



PLECS Tutorial

Implicit Model Vectorization

コンポーネントのベクトル化

PLECSの様々なベクトル化手法とその実装方法についての学習

Tutorial Version 1.0

1 はじめに

PLECSでは、複数のコンポーネントとして解釈したり、複数の信号を一度に処理するために、複数のコンポーネントをベクトル化することができます。この機能は回路の一部を何度も重複して記述することを防ぐのに役立ち、小さな回路図スペース内で大規模なシステムを記述することが可能になります。この機能はさらに広範囲に使用でき、ベクトルの長さやパラメータマスクの値を変更するだけでシステムのサイズを変更することもできます。この再構成能力は、直列/並列マルチレベルコンバータやソーラーパネルのモデリングなど、最適なセル数を検索するとき特に役立ちます。

この演習では、次の内容を学習します:

- コンポーネントパラメータにベクトルを追加する方法
- 基本的なベクトル化機能を実行するブロック
- より簡潔な形で回路を理解し、設計する方法
- ユーザ定義パラメータを実装して変数構造を作成する方法

始める前に: `implicit_model_vectorization_1.plecs`から`implicit_model_vectorization_9.plecs`までのファイルが作業ディレクトリに配置されていることを確認します。

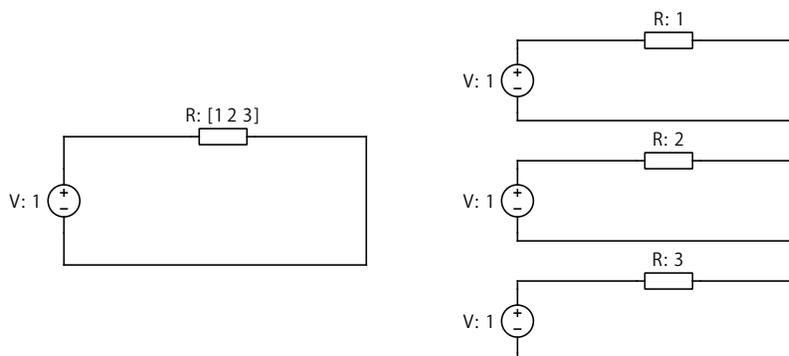
2 ベクトル化の例

2.1 コンポーネントにベクトルを組み込む

ベクトルパラメータは要素を区切るための角括弧と、オプションのカンマを使用して入力します(例: `[1, 2, 3]` や `[1 2 3]`)。PLECSの多くのコンポーネントはベクトル化できます。これには、正弦波信号ブロック、定数ブロック、電源などが含まれます。どのブロックがベクトル化できるかについては、PLECSユーザマニュアルを参照してください。

ベクトル化の簡単な例を [図1](#) に示します。ここで、左側の抵抗器は3つの抵抗値のベクトルとして構成されており、この回路は実質的に右側に示す3つの個別の回路と同等です。[図1](#)の左側の回路のDC電圧源をベクトル化する必要はありません。回路内にベクトル化されたコンポーネントがあると、この回路内の追加のスカラーコンポーネントが効果的にベクトル化されるためです。したがって1Vの電圧は、`[1 1 1]`Vとして拡張され、`[1 2 3]`Ωの抵抗器のベクトル要素に接続されます。この電圧源値の暗黙的な拡張は、必要な値を持つ長さ3のベクトルを直接指定することで回避できます。

図1: ベクトル化された抵抗器(左)と非ベクトル化抵抗器(右)の等価回路



あなたのタスク: モデル`implicit_model_vectorization_1.plecs`を開き、シミュレーションを実行して数値表示ブロックの結果を比較することで、2つの回路実装が実際に同等であることを確認します。

? 左側(ベクトル化された)の回路を見ると、抵抗器のベクトルの要素([1 2 3])に2Aを供給するためには、入力電圧の値はどれくらいにすべきですか?

