

平成25 年度

卒業研究報告書

研究題目

光学的手法による油中の異物検出

富山高等専門学校

電子制御工学科 古市将義

指導教官 由井四海

平成 26 年 3 月 1 日 提出

目次

1.	はじめに	1
1.1	異物検出の種類	2
2.	光学式油中異物検出装置の測定原理	5
2.1	本実験で使用した実験装置	6
3.	油中異物検出装置の特性評価の際に使用したプログラム	7
4.	4 μ m、7.9 μ mの粒子が混入している油のろ過方法	10
5.	光学式油中異物検出装置の特性評価	12
6.	まとめ	30
7.	付録	31
8.	謝辞	41

1. はじめに

機械設備において潤滑油は機械要素を円滑に動かすために必要不可欠である。潤滑油中の摩耗粒子を分析することによって、その設備の状態を確認する状態監視技術をトライボ診断という。人間の健康診断と同様に、機械設備にも定期的な健康診断が必要である。人間の血液に相当する機械設備の潤滑油を定期的に診断し、それに基づく適切な予防、治療(保守・保全)を施すことにより、機器を長期間健全な状態に保つことが可能となる。この技術により予防保全・計画保全は現実のものとなっている。また、必要十分で効率の良い補修は、メンテナンスコストの大幅な削減に繋がる。本研究では、トライボ診断の際に使用される機器の一つである光学式油中異物検出装置(トライボテックス社 ParticleMATE)の特性を評価することを目的とする。

メーカー、製品名 トライボテックス、ParticleMATE

光学方式	光遮断式+光散乱式
光源	半導体レーザー(LD)
受光部	フォトダイオード(PD)
測定範囲	5 μ m~100 μ m
供給電源	AC100V 50/60Hz

1.1 異物検出装置の種類

動的光散乱法

溶液中に分散した微粒子は、通常ブラウン運動をしている。その動きは大きな粒子では遅く、小さな粒子になるほど早くなる。このとき、ブラウン運動をしている粒子にレーザー光を照射すると、粒子からの散乱光はそれぞれのブラウン運動の速度に対応した揺らぎが観測される。この揺らぎを観測し、光子相関法により自己相関関数を求め、キュムラント法およびヒストグラム法解析を用いることで、ブラウン運動速度を示す拡散係数、さらに粒度分布が求めことができる。

メーカー、製品名 ベックマン・コールター、DelsaMax PRO

光学方式	動的光散乱法
測定範囲(粒子径測定)	0.4 μ m～10 μ m
測定時間	1秒～
温度範囲	4～70 $^{\circ}$ C
電源	100V \pm 10% AC、50/60Hz、1,000VA(W)

レーザー回折・散乱法

粒子にレーザー光を照射すると、その粒子の大きさに応じて様々な方向へ回折光、散乱光が発せられる。粒子径が大きい場合(mm～ μ m)の回折・散乱光は、粒子の後ろ側、即ちレーザー光の進行方向側に集中する。粒子径が小さくなるにつれ(μ m～nm)、回折・散乱光はレーザー光の進行方向の後ろ側へも広がってくる。これらの光の回折・散乱強度分布を解析し

て粒子径を求めることができる。

メーカー、製品名 ベックマン・コールター、LS 13 320 (エルエス 13 320)

光学方式	レーザー回折・散乱法
光源	長波長半導体レーザー 750nm
測定範囲	0.017～2000μm
電源	AC100V±10%, 50/60Hz, 600W

画像イメージング法

光学顕微鏡・電子顕微鏡などで直接、粒子の画像を取得し、その画像イメージから粒子の大きさに換算する。測定する粒子の数が膨大になると解析に時間がかかり、多量粒子の処理には向かない、という短所がある。しかし、動的光散乱法やレーザー回折法と比較して、屈折率や粘度などを考慮しなくてよく、直感的に理解しやすい手法であるのが長所である。

メーカー、製品名 NIKKISO、マイクロトラック SIA

光学方式	画像イメージング法
測定範囲(粒子径測定)	0.75～2000μm
測定時間	10～30秒
光源	LED(ストロボ発光)

重力沈降法

分析する試料を溶媒中に均一に分散させ、粒子の沈降速度からストークスの式を用いて粒度分布を求める方法である。分析装置としては通常アンドレアゼンピペットが利用される。シリンダーの下部から一定時間ごとに試料を吸い上げて分取し、分取した液を蒸発乾固

させることで一定時間に一定距離を沈降した試料の重量を測定し、これを繰り返すことで

粒度分布を得ることができる

メーカー、製品名 ベックマン・コールター、ProteomeLab XL-I

光学方式	重力沈降法
測定範囲(粒子径測定)	0.5nm~100nm
最大波長	190~800nm
電源	単相200V, 50/60Hz, 30A

2. 光学式油中異物検出装置の測定原理

図1に光学式油中異物検出装置の測定原理を示す。光学式油中異物検出装置は図1の通り、3つのセンサーを用いて異物の測定をする。受光部①ではセルの中を通過する異物によってレーザー光が遮断される。その遮断回数と受光電圧の大きさから異物の数と大きさを計測する。さらに、受光部②、③によって異物と気泡の識別が可能となる。気泡が通過するとする場合、気泡は真円なのでレーザー光が乱反射する角度は受光部②も③において同じである。そのため受光部②、③での計測できる受光電圧はほぼ等しい。しかし異物は真円ではないことが多くレーザー光が乱反射する角度が異なる。そのため受光部②、③の受ける受光電圧は等しくならない。この違いを利用して異物と気泡の識別が可能である。

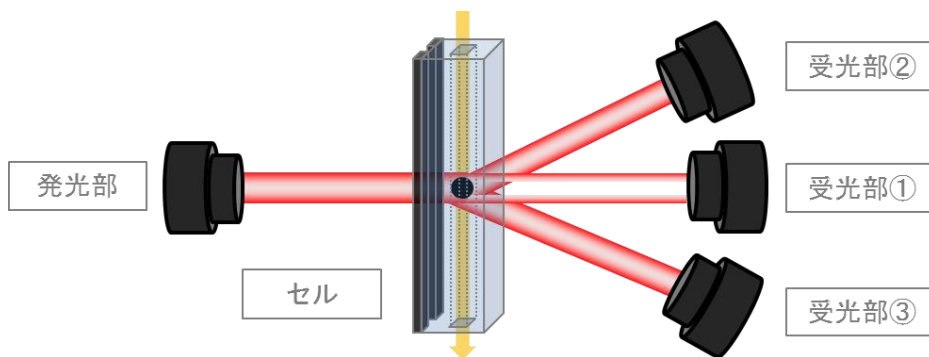


図1 光学式油中異物検出装置の測定原理

2.1 本実験で使用した実験装置

図2に本実験で使用した実験装置を示す。まず流量コントローラー(テラル、と光学式油中異物検出装置をチューブで接続し、装置とチューブ内を異物が混入した油で満たす。油は流量コントローラーによってチューブ内を循環している。流量は20ml/minである。

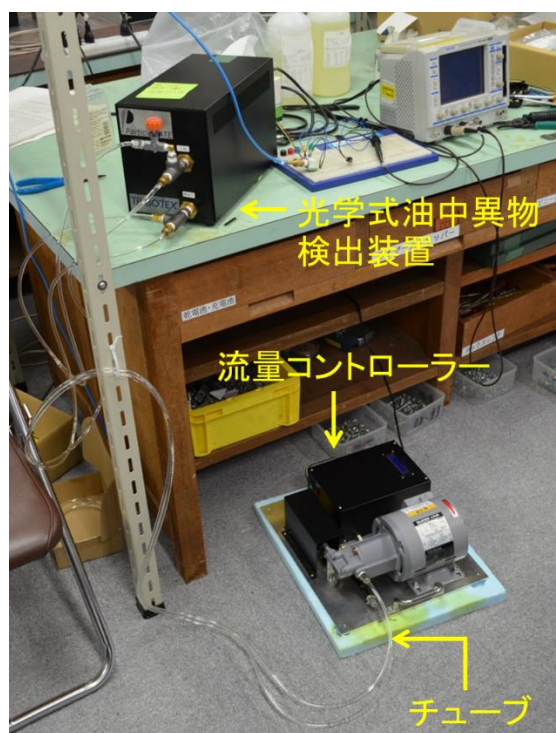


図2 実験装置

3. 油中異物検出装置の特性評価の際に使用したプログラム

本実験では図 1 の受光部①～③の受光電圧の大きさとその頻度を記録するプログラムをいくつか作成した。その詳しい内容を以下に示す。

初めに図 3 のような波形がオシロスコープで読み取れる。その波形を BRINGO Link Tool というソフトを用いて PC に保存した。このデータから異物が通ったとされる電圧の部分だけをプログラムを用いて保存した。以下にそのプログラムのフローチャートを示す。

