

2021 年度 修士学位論文

# Belle II 実験における ARICH 検出器のアライメント

指導教員 角野秀一 教授

東京都立大学大学院 理学研究科 物理学専攻  
博士前期課程 2 年 高エネルギー物理実験研究室

20844426 朴兌秀

2022 年 1 月 7 日

## 概要

Belle II 実験とは、茨城県つくば市にある高エネルギー加速器研究機構 (KEK) において、Super KEKB 加速器と衝突点に設置された Belle II 測定器を使って  $B$  中間子および  $\tau$  レプトン崩壊を詳細に調べる電子・陽電子衝突実験である。 $B$  中間子の崩壊過程における稀崩壊事象を観測し、それを精密に解析することにより、素粒子標準模型を超える新しい物理の探索を目的としている。

本実験における Belle II 測定器は、7 種類の異なる検出器で構成されている複合型の測定器であり、 $B$  中間子崩壊で生じる粒子の検出及び各種測定を行う。それぞれの検出器は、例えば荷電粒子の飛跡測定、粒子識別、中性粒子検出とレプトン識別などといった役割を担っており、個々の目的に合わせて測定器の構成要素、測定原理なども異なっている。 $B$  中間子や  $\tau$  レプトンを生成する加速器として Super KEKB 加速器があり、この加速器を使うことによってそれぞれ 7 GeV と 4 GeV に加速された電子と陽電子が Belle II 測定器のビーム衝突点で衝突する。この際に  $B$  中間子対が大量に生成され、その稀崩壊事象を観測することができる。

本研究では、Belle II 測定器の構成要素であり、荷電  $K/\pi$  中間子の識別を担っている ARICH 検出器 (Aerogel Ring Imaging Cherenkov counter) の内部飛跡検出器 (CDC) に対する位置のずれを、実データを用いて算出・補正した。このずれを算出し、補正するような作業のことをアライメントと呼ぶ。ARICH は主に光検出器 HAPD と、輻射体シリカエアロゲルで構成され、このエアロゲルを通過した荷電粒子が円錐状にチェレンコフ光を発生させた際の粒子のトラックと、チェレンコフ光のなす放射角 (チェレンコフ角度) を算出することにより粒子識別を行っている。ARICH が設計上の位置からずれている場合、測定されるチェレンコフ角度が真の値から系統的にずれてしまい、同検出器の荷電  $K/\pi$  中間子識別能力が低下してしまうため、ARICH のアライメントを行うということは、検出器の性能上先述の中間子識別において識別精度の向上に繋がる重要な作業である。

本研究のアライメントでは、第一に先行研究が行われた時期よりも後に取得された実データ (Phase III の Bucket 15 という期間に取得されたもの) を用いて、さらに実データにおける粒子の選別条件を変更しその他の条件は先行研究と同一な方法で ARICH のアライメントを試みた。まず ARICH を 36 領域に分割し、ARICH を一つの剛体とみなした場合の並進方向および回転方向のずれを確認した。その後 36 領域の放射角と放射方位角の依存性を、ARICH の並進方向および回転方向のずれに対応する六つのパラメータでフィットし、ARICH の並進方向および回転方向のずれの大きさを算出した。その際に得られた結果から従来のアライメントの方法にはいくつか問題点があったため、新しいアライメントの手法の開発を試みた。

先行研究とは異なり本研究における新しいアライメントの手法の開発ではプロセス  $e^+ + e^- \rightarrow \mu^+ + \mu^-$  を使用した。このプロセスは  $\mu$  粒子が電子と比較し電磁シャワーやハドロン相互作用を発生させないため荷電粒子の飛跡が比較的単純なものになり、本研究のアライメントに適しているために採用した。さらに、先行研究で用いられていた分割された ARICH の 36 領域ごとのアライメントフィットはフィッティング精度が低かったため、フィットの手法を変更してこれの向上を目指した。具体的にはアライメントフィットのためのプログラムを新たに作成し、そのプログラム上で ARICH のずれを計算する際に考慮すべき誤差やフィットの手法の見直しを試みた。その結果、この新しい

プログラムを用いてアライメントフィットをやり直すことにより、計算された ARICH のずれの大きさは先行研究の手法によって計算されたものよりも信用できる値となり、それに伴い  $\chi^2$  の値も先行研究の手法で計算された場合の 1/70 程度まで小さくなった。この結果 ARICH の分解能はアライメント前後で約 6.5 % 向上した。

# 目次

第 1 章	序論	7
1.1	素粒子標準模型と $B$ 中間子系の物理	7
1.1.1	素粒子標準模型	7
1.1.2	$B$ 中間子系の物理	8
1.1.3	CP 対称性の破れ	10
1.1.4	Belle II 実験で期待される新物理	13
1.2	本研究の概要	15
第 2 章	Belle II 実験	16
2.1	Belle 実験から Belle II 実験へ	16
2.2	SuperKEKB 加速器	17
2.2.1	概要	17
2.2.2	性能	18
2.3	Belle II 測定器	20
2.3.1	PXD, SVD	22
2.3.2	CDC	23
2.3.3	TOP	24
2.3.4	ARICH	26
2.3.5	ECL	26
2.3.6	KLM	27
第 3 章	ARICH	29
3.1	チェレンコフ光	29
3.2	ARICH 検出器の粒子識別原理	30
3.3	ARICH 検出器の構造	32
3.3.1	輻射体シリカエアロゲル	32
3.3.2	光検出器 HAPD	33
3.4	ARICH 検出器への要求性能	35
第 4 章	ARICH 検出器のアライメント手法	36
4.1	アライメントの概要	36

4.2	先行研究 . . . . .	37
4.3	先行研究の問題点と改良 . . . . .	37
4.4	ARICH 検出器をずらした時のシミュレーション . . . . .	38
	4.4.1 ARICH 検出器の並進方向のずれのシミュレーション . . . . .	40
	4.4.2 ARICH 検出器の回転方向へのずれのシミュレーション . . . . .	47
4.5	ARICH 検出器の位置のずれの求め方の詳細 . . . . .	55
<b>第 5 章</b>	<b>ARICH 検出器のずれの見積もりとアライメント</b>	<b>56</b>
5.1	使用した実データに関して . . . . .	56
5.2	実データを用いたずれの算出結果 . . . . .	59
5.3	結果の改良の余地 . . . . .	63
<b>第 6 章</b>	<b>ARICH 検出器のアライメント手法の改良とその結果</b>	<b>65</b>
6.1	アライメントフィットの結果の精度向上 . . . . .	65
6.2	得られたずれの値を用いた実データへのアライメントの結果 . . . . .	73
6.3	アライメントの結果の妥当性 . . . . .	75
<b>第 7 章</b>	<b>結論</b>	<b>78</b>
	謝辞	80
	付録 A	81
	付録 B	87
	参考文献	96

# 目次

1.1	素粒子標準模型 [1]	8
1.2	$\gamma(4S)$ 中間子の崩壊例	9
1.3	$B^0 - \bar{B}^0$ 混合 (mixing) の様子を表す Feynman diagram	9
1.4	(上段) $B^0$ が直接終状態に遷移した場合と、(下段) $B^0$ が $\bar{B}^0$ へと遷移して終状態に遷移した場合の Fynman diagram	10
1.5	ユニタリー三角形	11
1.6	様々な実験結果によるユニタリー三角形成立のためのパラメータ制限条件のプロット [5]	12
1.7	各実験における $\sin 2\phi_1$ の値と世界平均 [4] (2018 年)	13
1.8	(左) 標準模型における $b \rightarrow s\gamma$ 遷移、(右) 新物理で期待される $b \rightarrow s\gamma$ 遷移	14
1.9	$X(3872)$ の質量領域における $B^\pm \rightarrow K^\pm \pi^+ \pi^- J/\psi$ 崩壊中の $\pi^+ \pi^- J/\psi$ 不変質量分布 [7]	15
2.1	Belle 実験及び Belle II 実験におけるピークルミノシティ [9]	17
2.2	Super KEKB 加速器の全体図 [10]	18
2.3	ナノビーム大角度交差衝突方式	19
2.4	Belle II 測定器 [11]	20
2.5	Belle II 測定器の断面図 [12]	21
2.6	PXD, CVD の概略図 [12]	22
2.7	PXD の概略図	22
2.8	PXD, CVD に搭載されたセンサーの断面図 [13]	23
2.9	CDC におけるセンスワイヤーの配置図 [12]	23
2.10	シミュレーションによる CDC で測定された $dE/dx$ と $p$ の関係 [8]	24
2.11	TOP の外観図 [13]	25
2.12	TOP の粒子識別の概念図 [13]	25
2.13	ARICH の外観図	26
2.14	ECL の構成図 [12]	27
2.15	KLM の外観図	28
2.16	KLM 内の検出器の概略図 [13]	28
3.1	チェレンコフ光の発生イメージ図	30
3.2	ARICH の識別原理	31

3.3	荷電 $K/\pi$ 中間子の放射角分布	31
3.4	ARICH の各構成要素とその配置の概略図	32
3.5	(左) 単層方式、(右) デュアルレイヤー方式	33
3.6	(左) エアロゲルを 1 枚使用した場合、(右) エアロゲルを 2 枚使用した場合における ARICH の角度分解能と平均検出光子数	33
3.7	光検出器 HAPD	34
3.8	光検出器 HAPD の動作原理	34
3.9	APD の動作原理	34
4.1	(左) ARICH が正しい位置にあるとき、(右) ARICH がずれたとき	36
4.2	ずれの表現における ARICH の座標軸の定義	38
4.3	ARICH のアライメントに用いる変数の定義	39
4.4	ARICH が正しい位置にある場合の $n \cos \theta_c$ の $\phi_{ch}$ 依存性	40
4.5	36 領域に分割された ARICH	40
4.6	ARICH の並進方向 $x$ 軸 2 mm ずらしたときのシミュレーション結果のフィット	41
4.7	ARICH の並進方向 $x$ 軸 -2 mm ずらしたときのシミュレーション結果のフィット	42
4.8	ARICH の並進方向 $y$ 軸 2 mm ずらしたときのシミュレーション結果のフィット	43
4.9	ARICH の並進方向 $y$ 軸 -2 mm ずらしたときのシミュレーション結果のフィット	44
4.10	ARICH の並進方向 $z$ 軸 5 mm ずらしたときのシミュレーション結果のフィット	45
4.11	ARICH の並進方向 $z$ 軸 -5 mm ずらしたときのシミュレーション結果のフィット	46
4.12	ARICH 並進方向 $x$ 軸、 $y$ 軸、 $z$ 軸に対する正の仮想的ずれのシミュレーション結果 の矢印表示	46
4.13	ARICH 並進方向 $x$ 軸、 $y$ 軸、 $z$ 軸に対する負の仮想的ずれのシミュレーション結果 の矢印表示	47
4.14	先行研究における ARICH 並進方向 $x$ 軸、 $y$ 軸、 $z$ 軸に対する仮想的ずれのシミュ レーション結果の矢印表示 [14]	47
4.15	ARICH の回転方向 $x$ 軸 $0.5^\circ$ ずらしたときのシミュレーション結果のフィット	48
4.16	ARICH の回転方向 $x$ 軸 $-0.5^\circ$ ずらしたときのシミュレーション結果のフィット	49
4.17	ARICH の回転方向 $y$ 軸 $0.5^\circ$ ずらしたときのシミュレーション結果のフィット	50
4.18	ARICH の回転方向 $y$ 軸 $-0.5^\circ$ ずらしたときのシミュレーション結果のフィット	51
4.19	ARICH の回転方向 $z$ 軸 $0.5^\circ$ ずらしたときのシミュレーション結果のフィット	52
4.20	ARICH の回転方向 $z$ 軸 $-0.5^\circ$ ずらしたときのシミュレーション結果のフィット	53
4.21	ARICH の回転方向 $x$ 軸、 $y$ 軸、 $z$ 軸への正の仮想的ずれのシミュレーション結果の 矢印表示	54
4.22	ARICH の回転方向 $x$ 軸、 $y$ 軸、 $z$ 軸への負の仮想的ずれのシミュレーション結果の 矢印表示	54
4.23	先行研究による回転方向 $x$ 軸、 $y$ 軸、 $z$ 軸の仮想的ずれのシミュレーション結果 [14]	55
5.1	Bucket 15 で取得されたデータの $n \cos \theta_c$	57

5.2	Bucket 15 の実データにおける (左) $n \cos \theta_c$ の $\phi_{ch}$ 依存性の二次元ヒストグラムと (右) そのフィット結果 . . . . .	58
5.3	実データにおいて 36 領域に分割された ARICH における $n \cos \theta_c$ の $\phi_{ch}$ 依存性 . . . . .	59
5.4	36 領域に分割された実データのフィット結果 . . . . .	60
5.5	ARICH 各領域におけるデータ点とずれの値から再構成されたずれの比較 . . . . .	61
5.6	フィット関数の振幅と位相によって表現された ARICH のずれの矢印表示 . . . . .	62
5.7	7 個のパラメータによって表現された ARICH のずれの矢印表示 . . . . .	63
6.1	並進方向 $z$ 軸における $t_z$ と $f$ の関係 . . . . .	66
6.2	ARICH を並進方向 $z$ 軸に 5 mm ずらしたときのモンテカルロシミュレーションによる $f(t_z), \Delta f(t_z)$ のプロット . . . . .	67
6.3	ARICH にずれが無い場合のモンテカルロシミュレーションによる $f(t_z), \Delta f(t_z)$ のプロット プロット . . . . .	68
6.4	プログラム (1) のフローチャート . . . . .	69
6.5	プログラム (2) のフローチャート . . . . .	70
6.6	プログラム (1) を用いた実データの領域 1 におけるデータ点のフィット結果 . . . . .	71
6.7	プログラム (1) を用いた実データのフィット結果 . . . . .	72
6.8	新しいフィット手法を用いた場合の ARICH 各領域におけるデータ点とずれの値から 再構成されたずれの比較 . . . . .	73
6.9	(上) アライメント結果適応前、(下) アライメント結果適応後の $n \cos \theta_c$ の $\phi_{ch}$ 依存性	74
6.10	アライメント前の $\theta_c$ 分布 . . . . .	75
6.11	アライメント後の $\theta_c$ 分布 . . . . .	75
6.12	新たな手法によるシミュレーションで生成されたデータ点とずれの値から再構成された ずれの比較 . . . . .	76



# 表目次

2.1	Super KEKB 加速器と KEKB 加速器における各パラメータの比較 (LER/HER) . . . . .	19
4.1	先行研究による ARICH のずれの値 . . . . .	37
5.1	Bucket 15 を用いた場合の ARICH のずれの値 . . . . .	62
6.1	新たな手法を使った Bucket 15 を用いた場合の ARICH のずれの値 . . . . .	73
6.2	入力値としたアライメントフィットの結果 . . . . .	75
6.3	新たな手法にシミュレーションを用いた場合のアライメントフィットの結果 . . . . .	76

# 第 1 章

## 序論

我々が生活を送るこの宇宙の全ての物質は、細かく分割をしていくうちに最終的にはそれ以上細かくなれないような最小単位に分けることができる。現代の物理学では、この物質の最小単位は素粒子と呼ばれており、20 世紀から始まり今なお世界中の物理学者たちの手によって、これら素粒子がどのような振る舞いをするかということが研究されている。

時代の進歩と共に加速器の性能や実験技術も大きな発展を遂げ、そのような素粒子の性質や振る舞いを徐々にまとめて体系化することが可能となり、今日ではこれは素粒子標準模型と呼ばれている。この模型を用いることにより、これまで行われてきたありとあらゆる素粒子実験の結果のほとんどをうまく説明することができる一方で、いくつかの知られている事実にはどうしても矛盾が生じてしまうという点も存在する。この章では、そんな素粒子標準模型についてと、本研究に関連する  $B$  中間子系の物理を中心に述べる。

### 1.1 素粒子標準模型と $B$ 中間子系の物理

#### 1.1.1 素粒子標準模型

素粒子標準模型 (図 1.1) は、現在の素粒子物理学において確立されている素粒子に関する標準理論であり、素粒子の性質や相互作用を記述するためのモデルのひとつである。これまでのほとんどの素粒子実験の結果を矛盾無く説明することができ、いくつか存在する素粒子の振る舞いについて記述されたモデルの内でも最も用いられる機会が多い模型である。図 1.1 に標準模型に組み込まれている素粒子の一覧を示す。この模型に組み込まれている素粒子には大きく分けてクォークとレプトンの 2 種類があり、それぞれクォークは  $u, d, c, s, t, b$  の 6 種類に分類され、レプトンは電子、 $\mu$  粒子、 $\tau$  粒子、電子ニュートリノ、 $\mu$  ニュートリノ、 $\tau$  ニュートリノの 6 種類にさらに細かく分類される。さらに、これらの物質粒子とは別に強い相互作用、弱い相互作用、電磁相互作用の三つの相互作用を媒介する粒子や、物質に質量を与えるヒッグス粒子がここに組み込まれている。(重力相互作用を媒介する粒子は含まれず)

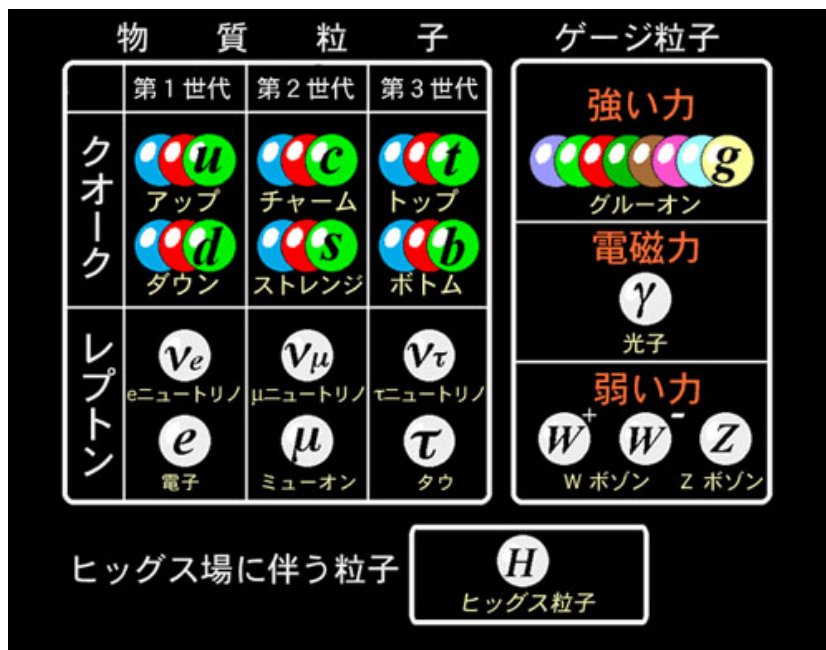


図 1.1 素粒子標準模型 [1]

### 1.1.2 B 中間子系の物理

B 中間子は  $b$  クォークの反粒子である  $\bar{b}$  クォークとそれぞれ  $u, d, s, c$  クォークの組み合わせからなる中間子である。これらの B 中間子のうち本研究で主に着目される  $B^0$  中間子は  $d, \bar{b}$  クォークで構成され、その反粒子として  $\bar{d}, b$  クォークで構成された  $\bar{B}^0$  中間子が存在する。Belle II 実験では、Super KEKB 加速器によってそれぞれ 7 GeV と 4 GeV に加速された電子と陽電子が衝突した際に生成される大量の B 中間子対を観測するが、これは衝突の際に生成される質量  $10.58 \text{ GeV}/c^2$  の  $\gamma(4S)$  中間子の共鳴状態のうちほとんど全てが B 中間子に崩壊する (約 48.6 % が  $B^0\bar{B}^0$ 、約 51.4 % が  $B^+B^-$  へ崩壊) ことを用いている。図 1.2 に  $\gamma(4S)$  中間子の崩壊例を示す。

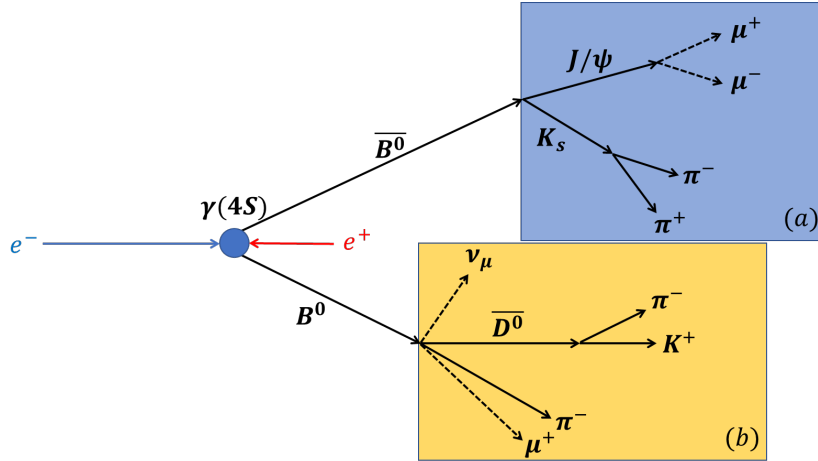


図 1.2  $\gamma(4S)$  中間子の崩壊例

$B^0$  中間子とその反粒子である  $\bar{B}^0$  中間子のペアは  $B^0 - \bar{B}^0$  混合 (mixing) により常に入れ替わっている。この過程における Feynman diagram を図 1.3 に示す。

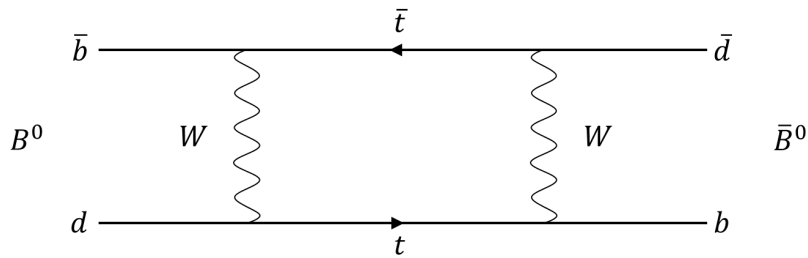


図 1.3  $B^0 - \bar{B}^0$  混合 (mixing) の様子を表す Feynman diagram

生成した  $B$  中間子対はさらに連鎖的に崩壊を起こして終状態へ遷移するが、ここで例えば  $B^0$  中間子と  $\bar{B}^0$  中間子の両方から崩壊する可能性のある状態である  $J/\psi$  中間子と  $K_s$  中間子への崩壊を考える。ここでこの反応における生成物である  $K_s$  中間子は  $K^0, \bar{K}^0$  中間子のペアが図 1.3 で述べたような  $K^0 - \bar{K}^0$  混合を起こし CP 固有状態になったものである。 $J/\psi K_s$  への崩壊は図 1.4 に述べるような 2 種類の経路が考えられ、直接終状態へ遷移する場合 (図 1.4 上段) と、 $B^0 - \bar{B}^0$  混合 (mixing) により  $\bar{B}^0$  中間子に遷移したものがさらに遷移し終状態へ達する場合 (図 1.4 下段) に分かれるが、この 2 種類の経路は  $K^0$  中間子とその反粒子である  $\bar{K}^0$  が同じ  $K_s$  中間子として崩壊するため終状態が同じとなり、終状態からその親粒子を区別することが出来ない。そこで、Belle II 実験では本来観測対象である  $J/\psi K_s$  に崩壊した側 (signal side、図 1.2 における (a) の側) でないもう一方の崩壊側 (tag side、図 1.2 における (b) の側) の崩壊を観測し、 $K$  中間子やレプトンの電荷を調べることで tag side の親粒子を求めている。この際、 $\gamma(4S)$  中間子の崩壊で生じた  $B$  中間子対が同時に  $B^0, B^0$  もしくは  $\bar{B}^0, \bar{B}^0$  のペアとなることは角運動量保存などの観点から禁止されているため、ある時刻において観測対象である signal side の親粒子が  $B^0$  であるか  $\bar{B}^0$  であるかを求めることができる。これは図 1.4 の Feynman diagram の左端の状態が  $B^0$  であるか  $\bar{B}^0$  であるかを特定することに対応する。このような

方法をフレーバータギングという。

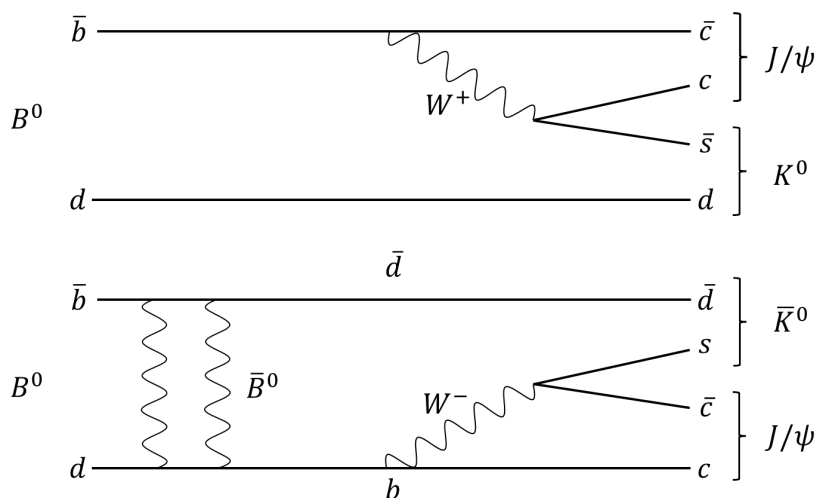


図 1.4 (上段)  $B^0$  が直接終状態に遷移した場合と、(下段)  $B^0$  が  $\bar{B}^0$  へと遷移して終状態に遷移した場合の Fynman diagram

### 1.1.3 CP 対称性の破れ

CP 対称性の破れとは荷電共役変換 (C)、鏡像変換 (P) を組み合わせた粒子と反粒子の変換の対称性の破れを表しており、我々が生活する宇宙が物質優勢であり反物質がほとんど存在しないのはなぜかという現代物理学における未解決問題を説明するのに必要な 3 条件のうちの一つである。

この CP 対称性の破れは、1964 年に J.W.Cronin、V.L.Fitch らによって  $s$  クォークを含む  $K_L^0$  中間子の崩壊を調べることによって世界で初めて観測された [2]。さらにその後 1973 年に小林誠・益川敏英両氏がクォークの世代を 3 世代 (6 種類) 以上存在することを仮定し、クォークにおける質量固有状態とフレーバ固有状態との変換行列である Cabibbo-Kobayashi-Maskawa (CKM) 行列中に複素位相が一つ以上含まれていれば CP 対称性が破れているということを示した [3]。式 (1.1) に CKM 行列の変換式を示す。

$$\begin{pmatrix} d' \\ s' \\ b' \end{pmatrix} = \begin{pmatrix} V_{ud} & V_{us} & V_{ub} \\ V_{cd} & V_{cs} & V_{cb} \\ V_{td} & V_{ts} & V_{tb} \end{pmatrix} \begin{pmatrix} d \\ s \\ b \end{pmatrix} \quad (1.1)$$

ここで  $(d, s, b)$ ,  $(d', s', b')$  はそれぞれクォークの質量固有状態、クォークのフレーバ固有状態 ( $W$  ボソンが相互作用している状態) を表現しており、式 (1.1) は弱い相互作用によって  $(u, c)$  クォークへと崩壊するクォークは  $(d, s, b)$  クォークの重ね合わせの状態になっていることを示している。

さらに CKM 行列は混合角  $\theta_{ij}$  ( $i, j = 1, 2, 3$ ) と CP を破る複素位相 (CP 位相)  $\delta_{13}$  を用いて式 (1.2) と表される。なおここでは  $c_{ij} = \cos \theta_{ij}$ ,  $s_{ij} = \sin \theta_{ij}$  と略記している。

$$\begin{pmatrix} V_{ud} & V_{us} & V_{ub} \\ V_{cd} & V_{cs} & V_{cb} \\ V_{td} & V_{ts} & V_{tb} \end{pmatrix} = \begin{pmatrix} c_{12}c_{13} & s_{12}c_{13} & s_{13}e^{-i\delta_{13}} \\ -s_{12}c_{13} - c_{12}s_{23}s_{13}e^{i\delta_{13}} & c_{12}s_{23}s_{13} - s_{12}s_{23}s_{13}e^{i\delta_{13}} & s_{23}c_{13} \\ s_{12}s_{23} - c_{12}c_{23}s_{23}e^{i\delta_{13}} & -s_{22}c_{12} - s_{12}c_{23}s_{13}e^{i\delta_{13}} & c_{23}c_{13} \end{pmatrix} \quad (1.2)$$

また  $\lambda = \sin \theta_{12}$ 、 $A\lambda^2 = \sin \theta_{23}$ 、 $A\lambda^3 = \sin \theta_{13}e^{-i\delta_{13}}$  とおくと近似的に式 (1.3) が成り立つ [6]。

$$\begin{pmatrix} V_{ud} & V_{us} & V_{ub} \\ V_{cd} & V_{cs} & V_{cb} \\ V_{td} & V_{ts} & V_{tb} \end{pmatrix} \sim \begin{pmatrix} 1 - \frac{\lambda^2}{2} & \lambda & A\lambda^3(\rho - i\eta) \\ -\lambda & 1 - \frac{\lambda^2}{2} & A\lambda^2 \\ A\lambda^3(1 - \rho - i\eta) & A\lambda^2 & 1 \end{pmatrix} \quad (1.3)$$

さらに CKM 行列はユニタリ行列であるため、行列  $V$  の随伴行列を  $V^*$  として式 (1.4) が成立し、式 (1.4) は複素数平面上で図 1.5 のように表すことができる。

$$V_{tb}^* V_{td} + V_{ub}^* V_{ud} + V_{cb}^* V_{cd} = 0 \quad (1.4)$$

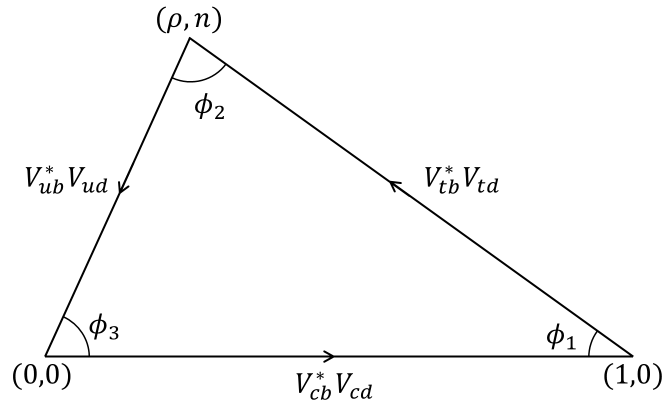


図 1.5 ユニタリー三角形

図 1.5 のユニタリー三角形における内角  $\phi_1, \phi_2, \phi_3$  の値が 0 でなく、なおかつ内角の総和が  $180^\circ$  のとき、つまり三角形の成立条件が図 1.5 の三角形に当てはまるとき 3 世代のクォーク混合により CP 対称性が破れているということがいえる。図 1.6 に、様々な実験により算出された CKM 行列におけるユニタリー三角形の成立条件のためのパラメータの制限条件をプロットしたものを示す。

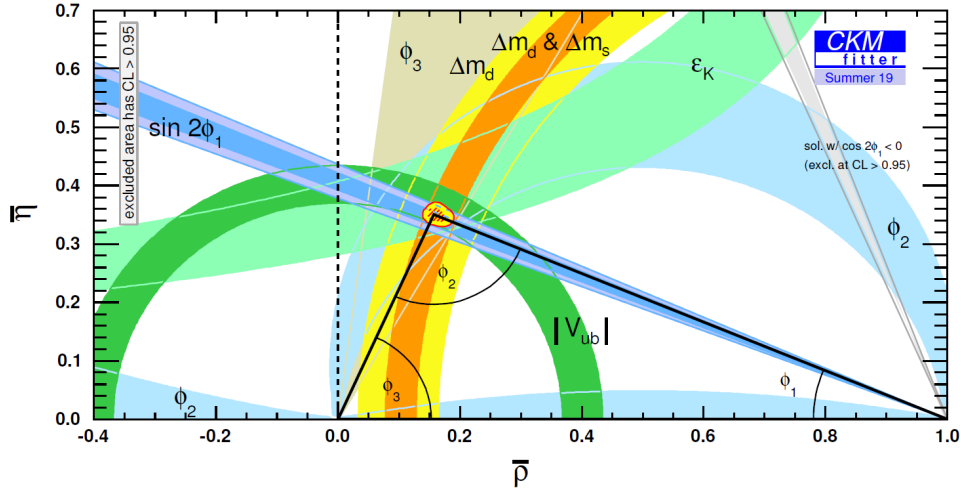


図 1.6 様々な実験結果によるユニタリー三角形成立のためのパラメータ制限条件のプロット [5]

これらの実験の中でも特に Belle II 実験の前身実験である Belle 実験と SLAC (Stanford Linear Accelerator Center) において PEP-II 加速器を用いた BaBar 実験では、節 1.1.2 で述べたように観測された  $B$  中間子が  $B^0, \bar{B}^0$  のどちらであるかを特定し、さらにそれらの崩壊時刻を比較することによって時間に依存した CP 対称性の破れを 2001 年にほぼ同時に発見した。以下では具体的な CP 対称性の破れに関する式について述べる。

まず  $B^0$  中間子と  $\bar{B}^0$  中間子の崩壊分岐比を求めると式 (1.5) で表される。

$$\Gamma(B^0 \bar{B}^0 \text{ pair} \rightarrow f_{\text{sig}}, f_{\text{tag}}) = \frac{e^{-\frac{|\Delta t|}{\tau}}}{\tau} (1 + q(S \sin(\Delta m \Delta t)) + A \cos(\Delta m \Delta t)) \quad (1.5)$$

なお、signal side と tag side 間での崩壊時間差を  $\Delta t = f_{\text{sig}} - f_{\text{tag}}$ 、 $B$  中間子のフレーバーを  $q$ 、 $B$  中間子の寿命を  $\tau$ 、 $B^0$  と  $\bar{B}^0$  の質量差を  $\Delta m$  とする。 $S, A$  はそれぞれ  $B^0 - \bar{B}^0$  混合における間接的 CP 対称性の破れと直接的 CP 対称性の破れに対応するパラメータであり、標準模型におけるこのモード ( $b \rightarrow c\bar{c}s$ ) の場合は  $S = -\xi_f \sin 2\phi_1$ 、 $A = 0$  となる。また、 $J/\psi K_{S/L}^0$  へ崩壊するときの CP 非対称度  $A_{CP}$  は式 (1.6) で与えられる。

$$A_{CP}(\Delta t) = \frac{\Gamma(\bar{B}^0 \rightarrow f_{CP}) - \Gamma(B^0(t) \rightarrow f_{CP})}{\Gamma(\bar{B}^0 \rightarrow f_{CP}) + \Gamma(B^0(t) \rightarrow f_{CP})} = -\xi_{CP} \sin 2\phi_1 \sin(\Delta m \Delta t) \quad (1.6)$$

