

MapleSim ユーザガイド

Copyright © Maplesoft, a division of Waterloo Maple Inc.

2024

MapleSim ユーザガイド

著作権

Maplesoft、MapleSim、Maple は、すべて Waterloo Maple Inc. の商標です。

© Maplesoft, a division of Waterloo Maple Inc. 2008-2024. All rights reserved.

本書のいかなる部分も、電子的、機械的、写真、記録など形式および手段を問わず、複製、検索システムへの保存、転写を行うことを禁止します。本書の情報は、予告なく変更することがあり、販売元がその内容を保証するものではありません。本書で説明しているソフトウェアは、ライセンス合意に基づいて提供されるもので、その合意に従う場合に限り使用および複製を許可します。合意で明示的に許可されている場合を除き、メディアを問わずソフトウェアをコピーすることは法律違反となります。

Linux は、Linus Torvalds の登録商標です。

Macintosh は、Apple Inc. のアメリカ合衆国および他の国で登録された商標です。

Microsoft、Excel、Windows は、Microsoft Corporation の登録商標です。

Modelica は、Modelica Association の登録商標です。

他のすべての商標は、その所有者が所有権を有します。

本書は、Maple の特別バージョンおよび DocBook を使用して作成されています。

目次

はじめに	xv
1 はじめての MapleSim	1
1.1 MapleSim での物理モデリング	1
トポロジ (非因果的) システム表現	1
数理モデルの作成と簡略化	1
拡張微分代数方程式ソルバ	2
非因果的モデリングと因果的モデリング	2
1.2 MapleSim ウィンドウ	6
1.3 基本チュートリアル: RLC 回路と DC Motor のモデリング	9
RLC 回路モデルの作成	10
コンポーネントプロパティの指定	13
プローブの追加	14
RLC 回路モデルのシミュレーション	15
シンプルな DC Motor モデルの作成	16
DC Motor モデルのシミュレーション	18
2 モデルの構築	21
2.1 MapleSim コンポーネントライブラリ	21
コンポーネントに対するヘルプトピックの表示	22
旧バージョンの MapleSim で作成したモデルの更新	23
2.2 モデルの閲覧	23
モデルツリー	23
モデルナビゲーションコントロール	27
2.3 システムにおけるコンポーネント間の相互作用の定義	27
2.4 コンポーネントプロパティの指定	29
パラメータ単位の指定	29
初期条件の指定	30
2.5 サブシステムの作成と管理	32
例: サブシステムの作成	33
サブシステムの詳細を表示する	35
モデルにサブシステムのコピーを複数追加	35
サブシステム定義と共有サブシステムの編集	39
スタンドアロンサブシステムの操作	44

2.6 グローバルパラメータとサブシステムパラメータ	47
グローバルパラメータ	47
サブシステムパラメータ	49
パラメータブロックの作成	51
パラメータセットの作成	57
パラメータと変数の詳細設定の利用	58
2.7 モデルへのファイルの添付	65
2.8 カスタムライブラリの作成と管理	66
例：既存モデルからカスタムライブラリを作成	67
2.9 モデルへの注釈の追加	70
例：モデルへのテキスト注釈の追加	70
2.10 2-D Math 表記によるテキスト入力	72
2.11 補間テーブルコンポーネントのデータセットの作成	73
例：Maple でのデータセットの作成	74
2.12 ベストプラクティス：モデルの構築	75
ベストプラクティス：サブシステムの作成と配置	75
ベストプラクティス：電気モデルの作成	76
ベストプラクティス：1-D 並進モデルの作成	78
ベストプラクティス：マルチボディモデルの作成	79
ベストプラクティス：油圧モデルの作成	80
ベストプラクティス：初期条件の強制	81
3 カスタムコンポーネントの作成	83
3.1 カスタムコンポーネントについて	83
シンプルなカスタムコンポーネントの作成	84
一般的な使用方法	86
カスタムコンポーネントテンプレートの使用	87
3.2 信号フローの動作を使用したカスタムコンポーネントの作成	87
シンプルな信号フローのカスタムコンポーネントの作成	87
カスタムコンポーネントでの微分方程式の使用	93
3.3 物理的な接続を持つカスタムコンポーネントの作成	93
抵抗のシステム方程式の抽出	94
3.4 MapleSim でのカスタムコンポーネントの使用	95
カスタムコンポーネントを現在のモデルの一部として保存する	96
カスタムコンポーネントをカスタムライブラリに追加する	96

カスタムコンポーネントを編集する	96
3.5 例: 非線形バネダンパカスタムコンポーネントの作成	97
カスタムコンポーネントテンプレートを開く	98
コンポーネントの名前と方程式の定義	98
コンポーネントのポートの定義	99
次元のチェック	102
カスタムコンポーネントの生成	103
4 モデルのシミュレーションと可視化	105
4.1 MapleSim によるモデルのシミュレーションのしくみ	105
Modelica 記述	105
モデルの記述	105
システム方程式	106
方程式の簡略化	106
積分とイベント処理	106
シミュレーション結果	107
4.2 モデルのシミュレーション	108
シミュレーションとシミュレーションオプションの設定	109
プローブ値の編集	116
シミュレーション結果比較のためのパラメータセットの保存	117
4.3 シミュレーションの進捗状況メッセージ	117
4.4 シミュレーション結果とスナップショットの管理	119
結果の保存	119
スナップショットの保存と使用	120
4.5 プロットウィンドウ設定のカスタマイズ	121
例: 複数の物理量を個別のグラフにプロットする	122
例: 1 つの物理量を別の物理量に対してプロットする	125
4.6 マルチボディモデルの可視化	128
3-D アニメーションとマルチボディの設定	129
3-D ワークスペース	132
3-D モデルの表示と閲覧	133
3-D モデルへの図形の追加	135
3-D ワークスペースでモデルを作成する	140
例: 3-D ワークスペースで二重振り子のモデルを作成して動画を再生す る	144

4.7 ベストプラクティス: モデルのシミュレーションと可視化	153
時間がかかるシミュレーションの実行には外部の C コンパイラを使用する	153
モデルの一部で生成される結果を比較する	154
5 モデルの解析と操作	155
5.1 概要	155
MapleSim Apps とテンプレート	155
Apps の操作	158
Maple ワークシートで MapleSim の方程式とプロパティを扱う	158
サブシステムの使用	159
5.2 モデルからの方程式とプロパティの取得	160
5.3 線形系の解析	161
線形解析	161
コンポーネントの作成	162
5.4 パラメータの最適化	162
5.5 モデルからの C コードの生成とエクスポート	164
MapleSim でエクスポートする場合のモデルの準備	165
コード生成 App を開く	167
サブシステムのロード	168
特定のポートに対するパラメータ値のカスタマイズ、定義、および割り当て	168
コード生成オプションの選択	170
C コードの生成と保存	172
5.6 外部 C コード/ライブラリ定義からカスタムコンポーネントを生成	173
外部 C コード/ライブラリ定義テンプレートを開く	174
C/ライブラリコードの場所とオプションの指定	174
C/ライブラリコードの場所とオプションの定義	176
C コードの生成と保存	177
5.7 MapleSim API と Maple コマンドの操作	178
5.8 Maple の埋め込みコンポーネントを扱う	178
6 MapleSim チュートリアル	181
6.1 チュートリアル 1: ギアボックス付き DC Motor のモデリング	181
DC Motor のモデルにギアボックスを追加する	182
ギアボックス付き DC Motor モデルのシミュレーション	183

DC Motor コンポーネントをグループ化してサブシステムを作成する ...	184
グローバルパラメータをモデルに代入する	185
入出力値の変更	187
6.2 チュートリアル 2: ケーブル張力コントローラのモデリング	189
ケーブル張力コントローラのモデルを作成する	190
コンポーネントプロパティの指定	192
ケーブル張力コントローラのシミュレーション	192
6.3 チュートリアル 3: 非線形ダンパのモデリング	194
バネダンパカスタムコンポーネントの生成	194
減衰係数値を与える	194
非線形ダンパモデルの作成	195
サブシステムにパラメータを代入する	200
線形バネ付き非線形ダンパモデルのシミュレーション	201
6.4 チュートリアル 4: 平面スライダクランク機構のモデリング	203
平面リンクサブシステムの作成	204
パラメータの定義と代入	207
クランクとコネクティングロッド要素の作成	208
固定フレーム、Sliding Mass、ジョイント要素を追加する	209
初期条件の指定	211
平面スライダクランク機構のシミュレーション	212
6.5 チュートリアル 5: カスタムコンポーネントテンプレートの使用	213
例: 温度依存抵抗のモデリング	215
例: コンプライアント接地と区分関数	220
例: カスタムポート	226
カスタムコンポーネントの高度な使用方法	233
6.6 チュートリアル 6: C コード/DLL カスタムコンポーネントテンプレート の使用	240
6.7 チュートリアル 7: 方程式の抽出 App の使用	247
App の説明	247
方程式の生成	250
6.8 チュートリアル 8: 油圧システムのモデリング	253
計算上の問題	253
基本的な油圧ライブラリコンポーネント	254
基本的な油圧方程式	256

シンプルな油圧ネットワークの解析	258
油圧流路の制御の概要	262
メカニカルシステムと油圧システム	263
油圧作動油の圧縮率の概要	271
流体慣性モデルの概要	273
ウォーターハンマーモデルの概要	274
油圧カスタムコンポーネントの概要	283
7 リファレンス : MapleSim のキーボードショートカット	287
用語集	295
索引	297

目次

図 1.1: 因果的モデルブロックダイアグラム	2
図 1.2: 非因果的モデルブロックダイアグラム	3
図 1.3: 通過変数と介在変数のシンプルなモデル	4
図 1.4: 通過変数と介在変数のシンプルな電気モデル	4
図 1.5: RLC 回路	5
図 1.6: RLC MapleSim 回路	6
図 1.7: MapleSim ウィンドウ	7
図 1.8: 電圧レスポンスのプロット	16
図 1.9: EMF と Inertia の接続	18
図 1.10: DC Motor のトルクとスピードのプロット	19
図 2.1: モデルツリーの [コンポーネント] ビュー	24
図 2.2: モデルツリーを使用したコンポーネントの選択	26
図 2.3: サブシステムの表示	26
図 2.4: モデルナビゲーションコントロール	27
図 2.5: Conversion Block を使用した単位の指定	30
図 2.6: 初期条件	31
図 2.7: サブシステムグループ	32
図 2.8: サブシステムの作成	34
図 2.9: 複数のサブシステムの定義	36
図 2.10: サブシステムの定義	37
図 2.11: モデルに複数のサブシステムを追加する	38
図 2.12: DC Motor サブシステム	40
図 2.13: サブシステムダイアログのコピー	46
図 2.14: 添付	66
図 2.15: 力を示す矢印の確認	79
図 2.16: 重心配置のベストプラクティス	80
図 2.17: 油圧モデル	81
図 3.1: [Apps とテンプレートを追加] タブ	84
図 3.2: カスタムコンポーネントを定義する方程式	88
図 3.3: ポートのマッピング	88
図 3.4: ポートマッピングの変数	89
図 3.5: 生成されたカスタムコンポーネント	90

図 3.6: 完成したカスタムコンポーネントモデル	91
図 3.7: 2つの質量バネダンパの方程式	93
図 3.8: 2つの質量バネダンパのポートのマッピング	93
図 3.9: 抵抗のポートのマッピング	95
図 3.10: 非線形バネダンパカスタムコンポーネント	97
図 4.1: シミュレーションプロセス	108
図 4.2: シミュレーション結果の進捗状況メッセージ	118
図 4.3: シミュレーションオプション設定のスナップショット	120
図 4.4: 保存結果パレットとスナップショット	120
図 4.5: シミュレーショングラフ	123
図 4.6: カスタムプロットウィンドウ	125
図 4.7: 1つの物理量を別の物理量に対してプロットする	127
図 4.8: 3-D ワークスペース	132
図 4.9: 3-D 表示コントロール	134
図 4.10: 二重振り子の遠近表示	134
図 4.11: 二重振り子の直交表示	135
図 4.12: 二重振り子のインプリシットジオメトリ	136
図 4.13: 添付形状	137
図 6.1: ケーブル張力コントローラ	193
図 6.2: 非線形ダンパモデル	197
図 6.3: Planar Slider-Crank Mechanism	204
図 6.4: 温度依存抵抗	219
図 6.5: 落下するボール	220
図 6.6: バウンドするボールの力学	221
図 6.7: バウンドするボール	224
図 6.8: バウンドするボールの結果	226
図 6.9: カスタムポート	228
図 6.10: カスタムポートの使用	230
図 6.11: 遠心ポンプの水頭流量曲線	234
図 6.12: 遠心ポンプのカスタムコンポーネント	237
図 6.13: Windows 用外部 C コード定義	241
図 6.14: Unix 用外部 C コード定義	242
図 6.15: 引数テーブル	243
図 6.16: パイプの流れ	259

図 6.17: 流路の制御	263
図 6.18: 固定流量源	264
図 6.19: 固定圧力源を使用した並進運動	266
図 6.20: 固定圧力源の結果	268
図 6.21: 固定並進フランジ油圧コンポーネント	269
図 6.22: 固定回転フランジ油圧コンポーネント	269
図 6.23: パスカルの原理の例	271
図 6.24: 油圧作動油の圧縮率	272
図 6.25: 流体慣性を使用しないシステム	273
図 6.26: 流体慣性を使用するシステム	274
図 6.27: 流体慣性を使用する場合と使用しない場合のシステム	274
図 6.28: ウォーターハンマー	276
図 6.29: 離散化されたパイプラインのセグメント	277
図 6.30: ウォーターハンマーの圧力の流量	280
図 6.31: アキュムレータを持つ場合の圧力サージ	283
図 6.32: 水頭流量	284
図 6.33: 遠心ポンプのカスタムコンポーネントの方程式	284
図 6.34: 重力ヘッドのカスタムコンポーネントの方程式	285

表目次

表 1.1: 通過変数と介在変数のドメインタイプ	4
表 1.2: MapleSim ウィンドウコンポーネント	7
表 2.1: MapleSim コンポーネントライブラリ	21
表 2.2: モデルナビゲーションコントロール	27
表 2.3: ドメイン固有の結線の色	28
表 2.4: 2-D Math 表記のキーの組み合わせ	73
表 3.1: ポートのマッピング	89
表 3.2: 信号フローコンポーネント	91
表 3.3: 通過変数と介在変数の特徴	94
表 3.4: 通過変数と介在変数の数学的關係	94
表 3.5: 抵抗の変数とパラメータ	94
表 4.1: シミュレーションの設定	110
表 4.2: シミュレーションオプション設定	113
表 4.3: アニメーション設定	130
表 4.4: マルチボディのパラメータ値	131
表 4.5: 3-D アニメーションのパラメータ値	131
表 4.6: 3-D ワークスペースのコントロール	133
表 5.1: MapleSim Apps	156
表 5.2: MapleSim テンプレート	158
表 6.1: 温度依存抵抗コンポーネント	218
表 6.2: バウンドするボールのマルチボディコンポーネント	225
表 6.3: 遠心ポンプのデータ	234
表 6.4: 円形パイプのパラメータ	235
表 6.5: 遠心ポンプのコンポーネント	236
表 6.6: 外部 C コード DLL カスタムコンポーネントと必要な設定	245
表 6.7: 基本的な油圧ライブラリコンポーネント	255
表 6.8: Bernoulli と Darcy の式の表記	257
表 6.9: 円形パイプのパラメータ	258
表 6.10: 油圧コンポーネントと必要な設定	260
表 6.11: スプール弁	262
表 6.12: 固定流量源を使用した並進運動	265
表 6.13: 固定圧力源を使用した並進運動	267

表 6.14: マルチボディコンポーネントの動作	269
表 6.15: 油圧作動油の圧縮率のコンポーネント	272
表 6.16: 密閉された油圧システムコンポーネント	272
表 6.17: 流体慣性	273
表 6.18: Fluid Properties の値	277
表 6.19: ウォーターハンマーのパラメータ	278
表 6.20: アクキュムレータパラメータのカスタムコンポーネント	282
表 7.1: モデルの開閉と保存	287
表 7.2: ブロックダイアグラムでのモデルの作成	287
表 7.3: ブロックダイアグラムでのモデルの表示	288
表 7.4: 3-D 表示でのモデルの表示	289
表 7.5: モデルのシミュレーション	291
表 7.6: コンソールペインのナビゲーション	292
表 7.7: [プロットウィンドウ] のレイアウト変更	292
表 7.8: Modelica カスタムコンポーネントの編集	292
表 7.9: その他	294

はじめに

MapleSim の概要

MapleSimTM は、複雑なマルチドメイン物理システムを作成してシミュレーションを行うためのモデリング環境です。物理システムをグラフィック形式で表すコンポーネントダイアグラムを作成することができます。数式処理および数値計算の両方のアプローチを使用することで、MapleSim はコンポーネントダイアグラムから自動的にモデルの数式を生成し、精度の高いシミュレーションを実行します。

複雑なマルチドメインモデルの作成

MapleSim を使用すると、さまざまな工学分野のコンポーネントを1つのシステムに統合するモデルを作成できます。MapleSim には、700 以上ものモデリングコンポーネントのライブラリが用意されています。これには、電気、油圧、メカニカル、伝熱といったデバイス、センサやソース、および信号ブロックなどが含まれます。さらに、モデリングやシミュレーションに対するユーザーのニーズに合わせてカスタムコンポーネントを作成することもできます。

高度な数式処理および数値計算機能

MapleSim は、MapleTM の高度な数式処理機能および数値処理機能を使用して、物理システムの挙動をシミュレーションする数理モデルを生成します。そのため、数式に簡略化テクニックを適用して簡潔かつ数値的効率の高いモデルを作成することができます。

組み込みの分析ツールとテンプレート

MapleSim には、Maple ワークシート形式のさまざまな Apps とテンプレートが用意されています。これらのテンプレートを使用して、モデル数式を表示したり、パラメータ最適化などの高度な解析タスクを実行したりすることができます。モデルを解析してシミュレーション結果を対話形式で表示するには、埋め込みコンポーネント、プロットツール、ドキュメント作成ツールなどの Maple 機能を使用できます。また、モデルを C コードに変換して別のアプリケーションや

ツールを使用して作業することもできます。これには、リアルタイムでシミュレーションを実行できるアプリケーションなどが含まれます。

対話型 3-D 可視化ツール

MapleSim の 3-D アニメーション環境では、マルチボディメカニカルシステムモデルの 3 次元グラフィック表現を構築してアニメーションを作成することができます。この環境を使用して、モデルの 3-D 設定を検証したり、条件やシミュレーション開始時間を変更した場合のシステムの挙動を視覚的に解析したりすることができます。

関連製品

MapleSim 2024.1 を使用するには、Maple 2024.1 が必要です。

また、MaplesoftTM は、工学設計プロジェクト向けに、Maple および MapleSim の機能を拡張するツールボックス、アドオン、その他のアプリケーションを提供しています。これらの製品の一覧については、

<http://www.maplesoft.com/products> を参照してください。

関連リソース

リソース	説明
MapleSim インストールガイド	MapleSim のシステム要件およびインストール手順。 MapleSim インストールガイド は、MapleSim インストール DVD の Install.html ファイルに含まれています。
MapleSim ヘルプシステム	次の情報が提供されています。 <ul style="list-style-type: none"> • MapleSim ユーザガイド : MapleSim に関する概念的な情報、MapleSim 機能の概要、および始めて使用する際に役立つ例題など。 • MapleSim 操作方法 : モデルの構築、シミュレーション、解析タスクに関するヘルプ。 • MapleSim コンポーネントライブラリ : MapleSim で利用できるモデリングコンポーネントの説明。
MapleSim 例題	さまざまなエンジニアリング分野からのモデルの例。[ヘルプ]メニューから[例題]を選択してこれらの例題にアクセスします。
MapleSim ユーザガイドの例題	ユーザガイドで使用されているモデルと例題。これらの例題にアクセスするには、[ヘルプ]メニューから[例題]>[ユーザガイドの例題]を選択してください。例題は章ごとにリストされており、ユーザガイドで使用される順に並んでいます。
MapleSim オンラインリソース	トレーニングウェビナー、製品デモ、ビデオ、サンプルアプリケーションなど。 詳細は、 http://www.maplesoft.com/products/maplesim を参照してください。
MapleSim モデルギャラリー	サンプルモデル、カスタムコンポーネント、解析テンプレートの集合。これらは、ダウンロードしてユーザの MapleSim プロジェクトで使用することができます。 詳細は、 http://www.maplesoft.com/products/maplesim/modelgallery/ を参照してください。

追加情報については、http://www.maplesoft.com/site_resourcesを参照してください。

サポートについて

カスタマーサポートまたはテクニカルサポートを受けるには、<http://www.maplesoft.com/support>を参照してください。

カスタマーフィードバック

Maplesoftはユーザからのフィードバックを歓迎します。MapleSim製品のドキュメントに関連するコメントは、doc@maplesoft.comまでご連絡ください。

第1章 はじめての MapleSim

この章の内容は以下のとおりです。

- *MapleSim* での物理モデリング [1ページ]
- *MapleSim* ウィンドウ [6ページ]
- 基本チュートリアル: RLC 回路と DC Motor のモデリング [9ページ]

1.1. MapleSim での物理モデリング

物理モデリング (物理ベースのモデリング) は、数学と物理法則を組み合わせ、エンジニアリングにおけるコンポーネントまたは相互接続したコンポーネントシステムの動作を記述します。大部分のエンジニアリングシステムは力学と関連しているため、その動作は一般的に常微分方程式 (ODE) を使用して定義されます。

モデルを速く簡単に作成できるように、MapleSim は以下の機能を備えています。

トポロジ (非因果的) システム表現

従来のモデリングツールで使用される信号フローアプローチでは、システムの入出力を明確に定義する必要があります。一方、MapleSim では、トポロジ表現を使用することにより、信号のフローを考慮する必要なしに、関連コンポーネントを接続することができます。

数理モデルの作成と簡略化

トポロジ表現は、容易に数学表現に対応付けられ、MapleSim の記号機能によってシステム方程式が自動的に生成されます。

MapleSim がシステム方程式を生成するとき、冗長な方程式および、0 や 1 による乗算は数式簡略化ツールによってすべて取り除かれます。そして式は、まとめたり、減らしたりされ、正確さを失わずにシステムを表現する最小集合の方程式になります。

拡張微分代数方程式ソルバ

トポロジアプローチでは、モデル定義に代数制約が取り入れられています。ODEと代数制約を組み合わせた問題は微分代数方程式 (DAE) と呼ばれます。制約の性質によって DAE の問題の複雑度が変わる場合があります。問題の複雑度は DAE のインデックスで示されます。DAE のインデックスが大きくなると、複雑度も高くなります。

複雑な DAE のための汎用ソルバの開発は、記号計算の分野における多くの研究のテーマです。計算エンジンとして Maple を採用している MapleSim は、高インデックス DAE を解く、最新の記号技術と数値技術を集結した高度な DAE ソルバを実現しています。

非因果的モデリングと因果的モデリング

モータやパワートレインなど本物の機械は相互作用する一連の物理コンポーネントから成り立っており、一般的に、ブロックダイアグラムを使用してモデリングされます。2つのブロックがラインで接続されている場合、これらは物理法則によって結ばれていることを示します。ソフトウェアによるシミュレーションについて、ブロックダイアグラムは因果的または非因果的に分類することができます。

因果的モデリング

多くのシミュレーションツールは因果的 (信号フロー) モデリングに限定されています。これらのツールでは、基本的に時変数値である単方向信号がブロックの方向に流れます。ブロックでは、信号に対しあらかじめ定義されている数学操作が実行され、結果はブロックの反対側から出力されます。このアプローチは、制御システムやデジタルフィルタなど信号が単方向にしか進まないモデリングシステムに適しています。

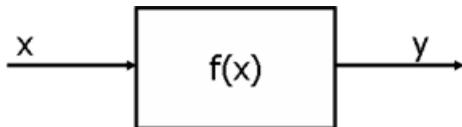


図1.1 因果的モデルブロックダイアグラム

このアプローチは代入に似ており、右辺で既知変数または変数集合に関する計算が行われ、結果が左辺の別の変数に代入されます。

$$y := f(x)$$

非因果的モデリング

実際の物理コンポーネントがどのように相互作用するかをモデリングするには別のアプローチが必要です。非因果的モデリングでは、接続された2つのブロックからの信号は両方向に流れます。プログラミングでは、単純な等式に似ているといえます。

$$y = f(x)$$

信号には、エネルギーや電流、回転力、熱流量、質量流など、どの物理量を保存する必要があるかなどの情報が含まれます。そしてブロックには、従う必要がある物理法則(方程式)などの情報、つまり、保存する必要がある物理量が含まれません。

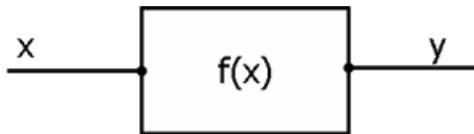


図1.2 非因果的モデルブロックダイアグラム

MapleSim では、両方のアプローチを使用することができます。たとえば、関連するロジックとのシミュレーションには物理システムを非因果的モデリングで表し、制御ループのシミュレーションには因果的モデリングを使用するなど、実行するタスクに最適なアプローチを選択することができます。

通過変数と介在変数

非因果的モデリングアプローチを使用する場合、モデリングしているコンポーネントの通過変数と介在変数の確認が有効です。一般的に、介在変数はシステム内の原動力を表し、通過変数は保存量の流れを表します。通過変数は、保存量の符号の規則で流れ方向も決定します。

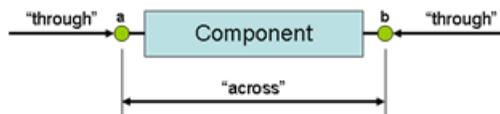


図1.3 通過変数と介在変数のシンプルなモデル

符号の規則の例および矢印の方向がモデル上で作用する力をどのように表すかについては、[ヘルプ]メニューの[例題]>[ユーザガイドの例題]>[第1章]から[Constant Acceleration]、[Sign Convention]、および[Arrow Convention]の各例題を参照してください。

次の例では、電子回路の通過変数 i は電流、介在変数 V は電圧降下を表しています。

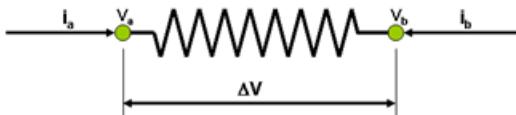


図1.4 通過変数と介在変数のシンプルな電気モデル

下の表は、ほかのドメインにおける通過変数と介在変数の例です。

通過変数と介在変数のドメインタイプ

ドメイン	通過	介在
電気	電流 (A)	電圧 (V)
磁気	磁束 (Wb)	MMF (A)
メカニカル (並進)	力 (N)	位置 ($\frac{m}{s}$)
メカニカル (回転)	トルク ($N.m$)	角度 ($\frac{rad}{s}$)
油圧	流量 ($\frac{m^3}{s}$)	圧力 ($\frac{N}{m^2}$)
熱流量	熱流量 (W)	温度 (K)

簡単な例を挙げると、抵抗の支配方程式は以下のように表されます。

$$V = R \cdot i$$

上記方程式に Kirchhoff の電流保存の法則を組み合わせると、回路を完全に表現することができます。

$$R \cdot i_b = V_b - V_a \text{ および } i_b + i_a = 0$$

この例を拡張した下の回路図は、Resistor (抵抗)、Inductor (インダクタ) と Capacitor (コンデンサ) が直列に接続された電気回路、RLC 回路です。

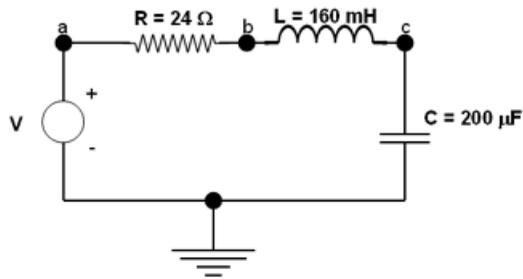


図1.5 RLC 回路

この回路を手動でモデリングする場合、抵抗、インダクタ、およびコンデンサはそれぞれ以下の特性方程式で表すことができます。

$$R \cdot i_R = V_a - V_b$$

$$L \frac{d}{dt} i_L = (V_b - V_c)$$

$$i_c = C \cdot \frac{d}{dt} V_c$$

Kirchhoff の法則を適用すると、それぞれポイント a、ポイント b、ポイント c について以下の保存方程式が成り立ちます。

$$i_V - i_R = 0$$

$$i_R - i_L = 0$$

$$i_L - i_C = 0$$

これらの方程式を、入力電圧の定義 (シミュレーションを開始して 1 秒後に 0 ボルトから 1 ボルトへ変動と定義) と合わせると、以下の式になります。

$$V_a = \begin{cases} 0.0 & 0.0 \leq t < 1.0 \\ 1.0 & t \geq 1.0 \end{cases}$$

この式からは、モデルを定義し、回路を流れる電流と電圧を求めるために十分な情報が得られます。

MapleSim では、これらの計算がすべて自動的に行われます。つまり、ユーザは回路を描き、コンポーネントのパラメータを指定するだけで十分です。この原則は、MapleSim における全エンジニアリングドメインにおいて同じように適用され、あるドメインのコンポーネントを簡単に別のドメインのコンポーネントと接続することができます。

基本チュートリアル: RLC 回路と DC Motor のモデリング [9ページ]では、上述の RLC 回路をモデリングして、MapleSim の因果的モデルと非因果的モデルを組み合わせる機能について説明しています。以下の図は、作成した RLC 回路図が MapleSim でどのように表示されるかを示しています。

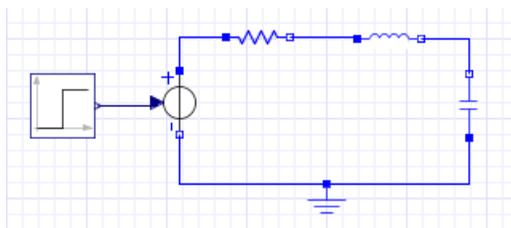


図1.6 RLC MapleSim 回路

因果的コンポーネントと非因果的コンポーネントを使用してモデルを表す別の方法については、[ヘルプ]メニューの [例題] > [ユーザガイドの例題] > [第 1 章] から [Double Mass Spring Damper] の例題を参照してください。

1.2. MapleSim ウィンドウ

MapleSim ウィンドウは、以下のペインとコンポーネントで構成されています。

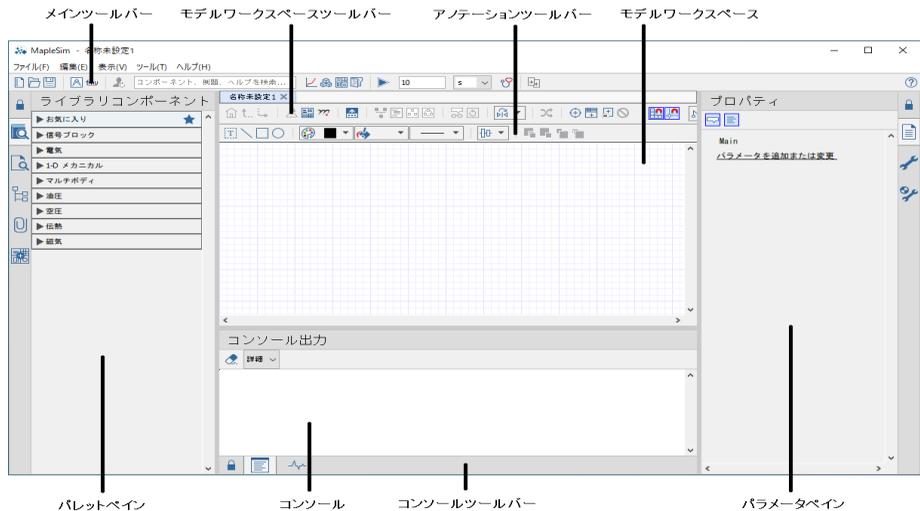


図1.7 MapleSim ウィンドウ

MapleSim ウィンドウコンポーネント

コンポーネント	説明
メインツールバー	シミュレーションを実行するツール、シミュレーション結果を表示するツール、MapleSim ヘルプシステムを検索するツール、その他の一般的なタスクを実行するツールなどが配置されています。
モデルワークスペースツールバー	モデルやサブシステムを階層的に参照したり、モデルの表示を変更したりするツールが配置されています。また、対応する Modelica コードを表示するためのツールや、コンポーネントをグループ化したり、プローブを追加するツールもあります。
アノテーションツールバー	注釈を追加し、オブジェクトを配置するためのツールが含まれます。
モデルワークスペース	ブロックダイアグラムでモデルの構築と編集を行う領域です。

コンポーネント	説明
パレットペイン	<p>モデルの作成と、MapleSim プロジェクトの管理に使用できるツールを含む展開可能なメニューが配置されています。このペインは次の5つのタブで構成されています。</p> <ul style="list-style-type: none"> • [ライブラリコンポーネント] (): モデルの例題およびモデルに追加できるドメイン特有のコンポーネントを含むパレットが配置されています。 • [ローカルコンポーネント] (): モデル内のサブシステムおよびカスタムコンポーネントのパレットが配置されています。 • [モデルツリー] (): モデル内のナビゲーション用のモデルツリーが配置されています。 • [添付ファイル] (): ドキュメント、パラメータセット、CAD 図面などを含むモデルの添付が配置されています。 • [Apps とテンプレートを追加] (): モデルの構築と解析用のプリビルトツールを備えたパレットが配置されています。
コンソール	<p>コンソールツールバー上のボタンを使用すると、以下のペインを表示できます。</p> <ul style="list-style-type: none"> • [コンソール出力]: シミュレーション中に MapleSim エンジンの状態を示す進捗状況メッセージを表示します。メッセージコンソールの消去 () を使用するとコンソールを消去することができます。 • [診断情報]: モデルの構築時にエラーのあるサブシステムを特定し、デバッグのために診断メッセージを表示します。
コンソールツールバー	<p>コンソールに表示するメッセージタイプを選択するためのコントロール () が含まれます。</p>

コンポーネント	説明
パラメータペイン	<p>以下のタブで構成されています。</p> <ul style="list-style-type: none"> • [プロパティ] (📄): 名前やパラメータ値、初期条件、プローブ設定などのモデリングコンポーネントのプロパティを表示および編集することができます。 • [シミュレーションの設定] (🔧): シミュレーション時間や、ソルバおよびシミュレーションエンジンの任意のパラメータ値などのシミュレーションのオプションを指定できます。 • [マルチボディ設定] (🔧): 3-D ワークスペース および [3-D アニメーションウィンドウ] でマルチボディコンポーネントを可視化するためのオプションを指定できます。 <p>このペインの内容はモデルワークスペースでの選択に応じて変わります。</p>

1.3. 基本チュートリアル: RLC 回路と DC Motor のモデリング

このチュートリアルでは、MapleSim のモデリングコンポーネントおよび基本ツールを紹介し、因果的モデルと非因果的モデルを組み合わせる機能について説明します。

このチュートリアルでは、以下のタスクを実行します。

1. RLC 回路モデルを作成します。
2. コンポーネントのプロパティを指定するパラメータ値を設定します。
3. シミュレーションで確認するデータの値を読むプローブを追加します。
4. RLC 回路モデルをシミュレートします。
5. RLC 回路ダイアグラムを修正し、シンプルな DC motor モデルを作成します。
6. さまざまなパラメータ値で DC motor モデルをシミュレートします。

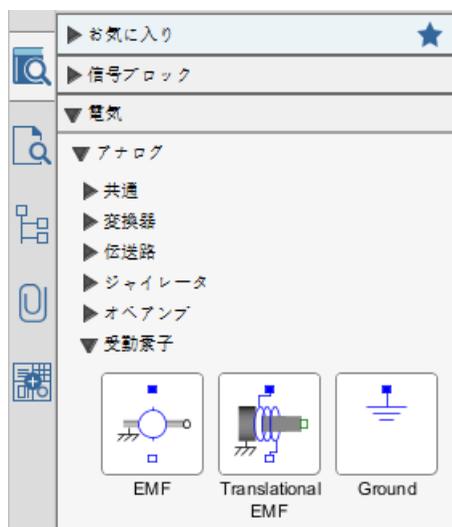
RLC 回路モデルの例は、[ヘルプ]メニューの[例題]>[ユーザガイドの例題]>[第1章]から[RLC Circuit]の例題を参照してください。作成するモデルはRLC 回路モデルと同じです。

RLC 回路モデルの作成

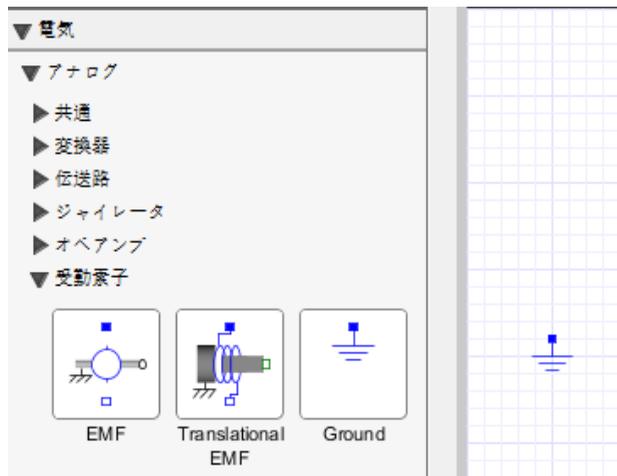
RLC 回路を作成するには、コンポーネントをモデルワークスペースに追加し、システム内の各コンポーネントを接続してダイアグラムを形成します。このRLC 回路モデルの例では、電気コンポーネントライブラリの Ground、Resistor、Inductor、Capacitor、Signal Voltage ソースコンポーネントを使用します。また、Step 入力ソースも使用しますが、これは回路内で入力電圧レベルを動かす信号発生器です。

RLC 回路を作成するには

1. モデルワークスペースの左側にある[ライブラリコンポーネント]タブ(🔍)で、[電気]の隣の三角形をクリックしてパレットを展開します。同様に、[アナログ]メニューを展開し、[受動素子]サブメニューを開きます。



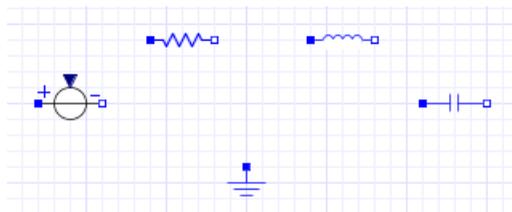
2. [電気]>[アナログ]>[受動素子]メニューから、**Ground** コンポーネントをモデルワークスペースにドラッグします。



3. モデルワークスペースに以下の電気コンポーネントを追加します。

- [電気] > [アナログ] > [受動素子] > [抵抗] メニューから、**Resistor** コンポーネントを追加します。
- [電気] > [アナログ] > [受動素子] > [インダクタ] メニューから、**Inductor** コンポーネントを追加します。
- [電気] > [アナログ] > [受動素子] > [コンデンサ] メニューから、**Capacitor** コンポーネントを追加します。
- [電気] > [アナログ] > [ソース] > [電圧] メニューから、**Signal Voltage** コンポーネントを追加します。

4. 各コンポーネントをドラッグし、下の図のように配置します。



5. **Signal Voltage** コンポーネントを時計回りに回転させるには、**モデルワークスペース**で **Signal Voltage** コンポーネントを右クリック (Macintosh では **[Control]** キーを押しながらクリック) し、**[時計回りに回転]** を選択します。

6. **Signal Voltage** コンポーネントを左右に反転させるには、もう一度コンポーネントを右クリック (Macintosh では **[Control]** キーを押しながらクリック) し、**[水平方向に反転]** を選択します。プラスのポート (青色) が上にあることを確認します。

7. **Capacitor** コンポーネントを時計回りに回転させるには、**モデルワークスペース** で **Capacitor** コンポーネントを右クリック (Macintosh では **[Control]** キーを押しながらクリック) し、**[時計回りに回転]** を選択します。

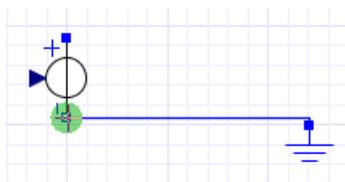
次に、モデリングコンポーネントを接続し、システムにおける相互関係を定義します。

8. マウスポインタを **Ground** コンポーネントのポート上に移動させます。ポートは緑で強調表示されます。



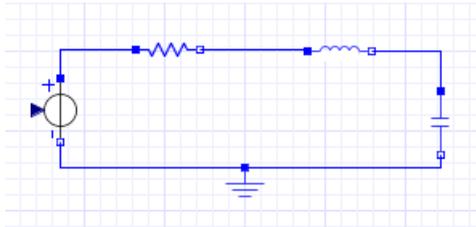
9. **Ground** 入力ポートをクリックし、結線を引きはじめます。

10. マウスポインタを **Signal Voltage** コンポーネントのマイナスポート上に移動させます。



11. ポートを 1 回クリックします。 **Ground** コンポーネントと **Signal Voltage** コンポーネントが接続されます。

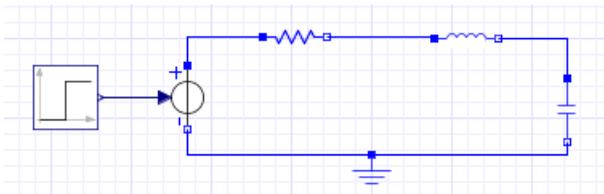
12. 残りのコンポーネントを下の図のように接続します。



13 次に、モデルにソースを追加します。[信号ブロック] パレットを展開して、[ソース] メニューから [実数] サブメニューを選択します。

14 パレットから [Step] ソースをドラッグして、モデルワークスペースの **Signal Voltage** コンポーネントの左側に配置します。Step ソースには明確な信号フローがあり、結線上の矢印で表されます。このフローによって回路は入力信号に反応します。

15 **Step** ソースを **Signal Voltage** コンポーネントに接続します。完成した RLC 回路モデルを下の図に示します。

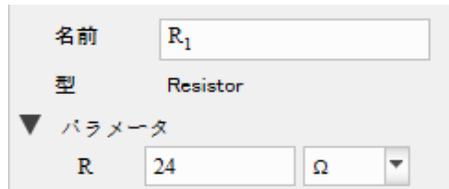


コンポーネントプロパティの指定

コンポーネントのプロパティは、モデルのコンポーネントにパラメータ値を設定することで指定できます。

コンポーネントプロパティを指定するには

1. モデルワークスペースで **Resistor** コンポーネントをクリックします。モデルワークスペースの右側にある [プロパティ] タブ (📄) に、抵抗の名前とパラメータ値が表示されます。
2. [R] フィールドに「24」を入力し、[Enter] キーを押します。これで抵抗が 24Ω に変更されます。



3. ほかのコンポーネントに以下のパラメータ値を指定します。パラメータの単位を指定する場合は、パラメータ値フィールド横のドロップダウンメニューからその値を選択します。

- **Inductor** のインダクタンスに 160 mH を指定します。
- **Capacitor** のキャパシタンスに $200\mu\text{F}$ を指定します。
- **Step** ソースの $[T_0]$ の値に 0.1 s を指定します。

プローブの追加

シミュレーションのためのデータ値を指定するには、モデルのラインまたはポートにプローブを追加します。例として、RLC 回路の電圧を測ります。

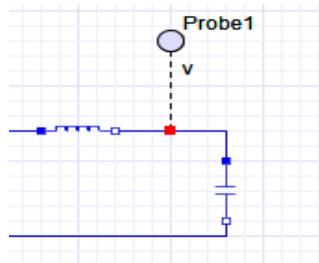
プローブを追加するには

1. モデルワークスペースツールバーの**プローブを追加**ボタン (⊕) をクリックします。
2. マウスポインタを、**Inductor** コンポーネントと **Capacitor** コンポーネントを接続しているライン上に移動させます。ラインが強調表示されます。
3. ラインを1回クリックします。**モデルワークスペース**にプローブが表示されます。
4. **モデルワークスペース**の空白の場所にプローブを移動し、ワークスペースをクリックしてプローブを配置します。
5. プローブを選択します。**モデルワークスペース**の右側にある**[プロパティ]**タブ (📄) にプローブのプロパティが表示されます。
6. **[プロパティ]** タブの **[Voltage]** チェックボックスを選択して、シミュレーショングラフに電圧量を組み込みます。

7. この量にカスタム名を付けて**モデルワークスペース**に表示するには、下図のように「**Voltage**」と入力して、**[Enter]** キーを押します。



名前が付けられたプローブが結線に追加されます。



シミュレーションでプローブ値を使用する別の例については、**[ヘルプ]** メニューの **[例題]** > **[ユーザガイドの例題]** > **[第 1 章]** から **[Sensors and Probes]** の例題を参照してください。

RLC 回路モデルのシミュレーション

モデルのシミュレーションを開始する前に、シミュレーション時間を指定できます。

RLC 回路のシミュレーションを実行するには

1. **[パラメータ]** ペインの右側にある **[シミュレーションの設定]** タブ (🔧) をクリックし、**[Simulation]** セクションでシミュレーション時間 (t_d) を 0.5 s に設定します。
2. **[オプション]** セクションで **[Compiler]** チェックボックスをオフにします。

3. **メインツールバーでシミュレーションの実行**(▶)をクリックします。MapleSimによってシステム方程式が生成され、ステップ入力に対するレスポンスがシミュレートされます。

シミュレーションが完了すると、電圧レスポンスがグラフ上にプロットされます。

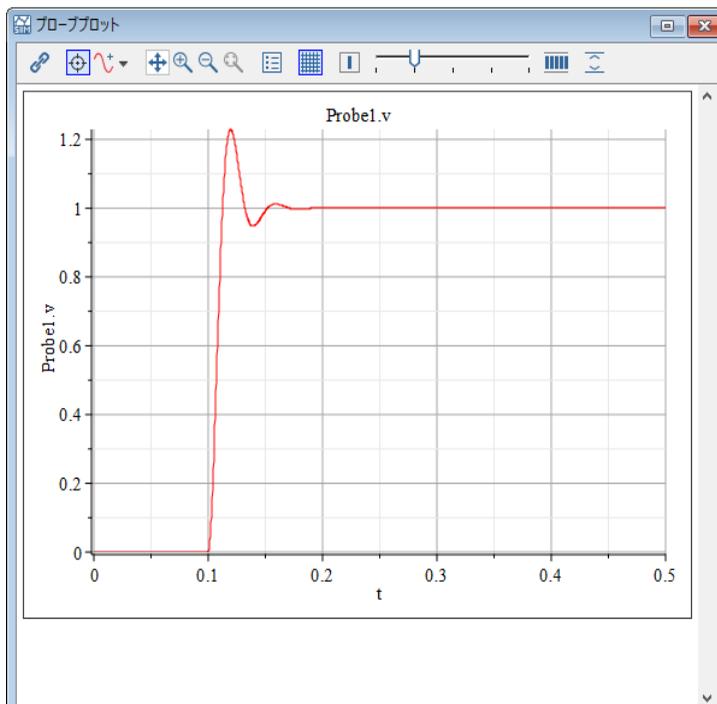


図1.8 電圧レスポンスのプロット

4. モデルを「**RLC_Circuit1.msimsim**」という名前で保存します。プローブおよび変更したパラメータ値はモデルの一部として保存されます。

シンプルな DC Motor モデルの作成

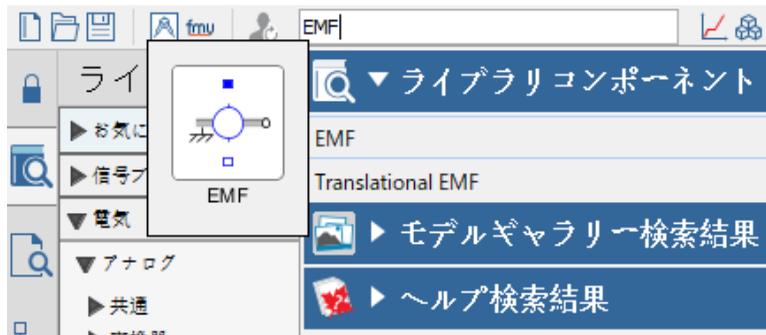
次に、起電力 (EMF) とメカニカル Inertia コンポーネントを RLC 回路モデルに追加し、DC Motor モデルを作成します。この例では、検索機能を使用して RLC 回路モデルにコンポーネントを追加します。

シンプルな DC Motor モデルを作成するには

1. メインツールバーの検索バーに、「EMF」と入力します。ドロップダウンリストに検索結果が表示されます。



2. ドロップダウンリストの [ライブラリコンポーネント] セクション内にある [EMF] の上にカーソルを合わせます。EMF コンポーネントが検索フィールド横の四角形のツール領域内に表示されます。



3. EMF コンポーネントをモデルワークスペースにドラッグして、Capacitor コンポーネントの右側に配置します。
4. 検索バーに「Inertia」と入力します。
5. モデルワークスペースに Inertia コンポーネントをドラッグして EMF コンポーネントの右側に配置します。
6. 各コンポーネントを下の図のように接続します。

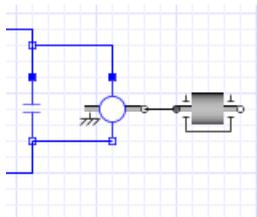


図1.9 EMF と Inertia の接続

注: EMF コンポーネントのプラスのポート (青色) を接続するには、ポートを1回クリックし、マウスポインタを Capacitor (コンデンサ) と Inductor (インダクタ) を接続するラインにドラッグして、ラインをクリックします。

7. モデルワークスペースで EMF コンポーネントをクリックします。
8. [プロパティ] タブ (📄) で変換係数 (k) の値を「 $10 \frac{N \cdot m}{A}$ 」に設定します。
9. Step コンポーネントをクリックし、パラメータ $[T_0]$ の値を 1 s に変更します。

DC Motor モデルのシミュレーション

DC Motor モデルのシミュレーションを実行するには

1. モデルワークスペースで Probe1 を削除します。
2. モデルワークスペースツールバーのプローブを追加ボタン (⊕) をクリックします。
3. マウスポインタを EMF コンポーネントと Inertia コンポーネントを接続するライン上に移動させます。
4. ラインをクリックし、次にワークスペースの空白部分をクリックしてプローブを配置します。
5. プローブを選択して、[プロパティ] タブ (📄) をクリックし、[Speed] チェックボックスと [Torque] チェックボックスにチェックを付け、[Angle] チェックボックスのチェックを外します。保存量のフロー方向を示す矢印がついたプローブがモデルに追加されます。プローブを選択して[プロパティ] タブの **プローブの反転** (↺) をクリックすると、保存量のフロー方向 (Torque) を逆にすることができます。

6. プロープ名 **Output** を変更します。
7. モデルワークスペースで空白部分をクリックします。
8. [シミュレーションの設定] タブ (🔧) で、シミュレーション時間 (t_d) を 5 s に設定します。
9. メインツールバーでシミュレーションの実行 (▶) をクリックします。
10. シミュレーション結果を表示 (📈) をクリックします。次のグラフが表示されます。

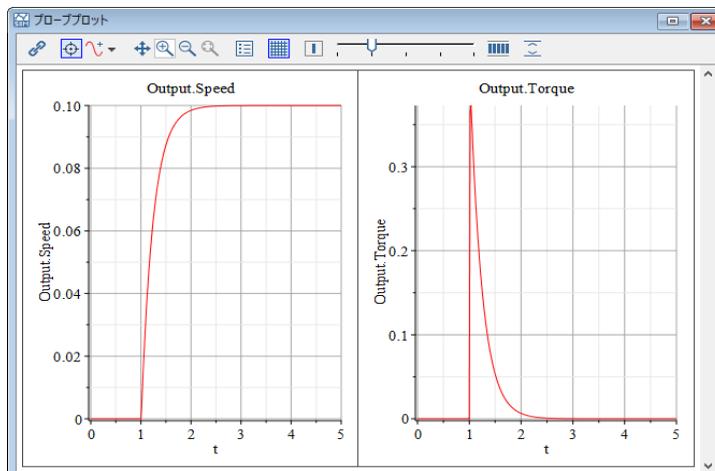


図1.10 DC Motor のトルクとスピードのプロット

11. モデルを「DC_Motor1.msim」という名前で保存します。

第2章 モデルの構築

この章の内容は以下のとおりです。

- *MapleSim* コンポーネントライブラリ [21ページ]
- モデルの閲覧 [23ページ]
- システムにおけるコンポーネント間の相互作用の定義 [27ページ]
- コンポーネントプロパティの指定 [29ページ]
- サブシステムの作成と管理 [32ページ]
- グローバルパラメータとサブシステムパラメータ [47ページ]
- モデルへのファイルの添付 [65ページ]
- カスタムライブラリの作成と管理 [66ページ]
- モデルへの注釈の追加 [70ページ]
- 2-D Math 表記によるテキスト入力 [72ページ]
- 補間テーブルコンポーネントのデータセットの作成 [73ページ]
- ベストプラクティス: モデルの構築 [75ページ]

2.1. MapleSim コンポーネントライブラリ

MapleSim コンポーネントライブラリには、モデルを構築するための 500 以上のコンポーネントが用意されています。これらのコンポーネントはすべて、ドメイン(電気、磁気、油圧、1-D メカニカル、マルチボディ、信号ブロック、伝熱)ごとにパレットにまとめられています。ほとんどのコンポーネントは Modelica Standard Library 3.2.3 をベースとしています。

MapleSim コンポーネントライブラリ

ライブラリ	説明
信号ブロック	入力および出力信号を操作または生成するためのコンポーネント
電気	アナログの電気回路や単相および多相システム、電気機械などをモデリングするためのコンポーネント

ライブラリ	説明
1-D メカニカル	1-D並進および1-D回転システムをモデリングするためのコンポーネント
マルチボディ	力、運動およびジョイントコンポーネントを含むマルチボディメカニカルシステムをモデリングするためのコンポーネント
油圧	流体動力システムやシリンダ、アクチュエータなどの油圧システムをモデリングするためのコンポーネント
空圧	シリンダ、方向制御弁、オリフィス、アクチュエータによって空圧システムをモデリングするためのコンポーネント
伝熱	熱流量および熱伝達をモデリングするためのコンポーネント
磁気	磁気回路をモデリングするためのコンポーネント

ライブラリには、完全な電気回路やフィルタなど、参照およびシミュレート可能なサンプルモデルも用意されています。MapleSim ライブラリの構造とモデリングコンポーネントに関する詳細は、MapleSim ヘルプシステムの **MapleSim コンポーネントライブラリ**を参照してください。

デフォルトのライブラリを拡張するために、数理モデルからカスタムコンポーネントを作成し、カスタムライブラリに追加することもできます。詳細は、*カスタムコンポーネントの作成 [83ページ]*を参照してください。

コンポーネントに対するヘルプトピックの表示

MapleSim ヘルプシステムでヘルプトピックを表示するには、以下のいずれかの操作を実行します。

- いずれかのパレットでモデリングコンポーネントを右クリック (Mac では **[Control]** キーを押しながらクリック) し、コンテキストメニューから **[ヘルプ]** を選択します。
- メインツールバーのヘルプ検索ボックスを使用して任意のトピックを検索します。入力したキーワードに関連するヘルプトピックが **[ヘルプ検索結果]** セクションに表示されます。
- MapleSim ヘルプシステムでコンポーネントのヘルプページを検索します。

旧バージョンの MapleSim で作成したモデルの更新

MapleSim 7 以前のバージョンでは、Modelica Standard Library 3.1 のコンポーネントが MapleSim コンポーネントライブラリに含まれていましたが、MapleSim 2015 では Modelica Standard Library 3.2.1、MapleSim 2019 では Modelica Standard Library 3.2.3 に更新されました。

以前のバージョンの MapleSim で作成したモデルは、MapleSim 2019 で開くことができます。その際には、対応する Modelica Standard Library 3.2.3 のコンポーネントを使うように自動的に更新されます。MapleSim 7 以前で作成したファイルのバックアップ方法などの詳細については、MapleSim ヘルプシステムで、**MapleSim 操作方法 > 旧バージョンの MapleSim で作成したモデルの更新**を参照してください。

2.2. モデルの閲覧

[**モデルツリー**] またはモデルのナビゲーションコントロールを使用してモデルを参照すると、**モデルワークスペース**に表示されているコンポーネントの階層レベルを閲覧することができます。トップレベルを閲覧すると、システムの全体図が表示されます。トップレベルは、モデルにおける最上位のレベルで、個別モデリングコンポーネントやコンポーネントの組み合わせを表すサブシステムブロックを含む、システム全体を表します。また、モデルのサブレベルを閲覧し、個々のサブシステムの詳細やコンポーネントを表示することもできます。

モデルツリー

[**モデルツリー**] タブ () はパレットペインに配置されています。**モデルツリー**を使用して、モデルを閲覧したり、モデルの要素を検索したりすることもできます。モデルツリーのノードは、モデルツリービューに応じて、添付、コンポーネントタイプ、コンポーネント、パラメータ、またはプローブを表します。モデルツリービューを変更するには、[**検索**] ボックスの下にあるリストからビューの種類を選択します。以下の図は、[5 DoF Robot] のマルチボディモデルで利用可能なドロップダウンメニューを示しています。

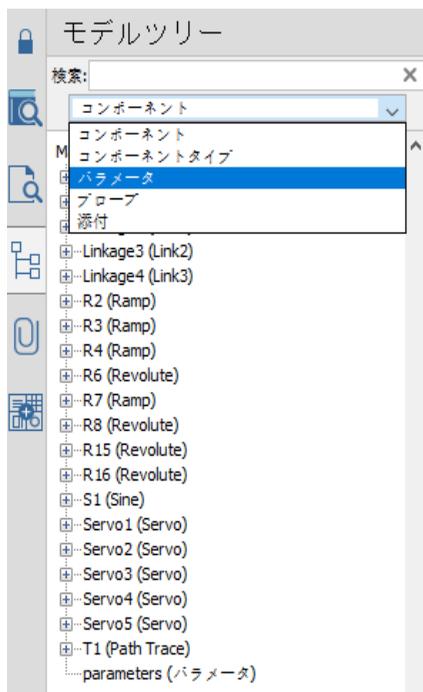


図2.1 モデルツリーの [コンポーネント] ビュー

以下のモデルツリービューから選択できます。

- **[添付]**: このビューは、モデルに添付されているファイルを表示します。添付の例としては、ワークシート、スプレッドシート、およびCAD図面があります。添付をダブルクリックすると、適切なプログラムで添付ファイルを開くことができます。**[検索]** ボックスにキーワードを入力して、一致するドキュメントを探ることができます。
- **[コンポーネントタイプ]**: このビューでは、コンポーネントやサブシステムの型順にモデルツリービューを表示します。コンポーネントやサブシステムのノードは、型に続いて名前で識別されます。**[検索]** ボックスにキーワードを入力して、一致するコンポーネントタイプを探ることができます (コンポーネント名は無視されます)。
- **[コンポーネント]**: このビューでは、コンポーネントやサブシステムの名前順にモデルツリービューを表示します。コンポーネントやサブシステムのノードは、名前に続いて型で識別されます (図2.1 「モデルツリーの [コンポーネント]

ビュー」を参照)。**[検索]**ボックスにキーワードを入力して、一致するコンポーネントおよびサブシステムの名前を探すことができます(コンポーネントタイプは無視されます)。この[コンポーネント]ビューが、デフォルトのモデルツリービューとなります。

- **[パラメータ]**:このビューは、モデルのパラメータ定義を表示します。パラメータ定義は、パラメータテーブル、パラメータブロック、Modelica Record、またはToVariableコンポーネントから取得されます。**[検索]**ボックスにキーワードを入力して、一致する名前のパラメータを探すことができます。パラメータビュー(パラメータへの参照をすべて検索する方法も含め)の詳細は、MapleSimヘルプシステムで、**MapleSim 操作方法 > モデルの構築 > モデルツリーの操作と検索 > パラメータビュー**を参照してください。
- **[プローブ]**:このビューでは、モデルのプローブをすべて表示します。プローブへのフルパスは、プローブの名前の後ろに括弧付きで示されます。**[検索]**ボックスにキーワードを入力して、一致する名前のプローブを探すことができます。

コンポーネントまたはサブシステムと関連付けられているパラメータを表示するには、モデルツリーでコンポーネントノードを展開し、ノードを選択します。**モデルワークスペース**で要素がハイライトされるとともに、**モデルワークスペース**のビューが要素の表示に切り替わり、その要素の設定可能なパラメータが**[プロパティ]**タブ()に表示されます。コンポーネントの選択例については、**図2.2 「モデルツリーを使用したコンポーネントの選択」**を参照してください。

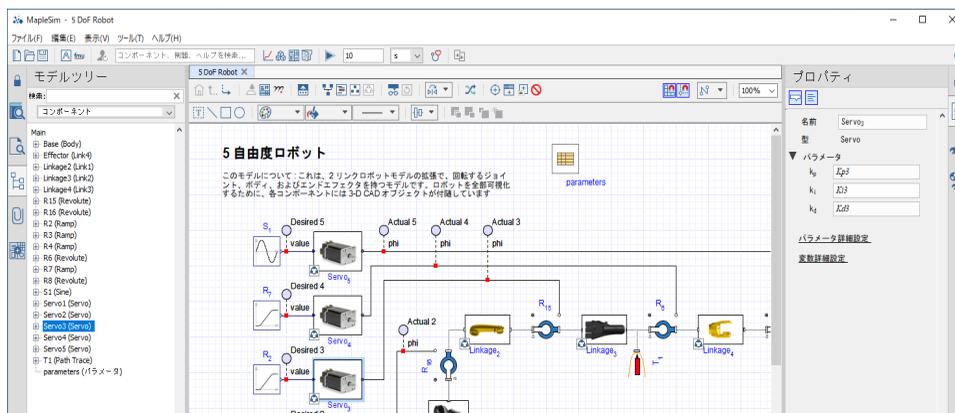


図2.2 モデルツリーを使用したコンポーネントの選択

モデルツリーから複数のコンポーネントを選択すると、コンポーネントに設定可能な共通のパラメータがすべて【プロパティ】タブに表示されます。【プロパティ】タブでパラメータを変更すると、選択したすべてのコンポーネントのパラメータが更新されます。

コンポーネントまたはサブシステムを表示するには、モデルツリーのノードをダブルクリックするか展開し、その子ノードを選択します。モデルワークスペースビューが、適切な階層レベルに展開され、コンポーネントまたはサブシステムが表示されます。

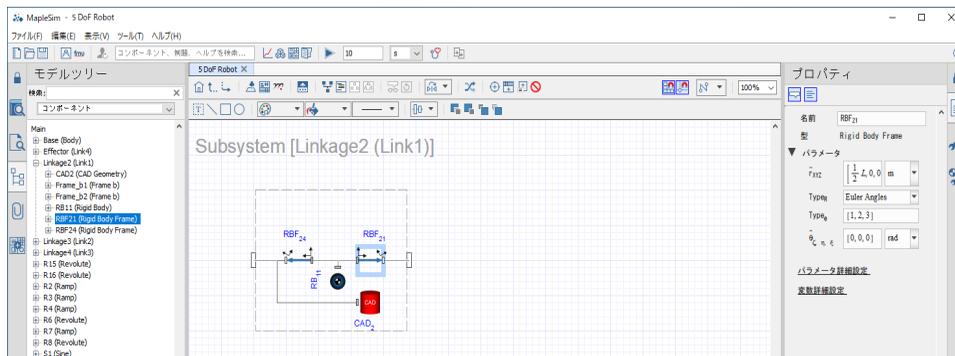


図2.3 サブシステムの表示

[モデルツリー] および複雑なモデル管理の詳細については、MapleSim ヘルプシステムで、**MapleSim 操作方法 > モデルの構築 > モデルツリーの操作と検索**セクションを参照してください。

