

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 1 月

# 目次

1	はじめに	1
2	研究の準備	2
2.1	モータ作成	2
2.1.1	仕様書	2
2.1.2	シミュレータの役割	2
2.2	モータ特性表	2
2.2.1	特性表の種類	2
2.2.2	特性表の要素	2
2.3	OpenModelica	2
2.4	modelica	2
3	機能	3
3.1	対応するモデル	3
3.1.1	モータ単体のモデル	3
3.1.2	モータ単体のモデルをサブシステムとするモデル	4
3.2	特性表生成	4
3.2.1	特性表生成機能	4
4	実装	6
4.1	シミュレーション結果解析機能	6
4.2	特性表生成機能	6
5	適用例	7
5.1	モータ単体のモデル	7
5.2	パッケージ化されたモデル	7
6	考察	8

6.1	評価 . . . . .	8
6.1.1	評価方法 . . . . .	8
6.1.2	結果 . . . . .	8
6.2	関連研究 . . . . .	8
6.3	ツールの問題点 . . . . .	8
7	おわりに	9
	謝辞	10
	参考文献	11

# 第 1 章

## はじめに

本論文の構成は、以下の通りである。

第 2 章では、試作したモータ特性表自動生成ツールを開発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、モータ特性表自動生成ツールの実装について説明する。

第 5 章では、試作したモータ特性表自動生成ツールの機能が正しく動作することを検証する。

第 6 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 7 章では、本論文のまとめと今後の課題を述べる。

## 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 モータ作成

#### 2.1.1 仕様書

#### 2.1.2 シミュレータの役割

### 2.2 モータ特性表

#### 2.2.1 特性表の種類

#### 2.2.2 特性表の要素

### 2.3 OpenModelica

### 2.4 modelica

## 第 3 章

## 機能

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の機能について説明する。

モータ特性表自動生成ツールは、OpenModelica でモータのモデルをシミュレーションした時に出力される csv ファイルを読み込み、実行することによって、モータ特性表を生成する。

### 3.1 対応するモデル

今回試作したモータ特性表自動生成ツールでは以下の Modelica モデルに対応する。

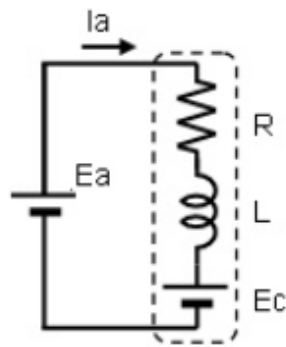
- モータ単体のモデル
- モータ単体のモデルを一つに統一したモデルを使用するモデル

なお、今回はモータの中でもブラシ付き DC モータに対応する。

以降、上記のモデルについて具体的に説明する。

#### 3.1.1 モータ単体のモデル

モータ単体のモデルとは、電源部品、抵抗部品、インダクター部品、起電力部品、慣性部品、接地部品の OpenModelica でブラシ付き DC モータについてシミュレーションするために、最低限必要となる部品を持つモデルのことである。



$E_a$  : 電源電圧、 $I_a$  : モータの電流、 $R$  : 電機子抵抗  
 $L$  : コイルのインダクタンス、 $E_c$  : モータの発電電圧

図 3.1: 等価回路

上記 6 つの部品が必要な理由は、ブラシ付き DC モータの等価回路 [1] を Modelica で表す際に使用する部品 [2] だからである。

ブラシ付き DC モータの等価回路を図 3.1 に、モータ単体のモデルを図 3.2 に、モータ単体のモデルを Modelica コードで表したものを図 3.3 に示す。

