

2023年度 修士論文

# Belle II 実験レベル 1 トリガーのための 3次元飛跡再構成アルゴリズムの開発

(Development of the Three-Dimensional Track Reconstruction  
Algorithm for the Belle II Level-1 Trigger)

東京大学大学院 理学系研究科 物理学専攻

須藤裕人

2024年1月26日提出

指導教員: 後田裕

## 概要

Belle II 実験は SuperKEKB 加速器での電子陽電子衝突で生成された  $B$  中間子や  $D$  中間子、 $\tau$  レプトンの崩壊を精密に測定し、標準理論を超える物理の探索を行う実験である。本実験はデータ収集の前にリアルタイムで背景事象を排除して読み出しレートを許容範囲内に抑えるために、Level-1 トリガーと呼ばれる電子回路のトリガーシステムを用いている。しかし、近年ルミノシティが高くなるにつれビーム由来の背景事象が増えてきたことで、目標ルミノシティにおいてトリガーレートが上限値の 30 kHz を超えてしまうことが予想される。問題となっている背景事象の多くは衝突点から離れた場所から来るため、中央飛跡検出器と呼ばれるドリフトチェンバーの情報を用いて飛跡を 3 次元的に再構成することで、発生点の場所をもとに排除することができる。

本研究では、飛跡を 3 次元再構成するための電子回路ボードを Belle II トリガーグループによって開発された UT4 とよばれるボードにアップグレードし、回路に実装するアルゴリズムとして Hough selection アルゴリズムを新たに開発した。ボードのアップグレードに伴いより多くのヒットを再構成に使用できるようになったことでより精度良く再構成できるようになった。また、Hough selection アルゴリズムでは Hough voting とよばれるパターン検出手法をヒット選択に用いることで、旧アルゴリズムが持っていたノイズによるヒットを選択してしまう問題を解決することができた。シミュレーションの結果、新アルゴリズムでは旧アルゴリズムと同じ信号飛跡の再構成効率を保ちつつ 144% 多くの背景事象による飛跡を排除することができ、トリガーレートを上限以下に抑えられることが示された。

# 目次

|       |                         |    |
|-------|-------------------------|----|
| 第 1 章 | 序論                      | 1  |
| 第 2 章 | Belle II 実験             | 2  |
| 2.1   | Belle II の物理            | 2  |
| 2.2   | SuperKEKB 加速器           | 4  |
| 2.3   | Belle II 測定器            | 5  |
| 2.3.1 | 中央飛跡検出器 (CDC)           | 6  |
| 第 3 章 | Level-1 トリガー            | 10 |
| 3.1   | Level-1 トリガーの概要         | 10 |
| 3.2   | Universal Trigger Board | 11 |
| 3.3   | CDC トリガー                | 12 |
| 3.3.1 | Track Segment Finder    | 13 |
| 3.3.2 | 2D Track Finder         | 14 |
| 3.3.3 | Event Timing Finder     | 14 |
| 3.3.4 | 3D Track Finder         | 14 |
| 3.4   | 3D Fitter               | 14 |
| 3.4.1 | 原理                      | 15 |
| 3.4.2 | ヒット選択                   | 20 |
| 3.4.3 | フィッティング                 | 21 |
| 3.4.4 | 旧アルゴリズムの問題点             | 22 |
| 3.5   | トリガーレート                 | 23 |

|              |  |           |
|--------------|--|-----------|
| 3.5.1        | Beam background . . . . .                | 24        |
| 3.5.2        | イベントの分類 . . . . .                        | 25        |
| 3.5.3        | トリガービット . . . . .                        | 26        |
| 3.5.4        | 目標ルミノシティにおけるトリガーレート . . . . .            | 26        |
| <b>第 4 章</b> | <b>3次元飛跡再構成の新アルゴリズム: Hough selection</b> | <b>29</b> |
| 4.1          | Hough selection アルゴリズムの概要 . . . . .      | 29        |
| 4.1.1        | 使用するヒット数の増加 . . . . .                    | 29        |
| 4.1.2        | Hough voting . . . . .                   | 31        |
| 4.1.3        | ヒット選択 . . . . .                          | 34        |
| 4.1.4        | フィッティング . . . . .                        | 35        |
| 4.2          | 解析手法 . . . . .                           | 35        |
| 4.2.1        | 使用ソフトウェア . . . . .                       | 36        |
| 4.2.2        | 使用データ . . . . .                          | 36        |
| 4.2.3        | トリガー飛跡とオフライン飛跡のマッチング . . . . .           | 37        |
| 4.2.4        | 飛跡の選別 . . . . .                          | 38        |
| 4.2.5        | 性能指標 . . . . .                           | 39        |
| 4.3          | ピーク検出手法およびパラメータの最適化 . . . . .            | 39        |
| 4.3.1        | 各ピーク検出手法の概要 . . . . .                    | 39        |
| 4.3.2        | 最適化するパラメータ . . . . .                     | 40        |
| 4.3.3        | 最適化の結果 . . . . .                         | 41        |
| 4.4          | 性能評価 . . . . .                           | 41        |
| 4.5          | トリガーレートの見積もり . . . . .                   | 46        |
| <b>第 5 章</b> | <b>Hough selection アルゴリズムのファームウェア実装</b>  | <b>48</b> |
| 5.1          | ファームウェアの概略 . . . . .                     | 48        |
| 5.2          | Hough voter モジュールの詳細 . . . . .           | 52        |
| 5.2.1        | 直線の投票 . . . . .                          | 52        |
| 5.2.2        | ヒットの $z$ 情報の反映 . . . . .                 | 53        |

|              |  |           |
|--------------|--|-----------|
| 5.2.3        | 集計 . . . . .   | 55        |
| 5.2.4        | ピーク検出 . . . . .  | 56        |
| 5.3          | リソース使用量とレイテンシ . . . . .  | 56        |
| <b>第 6 章</b> | <b>結論</b>  | <b>58</b> |
|              | <b>謝辞</b>  | <b>59</b> |
| <b>付録 A</b>  | <b>Hough selection アルゴリズムのパラメータ最適化の詳細</b>  | <b>60</b> |
| A.1          | $z_0$ 方向のセル数 $N_{\text{cell},z_0}$ 、 $\cot \theta$ 方向のセル数 $N_{\text{cell},\cot \theta}$ 、<br>得票数のしきい値 $N_{\text{threshold}}$ 、およびピーク検出手法の最適化 . . . . . | 60        |
| A.2          | ヒット選択範囲 $z_{\text{selection}}$ の最適化 . . . . .  | 61        |
| A.3          | ファームウェア実装を考慮したパラメータの決定 . . . . .   | 61        |
|              | <b>参考文献</b>  | <b>64</b> |

# 第 1 章

## 序論

標準理論は高精度で素粒子の相互作用を記述できるが、宇宙に存在する物質・反物質の非対称性など標準理論では説明できない問題が知られている。Belle II 実験はつくばにある SuperKEKB 加速器で生成した  $B$  中間子や  $D$  中間子、 $\tau$  レプトンの崩壊の精密測定を通じて、標準理論を超える物理の探索を行う実験である。Belle II 測定器は目的に応じた複数の検出器とデータ収集システム、そして電子回路のトリガーシステムからなる。このトリガーシステムは Level-1 トリガーシステムと呼ばれ、データ収集の前にリアルタイムで背景事象を排除してデータ読み出しの頻度を許容範囲以下に抑えつつ、興味ある物理事象に対する検出効率を高く保つ役割がある。しかし、近年ルミノシティが高くなるにつれビーム由来の背景事象が増えてきたことで、目標ルミノシティにおいてトリガーレートが上限値を超えてしまうことが危惧されている。問題となっている背景事象の多くは衝突点から離れた場所から来るため、中央飛跡検出器と呼ばれるドリフトチェンバーの情報を用いて飛跡を再構成し、発生点の場所をもとに排除できる。本研究は、中央飛跡検出器トリガーにおける 3 次元飛跡再構成の新たなアルゴリズムの開発に関する研究である。

第 2 章では Belle II 実験の概略として、探索できる物理、SuperKEKB 加速器、そして Belle II 測定器の説明を行う。第 3 章では Level-1 トリガーシステムの概略と、中央飛跡検出器トリガーの 3 次元飛跡再構成アルゴリズムについて説明し、本研究の動機となる現在のアルゴリズムの問題点とトリガーレートに関する問題を指摘する。第 4 章では新たに開発した Hough selection アルゴリズムについて説明し、現在のアルゴリズムと比較したシミュレーションによる性能評価の結果を示す。第 5 章では電子回路にアルゴリズムを実装するためのファームウェア開発について説明する。最後の第 6 章では本研究の結論を述べる。

## 第 2 章

# Belle II 実験

本章では Belle II 実験で探索できる物理の一例を示し、その後実験で用いられる SuperKEKB 加速器と Belle II 測定器について説明する。

### 2.1 Belle II の物理

標準理論において  $W$  ボソンとクォークの相互作用ラグランジアンは、クォークのフレーバー固有状態  $u'_{Li}, d'_{Li}$  を用いて

$$\mathcal{L} = \frac{g}{\sqrt{2}} [\bar{u}'_{Li} \gamma^\mu d'_{Li} W_\mu^+ + \text{h.c.}] \quad (2.1)$$

と表せる。 $u, d$  はそれぞれアップタイプ、ダウンタイプのクォークを表し、添え字の  $L$  は左巻きであることを表し、 $i (= 1, 2, 3)$  は世代を表す。ここで、クォークの質量固有状態  $u_{Li}, d_{Li}$  を基底変換のユニタリ行列  $V_L^u, V_L^d$  を用いて

$$u_{Li} = V_{Lij}^u u'_{Lj}, \quad d_{Li} = V_{Lik}^d d'_{Lk} \quad (2.2)$$

と表すと、式 (2.1) はクォークの質量固有状態を用いて以下のように表せる。

$$\begin{aligned} \mathcal{L} &= \frac{g}{\sqrt{2}} [\bar{u}_{Lj} \gamma^\mu V_{Lji}^u V_{Lik}^{d\dagger} d_{Lk} W_\mu^+ + \text{h.c.}] \\ &= \frac{g}{\sqrt{2}} [\bar{u}_{Lj} \gamma^\mu V_{jk} d_{Lk} W_\mu^+ + \text{h.c.}] \end{aligned} \quad (2.3)$$

ここで、ユニタリ行列

$$V = V_L^u V_L^{d\dagger} = \begin{pmatrix} V_{ud} & V_{us} & V_{ub} \\ V_{cd} & V_{cs} & V_{cb} \\ V_{td} & V_{ts} & V_{tb} \end{pmatrix} \quad (2.4)$$

は Cabibbo-Kobayashi-Maskawa (CKM) 行列として知られ、クォーク質量固有状態の世代間の混合を表す [1] [2]。

CKM 行列のユニタリティ条件を複素平面上で表したものはユニタリティ三角形として知られ、標準理論ではユニタリティ三角形は閉じた三角形をなす。よく使われるユニタリティ三角形に、ユニタリティ条件  $V^\dagger V = 1$  の (3,1) 成分

$$V_{ud}V_{ub}^* + V_{cd}V_{cb}^* + V_{td}V_{tb}^* = 0 \quad (2.5)$$

に基づくものがある。式 (2.5) を  $V_{cd}V_{cb}^*$  で割った

$$\frac{V_{ud}V_{ub}^*}{V_{cd}V_{cb}^*} + 1 + \frac{V_{td}V_{tb}^*}{V_{cd}V_{cb}^*} = 0 \quad (2.6)$$

を複素平面上に表すと、図 2.1 に示すようなユニタリティ三角形が描ける。ここで、実数のパラメータ  $\bar{\rho}, \bar{\eta}$  を  $\bar{\rho} + \bar{\eta}i := -\frac{V_{ud}V_{ub}^*}{V_{cd}V_{cb}^*}$  で定義した。この時、三角形の内角  $\phi_1, \phi_2, \phi_3$  は

$$\phi_1 = \arg \left[ -\frac{V_{cd}V_{cb}^*}{V_{td}V_{tb}^*} \right], \quad \phi_2 = \arg \left[ -\frac{V_{td}V_{tb}^*}{V_{ud}V_{ub}^*} \right], \quad \phi_3 = \arg \left[ -\frac{V_{ud}V_{ub}^*}{V_{cd}V_{cb}^*} \right] \quad (2.7)$$

と表すことができる。これらの角度はすべて添え字に  $b$  をもつ CKM 行列要素を含むため、 $b$  クォークが関わる物理過程を通じて測定できる。Belle II 実験では  $B$  中間子の崩壊を精密測定できるため、ユニタリティ三角形の測定を通して標準理論の検証および標準理論を超えた新物理の探索ができる。

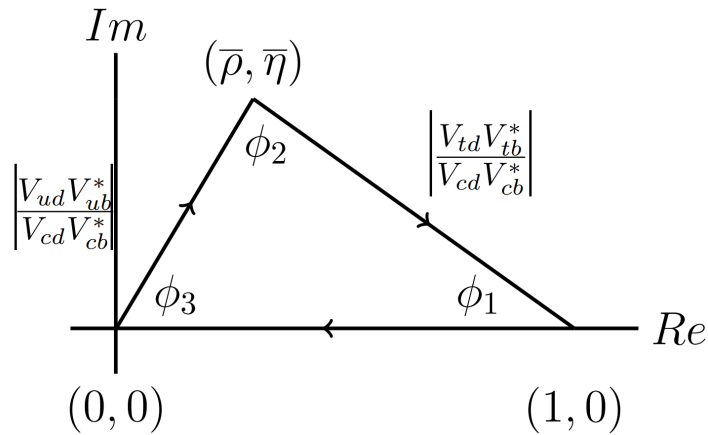


図 2.1 ユニタリティ三角形 [3]



## 2.2 SuperKEKB 加速器

Belle II 実験では、 $B$  中間子を生成するために、茨城県つくば市の高エネルギー加速器研究機構にある SuperKEKB 加速器を用いる。SuperKEKB 加速器は図 2.2 に示すような円型の衝突型加速器であり、別々のリングで加速された電子ビームと陽電子ビームを Belle II 測定器の地点で衝突させる。電子、陽電子のエネルギーはそれぞれ 7 GeV、4 GeV に設定されており、重心系エネルギーは 10.58 GeV である。この重心系エネルギーは質量  $10.58 \text{ GeV}/c^2$  の  $\Upsilon(4S)$  粒子の生成のために調整されており、 $\Upsilon(4S)$  粒子は崩壊分岐比 96% 以上で  $B$  中間子対に崩壊する [4]。電子と陽電子のエネルギーが非対称になっていることにより、重心系でほぼ静止して生成された  $B$  中間子対が実験室系で電子ビームの方向にブーストされ、崩壊点の位置に差が生じる。この位置の差を測定することで CP 非対称性の測定に重要な崩壊の時間差を得ることができる。

Belle II 実験は発生確率の低い事象でも高いルミノシティによって探索できるルミノシティフロンティアの実験であり、ルミノシティの向上にむけて様々な工夫がされている。以下にルミノシティの表式を示す [6]。

$$L = \frac{\gamma_{\pm}}{2er_e} \left( \frac{I_{\pm} \xi_{y\pm}}{\beta_{y\pm}^*} \right) R \quad (2.8)$$

ここで  $\gamma$  は Lorentz 因子、 $e$  は電気素量、 $r_e$  は古典電子半径、 $I$  はビーム電流、 $\xi_y$  はビーム同士の相互作用の影響を表すビームビームパラメータ、 $\beta_y^*$  はビームの包絡線を表す衝突点におけるペー

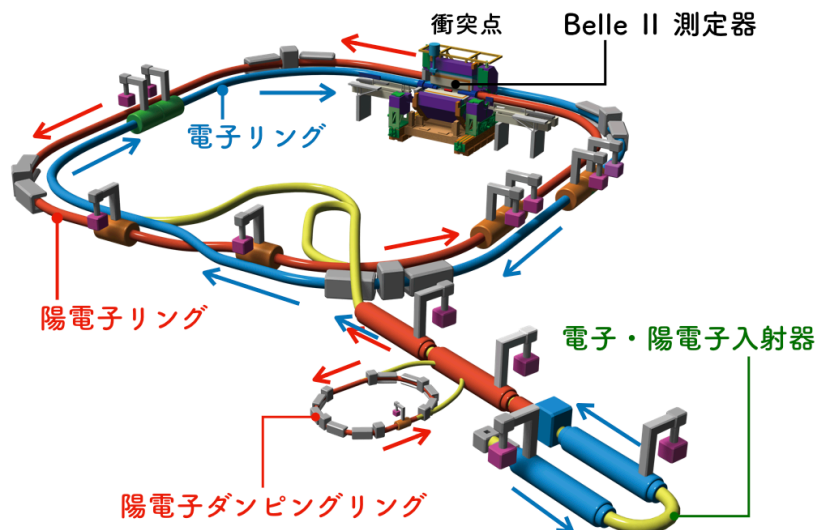


図 2.2 SuperKEKB 加速器 [5]

タ関数、 $R$  はビームの交差に伴う補正因子である。添え字の  $+$  と  $-$  はそれぞれ陽電子ビームと電子ビームを表すが、どちらを選んでも  $\xi_y$  がもう片方のビームの情報を含んでいるためルミノシティは同じ値となる。 $\beta_y^*$  を小さくするために、SuperKEKB 加速器では Nano-Beam scheme と呼ばれるビーム衝突方式を採用している [6]。Nano-Beam scheme では、ビームを大きな交差角をつけて交差させることで実効的な衝突領域を小さくし、衝突領域の中心から離れるのにしたがってビームが広がってしまうというビームの性質による問題を解決している。図 2.3 に Belle II 実験におけるピークルミノシティおよび積分ルミノシティの計画のプロットを示す。実験終了までに  $50 \text{ ab}^{-1}$  の積分ルミノシティを目指している。

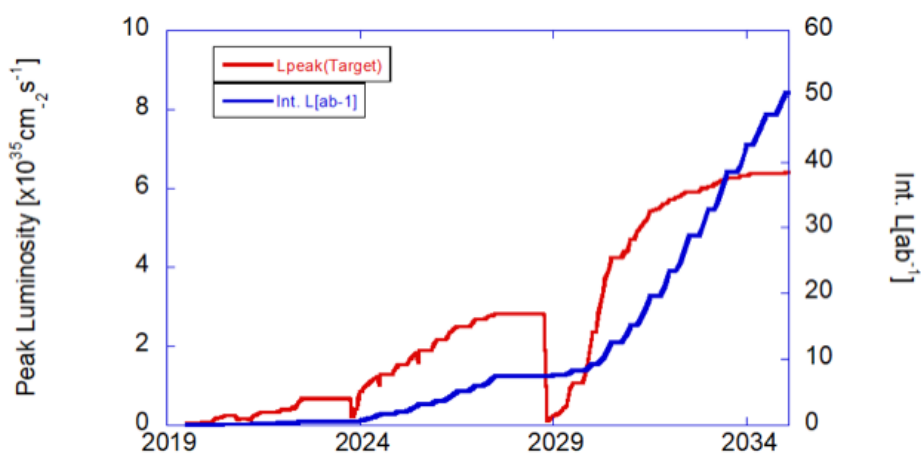


図 2.3 Belle II 実験におけるピークルミノシティおよび積分ルミノシティの計画 [7]

## 2.3 Belle II 測定器

Belle II 測定器は図 2.4 に示すように複数の検出器からなり、それぞれ異なる特徴と役割をもつ。PiXel Detector (PXD) と Silicon Vertex Detector (SVD) はシリコンセンサーで、崩壊点を精度よく測定する。Central Drift Chamber (CDC) はドリフトチェンバーで、荷電粒子の飛跡再構成とエネルギー損失の測定を行う。Time-Of-Propagation counter (TOP) はチェレンコフ光輻射体の石英と光検出器からなり、バレル部 (円筒側面部) における  $K$  中間子と  $\pi$  中間子の識別を行う。Aerogel Ring-Imaging Cherenkov detector (ARICH) はチェレンコフ光輻射体のエアロゲルと光検出器からなり、エンドキャップ部 (円筒底面部) における  $K$  中間子と  $\pi$  中間子の識別を行う。Electromagnetic Calorimeter (ECL) はカロリメータであり、電磁シャワーを起こす光子や電子、陽電子のエネルギー測定を行う。KLong/Muon detector (KLM) は吸収材としての鉄の層と検出器 (ガスチェンバーまたはプラスチックシンチレータ) の層を交互に配置した検出器であり、 $K_L^0$  粒子と  $\mu$  粒子の識別を行う。これらの検出器に加え、事象選択を行うトリガーシステム

や、データ収集システムによってデータの取得が行われる。トリガーシステムにはデータ収集前に電子回路上でリアルタイム処理を行う Level-1 トリガーと、データ収集後に PC サーバ上のソフトウェアで処理を行う High Level Trigger の 2 種類が存在する。

ここで、以降用いる座標系として図 2.5 に示す座標系を定義する。この座標系はおおよそ電子ビームの進行方向を  $z$  軸、鉛直方向を  $y$  軸とする右手系をなす。また、CDC 内での位置や飛跡を表現するために、円筒形の CDC の半径方向を  $r$  軸、角度方向を  $\varphi$  軸、電子ビームの進行方向を  $z$  軸とする円筒座標系  $(r, \varphi, z)$  を用いることがある。

### 2.3.1 中央飛跡検出器 (CDC)

本項では、本研究に関連する検出器である CDC について説明する。CDC は多数のワイヤーとその間を満たすガスからなるドリフトチェンバーであり、通過した荷電粒子の位置情報とエネルギー損失の測定を担っている。CDC の中を通過した荷電粒子はガス分子を電子と陽イオンに電離させる。この電子が電場によって発生位置からワイヤーまで移動するのにかかるドリフト時間を用いた位置情報の補正によって、約  $100 \mu\text{m}$  の高い位置分解能が得られるのが CDC の特徴である [6]。解析において、位置情報から再構成された飛跡は運動量、電荷、崩壊点の決定や粒子識別に用いられ、エネルギー損失は粒子識別に用いられる。また、位置情報は Level-1 トリガーでの飛跡再構成を経てトリガー信号を出すためにも用いられる。

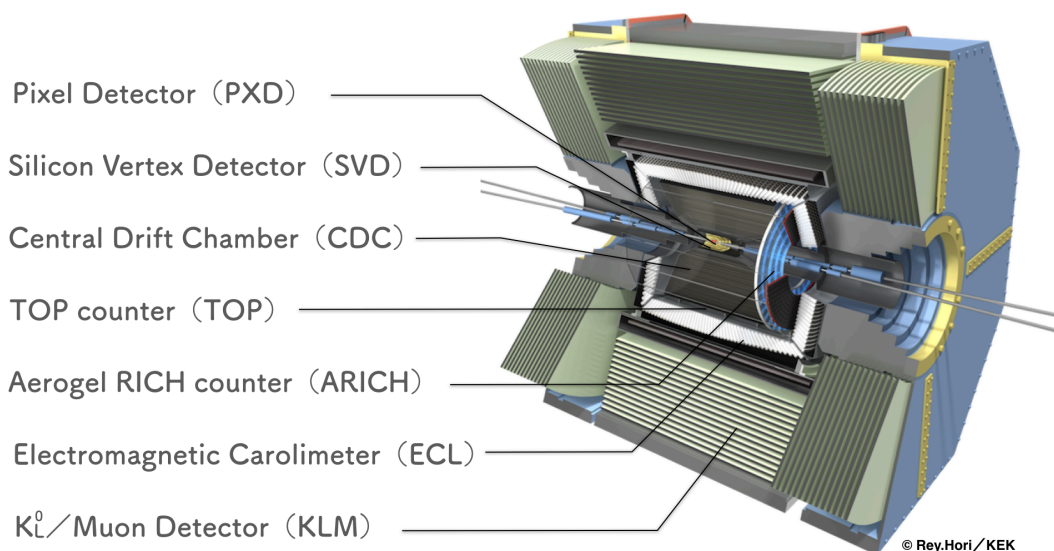


図 2.4 Belle II 測定器 [8]

