

修 士 学 位 論 文

高輝度 LHC-ATLAS 実験 初段ミュオントリガーのための 転移学習を用いた横運動量算出手法

令和8年2月6日

専 攻 名 物理学専攻
学籍番号 248S121S
氏 名 山口 凜太郎

神戸大学大学院理学研究科博士課程前期課程

概要

LHC-ATLAS 実験は陽子陽子衝突で生成される粒子を測定し、標準模型の精密検証および新物理探索を行う。2030 年以降に予定される HL-LHC 運転では瞬間ルミノシティ増大に伴い、限られたストレージのもとで必要な事象を選別するトリガーの重要性が一層高まる。

これに対応するため ATLAS では、検出器およびトリガー、データ収集系の大規模アップグレードが計画される。このアップグレードに伴い、トリガーは高レート環境下でも物理感度を維持できる判定性能と運用上の堅牢性が求められる。

ミュオンは多くの物理過程で重要なシグネチャであり、初段ミュオントリガーはミュオン候補を高速に生成して後段へ引き渡す役割を担う。エンドキャップ領域では主として Thin Gap Chamber 検出器で得られるヒット情報を用い、磁場中での軌跡の曲がりに基づいて横運動量の判定を行う。

初段トリガーでは処理時間に厳しい制約があるため、ヒットの組み合わせと運動量判定を対応づけた参照表 (Coincidence Window) を用いて迅速に判定する方式が採られる。高輝度化に向けたアップグレードでは読み出し回路の刷新や処理単位の粒度向上により、領域ごとに最適化された Coincidence Window を用意し、検出器条件の変化にも柔軟に追従できる枠組みが重要となる。

本研究では、高輝度化に向けたアップグレードを見据えた初段エンドキャップミュオントリガーにおいて、Coincidence Window を実データから構成するための一連の手順を整備し、領域ごとの生成から性能評価までを一貫して実行できる枠組みを構築した。

シミュレーションデータを用いた検証において、従来の Coincidence Window 作成手法と比較してトリガー取得効率の改善と運動量判定の分解能向上が得られることを示した。実データを用いた検証では従来手法と比べて転移学習によって性能が改善することを示し、少統計でもデータ取得時の検出器条件の違いを取り込みながら学習が収束することを確認した。

本論文では、TGC を用いたエンドキャップミュオントリガーの概要と Coincidence Window の位置づけを述べ、本研究で構築した手順と評価結果を示す。

目次

| | | |
|----------|---------------------------------|-----------|
| 1 | 序論 | 1 |
| 2 | LHC-ATLAS 実験 | 4 |
| 2.1 | LHC の概要 | 4 |
| 2.2 | ATLAS 検出器の概要 | 4 |
| 2.2.1 | ATLAS 検出器の全体像 | 4 |
| 2.2.2 | ATLAS 実験における座標系 | 5 |
| 2.2.3 | マグネットシステム | 7 |
| 2.3 | ATLAS 検出器サブシステム | 7 |
| 2.3.1 | 内部飛跡検出器 | 7 |
| 2.3.2 | カロリメータ | 9 |
| 2.3.3 | ミュオンスペクトロメータ | 9 |
| 2.4 | トリガー・データ収集 (TDAQ) システム | 15 |
| 2.4.1 | Run 3 におけるトリガーシステム (L1 + HLT) | 15 |
| 2.5 | 高輝度化と課題 | 16 |
| 2.5.1 | 高輝度化に向けたアップグレードと運転計画 | 16 |
| 2.5.2 | HL-LHC におけるトリガーシステム | 16 |
| 2.5.3 | 本研究との関係 | 17 |
| 3 | 初段ミュオントリガー | 19 |
| 3.1 | 初段ミュオントリガーの概要 | 19 |
| 3.1.1 | 初段ミュオントリガーにおける TGC 検出器 | 19 |
| 3.2 | ヒット情報の処理単位 | 19 |
| 3.3 | 高輝度化に向けたアップグレードと読み出し | 20 |
| 3.3.1 | 高輝度化に向けたアップグレード方針 | 20 |
| 3.3.2 | TGC 読み出し回路のアップグレード | 22 |
| 3.3.3 | TGC 検出器のエレクトロニクス | 23 |
| 3.4 | Sector Logic Board におけるトリガーロジック | 24 |
| 3.4.1 | ロジックの全体像 | 24 |
| 3.4.2 | Channel Mapping | 26 |
| 3.4.3 | Station Coincidence | 26 |
| 3.4.4 | Segment Reconstruction | 27 |
| 3.4.5 | Wire Strip Coincidence | 27 |
| 3.4.6 | Inner Coincidence | 29 |
| 3.4.7 | Track Selector | 29 |

| | | |
|----------|--|-----------|
| 3.5 | 高輝度化アップグレードに向けた CW 作成の先行手法 | 29 |
| 3.5.1 | ヒットマップの作成と Window 抽出 | 30 |
| 3.5.2 | Coincidence Window の最適化 | 30 |
| 3.5.3 | 先行手法における課題 | 31 |
| 3.6 | 機械学習を用いた Coincidence Window の作成手法 | 32 |
| 3.6.1 | 機械学習 | 32 |
| 3.6.2 | 教師あり学習 | 33 |
| 3.6.3 | ニューラルネットワーク | 33 |
| 3.6.4 | 誤差逆伝播と最適化 | 34 |
| 3.6.5 | 汎化と過学習 | 34 |
| 3.6.6 | Coincidence Window の作成への応用 | 35 |
| 3.6.7 | 機械学習手法の課題点 | 37 |
| 3.7 | 少統計適応としての転移学習 | 38 |
| 3.7.1 | 転移学習の目的と基本概念 | 38 |
| 3.7.2 | 深層学習における転移 | 39 |
| 3.7.3 | 代表的な転移学習の戦略 | 40 |
| 3.7.4 | 転移学習による実データのためのファインチューニング | 41 |
| 3.8 | 本研究の目的 | 42 |
| 4 | 転移学習を用いた Coincidence Window の作成 | 43 |
| 4.1 | 機械学習の学習設計 | 43 |
| 4.1.1 | 用いるデータ | 43 |
| 4.1.2 | データの処理 | 44 |
| 4.1.3 | モデル設計 | 46 |
| 4.2 | モンテカルロシミュレーションデータを用いた機械学習 | 48 |
| 4.2.1 | MC データの学習結果 | 49 |
| 4.2.2 | 機械学習の性能評価 | 49 |
| 4.3 | 実データを用いた転移学習 | 52 |
| 4.3.1 | モデル構造と更新の方針 | 52 |
| 4.3.2 | 実データの学習結果 | 53 |
| 4.3.3 | 転移学習の性能評価 | 53 |
| 4.4 | 電荷判定 | 56 |
| 4.4.1 | 学習データの作成 | 56 |
| 4.4.2 | 学習モデル | 56 |
| 4.4.3 | 電荷マップの作成 | 57 |
| 4.4.4 | 電荷 Window の作成 | 58 |
| 5 | Coincidence Window の評価 | 59 |
| 5.1 | Coincidence Window の作成 | 59 |
| 5.1.1 | MC データのための Coincidence Window | 60 |

| | | |
|----------|--|-----------|
| 5.1.2 | 実データのための Coincidence Window | 60 |
| 5.1.3 | 電荷学習の結果 | 60 |
| 5.2 | Coincidence Window の評価方法 | 62 |
| 5.2.1 | Turn-on curve | 62 |
| 5.3 | MC シミュレーションデータを用いた先行手法との性能比較 | 64 |
| 5.3.1 | p_T ごとの取得効率 | 64 |
| 5.3.2 | 位置ごとの取得効率 | 64 |
| 5.3.3 | p_T ごとの電荷判定効率 | 65 |
| 5.3.4 | 位置ごとの電荷判定効率 | 66 |
| 5.4 | 実データを用いた転移学習による Coincidence Window の評価 | 66 |
| 5.4.1 | MC 学習モデルと実データ転移学習モデルの比較 | 67 |
| 5.4.2 | 転移学習モデルから作成した Coincidence Window の評価 | 68 |
| 5.4.3 | 位置ごとの取得効率 | 69 |
| 5.4.4 | 3 種類の Coincidence Window の比較 | 70 |
| 5.4.5 | 学習する層の比較 | 73 |
| 5.4.6 | 負の転移の対策 | 73 |
| 5.5 | 実データ転移学習による電荷 Window の評価 | 74 |
| 6 | 結論・展望 | 77 |
| | 謝辞 | 79 |
| | 参考文献 | 81 |

第1章

序論

我々の身の回りにある物質を構成する最小単位は素粒子であり、それらの相互作用を記述する理論として素粒子標準模型が確立されている。図 1.1 のように、標準模型は、物質を構成するフェルミオンと相互作用を媒介するゲージボソン、さらに電弱対称性の自発的破れを担うヒッグス粒子から構成される。2012 年のヒッグス粒子発見により、標準模型は多くの現象を精密に説明する理論として確立した一方で、暗黒物質の正体、物質・反物質非対称性、ニュートリノ質量の起源、階層性問題など、標準模型の枠内では説明が難しい未解決問題も数多く残されている。これらの課題に迫るためには、より高いエネルギー、より大きな統計量を取得し、稀な過程や微小な効果を高精度に測定することが不可欠である。中でもミュオンは物質中での透過力が高く同定が比較的正確に行えるため、電弱相互作用やヒッグス粒子の精密測定、重フレーバーを含む多様な解析、さらには標準模型を超える新物理探索において重要なシグネチャの一つとなる。したがって、ミュオンを含む事象を高効率かつ高い純度で取得できることは、今日の素粒子実験における物理感度を左右する重要な要素である。

欧州原子核研究機構 (CERN) に設置された大型ハドロン衝突型加速器 (Large Hadron Collider, LHC) では、陽子を TeV 領域まで加速し、25 ns 間隔のバンチ衝突によって最大 40 MHz の頻度で陽子・陽子衝突事象が生成される。ATLAS (A Toroidal LHC Apparatus) 検出器は LHC の汎用検出器として、ヒッグス粒子の精密測定や標準模型の検証、ならびに標準模型を超える新物理探索を目的に、衝突で生成される荷電粒子や光子、ジェット、ミュオンなど多様な最終状態を測定している。しかし、衝突頻度が非常に高いため ATLAS 検出器が出力するデータ量は極めて膨大であり、そのすべてを保存することはできない。このため ATLAS では、限られた記録帯域の中で物理的に重要な事象を選別するトリガー・データ収集 (Trigger and Data Acquisition, TDAQ) 系が不可欠である。

ATLAS のトリガーシステムは、短い時間で高速に事象を選別する初段トリガーと、より多くの情報を用いて高精度に選別する後段トリガーの二段階により構成される。初段トリガーはハードウェアで実装され、固定レイテンシの制約下で、衝突の中から後段で詳細に再構成すべき候補事象を即座に選別する。後段トリガーはソフトウェアにより実装され、初段トリガーで特定された領域情報などを活用しつつ、より詳細な再構成・選別を行って最終的に記録する事象を決定する。この二段階構成により、40 MHz という膨大な入力レートを最終的に数 kHz という現実的な保存レートへと低減しつつ、物理解析に必要な事象を効率よく取得している。

LHC では今後、より大きな統計量を獲得するための高輝度化 (High-Luminosity LHC, HL-LHC) が計画されている。HL-LHC では瞬間ルミノシティの増加に伴って同一バンチ

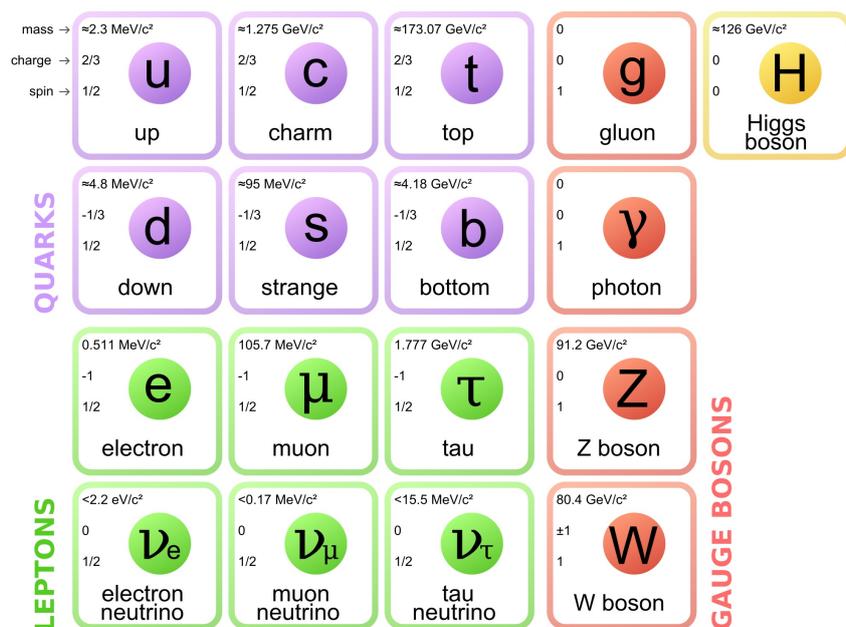


図 1.1: 素粒子標準模型を構成する粒子群 [1].

衝突あたりで衝突する陽子数が大幅に増大し、背景事象の増加による誤判定の増加やイベントサイズの増大が課題となる。こうした高レート環境においても物理感度を維持するため、ATLASでは検出器およびTDAQ系の大規模アップグレードが予定されている [2]。特に初段トリガーに対しては、許容するトリガーレートをおよそ1 MHz規模へ拡張し、判定に利用できるレイテンシもおよそ10 μs 程度へ拡張することが計画されている。これにより初段トリガーで扱える情報量・アルゴリズムの自由度は増す一方で、より高い性能と堅牢性を満たすトリガー判定が強く要求される。

ミュオンを含む事象を効率良く取得するため、ATLASの初段ミュオントリガーはミュオン検出器の情報を用いて短時間で候補を選別し、後段トリガーへ渡す役割を担う。初段ミュオントリガーは大きくバレル部とエンドキャップ部に分けられ、エンドキャップ部では主としてTGC(Thin Gap Chamber)検出器を用いてミュオンの再構成および運動量の判定を行う。トロイド磁場中でのミュオン軌跡の曲がり具合が運動量に依存することを利用し、ヒット情報から求めた曲がり具合の情報を用いて横運動量 p_T を推定する。ただし、初段トリガーは限られたレイテンシの下で動作するため、詳細な飛跡の再構成を逐次実行することは難しい。そこで、位置の差から対応する p_T 閾値(および電荷符号)を即座に返すLook-Up Table(LUT)が用いられる。このLUTはCoincidence Windowと呼ばれてFPGA上に実装される [2]。Coincidence Windowは、エンドキャップ初段ミュオントリガーにおける高速な p_T 判定の中核を担い、その設計・最適化は高レート環境での高効率と低トリガーレートを両立する上で本質的である。

高輝度化に向けたアップグレードでは、エンドキャップ初段ミュオントリガーの精度向上のためにCoincidence Windowの更新が必要となる。現在アップグレード開発で用いられているCoincidence Window作成手法はシミュレーションで生成されたデータに基づき、理想的な検出器配置を想定している [3]。しかし実機では検出器位置が設計時の理

想配置から外れているため、シミュレーション由来の Coincidence Window をそのまま実データへ適用すると、領域によっては性能劣化を招く可能性がある。さらに、Coincidence Window における分割領域を細かく取るほど作成に必要な統計量は増大する。

これらの課題への対処方法として、現行の運転では機械学習を用いて Coincidence Window を作成する手法が導入されている [4]。この手法では、ヒット位置と曲がり具合を入力として p_T を推定する回帰モデルを学習し、その出力をしきい値段階へ離散化することで対応表を構成する。対応表作成を学習と離散化として統一的に扱えるため、従来のようなヒットマップの穴埋めや段階構造の調整を一貫した枠組みで実装できる利点を持つ。さらに学習に実データを用いることにより、アライメントや磁場の不完全性、検出器応答の揺らぎなど、シミュレーションでは完全に再現しきれない効果をまとめて取り込んだ形で対応表を構成できる。

一方で、現行運転においても領域によって少統計によって学習が収束せず、結果として一部領域で効率が低下する事例が確認された。加えてアップグレードでは処理単位の粒度向上に伴い必要な統計量が増大するため、統計不足の問題はより深刻化すると予想される。そこで本研究では、より少ない統計量でも学習を収束させて Coincidence Window を作成できる手法として転移学習 (Transfer Learning) [5] を導入する。まず統計量が相対的に豊富な MC を用いて回帰モデルを学習させ、その後実データでモデルの一部を微調整することにより、曲がり量と p_T の関係に関する物理的に妥当な表現を初期状態として保持しつつ、ミスアライメントなど実データ固有のずれへ少ない更新量で追従し、少統計により学習が収束しないといった問題を抑えて対応表の品質を維持することを目指す。さらに、作成した Coincidence Window を用いたトリガー性能を評価し、本研究における手法の有効性を検証する。

本論文の構成は以下の通りである。第2章では HL-LHC に向けた ATLAS 実験の概要とアップグレード計画、ならびに高レート環境での課題を述べる。第3章ではエンドキャップ初段ミュオントリガーのアルゴリズムと Coincidence Window の役割、システム構成を詳述する。第4章では機械学習および転移学習を用いた Coincidence Window 作成手法を示し、学習・出力の手順を説明する。第5章では作成したモデルおよび Coincidence Window の性能評価を行い、Turn-on curve 等の指標を用いて比較・検証する。最後に第6章で本研究の結論をまとめ、今後の展望について述べる。

第2章

LHC-ATLAS 実験

本章では、本研究の背景として LHC および高輝度化 (HL-LHC) 計画、ATLAS 検出器の座標系と構成、ならびにトリガー・データ収集 (TDAQ) システムの概要を述べる。特に HL-LHC では、瞬間ルミノシティが増加のために、バンチ交差ごとの陽子衝突事象数が格段に増加する。これによって背景事象によるトリガーレートの増大やイベントサイズの増加が課題となる。このため ATLAS 実験ではアップグレードが計画されており、次章以降で扱う初段ミュオントリガーも大きく刷新される。

2.1 LHC の概要

LHC (Large Hadron Collider) は CERN の地下に建設された周長約 27 km の陽子陽子衝突型円形加速器である。LHC は重心系エネルギー $\sqrt{s} = 14$ TeV、瞬間ルミノシティ $1 \times 10^{34} \text{ cm}^{-2}\text{s}^{-1}$ を設計目標としている。陽子ビームはバンチと呼ばれる約 10^{11} 個の陽子の集団が連なっており、衝突点において 25 ns 間隔でバンチを交差させる。一度のバンチ交差で複数の陽子衝突が起こる現象をパイルアップと呼ぶ。LHC には 4 つの主要な衝突点があり、汎用検出器の ATLAS [6] および CMS [7] の他、ALICE [8]、LHCb [9] が設置されている。

陽子ビームは図 2.1 に示すように、線形加速器 (LINAC) から PS、SPS を経て段階的に加速された後に LHC へ入射される。

2.2 ATLAS 検出器の概要

本節では、ATLAS 検出器の全体構成と設計思想を概観したのち、以降の検出器説明やトリガー記述で共通に用いる座標系を定義する。

2.2.1 ATLAS 検出器の全体像

ATLAS 検出器は LHC の衝突点の 1 つに設置された大型汎用検出器である。ATLAS 検出器は複数の検出器を組み合わせて構成されており、内側から内部飛跡検出器、カロリメータ、ミュオンスペクトロメータが配置される。また、内部飛跡検出器とカロリメータの間には超伝導ソレノイド磁石、カロリメータの外側にはトロイド磁石がそれぞれ設置されている。これらの検出器から得られる情報を組み合わせることで、粒子識別や粒子の運動量・エネルギー測定などを行う。

The CERN accelerator complex Complexe des accélérateurs du CERN

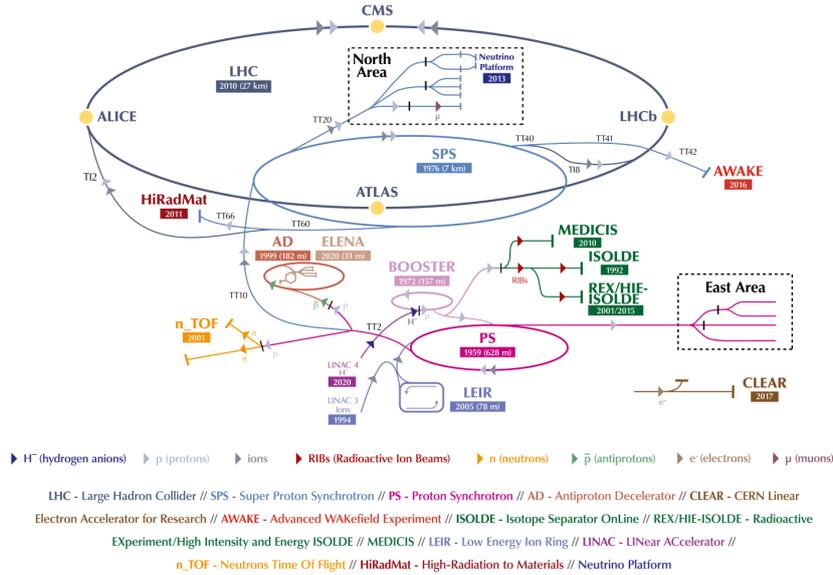


図 2.1: CERN における陽子ビーム加速器群の概観 [10]。線形加速器およびシンクロトロン (PSB, PS, SPS) で段階的に加速された陽子ビームが LHC へ入射され、各衝突点における ATLAS などの実験へと陽子が供給される。

CERN 本部地下に設置された ATLAS 検出器の外観を [図 2.2](#) に示す。

ATLAS では粒子種に応じて各サブ検出器で異なる応答が得られる。各検出器に到達する粒子と代表的な振る舞いを [図 2.3](#) に示す。

2.2.2 ATLAS 実験における座標系

ATLAS 実験では、直交座標系 (x, y, z) および円筒座標系 (R, ϕ, z) を用いて検出器内の位置を記述する。座標系の定義の概略を [図 2.4](#) に示す。

直交座標系では原点を衝突点 (検出器中心) に取り、 x 軸を LHC 中心方向、 y 軸を地上方向に取る右手系になっている。 z 軸はビーム軸に平行に取り、 $z > 0$ 側を A-side、 $z < 0$ 側を C-side と呼ぶ。円筒座標系では、動径方向 $R = \sqrt{x^2 + y^2}$ 、方位角 ϕ 、およびビーム軸からの極角 θ を用いる。

一般にハドロン衝突型加速器では、ビーム軸方向のローレンツブーストに対して扱いやすい変数として擬ラピディティ η を用いる。 η は θ により

$$\eta \equiv -\ln \tan \left(\frac{\theta}{2} \right) \quad (2.1)$$

と定義される。特に $|\eta| < 1.05$ をバレル領域、 $|\eta| > 1.05$ をエンドキャップ領域と呼ぶ。また、粒子の運動量 p に対して横方向運動量 $p_T = |p| \sin \theta$ を定義する。この値はトリガーを含む多くの解析で主要な指標として用いられる。

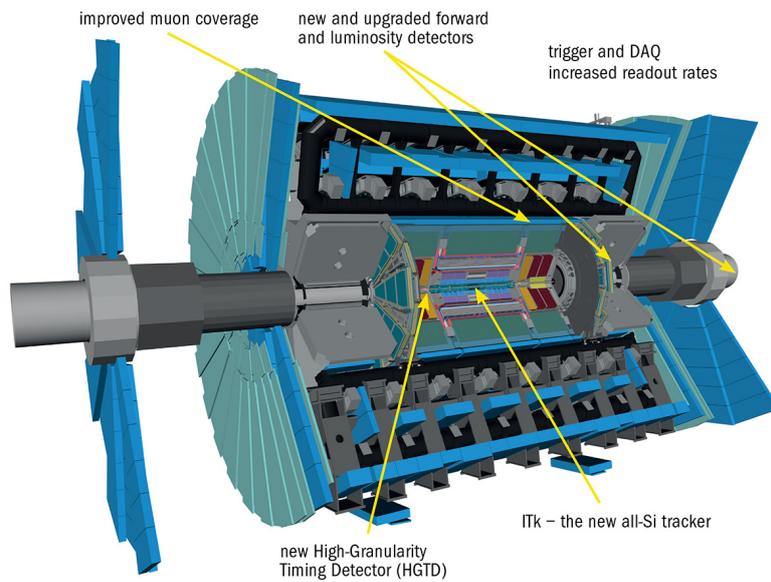


図 2.2: ATLAS 検出器の外観 [11]。CERN 本部地下にある LHC の衝突点に設置された大型汎用検出器であり、内部飛跡検出器・カロリメータ・ミュオンスペクトロメータから構成される。

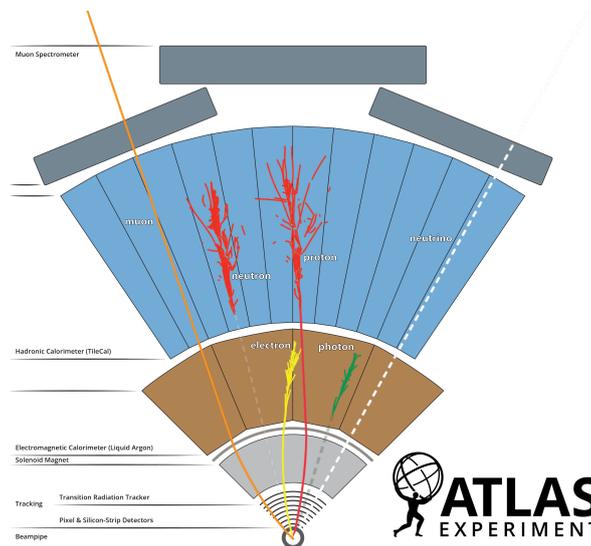


図 2.3: ATLAS 検出器の断面模式図と、各サブ検出器における粒子の典型的な応答 [12]。内側から内部飛跡検出器、電磁・ハドロンカロリメータ、ミュオンスペクトロメータが同心円状に配置される。

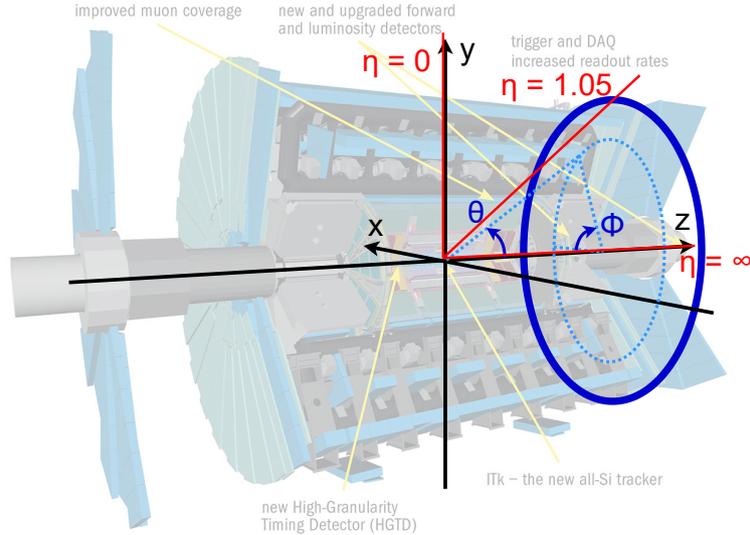


図 2.4: ATLAS 検出器で用いられる座標系の定義。原点を相互作用点とし、 z 軸をビーム方向、 x 軸を LHC リング中心方向、 y 軸を鉛直上向きとする右手系を用いる。方位角 ϕ および擬似ラピディティ η で角度を表す。

2.2.3 マグネットシステム

ATLAS では荷電粒子の運動量測定のために超伝導磁石により磁場を形成する。内部飛跡検出器ではソレノイド磁場により荷電粒子の飛跡を曲げ、その曲率から横方向運動量 p_T を測定する。一方、ミュオンスペクトロメータではカロリメータを透過したミュオンを主にトロイド磁場で曲げ、外層の測定器で運動量を決定する。

ATLAS に設置された超伝導磁石の配置と作られる磁場の概略をそれぞれ図 2.5 と図 2.6 に示す。ソレノイド磁石は z 方向におよそ 2T の磁場を生成する。トロイド磁石はバレル部とエンドキャップ部に分かれて配置され、 ϕ 方向に 8 回対称の構造を持つ。バレル部・エンドキャップ部のトロイド磁石は互いの干渉を避けるため、 ϕ 方向に 22.5° ($\pi/8$) 回転して配置される。トロイド磁場は構造上、 η 方向 - ϕ 方向に一樣ではなく、ミュオンスペクトロメータの検出器配置や再構成・トリガーの設計ではこの非一樣性を踏まえる必要がある。

2.3 ATLAS 検出器サブシステム

本節では ATLAS 検出器を構成する主要サブシステムについて概説する。衝突点から近い順に内部飛跡検出器、カロリメータ、ミュオンスペクトロメータと構成されている。

2.3.1 内部飛跡検出器

内部飛跡検出器は衝突点に最も近い位置に設置され、衝突点で生成された荷電粒子の飛跡を測定する。ソレノイド磁場中での曲率測定により運動量情報を与えるほか、頂点再構成

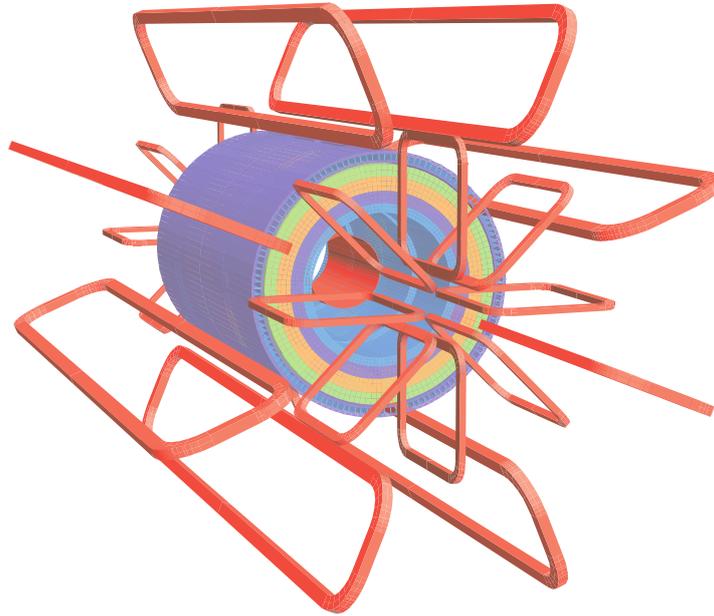
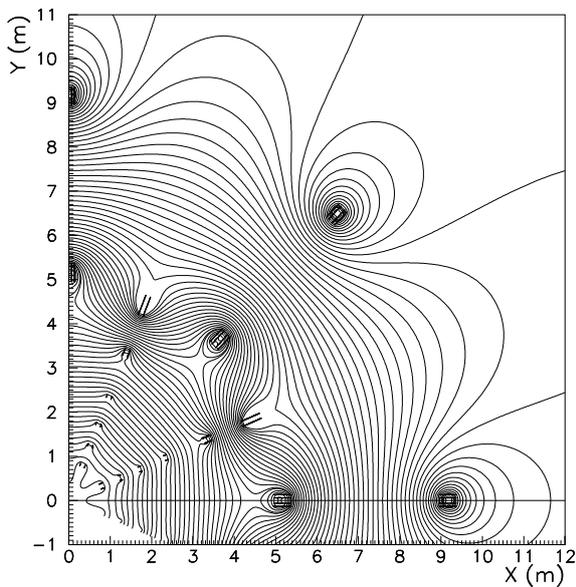
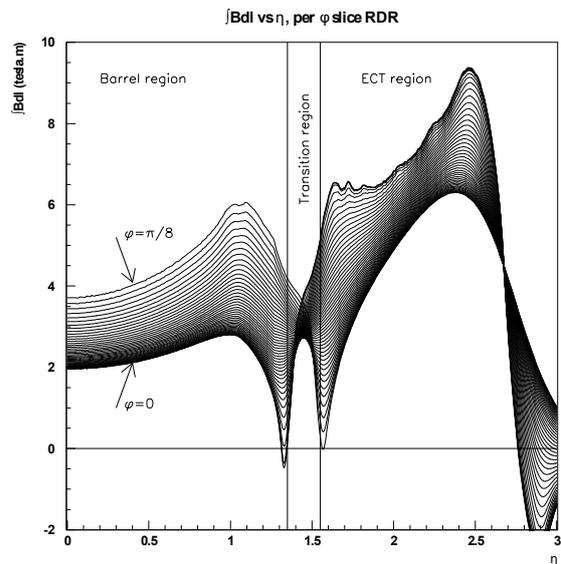


図 2.5: ATLAS 検出器に設置された超伝導磁石の配置 [6]。内部飛跡検出器はソレノイド磁場中で荷電粒子の曲率から運動量を測定し、ミュオンスペクトロメータはバレル, エンドキャップのトロイド磁場を用いてミュオン運動量を測定する。



(a) $z = 0$ 平面での磁場強度分布 [13].



