algorithms can take advantage of massively parallel GPU processing, the authors developed an implementation of the Hough transform using CUDA in order to measure in real time the transverse momentum of prompt or non-prompt tracks. The Hough transform is an image processing algorithm for feature detection that considers all possible instances of a parameterized feature such as a line or circle. Each possible instance of a feature starts with zero votes in the parameter space, and then for each piece of input data votes are added to the feature instances that would include that input data. After all input data has been processed the votes in the parameter space are processed. Locations in the parameter space with more votes are likely to be actual features in the input data so this step amounts to looking for local maxima in the parameter space. Once candidate features have been identified, more expensive computations can be applied to confirm the existence of the feature. One should note that the Hough transform approach as presented above is computationally expensive. Hence, simplification of the standard Hough transform method had to be used in the past for the purpose of track finding in high energy physics.

One clear advantage of the Hough transform over the combinatorial track finding algorithms that are currently standard is that the execution time is linear with respect to the number of hits present in the event, while combinatorial algorithms, as the number of combinations increases much more rapidly with respect to the number of hits, show a worse dependence on the number of hits. In addition, the Hough transform is naturally tolerant of missing hits or hits that do not exactly fit the candidate features, due to limited resolution or the discretization used in computing the parameter space. While the Hough transform is more expensive to implement for a small number of hits, the ability to take advantage of multithreading offers the chance of significantly improved performance for events with a large number of tracks.

One should note that this paper does not imply that executing machine vision and pattern recognition algorithms is necessarily more appropriate for parallel processing compared to the conventional combinatorial algorithms used in high energy physics. Rather, it is a complementary way that was first studied by the authors due to the similarity of the problem with computer image analysis. These techniques have proven successful in image processing using the Hough transform on GPUs.

In our studies, the sample input data was generated using a Monte Carlo simulation of a simple detector model where only the transverse plane is considered. The model contains a simulated beam pipe with a radius of 3.0 cm surrounded by ten concentric, evenly-spaced tracking layers with an overall radius of 110.0 cm. A hit resolution of 0.4 mm in each direction is used, corresponding to a relative pT resolution of approximately 7% at 100 GeV=c. In order to obtain a relative momentum resolution similar to the one obtained using current HEP tracking, one would need to approximately double the time performance results so far obtained.

The code for performing the Hough transform was implemented and optimized on three different platforms: a standard Intel CPU architecture, tested on a quad-core Intel i7-3770 (Ivy Bridge) CPU, and also with a two-socket Intel Xeon E5-2697v2 configuration, with 12 cores per socket clocked at 2.7 GHz (up to a turbo frequency of 3.5 GHz); the NVIDIA Tesla architecture, tested on a Tesla K20c, which contains 2496 cores clocked at 706 MHz; and the Intel Xeon Phi coprocessor architecture, tested on a Xeon Phi of the QS-7120P series, with 61 cores at 1.33 GHz.

The ability to handle displaced vertices provides the opportunity to identify interesting new physics such as non-prompt jets and black holes. As an example, we consider a simulated set of tracks making up a jet. The Hough transform is applied to the corresponding hits making up these tracks to generate the parameter space. Recall that the parameter space identifies candidate tracks as locations where many sinusoids cross one another. When the parameter space is viewed, the locations of these crossings are visually apparent as dark spots. However, the analysis is taken one step further by noting that the crossings all lie along a single sinusoidal curve in the parameter space. This single sinusoidal curve can be isolated and is significant because that one sinusoidal curve in the parameter space corresponds to a point in the original x-y space. More importantly, that point in the original x-y space corresponds to the vertex of the displaced jet. Interestingly, the most convenient way to implement this process is to use the Hough transform a second time, this time applying it not to the hits in the x-y space but the crossings in the parameter space. Using the appropriate parameterization to look for sinusoidal curves instead of straight lines, the curve is identified and the vertex of the jet is now located. One can apply this technique to identify any number of displaced jets in a single event. As previously mentioned it is the number of parameters, and not the details of the particular parameterization, that determines the computational cost of the Hough transform. Therefore, compared to the performance results for a single Hough transform, the jet finding has double the computational cost because of the need to perform two transforms.

