令和 元年度 卒業論文

OpenModelica のシミュレーション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

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～

モータの開発は～で、～の課題がある。それを解決する手段としてシミュレーションがある。シミュレーションツールの中に OpenModelica がある。OpenModelica は～で、～する。また、結果を画面所にプロットすることで結果を確認できる。シミュレーションを行った場合、期待通りか結果と比較する。

比較する際は、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。

しかし、OpenModelica ではグラフでしか確認できず、具体的な値を取得することが困難である。

そこで、本研究では、モータのシミュレーションを OpenModelica で行った際のシミュレーション結果の確認にかかる手間を削減することを目的として、シミュレーション結果の csv ファイルからモータ特性表を自動生成するツールを試作する。

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構成及び手順について説明する。

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。

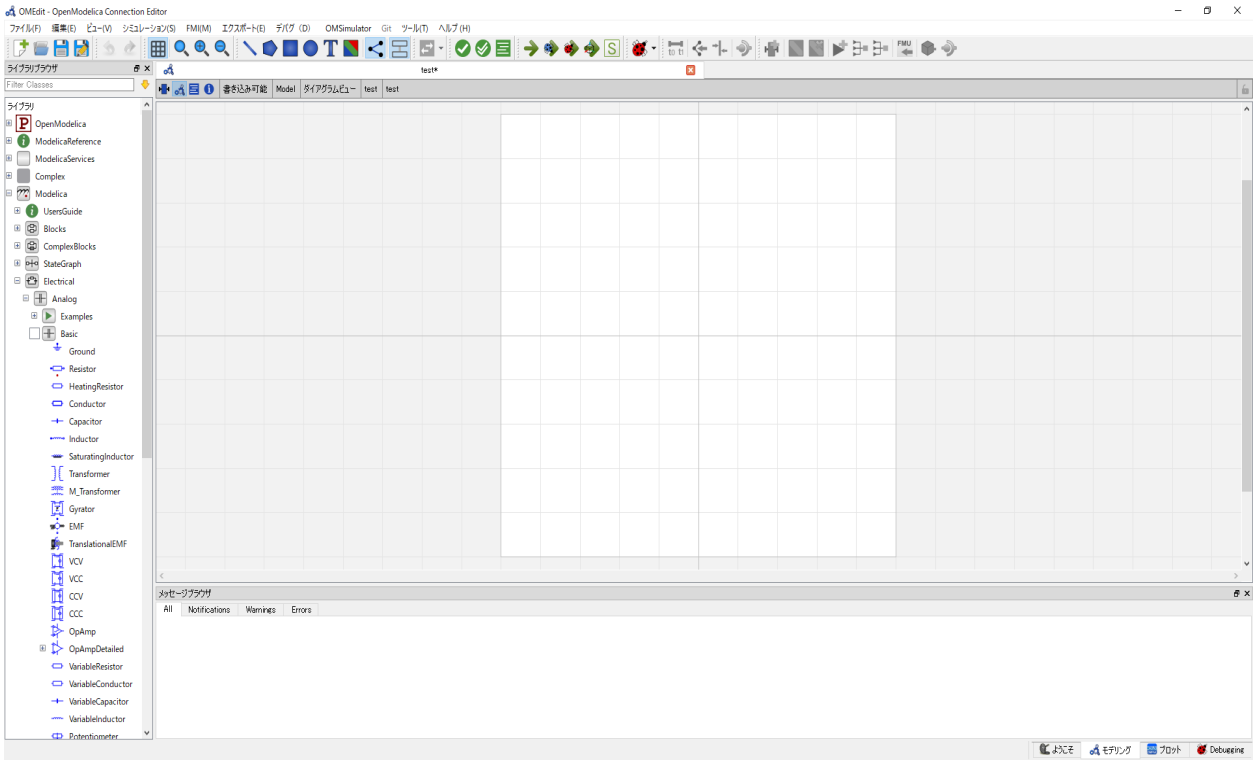

図 2.1: グラフィカルモデルエディタの外観

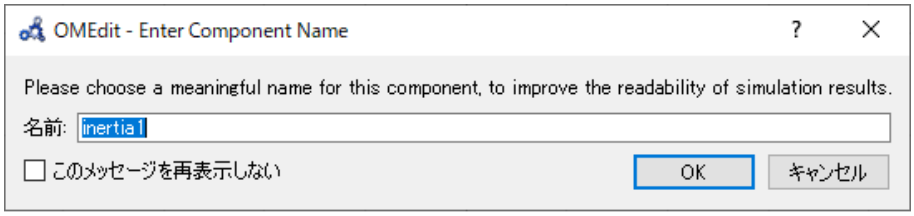

図 2.2: オブジェクト名の確認のポップアップウィンドウの例

- 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、「0.0 秒」から始まる時刻を表すデータが必ず入る。そして 1 行目には、1 列目の「time」を除いて、オブジェクト名を含んだ変数名が必ず入る。また以降には、1 列目の「time」における各変数の値が 2 列目以降にそれぞれ入る。

```
"time","inductor1.i","inertial.phi","inertial.w","der(inductor1.i)","der(inertial.phi)","der(inertia
0,0,0,0,122448.9795918368,0,0,0,0,-0,0,0,0,0,9,9,0,0,0,1.78,0,9,0,-0,0,-0,0,0,0,0,-0,0,0,0,0,0,0,-0,
0.1,5.047915163691204,0.0707464037147575,1.415104150827973,-0.08264678851114857,1.415104150827973,14
0.2,5.039657246826229,0.2829450187532758,2.828477427971375,-0.08251158624533513,2.828477427971375,14
0.3,5.031412839149343,0.6363654143980045,4.239538557176244,-0.08237660493127441,4.239538557176244,14
0.4,5.023181918532681,1.130776271214502,5.648291325710747,-0.08224184440823901,5.648291325710747,14.
0.5,5.014964462926303,1.765947084414186,7.054739507505285,-0.08210730441070999,7.054739507505285,14.
0.6,5.006760450312931,2.541647690661693,8.458886870899837,-0.08197298469436279,8.458886870899837,14.
0.7,4.998569858707249,3.45764826095431,9.860737178760534,-0.08183888845401014,9.860737178760534,14.0(
0.8,4.990392666186916,4.513719624349956,11.26029418317943,-0.08170500376628731,11.26029418317943,13.
0.9,4.982228850797614,5.70963227572398,12.65756164171997,-0.08157134205288172,12.65756164171997,13.9
1,4.974078390690659,7.045157814256005,14.05254329386543,-0.08143789896712425,14.05254329386543,13.93
1.1,4.965941263932,8.520066946679792,15.44524289371061,-0.08130467414554903,15.44524289371061,13.915
1.2,4.957817448793185,10.1341325290293,16.83566416016098,-0.08117166726641792,16.83566416016098,13.8
1.3,4.94970692345335,11.88712645122738,18.22381082793964,-0.08103887803442675,18.22381082793964,13.8
1.4,4.941609666177463,13.77882150055004,19.60968661707778,-0.0809063060255005,19.60968661707778,13.8
1.5,4.933525655273357,15.80899091233752,20.99329524027073,-0.0807739509069499,20.99329524027073,13.8
1.6,4.925454869029084,17.97740771513312,22.37464041359953,-0.0806418123056796,22.37464041359953,13.8
1.7,4.917397285872307,20.28384639697651,23.75372582924972,-0.08050988983726544,23.75372582924972,13.
1.8,4.909352884122416,22.72808031399924,25.13055519793894,-0.08037818322415165,25.13055519793894,13.
1.9,4.901321642311751,25.30988504848918,26.50513219393718,-0.08024669201469649,26.50513219393718,13.
2,4.893303538801044,28.02903438879071,27.8774605208857,-0.08011541590755371,27.8774605208857,13.712(
2.1,4.885298552177078,30.88530448629863,29.24754384373682,-0.07998435457041164,29.24754384373682,13.
2.2,4.877306660941829,33.87847060588226,30.61538584195743,-0.07985350765434308,30.61538584195743,13.
2.3,4.869327843661874,37.00830868772714,31.98099018395787,-0.07972287479153942,31.98099018395787,13.
2.4,4.861362078936959,40.27459501875122,33.34436053247161,-0.07959245568140967,33.34436053247161,13.
2.5,4.853409345433533,43.67710658319652,34.70550053881518,-0.07946224987080158,34.70550053881518,13.
```