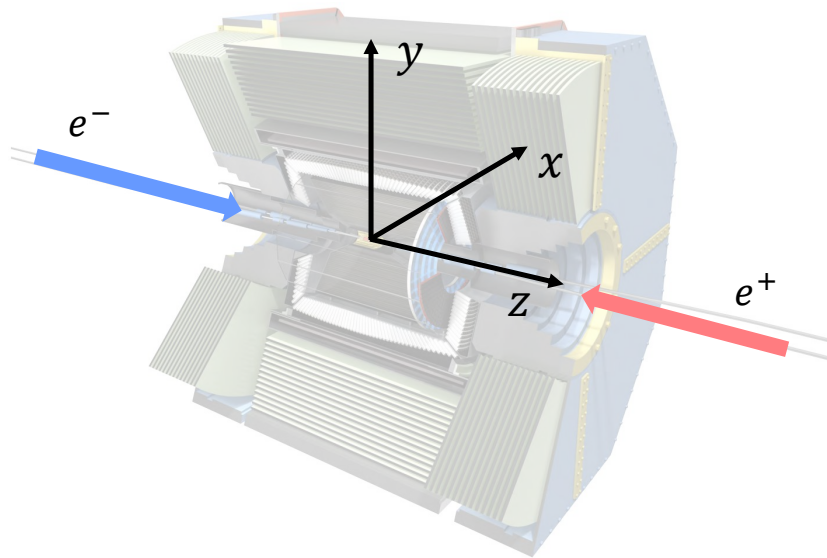


図 2.5 本論文で用いる座標系

### (1) CDC の構造

CDC のワイヤーには、高電圧が印加され信号の読み出しに使われるセンスワイヤーと、接地され電場の形成に用いられるフィールドワイヤーが存在する。1つのセンスワイヤーとその周辺のフィールドワイヤーをまとめてセルと呼び、図 2.6 のようにワイヤーが配置されている。以降はワイヤーといった場合センスワイヤーを指すものとする。

図 2.7 に CDC の  $r$ - $z$  平面での断面図を示す。 $r$  軸は半径方向を表す。赤線はスモールセルのワ

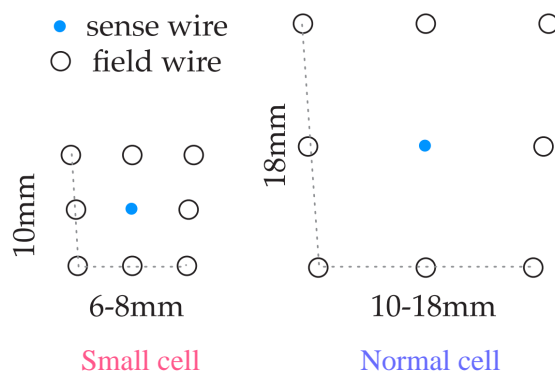


図 2.6 1つのセル内におけるセンスワイヤーとフィールドワイヤーの配置 [9]。大きさの異なる2種類のセルが存在し、スモールセルとノーマルセルと呼ばれる。

ワイヤー、青線はノーマルセルのワイヤーを表す。また、実線は  $z$  軸に平行なアキシャルワイヤー、破線は  $z$  に対し傾きを持つステレオワイヤーを表す。同じ種類のワイヤーからなる数レイヤーをまとめて superlayer とよび、内側から  $0, 1, \dots, 8$  とナンバリングされている。Superlayer0 は8レイヤー、superlayer1 から superlayer8 は6レイヤーからなり、CDC 全体では56レイヤーからなる。

図 2.8 に  $\varphi$ - $z$  平面でのアキシャルワイヤーおよびステレオワイヤーの模式図を示す。ステレオワイヤーと  $z$  軸のなす角をステレオ角  $\theta_{\text{stereo}}$  と呼び、 $z$  軸負方向の端点の  $\varphi$  座標より正方向の端点の  $\varphi$  座標のほうが大きい場合に  $\theta_{\text{stereo}}$  を正と定め、逆の場合は負と定めている。ステレオ角の大きさは  $45 \text{ mrad}$ – $74 \text{ mrad}$  程度である [6]。Superlayer のうち、アキシャルワイヤーからなるものを A、 $\theta_{\text{stereo}}$  が正のステレオワイヤーからなるものを  $S^+$ 、 $\theta_{\text{stereo}}$  が負のステレオワイヤーからなるものを  $S^-$  と表すと、内側から順に (AS $^+$ AS $^-$ AS $^+$ AS $^-$ A) と並んでいる。

## (2) 読み出し回路

CDC ワイヤーから出された信号は、CDC の円筒底面部にある円板の外側に配置された読み出し回路によってデジタル信号に変換され、トリガーやデータ収集システムに送られる。読み出し回路には Amp-Shaper-Discriminator chips (ASDs) と呼ばれる信号の増幅、整形をするチップが存在する [10]。ASD チップは複数のチャンネルからなるが、あるチャンネルに信号が入力された場合に周辺のチャンネルにも信号が生じることがある。この現象をクロストークと呼び、トリガーで

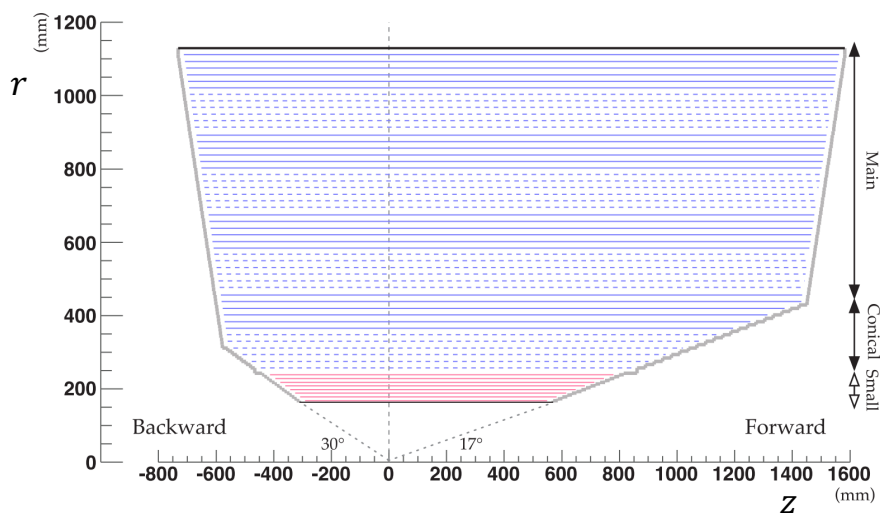


図 2.7 CDC の  $r$ - $z$  平面での断面図 [9]。赤線はスモールセルのワイヤー、青線はノーマルセルのワイヤーを表す。また、実線は  $z$  軸に平行なアキシャルワイヤー、破線は  $z$  に対し傾きを持つステレオワイヤーを表す。

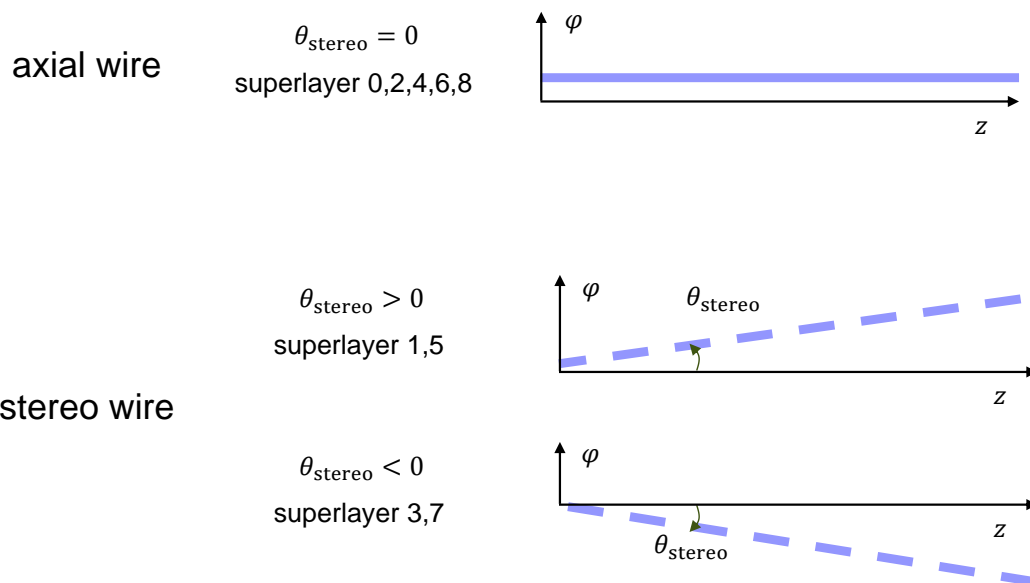


図 2.8  $\varphi$ - $z$  平面でのアキシアルワイヤーおよびステレオワイヤーの模式図。ワイヤーと  $z$  軸のなす角をステレオ角  $\theta_{\text{stereo}}$  と呼び、 $z$  軸負方向の端点の  $\varphi$  座標より正方向の端点の  $\varphi$  座標のほうが大きい場合に  $\theta_{\text{stereo}}$  を正と定め、逆の場合は負と定めている。

の飛跡再構成を難しくする原因の一つとなっている。

## 第3章

# Level-1 トリガー

一般に、複数のデータの中から興味あるデータのみを取捨選択するシステムのことをトリガーという。Belle II 実験においても限られたデータ読み出し性能と計算機台数のもとでなるべく多くの興味ある物理事象のデータを取得するために、Level-1 トリガーと呼ばれる電子回路によるトリガーシステムを用いている。本章では Level-1 トリガーの概要を説明した後、本研究の主題である CDC トリガー、特に 3 次元飛跡再構成のアルゴリズムについて説明し、最後に本研究の動機となるトリガーレートに関して議論する。

### 3.1 Level-1 トリガーの概要

Belle II の Level-1 トリガーシステムには以下の要求が定められている [6]。レイテンシはイベント\*1が発生してからトリガー信号を発行するまでの時間を指す。

1.  $\Upsilon(4S) \rightarrow B\bar{B}$  崩壊事象および continuum ( $e^+e^- \rightarrow q\bar{q}$ ) 事象に由来するハドロニック崩壊事象へのトリガー効率がほぼ 100%
2. 平均トリガーレートは 30 kHz 以下
3. トリガーシステムのレイテンシは固定で 4.5  $\mu$ s
4. イベント発生時間の同定精度は 10 ns 以下
5. 二つのイベントの時間間隔は 400 ns 以上

現時点でこれらの要求は満たされているが、後述するように目標ルミノシティでは 2 つめのトリガーレートへの要求が満たされなくなる。この問題を解決するために本研究では Level-1 トリガー

---

\*1 本論文ではイベントという用語を事象とほぼ同じ意味で用いているが、ある時間間隔内に起こった事象をまとめて扱うための解析におけるデータの単位といったニュアンスを含ませている。

の一部に改良を加えるが、その際は1つめのトリガー効率への要求と3つめのレイテンシへの要求も満たすように気を配る必要がある。なお、4つめと5つめの要求は影響を受けないので問題ない。

図 3.1 に Level-1 トリガーの構成図を示す。Level-1 トリガーシステムは CDC、ECL、TOP、KLM の 4 つの検出器のサブトリガーと、その後段に位置する Global Reconstruction Logic (GRL)、そして最後段に位置する Global Decision Logic (GDL) からなる。GRL は CDC トリガーからの飛跡情報と ECL トリガーからのエネルギー情報など、複数のサブトリガーからの情報のマッチングを行う。GDL は各サブトリガーと GRL の情報を受け取って入力タイミグを揃え、トリガー条件の論理演算を行い最終的な Level-1 トリガー信号を出す。

## 3.2 Universal Trigger Board

Belle II の Level-1 トリガーシステムでは Universal Trigger Board という汎用トリガーボードを開発し使用している。現在使用されている最新のバージョンのものは UT4、その前のバージョンのものは UT3 と名前が付けられている。後に説明する、3次元飛跡再構成を行うためのアルゴリズムは UT3 に実装されていたが、本研究で新たに開発したアルゴリズムは UT4 に実装した。表 3.1 に UT3 と UT4 の性能の比較を示す。デジタル信号の処理には Field-Programmable Gate Array (FPGA) を用いている。表中の入出力の項目に記載されている GYX、GTH および GTY は Xilinx 社の高速シリアルトランシーバーの名称である。ボード間では電気ノイズを避けるために光信号で通信しており、ボードの入出力部分で光トランシーバーによって電気信号と光信号が相互に変換される。

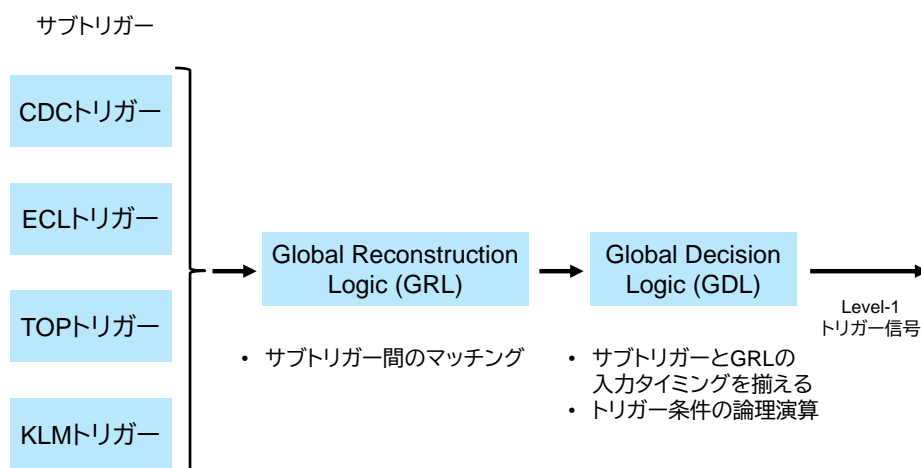


図 3.1 Level-1 トリガーの構成図



表 3.1 Belle II 汎用トリガーボード UT3 と UT4 の性能比較

|              | UT3                      | UT4                             |
|--------------|--------------------------|---------------------------------|
| FPGA シリーズ    | Xilinx 社 Virtex-6 [11]   | Xilinx 社 Virtex UltraScale [12] |
| FPGA 型番      | XC6VHX380/565T           | XCVU080/160                     |
| Logic gate 数 | 382k/580k                | 975k/2026k                      |
| 全体の通信速度      | 530 Gbps                 | 1300 Gbps                       |
| 入出力          | GTH 11.2 Gbps × 24 lanes | GTY 25 Gbps × 32 lanes          |
|              | GTX 6.6 Gbps × 40 lanes  | GTH 15 Gbps × 32 lanes          |

### 3.3 CDC トリガー

CDC トリガーの役割は、CDC からヒット情報を受け取って飛跡を再構成し後段の GRL へ結果を送ることである。図 3.2 に CDC トリガーの構成図を示す。CDC 読み出し回路の次段に存在する Merger は複数の CDC 読み出し回路からの情報を圧縮しているだけなので、本節の説明では省略する。

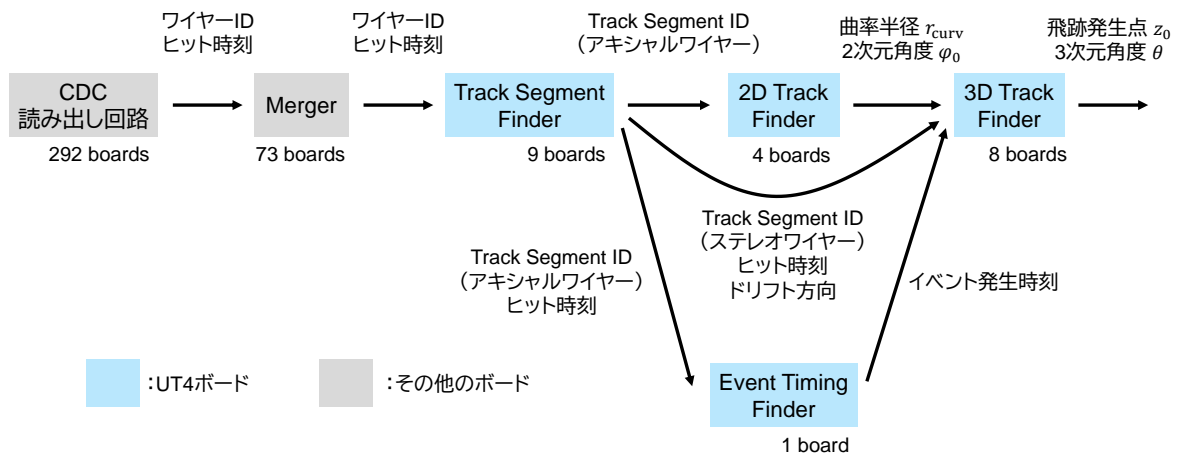


図 3.2 CDC トリガーの構成図

### 3.3.1 Track Segment Finder

Track Segment Finder は CDC 読み出し回路からワイヤーの識別番号 (ID) とヒット時刻を受け取り、ヒット情報の圧縮を行う。ボード間の光トランシーバによる通信速度には上限があるため、全ての CDC ヒットを後段に送ることはできない。仮に送ることができたとしても、すべてのヒットを処理するための回路の規模は大きくなってしまふ。そのため、図 3.3 に示すような Track Segment という構造を用いてヒットが連なった飛跡の断片を検出し、その代表として図中赤いセルで示した priority wire のヒットのみを後段に送っている。Track segment は様々な曲率や角度をもつ飛跡に対応するために最内層の superlayer に対しては逆三角型、その他の superlayer に対しては砂時計型の形をしている。 $\varphi$  方向のセルの数だけ track segment が定義され、それぞれ重なっている。

Track Segment Finder は CDC 読み出し回路からワイヤーの ID とヒット時刻を受け取り、ドリフト時間の最大値である約 500 ns の間に track segment 内で 5 レイヤーのうち 4 レイヤーにヒットが存在するとき、priority wire の ID とヒット時刻、およびドリフト方向を後段に送る [13]。ドリフト方向は粒子がワイヤーに対しどの方向からドリフトしてきたかを表す情報で、後述する 3D Track Finder で用いられる。Priority wire にヒットが存在しないときは、その上のレイヤーの 2 つのワイヤーのうちヒット時刻が早いほうの情報が送られる。なお、UT3 から UT4 へのアップグ

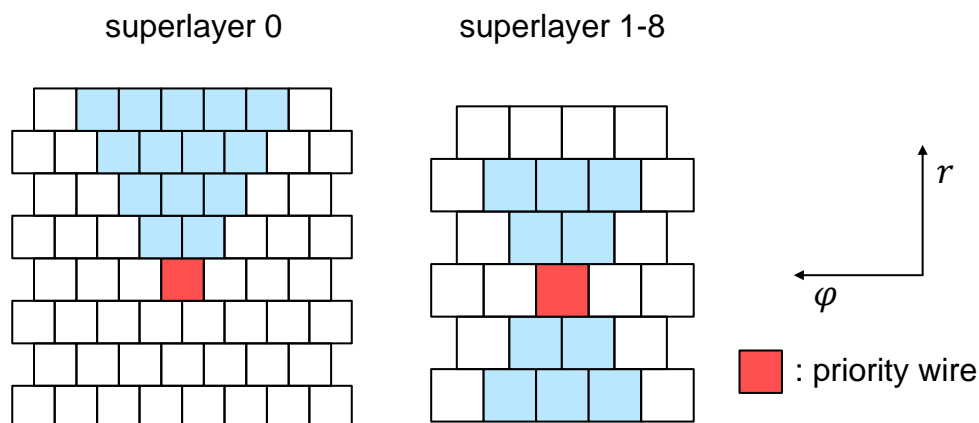


図 3.3 Track Segment の構造。色のついたセルが 1 つの track segment をなす。赤色のセルは priority wire と呼ばれる代表のワイヤーであることを示す。

レードに伴う通信速度の向上により、priority wire の情報だけでなく track segment 内の全ワイヤーの情報を送れるようになってきているが、現時点では次に述べる 2D Track Finder のみが全ワイヤーの情報を利用している。

### 3.3.2 2D Track Finder

Track Segment Finder からアキシアルワイヤーのヒット情報を受け取り 2次元飛跡を再構成する。Hough 変換とよばれる、画像処理の分野において円や直線などのパターン検出に用いられるアルゴリズムを使い、円弧の曲率半径  $r_{\text{curv}}$  と  $x$ - $y$  平面における原点での角度  $\varphi_0$  を求める。

2022 年に行われた 2D Track Finder のアップグレードで、track segment 中央の priority wire のみを使用していたアルゴリズムが、track segment 内の全ヒットを使用するよう変更された [14]。この変更により、実際に飛跡が存在しないにも関わらず CDC 読み出し回路のクロストークのせいでトリガーが誤検知してしまう飛跡を削減できるようになった。このような飛跡は fake track と呼ばれる。

### 3.3.3 Event Timing Finder

Track Segment Finder からのヒット情報を入力として受け取り、イベントの発生時刻を求める。クロストークなどによるヒットを排除するために、Hough 変換で 2次元飛跡を再構成しその 2次元飛跡に紐づいたヒット情報を用いてイベント発生時刻を決定する [15]。

### 3.3.4 3D Track Finder

Track Segment Finder からステレオワイヤーのヒット情報、2D Track Finder から 2次元飛跡の曲率半径  $r_{\text{curv}}$  と  $x$ - $y$  平面における角度  $\varphi_0$ 、そして Event Timing Finder からイベント発生時刻を入力として受け取り、3次元飛跡を再構成する。3D Track Finder にはフィッティングを用いる 3D Fitter と、ニューラルネットワークを用いた Neurotrigger の 2種類があり、現在はより性能の良い Neurotrigger が用いられている。本研究は 3D Fitter についての研究である。

## 3.4 3D Fitter

本節では、3D Fitter の原理と旧アルゴリズムについて説明し、その後旧アルゴリズムの問題点を指摘する。

### 3.4.1 原理

#### 3次元飛跡のパラメータ表示

CDC 内部には  $z$  軸正方向に磁場がかけられているため、図 3.4 に示すように CDC 内で荷電粒子がなす 3次元飛跡はらせんとなり、3次元飛跡を  $x$ - $y$  平面上に射影した 2次元飛跡は円弧となる。2次元飛跡上の弧長  $s$  をパラメータとする 3次元飛跡のパラメータ表示は以下ようになる [13]。

$$\begin{pmatrix} x(s) \\ y(s) \\ z(s) \end{pmatrix} = \begin{pmatrix} r_{\text{curv}}(\sin(s/r_{\text{curv}} - \varphi_0) + \sin \varphi_0) + x_0 \\ r_{\text{curv}}(\cos(s/r_{\text{curv}} - \varphi_0) - \cos \varphi_0) + y_0 \\ \cot \theta \cdot s + z_0 \end{pmatrix} \quad (3.1)$$

ここで  $r_{\text{curv}}$  は 2次元飛跡の曲率半径、 $\varphi_0$  は  $x$ - $y$  平面での 2次元飛跡の飛跡発生点における運動量方向の角度、 $(x_0, y_0, z_0)$  は飛跡発生点の座標、 $\theta$  は飛跡発生点における 3次元飛跡の運動量方向が  $z$  軸となす角を表す。