図 4 のフローチャートは実験 1、2 で使用した。実験 1、2 で使用したオシロスコープはプロットの点が多く、(5 秒間で 2000000 プロット) 1 つの波形にたくさんのプロットを使用したため単純移動平均を求めることで波形データ平滑化し、間違っただけが出ないようにした。プログラムの手順は、まず、PC に保存されている波形データを開く。次に 22 点の単純移動平均を求め、波形の平滑化を行う。そしてその状態でピーク検出をする。最後にピーク値をファイルに保存する。

図 5 のフローチャートは実験 3、4 で使用した。実験 3、4 で使用したオシロスコープは実験 1、2 で使用したオシロスコープと違ってプロットの数が少ない (5 秒間で 102400 プロット) ので波形データを単純移動平均で平滑化する必要がなかった。

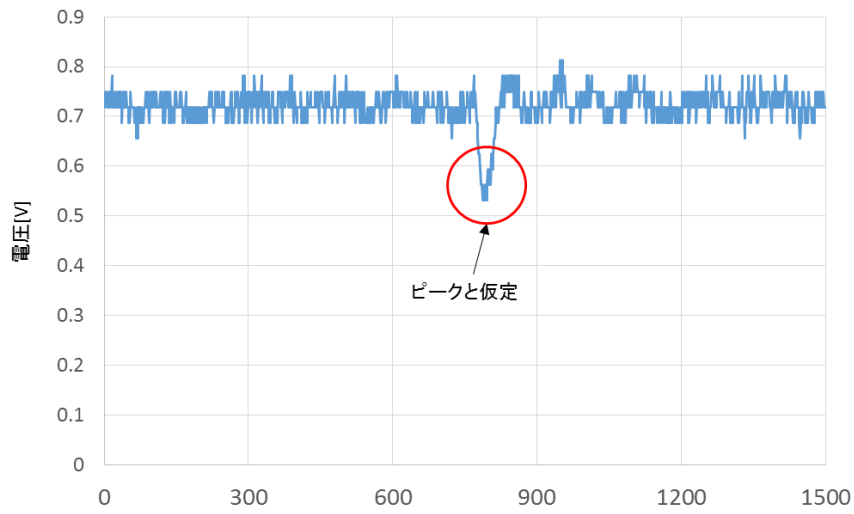


図 4 光学式油中異物検出装置の電圧波形

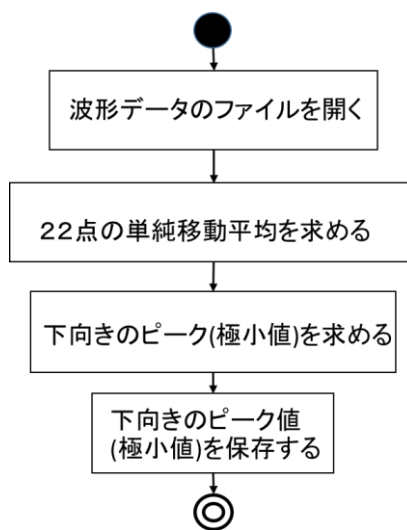


図 5 実験 1、2 で使用したプログラムのフローチャート

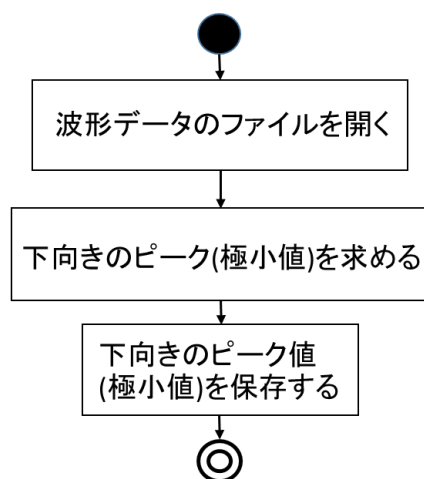


図 6 実験 3、4 で使用したプログラムのフローチャート

4. $4\mu\text{m}$ 、 $7.9\mu\text{m}$ の粒子が混入している油のろ過方法

漏斗を漏斗台に乗せ、漏斗の下にビーカーを置く。ろ紙を4つ折りにして、円錐形に開いて漏斗に入れる。漏斗に蒸留水を垂らしてメンブランフィルターを漏斗に密着させる。そのあとに粒子をろ紙に垂らす。どちらの粒子の場合も一回の実験で容器の内の4分1程度を使用した。粒子の乾燥には1~2日の期間を要した。乾燥させた粒子は少量の油と混ぜた後に実験装置内に混入させた。



図7 粒子のろ過装置



図8 メンブランフィルター(左が孔径 $0.4\mu\text{m}$ 、右が孔径 $5.0\mu\text{m}$)



図9 使用した粒子(左が $7.9\mu\text{m}$ 、右が $4.0\mu\text{m}$)

5. 光学式油中異物検出装置の特性評価

実験 1

図 1 の受光部①の受光電圧を 5 秒間 (200KS/s) オシロスコープ 1 (Agilent Technologies, MS06034A) で測定し、これを 5 回繰り返し、受光電圧の大きさとその頻度をグラフで表した。その結果と、別の油中異物検出装置で測定した結果が同じようになるかどうかグラフから読み取った。

まず受光部①の電圧の大きさとその頻度から図 11 の粒径度数分布表を示した。図 11 は光学式油中異物検出装置で 1 回～5 回測定した結果をグラフで示している。図 10 は油ボトルのラベルに書いてあったデータを示している。これは企業から送られてきたものである。なので図 10 を基準として考えた。図 10 と図 11 の横軸は単位が違うので計算によって図 11 の横軸を区切った。以下にその方法を示す。

