

令和 元年度 卒業論文

**OpenModelica のシミュレーション結果を
用いたモータ特性表自動生成ツールの試作**

指導教員 片山 徹郎 教授

宮崎大学 工学部 情報システム工学科

原田 海人

2020 年 2 月

目次

1	はじめに	1
2	研究の準備	2
2.1	Modelica 言語	2
2.2	OpenModelica	2
2.3	ブラシ付き DC モータ	4
2.4	対応するモデル	5
2.4.1	ブラシ付き DC モータの Modelica モデル	5
2.4.2	ブラシ付き DC モータの Modelica モデルをサブシステムとするモデル	7
2.5	モータ特性表	7
2.5.1	特性表	8
2.5.2	特性グラフ	10
2.6	Python	10
3	モータ特性表自動生成ツール	12
3.1	csv ファイル解析部	13
3.1.1	実行コマンドの取得	13
3.1.2	csv ファイルの読み込み	15
3.2	特性表の要素算出部	17
3.2.1	基礎データの算出	18
3.2.2	特性表の各要素の算出	18
3.3	モータ特性表生成部	20
3.3.1	特性表生成	20
3.3.2	特性グラフ生成	20
3.3.3	モータ特性表生成	22
4	適用例	26
4.1	ブラシ付き DC モータの Modelica モデル	26

4.1.1	特性表の確認	30
4.1.2	特性グラフの確認	31
4.2	ブラシ付き DC モータの Modelica モデルをサブシステムとするモデル	31
5	考察	32
5.1	評価	32
5.1.1	評価方法	32
5.1.2	結果	32
5.2	関連研究	32
5.3	ツールの問題点	32
6	おわりに	33
	謝辞	34
	参考文献	35

第 1 章

はじめに

近年、モータは、エアコン・洗濯機・掃除機などの家電製品をはじめ、自動車関係、医療関係など様々な分野に用いられており [1]、社会に必要不可欠な存在となっている。

～はじめに 流れ 案 1～

シミュレーションを行った場合、期待通りか確認する。

確認する際は、シミュレーション結果から目的のグラフや値を計算等して作成しなければならない。

今回試作したツールで、グラフや値を作成する手間を省くことで、モータ開発の効率化を図る。

本論文の構成は、以下の通りである。

第 2 章では、モータ特性表自動生成ツールを試作するために必要となる前提知識について説明する。

第 3 章では、試作したモータ特性表自動生成ツールの機能について説明する。

第 4 章では、試作したモータ特性表自動生成ツールが正しく動作することを検証する。

第 5 章では、試作したモータ特性表自動生成ツールについて考察する。

第 6 章では、本論文のまとめとの課題を述べる。

第 2 章

研究の準備

本章では、本研究で必要となる前提知識を説明する。

2.1 Modelica 言語

Modelica 言語とは、微分代数方程式を用いた、複合領域のマルチドメインモデリングのために開発されたオブジェクト指向言語である [2]。その言語仕様は、非営利団体の Modelica Association が策定している。Modelica Association では、Modelica 言語による様々な物理領域のモデルライブラリを開発しており、数学、機械、電気、熱、流体、制御系、状態遷移機械などを含んだフリーの Modelica 標準ライブラリ (Modelica Standard Library : MSL) をリリースしている [2]。

2.2 OpenModelica

OpenModelica とは、Open Source Modelica Consortium (OSMC) が開発している Modelica コードのモデリング、シミュレーション、デバッグのための機能などを持つオープンソースプラットフォームである [3]。OpenModelica では、グラフィカルにモデリングすることが可能であり、専用のグラフィカルモデルエディタが用意されている。グラフィカルモデルエディタの外観を、図 2.1 に示す。また、グラフィカルモデルエディタ上にモデルを配置する際、オブジェクト名の確認をポップアップウィンドウで行う。ポップアップウィンドウの例を、図 2.2 に示す。OpenModelica は、シミュレーション結果をグラフとして画面上に描画できる。また、シミュレーション結果は、以下の 3 つのファイル形式で保存できる。