CDC トリガーでは衝突点から生成される飛跡のみを再構成するために、 $x_0 = 0, y_0 = 0$  と近似している。そのため CDC トリガーで求めるべきパラメータは  $r_{\text{curv}}, \varphi_0, z_0$ 、そして  $\theta$  の 4つである。この 4つのパラメータのうち  $r_{\text{curv}}$  と  $\varphi_0$  は 2D Track Finder で求められ、 $z_0$  と  $\theta$  は 3D Track Finder で求められる。式 (3.1) の  $z$  成分を見ると、ある  $s$  と  $z$  の組は  $s$ - $z$  平面において傾き  $\cot \theta$ 、 $z$  切片  $z_0$  の直線上にあることがわかる。したがって、各ステレオワイヤーごとに求めた

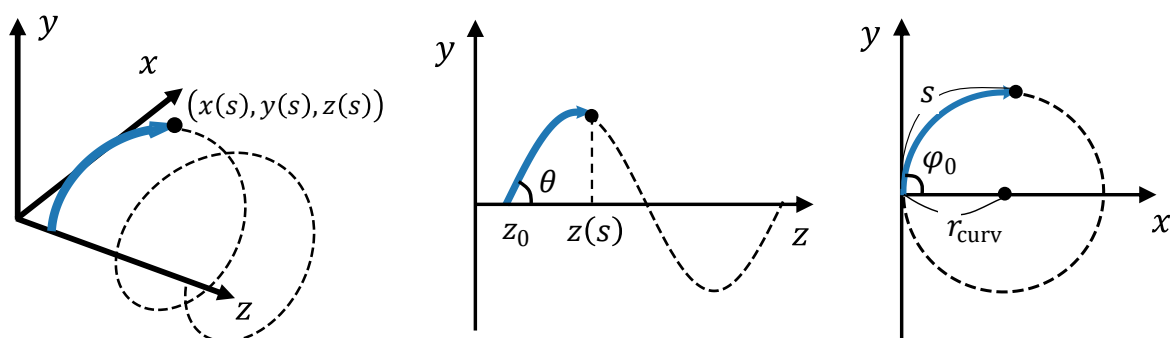


図 3.4 3次元飛跡のらせんを様々な視点からみた図。 $x$ - $y$  平面における 2次元飛跡は円弧となる。

$s$  と  $z$  を用いて直線のフィッティングを行うことで  $z_0$  と  $\cot \theta$  を決定できる。

### ドリフト補正がない場合の $s$ と $z$ の導出

簡単のためドリフト補正がない場合、すなわちヒットのドリフト時間  $t_{\text{drift}}$  が 0 の場合を考える。入力として、2次元飛跡と、それに伴う一つのステレオワイヤーのヒット情報が与えられたとする。すなわち、2次元飛跡の曲率半径  $r_{\text{curv}}$ 、 $x$ - $y$  平面における角度  $\varphi_0$ 、そしてステレオワイヤーの位置の  $r$  座標  $r_{\text{wire}}$ <sup>\*2</sup>が入力変数として与えられる。

まず  $s$  を求める。図 3.5 のように、原点かつ飛跡発生点を点  $O$ 、2次元飛跡とステレオワイヤーの交点を点  $P_{\text{cross}}$ 、2次元飛跡の曲率円の中心を点  $C$ 、 $\angle OCP_{\text{cross}}$  を  $2\alpha$  と定義する。三角形  $OCP_{\text{cross}}$  で余弦定理を使うと以下のように  $\alpha$  を入力変数  $r_{\text{wire}}$ 、 $r_{\text{curv}}$  で表せる。

$$\alpha = \arcsin\left(\frac{r_{\text{wire}}}{2r_{\text{curv}}}\right) \quad (3.2)$$

なお、ここでは  $r_{\text{wire}}/2r_{\text{curv}} \leq 1$ 、すなわち 2次元飛跡の曲率半径はワイヤーと交差しうる大きさをもつと仮定する。この  $\alpha$  を用いて弧  $OP_{\text{cross}}$  の弧長  $s_{\text{cross}}$  は

$$s_{\text{cross}} = r_{\text{curv}} \cdot 2\alpha \quad (3.3)$$

$$(3.4)$$

と求められる。また、図 3.6 に示すように点  $P_{\text{cross}}$  の角度座標  $\varphi_{\text{cross}}$  も  $\alpha$  を用いて

$$\varphi_{\text{cross}} = \varphi_0 - \alpha \quad (3.5)$$

と表すことができる。

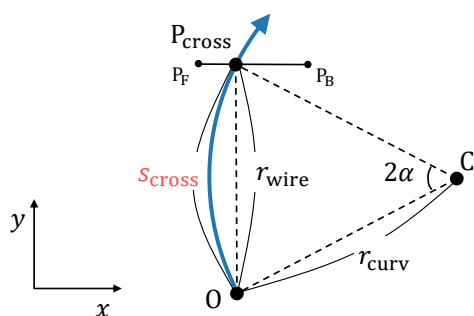


図 3.5 2次元飛跡上の弧長  $s_{\text{cross}}$  は入力変数  $r_{\text{wire}}$  と  $r_{\text{curv}}$  を用いて表せる。

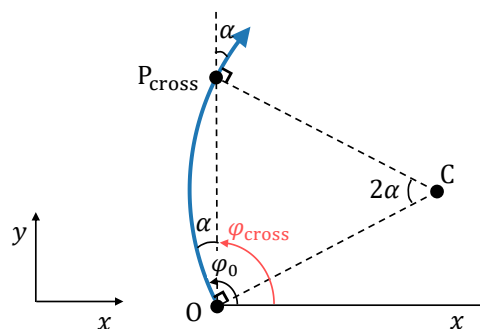


図 3.6 図 3.5 の角度情報に着目した図。点  $P_{\text{cross}}$  の角度座標  $\varphi_{\text{cross}}$  は入力変数  $\varphi_0$  と式 (3.2) で求めた  $\alpha$  を用いて表せる。

<sup>\*2</sup> 厳密にはステレオワイヤー上の位置によって  $r_{\text{wire}}$  は異なるが、その差は無視できると近似する。

次に  $z$  を求める。図 3.7 のように  $x$ - $z$  平面で議論する\*3。ステレオワイヤーと 2 次元飛跡の交点  $P_{\text{cross}}$  はステレオワイヤーの  $z$  軸正方向 (forward) の端点  $P_F$  と  $z$  軸負方向 (backward) の端点  $P_B$  との間を各軸同じ比で分割するから、

$$\frac{z_{\text{cross}} - z_B}{z_F - z_B} = \frac{x_{\text{cross}} - x_B}{x_F - x_B} \quad (3.6)$$

が成り立つ。 $z_{\text{cross}}$  について解くと、

$$\begin{aligned} z_{\text{cross}} &= z_B + \frac{z_F - z_B}{x_F - x_B} \cdot (x_{\text{cross}} - x_B) \\ &\simeq z_B + \cot \theta_{\text{stereo}} \cdot r_{\text{wire}} (\varphi_{\text{cross}} - \varphi_B) \end{aligned} \quad (3.7)$$

ここで  $\theta_{\text{stereo}}$  は 2.3.1 項で説明したステレオワイヤーの  $z$  軸に対する傾きを表すステレオ角であり、 $\cot \theta_{\text{stereo}} = (z_F - z_B)/(x_F - x_B)$  と表せる。 $\varphi_{\text{cross}}$  と  $\varphi_B$  はそれぞれ点  $P_{\text{cross}}$  と点  $P_B$  の  $\varphi$  座標であり、 $x$ - $y$  平面上での直線  $P_{\text{cross}}P_B$  の長さ ( $= x_{\text{cross}} - x_B$ ) を弧  $P_{\text{cross}}P_B$  の長さ ( $= r_{\text{wire}} (\varphi_{\text{cross}} - \varphi_B)$ ) で近似している。

以上の議論は正の電荷をもつ飛跡を例にとったが、負の電荷をもつ飛跡でも  $r_{\text{curv}} \rightarrow -r_{\text{curv}}$  とすれば同様の式が成り立つ。よって、以降は  $r_{\text{curv}}$  を電荷と同じ符号を持つ、符号付きの二次元飛跡の曲率半径と定める。

### ドリフト補正がある場合の $s$ と $z$ の導出

次にドリフト補正がある場合、すなわち  $t_{\text{drift}} > 0$  の場合を考える。 $t_{\text{drift}}$  は Track Segment Finder から入力されるワイヤーのヒット時刻  $t_{\text{wire}}$  と Event Timing Finder から入力されるイベ

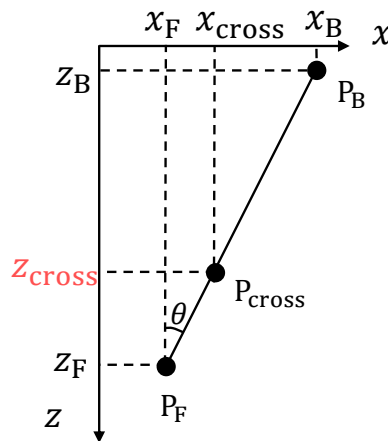


図 3.7 点  $P_{\text{cross}}$  の  $x$  座標  $x_{\text{cross}}$  とステレオ角がわかれば、 $z$  座標  $z_{\text{cross}}$  が求められる。

\*3 今は  $x$ - $y$  平面においてステレオワイヤーが  $x$  軸と平行だと仮定しているが、そうでない場合は  $x$  軸の代わりにステレオワイヤーと平行な軸を用いる必要がある。

ント発生時刻  $t_0$  を用いて  $t_{\text{drift}} = t_{\text{wire}} - t_0$  と求められる。この時、ヒット点  $P_{\text{hit}}$  は図 3.8 のようにワイヤー位置を中心とし距離  $x_{\text{drift}}(t_{\text{drift}})$  を半径とするドリフト円と呼ばれる円上のどこかに存在する。ここで  $x_{\text{drift}}(t)$  はドリフト時間とドリフト距離の関係を表す  $x$ - $t$  カーブと呼ばれる曲線であり、レイヤーごとに実験から求められている。ヒット情報だけからはヒット点がドリフト円上のどこかを定めることはできないが、2次元飛跡の情報から図 3.6 に示したように粒子はワイヤーの半径方向に対し  $\alpha$  の傾きをもって通過したことがわかっているため、図 3.8 に示すようにドリフト方向が Left か Right かの 2 点に絞られる。

さらに、周辺のヒット情報を用いることで Left/Right のどちらかを定めることができる。モンテカルロシミュレーションでヒットパターンとドリフト方向の対応関係を解析し、図 3.9 のように track segment 内の各ヒットパターンのそれぞれに対し Left/Right/Undetermined を設定している [13]。粒子が priority wire の近傍を通るヒットパターンや、ノイズを含むヒットパターンでは Undetermined と判定され、ドリフト補正は適用されない。ここで、別の粒子によるヒットや CDC 読み出し回路のクロストークによるヒットをノイズと定めた。

ドリフト方向を  $\sigma_{L/R}$  (Left の時は +1、Right の時は -1、Undetermined の時は 0) で表すとき、点  $P_{\text{cross}}$  の  $\varphi$  座標  $\varphi_{\text{cross}}$  に対する補正は角度差  $\varphi_{\text{drift}}$  を用いて  $\sigma_{L/R} \cdot \varphi_{\text{drift}}$  と表すことができる。ここで  $\varphi_{\text{drift}}$  は図 3.10 より

$$\begin{aligned} \varphi_{\text{drift}} &= \arctan \left( \frac{x_{\text{drift}}(t_{\text{drift}}) \cdot \cos \alpha}{r_{\text{wire}} - x_{\text{drift}}(t_{\text{drift}}) \cdot \sin \alpha} \right) \\ &\simeq \frac{x_{\text{drift}}(t_{\text{drift}})}{r_{\text{wire}}} \end{aligned} \quad (3.8)$$

と求められる。ここで近似として、 $r_{\text{wire}} \gg x_{\text{drift}}(t_{\text{drift}})$  であることを用いた。また、 $\cos \alpha \sim 1$  と近似しているが、これは横運動量が十分大きく曲率の小さい飛跡に対してのみ成り立つ。

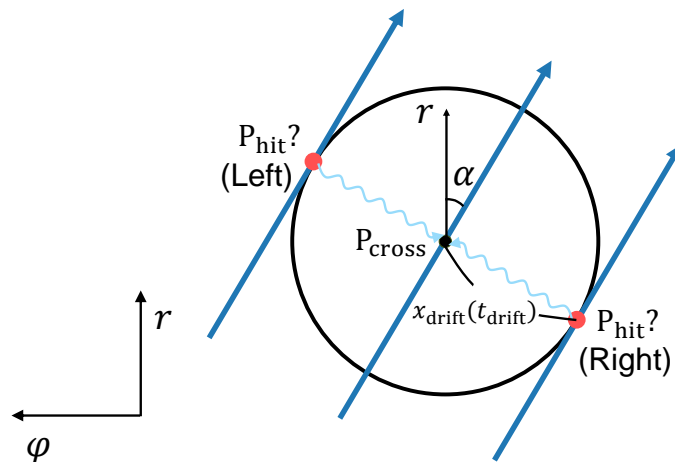


図 3.8 ドリフト距離の補正を加えると、ヒット点はドリフト円上の 2 点のいずれかに絞られる。

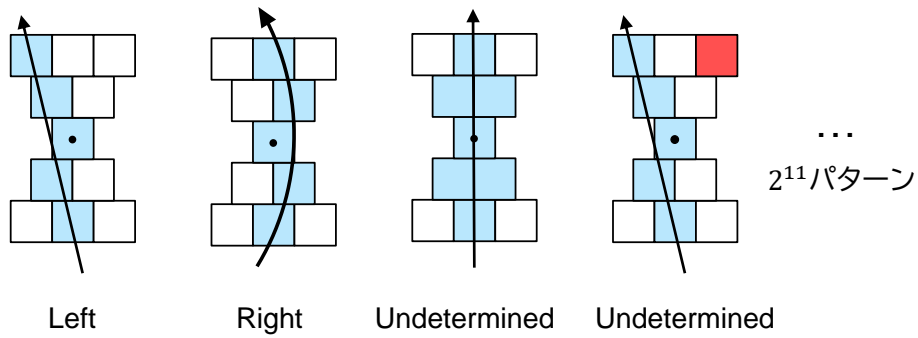


図 3.9 Track segment 内の各ヒットパターンに対しドリフト方向 Left/Right/Undetermined が設定されている。粒子が priority wire の近傍を通る場合や、図中赤いセルで示したようなノイズヒットを含む場合は Undetermined と設定されている。

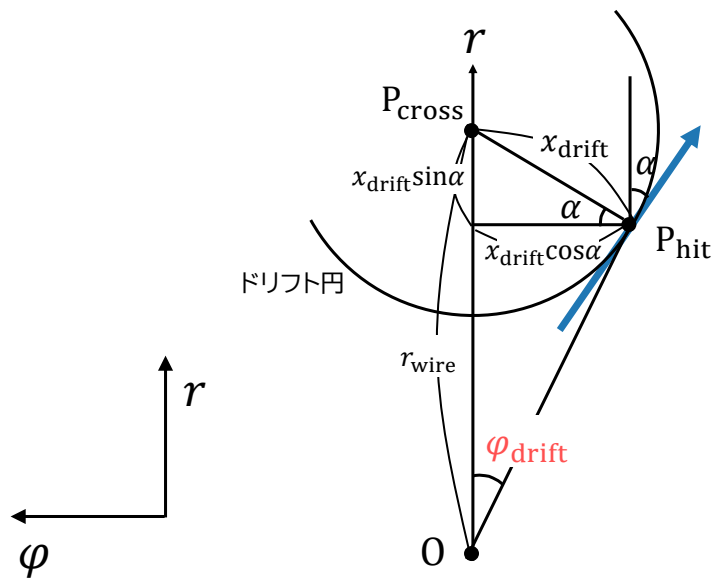


図 3.10 ドリフト補正による角度差  $\varphi_{\text{drift}}$  はワイヤー位置の  $r$  座標  $r_{\text{wire}}$ 、ドリフト距離  $x_{\text{drift}}(t_{\text{drift}})$ 、式 (3.2) で求めた角度  $\alpha$  を用いて表せる。ドリフト距離  $x_{\text{drift}}(t_{\text{drift}})$  を単に  $x_{\text{drift}}$  と省略している。なお、わかりやすさのためドリフト円の大きさを誇張して図示している。



ドリフト補正を考慮すると、2次元飛跡とステレオワイヤーの交点における2次元飛跡の弧長  $s_{\text{cross}}$  と  $z$  座標  $z_{\text{cross}}$  をもとにして、粒子が通過したヒット点における弧長  $s_{\text{hit}}$ <sup>\*4</sup> と  $z$  座標  $z_{\text{hit}}$  を以下のように求めることができる。

$$s_{\text{hit}} = r_{\text{curv}} \cdot 2\alpha \quad (3.9)$$

$$z_{\text{hit}} = z_{\text{B}} + \cot \theta_{\text{stereo}} \cdot r_{\text{wire}} (\varphi_0 - \alpha + \sigma_{\text{L/R}} \cdot \varphi_{\text{drift}} - \varphi_{\text{B}}) \quad (3.10)$$

再度変数を説明すると、

- $r_{\text{curv}}$  は2次元飛跡の曲率半径
- $\alpha$  は式(3.2)で求めた2次元飛跡とワイヤー半径方向の交差角
- $z_{\text{B}}$  はワイヤーの  $z$  軸負方向の端点の  $z$  座標
- $\theta_{\text{stereo}}$  はステレオワイヤーと  $z$  軸のなす角であるステレオ角
- $r_{\text{wire}}$  はワイヤー位置の  $r$  座標
- $\varphi_0$  は  $x$ - $y$  平面における2次元飛跡の原点での角度
- $\sigma_{\text{L/R}}$  はドリフト方向 (Left の時 +1、Right の時 -1、Undetermined のとき 0)
- $\varphi_{\text{drift}}$  は式(3.8)で求めた  $\varphi$  座標に対するドリフト補正
- $\varphi_{\text{B}}$  はワイヤーの  $z$  軸負方向の端点の  $\varphi$  座標

を表す。なお、以降はステレオワイヤーに関して単に  $s$ 、 $z$  といった場合は  $s_{\text{hit}}$ 、 $z_{\text{hit}}$  を指すものとする。

### 3.4.2 ヒット選択

前項の議論から、 $x$ - $y$  平面で2次元飛跡とステレオワイヤーが交わる限り  $s$  と  $z$  が計算できることがわかる。ステレオワイヤーは  $z$  軸に対して傾いているので、図 3.11 に示すように  $x$ - $y$  平面において2次元飛跡と交わりうるステレオワイヤーは1レイヤーの中に複数本存在する。この本数はステレオ角とワイヤーの間隔によって決まっており、レイヤーによって8本から10本である。もしその8-10本のうち複数のワイヤーにヒットが生じた場合は、フィッティングに使うヒットを1つ選ぶ必要がある。複数のヒットが実際の粒子によるもの場合はどれを選んでも正しい  $s$  と  $z$  が計算されるが、ヒットがノイズによる場合はでたらめな  $s$  と  $z$  が計算されてしまう。旧アルゴリズムでは同レイヤーに複数ヒットがある場合には  $z = 0$  に一番近いヒットを選択している [16]。

\*4 実際には弧長  $s_{\text{hit}}$  も補正を受けるため  $s_{\text{cross}}$  からずれるが、そのずれは無視できると近似する。

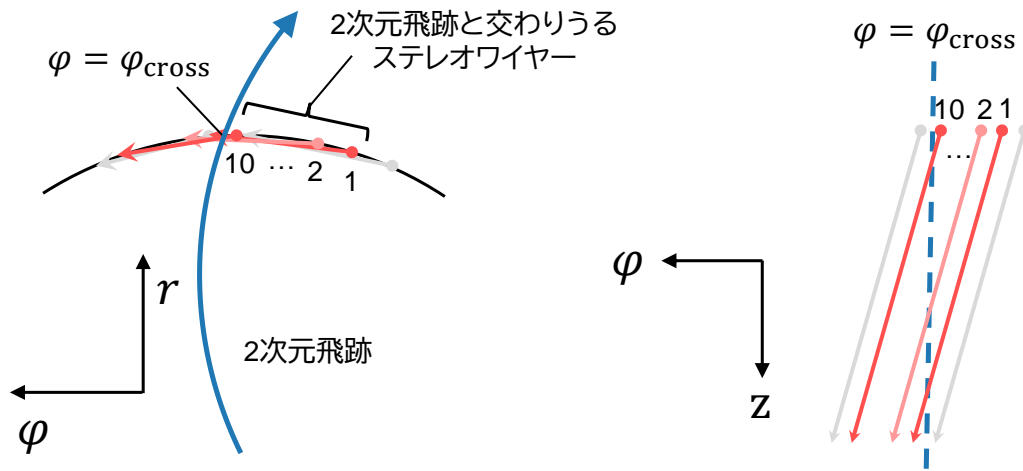


図 3.11  $x$ - $y$  平面において 2 次元飛跡と交わりうるステレオワイヤーは、図中赤もしくはピンクで示したように 1 レイヤーの中に最大 10 本存在する。(見やすさのために 2 色用いたが、意味的な違いはない。)

### 3.4.3 フィッティング

式 (3.1) の  $z$  成分の直線  $z = \cot \theta \cdot s + z_0$  を最小二乗法でフィッティングする。 $\chi^2$  を

$$\chi^2 = \sum_i \frac{[z_i - (\cot \theta \cdot s_i + z_0)]^2}{\sigma_i^2} \quad (3.11)$$

で定め、この値の最小化を考える。ここで、 $s_i$  と  $z_i$  は  $i$  番目のレイヤーにおける  $s$  と  $z$  を表し、定数  $\sigma_i$  はレイヤー  $i$  における  $z_i - (\cot \theta \cdot s_i + z_0)$  の分布の標準偏差で、ワイヤー固有の分解能を表す。 $z_0$  に関する停留条件より以下の式が成り立つ。

$$\begin{aligned} 0 &= \frac{\partial \chi^2}{\partial z_0} \\ &= \sum_i \frac{2 \cdot [z_i - (\cot \theta \cdot s_i + z_0)] \cdot (-1)}{\sigma_i^2} \\ &= (-2) \left[ \left( \sum_i \frac{z_i}{\sigma_i^2} \right) - \cot \theta \left( \sum_i \frac{s_i}{\sigma_i^2} \right) - z_0 \left( \sum_i \frac{1}{\sigma_i^2} \right) \right] \end{aligned} \quad (3.12)$$

また、 $\cot \theta$  に関する停留条件より以下の式が成り立つ。

$$\begin{aligned}
0 &= \frac{\partial \chi^2}{\partial \cot \theta} \\
&= \sum_i \frac{2 \cdot [z_i - (\cot \theta \cdot s_i + z_0)] \cdot (-s_i)}{\sigma_i^2} \\
&= (-2) \left[ \left( \sum_i \frac{z_i s_i}{\sigma_i^2} \right) - \cot \theta \left( \sum_i \frac{s_i^2}{\sigma_i^2} \right) - z_0 \left( \sum_i \frac{s_i}{\sigma_i^2} \right) \right]
\end{aligned} \tag{3.13}$$