式 (1.6) を見ると  $A_{CP}$  の値は  $\sin 2\phi_1$  の値に比例するため、現在も世界中にある様々な実験グループがこの  $\sin 2\phi_1$  の値を測定することで CP 非対称性の大きさを調べている。図 1.7 に  $\sin 2\phi_1$  の値の世界平均の値を示す。

$$\sin(2\beta) \equiv \sin(2\phi_1)$$

**HFLAV**  
Moriond 2018  
PRELIMINARY

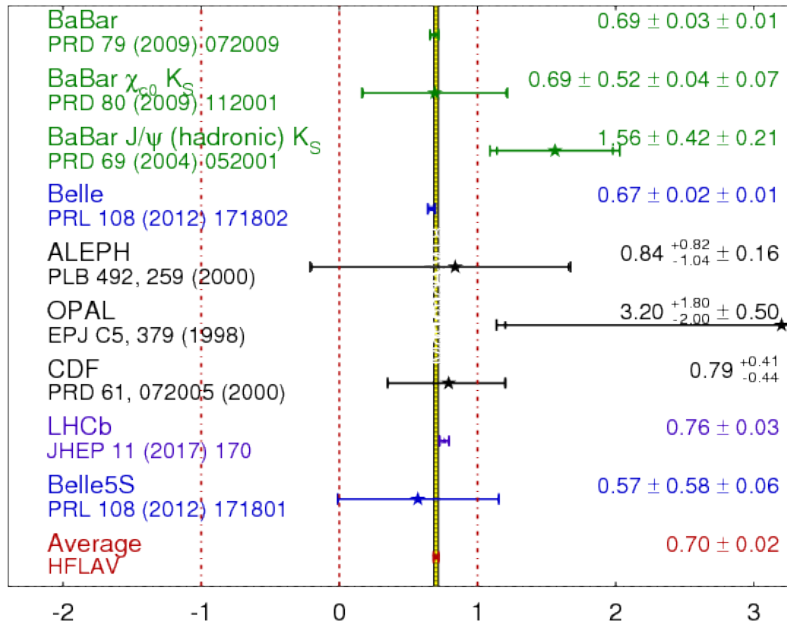


図 1.7 各実験における  $\sin 2\phi_1$  の値と世界平均 [4] (2018 年)

### 1.1.4 Belle II 実験で期待される新物理

標準模型を超える新しい物理事象の探索をする実験には、大きく分けて以下の (I). と (II). の 2 通りの方法が用いられている。

- (I). 加速器を用いて高いエネルギー (TeV = 1000 GeV 程度) 状態の粒子を作り出し、このような非常に高いエネルギー領域での粒子の生成や崩壊を直接観測することにより未知の質量を持つ新粒子の発見を目指す方法。このような方法による実験をエネルギーフロンティア実験と呼ぶ。エネルギーフロンティア実験に代表されるのはスイスの欧州原子核研究機構 (CERN) にある大型ハドロン衝突型加速器 (LHC 加速器) を用いた LHC 実験などがある。
- (II). 加速器を用いて GeV 程度のエネルギー状態の粒子を作り出し、それら粒子の崩壊過程に含まれる新粒子の寄与を観測することによって標準模型からのずれや、標準模型からは起こり得ない粒子の崩壊の観測を目指す方法。エネルギーフロンティア実験と対比してこのような実験をルミノシティフロンティア実験と呼ぶ。

Belle II 実験は (II). のルミノシティフロンティア実験に分類され、 $B$  中間子対に関する膨大な統計データと前身実験からアップグレードされた加速器・測定器を用いて新粒子の寄与を探索している。以下では Belle II 実験に期待される新物理事象について述べる。



### $b \rightarrow s\gamma$ 崩壊

$b \rightarrow s\gamma$  はフェルミオンの電荷が変化せずフレーバーのみが変化するような崩壊過程である。このような過程は Flavor-Changing Neutral Current (FCNC) と呼ばれ、標準模型の範囲内では tree diagram のみでの崩壊は禁止されており、loop diagram でのみ崩壊する。この際 loop を介した遷移によってウィークボソン  $W$  が現れるのに対し、標準模型を超えるような新しい物理事象として loop を介した遷移を通して未知の粒子である荷電ヒッグス粒子  $H^\pm$  が現れることが期待されており (図 1.8)、その場合崩壊過程に含まれる新粒子の寄与による従来の崩壊分岐比の修正が必要になる。Belle II 実験では終状態に  $K/\pi$  中間子をよく含む  $b \rightarrow s\gamma$  崩壊を詳細に調べることにより、 $H^\pm$  や超対称性理論によって存在が予言されている超対称性粒子 (SUSY 粒子) などの新粒子の発見を目指している。

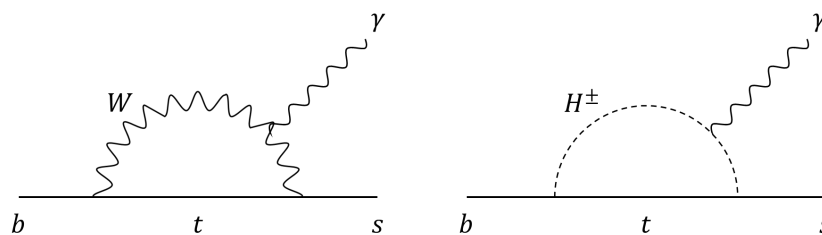


図 1.8 (左) 標準模型における  $b \rightarrow s\gamma$  遷移、(右) 新物理で期待される  $b \rightarrow s\gamma$  遷移

### エキゾチックハドロン

現在までに確認されているハドロンは全て三つのクォークとグルーオンが強い相互作用で結びついたバリオンと、クォークと反クォークのそれぞれ一つずつがグルーオンで結びついた中間子のいずれかに分類される。しかし量子色力学における理論中ではこの分類に当てはまらない新粒子の存在が予言されており、これらをエキゾチックハドロン (異種ハドロン) という。具体的には二つのクォーク、二つの反クォークとグルーオンが結びついたテトラクォーク、またグルーオンが同時に複数結合し全体として無色な状態になっているとされるグルーボールなどが挙げられる。

エキゾチックハドロンの存在は長く議論され多くの探索実験が行われてきたが、2003 年 7 月に Belle 実験が世界で初めてエキゾチックハドロンと思われる新粒子  $X(3872)$  を発見した [7]。実験で測定された  $X(3872)$  の質量は  $3872.0 \pm 0.6 \pm 0.5 \text{ MeV}/c^2$  であり、これは  $c$  クォークと  $\bar{u}$  クォークで構成された  $D^0$  中間子とその反粒子である  $D^{*0}$  中間子の質量和  $3871.3 \pm 1.0 \text{ MeV}/c^2$  に極めて近く、そこから  $X(3872)$  はテトラクォークではないかと予想されている。図 1.9 に  $X(3872)$  が発見された質量領域における  $B^\pm \rightarrow K^\pm \pi^+ \pi^- J/\psi$  崩壊中の  $\pi^+ \pi^- J/\psi$  不変質量分布を示す。

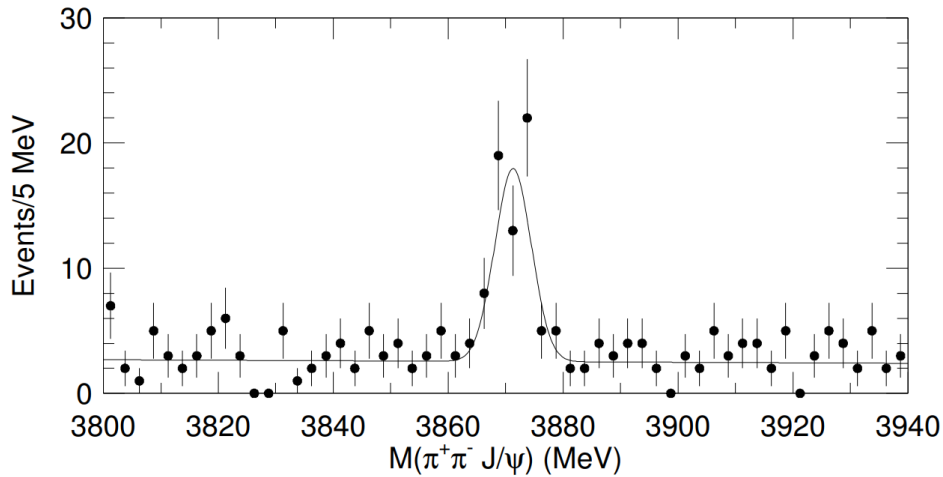


図 1.9  $X(3872)$  の質量領域における  $B^\pm \rightarrow K^\pm \pi^+ \pi^- J/\psi$  崩壊中の  $\pi^+ \pi^- J/\psi$  不変質量分布 [7]

Belle 実験は後に  $X(3872)$  以外にもいくつかのエキゾチックハドロンの候補となる新粒子を発見しており、後継実験である Belle II 実験は統計量の増加や加速器のアップグレードに伴いより多くの新粒子発見の可能性を秘めている。

## 1.2 本研究の概要

本章で述べているように、Belle II 実験において  $B$  中間子の稀崩壊を観測するという事は新しい物理の探索にも繋がるため、 $B$  中間子の崩壊生成物である荷電  $K$  中間子と荷電  $\pi$  中間子を高精度で分離する必要がある。Belle II 測定器に搭載されている Aerogel Ring Imaging CHerenkov counter (通称 ARICH) は、この 2 粒子を識別するという役割を担っているが、粒子識別の際に飛跡検出器から取得される粒子の飛跡情報を使用しているため、二つの検出器間の相対的な位置の精度は ARICH の性能に大きく依存する。本研究では、ARICH が最大限にその性能を発揮し、観測データの精度を向上させるという目的でアライメントを行った。また、類似した内容で ARICH のアライメントを行った先行研究の結果に関して、何点か問題点と改良の余地が存在したため、本研究ではアライメントの手法の改良に取り組み、改良を施した結果 ARICH のアライメントが定量的にどう変化したかについてを述べる。

## 第 2 章

# Belle II 実験

Belle II 実験は、Belle 実験の後継実験として始まった  $B$  中間子対の稀崩壊を通して標準模型では説明できないような新たな物理を探索することを目的とした電子・陽電子衝突実験である。Belle 実験から得られた解析データを参考に大型のアップグレードを実施し、実験によって取得されたデータ量を示す積分ルミノシティを Belle 実験の約  $1 \text{ ab}^{-1}$  から最終的に  $50 \text{ ab}^{-1}$  まで増やすことを目標としている。本章ではこの Belle II 実験についてを詳しく述べる。

### 2.1 Belle 実験から Belle II 実験へ

Belle II 実験の前身実験である Belle 実験は、茨城県つくば市にある高エネルギー加速器研究機構 (KEK) において 1999 年から 2010 年まで運用された電子と陽電子の衝突実験である。KEKB 加速器によって電子ビームは  $8 \text{ GeV}$  に、陽電子ビームは  $3.5 \text{ GeV}$  に加速され、重心系エネルギー  $10.58 \text{ GeV}$  で衝突した二つのビームは大量の  $B$  中間子対を生成しそれらの崩壊過程を Belle 測定器で詳細に測定した。

Belle 実験は 11 年間の運転を通して約 8 億個の  $B$  中間子対のデータを蓄積し、このデータの解析により  $B$  中間子とその反粒子である  $\bar{B}$  中間子間の CP 対称性の破れを発見した。この発見は、小林誠・益川敏英両氏が 1973 年に発表した、もしクォークが 3 世代 (6 種類) 以上存在し混合するならば、CP 対称性の破れを理論的に説明できるという小林・益川理論が正しいことを示し、両氏の 2008 年ノーベル物理学賞受賞に大いに貢献した。

Belle 実験の後継実験として発足した Belle II 実験は、大規模なアップグレードの末により高精度な測定を可能にした。新しくなった Super KEKB 加速器では、Belle 実験における前 KEKB 加速器の  $\frac{1}{20}$  に絞ったビームを衝突点で大角度交差させるというビーム方を採用し、さらにビーム電流を最大 2 倍に強化している。これにより、ビーム衝突性能を 40 倍に増加させることを目標としている。衝突型加速器の性能を示す指標はルミノシティと呼ばれるパラメータで表されるが、Belle 実験で 2009 年 6 月に得られたピークルミノシティ  $2.11 \times 10^{-34} \text{ cm}^{-2}\text{s}^{-1}$  に対して、Belle II 実験では 2021 年現在  $10 \times 10^{-34} \text{ cm}^{-2}\text{s}^{-1}$  程度の値を達成しており、最終的には設計値の  $80 \times 10^{-34} \text{ cm}^{-2}\text{s}^{-1}$  を記録することを目標としている。図 2.1 に Belle 実験及び Belle II 実験におけるピークルミノシティの変化を示す。

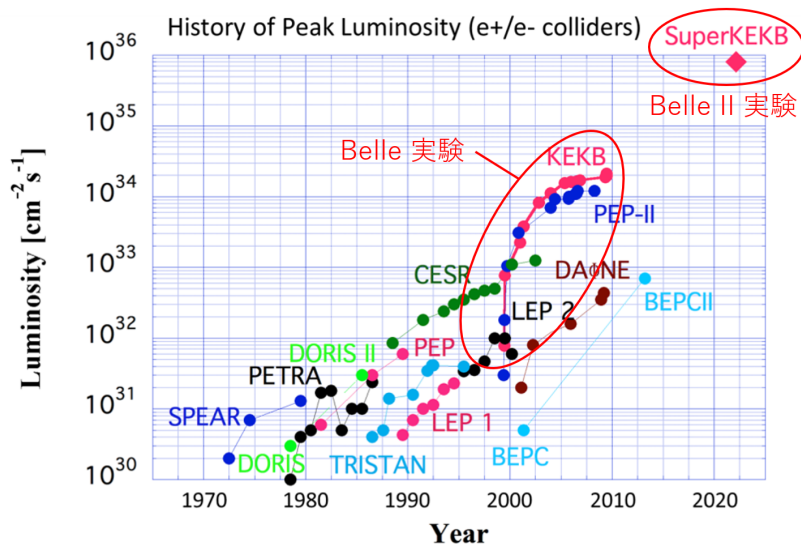


図 2.1 Belle 実験及び Belle II 実験におけるピーク luminositi [9]

## 2.2 SuperKEKB 加速器

### 2.2.1 概要

Super KEKB 加速器は茨城県つくば市の高エネルギー加速器研究機構 (KEK) 内部において地下 11 m に掘られた一周約 3 km、直径約 1 km のトンネル内に設置されており、電子と陽電子をそれぞれ 7 GeV と 4 GeV に加速させ、重心系エネルギー  $\sqrt{s} = 10.58$  GeV にて衝突させる非対称エネルギー衝突型加速器である。周長は約 3 km で、7 GeV 電子用高エネルギーリング (HER) と 4 GeV 陽電子用低エネルギーリング (LER) を持つ 2 リング型の加速器となっている。図 2.2 に Super KEKB 加速器の全体図を示す。

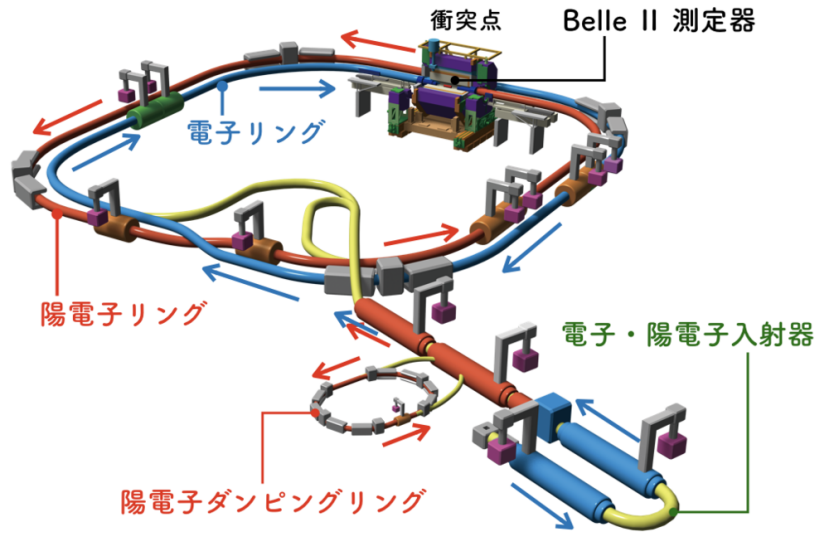


図 2.2 Super KEKB 加速器の全体図 [10]

## 2.2.2 性能

Belle II 実験のような加速器を用いた実験において、加速器の性能を表す重要な指標としてルミノシティがある。ビーム衝突型加速器において単位面積単位時間あたりに粒子が衝突した頻度を表す量であるルミノシティ  $\mathcal{L}$  は式 (2.1) で表される。

$$R = \mathcal{L}\sigma \quad (2.1)$$

ここで  $R$  は単位時間あたりに起きる素粒子反応の回数、 $\sigma$  は反応断面積を表す。さらに、ルミノシティ  $\mathcal{L}$  はビーム衝突型加速器のパラメータを用いると式 (2.2) と表すこともできる。

$$\mathcal{L} = \frac{\gamma_{\pm}}{2qr_e} \left( 1 + \frac{\sigma_y^*}{\sigma_x^*} \right) \frac{I_{\pm} \xi_{y\pm}}{\beta_{y\pm}^*} \left( \frac{R_L}{R_{\xi y}} \right) \quad (2.2)$$

$\gamma_{\pm}$  はローレンツ因子、 $q$  は素電荷、 $r_e$  は古典電子半径である。また、 $\frac{\sigma_y^*}{\sigma_x^*}$  は衝突点におけるビームサイズの  $x$  軸、 $y$  軸方向の大きさの比、 $I_{\pm}$  はビーム電流の大きさ、 $\xi_{y\pm}$  はビームビームパラメータと呼ばれるパラメータであり、衝突点において二つのビームがお互いに及ぼす力の大きさを表す値、 $\beta_{y\pm}^*$  は  $y$  方向衝突点におけるベータ関数で、ビームサイズを決定する際の絞り込みの大きさを表す値、 $\frac{R_L}{R_{\xi y}}$  はルミノシティ  $\mathcal{L}$  と  $\xi_{y\pm}$  の減少因子である。なお、式 (2.2) における変数  $\gamma_{\pm}$ 、 $I_{\pm}$ 、 $\xi_{y\pm}$ 、 $\beta_{y\pm}^*$  に付いている  $\pm$  の添え字はそれぞれ電子と陽電子を表している。

節 2.1 で述べた通り、Super KEKB 加速器は Belle 実験で用いられていた KEKB 加速器と比較して衝突点のビームサイズを  $\frac{1}{20}$  に絞ることにより式 (2.2) における  $\beta_{y\pm}^*$  の値を  $\frac{1}{20}$  に減少させ、ビーム電流  $I_{\pm}$  の値を約 2 倍に高めている。具体的には陽電子ビームを 3.6 A、電子ビームを 2.6 A にしてお

り、これによりビームのルミノシティは KEKB 加速器の 40 倍を目標とすることが出来るようになっていいる。ビームサイズとビーム電流の調節が加速器のルミノシティに影響を及ぼすことは確かだが、この調節を続けてもどこまでもルミノシティが向上する訳ではなく、ビームをある一定以上絞って衝突させるとビーム同士がお互いに及ぼし合う力が限界を超えてしまい、衝突によってビームサイズの増大やビームの寿命が短くなってしまふ。また、衝突点におけるビームサイズは小さすぎてもならず適切な値が存在する。衝突点におけるビームの大きさが小さすぎた場合、衝突点から離れた場所でのビームの大きさが急激に増大するという砂時計効果 (Hourglass 効果) が起こる。この問題への解決策として、Super KEKB 加速器の衝突点では、バンチと呼ばれるビームの塊に角度をつけて衝突領域が短くなるように交差させるようなナノビーム大角度交差衝突方式 (図 2.3) を採用している。

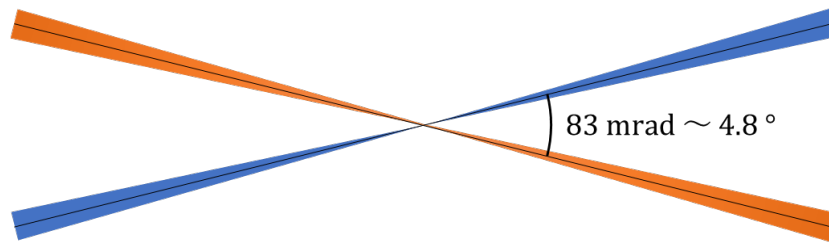


図 2.3 ナノビーム大角度交差衝突方式

ナノビーム大角度交差衝突方式を採用することによって砂時計効果の影響を無視することが出来るようになり、式 (2.2) における  $\frac{R_L}{R_{Ly}}$  の値を 1 と近似することができる。さらに、Super KEKB 加速器における電子ビームと陽電子ビームが両方フラットであり両ビームのサイズを  $x$  軸、 $y$  軸でほぼ等しいと仮定すると、フラットなビームにおける  $\frac{\sigma_y^*}{\sigma_x}$  の値は 1 と比較して十分小さいため式 (2.2) は最終的に式 (2.3) と近似して表すことができる。

$$\mathcal{L} \sim \frac{\gamma_{\pm}}{2qr_e} \left( \frac{I_{\pm} \xi_{y\pm}}{\beta_{y\pm}^*} \right) \quad (2.3)$$

表 2.1 に Super KEKB 加速器と KEKB 加速器における式 (2.3) に含まれる各パラメータの値の比較 (LER/HER) を示す。なお、KEKB 加速器のパラメータは最終到達値を示し Super KEKB 加速器のパラメータは設計値を示している。

表 2.1 Super KEKB 加速器と KEKB 加速器における各パラメータの比較 (LER/HER)

パラメータ	KEKB 加速器	Super KEKB 加速器
ビームエネルギー [GeV/c]	3.5/8.0	4.0/7.0
$\xi_{y\pm}$	0.129/0.090	0.0881/0.0807
$\beta_{y\pm}^*$ [mm]	5.9/5.9	0.270/0.300
$I_{\pm}$ [A]	1.64/1.19	3.6/2.6
$\mathcal{L}$ [ $10^{34} \text{cm}^2 \text{s}^{-1}$ ]	2.11	80

## 2.3 Belle II 測定器

Belle II 測定器 (図 2.4, 2.5) は、Super KEKB 加速器のビーム衝突点に設置された複合型の測定器である。様々な役割を担うために七つの異なる検出器から構成され、 $B$  中間子の崩壊によって生じた荷電粒子 (電子・ $\mu$  粒子・ $\pi$  粒子・ $K$  粒子・陽子・重陽子など) の運動量やエネルギー等を正確に測定する。本節では、この Belle II 測定器に搭載されているそれぞれの検出器の詳細について述べる。

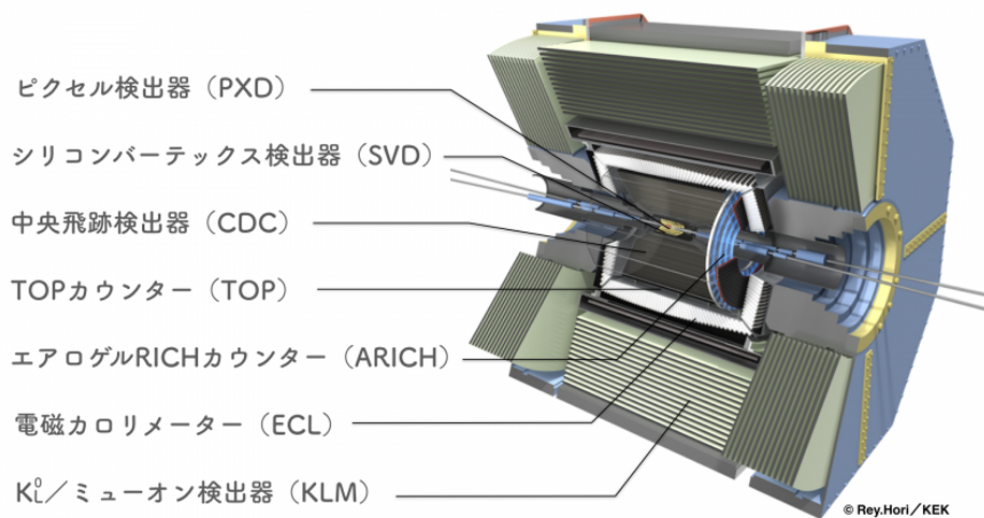


図 2.4 Belle II 測定器 [11]





### 2.3.1 PXD, SVD

Pixel Detector (PXD) と Silicon Vertex Detector (SVD) は共に、Belle II 測定器の中央部で  $B$  中間子の崩壊点を測定する検出器である。PXD と SVD を合わせて Vertex Detector (VXD) と呼ぶ場合もある。図 2.6 に PXD と SVD の概略図を示す。Belle II 測定器の最も内側のビームパイプを覆うように位置し、内側から 2 層の PXD と 4 層の SVD で構成される。

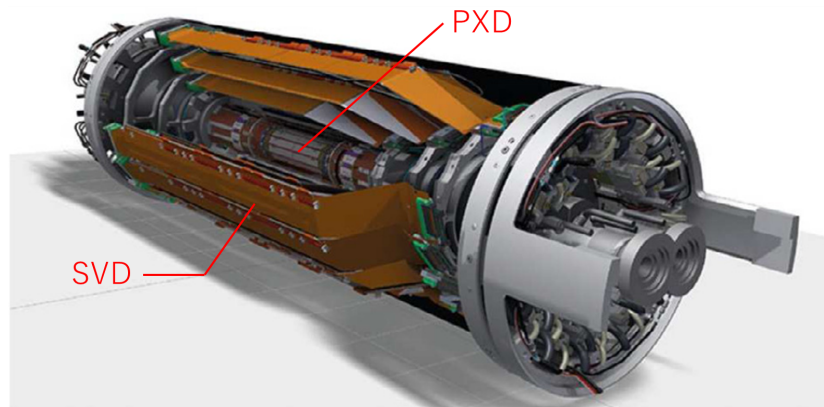


図 2.6 PXD, SVD の概略図 [12]

PXD (図 2.7) は 1 層目と 2 層目に位置し、半径方向 14 mm と 22 mm の場所に設置されている。センサーはピクセル化された半導体検出器 DEpleted P-channel Field Effect Transistor (DEPFET) で構成され、ストリップ型のセンサーと比較して占有率が低い、S/N 比が高い、ゴーストヒットが発生しないなどのメリットを有するため、衝突点近傍の高いビーム BG 下でも測定が可能である。図 2.8 に PXD と SVD にそれぞれ配置されたセンサーの断面図を示す。

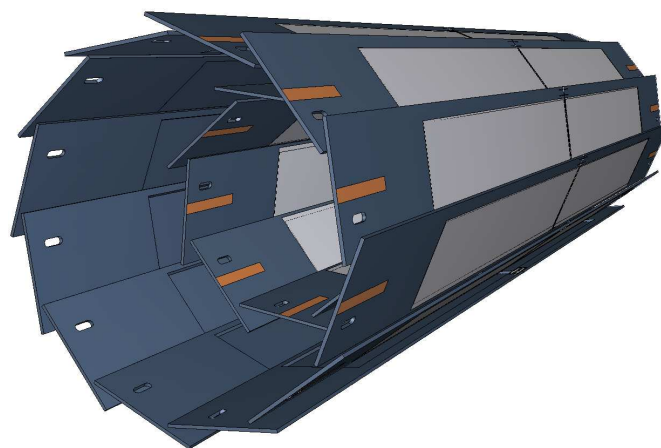


図 2.7 PXD の概略図

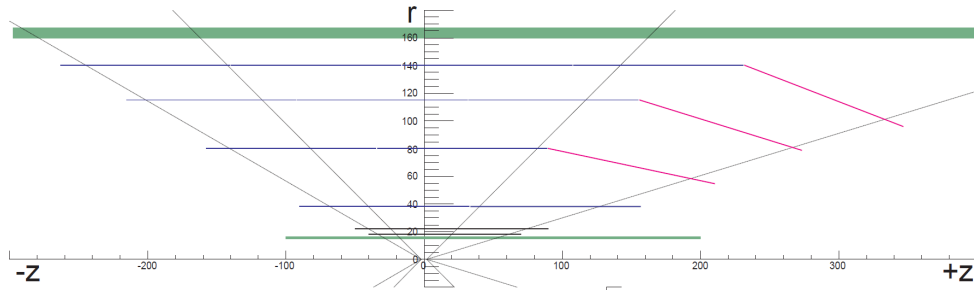


図 2.8 PXD, CVD に搭載されたセンサーの断面図 [13]

SVD は 3~6 層目に位置し、半径方向 39 mm、80 mm、104 mm、135 mm の場所に設置されている。Double-sided Silicon Detector (DSSD) 方式を用いており、p 型と n 型の半導体ストリップを裏表に直行させるように配置し、これを複数層使用することによって荷電粒子の 3 次元座標を得ている。

### 2.3.2 CDC

Central Drift Chamber (CDC) はエンドキャップ部に 14,336 本の信号を観測する役割を持つセンスワイヤーと、42,240 本の電場を生成する役割を持つフィールドワイヤーを中心軸方向まで張った円筒型のガスチェンバーである。半径は 16.0 - 113.0 cm、長さは 220 cm であり、内側にはヘリウムとエタンの混合ガス (He - C<sub>2</sub>H<sub>6</sub> ガス) が流れている。荷電粒子が CDC を通過すると、ガスから電離した電子がセンスワイヤーまでドリフトし、その時間を使ってワイヤーまでの距離を測定することにより正確に荷電粒子の飛跡を再構成する。これにより CDC では粒子の通過位置と運動量を測定しており、特にこの運動量情報は ARICH での粒子識別の際にも用いられている。図 2.9 に CDC に搭載されているワイヤーの内センスワイヤーの配置図を示す。なお図 2.9 中にある 2 種類のセンスワイヤーのうち、Axial wire はビーム軸に対して平行に張られているワイヤー、Stereo wire はビーム軸に対して斜めに張られているワイヤーを示している。

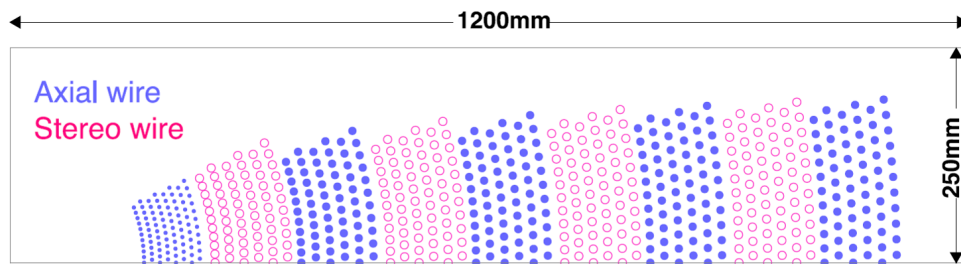


図 2.9 CDC におけるセンスワイヤーの配置図 [12]

さらに CDC は低運動領域における粒子の識別も行うことができ、測定された荷電粒子の情報から CDC 内でのエネルギー損失  $dE/dx$  を計算することで粒子を識別する。図 2.10 にシミュレーションによる CDC で測定された  $dE/dx$  と  $p$  の関係を示す。

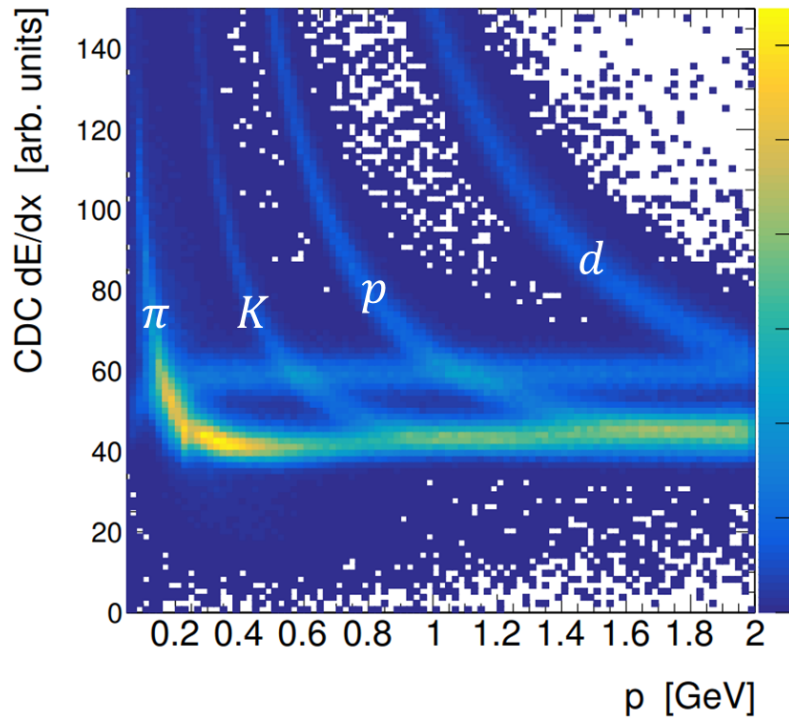


図 2.10 シミュレーションによる CDC で測定された  $dE/dx$  と  $p$  の関係 [8]

### 2.3.3 TOP

Time Of Propagation (TOP) (図 2.11) は Belle II 測定器のバレル部で荷電  $K/\pi$  中間子の識別を行う検出器である。CDC と ECL の間の動径方向 12 cm の空間に位置し、一つのモジュールに厚さ 20 mm、幅 450 mm、長さ 125 cm の石英版が 2 枚  $z$  方向に連なり、CDC の周りに 16 モジュール配置されている。この石英版に荷電粒子が通過する際にチェレンコフ光を発生し、発生したチェレンコフ光は石英内部を全反射して石英版の端に設置されている光検出器 Micro-Channel-Plate-Photomultiplier (MCP-PMT) によって検出される。図 2.12 に TOP における粒子識別概念図を示す。チェレンコフ光子発生から光検出器 MCP-PMT までの伝搬時間と、荷電粒子の位置情報を使用してチェレンコフ光の伝搬時間がチェレンコフ光の放射角に依存することによりその違いによる粒子識別を行っている。さらに ARICH 同様、チェレンコフ光の発光によって生じるチェレンコフリングの半径差を用いた粒子識別も行っており、ARICH で取得される粒子情報と組み合わせることにより高精度の粒子識別を実現している。TOP は比較的 low 運動領域である  $3 \text{ GeV}/c$  以下の領域において、97% の  $K/\pi$  識別効率を目標としている。

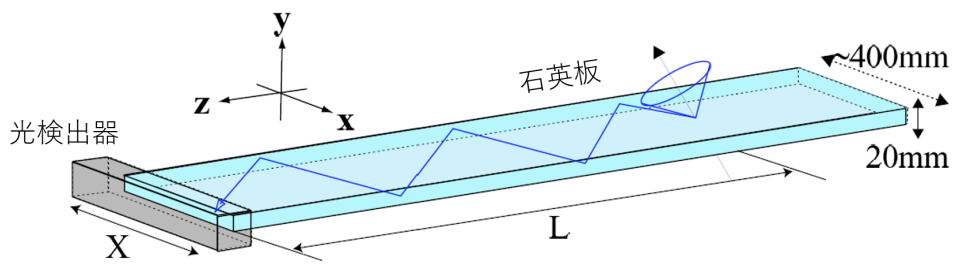


図 2.11 TOP の外観図 [13]