図 2.3: シミュレーション結果の csv ファイルの一部

OpenModelica から出力される csv ファイルのファイル名は、「(シミュレーションしたモデルの名前)_res.csv」で生成される。例えば、シミュレーションしたモデルの名前が「hoge」だった場合、csv ファイルのファイル名は「hoge_res.csv」となる。

2.3 ブラシ付き DC モータ

DC モータとは、磁場の中にあるコイルに電流を流す事で発生するローレンツ力を回転方向に利用することで回すモータである [4]。また、ブラシ付き DC モータとは、モータ内部に「ブラシ」と呼ばれる電極と、「コミュテータ」と呼ばれる整流子を設け、その二つを接触させ機械的に電流の切り替えを行い、回転させる DC モータである。ブラシ付き DC モータは、シンプルな構造で、制御しやすく、汎用性が高く、模型用モータや自動車補機用モータなど世界で一番多く使われているモータである [5]。

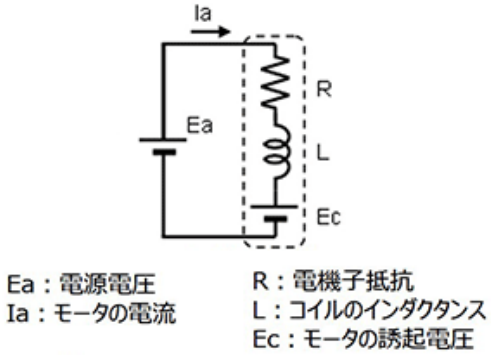

図 2.4: ブラシ付き DC モータの等価回路

2.4 対象とするモデル

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を対象とする。

- 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。また、ブラシ付き DC モータの Modelica モデルの例を図 2.5 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。各部品で使用する MSL を表 2.1 に示す。

- 電源部品 ・・・ 電圧値 (V)
- 抵抗部品 ・・・ 抵抗値 (Ω)

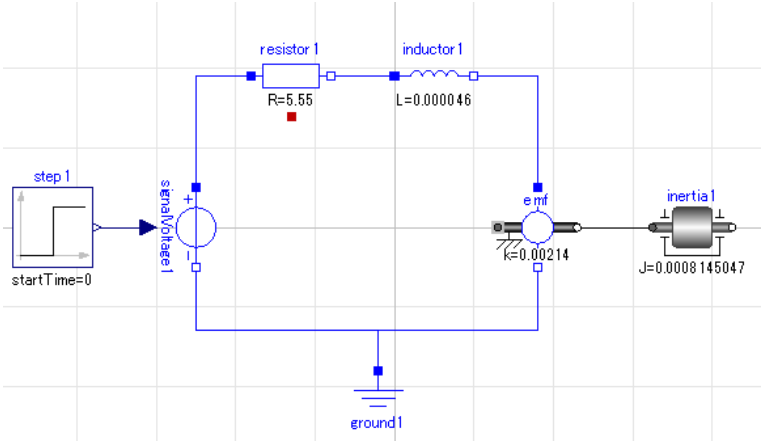

図 2.5: ブラシ付き DC モータの Modelica モデルの例

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

- 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 ($N \cdot m/A$)
- 慣性部品 ・・・ 慣性モーメント ($kg \cdot m^2$)

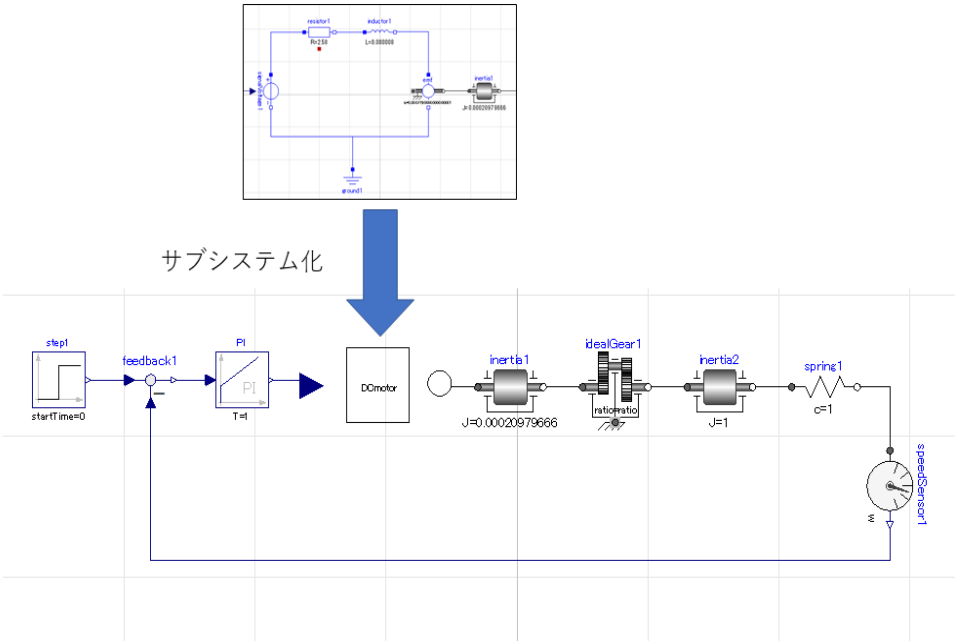

図 2.6: DC サーボモータのモデル

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]。例えば、図 2.7 に示すモータ特性表がある。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 社のブラシ付き DC モータのモータ特性表 [8, 9, 10, 11, 12, 13, 14, 15, 16, 17] に含まれる要素を集計した。集計結果を表 2.2 に示す。表 2.2 で示した要素の中で、対象とするモデルの性質により、「無負荷電流」、「無負荷回転数」、「起動トルク」を除く 9 個の要素と 4 つのグラフを、今