モデルナビゲーションコントロール

モデルワークスペースツールバーに配置されているモデルナビゲーションコントロールも、**モデルワークスペース**に表示されているダイアグラムのモデリングコンポーネント間やサブシステム間など、階層レベル間を移動して閲覧するために使用できます。



図2.4 モデルナビゲーションコントロール

以下の表ではこれらのコントロールの機能を説明し、関連するキーボードショートカットを示しています。

モデルナビゲーションコントロール

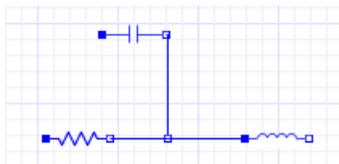
コントロール	キーボードショートカット	説明
	[Home]	モデルの Main レベルに戻る
	[Ctrl] + 上矢印 [Command] + 上矢印 (Mac)	上位階層に移動
	[Ctrl] + 下矢印 [Command] + 下矢印 (Mac)	選択したコンポーネントの内部(または子)を表示

2.3. システムにおけるコンポーネント間の相互作用の定義

作成しているコンポーネント間の相互作用を定義するには、システム内でそれらのコンポーネントを接続します。**モデルワークスペース**では、2つの接続ポートのあいだに結線を引くことができます。



また、結線はポートと別の結線のあいだにも引くことができます。



MapleSim では、互換性のあるドメイン間でのみ結線を引くことができます。デフォルトでは、各ラインタイプはドメインに固有の色で表示されます

ドメイン固有の結線の色

ドメイン	ライン色
メカニカル 1-D 回転	黒
メカニカル 1-D 並進	緑
メカニカルマルチボディ	黒
アナログ電気	青
電気多相	青
磁気	オレンジ
デジタルロジック	紫
ブール値信号	ピンク
因果的信号	濃紺
整数値信号	オレンジ
伝熱	赤
空圧	水色

各ドメインの接続ポートも異なる色と形状で表示されます。接続ポートの詳細は、MapleSim ヘルプシステムの **MapleSim コンポーネントライブラリ > コネクタの概要** を参照してください。

コンポーネントの接続ポートは、スカラーポートまたはベクトルポートのいずれかです。スカラーポートに関連付けられる値は1つだけですが、ベクトルポート

には複数の値(または信号数)を関連付けることができます。信号数の異なるポート間の接続の場合、物理的に接続されている値が**モデルワークスペース**で明らかでない場合があるため、管理が困難です。**接続マネージャ**はこのようなタイプの接続を簡単化し、ユーザが接続されているポートを調査し、必要に応じてポートを変更できるようにします。**接続マネージャ**にアクセスするには、**モデルワークスペース**で結線を選択してから、**[プロパティ]**タブ()を選択します。詳細については、MapleSim ヘルプシステムで、**MapleSim 操作方法 > モデルの構築 > 接続マネージャの使用**を参照してください。

2.4. コンポーネントプロパティの指定

コンポーネントのプロパティは、モデルのコンポーネントにパラメータ値を設定することで指定できます。**モデルワークスペース**でコンポーネントを選択すると、そのコンポーネントの設定可能なパラメータ値が MapleSim ウィンドウ右側の**[プロパティ]**タブ()に表示されます。

注: すべてのコンポーネントに編集可能なパラメータ値があるわけではありません。

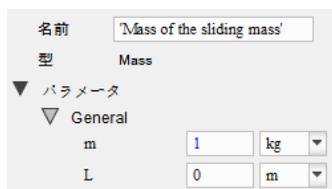
パラメータ値の入力には、上付き文字や下付き文字、ギリシア文字などの数学的文章を追加することができる書式オプションである 2-D Math 表記を使用できます。詳細については、*2-D Math 表記によるテキスト入力*[72ページ]を参照してください。

注: MapleSim コンポーネントライブラリのパラメータのほとんどにはデフォルト値が設定されていますが、一部のパラメータでは、このデフォルト値はシミュレーションする上で無意味な暫定値が入っています。ほかのパラメータ値と区別するため、この暫定値には青いフォントが使用されています。これらの値は、シミュレーションに対し、より適切な値に置き換える必要があります。詳細は、MapleSim ヘルプシステムで、**MapleSim 操作方法 > モデルの構築 > パラメータの指定**を参照してください。

パラメータ単位の指定

パラメータ値の単位を指定するには、パラメータフィールドの横にあるドロップダウンメニューを使用します。たとえば、下の図は **Mass** コンポーネントに設定

可能なパラメータフィールドを示しています。質量には kg 、 lb_m 、 g 、または $slug$ 、そして長さには m 、 cm 、 mm 、 ft 、または in を指定することができます。



モデルをシミュレートする際、MapleSim はすべてのパラメータ単位を自動的に国際単位系 (SI) に変換します。したがって、1つのモデルの中で、パラメータ値に複数の単位系を選択することができます。

信号の単位を変換するには、[信号ブロック] パレットに配置されている [信号変換] メニューの **Conversion Block** コンポーネントを使用します。このコンポーネントは、時間、温度、速度、圧力、容量などの単位を変換することができます。以下の例では、出力信号の単位を変換する **Conversion Block** コンポーネントが並進用の **Position Sensor** と **Feedback** コンポーネントのあいだに接続されています。

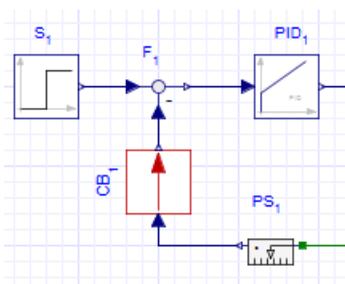


図2.5 Conversion Block を使用した単位の指定

モデルに電気、1-D メカニカル、油圧、または熱力学センサが含まれている場合は、生成する出力信号の単位を選択することもできます。

初期条件の指定

MapleSim のすべてのドメインからのコンポーネントに対して、パラメータ値を設定して初期条件を指定できます。モデルワークスペースで状態変数を含むコン

コンポーネントを選択すると、そのコンポーネントに設定可能なほかのパラメータ値とともに、有効な初期条件フィールドが[パラメータ]タブに表示されます。

たとえば、以下の図は **Mass** コンポーネントで初期位置および初期速度フィールドが設定可能であることを示しています。

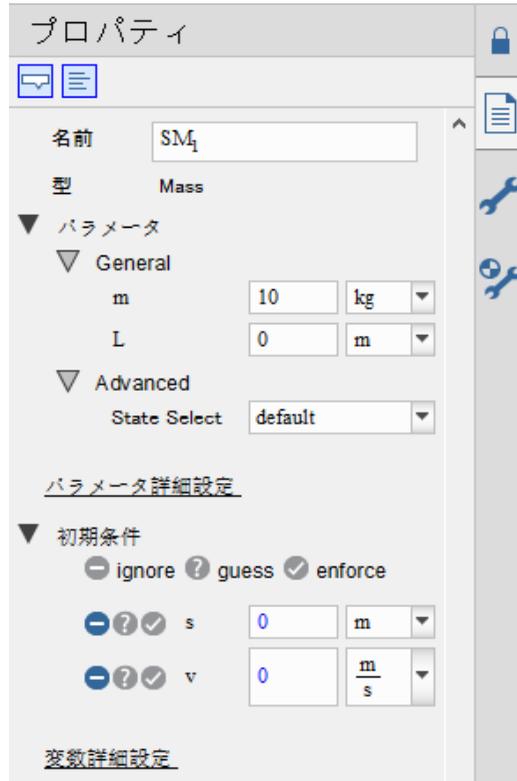


図2.6 初期条件

初期条件の強制方法の指定

特定のコンポーネントに対して指定した初期条件をどのように強制するかを決定できます。指定可能なオプションは、**ignore** (⊖)、**guess** (?)、**enforce** (✓) の3種類です。初期条件フィールド横のボタンを適宜クリックすることで、各初期条件パラメータにつきこれらのオプションを独立に選択することができます。

ignore オプションを選択すると、初期条件フィールドに入力するパラメータ値は無視され、ソルバは初期条件としてデフォルト値(通常ゼロ)を用います。すべての初期条件フィールドは、このオプションにデフォルト設定されています。

guess オプションを選択すると、ソルバは、初期条件フィールドに入力したパラメータ値を最良推定値として使用します。最良推定値は、システムを記述する方程式系が解を持つような初期状態を見つけるための出発点です。ソルバは、まず最良推定値を用いて方程式系の解を求めますが、解が存在しない場合、最良推定値に近い初期条件値を用いて方程式系の解を求めます。

enforce オプションを選択すると、ソルバは、初期条件フィールドに入力されているパラメータ値をシミュレーションの出発点として使用します。このオプションでは、**guess** オプションと同様、初期条件フィールドに入力したパラメータ値に対する方程式系の解を探しますが、解が存在しない場合は**guess** オプションの場合と異なり、ほかの値に置き換えずエラーメッセージを表示します。

これらのオプションの選択についての詳細は、ベストプラクティス:初期条件の強制 [81ページ]を参照してください。

初期条件の強制方法の例は、[ヘルプ] メニューの [例題] > [ユーザガイドの例題] > [第2章] から [Relative Positions] の例題を参照してください。

2.5. サブシステムの作成と管理

サブシステム(または複合コンポーネント)は、いくつかのモデリングコンポーネントを1つのブロックコンポーネントにまとめたものです。シンプルな DC モータのサブシステムを下図に示します。

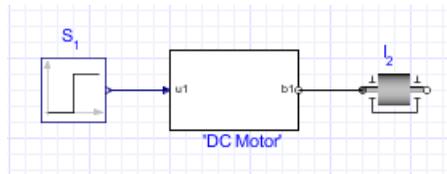


図2.7 サブシステムグループ

サブシステムは、たとえばタイヤや DC モータなど、完全なシステムを構成するコンポーネントをグループ化するのに利用できます。また、**モデルワークスペー**

スでダイアグラムの配置を改善したり、モデルにシステムのコピーを複数追加できるようにしたり、コンポーネントのグループをMapleで解析したり、パラメータや変数をすばやく割り当てるためにも利用できます。ほかにも、サブシステム内に別のサブシステムを作成し、モデルを階層的に整理することもできます。

サブシステムを作成したら、**[プロパティ]** タブ (📄) 内の **[パラメータ詳細設定]** および **[変数詳細設定]** を使用して、サブシステム内の全コンポーネントに対してパラメータや変数を割り当てることができます。



MapleSim でサブシステムを作成するときのベストプラクティス (最良の方法) については、[ベストプラクティス: サブシステムの作成と配置 \[75ページ\]](#)を参照してください。

例: サブシステムの作成

以下の例では、DC モータモデルの電気コンポーネントをグループ化してサブシステムを作成します。

サブシステムを作成するには

1. **[ヘルプ]** メニューの **[例題]** > **[ユーザガイドの例題]** > **[第 2 章]** から、**[Simple DC Motor]** の例題を選択します。
2. マウスをドラッグして、すべての電気コンポーネントを囲む四角形を描きます。

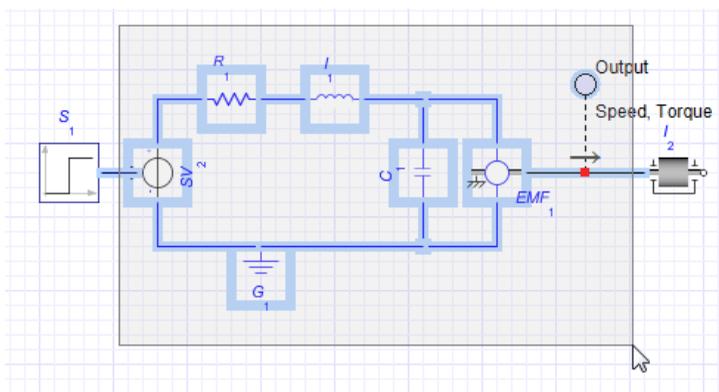
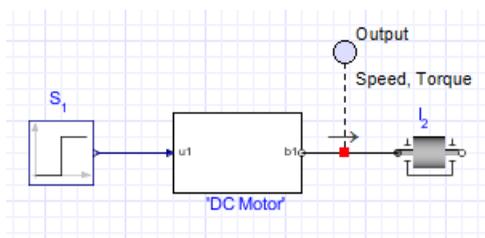


図2.8 サブシステムの作成

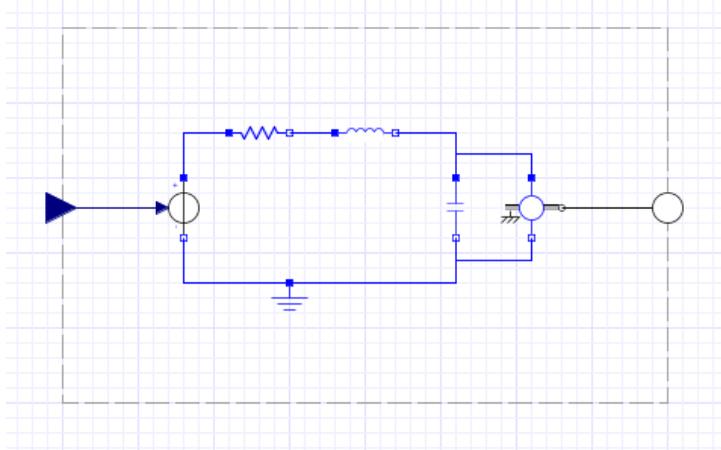
3. [編集] メニューから [サブシステムの作成] を選択 (または、四角に囲まれた領域内を右クリックし、[サブシステムの作成] を選択) します。
4. ダイアログボックスで、「DC Motor」と入力します。
5. [OK] をクリックします。DC モータを表す白いブロックがモデルワークスペースに表示されます。



この例では、モデルに含まれるほかのサブシステムに影響することなく編集および操作が可能な「スタンドアロンサブシステム (*stand-alone subsystem*)」を作成しています。モデルに同じサブシステムのコピーを複数追加し、それらをグループとして編集したい場合は、サブシステム定義を作成することができます。詳細については、モデルにサブシステムのコピーを複数追加 [35ページ]を参照してください。

サブシステムの詳細を表示する

サブシステムに含まれているコンポーネントを表示するには、**モデルワークスペース**でサブシステムをダブルクリックします。サブシステムの詳細ビューが表示されます。



この表示で、点線はサブシステムの境界を示しています。境界線内の結線およびコンポーネントの編集、境界線外へのコンポーネントの追加と接続、ほかのコンポーネントへ接続するためのサブシステムポートの追加などの操作が可能です。境界のサイズを変更するには、点線をクリックし、ボックスの周りに表示されるサイズ変更ハンドルのいずれかをドラッグします。

モデルのトップレベルまたは別のサブシステムを表示するには、**モデルワークスペースツールバー**にあるモデルナビゲーションコントロールを使用します。モデルナビゲーションコントロールの詳細については、[ベストプラクティス: サブシステムの作成と配置 \[75ページ\]](#)を参照してください。



モデルにサブシステムのコピーを複数追加

モデルに同一構成を持つサブシステムのコピーを複数追加する場合は、「サブシステム定義」を作成することができます。サブシステム定義とは、一連のサブシステムで共有される属性と構成を定義する基本サブシステムを指します。

たとえば、同じコンポーネントで構成され、同じ抵抗値を持った3つのDC Motor サブシステムをモデルに追加する場合、以下のタスクを実行することで実現できます。

1. **モデルワークスペース**で、希望する構成の DC Motor サブシステムを作成します。
2. 上記サブシステム構成を元にサブシステム定義を作成し、それを **[ローカルコンポーネント]** タブの **[コンポーネント]** パレットに追加します。
3. 最後に、サブシステム定義をソースとして、DC Motor サブシステムのコピーをモデルに追加します。

DC Motor サブシステムのコピーをモデルに追加するときは、DC Motor サブシステムのサブシステム定義アイコンを **[ローカルコンポーネント]** タブ (🔍) の **[コンポーネント]** パレットから **モデルワークスペース** にドラッグします。 **モデルワークスペース** に追加されたコピーは、 **[ローカルコンポーネント]** タブ内のサブシステム定義と構成を共有するため、「共有サブシステム (*shared subsystem*)」と呼ばれます。

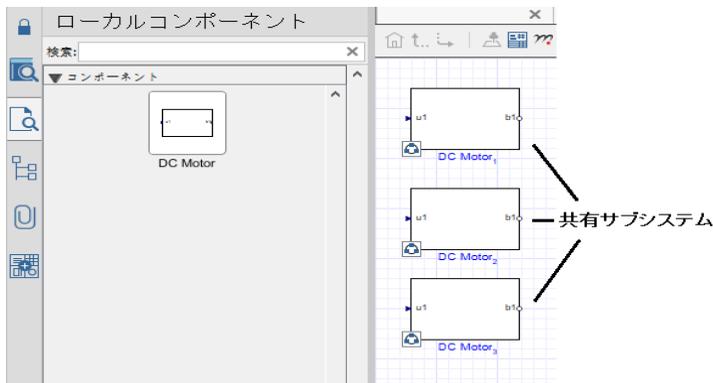


図2.9 複数のサブシステムの定義

同じサブシステム定義からコピーされた共有サブシステムは「リンク」しています。つまり、共有サブシステムのどれかに変更を加えると、その変更は同じサブシステム定義から作成されたほかの共有サブシステムすべてに反映されます。変更は、 **[ローカルコンポーネント]** タブのサブシステム定義の項目にも反映されます。共有サブシステムはモデルワークスペースにアイコン (🔍) で表示されます。

上記例で、たとえば **DC Motor2** 共有サブシステムの **Resistor** コンポーネントの抵抗パラメータを 24Ω から 10Ω に変更すると、**DC Motor1** および **DC Motor3** 共有サブシステムはもとより、**[ローカルコンポーネント]** タブの **DC Motor** サブシステム定義の **Resistor** コンポーネントの抵抗値も 10Ω に変わります。

詳細については、**サブシステム定義と共有サブシステムの編集 [39ページ]**を参照してください。

例：モデルにサブシステム定義と共有サブシステムを追加する

次の例では、**DC Motor** サブシステム定義を作成し、モデルに共有サブシステムを複数追加します。

[ローカルコンポーネント] タブにサブシステム定義を追加する

サブシステム定義を追加するには

1. **モデルワークスペース**で、例：**サブシステムの作成 [33ページ]**で作成した **DC Motor** スタンドアロンサブシステムを右クリック (Mac では **[Control]** キーを押しながらクリック) します。
2. コンテキストメニューから、**[共有サブシステムに変換]** を選択します。
3. サブシステム定義の名前に「**DC Motor**」と入力し、**[OK]** をクリックします。
4. **モデルワークスペース**左側の **[ローカルコンポーネント]** タブ (🔍) で、**[コンポーネント]** パレットを展開します。



図2.10 サブシステムの定義

[コンポーネント] パレットにサブシステム定義が追加され、**モデルワークスペース**のサブシステムは、**DC Motor₁**という名前の共有サブシステムに変わります。

この共有サブシステムは、**DC Motor**というサブシステム定義にリンクされます。

5. モデルを「**DCMotorSubsystem.msim**」という名前で保存します。このモデルは例: 同一サブシステム定義にリンクしている共有サブシステムを編集する [39ページ]で作成します。

このサブシステム定義を使用すると、MapleSim モデルに DC Motor 共有サブシステムを複数追加することができます。

ヒント: サブシステム定義を別のモデルで使用する場合は、サブシステム定義をカスタムライブラリに追加します。詳細については、**カスタムライブラリの作成と管理** [66ページ]を参照してください。

モデルに DC Motor 共有サブシステムを複数追加する

モデルに **DC Motor** 共有サブシステムを複数追加するには、**DC Motor** サブシステム定義アイコンを [ローカルコンポーネント] タブからドラッグし、**モデルワークスペース**に配置します。

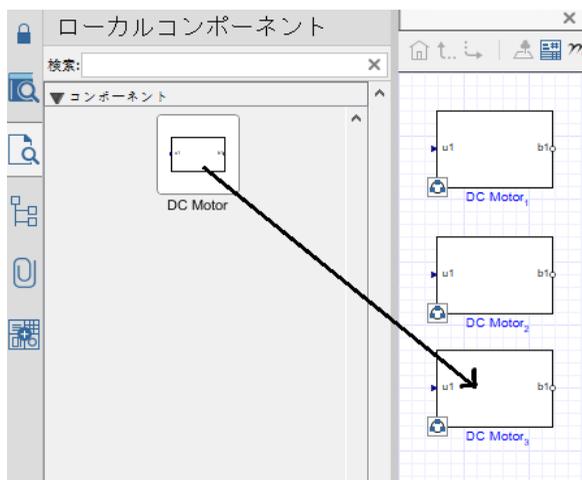


図2.11 モデルに複数のサブシステムを追加する

モデルに新しいスタンドアロンサブシステムまたは共有サブシステムを追加した場合、**モデルワークスペース**に表示されるそのサブシステムの名前には固有の下付き数字が付与されます。上記図で示されているように、各 DC Motor 共有サブシステムの名前には下付き数字が付与されています。ユーザはこの番号で、モデル内に配置されているサブシステムの複数のコピーを識別することができます。

サブシステム定義と共有サブシステムの編集

モデルワークスペースで共有サブシステムを編集すると、変更はその共有サブシステムがリンクしているサブシステム定義はもとより、同じサブシステム定義からコピーされたほかの共有サブシステムにも反映されます。

例：同一サブシステム定義にリンクしている共有サブシステムを編集する

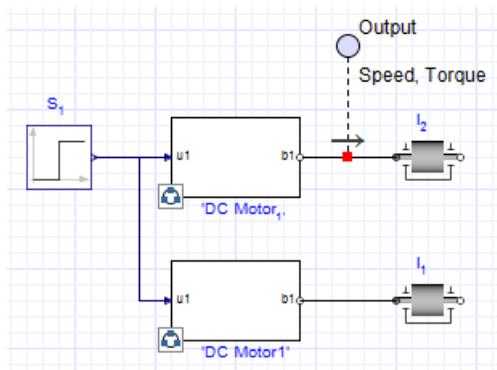
この例では、2つの **DC Motor** 共有サブシステムを含むモデルを作成し、その共有サブシステムの抵抗値とアイコンを編集します。これらの共有サブシステムは、例：モデルにサブシステム定義と共有サブシステムを追加する [37ページ]で作成された **DC Motor** 共有サブシステム定義にリンクされています。1つの **DC Motor** 共有サブシステムの1つのコンポーネント値と1つのアイコンを変更すると、モデル内のほかの **DC Motor** 共有サブシステムと、これ以降に追加される新しい **DC Motor** 共有サブシステムにもその変更が追加されることを確認します。

注：この例を実行する前に、例：モデルにサブシステム定義と共有サブシステムを追加する [37ページ]を実行済みで、その結果を保存している必要があります。

共有サブシステムを使用するには

1. 例：モデルにサブシステム定義と共有サブシステムを追加する [37ページ]で作成した **DCMotorSubsystem.msim** ファイルを MapleSim で開きます。
2. **[ローカルコンポーネント]** タブ () で **[コンポーネント]** パレットを展開し、2番目の **DC Motor** 共有サブシステムをワークスペースにドラッグし、既存の **DC Motor** 共有サブシステムの下に置きます。
3. **[ライブラリコンポーネント]** タブ () で **[1-D メカニカル]** > **[回転]** > **[共通]** メニューを展開し、2番目の **Inertia** コンポーネントをワークスペースにドラッグし、既存の **Inertia** コンポーネントの下に置きます。

4. モデル内で、新しく追加したコンポーネントと既存のコンポーネントのあいだを次のように接続します。



5. モデルワークスペースで、**DCMotor₁** 共有サブシステムをダブルクリックします。共有サブシステムが詳細表示されます。

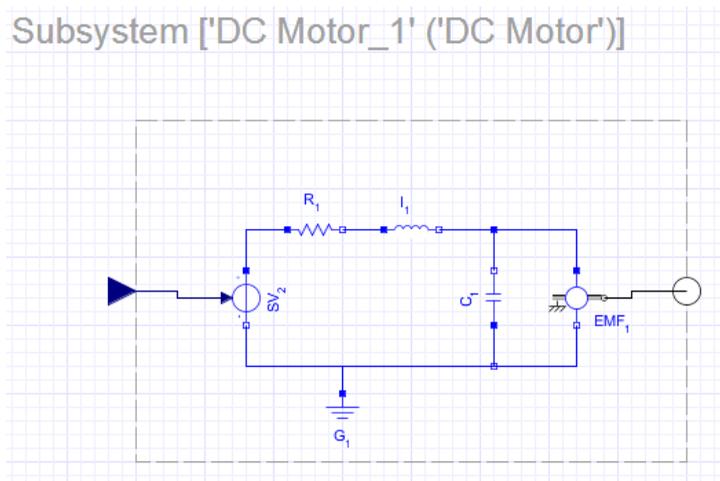
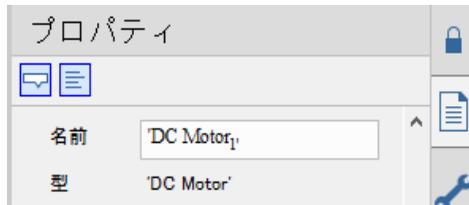


図2.12 DC Motor サブシステム

共有サブシステム名 (**DCMotor₁**) と、それに続くサブシステム定義名 (**DC Motor**) から成る見出しが **モデルワークスペース** の上部に表示されます。すべての共有サブシステムの詳細表示には、モデルにおける共有サブシステムの複数のコピーを識別できるように、この見出しが表示されます。また、共有サブシステムを選択

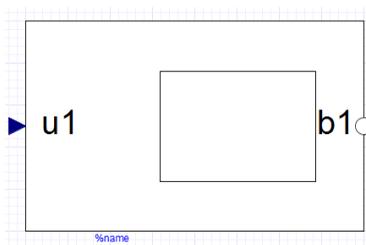
すると、そのサブシステム定義の名前が[プロパティ]タブ (📄) の[型]フィールドに表示されます。



6. **Resistor** コンポーネント (R_1) を選択し、[プロパティ] タブで [パラメータ] をクリックします。抵抗値を 50Ω に変更します。

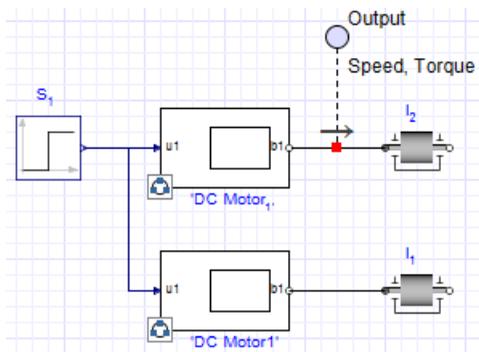


7. モデルワークスペースツールバーでアイコン (👤) をクリックします。
8. モデルワークスペースツールバーにある長方形ツール (📏) を使用して、マウスポインタでクリックし、四角形を描くようにドラッグします。



9. モデルワークスペースツールバーでダイアグラム (📐) をクリックします。

- 10 モデルワークスペースツールバーの **Main** (🏠) をクリックし、モデルのトップレベルを表示します。**DC Motor** 共有サブシステムの表示は両方ともユーザーによって描かれた四角形に変わっています。



- 11 MapleSim ウィンドウ左側の [ローカルコンポーネント] タブで、[コンポーネント]パレットを展開します。下図のように、変更はこのパレットの **DC Motor** 項目にも反映されています。



モデルワークスペースに配置されている **DC Motor** サブシステムをダブルクリックし、それらの **Resistor** コンポーネントを選択すると、両方の共有サブシステムの抵抗値が 50Ω になっていることが確認できます。

- 12 [ローカルコンポーネント] タブから **DC Motor** サブシステムの新たなコピーをドラッグし、モデルワークスペース内の任意の場所に配置します。新しいコピーがユーザーが描いた四角形で表示され、その抵抗値も 50Ω であることを確認したら、それをワークスペースから削除します。

- 13 モデルを「**DCMotorSharedSubsystem.msim**」という名前で保存します。
このモデルは例: 共有サブシステムとそのサブシステム定義とのあいだのリンクの解除 [43ページ]で作成します。

例: 共有サブシステムとそのサブシステム定義とのあいだのリンクの解除

リンクしている共有サブシステムが複数含まれているモデルで特定のコピーのみを編集するには、その共有サブシステムとサブシステム定義のあいだのリンクを解除することで、**モデルワークスペース**に配置されているほかのサブシステムに影響を与えることなくそのサブシステムを編集することができます。

注: この例を実行する前に、例: 同一サブシステム定義にリンクしている共有サブシステムを編集する [39ページ]を実行済みでその結果を保存し、かつその例題の結果を保存している必要があります。

共有サブシステムのリンクを解除するには

1. 例: 同一サブシステム定義にリンクしている共有サブシステムを編集する [39ページ]で作成した **DCMotorSharedSubsystem.msim** モデルを開きます。
2. **モデルワークスペース**で、**DC Motor₁** 共有サブシステムを右クリック (Macでは **[Control]** キーを押しながらクリック) します。
3. **[スタンドアロンサブシステムに変換]** を選択します。**DC Motor₁** サブシステムと **[ローカルコンポーネント]** タブの **DC Motor** サブシステム定義のあいだのリンクが解除され、**DC Motor** のコピーとなります。
4. **DC Motor₁** 共有サブシステムをダブルクリックします。
5. **アイコン** () をクリックします。
6. **長方形ツール** () を使用して、マウスポインタでクリックし、**モデルワークスペース**で四角形を描くようにドラッグします。
7. **ダイアグラム** () をクリックしてから **Main** () をクリックし、モデルのトップレベルを表示します。変更は、**モデルワークスペース**に配置されている **DC Motor₁** 共有サブシステムと、**[ローカルコンポーネント]** タブの **DC Motor** サブシステム定義に反映されます。**DC Motor** サブシステム定義とのリンクが解

除されたため、**DC Motorのコピー** サブシステムには変更が反映されないことに注意してください。

ヒント: 共有サブシステムをスタンドアロンサブシステムに変換する場合、既存の共有サブシステムやサブシステム定義と区別するためにもスタンドアロンサブシステムには意味のある名前を付けることがベストプラクティスとして推奨されます。

スタンドアロンサブシステムの操作

スタンドアロンサブシステムとは、サブシステム定義とリンクしていないサブシステムを指します。スタンドアロンサブシステムを作成する方法は2つあります。例: 共有サブシステムとそのサブシステム定義とのあいだのリンクの解除 [43ページ]で説明されている、新規サブシステムとして作成する方法と、例: サブシステムの作成 [33ページ]で説明されている、共有サブシステムをスタンドアロンサブシステムに変換する方法です。スタンドアロンサブシステムは、**モデルワークスペース**に配置されているほかのサブシステムに影響を与えることなく編集することができます。

特定のサブシステムがスタンドアロンであるかどうかを確認するには、**モデルワークスペース**でサブシステムを選択し、**[プロパティ]**タブ(📄)を見ます。サブシステムがスタンドアロンである場合、**[型]** フィールドに **[スタンドアロンサブシステム]** と表示されます。



スタンドアロンシステムは共有サブシステムアイコン(🔗)を**モデルワークスペース**に表示しません。また、スタンドアロンサブシステムをダブルクリックして詳細表示を開いても、**モデルワークスペース**にサブシステムの見出しは表示されません。

スタンドアロンサブシステムを**モデルワークスペース**でコピーして貼り付けるときに、オプションで共有サブシステムに変換し、新しいサブシステム定義を作成することができます。詳細については、[例: スタンドアロンサブシステムのコピーと貼り付け \[46ページ\]](#)を参照してください。

例: デバッグコンソールに表示される警告メッセージを処理する

共有サブシステムをスタンドアロンサブシステムに変換すると、**モデルワークスペース**内でサブシステムがハイライトされ、サブシステム定義とのリンクが解除されたことを知らせる警告メッセージが表示されます。

注: この例は[例: 共有サブシステムとそのサブシステム定義とのあいだのリンクの解除 \[43ページ\]](#)の延長です。

警告メッセージを処理するには

1. MapleSim ウィンドウ下部の **診断情報** () をクリックし、デバッグコンソールを表示します。コンソールには、次のメッセージが表示されます。

Description	Location
 The stand-alone subsystem 'copy of DC Motor' is identical to the shared subsystem 'DC Motor'.	Main

2. **DC Motor のコピー**サブシステムをスタンドアロンサブシステムとして扱うためには、警告メッセージを右クリック (Mac では **[Control]** キーを押しながらクリック) し、**['DC Motor のコピー'] に対する重複警告を無視する** を選択し、デバッグコンソールで警告メッセージが非表示となるようにします。

ヒント: デバッグコンソールで非表示にした警告メッセージを表示したい場合は、コンソールの上の**無視された警告をリセット** () をクリックします。前に非表示にしたすべての警告メッセージがデバッグコンソールに再び表示されます。

また、**DC Motor のコピー**スタンドアロンサブシステムを再度 **DC Motor** サブシステム定義とリンクさせたい場合は、警告メッセージを右クリック (Mac では **[Control]** キーを押しながらクリック) し、**['共有サブシステム 'DC Motor' を使用するために 'DC Motor のコピー' を更新します。]** を選択します。

例: スタンドアロンサブシステムのコピーと貼り付け

注: この例は例: 共有サブシステムとそのサブシステム定義とのあいだのリンクの解除 [43ページ]の延長です。

スタンドアロンサブシステムをコピーおよび貼り付けるには

1. モデルワークスペースで、**DC Motorのコピー** スタンドアロンサブシステムをコピーし、貼り付けます。ダイアログボックスが表示されます (図2.13 「サブシステムダイアログのコピー」を参照してください)。
2. [上記のスタンドアロンサブシステムを共有サブシステムに変換 (推奨)] を選択します。SharedSubsystem_1 という名前の新しいサブシステム定義が [ローカルコンポーネント] タブの [コンポーネント] パレットに追加されます。

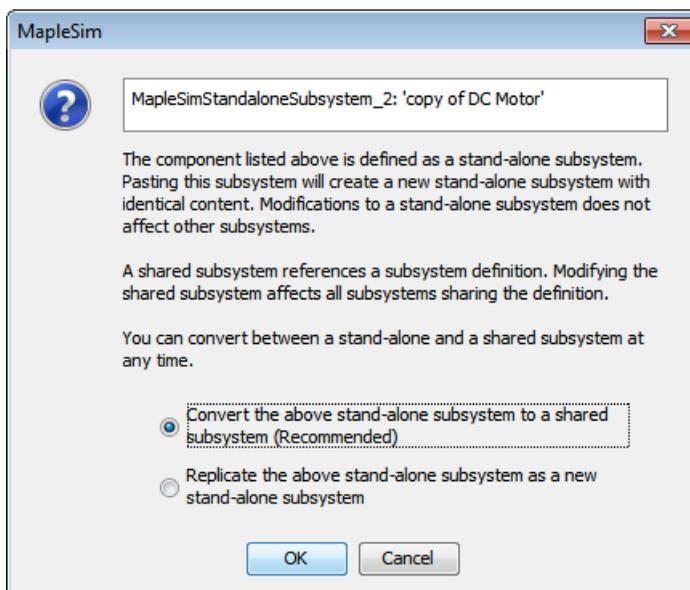


図2.13 サブシステムダイアログのコピー

モデルワークスペースで、**DC Motorのコピー** スタンドアロンサブシステムは、**DC Motorのコピー** という名前の共有サブシステムになり、この共有サブシステムのコピーで **DC Motorのコピー₁** という名前の共有サブシステムがモデルワークスペースに追加されます。共有サブシステムの**DC Motorのコピー**と**DC Motor**

のコピー₁は、両方とも新しい **SharedSubsystem_1** サブシステム定義にリンクしていることとなります。そのため、**DC Motor** のコピー または **DC Motor** のコピー₁ をモデルワークスペースで編集しても、これらの変更は元の **DC Motor** サブシステム定義にリンクしているサブシステムには反映されません。

注: または、[上記のスタンドアロンサブシステムを複製し、新しいスタンドアロンサブシステムとして作成] を選択すると、モデルワークスペースに存在するほかのサブシステムに影響することなく編集できる、別のスタンドアロンサブシステムが追加されます。

2.6. グローバルパラメータとサブシステムパラメータ

MapleSimでは、グローバルパラメータとサブシステムパラメータの値を定義し、その値を [パラメータを追加または変更] エディタ、パラメータブロック、パラメータセット、[プロパティ] タブの [パラメータ詳細設定]、[変数詳細設定] を使用してコンポーネントに代入することができます。

グローバルパラメータ

共通のパラメータ値を共有するコンポーネントがモデルに複数含まれている場合、グローバルパラメータを作成することができます。グローバルパラメータを使用すると、共通のパラメータ値を1か所で定義し、その共通の値をモデルに配置されている複数のコンポーネントに代入することができます。

以下の例は、グローバルパラメータを定義して代入する方法を示すものです。より詳細な例については、本ガイドの第6章のチュートリアル1: ギアボックス付き **DC Motor** のモデリング [181ページ]を参照してください。

例: グローバルパラメータの定義と代入

モデルに共通の抵抗値を持つ **Resistor** コンポーネントが複数含まれている場合、パラメータエディタ画面で抵抗値のグローバルパラメータを定義することができます。

グローバルパラメータを定義および代入するには

1. [ライブラリコンポーネント] タブ (🔍) で [電気] パレット > [アナログ] メニュー > [受動素子] メニューを開き、続いて [抵抗] メニューを開きます。
2. パレットから、**Resistor** コンポーネントのコピーを3つモデルワークスペースにドラッグします。



3. モデルワークスペースツールバーで、**パラメータ** (📄) をクリックするか、ワークスペースをクリックして [プロパティ] タブ (📄) から [パラメータを追加または変更] をクリックします。[Main サブシステム デフォルト設定] 画面が表示されます。この画面から、グローバルパラメータを定義し、モデル中の **Resistor** コンポーネントに代入します。

Main サブシステム デフォルト設定

名前	型	デフォルト値	デフォルト単位	説明
<input type="text"/>				

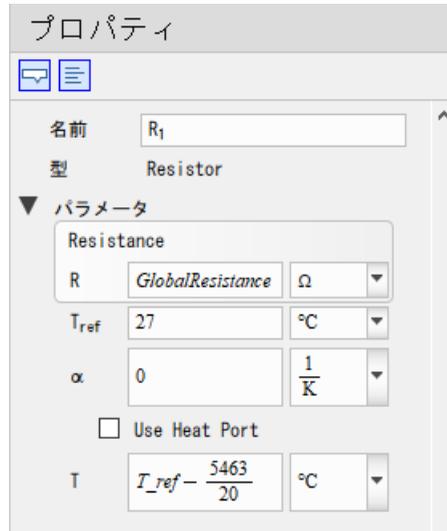
4. [Main サブシステム デフォルト設定] テーブルで、[名前] 列の1つ目のフィールドをクリックします。
5. グローバルパラメータ名に「**GlobalResistance**」と入力し、[Enter] を押します。
6. [型] で [Resistance [[Ω]]] を選択し、デフォルト値として「2」を指定します。
7. 説明に「**Global resistance variable**」と入力し、[Enter] を押します。

Main サブシステム デフォルト設定

名前	型	デフォルト値	デフォルト単位	説明
GlobalResistance	Resistance [[Ω]]	2	Ω	Global resistance value
<input type="text"/>				

これで抵抗値のためのグローバルパラメータが定義されました。次に、共通の **GlobalResistance** パラメータ値をモデルワークスペースに配置されている各 **Resistor** コンポーネントに代入します。

8. 図 () をクリックして、**R1コンポーネント**を選択します。抵抗値として **GlobalResistance**を入力します。 **R2コンポーネント**についても繰り返します。



これで **GlobalResistance** パラメータの抵抗値 ([Main サブシステム デフォルト設定] テーブルで「2」と定義) が **R₁** および **R₂** コンポーネントの抵抗値に代入されました。

R₁ および **R₂** コンポーネントはこれ以降、 [Main サブシステム デフォルト設定] テーブルにて行われる **GlobalResistance** パラメータ値への変更を受け継ぐようになります。たとえば、 [Main サブシステム デフォルト設定] テーブルで **GlobalResistance** パラメータのデフォルト値が「5」に変更されると、 **R₁** および **R₂** コンポーネントの抵抗値も 5 になります。ただしパラメータ値として **GlobalResistance** が代入されていない **R₃** コンポーネントには、いかなる **GlobalResistance** パラメータ値の変更も適用されません。

サブシステムパラメータ

サブシステム内の複数のコンポーネントで共有する共通のパラメータ値を作成するには、サブシステムパラメータを作成します。グローバルパラメータと同様、

サブシステムパラメータはパラメータエディタ画面で定義し、コンポーネントに代入する共通の値です。

サブシステムパラメータを代入する方法は2つあります。ひとつは**パラメータ** (🔗) をクリックする方法で、もうひとつは**[プロパティ]** (📄) タブの**[パラメータ詳細設定]** ツールを使用する方法です。パラメータは、それが定義されているサブシステム内のコンポーネントにしか割り当てられません。**モデルワークスペース**のサブシステムをクリックして**パラメータ** (🔗) または**[パラメータ詳細設定]** をクリックし、パラメータエディタ画面からパラメータを定義すると、定義したパラメータは、選択したサブシステムおよびネスト化されたあらゆるサブシステムのコンポーネントに代入されます。

例については、本ガイドの第6章のチュートリアル3: *非線形ダンパのモデリング* [194ページ]を参照してください。

注: サブシステムでパラメータを作成し、その値をトップレベルに配置されているコンポーネントに代入した場合、トップレベルに配置されているコンポーネントはパラメータの値を受け継ぎません。

例: 共有サブシステムへのサブシステムパラメータの代入

モデルの共有サブシステムにサブシステムパラメータを代入すると、そのデフォルトのサブシステムパラメータはリンクしているほかの共有サブシステムにも代入されます。ただし、デフォルトのサブシステムパラメータを代入した後も、各共有サブシステムのサブシステムパラメータ値は、モデルに含まれるほかのパラメータ値に影響を与えることなく、個々に編集することができます。

共有サブシステムにサブシステムパラメータを代入するには以下の手順に従います。

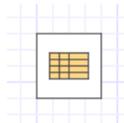
1. **[ヘルプ]** メニューの **[例題]** > **[ドメイン別例題]** > **[マルチボディ]** から **[Double Pendulum]** モデルを選択します。このモデルには、**L** という名前のサブシステム定義とリンクしている2つの共有システム、**L₁** と **L₂** が含まれています。
2. **L₁** 共有サブシステムをダブルクリックします。
3. **パラメータ** (🔗) をクリックします。

4. [Lサブシステムデフォルト設定]テーブルの下部に配置されている、空のフィールドをクリックします。
5. パラメータ名に「c」と入力し、デフォルトの値を「1」のままにして [Enter] キーを押します。
6. **ダイアグラム** (📐) をクリックします。新しいサブシステムパラメータ c が、 L_1 共有サブシステムの [プロパティ] タブ (📄) に表示されます。
7. **Main** (🏠) をクリックし、 L_2 サブシステムを選択して、[プロパティ] タブを確認します。新しいサブシステムパラメータは、 L_2 共有システムに対しても表示されています。
8. [プロパティ] タブで、c の値を「50」に変更します。
9. **モデルワークスペース**で、 L_1 共有サブシステムをクリックし、[プロパティ] タブを確認します。cパラメータの値が変わっていないことに注意してください。

パラメータブロックの作成

上記方法によってサブシステムパラメータを定義する代わりに、サブシステムパラメータ式を定義できるパラメータブロックを作成し、それをモデルのコンポーネントに代入することもできます。パラメータブロックを使用すると、**モデルワークスペース**のトップレベルの複数のモデルに対し、パラメータ値を適用できます。

下の図は、**モデルワークスペース**に追加されたパラメータブロックを表しています。



このブロックをダブルクリックすると、パラメータエディタ画面が表示されます。この画面から、ブロックのパラメータ値が定義できます。

パラメータ デフォルト設定

名前	型	デフォルト値	デフォルト単位	説明
<input type="text"/>				

パラメータ値の定義が終わると、それらの値をモデル内に配置されているコンポーネントのパラメータに代入することができます。

パラメータ値をほかのモデルで使用するには、パラメータブロックをカスタムライブラリに追加します。カスタムライブラリの詳細は、*カスタムライブラリの作成と管理* [66ページ]を参照してください。

注:

- パラメータブロックは、パラメータ値を代入したいコンポーネントが配置されているサブシステムに作成する必要があります。
- モデルの同一階層レベルにおいて、同一名のパラメータを含むパラメータブロックを作成することはできません。たとえば、同一サブシステムにおいて、それぞれが **mass** という名前のパラメータを含む2つのパラメータブロックを作成することはできません。

例: パラメータブロックの作成と使用

この例では、モデル内の複数のコンポーネントで共有できるパラメータ式を作成します。パラメータブロックを作成すると、1か所でパラメータ値を変更するだけで複数のシミュレーション結果を比較することができます。

パラメータブロックを作成して使用するには

1. [ヘルプ] メニューの [例題] > [ドメイン別例題] > [1-D メカニカル] から [PreLoad] の例題を選択します。
2. [シミュレーションの設定] タブ () で、[t_d] (シミュレーション時間) に 「0.012」 秒を入力します。
3. ワークスペースで **SM₁** という Mass コンポーネントをクリックしてから [プロパティ] タブ () をクリックします。これからはパラメータブロックを使用して、 m 、 L 、 s_0 、 v_0 パラメータの値を設定します。
4. モデルワークスペースツールバーからパラメータブロックを追加 () をクリックし、モデルワークスペースの空白部分をクリックします。
5. [プロパティ] タブをクリックし、パラメータブロックの名前に 「SlidingMassParams」と入力します。

6. モデルワークスペースで、パラメータブロック **SlidingMassParams** をダブルクリックします。パラメータエディタ画面が表示されます。

パラメータ デフォルト設定

名前	型	デフォルト値	デフォルト単位	説明

7. テーブルの最初のフィールドをクリックし、「**MASS**」という名前の記号パラメータを定義します。
8. **[Enter]** を押します。この行の残りのフィールドが有効になります。
9. **[型]** ドロップダウンメニューで、**Mass [[kg]]** を選択します。
10. デフォルト値 **「5」** を入力します。
11. **[デフォルト単位]** ドロップダウンメニューで、**[kg]** を選択します。
12. **[説明]** フィールドに **「Mass of the sliding mass」** と入力します。
13. 同様に、以下のパラメータと値を **Parameters サブシステムの標準設定テーブル** に定義します。

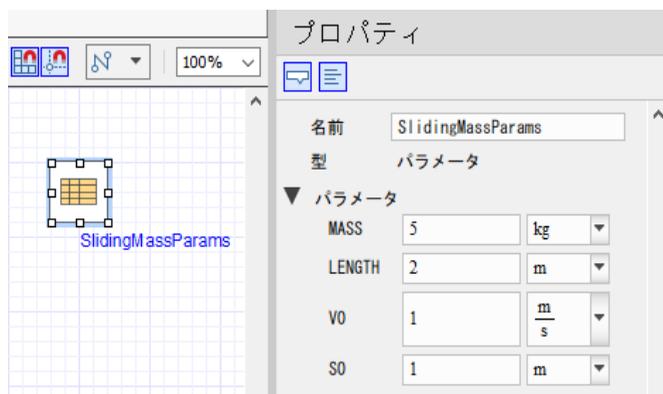
名前	型	デフォルト値	デフォルト単位	説明
LENGTH	Length [[m]]	2	m	Length of the sliding mass
V0	Velocity [[$\frac{m}{s}$]]	1	$\frac{m}{s}$	Initial velocity of the sliding mass
S0	Position [[m]]	1	m	Initial position of the sliding mass

下の図は、値がすべて定義された状態のパラメータエディタ画面を表しています。

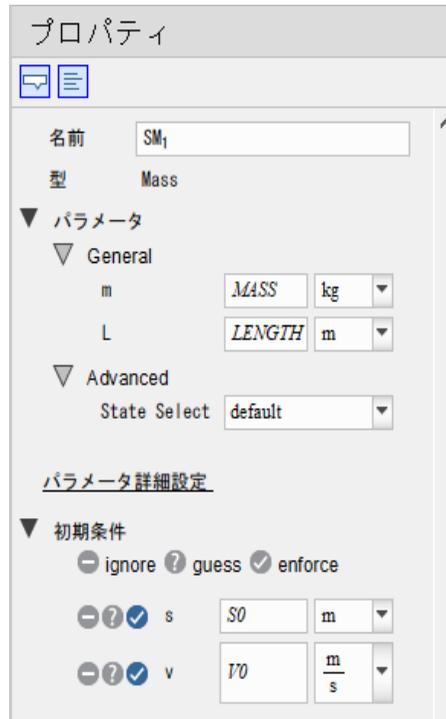
パラメータ デフォルト設定

名前	型	デフォルト値	デフォルト単位	説明
MASS	Mass [kg]	5	kg	Mass of the sliding mass
LENGTH	Length [m]	2	m	Length of the sliding mass
V0	Velocity [$\frac{m}{s}$]	1	$\frac{m}{s}$	Initial velocity of the sliding mass
S0	Position [m]	1	m	Initial position of the sliding mass

- 14 **ダイアグラム** (📐) をクリックします。**モデルワークスペース**でパラメータブロックを選択すると、定義したパラメータが、MapleSimウィンドウ右側の[**プロパティ**] タブに表示されます。



- 15 **モデルワークスペース**で、ダイアグラム内の **SM₁ Mass** コンポーネントを選択します。
- 16 [**プロパティ**] タブで以下の値を代入し、[**Enter**] キーを押します。



これで、この **Mass** コンポーネントのパラメータはパラメータブロックで定義された数値を受け継ぐようになります。

17. 同様に、モデル内の **SM₂** および **SM₃** という Mass コンポーネントにも同じ値を代入します。

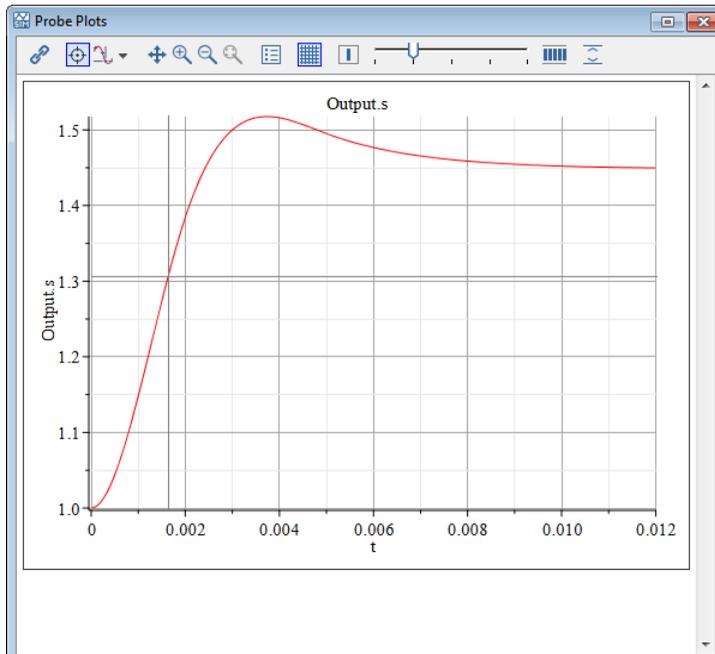
18. **モデルワークスペース**で、**Input** というラベルのプローブを削除します。

19. **Output** というラベルのプローブを選択します。

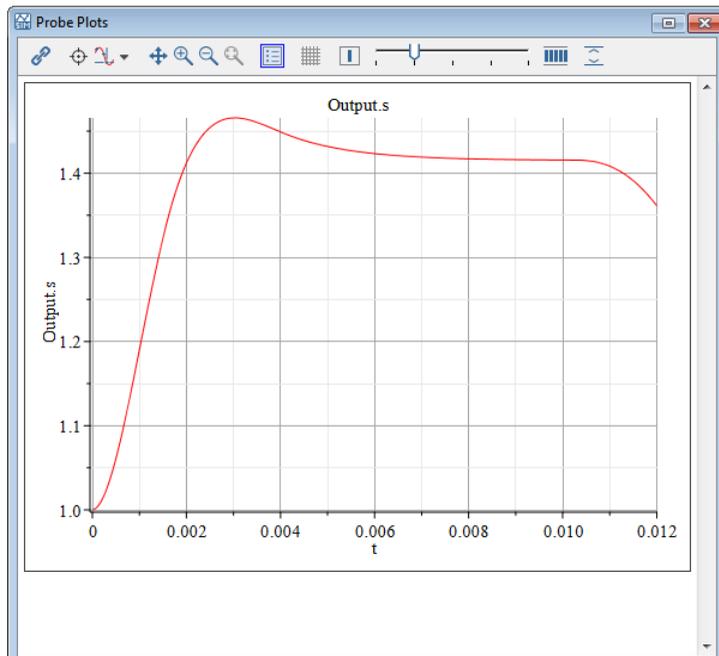
20. [**プロパティ**] タブで、[**Velocity**] の横にあるチェックボックスのチェックを外します。

21. モデルをシミュレートするには、**メインツールバー**の**シミュレーションの実行** (▶) をクリックします。

22. **シミュレーション結果を表示** (📄) をクリックします。[**解析ウィンドウ**] に以下のグラフが表示されます。



- 23 モデルワークスペースで、パラメータブロックをクリックします。
- 24 [プロパティ] タブで、質量 (MASS) を「3.5」、初期速度を「5」に変更します。[Enter] を押します。これらの変更は、記号パラメータ値を代入したすべての Mass コンポーネントに適用されます。
- 25 モデルのシミュレーションを再度実行し、[解析ウィンドウ] を最前面に移動します。新しいシミュレーショングラフが表示され、最初のグラフと比較できます。



パラメータセットの作成

モデルのための作成するパラメータは、再利用可能なパラメータセットとして保存できます。パラメータセットを利用すると、ワークスペースに表示された1つのモデルに対し、パラメータ値のセットを保存と再利用したり、異なるパラメータ値のセットどうしを比較することができます。いつでも多様なシミュレーションを簡単に適用、実行でき、その都度、モデルに対して新しい値を保存できます。パラメータセットは、**モデルワークスペース**の全パラメータのスナップショットを提供します。

モデルのパラメータセットは、[添付ファイル] タブ (📎) の [パラメータセット] 下に表示されます (下図参照)。



パラメータセットを右クリック (Macintosh では **[Control]** キーを押しながらクリック) することで、同じモデルで異なるパラメータセットを保存、再利用、比較することができます。詳細については、**MapleSim ヘルプシステムの MapleSim 操作方法 > モデルの構築 > パラメータセットの利用 > パラメータセットの保存と適用セクション**を参照してください。

パラメータと変数の詳細設定の利用

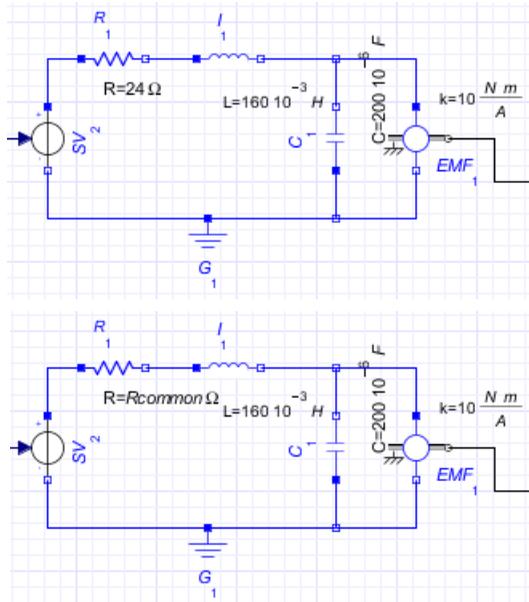
モデルのトップレベルの **[Main サブシステム デフォルト設定]** ウィンドウで、パラメータを追加してそのデフォルト値を設定することで、サブシステムを定義します。別の方法としては、**[プロパティ]** タブ (📄) の **[パラメータ詳細設定]** および **[変数詳細設定]** ツールを使用して、サブシステムのコンポーネントに対し、サブシステムのパラメータ、変数、初期値を直接設定することもできます。詳細設定を使用すると、1つまたは複数のデフォルト値を上書きすることができます。

パラメータ詳細設定

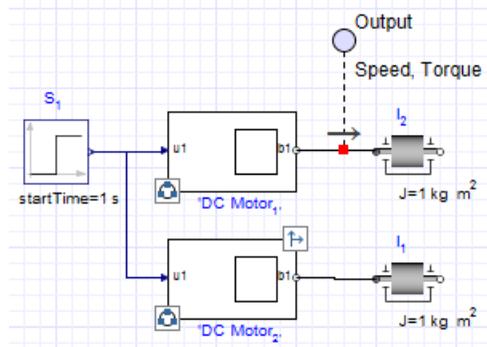
[パラメータ詳細設定] を使用すると、選択されたサブシステムコンポーネントのデフォルト値を上書きすることができます。必要であれば、パラメータ化機能 (🔑) を使用してオーバーライドをパラメータ化することもできます。ひとつのサ

ブシステムでのコンポーネントオーバーライドは、その他すべてのサブシステムから見えるパラメータに変換することができます。

次のモデルでは、Rの初期値にオーバーライドが適用され、パラメータ**Rcommon**に変換されています。



モデルワークスペースでは、パラメータオーバーライドを持つコンポーネントがオーバーライドアイコン () により識別されます。以下のモデルの **DC Motor₂** サブシステムにはパラメータオーバーライドが含まれます。



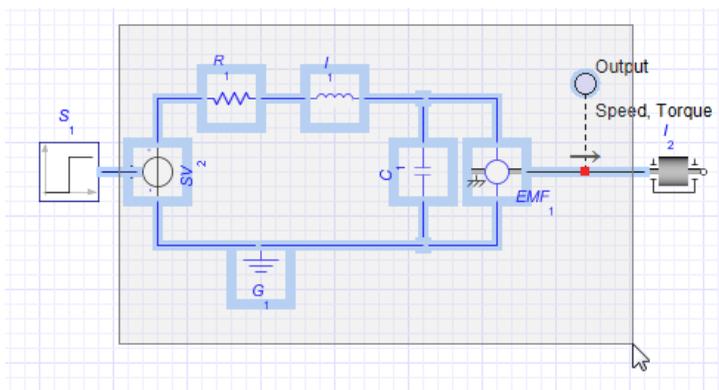
変数詳細設定

[**変数詳細設定**]を使用すると、サブシステムコンポーネントの初期条件を指定できます。[**変数詳細設定**]を選択すると、そのサブシステムのすべての設定可能なコンポーネントに対する初期条件フィールドが表示されます。

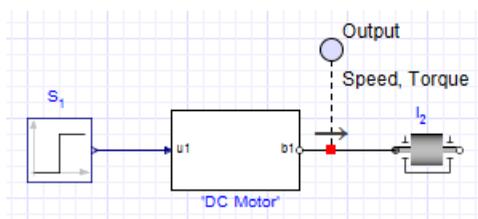
例：パラメータオーバーライドの作成

パラメータオーバーライドを作成するには

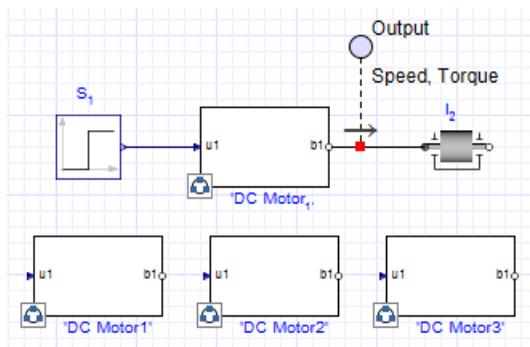
1. [ヘルプ]メニューの[例題]>[ユーザガイドの例題]>[第2章]から、[Simple DC Motor]の例題を選択します。
2. マウスをドラッグして、すべての電気コンポーネントを囲む四角形を描きます。



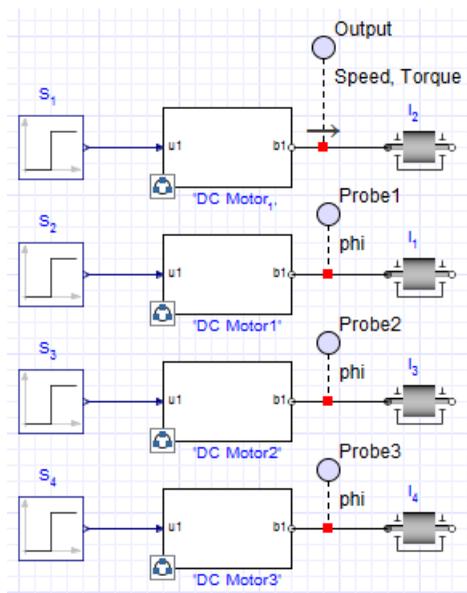
3. [編集]メニューから[サブシステムの作成]を選択するか、四角に囲まれた領域内を右クリック (Macでは[Control]キーを押しながらクリック) し、[サブシステムの作成]を選択します。
4. ダイアログボックスに「DC Motor」と入力し、[OK]をクリックします。DC Motor サブシステムが表示されます。



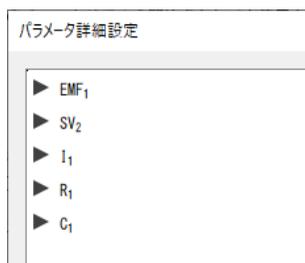
5. **DC Motor** サブシステムを右クリック (Macintosh では **[Control]** キーを押しながらクリック) し、**[共有サブシステムに変換]** を選択して **[OK]** をクリックします。これにより作成した共有サブシステム定義を、**[ローカルコンポーネント]** タブの **[コンポーネント]** パレットに追加します。
6. **[ローカルコンポーネント]** タブ (🔍) で **[コンポーネント]** パレットを展開し、DC Motor の共有サブシステムの3つのコピーを **モデルワークスペース** 内にドラッグします。



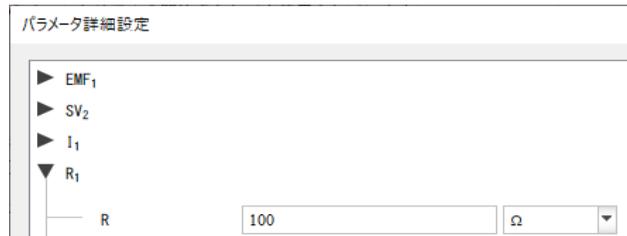
7. DC Motor サブシステムを3つ、下図のように作成します。



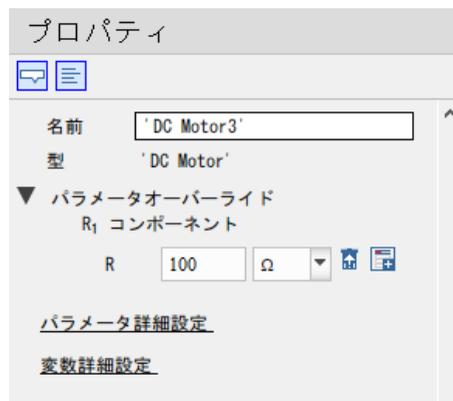
8. S_2 サブシステムをクリックし、[プロパティ] タブ () で $[T_0]$ を「1」に設定します。 S_3 と S_4 も同様に設定します。
9. DC Motor3 サブシステムをクリックし、[プロパティ] タブ () で [パラメータ詳細設定] をクリックします。[パラメータ詳細設定] ウィンドウが表示され、サブコンポーネントシステムすべてが表示されます。



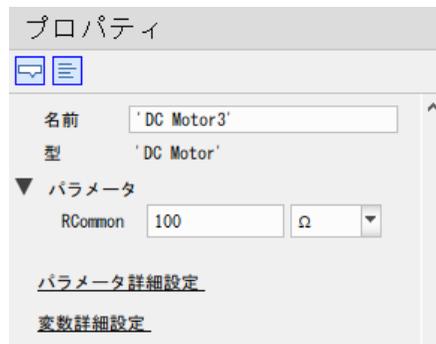
10. $[R_1]$ を開き、抵抗パラメータ (R) に値「100」を入力します。