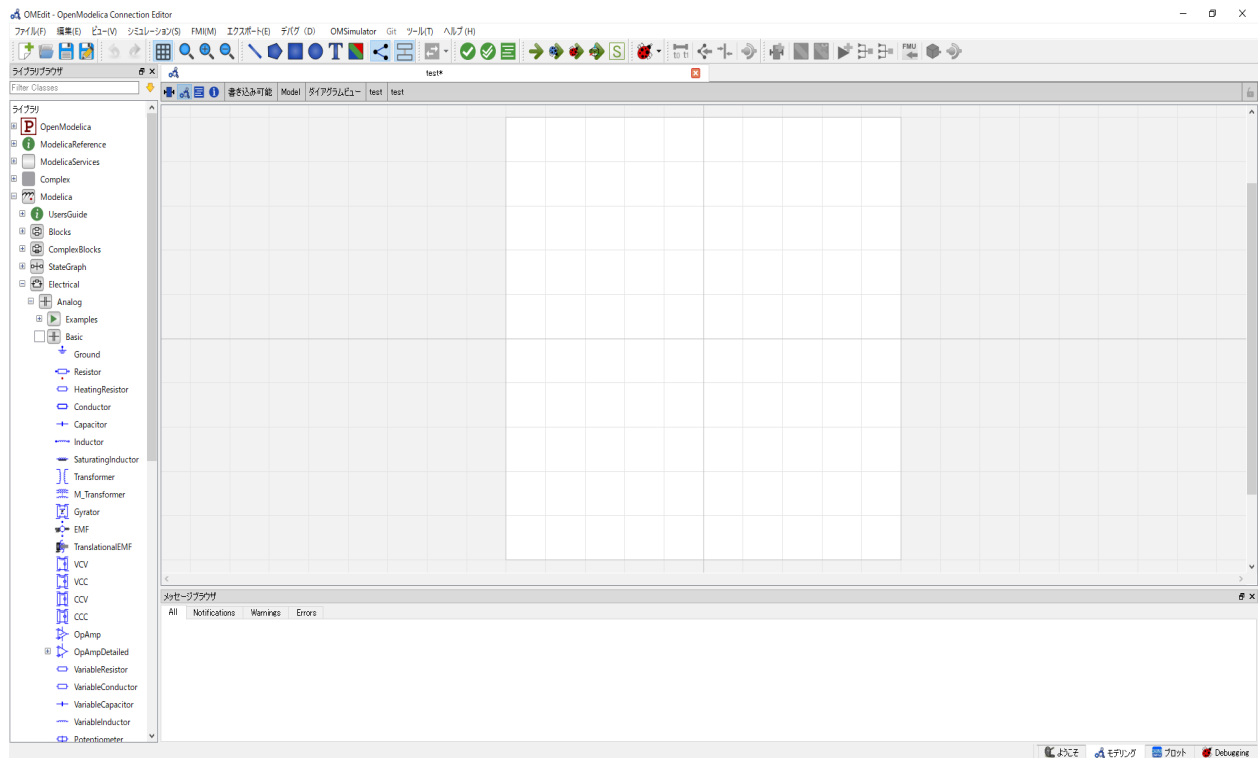


図 2.1: グラフィカルモデルエディタの外観

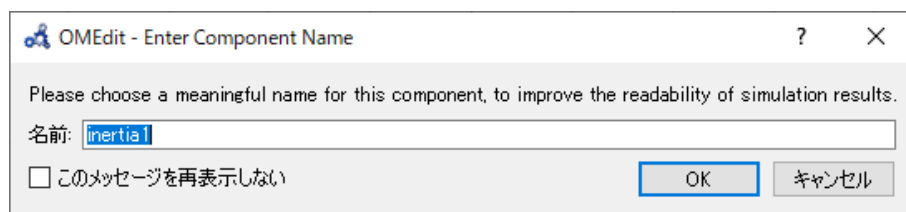


図 2.2: オブジェクト名の確認のポップアップウィンドウ

- mat ファイル
- plt ファイル
- csv ファイル

今回試作するモータ特性表自動生成ツールでは、csv ファイルにのみ対応する。

OpenModelica から出力される csv ファイルの一部を、図 2.3 に示す。

図 2.3 に示す csv ファイルの 1 列目には、時間を表すデータが必ず入る。そして 1 列目の「time」を除いた 1 行目には、オブジェクト名を含んだ変数名が必ず入る。また、2 行目には、時刻が「0.0 秒」の時の値が必ず入る。

図 2.3: シミュレーション結果の csv ファイルの一部

2.3 ブラシ付き DC モータ

宮崎大学 工学部 情報システム工学科

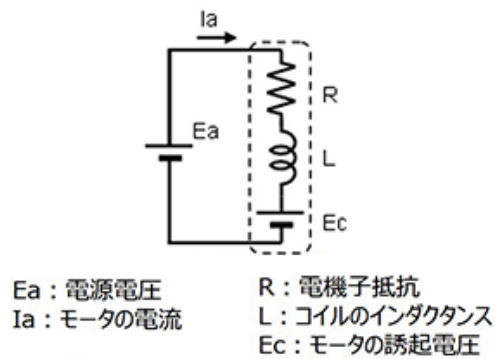


図 2.4: ブラシ付き DC モータの等価回路

2.4 対応するモデル

試作するモータ特性表自動生成ツールでは、以下に示す 2 つの Modelica モデルのシミュレーション結果に対応する。

- ブラシ付き DC モータの Modelica モデル
- ブラシ付き DC モータの Modelica モデルをサブシステムとするモデル

以下にて、それぞれについて説明する。

2.4.1 ブラシ付き DC モータの Modelica モデル

ブラシ付き DC モータの Modelica モデルとは、ブラシ付き DC モータの等価回路 [6] を Modelica 言語で表したモデルのことである。ブラシ付き DC モータの等価回路を、図 2.4 に示す。ブラシ付き DC モータの等価回路を Modelica 言語で表すためには、電源部品、抵抗部品、インダクタ部品、起電力部品、慣性部品、接地部品が必要である。

また、電源部品、抵抗部品、インダクタ部品、起電力部品、慣性部品には、それぞれ以下のパラメータを設定しなければならない。

- 電源部品 . . . 電圧値 (V)
- 抵抗部品 . . . 抵抗値 (Ω)

表 2.1: MSL 対応表

部品名	使用する MSL
電源部品	Modelica.Electrical.Analog.Sources
抵抗部品	Modelica.Electrical.Analog.Basic
インダクタ部品	Modelica.Electrical.Analog.Basic
起電力部品	Modelica.Electrical.Analog.Basic
慣性部品	Modelica.Mechanics.Rotational.Components
接地部品	Modelica.Electrical.Analog.Basic

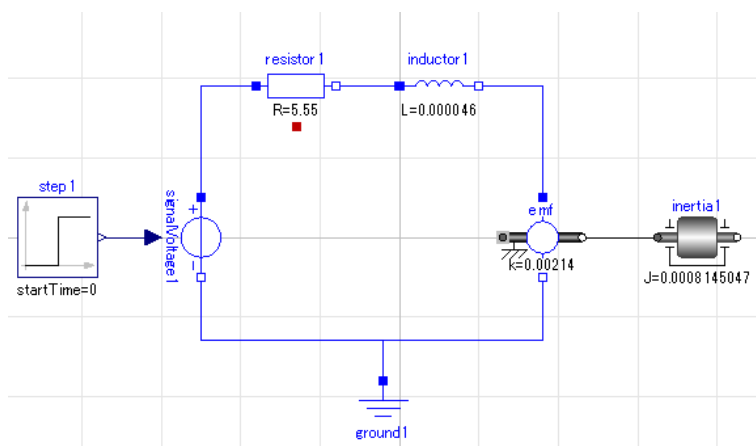


図 2.5: ブラシ付き DC モータの Modelica モデルの例

- インダクタ部品・・・インダクタンス値 (H)
- 起電力部品・・・トルク定数 (N・m/A)
- 慣性部品・・・慣性モーメント (kg・m²)

各部品で使用する MSL を表 2.1 に、ブラシ付き DC モータの Modelica モデルの例を図 2.5 に、それぞれ示す。

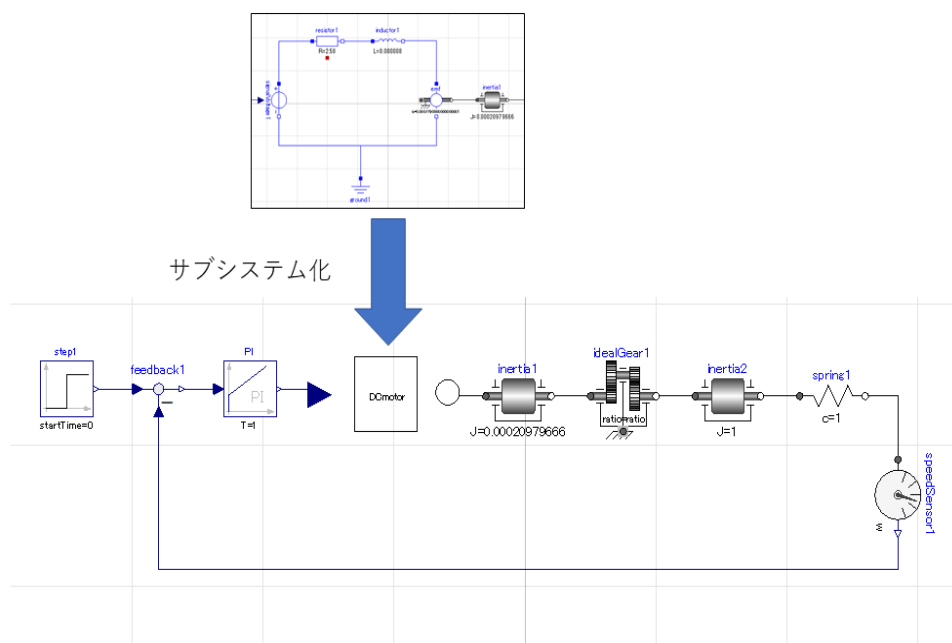


図 2.6: DC モータサーボのモデル

2.4.2 ブラシ付き DC モータの Modelica モデルをサブシステムとするモデル

ブラシ付き DC モータの Modelica モデルをサブシステムとするモデルとは、2.4.1 節で説明したブラシ付き DC モータの Modelica モデルを 1 つのサブシステムとして扱い、他の部品と合わせたモデルのことである。