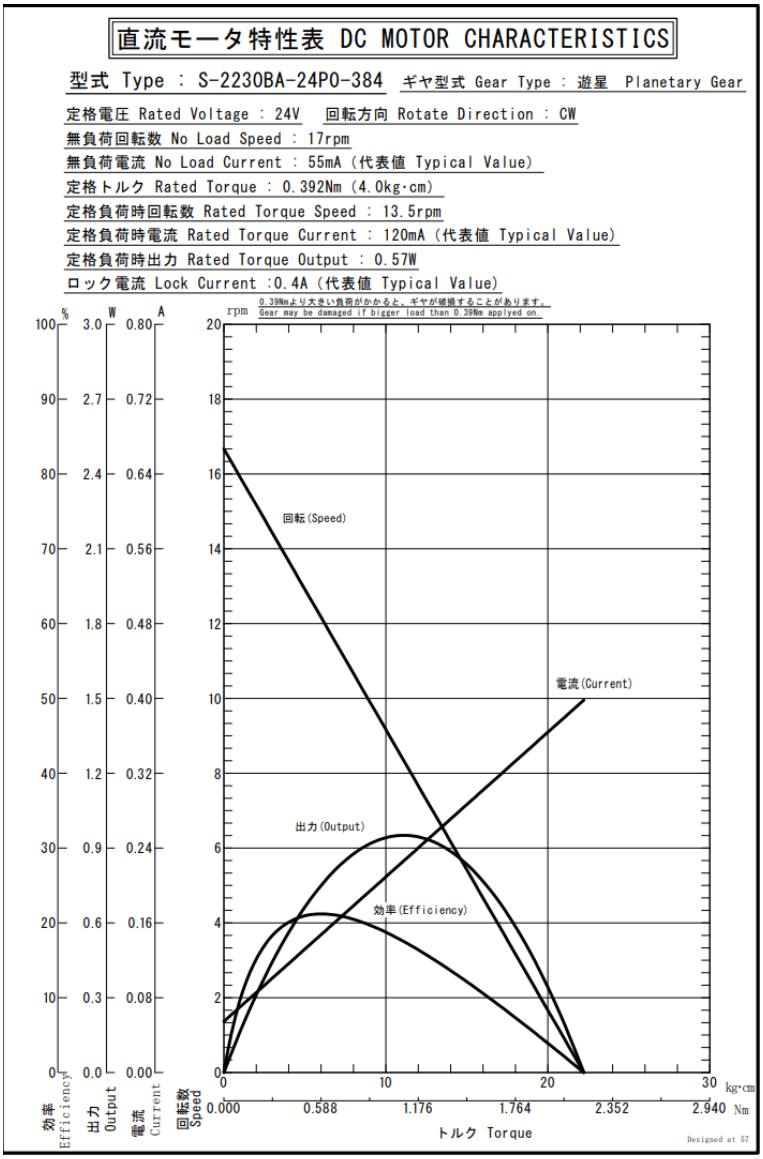

図 2.7: モータ特性表の例

回自動生成するモータ特性表の要素とした。

以下に、自動生成するモータ特性表の構成を示す。

- モータ特性表
  - 特性表
  - 特性グラフ

以下にて、本研究で対象とするモータ特性表の、特性表と特性グラフの内容について述べる。

表 2.2: モータ特性表の要素の集計結果

要素名	出現回数	要素名	出現回数
定格トルク	10	定格出力	5
定格電流	9	「負荷トルク × 効率」グラフ	5
定格回転数	9	「負荷トルク × 出力」グラフ	5
無負荷電流	8	停動トルク	4
無負荷回転数	8	始動電流	3
「負荷トルク × 電流」グラフ	7	最大回転数	2
「負荷トルク × 回転数」グラフ	7	起動トルク	2
定格電圧	5	最大効率	1

2.5.1 特性表

特性表とは、以下の 9 つの要素で構成する。

定格電圧

定格電圧とは、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 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 である。

停動トルク

停動トルクとは、モータが停止する負荷トルクの値である。単位は、mN · m 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［%］で表したものである。最大効率とは、効率の最大値を表す。

単位は、% 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 · m 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 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 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 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 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。

「負荷トルク × 電流」グラフ

「負荷トルク × 電流」グラフとは、横軸が「負荷トルク mN · 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。

「負荷トルク × 回転数」グラフ

「負荷トルク × 回転数」グラフとは、横軸が「負荷トルク mN · 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。

「負荷トルク × 効率」グラフ

「負荷トルク × 効率」グラフとは、横軸が「負荷トルク mN · 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。

「負荷トルク × 出力」グラフ

「負荷トルク × 出力」グラフとは、横軸が「負荷トルク mN · 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 年にオランダ人のグイド・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、組込み開発や、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。

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

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、本研究で試作する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、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。

- 2.4 節で述べたモデルを対象とする
- モータの回路に印加する電圧値は一定とする
- 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