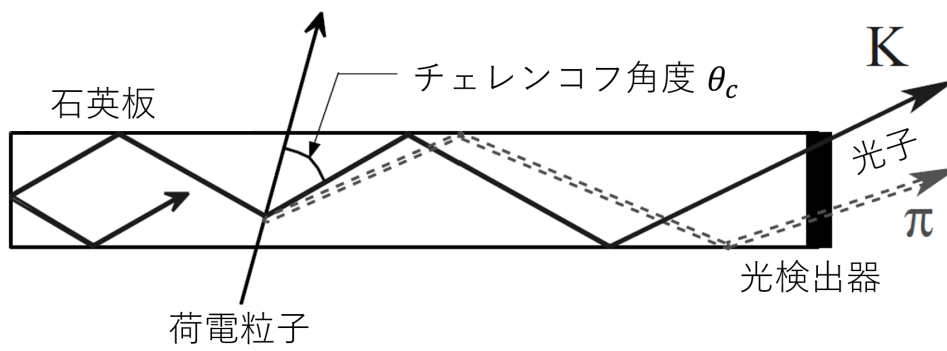


図 2.12 TOP の粒子識別の概念図 [13]

### 2.3.4 ARICH

Aerogel Ring Imaging CHerenkov counter (ARICH) (図 2.13) は、前方エンドキャップ部で荷電  $K/\pi$  中間子の識別を  $4\sigma$  の精度で行うドーナツ型の検出器である。主に輻射体シリカエアロゲルと 420 個の光検出器 Hybrid Avalanche Photo Detector (HAPD) から構成されている。詳細は第 3 章で述べる。

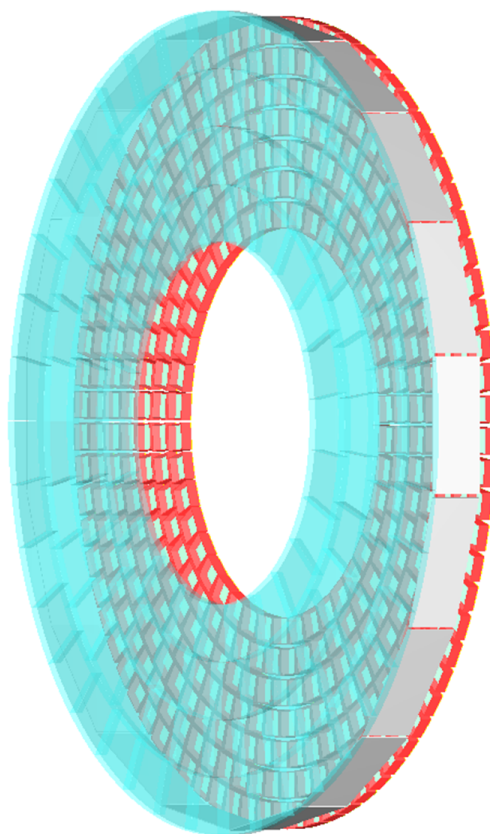


図 2.13 ARICH の外観図

### 2.3.5 ECL

Electromagnetic CaLorimeter (ECL) (図 2.14) は Belle II 測定器のバレル部とエンドキャップ部において  $\gamma, e^\pm$  のエネルギーと入射位置、方向を測定するための検出器である。内径 1.25 m、長さ 3 m のバレル部分と  $-1.02 \text{ m} < z < 1.96 \text{ m}$  のエンドキャップ部から成り、構成物にはシンチレーター (CsI(Tl)) と電子や光子が放出した光の量を測定するフォトダイオードなどがある。シンチレーターはバレル部に 6,621 個、前方エンドキャップ部に 1,152 個、後方エンドキャップ部に 960 個放射状に配置されている。 $\gamma, e^\pm$  が ECL に入射すると電磁シャワーを起こし、ほぼすべてのエネルギーをシンチレーター内部で失う。この際のシャワーによるシンチレーション光の光量が  $\gamma, e^\pm$  のエネルギーに比例するため、これを利用してエネルギー測定を行っている。また、ECL の情報はトリガーに使用されたり、電磁シャワーの形状とエネルギー/運動量比を比較することによる電子/ハドロン識別にも

利用される。

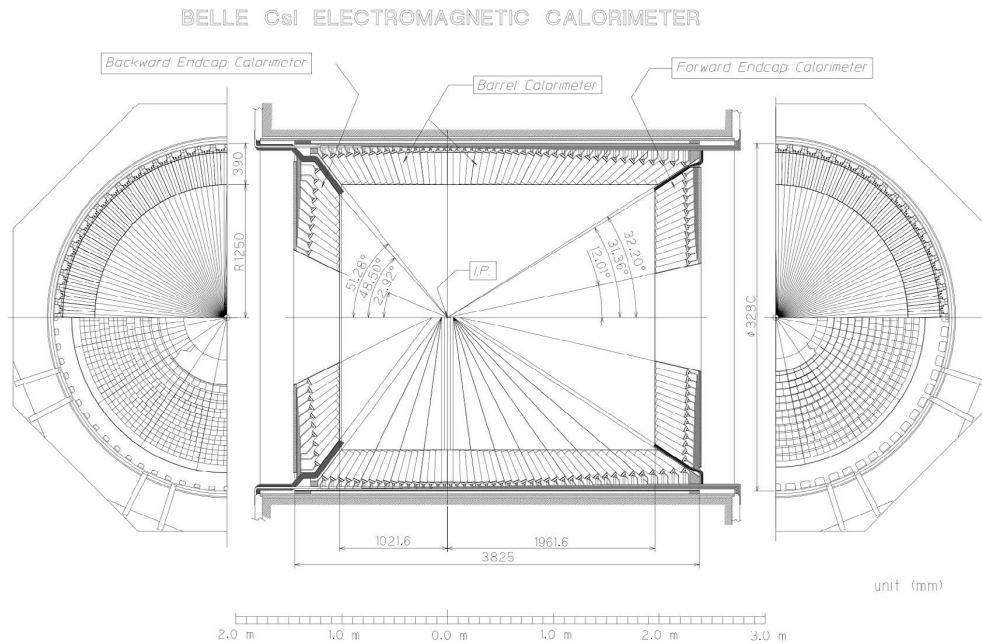


図 2.14 ECL の構成図 [12]

### 2.3.6 KLM

K-Long-Muon detector (KLM) (図 2.15) は Belle II 測定器のエンドキャップ部とバレル部の最外層に位置し、長寿命で透過力が高い  $\mu$  粒子と  $K_L^0$  中間子を識別する検出器である。 $K_L^0$  中間子は本来中性でなおかつ長寿命であるため荷電  $\mu$  粒子との区別をつけることが出来るが、低頻度で検出器内で反応を起こすことがあるためこれらを区別する必要があるが、KLM では測定器内での 2 粒子の散乱の様子の違いから粒子識別を行っている。 $\mu$  粒子が物質中で電離相互作用による電離損失以外ではあまり散乱されずにそのまま通過していく一方で、 $K_L^0$  中間子は強い相互作用により多重散乱を起こすため、KLM の薄い鉄板と検出層を交互に重ねたサンドウィッチ構造を利用することによって物質との相互作用の違いを観測し粒子の軌跡を再構成する。また、バックグラウンドの多さによって使用する検出器を使い分けており高バックグラウンド領域であるエンドキャップ部には光検出器とプラスチックシンチレータ、低バックグラウンド領域ではガスチェンバー Resistive Plate Chambers (RPC) が用いられる。図 2.16 に KLM に設置されている検出器の概略図を示す。

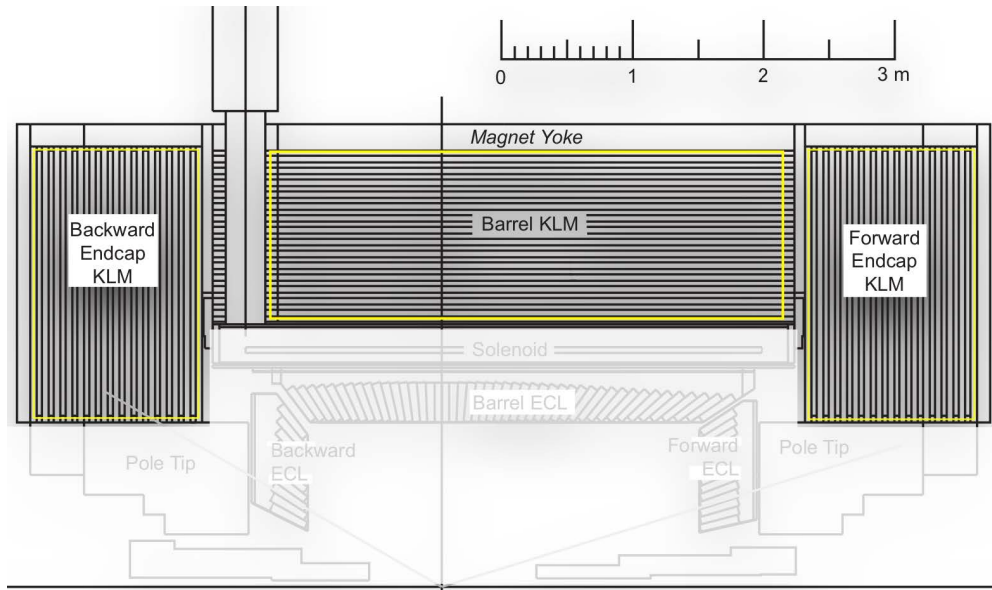


図 2.15 KLM の外観図

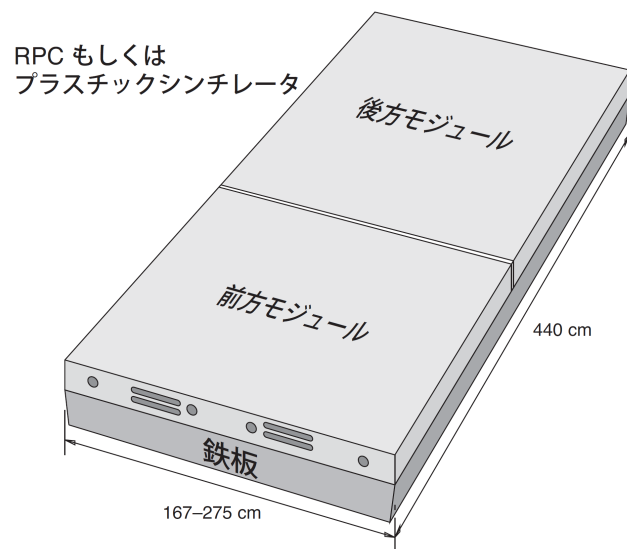


図 2.16 KLM 内の検出器の概略図 [13]

## 第 3 章

# ARICH

Aerogel Ring Imaging Cherenkov counter (ARICH) は第 2 章で述べた通り、Belle II 測定器の前方エンドキャップ部において荷電  $K/\pi$  中間子を識別する役割を持つ検出器である。荷電粒子が輻射体シリカエアロゲルを通過した際に発生するチェレンコフ光を光検出器 Hybrid Avalanche Photo Detector (HAPD) によってリングイメージとして検出し、荷電粒子の通過するトラックとチェレンコフ光が成す放射角の値を用いて粒子識別を行っている。

Belle II 実験において  $B$  中間子の稀崩壊を詳細に調べる際には、崩壊の終状態によく含まれる  $K/\pi$  中間子の識別が非常に重要になってくる。この章では、本研究において Belle II 測定器に搭載されている 7 個の検出器の中でも特に重要な役目を持つ ARICH についてをより詳細に述べる。

### 3.1 チェレンコフ光

通常真空中において光速は常に一定の値を示すが、荷電粒子が空気や水などの媒質中を通過する際には光速は真空中でのそれよりも遅くなる。荷電粒子が媒質中を通過する際にその速度が媒質中の光速度を超えた際に光が放出される現象をチェレンコフ放射といい、その際発生された光のことをチェレンコフ光という。媒質の屈折率を  $n$ 、荷電粒子の速度を  $\beta$ 、真空中の光速度を  $c$  とすると、チェレンコフ光の放射角  $\theta_c$  は式 (3.1) で表される。

$$\cos \theta_c = \frac{\frac{c}{n} \Delta t}{\beta c \Delta t} = \frac{1}{n\beta} \quad (3.1)$$

さらに、発生する光子数  $N$  は式 (3.2) で表される。ここで  $x$  は媒質の厚み、 $\lambda$  は光の波長、 $\alpha$  は微細構造定数 ( $\sim \frac{1}{137}$ )、 $z$  は荷電粒子の電荷 (電子電荷単位における電荷の絶対値で通常 1) である。

$$\frac{dN^2}{dx d\lambda} = \frac{2\pi\alpha z^2}{\lambda^2} \left( 1 - \frac{1}{\beta^2 n^2} \right) \quad (3.2)$$

図 3.1 にチェレンコフ光発生イメージ図を示す。



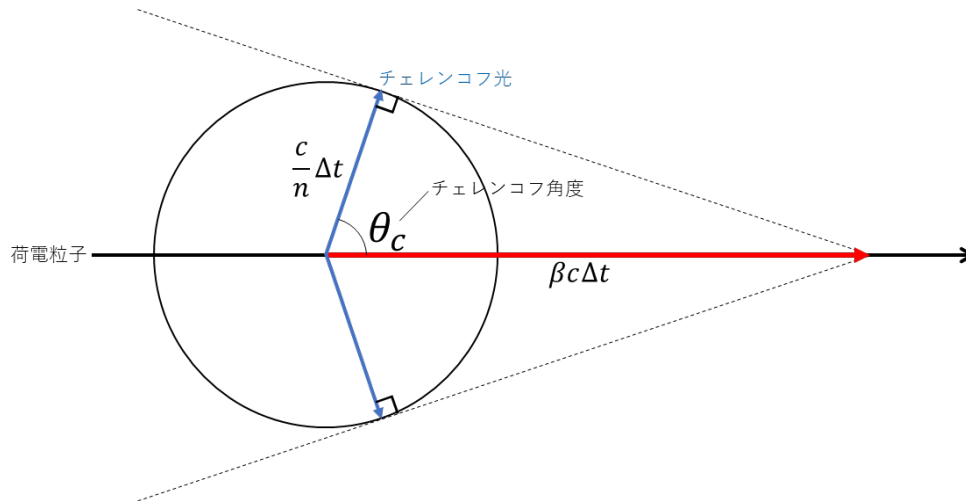


図 3.1 チェレンコフ光の発生イメージ図

### 3.2 ARICH 検出器の粒子識別原理

ARICH の輻射体シリカエアロゲルに荷電  $K/\pi$  中間子が通過すると、前述の通り円錐状のチェレンコフ光が発生する。これを光検出器 HAPD で検出し、リングイメージとして観測する。その際の荷電粒子が通過したトラックとチェレンコフ光が成す角度である放射角を測定することにより、粒子識別を行っている。荷電粒子の質量と放射角の関係は式 (3.1) の  $\beta$  の代わりに CDC から  $p$  を測定する場合から直接導かれ、式 (3.3) で表される。ここで、 $n$  は輻射体シリカエアロゲルの屈折率、 $p$  は荷電粒子の運動量、 $m$  は荷電粒子の質量であり、さらにこれより記述される式は特別な表記が無い限り全て  $\hbar = c = 1$  とする自然単位系の元に表記されているものとする。

$$\cos \theta_c = \frac{1}{n} \frac{\sqrt{p^2 + m^2}}{p} \quad (3.3)$$

図 3.2 に ARICH の識別原理の概念図を示す。

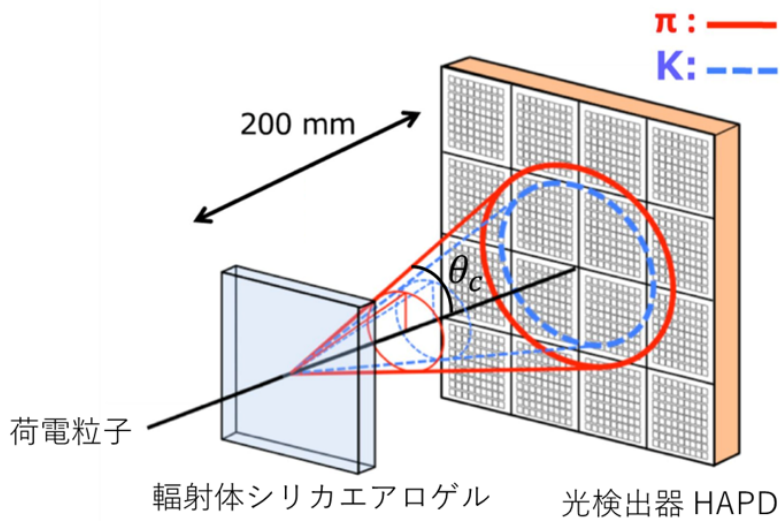


図 3.2 ARICH の識別原理

荷電粒子の運動量を  $p = 3.5 \text{ GeV}/c$ 、輻射体シリカエアロゲルの屈折率を  $n = 1.05$  とすると、式 (3.3) より通過粒子が荷電 K 中間子だった場合の放射角は  $277 \text{ mrad}$ 、荷電  $\pi$  中間子だった場合の放射角は  $307 \text{ mrad}$ 、また HAPD 側におけるリングイメージの半径はそれぞれ  $48 \text{ mm}$ 、 $54 \text{ mm}$  となり、リングイメージの半径差は  $6 \text{ mm}$  となるため、この半径差を利用することで粒子識別が可能になる。(図 3.3)

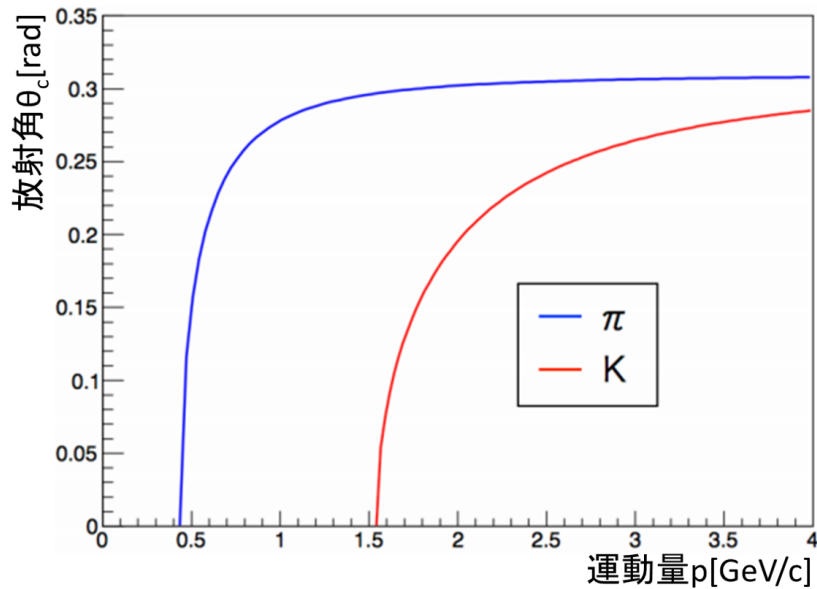


図 3.3 荷電 K/ $\pi$  中間子の放射角分布

### 3.3 ARICH 検出器の構造

ARICH は中心部に穴が開いたドーナツ型の構造をしており、その外径は 1.1 m、内径は 0.4 m である。主な構成物に厚さ 4 cm の輻射体シリカエアロゲルと厚さ 3 cm の光検出器 HAPD があり、その間にはチェレンコフ光が拡がるための 16 cm の空間をとっている。図 3.4 に ARICH の各構成要素の配置の概略図を示す。

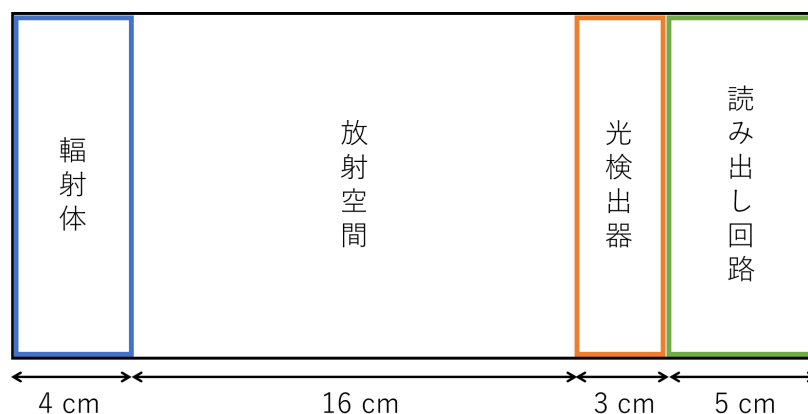


図 3.4 ARICH の各構成要素とその配置の概略図

また、ARICH には外周の側面に鏡が設置されており、発生したチェレンコフ光を ARICH 内部に反射させ HAPD で観測することができるようになっている。

#### 3.3.1 輻射体シリカエアロゲル

輻射体シリカエアロゲルは二酸化ケイ素 ( $\text{SiO}_2$ ) の骨格に細かい気泡を大量に含む多孔性の物質である。半透明な見た目をしており、非常に低密度 ( $10 - 150 \text{ mg/cm}^3$ ) で高い透明度 (減衰長は 60 mm) を持つ。ARICH では屈折率 1.045 と 1.055 の 2 種類のを重ね合わせ、 $17 \text{ cm} \times 17 \text{ cm}$  の大きさに加工したものを 248 ( $124 \times 2$ ) 枚設置している。

輻射体シリカエアロゲルを 2 枚組みこむ理由は、発生光子数を減らさずにチェレンコフ光の放射角不定性を抑制するためである。チェレンコフ光による光子数は一般的に媒質の厚みに比例し増加するが、媒質を厚くすればするほどその分媒質内部のどの部分で発光したかが不定になってしまうという欠点も存在する。したがって本実験における ARICH に搭載されている輻射体シリカエアロゲルでは、屈折率の異なる 2 種類のシリカエアロゲルを重ねて用いることにより、発生したチェレンコフ光をより集光し発生位置範囲狭めることでチェレンコフ光の放射角の不定性を抑制している。このシリカエアロゲルを 2 枚重ねて用いる方式をデュアルレイヤー方式 (図 3.5) と呼ぶ。

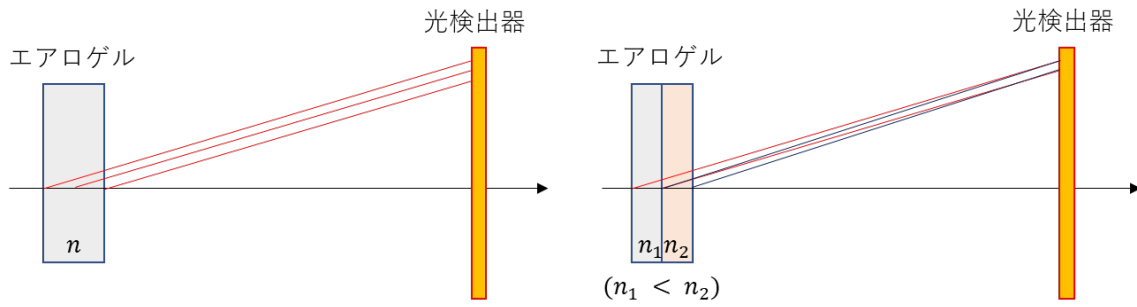
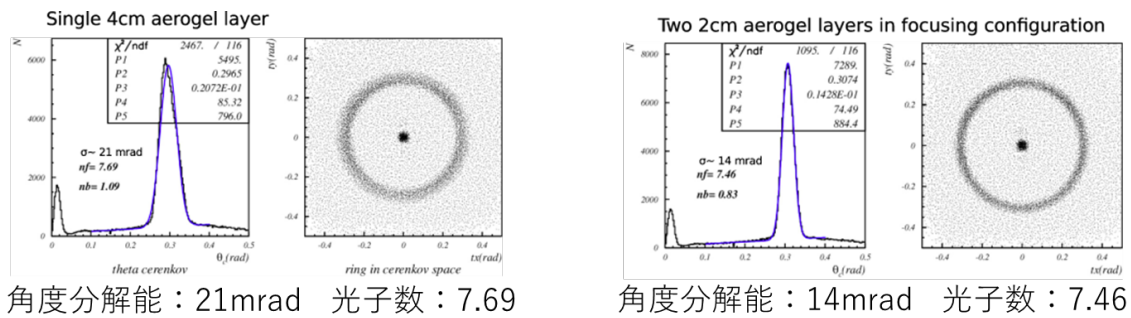


図 3.5 (左) 単層方式、(右) デュアルレイヤー方式

このデュアルレイヤー方式を用いることにより、ARICH のチェレンコフ光の放射角検出における角度分解能は単層方式の場合の 21 mrad から 14 mrad へ向上し、この際の平均検出光子数は 7.69 から 7.49 となり光子数は概ね保持される。図 3.6 に輻射体シリカエアロゲルをそれぞれ (左) 1 枚使用した場合 (奥行き 4 cm) と (右) 2 枚使用した場合 (奥行き 2 cm × 2 枚) の角度分解能と平均検出光子数を示す。



角度分解能：21mrad 光子数：7.69

角度分解能：14mrad 光子数：7.46

図 3.6 (左) エアロゲルを 1 枚使用した場合、(右) エアロゲルを 2 枚使用した場合における ARICH の角度分解能と平均検出光子数

### 3.3.2 光検出器 HAPD

Belle II 実験で用いられる光検出器は Hybrid Avalanche Photo Detector (HAPD) と呼ばれ、粒子識別のためのリングイメージを得る際に重要となる 1 光子検出が可能な検出器である。図 3.7 に光検出器 HAPD の外形を示す。

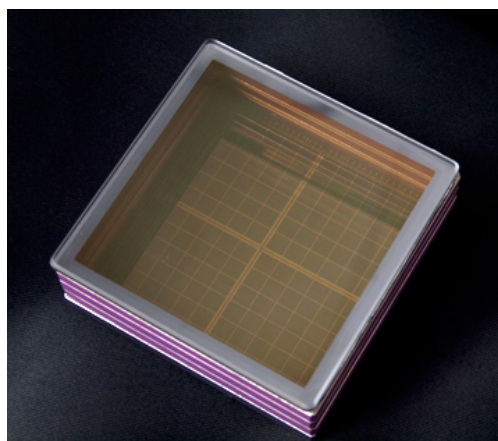


図 3.7 光検出器 HAPD

大きさは 73 mm × 73 mm であり、真空状態の HAPD 内部には 36 チャンネルにピクセル化 (ピクセルサイズは 4.9 mm × 4.9 mm) された Avalanche Photo Diode (APD) が 4 個搭載されている。HAPD 一つあたりのチャンネル数は 144 であり、高磁場中 (1.5 T) でも動作が可能のように設計されている。さらに、Belle II 実験で予定されている運転期間 10 年の間に曝される放射線にも耐性がある。

ここで HAPD と APD の動作原理をそれぞれ図 3.8, 3.9 に示す。まずチェレンコフ光が HAPD 陰極側の光電面に入射すると、HAPD 内部に電子が放出される。この電子に電圧 ( $\sim -8$  kV) をかけて APD に衝突させる。この際 APD には逆バイアス電圧 ( $\sim 330$  V) がかけられており、これにより APD 内に空乏層が作られ加速された電子が空乏層内で電子正孔対を約 1700 対生成する。この電子と正孔がそれぞれ電極へとドリフトする際に、空乏層内の格子電子や不純物などと衝突し最終的に電子を約 40 倍に Avalanche 増幅させる。この現象を電子雪崩という。これら二つの増幅により、最終的に HAPD は  $\sim 68,000$  倍という高い増幅率を有している。さらに、HAPD 内の APD ではバイアス電圧の印加によるエネルギーギャップのためノイズが抑制され、増幅率と相まって高い S/N 比での 1 光子検出が可能となる。

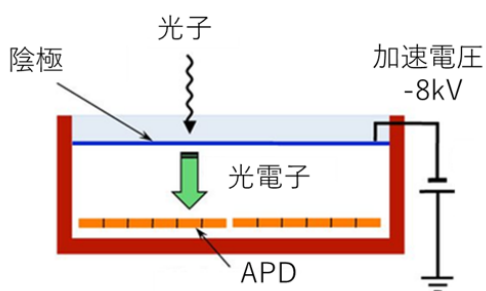


図 3.8 光検出器 HAPD の動作原理

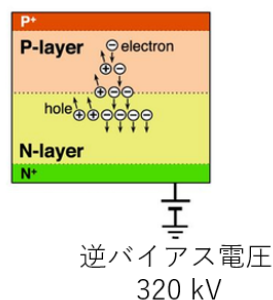


図 3.9 APD の動作原理

### 3.4 ARICH 検出器への要求性能

Belle II 実験の大きな目的の一つである新物理事象探索において、 $2 \text{ GeV}/c$  を超える高エネルギー領域での  $B$  中間子の稀崩壊を行うためには、高エネルギー領域での荷電粒子識別が必要である。本実験の観測で具体的に用いられる  $B$  中間子の稀崩壊モードとして  $B \rightarrow KK, \pi\pi, K\pi$  などがあるが、これらに含まれる  $K/\pi$  中間子は最大  $3.5 \text{ GeV}/c$  程度の運動量を持つため、ARICH はこの運動量付近での  $4\sigma$  以上の精度での識別能力を持つことが要求される。

## 第4章

# ARICH 検出器のアライメント手法

### 4.1 アライメントの概要

先述の通り ARICH はその設計位置からのずれに  $K/\pi$  中間子識別能力が大きく依存するため、検出器の性能を最大限に引き出すには検出器の設計上の位置からのずれを算出・修正するアライメントという作業が不可欠である。本研究は Belle II 実験で取得された実データを用いることにより、ARICH の位置を確認しずれていたら補正するという作業を行うことによって ARICH の更なる識別能向上を目指す。図 4.1 に ARICH がそれぞれ正しい位置 (設計上の位置) にある場合と、ずれている場合における荷電粒子の通過するトラックとチェレンコフリングの関係を示す。ARICH が正しい位置にあるとき、チェレンコフリングの円の中心をトラックが貫くためチェレンコフ光の放射角の測定値はリング上の位置とは無関係に一定になる一方、ARICH がずれているときは観測される放射角がずれの方向に依存して誤った角度に再構成されてしまい、放射角がリング上の位置に依存して変化しているように見える。

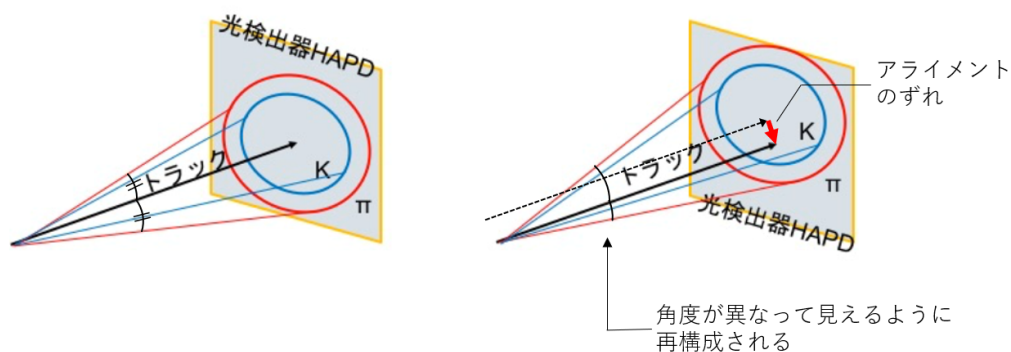


図 4.1 (左) ARICH が正しい位置にあるとき、(右) ARICH がずれたとき

ARICH のアライメントには電子・陽電子衝突による  $e^+ + e^- \rightarrow \mu^+ + \mu^-$  事象を使用した。 $\mu$  粒子は電子と比較し電磁シャワーやハドロン相互作用を発生させず粒子が描くトラックが比較的単純なものとなるためアライメントに適しており、本研究にはこの事象を用いるのが最も適切であるということが言える。本研究では ARICH の設計上の位置からのずれを測定し補正するために、この事象による

モンテカルロシミュレーションを用いてチェレンコフ光の放射角の位置依存性分布を測定するというアライメントの手法の開発を行った。ARICH の飛跡検出器に対するずれを加えた Belle II 測定器のモデルを作成し、ARICH と飛跡検出器間の相対的な位置のずれがチェレンコフ光の放射角分布にどのように影響するかを調べた。

## 4.2 先行研究

本研究の先行研究は、Belle II 実験初期のビーム衝突データを用いて ARICH と飛跡検出器間の相対的なアライメントを行った。先行研究にて使用された実データは Belle II 実験初期 (Phase II) に収集された Prod5 と呼ばれるものである。Prod 5 データにはカット条件をかけ、電子・陽電子散乱 (Bhabha 散乱)  $e^+ + e^- \rightarrow e^+ + e^-$  における電子のイベント (イベント数: 4,238,953) を使用し、電荷が負、 $E/p > 0.9$ 、 $E_9/E_{21} > 0.95$ 、再構成された粒子の運動量が 4 GeV/c 以上の選別を行った。さらに、モンテカルロシミュレーションにおいても同様の Bhabha 散乱事象を用いて、実データと同様の実験環境及び解析手法で ARICH が正しい設計位置からずれたときの応答を定量的に調べた。

ずれの値を実際に解析した結果、ARICH の並進方向、回転方向のずれの大きさは表 4.1 のとおりであった。

表 4.1 先行研究による ARICH のずれの値

	並進移動 [mm]	回転移動 [°]
x 軸	$-0.816 \pm 0.0007$	$0.0537 \pm 0.0010$
y 軸	$0.354 \pm 0.006$	$0.1034 \pm 0.0010$
z 軸	$1.790 \pm 0.009$	$-0.0130 \pm 0.0002$

これらのずれの値を補正した結果、ARICH の  $K/\pi$  中間子識別精度は 7% 向上し、それによって ARICH の分解能も 7% 向上した [14]。

## 4.3 先行研究の問題点と改良

先行研究で行ったアライメントの手法にはいくつか改善が見込める所がある。本研究では、以下で述べるような ARICH のアライメントの手法の改良を施すことによりさらなる分解能の向上を目指した。

第一に、アライメントに用いた実データを Phase II の Prod 5 から Phase III の Bucket 15 に変更し、それに伴いモンテカルロシミュレーションで作成された仮想的なずれがある場合のファイルを再度作成し直した。先行研究ではシミュレーションにプロセス  $e^+ + e^- \rightarrow \mu^+ + \mu^-$  における  $\mu$  粒子でのアライメントを予定していたが、Phase II では統計量が不足しておりトリガー条件よりこの事象の取得が困難だったために電子・陽電子散乱のプロセス  $e^+ + e^- \rightarrow e^+ + e^-$  を用いていた。新しいプロセスでは  $e^+ + e^- \rightarrow \mu^+ + \mu^-$  を用いることにより、 $\mu$  粒子の電磁シャワーや強い相互作用が少ないというような特徴を利用し先行研究と比較しより正確な ARICH のずれの値の算出を目指した。

第二に、先行研究のアライメントフィットの方法ではフィッティング精度が低かったため、フィッ



トの手法を変更することによってこれの向上を目指した。詳しくは第 6 章で述べる。

## 4.4 ARICH 検出器をずらした時のシミュレーション

節 4.1 や 4.2 で述べたように、ARICH のアライメントを行うにあたりまずは飛跡検出器に対するずれを ARICH に加えた Belle II 検出器のモデルでのモンテカルロシミュレーションを実施し、ARICH のずれを仮定したときの解析結果を確認した。ARICH の座標軸の定義を図 4.2 に示す。