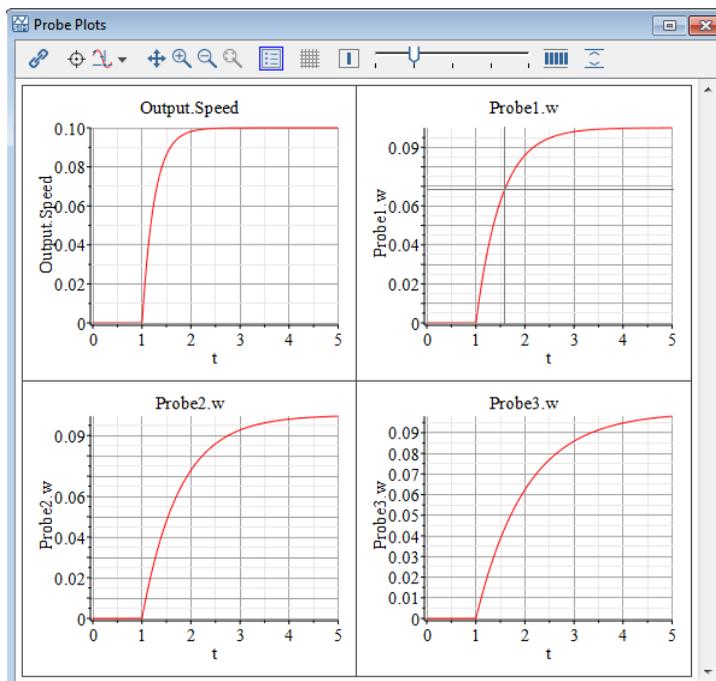
- 11 [OK] をクリックします。新しいパラメータが **[プロパティ]** タブにオーバーライドとして表示されます。



- 12 このオーバーライドを再利用可能なパラメータに変更するには、**パラメータ化** (🔗) をクリックし、新しいパラメータ名 **「Rcommon」** を入力して、[OK] をクリックします。[Rcommon] がほかのサブシステムで再利用できるパラメータとして **[プロパティ]** タブに表示されます。これはすでにオーバーライドではなくなっています。



- 13 その他のサブシステムでは、各サブシステムをクリックし、[プロパティ]タブの [Rcommon] に以下の値を入力します。
- DC Motor₁ では、[Rcommon] を「25」 [Ω] に設定
 - DC Motor1 では、[Rcommon] を「50」 [Ω] に設定
 - DC Motor2 では、[Rcommon] を「75」 [Ω] に設定
- 14 各サブシステムでプローブを選択し、[プロパティ] タブで [Speed] チェックボックスにチェックを付けて、ほかのすべてのチェックボックスのチェックを外します。
- 15 メインツールバーでシミュレーションの実行 (▶) をクリックします。それぞれのサブシステムに対し次のグラフが表示されます。



初期条件オーバーライドの指定

特定のサブシステムコンポーネントについて、既存の初期条件を上書きして初期条件値を設定することができます。コンポーネントを選択すると、[プロパティ]

タブ(📄)に利用可能な初期条件フィールドと既存のオーバーライドが表示され、コンポーネントのそれ以外の設定可能なパラメータ値も表示されます。

サブシステムを選択して **[変数詳細設定]** をクリックすると、サブシステムコンポーネントがすべて表示されます。コンポーネントを選択し、そのコンポーネントの初期値を指定することができます。この機能は、共有サブシステムを複数持つモデルに特に有用です。

2.7. モデルへのファイルの添付

[添付ファイル] タブ(📎)を使用すると、外部アプリケーションで作成されたスプレッドシートや設計文書など、あらゆる形式のファイルをモデルに添付することができます。**[添付ファイル]** タブで添付されたファイルは、後の MapleSim セッションでモデルを扱うときに参照できるように、現在のモデルの一部として保存することができます。ファイルを保存するには、添付ファイルを保存するカテゴリを右クリック (Macintosh では **[Control]** キーを押しながらクリック) し、**[ファイルを添付]** を選択します。

また、メニューバーで **[編集] > [ファイルを添付...]** を選択してモデルにファイルを添付することもできます。デフォルトでは、この方法を使用すると、ファイルは **[ドキュメント]** カテゴリに分類されます。この添付ファイルを移動するには、その項目をクリックして別のカテゴリにドラッグします。

下の図は、*CustomComponent.mw*、*NonLinearMSD.mw*、および *DamperCurve.xlsx* という名前のファイルを含む **[添付ファイル]** タブを示しています。



図2.14 添付

[添付ファイル] タブを使用して MapleSim テンプレートを開き、モデルで使用するカスタムコンポーネントやポートを作成することもできます。詳細については、本ガイドのモデルの解析と操作 [155ページ]を参照してください。

2.8. カスタムライブラリの作成と管理

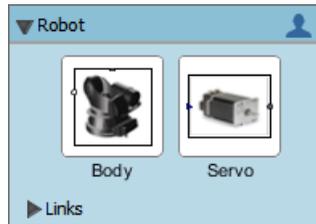
複数のファイルや MapleSim セッションで再利用したいサブシステム、およびカスタムコンポーネントを保存するためのカスタムライブラリを作成することができます。作成したカスタムライブラリは、MapleSim ウィンドウの左側にある[ライブラリコンポーネント]タブ(🔍)のカスタムパレットに表示され、ユーザのコンピュータに.msimlibファイルとして保存されます。将来の MapleSim セッションでこれらのカスタムパレットは MapleSim ウィンドウに表示されます。

ほかのパレットからコンポーネントを追加する場合と同様に、モデル構築時にはカスタムライブラリパレットからサブシステムやコンポーネントを利用できません。

カスタムライブラリを別のユーザと共有することも可能です。たとえば、ネットワークドライブ上にカスタムライブラリを格納した場合、対象の格納場所へのア

クセス権限を持つ別のユーザは、カスタムライブラリを各自の MapleSim セッションに読み込むことができます。

カスタムライブラリパレットは、[ライブラリコンポーネント] (🔍) タブの上部にアイコン 👤 で表示されます。カスタムパレットのサンプルを以下に示します。



カスタムライブラリの作成についての詳細は、MapleSim ヘルプシステムから **MapleSim 操作方法 > モデルの構築 > カスタムライブラリの作成と管理 > カスタムライブラリの作成** を参照してください。

Modelica 3.2.2 プログラミング言語に基づいたモデルライブラリやモデルを、他社のツールを使用して作成する場合、そのモデルライブラリやモデルの .mo ファイルを .msimlib ファイルとして MapleSim にインポートすることができます。そうした場合、インポートしたモデルやライブラリを、ほかのモデリングコンポーネントと同じように MapleSim モデルで使用することができます。詳細については、**MapleSim 操作方法 > モデルの構築 > Modelica のモデルとライブラリのインポート > Modelica ライブラリのインポート** を参照してください。

例：既存モデルからカスタムライブラリを作成

この例では、既存の MapleSim モデルの共有サブシステム定義からカスタムライブラリを作成します。カスタムライブラリに追加されたコンポーネントは、今後の MapleSim セッションで利用できます。

モデルからカスタムライブラリを作成するには

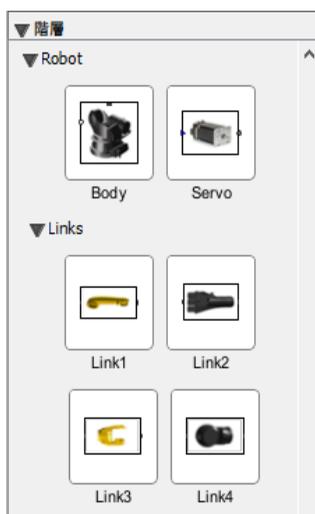
1. [ヘルプ] メニューの [例題] > [ドメイン別例題] > [マルチボディ] から [5 DoF Robot] の例題を選択します。

このモデルには、[ローカルコンポーネント]タブ()の[コンポーネント]パレットにリストされる6つの共有サブシステムが含まれます。

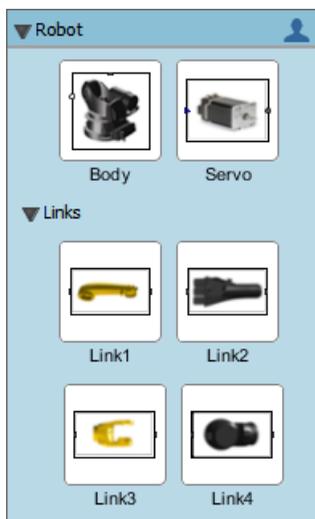
2. モデルを保存します。
3. [ツール]メニューから、[MapleSim ライブラリにエクスポート]を選択します。
4. [パッケージ]に「Robot」と入力してライブラリに名前を付けます。
注: 指定したパッケージ名は、MapleSim インターフェースに表示されるカスタムパレットの名前になります。
5. [OK]をクリックします。
6. ライブラリをエクスポートしたら、[閉じる]をクリックします。

ライブラリ編集モード(ワークスペースに透かし文字が、パラメータペインの[プロパティ]タブにライブラリプロパティが表示される)になります。MapleSim ウィンドウ左側の[ライブラリコンポーネント]タブ()に、新しいカスタムライブラリのパレットが表示されます。各要素の階層を定義していないため、パレットは空の状態になっています。

7. [ローカルコンポーネント]タブ()に切り替え、[階層]タブ内のルート名(Robot)下にコンポーネントをドラッグします。必要に応じて、要素をサブグループに整理します。

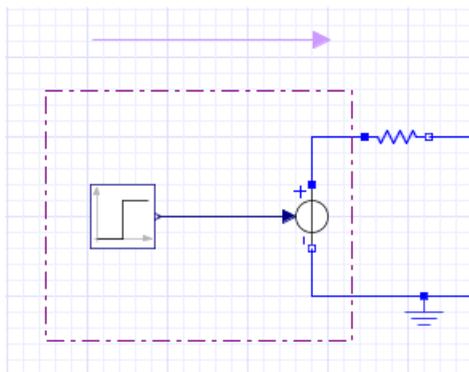


8. メインツールバーで現在のライブラリをリロード (🔄) をクリックして変更を保存し、[ライブラリコンポーネント] タブのカスタムライブラリにパレットと更新情報をリロードします。カスタムライブラリパレットにこれらのコンポーネントがすべて表示されるようになります。



2.9. モデルへの注釈の追加

モデルワークスペースツールバーに配置されているツールを使用すると、線や矢印、図形を描くことができます。また、MapleSim は色や、線のスタイル、塗りつぶしをカスタマイズするためのツールも多く備えています。



モデルワークスペースツールバーに配置されているテキストツール()を使用すると、モデルにテキスト注釈を追加することができます。テキスト注釈では、2-D Math 表記で数文を入力し、テキストのスタイル、色、フォントをフォーマットすることができます。2-D Math 表記の詳細は、*2-D Math 表記によるテキスト入力* [72ページ]を参照してください。

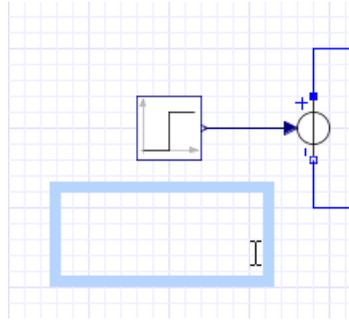
例：モデルへのテキスト注釈の追加

注釈をモデルに追加するには

1. [ヘルプ] メニューの [例題] > [ユーザガイドの例題] > [第 2 章] から、[Simple DC Motor] の例題を選択します。
2. アノテーションツールバーで、テキストツール() をクリックします。

注：アノテーションツールバーが表示されていない場合は、モデルワークスペースツールバーで描画ツールの表示/非表示() をクリックしてください。

3. モデルワークスペース内の Step コンポーネントの下に、注釈を入力するためのテキストボックスを描きます。



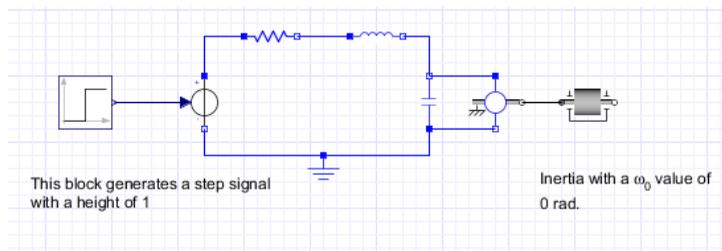
マウスの左ボタンを開放すると、**モデルワークスペース**の上のツールバーが書式設定のツールバーに切り替わります。



4. 次のテキストを入力します。 **「This block generates a step signal with a height of 1.」**
5. 入力したテキストを選択し、フォントを **Arial** に変更します。
6. テキストボックス外で、任意の場所をクリックします。
7. **Inertia** コンポーネントの下に別のテキストボックスを描きます。
8. 次のテキストを入力します。 **「Inertia with a ω_0 value of 0 rad.」**

ヒント：ギリシア文字のオメガ (ω) を入力するには、コンテキストバーで数学アイコン **Text Math** をクリックして 2D Math モードに切り替え、「**omega**」と入力し **[Esc]** キーを押します。下付きの値を入力するには、**[Ctrl] + [Shift] + 下線 (L_)** キー (Windows および Linux) または **[Command] + [Shift] + 下線 (L_)** キー (Macintosh) に続けて「**0**」を入力します。下付き文字の入力を終了するには、キーボードの右矢印キーを押します。コンテキストバーのテキストツールをクリックしてテキスト入力モードに切り替え、残りのテキストを入力します。

9. 入力したテキストを選択し、フォントを **Arial** に変更します。
- 10 注釈の入力を終了するには、テキストボックス外で任意の場所をクリックします。



2.10. 2-D Math 表記によるテキスト入力

パラメータ値および注釈には、2-D Math 表記のテキストを入力できます。2-D Math 表記とは、下付き文字、上付き文字、ギリシャ文字のような数学要素を入力する際に使用する書式オプションです。2-D Math 表記のテキストを入力する際、使用可能なMapleコマンドや数学記号のリストを表示する補完機能が使用できます。

2-D Math 表記で入力する場合は、書式設定ツールバーで**数学** (Math) を選択します。

下記はよく使用する 2-D Math 表記の入力ショートカット一覧です。

2-D Math 表記のキーの組み合わせ

タスク	キーの組み合わせ	例
コマンドおよび記号の補完(パラメータと注釈の入力時のみ有効)	<ol style="list-style-type: none"> 記号名、ギリシア文字、または Maple コマンドの先頭数文字を入力します。 プラットフォームに応じて、次のキーの組み合わせのいずれかを入力します。 <ul style="list-style-type: none"> [Esc] キー : Mac、Windows、Linux [Ctrl] + [Shift] + [Space] キー : Linux メニューから、挿入したい記号またはコマンドを選択します。 	-
変数に対する下付き文字の入力	[Ctrl] (Macintosh では [Command]) + [Shift] + 下線 ([_]) キー	x_a
上付き文字の入力	caret (^)	x^2
分数の入力	スラッシュ (/)	$\frac{1}{8}$

詳細は、MapleSim ヘルプシステムの [MapleSim 操作方法 > モデルの構築 > モデルへの注釈の追加 > 2-D Math 表記のキーの組み合わせ](#) を参照してください。

2.11. 補間テーブルコンポーネントのデータセットの作成

モデルの補間テーブルコンポーネントに値を与えるデータセットを作成し、たとえば、入力信号や、電気の **Current Table** および **Voltage Table** ソースなどにカスタム値を与えることができます。データセットを作成するには、カスタム値が格納されている Microsoft® Excel® スプレッドシート (.xls または .xlsx)、またはカンマ区切り (.csv) ファイルを添付するか、[データ生成] App または [乱数データの生成] App を使用して Maple にデータセットを作成します。これらの Apps は、

[Apps とテンプレートを追加] タブ () の [Apps] パレットから利用できます。

補間テーブルコンポーネントの詳細は、MapleSim ヘルプシステムの **MapleSim コンポーネントライブラリ > 信号ブロック > 補間テーブル > 概要**を参照してください。

例：Maple でのデータセットの作成

この例では、[データ生成] App を使用して MapleSim の **Lookup Table 1D** コンポーネントのためにデータセットを作成します。この App では、あらゆる Maple コマンドを使用してデータセットを作成できますが、説明のため、ここでは、定義済みの計算を使用してデータセットを作成します。

Maple でデータセットを作成するには

1. 新しい MapleSim ドキュメントを開きます。
2. [ライブラリコンポーネント] タブ () で [信号ブロック] パレットを展開し、[補間テーブル] メニューを開きます。
3. モデルワークスペースに **Lookup Table 1D** コンポーネントを配置します。
4. [Apps とテンプレートを追加] () をクリックします。
5. [Apps] パレットで、[データ生成] をダブルクリックします。[データ生成] App が [解析ウィンドウ] の [Apps] タブに開かれます。
6. App の下部に配置されている [データセット名] フィールドに「**TestDataSet**」と入力します。
7. MapleSim でデータセットを使用可能にするには、[MapleSim にデータ添付] をクリックします。
8. MapleSim の [添付ファイル] タブ () で [データセット] パレットを展開します。作成したデータセットファイルが一覧に表示されます。これで、このデータセットを **モデルワークスペース** の補間テーブルコンポーネントに設定できます。
9. モデルワークスペースで、**Lookup Table 1D** コンポーネントを選択します。
10. [プロパティ] タブ () の [datasourcemode] リストで [attachment] を選択します。

11 **[data]** ドロップダウンメニューで **TestDataSet.csv** を選択します。これで、データセットは **Lookup Table 1D** コンポーネントに代入されます。

12 モデルを MapleSim で保存します。

2.12. ベストプラクティス : モデルの構築

このセクションでは、MapleSim モデルのレイアウト (配置) および作成におけるベストプラクティス (最良の方法) について説明します。

ベストプラクティス : サブシステムの作成と配置

モデルを作成する場合は、まずコンポーネントをパレットから **モデルワークスペース** の中央にドラッグします。 **モデルワークスペース** 内のコンポーネントを望みどおりに再配置します。必要な場合、コンポーネントの向きを変えます。コンポーネントの位置と向きを定めたら、 **モデルワークスペース** でそれらを接続します。

コンポーネントをサブシステム化する場合、画面の表示範囲に収まるようにコンポーネントを選択してください。これによりサブシステムコンポーネントのすべてをスクロールせずに見ることができます。

再利用が考えられるコンポーネントグループをサブシステム化する

ダイアグラムの至るところ、または複数のファイルでの再利用が考えられるコンポーネントグループをサブシステム化してください。たとえば、振り子システムに複数の平面リンクモデルを含める場合、リンクサブシステムを作成すると、モデルにそのコンポーネントグループを複数コピーできます。リンクサブシステムを別の振り子モデルで使用する場合は、カスタムライブラリを作成することで、そのサブシステムは別のファイルでも使用できるようになります。

解析対象コンポーネントグループをサブシステム化する

より詳細な解析または試験、あるいはソースコードへの変換が考えられるコンポーネントグループをサブシステム化してください。解析の実行や、特定のサブシステムから方程式を取得するには、いくつかの MapleSim テンプレートが使用できます。コード生成テンプレートはサブシステムからしかソースコードを生成することができません。

解析タスクの実行に関する詳細は、本ガイドの [モデルの解析と操作](#) [155ページ] を参照してください。

デバッグコンソールを使用してサブシステムのコピーおよび未接続の結線を特定する

MapleSim ウィンドウの下部にある **モデルの診断情報を表示** () をクリックしてデバッグペインを表示します。

シミュレーション実行後、デバッグペインには、モデル作成時に発生するエラーに対処するための診断メッセージが表示されます。デバッグペインの上にある **診断テストの実行** () をクリック (または [編集] メニューから [モデルをチェック] を選択) すると、MapleSim はモデルに未接続結線、または同一内容のサブシステムでありながらサブシステム定義にリンクしていないサブシステムがないかを確認します。いずれかの問題が検出されると、問題箇所を含むサブシステムを特定するメッセージがデバッグコンソールに表示されます。デバッグペインのメッセージを右クリック (Macintosh では [Control] キーを押しながらクリック) すると、問題の解決に役立つ表示オプションが表示されます。

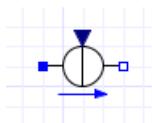
ベストプラクティス: 電気モデルの作成

電気回路に Ground コンポーネントを配置する

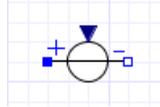
どんな電気回路モデルにも、電圧信号に対して基準を示す **Ground** コンポーネントを配置し、接続する必要があります。

電流源および電圧源の接続を確認する

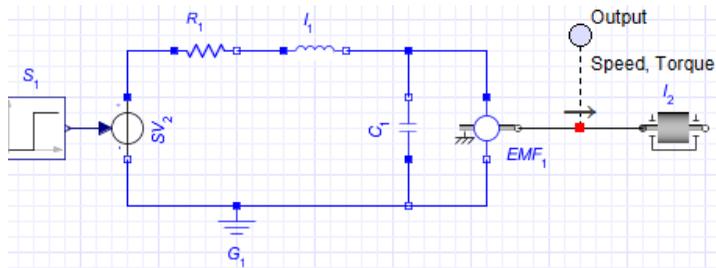
シミュレーションの結果は、電流源や電圧源がモデルでどのように接続されているかによって変わります。シミュレーション結果が予期しないものであった場合は、モデルに配置されているほかのコンポーネントと電源のあいだの接続を確認してください。MapleSim コンポーネントライブラリに用意されている電流源では、正電流の方向が矢印で示されます。



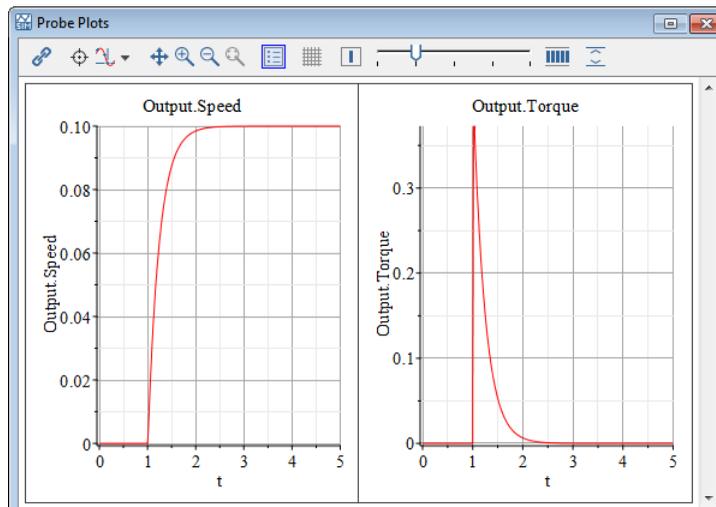
また、電圧源では、正の電圧の位置がプラス記号、負の電圧の位置がマイナス記号で示されます。



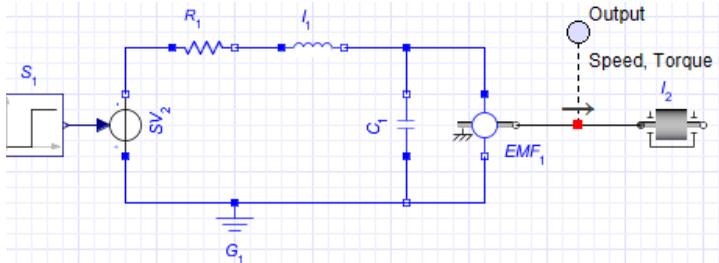
たとえば、下の **Simple DC Motor** モデルでは、ダイアグラム左側に配置されている **Signal Voltage** ソースのプラスのポートは **Resistor** コンポーネントのプラスのポートに接続されています。



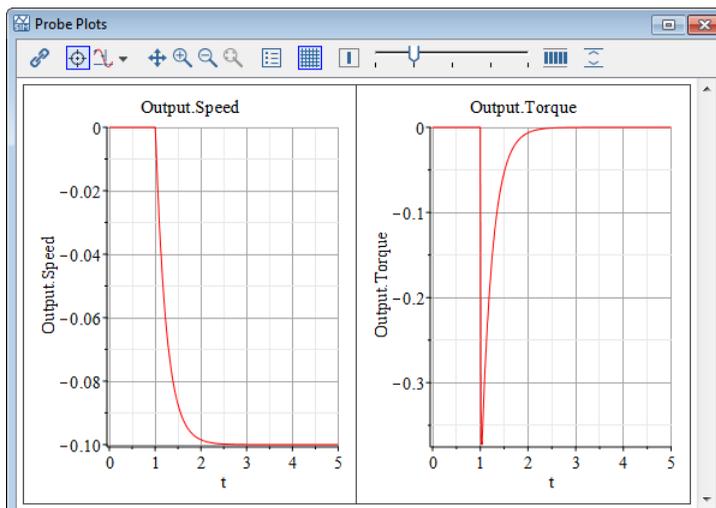
このモデルをシミュレートした場合、MapleSim は下に示すトルクと速度の量を結果として返します。



逆に、下の図に示されているように、**SignalVoltage**ソースのマイナスのポートが **Resistor** コンポーネントのプラスのポートに接続されているとします。



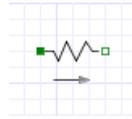
その場合、MapleSim は異なる速度とトルク量の結果を返します。



ベストプラクティス：1-D 並進モデルの作成

力を示す矢印の向きがすべて同じであることを確認する

MapleSim では、1-D メカニカルな並進コンポーネントはすべて 1-D 座標系で定義され、正の方向はコンポーネントのアイコンとともに表示される灰色の矢印の方向として定義されます。



モデルに力が加えられると、コンポーネントは矢印の方向に移動します。このため、モデルにおける 1-D メカニカル の並進コンポーネントの向きがすべて同じであることを確認する必要があります。たとえば、下記のスプリングダンパーモデルでは、力を示す矢印はすべて右向きになっています。

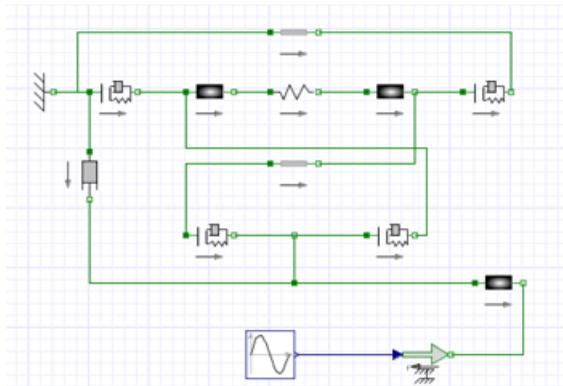


図2.15 力を示す矢印の確認

符号の規則の例および矢印の方向がモデル上で作用する力をどのように表すかについては、[ヘルプ]メニューの[例題] > [ユーザガイドの例題] > [第 2 章] から [Constant Acceleration]、[Sign Convention]、または [Arrow Convention] のいずれかの例題を参照してください。

ベストプラクティス：マルチボディモデルの作成

Rigid Body Frame の内側のポートは重心フレームに接続する

モデルに配置されている **Rigid Body Frame** コンポーネントの内側のポートは **Rigid Body** コンポーネントの重心フレームに接続される必要があります。これにより、**Rigid Body Frame** コンポーネントの変位および回転を表すローカル座標系が、**Rigid Body** コンポーネントで定義されている重心系と確実に等しくなります。

下に示す平面リンクの例では、**Rigid Body Frame** の内側のポート (つまり、 のアイコンを持つポート) は両方とも **Rigid Body** コンポーネントに接続されています。

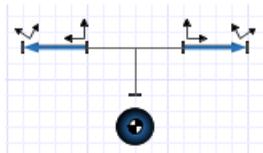


図2.16 重心配置のベストプラクティス

ベストプラクティス：油圧モデルの作成

流体特性を定義する

油圧モデルを作成する場合は、モデルのトップレベルまたは油圧サブシステムと同じレベルに **Hydraulic Fluid Properties** コンポーネントを配置して、流体の特性を定義する必要があります。このコンポーネントをモデルのトップレベルに配置すると、モデルにおけるすべての油圧コンポーネントとサブシステムがそのコンポーネントで定義されている流体特性を受け継ぎます。一方、**Hydraulic Fluid Properties** コンポーネントをサブシステムと同じレベルに配置すると、そのコンポーネントで定義される特性はそのサブシステムやその中に入っているすべてのサブシステムに含まれている油圧コンポーネントが受け継ぎます。

下の例では、図の右上に示されている **Hydraulic Fluid Properties** コンポーネントで定義される流体特性はモデルに配置されているすべての油圧コンポーネントに受け継がれます。

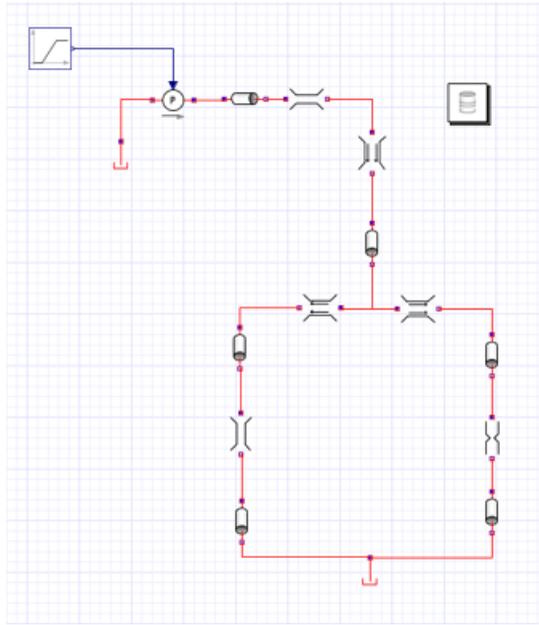


図2.17 油圧モデル

油圧システムのモデリング方法に関する完全なチュートリアルについては、チュートリアル 8：油圧システムのモデリング [253ページ] を参照してください。

ベストプラクティス：初期条件の強制

複雑なモデルでは、初期条件がすべて相互に独立しているとは限らない場合があります。一般的には、モデルの自由度により、**enforce** (👍) オプションを使用し、初期条件を指定した値に強制的に設定することができます。また、指定した初期条件のパラメータ値に対して **guess** オプション (🔍) を使用して、ソルバがシステムの適切な初期状態をより速く探せるようにすることもできます。

第3章 カスタムコンポーネントの作成

この章の内容は以下のとおりです。

- [カスタムコンポーネントについて \[83ページ\]](#)
- [信号フローの動作を使用したカスタムコンポーネントの作成 \[87ページ\]](#)
- [物理的な接続を持つカスタムコンポーネントの作成 \[93ページ\]](#)
- [MapleSim でのカスタムコンポーネントの使用 \[95ページ\]](#)
- [例: 非線形バネダンパカスタムコンポーネントの作成 \[97ページ\]](#)

3.1. カスタムコンポーネントについて

カスタムコンポーネントを作成して MapleSim コンポーネントライブラリを拡張すると、ユーザが定義した数理モデルに基づいてカスタムコンポーネントを作成できます。カスタムコンポーネントでは、物理ドメインに関連付けられた信号やポートを使用したり、この2つを組み合わせ使用できます。また、カスタムコンポーネントのライブラリを作成したり、特別な機能を持つ特定のサブシステムを含むカスタムコンポーネントも作成できます。

ドメイン固有のカスタムコンポーネントの作成方法に関するチュートリアルについては、[チュートリアル5: カスタムコンポーネントテンプレートの使用 \[213ページ\]](#)を参照してください。

以下にいくつかのカスタムコンポーネントテンプレートを示します。これらは、**[Apps とテンプレートを追加]** タブ () の **[テンプレート]** パレットから利用できます。

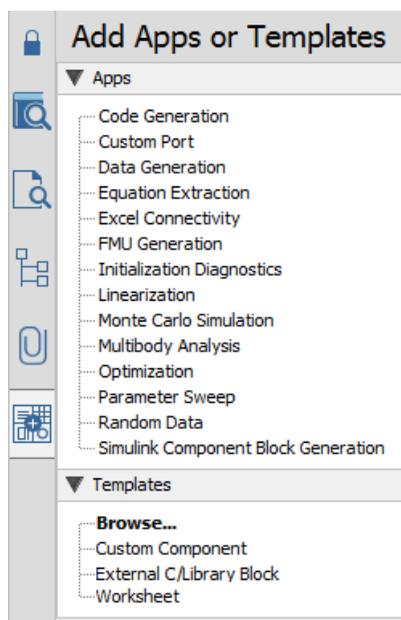


図3.1 [Appsとテンプレートを追加] タブ

シンプルなカスタムコンポーネントの作成

MapleSim モデルのカスタムコンポーネントを作成する一般的なプロセスには、カスタムコンポーネントのコンポーネント方程式、コンポーネントパラメータ、およびシステムモデルの指定、ポートタイプと値の指定、およびコンポーネントの生成が含まれます。

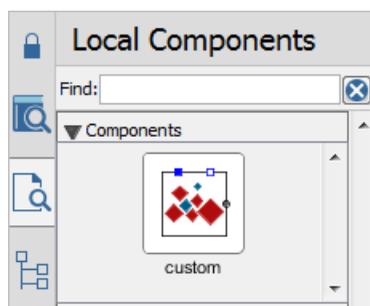
カスタムコンポーネントを作成するには

1. MapleSim モデルを新規作成し、[Appsとテンプレートを追加] タブ () を選択します。
2. [テンプレート] パレットで [カスタムコンポーネント] をダブルクリックします。
3. [添付を作成] ウィンドウにテンプレート名を入力し、添付を作成 () をクリックします。DAE カスタムコンポーネントテンプレートが Maple で開かれます。

4. **[方程式の定義]** セクションで、カスタムコンポーネントの方程式を入力します。方程式、パラメータおよび初期条件はすべてここに入力します。行の最後で **[Enter]** キーを押します。

$$eq := \left[s(t) = sa(t) - sb(t), 0 = Fa(t) + Fb(t), Fa(t) = \text{piecewise}\left(s(t) < 0, K \cdot s(t) + B \cdot \frac{d}{dt}s(t), 0\right), K = 1000, B = 10 \right]$$

5. **[設定]** セクションで **[パラメータ]** を選択し、**[すべて更新]** をクリックしてから、モデルパラメータにデフォルト値とタイプを割り当てます。
6. **[変数]** を選択し、**[すべて更新]** をクリックしてから、モデル変数に初期値とタイプを割り当てます。
7. **[ポート]** を選択し、**[ポートを追加]** をクリックしてカスタムコンポーネントにポートを追加します。カスタムコンポーネントで使用するポートのレイアウトやアイコンも設定できます。
 カスタムポートを定義することができます。この場合、あらかじめ**カスタムポート** Apps を使用してカスタムポートを定義しておく必要があります。定義が完了すると、この Apps のカスタムポートの使用が可能になります。カスタムポートの使用方法に関する完全なチュートリアルについては、第6章の例：**カスタムポート [226ページ]**を参照してください。
8. ポートのドメイン、スタイル、ポート名、およびポート変数についての詳細を入力します。
9. **コンポーネントの生成**セクションに、コンポーネント名を入力します。これは、MapleSim の **[ローカルコンポーネント]** タブ () の **[コンポーネント]** パレットで、このカスタムコンポーネント名として表示されます。
10. **[MapleSim コンポーネントを生成]** をクリックしてコンポーネントを作成し、MapleSim 環境に戻ります。カスタムコンポーネントは、**[ローカルコンポーネント]** タブの **[コンポーネント]** パレットに表示されます。



一般的な使用方法

カスタムコンポーネントテンプレートは最も汎用的なテンプレートです。具体的には、代数式、微分方程式、または微分代数方程式からカスタムコンポーネントを作成することを目的としています。カスタムコンポーネントテンプレートは、新しいMapleSimコンポーネントを簡単に作成するための、特定のMapleコマンドに関連するプリビルトのコントロールとプロシージャの集合です。カスタムコンポーネントのほか、**Modelica カスタムコンポーネント**を使用すると、ユーザが提供するModelicaコードを使用してカスタムコンポーネントを作成できます。

カスタムコンポーネントテンプレートは、単なる方程式のコンテナではありません。モデルのカスタムコンポーネントを生成する前に、Mapleのすべての機能を使用して方程式を開発することもできます。これには、コンポーネントの動作をすぐに解析して確認するためのMapleのプログラミング言語、数式処理機能、およびドキュメントツールへのアクセスも含まれます。

カスタムコンポーネントテンプレートを使用して以下のタスクを実行し、Mapleでカスタムコンポーネントを作成します。

- モデルにカスタムコンポーネントテンプレートを添付する
- コンポーネントの動作を決める支配方程式およびプロパティを定義する (パラメータ、ポート変数など)
- コンポーネントのポートを指定する
- 関連するポート変数のマッピングを定義する
- 方程式の変数をポートにマッピングする
- コンポーネントを生成し、MapleSimで使用可能にする

- ユーザの数理モデルをテストおよび解析する

カスタムコンポーネントテンプレートには、組み込みコンポーネントと同じ検証を行い、接続とパラメータの無効な値を排除しながら、これらのタスクを実行できるプリビルトのコントロールが含まれています。

カスタムコンポーネントテンプレートの使用

このテンプレートの使用方法の詳細については、MapleSim DAE カスタムコンポーネントテンプレートの使用ヘルプページを参照してください。

3.2. 信号フローの動作を使用したカスタムコンポーネントの作成

カスタムコンポーネントは、多くの信号フローコンポーネントをまとめて接続する必要性を減らすことで、モデルの構築を簡単にします。この例では、シンプルな信号フロー方程式のカスタムコンポーネントを作成する方法について示します。

シンプルな信号フローのカスタムコンポーネントの作成

次の方程式を実装するカスタムコンポーネントを作成します。

$$x(t) = y(t) + z(t)$$

カスタムコンポーネントを作成するには

1. MapleSim モデルを新規作成し、[Apps とテンプレートを追加] () をクリックします。
2. [テンプレート] パレットで [カスタムコンポーネント] をダブルクリックします。
3. 添付名として「**custom**」と入力し、添付を作成 (✓) をクリックします。
4. [方程式の定義] セクションで、以下の方程式を入力します。

$$eq := [x(t) = y(t) + z(t)];$$

注: eq には方程式のリストが割り当てられる必要があるため、この方程式は角括弧で囲まれています。

5. **[Enter]** キーを押し、方程式を登録します。

$$eq := [x(t) = y(t) + z(t)];$$

$$[x(t) = y(t) + z(t)]$$

図3.2 カスタムコンポーネントを定義する方程式

ヒント: カスタムコンポーネントの方程式を明示的な形式に置き換える必要はありません。たとえば、この方程式を次のように置き換えることができます。

$$eq := [x(t) + \log(x(t)^2) = y(t) + z(t)];$$

この場合、出力 $x(t)$ には明示的な解がありません。MapleSim は $x(t)$ の解を自動的に求めます。

6. **[設定]** セクションで **[ポート]** を選択し、**[すべて更新]** をクリックしてデータテーブルを更新します。
7. **[全ポートを削除]** をクリックします。
8. **[ポートを追加]** を3回クリックして以下の場所にドラッグし、新しいポートを3つ追加します。

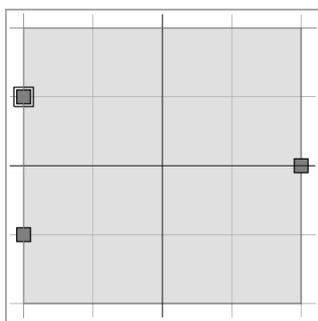


図3.3 ポートのマッピング

9. 左側の一番上のポートをクリックして選択します。

- 10 [タイプ] ドロップダウンリストから [Real Signal] を選択します。[スタイル] では、[in] ラジオボタンを選択します。ポートの名前はデフォルトで `real_i` になります。
- 11 次に、変数をポート信号に関連付けます。図3.4 「ポートマッピングの変数」に示すように、ドロップダウンリストから `y(t)` を選択します。

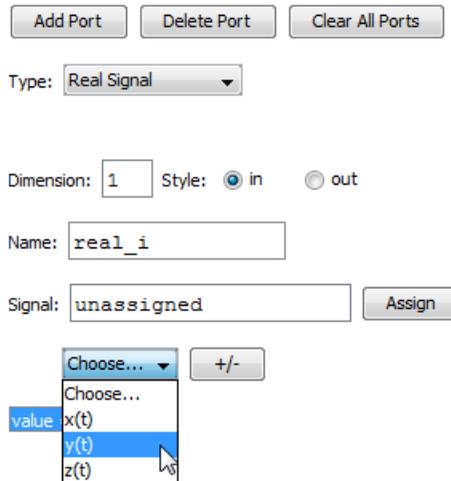


図3.4 ポートマッピングの変数

- 12 表3.1 「ポートのマッピング」の設定を使用して、残りのポートマッピングを割り当てます。

ポートのマッピング

ポートの名前	ポートの位置	ポートの種類	ポートのスタイル	ポートのコンポーネント
real_i	左上	Real Signal	in	$y(t)$
real1_i	左下	Real Signal	in	$z(t)$
real_o	右	Real Signal	out	$x(t)$

- 13 [アイコン] リストから、[デフォルトを使用] を選択します。
- 14 [コンポーネントの生成] セクションで、[名前] フィールドに「**custom**」と入力します。
- 15 [MapleSim コンポーネントを生成] をクリックします。カスタムコンポーネントの方程式が生成され、モデルに代入されます。カスタムコンポーネントのアイコンが、MapleSimの[ローカルコンポーネント] タブの[コンポーネント] パレットに表示されます。

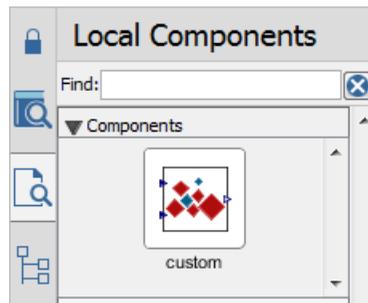
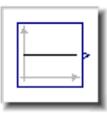
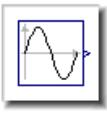


図3.5 生成されたカスタムコンポーネント

- 16 表3.2 「信号フローコンポーネント」 に示すように、カスタムコンポーネントを [コンポーネント] パレットからワークスペースにドラッグします。

信号フローコンポーネント

コンポーネント	コンポーネント数	シンボル	ライブラリの場所	必要な設定
カスタムコンポーネント	1		ローカルコンポーネント (🔍) > [コンポーネント]	カスタム設定
Constant	1		ライブラリコンポーネント (🔍) > [信号ブロック] > [ソース] > [実数]	デフォルト設定を使用
Sine	1		ライブラリコンポーネント (🔍) > [信号ブロック] > [ソース] > [実数]	デフォルト設定を使用

17. 2つの信号ソースを左側の入力ポートに接続し、一番右のポートにプローブを配置 (右クリックして [プローブを追加] を選択) します。

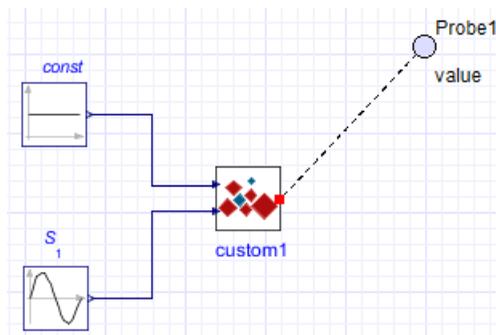
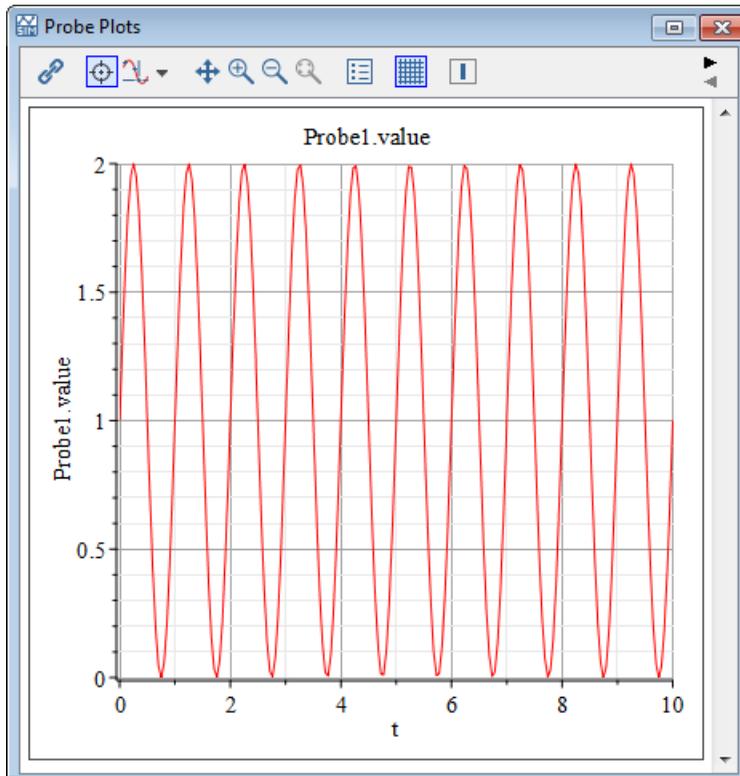


図3.6 完成したカスタムコンポーネントモデル

18. メインツールバーでシミュレーションの実行 (▶) をクリックします。シミュレーションが終了すると、下のグラフが表示されます。



非因果的マッピングの利点

カスタムコンポーネントのポートは入力信号および出力信号として指定されますが、実際には MapleSim は非因果的であるため、ピンのタイプに関係なく信号を入力または出力にすることができます。たとえば、 $x(t)$ と $z(t)$ が指定され、 $y(t)$ にプローブが配置されると、MapleSim は自動的に指定された方程式を $y(t) = x(t) + z(t)$ に置き換えます。

入力信号と出力信号の概念は、MapleSim のコード生成機能に必要です。これは、コードが「因果的」であり、MapleSim が入力を期待して出力を提供するためです。

カスタムコンポーネントでの微分方程式の使用

モデルにライブラリのコンポーネントを使用する代わりに、微分方程式を使用してカスタムコンポーネントを定義することもできます。たとえば **図3.7 「2つの質量バネダンパの方程式」** は、ひとつの外力を受ける2つの質量スプリングダンパーの組み合わせの運動を示す方程式を表しています。

$$\begin{aligned}
 \text{eq} := & \left[m1 \cdot \frac{d^2}{dt^2} x1(t) = -K1 x1(t) - B1 \frac{d}{dt} x1(t) - K2 (x1(t) - x2(t)) - B2 \left(\frac{d}{dt} x1(t) \right. \right. \\
 & \left. \left. - \frac{d}{dt} x2(t) \right), m2 \frac{d^2}{dt^2} x2(t) = -K2 (x2(t) - x1(t)) - B2 \left(\frac{d}{dt} x2(t) - \frac{d}{dt} x1(t) \right) + F(t) \right]; \\
 \text{params} := & [K1 = 1, B1 = 1, K2 = 1, B2 = 1, m1 = 1, m2 = 1]; \\
 \text{initialconditions} := & [x1(0) = 0, x2(0) = 0, D(x1)(0) = 0, D(x2)(0) = 0];
 \end{aligned}$$

図3.7 2つの質量バネダンパの方程式

図3.8 「2つの質量バネダンパのポートのマッピング」 に、パラメータをコンポーネントのポートにマッピングする方法を示します。

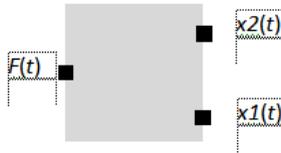


図3.8 2つの質量バネダンパのポートのマッピング

3.3. 物理的な接続を持つカスタムコンポーネントの作成

物理的な接続に基づいてカスタムコンポーネントを作成する場合、それぞれの接続ポートには関連する2つの変数(介在変数と通過変数)があります。介在変数はシステム内の原動力(温度差、気圧差、電圧降下、速度、または相対的な角速度など)を表し、通過変数は、保存量(熱、質量、電流、力、トルクなど)の流れを表します。

通過変数と介在変数の特徴

通過変数の特徴	介在変数の特徴
保存量 (熱や質量など) フロー方向がある 入力 = 出力 + 蓄積という関係を満たす ドメイン間で同一	保存量の流れを動かす スカラーである 物理ドメイン内の2点間の違いを定義する

表3.4「通過変数と介在変数の数学的関係」に、さまざまな介在変数および通過変数間の接続を定義する数学的関係を示します。

通過変数と介在変数の数学的関係

ドメイン	支配方程式	通過変数	介在変数
オームの法則 電気ドメイン	$I = \frac{V}{R}$	I	V
ハーゲンポアズイコ式 油圧ドメイン	$\frac{dm}{dt} = \frac{\pi D^4 \rho}{128 L \mu} \cdot P$	$\frac{dm}{dt}$	P
フーリエの法則 伝熱ドメイン	$\frac{dQ}{dt} = h A T$	$\frac{dQ}{dt}$	T

抵抗のシステム方程式の抽出

表3.5「抵抗の変数とパラメータ」に、複数の変数と1つのパラメータを持つシンプルな抵抗のモデルを示します。

抵抗の変数とパラメータ

変数	パラメータ	説明
i(t)		電流
v(t)		電圧
vLeft(t)		左側のポートの電圧
vRight(t)		右側のポートの電圧

変数	パラメータ	説明
	R	抵抗

オームの法則では、電圧と電流を次のように定義します。

$$v(t) = V_{right(t)} - V_{left(t)}$$

$$v(t) = i(t) \cdot R$$

図3.9「抵抗のポートのマッピング」は、カスタムコンポーネントのポートにマッピングされた方程式を示しています。



図3.9 抵抗のポートのマッピング

右側のポートの電流 $i(t)$ にはマイナス記号があり、抵抗から流出していることを表しています。左側のポートの電流はプラスであり、抵抗へ流入していることを表しています。抵抗 (R) は、[プロパティ] タブ (📄) で使用可能なパラメータとして定義されています。

3.4. MapleSim でのカスタムコンポーネントの使用

MapleSim では、カスタムコンポーネントをサブシステムと同じように扱うことができます。カスタムコンポーネントに対して実行できるタスクは以下のとおりです。

- カスタムコンポーネントを現在のモデルの一部として保存する
- カスタムコンポーネントをカスタムライブラリに追加する
- カスタムコンポーネントを編集する
- カスタムコンポーネントの例題を開く

カスタムコンポーネントを現在のモデルの一部として保存する

モデルを保存するとき、カスタムコンポーネントがモデルの一部として保存されます。**[編集]**メニュー>**[モデルの整理]**をクリックすると、未使用のカスタムコンポーネント定義がモデルから削除されます。カスタムコンポーネントが**モデルの整理**機能によって削除されることを防ぐには、以下の手順を実行します。

- カスタムコンポーネントを**[ローカルコンポーネント]**タブの**[コンポーネント]**パレットから**[階層]**パレットに移動します。
- モデル内のカスタムコンポーネントを使用するには、カスタムコンポーネントを**[コンポーネント]**パレットからモデルワークスペースにドラッグします。

詳細は、MapleSim ヘルプシステムで、**MapleSim 操作方法 > モデルの構築 > モデルの整理**を参照してください。

カスタムコンポーネントをカスタムライブラリに追加する

カスタムコンポーネントを現在のモデル以外のファイルで使用するには、そのコンポーネントをカスタムライブラリに追加します。詳細は、**カスタムライブラリの作成と管理 [66ページ]**を参照してください。

カスタムコンポーネントを編集する

生成したカスタムコンポーネントを編集するには、対応するMapleワークシートで修正し、コンポーネントを再生成します。

カスタムコンポーネントを編集するには

1. MapleSim **モデルワークスペース**で、編集したいカスタムコンポーネントをダブルクリックします。Maple が起動し、当該カスタムコンポーネントテンプレートが開きます。
2. Maple ワークシートで、数式、プロパティ、またはポート値を編集します。
3. ワークシートの下部に配置されている**[MapleSim カスタムコンポーネントを生成]**をクリックします。MapleSim に表示されているカスタムコンポーネントに変更箇所が生成されます。
4. カスタムコンポーネントを追加した .mw ファイルおよび .msim ファイルの変更内容を保存します。

3.5. 例 : 非線形バネダンパカスタムコンポーネントの作成

この例では、カスタムコンポーネントテンプレートを使用して、非線形バネダンパコンポーネントを作成します。この例に定義されている方程式は、MapleSim の **Translational Spring Damper** コンポーネントに基づいているため、剛性と減衰係数は、コンポーネントへの入力関数に置き換えます。

支配法則を得るために、まず free-body diagram (自由物体図) を描きます。このスプリングダンパシステム図を以下に示します。

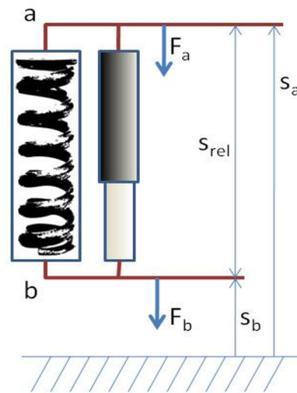


図3.10 非線形バネダンパカスタムコンポーネント

終点の a と b は、コンポーネントのポートとして定義することができ、方程式はこれらのポートに対して導き出します。その場合、運動の一般式は次のようになります。

$$d \cdot \frac{d}{dt} s_{rel}(t) + c \cdot s_{rel}(t) = F(t)$$

ここで、 d は減衰係数、 c はバネの剛性、 s_{rel} は次の式で表すことができる s_a と s_b の 2 つのポート間における相対変位を表しています。

$$s_{rel}(t) = s_b(t) - s_a(t)$$

また、システムにかかる力を見ると、合力は $F(t) = F_b(t)$ で、次の等式が成り立つことがわかります。

$$F_a(t) + F_b(t) = 0$$

上記は、すべてシステムの動作を定義するために必要な関係です。

カスタムコンポーネントテンプレートを開く

カスタムコンポーネントテンプレートは MapleSim テンプレートの一部で、**メインツールバー**からアクセスします。

カスタムコンポーネントテンプレートを開くには

1. MapleSim で、カスタムコンポーネントを追加するモデルを開きます。
2. **[Apps とテンプレートを追加]** タブ () を選択します。
3. **[テンプレート]** パレットで **[カスタムコンポーネント]** をダブルクリックします。
4. テンプレートの名前に **「Nonlinear Spring-Damper」** と入力し、**[添付を作成]** () をクリックします。Maple が起動し、カスタムコンポーネントテンプレートが開きます。

コンポーネントの名前と方程式の定義

次に、MapleSim インターフェースに表示されるコンポーネント名を指定し、方程式を定義します。

カスタムコンポーネントを定義するには：

1. **方程式** セクションで以下の方程式を入力し、非線形系を定義します。

>

```
eq := [d(t)*(diff(s_rel(t), t)) + c(t)*s_rel(t) = F(t), s_rel(t) = s_b(t)
      - s_a(t), v_rel(t) = diff(s_rel(t), t), F(t) = F_b(t), F_a(t) + F_b(t) = 0]
;
```

方程式は、Maple リストに入力されることにご注意ください。定数の d (減衰) と c (剛性) は、システムへの入力状態として定義するために、関数の $d(t)$ と $c(t)$ に置き換えます。

2. 方程式にカーソルを置いて、**[Enter]** キーを押します。
3. **[設定]** セクションで**[変数]** を選択し、**[すべて更新]** をクリックして変数のリストを更新します。

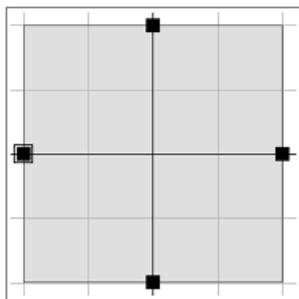
これで、生成されるカスタムコンポーネントに追加するポートに入出力変数を代入することができます。

コンポーネントのポートの定義

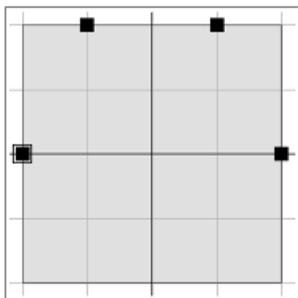
テンプレートのポートセクションでは、生成されたコンポーネントに表示されるポートに代入する入出力変数とポートの配置を指定します。

ポートを定義するには

1. **[設定]** セクションで**[ポート]** を選択します。
2. **[全ポートを削除]** をクリックします。
3. **[ポートを追加]** を4回クリックします。ダイアグラムに4つの四角形が表示されます。これらは、ユーザが配置し、定義するポートです。



4. ポートをドラッグして、左右両側に1つずつ、上側に2つ配置します。



5. ダイアグラム左側のポートを選択します。
6. [タイプ] ドロップダウンメニューから、[Translational] を選択します。
7. ポートのスタイルには、[b] を選択します。ポートの名前はデフォルトで **tflange_b** になります。
8. 最初に、位置変数を定義します[変数] の下のリストで [Position = unassigned] が選択されていることを確認します。[変数] の下のドロップダウンリストから [s_b(t)] を選択します。
9. 次に、力変数を定義します。[変数] の下のリストで [Force = unassigned] を選択します。[変数] の下のドロップダウンリストから [F_b(t)] を選択します。これで、左側のポートは位置変数 **s_b(t)** と力変数 **F_b(t)** が関連付けられている並進フランジとして定義されたことになります。
10. ダイアグラム右側のポートを選択します。
11. [タイプ] ドロップダウンメニューから、[Translational] を選択します。
12. ポートのスタイルには、[a] を選択します。ポートの名前はデフォルトで **tflange_a** になります。
13. 最初に、位置変数を定義します。[変数] の下のリストで [Position = unassigned] が選択されていることを確認します。[変数] の下のドロップダウンリストから [s_a(t)] を選択します。
14. 次に、力変数を定義します。[変数] の下のリストで [Force = unassigned] を選択します。[変数] の下のドロップダウンリストから [F_a(t)] を選択します。これで、右側のポートは位置変数 **s_a(t)** と力変数 **F_a(t)** が関連付けられている並進フランジとして定義されたことになります。
15. ダイアグラム左上部のポートを選択し、以下の手順に従います。

- [タイプ] ドロップダウンメニューから **[Real Signal]** を選択します。
- [スタイル] で、[in] を選択します。
- [名前] を「cin」に変更します。
- [信号] の下のドロップダウンリストから **[c(t)]** を選択します。

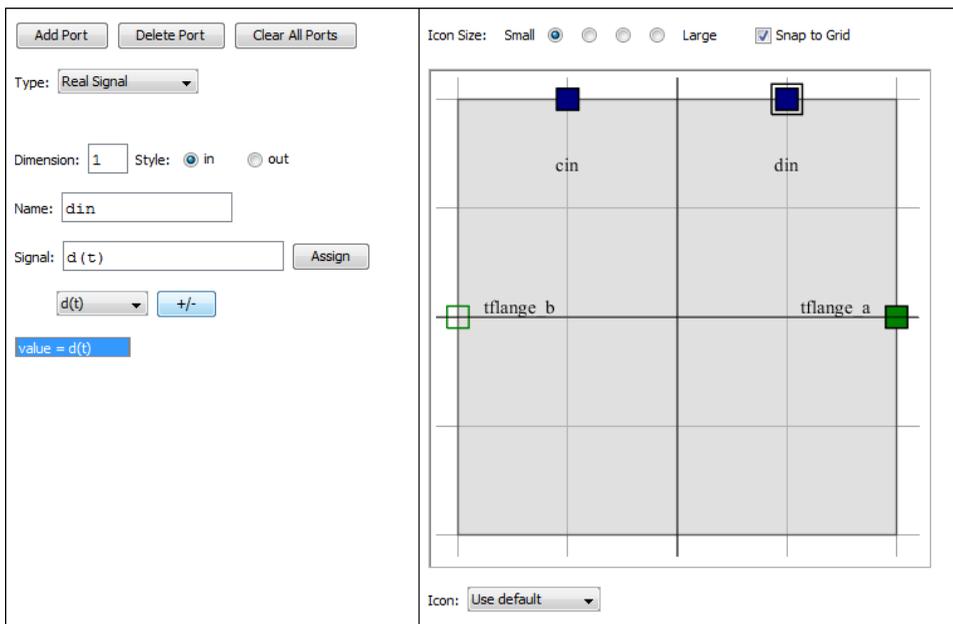
これで、このポートは剛性変数 $c(t)$ が関連付けられている入力信号として定義されたことになります。

16 ダイアグラム右上部のポートを選択し、以下の手順に従います。

- [タイプ] ドロップダウンメニューから **[Real Signal]** を選択します。
- [スタイル] で、[in] を選択します。
- [名前] を「din」に変更します。
- [変数] の下のドロップダウンメニューから **[d(t)]** を選択します。

これで、このポートは減衰変数 $d(t)$ が関連付けられている入力信号として定義されたことになります。

17. [アイコン] リストから、[デフォルトを使用] を選択します。



MapleSim で生成されるカスタムコンポーネントでは、ポートはこの配置で表示されます。

次元のチェック

この操作は省略可能です。

1. **[設定]** セクションで **[次元解析]** を選択し、**[次元のチェック]** をクリックします。単位系に矛盾があるシステム方程式の代数式は、数式コンポーネントに表示されます。
2. 単位系を修正するには、**[変数]** セクションで、**[タイプ]** 列のシステム変数に以下の単位を割り当てます。
 - $c(t)$ には、「**Force/Distance**」を入力します。
 - $d(t)$ には、「**Force/Velocity**」を入力します。
 - $F(t)$ には、「**Force**」を入力します。
 - $s_{\text{rel}}(t)$ には、「**Length**」を入力します。
 - $v_{\text{rel}}(t)$ には、「**Velocity**」を入力します。
3. **[すべて更新]** をクリックします。代数式 **Force/Distance** および **Force/Velocity** は、対応する Modelica タイプ、**TranslationalSpringConstant** および **TranslationalDampingConstant** にそれぞれ変換されます (直接これらの値を入力することも可能)。
4. **[次元解析]** を選択し、**[次元のチェック]** をクリックします。結果は、以下のような 2 つの方程式となります。

$$cin(t) = c(t) \frac{\text{kg}}{\text{s}^2}$$

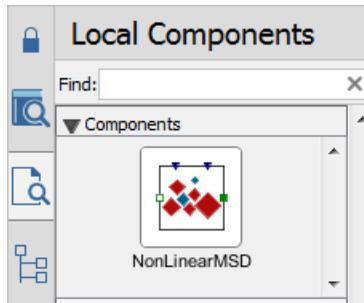
$$din(t) = d(t) \frac{\text{kg}}{\text{s}}$$

この矛盾には特に問題はなく、実数の入力信号 $cin(t)$ および $din(t)$ は実際には単位が適用されていることを示しています。

カスタムコンポーネントの生成

カスタムコンポーネントを生成し、[ローカルコンポーネント] タブの [コンポーネント] パレットに追加します。

1. [コンポーネントの生成] セクションで、[名前] テキストボックスに「NonLinearMSD」と入力します。
2. [MapleSim コンポーネントを生成] をクリックします。生成されたカスタムコンポーネントは、MapleSim 内の [ローカルコンポーネント] タブ (🔍) の [コンポーネント] パレットに表示されます。



ほかのコンポーネントと同様に、モデルワークスペースにドラッグすることで、カスタムコンポーネントをモデルに追加することができます。

3. MapleSim モデルを「NonlinearSpringDamper.msim」として保存します。第6章のチュートリアル3: 非線形ダンパのモデリング [194ページ]はこのカスタムコンポーネントをモデル内で使用しています。

第4章 モデルのシミュレーションと可視化

この章の内容は以下のとおりです。

- *MapleSim* によるモデルのシミュレーションのしくみ [105ページ]
- モデルのシミュレーション [108ページ]
- シミュレーションの進捗状況メッセージ [117ページ]
- シミュレーション結果とスナップショットの管理 [119ページ]
- プロットウィンドウ設定のカスタマイズ [121ページ]
- マルチボディモデルの可視化 [128ページ]
- ベストプラクティス: モデルのシミュレーションと可視化 [153ページ]