例として、ブラシ付き DC モータのサブシステムを用いた DC サーボモータのモデルを、図 2.6 に示す。

2.5 モータ特性表

モータ特性表とは、モータを選定する際に、参考にする資料である [7]。一般的に決まった形式はなく、企業によって掲載する要素は異なるため、10 社のモータ特性表 [8, 9, 10, 11, 12, 13, 14, 15, 16, 17] に書かれている要素を集計した。集計結果を表 2.2 に示す。その中でも出現回数が多く、ブラシ付き DC モータのシミュレーション結果から作成できる要素とグラフを、今回自動生成するモータ特性表の要素とした。以下に、モータ特性表の構成を示す。

表 2.2: 集計結果

要素名	出現回数	要素名	出現回数	要素名	出現回数	要素名	出現回数
定格電流	9	効率	6	回転数	3	容量	1
電圧	8	枠番	4	極数	2	特記事項	1
出力	7	電流	4	回転方向	2	IE コード	1
始動トルク	7	停動トルク	4	無負荷回転数	2	定格電圧	1
始動電流	7	力率	4	無負荷電流	2	停動電流	1
定格回転数	7	最大トルク	4	定格出力	2	ブラシ	1
周波数	6	定格トルク	3	慣性モーメント	1	ノイズ素子	1

- モータ特性表

- － 特性表

- － 特性グラフ

以下にて、モータ特性表を構成する、特性表と特性グラフの内容について述べる。

2.5.1 特性表

特性表とは、以下の 9 つの要素で構成する。

電圧

電圧とは、シミュレーション時に、回路に印加された電圧値を表す。

単位は、V(ボルト)である。

始動電流

始動電流とは、モータの起動時に流れる電流値を表す。

単位は、mA(ミリアンペア)である。

停動トルク

停動トルクとは、モータが出しうる最大トルクで、このトルク以上の負荷がかかれば、モータが停止する値を表す。

単位は、 $mN \cdot m$ (ミリニュートンメートル) である。

最大効率

効率とは、入力電力に対する機械出力の比を百分率 [%] で表したものであり、最大効率は、その中で最大値を表す。

単位は、%(パーセント) である。

定格トルク

定格トルクとは、最大効率時のトルク値を表す。

単位は、 $mN \cdot m$ (ミリニュートンメートル) である。

定格回転数

定格回転数とは、最大効率時の回転数値を表す。

単位は、rpm(アールピーエム) である。

定格電流

定格電流とは、最大効率時の電流値を表す。

単位は、mA(ミリアンペア) である。

定格出力

定格出力とは、最大効率時の出力値を表す。

単位は、W(ワット) である。

最大回転数

最大回転数とは、回転数値の中で最大値を表す。

単位は、rpm(アールピーエム) である。

2.5.2 特性グラフ

特性グラフは、以下 4 つのグラフで構成する。

「トルク × 電流」グラフ

「トルク × 電流」グラフとは、横軸が「トルク $mN \cdot m$ 」、縦軸が「電流 mA」のグラフである。このグラフでは、トルクに対する電流の変化量を表している。

「トルク × 回転数」グラフ

「トルク × 回転数」グラフとは、横軸が「トルク $mN \cdot m$ 」、縦軸が「回転数 rpm」のグラフである。このグラフでは、トルクに対する回転数の変化量を表している。

「トルク × 効率」グラフ

「トルク × 効率」グラフとは、横軸が「トルク $mN \cdot m$ 」、縦軸が「効率 %」のグラフである。このグラフでは、トルクに対する効率の変化量を表している。

「トルク × 出力」グラフ

「トルク × 出力」グラフとは、横軸が「トルク $mN \cdot m$ 」、縦軸が「出力 W」のグラフである。このグラフでは、トルクに対する出力の変化量を表している。

2.6 Python

Python は、1991 年にオランダ人のガイド・ヴァンロッサムによって開発され、オープンソースで運営されている動的プログラミング言語である [18]。Python の用途は様々で、組み込み開発

表 2.3: 使用する Python のライブラリ

ライブラリ	役割
csv	csv ファイルの操作
math	数学関数の使用
matplotlib	グラフ描画
numpy	配列の計算
decimal	指数表記の計算
reportlab	PDF ファイル作成
PIL	画像処理
pdf2image	PDF ファイルの画像化
sys	スクリプトの起動パラメータを取得
os	不要なファイルを削除

や、Web アプリケーション、デスクトップアプリケーション、さらには人工知能開発、ビッグデータ解析などと多岐に渡る [19]。Python のプログラミング言語としての主な特徴は、少ないコードで簡潔にプログラムを書けること、専門的なライブラリが豊富にあることが挙げられる。

今回試作するモータ特性表自動生成ツールの開発言語には、Python を用いる。また、使用する Python のライブラリを、表 2.3 に示す。

第 3 章

モータ特性表自動生成ツール

本章では、本研究で試作したモータ特性表自動生成ツールについて説明する。モータ特性表自動生成ツールは、モータのシミュレーション結果から、2.5 節で述べたモータ特性表を自動生成する。モータ特性表自動生成ツールの処理の流れを、図 3.1 に示す。モータ特性表自動生成ツールの入力は、モータに関してシミュレーションした OpenModelica から出力される csv ファイルである。ここで、現時点のツールは、実装上の都合により、入力となる csv ファイルは次の 3 つの制約をすべて満たす必要がある。