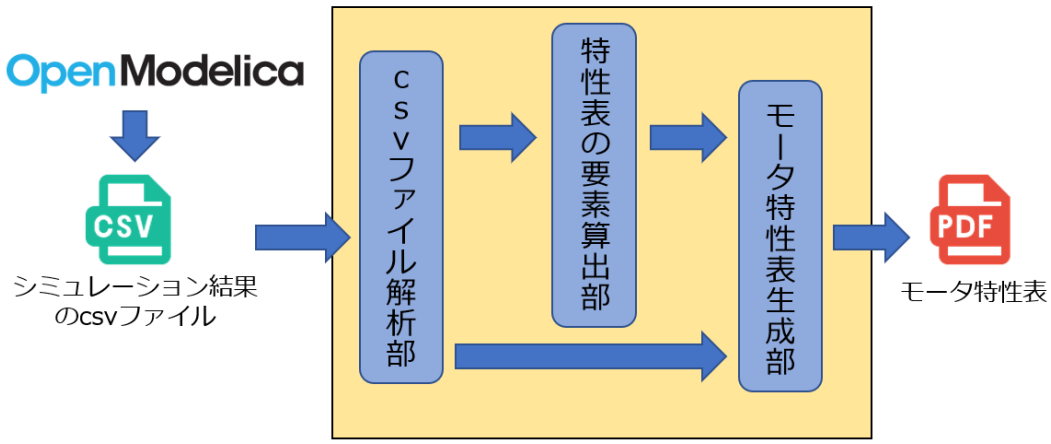

図 3.1: モータ特性表自動生成ツールの構造

  - 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 モータ特性表自動生成ツールの実行

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

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

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. 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
5. 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
6. 取得したファイル名、慣性部品のオブジェクト名、電源部品のオブジェクト名をもとに 3.1.2 節で述べる処理を実行する

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となるデータの導出計算式は、以下の通りである。

表 3.1: 各データを持つ拡張子

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

- 入力

$$\text{入力 (W)} = \text{電圧値 (V)} \times \text{電流値 (A)} \tag{3.1}$$

- 出力

$$\text{出力 (W)} = \text{トルク (N} \cdot \text{m)} \times \text{角速度 (rad/s)} \tag{3.2}$$

- 効率

$$\text{効率 (\%)} = \frac{\text{出力 (W)}}{\text{入力 (W)}} \times 100 \tag{3.3}$$

- 回転数

$$\text{回転数 (rpm)} = \frac{60 \times \text{角速度 (rad/s)}}{2\pi} \tag{3.4}$$

- 定格出力

$$\text{定格出力 (W)} = \text{定格トルク (N} \cdot \text{m)} \times \text{定格回転数 (rad/s)} \times \frac{2\pi}{60} \tag{3.5}$$

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。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の拡張子を、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。

1. 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

2. csv ファイルを読み込み専用で開く

3. 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

4. csv ファイルの各行に対して、以下の処理を繰り返す

   (a) csv ファイルから取得した行の各列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

   (c) 4.(b) で配列 row のインデックスを 4 つ取得し、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

   (d) csv ファイルの 2 行目以下を読み込んでいる場合、以下の処理を行う

      i. トルク値を格納する配列 torque に、4.(b).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

      ii. 角速度値を格納する配列 angularvelocity に、4.(b).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

```
$ python characteristic.py DCmotor_res.csv inertia1

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

```
$ 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

iii. 電圧値を格納する配列 voltage に、4.(b).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

iv. 電流値を格納する配列 current に、4.(b).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 の各要素を、それぞれ 3.4 式に適用し、回転数の値を持つ配列 speed を生成する。

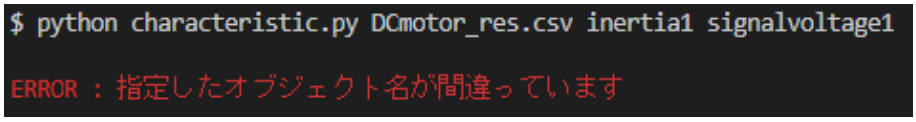

図 3.4: 第 2 引数、第 3 引数に誤りがあった場合のエラー文の例

出力

3.2 式を用いて、出力を算出する。配列 torque と、配列 angularvelocity の同じインデックスの要素を、3.2 式にあてはめ、出力を算出する。配列 torque と、配列 angularvelocity の最後のインデックスまでの出力を計算し、配列 output を生成する。

効率

3.3 式を用いて、効率を求める。配列 voltage と配列 current と配列 output それぞれの最初のインデックスの要素を、3.1 式にあてはめ、効率を求める。この処理を各配列の最後まで繰り返し、配列 efficiency を生成する。また、配列 voltage と配列 current のどちらか一方でも「0」だった場合、効率値を「0」とする。これは、0 除算を防ぐために行う。

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。

定格電圧

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 番目の値を定格電圧とする。

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る。モータの起動直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負荷トルク値である。したがって、配列 torque の要素の最大値を停動トルクとする。

最大効率

配列 efficiency の要素の最大値を最大効率とする。

定格トルク

定格トルクとは、最大効率時の負荷トルク値である。まず、配列 efficiency の中で、最大効率である要素のインデックス X を取得する。そして、配列 torque のインデックス X の要素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。まず、配列 efficiency の中で、最大効率である要素のインデックス X を取得する。そして、配列 speed のインデックス X の要素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。まず、配列 efficiency の中で、最大効率である要素のインデックス X を取得する。そして、配列 current のインデックス X の要素を定格電流とする。

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。3.5 式を用いて、定格出力を求める。上記で求めた定格トルクと定格回転数を、3.5 式に代入し、求めた解が定格出力である。

表 3.2: 特性表の枠組み

要素	データ
定格電圧 V	3.2.2 節で求めた定格電圧の値
始動電流 mA	3.2.2 節で求めた始動電流の値
停動トルク mN · m	3.2.2 節で求めた停動トルクの値
最大効率 %	3.2.2 節で求めた最大効率の値
定格トルク mN · m	3.2.2 節で求めた定格トルクの値
定格回転数 rpm	3.2.2 節で求めた定格回転数の値
定格電流 mA	3.2.2 節で求めた定格電流の値
定格出力 W	3.2.2 節で求めた定格出力の値
最大回転数 rpm	3.2.2 節で求めた最大回転数の値

最大回転数

配列 speed の要素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

特性表生成処理では、特性表を生成する。以下に、処理の流れを示す。

1. 3.2.2 節で求めたそれぞれの値を、表 3.2 にあてはめて、特性表を生成する

2. 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。


モータ特性表

定格電圧 (V)	2.4
始動電流 (mA)	431.9945756634572
停動トルク (mN.m)	0.9244683919197983
最大効率 (%)	99.9167493101763
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

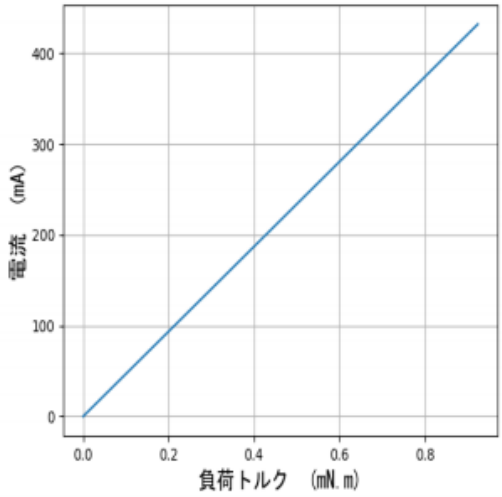

図 3.6: 「負荷トルク × 電流」グラフ

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。

1. x 軸に配列 torque を、y 軸に配列 current を指定し、表 2.3 に示したライブラリの中の、matplotlib を用いてグラフを生成し、画像として保存する