式 (3.12) と式 (3.13) を連立して解くと、

$$z_0 = \frac{- \left( \sum_i \frac{s_i}{\sigma_i^2} \right) \left( \sum_i \frac{s_i z_i}{\sigma_i^2} \right) + \left( \sum_i \frac{s_i^2}{\sigma_i^2} \right) \left( \sum_i \frac{z_i}{\sigma_i^2} \right)}{\left( \sum_i \frac{1}{\sigma_i^2} \right) \left( \sum_i \frac{s_i^2}{\sigma_i^2} \right) - \left( \sum_i \frac{s_i}{\sigma_i^2} \right) \left( \sum_i \frac{s_i}{\sigma_i^2} \right)} \tag{3.14}$$

$$\cot \theta = \frac{\left( \sum_i \frac{1}{\sigma_i^2} \right) \left( \sum_i \frac{s_i z_i}{\sigma_i^2} \right) - \left( \sum_i \frac{s_i}{\sigma_i^2} \right) \left( \sum_i \frac{z_i}{\sigma_i^2} \right)}{\left( \sum_i \frac{1}{\sigma_i^2} \right) \left( \sum_i \frac{s_i^2}{\sigma_i^2} \right) - \left( \sum_i \frac{s_i}{\sigma_i^2} \right) \left( \sum_i \frac{s_i}{\sigma_i^2} \right)} \tag{3.15}$$

を得る。

### 3.4.4 旧アルゴリズムの問題点

旧アルゴリズムでは  $|z_0| < 30 \text{ cm}$  を満たす飛跡がトリガーされるため、 $z_0$  の位置分解能が悪いと本来排除したい  $|z_0|$  の大きな背景事象がトリガー条件を満たし、誤ってトリガーしてしまう。これにより、次節の最後 3.5.4 項で示すようにトリガーレートが上限値を超えてしまうことが予想されるため、新アルゴリズムでは、 $z_0$  の位置分解能を改善する必要がある。本項では、 $z_0$  の位置分解能の悪化に寄与している 3 つの問題点を指摘する。

#### (1) 使用するヒット点が少ない

3D Fitter の旧アルゴリズムでは 4 つのステレオ superlayer の中央ワイヤーのみを用いているため、フィッティングに使用するヒット点は最大 4 点である。これでは十分に精度の良いフィッティングが行えない。

#### (2) ノイズによるヒットを選択している

3.4.2 項で説明したように、旧アルゴリズムでは同レイヤーに複数のヒットがある場合  $z = 0$  に一番近いヒットを選択しているが、この選択方法ではノイズかそうでないかを判別することができず、図 3.12 のようにノイズを選択してしまうことがある。

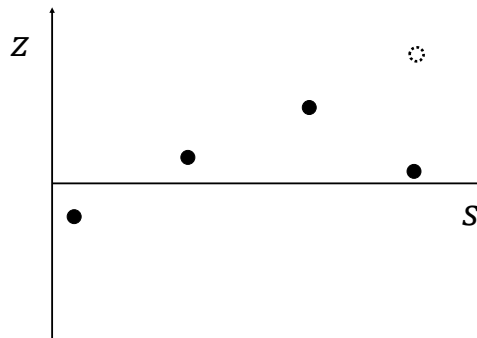


図 3.12 旧アルゴリズムでは同レイヤーに複数のヒットがある場合  $z = 0$  に一番近いヒットを選択するため、ノイズによるヒットを選択してしまうことがある。

### (3) ノイズの多いヒットパターンではドリフト補正が適用されない

3.4.1 項で述べたように、旧アルゴリズムでは track segment 内のヒットパターンごとに決められたドリフト方向を用いている。ヒットパターンとドリフト方向の対応関係はノイズを考慮していないモンテカルロシミュレーションに基づいて決めているため、ノイズを含むヒットパターンはシミュレーションでのサンプル数が少なくドリフト方向は Undetermined と定められているものが多い。表 3.2 に低ルミノシティ ( $2.3 \times 10^{34} \text{ cm}^{-2}\text{s}^{-1}$ ) のデータおよび高ルミノシティ ( $4.6 \times 10^{34} \text{ cm}^{-2}\text{s}^{-1}$ ) のデータにおけるドリフト方向 (Left/Right/Undetermined) の割合を示す。ノイズの多い高ルミノシティのデータではドリフト方向が Undetermined の割合が多く、ドリフト時間の情報が無視されていることがわかる。

表 3.2 低ルミノシティのデータと高ルミノシティのデータにおける track segment 内のヒットパターンから決めたドリフト方向 (Left/Right/Undetermined) の割合

| ドリフト方向           | 低ルミノシティ | 高ルミノシティ |
|------------------|---------|---------|
| Left (%)         | 23.7    | 14.7    |
| Right (%)        | 20.7    | 12.4    |
| Undetermined (%) | 55.6    | 72.9    |

## 3.5 トリガーレート

最後に、本研究の動機となるトリガーレートについて議論する。トリガーレートは単位時間あたりにトリガー信号を出す回数であり、3.1 節で述べたようにデータ収集システムの性能から上限と

して 30 kHz が要求されている。上限を超えてしまうと興味ある物理事象のデータを取りこぼしてしまい検出効率が落ちるため、興味ある物理事象以外の事象からくるトリガーレートを減らす必要がある。

本節ではトリガーレート増加の原因となっている beam background について説明したのち、トリガーレートの見積もりに必要となるイベント分類やトリガービットの定義を行い、最後に目標ルミノシティにおけるトリガーレートを見積もる。

### 3.5.1 Beam background

Beam background はビームが原因で引き起こされる背景事象の総称であり、物理事象由来のトリガーレートに対し約 3 倍のトリガーレートを記録している。Beam background にはいくつかの発生源が知られており、電子ビームと陽電子ビームのうち片方のビームのみでも生じる single beam background として、ビームが磁場で曲げられるときに出るシンクロトロン放射、ビームとガスの相互作用、バンチ内での粒子の散乱である Touschek 散乱などがあげられる [6]。それに対し、ビームの衝突によって生じる beam background として、radiative Bhabha 散乱や 2 光子過程を通じたレプトン対生成があげられる。Single beam background のは図 3.13 と図 3.14 の発生源分布に示されているように、広い  $z_0$  領域で起こりうる。

本研究では、 $z_0$  を決める 3D Fitter を改良することで、 $|z_0|$  の大きな single beam background を排除することを目標とする。

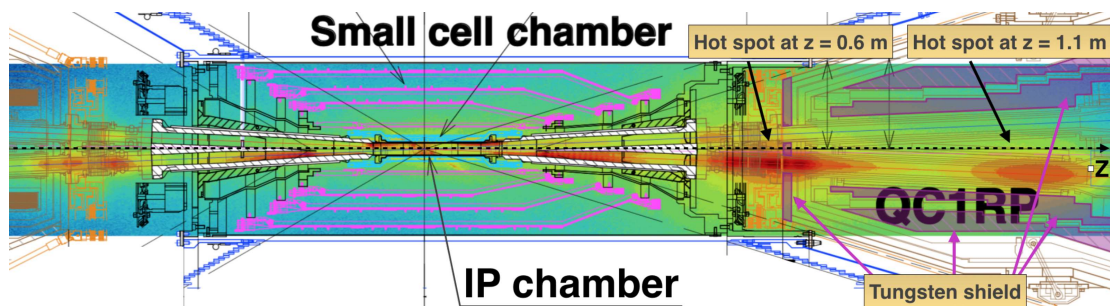


図 3.13 衝突点付近の検出器の構造と重ね合わせて表示した飛跡発生点の分布 [17]。飛跡発生点が多い部分が赤色で示されている。

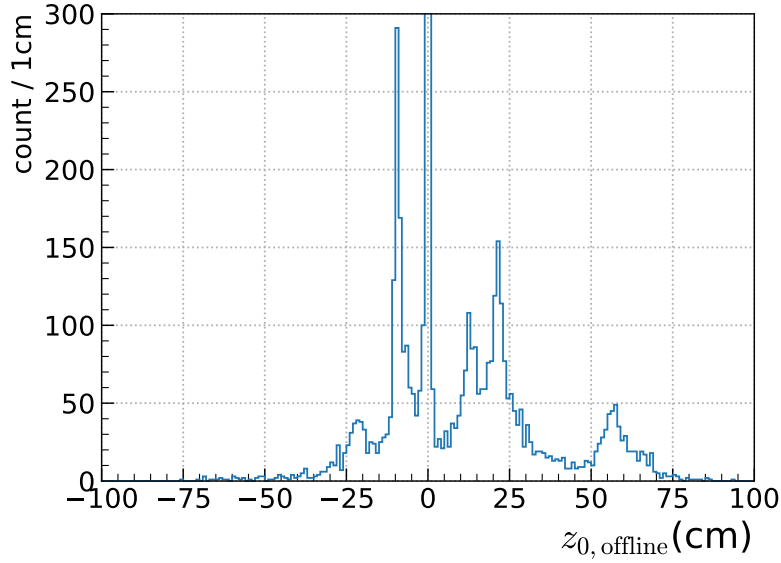


図 3.14  $z_0$  (=飛跡発生点の  $z$  座標) の分布。2022 年の実データをもとにソフトウェアでオフライン再構成を行った結果を示している。 $z_0 = 0$  付近の信号飛跡のピークのまわりに、single beam background によるピークがいくつか見えている（このピークが見やすいように縦軸を調整している）。

### 3.5.2 イベントの分類

Beam background による beam background event とビームの衝突による signal event、そして実際には飛跡が存在しない fake track event を分けるためにイベントの分類を行う。トリガーで再構成された飛跡（以下トリガー飛跡と呼ぶ）と、データ取得後にソフトウェアで再構成された飛跡（以下オフライン飛跡と呼ぶ）の情報を用いて、表 3.3 に示す条件で 3 つのイベントに分類する。なお、オフラインという用語はソフトウェアによる非リアルタイム処理のことを指す。

表 3.3 イベントの分類

| イベント分類                | 条件  |
|-----------------------|---|
| Signal event          | $ z_0  < 1$ のオフライン飛跡が存在する                   |
| Beam background event | オフライン飛跡は存在するが<br>$ z_0  < 1$ のオフライン飛跡は存在しない |
| Fake track event      | オフライン飛跡が存在しない<br>(トリガー飛跡のみ存在する)             |

### 3.5.3 トリガービット

Belle II 実験では、 $B$  中間子の物理や dark sector の物理など興味ある物理事象と使用する検出器とに応じて様々なトリガー条件が用意されており、トリガービットと呼ばれている。あるイベントが設定されたトリガービットの条件を満たすときにトリガーから検出器にトリガー信号が送られ、検出器からデータ収集システムにデータが送られる。

本研究では  $B$  中間子の物理に対する CDC トリガービットを表 3.4 のように定め、このトリガービットによるトリガーレートについて議論する。3次元飛跡の  $|z_0|$  に対するカットの値  $z_{\text{cut}}$  は任意に設定できるパラメータであり、3D Fitter の旧アルゴリズムでは 30 cm と設定していた。表中の Bhabha は Bhabha 散乱事象を指し、ECL トリガーから出される veto 信号によって排除される。

$B$  中間子の物理に対するトリガービットとして、粒子のエネルギー情報を用いる ECL トリガービット\*7も用いられる。2022 年に最高ルミノシティを記録した物理ランでは、 $B$  中間子の物理に対する CDC または ECL トリガービットのトリガーレートのうち、CDC によるものが 62% を占める。そこで、目標ルミノシティでもおおよそこの割合が保たれると仮定し、本研究では CDC トリガーレートの上限を 20 kHz と定める。

### 3.5.4 目標ルミノシティにおけるトリガーレート

2022 年に最高ルミノシティ  $4.6 \times 10^{34} \text{ cm}^{-2}\text{s}^{-1}$  を記録した物理ランでのトリガーレートを外挿し、目標ルミノシティ  $6.0 \times 10^{35} \text{ cm}^{-2}\text{s}^{-1}$  におけるトリガーレートを見積もる。トリガーレート

表 3.4 本研究で用いるトリガービットの定義

| トリガービット                       | 条件   |
|-------------------------------|--|
| 2-track ビット *5                | (2次元飛跡数 $\geq 2$ ) AND ( $ z_0  < z_{\text{cut}}$ の 3次元飛跡数 $\geq 1$ )<br>AND (2次元飛跡の角度差 $> 90^\circ$ のものが存在する)<br>AND (NOT Bhabha) |
| 3-track ビット *6                | (2次元飛跡数 $\geq 3$ ) AND ( $ z_0  < z_{\text{cut}}$ の 3次元飛跡数 $\geq 1$ )  |
| $B$ 中間子の物理に対する<br>CDC トリガービット | 2-track ビット OR 3-track ビット   |

\*5 Belle II 実験における fzo ビット

\*6 Belle II 実験における ffz ビット

\*7 Belle II 実験における c4 ビットまたは hie ビット

はルミノシティに比例すると仮定する。図 3.15 に 2022 年の 4 月から 6 月に行われた物理ランにおける、瞬間ルミノシティと 3-track ビットによるトリガーレートの関係を示す。なお 3D Fitter は 2022 年の物理ランでデータを取得していないため、以降の議論では Neurotrigger で取得したデータをもとに、シミュレーションにおける 3D Fitter と Neurotrigger の性能差を考慮して 3D Fitter のトリガーレートに換算した結果を示している。4 月はトリガーレートがルミノシティに対し線形に増加していくのが確認できるが、5 月から 6 月は、4 月に対してトリガーレートの増加が著しい。これは 5 月から 6 月にかけて、コリメータと呼ばれるビームの中心から離れた成分を削る装置に損傷を受けた影響だと考えられている [18]。2024 年からの物理ランでは background の対策をしたコリメータを用いる予定であり、トリガーレートの振る舞いも図中オレンジ点よりも青色の点に近づくと考えられる。青色の点をもとに外挿すると CDC トリガーレートは約 9 kHz となり上限値 20 kHz に収まるが、本研究では保守的な見積もりを行うため図中赤点の最高ルミノシティでの情報を用いる。なお参考までに、より厳しい見積もりとしてトリガーレートがルミノシティの 2 乗に比例する場合を考えると、トリガーレートは図 3.16 に示したように目標ルミノシティで 100 kHz のオーダーのトリガーレートとなる。この場合は本研究の結果では上限値以下に抑えることができないため、さらなる削減が必要である。

表 3.5 に、2022 年の物理ランで得られた最高ルミノシティにおけるトリガーレートと、目標ルミノシティに外挿したトリガーレートを示す。この表を見ると、目標ルミノシティにおいて beam

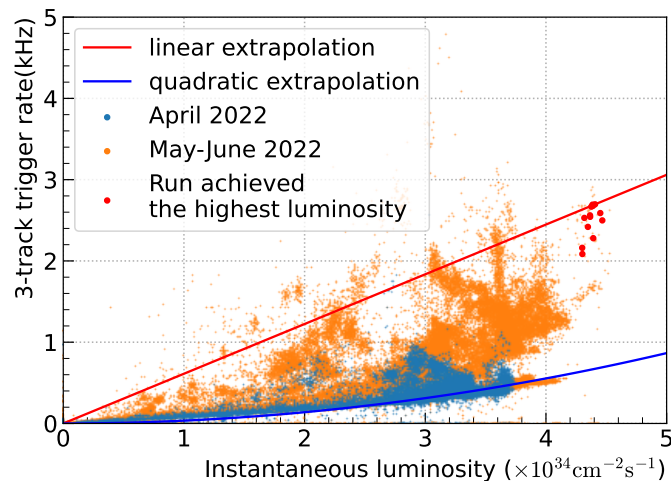


図 3.15 2022 年の 4 月から 6 月に行われた物理ランにおける、瞬間ルミノシティと 3-track ビットによるトリガーレートの関係。赤点で示した 2022 年の最高ルミノシティを記録した物理ランのトリガーレートをもとに、赤線のように線形に外挿し目標ルミノシティでのトリガーレートを見積もっている。また、青線は 4 月のデータを 2 次関数でフィットした結果である。散布図の一つの点は 1 分ごとのデータを示している。



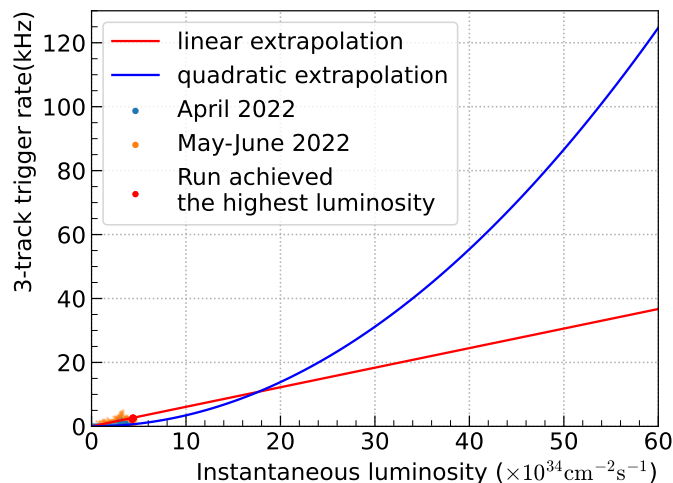


図 3.16 図 3.15 の横軸を広げたもの。本研究では赤線のように線形に外挿しているが、青線のように二次関数で外挿した場合は目標ルミノシティで 100 kHz のオーダーのトリガーレートとなる。

background event の占める割合が大きく、CDC トリガーレートが上限値 20 kHz を超えてしまうことがわかる。なお、Fake track event については、2022 年に行われたアップグレードによりほぼ抑えられることがモンテカルロデータを用いたシミュレーションにより示されているため 0 kHz と見積もっている [14]。新しい 3D Fitter の開発により 3.4.4 項で述べた問題点を解決することで、旧アルゴリズムで得られていた信号事象に対するトリガー効率を保ちつつ、beam background event のトリガーレートを 40% 以上削減し、CDC トリガーレートを 20 kHz 以下に抑えることを本研究の目標とする。

表 3.5 現在のルミノシティにおけるトリガーレートを目標ルミノシティに外挿した結果。現在のルミノシティは 2022 年の物理ランで得られた最高ルミノシティを指す。B 中間子の物理に対する CDC トリガービットを適用している。Fake track event のトリガーレートの見積もりはモンテカルロデータを用いたシミュレーションに基づく [14]。

| イベント分類                | 現在のルミノシティにおける<br>トリガーレート (kHz) | 目標ルミノシティにおける<br>トリガーレート (kHz) |
|-----------------------|--------------------------------|-------------------------------|
| Signal event          | 0.5                            | 6.9                           |
| Beam background event | 1.6                            | 21.3                          |
| Fake track event      | 1.0                            | 0                             |
| Total                 | 3.0                            | 28.2                          |

## 第 4 章

# 3 次元飛跡再構成の新アルゴリズム: Hough selection

この章では新たに開発した 3 次元飛跡再構成のための Hough selection アルゴリズムの概要を説明したあと、Hough selection アルゴリズムで用いられるパラメータの最適化と性能評価の結果を述べる。

### 4.1 Hough selection アルゴリズムの概要

Hough selection アルゴリズムでは 3.4.4 節で説明した、旧アルゴリズムの問題点を改善するために、主に 2 点を変更した。図 4.1 に Hough selection アルゴリズムの流れを示す。主な 2 点の変更点は、使用するヒット数を増やしたことと、ヒット選択の前段階に Hough voting とよばれる手法を追加したことである。以下では各変更点について説明する。

#### 4.1.1 使用するヒット数の増加

3.2 節で述べたトリガーボードのアップグレードによって、搭載される FPGA が Xilinx 社の Virtex-6 シリーズ [11] から Virtex Ultrascale シリーズ [12] にアップグレードされ、通信に用いるトランシーバーの 1 レーンあたりの通信速度が 5 Gbps から 25 Gbps に向上し、使用できるリソースの数が約 4 倍に増加した。これにより使用できるヒット数が増加し、図 4.2 に示すように旧アルゴリズムではステレオレイヤーにおける track segment 中央のワイヤーのみを使用していたが、Hough selection アルゴリズムでは track segment 内の全ワイヤーを使用した。これまで 3 次元飛跡の再構成のために全 20 レイヤーのうち 4 レイヤーしか使用できなかったが、この変更により全 20 レイヤーをフルに使用できるようになり、3.4.4 項で挙げた“使用するヒット点が少ない”

という問題が解決できる。

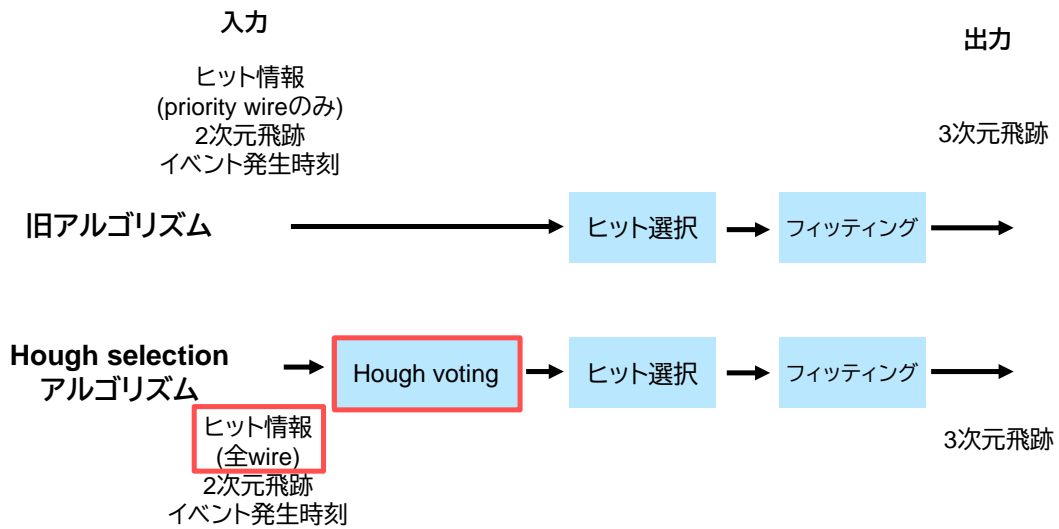


図 4.1 旧アルゴリズムと Hough selection アルゴリズムの処理の流れの違い。赤枠で囲った部分が主な変更点である。

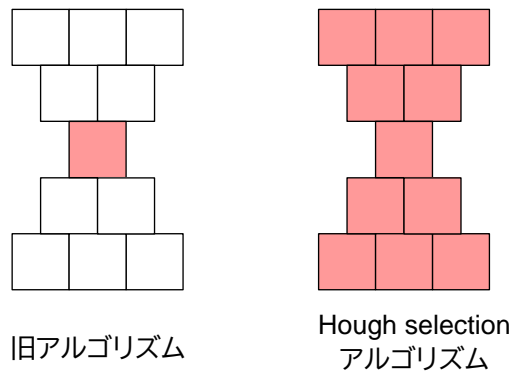


図 4.2 ステレオレイヤーにおける track segment の模式図。赤いセルが使用するワイヤーを表す。旧アルゴリズムでは中央の 1 つのワイヤーの情報のみ使用していたが、Hough selection アルゴリズムでは track segment 内の全 11 ワイヤーの情報を使用する。

## 4.1.2 Hough voting

ヒット選択のためのアルゴリズムとして、Hough voting<sup>\*1</sup>を採用した。Hough voting は当初、泡箱の写真分析のために Paul Hough によって開発された手法であり、現在では画像処理の分野において円や直線などのパターン検出に用いられている [19]。実空間の点を円や直線のパラメータ空間に曲線として投票することで、“円や直線を見つける”という複雑な問題を、“得票数のピークを見つける”というコンピュータや FPGA でも扱いやすい単純な問題に変換できることが特徴である。

