

2018 前期 A2 最終レポート オルソポジトロニウムの寿命測定

井上大輔 久代翔大 杉浦駿
福田篤人 古田悠馬 細谷享平

2018/11/12

概要

ポジトロニウム (Ps) は電子と陽電子の束縛状態であり、構成要素の電子、陽電子のスピン向きによって、パラポジトロニウム (p-Ps) とオルソポジトロニウム (o-Ps) に分けられる。後者の寿命の測定は量子電磁気学 (QED) の検証に用いられていて、本実験の目的でもある。本実験では、陽電子 (Ps の発生) をプラスチックシンチレータ、Ps の消滅により出る γ 線を NaI シンチレータで観測し、そこから o-Ps の寿命を測定した。測定データを解析し求められた寿命は 208.6 ns, 188.5 ns であり、理論値の 142 ns とは大きく異なるものだった。その結果を踏まえ、解析手法の見直し・改善を考察した。

目次

第 1 章	目的	2
第 2 章	理論	3
2.1	パラポジトロニウム	3
2.2	オルソポジトロニウム	4
第 3 章	実験	5
3.1	実験原理	5
3.2	実験器具	5
3.3	セットアップ	5
3.4	回路	7
3.5	calibration	8
第 4 章	結果・解析	10
4.1	データ取得	10
4.2	生データ	11
4.3	物理量への変換	12
4.4	TQ 補正	13
第 5 章	誤差評価・考察	27
5.1	誤差評価	27
5.2	考察	30
第 6 章	結論	32
	謝辞	33
	付録 A	33
	参考文献	52

第 1 章

目的

ポジトロニウムとは電子と陽電子の束縛状態であり、スピンの値によってオルソポジトロニウムとパラポジトロニウムの 2 種類に分類される。この 2 つの状態の大きな違いは寿命である。QED による理論計算ではオルソポジトロニウムの寿命は約 142 ns と計算されているのに対してパラポジトロニウムの寿命は約 0.125 ns となっている。本実験の目的は寿命の長いオルソポジトロニウムについて実験により寿命を測定し、理論値と一致するか確かめることで QED を検証することである。

第 2 章

理論

この章ではポジトロニウムの説明を行う。寿命計算についての詳しい計算については参考文献を参照していただきたい。

2.1 パラポジトロニウム

パラポジトロニウムとは電子陽電子対の合成スピンの状態、つまり singlet 状態をさす。

$$|p - Ps\rangle = \frac{1}{\sqrt{2}}(|\uparrow\downarrow\rangle - |\downarrow\uparrow\rangle) \quad (2.1)$$

また QED の結果によると p-Ps は荷電共役変換に対して符号が変わらない。つまり荷電共役変換演算子を C とおくと

$$C |p - Ps\rangle = |p - Ps\rangle \quad (2.2)$$

が成り立つ。これは p-Ps に対する荷電共役変換がパリティ変換とスピンの交換の積の -1 倍と等しくなり、p-Ps はパリティ偶でスピンの入れ替えに対して奇であることから従う。

電子陽電子束縛状態は一般に不安定なのですぐに崩壊し、いくつかの光子を生成する。以下で p-Ps の崩壊によって生じる光子数が偶数であることを見よう。

崩壊後の終状態、つまり光子が n 個ある状態は真空状態に対して n 個の光子生成演算子をかけた状態である。

$$|n\gamma\rangle = a^\dagger(1)a^\dagger(2)\cdots a^\dagger(n)|0\rangle \quad (2.3)$$

ここで演算子の引数は i 番目の光子の波数ベクトルと偏極である。QED ではハミルトニアン密度が荷電共役変換に対して不変であるので*1光子の生成演算子は荷電共役変換により

$$a^\dagger \xrightarrow{C\text{-transf.}} (-1)a^\dagger \quad (2.4)$$

と変換される。従って簡単な計算により

$$|n\gamma\rangle \xrightarrow{C\text{-transf.}} (-1)^n |n\gamma\rangle \quad (2.5)$$

が導かれる。ここで荷電共役変換演算子はユニタリ演算子であることに注意。

*1 適当な場の量子論の教科書 ([2] とか) を参照してください。ただし光子を記述するのは電磁場の Lagrangian であり Dirac 場の Lagrangian ではないことに注意。

2.2 オルソポジトロニウム

オルソポジトロニウムは合成スピンの1すなわち triplet の状態で記述される。

$$|o-Ps\rangle = \begin{cases} |\uparrow\uparrow\rangle \\ \frac{1}{\sqrt{2}}(|\uparrow\downarrow\rangle + |\downarrow\uparrow\rangle) \\ |\downarrow\downarrow\rangle \end{cases} \quad (2.6)$$

$o-Ps$ はスピンの交換で符号を変えないので、荷電共役変換に対して奇であることが分かる。すなわち、

$$C|o-Ps\rangle = (-1)|o-Ps\rangle \quad (2.7)$$

となる。よって前セクションで行った議論によると $o-Ps$ は奇数個の光子に崩壊することが分かる。ただし、電子と陽電子の重心座標系で考えると運動量保存則より $o-Ps$ は1個の光子に崩壊することはあり得ないことが分かるので、 $o-Ps$ は少なくとも3個の光子に崩壊することになる。

以下にオルソポジトロニウムの3光子崩壊のファインマンダイアグラムを示す。

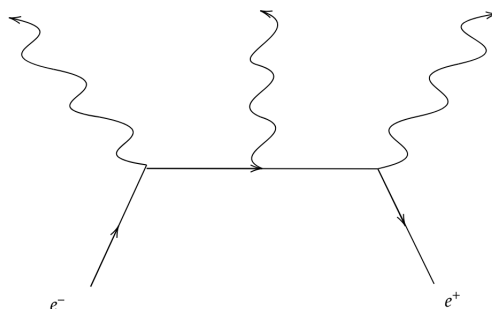


図 2.1: $o-Ps \rightarrow 3\gamma$ のファインマンダイアグラム

みての通りダイアグラムに結節点が3つある。最も注意すべきは結節点が1つ増えるごとに崩壊確率がおおよそ微細構造定数のオーダーだけ小さくなるということである。実際、オルソポジトロニウムの寿命は約 142 ns でありパラポジトロニウムの寿命は約 0.125 ns であることを考えると $o-Ps$ の寿命がおおよそ 10^2 のオーダーで長くなっていることが確かめられる。今回のレポートでは寿命の具体的な計算については実験内容と関係ないため詳細な計算方法については解説しない。各自適当な参考書 ([2] や [1] がオススメ) を参考にされたい。因みに [1] によれば上記のファインマンダイアグラムによる寄与を計算すると寿命は約 135 ns と求まる。より高次の項を計算した結果、寿命は約 142 ns となる。

第3章

実験

3.1 実験原理

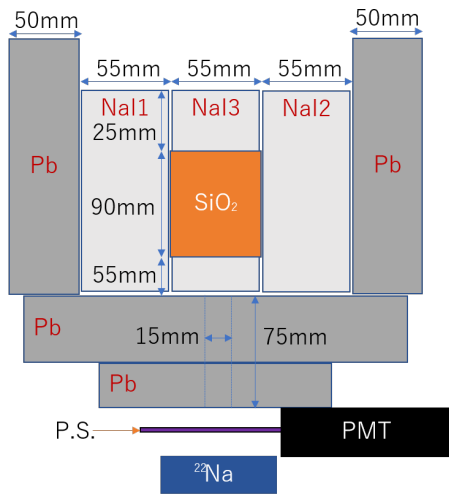
線源の ^{22}Na は β^+ 崩壊し、陽電子 e^+ を放出する。陽電子はシリカパウダー (SiO_2) 内の電子 e^- に束縛され o-Ps、p-Ps を形成する。これでできた Ps は数 ns で崩壊し、 γ 線を放出する。本実験では、陽電子が放出された瞬間にプラスチックシンチレータ (P.S.) で検出し、Ps の崩壊後に出る γ 線を NaI のシンチレータで検出し、その時間差を測定した。ただし、Ps が形成されてから崩壊までにかかる時間以外の経過時間は 0 と近似して解析を行った。

3.2 実験器具

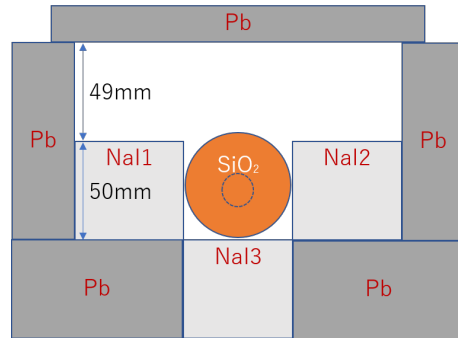
- ^{22}Na :陽電子 e^+ の放射線源。
- シリカパウダー:電子 e^- を多く保有している。
- P.S.: e^+ を検出するシンチレータ。
- NaI: γ 線を検出するシンチレータ。
- PMT:シンチレータからの光子を受け電子に変換し増幅する。
- 鉛ブロック:外部からの放射線を遮断する。
- 遮光ビニール:P.S. に光が入らないようにセットアップ全体を覆う。

3.3 セットアップ

実験器具は図 3.1 のように配置した。点線の部分は鉛に開けられた穴で、線源から出た e^+ はこの穴を通過してシリカパウダーに入る。シリカパウダーは後述の pickoff 反応を減らすため、過熱して水分を飛ばし、容器に入れて真空にした。この容器の配置は最も検出効率が良くなるようにした。

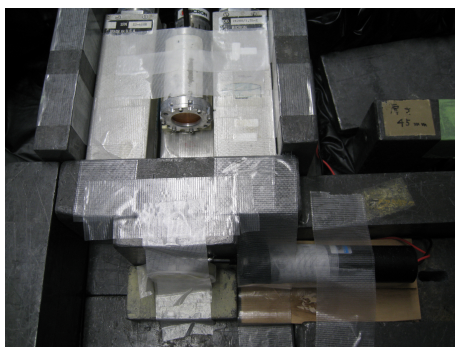


(a) 上から見た図

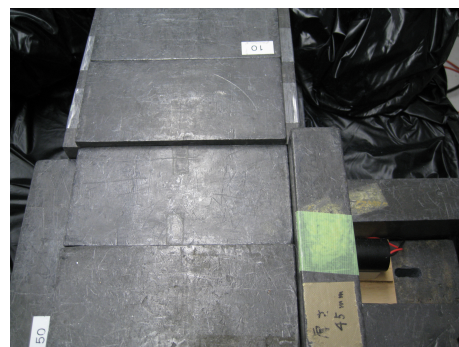


(b) 線源側から見た図

図 3.1: セットアップの模式図



(a) 上から見た配置



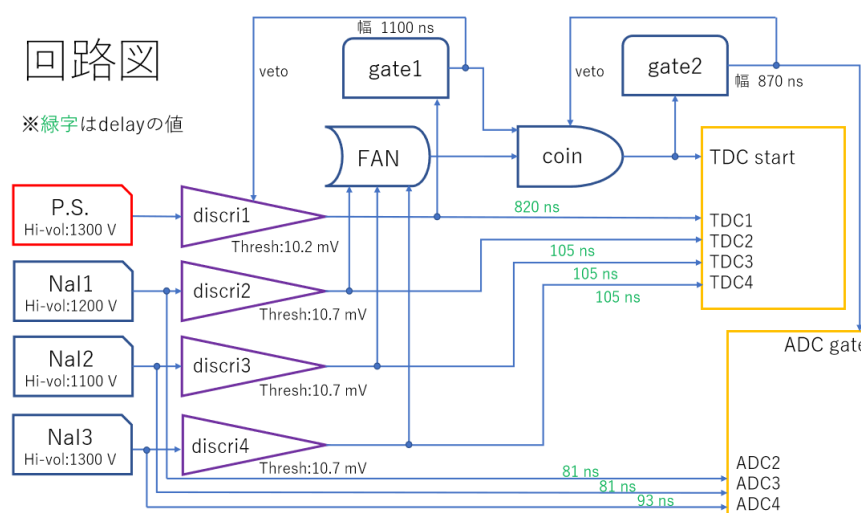
(b) 遮蔽の様子

図 3.2: 実際の配置

3.4 回路

回路は図 3.3 のように組み、また実験における信号の概念図を図 3.4 に示した。この実験において e^+ が P.S. を通過したのちにポジトロニウムが崩壊し γ 線を出して NaI がそれを検出するので、P.S. の信号を TDC の start、NaI の信号を TDC の stop にすることが理想的である。しかし P.S. を通過した e^+ が必ずしもポジトロニウムを形成するとは限らないため目的の現象の検出効率が悪くなる。よって P.S. の信号から gate を生成し、この gate と NaI の信号で coincidence をとりこれを TDC の start とし、TDC の start として P.S. の信号に適切な delay をかけたものを採用する。なお、gate を生成する際に後からやってくる e^+ による信号を無視できるように veto をとっている。このとき、信号にかけた delay の値から TDC の値から計算される時間を引いたものが求めたい崩壊時間である。ここで、実験に使用したモジュール及び機能について以下に説明する。

- Discriminator: 入力された信号が threshold を超えたとき NIM 信号を出力する。
- Coincidence: 入力された NIM 信号の and をとって出力する。
- FAN: 入力された NIM 信号の or をとって出力する。
- Gate Generator: 信号が入力されたとき一定の時間幅の NIM 信号を出力する。
- Veto: gate の信号が出力されている間は次の信号が入力されないようにする。
- TDC: start に信号が入ってから stop に信号が入るまでの時間に比例した値を出力する。
- ADC: gate が開いている間に来た信号の時間積分である電荷に比例した値を出力する。



信号の概念図

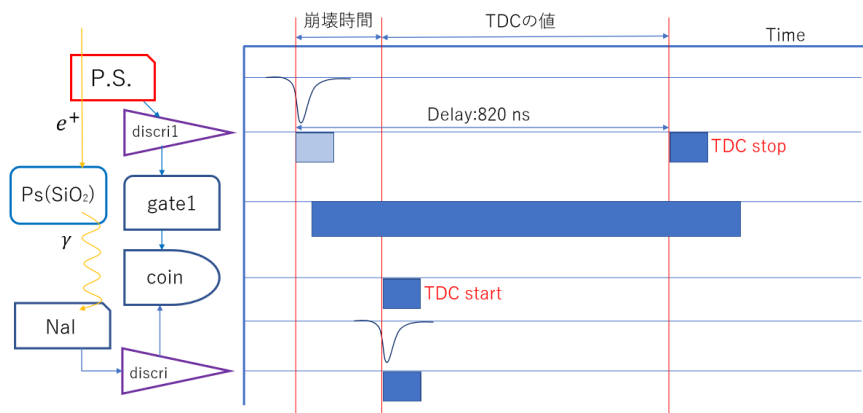


図 3.4: 信号の概念図

3.5 calibration

解析の際、測定した ADC count と TDC count の値を実際のエネルギーと時間に変換する必要がある。そのため、本実験を始める前にこれらに対応させる calibration を行った。

3.5.1 ADC calibration

まずは ADC の calibration から行った。 ^{60}Co , ^{137}Cs , ^{22}Na の 3 つの線源を用いて、それぞれで測定を行った。それぞれの ADC 値のピークを Gaussian で fitting し、それに対応するエネルギー値を表 3.1 にまとめた。ここで pedestal とは何の信号が入らなくても暗電流等によって流れる一定値である。

表 3.1: 各エネルギーにおける ADC count

Energy(source)	NaI1	NaI2	NaI3
0keV(pedestal)	219.51	201.34	251.58
511keV(^{22}Na)	730.35	722.12	800.67
661.7keV(^{137}Cs)	900.22	871.24	957.84
1173.2keV(^{60}Co)	1394.3	1361.5	1467.2
1275keV(^{22}Na)	1493.7	1460.2	1571.8
1332.5keV(^{60}Co)	1548.4	1512.8	1619.8

各 NaI において直線で fitting を行うと、以下の式が得られた。

$$\text{Energy1}[keV] = 1.00765 \times \text{ADC1} - 233.55 \quad (3.1)$$

$$\text{Energy2}[keV] = 1.01948 \times \text{ADC2} - 215.88 \quad (3.2)$$

$$\text{Energy3}[keV] = 0.97571 \times \text{ADC3} - 258.92 \quad (3.3)$$

3.5.2 TDC calibration

次に TDC の calibration を行った。今回、実際の時間が必要なのは寿命を出すために用いる TDC0 の値である。calibration では、ある信号を start に入れるものと delay をかけてから end にわけて、その TDC 値を測定した。delay は実際にオシロスコープを用いて確認した。その結果は表 3.2 になった。