2. 1. の処理に対して、y 軸に配列 speed を指定した場合、y 軸に配列 efficiency を指定した場合、y 軸に配列 output を指定した場合を適用し、それぞれグラフを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。

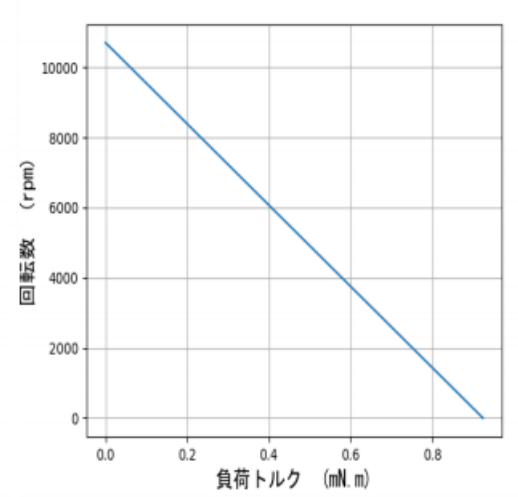

図 3.7: 「負荷トルク × 回転数」グラフ

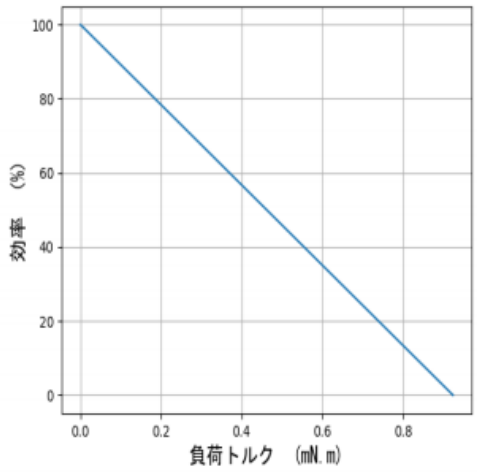

図 3.8: 「負荷トルク × 効率」グラフ

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、表 2.3 に示したライブラリの中の、reportlab を用いて、PDF ファイルに書き込み、モータ特性表を生成する。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。この処理で生成するモータ特性表を、図 3.10 に示す。

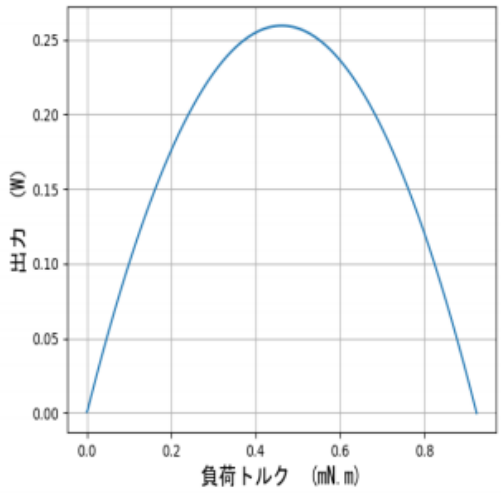

図 3.9: 「負荷トルク × 出力」グラフ

モータ特性表

定格電圧 (V)	2.4
始動電流 (mA)	431.9945756634572
停動トルク (mN.m)	0.9244683919197983
最大効率 (%)	99.9167493101763
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

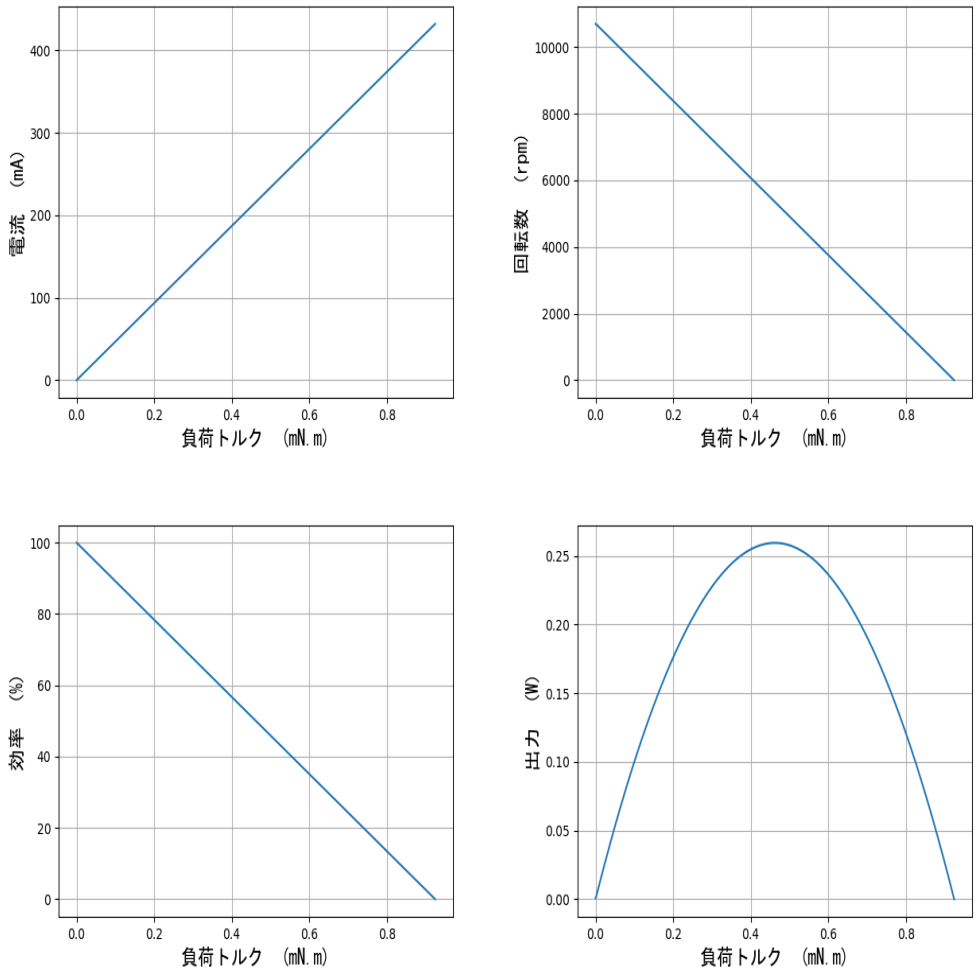

図 3.10: モータ特性表

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。適用例として、ブラシ付き DC モータの Modelica モデルと、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、の 2 つが出力する csv ファイルを適用例として用いる。また、2 つのモータの性能は同じである。正しく動作しているか確認するために、以下の 2 つの点に着目する。