- モータのモデルがブラシ付き DC モータである
- モータの回路に印加する電圧値は一定である
- 0 秒からモータに入力を与える

モータ特性表自動生成ツールは、3 つの処理部で構成しており、それぞれ以下の処理を行う。

- csv ファイル解析部
 - 実行コマンドの取得
 - csv ファイルの読み込み
- 特性表の要素算出部
 - 基礎データの算出

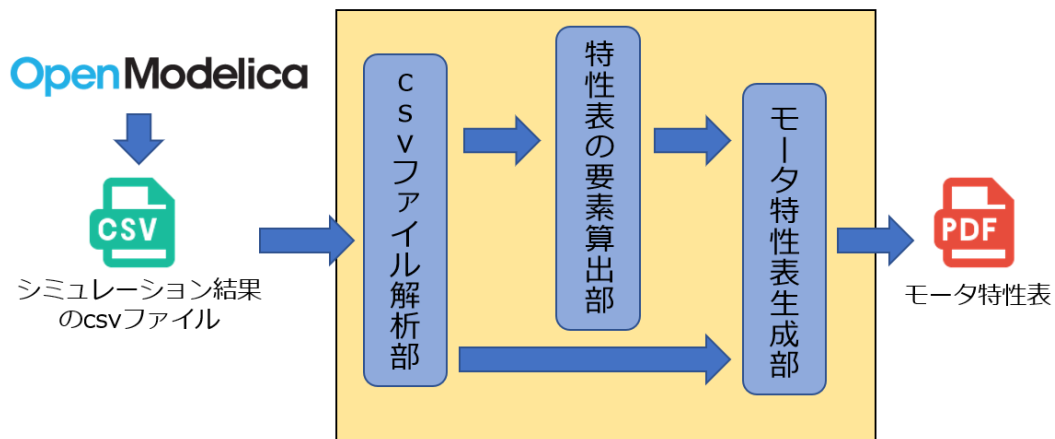


図 3.1: モータ特性表自動生成ツールの構造

- 特性表の構成要素の算出
- モータ特性表生成部
 - 特性表の生成
 - 特性グラフの生成
 - モータ特性表の生成

以下に、それぞれの処理部について説明する。

3.1 csv ファイル解析部

csv ファイル解析部では、モータ特性表自動生成ツールを実行する際のコマンドから、読み込む csv ファイルを決定する。そして、指定した csv ファイルを読み込み、モータ特性表を生成するために必要なデータを取得する。以下に、各処理について説明する。

3.1.1 実行コマンドの取得

読み込む csv ファイルを特定するために、モータ特性表自動生成ツールを実行するコマンドの引数に、ファイル名を指定する。モータ特性表自動生成ツールを実行するためのコマンドを、コード 3.1 に示す。なお、このコマンドは、ツールの実行ファイルが存在するディレクトリで実

コード 3.1: 実行コマンド

```
1 python characteristic.py 第1引数 第2引数 第3引数
```

```
$ python characteristic.py DCmotor_res.csv inertial  
ERROR : 引数の数が間違っています
```

図 3.2: 引数の数に誤りがあった場合のエラー文の例

行する必要がある。

第1引数には、入力とする csv ファイルのパスを含めたファイル名を指定する。

第2引数には、第1引数で指定した csv ファイルの中の、モータ特性表を自動生成したいモータのモデルに含まれる、慣性部品のオブジェクト名を指定する。

第3引数には、第1引数で指定した csv ファイルの中の、モータ特性表を自動生成したいモータのモデルに含まれる、電源部品のオブジェクト名を指定する。

第2引数と、第3引数に慣性部品と電源部品のオブジェクト名を指定する理由については、3.1.2 節で述べる。

引数を取得するために、コマンドの引数を、1次元の配列で保持する sys ライブラリの argv を使用する。以下に、処理の流れを示す。

1. argv の要素数を取得する
2. 要素数が4以外であれば、図 3.2 のエラーを表示し、全体の処理を終了する
3. argv の1番目の要素からファイル名を取得する
4. 取得したファイル名の拡張子が csv でない場合、図 3.3 のエラーを表示し、全体の処理を終了する
5. argv の2番目の要素から慣性部品のオブジェクト名を取得する
6. argv の3番目の要素から電源部品のオブジェクト名を取得する

```
$ python characteristic.py Dcmoto_res.csv inertia1 signalVoltage1  
ERROR : 指定したファイル名が間違っています
```

図 3.3: 第 1 引数に誤りがあった場合のエラー文の例

3.1.2 csv ファイルの読み込み

モータ特性表の要素を算出するために必要なデータを、csv ファイルから取得する。今回、モータ特性表の自動生成に必要なデータの導出計算式は、以下の通りである。

- 効率

$$\text{効率} = \frac{\text{出力}}{\text{入力}} \times 100 \quad (3.1)$$

- 出力

$$\text{出力} = \text{トルク} \times \text{角速度} \quad (3.2)$$

- 入力

$$\text{入力} = \text{電圧値} \times \text{電流値} \quad (3.3)$$

- 回転数

$$\text{回転数} = \frac{60 \times \text{角速度}}{2\pi} \quad (3.4)$$

- 定格出力

$$\text{定格出力} = \text{定格トルク} \times \text{定格回転数} \times \frac{2\pi}{60} \quad (3.5)$$

上記の式より、モータ特性表の要素を算出するために必要なデータは、トルク、角速度、電圧、電流であると言える。また、トルク、角速度、電圧、電流の値を持つ csv ファイル内の変数名を、表 3.1 に示す。

モータ特性表自動生成ツールで、csv ファイルを読み込むために、表 2.3 で挙げた csv ライブラリを使用する。csv ライブラリを用いた場合、csv ファイルを 1 行ごとに分けて読み込む。以下に、csv ファイルの読み込み処理の流れを示す。

表 3.1: 各値を持つ変数

必要なデータ	変数名
トルク値	(慣性部品のモジュール名).flange_a.tau
角速度値	(慣性部品のモジュール名).w
電圧値	(電源部品のモジュール名).p.v
電流値	(電源部品のモジュール名).n.i

1. csv ファイルを読み込み専用で開く
2. ファイルが開けなかった場合、図 3.3 のエラーを表示し、全体の処理を終了する
3. csv ファイルの行数分、以下の処理を繰り返す
 - (a) csv ファイルから取り出した行を、csv ファイルの行の値を保持する配列 `row` に格納する。
 - (b) csv ファイルの 1 行目を読み込んでいる場合、配列 `row` の要素数分、以下の処理を繰り返す
 - i. 変数名に、慣性部品のオブジェクト名が含まれている場合、以下の処理を行う
 - A. 変数名の末尾に、「.flange_a.tau」が含まれている場合、その変数名を格納している配列 `row` のインデックスを取得する
 - B. 変数名の末尾に、「.w」が含まれている場合、その変数名を格納している配列 `row` のインデックスを取得する
 - ii. 変数名に、電源部品のオブジェクト名が含まれている場合、以下の処理を行う
 - A. 変数名の末尾に、「.p.v」が含まれている場合、その変数名を格納している配列 `row` のインデックスを取得する
 - B. 変数名の末尾に、「.n.i」が含まれている場合、その変数名を格納している配列 `row` のインデックスを取得する
 - iii. 3.(a) で配列 `row` のインデックスを 4 つ取得するが、いずれか 1 つでも取得できない場合、図 3.4 のエラーを表示し、全体の処理を終了する

```
$ python characteristic.py DCmotor_res.csv inertia1 signalvoltage1  
ERROR : 指定したオブジェクト名が間違っています
```

図 3.4: 第 2 引数、第 3 引数に誤りがあった場合のエラー文の例

(c) csv ファイルの 2 行目を読み込んでいる場合、以下の処理を行う

- (a) トルク値の 0.0 秒段階の値を変数 `torque_default` に代入する
- (b) 角速度値の 0.0 秒段階の値を変数 `angularvelocity_default` に代入する
- (c) 電圧値の 0.0 秒段階の値を変数 `volyage_default` に代入する
- (d) 電流値の 0.0 秒段階の値を変数 `current_default` に代入する