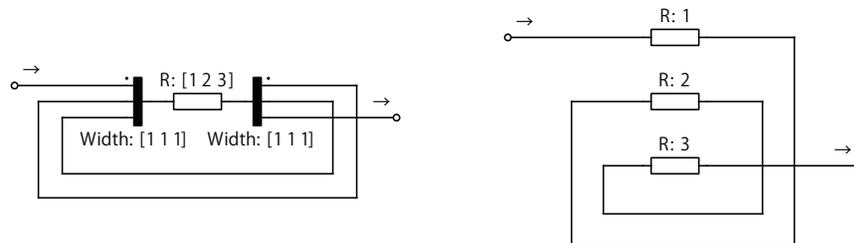
A 入力電圧を[2 4 6]Vのベクトルに変更する必要があります。回路内で複数のベクトル化されたコンポーネントを接続する場合、それらはすべて同じサイズである必要があります。

2.2 ベクトル化したコンポーネントの直列化

ワイヤ・マルチプレクサブロックの使用

複数の個別のワイヤをワイヤバスに相互接続するには、ワイヤ・マルチプレクサを使用します。図2では、コンポーネントの直列接続の例を、ベクトル化された構成と同等の非ベクトル化された構成で示しています。左側のベクトル化回路の信号の流れをたどるには、左側の未接続端からワイヤ・マルチプレクサの最初のポートを経由し、抵抗器の最初の値を通過し、2番目のワイヤ・マルチプレクサの最初の(ドットの付いた)ポートから出て、右下の他の未接続のワイヤポートまでワイヤをトレースします。右側のベクトル化されていない等価回路は、視覚的な補助のために同じ幾何学的パターンで配置されています。2つのワイヤ・マルチプレクサブロック内のベクトル要素の幅([1 1 1])により、電流が各ポートを1回だけ流れることに注意してください。電流がその抵抗器に入るたびに、新しい抵抗値のベクトル要素を使用します。すべての要素の幅が1の場合、角括弧なしでスカラー値(例: 3)を使用することもできます。

図2: ベクトル化された抵抗器の直列回路(左)と非ベクトル化抵抗器(右)の等価回路



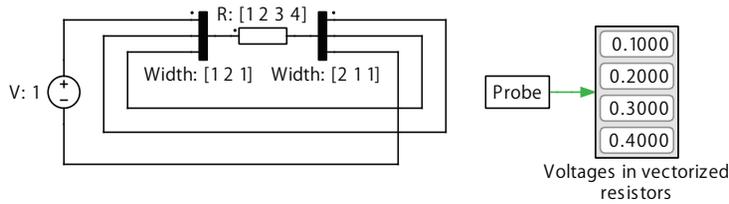
注意: ワイヤ・マルチプレクサ内の横にドットがあるポートは、(デ)マルチプレクサ信号の最初の要素を示します。これは、(デ)マルチプレクサや変圧器コンポーネント(ドットは一次側と二次側の一次巻線を示します)などの他のライブラリブロックの規則に従います。

あなたのタスク:

- 1 モデルimplicit_model_vectorization_2.plecsを開き、シミュレーションを実行して数値表示ブロックの結果を比較することで、2つの回路実装が実際に同等であることを確認します。
- 2 **抵抗**ベクトルの末尾に4番目の値(例: 4)を追加し、左側と右側のワイヤ・マルチプレクサの**信号数**をそれぞれ[1 2 1]と[2 1 1]に変更します。シミュレーションを実行し、数値表示ブロックの下隅を下にドラッグして、4番目のウィンドウを表示します。

モเดลは図3のようになるはずです。

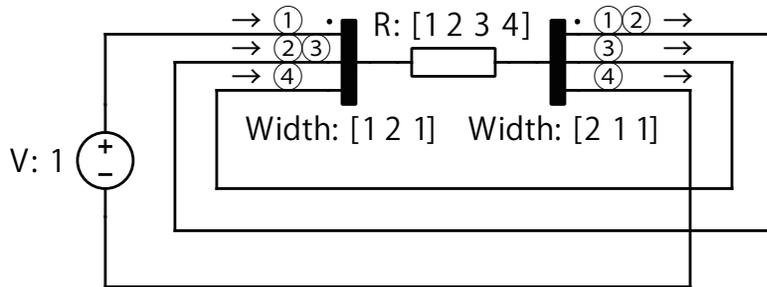
図3: 変更した直列のベクトル化抵抗器



基本的には、別の抵抗値を追加し、ワイヤループの1つを再利用して、そのワイヤの周囲の2つのワイヤ・マルチプレクサポートの幅を拡張しました。信号数[1 2 1]と[2 1 1]は、異なるワイヤループを使用しているだけで、同じ結果になります。

図4の数字は、電流の流れを説明するために使用しています。左上の①から始まり、それを①に接続して抵抗器のバスを横切ってワイヤループをたどり、抵抗器の左②に到達したときにのみ②にインクリメントします。同じ手順を繰り返して、右下の④に到達します。これは、DC電圧源のマイナス端子に接続されるワイヤになります。