表 3.2: TDC count と時間

Delay[ns]	120	240	360	480	600	720
TDC0	542.8	1009	1469	1936	2404	2871

直線 fitting すると、以下の式が得られた。

$$\text{TIME}[ns] = 0.2578 \times \text{TDC0} - 19.59 \quad (3.4)$$

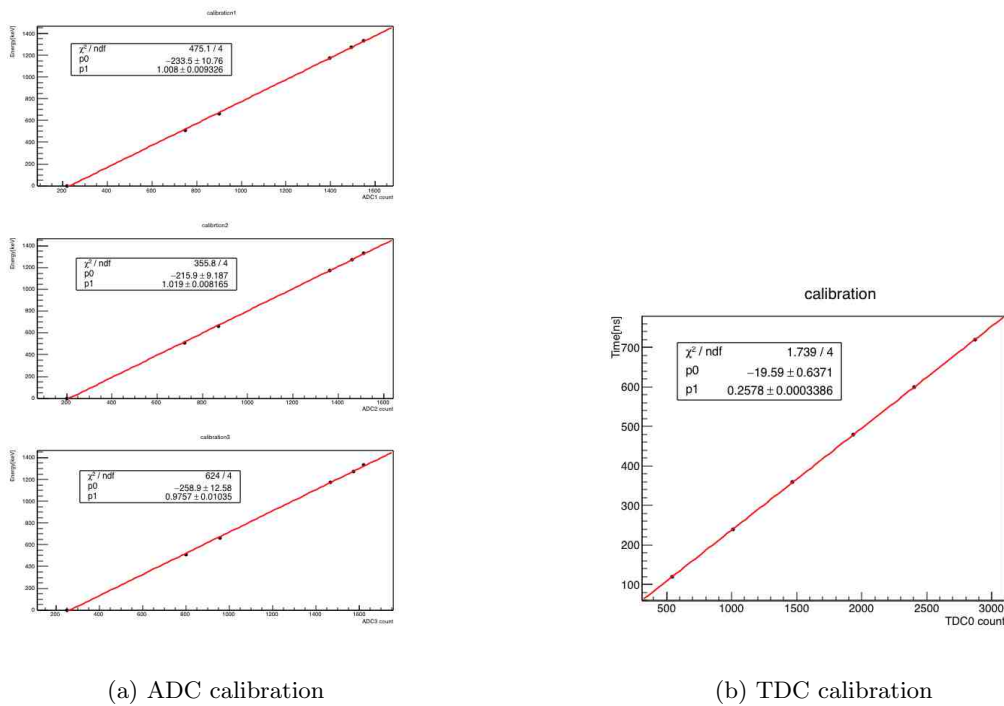


図 3.5: calibration の様子

第 4 章

結果・解析

4.1 データ取得

図 4.1 に示したのは本実験中における各 ADC count の時間変動の様子である。ただし ADC1,2,3 はそれぞれ NaI1,2,3 に対応する。これを見ると実験中の無視できない多くの時間、ADC2 においてデータ取得が正常に行われていなかったことがわかる。その原因としてはセットアップ中の配線の接触不良などが考えられるが、ともかく以降の解析においては NaI2 から得られたデータについては取り扱わないものとした。

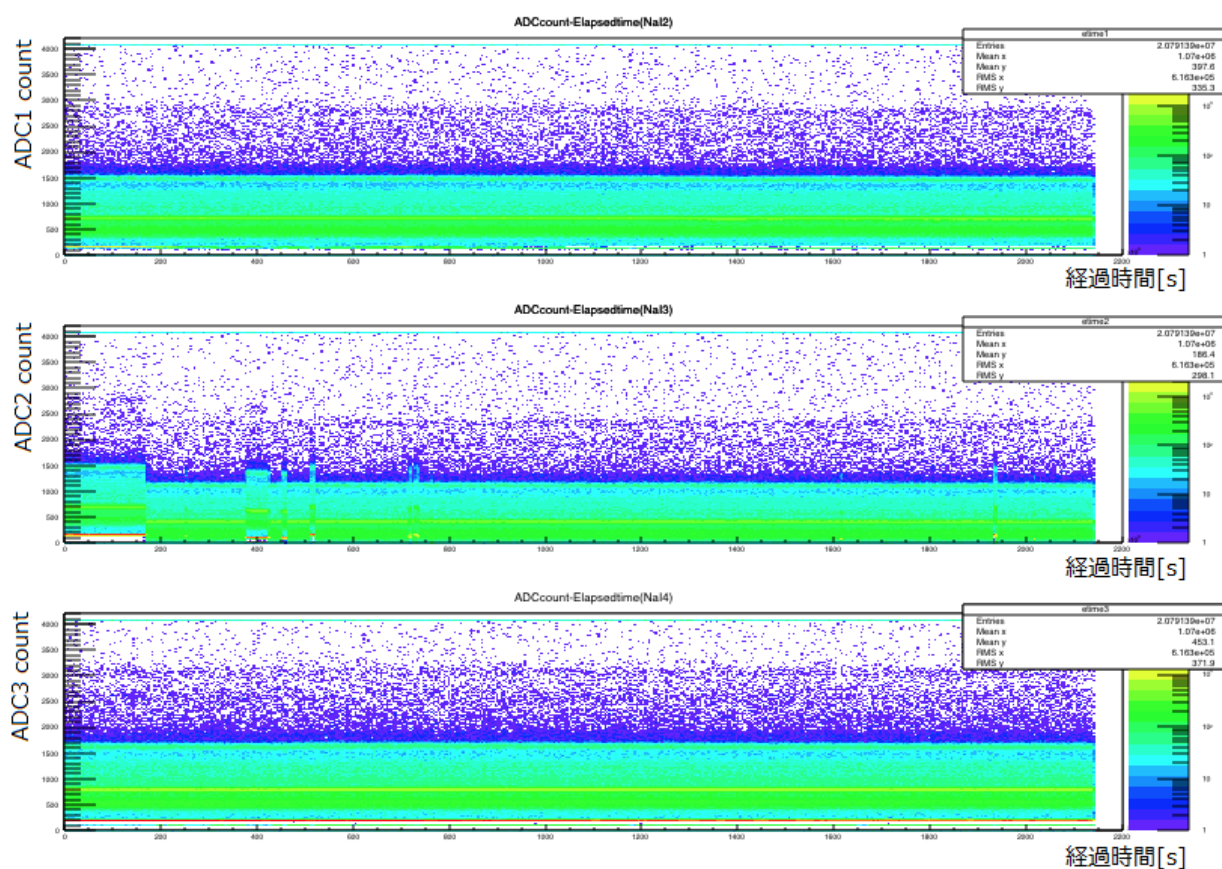


図 4.1: 各 ADC count の時間変動

4.2 生データ

図 4.3 に実際に実験で得られた各 ADC の生データを示す。ここで 511keV,1275keV の各ピーク、およびそれらの Compton 散乱が確かに確認できる。なお取得したデータはおよそ 2.08×10^7 イベントであった。

図 4.4 は実際に実験で得られた各 TDC の生データである。ADC と同様に、TDC1,3 はそれぞれ NaI1,3 に対応する。Ps の寿命測定に際して測定値 Time として使用するのは TDC0 のデータであって、TDC1 および 3 は ADC データの cut 条件として用いる。すなわち、以降の解析において扱いたいデータは各 NaI が Ps 崩壊による γ 線を検出した時のものであるが、現状生データはそれ以外の原因によるデータを多く含んでいる。そこで TDC1,3 が取得した生データ (図 4.4(a),(b)) をもとに目的のデータを選び出すことを考える。ここでは TDC1,3 ともに TDCcount=400 付近に見えるピークがそれぞれ対応する NaI が Ps 崩壊による γ 線を検出したことを表している。そこで目視にて判別のうえ、実際に次の条件を cut 条件として採用した:

表 4.1: 各 NaI が Ps 崩壊による γ 線を検出した条件

	cut 条件
NaI1	$417 \leq \text{TDC1} \leq 423$
NaI3	$414 \leq \text{TDC3} \leq 418$

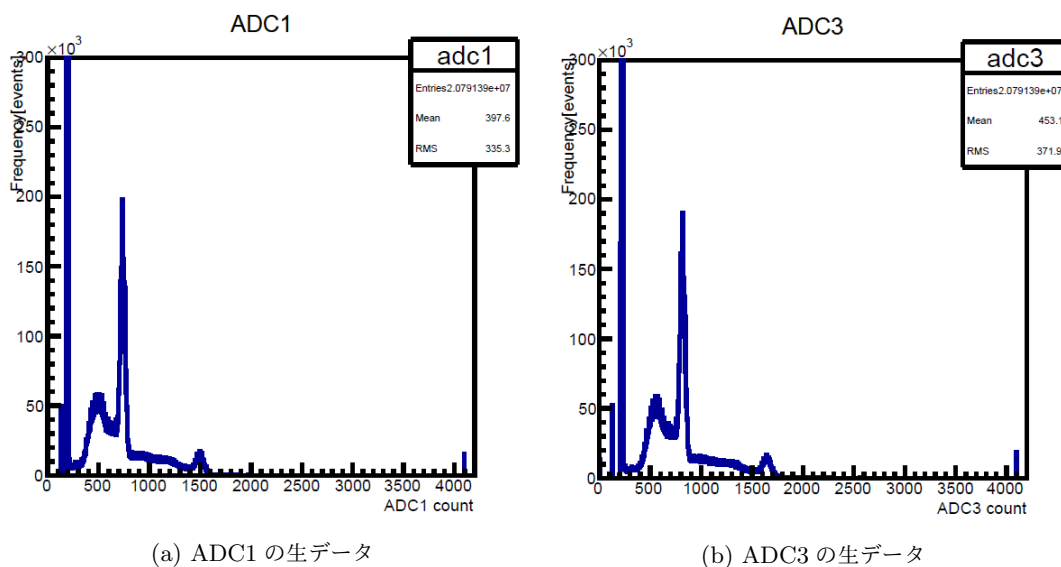
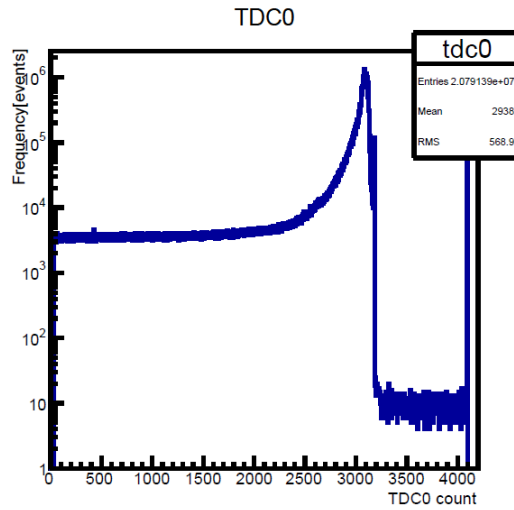
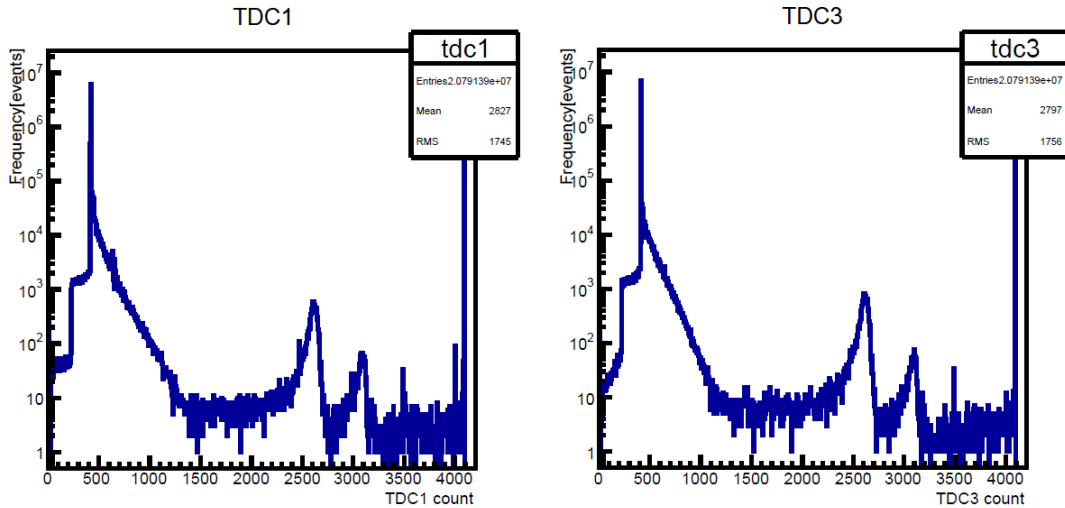


図 4.2: 各 ADC の生データ



(a) TDC0 の生データ



(b) TDC1 の生データ

(c) TDC3 の生データ

図 4.3: 各 TDC の生データ

4.3 物理量への変換

これまで示した calibration 変換式と TDC1,3 による cut 条件を用いて、TDC0 および ADC1,3 の生データ (count の値) をそれぞれ Time および Energy といった物理量へと変換する。変換後の Time[ns] を図 4.4、Energy[keV] を図 4.5 に示す。また、縦軸を Time、横軸を各 NaI に対する Energy として三次元ヒストグラムを描画したものを図 4.6 に示す。なお、TDC0 count を Time に変換する際に用いた変換式は

$$\text{Time[ns]} = -0.2578 \times \text{TDC0} + 19.59 \quad (4.1)$$

であって、先の calibration の節で得た変換式と比べて符号が反転している。これは物理現象の時系列に対して TDC の start と stop が逆転していることによる。

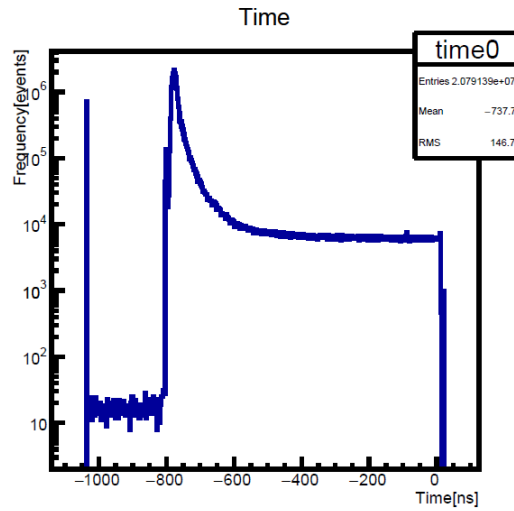
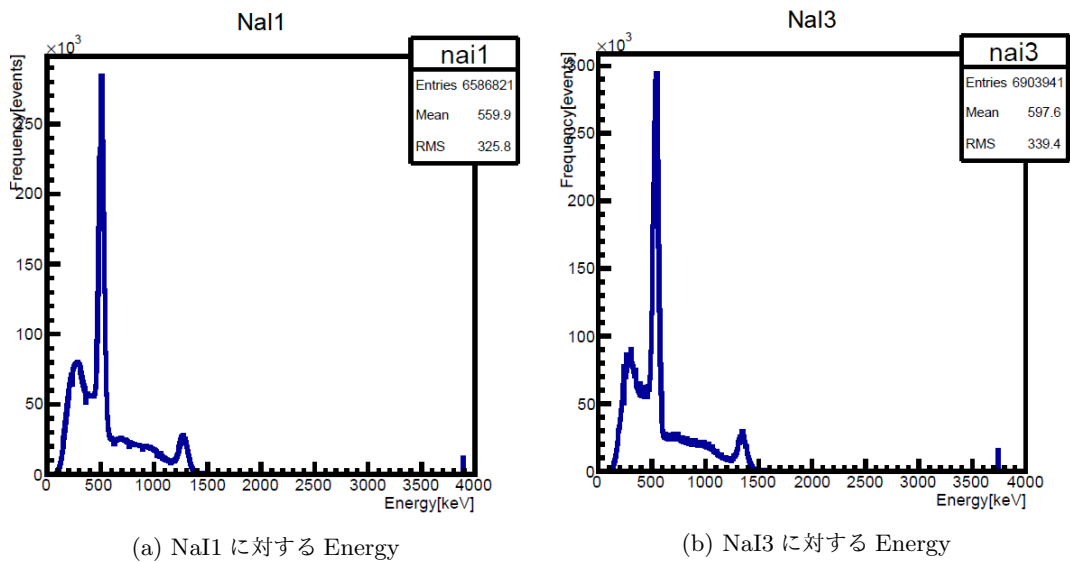


図 4.4: Time



(a) NaI1 に対する Energy

(b) NaI3 に対する Energy

図 4.5: 各 NaI に対する Energy

4.4 TQ 補正

図 4.6 を見ると、Ps 崩壊による 511keV の γ 線およびその Compton 散乱に対応した信号について、Energy が小さいほど Time が大きくなっていることがわかる。すなわち、エネルギーが低いほど遅れて観測されているのである。しかしこの信号は本来同時刻に来ているはずのものであって、この実際の物理現象からのずれは測定装置である discriminator の特性によるものである。そしてこのずれを補正する過程は TQ 補正と呼ばれる。