4. csv ファイルの 3 行目以下を読み込んでいる場合、以下の処理を行う

- (a) トルク値を格納する配列 `torque` に、3.(a).i.A. で取得した配列 `row` のインデックスをキーに持つ値を格納する
- (b) 角速度値を格納する配列 `angularvelocity` に、3.(a).i.B. で取得した配列 `row` のインデックスをキーに持つ値を格納する
- (c) 電圧値を格納する配列 `voltage` に、3.(a).ii.A. で取得した配列 `row` のインデックスをキーに持つ値を格納する
- (d) 電流値を格納する配列 `current` に、3.(a).ii.B. で取得した配列 `row` のインデックスをキーに持つ値を格納する

3.2 特性表の要素算出部

特性表の要素算出部では、3.1 節で取得したデータをもとに、特性表の各要素を算出する。まず、特性表の各要素を算出するために必要となる基礎データを算出する。基礎データとは、回転数、出力、効率の値のことを指す。そして、3.1 節で取得したデータと、基礎データから特性表の各要素を算出する。以下より、各処理について説明する。

3.2.1 基礎データの算出

基礎データの算出処理では、回転数、出力、効率の値を持つ配列を、それぞれの値に対して生成する。生成方法を以下に示す。

回転数

回転数を算出する式は、(3.4) 式を用いる。

配列 `angularvelocity` から、配列の要素数分、繰り返し処理で、回転数の値を持つ配列 `speed` を生成する。

出力

出力を算出する式は、(3.2) 式を用いる。配列 `torque` と、配列 `angularvelocity` から、配列の要素数分、繰り返し処理で、出力の値を持つ配列 `output` を生成する。

効率

効率を算出する式は、(3.1) 式と、(3.3) 式を用いる。まず、配列 `voltage` と、配列 `current` から、(3.3) 式を用いて入力値を求める。そして、配列 `output` と、求めた入力値から、配列の要素数分、繰り返し処理で、効率の値を持つ配列 `efficiency` を生成する。

3.2.2 特性表の各要素の算出

特性表の各要素の算出処理では、2.5.1 節で述べた 9 つの要素を算出する。それぞれの算出方法を以下に示す。

電圧

モータ特性表自動生成ツールが対応するモデルでは、電圧値が一定のため、配列 `voltage` の要素はすべて同じ値になる。今回は、0 番目の値を電圧とする。

始動電流

始動電流とは、モータの起動時に流れる大きな電流であり、モータが起動した後は逆起電力が発生するため、モータ・コイル部分にかかる電圧が下がり、電流値も下がる。したがって、配列 `current` の要素の最大値を始動電流とする。

停動トルク

停動トルクとは、モータが出しうる最大のトルク値である。したがって、配列 `torque` の最大値を停動トルクとする。

最大効率

配列 `efficiency` の最大値を最大効率とする。

定格トルク

定格トルクとは、最大効率時のトルク値である。まず、配列 `efficiency` の中で、最大効率である要素のインデックスを取得する。そして、配列 `torque` の中で、取得した最大効率のインデックスと同じ位置にある値を定格トルクとする。

定格回転数

定格回転数とは、最大効率時の回転数の値である。配列 `speed` の中で、取得した最大効率のインデックスと、同じ位置にある値を定格回転数とする。

定格電流

定格電流とは、最大効率時の電流値である。配列 `current` の中で、取得した最大効率のインデックスと、同じ位置にある値を定格電流とする。

定格出力

定格出力とは、定格動作点における出力の値である。定格出力を算出する式は、(3.5) 式を用いる。上記の定格トルクと定格回転数で求めた値を、定格出力を算出する式に代入し、得た値が定格出力である。

最大回転数

配列 `speed` の最大値を最大回転数とする。

3.3 モータ特性表生成部

モータ特性表生成部では、3.1 節と 3.2 節で算出した要素を用いて、モータ特性表を自動生成する。まず、特性表を生成し、画像として保存する。次に、2.5.2 節で挙げた 4 つの特性グラフを生成し、画像として保存する。最後に、特性表の画像と、特性グラフの画像を PDF ファイルに書き込むことで、モータ特性表を自動生成する。以下に、各処理について説明する。

3.3.1 特性表生成

この処理では、3.2.2 節で作成した特性表の要素の配列を用いて、特性表を生成する。処理の流れを以下に示す。

1. 特性表の各要素の名前を持つ配列を作成する (コード 3.2 参照)
2. 特性表の要素の配列と、特性表の各要素の名前を持つ配列から、特性表用の配列を作成する
3. 特性表用の配列から特性表を生成し、生成した表の上部にタイトルとして「モータ特性表」の文字を追加し、画像として保存する。

上記の処理で生成した特性表の画像を、図 3.5 に示す。

3.3.2 特性グラフ生成

3.1.2 節、3.2.1 節で求めた要素の配列を用いて、特性グラフを生成する。処理の流れを以下に示す。

コード 3.2: 特性表を構成する要素名の配列

```
1 ["電圧 (V)", "始動電流 (mA)", "停動トルク (mN.m)", "最大効率 (%)", "定格トルク (mN.m)", "定格回転数 (rpm)", "定格電流 (mA)", "定格出力 (W)", "最大回転数 (rpm)"]
```

モータ特性表

電圧 (V)	2.4
始動電流 (mA)	431.9945756634572
停動トルク (mN.m)	0.9244683919197983
最大効率 (%)	99.89790026553919
定格トルク (mN.m)	0.0007704063898538723
定格回転数 (rpm)	10700.575772223889
定格電流 (mA)	0.3600029859130245
定格出力 (W)	0.0008632878742652782
最大回転数 (rpm)	10700.575772223889

図 3.5: タイトルと特性表の画像

1. 配列 torque に、torque_default を格納する
2. 配列 current に、current_default を格納する
3. 配列 speed に、angularvelocity_default から (3.4) 式を適用した値を格納する
4. current_default が 0 か判定し、以下の処理を行う
 - (a) 0 だった場合、配列 efficiency に、0 を格納する
 - (b) 0 ではない場合、以下の処理を行う
 - i. current_default と、voltage_default を用いて (3.3) 式から、0.0 秒段階の入力の値を保持する input_default に代入する
 - ii. torque_default と、angularvelocity_default を用いて (3.2) 式から、0.0 秒段階の入力の値を保持する output_default に代入する
 - iii. 配列 efficiency に、input_default と、output_default を (3.1) 式に適用した値を格納する
 - (c) 配列 output に、torque_default と、angularvelocity_default を (3.2) 式に適用した値を格納する

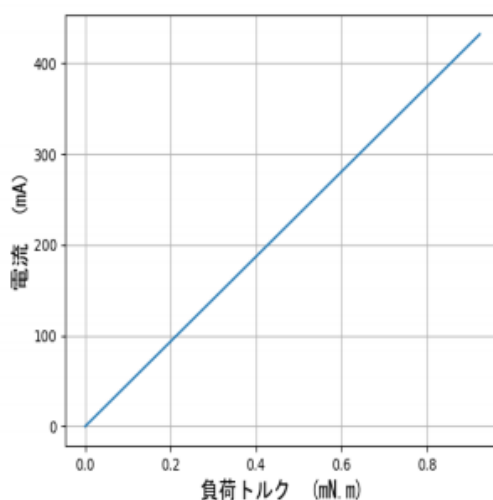


図 3.6: 「トルク × 電流」 グラフ

5. x 軸にトルク値を持つ配列を、y 軸に電流値を持つ配列を指定してグラフを生成し、画像として保存する
6. x 軸にトルク値を持つ配列を、y 軸に回転数値を持つ配列を指定してグラフを生成し、画像として保存する
7. x 軸にトルク値を持つ配列を、y 軸に効率値を持つ配列を指定してグラフを生成し、画像として保存する
8. x 軸にトルク値を持つ配列を、y 軸に出力値を持つ配列を指定してグラフを生成し、画像として保存する