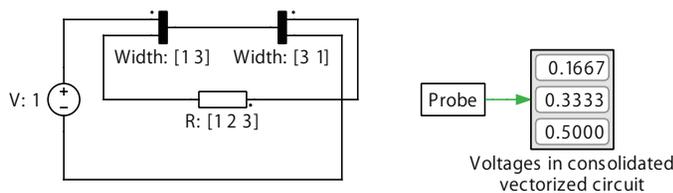
図4: 電流の流れの説明



さらなる統合

ご覧のとおり、ワイヤ・マルチプレクサポートの信号数を増やすことで、ベクトル化された回路内のポートを再利用することができます。ループする回数を1つに減らし、その中に抵抗器を配置すると、図5に示すように、回路の要素はさらに少なくなります。このループを電流が通過する回数は、ワイヤループが接続されている2つのポート幅によって異なります。この場合、ポートの幅は3なので、ループを3回使用し、同じ回路の同等で、さらに凝縮されたバージョンが作成されます。

図5: さらに統合されたベクトル化モデル



モデルは図5のようになるはずですが。



あなたのタスク: モデルimplicit_model_vectorization_3.plecsを開き、統合した回路の実装が元の実装と同じ結果を生成することを確認します。



この回路を別の抵抗値に対応するには、どのように変更しますか？

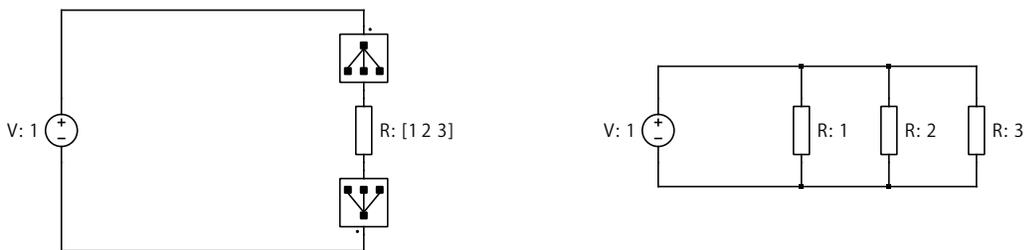


A 抵抗ベクトルに別の値を追加し、ワイヤ・マルチプレクサの信号数を3から4に変更します(つまり[1 4]と[4 1])。

2.3 ベクトル化したコンポーネントの並列化

並列に接続されたコンポーネントをベクトル化するには、ワイヤ・セレクタブロックを使用します。図6のように、抵抗器はベクトル化され、両側がワイヤ・セレクタブロックで囲まれています。ワイヤ・セレクタブロックの出力インジケータは、抵抗器のベクトルサイズと一致します。

図6: ベクトル化された抵抗器と非ベクトル化抵抗器の等価回路



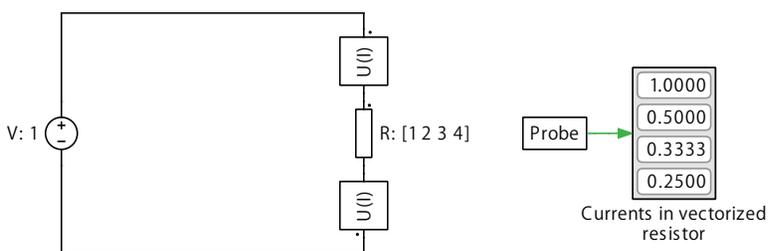
あなたのタスク:

- 1 モデルimplicit_model_vectorization_4.plecsを開き、シミュレーションを実行して数値表示ブロックの結果を比較することで、2つの回路実装が実際に同等であることを確認します。
- 2 抵抗ベクトルの末尾に4番目の値(例: 4)を追加し、左側と右側のワイヤ・セレクタの出力ベクトルに4番目の要素を追加します。シミュレーションを実行し、数値表示ブロックの下隅を下にドラッグして、4番目のウィンドウを表示します。