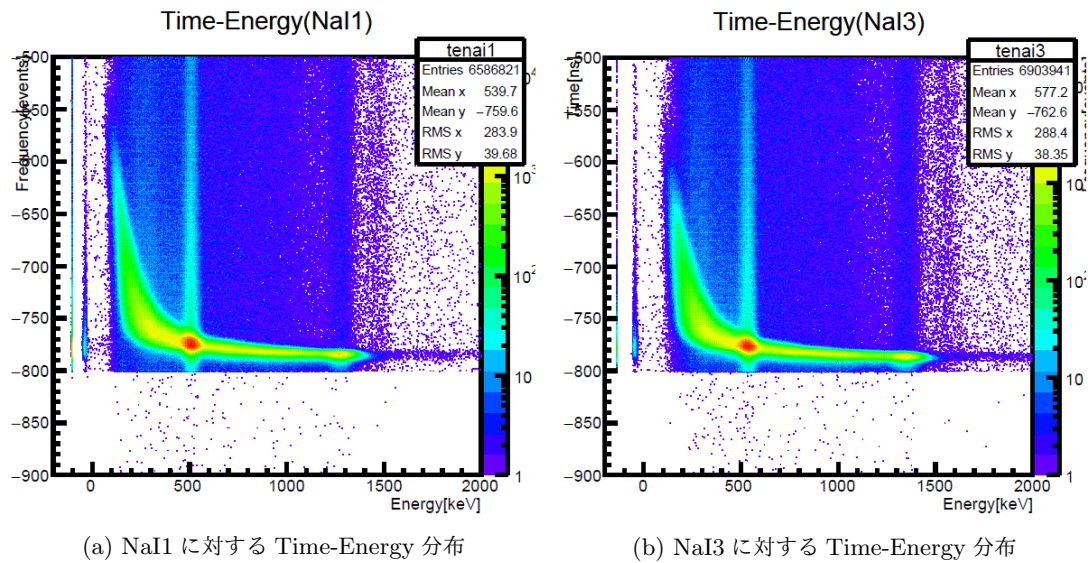


図 4.6: 各 NaI に対する Time-Energy 分布

4.4.1 TQ 補正の理論

まずはなぜこのようなずれが生じるのかを解説する。図 4.7 を参照せよ。

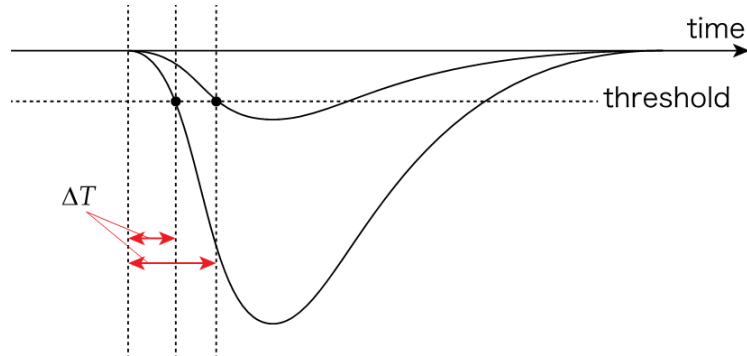


図 4.7: discriminator によるずれ

discriminator とは入力信号の大きさが threshold を超えたときに NIM 信号を出力するものであった。図 4.7 は同時に到達したエネルギーの異なる 2 つの入力信号を模式的に表したものであるが、エネルギーの小さい信号のほうが、エネルギーの大きい信号に比べて threshold までの立ち上がりが遅いことがわかる。すなわち入力信号のエネルギーが小さいほど、波が threshold に至るまでの時間 ΔT は大きい。これがまさに、Energy が小さいほど Time が大きくなっているという、我々が補正すべきずれの原因となっている特性である。

さて、このずれを補正するためには threshold に達するまでの遅れ ΔT が入力信号のエネルギー E に対してどのように変化するかを知る必要がある。すなわちエネルギー E の関数として $\Delta T = \Delta T(E)$ の関数形を

知らねばならない。まずは粗い近似によってこの問題を考えてみよう。図 4.8 を見よ。

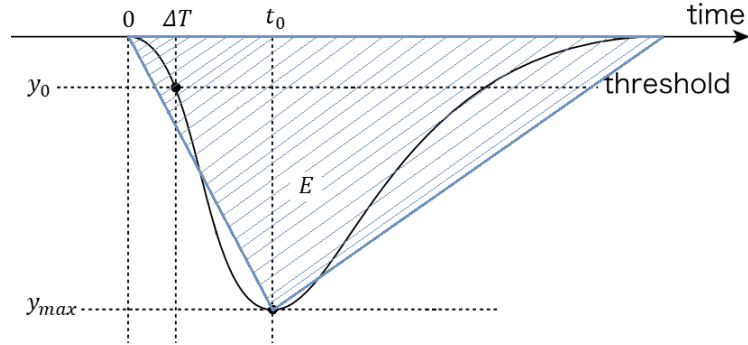


図 4.8: TQ 補正の考え方

入力信号の波形を三角形によって近似することを考える。ここで t_0 は信号がピークに到達するまでの時間、 y_0, y_{max} はそれぞれ threshold, ピークの高さである。いま t_0 を E によらない定数と仮定すると、

$$\Delta T : t_0 = y_0 : y_{max} \quad (4.2)$$

$$\therefore \Delta T = \frac{y_0 t_0}{y_{max}} \propto \frac{1}{E} \quad (4.3)$$

これはかなり粗い近似によって得られた結果であるから、これを元に TQ 補正関数として次を仮定する:

$$\Delta T(E) [ns] = \frac{p_0}{(E [keV] - p_1)^{p_2}} + p_3 \quad (4.4)$$

ただし p_i ($i = 0, 1, 2, 3$) はパラメータであって、以下実際のデータに対する fitting を行って決定する。ただし三角形による近似の結果から、 p_2 は 1 に近い値であることが期待される。

具体的な TQ 補正の手順は次のとおりである:

1. 160keV \pm 5keV の範囲で cut した Time に関するヒストグラムを Gaussian fitting し、その mean の値を 160keV における ΔT の値として採用する。
2. 20keV ごとに 460keV までの 16 点で同様の操作を繰り返す。
3. 得られた ΔT の値を Energy に対してプロットし、TQ 補正関数 (4.4) で fitting、各パラメータ p_i の値を決定する。

このようにして TQ 補正関数 (4.4) が決定されたのち、

4. 各 NaI において、各 Energy E に対して Time[ns] - $\Delta T(E)$ を新たに Time として再定義し、それぞれを各 NaI における Time として採用する。

4.4.2 TQ 補正の実践

図 4.9、および図 4.10 に本解析における実際の TQ 補正の様子を示す。この結果、各 NaI に対する TQ 補正関数 (4.4) の各パラメータは次のように決定された:

表 4.2: 各 NaI に対する TQ 補正関数のパラメータ

	p_0	p_1	p_2	p_3
NaI1	1.567×10^4	61.15	1.096	-794.3
NaI3	1.906×10^4	49.42	1.067	-804.8

ここで、期待されたとおり p_2 の値は 1 に近いものとなっている。

このようにして得られた TQ 補正関数 (4.4) によって各 NaI に対して定義された新しい Time Time[ns] - $\Delta T(E)$ を図 4.11 に示す。また図 4.12 は、その新しい Time と Energy についての三次元ヒストグラムを示したものである。これと TQ 補正前の図 4.6 を比較すると、確かにエネルギーによる時間のずれが補正されていることが見てとれる。

4.4.3 TQ 補正後の寿命

再び図 4.11 を見てほしい。TQ 補正後の Time を関数

$$p_0 e^{-\frac{t[\text{ns}]}{p_1}} + p_2 \quad (4.5)$$

によって fitting したものが図中の赤線である。ただし $p_i (i = 0, 1, 2)$ はパラメータであって、TQ 補正関数のそれとは無関係である。この結果、各パラメータは次のように決定された:

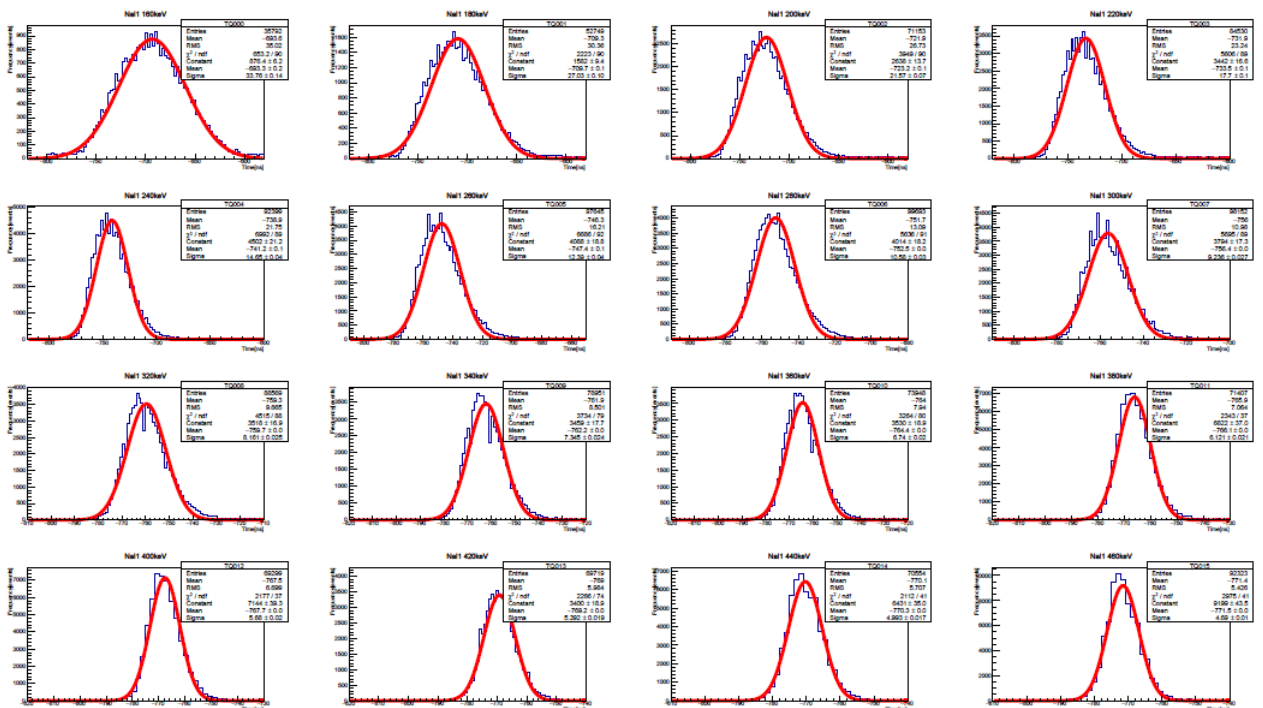
表 4.3: 各 NaI に対する Time fitting 関数のパラメータ

	p_0	p_1	p_2
NaI1	1448 ± 186.0	135 ± 11.5	700.9 ± 8.4
NaI3	1539 ± 250.4	108.6 ± 8.9	541.1 ± 5.0

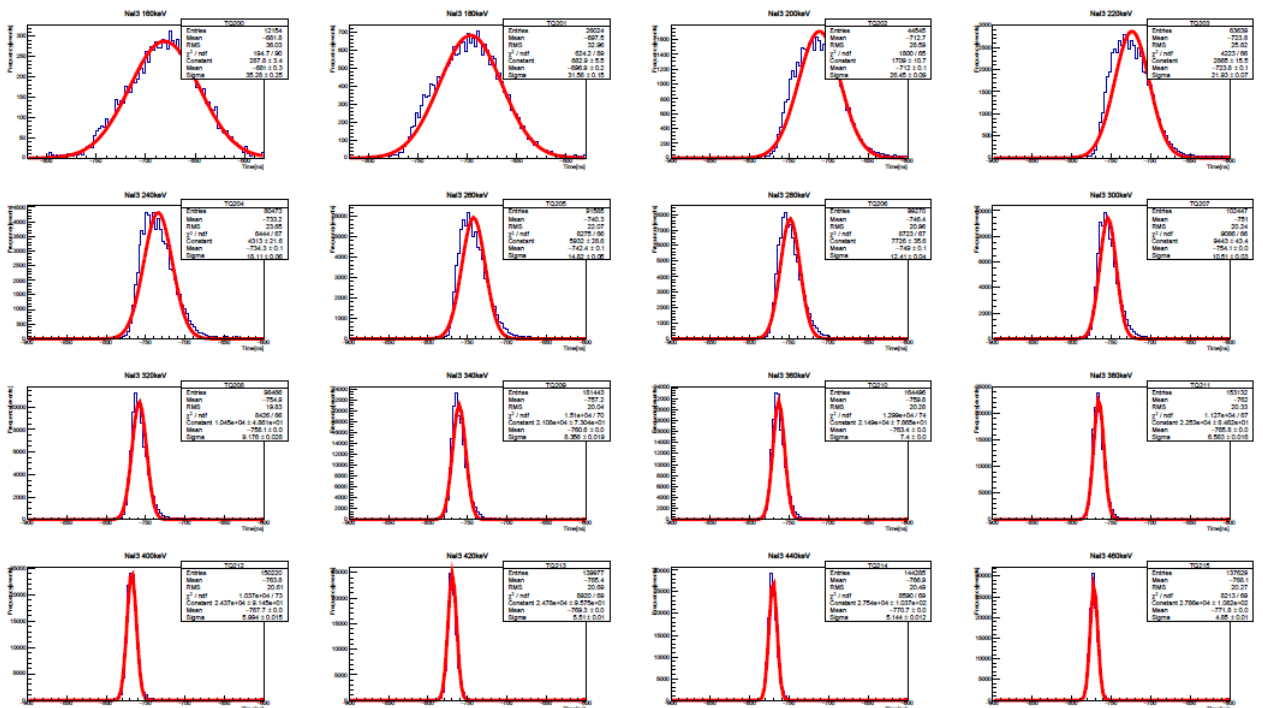
ここで、パラメータ p_1 が求めたい Ps 崩壊の寿命であると考えられる。すなわち、改めて p_1 の値のみを掲載すれば、TQ 補正後の寿命は次のように求めることができたことになる:

表 4.4: 各 NaI に対する TQ 補正後の寿命

	寿命 [ns]
NaI1	135 ± 11.5
NaI3	108.6 ± 8.9

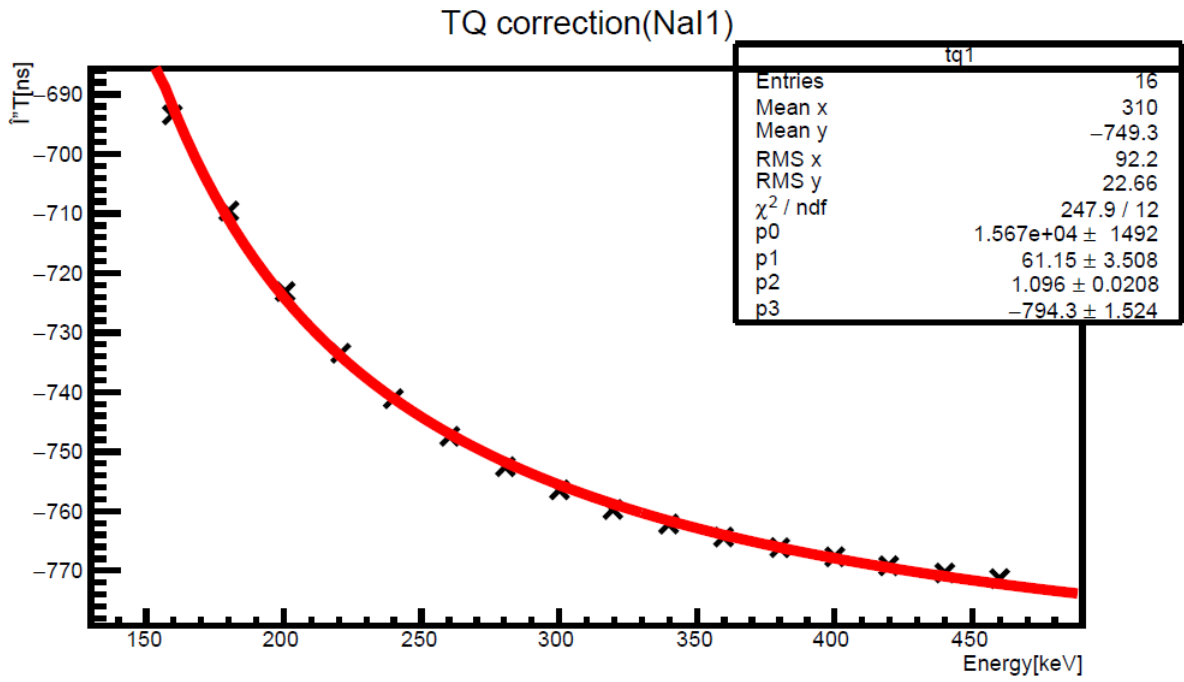


(a) NaI1 に対する TQ 補正

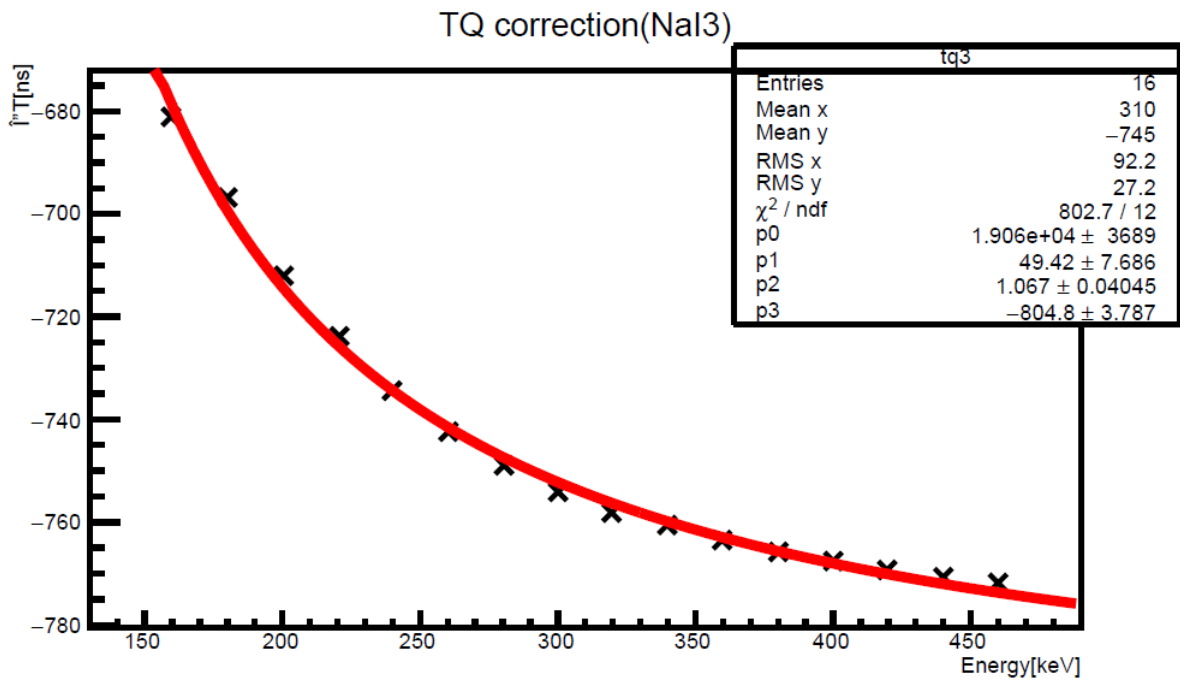


(b) NaI3 に対する TQ 補正

図 4.9: 各 NaI に対する TQ 補正

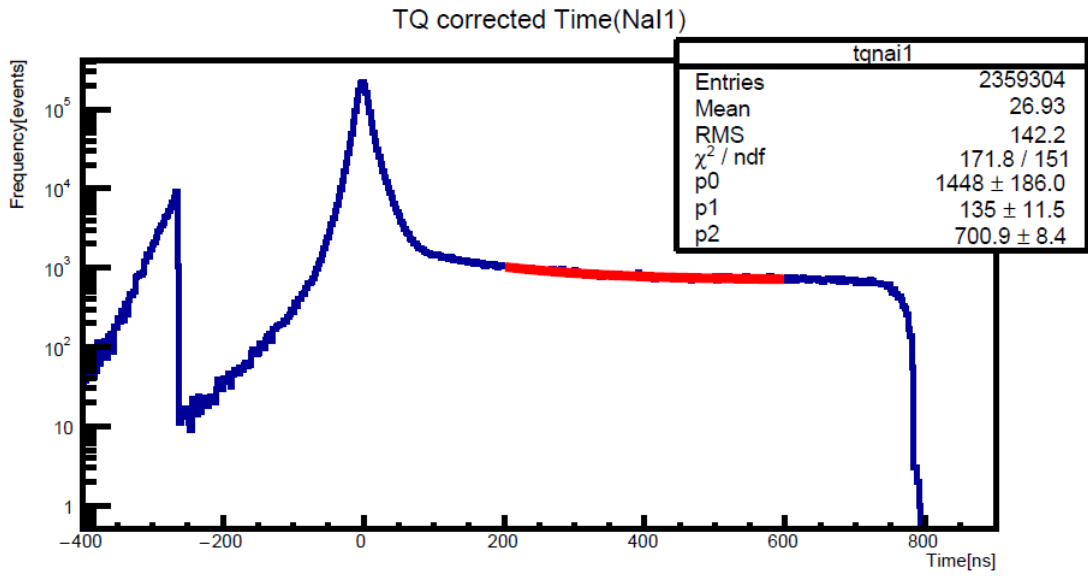


(a) NaI1 に対する TQ 補正関数 fitting

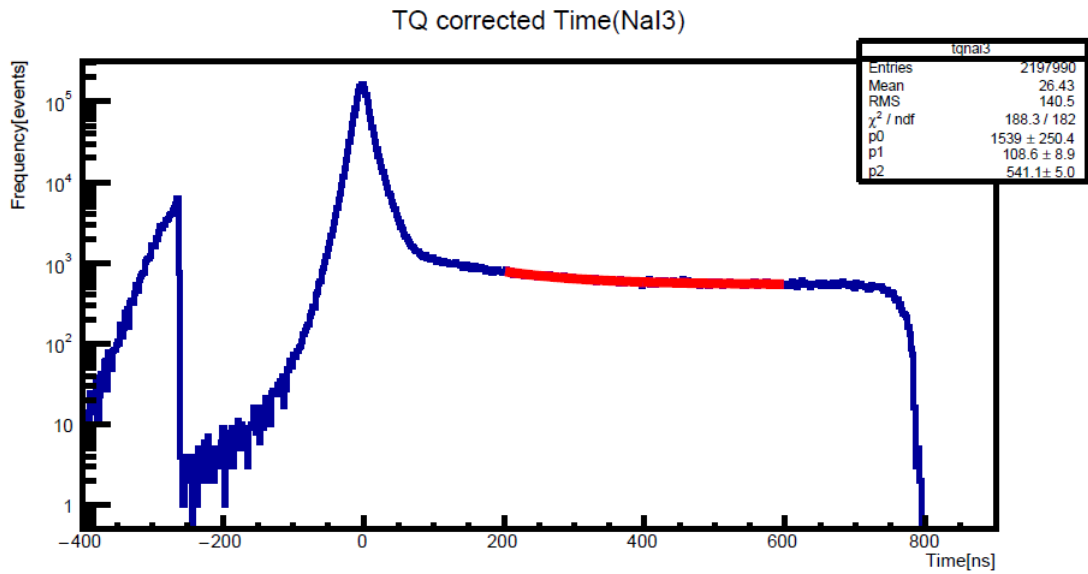


(b) NaI3 に対する TQ 補正関数 fitting

図 4.10: 各 NaI に対する TQ 補正関数 fitting

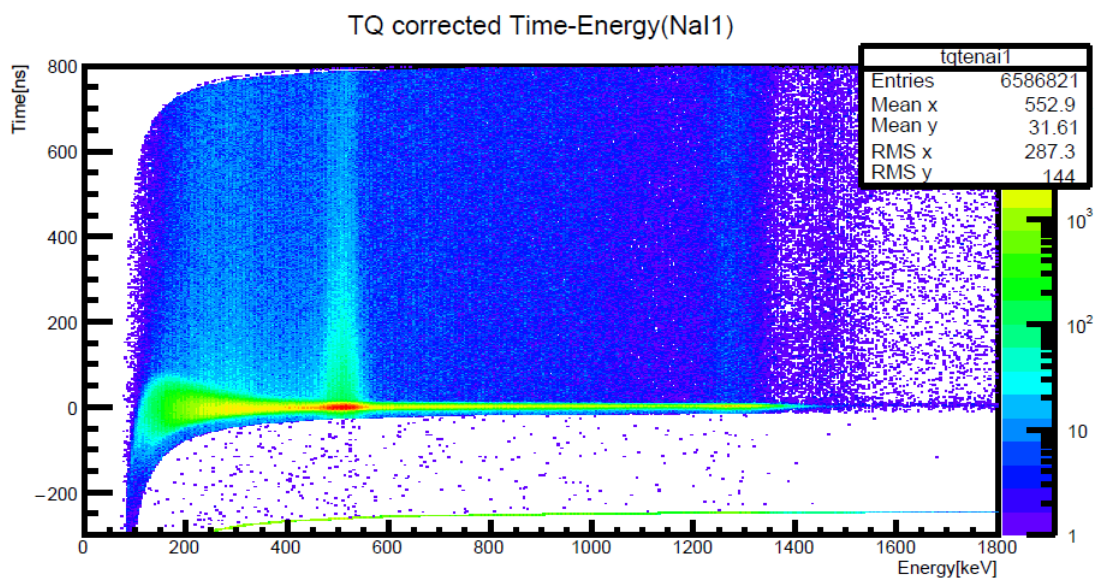


(a) NaI1 における TQ 補正後の Time

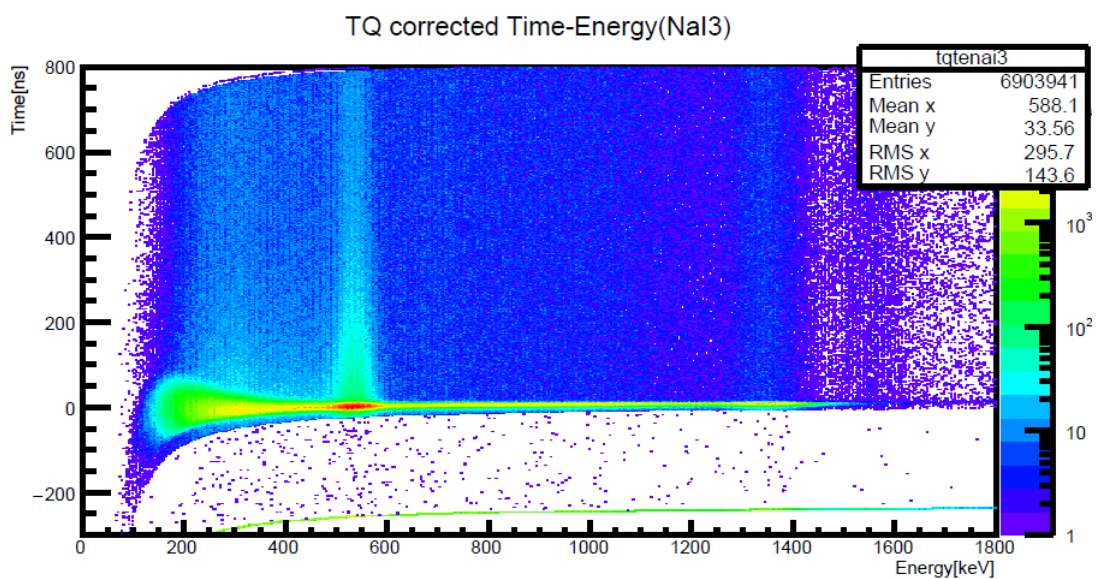


(b) NaI3 における TQ 補正後の Time

図 4.11: 各 NaI における TQ 補正後の Time



(a) NaI1 における TQ 補正後の Time-Energy 分布



(b) NaI3 における TQ 補正後の Time-Energy 分布

図 4.12: 各 NaI における TQ 補正後の Time-Energy 分布

4.5 Pick-Off 補正

4.5.1 Pick-Off 反応とは

オルソポジトロニウム(o-Ps)は以下の反応により 3γ ではなく 2γ を放出することがある。

1. Pick-Off 反応

o-Ps の持つ陽電子が他の原子の持つ電子と衝突して対消滅を起こす。

2. スピン交換反応

物質を構成する不対電子を持つ分子の電子と o-Ps の電子がスピンを入れ替えて p-Ps となる。

3. 化学反応

o-Ps が酸化されて電子が奪われ残った陽電子が対消滅する

以下この 3 つを合わせて Pick-Off 反応と呼ぶことにする。

先の TQ 補正後の崩壊寿命はこの pick-off 反応によるエネルギー線のデータを混ぜて算出していたと思われる。以降我々が算出したいのは「o-Ps が 3γ 線に崩壊する崩壊寿命(pick-off 反応による p-Ps への崩壊は含めない)」であるとする。

4.5.2 方法論

※付録 A2 にて以下の崩壊寿命の算出理論の可否について再考する。

- ・ o-Ps による 3γ のエネルギーは 511keV 未満であること
- ・ o-Ps は p-Ps と比べて寿命が比較的長いこと(p-Ps のそれは o-Ps のそれと比べて 1000 分の 1 程度と言われている)
 - ・ よって Time=0ns では p-Ps(と 1275keV のエネルギー)が支配的であるというモデルを採用する

これらのことより TQ 補正後の Time-Energy 2 次元ヒストグラムを以下の領域に大別してみる。

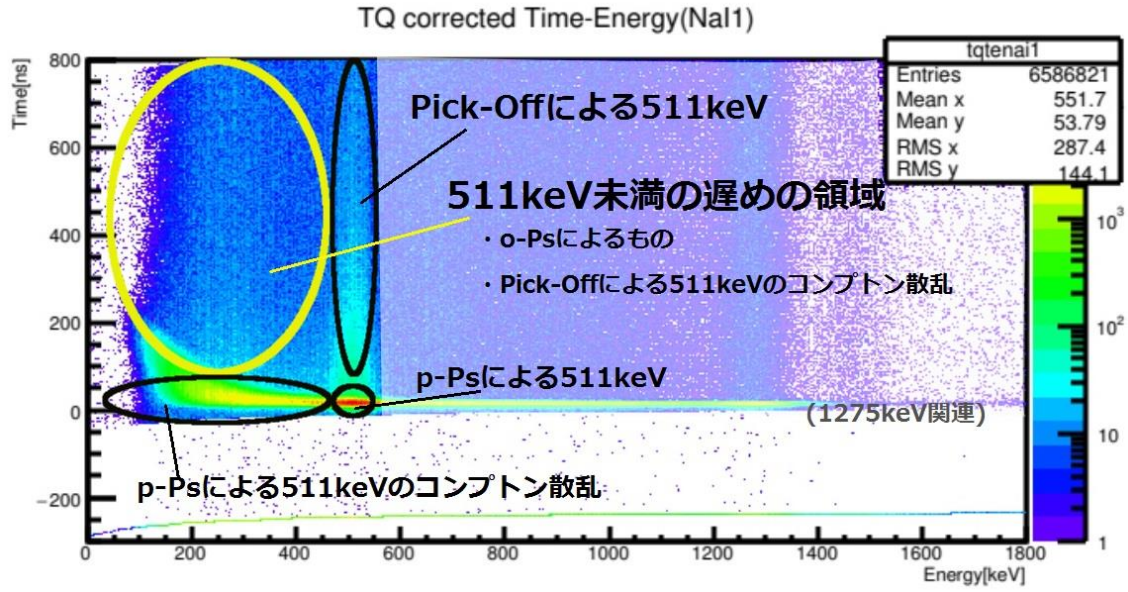


図 4.13: TQ 補正後の Time Energy ヒストグラムの見取りイメージ図

「511keV未満の(遅めの)領域」には *o*-Ps の 3 γ 線と「Pick-Offによる511keV」の領域による511keV線のコンプトン散乱が含まれると考える(図 4.13 は NaI1 におけるヒストグラム)。またこの図はあくまでもイメージ図であり本実験におけるデータを反映したものではない。

$\Gamma_{3\gamma}$: *o*-Ps の 3 γ への崩壊幅、 $\Gamma_{pickoff}$: *o*-Ps の pick-off 反応への崩壊幅 として、

$$f(t) \equiv \frac{\Gamma_{pickoff}}{\Gamma_{3\gamma}} \quad (4.6)$$

なる $f(t)$ を定める。

$N_{o-Ps}(t)$: 時刻 t にて生存している *o*-Ps の数、また $\tau_{3\gamma} = \frac{1}{\Gamma_{3\gamma}}$ (=const とする) より

$$\begin{aligned} \frac{dN_{o-Ps}(t)}{dt} &= -(\Gamma_{3\gamma} + \Gamma_{pickoff})N_{o-Ps}(t) = -\Gamma_{3\gamma}(1 + f(t))N_{o-Ps}(t) \\ &= -\frac{1 + f(t)}{\tau_{3\gamma}}N_{o-Ps}(t) \end{aligned} \quad (4.7)$$

この微分方程式を解くと、

$$N_{o-Ps}(t) = N_0 \exp\left(-\int_{t_0}^t \frac{1 + f(t')}{\tau_{3\gamma}} dt'\right) \quad (4.8)$$

となる(N_0 : 定数)。よって、

$$\frac{dN_{o-Ps}(t)}{dt} = -\frac{1 + f(t)}{\tau_{3\gamma}}N_0 \exp\left(-\int_{t_0}^t \frac{1 + f(t')}{\tau_{3\gamma}} dt'\right) \quad (4.9)$$

となる。

以下ではこの $f(t)$ の決定と、それを以ての上記の $N_{o-Ps}(t)$ の式を TQ 補正後の Time の 1 次ヒストグラムに fitting することで $\tau_{3\gamma}$ の値を決定する。

4.5.3 $f(t)$ を決定する

$$f(t) = \frac{\Gamma_{2\gamma}}{\Gamma_{3\gamma}} = \frac{\Delta N_{2\gamma}(t)}{\Delta N_{3\gamma}(t)} \quad (4.10)$$

を用いて $f(t)$ を決める。

例年にならい、

(1) 図 4.13 より各 Time=0ns では p-Ps による反応(と 1275keV による反応)が支配的であるとして、

(2) 各 t にての pick-off 反応による 2γ 線の Energy ヒストグラムにおける分布は、Time=0ns での p-Ps の崩壊による 2γ 線の Energy ヒストグラムにおける分布をスケール調節したものであるとする。すなわち、

・ Time=0([ns])における Δt の範囲にての 511keV のピークのイベント数(とはこのピーク内のイベント数を数え上げることを意味する(下図 4.14 参照))を $y(0)$ 、ある Energy 範囲のイベント数を $A(0)$ 、

・ Time=t([ns]) における Δt の範囲にての 511keV のピークのイベント数を $y(t)$ 、Time=t での上の Time=0 での $A(0)$ に該当する Energy 範囲のイベント数を $A(t)$ とすれば、

$$A(t) = \frac{y(t)}{y(0)} \times A(0) \quad (4.11)$$

であるとする。(「pick-off による 511keV の領域」は o-Ps の崩壊による 3γ 線を含まないとするため以下ではここを基準に「511keV 未満の遅めの領域」内の pick-off 反応による 2γ 線(コンプトン散乱するもの)のイベント数を測ることを試みる。)

というモデルを採用する。

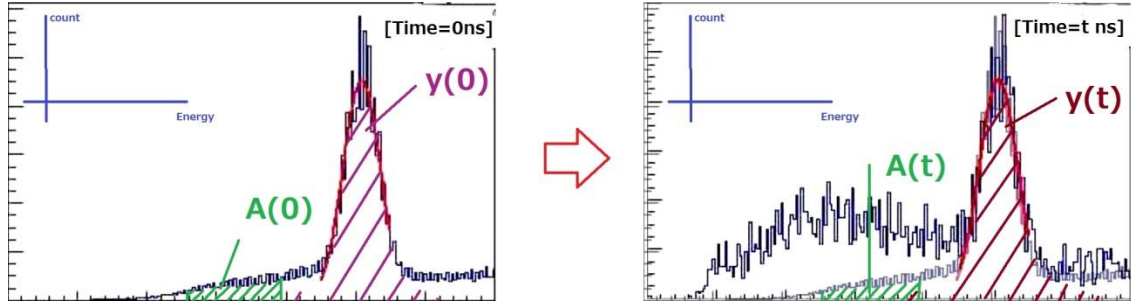


図 4.14: スケール調節のイメージ図

右図は $\text{Time}=t[\text{ns}]$ における Δt の範囲にての Energy のヒストグラムに、スケール調節した $\text{Time}=0[\text{ns}]$ におけるヒストグラムを重ねたもので、式(4.11)を満たすとする。