4.1. MapleSim によるモデルのシミュレーションのしくみ

Modelica 記述

MapleSim ライブラリの多くのコンポーネントの方程式は、Modelica 物理モデリング言語で記述されています。一方、マルチボディコンポーネントの方程式は、専用エンジンで生成した後、Modelica に変換しています。専用エンジンは、高度な数学的手法を用いて、可能な限り簡潔かつ効率的であるように方程式を生成します。

Modelica の詳細は、<http://www.modelica.org> を参照してください。

モデルの記述

モデルの各コンポーネントには、コンポーネントの動作を表す方程式系が含まれます。これらの方程式系には、純粋な代数方程式または微分方程式しか含めることができません。また、コンポーネントには、連立方程式の一部を有効または無効にしたり、状態値を変更するなどして、シミュレーション中にコンポーネントの動作を変えることができるイベントをいくつも定義することもできます。さらに、2つ以上のコンポーネントを接続すると、その相互関係を表す方程式が追加で生成されます。

システム方程式

コンポーネントのポート間の方程式と、トポロジ(コンポーネントの接続)で決まる方程式が収集され、1つの大きな連立方程式にされます。この段階で、パラメータ値も代入されます。ここで、MapleSim シミュレーションエンジンは、微分代数方程式が混在した大きな連立方程式をかかえているかもしれません。これは、大きな連立方程式に、離散イベントや代数制約を含む微分方程式があることを意味します。

方程式の簡略化

「*indexreduction*(低インデックス化)」と呼ばれるプロセスを使用すると、代数制約は可能な限り減らされます。この段階の中心となるのは、方程式と変数の数を削減するために、数式処理の簡素化テクニックを使用して修正された1次の連立微分代数方程式(DAE)を作成するアルゴリズムです。これらのテクニックの多くは混在した連立方程式を処理します。

変数には、MapleSim ウィンドウ右側の [**プロパティ**] タブで特定コンポーネントにパラメータ値を指定することによってその初期値を設定できるものがあります。指定されている初期状態に矛盾がある場合は、シミュレーション中にエラーが発生します。この問題を解決するには、初期化診断 App を使用します。

積分とイベント処理

前処理のステップがすべて終了すると、積分とイベント処理プロセスが始まります。選択したソルバタイプに基づいて、高度な DAE ソルバが連立方程式の数値積分を行います。可変ソルバの場合、解の精度に影響を与える制約のドリフトを避けるために、代数制約は常にモニターされています。固定ソルバタイプの場合、代数制約は固定された時間ステップごとにモニターされます。

積分の実行中、モデルの一部である不等条件がモニターされ、これらの条件のうち、1つ以上が変わるとイベントが発生します。そのようなイベントが発生すると、数値ソルバは停止し、シミュレーションエンジンはイベント条件に基づいて、連立方程式の新しい構成を計算します。このステップでは、連立方程式の新しい構成における初期条件の再計算も行われます。その後、ソルバは再始動し、別のイベントが発生するか、シミュレーション時間が終了するまで連立方程式の数値解を求め続けます。

注: 可変ステップソルバと固定ステップソルバの両方でイベントが発生します。固定ステップソルバの場合、イベントは固定時間ステップの実行時のみに処理されるのに対して、可変ソルバの場合は、ソルバが時間ステップを調整し、積分の実行中にイベントが発生する正確な時刻にイベントが処理される点が異なります。

シミュレーション結果

結果はシミュレーションプロセスの最後のステップで生成され、重要な物理量を描いたグラフで表示されます。また、マルチボディシステムの場合は、そのオプションが指定されていると、3-D 動画が表示されます。

シミュレーションプロセスの概要は、次のチャートのとおりです。

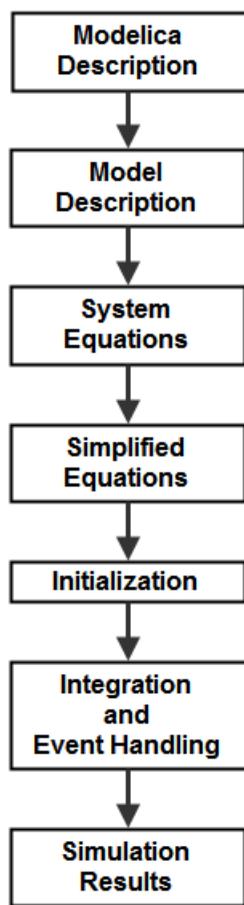


図4.1 シミュレーションプロセス

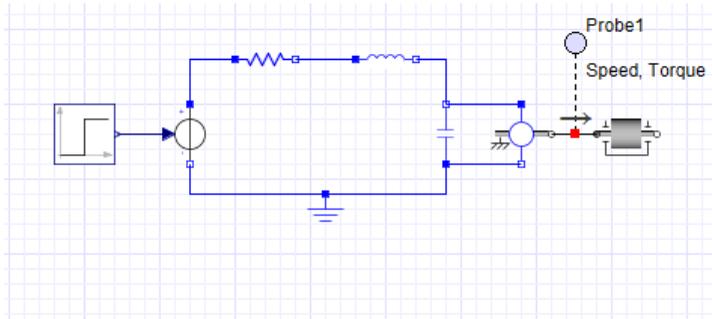
このセクションの情報は、シミュレーションプロセスの簡単な説明です。シミュレーションエンジンで使用される DAE ソルバの詳細については、Maple ヘルプシステムの `dsolve/numeric` ヘルプトピックを参照してください。

4.2. モデルのシミュレーション

2-D または 3-D のモデル内の結線やポート、コンポーネントなどにプローブを追加すると、電流、電圧などの物理的特性がどのように反応または応答するか確認

することができます。MapleSim では、プローブで接続ポートに関連付けられている確認対象の変数を識別します。

通過変数を計測するプローブを追加した場合、**モデルワークスペース**には流れの正方向を示す矢印が表示されます。



ユーザは、シミュレーション時間や使用するソルバの種類のほか、ソルバやシミュレーションエンジン、**3-D ワークスペース**のその他のパラメータ値などを指定することができます。シミュレーションの実行後、デフォルトでは、指定された物理量ごとにグラフが表示されます。

その後、最初のプローブまたはパラメータ値を変更し、別のシミュレーションを実行して結果を比較することができます。

符号の規則の例および矢印の方向がモデル上で作用する力をどのように表すかについては、**[ヘルプ]メニュー > [例題] > [ユーザガイドの例題] > [第4章]** から **[Constant Acceleration]**、**[Sign Convention]**、または **[Arrow Convention]** のいずれかの例題を参照してください。

シミュレーションとシミュレーションオプションの設定

シミュレーションに使用するパラメータは、パラメータペインの**[シミュレーションの設定]** タブにあります。このタブには、**[Simulation]** および **[Advanced Simulation]** 設定のセクションがあります。

シミュレーションの各種設定にアクセスするには、パラメータペインの右側の**[シミュレーションの設定]** タブ (🔧) をクリックします。

マルチボディと3-Dアニメーション設定の説明については、3-Dアニメーションとマルチボディの設定 [129ページ] を参照してください。

シミュレーションの設定

[Simulation] セクションでは、シミュレーション時間、プロット点の数、ソルバ、ソルバのパラメータを指定できます。[Simulation] 設定セクションで使用可能なパラメータの一覧と説明については、表4.1「シミュレーションの設定」を参照してください。

シミュレーションの設定

パラメータ	デフォルト	説明
t_d	10s	シミュレーションの実行時間。浮動小数値を含む、あらゆる正値を指定することができます。 注: 実行時間はシミュレーションの終了時間とは異なります。シミュレーションの終了時間は $t_d + t_s$ によって得られます。ここで t_s はシミュレーションの開始時間を表します(表4.2「シミュレーションオプション設定」を参照)。
Solver Type	Variable	シミュレーションに使用するソルバの種類。 <ul style="list-style-type: none"> • Variable: 可変時間ステップを使用して許容誤差を管理します。 • Fixed: 固定時間ステップを使用し、積分誤差を無視します。 注: 固定ステップソルバは MapleSim がエクスポートしたコードによって使用される固定ステップソルバと同じものです。

パラメータ	デフォルト	説明
Solver	Variable : CK45 (semi-stiff) Fixed : Euler	<p>シミュレーションに使用する DAE ソルバを指定します。[Solver Type] を [Variable] にした場合は、以下の選択肢を使用できます。</p> <ul style="list-style-type: none"> • CK45 (semi-stiff) : semi-stiff DAE ソルバ (ck45 法) を使用する • RKF45 (non-stiff) : non-stiff DAE ソルバ (rkf45 法) を使用する • Rosenbrock (stiff) : stiff DAE ソルバ (Rosenbrock 法) を使用する <p>複雑なモデルの場合は、モデルのシミュレーションに必要な時間を短縮するために、stiff DAE ソルバの使用が推奨されます。</p> <p>[Solver Type] を [Fixed] にした場合は、以下の選択肢を使用できます。</p> <ul style="list-style-type: none"> • Euler : Forward Euler ソルバを使用する • Implicit Euler : Implicit Euler ソルバを使用する (stiff システム用) • RK2 : 2 次ルンゲクッタソルバを使用する • RK3 : 3 次ルンゲクッタソルバを使用する • RK4 : 4 次ルンゲクッタソルバを使用する
ϵ_{abs}	$1 \cdot 10^{-5}$	<p>可変ステップソルバを使用してシミュレーションを実行する際に、積分ステップを成功と判断するための絶対許容誤差の限度を指定します。このオプションには浮動小数値を指定できます。</p>
ϵ_{rel}	$1 \cdot 10^{-5}$	<p>可変ステップソルバを使用してシミュレーションを実行する際に、積分ステップを成功と判断するための相対許容誤差の限度を指定します。このオプションには浮動小数値を指定できます。</p>

パラメータ	デフォルト	説明
Step size	0.0010	固定ステップソルバを使用してシミュレーションを実行する際の、固定のサンプリング周期の間隔を指定します。このオプションには浮動小数値を指定できます。
Plot Points	200	<p>シミュレーションで描画する点数の最小値を指定します。データ点は、シミュレーション時間に応じて、均等に分布されます。正の整数を指定することができます。イベントにさらに多くの点を追加できます(表4.2「シミュレーションオプション設定」のプロットイベントを参照)。</p> <p>シミュレーションエンジンでは、以下のうち最大の [Plot points] の値を使用します。</p> <ul style="list-style-type: none"> • [Plot Points] の設定、または • [3D Sample Rate] (fps) * シミュレーション時間 ([3-D Animation] が有効化されている場合)、または • [3D Sample Rate] (fps) * [3-D Playback Time] ([3-D Animation] が有効化されていて、[3-D Playback Time] が指定されている場合) <p>注: このオプションで指定する点数は、表示目的のみに使用されます。シミュレーションに使用された実際の点数は、シミュレーショングラフに表示される点数と異なる場合があります。</p>

シミュレーションオプション設定

[Advanced Simulation] セクションでは、シミュレーションの開始時間、使用するスナップショット、コンパイルオプション、およびその他の設定を指定できます。これらの設定には、シミュレーションの設定で選択したソルバの種類 (Variable または Fixed) に固有の設定もあります。[Advanced Simulation] 設定で使用可能なパラメータの一覧と説明については、表4.2「シミュレーションオプション設定」を参照してください。

シミュレーションオプション設定

パラメータ	デフォルト	Solver Type	説明
t_s	0	すべて	シミュレーションの開始時間。負の値を含む、あらゆる浮動小数値を指定できます。 注: シミュレーションの開始時間はシミュレーションの終了時間に影響しますが、シミュレーションの実行時間 t_d には影響しません。シミュレーションの終了時間は、 $t_d + t_s$ によって求められます。
Use Snapshot	None	すべて	スナップショットは、特定の時刻におけるシミュレーションの状態を取得します。シミュレーションでスナップショットを使用する場合は、モデルで使用する初期条件を上書きし、そのモデルのスナップショットの時刻の状態に置き換えることができます。 スナップショットの詳細は、シミュレーション結果とスナップショットの管理[119ページ]を参照してください。
Jacobian	Symbolic	すべて	システムのヤコビアンに対して数式処理または数値近似を選択します。数式処理による定式化を使用するとより高速で正確なシミュレーションを実行できますが、定式化するのに時間がかかる場合があります。 注: 数値近似定式は stiff ソルバ (Rosenbrock 法または Implicit Euler 法) のみで使用できます。
Baumgarte	<input type="checkbox"/>	すべて	モデルに Baumgarte の拘束安定化を適用します。これを選択して、モデルに適した微分ゲイン (α) および比例ゲイン (β) の値を入力します。

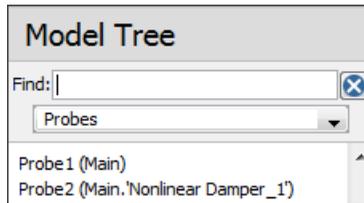
パラメータ	デフォルト	Solver Type	説明
Projection	<input checked="" type="checkbox"/>	すべて	モデルに拘束条件に対する射影法を適用します。シミュレーションの各ステップで得られた解を拘束の多様体に対して射影する場合に選択します。射影は、[Projection Iterations] の最大数に到達した場合、またはかい離が [Projection Tolerance] の値を下回った場合に終了します。
Projection Iterations	50	すべて	拘束条件に対する射影法における最大収束計算回数。 注: このパラメータは [Projection] が選択されている場合にのみ使用できます。
Projection Tolerance	0.000010	Fixed	射影法の収束計算が終了する許容値。あらゆる正の浮動小数値を指定できます。
Event Projection	<input checked="" type="checkbox"/>	Fixed	イベント処理中に拘束条件に対する射影法を実行する場合に選択します。ただし、積分の結果を得るのに時間がかかります。このオプションを選択しない場合、次のステップで、イベントが拘束条件に対する射影法を適用できない点に解を変更すると、シミュレーションが失敗する場合があります。
Event Iterations	100	すべて	積分器がエラーを返す前に実行できるイベント処理の最大収束計算回数。正の整数値を指定できます。
Event Hysteresis	$1.0 \cdot 10^{-7}$	Fixed	イベントのヒステリシス幅。0 以上の浮動小数値を指定できます。0 を設定すると、このパラメータが無効になります。
Initial Hysteresis	$1.0 \cdot 10^{-10}$	Variable	シミュレーションの開始時点で発生するすべてのイベントのイベントヒステリシス幅。0 以上の浮動小数値を指定できます。
Index1 Error Control	<input type="checkbox"/>	Variable	これを選択して、すべての代数変数に誤差制御を適用します。デフォルトでは、誤差制御はイベントをトリガする代数変数、プロットされた代数変数または関数の出力である代数変数にのみ適用されます。

パラメータ	デフォルト	Solver Type	説明
Index1 Tolerance	1.0	Variable	微分変数と比較した代数変数の相対誤差を制御します。たとえば 10 という値は、代数変数が微分変数の誤差の 10 倍である可能性があることを意味します。
Minimum Step Size	0	Variable	最小ステップサイズを設定します。
Maximum Step Size	値なし	Variable	最大ステップサイズを設定します。
Scaling	None	Variable	システムに適用する変数のスケールリング方法を指定します。使用可能な選択肢は以下のとおりです。 <ul style="list-style-type: none"> • None : スケールリングを適用しない • Minimum : 公称の最小値を使用する • Maximum : 公称の最大値を使用する • Geometric : 公称値の幾何平均を使用する
Minimize Events	<input type="checkbox"/>	Variable	このオプションは、シミュレーションの実行中にヒューリスティックスを使用し、発生するイベント数を減らすかどうかを指定します。選択した場合、区分遷移のイベントへのマッピングは行われません。
Plot Events	<input checked="" type="checkbox"/>	すべて	シミュレーションの実行中のイベントポイントに、追加のプロット点を含めるかどうかを指定します。

パラメータ	デフォルト	Solver Type	説明
Solver Diagnostics	<input type="checkbox"/>	すべて	<p>これを選択した場合、シミュレーションで拘束条件処理の収束計算回数、拘束条件の残差、イベント処理、およびステップサイズの診断を行います。結果は、診断終了後に[シミュレーション結果]タブの[ソルバ診断]プロットに表示されます。投影オプションをクリアにしているシステムでは、追加の計算コストが発生します。</p> <p>矛盾した連立方程式またはランタイム問題を持つモデルの場合、このオプションを選択して、エラーを発生させている変数、方程式、およびコンポーネントの詳細を表示します。</p>
Compiler	<input checked="" type="checkbox"/>	Variable	<p>シミュレーションにネイティブCコンパイラを使用するかどうかを指定します。このオプションが選択されている場合、シミュレーションエンジンによって生成されるMapleプロシージャはCコードに変換され、外部のCコンパイラによってコンパイルされます。</p> <p>複雑なモデルの場合は、シミュレーションの実行に必要な時間を短縮するために、このオプションをオンにすることをお勧めします。</p>
Compile Optimized	<input checked="" type="checkbox"/>	すべて	<p>コンパイル中にコードを最適化します。このパラメータがオフになっていると、コンパイル時間は短縮されますが、シミュレーションの実行には時間がかかります。</p>

プローブ値の編集

[モデルツリー] タブ () をクリックし、ドロップダウンリストから [プローブ] を選択します。現在の MapleSim モデルに追加されているすべてのプローブが、ここに一覧表示されます。



プローブがモデルのトップレベルに追加されている場合、プローブ名の横には括弧で **[Main]** と表示され、それ以外の場合は、プローブのサブシステム名がプローブ名の横に表示されます。上記例の場合、モデルには、モデルのトップレベルに **[Probe1]**、そして **[Main.Nonlinear Damper_1]** という名前のサブシステム（つまり、**[Main]** にある **[Nonlinear Damper_1]** サブシステム）に **[Probe2]** の2つのプローブが追加されています。

このパレットの項目をクリックすると **モデルワークスペース** でそのプローブに移動し、**[プロパティ]** タブ (📄) からプローブ値を閲覧、編集することができます。また、パレットの項目を右クリック (Macでは **[Control]** キーを押しながらクリック) し、コンテキストメニューを用いてプローブを操作することもできます。

詳細は、MapleSim ヘルプシステムの **MapleSim 操作方法 > モデルのシミュレーション > プローブの使用 > プローブ値の編集** を参照してください。

シミュレーション結果比較のためのパラメータセットの保存

モデルに代入されているパラメータ値の組み合わせをパラメータセットに保存することができます。その結果、あるパラメータセットを使用してシミュレーションを実行し、その後に別のパラメータセットでパラメータ値を置き換えて、シミュレーションを再実行し、結果を比較することができます。

詳細は、MapleSim ヘルプシステムの **MapleSim 操作方法 > モデルの構築 > パラメータセットの利用 > パラメータセットの保存と適用** セクションを参照してください。

4.3. シミュレーションの進捗状況メッセージ

シミュレーション中、進捗状況は **モデルワークスペース** の下にある **コンソール** ペインに表示されます。これらのメッセージは MapleSim エンジンが数理モデルを

生成しているときの状態を示し、シミュレーションエラーのデバッグに役立ちます。シミュレーションの各ステップに対するメッセージが、固有のセクション(例: 方程式の生成や初期値の計算)に表示されます。新しいステップが開始すると、前のセクションは自動的に折り畳まれます。セクションに対するメッセージを確認するには、そのセクションのグレーの矢印をクリックしてセクションを展開します。または、以下のように矢印キーを使用して、コンソールペインを操作します。

- 右矢印キーを押してセクションを展開する
- 左矢印キーを押してセクションを折り畳む
- 下矢印キーを押して次のセクションに移動する
- 上矢印キーを押して前のセクションに移動する

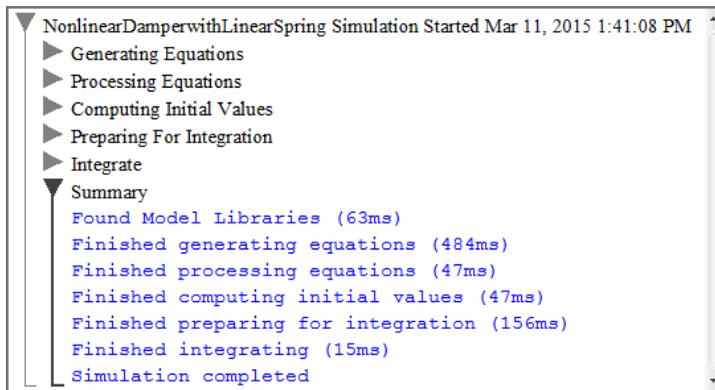
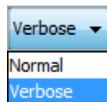


図4.2 シミュレーション結果の進捗状況メッセージ

オプションとして、シミュレーションを実行する前に、MapleSim ウィンドウの下部にある [コンソール出力] () をクリックし、ドロップダウンメニューでレベル ([標準] または [詳細]) を選択することで、進捗状況メッセージに表示する情報の詳細度を指定することができます。



コンソールのメッセージを消去するには、**メッセージコンソールの消去** () をクリックします。

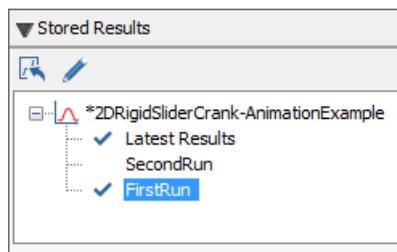
4.4. シミュレーション結果とスナップショットの管理

[シミュレーション結果] タブの [保存結果] パレットを使用すると、複数のシミュレーション結果を表示、保存、エクスポートすることができます。また、それぞれの保存結果に関して、シミュレーション中の特定の時刻のモデルの状態変数をすべて記録したスナップショットを保存およびエクスポートできます。

結果の保存

モデルのシミュレーションを実行するたびに、最新の結果が [Latest Results] として [保存結果] パレットに保存されます。この項目には、シミュレーションに関連するすべてのグラフ、進捗状況メッセージ、3-D 動画 (該当する場合) が含まれています。ただし、[Latest Results] の項目に保存されている結果は、新しくシミュレーションを実行するたびに上書きされます。

新しい名前でシミュレーション結果を保存するには、[Latest Results] の入力内容を右クリック (Mac では [Control] キーを押しながらクリック) し、コンテキストメニューから [名前を付けて保存] を選択して、保存する結果の名前を入力します。MapleSim の現在のセッションにおいて複数のシミュレーションで生成されたグラフを比較したり、参照したりすることができます。



結果を保存したあとにモデルを保存しておく、今後の MapleSim セッションでモデルを開いたときに [保存結果] パレットから保存した結果を利用できます。ただし、モデルを閉じて再び開いた場合、[Latest Results] は保存されません。

スナップショットの保存と使用

特定の時刻にシミュレーションのスナップショットを取得し、スナップショットを保存結果の一部として保存することで、状態情報を保存できます。スナップショットを保存しておくことで、状態情報を今後のシミュレーションで利用できるようになります。新しいスナップショットの作成や次のシミュレーションで使用するスナップショットの選択など、スナップショットを管理するには、シミュレーションの **[Advanced Simulation]** 設定 (表4.2 「シミュレーションオプション設定」) を参照してください。

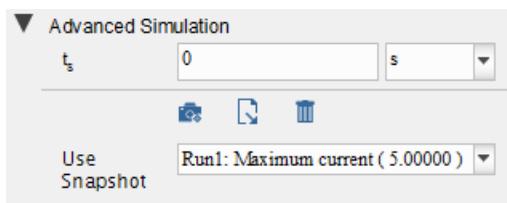


図4.3 シミュレーションオプション設定のスナップショット

この方法でスナップショットを使用すると、スナップショットに記録された情報から、今後のシミュレーションの初期値を設定します。

シミュレーションで使用するスナップショットが選択されない場合、シミュレーションの実行アイコンはツールバー内で  と表示されます。シミュレーションで使用するスナップショットが選択された場合、シミュレーションの実行アイコンは  に表示が変わります。シミュレーションでスナップショットが使用された場合、**[保存結果]** パレットに表示される説明から設定情報を確認できます。

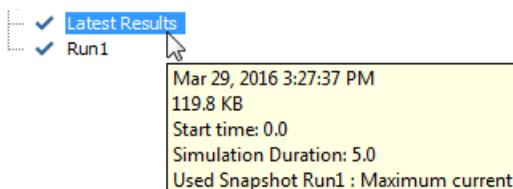


図4.4 保存結果パレットとスナップショット

詳細は、MapleSim ヘルプシステムの **MapleSim 操作方法 > モデルのシミュレーション > シミュレーション結果の管理 > シミュレーションのスナップショットの管理**セクションを参照してください。

4.5. プロットウィンドウ設定のカスタマイズ

デフォルトでは、プローブの物理量は**プローブプロット**と呼ばれるプロットウィンドウに個別にプロットされます。各グラフで、*y*軸は物理量の値を表し、*x*軸はシミュレーション時間の値を表します。

オプションとして、カスタムプロットウィンドウ設定を作成できます。カスタムプロットウィンドウ設定は、たとえば、1つのグラフで複数の物理量を比較したり、ある物理量に対して別の物理量をプロット、またはほかのプローブの値をそのままにして特定の物理量に対するシミュレーショングラフを表示するときなどに使用できます。2つの異なるモデルから物理量を比較することも可能です。プロットのタイトルをカスタマイズしたり、[プロット]ウィンドウに表示する列数などを指定したりして、[プロット]ウィンドウをさらにカスタマイズすることができます。

プロットウィンドウの作成についての詳細は、**新しいプロットウィンドウ設定の生成**を参照してください。[プロット]ウィンドウに関する詳細は、MapleSim ヘルプシステムの **MapleSim 操作方法 > モデルのシミュレーション > プロットウィンドウ設定の操作**セクションを参照してください。

新しいプロットウィンドウを作成すると、シミュレーションを再度実行せずにデータの入力が行われます。

ヒント: モデルと保存結果を保存して再び開くと、プロットは最初に「No data available」のメッセージを表示する場合があります。プロットを読み込むには、**[保存結果]**パレットで項目を右クリックし、**[プローブプロットを表示]**を選択します。

シミュレーションデータを別のアプリケーションで使用する場合は、結果を Microsoft Excel (.xls) またはカンマ区切り (.csv) ファイルにエクスポートします。詳細については、**シミュレーショングラフデータのエクスポート**を参照してください。

以下の例で、カスタムプロットウィンドウ設定を作成します。

例：複数の物理量を個別のグラフにプロットする

この例では、**プローブプロット**を表示し、2つ目の変数をそれぞれのカスタムプロットに追加するプロットウィンドウを作成します。

プローブプロットを表示するには

1. [ヘルプ] メニューの [例題] > [ドメイン別例題] > [マルチボディ] から [Double Pendulum] の例題を選択します。
2. メインツールバーでシミュレーションの実行 () をクリックします。
3. シミュレーション結果を表示 () をクリックします。[シミュレーション結果] タブが選択された状態で [解析ウィンドウ] が開きます。[3-D アニメーションウィンドウ] と [プローブプロット] が [シミュレーション結果] タブに表示されます。図4.5「シミュレーショングラフ」を参照してください。

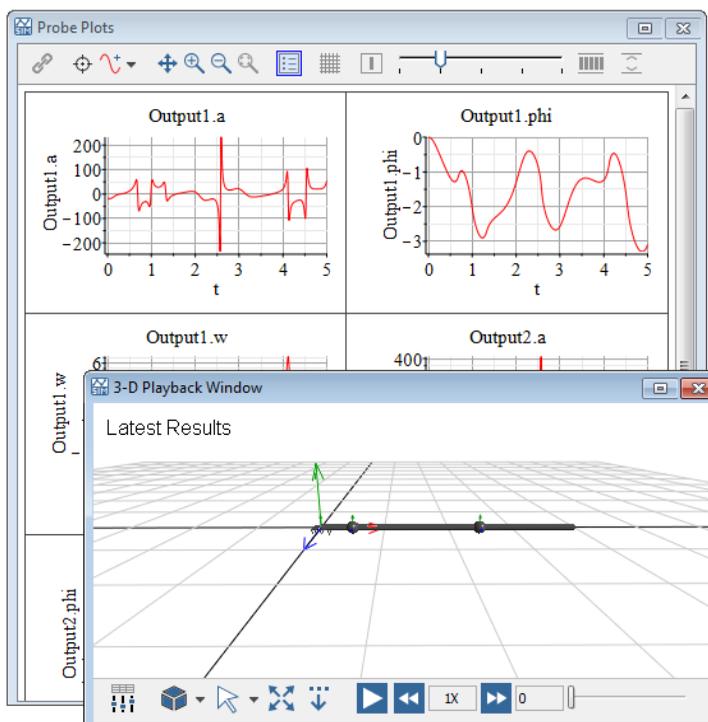


図4.5 シミュレーショングラフ

カスタムプロットウィンドウ設定を作成するには

1. [プロットウィンドウ] パレットで、Output1.w、Output2.a、Output2.phi、Output2.wの各プロットをダブルクリックして非表示にします。これによりプローブプロットは **[Output1.a]** および **[Output1.phi]** の2つのプロットだけの表示になります。
2. 選択したプロットウィンドウの複製ボタン (📄) をクリックします。
3. [プロットウィンドウを複製] ダイアログボックスで、名前「**Acceleration and Angle Comparison**」を入力します。[OK] をクリックします。現在表示可能なプロットを含んだ新規プロットウィンドウが、「Acceleration and Angle Comparison」の名前で作成されます。
ヒント: プロブプロットを元の表示に戻すには、[プロットウィンドウ] パレットのプローブプロットを右クリックし、[全プロットの表示] を選択します。

4. [プロットウィンドウ] で、プロット [**Output1.a**] を [**Acceleration and Angle Comparison**] プロットウィンドウから選択します。
5. [**変数**] パレットの **Output2** で、**[a]** を選択します。
6. 選択したプロットに選択した変数を追加 (📌) をクリックします。プロットが Output1.a および Output2.a の2つの曲線を表示します。
7. [プロットウィンドウ] で、プロット [**Output1.phi**] を [**Acceleration and Angle Comparison**] プロットウィンドウから選択します。
8. [**変数**] パレットの [**Output2**] で、**[phi]** を選択し、[**選択したプロットに選択した変数を追加**] (📌) をクリックします (または、**[phi]** をプロット上にドラッグします。) プロットが両方の曲線を表示します。
9. [プロットウィンドウ] パレットの **Acceleration and Angle Comparison** で、**[Output1.a]** (1番目のプロット名) を右クリックして [**名前の変更**] を選択します。
10. 「**Acceleration (a)**」と入力して名前を付けます。これでプロットのタイトルが変更になります。
11. [プロットウィンドウ] パレットの **Acceleration and Angle Comparison** で、**[Output1.phi]** (2番目のプロット名) を右クリックして [**名前の変更**] を選択します。
12. 「**Angle (phi)**」と入力して名前を付けます。

作成したプロットを 図4.6 「カスタムプロットウィンドウ」 に示します。

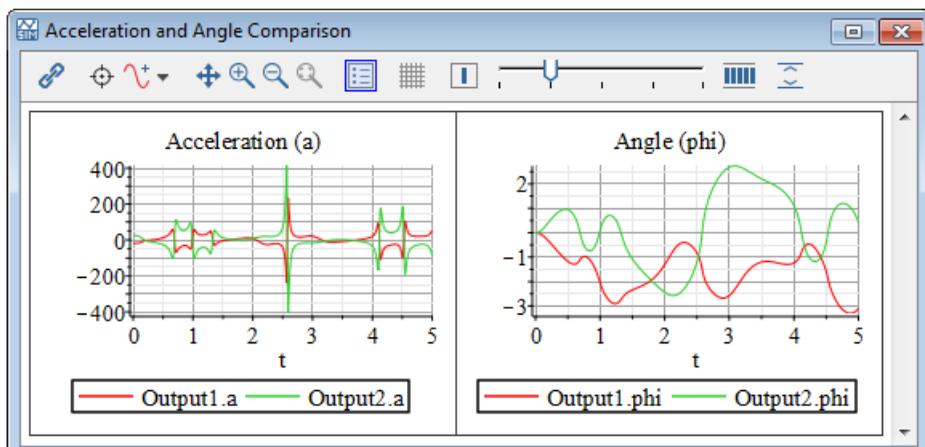


図4.6 カスタムプロットウィンドウ

例：1つの物理量を別の物理量に対してプロットする

この例では、二重振り子の各リンクのXとY位置をプロットするためのカスタムプロットウィンドウ設定を作成します。

1つの物理量をカスタムプロットウィンドウの別の物理量に対してプロットするには

1. [ヘルプ] メニューの [例題] > [ドメイン別例題] > [マルチボディ] から [Double Pendulum] の例題を選択します。
2. モデルワークスペースツールバーのプローブを追加ボタン (+) をクリックします。
3. L_1 共有サブシステムの右ポートをクリックします。
4. モデルワークスペースの空白の領域をクリックし、プローブを配置します。
5. [プロパティ] タブ (≡) でこのプローブに「FirstLink」というラベルを付け、[Length[1]]と[Length[2]]を選択します。
6. L_2 共有サブシステムの右ポートに、Length[1]とLength[2]の物理量を計測するプローブをもう1つ追加し、このプローブに「SecondLink」というラベルを付けます。

7. **メインツールバーでシミュレーションの実行** () をクリックします。
8. **シミュレーション結果を表示** () をクリックします。[解析] ウィンドウが開き、[シミュレーション結果] タブにプローブプロットが表示されます。
9. ここでは2つのプロット **FirstLink.r_0[2]** と **SecondLink.r_0[2]** のみをプロットウィンドウに表示させる場合を考えます。これが新規プロットウィンドウを作成するベースとなります。[プロットウィンドウ] パレットで、その他のプロット名をダブルクリックして非表示にします。
10. **選択したプロットウィンドウの複製** () をクリックします。
11. [プロットウィンドウを複製] ダイアログボックスで、名前「**X versus Y**」を入力します。[OK] をクリックします。現在表示可能なプロットを含んだ新規プロットウィンドウが、「X versus Y」の名前で作成されます。
ヒント: プローブプロットを元の表示に戻すには、[プロットウィンドウ] パレットの**プローブプロット**を右クリックし、[全プロットの表示] を選択します。
12. [プロットウィンドウ] パレットの **X versus Y** で、[**FirstLink.r_0[2]**] (1番目のプロット名) を右クリックして [**名前の変更**] を選択します。
13. 「**TopLink**」と入力して名前を付けます。これでプロットのタイトルが変更になります。
14. 同様に、プロット名 [**SecondLink.r_0[2]**] を「**BottomLink**」に変更します。
15. [プロットウィンドウ] で、プロット [**Top Link**] を [**X versus Y**] プロットウィンドウから選択します。
16. [**変数**] パレットで、[**FirstLink: r_0[1]**] を選択します。
17. **選択したプロットの X 軸に選択した変数を適用** () をクリックします。
18. [プロットウィンドウ] で、プロット [**Bottom Link**] を [**X versus Y**] プロットウィンドウから選択します。
19. [**変数**] パレットで、[**SecondLink: r_0[1]**] を選択します。
20. **選択したプロットの X 軸に選択した変数を適用** () をクリックします。
21. **プロットツールバー**で、スライダを使用してプロットの列数を1に変更します。

作成したプロットを図4.7「1つの物理量を別の物理量に対してプロットする」に示します。

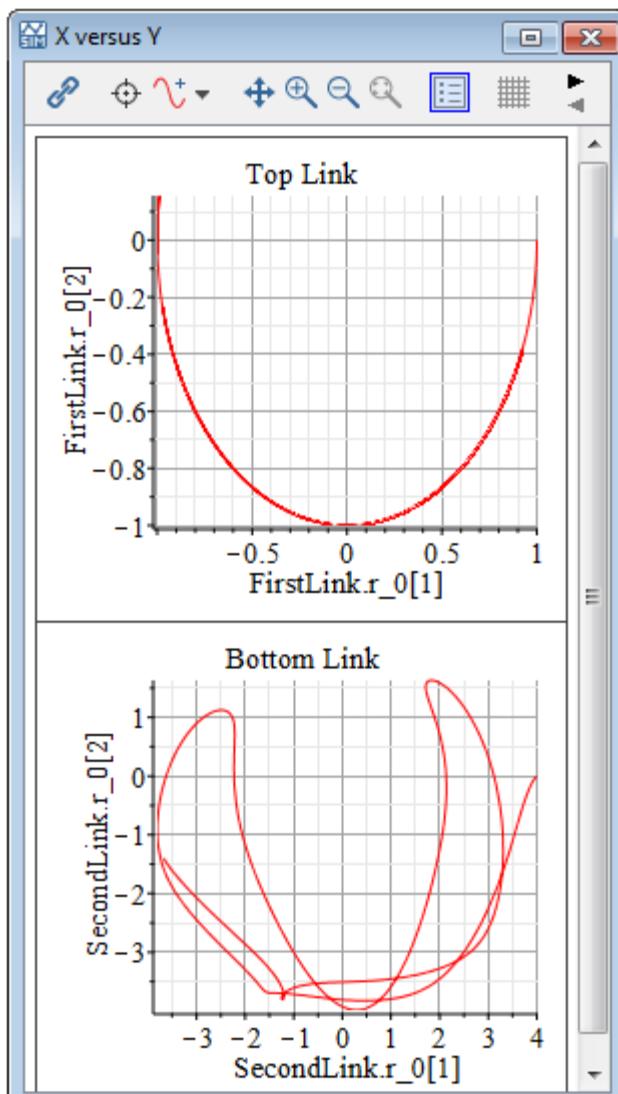


図4.7 1つの物理量を別の物理量に対してプロットする

上記のプロットは、それぞれ振り子の各リンクの端点の運動を表しています。下部リンクは、上部リンクとの相互作用により、より複雑な動きを示しています。

4.6. マルチボディモデルの可視化

MapleSim では、3-D アニメーション環境を使用して、マルチボディシステムの 3-D 図を作成したり、解析することができます。モデルの作成過程では、パラメータを変更し、そのモデルの 3-D 構成を検証したり、シミュレーション結果を視覚的に解析することができます。3-D モデルは、**3-D ワークスペース**にオブジェクトをドラッグし、接続して作成し、シミュレーション結果は、オブジェクトの動きを描写する動画再生で可視化することができます。

モデルワークスペースでブロックダイアグラムを作成すると、そこで加えられた変更は自動的に **3-D ワークスペース**の 3-D 図に反映されます。同様に、**3-D ワークスペース**でモデルを作成すると、そこで加えられた変更は**モデルワークスペース**のブロックダイアグラムに自動的に反映されます。変更は、どちらのワークスペースで加えられても、**モデルワークスペース**、**3-D ワークスペース**の両方に反映されます。

3-D ワークスペースでは、モデルをあらゆる方向から表示できます。さらに、モデルの表現をより現実的なものにするために、モデルの部品に 3-D 図形を結びつけることもできます。これらの図形は、外部の CAD ファイルからインポートすることも、**[ライブラリコンポーネント]** タブの **[マルチボディ]** > **[可視化]** パレットから選択することができます。

モデルのシミュレーション後、**[3-D アニメーションウィンドウ]** で 3-D モデルの動画を再生することができます。再生オプションを指定することにより、特定のコンポーネントとその動きに焦点を当てることもできます。たとえばアニメーションの実行中に、カメラの追跡オプションを指定して**[3-D アニメーションウィンドウ]** の特定のオブジェクトに注意を向けることが可能です。また、トレース線を追加して、動画でコンポーネントが移動する軌跡を示すこともできます。

ヒント: **[3-D アニメーションウィンドウ]** の背後に開いているプロットウィンドウがある場合、可視化の質に影響します。アニメーションの再生で問題が発生する場合は、**[3-D アニメーションウィンドウ]** を移動してプロットウィンドウに重ならないようにしてください。あるいは、開いているプロットウィンドウを最小化するか閉じてください。

CAD ジオメトリと可視化された図形は、3-D ワークスペースでは透明に、**[3-D アニメーションウィンドウ]** では不透明に描画されます。

3-D 図形の追加と **3-D ワークスペース**の使用に関する詳細は、MapleSim ヘルプシステムの **MapleSim 操作方法 > 3-D モデルの表示**セクションを参照してください。

3-D アニメーションとマルチボディの設定

マルチボディコンポーネントの3-D アニメーションに使用するパラメータは、パラメータペインにある **[マルチボディ設定]** タブの **[Animation]**、**[Multibody]**、および **[Visualization]** にあります。

シミュレーションの各種設定にアクセスするには、パラメータペインの右側の **[マルチボディ設定]** タブ () をクリックします。

アニメーション設定

マルチボディコンポーネントを含むモデルでは、次のアニメーション設定を指定できます。

アニメーション設定

パラメータ	デフォルト	説明
3-D Playback Time	-	<p>3-D 動画の再生時間を秒単位で指定します。この値は、シミュレーション時間を指定する [シミュレーションの設定] タブの $[t_d]$ 値とは異なります。このオプションには浮動小数値を指定できます。</p> <p>このオプションには、動画の再生速度を速くまたは遅くするための値を指定します。たとえば、$[t_d]$ 値が 0.5 秒に設定されているときに [3-D Playback Time] に 10 秒を設定すると、0.5 秒のシミュレーション動画は、実時間で 10 秒かけて再生されるため、再生速度は遅くなります。</p> <p>このフィールドに値が指定されていない場合、[3-D Playback Time] の値は、$[t_d]$ の値と同じになり、たとえば、10 秒のシミュレーションの動画は、実時間で 10 秒かけて再生されます。アニメーションに描写されるフレームの数は [Plot Points] に指定されている値、および [3-D Playback Time] の値と [3-D Sampling Rate] の値の積で決まります。</p>
3-D Sampling Rate	40	<p>3-D 動画再生で表示する 1 秒ごとのフレーム数を指定します。この値を大きくすると、よりスムーズに再生されるアニメーションを作成できます。このオプションには、[3-D Animation] フィールドの値が true の場合に正の整数を指定することができます。</p> <p>新規モデルに対するデフォルト値は 40 fps です。</p>

マルチボディ設定

マルチボディコンポーネントを含むモデルでは、次のパラメータ値を指定することができます。

マルチボディのパラメータ値

パラメータ	デフォルト	説明
Gravity	9.81	地球の重力による、地表上での加速度を指定します。 デフォルトの単位は $\frac{m}{s^2}$ です。
Gravity vector	[0, -1, 0]	重力の方向を指定します。

可視化設定

マルチボディコンポーネントを含むモデルでは、次の3-Dアニメーションの値を指定できます。

3-Dアニメーションのパラメータ値

パラメータ	デフォルト	説明
Enable Translational Snapping	<input type="checkbox"/>	このオプションをオンにすると、コンポーネントは3-D空間において Translation Snap Delta に基づく最近接の位置に配置されます。
Translation Snap Delta	1.0	Translation Snap Delta の間隔を指定します。
Enable Rotational Snapping	<input type="checkbox"/>	このオプションをオンにすると、コンポーネントは3-D空間において Rotation Snap Delta に基づく最近接の位置に配置されます。
Rotation Snap Delta	0.785398163	Rotation Snap Delta の間隔を指定します。
Perspective Grid Extent	10.0	遠近表示で描画されるグリッドの範囲を指定します。グリッドは、水平面上の両方向に、この長さで示す範囲まで広がります。
Grid Spacing	1.0	グリッド線の間隔を指定します。
Base Radius	0.2	3-D ワークスペース でマルチボディコンポーネントを表現する球と円柱で構成されるインプリシットジオメトリ。円柱は Base Radius を使用して描画され、球(剛体とジョイント)は Base Radius * 2 を使用して描画されます。

パラメータ	デフォルト	説明
Enable View Change Animations	<input checked="" type="checkbox"/>	有効化すると、3-D 直交表示と遠近表示の切り替え、および3-Dワークスペースや[3-Dアニメーションウィンドウ]で[表示ウィンドウに合わせて拡大/縮小]、[選択範囲に合わせて拡大/縮小]または[アニメーションの全体範囲を表示]を使用する際の遷移がスムーズになります。

3-D ワークスペース

3-D ワークスペースは、MapleSim ウィンドウで3-D モデルの構築と表示をするためのエリアです。このワークスペースは、[解析] ウィンドウの [3-D ワークスペース] タブにあります。

3-D ワークスペースを開くには：

- [表示] メニューから、[3-D ワークスペースを表示] を選択します。[解析ウィンドウ]が開いて3-Dワークスペースが表示されます。別の方法として、MapleSim メインツールバーから、3-D ワークスペースを表示 (📐) をクリックします。

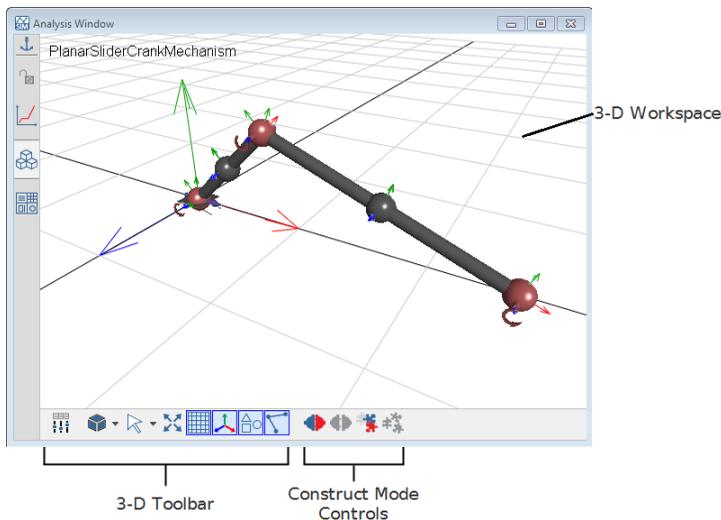
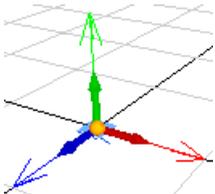


図4.8 3-D ワークスペース

3-D ワークスペースのコントロール

コンポーネント	説明
3-D ワークスペース	<p>3-D モデルの構築および表示を実行するエリア。原点の矢印は空間軸の方向を示し、以下の色で表示されます。</p> <div style="display: flex; align-items: center; justify-content: center;">  <div style="margin-left: 20px;"> <p>X - 赤</p> <p>Y - 緑</p> <p>Z - 青</p> </div> </div> <p>3-D モデル内要素の相対サイズと位置の計測には基準としてグリッドを使用することができます。</p>
3-D ツールバー	<p>3-D ワークスペース内にあるコンポーネントを表示、非表示にするツールや、さまざまなモードを切り替えるためのツール、カメラナビゲーションツールを選択するためのツール、3-D モデルの表示を変更するツールなどが配置されています。</p>
コンストラクトモードコントロール	<p>3-D ツールバーの一部で、3-D モデルの構築とアセンブル、および 3-D オブジェクトを接続するためのコントロールが配置されています。</p>

マウスポインタをツールバーのボタン上に移動させると、ボタンの説明が表示されます。

3-D モデルの表示と閲覧

3-D ワークスペースでは**3-D 表示のコントロール**ツールを使用して、遠近表示またはいずれかの直交表示で 3-D モデルを表示したり、閲覧することができます。



図4.9 3-D 表示コントロール

遠近表示を使用すると、3-D空間のあらゆる方向からモデルを観察したり、視点を移動することができるため、モデルに含まれる要素間の3-D空間的關係を確認することができます。遠近表示では、カメラにより近いオブジェクトは遠くのものより大きく見えます。

下は、二重振り子のモデルを透視法を使用して表示した場合のイメージです。

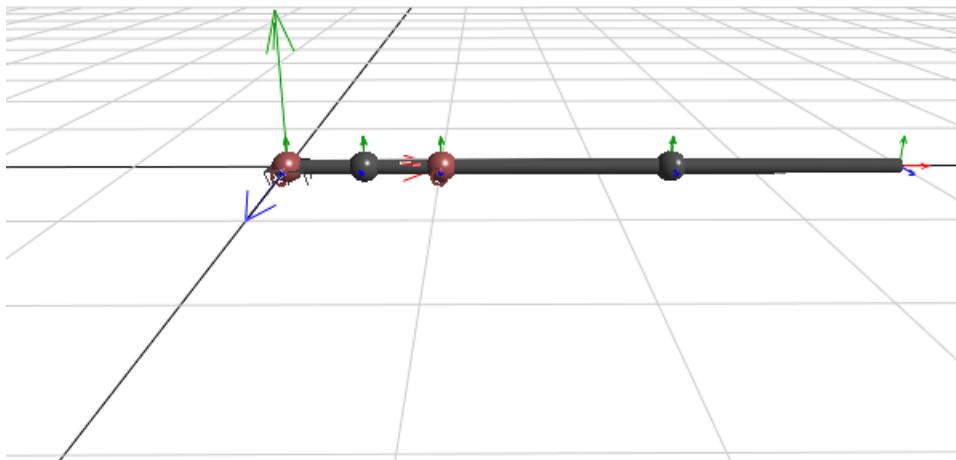


図4.10 二重振り子の遠近表示

また、直交表示では、3-Dモデルを前、上、および横から表示することができます。直交表示では、遠近表示と異なり平行射影が使用され深度情報がないため、3-Dモデルは平なオブジェクトとして表示されます。直交表示は、カメラに垂直な表示面において線や距離を忠実に表示するため、「実寸」表示と呼ばれる場合

もあります。これらの表示は、オブジェクト間の空間関係や隙間を解析するのに便利です。

下は、二重振り子のモデルを上からの正投影法を使用して表示した場合のイメージです。

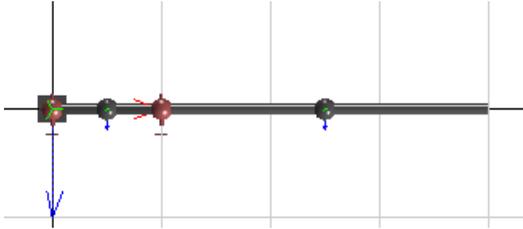


図4.11 二重振り子の直交表示

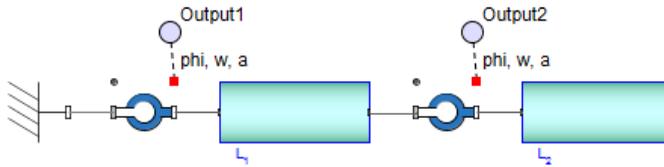
動画が静止表示中であっても再生中であってもモデルを閲覧したり、モデルの表示を変更することができます。また、どの表示でもパンしたり、拡大/縮小することができます。透視法ではさらに、カメラを移動させ、上から、下からと、あらゆる方向からモデルを表示することができます。

ヒント: カメラをパン、ズームしたり、大きな 3-D モデルのまわりを移動させる前に、焦点を当てたいオブジェクト上にマウスポインタを置いてください。MapleSim は、マウスポインタを置いたオブジェクトに応じてナビゲーション設定機能を調節します。

3-D モデルへの図形の追加

インプリシットジオメトリの追加

デフォルトでは、モデルの物理コンポーネントは「インプリシットジオメトリ」と呼ばれる基本的な球や円柱で 3-D ワークスペースに表示されます。たとえば、2つの回転ジョイントと平面リンクを表す 2つのサブシステムを含む二重振り子のモデルがあるとしたします。



3-D ワークスペースでは、完全にアSEMBLされた振り子のモデルにおけるインプリシットジオメトリは次のように表示されます。

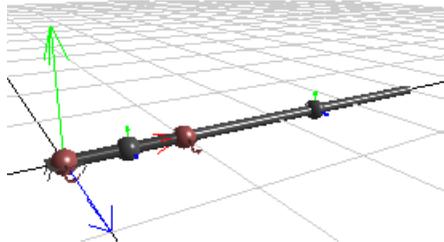


図4.12 二重振り子のインプリシットジオメトリ

この例では、球は回転ジョイントと剛体を表し、円柱は平面リンクを表しています。

インプリシットジオメトリで、別のインプリシットジオメトリに接続されていないものは薄い灰色、接続されているものは濃い灰色で描画され、例外としてジョイントオブジェクトは赤で描画されます。

注: モデルワークスペースで無効にされているコンポーネントは、**3-D ワークスペース**に表示されません。

添付形状の追加

モデルの表現をより現実的なものにするには、「添付形状」と呼ばれる図形や線をモデルに追加することができます。これらを追加するには、まず、**[マルチボディ]>[可視化]**パレットから添付形状コンポーネントを**モデルワークスペース**内のブロックダイアグラムに追加し、接続します。

モデルのシミュレーションを実行すると、**3-Dワークスペース**には、インプリシットジオメトリに加えて添付形状が表示されます。下の図では、振り子のロッドと

おもりを視覚的に表す目的で添付形状が追加されています。またトレース線(図に示されている曲線)は、シミュレーション中にモデルの特定部分によって描かれる軌跡を表しています。

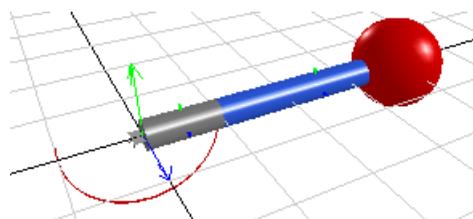


図4.13 添付形状

添付形状の色、サイズ、縮尺度、およびその他の視覚的側面は、シミュレーションを実行する前に、**[プロパティ]** タブで各コンポーネントにパラメータ値を設定してカスタマイズすることができます。

3-D ワークスペースでインプリシットジオメトリのみを表示したい場合は、**3-D ツールバー**の**添付形状の表示/非表示** (🔍) をクリックし、添付形状を非表示にします。添付形状のみを表示したい場合は、**インプリシットジオメトリの表示/非表示** (🔍) をクリックし、インプリシットジオメトリを非表示にします。

添付形状コンポーネントの詳細は、MapleSim ヘルプシステムの **MapleSim コンポーネントライブラリ > マルチボディ > 可視化 > 概要** を参照してください。

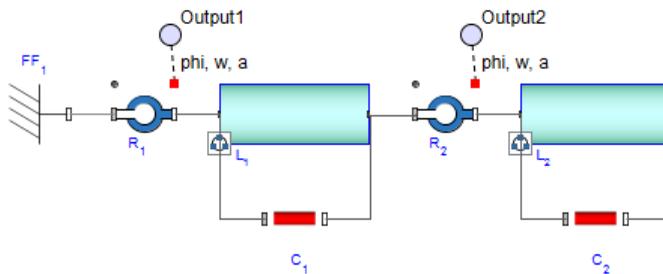
注: マルチボディモデルに **Flexible Beam** コンポーネントが含まれている場合、梁のたわみは 3-D モデルのインプリシットジオメトリでは描写されません。

例: 二重振り子のモデルへの添付形状の追加

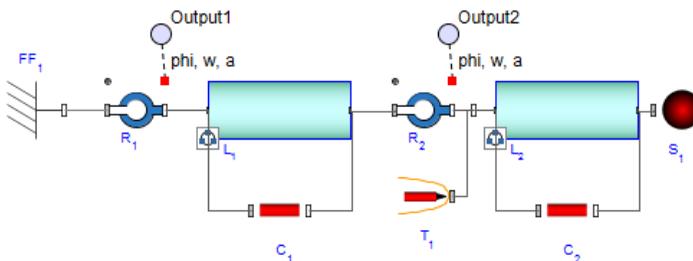
次の例では、振り子のロッドを表す円柱と振り子のおもりを表す球コンポーネントを追加します。また、動画再生時に回転ジョイントが移動する軌跡を示す **Path Trace** コンポーネントも追加します。

添付形状を追加するには

1. [ヘルプ] メニューの [例題] > [ドメイン別例題] > [マルチボディ] から [Double Pendulum] の例題を選択します。
2. [マルチボディ] パレットを展開し、[可視化] メニューを開きます。
3. モデルワークスペース内の平面リンクサブシステムの下部に2つの **Cylindrical Geometry** コンポーネントを追加します。
4. 各コンポーネントを下の図のように接続します。

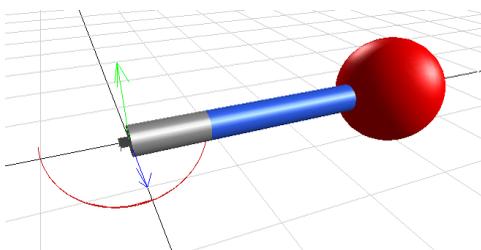


5. 同じメニューから、**Spherical Geometry** コンポーネントを1つ追加し、 L_2 共有サブシステムの右側に配置します。
6. Spherical Geometry コンポーネントを右クリック (Mac では [Control] キーを押しながらクリック) し、[水平方向に反転] を選択します。
7. **Path Trace** コンポーネントを1つ追加し、2つの **Cylindrical Geometry** コンポーネントのあいだに配置します。
8. 各コンポーネントを下の図のように接続します。



9. モデルワークスペースで、最初の **Cylindrical Geometry** コンポーネント (前の図の C_1) を選択します。
10. MapleSim ウィンドウ右側の [**プロパティ**] タブ () で、円柱の半径を「0.2」に変更します。
11. 円柱の色を選択するために、[**color**] フィールドの横に配置されているボックスをクリックし、いずれかの色見本をクリックします。
12. モデルワークスペースで、2つ目の **Cylindrical Geometry** コンポーネント (前の図の C_2) を選択します。
13. この円柱の半径を「0.2」に変更し、色も変更します。
14. **Spherical Geometry** コンポーネント (前の図の S_1) を選択します。
15. この球の半径を「0.8」に変更し、色も変更します。
16. モデルをシミュレートするには、**メインツールバーのシミュレーションの実行** () をクリックします。

シミュレーションが完了すると、[シミュレーション結果] タブが選択された状態で [解析] ウィンドウが開きます。[3-D アニメーションウィンドウ] は、添付形状でモデルを表示します。



17. モデルの動画を再生するには、[3-D アニメーションウィンドウ] にある **再生** () をクリックします。

Path Trace コンポーネントを使用する別の例については、[ヘルプ] メニューの [例題] > [ユーザガイドの例題] > [第 4 章] から、[Lorenz Attractor] の例題を選択します。

3-D ワークスペースでモデルを作成する

MapleSim モデルは、**3-D ワークスペース**にオブジェクトを追加、接続して作成することができます。3-Dモデルにマルチボディコンポーネントを追加するには、**[3-D モデル構築] ウィンドウ**をメインの MapleSim ウィンドウの横に配置し、任意のコンポーネントを**[マルチボディ]**パレット、**[お気に入り]**パレット、以前に作成したカスタムライブラリ、または**[ライブラリコンポーネント]**タブの検索ペインから**3-D ワークスペース**にドラッグします。

3-D ワークスペースでは、**3-D ワークスペース**内に配置されているグラフィカルコントロールを使用して、3-D オブジェクトを追加、接続、および配置することができます。ジョイントやその他のマルチボディコンポーネントについては、初期条件を設定することができます。

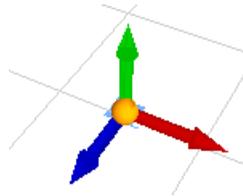
3-D モデルに変更が加えられると、それは自動的に**モデルワークスペース**に表示されているブロックダイアグラムに反映され、逆に、ブロックダイアグラムに変更が加えられると、それは自動的に3-D ワークスペースの3-D モデルに反映されます。たとえば、**3-D ワークスペース**で**Flexible Beam**コンポーネントを追加して接続すると、**モデルワークスペース**には、**Flexible Beam**のブロックダイアグラムと追加された結線が同時に表示されます。

注：

- **3-D ワークスペース**でサブシステムを作成することはできません。サブシステムは、**モデルワークスペース**でのみ作成することができます。
- コンポーネントライブラリの**[マルチボディ]**にある**[力・モーメント]**、**[センサ]**、および**[可視化]**パレット内のコンポーネントを**3-D ワークスペース**にドラッグすることはできません。そのため、これらのコンポーネントを追加する場合は、**モデルワークスペース**で行う必要があります。

3-D ワークスペースでオブジェクトを移動する

3-Dワークスペースで3-D操作ツールをクリックおよびドラッグすることにより、個別のオブジェクトまたはオブジェクトのグループを配置することができます。



単独の未接続オブジェクト用の3-D操作ツールを表示するには、**3-Dワークスペース**内のオブジェクトを1回クリックします。次に、3-D操作ツールの青の矢印をクリックしてからドラッグすると、オブジェクトがZ軸に沿って移動します。緑の矢印を操作するとY軸に沿って、赤の矢印を操作するとX軸に沿って、オブジェクトが移動します。また、3-D操作ツールの中央にある球をクリックしてからドラッグすると、オブジェクトを全方向に移動させることができます。

つながったオブジェクトのグループの場合、3-D操作ツールの位置はモデルの構成によって決まります。

- **Fixed Frame** コンポーネントを含む 3-D モデルでは、**Fixed Frame** コンポーネントを表す四角形をクリックすると 3-D 操作ツールが表示されます。
- **FixedFrame** コンポーネントを含まないモデルでは、システムの初期条件を定義するオブジェクトをクリックすると 3-D 操作ツールが表示されます。たとえば、モデルには **Rigid Body** コンポーネントが含まれており、その初期条件パラメータが **[Strictly Enforce]** に設定されていた場合、3-D 操作ツールはその **RigidBody** コンポーネントに表示されます。**3-Dワークスペース**内でモデルが移動されると、初期条件が **[Strictly Enforce]** に設定されている **Rigid Body** コンポーネントに依存するその他すべての **Rigid Body** コンポーネントの初期条件が更新されます。
- モデルに **Fixed Frame** コンポーネント、または、初期条件が **[Strictly Enforce]** に設定された **Rigid Body** コンポーネントが含まれていない場合は、3-D モデル内のいずれかのオブジェクトをクリックして、3-D操作ツールを表示します。複数のオブジェクトを移動させるときには、モデル内のすべてのマルチボディコンポーネントの初期条件は **[Treat as Guess]** に設定されています。

注: 3-D 操作ツールを表示するには、モデルのマルチボディコンポーネントが数値パラメータを持つ必要があります。パラメータブロックで定義されたカスタムパラメータ値、グローバルパラメータ、またはサブシステムパラメータがマルチボディコンポーネントに割り当てられていると、3-D 操作ツールは**3-D ワークスペース**のコンポーネントをクリックしても表示されません。

3-D モデルのアセンブル

動画を再生する前に、3-D モデルをアセンブルする必要があります。3-D モデルのアセンブルとは、**3-D ワークスペース**に表示されたモデルを、モデルの初期設定(割り当てられたパラメータ値と推定された初期条件によって定義される)と同期させることを意味します。同期プロセスは、モデルをシミュレーションしたときに自動的に発生します。**3-D ツールバー**で、**3-D 表示の更新**()をクリックしてモデルをアセンブルします。**3-D ワークスペース**では、組み立てられたインプリシットジオメトリが濃いグレーで描画され、例外としてジョイントオブジェクトは赤で描画されます。

注: 有効な構成および結線を使う場合にのみ 3-D モデルをアセンブルできます。たとえば、結線が不足している 3-D モデルのアセンブルを試みると、コンソールペインにエラーメッセージが表示され、動画は生成されません。

詳細は、MapleSim ヘルプシステムの 3-D モデルのアセンブルを参照してください。

拘束解除ボタンを使用して 3-D ワークスペースのジョイントを操作する

3-D ワークスペース内のジョイントオブジェクトを選択し、**運動学拘束を無効化**()をクリックして、モデルを作成するあいだ**3-D ワークスペース**内でジョイントの運動学的拘束が適用されないようにすることができます。運動学的拘束条件を一時的に解除された関節は、**3-D ワークスペース**でピンク色で表示され、モデル構築中その初期条件は**3-D ワークスペース**に表示されません。

運動学拘束を無効化()は、たとえば、**3-D ワークスペース**で閉ループの 3-D モデルを作成し、配置しているあいだ、ジョイントを特定の位置に固定しておきたい場合などに使用します。

注:

- 拘束解除ボタンによる拘束解除は、[プロパティ] タブに指定されているジョイントコンポーネントに対する実際の初期条件には影響しません。このボタンは、**3-D ワークスペース**に表示される初期条件に対してのみ作用します。
- 運動学的拘束が適用されているほかのジョイントについては、その初期条件は **3-Dワークスペース**に表示されますが、運動学的拘束が解除されている関連ジョイントに影響をあたえるようなことはありません。

たとえば、運動学的拘束が解除されている回転ジョイント1つと、運動学的拘束が適用されている回転ジョイント1つをそれぞれ含む二重振り子の3-Dモデルがあるとします。運動学的拘束が適用されているジョイントの初期角度を変更すると、運動学的拘束が解除されているジョイントの位置は元のまま変わりませんが、運動学的拘束が適用されているジョイントは新しい初期角度で表示されます。**3-D ワークスペース**の新しい初期条件をすべて表示するには、**3-D 表示の更新** () をクリックするか、シミュレーションを実行してモデルをアSEMBルする必要があります。

3-D モデル作成中に添付形状を表示する

モデルワークスペースのブロックダイアグラムに **Cylindrical Geometry**、**Tapered Cylinder Geometry**、**Box Geometry**、または **Spherical Geometry** コンポーネントを接続すると、**3-D ワークスペース**と **[3-D アニメーションウィンドウ]**のどちらの場合でも対応する添付形状が表示されます。すべてのポートが**モデルワークスペース**に存在するマルチボディコンポーネントの対象ポートに接続されると、添付形状が**3-D ワークスペース**に表示されます。

CAD ジオメトリの操作

CAD ジオメトリは**3-D ワークスペース**および **[3-D アニメーションウィンドウ]**の両方で表示されます。**CAD Geometry** コンポーネントを**モデルワークスペース**に追加すると、該当する CAD 画像が **3-D ワークスペース**に表示されます。**CAD Geometry** コンポーネントは、モデル内のほかのコンポーネントに接続されている、いないにかかわらず、表示されます。**CAD Geometry** コンポーネントがほかのコンポーネントに接続されていない場合は、3-D グリッドの原点に表示されます。**CAD Geometry** がほかのコンポーネントに接続されている場合は、追加されているモデリングコンポーネントの座標フレームの原点に表示されます。

CAD 画像の並進オフセットおよび回転オフセットについては、該当する **CAD Geometry** コンポーネントをモデルに接続する前でも、接続した後でも定義することができます。これらのオフセットを定義するには、**モデルワークスペース**で **CAD Geometry** コンポーネントを選択し、[プロパティ] タブでパラメータ値を指定します。

例：3-D ワークスペースで二重振り子のモデルを作成して動画を再生する

この例では、二重振り子のモデルを作成し、動画を再生します。ここでは、以下のタスクを実行します。

二重振り子を作成して動画を再生するには

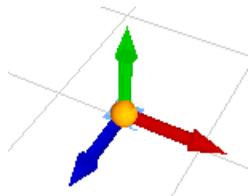
1. **3-D ワークスペース**にオブジェクトを追加し、移動します。
2. 3-D オブジェクトを接続します。
3. モデル内のジョイントに初期条件を設定します。
4. 3-D モデルの動画を再生します。

3-D ワークスペースにオブジェクトを追加し、移動する

オブジェクトを追加または移動するには

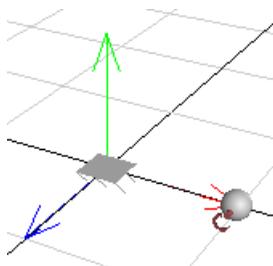
1. 新しい MapleSim ドキュメントを開きます。
2. MapleSim メインツールバーから **3-D ワークスペースを表示** () をクリックし、MapleSim のメインウィンドウの横に配置します。
3. **常に最前面に表示** () をクリックします。これにより、MapleSim ウィンドウで作業するあいだ常に [解析ウィンドウ] を最前面に表示できます。
4. [ライブラリコンポーネント] タブで [マルチボディ] パレットを展開し、[ボディ・フレーム] メニューを開きます。

5. パレットから、**Fixed Frame** コンポーネントを **3-D ワークスペース** にドラッグします。**Fixed Frame** コンポーネントを表す灰色の四角形が **3-D ワークスペース** に追加され、その 3-D 操作ツールが表示されます。

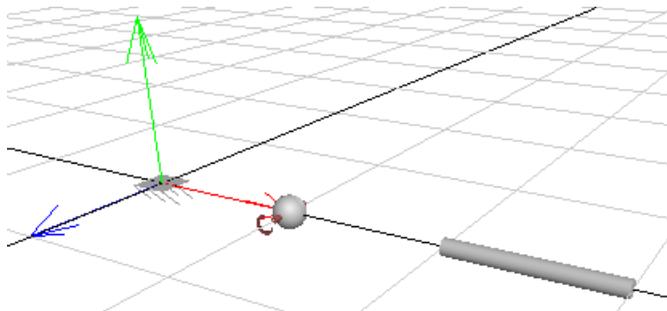


このマニピュレータを使用すると、**3-D ワークスペース** でオブジェクトを配置することができます。

6. 3-D 操作ツールの矢印コントロールをクリックし、ドラッグして **Fixed Frame** オブジェクトをグリッドの原点に配置します。
7. [マルチボディ]>[ジョイント・モーション]メニューから、**Revolute** コンポーネントを **3-D ワークスペース** にドラッグし、**Fixed Frame** の右側に配置します。



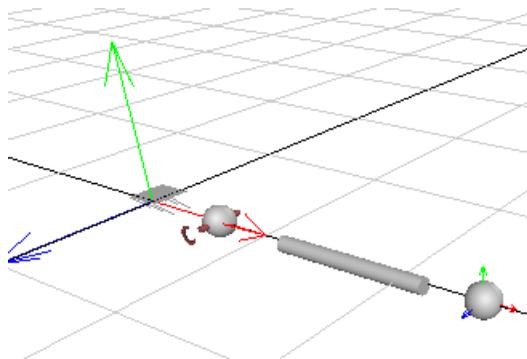
8. [マルチボディ]>[ボディ・フレーム]メニューから、**Rigid Body Frame** コンポーネントを**3-Dワークスペース**にドラッグし、**Revolute**の右側に配置します。



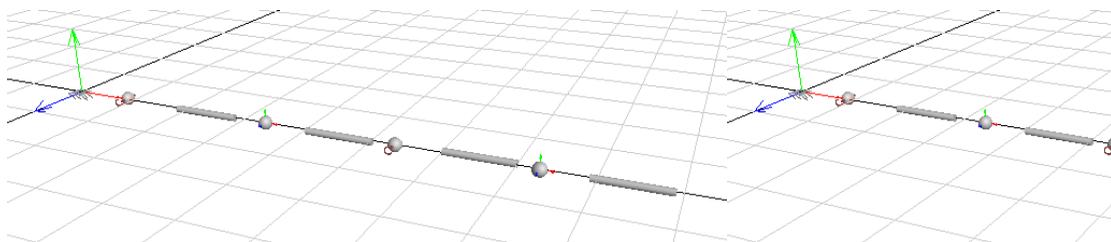
9. 同じメニューから、**RigidBody** コンポーネントを**3-Dワークスペース**にドラッグし、**Rigid Body Frame**の右側に配置します。

ヒント: 3-Dワークスペースで拡大/縮小するには、焦点を当てたいオブジェクト上にマウスポインタを置き、マウスホイールを回転します。マウスホイールを使ってズームするときは、ポインタの下の位置は不動となるため、その位置に拡大することができます。モデルをパンする場合は、[Shift]キーを押しながら**3-Dワークスペース**でマウスポインタをドラッグします。

10. 同じメニューから、**Rigid Body Frame** コンポーネントをもう1つ**3-Dワークスペース**にドラッグし、**Rigid Body**の右側に配置します。これで、振り子の1つ目のリンクに必要なコンポーネントが揃います。



- ステップ6から9を繰り返し、最後に追加した **Rigid Body Frame** コンポーネントの右側に振り子の2つ目のリンクに必要なコンポーネントを追加します。

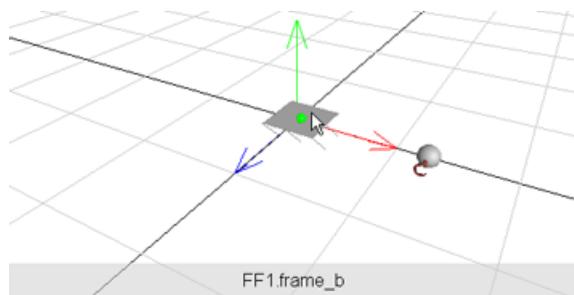


3-D オブジェクトの接続

次に、前のタスクで追加したオブジェクトを接続します。

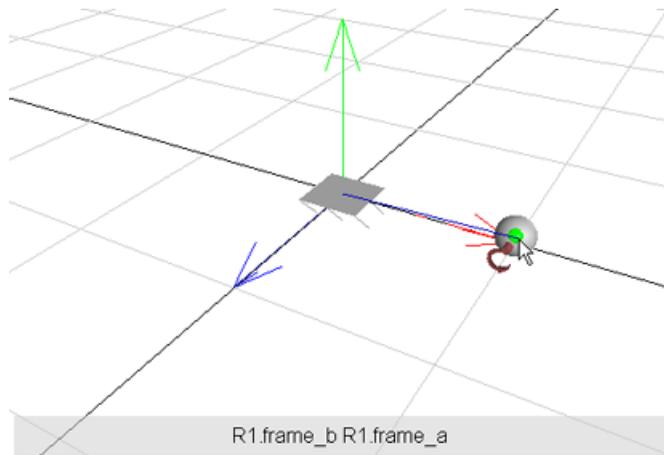
オブジェクトを接続するには

- ポートを接続 () をクリックします。
- マウスポインタを **Fixed Frame** オブジェクトの上に移動させます。緑色の点が表示されます。

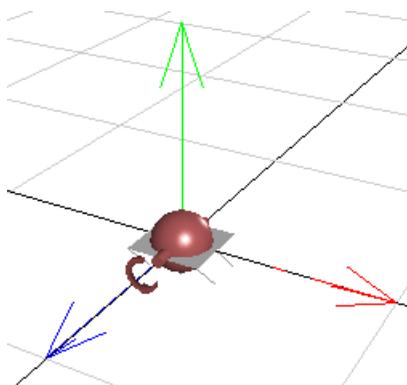


- 緑色の点を1回クリックし、結線を引きはじめます。

4. マウスポインタを最初の **Revolute** ジョイントコンポーネントの上に移動させます。3-D ワークスペースの下部にある灰色のパネルに **Revolute** ジョイントフレームの名前が表示されます。



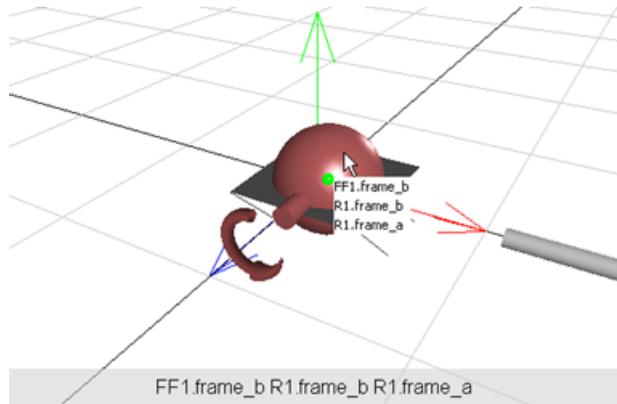
5. **Revolute** ジョイントコンポーネントを 1 回クリックします。ラインを接続することができるフレーム名を表示するコンテキストメニューが表示されます。
6. **[R1.Frame_a]** を選択します。3-D ワークスペースでコンポーネントが接続されます。



接続されると、ジョイントコンポーネントは赤で表示されることにご注目ください。

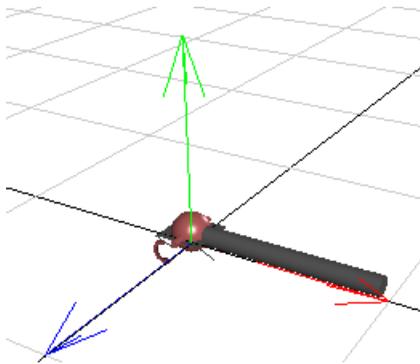
7. **ポートを接続** (🔗) をクリックし、次の結線を引き始めます。

8. **Revolute** ジョイントを表す球をクリックします。**Revolute** ジョイントのフレーム、および Revolute ジョイントが接続されている **Fixed Frame** を表示するコンテキストメニューが表示されます。



9. コンテキストメニューから、**[R1.frame_b]** を選択します。

10. **Rigid Body Frame** を表す円柱の端までマウスポインタをドラッグし、緑色の点をクリックします。



これで、1つ目の回転ジョイント (R_1) の **frame_b** は、1つ目の剛体フレーム (RBF_1) の **frame_a** に接続されます。

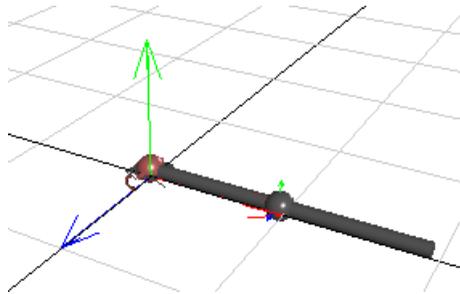
11. **[ポートを接続]** をクリックし、新しい結線を引き始めます。

12. RBF_1 コンポーネントを表す円柱のもう一方の端の上にマウスポインタを移動し、円柱を1回クリックします。

13 1つ目の **Rigid Body** コンポーネント (RB_1) を表す球にマウスポインタをドラッグし、1回クリックします。これで、 RB_1 が RBF_1 に接続されます。

14 同様に、 RB_1 の **frame_a** を 2つ目の **Rigid Body Frame** (RBF_2) の **frame_a** に接続します。

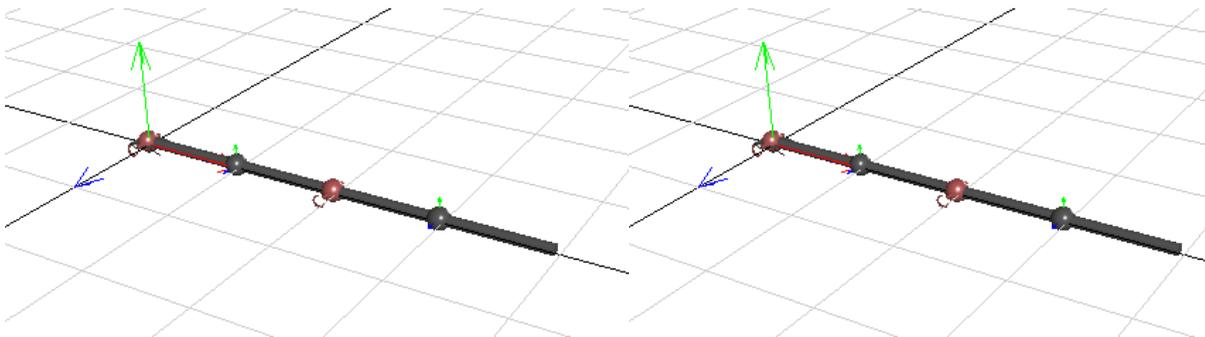
注: 各結線を引き始めるときは、接続ボタンをクリックしてください。



15 2つ目の **Rigid Body Frame** の **frame_b** を 2つ目の **Revolute** ジョイントの **frame_a** に接続します。

16 2つ目の **Revolute** ジョイントの **frame_b** を 3つ目の **Rigid Body Frame** の **frame_a** に接続します。

17 3つ目の **Rigid Body Frame** を 2つ目の **Rigid Body** に接続し、2つ目の **Rigid Body** を 4つ目の **Rigid Body Frame** に接続します。完成した 3-D モデルを下の図に示します。



2-D モデルワークスペースでは、すべてのコンポーネントが適宜追加され、接続されていることが確認できます。

ヒント: 3-D モデルを作成するときは、定期的にブロックダイアグラム表示に切り替え、ブロックダイアグラムでモデルが構想どおりにレイアウトされているか確認することがベストプラクティスとして推奨されます。

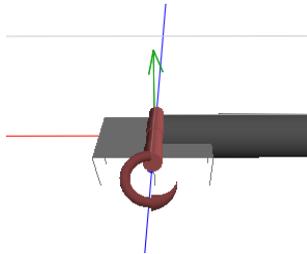
ジョイントコンポーネントに初期条件を設定する

3-Dワークスペースに配置されているグラフィカルコントロールを使用して、ジョイントコンポーネントに初期条件を設定することができます。

注: パラメータブロック、グローバルパラメータ、サブシステムパラメータの何れかで定義されたカスタムパラメータ値を割り当てられたジョイントコンポーネントでは、初期値設定のためのグラフィカルコントロールは使えません。このような場合は、**[プロパティ]** タブのフィールドから初期条件を設定してください。

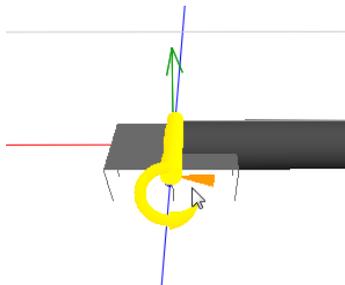
初期条件を設定するには

1. 1つ目の回転ジョイントの初期角度を設定するために、**3-Dワークスペース**で、1つ目の回転ジョイントを表している球をクリックします。ジョイントコンポーネントを表す赤い球が**3-Dワークスペース**から一時的に削除され、ジョイントのマニピュレータが表示されます。



2. マウスポインタをマニピュレータの上に移動します。マニピュレータが黄色で表示されます。

3. 回転ジョイントに設定する初期角度の値を表すメータを表示するために、マウスポインタをクリックし、マニピュレータの周りをドラッグします。オレンジ色の、円グラフの形をしたメータが表示されます。



マウスポインタをドラッグすると、グラフで表されている自由度の初期角度値を調整することができます。マウスポインタを上または右にドラッグすると角度値は大きくなり、下または左にドラッグすると、角度値は小さくなります。

4. メータが設定したい初期条件値に近づいたら、マウスのボタンを放します。[プロパティ] タブ (📄) の θ_0 パラメータに選択した値が表示されます。この値が 3-D ワークスペースのインプリシットジオメトリに設定されます。

ヒント:

- モデルの初期条件は、[プロパティ] タブの $[\theta_0]$ パラメータに値を入力して設定することもできます。[プロパティ] タブで指定された初期条件は、3-D モデルに反映されます。
- 角度の初期条件を正確に指定するには、[マルチボディ設定] タブ (🔧) の [Visualization] で [Enable Rotational Snapping] を選択して、スナップを有効にします。

3-D モデルの動画を再生する

ここでは、3-D モデルをシミュレートし、アニメーションウィンドウで表示可能な動画を生成します。

3-D モデルの動画を再生するには

1. **メインツールバーのシミュレーション実行** () をクリックし、モデルのシミュレーションを実行します。シミュレーションが完了すると、[解析ウィンドウ] が開き、シミュレーションのプロットと [3-D アニメーションウィンドウ] の両方を表示します。
2. [3-D アニメーションウィンドウ] を選択します (アニメーションウィンドウが表示されない場合は、[解析ウィンドウ] の左ペインにある [プロットウィンドウ] パレットで、[3-D アニメーションウィンドウ] をダブルクリックします)。
3. アニメーションを再生するには、**再生ツールバーの再生** () をクリックします。

3-D モデルの動画のエクスポート

動画エクスポート機能を使用すると、記録されたシミュレーションを動画ファイル (AVI など) としてエクスポートして、MapleSim を使用せずに再生したり、MapleSim を持っていない可能性のあるユーザと共有することができます。詳細は、MapleSim ヘルプシステムの **MapleSim 操作方法 > 3-D アニメーション > シミュレーションを動画としてエクスポート** を参照してください。

4.7. ベストプラクティス: モデルのシミュレーションと可視化

このセクションでは、モデルのシミュレーションと可視化におけるベストプラクティス (最良の方法) について説明します。

時間がかかるシミュレーションの実行には外部のCコンパイラを使用する

[シミュレーションの設定] タブ () で [Compiler] のチェックボックスがオンになっている場合、シミュレーションエンジンによって生成される Maple プロシージャはCコードに変換され、外部のCコンパイラによってコンパイルされます。その結果、シミュレーションを実行するのに必要な時間が短縮されます。一般的に、Cコンパイラを使用してモデルのシミュレーションを実行すると、時間がかかるシミュレーションほどコンパイル時間が短くなります。

モデルの一部で生成される結果を比較する

デバッグのため、モデルの特定部分または特定サブシステムのシミュレーション結果を確認したい場合があります。モデル内の特定の部分を選択し、**モデルワークスペース**の上にある**無効化** (🚫) をクリックするか、**[Ctrl] + [E]** キー (Mac では **[Command] + [E]** キー) を押して、次に実行するシミュレーションからモデル内の対象部分を除外することができます。この状態でシミュレーションを実行すると、モデルのうち除外されなかった部分の結果のみが表示されます。つまり、この機能を使用すると、**モデルワークスペース**からコンポーネントを削除したり、複数のモデルを作成する必要なく、モデルの特定部分によって生成されるシミュレーション結果を表示し、比較することができます。

詳細は、MapleSim ヘルプシステムの **MapleSim 操作方法 > モデルのシミュレーション > シミュレーションからオブジェクトを除外する** を参照してください。

第5章 モデルの解析と操作

この章の内容は以下のとおりです。

- 概要 [155ページ]
- モデルからの方程式とプロパティの取得 [160ページ]
- 線形系の解析 [161ページ]
- パラメータの最適化 [162ページ]
- モデルからのCコードの生成とエクスポート [164ページ]
- 外部Cコード/ライブラリ定義からカスタムコンポーネントを生成 [173ページ]
- Maple の埋め込みコンポーネントを扱う [178ページ]

5.1. 概要

MapleSim は Maple 環境に完全に統合されています。

- Maple ベースの **Apps** を MapleSim インターフェース内で使用できます。
- モデルに**テンプレート**を添付し、Maple インターフェースで開くことができます。
- MapleSim モデルに **Maple ワークシート**を添付することにより Maple のコマンド、埋め込みコンポーネント、プロットツール、およびその他の機能に完全にアクセスすることができ、MapleSim モデルやサブシステムの動的挙動を解析し、操作することができます。
- Maple ワークシート内で MapleSim アプリケーションプログラミングインターフェース (**API**) を使用して、既存の MapleSim モデルをプログラムで操作、シミュレーション、および解析することができます。

MapleSim Apps とテンプレート

Maple で MapleSim のモデルを操作するには、まず **[Apps とテンプレートを追**

加] タブ () で利用可能な Apps およびテンプレートを使用します。Apps は、モデルの作成や解析タスク用のプリビルトツールです。最初に MapleSim モデルを作成し、用意されている Apps のいずれかで開いて解析タスクを実行します。

AppsはMapleSimの[解析ウィンドウ]で開きます。たとえば、Appsを使用してパラメータスイープ、モンテカルロシミュレーション、またはコード生成を実行できます。

テンプレートは、モデルに添付するMapleワークシートです。Mapleが起動し、テンプレートが開きます。テンプレートを使用して、サブシステムの方方程式を取得および操作したり、カスタムコンポーネントの構築を行うことができます。

以下の表は、[Appsとテンプレートを追加]タブ()で利用可能なMapleSim Appsおよびテンプレートをリストしています。

MapleSim Apps

App名	タスク
コンポーネント作成	
1-D モーション生成	定義された速度、加速度、加加速度の拘束に基づく1-Dモーションのモーションプロファイルを作成します。
カスタムポート	カスタムコンポーネントにカスタムポートを作成する詳細は、 カスタムコンポーネントの作成 [83ページ] カスタムコンポーネントの作成 [83ページ] を参照してください。
方向制御弁ビルダー	カスタムな油圧方向制御弁を作成します。
外部C/ライブラリブロック	モデルをCコードに変換します。詳細は、 モデルからのCコードの生成とエクスポート [164ページ] を参照してください。
運動学的カムの生成	カムとフォロアの運動学的挙動をモデル化します。
線形化	MapleSim連続サブシステムを線形化します。線形システムオブジェクトに対して線形解析を実行します(例: ボード線図および根軌跡プロット)。
モデル解析	
方程式の抽出	線形または非線形モデルから方程式を取得します。詳細は、 チュートリアル7: 方程式の抽出 App の使用 [247ページ] チュートリアル7: 方程式の抽出 App の使用 [247ページ] を参照してください。

App 名	タスク
初期化診断	不整合な初期条件とシミュレーション中に発生したエラーを解決し、元のシステム構成に関する説明を提供します。
モーダル解析	マルチボディモデルの振動モードを可視化します。
モンテカルロシミュレーション	パラメータに乱数分布を定義し、この分布を用いてシミュレーションを実行します。
マルチボディ解析	操作と解析に適した形式でマルチボディシステムモデルの方程式を取得します。
最適化	モデルのパラメータを解析および編集し、シミュレーションで見込まれる結果をグラフに表示します。詳細については、パラメータの最適化 [162ページ] パラメータの最適化 [162ページ] を参照してください。
パラメータスイープ	パラメータスイープを実行します。
ユーティリティ	
コード生成	モデルを C コードに変換します。詳しくはモデルからの C コードの生成とエクスポート [164ページ] を参照してください。
データ生成	MapleSim で使用するデータセット (たとえば、補間テーブルコンポーネントのデータセット) を定義し、生成します。詳細は、補間テーブルコンポーネントのデータセットの作成 [73ページ] 補間テーブルコンポーネントのデータセットの作成 [73ページ] を参照してください。
Excel 接続	Excel スプレッドシートから MapleSim パラメータセットをインポートし、MapleSim パラメータセットを Excel スプレッドシートへエクスポートします。
乱数データの生成	MapleSim で使用するランダムなデータ点の集合 (たとえば、補間テーブルコンポーネント用データセット) を定義し、生成します。
Heat Transfer (MapleSim Heat Transfer Library は、アドオンとして提供されています。)	
基板と部品のコンポーネント生成	基板と部品のコンポーネントを生成します。

App 名	タスク
温度分布の可視化	色に温度をマッピングする Heat Transfer 形状コンポーネントの温度分布の3-Dアニメーションをプロットします。(MapleSim Heat Transfer Library は、アドオンとして提供されています。)

MapleSim テンプレート

テンプレート名	タスク
カスタムコンポーネント	数理モデルに基づいたカスタムコンポーネントを作成します。詳細は、 カスタムコンポーネントの作成 [83ページ] を参照してください。
ワークシート	埋め込みコンポーネントで MapleSim モデルを開き、ワークシートを作成します。

注: MapleSim テンプレートを使用した後は、まず .mw ファイルを保存してから .mw ファイルが添付された .msim ファイルを保存してください。

Apps の操作

App を閉じて再び開いた場合、Apps マネージャによってアプリの以前の状態が記憶されています。

Apps マネージャには3つのオプションが表示されます。

- **更新** (↕): App をデフォルト設定に戻します。
- **エクスポート** (↑): App の現在の設定を保存します。設定はデータファイルとして保存されます。
- **インポート** (↓): 保存された App 設定を取得します。

Maple ワークシートで MapleSim の方程式とプロパティを扱う

Maple テンプレートで MapleSim の方程式またはプロパティを表示および操作する場合、対応するパラメータ、変数、コネクタ、下付き文字、上付き文字が異なるものにマッピングされたり、表記が異なる場合があります。

MapleSim のプログラム名の Maple へのマッピング

Maple ワークシートに表示される、特定パラメータや変数、コネクタのプログラム名は、MapleSim インターフェースで表示される当該要素の名前と異なる場合があります。たとえば、モデルに **Inertia** コンポーネントが含まれている場合、角速度の初期値として MapleSim インターフェースに表示されるパラメータは ω_0 ですが、Maple ワークシートでは w_start です。パラメータ名、変数名およびコネクタ名のマッピングに関する詳細は、MapleSim ヘルプシステムの **MapleSim コンポーネントライブラリ** を参照してください。

MapleSim の上付き文字と下付き文字の Maple での表記

MapleSim インターフェースの下付き文字および上付き文字は、Maple ワークシートでは異なる表記で表示されます。MapleSim インターフェースの下付き文字は、Maple ワークシートでは下線とともに表示されます。たとえば、MapleSim インターフェースで $flange_a$ と表されるコネクタは、Maple ワークシートでは $flange_a$ と表示されます。また、上付き文字は、Maple ワークシートでは通常文字として表示されます。たとえば、MapleSim インターフェースで a^2 と表される変数は、Maple ワークシートでは $a2$ と表示されます。

サブシステムの使用

モデルをエクスポートする場合の基本構造はサブシステムです。App またはテンプレートにより、解析および操作するサブシステム全体を選択できます。モデルまたはモデルの一部をサブシステムに変換することによって、調査対象のモデリングコンポーネントセットをより簡単に識別したり、サブシステムの入出力を定義したり、調査するコンポーネントをブロックコンポーネントとして識別できます。MapleSim でサブシステムを作成するときのベストプラクティス (最良の方法) については、[ベストプラクティス: サブシステムの作成と配置 \[75ページ\]](#) を参照してください。

モデルをエクスポートする際の基本構造については、**[ヘルプ]** メニューの **[例題]** > **[ユーザガイドの例題]** > **[第 5 章]** を選択し、**[Preparing a Model for Export]** の例題を選択します。

注: サブシステムのコードを生成する場合、含まれるポートは実数の入力ポートと実数の出力ポートである必要があります。トップレベルのシステムのコードを生成する場合、システムには入力がないとみなされ、すべてのプローブ値は出力として処理されます。

ヒント: モデル全体を使用する場合は、モデルのトップレベルですべてのコンポーネントをグループ化し、1つのサブシステムにします。

5.2. モデルからの方程式とプロパティの取得

方程式の抽出Appは、モデルのパラメータや変数などのプロパティと方程式を取得、定義、解析するために使用することができます。このApp内の追加機能は、複数のサブシステムが存在する場合に再利用可能な方程式を生成する場合に便利です。

方程式の抽出Appの完全なチュートリアルについては、チュートリアル7: 方程式の抽出Appの使用 [247ページ]を参照してください。

方程式とプロパティを取得するには

1. MapleSim で、方程式またはプロパティを取得するモデルを開きます。
2. **[Apps とテンプレートを追加]** タブ () をクリックします。
3. **[Apps]** パレットから **[方程式の抽出]** を選択します。
4. **[サブシステムの選択]** の下にあるナビゲーションツールを使用して、方程式を表示するサブシステムを選択します。システム全体の方程式を取得する場合は、**[Main]** をクリックします。
5. **[選択したサブシステムを読み込む]** をクリックします。モデルの方程式が抽出され、システムパラメータと変数がロードされます。**[方程式の表示]** セクションにある **[方程式の抽出]** をクリックします。システム方程式が表示され、変数 DAE に自動的に保存されます。

5.3. 線形系の解析

線形化Appは、線形系の方程式の取得、表示、解析、入出力値のテスト、およびシミュレーション結果をボード線図、ナイキストプロット、または根軌跡プロットに表示するために使用することができます。

注:線形解析の対象としてシステム全体を指定することはできません。テンプレートの**解析とシミュレーション**セクションのツールを使用して線形解析を行うには、サブシステムを選択する必要があります。

MapleSim の線形系モデルを解析するには

1. MapleSim で、解析する線形系モデルを開きます。
2. **[Apps とテンプレートを追加]** タブ () をクリックします。
3. **[Apps]** パレットから **[線形化]** を選択します。App は **[解析ウィンドウ]** で開きます。
4. モデルダイアグラムの上にあるナビゲーションツールを使用して、方程式を表示するサブシステムを選択します。
5. **[選択したサブシステムを読み込む]** をクリックします。
6. (オプション) **[設定]** セクションで変更を行います。
7. **[線形化]** をクリックします。線形系オブジェクトが作成され、システムの方程式が表示されます。
8. (オプション) ボード線図、ナイキストプロット、根軌跡プロット、または応答プロットを作成します。
9. システムの解析および設定後、システムに基づいてカスタムコンポーネントを作成し、MapleSim モデルに添付することができます。

線形解析

[解析] セクションのツールを使用して、線形系の解析やシステムの出力に対するさまざまな入力の効果を表示することができます。

以下のツールを解析に使用できます。

- ボード線図

- ナイキストプロット
- 根軌跡プロット
- 応答プロット

[応答] プロットで、システムに適用する入力信号を選択し、出力に対する効果をシミュレーションすることができます。

コンポーネントの作成

システムの解析および設定後、システムに基づいてカスタムコンポーネントを作成し、MapleSim モデルに添付することができます。

システムからカスタムコンポーネントを作成するには

1. [モデルの作成] セクションで、[コンポーネントの名前] テキストボックスに名前を入力します。
2. [説明] テキストボックスにコンポーネントの説明を入力します。
3. [作成] をクリックします。

カスタムコンポーネントは、MapleSim モデルの [ローカルコンポーネント] タブ () にある [コンポーネント] パレットに表示されます。

5.4. パラメータの最適化

パラメータの最適化 App を使用して、モデルのパラメータをテストしたり、シミュレーションのプロットを表示したり、Maple プロシージャにパラメータを割り当ててパラメータスイープやその他の高度な最適化タスクを実行できます。

パラメータの最適化は、Global Optimization Toolbox のコマンドで実行することもできますが、この製品は、MapleSim には含まれていません。この製品に関する詳細は、Maplesoft Global Optimization Toolbox の、次のウェブサイトで参照してください。

<http://www.maplesoft.com/products/toolboxes/globaloptimization/>

パラメータを最適化するには

1. MapleSim で、解析する線形系モデルを開きます。
2. [Apps とテンプレートを追加] タブ () をクリックします。
3. [Apps] パレットで、[最適化] をダブルクリックします。App は [解析ウィンドウ] で開きます。
4. [サブシステムの選択] の下にあるナビゲーションツールを使用して、方程式を表示するサブシステムを選択します。システム全体の方程式を取得する場合は、[Main] をクリックします。
5. [システムを読み込む] をクリックします。モデルのシミュレーション設定がインポートされます。
6. [パラメータ] セクションでは、コンボボックスのリストを使って最適化したいパラメータを選択し、[追加] をクリックします。他のパラメータも同様に選択すると、リストボックスに選択したパラメータが表示されます。

設定: パラメータ 目的関数 ?

フィルタ: 明示的なパラメータのみ

L 評価範囲:

L
C
R

Main, RL, C, L

単位: 基準値: 1 最小: 最大:

注: パラメータを選択すると、現在の値 (基準値) が [基準値] に表示されます。

7. パラメータが変化する範囲を、[最小] および [最大] の各フィールドに設定します。
8. 上記手順を繰り返し、最適化したいその他のパラメータの [最小] および [最大] の各フィールドを設定します。
9. すべてのパラメータを定義したら、[目的関数] を選択して目的関数の構築方法の詳細と、[目的関数の最小化] または [目的関数の最大化] を指定できます。目的関数は Maple プロシージャとして定義されます。

Maple のプロシージャに関する詳細は、Maple ヘルプシステムの プロシージャを参照してください。

10.これで、パラメータの最適化を実行できるようになります。Maplesoft Global Optimization Toolbox を保有している場合は、このステップで使用できます。**[パラメータ最適化を実行]** をクリックしてパラメータ最適化を実行します。**[結果]** セクションには、目的関数を最適化するパラメータ値が表示されます。

別の値を使用してパラメータをテストするには、スライダを動かしてから、**[シミュレーションを実行]** をクリックします。ステップ 10 で計算されたパラメータ値に戻すには、**[最適値のリストア]** をクリックします。

11.リンクされたモデルでスライダのパラメータ値を使用するには、**[MapleSim モデルのパラメータを更新]** をクリックします。

5.5. モデルからの C コードの生成とエクスポート

C 言語をサポートするアプリケーションでモデルを使用またはテストする場合は、**コード生成App** を使用して、モデルまたはモデル内のサブシステムを C コードに変換できます。基本的な C コードとコンパイルおよび実行機能は、Maple からアクセスできます。このコードの拡張機能については、Connector の追加のツールボックスとして、さまざまなソフトウェアツールで利用できます。

MapleSim コードのエクスポート処理に関する詳しい説明については、エクスポートするためのコード生成を参照してください。

コード生成App を使用すると、システムの入出力を定義したり、コードの最適化レベルを設定したり、ソースコードを生成したり、結果のコンポーネントやライブラリコードの形式を選択したりすることができます。タスク解析の実行、モデル方程式の変数への代入、入出力のグループ化、変数の追加の入出力ポートの定義を実行するには、Maple コマンドを使用できます。

注: C コードの生成では、定義された信号入力 (RealInput) および信号出力 (RealOutput) ポートを持つ混在した連立方程式を含め、MapleSim でモデリングするすべてのシステムを処理できます。

コードをエクスポートしたり、方程式を生成したりする場合、そのモデルに対するパラメータのサブセットのみが表示されることがあります。以下のパラメータはエクスポートできません。

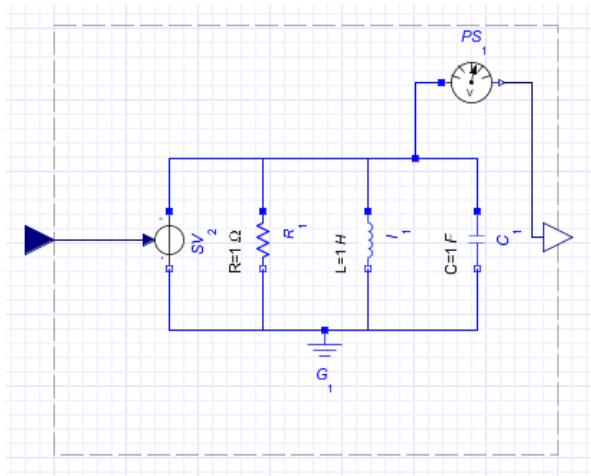
- マルチボディパラメータを直接エクスポートすることはできません。[プロパティ] タブ (📄) でマルチボディパラメータが割り当てられているユーザ定義のパラメータのみ、エクスポートできます。
- 従属パラメータはエクスポートできません。パラメータ A が b の関数である場合 ($A=b$ 、 $A=\sin(b)$ 、 $A=1+3/b$ など)、A をエクスポートすることはできません。A は b の関数として方程式に直接代入されます。b はエクスポートすることができます。
- 方程式の数が変わるパラメータはエクスポートできません。
- 離散値に対するパラメータはエクスポートできません。

MapleSim モデルから C コードを生成するプロセスには次のステップが含まれています。

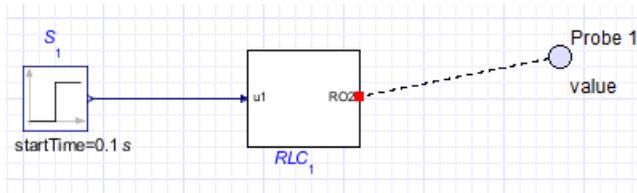
- MapleSim モデルの準備
- コード生成 App を開く
- サブシステムのロード
- 特定のポートに対するパラメータ値のカスタマイズ、定義、および割り当て
- コード生成オプションの選択
- C コードの生成と保存

MapleSim でエクスポートする場合のモデルの準備

モデルをエクスポートする場合の基本構造はサブシステムです。このサブシステムには、生成されたコードからの入力信号と出力信号を定義します。サブシステムを作成することによって、**モデルワークスペース**でのシステムの視覚的なレイアウトを改善することもできます。次の図に、定義した入力(青の矢印)と定義した出力(白の矢印)を持つサブシステムを示します。サブシステムのコードを生成する場合、すべてのポートは実数の入力ポートと実数の出力ポートである必要があります。

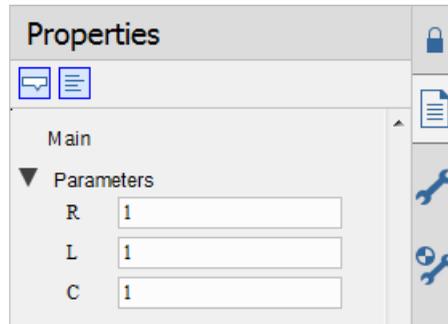


トップレベルのシステムのコードを生成する場合、次の図に示すように、入力はありませんがすべてのプローブ値が出力であるとみなされます。



ヒント: モデル全体に対してコードを生成するには、モデルのトップレベルですべてのコンポーネントをグループ化し、1つのサブシステムにします。

入出力に加えて、生成されたコードにはユーザーが変更可能なパラメータが定義されている場合があります。デフォルトでは、エクスポートされるコードのすべてのパラメータが変更可能になってはなりません。一般的に、編集可能なパラメータが少ないほど、エクスポートされるコードの生成や実行にかかる時間が短縮されます。デフォルトでは、エクスポートされるサブシステムに定義されているパラメータのみが、生成されるコードでも変更可能になります。上記の例でRLCサブシステムのコードを生成する場合、デフォルトではパラメータR、L、Cのみがエクスポートされるコード内で変更可能です。



注: 生成されるコードですべてのパラメータが変更可能というわけではありません。変更により、システムに変数の追加や削除が必要となり方程式の構造に影響するパラメータは、エクスポート対象のパラメータのリストから自動的に削除されます。これは、そのパラメータがエクスポートされたサブシステムに定義されている場合も同様です。

初期化

すべての離散イベントは、対応する MapleSim モデルと同じ値に初期化されます。たとえば、MapleSim モデルでクラッチが「ロック済み」として初期化された場合、生成されるコードでは、最初にクラッチは「ロック済み」という設定であると想定します。同じことが連続変数やその導関数にも当てはまります。

エクスポートされるコードは初期状態の MapleSim モデルから初期条件を取得するため、シミュレーション可能なモデルに含まれるサブシステムのコードのみがエクスポートされる場合があります。

注: MapleSim でモデルを実行または初期化できない場合、そのモデルまたはサブシステムのコードをエクスポートできません。

コード生成 App を開く

コード生成を実行するには、まずコード生成 App を開きます。

1. [Apps とテンプレートを追加] タブ () をクリックします。
2. [Apps] パレットから [コード生成] を選択します。C コード生成 App が [解析ウィンドウ] で開きます。

サブシステムのロード

Appの[サブシステムの選択]セクションで、コードを生成およびエクスポートするサブシステムを選択します。サブシステムを選択したら、**[選択したサブシステムを読み込む]**をクリックします。定義したすべての入出力ポートがロードされます。

特定のポートに対するパラメータ値のカスタマイズ、定義、および割り当て

[設定] インターフェースを使用すると、特定のポートに対するパラメータ値をカスタマイズ、定義、および割り当てることができます。パラメータを割り当てられたサブシステムコンポーネントは、サブシステムのレベルで定義されたパラメータ値を継承します。

Configuration: Inputs Outputs Parameters Code Export Options

ヒント: Appを閉じて再び開いた場合、Apps マネージャによってアプリの以前の状態が記憶されています。

更新、エクスポート、およびインポートボタンを使用すると、Appを閉じて再び開いた場合に設定を維持または取得することができます。

- **更新** (↕): App をデフォルト設定に戻します。
- **エクスポート** (↑): App の現在の設定を保存します。
- **インポート** (↓): 保存された App 設定を取得します。

Inputs:

Input Variables	Change Row
1	

[入力]: モデルの入力変数です。

[行を変更]: 指定された行で方程式を選択します。

Outputs:

Toggle Export Column

	Output Variables	Export	Change Row
1		x*	

Add an additional output port for subsystem state variables

[出力]: モデルの出力変数です。

[エクスポート]: 数式形式で残したい変数を選択します。

[行を変更]: 指定された行で方程式を選択します。

[すべてエクスポート]/[エクスポートしない]: すべてのパラメータをエクスポート用に選択または削除できます。

[内部状態変数を出力ポートとして追加]: 選択したサブシステムの状態変数にポートを追加する場合は、このオプションを選択します。

Parameters:

Toggle Export Column

	Parameters	Value	Export	Change Row
1	C1_C	0.4e-3		
2	EMF1_fixed_phi0	0.		
3	EMF1_k	0.1e2		
4	I1_L	.76		
5	R1_R	14.1		
6	R1_T_ref	300.15		

[パラメータ]: モデルのパラメータです。

[フィルタ]: 特定のパラメータをフィルタします。

[すべて]/[エクスポート]: 表示を切り替えます。

[エクスポート]: 数式形式でエクスポートしたいパラメータを選択します。

[値]: システムパラメータの値を表示します。

[すべてエクスポート]/[エクスポートしない]: すべてのパラメータをエクスポート用に選択または削除できます。

サブシステムがロードされたら、個々の入力および出力変数の要素を vector 配列にグループ化したり、カスタマイズされたパラメータ値の入出力ポートを追加できます。入力ポートには変数の導関数を含めることができます。また、出力ポートにはサブシステムの状態変数を含めることができます。

注: パラメータにエクスポート用のマークが付けられていない場合、そのパラメータは数値で置換されます。

コード生成オプションの選択

[コードのエクスポートオプション] の設定により、コード生成プロセスの詳細オプションを指定します。

ソルバのオプション

このセクションでは、ソルバの種類を指定できます。

Fixed step solver: Euler RK2 RK3 RK4 Implicit Euler

拘束条件の計算オプション

[拘束条件の計算オプション] では、生成されたファイルで拘束条件に対する射影法を使用することで DAE で拘束条件を満たすかどうかを指定します。このオプションを使用すると、拘束条件を持つ DAE の精度が向上します。拘束条件が満たされていない場合、システムの結果は実際の解から外れる場合があり、指数関数的な割合でエラーが増加する可能性があります。

Max projection iterations:

Error tolerance:

Apply projection during event iterations

より正確な解を得るために射影を繰り返し実行できる最大回数を指定するには、[射影法における最大収束計算回数] を設定します。

射影後に得られる必要な許容誤差を指定するには、[許容誤差] を設定します。

より正確な解を得るために繰り返しを挿入するには、[イベント処理中の射影法の適用] を選択します。

MathWorks™ のウェブサイトでは説明されている External Model Interface の constraint projection ルーチンを使用して **拘束条件に対する射影** が実行され、DAE の結果のドリフトが制御されます。

Baumgarte 拘束安定化法

Baumgarte 拘束安定化法は、位置、速度、加速度の各拘束を次のように単一式にまとめて、位置拘束方程式を安定化します。この線形方程式を加速度に対して積分すると、Baumgarte パラメータ、Alpha および Beta が、この位置の拘束を安定化させます。

Baumgarte Constraint Stabilization:

Apply Baumgarte constraint stabilization

Export Baumgarte parameters

alpha

beta

[Baumgarte 拘束安定化法の適用]: モデルに Baumgarte 拘束安定化法を適用するときを選択します。

[Baumgarte パラメータのエクスポート]: 生成した C コードに Alpha および Beta の定数を含める場合を選択します。これにより、ソースコードで Alpha と Beta の値を変更することができるようになります。コードを再コンパイルして実行し、モデルに与える影響を確認してください。

[alpha]: モデルに適した微分ゲインの値を入力します。

[beta]: モデルに適した比例ゲインの値を入力します。

イベント処理のオプション

イベント処理のオプション セクションでは、生成されたファイルで拘束条件に射影法を使用することにより、DAE 系のイベント条件を満たすかどうかを指定します。このオプションを使用すると、イベントを持つ DAE の精度が向上します。拘束条件が満たされていない場合、システムの結果は実際の解から外れる場合があります。指数関数的な割合でエラーが増加する可能性があります。

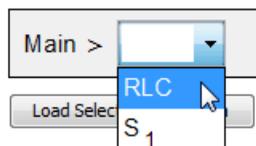
Event Handling Options:Max event iterations: Width of event hysteresis band:

より正確な解を得るために射影を繰り返し実行できる最大回数を指定するには、**[イベント処理の最大収束計算回数]**を設定します。

射影後に得られる必要な許容誤差を指定するには、**[イベントヒステリシスバンドの幅]**を設定します。

C コードの生成と保存**C コードを生成するには**

1. MapleSim で、コードを生成するモデルを開きます。
2. **モデルワークスペース**で、コードを生成してサブシステムにグループ化したいコンポーネントを確認します。
3. **[Apps とテンプレートを追加]** タブ () をクリックします。
4. **[Apps]** パレットで、**[コード生成]** をダブルクリックします。App が **[解析ウィンドウ]** の **Apps マネージャ** タブに開かれます。
5. ドロップダウンリストから、コードを生成するサブシステムを選択します。サブシステムの**選択ウィンドウ**に**サブシステムとその内容が表示されます**。



6. モデルダイアグラムの直下にある**[選択したサブシステムを読み込む]**をクリックします。サブシステムとすべての入出力変数がコード生成Appにロードされます。
7. 入力、出力、およびパラメータを設定します。
8. **[コードのエクスポートオプション]**で、ソルバを選択します。デフォルトでは Euler ソルバが選択されています。

9. コードの保存場所とファイル名を選択します。ファイルには、「c」という接頭辞と「c」という拡張子が自動的に追加されます。
10. **[Cコードを生成]**をクリックします。Cコードは指定した場所に保存されます。Cコードが生成されたら、Appの下部にある**[コードの表示]**セクションでコードを表示できます。

```

/*****
 * Automatically generated by Maple.
 * Created On: Tue May 04 17:14:50 2021.
 *****/
#ifdef WMI_WINNT
#define EXP __declspec(dllexport)
#else
#ifdef X86_64_WINDOWS
#define EXP __declspec(dllexport)
#else
#define EXP
#endif
#endif
#include <stdlib.h>
#include <stdio.h>
#include <math.h>
#ifdef FROM_MAPLE
#include <mplshlib.h>
static MKernelVector kv;
EXP ALGEB_M_DECL SetKernelVector(MKernelVector kv_in, ALGEB args) { ((void)(args)); kv=kv_in; return(kv->toMapleNULL()); }
#include <string.h>
#else
#include <string.h>
#include <stdarg.h>
<

```

5.6. 外部 C コード/ライブラリ定義からカスタムコンポーネントを生成

MapleSimでは、モデル内から外部コードを直接呼び出すことができます。**外部 C コード/ライブラリ定義**テンプレートを使用すると、外部 C コードや DLL 関数をモデルやサブシステムに直接呼び出すためのカスタムコンポーネントを作成できます。基本的な C コードにアクセスし、それらのコードをコンパイルして Maple で実行することができます。このコードの拡張機能については、Connectorの追加のツールボックスとして、さまざまなソフトウェアツールで利用できます。

このテンプレートを使用すると、外部入出力の定義、関数名と引数の指定、ソースコードの生成、結果のコンポーネントおよびライブラリコードの書式の選択などを行うことができます。タスク解析の実行、モデル方程式の変数への代入、入出力のグループ化、変数の追加の入出力ポートの定義を実行するには、Maple コマンドを使用できます。

パラメータ、入力、出力に加えられた変更は記憶され、外部Cコード/ライブラリ定義テンプレートを使用してシステムに再読み込みされるときに使用されます。

MapleSim モデルに対して外部コードカスタムコンポーネントを作成するプロセスには、次のステップが含まれています。

- カスタムコンポーネントの指定
- 外部C/ライブラリの場所の指定
- 外部C/ライブラリコードオプションを定義する
- 生成する Modelica コードを保存するディレクトリの指定
- 外部コードカスタムコンポーネントの生成と保存

外部Cコード/ライブラリ定義テンプレートを開く

[Appsとテンプレートを追加] () をクリックし、[外部C/ライブラリブロック] テンプレートを選択します。[添付を作成] ウィンドウにテンプレートの名前を入力し、✓ をクリックします。Maple が起動し、テンプレートが開きます。

C/ライブラリコードの場所とオプションの指定

外部Cコード/ライブラリ定義テンプレートを使用して、ライブラリコードの場所、コードの検証とモデルへの割り当てを行います。ヘッダファイルを指定するか、既存のCファイルや共有ライブラリファイルを使用するか、またはテキスト領域を使用して新しいCファイルを作成することができます。

ヘッダファイルの指定

必要な場合は、[ヘッダファイルの指定] を選択し、[ヘッダファイルの保存場所] テキストボックスで既存のヘッダファイルの場所を指定します。

Provide External Code using Text Area Specify Header File

Attached C/DLL/SO f1.c Refresh Attachments List

Location of C/DLL File: Select File

Location of Header File: Select File

Validate and Save C/Library File Validate and Attach C/Library File to MapleSim Model

既存の C ファイルまたはライブラリファイルの使用

[C/DLL ファイルの保存場所] テキスト領域を使用して、既存の C ファイルまたはライブラリファイルの場所を指定します。

Provide External Code using Text Area Specify Header File

Attached C/DLL/SO

Location of C/DLL File:

Location of Header File:

指定した C ファイルやライブラリファイルの有効性を検証するには、**[C/ライブラリファイルの検証と保存]** をクリックします。

指定した C ファイルやライブラリファイルの有効性を検証して MapleSim モデルに添付するには、**[C/ライブラリファイルを検証して MapleSim モデルに添付]** をクリックします。添付したファイルは、**[添付ファイル]** タブ (📎) の **[その他]** パレットに保存されます。

Text Area を使用した外部コードの指定

[Text Area を使用した外部コードの指定] を選択すると、テキスト領域が表示されます。テキスト領域に C コードを入力し、**[Save to C File]** テキストボックスでファイルを保存する場所を指定します。

Provide External Code using Text Area Specify Header File

Attached C/DLL/SO

Save to C File:

Location of Header File:

C ファイルの有効性を検証するには、[C/ライブラリファイルの検証と保存] をクリックします。ファイルは [Save to C File] テキストボックスで指定した場所に保存されます。

指定した C ファイルの有効性を検証して MapleSim モデルに添付するには、[C/ライブラリファイルを検証して MapleSim モデルに添付] をクリックします。添付ファイルは、[添付ファイル] タブ (U) の [その他] パレットに保存されます。

添付した C またはライブラリファイルの使用

C またはライブラリが添付されている場合、[添付された C/DLL/SO] を選択して、ドロップダウンリストから特定の C またはライブラリファイルを選択することができます。

Provide External Code using Text Area Specify Header File

Attached C/DLL/SO

Location of C/DLL File:

Location of Header File:

C/ライブラリコードの場所とオプションの定義

[設定] セクションで、外部 C/ライブラリ関数名の定義、外部 C/ライブラリプロトタイプの種類、パラメータ名の選択、データ型の選択、配列かどうかの指定、外部関数の入力または出力の指定などを行うことができます。指定した入出力は、カスタムコンポーネントの入出力になります。

必要に応じて、行 ID を指定して [パラメータを削除] をクリックすると、そのパラメータを削除することができます。

また、[テーブルをクリア] をクリックすると、パラメータテーブルをリセットすることができます。

Function Name:

External Function Name:

Arguments:

Parameter Name: Passed By Reference
 Data Type: Array?

	Parameter Name	Data Type	Change Row
1			

Row ID to Remove:

Output:

Specify whether the external C/Library function returns a value. If it does, specify the name of the return parameter and its datatype.

Return?