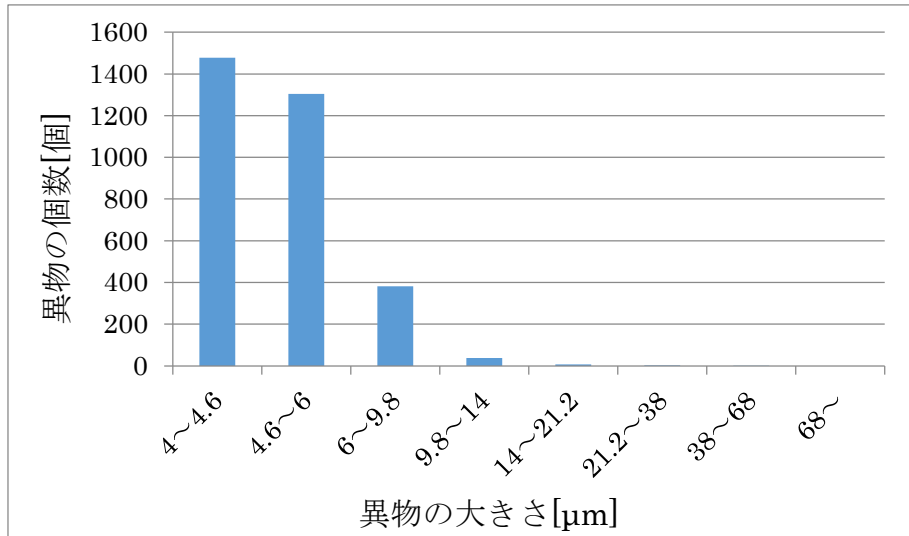


図 10 油ボトルのラベルに書かれていたデータ

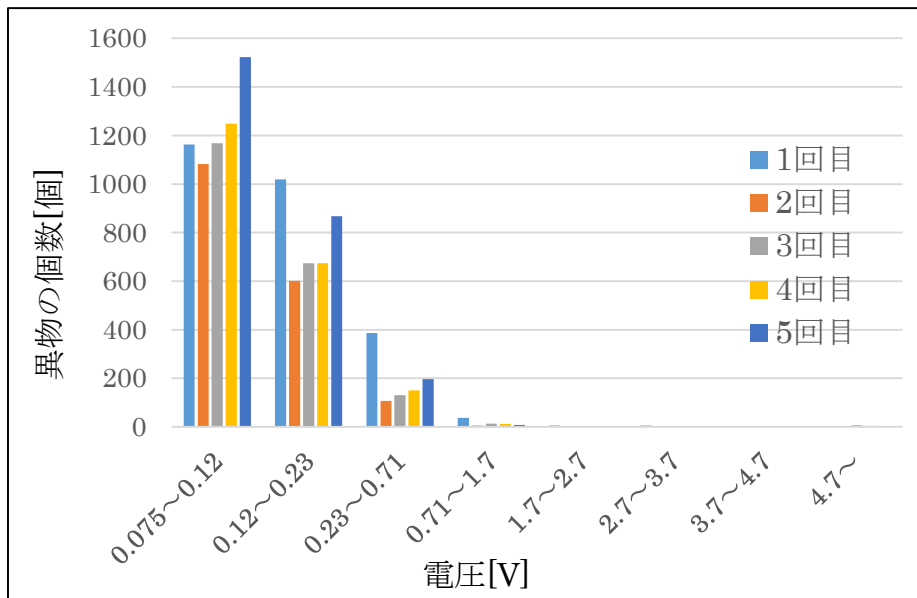


図 11 粒径度数分布図

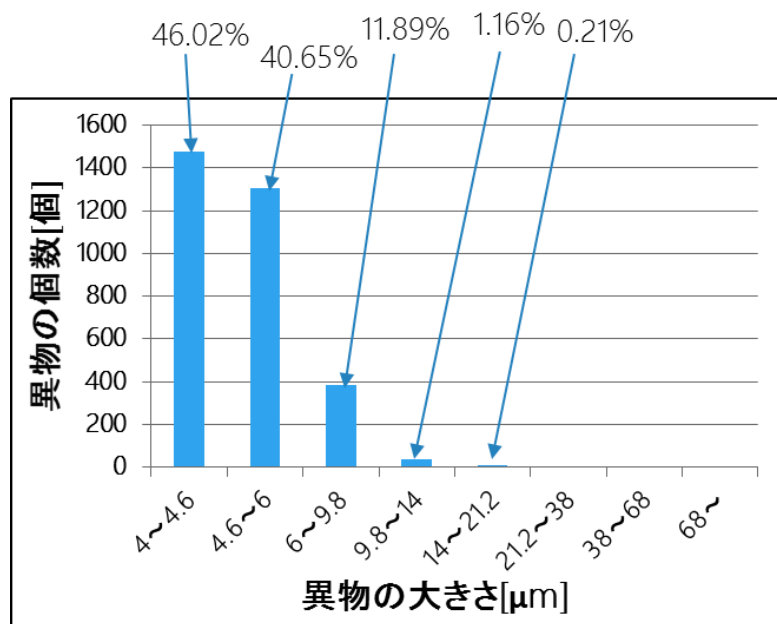


図 12 図 10 の異物の大きさの区切りごとの割合

図 12 のように図 10 の異物を大きさの区切りごとに割合で表した。次に図 11 の 1 回目の測定結果が図 10 の割合になるように電圧を変更した。この横軸の区切りで 2 ~ 5 回目の測定結果を図 11 に示した。

表 1 に図 11 の 1 回目の測定結果の割合を基準としたときの 2 ~ 5 回目の誤差率を示した。

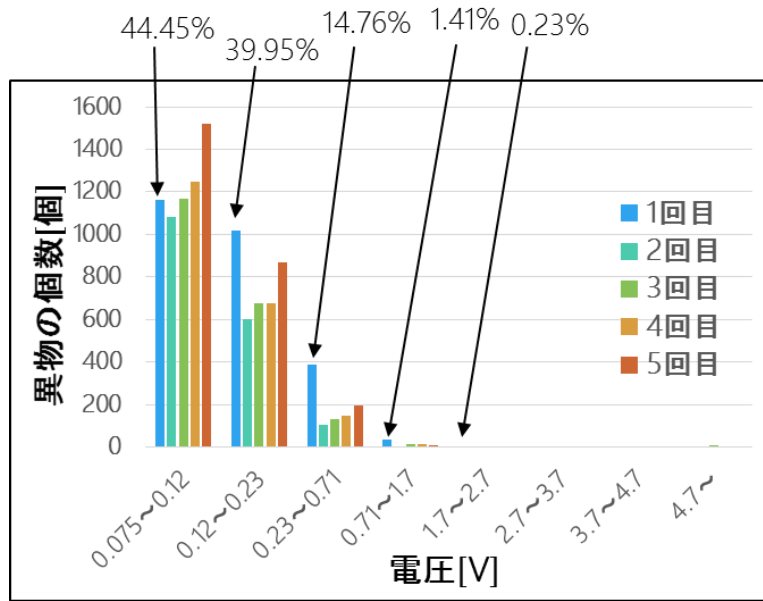


図 13 図 11 の 1 回目の電圧の大きさの区切りごとの割合

電圧[V]	2回目誤差率[%]	3回目誤差率[%]	4回目誤差率[%]	5回目誤差率[%]
0.075~0.12	23.6	21.5	23.9	21.5
0.12~0.23	21.2	20.2	26.1	21.6
0.230.71	100	80.9	65.7	56.7
0.71~1.7	248	77.8	102	276

表 1 図 11 の 1 回目の測定結果の割合を基準としたときの 2 ~ 5 回目の誤差率

実験 2

受光部①、②、③の受光電圧を同時に測定し、その結果からそのときのチューブ内に気泡がどのくらい混入しているかを調べた。まず光学式油中異物検出装置の受光部①の受光電圧から異物の大きさとその頻度を測定して図 14 に示した。そのときの受光部②、③の受光電圧の関係から気泡がどのくらいあるのかを調べた。図 15 に受光部①、②、③の受光電圧を同時に測定したときの受光部②、③の電圧の関係を示した。そのときの受光部②と③の電圧の差がほぼ同じもの(変化量の差が 0.003V 以内)を気泡と仮定してその数を測定した。この方法で 10 回の測定を行い、その結果を表 2 に示した。表 2 から気泡は平均で 12.1 個検出されることが分かった。このとき検出された異物の個数は平均で 1248.1 個だった。つまり気泡は測定結果全体の 0.96%を占めていることが分かった。