(b) η 方向に沿った積分磁場の分布 [13].

図 2.6: ATLAS 検出器における磁場分布の例。ソレノイド・トロイド磁場の重ね合わせにより、空間的に非一様な磁場が形成される。

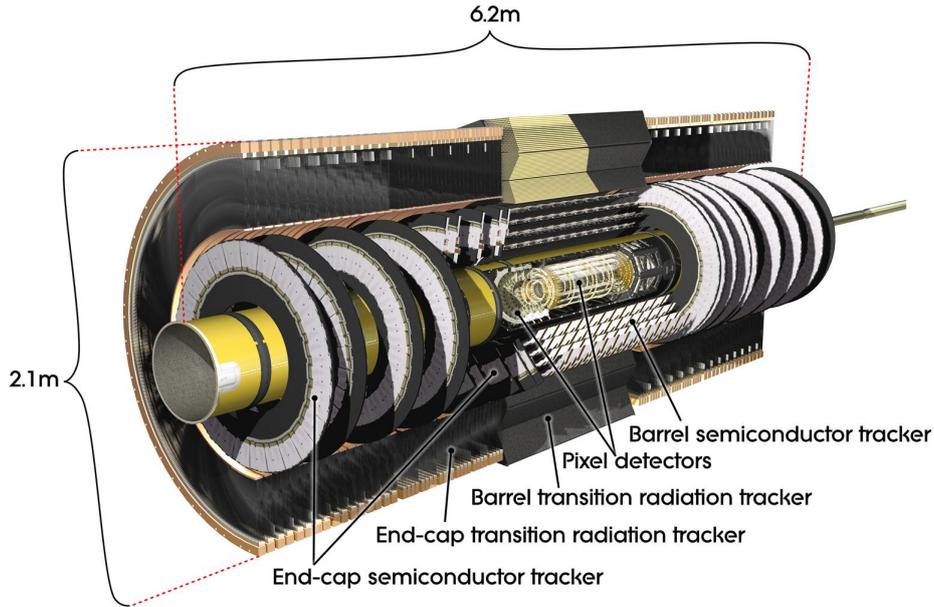


図 2.7: 内部飛跡検出器の概観 [12]。Pixel、SCT および TRT 等により、ソレノイド磁場中で荷電粒子の飛跡と生成頂点を高精度に再構成する。

や荷電粒子の同定などに重要な役割を果たす。

内部飛跡検出器は図 2.7 に示す通り内側からピクセル検出器 (Pixel)、Semiconductor Tracker (SCT)、および Transition Radiation Tracker (TRT) から構成される。

2.3.2 カロリメータ

カロリメータは図 2.8 に示すように、内部飛跡検出器の外側に配置され、粒子が物質中で落とすエネルギーを測定する。内側に電磁カロリメータ、外側にハドロンカロリメータが配置されており、電子・光子のエネルギー測定、ジェットのエネルギ測定、ならびに消失横運動量の算出に寄与する。

電磁カロリメータは電磁シャワーにより電子・光子のエネルギーおよび位置を測定する。ハドロンカロリメータはハドロンシャワーによりジェット等のエネルギー測定を担う。

2.3.3 ミューオンスペクトロメータ

ミューオンスペクトロメータは ATLAS 検出器の最外層に設置され、カロリメータを透過してきたミューオンを検出する。トロイド磁場中での曲がり具合からミューオンの運動量を精密に測定することができる。また時間分解能の高い検出器は初段ミューオントリガーとして用いられる。

ミューオンスペクトロメータはトロイド磁石の幾何構造に合わせて ϕ 方向に 8 回対称に配置される。ミューオン検出器サブシステムの概観を図 2.9 に示す。

ATLAS のミューオン検出器は、トリガー判定用の検出器と精密測定用の検出器に大別される。前者は高い時間分解能を持つ一方で位置分解能は比較的 low、後者は高い位置

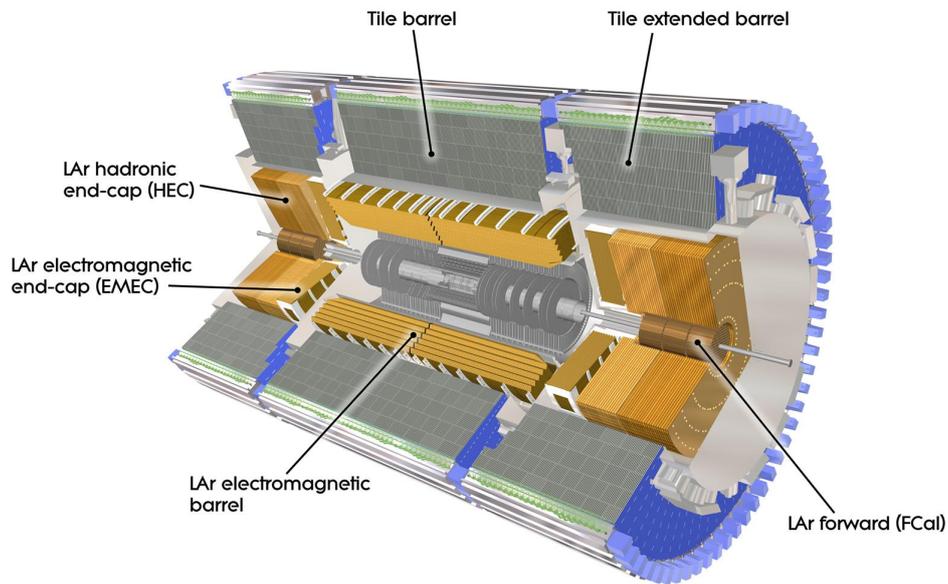
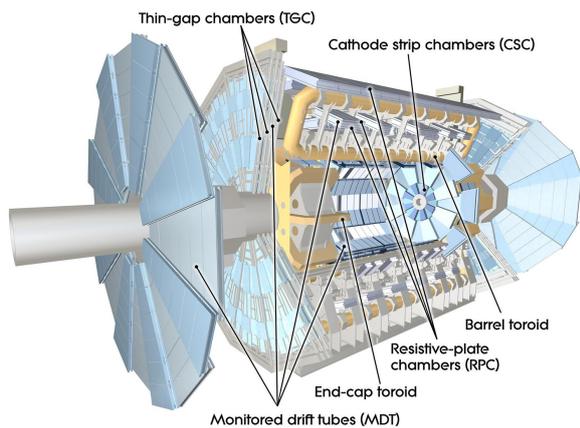
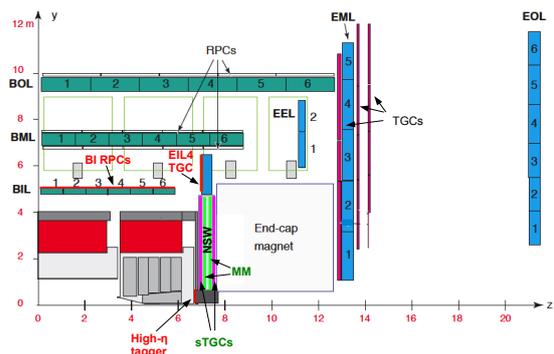


図 2.8: ATLAS カロリメータ系の概観 [14]。電磁カロリメータとハドロンカロリメータで構成され、電磁シャワーおよびハドロンシャワーの発達を利用して粒子のエネルギーを測定する。



(a) ミューオン検出器サブシステム [15]



(b) ビーム軸を含む断面におけるバレルおよびエンドキャップのステーション配置 [2]

図 2.9: ATLAS ミューオンスペクトロメータの概観。

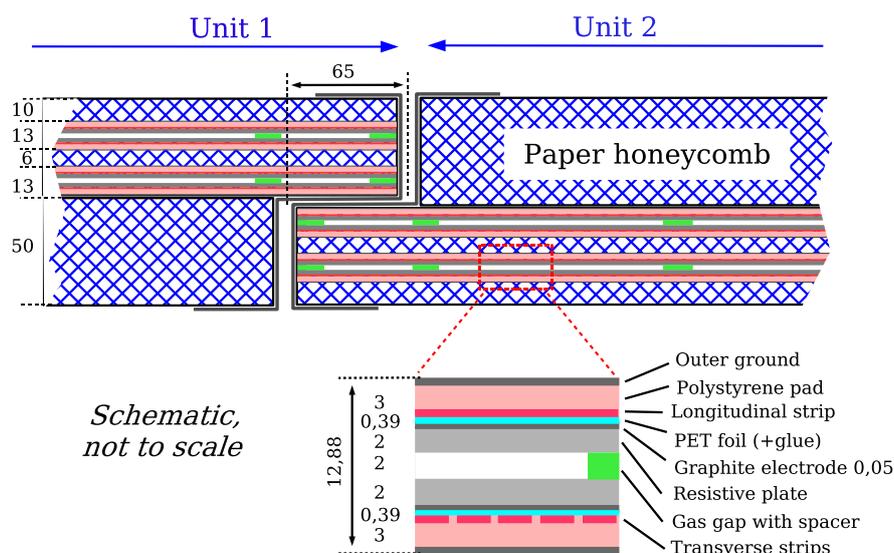


図 2.10: RPC (Resistive Plate Chamber) 検出器の構造模式図 [6]。高抵抗電極板間のガスギャップで生じた信号を読み出しストリップで検出する。主にバレル領域のミュオントリガーに用いられる。

分解能を持つ一方で時間分解能は比較的低い。以下では ATLAS 検出器を構成するミュオン検出器について述べる。

RPC 検出器

Resistive Plate Chamber (RPC) は、主にバレル領域 ($|\eta| < 1.05$) に設置されるトリガー用ミュオン検出器である。RPC は図 2.10 に示すような構造をしており、2 枚の高抵抗プレート間に薄いギャップを設けた平行電極板型のガス検出器であり、ギャップ内の電場による雪崩増幅信号をストリップで読み出す。ストリップは互いに直交する向きに配置され、 η 方向および ϕ 方向の位置情報を取得できる。この高い時間分解能により、バンチ交差の識別と初段トリガーへの利用が可能となる。

TGC 検出器

Thin Gap Chamber (TGC) はエンドキャップ領域に設置されるトリガー用ミュオン検出器であり、主に $1.05 < |\eta| < 2.4$ をカバーする。図 2.11 にビーム軸方向から撮影した TGC 検出器の図を示す。

TGC は Multi Wire Proportional Chamber (MWPC) の一種で、図 2.12 のような構造をしており、アノードワイヤーとカソードストリップの二次元読み出しによりヒット位置を測定する。ワイヤー情報から R 方向、ストリップ情報から ϕ 方向の位置が得られ、これらを組み合わせることでトリガーに必要な方位・角度情報を構成する。

TGC はトロイド磁場領域の内側に位置する Endcap Inner (EI) などのステーション

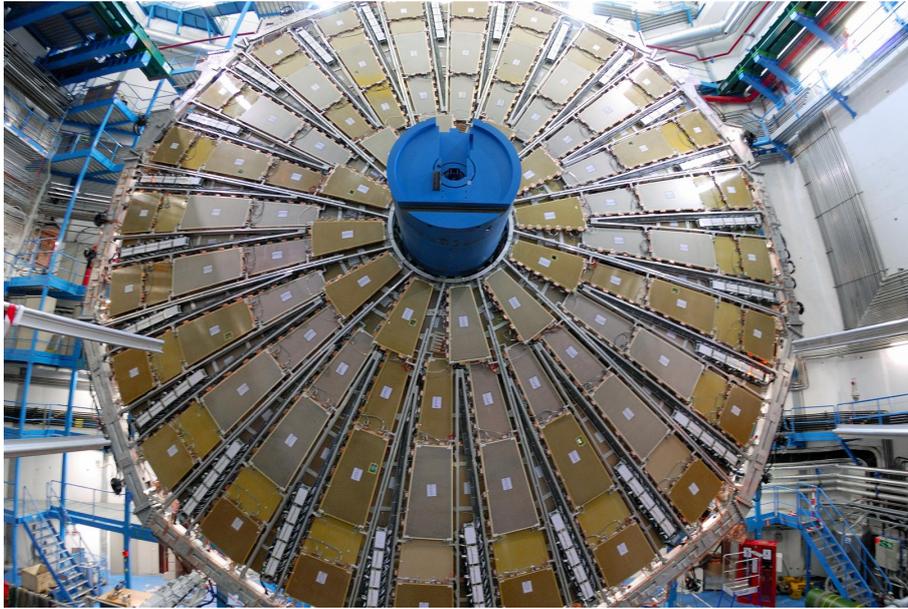


図 2.11: エンドキャップ部に設置された TGC 検出器を正面から撮影した写真 [16]。ミュオンの飛来を高い時間分解能を以て検出する。主にミュオンの飛来を検出してトリガーをかけるために用いられる。

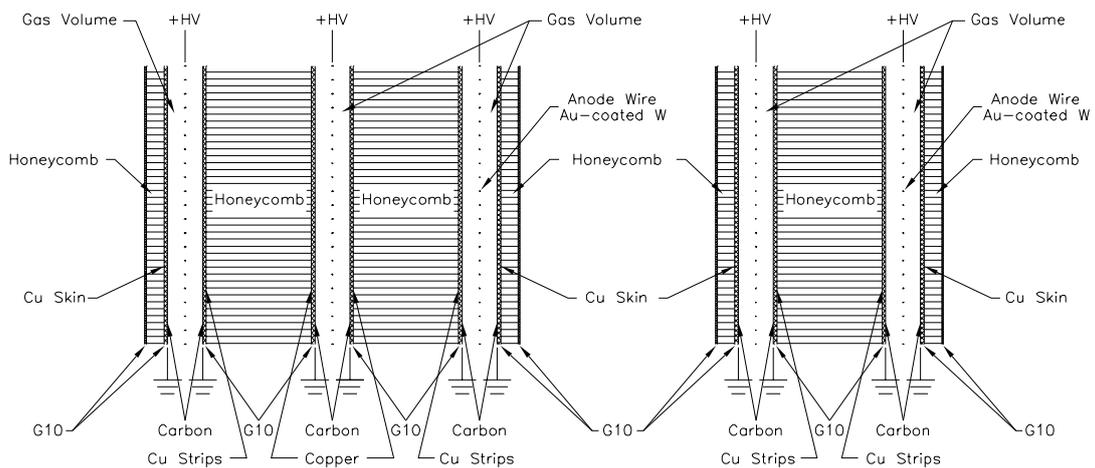


図 2.12: TGC (Thin Gap Chamber) 検出器の構造模式図 [17]。アノードワイヤと両側カソードからなる多線式ガス検出器であり、高い時間分解能を活かしてエンドキャップ領域のミュオントリガーに用いられる。

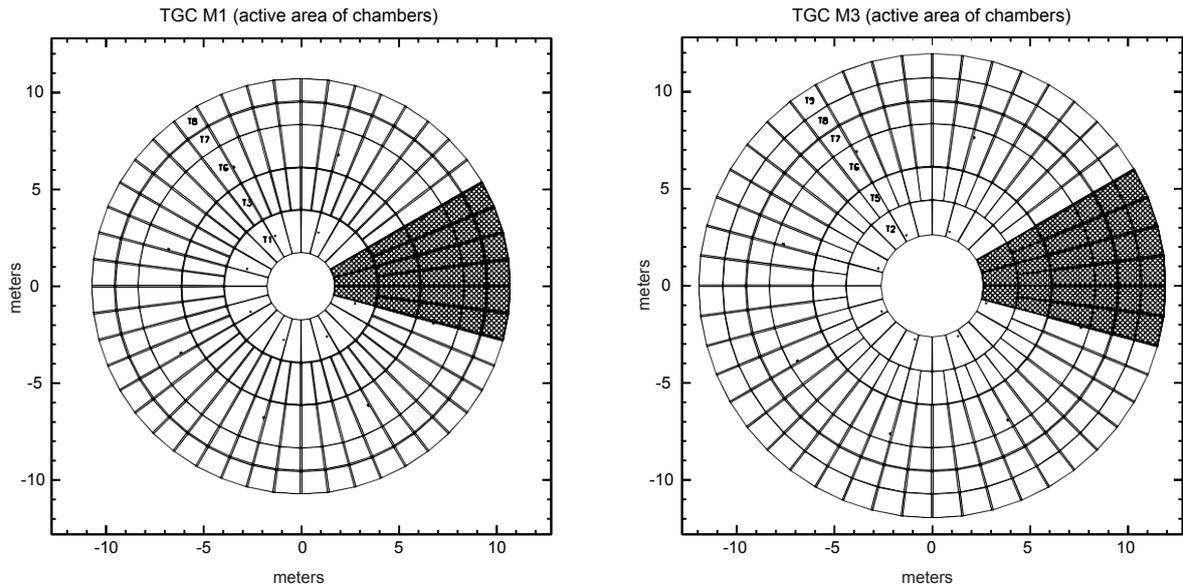


図 2.13: TGC Big Wheel におけるステーション構成例 (M1/M3) [17]。M1 は 3 層 1 組 (Triplet)、M3 は 2 層 1 組 (Doublet) として構成され、複数層のヒット情報を用いてコインシデンス判定を行う。

と、磁場領域外側に位置する Big Wheel (BW) で構成される。BW は衝突点に近い側から M1、M2、M3 のステーションに分けられ、M1 は 3 層構造 (Triplet)、M2/M3 は 2 層構造 (Doublet) を基本とする。TGC BW を構成する Triplet, Doublet の構造を図 2.13 に示す。

MDT 検出器

Monitored Drift Tube (MDT) はミュオンの飛跡を精密に測定するための検出器であり、ドリフトチューブを多層に配置した構造を持つ (図 2.14)。荷電粒子がチューブ内のガスを電離し、生成した電子が中心ワイヤーへドリフトする時間を測定することで、通過位置を高い分解能で求めることができる。

New Small Wheel (NSW)

New Small Wheel (NSW) はトロイド磁場領域の内側に導入されたミュオン検出器であり、高レート環境下での背景抑制および角度分解能向上を目的とする。NSW はトリガー判定と精密測定の両方に寄与するよう設計されており、主に $1.3 < |\eta| < 2.7$ の領域をカバーする。NSW は図 2.15 のような、small-strip TGC (sTGC) と Micromegas (MM) を組み合わせた構造を持ち、通過位置と飛跡角度の情報を高精度に与えることで、エンドキャップ領域でのミュオントリガー性能向上に寄与する。

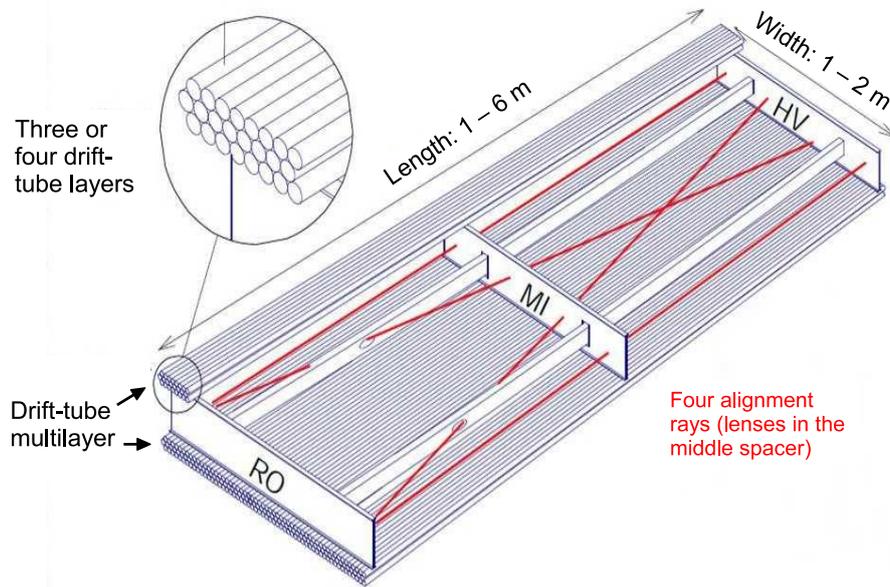


図 2.14: MDT (Monitored Drift Tube) 検出器の構造模式図 [6]。ドリフトチューブ内のセンスワイヤに対する電離電子のドリフト時間を測定し、ミューオン飛跡位置を高精度に決定する。

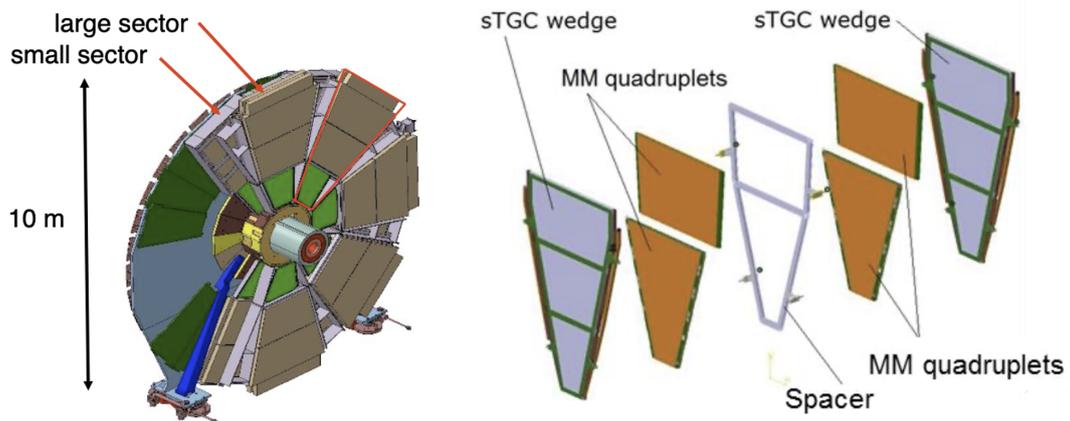


図 2.15: New Small Wheel (NSW) の概略 [18]。sTGC および Micromegas により高レート環境での精密測定とトリガー情報を提供する。

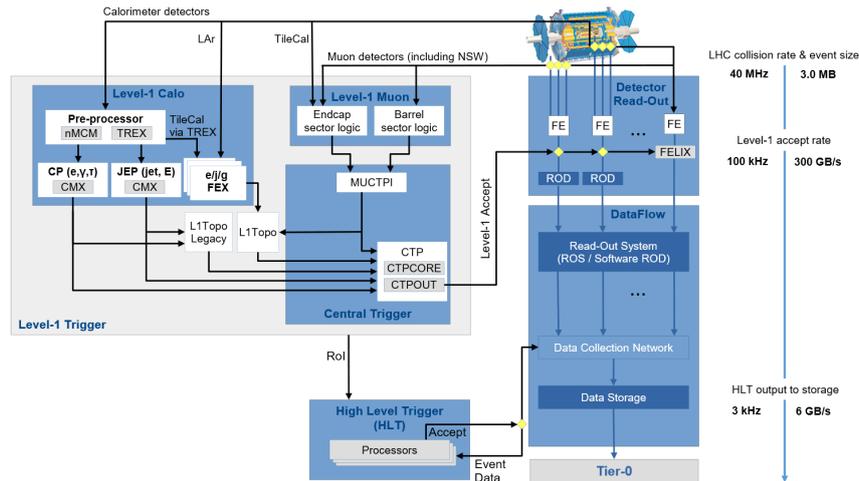


図 2.16: Run 3 における ATLAS トリガー・データ取得 (TDAQ) システムの模式図 [19]。ハードウェアベースの Level-1 トリガーとソフトウェアベースの High Level Trigger により事象を選別し、データをストレージへ転送する。

2.4 トリガー・データ収集 (TDAQ) システム

本節では ATLAS 実験におけるトリガー・データ収集 (Trigger and Data Acquisition: TDAQ) システムの役割を概説する。

ATLAS では 1 バンチ交差ごとに出てくるデータは 2MB 程度であり、これが 40MHz で起こるため最終的なデータ量は毎秒 80TB にも上る。これらすべてを保存することはストレージの都合上できないため、オンラインで保存するイベントを選択する必要がある。この機能を担うのがトリガーシステムであり、トリガーと DAQ が連動して事象の選別と読み出しを行う。

2.4.1 Run 3 におけるトリガーシステム (L1 + HLT)

Run 3 の ATLAS トリガーシステムは二段階構成であり、初段ハードウェアトリガー (Level-1, L1) と後段ソフトウェアトリガー (High Level Trigger, HLT) からなる。L1 は固定レイテンシ (約 $2.5 \mu\text{s}$) でトリガー判定を行い、イベントレートを 100 kHz 程度に抑えることが要求される。L1 は主にカロリメータ情報を用いる L1Calo、ミュオン検出器情報を用いる L1Muon、およびそれらのトリガーオブジェクトのトポロジーを扱う L1Topo に分かれ、最終的な L1 判定は Central Trigger Processor (CTP) で行われる。HLT は L1 で選別された事象に対して、Region-of-Interest (RoI) 情報を活用しつつ、ソフトウェアでより精密な再構成・判定を行うことで、最終的な記録レートをさらに低減する。この Run 3 における ATLAS のトリガー・データ取得 (TDAQ) システムの全体構成 (L1/HLT とデータフロー) を図 2.16 に示す。

| | Run-1 | Run-2 | Run-3 | HL-LHC |
|--|-----------------------|----------------------|----------------------|--------------------------|
| 重心系エネルギー [TeV] | 7 | 13 | 13.6 | 14 |
| 最大瞬間ルミノシティ [$\text{cm}^{-2}\text{s}^{-1}$] | 0.77×10^{34} | 2.0×10^{34} | 2.0×10^{34} | $(5-7.5) \times 10^{34}$ |
| ピークパイルアップ数 | 45 | 60 | 60 | 200 |

表 2.1: LHC の各運転期における重心系エネルギー, 最大瞬間ルミノシティおよびピーク時のパイルアップ数。

| | Run 3 | HL-LHC |
|-------------|-------------------|------------------|
| 初段トリガーレート | 100 kHz | 1 MHz |
| 初段トリガーレイテンシ | $2.5 \mu\text{s}$ | $10 \mu\text{s}$ |

表 2.2: Run 3 と HL-LHC における初段トリガーレートと初段トリガーレイテンシの比較

2.5 高輝度化と課題

LHC-ATLAS 実験では、より大きな統計量を得るために高輝度化 (High-Luminosity LHC, HL-LHC) が計画されている。HL-LHC では瞬間ルミノシティの増加に伴いパイルアップ数も増大し、背景事象の増加によるフェイクトリガーの増大などが課題となる。

表 2.1 に示す通り、HL-LHC では瞬間ルミノシティおよびパイルアップが大幅に増大する。

この高レート環境に対応し、可能な限り多くの有用な物理事象を取得するため、初段トリガー系の性能強化が不可欠である。その一例として、Run 3 と HL-LHC における TDAQ 要求の比較を表 2.2 に示す。

2.5.1 高輝度化に向けたアップグレードと運転計画

LHC のこれまでの運転の経過と、これからのアップグレードの概観を図 2.17 に示す。LHC は 2010 年より本格運転を開始し、2010–2012 年 (Run-1)、2015–2018 年 (Run-2) の期間に運転が行われてきた。Run-1 および Run-2 で得られた積分ルミノシティは約 190 fb^{-1} である。その後、一度アップグレードを経て、現在は 2022–2026 年の運転期間 (Run-3) が進行している。Run-3 終了後には高輝度化に向けたアップグレードを経て、2030 年頃から HL-LHC 運転が予定されており、10 年間で積分ルミノシティ 4000 fb^{-1} の取得が見込まれている。

2.5.2 HL-LHC におけるトリガーシステム

HL-LHC ではルミノシティ増加に伴うパイルアップ増大により背景事象が増加し、現行システムだとトリガーレートが増加してしまう。そのため、高輝度化に向けてより精度よくトリガーを行うために、トリガーシステムの大規模アップグレードが計画されている。初段トリガーレートの許容量を Run 3 の 100 kHz から 1 MHz へ増強し、さらに初段トリガーにおける判定時間 (レイテンシ) を $2.5 \mu\text{s}$ から $10 \mu\text{s}$ へ拡張することで、より複雑なアルゴリズムを初段に導入することができる。

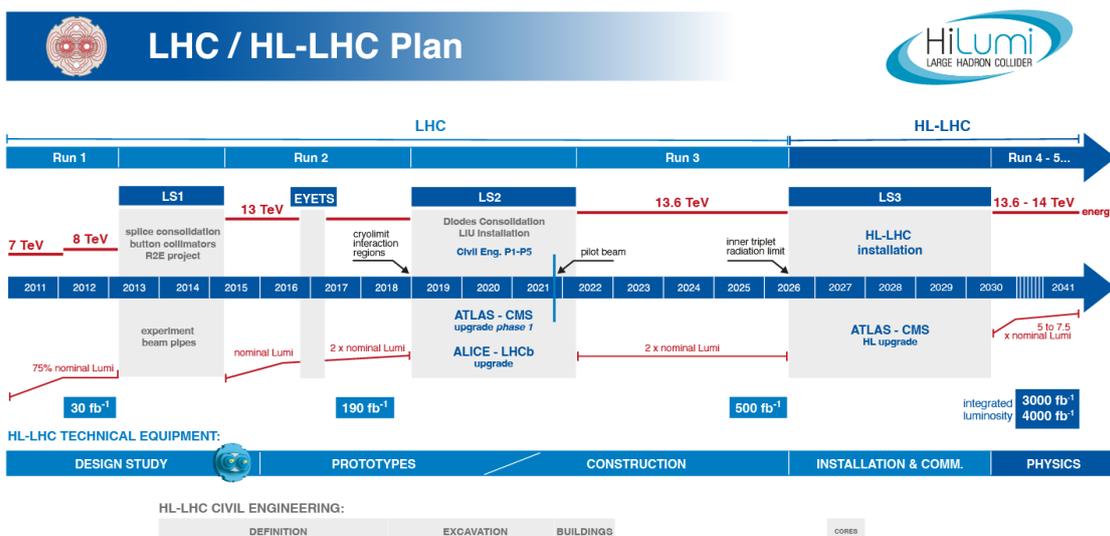


図 2.17: これまでの LHC の運転と高輝度化 (HL-LHC) に向けた運転計画の概略 [2]。各 Run と Long Shutdown の期間、および HL-LHC 運転への移行計画のタイムスケジュールを示す。

HL-LHC におけるトリガーシステムは、ハードウェアトリガーである Level-0 (L0) と、ソフトウェアトリガーである Event Filter (EF) から構成される。L0 は L0Calo および L0Muon を中心に構成され、ミュオン側では高速検出器 (TGC/RPC) に加えて MDT など精密測定検出器情報も組み合わせることで、より高精度な初段判定を目指す。EF ではよりオフライン解析に近いアルゴリズムを適用し、トリガーレートを 10 kHz 程度まで削減する。HL-LHC における ATLAS のトリガー・データ取得 (TDAQ) システムの全体構成を図 2.18 に示す。

2.5.3 本研究との関係

本研究では、HL-LHC 環境における初段ミュオントリガー、特にエンドキャップ領域を対象とする。エンドキャップ初段ミュオントリガーでは TGC および NSW 等の情報を用いた高度なトリガー判定が重要であり、初段トリガーレイテンシ拡張・読み出し刷新のもとで処理方式が大きく変化する。次章ではエンドキャップ初段ミュオントリガーの構成と処理フローを述べ、以降の章で本研究の手法と結果を示す。

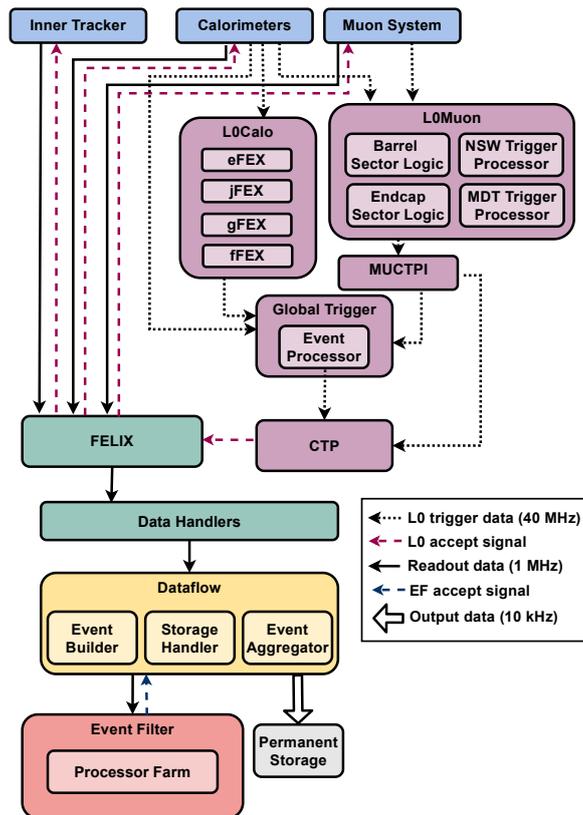


図 2.18: HL-LHC における ATLAS TDAQ システムの概要 [20]。ハードウェアベースの Level-0 トリガーとソフトウェアベースの Event Filter の二段階で事象を選別し、選別されたイベントのデータがストレージへ保存される