3.4 節でも述べたように、らせん上の任意のヒット点において、ヒット点の  $z$  座標と、原点からヒット点までの 2 次元飛跡上の弧長  $s$  との間に

$$z = \cot \theta \cdot s + z_0 \quad (4.1)$$

という関係が成り立つ。すなわち、粒子飛跡によるヒット点を  $s$ - $z$  平面上にプロットすると、図 4.3 のように  $z_0$  を切片  $\cot \theta$  を傾きとする直線をなす。この直線を検出するために Hough voting を用いる。本研究で用いるパラメータ空間は  $\cot \theta$  と  $z_0$  からなる 2 次元空間であり、以降  $\cot \theta$ - $z_0$  平面と呼ぶ。式 (4.1) を移項すると、

$$z_0 = -s \cdot \cot \theta + z \quad (4.2)$$

を得るが、この式は図 4.4 に示すように“ある一つのヒット点  $(z, s)$  を通るような直線の切片  $z_0$  と傾き  $\cot \theta$  の組全体は、 $\cot \theta$ - $z_0$  平面において傾き  $s$ 、切片  $z$  の直線をなす”ことを意味する。 $s$ - $z$

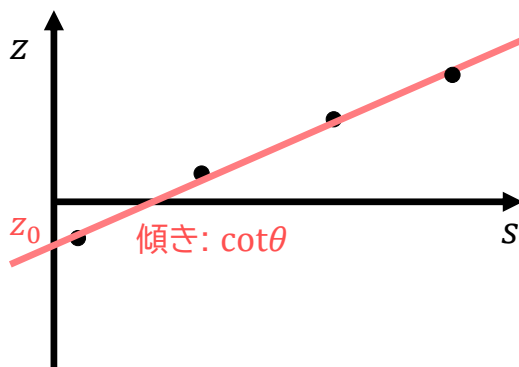


図 4.3 ヒット点が  $s$ - $z$  平面上になす直線。  $s$  は 2 次元飛跡上の原点からヒット点までの弧長、  $z$  はヒット点の  $z$  座標を表す。この直線の傾き  $\cot \theta$  と切片  $z_0$  を求めることが目標である。

<sup>\*1</sup> Hough 変換とも呼ばれる。本研究ではパラメータの変換の後、セルへの投票を行うことを強調するために Hough voting と呼ぶ。

平面にあるヒット点の数だけこの変換を行うと、図 4.5 のようにヒット点の数だけ  $\cot\theta-z_0$  平面上に直線が投票され、最も重なった点が求めたい 3 次元飛跡のパラメータ  $z_0$  と  $\cot\theta$  となる。

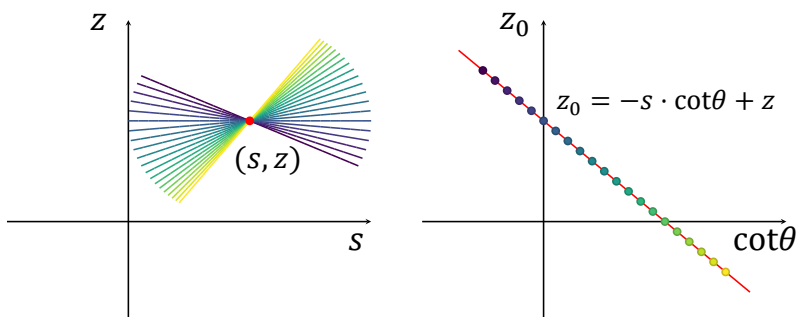


図 4.4  $s$ - $z$  平面上の 1 点が  $\cot\theta$ - $z_0$  平面上に描く直線 [13]

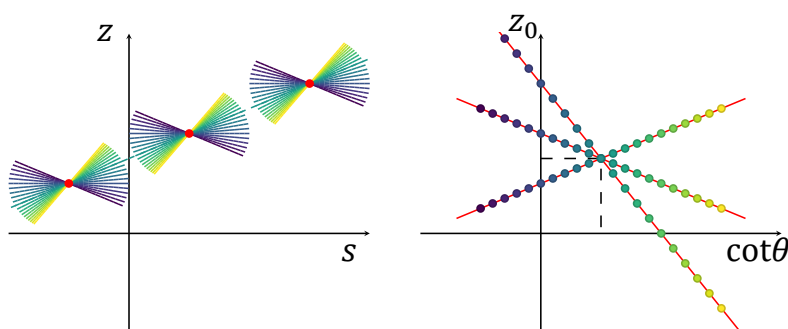


図 4.5  $s$ - $z$  平面上の複数点が  $\cot\theta$ - $z_0$  平面上に描く直線 [13]

### 耐ノイズ性

Hough voting のメリットとして、多少のノイズが混入しても結果に影響しない耐ノイズ性が挙げられる。ノイズとは、別の飛跡によるヒットや CDC 読み出し回路におけるクロストークによるヒットなど、入力された 2 次元飛跡と紐づいていないヒットを指す。図 4.6 に示すように、ノイズによるヒットは  $\cot\theta$ - $z_0$  平面においてピークとは離れた場所に投票されるため、本来の結果に影響を与えない。ピークに寄与したヒットのみを選ぶことで、3.4.4 項で挙げた“ノイズによるヒットを選択してしまう”という問題が解決できる。

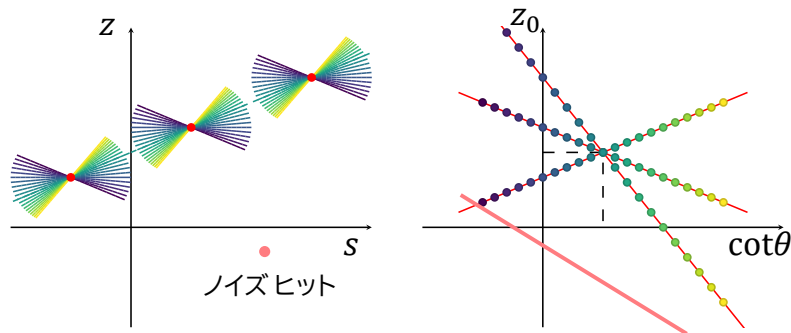


図 4.6 Hough voting ではノイズによるヒットが  $\cot \theta - z_0$  平面においてピークとは離れた場所に投票されるため耐ノイズ性がある。ピークに寄与するヒットを選ぶことによりノイズによるヒットを選択してしまうことを避けられる。

### ドリフト補正

3.4 節で述べたように、ドリフト時間による補正でより精度良い再構成が可能である。Hough selection アルゴリズムでは、ヒットパターンによってドリフト方向を決めるのを廃止し、左右どちらのドリフト方向にも投票することにした。この方法が有効なのは、耐ノイズ性の項で説明したように間違った方向への投票はピークに影響しないからである。これによりドリフト時間の情報を失うことなく利用でき、3.4.4 項で述べた“ノイズの多いヒットパターンに対しドリフト補正が適用されない”という問題が解決できる。

### セルへの投票

$\cot \theta - z_0$  平面への直線の投票は、具体的には図 4.7 のように  $\cot \theta - z_0$  平面を格子状のセルに分割し、直線がセルを通過している場合そのセルの得票数に 1 を加えるという操作を行う。 $z_0$  方向のセル数  $N_{\text{cell},z_0}$  と  $\cot \theta$  方向のセル数  $N_{\text{cell},\cot \theta}$  は後に最適化するパラメータである。同じレイヤーからの投票は何票あっても 1 票として取り扱っているため、ステレオレイヤー数の 20 が最大得票数となる。

$\cot \theta$  の範囲は  $-0.8 < \cot \theta < 1.5$  と設定した。これは図 4.8 に示すように CDC のバレル部を通り抜ける飛跡をカバーするように定められている。 $z_0$  の範囲は図 3.14 のオフライン飛跡の  $z_0$  分布に基づき、 $-50 \text{ cm} < z_0 < 78 \text{ cm}$  と設定した。78 cm に設定したのは、範囲が 2 の累乗である 128 cm となりファームウェアへの実装上都合がよいからである。

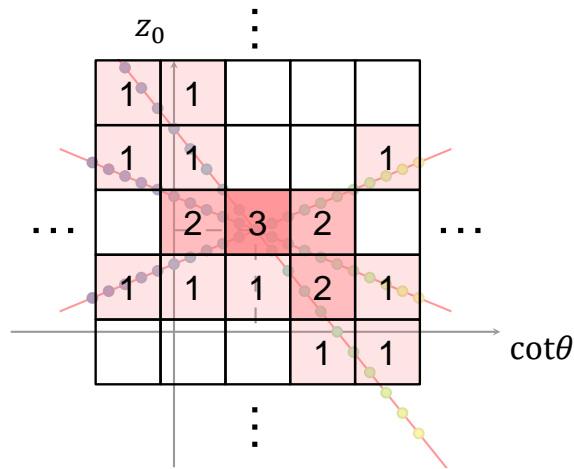


図 4.7  $\cot \theta - z_0$  平面のセルに直線が投票される様子

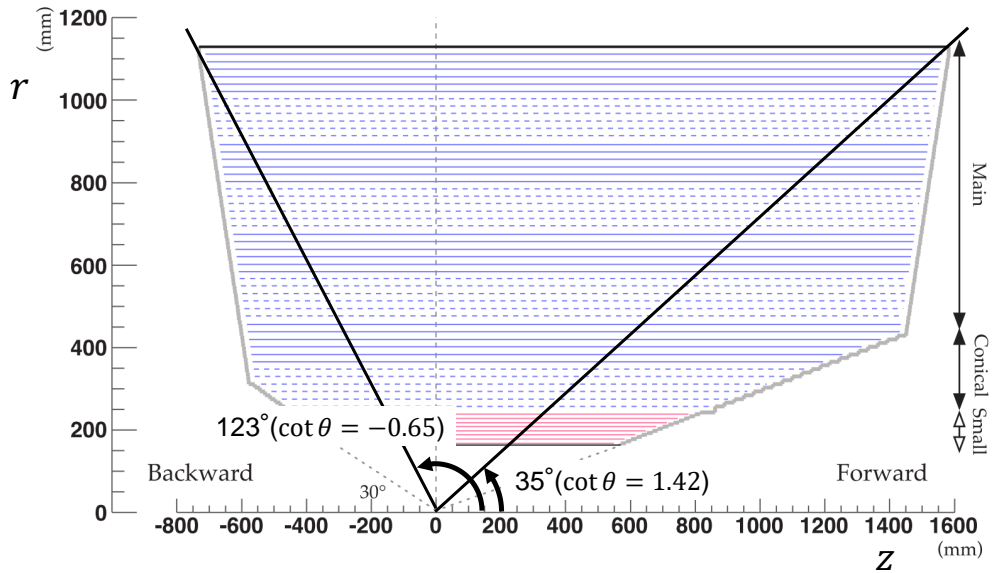


図 4.8 CDC のバレル部を通り抜ける飛跡の角度範囲

### 4.1.3 ヒット選択

図 4.9 に示すように、Hough voting の結果を受けて本来のヒットを選択することで、3.4.4 項で挙げた“ノイズによるヒットを選択してしまう”という問題が解決できる。選び方は、ヒット点と Hough voting で得られた直線との  $z$  方向の距離  $\Delta z := |z_{\text{hit}} - z(s)|$  が  $\Delta z < z_{\text{selection}}$  を満たすヒットを選ぶ。 $z_{\text{selection}}$  は後で最適化するパラメータである。同レイヤーに複数の点がある場合は  $\Delta z$  が最小のヒットを選択し、1 レイヤーにつき 1 点、合計最大 20 点を選択する。

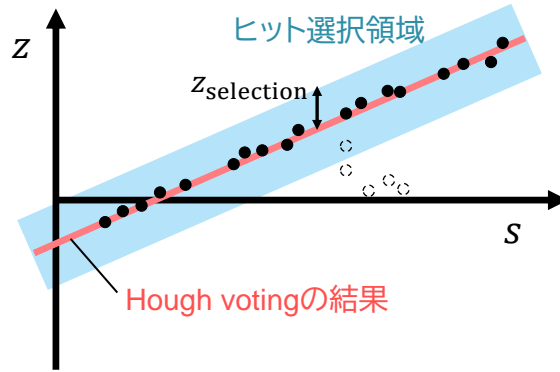


図 4.9 図中赤線で示した Hough voting の結果に対し、 $\Delta z := |z_{\text{hit}} - z(s)|$  が  $\Delta z < z_{\text{selection}}$  を満たす図中青色の領域のヒットを選択する。同レイヤーに複数の点がある場合は  $\Delta z$  が最小のヒットを選択する。

#### 4.1.4 フィッティング

式 (3.14) と式 (3.15) で求めたように、 $z_0$  と  $\cot \theta$  は

$$z_0 = \frac{-\left(\sum_i \frac{s_i}{\sigma_i^2}\right) \left(\sum_i \frac{s_i z_i}{\sigma_i^2}\right) + \left(\sum_i \frac{s_i^2}{\sigma_i^2}\right) \left(\sum_i \frac{z_i}{\sigma_i^2}\right)}{\left(\sum_i \frac{1}{\sigma_i^2}\right) \left(\sum_i \frac{s_i^2}{\sigma_i^2}\right) - \left(\sum_i \frac{s_i}{\sigma_i^2}\right) \left(\sum_i \frac{s_i}{\sigma_i^2}\right)} \quad (4.3)$$

$$\cot \theta = \frac{\left(\sum_i \frac{1}{\sigma_i^2}\right) \left(\sum_i \frac{s_i z_i}{\sigma_i^2}\right) - \left(\sum_i \frac{s_i}{\sigma_i^2}\right) \left(\sum_i \frac{z_i}{\sigma_i^2}\right)}{\left(\sum_i \frac{1}{\sigma_i^2}\right) \left(\sum_i \frac{s_i^2}{\sigma_i^2}\right) - \left(\sum_i \frac{s_i}{\sigma_i^2}\right) \left(\sum_i \frac{s_i}{\sigma_i^2}\right)} \quad (4.4)$$

で与えられる。なお、ファームウェア実装を簡単にするために、各ワイヤーの分解能に相当するパラメータ  $\sigma_i$  は、すべての  $i$  で  $\sigma_i = 1$  とした。

## 4.2 解析手法

本節では、次節のパラメータ最適化と次々節の性能評価に用いた解析手法について説明する。図 4.10 に示すように、実データに対しトリガー再構成のシミュレーションとオフライン再構成を行った。各方法で再構成された飛跡をそれぞれトリガー飛跡、オフライン飛跡と呼び、両者をマッチングして比較することで性能評価を行った。



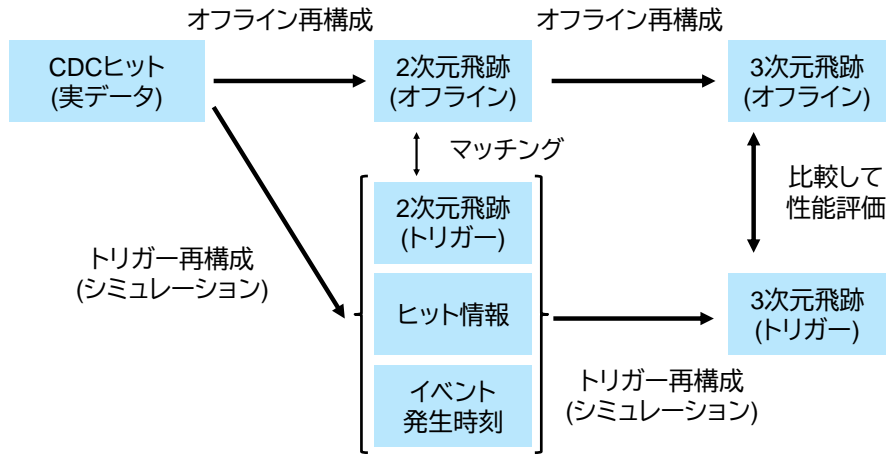


図 4.10 ソフトウェアシミュレーションによる飛跡再構成の流れ

#### 4.2.1 使用ソフトウェア

Belle II の解析ソフトウェア (basf2 [20] release-06-00-14) を用いて各トリガーモジュールをシミュレートし、トリガー飛跡を再構成した。また、オフライン再構成はトリガーによる再構成に比べ十分精度が良いため、性能評価の際の比較対象としてオフライン飛跡を求めた。

#### 4.2.2 使用データ

使用したデータは、2022 年の物理ランで取得した CDC から読み出したデータである。表 4.1 に使用したデータのプロパティをまとめる。背景事象を多く含むデータを使用するため、オフライン再構成によってトリガーをかける High Level Trigger を使用していないデータを選択した。このデータを取得する際は既存の CDC トリガーが使われているため、取得された背景事象は実際に起こった背景事象よりも少ないというバイアスが生じる。後述する性能指標である efficiency や background rejection rate はその影響を受けるが、最終的にトリガーレートを求める際には式 (4.12) のように新旧アルゴリズムにおける性能指標の比を用いて求めるので、そのバイアスはキャンセルされる。

表 4.1 使用したデータのプロパティ

| 項目                 | 値   |
|--------------------|---|
| データ取得日時            | 2022 年 6 月 18 日 10 時 36 分                         |
| データ取得時間            | 14 分 36 秒   |
| High Level Trigger | disabled  |
| 最大ルミノシティ           | $2.3 \times 10^{34} \text{ cm}^{-2}\text{s}^{-1}$ |
| 最大トリガーレート          | 4.04 kHz  |
| 最適化に使用したイベント数      | 10000 イベント  |
| 性能評価に使用したイベント数     | 23403 イベント  |

### 4.2.3 トリガー飛跡とオフライン飛跡のマッチング

一般にトリガー飛跡の数とオフライン飛跡の数は異なるため、常にトリガー飛跡に対し 1 つのオフライン飛跡がマッチするとは限らない。そこで、トリガー飛跡からオフライン飛跡へのマッチングと、オフライン飛跡からトリガー飛跡へのマッチングを定義し、それらの組み合わせで飛跡を分類する。図 4.11 にマッチングの流れを示す。

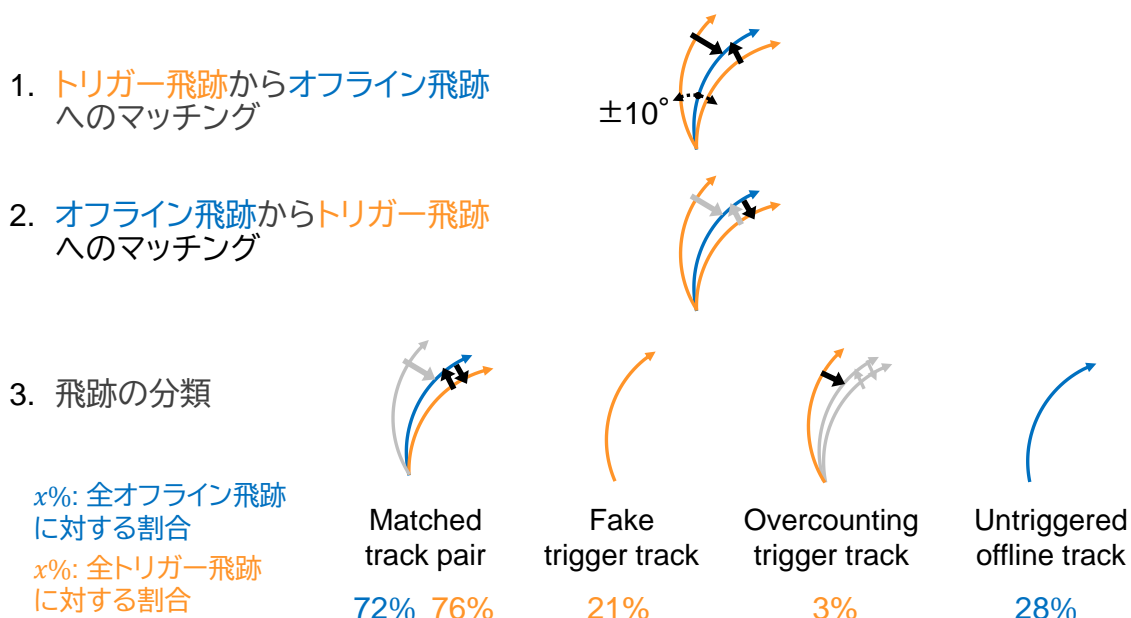


図 4.11 トリガー飛跡とオフライン飛跡のマッチングの流れ、および飛跡の分類割合

### (1) トリガー飛跡からオフライン飛跡へのマッチング

各トリガー飛跡に対し、2次元飛跡の角度差  $\Delta\varphi$  が最も小さいオフライン飛跡へとマッチングを行う。ただし  $\Delta\varphi < 10^\circ$  を満たす必要がある。複数のトリガー飛跡から同じオフライン飛跡へのマッチングが存在する可能性があることに注意する。

### (2) オフライン飛跡からトリガー飛跡へのマッチング

各オフライン飛跡に対し、自身へのマッチングが存在するトリガー飛跡の中から  $\Delta\varphi$  が最も小さいトリガー飛跡へとマッチングを行う。

### (3) 飛跡の分類

上で定めたマッチングの組み合わせによって、すべてのオフライン飛跡とトリガー飛跡は以下の4つのいずれかに分類される。使用データにおける各飛跡分類の割合を図 4.11 に示した。

#### Matched track pair

双方向のマッチングが存在するトリガー飛跡とオフライン飛跡のペア。

#### Fake trigger track

どのオフライン飛跡へのマッチングも存在しないトリガー飛跡。

#### Overcounting trigger track

あるオフライン飛跡へのマッチングが存在するが、そのオフライン飛跡から自身へのマッチングが存在しないトリガー飛跡。

#### Untriggered offline track

どのトリガー飛跡からのマッチングも存在しないオフライン飛跡。

## 4.2.4 飛跡の選別

オフライン飛跡に対し以下の条件を設け、満たさない場合はそのオフライン飛跡が属する matched track pair ごと排除した。

1. matched track pair である
2. 横方向運動量  $p_T$  が  $0.3 \text{ GeV}/c$  以上
3. CDC の内側から 5 レイヤー以内および外側から 5 レイヤー以内にヒットが存在する

Matched track pair のみを使用したのは、fake trigger track、overcounting trigger track や untriggered offline track が生じるのは 2 次元飛跡再構成における問題であり、今回はその影響を排除して 3D Fitter の性能を評価するためである。また、CDC トリガーのアルゴリズムは CDC 内で回転するような横方向運動量  $p_T$  が小さい飛跡や、十分な数のヒットを残さない短い飛跡を再構成するようデザインされていないので、そのような飛跡も排除した。

## 4.2.5 性能指標

オフライン飛跡に対し、信号飛跡と背景飛跡を以下のように定義する。

$$\text{信号飛跡} := |z_0| < 1 \text{ cm を満たすオフライン飛跡} \quad (4.5)$$

$$\text{背景飛跡} := |z_0| > 1 \text{ cm を満たすオフライン飛跡} \quad (4.6)$$

また、トリガー飛跡に対するトリガー条件を以下で定義する。

$$|z_0| < z_{\text{cut}} \quad (4.7)$$

ここで、カットの値  $z_{\text{cut}}$  はフリーパラメータである。

性能指標として、3D track reconstruction efficiency と background rejection rate 以下のように定義する。

$$\text{3D track reconstruction efficiency} := \frac{\text{トリガーされた信号飛跡の数}}{\text{信号飛跡の数}} \quad (4.8)$$

$$\text{Background rejection rate} := \frac{\text{トリガーされなかった背景飛跡の数}}{\text{背景飛跡の数}} \quad (4.9)$$

以降、3D track reconstruction efficiency を単に efficiency と呼ぶ。

## 4.3 ピーク検出手法およびパラメータの最適化

本節では、3D Fitter に用いられたピーク検出手法およびパラメータの最適化について説明する。

### 4.3.1 各ピーク検出手法の概要

セル上の得票数のピークを検出する方法として、1.maximum、2.clustering の 2 種類を試した。

#### (1) maximum

単に最も得票数の多いセルを選ぶ。セルの中央の値を結果とする。最も得票数の多いセルが複数あった場合は、信号飛跡を優先するため  $z_0 = 0$  に一番近いセルを選ぶ。図 4.12 に具体例を示す。