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 モデルを OpenModelica で実行し、生成された 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 4.1 のシミュレーション結果である csv ファイルからモータ特性表を作成する際に使用する要素を、図 4.2 にそれぞれ示す。図 4.2 の赤丸は最大値を、青い四角形は最大効率時の行を表す。 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。

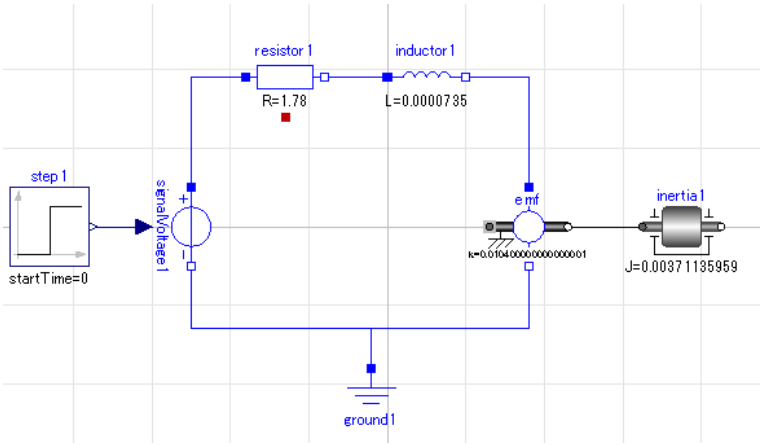

図 4.1: 適用するブラシ付き DC モータの Modelica モデル

inertia1.w	inertia1.flange_a.tau	signalVoltage1.n.i	signalVoltage1.p.v	効率	回転数	出力
0	0	0	9	0	0	0
106.2379105	46.12885568	4435.466893	9	12.27638076	1014.497316	4.900633
199.4341076	40.46587394	3890.949417	9	23.04571911	1904.455443	8.070275
281.1891208	35.49810528	3413.279354	9	32.49296508	2685.158312	9.981681
352.9075185	31.1402024	2994.250231	9	40.78042436	3370.018561	10.98961
415.8213942	27.31729825	2626.663293	9	48.05047222	3970.80182	11.35912
471.0117012	23.96370968	2304.202854	9	54.4280188	4497.83043	11.28719
519.4266309	21.02182073	2021.328916	9	60.0226329	4960.158953	10.91929
561.8979184	18.44109176	1773.1819	9	64.93042613	5365.729874	10.36201
599.155251	16.177184	1555.498462	9	69.23571789	5721.511192	9.692645
631.8387142	14.19120323	1364.538772	9	73.01247364	6033.615276	8.966552
660.5098152	12.44902998	1197.022114	9	76.32557865	6307.404123	8.222706
685.6611266	10.92073354	1050.070532	9	79.2319524	6547.581455	7.487922
707.7247559	9.580057262	921.1593521	9	81.78152735	6758.273595	6.780044
727.0797599	8.403968134	808.073859	9	84.01810559	6943.100269	6.110355
744.0586582	7.372260901	708.8712405	9	85.98011161	7105.236804	5.485395
758.9531557	6.467210369	621.8471509	9	87.70125355	7247.468779	4.90831
772.0191392	5.673267828	545.5065219	9	89.21110053	7372.23973	4.379871
783.4810874	4.976793022	478.5377906	9	90.53559232	7481.693272	3.899223
793.5359167	4.365820507	419.7904334	9	91.69748371	7577.70982	3.464435
802.3563724	3.829853579	368.2551519	9	92.71673636	7661.938967	3.072907
810.0939909	3.359684262	323.0465637	9	93.61086117	7735.827781	2.72166
816.8817058	2.947234972	283.387978	9	94.39521933	7800.645684	2.407542
822.8361321	2.585419701	248.5980481	9	95.08328637	7857.506266	2.127377
828.0595667	2.268022469	218.0790835	9	95.68688326	7907.386393	1.878058
832.6417393	1.989590939	191.3068211	9	96.21637876	7951.142918	1.656616
836.6613385	1.745343666	167.8215063	9	96.68086579	7989.527263	1.460262
840.1874723	1.531081369	147.2193624	9	97.08833013	8023.199361	1.286395
843.2807017	1.343124113	129.1465494	9	97.44576997	8052.737525	1.132631
845.9942023	1.178240728	113.2923777	9	97.75933004	8078.649547	0.996785
848.3745867	1.033598846	99.38450439	9	98.03439669	8101.380544	0.876879
850.462754	0.906713269	87.18396816	9	98.27569602	8121.321073	0.771126
852.2945765	0.795404251	76.48117794	9	98.48737328	8138.813689	0.677919
853.9015204	0.697759774	67.09228596	9	98.67306457	8154.158873	0.595818
855.3111939	0.61210225	58.85598561	9	98.83596018	8167.620263	0.523538
856.5478149	0.536960103	51.63077914	9	98.97885861	8179.429124	0.459932
857.6326283	0.471042412	45.29253958	9	99.10421482	8189.788329	0.403981
858.5842687	0.413216848	39.73238924	9	99.21418217	8198.875826	0.354781
859.4190841	0.362490055	34.85481297	9	99.31064972	8206.847726	0.311531
860.1514173	0.317990497	30.57600936	9	99.39527489	8213.840992	0.27352
860.7938489	0.278953731	26.82247417	9	99.46951143	8219.975762	0.240122
861.3574152	0.244709139	23.52972492	9	99.53463464	8225.357424	0.210782
861.8517978	0.21466843	20.64119516	9	99.59176331	8230.07843	0.185012
862.2854898	0.188315536	18.1072631	9	99.64187882	8234.219883	0.162382
862.6659413	0.165197748	15.88439886	9	99.6858421	8237.852927	0.14251
862.9996875	0.144917963	13.93441949	9	99.72440833	8241.039969	0.125064
863.2924629	0.127127734	12.22382062	9	99.75824016	8243.835768	0.109748
863.5492972	0.111521435	10.72321489	9	99.78791878	8246.288355	0.096304
863.7746024	0.097830969	9.406823936	9	99.81395405	8248.439861	0.084504
863.972249	0.085821152	8.252033823	9	99.83679322	8250.327247	0.074147
864.1456324	0.075285669	7.239006641	9	99.85682863	8251.982937	0.065058
864.1456324	0.075285669	7.239006641	9	99.85682863	8251.982937	0.065058