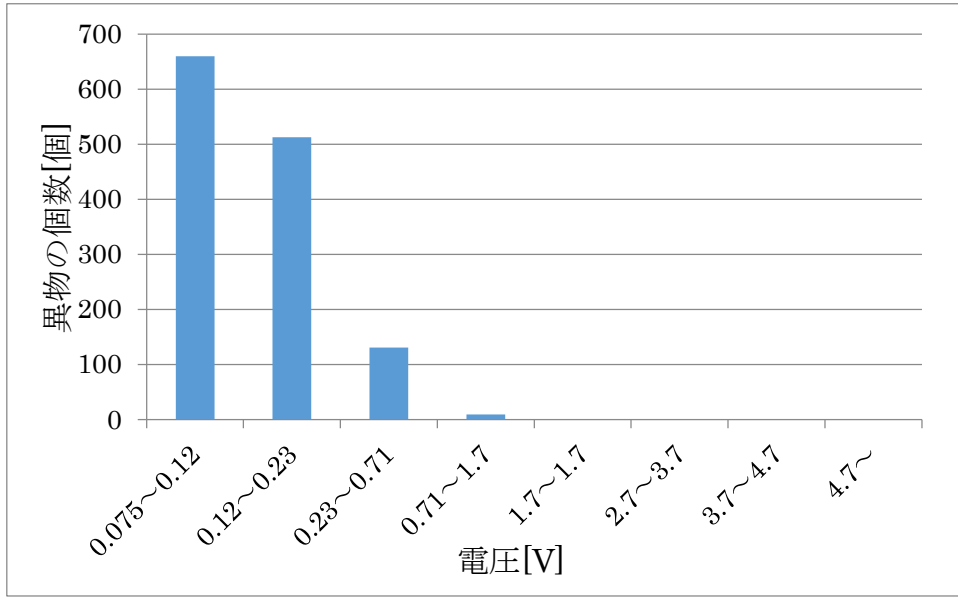


図 14 粒径度数分布図

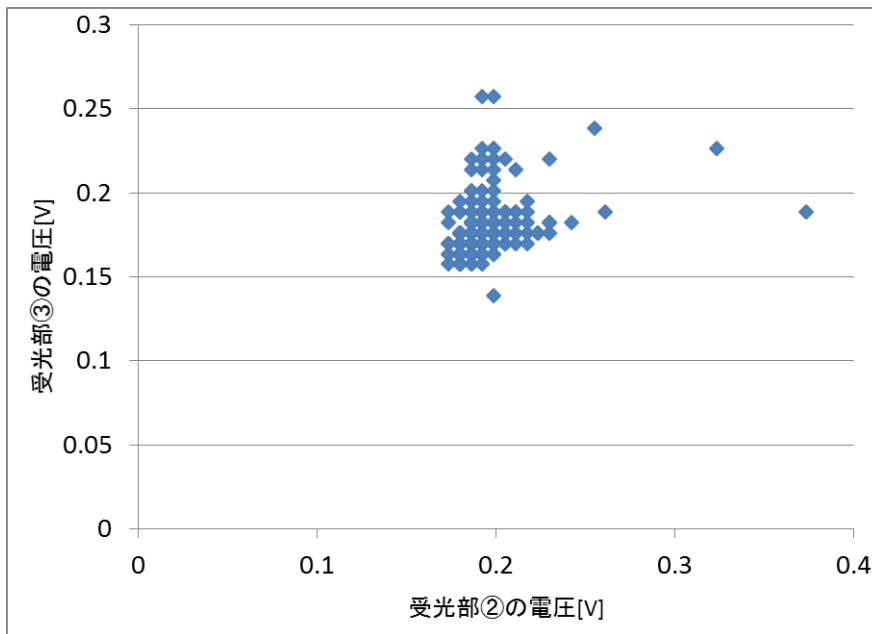


図 15 受光部②、③の電圧の関係

測定回数	1回目	2回目	3回目	4回目	5回目	
気泡の数[個]	6	16	11	12	10	
	6回目	7回目	8回目	9回目	10回目	平均
	12	25	10	6	13	12.1

表2 検出した気泡の数

実験 3

次にオシロスコープ 1 ではなくオシロスコープ 2(IWATSU、DS-8812)を用いて受光部①の受光電圧を 5 秒間(20KS/s)測定し、これを 5 回繰り返して、受光電圧の大きさとその頻度をグラフで表した。そのあとに発光部の位置を図 3 のように 2.5,5.0,7.5,10mm 移動させて受光電圧を 5 秒間測定し、これをそれぞれ 5 回繰り返したときの受光電圧の大きさとその頻度を測定した。発光部の位置を遠ざけると電圧がどのように変化するかを調べた。

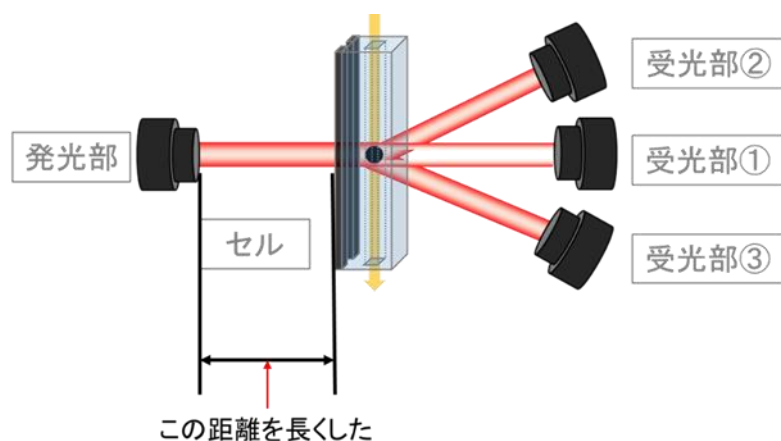


図 16 光学式油中異物検出装置

図 17 に発光部①の位置を変更したときに検出される異物の個数がどのように変化したのかをグラフで示した。次に発光部の距離を変更したときの受光電圧の大きさとその頻度の平均のグラフを図 17 に示す。図 17 から発光部とセルの距離を離すと異物の個数は増加することが分かった。5mm と 7.5mm 離れたときは異物の検出が安定しなかった。10mm 離すと異物の個数が安定して増加した。図 18 には発光部を遠ざけたときの異物の個数と大きさの種類の変化を示した。異物の大きさの変化においては、まず 0.12~0.23[V]と

0.23~0.71[V]の異物の個数がそれぞれ増加した。そして5mm 離れたときには0.12~0.23[V]

より0.23~0.71[V]の異物の個数が多くなった。

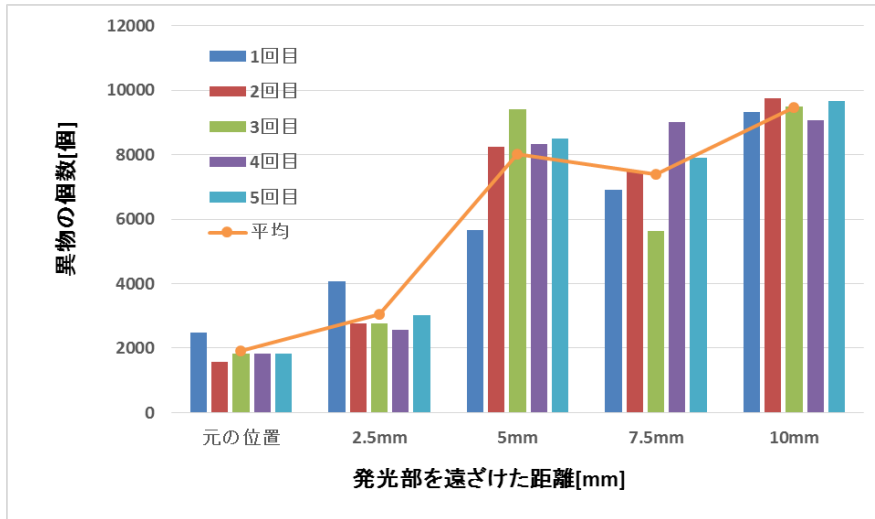


図 17 発光部①の位置を変更したときに検出される異物の個数

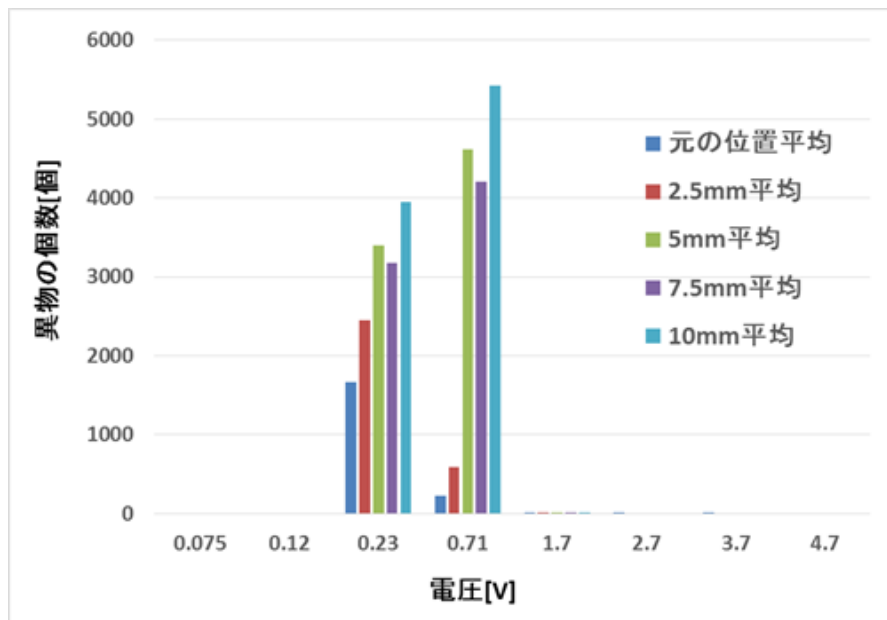


図 18 発光部を遠ざけたときの異物の個数と大きさの種類の変化

実験 4

次に今まで異物が混入している油で測定したが、異物がほとんど混入していない油に変更して受光部①の受光電圧を 5 秒間測定し、これを 5 回繰り返したときのそれぞれの受光電圧の大きさとその頻度を図 19 で示した。この結果から図 17 の元の位置平均の異物の検出数と比較して、異物がほとんど混入していない油は異物の数が 10 分の 1 になったことが読み取れた。