上記の処理で生成した 4 つのグラフを、それぞれ図 3.6、図 3.7、図 3.8、図 3.9 に示す。

3.3.3 モータ特性表生成

3.3.1 節と、3.3.2 節で生成した合計 5 つの画像を、PDF ファイルに書き込み、モータ特性表を自動生成する。モータ特性表の PDF ファイルのファイル名は、「characteristicTable.pdf」で生成する。同じファイル名がある場合、上書き保存する。この処理で自動生成するモータ特性表を、図 3.10 に示す。

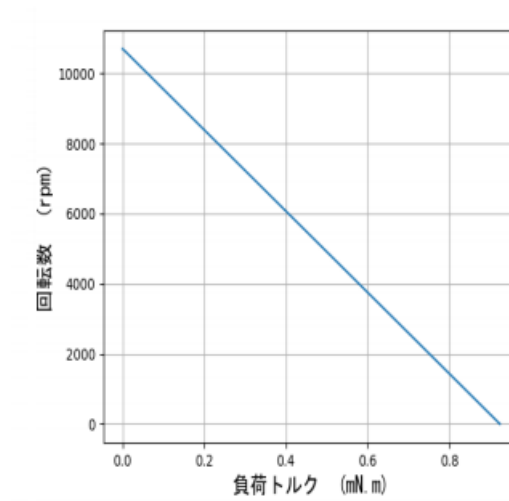


図 3.7: トルク × 回転数」グラフ

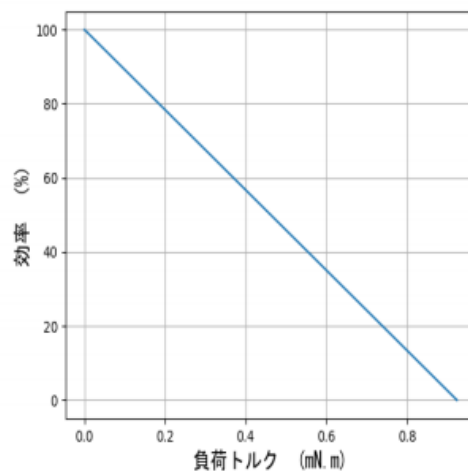


図 3.8: 「トルク × 効率」グラフ

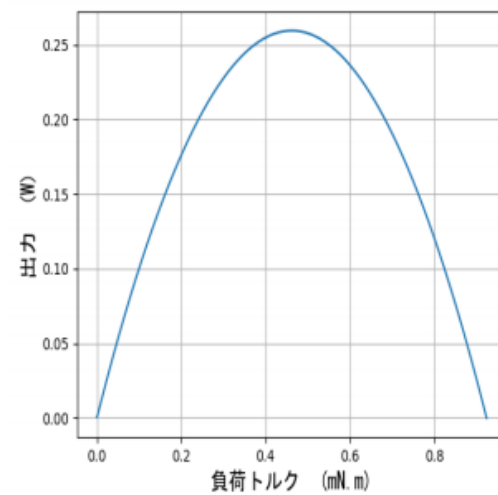


図 3.9: 「トルク × 出力」 グラフ

モータ特性表

電圧 (V)	2.4
始動電流 (mA)	431.9945756634572
停動トルク (m N.m)	0.9244683919197983
最大効率 (%)	99.89790026553919
定格トルク (m N.m)	0.0007704063898538723
定格回転数 (rpm)	10700.575772223889
定格電流 (mA)	0.3600029859130245
定格出力 (W)	0.0008632878742652782
最大回転数 (rpm)	10700.575772223889

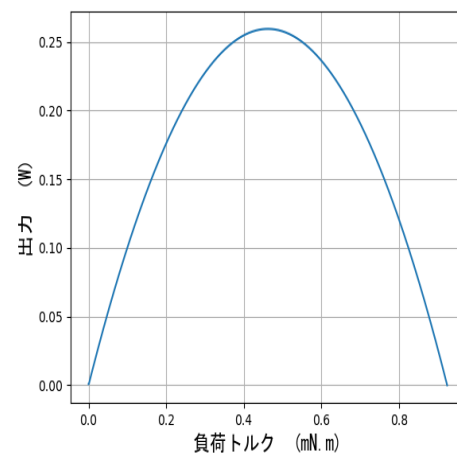
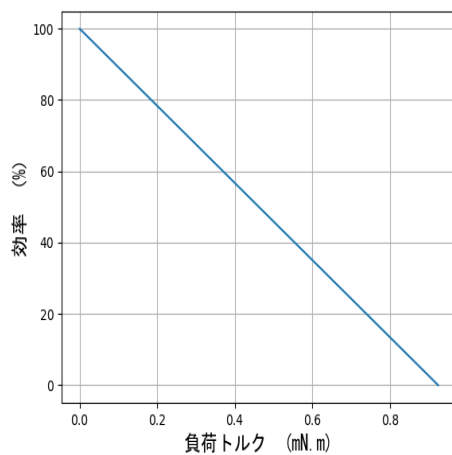
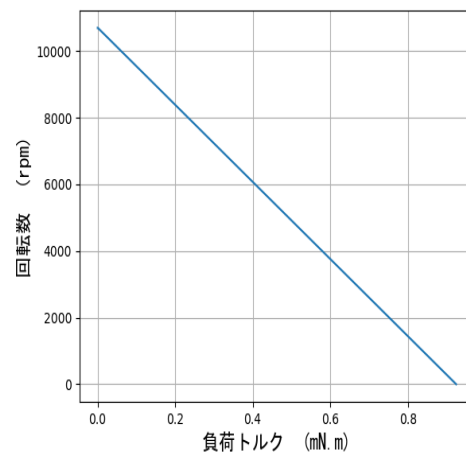
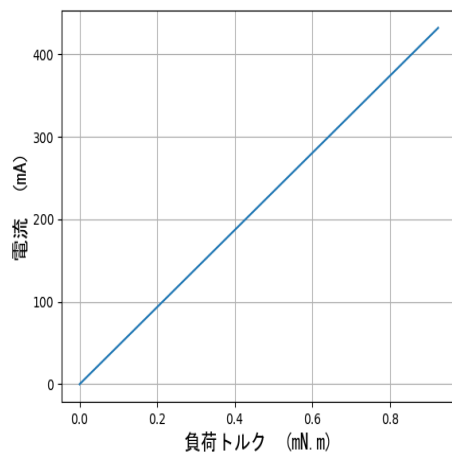