To illustrate the increased physics power afforded by these new capabilities, we consider a particular model with a new, long-lived boson X, which is produced by the decay of a Higgs boson and in turn decays to two b quarks. The lifetime of the X boson is such that the resulting jets are produced still within the CMS tracker, but significantly displaced from the primary interaction. For such a model, the current CMS trigger is highly inefficient, with efficiencies typically on the order of 1%. This makes observation of such a model extremely unlikely even with the increased luminosity expected in Run II at the LHC. In contrast, a Hough-transform-based algorithm exhibits an efficiency of up to 85%, making observation of such new models feasible in less than a year of data collected in Run II.

**REPRESENTATIVE RESULTS:**

Figure 1 shows the operation of the Hough transform algorithm for a simple case with 500 curved tracks. First, the tracks are constructed and the resulting hits in the detector are simulated, as shown in Figure 1(a). Using these hits, the parameter space in Figure 1(b) is constructed. In this case, the parameter space is two-dimensional, corresponding to the two track parameters: curvature ρ and angle φ. Each hit in Figure 1(a) corresponds to a curve in Figure 1(b). When integrated over the entire data set, peaks appear in parameter space at the values corresponding to the actual, physical trajectories; the final reconstructed tracks are shown in Figure 1(c). The tracks are reconstructed with 86% efficiency.

Figure 2 shows the resulting time performance as a function of the number of tracks in the event for four computing platforms: an Intel i7-3770 CPU, a dual-socket Intel Xeon E5-2697v2 CPU system, an NVIDIA Tesla K20c GPU, and an Intel Xeon Phi 7120P coprocessor. The reason for comparing the accelerators to a dual-socket CPU system (rather than single-socket) is that the thermal dissipation power (TDP) of the GPU and of the coprocessor is 225 and 300 W, respectively, whereas a single CPU chip is rated at 130 W. Therefore, two CPUs make for a more meaningful watt-to-watt comparison.

Figure 3 shows the Hough transform algorithm applied to a displaced jet. In this event, four displaced jets are simulated, each with a number of straight tracks originating from a single displaced vertex. The simulated tracks are shown in Figure 3(a). Using the simulated hits, the Hough transform is applied, resulting in the parameter space shown in Figure 3(b). This parameter space is difficult to interpret visually. To identify the displaced vertices in the event, a second Hough transform is applied, resulting in the parameter space in Figure 3(c). The curves in this parameter space correspond to the original displaced vertices, allowing for rapid identification of the displaced jets in the event.

Figure 4 shows the efficiency of various triggers for the particular model discussed in the text with a Higgs boson of mass 125 GeV/c2 decaying to two long-lived, neutral bosons X with mass 20 GeV/c2, where the X bosons decay in turn into a bb quark pair. The efficiency is shown as a function of the proper lifetime cτ of the X boson, varying from 10 to 600 mm. The efficiency is plotted for the current CMS L1 trigger, as well as a simulation of the current single best trigger at the CMS HLT, which identifies events containing displaced jets by requiring a large amount of total transverse energy (HT > 250 GeV) and two jets with transverse momentum pT > 60 GeV/c, with most tracks in the jet having an impact parameter greater than 300 μm and no more than 15% of the energy of the jet originating from tracks with an impact parameter less than 500 μm. Finally, the efficiency of a proposed massively parallel computing (MPC)-based trigger is shown, where we assume an efficiency of 85% for events which pass the CMS L1 trigger.

**Figure Legends:** *(provide separate files for figures and tables; do not include figures or tables within manuscript document. The default placement for all figures and results tables is below the Representative Results text. Please indicate— via brackets, i.e. [Place Figure 1 here]— if you prefer figure/table placement at another location in the text.)*

**Figure 1:** Sample application of the Hough transform for a simulation of 500 curved tracks. Figure 1(a) shows the initial simulated hits in the detector. Figure 1(b) shows the resulting parameter space after the Hough transform has been applied. Figure 1(c) shows the reconstructed curved tracks.