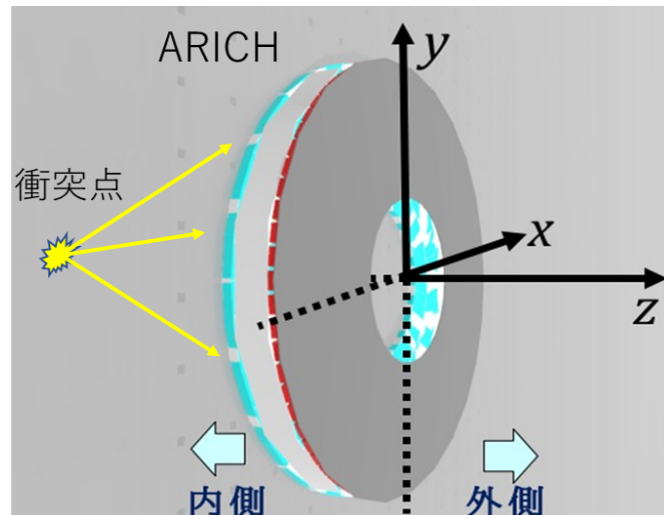


図 4.2 ずれの表現における ARICH の座標軸の定義

ARICH に荷電粒子によるチェレンコフ光が入射する面にそれぞれ  $x$  軸、 $y$  軸をとり、この面に垂直な方向のうち、電子の入射側である後方エンドキャップ側から陽電子の入射側である前方エンドキャップ側に向かう方向に  $z$  軸をとる。

また、節 4.1 で述べたとおり、ARICH が設計上の正しい位置にある場合とずれている場合におけるチェレンコフ光の放射角に関する情報を定量的に評価するために、図 4.3 のように変数を定義する。

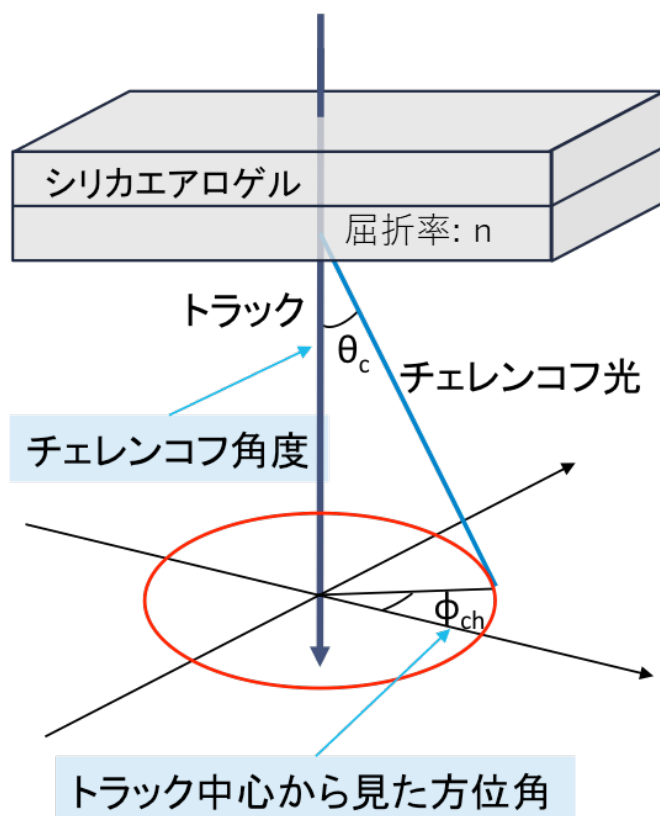


図 4.3 ARICH のアライメントに用いる変数の定義

実験により取得されたデータにおいて、それぞれ輻射体シリカエアロゲルに入射した荷電粒子のトラックの位置を再構成した情報と、荷電粒子が発生させたチェレンコフ光の位置を再構成した情報の値を用いし、そしてそれらが成す角の値を計算したものを  $\theta_c$ 、またトラック中心から HPAD の向きに見た際の方位角の値を  $\phi_{ch}$  と定義する。これらの変数を用いると、ARICH が設計上の正しい位置にある場合  $\theta_c$  は  $\phi_{ch}$  に依存しない変数となるが、ARICH がずれている場合  $\theta_c$  の  $\phi_{ch}$  依存性が見られ、これによって検出器のずれの有無を確認することができる。

本研究では、節 3.3 で述べたような輻射体シリカエアロゲルのデュアルレイヤー方式による屈折率の違いを考慮するために  $\theta_c$  の代わりに  $n \cos \theta_c$  を用いた。ここで  $n$  は二つのシリカエアロゲルの層からの光子の放出確率を重みとした加重平均の値であり、さらに式 (3.1) を変形して粒子の速度  $\beta$  についての式にすると  $(n \cos \theta_c)^{-1}$  は粒子の速度に対応していることが分かることから、これは屈折率に依存しない。図 4.4 はモンテカルロシミュレーションにより算出された ARICH が設計上の正しい位置にある時のチェレンコフ光による放射角  $n \cos \theta_c$  の  $\phi_{ch}$  依存性を調べた図である。このように ARICH が正しい位置にある場合、二次元ヒストグラムにおけるイベント数が多いピンは  $n \cos \theta_c = 1.000$  付近に集中し、かつヒストグラムにおける  $n \cos \theta_c = 1.000$  の軸を中心にほとんど完全に上下対称に分布していることが分かる。

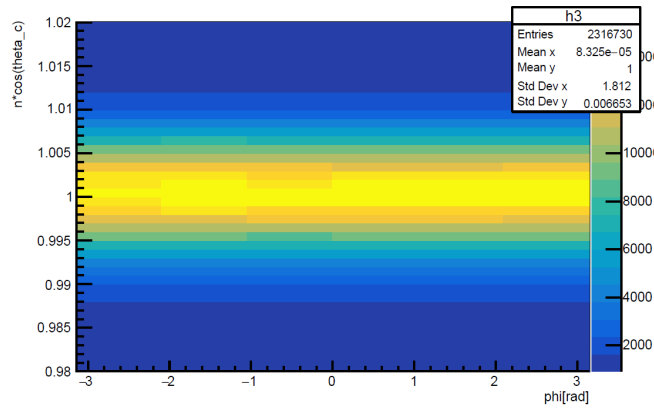


図 4.4 ARICH が正しい位置にある場合の  $n \cos \theta_c$  の  $\phi_{ch}$  依存性

さらに、ARICH の設計上の位置からのずれを算出するにあたり、ずれの分布が一様ではなく位置によって異なり依存性を持つ可能性を考慮する必要があるため ARICH を 36 個の領域に分割し、ずれの位置依存性が考慮されたずれの値の算出を行う。図 4.5 に分割された 36 領域と各領域の番号を示す。軸の向きは図中に示すとおりであり、図 4.2 における内側から見た図である。

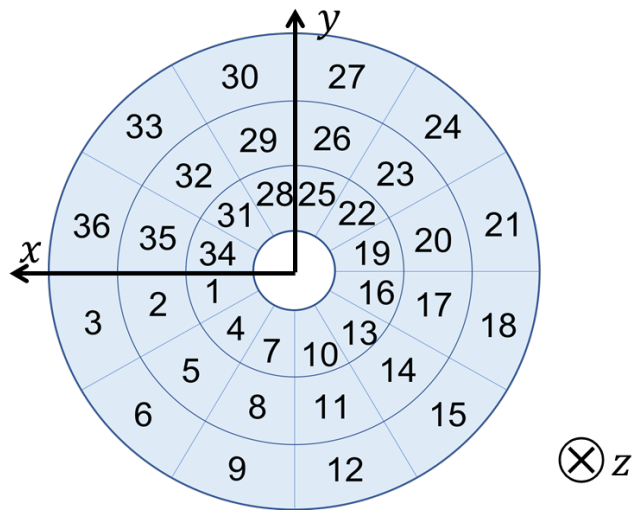


図 4.5 36 領域に分割された ARICH

#### 4.4.1 ARICH 検出器の並進方向のずれのシミュレーション

ARICH を 36 領域に分割することによって設計上の正しい位置からのずれを並進方向  $x$  軸、 $y$  軸、 $z$  軸、回転方向  $x$  軸、 $y$  軸、 $z$  軸の合計 6 個のパラメータで表現する。この節ではまず並進方向に仮想的なずれがあった場合のモンテカルロシミュレーション結果についてを述べる。なおこれ以降モンテカルロシミュレーションによって再現された ARICH のずれを「仮想的なずれ」と呼び、実データに

よって算出された ARICH のずれと区別することにする。

それにあたりまずは図 4.5 のように ARICH を 36 領域に分割した際の各領域の  $n \cos \theta_c$  の  $\phi_{ch}$  依存性を関数  $n \cos \theta_c = a + b \cos(\phi_{ch} + c)$  でフィットした結果を示す。フィットをする際には得られた  $n \cos \theta_c$  の  $\phi_{ch}$  依存性を示す二次元ヒストグラムにおける  $\phi_{ch}$  の区域を 13 個に分割し、各々の区間をガウス関数などでフィットすることによりデータ点を 13 個作成し、さらにその 13 個の点を  $\cos$  関数でフィットするという方法を用いた。

図 4.6, 4.7 は ARICH を 並進方向  $x$  軸にそれぞれ  $\pm 2$  mm ずらしたときの  $n \cos \theta_c$  の  $\phi_{ch}$  依存性を示したグラフである。フィット関数の振幅は 0.002-0.003 程度であり、36 個のグラフはおおよそ同じ形をしている。

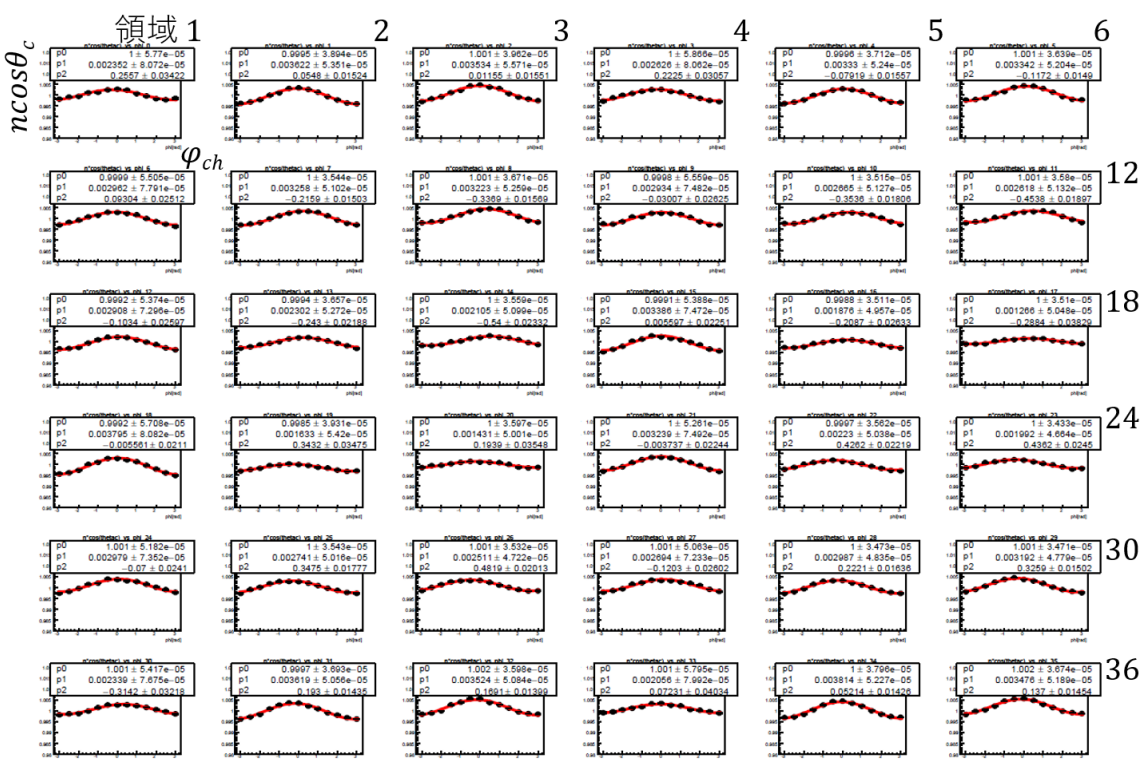


図 4.6 ARICH の並進方向  $x$  軸 2 mm ずらしたときのシミュレーション結果のフィット

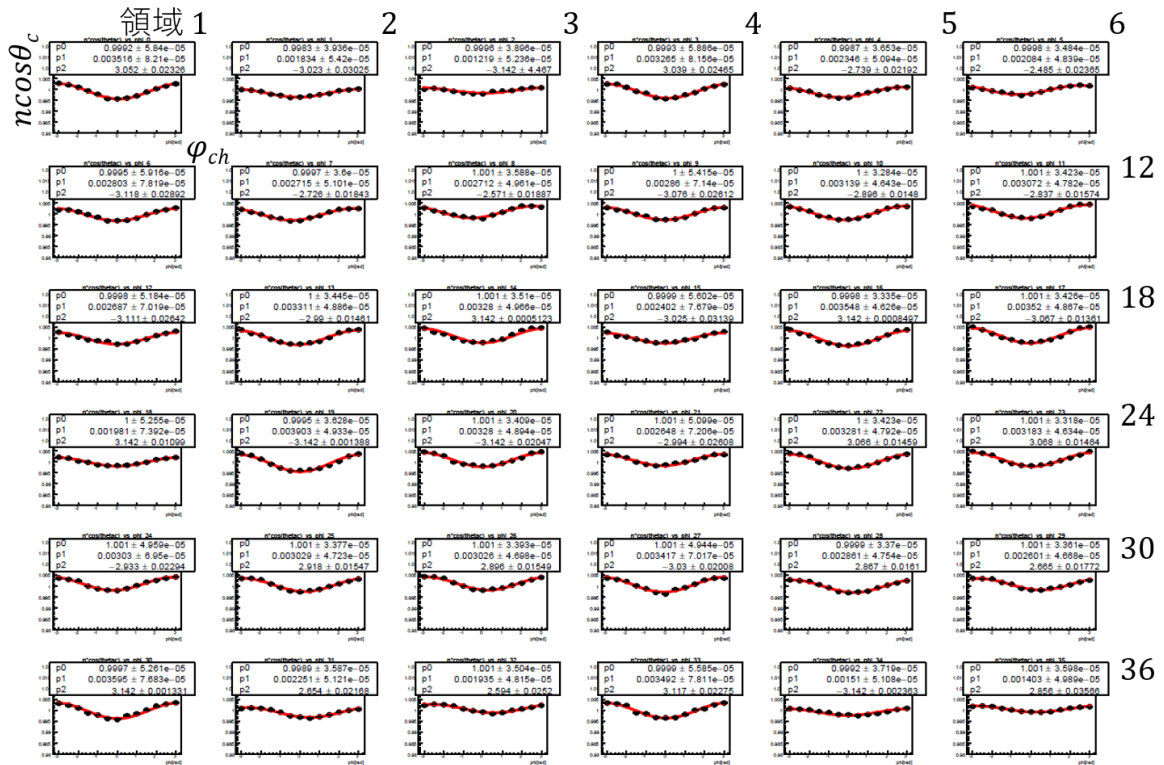


図 4.7 ARICH の並進方向  $x$  軸  $-2$  mm ずらしたときのシミュレーション結果のフィット

図 4.8, 4.9 は ARICH を並進方向  $y$  軸にそれぞれ  $\pm 2$  mm ずらしたときの  $n \cos \theta_c$  の  $\phi_{ch}$  依存性を示したグラフである。フィット関数の振幅は  $0.002$ - $0.003$  程度である。並進方向  $x$  軸のずれのグラフと比較すると、フィット関数である  $\cos$  関数の位相がそれぞれ  $\pm \frac{\pi}{2}$  ずれているが 36 個のグラフはおおよそ同じ形をしている。

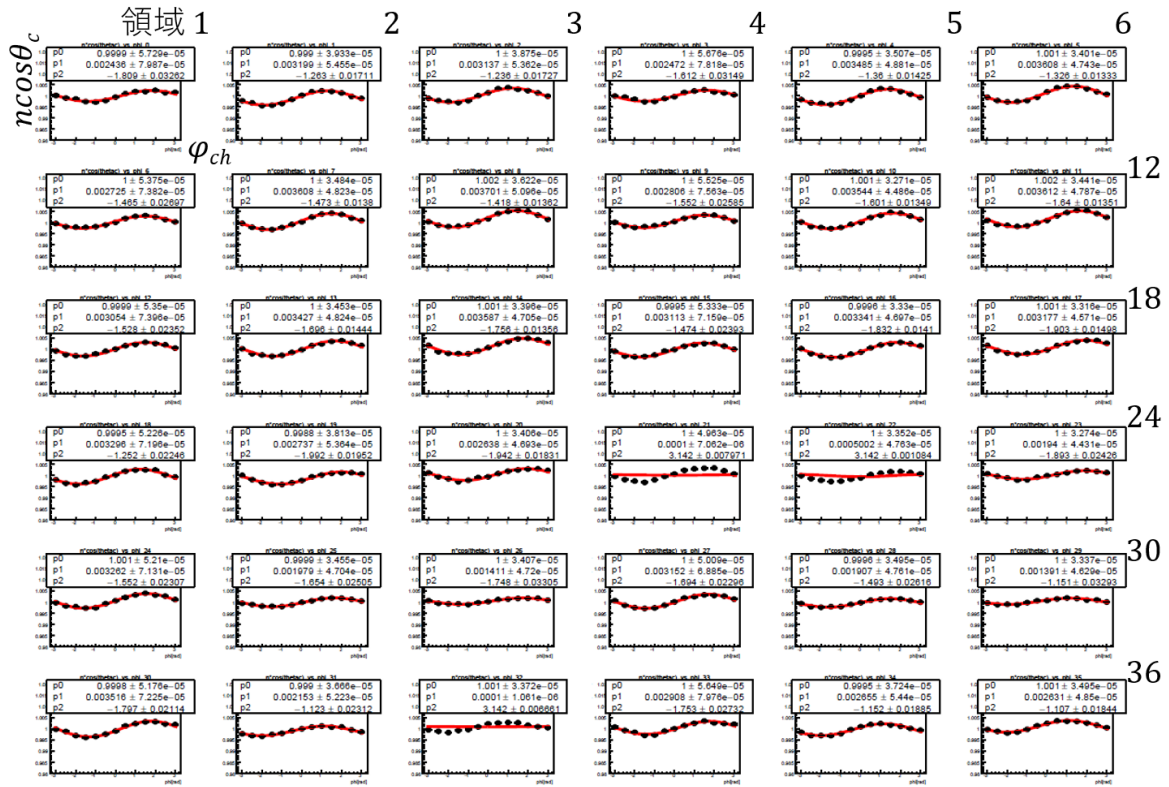


図 4.8 ARICH の並進方向 y 軸 2 mm ずらしたときのシミュレーション結果のフィット

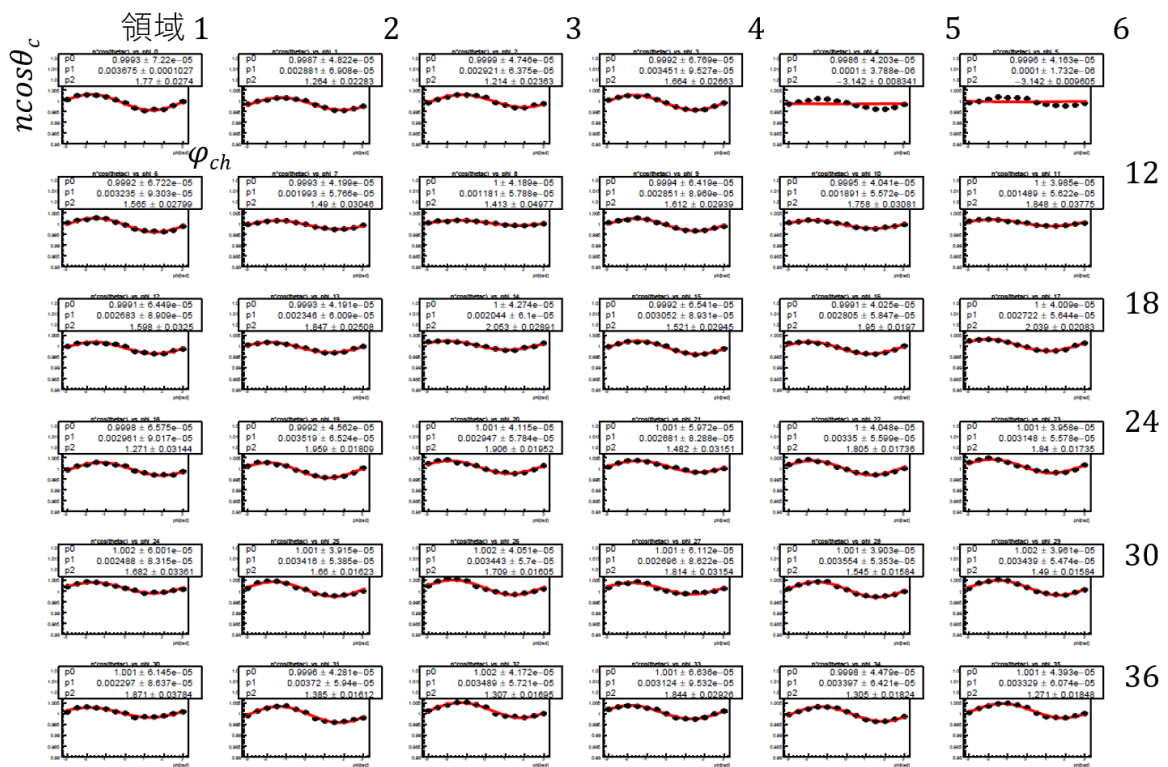


図 4.9 ARICH の並進方向 y 軸 -2 mm ずらしたときのシミュレーション結果のフィット

図 4.10, 4.11 は ARICH を並進方向  $z$  軸にそれぞれ  $\pm 5$  mm ずらしたときの  $n \cos \theta_c$  の  $\phi_{ch}$  依存性を示したグラフである。フィット関数の振幅は 0.001-0.004 程度であり、36 領域においてそれぞれフィット関数の位相や振幅が変化している。ARICH の内側の領域 (図 4.5 における領域 1, 4, 7,  $\dots$ , 34) ではフィット関数の振幅が小さくなり、外側の領域 (図 4.5 における領域 3, 6, 9,  $\dots$ , 36) ではフィット関数の振幅が大きくなる傾向がある。

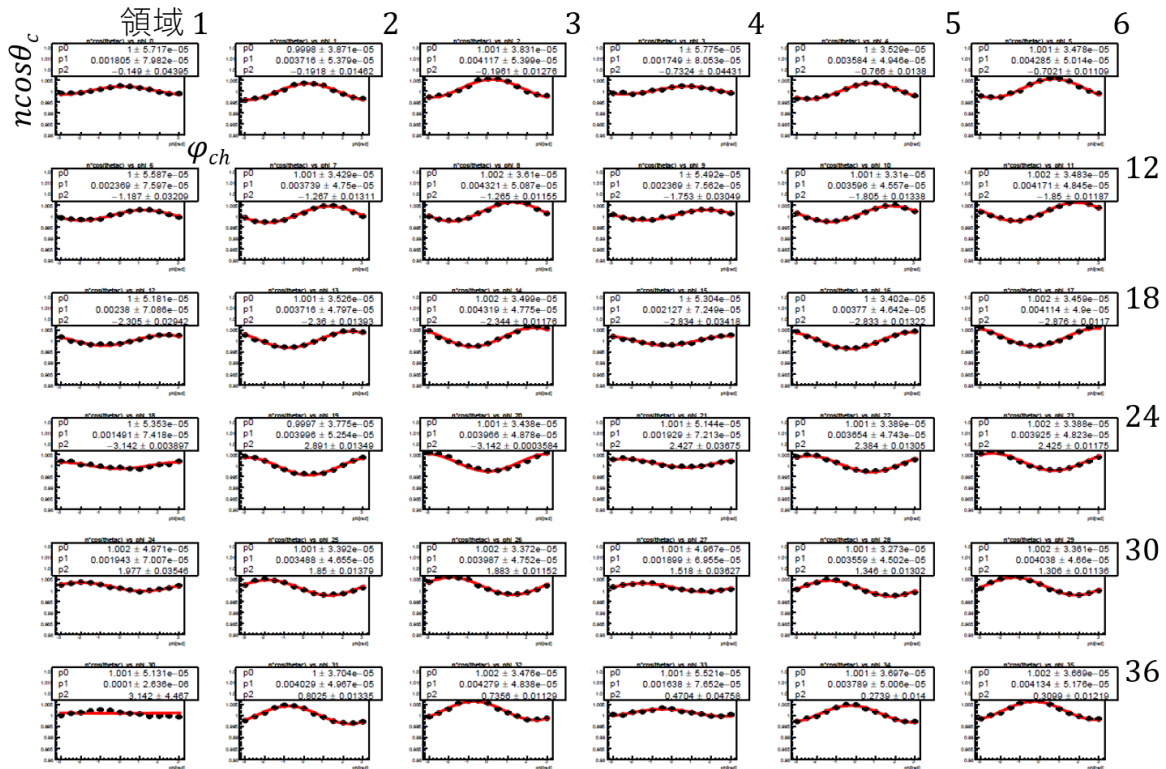


図 4.10 ARICH の並進方向  $z$  軸 5 mm ずらしたときのシミュレーション結果のフィット



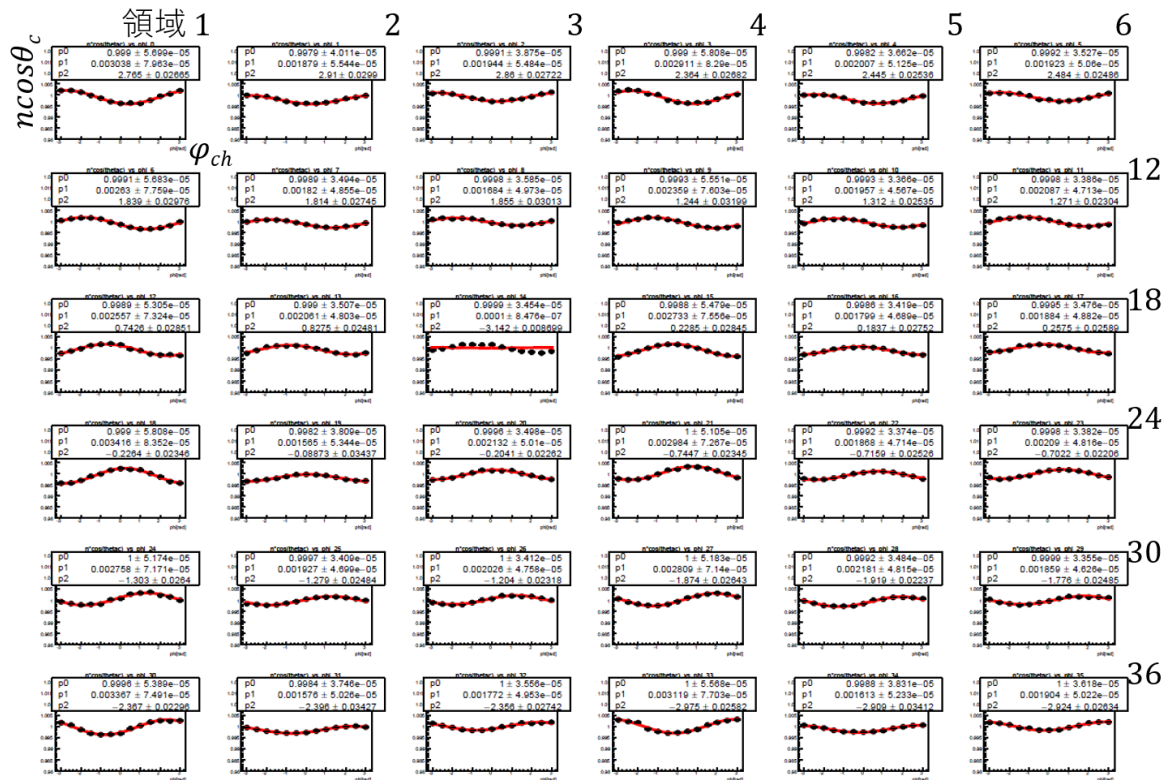


図 4.11 ARICH の並進方向  $z$  軸  $-5$  mm ずらしたときのシミュレーション結果のフィット

これらの 6 個の図 (図 4.6 - 4.11) について、各 36 領域における  $\cos$  関数のフィットから得られたそれぞれ振幅と位相の数値を利用し各 36 領域ごとの仮想的ずれを青色の矢印で表示する。図 4.12, 4.13 に各 36 領域ごとの仮想的ずれのシミュレーション結果の矢印表示を示す。なおこれ以降図 4.12, 4.13 のようなずれをそれぞれ「正の仮想的ずれ」、「負の仮想的ずれ」と呼ぶことにする。36 個の矢印について、 $\cos$  関数の振幅を矢印の長さ、 $\cos$  関数の位相を矢印の向きに対応させて表示している。

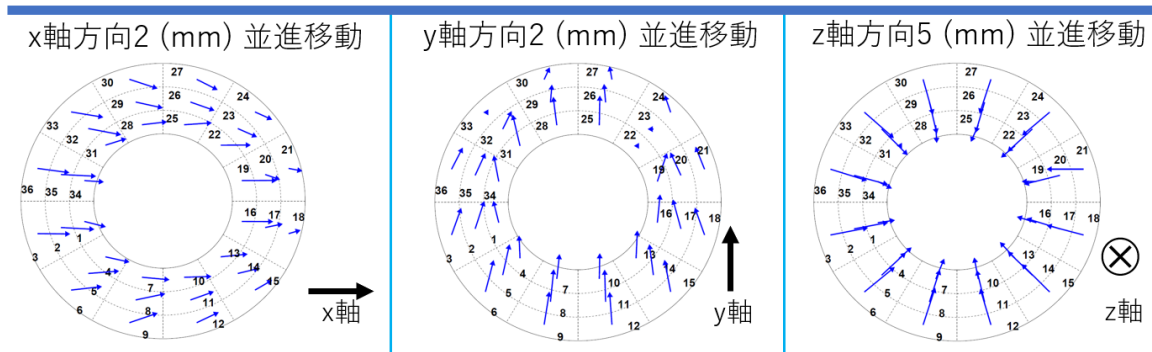


図 4.12 ARICH 並進方向  $x$  軸、 $y$  軸、 $z$  軸に対する正の仮想的ずれのシミュレーション結果の矢印表示

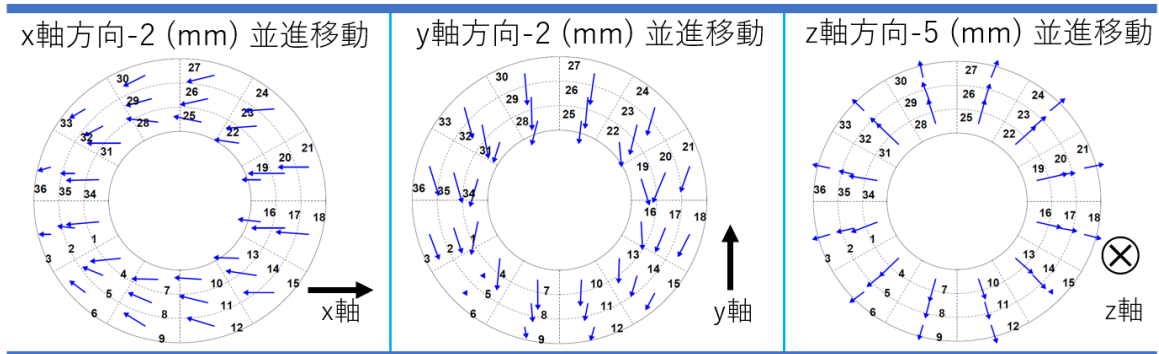


図 4.13 ARICH 並進方向  $x$  軸、 $y$  軸、 $z$  軸に対する負の仮想的ずれのシミュレーション結果の矢印表示

図 4.12, 4.13 の ARICH 並進方向  $x$  軸、 $y$  軸の矢印表示を見ると、先行研究の結果と比較して矢印の向きが ARICH の内側に歪んでおり、並進方向  $x$  軸、 $y$  軸のずれに並進方向  $z$  軸の結果が混ざっているように見受けられる。本研究では正負両方の仮想的ずれの値のシミュレーション結果を取得し、さらに並進方向  $z$  軸の仮想的ずれの大きさを他の 2 方向のそれより大きく仮定することによってこの ARICH 並進方向  $z$  軸のオフセットを求めることも目的としている。図 4.14 に先行研究でのモンテカルロシミュレーションの ARICH の仮想的ずれの結果を示す。なお先行研究のシミュレーションでは仮想的ずれの値を  $x$  軸、 $y$  軸、 $z$  軸でそれぞれ 5 mm としている。

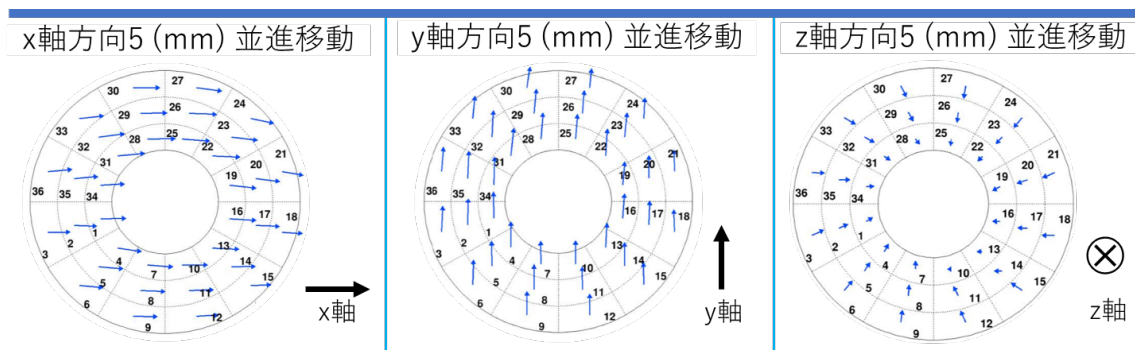


図 4.14 先行研究における ARICH 並進方向  $x$  軸、 $y$  軸、 $z$  軸に対する仮想的ずれのシミュレーション結果の矢印表示 [14]

#### 4.4.2 ARICH 検出器の回転方向へのずれのシミュレーション

ARICH がそれぞれ  $x$  軸、 $y$  軸、 $z$  軸を中心に回転移動をする場合を考える。

図 4.15, 4.16 は ARICH を回転方向  $x$  軸にそれぞれ  $\pm 0.5^\circ$  ずらしたときの  $n \cos \theta_c$  の  $\phi_{ch}$  依存性を示したグラフである。フィット関数の振幅は 0.001-0.007 程度であり、36 領域においてそれぞれフィット関数の位相や振幅が変化している。