Return Name:
 Return Type:

外部コードカスタムコンポーネントの作成方法についての完全なチュートリアルは、チュートリアル 6: C コード/DLL カスタムコンポーネントテンプレートの使用 [240ページ]を参照してください。

C コードの生成と保存**カスタムコンポーネントを生成するには**

1. **[外部コード/ライブラリの場所]** で、生成された Modelica カスタムコンポーネントコードを保存する場所を指定します。

Target directory:

2. **[外部コードコンポーネントの生成]** をクリックします。カスタムコンポーネントが生成されると、そのコードがテンプレートの下部にあるソースの詳細に表示されます。また、テンプレートは **[添付ファイル]** タブ (📎) の **[ドキュメント]** セクションに自動的に保存されます。

5.7. MapleSim API と Maple コマンドの操作

Apps とテンプレートを使用したモデルの操作と解析に加え、Maple ワークシートでは MapleSim アプリケーションプログラミングインターフェース (API) を使用することができます。まず、LinkModel コマンドを使用して MapleSim モデルへのリンクを作成します。LinkModel コマンドは、MapleSim モデルにアクセスできる接続モジュールを返します。MapleSim API の詳細は、ヘルプページおよび **LinkModel** ヘルプページの例題セクションを参照してください。

Maple 内では、モデルを操作する Maple 機能を最大限に活用できます。任意の Maple パッケージ (および **DynamicSystems** を含む) のコマンドを使用して、プログラムによりモデルを操作することができます。

5.8. Maple の埋め込みコンポーネントを扱う

埋め込みコンポーネントとは、MapleSim モデルについて、アクションを表示、編集、作成したり、情報を表示したり、解析したりするために、Maple ワークシートやドキュメントに埋め込まれる単純なグラフィカルインターフェースの要素です。また、モデルプロパティに、スライダ、プロットなど、ほかの Maple 埋め込みコンポーネントを関連付けてカスタム解析ツールを作成することもできます。

たとえば、**DocumentTools** パッケージのコマンドを使用して、パラメータ値を表示および変更することができます。また、モデルまたはサブシステムの方程式を **MapleSim** パッケージのコマンドを使用して取得し、任意の入力関数を使用してモデルまたはサブシステムの動作を解析するためにモデルを **DynamicSystems** のオブジェクトとして扱うこともできます。埋め込みコンポーネントは、[コンポーネント] パレットを使用して挿入します。

ヒント: テンプレートにあらかじめ用意された解析ツールでは、Maple の埋め込みコンポーネントを使用しています。対話型のグラフィカルコンポーネントを介して Maple コードを操作することが可能です。各埋め込みコンポーネントに関連付けられているコードには、**MapleSim** および **DynamicSystems** を含む Maple パッケージのコマンドが使用されています。

埋め込みコンポーネントに関連付けられているコードを表示するには、Maple ワークシートで、いずれかのツールを右クリック (Macintosh では **[Control]** キー

を押しながらクリック)して、**[コンポーネントプロパティ]**を選択し、**[編集]**をクリックします。埋め込みコンポーネントに関する詳細は、Maple ヘルプシステムの **埋め込みコンポーネント** トピックを参照してください。

高度な解析タスクについての詳細は、**[ヘルプ]** > **[例題]** > **[ユーザガイドの例題]** > **[第 5 章]** メニューから **[Sliding Table]** の例題を開いてから、添付されている **AdvancedAnalysis.mw** ワークシートを開きます (MapleSim の **[添付ファイル]** タブ () にある **[ドキュメント]** を展開)。

第6章 MapleSim チュートリアル

MapleSim チュートリアルを使用すると、説明用のタスク、問題、およびベストプラクティス (最良の方法) を使用した例題に取り組むことで、MapleSim で使用できる主な機能、ツール、テンプレート、およびシステムについて学習することができます。これらの例題の多くは、[ヘルプ] > [例題] > [ユーザガイドの例題] メニューにあります。例題はユーザガイドで使用される順に並んでいます。

この章の内容は以下のとおりです。

- チュートリアル 1: ギアボックス付き DC Motor のモデリング [181ページ]
- チュートリアル 2: ケーブル張力コントローラのモデリング [189ページ]
- チュートリアル 3: 非線形ダンパのモデリング [194ページ]
- チュートリアル 4: 平面スライダクランク機構のモデリング [203ページ]
- チュートリアル 5: カスタムコンポーネントテンプレートの使用 [213ページ]
- チュートリアル 6: C コード/DLL カスタムコンポーネントテンプレートの使用 [240ページ]
- チュートリアル 7: 方程式の抽出 App の使用 [247ページ]
- チュートリアル 8: 油圧システムのモデリング [253ページ]

6.1. チュートリアル 1: ギアボックス付き DC Motor のモデリング

このチュートリアルでは、DC Motor モデルを拡張するために、以下のタスクを実行します。

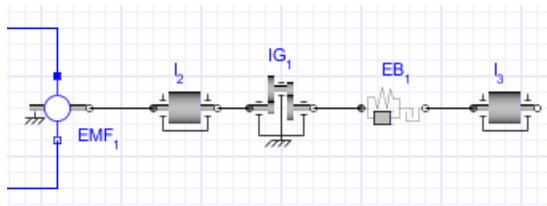
- DC Motor のモデルにギアボックスを追加する
- ギアボックス付き DC Motor のモデルをシミュレートする
- DC Motor コンポーネントをグループ化してサブシステムを作成する
- グローバルパラメータをモデルに代入する
- 信号ブロックコンポーネントと PI コントローラをモデルに追加する
- 変更した DC Motor モデルにさまざまな条件を与えてシミュレートする

DC Motor のモデルにギアボックスを追加する

この例では、1-D メカニカルライブラリから ideal gearbox コンポーネント、線形バネおよびダンパ付きの backlash コンポーネント、inertia コンポーネントを追加し、それらを接続してギアボックスを作成します。モデルワークスペースでコンポーネントをドラッグして配置するには、選択ツールを使用します。

ギアボックスを追加するには

1. [ヘルプ] メニューから [例題] > [ユーザガイドの例題] > [第 1 章] の順に選択し、[Simple DC Motor] の例を選択します。
2. ワークスペースから、既存のプローブを削除します。
3. [ライブラリコンポーネント] タブ(🔍) を選択し、以下のタスクを実行します。
 - [1-D メカニカル] > [回転] > [ベアリング・ギア] メニューから **Ideal Gear** コンポーネントをモデルワークスペースに追加し、**Inertia** コンポーネントの右側に配置します。
 - [1-D メカニカル] > [回転] > [バネ・ダンパ] メニューから **Elasto-Backlash** コンポーネントをモデルワークスペースに追加し、**Ideal Gear** コンポーネントの右側に配置します。
 - [1-D メカニカル] > [回転] > [共通] メニューから **Inertia** コンポーネントをもう 1 つモデルワークスペースに追加し、**Elasto-Backlash** コンポーネントの右側に配置します。
4. 以下の図のようにコンポーネントを接続します。



5. モデルワークスペースで、**Ideal Gear** コンポーネントをクリックします。
6. [プロパティ] タブ(📄) で [Transmission ratio] (速度伝達比) [r] を「10」に変更し、[Enter] キーを押して値を確定します。

7. ほかのコンポーネントに以下のパラメータ値を指定します。

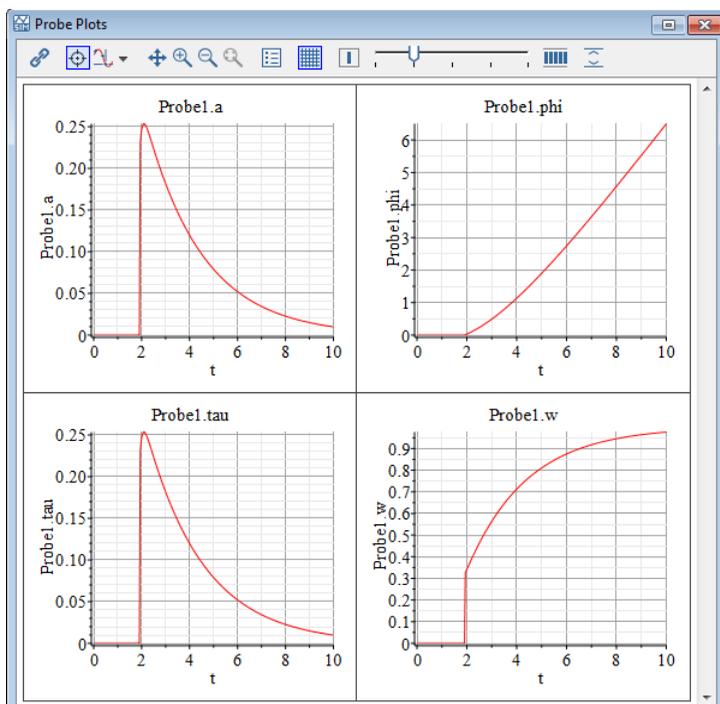
- **Elasto-Backlash** コンポーネントの [**b**] フィールドで、総合バックラッシュの値を **0.3rad** に変更します。 [**d**] フィールドで、減衰係数を $10^4 \frac{N \cdot m \cdot s}{rad}$ に変更します。
- 1 つ目の **Inertia** コンポーネント (I_2) の [**J**] フィールドで、慣性モーメントの値を $10kg \cdot m^2$ に変更します。
- 2 つ目の **Inertia** コンポーネント (I_3) の [**J**] フィールドで、慣性モーメントの値を $1kg \cdot m^2$ に変更します。
- **Step** ソースの [**height**] で、高さの値を「**100**」に変更します。

ギアボックス付き DC Motor モデルのシミュレーション

DC Motor モデルのシミュレーションを実行するには

1. モデルワークスペースツールバーの**プローブを追加** () をクリックします。
2. マウスポインタを **Elasto-Backlash** コンポーネントと 2 つ目の **Inertia** コンポーネント (I_3) を接続するライン上に動かします。ラインが強調表示されます。
3. ラインを 1 回クリックし、ワークスペース内の空白部分をクリックしてプローブを固定します。
4. **モデルワークスペース** で、プローブを選択します。
5. 角度 (ϕ)、速度 (w)、加速度 (a)、トルク (τ) の値をシミュレーショングラフに含めるために、**[プロパティ]** タブ () で、**[Angle]**、**[Speed]**、**[Acceleration]**、**[Torque]** を選択します。
6. **モデルワークスペース** で、空白部分をクリックします。
7. **[シミュレーションの設定]** タブ () で t_d パラメータを「**10**」秒に設定し、**[Enter]** キーを押します。
8. **メインツールバー** で**シミュレーションの実行** () をクリックします。

9. シミュレーション結果を表示 (📊) をクリックします。[解析ウィンドウ] に以下のグラフが表示されます。

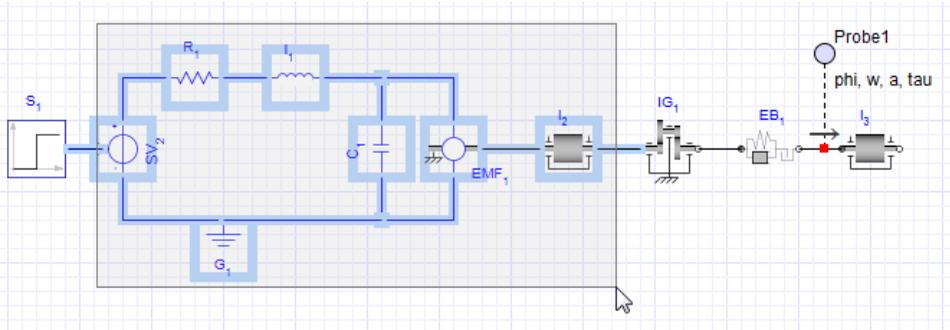


10. 結果を検証するには、[ヘルプ] メニューから [例題] > [ユーザガイドの例題] > [第6章] の順に展開し、[DC Motor with Gearbox] の例を選択します。

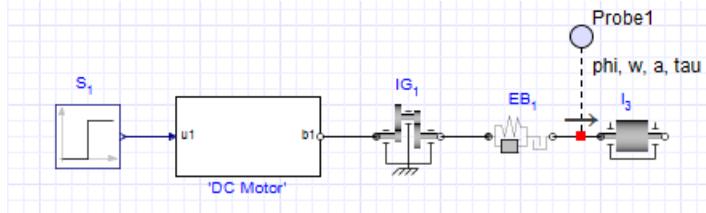
DC Motor コンポーネントをグループ化してサブシステムを作成する

DC Motor コンポーネントをグループ化するには

1. すべての電気コンポーネントと1つ目の inertia コンポーネントの上でマウスをドラッグして、これらを囲む四角形を描きます。



2. [編集] メニューから、[サブシステムの作成] を選択します。
3. [サブシステムの作成] ダイアログボックスで「DC Motor」と入力します。
4. [OK] をクリックします。DC モータを表す白いブロックがモデルワークスペースに表示されます。



ヒント: サブシステム内のコンポーネントを表示するには、**モデルワークスペース**で **DC Motor** サブシステムをダブルクリックします。モデルのトップレベルを表示するには、**モデルワークスペースツールバー**の **Main** (🏠) をクリックします。

グローバルパラメータをモデルに代入する

グローバルパラメータを定義し、その値を変数としてモデル内の複数のコンポーネントに代入することができます。

グローバルパラメータを代入するには

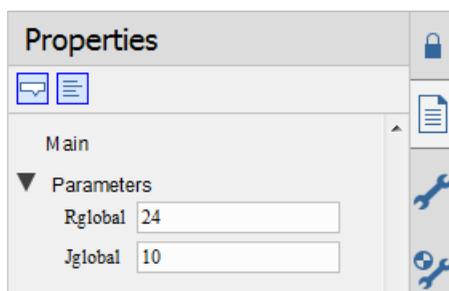
1. **モデルワークスペースツールバー**の **Main** (🏠) をクリックし、モデルのトップレベルを表示します。
2. **モデルワークスペースツールバー**の **パラメータ** (🔧) をクリックしてパラメータエディタ画面に切り替えます。

3. [Main サブシステム デフォルト設定] テーブルの 1 行目の [名前] フィールドに「Rglobal」と入力し、[Enter] キーを押します。
4. [デフォルト値] に「24」を指定し、[説明] として「Global resistance variable」を入力します。
5. テーブルの 2 行目の [名前] フィールドに「Jglobal」と入力し、[Enter] キーを押します。
6. [デフォルト値] に「10」を指定し、[説明] として「Global moment of inertia value」を入力します。

Main subsystem default settings

Name	Type	Default Value	Default Units	Description
Rglobal	Real	24		Global resistance value
Jglobal	Real	10		Global moment of inertia value

7. モデルワークスペースツールバーのダイアグラム表示 (📐) をクリックしてモデルダイアグラムに戻ります。新しい [Rglobal] と [Jglobal] パラメータが [プロパティ] タブ (📄) に表示されます。これで、これらのパラメータ値をモデル内のほかのコンポーネントに代入することができます。



8. モデルワークスペースツールバーのパラメータ (🔧) をクリックします。
9. [I_3 コンポーネント] テーブルで、[Moment of inertia] (慣性モーメント) パラメータの [値] フィールドに「Jglobal」と入力し、[Enter] キーを押します。これで Moment of inertia パラメータは、グローバルパラメータ Jglobal の数値 (この例では 10) を受け継ぎます。
10. ダイアグラム表示 (📐) をクリックし、DC Motor サブシステムをダブルクリックします。

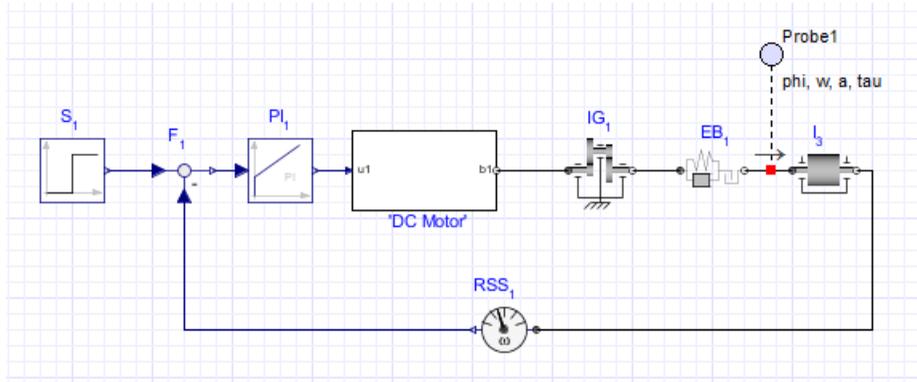
11. モデルワークスペースツールバーの**パラメータ** () をクリックします。
 12. [EMF₁ コンポーネント] テーブルで、[Transformation coefficient] (変換係数) パラメータ [k] の [値] フィールドに「Rglobal·Jglobal」と入力し、[Enter] キーを押します。
- 注:** この値は、変換係数の近似値です。
13. [R₁ コンポーネント] テーブルで、Resistance (抵抗) パラメータの [値] フィールドに「Rglobal」と入力し、[Enter] キーを押します。
 14. **ダイアグラム表示** () をクリックしてから **Main** () をクリックし、モデルのトップレベルを表示します。
 15. モデルを「DC_Motor2.msim」という名前で保存します。

入出力値の変更

この例では、モデルの入出力値を変更し、さまざまな条件でシミュレーションを実行します。

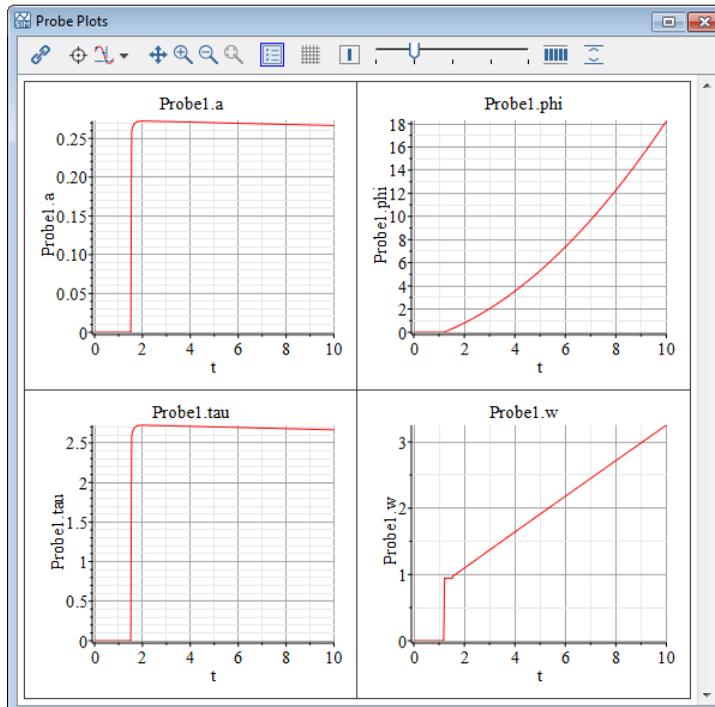
入出力値を変更するには

1. [ライブラリコンポーネント] タブ () から、[1-D メカニカル] > [回転] > [センサ] メニューを展開し、**Rotational Speed Sensor** コンポーネントをモデルワークスペースに追加して Gearbox コンポーネントの下に配置します。
2. **Rotational Speed Sensor** コンポーネントを右クリック (Macintosh の場合は [Control] キーを押しながらクリック) し、[水平方向に反転] を選択します。
3. **Step** ソースと **DC Motor** サブシステムのあいだの結線を削除します。
4. [信号ブロック] > [コントローラ] メニューから **PI** コンポーネントをモデルワークスペースに追加し、**DC Motor** サブシステムの左側に配置します。
5. [信号ブロック] > [数学] > [演算子] メニューから **Feedback** コンポーネントをモデルワークスペースに追加し、**PI** コンポーネントの左側に配置します。
6. 各コンポーネントを下図のように接続します。



垂線を描くには、結線を引いているときに、**モデルワークスペース**内の1点でクリックしてその点までの線分を確定し、そこからマウスカursorを別の方向に動かし、次の線分を描きます。

7. **モデルワークスペース**で、**PI** コンポーネントをクリックします。
8. [**プロパティ**]タブ(📄)で、[**k**]フィールドにgain(利得)の「**20**」を、[**T**]フィールドにtime constant(時定数)を「**3**」秒に指定します。
9. モデルのシミュレーションを再度実行します。シミュレーションが終了すると、下のグラフが表示されます。



10. モデルを「DC_Motor3.msimsim」という名前で保存します。

11. 結果を検証するには、[ヘルプ]メニューから[例題]>[ユーザガイドの例題]>[第6章]の順に展開し、[DC Motor Subsystem with Gearbox and PI Controller]の例を選択します。

6.2. チュートリアル 2 : ケーブル張力コントローラのモデリング

このチュートリアルでは、DC Motor の例を拡張して、あらかじめ定義された力で張られるケーブルをモデリングします。張力は **Constant** ソースで定義され、PIコントローラによって、モータを駆動する電圧が与えられます。ここでは、以下のタスクを実行します。

- ケーブル張力コントローラのモデルを作成する
- コンポーネントのプロパティを指定する
- ケーブル張力コントローラのモデルをシミュレートする

ケーブル張力コントローラのモデルを作成する

この例では、1-D メカニカルの回転および並進コンポーネントを使用して、ケーブル張力コントローラのモデルを作成します。また、コンポーネントをグループ化して **Gear** サブシステムを作成し、サブシステムにポートを追加します。

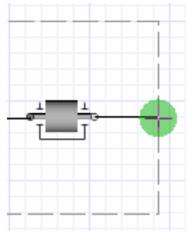
ケーブル張力コントローラを作成するには

1. 前のチュートリアルで作成した **DC_Motor3.msim** ファイルを開き、「**Cable_Tension.msim**」という名前で保存します。
2. **Elasto-Backlash** コンポーネントと **Inertia** コンポーネントを接続するラインに追加されているプローブを削除します。
3. **Rotational Speed Sensor** コンポーネントとその結線を削除します。
4. **Ideal Gear**、**Elasto-Backlash**、**Inertia** コンポーネントを選択し、グループ化して、サブシステム **Gear Components** にします。
5. [ライブラリコンポーネント] タブ () から、次のコンポーネントをモデルワークスペースに追加します。
 - [1-D メカニカル] > [回転] > [ベアリング・ギア] メニューから **Ideal Gear R 2 T** コンポーネントをモデルワークスペースに追加し、**Gear Components** サブシステムの右側に配置します。
 - [1-D メカニカル] > [並進] > [センサ] メニューから **Force Sensor** コンポーネントをモデルワークスペースに追加し、**Ideal Gear R 2 T** コンポーネントの右側に配置します。
 - [1-D メカニカル] > [並進] > [バネ・ダンパ] メニューから **Spring** コンポーネントをモデルワークスペースに追加し、**Force Sensor** コンポーネントの右側に配置します。
 - [1-D メカニカル] > [並進] > [共通] メニューから **Fixed** コンポーネントをモデルワークスペースに追加し、**Spring** コンポーネントの右側に配置します。
6. モデルワークスペース内の **Fixed** コンポーネントを右クリック (Macintosh の場合は [Control] キーを押しながらクリック) し、[反時計回りに回転] を選択します。

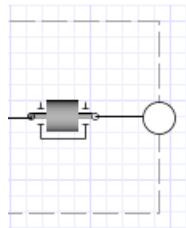
7. **Step** ソースを削除し、[ライブラリコンポーネント]>[信号ブロック]>[ソース]>[実数] メニューにある [Constant] ソースに置き換えます。

ヒント: Constant ソースは、未接続のラインエンドにドラッグすると、接続することができます。

8. **GearComponents** サブシステムをダブルクリックします。次に、このサブシステムを別のコンポーネントに接続するためのポートを追加します。
9. **Inertia** コンポーネントのマイナス (白) フランジをクリックし、マウスカursorをサブシステムのコンポーネントを囲む境界線にドラッグします。

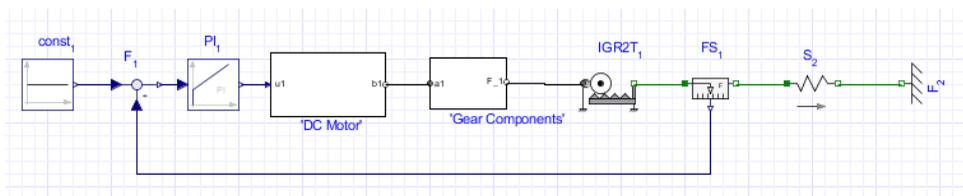


10. ラインを1回クリックします。サブシステムのポートがライン上に追加されます。



11. モデルワークスペースツールバーの **Main** (🏠) をクリックし、モデルのトップレベルを表示します。

12. 各コンポーネントを下図のように接続します。



コンポーネントプロパティの指定

コンポーネントプロパティを指定するには

1. モデルワークスペース内の **Gear Components** サブシステムをダブルクリックします。
2. [プロパティ] タブ () で、サブシステムコンポーネントに以下のパラメータ値を指定します。
 - **Ideal Gear** コンポーネントで、[r] を 「0.01」 に変更します。
 - **Inertia** コンポーネントで、[J] を 「0.1」 $kg \cdot m^2$ に変更します。
3. モデルワークスペースツールバーの **Main** () をクリックし、モデルのトップレベルを表示します。
4. ほかのコンポーネントに以下のパラメータ値を指定します。
 - **Spring** コンポーネントの [c] フィールドで、[Spring constant] (バネ定数) の値を $2110 \cdot 10^9 \frac{N}{m}$ に変更します。
 - **PI** コントローラで、[T] の値を **0.1s** に変更します。
 - **Constant** ソースの [k] フィールドで、[Constant output (一定出力)] の値を 「77.448」 に変更します。

ケーブル張力コントローラのシミュレーション

ケーブル張力コントローラをシミュレートするには

1. **プローブを追加** () をクリックします。
2. **Feedback** コンポーネントと **PI** コンポーネントを接続するラインをクリックします。ワークスペースをクリックしてプローブを配置します。
3. モデルワークスペースで、プローブを選択します。
4. [プロパティ] タブ () で物理量 [Real] を選択し、名前を 「Error」 に変更します。

5. 物理量 **Real** を計測するプローブをもう 1 つ、**PI** コンポーネントと **DC Motor** サブシステムを接続するラインに追加します。物理量の名前を「**Controller**」に変更します。

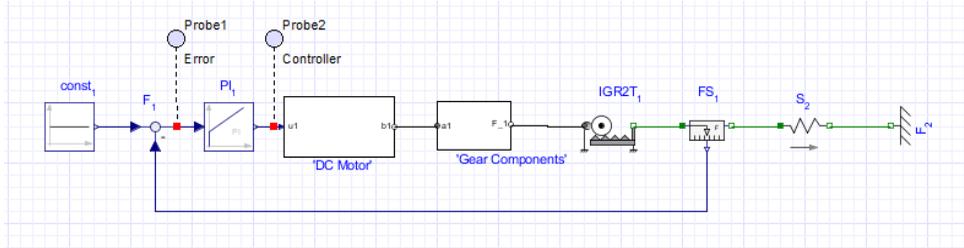
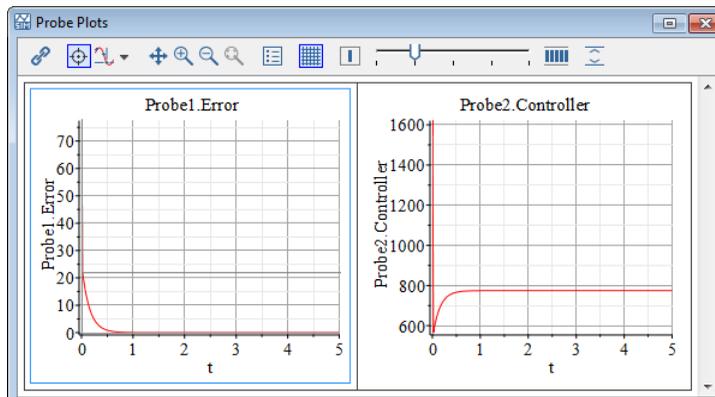


図6.1 ケーブル張力コントローラ

6. モデルワークスペースで、空白部分をクリックします。
7. [シミュレーションの設定] タブ (🔧) で、以下のパラメータを指定します。
- シミュレーション時間 [t_d] を「5」sに設定します。
 - [Solver Type] ドロップダウンメニューから [Variable] を選択します。
 - [Solver] ドロップダウンメニューから [Rosenbrock (stiff)] を選択します。
8. メインツールバーでシミュレーションの実行 (▶) をクリックします。
9. シミュレーション結果を表示 (📊) をクリックします。[解析ウィンドウ] に以下のグラフが表示されます。



10. ファイルを保存します。

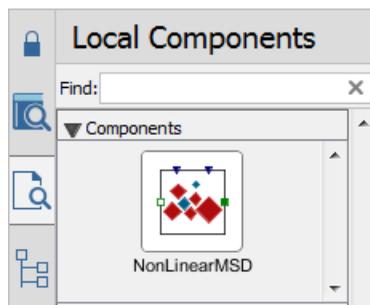
6.3. チュートリアル 3: 非線形ダンパのモデリング

このチュートリアルでは、線形バネ付きの非線形ダンパをモデリングします。このチュートリアルは、前のチュートリアルで説明した概念をベースにしています。ここでは、以下のタスクを実行します。

- 微分方程式で定義されたカスタムのバネダンパを生成する
- 入力信号としてカスタムの damping coefficient (減衰係数) 値を与える
- 線形バネ付き非線形ダンパのモデルを作成する
- 変数をサブシステムに代入する
- 線形バネ付き非線形ダンパのモデルをシミュレートする

バネダンパカスタムコンポーネントの生成

この例は、第3章の *例: 非線形バネダンパカスタムコンポーネントの作成* [97ページ] で作成された非線形バネダンパカスタムコンポーネントを使用しています。



減衰係数値を与える

モデルに追加した補間テーブルコンポーネントにカスタム値を与えることができます。この例では、外部ファイルで減衰係数値を与えます。

減衰係数値を作成するには

1. Microsoft Excel スプレッドシート (.xlsx) またはカンマ区切り (CSV) ファイル (*.csv) のいずれかを作成して、下の値を入力します。

	A	B
1	0	750
2	0.05	500
3	0.1	250
4	0.2	75
5	0.25	250
6	0.3	650

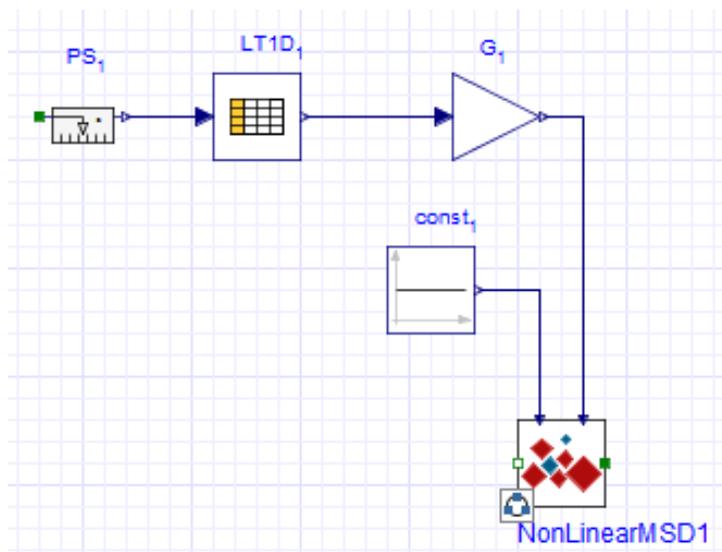
- 1 列目はダンパの相対変位で、2 列目は減衰係数値です。
2. ファイルを「**DamperCurve.xlsx**」または「**DamperCurve.csv**」という名前で保存します。
3. 第 3 章の *例: 非線形バネダンパカスタムコンポーネントの作成 [97ページ]* で作成した **NonlinearSpringDamper.msimsim** モデルを MapleSim で開きます。
4. [添付ファイル] タブ () を選択します。
5. [データセット] を右クリック (Macintosh の場合は [Control] キーを押しながらクリック) し、[ファイルを添付] を選択します。
6. 前に作成した Excel スプレッドシートまたは .csv ファイルを探して選択し、**添付する** をクリックします。これで、データセットを含むファイルがモデルに添付されます。このファイルは次のタスクで使用します。

非線形ダンパモデルの作成

この例では、コンポーネントライブラリのコンポーネントを使用して非線形ダンパを作成します。

非線形ダンパを作成するには

1. [ローカルコンポーネント] タブ (🔍) を選択し、**NonLinearMSD** カスタムコンポーネントをモデルワークスペースにドラッグします。
2. [ライブラリコンポーネント] タブ (🔍) を選択し、次のコンポーネントをモデルワークスペースに追加します。
 - [信号ブロック] > [数学] > [演算子] メニューから **Gain** コンポーネントを追加し、**NonLinearMSD** コンポーネントの上に配置します。
 - [信号ブロック] > [ソース] > [実数] メニューから **Constant** コンポーネントを追加し、**NonLinearMSD** と **Gain** コンポーネントのあいだに配置します。
 - [信号ブロック] > [補間テーブル] メニューから **Lookup Table 1D** コンポーネントを追加し、**Gain** コンポーネントの左側に配置します。
 - [1-D メカニカル] > [並進] > [センサ] メニューから **Position Sensor** コンポーネントをモデルワークスペースに追加し、**Lookup Table 1D** コンポーネントの左側に配置します。
3. 以下の図のようにコンポーネントを接続します。



4. モデルワークスペースに以下のコンポーネントを追加します。

- [1-D メカニカル] > [並進] > [共通] メニューから **Mass** コンポーネントと **Force** コンポーネントをモデルワークスペースに追加し、**Position Sensor** コンポーネントの左側に配置します。
 - 同じメニューから、**Fixed** コンポーネントをモデルワークスペースに追加し、**NonLinearMSD** コンポーネントの右側に配置してから、**反時計回りに回転**させます。
 - [信号ブロック] > [ソース] > [実数] メニューから [Step] ソースを追加します。
5. 以下の図のようにコンポーネントを接続します。

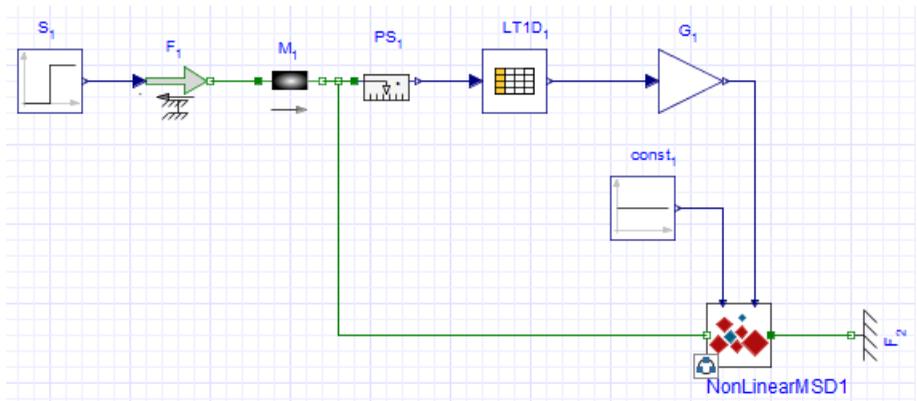
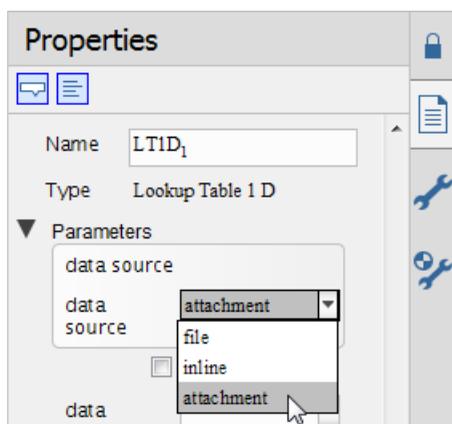
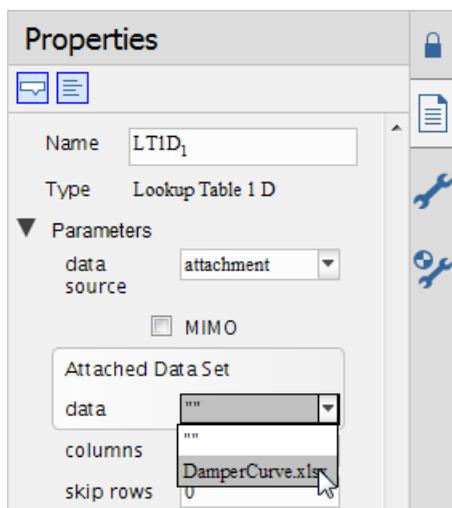


図6.2 非線形ダンパモデル

6. モデルワークスペースで、**Lookup Table 1D** コンポーネントを選択します。
7. [プロパティ] タブ (📄) で、[data source] リストから [attachment] を選択します。



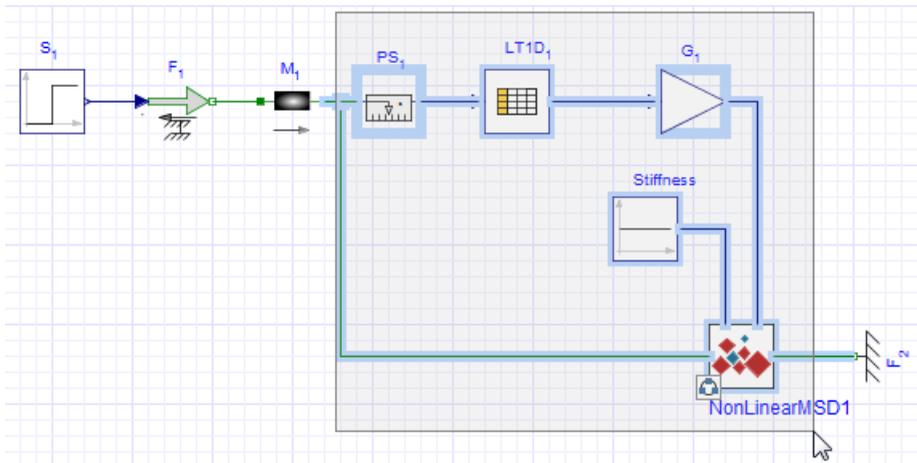
8. [data] リストから、作成した添付ファイルを選択します (**DamperCurve.xlsx** または **DamperCurve.csv** のいずれか)。



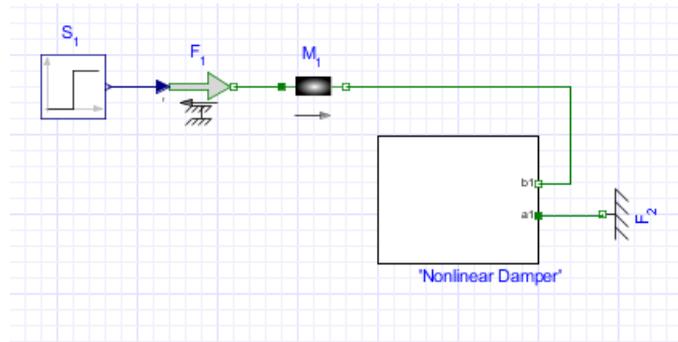
9. モデルワークスペースで、**Constant** コンポーネントを選択します。
10. [プロパティ] タブの [名前] フィールドで、コンポーネント名を「**Stiffness**」に変更します。



- 11 **Step** コンポーネントを選択し、[height] を「100」に変更します。
- 12 **Mass** コンポーネントを選択し、mass (質量) [m] を **100kg** に変更します。
- 13 非線形ダンパモデルの全コンポーネントを囲む四角形を描きます。



- 14 選択したコンポーネントをグループ化し、「**Nonlinear Damper**」という名前のサブシステムにします。完成したモデルを下の図に示します。



サブシステムにパラメータを代入する

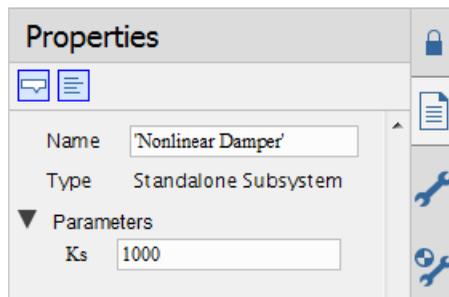
サブシステムにパラメータを代入するには

1. モデルワークスペースで、**Nonlinear Damper** サブシステムをダブルクリックします。
2. モデルワークスペースツールバーの**パラメータ** () をクリックします。
3. [スタンドアロンサブシステム デフォルト設定] テーブルの 1 行目に、「**Ks**」という名前のバネ定数パラメータを定義し、[Enter] キーを押します。
4. 同じ行の [デフォルト値] に「**1000**」を指定し、[説明] として「**Spring constant**」を入力します。これでパラメータ値 **Ks** を **Nonlinear Damper** サブシステムのほかのコンポーネントに代入することができます。

Standalone Subsystem default settings

Name	Type	Default Value	Default Units	Description
Ks	Real	1000		Spring constant

5. モデルワークスペースツールバーの**ダイアグラム表示** () をクリックします。**[Ks]** パラメータが **[プロパティ]** タブのフィールドとして表示され、設定してあるデフォルト値が入っています。



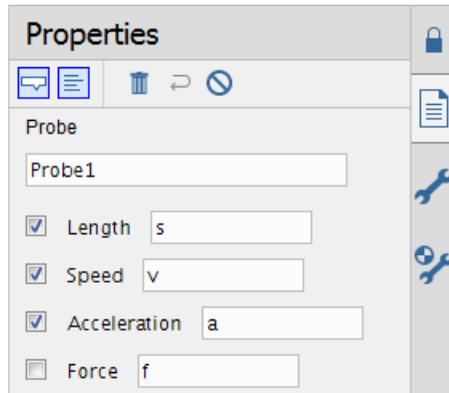
6. モデルワークスペースで **Stiffness** コンポーネントを選択し、一定出力のパラメータ **k** を **Ks** に変更します。これで、このコンポーネントは、**Ks** の数値 (この例では、**1000**) を受け継ぎます。つまり、サブシステムのレベルで **Ks** の数値を編集した場合には、**k** パラメータにもその変更が受け継がれます。



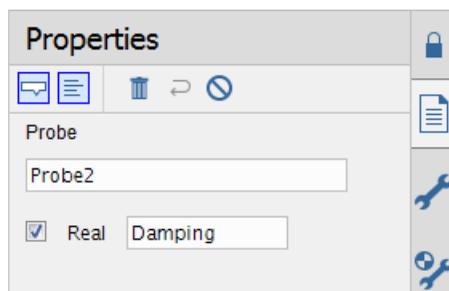
線形バネ付き非線形ダンパモデルのシミュレーション

非線形ダンパのシミュレーションを実行するには

1. モデルワークスペースツールバーの **Main** (🏠) をクリックし、モデルのトップレベルを表示します。
2. モデルワークスペースツールバーの **プローブを追加** (⊕) をクリックします。
ワークスペースにカーソルを移動すると、カーソルがプローブアイコンに変わります。
3. プローブを追加するには、**Mass** と **Nonlinear Damper** サブシステムを接続するラインをクリックし、ワークスペース内の空白部分をクリックしてプローブを固定します。
4. **モデルワークスペース** で、プローブを選択します。
5. [**プロパティ**] タブ (📄) で、length (変位)、speed (速度)、acceleration (加速度) の物理量を選択します。

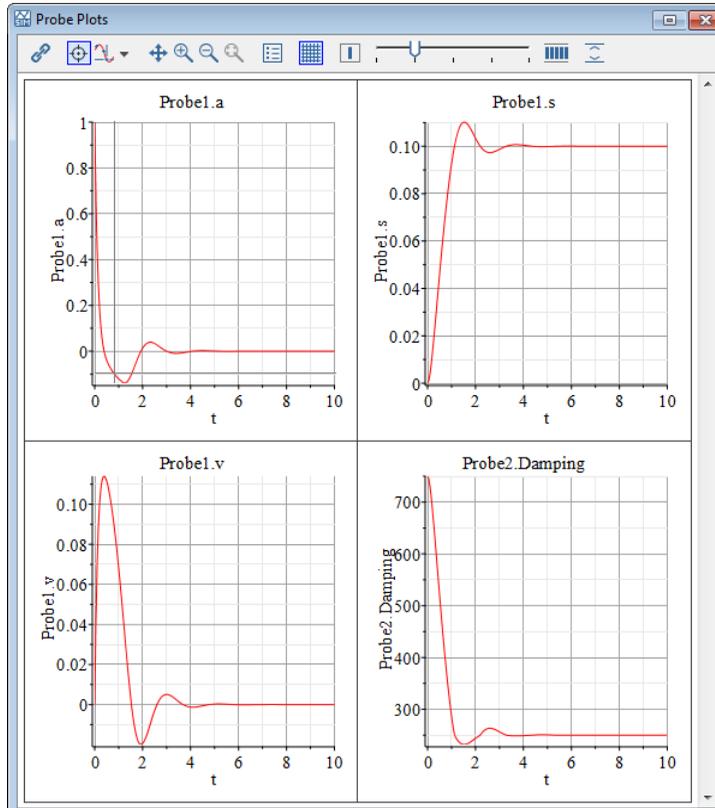


6. モデルワークスペースで、空白部分をクリックします。プローブの横にlength (変位)、speed (速度)、acceleration (加速度) の物理量 (s、v、a) が表示されます。
7. **Nonlinear Damper** サブシステムをダブルクリックします。
8. **Gain** と **NonLinearMSD** カスタムコンポーネントを接続するラインをクリックしてプローブを追加し、ワークスペース内の空白部分ををクリックしてプローブを固定します。
9. モデルワークスペースで、プローブを選択します。
10. [プロパティ] タブで物理量 [Real] を選択し、名前を「**Damping**」に変更します。



11. [シミュレーションの設定] タブ (🔧) で、 $[t_d]$ パラメータを「**10**」秒に設定します。
12. メインツールバーでシミュレーションの実行 (▶) をクリックします。

- 13 シミュレーション結果を表示 () をクリックします。[解析ウィンドウ] に以下のグラフが表示されます。



- 14 ファイルを「NonLinearMSD.msim」という名前で保存します。

6.4. チュートリアル 4: 平面スライダクランク機構のモデリング

マルチボディメカニカルライブラリのコンポーネントを使用して、下の図に示すような平面スライダクランク機構をモデリングします。

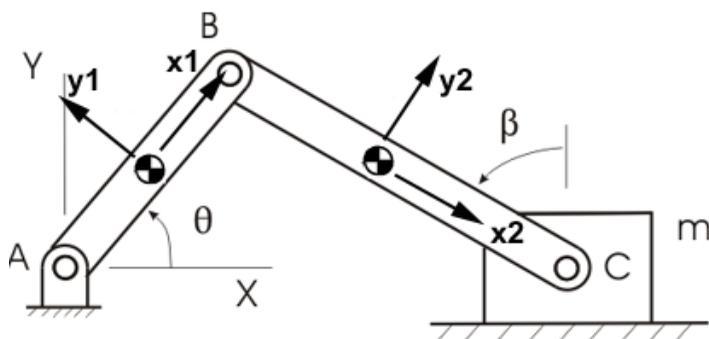


図6.3 Planar Slider-Crank Mechanism

このモデルは、平面リンクに接続された revolute joint (回転ジョイント) A で構成されています。この平面リンクは、2つ目の回転ジョイント B によってコネクティングロッドに接続されています。さらにコネクティングロッドは3つ目の回転ジョイント C によって sliding mass に接続されており、sliding mass は、prismatic joint (並進ジョイント) によって地面に接続されています。実際に、このメカニズムは、クランクにおける回転運動を sliding mass における並進運動に変換する、またはその逆に変換するために使用されます。ダイアグラムに示されているシステムでは、Y 軸 (慣性系の y 軸) のマイナス方向に働く重力を唯一の外力と考えます。

このチュートリアルでは、以下のタスクを実行します。

- 平面リンクサブシステムを作成する
- サブシステムパラメータを定義し、代入する
- クランクを作成し、ロッドエレメントと接続する
- 固定フレーム、sliding mass、ジョイントエレメントをモデルに追加する
- 初期条件を指定する
- 平面スライダクランク機構をシミュレートする

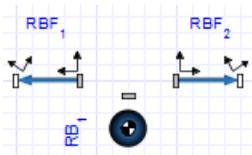
平面リンクサブシステムの作成

上のダイアグラムは、このスライダクランクには2つの連結された平面リンクがあることを示しています。1つ目はポイント A とポイント B を連結するクランク、

2つ目はポイント B とポイント C を連結するコネクティングロッドです。どちらにも、局所的な x 軸 (それぞれ x_1 、 x_2) に沿った縦軸があります。したがって、まず 2 つのポートを持つ一般的な平面リンクを作成します。内側のポート (ベース) は、リンクの x 軸に沿って、 $-\frac{L}{2}$ に位置し、外側のポート (先端) は、リンクの x 軸に沿って、 $\frac{L}{2}$ に位置します。この例では、 L はリンクの長さを表しており、重心はリンクの中央に位置すると考えます。

平面リンクサブシステムを作成するには

1. 新しい MapleSim ドキュメントを開きます。
2. [ライブラリコンポーネント] タブ () の [マルチボディ] > [ボディ・フレーム] メニューから、**Rigid Body Frame** コンポーネントを 2 つと **Rigid Body** コンポーネントを 1 つ追加します。
3. モデルワークスペースで、いずれかの **Rigid Body Frame** コンポーネントを右クリック (Mac の場合は [Control] キーを押しながらクリック) し、[水平方向に反転] を選択します。
4. **Rigid Body** コンポーネントを右クリック (Macintosh の場合は [Control] キーを押しながらクリック) し、**反時計回りに回転** を選択します。
5. 各コンポーネントをドラッグし、下の図のような配置にします。

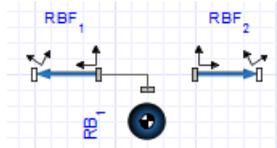


注:

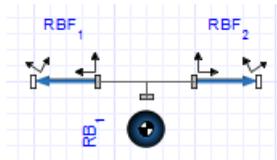
- コンポーネントのラベルが表示されていない場合は、[表示] メニューから [ラベルを表示] を選択してください。
- コンポーネントのラベルが前の図のラベル (つまり、 RB_1 、 RBF_1 、および RBF_2) と異なる場合があります。コンポーネントを選択し、[プロパティ] タブの [名前] フィールドに新しいラベルを入力することで、モデル内のラベ

ルを変更できます。このチュートリアルでは、特定のコンポーネントを参照する場合には前の図で示したラベルを使用します。

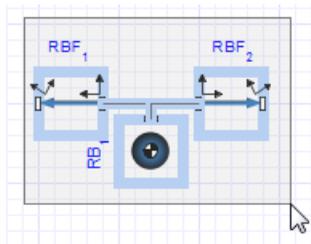
6. RB_1 コンポーネントと RBF_1 コンポーネントの右フレームのあいだに結線を引きます。



7. RB_1 コンポーネントと RBF_2 コンポーネントの左フレームのあいだに別の結線を引きます。



8. マウスをドラッグして、これら2つのコンポーネントを囲む四角形を描きます。



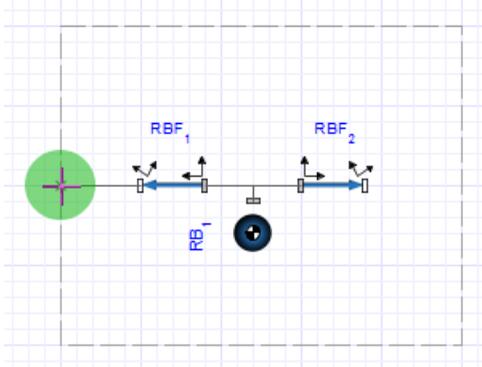
9. [編集] メニューから、[サブシステムの作成] を選択します。

10. [サブシステムの作成] ダイアログボックスで「Link」と入力し、[OK] をクリックします。

これで、このサブシステムをほかのコンポーネントに接続するためのポートを追加することができます。

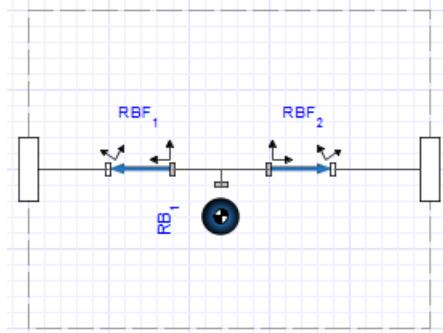
11. Link サブシステムをダブルクリックします。

- 12 **RBF₁** コンポーネントの左フレームをクリックし、マウスポインタをサブシステムの左側の境界にドラッグします。



- 13 ラインを 1 回クリックします。サブシステムにポートが追加されます。

- 14 同じように、**RBF₂** コンポーネントの右フレームを使用し、サブシステムの右側の境界にもう 1 つのポートを作成します。



パラメータの定義と代入

ここでは、リンクの長さを表すサブシステムのパラメータ L を定義し、そのパラメータ値を変数として **Rigid Body Frame** コンポーネントのパラメータに代入します。それによって **Rigid Body Frame** コンポーネントは、 L の数値を受け継ぎます。

パラメータを定義および代入するには

1. **Link**サブシステムにいない場合は、**モデルワークスペースツールバー**の**Main** (🏠) をクリックしてから **Link** サブシステムをダブルクリックします。
2. **モデルワークスペースツールバー**の**パラメータ** (🔧) をクリックするか、**[プロパティ]** タブ (📄) の **[パラメータを追加または変更]** をクリックします。**[スタンドアロンサブシステム デフォルト設定]** ウィンドウが表示されます。
3. **[スタンドアロンサブシステム デフォルト設定]** テーブルの1行目の**[名前]** フィールドに「**L**」と入力し、**[Enter]** キーを押します。
4. **[デフォルト値]** に「**1**」を指定し、**[説明]** として「**Length**」を入力します。
5. **RBF₁ コンポーネント**セクションまでスクロールします。
6. $\bar{\mathbf{r}}$ の**[値]** フィールドに、 $\left[-\frac{L}{2}, \mathbf{0}, \mathbf{0}\right]$ のオフセットの位置を指定し、**[単位]** ドロップダウンメニューから **[m]** を選択します。分数を入力するときは、スラッシュキー (/) を使用してください。
7. **RBF₂ コンポーネント**セクションまでスクロールします。
8. $\bar{\mathbf{r}}$ の**[値]** フィールドに、 $\left[\frac{L}{2}, \mathbf{0}, \mathbf{0}\right]$ のオフセットの位置を指定し、**[単位]** ドロップダウンメニューから **[m]** を選択します。
9. **ダイアグラム表示** (🖨) をクリックします。

クランクとコネクティングロッドエレメントの作成

ここでは、クランクとコネクティングロッドエレメントを作成するために、**Link**サブシステムの定義をモデルに追加し、**Crank** および **ConnectingRod** 共有サブシステムを作成します。さらに、コネクティングロッドエレメントに別の長さの値を代入します。

クランクを作成し、ロッドエレメントと接続するには

1. モデルワークスペースツールバーの **Main** (🏠) をクリックし、モデルのトップレベルを表示します。モデルワークスペースに **Link** サブシステムが表示されます。
2. **Link** サブシステムを右クリック (Macintosh の場合は **[Control]** キーを押しながらクリック) し、**[共有サブシステムに変換]** を選択します。**共有サブシステム** を作成ウィンドウが表示されます。**[OK]** をクリックします。**[ローカルコンポーネント]** タブ (🔍) の **[コンポーネント]** パレットに **Link** サブシステム定義が追加され、モデルワークスペースに配置されている **Link** サブシステムが共有サブシステムになります。
3. モデルワークスペースで、**Link₁** 共有サブシステムを選択します。**[プロパティ]** タブ (📄) の **[名前]** フィールドで、共有サブシステムの名前を「**Crank**」に変更します。
4. **[ローカルコンポーネント]** タブから **Link** アイコンをモデルワークスペースにドラッグし、**Crank** 共有サブシステムの右側に配置します。
5. モデルワークスペースで、**Link** 共有サブシステムの 2 つ目のコピーを選択します。
6. **[プロパティ]** タブ (📄) で、共有サブシステム名を「**ConnectingRod**」に変更します。下図を参照してください。



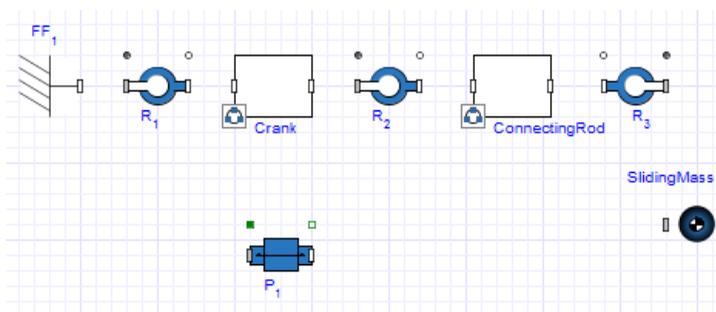
7. **ConnectingRod** の長さパラメータの値 (L) を「**2**」に変更します。

固定フレーム、Sliding Mass、ジョイントエレメントを追加する

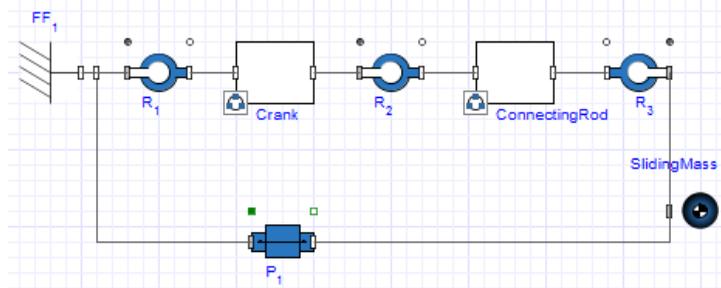
ここでは、**Fixed Frame** コンポーネント、sliding mass を表す **Rigid Body** コンポーネント、および **Revolute** ジョイントコンポーネントを追加します。

固定フレーム、Sliding Mass、ジョイントエレメントを追加するには

1. [ライブラリコンポーネント] タブ (🔍) の [マルチボディ] > [ボディ・フレーム] メニューから **Fixed Frame** コンポーネントを選択し、**Crank** 共有サブシステムの左側に配置します。
2. 同じメニューから **Rigid Body** コンポーネントを追加し、**Connecting Rod** 共有サブシステムサブシステムのやや右下に配置します。
3. 以下のジョイントを追加します。
 - [マルチボディ] > [ジョイント・モーション] メニューから、1つ目の [Revolute] ジョイントを **Fixed Frame** コンポーネントとクラークのあいだに、2つ目の [Revolute] ジョイントをクラークとコネクティングロッドのあいだに、3つ目の [Revolute] ジョイントをコネクティングロッドと剛体ボディのあいだに追加します。
 - 同じメニューから [Prismatic] ジョイントを追加し、**Crank** 共有サブシステムの下部に配置します。
4. モデルワークスペースで **Rigid Body** コンポーネントを選択し、名前を「**SlidingMass**」に変更します。
5. **SlidingMass** コンポーネントを右クリック (Macintoshの場合は [Control] キーを押しながらクリック) し、[水平方向に反転] を選択します。
6. 同様に、コネクティングロッドと剛体フレームのあいだにある回転ジョイントを右クリック (Mac の場合は **Control** キーを押しながらクリック) し、[水平方向に反転] を選択します。下図を参照してください。



7. 以下の図のようにコンポーネントを接続します。



ヒント:この例では、回転ジョイントおよび並進ジョイントの標準運動軸が、望ましい運動軸と一致しています。たとえば回転ジョイントは、内側のフレームのz軸の周りを回転しますが、それは常にXY平面システムの慣性のZ軸と一致しています。非平面モデルを作成する場合には、正しい方向に沿って動いたり回転できるように、軸を変更する必要がある場合があります。

初期条件の指定

モデルの特定のコンポーネントに初期条件の値を指定できます。

初期条件を指定するには

1. 1つ目の回転ジョイント (前の図の R_1) の θ_0 フィールドで、初期角度を $\frac{\pi}{4}$ radに変更します。

ヒント: π を入力するには、「pi」と入力し、[Esc] キーを押して、メニューから $[\pi]$ 記号を選択します。

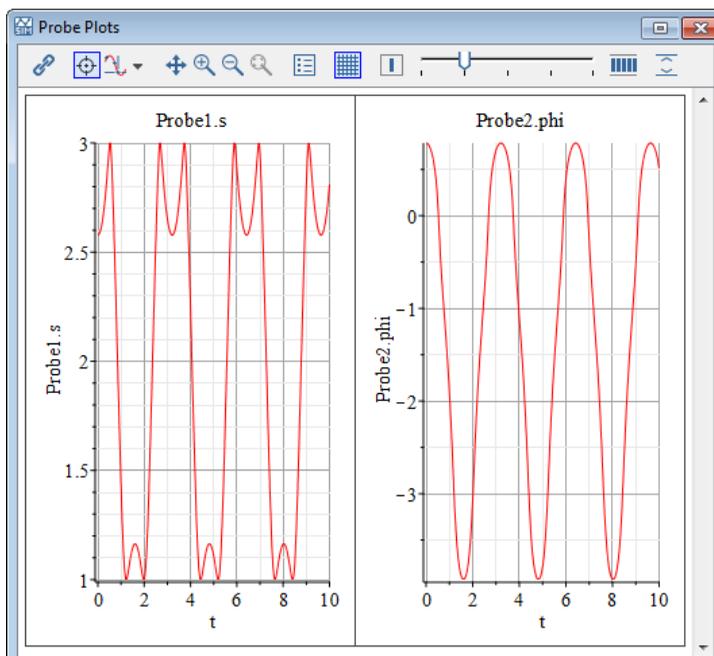
2. $[IC_{\theta, \omega}]$ のドロップダウンメニューから、[Strictly Enforce] を選択します。

MapleSim はこの初期条件を解決するにあたって、ほかのジョイントの角度を設定する前に、最初の角度を $\frac{\pi}{4}$ rad に設定します。

平面スライダクランク機構のシミュレーション

平面スライダクランク機構をシミュレートするには

1. モデルワークスペースツールバーの**プローブを追加** () をクリックします。
2. モデルワークスペースで、**Prismatic** コンポーネントのアイコンの右上にある白い1-D並進フランジをクリックします。次にプローブをクリックして配置します。
3. モデルワークスペースで、プローブをクリックします。
4. **[プロパティ]** タブ () で変位を測定するための物理量 **[Length]** を選択します。
5. 同様に、物理量 **Angle** を計測するプローブを、**R₁** コンポーネントアイコンの右上にある白い1-D回転フランジ (flange_b) (つまり、**Fixed Frame** と **Crank** コンポーネントのあいだにある回転ジョイント) に追加します。
6. モデルワークスペースで、空白部分をクリックします。
7. **[シミュレーションの設定]** タブ () で **[Simulation]** を展開し、**[t_d]** パラメータを「10」秒に設定します。
8. **メインツールバー**で**シミュレーションの実行** () をクリックします。
9. **シミュレーション結果を表示** () をクリックします。[解析ウィンドウ] に以下のグラフが表示されます。



10. [3-D アニメーションウィンドウ] を選択し、3-D ツールバーの再生 (▶) をクリックすると、シミュレーションに関するビデオを表示できます。

ヒント: [3-D アニメーションウィンドウ] の背後に開いているプロットウィンドウがある場合、可視化の質に影響します。アニメーションの再生で問題が発生する場合は、[3-D アニメーションウィンドウ] を移動してプロットウィンドウに重ならないようにしてください。あるいは、開いているプロットウィンドウを最小化するか閉じてください。

11. ファイルを「SliderCrank.msim」という名前で保存します。

6.5. チュートリアル 5: カスタムコンポーネントテンプレートの使用

このチュートリアルでは、MapleSimのさまざまなドメインでのカスタムコンポーネントテンプレートの使用について説明します。このテンプレートを使用すると、システムパラメータや変数の定義、方程式の最適化レベルの設定、方程式の生成、結果から生じたの式の解析を実行できます。Maple コマンドを使用する

と、方程式の詳細な解析を実行したり、モデルの方程式を変数またはパラメータに割り当てたり、追加のシステム変数やパラメータを定義できます。これらの機能は、複数のサブシステムが存在する場合に、再利用可能な方程式を生成するのに便利です。

カスタムコンポーネントテンプレートにはプリビルトの埋め込みコンポーネントが含まれています。これを使用して、MapleSim モデルによって生成された数式のシステム方程式を抽出、操作、および解析できます。ライブラリのさまざまなコンポーネントを使用して、モデルを作成したり、初期条件やコンポーネントプロパティを設定したり、パラメータや変数に新しい値を代入することができます。

このチュートリアルでは、カスタムコンポーネントテンプレートを使用して以下のタスクを実行し、さまざまなモデルの方程式を抽出します。

- モデルを作成する
- モデルにカスタムコンポーネントテンプレートを添付する
- 支配方程式を入力する
- コンポーネントのプロパティを指定して初期条件を設定する
- パラメータと変数に新しい値を代入する
- 方程式を表示、操作、再割り当てを行う
- 方程式のシミュレーションを行い伝達関数に変換する
- 方程式の変数をポートにマッピングする
- ブロックのポートを指定する
- カスタムポート App を使用してカスタムポートを作成する

カスタムコンポーネントテンプレートの説明については、*カスタムコンポーネントの作成 [83ページ]*を参照してください。

例: 温度依存抵抗のモデリング

このチュートリアルの例では、抵抗が $r(t) = R0 \cdot \left(1 + \frac{(T(t) - T0)^2}{Tk^2} \right)$ の式に従って変化する温度依存抵抗のカスタムコンポーネントを使用して、回路のモデルを作成します。ここで $R0$ 、 $T0$ 、および Tk はパラメータです。

カスタムコンポーネントを作成するには

1. MapleSim モデルを新規作成し、**[Apps とテンプレートを追加]** タブ () を選択します。
2. **[テンプレート]** パレットの**[カスタムコンポーネント]** 項目をダブルクリックします。
3. **[添付を作成]** に「**TempResistor**」と入力し、チェックマーク () をクリックします。Maple のカスタムコンポーネントテンプレートがロードされます。
4. **[方程式の定義]** セクションで、次のシステム方程式を入力してコンポーネントを定義します。行の末尾で **[Enter]** キーを押します。

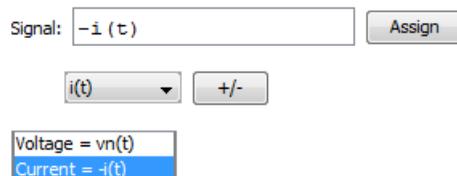
$$eq := \left[v(t) = vp(t) - vn(t), r(t) = R0 \cdot \left(1 + \frac{(T(t) - T0)^2}{Tk^2} \right), v(t) = i(t) \cdot r(t), \right. \\ \left. qdot(t) = i(t) \cdot v(t) \right];$$

5. **[設定]** セクションで**[パラメータ]** を選択し、**[すべて更新]** をクリックします。
6. **[設定]** セクションで**[ポート]** を選択します。
7. **[全ポートを削除]** をクリックします。
8. **[ポートを追加]** をクリックします。左側に新規ポートが表示されます。これが正の電気ピンになります。
9. **[タイプ]** ドロップダウンリストから **[Electrical]** を選択します。
10. **[+]** と表記されている **[スタイル]** ラジオボタンをクリックします。
11. リストボックスで **[Voltage = unassigned]** を選択し、**[信号]** の下のドロップダウンリストで **[vp(t)]** を選択します。ポートの介在変数が割り当てられます。

- 12 リストボックスで **[Current = unassigned]** を選択し、**[信号]** の下のドロップダウンリストで **[i(t)]** を選択します。ポートの通過変数が割り当てられます。



- 13 **[ポートを追加]** をクリックします。右側に新規ポートが表示されます。これが負の電気ピンになります。
- 14 **[タイプ]** ドロップダウンボックスから **[Electrical]** を選択します。
- 15 **[-]** と表記されている **[スタイル]** ラジオボタンをクリックします。ポートスタイルが白抜ききの四角形に変わります。
- 16 リストボックスで **[Voltage = unassigned]** を選択し、**[信号]** の下のドロップダウンリストで **[vn(t)]** を選択します。ポートの介在変数が割り当てられます。
- 17 リストボックスで **[Current = unassigned]** を選択し、**[信号]** の下のドロップダウンリストで **[i(t)]** を選択します。次に **+/-** ボタンをクリックして信号の符号をマイナスに変更します。

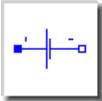
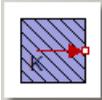


- 18 **[ポートを追加]** をクリックします。上辺に新規ポートが表示されます。ポートを底辺の中央までドラッグします。
- 19 **[タイプ]** ドロップダウンボックスから **[Thermal]** を選択します。
- 20 **[a]** と表記されている **[スタイル]** ラジオボタンをクリックします。
- 21 **Temperature** 変数に対して **[T(t)]** を選択し、**Heat Flow Rate** 変数に対して **[qdot(t)]** を選択します。
- 22 **[アイコン]** リストから、**[デフォルトを使用]** を選択します。

- 23 [設定] セクションで [変数] を選択し、[すべて更新] をクリックします。[変数] テーブルの [タイプ] 列の入力内容が更新されます。
 - 24 [変数] テーブルをスクロールして、変数 $r(t)$ と $v(t)$ を確認します。 $r(t)$ と $v(t)$ のタイプはまだ **real** としてリストされています。
 - 25 $r(t)$ と $v(t)$ の [タイプ] 列の入力内容をそれぞれ「**Resistance**」と「**Voltage**」に変更し、[すべて更新] をクリックして確定します (確定しない場合は **real** に戻ります)。
 - 26 [設定] セクションで [パラメータ] を選択します。
 - 27 [パラメータ] テーブルで、 R_0 のタイプ列に「**Resistance**」と入力し、 T_0 および T_k の両方に「**ThermodynamicTemperature**」と入力します。 T_0 の [デフォルト] 列に「**300**」と代入します (単位は Kelvin です)。[すべて更新] をクリックして確定します。
- 注:** [タイプの参照] セクションを展開して適切なドメインとタイプを検索することで、前のステップで使用したタイプ名を見つけることもできます。
- 28 [設定] セクションで [次元解析] を選択し、[次元のチェック] をクリックします。テキストエリアに「*noissuesfound*」というメッセージが表示されます。
 - 29 [コンポーネントの生成] セクションで、名前を「**TempResistor**」に変更します。
 - 30 [MapleSim コンポーネントを生成] をクリックしてコンポーネントを作成し、MapleSim 環境に戻ります。カスタムコンポーネントは、[ローカルコンポーネント] タブ () の [コンポーネント] パレットに表示されます。
 - 31 カスタムコンポーネントをモデル領域にドラッグします。
 - 32 表 6.1 「温度依存抵抗コンポーネント」 で指定されたコンポーネントおよび設定で、図 6.4 「温度依存抵抗」 に表示されたモデルを作成します。

注: モデルを作成するときは、電気および伝熱の物理量を測定するカスタムコンポーネントにプローブを追加してください。

温度依存抵抗コンポーネント

コンポーネント	記号	コンポーネントの場所	必要な設定
TempResistor カスタムコンポーネント		[ローカルコンポーネント] > [コンポーネント]	デフォルト設定を使用
Constant Voltage		[ライブラリコンポーネント] > [電気] > [アナログ] > [共通]	デフォルト設定を使用
Ground		[ライブラリコンポーネント] > [電気] > [アナログ] > [共通]	デフォルト設定を使用
Thermal Resistor		[ライブラリコンポーネント] > [伝熱] > [伝熱]	$R = 10 \text{ K/W}$ に設定
Heat Capacitor		[ライブラリコンポーネント] > [伝熱] > [伝熱]	$C = 0.1 \text{ J/K}$ に設定 $T_0 = 280\text{K}$ に設定
Fixed Temperature		[ライブラリコンポーネント] > [伝熱] > [ソース]	$T = 298\text{K}$ に設定

33 TempResistor カスタムコンポーネントを右クリック (Macでは[Control] キーを押しながらクリック) して [プローブを追加] を選択し、次にワークスペース内をクリックしてプローブを配置します。下図を参照してください。