以下、

- ・ $\Delta N_{2\gamma}(t) : \text{Time}=t[\text{ns}]$ における Δt の範囲にての「511keV 未満の遅めの領域」における(コンプトン散乱する)2 γ 線のイベント数、
- ・ $\Delta N_{3\gamma}(t) : \text{Time}=t[\text{ns}]$ における Δt の範囲にての「511keV 未満の遅めの領域」における 3 γ 線のイベント数、
- ・ $S(t) : \text{Time}=t[\text{ns}]$ における Δt の範囲にての「511keV 未満の遅めの領域」における総イベント数とし、

$$S(0) = \Delta N_{2\gamma}(0)$$

$$\Delta N_{2\gamma}(t) = \frac{y(t)}{y(0)} S(0) \quad (4.12)$$

$$\Delta N_{3\gamma}(t) = S(t) - \Delta N_{2\gamma}(t)$$

とする。(離散的な) t における $\left(t, f(t) = \frac{\Delta N_{2\gamma}(t)}{\Delta N_{3\gamma}(t)} = \frac{y(t)\Delta N_{2\gamma}(0)}{S(t)y(0) - y(t)\Delta N_{2\gamma}(0)} \right)$ の点々から $f(t)$ を推量する。

4.5.4 データ処理

- ・ 用いたソースプログラム (ROOT マクロ) は付録 A1 を参照すべし。

・ 「511keV 未満の遅めの領域」に $\frac{1+f(t)}{\tau_{3\gamma}} N_0 \exp\left(-\int_{t_0}^t \frac{1+f(t')}{\tau_{3\gamma}} dt'\right)$ の式を当てはめることを考えているため、図 4.12 をもとに「p-Ps による 511keV のコンプトン散乱 + p-Ps による 511keV」領域のかからぬ $\text{Time}=150[\text{ns}]$ より、 $50[\text{ns}]$ の間隔で $500[\text{ns}]$ まで $\pm 3[\text{ns}]$ の範囲でデータを抽出して $\frac{\Delta N_{2\gamma}(t)}{\Delta N_{3\gamma}(t)}$ をプロットする。

(後述 : $500[\text{ns}]$ 以上の範囲まで抽出してよかったのではと思われる。)

・ $y(t)$ は、各 511keV ピークをガウス関数(ROOT における "gaus") で fitting してそれから得た parameter(この場合「 $[0] \cdot \exp(-0.5 \cdot ((x-[1])/[2])^2)$ 」に fitting している)をもとに、 $z(t)=[0] \cdot [2]$ として

$\frac{y(t)}{y(0)} \cong \frac{z(t)}{z(0)}$ として計算している(これらの説明は付録 A1 にて行う)。

・ $S(t)$ は各 t 一律に $0 \leq \text{Energy} < 450$ の範囲のイベント数とした

その結果求めた $f(t)$ は下図のようにプロットされ、それに対して例年にならって

$$f(t) = p0' \exp\left(-\frac{t}{p1'}\right) + p2' \quad (4.13)$$

で fitting する($p0', p1', p2'$ の初期値は共に(0.5,115,0.2))。結果は図 4.15、表 4.5 のようになった。

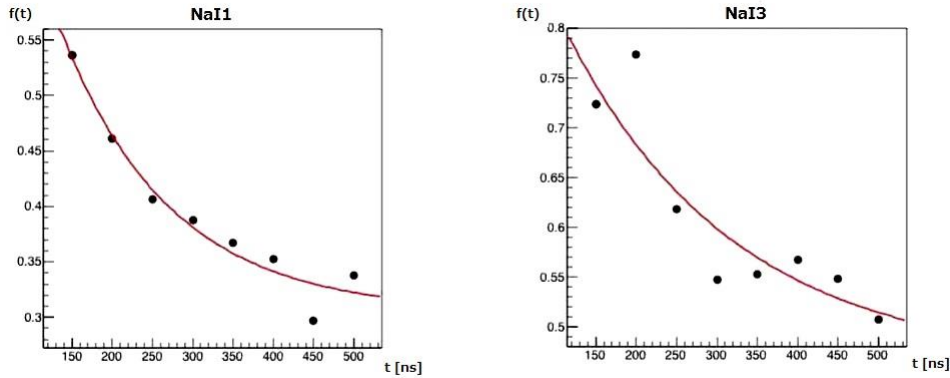


図 4.15: $f(t)$ のプロット

表 4.5: $f(t)$ の fitting における各パラメータ値

	<i>NaI1</i>	<i>NaI3</i>
$p0'$	0.6918±0.1823	0.5773±0.1825
$p1'$	136.2±41.4	208.0±177.4
$p2'$	0.3048±0.0260	0.4625±0.1198

4.5.5 崩壊寿命 $\tau_{3\gamma}$ の決定

$$\frac{dN_{o-ps}(t)}{dt} = -\frac{1+f(t)}{\tau_{3Y}} N_0 \exp\left(-\int_{t_0}^t \frac{1+f(t')}{\tau_{3Y}} dt'\right) \quad (4.14)$$

の式に先程決めた $f(t) = p0' \exp\left(-\frac{t}{p1'}\right) + p2'$ を代入すると式は

$$\frac{dN_{o-ps}(t)}{dt} = \frac{C}{\tau_{3Y}} \left(p0' \exp\left(-\frac{t}{p1'}\right) + p2' + 1\right) \exp\left(-\frac{1}{\tau_{3Y}} \left(p0' p1' \exp\left(-\frac{t}{p1'}\right) + (p2' + 1)t\right)\right) \quad (4.15)$$

となるので、よって「511keV 未満の(遅めの)領域」に再度以下の式

$$q0 \left(p0' \exp\left(-\frac{t}{p1'}\right) + p2' + 1\right) \exp\left(-\frac{1}{q1} \left(p0' p1' \exp\left(-\frac{t}{p1'}\right) + (p2' + 1)t\right)\right) + q2 \quad (4.16)$$

で fitting する。求める τ_{3Y} は $q1$ の値である。fitting は、(一律に) $0 \leq \text{Energy} < 450$ ([keV]) の範囲で抽出した Time の 1 次ヒストグラムに 150 から 500 ([ns]) の範囲で行う ($q0, q1, q2$ の初期値は共に (200, 160, 170))。結果は図 4.16、表 4.6 のようになった。

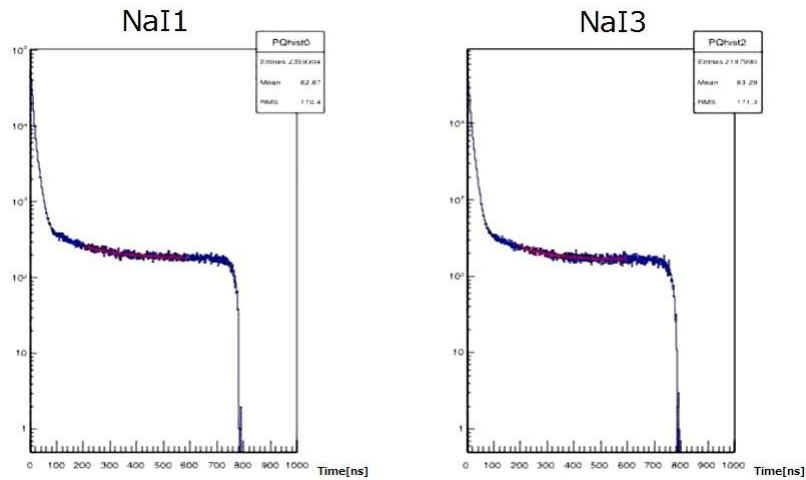


図 4.16: Time ヒストグラムへの fitting

表 4.6: Time ヒストグラムへの fitting における各種パラメータ値

	<i>NaI1</i>	<i>NaB</i>
$q0$	186.1±20.6	169.8±21.4
$q1$	208.6±20.5	188.5±16.8
$q2$	177.5±2.6	164.9±1.7

第 5 章

誤差評価・考察

改めてここまでの解析の結果、o-Ps の寿命は以下のように求められた。

表 5.1: 最終的に求められた寿命値

	寿命 [ns]
NaI1	208.6 ± 20.5
NaI3	188.5 ± 16.8

5.1 誤差評価

上記結果における誤差は最後のフィッティングによる誤差のみである。しかし、本解析では複数回のフィッティングを行っており、それぞれのフィッティングで求めたパラメータ値には誤差が生じる。各フィッティングでパラメータ値に生じた誤差が、最終的な寿命に与える影響を、以下で評価していく。

5.1.1 TDC1 Calibration 関数の誤差

TDC1 の値を時間に変換する calibration の変換式は以下のように求められる。

$$Time[ns] = a \times (TDC) + b \quad (5.1)$$

$$a = 0.2578 \pm 0.0003386 \quad (5.2)$$

$$b = -19.51 \pm 0.6371 \quad (5.3)$$

a,b の値の誤差により、式 5.2 にも誤差 $\delta Time$ が生じ、それは

$$\begin{aligned} \delta Time[ns] &= \sqrt{\left(\frac{\partial Time}{\partial a}\right)^2 (\delta a)^2 + \left(\frac{\partial Time}{\partial b}\right)^2 (\delta b)^2} \\ &= \sqrt{(\delta a)^2 x^2 + (\delta b)^2} \end{aligned} \quad (5.4)$$

と求められる。Calibration 式 5.2 を $Time \pm \delta Time$ で与えた場合の寿命をそれぞれ τ_+, τ_- とし求めると、寿命は表 5.2 のようになる。なお、表の再右列には最終的な寿命値の誤差を載せており、

$$\sigma = \max \{ |\tau_+ - \tau|, |\tau_- - \tau| \}$$

とした。

表 5.2: Calibration 誤差を考慮した寿命値と誤差

	TQ 補正後 [ns]	PickOff 補正後 [ns]	誤差 [ns]
NaI1	$\tau_+ = 135.1$	$\tau_+ = 208.0$	$\sigma = 6.6$
	$\tau = 135.0$	$\tau = 208.6$	
	$\tau_- = 135.6$	$\tau_- = 215.2$	
NaI3	$\tau_+ = 109.0$	$\tau_+ = 187.7$	$\sigma = 3.6$
	$\tau = 108.6$	$\tau = 188.6$	
	$\tau_- = 108.8$	$\tau_- = 192.2$	

5.1.2 TQ 補正関数による誤差

TQ 補正関数についても同様に考えていく。TQ 補正関数

$$\Delta T(E)[ns] = \frac{p_0}{(E[keV] - p_1)^{p_2}} + p_3 \quad (5.5)$$

のパラメータ p_0, p_1, p_2, p_3 はフィッティングにより誤差つきで求まる。

表 5.3: フィッティングにより求められた TQ 補正関数のパラメータ

	p_0	δp_0	p_1	δp_1	p_2	δp_2	p_3	δp_3
NaI1	16618.1	3263.46	60.0457	4.39879	1.10727	0.0399413	-794.041	1.67997
NaI3	18217.9	3913.14	50.4783	7.1695	1.05872	0.0416609	-804.968	3.0664

各パラメータ値誤差による TQ 補正関数 ΔT の誤差 $\delta\Delta T$ は次のように求められる。

$$\begin{aligned} \delta\Delta T(E) &= \sqrt{\sum_{i=0}^3 \left(\frac{\partial\Delta T}{\partial p_i} \delta p_i \right)^2} \\ &= \sqrt{\left\{ \frac{1}{(E-p_1)^{p_2}} \delta p_0 \right\}^2 + \left\{ \frac{p_0 p_2}{(E-p_1)^{p_2+1}} \delta p_1 \right\}^2 + \left\{ \frac{-p_0 \ln(E-p_1)}{(E-p_1)^{p_2}} \delta p_2 \right\}^2 + (p_3)^2} \\ &= \frac{1}{(E-p_1)^{p_2}} \sqrt{(\delta p_0)^2 + \left(\frac{p_0 p_2}{E-p_1} \right)^2 (\delta p_1)^2 + (p_0 \ln(E-p_1))^2 (\delta p_2)^2 + ((E-p_1)^{p_2})^2 (\delta p_3)^2} \quad (5.6) \end{aligned}$$

TQ 補正関数を ΔT の代わりに、 $\Delta T + \delta\Delta T$ 、 $\Delta T - \delta\Delta T$ で与えた場合の寿命をそれぞれ $\tau_{+\delta\Delta T}$ 、 $\tau_{-\delta\Delta T}$ とし、それぞれの寿命を求めると次の表 5.4 のようになった。

表 5.4: TQ 補正関数の誤差を考慮した寿命値と誤差

	TQ 補正後 [ns]	PickOff 補正後 [ns]	誤差 [ns]
	$\tau_{+\delta\Delta T} = 134.7$	$\tau_{+\delta\Delta T} = 221.7$	
NaI1	$\tau = 135.0$	$\tau = 208.6$	$\sigma = 13.1$
	$\tau_{-\delta\Delta T} = 132.0$	$\tau_{-\delta\Delta T} = 205.0$	
	$\tau_{+\delta\Delta T} = 106.3$	$\tau_{+\delta\Delta T} = 172.7$	
NaI3	$\tau = 108.6$	$\tau = 188.6$	$\sigma = 15.3$
	$\tau_{-\delta\Delta T} = 114.3$	$\tau_{-\delta\Delta T} = 182.2$	

5.1.3 Pick-Off 補正による誤差

最後に Pick-Off 補正における誤差を考えていく。Pick-Off 補正関数 $f(t)$ と最終的な寿命フィッティング関数 $g(t)$ は以下のように表される。

$$f(t) = p_0 \exp\left(-\frac{t}{p_1}\right) + p_2 \quad (5.7)$$

$$\begin{aligned} g(t) &= q_0(1 + f(t)) \exp\left(-\frac{1}{q_1} \int^t (1 + f(t')) dt'\right) + q_2 \\ &= q_0 \left\{ p_0 \exp\left(-\frac{t}{p_1}\right) + p_2 + 1 \right\} \exp\left[-\frac{1}{q_1} \left\{ -p_0 p_1 \exp\left(-\frac{t}{p_1}\right) + (p_2 + 1)t \right\}\right] + q_2 \end{aligned} \quad (5.8)$$

Pick-Off 補正関数 $f(t)$ のパラメータ p_0, p_1, p_2 はフィッティングにより誤差つきで求められる。

表 5.5: フィッティングにより求められた Pick-Off 補正関数のパラメータ

	p_0	δp_0	p_1	δp_1	p_2	δp_2
NaI1	0.691831	0.18221	136.175	41.3074	0.304807	0.0259341
NaI3	0.577303	0.182485	207.988	177.357	0.462527	0.119716

その誤差により、最終的な寿命フィッティング関数 $g(t)$ にも誤差が生じる。その誤差 $\delta g(t)$ は以下のように求

められる。

$$\begin{aligned} \frac{\partial g}{\partial p_0} &= q_0 \exp \left[-\frac{1}{q_1} \left\{ -p_0 p_1 \exp \left(-\frac{t}{p_1} \right) + (p_2 + 1) t \right\} \right] \\ &\quad \times \left[1 + \frac{p_1}{q_1} \left\{ p_0 \exp \left(-\frac{t}{p_1} \right) + p_2 + 1 \right\} \right] \exp \left(-\frac{t}{p_1} \right) \end{aligned} \quad (5.9)$$

$$\begin{aligned} \frac{\partial g}{\partial p_1} &= q_0 \exp \left[-\frac{1}{q_1} \left\{ -p_0 p_1 \exp \left(-\frac{t}{p_1} \right) + (p_2 + 1) t \right\} \right] \\ &\quad \times \frac{p_0}{p_1} \left[\frac{t}{p_1} + \frac{p_1 + t}{q_1} \left\{ p_0 \exp \left(-\frac{t}{p_1} \right) + p_2 + 1 \right\} \right] \exp \left(-\frac{t}{p_1} \right) \end{aligned} \quad (5.10)$$

$$\frac{\partial g}{\partial p_2} = q_0 \exp \left[-\frac{1}{q_1} \left\{ -p_0 p_1 \exp \left(-\frac{t}{p_1} \right) + (p_2 + 1) t \right\} \right] \left[1 - \frac{t}{q_1} \left\{ p_0 \exp \left(-\frac{t}{p_1} \right) + p_2 + 1 \right\} \right] \quad (5.11)$$

$$\delta g(t) = \sqrt{\sum_{i=0}^2 \left(\frac{\partial g}{\partial p_i} \delta p_i \right)^2} \quad (5.12)$$

最終的なフィッティング関数 $g(t)$ を、 $g(t) + \delta g(t)$ 、 $g(t) - \delta g(t)$ に置き換えてフィッティングした場合の寿命をそれぞれ $\tau_{+\delta g}$ 、 $\tau_{-\delta g}$ とし、それぞれの寿命を求めると次のようになった。

表 5.6: Pick-Off 補正関数の誤差を考慮した寿命値

	TQ 補正後 [ns]	PickOff 補正後 [ns]	誤差 [ns]
		$\tau_{+\delta g} = 231.9$	
NaI1	$\tau = 135.0$	$\tau = 208.6$	$\sigma = 23.3$
		$\tau_{-\delta g} = 191.1$	
		$\tau_{+\delta g} = 211.7$	
NaI3	$\tau = 108.6$	$\tau = 188.6$	$\sigma = 59.3$
		$\tau_{-\delta g} = 129.3$	

5.2 考察

5.2.1 寿命値とその誤差

最終的に求められた Pick-Off 補正後の寿命値は文献値とかけ離れたものとなった。さらに前節の評価から、Pick-Off 補正において特に大きな誤差が生じてしまっていると推察される。TQ 補正後に算出された寿命値は打倒であることも踏まえ、Pick-Off 補正に修正すべき点があると考えられる。以下で Pick-Off 補正の問題点・改善点を考える。