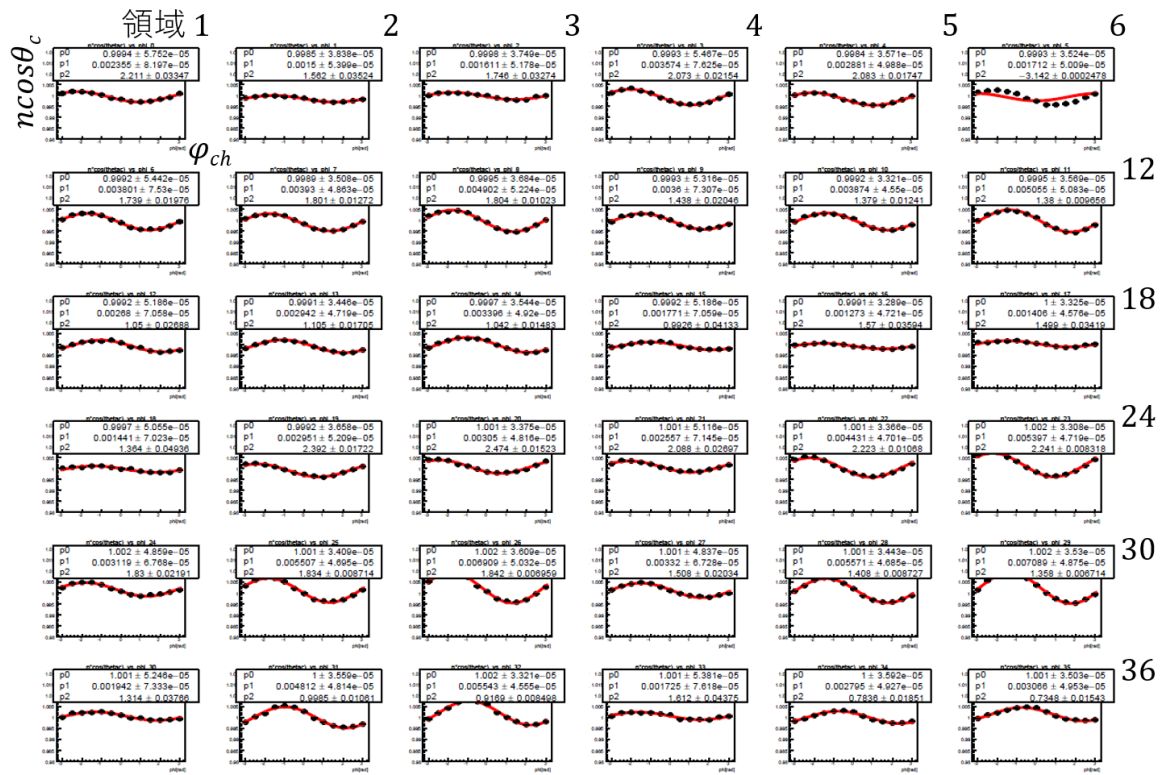


図 4.15 ARICH の回転方向  $x$  軸  $0.5^\circ$  ずらしたときのシミュレーション結果のフィット

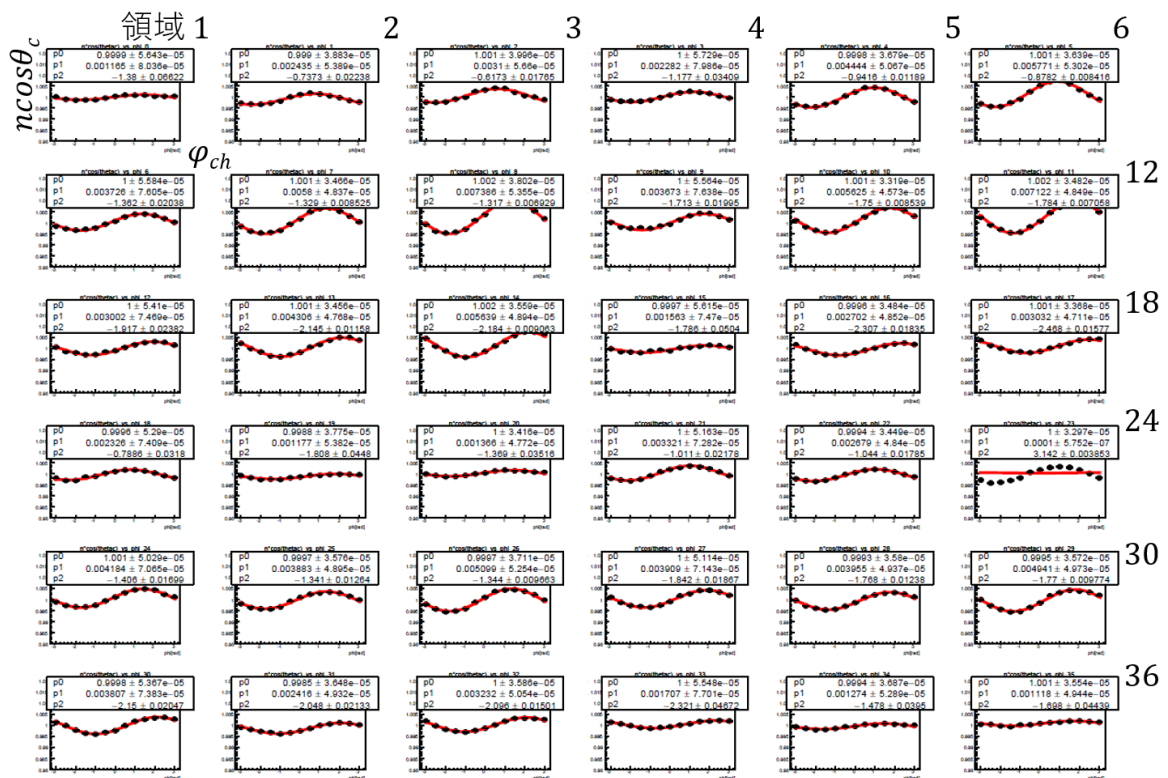


図 4.16 ARICH の回転方向  $x$  軸  $-0.5^\circ$  ずらしたときのシミュレーション結果のフィット

図 4.17, 4.18 は ARICH を回転方向  $y$  軸にそれぞれ  $\pm 0.5^\circ$  ずらしたときの  $n \cos \theta_c$  の  $\phi_{ch}$  依存性を示したグラフである。フィット関数の振幅は 0.001-0.007 程度であり、36 領域においてそれぞれフィット関数の位相や振幅が変化している。回転方向  $x$  軸のずれのグラフと比較すると、フィット関数である  $\cos$  関数の位相がそれぞれ  $\pm \frac{\pi}{2}$  ずれている。

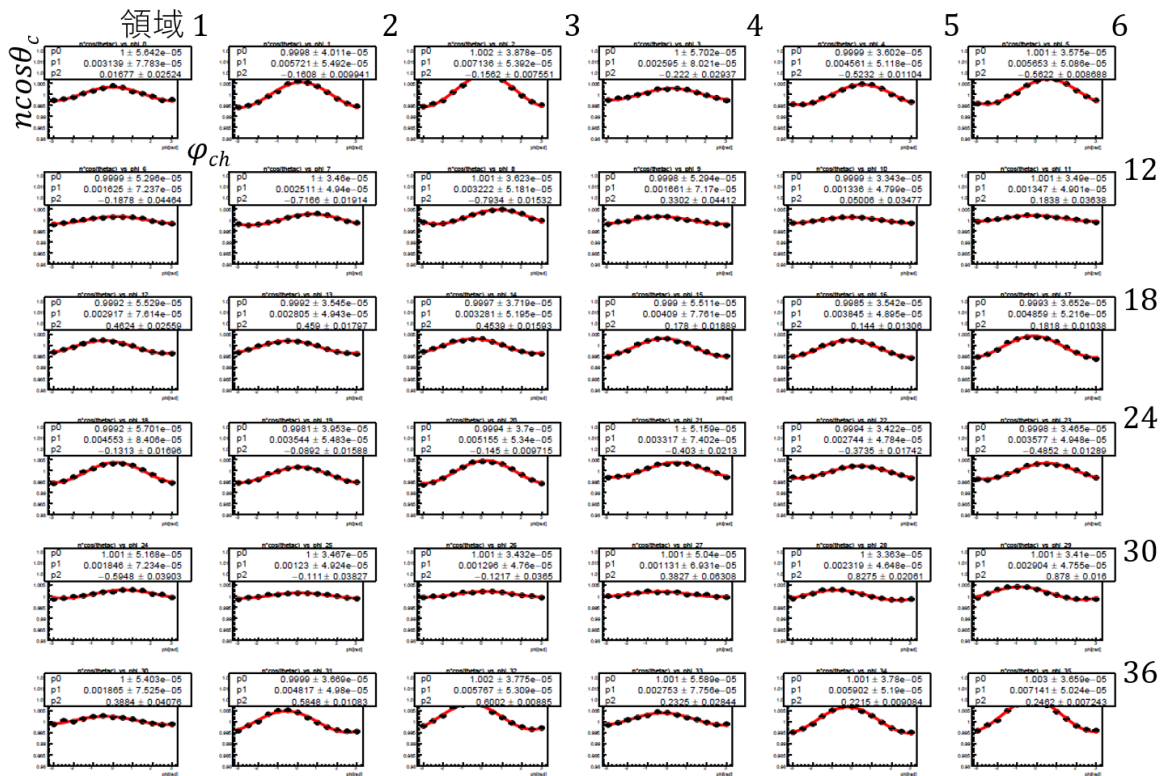


図 4.17 ARICH の回転方向  $y$  軸  $0.5^\circ$  ずらしたときのシミュレーション結果のフィット

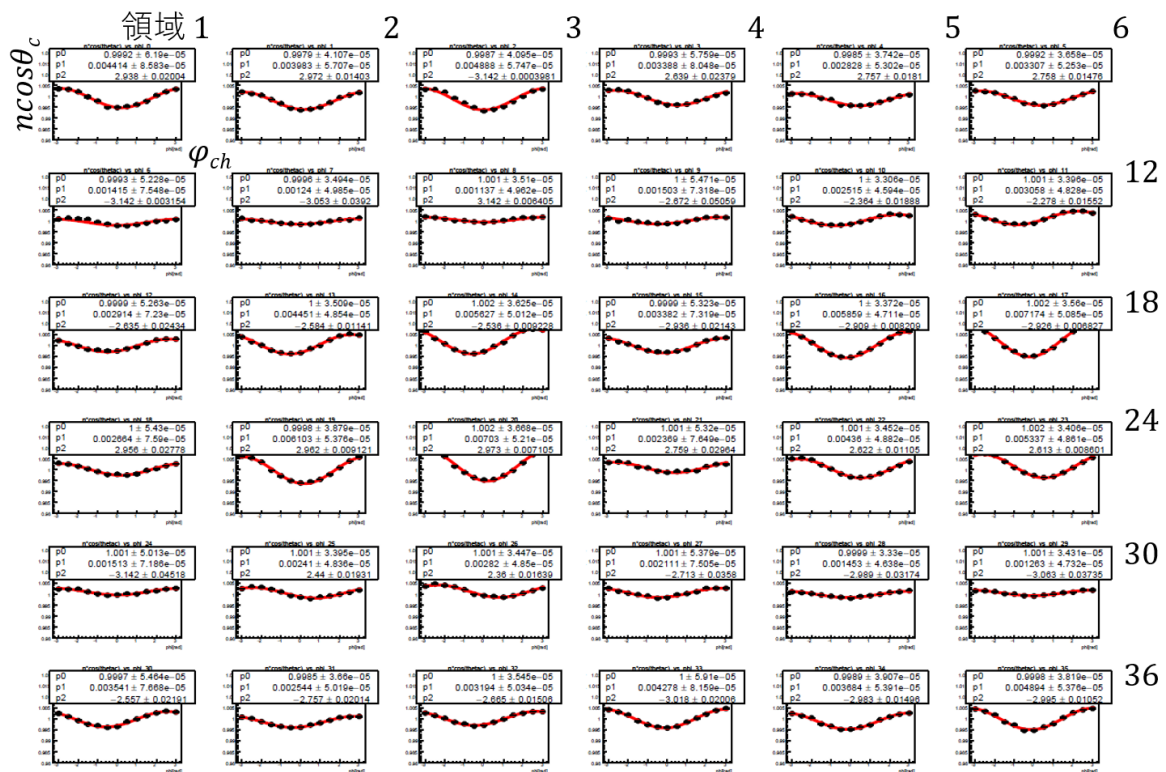


図 4.18 ARICH の回転方向  $y$  軸  $-0.5^\circ$  ずらしたときのシミュレーション結果のフィット

図 4.19, 4.20 は ARICH を回転方向  $z$  軸にそれぞれ  $\pm 0.5^\circ$  ずらしたときの  $n \cos \theta_c$  の  $\phi_{ch}$  依存性を示したグラフである。フィット関数の振幅は 0.001-0.007 程度であり、36 領域においてそれぞれフィット関数の位相や振幅が変化している。

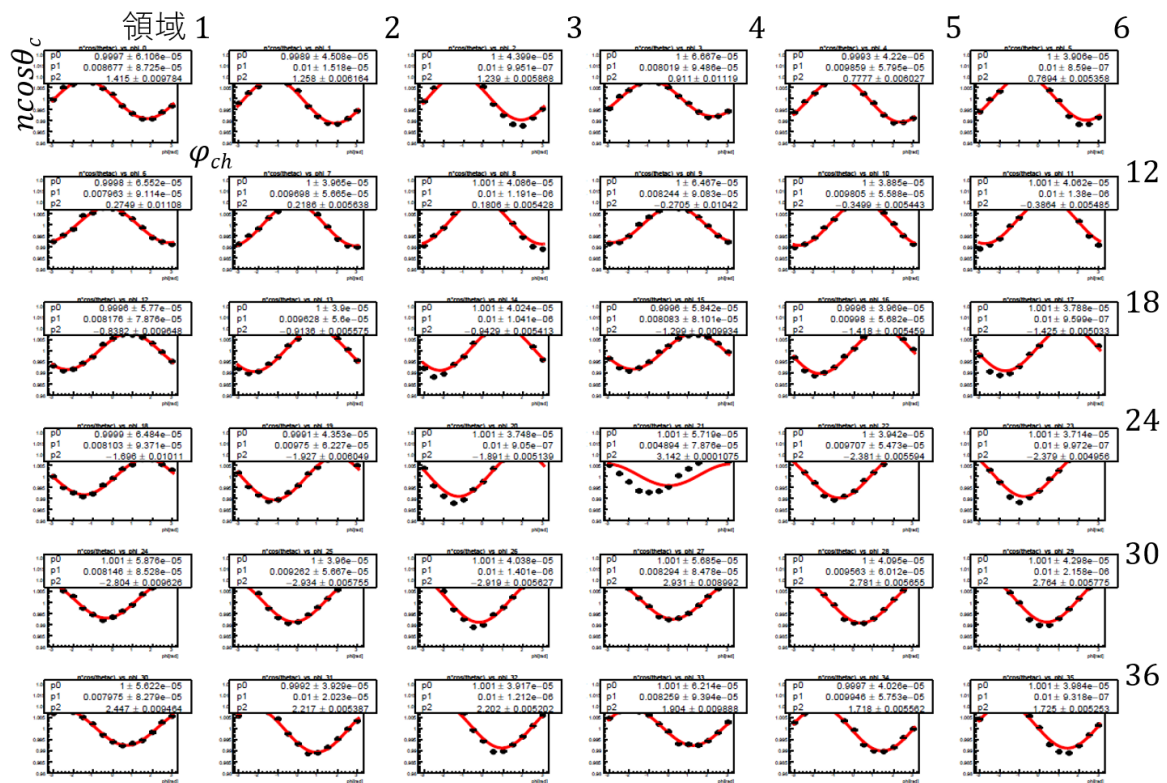


図 4.19 ARICH の回転方向  $z$  軸  $0.5^\circ$  ずらしたときのシミュレーション結果のフィット

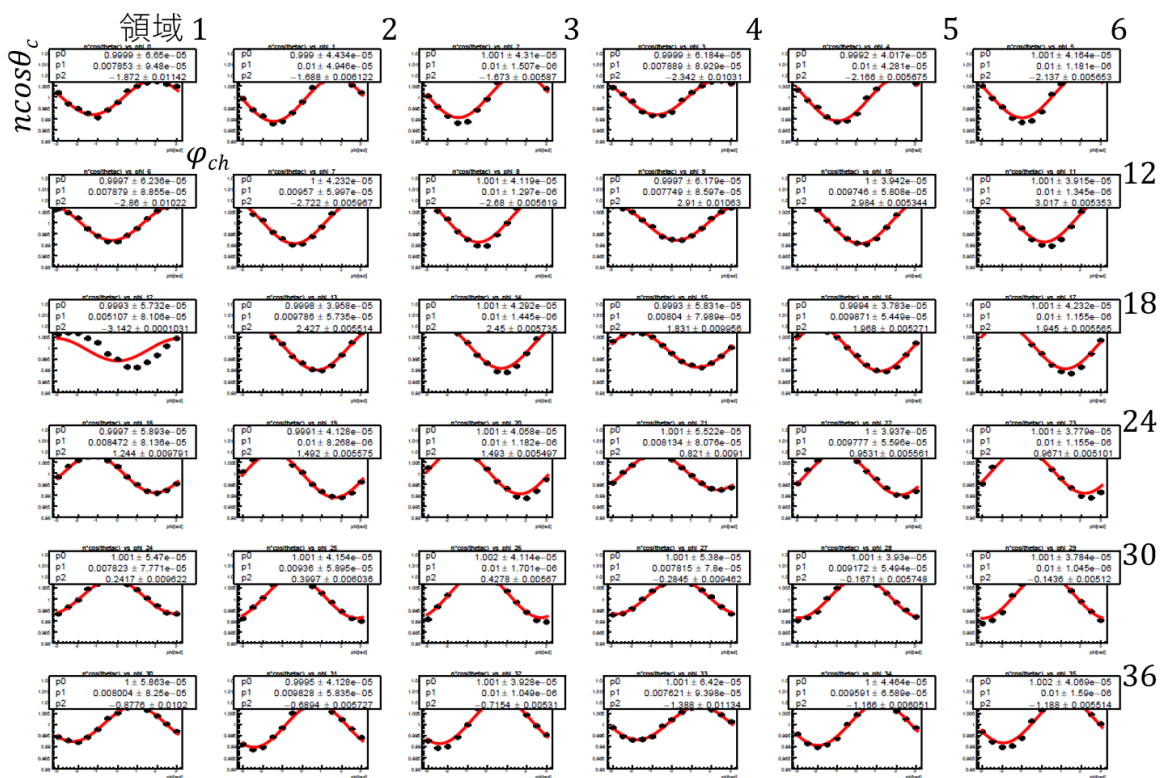


図 4.20 ARICH の回転方向  $z$  軸  $-0.5^\circ$  ずらしたときのシミュレーション結果のフィット



これらの6個の図(図4.15 - 4.20)について、節4.4.1と同様に各36領域におけるcos関数のフィットから得られたそれぞれ振幅と位相の数値を利用し各36領域ごとの仮想的ずれを青色の矢印で表示する。図4.21, 4.22にそれぞれ回転方向における36領域ごとの仮想的ずれのシミュレーション結果の矢印表示を示す。36個の矢印について、cos関数の振幅を矢印の長さ、cos関数の位相を矢印の向きに対応させて表示している。

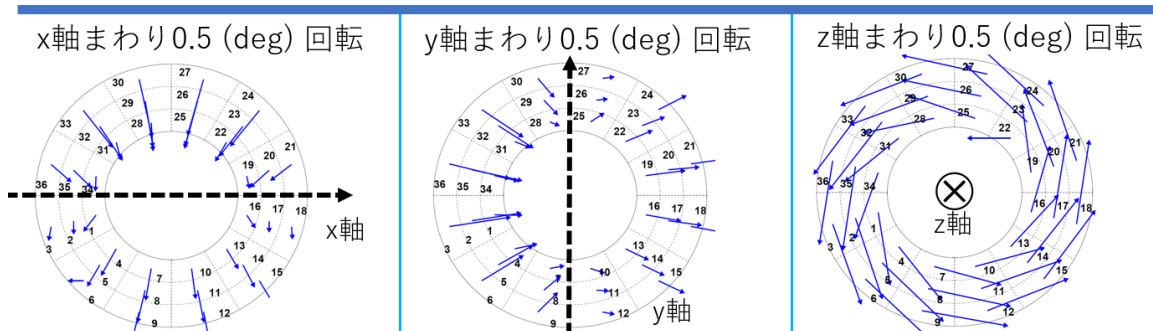


図4.21 ARICHの回転方向x軸、y軸、z軸への正の仮想的ずれのシミュレーション結果の矢印表示

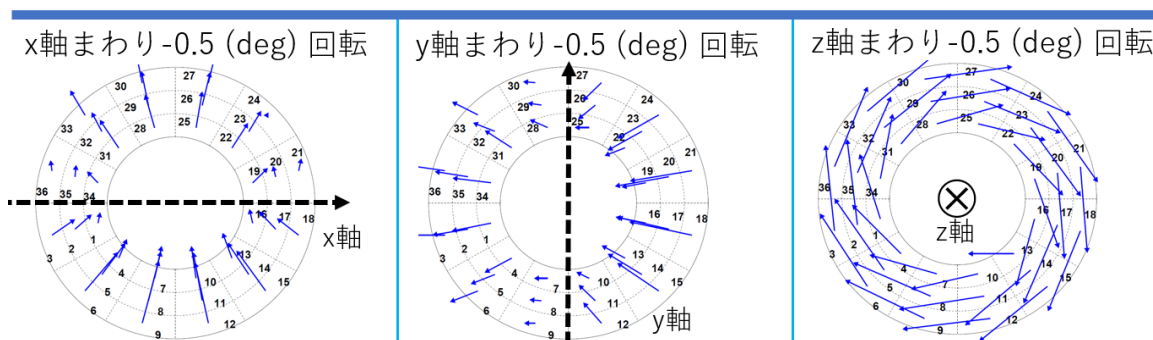


図4.22 ARICHの回転方向x軸、y軸、z軸への負の仮想的ずれのシミュレーション結果の矢印表示

ARICHがx軸まわりに回転する場合はARICHのy軸正と負の部分が大きな影響を受け、y軸まわりに回転する場合はARICHのx軸正と負の部分が大きな影響を受ける。また、ARICHがz軸まわりに回転する場合はARICHがxy平面から見て時計回りまたは反時計回りに回転することを示しており、図4.21, 4.22における矢印もそれぞれ反時計回り、時計回りに表示される。

また、図4.14と同様に先行研究の結果を図4.23に示す。なお先行研究のシミュレーションではずれの値をx軸、y軸、z軸でそれぞれ $1.0^\circ$ 、 $1.0^\circ$ 、 $0.5^\circ$ としている。

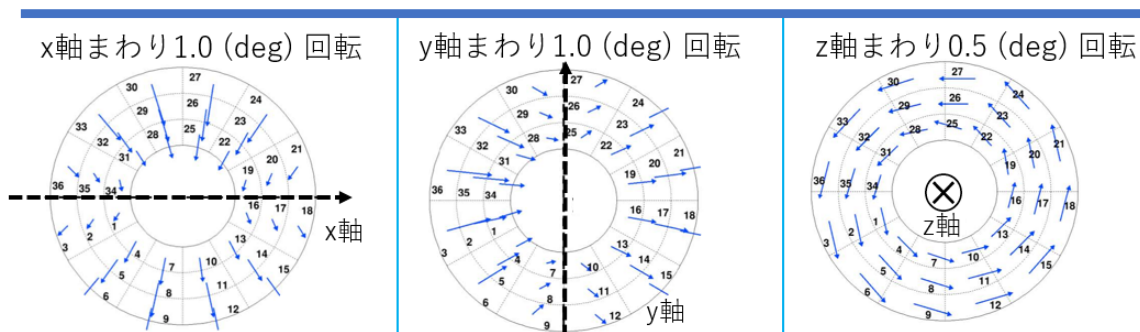


図 4.23 先行研究による回転方向  $x$  軸、 $y$  軸、 $z$  軸の仮想的ずれのシミュレーション結果 [14]

## 4.5 ARICH 検出器の位置のずれの求め方の詳細

この節では ARICH によって取得された  $n \cos \theta_c$  の  $\phi_{ch}$  依存性のデータを利用して位置のずれを求める手法を述べる。

1. 前節で述べたプロセス  $e^+ + e^- \rightarrow \mu^+ + \mu^-$  によるモンテカルロシミュレーションの結果において、36 領域に分割された ARICH の並進方向  $x$  軸、 $y$  軸、 $z$  軸にそれぞれ 2.0 mm, 2.0 mm, 5.0 mm ずらしたとき、回転方向  $x$  軸、 $y$  軸、 $z$  軸にそれぞれ  $0.5^\circ$  ずらしたときの  $n \cos \theta_c$  の  $\phi_{ch}$  依存性の二次元ヒストグラムを用意する。
2. これらのヒストグラムについて、下記 3. と同様の処理を行い、6 種類のずれに対する応答を下記 5. の「シミュレーションデータ」として準備する。
3. フィットを行う実データを用意し、36 領域に分割された  $n \cos \theta_c$  の  $\phi_{ch}$  依存性からデータ点 13 個を作成する。このときデータ点とはヒストグラムにおける  $\phi_{ch}$  を  $-\pi \sim \pi$  の範囲で 13 分割し、13 領域の  $n \cos \theta_c$  に対してそれぞれガウス関数と一次関数でフィットした際のガウス関数の中心値 (Mean) のことをいう。
4. ARICH の並進方向  $x$  軸、 $y$  軸、 $z$  軸のずれの大きさ (単位 mm) と回転方向  $x$  軸、 $y$  軸、 $z$  軸のずれの大きさ (単位  $^\circ$ ) とオフセットの合計七つのフリーパラメータに設定する。
5. 式 (4.1) が最小になるように七つのフリーパラメータを Minuit を用いてフィットする。このとき 36 領域間で位相と振幅は相対的に固定されている。

$$\chi^2 = \sum \frac{(x - a)^2}{x_{err}^2} \quad (4.1)$$

ここで  $x$  は実データの  $36 \times 13$  個の点の  $n \cos \theta_c$  の値、 $x_{err}$  はそれらのエラーの値、 $a$  は規格化した 6 種類のずれに対応するシミュレーションデータをそれぞれフィットパラメータの倍率をかけて足し上げ、さらにオフセットパラメータを足したものである。

## 第 5 章

# ARICH 検出器のずれの見積もりとアラ イメント

### 5.1 使用した実データに関して

本研究では Belle II 実験における Phase III と呼ばれる期間に取得されたデータうち、Bucket 15 のもの (データ取得期間: 2020 年 6 月 4 日 - 2020 年 7 月 7 日) を用いて ARICH のアライメントを行った。データの選別条件としてさらに以下のような条件 a. ~ f. を加えた。

- a. 採用するプロセス:  $e^+ + e^- \rightarrow \mu^+ + \mu^-$  (1,320,939 イベント)
- b. 再構成された粒子の運動量  $p_\mu$  の範囲:  $5 \text{ GeV}/c < p_\mu < 8 \text{ GeV}/c$
- c. 再構成された粒子の ECL におけるエネルギー損失  $E_\mu$  の範囲:  $0.1 \text{ GeV} < E_\mu < 0.4 \text{ GeV}$
- d. 最も大きいエネルギーを検出したシンチレータの CsI 結晶を中心に 9 個の結晶で観測されたエネルギー  $E_9$  と、その結晶を取り囲む 25 個の結晶のうち四隅を除いた 21 個の結晶で観測されたエネルギー  $E_{21}$  の比率:  $\frac{E_9}{E_{21}} > 0.95$
- e. 粒子の飛跡をビーム軸に内挿した点の  $z$  軸の値  $z_0$  と  $x, y$  軸の値の平方和  $d_0 = \sqrt{x^2 + y^2}$  (これらのパラメータをインパクトパラメータと呼ぶ):  $-0.3 \text{ cm} < z_0 < 0.3 \text{ cm}$  かつ  $d_0 < 0.1 \text{ cm}$
- f. 検出器全体のエネルギー和  $E_{tot}$ :  $E_{tot} < 0.1 \text{ GeV}$

条件 a. に関して、節 4.1 で述べた通りプロセス  $e^+ + e^- \rightarrow \mu^+ + \mu^-$  における  $\mu$  粒子は電子と比較し電磁シャワーやハドロン相互作用を発生させないためクリーンな飛跡を得ることができ、本研究のアライメントに適しているために採用した。

条件 b. に関して、ARICH の前方に射出される  $\mu$  粒子の運動量は電子・陽電子ビームの運動量  $7 \text{ GeV}/c$  よりも若干低くなるためにこのような範囲に設定した。

条件 c. に関して、Bhabha 散乱による電子は放射損失でほぼ全てのエネルギーを失うのに対して  $\mu$  粒子は電離損失による  $200 \text{ MeV}$  程度のエネルギー損失をするため、 $\mu$  粒子ペアのエネルギー損失は合計  $400 \text{ MeV}$  程度となる。したがってこのような条件をかけた。

条件 d. に関して、 $\mu$  粒子は電離損失により粒子の飛跡に沿ったエネルギー損失をし  $E_9$  と  $E_{21}$  の比率  $\frac{E_9}{E_{21}}$  の値はほぼ 1 となるためにこのような条件をかけた。なお、ハドロンは  $\mu$  粒子と比較した場合ハドロン相互作用により粒子の飛跡の少し離れた位置でもエネルギー損失が起こるため  $\frac{E_9}{E_{21}}$  の値は 1

よりも大きくなる。

条件 e. に関して、粒子の飛跡が正しく再構成されている場合飛跡を電子・陽電子のビーム軸に内挿すると内挿点はビームの衝突点付近になるため、内挿点のインパクトパラメータは共にほぼ 0 に近い値をとる。したがってこのような範囲を設定した。

条件 f. に関して、プロセス  $e^+ + e^- \rightarrow \mu^+ + \mu^-$  と比較して Bhabha 散乱 ( $e^+ + e^- \rightarrow e^+ + e^-$ ) や他にも  $\gamma$  線の放出を伴う  $\mu$  粒子ペアの散乱 ( $e^+ + e^- \rightarrow \mu^+ + \mu^- + \gamma$ ) の場合は電子や  $\gamma$  線によるエネルギー損失が大きいため、検出器全体のエネルギー和  $E_{tot}$  の値の範囲はこのような条件をかけた。

図 5.1 に Bucket 15 で取得され選別条件が加味された実データにおける  $n \cos \theta_c$  の二次元ヒストグラムを示す。

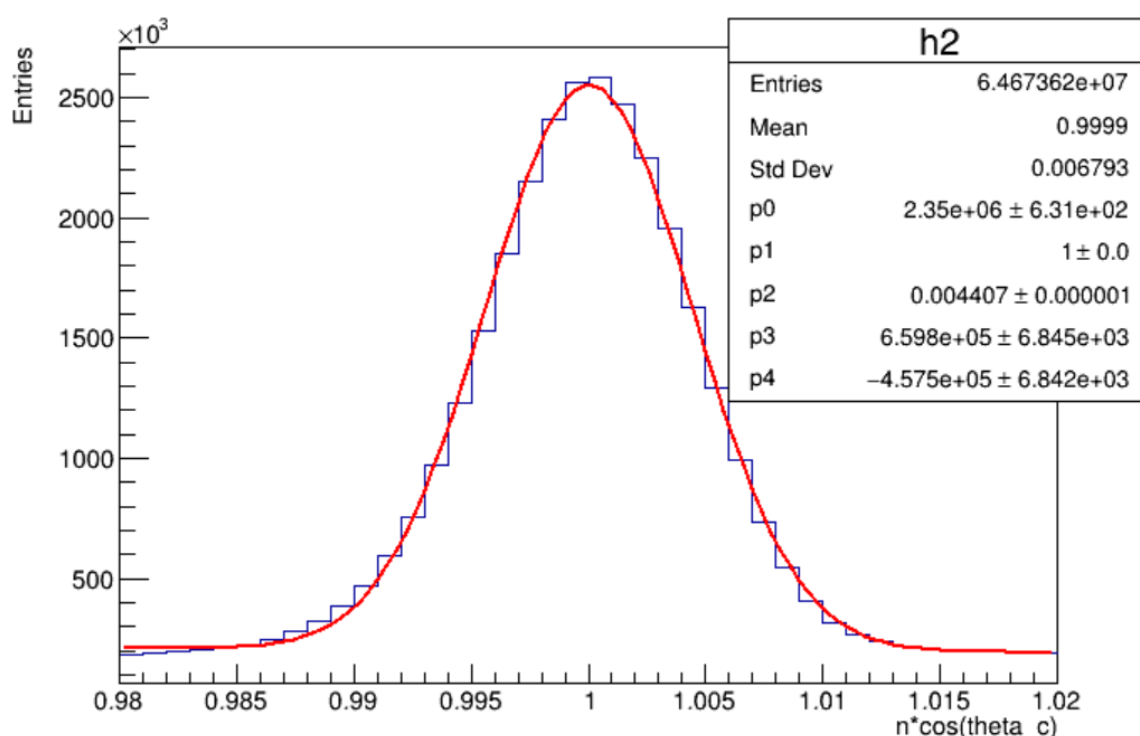


図 5.1 Bucket 15 で取得されたデータの  $n \cos \theta_c$

また、図 5.2 に Bucket 15 で得られた実データにおける  $n \cos \theta_c$  の  $\phi_{ch}$  依存性の二次元ヒストグラム (左) とそのフィット結果 (右) を示す。なお、この際のフィットは節 4.5 で述べた方法と同じ方法を用いている。図 5.2 (左) は図 4.4 と比較してヒストグラムにおけるイベント数が多い bin は  $n \cos \theta_c = 1.000$  付近に集中している一方で、ヒストグラムにおけるイベント数の分布が  $n \cos \theta_c = 1.000$  の軸を中心に上下非対称に分布しているため  $n \cos \theta_c$  は  $\phi_{ch}$  依存性を有する可能性があることを示している。さらにより定量的な  $\phi_{ch}$  依存性を確認するためにヒストグラムを  $\cos$  関数でフィットを行った図 5.2 (右) の結果を見ると、フィット関数の振幅と位相の値 (それぞれ図 5.2 (右) における  $p_1, p_2$ ) は

$$p_1 = (1.4583 \pm 0.0009) \times 10^{-3} \quad (5.1)$$

$$p_2 = 2.2289 \pm 0.0006 \quad (5.2)$$

となっており  $p_1$  の値が 0 でないため、図 5.2 の二次元ヒストグラムにおける  $n \cos \theta_c$  は有意な  $\phi_{ch}$  依存性を持っているということが言え、以上のことからつまり ARICH は設計上の正しい位置からずれているということが言える。