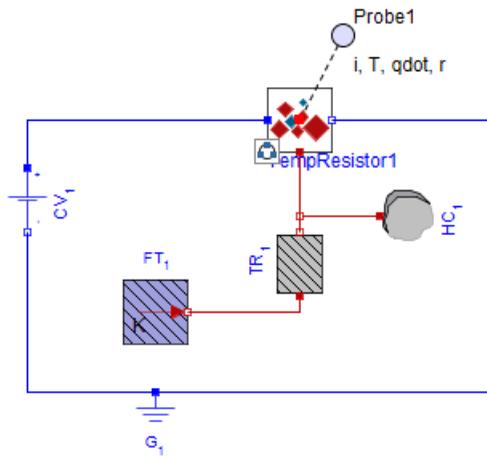
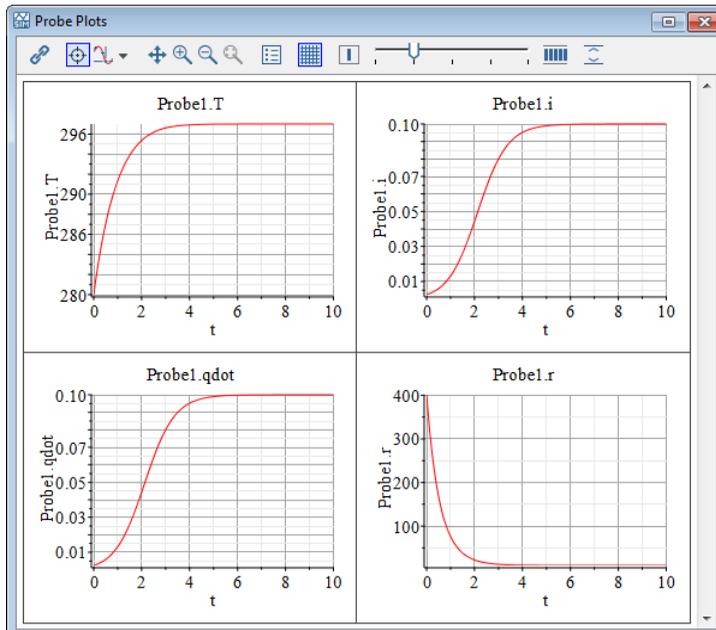


図6.4 温度依存抵抗

34. プローブを選択して【プロパティ】タブ () を選択し、以下の数量を選択します。
- Current
 - ThermodynamicTemperature
 - HeatFlowRate
 - Resistance
35. メインツールバーでシミュレーションの実行 () をクリックします。
36. シミュレーション結果を表示 () をクリックします。[解析ウィンドウ] に以下のグラフが表示されます。



例：コンプライアント接地と区分関数

このチュートリアル例では、コンプライアント接地をモデリングするためのカスタムコンポーネントを使用して、バウンドするボールのモデルを作成します。

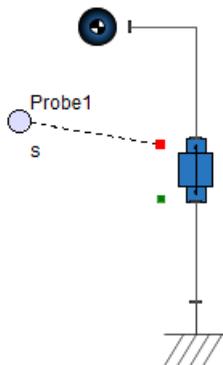


図6.5 落下するボール

図6.5 「落下するボール」 モデルの並進ジョイントは、縦の y 軸に沿って剛体を移動できるようにすることで、落下するボールをモデリングします。落下するボールをバウンドするボールに変更するには、カスタムコンポーネントを使用して、バネダンパ配置を使用したコンプライアントな地面との接地をモデリングします。カスタムコンポーネントは、次の条件によって並進ジョイントの 1-D 並進ポートに添付されます。

- ボールは $s=0$ で地面に当たり、これによってバネダンパが縮む(したがってボールの位置は $s < 0$ になる)
- バネダンパは、 $s=0$ に戻るまでボールに復元力 $F(t)$ を伝える

図6.6 「バウンドするボールの力学」 にこのプロセスの図を示します。

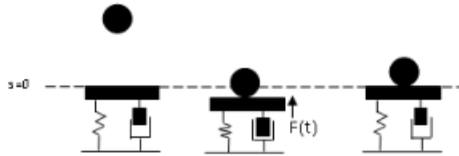


図6.6 バウンドするボールの力学

方程式は以下のとおりです。

$$f(x) = Ks(t) + B \frac{ds(t)}{dt}, \text{ if } s(t) < 0$$

$$f(x) = 0 \text{ if } s(t) \geq 0$$

カスタムコンポーネントを作成するには

1. MapleSim モデルを新規作成し、[Appsとテンプレートを追加] タブ () を選択します。
2. [テンプレート] パレットの [カスタムコンポーネント] 項目をダブルクリックします。
3. 添付名として「**contact**」と入力し、**添付を作成** () をクリックします。Maple の **カスタムコンポーネントテンプレート** がロードされます。

4. **[方程式の定義]** セクションで、次の方程式、パラメータ、初期条件を入力してカスタムコンポーネントを定義します。行の末尾で **[Enter]** キーを押します。

$$eq := \left[s(t) = sa(t) - sb(t), 0 = Fa(t) + Fb(t), Fa(t) = \text{piecewise}\left(s(t) < 0, K \cdot s(t) + B \cdot \frac{d}{dt} s(t), 0\right) \right];$$

5. **[設定]** セクションで **[ポート]** を選択します。
6. **[すべて更新]** をクリックします。
7. **[全ポートを削除]** をクリックします。
8. **[ポートを追加]** をクリックします。
9. 左側のポートの **[タイプ]** に **[Translational]** を選択します。スタイルは **[a]** (デフォルト) を使用します。 **Position** の変数に **[sa(t)]** を関連付け、 **Force** の変数に **[Fa(t)]** を関連付けます。
10. **[ポートを追加]** をクリックして 2 つ目のポートを右側に追加します。
11. 右側のポートの **[タイプ]** に **[Translational]** を選択します。スタイルは **[b]** を選択します。 **Position** の変数に **[sb(t)]** を関連付け、 **Force** の変数に **[Fb(t)]** を関連付けます。
12. **[アイコン]** リストから、 **[デフォルトを使用]** を選択します。
13. **[すべて更新]** をクリックします。
14. **[設定]** セクションで **[変数]** を選択します。
15. **[すべて更新]** をクリックします。
16. このテーブルで、 **[s(t)]** の **[タイプ]** を **「Position」** に変更し、 **[すべて更新]** をクリックします。
17. **[設定]** セクションで **[パラメータ]** を選択し、 **[すべて更新]** をクリックします。
18. **[B]** の **[デフォルト]** 値を **「10」** に変更します。
19. **[K]** の **[デフォルト]** 値を **「1000」** に変更し、 **[タイプ]** を **real** に変更します。
20. **[すべて更新]** をクリックします。

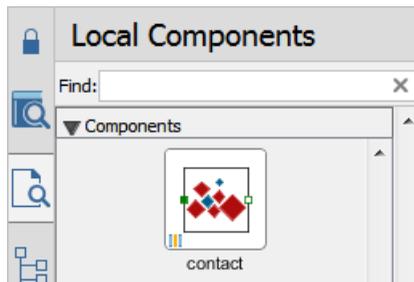
- 21 [設定] セクションで [次元解析] を選択し、[次元のチェック] をクリックします。次の式が表示されます。

$$B \frac{d}{dt} s(t) \frac{\text{m}^3 \text{kg}}{\text{s}^2 \text{A}} + K s(t) \text{ m}$$

これは、単位 (次元) に矛盾があることを示しています。総和において各要素に関連付けられている単位が表示されます。

MapleSim のエンジンは単位を使用しないため、この矛盾は無視することが可能で、モデルは意図したとおりに動作します。それでも、次元の整合性をチェックすることにより、単純な代数エラーを簡単に回避することができます。矛盾を解消するには、パラメータ列の B と K に適切なタイプを追加する必要があります。

- 22 [パラメータ] を選択し、[B] と [K] のタイプ列にそれぞれ 「Force/Velocity」 および 「Force/Distance」 を入力します。
- 23 [すべて更新] をクリックします。タイプがそれぞれ同等の次元タイプに更新され、TranslationalDampingConstant および TranslationalSpringConstant となります。
- 24 [次元解析] を選択し、[次元のチェック] をクリックします。「No issues found」が表示されます。
- 25 コンポーネントの生成セクションで、名前を 「contact」 に変更します。
- 26 [MapleSim コンポーネントを生成] をクリックしてコンポーネントを作成し、MapleSim 環境に戻ります。カスタムコンポーネントは、[ローカルコンポーネント] タブ (🔍) の [コンポーネント] パレットに表示されます。



27. カスタムコンポーネントをワークスペースにドラッグし、指定されたモデルコンポーネントと表6.2 「バウンドするボールのマルチボディコンポーネント」 に示す設定を使用して、図6.7 「バウンドするボール」 に示すコンポーネント

を作成します。並進ジョイントがy方向に沿って平行移動することを確認します。

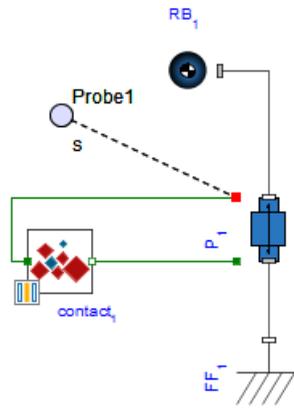
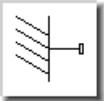
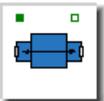


図6.7 バウンドするボール

バウンドするボールのマルチボディコンポーネント

コンポーネント	記号	コンポーネントの場所	必要な設定
カスタムコンポーネント		[ローカルコンポーネント]>[コンポーネント]	デフォルト設定を使用
Rigid Body		[ライブラリコンポーネント]>[マルチボディ]>[ボディ・フレーム]	デフォルト設定を使用
Fixed Frame		[ライブラリコンポーネント]>[マルチボディ]>[ボディ・フレーム]	デフォルト設定を使用
Prismatic		[ライブラリコンポーネント]>[マルチボディ]>[ジョイント・モーション]	<p>並進ジョイントを y 方向に沿って平行移動させるには、以下のように設定します。</p> <p>$\hat{\mathbf{e}}_1$ を [0, 1, 0]</p> <p>並進ジョイントに並進初期条件を以下のように設定します。</p> <p>$\mathbf{IC}_{s,v}$ を [Treat as Guess]</p> <p>並進ジョイントは初期変位を 0 より大きい値にする必要があります。以下のように設定します。</p> <p>s_0 を 10m</p>

28 メインツールバーでシミュレーションの実行 (▶) をクリックします。

29 シミュレーションが完了すると、[解析ウィンドウ] の [シミュレーション結果] タブにプローブプロットが表示されます。

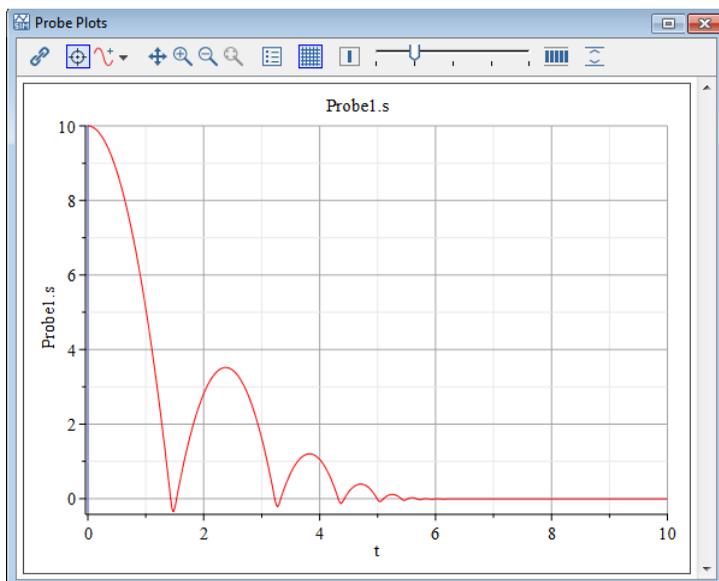


図6.8 バウンドするボールの結果

- 30 バウンドするボールのアニメーションを再生するには、[シミュレーション結果] タブの [3-D アニメーションウィンドウ] を選択してから、再生ツールバーの再生 (▶) をクリックします。
- 31 ツールバー上のナビゲーションコントロールを使用して、カメラをパン、ズーム、またはモデルの周囲を回るように移動し、最適な表示を探します。詳細は、3-D ツールバーを参照してください。
- 32 より滑らかなアニメーションを生成するには、3-D 設定の変更アイコン (⋮) をクリックし、ドロップダウンメニューから [フレーム補間] を選択します。詳細は、補間フレームを用いた 3-D モデルのアニメーションを参照してください。

例：カスタムポート

このチュートリアルの例では、ダミーの自動車モデルにより生成され、関係演算子 $f(t) = \frac{p(t)}{v(t)}$ を使用して力を計算するモニターブロックにより組み合わせられ

た因果的信号、パワーおよび速度を伴うカスタムポートのペアを作成します。ダミーの自動車モデルについて、ここではパワーが一定 $p(t) = P$ であり、速度は $v(t) = v_0 + a \cdot t$ によって求められ、それぞれ P 、 v_0 および a はパラメータであると仮定します。

カスタムポートの作成

カスタムポートを 2 つ作成する必要があります。1 つは出力ポート *bus_out*、もう 1 つは入力ポート *bus_in* です。

1. MapleSim モデルを新規作成し、**[Appsとテンプレートを追加]** タブ () を選択します。
2. **[Apps]** パレットの **[カスタムポート]** 項目をダブルクリックします。Maple の **Custom Port Definition App** がロードされます。
3. **[Configuration]** セクションで **[Signals]** を選択し、**[Add Signal]** をクリックして信号を追加します。
4. 信号の **[name]** に 「*p*」、**[type]** に 「*Power*」 と入力します。**[output]** ラジオボタンを選択し、**[desc]** 列に 「*Vehicle power*」 と入力します。
ヒント: タイプは [Type Reference] セクションから検索することもできます。タイプ名をクリックすると、**[type]** フィールドにコピーできます。
5. 2 つ目の信号を追加するには、**[Add Signal]** を再度クリックします。
6. 信号の **[name]** に 「*v*」、**[type]** に 「*Velocity*」 と入力します。**[output]** ラジオボタンを選択し、**[desc]** 列に 「*Vehicle velocity*」 と入力します。
7. **[Generate Port]** セクションで、**[Name]** フィールドに 「*bus_out*」 を、**[Description]** フィールドに 「*vehicle bus port*」 を入力し、**[output]** ラジオボタンを選択します。
8. **[Generate MapleSim Port]** をクリックして出力ポートを作成します。自動的に MapleSim モデルに戻ります。
9. 入力ポートを作成するため、同じ App に戻り修正を加えます。
10. **[Signals]** セクションで、両方の信号について **[input]** ラジオボタンを選択します。

- 11 **[Generate Port]** セクションで、*bus_out*を *bus_in*に変更し、**[input]** ラジオボタンをクリックします。
- 12 **[Generate MapleSim Port]** をクリックして入力ポートを作成します。自動的に MapleSim モデルに戻ります。
定義した2つのカスタムポートは、**[ローカルコンポーネント]** タブの**[コンポーネント]** パレットに表示されます。図6.9「カスタムポート」を参照してください。

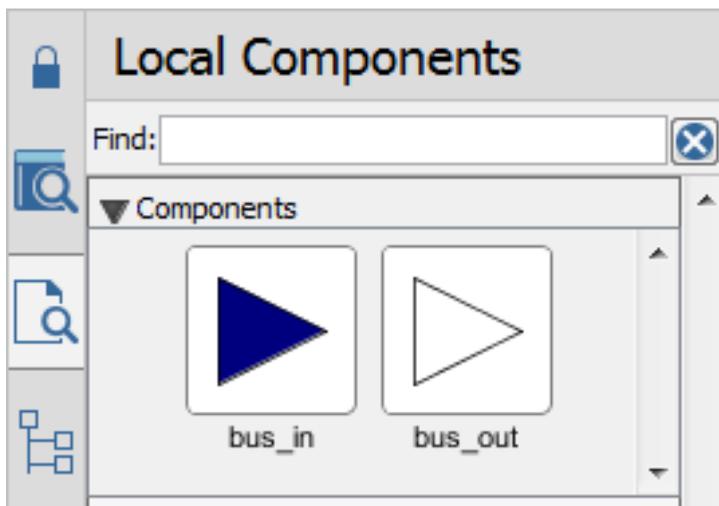


図6.9 カスタムポート

エンジンモデルの作成

最初にダミーの自動車モデルを作成します。

1. MapleSim で、**[Apps とテンプレートを追加]** タブ () を選択します。
2. **[テンプレート]** パレットの **[カスタムコンポーネント]** 項目をダブルクリックし、**[添付を作成]** のチェックマーク () をクリックします。Maple の **カスタムコンポーネント** テンプレートがロードされます。
3. **[方程式の定義]** セクションで、次のシステム方程式を入力してコンポーネントを定義し、行の末尾で **[Enter]** キーを押します。

$$eq := [p(t) = P, v(t) = v0 + a \cdot t]$$

4. [設定] セクションで [パラメータ] を選択し、[すべて更新] をクリックします。
5. P 、 a 、および v_0 の [タイプ] を「Power」、 「Acceleration」、および 「Velocity」 にそれぞれ変更します。
6. [設定] セクションで [ポート] を選択します。
7. [全ポートを削除] をクリックします。
8. [ポートを追加] をクリックします。左辺に新規ポートが表示されたら、右辺までドラッグします。
9. [タイプ] ドロップダウンメニューから、[Custom] を選択します (リストの一番最後の項目です)。
10. [タイプ] ドロップダウンメニュー下のテキストエリアに、「bus_out」と入力します。
11. スタイルのラジオボタン [out] を選択します。
12. [カスタムを適用] をクリックします。
13. [アイコン] リストから、[デフォルトを使用] を選択します。
14. このセクションの下部にあるドロップダウンメニューとリストボックスを使用して、Power 変数を $p(t)$ に、Velocity 変数を $v(t)$ に割り当てます。図6.10 「カスタムポートの使用」に、テンプレートの完成したポートセクションを示します。

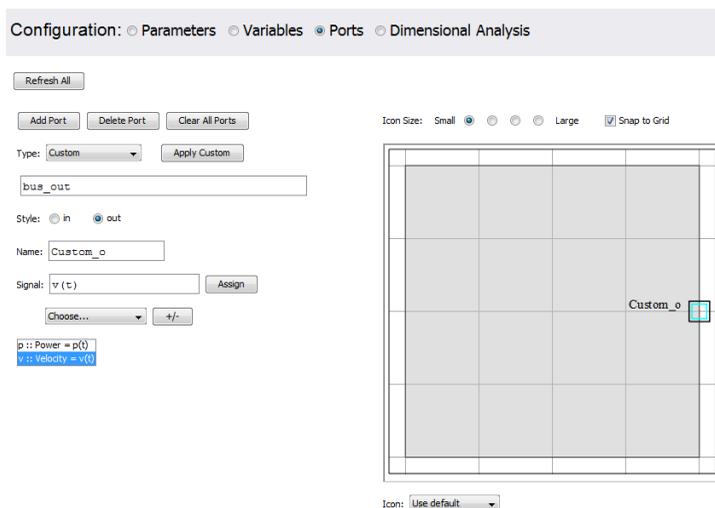


図6.10 カスタムポートの使用

- 15 [設定] セクションで [次元解析] を選択し、[次元のチェック] をクリックします。テキストエリアに 「noissuesfound」 というメッセージが表示されます。
- 16 [コンポーネントの生成] セクションで、名前を 「engine」 に変更してから [MapleSim コンポーネントを生成] をクリックします。エンジンコンポーネントが [ローカルコンポーネント] タブの [コンポーネント] パレットに表示されます。

モニターモデルの作成

1. MapleSim で、[Apps とテンプレートを追加] タブ () を選択します。
2. [テンプレート] パレットの [カスタムコンポーネント] 項目をダブルクリックし、[添付を作成] のチェックマーク () をクリックします。Maple のカスタムコンポーネントテンプレートがロードされます。
3. [方程式の定義] セクションで、次のシステム方程式を入力してコンポーネントを定義し、行の末尾で [Enter] キーを押します。

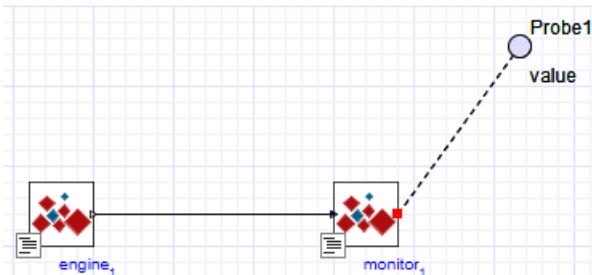
$$eq := \left[f(t) = \frac{p(t)}{v(t)} \right]$$

4. [設定] セクションで [パラメータ] を選択し、[すべて更新] をクリックします。

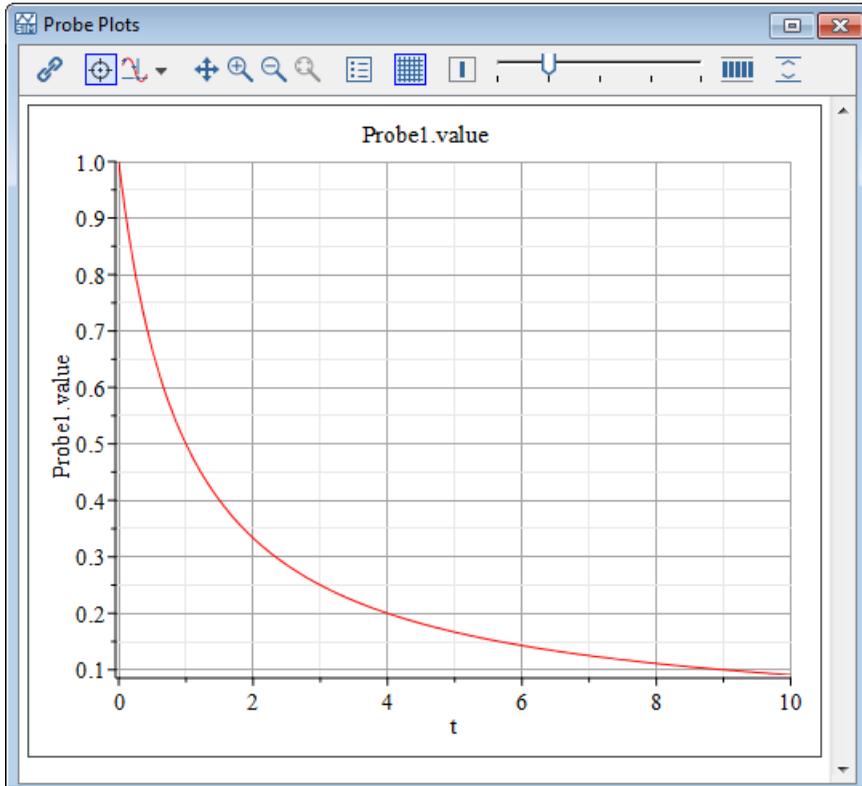
5. [ポート] を選択して、[全ポートを削除] をクリックします。
6. [ポートを追加] をクリックします。左辺に新規ポートが表示されたら、これがバス入力となります。
7. [タイプ] ドロップダウンメニューから、[Custom] (リストの一番最後の項目) を選択します。
8. [タイプ] ドロップダウンメニュー下のテキストエリアで、カスタムタイプの名前を「*bus_in*」と入力します。
9. [カスタムを適用] をクリックします。
10. このセクションの下部にあるドロップダウンメニューとリストボックスを使用して、Power 変数を $p(t)$ に、Velocity 変数を $v(t)$ に割り当てます。
11. [ポートを追加] をクリックします。右辺に新規ポートが表示され、これが計算された力の出力となります。
12. [タイプ] ドロップダウンメニューから [Real Signal] を選択します。スタイルのラジオボタン [out] をクリックします。信号の値に $f(t)$ を割り当てます。
13. [アイコン] リストから、[デフォルトを使用] を選択します。
14. [すべて更新] をクリックします。
15. [設定] セクションで [変数] を選択し、[すべて更新] をクリックします。
16. 信号 $f(t)$ の [タイプ] フィールドに「Force」と入力します。
17. [次元解析] を選択し、[次元のチェック] をクリックします。 $real_o(t) = f(t) N$ というメッセージが表示されます。これは出力ポート $real_o(t)$ が、実信号であるものの、力(ニュートン)の単位を持つ $f(t)$ と等しいことを示しています。
18. [コンポーネントの生成] セクションで、名前を「*monitor*」に変更します。
19. [MapleSim コンポーネントを生成] をクリックします。

モデルの完成

1. MapleSim で、**engine** と **monitor** コンポーネントを [ローカルコンポーネント] タブ (🔍) の [コンポーネント] パレットからワークスペースにドラッグします。
2. **engine** ブロックの出力を **monitor** ブロックの入力に接続します。次に、下図のように **monitor** ブロックの出力にプローブを追加します。



3. シミュレーションを実行します。[解析ウィンドウ] の [シミュレーション結果] タブに、以下の [プローブプロット] が表示されます。



カスタムコンポーネントの高度な使用方法

Mapleのすべての機能を使用して、カスタムコンポーネントテンプレートのシステム方程式を導くことができます。このセクションでは、高度なアプリケーションのサンプルを提供します。

例：水頭流量曲線からの遠心ポンプのモデリング

以下の油圧の例では、遠心ポンプからの外挿データをカスタムコンポーネントに適用する方法について説明します。遠心ポンプカスタムコンポーネントの作成には次のタスクが含まれます。

- グラフからデータを取得する
- データセットに最適な曲線を適用して方程式を生成する
- 複数の引数を持つ演算子を取得する

- 演算子を適用してカスタムコンポーネントを生成する

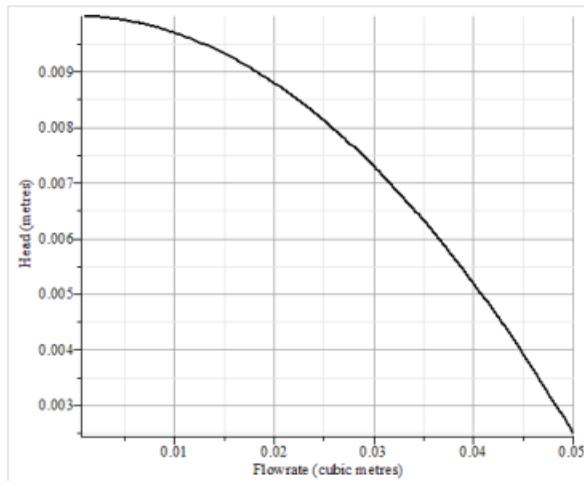


図6.11 遠心ポンプの水頭流量曲線

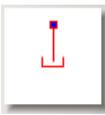
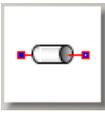
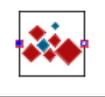
遠心ポンプのデータ

流量 (立法メートル)	圧力水頭 (メートル)
0.01	0.0098
0.02	0.00874
0.03	0.00725
0.04	0.005
0.05	0.0025

円形パイプのパラメータ

記号	説明	値
D	油圧配管直径	0.01 m
L	パイプの長さ	5 m
ϵ	内部パイプの凸凹の高さ	$1.5 \cdot 10^{-5}\text{ m}$
ReL	層流域の最大レイノルズ数	2000
ReT	乱流域の最小レイノルズ数	4000

遠心ポンプのコンポーネント

コンポーネント	記号	コンポーネントの場所	用途	必要な設定
Tank		[ライブラリコンポーネント] > [油圧] > [リファレンス]	このコンポーネントは基準の圧力(電気ドメインのグラウンドと同じ)を定義し、大気との接続を表します。	デフォルト設定を使用
Hydraulic Fluid Properties		[ライブラリコンポーネント] > [油圧] > [リファレンス]	すべての油圧モデルには Hydraulic Fluid Properties コンポーネントが必要です。このコンポーネントはパラメータブロックと同様に モデルワークスペース に配置され、次の油圧流体プロパティを定義します。 <ul style="list-style-type: none"> • rhoFluid : 液体の密度 • K : 流体の圧縮率を定義する体積弾性率 • nuFluid : 動粘度。絶対粘度を液体の密度で割って定義します。 	rhoFluid : $850 \frac{kg}{m^3}$ K : 8000 bar nuFluid : $0.000018 \frac{m^2}{s}$
Circular Pipe		[ライブラリコンポーネント] > [油圧] > [絞り]	円形パイプには油圧管路の圧力低下を定義します。圧力低下は Darcy の式によって求められます。	表6.4 「円形パイプのパラメータ」を参照
カスタムコンポーネント		[ローカルコンポーネント] > [コンポーネント]	このカスタムコンポーネントには、油圧管路の油圧と流量のプロパティを定義します。	ユーザ定義

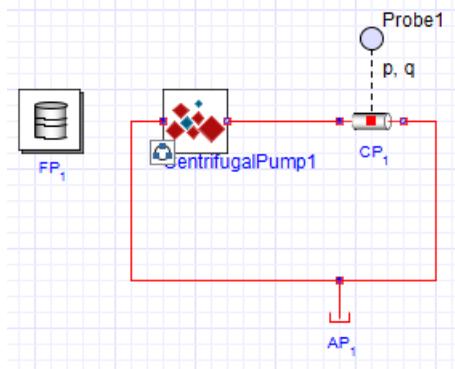


図6.12 遠心ポンプのカスタムコンポーネント

カスタムコンポーネントを作成するには

1. MapleSim モデルを新規作成し、[Appsとテンプレートを追加] タブ () を選択します。
2. [テンプレート] パレットの[カスタムコンポーネント]項目をダブルクリックします。
3. [添付を作成]のチェックマーク () をクリックします。Mapleのカスタムコンポーネントテンプレートがロードされます。
4. [方程式の定義] セクションで、最初の Maple コマンドライン (つまり、 eq を含むもの) にカーソルをあて、ドキュメントブロックを2つ挿入します (メインメニューから [編集] > [ドキュメントブロック] > [ドキュメントブロックを作成] の選択を2回繰り返します)。
5. 最初の行の中身を以下の Maple コマンドで置き換え、行の末尾で [Enter] キーを押します。このコマンドは、表6.3 「遠心ポンプのデータ」からの値をリスト L に配置します。

$$L := [[0.01, 0.0098], [0.02, 0.00874], [0.03, 0.00725], [0.04, 0.005], [0.05, 0.0025]]$$
6. 以下の Maple コマンドを2番目のドキュメントブロックに入力し、行の末尾で [Enter] キーを押します。このコマンドは、2次曲線をデータポイントに合わせます。

$f := \text{unapply}(\text{CurveFitting}[\text{LeastSquares}](L, x, \text{curve} = a \cdot x^2 + b \cdot x + c), x)$

7. 3番目のドキュメントブロックに以下のシステム方程式、パラメータ、初期条件を入力してコンポーネントを定義し、行の末尾で **[Enter]** キーを押します。このコマンドは、回帰曲線とブロックのパラメータに関する方程式を定義することで、カスタムコンポーネントに多項式を実装します。

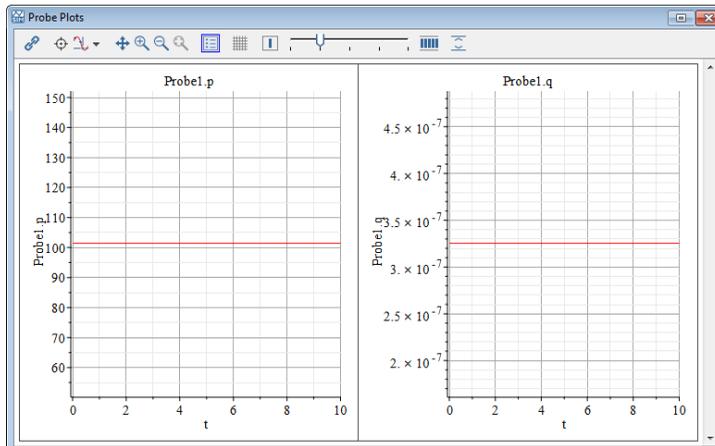
$eq := [P(t) = f(Q(t)) \cdot \rho \cdot g, P(t) = Pr(t) - Pl(t), \rho = 1000, g = 9.81]$

8. **[設定]** セクションで **[ポート]** を選択します。
9. **[全ポートを削除]** をクリックします。
10. **[ポートを追加]** をクリックして左辺にポートを追加します。 **[Hydraulic]** タイプを選択し、 **Pressure** 変数に $Pl(t)$ を、 **Volume Flow Rate** 変数に $Q(t)$ を割り当てます。
11. **[ポートを追加]** をクリックして右辺にポートを追加します。 **[Hydraulic]** タイプを選択し、 **[スタイルb]** をクリックし、 **Pressure** 変数に $Pr(t)$ を、 **Volume Flow Rate** 変数に $-Q(t)$ を割り当てます (+/- ボタンをクリックして記号を変更します)。
12. **[アイコン]** リストから、 **[デフォルトを使用]** を選択し、 **[すべて更新]** をクリックします。
13. **[設定]** セクションで **[パラメータ]** を選択し、 **[すべて更新]** をクリックします。
14. g のデフォルトを「9.81」に変更し、 [タイプ] に **Acceleration** を割り当てます。 ρ に対しては、 [デフォルト] を「1000」に変更し、 [タイプ] を **Density** とします。 **[すべて更新]** をクリックします。
15. **[変数]** を選択して、 **[すべて更新]** をクリックします。
16. $P(t)$ の [タイプ] を **Pressure** に変更します。
17. **[次元解析]** を選択し、 **[次元のチェック]** をクリックします。式

$\frac{517}{50000} - \frac{13 Q(t)^2}{5} \frac{\text{m}^6}{\text{s}^2} - \frac{137 Q(t)}{5000} \frac{\text{m}^3}{\text{s}}$ が表示されます。これは、次

元に矛盾が発生していることを示していますが、この矛盾は特に問題ではないため、無視することが可能です。この矛盾を除去したい場合は、元の数式の

- $Q(t)$ を $\frac{Q(t)}{Q1}$ に置き換え、パラメータ $Q1$ にデフォルト値 1 を割り当て、タイプを **VolumeFlowRate** として保護します ([**Protected**] 列に **X** を追加します)。
- 18 [コンポーネントの生成] セクションで、名前を「**CentrifugalPump**」に変更します。
 - 19 [MapleSim コンポーネントを生成] をクリックしてコンポーネントを作成し、MapleSim 環境に戻ります。カスタムコンポーネントは、[ローカルコンポーネント] タブ () の [コンポーネント] パレットに表示されます。
 - 20 カスタムコンポーネントをモデルワークスペースにドラッグし、指定されたモデルコンポーネントと表 6.5 「遠心ポンプのコンポーネント」に示す設定を使用して、図 6.12 「遠心ポンプのカスタムコンポーネント」に示すモデルを作成します。
- ヒント:** Circular Pipe コンポーネントにプローブを追加するには、コンポーネントを右クリック (Mac の場合は [**Control**] キーを押しながらクリック) し、[プローブを追加] を選択してからワークスペースをクリックし、プローブを配置します。
- 注:** 出力の圧力と体積流量の物理量を表示するには、プローブを選択し、[プロパティ] タブで [**Pressure**] および [**VolumeFlowRate**] の物理量を選択します。
- 21 メインツールバーでシミュレーションの実行 () をクリックします。
 - 22 シミュレーション結果を表示 () をクリックします。[解析ウィンドウ] に以下のグラフが表示されます。



6.6. チュートリアル 6:C コード/DLL カスタムコンポーネントテンプレートの使用

このチュートリアルでは、**外部C/ライブラリブロック**テンプレートを使用して、外部Cコードパラメータをインポートし、以下のタスクを実行してモデルを作成します。

- カスタムコンポーネント名を指定する
- 外部C/ライブラリの場所を指定する
- 外部C/ライブラリコードオプションを定義する
- 生成される Modelica コードのディレクトリを指定する
- 外部コードカスタムコンポーネントを生成し、保存する
- シンプルな外部関数モデルを作成する

このモデルは、**Step** 関数、**Constant Vector**、および**外部Cコード/DLL**の3つのコンポーネントで構成されます。

外部Cコードパラメータは、以下の入力を取り込む関数で定義されます。

- ダブルスカラー入力
- サイズ2の入力用ダブル配列
- サイズ3の出力用ダブル配列

そして、ダブルスカラーを返します。

外部コードカスタムコンポーネントを作成するには

1. 外部コードを呼び出す新規 MapleSim モデルを作成します。
2. [Apps とテンプレートを追加] タブ (🔗) を選択します。
3. [テンプレート] パレットの [外部 C/ライブラリブロック] 項目をダブルクリックします。
4. テンプレート名を入力し、[添付を作成] のチェックマーク (✓) をクリックします。Maple で外部 C コード/ライブラリ定義テンプレートが開きます。
5. 外部コード/ライブラリの場所セクションで、[Text Area を使用した外部コードの指定] を選択します。
6. Windows プラットフォームでは、**図6.13 「Windows 用外部 C コード定義」** に示されている関数宣言を使用してテキスト領域に C コードを入力します。Unix プラットフォームでは、**図6.14 「Unix 用外部 C コード定義」** に示されている関数宣言を使用して C コードを入力します。

Provide External Code using Text Area Specify Header File

```
/*
*****
*/
#ifdef WMI_WINNT
#define EXP __declspec(dllexport)
#define M_DECL __stdcall
#else
#define EXP
#define M_DECL
#endif

EXP double M_DECL f1(double a, double *b, double *c)
{
    c[0] = a*a;
    c[1] = b[0] + b[1];
    c[2] = c[0]*c[1];

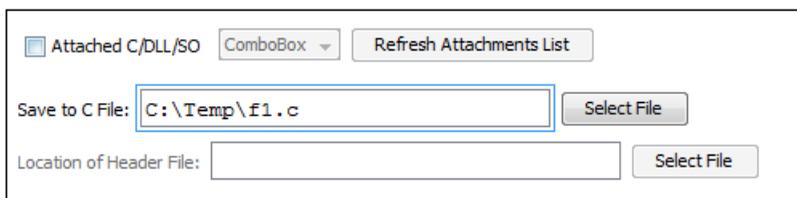
    return 1.0;
}
/*
*****
*/
```

図6.13 Windows 用外部 C コード定義



図6.14 Unix 用外部 C コード定義

7. [Save to C File] に、外部 C/ライブラリの場所を指定します。



8. [C/ライブラリファイルの検証と保存] をクリックします。

9. [設定] セクションで [関数] を選択し、[外部関数名] に「f1」と入力します。



10. [設定] セクションで [引数] を選択します。

11. ダブルスカラー入力 'a' に次の値を指定し、[パラメータを追加] をクリックします。引数テーブルに図6.15 「引数テーブル」のようにパラメータが表示されます。



12. サイズ2の入力ダブル配列 'b' に次の値を指定し、[パラメータを追加] をクリックします。引数テーブルに図6.15 「引数テーブル」のようにパラメータが表示されます。

Parameter Name: Passed By Reference

Data Type: Array?

- 13 サイズ3の出力ダブル配列 'c' に次の値を指定し、[パラメータを追加] をクリックします。引数テーブルに図6.15 「引数テーブル」のようにパラメータが表示されます。

Parameter Name: Passed By Reference

Data Type: Array?

	Parameter Name	Data Type	Change Row
1	a	float	
2	b	float, 2	
3	c	float, 3, output	

Row ID to Remove:

図6.15 引数テーブル

- 14 [設定] セクションで [戻り値] を選択します。
- 15 C 関数の戻り値のパラメータに次の値を指定します。

Return?

Return Name:

Return Type:

- 16 [外部コード/ライブラリの場所] セクションで、以下を実行します。
- 作成した Modelica カスタムコンポーネントコードを保存するディレクトリを、[生成先ディレクトリ] に入力します。

- 「ExternalCode」と [ブロック名] フィールドに入力します。

Target directory:

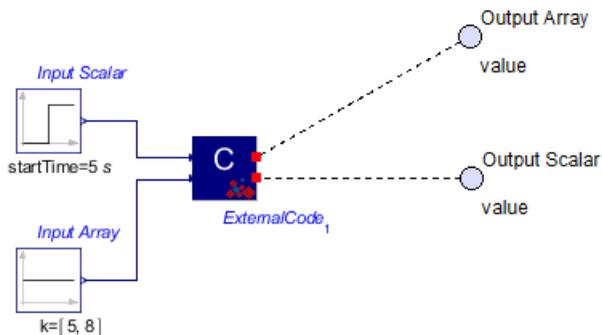
Block Name:

17. [外部コードコンポーネントの生成] をクリックします。MapleSim ウィンドウ

左側の [ローカルコンポーネント] タブ (🔍) の [コンポーネント] パレットに、
カスタムコンポーネントが表示されます。

外部コードカスタムコンポーネントを使用するには

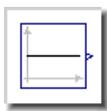
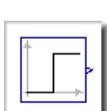
1. 指定したモデルコンポーネントと 表6.6 「外部Cコード DLL カスタムコンポーネントと必要な設定」 からの設定を使用して、コンポーネントをモデルワークスペースにドラッグし、値を設定します。



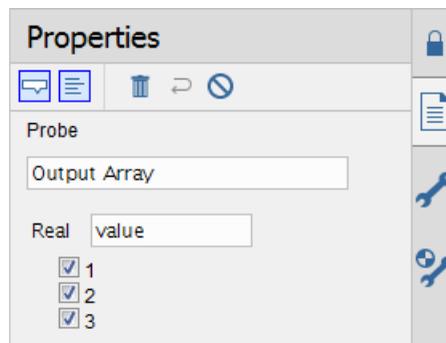
注: モデルにモデルコンポーネントのパラメータ値が設定されていることを確認します。モデルワークスペースでコンポーネントを選択すると、そのコンポーネントの設定可能なパラメータ値が MapleSim ウィンドウ右側の [プロパティ] タブ (📄) に表示されます。

表6.6 「外部Cコード DLL カスタムコンポーネントと必要な設定」 に必要なコンポーネントと設定を示します。

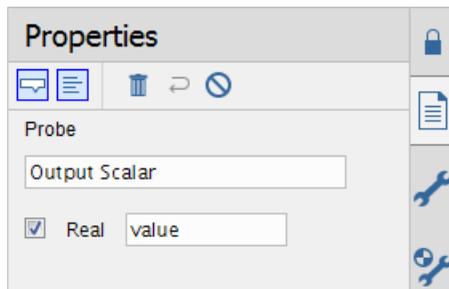
外部 C コード DLL カスタムコンポーネントと必要な設定

コンポーネント	記号	コンポーネントの場所	必要な設定
カスタムコンポーネント		[ローカルコンポーネント]>[コンポーネント]	デフォルト設定を使用
Constant Vector		[ライブラリコンポーネント]>[信号ブロック]>[ソース]>[実数]	定数出力値として、 K を [5, 8] に設定
Step		[ライブラリコンポーネント]>[信号ブロック]>[ソース]>[実数]	高さ: 4 オフセット: 0 T_0 : 5

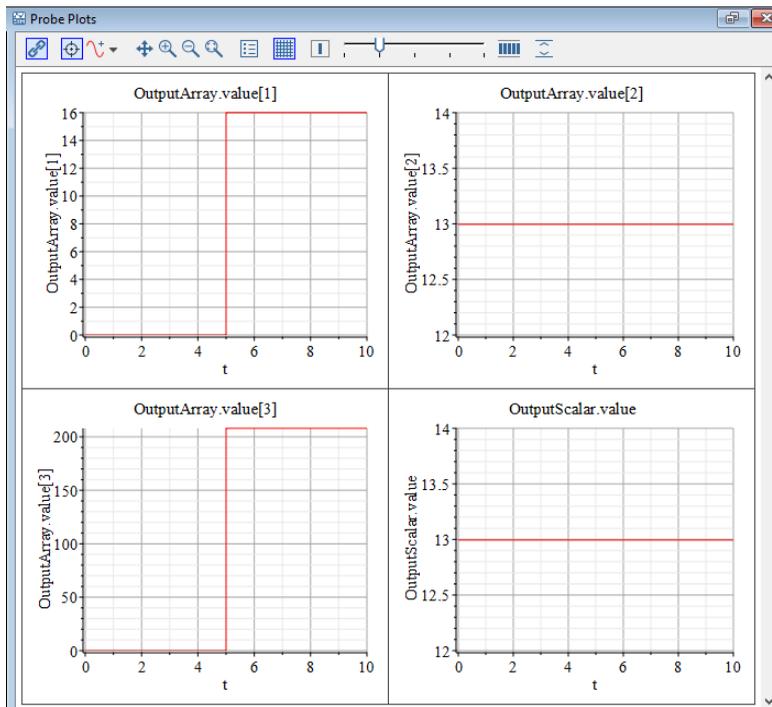
2. **Step** コンポーネントをカスタムコンポーネントの入力ポート **a** に接続します。
3. **Constant Vector** コンポーネントをカスタムコンポーネントの入力ポート **b** に接続します。
4. カスタムコンポーネントの出力ポート **c** にプローブを追加し、以下の値を入力します。



5. カスタムコンポーネントの出力ポート **r** にプローブを追加し、以下の値を入力します。



6. メインツールバーでシミュレーションの実行 (▶) をクリックします。
7. シミュレーション結果を表示 (📊) をクリックします。[解析ウィンドウ] に以下のグラフが表示されます。



8. 結果を検証するには、[ヘルプ] メニューから [例題] > [ユーザガイドの例題] > [第6章] の順に展開し、[Simple External C Code Function] の例を選択します。

6.7. チュートリアル 7: 方程式の抽出 App の使用

このチュートリアルでは、方程式の抽出 App を使用して以下のタスクを実行し、モデルの方程式を抽出します。

- モデルを開く
- 方程式の抽出 App を開く
- 方程式を表示、操作、再割り当てを行う

方程式の抽出 App にはプリビルトの埋め込みコンポーネントが含まれています。これを使用して、MapleSim モデルによって生成された数式のシステム方程式を抽出、操作、および解析できます。

任意の変数およびパラメータを選択し、それらにユーザ定義の名前を割り当てることができます。これらの機能は、複数のサブシステムが存在する場合に、再利用可能な方程式を生成するのに便利です。

App の説明

方程式の抽出 App は、MapleSim モデルから方程式を簡単に作成するための、特定の Maple コマンドに関連するプリビルトのコントロールとプロシージャの集合です。

方程式の抽出 App は、**[方程式の詳細]** と **[方程式の表示]** の 2 つの主なセクションで構成されています。

サブシステムの選択

このセクションでは、MapleSim のモデルをロードし、すべてのサブシステムとそのコンポーネントを表示します。ツールバーからサブシステムを選択し、そのサブシステムの方程式をロードします。

[選択したサブシステムを読み込む]: サブシステムのパラメータと変数を読み込みます。サブシステムが選択されていない場合に **[選択したサブシステムを読み込む]** をクリックすると、モデル全体の方程式がロードされます。

Subsystem Selection:

Main >

Load Selected Subsystem

パラメータや変数の操作

このセクションでは、生成された方程式のポート、DAE 変数、およびパラメータのカスタマイズおよび定義ができます。

ポート

非因果ポートの場合、通過変数または介在変数が(既知の)入力となるようにポートを設定することができます。

入力として、2つの信号のうち1つを選択する必要があります。各ポートを選択し、入力信号として **[通過変数]** または **[介在変数]** のいずれかを選択します。

DAE 変数

任意の DAE 変数を選択し、名前を変更します。

[変数]: モデルの DAE 変数です。

[新しい名前]: 変数を選択し、**[新しい名前]** フィールドに新しい変数名を指定して、DAE 変数の名前を変更します。

[保持]: 任意の変数を選択して **[保持]** にチェックを付けると、変数にフラグを設定することができます。この機能を使用すると、**[方程式の抽出]** では、**[保持]** が設定された変数の方程式のみが表示されます。どの変数やパラメータも保持として設定されていない場合、すべての方程式が表示されます。

[下付き文字を使用]: 変数名に下付き文字を使用します。

[代入をリセット]: すべての変数を元の名前に戻します。

パラメータ

任意のパラメータを選択し、名前を変更します。

[パラメータ]: モデルのパラメータです。

[新しい名前]: パラメータを選択し、**[新しい名前]** フィールドで新しい名前を指定して、モデルパラメータの名前を変更します。

[保持]: 任意のパラメータを選択して **[保持]** にチェックを付けると、パラメータにフラグを設定することができます。この機能を使用すると、**[方程式の抽出]** では、**[保持]** が設定されたパラメータの方程式のみが表示されます。どの変数やパラメータも保持として設定されていない場合、すべての方程式が表示されます。

[記号]: 記号形式で残すパラメータを指定します。デフォルトでは、パラメータは方程式内で評価されます (または、**[記号/数値のトグル]** を使用して、すべてのパラメータに対してこの設定を切り替えます)。

[下付き文字を使用]: パラメータ名に下付き文字を使用します。

[代入をリセット]: すべてのパラメータを元の名前に戻します。

方程式を表示

このセクションには、割り当てられたパラメータを持つ数式形式のシステム方程式が表示されます。参照する方程式を選択するには、方程式のいずれかの種類 (DAEs、Definitions、Relations、Events、ODEs、AEs) を選択します。方程式の種類に関する詳細については、`GetEquations` を参照してください。

コードエディタには、方程式にアクセスするためにワークシートテンプレートで使用できる Maple コードが表示されます。

View Equations:

 Simplify returned equations (may take time for large equations)

Extract Equations

 Auto Update

 Typeset Math

 Equations Types: DAEs Definitions Relations Events ODEs AEs

```

1 A := MapleSim:-LinkModel():
2 A:-SetSubsystemName("sub"):
3 A:-SetSubstitutions(['Main.sub.contact1.B' = B_Cont1, `
4 eqs := A:-GetEquations('output' = 'all', 'inputs' = {'Ma
5 daes := eqs[1];

```

$$\left\{ \begin{array}{l} \frac{981}{100} + \frac{d^2}{dt^2} y_{RB1}(t) = 0, - \left(\left(\frac{d}{dt} \zeta_{RB1}(t) \right) \cos(\zeta_{RB1}(t)) \left(\frac{d}{dt} \xi_{RB1}(t) \right) \cos(\right. \\ \left. + \left(\frac{d}{dt} \zeta_{PR1}(t) \right) \left(\frac{d}{dt} \eta_{PR1}(t) \right) \sin(\zeta_{PR1}(t)) + \left(\frac{d^2}{dt^2} \xi_{PR1}(t) \right) \cos(\eta_{PR1}(t)) \right) \end{array} \right.$$

方程式の生成

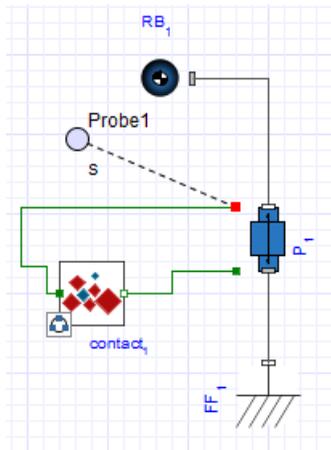
この例では、チュートリアル5: カスタムコンポーネントテンプレートの使用 [213ページ] の例: コンプライアント接地と区分関数 [220ページ] で使用されているモデルを使用します。このモデルは、[ヘルプ] > [例題] > [ユーザガイドの例題] > [第6章] から取得することもできます。このモデルでは、落下するボールをモデリングするためのプリズムジョイントと、バネダンパ配置を使用したコンプライアントな地面との接地をモデリングするためのカスタムコンポーネントを使用して、バウンドするボールを解析します。

これらのコンポーネントをサブシステムにグループ化し、方程式の抽出Appで使用できるようにします。

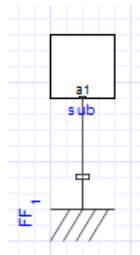
システム方程式の生成

システム方程式を生成するには

1. [ヘルプ] メニューの [例題] > [ユーザガイドの例題] > [第6章] から、[Compliant Contact and Piecewise Functions] を選択します。

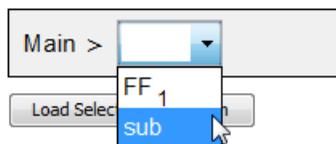


2. **sub** と呼ばれるサブシステムに **Prismatic** ジョイント、**contact** カスタムコンポーネント、プローブ、および **Rigid Body** を配置します。これにより、方程式の抽出テンプレートが選択したサブシステム固有の方程式を生成します。



3. [Apps とテンプレートを追加] タブ () を選択します。
4. [Apps] パレットの [方程式の抽出] 項目をダブルクリックします。方程式の抽出 App は [解析ウィンドウ] で開きます。[サブシステムの選択] ウィンドウのツールバーのプルダウンに、モデル内のすべてのサブシステムが表示されます。
5. ツールバーで「**sub**」という名前のサブシステムを選択します。

Subsystem Selection:



6. [選択したサブシステムを読み込む] をクリックします。サブシステムのコンポーネントのポート、DAE 変数、パラメータが、[ポート]、[DAE 変数]、および [パラメータ] セクションに自動的にロードされます。
7. [方程式の表示] セクションで [方程式の抽出] をクリックします。

方程式の表示セクションにシステム方程式が表示されます。

View Equations:

Simplify returned equations (may take time for large equations)

Extract Equations

Auto Update

Typeset Math

Equations Types: DAEs Definitions Relations Events ODEs AEs

```

1 A := MapleSim:-LinkModel():
2 A:-SetSubsystemName("sub"):
3 A:-SetSubstitutions(['Main.sub.contact1.B' = B__Cont1, `
4 eqs := A:-GetEquations('output' = 'all', 'inputs' = {'Ma
5 daes := eqs[1];

```

$$\left\{ \begin{array}{l} \frac{981}{100} + \frac{d^2}{dt^2} y_{RB1}(t) = 0, - \left(\left(\frac{d}{dt} \zeta_{RB1}(t) \right) \cos(\zeta_{RB1}(t)) \left(\frac{d}{dt} \xi_{RB1}(t) \right) \cos \right. \\ \left. + \left(\frac{d}{dt} \zeta_{PB1}(t) \right) \left(\frac{d}{dt} \eta_{PB1}(t) \right) \sin(\zeta_{PB1}(t)) + \left(\frac{d^2}{dt^2} \xi_{PB1}(t) \right) \cos(\eta_{PB1}(t)) \right\} \end{array} \right.$$

6.8. チュートリアル 8 : 油圧システムのモデリング

このチュートリアルでは、油圧システムの基本的な説明を行います。この説明は、MapleSim でこれらのシステムのモデルの作成方法を理解する際に役立ちます。油圧ライブラリのコンポーネントを使用して、モデルを作成したり、初期条件やコンポーネントプロパティを設定したり、パラメータや変数に新しい値を代入することができます。

油圧コンポーネントは、主に油圧流路を機械的な動きに変換することを目的としていますが、純粋な油圧回路のモデリングにも使用できます。

このチュートリアルでは、以下の基本原則と概念に基づいてタスクを実行します。

- 基本的な油圧ライブラリコンポーネント
- 基本的な油圧方程式
- シンプルな油圧ネットワークの解析
- 第一原則のモデリング
- メカニカルシステムと油圧システム

次のセクションでは、油圧ライブラリコンポーネントを使用して構築できる概念モデルについて説明します。

- 油圧流路の制御
- 油圧コンポーネントを持つマルチボディシステムの動作
- 油圧作動油の圧縮率
- 流体慣性モデル
- ウォーターハンマーモデル
- 油圧カスタムコンポーネント

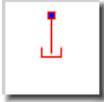
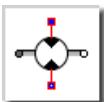
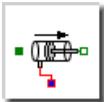
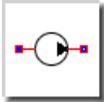
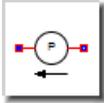
計算上の問題

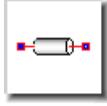
油圧ネットワークは、数値的に stiff になる傾向があります。一般的に、Rosenbrock 法の stiff ソルバの使用を推奨します。

基本的な油圧ライブラリコンポーネント

このチュートリアルでは、次の基本的な油圧ライブラリコンポーネントを使用します。

基本的な油圧ライブラリコンポーネント

コンポーネント	記号	ライブラリの場所	用途
Tank		[ライブラリコンポーネント]>[油圧]>[リファレンス]	このコンポーネントは基準の圧力 (電気ドメインのグラウンドと同じ) を定義し、大気との接続を表します。
Hydraulic Fluid Properties		[ライブラリコンポーネント]>[油圧]>[リファレンス]	すべての油圧モデルには Hydraulic Fluid Properties コンポーネントが必要です。このコンポーネントはパラメータブロックと同様に モデルワークスペース に配置され、次の油圧流体プロパティを定義します。 <ul style="list-style-type: none"> • rhoFluid : 流体の密度 • K : 体積弾性率 (流体の圧縮率) • nuFluid : 動粘度 (絶対粘度を液体の密度で割ったもの)
Hydraulic Motor		[ライブラリコンポーネント]>[油圧]>[アクチュエータ]	アクチュエータは油圧流路をメカカルボディの運動に変換します。MapleSim では油圧シリンダ (並進運動用) と油圧モータ (回転運動用) を提供しています。
Hydraulic Cylinder			
Fixed Flow Rate		[ライブラリコンポーネント]>[油圧]>[ソース]	油圧ソースの流量または圧力のいずれかを指定できます (MapleSim でほかの物理量を計算)。圧力源を使用する場合、MapleSim は圧力源に対する油圧システムの負荷を調整し、流量を求めます。また、その逆も行います。
Fixed Pressure Source			

コンポーネント	記号	ライブラリの場所	用途
Circular Pipe		[ライブラリコンポーネント]>[油圧]>[絞り]	円形パイプには油圧管路の圧力低下が取り入れられます。圧力低下は Darcy の式と、事前に定義された方程式によって決定される摩擦係数によって求められます。

基本的な油圧方程式

Bernoulli および Darcy の式は、油圧システムの解析、流れに沿ったポイントの流圧および流量の特性の定義に必要な、基本的な方程式です。このチュートリアルでは、次の基本的な流体方程式を使用します。

- Bernoulli の式
- Darcy の式
- 摩擦係数

Bernoulli の式

Bernoulli の式は、パイプ内の非圧縮性流体の流れの圧力と流量の特性を定義します。流線沿いにあるポイントには次の関係が適用されます。

$$\frac{p}{\rho \cdot g} + \frac{V^2}{2 \cdot g} + z = constant$$

Darcy の式

定直径を持つパイプを流れる非圧縮性流体の場合、パイプの摩擦によって起こる圧力低下は Darcy の式によって求められます。

$$\Delta P = f \cdot \frac{L \cdot V^2}{D \cdot 2 \cdot g}$$

したがって、次のようになります。

$$\frac{p}{\rho \cdot g} + \frac{V^2}{2 \cdot g} + z + f \cdot \frac{L \cdot V^2}{D \cdot 2 \cdot g} = constant$$

Bernoulli と Darcy の式の表記

記号	説明	単位
P	圧力	Pa
ρ	密度	$\frac{kg}{m^3}$
g	重力定数	$\frac{m}{s^2}$
V	速度	$\frac{m}{s}$
z	高度	m
L	パイプの長さ	m
D	パイプの直径	m
f	摩擦係数	無次元

液体 (層流) 内の摩擦による内部の影響、およびパイプの表面の凸凹による影響 (乱流) を取り除くために、圧力が適用されます。流速を判断するには、適用された圧力に対する摩擦損失 (およびシステム内のその他の負荷) のバランスを取る必要があります。

MapleSim のメカニカル油圧システムでは、垂直変位 (z) はほかの条件と比べてそれほど重要ではないため、無視されます。

摩擦係数

層流では、内部摩擦 (f) の影響は次の式によって求められます。

$$f = \frac{64}{Re}$$

$$Re = \frac{D \cdot V}{\nu}$$

ここで、

f は内部摩擦を表します。

Re はレイノルズ数を表します。

D はパイプの直径を表します。

V は流体速度を表します。

ν は動粘度を表します。

乱流では、パイプ表面の凸凹の摩擦効果は **Haaland の式** によって特徴付けられます。

$$f = \frac{1}{\left(1.8 \log_{10} \left(\frac{6.9}{Re} + \left(\frac{\epsilon}{3.7 \cdot D} \right)^{1.11} \right)\right)^2}$$

レイノルズ数 (Re) は、パイプ内の流れが層流なのか、乱流なのか、またはこの2つのあいだを移行しているのかを示します。たとえば、表6.9「円形パイプのパラメータ」の円形パイプのパラメータは、層流 (ReL) と乱流 (ReT) のレイノルズ数を示しています。この2つのパラメータのあいだの摩擦係数は線形補間によって決まります。

円形パイプのパラメータ

記号	説明	値
D	油圧配管直径	0.01 m
L	パイプの長さ	5 m
ϵ	内部パイプの凸凹の高さ	$1.5 \cdot 10^{-5}$ m
ReL	層流域の最大レイノルズ数	2000
ReT	乱流域の最小レイノルズ数	4000

シンプルな油圧ネットワークの解析

このセクションでは、シンプルな油圧システムのシミュレーションを行い、第一原理からの結果を解析し、以下の方法について説明します。

- シンプルな層流パイプの油圧システムを作成する
- さまざまな法則 (質量保存、Bernoulli の式、Darcy の式など) を適用して支配方程式を解析する

パイプの流れ

図6.16 「パイプの流れ」では、内部摩擦の影響を減らすために圧力を適用した場合の、パイプからの圧力と層流量の特性を解析します。

パイプからの流れを解析するには

1. 指定されたモデルコンポーネントと表6.10「油圧コンポーネントと必要な設定」の設定を使用して、次のモデルを作成します。

ヒント: **Circular Pipe** コンポーネントにプローブを追加するには、コンポーネントを右クリック (Macの場合は[Control]キーを押しながらクリック) し、[プローブを追加]を選択してからワークスペースをクリックし、プローブを配置します。

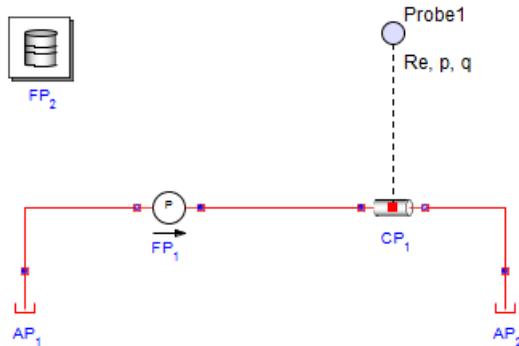
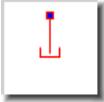
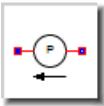
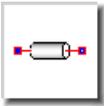
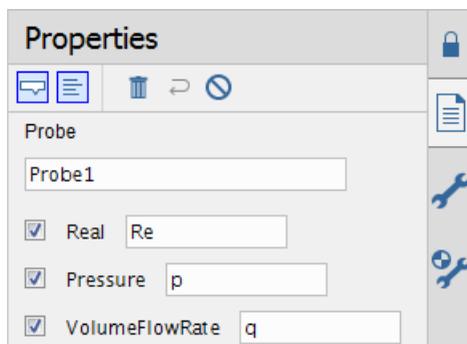


図6.16 パイプの流れ

油圧コンポーネントと必要な設定

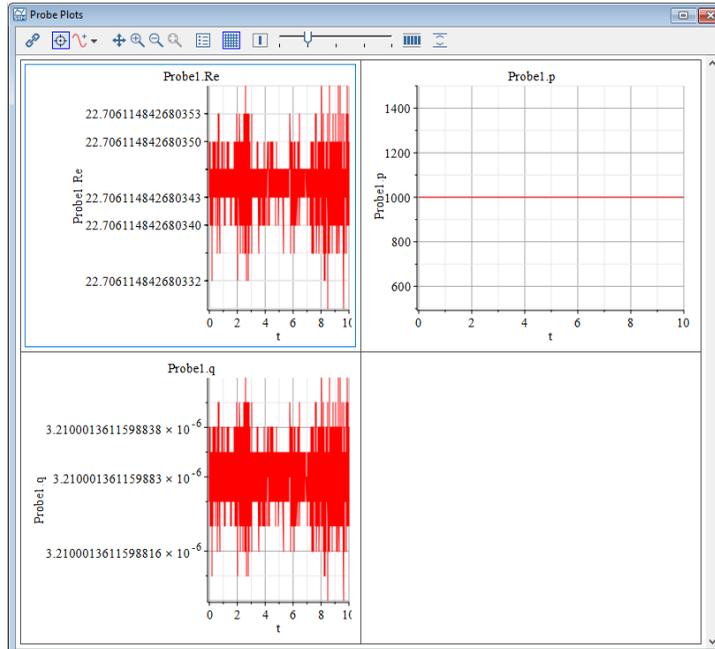
コンポーネント	数量	記号	ライブラリの場所	必要な設定
Tank	2		[油圧] > [リファレンス]	デフォルト設定を使用
Hydraulic Fluid Properties	1		[油圧] > [リファレンス]	rhoFluid : $850 \frac{kg}{m^3}$ K : 8000 bar nuFluid : $0.000018 \frac{m^2}{s}$
Fixed Pressure Source	1		[油圧] > [ソース]	P = $1 \cdot 10^3 Pa$ に設定
Circular Pipe	1		[油圧] > [絞り]	表6.9 「円形パイプのパラメータ」を参照