**Figure 2:** Performance of the Hough transform algorithm on four platforms: NVIDIA Tesla K20c GPU (red squares), Intel i7-3770 CPU (blue triangles), dual-socket Intel Xeon E5-2697v2 CPU (solid black circles), and Intel Xeon Phi 7120P coprocessor (open black circles), as a function of the number of simulated tracks in the event. A 2048x2048 grid is used in the parameter space. The error bars indicate variations between successive runs, which are negligibly small in most cases.

**Figure 3:** The Hough transform algorithm applied to an event with multiple displaced jets. Figure 3(a) shows the initial simulated tracks. Figure 3(b) shows the result of the first Hough transform. This results in a complex parameter space. Figure 3(c) shows the result of the second Hough transform. The intersections of the tracks are now identified as single curves in parameter space, which can be interpreted as the four vertices of the original displaced jets.

**Figure 4:** Trigger efficiencies for a model with h → XX, X → bb, with mH = 125 GeV/c2 and mX = 20 GeV/c2, as a function of the proper lifetime cτ of the X boson. The blue dots show the efficiency of the current CMS L1 trigger, the red dots show the efficiency of the current best individual CMS HLT trigger, and the green dots show the efficiency of a proposed trigger using massively parallel computing (MPC) with an 85% efficiency after the L1 trigger.

**DISCUSSION:** *(3-6 paragraphs)*

The results here show that new physics models which are not currently detectable at the LHC could be discovered using the enhanced capabilities of triggers which take advantage of parallel computing. The model considered is just one example of a wide class of models containing displaced tracks which are difficult to detect with the current configuration of the CMS trigger. These models could be very simply implemented with straightforward extensions of the Higgs sector and are not well-constrained by existing searches. In general, however, the presence of multiple displaced tracks in a single event would represent a “smoking gun” for new physics and so having the ability to quickly identify and preserve these events would be an invaluable tool in our searches at the LHC.

In addition, the LHC will continue to face further challenges with the start of Run 2 in 2015. The increased energy and luminosity will place ever-increasing demands on the trigger system. The addition of parallel processing can help to cope with this increased complexity, both by simply increasing the computing capabilities of the current trigger, and by enabling the implementation of algorithms better-equipped to process events with a very large number of tracks. The high luminosity expected in Run 2 will certainly mean that many of the thresholds in the current trigger will need to be raised, resulting in even lower efficiencies for the models discussed in the text and thus increasing the benefits of an alternative trigger strategy even further.

The results presented here are preliminary results designed to show the potential physics impact of a new trigger; considerable optimization has to be done to improve the performance, efficiency, and purity of the trigger. Nevertheless, adding a Hough transform-based trigger capability to the existing Combinatorial Track Finder would clearly represent a great advantage in searching for events representing a wide variety of new physics possibilities.

**ACKNOWLEDGMENTS:** This work is supported by the US Department of Energy, Office of Science Early Career Research Program under Award Number DE-SC0003925.