### 3.1.2 モータ単体のモデルをサブシステムとするモデル

モータ単体のモデルをサブシステムとするモデルとは、3.1.1 節で説明したモータ単体のモデルを一つのモデルにまとめ、

## 3.2 特性表生成

今回試作したモータ特性表自動生成ツールには次の～～～個の機能がある。

- are
- sore

### 3.2.1 特性表生成機能

特性表生成機能は、

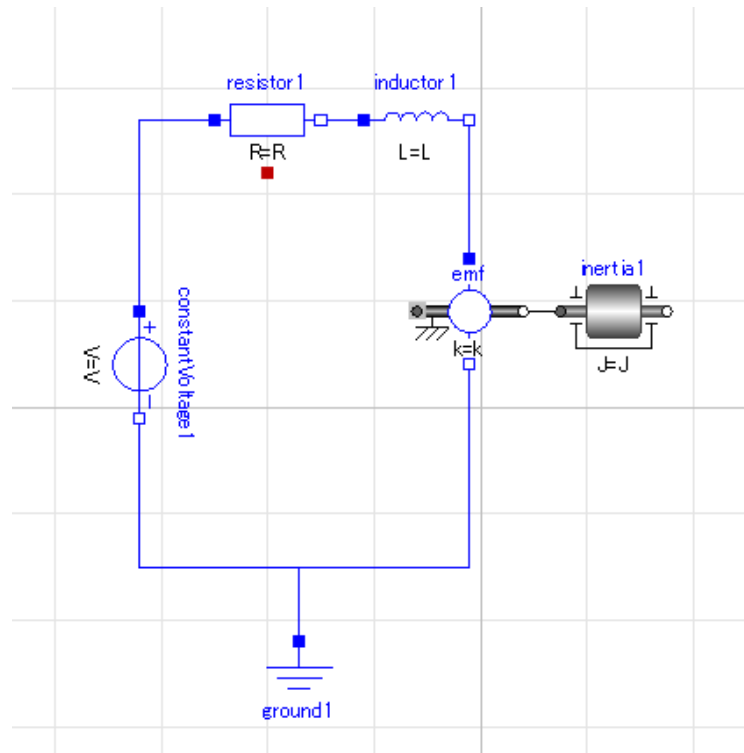


図 3.2: モータ単体のモデル

```

1 model DCmotor
2   Modelica.Electrical.Analog.Basic.Resistor resistor1(T = 283.15) annotation(
3     Placement(visible = true, transformation(origin = {-40, 54}, extent = {{-10, -10}, {10, 10}}, rotation = 0)));
4   Modelica.Electrical.Analog.Basic.Inductor inductor1 annotation(
5     Placement(visible = true, transformation(origin = {-12, 54}, extent = {{-10, -10}, {10, 10}}, rotation = 0)));
6   Modelica.Electrical.Analog.Basic.Ground ground1 annotation(
7     Placement(visible = true, transformation(origin = {-34, -54}, extent = {{-10, -10}, {10, 10}}, rotation = 0)));
8   Modelica.Mechanics.Rotational.Components.Inertia inertia1(a(start = 0), phi(start = 0), w(start = 0)) annotation(
9     Placement(visible = true, transformation(origin = {25, 18}, extent = {{-10, -10}, {10, 10}}, rotation = 0)));
10  Modelica.Electrical.Analog.Basic.EMF emf(useSupport = false) annotation(
11    Placement(visible = true, transformation(origin = {-2, 18}, extent = {{-10, -10}, {10, 10}}, rotation = 0)));
12  Modelica.Electrical.Analog.Sources.ConstantVoltage constantVoltage1 annotation(
13    Placement(visible = true, transformation(origin = {-64, 8}, extent = {{-10, -10}, {10, 10}}, rotation = -90)));
14  equation
15    connect(constantVoltage1.n, ground1.p) annotation(
16      Line(points = {{-64, -2}, {-64, -2}, {-64, -30}, {-64, -30}, {-34, -30}, {-34, -44}, {-34, -44}, {-34, -44}}, color = {0, 0, 255}));
17    connect(constantVoltage1.n, emf.n) annotation(
18      Line(points = {{-64, -2}, {-64, -2}, {-64, -30}, {-2, -30}, {-2, 8}, {-2, 8}}, color = {0, 0, 255}));
19    connect(constantVoltage1.p, resistor1.p) annotation(
20      Line(points = {{-64, 18}, {-64, 18}, {-64, 54}, {-50, 54}, {-50, 54}}, color = {0, 0, 255}));
21    connect(resistor1.n, inductor1.p) annotation(
22      Line(points = {{-30, 54}, {-22, 54}}, color = {0, 0, 255}));
23    connect(inductor1.n, emf.p) annotation(
24      Line(points = {{-2, 54}, {-2, 28}}, color = {0, 0, 255}));
25    connect(emf.flange, inertia1.flange_a) annotation(
26      Line(points = {{8, 18}, {15, 18}}));
27  annotation(
28    uses(Modelica(version = "3.2.3"));end DCmotor;

```

図 3.3: モータ単体のモデルの Modelica コード



## 第 4 章

## 実装

### 4.1 シミュレーション結果解析機能

### 4.2 特性表生成機能

## 第 5 章

## 適用例

本章では、本研究で作成した

### 5.1 モータ単体のモデル

### 5.2 パッケージ化されたモデル

## 第 6 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。

### 6.1 評価

#### 6.1.1 評価方法

#### 6.1.2 結果

本論文で試作したモータ特性表自動生成ツールは、

### 6.2 関連研究

関連研究について述べる。

### 6.3 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 7 章

## おわりに

以下に、今後の課題を示す。

## 謝辞

## 参考文献

- [1] Device Plus - デバプラ ,”モータに最大の電流が流れる状態について” : [https://deviceplus.jp/glossary/qa\\_006/](https://deviceplus.jp/glossary/qa_006/), アクセス日:2020/01/17.
- [2] Peter Fritzson 著 (監訳:大畠 明, 訳:広野 友英):”Modelica によるシステムシミュレーション入門”, TechShare 社 (2015).