Pick-Off 補正の問題点

本実験で行った Pick-Off 補正における問題点として以下のものが考えられる。

1. Pick-Off 補正の理論・モデルの整合性。
2. プログラミングでの解析手法。

1. そもそも Pick-Off 補正で採用している仮定が本実験において不適切だった可能性が考えられる。また、Pick-Off 補正でフィッティング関数導出の理論において、不適切と思われる箇所が見つかった (付録 A2 参照)。
2. プログラミング上での解析の中で、大きな誤差が生じた可能性が考えられる。具体的には、511keV のイベント数や総イベント数の数え上げにおける、データ範囲指定の方法に不備があったと推察される。

改善方法

Pick-Off 補正を改善するには、まずは理論の見直し。その後、プログラミング上での、使用するデータ範囲・イベント数の数え上げ方・フィッティングでの初期値などを調整し、理論と整合する適切なプログラミングを作り上げる必要がある。本演習では、時間的な制約上、その改善まで行うことはできなかった。

他に、多数の NaI 検出器をポジトロニウムを取り囲むように配置し、3つの検出器が同時に反応した場合のみを取り出す、という方法も考えられる。こうすることで、初めから 3γ の崩壊のみをデータとして取り出し、Pick-Off 補正を行うことなく寿命値を算出できると思われる。こちらも、実験設備と時間の制約上、本演習で試行することはできなかった。

5.2.2 真空の有無による差異

本実験では、少しでも Pick-Off 反応を減らすため、シリカゲルを真空状態において実験を行った。その真空の有無の影響を比較した実験結果が図 5.1 である。縦軸の PickOff 補正関数の値は、Pick-Off 反応が起こる

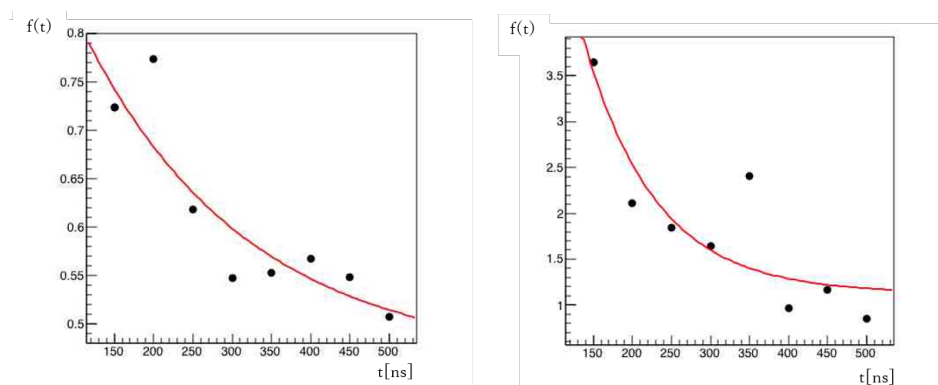


図 5.1: 真空の有無による PickOff 補正関数の違い。左が、真空状態 (本実験) での NaI3 における $f(t)$ フィッティング結果。右が真空状態にせずに測定した場合の、NaI3 における $f(t)$ のフィッティング結果。

頻度を表しており、この値が大きい程 Pick-Off 反応の割合が大きいことになる。図 5.1 のグラフからわかるように、真空状態にすることで Pick-Off 反応の割合が減っていることがわかる。

第 6 章

結論

改めて、本実験により得られた o- Ps の寿命値は

NaI1 : 208.6 [ns]

NaI3 : 188.5 [ns]

となった。文献値との比較、及び見積もった誤差の大きさから、決して精度の良い測定とは言えない結果となった。今後の演習で上で述べた事柄が改善されることを期待したい。

謝辞

今回実験を進めるにあたり、半年間丁寧に指導・助言してくださった木河達也さんに感謝いたします。また実験・解析にお付き合いいただき、手助けしていただいた TA の岡崎佑太さん、池満拓司さんに感謝いたします。

付録 A1 Pick-off 補正に用いたソースコードとその説明

Pickoff_analyze003.c の概説

Pick-off 補正に用いたソースコード Pickoff_analyze003.c を概説する。ソースコードは本章の後半に記載する。

Pickoff_analyze003.c では、

・S(t):Time=t[ns] における Δt の範囲にての(図 4.13 における)「511keV 未満の遅めの領域 + pick-off による 511keV」の領域における総イベント数、

・S(0)_alpha : Time=0[ns] における Δt の範囲にての「511keV 未満の遅めの領域」の領域における総イベント数として、

$$\Delta N_{2\gamma}(t) = \frac{z(t)}{z(0)} S(0)_alpha \quad (A1.1)$$

$$\Delta N_{3\gamma}(t) = S(t) - \frac{z(t)}{z(0)} S(0)$$

を用いて計算を行った。以下にこのS(0)_alphaと z(t)を説明する。

・S(0)_alpha とは：

S(0)_alphaをわざわざ用いているのは、各 t の S(t)はある同じ Energy 範囲にて一律に Cut する方法でイベント数を数え上げる方法を用いたが、

・511keV のピークより大きな Energy 値で Cut して求めた S(t) による $\Delta N_{3\gamma}(t)$ の値は(ヒストグラムを目視で確認して)「 $\Delta N_{3\gamma}(t)$ の値としてより相応しい値」と比べて誤差はより少なく見え、

・またそうならば「511keV 未満の遅めの領域」における Cut によるイベント数の抽出はS(0)_alpha 1つにして誤差の可能性をより減らさるよう思え、
たことからである。

(また Time=0[ns]では、 $0 \leq \text{Energy} < 450[\text{keV}]$ にて Cut した場合余分に数えたガウスピークの末端とヒストグラムの数え損ねた分の相殺は幾らかはあるだろう(図 A1.1 参照)。丁度相殺しそうなところに範囲を設定すべきだったかもしれない)

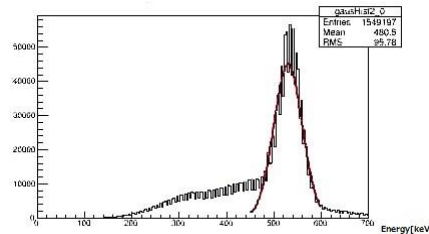


図 A1.1: NaI3 の $0 - 3 \leq \text{Time} < 0 + 3$ における 0 から 700[keV]までの Energy ヒストグラム

• $z(t)$ とは：

図 A1.2 のように考えると分かりやすいだろう。

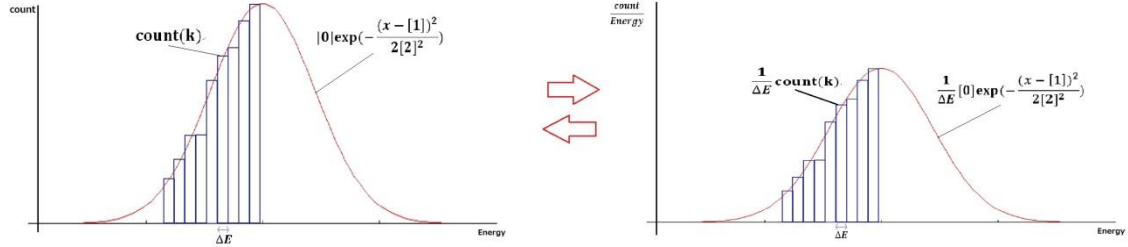


図 A1.2: $z(t)$ のイメージ図

すなわち、

- Energy の 1 次ヒストグラムにおいて、bin 幅を ΔE とすると各 k 番目の $\text{count}(k)$ を ΔE で割った値は $\frac{\text{count}(k)}{\Delta E}$ となり、
- またヒストグラムを "gaus" で fitting した場合はその関数は $[0]\exp(-\frac{(x-[1])^2}{2[2]^2})$ であり、 ΔE で割った値は $\frac{1}{\Delta E} [0]\exp(-\frac{(x-[1])^2}{2[2]^2})$ となる。(もとより $\text{count}(k) = \Delta N(\text{Energy})$ のようなものである)

右側の図における棒グラフの面積を求めることはすなわち左側の図の count を数え上げていくことであり、よって右側の図における棒グラフの面積を関数 $\frac{1}{\Delta E} [0]\exp(-\frac{(x-[1])^2}{2[2]^2})$ の面積で近似するならば、ガウス関数の面積公式

$$\int_{-\infty}^{\infty} \exp(-ax^2) dx = \sqrt{\frac{\pi}{a}} \quad (\text{A1.2})$$

より、同じ bin 幅 (ΔE) で区切られたヒストグラムならば、2 つのガウスピークの面積比は、

$$\frac{\int_{-\infty}^{\infty} \frac{1}{\Delta E} [0]\exp(-\frac{(\text{Energy} - [1])^2}{2[2]^2}) dE}{\int_{-\infty}^{\infty} \frac{1}{\Delta E} [0']\exp(-\frac{(\text{Energy} - [1'])^2}{2[2']^2}) dE} = \frac{\frac{1}{\Delta E} [0][2]\sqrt{\frac{\pi}{2}}}{\frac{1}{\Delta E} [0'] [2']\sqrt{\frac{\pi}{2}}} = \frac{[0][2]}{[0'] [2']} \quad (\text{A1.3})$$

となるので、

$$\frac{y(t)}{y(0)} \cong \frac{[0][2]}{[0'] [2']} \quad (\text{A1.4})$$

から、

$$z(t) = [0] \times [2] \quad (\text{A1.5})$$

として

$$\frac{y(t)}{y(0)} \cong \frac{z(t)}{z(0)} \quad (\text{A1.6})$$

としていた。

この方法は、ある Energy の範囲で Cut してイベント数を数えた場合にはガウスピークでない部分をも数え上げている事も考えられるが、それを防ぐことが出来るだろうと思われる。

続いて Pickoff_analyze003.c を記す。

```

1  //#define-----
2  #define NTUPLE0_INFILENAME "ntuple2"    //読み込む root ファイルの ntuple、Energy・time ブランチの名前
3  #define NTUPLE1_INFILENAME "ntuple3"
4  #define NTUPLE2_INFILENAME "ntuple4"
5  #define ENERGY0_INBRANCHNAME "a11"
6  #define ENERGY1_INBRANCHNAME "a22"
7  #define ENERGY2_INBRANCHNAME "a33"
8  #define TIME0_INBRANCHNAME "t11"
9  #define TIME1_INBRANCHNAME "t21"
10 #define TIME2_INBRANCHNAME "t31"
11 //-----
12 #define HIST_ENERGY_LOWEDGE 0           //Histogram に詰め込むときの範囲指定
13 #define HIST_ENERGY_UPPEREDGE 700      //←は S(t)の決定に直に関わる
14 #define HIST_TIME_LOWEDGE 0
15 #define HIST_TIME_UPPEREDGE 1000
16
17 #define ALPHA_HIST_ENERGY_UPPEREDGE 450 //←は S(t)_alpha の決定に直に関わる
18 //-----
19 #define POINT_NUM 30 //POINT_NUM >= (TIME_END-TIME_START)/TIME_INTERVAL+2 : plot する f(t)点数
20
21 #define DELTA_T 3 //f(t)の S(t)等を決める際の t からの幅 (t±DELTA_T)
22 #define TIME_T0 0 //p-Ps(pickoff でない)が占めるという Time
23 #define TIME_START 150 //f(t)の plot を開始する点と終える点
24 #define TIME_END 500
25 #define TIME_INTERVAL 50 //f(t)の plot の間隔
26
27 #define LEFT0_511KEV 450 //各 t±DELTA_T で抽出した histogram における 511keV ガウスの(fitting する)範囲
28 #define RIGHT0_511KEV 550
29 #define LEFT1_511KEV 450
30 #define RIGHT1_511KEV 550
31 #define LEFT2_511KEV 450
32 #define RIGHT2_511KEV 600
33 //-----
34 #define INITIALPARAMETER_P0 0.8 //p0p1p2、q0q1q2 の初期値
35 #define INITIALPARAMETER_P1 815
36 #define INITIALPARAMETER_P2 0.2

```

```

37
38 #define INITIALPARAMETER_Q0 4500
39 #define INITIALPARAMETER_Q1 155
40 #define INITIALPARAMETER_Q2 4460
41 //-----
42 #define PQFIT_LEFT 150 //should be = TIME_START : 寿命決定に使う Time の Histogram への fitting 範囲
43 #define PQFIT_RIGHT 500 //should be = TIME_END
44
45 #define PQHISTUPPEREDGE 450 //寿命決定に使う Time の Histogram の抽出範囲
46 //-----
47 #define CANDIVRATE_LEFT 4 //used like CT>Divide(CANDIVRATE_LEFT,CANDIVRATE_RIGHT)
48 #define CANDIVRATE_RIGHT 3
49
50 //function prototype-----
51 //function prototype-----
52 void Draw_option(TH1 aaa);
53 void Load_ntuple(TFile* File,TNtuple* Ntuple_Pickoff0,TNtuple* Ntuple_Pickoff1,TNtuple* Ntuple_Pickoff2);
54 void Load_ntuple_older(TFile* File,TNtuple* Ntuple_Pickoff0,TNtuple* Ntuple_Pickoff1,TNtuple*
55 Ntuple_Pickoff2);
56
57 //Pickoff_analyze003-----
58 //Pickoff_analyze003-----
59
60 void Pickoff_analyze003(TString input_file){
61
62 //load rootfile -----
63 TFile *file = TFile::Open(input_file);
64 TNtuple* ntuple_pickoff0 = (TNtuple *)file -> Get(NTUPLE0_INFILENAME);
65 TNtuple* ntuple_pickoff1 = (TNtuple *)file -> Get(NTUPLE1_INFILENAME);
66 TNtuple* ntuple_pickoff2 = (TNtuple *)file -> Get(NTUPLE2_INFILENAME);
67
68 // S(t) y(t)_settijngs -----
69 TH1F **gausHist0 = new TH1F*[POINT_NUM];
70 TH1F **gausHist1 = new TH1F*[POINT_NUM];
71 TH1F **gausHist2 = new TH1F*[POINT_NUM];
72 TF1 **gausfit0 = new TF1*[POINT_NUM];

```

```

73 TF1 **gausfit1 = new TF1*[POINT_NUM];
74 TF1 **gausfit2 = new TF1*[POINT_NUM];
75
76 TH1F* alphaHist = new
77 TH1F("Alpha","alpha",200,HIST_ENERGY_LOWEDGE,ALPHA_HIST_ENERGY_UPPEREDGE);
78
79 TCanvas* CT0 = new TCanvas("gausfit0","gausfit0",1800,1200);
80 CT0 -> Divide(CANDIVRATE_LEFT,CANDIVRATE_RIGHT);
81 TCanvas* CT1 = new TCanvas("gausfit1","gausfit1",1800,1200);
82 CT1 -> Divide(CANDIVRATE_LEFT,CANDIVRATE_RIGHT);
83 TCanvas* CT2 = new TCanvas("gausfit2","gausfit2",1800,1200);
84 CT2 -> Divide(CANDIVRATE_LEFT,CANDIVRATE_RIGHT);
85
86
87 float y_t[3][POINT_NUM] , S_t[3][POINT_NUM] , f_t[3][POINT_NUM] , t[POINT_NUM] , z_t[3][POINT_NUM];
88 // S(t) z(t)_calc -----
89 int i , cur_start = TIME_T0 ;
90
91 //for loop-----
92 for(i=0 ; cur_start <= TIME_END ; i++){
93 //settings (i)-----
94 TCut t_cut0 =Form("%s >= %d && %s < %d
95 ",TIME0_INBRANCHNAME,cur_start-DELTA_T,TIME0_INBRANCHNAME,cur_start+DELTA_T);
96 TCut t_cut1 =Form("%s >= %d && %s < %d
97 ",TIME1_INBRANCHNAME,cur_start-DELTA_T,TIME1_INBRANCHNAME,cur_start+DELTA_T);
98 TCut t_cut2 =Form("%s >= %d && %s < %d
99 ",TIME2_INBRANCHNAME,cur_start-DELTA_T,TIME2_INBRANCHNAME,cur_start+DELTA_T);
100
101 gausHist0[i] = new TH1F(Form("gausHist0_%d",i),Form("%d - %d <= Time0 <= %d
102 + %d",cur_start,DELTA_T,cur_start,DELTA_T),200,HIST_ENERGY_LOWEDGE,HIST_ENERGY_UPPEREDGE);
103 gausHist1[i] = new TH1F(Form("gausHist1_%d",i),Form("%d - %d <= Time1 <= %d
104 + %d",cur_start,DELTA_T,cur_start,DELTA_T),200,HIST_ENERGY_LOWEDGE,HIST_ENERGY_UPPEREDGE);
105 gausHist2[i] = new TH1F(Form("gausHist2_%d",i),Form("%d - %d <= Time2 <= %d
106 + %d",cur_start,DELTA_T,cur_start,DELTA_T),200,HIST_ENERGY_LOWEDGE,HIST_ENERGY_UPPEREDGE);
107 gausfit0[i] = new TF1(Form("gausfit0_%d",i) , "gaus");
108 gausfit1[i] = new TF1(Form("gausfit1_%d",i) , "gaus");