第3章

初段ミュオントリガー

本章では初段ミュオントリガーについて述べる。初段ミュオントリガーはその領域によってエンドキャップとバレルの二つに大別されるが、本研究ではエンドキャップにおける初段ミュオントリガーを取り扱う。

3.1 初段ミュオントリガーの概要

ATLAS 実験の初段ミュオントリガーは、バレル部では RPC、エンドキャップ部では TGC を用いて構成される。初段トリガーでは LHC のバンチ交差に同期して高速にトリガー判定を行い、興味のないイベントを落として後段トリガーへと送るイベント数を抑制することが求められる。エンドキャップ領域では、トロイド磁場を通過したミュオンが TGC に到達する。トロイド磁場においてミュオンはその飛跡を曲げられるが、横運動量が大きいほど曲がり具合は小さくなる。エンドキャップ部における初段ミュオントリガーではこの性質を用いて横運動量 p_T の概算を行うことで、閾値を超える高 p_T ミュオンを選別する。

3.1.1 初段ミュオントリガーにおける TGC 検出器

エンドキャップ部初段トリガーでは、主としてトロイド磁場の外側に配置された TGC Big Wheel (TGC BW) の情報を用いる。ミュオンの曲がり具合は、主としてトロイド磁場による寄与で η 方向に大きく現れる。また、衝突点近傍のソレノイド磁場成分やトロイド周辺での磁場成分の影響により、 ϕ 方向にも曲がりが生じる。図 3.1 に示すように、 p_T が大きいほど磁場で曲げられにくく、飛跡の曲がり具合は小さくなる。エンドキャップ部初段トリガーではこの性質を利用し、TGC BW の多層ヒットパターンから再構成した飛跡と、無限運動量を仮定した飛跡とのなす角度を用いて p_T の閾値判定を行う。

3.2 ヒット情報の処理単位

TGC BW は広い η - ϕ 領域を覆うため、トリガー処理の単位として検出器の幾何に沿った領域分割が導入されている。TGC BW のカバレッジは、概ね $1.05 < |\eta| < 1.92$ のエンドキャップ領域と $1.92 < |\eta| < 2.4$ の Forward 領域に大別され、全方位角を覆うように複数の Trigger Sector へ分割される。Trigger Sector の分割概念を図 3.2 に示す。Trigger Sector

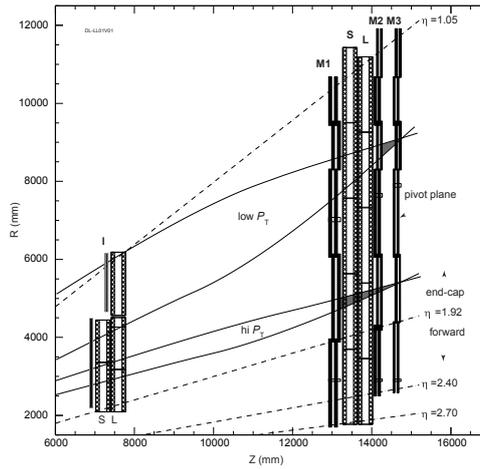


図 3.1: TGC システムについて、ビーム軸を通る面で見えた図 [17]。衝突点からエンドキャップ部までの幾何配置と、トロイド磁場中でのミュオン飛跡の曲がりの違いを示す。低 p_T ミュオンほど曲率が大きく、高 p_T ミュオンはほぼ直進するため、初段トリガーではこの曲がり量を利用して p_T の概算を行う。

は初段トリガーにおいて独立に処理される基本単位であり、Trigger Sector に対する処理はそれぞれに対応する Sector Logic (図 3.7) という回路基板で行われる。

Sector Logic 内部では、FPGA 上での並列処理を実現するため Trigger Sector 内の領域をさらに小領域に分割して演算を行う。この小領域を Unit と呼び、後述するトリガーロジックにおいては Unit ごとに独立に処理を進める。さらに、角度情報から運動量閾値を決定する段階では候補飛跡の組合せを扱いやすくするため、Unit 内をまとめた Unit Region (UR) を定義する。例えば、8 UR ではストリップ 4 本とワイヤー 2 本の組合せにより複数の飛跡候補を形成し、その中から代表候補を選別する。一方、すべてを 8 UR 単位で実装すると FPGA のリソースが不足するため、ストリップ 4 本とワイヤー 8 本からなる 32UR を定義し、候補を段階的に絞り込む構成が用いられる。このようにして、ある Trigger Sector における Unit Region は図 3.3 に示すように定義される。

3.3 高輝度化に向けたアップグレードと読み出し

本節では、LHC の高輝度化に向けた初段ミュオントリガー系のアップグレード方針と、TGC の信号がフロントエンド回路から Sector Logic へ到達するまでの読み出しの流れを説明する。

3.3.1 高輝度化に向けたアップグレード方針

先述の通り高輝度化に伴ってパイルアップが増加するとミュオン系ではフェイクトリガーが増え、限られたトリガーレートの中で物理的に重要な事象を効率良く選別することがより難しくなる。そのため、初段トリガーでは検出効率を保ちつつトリガーレートを抑えるために、位置・角度分解能の向上や内層検出器情報を用いたコインシデンスによる背

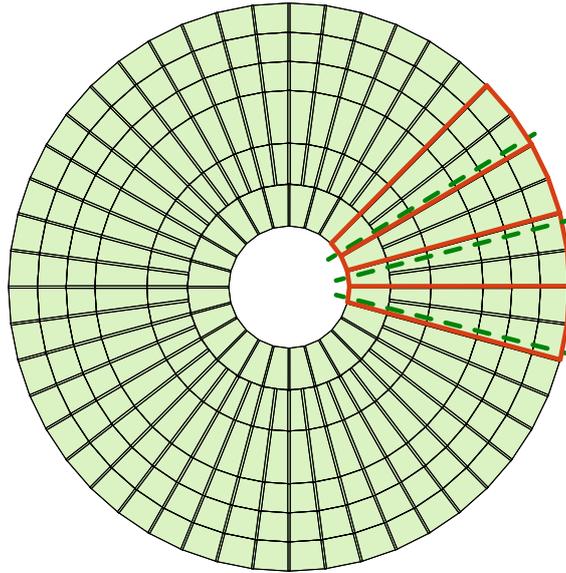


図 3.2: TGC Big Wheel における Trigger Sector 分割の模式図 [21]。赤枠で示した領域がそれぞれ1つの Trigger Sector に対応し、以降のトリガーロジック（セグメント再構成、Coincidence 判定、候補選別など）は Trigger Sector 単位で並列に処理される。

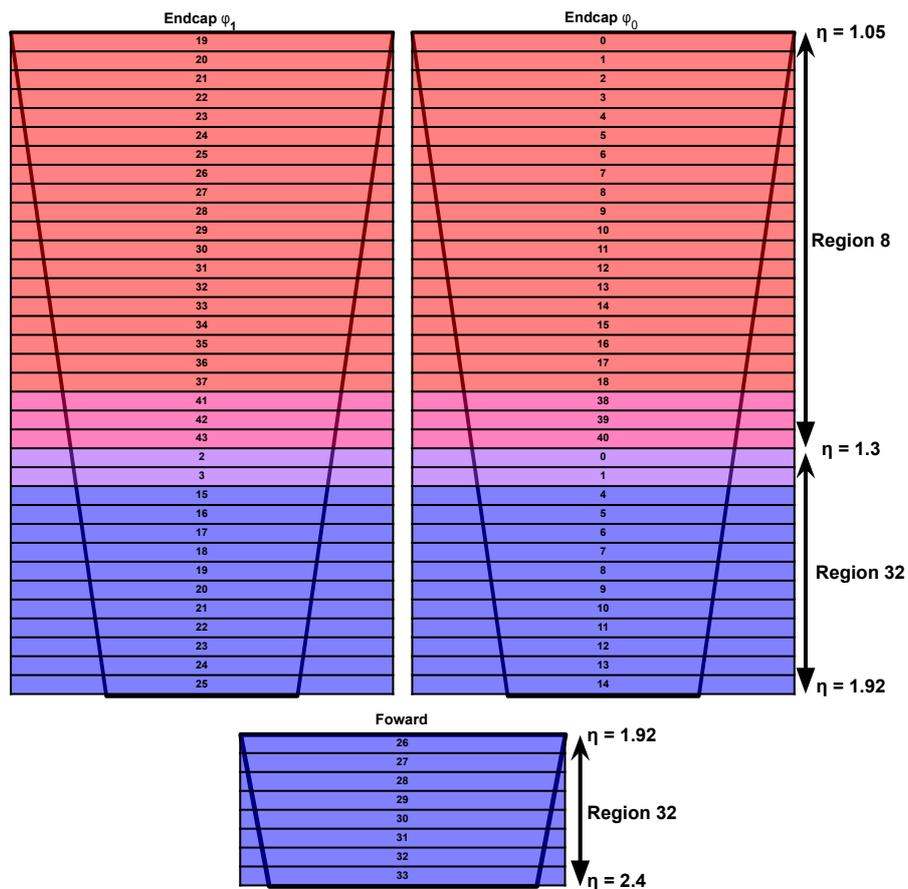


図 3.3: トリガーロジックで用いる Unit Region (UR) の定義の模式図 [22]。Trigger Sector 内を η 方向・ ϕ 方向に分割し、各 UR を処理単位として $\Delta\theta$, $\Delta\phi$ などの曲がり量や位置情報を定義する。

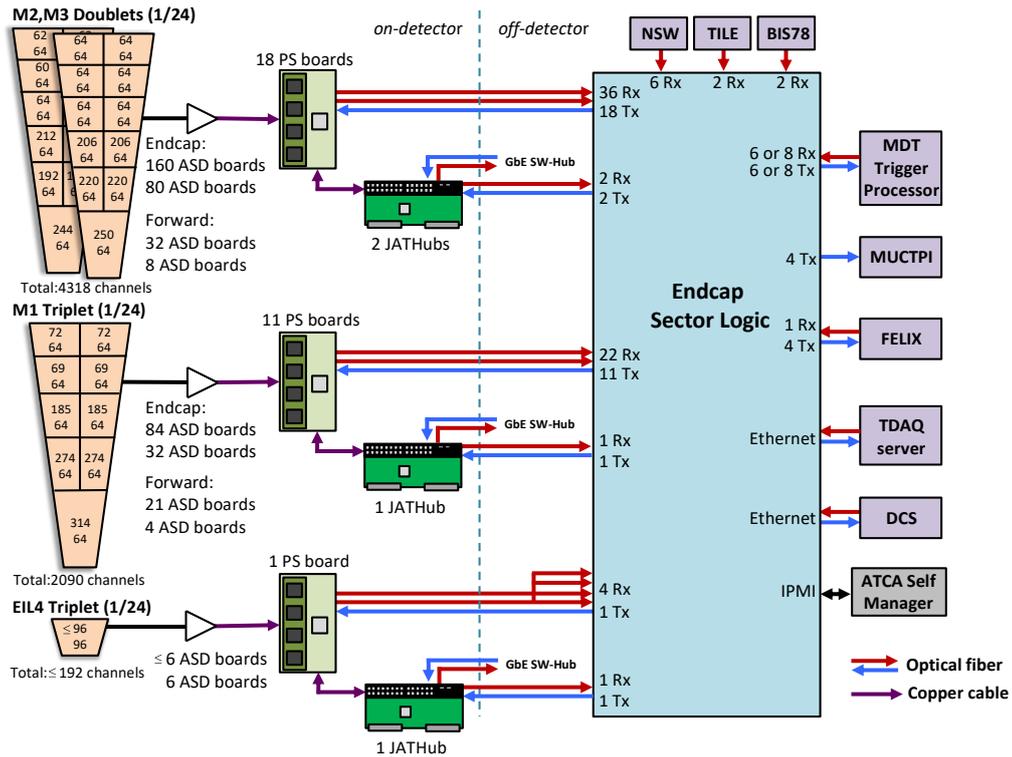


図 3.4: 高輝度化に向けたアップグレードを経た TGC 初段ミュオントリガーにおけるエレクトロニクスの概略 [23]。検出器からの信号は前段基板で整形・バンチ分けされ、Sector Logic に集約されてトリガーロジック（マッピング、セグメント再構成、Coincidence 判定、候補形成）を実行する。形成されたミュオン候補はトリガーに応じてここからさらに後段へと読み出されていく。

景事象の抑制など判定性能の強化が求められる。

アップグレードでは読み出し全体の回路も刷新される。TGC ミュオントリガー回路の全体像を図 3.4 に示す。TGC 検出器の信号はフロントエンドで整形・デジタル化された後、バンチ交差識別を経て Sector Logic へ送られ、Sector Logic 上の FPGA で飛跡再構成と運動量閾値の概算が行われる。また、初段トリガーにより選別されたイベントに関しては、Sector Logic にバッファされていたヒットデータが後段へ読み出される。

3.3.2 TGC 読み出し回路のアップグレード

高輝度化に向けたアップグレードでは、Sector Logic が Trigger Sector 内の TGC BW 7 層からのヒットのビットマップを、ヒットの有無に関わらず全バンチ交差について受信し、トリガー処理を行う構成となる。この変更により、従来複数基板に分散していたトリガーロジックは Sector Logic の FPGA へ集約され、入力ヒット情報を最大限に活用した飛跡再構成と高速な判定が可能となる。

このような高密度入力・大規模ロジックを安定に動作させるため、読み出しの帯域確保とレイテンシ制約の下での実装最適化が重要となる。

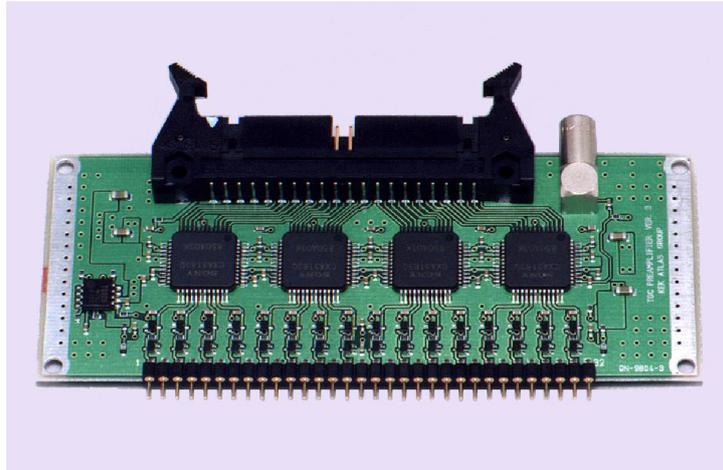


図 3.5: ASD (Amplifier – Shaper – Discriminator) ボードの外観 [23]。TGC 検出器からのアナログ信号を増幅・整形し、閾値判定によって信号があった時間幅としてデジタル化して後段へ出力する。

3.3.3 TGC 検出器のエレクトロニクス

本小節では、TGC 検出器からの信号が Sector Logic へ到達するまでの処理を、ASD ボード、PS ボード、Sector Logic ボードの順に説明する。回路全体の位置づけは図 3.4のとおりであり、各段で信号整形・同期・集約を行い、最終的に Sector Logic でトリガー判定が実行される。

ASD ボード

TGC 検出器から読み出されるアナログ信号は、図 3.5のような ASD (Amplifier Shaper Discriminator) ボードに入力される。ASD では信号を増幅し、読み出された信号波形の整形を行った上で閾値と比較し、閾値を超えた時間に対応するデジタル信号として出力する。この出力は後段回路へ伝送可能な差動信号 (LVDS) として生成される。また、回路試験や校正のために、TGC 信号を模擬するテストパルス入力機構を備える。

PS ボード

図 3.6に示す PS (Primary Processing) ボードは ASD からの LVDS 信号を受け、バンチ交差に同期したヒット情報として整形する回路である。具体的には、信号遅延の調整により検出器や配線長に起因する到達時間差を吸収した上で、ヒット信号に対して BCID を行う。BCID されたヒット情報はビットマップへと整形され、光通信により Sector Logic へ送信される。

Sector Logic ボード

図 3.7に示す Sector Logic ボードは、PS ボードから送られてくる TGC のヒット情報を用いて、ミューオンの飛跡再構成と運動量閾値の概算、衝突点付近以外からの信号の排除な



図 3.6: PS ボードの外観 [23]。ASD からのデジタル信号を受け、ケーブル長などを考慮してバンチ交差に同期した時刻情報の付与を行ったうえで、後段の Sector Logic へ送る。

どを行う回路である。また、トリガーにより選別されたイベントに関しては、Sector Logic 上に一時保存されたヒットデータが後段の読み出し系へ転送される。さらに、PS ボード等のモニター・コントロールやクロック分配といったシステム制御も担う。

3.4 Sector Logic Board におけるトリガーロジック

本節では、図 3.7 の Sector Logic ボード上の FPGA に実装されるトリガーロジックを説明する。

3.4.1 ロジックの全体像

本小節では、Sector Logic 上で実行される処理の流れを概観する。トリガーロジックは図 3.8 に示す通り、Channel Mapping → Station Coincidence → Segment Reconstruction → Wire Strip Coincidence → Inner Coincidence → Track Selector の順にパイプライン処理として構成される。各モジュールは LHC クロック (40MHz) に同期した入力を受けつつ、FPGA 内部ではより高い周波数 (160MHz) で駆動することでレイテンシ制約下の高速処理を実現する。

まず Channel Mapping では、PS ボードから転送されたビットマップを後段ロジックで扱いやすい配置へ並べ替え、検出器幾何に由来する重複チャンネルを解消する。次に Station Coincidence では各ステーション内の多層ヒットから代表点を形成し、データ量を削減しながら位置分解能を向上させる。Segment Reconstruction では、Station Coincidence で得た代表点の組み合わせから角度情報を算出する。最後に Wire Strip Coincidence で、Wire/Strip それぞれから得た角度情報を組み合わせて運動量閾値の概算を行う。さらに内層検出器との Inner Coincidence で衝突点付近以外からのミューオンを抑制し、Track

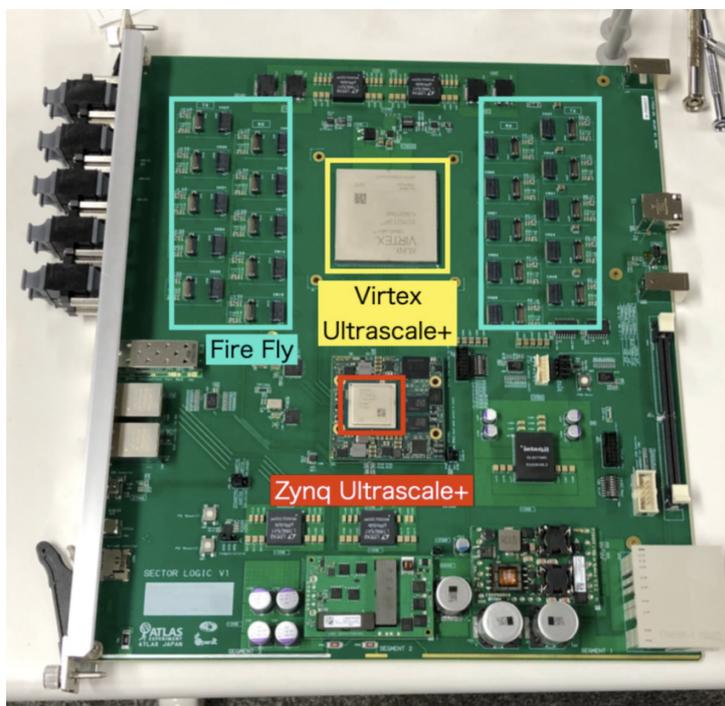


図 3.7: Sector Logic ボードの外観 [24]。前段から集約された TGC 信号を入力として、チャンネルマッピング、セグメント再構成、Coincidence 判定、候補選別などのトリガーロジックを実行し、初段ミュオン候補を後段に出力し、トリガーされた場合にはその情報を送り出す。

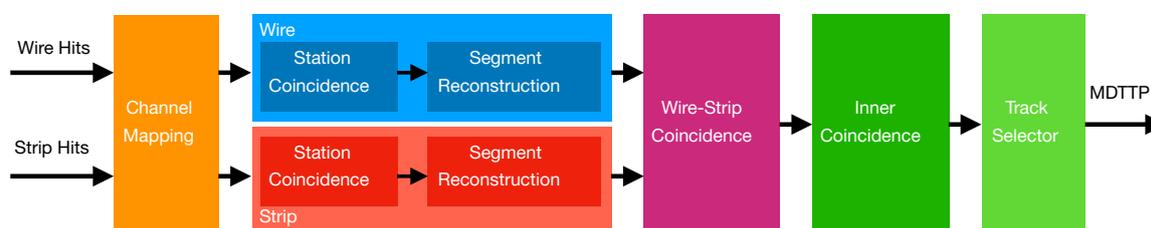


図 3.8: Sector Logic に実装されるトリガーロジックのブロック図 [25]。前段から入力されたヒット情報に対して、Channel Mapping や Station Coincidence、Wire/Strip の Segment Reconstruction を経て曲がり具合を構成し、Coincidence Window の参照や Inner Coincidence、Track Selector 等により最終的なミュオン候補を形成する。



図 3.9: Channel Mapping におけるチェンバー間 OR 処理の概要 [25]。幾何的に重なりを持つ領域で複数チェンバーの情報を統合し、後段のセグメント再構成・Coincidence 判定で扱いやすいチャンネル表現へ変換する。

Selector で後段へ送る候補数まで絞り込む。

3.4.2 Channel Mapping

Channel Mapping は PS ボードから送られたデータを、後段の処理で参照しやすい形式に整形する。TGC BW はチェンバーがわずかに重なって配置されており、この重複領域がそのまま残ると後段ロジックでの参照が煩雑になる。そこで、重複チャンネル間で OR を取って信号の重複を解消する。ワイヤーでは重複チャンネル間の OR により統一した代表化を行う。一方ストリップでは、 η 方向に曲げられたミュオンが複数チェンバーに跨ってヒットを残すことがあるため、ステーション間コインシデンスが破綻しないよう、M1・M2 でチェンバー間 OR を取って広い範囲を参照可能にする (図 3.9)。

3.4.3 Station Coincidence

Station Coincidence は、各ステーション内の多層ヒットから代表点を算出するロジックである。TGC BW では図 3.10 のように層が互い違いに配置されており、重複領域を代表点として定義することで位置情報を高精度化することができる。

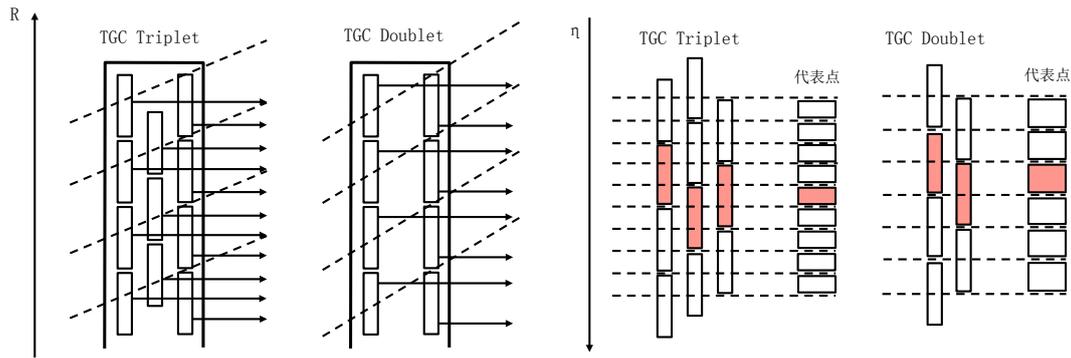


図 3.10: Station Coincidence の概念図 [26]。複数ステーションのヒット整合性を用いてバックグラウンドを抑制し、後段のセグメント再構成および p_T 判定に用いる候補を形成する。

3.4.4 Segment Reconstruction

Segment Reconstruction は、無限運動量飛跡と実際の飛跡のなす角度を出力する。エンドキャップトロイド磁場によりミューオンは主に η 方向へ曲げられるため、ワイヤー側の角度情報 $\Delta\theta$ は p_T 推定における主要な変数となる。一方 ϕ 方向の曲がり、特に多重散乱などによって低い p_T のミューオンにおいて大きくなる傾向にある。

3.4.5 Wire Strip Coincidence

Wire Strip Coincidence では、図 3.11 Segment Reconstruction の算出した角度情報と p_T 閾値を対応付けた Look-up Table を参照して p_T を得る。この LUTこそが、以降本研究で作成対象とする Coincidence Window である。

Wire Strip Coincidence では、Wire/Strip Segment Reconstruction の出力として得られる角度情報 $\Delta\theta$ および $\Delta\phi$ を入力とし、対応する p_T 閾値を Look-up table として参照する方式が用いられる (図 3.12)。CW は FPGA の BRAM に格納され、 $\Delta\theta$ と $\Delta\phi$ を bin 化したアドレスを入力することで、4 bit で表現された p_T 閾値が出力される。

実装上は、 $\Delta\theta$ は例えば $-0.16 < \Delta\theta < 0.16$ を 128 bin (7 bit)、 $\Delta\phi$ は例えば $-0.032 < \Delta\phi < 0.032$ を 16 bin (4 bit) として bin 化し、その組をアドレスとして BRAM 上の CW を参照する。

図 3.12 の通り、CW は $\Delta\theta$ - $\Delta\phi$ の 2次元平面上でどの bin がどの p_T 閾値に対応するかを与える対応表であり、Wire Strip Coincidence はこの参照結果に基づいて後段へ渡す飛跡候補の p_T 情報を付与する。

さらに、Wire Strip Coincidence では p_T 閾値判定と同様に、ミューオン候補の電荷符号判定も行う。電荷判定には $\Delta\theta$ および $\Delta\phi$ を入力とする電荷判定用の Coincidence Window を用い、トロイド磁場中での飛跡の曲がり方向の違いに基づいて、候補が正電荷・負電荷のいずれであるかを判定する。このようにして得られた電荷情報は、 p_T 情報とあわせて後段へ渡す飛跡候補に付与され、その後のトリガー処理に用いられる。

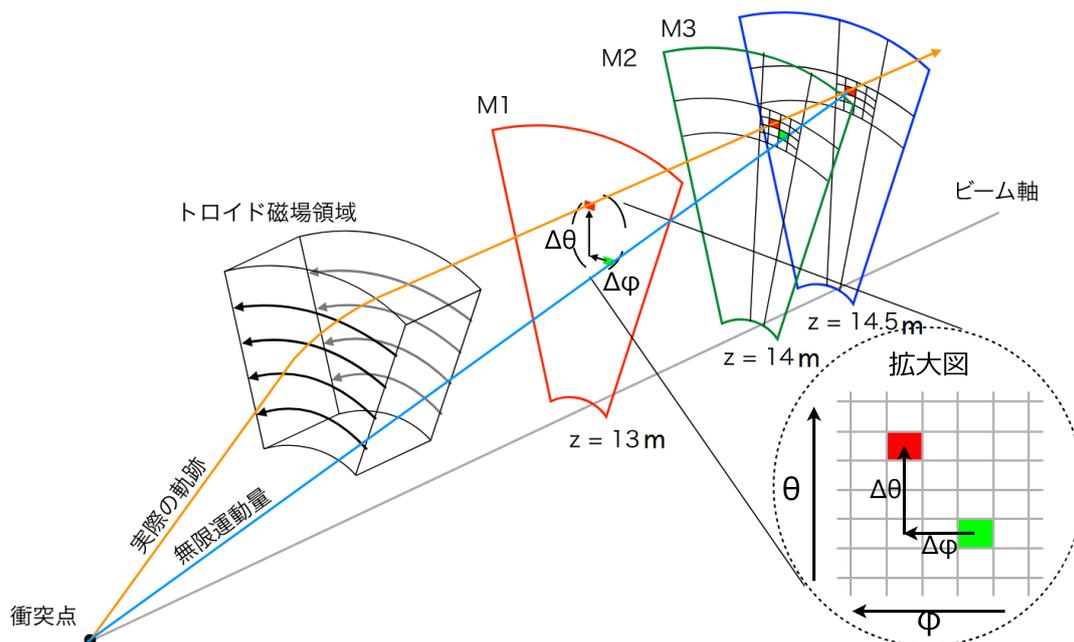


図 3.11: 角度情報から p_T を推定する概念図 ([27] より角度差を $\Delta\theta$, $\Delta\phi$ と改定して引用)。トロイド磁場による曲がりにより、外側ステーションで再構成した方向と、衝突点から衝突点を結んだ無限運動量飛跡との間に角度差が生じる。初段トリガーではこの $\Delta\theta$, $\Delta\phi$ を用いて p_T 閾値判定を行う。

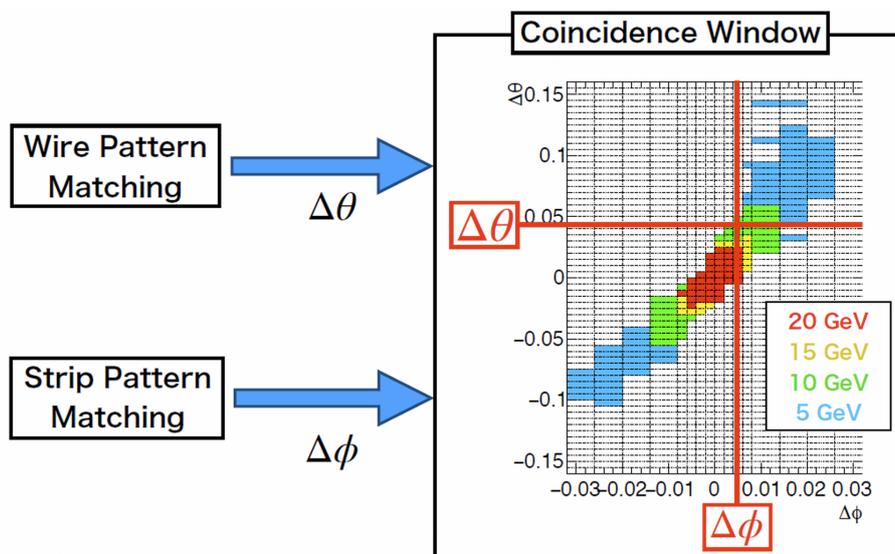


図 3.12: Wire/Strip Coincidence による p_T 閾値判定の概念図 [28]。Wire 側・Strip 側でそれぞれ再構成したセグメントを整合させて $\Delta\theta$, $\Delta\phi$ を得て、Coincidence Window 参照により p_T 閾値判定を行い、ミューオン候補を形成する。

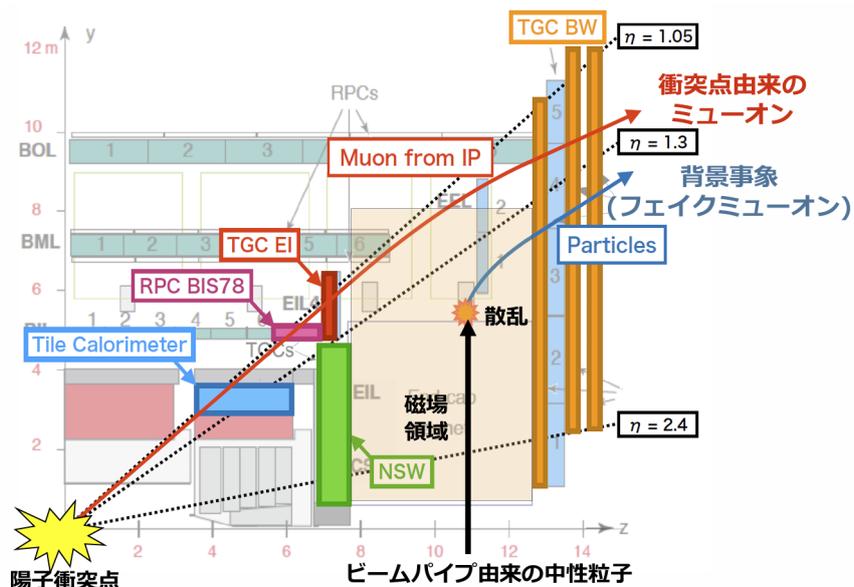


図 3.13: Inner Coincidence の概略 [22]。Big Wheel で得た候補に対し、内層に配置した TGC や NSW 等の情報との整合を要求することで、衝突点起源でない候補を抑制し、候補品質の向上を図る。

3.4.6 Inner Coincidence

Inner Coincidence では、TGC BW コインシデンスで再構成された飛跡候補と、磁場領域内部に配置された検出器 (NSW、RPC BIS78、TGC EI、Tile カロリメータ等) とのコインシデンスをとることで、衝突点に由来しないフェイクトリガーの削減と p_T 分解能の向上を図る。コインシデンスをとる検出器は η 領域に応じて選択され、Unit Region 単位で並列に処理される。例えば、8 UR は Wire Strip Coincidence から 1 候補を受け取り 1 候補を出力し、32 UR は 4 候補を受け取って最大 2 候補に絞ることで、Sector Logic 全体として最大 112 候補を出力する。