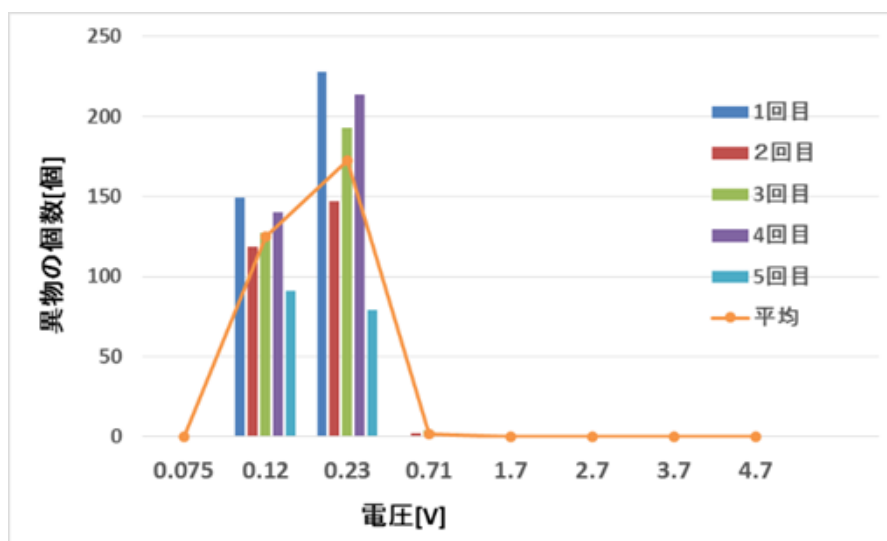


図 19 粒径度数分布図(異物がほとんど混入していない場合)

実験 5

次に $4\mu\text{m}$ の粒子が混入している油に変更して受光部①の受光電圧を5秒間測定し、これを5回繰り返したときのそれぞれの受光電圧の大きさとその頻度を図 20 に示した。

図 18 の元の位置平均の異物の検出数と比較すると $0.12\sim 0.23\text{V}$ の区間異物の個数が増加していることが分かった。

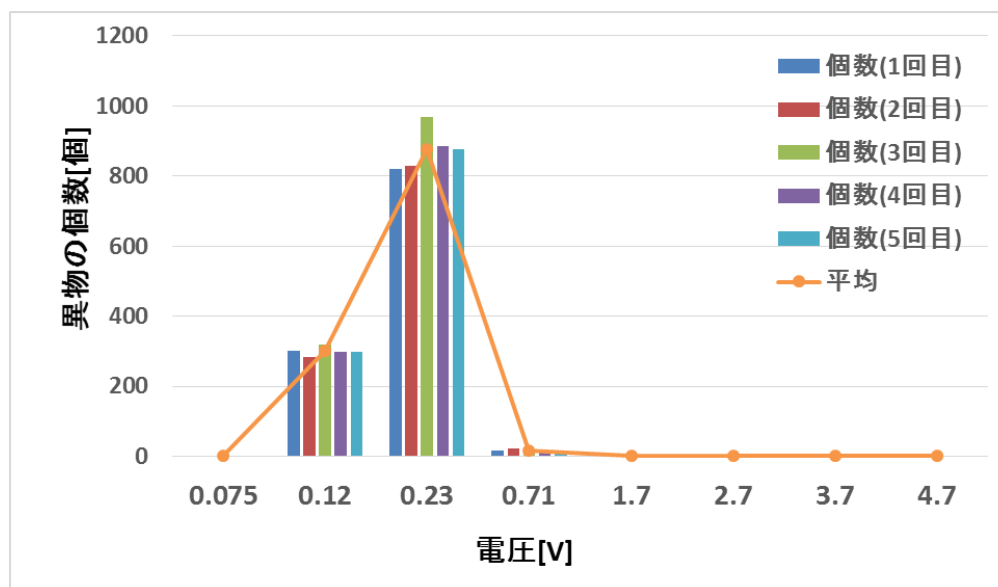


図 20 粒径度数分布図($4\mu\text{m}$ の粒子混入時)

実験 6

実験 3~5 を行った後にいくつか疑問が見つかったため、実験方法を変更して実験を行った。実験 3~5 の横軸の電圧の区切りに疑問が残ったため横軸の区切りかたを変更した。この状態で実験 1~3 で使用した油を使用して受光部①の受光電圧を 5 秒間測定し、それを 5 回繰り返したときのそれぞれの受光電圧の大きさとその頻度を図 21 に示した。このときの横軸の電圧の区切りは実験 1 と同様の方法で変更した。この結果、図 10 の横軸の割合と同じようになった。

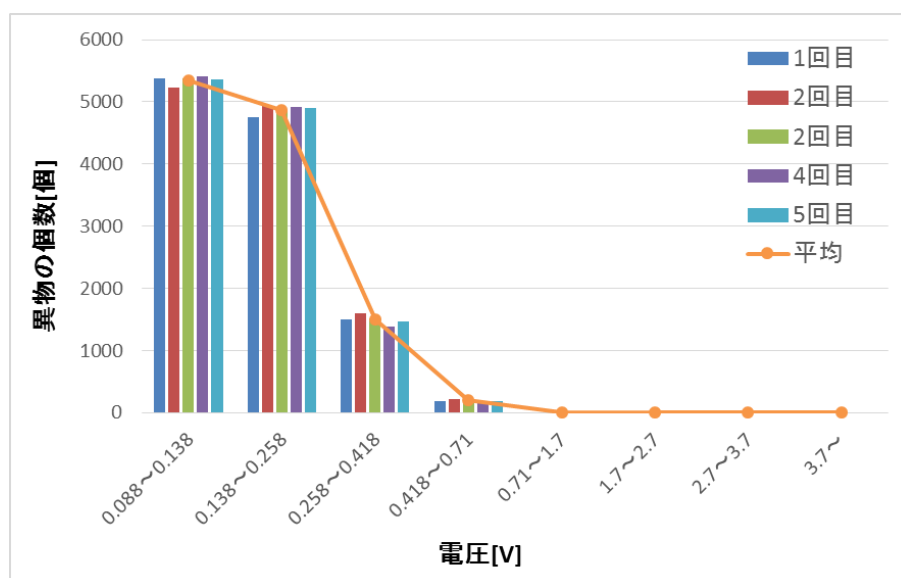


図 21 粒径度数分布図

実験 7

次に実験 6 で使用した油に 4 μm の粒子を混入させ、受光部①の受光電圧を 5 秒間測定し、これを 5 回繰り返したときのそれぞれの受光電圧の大きさとその頻度を図 22 で示した。0.88~0.138V の区間の割合が大きく増加するのではないかと考えたが 0.138~0.258V、0.258~0.418V、0.418~0.71V、0.71~1.7V の区間の割合が図 17 と比べて増加した。0.88~0.138V の区間の割合は減少した。

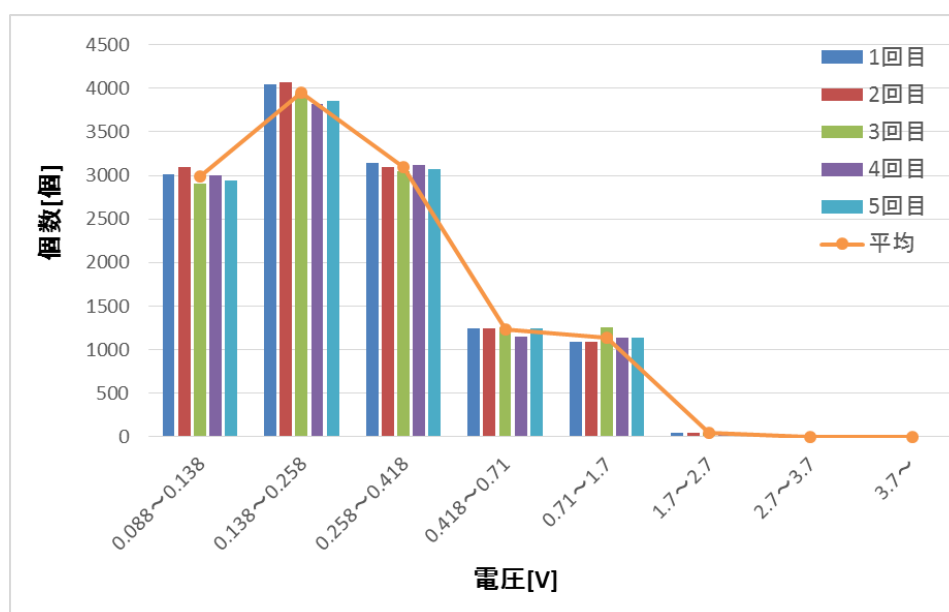


図 22 粒径度数分布図(4 μm の粒子混入時)