```



```

109     gausfit2[i] = new TF1(Form("gausfit2_%d",i) , "gaus");
110
111 //Fit and Draw (i)-----
112     CT0->cd(i+1);
113     ntuple_pickoff0 -> Draw(Form("%s >> gausHist0_%d",ENERGY0_INBRANCHNAME,i),t_cut0);
114     gausHist0[i] -> Fit(Form("gausfit0_%d",i) , "" , "" ,LEFT0_511KEV,RIGHT0_511KEV);
115     CT1->cd(i+1);
116     ntuple_pickoff1 -> Draw(Form("%s >> gausHist1_%d",ENERGY1_INBRANCHNAME,i),t_cut1);
117     gausHist1[i] -> Fit(Form("gausfit1_%d",i) , "" , "" ,LEFT1_511KEV,RIGHT1_511KEV);
118     CT2->cd(i+1);
119     ntuple_pickoff2 -> Draw(Form("%s >> gausHist2_%d",ENERGY2_INBRANCHNAME,i),t_cut2);
120     gausHist2[i] -> Fit(Form("gausfit2_%d",i) , "" , "" ,LEFT2_511KEV,RIGHT2_511KEV);
121
122 //input S_t y_t t f_t (i)-----
123     S_t[0][i] = gausHist0[i] -> GetEntries();
124     S_t[1][i] = gausHist1[i] -> GetEntries();
125     S_t[2][i] = gausHist2[i] -> GetEntries();
126
127     /* y_t[0][i] = gausfit0[i] -> GetParameter(0);
128     y_t[1][i] = gausfit1[i] -> GetParameter(0);
129     y_t[2][i] = gausfit2[i] -> GetParameter(0);*/
130
131     z_t[0][i] = gausfit0[i] -> GetParameter(0) * gausfit0[i] -> GetParameter(2);
132     z_t[1][i] = gausfit1[i] -> GetParameter(0) * gausfit1[i] -> GetParameter(2);
133     z_t[2][i] = gausfit2[i] -> GetParameter(0) * gausfit2[i] -> GetParameter(2);
134
135
136     t[i] = cur_start ;
137     cur_start += TIME_INTERVAL;
138
139     if(i == 0) {
140         cur_start = TIME_START;
141
142         float S_t0alpha0, St0_alpha1, S_t0alpha2 ;
143         S_t0alpha0 = ntuple_pickoff0 -> GetEntries(t_cut0 && Form("%s>=%d
144 && %s<=%d",ENERGY0_INBRANCHNAME,HIST_ENERGY_LOWEDGE,ENERGY0_INBRANCHNAME,ALPH

```

```

145  A_HIST_ENERGY_UPPEREDGE) ;
146      S_t0alpha1      =      ntuple_pickoff1      ->      GetEntries(t_cut1      &&      Form("%s>=%d
147  && %s<=%d",ENERGY1_INBRANCHNAME,HIST_ENERGY_LOWEDGE,ENERGY1_INBRANCHNAME,ALPH
148  A_HIST_ENERGY_UPPEREDGE) ) ;
149      S_t0alpha2      =      ntuple_pickoff2      ->      GetEntries(t_cut2      &&      Form("%s>=%d
150  && %s<=%d",ENERGY2_INBRANCHNAME,HIST_ENERGY_LOWEDGE,ENERGY2_INBRANCHNAME,ALPH
151  A_HIST_ENERGY_UPPEREDGE) ) ;
152
153  }
154  else {
155      //f_t0[i] = (y_t0[i]*y_t0[i]*S_t0[0])/(y_t0[0]*y_t0[0]*S_t0[i]-y_t0[i]*y_t0[i]*S_t0[0]);
156      //f_t1[i] = (y_t1[i]*y_t1[i]*S_t1[0])/(y_t1[0]*y_t1[0]*S_t1[i]-y_t1[i]*y_t1[i]*S_t1[0]);
157      //f_t2[i] = (y_t2[i]*y_t2[i]*S_t2[0])/(y_t2[0]*y_t2[0]*S_t2[i]-y_t2[i]*y_t2[i]*S_t2[0]);
158
159      f_t0[i] = (z_t0[i]*S_t0alpha0)/(z_t0[0]*S_t0[i]-z_t0[i]*S_t0[0]);
160      f_t1[i] = (z_t1[i]*S_t0alpha1)/(z_t1[0]*S_t1[i]-z_t1[i]*S_t1[0]);
161      f_t2[i] = (z_t2[i]*S_t0alpha2)/(z_t2[0]*S_t2[i]-z_t2[i]*S_t2[0]);
162
163  }
164  CT0 -> Update0 ; CT1 -> Update0 ; CT2 -> Update0 ;
165  }
166
167  //adjust var or fill empty var -----
168  t[0] = t[1];
169  f_t0[0] = f_t0[1]; f_t1[0] = f_t1[1]; f_t2[0] = f_t2[1];
170
171  //Fit_p0p1p2-----
172  //settings-----
173  TGraph* tg0 = new TGraph(i,t,f_t[0]);
174  TGraph* tg1 = new TGraph(i,t,f_t[1]);
175  TGraph* tg2 = new TGraph(i,t,f_t[2]);
176
177  TF1* fit0_p0p1p2 = new TF1("fit0_p0p1p2","[0]*exp(-x/[1]+[2]);
178
179  fit0_p0p1p2->SetParameters(INITIALPARAMETER_P0,INITIALPARAMETER_P1,INITIALPARAMETER_P2);
180  fit0_p0p1p2->SetParNames("p0","p1","p2");

```

```

181     TF1* fit1_p0p1p2 = new TF1("fit1_p0p1p2","[0]*exp(-x/[1])+[2]");
182
183 fit1_p0p1p2->SetParameters(INITIALPARAMETER_P0,INITIALPARAMETER_P1,INITIALPARAMETER_P2);
184     fit1_p0p1p2->SetParNames("p0","p1","p2");
185     TF1* fit2_p0p1p2 = new TF1("fit2_p0p1p2","[0]*exp(-x/[1])+[2]");
186
187 fit2_p0p1p2->SetParameters(INITIALPARAMETER_P0,INITIALPARAMETER_P1,INITIALPARAMETER_P2);
188     fit2_p0p1p2->SetParNames("p0","p1","p2");
189
190 //fit and draw-----
191     tg0 -> Fit("fit0_p0p1p2");
192     tg1 -> Fit("fit1_p0p1p2");
193     tg2 -> Fit("fit2_p0p1p2");
194
195     tg0->SetMarkerStyle(8); tg1->SetMarkerStyle(8); tg2->SetMarkerStyle(8);//to Draw_Option
196
197     TCanvas *CTP = new TCanvas("P0P1P2","P0P1P2",1500,500);
198     CTP->Divide(3,1);
199
200     CTP->cd(1);  tg0 -> Draw("AP");
201     CTP->cd(2);  tg1 -> Draw("AP");
202     CTP->cd(3);  tg2 -> Draw("AP");
203
204     CTP -> Update() ;
205
206 //Fit_p0p1p2q0q1q2-----
207 //settings1-----
208     float p[3][3];
209     p[0][0] = fit0_p0p1p2 -> GetParameter(0);
210     p[0][1] = fit0_p0p1p2 -> GetParameter(1);
211     p[0][2] = fit0_p0p1p2 -> GetParameter(2);
212     p[1][0] = fit1_p0p1p2 -> GetParameter(0);
213     p[1][1] = fit1_p0p1p2 -> GetParameter(1);
214     p[1][2] = fit1_p0p1p2 -> GetParameter(2);
215     p[2][0] = fit2_p0p1p2 -> GetParameter(0);
216     p[2][1] = fit2_p0p1p2 -> GetParameter(1);

```

```

217     p[2][2] = fit2_p0p1p2 -> GetParameter(2);
218
219 //write formula p0p1p2q0q1q2-----
220     char*                               Formula0_p0p1p2q0q1q2           =
221 Form("[0]*(%f*exp(-x/%f)+%f+1)*exp(-1/[1]*(-%f*%f*exp(-x/%f)+(%f+1)*x))+[2]",p[0][0],p[0][1],p[0][2],p[0][0],p[0][1],
222 p[0][1],p[0][2]);
223     char*                               Formula1_p0p1p2q0q1q2           =
224 Form("[0]*(%f*exp(-x/%f)+%f+1)*exp(-1/[1]*(-%f*%f*exp(-x/%f)+(%f+1)*x))+[2]",p[1][0],p[1][1],p[1][2],p[1][0],p[1][1],
225 p[1][1],p[1][2]);
226     char*                               Formula2_p0p1p2q0q1q2           =
227 Form("[0]*(%f*exp(-x/%f)+%f+1)*exp(-1/[1]*(-%f*%f*exp(-x/%f)+(%f+1)*x))+[2]",p[2][0],p[2][1],p[2][2],p[2][0],p[2][1],
228 p[2][1],p[2][2]);
229
230
231 //settings2-----
232     TF1* fit0_p0p1p2q0q1q2 = new TF1("fit0_p0p1p2q0q1q2",Formula0_p0p1p2q0q1q2);
233
234 fit0_p0p1p2q0q1q2->SetParameters(INITIALPARAMETER_Q0,INITIALPARAMETER_Q1,INITIALPARAMETER
235 _Q2);
236     fit0_p0p1p2q0q1q2->SetParNames("q0","q1","q2");
237     TF1* fit1_p0p1p2q0q1q2 = new TF1("fit1_p0p1p2q0q1q2",Formula1_p0p1p2q0q1q2);
238
239 fit1_p0p1p2q0q1q2->SetParameters(INITIALPARAMETER_Q0,INITIALPARAMETER_Q1,INITIALPARAMETER
240 _Q2);
241     fit1_p0p1p2q0q1q2->SetParNames("q0","q1","q2");
242     TF1* fit2_p0p1p2q0q1q2 = new TF1("fit2_p0p1p2q0q1q2",Formula2_p0p1p2q0q1q2);
243
244 fit2_p0p1p2q0q1q2->SetParameters(INITIALPARAMETER_Q0,INITIALPARAMETER_Q1,INITIALPARAMETER
245 _Q2);
246     fit2_p0p1p2q0q1q2->SetParNames("q0","q1","q2");
247
248 //fit and draw-----
249     TCanvas *CTQ = new TCanvas("Nameeee","P0P1P2Q0Q1Q2",900,600);
250     CTQ->Divide(3,1);
251
252     TH1F* PQhist0 = new TH1F("PQhist0","PQhist0",500,HIST_TIME_LOWEDGE,HIST_TIME_UPPEREDGE);

```

```

253 TH1F* PQhist1 = new TH1F("PQhist1","PQhist1",500,HIST_TIME_LOWEDGE,HIST_TIME_UPPEREDGE);
254 TH1F* PQhist2 = new TH1F("PQhist2","PQhist2",500,HIST_TIME_LOWEDGE,HIST_TIME_UPPEREDGE);
255
256 CTQ->cd(1); gPad->SetLogy0;
257 ntuple_pickoff0 -> Draw(Form("%s >> PQhist0",TIME0_INBRANCHNAME),Form("%s <= %d
258 ",ENERGY0_INBRANCHNAME,PQHISTUPPEREDGE));
259 PQhist0 -> Fit("fit0_p0p1p2q0q1q2","","PQFIT_LEFT,PQFIT_RIGHT);
260 CTQ->cd(2); gPad->SetLogy0;
261 ntuple_pickoff1 -> Draw(Form("%s >> PQhist1",TIME1_INBRANCHNAME),Form("%s <= %d
262 ",ENERGY1_INBRANCHNAME,PQHISTUPPEREDGE));
263 PQhist1 -> Fit("fit1_p0p1p2q0q1q2","","PQFIT_LEFT,PQFIT_RIGHT);
264 CTQ->cd(3); gPad->SetLogy0;
265 ntuple_pickoff2 -> Draw(Form("%s >> PQhist2",TIME2_INBRANCHNAME),Form("%s <= %d
266 ",ENERGY2_INBRANCHNAME,PQHISTUPPEREDGE));
267 PQhist2 -> Fit("fit2_p0p1p2q0q1q2","","PQFIT_LEFT,PQFIT_RIGHT);
268
269 CTQ -> Update();
270
271 //-output result-----
272 ntuple_pickoff0->Print();
273 int nn; for(nn=0;nn<i;nn++){
274 printf("t[%d]=%f S_t[0][%d]=%f z_t[0][%d]=%f
275 f_t[0][%d]=%f\n",nn,t[nn],nn,S_t[0][nn],nn,z_t[0][nn],nn,f_t[0][nn] ); }
276
277 }
278
279 //-----
280 //-functions-----
281 //-functions-----
282 void Draw_option(TH1 aaa){
283 // gStyle->SetOptFit0; gStyle->SetOptFit(1111);
284
285 }
286 //-----
287
288 /*void Load_ntuple(TFile* File,TNtuple* Ntuple_Pickoff0,TNtuple* Ntuple_Pickoff1,TNtuple* Ntuple_Pickoff2){

```

```

289
290 TTreeReader ntuple0_name(NTUPLE0_INFILENAME,File);
291 TTreeReaderValue<Float_t> energy0_name(ntuple0_name,ENERGY0_INBRANCHNAME);
292 TTreeReaderValue<Float_t> time0_name(ntuple0_name,TIME0_INBRANCHNAME);
293 TTreeReader ntuple1_name(NTUPLE1_INFILENAME,File);
294 TTreeReaderValue<Float_t> energy1_name(ntuple1_name,ENERGY1_INBRANCHNAME);
295 TTreeReaderValue<Float_t> time1_name(ntuple1_name,TIME1_INBRANCHNAME);
296 TTreeReader ntuple2_name(NTUPLE2_INFILENAME,File);
297 TTreeReaderValue<Float_t> energy2_name(ntuple2_name,ENERGY2_INBRANCHNAME);
298 TTreeReaderValue<Float_t> time2_name(ntuple2_name,TIME2_INBRANCHNAME);
299
300 while(ntuple0_name.Next())
301     Ntuple_Pickoff0 -> Fill(*energy0_name , *time0_name);
302 while(ntuple1_name.Next())
303     Ntuple_Pickoff1 -> Fill(*energy1_name , *time1_name);
304 while(ntuple2_name.Next())
305     Ntuple_Pickoff2 -> Fill(*energy2_name , *time2_name);
306
307     */
308
309 void Load_ntuple_older(TFile* File,TNtuple* Ntuple_Pickoff0,TNtuple* Ntuple_Pickoff1,TNtuple*
310 Ntuple_Pickoff2){
311     int i=0; float e[3],t[3];
312
313     TNtuple* Ntuple_Pickoff0_dummy = (TNtuple *)File->Get(NTUPLE0_INFILENAME);
314     Ntuple_Pickoff0_dummy -> SetBranchAddress(ENERGY0_INBRANCHNAME,&e[0]);
315     Ntuple_Pickoff0_dummy -> SetBranchAddress(TIME0_INBRANCHNAME,&t[0]);
316     for(i=0;Ntuple_Pickoff0_dummy->GetEntry(i)>0;i++)
317         Ntuple_Pickoff0 -> Fill(e[0],t[0]);
318
319     TNtuple* Ntuple_Pickoff1_dummy= (TNtuple *)File->Get(NTUPLE1_INFILENAME);
320     Ntuple_Pickoff1_dummy -> SetBranchAddress(ENERGY1_INBRANCHNAME,&e[1]);
321     Ntuple_Pickoff1_dummy -> SetBranchAddress(TIME1_INBRANCHNAME,&t[1]);
322     for(i=0; Ntuple_Pickoff1_dummy->GetEntry(i)>0;i++)
323         Ntuple_Pickoff1-> Fill(e[1],t[1]);
324

```

```

325 TNtuple* Ntuple_Pickoff2_mummy= (TNtuple *)File->Get(NTUPLE2_INFILENAME);
326 Ntuple_Pickoff2_mummy -> SetBranchAddress(ENERGY2_INBRANCHNAME,&e[2]);
327 Ntuple_Pickoff2_mummy -> SetBranchAddress(TIME2_INBRANCHNAME,&t[2]);
328 for(i=0;Ntuple_Pickoff2_mummy->GetEntry(i)>0;i++)
329     Ntuple_Pickoff2-> Fill(e[2],t[2]);
330
331 }

```

Pickoff_analyze003.c

— : $f(t)$ の算出 — : $f(t)$ の fitting — : 崩壊寿命の算出 — : 不使用

4.5 章の解析で用いられたソースコードは上記の行 121 から 130 までを以下で取り換え、

```

int minn=gausHist0[i]->FindBin(0);
int maxx=gausHist0[i]->FindBin(450);

S_t[0][i]=0;
for(int j=minn;j<=maxx;j++){
    S_t[0][i]= S_t[0][i] + gausHist0[i] -> GetBinContent(j);
}
S_t[1][i]=0;
for(int k=minn;k<=maxx;k++){
    S_t[1][i]= S_t[1][i] +gausHist1[i]->GetBinContent(k);
}
S_t[2][i]=0;
for(int l=minn;l<=maxx;l++){
    S_t[2][i]=S_t[2][i]+gausHist2[i]->GetBinContent(l);
}

//input S_t y_t t f_t (i)-----
//S_t[0][i] = gausHist0[i] -> GetEntries();
//    S_t[1][i] = gausHist1[i] -> GetEntries();
//S_t[2][i] = gausHist2[i] -> GetEntries();

```

また行 159 から 161 を以下で取り換えたものである。

```
f_t[0][i] = (y_t[0][i]*S_t[0][0])/(y_t[0][0]*S_t[0][i]-y_t[0][i]*S_t[0][0]);
```

$$f_t[1][i] = (y_t[1][i]*S_t[1][0]) / (y_t[1][0]*S_t[1][i] - y_t[1][i]*S_t[1][0]);$$

$$f_t[2][i] = (y_t[2][i]*S_t[2][0]) / (y_t[2][0]*S_t[2][i] - y_t[2][i]*S_t[2][0]);$$

(#define にある p0p1p2 の初期値等は適宜変えてある)

・ちなみに行 142 から 151 を上記のような

```
(for(int j~) S_t0alpha0+= gausHist0[0] -> GetBinContent(j);
```

とした方が処理は早いと思われる)