3.4.7 Track Selector

Track Selector は、Inner Coincidence から出力される最大 112 個のミュオン候補から、 p_T が高いものを優先して後段に送信する候補を 6 つ選択する。この処理はソーティングロジックとして実装され、Batcher の奇偶マージソート法を用いた並列なソーティングネットワークにより低レイテンシで候補選別を行う。

3.5 高輝度化アップグレードに向けた CW 作成の先行手法

本節では、図 3.12 で述べた CW について、高輝度化に向けたアップグレードに向けてこれまで進められてきた作成手法を説明する。基本的な考え方は、(i) p_T 閾値ごとに $\Delta\theta$ - $\Delta\phi$ のヒットマップを作成し、(ii) それらを積み上げて 1 つの Coincidence Window として整形する、という流れである。

3.5.1 ヒットマップの作成と Window 抽出

本小節では、 p_T 閾値ごとにヒットマップを作り、そこから Window として採用する bin を抽出する手順を述べる。

閾値ごとのヒットマップ作成

まずは単一ミュオンシミュレーションデータを用いてヒットマップを作成する。 p_T 閾値 ± 1 GeV 程度のミュオンを抽出して、各 Unit ごとに、再構成された $\Delta\theta, \Delta\phi$ をハードウェアと同じビニングで 2 次元ヒストグラムに詰め、閾値に対応するヒットマップを得る。

Window 抽出

得られた 2 次元ヒットマップに対して、エントリー数が多い bin から順に足し上げ、累積で 99% のミュオンを含むまで bin を選択することで、閾値に対応する Window を定義する。この操作により、統計揺らぎの大きい低イベント bin を抑えつつ、閾値近傍ミュオンの大部分をカバーする領域をコンパクトに抽出できる。

3.5.2 Coincidence Window の最適化

本小節では、閾値ごとに得た Window 群を 1 つの Coincidence Window へと統合し、運用上扱いやすい形に整形する手順を述べる。

閾値 Window の積み上げ

Coincidence Window は最終的に、入力 ($\Delta\theta, \Delta\phi$) に対しどの p_T 閾値を返すかを一意に決める必要がある。そのため、閾値ごとの Window を高 p_T 側から順に優先して割り当て、未割当の bin に対して低い閾値の Window を追加していくことで、1 つの対応表として統合する。こうして高 p_T 優先の積み上げることによって、同一 bin に複数閾値が競合する場合でも整合的に解決できる。

高 p_T 領域 Window の統合

さらに高 p_T 領域については、複数の閾値 Window を統合して専用の CW を作成する。例えば、5/10/15/20 GeV 閾値ではそれぞれ独立に CW を作成し、40/60/80 GeV といった高 p_T 側はそれ以上の p_T を持つミュオンに対する Window を統合する。これにより、高 p_T 側で Window が小さくなる領域でも Window 形状を維持しやすい。

形状の平滑化

積み上げ後の Window は、統計に限られることによって孤立 bin やそこだけ Window がない部分が残ることがある。そのため、孤立 bin の除去、小さい穴の補完など簡単な平滑化

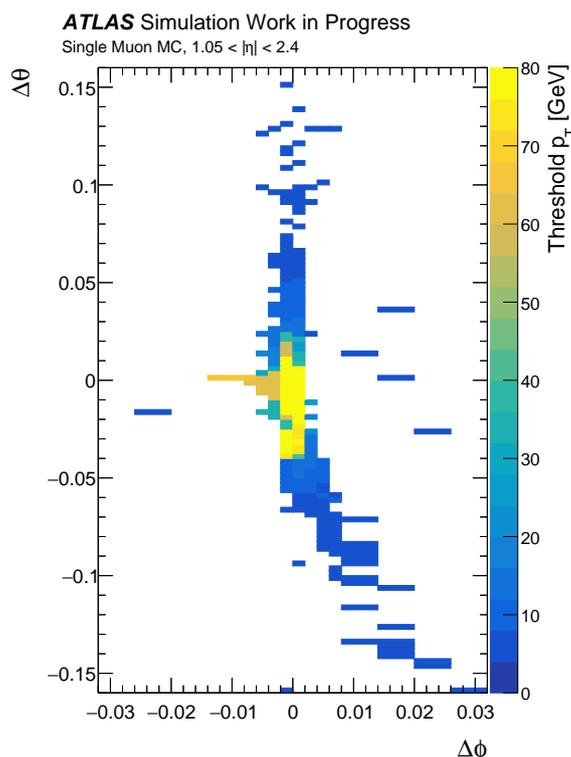


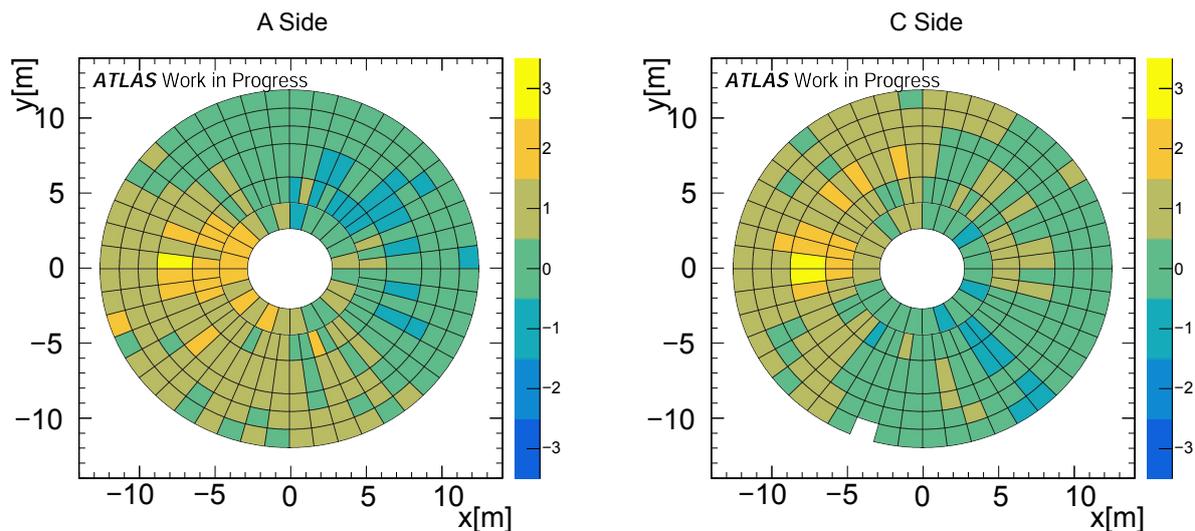
図 3.14: 高輝度化アップグレードに向けた先行手法で作成された Coincidence Window の例。所定の p_T 閾値に対して、 $\Delta\theta$ - $\Delta\phi$ でミュオンに対応する領域を選択し、Coincidence Window として定義する。

を適用し最終的に用いる形状に整える。ここまでの流れで作成された実際のある領域における Coincidence Window を図 3.14 に示す。

3.5.3 先行手法における課題

本小節では、上記の先行手法が高輝度化に向けたアップグレード環境で直面する課題を整理する。

第一に、Coincidence Window における分割領域を細かく取るほど、閾値ごとのヒットマップに必要な統計量が増大する。特に高 p_T 側では単一ミュオンのシミュレーションデータでも統計が得にくく、99% 包含の Window 抽出が不安定になりうる。第二に、CW は検出器アライメントや磁場、物質分布などの影響を受けるため、シミュレーション由来の CW をそのまま実データへ適用すると、領域によっては性能劣化を招く可能性がある。理想的には図 2.6a で示されるように磁場構造が 8 回対称であるため、先行手法では全領域のうち円周方向については $1/8$ の領域 (オクタント) でのみ Coincidence Window を作成してそれを全領域で用いている。しかし実際には、図 3.15 で示す通りチェンバーごとそれぞれに数チャンネル分のずれがある。図 3.15 は半径方向のずれの大きさだが、実際はさらに円周方向やビーム軸方向、加えて傾きなど複雑な影響が考えられる。そのため、先行手法のようにオクタントを考慮した Coincidence Window をそのまま実機用として使うことはできない。



(a) A-side におけるチェンバーずれ補正量の分布 (b) C-side におけるチェンバーずれ補正量の分布

図 3.15: A-side および C-side におけるチェンバー位置ずれ（ミスアライメント）の 2022 年の状態 [29]。各チェンバーの色は、ずれの影響を軽減するために半径方向へ何チャンネル相当の補正をすればよいかを表す。一方で実際のずれは円周方向・ビーム軸方向・傾き等の成分も持つため、単一パラメータでは記述しきれない複雑さがある。

第三に、実データ側で CW を調整しようとしても、ミューオンの p_T スペクトルは一般にフラットではなく、閾値近傍の統計を均一に揃えることが難しい。

これらの課題への対処方法として、現行の Run3 で用いられる機械学習を用いた Coincidence Window の作成手法がある。

3.6 機械学習を用いた Coincidence Window の作成手法

本節では、Run3 で用いられる Coincidence Window を作成する際に用いられた機械学習を用いた手法について説明する。

3.6.1 機械学習

機械学習とは、コンピュータがデータから規則性や知識を獲得し、その結果を用いて予測や分類などのタスクを実行する枠組みである。人間の手でルールを設計する従来手法では、複雑な条件分岐や例外処理が増えるほど調整コストが膨大になるが、機械学習を用いるとデータを与えて学習させることで、同等のルールを自動的に構築できる。

機械学習の学習過程は予測値 \hat{y} と教師信号 y の誤差を損失関数 $L(y, \hat{y})$ で定義し、その平均を最小化するようにモデルのパラメータを更新する最適化問題として表される。以下でこの機械学習についてより詳しく説明する。

3.6.2 教師あり学習

機械学習の代表的な設定として、入力に対する正解（教師信号）が与えられる教師あり学習がある。教師あり学習はさらに、出力が連続値である回帰問題と、離散ラベルである分類問題に大別できる。

回帰問題

回帰問題とは、入力 x に対して連続値 y を予測する問題である。出力が数値で表される量であるため、予測値が正解にどれだけ近いかが性能の基準となる。回帰モデルは、入力と出力の関係が単純な場合から、非線形で複雑な場合まで幅広く扱うことができ、データに含まれる傾向を捉えて数値を推定する。

回帰では、予測と正解のずれを定量化するために何らかの尺度を用意し、その尺度が小さくなる方向へ学習を進める。評価では、平均的にどの程度ずれるか、外れた予測がどの程度生じるか、といった観点で性能を確認することが多い。

分類問題

分類問題とは、入力 x に対して離散的なラベル y を予測する問題である。 $K = 2$ の場合は二値分類、 $K \geq 3$ の場合は多クラス分類と呼ばれる。また、1つの入力に対して複数のラベルが同時に成り立ち得る設定は多ラベル分類として区別される。

分類では、各クラスに属する尤もらしさをモデルが出力し、その中から最も尤もらしいクラスを選ぶ、あるいは尤もらしさが一定以上のクラスを採用するといった形式で予測を行う。

3.6.3 ニューラルネットワーク

ニューラルネットワークは、多数のパラメータを持つ関数について、ユニットを多数接続し層として積み重ねることでそれを再現するモデルである。非線形な入出力関係を柔軟に近似できることが特徴である。

ユニットと層

ニューラルネットワークの基本単位として、入力 x_i に重み w_i を作用させて足し合わせ、バイアス b を加えた量を

$$a = \sum_{i=1}^n w_i x_i + b \quad (3.1)$$

と定義する。この a を活性化関数に通すことで出力

$$y = f(a) \quad (3.2)$$

を得る。活性化関数は線形変換だけでは表現できない非線形性を表現する役割を持ち、ネットワーク全体として複雑な関数を再現するために重要である。

実際のモデルでは、ユニットを複数並べたものを層として扱い、層ごとに線形変換→活性化関数という処理を繰り返す。各層は、直前の層の出力を入力として受け取り、重みとバイアスによる変換の後に活性化関数をとおして次の層へ渡す。この処理を入力層から出力層まで順に適用していく計算を順伝播と呼ぶ。

層数や各層のユニット数を増やすほど表現力は高くなり、複雑な関数を近似しやすくなる一方で、学習すべきパラメータ数が増え、計算量が膨大なものになったり過学習のリスクも高まる。そのため、利用できるデータ量や目的に応じてネットワークの規模を設計する必要がある。

3.6.4 誤差逆伝播と最適化

教師あり学習では、入力に対する出力値の正解が与えられ、モデルの予測と教師信号のずれを損失関数として定義する。回帰問題では、予測値と正解値の差を二乗して平均した値などが代表例であり、予測が正解に近いほど損失が小さくなるように設計される。

学習の目的は、損失が小さくなるようにモデルのパラメータを調整することである。そのために、損失をパラメータで微小に変化させたときの変化量を用い、損失が減る方向へ少しずつパラメータを更新する。このとき、一回の更新でどれだけ動かすかを定める係数が学習率である。学習率が大きすぎると収束せず、逆に小さすぎると収束に時間がかかるため、適切な値の設定や学習途中での調整が重要となる。

ニューラルネットワークではパラメータが多く層も深いため、各パラメータに対する勾配を効率良く求める必要がある。誤差逆伝播は、順伝播で得られた中間量を利用し、連鎖律にもとづいて出力側から入力側へ順に勾配を伝播させることで、各層の重みやバイアスに対する勾配をまとめて計算する方法である。

実装上は、全データを一度に用いて更新するのではなく、データの一部を単位として更新を繰り返すことが多い。訓練データ全体を一巡する学習の単位をエポックと呼び、学習率やエポック数、層構造など学習前に与える設定はハイパーパラメータとして扱う。

3.6.5 汎化と過学習

学習の最終目標は訓練データに対する損失を下げることでなく、未知データに対しても精度良く予測できることである。訓練データでは損失が小さいにもかかわらず未知データで性能が悪化する現象を過学習と呼び、未知データでも性能が保たれる性質を汎化性能と呼ぶ。

過学習を避ける基本的な方法として、データを訓練用・検証用・評価用に分割し、検証用データに対する損失や評価指標を監視しながら学習を行う。例えば、訓練用データによる損失は減少し続ける一方で検証データによる損失が増加に転じた場合、モデルがちょっとした揺らぎなど訓練データの細部まで適合しすぎている可能性がある。そこで、検証損失が改善しなくなった時点で学習を打ち切るなど、学習の進行を制御する必要がある。

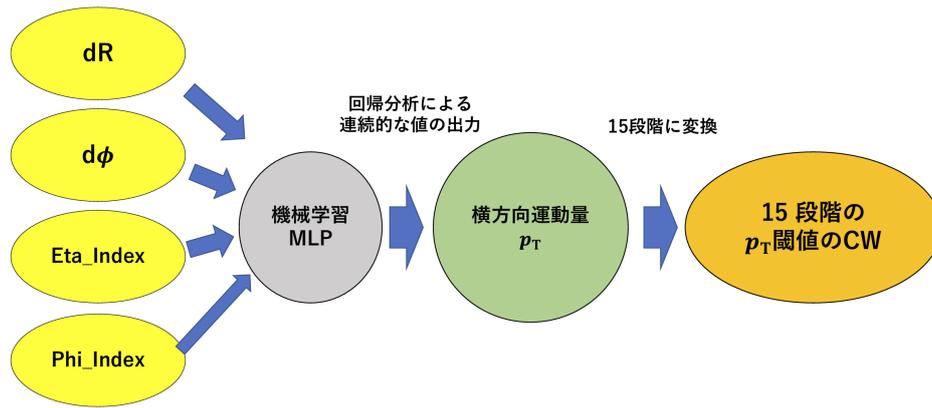


図 3.16: 機械学習を用いた Coincidence Window 作成の流れ [4]。曲がり量・位置情報を入力として p_T を回帰推定するモデルを学習し、推定結果を曲がり量空間へ展開して p_T マップを構成する。得られたマップを閾値で離散化することで、Coincidence Window を作成する。

データ分割による監視とモデル複雑さの制御を組み合わせることで、過学習を抑えつつ汎化性能の高いモデルを得ることができる。

3.6.6 Coincidence Window の作成への応用

本小節では、Run3 の初段ミュオントリガーで用いられた、機械学習により p_T を推定し、その出力を離散的な p_T 閾値へ変換して Coincidence Window を構成する手法の流れをまとめる。先行手法では、閾値ごとにヒットマップを作成し Window 抽出と積み上げ最適化を行ったが、本手法では曲がり量と位置情報から p_T を回帰推定するモデルを学習し、その推定結果を用いて対応表を構成する点が異なる。本手法は先述の通り現行の Run3 で用いられる Coincidence Window の作成手法であり、神戸大学の中村氏の修士論文において詳しくまとめられている [4]。

機械学習を用いた Coincidence Window 作成の概要

機械学習を用いた対応表作成は、概ね

- (i) 学習用データの整形
- (ii) 曲がり具合・位置情報 → p_T 回帰モデルの学習
- (iii) 連続値として出力される p_T 推定値を、実装で用いる離散閾値へ変換

という三段階で構成される。全体の概念図を図 3.16 に示す。

ここで重要なのは、対応表の中身はあくまでトリガーロジックで得られる曲がり量と位置情報と p_T 閾値の対応であり、作成手順こそ異なるものの、トリガーロジックにおける参照の方法は先行手法と同様に対応表の参照として実装できる点である。

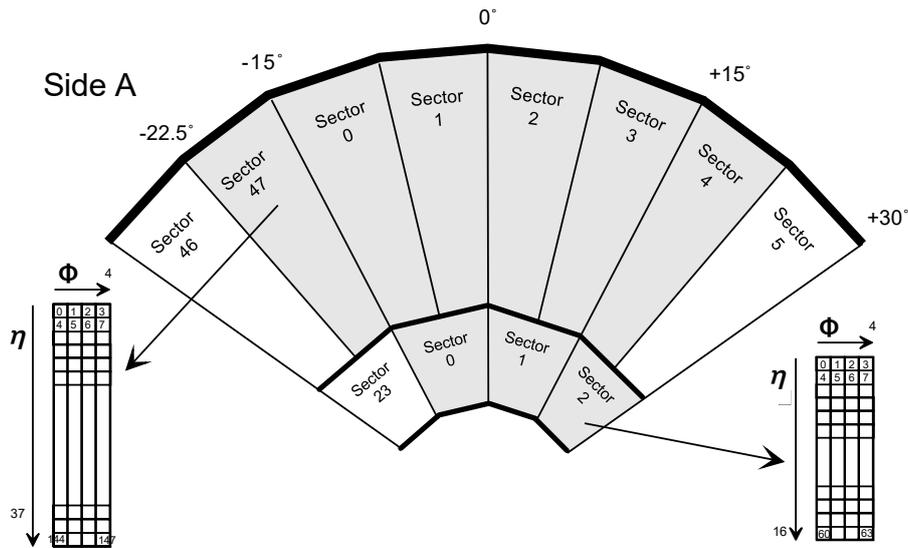


図 3.17: 位置情報の番号付けの模式図 [30]。本研究で用いる領域分割に対応した離散的な位置表現を定義する。

p_T 推測モデルの学習

本手法では、ヒットのあった位置と曲がり量を入力として p_T を回帰推定するモデルを学習する。

このように、トリガー由来の低次元特徴量のみから p_T 推定を学習することで、従来の閾値ごとの Window 抽出・積み上げを、学習器による回帰推定へ置き換えることができる。

学習に先立ち、入力データは学習に適した形式へ整形する必要がある。位置情報については、図 3.17 で示すように隣接する領域が連続する番号になるように読み替えることで、空間的に近い領域の情報が学習で活用されやすくなる。

またトロイド磁場が一様ではないことに対応するため、全領域を単一モデルで扱うのではなく磁場構造を考慮した領域分割を行い、分割領域ごとにモデルを学習する。さらに、多重散乱等により外れた曲がり量を持つ候補が混入すると、本来不要な領域まで対応表が広がりトリガーレート増加に繋がりをうため、孤立 bin 除去等ヒットマップに基づくデータ選別を行ってから学習に用いる。

モデル構造の一例を図 3.18 に示す。入力層と複数の隠れ層からなる中間層、連続値 p_T を出力する出力層からなるモデルで回帰を行う。ハイパーパラメータ（層の数・ノード数・学習率など）は探索により最適化し、過学習を避けるために正規化層やドロップアウトを併用する。

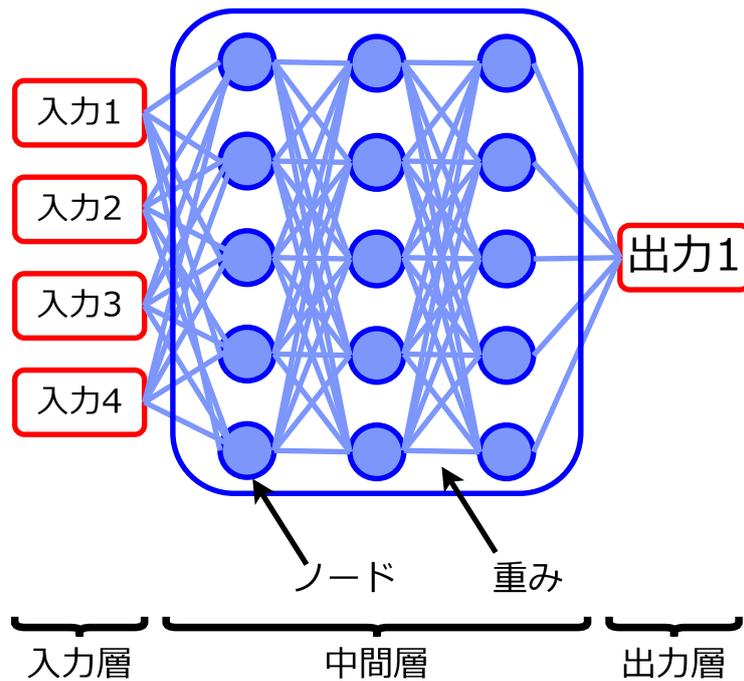


図 3.18: 機械学習手法における p_T 回帰モデルの構造例。曲がり量や位置情報を入力し、複数の中間層を経て p_T 推定値を出力する。

閾値決定による p_T マップの Coincidence Window 化

学習した回帰モデルは、入力に対して連続値の p_T 推定値を出力する。そのため、出力値は閾値段階へと変換して最終的な Coincidence Window を構成する必要がある。

まず、回帰出力の p_T マップの例を図 3.19 に示す。この p_T マップに対して、ある境界値で閾値以上/未滿を判定していくことで、最終的に各 bin へどの閾値段階を割り当てるかを決めた Coincidence Window を得る。

3.6.7 機械学習手法の課題点

本手法は、対応表 Coincidence Window の作成を学習と離散化に置き換えることで、従来のようなヒットマップの穴埋めや段階構造の調整といった最適化手順を統一的に扱える利点を持つ。また、学習に実データを用いる場合には、アライメントや磁場・物質分布の不完全性、検出器応答の揺らぎなど、シミュレーションでは完全に再現しきれない効果をまとめて取り込んだ形で対応表を構成できる点も重要である。一方で、現行の運転においては、領域によって統計量が十分に確保できないことに起因して学習が収束せず、結果として一部領域で効率が低下する事例が確認されている。

加えてアップグレードでは処理単位の粒度向上に伴い、Unit ごとの bin の分割数が増加する。分割数の増加は各 bin に割り当てられる統計量を減少させるため、現行運転で既に顕在化している統計不足の問題は高輝度化に向けたアップグレードにおいてより大きな影響が出ると予想される。

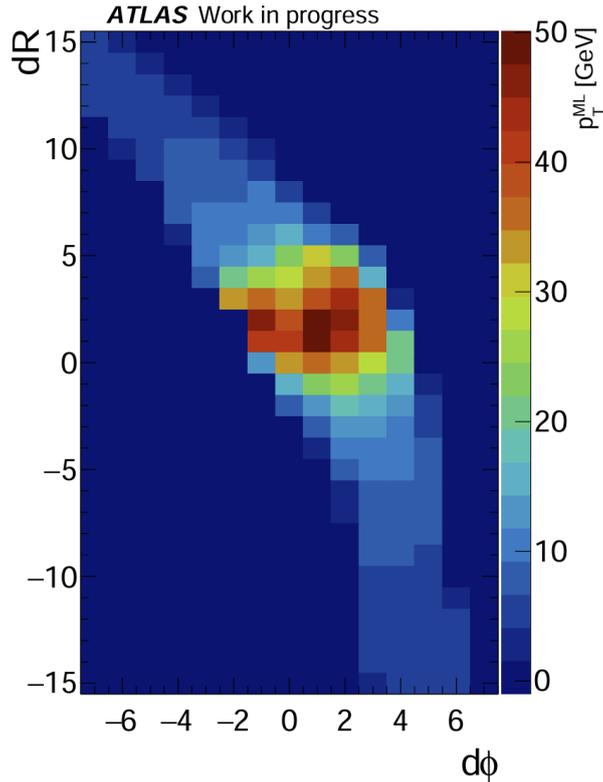


図 3.19: 回帰モデル出力から得られる p_T マップの例 [4]。固定した位置条件のもとで $\Delta\theta-\Delta\phi$ を走査し、各ビンでの推定 p_T を可視化したもの。閾値で離散化することで Coincidence Window を構成できる。

3.7 少統計適応としての転移学習

上記の問題に対し、本研究では転移学習を少統計適応の有効な手段として検討する。転移学習とは、すでに学習したモデルを初期値として用い、元のモデルの学習で用いたデータと似た別の少数データで追加学習を行う手法である [5]。

本研究における Coincidence Window 作成フローを図 3.20 に示す。まずシングルミュオンのモンテカルロシミュレーションデータを用いて $(UR, \Phi, \Delta\theta, \Delta\phi)$ から p_T を推定する回帰モデルを十分に学習させ、ついで実データでモデルの一部を微調整する。これにより、曲がり量と p_T の関係に関する物理的に妥当な表現を初期状態として保持しつつ、ミスアライメントなど実データ固有のずれへ少ない更新量で追従し、少統計領域でも学習を収束させ、対応表の品質を維持することが期待できる。微調整後は、前小節と同様に回帰出力を離散閾値へ変換し、対応表 Coincidence Window を構成する。

3.7.1 転移学習の目的と基本概念

転移学習 (Transfer Learning) は、あるタスクで学習した知識を別のタスクへ移すことで、学習に必要なデータ量や計算量を削減しつつ、目標タスクの性能を向上させる枠組みである。ここでいう知識とは、学習済みモデルの重みや、特徴量表現、あるいは入力の前処理・表現方法など、学習により獲得された情報を指す。

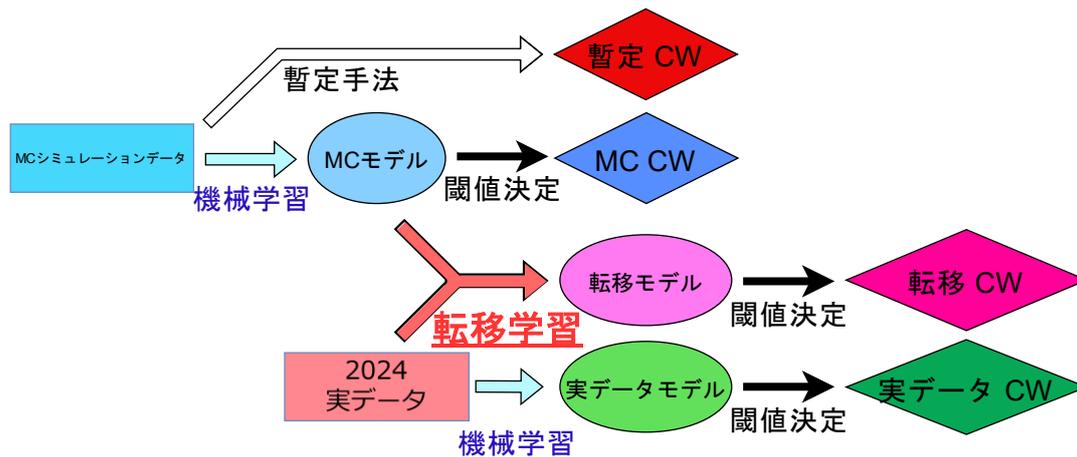


図 3.20: 転移学習を用いた Coincidence Window 作成の処理フロー。Segment Reconstruction で得た (UR, Φ , $\Delta\theta$, $\Delta\phi$) を入力とし、MC シミュレーションで回帰 (p_T)・分類 (電荷) モデルを学習した後、その重みを初期値として実データで転移学習をする。学習済みモデルを用いて $\Delta\theta$ - $\Delta\phi$ 格子点上の推論から p_T (および電荷) マップを作成し、所定の閾値へ写像することで各閾値の Coincidence Window を定義する。比較のために、実データにおいても転移学習を用いない、機械学習手法でモデルと Coincidence Window の作成も行う。

転移学習を考える際には、元タスクと目標タスク、およびそれぞれで用いる元ドメインと目標ドメインを区別する。一般に、ドメインは入力データの分布を含み、タスクはなにを学習させたいかを含む概念として整理できる。

ただし、元ドメインと目標ドメインの差が大きすぎる場合、元タスクで獲得した表現が目標タスクに適さず、学習がかえって悪化することがある。これを負の転移と呼ぶ。そのため転移学習では、いつ転移するか、何を転移するか、どう転移するかを意識し、目標ドメインに適応させる範囲や強さを適切に設計する必要がある。

3.7.2 深層学習における転移

深層ニューラルネットワーク (DNN) は、多段の非線形変換を通じて入力から出力への写像を構成する。このとき、モデル全体を特徴抽出部とタスク固有部に分けて考えると、転移学習の設計が明確になる。概念的には、入力 x に対する予測 \hat{y} は

$$\hat{y} = f(x) = g(h(x)) \quad (3.3)$$

のように、特徴抽出部 $h(\cdot)$ (中間表現への変換) と、その表現から目的変数を出力するヘッド $g(\cdot)$ の合成として記述できる。DNN では経験則的に入力に近い層ほど比較的一般的な特徴を、出力に近い層ほどタスク固有の特徴を表現すると解釈されることが多く、この性質を利用して一部の層を再利用することで、目標ドメインのデータが少なくても学習を成立させられる。

一方で、ドメイン差が大きい場合には、特徴抽出部そのものの表現が目標ドメインに適さなくなる可能性もある。したがって、どの層までを再利用し、どの層を再学習するかは、ドメイン差の大きさと目標データ量のバランスに依存して決める必要がある。

3.7.3 代表的な転移学習の戦略

転移学習を実装する代表的な戦略は、学習済み重みを初期値として流用し、再学習の範囲と強さを制御することである。特に深層学習では、どの層を更新するか、どれだけ重みを更新するか、過学習や負の転移をどう抑えるかという観点で整理できる。

層の凍結とファインチューニング

転移学習の実装上の代表的な戦略として、以下のようなものがある。

- **全層ファインチューニング**：初期値だけ流用し全層を学習する
- **下位層の凍結**：特徴抽出に相当する下位層を固定し、上位層のみ学習する
- **出力層の再初期化**：出力層のみをランダム初期化し直し、適応を促す

全層ファインチューニングは表現の自由度が高く、ドメイン差が大きい場合にも適応できる可能性がある一方、目標データが少ないと過学習や学習が収束しないといった問題を招きやすい。下位層の凍結は更新すべきパラメータ数を減らし、少統計でも学習を収束させやすい反面、ドメイン差が大きいと十分に補正しきれない場合がある。ヘッダの再初期化は、目標ドメインでの出力分布やラベル定義が実質的に変化している場合に有効だが、本研究のようにタスク自体 (p_T 推定・電荷推定) が同じ設定では、必ずしも毎回必要になるとは限らない。

学習率・エポック・正則化の取り扱い

転移学習では、学習済みモデルがすでに一定の性能を持つため、ゼロから学習する場合と同じ設定で最適化を進めると、短時間で過学習したり、獲得済みの有用な表現を壊してしまうことがある。このため、学習率を小さくする、学習エポックを短めにし早期終了を用いるといった制御が有効である。

また、下位層を凍結し上位層のみを更新する場合、更新パラメータが少ないことから比較的強い正則化を入れても学習が破綻しにくい。一方で全層ファインチューニングを行う場合には、重み減衰やドロップアウト、検証損失に基づくアーリーストップなどを併用し、目標ドメインへの適応と汎化のバランスを取る必要がある。さらに、転移元の重みから大きく逸脱しないような制約を入れる考え方もあり、少統計における学習の収束に寄与しうる。

負の転移と回避

負の転移は、ドメイン差が大きい、転移する表現が目標タスクに不適切、目標データが極端に少ないといった条件で起こりやすい。実務的には、転移前のモデルよりも転移後の検証性能が悪化する、特定の領域でのみ誤差が増える、といった形で現れる。