## (2) clustering

得票数に対するしきい値  $N_{\text{threshold}}$  以上の得票数を持つセルをピーク候補として定め、クラスタリングを行う。 $N_{\text{threshold}}$  は後に最適化するパラメータである。ここで投票される直線である式 (4.2) を考えると、2次元飛跡上の弧長  $s$  は常に正なので、傾き  $-s$  は常に負である。このことを踏まえ、候補のセル同士が上下左右に接する場合、および左上と右下の関係にある場合に同一のクラスターに含めた。最も得票数の多いクラスターを選び、得票数を重みとした重心の値を結果とした。最も得票数の多いクラスターが複数あった場合は maximum と同様の理由で  $z_0 = 0$  に近いクラスターを選ぶ。図 4.13 に具体例を示す。

### 4.3.2 最適化するパラメータ

Efficiency が 90% になる  $z_{\text{cut}}$  を設定したときの background rejection rate を目的関数と定め最大化を行った。最適化するパラメータを表 4.2 にまとめた。パラメータの最適化はグリッドサーチによる全探索を行った。すべてのパラメータで一度にグリッドサーチ行くと探索すべきパラメータの組が増えてしまうため、セル数と得票数のしきい値のような、強い相関があると考えられる変数はまとめて最適化し、その他の変数は一つずつ逐次的に最適化を行った。この時、まだ定まっていないパラメータの値は、表 4.2 に示した初期値を暫定的に用いた。

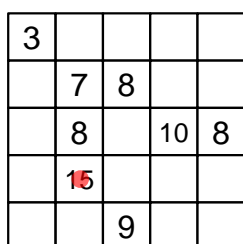


図 4.12 maximum: 最も得票数の多いセル (図の例では得票数 15 のセル) を選び、セルの中央を結果とする。

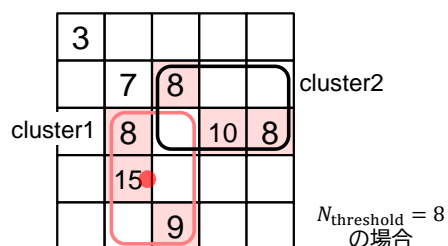


図 4.13 clustering: 設定したしきい値  $N_{\text{threshold}}$  以上の得票数を持つセルに対し clustering を行い、最も得票数の多いクラスター (図の例では得票数 32 の cluster1) の、得票数を重みとした重心の位置 (図中の赤いマーカー) を結果とする。

表 4.2 最適化するパラメータの一覧

| 最適化の順番 | パラメータ名   | 試した値                 | 刻み幅 | 初期値 |
|--------|--|----------------------|-----|-----|
| 1      | $z_0$ 方向のセル数 $N_{\text{cell},z_0}$                 | [10, 50]             | 5   | -   |
|        | $\cot \theta$ 方向のセル数 $N_{\text{cell},\cot \theta}$ | [10, 50]             | 5   | -   |
|        | 得票数のしきい値 $N_{\text{threshold}}$                    | [5, 15]              | 1   | -   |
|        | ピーク検出手法  | {maximum,clustering} | -   | -   |
| 2      | ヒット選択範囲 $z_{\text{selection}}$ [cm]                | [5, 50]              | 5   | 20  |

### 4.3.3 最適化の結果

最適化の結果およびファームウェア実装の都合のよさを考慮し、表 4.3 に示すパラメータに決定した。次節の性能評価はこのパラメータ設定に基づいて行われる。最適化の詳細は付録 A に示す。

表 4.3 パラメータの最適化結果

| パラメータ名                              | 値       |
|-------------------------------------|---------|
| $z_0$ 方向のセル数                        | 16      |
| $\cot \theta$ 方向のセル数                | 16      |
| ピーク検出手法                             | maximum |
| ヒット選択範囲 $z_{\text{selection}}$ [cm] | 25      |

## 4.4 性能評価

図 4.14 に新旧アルゴリズムにおける efficiency と background rejection rate の 2 次元プロットを示す。以下では efficiency や background rejection rate に対する統計誤差として、Clopper-Pearson の 68.3 % 信頼区間を表示している。Clopper-Pearson の信頼区間は正規分布への近似を使用しない、二項検定に基づく信頼区間である [21]。比較のために、efficiency=86% (2022 年の物理ランで設定されていた  $z_{\text{cut}} = 30$  cm での旧アルゴリズムの efficiency) における background rejection rate を見ると、新アルゴリズムでは旧アルゴリズムに対し 144% 多くの背景飛跡を棄却できることがわかる。

図 4.15 に新旧アルゴリズムにおける横運動量  $p_T$  ごとの efficiency を示す。全  $p_T$  領域で

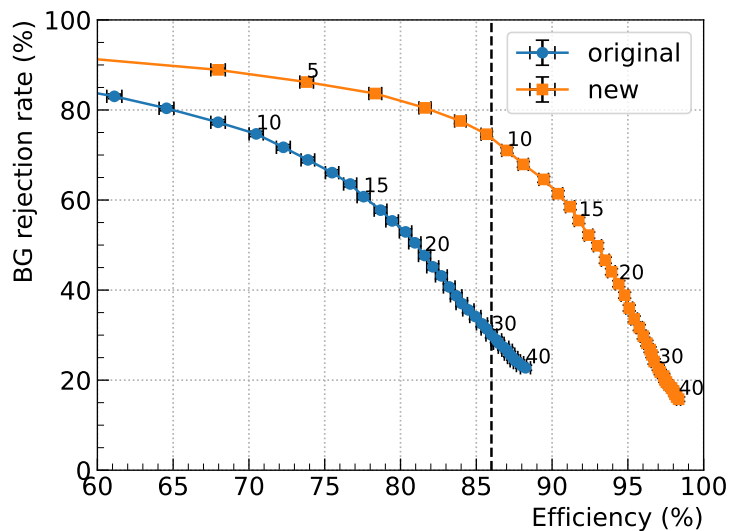


図 4.14 新旧アルゴリズムにおける 3D Fitter の efficiency と background rejection rate の 2 次元プロット。点上の数値  $z_{\text{cut}}$  (cm) は  $|z_0| < z_{\text{cut}}$  を満たす飛跡をトリガーしていることを表す。図中の破線は、 $z_{\text{cut}} = 30$  cm (2022 年の物理ランでの設定値) での旧アルゴリズムの efficiency=86% を示している。統計誤差として、Clopper-Pearson の 68.3 % 信頼区間を表示している。

の efficiency が 86 % となるように、旧アルゴリズムでは  $z_{\text{cut}} = 30$  cm、新アルゴリズムでは  $z_{\text{cut}} = 9.3$  cm と選んでいる。新旧アルゴリズムとも  $p_T$  が小さい領域では efficiency が低く、 $p_T$  が大きくなるに従い efficiency が高くなる傾向がみられる。

Background rejection rate が向上した理由は、図 4.16 に示した  $\Delta z_0$  ( $:= z_{0,\text{trigger}} - z_{0,\text{offline}}$ ) 分布のピークが細くなっていることからわかるように、Hough selection アルゴリズムによってより精度の良い  $z_0$  の再構成ができるようになったからである。

図 4.17 と図 4.18 に新旧アルゴリズムにおける各  $z_{0,\text{offline}}$  領域ごとの  $\Delta z_0$  の分布を示す。特に  $|z_0| < 1$  の領域のピークが鋭くなり、カット  $z_{\text{cut}}$  を狭めても efficiency が保てることがわかる。

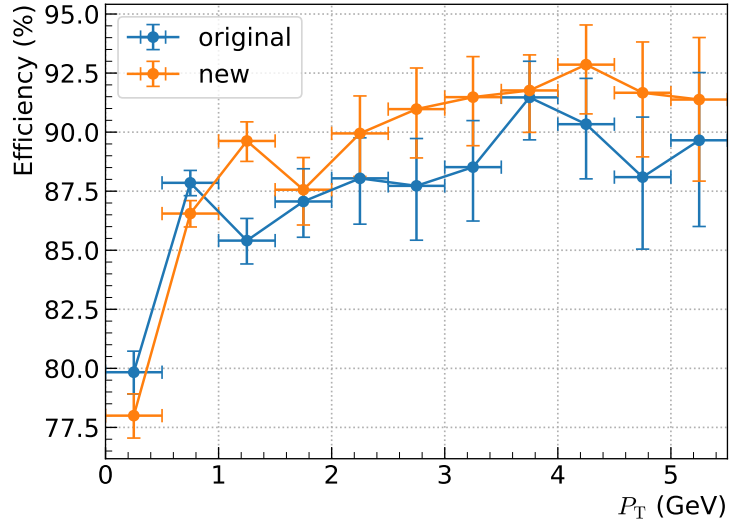


図 4.15 新旧アルゴリズムにおける横運動量  $p_T$  ごとの efficiency。全  $p_T$  領域での efficiency が 86 % となるように、旧アルゴリズムでは  $z_{cut} = 30$  cm、新アルゴリズムでは  $z_{cut} = 9.3$  cm と選んでいる。統計誤差として、Clopper-Pearson の 68.3 % 信頼区間を表示している。

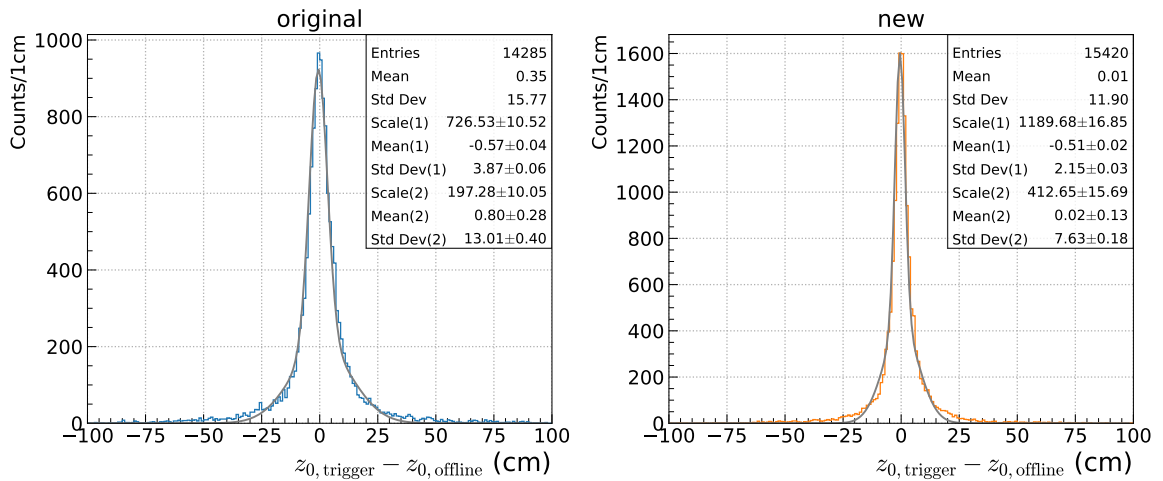


図 4.16 新旧アルゴリズムにおける  $\Delta z_0$  ( $:= z_{0,trigger} - z_{0,offline}$ ) の分布。灰色の線は double gaussian によるフィット結果を表す。



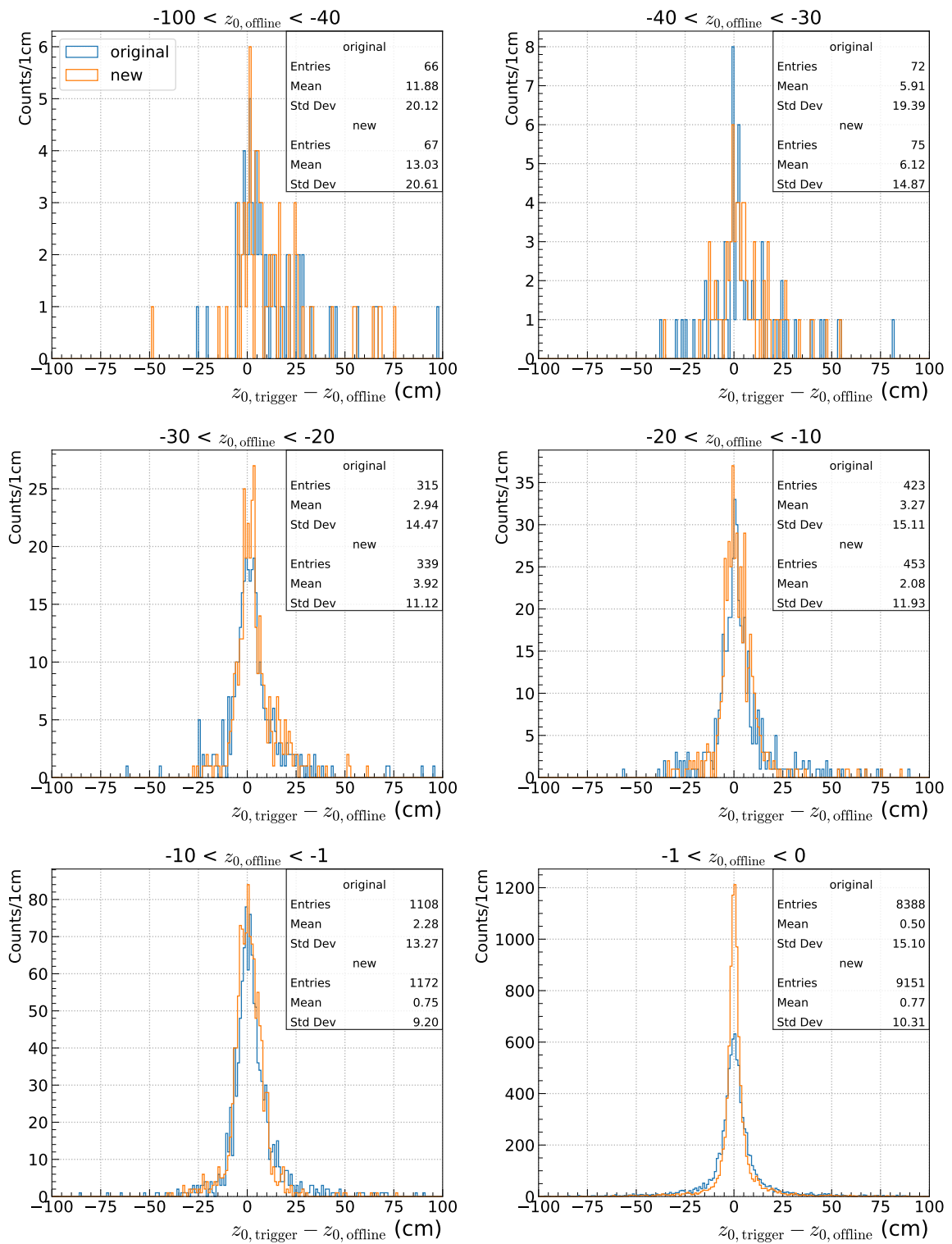


図 4.17 新旧アルゴリズムにおける各  $z_{0,offline}$  領域 ( $z_{0,offline} < 0$ ) ごとの  $\Delta z_0$  ( $:= z_{0,trigger} - z_{0,offline}$ ) の分布

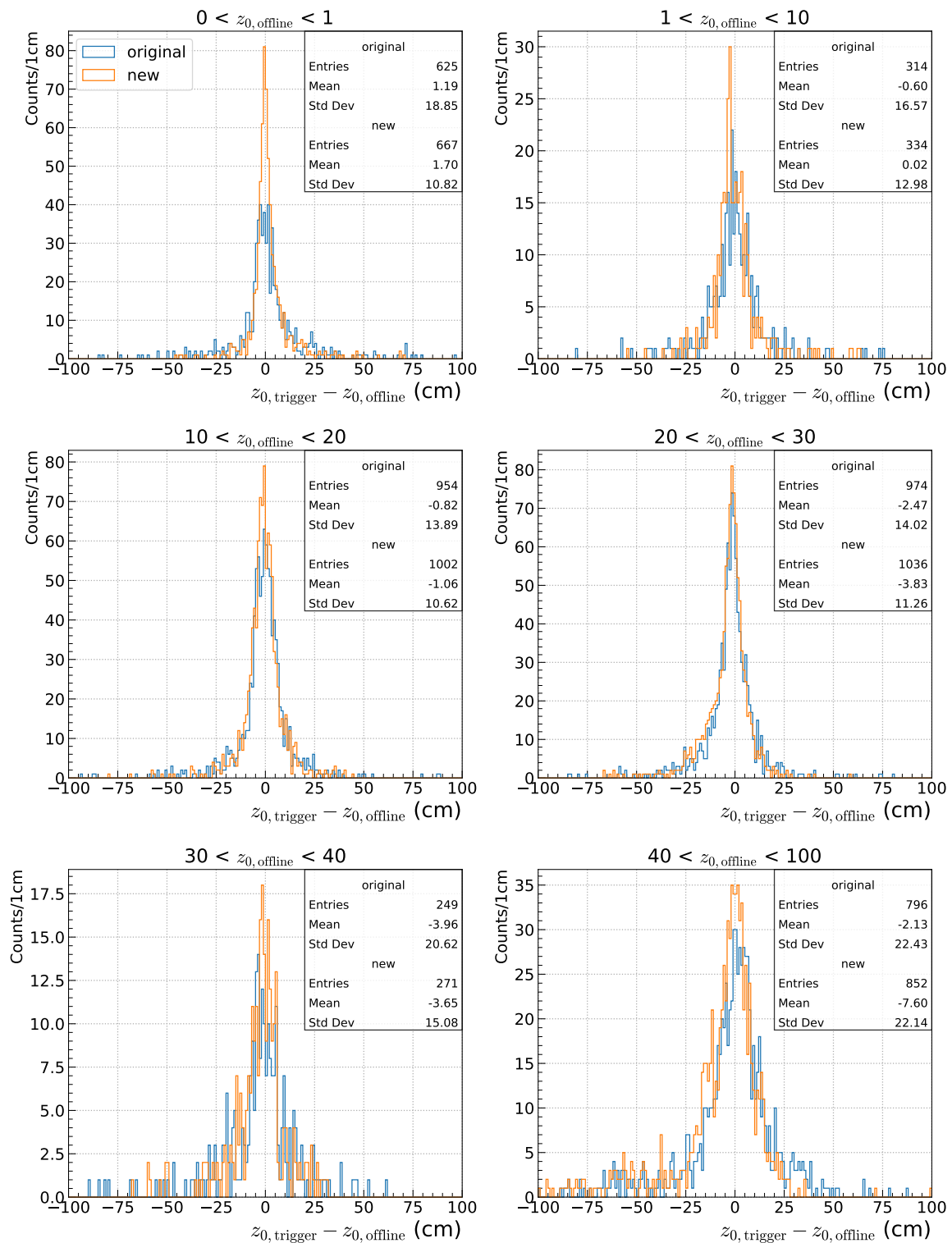


図 4.18 新旧アルゴリズムにおける各  $z_{0, \text{offline}}$  領域 ( $z_{0, \text{offline}} > 0$ ) ごとの  $\Delta z_0$  ( $:= z_{0, \text{trigger}} - z_{0, \text{offline}}$ ) の分布

## 4.5 トリガーレートの見積もり

4.2.5 項で定義してこれまで議論していた efficiency と background rejection rate は、飛跡の数に基づく track-level の性能指標であり、トリガーレートを議論するときには 1 イベント内の飛跡数の分布を考慮した event-level の性能指標に換算する必要がある。換算の結果 event-level の性能指標が track-level の性能指標からどう変化するのか定性的に考えると、表 3.4 で示したトリガービット条件からわかるように efficiency に関してはイベント内のどれか 1 つの信号飛跡をトリガーすればよいので条件が緩くなり、background rejection rate に関してはイベント内のすべての背景飛跡を排除しなければならないので条件がきつくなる。定量的には、event-level の efficiency  $E_{\text{event}}$  と background rejection rate  $R_{\text{event}}$  は、track-level の指標  $E_{\text{track}}$ 、 $R_{\text{track}}$  を用いて

$$1 - E_{\text{event}} = \sum_{n=1} r_n^{\text{signal}} \cdot (1 - E_{\text{track}})^n \quad (4.10)$$

$$R_{\text{event}} = \sum_{n=1} r_n^{\text{BG}} \cdot R_{\text{track}}^n \quad (4.11)$$

と表せる。ここで  $r_n^{\text{signal}}$  は signal event 全体における  $n$  本の信号飛跡を持つ signal event の割合、 $r_n^{\text{BG}}$  は beam background event 全体における  $n$  本の背景飛跡を持つ beam background event の割合であり、図 4.19 と図 4.20 に示した飛跡数分布に基づく。なおこの飛跡数分布は、トリガーレートの外挿点に用いた 2022 年度の最高ルミノシティの物理ランのデータを用いている。

表 4.4 に新旧アルゴリズムにおける換算前後の efficiency と background rejection rate を示す。旧アルゴリズムではトリガー条件のカットは 2022 年の物理ランで設定されていた  $z_{\text{cut}} = 30 \text{ cm}$  で評価し、新アルゴリズムでは旧アルゴリズムで得られていた efficiency=86% に対応する  $z_{\text{cut}} =$

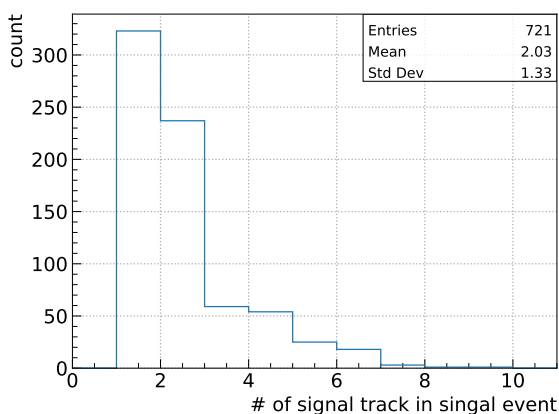


図 4.19 Signal event における信号飛跡数の分布

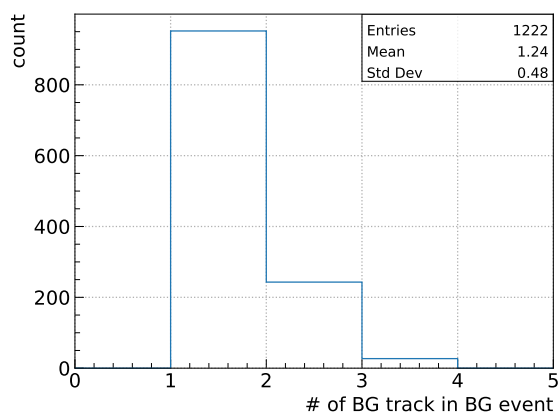


図 4.20 Beam background event における背景飛跡数の分布

9.3 cm で評価している。

Event-level の指標を用いると、新アルゴリズムの signal event と beam background event のトリガーレート  $T_{\text{new}}^{\text{signal}}$ 、 $T_{\text{new}}^{\text{BG}}$  は、旧アルゴリズムのトリガーレート  $T_{\text{original}}^{\text{signal}}$ 、 $T_{\text{original}}^{\text{BG}}$  を用いて

$$T_{\text{new}}^{\text{signal}} = \frac{E_{\text{event,new}}}{E_{\text{event,original}}} T_{\text{original}}^{\text{signal}} \quad (4.12)$$

$$T_{\text{new}}^{\text{BG}} = \frac{1 - R_{\text{event,new}}}{1 - R_{\text{event,original}}} T_{\text{original}}^{\text{BG}} \quad (4.13)$$