図 4.2: 図 4.1 のシミュレーション結果である csv ファイルの一部と効率と回転数

モータ特性表

定格電圧 (V)	9
始動電流 (mA)	4435.466892607765
停動トルク (mN.m)	46.12885568312076
最大効率 (%)	99.85682863210921
定格トルク (mN.m)	0.07528566906383065
定格回転数 (rpm)	8251.98293679948
定格電流 (mA)	7.239006640752946
定格出力 (W)	0.06505778210331306
最大回転数 (rpm)	8251.98293679948

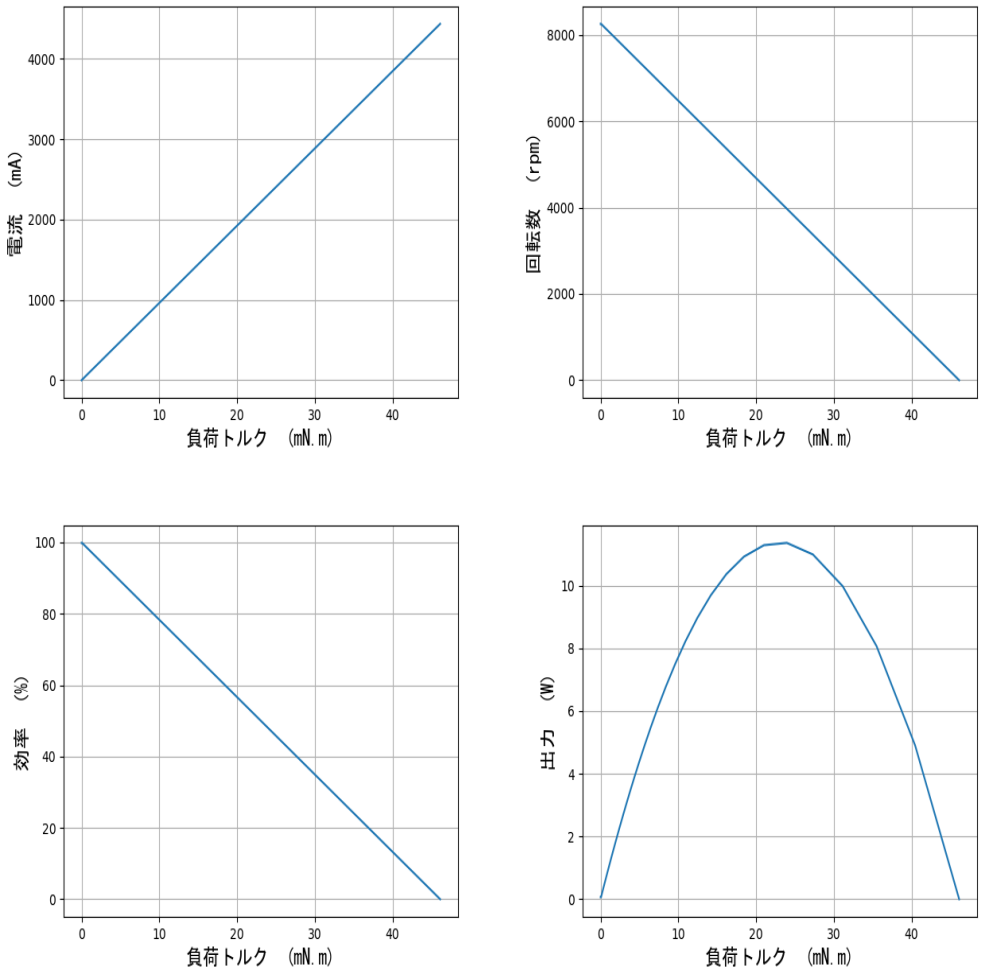

図 4.3: 適用例で生成したモータ特性表

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。

定格電圧

図 4.3 の特性表にある定格電圧の値と、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 inertia1.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 inertia1.flange_a.tau 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 signalVoltage1.n.i 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地である。よって、正しく出力していることが確認できる。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。

「負荷トルク × 電流」グラフ

図 4.2 の inertia1.flange_a.tau の持つ値と、signalVoltage1.n.i が持つ値を確認できるので、正しく生成していることが確認できる。

「負荷トルク × 回転数」グラフ

図 4.2 の inertia1.flange_a.tau の持つ値と、回転数が持つ値を確認できるので、正しく生成していることが確認できる。

「負荷トルク × 効率」グラフ

図 4.2 の inertia1.flange_a.tau の持つ値と、効率が持つ値を確認できるので、正しく生成していることが確認できる。

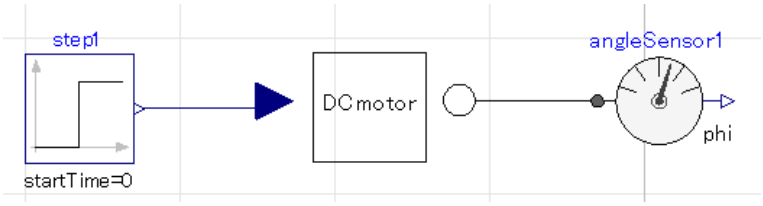

図 4.4: 適用するブラシ付き DC モータの Modelica モデルをサブシステムとするモデル

「負荷トルク × 出力」グラフ

図 4.2 の inertia1.flange_a.tau の持つ値と、出力が持つ値を確認できるので、正しく生成していることが確認できる。

したがって、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を確認できる。

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

今回適用するブラシ付き DC モータの Modelica モデルをサブシステムとするモデルは、図 4.1 のモデルと同じ性能になるよう設計しており、どちらのモータ特性表も同じ内容になっていることを確認する。

今回適用するブラシ付き DC モータの Modelica モデルをサブシステムとするモデルを図 4.4 に、生成したモータ特性表を図 4.5 にそれぞれ示す。

図 4.3 と図 4.5 の内容が同じであるため、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が確認できる。

よって、モータ特性表自動生成ツールは、「ブラシ付き DC モータの Modelica モデル」と「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」に対応していることが確認できる。

モータ特性表

定格電圧 (V)	9
始動電流 (mA)	4435.466892607765
停動トルク (mN.m)	46.12885568312076
最大効率 (%)	99.85682863210921
定格トルク (mN.m)	0.07528566906383065
定格回転数 (rpm)	8251.98293679948
定格電流 (mA)	7.239006640752946
定格出力 (W)	0.06505778210331306
最大回転数 (rpm)	8251.98293679948