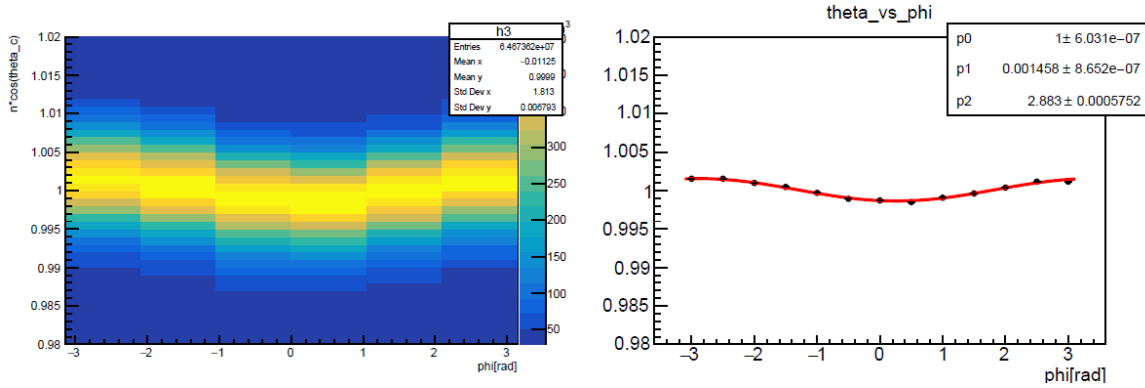


図 5.2 Bucket 15 の実データにおける (左)  $n \cos \theta_c$  の  $\phi_{ch}$  依存性の二次元ヒストグラムと (右) そのフィット結果

この実データを利用して ARICH の設計上の位置からのずれを算出するにあたり節 4.4.1 で述べたようにずれが位置依存性を持つ可能性を考慮する必要があるため、ARICH を 36 個の領域に分割し各領域ごとの二次元ヒストグラムを用いてフィットを行った。図 5.3 に ARICH を 36 領域に分割した際の各領域ごとの  $n \cos \theta_c$  の  $\phi_{ch}$  依存性を表す二次元ヒストグラムを示す。なお、ARICH の内側に位置する領域 13, 16, 19, 22 では統計量が不足しているためヒストグラム内に一部イベントが無い部分が存在した。これは Belle II 実験においてはルミノシティ向上のために電子・陽電子ビームを角度をつけて衝突させるというナノビーム大角度交差衝突方式 (節 2.1, 2.2.2 参照) を用いており、この影響によって粒子がローレンツブーストを起こし一方は ARICH の  $x$  軸正の方向へ、もう一方は ARICH ではない方向に飛んで行ってしまい検出されないためである。

図 5.3 より各領域のヒストグラムごとにイベントが集中しているビンの分布が異なっているように見えるため、ARICH のずれは位置依存性を有するというを示している。

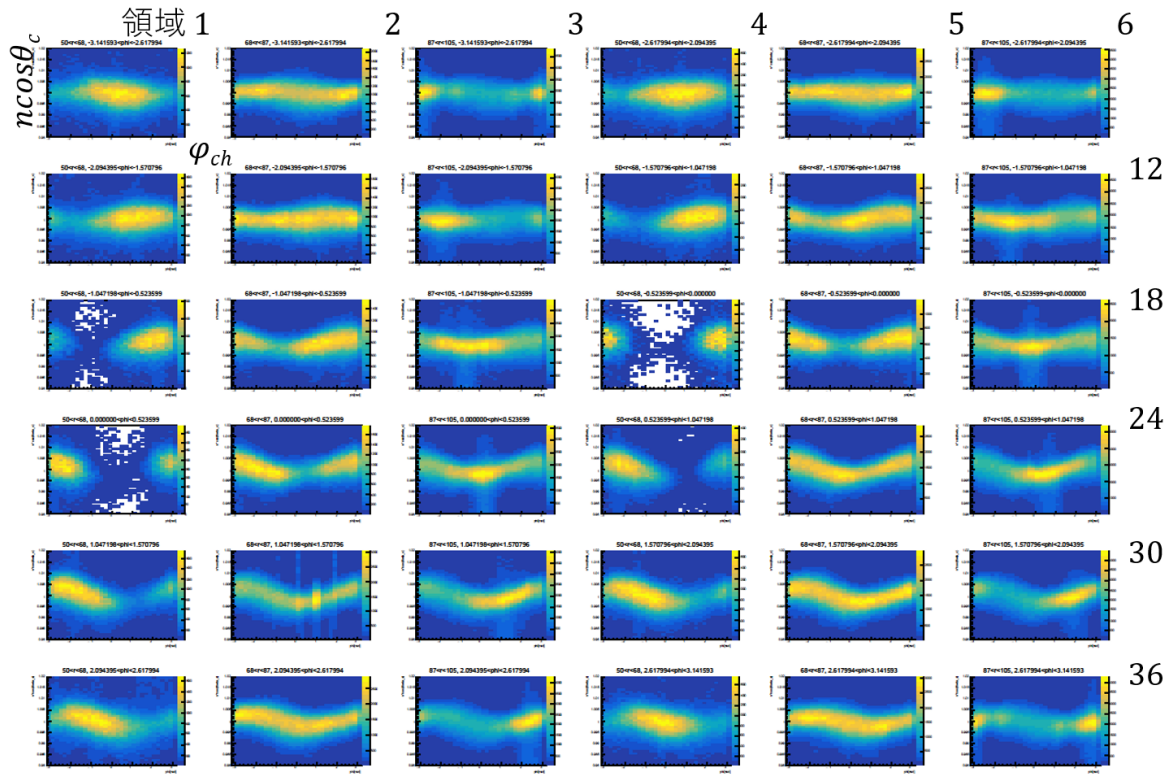


図 5.3 実データにおいて 36 領域に分割された ARICH における  $n \cos \theta_c$  の  $\phi_{ch}$  依存性

## 5.2 実データを用いたずれの算出結果

節 5.1 で述べた実データを利用し 36 領域に分割された ARICH の  $n \cos \theta_c$  の  $\phi_{ch}$  依存性が定量的にどの程度かを調べることで、ARICH の並進方向  $x$  軸、 $y$  軸、 $z$  軸、そして回転方向  $x$  軸、 $y$  軸、 $z$  軸の合計 6 方向のずれの値を求める。

図 5.4 は図 5.3 の 36 領域ごとの二次元ヒストグラムをそれぞれ  $\cos$  関数でフィットした際のフィット結果である。図 5.4 の結果を見ると、フィット関数の振幅と位相の値 (それぞれ図 5.4 における  $p_1$ 、 $p_2$ ) が領域ごとに異なりさらにどの領域においても  $p_1 \neq 0$  となっているため、図 5.3 のヒストグラムは全ての領域において  $n \cos \theta_c$  は有意な  $\phi_{ch}$  依存性を持っていると言える。以上のことからつまり ARICH のずれには位置依存性が認められ、これを定量的に表現するには六つの異なる方向ごとにずれを記述する必要があることが分かる。

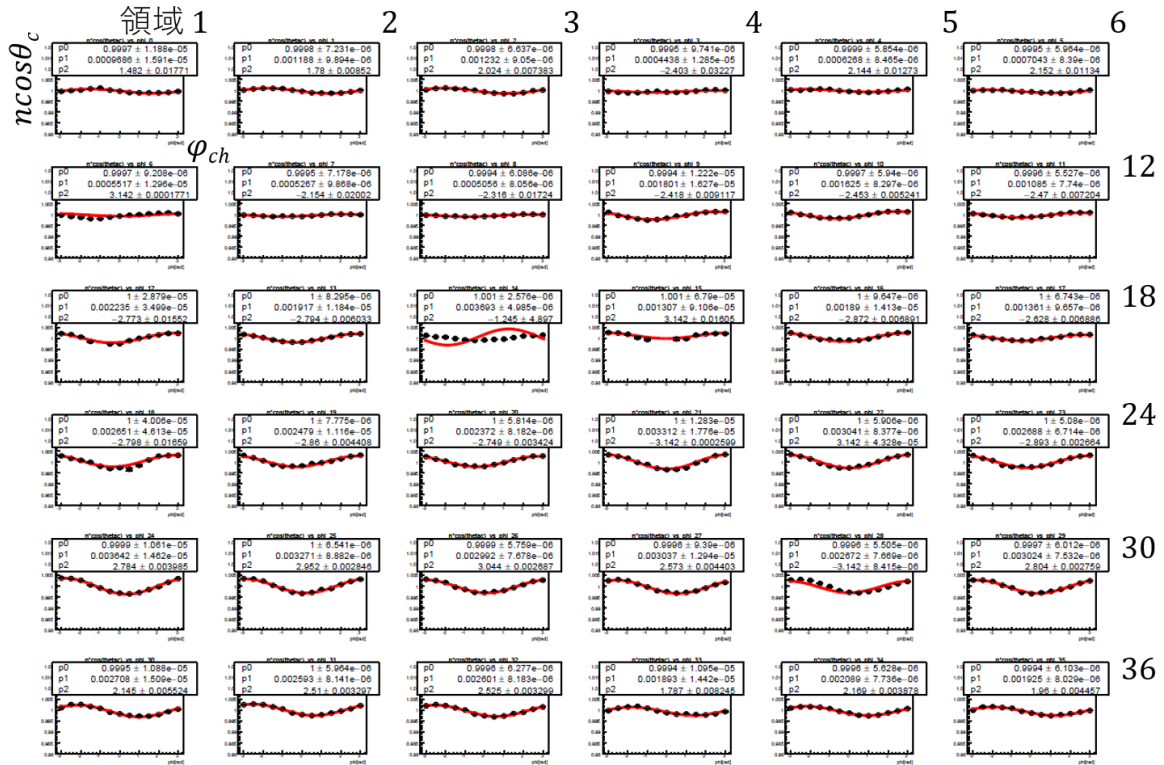


図 5.4 36 領域に分割された実データのフィット結果

図 5.4 で得られた 36 領域ごとの ARICH のずれのフィット結果と、先行研究によって得られたモンテカルロシミュレーションによる ARICH の仮想的ずれの結果 (図 4.14, 4.23) を線形結合することによって ARICH のずれの値を 6 方向ごとに算出した。さらにそれらずれの値を元に ARICH のずれを再構成することによって結果に再現性があるかを確認した。図 5.5 に ARICH における 36 領域ごとのデータ点  $13 \times 36$  個と、それに対応した各領域ごとに再構成されたずれの比較図を示す。36 個のグラフにおける黒い曲線が算出されたずれの数値を元に再構成された ARICH のずれを表しており、データ点とこの曲線が重なっていればいるほどずれの再構成がうまくいっていることを示しており、データ点と曲線が重なっていなければずれの再構成ができていないことを示している。

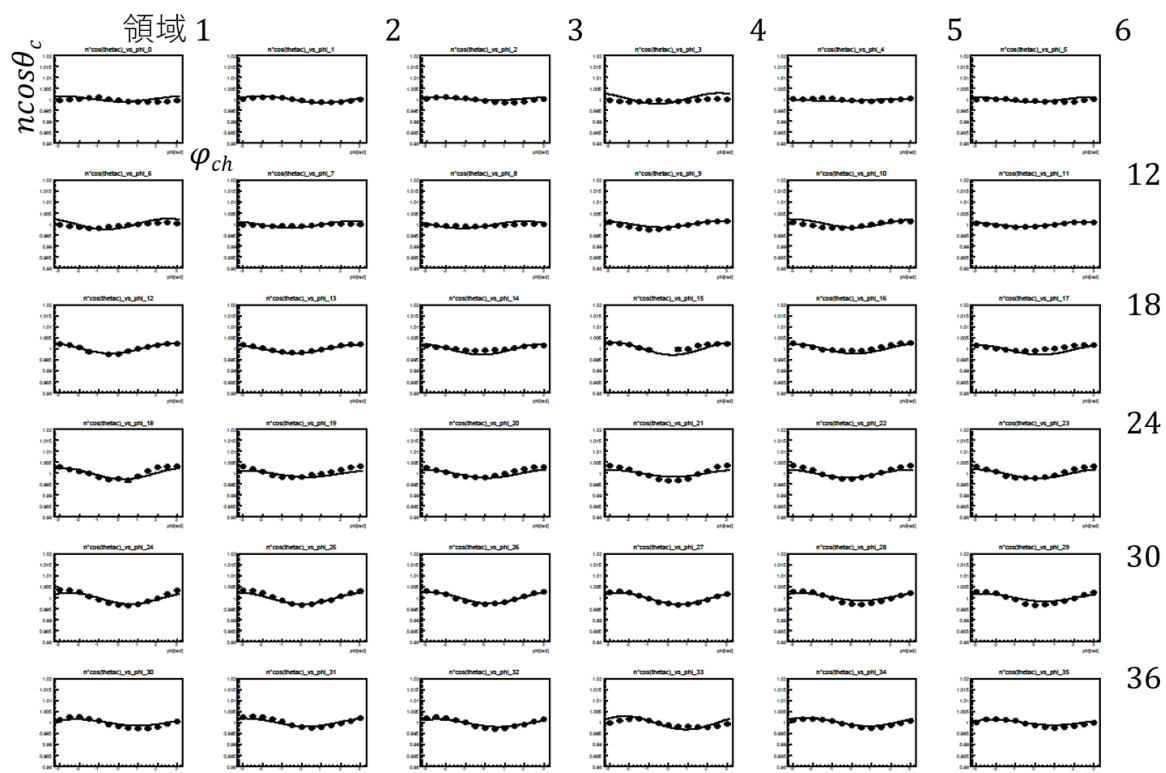


図 5.5 ARICH 各領域におけるデータ点とずれの値から再構成されたずれの比較

図 5.4 のフィットにより得られた  $\cos$  関数の振幅と位相の値  $p_1, p_2$  を用いて、ARICH の 36 領域ごとのずれを 36 個の矢印で表した。節 4.4.1 で述べた際と同様に  $p_1$  の値を矢印の長さ、 $p_2$  の値を矢印の向きに対応させて表示している。図 5.6 にこの矢印表示の結果を示す。



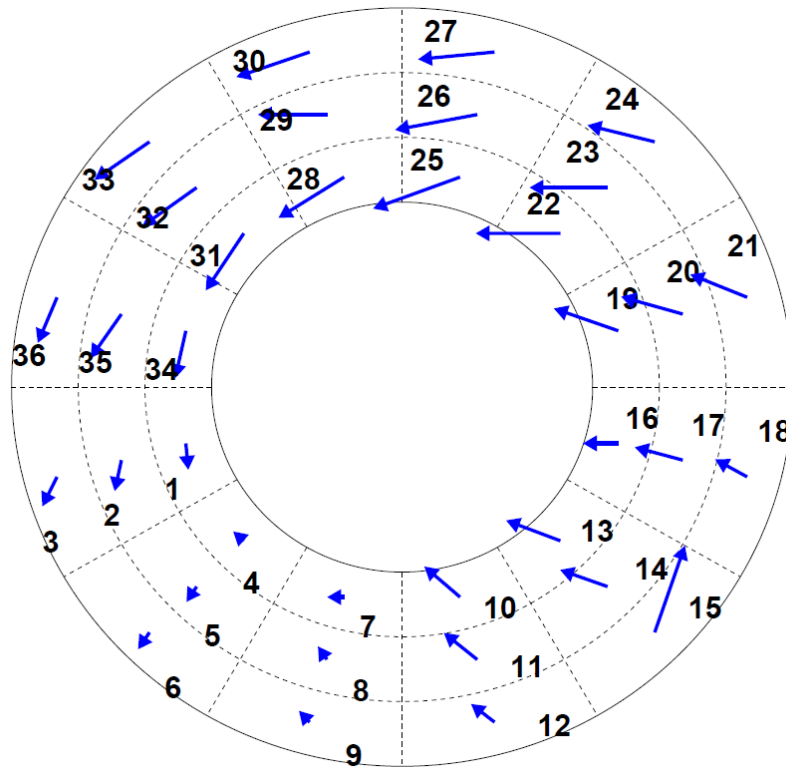


図 5.6 フィット関数の振幅と位相によって表現された ARICH のずれの矢印表示

ARICH の 6 方向のずれの値は、先行研究によって得られたモンテカルロシミュレーションによる ARICH のずれの結果 (図 4.14, 4.23) と図 5.4 の実データを用いた際の ARICH のずれの矢印表示の結果の二つの解析結果を線形結合することによって表現する。以下の表 5.1 に Bucket 15 を用いて計算した場合の ARICH の 6 方向のずれの値を示す。

表 5.1 Bucket 15 を用いた場合の ARICH のずれの値

	並進移動 [mm]	回転移動 [°]
x 軸	$-1.228 \pm 0.001$	$2.561 \pm 0.006$
y 軸	$-0.4222 \pm 0.0021$	$-1.197 \pm 0.005$
z 軸	$1.696 \pm 0.003$	$1.389 \pm 0.003$

ARICH の設計において予測されているずれとしては、主に ECL 側の設置されている構造体がネジで止められているためその部分が 1 mm 程度動く可能性があり、また全体的な Belle II 測定器本体との取り付けの不定性も考えられるため、並進方向には 2 mm 程度、回転方向には  $0.10^\circ$  程度のずれが予測されている。これを表 5.1 と照らし合わせると全体的に回転方向のずれの値が予測よりも少し大きく計算される結果となった。

表 5.1 で得られた ARICH の 6 方向のずれの値にオフセットを加えた 7 個のパラメータの値を元に再構成された ARICH のずれの矢印表示を図 5.7 に示す。図 5.6 では矢印表示にフィット関数であ

る  $\cos$  関数の振幅と位相を用いていたのに対し、図 5.7 では算出された ARICH の六つのずれの値とオフセットの合計七つのパラメータの値を用いることによって ARICH のずれの値からずれの大きさと方向を再構成して表示しているという点が異なっている。なお、図 5.7 と図 5.6 において矢印のスケールは等しくなるように設定している。

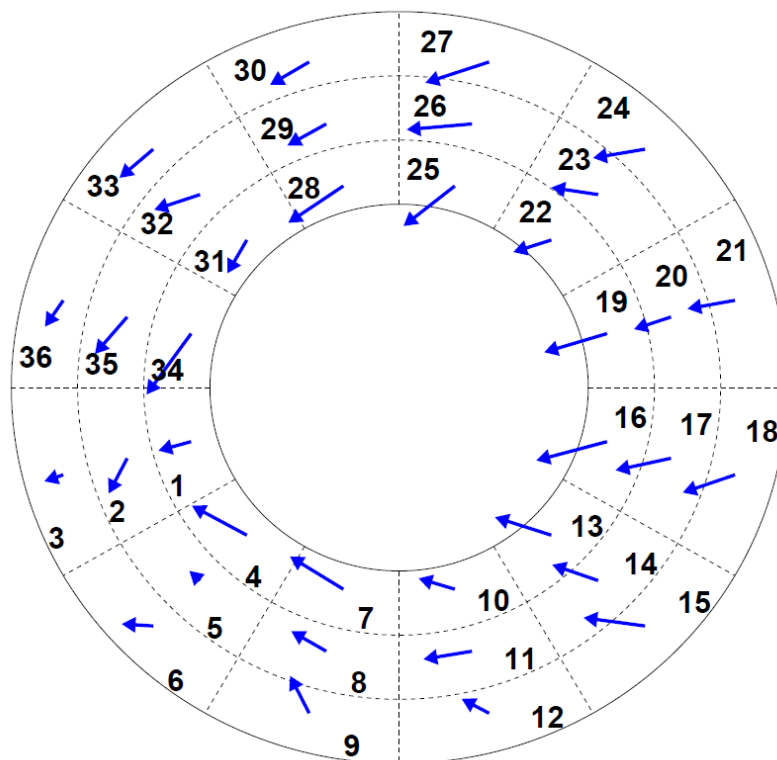


図 5.7 7 個のパラメータによって表現された ARICH のずれの矢印表示

図 5.6 と 5.7 を比較することにより、ARICH のずれの再構成前後の傾向が目で確認できるようになる。各領域において矢印の大きさと向きが一致していればしているほどずれの再構成が正しく出来ていることを表し、一方傾向の不一致が見られる領域ではずれの再構成が正しく行われていないということを表している。実際にこれら二つの図を比較すると、各領域において矢印の大きさと向きのおおよその傾向は一致しているものの厳密には矢印の向きや長さが異なって見える領域が存在するため、ARICH の 6 方向のずれ再構成はさらに正しくなる余地を残しているということが言える。

### 5.3 結果の改良の余地

先行研究の手法で行った ARICH のずれの算出結果は、Bucket 15 で取得されたデータを使用した場合 6 方向ごとのずれを表現することができたが、一方で表 5.1 で示されていたとおり回転方向のずれの大きさが想定されるずれの値よりも大きく算出されてしまい、また 4.5 で述べたアライメントフィットの際に同時に計算された  $\chi^2$  を自由度  $n = 36 \times 13 - 7$  で割った値は  $\chi^2/n = 1493.58$  となり、かなり大きくなってしまった。これでは統計学的に正しいアライメントフィットが行われているとは

言い難いため、より高精度な手法を用いてアライメントフィットをやり直す必要がある。

また、図 5.6 と図 5.7 の比較をしてみると ARICH のずれにはそもそも並進方向  $x$  軸、 $y$  軸、 $z$  軸、回転方向  $x$  軸、 $y$  軸、 $z$  軸の計 6 方向での記述では表現しきれない小さなずれがまだ残っており、それが  $\chi^2$  の値に影響を及ぼしているとも考えられるので、この 6 方向を表すパラメータの他に新しいパラメータを導入することの検討も必要である。

## 第 6 章

# ARICH 検出器のアライメント手法の改良とその結果

節 4.3 で述べたように先行研究での ARICH のアライメントの手法にはいくつか問題点があり、それらを改善し新たな手法でアライメントを行うことによって ARICH のさらなる粒子識別性能の向上が見込める。この章では実際に新しいアライメントフィットの手法の開発と、その結果アライメントの精度が先行研究と比較してどのように変わったかを述べる。

### 6.1 アライメントフィットの結果の精度向上

表 5.1 で述べた結果を用いて ARICH のアライメントを行った結果、 $n \cos \theta_c$  の  $\phi_{ch}$  依存性が劇的に減少することは無く、アライメントフィットを行った際に計算した  $\chi^2$  の値もかなり大きくなってしまったため新たな手法でアライメントフィットをやり直し精度を向上させる必要がある。そこで、本研究のアライメントでは大きく分けて二つの新しい作業を追加した。

まずは、節 4.4.1, 4.4.2 でも述べたようにプロセス  $e^+ + e^- \rightarrow \mu^+ + \mu^-$  によるモンテカルロシミュレーションを作成した。アライメントフィットの精度を向上させるために正負両方の仮想的ずれのファイルを作成し、合計  $6 \times 2 = 12$  個の仮想的ずれの結果ファイルを用意した。

さらに、節 4.5 で述べた ARICH のずれの求め方の詳細のうち、3. の 13 個のデータ点を  $\cos$  関数でフィットするというところ以降を異なる手法を用いて新しい ARICH のずれの求め方を作成した。以下にはまず ARICH のずれをフィットすの際に同時に計算される  $\chi^2$  の値に関して、考慮すべき誤差の種類を見直した結果新たに採用した  $\chi^2$  の計算式と、さらにそれらの計算をする機能を組み込み実際に Bucket 15 で取得された実データを用いて ARICH のずれを計算するコードの詳細を述べる。

統計学的にフィットの精度を高めるためには、取得された 13 個のデータ点をフィットする際に来るだけ適した関数を選びさらにフィットの際の初期値などをうまく選んでやる必要がある。しかし、Bucket 15 で取得された実データの 36 領域ごとのデータ点 13 個を詳しく調べた結果  $\cos$  関数ではフィットしきれない形であることが判明したため、先行研究で行われていたような各領域のデータ点 13 個を  $\cos$  関数でフィットをすることの代わりに、本研究では ARICH の 36 領域ごとのデータ点  $36 \times 13$  個それぞれにおいて個別にフィットを試みた。図 6.1 に例として並進方向  $z$  軸に関して、あるデータ点のずれの値を  $t_z$ 、 $f = n \cos \theta_c - 1$  とした場合の  $t_z$  と  $f$  の関係図を示す。

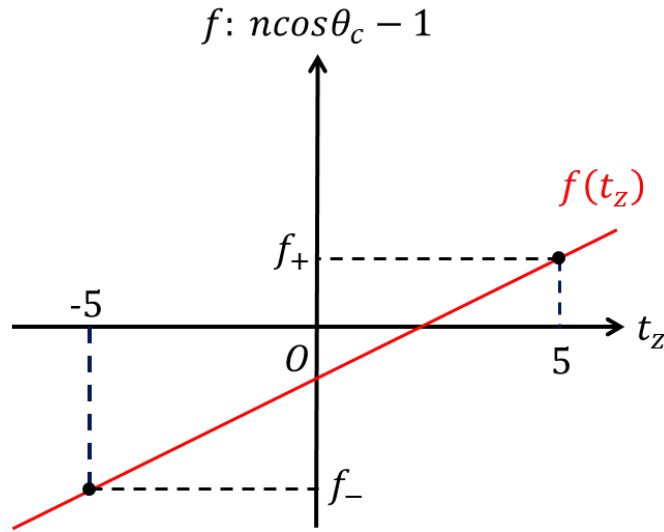


図 6.1 並進方向  $z$  軸における  $t_z$  と  $f$  の関係

ある点  $t_z$  における関数  $f(t_z)$  (図 6.1 における赤い直線) の式は、シミュレーションにおいて並進方向  $z$  軸の仮想的ずれの値が  $\pm 5$  mm のときの  $f$  の値をそれぞれ  $f_{\pm}$ 、シミュレーションによる仮想的ずれの値の絶対値を  $l_{denom}$  として式 (6.1) のように表される。

$$\begin{aligned} f(t_z) &= \frac{f_+ - f_-}{2l_{denom}} t_z + \frac{f_+ + f_-}{2} \\ &= \frac{f_+ - f_-}{10} t_z + \frac{f_+ + f_-}{2} \end{aligned} \quad (6.1)$$

また、ある点  $t_z$  における関数  $f(t_z)$  の誤差関数  $\Delta f(t_z)$  は、並進方向  $z$  の仮想的ずれの値が  $\pm 5$  mm のときの誤差をそれぞれ  $\Delta f_{\pm}$  として式 (6.2) と表される。

$$\begin{aligned} \Delta f(t_z) &= \sqrt{\left(\frac{1}{2} + \frac{t_z}{2l_{denom}}\right)^2 \Delta f_+^2 + \left(\frac{1}{2} - \frac{t_z}{2l_{denom}}\right)^2 \Delta f_-^2} \\ &= \sqrt{\left(\frac{1}{2} + \frac{t_z}{10}\right)^2 \Delta f_+^2 + \left(\frac{1}{2} - \frac{t_z}{10}\right)^2 \Delta f_-^2} \end{aligned} \quad (6.2)$$

図 6.2 に ARICH の並進方向  $z$  軸においてモンテカルロシミュレーションの仮想的ずれの値を  $t_z = 5$  mm としたときの関数  $f(t_z)$ 、 $\Delta f(t_z)$  の値の計算結果を示す。36  $\times$  13 個の誤差付きプロットがそれぞれ図 6.1 で定義した  $f(t_z)$ 、 $\Delta f(t_z)$  を表している。

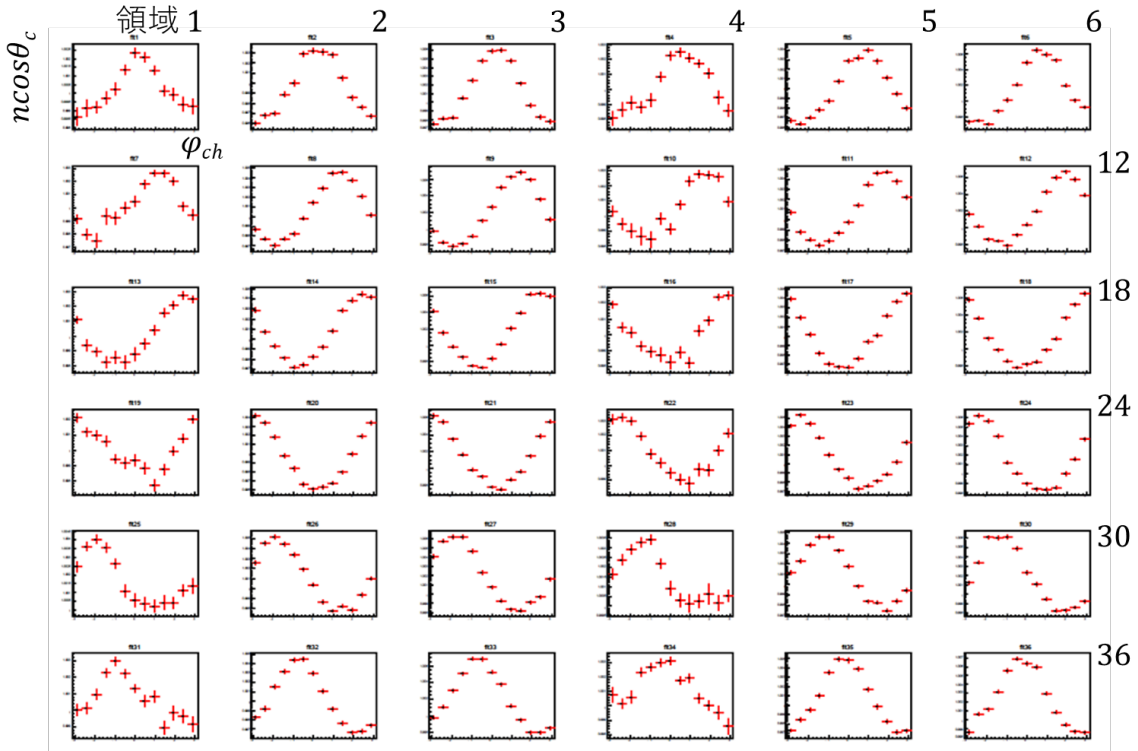


図 6.2 ARICH を並進方向  $z$  軸に 5 mm ずらしたときのモンテカルロシミュレーションによる  $f(t_z)$ ,  $\Delta f(t_z)$  のプロット

また、モンテカルロシミュレーションにおいて ARICH のずれが全く無い場合の関数  $f = n \cos \theta_c - 1$  を 6 方向分の関数 6 個について平均したものを  $f_{zero,k,i}$  を式 (6.3) と定義する。なお、簡単のために並進方向  $x, y, z$  軸、回転方向  $x, y, z$  軸の計 6 方向を順番に 0 - 5 と数字で表している。例えば、6 方向での関数  $f_p$  は順番に  $f_0, f_1, f_2, f_3, f_4, f_5$  ( $p = 0, 1, 2, 3, 4, 5$ ) となる。さらに、分割された ARICH の 36 領域のうちの領域  $k$ 、13 個のデータ点のうち  $\phi_{ch}$  の値が  $i$  番目に小さいものの関数  $f$  を  $f_{p,k,i}$  ( $k = 0, 1, \dots, 35, i = 0, 1, \dots, 12$ ) としている。

$$f_{zero,k,i} \equiv \frac{f_{0,k,i}(0) + f_{1,k,i}(0) + f_{2,k,i}(0) + f_{3,k,i}(0) + f_{4,k,i}(0) + f_{5,k,i}(0)}{6} \quad (6.3)$$

図 6.3 に式 (6.3) で定義された  $f_{zero,k,i}$  の計算結果を示す。

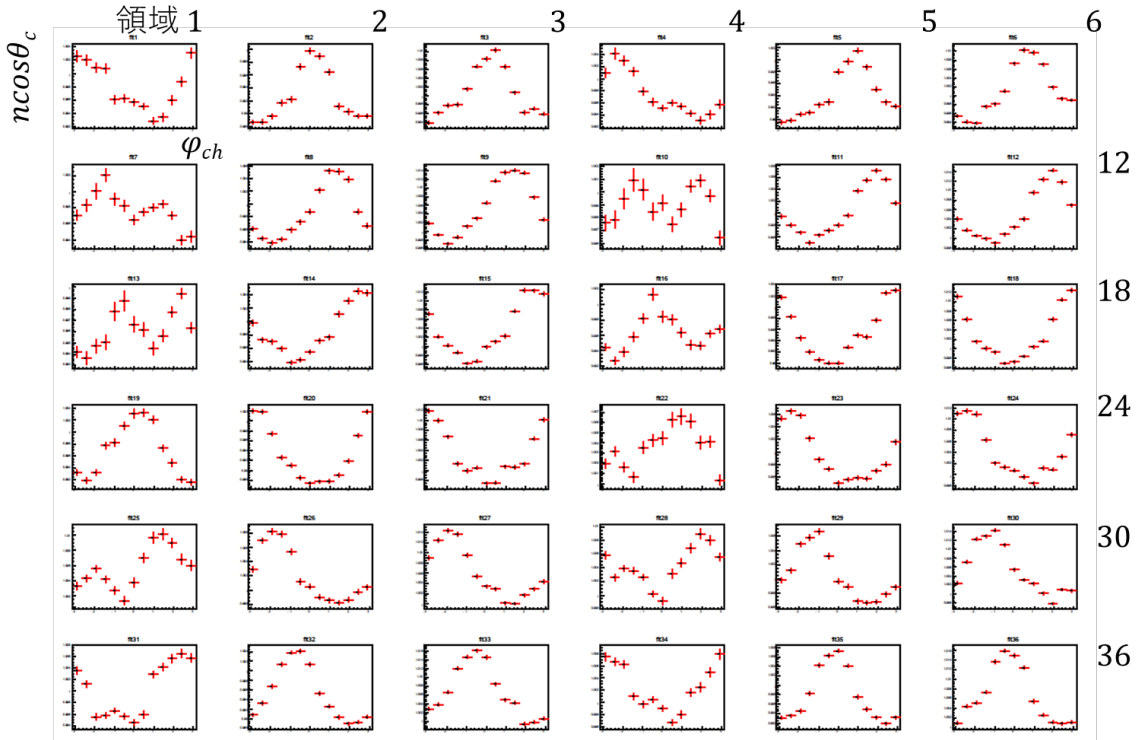


図 6.3 ARICH にずれが無い場合のモンテカルロシミュレーションによる  $f(t_z)$ ,  $\Delta f(t_z)$  のプロット

$f_{p,k,i}$  ( $p = 0, 1, \dots, 5, k = 0, 1, \dots, 35, i = 0, 1, \dots, 12$ ),  $f_{zero,k,i}$  の値と、Bucket 15 で取得された実データにおける  $n \cos \theta_c$  の値  $d_{k,i}$ 、 $n \cos \theta_c$  の誤差の値  $\Delta d_{k,i}$  (それぞれ  $36 \times 13$  個) を用いることによって 6 方向の各フィット関数からそれぞれ  $f_{zero,k,i}$  を引いたものと実データから  $f_{zero,k,i}$  を引いたものを比較しこれを  $\chi^2$  とする。フィットパラメータは先行研究で用いられていたものと同じく ARICH の並進方向  $x, y, z$  軸、回転方向  $x, y, z$  軸の計 6 方向のずれの値  $t_x, t_y, t_z, t_\alpha, t_\beta, t_\gamma$  とアライメントフィットのオフセット  $t_o (= t_{p=6})$  の値の計 7 個を用いると、求める  $\chi^2$  の式は式 (6.4) と表される。

$$\begin{aligned}
 & \chi^2(t_x, t_y, t_z, t_\alpha, t_\beta, t_\gamma, t_o) \\
 &= \sum_{k=0}^{35} \sum_{i=0}^{12} \frac{(d_{k,i} - f_{0,k,i}(t_x) - f_{1,k,i}(t_y) - f_{2,k,i}(t_z) - f_{3,k,i}(t_\alpha) - f_{4,k,i}(t_\beta) - f_{5,k,i}(t_\gamma) - t_o + (6-1)f_{zero,k,i})^2}{\Delta d_{k,i}^2 + \Delta f_{0,k,i}^2(t_x) + \Delta f_{1,k,i}^2(t_y) + \Delta f_{2,k,i}^2(t_z) + \Delta f_{3,k,i}^2(t_\alpha) + \Delta f_{4,k,i}^2(t_\beta) + \Delta f_{5,k,i}^2(t_\gamma)} \\
 &= \sum_{k=0}^{35} \sum_{i=0}^{12} \frac{(d_{k,i} - \sum_{p=0}^5 f_{p,k,i}(t_p) - t_{p=6} + 5f_{zero,k,i})^2}{\Delta d_{k,i}^2 + \sum_{p=0}^5 \Delta f_{p,k,i}^2(t_p)} \tag{6.4}
 \end{aligned}$$