と換算できる。新アルゴリズムでは旧アルゴリズムと同じ event-level efficiency を保ちつつ、event-level background rejection rate を相対的に 172% 向上させ、beam background event の大幅な削減を実現している。表 4.5 に示した目標ルミノシティにおけるトリガーレートを見ると、旧アルゴリズムでは上限値の 20 kHz を超えていたが、新アルゴリズムでは上限値以下に抑えられることがわかる。なお、 $B$  中間子の崩壊においては少なくとも 2 本の荷電粒子が生成されるため、86% の track-level efficiency は event-level に換算すると 98% 以上となり、3.1 節で示した  $B$  中間子の崩壊事象に対するトリガー効率がおおよそ 100% という要求は満たされている。

表 4.4 新旧アルゴリズムにおける各性能指標の比較

| 性能指標                   | 旧アルゴリズム                              | 新アルゴリズム                               |
|------------------------|--------------------------------------|---------------------------------------|
|                        | ( $z_{\text{cut}} = 30 \text{ cm}$ ) | ( $z_{\text{cut}} = 9.3 \text{ cm}$ ) |
| $E_{\text{track}}$ (%) | 86.0                                 | 86.1                                  |
| $R_{\text{track}}$ (%) | 30.0                                 | 73.3                                  |
| $E_{\text{event}}$ (%) | 93.1                                 | 93.1                                  |
| $R_{\text{event}}$ (%) | 25.3                                 | 68.7                                  |

表 4.5 目標ルミノシティにおけるトリガーレートの新旧アルゴリズムでの比較。 $B$  中間子の物理に対する CDC トリガービットを適用している。

| イベント分類                | 目標ルミノシティにおけるトリガーレート (kHz)            |                                       |
|-----------------------|--------------------------------------|---------------------------------------|
|                       | 旧アルゴリズム                              | 新アルゴリズム                               |
|                       | ( $z_{\text{cut}} = 30 \text{ cm}$ ) | ( $z_{\text{cut}} = 9.3 \text{ cm}$ ) |
| Signal event          | 6.9                                  | 6.9                                   |
| Beam background event | 21.3                                 | 8.9                                   |
| Total                 | 28.2                                 | 15.8                                  |

## 第 5 章

# Hough selection アルゴリズムのファームウェア実装

本アルゴリズムは 3.2 節で詳細を説明した UT4 ボード上の FPGA (Xilinx 社の Virtex Ultra-Scale XCVU160) に実装した。現時点でファームウェアの論理合成と配置配線が完了しており、シミュレーションによるデバッグを経て実機に実装する予定である。3D Fitter は 4 台の UT4 ボードを使用し、それぞれのボードは 2 次元飛跡の角度をもとに  $90^\circ$  ごとに分けられた 4 つの領域の内 1 つを担当する。現時点における 3D Fitter 以外の回路のレイテンシから、本アルゴリズムに対するレイテンシは約  $0.8 \mu\text{s}$  以下であることが要求されている。ファームウェアは同期回路として設計されており、システムクロックには SuperKEKB 加速器の加速空洞に使用している交流周波数 509 MHz を 4 分周した 127 MHz のクロックが用いられている。一連の処理はパイプライン化されており、毎クロック異なる入力を受け付けることができる。開発には、ハードウェア記述言語として VHDL、ソフトウェアとして Xilinx 社 Vivado を使用した。本章では、ファームウェアの概略、その中でも特に工夫した Hough voting の実装、最後にリソース使用量やレイテンシの評価を示す。

### 5.1 ファームウェアの概略

図 5.1 に Hough selection アルゴリズムにおけるファームウェアのモジュール構成を示す。ここでのモジュールとはハードウェア記述言語で 1 つのファイルに記述される、ひとまとまりの機能の単位である。以下に各モジュールのはたらきを簡単に説明する。

■Receiver&Decoder Track Segment Finder、2D Track Finder、そして Event Timing Finder から入力されるシリアル通信の信号を受信する。高速シリアル通信の送受信を行うトランシーバー

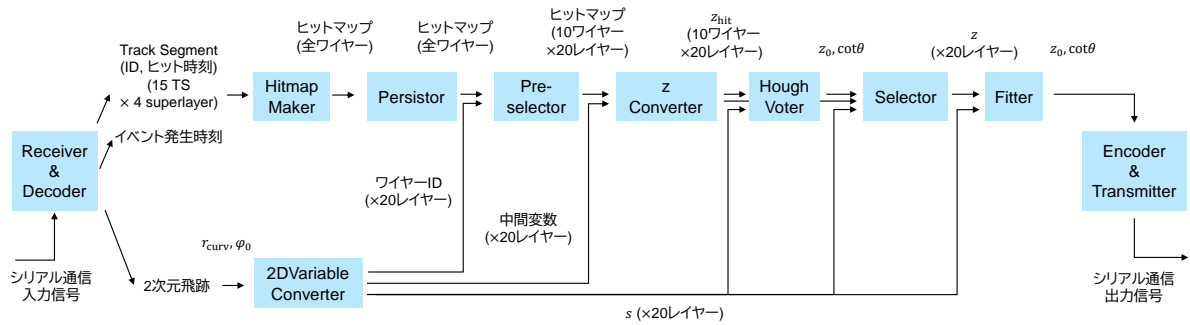


図 5.1 3D Fitter の Hough selection アルゴリズムにおけるファームウェアのモジュール構成

として Xilinx 社の GTY トランシーバーを用いている。また、シリアル通信には 64b/66b ある  
いは 8b/10b エンコーディングを用いたトリガーシステム独自の通信プロトコルが用いられてお  
り [22]、ここで信号のデコードを行う。

■Hitmap Maker 独立に入力されるヒット情報をまとめて扱うため、図 5.2 に示すようなヒット  
マップとよばれる形式に変換する。このヒットマップの各セルは 1 つのステレオワイヤーに対応  
し、ヒットの有無とドリフト時間の情報を持つ。

■Persistor 時間的にばらばらに入力されるヒット情報を一つのイベントとして扱うため、ヒット  
マップ上の情報をドリフト時間の最大値に相当する約 500 ns 保持する。ヒットマップの各セルに  
ダウンカウンタを設定し、500 ns に相当するクロック数が経過したときにヒット情報を消去し、新  
たなヒット情報が入力された時には上書きしダウンカウンタをリセットする。

■2D Variable Converter 入力された 2 次元飛跡のパラメータ  $r_{\text{curv}}$ 、 $\varphi_0$  を、各入力に対する  
出力を表として表す Look Up Table や加算器を用いて後段で使われる変数に変換する。図 5.3  
に回路図を示す。Look Up Table の実装に用いられる素子は Block Random Access Memory  
(Block RAM、BRAM) であるため、Look Up Table は図中では BRAM と表記している。図中  
の Preselection ID は、後段の Preselector で選ばれる 10 ワイヤーのうち、最もワイヤー ID が小

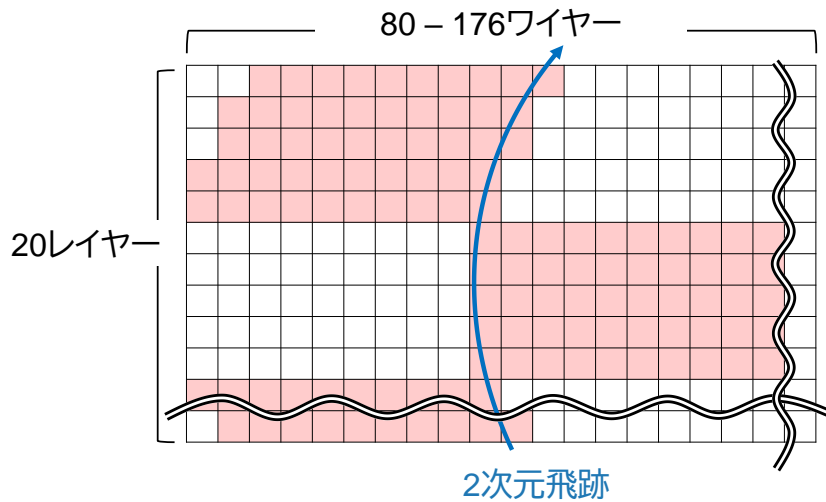


図 5.2 Hitmap Maker、Persistor、および Preselector で用いられるヒットマップの模式図。長方形で示されているが、実際には各 superlayer ごとにワイヤー数が異なる。Preselector では図中赤色で示した 2 次元飛跡と交わりうる 10 本のステレオワイヤーが選ばれ後段に送られる。赤色の領域が左右に分かれているのは、superlayer ごとにステレオ角の正負が互い違いになっているためである。

さいものを指す。

■Preselector 図 5.2 に示すように入力された 2 次元飛跡と交わるステレオワイヤーは、各レイヤーにつき最大 10 ワイヤー存在する。2D Variable Converter で求めた Preselection ID をもとに、ヒットマップの中から各レイヤーにつき 10 ワイヤーを選び出して後段に送る。ヒット候補は

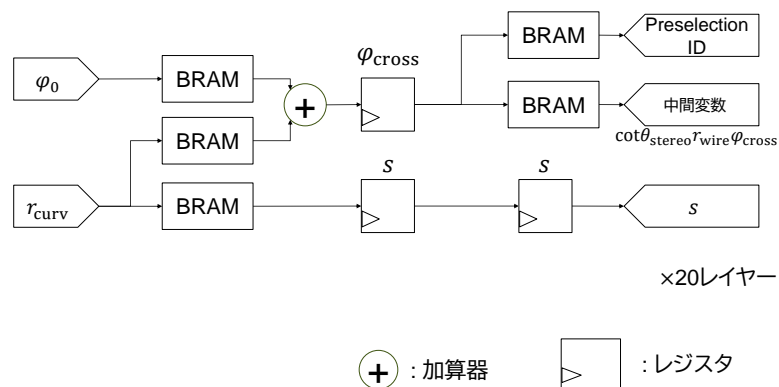


図 5.3 2D Variable Converter の回路図。図中の  $\varphi_{\text{cross}}$  は式 (3.5) で定義した変数である。

最大 10 ワイヤーなので、後段に送る 10 ワイヤーの中には 2 次元飛跡と交わらないワイヤーも数本存在するが、後段の z Converter で Hough voting の範囲外の  $z$  が計算されてそのワイヤーは無視されるため問題ない。

■z Converter 2D Variable Converter で求めた中間変数と各ワイヤーの ID、ドリフト時間から式 (3.10) にしたがって  $z_{hit}$  を求める。 $z_{hit}$  は式 (3.10) で求めた変数である。図 5.4 に回路図を示す。

■Hough Voter Hough voting によって 3 次元飛跡パラメータ  $z_0$  と  $\cot \theta$  を決定する。詳細は次節で説明する。

■Selector 4.1.3 項で説明したヒット選択方法に基づき、ヒット候補の中からフィッティングに用いるヒットの  $z$  を後段に送る。ヒット候補それぞれに対し z Converter と Hough Voter で求めた変数から  $\Delta z := |z_{hit} - z(s)|$  を求め、各セルをトーナメント式に比較して  $\Delta z$  の小さなヒットを各レイヤーごとに 1 つ選択する。

■Fitter 4.1.4 項で示したフィッティングの計算式に基づき最終的な  $z_0$  と  $\cot \theta$  を決定する。乗算と加算の一部は Digital Signal Processor と呼ばれる、デジタル信号の演算に特化した FPGA リソースを利用した。除算は Look Up Table を用いて実装した。

■Encoder&Transmitter Receiver&Decoder とは逆に、GRL への出力信号をエンコードし、GTY トランシーバによって送信する。

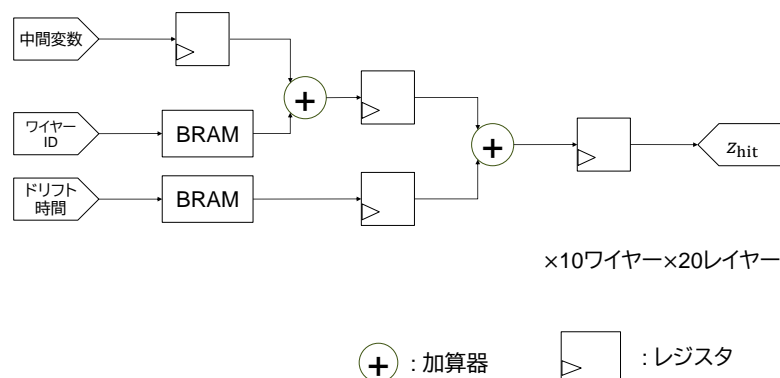


図 5.4 z Converter の回路図



## 5.2 Hough voter モジュールの詳細

Hough voter モジュールは図 5.5 に示すように、大きく分けて 4 つの処理からなる。以下ではそれぞれの処理を簡単に説明する。

### 5.2.1 直線の投票

各レイヤーごとにヒットの  $s$  と  $z$  の情報に基づき直線を  $\cot\theta$ - $z_0$  平面のセルに投票することを考える。 $k$  番目のヒットの  $z$  を  $z_k$  とおくと投票すべき直線の式は  $z_0 = -s \cdot \cot\theta + z_k$  であるが、まずはヒットの  $z$  情報を無視して、直線  $z_0 = -s \cdot \cot\theta$  を投票する。同じレイヤーに複数のヒットがあっても、 $x$ - $y$  平面における 2 次元飛跡とワイヤーの交点は共通なので、 $s$  も共通であることに注意する。図 5.6 のようにセルとその境界値が与えられているとする。傾き  $-s$  は常に負であるから、直線が  $\text{cell}_{i,j}$  を通るための条件は以下のように表せる。

$$\text{直線 } z_0 = -s \cdot \cot\theta \text{ が } \text{cell}_{i,j} \text{ を通る} \quad (5.1)$$

$$\iff \text{直線が } \begin{array}{|c|} \hline \text{---} \\ \hline \end{array} \text{ を通る または } \begin{array}{|c|} \hline \text{---} \\ \hline \end{array} \text{ を通る} \quad (5.2)$$

$$\iff z_{0,j} \leq s \cdot \cot\theta_i \leq z_{0,j+1} \text{ または } \cot\theta_i \leq -\frac{z_{0,j+1}}{s} \leq \cot\theta_{i+1} \quad (5.3)$$

1. 各レイヤーごとに直線を投票
2. 各ヒットの  $z$  情報  $z_k$  を反映
3. セルごとに得票数を集計
4. ピーク検出

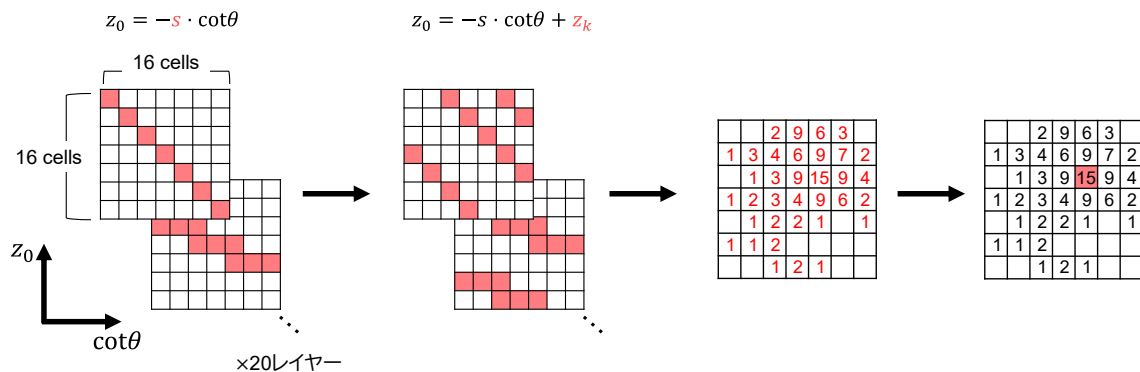


図 5.5 Hough voter モジュールの処理の流れ

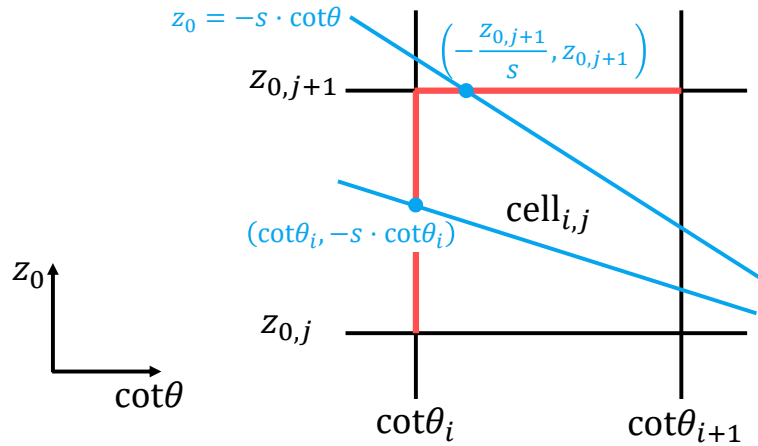


図 5.6  $\cot \theta - z_0$  平面のセル  $\text{cell}_{i,j}$  の上を直線  $z_0 = -s \cdot \cot \theta$  (図中青線) が通るためには、図中赤線で示した辺のいずれかを通る必要がある。

ここで、 $c_{i,j} := -z_{0,j} / \cot \theta_i$  と定義し、パラメータの正負による不等号の向きに注意しながら、 $s$  に関する条件になるように式変形を進めると、

$$(5.3) \iff \begin{cases} c_{i,j+1} \leq s \leq c_{i,j} \text{ または } c_{i+1,j+1} \leq s \leq c_{i,j+1} & (\cot \theta_i > 0) \\ c_{i,j} \leq s \leq c_{i,j+1} \text{ または } \max(c_{i,j+1}, c_{i+1,j+1}) \leq s & (\cot \theta_i < 0 \text{ かつ} \\ c_{i,j} \leq s \leq c_{i,j+1} \text{ または } c_{i,j+1} \leq s \leq c_{i+1,j+1} & \cot \theta_{i+1} > 0) \\ & (\cot \theta_{i+1} < 0) \end{cases} \quad (5.4)$$

を得る。ここで注目してほしいのは、セルへの投票条件が入力変数  $s$  と定数  $c_{i,j}$  との比較および論理演算で表されていることである。この定数  $c_{i,j}$  の計算と  $\cot \theta_{i(+1)}$  による場合分けは事前にプログラムできるので、実行時には比較と論理演算さえすればよい。各セルで平行してこの処理を行うことで、図 5.5 の 1 番に示したような、1 つの直線が投票されたセルマップが 20 レイヤー分得られる。このセルマップは、セル上を直線が通る場合に 1 をとるビット変数の 2 次元配列として表現する。

## 5.2.2 ヒットの $z$ 情報の反映

前項で無視した  $z_k$  を反映するには、上で投票した直線  $z_0 = -s \cdot \cot \theta$  を  $z_0$  軸方向に  $z_k$  だけ平行移動させればよい。その前段階の準備として、入力変数  $z_k$  のフォーマットを変換する。この変換の目的は、図 5.7 のように各ヒットの  $z_k$  を、 $z$  方向に分割されたセル上にヒットマップとしてまとめることである。このヒットマップを  $\text{zmap}$  と呼ぶことにする。 $\text{zmap}$  はビット列として定義され、 $\text{zmap}_j = 1$  のとき、 $j \cdot 8 \leq z < (j+1) \cdot 8$  (単位: cm) を満たす  $z$  の範囲にヒットが存在することを表す。8 cm という値は  $\text{zmap}$  の 1 セルの幅で、前項で議論したセルの  $z_0$  軸方向のセル幅

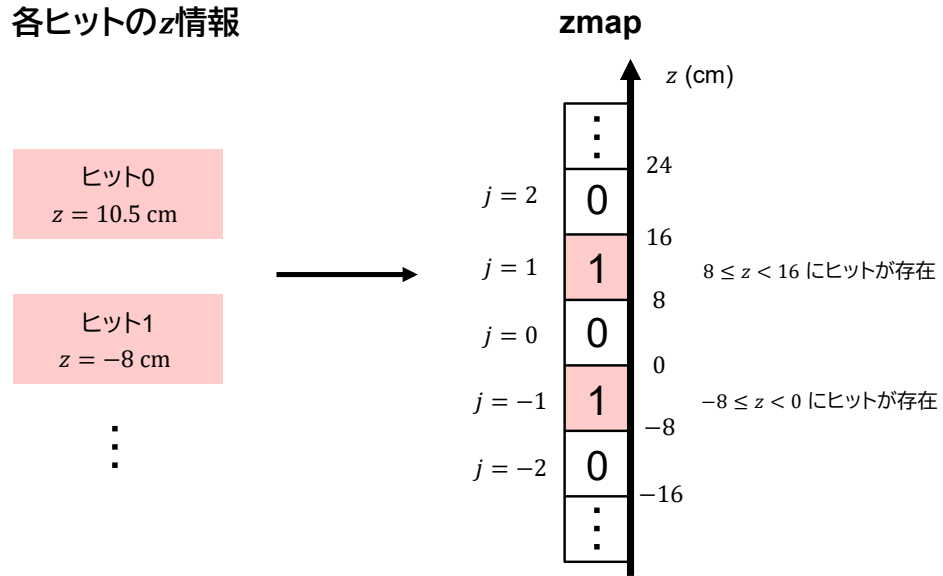


図 5.7 各ヒットの  $z$  情報  $z_k$  を zmap という形式に変換する。zmap は  $z$  方向に分割された領域内にヒットが存在するかどうかを表すビット列として定義される。

と同じ値に設定されている。 $z_k$  が与えられたとき、そのヒットが属する zmap の添字  $j$  は

$$j = \left\lfloor \frac{z_k}{8} \right\rfloor \quad (5.5)$$

と表せる。ここで床関数  $\lfloor x \rfloor$  は  $x$  を超えない最大の整数を表し、例えば  $\lfloor -0.1 \rfloor = -1$  である。幅を 8 cm と選んだおかげで、この計算は簡単にファームウェアで行うことができる。例として、 $z_k = 10.5$  cm が与えられ、固定小数点表示の 2 進数で 1010.100 と表されているとする。8 で割ると右に 3 ビットシフトするので 0001.010、床関数は単に整数部分を取り出す操作なので 0001 となり、 $j = 1$  が得られる\*1。各ヒットこの操作をすることで図 5.7 右のような zmap が得られる。

次に、zmap を使って前項で求めたセルマップを平行移動させる。前項で求めたセルマップを cell、平行移動後のセルマップを cell' と呼んで区別する。cell' が図 5.8 左のように表されているとき、 $cell'_{i,j} = 1$  となるのは  $cell_{i,j-1} = 1$  を上に 1 セル平行移動させる場合、または  $cell_{i,j} = 1$  を平行移動させない場合、または  $cell_{i,j+1} = 1$  を下に 1 セル平行移動させる場合など、平行移動させるセル数に応じた複数の場合の or で表すことができる。論理式で表すと以下のようなになる。

$$cell'_{i,j} = \dots \text{ or } (cell_{i,j-1} \text{ and } zmap_1) \text{ or } (cell_{i,j} \text{ and } zmap_0) \text{ or } (cell_{i,j+1} \text{ and } zmap_{-1}) \text{ or } \dots \quad (5.6)$$

$$= \text{or\_reduce}(cell^{(i)} \text{ and } \overleftarrow{zmap}^{(j)}) \quad (5.7)$$

\*1 負数の場合は 2 の補数表現に対する適切な処理が必要となる。今は説明が簡単になるよう負の添字も許しているが、実際の実装では正の添字しかとらないのでこの処理で問題ない。