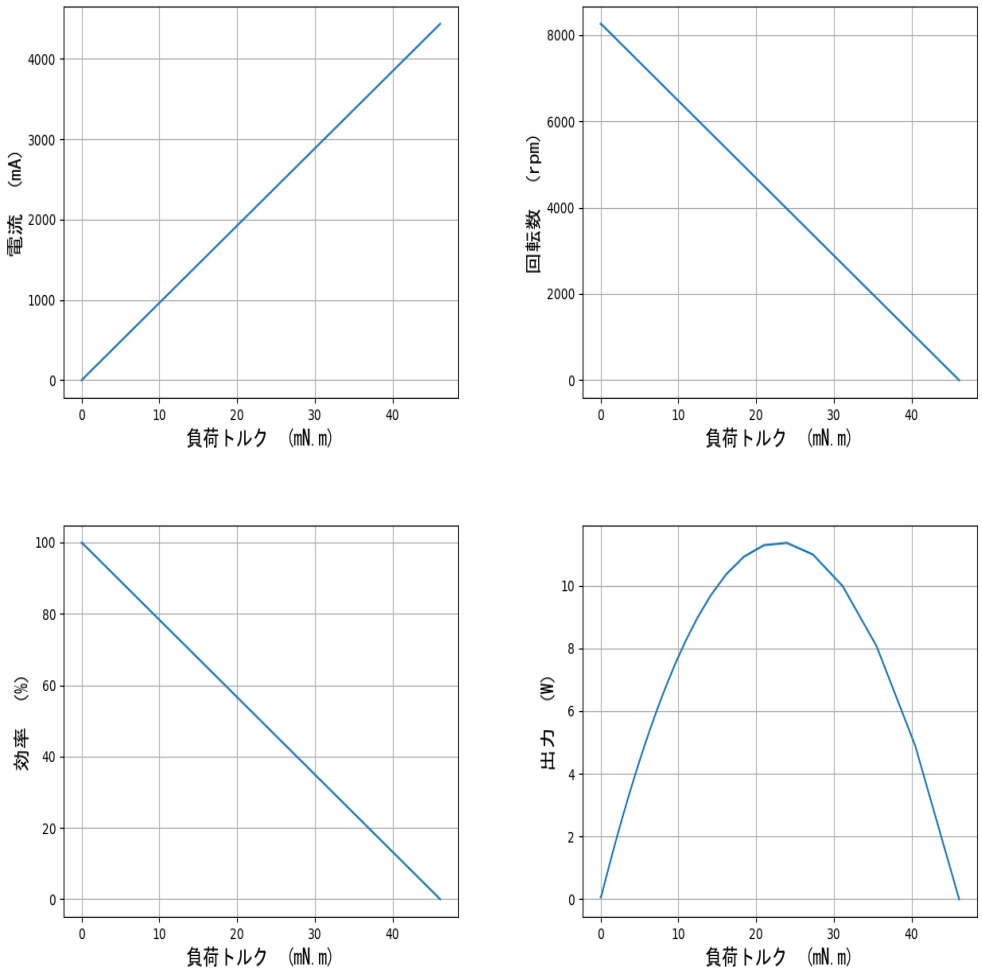

図 4.5: 図 4.4 のモータ特性表

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

(TODO:下記のコマンドを書き換える！)

既存の手法と、本研究の手法で、作成 (生成) に要した時間の比較検証を行った。その結果を、表 5.1 に示す。

対象とした仕様書は、**??**節で用いた コード**??**である。仕様書を作成する時間を計測した。生成する仕様書としては、以下を基準とした。(TODO:なにをもって完成かを書く！)

1. on ポイント、off ポイント、in ポイント、out ポイントを出力 (記述) する
2. on ポイント、off ポイント、out ポイントには、着目条件式も出力 (記述) する
3. off ポイントには、着目変数も出力 (記述) する
4. 各ポイントには、期待出力と正常系であるかどうかも出力 (記述) する

検証に参加したメンバーは本研究室の大学院生 (TODO:X) 人と学部 4 年生 (TODO:Y) 人であり、普段からソースコードの読み書きを行い、基本的なプログラミングの知識を有している。仕様書の知識を持たない者も含まれるが、今回の検証に必要な文法は、事前に他の仕様書の例を用

いてレクチャーした。また、仕様書生成についても、事前に他の仕様書と仕様書の例を用いてレクチャーした。

人手による検証では、コード**??**を印刷した紙を渡し、仕様書を確認後、仕様書を書き始めてから、仕様書を記述し終えるのに要した時間を計測した。(TODO:なんか) が不正確な場合、間違いを指摘し、被験者が正しい仕様書を記述した時点で時間計測終了とした。また、制限時間を (TODO:Z) 分とし、制限時間を超えた場合、その場で時間計測終了とした。

本研究の手法による検証では、(TODO:計測はじめから終わりの条件と、使った PC の仕様を書く) コマンドライン上での命令操作で、本研究の手法による仕様書生成を行うのに要した時間を計測した。また、実験に用いたコンピュータは、OS:macOS 10.14.5、CPU:2.3GHz Intel Core i5、メモリ:16GB である。

(TODO:純粋な、実行時間を書く) なお、Java の System.nanoTime[**?**] メソッドを用いて、命令操作を省いた純粋な仕様書生成処理に本研究の手法が要した時間を計測した結果、(TODO:A) 秒であった。

人手による作成と比較した結果、平均で (TODO:B) 分程の時間短縮を確認できた。対象にした仕様書には、(TODO:なにか) 独特の文法等は含まれないため、(TODO:なにか) に対する慣れなどの影響は無視できるものと思われる。また、人手による仕様書生成の場合、ヒューマンエラーも見られた。(TODO:ヒューマンエラーがあったら、具体例を書く) 具体的には、off ポイントの記述時に、条件式の解釈を間違え、誤った期待出力を記述してしまった。(例：入力 (17、 20) の期待出力を “遊園地チケットは割引価格とならない。(妻の年齢 < 16)” と記述した。) 仕様書の規模が拡大すると、人手とコンピュータとの処理効率の差に加えて、ヒューマンエラーの有無などにより、仕様書生成に要する時間の差は更に拡大していくと思われる。以上から、本研究の手法は有用性が向上したと考える。

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。

表 5.1: コード**??**の仕様書作成に要した時間の比較

被験者	時間
被験者 A	8m 16s
被験者 B	10m 23s
被験者 C	30m(制限時間超過)
被験者 D	24m 04s

	時間
被験者 (平均)	18m 10s
BWDM	0m 15s

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://arduinopid.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/l001040/l01040.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%ABIE3%E9%AB%98%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.