負の転移を避けるための基本方針は、ドメイン差を小さく見せる工夫と転移の強さを制御する工夫に分けられる。前者は、入力変数の定義やカット、欠損の扱いなど前処理をMCと実データで可能な限り揃えることが重要である。後者は、凍結範囲を広げて更新自由度を下げる、ヘッドのみを更新する、あるいは学習率を下げて更新量を抑えるといった手段がある。また、目標ドメインでの検証指標を監視し、改善が止まった時点で学習を打ち切ることも有効である。

3.7.4 転移学習による実データのためのファインチューニング

実データの学習で扱う状況は、タスク自体が同一である一方、入力の分布がシミュレーションと実データで異なる、いわゆるドメイン差を伴う設定に対応する。加えて、Unit ごとに見ると実データの統計量が限られ、領域単体でモデルをゼロから学習すると過学習や学習が収束しないといった問題が起きやすい。このような少統計とドメイン差の二つが、転移学習を導入する主たる動機である。

転移元のモデルはMCシミュレーションにより曲がり量と p_T の対応や符号に依存した曲がり方向といった物理的に妥当な対応関係を大規模データで学習し、それを初期値として実データへ適応させる。ここで実データ側で必要となるのは、理想配置からのずれなどに起因する分布の変化を吸収するための微調整であり、モデルを一から構築するよりも、学習済み表現を出発点にしたほうが少統計下でも学習を収束させやすい。

本研究では、MCで学習したモデルを初期値としつつ、実データ側では少統計でも過度に重みが変わらないよう更新範囲と学習設定を調整することで、物理的に妥当な対応関係を保持したまま実データへの適応を行うことを目指す。

MCと実データの差がモデル出力に与える影響

MCで学習したモデルを実データへそのまま適用した場合、領域ごとにモデル出力の分布が系統的に変化することがある。これは、実データにおける検出器のミスアライメントや応答差により、同一の物理量(例えば同一の真の p_T) に対して Segment Reconstruction が出力する $(\Delta\theta, \Delta\phi)$ の分布がMCと一致しないことに起因する。

概念的には、MCで学習した回帰モデルを

$$p_T = f_{MC}(UR, \text{Phi}, \Delta\theta, \Delta\phi) \quad (3.4)$$

と書くと、実データでは入力分布が

$$(\Delta\theta, \Delta\phi)_{\text{data}} \approx (\Delta\theta, \Delta\phi)_{MC} + \delta(UR, \text{Phi}) \quad (3.5)$$

のように領域依存のずれ δ を持ちうる。このとき同一事象に対する予測値も

$$p_{T,\text{data}} = f_{MC}(UR, \text{Phi}, (\Delta\theta, \Delta\phi)_{MC} + \delta) \quad (3.6)$$

となり、結果として p_T 分布が領域ごとに偏る。

特に高 p_T 領域では、磁場による曲がり小さく ($\Delta\theta, \Delta\phi$) の変化量自体が小さいため、わずかな系統ずれでも見かけ上の曲がり量の解釈が変わりやすい。その結果、モデル出力の高 p_T 側の位置が、領域ごとのずれに応じてシフトして見えることがある。これは、実機での閾値判定に対しては有効な p_T 閾値が領域ごとにずれることに相当し、CW を設計するうえで無視できない影響を与える。

3.8 本研究の目的

本章では、初段ミュオントリガーにおける TGC の役割、処理単位、および Sector Logic 上のトリガーロジックを述べ、高輝度化に向けたアップグレードに向けた Coincidence Window 作成の先行手法とその課題を整理した。先行手法では、bin の細分化に伴う統計要求の増大や、実データへの適用におけるずれへの追従が課題となる。

これに対し、機械学習を用いる手法は、最適化を含む対応表作成を学習問題として統一的に扱えること、さらに実データを用いた学習により諸々のずれをまとめて取り込めることから、アップグレードに向けた有力な選択肢である。一方で、現行運転において既に一部領域で統計不足に起因する効率低下が確認されており、高輝度化に向けたアップグレードでの Unit 領域の粒度向上によりこの問題はより深刻化することが予想される。

そこで本研究では、まずシミュレーションデータを用いて機械学習による学習と Coincidence Window の作成が成立するかを確認し、先行手法の課題に対して有効であるかを検証する。さらに、統計が限られた条件でも実データへ適応できるようにシミュレーションで事前学習したモデルを基に転移学習を適用し、少ない実データでも Coincidence Window の性能が出せるかを評価する。これらの具体的な方法と結果は、次章で詳述する。

第4章

転移学習を用いた Coincidence Window の作成

本章では、第3章で述べた先行手法の課題に対処するため、転移学習を用いた Coincidence Window の作成手法を説明する。まず、モンテカルロ (MC) シミュレーションデータを用いた p_T 回帰モデルの学習と評価を示し、続いて実データへの転移学習とその評価を述べる。最後に、同様の枠組みで学習した電荷判定モデルについて述べる。

4.1 機械学習の学習設計

本節では、 p_T を出力する回帰型の機械学習と、電荷を出力する分類型の機械学習の両モデルに共通する学習設定をまとめる。本研究における全体フローは図 3.20 に示した通りである。

まずデータの形式と前処理を述べた後、機械学習・転移学習の双方に共通するモデル構成と学習設定を示す。

4.1.1 用いるデータ

学習に用いる入力、アップグレード後の Sector Logic ボード動作を再現する Segment Reconstruction の出力である。各ミューオン候補について、領域を表す位置情報 (UR, Phi) と、曲がり量 ($\Delta\theta, \Delta\phi$) が得られる。これらを機械学習モデルへの入力特徴量とし、教師信号として p_T (回帰) および電荷符号 (分類) を与える。

シミュレーションデータ

シミュレーションには、衝突点から単一のミューオンを打ち出すシングルミューオンデータを用いた。生成条件は、擬似ラピディティが $-2.4 < \eta < +2.4$ の範囲、 ϕ 方向は一周一様とし、 p_T は図 4.1 で示す通り 0–100 GeV の範囲でほぼフラットに与えた。生成イベント数は約 2.0×10^7 事象であり、このうち TGC にヒットを残して Segment Reconstruction の出力が得られたおよそ 8×10^6 ミューオンを抽出して学習に用いた。

実データ

実データには、2024 年に取得された Run3 実データを用いた。オフラインで再構成されたミューオンに対し、アップグレード後の Sector Logic ボード動作をシミュレートするソ

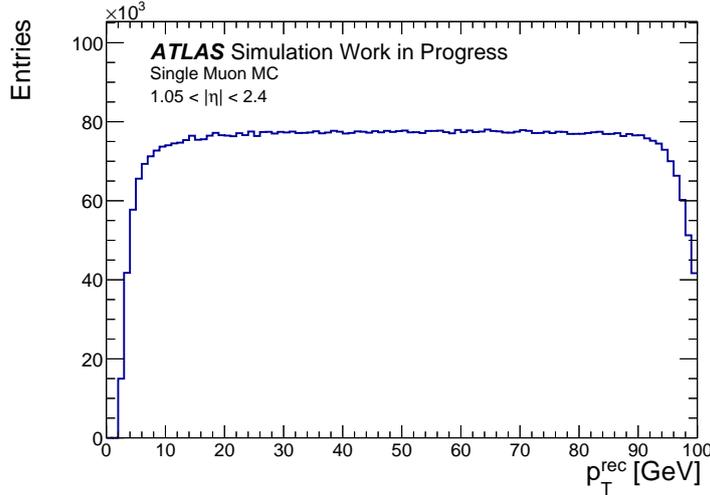


図 4.1: シングルミュオンMCシミュレーションデータにおける真の p_T 分布。生成時には $p_T = 0\text{--}100$ GeV をほぼフラットに与え、TGC にヒットを残して Segment Reconstruction の出力が得られた事象を学習に用いる。図は代表的な (UR, Phi) 領域について示しており、ベースモデル学習時のサンプル重み付け等で参照する。オクタント対称性により 1/8 の領域に集約して学習に用いることができるため、実際の統計量はこの 8 倍にみなせる

ソフトウェアを用いて Segment Reconstruction の出力 (UR, Phi, $\Delta\theta$, $\Delta\phi$) を作成し、学習用データとして整形した。

また、実データではトリガー条件に由来するバイアスを抑え、比較的純粋なミュオンデータを得るために、 $Z \rightarrow \mu\mu$ 事象を用いた Tag and Probe 法により事象選別を行った。これにより、 p_T 分布は図 4.2 に示すように、MC シミュレーションとは異なる分布をとっている。

4.1.2 データの処理

本節では、前節で述べたシミュレーションデータおよび実データを、機械学習の入力として用いるために施す前処理を述べる。本研究の学習は各 Unit ごとに、曲がり量 ($\Delta\theta$, $\Delta\phi$) と p_T の対応を学習する構成である。したがって、各 Unit 内のヒット分布に基づき学習に用いる ($\Delta\theta$, $\Delta\phi$) の bin を選別し、さらに空 bin に対するダミー情報の付加を行うことで、統計不足領域における誤った外挿を抑制する。

ヒット分布に基づく学習 bin の選別

本研究では、($\Delta\theta$, $\Delta\phi$) 平面を Coincidence Window と同一の binning で量子化し、各 bin を学習の基本単位として扱う。しかし、分布の端に現れる極端にまれな bin や、多重散乱などに起因する局所的な外れ値的ヒットまで機械学習に与えると、学習が収束しにくくなったり、モデルが本質的でない特徴を学習してしまう可能性がある。そこで、各 Unit ごとに、ヒット分布に基づいて学習に用いる bin をあらかじめ選別する。

具体的な手順は以下の通りである。

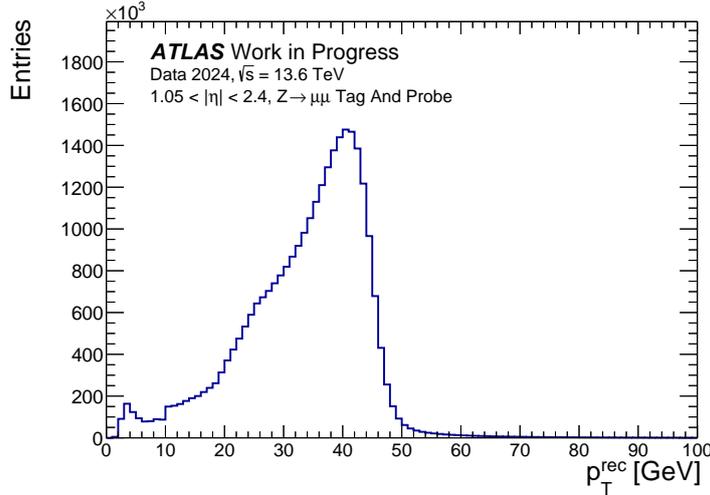


図 4.2: 2024 年 Run 3 実データから Tag and Probe 法により選別した $Z \rightarrow \mu\mu$ 事象の p_T 分布。オフライン再構成ミューオンに対して Sector Logic の動作をシミュレートした Segment Reconstruction 出力を作成し、学習用に整形した。MC とは異なる p_T 分布をとるため、実データでの転移学習ではこの分布差を考慮して学習を行う。

1. **1 GeV ごとに分割した二次元ヒストグラム作成:** データのミューオンを 0 ~ 1 GeV, 1 ~ 2 GeV, ……というように 1 GeV 区切りで分け、それぞれについて $(\Delta\phi, \Delta\theta)$ の二次元ヒストグラムを作成する。以下で掛ける選別を p_T に対して均一に行うことを目的とする。
2. **統計量に基づく bin の除去:** 各帯域のヒストグラムについて、極端に統計量が少ない bin を除く。実装上はその帯域の総ヒット数に対して十分なイベント数が入っていない bin を除外する形にしており、まれにしか現れない外れ値的ヒットの bin を学習から取り除くことが目的である。
3. **孤立 bin の除去:** 占有率カットを通過した bin のうち、周囲に連続して残る bin がほとんど存在しないものを除く。具体的には、注目 bin の近傍 (3×3) に残存 bin が一定数以上あることを要求し、単発で飛び出した bin を学習に使わないようにする。
4. **各 p_T ヒストグラムの統合:** これらのカットにより残ったイベントを統合して学習対象として採用する。こうすることで、いずれの帯域でも平等な条件の下で多重散乱などの外れ値的ヒットを抑制することができる。

以上により得られた学習対象 bin に属する事象のみを用いて学習データを構成する。以降で述べる空 bin へのダミーイベント付加も、この選別結果を基準として行う。

空 bin へのダミーイベント付加

前項のカット後も、学習対象外となった bin や、そもそもイベントが存在しない bin が多数残る。これらの bin に対して制約を与えずに回帰モデルを学習すると、学習データが存在しない領域で実際とは異なる出力となり、結果として Coincidence Window 構成時に不

要な窓が生成される可能性がある。この不要な窓の生成を抑えるため、本研究では空 bin の一部に低い p_T を持つダミーイベントを付加する。

ダミーイベントに付与する p_T は TGC に到達しないほどに低い p_T を用いる。この処理により、データが存在しない領域に対して十分低い p_T という意味付けを与え、モデルの不必要な外挿を抑制する。

ただし、空 bin 全てにダミーイベントを付加すると、学習データがダミーで支配され、入力に依らず低 p_T へ収束するなどの負作用が生じ得る。そこで、各 Unit において、カット後に残ったイベント数に対する一定割合だけ空 bin をランダムに重複なしで選び、選択した bin の中心座標にダミーイベントを 1 つずつ付加する。

入力特徴量の量子化と正規化

学習入力は 1 イベントあたり $(UR, \phi, \Delta\theta, \Delta\phi)$ の 4 変数である。UR と ϕ は領域 ID として既に離散化されているため、最大 ID で割ることで $[0, 1]$ に規格化する。一方、 $\Delta\theta$ と $\Delta\phi$ は Coincidence Window と同一の binning に量子化した上で、bin 中心を 0 とする整数 bin 番号へ写像し、最終的に $[-1, 1]$ へ正規化して入力とする。目的変数である p_T は、100 GeV で規格化した無次元量として学習し、推論時に元へ戻す。

学習・検証・評価用データへの分割

正規化後のデータは、モデルの汎化性能を確認するため、訓練用・検証用・評価用に分割する。分割後は、訓練用データのみを用いてパラメータ更新を行い、検証用データで過学習の兆候を監視しつつ学習停止や学習率調整を行う。評価用データは最終的な性能見積りにのみ用いる。

4.1.3 モデル設計

本節では、今回の学習におけるモデルの設計について述べる。

用いるツール

機械学習は行列演算や自動微分、CPU や GPU などの計算器資源の利用、学習ループの管理など 0 からやろうとすれば非常に高度なシステムづくりが必要になる。そこで、関数やクラスなどをまとめたライブラリや、アプリケーション全体の処理フローを提供してくれるフレームワークを用いて機械学習用の環境を作成した。

機械学習分野の代表的なツールとして、TensorFlow/Keras、PyTorch、JAX などがある [31, 32]。TensorFlow はオープンソースのフレームワークとして、非常に便利なことに周辺ツールも含むエコシステムが整備されている [31]。また CPU、GPU、TPU など複数デバイス上で計算を実行でき、自動微分を用いた学習が可能である。Keras は TensorFlow のためのライブラリで、モデル構築から学習・評価までを簡潔に記述でき、迅速な試行

錯誤に適したインターフェースを提供してくれる。本研究ではモデル実装および学習に TensorFlow と Keras を用いる。

ネットワーク構造とハイパーパラメータ

ハイパーパラメータとは、学習によって最適化される重みとは異なり、学習の前に人の手で与える設定値である。例えば、学習率、バッチサイズ、学習エポック数上限、ネットワークの層構造、正則化の強さなどが挙げられる。

本研究の p_T 回帰では、入力 (UR, Phi, $\Delta\theta$, $\Delta\phi$) を受け取り、正規化後の p_T を 1 出力する多層パーセプトロンを用いた。入力側には正規化層を置き、その後に複数の全結合層を積み重ね、各層の活性化関数には ReLU を用いる。過学習の抑制と学習が収束するようにするため、各層の間に Dropout を導入した。出力層は回帰として 1 ノードを持ち、非物理的な負の出力を避けるため非負制約を持つ活性化を用いる。

以下に主要な設定の例を示す。

- エポック数上限：400
- バッチサイズ：1024
- 初期学習率： 3×10^{-4}
- 隠れ層ユニット数：[256,512,1024,1024,512,256]
- Dropout 率：0.01

ここで、エポックとは訓練データ全体を一巡する学習の単位であり、バッチサイズは一度の重み更新に用いるサンプル数である。エポック数上限と学習率の初期値は固定値として与えるだけでなく、学習の進行状況に応じて自動調整されるようにした。そのためには学習が改善したかどうかを判定する基準が必要であるため、次小節で損失関数と評価指標を述べる。

損失関数と評価指標

ニューラルネットワーク学習では、教師信号 y とモデル出力 \hat{y} のずれを損失関数 $L(y, \hat{y})$ で定義し、その平均値が小さくなるように重みを更新する。本研究では p_T を p_T^{\max} で割って正規化し、

$$y = \frac{p_T}{p_T^{\max}} \in [0, 1] \quad (4.1)$$

として回帰を行った。

損失関数には、相対誤差に対する Huber 損失を用いた。相対誤差 e を

$$e = \frac{\hat{y} - y}{\max(|y|, \varepsilon)} \quad (4.2)$$

と定義したうえで、 $\delta > 0$ を閾値として

$$L_{\text{Huber}}(e) = \begin{cases} \frac{1}{2}e^2 & (|e| \leq \delta) \\ \delta(|e| - \frac{1}{2}\delta) & (|e| > \delta) \end{cases} \quad (4.3)$$

で与える。ここで ε は y が極小のときに分母が過小になって発散することを防ぐための安全項である。この損失は、平均二乗誤差に比べ外れ値の影響を受けにくく、また相対誤差を用いることで低 p_T 領域の精度を重視した学習となる。後述する学習率調整およびアーリーストップでは、検証用データに対する損失が改善したかどうか、つまり最小値を更新したかどうかを基準とした。

学習率の自動調整

機械学習では、[図 4.3](#)に示すように損失が小さくなる方向へ重みを更新していく。このとき一回の更新で進む矢印の長さが学習率である。学習率が大きすぎると最小値あたりで振動し、小さすぎると最小値の近くまで行きつくのが遅くなる。そこで本研究では、損失の改善が一定期間見られない場合に学習率を段階的に下げるスケジューリングを導入した。具体的には、損失があるエポック数だけ更新されない場合に学習率を $1/2$ 倍していく。ただし、最小学習率を 10^{-6} とした。これにより、学習序盤では大きめの学習率で素早く改善し、収束に近づいた段階では学習率を下げて微調整することで最適な重みに早く収束する。

アーリーストップ

エポック数を増やすと訓練データに対する損失は単調に下がりやすい一方、過学習により検証データに対する損失がある時点から悪化することがある。そのため、検証データの出す損失が最小となる時点をもっと良い学習結果とみなし、そこで学習を打ち切るアーリーストップを用いた。

本研究では、損失が改善しない状態が 20 エポック続いた場合に学習を停止するようにした。また、学習停止時には学習終了時点の重みではなく最良点の重みを最終モデルとして採用した。

4.2 モンテカルロシミュレーションデータを用いた機械学習

今回用いるモンテカルロシミュレーションデータは理想配置に基づくため、 ϕ 方向に 8 回の対称性 (オクタント対称性) を持つ。したがって、学習はこのオクタント対称性を用いて領域を折り返し、全領域のうち一部の領域のみで行う。このとき、単純に全周の ϕ 番号をそのまま入力へ与えると、オクタント境界で入力表現が不連続となり、転移学習で別オクタントへ展開した際に、同等の幾何学条件であっても入力の見え方が変化してしまう。そこで本研究では、 ϕ 方向の位置情報としてオクタント内での相対的な ϕ_{ID} を用いる。す

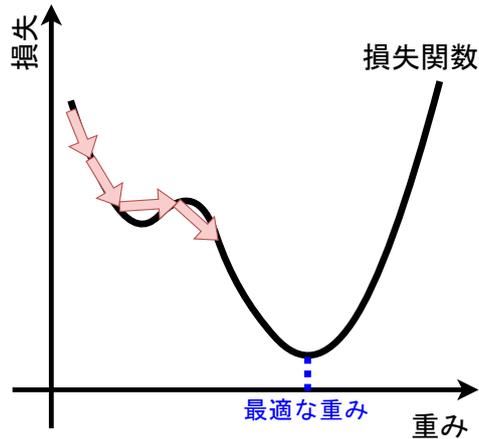


図 4.3: 勾配降下法によるニューラルネットワーク学習の概念図。損失関数のグラフ上で勾配方向へパラメータを更新し、矢印の長さが学習率に対応する。

なわち、各オクタント内で ϕ_{ID} を振り直し、クラスタ内の局所座標として正規化した値を入力に用いることで、オクタント折り返しと整合した入力表現を与え、後段の実データ転移学習における初期モデル割り当てが破綻しないようにした。

4.2.1 MC データの学習結果

モンテカルロシミュレーションデータを用いて学習した回帰モデルの出力例を図 4.4 に示す。左図は、学習に用いた入力マップ ($\Delta\theta$ - $\Delta\phi$ 平面上の分布) を例示したものである。十分な統計量がないために穴あきや孤立した島状の bin などの構造が残る。一方で、右図で示すような学習後にモデルから得られる出力では、入力が疎であった領域に対しても周辺の情報から補間的に推定が行われ、全体の勾配や構造を保ったまま滑らかな p_T 分布が得られていることが確認できる。このことは、学習が単なる局所的な当てはめに留まらず、入力特徴量と p_T の対応関係を領域内で獲得していることを示唆する。

4.2.2 機械学習の性能評価

学習した回帰モデルの基本性能を、検証データに対する予測値と真値の対応、および残差分布により確認する。

まず図 4.5 に、検証データにおける回帰モデル出力 ($\text{Pred } p_T$) と真値 ($\text{Actual } p_T$) の対応を示す。低 p_T 領域では概ね単調な対応が得られており、予測値が真値を追従していることが分かる。一方で高 p_T 側では、曲がり量が小さくなることに加えて入力が量子化されているため、真値の変化に対して予測値の分離が難しくなり、対応が次第に鈍る傾向が見られる。以降では、この傾向をより定量的に見るため、 $1/p_T$ 表現での残差分布を用いて評価する。

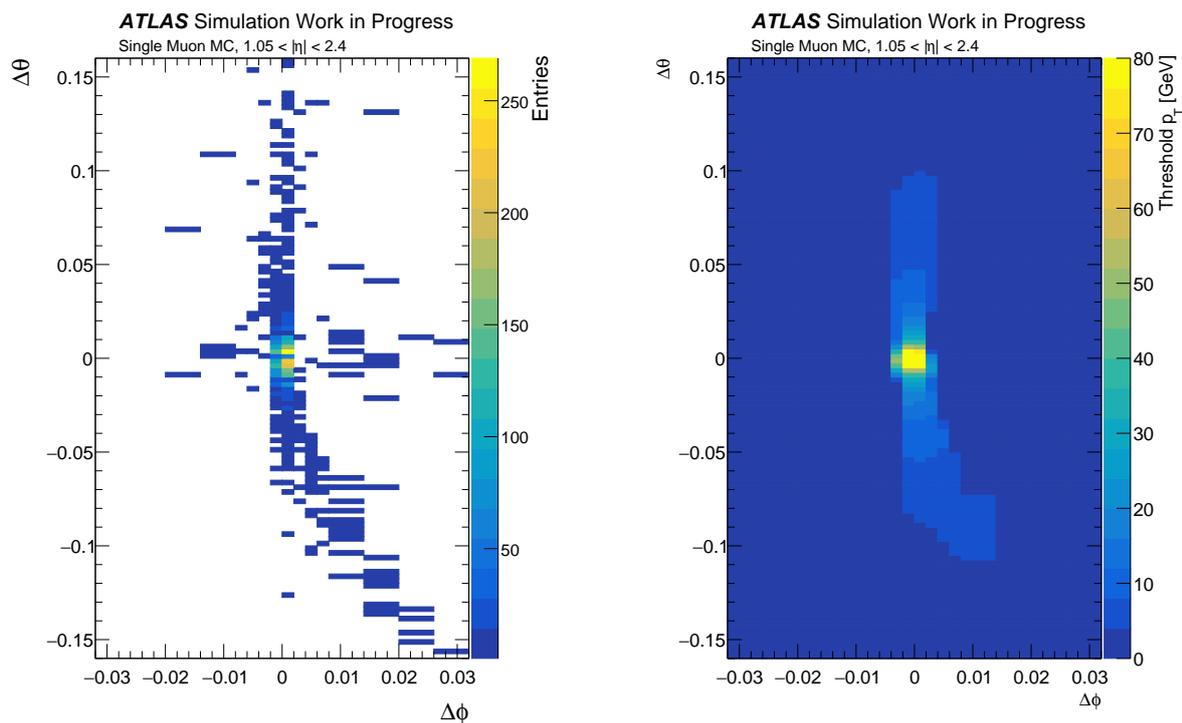


図 4.4: 機械学習モデルの p_T 出力値マップの例。左図機械学習の入力となったデータのヒットマップの例を、右図はある学習領域における機械学習モデル出力値の分布を示す

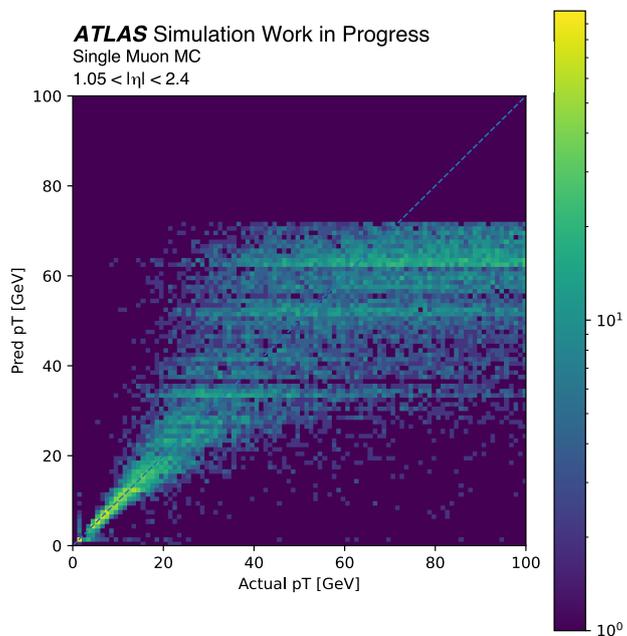


図 4.5: 検証データにおける回帰モデル出力 (Pred p_T) と真値 (Actual p_T) の対応。低 p_T 側では追従が良い一方、高 p_T 側では分離が難しくなり対応が鈍る傾向が見られる。

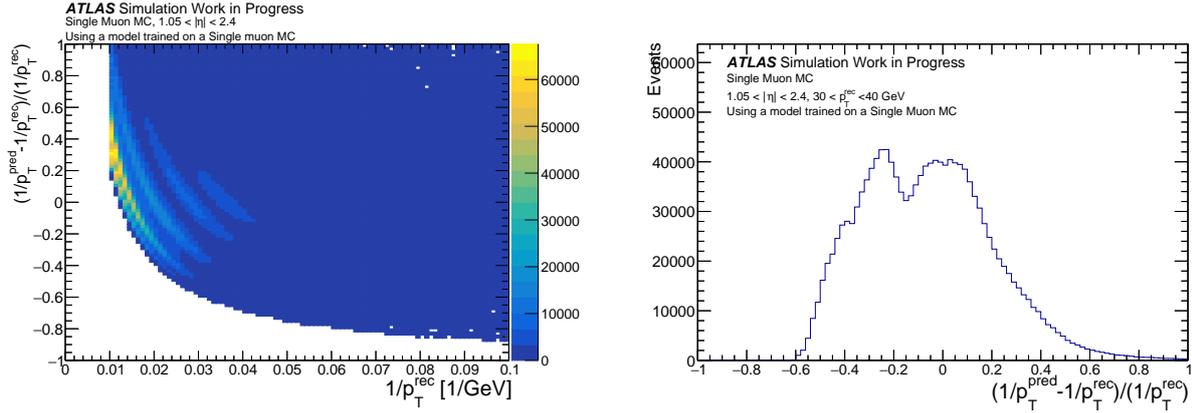


図 4.6: 機械学習したモデルに検証用のシミュレーションデータを入れたときの残差。左図は残差の二次元分布を、右図は 30 GeV から 40 GeV の範囲で区切った一次元ヒストグラムを示す。0 を中心に分布するものの高 p_T 部分は入力が量子化されている影響でいくつかの構造が見える

| | 10 GeV | 20 GeV | 30 GeV | 40 GeV | 50 GeV |
|--------|--------|--------|--------|--------|--------|
| MC データ | 0.11 % | 0.19 % | 0.27 % | 0.27 % | 0.29 % |
| 実データ | 0.24 % | 0.33 % | 0.44 % | 0.48 % | 0.51 % |

表 4.1: 機械学習したモデルに検証用のシミュレーションデータと実データを入れた際の残差分布における p_T^{rec} に対する σ の比較

残差の定義として、真値と予測値を $1/p_T$ へ写像した上で

$$r \equiv \frac{(1/p_T^{\text{pred}}) - (1/p_T^{\text{true}})}{(1/p_T^{\text{true}})} \quad (4.4)$$

を用いる。

図 4.6 に、全領域をまとめた残差 r の二次元分布と、 r の一次元分布を示す。一次元分布では、残差が 0 以外の部分でもいくつかの構造をしていることが見て取れる。二次元分布を見るとこの構造が $1/p_T^{\text{true}}$ が小さい領域で広がっていることが分かる。これは高 p_T では曲がりの情報が量子化されている影響で、同じ入力に対する p_T の分離が難しくなるためであり、推定の不確かさや系統的な偏りが増大しやすいことに対応する。

図 4.7 に、残差分布の mean 値と σ を $1/p_T^{\text{true}}$ の関数として示す。検証用の MC では mean 値は概ね $r = 0$ 付近に保たれている一方で、高 p_T 側に向かうにつれて σ が増大する傾向が確認できる。 σ の具体的な値は表 4.1 に示す通りであり、表に示した点でも低 p_T 側より高 p_T 側で σ が大きく、高 p_T 側で分解能が悪化する傾向が確認できる。また同表には、同じ MC 学習モデルを実データに適用した場合の σ も併せて示しており、MC に比べて分布幅が大きくなる。これは、MC と実データの差が残差の平均・幅として現れることを示しており、実データに対しては追加の補正が必要であることを示唆する。

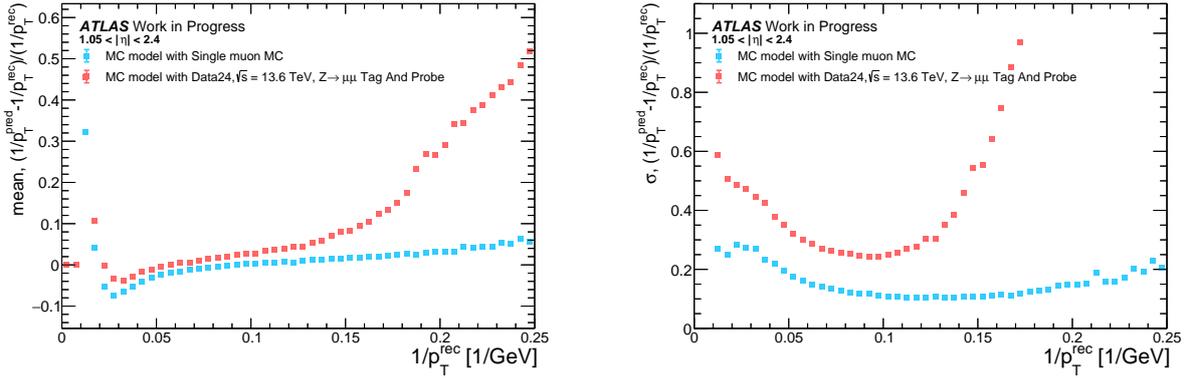


図 4.7: 機械学習したモデルに検証用のシミュレーションデータと実データを入れた際の残差における mean 値と σ の比較。左図は MC モデルの残差分布における mean の比較を、右図は MC モデルの残差分布における σ の比較を示す

4.3 実データを用いた転移学習

本節では、転移学習の手法について述べる。特に、MC データをベースにしたモデルの学習との差異やモデルの利用方法、学習設定について詳しく説明する。

4.3.1 モデル構造と更新の方針

転移学習に用いるネットワーク構造は MC 学習と同一であり、MC 学習で得た重みを初期値として実データで追加学習を行う。ただし、転移学習では学習率を大きく与えると逆に局所解に陥るといった問題が起きるため、学習率や学習の打ち切り条件などの運用上の設定は MC 学習とは独立に与える。本研究では、まず基準設定として全層を再学習する転移学習モデルを用い、MC から作成した Coincidence Window および実データのみで学習した機械学習方式の Coincidence Window との比較を行う。さらに、転移学習で再学習する層の範囲を変更した場合についても比較し、議論する。