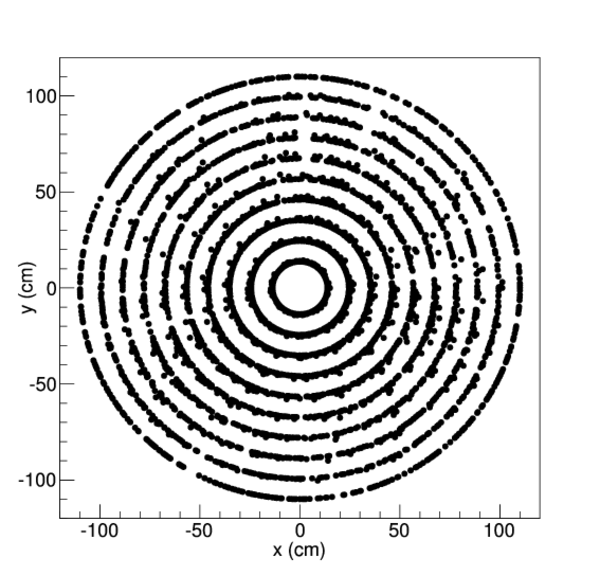
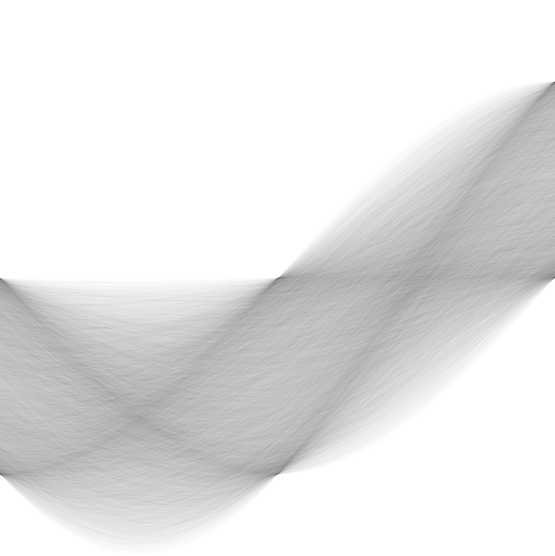
**DISCLOSURES:** The authors have nothing to disclose.

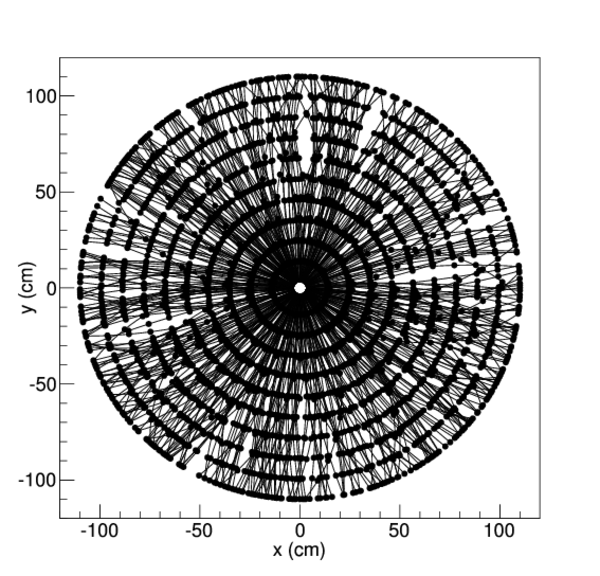
**REFERENCES** *(10 minimum— use* JoVE*’s style files to interface with reference software)*

1. Lastname, F.I., Lastname, F.I., Lastname, F.I., Lastname, F.I. Article Title. *Source*. **Volume** (issue), FirstPage – LastPage, PII, DOI, (YEAR).
2. Lastname, F.I., *et al*. Article Title. *Source*. **Volume** (issue), FirstPage – LastPage, PII, DOI, (YEAR).

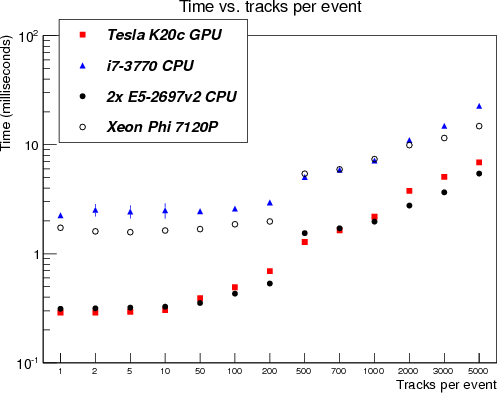
\*Reference style 2 should be used if there are more than 6 authors