2. プローブをクリックし、**[Real]** (瞬時レイノルズ数)、**[Pressure]**、および **[VolumeFlowRate]** プローブパラメータを選択します。



3. メインツールバーでシミュレーションの実行 (▶) をクリックします。

4. シミュレーション結果を表示(📄)をクリックします。 $Q = 3.21 \cdot 10^{-6} \frac{m^3}{s}$ の予測流量を示す以下のようなグラフが表示されます。



第一原理のモデリング結果の確認

図6.16 「パイプの流れ」 に示すシステムを解析する場合は、次のDarcyの式を適用します。

$$\frac{\Delta P}{\rho \cdot g} = f \cdot \frac{L \cdot V^2}{D \cdot 2 \cdot g}$$

システムが層流であると仮定すると、次のようになります。

$$f = \frac{64}{Re}$$

したがって、次のようになります。

$$\frac{1 \cdot 10^3}{850 \times 9.81} = \frac{64}{\frac{0.01 \times V}{0.000018}} \times \frac{5}{0.01} \times \frac{V^2}{2 \times 9.81}$$

$$0.1199256461 = 2.935779816 V$$

ゆえに、以下のようになります。

$$V = 0.0408 \frac{m}{s}$$

流量の方程式で **V** を使用すると、次のような結果になります。

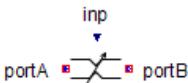
$$Q = \frac{1}{4} V \pi D^2 = 0.0000408 \times \frac{\pi \times 0.01^2}{4} = 3.2 \times 10^{-6} \frac{m^3}{s}$$

これは、MapleSim で求めた値と同じです。計算された **V** の値を使用すると **Re = 22.7** になります。これは、臨界値 2000 よりかなり少ないため、このシステムが層流であることがわかります。

油圧流路の制御の概要

スプール弁には鋭利で可変領域なオリフィスがあります。このオリフィスによってパイプの流れを可能にしたり、部分的に制限できます。また、油圧ネットワークをある部分から別の部分に切り替えることもできます。スプール弁には3つのポートがあります。

スプール弁

スプール弁	名前	説明	ID
	<i>PortA</i>	上流ポート	portA
	<i>PortB</i>	下流ポート	portB
	<i>Area</i>	入力信号;オリフィスの面積 (選択単位)	Area

上部のポート (**inp**) は、開放弁領域と同じ信号入力を受け入れます。このバルブ領域を制限することで、流れをオンまたはオフに切り替えます。左右のポート

(portA および portB) は油圧コネクタです。次の図では、シミュレーション時間が 5 秒に達した場合に、上部の経路から下部の経路にモデルが切り替えられます。つまり、上部のスプール弁は開いており、下部のスプール弁は閉じています。5 秒後に上部のスプール弁が閉じられ、下部のスプール弁が開かれます。

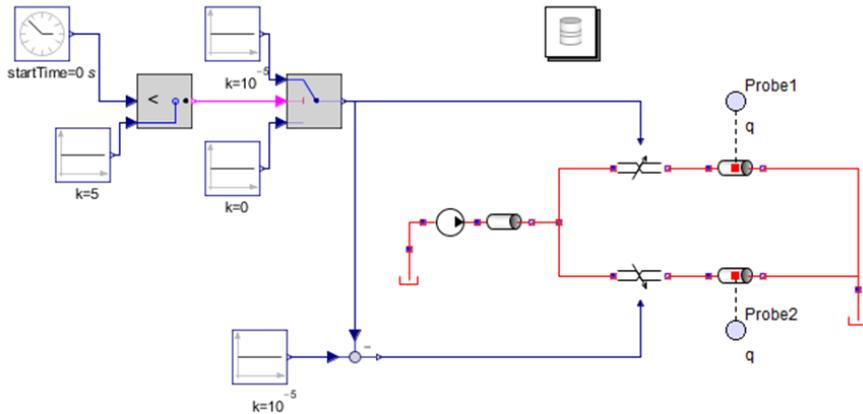


図6.17 流路の制御

メカニカルシステムと油圧システム

次の例では、マルチドメインコンポーネントを使用し、以下のソースを持つメカニカルモデルおよび油圧モデルでの並進運動をシミュレーションします。

- 固定流量源
- 固定圧力源

固定流量源を使用した並進運動のシミュレーション

次のモデルは、表6.12「固定流量源を使用した並進運動」に示すコンポーネントと設定を使用して、固定流量源からの流れを並進運動に変換します。

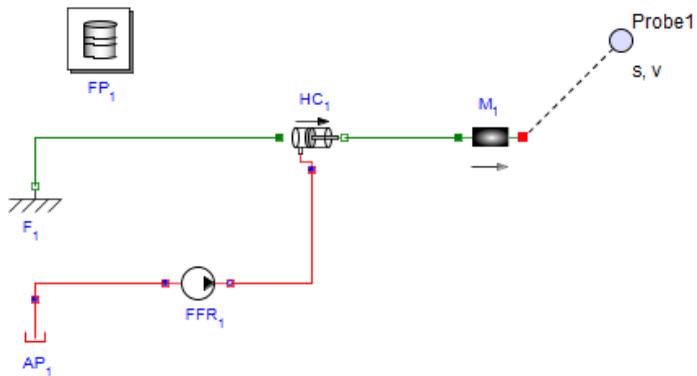
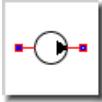
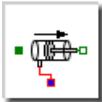
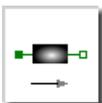
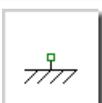


图6.18 固定流量源

固定流量源を使用した並進運動

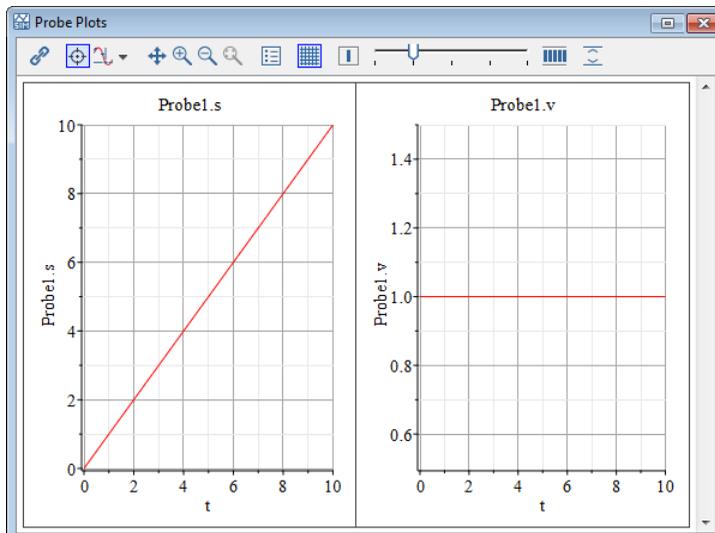
コンポーネント	数量	記号	ライブラリの場所	必要な設定
Tank	1		[油圧] > [リファレンス]	デフォルト設定を使用
Hydraulic Fluid Properties	1		[油圧] > [リファレンス]	rhoFluid : $850 \frac{kg}{m^3}$ K : 8000 bar nuFluid : $18 \times 10^{-6} \frac{m^2}{s}$
Fixed Flow Rate	1		[油圧] > [ソース]	デフォルト設定を使用
Hydraulic Cylinder	1		[油圧] > [アクチュエータ]	デフォルト設定を使用
Mass	1		[1-D メカニカル] > [並進] > [共通]	デフォルト設定を使用
Translational Fixed	1		[1-D メカニカル] > [並進] > [共通]	デフォルト設定を使用

注: 油圧シリンダには断面領域 A (1 m^2) があります。一方で、固定流量源には流れ Q ($1 \frac{\text{m}^3}{\text{s}}$) があります。

シリンダは以下の速度で sliding mass を押しします。

$$V = \frac{Q}{A} = \frac{1 \text{ m}^3 \text{ s}^{-1}}{1 \text{ m}^2} = 1 \frac{\text{m}}{\text{s}}$$

これは、シミュレーションを実行してsliding massの速度を調べると確認できます。



固定圧力源を使用した並進運動のシミュレーション

図6.19「固定圧力源を使用した並進運動」および表6.13「固定圧力源を使用した並進運動」に示すように、固定流量源を固定圧力源に置き換えます。次のモデルは、固定圧力源からの流れを並進運動に変換します。

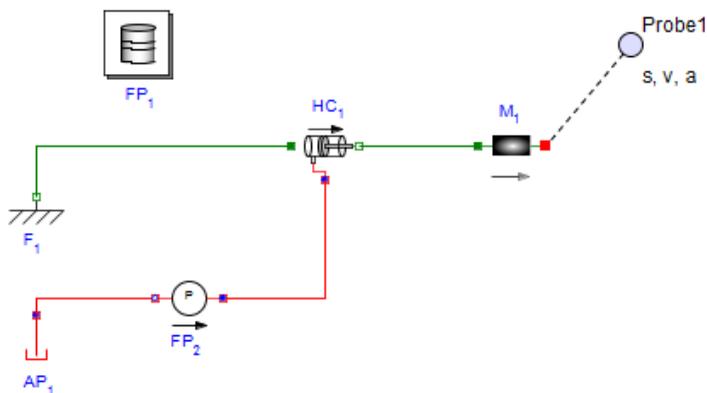
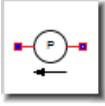


図6.19 固定圧力源を使用した並進運動

固定圧力源を使用した並進運動

コンポーネント	記号	ライブラリの場所	必要な設定
Fixed Pressure Source		[油圧] > [ソース]	デフォルト設定を使用

Sliding Mass の力は、油圧シリンダの断面領域 A に油圧流体の圧力 P をかけた値と等しくなります。

$$F = A \cdot P = 1 \text{ m}^2 \cdot 1 \text{ Pa} = 1 \text{ N}$$

Sliding Mass の加速度は次の式によって求められます。

$$F = m \cdot a = 1 \text{ m}^2 \cdot 1 \text{ Pa} = 1 \text{ N}$$

$$1 \text{ N} = 1 \text{ kg} \cdot a$$

ゆえに、次のようになります。

$$a = 1 \frac{\text{m}}{\text{s}^2}$$

Sliding Mass の加速度、速度、変位を調べ、**図6.20 「固定圧力源の結果」** の結果を使用してこれらの値を確認します。

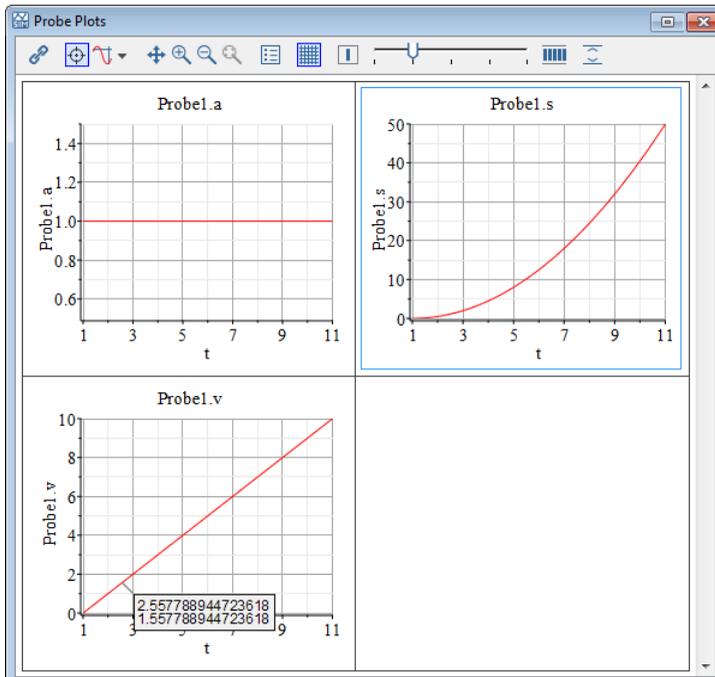


図6.20 固定圧力源の結果

油圧コンポーネントを持つマルチボディシステムの動作の概要

次のモデルでは、表6.14「マルチボディコンポーネントの動作」の固定フランジの並進と剛体の質量を使用して、油圧シリンダの1D並進ポートをマルチボディの並進ジョイントにある1-D並進ポートに接続します。各コンポーネントには、デフォルトの設定を使用します。

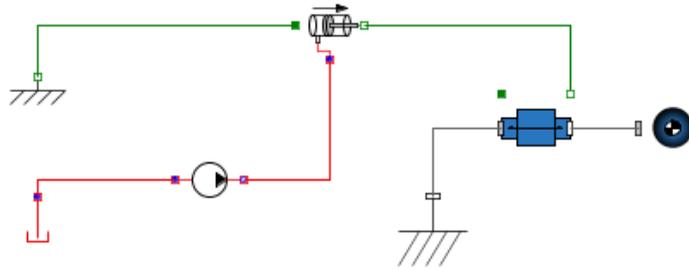


図6.21 固定並進フランジ油圧コンポーネント

同様に、次のモデルでは、表6.14「マルチボディコンポーネントの動作」の固定フランジの回転と剛体の質量を使用して、油圧モータの1-D回転ポートをマルチボディの回転ジョイントにある1-D回転ポートに接続します。各コンポーネントには、デフォルトの設定を使用します。

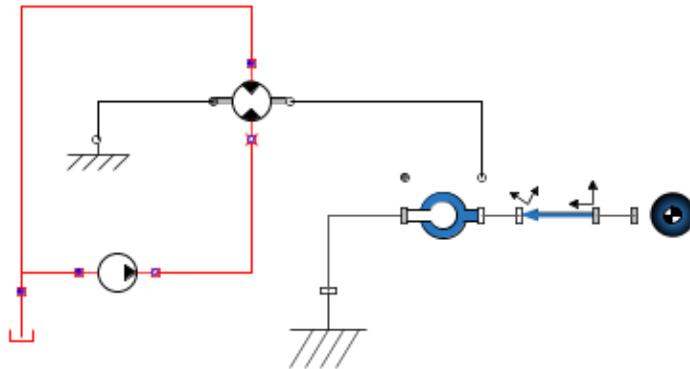
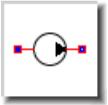
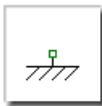
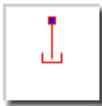
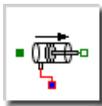
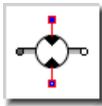
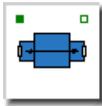
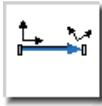
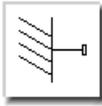


図6.22 固定回転フランジ油圧コンポーネント

マルチボディコンポーネントの動作

コンポーネント	記号	ライブラリの場所
Fixed Flow Rate		[油圧] > [ソース]

コンポーネント	記号	ライブラリの場所
Rotational Fixed Flange		[1-D メカニカル] > [回転] > [共通]
Translational Fixed Flange		[1-D メカニカル] > [並進] > [共通]
Tank		[油圧] > [リファレンス]
Hydraulic Cylinder		[油圧] > [アクチュエータ]
Hydraulic Motor		[油圧] > [アクチュエータ]
Prismatic		[マルチボディ] > [ジョイント・モーション]
Rigid Body Frame		[マルチボディ] > [ボディ・フレーム]
Rigid Body		[マルチボディ] > [ボディ・フレーム]
Revolute		[マルチボディ] > [ジョイント・モーション]
Fixed Frame		[マルチボディ] > [ボディ・フレーム]

パスカルの原理

パスカルの原理によれば、密閉された油圧システムに対する圧力は、すべての場所に均等に伝達されます。この原理は、適用された力を増幅し、通常は移動できない荷重を移動できることを示しています。

図6.23 「パスカルの原理の例」のモデルは、パスカルの原理の簡単な例を示しています。1 N の力 (0.1 m^2 の油圧シリンダに作用) は、油圧を 1 m^2 の油圧シリンダに伝達し、1kg の重さを垂直に持ち上げます。通常は、9.81 N の力は 1kg の力の高さを維持しますが、この簡単な油圧システムでは、10倍または $\frac{1 \text{ m}^2}{0.1 \text{ m}^2}$ の係数で荷重 1 N の大きさを増やしています。

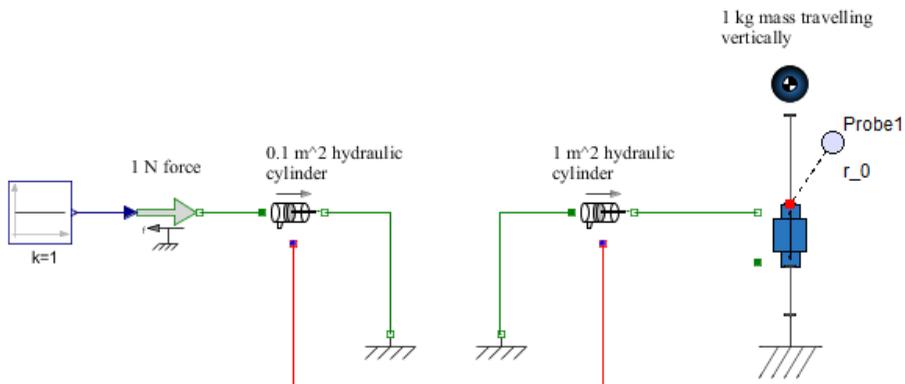


図6.23 パスカルの原理の例

油圧作動油の圧縮率の概要

Compliant Cylinder と Constant Volume のチャンバコンポーネント (表6.16 「密閉された油圧システムコンポーネント」) は、高圧時の油圧作動油の圧縮率をモデリングします。Compliant Cylinder コンポーネントは、管壁のコンプライアンスもモデリングします。いずれも、**図6.24 「油圧作動油の圧縮率」** に示すように、パイプ間のノードまたは慣性に添付する必要があります。

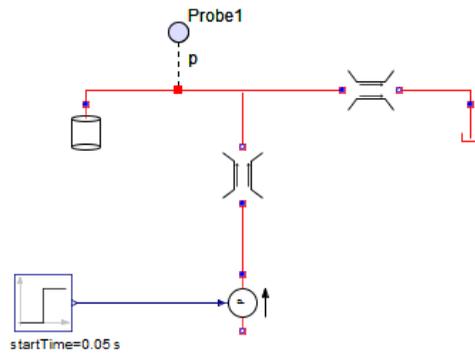
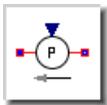
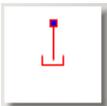
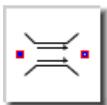


図6.24 油圧作動油の圧縮率

油圧作動油の圧縮率のコンポーネント

コンポーネント	記号	ライブラリの場所	必要な設定
Signal Pressure		[油圧] > [ソース]	デフォルト設定を使用
Tank		[油圧] > [リファレンス]	デフォルト設定を使用
Linear Resistance		[油圧] > [絞り]	デフォルト設定を使用

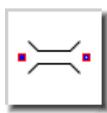
密閉された油圧システムコンポーネント

コンポーネント	数量	記号	ライブラリの場所	必要な設定
Compliant Cylinder	1		[油圧] > [チャンバ]	デフォルト設定を使用
Constant Volume	1		[油圧] > [チャンバ]	デフォルト設定を使用

流体慣性モデルの概要

Fluid Inertia コンポーネントは、パイプ内で加速または減速する液体の慣性をモデリングします。これはメカニカル inertia コンポーネントとよく似ています。流体慣性は、パイプの直径が大きい場合や、加速/減速が大きい場合に重要になる場合があります。このコンポーネントはウォーターハンマーをモデリングする場合に便利です。

流体慣性

コンポーネント	数量	記号	ライブラリの場所	必要な設定
Fluid Inertia	1		[油圧] > [絞り]	デフォルト設定を使用

流体慣性を使用しないシステム

図6.25 「流体慣性を使用しないシステム」 に、流体慣性を使用しないシステムを示します。

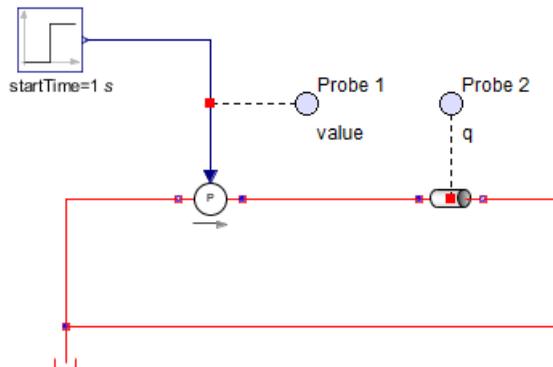


図6.25 流体慣性を使用しないシステム

流体慣性を使用するシステム

図6.25 「流体慣性を使用しないシステム」 に、流体慣性を使用するシステムを示します。

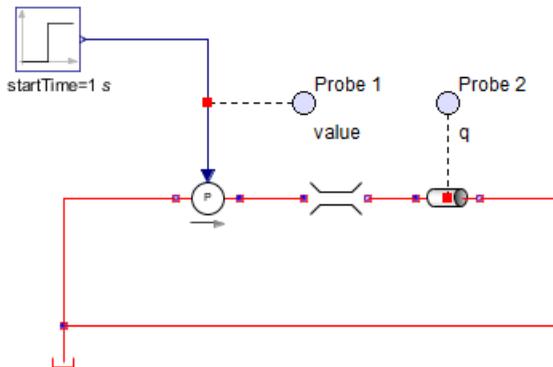


図6.26 流体慣性を使用するシステム

図6.27 「流体慣性を使用する場合と使用しない場合のシステム」は、流体慣性を使用する場合 (緑) と使用しない場合 (赤) のシステムの流量を示しています。流体慣性を使用すると、システムにラグが発生します。

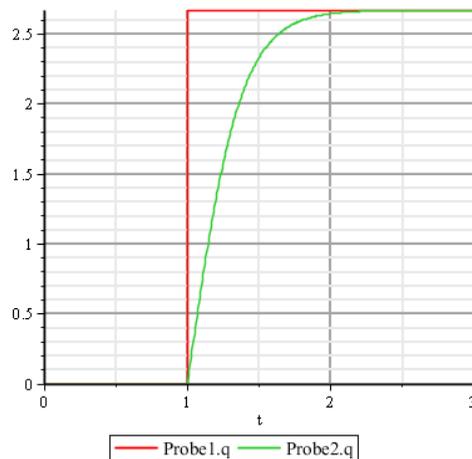


図6.27 流体慣性を使用する場合と使用しない場合のシステム

ウォーターハンマーモデルの概要

ウォーターハンマーは、バルブによってパイプ内の流れを急に止めた (または大きく制限した) 場合に発生し、流体慣性の運動量が変わるために結果として圧力

サージが発生します。この圧力サージは閉じたバルブに当たって跳ね返り、パイプを上下に移動して、パイプ全体を著しく破損させる可能性があります。従来、ウォーターハンマーは次の方程式の数値解によってモデリングされていました。

$$\frac{dV(x, t)}{dt} + \frac{1}{\rho} \frac{dP(x, t)}{dt} + \frac{fV(x, t)|V(x, t)|}{2D}$$

$$\frac{dV(x, t)}{dx} + \frac{1}{Ks} \frac{dP(x, t)}{dt} = 0$$

$$Ks = \frac{1}{\frac{1}{K} + \frac{D}{Et}}$$

ここで、

$V(x,t)$ はパイプの速度を表します。

$P(x,t)$ はパイプの圧力を表します。

ρ は液体の密度を表します。

D はパイプの直径を表します。

t はパイプの壁の厚さを表します。

K は液体の体積弾性率を表します。

E はパイプのヤング率を表します。

f は摩擦係数を表します。

これらの方程式 (適切な境界と初期条件を持つ) は、通常は数値的に解決されますが、特性曲線法を使用して方程式を解く場合はカスタムコードが必要です。

例: ウォーターハンマー

ウォーターハンマーのシミュレーションのもう 1 つの方法は、集中パラメータのパイプラインモデルを構築することです。パイプラインモデルには、流動慣性、流れ抵抗 (管摩擦から)、パイプコンプライアンス、および流体の圧縮率などの影響も含まれます。

次の図では、慣性特性と抵抗特性を持つ離散化されたパイプラインモデルで、片方の端に圧力をかけて流れを作ります。2秒後にもう一方の端にあるバルブを閉めると、圧力サージが発生します。

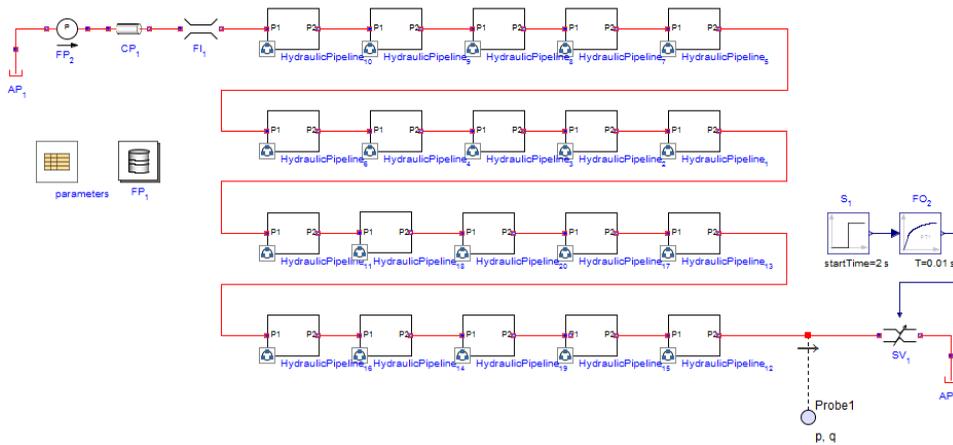


図6.28 ウォーターハンマー

図6.29「離散化されたパイプラインのセグメント」に示すように、各サブシステムは、Compliant Cylinder、Circular Pipe、および Fluid Inertia コンポーネントで構成されています。全長 L 、容量 V 、 N 個に分割されたパイプラインには、 $N + 1$ 個のパイプ(それぞれの長さは

$$\frac{L}{N + 1}$$

)、 $N+1$ 個の流体慣性コンポーネント(それぞれの長さは

$$\frac{L}{N + 1}$$

)、および N 個の一定容量のチャンバがあります。各チャンバの容量は以下のとおりです。

$$\frac{V}{N}$$

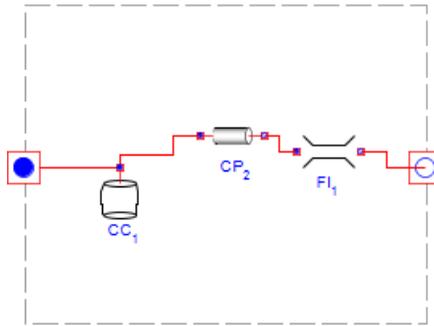


図6.29 離散化されたパイプラインのセグメント

ウォーターハンマーモデルを構築するには

1. 図6.28 「ウォーターハンマー」 や 図6.29 「離散化されたパイプラインのセグメント」 にあるようなコンポーネントと接続のあるモデルを作成します。
2. 表6.18 「Fluid Properties の値」 で指定した値で **Fluid Properties** コンポーネントを設定します。

Fluid Properties の値

パラメータ名	記号	説明	値
rhoFluid	ρ	流体の密度	1000 kg m^{-3}
K	K	体積弾性率	$200 \cdot 10^6 \text{ Pa}$
nuFluid	ν	流体の動粘度	$\frac{10^{-3} \text{ m}^2}{\text{rhoFluid} \text{ s}}$

3. 表6.19 「ウォーターハンマーのパラメータ」 に表示されるパラメータおよび値でパラメータブロックを設定します。

ウォーターハンマーのパラメータ

パラメータ名	説明	値
N	パイプのセグメント数	20
Dia	油圧配管直径 (D)	0.1 m
Len	パイプの長さ	25 m
Em	パイプのヤング率 (E)	$70 \cdot 10^9 Pa$
ef	パイプ内部の凸凹	0.0001 m
$thickness$	パイプの壁の厚さ (t)	0.001 m

4. **Fixed Pressure** コンポーネントの (図6.28 「ウォーターハンマー」 の FP_2) 圧力 (P) を $500kPa$ に設定します。
5. **Main** と **HydraulicPipeline** 共有サブシステムの **Circular Pipe** コンポーネントを、以下のように設定します。

パラメータ名	設定
D	Dia
L	$\frac{Len}{N+1}$
ϵ	ef
Re_L	2000
Re_T	4000

6. **Main** と **HydraulicPipeline** 共有サブシステムの **Fluid Inertia** コンポーネントを、以下のように設定します。

パラメータ名	設定
A	$\frac{1}{4} \cdot \pi \cdot Dia^2$
L	$\frac{Len}{N+1}$
ρ	$rhoFluid$

7. **HydraulicPipeline** 共有サブシステムの **Compliant Cylinder** コンポーネントを、以下のように設定します。

パラメータ名	設定
α	0
k	1.4
L	$\frac{Len}{N}$
D	<i>Dia</i>
D_o	<i>Dia + 2*thickness</i>
Em	<i>Em</i>

8. **First Order** コンポーネントの、[プロパティ] タブ (📄) で **T** を **0.01s** に、**y0** を 「**0.01**」 に設定します。

9. **Step** コンポーネントを、以下のように設定します。

- [height] には 「**-0.009999**」 と入力します。
- [offset] には 「**0.01**」 と入力します。
- [T_0] には **2s** と入力します。

10. [シミュレーションの設定] タブで以下のシミュレーションパラメータを設定します。

- [t_d] には **3s** と入力します。
- [Solver Type] には [Variable] を選択します。
- [Solver] には [Rosenbrock (stiff)] を選択します。

図6.30 「ウォーターハンマーの圧力の流量」 に、2 秒後に急に閉じたバルブのパイプの端における圧力と流量を示します。

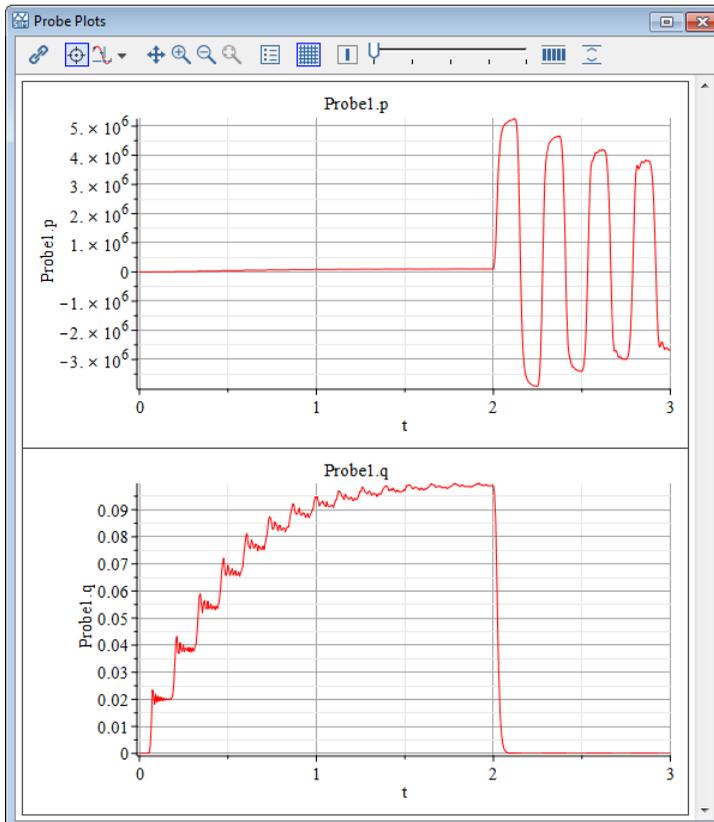


図6.30 ウォーターハンマーの圧力の流量

最大圧力は、約 $5 \times 10^6 \text{ Pa}$ で、液体の流量は $0.099 \frac{\text{m}^3}{\text{s}}$ に達します。最大圧力は次の Joukowsky の式を使用して計算することもできます。

$$\Delta P = \rho c \Delta Q$$

$$c = \sqrt{\frac{Ke}{\rho}}$$

$$Ke = \frac{1}{\frac{1}{K} + \frac{D}{Et}}$$

表6.18 「Fluid Properties の値」と表6.19 「ウォーターハンマーのパラメータ」のパラメータを前の式に代入した場合、以下のように仮定できます。

$$\Delta Q = 0.099 \frac{m^3}{s}$$

結果は以下のとおりです。

$$\Delta P \approx 5 \times 10^6 \text{ Pa}$$

この結果は MapleSim モデルと一致します。

例: アクキュムレータを使用したウォーターハンマーの軽減

油圧アクキュムレータは、多くの場合はバルブの近くに配置されているタンクで、圧力下での圧縮不可能な油圧流体を保存します。アクキュムレータは、特定のしきい値を超えるほど圧力が高まった場合に液体をタンクに入れられるようにすることで、安全弁としての役割を果たします。この動作によって、圧力波の大きさと頻度を軽減します。

MapleSim には組み込みのアクキュムレータブロックがありません。ただし、次の方程式を使用したカスタムコンポーネントテンプレートによって、この機能を簡単にモデリングできます。カスタムコンポーネントの作成方法に関する詳細については、[カスタムコンポーネントの作成 \[83ページ\]](#)を参照してください。

$$\begin{aligned} eq &:= \left[q(t) = \dot{V}F(t), VF(t) \right. \\ &= \left\{ \begin{array}{ll} Ks p(t) & p(t) \leq ppr \\ Vpr + (k \cdot (p(t) - ppr)) & ppr < p(t) < pmax, k \\ Vmax + (Ks (p(t) - pmax)) & pmax \leq p(t) \end{array} \right. \\ &= \left. \frac{(Vmax - Vpr)}{(pmax - ppr)} \right] \end{aligned}$$

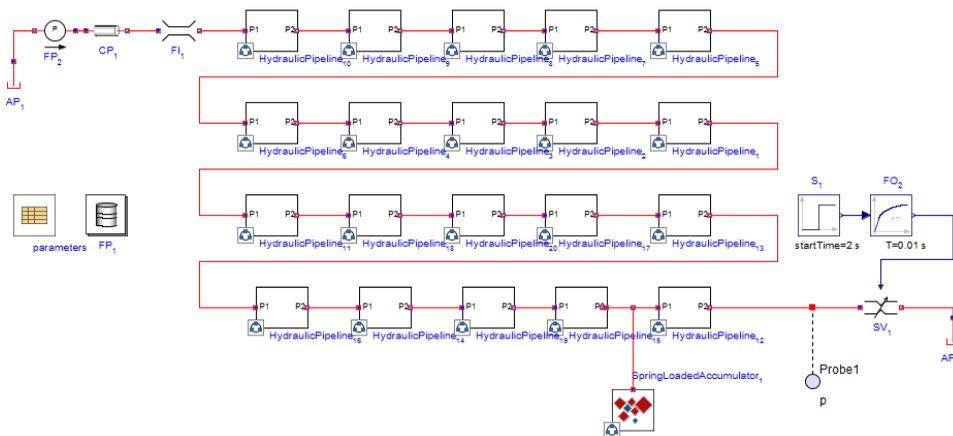
$$params := [Vmax = 0.1, ppr = 10^5, pmax = 3 \cdot 10^6, Ks = 4 \cdot 10^{-10}, Vpr = 0]$$

$initialconditions := [VF(0) = 0]$

アキュムレータパラメータのカスタムコンポーネント

説明	値
Vmax	0.1m
ppr	100000
Pmax	3000000
Ks	$10 \times 10^{-10} \text{ m}$
Vpr	0

次の図に、圧力アキュムレータと同じパイプラインを示します。2秒後にもう一方の端にあるバルブを閉めると、圧力サージが発生します。図6.31「アキュムレータを持つ場合の圧力サージ」に、アキュムレータを持つ場合のパイプラインの終端での圧力サージを示します。



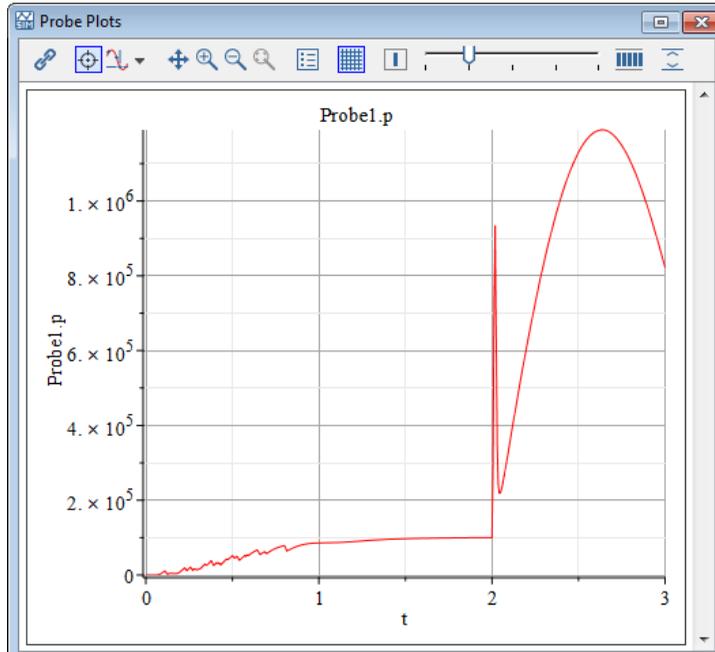


図6.31 アキュムレータを持つ場合の圧力サージ

油圧カスタムコンポーネントの概要

油圧カスタムコンポーネントの2つの例では、遠心ポンプと垂直管を示します。カスタムコンポーネントの作成方法に関する詳細については、[カスタムコンポーネントの作成 \[83ページ\]](#)を参照してください。

遠心ポンプ

通常、[図6.32「水頭流量」](#)に示すように、メーカーは遠心力ポンプの水頭流量のチャートを提供します。

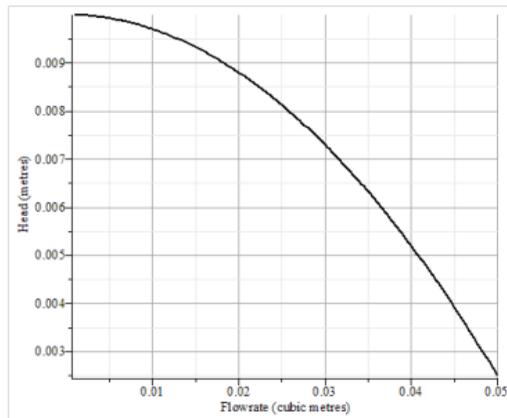


図6.32 水頭流量

これらのチャートからのデータは、カスタムコンポーネントに簡単に実装できます。

チャートのデータを実装するには

1. プロットから水頭流量ポイントの複数のセットを読み込みます。
2. Mapleのカーブフィッティング機能を使用して、これらのデータポイントを多項式に当てはめます。
3. この多項式をカスタムコンポーネントに実装します。たとえば図6.33「遠心ポンプのカスタムコンポーネントの方程式」は、最適なパラメータを含む遠心ポンプのカスタムコンポーネントの方程式を示しています。

注: この方程式は多項式であるため、複数の解が存在する場合があります。

$$\begin{aligned}
 eq &:= [P(t) = (d + c \cdot Q(t) + b \cdot Q(t)^2 + a \cdot Q(t)^3) \cdot \rho \cdot g, P(t) = Pr(t) - Pl(t)]: \\
 params &:= [a = -3.3521 \cdot 10^{-8}, b = -0.0000039589, c = 0.009948, d = 35.04, \rho = 1000, g = 9.81]: \\
 initialconditions &:= []:
 \end{aligned}$$

図6.33 遠心ポンプのカスタムコンポーネントの方程式

注: rhoFluid の値を Density パラメータに代入すると、Hydraulic Fluid Properties コンポーネントで定義された値が密度に割り当てられます。

垂直管

通常、メカニカル油圧システムでは重力ヘッドはそれほど重要ではないため、MapleSim の基本的なパイプコンポーネントではパイプの垂直移動がモデリングされません。圧力の低いシステムでは、重力ヘッドが重要になる場合があります。図6.34「重力ヘッドのカスタムコンポーネントの方程式」に、重力ヘッドをシミュレーションするためのカスタムコンポーネントの方程式を示します。

```
eq := [ dP(t) = rho · g · z, Pin(t) - Pout(t) = dP(t), Qin(t) = Qout(t), ]:  
params := [ z = 0, rho = 1000, g = 9.81 ]:  
initialconditions := [ ]:
```

図6.34 重力ヘッドのカスタムコンポーネントの方程式

第7章 リファレンス : MapleSim のキーボード ショートカット

モデルの開閉と保存

タスク	Windows と Linux	Mac
新規モデルを作成	[Ctrl] + [N]	[Command] + [N]
既存モデルを開く	[Ctrl] + [O]	[Command] + [O]
作業中のドキュメントを閉じる	[Ctrl] + [F4] (Windows) [Ctrl] + [W] (Linux)	[Command] + [W]
モデルを .msim ファイルとして保存	[Ctrl] + [S]	[Command] + [S]

ブロックダイアグラムでのモデルの作成

タスク	Windows と Linux	Mac
結線の描画中に操作をキャンセル	[Esc]	[Esc]
選択したモデリングコンポーネントを時計回りに90度回転	[Ctrl] + [R]	[Command] + [R]
選択したモデリングコンポーネントを反時計回りに90度回転	[Ctrl] + [L]	[Command] + [L]
選択したモデリングコンポーネントを左右反転	[Ctrl] + [H]	[Command] + [K]
選択したモデリングコンポーネントを上下反転	[Ctrl] + [F]	[Command] + [F]
選択したモデリングコンポーネントをサブシステム化	[Ctrl] + [G]	[Command] + [G]
選択した結線を自動整形	[Ctrl] + [D]	[Command] + [D]

タスク	Windows と Linux	Mac
すべての結線を自動整形	[Ctrl] + [A]、[Ctrl] + [D]	[Command] + [A]、 [Command] + [D]

ブロックダイアグラムでのモデルの表示

タスク	Windows と Linux	Mac
選択したモデリングコンポーネントまたはサブシステムの詳細を表示	[Ctrl] + [M]、または [Ctrl] + 下矢印	[Command] + [M]、または [Command] + 下矢印
現在のサブシステムの親レベルを表示	[Ctrl] + 上矢印	[Command] + 上矢印
Main に戻る	[Home]	[Home]
上位階層に移動	[Ctrl] + 上矢印	[Command] + 上矢印
モデルワークスペースを拡大表示	[Ctrl] + 数値キーの [+] キー、または [Ctrl] を押しながらマウスのホイールを前に回転させる	[Command] + 数値キーの [+] キー、または [Command] を押しながらマウスのホイールを前に回転させる
モデルワークスペースを縮小表示	[Ctrl] + 数値キーの [-] キー、または [Ctrl] を押しながらマウスのホイールを後ろに回転させる	[Command] + 数値キーの [-] キー、または [Command] を押しながらマウスのホイールを後ろに回転させる
モデルダイアグラムのサイズをモデルワークスペースに合わせる	[Ctrl] + [T]	[Command] + [T]
ズーム比をデフォルトのズーム比(100%)にリセットする	[Ctrl] + [0] (ゼロ)	[Command] + [0] (ゼロ)

3-D 表示でのモデルの表示

タスク	Windows と Linux	Mac
ポートを接続	[Ctrl] + [Shift] + [C]	[Command] + [Shift] + [C]
現在選択中のアイテムを削除	[Backspace] または [Delete]	[Delete]
表示ウィンドウに合わせて拡大/縮小	[V]	[V]
選択モード	[Esc]	[Esc]
モデルをパン	[F2]	[F2]
ズームモード	[F3]	[F3]
回転モード	[F4]	[F4]
遠近表示で、カメラを 3-D モデルの周囲に移動	[Ctrl] + マウスの左ボタンをクリックし、ドラッグする	[Command] + マウスをクリックし、ドラッグする
3-D モデルをパン	[Shift] + マウスの左ボタンをクリックし、ドラッグする	[Shift] + マウスをクリックし、ドラッグする
3-D ワークスペースを拡大または縮小	[Alt] + マウスの左ボタンをクリックし、ドラッグする。またはマウスホイールを前(拡大)または後ろ(縮小)に回転させる	[Alt] + マウスをクリックし、ドラッグする。またはマウスホイールを回転させる
遠近表示を変更し、X/Y/Z 軸の負の方向を表示	[X]	[X]
	[Y]	[Y]
	[Z]	[Z]
	[Shift] + [X]	[Shift] + [X]
	[Shift] + [Y]	[Shift] + [Y]

タスク	Windows と Linux	Mac
遠近表示を変更し、X/Y/Z 軸の正の方向を表示	[Shift] + [Z]	[Shift] + [Z]
遠近表示を変更し、シーンの右側を表示	[R] または [Ctrl] + [4]	[R] または [Command] + [4]
遠近表示を変更し、シーンの左側を表示	[L] または [Ctrl] + [3]	[L] または [Command] + [3]
遠近表示を変更し、シーンの上部を表示	[T] または [Ctrl] + [5]	[T] または [Command] + [5]
遠近表示を変更し、シーンの下部を表示	[M] または [Ctrl] + [6]	[M] または [Command] + [6]
遠近表示を変更し、シーンの前面を表示	[F] または [Ctrl] + [1]	[F] または [Command] + [1]
遠近表示を変更し、シーンの背面を表示	[B] または [Ctrl] + [2]	[B] または [Command] + [2]
現在のカメラ設定を保存 (カメラ設定は1つのみ保存可能)	[S]	[S]
最後に保存したカメラ設定に戻る	[H]	[H]
3-D 表示をリフレッシュ	[Shift] + [F5]	[Shift] + [F5]

モデルのシミュレーション

タスク	Windows と Linux	Mac
シミュレーションの実行	[F5]	[F5]
シミュレーション結果の表示。 シミュレーション結果が現在アクティブな場合は MapleSim ウィンドウを表示する	[F6]	[F6]
3-D ワークスペースの表示。 3-D ワークスペースが現在アクティブな場合は MapleSim ウィンドウを表示する	[F7]	[F7]
Apps マネージャを表示。 Apps マネージャが現在アクティブな場合は MapleSim ウィンドウを表示する	[F8]	[F8]
選択したコンポーネントまたは接続の有効/無効を切り替えて、次のシミュレーションから除外する	[Ctrl] + [E]	[Command] + [E]

コンソールペインのナビゲーション

タスク	Windows と Linux	Mac
セクションを展開 <ul style="list-style-type: none"> 右矢印キーを押してセクションを展開する 左矢印キーを押してセクションを折り畳む 下矢印キーを押して次のセクションに移動する 上矢印キーを押して前のセクションに移動する 	右矢印	右矢印
セクションを折り畳む	左矢印	左矢印
前または次のセクションにカーソルを移動	上下矢印	上下矢印

[プロットウィンドウ] のレイアウト変更

タスク	Windows と Linux	Mac
プロットウィンドウをタイル表示	[Shift] + [T]	[Shift] + [T]
プロットウィンドウをカスケード表示	[Shift] + [C]	[Shift] + [C]
プロットを[プロットウィンドウ]に合わせる	[Shift] + [F]	[Shift] + [F]
プロットの高さを自動的に調整して[プロットウィンドウ]の高さに合わせる		

Modelica カスタムコンポーネントの編集

タスク	Windows と Linux	Mac
Modelica コードエディタを表示。	[F9]	[F9]

タスク	Windows と Linux	Mac
Modelica コードエディタが現在アクティブな場合は MapleSim ウィンドウを表示する		
新規 Modelica カスタムコンポーネント	[Ctrl] + [N]	[Command] + [N]
Modelica カスタムコンポーネントを保存	[Ctrl] + [S]	[Command] + [S]
検索/置換	[Ctrl] + [F]	[Command] + [F]
行の移動	[Ctrl] + [G]	[Command] + [G]
部分的に入力したキーワードまたはコンポーネント名のオートコンプリート	[Ctrl] + スペースキー	[Command] + スペースキー
「block」、 <i>「if」</i> 、 <i>「ife」</i> 、または <i>「for」</i> などの用語入力後に Modelica 構文のテンプレートを挿入する。すべての用語の一覧については、 MapleSim 操作方法 > モデルの構築 > カスタムコンポーネント > Modelica コードエディタ > 構文のテンプレート から参照	[Ctrl] + [Shift] + スペースキー	[Command] + [Shift] + スペースキー
左側のナビゲーションペインの切り替え(ロックされていない場合)	[Ctrl] + [Tab]	[Command] + [Tab]

その他

タスク	Windows と Linux	Mac
検索を実行	[Alt] + [S]	[Alt] + [S]
MapleSim から Maple を起動する	[F10]	[F10]
パレットペインの表示切り替え(ロックされていない場合)	[Ctrl] + [Tab]	[Command] + [Tab]

2-D Math 表記のキーボードショートカットについては、**MapleSim 操作方法 > モデルの構築 > モデルへの注釈の追加 > 2-D Math 表記のキーの組み合わせ**を参照してください。

用語集

用語	説明
2-D Math 表記	上付き文字、下付き文字、ギリシャ文字などの数式テキストを入力するための書式オプションです。
3-D ワークスペース	3-D モデルを構築し、編集するための MapleSim ウィンドウ領域です。
添付形状	システムモデルの現実的な表現を作成するために 3-D モデルで表示可能な形状です。添付形状には、円柱、トレース線、別のファイルからインポートする CAD ジオメトリなどがあります。
カメラ	3-D を表示する際の視点です。
カメラ追跡	選択した対象 3-D コンポーネントの動きをカメラが追従するプロセスです。アニメーション中、対象コンポーネントは常に [3-D アニメーションウィンドウ] の中心に表示されます。
カスタムコンポーネント	MapleSim のカスタムコンポーネントテンプレートで作成したユーザ定義のカスタムコンポーネントです。
カスタムライブラリ	ユーザ定義のパレットに保存した今後の MapleSim セッションで使用可能なモデリングコンポーネントとサブシステムの集合です。
埋め込みコンポーネント	Maple の標準的なワークシートで解析、操作、可視化可能なグラフィカルコントロール、ボタン、メータなどの対話形式コンポーネントです。
インプリシットジオメトリ	モデリングコンポーネントを表現するために 3-D モデルで表示されるデフォルトの円柱および球です。
Maple パッケージ	Maple で使用可能なルーチンやコマンドの集合です。多くの Maple パッケージには、特定の数学領域や科学領域、または研究分野に特化した一連のコマンドが用意されています。
MapleSim コンポーネントライブラリ	MapleSim に用意されているドメイン固有のモデリングコンポーネントのデフォルト集合です。これらのモデリングコンポーネントは、 [ライブラリ] タブのグレーのパレットにあります。

用語	説明
モデルワークスペース	ブロックダイアグラムビューでモデルを構築し、編集するための MapleSim ウィンドウの領域です。
正投影図	平行に投影し、表示平面上の線を「実際の長さ」で表示する3-Dビューのタイプです。MapleSimでは、モデルの正面、上面、および側面からの正射影図を表示することができます。
遠近表示	3-D空間のすべての方向からモデルを調べることができる3-Dビューです。
プローブ	MapleSim モデルのシミュレーションを行うために対象の数量を特定するためのツールです。
共有サブシステム	別のサブシステムと同じ設定を共有するサブシステムのコピーです。すべての共有サブシステムは、設定を定義する特定のサブシステムの定義にリンクしています。
スタンドアロンサブシステム	サブシステムの定義にリンクしておらず、モデル内の他のサブシステムとは無関係に編集や操作を行うことのできるサブシステムです。
サブシステム	単一ブロックにグループ化されたモデリングコンポーネントの集合です。
サブシステムの定義	一連の共有サブシステムの設定を定義するサブシステムブロックです。

索引

シンボル

- 2-D math notation, 72
- 2-D Math 表記, 72
- 3-D アニメーションウィンドウ, 128
- 3-D 再生時間, 129
- 3-D サンプリングレート, 129
- 3-D 動画, 152
- 3-D 表示
 - 直交, 133
- 3-D 表示
 - 遠近, 133
- 3-D 表示のコントロール
 - インプリシットジオメトリ, 135
 - 初期条件, 151
 - 添付形状, 136, 143
 - トレースの追加, 137
- 3-D 表示のナビゲーション, 133
- 3-D モデルの構築, 140
- 3-D モデルの動画を再生する, 152
- 3-D ワークスペース, 132
 - 軸の指定, 133
- アルファ, 113
- 因果的モデリング, 2, 6, 9
- インプリシットジオメトリ, 135
- 埋め込みコンポーネント, 178
- 運動学的拘束, 142
- 介在変数, 3
 - カスタムコンポーネント, 93
- 外部 C コード/DLL
 - カスタムコンポーネント, 177, 245
- 可視化, 128
 - 透明度, 128
- 可視化のパラメータ
 - 設定, 129
- カスタムコンポーネント
 - 外挿データからのモデリング, 233
 - 外部 C コード/DLL, 177, 245
 - カスタムコンポーネントについて, 85
 - テンプレート, 239
 - 編集, 96
 - 方程式の定義, 98
 - ポートの定義, 99
- カスタムプロットウィンドウ, 121
- カスタムライブラリ, 66
- 可変時間ステップ, 110
- キーボードショートカット, 287
- 許容誤差, 111
- グリッド
 - 3-D グリッド, 143
 - CAD ジオメトリの使用, 143
- グローバルパラメータ, 185
- 結線, 27
 - 色, 28
- 拘束安定化, 113
- 拘束条件に対する射影法, 114, 170
 - イベント処理, 114
 - 許容値, 114
 - 収束計算, 114
- 拘束条件の計算オプション, 170
- コード生成
 - C コード, 164
 - サブシステム, 165
 - 初期化, 167
- 固定時間ステップ, 110
- コンストラクトモード, 140

- コンポーネントプロパティの指定, 13
- サブシステム
 - 共有, 36
 - コード生成, 165
 - 作成と管理, 32, 184
 - 定義, 36
 - パラメータ, 49
 - 複数インスタンスの編集, 39
 - ポートの追加, 191
 - モデルにサブシステム定義と共有サブシステムを追加する, 37
 - モデルにサブシステムのコピーを複数追加, 35
 - リンク, 36, 44
- 時間
 - シミュレーションの終了, 113
- 時間ステップ, 110
- シミュレーション, 115
 - 初期条件, 120
 - 設定, 110
- シミュレーションオプション設定, 112
- シミュレーショングラフ, 127
- シミュレーション結果
 - 管理, 119
 - グラフデータのエクスポート, 121
 - 進捗状況メッセージ, 117
 - スナップショット, 120
 - 比較, 119
 - 表示, 119
 - 保存, 119
 - メッセージコンソールの消去, 119
- シミュレーションのパラメータ
 - Compiler, 116
 - 設定, 109
 - 状態, 120
 - 初期化, 167
 - 初期条件, 60
 - オーバーライド, 64
 - 指定, 30
 - 初期条件の強制のベストプラクティス, 81
 - 初期条件の強制方法の指定, 31
 - 信号フロー, 91
 - 診断メッセージ, 8, 76
 - 進捗の情報メッセージ, 118
 - スタンドアロンサブシステム, 44
 - スナップショット, 120
 - 使用, 113
 - 接続ポート, 27
 - 設定
 - シミュレーション, 110
 - シミュレーションオプション, 112
 - 線形化, 161
 - 線形系
 - 解析, 161
 - 単位
 - パラメータ単位の指定, 29
 - 注釈, 70
 - チュートリアル, 181
 - 基本チュートリアル: RLC 回路と DC Motor のモデリング, 19
 - 通過変数, 3
 - カスタムコンポーネント, 93
 - 矢印方向の規則, 18, 79, 109
 - データセット
 - Maple で作成, 74
 - モデルの生成, 66
 - デバッグコンソール, 45

- 添付パレット, 66
- テンプレート, 159
 - カスタムコンポーネント, 98
 - カスタムコンポーネントについて, 87
- トレース線, 128, 136
 - 例, 137, 139
- 流れ方向, 3
- パラメータ
 - グローバルパラメータ, 47
 - サブシステムパラメータ, 49
 - パラメータ詳細設定, 50, 58
 - パラメータセット, 57, 117
 - パラメータ値, 13
 - 変数詳細設定, 58
- パラメータオーバーライド, 60
- パラメータの最適化, 162
- パラメータブロック, 51
- パレット, 8, 21
- 非因果的マッピング, 92
- 非因果的モデリング, 2, 3, 6, 9
- 微分代数方程式, 2
- 描画, 70
- 符号の規則, 3
- 物理コンポーネント, 93
- 物理モデル
 - 解析, 155
 - ナビゲーション, 25
- プローブ, 108
 - 追加, 14
 - 矢印方向の規則, 18, 79, 109
- プローブの追加, 14
- プローブパレット, 116
- プロット
 - 2つ目の変数をグラフに追加する, 122
 - X 軸変数の設定, 125
- プロットウィンドウ, 127
- ベータ, 113
- ベストプラクティス, 81
 - 1-D 並進モデルの作成, 78
 - サブシステムの作成と配置, 75
 - 初期条件の強制, 81
 - 電気モデルの作成, 76
 - マルチボディモデルの作成, 79
 - モデルのシミュレーションと可視化, 153
 - 油圧モデルの作成, 80
- ヘルプペイン, 22
- 変数のスケージング, 115
- 方程式
 - App, 251
 - 取得, 160
- ポートとパラメータの管理, 168
- 補間テーブル, 194
- 保存結果
 - シミュレーションから, 119
- 保存量のフロー
 - 矢印方向の規則, 18, 79, 109
- マルチボディ
 - 設定, 129
 - マルチボディモデルを作成するためのベストプラクティス, 79
- マルチボディのパラメータ値, 130
- モデリングコンポーネント
 - 接続, 12
- モデル
 - 作成, 10
- モデルツリー, 23
- Apps とテンプレート

- API の使用, 178
 - モデルの解析で使用, 159
 - モデルの構築
 - 3-D モデルのアセンブル, 142
 - 3-D モデル構築中に添付形状を表示する, 143
 - 3-D ワークスペースでのオブジェクトの移動, 141
 - 3-D ワークスペースでのオブジェクトの追加と移動, 144
 - 拘束解除ボタンの使用, 142
 - モデルへのファイルの添付, 65
 - モデルワークスペース, 7
 - 矢印方向の規則, 76, 78
 - 油圧システム, 285
 - Bernoulli の式, 256
 - Darcy の式, 256
 - Joukowsky の式, 280
 - 圧縮率, 271
 - ウォーターハンマー, 274
 - カスタムアキュムレータコンポーネント, 281
 - カスタムコンポーネント, 283
 - 基本的な油圧方程式, 256
 - 基本的な油圧ライブラリコンポーネント, 254, 258
 - スプール弁, 262
 - パスカルの原理, 271
 - 並進運動, 266
 - 摩擦係数, 257
 - マルチドメイン, 263
 - マルチボディの油圧, 268
 - 流体慣性, 273
 - 例
 - 1 つの物理量を別の物理量に対してプロットする, 125
 - 3-D ワークスペースでの二重振り子モデルの作成, 144
 - Maple でのデータセットの作成, 74
 - 既存モデルからカスタムライブラリを作成, 67
 - 共有サブシステムとそのサブシステム定義とのあいだのリンクの解除, 43
 - 共有サブシステムへのサブシステムパラメータの代入, 50
 - グローバルパラメータの定義と代入, 47
 - サブシステムの作成, 33
 - スタンドアロンサブシステムのコピーと貼り付け, 46
 - デバッグコンソールに表示される警告メッセージを処理する, 45
 - 同一サブシステム定義にリンクしている共有サブシステムを編集する, 39
 - 二重振り子のモデルへの添付形状の追加, 137
 - パラメータオーバーライドの作成, 60
 - パラメータブロックの作成と使用, 52
 - 非線形バネダンパカスタムコンポーネント, 103
 - 複数の物理量を個別のグラフにプロットする, 122
 - モデルへのテキスト注釈の追加, 70
- A**
- Annotations, 70
 - API, 178
 - Apps

コード生成, 164
パラメータの最適化, 162
方程式の抽出, 160
Arrow convention, 76, 78
Attaching Files to a Model, 65
Attachments palette, 66

B

Baumgarte, 113
アルファ, 113
ベータ, 113
Best Practices, 81
Building 1-D Translational Models, 78
Building Electrical Models, 76
Building Hydraulic Models, 80
Building Multibody Models, 79
Enforcing Initial Conditions, 81
Laying Out and Creating Subsystems, 75

C

CAD ジオメトリ, 143
Compile Optimized, 116
Compiler, 116
Connection lines, 27
Colors, 28
Connection ports, 27
Conserved Quantity Flow
arrow convention, 79
Custom libraries, 66

D

DAE 変数, 248

Data sets
creating in Maple, 74
generating for model, 66
Debugging console, 45
Diagnostic Messages, 76
DLL
カスタムコンポーネント, 177
Drawing, 70
DynamicSystems パッケージ, 178

E

EMI コンポーネントオプション, 170
Event Hysteresis, 114
Event Iterations, 114
Event Projection, 114
Examples
Adding Text Annotation to a Model, 70
Assigning a Subsystem Parameter to a Shared Subsystem, 50
Copying and Pasting a Standalone Subsystem, 46
Creating a Data Set in Maple, 74
Creating a Parameter Override, 60
Creating a Subsystem, 33
Creating and Using a Parameter Block, 52
Defining and Assigning a Global Parameter, 47
Removing the Link Between a Shared Subsystem and its Subsystem Definition, 43
Resolving Warning Messages in the Debugging Console, 45

H

Help pane, 22

I

Index 1 Error Control, 114

Index 1 Tolerance, 115

Initial conditions, 60

- Best practices for enforcing initial conditions, 81

- overrides, 64

- specifying, 30

- Specifying How Initial Conditions Are Enforced, 31

Initial Hysteresis, 114

J

Jacobian, 113

L

Latest results

- シミュレーションから, 119

LinkModel, 178

M

MapleSim component library, 21

MapleSim ウィンドウ, 7

MapleSim コンポーネントライブラリ, 6, 9, 21

MapleSim モデル

- 埋め込みコンポーネント, 178

Minimize Events, 115

Model tree, 23

Modelica, 105

Modelica カスタムコンポーネント, 86

Multibody

- Best Practices for building multibody models, 79

P

Palettes, 21

Parameter block, 51

Parameter Override, 60

Parameters

- Advanced Parameter Settings, 50, 58

- Advanced Variable Settings, 58

- Global parameters, 47

- Parameter sets, 57

- Parameter values, 29

- Subsystem parameters, 49

Physical models

- Navigating, 25

Plot Events, 115

Plot Points, 112

Probes

- arrow convention, 79

Projection, 114

Projection Iterations, 114

Projection Tolerance, 114

S

Scaling, 115

Simulation

- Duration time, 110

- Start time, 113

Solver, 111

Solver Diagnostics, 116

Solver Type, 110

Standalone subsystem, 44

Step size, 112

Subsystem(s)

- Adding Multiple Copies of a Subsystem to a Model, 35

- Adding Subsystem Definitions and Shared Subsystems to a Model, 37

- Creating and Managing, 32
- definition, 36

- Editing multiple instances, 39

- linking, 36

- parameters, 49

- shared, 36

- standalone, 44

T

Through Variables

- arrow convention, 79

Time

- Simulation duration, 110

- Simulation start, 113

U

Units

- Specifying Parameter Units, 29