実験 8

次に実験 6 で使用した油に $7.9\mu\text{m}$ の粒子を混入させ、受光部①の受光電圧を 5 秒間測定し、これを 3 回繰り返したときのそれぞれの受光電圧の大きさとその頻度を図 23 に示した。 $0.258\sim 0.418\text{V}$ の区間の割合が大きく増加するのではないかと考えたが $0.138\sim 0.258\text{V}$ の区間の割合が図 21 と比べて最も増加した。 $0.258\sim 0.418\text{V}$ 、 $0.71\sim 1.7\text{V}$ の区間の割合も図 21 と比べて増加した。 $0.88\sim 0.138\text{V}$ の区間の割合は減少した。

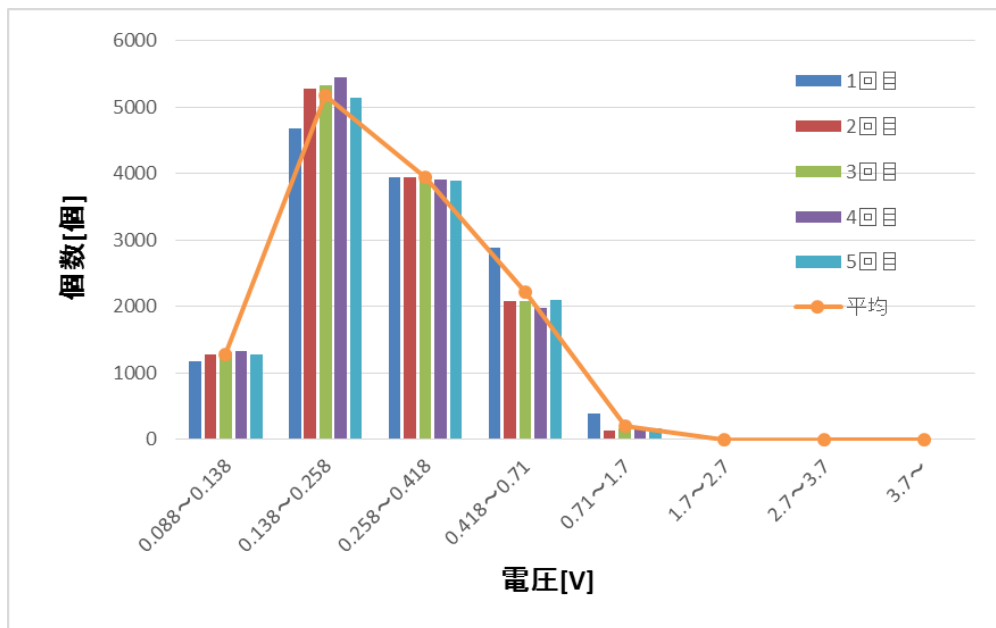


図 23 粒径度数分布図($7.9\mu\text{m}$ の粒子混入時)

実験 9

使用するオシロスコープをオシロスコープ 1 に戻し、あらかじめ異物が混入している油で光学式油中異物検出装置の受光電圧の大きさとその頻度を 20 分間測定した。この 20 分というのは油が実験装置内を 1 周するのにかかる時間である。これを 3 回繰り返したときの平均を図 24 で示した。実験 1、6 と同様に図 10 を基準として考えた。図 10 と図 24 の横軸は単位が違うので計算によって図 24 の横軸を区切った。以下にその方法を示す。

図 24 の結果が出た後、図 10 を見て横軸を異物の大きさの区切りごとに割合で表した。次に図 24 の横軸の電圧を図 10 の割合になるように横軸の電圧の区切りを変更した。こうすることで図 25 のグラフが求まった。この図 25 の横軸の電圧の区間と図 10 の横軸の異物の大きさは対応していると考えました。この横軸の区切りでこの後の実験を行った。