モデルは図7のようになるはずです。

図7: 変更した並列のベクトル化抵抗器



2.4 ベクトル化回路

達成しようとしていることに応じて、いくつかの異なる方法で回路全体をベクトル化できる場合があります。

 **あなたのタスク:** モデルimplicit_model_vectorization_5.plecsを開き、用意されている3つのベクトル化回路を比較します。

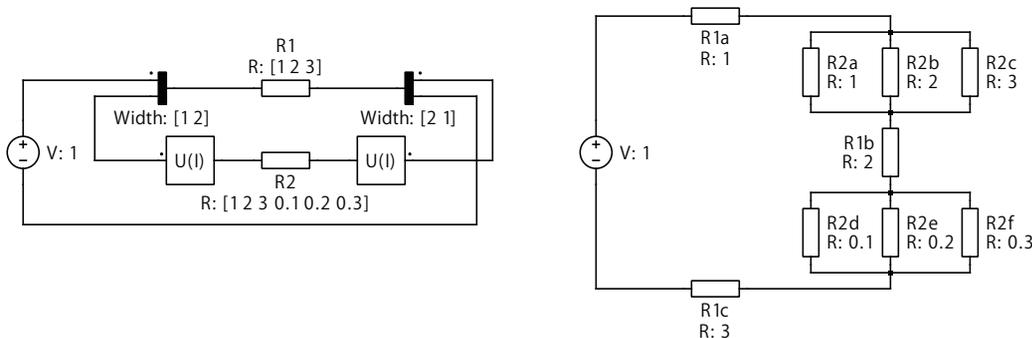
 これらの回路の違いは何ですか？

-  ・ 最初(一番上)の回路では、ベクトル化されたコンポーネント(インダクタ)が残りのスカラーコンポーネントを効果的にベクトル化するため、機能的には3つの個別の回路を持つことと同等になります。このため、PLECSスコープには3つの負荷電圧が表示されます。これは図1でも起こっていることです。
- ・ 2番目のケースでは、ベクトル化されたインダクタは、ワイヤ・セレクトブロックを介して残りのスカラーコンポーネントから分離します。このような構成は、3つのインダクタを並列に接続することと同等であり、図6と同じ考え方はです。
- ・ 3番目のモデルでは、降圧コンバータ全体が、ワイヤ・セレクトブロックを介してキャパシタ(理想モデル)と負荷抵抗器から分離されます。これにより、インタリーブ降圧コンバータが作成され、回路の左側部分の3つのインスタンスがすべて並列化され、共通のキャパシタ(理想モデル)と負荷抵抗器に接続されます。この場合、上の回路と区別するためにパルス発生器に位相遅れを追加しています。

2.5 直列ベクトル化と並列ベクトル化の組み合わせ

次に、図5と図6の考え方を使用して、図8の左側の回路のように、直列抵抗と並列抵抗で構成されるベクトル化回路を描いてみましょう。電流の流れを追ってみると、最初に"R1"(右側の回路の値が1Ωの"R1a"に対応)に入り、次に並列抵抗器"R2"(右側の"R2a"、"R2b"、"R2c"に対応)を含むワイヤループを流れ、その後"R1"に戻り、そのベクトルから次の値("R1b"これは2Ω)を取得し、これを繰り返します。

図8: 直列抵抗と並列抵抗のベクトル化回路(左)と等価回路(右)



 **あなたのタスク:** モデルimplicit_model_vectorization_5.plecsを開き、用意されている3つのベクトル化モデルimplicit_model_vectorization_6.plecsを開き、シミュレーションを実行して抵抗器の電流と電圧の数値表示ブロックの結果を比較し、2つの回路実装が同等であることを確認します。

2.6 変数の使用

ワイヤループの反復回数や並列コンポーネントの数をハードコーディングする代わりに、変数を使用してより動的な手法を取ることができます。この変数は、**シミュレーションパラメータ(Ctrl+E)** -> **初期化**タブ、またはサブシステムブロックのマスクで定義できます。

初期化タブで定義



あなたのタスク:

- 1 セクション2.2と2.3の最終的な直列モデルと並列モデルが含まれているimplicit_model_vectorization_7.plecsを開きます。
- 2 両方の抵抗器の**抵抗**ベクトルとワイヤ・セレクトアの**出力インジケータ**を $\text{ones}(1, n) * 1$ に置き換えます。これは、長さが n であるすべて1の配列であり、これについてはすぐに定義します。これは配列なので、この場合、角括弧はオプションです。この場合、ベクトルはすでに1の連続なため、末尾の*1はオプションですが、抵抗器のベクトルを1以外の均一な値で初期化する場合は、この値を変更します。
- 3 ワイヤ・マルチプレクサのパラメータウィンドウで、**信号数**パラメータも n (つまり、 $[1 \ n]$ および $[n \ 1]$)に変更します。ワイヤ(デ)マルチプレクサの最初の要素は、横にドットがあるポートに与えられた値を参照することに注意してください。
- 4 **シミュレーションパラメータ(Ctrl+E)** -> **初期化**タブで、 $n=5$ を定義します。これは、ベクトル化された回路を動的に再構成するために変更できるパラメータです。
- 5 シミュレーションを実行し、取得する電流と電圧が適切であることを確認します。