実データ転移学習における初期モデルの割り当て

本研究では、MC は理想配置に基づくため、学習をオクタント対称性を用いて一部領域に限定して行っている。一方で、実データの転移学習は全領域で実施する必要がある。そこで、図 4.8 に示すように実データの各 ϕ 領域 (クラスタ) に対してオクタント対称性により対応づけられる MC 側の領域を選び、その MC モデルを初期モデルとして割り当てる。

具体的には、実データ側の ϕ クラスタ番号を、オクタントあたりのクラスタ数で割った剰余を用いて対応する MC 側クラスタを決定し、そのモデル重みを読み込んだ上で転移学習を行う。この手順により、MC で直接学習していない領域についても、対称位置の MC モデルを出発点として学習を開始できる。

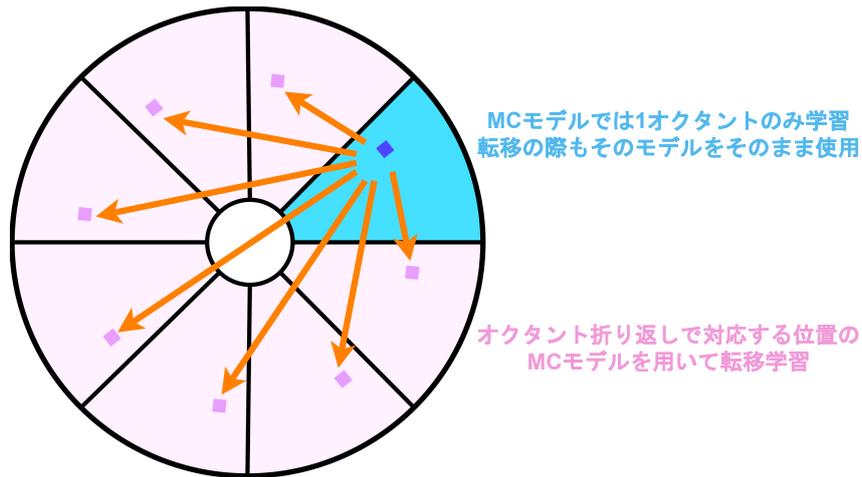


図 4.8: 学習した MC モデルをもとにして転移学習を行う際に各位置で用いる概念図

4.3.2 実データの学習結果

実データを用いた転移学習により得られたモデル出力例を図 4.9 に示す。左図は転移学習に用いた入力マップの例であり、こちらも MC データと同様に穴あきや孤立した島状の bin などが含まれる。これに対し、右図に示す学習後のモデル出力では、入力が疎である領域に対しても推定値が与えられ、領域内の大域的な傾向を保ちながら滑らかな p_T マップが得られている。特に、今回用いた実データでは特に 45 GeV 以上のミュオン統計が限られるが、転移学習ではシミュレーションで獲得した入力 $-p_T$ 対応関係を初期値として利用するため、実データのみでの学習では収束しにくい高 p_T 領域についても破綻を避けつつ推定を行うことができる。

転移学習モデルにおける予測値と正解値の対応関係を図 4.10 に示す。モデルの出力値がほぼ一直線上に集中しており、転移学習後のモデルが正確な予測ができていることが分かる。以後の Coincidence Window 作成では、ここで得られる対応関係に基づく較正を適用し、閾値判定に用いる p_T マップを構成する。

4.3.3 転移学習の性能評価

本小節では、実データに対するモデルの予測性能を、残差分布を用いて評価する。比較として、MC シミュレーションデータで学習したモデルへ実データを入力した場合と、実データで転移学習したモデルへ実データを入力した場合の残差分布を並べて示す。

図 4.11 に残差の二次元分布を示す。左図のように MC 学習モデルへ実データを入力した場合だと、残差分布は $r = 0$ から系統的に偏っており、さらに幅も大きい。これは、MC で獲得した対応関係が実データにはそのまま適用できず、MC と実データの条件差が予測のバイアスと分散として現れていることを示す。一方、右図に示す転移学習後のモデルでは、分布の中心が $r = 0$ 近傍へ移動し、全体として偏りが緩和される。したがって転移学習により、実データ固有の差分がモデル出力へ取り込まれていることが確認できる。

図 4.12 に、実データを MC 学習モデル、転移学習モデルそれぞれに入力した際の残差

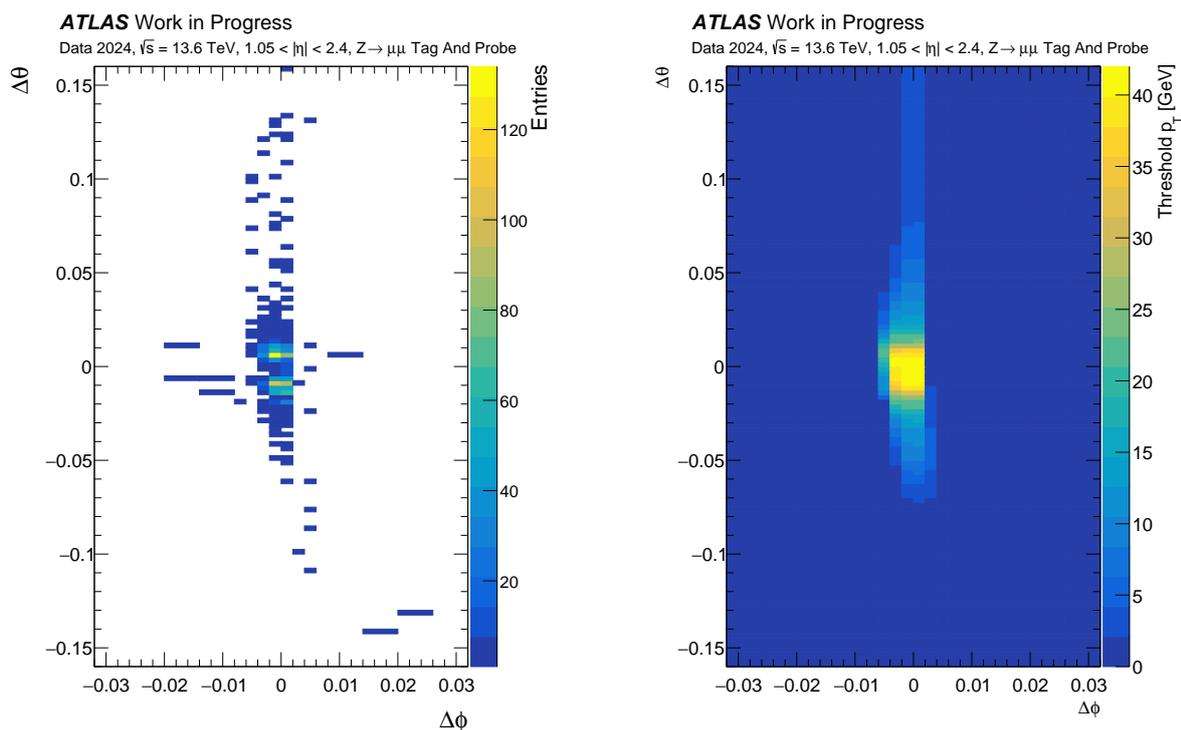


図 4.9: モデルの p_T 出力値マップの例。左図は転移学習の入力となったデータのヒットマップの例を、右図はある学習領域における転移学習モデル出力値の分布を示す

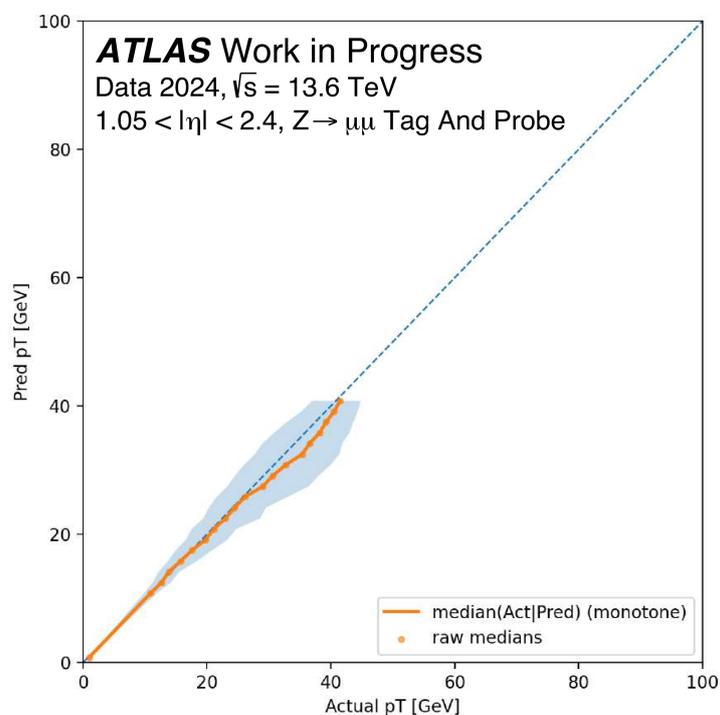


図 4.10: 転移学習モデル出力値と正解値の較正

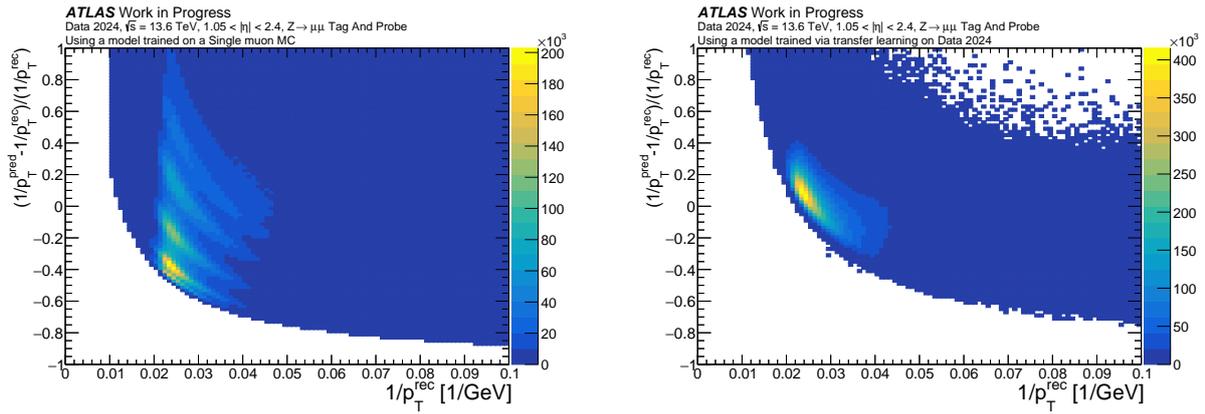


図 4.11: 実データに対する残差の二次元分布。左図は MC 学習モデルに実データを入力した際の二次元残差分布を、右図は転移学習後モデルに実データを入力した際の二次元残差分布を示す。転移学習により分布中心が $r = 0$ 付近へ移動し、残差の偏りが緩和される

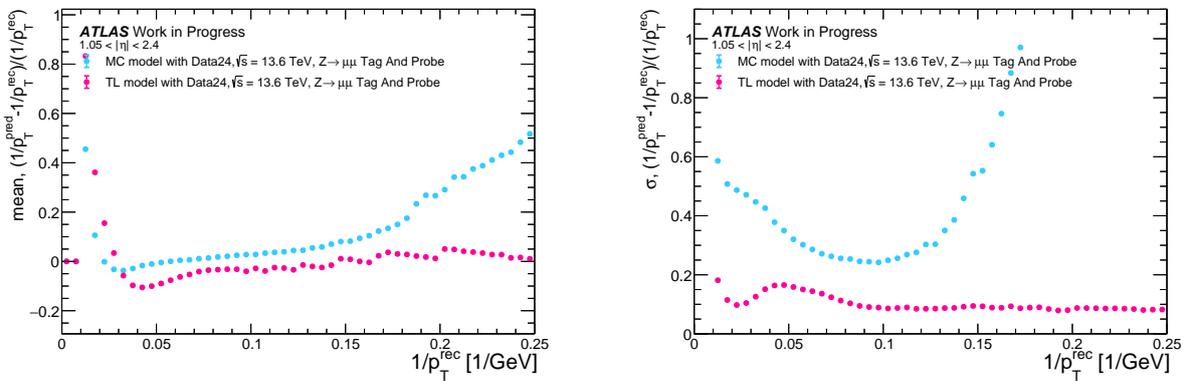


図 4.12: 実データを機械学習モデル・転移学習モデルに入れた際の残差における mean 値と σ の比較。左図は mean 値の比較を、右図は σ の比較を示す

分布の mean 値と σ の分布を示す。MC 学習モデルでは mean 値が $r = 0$ から系統的にずれる領域が見られるのに対し、転移学習後は mean 値が広い範囲で $r = 0$ 近傍へ移動し、MC 由来のバイアスが緩和されていることが分かる。さらに σ についても、転移学習モデルは MC 学習モデルに比べて全ての評価点で小さくなっており、残差分布の幅が抑えられている。 σ の具体的な値は表 4.2 に示す通りで、どの p_T においても実データのずれがモデルに取り込まれ、誤差が抑えられたことが分かる。

| | 10 GeV | 20 GeV | 30 GeV | 40 GeV | 50 GeV |
|---------|--------|--------|--------|--------|--------|
| MC モデル | 0.24 % | 0.33 % | 0.44 % | 0.48 % | 0.51 % |
| 転移学習モデル | 0.09 % | 0.16 % | 0.13 % | 0.09 % | 0.08 % |

表 4.2: 実データを機械学習モデル・転移学習モデルに入れた際の残差分布における p_T^{rec} に対する σ の比較。

4.4 電荷判定

本研究では、 p_T 推定に基づく Coincidence Window の生成に加えて、 $(UR, \text{Phi}, \Delta\theta, \Delta\phi)$ からミューオンの電荷符号 ($q = \pm 1$) を推定するモデルを別途学習し、各 (UR, ϕ) 領域ごとに $\Delta\theta$ - $\Delta\phi$ 平面上の電荷マップを作成した。電荷は磁場中の曲がり方向に対応するため、同一の入力特徴量からでも電荷情報を抽出できる可能性がある。以降では、学習データ作成、モデル学習、電荷マップ化、そして電荷 Window への変換の流れを述べる。

4.4.1 学習データの作成

電荷用の学習データは、 p_T 学習で用いた入力特徴量に電荷情報を付与して構成する。すなわち各イベントは

$$(UR, \text{Phi}, \Delta\theta, \Delta\phi, p_T, q)$$

を持ち、 q は電荷符号で正電荷を $+1$ 、負電荷を -1 とする。シミュレーションでは生成情報に基づく電荷符号を用い、実データではオフライン再構成ミューオンの電荷符号を教師信号として用いる。

なお、前節で述べた空 bin へのダミーイベント付加は、主にイベントがない領域の外挿抑制を目的とする。このうちランダムに付加したダミーイベントはその bin で事象が観測されていないことを表すため、電荷が物理的に定義できない。そこで電荷欄には $q = 0$ を付与し、電荷学習の教師信号からは除外する。一方で、後述の端部において誤った電荷の推定を避けるために付加する電荷ダミー点は $q = \pm 1$ を持つ点として扱い、学習に含める。

4.4.2 学習モデル

入力と教師信号

入力特徴量は 4 変数 $(UR, \text{Phi}, \Delta\theta, \Delta\phi)$ であり、正規化は p_T 回帰と同一の手順に揃える (離散 ID は $[0, 1]$ に線形スケールし、曲がり量は量子化後に $[-1, 1]$ へ正規化する)。電荷学習は二値分類として扱うため、教師信号 $q = \pm 1$ を

$$y = \frac{q + 1}{2} \quad (4.5)$$

により $y \in \{0, 1\}$ へ変換し、負電荷を 0、正電荷を 1 とする。前小節で述べた $q = 0$ は教師信号から除外し、学習時には損失計算の対象に含めない。

ネットワーク出力と損失

電荷推定モデルは全結合層からなる分類器とし、基本構造は p_T 回帰で用いた多層パーセプトロンを踏襲する。異なる点は出力層であり、最終出力をシグモイド関数で確率として

解釈できるようにする。

$$p = P(q = +1 \mid \text{inputs}) \in [0, 1]. \quad (4.6)$$

損失関数には交差エントロピーを用い、検証データに対する損失を監視して学習の進行を制御する。電荷の出現頻度に偏りがある場合には、クラス重み付け等により学習が片側へ引きずられないようにする。

判定に用いる電荷符号 \hat{q} は、確率 p を閾値 t で二値化して

$$\hat{q} = \begin{cases} +1 & (p \geq t) \\ -1 & (p < t) \end{cases} \quad (4.7)$$

と定義する。本研究では既定値として $t = 0.5$ を用いる。

学習の進行制御と学習方針

学習率の段階的減少やアーリーストップなど、学習の進行制御は p_T 回帰で用いた設定を同様に適用する。また電荷推定も p_T と同様に、(i) シミュレーションでベースモデルを学習し、(ii) 実データで転移学習を行う。実データ転移学習における初期モデルの割り当ては、 p_T の場合と同じ考え方でいい、オクタント対称性に基づく対応領域の MC モデルを初期値として読み込む。

4.4.3 電荷マップの作成

学習済みモデルを用い、各 (UR, ϕ) 領域について $\Delta\theta$ - $\Delta\phi$ 平面上の各 bin 中心に対して推論を行い、各 bin での正電荷確率 p を並べた二次元分布を電荷マップとして作成する。電荷マップはその bin で正電荷らしいかどうかを連続量で表現するため、後述する電荷 Window 化に先立つ中間生成物ともいえる。

端部における予測安定化のための電荷ダミー点

$\Delta\theta$ - $\Delta\phi$ 平面のトレンドを外れる領域は統計が薄く、本来とは異なる電荷が予測されることがある。この影響は、学習データがほとんど存在しない端部で確率 p が不定となり、電荷マップの端でまだらな構造が生じる形で現れうる。そこで本研究では、各領域の平面四隅に対して電荷符号を固定した少数の電荷ダミー点を付加し、端部での出力の発散や不定な揺らぎを抑制する。

具体的には、 $\Delta\theta$ が最も正側の端にある 2 つの角には $q = +1$ 、 $\Delta\theta$ が最も負側の端にある 2 つの角には $q = -1$ を割り当てる。これらはダミーイベントとは異なり教師信号 $q = \pm 1$ を持つ点として学習に含める。

4.4.4 電荷 Window の作成

電荷 Window は、電荷マップを離散化し、各 bin に電荷符号を割り当てたものである。閾値 t を用いて

$$C(i_{\Delta\theta}, j_{\Delta\phi}) = \begin{cases} +1 & (p \geq t) \\ -1 & (p < t) \end{cases}$$

と定義し、 (UR, ϕ) 領域ごとに二次元の符号マップとして保持する。この電荷 Window は、電荷判定効率の評価に用いるだけでなく、 p_T に基づく CW と組み合わせた、電荷を考慮した Coincidence Window の構成にも利用できる。

第5章

Coincidence Window の評価

本章では、学習したモデルの出力を示したうえで Coincidence Window の作成を行う。さらに、作成した Coincidence Window の性能を、実際のトリガー処理をシミュレートするソフトウェアを用いて評価する。評価には主として Turn-on curve を用い、先行手法と比較しながら、

- 立ち上がり後の効率
- 立ち上がりの鋭さ
- 実効的な閾値

を指標として議論する。

5.1 Coincidence Window の作成

学習した回帰モデルの出力を用いて、各 (UR, ϕ) 領域ごとに $\Delta\theta$ - $\Delta\phi$ 平面上の p_T マップを作成し、これを閾値化することで Coincidence Window を作成する手順について述べる。処理は大きく分けて、

- (i) モデルから p_T の値を出力
- (ii) 出力値を較正して p_T マップを作成
- (iii) p_T マップを指定した閾値ごとに判定し、Coincidence Window を作成

の3段階である。

モデル出力からの p_T マップ生成

まずは各 Unit に対して $\Delta\theta$ と $\Delta\phi$ の bin 中心を用意し、各 (UR, ϕ) について全 bin の p_T をモデルで求め、後述の補正を掛けて決定する。具体的には、入力

$$(UR, \text{Phi}, \Delta\theta, \Delta\phi)$$

を正規化した4変数としてモデルへ与え、全格子点の推定値 \hat{y} を一括推論する。

閾値の設定

今回のモデルは各 bin に対して連続値として p_T を出力するが、実機でのトリガー判定は 16 段階の閾値を以て行われる。そこで、得られた p_T マップを、あらかじめ定めた離散的な閾値へ写像し、各 bin に閾値を割り当てることで Coincidence Window を定義する。

具体的には、各 bin の連続推定値 p_T に対して p_T を超えない範囲で最も大きい閾値を割り当て、これを閾値マップ $T(i_{\Delta\theta}, j_{\Delta\phi})$ とする。このとき、ある閾値 t に対する Coincidence Window は、 t より大きな閾値を持つ bin の集合として与えられる。この表現を用いると、閾値 t を変えるだけで同一の閾値マップから任意の Coincidence Window を取り出せる。

5.1.1 MC データのための Coincidence Window

MC 学習モデルの p_T 出力マップを Coincidence Window の形式へ変換した。例として、ある (UR, ϕ) 領域に対して得られた Coincidence Window を [図 5.1](#) に示す。左図は MC 学習モデルの出力を用いた Coincidence Window であり、各 bin に割り当てられた離散閾値の 2 次元分布を表している。高い閾値ほど高 p_T を意味する領域、低い閾値ほど低 p_T でも通過しうる領域に対応する。この形式で Coincidence Window を構成することで、任意の閾値 t に対する Window は割り当て値が t 以上となる bin 集合として一意に導出できる。

5.1.2 実データのための Coincidence Window

実データへの転移学習により得られたモデル出力についても、MC の場合と同一の手順で Coincidence Window へ変換した。同じ (UR, ϕ) 領域に対して得られた Coincidence Window を [図 5.1](#) の右図に示す。転移学習モデルは、左図を作成した MC モデルで学習した一般的な対応関係を初期値として保持しつつ、実データ特有の位置依存のずれを吸収するよう調整されているため、MC ベースの Coincidence Window と比較した際に、特定領域での閾値分布のシフトとして差が現れうる。こうした差は、後節の評価において実効閾値や立ち上がり形状の違いとして現れる。

5.1.3 電荷学習の結果

電荷判定モデルについて、MC シミュレーションデータにおける入力マップと学習後の出力例を [図 5.2](#) に示す。ここでは、ある (UR, ϕ) 領域に対して $\Delta\theta$ - $\Delta\phi$ 平面上で電荷情報を推定し、各 bin における正電荷らしさを 2 次元マップ化した結果を可視化している。入力側は統計の疎密などが見られる一方、学習後の出力では領域内の連続性を保った電荷構造が得られる。

この MC モデルとともに、実データで転移学習を行った際の入力マップと学習後の出力例を [図 5.3](#) に示す。こちらも [図 5.2](#) と同様電荷構造が見られる。

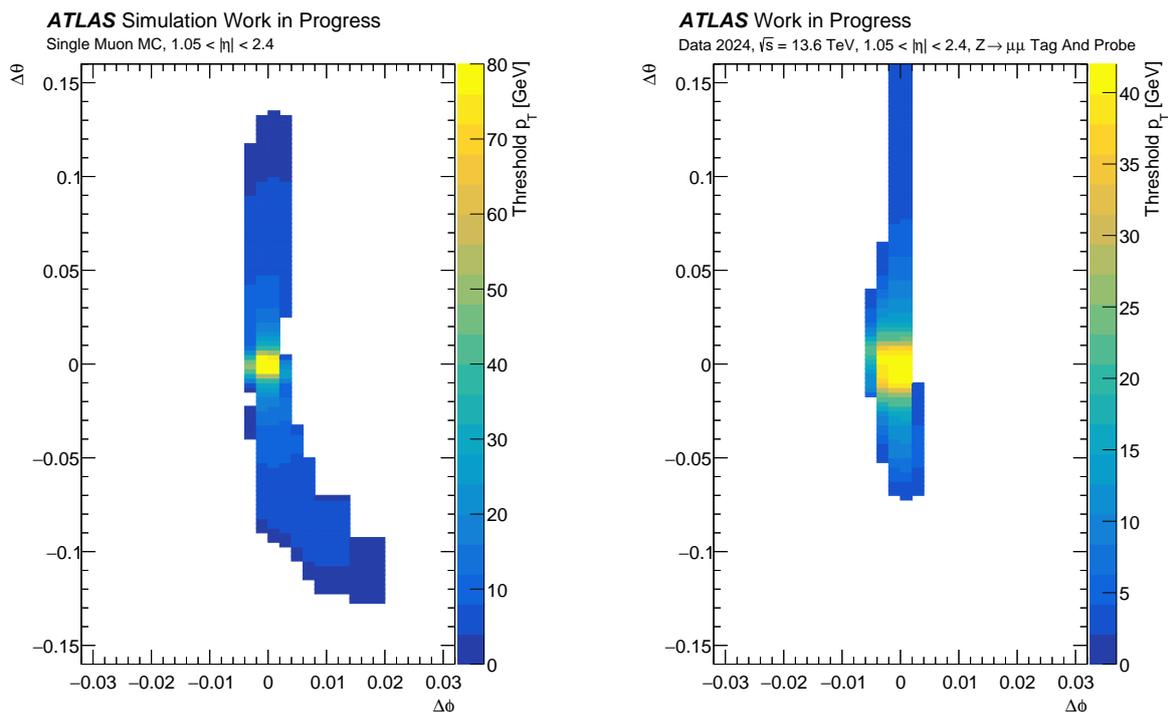


図 5.1: オクタント対称な領域における MC モデル/実データモデル Coincidence Window。左図は MC モデルの出力を用いた Coincidence Window の例を、右図は実データモデルの出力を用いた Coincidence Window の例を示す。右図は左図を作成したモデルをベースにして学習したモデルによるものである

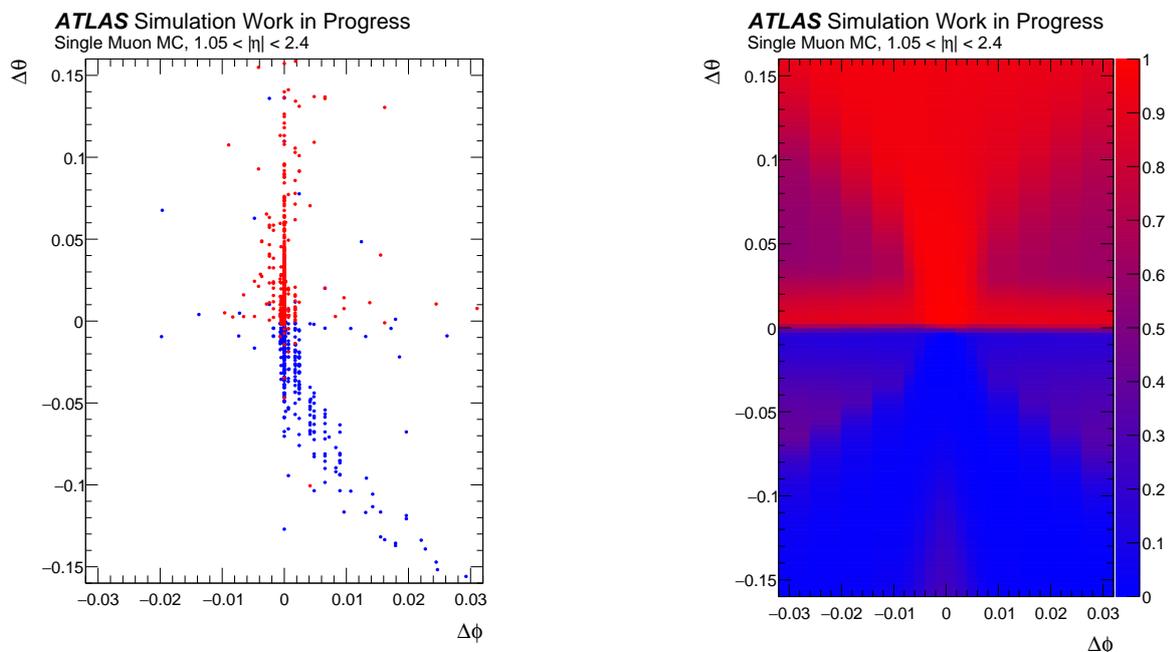


図 5.2: 電荷判定のための MC モデルの入力分布と出力値の例。左図は電荷学習の入力となったデータの例を、右図は電荷判定モデル出力値の分布の例を示す

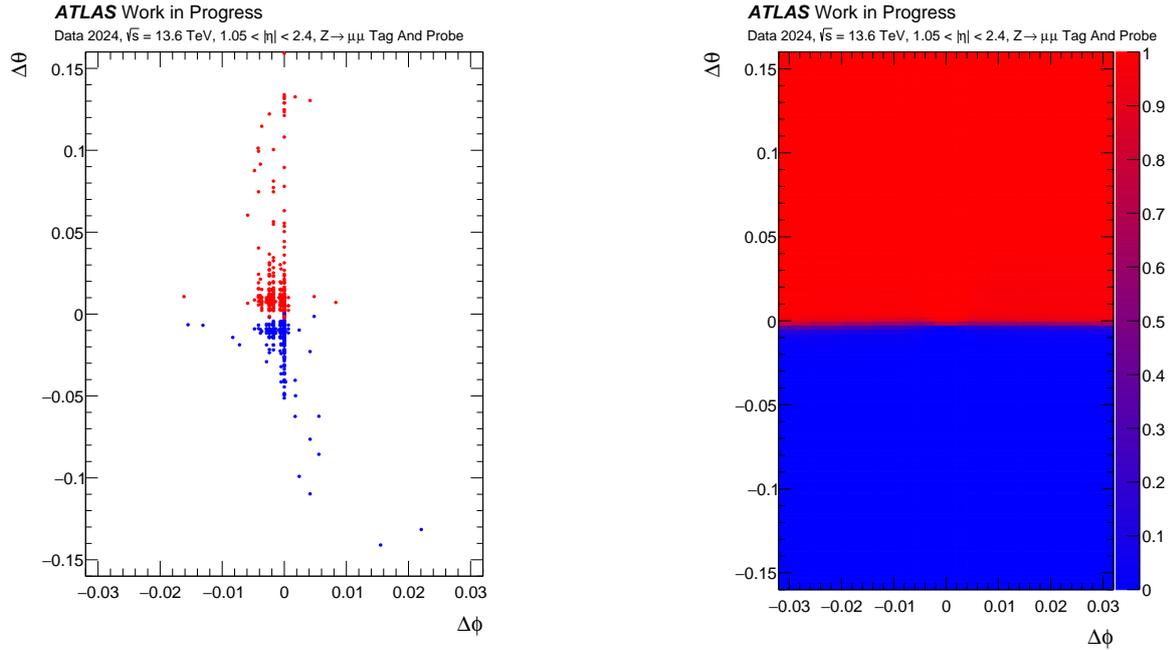


図 5.3: 電荷判定のための実データ転移学習モデルの入力分布と出力値の例。左図は電荷のための転移学習の入力となったデータの例を、右図は電荷のための転移学習モデル出力値の分布の例を示す

5.2 Coincidence Window の評価方法

Coincidence Window は、通過したミューオンの p_T が閾値以上かどうかを判定するものなので、理想的にはステップ関数の形をとるが実際は分解能などで段がなだらかに広がる。この節では、このなだらかに広がった関数やそのパラメーターの意味するところを説明し、それに基づいて Coincidence Window の性能評価方法について述べる。

5.2.1 Turn-on curve

Turn-on curve は、オフライン再構成 $p_T(p_T^{\text{offline}})$ に対するトリガー効率 ϵ を

$$\epsilon(p_T^{\text{offline}}) = \frac{\text{ある閾値以上のトリガーを発行したミューオン数}}{\text{全オフライン再構成ミューオン数}}$$

として定義し、 p_T^{offline} の関数として描いたものである。

理想的にはトリガーはある閾値で完全に ON/OFF が切り替わるためステップ関数に近い形状となるが、実際には検出器分解能や多重散乱、Segment Reconstruction からの入力値の離散化などにより立ち上がりが有限幅をもってなだらかになる。そこで Turn-on curve の形状を、ある単調増加関数でフィットし、立ち上がりの鋭さやプラトー効率、実効閾値をパラメータとして比較する。