式 (6.4) のように求まった  $\chi^2$  に関する新しい計算式を用いて、新しい手法で  $\chi^2$  の計算をしてアライメントフィットをするコードを作成した。コードは大きく分けてプログラム (1)、プログラム (2) の 2 種類作成し、プログラム (1) は主に  $\chi^2$  の計算に必要な  $f_{\pm}$ ,  $\Delta f_{\pm}$  の値の取得を目的とし、プログラム (2) は (1) で得られた  $f_{\pm}$ ,  $\Delta f_{\pm}$  の値を用いて実際に  $\chi^2$  を計算し、アライメントフィットをすることを目的としたものである。

図 6.4 にプログラム (1) の内容のフローチャートを示す。

## プログラム (1)

6 (並進・回転方向) × 2 (ずれの値の正負) = 12 個のデータファイルを用意

$f_{\pm}$  はデータ点の  $n\cos\theta_c$  の値をフィットし、Mean の値を採用し計算  
また、 $\Delta f_{\pm}$  は  $n\cos\theta_c$  の値をフィットし、Error の値を採用  
データの統計量が不十分な場合は  $n\cos\theta_c = 1.0$ 、 $f(t_p) = 0.0$ 、 $\Delta f(t_p) = 1.0$  とした

(1) データに Mupair 用のカットをかける

(2) MC によるずれのデータにおける 36 領域ごとのデータ点をフィット

(3) 上のフィットの際に得られた情報によりより  $f_{\pm}$ 、 $\Delta f_{\pm}$  の値を計算

(4) 取得された  $f_{\pm}$ 、 $\Delta f_{\pm}$  の値をテキストファイルに出力

(5) データ点のフィットをヒストグラムに出力

図 6.4 プログラム (1) のフローチャート



図 6.5 にプログラム (2) の内容のフローチャートを示す。

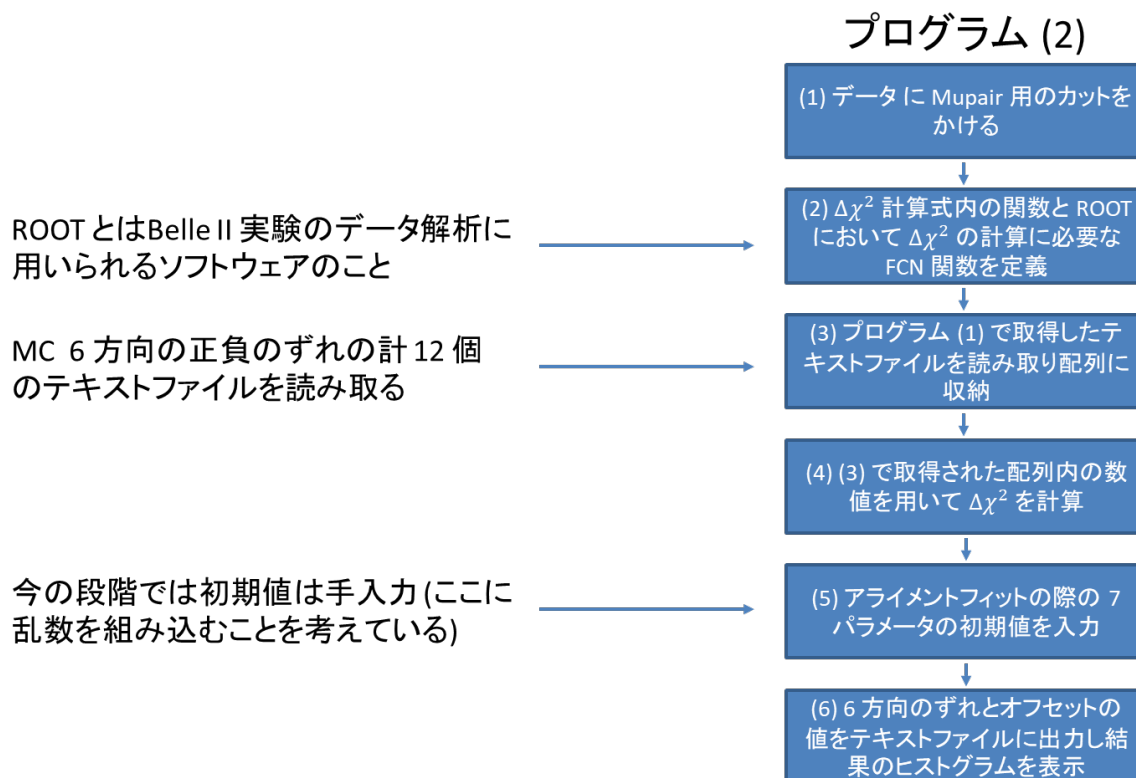


図 6.5 プログラム (2) のフローチャート

これらの新しいプログラム (1), (2) を用いて実際に Bucket 15 で採取された実データにおけるデータ点をフィットし、ARICH ずれの計算を行った。例として図 6.6 に Bucket 15 で取得された実データのフィットファイルのうち、領域 1 におけるデータ点 13 個  $d_{0,i}$  ( $i = 0, 1, \dots, 12$ ) のフィット結果についてを示す。図 6.6 の 14 個の図は、13 個の  $d_{0,0} - d_{0,12}$  のフィットのグラフと 13 個のデータ点を並べて表示するグラフを表示している。さらにデータ点を決定する際のフィット関数は先行研究の手法と同様にガウス関数と一次関数を足し合わせたものを用いている。

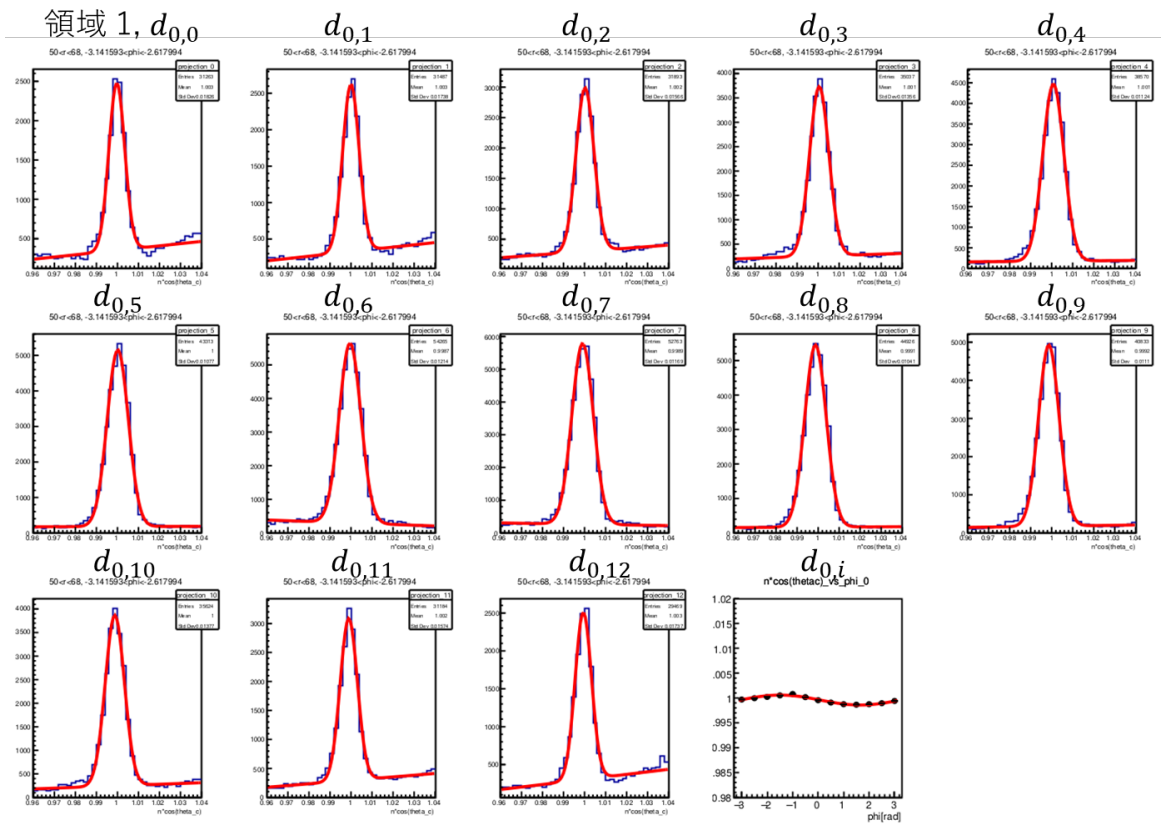


図 6.6 プログラム (1) を用いた実データの領域 1 におけるデータ点のフィット結果

図 6.7 に図 6.6 におけるデータ点 13 個のデータ点  $d_{0,i}$  を並べて表示したグラフについて、ARICH の 36 領域全てにおける実データのフィット結果を示す。

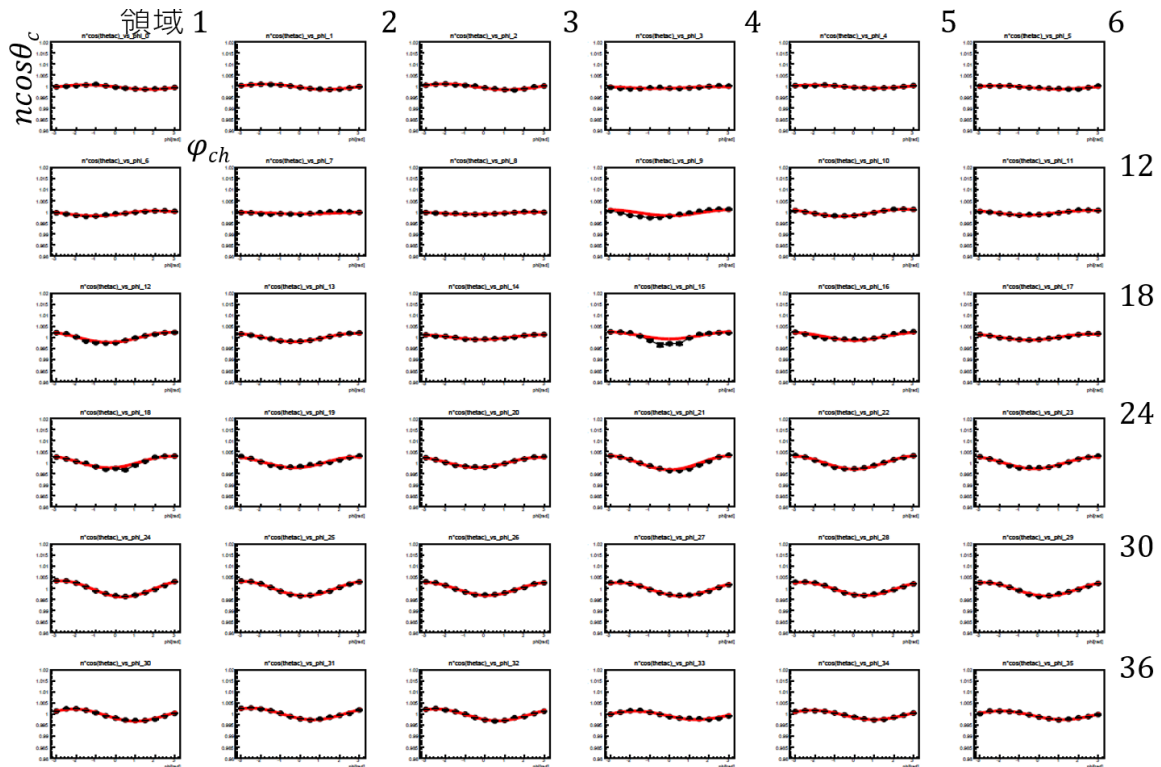


図 6.7 プログラム (1) を用いた実データのフィット結果

図 6.7 で得られた 36 領域ごとの ARICH のずれのフィット結果と、モンテカルロシミュレーションによる ARICH の正負の仮想的ずれの結果 (図 4.12, 4.13, 4.21, 4.22) を線形結合することによって ARICH のずれの値を 6 方向ごとに算出した。さらにそれらずれの値を元に ARICH のずれを再構成することによって結果に再現性があるかを確認した。図 6.8 に新しいフィット手法を用いた場合の ARICH 36 領域ごとのデータ点  $13 \times 36$  個と、それに対応した各領域ごとに再構成されたずれの比較図を示す。36 個のグラフにおける誤差付きの赤い点が算出されたずれの数値を元に再構成された ARICH のずれを表しており、誤差付きの黒いデータ点とこの赤い点が重なっていればいるほどずれの再構成がうまくいっていることを示しており、データ点と赤い点が重なっていなければずれの再構成ができていないことを示している。

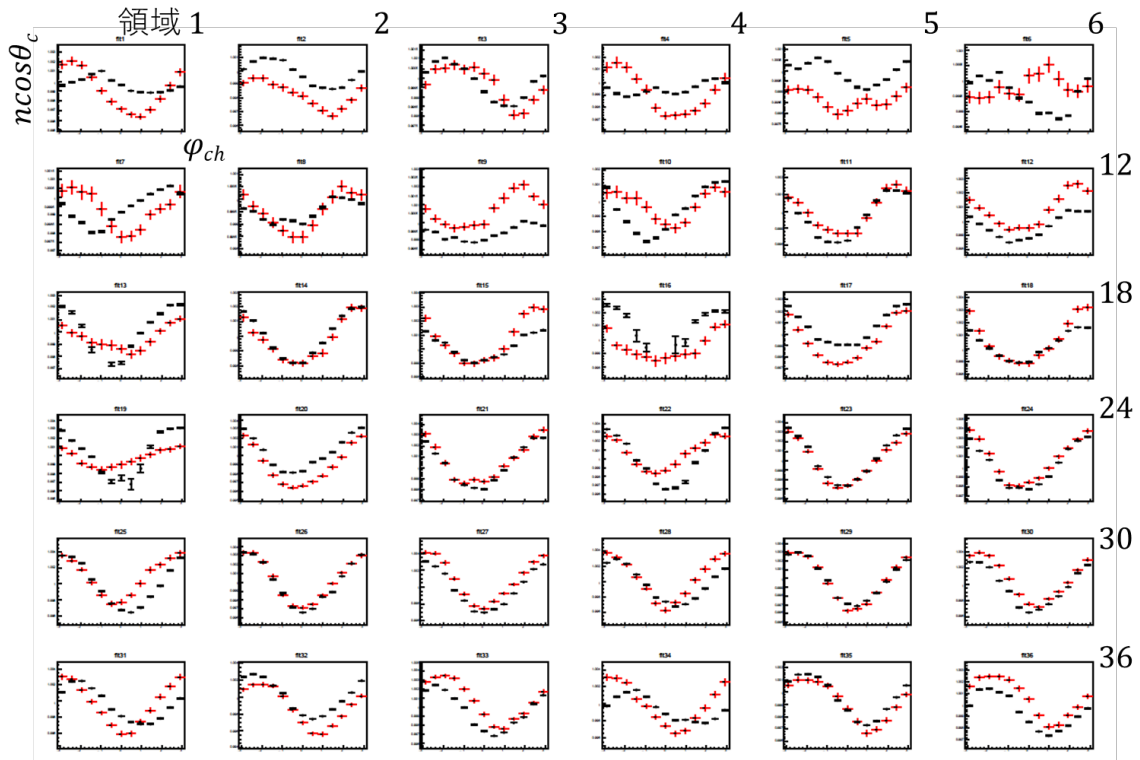


図 6.8 新しいフィット手法を用いた場合の ARICH 各領域におけるデータ点とずれの値から再構成されたずれの比較

節 5.1 で述べた際と同様に図 6.7 の新しいフィット結果と図 4.12, 4.13, 4.21, 4.22 の新しいシミュレーションによる正負の仮想的ずれのファイル 12 個の結果を利用し、ARICH の 6 方向のずれの値を計算した。表 6.1 にずれの計算結果を示す。

表 6.1 新たな手法を使った Bucket 15 を用いた場合の ARICH のずれの値

	並進移動 [mm]	回転移動 [°]
$x$ 軸	$-1.59 \pm 0.02$	$-0.035 \pm 0.004$
$y$ 軸	$-0.363 \pm 0.022$	$0.066 \pm 0.004$
$z$ 軸	$-0.188 \pm 0.028$	$0.052 \pm 0.001$

この際、 $\chi^2$  を自由度  $n = 36 \times 13 - 7$  で割った値は  $\chi^2/n = 21.2694$  であり、節 5.3 で述べた計算結果の  $\chi^2/n = 1364.26$  と比較して値の大きさが 1/70 程度にまで小さくなった。

## 6.2 得られたずれの値を用いた実データへのアライメントの結果

節 6.1 で算出された ARICH の 6 方向ごとのずれの値を実データに適用し、定量的にどの程度 ARICH のずれが改善されたか評価することを考える。選別条件として節 5.1 で述べたものと全く同じものをかけた状態の実データに表 6.1 の結果を適用させ ARICH のアライメントを試みた結果を図 6.9 に示す。

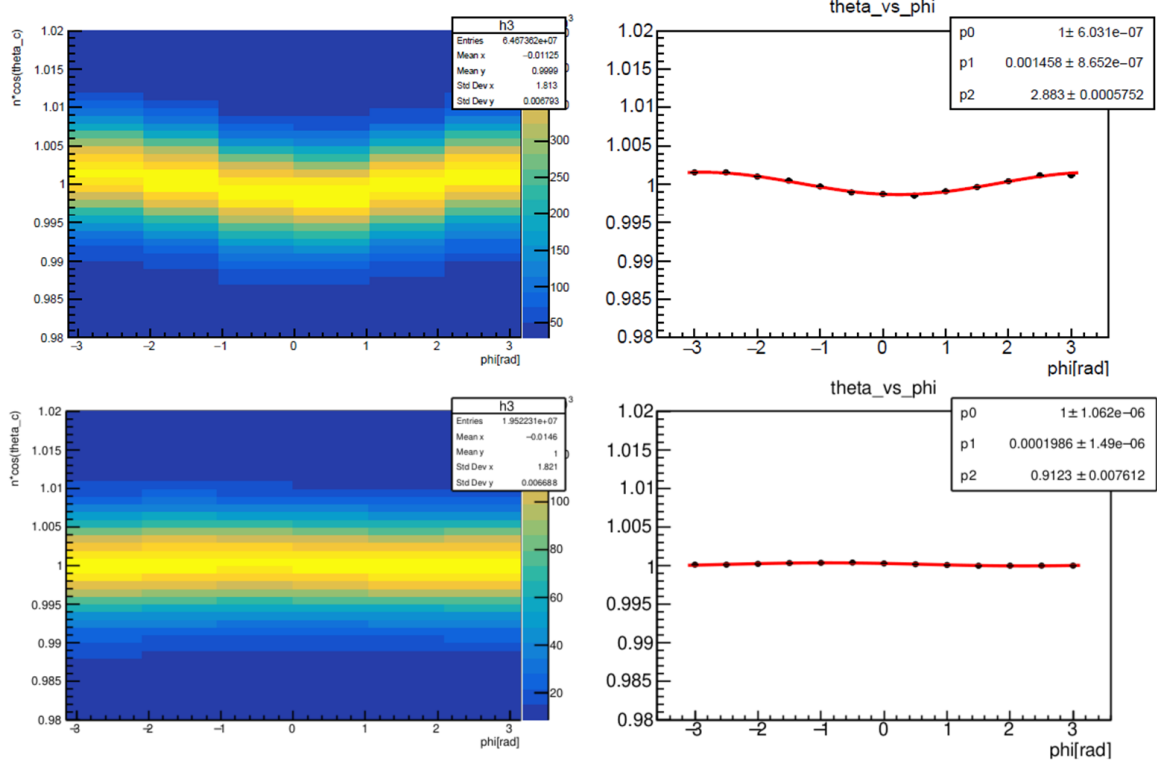


図 6.9 (上) アライメント結果適応前、(下) アライメント結果適応後の  $n \cos \theta_c$  の  $\phi_{ch}$  依存性

図 6.9 に示されているように、ARICH 全体における  $n \cos \theta_c$  の  $\phi_{ch}$  依存性を表示した二次元ヒストグラムを  $\cos$  関数でフィットした際の振幅 (図 6.9 における  $p_1$ ) の値はアライメント適応前後で 0.0014583 から 0.0001986 に改善した。

さらに、アライメント前後におけるチェレンコフ角度  $\theta_c$  の分布を図 6.10, 6.11 に示す。

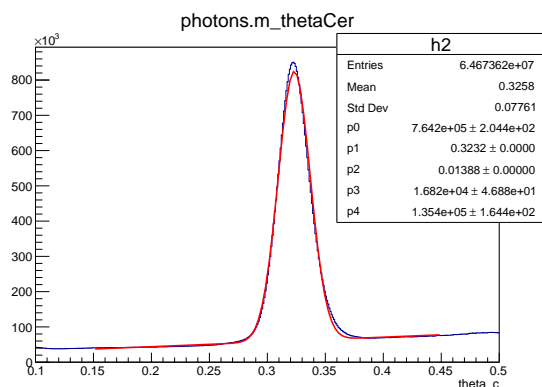


図 6.10 アライメント前の  $\theta_c$  分布

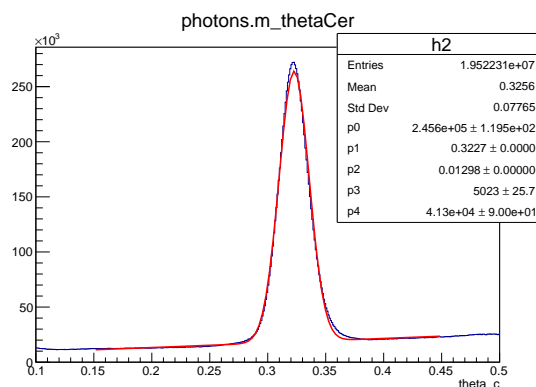


図 6.11 アライメント後の  $\theta_c$  分布

図 6.10, 6.11 の角度分布のヒストグラムをガウス関数と一次関数を足し合わせた関数でフィットした結果、ガウス関数の  $\sigma$  の値 (図 6.10, 6.11 における  $p_2$ ) はアライメント前後で 0.01388 rad から 0.01298 rad となり、約 6.5 % 改善した。

ここでチェレンコフ角度  $\theta_c$  が ARICH の  $K/\pi$  中間子の識別能力  $S$  に与える影響について考察する。 $S$  は  $K/\pi$  中間子それぞれによって測定されたチェレンコフ角度  $\theta_c$  の差  $\Delta\theta_c$ 、光子数  $N_{pe}$ 、チェレンコフ角度の角度分解能  $\sigma_\theta$  を用いて式 (6.5) に示す。

$$S = \frac{\Delta\theta_c}{\sigma_\theta} \sqrt{N_{pe}} \quad (6.5)$$

本研究において式 6.5 における  $\Delta\theta_c$ ,  $N_{pe}$  の値は ARICH のアライメントとは無関係でありアライメント前後で一定であるため、 $\sigma_\theta$  の値が 6.5 % 改善したという結果は  $S$  が 6.5 % 向上したことに対応する。

### 6.3 アライメントの結果の妥当性

本研究によって得られた ARICH のアライメントの結果の妥当性を検証する。表 6.2 に示されたアライメントフィットの結果の値を入力値として、プロセス  $e^+ + e^- \rightarrow \mu^+ + \mu^-$  をモンテカルロシミュレーションにより 85,617 イベント生成した。

表 6.2 入力値としたアライメントフィットの結果

	並進移動 [mm]	回転移動 [°]
x 軸	-1.59	-0.035
y 軸	-0.363	0.066
z 軸	-0.188	0.052

表 6.2 のパラメータを入力値としたモンテカルロシミュレーションにアライメントフィットを行ったところ、表 6.3 のような結果が得られた。

表 6.3 新たな手法にシミュレーションを用いた場合のアライメントフィットの結果

	並進移動 [mm]	回転移動 [°]
x 軸	$1.604 \pm 0.030$	$0.031 \pm 0.005$
y 軸	$0.274 \pm 0.031$	$-0.066 \pm 0.005$
z 軸	$0.381 \pm 0.043$	$-0.550 \pm 0.001$

また、表 6.3 ずれの値を元に ARICH のずれを再構成することによって結果に再現性があるかを確認した結果を図 6.12 に示す。図 6.8 と同様に、36 個のグラフにおける誤差付きの赤い点が算出されたずれの数値を元に再構成された ARICH のずれを表しており、誤差付きの黒いデータ点とこの赤い点が重なっていればいるほどずれの再構成がうまくいっていることを示しており、データ点と赤い点が重なっていない場合はずれの再構成ができていないことを示している。

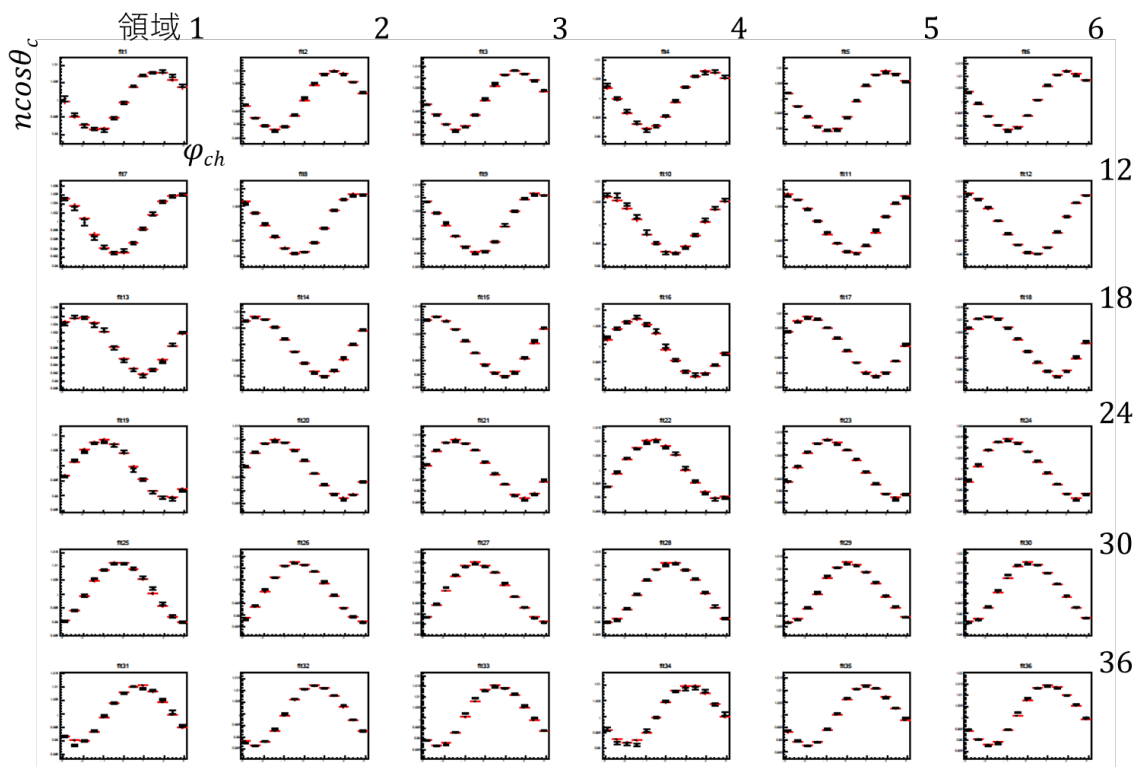


図 6.12 新たな手法によるシミュレーションで生成されたデータ点とずれの値から再構成されたずれの比較

表 6.2 と 6.3 の結果を比較すると、再構成されたずれの値の絶対値は回転方向 z 軸を除き並進方向に 0.1 mm 程度、回転方向に 0.01° 程度の精度で再現出来ているように見える一方で、それぞれの入力値が符号反転してアライメントフィットされる結果が得られた。さらに回転方向 z 軸のアライメントフィット結果に関しては、ずれの値が符号反転しなおかつ値が 10 倍程大きく出力された。考えられる符号反転の原因としては、節 4.4.1, 4.4.2 でモンテカルロシミュレーションを用いて各 12 方向に ARICH の仮想的なずれのデータを用意する際に誤って符号を反転して作成してしまった、または節 6.3 において表 6.2 の値を入力値としてモンテカルロシミュレーションの仮想的なずれのデータを

用意する際に誤って符号が反転するような操作をしてしまったこと等が挙げられる。

節 4.4.1, 4.4.2 で作成した ARICH の各 12 方向の仮想的ずれのデータを用いてアライメントフィットを行うことでずれの値の符号反転の有無を確認したところ、そのような符号反転現象は確認されなかった。従って、節 6.3 における表 6.2 の値を入力値としてモンテカルロシミュレーションの仮想的なずれのデータを作成した段階で何らかの誤った操作を行ってしまったことが一番有力な原因ではないかと考えられる。しかしこの符号反転現象がデータ作成の段階で人為的な誤った操作から来るものか、または使用したソフトウェアのバグなどその他の要因から来るものなのか、さらには再構成された回転方向  $z$  軸のずれが良い精度で出力されなかったのはなぜかなどの詳しい原因は不明である。



## 第 7 章

# 結論

Belle II 実験は茨城県つくば市にある高エネルギー加速器研究機構 (KEK) において、Super KEKB 加速器が作り出した電子・陽電子ビームを非対称エネルギーで衝突させ、その際衝突点に設置された Belle II 測定器を用いて生成される大量の  $B$  中間子対を観測し、それら粒子の崩壊過程における未知粒子の寄与を探ることによって新物理を探索を行う実験である。加速器のアップデートや各種測定器のグレードアップに伴いデータの統計量は前身の Belle 実験の 50 倍程度を目標としており、新たな物理発見の可能性がさらに期待されている。

Belle II 検出器は七つの異なる検出器から構成されるが、その中でも本研究を行うにあたり重要な検出器 Aerogel Ring Imaging Cherenkov counter (ARICH) は Belle II 測定器のエンドキャップ部に位置し、荷電  $K/\pi$  中間子の粒子識別を担っている。主に輻射体シリカエアロゲルと光検出器 Hybrid Avalanche Photo Detector (HAPD) で構成される ARICH は、荷電粒子がシリカエアロゲルを通過した際に発生するチェレンコフ光の放射角の値を用いて粒子識別を行っているため、新物理探索の際に  $B$  中間子の稀崩壊を詳細に調べる段階において崩壊の終状態によく含まれる  $K/\pi$  中間子の識別が非常に重要になってくる。そのため実験で得られた実データを利用し ARICH の設計上の位置からのずれをソフトウェアで算出し補正するアライメントという作業を行うことで、 $K/\pi$  中間子の識別能力の向上が見込める。

本研究ではまず、先行研究で行われたソフトウェアをそのまま使用し用いる実データをより最近取得された Phase III の Bucket 15 に変更してアライメントを試みた。ARICH のずれが位置依存性を有する可能性を考慮し 36 領域に分割し、各領域ごとにチェレンコフ光の放射角  $n \cos \theta_c$  の方位角  $\phi_{ch}$  に対する依存性を確認し、その後 36 個のグラフそれぞれを  $\cos$  関数でフィットした。さらにモンテカルロシミュレーションを用いて ARICH のモデルを使用することによりシミュレーション上での仮想的なずれがあった場合のデータファイルを 6 方向ごとに用意した。最後にこれらの結果を線形結合で表すことにより、ARICH のずれを算出しアライメントを行った。

しかし先行研究の手法で試みたアライメントでは満足の行く結果が得られず、さらに先行研究の手法にはいくつか問題点があったため、本研究では用いるシミュレーションファイルにおける素粒子反応を新たに  $e^+ + e^- \rightarrow \mu^+ + \mu^-$  とした。さらに新しいプログラムを作成することにより、従来の手法によるアライメントフィットの際に算出されていた  $\chi^2$  が大きくなってしまふ問題の改善を試みた。新しいプログラムを用いて  $\chi^2$  の値を計算した結果、先行研究の際と比較し値が 1/70 程度まで減少し

た。また、この結果 ARICH の分解能はアライメント前後で約 6.5 % 向上した。

今後はまずアライメントフィットの際のパラメータの初期値入力に乱数を用いてより適当な初期値を計算させることによりもう一段階フィット精度を高める、また ARICH ずれの表現に使用されていた六つのパラメータに新たなパラメータを導入させることによりさらに細かいアライメントを行っていくことが課題として挙げられる。

# 謝辞

本研究を進めるにあたり、大変多くの方々にご協力を頂きました。この二年間、とても私一人では成し遂げられなかった様々なことに関して、助けて頂いた全ての方々へここで深く感謝いたします。

まず所属研究室の角野秀一教授、汲田哲郎助教授、そして前年度前期まで特任教授でいらっしゃった住吉孝行先生に感謝を申し上げます。学部4年生でこの研究室に配属されてから3年の間で、私はお三方に沢山のことを学びました。日々の研究に関することはもちろんのこと、少人数制のゼミで面白い話を交えながら素粒子物理学や測定器に関する授業をして下さったことは非常に勉強になり、その後の研究生活に生かされるものになったと思います。そしてなによりも、今こうして執筆している修士論文が無事完成したのもお三方の熱心なアドバイスやフィードバックがあってこそのもので、本当にありがとうございました。

そして、研究をするにあたり日々生活を送っていた研究室のメンバーの皆様にも大変お世話になりました。先輩の在原拓司先輩はどんな時も優しく話を聞いて下さり、まだまだ勉強が足りない私にも分かりやすく説明をして下さいました。他人にものごとを分かりやすくしっかり伝える能力は誰しもが持っている能力ではないはずです。ですので私は在原先輩の圧倒的な知識量と共に、この伝える能力にまで長けているところを常々尊敬していました。さらに、同期の安藤晴也君、大島英里香さん、渡辺啓太君とは研究室で落ち合った際にはくだらない話をしたり、時には真剣に研究内容を議論しあったりととても楽しい2年間を過ごすことが出来ました。皆さんのお陰で研究が行き詰ってしまった時も乗り越えることが出来たといっても過言ではありません。後輩の北村君、鮫島君、澤君、竹淵君、そして2021年12月から新しく研究室に来て下さった博士研究員のThomasさんにも短い間でしたがお世話になりました。新型コロナウイルス流行の影響で研究室でイベント行事が出来なくなってしまうかもしれませんが、そんな状況でも感染対策を講じながらパーティーを開いてくれたり、備品の買出しに行ってくれた後輩の皆さんにはとても感謝しています。また、Thomasさんは私と同じ Belle II 実験に昔から携わっている大先輩として、私に様々なお話を下さったり助言を下さいました。Obrigado! 一生の思い出に残る研究室生活を皆で作ることが出来て本当に良かったです。