下図 A1.3 は上のソースコードを走らせると表示されるものであり、これを見て 511keV ピークへの fitting が上手く行っているかを確認する

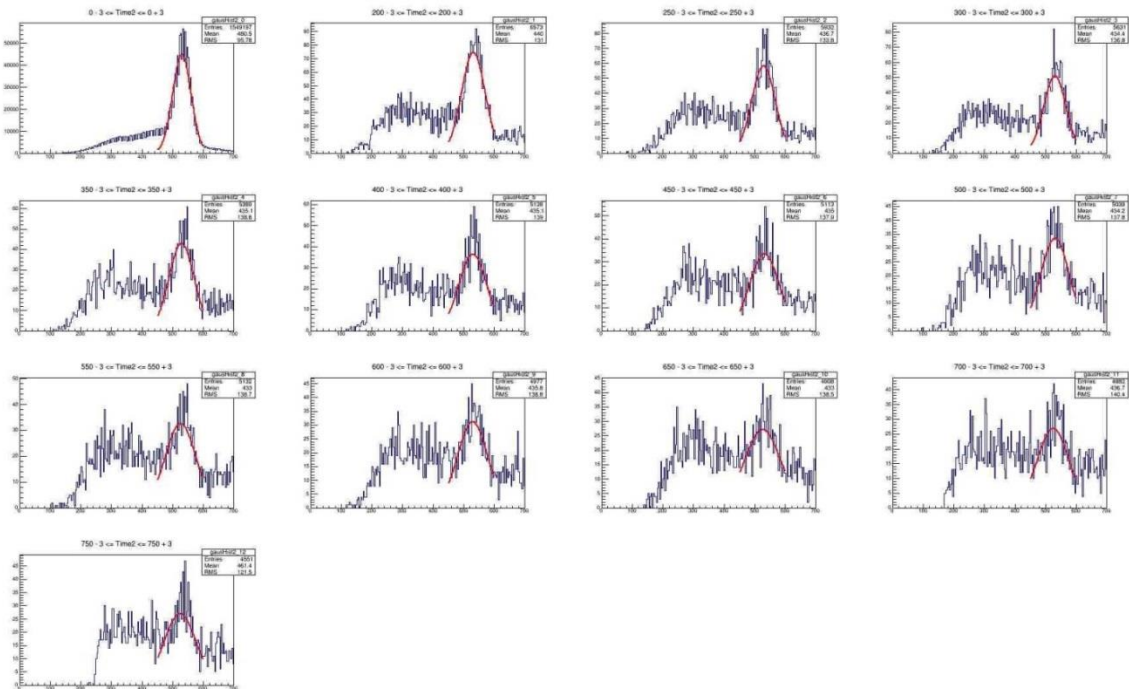


図 A1.3: NaI3 の各 t にての Energy ヒストグラム

付録 A2 Pick-off 補正理論の検証

A2.1 間違っただけから算出された寿命?

4.5 章での手順はまるで、例えば「 $f(t)$ の $\Delta N_{2\gamma}$ は 511keV 領域(図 4.13 参照)をも含める」と実際はしつつ、 $\Delta N_{2\gamma}$ の決定や最後の fitting では「511keV 未満の(遅めの)領域」にて処理するという混乱したことを行っているかのようである。

以下、こうあるべきだっただけではないかと思われるものを、もう少し丁寧に説明してみる。実際の測定されたデータたる「観測総数 $N_{3\gamma}(t)$ 」等を用いて議論を進める。

- Time=t での o-Ps の総数を $N_{o-Ps}(t)$
- Time=t での o-Ps の崩壊による 3γ 線の観測総数を $N_{3\gamma}(t)$

$$\begin{aligned}\frac{dN_{o-Ps}}{dt} &= -(\Gamma_{3\gamma} + \Gamma_{pickoff})N_{o-Ps} \\ \frac{dN_{3\gamma}}{dt} &= \Gamma_{3\gamma}N_{o-Ps}\end{aligned}\tag{A2.1}$$

この式はこの実験データにて成り立つものであるとする。また、

- Time=t でのコンプトン散乱した 2γ 線の観測総数を $N_{2\gamma_{under511}}(t)$ として
- $N_{under511}(t) \equiv N_{2\gamma_{under511}}(t) + N_{3\gamma}(t)$ とすると

$$\begin{aligned}\frac{dN_{under511}}{dt} &= \frac{d(N_{2\gamma_{under511}} + N_{3\gamma})}{dt} = \frac{dN_{2\gamma_{under511}}}{dt} + \frac{dN_{3\gamma}}{dt} \\ &= \frac{dN_{3\gamma}}{dt} \left(\frac{\frac{dN_{2\gamma_{under511}}}{dt}}{\frac{dN_{3\gamma}}{dt}} + 1 \right) \\ &= \Gamma_{3\gamma} \left(\frac{\frac{dN_{2\gamma_{under511}}}{dt}}{\frac{dN_{3\gamma}}{dt}} + 1 \right) N_{o-Ps}\end{aligned}\tag{A2.2}$$

ここで、

$$\frac{\frac{dN_{2\gamma_{under511}}}{dt}}{\frac{dN_{3\gamma}}{dt}} = \frac{\frac{dN_{2\gamma_{under511}}}{dt} \times \Delta t}{\frac{dN_{3\gamma}}{dt} \times \Delta t} = \frac{\Delta N_{2\gamma_{under511}}}{\Delta N_{3\gamma}}\tag{A2.3}$$

からもこれが 4.5 章で取り決めた $f(t)$ であるとわかる。

よって実際にするべきだったことは例えば、 $(\Gamma' = \Gamma_{3\gamma} + \Gamma_{pickoff})$ とすれば)

(I-i) : $\frac{dN_{o-Ps}(t)}{dt} = -\Gamma' N_{o-Ps}(t)$ より、 $N_{o-Ps}(t) = N_0 \exp(-\Gamma' \times t)$ であるから、

$$\Delta N_{under511}(t) = \frac{dN_{under511}(t)}{dt} \Delta t = \frac{N_0 \Delta t}{\tau_{3\gamma}} (f(t) + 1) \exp(-\Gamma' \times t) \quad (A2.4)$$

の式を用いて $\Delta N_{under511}(t)$ を「under511」たる抽出した「511keV 未満の(遅めの)領域」の Time ヒストグラムに fitting して上手くやるか、

(I-ii) : $\frac{dN_{o-Ps}}{dt}$ を $f(t)$ と同じくして、 $g(t) \equiv \frac{\Gamma_{pickoff}}{\Gamma_{3\gamma}}$ として

$$\frac{dN_{o-Ps}(t)}{dt} = -(\Gamma_{3\gamma} + \Gamma_{pickoff})N_{o-Ps}(t) = -\Gamma_{3\gamma}(1 + g(t))N_{o-Ps}(t) \quad (A2.5)$$

の式より、

$$N_{o-Ps}(t) = N_0 \exp\left(-\int_{t_0}^t \frac{1 + g(t')}{\tau_{3\gamma}} dt'\right) \quad (A2.6)$$

から、

$$\Delta N_{under511}(t) = \frac{N_0 \Delta t}{\tau_{3\gamma}} (f(t) + 1) \exp\left(-\int_{t_0}^t \frac{1 + g(t')}{\tau_{3\gamma}} dt'\right) \quad (A2.7)$$

に fitting することか、

(もし例えば

- Time=t での p-Ps の総数を $N_{p-Ps}(t)$
- Time=t での 2γ 線の観測総数を $N_{2\gamma}(t)$

として、ある Time 以上に存在する p-Ps は全て pickoff 反応によるものとみなせるとして、pick-off 反応による p-Ps は即座に崩壊して 2γ 線となり

$$\frac{dN_{2\gamma}}{dt} = \frac{dN_{p-Ps}(t)}{dt} = \Gamma_{pickoff} N_{o-Ps} \quad (A2.8)$$

とできるとするならば、

$$g(t) = \frac{\Gamma_{pickoff}}{\Gamma_{3\gamma}} = \frac{\Gamma_{pickoff} N_{o-Ps}(t)}{\Gamma_{3\gamma} N_{o-Ps}(t)} = \frac{\frac{dN_{2\gamma}(t)}{dt}}{\frac{dN_{3\gamma}(t)}{dt}} = \frac{\frac{dN_{2\gamma}(t)}{dt} \Delta t}{\frac{dN_{3\gamma}(t)}{dt} \Delta t} = \frac{\Delta N_{2\gamma}(t)}{\Delta N_{3\gamma}(t)} \quad (A2.9)$$

を考えて、今度は「511keV 未満の(遅めの)領域 + Pick-Off による 511keV」領域(図 4.13 参照)における 2γ 線の数をを以て $\Delta N_{2\gamma}(t)$ を算出すればよいのかもしれない。ただし、図 4.12 を見るに pick-off 化した p-Ps は即座に崩壊するとしていいのだろうかとは思う。)

(II)あるいは、 $N(t) \equiv N_{2\gamma}(t) + N_{3\gamma}(t)$ として、

$$\begin{aligned} \frac{dN(t)}{dt} &= \frac{d(N_{2\gamma}(t) + N_{3\gamma}(t))}{dt} = \frac{dN_{2\gamma}(t)}{dt} + \frac{dN_{3\gamma}(t)}{dt} \\ &= \frac{dN_{3\gamma}}{dt} \left(\frac{\frac{dN_{2\gamma}(t)}{dt}}{\frac{dN_{3\gamma}}{dt}} + 1 \right) = \Gamma_{3\gamma} \left(\frac{\frac{dN_{2\gamma}(t)}{dt}}{\frac{dN_{3\gamma}}{dt}} + 1 \right) N_{o-Ps} \end{aligned} \quad (A2.10)$$

$$h(t) = \frac{\frac{dN_{2\gamma}(t)}{dt}}{\frac{dN_{3\gamma}}{dt}} = \frac{\frac{dN_{2\gamma}(t)}{dt} \Delta t}{\frac{dN_{3\gamma}}{dt} \Delta t} = \frac{\Delta N_{2\gamma}(t)}{\Delta N_{3\gamma}(t)} \quad (A2.11)$$

とし、 $\frac{dN_{o-Ps}(t)}{dt} = -\Gamma_{3\gamma} \left(1 + \frac{\Gamma_{pickoff}}{\Gamma_{3\gamma}} \right) N_{o-Ps}(t)$ のようにして、 $\frac{dN_{2\gamma}}{dt} = \frac{dN_{pickoff}}{dt} = \Gamma_{pickoff} N_{o-Ps}$

として、

$$\Delta N(t) = \frac{N_0 \Delta t}{\tau_{3\gamma}} (h(t) + 1) \exp \left(- \int_{t_0}^t \frac{1 + h(t')}{\tau_{3\gamma}} dt' \right) \quad (A2.12)$$

の式を「511keV 未満の(遅めの)領域 + Pick-Off による 511keV」領域より抽出した Time の 1 次ヒストグラムに fitting するのもいいかもしれない。

何にせよ、

特定の式を fitting して、とある係数こそが求めたい崩壊寿命だ、とするのならその係数はそう考えるのに正当なものであるかを考えるべきだろう。

A2.2 4.5 章を踏襲した方法、式(A2.12)を用いた方法で各々の q1 を出してみる

以下に 4.5 章を踏襲した方法とこの h(t)用いた式(A2.12)を用いた方法にて q1 の値を算出し、その結果が大きく異なったことのみを記す。

4.5 章で用いたものと同じソースコードを以て解析した場合(S(t)を $0 \leq \text{Energy} < 600$ の範囲のイベント数とした)、h(t)が exp の形とはとても言えないものになった。

別のソースコードを用いた場合 exp の形となったがこれの結果を以下に記す。用いたソースコードは付録 A1 に示す Pickoff_analyze003.c である。h(t)用いた式(A2.12)を用いた方法でも、抽

出範囲を変えて S(0)_alpha を用いて算出する。

4.5 章の方法を踏襲した場合

S(t)は $0 \leq \text{Energy} < 700$ のイベント数とし、S(0)_alphaは $0 \leq \text{Energy} < 450$ のイベント数とした。f(t)の plot する範囲は $200 \leq \text{Time} \leq 750$ とした。f(t)のグラフは図 A2.1 のようになり、また $0 \leq \text{Energy} < 450$ の範囲にて(一律にだが)抽出した Time ヒストグラムに200 から 750の範囲で $q0 \left(p0' \exp\left(-\frac{t}{p1'}\right) + p2' + 1 \right) \exp\left(-\frac{1}{q1} \left(p0' p1' \exp\left(-\frac{t}{p1'}\right) + (p2' + 1)t \right)\right) + q2$ で fitting した。範囲の開始点を 150 でなく 200 としたのは手違いによるものである(h(t)用いた式(A2.12)を用いた方法の場合でも同様)

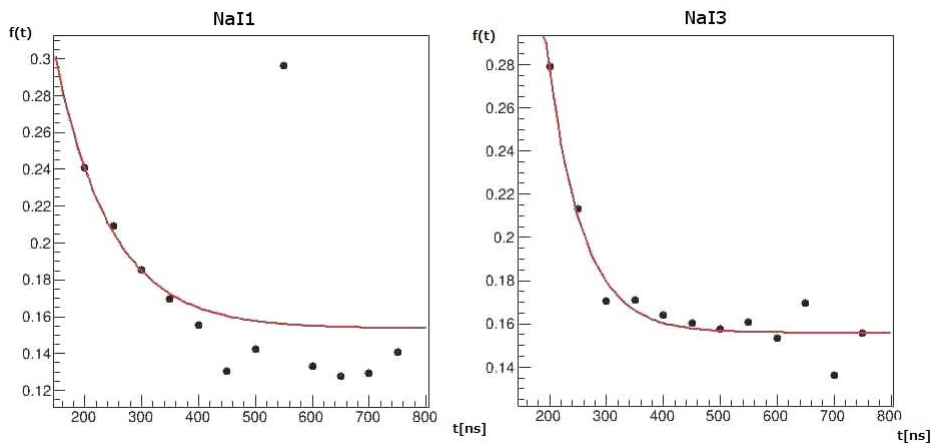


図 A2.1: 4.5 章を踏襲した f(t)のプロット図

NaI3 の崩壊寿命は $q1=191.8 \pm 16.11$

(これは 4.5 章で算出した 188.5 ± 16.8 に近い)

h(t)を用いた式(A2.12)を用いた方法の場合

S(t)は $0 \leq \text{Energy} < 700$ のイベント数とし、S(0)_alphaは $0 \leq \text{Energy} < 600$ のイベント数とした。h(t)の plot する範囲は $200 \leq \text{Time} \leq 750$ とした。h(t)のグラフは図 A2.2 のようになり、また $0 \leq \text{Energy} < 600$ の範囲にて(一律にだが)抽出した Time ヒストグラムに200 から 750の範囲で $q0 \left(p0' \exp\left(-\frac{t}{p1'}\right) + p2' + 1 \right) \exp\left(-\frac{1}{q1} \left(p0' p1' \exp\left(-\frac{t}{p1'}\right) + (p2' + 1)t \right)\right) + q2$ で fitting した。

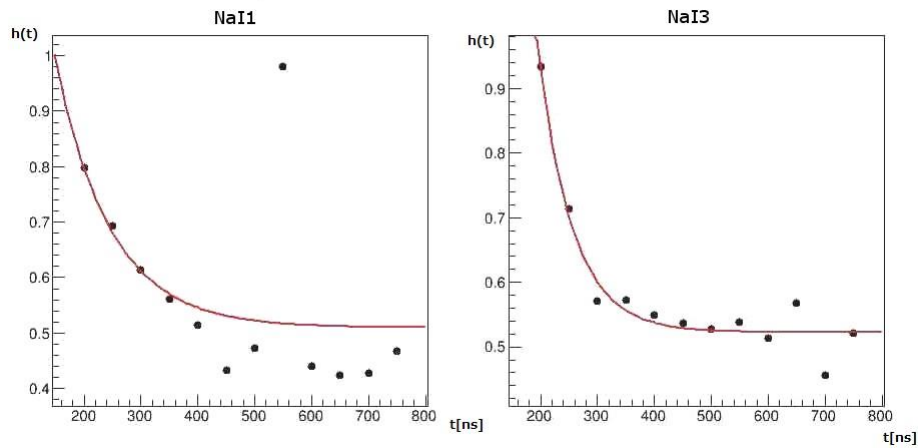


図 A2.2: $h(t)$ を用いた式(A2.12)を用いた方法のプロット図

NaI3 の崩壊寿命は $q1=284.1 \pm 13.50$

2つを比べると $h(t)$ を用いた式(A2.12)を用いた方法は $q1$ の値が大きく異なると分かる。また(4.5章を踏襲した) $f(t)$ と $h(t)$ のグラフの形は似ているが値は異なっている。

ここに $q1$ の値を算出して結果が大きく異なったことだけを記したがこれで終わりではなく、更に **fitting** により求めた各式は元のデータにいかほど近いかを評価し、その指標を以て各式の正当性を議論したいところではある(もちろん求めた式のヒストグラム上での形がデータに沿っていると云えるかも確認して)。が、このレポートはここまででおしまいとする。

参考文献

- [1] J.M.Jauch and F.Rohrlich, *The Theory of Photons and Electrons, 2nd edn*, Springer-Verlag, New York(1976)
- [2] Micheal E.Peskin and Daniel V.Schroeder, *An Introduction to Quantum Field Theory*, Westview(1995)
- [3] F. マンドル・G. ショー (2011)「場の量子論 第一巻 量子電磁力学」(樺沢 宇紀 訳)丸善プラネット
- [4] 2016 年度後期 A2 レポート <https://www-he.scpphys.kyoto-u.ac.jp/gakubu/A2/reports/a2report16b.pdf>