この段階では、モデルは参照モデルimplicit_model_vectorization_8.plecsと同じになっているはずです。

サブシステムのマスクで定義

代わりに回路からサブシステムを作成し、そのマスクを介してパラメータ値を渡すこともできます。これを行うには、両方の回路に対して以下の手順に従います。



あなたのタスク:

- 1 それぞれの回路で、DC電圧源以外のすべてを選択し、抵抗器を右クリックして**サブシステムの作成(Ctrl+G)**を選択します。
- 2 オプションの手順として、**Shift** キーを押しながらクリックしてドラッグし、新しく作成したポート1と2をサブシステムの周囲のより適切な場所に移動することもできます。サブシステムのラベルをブロック自体の外側に移動することもできます。
- 3 サブシステムを右クリックし、**サブシステム -> マスクの作成...(Ctrl+M)**を選択します。
- 4 **ダイアログ**タブ内で+アイコンをクリックして、新しいダイアログパラメータを作成します。直列(series)回路を使用している場合は、**プロンプト**にNumber of resistors in series、**変数**に n などを入力します。**種類**は編集のまま構いません。**OK**をクリックしてください。PLECSでサブシステムのマスクを使用してカスタムコンポーネントを作成する方法の詳細については、[サブシステムのチュートリアル](#)、またはサブシステムコンポーネントのパラメータウィンドウ内の**ヘルプ**ボタンから参照してください。

- 5 サブシステムをダブルクリックし、 n に希望する値(5など)を入力します。シミュレーションを実行すると、適切な電流/電圧を持つ5行の数値表示ブロックが生成されます。サブシステムコンポーネントのマスク内で定義する n は、そのサブシステムブロック自体内の n のインスタンスにのみ影響することに注意してください。
- 6 同じ方法で、抵抗値を定義する変数 R をマスクに追加します。

この概念は、複数のHブリッジセルを直列に実装するためのstatcom_cascaded_h_bridge_converterデモモデルにも収録してあります。PLECSで複雑なベクトル化モデルを作成するための詳細な例については、[\[1\]](#)を参照してください。

 この段階では、モデルは参照モデルimplicit_model_vectorization_9.plecsと同じになっているはずです。

3 まとめ

ベクトル化された回路は、繰り返されるコンポーネントの数を一つのパラメータで構成でき、繰り返しコンポーネントの構造を作成するのに役立ちます。ベクトル化された実装が、個別のコンポーネントを持つ標準形式よりも優れている主な利点は、ベクトル化された設計では"コピー&ペースト"エラーが発生しにくいことです。一方、このようなベクトル化されたモデルの簡略化した形式は、一見すると少し読みにくくなる可能性もあります。このチュートリアルでは、基本的なベクトル化の例と、そのような構造を作成するために使用するブロックの機能について説明しました。このチュートリアルで学んだ概念を、今後のPLECSモデルに活かしていただければ幸いです。

4 参考文献

- [1] Thierry Meynard, *Analysis and Design of Multicell DC/DC Converters Using Vectorized Models*, ISTE Ltd, 2015, pp 23-24.

改訂履歴:

Tutorial Version 1.0 初版

plexim

☎ +41 44 533 51 00

+41 44 533 51 01

✉ Plexim GmbH

Technoparkstrasse 1

8005 Zurich

Switzerland

@ info@plexim.com

<http://www.plexim.com>

Pleximへの連絡方法:

Phone

Fax

Mail

Email

Web

KESCO KEISOKU ENGINEERING SYSTEM

計測エンジニアリングシステム株式会社

<https://kesco.co.jp>

PLECS Tutorial

© 2002–2021 by Plexim GmbH

このマニュアルに記載されているソフトウェアPLECSは、ライセンス契約に基づいて提供されています。ソフトウェアは、ライセンス契約の条件の下でのみ使用またはコピーできます。Plexim GmbHの事前の書面による同意なしに、このマニュアルのいかなる部分も、いかなる形式でもコピーまたは複製することはできません。

PLECSはPlexim GmbHの登録商標です。MATLAB、Simulink、およびSimulink Coderは、The MathWorks, Inc.の登録商標です。その他の製品名またはブランド名は、それぞれの所有者の商標または登録商標です。