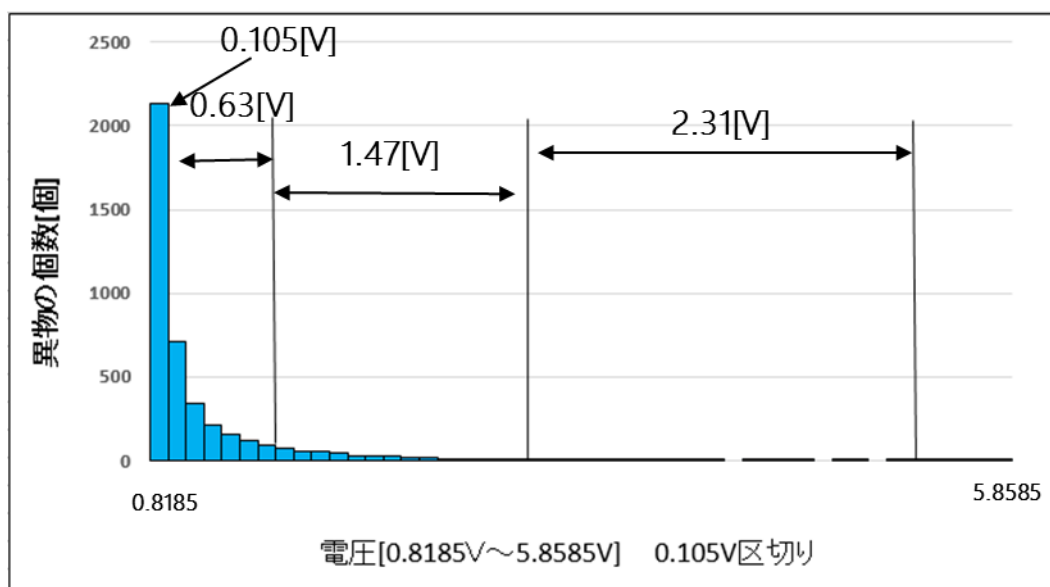


図 24 ヒストグラム

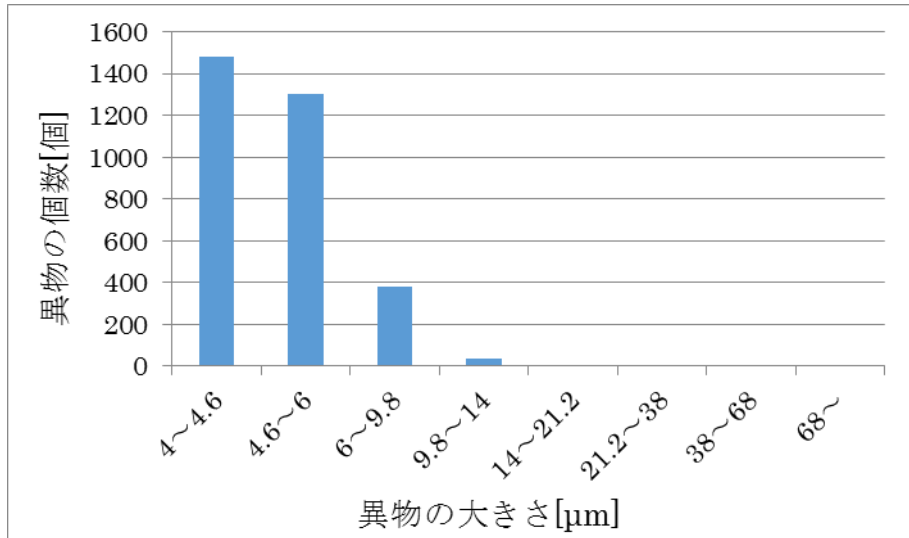


図 10 油ボトルのラベルに書かれていたデータ

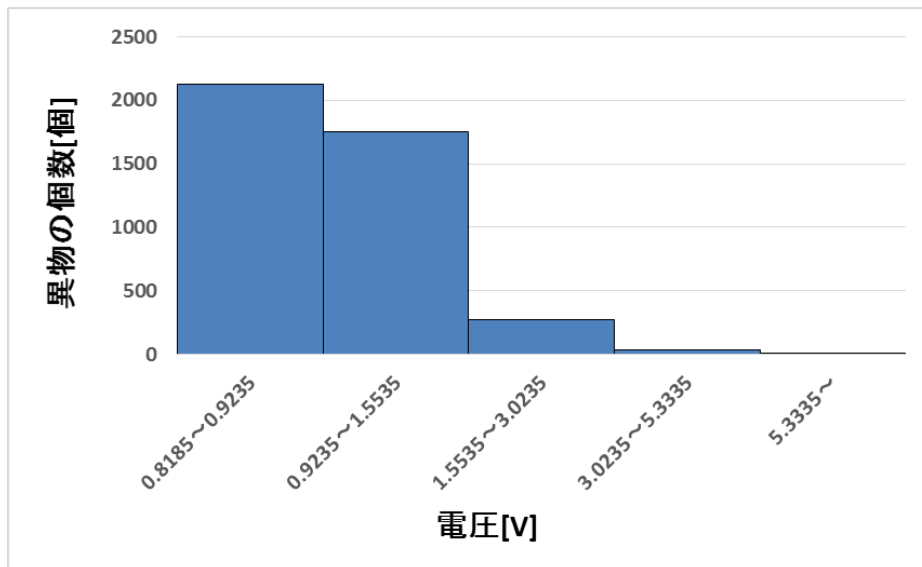


図 25 粒径度数分布図

実験 10

実験 7 で用いた油に 4 μm の粒子を混入させて受光電圧の大きさとその頻度を 20 分間測定した。これを 3 回繰り返したときの平均をグラフで図 22 に示した。図 22 は 4 μm の区間の割合が図 21 に比べて増加した。

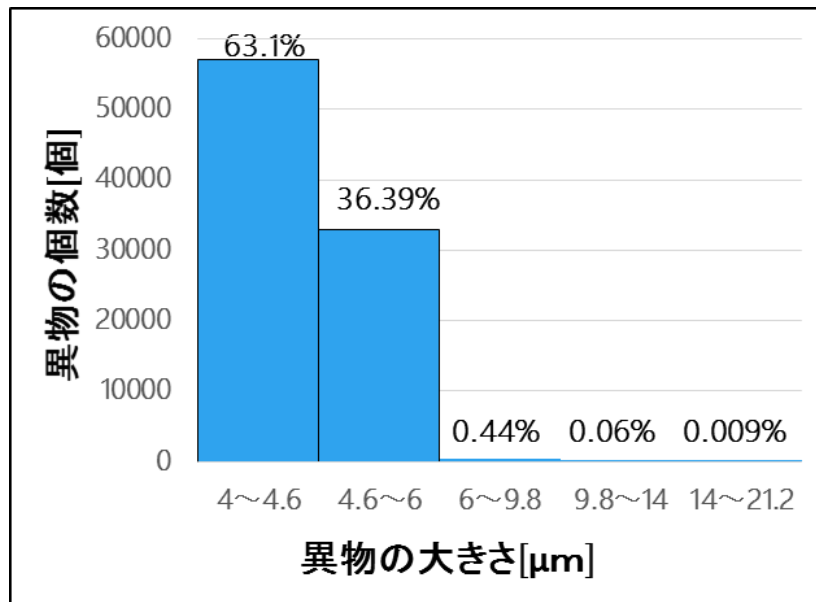


図 26 粒径度数分布図(4 μm の粒子混入時)

実験 11

実験 7 で用いた油に $7.9\mu\text{m}$ の粒子を混入させて受光電圧の大きさとその頻度を 20 分間測定した。これを 3 回繰り返したときの平均を図 27 で示した。図 26 は $4\mu\text{m}$ の区間の割合が図 25 に比べて増加した。 $7.9\mu\text{m}$ の区間の割合は図 25 に比べて減少した。

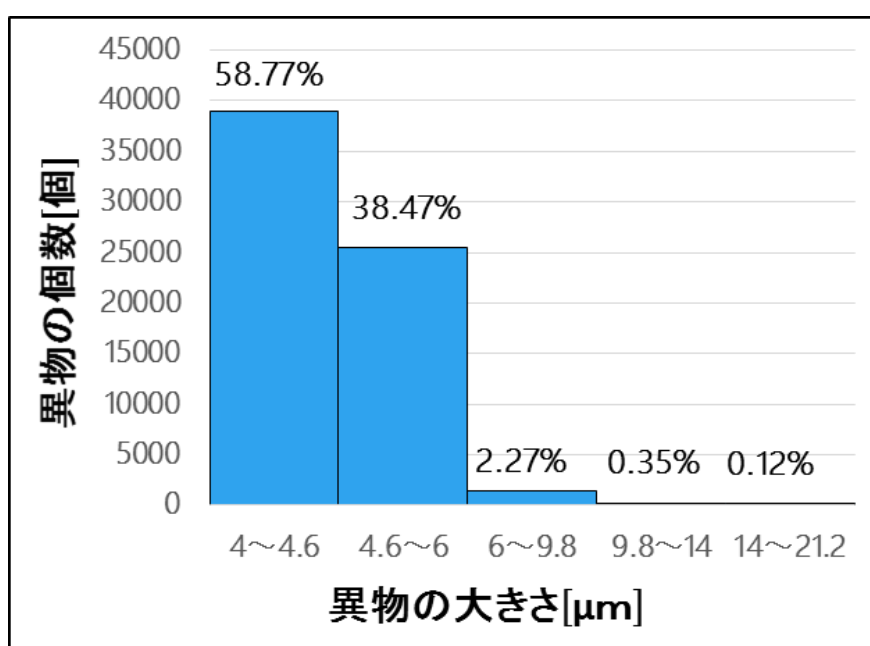


図 27 粒径度数分布図($7.9\mu\text{m}$ の粒子混入時)

6. まとめ

実験 1～5 をやってみた結果、気泡判別は本実験には関係が薄いと考えた。実験で使用した横軸の電圧の区切りかたと、オシロスコープ 1 の細かいプロットに作成したプログラムではうまくいかないと感じた。そのためオシロスコープを変更し、横軸の電圧の区切りも変更した。そうしたがまたしてもうまく測定できていないと感じた。そのため測定方法をもう一度見直した。だが満足する結果は得られていない。これにはいくつかの要因が挙げられる。

まず、横軸の電圧の区切りかたである。区切りがうまくいけば $4\mu\text{m}$ の粒子が混入すると $4\mu\text{m}$ の区間の割合だけが増加し、 $7.9\mu\text{m}$ の粒子を混入させたときは $7.9\mu\text{m}$ の区間の割合が増加すると考えている。そして異物の個数が 1 回 1 回の測定でばらけることから測定時間を見直す必要もあると考えている。

7. 付録

```
#include <stdio.h>
#include <math.h>

#define MAX 100240
#define UNDER_LIMIT 0.67493318

//変更禁止
#define ZERO_LEVEL 0
//重み係数の最大個数
//平滑化区間幅を最大 21 点とするため
void calculate(double[],double[],int[],int,int);

int getUnderMaxPeak(double[],int,int[],double,double);

int main(void)
{
    FILE *fp,*rp;
    int np,i,j;//平滑化区間幅 np は 3 以上,i はループ変数
    static double inData[MAX];//入力
    double gomi[4],hit[2];//ファイルから不必要なデータを読み取るときに使う変数
    static int peak[MAX];//ピーク値の格納
    double sum,average;
    int peakEle;
    char readFileName[25]="ffff12.csv";
    char memoryFileName[25]="wave_3.csv";
    char writeFileName[25]="histgramwave_053.csv";
    //平滑化区間幅の設定
    /*printf("np:");
    scanf("%d",&np);
    if(np<3){
        printf("np は 3 以上です¥n");
        return 0;
```

```

}

printf("np=%d\n",np);*/
np=3;
//
//
//ファイルオープン
fp= fopen(readFileName,"r");
if(np<3){
    printf("np は 3 以上です\n");
    return 0;
}
if(fp != NULL){
    printf("file was opened\n");
}
else{
    printf("file not found\n");
    return 0;
}

//
//データ入力と初期化
for(i=0;i<MAX;i++){
    fscanf(fp,"%lf,%lf",&inData[i],&gomi[0]);
}
fclose(fp);

sum=0;
for(i=0;i<MAX;i++){
    sum += inData[i];
}
average=sum/MAX;

//平均を引いた波形データを保存する
fp= fopen(memoryFileName,"w");
for(i=0;i<MAX;i++){