**Figure 1**

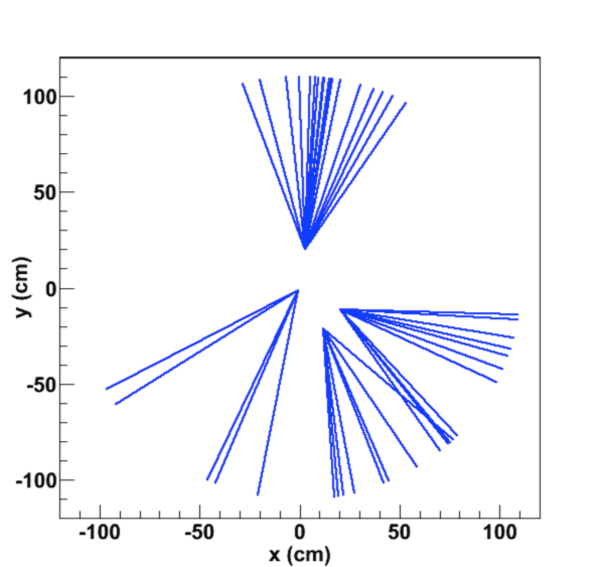
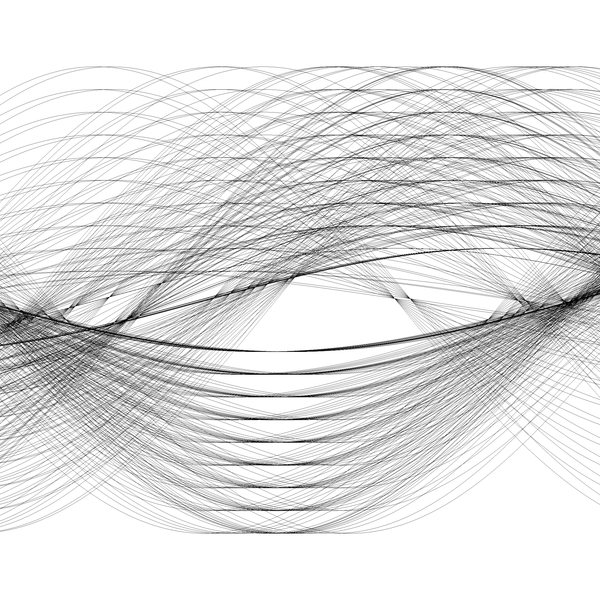
1. **b) **

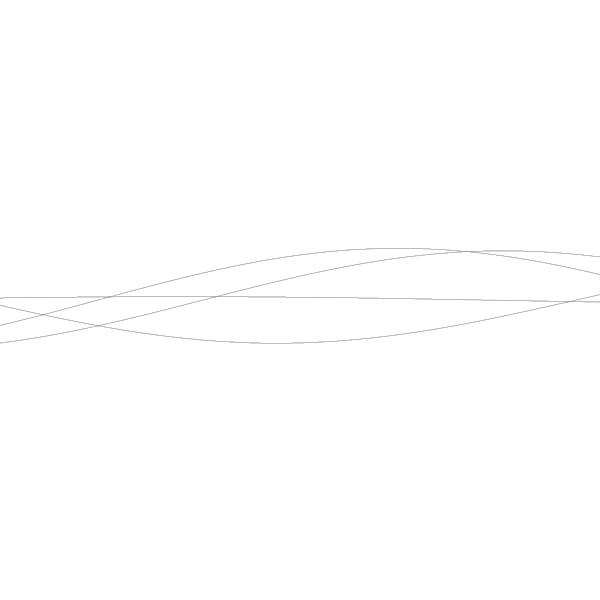
**c) **

**Figure 2**

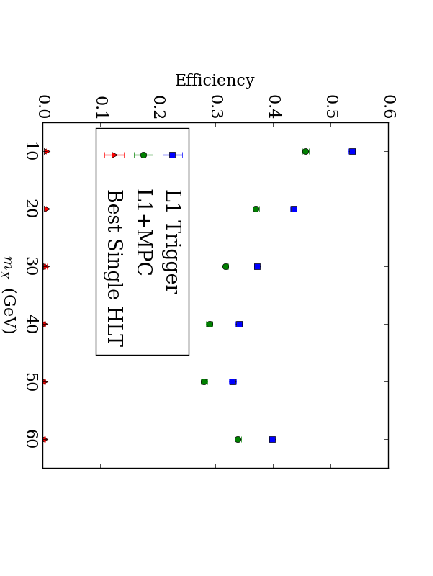
****

**Figure 3**

1. **b) **

**c) **

**Figure 4**

****