図 3.10: モータ特性表

第 4 章

適用例

本章では、本研究で作成したモータ特性表自動生成ツールが正しく動作することを確認するため、以下の 2 種類の Modelica モデルのシミュレーション結果の csv ファイルを適用する。

- ブラシ付き DC モータの Modelica モデル
- ブラシ付き DC モータの Modelica モデルをサブシステムとするモデル

具体的には、以下の 2 つの項目を確認する。

1. 生成したモータ特性表の各要素の値が、正しい値になっていること
2. 2 つのモデルから生成したモータ特性表の各要素が、同値になっていること

4.1 ブラシ付き DC モータの Modelica モデル

ブラシ付き DC モータの Modelica モデルから生成した csv ファイルを、モータ特性表自動生成ツールに適用する。適用例に用いるモデルを、図 4.1 に、図 4.3 のシミュレーション結果である csv ファイルの一部を、図 4.2 にそれぞれ示す。図 4.2 の csv ファイルをモータ特性表自動生成ツールに適用した結果、出力するモータ特性表を図 4.3 に示す。

inertia1.w	inertia1.flange_a.tau	signalVoltage1.n.i	signalVoltage1.p.v
0	0	0	9
106.2379105	46.12885568	4435.466893	9
199.4341076	40.46587394	3890.949417	9
281.1891208	35.49810528	3413.279354	9
352.9075185	31.1402024	2994.250231	9
415.8213942	27.31729825	2626.663293	9
471.0117012	23.96370968	2304.202854	9
519.4266309	21.02182073	2021.328916	9
561.8979184	18.44109176	1773.1819	9
599.155251	16.177184	1555.498462	9
631.8387142	14.19120323	1364.538772	9
660.5098152	12.44902998	1197.022114	9
685.6611266	10.92073354	1050.070532	9
707.7247559	9.580057262	921.1593521	9
727.0797599	8.403968134	808.073859	9
744.0586582	7.372260901	708.8712405	9
758.9531557	6.467210369	621.8471509	9
772.0191392	5.673267828	545.5065219	9
783.4810874	4.976793022	478.5377906	9
793.5359167	4.365820507	419.7904334	9
802.3563724	3.829853579	368.2551519	9
810.0939909	3.359684262	323.0465637	9
816.8817058	2.947234972	283.387978	9
822.8361321	2.585419701	248.5980481	9
828.0595667	2.268022469	218.0790835	9
832.6417393	1.989590939	191.3068211	9
836.6613385	1.745343666	167.8215063	9
840.1874723	1.531081369	147.2193624	9
843.2807017	1.343124113	129.1465494	9
845.9942023	1.178240728	113.2923777	9
848.3745867	1.033598846	99.38450439	9
850.462754	0.906713269	87.18396816	9
852.2945765	0.795404251	76.48117794	9
853.9015204	0.697759774	67.09228596	9
855.3111939	0.61210225	58.85598561	9
856.5478149	0.536960103	51.63077914	9
857.6326283	0.471042412	45.29253958	9
858.5842687	0.413216848	39.73238924	9
859.4190841	0.362490055	34.85481297	9
860.1514173	0.317990497	30.57600936	9
860.7938489	0.278953731	26.82247417	9
861.3574152	0.244709139	23.52972492	9
861.8517978	0.21466843	20.64119516	9
862.2854898	0.188315536	18.1072631	9
862.6659413	0.165197748	15.88439886	9
862.9996875	0.144917963	13.93441949	9
863.2924629	0.127127734	12.22382062	9
863.5492972	0.111521435	10.72321489	9
863.7746024	0.097830969	9.406823936	9
863.972249	0.085821152	8.252033823	9
864.1456324	0.075285669	7.239006641	9
864.1456324	0.075285669	7.239006641	9

図 4.2: 図 4.3 のシミュレーション結果である csv ファイルの一部

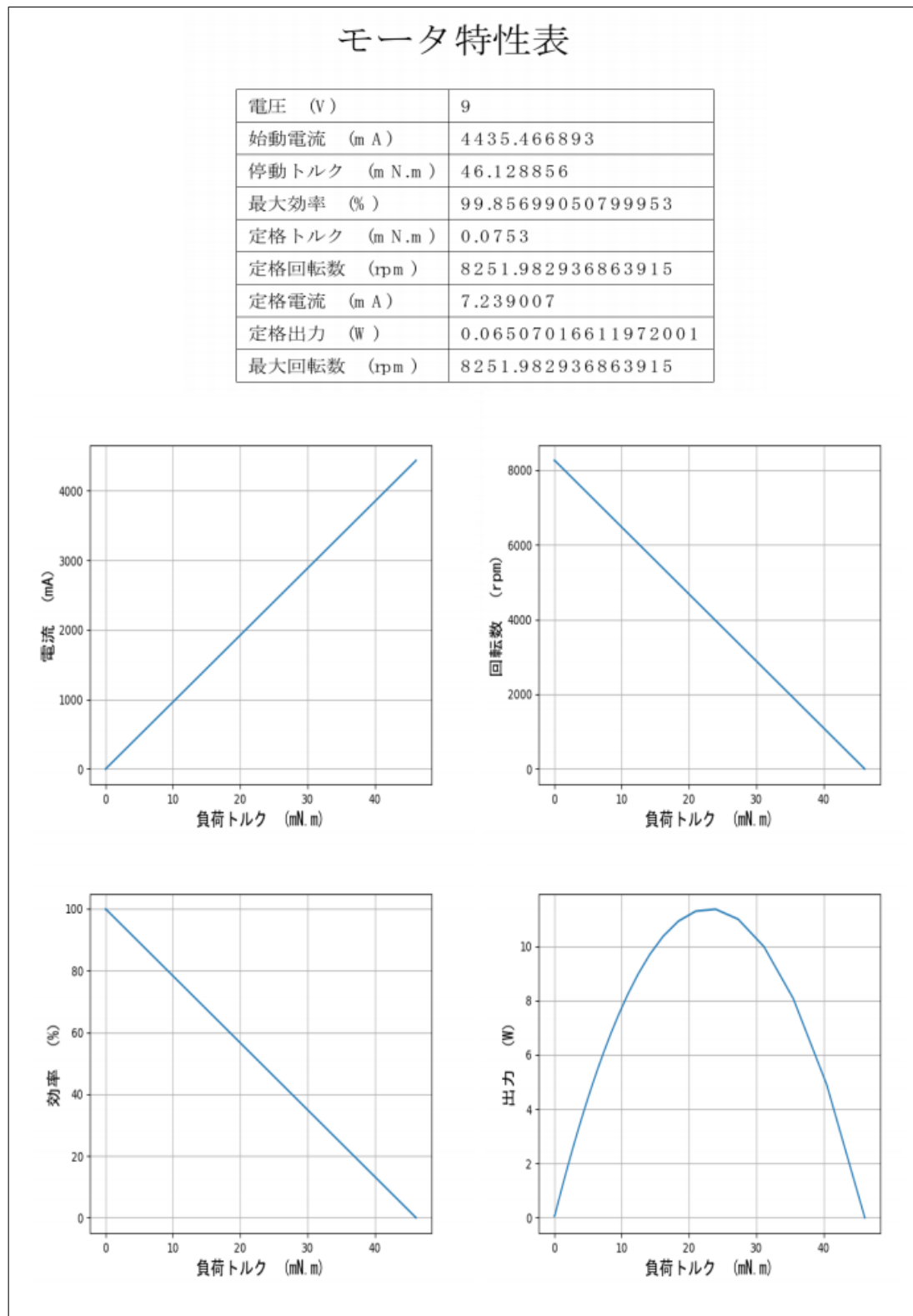


図 4.3: 適用例で生成したモータ特性表

4.1.1 特性表の確認

以下に、特性表で出力した各要素が正しく算出できているか確認する。

電圧

図 4.3 の特性表にある電圧の値と、図 4.2 の `signalVoltage1.p.v` が持つ値の中で、先頭にある値が同値である。よって、正しく出力していることが確認できる。