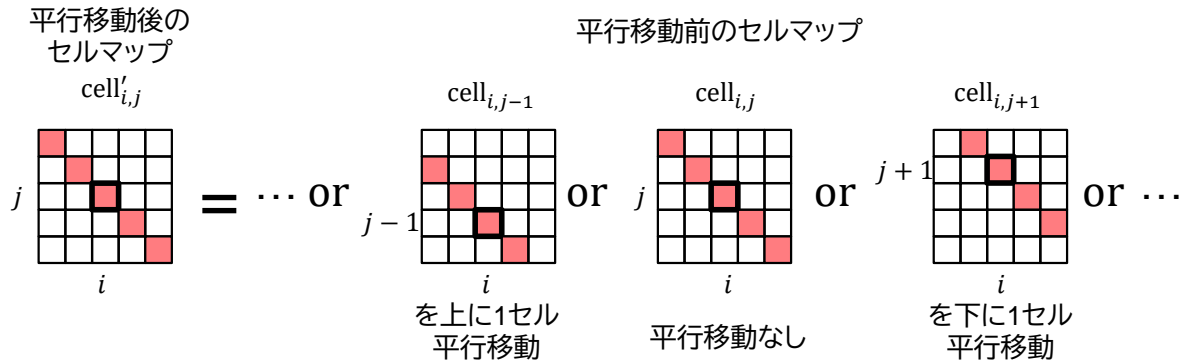


図 5.8 各ヒットの  $z$  情報を反映したセルマップは、前項で求めたセルマップを  $zmap$  にしたがって平行移動させることで得られる。

ここで  $cell^{(i)}$  は  $j$  番目の要素に  $cell_{i,j}$  をもつビット列、 $\overleftarrow{zmap}^{(j)}$  は  $zmap$  の要素の並びを逆転させたうえで (式 (5.6) において  $zmap$  の添え字が  $\dots, 1, 0, -1, \dots$  と逆順に並んでいることに注意)  $j$  番目の要素に  $zmap_0$  がくるよう要素をスライドさせたビット列である。また、ビット列に作用している  $and$  はビットごとの  $and$  であり、 $or\_reduce$  はビット列の各要素の  $or$  をとった結果を返す  $or$  リダクション演算である。ビット列  $vec$  に対する  $or$  リダクション演算は、VHDL 言語では  $or\_reduce(vec)$ \*<sup>2</sup>、Verilog 言語では  $|vec$  と表す。各セルごとに式 (5.7) の演算を行うことで、図 5.5 の 2 番目の図のように、各ヒットの  $z$  座標が反映されたセルマップが得られる。

### 5.2.3 集計

各セルごとに、20 レイヤーの得票数を足し上げる。トーナメント式に加算することで  $\log_2 20 \sim 5$  回の加算で処理できる。

\*<sup>2</sup> ieee.std.logic.misc パッケージが必要

## 5.2.4 ピーク検出

$16 \times 16 = 256$  セルの中で最多得票数のセルを見つける。トーナメント式に比較することで  $\log_2 256 = 8$  回の比較で処理できる。 $z_0 = 0$  からくる飛跡信号に対する efficiency を高く保つために、同じ得票数の場合は  $|z_0|$  が 0 に近いほうを優先させる。

## 5.3 リソース使用量とレイテンシ

配置配線後のリソース使用割合は表 5.1 に示すようにどれも 100% 以内に収まっている。組み合わせ回路に用いられる素子である Look Up Table (LUT)\*<sup>3</sup>が一番消費割合が多く、その後は大規模なメモリ構造である Block RAM、記憶素子として用いられる Register、加算器での桁上りを高速に計算する CARRY8 (8 は 8 ビットを指す)、デジタル信号の演算に特化した Digital Signal Processor (DSP) と続く。

また、各モジュールのレイテンシを図 5.9 に示す。合計で約  $0.3 \mu\text{s}$  となり、3D Fitter に対する要求値の  $0.8 \mu\text{s}$  に収まっている。また、配置配線の結果タイミング違反がないことを確認した。

今後はファームウェアのシミュレーションを行いデバッグを行い、実機へ実装する予定である。

表 5.1 配置配線後の各リソースの使用割合。使用量が 1% 未満のリソースは省略している。

| リソース名     | 使用割合 (%) |
|-----------|----------|
| LUT       | 40.1     |
| Block RAM | 6.5      |
| Register  | 6.5      |
| CARRY8    | 3.1      |
| DSP       | 2.5      |

\*<sup>3</sup> LUT という用語には、素子としての LUT と、Block RAM を用いて実装される機能としての LUT の 2 種類がある。

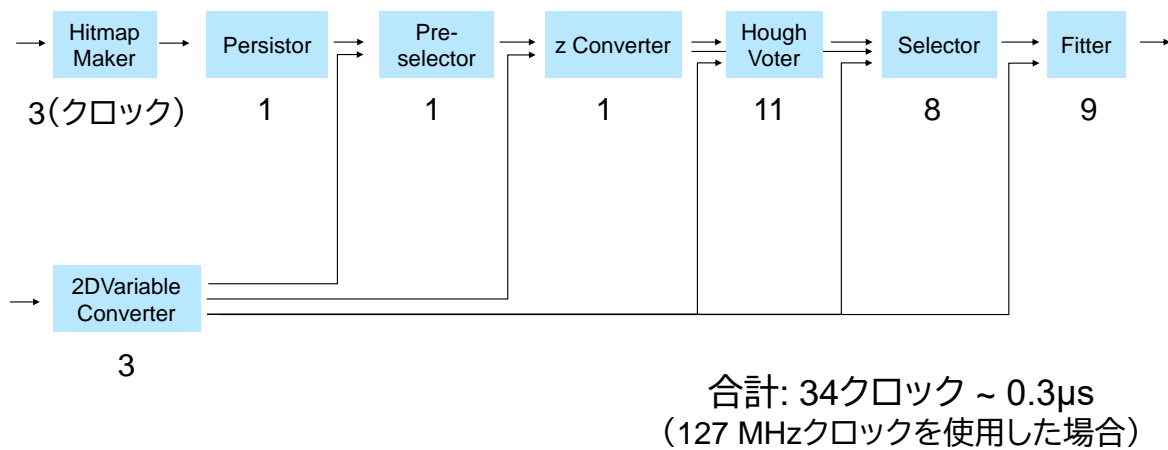


図 5.9 各モジュールのレイテンシ

## 第 6 章

# 結論

$B$  中間子などの崩壊を通じて新物理の探索を目指す Belle II 実験は、実験終了までに目標ルミノシティ  $6.0 \times 10^{35} \text{ cm}^{-2}\text{s}^{-1}$  を目指している。Level-1 トリガーシステムではトリガーレートの上限值として 30 kHz が定められているが、2022 年の最高ルミノシティ  $4.6 \times 10^{34} \text{ cm}^{-2}\text{s}^{-1}$  を記録した物理ランをもとにトリガーレートを線形に外挿すると、上限値を超えてしまう。トリガーレート増加の原因となっている beam background は飛跡発生点の  $z$  座標  $z_0$  が大きい領域からくるため、CDC のデータを用いて 3 次元飛跡を再構成し、 $z_0$  を見ることで beam background を排除できる。

そこで本研究では、3 次元飛跡再構成回路の一つである 3D Fitter の改良を行った。この改良ではトリガーボードを UT4 ボードにアップグレードし、回路に実装するアルゴリズムとして Hough selection アルゴリズムを新たに開発した。これにより旧アルゴリズムが持っていたフィッティングに使用するヒット数が少ない、ノイズによるヒットを選択している、ノイズの多いヒットパターンでドリフト補正が適用されないという問題を解決することができた。実データを用いたシミュレーションでは、新アルゴリズムでは旧アルゴリズムと同じ信号飛跡の再構成効率を保ちつつ 144% 多くの背景飛跡を排除できることが示された。目標ルミノシティにおけるトリガーレートに換算すると、beam background event のトリガーレートを 21 kHz から 9 kHz へと削減できたことになり、signal event のトリガーレート 7 kHz と合わせても CDC トリガーの上限値 20 kHz 以下に抑えられることが示された。

現時点で FPGA へのファームウェア実装は完了しており、リソース使用量とレイテンシが許容範囲内に収まることを確認した。今後はシミュレーションによるデバッグを行い、その後 FPGA 実機への実装を行う予定である。

# 謝辞

本研究を遂行し修士論文にまとめるにあたり、多くの方からの助けをいただきました。指導教員である後田裕教授には日々のゼミでの的確なアドバイスや、学会や勉強会への参加の援助を通じて成長の機会を与えていただきました。古賀太一朗助教には毎週のミーティングで研究へのコメントをいただいたり、本論文の添削にお付き合い下さったりなど、多くの助けをいただきました。本論文の内容に含めることはできませんでしたが、スタッフの中沢遊氏には CDC 読み出し回路の放射線耐性の試験に参加する機会を与えてくださり、学びの多い貴重な経験をさせていただきました。研究室およびトリガーグループの皆様には議論を通じて多くの有用なアドバイスをいただきました。本論文を審査して下さった審査委員の先生方にもお世話になりました。深く感謝申し上げます。最後に、故郷の地で研究を応援してくれた両親に感謝します。ありがとうございました。



## 付録 A

# Hough selection アルゴリズムのパラメータ最適化の詳細

本文の 4.3.3 項で省略した、Hough selection アルゴリズムで使われるパラメータの最適化の詳細を説明する。表 4.2 で示した最適化するパラメータの一覧を表 A.1 に再掲する。パラメータの最適化はグリッドサーチによる全探索で行い、efficiency が 90% での background rejection rate を最大化する目的関数として定める。

### A.1 $z_0$ 方向のセル数 $N_{\text{cell},z_0}$ 、 $\cot \theta$ 方向のセル数 $N_{\text{cell},\cot \theta}$ 、得票数のしきい値 $N_{\text{threshold}}$ 、およびピーク検出手法の最適化

まずピーク検出手法として clustering を用いた場合のグリッドサーチの結果を示す。各パラメータでの efficiency=90% での background rejection rate を図 A.1 に示す。この図では得票数のしきい値  $N_{\text{threshold}}$  は試した中で最適な値が選択されており、その値は図 A.2 に示されている。

表 A.1 最適化するパラメータの一覧

| 最適化の順番 | パラメータ名   | 試した値                 | 刻み幅 | 初期値 |
|--------|--|----------------------|-----|-----|
| 1      | $z_0$ 方向のセル数 $N_{\text{cell},z_0}$                 | [10, 50]             | 5   | -   |
|        | $\cot \theta$ 方向のセル数 $N_{\text{cell},\cot \theta}$ | [10, 50]             | 5   | -   |
|        | 得票数のしきい値 $N_{\text{threshold}}$                    | [5, 15]              | 1   | -   |
|        | ピーク検出手法  | {maximum,clustering} | -   | -   |
| 2      | ヒット選択範囲 $z_{\text{selection}}$ [cm]                | [5, 50]              | 5   | 20  |

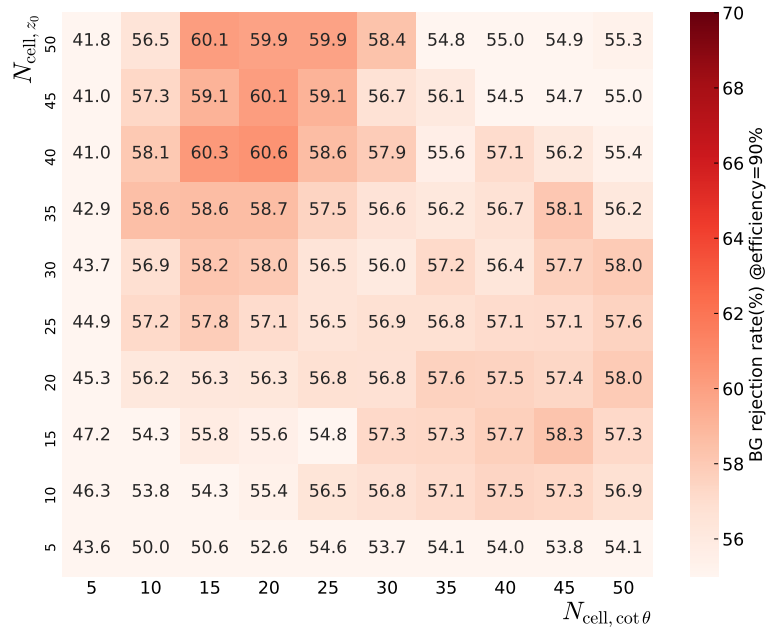


図 A.1 ピーク検出手法として clustering を用いた場合のグリッドサーチの結果。各  $N_{\text{cell},z_0}$ 、 $N_{\text{cell},\cot\theta}$  における efficiency=90% での background rejection rate がヒートマップで示されている。得票数のしきい値  $N_{\text{threshold}}$  は試した中で最適な値が選択されており、その値は図 A.2 に示されている。

次に、ピーク検出手法として maximum を用いた場合の各パラメータでの efficiency 90% での background rejection rate を図 A.3 に示す。

以上の結果から、ピーク検出手法=maximum、 $N_{\text{cell},z_0}=25$ 、 $N_{\text{cell},\cot\theta}=10$  と選ぶのが最適である。

## A.2 ヒット選択範囲 $z_{\text{selection}}$ の最適化

図 A.4 にヒット選択範囲  $z_{\text{selection}}$  を変えたときの efficiency=90% での background rejection rate を示す。この結果から、 $z_{\text{selection}}=25$  cm と選ぶのが最適である。

## A.3 ファームウェア実装を考慮したパラメータの決定

上に示したように最適なパラメータが求められたが、セル数が 2 の累乗のほうが都合がよい、 $z_0$  方向のセルサイズ [cm] が 2 の累乗のほうが都合がよい、というファームウェア実装上の理由から、実装に用いるパラメータは図 A.1 に示す値に決定した。この変更による background rejection

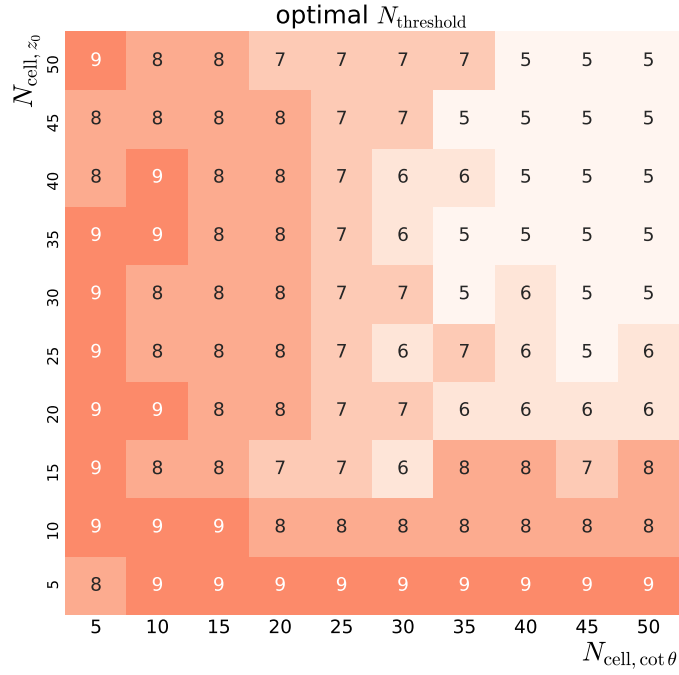


図 A.2 ピーク検出手法として clustering を用いた場合のグリッドサーチの結果。各  $N_{\text{cell},z_0}$ 、 $N_{\text{cell},\cot\theta}$  における最適な得票数のしきい値  $N_{\text{threshold}}$  が示されている。

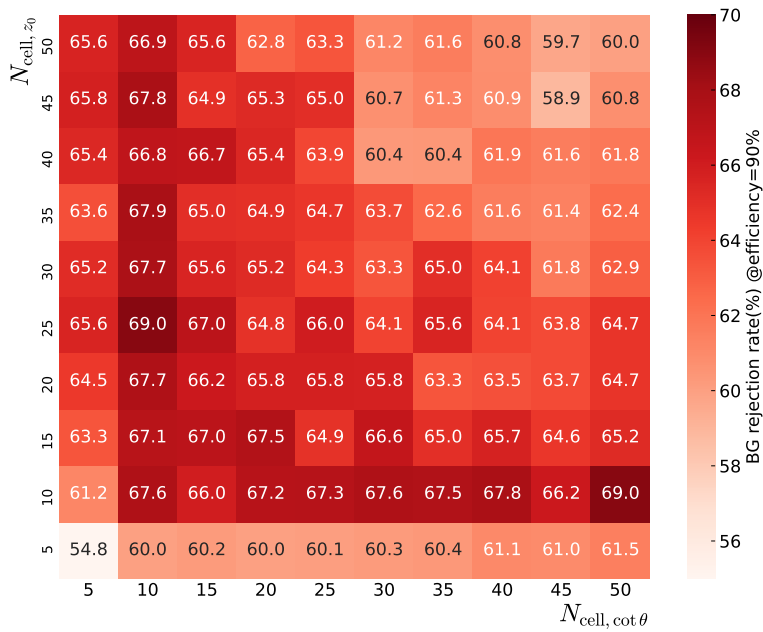


図 A.3 ピーク検出手法として maximum を用いた場合のグリッドサーチの結果。各  $N_{\text{cell},z_0}$ 、 $N_{\text{cell},\cot\theta}$  における efficiency=90% での background rejection rate がヒートマップで示されている。

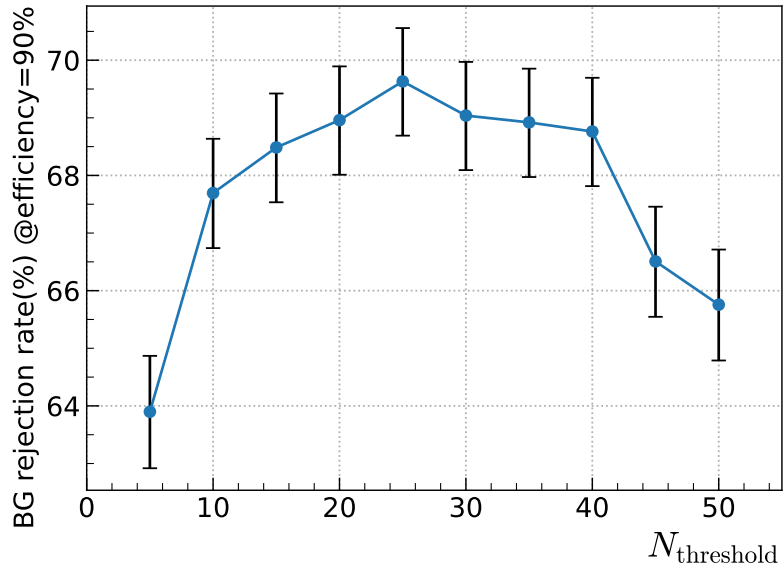


図 A.4 ヒット選択範囲  $z_{\text{selection}}$  を変えたときの efficiency=90% での background rejection rate。統計誤差として、Clopper-Pearson の 68.3 % 信頼区間を表示している。

rate の差は 2% 程度である。

表 A.2 パラメータの最適化結果

|                                     | 最適値     | 実装値     |
|-------------------------------------|---------|---------|
| $z_0$ 方向のセル数                        | 25      | 16      |
| $\cot \theta$ 方向のセル数                | 10      | 16      |
| ピーク検出手法                             | maximum | maximum |
| ヒット選択範囲 $z_{\text{selection}}$ [cm] | 25      | 25      |
| background rejection rate [%]       | 69.6    | 67.3    |

## 参考文献

- [1] Nicola Cabibbo. Unitary symmetry and leptonic decays. *Phys. Rev. Lett.*, 10:531–533, Jun 1963.
- [2] Makoto Kobayashi and Toshihide Maskawa. CP Violation in the Renormalizable Theory of Weak Interaction. *Prog. Theor. Phys.*, 49:652–657, 1973.
- [3] W. Altmannshofer et al. The Belle II Physics Book. *PTEP*, 2019(12):123C01, 2019. [erratum: PTEP2020,no.2,029201(2020)].
- [4] R. L. Workman et al. Review of Particle Physics. *PTEP*, 2022:083C01, 2022.
- [5] KEK. ”superkekb プロジェクト全体図”. [https://www.kek.jp/wp-content/uploads/2021/10/20180320\\_superkekb\\_001.png](https://www.kek.jp/wp-content/uploads/2021/10/20180320_superkekb_001.png). (アクセス日 2023 年 11 月 29 日).
- [6] Belle II Collaboration. Belle II Technical Design Report. <https://arxiv.org/abs/1011.0352>, 2010.
- [7] KEK. ”luminosity projection”. [https://www-superkekb.kek.jp/Luminosity\\_projection.html](https://www-superkekb.kek.jp/Luminosity_projection.html). (アクセス日 2023 年 11 月 30 日).
- [8] KEK. ”プレスキット「superkekb/belle ii で電子・陽電子の初衝突を観測」”. [https://www.kek.jp/wp-content/uploads/2021/10/20180425\\_belle2\\_002.png](https://www.kek.jp/wp-content/uploads/2021/10/20180425_belle2_002.png). (アクセス日 2023 年 11 月 29 日).
- [9] 谷口七重. Belle II CDC. *高エネルギーニュース*, 32(4):241, 2013.
- [10] Tomohisa Uchida, Masahiro Ikeno, Yoshihito Iwasaki, Masatoshi Saito, Shoichi Shimazaki, Manobu Tanaka, Nanae Taniguchi, and Shoji Uno. Readout electronics for the central drift chamber of the Belle II detector. In *2011 IEEE Nuclear Science Symposium and Medical Imaging Conference*, pages 694–698, 2011.
- [11] Xilinx. Virtex-6 family overview. <https://docs.xilinx.com/v/u/en-US/ds150>, 2015. DS150 (v2.5).
- [12] Xilinx. Ultrascale architecture and product data sheet: Overview. <https://docs.xilinx.com/v/u/en-US/ds890-ultrascale-overview>, 2023. DS890 (v4.4.1).

- [13] Sara Pohl and Christian Kiesling. *Track Reconstruction at the First Level Trigger of the Belle II Experiment*. PhD thesis, Munich, Ludwig-Maximilians-Universität, Munich, 2018.
- [14] Ping Ni. Upgrade of Two Dimensional Track Trigger on Central Drift Chamber aimed for Belle II Targeted Luminosity. Master's thesis, Tokyo, University of Tokyo, 2022.
- [15] Yuki Sue, Bae Hanwook, Toru Iijima, Yoshihito Iwasaki, Taichiro Koga, Yun-Tsung Lai, Hideyuki Nakazawa, and Kai Lukas Unger. The Event Timing Finder for the Central Drift Chamber Level-1 Trigger at the Belle II experiment. *J. Phys. Conf. Ser.*, 2374(1):012103, 2022.
- [16] E. Won, J. B. Kim, and B. R. Ko. Three-dimensional fast tracker for the central drift chamber based level-1 trigger system in the belle ii experiment. *Journal of the Korean Physical Society*, 72(1):33–37, January 2018.
- [17] A. Paladino. Beam background evaluation at superkekb and belle ii. *Journal of Instrumentation*, 15(07):C07023, jul 2020.
- [18] H. Ikeda et al. Observation of sudden beam loss in SuperKEKB. *JACoW, IPAC2023:MOPL072*, 2023.
- [19] P. V. C. Hough. Machine Analysis of Bubble Chamber Pictures. *Conf. Proc. C*, 590914:554–558, 1959.
- [20] T. Kuhr, C. Pulvermacher, M. Ritter, T. Hauth, and N. Braun. The Belle II Core Software. *Comput. Softw. Big Sci.*, 3(1):1, 2019.
- [21] C. J. Clopper and E. S. Pearson. The use of confidence or fiducial limits illustrated in the case of the binomial. *Biometrika*, 26(4):404–413, 1934.
- [22] 岩崎義仁 他. Belle II 実験 トリガーシステム. *高エネルギーニュース*, 34(2):105, 2015.