Table of Contents

[vdPauw\_f.py マニュアル 2](#_Toc221697433)

[1. プログラムの目的 2](#_Toc221697434)

[2. 原理 2](#_Toc221697435)

[2.1. van der Pauwの基本式 2](#_Toc221697436)

[2.2. 補正因子 の導入とプログラム中の式 2](#_Toc221697437)

[2.3. Newton法による数値計算 3](#_Toc221697438)

[2.4. 継続法による安定化 4](#_Toc221697439)

[3. 使い方 (Usage) 4](#_Toc221697440)

[3.1. 必要なライブラリ 4](#_Toc221697441)

[3.2. プログラムの実行 5](#_Toc221697442)

[4. 出力ファイル 5](#_Toc221697443)

[5. 使用例 5](#_Toc221697444)

[5.1. 出力ファイル vdp\_f\_results.csv の例 5](#_Toc221697445)

[5.2. 出力画像 vdp\_f\_plot.png の概要 6](#_Toc221697446)

このマニュアルは、Pythonプログラム vdPauw\_f.py の機能、原理、使用方法について詳細に説明します。

# vdPauw\_f.py マニュアル

## 1. プログラムの目的

本プログラム vdPauw\_f.py は、van der Pauw法における抵抗率の補正因子 を、抵抗比 の関数として数値的に計算し、その結果をプロットおよびデータファイルとして出力します。これにより、van der Pauw法で測定された抵抗値から正確なシート抵抗を求めるために必要な補正因子を、広範囲の抵抗比に対して得ることができます。

## 2. 原理

### 2.1. van der Pauwの基本式

van der Pauwの測定法では、薄膜サンプルのシート抵抗 は、直交する方向で測定された二つの抵抗値 と から以下の関係式によって求められます。

### 2.2. 補正因子 の導入とプログラム中の式

本プログラムでは、補正因子 を導入し、シート抵抗 を以下のように定義します。

ここで、抵抗比 を以下のように定義します。

この定義から、 となり、 と書けます。 上記の の定義式に と の関係を代入すると、

となります。 これを van der Pauwの基本式に代入するために、 と の項を計算します。

これらを van der Pauwの基本式に代入すると、

整理すると、以下の と の関係式が得られます。

本プログラムの vdp\_residual 関数は、この方程式を となる形で表現しています。

### 2.3. Newton法による数値計算

上記の方程式は解析的に について解くことが困難であるため、本プログラムではニュートン法 (Newton-Raphson method) を用いて数値的に解を求めます。ニュートン法は、関数 の根を探索する反復法であり、初期値 と関数の導関数 を与えることで、次の近似解 を計算し、高速かつ安定した収束が期待できます。

プログラムでは、ニュートン法の反復計算の安定性と速度を向上させるため、以下の導関数 vdp\_derivative を使用しています。関数を とすると、

および と置くと、

に関する導関数 は以下のようになります。

これはプログラムの vdp\_derivative 関数に実装されている式と一致します。

### 2.4. 継続法による安定化

広範囲の 値に対して安定して解を得るために、「継続法 (continuation method)」を採用しています。これは、 のときに という既知の解からスタートし、 の値を少しずつ変化させながら、直前の で得られた の解を次の計算の初期値として利用することで、ニュートン法の収束を助ける手法です。これにより、ニュートン法が初期値に強く依存することによる収束不良を防ぎます。

## 3. 使い方 (Usage)

### 3.1. 必要なライブラリ

本プログラムを実行するには、以下のPythonライブラリが必要です。

* numpy: 数値計算
* matplotlib: グラフ描画
* scipy: 最適化 (ニュートン法)
* pandas: データ処理、CSV出力

これらのライブラリがインストールされていない場合は、以下のコマンドをターミナルまたはコマンドプロンプトで実行してインストールしてください。

pip install numpy matplotlib scipy pandas

### 3.2. プログラムの実行

プログラムを実行するには、ターミナルまたはコマンドプロンプトで以下のコマンドを入力します。

python vdPauw\_f.py

プログラムは自動的に の値を 1 から 1000 まで対数スケールで300点生成し、それぞれの に対する の値を計算します。計算結果はCSVファイルに保存され、またグラフとして描画・保存されます。

## 4. 出力ファイル

プログラムを実行すると、以下のファイルが生成されます。

* vdp\_f\_results.csv: 計算された と の値を含むデータファイル。
* vdp\_f\_plot.png: 対 の関係をプロットした画像ファイル。

## 5. 使用例

コマンドプロンプトで python vdPauw\_f.py を実行すると、以下のようなメッセージが表示され、ファイルが生成されます。

✅ 計算結果を 'vdp\_f\_results.csv' に保存しました。  
✅ プロット図を 'vdp\_f\_plot.png' に保存しました。

### 5.1. 出力ファイル vdp\_f\_results.csv の例

このCSVファイルには、抵抗比 とそれに対応する補正係数 の値が格納されています。

r,f  
1.0,1.0  
1.0234771746200213,0.9996020584283896  
1.0475252876101918,0.9992047879105423  
...  
976.216398934335,0.08182963032545802  
1000.0,0.08051662912444654

### 5.2. 出力画像 vdp\_f\_plot.png の概要

生成されるプロット画像は、横軸に対数スケールで抵抗比 を、縦軸に補正係数 をプロットしたグラフです。グラフには以下の特徴があります。

* のとき となる理想的なケースが赤色の点線で示されます。
* の増加に伴い、補正係数 が徐々に減少していく様子が確認できます。
* グラフのタイトルは “van der Pauw Correction Factor (Newton Method)” です。
* 軸ラベルはそれぞれ “Resistance Ratio ” と “Correction Factor ” です。

このプロットにより、異なる抵抗比における補正因子 の挙動を視覚的に理解することができます。