```

```

        inData[i] -= average;
        fprintf(fp, "%lf\n", inData[i]);
    }
fclose(fp);
//

//下向きのピーク(極小値)を求める
peakEle = getUnderMaxPeak(inData, (np-1)/2, peak, UNDER_LIMIT-average, average);

int peakNum[peakEle];
for(i=0; i<peakEle; i++){
    peakNum[i]=peak[i];
}

fp= fopen(writeFileName, "w");
rp= fopen(readFileName, "r");
fprintf(fp, "平均値, %lf\n", average);
fprintf(fp, "1 列目\n");
j=0;
//下向きのピーク(極小値)をファイルに保存する
for(i=0; i<MAX; i++){
    if(j<peakEle){
        if(i==peakNum[j]){
            fprintf(fp, "%lf\n", fabs(inData[peakNum[j]]));
            j++;
        }
    }
    else{
        break;
    }
}
fclose(fp);
fclose(rp);

//

```

```

        return 0;
    }

int getUnderMaxPeak(double inData[MAX],int m,int peak[MAX],double under,double ave){
    FILE *fp;
    double p=0;//微分係数
    double old_p=0;//1つ前の微分係数
    int i,j;
    int peakNum=0;
    int last;//peak は一時保存用
    int storage;
    for(i=m;i<MAX-m+1;i++){
        p=0;
        for(j=1;j<m+1;j++){
            p+=(inData[i+j]-inData[i-j])*j;
        }
        if(old_p<0 && p>=0){//極小値の検出
            peak[peakNum]=i;
            peakNum++;//ピーク総数
        }
        old_p=p;
    }
    for(i=0;i<peakNum;i++){
        //ピーク位置を再調査
        for(j=peak[i]-m;j<peak[i]+m+1;j++){
            last = peak[i];
            if(inData[j]<inData[peak[i]]){
                last = j;
            }
        }
        peak[i] = last;
    }

    storage = 0;
    for(i=0;i<peakNum;i++){
        //信号が入力されておらず基準値近傍のピーク信号値の場合

```

```

        if(inData[peak[i]]-ZERO_LEVEL<=under){
            peak[storage] = peak[i];
            storage++;
        }
    }

    return storage;
}

#include <stdio.h>
#include <math.h>

#define MAX 2000000
#define UNDER_LIMIT 0.634

//変更禁止
#define ZERO_LEVEL 0
//重み係数の最大個数
//平滑化区間幅を最大 21 点とするため
void calculate(double[],double[],int[],int,int);

int getUnderMaxPeak(double[],double[],int,int[],double,double);

int main(void)
{
    FILE *fp,*rp;
    int np,i,j,m;//平滑化区間幅 np は 3 以上,i はループ変数
    static double inData[MAX],mean[MAX];//入力
    double gomi[4],hit[2];//ファイルから不必要なデータを読み取るときに使う変数
    static int peak[MAX];//ピーク値の格納
    double sum,average,halfPeakValue;
    int peakEle;
    char readFileName[25]="20130911103048.csv";
    char memoryFileName[25]="mean22_38.csv";

```

```

char writeFileName[25]="histgramwave22_038.csv";
//m は単純移動平均の点数
m = 22;

//平滑化区間幅の設定
/*printf("np:");
scanf("%d",&np);
if(np<3){
    printf("np は 3 以上です¥n");
    return 0;
}

printf("np=%d¥n",np);*/
np=3;
//
//
//ファイルオープン
fp= fopen(readFileName,"r");
if(np<3){
    printf("np は 3 以上です¥n");
    return 0;
}
if(fp != NULL){
    printf("file was opened¥n");
}
else{
    printf("file not found¥n");
    return 0;
}

//
//データ入力と初期化
for(i=0;i<MAX;i++){
    fscanf(fp,"%lf,%lf",&gomi[0],&inData[i]);
}
fclose(fp);

```

```

sum=0;
for(i=0;i<MAX;i++){
    sum += inData[i];

average=sum/MAX;

for(i=0;i<MAX;i++){
    inData[i] -= average;
}

//22 点の単純移動平均値を求める
for(i=0;i<MAX;i++){
    mean[i] = inData[i];
}

for(i=m;i<MAX-m;i++){
    sum=0.0;
    for(j=-m;j<=m;j++){
        sum += inData[i+j];
    }
    mean[i]=sum/(2*m+1);
}
//

fp= fopen(memoryFileName,"w");
for(i=0;i<MAX;i++){
    fprintf(fp,"%lf\n",mean[i]);
}
fclose(fp);

//下向きのピーク(極小値)を求める
peakEle      =      getUnderMaxPeak(inData,mean,(np-1)/2,peak,UNDER_LIMIT-
average,average);

int peakNum[peakEle],gapNum[peakEle];

```

```

long startNum[peakEle],endNum[peakEle];
for(i=0;i<peakEle;i++){
    peakNum[i]=peak[i];
}

//半値幅求める
for(i=0;i<peakEle;i++){
    halfPeakValue = inData[i]/2;
    for(j=peak[i];inData[j]<halfPeakValue;j--){
    }
    startNum[i]=j;
    for(j=peak[i];inData[j]<halfPeakValue;j++){
    }
    endNum[i]=j;
    gapNum[i]=endNum[i]-startNum[i];
}
//

int gapCount=0;
//半値幅が重なっている箇所を数える
for(i=1;i<peakEle;i++){
    if(gapNum[i-1] != gapNum[i]){
        gapCount++;
    }
}

//ファイルに保存する
fp= fopen(writeFileName,"w");
rp= fopen(readFileName,"r");
fprintf(fp,"平均値,%lf,重なりなしピーク,%d,ピーク総数,%d,誤検出ピーク
数,%d\n",average,gapCount,peakEle,peakEle-gapCount);
fprintf(fp,"ピーク値,半値幅,ピーク番号,半値開始番号,半値終了番号\n");
for(i=0;i<peakEle;i++){

    fprintf(fp,"%lf,%d,%d,%ld,%ld\n",fabs(inData[peakNum[i]]),gapNum[i],peakNum[i
],startNum[i],endNum[i]);

```



```

    }
    fclose(fp);
    fclose(rp);
    //

    return 0;
}

int getUnderMaxPeak(double inData[MAX],double mean[MAX],int m,int peak[MAX],double
under,double ave){
    FILE *fp;
    double p=0;//微分係数
    double old_p=0;//1つ前の微分係数
    int i,j;
    int peakNum=0;
    int last;//peak は一時保存用
    int storage;
    for(i=m;i<MAX-m+1;i++){
        p=0;
        for(j=1;j<m+1;j++){
            p+=(mean[i+j]-mean[i-j])*j;
        }
        if(old_p<0 && p>=0){//極小値の検出
            peak[peakNum]=i;
            peakNum++;//ピーク総数
        }
        old_p=p;
    }
    for(i=0;i<peakNum;i++){
        //ピーク位置を再調査
        for(j=peak[i]-m;j<peak[i]+m+1;j++){
            last = peak[i];
            if(inData[j]<inData[peak[i]]){
                last = j;
            }
        }
    }
}

```

```

        peak[i] = last;
    }

    storage = 0;
    for(i=0;i<peakNum;i++){
        //信号が入力されておらず基準値近傍のピーク信号値の場合
        if(inData[peak[i]]-ZERO_LEVEL<=under){
            peak[storage] = peak[i];
            storage++;
        }
    }

    last=0;
    //ピーク場所が重なっている箇所を取り除く
    for(i=1;i<storage;i++){
        if(peak[i-1] != peak[i]){
            peak[last] = peak[i-1];
            last++;
        }
    }
    return last;
}

```

8. 謝辞

本研究を行うにあたり、実験において適切なお指導をくださいました富山高等専門学校制御情報システム工学専攻科由井四海教員、プログラムを作成するにあたって適切な助言をくださった制御情報システム工学専攻科2年宮田大暉氏に深く感謝いたします。