最後になりますが、私をここまで育ててくださった両親、そして共に成長し現在立派に働いている弟にもこの場をお借りして感謝を伝えたいと思います。これから社会人となる私ですが、自分を信じてこれからの人生が楽しいものとなるように一生懸命頑張りますので、どうかこれからも見守って下さい。そして私も家族の皆がこれから先も幸せに暮らせるように祈っています。

# 付録 A

プログラム (1) のソースコードを、ソースコード 7.1 に示す。

Listing 7.1 プログラム (1)

```
1 #include <TH2.h>
2 #include <fstream>
3 #include <TLegend.h>
4
5 void alignmentFit_park_editedbykakuno_4(const char* tag){
6     TFile *_file0 = new TFile(Form("root2/Bucket15Mcmupair_%s.root", tag));
7     //TFile *_file0 = new TFile("root2/Bucket15Exp12.root");
8     TTree* ch = (TTree*)_file0->Get("arich");
9
10    TCut mupair_cut = "5.<recHit.p&&recHit.p<8.1&&0.1<recHit.p*eop&&recHit.p*
11    eop<0.4&&e9e21>0.95&&-0.3<z0&&z0<0.3&&d0<0.1&&etot<1.0";
12
13    //cout << "Select region?" << endl;
14    //cin >> int R_n;
15
16    //Difine Canvas 6*6
17    TCanvas* c2=new TCanvas("c2");
18    c2->Update();
19    c2->Divide(6,6);
20
21    //Make histogram ARICH is subdivided into 24 regions.
22    TH2D* h4[36];
23    for(int t=0;t<12;t++){ //Divide 12 parts (-pi~pi) sector
24        for(int m=0;m<3;m++){ //Divide 3 parts (0~105)
25            int s=m+3*t; //0~35
26            c2->cd(s+1); //change divided canvas
27            h4[s]=new TH2D(Form("h4_%d",s),";phi[rad];n*cos(theta_c)", 26, -M_PI,
28                M_PI, 40, 0.96, 1.04); //Difine24 histograms
29            //Difine range
30            int rmin=50+m*18.5; //50~87
31            int rmax=50+(m+1)*18.5; //68.5~105.5
32            double phiMin=-TMath::Pi()+t*TMath::Pi()/6.0; //-pi^(2pi/3)
33            double phiMax=-TMath::Pi()+(t+1.0)*TMath::Pi()/6.0; //(-pi/2)~pi
34            //Fill histograms
```

```

33     cout << "t=" << t << ",m=" << m << endl;
34     ch->Draw(Form("photons.m_nCosThetaCh_mu:photons.m_phiCh_mu>>h4_%d",s),
        mupair_cut && Form("%d<sqrt(photons.m_x*photons.m_x+photons.m_y*
        photons.m_y)<&&sqrt(photons.m_x*photons.m_x+photons.m_y*photons.m_y)
        <%d<&&%f<atan2(photons.m_y,photons.m_x)<&&atan2(photons.m_y,photons
        .m_x)<%f",rmin, rmax, phiMin, phiMax));
35     cout << "a" << endl;
36     h4[s]->SetTitle(Form("%d<r<%d,<%f<phi<%f", rmin, rmax, phiMin, phiMax)
        );
37     h4[s]->Draw("colz");
38     h4[s]->SetStats(0);
39     }
40 }
41
42 // Fit 36 regions
43 TGraphErrors *g2[36];
44 TF1* fit1[36];
45 // TFile *g_file=new TFile(Form("simbha/Graph/g36_Bhabha_%s.root",file),"
    RECREATE");
46 // TFile *g_file=new TFile("g36_Bucket15MuPairMC_202108027.root","RECREATE")
    ;
47 // TFile *g_file=new TFile("
    g36_Bucket15Exp12MCmupair_alignmentFit_park_editedbykakuno_y-2mm_20211201
    .root","RECREATE");
48 // ofstream fout4("36x13_ncostheta_c_Bucket15Exp12_20211115.txt");
49 // ofstream fout5("36x13_error_Bucket15Exp12_20211111.txt");
50 // ofstream fout6("36x13_f_deltaf_Bucket15Exp12_20211111.txt");
51
52 ofstream fout6(Form("36x13_f_deltaf_20220111/36x13_f_deltaf%s.txt", tag));
53 // ofstream fout6("36x13_f_deltaf/36x13_f_deltafBucket15Exp12.txt");
54 TCanvas *c4=new TCanvas("c4");
55 c4->Divide(5,3);
56 for(int k=0;k<36;k++){
57     // c3->cd(k+1);
58     TH1D* projection[13];
59     Int_t first=0;
60     Int_t last=0;
61     TF1* fit[13];
62     Int_t N=13;
63     Float_t x[13],x_err[13];
64     Float_t y[13],y_err[13];
65     Float_t x_MAX;
66     Float_t y_MAX = -9999;
67     Float_t x_MIN;
68     Float_t y_MIN = 9999;
69     Int_t j=0;
70     for(Int_t i=0;i<N;i++){

```

```

71     c4->cd(i+1);
72     // first=h4[k]->GetXaxis()->FindBin(-3.0+i*0.5-0.1); // Dfine first bin
       0.92,0.92,...,1.05
73     // last=h4[k]->GetXaxis()->FindBin(-3.0+i*0.5+0.1); // Dfine last bin
       0.92,0.93,...,1.06
74     first=i*2+1;
75     last=(i+1)*2;
76     projection[i]=h4[k]->ProjectionY(Form("projection_%d",i),first,last);
       // Divide h2 into 13 (ProjectionY)
77
78     Float_t Max=projection[i]->GetMaximum();
79     Float_t Mean=projection[i]->GetMean();
80     Float_t Peak=projection[i]->GetBinCenter(projection[i]->GetMaximumBin
       ());
81     Float_t RMS=projection[i]->GetRMS();
82
83
84     Int_t Entries=projection[i]->Integral();
85     fit[i]=new TF1(Form("fit_%d",i),"gaus(0)+pol1(3)",0.95,1.05);
86     // fit[i]=new TF1(Form("fit_%d",i),"gaus(0)+pol1(3)",Mean-RMS*3.0,Mean+
       RMS*3.0);
87     fit[i]->SetParameter(0,Max);
88     fit[i]->SetParameter(1,Peak);
89     fit[i]->SetParameter(2,0.004);
90     fit[i]->SetParameter(3,0);
91     fit[i]->SetParameter(4,0);
92     // fit[i]->SetParameter(5,0); // pol2
93     projection[i]->Draw();
94     projection[i]->Fit(Form("fit_%d",i),"R");
95     c4->Update();
96     if(Entries>20){
97     // if(Entries>100 && fit[i]->GetParError(1)<0.003 && fit[i]->GetParameter
       (2)<0.01){
98         // x[j]=-3.0+i*0.5; // phi
99         x[j] = h4[k]->GetXaxis()->GetBinLowEdge(last);
100     x_err[j]=0; // phi err
101     y[j]=fit[i]->GetParameter(1); // Get n*cos(theta_c) Mean
102     y_err[j]=fit[i]->GetParError(1); // Get Mean Error
103
104     if(y[j]<y_MIN){
105         y_MIN = y[j];
106         x_MIN = x[j];
107     }
108     if(y[j]>y_MAX){
109         y_MAX = y[j];
110         x_MAX = x[j];
111     }

```

```

112 j++;
113 // Save 36x13 ncostheta_c and error
114 //fout4 << k << " " << i << " " << fit[i]->GetParameter(1) << " " << fit
    [i]->GetParError(1) << endl;
115 //fout4 << "In region " << k+1 << ", ncostheta_c of data point No." << i
    +1 << " -> " << fit[i]->GetParameter(1) << ", " << "f_m = " << (fit[i]
    ]->GetParameter(1)-1.0) << endl;
116 //fout5 << "In region " << k+1 << ", error of data point No." << i+1 <<
    " -> " << fit[i]->GetParError(1) << ", " << "deltaf_m = " << fit[i]->
    GetParError(1) << endl;
117 fout6 << k << "□" << i << "□" << (fit[i]->GetParameter(1)-1.0) << "□" <<
    fit[i]->GetParError(1) << endl;
118 } else {
119 // if the statistics is insufficient, return ncostheta_c=1.0, f=0.0 and
    delta_f=1.0
120 //fout4 << k << " " << i << " " << 0.0 << " " << 1.0 << endl;
121 //fout4 << "In region " << k+1 << ", ncostheta_c of data point No." << i
    +1 << " -> " << 1.0 << ", " << "f_m = " << 0.0 << endl;
122 //fout5 << "In region " << k+1 << ", error of data point No." << i+1 <<
    " -> " << 1.0 << ", " << "deltaf_m = " << 1.0 << endl;
123 fout6 << k << "□" << i << "□" << 0.0 << "□" << 1.0 << endl;
124 }
125 }
126 c4->cd(N+1);
127 // c4[k]->Print(Form("simbha/PNG/%s-%d.png",file,k+1));
128 // c3->cd(k+1); // with c4
129
130 g2[k]=new TGraphErrors(j,x,y,x_err,y_err); // phi vs n*cos(thetac) graph
131 g2[k]->SetTitle(Form("n*cos(thetac)_vs_phi_%d",k));
132 g2[k]->GetXaxis()->SetTitle("phi[rad]");
133 g2[k]->GetYaxis()->SetTitle("n*cos(thetac)");
134 g2[k]->GetXaxis()->SetTitleSize(0.05);
135 g2[k]->GetYaxis()->SetTitleSize(0.05);
136 g2[k]->GetXaxis()->SetLabelSize(0.05);
137 g2[k]->GetYaxis()->SetLabelSize(0.05);
138 g2[k]->SetMarkerStyle(7);
139 g2[k]->SetMaximum(1.02);
140 g2[k]->SetMinimum(0.98);
141 g2[k]->GetXaxis()->SetLimits(-3.3,3.3);
142 g2[k]->Draw("AP");
143 g2[k]->Write();
144 fit1[k]=new TF1(Form("fit1_%d",k+1),"[0]+[1]*cos(x+[2])",-TMath::Pi(),
    TMath::Pi());
145 fit1[k]->SetParameter(0,(y_MIN+y_MAX)/2.);
146 fit1[k]->SetParLimits(0,y_MIN,y_MAX);
147 //fit1[k]->SetParLimits(0,0.0,10.0);
148 //fit1[k]->SetParameter(1,0.003); // z_tra, z_rot

```

```

149 fit1[k]->SetParameter(1, ((y_MAX)-(y_MIN))/2.); // x_tra, y_tra
150 fit1[k]->SetParLimits(1, (y_MAX-y_MIN)/8., (y_MAX-y_MIN));
151 // fit1[k]->SetParLimits(1, 0.0005, 0.1); // x_tra, z_tra, x_rot
152 // fit1[k]->SetParLimits(1, 0.0001, 0.01); // y_tra
153 // fit1[k]->SetParLimits(1, 0.002, 0.1); // y_rot
154 // fit1[k]->SetParameter(2,0); // x_tra, z_tra, y_rot, z_rot
155 fit1[k]->SetParameter(2, -x_MAX); // y_tra
156 // fit1[k]->SetParameter(2, TMath::Pi()/2); // x_rot
157 // fit1[k]->SetParLimits(2, -TMath::Pi(), TMath::Pi()); // x_tra, z_rot,
    z_tra
158 fit1[k]->SetParLimits(2, -TMath::Pi(), TMath::Pi());
159 // fit1[k]->SetParLimits(2, -TMath::Pi()*2/3, -1/3*TMath::Pi()); // y_tra
160 // fit1[k]->SetParLimits(2, TMath::Pi()/3, 2/3*TMath::Pi()); // x_rot
161 // fit1[k]->SetParLimits(2, -TMath::Pi()/2, TMath::Pi()/2); // y_rot
162 gStyle->SetStatH(0.2);
163 gStyle->SetStatW(0.5);
164 g2[k]->Fit(Form("fit1_%d",k+1),"R");
165 cout << "x_MAX_=" << x_MAX << endl;
166 cout << "y_MAX_=" << y_MAX << endl;
167 cout << "x_MIN_=" << x_MIN << endl;
168 cout << "y_MIN_=" << y_MIN << endl;
169 c4->SaveAs(Form("region%02d.pdf",k));
170 }
171 gSystem->Exec(Form("pdfunite_region?.pdf_%s.pdf", tag));
172 // gSystem->Exec("pdfunite_region?.pdf Bucket15Exp12.pdf");
173 gSystem->Exec("rm_region?.pdf");
174 // g_file ->Close();
175
176 // Save 36 Offsets (p0)
177 ofstream fout3(Form("simmu_extraevents/Offset/Offset_Mupair_%s.txt",tag));
178 // ofstream fout3(Form("simbha/Offset/Offset_Bhabha_%s.txt",file));
179 // ofstream fout3("Offset_Bucket15MuPairMC_202108027.txt");
180 // ofstream fout3("Offset_Bucket15Exp12MCmupair_c05deg_20211119.txt");
181
182 for(int j=0;j<36;j++){
183     fout3<<fit1[j]->GetParameter(0)<<","<<fit1[j]->GetParError(0)<<","<<endl;
184     ;
185 }
186
187 // Save 36 Amplitudes (p1)
188 ofstream fout(Form("simmu_extraevents/Amplitude/Amp_Mupair_%s.txt",tag));
189 // ofstream fout(Form("simbha/Amplitude/Amp_Bhabha_%s.txt",file));
190 // ofstream fout("Amplitude_Bucket15MuPairMC_202108027.txt");
191 // ofstream fout("Amplitude_Bucket15Exp12MCmupair_c05deg_20211119.txt");
192 for(int j=0;j<36;j++){
193     fout<<fit1[j]->GetParameter(1)<<","<<fit1[j]->GetParError(1)<<","<<endl;
194     }

```



```

194
195 //Save 36 phase (p2)
196 ofstream fout2 (Form("simmu_extraevents/Phase/Phase_Mupair_%.txt",tag));
197 //ofstream fout2 (Form("simbha/Phase/Phase_Bhabha_%.txt",file));
198 //ofstream fout2 ("Phase_Bucket15MuPairMC_202108027.txt");
199 //ofstream fout2 ("Phase_Bucket15Exp12Mcmupair_c05deg_20211119.txt");
200 for (int j=0;j<36;j++){
201     fout2 <<fit1 [j]->GetParameter (2)<<","<<fit1 [j]->GetParError (2)<<","<<endl
202         ;
203
204 }
205 //cout << "ncostheta_c of data point No." << i+1 << "->" << y[j] << endl
206     ;
207 c2->SaveAs (Form ("%s_2.pdf",tag));
208 //c2->SaveAs ("Bucket15Exp12_2.pdf");
209
210 TCanvas*c3=new TCanvas ("c3");
211 c3->Update ();
212 c3->Divide (6,6);
213 for (int k=0;k<36;k++){
214     c3->cd (k+1);
215     g2 [k]->Draw ();
216 }
217 c3->SaveAs (Form ("%s_3.pdf",tag));
218 //c3->SaveAs ("Bucket15Exp12_3.pdf");
219 }

```

# 付録 B

プログラム (2) のソースコードを、ソースコード 7.2 に示す。

Listing 7.2 プログラム (2)

```
1 #include <iostream >
2 #include <TH2.h>
3 #include <fstream >
4 #include <TLegend.h>
5 #include <stdlib.h>
6 #include <cmath>
7 TGraphErrors *g[36];
8 TH1 *h[36];
9
10 Double_t f_plus [6][36][13];
11 Double_t f_minus [6][36][13];
12 Double_t f_0 [36][13];
13 Double_t delta_f_plus [6][36][13];
14 Double_t delta_f_minus [6][36][13];
15 Double_t d_i [36][13];
16 Double_t de_i [36][13];
17
18 Double_t offset [6][36];
19 Double_t offset_err [6][36];
20 Double_t amp[6][36];
21 Double_t phase [6][36];
22 Double_t amp_err [6][36];
23 Double_t phase_err [6][36];
24 // Double_t data1 [36][13];
25 // Double_t data_err [36][13];
26 Double_t malt[6]={ 5.0, 5.0, 5.0, 1.0, 1.0, 0.5};
27
28 Double_t get_ft(const Int_t parameter, const Int_t region, const Int_t bin,
29               const Double_t t){
30     // Translation(0<=parameter<=1): +-2mm -> denom=4.0
31     // Translation(paeameter=2): +-5mm -> denom=10.0
32     // Rotation (parameter >=3): +-0.5deg -> denom=1.0
33     Double_t denom = ((parameter >=3) ? 1.0 : (((0 <= parameter) && (parameter
34     <=1)) ? 4.0 : 10.0));
```

```

33  if (parameter>=6 || region>=36 || bin>=13) return -100.;
34  Double_t grad=(f_plus [parameter][region][bin]-f_minus [parameter][region][
    bin])/denom;
35  Double_t offset=(f_plus [parameter][region][bin]+f_minus [parameter][region
    ][bin])/2.0;
36  return grad*t + offset;
37
38  //Double_t first_term = (std::pow((1/2 + t/denom), 2.0))*(std::pow(
    delta_f_plus [parameter][region][bin], 2.0));
39  //Double_t second_term = (std::pow((1/2 + t/denom), 2.0))*(std::pow(
    delta_f_minus [parameter][region][bin], 2.0));
40  //return sqrt(first_term + second_term);
41
42 }
43
44 Double_t get_dft2(const Int_t parameter, const Int_t region, const Int_t bin
    , Double_t t){
45  // Translation(0<=parameter<=1): +-2mm -> denom=4.0
46  // Translation(paeameter=2): +-5mm -> denom=10.0
47  // Rotation(parameter>=3): +-0.5deg -> denom=1.0
48  Double_t denom = ((parameter>=3) ? 1.0 : (((0<=parameter) && (parameter
    <=1)) ? 4.0 : 10.0));
49  if (parameter>=6 || region>=36 || bin>=13) return 100.;
50  Double_t first_term = (std::pow((1/2. + t/denom), 2.0))*(std::pow(
    delta_f_plus [parameter][region][bin], 2.0));
51  Double_t second_term = (std::pow((1/2. - t/denom), 2.0))*(std::pow(
    delta_f_minus [parameter][region][bin], 2.0));
52  //return sqrt(first_term + second_term);
53  // cout << "a" << parameter << " " << region << " " << bin << " " <<
    first_term << " " << second_term << " " << delta_f_plus [parameter][region
    ][bin] << " " << delta_f_minus [parameter][region][bin] << endl;
54  //cout << "a " << parameter << " " << region << " " << bin << " " << std::
    pow((1/2. - t/denom), 2.0) << " " << std::pow(delta_f_minus [parameter][
    region][bin], 2.0) << " " << delta_f_minus [parameter][region][bin] << " "
    << second_term << endl;
55  return first_term + second_term;
56 }
57
58 Double_t fit_val(const Int_t region, const Int_t bin, Double_t* t){
59  Double_t val = 0;
60  for(Int_t p=0; p<6; p++){
61  val += get_ft(p, region, bin, t[p]);
62  }
63  val += t[6];
64  // subtract 6*offset from fit function and add 1*offset ==> subtract (6-1)*
    offset
65  val -= (6-1)*f_0 [region][bin];

```

```

66     return val;
67 }
68
69 Double_t fit_val_error2(const Int_t region, const Int_t bin, Double_t* t){
70     Double_t error2 = 0;
71     for(Int_t p=0; p<6; p++){
72         error2 += get_dft2(p, region, bin, t[p]);
73     }
74     return error2;
75 }
76
77 static void FCN(int &npar, Double_t *gin, Double_t &f, Double_t *t, int
78     iflag){
79     f = 0;
80     // if (npar != 7) return;
81     for(Int_t k=0; k<36; k++){
82         for(Int_t i=0; i<13; i++){
83             Double_t residual = d_i[k][i] - fit_val(k, i, t);
84             Double_t error2 = de_i[k][i]*de_i[k][i] + fit_val_error2(k, i, t);
85             f += (error2 > 0) ? residual*residual/error2 : 0; // calculation of
86                 chi2
87             // cout << k << " " << i << " " << " " << d_i[k][i] << "-" << fit_val(k
88                 , i, t) << "=" << residual << "/" << sqrt(error2) << " ^2=" <<
89                 residual*residual/error2 << " " << f << endl;
90         }
91     }
92     // cout << "AAAA" << npar << " " << t[0] << " " << t[1] << " " << t[2] << "
93         " << t[3] << " " << t[4] << " " << t[5] << " " << t[6] << " " << f <<
94         std::endl;
95 }
96
97 /*
98 static void FCN(int &npar, Double_t *gin, Double_t &f, Double_t *t, int
99     iflag){
100     f = 0;
101     // if (npar != 7) return;
102     for(Int_t k=0; k<36; k++){
103         for(Int_t i=0; i<13; i++){
104             Double_t residual = d_i[k][i];
105             Double_t error2 = de_i[k][i]*de_i[k][i];
106             for(Int_t p=0; p<6; p++){
107                 residual -= get_ft(p, k, i, t[p]);
108                 error2 += get_dft2(p, k, i, t[p]);
109                 // cout << k << " " << i << " " << p << " " << get_dft2(p, k, i, t[p]) <<
110                     endl;
111             }
112             residual -= t[6];

```

```

105     // subtract 6*offset from fit function and data and 1*offset from data
        ==> add (6-1)*offset
106     residual += (6-1)*f_0[k][i];
107     f += (error2 > 0) ? residual*residual/error2 : 0; // calculation of
        chi2
108     }
109 }
110 cout << "AAAA" << npar << " " << t[0] << " " << t[1] << " " << t[2] << " "
        << t[3] << " " << t[4] << " " << t[5] << " " << t[6] << " " << f << std
        ::endl;
111 }
112 */
113 void alignmentFit_park5_kakuno_edited_2(){
114
115     // Initialization: value=0.0, error=1.0
116     for(Int_t par=0; par<6; par++){
117         for(Int_t k=0; k<36; k++){
118             for(Int_t i=0; i<13; i++){
119                 f_plus[par][k][i] = 0.0;
120                 f_minus[par][k][i] = 0.0;
121                 delta_f_plus[par][k][i] = 1.0;
122                 delta_f_minus[par][k][i] = 1.0;
123                 d_i[k][i] = 0.0;
124                 de_i[k][i] = 1.0;
125             }
126         }
127     }
128
129     // Read values of f_p, f_m, and deltaf_p, deltaf_m from data files
130     for(Int_t par=0; par<6; par++){
131         ifstream fa;
132         ifstream fb;
133         Int_t k, i;
134         switch( par ){
135             case 0:
136                 fa.open("36x13_f_deltaf_20220107/36
                        x13_f_deltafx2mm_20211213_extraevents.txt");
137                 fb.open("36x13_f_deltaf_20220107/36x13_f_deltafx-2
                        mm_20211213_extraevents.txt");
138                 break;
139             case 1:
140                 fa.open("36x13_f_deltaf_20220107/36
                        x13_f_deltafy2mm_20211215_extraevents.txt");
141                 fb.open("36x13_f_deltaf_20220107/36x13_f_deltafy-2
                        mm_20211219_extraevents.txt");
142                 break;
143             case 2:

```

```

144     fa.open("36x13_f_deltaf_20220107/36
        x13_f_deltafz5mm_20211219_extraevents.txt");
145     fb.open("36x13_f_deltaf_20220107/36x13_f_deltafz-5
        mm_20211222_extraevents.txt");
146     break;
147 case 3:
148     fa.open("36x13_f_deltaf_20220107/36
        x13_f_deltafa05deg_20211225_extraevents.txt");
149     fb.open("36x13_f_deltaf_20220107/36x13_f_deltafa-05
        deg_20211227_extraevents.txt");
150     break;
151 case 4:
152     fa.open("36x13_f_deltaf_20220107/36
        x13_f_deltafb05deg_20220101_extraevents.txt");
153     fb.open("36x13_f_deltaf_20220107/36x13_f_deltafb-05
        deg_20220101_extraevents.txt");
154     break;
155 case 5:
156     fa.open("36x13_f_deltaf_20220107/36
        x13_f_deltafc05deg_20220101_extraevents.txt");
157     fb.open("36x13_f_deltaf_20220107/36x13_f_deltafc-05
        deg_20220103_extraevents.txt");
158     break;
159 }
160 while (fa && !fa.eof()){
161     fa >> k >> i;
162     fa >> f_plus[par][k][i] >> delta_f_plus[par][k][i];
163     if (fa.eof()) break;
164 }
165 while (fb && !fb.eof()){
166     fb >> k >> i;
167     fb >> f_minus[par][k][i] >> delta_f_minus[par][k][i];
168     // std::cout << "par=" << par << ",k=" << k << ",i=" << i << ", f_minus
        " << f_minus[par][k][i] << ", delta_f_minus=" << delta_f_minus[par][k
        ][i] << std::endl;
169     if (fb.eof()) break;
170 }
171 }
172
173 // f_0: offset from ncostheta=1 at x=y=z=a=b=c=0
174 for (Int_t k = 0; k < 36; k++){
175     for (Int_t i = 0; i < 13; i++){
176         f_0[k][i] = 0.;
177         Int_t npar=0;
178         for (Int_t p = 0; p < 6; p++){
179             f_0[k][i] += get_ft(p, k, i, 0);
180             npar++;

```

```

181     }
182     f_0[k][i]/=(Float_t)npar;
183     }
184 }
185 // Read values of d_i from the data file
186 ifstream fc;
187 Int_t k, i;
188 fc.open("36x13_ncostheta_c_Bucket15Exp12_20220107.txt");
189 //fc.open("36x13_f_deltafb2e-1deg_20210930.txt");
190 while (fc && !fc.eof()){
191     fc >> k >> i;
192     fc >> d_i[k][i] >> de_i[k][i];
193     //if (d_i[k][i]<0.5) d_i[k][i]+=1.0;
194     //std::cout << "k= " << k << ",i=" << i << ", d_i = " << d_i[k][i] << ",
195         de_i = " << de_i[k][i] << std::endl;
196     if (fc.eof()) break;
197 }
198 // Make TGraphError objects for plots of data points (H.Kakuno)
199 Double_t y[13], ey[13];
200 for (Int_t i = 0; i < 13; i++){
201     y[i] = -M_PI + (i + 0.5)*2*M_PI/13.;
202     ey[i] = 0;
203 }
204 for (Int_t k = 0; k < 36; k++){ // loop over graphs
205     g[k] = new TGraphErrors(13, y, d_i[k], ey, de_i[k]);
206 }
207 //---- for test -----
208 /*
209 Double_t ft_i, delta_fti;
210
211 cout << "Enter the region from 0 to 35: ";
212 cin >> k;
213 cout << "Enter the bin from 0 to 12: ";
214 cin >> i;
215
216 cout << "In Bucket15, Exp12, region " << k << ", bin " << i << ", d_i = "
217     << d_i[k][i] << ", de_i = " << de_i[k][i] << endl;
218
219 example: x=1.0mm, y=0.2mm, z=-0.3mm, alpha=0.01deg, beta=0.02deg, gamma
220         =0.03deg, offset=0.99
221 Double_t t[7] = {-4.10737, -1.13175, 1.168, -0.519408, -0.453586,
222                 -0.385159, 0.999847};
223 Int_t npar=7;
224 Double_t chi_square;
225 FCN(npar, NULL, chi_square, t, 0);
226 cout << "chi square = " << chi_square << endl; */

```

```

224
225 Int_t ierflg = 0;
226 TMinuit minuit(7);
227 minuit.SetFCN(FCN);
228
229 minuit.mnparm(0, "trans_x", -1.55879, 0.01, -4.0, 4.0, ierflg); // Unit:
    mm
230 minuit.mnparm(1, "trans_y", -0.435761, 0.01, -4.0, 4.0, ierflg); // Unit:
    mm
231 minuit.mnparm(2, "trans_z", -0.194298, 0.01, -4.0, 4.0, ierflg); // Unit:
    mm
232 minuit.mnparm(3, "rot_x", -0.0464982, 0.01, -2.0, 2.0, ierflg); // Unit:
    deg
233 minuit.mnparm(4, "rot_y", 0.0617713, 0.01, -2.0, 2.0, ierflg); // Unit:
    deg
234 minuit.mnparm(5, "rot_z", 0.0833053, 0.01, -2.0, 2.0, ierflg); // Unit:
    deg
235 minuit.mnparm(6, "offset", 1.0, 0.00001, 0.99, 1.01, ierflg);
236 // minuit.FixParameter(0);
237 // minuit.FixParameter(1);
238 // minuit.FixParameter(2);
239 // minuit.FixParameter(3);
240 // minuit.FixParameter(4);
241 // minuit.FixParameter(5);
242 // minuit.FixParameter(6);
243 minuit.Migrad();
244 minuit.mnhess();
245 minuit.mnmnos();
246
247 double_t val[7], err[7];
248 for (Int_t p = 0; p < 7; p++){
249     minuit.GetParameter(p, val[p], err[p]);
250 }
251
252 // ofstream fout("fitresult.txt");
253 ofstream fout("
    fitresult_Bucket15Expl2_alignmentFit_park5_kakuno_edited_2_20220107_2.txt
    ");
254 Double_t chi2_ndf;
255 Double_t ln0, edm, errdef;
256 Int_t nvar, nparx, icstat;
257 for (Int_t p=0;p<7;p++){
258     fout<<val[p]<<" "<<err[p]<<" "<<endl;
259 }
260 minuit.mnstat(ln0, edm, errdef, nvar, nparx, icstat);
261 chi2_ndf=ln0/(36*13-7);
262 cout << "chi2/ndf=_" << chi2_ndf << endl;

```



```

263 fout << "chi2/ndf=_" << chi2_ndf << endl;
264
265 TCanvas*c1=new TCanvas("c1");
266 c1->Divide(6,6);
267 Float_t val_plot = 0;
268 Int_t id_plot=5;
269 for (Int_t k = 0; k < 36; k++){ // loop over graphs
270     c1->cd(k+1);
271     // Make histograms for fit curve (H.Kakuno)
272     char hname[256];
273     sprintf(hname, "fit%d", k);
274     h[k] = new TH1D(Form("fit%d",k+1), Form("fit%d",k+1), 13, -M_PI, M_PI);
275     h[k]->SetStats(0);
276     Double_t max = 0.9;
277     Double_t min = 1.1;
278
279     for (Int_t i = 0; i < 13; i++){
280         Double_t y = fit_val(k, i, val);
281         Double_t ey = sqrt(fit_val_error2(k, i, val));
282         h[k]->SetBinContent(i+1, y);
283         h[k]->SetBinError(i+1, ey);
284         //h[k]->SetBinContent(i+1, get_ft(id_plot, k, i, val_plot) + val[6]);
285         //h[k]->SetBinError(i+1, sqrt(get_dft2(id_plot, k, i, val_plot)));
286         if (y<1.1 && y+ey > max) max = y+ey;
287         if (y>0.9 && y-ey < min) min = y-ey;
288
289         y=g[k]->GetY()[i];
290         ey=g[k]->GetErrorY(i);
291         if (y<1.1 && y+ey > max) max = y+ey;
292         if (y>0.9 && y-ey < min) min = y-ey;
293     }
294
295     h[k]->SetLineColor(2);
296     h[k]->GetYaxis()->SetRangeUser(min-0.2*(max-min), max+0.2*(max-min));
297     // Write fit curve (histogram)
298     h[k]->Draw();
299     //h[k]->Draw("hist");
300     // Write data points
301     g[k]->SetMarkerStyle(6);
302     g[k]->Draw("P_same");
303 }
304 c1->SaveAs("Bucket15Exp12alignmentFit_park5_kakuno_edited_2_hist_20220114.
pdf");
305 //const char *type = "xyzabc";
306 //c1->SaveAs(Form("36x13_ft_%c%g%s.pdf",type[id_plot], val_plot, id_plot >
2 ? "deg" : "mm"));
307 }

```



# 参考文献

- [1] KEK, 標準理論を超えるためには, <https://www.kek.jp/ja/newsroom/2012/08/31/1800/>.
- [2] J.H. Christenson, J.W. Cronin, V.L. Fitch and R. Turlay. “Evidence for the  $2\pi$  Decay of the  $K_2^0$  Meson”. *Physical Review Letters* 13, 138, pp. 138–140, Jul. 1964.
- [3] Makoto Kobayashi, Toshihide Maskawa. “CP-Violation in the Renormalizable Theory of Weak Interaction”. *Progress of Theoretical Physics* 49, No. 2, pp. 652–657, Feb. 1973.
- [4] HFLAV, “Time-dependent CP violation & Unitarity Triangle angles”. Summer 2018. [https://www.slac.stanford.edu/xorg/hflav/triangle/summer2018/plots/btoocsS\\_CP.png](https://www.slac.stanford.edu/xorg/hflav/triangle/summer2018/plots/btoocsS_CP.png)
- [5] CKMfilter, “The global CKM fit in the small ( $\rho$ -bar, $\eta$ -bar) plane (zoom)”. Preliminary results as of Summer 2019 (EPS 2019 conference). [http://ckmfitter.in2p3.fr/www/results/plots\\_summer19/png/belle\\_rhoeta\\_small\\_global.png](http://ckmfitter.in2p3.fr/www/results/plots_summer19/png/belle_rhoeta_small_global.png)
- [6] Lincoln Wolfenstein. “Parametrization of the Kobayashi-Maskawa Matrix”. *Physical Review Letters* 51, 1945, Nov. 1983.
- [7] 上原貞治. “Belle で見つかった新粒子 X(3872) と Y(3940)”, 2005 年 12 月, 高エネルギー物理学研究者会議, 高エネルギーニュース, Vol. 24, No. 3.
- [8] E. Kou et al. (Belle II Collaboration). “The Belle II Physics Book”. BELLE2-PUB-PH-2018-001, Aug. 2018.
- [9] KEK. “世界の電子陽電子衝突型加速器とその瞬間ルミノシティのグラフ”. KEK 公式ウェブページ, イメージアーカイブ, 2018 年 3 月. <https://www.kek.jp/ja/imagearchive/images/20180319-luminosity.png>.
- [10] KEK. “SuperKEKB 加速器全体の模式図”. KEK 公式ウェブページ, イメージアーカイブ, 2018 年 3 月. [https://www.kek.jp/ja/imagearchive/images/20180320\\_superkekb\\_001.png](https://www.kek.jp/ja/imagearchive/images/20180320_superkekb_001.png).
- [11] Rey.Hori / KEK. “Overview of the Belle II detector”. KEK 公式ウェブページ, イメージアーカイブ, 2018 年 4 月. [https://www.kek.jp/ja/imagearchive/images/20180425\\_belle2\\_002.png](https://www.kek.jp/ja/imagearchive/images/20180425_belle2_002.png).
- [12] I. Adachi et al. (Belle II Collaboration). “Detectors for extreme luminosity: Belle II”. *Nuclear Instruments and Methods in Physics Research A*, Vol. 907, pp. 46–59, Nov. 2018.
- [13] T. Abe et al. (Belle II Collaboration). “Belle II Technical Design Report”. KEK Report 2010-1, Nov. 2010.
- [14] 爲近彩智 “Belle II 実験 Phase II ランにおける ARICH 検出器のアライメントおよび  $B \rightarrow K^*\gamma$  崩壊の探索”. 修士論文. 首都大学東京大学院 理工学研究科, 2018 年.