始動電流

図 4.3 の特性表にある始動電流の値と、図 4.2 の `signalVoltage1.n.i` が持つ値の中の最大値が同値である。よって、正しく出力していることが確認できる。

停動トルク

図 4.3 の特性表にある停動トルクの値と、図 4.2 の `inertial.flange_a.tau` が持つ値の中の最大値が同値である。よって、正しく出力していることが確認できる。

最大効率

図 4.3 の特性表にある最大効率の値と、図 4.2 の効率が持つ値の中の最大値が同値である。よって、正しく出力していることが確認できる。

定格トルク

図 4.3 の特性表にある定格トルクの値と、図 4.2 のトルクが持つ値の中の、最大効率を出した時の値が同値である。よって、正しく出力していることが確認できる。

定格回転数

図 4.3 の特性表にある定格回転数の値と、図 4.2 の回転数が持つ値の中の、最大効率を出した時の値が同値である。よって、正しく出力していることが確認できる。

定格電流

図 4.3 の特性表にある定格電流と、図 4.2 の電流が持つ値の中の、最大効率を出した時の値が同値である。よって、正しく出力していることが確認できる。

定格出力

図 4.3 の特性表にある定格出力と、図 4.2 の出力が持つ値の中の、最大効率を出した時の値が同値である。よって、正しく出力していることが確認できる。

最大回転数

図 4.3 の特性表にある最大回転数と、図 4.2 の回転数が持つ値の中の最大値が同値である。よって、正しく出力していることが確認できる。

4.1.2 特性グラフの確認

以下に、特性グラフが正しく生成できているか確認する。

4.2 ブラシ付き DC モータの Modelica モデルをサブシステムとするモデル

第 5 章

考察

本論文では、モータ特性表自動生成ツールを試作した。

5.1 評価

5.1.1 評価方法

5.1.2 結果

本論文で試作したモータ特性表自動生成ツールは、

5.2 関連研究

関連研究について述べる。

5.3 ツールの問題点

以下に、今回作成したモータ特性表自動生成ツールの問題点を示す。

- 対応するモータのモデルは 1 種類しかない
モータは～種類に分けることができ、今回は 1 つにしか対応していない。対応できる数を増やす必要がある。

第 6 章

おわりに

以下に、今後の課題を示す。

謝辞

参考文献

- [1] 日本電産株式会社. 身の回りのモータ. <https://www.nidec.com/jp/technology/scenes/>. Accessed: 2020-1-22.
- [2] 平野豊. Modelica によるモデルベースシステム開発入門. TechShare 社, 2017.
- [3] Peter Fritzson, Peter Aronsson, Adrian Pop, Hakan Lundvall, Kaj Nystrom, Levon Saldamli, David Broman, and Anders Sandholm. Openmodelica-a free open-source environment for system modeling, simulation, and teaching. In *2006 IEEE Conference on Computer Aided Control System Design, 2006 IEEE International Conference on Control Applications, 2006 IEEE International Symposium on Intelligent Control*, pp. 1588–1595. IEEE, 2006.
- [4] 制御工学の基礎あれこれ. ブラシ付き dc モータの仕組み. <http://arduino pid.web.fc2.com/K13.html>. Accessed: 2020-1-31.
- [5] 日本電産株式会社. ブラシ付き dc モータ. <https://www.nidec.com/jp/technology/motor/glossary/000/0604/>. Accessed: 2020-1-31.
- [6] Tech Web motor. ブラシ付き dc モータの特性. https://micro.rohm.com/jp/techweb_motor/knowledge/basics/basics-03/209. Accessed: 2020-1-23.
- [7] MekatoroNet. 仕様の見方. <http://www.mekatoro.net/digianaecatalog/orien-sougou/book/orien-sougou-P0042.pdf>. Accessed: 2020-1-30.
- [8] 三菱電機エンジニアリング株式会社. 1. 特長・標準仕様. http://www.mee.co.jp/sales/system-solution/machine/pdf/sales_fa_machine_kffk-mkb_kffk.pdf. Accessed: 2020-1-30.

- [9] 住友重機械ギヤモータ株式会社. モータ特性表. <https://cyclo.shi.co.jp/technical/pdf/tech032.pdf>. Accessed: 2020-1-30.
- [10] 株式会社ニッセイ. モータ特性表. <https://www.nissei-gtr.co.jp/wp-content/uploads/2017/05/faq-0124-01.pdf>. Accessed: 2020-1-30.
- [11] 株式会社日立産機システム. 日立三相モータ. <http://www.motor-shuri.com/new/images/hitachi.pdf>. Accessed: 2020-1-30.
- [12] MekatoroNet. モータ特性表. <http://www.mekatoro.net/digianaecatalog/sumit-sougou/Book/sumit-sougou-P0615.pdf>. Accessed: 2020-1-30.
- [13] 東芝産業機器システム株式会社. 低圧三相かご形誘導電動機プレミアムゴールドモートル. <https://ednjapan.com/edn/articles/1605/25/news042.html>. Accessed: 2020-1-30.
- [14] 相模マイクロ株式会社. 直流モータ特性表 dc motor characteristics. http://www.sagamimicro.co.jp/product/pdf_product/S-2230BA-24P0-384.pdf. Accessed: 2020-1-30.
- [15] 三菱電機株式会社. 三菱モータ (総合カタログ). http://dl.mitsubishielectric.co.jp/dl/fa/document/catalog/i_motor/1001040/101040.pdf. Accessed: 2020-1-30.
- [16] 富士電機株式会社. 富士低圧三相モータ. <https://www.chienkung.com.tw/upload/files/%E5%AF%8C%E5%A3%AB%E3%9A%B9%E6%95%88%E8%83%BD%E9%A6%AC%E9%81%94.pdf>. Accessed: 2020-1-30.
- [17] スリーピース株式会社. モータ単体 *外形図・特性表*. http://www.three-peace.com/cms/contents/data/3/4/CATALOG_4.pdf. Accessed: 2020-1-30.
- [18] python Japan. Python とは. <https://www.python.jp/pages/about.html>. Accessed: 2020-1-31.
- [19] Samurai Blog. Python とは? 言語の特徴から学習法まで初心者向けにわかりやすく解説. <https://www.sejuku.net/blog/7720>.