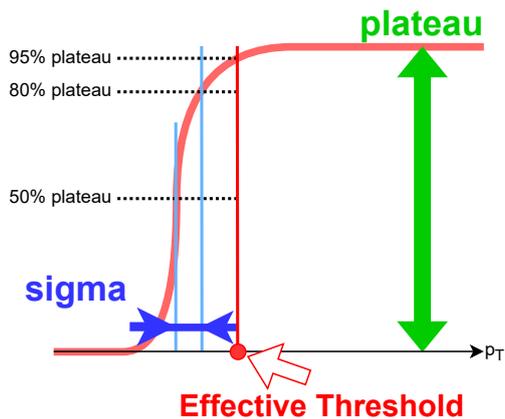


図 5.4: Turn-on curve から読み取る指標の模式図

Turn-on curve のパラメータ

中村氏の機械学習方式において用いていた Turn-on curve のフィット関数としてロジスティック関数

$$f(p_T) = \frac{a_0}{1 + \exp\left(-\frac{p_T - a_1}{a_2}\right)} \quad (5.1)$$

を示す。

ここで、 a_0 は立ち上がり後のプラトー効率、 a_1 は立ち上がり位置、 a_2 は立ち上がりの幅を表す。

本研究ではこれを参考にして、Turn-on curve から読み取る指標として以下を用いる (図 5.4)。

- **Plateau** : 高 p_T 側でのプラトー効率。本研究では 40 から 50 GeV の範囲における効率の平均値として評価する。
- σ : 立ち上がりの鋭さを表す幅の指標。plateau の 50% および 80% に到達する p_T をそれぞれ p_T^{50}, p_T^{80} とし、

$$\sigma \equiv p_T^{80} - p_T^{50} \quad (5.2)$$

と定義する。 σ が小さいほど立ち上がりが急峻であり、運動量分解能が良いことを意味する。

- **Effective threshold** : plateau の 95% に到達する p_T を実効閾値として定義する。

表 5.1: Effective threshold を揃えた条件下での先行手法と機械学習方式の比較

| | Effective threshold [GeV] | σ [GeV] | plateau [%] |
|------|---------------------------|----------------|-------------|
| 先行手法 | 12.7 | 2.2 | 94.8 |
| 機械学習 | 12.8 | 1.2 | 94.8 |
| 先行手法 | 20.3 | 3.3 | 94.4 |
| 機械学習 | 20.2 | 2.3 | 94.6 |

5.3 MC シミュレーションデータを用いた先行手法との性能比較

本節では、本手法と高輝度化に向けたアップグレードに向けて開発されていた先行手法について Turn-on curve を作成し、Effective threshold が近い組み合わせ同士で plateau、 σ を通して性能比較を行う。

5.3.1 p_T ごとの取得効率

先行手法 Coincidence Window と機械学習方式 Coincidence Window について、異なる Effective threshold 帯における Turn-on curve の比較例を [図 5.5](#) に示す。左図は Effective threshold が約 12 GeV 付近となるように揃えた比較であり、右図は同様に約 20 GeV 付近となるように揃えた比較である。

まず左図では、機械学習方式 Coincidence Window は立ち上がりの幅が先行手法に比べて小さく、Effective threshold 以下の取得効率がより急峻に抑制されている。これは、目標の閾値を維持しつつ低 p_T 側の背景をより強く抑えられることを意味し、トリガー選別の観点で有利となる。同時に、plateau 領域の効率が維持されていることは、高 p_T 事象の取りこぼしが増加していないことを示唆する。

右図においても同様の傾向が確認できる。機械学習方式 Coincidence Window では、Effective threshold 近傍の立ち上がりが先行手法よりも鋭く、 σ が小さい (すなわち分解能が高い) ことが分かる。これらの結果から、機械学習方式による Coincidence Window は、有効閾値を保ちながら Turn-on curve の幅を改善できることが確認された。

これらの比較を定量化するため、Effective threshold を揃えた代表点について比較した σ 、plateau を [表 5.1](#) にまとめる。

5.3.2 位置ごとの取得効率

前小節の結果が特定の位置に限った改善でないことを確認するため、領域依存性 (η 方向および ϕ 方向) に対する plateau 効率の比較を行った。[図 5.6](#) は各位置での plateau 効率を先行手法と機械学習方式で比較した結果である。plateau 効率には位置に応じた緩やかな凹凸が見られるが、これらの効率低下は MDT 検出器の位置較正のために空いている穴によってできた不感領域である。

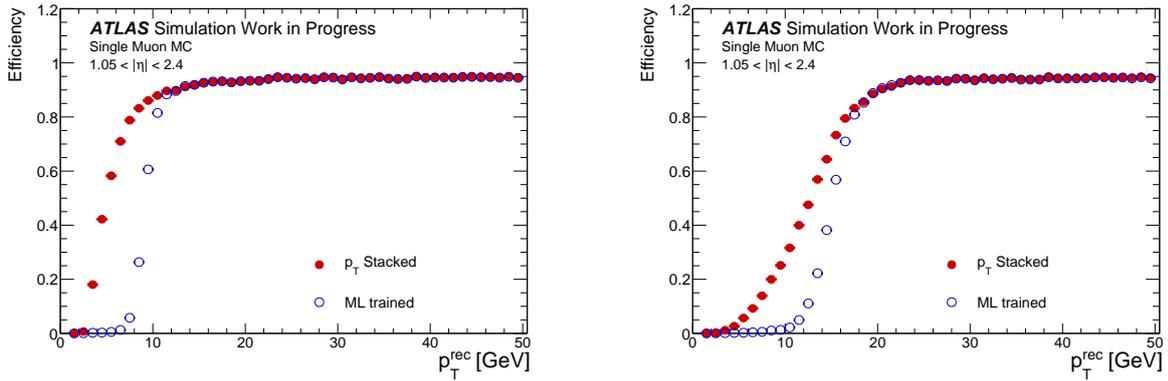


図 5.5: 異なる Effective threshold における Turn-on curve の比較。左図は Effective Threshold 12 GeV 付近のカーブの比較を、右図は Effective Threshold 20 GeV 付近のカーブの比較を示す。左図での機械学習によるカーブの Effective threshold は約 13 GeV であり、右図での機械学習によるカーブの Effective threshold は約 20 GeV である。 p_T Stacked はこれまで高輝度化アップグレードに向けた開発で用いられていた先行手法を、ML trained は機械学習を用いて作成した Coincidence Window を意味する

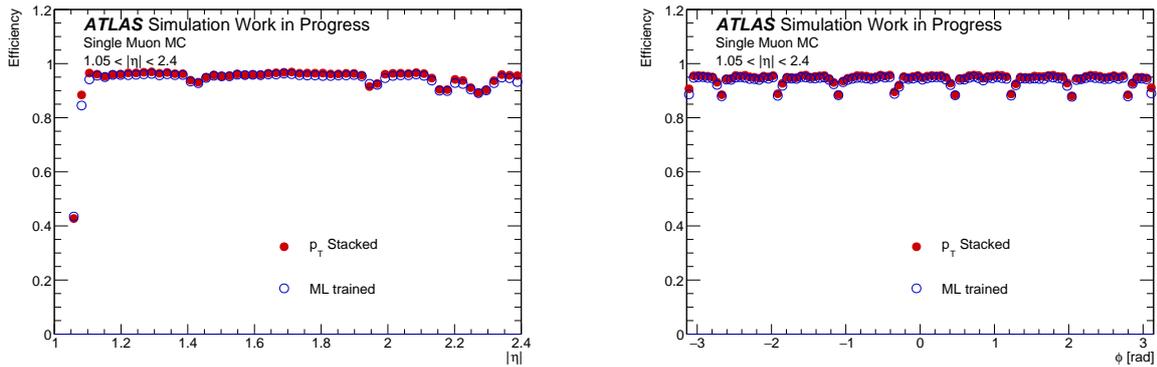


図 5.6: η および ϕ 各領域における plateau 効率の比較 (MC)。左図は η 方向における plateau 効率の比較を、右図は ϕ 方向における plateau 効率の比較を示す

左図より、 η 方向のいずれの領域においても plateau 効率は大きく低下しておらず、機械学習方式 Coincidence Window が局所的に効率を犠牲にして改善しているわけではないことが分かる。同様に右図から、 ϕ 方向についても plateau 効率が広い範囲で維持されている。以上より、機械学習方式 Coincidence Window は、位置依存の大きな効率劣化を伴うことなく、Turn-on curve の立ち上がり特性 (σ) を改善する傾向を示す。

5.3.3 p_T ごとの電荷判定効率

本節では、学習して得た電荷 Window を用い、Wire Strip coincidence まで到達したミュオン候補について、電荷符号が正しく判定できているかを評価する。具体的には、各事象の $(UR, \phi, \Delta\theta, \Delta\phi)$ から対応する電荷 Window の bin を参照し、そこで与えられる推定電荷 \hat{q} と教師信号の電荷 q が一致する割合を電荷判定効率として定義する。

電荷判定効率の p_T 依存性を図 5.7 に示す。一般に高 p_T になるほど曲がり量が小さく

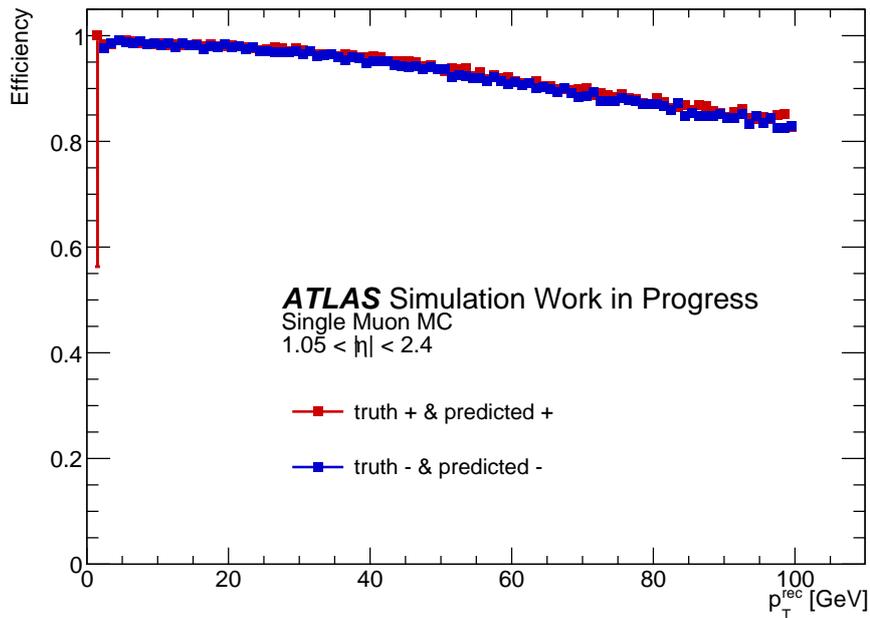


図 5.7: p_T に対する電荷判定効率。Wire Strip Coincidence まで到達した候補について、電荷 Window と教師信号の一致率を示す。

なり、 $(\Delta\theta, \Delta\phi)$ 空間で正負の分離が難しくなるため電荷判定効率は高 p_T 側で低下する傾向を示す。一方で低 p_T 領域では曲がり具合が大きく、正負の分離が比較的明瞭であるため、高い効率が得られている。

5.3.4 位置ごとの電荷判定効率

電荷判定効率が特定の位置に偏って劣化していないかを確認するため、 η 方向および ϕ 方向の位置依存性を評価した結果を図 5.8 に示す。

この図 5.8 より、効率が低下する領域は局所的に存在する。これらの領域は磁場が小さくなるために曲がり具合も小さく、その結果として正負の区別がつきにくくなっているためにできた不感領域と考えられる。

5.4 実データを用いた転移学習による Coincidence Window の評価

本節では、実データへの転移学習により得たモデル出力から構成した Coincidence Window の性能を評価する。

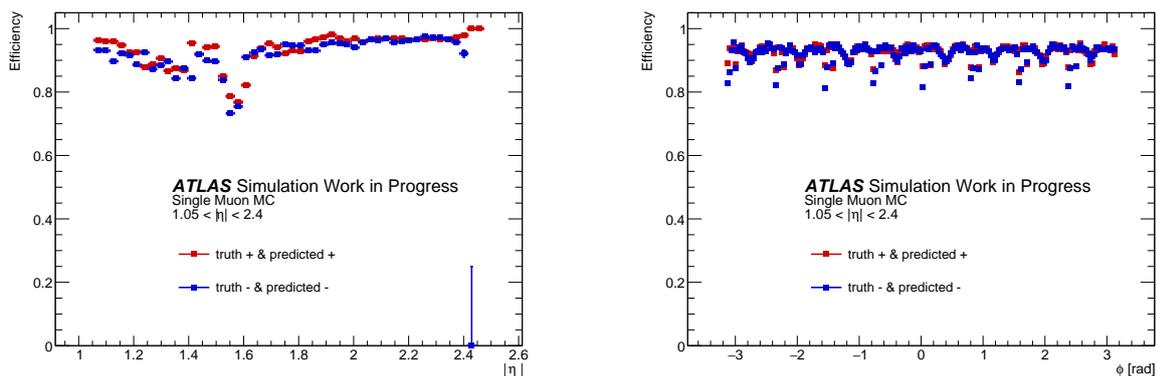


図 5.8: η および ϕ 方向の位置ごとの電荷判定効率。左図は η 方向における電荷判定効率を、右図は ϕ 方向における電荷判定効率を示す。

5.4.1 MC 学習モデルと実データ転移学習モデルの比較

本小節では、MC データで学習したモデルと、同モデルを初期値として実データで転移学習したモデルの出力を比較する。MC モデルはオクタント対称性を利用して学習領域を限定しているため、異なる位置であっても同一の初期モデルに由来する可能性がある。しかし、3章で示した通りチェンバーのずれの大きさは位置に依存する (図 3.15)。したがって、転移学習が有効に機能していれば、同じ初期モデルから出発していても、最終的な実データモデルの出力は位置ごとに異なるシフトを示すはずである。以下では、この出力分布のシフトを定量化することで、転移学習が位置依存の補正を取り込んでいるかを評価する。

MC モデルと実データモデルの出力分布比較方法

チェンバーのずれは、モデル出力マップの中でも高 p_T 側の領域がどこに現れるかに反映されやすい。そこで本研究では、各 (UR, ϕ) 領域について、MC モデルと実データモデルの出力マップを用意し、その中で高 p_T に対応する領域の中心位置を求め、両者の差として位置依存のシフトを評価した。

具体的な手順は次の通りである。まず、モデル出力マップは $\Delta\theta$ - $\Delta\phi$ 平面の各 bin に対して、その bin に割り当てられた p_T 閾値が与えられている。このマップに対して、40 GeV 以上の閾値が割り当てられている bin を高 p_T 領域として抽出する。すなわち、40 GeV 以上となった bin だけを塗り分けることで、高 p_T 側のまとまりを得る。

次に、この高 p_T 領域に含まれる bin の平均位置を、その領域の中心として定義する。ここで平均位置とは、高 p_T 領域に含まれる各 bin の中心座標 $(\Delta\theta, \Delta\phi)$ を単純平均したものであり、高 p_T 領域が全体としてどちら側に寄っているかを表す代表点として用いる。

最後に、MC モデルと実データモデルで求めた中心位置の差を、シフト量として比較する。シフトは $(\Delta\phi, \Delta\theta)$ 座標上の差として与えられるが、マップが離散 bin で構成されていることから、表では bin 番号での差も併せて示す。

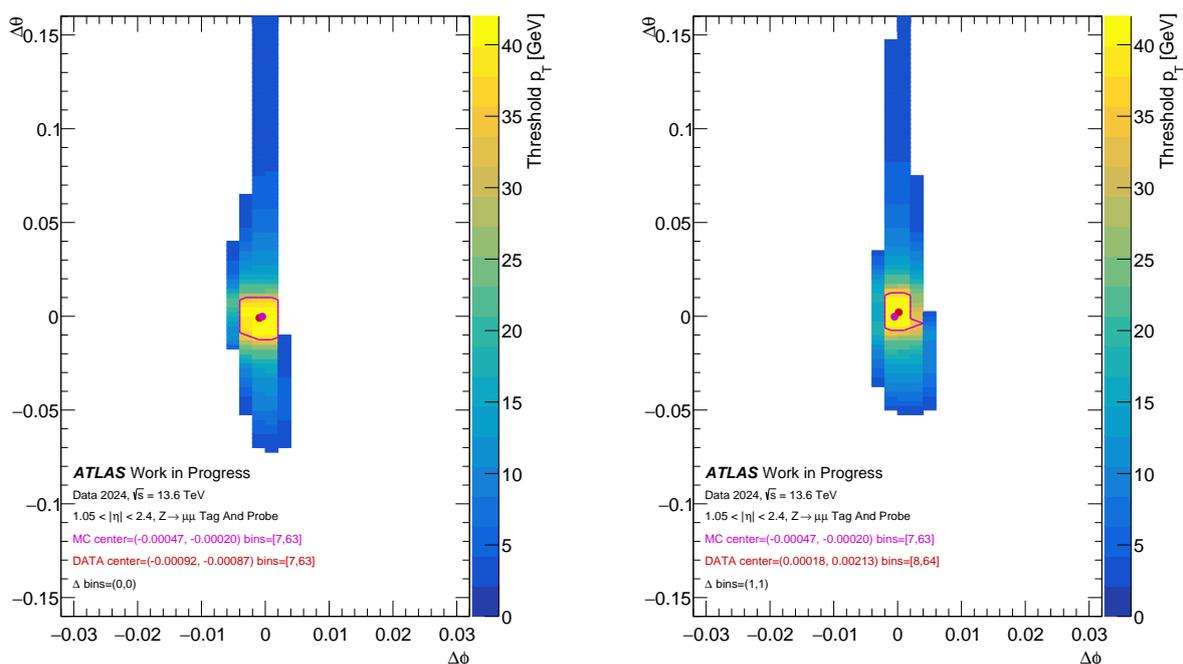


図 5.9: 同一の MC 初期モデルを基に転移学習した場合でも、位置に応じて実データモデル出力の中心位置が異なる例。左図はほぼずれがない領域における高 p_T 領域の中心位置を、右図はずれが大きい領域における高 p_T 領域の中心位置を示す

| 領域 | $\Delta\theta_{MC}$ | $\Delta\phi_{MC}$ | $\Delta\theta_{data}$ | $\Delta\phi_{data}$ | チャンネル差 |
|---------------------|---------------------|-------------------|-----------------------|---------------------|--------|
| (URID=31, PhiID=0) | -0.00047 | -0.00020 | -0.00092 | 0.00087 | (0,0) |
| (URID=31, PhiID=96) | -0.00047 | -0.00020 | 0.00020 | 0.00213 | (1,1) |

表 5.2: ずれ具合の異なる代表領域における、MC モデルを基準とした実データモデル出力の中心位置のシフト

位置依存の比較

結果の一例を図 5.9 に示す。左図は、図 3.15 でのずれが小さい領域の例であり、MC モデルと実データモデルの中心位置が概ね一致している。一方、右図は、ずれが大きい領域の例であり、実データモデルの中心位置が MC モデルに対してシフトしている。このことは、同一の初期モデルに由来する場合であっても、転移学習により位置ごとの補正が出力分布へ取り込まれていることを示唆する。

また、図 5.9 の例で求めた中心位置の値を表 5.2 にまとめる。

5.4.2 転移学習モデルから作成した Coincidence Window の評価

本小節では、実データへの転移学習により得たモデル出力から構成した Coincidence Window を、Turn-on curve を用いて評価する。前小節で示した通り、高 p_T 側の出力分布の中心位置は位置依存にシフトしており、転移学習により検出器条件の差分がモデル出力に反映されている兆候が見られる。一方、トリガー性能として重要なのは Turn-on curve がど

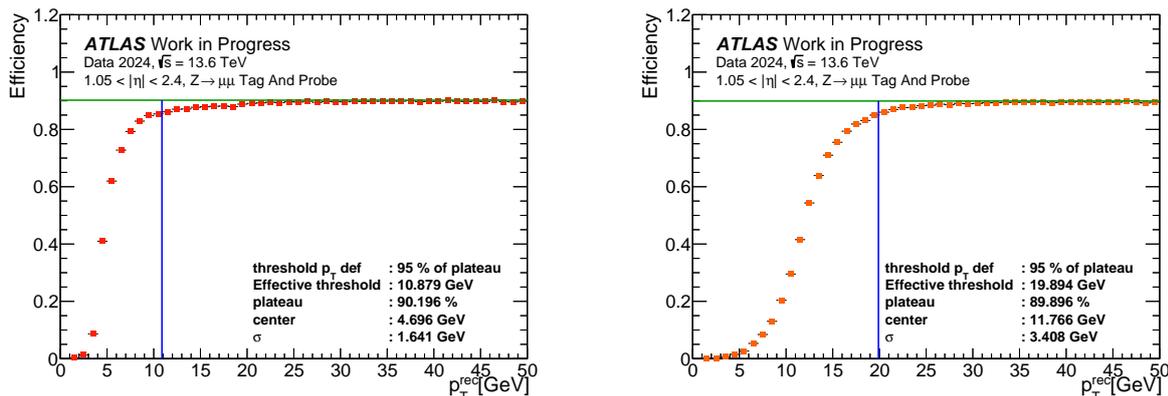


図 5.10: 異なる閾値に対する Coincidence Window の Turn-on curve。左図は Effective threshold が約 11 GeV における転移学習モデルから作成した Coincidence Window の Turn-on curve を、右図は Effective threshold が約 20 GeV における転移学習モデルから作成した Coincidence Window の Turn-on curve を示す

れだけ急峻に立ち上がるかであり、Turn-on curve の幅 σ はその指標となる。

図 5.10 に、転移学習モデルから作成した Coincidence Window の Turn-on curve を示す。ここでは Effective threshold が約 11 GeV と約 20 GeV の例を選んだ。いずれの閾値でも plateau はおよそ 90% 程度であり、 σ も数 GeV に収まっていることが確認できた。

5.4.3 位置ごとの取得効率

本小節では、転移学習 Coincidence Window の plateau 効率を η 方向および ϕ 方向に分解して評価し、効率低下が特定領域へ局在しているかを確認する。

図 5.11 に、 η 方向および ϕ 方向の plateau 効率を示す。これらは高 p_T 側の事象を用いて算出しており、Turn-on curve の σ とは独立に、plateau が位置によって大きく損なわれていないかを確認することを目的とする。結果として、平均的には高い効率が得られている一方で、方向によっては細かな構造や局所的な低下が見られる。

このような位置依存の構造は、検出器応答の非一様性に由来する可能性がある。特に EC では $|\eta| \sim 1.6$ 付近で磁場の影響が相対的に小さくなり曲がり弱くなる領域が存在するため、 p_T の分離が難しくなる領域が現れる。 ϕ 方向についてもこの磁場の影響がみられるために局所的に効率が低下する領域があるが、図 5.11 の右図を見ると特に ϕ がおよそ 1.4 から 2.0 の間に大きく効率低下をしている箇所を確認できる。この効率低下は Forward の C-side に広がっている。該当領域全体の実データの分布と、領域内の Coincidence Window の例を図 5.12 に示す。図 5.12 の左図を見るとこの領域全体でもわずか 254 イベントしか存在しないことが確認できる。右図は Forward 領域における、 ϕ 方向へはほぼ分布せず $\Delta\phi = 0$ を中心に $\Delta\theta$ 方向にのみ広がる典型的な形をしている。これらを照らし合わせると、効率低下の原因はこの領域において用いているデータの分布が Coincidence Window の範囲と大きく外れているという点であると考えられる。用いたデータの分布については、このデータが Run 3 のものである一方でシミュレータが Run 4 用のものでありトリ

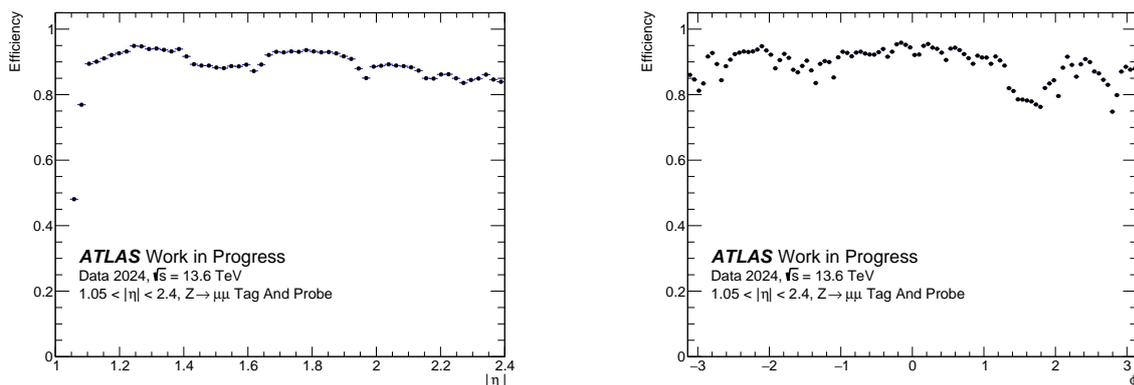


図 5.11: η および ϕ 各領域における実データを用いた場合の plateau 効率の比較。左図は η 方向における plateau 効率を、右図は ϕ 方向における plateau 効率を示す

ガーロジックの差などに起因している可能性があるが、この部分の調査は今後の課題となる。次小節では、転移学習で作成した Coincidence Window と MC データから機械学習方式で作成した Coincidence Window、さらに実データで機械学習した Coincidence Window の 3 種類について Turn-on curve を比較し、各手法の plateau および σ の違いを整理する。

5.4.4 3 種類の Coincidence Window の比較

本小節では転移学習で作成した Coincidence Window と MC データから機械学習方式で作成した Coincidence Window に加え、転移学習と同様の実データから機械学習方式で作成した Coincidence Window の 3 つを用いて比較を行う。

3 つの条件の Turn-on curve 比較

比較対象は以下の 3 条件の下で作成した Coincidence Window である。

- MC CW : MC データで機械学習したモデルから作成
- TL CW : MC モデルをベースに実データで転移学習したモデルから作成
- ML CW : 実データで機械学習したモデルから作成

これらは同一の実データおよび同一の評価手順で Turn-on curve を作成し、plateau と σ を比較する。

比較結果の Turn-on curve について、Effective threshold が約 11 GeV のものを [図 5.13](#) に、約 20 GeV に対するものを [図 5.14](#) にそれぞれ示す。対応する指標を [表 5.3](#) にまとめる。

Effective threshold 11 GeV では、TL CW の plateau は 90.2% であり、MC CW の 89.0% および ML CW の 88.9% より高い。 σ は MC CW および TL CW で 1.6 GeV と同程度であり、立ち上がりの広がりを悪化させることなく高い plateau が得られている。一方、ML CW では σ が 6.6 GeV と大きく、低 p_T 領域で立ち上がりが大きく広がる。このことから、実データのみで機械学習を行った場合は今回用いた統計量では低 p_T 領域の分解能が十分

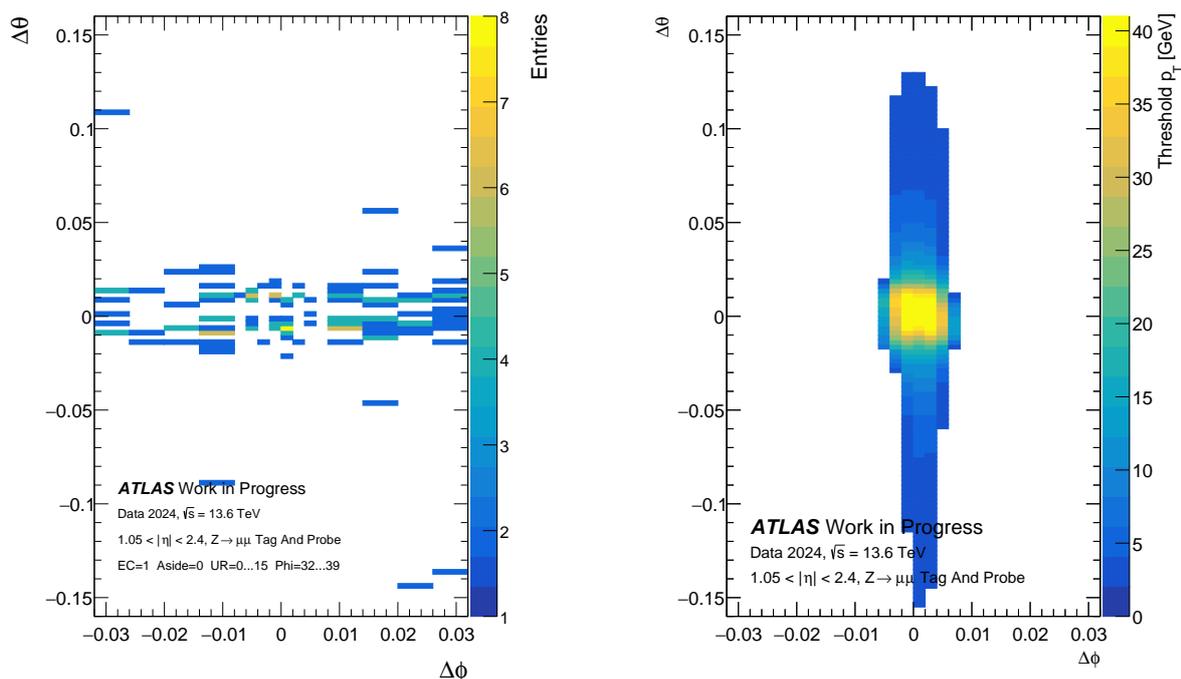


図 5.12: Forward 部 C-side の効率低下部におけるヒット分布と Coincidence Window の比較。左図は Forward 部 C-side の効率低下部全体のヒット分布を、右図は Forward 部 C-side の効率低下部の Coincidence Window の例を示す。入力データの分布が領域全体で少ないだけでなく、Forward 領域で典型的な分布からも大きく外れてしまっている部分が多い

でなく、MC モデルを初期値とする転移学習により低統計でも学習が収束し、性能を引き出せることが分かる。

Effective threshold 20 GeV でも、TL CW の plateau は 89.9% であり、MC CW の 88.4% および ML CW の 88.2% より高い。したがって、転移学習により幅広い領域で高い取得効率が維持されることが確認できる。一方で σ は、MC CW の 2.3 GeV に対して TL CW は 3.4 GeV へ増大している。ML CW の σ は 5.4 GeV とさらに大きく、転移学習は実データ機械学習のみの場合に比べれば立ち上がりの広がりを抑制できているが、分解能の観点では Effective threshold 20 GeV 付近でわずかな悪化が残る。これが plateau の上昇とトレードオフであるのか、それとも単に負の転移で分解能が下がっているだけなのかの調査は今後の展望の一つとなる。

以上より、転移学習により plateau を高く保った Coincidence Window が得られ、MC CW をそのまま適用した場合と比べても取得効率が改善することが分かる。また、実データのみの機械学習では低 p_T 領域で分解能が劣化しやすいのに対し転移学習では同じ統計量でも学習が収束し、良い性能が得られることも確認できた。一方で Effective threshold によっては σ が増大し、分解能の観点でわずかな悪化が生じているため、原因調査に加えて学習する層の範囲やモデル設計、 $Z \rightarrow \mu\mu$ 以外のデータの用意なども同時に行うことが必要である。

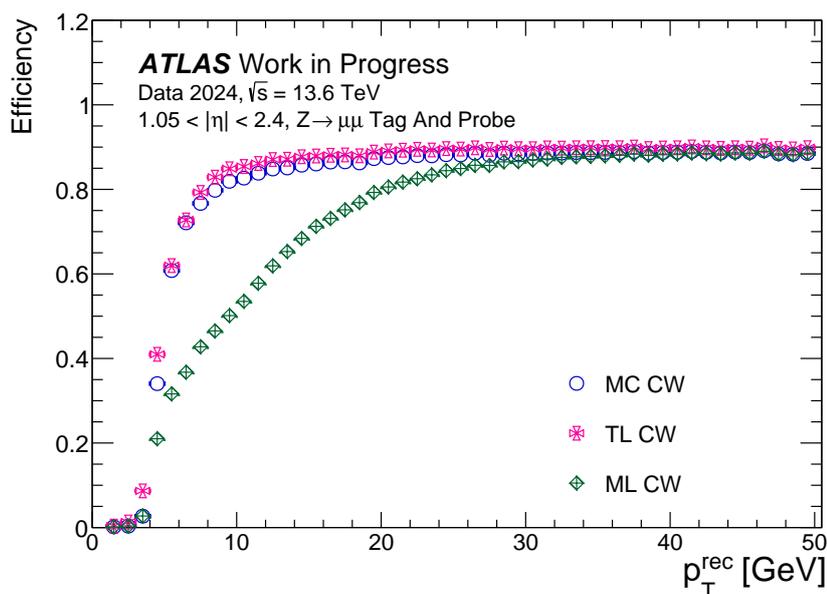


図 5.13: 実データで評価を行った 3 種類の Coincidence Window の Effective threshold 11 GeV における Turn-on curve 比較

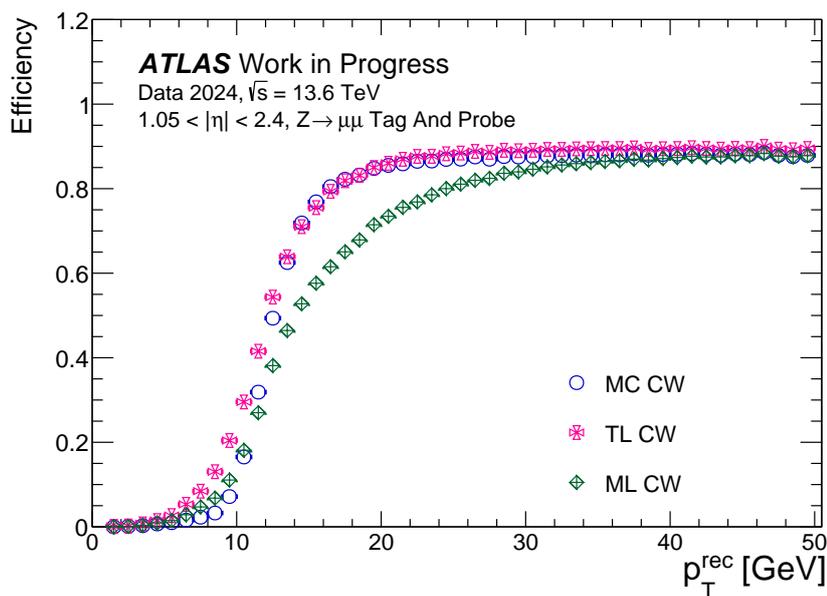


図 5.14: 実データで評価を行った 3 種類の Coincidence Window の Effective threshold 20 GeV における Turn-on curve 比較

| | plateau | | σ | |
|-------|---------|--------|----------|---------|
| | 11 GeV | 20 GeV | 11 GeV | 20 GeV |
| MC CW | 89.0 % | 88.4 % | 1.6 GeV | 2.3 GeV |
| TL CW | 90.2 % | 89.9 % | 1.6 GeV | 3.4 GeV |
| ML CW | 88.9 % | 88.2 % | 6.6 GeV | 5.4 GeV |

表 5.3: Effective threshold 11 GeV および 20 GeV に対する Turn-on curve の指標

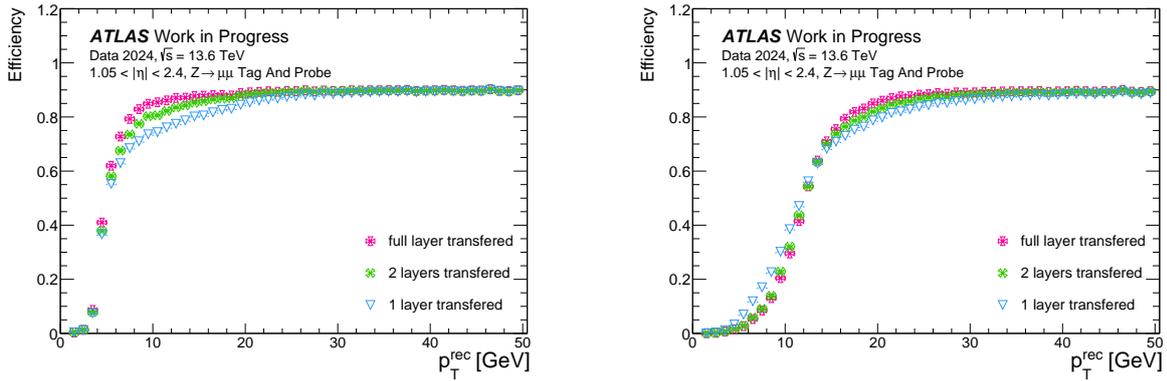


図 5.15: 学習する層を変更した際の Turn-on curve の比較

| | plateau | | σ | |
|------------|---------|--------|----------|---------|
| | 11 GeV | 20 GeV | 11 GeV | 20 GeV |
| full layer | 90.2 % | 89.9 % | 1.6 GeV | 3.4 GeV |
| 2 layers | 90.1 % | 89.8 % | 2.1 GeV | 3.7 GeV |
| 1 layer | 90.0 % | 89.1 % | 3.7 GeV | 5.2 GeV |

表 5.4: 学習する層を変えた際の Effective threshold 11 GeV および 20 GeV における Turn-on curve の指標

5.4.5 学習する層の比較

ここまでは転移学習の際にモデルの全層を再学習の対象としていたが、本小節ではこの全層再学習 (full layer transferred) の他に出力層から 2 層だけ再学習した場合 (2 layers transferred)、1 層だけ再学習した場合 (1 layer transferred) の転移学習モデルの Coincidence Window の性能比較を行う。再学習の対象とならない層については元のモデルの重みをそのまま用いることになる。

図 5.15 に Effective threshold 11 GeV、20 GeV の Turn-on curve をそれぞれ示す。各 Turn-on curve の指標は表 5.4 に示すとおりである。

いずれの p_T においても学習する層が多いほど立ち上がり幅が狭く、分解能が良くなることが分かる。また plateau に関してもわずかながら上昇することもわかる。

これらのことから、MC データと実データのずれやオクタントごとの差異に相当する情報は今回用いたモデルだと全層にわたって保存されているため、このように全層再学習によって最も性能が良くなるとわかる。

5.4.6 負の転移の対策

前小節までの比較より、転移学習により作成した Coincidence Window は、Effective threshold 11 GeV および 20 GeV のいずれにおいても高い plateau を維持できている。Effective threshold 11 GeV では、TL CW の plateau は 90.2% であり、MC CW の 89.0% および ML CW の 88.9% より高い。同時に σ は MC CW および TL CW で 1.6 GeV と同程度であり、

立ち上がりの鋭さを維持したまま取得効率が改善している。Effective threshold 20 GeV でも、TL CW の plateau は 89.9% であり、MC CW の 88.4% および ML CW の 88.2% より高い。したがって、転移学習により実データに対して高い取得効率が得られることが確認できる。

一方で Effective threshold 20 GeV では、TL CW の σ が 3.4 GeV となり、MC CW の 2.3 GeV より増大している。このことは閾値近傍の立ち上がりがわずかに広がっていることを意味し、分解能の観点では小さな負の転移が残っていると解釈できる。ただし同じ Effective threshold において ML CW の σ は 5.4 GeV であり、TL CW は実データのみの機械学習に比べて立ち上がりの広がりを抑制できている。今回用いた実データ量では、特に低い p_T 領域で実データの機械学習では学習が収束しにくく、転移学習では少ない統計でも収束することで性能を引き出せる点が重要である。

Effective threshold 20 GeV 付近で σ が増大する要因としては、実データ側の統計量や分布の偏りにより、閾値近傍の学習が難しくなることが考えられる。加えて、転移学習は plateau の改善に寄与する一方で、閾値近傍の境界を最適化するには学習設定やデータ条件に依存する余地が残る。このため今後は、plateau の改善を維持しつつ σ の増大を抑えるための最適化が必要である。

今後の改善案としては、実データの統計を補強するため $Z \rightarrow \mu\mu$ 以外の事象も用いて学習データを拡張することが挙げられる。またモデル構造について、層数を増やし各層のノード数を減らすことで特徴が層ごとに分離されやすい構造へ変更し、転移学習で更新する層の範囲を調整することも有効と考えられる。これらにより、転移学習の利得である高い plateau を保持したまま、閾値近傍の分解能をさらに改善できる可能性がある。

5.5 実データ転移学習による電荷 Window の評価

本節では、実データへの転移学習により得た電荷判定モデルから構成した電荷 Window について性能を評価する。電荷判定は二値分類であり、 p_T 推定とは異なる性質を持つため、転移学習の影響が同様に現れるとは限らない。ここでは、Wire Strip coincidence まで到達したミュオン候補に対して、電荷 Window が与える推定電荷 \hat{q} と教師信号の電荷 q の一致率を電荷判定効率として定義し、MC 学習モデルと実データ転移学習モデルの比較を通して、転移学習が電荷判定へ与える影響を整理する。

電荷判定効率の評価では、各事象の (UR, ϕ , $\Delta\theta$, $\Delta\phi$) から対応する電荷 Window の bin を参照し、そこで与えられる推定電荷 \hat{q} と教師信号の電荷 q が一致する割合を効率として算出する。以下では、先ほどと同様 p_T 依存性、 η 依存性、 ϕ 依存性の 3 つの観点から、MC CW と TL CW の差を比較する。

p_T 依存性

電荷判定効率の p_T 依存性を [図 5.16](#) に示す。一般に高 p_T になるほど曲がり量が小さくなり、($\Delta\theta$, $\Delta\phi$) 空間で正負の分離が難しくなるため、電荷判定効率は高 p_T 側で低下しやすい傾

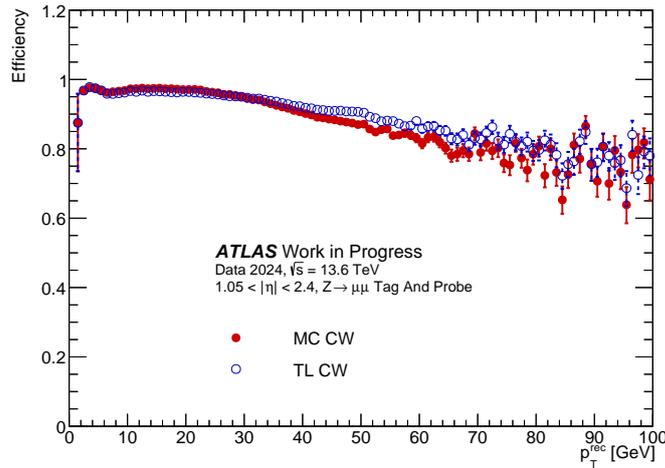


図 5.16: 電荷を分けずに各電荷 Coincidence Window をそのまま適用した場合の p_T ごとの電荷判定効率

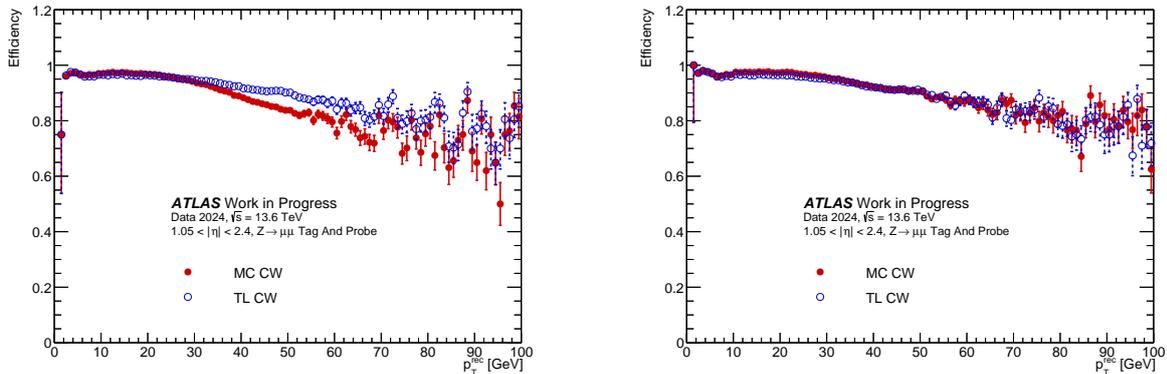


図 5.17: p_T における MC CW と TL CW に対する電荷判定効率の比較。左図は正電荷における p_T ごとの電荷判定効率を、右図は負電荷における p_T ごとの電荷判定効率を示す

向がいずれの Coincidence Window でも確認できる。転移学習後の Coincidence Window を用いた場合に、高い p_T において判定効率が上昇していることも確認できる。電荷ごとで比較した図を図 5.17 図をに示す。左図に示す正電荷については高い p_T における判定効率の向上が見られる。一方、右図の負電荷については判定効率は MC CW を用いた場合とほぼ同等であり、符号によって挙動が異なる。

このように、電荷判定でも高 p_T 側において TL CW が MC CW と同等または上回る領域が現れており、中心付近の特徴が転移学習で取り込まれているという結論と整合的である。

η 依存性

電荷判定効率の η 依存性を図 5.18 に示す。正電荷については広い領域で TL CW が MC CW を上回る傾向が見られ、位置依存の補正が有効に働いている可能性がある。負電荷については p_T の比較と同様大きな差は見られない。一方で、 $|\eta| = 1.4$ 付近の判定効率低下

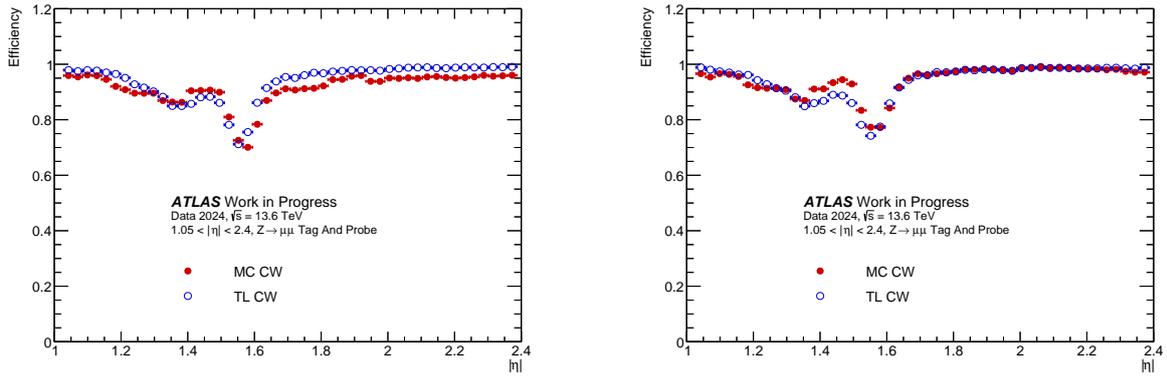


図 5.18: η 方向における MC CW と TL CW に対する電荷判定効率の比較。左図は正電荷における η ごとの電荷判定効率を、右図は負電荷における η ごとの電荷判定効率を示す

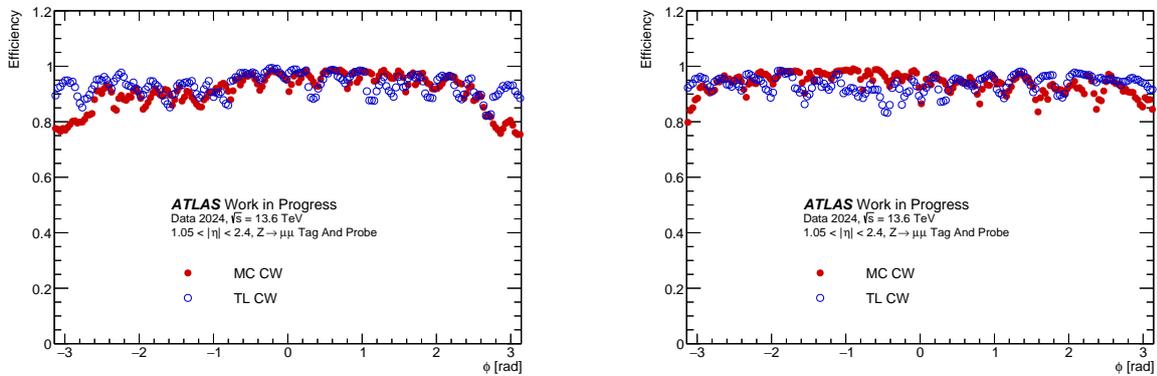


図 5.19: ϕ 方向における MC CW と TL CW に対する電荷判定効率の比較。左図は正電荷における ϕ ごとの電荷判定効率を、右図は負電荷における ϕ ごとの電荷判定効率を示す

部においては TL CW の方が効率が低下している部分も見られる。

ϕ 依存性

電荷判定効率の ϕ 依存性を図 5.19 に示す。正電荷、負電荷ともに、MC CW と TL CW はそれぞれ局所的な効率低下を起こしつつ概ね 90% 台を推移している。このことは、実データによる転移学習により ϕ 方向の大域的な偏りが緩和されている可能性を示唆するが、同時に、細かな構造が残っていることから局所的な非一様性や統計量の影響が依然として支配的であることも示している。

第6章

結論・展望

LHC-ATLAS 実験では、より大きな統計量を獲得して標準模型の精密検証および新物理探索の感度を高めるため、HL-LHC に向けた高輝度化と検出器・トリガー系の大規模アップグレードが計画されている。高輝度化に伴って同一バンチ衝突あたりのパイルアップが増大し、背景事象の増加によるフェイクトリガーの増加やイベントサイズの増大が課題となる。こうした高レート環境下でも物理感度を維持するため、初段トリガーにはより高い処理能力と判定性能が要求され、許容レートの増強やレイテンシ拡張を前提とした設計が進められている。その中で、エンドキャップ部の初段ミュオントリガーは TGC を用いてミュオン飛跡を高速に再構成し、ヒット位置の差に基づいて Coincidence Window を参照し、横運動量 p_T を短時間で判定する役割を担う。したがって、今後の運転条件に適した Coincidence Window を準備し、かつ運転条件・検出器条件の変化に対して柔軟に更新できる枠組みを整備することは、物理解析の感度確保に直結する重要課題である。

一方で、トリガー判断の粒度向上が進むほど、各領域で Coincidence Window を作成・最適化するために必要な統計量が不足しやすくなる。さらに、アップグレードに向けた Coincidence Window 開発では、シミュレーションデータを基盤とした設計がなされていたため、実データ特有のミスアライメント等に起因する差異を十分に取り込みにくいという問題があった。現行の Run3 においても機械学習を用いて Coincidence Window を自動生成する試みは既に行われているが、実データ側の統計量の制約により、領域によっては学習が収束せず性能が十分引き出せないといった問題がある。そこで本研究では、少ない実データ統計でも収束させたいうえで性能の良い Coincidence Window を構築するための方策として転移学習に着目し、MC で学習したモデルを土台として実データへ効率的に適応させることで統計量不足と MC-実データ差の双方に同時に対処する枠組みの構築を目指した。

本研究で得られた成果は次の三点にまとめられる。第一に、Coincidence Window を機械学習により作成するための一連の学習・評価システムを構築した。具体的には、作成した Coincidence Window の性能を Turn-on curve に基づいて定量化する評価フローを整備し、領域ごとの Coincidence Window 生成から性能評価までを一貫して実行できる枠組みを確立した。

第二に、MC シミュレーションデータを用いた学習により、先行手法と同等の plateau 効率を維持しつつ、しきい値近傍の立ち上がり特性が改善し得ることを示した。また、 η 方向および ϕ 方向の plateau 効率も広い範囲で維持されることが確認された。

第三に、MC 学習モデルを初期値として実データで転移学習を行う枠組みを実装し、実データに適応した Coincidence Window を作成できることを確認した。同一の MC 初期

モデルに由来する領域であっても、実データモデルの高 p_T 領域の中心位置は領域ごとに異なり、チェンバーの位置ずれを反映した補正が出力分布へ取り込まれている兆候が得られた。

さらに本研究では MC で学習したベースモデルを出発点として Run 3 実データで転移学習を行い、そのモデルから Coincidence Window を構成して MC 由来の Window および実データのみで学習した Window と比較した。転移学習は、MC で獲得した一般的な特徴を保持したまま実データ特有の差分を取り込むことにより限られた実データにおいてもトリガー効率の向上を達成することができた。

一方で、実データで転移学習をせずそのまま学習した場合には統計制約や分布差の影響を受けやすく、トリガー効率の低下や分解能の著しい低下も確認できた。このことは、転移学習はただ機械学習をする場合と比べても小統計下で高いトリガー性能が達成できるためにこの Coincidence Window 作成において非常に良い手法であることを示している。

また、転移学習で更新する層を変えた比較から、更新範囲を広く取るほど立ち上がりは鋭くなり、plateau もわずかに上昇する傾向が確認された。これは、MC と実データの差や領域ごとの差異に対応する情報がモデル全体に分散して保持されており、転移学習では十分な更新自由度を確保することが性能改善に寄与する可能性を示している。

電荷判定についても、MC 学習モデルと実データ転移学習モデルから電荷マップを作成し、電荷判定効率の p_T 依存性および位置依存性を評価した。その結果、電荷判定効率は高 p_T 側で低下する傾向を示し、 η および ϕ に依存した差も確認された。電荷判定は二値分類であるため学習自体は収束しやすい一方、局所的な被覆率の変化や非一様性など、位置依存の差を生む要因の切り分けと最適化は今後の課題である。

今後の展望として重要なのは、一部の領域で見られた σ の増大を抑えつつ、転移学習による高い plateau を維持する学習設計の確立である。学習データについては、 $Z \rightarrow \mu\mu$ に加えて他の事象も用いて統計を補強し、しきい値近傍の教師信号を増やすことが有効と考えられる。また、モデル構造について層数を増やし各層のノード数を減らすことで特徴が層ごとに分離されやすい構造とし、転移学習で更新する層の範囲を調整することも改善の余地がある。さらにトリガー実装を見据えると、運用で用いる名目閾値の選択や量子化設定を含め、学習から Coincidence Window 生成までの条件を整理し、再現性を持って更新できる形へ整備しておくことが重要である。

以上のように、本研究は機械学習および転移学習に基づく Coincidence Window 作成の枠組みを整備し、MC 上での有効性に加えて、実データ上で転移学習により plateau の改善と低 p_T 領域の分解能の向上が見られることを示した。本研究で構築した枠組みは、Coincidence Window の粒度向上や MC と実データの差といった条件下で顕在化しやすい課題に対して、実データに適応した Coincidence Window を継続的に更新するための基盤となり得る。今後、上記の最適化を進めることで、HL LHC の高レート環境下においても高効率な初段ミューオントリガーを実現するための手段として、本手法の実運用への適用が期待される。

謝辞

本研究を進めるにあたり多くの方に支えていただきました。心より感謝いたします。

担当教員の前田順平准教授には普段の研究から学会などの資料添削まで、幅広いご助言、ご指導を賜りました。Coincidence Window 開発における指針の提示など、本研究で重要なアドバイスも多くいただきました。特に、資料作りが得意ではないためにかんがりの手間をおかけしています。発表スライドのコツなども修論発表会に向けてかなり勉強になりました、ありがとうございます。いつも丁寧な指導に大変感謝しております。

副査を受けてくださった藏重久弥教授、西野友年准教授にもこの場を借りて御礼を申し上げます。

また、藏重久弥教授につきましては常日頃より初歩的な質問でも快く、そしてとてもわかりやすく優しい解説をしてくださったおかげで、(まだまだ道のりは長いですが)研究室に入った頃とは比べ物にならないくらい物理観が広がりました。趣味の電子工作の質問なども受けてくださるなど、知的好奇心をそのまま伸ばすスタイルが僕は好きでした。深く御礼申し上げます。山崎祐司教授の鋭い洞察により問題が解決することも多く非常に助けられています。地理など多角的な知識に裏打ちされたトークも楽しませていただいております。各種手続きにおいても大変お世話になっております。

神戸大学粒子物理学研究室のダークマターグループ身内賢太郎教授、東野聡特命助教にも学振において多大なるご助力をしていただきました。勢いで書き始めて迷走する私に道筋を示し、詳しい説明などもしていただけましたことに今一度感謝いたします。ニュートリノグループの竹内康雄教授、鈴木州講師、伊藤博士講師にも研究室生活の中でお世話になっております。研究室秘書の北川順子様も、円滑な出張などを支えてくださっていることに感謝しております。

研究室の先輩の高橋真斗氏、鐘海文氏、水引龍吾氏、生井凌太氏、鈴木大夢氏、張力氏は頼れる先輩として研究についてのみならず多くのことを学ばせていただいております。ありがとうございます。

後輩の青山真也君、稲葉有哉君、門田佳吾君、神吉陽介君、遠山和佳子さん、西上真央さん、西田汐里さん、野村脩貴君、番原大登君、橋本真理子さん、ズー君、そして学部4回生の皆さん。奥にこもっている先輩でしたが、それでもちょくちょく構ってくれてありがとうございます。来年以降も仲良くしてください。

同じ研究グループである日本 ATLAS グループの皆様にも大変お世話になっております。奥村恭幸准教授には、何かテーマが欲しいという曖昧な相談であったにもかかわらず、嫌な顔一つせず方々を巡り、Coincidence Window 作成のテーマをご用意いただきました。心より御礼申し上げます。齋藤智之助教には、毎週のミーティングで助言を賜り、研究において非常によいヒントとなっております。感謝申し上げます。堀井泰之准教授にも、普段のミーティングなどでアドバイスをいただきましたこと、厚く御礼申し上げます。泉山将大特任助教には、ミーティングのみならず対面での会合後の席などでも広くお

世話になりました。御礼申し上げます。青木雅人研究機関講師には、CERNにおいて地下作業のタイミングを含め気にかけていただいたおかげで、充実したCERN生活を送ることができました。深く感謝いたします。山口洋平助教には、KEK出張をはじめとしてお世話になることが多く、何かと助けていただきましたことに御礼申し上げます。須江祐貴博士研究員には、毎週のミーティングでの確かなコメントや助言をしていただき、研究に大きな活力を与えていただきました。心より感謝申し上げます。

中川徹郎氏と河本地弘氏は、このテーマを初めてすぐのころから根気強く教えてくださっただけでなく、学会などのタイミングでも気にかけてくださっており、とても励みになっております。ありがとうございます。近藤翔太氏と須部美咲氏はM1時点で参加した統合試験でとてもお世話になりました。SDカードを落としてしまったりとご迷惑もおかけしましたが、それでも良く接して下さってありがとうございます。藏嘉琦氏、林雄一郎氏、古川真林氏、山下恵理香氏、長坂錬氏、成川佳史氏、牧田藍瑠氏、大隅悠矢氏、橋本大輔氏、和田有咲氏はCERN出張で右も左もわからない私に優しくしてくださったり、会合の際にアドバイスを賜ったり、時には遊びに連れ出してくださったりしました。ありがとうございます。田上理沙子氏、水落永遠氏、大坪航氏、牧和真氏、糺翔太氏、千葉公太郎氏、間仁田省吾氏は他大学ながらも同期として、良い刺激をもらうことができました。感謝しています。

研究室同期の浅見優輝君、佐倉慶汰君、笹田真宏君、佐野友麻君、鈴木啓司君、曾根貴将君、田中亮祐君、谷口紘大君、和田伸一郎君。いろんなイベントや出来事があり、何の気なしにみんなでだべったりコンビニに行ったりしていたことがこれからはもうなくなると思うととてもさみしく思います。修論発表会に向けてスライドのチェックや発表練習にもよく付き合ってくれてありがとう、おかげさまで無事切り抜けることができました。加えて泉啓太君、井上真也君、清水涉夢君、西田光希君。大学入学から今まで、みんなで遊びに行ったり課題やテスト勉強をしたり、とりとめのないと思っていた日々もこうして思い返すとかけがえのない思い出になっていました。二度と手の届かない時間をいつくしむことができるのはみんなのおかげです。思い出に満ちた、決して短くはない6年でした。ありがとう。

これからも定期的に集まりたいです。年一か、できれば年二で。何ならまた旅行とか行きたいです。たぶんこれまでほど人が集まらないとは思いますが……

最後に、ここまで育ててくれた家族に今一度感謝の念を述べておきます。いろいろ大変なことも多い中で、博士課程への進学を応援してくれてありがとう。もうしばらく頑張ってみることにします。これからもどうぞよろしくお願いします。

参考文献

- [1] The Standard Model: a beautiful but flawed theory
URL: <https://www.quantumdiaries.org/2014/03/14/the-standard-model-a-beautiful-but-flawed-theory/>.
- [2] ATLAS Collaboration, Technical Design Report for the Phase-II Upgrade of the ATLAS TDAQ System. CERN-LHCC-2017-020.
- [3] 中川 徹郎, 高輝度 LHC-ATLAS 実験に向けた初段ミューオントリガーアルゴリズムの最適化 修士論文, 京都大学, 2025.
- [4] 中村 竜也, LHC-ATLAS 実験 Run-3 における初段ミューオントリガーの機械学習を用いた最適化 修士論文, 神戸大学, 2023.
- [5] 松井孝太, 熊谷亘, **機械学習プロフェッショナルシリーズ 転移学習**. 講談社, 3 ed., 2024.
- [6] ATLAS Collaboration, The ATLAS Experiment at the CERN Large Hadron Collider [JINST 3 \(2008\) S08003](#).
- [7] CMS Collaboration, The CMS experiment at the CERN LHC [JINST 3 \(2008\) S08004](#).
- [8] ALICE Collaboration, The ALICE experiment at the CERN LHC [JINST 3 \(2008\) S08002](#).
- [9] LHCb Collaboration, The LHCb Detector at the LHC [JINST 3 \(2008\) S08005](#).
- [10] E. Lopienska et al., The CERN accelerator complex, layout in 2022 CERN-GRAPHICS-2022-001.
- [11] ATLAS Collaboration, ATLAS Feature: A new ATLAS for the high-luminosity era 17 Jan 2023, URL: <https://cds.cern.ch/images/ATLAS-PHOTO-2023-002-4>. ATLAS-PHOTO-2023-002-4.
- [12] J. Pequeno, Computer generated image of the ATLAS inner detector 27 Mar 2008, URL: <https://cds.cern.ch/record/1095926>. CERN-GE-0803014.
- [13] ATLAS Collaboration, ATLAS MAGNETIC FIELD
URL: <https://atlas.web.cern.ch/Atlas/GROUPS/MUON/magfield/>.
- [14] J. Pequeno, Computer Generated image of the ATLAS calorimeter 27 Mar 2008, URL: <https://cds.cern.ch/record/1095927>. CERN-GE-0803014.

- [15] J. Pequeno, Computer generated image of the ATLAS Muons subsystem 2008, URL: <https://cds.cern.ch/record/1095929>. CERN-GE-08030174.
- [16] C. Marcelloni et al., Installation of the first of the big wheels of the ATLAS muon spectrometer, a thin gap chamber (TGC) wheel CERN-EX-0609016, 2006.
- [17] ATLAS Collaboration, *ATLAS level-1 trigger*. Technical design report. ATLAS. CERN, Geneva, 1998. URL: <https://cds.cern.ch/record/381429>. CERN-LHCC-98-014.
- [18] ATLAS Collaboration, Small-Strip Thin Gap Chambers for the Muon Spectrometer Upgrade of the ATLAS Experiment [JINST 15 \(2020\) C09064](#), ATL-MUON-PROC-2020-008.
- [19] ATLAS Collaboration, The ATLAS Trigger System for LHC Run 3 and Trigger performance in 2022, CERN-EP-2023-299, 2024.
- [20] ATLAS Collaboration, Technical Design Report for the Phase-II Upgrade of the ATLAS Trigger and Data Acquisition System - Event Filter Tracking Amendment, CERN-LHCC-2022-004, ATLAS-TDR-029-ADD-1.
- [21] ATLAS Collaboration, Technical Design Report for the Phase-I Upgrade of the ATLAS TDAQ System. CERN-LHCC-2013-018.
- [22] 水引 龍吾, 高輝度 LHC-ATLAS 実験に向けた初段エンドキャップミューオントリガーのファームウェア開発と検証 修士論文, 神戸大学, 2025.
- [23] ATLAS Collaboration, Technical Design Report for the Phase-II Upgrade of the ATLAS Muon Spectrometer, CERN-LHCC-2017-017, ATLAS-TDR-026.
- [24] 長坂 錬, 高輝度 LHC-ATLAS 実験に向けた L0 ミューオントリガー大規模エレクトロニクスの統合試験とコミッショニングシステムの開発-クロック・制御・読み出し系の開発の統合 修士論文, 東京大学, 2023.
- [25] 河本 地弘, 高輝度 LHC ATLAS 実験に向けた初段ミューオントリガーアルゴリズムの実装と検出器全体への拡張 修士論文, 京都大学, 2023.
- [26] 三野 裕哉, 高輝度 LHC ATLAS 実験に向けた初段ミューオントリガーアルゴリズムの開発およびハードウェアへの実装 修士論文, 京都大学, 2019.
- [27] 赤塚 駿一, LHC-ATLAS 実験 Run-3 に向けたミューオントリガーの改良 修士論文, 京都大学, 2017.
- [28] ATLAS TDAQ PhaseII Upgrade Project. Endcap Sector Logic. Technical report 2021.

- [29] M. Kikuchi, DQ and Performance in 2023, Special ATLAS-J L0/L1 muon meeting, Shinshu University, Matsumoto, Japan Dec 2022,
URL: <https://indico.cern.ch/event/1214296/>. Accessed: 2025-03-07.
- [30] D. Lellouch, L. Levinson, K. Hasuko, Naming and numbering scheme for the Endcap muon trigger system, ATL-MUON-2001-002.
- [31] TensorFlow <https://www.tensorflow.org/>.
- [32] Keras <https://keras.io/>. Accessed: 2025-12-28.
- [33] A. Maria, R. Vera, J. Pequena, ATLAS Detector Magnet System 28-05-2021,
URL: <https://cds.cern.ch/record/2770604>. ATLAS-PHOTO-2021-029.
- [34] M. Abadi, A. Agarwal, P. Barham, and others., TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems CoRR **abs/1603.04467** (2016).
- [35] F. Chollet, et al., Keras <https://keras.io>, 2015. Accessed: 2025-12-28.